



U.S. DEPARTMENT OF
ENERGY

PNNL-19863

Prepared for the U.S. Department of Energy
under Contract DE-AC05-76RL01830

TWINS Software Description

Revision 3

SJ Harris
TS Olund
JY Houchin
AJ Weber
IH Wright

September 2010



Pacific Northwest
NATIONAL LABORATORY

DISCLAIMER

United States Government. Neither the United States Government nor any agency thereof, nor Battelle Memorial Institute, nor any of their employees, makes **any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights.** Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or Battelle Memorial Institute. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

PACIFIC NORTHWEST NATIONAL LABORATORY
operated by
BATTELLE
for the
UNITED STATES DEPARTMENT OF ENERGY
under Contract DE-AC05-76RL01830

Printed in the United States of America

Available to DOE and DOE contractors from the
Office of Scientific and Technical Information,
P.O. Box 62, Oak Ridge, TN 37831-0062;
ph: (865) 576-8401
fax: (865) 576-5728
email: reports@adonis.osti.gov

Available to the public from the National Technical Information Service,
U.S. Department of Commerce, 5285 Port Royal Rd., Springfield, VA 22161
ph: (800) 553-6847
fax: (703) 605-6900
email: orders@ntis.fedworld.gov
online ordering: <http://www.ntis.gov/ordering.htm>



This document was printed on recycled paper.
(9/2003)

TWINS Software Description

Revision 3

SJ Harris
TS Olund
JY Houchin
AJ Weber
IH Wright

September 2010

Prepared for
Washington River Protection Solutions

Computational & Statistical Analytics
Pacific Northwest National Laboratory
Richland, Washington 99352

Acronyms and Abbreviations

ASP	Active Server Pages
AUTOSTAT	Automated Statistics
AUTOVECTOR	Automated Vector
BBI	Best-Basis Inventory
BBIM	Best Basis Inventory Maintenance
BDGRE	Buoyant Displacement Gas Release
FY	fiscal year
GRE	Gas Release
HEIS	Hanford Environmental Information Systems
HTML	Hyper-Text Markup Language
LFL	lower flammability limit
ODBC	database via open database connectivity
PC	personal computer
PNNL	Pacific Northwest National Laboratory
SACS	Surveillance Analysis Computer System
SEF	Standard Electronic Format
SEFX	Standard Electronic Format/Excel
SSIS	SQL Server Integration Services
TCD	Tank Characterization Database
TCR	Tank Characterization Report
TPA	Tri-Party Agreement
TVD	Tank Vapor Database
TWINS	Tank Waste Information Network System
WHC	Westinghouse Company
WRPS	Washington River Protection Solutions
WWW	World Wide Web

Contents

Acronyms and Abbreviations	iii
1.0 Introduction and Purpose.....	1.1
2.0 General Description.....	2.1
3.0 TWINS Server Configuration.....	3.1
4.0 Characterization Domain Core Software.....	4.1
4.1 Overview of Characterization Toolset	4.1
4.2 Databases.....	4.2
4.2.1 TCD.....	4.2
4.2.2 BBIM and BB_Published.....	4.5
4.2.3 TVD.....	4.6
4.3 Reporting Tools.....	4.10
4.3.1 TWINS Web Interface	4.10
4.3.2 TWINS Menu System	4.12
4.4 Support Applications.....	4.13
4.4.1 AUTOSTAT Tool	4.13
4.4.2 AUTOVECTOR Tool	4.15
4.4.3 TXFR Tool	4.16
4.4.4 Resolve Tool	4.17
5.0 References	5.1
Appendix A.....	A.1

Figures

Figure 3.1. TWINS Production Server Configuration	3.1
Figure 4.1. Characterization Data Development, Interpretation, and Reporting	4.1
Figure 4.2. BBIM and BB_Published	4.6
Figure 4.3. Tank Vapor Database	4.9
Figure 4.4. TWINS Web Interface Architecture	4.11
Figure 4.5. TWINS Menu System	4.13
Figure 4.6. AUTOSTAT Context Diagram	4.14
Figure 4.7. AUTOVECTOR Context Diagram	4.15
Figure 4.8. TFXR Tool Context Diagram	4.16
Figure 4.9. Resolve Data Review and Disposition Setting Tool Context Diagram	4.17

1.0 Introduction and Purpose

The Tank Waste Information Network System (TWINS) comprises many components of software and hardware, as well as procedures, all of which have been designed to facilitate the tasks of managing tank waste data, analyzing the data, preparing reports, and making decisions to safely manage the tank wastes. Over time, tools have been added to provide automation throughout the process of tank characterization and related reporting (1998-2010). TWINS contains a comprehensive set of databases and tools that are being used to characterize the Hanford tank wastes.

The purpose of this document is to provide a single source description of TWINS. TWINS consists of a number of primary databases, reporting tools, and support tools, all of which were created to support the “characterization” or “process engineering” needs. These database systems consist of Tank Characterization Database (TCD), Tank Vapor Database (TVD), TWINS, the Best Basis Inventory Maintenance (BBIM) tool, the TXFR tools, the Automated Statistics (AUTOSTAT) tool, and the Automated Vector (AUTOVECTOR) creation tool. This suite of software was developed primarily to support the customer organization responsible for characterizing the tank wastes and comprises the TWINS “core” software.

These systems have been developed by several different projects over a number of years. This document contains a mixture of technical and non-technical information; it is intended to be a first point of embarkation for any person requiring orientation to the projects. Technical material includes summary technical descriptions of the databases and tools, its functionality, and its software architecture. Non-technical subjects include a project background summary and descriptions of the project management and systems engineering controls and procedures governing the project.

2.0 General Description

The TCD is the official database of data and information describing the chemical and radiochemical contents of the 177 high-level waste tanks at Hanford. The TCD was implemented in support of the TWRS program initiative to remediate the high-level waste tanks at the Hanford Site. The TCD is the “electronic database” referred to by the Tri-Party Agreement (TPA) in Milestone 44 and Section 9.6. The TVD is the official database for the vapor phase of these tanks. These data for TCD and TVD are stored in Microsoft® SQLServer relational databases managed by Pacific Northwest National Laboratory (PNNL) and physically located in the ISB2 building at PNNL. During fiscal year (FY) 1996, the TCD project subsumed the TWINS project, which had been dropped from DOE Headquarters funding. TWINS property and capabilities were applied exclusively to Hanford TWRS Characterization needs for the purpose of expanding capabilities and reducing costs. During FY 1997, TVD became part of the TCD Project in efforts to cut costs.

The TCD and TVD are accessible to the general user across the Hanford Intranet via the TWINS interface and to selected addresses on the open internet. The TWINS interface is a graphical user interface that was developed to replace the original TWINS user interface to provide access to TCD and the Vapor database. However, because of the “TWINS Middleware” access layer (described later in this document), it is capable of providing effective access to many varieties of network accessible databases. The Surveillance Analysis Computer System (SACS) is also accessible through TWINS.

The TWINS user interface is “data driven” and is easily configured to provide access to any database table, and to specify which fields are queriable in the user interface. The TWINS user interface is displayed on the user’s personal computer (PC) by the Internet Explorer World Wide Web (WWW) network browser. The default viewer recommended to the Hanford Site users is Microsoft® Excel, which is universally available. The TCD data records are returned by the TWINS interface in a “text/tab-separated-values” format, which, in the standard Hanford Site PC configuration, is automatically displayed in the Microsoft® Excel spreadsheet utility. There is also an online User Guide to help the user with the application. This guide also allows access to an online data dictionary.

In addition to accessing data from database tables, the TWINS user interface provides access to Tank Reports that have been stored on a project server. These documents are maintained by the customer (WRPS).

The Tank Characterization Report (TCR) authors represent another class of TCD users having unique needs. A set of “standard reports” has been developed to assist the TCR authors in producing their deliverable TCR reports, while at the same time ensuring that the TCD data was used in the report, rather than another parallel data source.

A major part of TCD is the data loading operation. This consists of personnel, software, and procedures. Data are received from data generators, which include the Hanford Site analytical laboratories, 222-S Laboratory and 325 Laboratory, and various sources of waste inventory data. Most of the analytical data being received is now in a format compatible with the TCD loaders. However, some software to facilitate the preparation of manual data for loading is also maintained. In addition, a “data

maintenance” user interface exists to perform database corrections and manual updates. These operations—data loading and data maintenance—are governed by formal procedures designed to promote data integrity and documentation.

Data loading is also a major part of TVD. Currently, the analytical data are received from the Hanford Site contractors who furnish their data in electronic files. Software and procedures are maintained for the automated loading and processing of all TVD data.

The BBIM is a SQL Server database and set of .NET ASPX data maintenance tools used to maintain the Best-Basis Inventory (BBI) inventories at the Hanford Site. The BBIM system accepts measured data as inputs (analyte concentrations, densities, and waste volumes) and calculates the official site inventories according to the BBI calculation protocols. The BBIM database is a “working” area that is accessible only to the staff responsible for maintaining the BBI inventories. After BBI staff have reviewed and approved the data, they may “publish” individual tank data. When published, the information created for a given tank is copied from the working area to an area where the public can view it. That area is another database called BB_PUBLISHED. There are several TWINS web queries that access this data.

3.0 TWINS Server Configuration

TWINS has several different “faces,” and a single diagram representing the complete functionality is impractical. In order to understand the various “architecture” diagrams that are presented in this document, it will be helpful to the reader to have some feel for the hardware and software configurations, what the names of the production servers are, and what their basic functional assignments are.

Figure 3.1 depicts the configuration of TWINS production servers, the software and hardware configurations, and the key databases resident there.

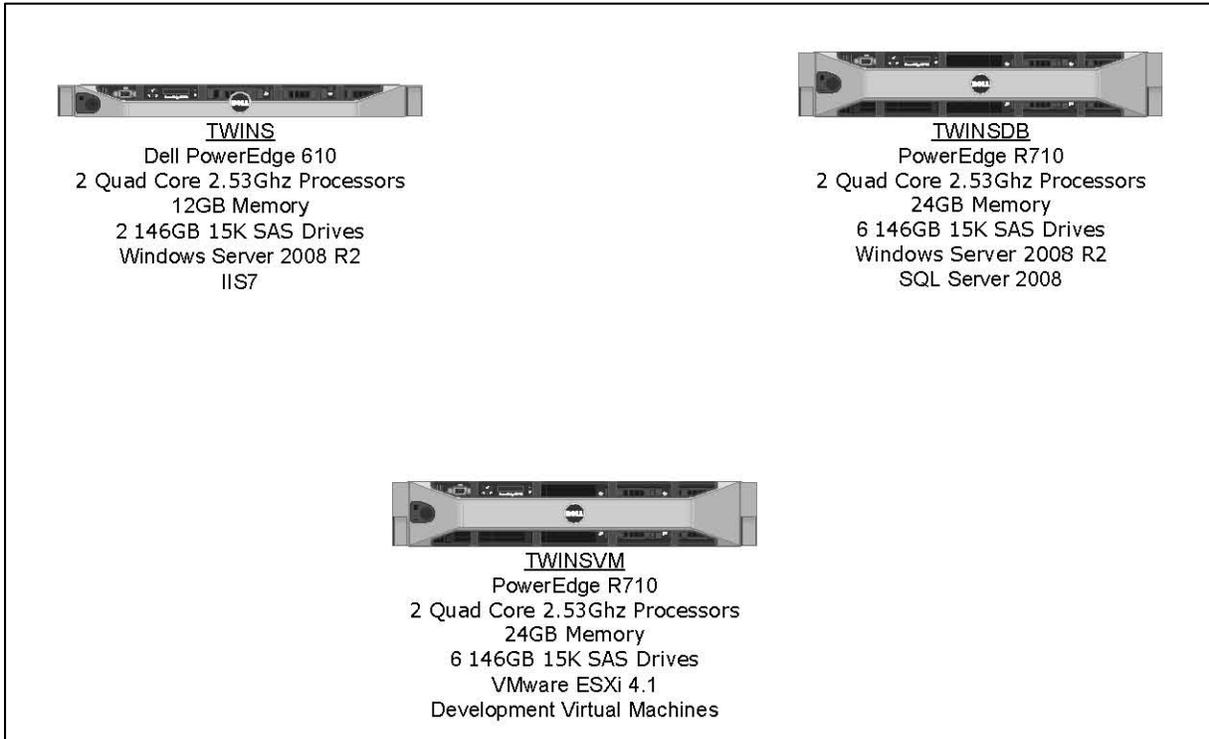


Figure 3.1. TWINS Production Server Configuration

Functionally, TWINS contains mainly the “operational databases,” where external data is being loaded or entered into the databases. Real-time updates are occurring. Early on in the evolution of TWINS, the technique of “nightly processing” was used to process, check, and transform raw data into user views and reports. Originally run in the UNIX environment as “cron” jobs, this processing now runs as SQL Server Integration Services (SSIS) packages.

TWINS also runs the TWINS web server and hosts the code that drives the “TWINS web interface” inside the Hanford firewall at <http://twins.pnl.gov/>. TWINSWEB is in the PNNL extranet and provides controlled access to the “TWINS web interface” to authorized internet users at <http://twinsweb.pnl.gov/>.

Over the 17-year life span of TWINS, tradeoffs were constantly being made as to whether replacement or upgrading of databases and/or applications was justified based on many factors. As a result, the TWINS “architecture” uses technologies from different eras of web computing: Active Server

Pages (ASP) and .NET. It would be difficult and confusing to try to depict this in a single “architecture” diagram. For this reason, a series of diagrams is presented throughout the remainder of this document.

In addition to the “Production” environment, there is a TWINS software development environment that includes a “Development” area where initial software development and testing are done.

4.0 Characterization Domain Core Software

4.1 Overview of Characterization Toolset

Starting with a single relational database, TCD, in 1993, and primarily existing to satisfy the TPA, the TWINS characterization suite of tools grew and expanded in scope over the years as the problems were better understood, as computer technologies increased in power, and as budget reductions dictated the need to do more with less. It will be helpful to see a single view of this set of components prior to reading descriptions of the individual parts. Figure 4.1 depicts most of the databases and tools that will be described in this section.

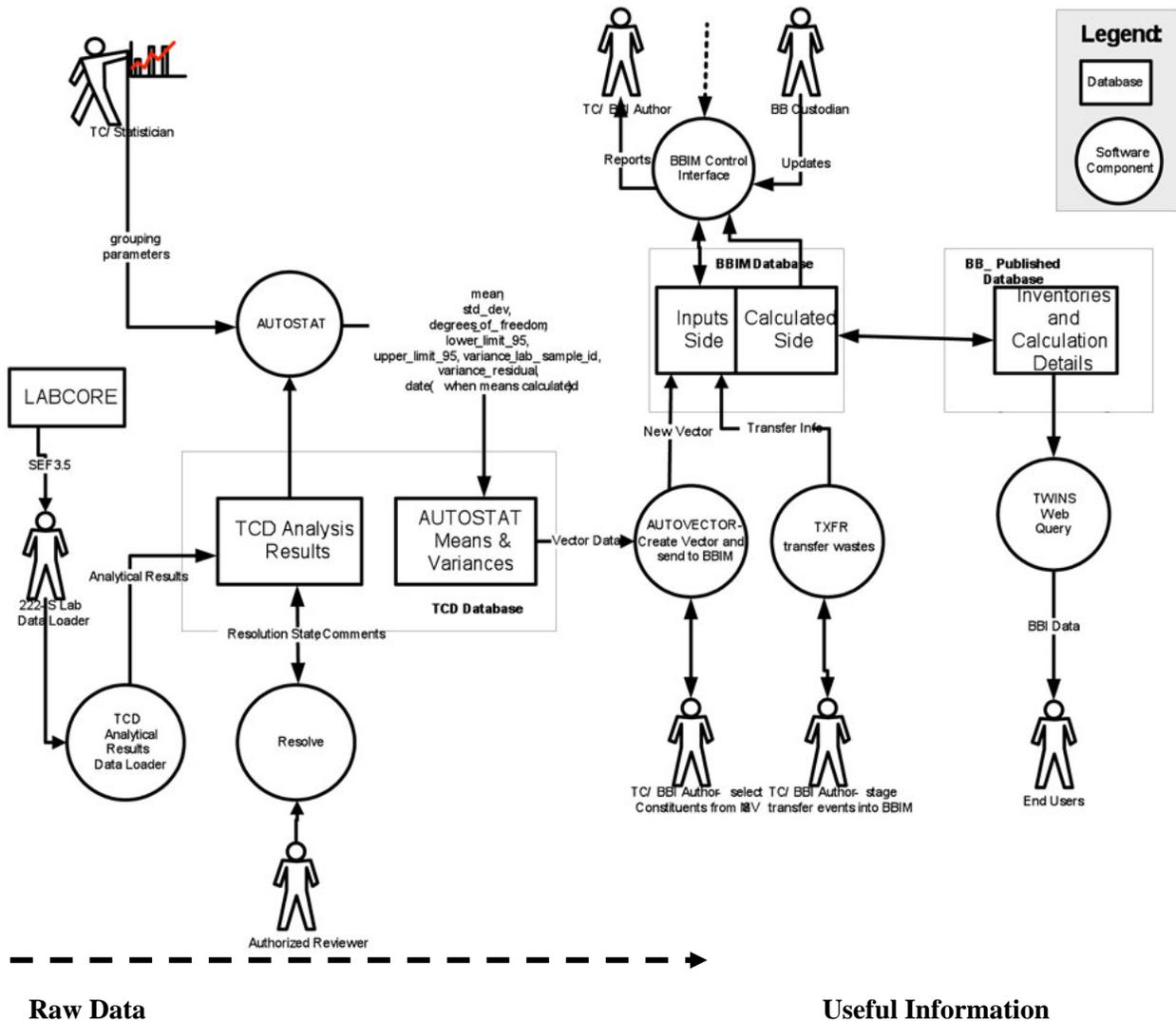


Figure 4.1. Characterization Data Development, Interpretation, and Reporting

Because of the sheer volume of data, the continuously changing nature of the tank waste itself, and the ongoing sampling and data development, good data management practices and automation wherever feasible are required to inspire confidence in the data. In Figure 4.1, all of the boxes and circles (except “LABCORE”) represent databases or tools that are components of TWINS. Together, they facilitate the acquisition of raw data (from LABCORE and other sources), the review of the raw data by both humans and automated checks, the statistical analysis and characterization of the raw data, the selection and development of Best-Basis Inventories, and the reporting of data to the user community.

4.2 Databases

4.2.1 TCD

4.2.1.1 Purpose and Content

Analytical Data are complex and require a background in analytical chemistry to begin to understand. Also, because of the complex layering and inhomogeneous nature of the waste, the concentrations expressed by the Analytical Data are widely variant; therefore, a statistical interpretation of the data is required.

Analytical Data are the data generated by the analytical laboratories as a result of performing analytical procedures to determine the concentration of chemical analytes and radiochemicals as requested by the requestor of the analytical services. Analytical data are either “TPA” data or historical analytical data. The TPA data were generated by the TWRS program, using TPA approved laboratory protocols from 1989 onward. Historical data are data from letter reports and various formats generated on the Hanford Site over the past 20 to 30 years.

4.2.1.2 History

The TCD project was initiated at PNNL in 1992 with requirements gathering and systems analysis and design. Preliminary requirements for TCD were furnished by Westinghouse Hanford Company (WHC) in the form of a requirements specification, *Tank Characterization Database Software Requirements Specification*¹. Using this document as a start, PNNL designed a data model for the integration of TCD as a Subject Area within the Hanford Environmental Information Systems (HEIS) database architecture. This initial design for TCD is documented in *Tank Characterization Data Integration with HEIS Software Design Description SDD*².

In its initial implementation in 1993, the TCD database was physically located on the HEIS Sequent computer system, under the Oracle Version 6 Relational Database Management System. The TCD received the first laboratory data load in 1993, meeting *Hanford Federal Facility Agreement and Consent Order* (TPA) Milestone M-44. The TCD continued to operate in this environment throughout FY 1994 and into FY 1995.

In the HEIS software environment, the TCD user interface was a character-based interface that was designed to support data entry and not data extraction; this interface is documented in the *Hanford Environmental Information System (HEIS) Volume 8, Tank Characterization Database Subject Area*³. The HEIS user interface was very expensive to maintain because the data validation logic was built into

the user interface. Therefore, in early FY 1995, the TCD project adopted the standard TWINS user interface as the only user interface for data access. In this same time frame, Bechtel took over the Environmental Restoration program and the HEIS database system. For this reason, it was decided to migrate the TCD database to Sybase 4.9.2 on the existing TWINS server. This major effort was completed by August 1995.

The original purpose of TCD was to be a managed repository of characterization data, serving the Hanford Site for engineering analysis and decision-making purposes. However, for a number of reasons, it had not fulfilled that purpose by 1995. It remained an active database in FY 1994 and FY 1995 primarily because of the TPA mandate for an “electronic database” of tank characterization data. Many of the problems that prohibited TCD from fulfilling its originally intended purpose pertained to the TWRS Characterization program as a whole, and WHC commissioned a task team in early 1995 to study the situation and make recommendations. The team’s findings are documented in their final report, *Tank Characterization Data Management Strategy*⁴.

The team found that TCD was under-utilized for several reasons, including questionable data quality, untimely availability of data, and cumbersome access methods. Also, TCD was an incomplete data set; other data sets or databases had been created within other areas of TWRS by organizations to fulfill their own needs. As a result, TCD was not viewed as a useful tool for analytical and/or site operational purposes. Perhaps the most important conclusion was that WHC needed to assume direct ownership of the database; PNNL would continue to provide technical services.

At that time (early 1995), the biggest single issue adversely affecting data quality and timely availability of data was that the laboratory data was being manually loaded into TCD, which was a time-consuming, error-prone process. The strategy team recommended that a planned electronic interface be put into operation on an expedited basis. This was done by June 1995 and was the first of several key pieces that needed to be in place before TCD could become a truly useful component of the TWRS Characterization infrastructure.

In 1998, TCD was ported to run under Microsoft® SQLServer 6.5 and the NT 4.0 operating system. At this same time, the TVD was also ported to this environment, thus consolidating operations and providing a secure base for current software development and future operations.

The TWINS suite of databases and applications are currently running on Microsoft® SQLServer 2008 and the Windows Server 2003 operating system.

4.2.1.3 Data Maintenance Tools and Interfaces

The TCD is a database on the TWINS server running under the Microsoft® SQLServer system. The TCD is a collection of related database tables, along with associated database logic that performs a variety of functions such as data input checking, consistency checking across tables, and data loading functions. The TCD tables feed the Sample Analysis “Subject Areas” of the TWINS web interface. The TCD is physically located on the TWINS Windows server located in the Information Sciences Building 2 (ISB2) at PNNL.

In addition to the database, an extensive set of tools has been developed to support the loading of several types of data that are managed by the TCD system.

4.2.1.3.1 Standard Electronic Format

In 1993, PNNL issued the initial description of an electronic data format to be used for transmitting laboratory data to the TCD. The current version of this specification is Standard Electronic Format (SEF) 3.5, which is described in *Standard Electronic Format Specification for Tank Characterization Data Loader, Version 3.5*¹⁰. The system produces this format as a report, which is then delivered to the TCD Data Loading staff for loading into TCD within the TPA-specified time limit.

In 2001, PNNL designed a new format for laboratory data reporting, known as SEFX (*Standard Electronic Format/Excel (SEFX) Specification for Tank Characterization Data Loader: Version 1.0*¹¹). The SEFX format is designed to provide a format for laboratories that are not equipped to produce the SEF 3.5 (or latest version). The SEFX format is a format designed around Microsoft® Excel workbooks and worksheets, and is less prescriptive and rigid in what it will accept than the strictly electronic SEF 3.5 format, which is designed to be created by a laboratory information management system. SEFX, on the other hand, is designed to be loaded by a person at the TCD end, and can accept more variation in the format of the “metadata” associated with the analytical results.

4.2.1.3.2 TCD Data Loader

The TCD Data Loader program is an ASP.NET application. Its purposes are to load the SEF 3.5 and non-SEF format into TCD and to allow manual corrections to TCD data. It provides a user interface for input and output of data, error, and reject files. It interacts with the TCD database via open database connectivity (ODBC) to exercise the various rules, triggers, and stored procedures associated with loading the database tables. Data for the loader come from LABCORE.

The five major functions that the data loader performs are:

- (1) Data Loading – this function processes the input file and attempts to load all records. All errors are printed to an error report and a reject file. All records with no errors are loaded into the appropriate database tables.
- (2) Data Verifying – this function compares the original input file with the results in the database tables. Any discrepancies or missing data are identified in an error report and corrective action must be taken.
- (3) Data Acceptance or Rejection – if many data errors are detected during data verification, the entire batch of data may be rejected and backed out of the loading process.
- (4) Data Upleveling – this function uses the original input file to determine what records are to be upleveled. The access level on these records is modified to the appropriate value.
- (5) Data Populating – this function uses the data in the *constituent_inventory* table to populate the *inventory_summary* table.

There are two major categories of data that are loaded:

- (1) Samples – this includes core data, segment data, supernatant sample data, surface sample data, sample descriptions, and sample relationships.
- (2) Analyses – this includes analysis data and analysis results data.

The requirements and design for this application are documented in *Requirements and Design for the Integrated Tank Characterization Database Data Loader (Visual Basic/SQLServer)*¹² and the user documentation can be found in *Procedures for Loading Analysis Result Data into the Tank Characterization Database (TCD)*¹³.

4.2.2 BBIM and BB_Published

4.2.2.1 Purpose and Content

Inventory data are simply the mass of a given analyte in each tank, or tank waste phase, expressed in either kilograms (kg) for chemicals, or curies (Ci) for radiochemicals. Inventory data are developed in the BBIM system and are derived from several sources:

Source	Description
Sample	Data are derived from the Analytical Data following a chemical and statistical analysis of the data to determine the most appropriate concentration (among possibly several different samples and methods) to use.
Template Sample	Data are derived from sample data obtained from the same waste type found in similar tanks.
Template Engineering	Data are calculated by a complex Excel® spreadsheet model (Hanford Defined Waste Rev. 5) based upon historical transfer records.
Process Knowledge	Data are based upon engineering estimates that can be derived from waste transfer data, knowledge of composition, older sample data (pre 1990), or any specific knowledge that can be used to determine the inventory.
Calculated	Data are derived using specific rules that use concentrations of other analytes found in the tank.

4.2.2.2 History

The Best-Basis Inventory Maintenance tool, or BBIM, was initially developed in FY 1999. The justification for developing it came from the realization that it was an impossible task to keep 177 paper documents up to date in the face of tank farm operations and the availability of new sample data, not to mention the discovery of mistakes in the existing calculation bases. In FY00, the BBIM went into production, becoming the official source for the BBI inventory numbers. The BBIM is a working area (a “sandbox”) where the inventory numbers are created through a defined process of analysis. After data have been modified and thoroughly reviewed, tanks are “published” to the BB_PUBLISHED area. In BB_PUBLISHED, the data are accessible by TWINS.

4.2.2.3 Data Maintenance Tools and Interfaces

The BBIM database is a “working area” (Figure 4.2) where tank coordinators and other expert reviewers collaborate to develop, review, and finally “publish” inventory data. The BBIM is a relational

database but can be thought of as consisting of an “Inputs” side and a “Calculated Values” side. Inputs are the raw measurements (analyte concentration, analysis date, density, waste volume) needed to calculate inventory. The BBIM “configuration control” function provides the ability for the BBI custodian to publish tanks. Publishing means moving the calculated inventory values and associated metadata to a publicly accessible area on TWINS. The database provided for this purpose is called BB_PUBLISHED. In terms of structures and complexity, it is a subset of the BBIM.

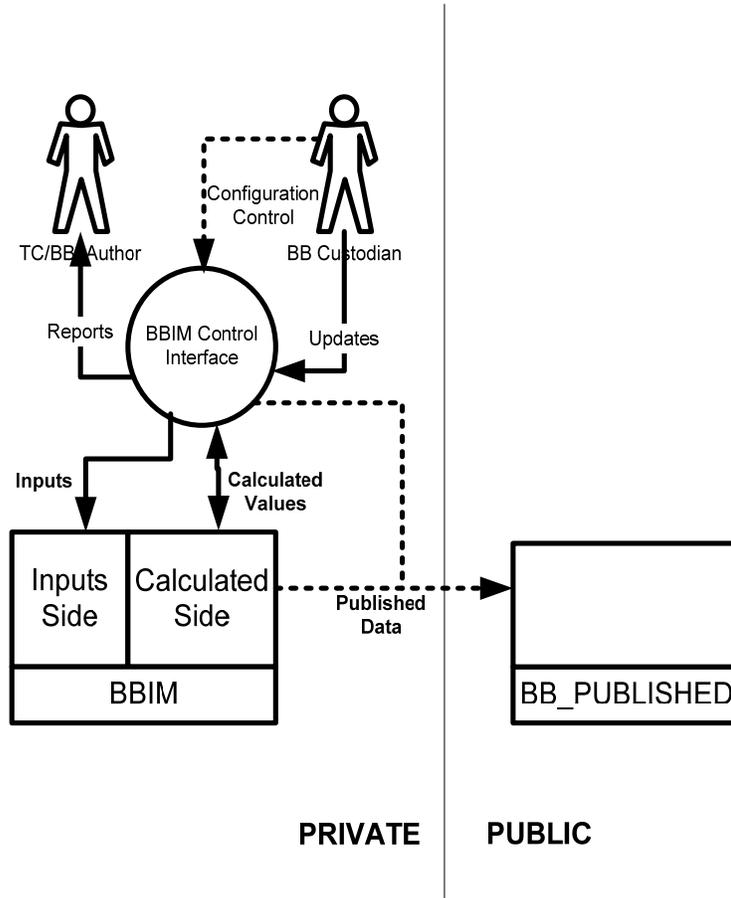


Figure 4.2. BBIM and BB_Published

4.2.3 TVD

4.2.3.1 Purpose and Content

The TVD data are tank headspace characterization data obtained after 1992. Sample collection data (e.g., date, time, sampling method, tank riser, etc.), sample analysis data (e.g., analytical laboratory, analysis date, chemical names, result values, units, etc.), averaged analytical results, and percents of the lower flammability limit (LFL) for selected analytes are included. Headspace flammability as total percent LFL is calculated by summing the percents of LFL for ammonia, carbon monoxide, hydrogen, methane, and total non-methane hydrocarbons.

Headspace characterization data are typically based on a single set of samples collected during a single sampling event, although some tanks have been sampled more than once. Most tank headspace samples have been collected in either SUMMA™ canisters or sorbent traps, using either the Vapor Sampling System or the In Situ Vapor Sampling methods. Samples were collected from the central regions of the tank headspaces, and are thought to be representative of the headspace at the time of sampling. Also provided are limited data from industrial hygiene field monitoring of the tank headspaces, including combustible gas meter readings.

Samples were analyzed for inorganic analytes (NH₃, NO₂, NO, H₂O), permanent gases (CO₂, CO, CH₄, H₂, N₂O), total non-methane hydrocarbons, and individual volatile and semi-volatile organic compounds by analytical laboratories using prescribed procedures. Concentrations are given at standard temperature and pressure (i.e., 273 K and 1 atm).

4.2.3.2 History

The PNNL Waste Tank Vapor Program began studying headspace vapors of the Hanford Site's single- and double-shell tanks in 1992 with sponsorship from the WHC Health and Safety organization. During 1993, the program grew to include offsite personnel from Oregon Graduate Institute (OGI), Oak Ridge National Laboratory, and Sandia National Laboratory in the analysis phase as well as PNNL's Analytical Laboratories staff. In 1993, the WHC manager of the program, Jerry Osborne, requested that PNNL develop extensive database capabilities to support the work being done. From November to December 1993, a needs analysis was conducted with Jerry Osborne and Jim Huckaby of WHC and the PNNL Waste Tank Vapor Project task leaders, resulting in a conceptual model of the desired system's capabilities, which would include meteorological, toxicological, and analytical data, and quality assurance information. The relational database model and data dictionary were presented to the WHC and PNNL managers and scientists involved who refined the model to reflect the most significant information that WHC wanted. The revised model was also reviewed by the computer scientists from TCD and TWINS because the long-term strategy was to have TVD share data with both of these systems and to have TWINS provide a convenient mechanism for the wide accessibility of the data. This design is documented in *Tank Vapor Database (TVD) System Design Description RPP-2441*¹⁴.

Time and budget constraints in FY 1994 prevented the simultaneous development of all of the model's features. Instead, the focus was on the development of a relational database that stored analytical results. The WHC manager explicitly specified that the vapor information be widely accessible over the Hanford Local Area Network, contain only cleared results (not interim or draft findings), all results be identifiable with their source document, and the final decision concerning the inclusion or exclusion of any data from the database be made by the WHC program manager.

In response to WHC's expressed desire to have a working system implemented by the end of FY 1994, a Rapid Application Development methodology was developed and implemented. The goal of this methodology was to create working prototypes of the system as soon as possible for customer review and comment, especially because in this case the arrival of the procured hardware and software took six months. In April 1994, a prototype database was constructed on an IBM personal computer to provide the Toxicology task with summarized and detailed information about sample job 6B for tank 241-C-103, which was presented at Vapor Conference 4 by WHC, PNNL, Oak Ridge National Laboratory, and OGI.

The decision to procure a SUN™ SPARCstation 1000 server and Sybase™ as the database because both were the products of choice by PNNL computer scientists and were being used by TWINS. A user interface on the database, DataPrism™, was chosen for its ability to be used without further customization with databases on the server and prototype databases. Instructions for running the user interface software were included in *Tank Vapor Database User's Guide*⁶ (which received a Recognition Award by the Society for Technical Communications). The first released TVD Data Dictionary and Data Model, Version 3.3, is documented in *Waste Tank Vapor Project: Tank Vapor Database Development*, PNL-10130⁵.

Because the procured hardware and software would not arrive in FY 1994, space was borrowed on the TWINS Sparc2, and their Sybase 4.9.2 was used. By the end of FY 1994, the system was implemented in this client/server environment and used DataPrism™ as the initial query tool. The core tables of the TVD were made available using Hanford Local Area Network in September 1994, meeting the formal milestone. The amount of data was limited to portions of tank 241-C-103 because of the small number of cleared reports. These tank data were revised and updated in FY 1996.

During FY 1995, support for the PNNL project shifted from the WHC Tank Waste Remediation System (TWRS) Safety Program to the WHC Characterization Program. The level of Quality Assurance was upgraded to PNNL Impact Level II; however, TVD funding to implement Quality Assurance Level II requirements was not available until FY 1996.

With the arrival of the procured hardware and software in FY 1995, TVD was moved from the Sparc2 to the Sparc1000, the operating system was upgraded from Sun O/S to Solaris 2.4, and the database was upgraded from Sybase 4.9.2 to Sybase 10.0.2.1. Also, TVD was integrated into TWINS to make the data accessible to a wide number of users. Omni-SQL™ Gateway was procured to provide connectivity between TVD and TCD that was under the Oracle™ DBMS.

In FY 1995, mechanisms were generated to obtain electronic laboratory data files where possible in Excel® or Lotus 1-2-3® spreadsheets. Only OGI data, WHC sampler's reports, and PNNL's inorganic data were in text form. To effectively load chemical information, the Hanford Technical Library sent chemical information in an ASCII file. The data loading into TVD was significantly improved with the development of diverse Visual Basic programs. To assist in verifying the vapor data, a Visual Basic program was designed to check the incoming laboratory Chemical Abstract Service numbers and chemical names against the Hanford Technical Library information. Any necessary corrections were checked with the appropriate laboratory staff. Also, to improve the efficiency of processing the chemical data, unique identifiers were created by a PNNL chemist for less well-defined chemicals and mixtures. A user interface to TVD was created in Microsoft Access® to readily enter the complete reference citation for the laboratory reports. The TVD was restructured to reflect the information that was provided by the laboratories and to make it more useful in the long term. By September 1995, data for 33 tanks were loaded from cleared reports. However, because the unique identifiers were still being assigned, there were a number of these less well-defined chemicals to be loaded in the future for the loaded tanks. There also were new concerns about the public availability of the vapor documents, which were resolved with the Clearance Office.

The major objectives of the Tank Vapor Database team in FY 1996 were to load and verify data about the vapor phase of the tank wastes, to load these data within 7 calendar days from the date of issue of the final report, to support the TWINS2 staff in providing accessibility on the web, to revise and complete

tank 241-C-103 data, to add the capability to include industrial hygiene field monitoring data and flammability results, to add data qualifiers, and to add holding time information. Other revisions were also made to the database structure to provide or clarify information. The revised TVD Data Dictionary and Data Model, Version 4.1, is documented in *Tank Vapor Characterization Project: Annual Status Report for FY 1996* (PNNL-11426)⁸.

In mid-FY 1996, a new data-loading requirement was implemented to meet TPA Milestone M 44. The analytical data has since been loaded consistently into the database within seven calendar days from the time the final report is issued. By end of FY 1996, TVD had a total of 64 tanks, and by mid-FY 1997, there were a total of 84 tanks or more specifically 119 events.

In mid-FY 1997, TVD became part of the TCD Project in efforts to reduce costs by consolidating hardware, software, and documentation. For this consolidation-planning phase, a separate Letter of Instruction was written for the administration and operation of TVD under TCD.

4.2.3.3 Data Maintenance Tools and Interfaces

The TVD data dictionary and data model, Version 4.1, is described in *Tank Vapor Characterization Project: Annual Status Report for FY 1996*⁸. The TVD and data loading components are depicted in Figure 4.3.

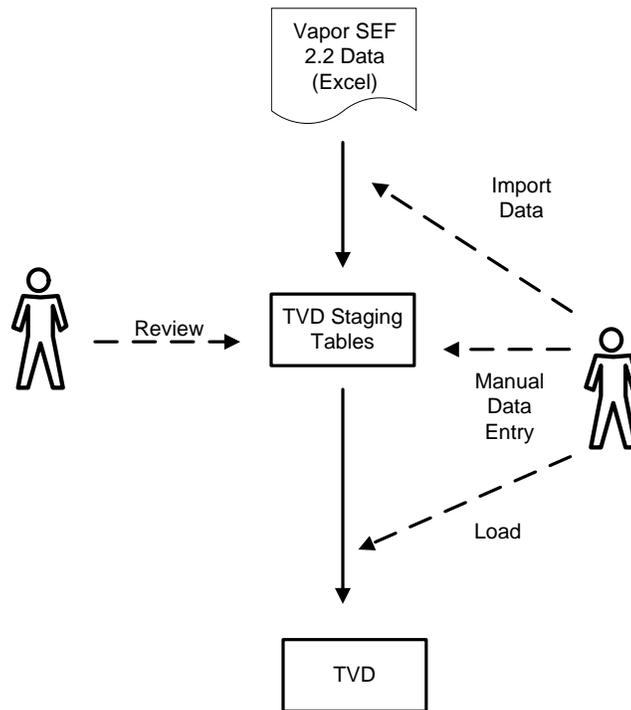


Figure 4.3. Tank Vapor Database

In addition to the database, the Vapor ASP.NET application has been developed to facilitate the loading process of TVD data. This application allows for the manual entry of sample data that are provided in hardcopy form; the import of analysis results that are provided in an Excel spreadsheet; the

review of TVD data that have been loaded input or imported into a staging area; and a data loading function that copies the staged data into the production TVD.

4.3 Reporting Tools

4.3.1 TWINS Web Interface

4.3.1.1 Purpose and Content

“TWINS” (<http://twins.pnl.gov>) is a web site providing access to several different heterogeneous databases, and doing so via an interface that makes the databases appear to be collocated and highly related. The power of TWINS has always been the ability to connect and map to legacy databases so as to provide a single point of access and, to the degree required, a normalization of databases so that information in related databases could be joined and analyzed as a single “virtual” database.

4.3.1.2 History

In December 1995, DOE Headquarters, reacting to budget cuts, dropped support from their budget for TWINS (up until this time, TWINS had been a project funded and managed by DOE Headquarters). Critical TWINS scope was picked up by the Hanford Site operating contractor, WHC, in the TWRS Characterization Technical Basis Program through the TCD project, and an effort was made to reduce the complexity and operational costs of TWINS. This was done primarily by replacing the Visual Basic-based TWINS user interface with a new web-based interface known at the time as “TWINS2.”

The *Tank Characterization Data Management Strategy*⁴ document and true commitment by WHC to support TCD laid a solid foundation for work to proceed in 1996. This document recognized the immediate technical and long-term strategic importance of an accurate and reliable TCD database. As a result, the following major steps were taken during FY 1996 to resolve the data management issues:

- confusing, overlapping databases were merged
- data problem tracking and change control procedures were implemented
- usage of TCD as the source of data for TCRs improved through the use of TCD “Standard Reports
- the TWINS2 HTML interface replaced the old TWINS Visual Basic-based desktop user interface.

The TWINS2 interface, in turn, was replaced by the TWINS web interface (or TWINS portal) in 1999. The TWINS portal uses ASP.NET technology and is designed around a “Library” metaphor – that is, it graphically looks like a library and functionally behaves like a library with several “rooms” representing the media types and underlying systems. The TWINS portal attempts to unify several different underlying systems into a single point of access, while also providing a path to incorporate new data sources, media types, and access to related links.

4.3.1.3 Architecture

Figure 4.4 depicts the TWINS architecture, including the databases presently connected through the TWINS Middleware and accessible through the TWINS user interface. As shown in the diagram,

TWINS is a three-level architecture: (1) the Database layer, (2) the TWINS Middleware layer, and (3) the User Interface.

Tank Waste Information Network System (TWINS) Architecture

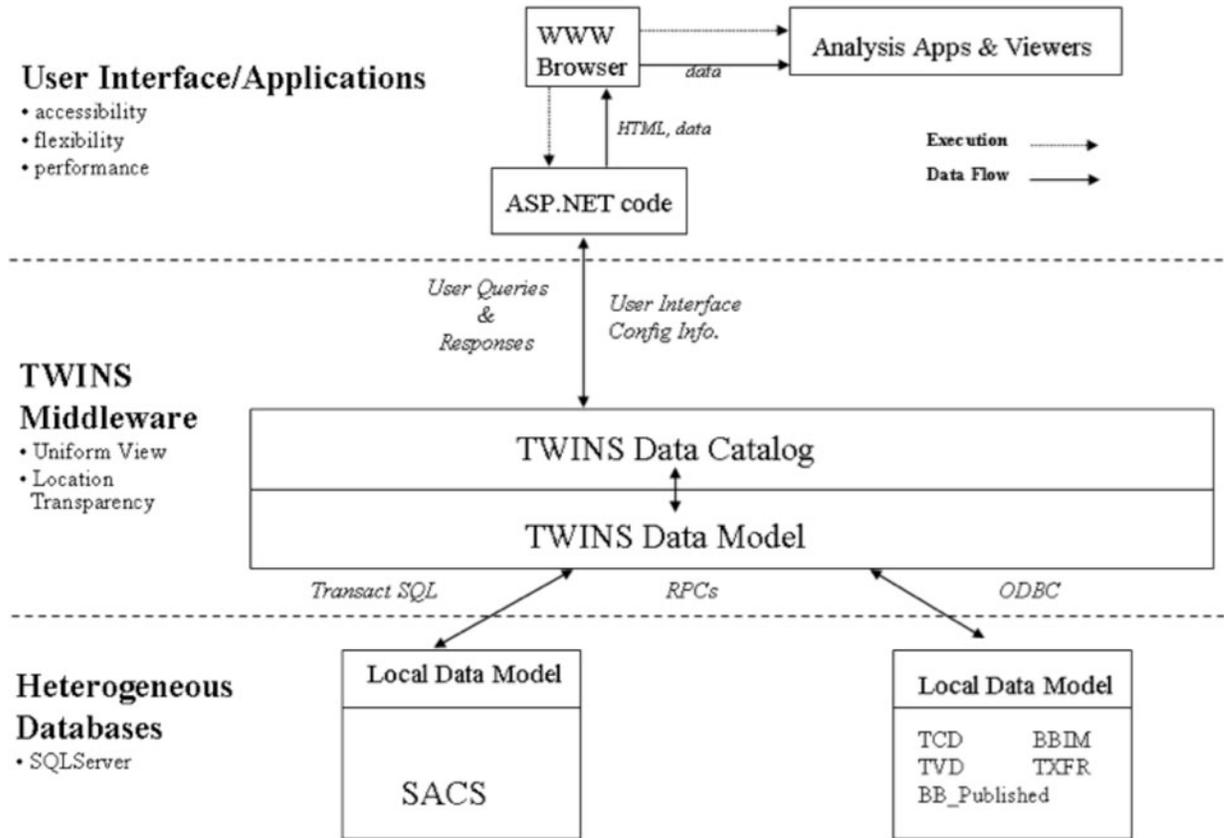


Figure 4.4. TWINS Web Interface Architecture

4.3.1.3.1 User Interface

The TWINS user interface is implemented using standard features of the WWW browser Internet Explorer. The browser communicates with web page ASP.NET code. This code performs queries on the TWINS server using the “Transact SQL” dialect of SQL. The code queries TWINS tables collectively known as the “Data Catalog” to determine which Subject Areas and which database tables to provide access to via the interface. As output, the code generates HTML that drives the WWW browser, resulting in the appearance of the TWINS user interface on the user’s PC.

Therefore, using the web browser method of interfacing with the TWINS Subject Area databases, applications can be configured to access and use the data as standard browser “viewers” or “applications.” For example, Microsoft® Excel is configured as the viewer for most users at the Hanford Site. Other viewers or analytical applications could easily be configured for this data. Additional code could be written to format data as required by other applications.

The TWINS user interface is described in detail in the requirements document, *Requirements for WWW-Based Query Interface to TCD*⁹.

In summary, TWINS is an HTML-based interface that is oriented toward users that have access to Microsoft® Excel, which is configured within the WWW browser of choice to be the “viewer” for tabular data returned from the database. Microsoft® Excel is universally available to the Hanford Site users. No other client software is required besides a WWW browser and Microsoft® Excel. The ASP.NET code invoked by the TWINS web pages performs database queries to configure a graphical user query interface.

When the user invokes a query, the data is returned in a “text/tab-separated-file” that is automatically imported into Excel®. A number of macros are also downloadable through the interface, which provides a handy way to view the data in relation to similar data in the database.

4.3.1.3.2 TWINS Data Catalog

The TWINS Data Catalog provides information about the structure and content of the data source. This enables the ASP.NET program to display the appropriate selection criteria to the end user, and, to format the results properly. The client program is written such that if new data sources are accessible and mapped in the ODBC communications layer, and the structure information of the source is stored in the data catalog, the data source will appear in the interface automatically; no changes to any program code are required.

4.3.1.3.3 Database Layer

The database layer currently contains multiple heterogeneous databases – ARCHIVE_DATA, HISTORICAL, BBIM, TXFR, AUTOTCR, TCD, SACS, BB_PUBLISHED and the Vapor database. These are all ODBC-accessible SQLServer databases.

4.3.2 TWINS Menu System

TWINS Menu System is an ASP.NET-based interface that provides a single point of access to TWINS capabilities that require passwords protection—for the most part, data maintenance interfaces. The URL is <http://twins.pnl.gov/apps/main.aspx>.

In general, there are many different roles that TWINS users play. Certain people are designated as custodians of specific databases or applications. This interface provides access to a set of applications individually configured for each authorized user. “Support Applications” such as the Best Basis Inventory Maintenance suite of tools, named “BBI,” are accessed via links listed across the top in Figure 4.5. Once an “Application” has been selected, the “Functions” available within the application are listed down the left side of the interface.

TWINS Support Applications

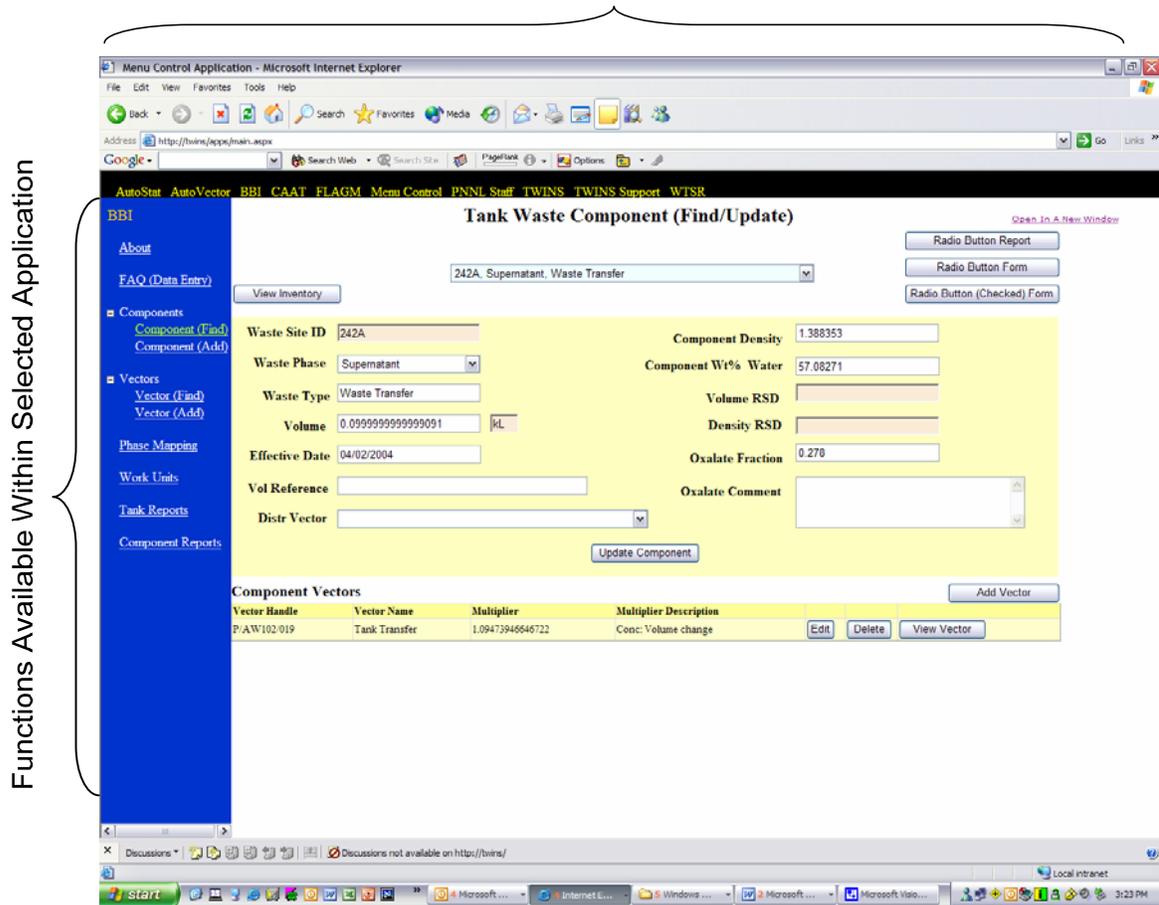


Figure 4.5. TWINS Menu System

4.4 Support Applications

4.4.1 AUTOSTAT Tool

4.4.1.1 Purpose and Content

4.4.1.2 History

AUTOSTAT was initiated as a task during FY03, as Change Request 237. Requirements and scope for this effort were defined in *Functional Requirements/Architecture Document for Twins Statistical Automation Upgrade*¹⁶. Due to other higher priorities, the programming and testing for this capability were not completed until June 2004.

4.4.1.3 Architecture

This ASP.NET application enables Tank Coordinators (TCs) and Best-Basis custodians to invoke the same statistical treatment of the analytical results data to calculate means and confidence intervals as heretofore has been done by statisticians (two-sided confidence interval using Searle equations). The tool takes as input the raw analytical results data stored in TCD, allows the user to select the precise set of analytical results to run the statistics on, and then stores the results into the AUTOSTAT means and confidence interval tables.

AUTOSTAT is an ASPX.NET application working in conjunction with SSIS to service queued-up user requests periodically in the background. See Figure 4.6.

Access to this tool was provided via the TWINS Support Applications Menu System.

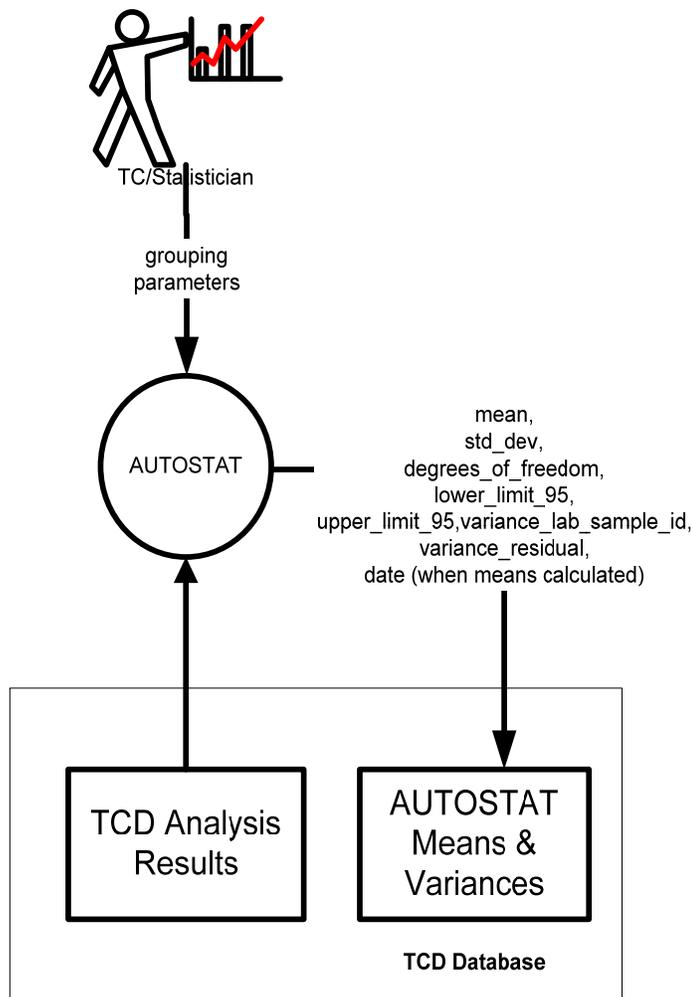


Figure 4.6. AUTOSTAT Context Diagram

4.4.2 AUTOVECTOR Tool

The AUTOVECTOR tool (Figure 4.7) is the link that bridges the gap between the BBI calculated inventories and the primary measured concentration values of TCD. It does this by linking through the AUTOSTAT “means and variances” tables; these tables are where the AUTOSTAT puts the calculated means and statistical parameters (See Figure 4.6).

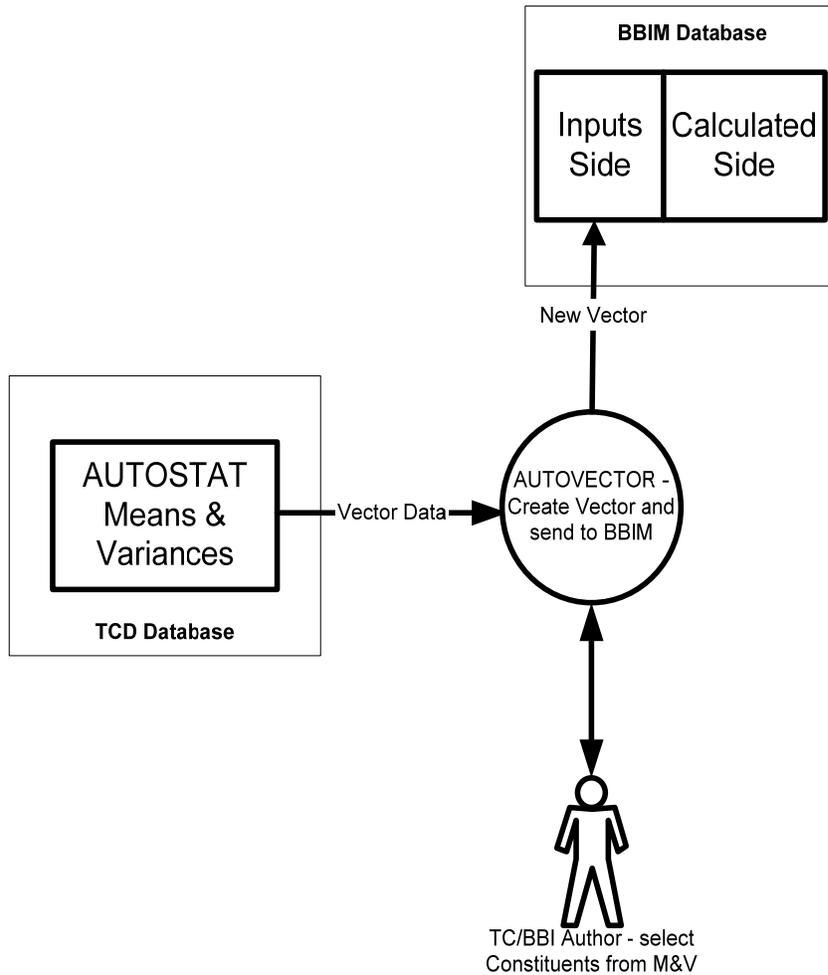


Figure 4.7. AUTOVECTOR Context Diagram

Once an “AutoVector” has been created by a TC, it is “submitted” by moving to the “Ready” status. Email is automatically sent to the staff designated as “vector-approver”, who is really a BBIM custodian. This person can then approve this candidate vector, which promotes it into the BBIM database (“Inputs Side” in Figure 4.7). In this process, a linkage is maintained back to the AUTOSTAT Means & Variances source, which in turn links back to the TCD analytical results and sample data.

The AUTOVECTOR tool, originally an Access 2000 tool, was enhanced and converted to ASP.NET and was put into production along with AUTOSTAT in June 2004.

Access to this tool was provided via the TWINS Support Applications Menu System.

4.4.3 TXFR Tool

4.4.3.1 Description

This tool is used to plan, manage, and track the transfer of wastes between tanks (single-shell tanks, double-shell tanks, and double-contained receiver tanks), and from outside the tank farm system to a tank, while maintaining the BBI pedigree. TXFR is an ASP.NET application that has the dual purpose of maintaining an official material balance of tank waste volumes across the tank farm and to update the BBIM inventories by uploading volume changes to the BBIM (see Figure 4.8). Although the tool has the capability of updating the BBIM inventories, this feature is no longer being used by the customer. Instead, they perform inventory calculations external to TWINS based upon volume changes and then create process knowledge vectors, entering these into the BBIM application for the purpose of modifying inventories.

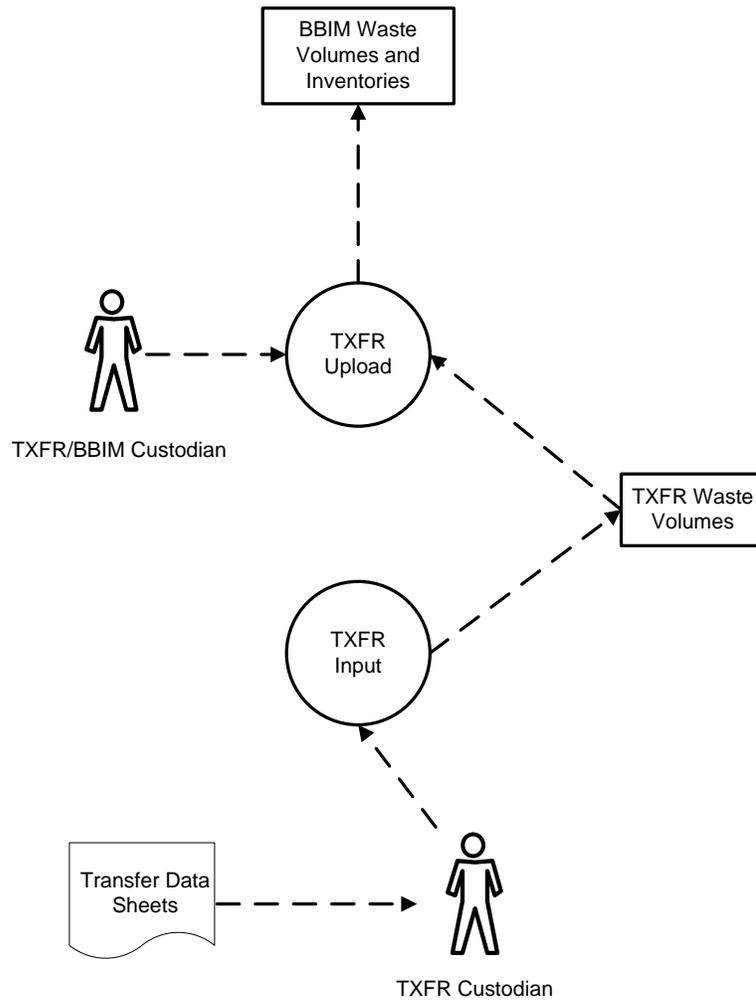


Figure 4.8. TXFR Tool Context Diagram

4.4.4 Resolve Tool

4.4.4.1 Description

Resolve is an ASP.NET application (See Figure 4.9) used by Tank Coordinators to review TCD analytical results data that have been flagged by the “p_check_rules” stored procedure that is run nightly to identify data that require a human assessment as to its worthiness to be used for tank characterization purposes. The Resolve user is provided the ability to either accept the computer check or override the computer check. This tool was first created (as an MS Access application) in the FY00 – FY01 time frame and later migrated to ASP and then ASP.NET technology. Data marked “Suspect” is not included in the AUTOTCR reports or in means creation using the AUTOSTAT tool. A more detailed overview of the Resolve tool can be found in *Resolve Application*¹⁵.

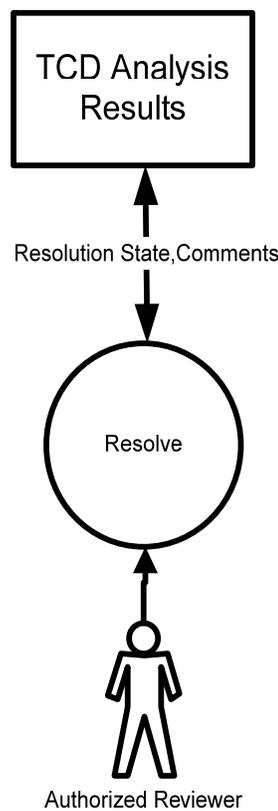


Figure 4.9. Resolve Data Review and Disposition Setting Tool Context Diagram

5.0 References

Author	Date	Location ¹	Title
1. F. M. Coony	6/8/92	Project File	<i>Tank Characterization Database Software Requirements Specification</i> (WHC-SD-EN-CSWD-036)
2. J. W. Brothers	4/30/92	Project File	<i>Tank Characterization Data Integration with HEIS Software Design Description</i> (WHC-SD-WM-CSDD-0066, PNL-8058)
3. S. F. Bobrowski	9/30/94	Project File	<i>Hanford Environmental Information System (HEIS) Volume 8, Tank Characterization Database Subject Area</i> (DOE/RL-93-24-8)
4. P. C. Ohl	6/30/95	Project File	<i>Tank Characterization Data Management Strategy</i> (WHC Correspondence No. 74A10-95-PC0-023)
5. P. R. Seesing	9/1994	Project File	<i>Tank Vapor Project: Tank Vapor Database Development</i> (PNL-10130)
6. K. L. Manke	10/1994	Project File	<i>Tank Vapor Database User's Guide</i> (PNL-10199)
7. M. W. Ligothke	11/1995	Project File	<i>Waste Tank Vapor Characterization Project: Annual Status Report for FY 1995</i> (PNL-10873)
8. K. L. Silvers	1/1997	Project File	<i>Waste Tank Vapor Characterization Project: Annual Status Report for FY 1996</i> (PNL-11426)
9. S. F. Bobrowski	7/11/96	tcd_doc	<i>Requirements for WWW-Based Query Interface to TCD, Version 2.0</i>

¹ Most "Project Files" are located on the TWINS SharePoint site at <https://pnlweb.pnl.gov/projects/TWINS/default.aspx>; "tcd_doc" is the name of a project share located at \\pnl\projects\tcd_doc.

Author	Date	Location	Title
10. L. L. Lang	1/01	tcd_doc	<i>Standard Electronic Format Specification for Tank Characterization Data Loader (Version 3.5)</i>
11. J. Smith	9/01	tcd_doc	<i>Standard Electronic Format/Excel (SEFX) Specification for Tank Characterization Data Loader: Version 1.0</i>
12. S. J. Harris	4/27/95	tcd_doc	<i>Requirements and Design for the Integrated Tank Characterization Database Data Loader (Visual Basic/Sybase)</i>
13. S. J. Harris	9/18/95	tcd_doc	<i>Procedures for Loading Analysis Result Data into the Tank Characterization Database (TCD) (Version 2.0)</i>
14. S.J. Harris	1/2005	Project File	<i>Tank Vapor Database (TVD) System Design Description</i>
15. T.S. Olund	2/2010	Project File	<i>RESOLVE Application</i>
16. M.R. Adams	4/2003	Project File	<i>Functional Requirements/Architecture Document for Twins Statistical Automation Upgrade</i>

Appendix A

Appendix A: Project Controls and Procedures

A number of project controls and procedures have been instituted at all levels of the project in order to make sure that the best quality services are being delivered to the customer and to the users. These controls and procedures span the range from administrative, or project management, to system operational procedures. The following list is not all-inclusive.

A1.0 PROJECT MANAGEMENT

To assure understanding of project scope, schedule, deliverables, budget, and related expectations for work to be performed, a Project Management Plan is prepared and updated each fiscal year or upon significant project changes. The name of the current document is *Tank Waste Information System (TWINS) fiscal year (FY) 2008 Project Management Plan, Version 7.0¹*.

A2.0 SOFTWARE DEVELOPMENT PROCEDURES

Software development procedures followed by the TWINS project during the software project lifecycle are based on those guidelines described in *Software Systems Engineering Process Guide, Version 3.2²*.

A3.0 SOFTWARE QUALITY ASSURANCE REQUIREMENTS

The *Tank Waste Information Network System (TWINS) Quality Assurance Plan, Version 7³*, specifies the applicable software Quality Assurance (QA) requirements and procedures for the TWINS project. The TWINS client (Washington River Protection Solutions [WRPS]) has specified that TWINS must conform to TFC-PLN-02, "Quality Assurance Program Description" or Pacific Northwest National Laboratory's (PNNL's) equivalent implementing documents for software QA. Conformance with the WRPS document is addressed in the TWINS QA Plan. Within the PNNL environment, TWINS is executed as an Impact Level II project with associated requirements for training, quality assurance, deficiency reporting, records management, and software control. In most cases, these requirements either meet or exceed the client requirements. In cases where they do not, the TWINS QA Plan specifies those additional requirements.

A4.0 TANK CHARACTERIZATION SOFTWARE CONFIGURATION MANAGEMENT PLAN

The *Software Configuration Management Plan Tank Waste Information Network System Project, Revision 6⁴* is the governing document for performance of Software Configuration Management (CM) activities for TCD. WRPS has specified that TWINS must conform to TFC-POL-19 Rev. A-1, "Configuration Management Policy" (or PNNL's equivalent implementing documents for software QA). Conformance with the WRPS document is addressed in the TWINS CM Plan. The CM activity verifies the integrity of software and related documentation through identification, establishment of baselines, control and status of changes to those baselines, and verification of the configuration through reviews and audits.

A5.0 TWINS SOFTWARE CHANGE MANAGEMENT

TWINS software shall be managed in accordance with *Software Configuration Management Plan Tank Waste Information Network System Project* (PNNL 2008)⁴. Change order configuration control shall be maintained in accordance with RPP-6831, Section 11.0, “Management of TWINS Design Changes” (Adams 2001)⁷.

A6.0 DATA LOADING OPERATION

WHC had prepared a “desk instruction” that documents the required procedure to be followed for loading laboratory data into TCD. This is called *Procedures for Loading Analysis Result Data and the Tank Characterization Data Base*⁵.

A7.0 COMPUTER SYSTEM SECURITY AND ADMINISTRATIVE (TWINS SERVER)

Procedures are maintained in the *TWINS Tape Backup Plan*¹³, which includes procedures for performing/recovering from backups, where and how to store and rotate tapes, powering down/up the systems, and shutting down the systems.

A plan covering physical security, Emergency Procedures, and Contingency Plans is contained in *Contingency Plan for TWINS Servers and Workstations*⁶.

A8.0 DATA CONFIGURATION MANAGEMENT

Data Deficiency Forms configuration control shall be maintained in accordance with RPP-6831, latest revision, Section 10.0, *Management of TWINS Data Deficiencies* (Adams 2001)⁸.

A9.0 USER RESPONSE FACILITY

An online capability has been included in the TWINS user interface for users to submit their feedback regarding the data or the user interface. Messages submitted using this mechanism are sent to the Characterization Data Quality Officer. From that point, data issues can be submitted to the Data Deficiency Forms System. Requests or suggestions for software changes can be considered by WRPS, approved or disapproved by WRPS, and then directed to PNNL if appropriate.

A10.0 USER AUTHORIZATION

Because the TWINS interface is available to all Hanford Local Area Network (HLAN) users without a password, there is no special requirement for User Authorization for TWINS user interface access from HLAN. Users who would like access to TWINS who are not on HLAN are granted access on a case-by-case basis. Request for access must be approved by the WRPS Customer Lead.

Applications managed by the TWINS menu system are password protected. Access to these applications is requested by the WRPS Customer Lead.

A11.0 SOFTWARE TESTING

The *Software Verification and Validation Procedure, Revision 5.0*⁹ specifies the applicable action steps for software testing for the TWINS project testing process. The primary objective of the testing process is to provide assurance that the software functions as intended and meets the requirements specified by the client. Validation and verification are the primary means for TWINS software product acceptance. WRPS requirements are mapped to PNNL equivalent requirements in the TWINS QA Plan³.

A12.0 USE OF CHANGE REQUEST TRACKING TOOL

Figure A.1 depicts the flow of changes from the customer organization to the software development organization, and use of the Change Request Tracking Tool (CRTT) within PNNL. This is governed by *TWINS Developer Procedure For Use Of CRTT* (Bobrowski 2002)¹⁰.

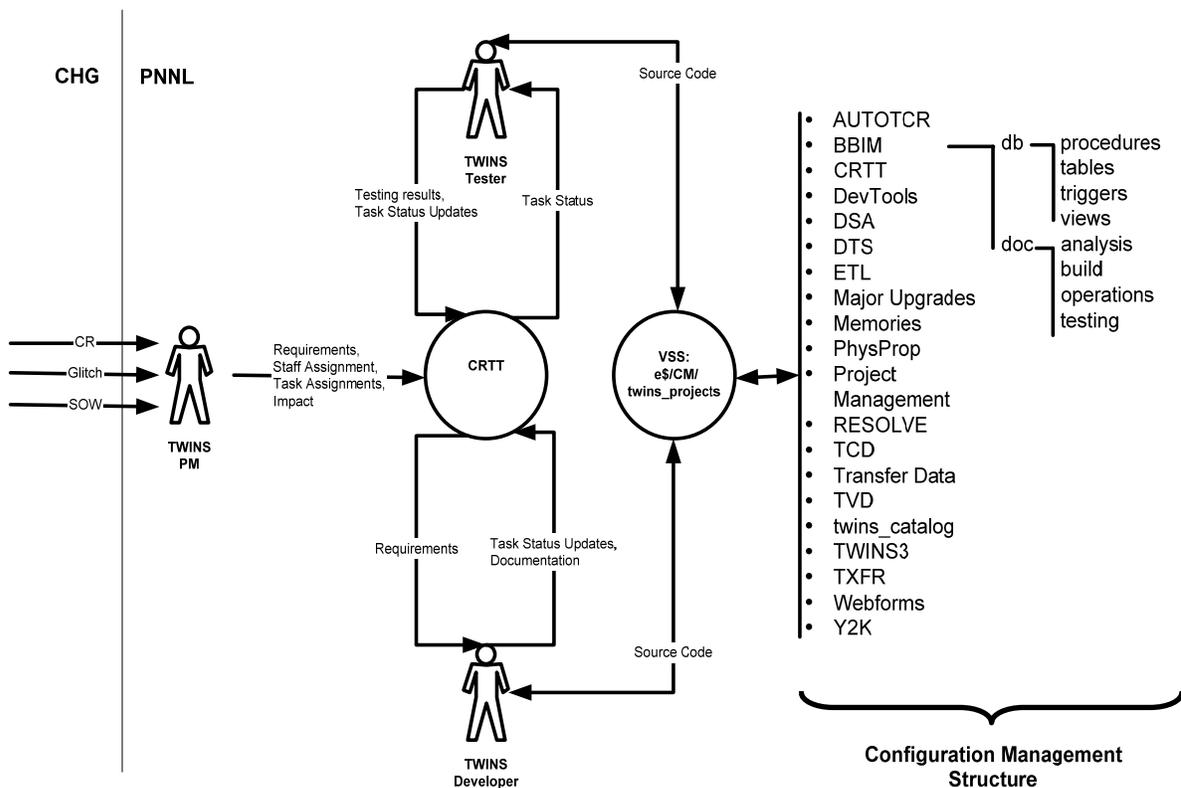


Figure A.1. TWINS Systems Engineering Process Flow

A13.0 REFERENCES

Author	Date	Location¹	Title
1. T.S. Olund	January 2008	TRIM	<i>Tank Waste Information System (TWINS) FY 2008 Project Management Plan, Version 7</i>
2. PNL/CISD	March 2008	ssep.pnl.gov	<i>Software Systems Engineering Process Guide, Version 3.2</i>
3. K.R. Middleton	April 2005	Project File	<i>Tank Waste Information Network System (TWINS) Quality Assurance Plan, Version 7</i>
4. T.S. Olund	March 2008	Project File	<i>Software Configuration Management Plan Tank Waste Information Network System Project, Revision 6</i>
5. J. Smart	9/15/1995	WHC DI	<i>Procedures for Loading Analysis Result Data and the Tank Characterization Database (DI-DM-003.00)</i>
6. B. Homer	3/6/2008	TRIM	<i>Contingency Plan for TWINS Servers and Workstations</i>
7. M. R. Adams	2001		RPP-6831, Section 11.0 “Management of TWINS Design Changes”
8. M. R. Adams	2001		RPP-6831, Section 10.0 “Management of TWINS Data Deficiencies”
9. K. R. Middleton	3/2008	Project File	<i>Software Verification And Validation Procedure, Revision 5.0</i>
10. S. F. Bobrowski	12/26/2002	Project File	<i>TWINS Developer Procedure For Use of CRTT, Revision 1.0</i>
11. B. Homer	March 2008	TRIM	<i>TWINS Tape Backup Plan</i>

¹Most “Project Files” are located on the TWINS SharePoint site at <https://pnlweb.pnl.gov/projects/TWINS/default.aspx>; “tcd_doc” is the name of a PNNL project share located at \\pnl\projects\tcd_doc.



Pacific Northwest
NATIONAL LABORATORY

902 Battelle Boulevard
P.O. Box 999
Richland, WA 99352
1-888-375-PNNL (7665)
www.pnl.gov



U.S. DEPARTMENT OF
ENERGY

TWINS Requirements Documents

AutoStat Requirements

[Type text]

AutoStat – Change Request Tracking Tool Requirements

Change Request 281: AutoStat Averaging Selections

See: <http://twins/crtt/CRTT.asp?ID=377&Type=CR>

Requirement #	Description
281	Make the following changes to the “Averaging” section of the “AutoStat: Edit Record” page: <ul style="list-style-type: none"> • Change section name from “Averaging” to “Averaging (select one)” • Remove “Recommended Grouping for Averaging: Method ...” text. • Change drop down box to radio buttons with no selections made. (We want this to be a required field where the user must actively make a selection.) • Remove “Use Recommended” from selection list. • Edit the three remaining selections as follows: <ul style="list-style-type: none"> o Change “Method 1 – No Averaging” to “Method 1 - Grabs and Single Cores/Augers with no subsegments”. o Change “Method 2 – SubSample Level” to “Method 2 – Single Cores/Augers with subsegments”. o Change “Method 3 – Sample Level” to “Method 3 – Multiple Cores/Augers”.

Change Request 237: Program AutoStat Function

<http://twins/crtt/CRTT.asp?ID=282&Type=CR>

Requirement #	Description
237	Program the AUTOSTAT function as described in the attached functional requirements/architecture document. NOTE: Those requirements are in the file STATISTICAL AUTOMATION ARCHITECTURE DOCUMENT1. doc and are included in this document. CR 237 is included here to link back to test documentation.

Change Request 294: AutoStat – Manual Entry

See: <http://twins/crtt/CRTT.asp?ID=394&Type=CR>

Requirement #	Description
294	Provide a tool to allow the Statistician to manually load means into the database that have been calculated outside of the AutoStat tool. JUSTIFICATION: Currently AutoStat provides a tool for Tank Coordinators to run means on new sample based data without needing a Statistician’s help. Occasionally there are situations, including the need for reconstituting data and weighting data, which require the Statistician to run the means manually. An area in AutoStat or tool is needed to allow the Statistician to load the means into the means database.

Change Request 293: AutoStat – PCBs

See: <http://twins/crtt/CRTT.asp?ID=393&Type=CR>

Requirement #	Description
293	Currently AutoStat calculates means and standard deviations for each individual Aroclor reported. For reporting purposes Aroclors need to be converted to

polychlorinated biphenyls (PCBs) using the rules below. Means for the individual Aroclors should be removed from the means report. o Results are generally reported as dry weight. AutoStat should return two sets of means – one for the dry weight (“Aroclors (Total PCB) – dry weight”) and one for the wet weight (“Aroclors (Total PCB)”). To convert to a wet weight basis use the following, where wet and dry are concentration units: $wet = dry \times (100 - \%H_2O) / 100$. %H₂O is the REML mean of the weight percent water for that particular sample. If there is no corresponding wet percent water leave all calculated values blank and append “Weight percent water is not available for this sample” to the comment field. o If all Aroclor concentrations are “less than detects,” the maximum Aroclor 1254 concentration is selected as the total PCB concentration. o If only one Aroclor is detected, the REML (restricted maximum likelihood method) mean of all concentrations for the specific Aroclor including “less than detects” is calculated. If 50% or more of the concentrations used in the mean calculation are detected, then the mean total PCB concentration is considered a detected value. Otherwise, it is a “less than detect.” o If multiple Aroclors are detected, the REML mean of each detected Aroclor including “less than detects” is calculated. If 50% or more of the Aroclor values used in the mean calculations are detected, then the mean is considered a detected value. The mean total PCB concentration is estimated by summing the means of the detected Aroclors. JUSTIFICATION: BBI requires the means and standard deviations to be calculated for the Aroclors (Total PCBs).

Change Request 377: Create TWINS Means and Variances Report

See: <http://twins.pnl.gov/crttview/CRTT.asp?ID=498&Type=CR>

The development web page is located at:

http://twinsdevui.pnl.gov/twinsdata/forms/BuildQuery.aspx?SourceName=tcd.dbo.v_means_and_variances

Requirement #	Description
CR377.01	The report should contain get the data from the S-Plus data (tcd.dbo.means_and_variances_detail table).
CR377.02	The report should contain the following columns: <ol style="list-style-type: none"> 1. Tank Name 2. Reference Date 3. Agg Level 4. Constituent Name 5. Chemical Formula 6. Analysis Method Group 7. Phase 8. Mean 9. Units 10. Standard Deviation 11. Degrees of Freedom 12. 95% LL 13. 95% UL

	14. Number Above Detect 15. Number Below Detect 16. Comment 17. Variance – Lab Sample 18. Variance – Residual 19. RSD 20. Variance – Sampling Event 21. Variance – Segment 22. Variance – Riser 23. Sampling Begin Date 24. Sampling End Date 25. Reference
CR377.03	When multiple sampling events were used to generate the tcd.dbo.means_and_variances_detail data, the latest Sampling Begin Date from the group shall be reported (this matches the Means and Confidence Interval reports generated by the AutoTCR application).
CR377.04	When multiple sampling events were used to generate the tcd.dbo.means_and_variances_detail data, the latest Sampling End Date from the group shall be reported.
CR377.05	The user should be able to query by Tank and/or Constituent.
CR377.06	A report that gets data from the AutoStat data is not needed as this already exists: http://twins.pnl.gov/twinsdata/forms/BuildQuery.aspx?SourceName=tcd.dbo.v_autostatresults .
CR377.07	The report should not be available from the Data Menu in TWINS.
CR377.08	The report should be available from the AutoStat .NET application via the “Review Data” link in the left navigation menu.
CR377.09	The data should be verified by the customer for correctness (Dave Place).

Change Request 613: Replace the AutoTCR Means and Confidence Interval report

See: <http://twins.pnl.gov/crttview/CRTT.asp?ID=613&Type=BUG>

The form references are located by following these instructions:

1. Log onto the TWINS/Applications development site: <http://twinsdevui/apps>
2. Click on the **AutoStat** link in the top menu
3. Click on the **Means And Confidence** link in the left menu

Requirement #	Description
BUG613.01	The report should contain the following elements found in the AutoTCR Means and Confidence Interval Report (http://twins.pnl.gov/reports/reports.asp) <ol style="list-style-type: none"> 1. Name of the tank 2. Whether the sample comes from the Solid or Liquid phase 3. Name of the sample 4. Sampling date range 5. Analyte 6. Method

	<ul style="list-style-type: none"> 7. Mean 8. Degrees of freedom 9. Lower limit 10. Upper limit 11. Units
BUG613.02	The report should be output as HTML to the form, or to MS WORD
BUG613.03	In the production database (http://twins.pnl.gov/apps) only data from August 1 st , 2009 should be reported.
BUG613.04	The form should allow the user to generate a Means and Confidence Interval report for a selected tank.
BUG613.05	In the production database (http://twins.pnl.gov/apps) only tanks that have AutoStat Results added or modified since August 1 st , 2009 should be reported. Note, according to the customer, these include the tanks AN-102, AN-106 and AW-106).
BUG613.06	In the production database (http://twins.pnl.gov/apps) only data that has its <i>use in AutoTCR</i> checkbox checked should be included. These include tanks 240-S-302, 241-A-101, 241-A-102, 241-AN-102, 241-AN-104, 241-AN-106, 241-AP-101, 241-AP-102, 241-AP-103, 241-AP-104, 241-AP-105, 241-AP-107, 241-AP-108, 241-AW-102, 241-AW-103, 241-AW-105, 241-AW-106, 241-AX-102, 241-AX-104, 241-AY-101, 241-AY-102, 241-AZ-102, 241-BY-104, 241-BY-106, 241-BY-107, 241-BY-108, 241-BY-110, 241-C-102, 241-C-103, 241-C-104, 241-C-106, 241-C-108, 241-C-201, 241-C-202, 241-C-203, 241-C-204, 241-S-112, 241-SY-101, 241-SY-102, 241-SY-103, 241-TX-104, 241-TX-118.
BUG613.07	The report should be available to AutoStat users (link in the left navigation bar for the AutoStat .NET application)
BUG613.08	Only data that has had its "User in AutoTCR checkbox" checked should be included in the report.
BUG613.09	The data values from the new Means and Confidence Interval Report (http://twins/apps/tcdloader2/Forms/MeansAndConfidence.aspx) should match the values from the AutoTCR Means and Confidence Interval report (http://twins/reports/reports.asp) which it is replacing.
BUG613.10	Analytes should have an asterisk appended to it if the number of non-detected values is greater than zero.
BUG613.11	The Mean should have a less than (<) sign for data that has its non-detected values greater than its detected values.
BUG613.12	The Lower Limit can report a value of "n/a" or be left blank when the data has a null value in the database.
BUG613.13	The Upper Limit can report a value of "n/a" or be left blank when the data has a null value in the database.

AutoVector Requirements

AutoVector – Change Request Tracking Tool Requirements

Change Request 289: AutoVector – Ammonia vs Ammonium

See: <http://twins/crtt/CRTT.asp?ID=385&Type=CR>

Requirement #	Description
289	The AutoVector Tool should be updated to automatically convert ammonium (NH ₄) to ammonia (NH ₃), so it can be used in the BBI. The following formula should be used in the conversion: $NH_3 = NH_4 * 17 / 18$ JUSTIFICATION: This update will eliminate the manual conversion of NH ₄ to NH ₃ .

Change Request 129: AutoVector items

See: <http://twins/crtt/CRTT.asp?ID=121&Type=BUG>

Requirement #	Description
129	After much discussion on our end, the plan is to enter PCBs by hand in the existing format "aroclors (Total PCB)". It will NOT go through the AutoVector tool due to the amount of manipulation and evaluation that will be required. I believe it is not worth the effort at this time to program any of the preliminary calculations or decision steps associated with PCBs.

Change Request 122: Automate Waste Vector Creation from TCD Means and Variances Tool

See: <http://twins/crtt/CRTT.asp?ID=121&Type=CR>

Requirement #	Description
122	The overall purpose of this change is to improve the electronic linkage of TCD analytical results with the BBI tool. The change is broken down into two distinct phases of work. (details in CHANGE 122.doc)

Tank Waste Information Network System Change Control Form

CHANGE CONTROL NUMBER: 122

DATE OF CHANGE REQUEST: 01/10/00

TYPE OF CHANGE REQUEST: TCD/BBI

TITLE OF CHANGE REQUEST: Automate Waste Vector Creation from TCD Means and Variances Tool

DESCRIPTION OF CHANGE:

The overall purpose of this change is to improve the electronic linkage of TCD analytical results with the BBI tool. The change is broken down into two distinct phases of work:

1. Provide the ability to automatically create BBI vectors from the Means and Variances tool.

Currently the statistical tool provides the ability to upload statistical data from a spreadsheet, review the data and load the data into TCD. This task will enhance this ability by adding the following capabilities:

- a. Capture required BBI vector metadata including: vector name, analysis date, decay date reported, density, density comment. To the extent possible generate vector metadata automatically including fields: reference document, reference data table, name, reference table number, and reference table column number.
 - b. Create ability to select only certain analytes for transfer from statistical tool to new vector. This may be done by adding a checkbox field to the statistical tool.
 - c. Create ability to enter a vector multiplier.
 - d. Create a new vector in the BBI containing the metadata captured in a. This new vector must be distinguishable from all pre-existing vectors in the BBI.
 - e. Create vector_constituents (for Best-Basis standard constituents) for each selected record in the statistical tool containing the following:
 - Analyte (with mapping of TCD long name to BBI constituent name).
 - Qualifier (automatically generate “<” according to the rule: if number less than > number of non-less-thans, then set the qualifier to “<”).
 - Mean value
 - Units.
 - f. Send an e-mail message to Thuy Tran (or designee) advising them that this new vector has been created and is ready for processing.
 - g. Create ability within BBI tool to easily locate these new vectors and move them to the production vector area.
2. Provide linkage between BBI vector_constituent value and the TCD analytical results from which the concentration was derived. Capture TCD analytical result ids and other metadata that are used to generate the statistical mean, or that are needed to trace the generation of the statistical mean. These data may

be saved in BBI vector_constituent record comment fields or new fields created for that purpose.

INITIATOR: Willmarth

DATE COMPLETION REQUIRED: after CR # 120. March 2000 target.

SYSTEM DOCUMENTATION REQUIRED: yes, change data dictionary and user guide as needed.

COMPLETION NOTIFICATION SENT:

DATE OF COMPLETION:

APPROVALS:

M.R. ADAMS

J.G. FIELD

BBIM Requirements

Best Basis Inventory Model Requirements

Last Updated On 7/8/2009

1.0 Introduction

This document describes the requirements for the changes occurring to the Best Basis Inventory Model (BBIM). The requirements are pulled from several requirements documents developed over the past 17 years. Requirements for developing BBI numbers can be found in *Best-Basis Inventory Process Requirements*, RPP-7625, developed by CH2M HILL Hanford Group, Inc. Test plans and results and other BBIM documentation are available (hard copy only) in the *Best Basis Inventory Tool (BBIM) Development Documentation* binder from Tom Olund (tom.olund@pnl.gov).

2.0 Requirements, Deliverables and Assumptions

Number	Requirement or Deliverable
1.	Project Objective Requirements
1.1	The overall objective of this task is to design, program, test, and place into operation an electronic tool to enable scientists and engineers to prepare, document, calculate, evaluate, edit, and report best-basis (standard) inventory values on a tank-by-tank and global basis. Sources of input values used to calculate inventories shall be captured by the database and means provided to record the relative and/or absolute confidence in those values.
1.2	Capture author assumptions and input data.
1.3	Calculate the inventory using a standard algorithm.
1.4	The standard algorithm must be controllable and editable by authorized personnel.
1.5	Ability to create/store standard tables and reports containing inventory source data and assumptions.
1.6	Ability to create/store standard inventory values and associated uncertainty values.
1.7	Provide database programming and reporting tools that are capable of detecting and assisting in resolving data discrepancies.
1.8	Linking the best-basis inventory data contained in the TCD to the appropriate supporting documentation for proper quality assurance and control. This shall include any and all changes to the best-basis inventories.
1.9	Ability to add new chemical and radioactive analytes, physical properties, soluble and insoluble splits for water and caustic leach, and information regarding waste phases.
1.10	Since this information is not currently contained in the TCD, tools and procedures must be developed for easily adding and retrieving new waste information.
1.11	In order to meet the requirements for a stable, configuration controlled inventory database that is readily available to data end-users, the ability to "publish" new versions of the best-basis inventories shall be provided. At the same time, a

	working version of the inventory database shall be available to adjust, analyze, and document newly-discovered information. Once the new information is analyzed and adjustments to the best-basis inventories are authorized by a formal Change Control Board, then the user-accessible tank inventory data in the TCD shall be updated on an annual basis, or more frequently as the need dictates.																		
2	Documentation Requirements																		
2.1	Requirements Document																		
2.2	Data Dictionary																		
2.3	Data Flow Diagram																		
2.4	Test Procedures and Results																		
2.5	Design and architecture																		
2.6	Configuration control																		
2.7	The documents may be in standard PNNL format and shall comply with PNNL procedures for software documentation. Copies of the documents shall be placed in the TCD document library at PNNL.																		
3	Functional Requirements																		
3.1	Tank Waste shall be found in one of these six waste phases: Retained Gas - Salt Cake; Retained Gas - Sludge; Supernatant; Salt Cake Liquid; Salt Cake Solid; Sludge.																		
3.2	Analyte Dependents: Inventory for some analytes can be calculated if their "parent" analyte has an inventory value. These analytes are: <table style="margin-left: 40px; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">Parent</th> <th style="text-align: left;">Dependent</th> <th style="text-align: left;">Factor</th> </tr> </thead> <tbody> <tr> <td>90Sr</td> <td>90Y</td> <td>1.0</td> </tr> <tr> <td>137Cs</td> <td>137mBa</td> <td>0.946</td> </tr> <tr> <td>90Sr</td> <td>Sr</td> <td>2.4436e-005</td> </tr> <tr> <td>232Th</td> <td>228Ra</td> <td>1.0</td> </tr> <tr> <td>NO2</td> <td>Free OH</td> <td>1.0</td> </tr> </tbody> </table> <p>The Dependents are calculated using this formula:</p> $\text{Dependent Inv} = \text{Parent Inv} * \text{Factor}$	Parent	Dependent	Factor	90Sr	90Y	1.0	137Cs	137mBa	0.946	90Sr	Sr	2.4436e-005	232Th	228Ra	1.0	NO2	Free OH	1.0
Parent	Dependent	Factor																	
90Sr	90Y	1.0																	
137Cs	137mBa	0.946																	
90Sr	Sr	2.4436e-005																	
232Th	228Ra	1.0																	
NO2	Free OH	1.0																	
3.3	Hanford Defined Waste (HDW) data is used in conjunction with sample data to calculate radionuclides that have no sample data. However, any vector can be substituted and treated as an HDW vector for the purposes of calculating missing radionuclides. This vector substitution or selection is to be performed at the waste component level (e.g. different components within a tank can use different "HDW" vectors).																		
3.3.1	UTOTAL is comprised of the following radionuclides: 232U, 233U, 234U, 235U, 236U, and 238U. If sample based data for any of these radionuclides does not exist, then they can be calculated by using the model based concentrations found in the HDW data. The calculation is performed using the following formula:																		

	${}^{nnn}\text{U} = {}^{nnn}\text{U}_{\text{HDW}} * (\text{UTOTAL}_{\text{SAMPLED}} / \text{UTOTAL}_{\text{HDW}})$ <p>where</p> ${}^{nnn}\text{U} = \text{non-sampled U radionuclide}$ ${}^{nnn}\text{U}_{\text{HDW}} = \text{HDW U radionuclide}$ $\text{UTOTAL}_{\text{SAMPLED}} = \text{UTOTAL sample based value}$
3.3.2	<p>${}^{154}\text{Eu}$ can be used to calculate inventories for radionuclides: ${}^{152}\text{Eu}$ and ${}^{155}\text{Eu}$. If sample based data for any of these radionuclides does not exist, then they can be calculated by using the model based concentrations found in the HDW data. The calculation is performed using the following formula:</p> ${}^{nnn}\text{Eu} = {}^{nnn}\text{Eu}_{\text{HDW}} * ({}^{154}\text{Eu} / {}^{154}\text{Eu}_{\text{HDW}})$ <p>where</p> ${}^{nnn}\text{Eu} = \text{non-sampled } {}^{152}\text{Eu} \text{ or } {}^{155}\text{Eu}$ ${}^{154}\text{Eu} = \text{is sample based}$ ${}^{nnn}\text{Eu}_{\text{HDW}} = \text{HDW Eu radionuclide}$ ${}^{154}\text{Eu}_{\text{HDW}} = \text{HDW } {}^{154}\text{Eu}$
3.3.3	<p>${}^{243/244}\text{Cm}$ is comprised of the following radionuclides: ${}^{243}\text{Cm}$ and ${}^{244}\text{Cm}$. If sample based data for any of these radionuclides does not exist, then they can be calculated by using the model based concentrations found in the HDW data. The calculation is performed using the following formula:</p> ${}^{nnn}\text{Cm} = {}^{nnn}\text{Cm}_{\text{HDW}} * ({}^{243/244}\text{Cm}_{\text{SAMPLED}} / {}^{243/244}\text{Cm}_{\text{HDW}})$ <p>where</p> ${}^{nnn}\text{Cm} = \text{non-sampled Cm radionuclide}$ ${}^{nnn}\text{Cm}_{\text{HDW}} = \text{HDW Cm radionuclide}$ ${}^{243/244}\text{Cm}_{\text{SAMPLED}} = {}^{243/244}\text{Cm sample based value}$ ${}^{243/244}\text{Cm}_{\text{HDW}} = {}^{243}\text{Cm HDW based concentration} + {}^{244}\text{Cm HDW based concentration}$
3.3.4	<p>${}^{239/240}\text{Pu}$ is comprised of the following radionuclides: ${}^{239}\text{Pu}$ and ${}^{240}\text{Pu}$. If sample based data for any of these radionuclides does not exist, then they can be calculated by using the model based concentrations found in the HDW data. The calculation is performed using the following formula:</p> ${}^{nnn}\text{Pu} = {}^{nnn}\text{Pu}_{\text{HDW}} * ({}^{239/240}\text{Pu}_{\text{SAMPLED}} / {}^{239/240}\text{Pu}_{\text{HDW}})$ <p>where</p> ${}^{nnn}\text{Pu} = \text{non-sampled Pu radionuclide}$ ${}^{nnn}\text{Pu}_{\text{HDW}} = \text{HDW Pu radionuclide}$ ${}^{239/240}\text{Pu}_{\text{SAMPLED}} = {}^{239/240}\text{Pu sample based value}$ ${}^{239/240}\text{Pu}_{\text{HDW}} = {}^{239}\text{Pu HDW based concentration} + {}^{240}\text{Pu HDW based concentration}$

3.3.5	<p>239Pu can be used to calculate inventories for radionuclides: 238Pu, 240Pu and 242Pu. If sample based data for any of these radionuclides does not exist, then they can be calculated by using the model based concentrations found in the HDW data. The calculation is performed using the following formula:</p> ${}^{nnn}\text{Pu} = {}^{nnn}\text{Pu}_{\text{HDW}} * ({}^{239}\text{Pu} / {}^{239}\text{Pu}_{\text{HDW}})$ <p>where</p> <ul style="list-style-type: none"> ${}^{nnn}\text{Pu}$ = non-sampled ${}^{238}\text{Pu}$ or ${}^{240}\text{Pu}$ or ${}^{242}\text{Pu}$ ${}^{239}\text{Pu}$ = is sample based, or has been calculated from ${}^{239/240}\text{Pu}$ ${}^{nnn}\text{Pu}_{\text{HDW}}$ = HDW Pu radionuclide ${}^{239}\text{Pu}_{\text{HDW}}$ = HDW 239Pu <p>The “<” qualifier should be assigned to the calculated Pu isotope, only if the parent Pu isotope had the qualifier. Note that ${}^{238}\text{Pu}$ and ${}^{242}\text{Pu}$ are never parents.</p>
3.3.6	<p>241Am can be used to calculate inventories for radionuclides: 243Am, 242Cm, 243Cm and 244Cm. If sample based data for any of these radionuclides does not exist, then they can be calculated by using the model based concentrations found in the HDW data. The calculation is performed using the following formula:</p> ${}^{nnn}\text{Xm} = {}^{nnn}\text{Xm}_{\text{HDW}} * ({}^{241}\text{Am}_{\text{SAMPLED}} / {}^{241}\text{Am}_{\text{HDW}})$ <p>where</p> <ul style="list-style-type: none"> ${}^{nnn}\text{Xm}$ = non-sampled Am or Cm radionuclide ${}^{nnn}\text{Xm}_{\text{HDW}}$ = HDW Am or Cm radionuclide ${}^{241}\text{Am}_{\text{SAMPLED}}$ = 241Am sample based value ${}^{241}\text{Am}_{\text{HDW}}$ = 241Am HDW based value
3.4	<p>TotalAlpha can be used to calculate the inventories for radionuclides: 239Pu, 242Pu, 232U, 238Pu, 240Pu, 241Am, 243Am, 243Cm, 244Cm, 242Cm, 235U, 236U, 238U, 233U and 234U. If sample based data for any of these radionuclides does not exist, then they can be calculated by using the model based concentrations found in the HDW data. The HDW data must first be decayed (intermediate decay) prior to using it in the distribution logic. If a 241Am sample exists, then decay the HDW data out to the 241Am Sample_Date, otherwise decay the data out to the TotalAlpha Sample_Date**. After performing the intermediate decay, perform the distribution using the following formula:</p> ${}^{nnn}\text{X} = {}^{nnn}\text{X}_{\text{HDW}} * \frac{[(\text{TotalAlpha}_{\text{SAMPLED}} * \text{Multiplier}) - \text{SUM} ({}^{nnn2}\text{X}_{\text{SAMPLED}} * \text{Multiplier})]}{[\text{SUM} ({}^{nnn1}\text{X}_{\text{HDW}})]}$ <p>where</p> <ul style="list-style-type: none"> ${}^{nnn}\text{X}$ = non-sampled Alpha emitting radionuclide ${}^{nnn}\text{X}_{\text{HDW}}$ = HDW Alpha emitting radionuclide Multiplier = Multiplier for the TotalAlpha value ${}^{nnn2}\text{X}_{\text{SAMPLED}}$ = Alpha emitting radionuclides which have sample values or calculated values (e.g. if UTOTAL is sampled, then the 232U, 233U... 238U calculated values are included)

	<p>Multiplier = Multiplier for the sampled value $^{nnn1}X_{HDW}$ = HDW Alpha emitting radionuclides which DO NOT have matching sample or calculated based radionuclides</p> <p>Lastly, decay ^{nnn}X out to the Standard_Decay_To_Date (1/1/2001).</p> <p>** Note, that it is possible that the HDW model vector has incorrect ratios of 241Pu to 241Am values. Hence if you “back decay” to a date prior to 1/1/1994, it is possible to come up with an intermediate decay value for 241Am that is negative. In this case, use the non-decayed 241Am value from the HDW vector instead of the negative value. This is not correct, but it is better than generating a negative inventory.</p>
3.5	All radionuclides in the database will be decayed to a single date. When this date is changed, then decay will be recalculated for all of the radionuclides
3.5.1	<p>First Order Decay: For a radionuclide that decays, but has not parent decaying into it, radioactive decay is calculated using the following formula:</p> $x(t) = X_o (e^{-\lambda\Delta t})$ <p>where</p> <p>$x(t)$ = concentration at time t X_o = initial concentration λ = $\ln(2) \div t_{1/2}$ $t_{1/2}$ = half life of the radionuclide Δt = time difference</p>
3.5.2	<p>Second Order Decay: For a radioisotope that decays (y), and has a parent decaying into it (x), the radioactive decay is calculated using the following formula:</p> <p>($x \rightarrow y \rightarrow z$)</p> $y(t) = \{ [\mu (X_o) (e^{-\lambda\Delta t} - e^{-\mu\Delta t})] \div (\mu - \lambda) \} + [Y_o e^{-\mu\Delta t}]$ <p>where</p> <p>$y(t)$ = concentration at time t of y Y_o = initial concentration of y X_o = initial concentration of x λ = $\ln(2) \div t_{1/2}$ of x μ = $\ln(2) \div t_{1/2}$ of y $t_{1/2}$ = half life of the radionuclide <i>(Use λ if $y(t)$ in mass, else use μ when $y(t)$ in Ci (or uCi/g))</i> Δt = time difference</p>
3.5.3	The following daughter radionuclides will use the second order decay equation to calculate their decay:

	<p style="text-align: center;">Parent Daughter</p> <p>241Pu → 241Am</p> <p>241Am → 237Np</p> <p>232Th → 228Ra</p> <p>231Pa → 227Ac</p> <p>93Zr → 93mNb</p>
3.5.4	In cases where there are many generations of decay, only the second order equation will be used. For example, if (w → x → y → z) then the effect of w will not be computed for y.
3.5.5	The half-life of 242mAm needs to be substituted for the 242Cm half-life to avoid having to add the 242mAm to the BBI analyte list. The 242mAm is the parent of 242Cm and continuously replenishes the 242Cm that decays (secular equilibrium). The currently accepted half-life for 242mAm is 141 years (Letter 7N100-00-009, B. A. Higley, 11/16/00). I will provide you with a copy of this letter. To be consistent with HTWOS, the BBIM should use the 141 year half life for 242Cm.
3.6	There is a definite order of computation when Analyte Dependents, HDW distribution , TotalAlpha distribution and radioactive decay and vector multipliers are involved.
3.6.1	If a waste component has been sampled for both UTOTAL and TotalAlpha, then the UTOTAL dependents are calculated first, followed by the TotalAlpha dependents.
3.6.2	If a waste component has been sampled for both 239/240Pu and TotalAlpha, then the 239/240Pu dependents are calculated first, followed by the TotalAlpha dependents.
3.6.3	If a waste component has been sampled for both 241Am and TotalAlpha, then the 241Am dependents are calculated first, followed by the TotalAlpha dependents.
3.6.4	If a waste component has been sampled for both 239Pu and 239/240Pu but not 240Pu, then the 239/240Pu value is used to calculate 240Pu.
3.6.4	<p>The decay and HDW distribution of UTOTAL, 239Pu, 241Am, 239/240Pu and TotalAlpha sample data will be done in the following order:</p> <ol style="list-style-type: none"> 1. Decay the HDW radionuclides to the sample date of UTOTAL (or 239Pu, 241Pu, 239/240Pu, TotalAlpha) 2. Create the HDW based dependent radionuclides using the decayed concentrations and the UTOTAL (or 239Pu, 241Pu, 239/240Pu, TotalAlpha) Sample concentration 3. Decay these dependent radionuclides from the sample date, to the “decay_to_date” <div style="text-align: center;"> </div>
3.6.5	

3.7	Allow multiple tanks to be entered into a work unit for publication. Current BBIM only allows one tank per work unit.
3.8	Waste volume inventories and material balances must be maintained (the Tank Farm Database (TFDB) functionality will be rolled into the system)
3.8.1	The data recorded for transfers between DSTs are sending tank, receiver tank, starting and ending dates for the transfer, kgal of waste transferred, kgal of flush water. (Also will need amounts of each component from sending tank)
3.8.2	There may be discrepancies between how much waste is reported to have been transferred to a tank, and the liquid level readings (from which tank volumes are calculated). These discrepancies are recorded in the TFDB as "UNKN". These discrepancies will be recorded as a volume variance in this system. This variance can be positive, negative or both.
3.8.3	Each time a volume variance changes, a reason should be given and recorded. A historical record of these should be saved.
3.8.4	Reasons for a volume variance change may be that no Transfer Data Sheet (TDS) or raw water data sheet has been generated, natural evaporation of liquid waste, increase due to slurry growth, decrease caused by gas release, or change in measuring device (instrument) used.
3.8.5	Where volume discrepancies exist, physical inventory (based upon liquid level measurements) is assumed correct.
3.8.6	When waste is transferred to a DST by way of a double-container receiver tank (DCRT), the waste may not show up in the DST for a month or more.
3.8.7	Transfers may be made to the 242-A Evaporator. (Find out more about this process)
3.8.8	BBIM currently only contains information on 177 241- tanks. Will need to add all tanks (63 additional?) (DCRT, Evaporators, Vaults, ...) to the database that can hold transferred waste.
3.8.9	Caustic liquid (NaOH) may be added to tanks.
3.8.10	Waste receipts of less than .2 inch (1000 gallons) are generally not logged. (why? How do we reconcile inventory in analytes?)
3.8.11	The following waste facilities may generate waste transfers to DSTs: PUREX; B Plant; T Plant; S Plant (222S Labs); PFP; 100 Area; 300 Area; 400 Area
3.8.12	Monthly reporting requirements – unknown since the document (HNF-SD-WM-SWD-038 REV. 0) describing these requirements provided to Battelle was incomplete.
3.8.13	The amount of waste transferred from a waste tank may be more that what is received in the sink tank because of waste being left in the transfer lines, or because of measurement error. You can also get more waste being received due to measurement error.
3.8.14	Formulas for the Component Density and Component Weight Percent Water are:

	$\rho_{\text{new}} = \frac{\left[(\rho_{\text{init}} \cdot V_{\text{init}}) + \sum_{\text{trans}} (\rho_{\text{trans}} \cdot V_{\text{trans}}) \right]}{V_{\text{final}}}$ <p>where ρ = density of the various phases transferred in. Assumed =1 for water V = volume of various phases. V_{trans} = volume of each stream transferred in. For evaporation, V_{trans} = volume evaporated = $(V_{\text{final}} - V_{\text{initial}})$ Wt\% = Wt% water of each stream. (=100% for water stream)</p> $\text{Wt\%H}_2\text{O}_{\text{new}} = \frac{\left[(\text{Wt\%}_{\text{init}} \cdot \rho_{\text{init}} \cdot V_{\text{init}}) + \sum_{\text{trans}} (\text{Wt\%}_{\text{trans}} \cdot \rho_{\text{trans}} \cdot V_{\text{trans}}) \right]}{V_{\text{final}} \cdot \rho_{\text{new}}}$
3.9	<p>Inventory calculations will have a “basis” that is determined for and analyte as follows:</p> <p>S : comes from a sample or samples of this tank E : determined using a sample or samples from a different tank. (note that if the inventory value is determined by averaging “vector values”, it only tanks one value coming from a different tank to meet this criteria). E : determined from Process Knowledge (note that if the value is determined by averaging “vector values”, it only takes one Process Knowledge type to meet this criteria). M : comes from a Hanford Defined Waste (HDW) Model value C : is calculated TS : template sample based TE : template engineering based S/E : If more than 10% of the inventory is based upon a vector_constituent of type “Sample” that has a qualifier of '<'.</p>
3.9.1	<p>If more than 10% of the inventory (Tank_Waste_Inventory) is based upon data that has a qualifier of '<', then the Tank_Inventory comment should contain “Upper Bounding Estimate”.</p>
3.10	<p>Relative Standard Deviations (RSD) will be associated with volumes, concentrations, densities and inventory.</p>
3.10.1	<p>The RSD for concentrations will be obtained from the Tank Characterization Database, and is equal to the Standard Deviation (Std Dev) / Concentration for a given analyte.</p>
3.10.2	<p>The RSD for density is associated with a standard waste phase within a type of tank (SST or DST). These values remain relatively fixed. The Density RSD may be overridden by the user.</p>
3.10.3	<p>The RSD for volume is associated the supernatant, sludge and saltcake phases within a tank. Note that waste phases of <i>Retained Gas – Salt Cake</i> and <i>Retained Gas – Sludge</i> are excluded from the RSD calculations (they are not considered to be part of the Salt Cake or Sludge phases). The Volume RSD are the same within each component, grouped by phase, i.e. all supernatant components will have the</p>

	<p>same Volume RSD (thought it may differ from the Sludge Volume RSD). Volume RSD for a phase is calculated as: (kL/inch waste) * (Phase Sigma) / (Phase Volume kL). Phase Sigma is fairly static and dependent upon which phases exist in the tank. They can be summed up in the following table (x indicates the presence of a phase).</p> <table border="1"> <thead> <tr> <th>Super</th> <th>Salt</th> <th>Sludge</th> <th>Super σ</th> <th>Salt σ</th> <th>Sludge σ</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>x</td> <td></td> <td></td> <td>0.25</td> <td>0</td> <td>0</td> </tr> <tr> <td></td> <td>x</td> <td></td> <td>0</td> <td>11.49</td> <td>0</td> </tr> <tr> <td></td> <td></td> <td>x</td> <td>0</td> <td>0</td> <td>4.32</td> </tr> <tr> <td>x</td> <td>x</td> <td></td> <td>6.5</td> <td>6.5</td> <td>0</td> </tr> <tr> <td>x</td> <td></td> <td>x</td> <td>6.5</td> <td>0</td> <td>6.5</td> </tr> <tr> <td></td> <td>x</td> <td>x</td> <td>0</td> <td>9.5</td> <td>9.5</td> </tr> <tr> <td>x</td> <td>x</td> <td>x</td> <td>6.5</td> <td>9.5</td> <td>9.5</td> </tr> </tbody> </table> <p>The Volume RSD may be overridden by the user (e.g., instead of being calculated, it is fixed).</p>	Super	Salt	Sludge	Super σ	Salt σ	Sludge σ				0	0	0	x			0.25	0	0		x		0	11.49	0			x	0	0	4.32	x	x		6.5	6.5	0	x		x	6.5	0	6.5		x	x	0	9.5	9.5	x	x	x	6.5	9.5	9.5
Super	Salt	Sludge	Super σ	Salt σ	Sludge σ																																																		
			0	0	0																																																		
x			0.25	0	0																																																		
	x		0	11.49	0																																																		
		x	0	0	4.32																																																		
x	x		6.5	6.5	0																																																		
x		x	6.5	0	6.5																																																		
	x	x	0	9.5	9.5																																																		
x	x	x	6.5	9.5	9.5																																																		
3.10.4	<p>For solid waste phases, the RSD for inventory is equal to: $((\text{density RSD})^2 + (\text{volume RSD})^2 + (\text{concentration RSD})^2)^{1/2}$</p> <p>For liquid waste phased, the RSD for inventory is equal to: $((\text{volume RSD})^2 + (\text{concentration RSD})^2)^{1/2}$</p>																																																						
3.11	Volume changes recorded in the Transfer Database (TXFR) will be used to adjust volumes and inventories in the BBIM.																																																						
3.12	TXFR and BBIM data will be linked via Waste Site ID and Standard Waste Phase.																																																						
3.13	When a tank is published, the What's New on the TWINS web will automatically be updated to show that the tank was published and the date that it was published.																																																						
3.14	Best-basis inventory shall be available as a separate menu item																																																						
3.15	The entry tool will be accessible via the TWINS2 interface																																																						
3.16	A means will be provided in the user interface for authorized users to enter all data and information associated with calculating and documenting the pedigree of the inventories for a standard constituent list for all 177 tanks.																																																						
3.17	This information and data consists at a minimum of the following: tank name, waste type or phase, waste density, constituent type (rad or chem), waste volume, analyte concentration.																																																						
3.18	In addition, a means will be provided to enter the sources of the above information (the reference).																																																						
3.19	The user interface which provides the ability to enter this information will be designed in a user-friendly manner, providing drop-down menus and graphical means to make the entry of information and data be as intuitive and convenient as possible.																																																						
3.20	The data entry user interface will prevent the entry of incomplete sets of data. For example, if the author wants to use a non-standard data source, he must also include																																																						

	an explanatory comment to document such an excursion from SOP.
3.21	The tool shall provide the capability to enter, store, and manage global inventory estimates for any analyte.
3.22	After completion of entry of input parameters, the database shall calculate the standard (best-basis) inventory for the given tank, waste type, and constituent using a standard algorithm in the database.
3.23	Following input of source parameters and calculation of the best-basis values, the author shall be provided the capability to store, review, and edit prior to final entry of the data.
3.24	Upon final entry of the best-basis inventory values and the associated source (input) values, the inventory and source data shall be made available via: tables in the TWINS 2 drop-down menu system.
3.25	Tables available for viewing shall include: the Standard Inventory tables (including uncertainty values), and tables and reports containing input data and values, input sources, assumptions, uncertainty information, revision histories, and global and tank-by-tank global values for comparison.
4	Schedule Requirements
4.1	
4.1	
5	Hardware Requirements
5.1	
5.2	
6	Software Requirements
6.1	
6.2	
7.0	Assumptions

3.0 System Requirements

The following table incorporates the functional requirements outlined above and turns them into system requirements for the development of the BBIM system.

Number	Requirement	Build
1	<i>Project Deliverables</i>	
1.1	Detailed schedule for completing the automated BBIM application by April 1, 1998.	1
1.2	Software Requirements Document	1
1.3	Design and Architecture Document	2

Number	Requirement	Build
1.4	Testing and Test Results	1,2
1.5	Configuration Control Document	2
1.6	Operational system	1,2
2	<i>Database Requirements</i>	
2.1	1.1.1 General Requirements	1.1.2
2.1.1	The BBIM database will be implemented under the same DBMS as TCD; it may be implemented as a separate database technically, but it can be fully linked and connected to related tables and views within the TCD family of databases.	1
2.1.2	All database tables and views within the BBIM can be accessed via the TWINS2-interface.	1
3	1.1.3 Structures	1.1.4
3.1.1	The ability must be provided to identify constituents as belonging to the “standard constituent list”.	1
3.1.2	The ability must be provided to add new standard constituents.	1
3.1.3	The ability must be provided to delete standard constituents.	1
3.1.4	Ability to store data by waste “components” consisting of standard waste phase (Retained Gas (Saltcake), Retained Gas (Sludge), Saltcake (Liquid & Solid), Saltcake Interstitial Liquid, Saltcake Solid, Sludge (Liquid & Solid), Sludge Interstitial Liquid, Sludge Solid, Supernatant, and a small number of others TBD in the best basis documents), and waste types. There are on the order of 20 standard waste types (SMMS1, R1, CWP, etc) which are also TBD in the documents.	1
3.1.5	The system can be easily expanded to contain new waste phases and waste types.	1
3.1.6	A tank can have multiple waste components.	1
3.1.7	A waste component can have a single waste phase.	1
3.1.8	A waste phase can contain multiple waste types.	1
3.1.9	The database must have the ability to keep “change records”	2
3.1.10	The database must have the ability to maintain a “public” and a “private” version or versions.	2
3.1.11	In the public version, select tables will be accessible via TWINS2	1
3.1.12	In the private version(s), the data will only be accessible via a special interface to authorized personnel.	1
3.1.13	The database will have the ability to be “uplevel” from private to public accessibility.	2
3.1.14	The database structure will be designed to provide the ability to “adjust waste tank	2

Number	Requirement	Build
	contents” to account for the transfer of material between tanks, use of the evaporator, dry well pumping, and retrievals. Such adjustments could be by defined waste type, by volume, or by waste tank.	
3.1.15	The database must be able to report global inventories for each analyte.	1
3.2	1.1.5 Calculations	1.1.6
3.2.1	Inventories are calculated for each Component/Waste-phase/Waste-type	1
3.2.2	Concentrations are calculated for each Component/Waste-phase/Waste-type	1
3.2.3	If a vector-constituent has a “<” qualifier, that qualifier is maintained along with the value in the vector, but it is not considered when calculating the inventory; the concentration is used as if there were no qualifier.	1
3.2.4	If a constituent value is reported as NR or NA, that flag is used as the Qualifier and the value is NULL.	1
3.2.5	Inventories can be calculated based on “representative vectors”, where each vector contains the representative <u>density</u> and the concentrations of a set of analytes.	1
3.2.6	A representative vector could be multiplied by a factor to account for concentration or inventory changes due to the sampled material having been run through the evaporator, or the sample having been on a wet basis to remove volume due to water. The factor is multiplied times each vector-constituent.	1
3.2.7	TBD	
3.2.8	Densities are always in ug/mL.	1
3.2.9	The ability to convert analyte concentrations into inventories using the following units: Inventory (in either kg or Ci) Volume (kgal , kL, or wt%), and Concentration (ug/g, ug/mL, uCi/g, uCi/L, M, kg, Ci, or wt%.) Database calculations automatically convert vector-constituent “concentrations” into inventories.	1
3.2.10	The database must provide the ability to select any vector-constituent within a vector for calculating an inventory; constituents not selected will not be used.	1
3.3	1.1.7 Queries	1.1.8
3.3.1	The database must provide the ability to query by tank for all standard analytes.	1
3.3.2	The database must provide the ability to query by waste phase for all standard analytes.	1
3.3.3	The database must provide the ability to query by waste type for all standard analytes.	1

Number	Requirement	Build
3.3.4	The database must have the ability to reproduce the vectors and vector-constituent.	1
3.3.5	Whenever an underlying inventory “source” parameter changes, the database software must be able to detect that the calculated inventory is out-of-date, and the database must be able to update the calculation. This may be automatic like a spreadsheet.	1
3.3.6	The database structures must accommodate the entry of tank data typically consisting of analyte concentrations by waste phase and/or waste type, or by homogenized sample concentrations (total tank).	1
3.3.7	The database must be able to differentiate between model-based and sample-based data sources.	1
3.3.8	A query to calculate and compare, by analyte in the standard analyte list the summation of tank-by-tank inventory versus the global inventory.	1
3.4	1.1.9 Reports	1.1.10
3.4.1	Input data report	1
3.4.2	Calculation report	1
3.4.3	Reference/Pedigree report	1
3.4.4	Revision history report	1
4	<i>User Interface Requirements</i>	
4.1	Ability to enter the best-basis document number	1
4.2	Ability to create/update waste phases and waste types	1
4.3	Ability to create/update/delete tank-waste-components and vectors. This data is entered directly from the paper Questionnaire form.	1
4.4	Ability to delete vector-associations from a tank-waste-component.	1
4.5	Ability to create/update/delete the vector and the vector-constituent data.	1
4.6	Ability to associate “representative vectors” with tank-waste-components	1
4.7	Ability to manually enter an inventory value, overriding the BBIM calculation; in such a case, an explanatory comment must be entered also.	1
4.8	Ability to select which constituents out of any given vector should be used to calculate the inventory for a given tank-waste-component. This includes “calculated” vectors, which are virtual vectors.	1
4.9	Ability to invoke a “tank waste contents adjustment” as a result of a transfer or retrieval of material from a waste tank.	2
5	<i>Security Requirements</i>	
5.1	The following user types will have varying permissions:	
5.1.1	Read-only user – this user will be able to view select tables via TWINS. No	1

Number	Requirement	Build
	password will be required for this type of user.	
5.1.2	BB Data Custodian – this is nominally a BB document author. This user has the ability to load data, change data, and perform basis selection and inventory calculation. This type of user will require a password to obtain access.	1
5.1.3	The BBIM database will be subject to the same standard operating procedures for the TCD family of databases: nightly full database backups, and daily increments disk backups and weekly full backups with long-term storage of archive tapes,	1
Number	Requirement	
6	Self-consistency calculations and adjustments	
6.11	The ability to adjust radionuclides known to have a given value at a specified date to a given new date (calculations based on Bob Watrous spreadsheet ezdk.xls).	
6.1.2	<p>Ability to perform OH (hydroxide) charge balance on a tank by tank basis. (calculations based on charbal.xls spreadsheet).</p> <p>For all constituents defined by the charbal.xls spreadsheet to be used in the calculation, compute the charge equivalent by multiplying the analyte mass times the analyte molecular weight times the analyte valence as defined by the charbal.xls spreadsheet. Calculate the summation of charge equivalents for each analyte. Calculate the mass of OH necessary to neutralize the summation of charge equivalents as follows:</p> <p>Mass of OH = (Molecular Weight of OH) * (Charge Equivalent Sum) * 1000</p> <p>This amount of OH results in a charge-neutral system within each waste tank.</p>	
6.1.3	<p>A. Isotope Estimation by Isotopic Distribution</p> <p>Estimation of specific isotopes based on the HDW Rev 4 isotope ratios: this is performed on a tank-by-tank basis for a specific set of radionuclides (calculations based on model Dave Place spreadsheet u&puds8.xls.). The ability to do this adjustment depends upon having specific sample-based data to work from as follows:</p> <p>^{232}U, ^{233}U, ^{234}U, ^{235}U, ^{236}U & ^{238}U are based on.....UTOTAL.</p> <p>^{238}Pu, ^{240}Pu, ^{241}Pu, and ^{242}Pu are based on..... ^{239}Pu</p> <p>^{243}Am, ^{242}Cm, ^{243}Cm & ^{244}Cm are based on..... ^{241}Am</p> <p>Sample-based source data that has been selected in a vector (ie, UTOTAL, ^{239}Pu, or ^{241}Am) takes precedence over any Model or Sample-Average-based isotope that has been selected in a vector. For example, if UTOTAL has been selected for inventory calculation, then all of the above-named uranium isotopes are calculated based on UTOTAL and the HDW ratios. If, for any reason, the uranium isotopes have been selected from the M-based vector, these isotope inventories must be over-ridden by the automatic calculations. This same principle applies to ^{239}Pu and ^{241}Am and their dependent isotopes. Similarly, if a Sample-Average based UTOTAL has been selected, then it over-rides any M-based isotopes that may have been selected.</p> <p>B. Isotope estimation by Total Alpha Distribution</p>	

	If sample-based UTOTAL , ²³⁹ Pu, or ²⁴¹ Am are not available, then a less-desirable form of isotopic estimations can be performed. If the analyte TotalAlpha exists, it may be used as a last resort to perform isotope estimation. This is done by first subtracting all alpha-emitting sample-based values from the gross TotalAlpha for the whole tank. Then, the remaining alpha is assigned to any of the previously identified isotopes that do not have sample-based data using the HDW Rev 4 ratios. radionuclides (calculations based on model Dave Place spreadsheet u&puds8.xls.).	
7	Dependent calculations and adjustments	
7.1.1	<p>Ability to automatically calculate the ⁹⁰Y inventory value based on the value of ⁹⁰Sr. The ratio of ⁹⁰Sr to ⁹⁰Y should be 1:1 (inventory ⁹⁰Sr = inventory ⁹⁰Y). ⁹⁰Y inventory should be referenced to the ⁹⁰Sr inventory in the comment field. ⁹⁰Y should have the same basis as the ⁹⁰Sr value.</p> <p>The only exception will be cases where both the ⁹⁰Y and ⁹⁰Sr are model-based values. In this instance, the HDW model values are derived and may not always be in the 1:1 ratio (ie if both are M-based, leave them alone).</p>	
7.1.2	<p>Ability to automatically calculate the ^{137m}Ba inventory value based on the value of ¹³⁷Cs. The ratio of ¹³⁷Cs to ^{137m}Ba should be 1:0.946 (inventory ^{137m}Ba = 0.946 x inventory ¹³⁷Cs). ^{137m}Ba inventory should be referenced to the ¹³⁷Cs inventory in the comment field. ^{137m}Ba should have the same basis as the ¹³⁷Cs value.</p> <p>The only exception will be cases where both the ^{137m}Ba and ¹³⁷Cs are model-based values. In this instance, the HDW model values are derived and may not always be in the 1:0.946 ratio. (ie if both are M-based, leave them alone).</p>	
7.1.3	<p>Ability to automatically calculate the inventory values of ²⁴³Cm and ²⁴⁴Cm based on the value of ^{243/244}Cm. ²⁴³Cm inventory = 0.04 x inventory ^{243/244}Cm and ²⁴⁴Cm inventory = 0.96 x ^{243/244}Cm (per HDW Rev. 4, pg. E-6). The hybrid inventory value (^{243/244}Cm should be decomposed into its respective isotopic (²⁴³Cm and ²⁴⁴Cm) inventories.</p>	
8	Input-to-Output Constituent Mapping Adjustments	
8.1.1	<p>Ability to accept either “TIC” or “CO3” or “TIC as CO3” as input, while converting to a different constituent name, “TIC as CO3” for output. If the input constituent name is TIC, it must be multiplied by a factor of 60/12 = 5 for conversion. (Total Inorganic Carbon, or TIC is measured in the lab, but is assumed to exist in the form of CO3 in the waste. Therefore, the concentration must be multiplied by the ratio of the molecular weights). Both input and output values must be maintained.</p>	
8.1.2	<p>Ability to accept either “P” or “PO4” or “P as PO4” as input, while converting to a the constituent name, “PO4” for output. If the input constituent name is P, it must be multiplied by a factor of 95/31 for conversion. (Phosphorous, or P is measured by the ICP analysis method in the lab, but is assumed to exist in the form of PO4 in the waste. Therefore, the concentration must be multiplied by the ratio of the molecular weights). However, PO4 can also be directly measured in the lab by the IC method. Both input and output values must be maintained.</p>	
8.1.3	<p>Ability to accept either “S” or “SO4” or “S as SO4” as input, while</p>	

	converting to a the constituent name, "SO4" for output. If the input constituent name is S, it must be multiplied by a factor of 96/32 for conversion. (Sulfur, or S is measured in the lab, but is assumed to exist in the form of SO4 in the waste. Therefore, the concentration must be multiplied by the ratio of the molecular weights). However, SO4 can also be directly measured. Both input and output values must be maintained.	
9	Volume Adjustments	
9.1.1	Ability to adjust the contents of a waste tank to account for transfers and retrievals. A user interface will be provided for controlling this process.	
10	CCB Enhancements	
10.1.1	<p>Capabilities to support the CCB in managing and tracking Best-Basis inventory changes and keeping the TCD Best-Basis standard inventory tables in synch with the Best-Basis documents. The following requirements are included:</p> <ol style="list-style-type: none"> 1. Tracer data checker (validates that Tracer fields meet specification and are compatible with TCD) 2. Tracer data loading, staging, review, reporting & approval capabilities. 3. Tracer uplevel to TCD capability by CCB log #. 4. Tracer format report or query from BBIM. <p>Ability to display CCB Change log, including proposed changes in TWINS2 (this includes an Access-based "maintenance" interface for updating the log).</p>	
11	Model Enhancements	
11.1.1	Ability to specify multiple vector-constituents within a single waste phase such that these constituents are <i>added</i> rather than exclusive-or'd. (ie the ability to tell the model to add the value of aluminum from several tables).	
12	Structure Enhancements and Data Loading	
12.1.1	Ability to load and store Solubility/Insolubility factors by tank by analyte for water wash and caustic leach. Data will be supplied in Excel spreadsheets.	
12.1.2	Ability to load and store Solubility/Insolubility factors by tank by analyte for water wash and caustic leach. Data will be supplied in Excel spreadsheets.	
12.1.2	Load the HDW Rev 5 radionuclide concentrations for all tanks	
13	Reports	
13.1.1	A report (or query) showing, for any given vector, which tanks inventories are based on that vector.	
13.1.2	BBIM change log	
13.1.3	Tracer reports: all/pending/approved	

BBI – Change Request Tracking Tool Requirements

Change Request 366.2: Upload ULD Spreadsheet Results to the BBI Source Terms

See: <http://twins/crtt/CRTT.asp?ID=496&Type=CR>

Requirement #	Description
366.2	<p>A mechanism will be developed to transfer the ULD spreadsheet output to the BBI Source Term area to make the data available to users. A warning dialog box will be provided to avoid inadvertent publishing of data and users will be restricted to BBI personnel.</p> <p>JUSTIFICATION: Spreadsheets have replaced ULD calculations previously performed by TWINS. These changes were made to improve performance (USOF calculations) and to isolate these safety calculations from the TWINS database. The results of the quarterly ULD calculations for tanks with changed BBIs need to be uploaded to the Source Term area to be readily available to users.</p>

Change Request 366.1: Upload USOF Spreadsheet Results to the BBI Source Terms

See: <http://twins/crtt/CRTT.asp?ID=495&Type=CR>

Requirement #	Description
366.1	<p>A mechanism will be developed to transfer the USOF spreadsheet output to the BBI Source Term area to make the data available to users. A warning dialog box will be provided to avoid inadvertent publishing of data and users will be restricted to BBI personnel. The headers in the USOF report will also be revised to read TEEL-O, PAC-1, PAC-2 and PAC-3.</p> <p>JUSTIFICATION: Spreadsheets have replaced USOF calculations previously performed by TWINS and the SOF Program respectively. These changes were made to improve performance (USOF calculations) and to isolate these safety calculations from the TWINS database. The results of the quarterly ULD/USOF calculations for tanks with changed BBIs need to be uploaded to the Source Term area to be readily available to users.</p>

Change Request 360: Calculate Free Hydroxide from Nitrite inventories in the BBIM

See: <http://twins/crtt/CRTT.asp?ID=489&Type=CR>

Requirement #	Description
360	<p>Calculate OH from NO₂ in the BBIM using the ratios of OH/NO₂ found for each tank in the SOF OH Trace Inputs table, and publish the results. Full requirements found in CR360Requirements.doc</p>

Best Basis Inventory – Change Request Tracking Tool Requirements

Detailed Change Request 360 Requirements: Calculate Free Hydroxide from Nitrite inventories in the BBIM

Req #	Requirement
1	Insert a calculated Free Hydroxide inventory value when none exists in a waste phase when a Nitrite inventory value is inserted.
2	Insert a calculated Free Hydroxide inventory value when none exists in a waste phase when a Nitrite inventory value is updated.
3	Update an existing a calculated Free Hydroxide inventory value when a Nitrite inventory value is updated.
4	Do not update an existing a non-calculated Free Hydroxide inventory value when a Nitrite inventory value is updated.
5	Do not update an existing a non-calculated Free Hydroxide inventory value when a Nitrite inventory value is inserted.
6	Delete a calculated Free Hydroxide inventory value when a Nitrite inventory value is deleted.
7	Do not delete a non-calculated Free Hydroxide inventory value when a Nitrite inventory value is deleted.
8	Verify that the updates to calculated Free Hydroxide are performed in accordance to the specification to the formula and data source provided in the change request.
9	The Formula column in the Calculation Detail report will contain the text “NO ₂ * ESP OH / ESP NO ₂ ” for each calculated Free Hydroxide.
10	The Calculation column of the Calculation Detail report will contain the data values used to calculate the inventory for each calculated Free Hydroxide.
11	Free Hydroxide is to be calculated for all waste layers in compliance with requirements 1 through 8.
12	<p>The Best-Basis Inventory Notice paragraph on the bottom of the http://twins.pnl.gov/twinsdata/forms/datamenu.htm web page will state:</p> <p>The best-basis inventories are updated quarterly based on new sample data, waste transfers and reconciliation efforts. Refer to the BBI Change Control Status for a list of pending and approved changes to the best-basis inventory data set. The standard BBI analytes include 25 chemicals and 46 radionuclides. Radionuclides are decay-corrected to January 1, 2008. Inventories for other analytes, including free hydroxide, may be found in the Best Basis Calculation – Supplemental Analytes report.</p>
13	<p>The last line(s) of the Calculation Detail report will state:</p> <p>NOTE: Inventories of other analytes, including free hydroxide may be found in the Best Basis Calculation – Supplemental Analytes report.</p>
14	The Free Hydroxide calculations will be verified in the production BBIM (sandbox) prior to publishing to the BB_Published database.
15	The calculated Free Hydroxide inventories will be published for all tanks. If the tank is not scheduled for publication in its entirety, then only the Free Hydroxide inventory is published. Effective Inventory dates are to remain unchanged for these tanks.
16	An automated routine will be developed to force recalculation of all BBIM free hydroxide inventories and to publish the resulting free hydroxide inventories to the

Best Basis Inventory – Change Request Tracking Tool Requirements

	BBI. The Effective Dates, BBI Document and Doc Release dates shall use the current published values (rather than values from the BBIM database).
17	After publication the TWINS What's New for BBI should have an entry stating "Inventory for the Free Hydroxide supplemental analyte are now estimated for all waste components".

Change Request 341: Conservative Waste Model Report

See: <http://twins/crtt/CRTT.asp?ID=468&Type=CR>

Requirement #	Description
341	<p>One of the tasks in our statement of work dealt with programing a comparison of new BBI updates with the conservative waste model. Attached are two spreadsheets. The first demonstrates the calculation methodology for this comparison. The second demonstrates the sample output for the comparison that we would like to see. New inputs for X-sectional areas and for minimum sub-critical ratios are highlighted in yellow on the second worksheet. The example is for sludge solids in one tank. The calculation should be done for each phase and each waste type. Thus there will be calculations done for sludge liquid, sludge solid, supernate, saltcake solids, and saltcake liquid for each waste type that exists in a tank. I would suggest that we put the output on the BBI drop down of TWINS similarly to wash and leach factors or BBI calc detail report. The intent is to update this calculation each time the BBI is updated.</p> <p>Tom, I'm assuming that you would be the one to program this since you also did the programing on the criticality query. Please let me know if this is not case or if you need additional information.</p>

Change Request 337: Create Separate RDR inventory area for calculating and publishing final tank inventories

See: <http://twins/crtt/CRTT.asp?ID=464&Type=CR>

Requirement #	Description
341	This will be parallel to the BBIM, use BBI vectors, and create an independent tank summary report.

Change Request 334: New BBIM Constituent

See: <http://twins/crtt/CRTT.asp?ID=461&Type=CR>

Requirement #	Description
334	We need the new constituent Cresol (m & p) available for selection in the BBIM by the end of the quarter. Thank you, Alexis

Change Request 333: Volume and Density RSDs

See: <http://twins/crtt/CRTT.asp?ID=460&Type=CR>

Requirement #	Description
333	<p>In TWINS Best-basis Inventory Maintenance tool, allow a manual entry option for Volume and Density RSDs.</p> <p>JUSTIFICATION: Change needed to allow for RSDs based on alternate measurement methods. Ex. Tanks that have completed retrieval are using AutoCAD and video to determine the final volume.</p>

Change Request 332: RSD Overestimated for Liquids

See: <http://twins/crtt/CRTT.asp?ID=459&Type=CR>

Requirement #	Description
332	<p>Revise the calculation of RSDs for liquid inventories to eliminate the contribution from the density RSD.</p>

Change Request 325: Modify BBIM to Calculate A 155eu and 152eu Inventory Based On 154eu And Th From 232th When Data Are Unavailable or Not Selected

See: <http://twins/crtt/CRTT.asp?ID=451&Type=CR>

Requirement #	Description
325	<p>Modify the BBIM to calculate a 155Eu inventory for each waste phase based on the distribution vector ratio of 155Eu to 154Eu. This is similar to how uranium isotopes are calculated. A concentration may be selected from the radiobuttons and an inventory calculated directly for each waste phase. If a 155Eu concentration is unavailable or is not selected, calculate the inventory based on multiplying the 154Eu within the waste phase by the distribution vector ratio of 155Eu/154Eu.</p> <p>Similarly, modify the BBIM to calculate at 152Eu inventory based on the distribution vector ratio of 152Eu to 154Eu.</p> <p>Modify BBIM to calculate a Thorium inventory for each waste phase based on the 232Th inventory and the specific activity. This is similar to the calculation for Sr from 90Sr.</p> <p>JUSTIFICATION: This change will increase rigor and efficiency for the BBI process by having the computer calculate inventory for 155Eu, 152Eu and Thorium.</p>

Change Request 317: Adjust Mass Balance Equation

See: <http://twins/crtt/CRTT.asp?ID=442&Type=CR>

Requirement #	Description
---------------	-------------

Best Basis Inventory – Change Request Tracking Tool Requirements

317	<p>The BBIM mass balance equation would become:</p> <p>sum the concentrations for Standard BBI analytes – 0.273 * oxalate concentration Al concentration / 26.98 * 17.01 * 3 Fe concentration / 55.847 * (16 17.01) Si concentration / 28.0855 * 16 * 2 Zr concentration / 91.224 * 17.01 * 4 component wt% water * 10⁴.</p> <p>JUSTIFICATION:</p> <p>We could improve the BBIM mass balance check by adding charge balancing mass for Fe, Si and Zr. This should improve the mass balance for iron-rich sludges, diatomaceous earth layers and the zirconium cladding wastes.</p>
-----	--

Change Request 296: Component Density

See: <http://twins/crtt/CRTT.asp?ID=296&Type=BUG>

Requirement #	Description
296	Item #4 from CR241 : Add Wt% Water to the Radiobutton selection forms to allow selection of the preferred density and wt% water for each waste phase. No inventories will be calculated from these selections. Report the selected values in the TWINS Best Basis Inventory/Best Basis Calculation Detail-Supplemental Analytes query.

Change Request 227: Add BBI TWINS Data Query Menu Item “Specific Activities”

See: <http://twins/crtt/CRTT.asp?ID=286&Type=CR>

Requirement #	Description
227	Add BBI TWINS Data Query Menu Item for “specific activities.”

Change Request 126: BBIM: Increase the Size Formula & Calculation Fields

See: <http://twins/crtt/CRTT.asp?ID=126&Type=BUG>

Requirement #	Description
126	Can you make the summary report volumes show up to only one decimal place? With the calculated volumes, I'm getting 7 digits past the decimal (AP-102 supernatant= 470.8993964 kL). I don't think our estimate is really that accurate. Also, for the calc detail report, can you expand the fields for the formula and calculation. Its getting truncated so I can't see all the changes I'm making.

Change Request 118: AN-103 BBI Glitch: alpha isotopes (Pu, Am, Cm), three "bases" (E/C/S) are showing up when only two phases are present

See: <http://twins/crtt/CRTT.asp?ID=118&Type=BUG>

Requirement #	Description
118	I was looking at the calc detail report in BBIM for AN-103. For the alpha isotopes (Pu, Am, Cm), three "bases" (E/C/S) are showing up when there are only two waste

Best Basis Inventory – Change Request Tracking Tool Requirements

phases. Inventories seem to be okay.

On a similar topic, AN-107 calc detail shows several "bases" as S based for some of the radionuclides, when it shows as process knowledge for each of the waste phases. It almost appears as though the bases (E/S/C) isn't being updated.

Need to modify the logic of the Tank_Waste_Inventory triggers so that 'Upper Bounding Estimate' appears in the comment field for Process Knowledge vectors when more than 10% contain '<' qualifiers. However, the basis should not be 'S/E'; should be 'E'.

Data in tank_waste_inventory:

241Am	Sample	p_Decay_And_Distribute 10	65427
241Am	Process Knowledge	p_Decay_And_Distribute 10	73145
241Pu	Calculated	p_Decay_And_Distribute 10	NULL
241Pu	Process Knowledge	p_Decay_And_Distribute 10	73143
242Cm	Calculated	p_Decay_And_Distribute 10	NULL
242Cm	Process Knowledge	p_Decay_And_Distribute 10	73147

Data in Tank_Inventory:

241Am	S/E
241Pu	E/C/S
242Cm	E/C/S

sql code:

```
select constituent_name, basis from tank_inventory ti, constituent c
where waste_site_id = '241-AN-103'
and ti.con_id = c.con_id
and molecular_weight > 240
order by sort_order
```

```
select constituent_name, twi.basis, Inv_routine, vc_id
from tank_waste_inventory twi, constituent c, tank_waste_component twc
where twi.con_id = c.con_id
and molecular_weight > 240
and twc.twt_id = twi.twt_id and waste_site_id = '241-an-103'
order by sort_order
```

Change Request 195: Statistical Calculation Change (Volume RSD)

See: <http://twins/crtt/CRTT.asp?ID=222&Type=CR>

Best Basis Inventory – Change Request Tracking Tool Requirements

Requirement #	Description
195	Change the way the Volume RSD is calculated so that it becomes a dynamic calculation, rather than a static, pre-set calculation. The BBI software will henceforth compute the total volume for each of the standard waste phases, convert that to inches of level, then use a factor depending on which tank it is to come up with the RSD. This would likely be done in the new waste transfer interface (TXFR_BBIM). See notes.

Change Request 151: Automate Generation of Best Business Inventory Uncertainty Numbers

See: <http://twins/crtt/CRTT.asp?ID=150&Type=CR>

Requirement #	Description
151	Using the concentration means and relative standard deviations (RSDs) stored in the Means and Variances template in TWINS; the density RSDs found in table 2-1 of RPP-6924, Rev. 0 and the volume and volume uncertainties for each waste phase for each tank which in the future will be placed in the Best-Basis Inventory Model (BBIM) vector creation tool or other appropriate place in the model; program the calculation of inventory RSD using the mathematical instructions found on page 5 of RPP-6924. (Note: in the future, all changes or entries of volume data in BBIM will be accompanied by uncertainty values for that volume data. However, uncertainties recently made available for 33 tanks will be entered into the BBIM vector creation tool or other appropriate place in BBIM to enable a test file to be created.) Program, if necessary, in the BBIM vector creation tool or other appropriate place in BBIM, a space for an uncertainty value to accompany each volume value. Provide a test file for inspection and approval by the CHG statisticians prior to placement into production.

Change Request 120: Phased Based Inventory

See: <http://twins/crtt/CRTT.asp?ID=119&Type=CR>

Requirement #	Description
120	Programming changes to the BBI tool are needed to perform alpha isotope calculations at the waste phase level (i.e. separate calculations for supernatant, salt cake and sludge phases).

Change Request 285: Changes to BBI Work Unit/Prep Area

See: <http://twins/crtt/CRTT.asp?ID=383&Type=CR>

Requirement #	Description
285	In the BBIM Tool Work Units Area: <ol style="list-style-type: none">1. Provide a confirmation message when deleting a work unit. "Are you sure you wish to delete this work unit?" It is easy to hit the wrong key and accidentally delete a work unit when you intended to edit it. This

Best Basis Inventory – Change Request Tracking Tool Requirements

will help prevent that. - **DONE**

2. Provide the ability to Open/Edit the comments for all analytes on the “Prepare for Publication” screen, with a single update button. It is extremely tedious to have to open the comment field for each analyte then wait to update before editing the next. This is particularly tedious when doing multiple updates, and every entry takes you back to the top of the screen. CAAT has this feature and it seemed to work well. Refer to Article at <http://www.microsoft.com/belux/nl/msdn/community/columns/ceulemans/multirecs.mspx> for ideas on how to best do this. - **DONE**
3. Provide the ability to Open/Edit all analytes on the “Vector Constituents” Screen. - **DONE**
4. Automatically populate the comment field for 90Y and 137mBa with the following: “Based on 90Sr” or “Based on 137Cs,” respectively.
5. If Sr is calculated from 90Sr, automatically fill in the comment field with “Based on 90Sr.” If oxalate is calculated, automatically fill in the comment field with “Based on ratio with TOC.”
6. Automatically populate the comment fields for total alpha, U, Pu, Am, and Cm isotopes with the previously existing comment.
7. The “Upper bounding estimate” comment, generated by a computer check, currently only applies to Sample based results. The comment needs to be applied to both Sample and Process Knowledge based results. - This is already occurring. Kristine will try to find a case where this is not happening (TSO - 11/14/2004)

Change Request 382: Revise Programming for Uploading of USOFs

See: <http://twins/crtt/CRTT.asp?ID=505&Type=CR>

Requirement #	Description
382	<p>The programming for uploading of new USOF values to the Source Term area of BBI is currently linked to the BBI update history. The programming will be revised to upload records and data directly from the Output worksheet of the USOF Calculation Spreadsheet without any links to or data import from the BBI update history.</p> <p>Justification: The current configuration creates the possibility of incorrect waste volumes and missing records if the USOF upload occurs after other changes to the BBI have been made. This potential error source needs to be eliminated.</p> <p>NOTE: The Due Date is a <i>hard date</i>. This needs to be in place prior to the next uploading of the USOF data.</p>

CR 382 Details:

Change Request 382: Revise Programming for Uploading of USOFs Requirements

See: <http://twins.pnl.gov/webforms/crttview/crtt.asp?ID=505&Type=CR>

Requirements

The form references are located by following these instructions:

1. Log onto the TWINS/Applications development site: <http://twinsdevui/apps>
2. Click on the **BBI** link in the top menu
3. Click on the **Upload USOF** link in the **Publishing** section of the left menu

A data input spreadsheet is attached in the **Documentation** section of the Change Request 382. The name of the file is **USOFCalc1.1_FY10_Q3.xlsm**.

Requirement #	Description
CR382.01	The Upload USOF Data to TWINS form shall copy the USOF data from the input file to TWINS where it can be viewed in the Toxicological Unit Sum of Fractions query (http://twinsdevui/twinsdata/forms/BuildQuery.aspx?SourceName=bb_published.dbo.p_TWINS_SOF).
CR382.02	The following field values shall be copied from the Output worksheet to TWINS: <ol style="list-style-type: none"> 1. Tank 2. Tank Type 3. Waste Phase 4. Waste Type 5. Volume 6. USOF Run Date 7. USOF TEEL-0 8. USOF PAC-1 9. USOF PAC-2 10. USOF PAC-3
CR382.03	The USOF TEEL-0, USOF PAC-1, 2 and 3 values shall be reported in the TWINS query to three significant digits.
CR382.04	If a record already exists in the TWINS query that has matching Tank, Tank Type, Waste Phase, Waste Type and USOF Run Date values as those in the input spreadsheet, then the TWINS query Volume and USOF TEEL and USOF PAC values will be updated with the values from the spreadsheet (a duplicate record shall not be created).
CR382.05	If a record already exists in the TWINS query that has matching Tank, Tank Type, Waste Phase, Waste Type and USOF Run Date values as those in the input spreadsheet, then a new record is added to the TWINS query.
CR382.06	If a new record is added to the TWINS query that has a USOF Run Date that is greater than other records with matching Tank, then the Most Recent selection from the TWINS Query should pick up this record and any other records for this Tank that have this (maximum) USOF Run Date.

Resolve Tool Requirements

RESOLVE Tool Requirements

Resolve Requirements

1. Modify Resolve Tool to Apply “Reviewed: See Comment” When Appropriate
 - Desk instruction TFC-ENG-CHEM-D-32, “Review and Resolution of TWINS Data” includes the following Review Status definitions.
 - Reviewed = Data not qualified or rejected, no reason given.
 - Reviewed: See Comment = Data not qualified or rejected, reason given for not rejecting the data.
 - Change Resolve so that when a result is reviewed, submitted for approval (using the “Submit for Approval” button), and approved, the Review Status is changed to:
 - “Reviewed” when the Review Comment field is blank
 - “Reviewed: See Comment” when there is a comment in the Review Comment field.
2. Document the criteria used by Resolve/p_check_rules to set status's

Computer Checking of Analysis Result Records

Overview of the Computer Check process

Selection of Records to be Checked.

The Computer Check routine, “p_check_rules”, runs each night. The first step counts all primary, duplicate and triplicate results by Sample, constituent, and method to identify multiple results. Any multiple results that are found (unless they already have a review_status of "DATA SUSPECT: SEE COMMENT" or “DATA RESOLUTION IN PROGRESS”) are updated to review_status "DATA RESOLUTION IN PROGRESS" with the review_comment, "Sample has multiple primary results" or a similar comment for duplicate and triplicate results.

For the remainder of the Computer Check routine, only records that have not previously been reviewed or that have been modified after the time they were reviewed (last_modified_date is later than review_date) are selected for evaluation. In addition, records must be primary, duplicate, or triplicate results from projects that are not identified as "altered".

Since several Change Requests have been incorporated into the computer checking routine since it was first developed in 2001, some records may have been evaluated with a somewhat different set of rules than those currently in use.

When the Computer Check is completed, all records that were evaluated will have one of four review_status values: "DATA SUSPECT: SEE COMMENT", "REVIEWED: NOT FOR

RESOLVE Tool Requirements

INVENTORY", "DATA RESOLUTION IN PROGRESS", or "REVIEWED". The reviewer is always "ComputerCheck". Those records with the status "DATA RESOLUTION IN PROGRESS" will appear in the Resolve Tool for further review by the Tank Coordinator.

Hierarchy of Rule Application

First, all results from Samples with aggregation level 'Liner Liquid' are found. These are all assigned the status "REVIEWED: NOT FOR INVENTORY" and no additional checking is applied to them.

Next, rules that result in the status "DATA SUSPECT: SEE COMMENT" are applied. Once the DATA SUSPECT status is established for a record, no further rules checking is done on that record. Consequently, all of the possible applicable comments may not appear on the record – only the comment associated with the DATA SUSPECT designation is added.

Records that have the status "DATA RESOLUTION IN PROGRESS" continue to go through any remaining additional checks and may accumulate additional comments.

Records that have not been flagged by any of the rules checks are assigned the status "REVIEWED".

Treatment of Records that were Previously Reviewed but later Modified

As noted above, records are re-checked if they have been modified since the time they were reviewed. If the name of a person, rather than 'Computercheck', appears as the Reviewer in the record, steps are taken to preserve the existing review information. The original review information is saved to a temporary location. On completion of the entire Computer Check routine, if the Computer Check assigned review_status is different than what previously existed, or if the original record contained a review_comment by the reviewer, then both the old and new review_status as well as old and new review_comment are concatenated and displayed in the review_comment field. The new review_status of the record is then set to "DATA RESOLUTION IN PROGRESS" so that the Tank Coordinator can make the decision about the current validity of the original comments and whether those comments should be retained or discarded.

Rules Applied and Review Status Assignment

Upper and Lower Limits

Results are compared to the upper and/or lower limits for the constituent and physical state (liquid or solid) that exist in the Limits Table. (Appendix A.) Records outside the limits are assigned the review_status and comment that are specified in the Limits Table unless the test is for the upper limit and the record has a "u" qualifier code.

RESOLVE Tool Requirements

Use of GEA method for Pu

Applies to constituents Pu239/240, Pu/Am241, Pu/Am242, U/Pu238, CM/Pu244, 13982-10-0, 14119-32-5, 144119-33-6, 15117-48-3, 15411-92-4, 7440-07-5, and Alpha/PUAMRT. If a GEA analysis method was used, the status is, "DATA SUSPECT: SEE COMMENT" with the comment, "GEA is not the preferred method-see AEA or ICP/MS results"

Ammonia

Applies to constituents 7664-41-7, NH3-N, 14798-03-9, 12125-02-9. If the analysis_date_time of the primary result is more than 180 days later than the sample_date_time, the review_status is, "DATA SUSPECT: SEE COMMENT" with the comment, "Time between sampling and analysis exceeds 180 days." Since primary and duplicate results are always analyzed on the same day, the same status and comment are applied to the duplicate result.

If no sample_date is available, the sampling_begin_date is used. If the difference between Sampling_begin_date and analysis_date is greater than 180 days, the status is, "DATA RESOLUTION IN PROGRESS" and the comment is, "Time between sampling and analysis may exceed 180 days". The same status and comment are applied to the duplicate result.

If neither the sample_date nor the sampling_begin_date is available, then the status for the Primary and Duplicate results is, "DATA RESOLUTION IN PROGRESS" and the comment is, "Time between sampling and analysis not determined."

Standard Units

If the result does not have standard value units, the status is set to, "DATA RESOLUTION IN PROGRESS" with the comment, "Not converted to standard units".

Zirconium Crucible

Applies to constituents 13967-71-0, 7440-67-7, Zr/Nb-95 where the analysis method group is ICP:F and the Lab Sample ID ends with the letter "z". Status is, "DATA RESOLUTION IN PROGRESS" and the comment is, "Potential bias from use of Zr crucible.

Nickel Crucible

Applies to constituents 13981-37-8, 14336-70-0, and 7440-02-0 where the analysis method group is ICP:F and the Lab Sample ID ends with the letter "f". Status is, "DATA RESOLUTION IN PROGRESS" and the comment is, "Potential bias from use of Ni crucible."

pH > 12.5

If pH is greater than 12.5, status is, "DATA RESOLUTION IN PROGRESS" and the review comment is, "pH is greater than 12.5".

pH and OH rules

If pH and OH are measured on the same sample, OH is calculated from the average value of pH for the sample and compared to average of the measured OH for the sample. If the measured and calculated OH values differ by a factor of 2 or more then both the pH and OH results for

RESOLVE Tool Requirements

this Sample are assigned status, "DATA RESOLUTION IN PROGRESS" and the comment, "OH and pH results are not consistent".

The formula used for OH in ug/g is $[17 * (10^{**3})] [10 E+14/10^{**pH}]$.

The formula used for OH in ug/L is $[17 * (10^{**6})] [10 E+14/10^{**pH}]$.

TOC and oxalate rules

Calculate average values for TOC and oxalate (338-70-5) from the same sample.

Compare the two values – if TOC average value is not $>$ oxalate average value * 24/88 status is, "DATA RESOLUTION IN PROGRESS" and the review comment is, "TOC equivalent of oxalate exceeds TOC result" for all TOC and oxalate results for this sample.

Eu154 (15585-10-1) and Eu155 (14391-16-3)

Compare Eu154 and Eu155 results from the same sample. If the results differ by more than a factor of 10 then the status is, "DATA RESOLUTION IN PROGRESS" and the review comment is, "154Eu to 155Eu ratio out of expected range."

Qualifier Codes B, E, Q, a, b, c, d, and e

Primary result with any of the above qualifier codes as well as the related duplicate and triplicate results, whether or not they have qualifier codes, are assigned status, "DATA RESOLUTION IN PROGRESS" and the following review comments:

B – "Result may be affected by blank contamination."

E – "Result is over the calibration range of the instrument."

Q – "Result is qualitative."

a – "Standard recovery is outside customer defined range."

b – "Spike recovery is outside customer defined range."

c – "RPD is outside customer defined range."

d – "RSD is outside customer defined range."

e – "Percent difference between sample and serial dilution result is greater than 10%."

More than one qualifier comment may be applied to each result.

LiBr intrusion

Applies to Li, Br, %water, and SpecGravity constituents from Core Samples. (Samples with the aggregation level "Liner Liquid" are not included.)

- %water values for each Sample are averaged. Samples with no %water value available are not checked for LiBr intrusion.
- The Li values and the Br values are each averaged for the Sample.
- SpecGravity values are averaged for the Sample. If no SpecGravity is available for the sample then the average SpecGravity for the whole Core is used. If this also is not available, a value of 1.0 is used.
- Li and Br intrusion is calculated by the following formulas:
 $10000 * \text{Average Li} * 0.9914 / (\text{ave_wt_water} * 2080)$ for Solids
 $10000 * \text{Average Li} * 0.9914 / (\text{ave_wt_water} * 2080 * \text{ave_sp_gr})$ for Liquids
(The same formulas are used with the average Br values).

RESOLVE Tool Requirements

- For each sample the greater value (Li or Br intrusion) is used to assign a review_status to all Li, Br, %water, and Specgravity results from the Sample:
 - 10% to < 50% - REVIEWED: SEE COMMENT with Comment "LiBr accounts for > 10% and < 50% of the wt% water in the sample."
 - 50% or more - DATA RESOLUTION IN PROGRESS with Comment "LiBr Accounts for >50% of wt% water in the Sample."
3. Resolve Limits
- Update the concentration limits in the Resolve tool with the attached file. These limits will only apply to new data and not to data currently in TCD. Leave all density limits as is. JUSTIFICATION: Updating the Resolve tool concentration limits will be completed periodically after a significant amount of new data has been uploaded to TCD.

Resolve Tool – Change Request Tracking Tool Requirements

Change Request 177: Resolve Tool Using Informal Name for Reviewer

See: <http://twins/crtt/CRTT.asp?ID=177&Type=BUG>

Gist of requirement: The reviewer's name should not be his/her log-in name (e.g., H0012345) but his/her common name (e.g., Jane Doe).

Change Request 290: Change "Review Status" and "Review Comment" to prevent samples being flagged by the Resolve tool

See: <http://twins/crtt/CRTT.asp?ID=292&Type=CR>

Requirement #	Description
244	<p>Liner Liquid samples in TWINS TCD should not have any computer checking rules applied and should not be used in any averages:</p> <ul style="list-style-type: none"> • Enter "REVIEWED: NOT FOR INVENTORY" in the Review Status field to prevent computer checking rules from being applied to Liner Liquid samples. • Enter "Liner Liquid" in the Review Comment field. • Liner Liquid samples should not be used in any averages or calculations.

Change Request 351: Create Reprogram the Resolve Tool

See: <http://twins/crtt/CRTT.asp?ID=479&Type=CR>

Requirement #	Description
351	<p>After reviewing the UI changes performed in the Original Change Request, Duc and company came up with significant changes. Therefore the original change request has been closed and a new one opened to record the new requirements and work.</p>

Change Request 351 Details:

The purpose of this document is to define the requirements necessary to facilitate the reprogramming of the RESOLVE Tool for Change Request 351.

RESOLVE Record Review -- 241-AW-105 -- Data Resolution in Progress

Lab Sample	Analyte	Value	Result Type	Comment	Disposition	Status	Update
804D00640	Hydroxide	300	ug/ml - primary_result	Ends comment.	Accept Reject Not BBI		<input type="checkbox"/>
804D00640	PH Measurement	11.5	unitless - duplicate_result		Accept Reject Not BBI		<input type="checkbox"/>
804D00640	PH Measurement	11.5	unitless - primary_result		Accept Reject Not BBI		<input type="checkbox"/>
804D00640	Hydroxide	410	ug/ml - primary_result	ED 1	Accept Reject Not BBI		<input type="checkbox"/>
804D00640	PH Measurement	11.5	unitless - primary_result	OR and PH results are not consistent. / pH is greater than 12.5 / OR result preferred.	Accept Reject Not BBI		<input type="checkbox"/>
804D00640	Hydroxide	410	ug/ml - primary_result	ED 1	Accept Reject Not BBI		<input type="checkbox"/>
804D00640	PH Measurement	11.5	unitless - primary_result	OR and PH results are not consistent. / pH is greater than 12.5 / OR result preferred.	Accept Reject Not BBI		<input type="checkbox"/>

Figure 1

User Interface Components and Appearance

1. Include a Title that contains the ID of the Tank which is being reviewed.
2. In a grid, include the following columns:
 - a. Include a column for *Lab Sample* similar to the column shown in figure 1.
 - b. Include a column for *Analyte* similar to the column shown in figure 1.
 - c. Include a column for *Value* similar to the column shown in figure 1.
 - d. Include a column for *Result Type* similar to the column shown in figure 1.
 - e. Include a column for *Comment* similar to the column shown in figure 1.
 - f. Include a *Disposition* column which contains:
 - i. A button labeled *Accept*.
 - ii. A button labeled *Reject*.
 - iii. A button labeled *Not for BBI*.
 - g. Include a *Status* column which can contain the same Status icons as the current Record Review form.
 - h. A checkbox for selecting this record to be updated.
3. Include appropriate column headings.
4. Order the above columns, from left to right shall be: *Lab Sample*, *Analyte*, *Value*, *Result Type*, *Comment*, *Disposition*, *Status*, *Update*.
5. Minimize white space between columns (current Record Review form uses too much white space).

Current Record Review form

RESOLVE - Record Review

241-AW-105 - DATA RESOLUTION IN PROGRESS

Lab Sample	Analyte	Value	Result Type	Comment	Update	Status
S04T000451	Hydroxide	3830 ug/mL	primary_result	<input type="text" value="Indi comment"/>	<input type="checkbox"/>	Be
S04T000451	Hydroxide	3660 ug/mL	duplicate_result	<input type="text" value="OH and PH results are not consistent."/>	<input type="checkbox"/>	
S04T000452	Hydroxide	4160 ug/mL	primary_result	<input type="text" value="ET 1"/>	<input type="checkbox"/>	+
S04T000453	Hydroxide	4350 ug/mL	primary_result	<input type="text" value="IVO 2"/>	<input type="checkbox"/>	+
S04T000451	PH Measurement	13.5 unitless	duplicate_result	<input type="text"/>	<input type="checkbox"/>	+
S04T000451	PH Measurement	13.5 unitless	primary_result	<input type="text"/>	<input type="checkbox"/>	+
S04T000452	PH Measurement	13.5 unitless	primary_result	<input type="text" value="OH and PH results are not consistent."/>	<input type="checkbox"/>	-
S04T000453	PH Measurement	13.5 unitless	primary_result	<input type="text" value="OH and PH results are not consistent."/>	<input type="checkbox"/>	-

To edit a comment paste the comment below. Edit as desired, then select the checkbox on the records to which the revised comment is to be applied. Click on the Apply

New Comment

Back

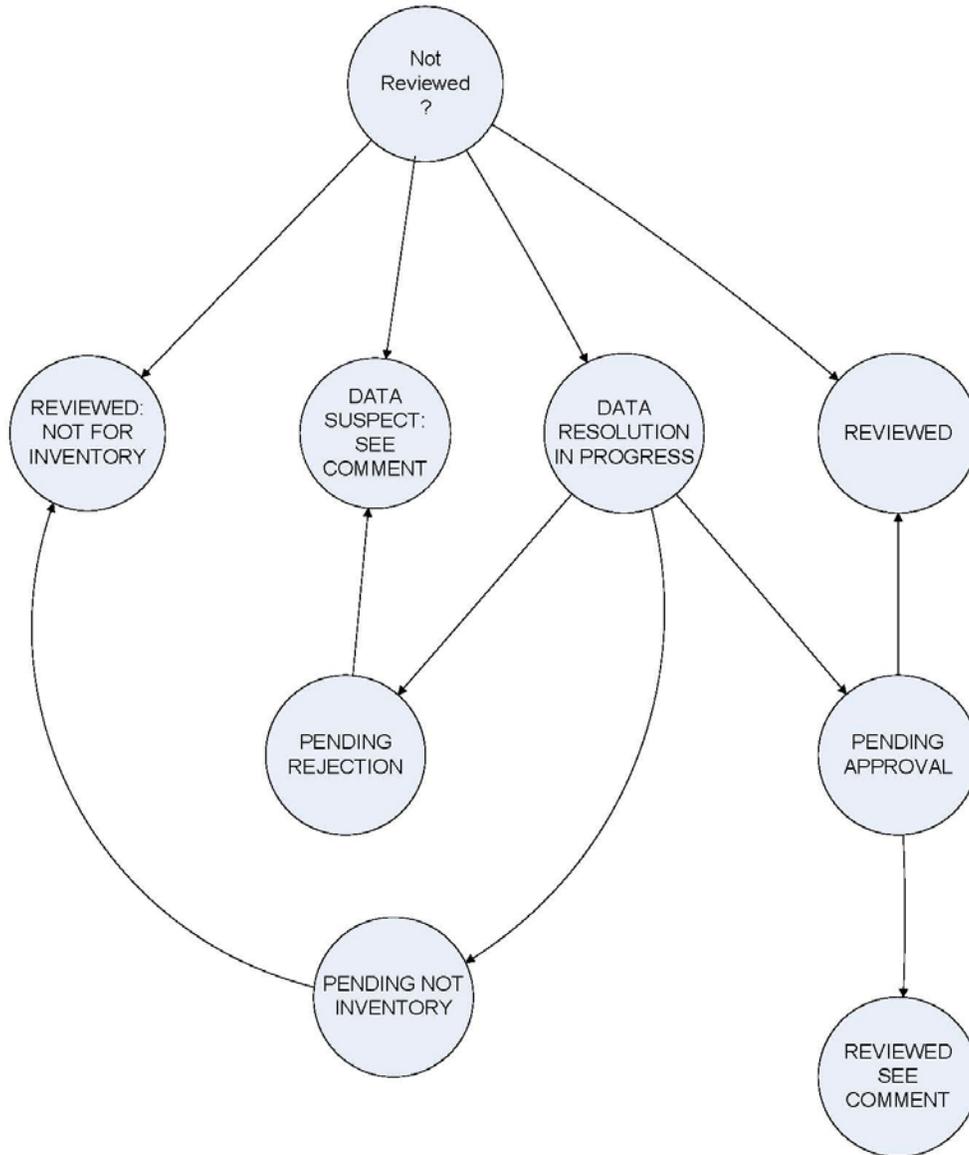
Figure 2

6. On the form, include the following:
 - a. A button (or checkbox) that allows the user to select all records to be updated
 - b. A button (or checkbox) that allows the user to de-select all records that are to be updated
 - c. A button labeled *Submit* that submits the selected records to be updated
 - d. A button labeled *View Limits* that redirects the user to the form that show the RESOLVE parameters and limits
 - e. A button labeled *View Log* that redirects the user to a form that displays the last 12 months of review status changes for the currently selected tank

User Interface Functionality

1. The form will display the same *Data Resolution in Progress* records as does the current Record Review form for any given tank.
2. Records in the form may be sorted by Lab Sample.
3. Records in the form may be sorted by Analyte.
4. Each Comment field should display 3 or 4 rows of text.
5. Each Comment field should be able to be scrolled to view more text.
6. Submitting the data should update the dispositions of the selected records with:
 - a. review_status: Approved | Pending | Not for BBI selections
 - b. review_comment: the text from the Comment textbox
 - c. reviewer: the name of the person submitting this change
 - d. review_date: timestamp of the review status change

Resolve Status Flow Diagram



Change Request 159.2: REMOVE ITEMS FROM TABLE in Resolve

See: <http://twins/crtt/CRTT.asp?ID=200&Type=CR>

Details below:

Tank Waste Information Network System (TWINS) Change Order Form

CHANGE CONTROL NUMBER: 159--REVISION 2F

DATE CHANGE FILED: 03/11/02

TYPE OF CHANGE: TWINS

TITLE OF CHANGE: REVISE RESOLVE HHF ALGORITHM AND COMMENTS

DESCRIPTION OF CHANGE:

- **Update the calculation for wt % water contribution from hydrostatic head fluid (HHF) intrusion to average the primary and duplicate, and triplicate where applicable, results for each of Li, Br, and % water concentrations for each sample (from core sampling events – grab and auger samples are excluded) prior to calculating % water from HHF. Then program the tool to use the following rules (1) to (3) to assign review status and review comments to analytical results.**

(1) If the LiBr results indicate $< 10\%$ contribution to the wt% water from HHF, all analytical results associated with the sample event name (core name), segment name, aggregation level, and sample portion can be marked as "REVIEWED", providing no other issues are associated with the analytical results.

(2) If the LiBr results indicate $\geq 10\%$ and $\leq 50\%$ contribution to the wt% water from HHF, mark the Li, Br, and wt% water analytical results (primary, duplicate, and triplicate results) associated with the sample event name (core name), segment name, aggregation level, and sample portion as "REVIEWED: SEE COMMENT", with the comment "LiBr accounts for $\geq 10\%$ and $\leq 50\%$ of wt% water in sample".

(3) If the LiBr results indicate $> 50\%$ contribution to the wt% water from HHF, send the Li, Br and wt% water results to the data resolution file. If it is determined that the data are suspect, the wt% water will be flagged by the data reviewer as "Data Suspect, See Comment" with the comment "LiBr accounts for $>50\%$ of wt% water in sample". This comment will cause the tool to attach this flag and comment to all analytical results (primary, duplicate, and triplicate results) associated with this sample portion (aggregation level).

- **Add the following dictionary definition:**

LiBr

Lithium bromide traced water associated with tank sampling events, usually as hydrostatic head fluid (HHF) for the core sampling trucks, or to soften tank waste.

INITIATOR: M.R. ADAMS

DATE COMPLETION REQUIRED: asap

SYSTEM DOCUMENTATION REQUIRED: see original CR

DOES CHANGE IMPACT DATA LOADING: no

APPROVALS:

M.R. ADAMS

J.G. FIELD

TXFR Tool Requirements

TXFR Tool – BBIM_TXFR Interface* Requirements (DRAFT)

(*Interface refers to the interaction between TXFR and BBIM)

2/28/2002

1. Waste Types in TXFR will be mapped to a “Standard Waste Phase”.
2. Waste Phases in BBIM will be mapped to a “Standard Waste Phase”.
3. TXFR and BBIM records are related to each other by their Waste_Site_ID and “Standard Waste Phase”.
4. It is possible for more than one BBIM waste component to be related to a single TXFR transaction. In this case, recording the change of the TXFR transaction in the BBIM will be equally divided among the components that relate the the TXFR transaction.
5. The TXFR transactions for any given BBIM Waste_Site_ID + “Standard Phase” will have a flag indicating whether the transaction has been reflected in the BBIM or not.
6. For TXFR transactions that involve the transfer of waste between two tanks, the flag (item #5) shall be visible to the user whether he/she is viewing the TXFR transactions as related to the source tank, or the sink tank.
7. The Interface shall have the capability to “undo” or “rollback” a TXFR transaction that has previously been used to update BBIM data. The “undo” will change the BBIM back to its previous state, and set the flag referred to in item #5 above to indicate that no update of the BBIM has occurred.
8. TXFR “Adjustment” type volume changes will cause inventory in the BBIM to change by a factor of $(\Delta \text{TXFR Volume} + \text{BBIM Volume}) / \text{BBIM Volume}$. Concentrations in the BBIM will remain the same.
9. TXFR “Evaporation” type volume changes will cause concentrations to increase in the BBIM by a factor of $(\Delta \text{TXFR Volume} + \text{BBIM Volume}) / \text{BBIM Volume}$. Inventories will remain the same.
10. TXFR “Flush Water” type volume changes will cause concentrations to decrease in the BBIM by a factor of $(\text{BBIM Volume} - \Delta \text{TXFR Volume}) / \text{BBIM Volume}$. Inventories will remain the same.
11. TXFR “Transfer” type transactions will cause the source tank’s volume and inventories to decrease by a factor of $(\text{BBIM Volume} - \Delta \text{TXFR Volume}) / \text{BBIM Volume}$; its concentrations will remain the same. The sink tank’s volume and inventory will increase by the same amount the the source tank’s inventory is decreased; its concentration may increase or decrease depending upon the makeup of the waste transferred to it.
12. TXFR “Generator” type transactions are not handled by the interface since there is no data available to characterize the generator waste stream. Currently, only volumetric data exists
13. TXFR “Destination” type transactions are not handled since there is no “Destination” data structure to its addition of waste.
14. When TXFR “Transfer” type transactions are recorded in the BBIM, and single component for the sink tank Waste_Site_ID and “Standard Phase” will be created with a Waste_Type of “Waste Transfer” to record the input of waste to the sink tank. In keeping with the structure of the BBIM, a vector will also be created that is linked to the component for the purposes of characterizing the component’s constituent concentrations.

- The individual constituents of the vector will have concentrations recorded in units of ug/mL or uCi/L. The vector will have a density of 1.0 ug/mL recorded.
15. Only one “Waste Transfer” component is allowed per Standard Waste Phase in a tank. If multiple transfers occur into the same Standard Waste Phase of a tank, then the existing “Waste Transfer” component will be updated rather than creating a new one.
 16. When a tank to tank transfer is “undone”, then if multiple transfers had occurred in the sink tank, the “Waste Transfer” component and its linked vector are updated to reflect their state prior to the transfer. If there is only one transfer that has gone into this “Waste Transfer” component, then the component and its vector are deleted.
 17. Changes to the BBIM from TXFR transactions will be recorded as both deltas to the volumes and inventories, and as snapshots (bbim state prior to transaction; bbim state after transaction).
 18. BBIM volume/inventory snapshots should be stamped with time periods, e.g. the snapshot is valid from one date/time to another date/time.
 19. In the TXFR database, transfers are managed at the tank level. This means that even though only one phase in the tank (e.g. supernatant) may be transferred from one tank to another, records for the other phases will exist with delta volumes equal to zero. The BBIM_TXFR application will display these records (delta volume = 0) and mark them with a bbi flag equal to ‘I’ (Ignore).

Test Plan

1. Test to see that all of the TXFR records are mapped to one of the Standard Waste Phases (this will probably not be the case, but it will be good to note the fact).
2. Test to see that all of the BBIM tank_waste_components map to one of the Standard Waste Phases (this is not the case, but it will be good to note the fact).
3. Test to make sure the the TXFR records are mapped correctly to the BBIM components. (Compare the contents of the tcd..TXFR_BBIM which is populated by tcd..p_pop_TXFR_BBIM with the tables: tcd..TXFR_Event, tcd..TXFR_Waste_Type_Event, tcd..waste_type, tcd..TXFR_std_phase, bbim..tank_waste_component and bbim..waste_phase_mapping).
4. When more than one BBIM component maps to a TXFR transaction, test to make sure that each of the components has its inventory/concentration changed appropriately. The change to each component should be equal to (TXFR delta)/ (# of components).
5. The initial state of the tcd..TXFR_BBIM.BBI_Flag and tcd..TXFR_Waste_Type_Event.bbi_flag should all be set to 'N'. Verify that for each operation type (e.g. tank to tank transfer, evaporation, flush water added, and adjustment) performed in the Access application, the flags in these two tables are set to 'Y'. 'Undo' the operations and verify that they are set back to 'N'.
6. Update the BBIM for a Tank to Tank transfer. Verify that both of the TXFR records (one for the source; one for the sink) have had their bbi_flags set to 'N'.
7. Test the change in state when using the Undo function:
 - a. Prior to the Undo, record the state of the component(s) involved in tables:
 - i. Tcd..TXFR_BBIM (bbi_flag should be set to 'Y')
 - ii. Tcd..TXFR_Waste_Type_Event (bbi_flag(s) should be set to 'Y')
 - iii. Bbim..TXFR_BBIM_Hist_Vol_Chg (record should exist)
 - iv. Bbim..TXFR_BBIM_Hist_Inv_Chg (records should exist)
 - v. Bbim..TXFR_BBIM_Hist_Waste_Add (record should exist if the transaction was for a Tank to Tank transfer)
 - vi. Bbim..TXFR_BBIM_Hist_Inv_Add (records should exist if the transaction was for a Tank to Tank transfer)
 - vii. Bbim..Tank_Waste_Component (note volume)
 - viii. Bbim..Tank_Waste_Inventory (note inventory and concentrations)
 - ix. Bbim..Tank_Waste_Component_Hist ("OPEN" record (End_Of_Period = NULL) should match bbim..Tank_Waste_Component record). (Note volume of previous state for this component).
 - x. Bbim..Tank_Waste_Inventory_Hist ("OPEN" records should match bbim..Tank_Waste_Inventory records). (Note inventories/concentrations of previous state for this component).
 - b. Following the Undo, record the state of the component(s) involved in tables:
 - i. Tcd..TXFR_BBIM (bbi_flag should be set to 'N')
 - ii. Tcd..TXFR_Waste_Type_Event (bbi_flag(s) should be set to 'N')
 - iii. Bbim..TXFR_BBIM_Hist_Vol_Chg (record should not exist)
 - iv. Bbim..TXFR_BBIM_Hist_Inv_Chg (records should not exist)
 - v. Bbim..TXFR_BBIM_Hist_Waste_Add (record should not exist)
 - vi. Bbim..TXFR_BBIM_Hist_Inv_Add (records should not exist)

- vii. Bbim..Tank_Waste_Component (volume should match volume from previous state recorded in step ix above)
 - viii. Bbim..Tank_Waste_Inventory (inventory and concentrations should match inventories and volumes from previous state recorded in step x above)
 - ix. Bbim..Tank_Waste_Component_Hist (record that was in the “OPEN” state recorded in step ix above should have been deleted; “previous” record should now be the “OPEN” record).
 - x. Bbim..Tank_Waste_Inventory_Hist (records that were in the “OPEN” state recorded in step x above should have been deleted; “previous” records should now be the “OPEN” record).
8. Test the TXFR “Adjustment” type volume changes:
- a. Prior to the Adjustment, record the state of the component(s) involved in tables:
 - i. Tcd..TXFR_BBIM (bbi_flag should be set to ‘N’)
 - ii. Tcd..TXFR_Waste_Type_Event (bbi_flag(s) should be set to ‘N’)
 - iii. Bbim..Tank_Waste_Component (note volume)
 - iv. Bbim..Tank_Waste_Inventory (note inventory and concentrations)
 - v. Bbim..Tank_Waste_Component_Hist (“OPEN” record (End_Of_Period = NULL) should match bbim..Tank_Waste_Component record).
 - vi. Bbim..Tank_Waste_Inventory_Hist (“OPEN” records should match bbim..Tank_Waste_Inventory records).
 - b. Following the Adjustment, record the state of the component(s) involved in tables:
 - i. Tcd..TXFR_BBIM (bbi_flag should be set to ‘Y’).
 - ii. Tcd..TXFR_Waste_Type_Event (bbi_flag(s) should be set to ‘Y’).
 - iii. Bbim..Tank_Waste_Component (volume should change by Δ TXFR Volume).
 - iv. Bbim..Tank_Waste_Inventory (inventory should change by $(\Delta \text{TXFR Volume} + \text{BBIM Volume}) / \text{BBIM Volume}$. Concentrations should remain unchanged).
 - v. Bbim..Tank_Waste_Component_Hist (“OPEN” record (End_Of_Period = NULL) should match bbim..Tank_Waste_Component record).
 - vi. Bbim..Tank_Waste_Inventory_Hist (“OPEN” records should match bbim..Tank_Waste_Inventory records).
 - vii. Bbim..TXFR_BBIM_Hist_Vol_Chg (a record should be created that reflects the volume change for this event, and should be linked via the Main_ID field back to the tcd..TXFR_Event table).
 - viii. Bbim..TXFR_BBIM_Hist_Inv_Chg
9. Test the TXFR “Evaporation” type volume changes:
- a. Prior to the Evaporation, record the state of the component(s) involved in tables:
 - i. Tcd..TXFR_BBIM (bbi_flag should be set to ‘N’)
 - ii. Tcd..TXFR_Waste_Type_Event (bbi_flag(s) should be set to ‘N’)
 - iii. Bbim..Tank_Waste_Component (note volume)
 - iv. Bbim..Tank_Waste_Inventory (note inventory and concentrations)
 - v. Bbim..Tank_Waste_Component_Hist (“OPEN” record (End_Of_Period = NULL) should match bbim..Tank_Waste_Component record).

- vi. Bbim..Tank_Waste_Inventory_Hist (“OPEN” records should match bbim..Tank_Waste_Inventory records).
- b. Following the Evaporation, record the state of the component(s) involved in tables:
 - i. Tcd..TXFR_BBIM (bbi_flag should be set to ‘Y’).
 - ii. Tcd..TXFR_Waste_Type_Event (bbi_flag(s) should be set to ‘Y’).
 - iii. Bbim..Tank_Waste_Component (volume should decrease by Δ TXFR Volume).
 - iv. Bbim..Tank_Waste_Inventory (concentrations should increase by $(\Delta$ TXFR Volume + BBIM Volume) / BBIM Volume. Inventories should remain unchanged).
 - v. Bbim..Tank_Waste_Component_Hist (“OPEN” record (End_Of_Period = NULL) should match bbim..Tank_Waste_Component record).
 - vi. Bbim..Tank_Waste_Inventory_Hist (“OPEN” records should match bbim..Tank_Waste_Inventory records).
 - vii. Bbim..TXFR_BBIM_Hist_Vol_Chg (a record should be created that reflects the volume change for this event, and should be linked via the Main_ID field back to the tcd..TXFR_Event table).
 - viii. Bbim..TXFR_BBIM_Hist_Inv_Chg
- 10. Test the TXFR “Flush Water” type volume changes:
 - a. Prior to the Flush Water, record the state of the component(s) involved in tables:
 - i. Tcd..TXFR_BBIM (bbi_flag should be set to ‘N’)
 - ii. Tcd..TXFR_Waste_Type_Event (bbi_flag(s) should be set to ‘N’)
 - iii. Bbim..Tank_Waste_Component (note volume)
 - iv. Bbim..Tank_Waste_Inventory (note inventory and concentrations)
 - v. Bbim..Tank_Waste_Component_Hist (“OPEN” record (End_Of_Period = NULL) should match bbim..Tank_Waste_Component record).
 - vi. Bbim..Tank_Waste_Inventory_Hist (“OPEN” records should match bbim..Tank_Waste_Inventory records).
 - b. Following the Flush Water, record the state of the component(s) involved in tables:
 - i. Tcd..TXFR_BBIM (bbi_flag should be set to ‘Y’).
 - ii. Tcd..TXFR_Waste_Type_Event (bbi_flag(s) should be set to ‘Y’).
 - iii. Bbim..Tank_Waste_Component (volume should increase by Δ TXFR Volume).
 - iv. Bbim..Tank_Waste_Inventory (concentrations should decrease by $(\text{BBIM Volume} - \Delta$ TXFR Volume) / BBIM Volume. Inventories should remain unchanged).
 - v. Bbim..Tank_Waste_Component_Hist (“OPEN” record (End_Of_Period = NULL) should match bbim..Tank_Waste_Component record).
 - vi. Bbim..TXFR_BBIM_Hist_Vol_Chg (a record should be created that reflects the volume change for this event, and should be linked via the Main_ID field back to the tcd..TXFR_Event table).
 - vii. Bbim..TXFR_BBIM_Hist_Inv_Chg
- 11. Test the TXFR “Transfer” type volume changes:
 - a. Prior to the Transfer, record the state of the component(s) involved in tables:

- i. Tcd..TXFR_BBIM (bbi_flag should be set to 'N')
 - ii. Tcd..TXFR_Waste_Type_Event (bbi_flag(s) should be set to 'N')
 - iii. Bbim..Tank_Waste_Component (note volume of both the source and sink tank components)
 - 1. Note volume for source component
 - 2. If a 'Waste Transfer' component already exists (transfer has previously occurred into this component), then note volume for this sink component.
 - iv. Bbim..Tank_Waste_Inventory
 - 1. Note the inventories and concentrations of the source component
 - 2. If a 'Waste Transfer' component already exists then note the inventories and concentrations.
 - v. Bbim..Tank_Waste_Component_Hist ("OPEN" record (End_Of_Period = NULL) should match bbim..Tank_Waste_Component record).
 - vi. Bbim..Tank_Waste_Inventory_Hist ("OPEN" records should match bbim..Tank_Waste_Inventory records).
- b. Following the Transfer, record the state of the component(s) involved in tables:
- i. Tcd..TXFR_BBIM (bbi_flag should be set to 'Y').
 - ii. Tcd..TXFR_Waste_Type_Event (bbi_flag(s) should be set to 'Y').
 - iii. Bbim..Tank_Waste_Component.
 - 1. Source Component volume should decrease by Δ TXFR Volume.
 - 2. If a 'Waste Transfer' component did NOT exist, then a new one should be created having the waste phase equal to the TXFR standard waste phase, and its volume = $\text{abs}(\Delta \text{ TXFR Volume})$. If a 'Waste Transfer component already existed, the its volume should be increased by Δ TXFR Volume.
 - iv. Bbim..Tank_Waste_Inventory (concentrations should decrease by $(\text{BBIM Volume} - \Delta \text{ TXFR Volume}) / \text{BBIM Volume}$. Inventories should remain unchanged).
 - 1. The inventories of the source component should be decreased by $(\text{BBIM Volume} - \Delta \text{ TXFR Volume}) / \text{BBIM Volume}$. The concentrations should remain unchanged.
 - 2. The inventories of the sink component should be increased by the same amount that the source component inventories were decreased. If a new 'Waste Transfer' component was created for this transfer, then its concentrations should equal the concentrations of the source component's concentrations. Whether or not this is a new 'Waste Transfer' component, its concentrations should equal its $(\text{Inventory} / \text{Concentration})$. (Note: $\text{Inv} = \rho * \text{Vol} * \text{Conc}$ where $\rho = 1.0 \text{ g/mL}$).
 - v. Bbim..Tank_Waste_Component_Hist ("OPEN" record (End_Of_Period = NULL) should match bbim..Tank_Waste_Component record).
 - 1. For the source component, its "OPEN" record should be closed (End_Of_Period set to current date/time), and a new "OPEN" record should created (Start_Of_Period set to current date/time & End_Of_Period set to NULL). The volume of the closed record

- should reflect the volume prior to the transfer; the volume of the new “OPEN” record should reflect the volume after the transfer has occurred.
2. If a new ‘Waste Transfer’ component was created, then a new “OPEN” Tank_Waste_Component_Hist record should be created to reflect the sink component after the transfer has occurred. If a ‘Waste Transfer’ component already existed prior to the transfer, then its “OPEN” record should be closed (End_Of_Period set to current date/time), and a new “OPEN” record should be created (Start_Of_Period set to current date/time & End_Of_Period set to NULL). The volume of the closed record should reflect the volume prior to the transfer; the volume of the new “OPEN” record should reflect the volume after the transfer has occurred.
- vi. Bbim..Tank_Waste_Inventory_Hist (“OPEN” records should match bbim..Tank_Waste_Inventory records).
 1. For the source and sink components, their closed Tank_Waste_Inventory_Hist records should reflect its inventories prior to the transfer. Its new “OPEN” history records should equal the current Tank_Waste_Inventory records. Note that the sink component will only have a “closed” record if this is not the first transfer into this component.
 - vii. Bbim..TXFR_BBIM_Hist_Vol_Chg (a record for the source component should reflect the reduction in volume and a description of the event. It should also link back via Main_ID to the tcd..TXFR_BBIM and tcd..TXFR_Event tables).
 - viii. Bbim..TXFR_Hist_Inv_Chg (records for the source component should reflect the reduction in inventory)
 - ix. BBIM..TXFR_Hist_Waste_Add (a record for the sink component should reflect the increase in volume).
 - x. BBIM..TXFR_Hist_Inv_Add (records for the sink component should reflect the inventory added).
12. No test necessary.
 13. No test necessary.
 14. Select a waste component that does not have a ‘Waste Transfer’ waste type (has not been the recipient of tank to tank transfer waste) for this test. This component will become the sink component. Perform a tank to tank transfer into this component and verify that:
 - a. A new tank_waste_component record is created with a waste_type = ‘Waste Transfer’
 - b. A new vector is created which characterizes this transferred waste.
 - c. A new representative_vectors record is created which links the vector to the sink component.
 - d. New vector_constituent records are created for the vector. There should be one vector_constituent record for each source tank component constituent.
 - e. New tank_waste_inventory records for the source tank component. There should be one inventory record for each vector_constituent record.

- f. In preparation for test #16, record the values for this sink component's records in tables:
 - i. Tank_waste_component
 - ii. Tank_waste_component_hist
 - iii. Tank_waste_inventory
 - iv. Tank_waste_inventory_hist
 - v. Representative_vectors
 - vi. Representative_vectors_hist
 - vii. Vector
 - viii. Vector_constituent
15. Select the same waste component chosen in item 14 above to be the sink component. Perform a tank to tank transfer into this component and verify that:
 - a. The existing tank_waste_component record with a waste_type = 'Waste Transfer' is used as the sink (volume should increase by Δ TXFR volume).
 - b. The existing vector is used to characterize the waste.
 - c. Existing representative_vector still used to link the vector to the tank_waste_component.
 - d. The existing vector is modified to properly characterize the waste (e.g. concentrations should reflect the weighted average concentrations of multiple transfers). Verify that if a constituent that was not perviously in the sink tank (none from prior transfers) has been added when it exists for this transfer.
 - e. Existing tank_waste_inventory records are updated (inventory should be increased by the amount being added; concentrations should be adjusted to a weighted average of the transferred waste). New tank_waste_inventory records should be added when they did not previously exist.
16. 'Undo' the tank to tank transfer in item 15 above and compare the values recorded in item 14f above to their current values. They should be identical.
17. No test for this: previous test validate this item.
18. Perform numerous operations on a give tank waste component (evaporation, transfer, flush water added...) and then select all of the records for this component from the tank_waste_component_hist table. Verify that the date/time periods are ordered and not overlapping (e.g. the End_Of_Period value of the previous record equals the Start_Of_Period of the next record), and that the last history record has an End_Of_Period set to NULL.

TXFR Tool – Change Request Tracking Tool Requirements

Change Request 275: TXFR_BBIM: Need to redo how it handles Sink Tanks when Phase changes

See: <http://twins/crtt/CRTT.asp?ID=275&Type=BUG>

Requirement #	Description																																																																																																																
275	<p>Problem when there are multiple source tank records from different phases all going into supernatant in the sink. Problem with data is shown below: txfr_waste_type_event contents:</p> <table border="1"> <thead> <tr> <th>txfr_event_id</th> <th>source</th> <th>sink</th> <th>waste_type</th> <th>recv_waste_type</th> <th>volume delta kgal</th> <th>wte volume kgal</th> <th>txfr wte ID</th> </tr> </thead> <tbody> <tr> <td>39057</td> <td>Salt Cake Solid</td> <td>Salt Cake Solid</td> <td>SC</td> <td>SC</td> <td>0</td> <td>305</td> <td>147482</td> </tr> <tr> <td>39057</td> <td>Salt Cake Liquid</td> <td>Supernatant</td> <td>IL</td> <td>DN</td> <td>-2</td> <td>14</td> <td>147483</td> </tr> <tr> <td>39057</td> <td>Sludge</td> <td>Sludge</td> <td>SL</td> <td>SL</td> <td>0</td> <td>2</td> <td>147484</td> </tr> <tr> <td>39057</td> <td>Sludge</td> <td>Supernatant</td> <td>IS</td> <td>DN</td> <td>-1</td> <td>0</td> <td>147485</td> </tr> <tr> <td>39058</td> <td>Sludge</td> <td>Sludge</td> <td>SL</td> <td>SL</td> <td>0</td> <td>23</td> <td>147486</td> </tr> <tr> <td>39058</td> <td>Supernatant</td> <td>Supernatant</td> <td>DN</td> <td>DN</td> <td>3</td> <td>106</td> <td>147487</td> </tr> </tbody> </table> <p>TXFR_BBIM data (made to look similar to txfr_waste_type_event data)</p> <table border="1"> <thead> <tr> <th>txfr_event_id</th> <th>Origin Phase</th> <th>Dest Phase</th> <th>Delta kgal</th> <th>Final Vol kgal</th> <th>Waste Site ID</th> <th>Std Phase</th> </tr> </thead> <tbody> <tr> <td>39057</td> <td>Salt Cake Solid</td> <td>Salt Cake Solid</td> <td>0</td> <td>305</td> <td>241-AX-101</td> <td>Salt Cake Solid</td> </tr> <tr> <td>39057</td> <td>Salt Cake Liquid</td> <td>Supernatant</td> <td>-2</td> <td>14</td> <td>241-AX-101</td> <td>Salt Cake Liquid</td> </tr> <tr> <td>39057</td> <td>Sludge</td> <td>Sludge</td> <td>0</td> <td>3</td> <td>241-AX-101</td> <td>Sludge</td> </tr> <tr> <td>39057</td> <td>Sludge</td> <td>Supernatant</td> <td>-1</td> <td>0</td> <td>241-AX-101</td> <td>Sludge</td> </tr> <tr> <td>39058</td> <td>Sludge</td> <td>Sludge</td> <td>0</td> <td>23</td> <td>241-AP-102</td> <td>Sludge</td> </tr> <tr> <td>39058</td> <td>Salt Cake Liquid</td> <td>Supernatant</td> <td>3</td> <td>106</td> <td>241-AP-102</td> <td>Supernatant</td> </tr> <tr> <td>39058</td> <td>Sludge</td> <td>Supernatant</td> <td>3</td> <td>106</td> <td>241-AP-102</td> <td>Supernatant</td> </tr> </tbody> </table>	txfr_event_id	source	sink	waste_type	recv_waste_type	volume delta kgal	wte volume kgal	txfr wte ID	39057	Salt Cake Solid	Salt Cake Solid	SC	SC	0	305	147482	39057	Salt Cake Liquid	Supernatant	IL	DN	-2	14	147483	39057	Sludge	Sludge	SL	SL	0	2	147484	39057	Sludge	Supernatant	IS	DN	-1	0	147485	39058	Sludge	Sludge	SL	SL	0	23	147486	39058	Supernatant	Supernatant	DN	DN	3	106	147487	txfr_event_id	Origin Phase	Dest Phase	Delta kgal	Final Vol kgal	Waste Site ID	Std Phase	39057	Salt Cake Solid	Salt Cake Solid	0	305	241-AX-101	Salt Cake Solid	39057	Salt Cake Liquid	Supernatant	-2	14	241-AX-101	Salt Cake Liquid	39057	Sludge	Sludge	0	3	241-AX-101	Sludge	39057	Sludge	Supernatant	-1	0	241-AX-101	Sludge	39058	Sludge	Sludge	0	23	241-AP-102	Sludge	39058	Salt Cake Liquid	Supernatant	3	106	241-AP-102	Supernatant	39058	Sludge	Supernatant	3	106	241-AP-102	Supernatant
txfr_event_id	source	sink	waste_type	recv_waste_type	volume delta kgal	wte volume kgal	txfr wte ID																																																																																																										
39057	Salt Cake Solid	Salt Cake Solid	SC	SC	0	305	147482																																																																																																										
39057	Salt Cake Liquid	Supernatant	IL	DN	-2	14	147483																																																																																																										
39057	Sludge	Sludge	SL	SL	0	2	147484																																																																																																										
39057	Sludge	Supernatant	IS	DN	-1	0	147485																																																																																																										
39058	Sludge	Sludge	SL	SL	0	23	147486																																																																																																										
39058	Supernatant	Supernatant	DN	DN	3	106	147487																																																																																																										
txfr_event_id	Origin Phase	Dest Phase	Delta kgal	Final Vol kgal	Waste Site ID	Std Phase																																																																																																											
39057	Salt Cake Solid	Salt Cake Solid	0	305	241-AX-101	Salt Cake Solid																																																																																																											
39057	Salt Cake Liquid	Supernatant	-2	14	241-AX-101	Salt Cake Liquid																																																																																																											
39057	Sludge	Sludge	0	3	241-AX-101	Sludge																																																																																																											
39057	Sludge	Supernatant	-1	0	241-AX-101	Sludge																																																																																																											
39058	Sludge	Sludge	0	23	241-AP-102	Sludge																																																																																																											
39058	Salt Cake Liquid	Supernatant	3	106	241-AP-102	Supernatant																																																																																																											
39058	Sludge	Supernatant	3	106	241-AP-102	Supernatant																																																																																																											

Change Request 201: Remove Function

See: <http://twins/crtt/CRTT.asp?ID=227&Type=CR>

Requirement #	Description
201	Change to TXFR Transfer tool 1. In a tank-to-tank transfer remove the function that automatically adds an "ADJ" entry in the receiving tank. This automatic addition causes conflicts with other "ADJ" entries. The user will be responsible for adding all "ADJ" in TXFR manually.

Change Request 15: Link the TXFR tables in TCD to the BBIM to allow for "automated" updates to volume data in the BBIM

See: <http://twins/crtt/CRTT.asp?ID=15&Type=BUG>

Requirement #	Description
15	Create an interface from BBIM to the TXFR tables so that a user can update the BBIM volumes/inventories by selecting TXFR "transfer" data (tank to tank transfers, evaporation, addition of flush water, volume adjustments). At some future time, will also need to incorporate transfers from waste streams into a tank, and transfers from a tank to a disposition site. See notes on progress.

Change Request 129: Add GEN (Waste Stream Capability) to TXFR-BBIM Interface (this is now in the TXFR application)

See: <http://twins/crtt/CRTT.asp?ID=129&Type=BUG>

Requirement #	Description
129	<p>Need to handle waste streams in the bbim. Have idea that we can store the waste streams as vectors that are NOT linked to tank_waste components. Steps involved to do this would be:</p> <ol style="list-style-type: none"> 1. Make the Vector.Waste_Site_ID field nullable 2. See if this affects the radio button form. May have to modify a stored proc and or the buildform MS Access function if it breaks. 3. Modify the Vector Insert trigger in its creation of the vector_handle. If the Waste_Site_ID field is blank, take the first word of the text found in the vector_name field and use this for the middle portion of the vector_handle, e.g. if the vector_name is 222S Waste Stream, then the vector_handle might be P/222S/01. Code that will be useful: <pre>select waste_site_id, vector_name, waste_site_id = case waste_site_id when NULL then case charindex(' ',vector_name) when 0 then vector_name else substring(vector_name,1, charindex(' ',vector_name)) end else waste_site_id end from vector where vector_name like 'mer%'</pre> 4. Enter a vector that will become a "waste stream" characterization vector (for

test purposes).

5. Implement code to support a "GEN" type record in the TXFR-BBIM. Since there is no link to the waste stream vector, will need to provide a combo box that displays all the vector handles so that the user can select one. Might consider creating a new vector type of "Waste Stream" which has a vector handle starting with "W/". That would be a way to limit the list of what is displayed.
6. Make sure to look at the rules (EventRules.xls) provided by Kristine (in my twins/bbim/xfr/ folder). Sometimes a GEN record is a concentration adjustment; other times it is an inventory adjustment. Need to figure out what makes sense.

Change Request 15: Link the TXFR tables in TCD to the BBIM to allow for "automated" updates to volume data in the BBIM

See: <http://twins/crtt/CRTT.asp?ID=15&Type=BUG>

Requirement #	Description
15	Create an interface from BBIM to the TXFR tables in TCD so that a user can update the BBIM volumes/inventories by selecting TXFR "transfer" data (tank to tank transfers, evaporation, addition of flush water, volume adjustments). At some future time, will also need to incorporate transfers from waste streams into a tank, and transfers from a tank to a disposition site. See notes on progress.

Change Request 114: TWINS-BBIM Incorporating Tank Transfer Data into BBIM

See: <http://twins/crtt/CRTT.asp?ID=113&Type=CR>

Requirement #	Description
114	The following are the original requirements for this change - See other note for revision 1. 1. The minimum input required to record a waste transfer is the amount (kL) of a waste phase from source tank + an amount (kL) of water, which is transferred to receiving tank on a given transfer date. 2. The Source Tank and waste phase must be in the BBIM. 3. The waste phase will be made up of one or more waste components in the BBIM. It is these waste components that have volume attributes. 4. Each source tank's waste component volume for the waste phase will be reduced by $(\text{transferred amount} / \text{original source tank's amount in this waste phase}) * \text{original waste component amount}$. The component(s) constituent inventories will be reduced by $(1 - \text{original volume of this component} / \text{amount transferred for this component})$. 5. If the Source Tank has a waste phase of "Total", and the waste phase that is transferred is from a non-Total phase (e.g. Supernatant), then the Total phase's volume will also be reduced by the amount. Its constituent inventories will be reduced by $(1 - \text{original volume for the "Total" waste phase} / \text{amount transferred})$. 6. A new waste component will be added to the Receiving Tank for each waste component contained in the source tank's waste phase 7. The new waste component will have the same waste phase and waste type of the matching waste component in the source tank. Its volume will be equal to $((\text{transferred}$

amount / original source tank's amount in this waste phase) * original source tank's matching waste component original amount) + (source tank's matching waste component original amount / source tank's original waste phase amount) * amount of water added in the transfer process. 8. Each new waste component's constituent inventories are equal to the "reduction" inventories defined in requirement # 4. 9. Each of the new waste component's constituent inventories will have the same basis as that of the source tank's matching waste component constituent inventories {Sample | Sample Average | HDW | Process Knowledge}. 10. Each of the new waste component's constituent inventories will be linked to the source tank's matching waste component constituent inventory vector data. This will require that a multiplier (Representative_Vectors.Multiplier) be used to adjust concentration and inventory calculations when water is added in the transfer process. The factor is equal to amount of waste transferred / (amount of waste transferred + amount of water added). 11. A history of the waste transfer will be maintained. It will include the user inputs defined in requirement #1. The Source Tank and Receiving Tank must be in an unlocked state in the BBIM for transfer data to be recorded.

Vapor/Tank Vapor Database Requirements

Vapor/Tank Vapor Database – Functional Requirements

Introduction

The purpose of this document is to specify the functional requirements for upgrades to the Tank Waste Information Network System (TWINS) Tank Vapor Database (TVD) data loader.

Background

This document supplements TWINS Change Request 363, entitled, “Tank Vapor Database (TVD) Loader Upgrade”. Requirements are stated in a very high-level manner in the change request, and so it is necessary to expand on the requirements and to state them in a format that will both drive the design and facilitate testing.

All of the ASP.NET applications have been migrated from .NET version 1.1 to .NET version 2.0 except for the current TVD Loader. The current version cannot be easily modified to run on the .NET 2.0 platform because of a 3rd party control used extensively throughout the application. Therefore a rewrite of ONLY the user interface is to be performed. Existing “back end” code (stored procedures, triggers DTS packages) have already been tested and will remain unchanged.

General Requirements Statement

The following is a general statement of the objectives of this upgrade:

Upgrade the Tank Vapor Database (TVD) data loading user interface to run on the .NET 2.0 framework to support the routine loading of future vapor headspace data in an automated, robust, and user-friendly manner. Remove any requirements in version 1.0 of this document that have since been deemed unnecessary in order to reduce the amount of development, testing and maintenance of the application.

Manual data loading of vapor data will continue to be done as needed.

Specific Requirements

Functional Requirements

Functional Requirements (FRs) specify a function that a system or system component, in this case, the TVD UI, must be able to perform. The following is an enumerated list of the FRs of the TVD UI:

1. Upload vapor headspace analytical results data in TVD SEF 2.2 format¹.
 - a. One record at a time
 - b. Multiple records at once - from Excel file
2. Load vapor headspace field sample data in TVD SEF 2.2 format.
 - a. One record at a time
3. Perform routine maintenance functions on database to enable operations without programmer assistance:
 - a. Add/Modify/Delete from data tables:
 - i. Sample Data
 - ii. Sample Reference Relation
 - iii. Reference Info

- iv. Analysis Results To Load (“raw data” table)
 - v. Analysis Results (operational database table)
- 4. Detect, identify, and prevent the loading of erroneous data into the “operational” database tables
 - a. Duplicate
 - i. Sample Data
 - ii. Analysis Results
 - b. Data Integrity
 - i. Constituent
 - ii. Units
 - iii. Qualifiers
- 5. Ability to review identified erroneous “results to be loaded” data and make corrections

Vapor/Tank Vapor Database – Change Request Tracking Tool Requirements

Change Request 176.2: Include results from field, trip and ambient air blanks and radiological data in the Twins Tank Vapor Database

See: <http://twins/crtt/CRTT.asp?ID=175&Type=CR>

Requirement #	Description
176.2	A separate area already exists for the radiological data. Reconfigure the vapor database queries for “analysis results” to allow the blank data to be displayed. To do this, add an “Aggregation Level” column and a “QA Sample Type” column to the “analysis results” table. For example: Aggregation Level QA Sample Type QA Sample Trip Blank Headspace sample None Ventilation sample None QA Sample Ambient Upwind. The lab will be sending the data using existing protocols. A change order to modify the auto TCR vapor standard report shall be filed separately.

Change Request 14: TCD/TVD Search Capability

See: <http://twins/crtt/CRTT.asp?ID=13&Type=CR>

Requirement #	Description
66.1	Add ability to search/query both TCD and TVD by use of CAS number.

Change Request 610: Add capabilities to the VAPOR2 Loader ASP.NET application

See: <http://twins/crtt/CRTT.asp?ID=610&Type=BUG>

Requirement #	Description
610	Based upon a Live Meeting review this morning with Leslie Diaz and Harold Baker, the following capabilities need to be added to the Vapor2 application: <ul style="list-style-type: none"> • Add the ability to Add a new Reference. Currently you can only update existing references • Add the ability to Add a new Sample Device. Currently you can only select from pre-existing sample devices in the <i>Sample Data To Load</i> form.

Change Request 149: Add load date and time to Vapor database

See: <http://twins/crtt/CRTT.asp?ID=148&Type=CR>

Requirement #	Description
149	Modify the Tank Vapor Database to add both date and time of loading vapor data.

Change Request 127: Vapor data unitsSee: <http://twins/crtt/CRTT.asp?ID=126&Type=CR>

Requirement #	Description
127	Allow vapor data to be reported in units other than mg/m3. For tank AN-107 (and probably many other tanks), the data, which was originally reported in mole percent (in units of ppmv), has been converted to concentration (in units of mg/m3). The conversion requires the vapor database people to enter the molecular weight of the compound. Unfortunately, there are two measured categories of analytes: other hydrocarbons and other nitrogen oxides, that don't have molecular weights. For these categories of analytes, the auto_TCR vapor data standard report is showing a concentration of <_____ (with no value shown). The solution is to report these two categories in the units they were measured in (ppmv).

Change Request 283: Add TAP table to vapor_stage databaseSee: <http://twins/crtt/CRTT.asp?ID=283&Type=BUG>

Requirement #	Description
283	As part of CR236 we received an Excel sheet with approximately 600 chemical names and CAS numbers to include in the Toxic Pollutants category for a TWINS vapor query. (Currently only 138 of these chemicals exist and have results in the Vapor database.) We need to keep a table of the TAP chemicals so that when new chemicals are added to Vapor results they can be included in the TAP query where appropriate.

Change Request 297: Labcore Headspace Vapor SEF and Data LoaderSee: <http://twins/crtt/CRTT.asp?ID=398&Type=CR>

Requirement #	Description
297	<p>Create a Headspace vapor SEF and data loader for Labcore:</p> <ul style="list-style-type: none"> • Due Date 11/30/04. Develop and issue Labcore Headspace Vapor SEF document. • Due Date 12/31/04. Develop/Program TWINS interface with Labcore and data loading tools. • Due Date 2/28/05. Load Previous Headspace Data. (Headspace samples taken since 2001). <p>JUSTIFICATION: Change needed to make Headspace vapor data available on TWINS.</p>

Details of CR297 are found in the [Format Specification](#).**Change Request 326: Update IH field sample and analysis results displayed on TWINS**See: <http://twins/crtt/CRTT.asp?ID=452&Type=CR>

Requirement #	Description
326	Update the TWINS Vapor/Industrial Hygiene/IH Sampling fields to be displayed on TWINS. Attached is the updated IH Field Sample Record Layout and IH Analysis Results Table Layout from LMIT. If you have any questions, contact Joyce Caldwell at LMIT or Kristine Bowen at CH2M Hill.

DRAFT

STANDARD ELECTRONIC FORMAT SPECIFICATION FOR TANK VAPOR DATA LOADER: VERSION 2.2

Stephen F. Bobrowski
Sylvia J. Harris
Joy Y. Smith
Ingrid H. Wright

Pacific Northwest National Laboratory

November 2, 2004

Prepared for
CH2M Hill Hanford Group

Standard Electronic Format Specifications for Tank Vapor Data Loader

Modification History

Version 1.0	December 31, 1998	Original document
Version 2.0	April 4, 2001	Automated loading format
Version 2.1	August 28, 2001	Include Radiological Data and QA Samples
Version 2.2	October 29, 2004	Update and publish

Table of Contents

1.0 INTRODUCTION.....	9
1.1 FILE NAMING CONVENTIONS	9
1.2 TRANSMISSION OF FILES AND DATA PACKAGES	9
1.3 DATA TYPE DESCRIPTIONS	9
2.0 ANALYSIS RESULTS TABLE	9
2.1 ANALYSIS RESULTS FIELD DESCRIPTIONS	10
2.2 ANALYSIS RESULTS TABLE LAYOUT	11
3.0 SAMPLE DATA INFORMATION.....	13
4.0 PUBLICATION INFORMATION.....	14
5.0 SAMPLE TO PUBLICATION CROSS-REFERENCE INFORMATION.....	15
APPENDIX A : SELECTED FIELD VALUES.....	16
A.1 SAMPLING DEVICES IN VAPOR DATABASE	16
A.2 SAMPLING METHODS IN VAPOR DATABASE.....	17
A.3 ANALYSIS METHODS IN VAPOR DATABASE.....	17
A.4 PROJECT TYPES IN VAPOR DATABASE	18
A.5 DATA QUALIFIER CODES IN VAPOR DATABASE.....	18
A.6 UNITS IN VAPOR DATABASE	19
A.7 LABORATORIES IN VAPOR DATABASE	19
A.8 RESULT TYPES IN VAPOR DATABASE.....	20
APPENDIX B : VAPOR DATA CHECKLIST.....	21

1.0 Introduction

The purpose of this document is to provide the format and instructions for the electronic data table for entry into the Tank Vapor Database. This table may be an Access Table, an Excel Spreadsheet or a delimited file. All table/file types must have one type of record per table or spreadsheet and one record per row. The records to be delivered in electronic format are for Analysis Results

The type of record must be identified in the table or spreadsheet name.

Also required for vapor data entry is document publication information and sample data.

1.1 File Naming Conventions

Files are required to have the tank farm and the tank number at the beginning of the name. An example would be TX117*.xls, with the wildcard replaced by any additional text in situations where it is necessary to distinguish these data from another TX117 data set.

1.2 Transmission of Files and Data Packages

The files shall be sent by e-mail to the designated contact for the Tank Vapor Database at PNNL. A hard copy of the data package shall be sent to the same contact person at PNNL. TVD staff can then extract the document publication information for the database.

1.3 Data Type Descriptions

Character fields	All character fields will contain standard ASCII characters.
Date fields	All dates will be represented by MM/DD/YY
Time fields	All times will be represented by HH:MM:SS
Date/Time fields	All date/time fields may have only the date MM/DD/YY or may have date and time MM/DD/YY HH:MM
Number fields	A number may be entered as an integer, a real, or using scientific notation. All numbers are to have been rounded by the analytical laboratory.

2.0 Analysis Results Table

Each analysis result has one record in the Analysis Result Table and must have a related record, with the same Field Sample ID, in the Sample Data Table. Note that CAS numbers are required

Standard Electronic Format Specifications for Tank Vapor Data Loader

except for unknown tentatively identified chemicals for which an appropriate code will be generated by the TVD staff.

Section 2.1 is an alphabetical list of the fields in the Analysis Results Table with descriptions of the contents. Section 2.2 is a table that provides the required table layout and includes field length and data type.

2.1 Analysis Results Field Descriptions

<u>Field Name</u>	<u>Description</u>
Analysis Date	Date of sample analysis.
Analysis Method	Primary instrumentation used in the analysis. Must be in TVD analysis method table. See Appendix A.3
Analysis Method Description	Description of the Analysis Method.
Basis of Holding Time	The document that specifies the allowed holding time for samples.
Chemical ID	Chemical Abstract Services (CAS) number or other code that uniquely identifies a chemical.
Chemical Name	Name of a chemical (a chemical is defined as an element or compound).
Comments	Any comments about the analysis result
Data Qualifier Code	Single-letter flags that indicate noteworthy aspects of the result. Must be in the table of vapor qualifier codes. See Appendix A.5.
Field Sample ID	Sample identification code used for the sample when collected from the field. Must be in accompanying Sample Table.
Holding Time	The allowed holding time in days between the time samples were collected and sample analysis.
Laboratory Abbreviation	Abbreviation or acronym for the analytical laboratory.
Laboratory Sample ID	Sample identification code used within the laboratory

Standard Electronic Format Specifications for Tank Vapor Data Loader

Molecular Weight	Sum of the naturally occurring isotopic atomic weights of all the atoms in a molecule (units are gram/mole).
Reference Temperature	Temperature for which mass per unit volume concentrations results are given Units are degrees C.
Reported Value	Estimated or quantified concentration of a chemical as reported by the analytical Lab. Mass per unit volume reported values are at the Reference Temperature and 1.01325 bar pressure.
Reporting Limit	A lower limit below which the concentration of a constituent is not reported.
Result Type	Description of the type of results the analysis produced. Examples include: Primary_Result and Duplicate_Result.
Units	Units that the Reported Value is given in

2.2 Analysis Results Table Layout

Field Name	Type and Max Size	Required	Contents	Example	TVD field
Field Sample ID	Chr(45)	Yes	Code assigned to field sample – must be in Sample Table	V7055-A06-105	Sample_id
Laboratory Sample ID	Chr(45)	Yes	Code assigned to lab sample	W04001598	Lab_sample_id
Chemical Name	Chr(240)	Yes.	Name of a chemical	Methane	chemical_name
Chemical ID	Chr(15)	Yes, but see exception above (para 2).	Chemical Abstract Services Number (CAS)	74-45-6	chemical_id
Molecular Weight	Number	No	Molecular weight of analyte	137.4	molecular_weight
Reference Temperature	Number	Yes	Temperature that mass per unit vol. is given.	25	ref_temperature
Analysis Date	date	Yes	Date of sample analysis	1/28/01	analysis_date

Standard Electronic Format Specifications for Tank Vapor Data Loader

Field Name	Type and Max Size	Required	Contents	Example	TVD field
Result Type	Chr(20)	Yes, must be in TVD. See Table A.8	Type of result	Primary_Result	Result_type
Reported Value	Number	No, if null then reporting limit required.	Concentration of a chemical	34.8	reported_value
Units	Chr(10)	Yes, must be in TVD. See Table A.6	Units result if given in	mg/m3 ppbv	units
Reporting Limit	Chr(20)	No, if null reported value must exist	Limit below which not reported	<0.01	reporting_limit
Data Qualifier	Chr(10)	No. Must exist in TVD See Table A.5	Single letter flags	JNHS	data_qual_code
Holding Time	Number	No	Allowed holding time in days	45	holding_time
Basis of Holding Time	Chr(100)	No	Document that specifies the allowed holding time	Administratively required by doc. num. WHC-SD-WM-QAPP-013 Rev. 2	basis_hold_time
Analysis Method	Chr(50)	Yes, must exist in TVD. See Table A.3	Method used in analysis.	GC/MS	Analysis_method
Analysis Method Description	Chr(255)	Yes	Description	Thermal Desorption	analysis_method_name
Laboratory Abbreviation	Chr(6)	Yes, must exist in TVD. See Table A.7	Acronym of the analytical laboratory.	PNNL	lab_abbrev
Comments	Chr(255)	No	Additional comments about analysis result	Tank temp not available	comments_results

3.0 Sample Data Information

The following information must be provided. It need not be in standard electronic format.

<u>Sample Data</u>	<u>Description</u>
Aggregation Level	Label applied to the whole, i.e. "aggregate", sample. Current Aggregation Levels include Headspace Sample, Ventilation Sample, and QA Sample.
Barometric Pressure	Barometric pressure measured at the start of the sample collection.
Comments	Any concerns or unusual information about collecting the sample
Date Sampled	Date the sample was taken.
Field Sample ID	Code assigned to a sample by the collecting organization (a sample is defined as a discrete quantity of material collected for analysis.)
Project Type	The project type. Must be in the Vapor Database. See Table A.4.
QA Sample Type	Type of quality assurance sample, if any. Current values include Trip Blank, Ambient Upwind, and None.
Sampling Device	Description of the type of container used to collect a sample . Must be in Vapor Database. See Table A.1.
Sampling Method	Description of the type of general method used to collect a sample. Must be in Vapor Database. See Table A.2.
Tank ID	Code that identifies the tank for tank farm system.
Tank Riser /Sampling Access	Riser number or identifier of other tank access from which the sample was taken.
Tank Temperature	Temperature inside the waste tank when the sample was taken. (units are C).
Time Sampled	The time the sample was taken.

<u>Sample Data Work Package Number</u>	<u>Description</u> Number of the work package used to obtain the sample
---	---

4.0 Publication Information

The following information must be provided. It need not be in standard electronic format. If the analysis has not been published, Document Number should read “Unpublished”.

<u>Publication Information</u>	<u>Description</u>
Authors	List of all the publication's authors.
City and State of the Publisher	The city and the state of the laboratory that published the document.
Document Number	Code assigned by government agencies or government contractors to documents and correspondence they publish; the code is used for tracking purposes. This code includes revision numbers. Examples: WHC-SD-WM-RPT-108 Rev 0; PNNL-11163
Publisher	The laboratory that published the document.
Reference Title	Title of the publication as it appears on the title page.
Year Published	The year that the publication was published

5.0 Sample to Publication Cross-Reference Information

The following information must be provided for any published data. It need not be in standard electronic format.

<u>Cross-Reference Information</u>	<u>Description</u>
Document Number	Code assigned by government agencies or government contractors to documents and correspondence they publish; the code is used for tracking purposes. This code includes revision numbers. Examples: WHC-SD-WM-RPT-108 Rev 0; PNNL-11163
Field Sample ID	Code assigned to a sample by the collecting organization (a sample is defined as a discrete quantity of material collected for analysis.)

Appendix A : Selected Field Values

A.1 Sampling Devices in Vapor Database

Sampling Device	Sampling Device Description
Acetate trap	Acetate trap
DST	Double sorbent trap is a tube contained 2 beds of sorbent media for collection of lower volatility organic compounds. Used to allow spiking with tributyl phosphate dissolved in a volatile solvent.
flow-through vessel	Any vessel fitted with inlet and outlet valves to allow continuous purge with sample air.
glass bubbler	Vessel containing an absorbing solution through which sample air was bubbled. Used with caustic solution to trap HCN.
H2O trap	Silica gel sorbent media trap for collection of water vapor.
HCN/H2O traps	Hydrogen cyanide and water vapor sorbent traps connected in series.
NH3 trap	Acidified sorbent media trap for collection of ammonia vapor.
NH3/H2O traps	Ammonia sorbent trap and water sorbent traps connected in series.
NH3/H2O/H2O traps	Ammonia sorbent trap and 2 water vapor traps connected in series.
NH3/NO2/NO traps	Ammonia, nitrogen dioxide, nitric oxide traps connected in series.
NH3/NO2/NO/H2O traps	Ammonia, nitrogen dioxide, nitric oxide, and water vapor sorbent traps connected in series.
NO2/NO traps	Nitrogen dioxide and nitric oxide sorbent traps connected in series.
NO2/NO/H2O traps	Nitrogen dioxide, nitric oxide, and water vapor sorbent traps connected in series.
OVS	OSHA versatile sampler is a glass tube containing a filter to trap particulates followed by sorbent media for organic vapors.
PUF	Polyurethane sorbent trap
Sorbent Tube Train	Sorbent Tube Train
SOx trap	Sulfur oxides sorbent trap.
SUMMA	Vessel with electropolished and passivated interior surfaces for collection of gases and organic vapors. SUMMA is a trademark of the Moletrics Corp.

Standard Electronic Format Specifications for Tank Vapor Data Loader

Sampling Device	Sampling Device Description
TST	Triple sorbent trap is a tube containing 3 beds of sorbent media, each successive bed having greater retentivity for organic vapors.

A.2 Sampling Methods in Vapor Database

Sampling Method	Sampling Method Description
direct	Sample collected without intervening manifold, or with minimum of transfer tubing.
ISS	In situ sampling methods collected SUMMA samples via unheated tubing lowered into headspace. Sorbent traps were lowered into headspace and sample volumes were determined with rotameters. Not used after 1994.
ISVS	In situ vapor sampling methods collect SUMMA samples via unheated tubing lowered into headspace. Sorbent traps are lowered into tank headspace and sample volumes are determined with mass flow meters. Similar to ISS but with better sorbent trap sample volume measurement.
ISVS w filter	ISVS method used with glass fiber particulate filter in-line between headspace and each sample device.
ISVS wo filter	ISVS method used without glass fiber particulate filter in-line between headspace and each sample device.
NEVS	Vapor Sampling Using the Non-Electrical Vapor Sampling System-LO-080-400
Riser Port	Air samples collected via riser without tubing to reach the central region of the headspace.
VSS	Vapor sampling system method used heated sampling probe, transfer lines, and manifold to deliver sample air to collection devices. Sorbent trap sample volumes determined by mass flow controllers.

A.3 Analysis Methods in Vapor Database

Analysis Method	Analysis Method Description
ADIC	Amperometric detection ion chromatography
CGM	Combustible gas meter
Colorimetric tube	Colorimetric tube
GC/ECD	Gas chromatography with electron capture detector
GC/FCD	Gas chromatography with flame conductivity detector

Standard Electronic Format Specifications for Tank Vapor Data Loader

Analysis Method	Analysis Method Description
GC/FID	Gas chromatography with flame ionization detector
GC/HGO/FID	Gas chromatograph - mercury oxide reduction - flame ionization detector
GC/MS	Gas chromatography with mass spectrometer detector
GC/RGD	Gas chromatography with reduction gas detector
GC/TCD	Gas chromatography with thermal conductivity detector
Gravimetric analysis	Gravimetric analysis
IC	Ion chromatography
MS	Mass spectrometer detector
OVM	Organic vapor monitor
SCIC	Suppressed-conductivity ion chromatography
SIE	Selective ion electrode
TD/GC/MS	Thermal desorption and gas chromatography with mass spectrometer detector
TO-12	EPA compendium task order 12
TO-14	EPA compendium task order 14

A.4 Project Types in Vapor Database

Project Type
Headspace Characterization Data
IMUST Data
Test Data

A.5 Data Qualifier Codes in Vapor Database

Code	Qualifier Description	Qualifier Comment
B	compound found in associated laboratory blank as well as sample	based on EPA contract laboratory protocol
D	target analyte reported is the result of a secondary dilution	based on EPA contract laboratory protocol
E	reported concentration was above the instrumental calibration range	based on EPA contract laboratory protocol
F	target analyte reported in sample was also found in field blank above action limit	chosen by Numatec Hanford Corporation - Special Analytical Support
H	analysis was performed after allowed analytical holding time had elapsed or analysis date is not available	chosen by the tank vapor database (TVD) staff unique to TVD

Standard Electronic Format Specifications for Tank Vapor Data Loader

Code	Qualifier Description	Qualifier Comment
J	reported concentration was estimated	based on EPA contract laboratory protocol
M	target analyte was absent (not detected) from sample	chosen by Numatec Hanford Corporation - Special Analytical Support
N	compound was tentatively identified	based on EPA contract laboratory protocol
Q	target analyte was detected, but at concentrations less than Vapor Program Required Quantitation Limits	chosen by Numatec Hanford Corporation - Special Analytical Support
S	result suspect - see comment field	chosen by the tank vapor database (TVD) staff unique to TVD
T	target analyte reported in sample was also found in trip blank above action limit	chosen by Numatec Hanford Corporation - Special Analytical Support
U	compound was analyzed for but not detected above reporting value	based on EPA contract laboratory protocol
X	analytical laboratory did not have approved quality assurance documentation, or that a significant quality assurance deficiency was associated with reported result	chosen by the tank vapor database (TVD) staff unique to TVD

A.6 Units in Vapor Database

Units	Description
mg/L	Milligrams per Liter
mg/m ³	Milligrams per Cubic Meter
mole%	Percent Mole
ppbC	Parts per Billion
ppbv	Parts per Billion Volume
ppmv	Parts per Million Volume
ug/m ³	Micrograms per Cubic Meter

A.7 Laboratories in Vapor Database

Lab Abbreviation	Name
NHC	Numatec Hanford Corporation

Standard Electronic Format Specifications for Tank Vapor Data Loader

Lab Abbreviation	Name
OGI	Oregon Graduate Institute
ORNL	Oak Ridge National Laboratory
PNNL	Pacific Northwest National Laboratory
WMT	Waste Management Hanford
WSCF	Waste Characterization and Storage Facility

A.8 Result Types in Vapor Database

Result Type	Description
Duplicate_Result	Duplicate analysis result value
Primary_Result	Primary analysis result value

Appendix B : Vapor Data Checklist

Tank: _____ **Date Sampled:** _____

Items needed for loading:

- _____ Work Package Number _____
- _____ Copy of the IH Sampling Information
 - _____ IH Survey Number _____
- _____ Copy of the Chain of Custody Forms
- _____ Copy of the analytical data from WSCF
 - _____ SUMMA Target Analytes & TICs
 - _____ SUMMA Permanent Gases
 - _____ Ammonia
 - _____ Other
- _____ Copy of the analytical data from other labs
 - _____ Total Mercury (TM) by GeoSciences
 - _____ Dimethyl mercury (DM) by GeoSciences
 - _____ Formaldehyde (CH₂) by DataChem
 - _____ SUMMA by SRS
 - _____ Other
- _____ Copy of the Report/Data Package if the data has been published
 - _____ Document Number _____

The following information is needed if the data is not being provided electronically: This information should only be needed once unless there are changes.

- _____ Results need to be reported in concentration (ppbv, ppmv or other appropriate units) not just mg per trap or we need to be provided with the conversion equation and the necessary information to make the conversion. The conversion to mg/m³ is made during loading.
- _____ Holding times for each constituent and the document number that supports this _____
- _____ Reference Temperature
- _____ Reporting Limit for each constituent
- _____ Information needed about other labs: Name, abbreviation, contact number, location, method titles and method number used for analysis.

COMMENTS:

TWINS SQL Server Integration Services Design Documentation

TWINS SQL Server Integration Services Design Documentation

September 2010

Tom Olund

Overview

The SQL Server Integration Services (SSIS) is used in TWINS to provide data processing in the background that would otherwise impact performance if executed at the time the user modifies or requests data. This processing is normally scheduled during the off hours.

Configuration

The TWINS packages are located in the TWINS SSIS Visual Studio 2008 project.

The package configuration XML files are located in the D:/SSISConfig directory. These include:

DBConn.dtsConfig	Contains database connectivity information used by all packages when loaded for execution. Modification of the values in the file will enable the packages to connect to different database servers (TWINS and TWINSDEVDB).
SMTP.dtsConfig	Contains the information used by all packages when loaded for execution. Modification of the values in this file will enable the packages to connect to different mail servers and to send email messages regarding package execution to different recipients.

Naming Conventions

The package task prefixes have the following naming conventions:

- EPT : Execute Process Task
- EST : Execute SQL Task
- SMT : Send Mail Task

Event Handlers

Each package has an event handler for the On Error event that will send mail to the MailTo recipients (stored in the TWINSConfig.dtsconfig file) stating the nature and source of the error.

Packages

Populate TCD.dtsx

This package is the top level package that is executed nightly. It process and sends control to the lower level packages that exist in order to compartmentalize the processing. The individual tasks in the package are shown in Figure 1.

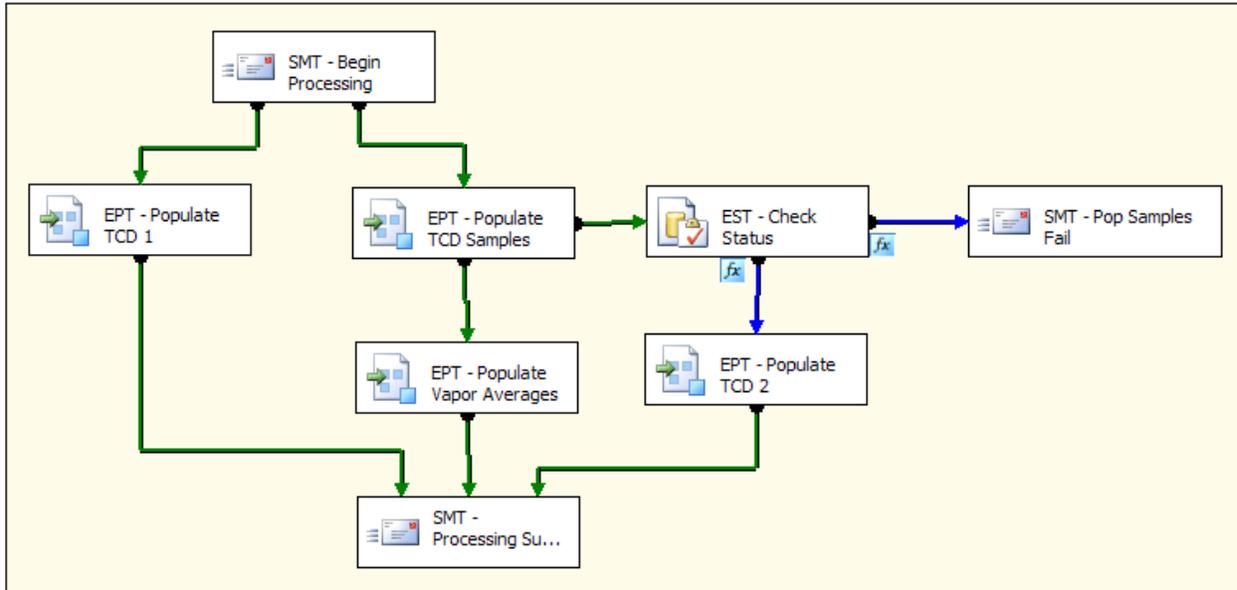


Figure 1

Tasks

SMT – Begin Processing

Send email notification to the “MailTo” recipients stored in the TWINSConfig.dtsconfig file that processing has begun.

EPT – Populate TCD 1

Execute the [Populate TCD 1] Package.

EPT – Populate TCD Samples

Execute the [Populate TCD Samples] Package.

EPT – Populate Vapor Averages

Execute the [Populate Vapor Averages] Package.

SMT – Processing Successfully completed

Send email notification to the “MailTo” recipients stored in the TWINSConfig.dtsconfig file that the Populate TCD SSIS package has been successfully completed.

EST – Check Status

Checks the dts_packages table in the TWINS_MGMT database to see if the [EPT – Populate TCD Samples] package completed successfully. If the package did succeed, then the EPT – Populate TCD 2 task is executed. Otherwise, the SMT – Pop Samples Fail task is executed.

EPT – Populate TCD 2

Execute the [Populate TCD 2] Package.

SMT – Pop Samples Fail

Send email notification to the “MailTo” recipients stored in the TWINSConfig.dtsconfig file that the Task "EST - Check Status" in the SSIS Package "Populate TCD" found that the "Populate TCD Samples" package did not complete.

SMT – On Error Notify

Send email notification to the “MailTo” recipients stored in the TWINSConfig.dtsconfig file that an error has occurred during processing of the package.

Populate TCD 1.dtsx

This package is executed by the Populate TCD.dtsx package.

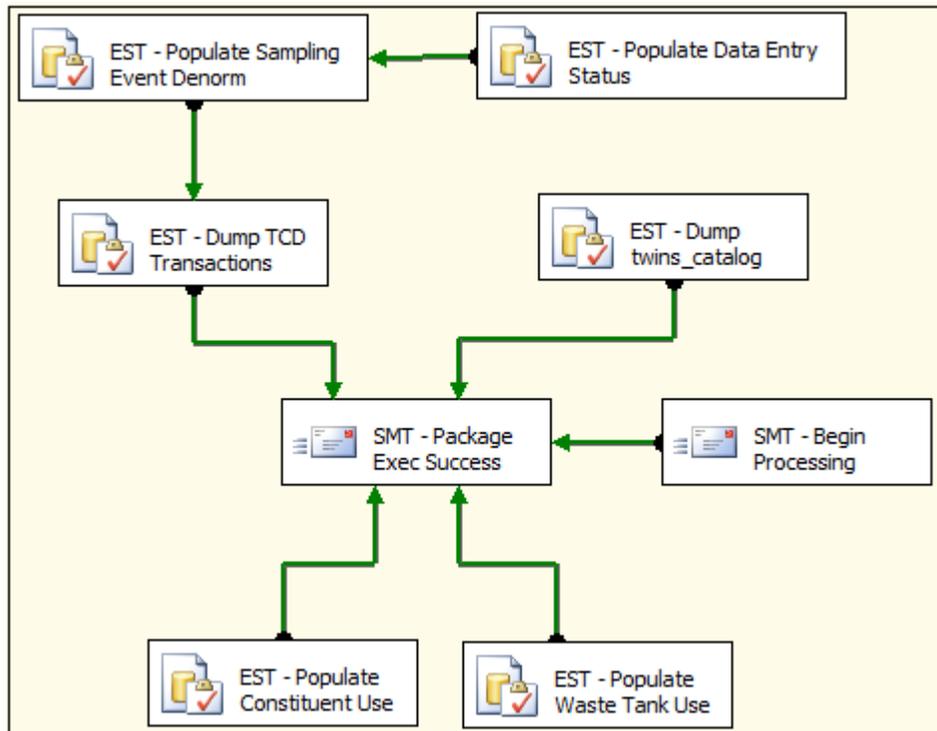


Figure 2

Tasks

SMT – Begin Processing

Send email notification to the “MailTo” recipients stored in the TWINSConfig.dtsconfig file that processing has begun.

EST – Populate Data Entry Status

Truncates and then populates the TCD t_data_loaded_status table with the tanks that have recently had new sampling events, published to the BB_Published database, changes to HTCE dates or changes to the TCR date.

EST – Dump twins_catalog

Dumps the transaction log for the twins_catalog database.

EST – Populate Sampling Event Denorm

Executes the tcd.dbo.p_sampling_event_denorm stored procedure. This procedure first deletes all records from the tcd.dbo.sampling_event_denorm table, and then reloads the data from the “normalized” tables.

EST – Dump TCD Transactions

Dumps the TCD database transaction log.

EST – Populate Constituent Use

Executes the tcd.dbo.p_set_constituent_use stored procedure. Updates the various dbo.Constituent flag fields (analysis_use, inventory_use, historical_use) to Y when the constituent is found to have records in various tables.

EST – Populate Waste Tank Use

Executes the tcd.dbo.p_set_waste_tank_use stored procedure. This stored procedure sets the used_in_sampling_event flag for those tanks that have had samples taken from them.

SMT – Package Exec Success

Send email notification to the “MailTo” recipients stored in the TWINSConfig.dtsconfig file that processing has successfully completed.

SMT – On Error Notify

Send email notification to the “MailTo” recipients stored in the TWINSConfig.dtsconfig file that an error has occurred during processing of the package.

Populate TCD 2.dtsx

This package is executed by the Populate TCD.dtsx package.

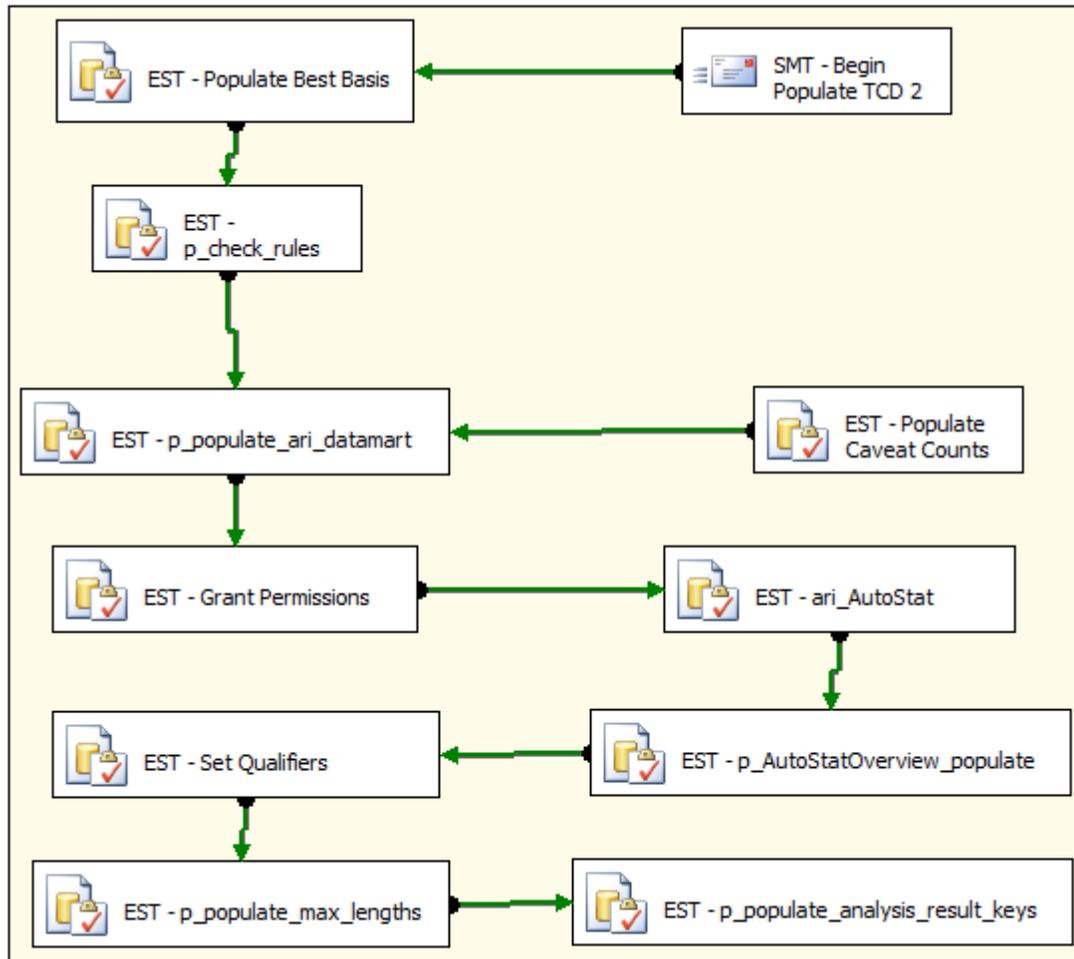


Figure 3

Tasks

SMT - Begin Populate TCD 2

Send email notification to the “MailTo” recipients stored in the TWINSConfig.dtsconfig file that processing has begun.

EST - Populate Best Basis

Executes the tcd.dbo.p_pop_best_basis stored procedure. Inserts data into the tcd.dbo.Best_Basis_Summary table. The TWINS Best Basis Summary Query (tcd.dbo.p_Best_Basis_Summary) retrieves data from this table.

EST – p_check_rules

Executes the tcd.dbo.p_check_rules stored procedure. This procedure sets the status of recently loaded TCD records for the Resolve application.

EST – Populate Caveat Counts

Executes the tcd.dbo.p_caveat_counts stored procedure. This may be an obsolete stored procedure.

EST – p_populate_ari_datamart

Executes the p_populate_ari_datamart stored procedure. Drops the tcd_read.dbo.ari_datamart table, creates it and then inserts data into it from the TCD database.

EST – Grant Permissions

Runs a script that grants “select” permissions on the following tables:

- tcd_read.dbo.ari_analysis_method
- tcd_read.dbo.ari_constituent
- tcd_read.dbo.ari_datamart
- tcd_read.dbo.ari_sample
- tcd_read.dbo.ari_sample_analysis
- tcd_read.dbo.ari_sample_description

EST – ari_AutoStat

Execute SQL to populate the TCD_READ ari_AutoStat table.

EST – p_AutoStatOverview_populate

Execute the tcd.dbo.p_AutoStatOverview_populate stored procedure for each AutoStatOverview record.

EST – Set Qualifiers

Execute SQL to update the tcd_read.dbo.ari_datamart Qualifiers.

EST – p_populate_max_lengths

Executes the tcd.dbo.p_populate_max_lengths stored procedure. This stored procedure records the field name and length of the largest value in the field for the analysis_result_mod, constituent_mod and log_changes tables.

EST – p_populate_analysis_result_keys

Executes the tcd.dbo.p_populate_analysis_result_keys stored procedure. Populates the tcd.dbo.analysis_result_keys table.

SMT – On Error Notify

Send email notification to the “MailTo” recipients stored in the TWINSConfig.dtsconfig file that an error has occurred during processing of the package.

Populate TCD Samples.dtsx

This package is executed by the Populate TCD.dtsx package.

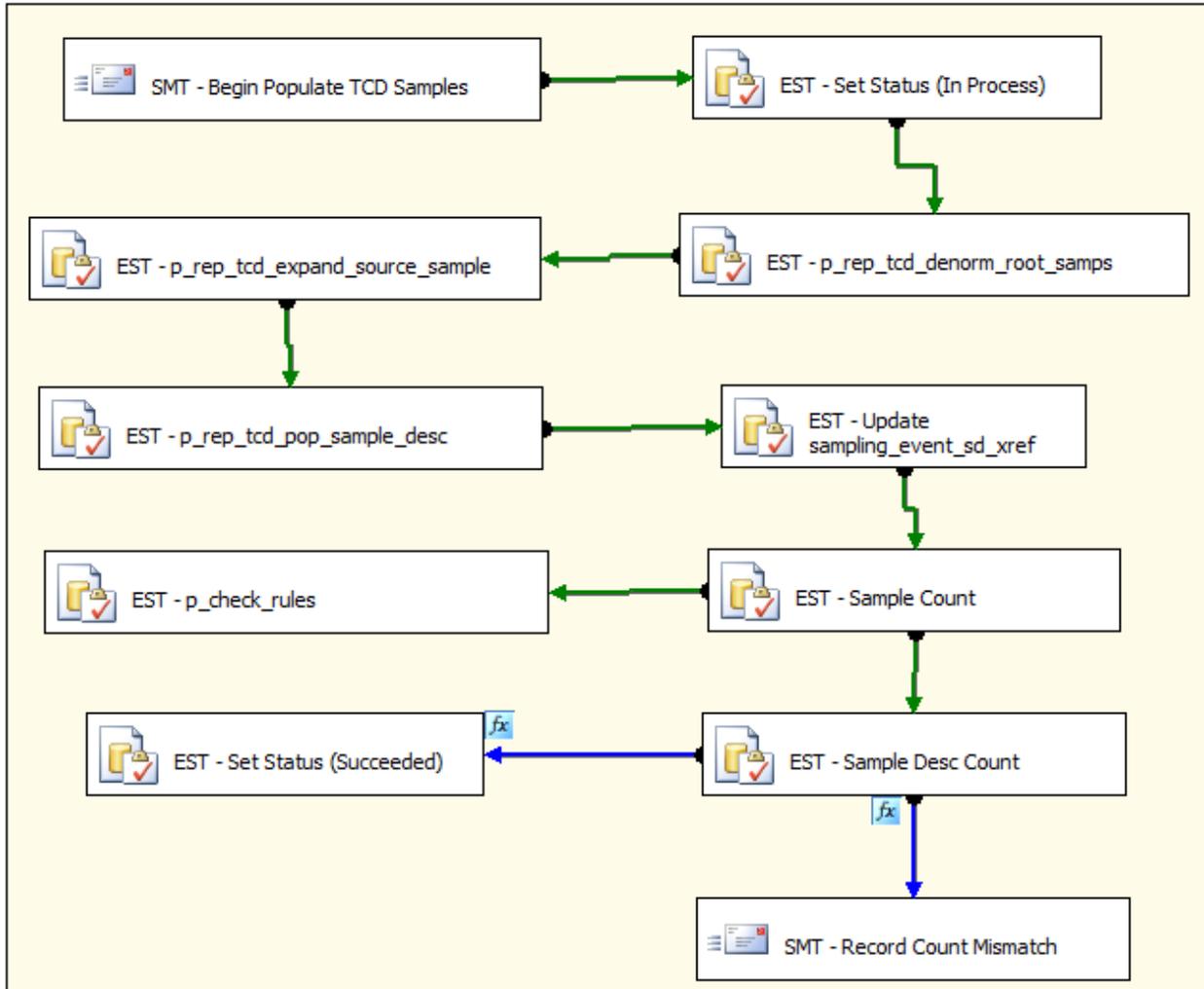


Figure 4

Tasks

SMT – Begin Processing

Send email notification to the “MailTo” recipients stored in the TWINSConfig.dtsconfig file that processing has begun.

EST – Set Status (In Process)

Update the DTS_Packages status in the TWINS_MGMNT database to "In Process."

EST – p_rep_tcd_denorm_root_samps

Execute the tcd.dbo.p_rep_tcd_denorm_root_samps stored procedure. This procedure populates the tcd.dbo.source_sample table.

EST – p_rep_tcd_expand_source_sample

Executes the tcd.dbo.p_rep_tcd_expand_source_sample stored procedure. This procedure adds related sample data (children) to the tcd.dbo.source_sample table.

EST – p_rep_tcd_pop_sample_desc

Executes the tcd.dbo.p_rep_tcd_pop_sample_desc stored procedure. This procedure populates the sample_description with one record for each distinct sample number in the source_sample table.

EST – Update sampling_event_sd_xref

Executes the tcd.dbo.p_pop_sampling_event_sd_xref stored procedure. This procedure updates the tcd.dbo.sampling_event_sd_xref table. This table may no longer be referenced as the new means and variances report does not utilize this table.

EST – Sample Count

Get the number of Sample records

EST – Sample Desc Count

Return the number of Sample Description Records into a local variable.

EST – Set Status (Succeeded)

Update the DTS_Packages status in the TWINS_MGMNT database to "Succeeded" if the Sample record count matches the Sample Description record count.

SMT – Record Count Mismatch

Send email notification to the "MailTo" recipients stored in the TWINSConfig.dtsconfig file that the Sample record count does not match the Sample Description record count.

SMT – On Error Notify

Send email notification to the "MailTo" recipients stored in the TWINSConfig.dtsconfig file that an error has occurred during processing of the package.

Populate TCD Vapor Averages.dtsx

This package is executed by the Populate TCD.dtsx package.

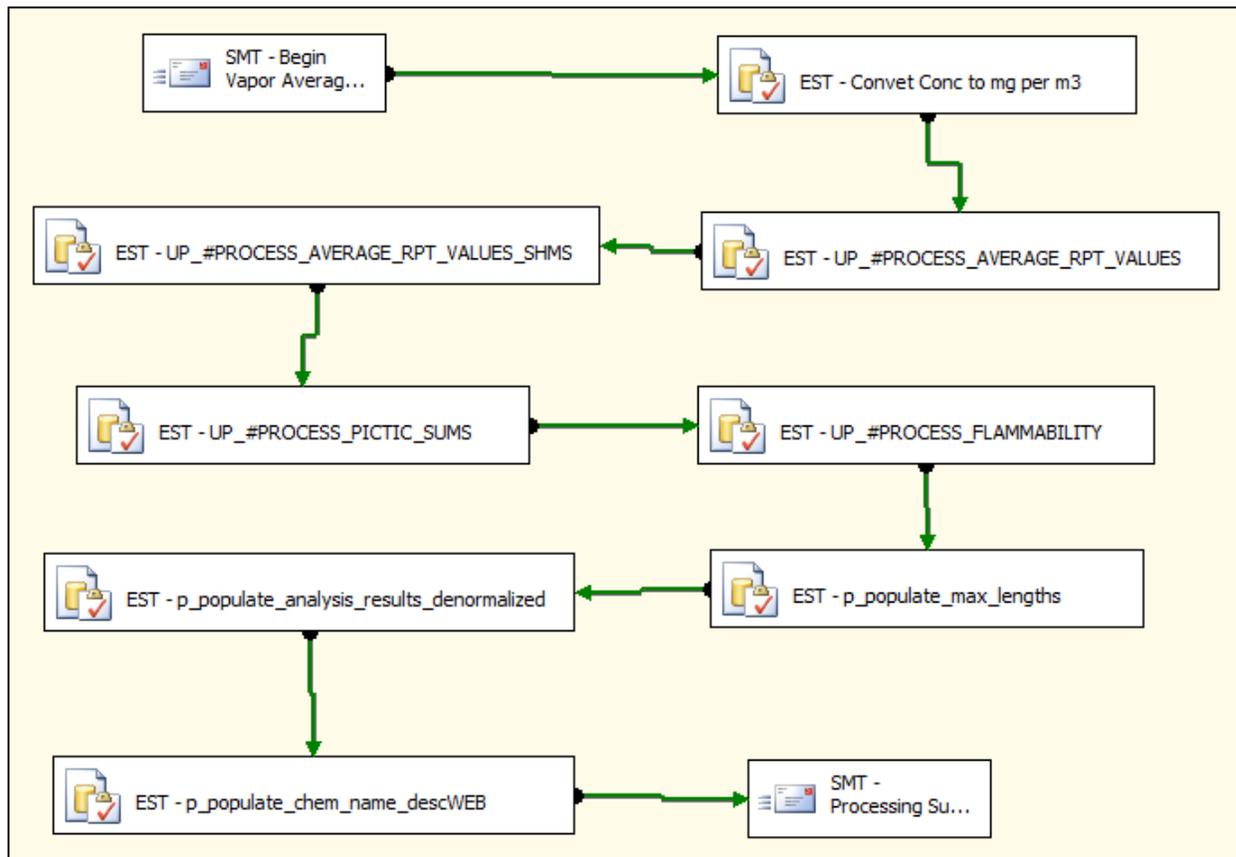


Figure 5

Tasks

SMT – Begin Vapor Average

Send email notification to the “MailTo” recipients stored in the TWINSConfig.dtsconfig file that processing has begun.

EST – Convert Conc to mg per m3

Execute the UP_#CONVERT_CONC_TO_MGM3 stored procedure.

EST – UP_#PROCESS_AVERAGE_RPT_VALUES

Execute the UP_#PROCESS_AVERAGE_RPT_VALUES stored procedure.

EST – UP_#PROCESS_AVERAGE_RPT_VALUES_SHMS

Execute the UP_#PROCESS_AVERAGE_RPT_VALUES_SHMS stored procedure.

EST – UP_#PROCESS_PICTIC_SUMS

Execute the UP_#PROCESS_PICTIC_SUMS stored procedure.

EST – UP_#PROCESS_FLAMMABILITY

Execute the UP_#PROCESS_FLAMMABILITY stored procedure.

EST – p_populate_max_lengths

Execute the p_populate_max_lengths stored procedure

EST – p_populate_analysis_results_denormalized

Execute the p_populate_analysis_results_denormalized stored procedure.

EST – p_populate_chem_name_descWEB

Execute the p_populate_chem_name_descWEB stored procedure.

SMT – Processing Successfully Completed

Send email notification to the “MailTo” recipients stored in the TWINSConfig.dtsconfig file stating that the processing completed successfully.

SMT – On Error Notify

Send email notification to the “MailTo” recipients stored in the TWINSConfig.dtsconfig file that an error has occurred during processing of the package.

Populate AutoStat Update.dtsx

This package performs the background processing needed by the AutoStat ASP.NET application. It is run every minute.

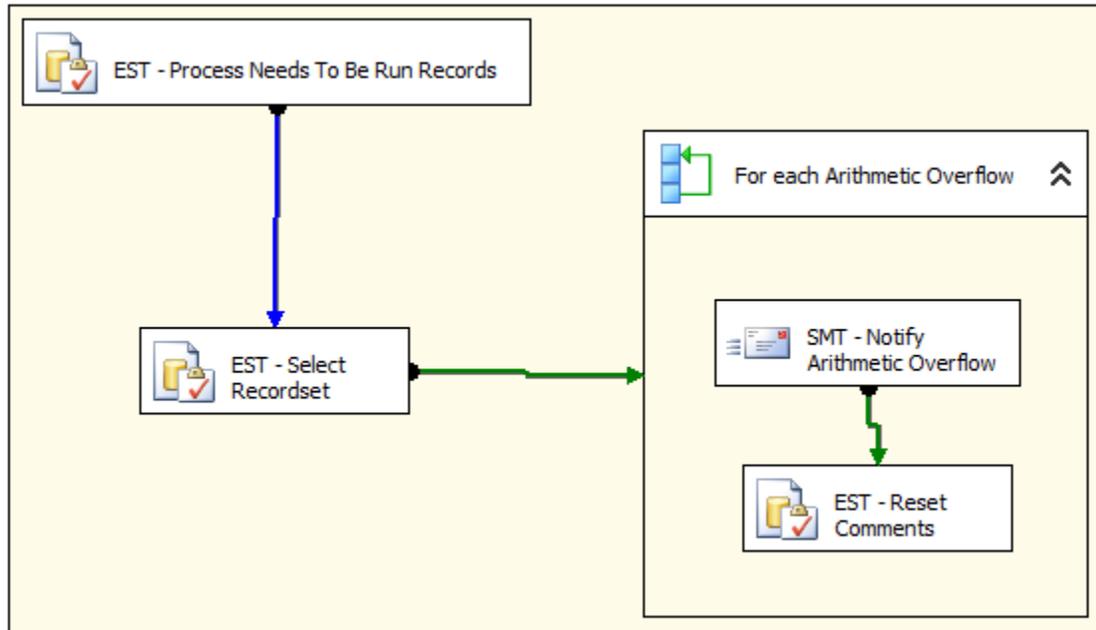


Figure 6

Tasks

EST – Process Needs To Be Run Records

Loop through all AutoStat records that have a "Needs to be Run" flag equal to 'Y' and recalculate their tcd.dbo.AutoStatResults data.

EST – Select Recordset

Select records flagged for Arithmetic Overflow.

EST – For Each Arithmetic Overflow

For each tcd.dbo.AutoStatResults record flagged as having Arithmetic Overflow.

SMT – Notify Arithmetic Overflow

Send email notification to the "MailTo" recipients stored in the TWINSConfig.dtsconfig file that arithmetic overflow occurred in the AutoStat calculations.

EST – Reset Comments

Reset the Comments of the Arithmetic Overflow record(s).

SMT – On Error Notify

Send email notification to the “MailTo” recipients stored in the TWINSConfig.dtsconfig file that an error has occurred during processing of the package.

Populate HTWOS.dtsx

This package is executed nightly.



Figure 7

Tasks

EST – Populate HTWOS

Executes the `bb_published.dbo.p_htwo` stored procedure which updates the `bb_published.dbo.htwo` and `bb_published.dbo.htwo_report_dates` tables.

SMT – On Error Send Mail

Send email notification to the “MailTo” recipients stored in the `TWINSConfig.dtsconfig` file that an error has occurred during processing of the package.

Data Integrity Check.dtsx

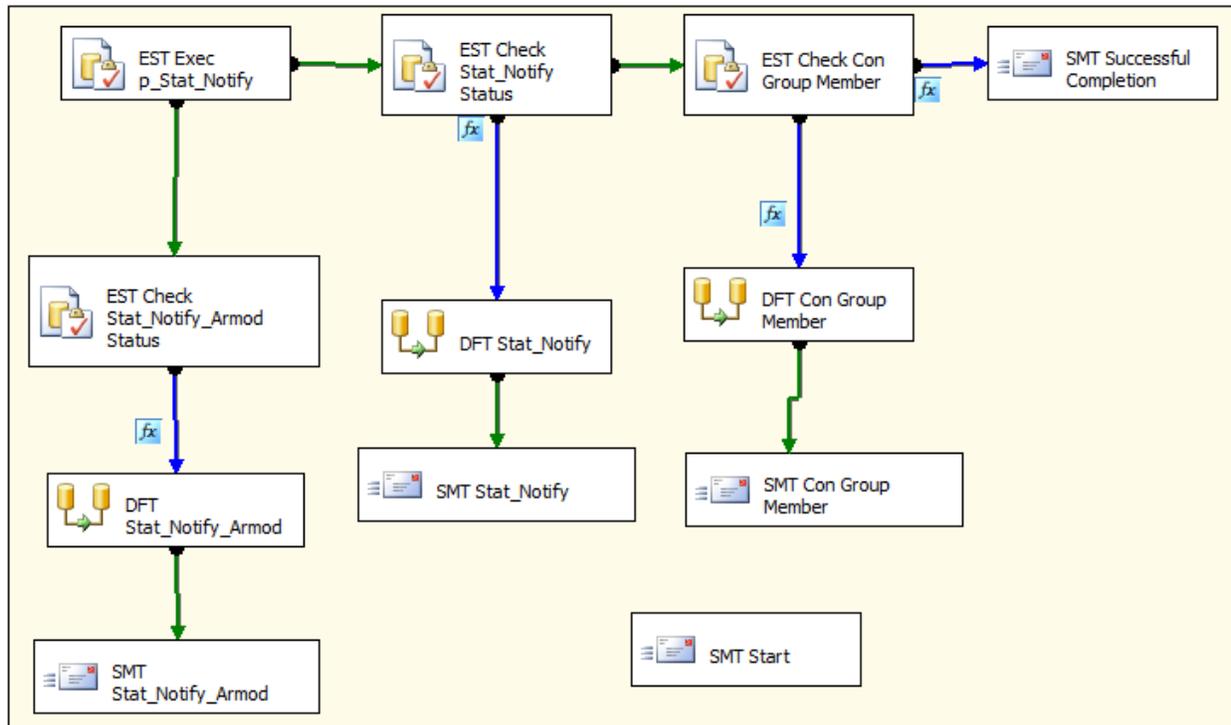


Figure 8

Tasks

SMT – Start

Send email notification to the “MailTo” recipients stored in the TWINSConfig.dtsconfig file that processing has begun.

EST – Exec p_Stat_Notify

Executes the tcd.dbo.p_Stat_Notify stored procedure. If this procedure finds data that has been changed that is of interest, it writes it to the tcd.dbo.statistician_notify and tcd.dbo.statistician_notify.armod tables.

EST – Check Stat_Notify_Armod

If there are records in the tcd.dbo.statistician_notify_armod table, it passes control to the DFT Stat_Notify_Armod data flow object.

DFT – Stat_Notify_Armod

Copies data from the tcd.dbo.statistician_notify_armod tables to the C:\temp\SSIS_Data_Integrity_Check_Stat_notify_Armod.txt file.

SMT – Stat_Notify_Notify_Armod

Sends an email to the “MailTo” recipients stored in the TWINSConfig.dtsconfig file. The file C:\temp\SSIS_Data_Integrity_Check_Stat_notify_Armod.txt is sent as an attachment.

EST – Check Stat_Notify

If there are records in the tcd.dbo.statistician_notify table, it passes control to the DFT Stat_Notify data flow object.

DFT – Stat_Notify

Copies data from the tcd.dbo.statistician_notify tables to the C:\temp\SSIS_Data_Integrity_Check_Stat_notify.txt file.

SMT – Stat_Notify

Sends an email to the “MailTo” recipients stored in the TWINSConfig.dtsconfig file. The file C:\temp\SSIS_Data_Integrity_Check_Stat_notify.txt is sent as an attachment.

EST – Check Con Group Member

If there are missing records in the constituent_group table, it passes control to the DFT Con Group Member data flow object.

DFT – Con Group Member

Copies constituent data that is missing from the constituent_group table to the C:\temp\SSIS_Data_Integrity_Check_Con_Group.txt file.

SMT – Con Group Member

Sends an email to the “MailTo” recipients stored in the TWINSConfig.dtsconfig file. The file C:\temp\SSIS_Data_Integrity_Check_Con_Group.txt is sent as an attachment.

SMT – Successful Completion

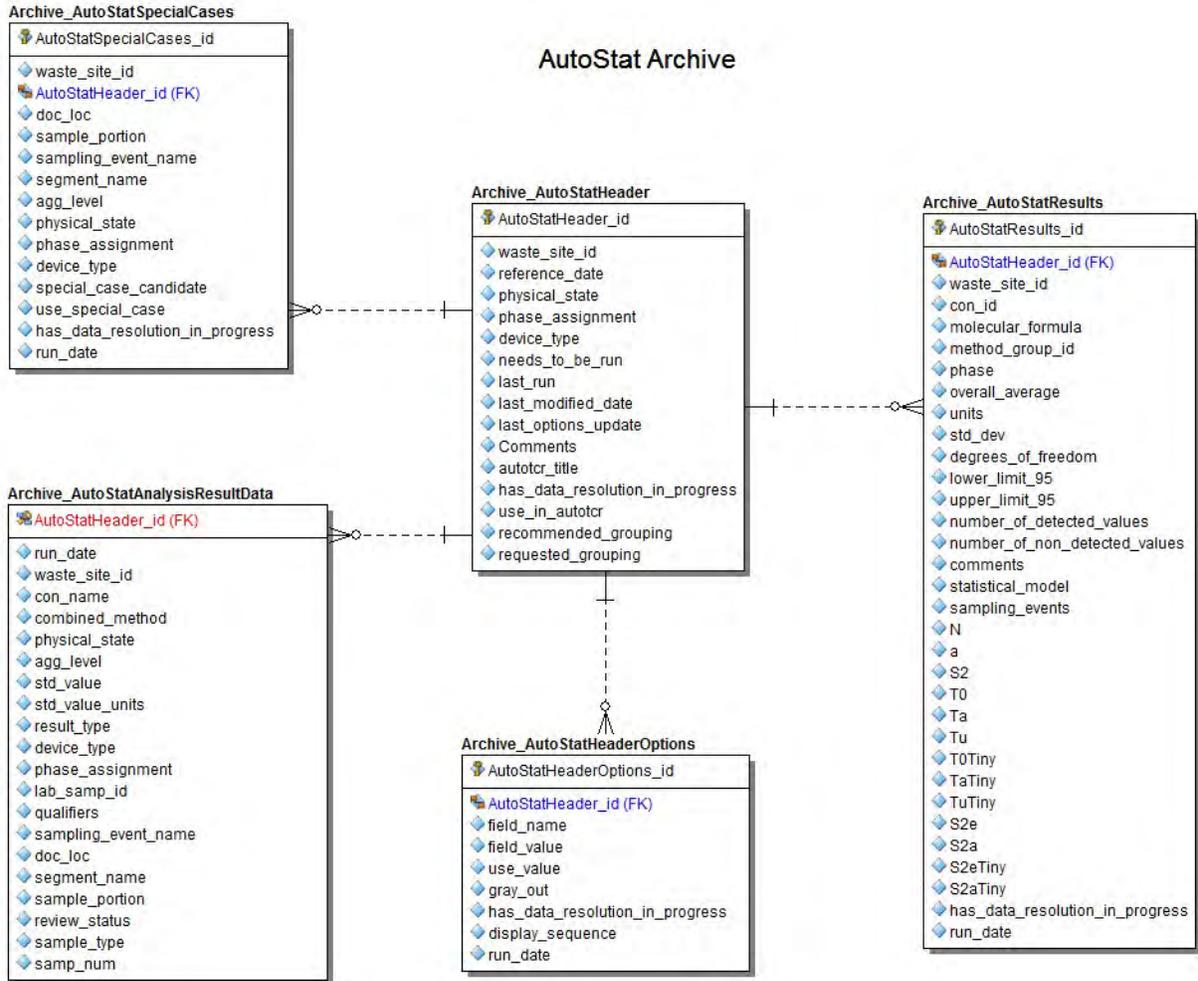
Send email notification to the “MailTo” recipients stored in the TWINSConfig.dtsconfig file stating that no problems were found. This message is sent only if email messages were not sent from the SMT Stat_Notify_Armod, SMT Stat_Notify and SMT Con Group Member tasks.

SMT – On Error Send Mail

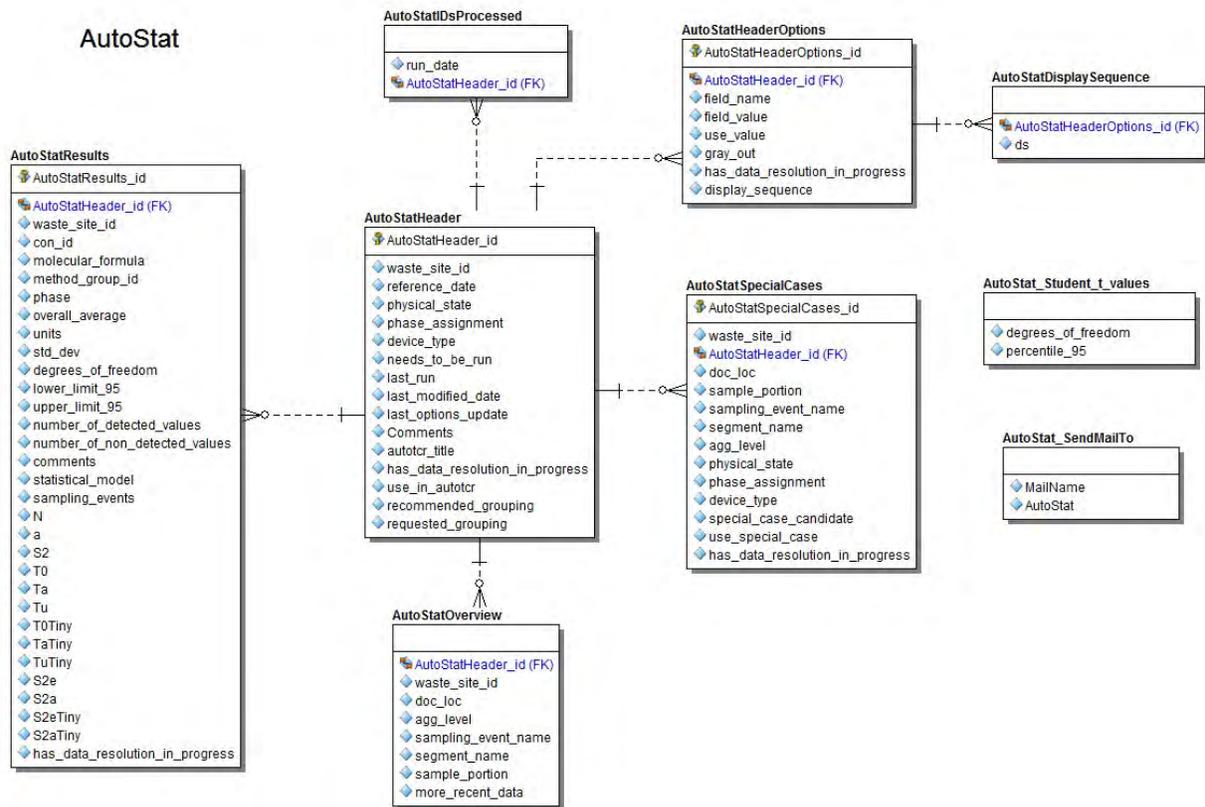
Send email notification to the “MailTo” recipients stored in the TWINSConfig.dtsconfig file that an error has occurred during processing of the package.

TWINS Entity Relationship Diagrams

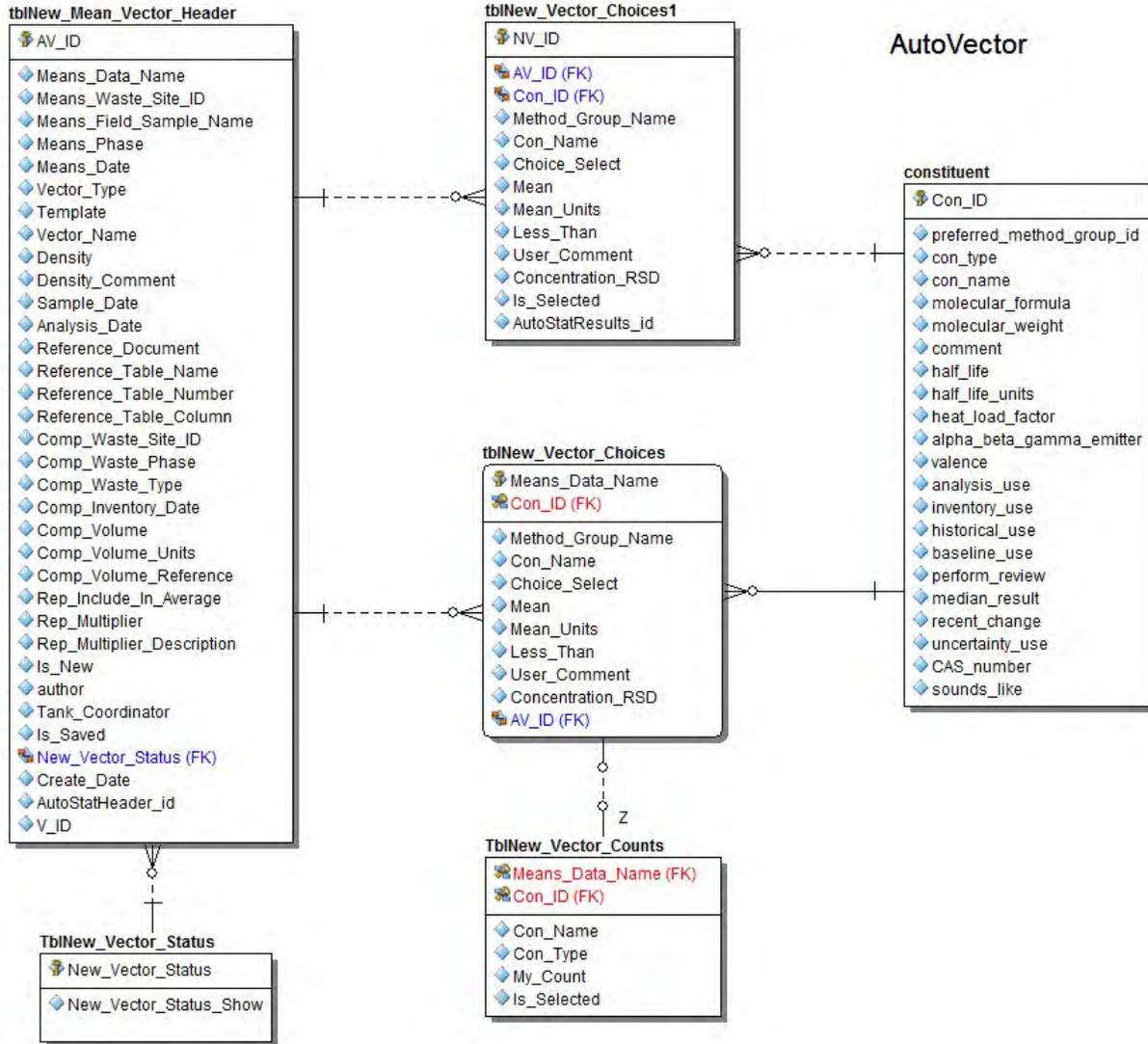
AutoStat Archive



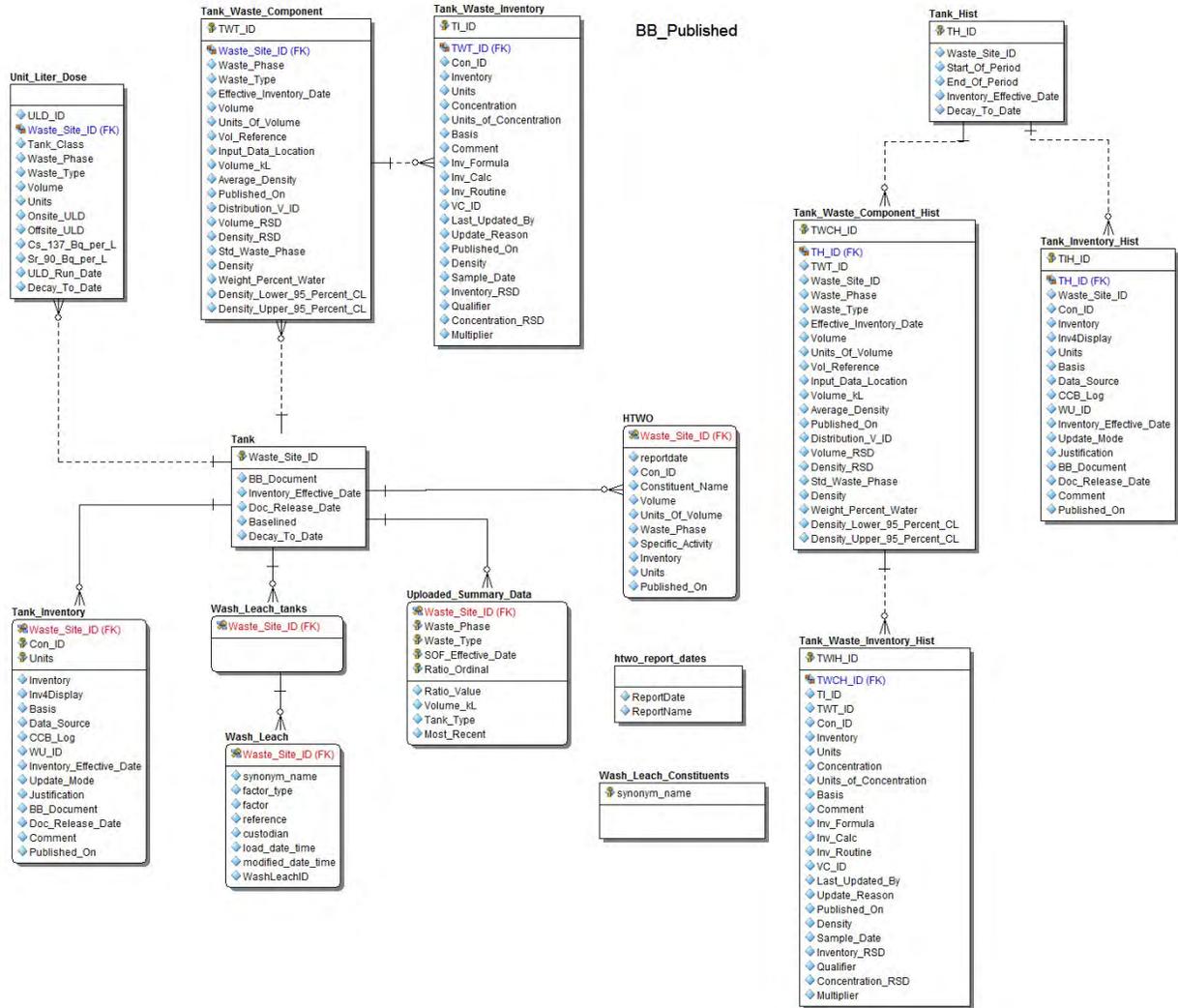
AutoStat

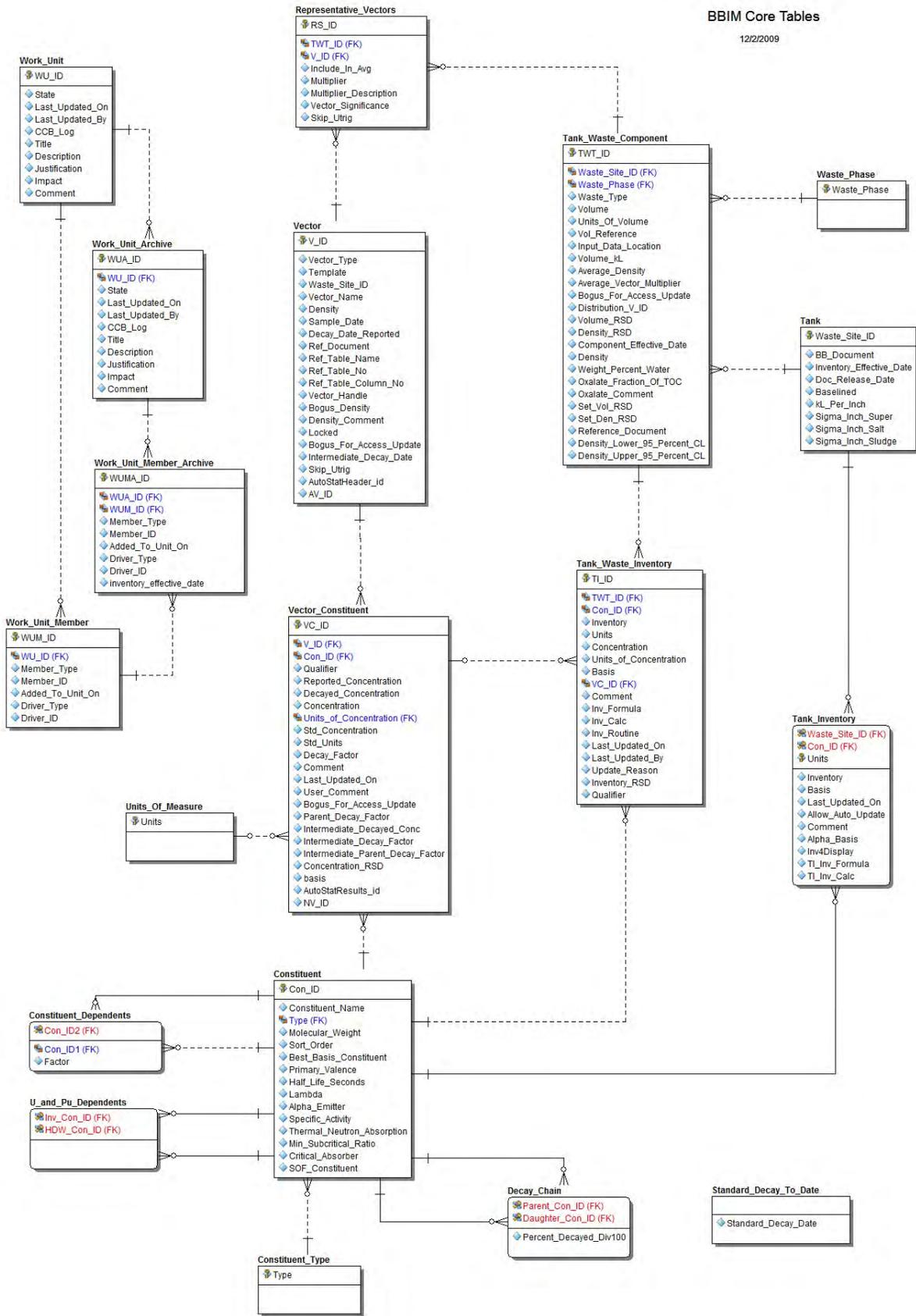


AutoVector

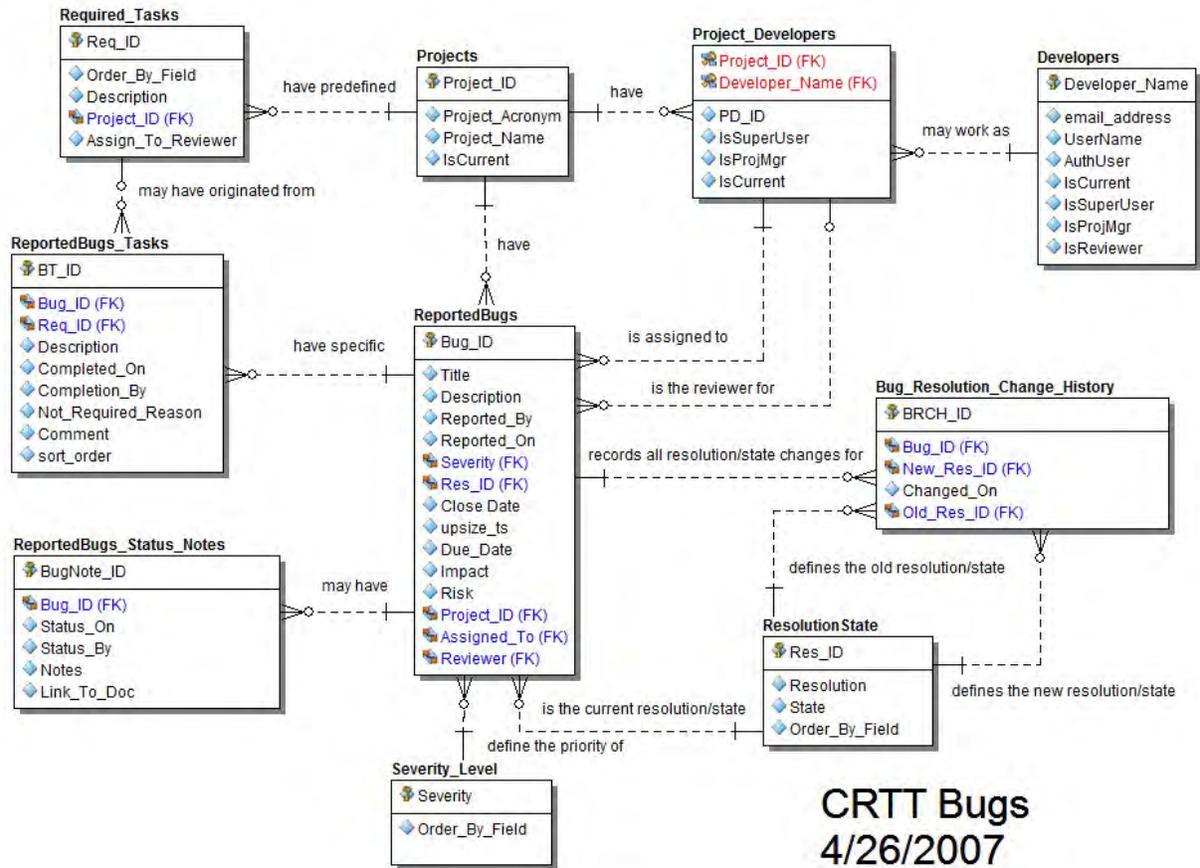


BBI_Published



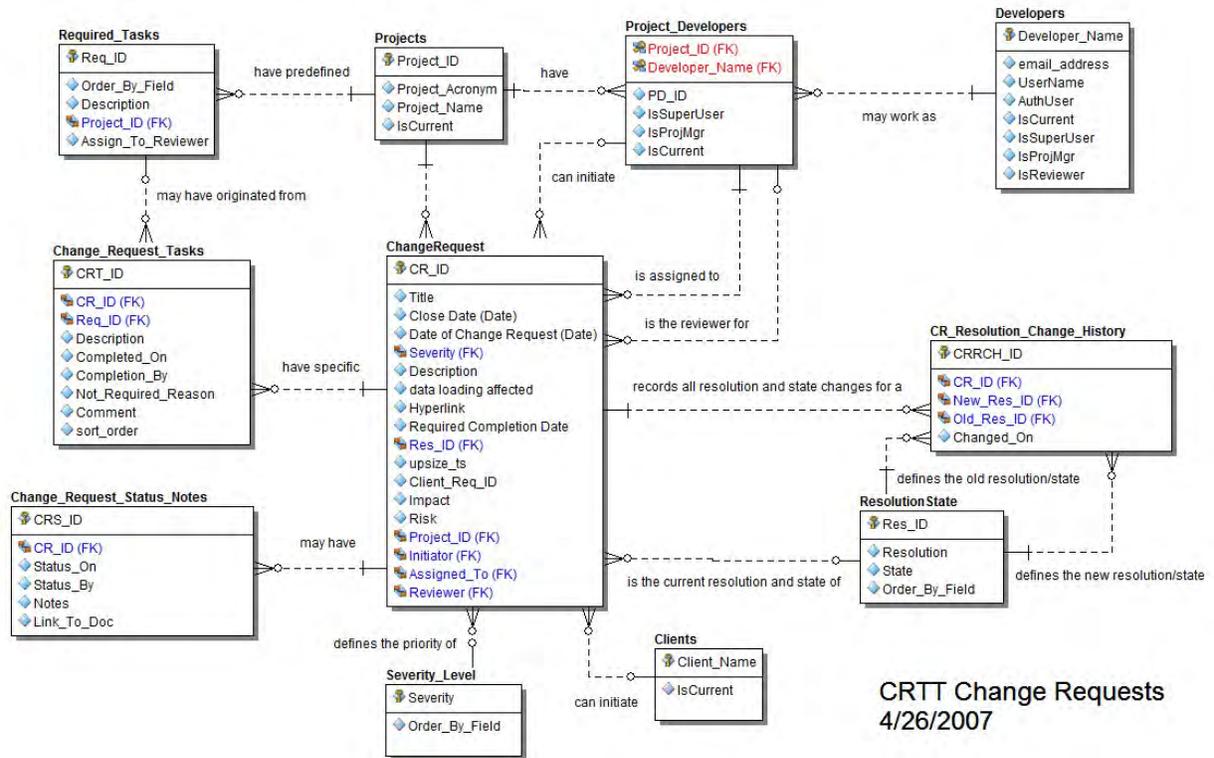


CRTT Bugs



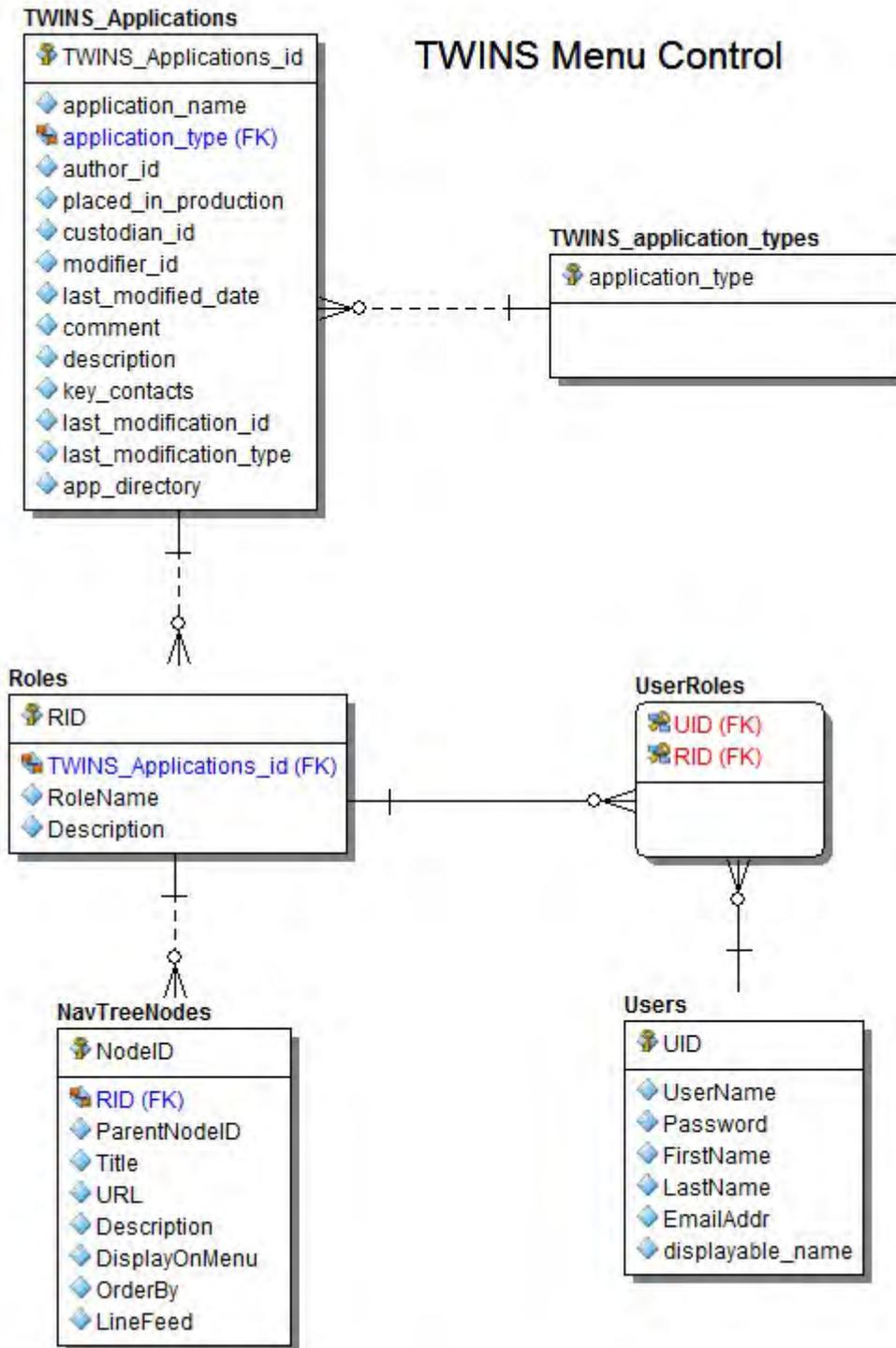
CRTT Bugs
4/26/2007

CRTT Change Requests

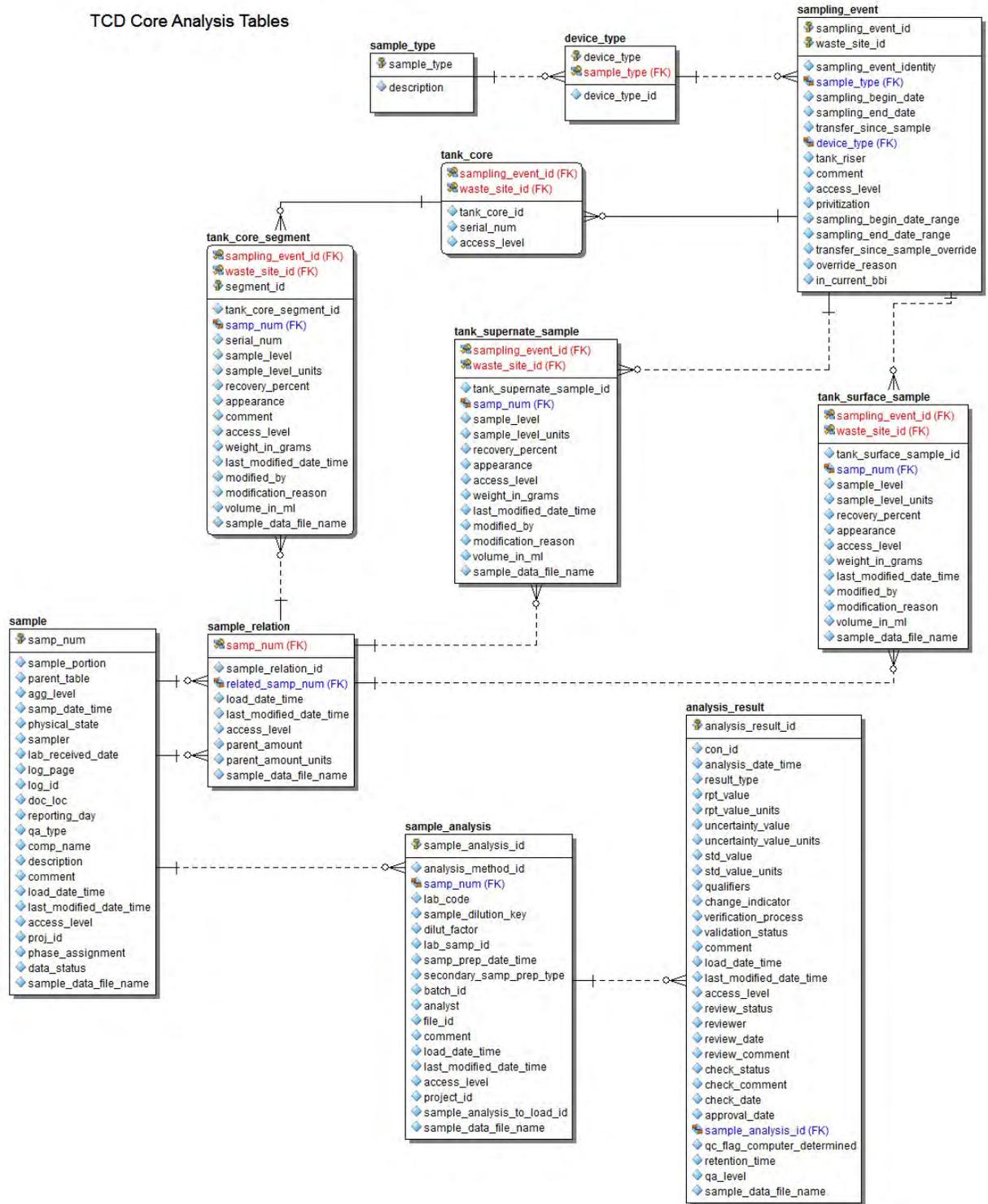


CRTT Change Requests
4/26/2007

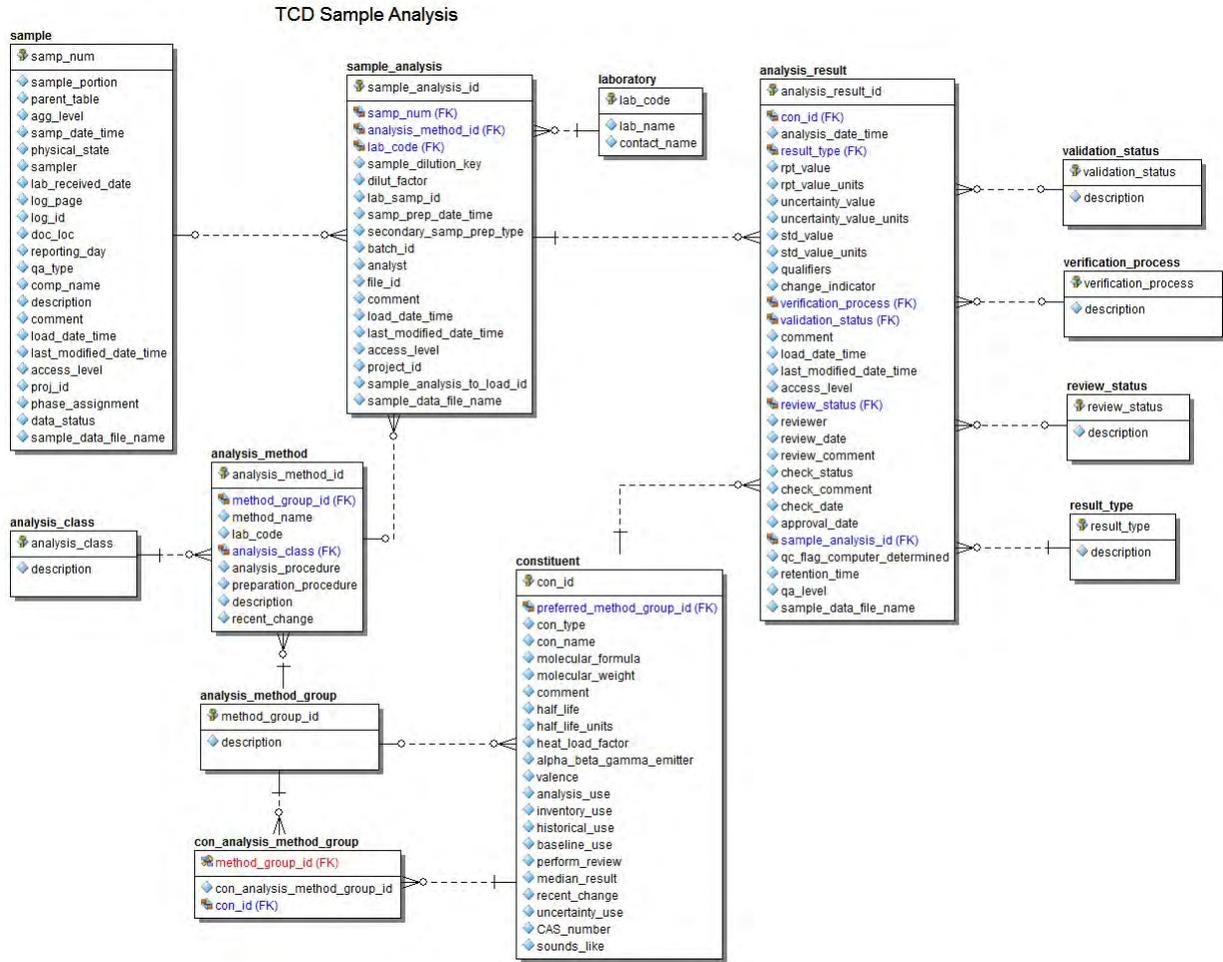
Menu Control



TCD Core Analysis Tables

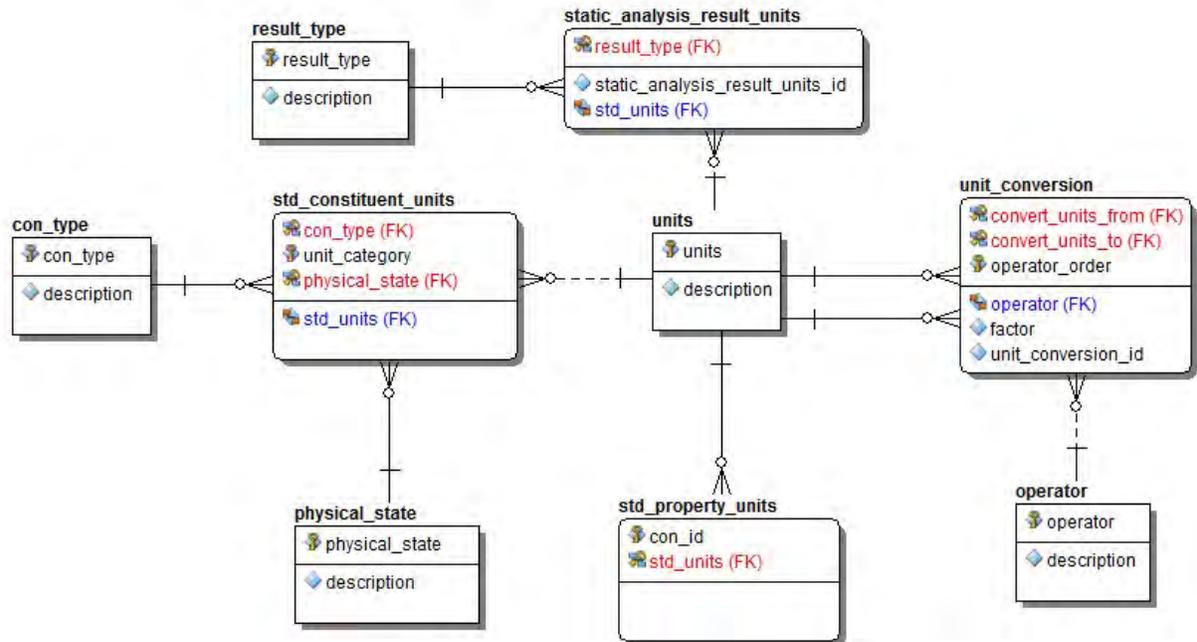


TCD Sample Analysis



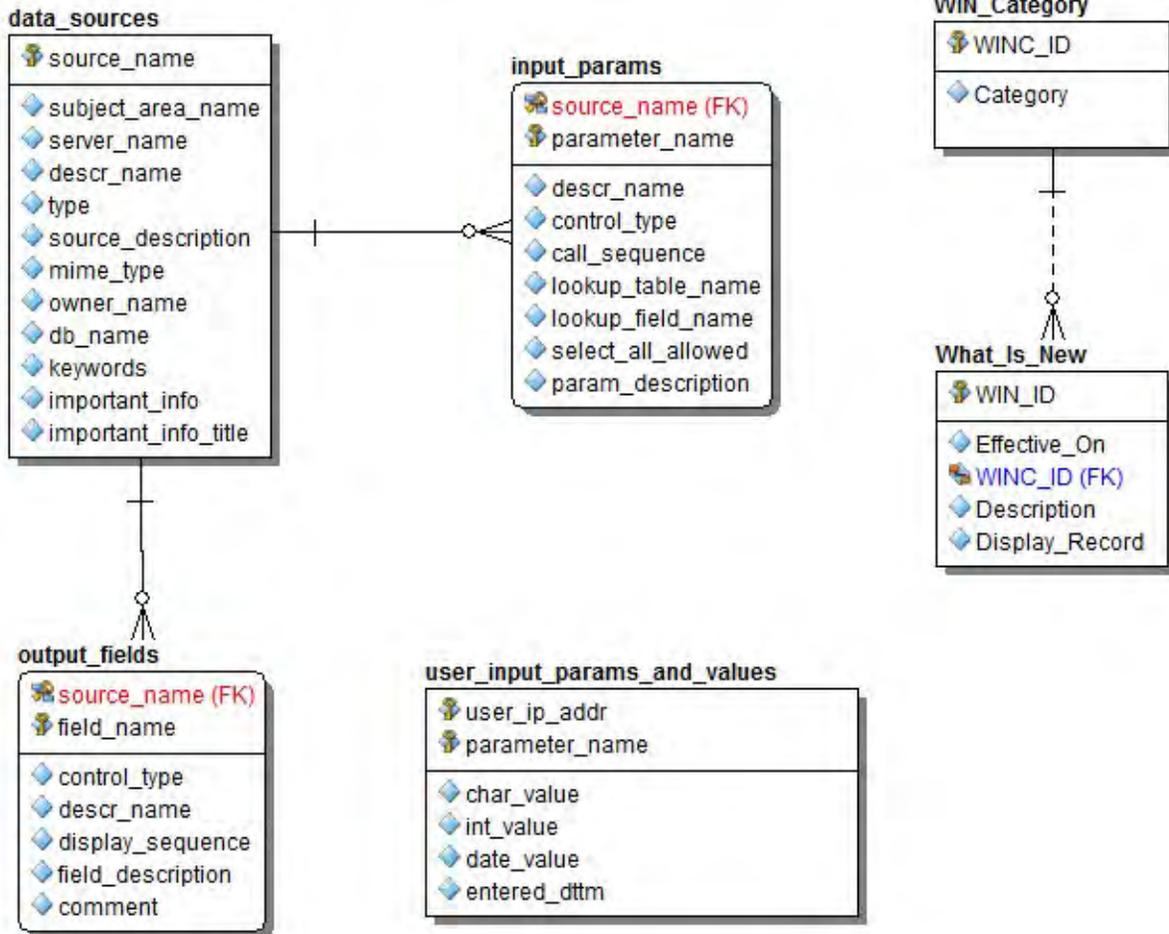
TCD Units

TCD Units

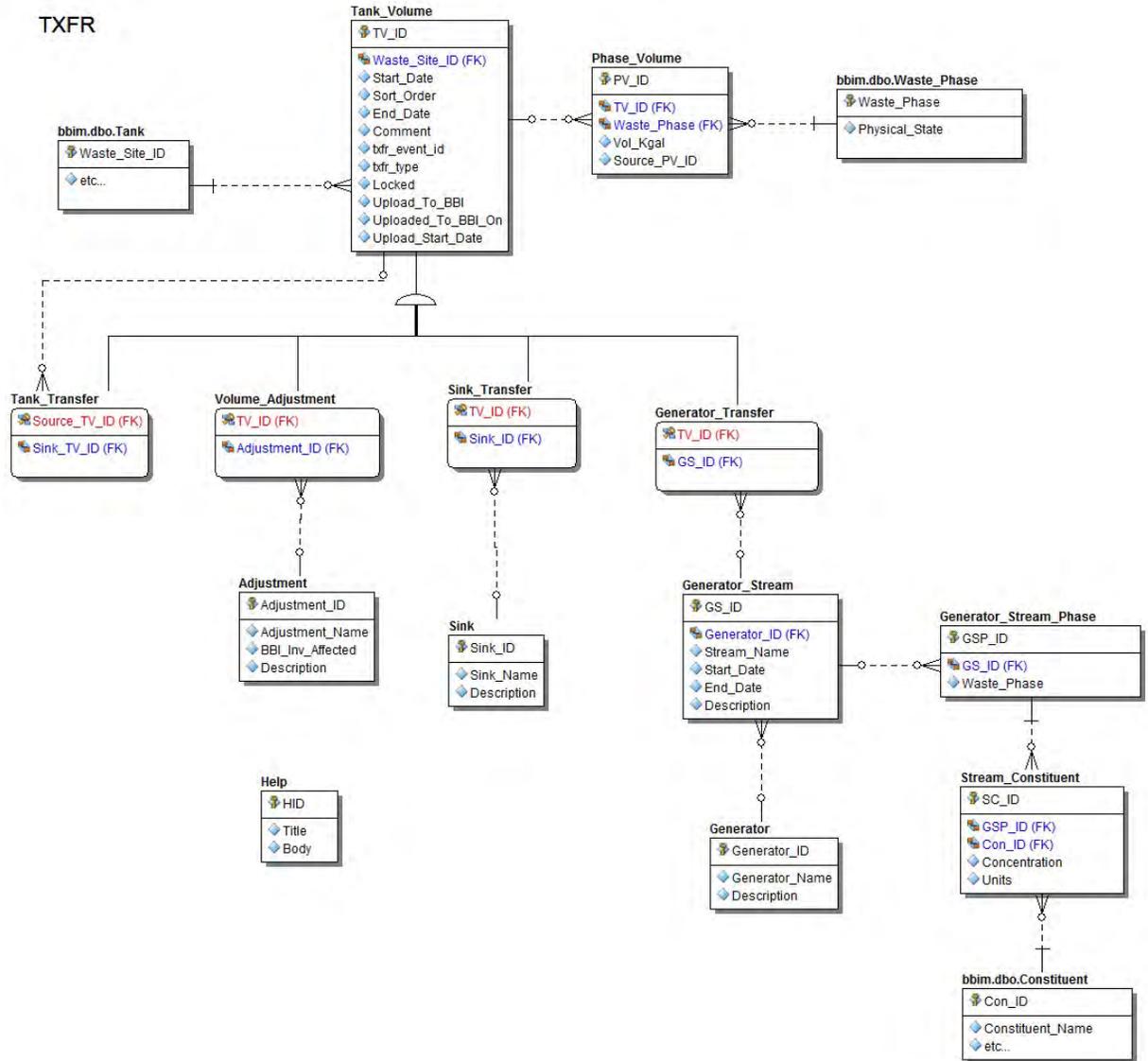


TWINS Web Interface

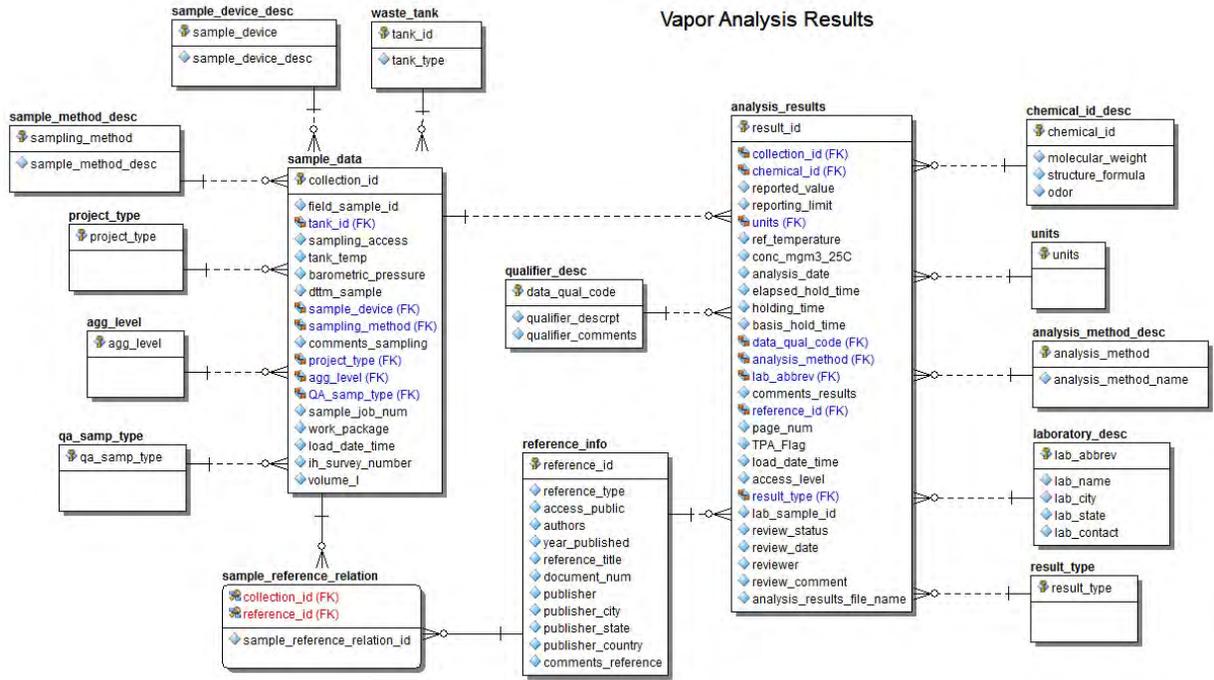
TWINS Web Interface



TXFR

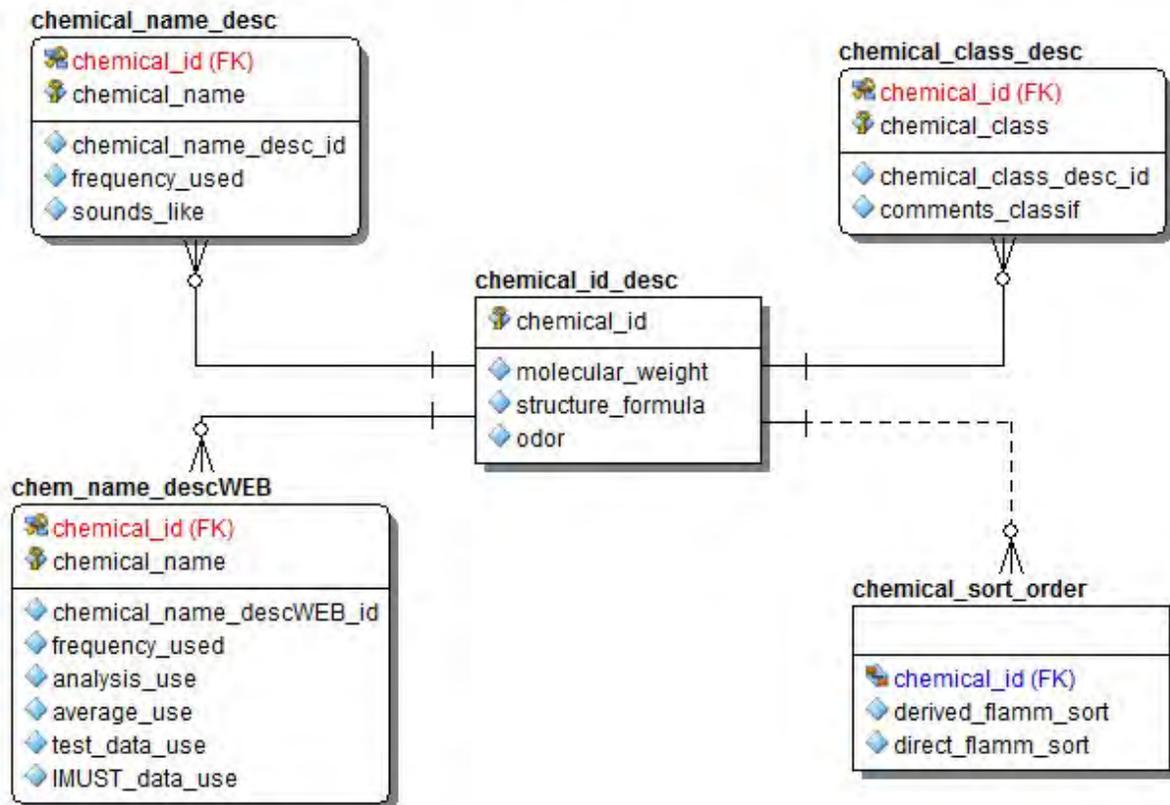


Vapor Analysis Results



Vapor Chemical IDs

Vapor Chemical IDs



Vapor Derived Tables

analysis_results_denormalized

<ul style="list-style-type: none"> 🔑 chemical_id 🔑 field_sample_id 🔑 result_type
<ul style="list-style-type: none"> 🔗 waste_site_id 🔗 chemical_name 🔗 molecular_weight 🔗 agg_level 🔗 QA_samp_type 🔗 reported_value 🔗 reporting_limit 🔗 units 🔗 ref_temperature 🔗 conc_mgm3_25C 🔗 ppmv 🔗 data_qual_code 🔗 sample_device 🔗 sampling_method 🔗 analysis_method 🔗 lab_abbrev 🔗 lab_sample_id 🔗 dttm_sample 🔗 analysis_date 🔗 elapsed_hold_time 🔗 holding_time 🔗 basis_hold_time 🔗 document_num 🔗 reference_title 🔗 comments_results 🔗 TPA_flag 🔗 access_level 🔗 result_id 🔗 table_name

Vapor Derived

avrgd_rprtd_values

<ul style="list-style-type: none"> 🔑 tank_id 🔑 chemical_id 🔑 analysis_method
<ul style="list-style-type: none"> 🔗 result_average_id 🔗 sample_job_num 🔗 sample_date 🔗 result_average 🔗 result_st_dev 🔗 reporting_limit 🔗 units 🔗 ref_temperature 🔗 num_sample_results 🔗 data_qual_code 🔗 sample_device 🔗 lab_abbrev 🔗 new_record 🔗 sampling_method 🔗 chemical_name 🔗 molecular_weight 🔗 mgm3 🔗 ppmv

flammability_results

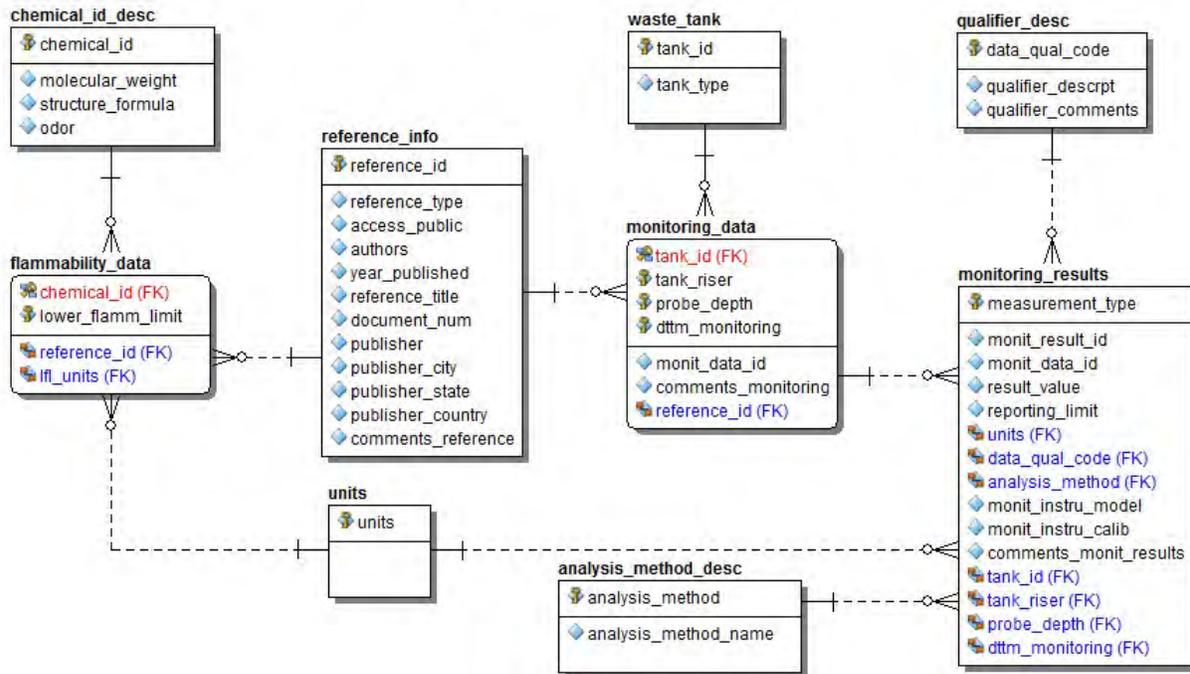
<ul style="list-style-type: none"> 🔑 tank_id 🔑 sample_job_num
<ul style="list-style-type: none"> 🔗 sample_date 🔗 chemical_name 🔗 lower_flamm_limit 🔗 lfl_units 🔗 avg_concentrations 🔗 units 🔗 percent_lfl

max_lengths

<ul style="list-style-type: none"> 🔑 max_lengths_id
<ul style="list-style-type: none"> 🔗 table_name 🔗 field_name 🔗 max_length

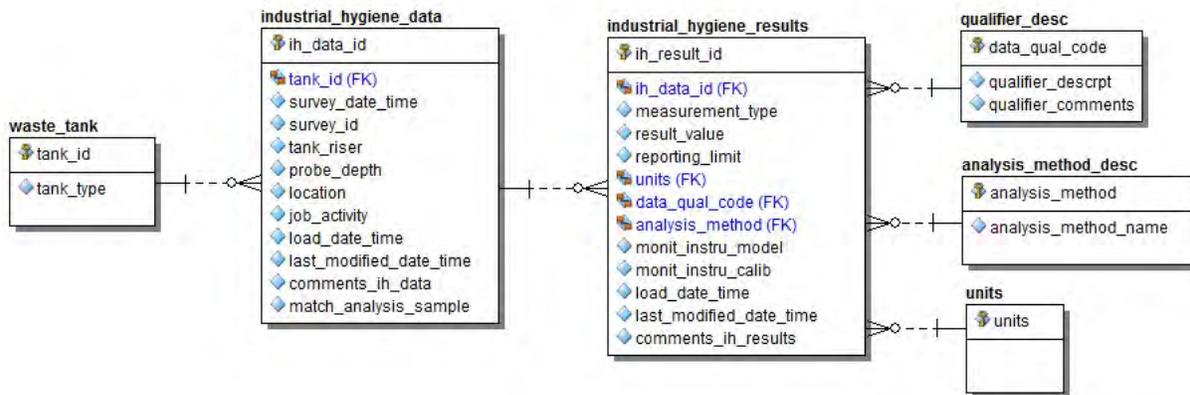
Vapor Flammability and Monitoring

Vapor Flammability & Monitoring



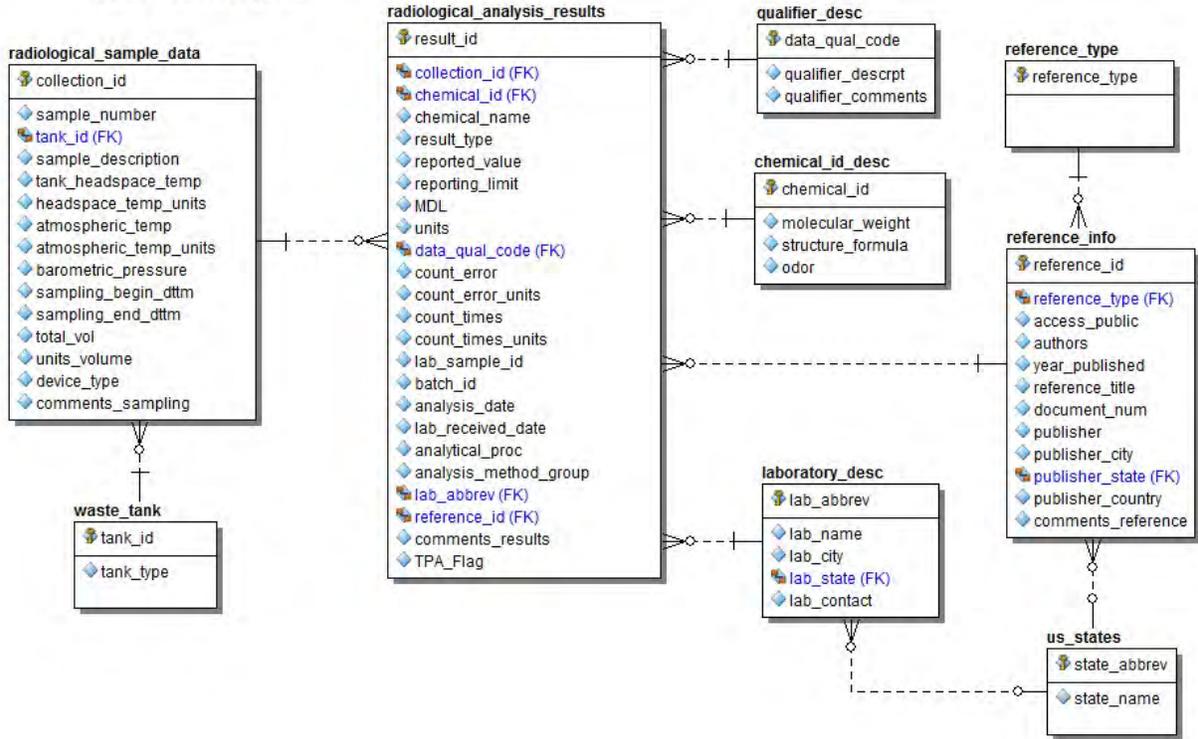
Vapor Industrial Hygiene

Vapor Industrial Hygiene



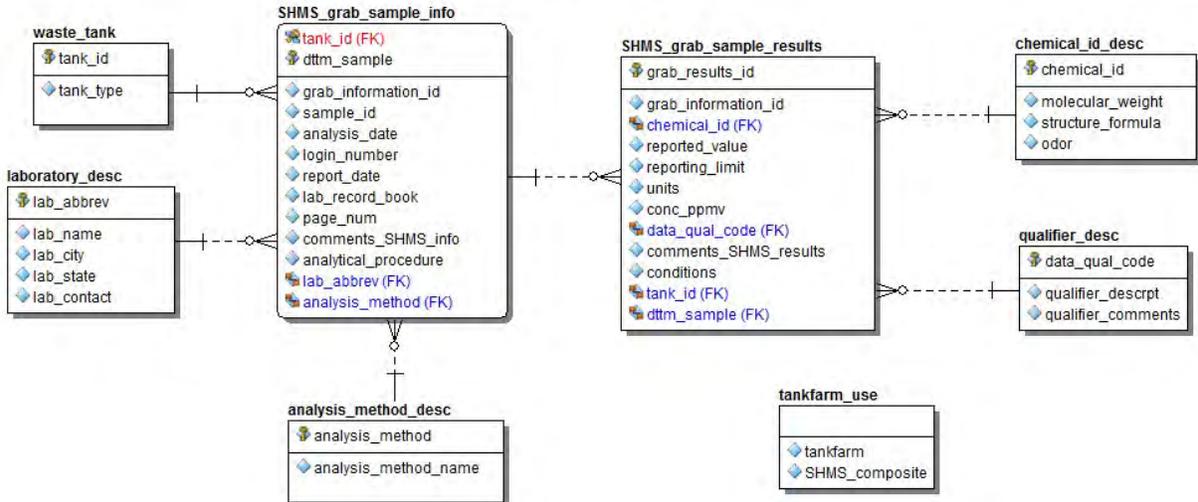
Vapor Radiological

Vapor Radiological



Vapor SHMS

Vapor SHMS



TWINS Data Dictionaries

BB_Published Database Documentation

Server name: twins

Database name: bb_published

Documentation date: 7/23/2010

Database size: 967.44 MB

Database description:

Table summary

	Name	Data size	Index size	Row Count
	dbo.Debug_Setting	8 KB	24 KB	0
	dbo.HTWO	147568 KB	8 KB	1789153
	dbo.htwo_report_dates	64 KB	8 KB	12
	dbo.OH_dates	8 KB	24 KB	11
	dbo.Standard_Waste_Phase	16 KB	24 KB	7
	dbo.Tank	32 KB	40 KB	183
	dbo.Tank_Hist	112 KB	56 KB	1227
	dbo.Tank_Inventory	4504 KB	576 KB	15427
	dbo.Tank_Inventory_Hist	21944 KB	1560 KB	104193
	dbo.Tank_Waste_Component	144 KB	32 KB	550
	dbo.Tank_Waste_Component_Hist	856 KB	104 KB	3980
	dbo.Tank_Waste_Inventory	15400 KB	2240 KB	44525
	dbo.Tank_Waste_Inventory_Hist	92528 KB	5616 KB	295830
	dbo.ULD_dose_conversion_factors	8 KB	16 KB	46
	dbo.Unit_Liter_Dose	192 KB	8 KB	1474
	dbo.Uploaded_Summary_Data	1008 KB	32 KB	9980
	dbo.Wash_Leach	1520 KB	8 KB	17964
	dbo.Wash_Leach_Constituents	8 KB	24 KB	127
	dbo.Wash_Leach_tanks	8 KB	24 KB	182
	dbo.Waste_Phase_Mapping	24 KB	40 KB	505

20 table(s)

TABLES

dbo.Debug_Setting

BBIM triggers and stored procedures "select" and on/off setting from the table Debug_Setting which enables them to turn on or off their debugging messages.

I	F	P	Column name	Data type	Nulls	Default	Description
			DS_ID	int	NO		Identity field, generated by the database.
			Type	varchar(4)	NO		The type of database object that can be debugged.
			Name	varchar(32)	NO		Name of the trigger or procedure in the BBIM database.
			Setting	smallint	NO		0 or -1, with 0 representing "turn OFF the debugging messages" and -1 representing "turn ON the debugging messages". -1 is used for "ON" because this corresponds to the MS Access checkbox in a form being "Checked".

Indexes

Index name	Column name	Sort direction	Is unique	Index type
U_Debug_Setting	Type	ASC	Yes	NONCLUSTERED
U_Debug_Setting	Name	ASC	Yes	NONCLUSTERED

Check constraints

Check name	Column name	Expression
CK_Setting	Setting	([Setting] = 0 or [Setting] = (-1))
CK_Type	Type	([Type] = 'PROC' or [Type] = 'TRIG')

dbo.HTWO

Hanford Tank Waste Operations Simulator (HTWO) data

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	Waste_Site_ID	varchar(15)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	reportdate	datetime	YES		Date that the data was reported.
	Con_ID	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
	Constituent_Name	varchar(50)	NO		Name of the constituent.
	Volume	float	YES		The volume, usually in kL, of this tank waste component. If a tank waste component with a waste phase of "Total" exists, then the sum of the volumes for any other waste components for this tank should equal the "Total" volume.
	Units_Of_Volume	varchar(12)	YES		Every instance in the database is kL, however we may run across other units such as kgal.
	Waste_Phase	varchar(50)	NO		The name of the waste phase. In addition, a value of "Total" is used to define all of the waste phases in a tank.
	Specific_Activity	float	YES		Of the radioisotope, measured in Ci/g. Used in the conversion of ug/g to uCi/g, and ug/mL to uCi/mL.
	Inventory	float	YES		The inventory value calculated for this constituent in the tank waste component.
	Units	varchar(12)	YES		Concentration units of the reported value or the reporting limit.
	Published_On	datetime	YES		Date and time that this record was copied over from either the TCD database or the BBIM database (update or insert).

 **dbo.htwo_report_dates**

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	ReportDate	datetime	YES		Date type format, down to the micro seconds.
	ReportName	varchar(50)	YES		More easlity readabile date format (month day, year).

dbo.OH_dates

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
			waste_phase	varchar(50)	NO		The name of the waste phase. In addition, a value of "Total" is used to define all of the waste phases in a tank.
			oh_date	datetime	NO		Date of the hydroxide value. (S) indicates an analytical sample and (E) indicates an engineering-based value.
			bbi_update	varchar(60)	YES		Best basis inventory update.
			comment	varchar(7398)	YES		Comment
			oh_date_id	int	NO		Identifer of the HO_dates record.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK_OH_dates	oh_date_id	ASC	Yes	CLUSTERED

dbo.Standard_Waste_Phase

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			Std_Waste_Phase	varchar(50)	NO		Baselined tanks have waste that is described by these phases: supernatant, sludge solid, sludge liquid, salt cake solid, salt cake liquid and retained gas.
			autotcr_sort_order	int	YES		A number indicating the order in which this item should appear in a list or document.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK_Standrd_Waste_Phase	Std_Waste_Phase	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.Waste_Phase_Mapping	FK_WPM_Std_Waste_Phase	PK_Standrd_Waste_Phase

dbo.Tank

The Tank table contains one record for each of the 177 HLW storage tanks at Hanford.

I	F	P	Column name	Data type	Nulls	Default	Description
			Waste_Site_ID	varchar(15)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
			BB_Document	varchar(50)	YES		The document number of the Best Basis Document where the loading of data for this tank was obtained.
			Inventory_Effective_Date	datetime	YES		The date on which the published inventory for this tank is in effect. Events, such as tank transfers and evaporations may have occurred since this date. In this case, the inventory would not reflect these events (changes). This also includes events such as core samples.
			Doc_Release_Date	datetime	YES		The date that the BB_Document was released.
			Baselined	smallint	YES		This Waste_site_id baselined , if -1 then baselined to 2001.
			Decay_To_Date	datetime	YES		Date inventory is decayed to

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_waste_site_id	Waste_Site_ID	ASC		NONCLUSTERED
Tank_PK	Waste_Site_ID	ASC	Yes	NONCLUSTERED

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.Unit_Liter_Dose	fk_ULD_Waste_Site_ID	Tank_PK

Check constraints

Check name	Column name	Expression
CK_Baselined	Baselined	([Baselined] = 0 or [Baselined] = (-1))

dbo.Tank_Hist

Snapshot of the Tank data over time.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			TH_ID	int	NO		Unique record identifier.
			Waste_Site_ID	varchar(15)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
			Start_Of_Period	datetime	NO		The beginning data of a date range.
			End_Of_Period	datetime	YES		The ending date of a date range. If End_Of_Period is NULL, then the date range is OPEN (e.g. range is from Start_Of_Period to NOW).
			Inventory_Effective_Date	datetime	YES		The date on which the published inventory for this tank is in effect. Events, such as tank transfers and evaporations may have occurred since this date. In this case, the inventory would not reflect these events (changes). This also includes events such as core samples.
			Decay_To_Date	datetime	YES		Date inventory is decayed to

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
TH_PK	TH_ID	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.Tank_Waste_Component_Hist	FK__Tank_Wast__TH_ID__753864A1	TH_PK
dbo.Tank_Inventory_Hist	FK__Tank_Inve__TH_ID__7814D14C	TH_PK

dbo.Tank_Inventory

The table where the calculated inventories for the constituents in each tank are stored. These data are an aggregate of the data found in the Tank_Waste_Inventory table, and are kept up-to-date by the Tank_Waste_Inventory triggers.

I	F	P	Column name	Data type	Nulls	Default	Description
			Waste_Site_ID	varchar(15)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
			Con_ID	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
			Inventory	float	NO		The inventory value calculated for this constituent in the tank waste component.
			Inv4Display	varchar(12)	NO	('N.NNE+00')	The inventory value, stored (converted) into scientific notation. Since this is a character field, it is only useful for display purposes. Examples are 1.23E+00; -5.09E99; 4.00E-31
			Units	varchar(12)	NO		Concentration units of the reported value or the reporting limit.
			Basis	varchar(20)	NO		The basis or data source of the inventory calculation. Multiple basis (Tank_Inventory table only) are concatenated using the "/" character (e.g. S/E/T (241-BX-112)).
			Data_Source	varchar(4)	NO		Database from which this record's data was copied. Originally, all records were copied over from the TCD database, however over time, the records now all come from the BBIM database.
			CCB_Log	varchar(40)	NO		The Log number of the Configuration Control Board's documentation that authorizes this work unit (change in BBIM data). Current examples are "CCB-001" up to "CCB-064".
			WU_ID	int	NO	(0)	Computer generated unique identifier for the Work_Unit record.
			Inventory_Effective_Date	datetime	NO		The date on which the published inventory for this tank is in effect. Events, such as tank transfers and evaporations may have occurred since this date. In this case, the inventory would not reflect these events (changes). This also includes events such as core samples.
			Update_Mode	varchar(7)	YES		Can be one of three states: 'Locked' which means that this record will not be overwritten when data is 'published' from the BBIM (e.g. the database triggers and stored procedures will NOT overwrite this record); 'Force Update' which means that when data is 'published' from the BBIM, the triggers and stored procedures should overwrite this record; 'Default' which means that this record will only get overwritten by BBIM 'published' data if their RPD is < 1.
			Justification	varchar(80)	YES		Reason for the change.
			BB_Document	varchar(50)	YES		The document number of the Best Basis Document where the loading of data for this tank was obtained.
			Doc_Release_Date	datetime	YES		The date that the BB_Document was released.
			Comment	varchar(255)	YES		Comment
			Published_On	datetime	YES	(getdate())	Date and time that this record was copied over from either the TCD database or the BBIM database (update or insert).

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_tank_inventory	Waste_Site_ID	ASC		NONCLUSTERED
Tank_Inventory_PK	Waste_Site_ID	ASC	Yes	CLUSTERED
Tank_Inventory_PK	Con_ID	ASC	Yes	CLUSTERED
Tank_Inventory_PK	Units	ASC	Yes	CLUSTERED

Triggers

Trigger name: dbo.Tank_Inventory_IUTrig
 Trigger type: INSERT, UPDATE
 Created on: 12 Dec 2001
 Trigger active: Yes

dbo.Tank_Inventory_Hist

Snapshot of the Tank Inventory data over time.

I F P	Column name	Data type	Nulls	Default	Description
	TIH_ID	int	NO		Unique record identifier.
	TH_ID	int	NO		Link to the parent Tank_Hist record.
	Waste_Site_ID	varchar(15)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	Con_ID	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
	Inventory	float	NO		The inventory value calculated for this constituent in the tank waste component.
	Inv4Display	varchar(12)	NO		The inventory value, stored (converted) into scientific notation. Since this is a character field, it is only useful for display purposes. Examples are 1.23E+00; -5.09E99; 4.00E-31
	Units	varchar(12)	NO		Concentration units of the reported value or the reporting limit.
	Basis	varchar(20)	NO		The basis or data source of the inventory calculation. Multiple basis (Tank_Inventory table only) are concatenated using the "/" character (e.g. S/E/T (241-BX-112)).
	Data_Source	varchar(4)	NO		Database from which this record's data was copied. Originally, all records were copied over from the TCD database, however over time, the records now all come from the BBIM database.
	CCB_Log	varchar(40)	NO		The Log number of the Configuration Control Board's documentation that authorizes this work unit (change in BBIM data). Current examples are "CCB-001" up to "CCB-064".
	WU_ID	int	NO		Computer generated unique identifier for the Work_Unit record.
	Inventory_Effective_Date	datetime	NO		The date on which the published inventory for this tank is in effect. Events, such as tank transfers and evaporations may have occurred since this date. In this case, the inventory would not reflect these events (changes). This also includes events such as core samples.
	Update_Mode	varchar(7)	YES		Can be one of three states: 'Locked' which means that this record will not be overwritten when data is 'published' from the BBIM (e.g. the database triggers and stored procedures will NOT overwrite this record); 'Force Update' which means that when data is 'published' from the BBIM, the triggers and stored procedures should overwrite this record; 'Default' which means that this record will only get overwritten by BBIM 'published' data if their RPD is < 1.
	Justification	varchar(80)	YES		Reason for the change.
	BB_Document	varchar(50)	YES		The document number of the Best Basis Document where the loading of data for this tank was obtained.
	Doc_Release_Date	datetime	YES		The date that the BB_Document was released.
	Comment	varchar(255)	YES		Comment
	Published_On	datetime	YES		Date and time that this record was copied over from either the TCD database or the BBIM database (update or insert).

Indexes

Index name	Column name	Sort direction	Is unique	Index type
TIH_PK	TIH_ID	ASC	Yes	NONCLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__Tank_Inve__TH_ID__7814D14C	TH_ID	TH_PK (dbo.Tank_Hist)	

dbo.Tank_Waste_Component

In the best basis model, inventories are calculated down to the waste component in each tank. Waste components are usually equivalent to the waste phases in the tank. Sometimes waste phases are broken down further by defining waste types within a waste phase. A tank inventory can be calculated by summing the inventories of each tank waste component.

I	F	P	Column name	Data type	Nulls	Default	Description									
			TWT_ID	int	NO		A computer generated unique identifier for the Tank_Waste_Component record. (TWT_ID is historicalthe table name was changed from Tank_Waste_Type to Tank_Waste_Component).									
			Waste_Site_ID	varchar(15)	YES		Tank Identifier, using the format of 241-Tank Farm-Tank #.									
			Waste_Phase	varchar(50)	NO		The name of the waste phase. In addition, a value of "Total" is used to define all of the waste phases in a tank.									
			Waste_Type	varchar(60)	YES		Abbeviation for the name of the waste. Some Examples: <table border="0"> <tr> <td>AW</td> <td>CC</td> <td>CP</td> </tr> <tr> <td>DC</td> <td>DN</td> <td>IL</td> </tr> <tr> <td>SLUDGE</td> <td>SOLID_CAKE</td> <td>SUPER</td> </tr> </table>	AW	CC	CP	DC	DN	IL	SLUDGE	SOLID_CAKE	SUPER
AW	CC	CP														
DC	DN	IL														
SLUDGE	SOLID_CAKE	SUPER														
			Effective_Inventory_Date	datetime	YES		The date on which the published inventory for this tank is in effect. Events, such as tank transfers and evaporations may have occurred since this date. In this case, the inventory would not reflect these events (changes). This also includes events such as core samples.									
			Volume	float	YES		The volume, usually in kL, of this tank waste component. If a tank waste component with a waste phase of "Total" exists, then the sum of the volumes for any other waste components for this tank should equal the "Total" volume.									
			Units_Of_Volume	varchar(12)	YES		Every instance in the database is kL, however we may run across other units such as kgal.									
			Vol_Reference	varchar(80)	YES		If the value for the volume was obtained from a document other than the best basis document for this tank (see tank.BB_Document), then the document name or document number from which the volume was obtained, otherwise NULL.									
			Input_Data_Location	varchar(80)	YES		During the loading of data into the bbim database, tables are extracted from the electronic versions of best basis documents and saved for future use. Input_Data_Location is the path and filename where these tables are stored, relative to the \\TCD_DOCS\best_basis_baseline file share.									
			Volume_kL	float	YES		The volume, in kL or kg, of this tank waste component. The Insert & Update triggers calculate this based upon the Volume & Units_of_Volume fields. If the volume is in kg and the Vector_Constituent "concentrations" are in units other than kg or wt%, inventories will not be calculated correctly.									
			Average_Density	float	YES	(0)	If Representative_Vectors exist for this Tank_Waste_Component, the Average_Density equals the average of all Vector.Density values for which the Representative_Vectors.Include_In_Avg field is "Y".									
			Published_On	datetime	YES	(getdate())	Date and time that this record was copied over from either the TCD database or the BBIM database (update or insert).									
			Distribution_V_ID	int	YES		Vector Identifier of the vector to be used to distribute the missing HDW and Alpha nuclides									
			Volume_RSD	float	YES		Relative Standard Deviation for Volume, in units of percentage. Equal to Std_Dev / Volume									
			Density_RSD	float	YES		Relative Standard Deviation for Density, in units of percentage. Equal to Std_Dev / Density.									
			Std_Waste_Phase	varchar(50)	YES		Baselined tanks have waste that is described by these phases: supernatant, sludge solid, sludge liquid, salt cake solid, salt cake liquid and retained gas.									
			Density	float	YES		The density, in g/ml of the vector. A density is needed to calculate inventories when vector constituents are recorded as concentrations. If vector constituents values are recorded in units of inventory (kg or Ci), then Density is not needed.									
			Weight_Percent_Water	float	YES		Weight percent water for this tank from BBI									

Density_Lower_95_Percent_C float YES

Density_Upper_95_Percent_C float YES

Component Density Lower 95% Confidence Limit in units of g/mL. The 95% Two-Sided Confidence Intervals are calculated only for Double-Shell Tanks and only when sufficient sample determinations of the waste density are available.
Component Density Upper 95% Confidence Limit in units of g/mL. The 95% Two-Sided Confidence Intervals are calculated only for Double-Shell Tanks and only when sufficient sample determinations of the waste density are available.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
I1_Tank_Waste_Component	Waste_Site_ID	ASC		CLUSTERED
TWC_PK	TWT_ID	ASC	Yes	NONCLUSTERED

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.Waste_Phase_Mapping	FK_WPM_TWT_ID	TWC_PK

Triggers

Trigger name: dbo.Tank_Waste_Component_DTrig
Trigger type: DELETE

Created on: 27 Apr 2000
Trigger active: Yes

dbo.Tank_Waste_Component_Hist

Snapshot of the Tank Waste Component data over time.

I F P	Column name	Data type	Nulls	Default	Description									
PK	TWCH_ID	int	NO		Unique Identifier for the Tank_Waste_Component_Hist table.									
FK	TH_ID	int	NO		Link to the parent Tank_Hist record.									
	TWT_ID	int	NO		A computer generated unique identifier for the Tank_Waste_Component record. (TWT_ID is historicalthe table name was changed from Tank_Waste_Type to Tank_Waste_Component).									
	Waste_Site_ID	varchar(15)	YES		Tank Identifier, using the format of 241-Tank Farm-Tank #.									
	Waste_Phase	varchar(50)	NO		The name of the waste phase. In addition, a value of "Total" is used to define all of the waste phases in a tank.									
	Waste_Type	varchar(60)	YES		Abbreviation for the name of the waste. Some Examples: <table border="0"> <tr> <td>AW</td> <td>CC</td> <td>CP</td> </tr> <tr> <td>DC</td> <td>DN</td> <td>IL</td> </tr> <tr> <td>SLUDGE</td> <td>SOLID_CAKE</td> <td>SUPER</td> </tr> </table>	AW	CC	CP	DC	DN	IL	SLUDGE	SOLID_CAKE	SUPER
AW	CC	CP												
DC	DN	IL												
SLUDGE	SOLID_CAKE	SUPER												
	Effective_Inventory_Date	datetime	YES		The date on which the published inventory for this tank is in effect. Events, such as tank transfers and evaporations may have occurred since this date. In this case, the inventory would not reflect these events (changes). This also includes events such as core samples.									
	Volume	float	YES		The volume, usually in kL, of this tank waste component. If a tank waste component with a waste phase of "Total" exists, then the sum of the volumes for any other waste components for this tank should equal the "Total" volume.									
	Units_Of_Volume	varchar(12)	YES		Every instance in the database is kL, however we may run across other units such as kgal.									
	Vol_Reference	varchar(80)	YES		If the value for the volume was obtained from a document other than the best basis document for this tank (see tank.BB_Document), then the document name or document number from which the volume was obtained, otherwise NULL.									
	Input_Data_Location	varchar(80)	YES		During the loading of data into the bbim database, tables are extracted from the electronic versions of best basis documents and saved for future use. Input_Data_Location is the path and filename where these tables are stored, relative to the \\TCD_DOCS\best_basis_baseline file share.									
	Volume_kL	float	YES		The volume, in kL or kg, of this tank waste component. The Insert & Update triggers calculate this based upon the Volume & Units_of_Volume fields. If the volume is in kg and the Vector_Constituent "concentrations" are in units other than kg or wt%, inventories will not be calculated correctly.									
	Average_Density	float	YES		If Representative_Vectors exist for this Tank_Waste_Component, the Average_Density equals the average of all Vector.Density values for which the Representative_Vectors.Include_In_Avg field is "Y".									
	Published_On	datetime	YES		Date and time that this record was copied over from either the TCD database or the BBIM database (update or insert).									
	Distribution_V_ID	int	YES		Vector Identifier of the vector to be used to distribute the missing HDW and Alpha nuclides									
	Volume_RSD	float	YES		Relative Standard Deviation for Volume, in units of percentage. Equal to Std_Dev / Volume									
	Density_RSD	float	YES		Relative Standard Deviation for Density, in units of percentage. Equal to Std_Dev / Density.									
	Std_Waste_Phase	varchar(50)	YES		Baselined tanks have waste that is described by these phases: supernatant, sludge solid, sludge liquid, salt cake solid, salt cake liquid and retained gas.									

Density	float	YES	The density, in g/ml of the vector. A density is needed to calculate inventories when vector constituents are recorded as concentrations. If vector constituents values are recorded in units of inventory (kg or Ci), then Density is not needed. Weight percent water for this tank from BBI
Weight_Percent_Water	float	YES	
Density_Lower_95_Percent_C L	float	YES	Component Density Lower 95% Confidence Limit in units of g/mL. The 95% Two-Sided Confidence Intervals are calculated only for Double-Shell Tanks and only when sufficient sample determinations of the waste density are available.
Density_Upper_95_Percent_C L	float	YES	Component Density Upper 95% Confidence Limit in units of g/mL. The 95% Two-Sided Confidence Intervals are calculated only for Double-Shell Tanks and only when sufficient sample determinations of the waste density are available.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
TWCH_PK	TWCH_ID	ASC	Yes	NONCLUSTERED

Foreign keys

<u>Constraint name</u>	<u>Column name</u>	<u>Reference</u>	<u>Description</u>
FK__Tank_Wast__TH_ID__753 864A1	TH_ID	TH_PK (dbo.Tank_Hist)	

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.Tank_Waste_Inventory_Hist	FK__Tank_Wast__TWCH__7AF13DF7	TWCH_PK

dbo.Tank_Waste_Inventory

The table where the calculated inventories for the constituents in each tank waste component are stored. A sum of constituent inventories by tank will result in the tank inventory. A sum of inventories by constituent will result in a "global" (site) inventory.

I	F	P	Column name	Data type	Nulls	Default	Description
ix	ix		TI_ID	int	NO		Computer generated unique identifier for the tank waste inventory record.
ix			TWT_ID	int	YES		A computer generated unique identifier for the Tank_Waste_Component record. (TWT_ID is historical the table name was changed from Tank_Waste_Type to Tank_Waste_Component).
ix			Con_ID	varchar(15)	YES		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
			Inventory	float	YES		The inventory value calculated for this constituent in the tank waste component.
			Units	varchar(12)	YES		Concentration units of the reported value or the reporting limit.
			Concentration	float	YES		The concentration value for this constituent in the tank waste component which was used to calculate the Inventory value. Note that this is the Adjusted Concentration, e.g. it has had the Multiplier applied to it, and it will be different than the unadjusted concentration that is shown in the Inventory Calculation field.
			Units_of_Concentration	varchar(12)	YES		Units of the concentration values. Note that units may be inventory units too (refer to Vector_Constituent.Concentration).
			Basis	varchar(20)	YES		The basis or data source of the inventory calculation. Multiple basis (Tank_Inventory table only) are concatenated using the "/" character (e.g. S/E/T (241-BX-112)).
			Comment	varchar(80)	YES		Comment
			Inv_Formula	varchar(255)	YES		The formula used in calculating the inventory. Currently only used for Total Alpha adjustments (only calculation done at the Tank_Inventory level; all others done at the Tank_Waste_Inventory level)
			Inv_Calc	varchar(255)	YES		The data (numbers) used in the formula to calculate the inventory.
			Inv_Routine	varchar(40)	YES		The name of the stored procedure or trigger which generated the inventory. Useful in tracing the code.
			VC_ID	int	YES		Unique Identifier of the Vector_Constituent record. This is an identity column set by the SQL Server upon insert.
			Last_Updated_By	varchar(30)	YES		The SQLServer user name of the person who performed the most recent update
			Update_Reason	varchar(40)	YES		If a trigger updates or inserts this record, the reason for the update (e.g. Density changed).
			Published_On	datetime	YES	(getdate())	Date and time that this record was copied over from either the TCD database or the BBIM database (update or insert).
			Density	float	YES		The density, in g/ml of the vector. A density is needed to calculate inventories when vector constituents are recorded as concentrations. If vector constituents values are recorded in units of inventory (kg or Ci), then Density is not needed.
			Sample_Date	datetime	YES		For vectors with a type of "Sample", Sample_Date contains the date that the sample was taken. Sometimes this data is not readily available.
			Inventory_RSD	float	YES		Relative Standard Deviation for Inventory, in units of percentage. Equal to $\text{SQRT}(\text{Density.RSD}^2 + \text{Vol.RSD}^2 + \text{Conc.RSD}^2)$
			Qualifier	varchar(3)	YES		Qualifies the concentration of the constituent.
			Concentration_RSD	float	YES		Relative Standard Deviation for Concentration, in units of percentage. Equal to $\text{Std_Dev} / \text{Concentration}$.

Multiplier

float

YES

A numeric value, which is used to modify the concentration values of a vector when inventories are calculated. For example, a sample is taken from tank A and a vector stored in the database for this sample. Next, the waste is pumped into an evaporator and the volume reduced by one half. The waste is then pumped into tank B. The waste in tank B is now twice as concentrated as it was when the sample was taken in tank A. The sample vector can be used to represent the waste in tank B if a multiplier of 2 is factored in.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
I1_TWI	TWT_ID	ASC	Yes	CLUSTERED
I1_TWI	Con_ID	ASC	Yes	CLUSTERED
I2_TWI	Con_ID	ASC		NONCLUSTERED
TWI_PK	TI_ID	ASC	Yes	NONCLUSTERED

dbo.Tank_Waste_Inventory_Hist

Snapshot of the Tank Waste Inventory data over time.

I	F	P	Column name	Data type	Nulls	Default	Description
			TWIH_ID	int	NO		Unique Identifier for the Tank_Waste_Inventory_Hist table.
			TWCH_ID	int	NO		Unique Identifier for the Tank_Waste_Component_Hist table.
			TI_ID	int	NO		Computer generated unique identifier for the tank waste inventory record.
			TWT_ID	int	YES		A computer generated unique identifier for the Tank_Waste_Component record. (TWT_ID is historicalthe table name was changed from Tank_Waste_Type to Tank_Waste_Component).
			Con_ID	varchar(15)	YES		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
			Inventory	float	YES		The inventory value calculated for this constituent in the tank waste component.
			Units	varchar(12)	YES		Concentration units of the reported value or the reporting limit.
			Concentration	float	YES		The concentration value for this constituent in the tank waste component which was used to calculate the Inventory value. Note that this is the Adjusted Concentration, e.g. it has had the Multiplier applied to it, and it will be different than the unadjusted concentration that is shown in the Inventory Calculation field.
			Units_of_Concentration	varchar(12)	YES		Units of the concentration values. Note that units may be inventory units too (refer to Vector_Constituent.Concentration).
			Basis	varchar(20)	YES		The basis or data source of the inventory calculation. Multiple basis (Tank_Inventory table only) are concatenated using the "/" character (e.g. S/E/T (241-BX-112)).
			Comment	varchar(80)	YES		Comment
			Inv_Formula	varchar(255)	YES		The formula used in calculating the inventory. Currently only used for Total Alpha adjustments (only calculation done at the Tank_Inventory level; all others done at the Tank_Waste_Inventory level)
			Inv_Calc	varchar(255)	YES		The data (numbers) used in the formula to calculate the inventory.
			Inv_Routine	varchar(40)	YES		The name of the stored procedure or trigger which generated the inventory. Useful in tracing the code.
			VC_ID	int	YES		Unique Identifier of the Vector_Constituent record. This is an identity column set by the SQL Server upon insert.
			Last_Updated_By	varchar(30)	YES		The SQLServer user name of the person who performed the most recent update
			Update_Reason	varchar(40)	YES		If a trigger updates or inserts this record, the reason for the update (e.g. Density changed).
			Published_On	datetime	YES		Date and time that this record was copied over from either the TCD database or the BBIM database (update or insert).
			Density	float	YES		The density, in g/ml of the vector. A density is needed to calculate inventories when vector constituents are recorded as concentrations. If vector constituents values are recorded in units of inventory (kg or Ci), then Density is not needed.
			Sample_Date	datetime	YES		For vectors with a type of "Sample", Sample_Date contains the date that the sample was taken. Sometimes this data is not readily available.
			Inventory_RSD	float	YES		Relative Standard Deviation for Inventory, in units of percentage. Equal to $\text{SQRT}(\text{Density.RSD}^2 + \text{Vol.RSD}^2 + \text{Conc.RSD}^2)$
			Qualifier	varchar(3)	YES		Qualifies the concentration of the consti tuent.
			Concentration_RSD	float	YES		Relative Standard Deviation for Concentration, in units of percentage. Equal to $\text{Std_Dev} / \text{Concentration}$.

Multiplier

float

YES

A numeric value, which is used to modify the concentration values of a vector when inventories are calculated. For example, a sample is taken from tank A and a vector stored in the database for this sample. Next, the waste is pumped into an evaporator and the volume reduced by one half. The waste is then pumped into tank B. The waste in tank B is now twice as concentrated as it was when the sample was taken in tank A. The sample vector can be used to represent the waste in tank B if a multiplier of 2 is factored in.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_Tank_Waste_Inventory_Hist	TWT_ID	ASC		CLUSTERED
TWIH_ID_PK	TWIH_ID	ASC	Yes	NONCLUSTERED

Foreign keys

<u>Constraint name</u>	<u>Column name</u>	<u>Reference</u>	<u>Description</u>
FK__Tank_Wast__TWCH___7A F13DF7	TWCH_ID	TWCH_PK (dbo.Tank_Waste_Component_Hist)	

dbo.ULD_dose_conversion_factors

All ULD calculations are now performed offline in the client's Excel spreadsheets. Therefore, this table is no longer used for any calculations, but is of interest if persons request how the older ULD data was calculated:

For Onsite ULD calculations, each radionuclide ULD is calculated using the formula: $Sv/L = (Ci) * [ICRP-68 (Sv/Bq)] * (3.7E10 Bq/Ci) / [(kL) * (1E3 L/kL)]$.

For Offsite ULD calculations, each radionuclide ULD is calculated using the formula: $Sv/L = (Ci) * [ICRP-71 (Sv/Bq)] * (3.7E10 Bq/Ci) / [(kL) * (1E3 L/kL)]$.

I	F	P	Column name	Data type	Nulls	Default	Description
			constituent_name	varchar(15)	NO		Name of the constituent.
			ICRP_71a_DCF_SvBq	float	YES		ICRP-71a DCF (Sv/Bq)
			Abs_Type	varchar(10)	YES		Abs type
			ICRP_71a_DCF_source	varchar(20)	YES		ICRP-71a DCF source
			ICRP_68a_DCF_SvBq	float	YES		ICRP-68a DCF (Sv/Bq)
			ICRP_68a_DCF_class	varchar(10)	YES		ICRP-68a DCF class
			ICRP_68a_DCF_forced_class	varchar(10)	YES		ICRP-68a DCF forced class
			ICRP_68a_DCF_source	varchar(10)	YES		ICRP-68a DCF source

Indexes

	Index name	Column name	Sort direction	Is unique	Index type
	PK__ULD_dose_convers__2E70E1FD	constituent_name	ASC	Yes	CLUSTERED

dbo.Unit_Liter_Dose

Radiological source terms to be used in Tank Farm safety analyses. Radiological source terms are calculated using the unit liter dose (ULD) methodology described in RPP-5924, Radiological Source Terms for Tank Farms Safety Analyses. Source term results are provided for all tanks with published inventories. The source term values include onsite ULD, offsite ULD, 137Cs concentration, and 90Sr concentration of each tank waste layer. The ULD is the dose obtained if an individual inhales one liter of waste. Onsite ULD values for workers differ somewhat from offsite ULD values for the public because of differences in the assumed physical and exposure characteristics for these two receptor groups. Radiological exposure calculation parameters such as release rates or atmospheric dispersion and leak path factors are not included in the source term values.

I F P	Column name	Data type	Nulls	Default	Description
	ULD_ID	int	NO		Unique record identifier.
	Waste_Site_ID	varchar(15)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	Tank_Class	varchar(10)	NO		Class of Tank, either Single-Shell Tank (SST), Double-Shell Tank (DST), or Double Contained Receiver Tank (DCRT)
	Waste_Phase	varchar(50)	NO		One of: Saltcake (Liquid & Solid) Saltcake Interstitial Liquid Saltcake Solid Sludge (Liquid & Solid) Sludge Interstitial Liquid Sludge Solid Supernatant
	Waste_Type	varchar(60)	NO		Retained Gas is excluded. Abbreviation for the name of the waste. Some Examples: <table border=0><tr><td>AW</td><td>CC</td><td>CP</td></tr><tr><td>DC</td><td>DN</td><td>IL</td></tr><tr><td>SLUDGE</td><td>SOLID_CAKE</td><td>SUPER</td></tr></table>
	Volume	float	NO		The volume, usually in kL, of this tank waste component. If a tank waste component with a waste phase of "Total" exists, then the sum of the volumes for any other waste components for this tank should equal the "Total" volume.
	Units	varchar(4)	NO		Concentration units of the reported value or the reporting limit.
	Onsite_ULD	float	NO		The sum of the radionuclide ULDs within the waste component. Each radionuclide ULD is calculated using the formula: $Sv/L = (Ci) * [ICRP-68 (Sv/Bq)] * (3.7E10 Bq/Ci) / [(kL) * (1E3 L/kL)]$
	Offsite_ULD	float	NO		The sum of the radionuclide ULDs within the waste component. Each radionuclide ULD is calculated using the formula: $Sv/L = (Ci) * [ICRP-71 (Sv/Bq)] * (3.7E10 Bq/Ci) / [(kL) * (1E3 L/kL)]$
	Cs_137_Bq_per_L	float	NO		The concentration of 137Cs in the waste component. The 137Cs concentration is reported as of the decay date
	Sr_90_Bq_per_L	float	NO		The concentration of 90Sr in the waste component. The 90Sr concentration is reported as of the decay date.
	ULD_Run_Date	datetime	NO		Date that this report was created. Published BBI data, as of this date, were used to calculate the radiological source terms.
	Decay_To_Date	datetime	NO		Date to which the values are decayed

Foreign keys

Constraint name	Column name	Reference	Description
fk_ULD_Waste_Site_ID	Waste_Site_ID	Tank_PK (dbo.Tank)	

dbo.Uploaded_Summary_Data

Data uploaded from the users Unit Sum of Fractions (USOF) spreadsheet. Toxicological source terms to be used in Tank Farm safety analyses. Toxicological source terms are calculated using the sum-of-fractions methodology described in RPP-30604, Tank Farms Safety Analyses Chemical Source Term Methodology. Source term results are provided for all tanks with published inventories. The source term values are the sum of the ratios of the waste concentrations as they exist in the tanks to the PAC airborne concentration limits. Toxicological exposure calculation parameters such as release rates or atmospheric dispersion and leak path factors are not included in the source term values.

Definitions:

USOF Unit sum-of-fractions. The USOF (toxicological source term values) are the sum of the ratios of the waste concentrations as they exist in the tanks to the PAC airborne concentration limits.

TEEL Temporary Emergency Exposure Limit

PAC Protective Action Criteria (in hierarchical order, Acute Exposure Guideline Levels, Emergency Response Planning Guidelines or TEELs depending on data availability)

I	F	P	Column name	Data type	Nulls	Default	Description
			Waste_Site_ID	varchar(15)	NO		Tank name used to identify the specific Hanford Site underground storage tank
			Waste_Phase	varchar(50)	NO		One of: Saltcake (Liquid & Solid) Saltcake Interstitial Liquid Saltcake Solid Sludge (Liquid & Solid) Sludge Interstitial Liquid Sludge Solid Supernatant
			Waste_Type	varchar(60)	NO		Retained Gas is excluded. Combined with Waste Phase, defines a layer or waste component in the tank
			SOF_Effective_Date	smalldatetime	NO		Published BBI data were extracted from the BBI database on this date and then used to calculate the USOF values
			Ratio_Ordinal	char(1)	NO		An integer from 0 through 3 that indicates whether the Ratio_Value is a TEEL-0, PAC-1, PAC-2 or PAC-3 value.
			Ratio_Value	float	NO		The USOF TEEL-0, PAC-1, PAC-2 or PAC-3 value.
			Volume_kL	float	YES		Volume of the waste in this waste component or layer, measured in kL
			Tank_Type	varchar(3)	NO		Single shell or double shell tank construction (SST or DST)
			Most_Recent	bit	NO		Flag indicating whether this data point is the most recent for a given Waste Site ID, Phase and Type. Used for speeding up queries.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
pk_Uploaded_Summary_Data	Waste_Site_ID	ASC	Yes	CLUSTERED
pk_Uploaded_Summary_Data	Waste_Phase	ASC	Yes	CLUSTERED
pk_Uploaded_Summary_Data	Waste_Type	ASC	Yes	CLUSTERED
pk_Uploaded_Summary_Data	SOF_Effective_Date	ASC	Yes	CLUSTERED
pk_Uploaded_Summary_Data	Ratio_Ordinal	ASC	Yes	CLUSTERED

dbo.Wash_Leach

Wash and Leach factors currently used in the Hanford Tank Waste Operations Simulator (HTWOS) model by waste tank and analyte. This set of wash and leach factors is based on the tank-specific wash factors reported in HNF-3157, on global leach factors reported in PNNL-11646, on data from (or derived from) laboratory tests performed on waste samples from tanks identified as early feed sources for the Waste Treatment Plant, or from engineering judgment. The global leach factors were developed as part of the effort to satisfy Tri-Party Agreement milestone M-50-03 (submitted in September 1997, RMIS accession number D197257414).

Used by the WashLeach ASP application.

I F P	Column name	Data type	Nulls	Default	Description
	waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	synonym_name	varchar(40)	NO		Standard best-basis analyte expressed as chemical symbol (reported without charge designation per the best-basis inventory convention).
	factor_type	varchar(20)	NO		Either Wash or Leach.
	factor	float	YES		A wash factor is the fraction of a waste constituent removed from the solid phase of a waste to the liquid phase by washing with water.
	reference	varchar(255)	YES		Reference document from which data was gathered.
	custodian	varchar(50)	YES		Not used: either NULL or blank.
	load_date_time	datetime	NO	(getdate())	The date and time that the record was saved to the database.
	modified_date_time	datetime	YES		Date that the record was last modified.
	WashLeachID	int	NO		Unique record identifier.

dbo.Wash_Leach_Constituents

The list of constituents that are reported by the WashLeach ASP application.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			synonym_name	varchar(40)	NO		Standard best-basis analyte expressed as chemical symbol (reported without charge designation per the best-basis inventory convention).

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK_Wash_Leach_Constituents	synonym_name	ASC	Yes	CLUSTERED

dbo.Wash_Leach_tanks

The list of tanks that are reported by the WashLeach ASP application.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK_Wash_Leach_tanks	waste_site_id	ASC	Yes	CLUSTERED

dbo.Waste_Phase_Mapping

A correlation table which links each (if possible) Tank_Waste_Component record with a Standard_Waste_Phase record.

I F P	Column name	Data type	Nulls	Default	Description
 	TWT_ID	int	NO		A computer generated unique identifier for the Tank_Waste_Component record. (TWT_ID is historicalthe table name was changed from Tank_Waste_Type to Tank_Waste_Component).
	Std_Waste_Phase	varchar(50)	YES		Baselined tanks have waste that is described by these phases: supernatant, sludge solid, sludge liquid, salt cake solid, salt cake liquid and retained gas.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
PK_Waste_Phase_Mapping	TWT_ID	ASC	Yes	NONCLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK_WPM_TWT_ID	TWT_ID	TWC_PK (dbo.Tank_Waste_Component)	
FK_WPM_Std_Waste_Phase	Std_Waste_Phase	PK_Standrd_Waste_Phase (dbo.Standard_Waste_Phase)	

Stored procedures

dbo.p_BBIM_BB_Pub_Inv_Diff

Created: 28 Apr 2009 Last updated: 28 Apr 2009

Get all bbim inventory that has different inventory in the bb_published database. Called from the BBI ASP.NET application (General Reports item).

dbo.p_Calc_Detail

Created: 29 Apr 2009 Last updated: 08 Jul 2009

TWINS Data Query: This report contains the detailed calculations showing the actual concentration, density, volume and unit adjustment values used to derive inventory values by analyte by BBI standard waste phases. (NOTE: The Retained Gas standard waste phase is not reported here because the inventory is zero.)

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@ip_addr	varchar	IN	

dbo.p_Calc_Detail_Supp

Created: 27 Nov 2006 Last updated: 27 Nov 2006

TWINS Data Query: This report contains the detailed calculations showing the actual concentration, density, volume and unit adjustment values used to derive inventory values by analyte by BBI standard waste phases. The Total tank inventory for supplemental analytes may be biased low, as analytical information may not be available for all waste phases in a tank. (NOTE: The Retained Gas standard waste phase is not reported here because the inventory is zero.)

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@ip_addr	varchar	IN	

dbo.p_Consolidated_Pub_Date_List

Created: 26 Jan 2007 Last updated: 26 Jan 2007

Get a list of Publication dates, however when the dates are in the same proximity, then choose only the latest date in this group.

dbo.p_Constr_Waste_Model

Created: 11 Dec 2006 Last updated: 11 Dec 2006

TWINS Data Query: This query provides a comparison of latest Best Basis Inventory information to the information used in the Conservative Waste Model used in the Criticality Safety Evaluation Report for Tank Farms. See: <http://twins/dictionary/ConservativeWasteModelRptDoc.htm> to see how the data is calculated.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@ip_addr	varchar	IN	

dbo.p_Criticality_Report

Created: 14 Sep 2001 Last updated: 16 Jul 2010

TWINS Query. No execute permission exists. Modified the procedure to exit immediately on 7/16/2010.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@ip_addr	varchar	IN	

dbo.p_Criticality_Report_prod

Created: 28 Aug 2004 Last updated: 28 Aug 2004

Called by the TWINS NCS Data query.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@ip_addr	varchar	IN	

dbo.p_Get_Calc_Detail_For

Created: 12 Apr 2006 Last updated: 12 Apr 2006

Get the calc detail data from the history tables for a specific point in tiime for a specific tank.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@thisDate	datetime	IN	
@wasteSiteID	varchar	IN	

 **dbo.p_GetPubDateSorted**

Created: 28 Jan 2003 Last updated: 28 Jan 2003

Get the HTWOS report dates, sorted by most recent date first.

 **dbo.p_htwo**

Created: 17 Jul 2002 Last updated: 17 Jul 2002

Updates the dbo.htwo and dbo.htwo_report_dates tables. Executed by the Populate HTWOS Report DTS package.

 **dbo.p_Inventory_Not_Published**

Created: 28 Apr 2009 Last updated: 28 Apr 2009

Get all bbim inventory that is not in the bb_published database

dbo.p_populate_caustic_report

Created: 06 Mar 2002 Last updated: 06 Mar 2002

TWINS query generates the caustic report.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@ip_addr	varchar	IN	

 **dbo.p_Publish_Sups**

Created: 14 Mar 2002 Last updated: 14 Mar 2002

Stored procedure used specifically for Change Request 356. Was run once to copy supplemental analytes from the BBIM over to the BBI.

 **dbo.p_SOF_Effective_Dates**

Created: 26 Mar 2010 Last updated: 26 Mar 2010

Retrieve USOF Run Date parameter values for the TWINS USOF query

dbo.p_SOF_Input

Created: 20 May 2009 Last updated: 12 Mar 2010

Used by the BBI ASP.NET application to generate the input file for the SOF external spreadsheet (as per CR 604).

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@ip_addr	varchar	IN	

dbo.p_TI_Hist_Diff

Created: 05 Aug 2003 Last updated: 05 Aug 2003

Used by the BBI ASP.NET report on Tank Inventory History Differences.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@Waste_Site_ID	varchar	IN	

dbo.p_TWINS_SOF

Created: 26 Mar 2010 Last updated: 26 Mar 2010

Retrieve the SOF TEEL values for TWINS

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@ip_addr	varchar	IN	

dbo.p_TWINS_ULD

Created: 30 Jan 2007 Last updated: 15 Sep 2009

TWINS Query: Radiological source terms to be used in Tank Farm safety analyses. Radiological source terms are calculated using the unit liter dose (ULD) methodology described in RPP-5924, Radiological Source Terms for Tank Farms Safety Analyses. Source term results are provided for all tanks with published inventories. The source term values include onsite ULD, offsite ULD, 137Cs concentration, and 90Sr concentration of each tank waste layer. The ULD is the dose obtained if an individual inhales one liter of waste. Onsite ULD values for workers differ somewhat from offsite ULD values for the public because of differences in the assumed physical and exposure characteristics for these two receptor groups. Radiological exposure calculation parameters such as release rates or atmospheric dispersion and leak path factors are not included in the source term values.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@ip_addr	varchar	IN	

dbo.p_ULD_Run_Date_List

Created: 15 Sep 2009 Last updated: 15 Sep 2009

Gets a list of unique ULD Run Dates, ignoring the time portion. Used in the TWINS ULD query.

 **dbo.p_Update_ULD_Pub_Date**

Created: 01 Feb 2007 Last updated: 01 Feb 2007

Sets the ULD_Publication_Date to the current date for each Unit_Liter_Dose record that has no date.

dbo.p_Upload_Summary_Data

Created: 26 Mar 2010 Last updated: 26 Mar 2010

Obsolete. Replaced by the p_Upload_USOF_Data stored procedure. Exec privilege revoked on July 16/2010.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@Source	varchar	IN	
@Effective_Date	varchar	IN	
@Ratio_Ordinal	varchar	IN	
@Ratio_Value	float	IN	

dbo.p_Upload_ULD_Data

Created: 13 Aug 2009 Last updated: 15 Sep 2009

Upload the Unit Liter Dose data from the user supplied spreadsheet.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@Waste_Site_ID	varchar	IN	
@Tank_Class	varchar	IN	
@Waste_Phase	varchar	IN	
@Waste_Type	varchar	IN	
@Volume	float	IN	
@Units	varchar	IN	
@Onsite_ULD	float	IN	
@Offsite_ULD	float	IN	
@Cs_137_Bq_per_L	float	IN	
@Sr_90_Bq_per_L	float	IN	
@ULD_Run_Date	datetime	IN	
@Decay_To_Date	datetime	IN	

dbo.p_Upload_USOF_Data

Created: 16 Jul 2010 Last updated: 16 Jul 2010

Upload the USOF data from the user spreadsheet into the bb_published.dbo.Uploaded_Summary_Data table.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@Waste_Site_ID	varchar	IN	
@Waste_Phase	varchar	IN	
@Waste_Type	varchar	IN	
@Effective_Date	varchar	IN	
@Volume	float	IN	
@Tank_Type	varchar	IN	
@TEEL0	float	IN	
@PAC1	float	IN	
@PAC2	float	IN	
@PAC3	float	IN	

dbo.p_Wash_Leach

Created: 25 Apr 2003 Last updated: 25 Apr 2003

TWINS Query: Wash and Leach factors currently used in the Hanford Tank Waste Operations Simulator (HTWOS) model by waste tank and analyte. This set of wash and leach factors is based on the tank-specific wash factors reported in HNF-3157, on global leach factors reported in PNNL-11646, on data from (or derived from) laboratory tests performed on waste samples from tanks identified as early feed sources for the Waste Treatment Plant, or from engineering judgment. The global leach factors were developed as part of the effort to satisfy Tri-Party Agreement milestone M-50-03 (submitted in September 1997, RMIS accession number D197257414).

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@ip_addr	varchar	IN	

dbo.p_Waste_Type_Volume

Created: 01 May 2006 Last updated: 01 May 2006

TWINS Query: Volume of each Best Basis waste-type by Tank.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@ip_addr	varchar	IN	

dbo.p1_Upload_USOF_Data

Created: 16 Jul 2010 Last updated: 16 Jul 2010

Upload the USOF data from the user spreadsheet into the bb_published.dbo.Uploaded_Summary_Data table. Called by the p_Upload_USOF_Data for each row in the spreadsheet.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@Waste_Site_ID	varchar	IN	
@Waste_Phase	varchar	IN	
@Waste_Type	varchar	IN	
@SOF_Effective_Date	smalldatetime	IN	
@Volume	float	IN	
@Tank_Type	varchar	IN	
@Ratio_Ordinal	varchar	IN	
@Ratio_Value	float	IN	

dbo.sp_ColumnHeaders

Created: 19 Nov 2009 Last updated: 19 Nov 2009

Utility procedure that will extract the column headers for a given table. Useful when copying data from a query to Excel where the data query has no column headings.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@TableName	varchar	IN	

User defined functions

 dbo.BBIFinalDoc

Created: 18 Dec 2009 Last updated: 18 Dec 2009

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	varchar	OUT	
@att	varchar	IN	

dbo.FGE

Created: 11 Dec 2006 Last updated: 11 Dec 2006

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	float	OUT	
@TWT_ID	int	IN	

BBIM Database Documentation

Server name: twins

Database name: BBIM

Documentation date: **7/22/2010**

Database size: 1035.69 MB

Database description:

Table summary

	Name	Data size	Index size	Row Count
	dbo.AverageDensityCalc	8 KB	16 KB	5
	dbo.CM_Control	8 KB	16 KB	1
	dbo.Constituent	32 KB	64 KB	206
	dbo.Constituent_Dependents	8 KB	16 KB	5
	dbo.Constituent_Type	8 KB	32 KB	4
	dbo.Debug_Setting	16 KB	40 KB	204
	dbo.Decay_Chain	8 KB	16 KB	4
	dbo.Densities	104 KB	16 KB	1394
	dbo.Density_Rel_Std_Dev	8 KB	8 KB	8
	dbo.Dependent_Basis	320 KB	16 KB	2045
	dbo.Inventory_Changes	24 KB	16 KB	101
	dbo.Max_Liquid_Concentration	24 KB	16 KB	354
	dbo.Phase_Sigma	8 KB	8 KB	8
	dbo.Representative_Vectors	536 KB	192 KB	1770
	dbo.Representative_Vectors_Hist	424 KB	40 KB	1445
	dbo.RPD4TanksInWorkUnit	520 KB	24 KB	1377
	dbo.Save_Vector_Locked_Status	8 KB	8 KB	207
	dbo.Standard_Decay_To_Date	8 KB	16 KB	1
	dbo.Standard_Waste_Phase	16 KB	24 KB	7
	dbo.Std_Unit_Conversion	8 KB	24 KB	19
	dbo.Tank	48 KB	48 KB	211
	dbo.Tank_Inventory	1944 KB	48 KB	16183
	dbo.Tank_Waste_Component	152 KB	24 KB	557
	dbo.Tank_Waste_Component_Hist	272 KB	64 KB	1014
	dbo.Tank_Waste_Inventory	19976 KB	4496 KB	45567
	dbo.Tank_Waste_Inventory_Hist	35896 KB	64 KB	84109
	dbo.U_and_Pu_Dependents	16 KB	16 KB	34
	dbo.Units_Of_Measure	8 KB	32 KB	21
	dbo.Vector	744 KB	648 KB	2826
	dbo.Vector_Constituent	20096 KB	2528 KB	94685
	dbo.Vector_Constituent_Excel_Data	88 KB	8 KB	600
	dbo.Waste_Phase	8 KB	16 KB	9
	dbo.Work_Unit	32 KB	24 KB	32
	dbo.Work_Unit_Archive	1936 KB	16 KB	8400
	dbo.Work_Unit_Member	8 KB	56 KB	29
	dbo.Work_Unit_Member_Archive	584 KB	24 KB	8585

36 table(s)

TABLES

dbo.AverageDensityCalc

Used by the Representative_Vectors insert trigger in updating the Average_Density of affected Tank_Waste_Component records.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			TWT_ID	int	NO		A computer generated unique identifier for the Tank_Waste_Component record. (TWT_ID is historicalthe table name was changed from Tank_Waste_Type to Tank_Waste_Component).
			V_ID	int	NO		Computer generated unique identifier for the vector.
			Density	float	NO		The density, in g/ml of the vector. A density is needed to calculate inventories when vector constituents are recorded as concentrations. If vector constituents values are recorded in units of inventory (kg or Ci), then Density is not needed.
			User_Name	varchar(20)	NO		The database login name of the user who inserted this record.

dbo.CM_Control

A single record table which is used to "inform" the triggers whether or not configuration management is turned on or off.

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	CM_On	int	YES	((-1))	When CM_On = 1, then only the Tanks are under configuration management (eg, data relating to a tank can only be modified if the tank exists in a work_unit in an unlocked state). When CM_On = -1, then both Tanks and Vectors are under configuration management control. Setting CM_On to 0 turns OFF configuration management.

Triggers

Trigger name:	dbo.CM_Control_DTrig	Created on:	27 Apr 2000
Trigger type:	DELETE	Trigger active:	Yes
Trigger name:	dbo.CM_Control_IUTrig	Created on:	27 Apr 2000
Trigger type:	INSERT, UPDATE	Trigger active:	Yes

dbo.Constituent

Chemicals and radionuclides found in Hanford HLW tanks. There are currently 25 major chemicals, 5 minor chemicals, and 46 radionuclides that are reported in the best basis documents. These are the constituents that will have the largest effect on the waste retrieval and processing (vitrification).

I F P	Column name	Data type	Nulls	Default	Description
	Con_ID	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
	Constituent_Name	varchar(50)	NO		Name of the constituent.
	Type	varchar(12)	NO		A constituent is either a Chemical, Radionuclide or Water.
	Molecular_Weight	float	NO		The molecular weight of the constituent is needed for converting M (moles/L) to inventory units (kg).
	Sort_Order	int	NO		Used for sorting the records in this table or view.
	Best_Basis_Constituent	char(1)	YES		Flag indicating whether this constituent is one of the best basis" 30 chemicals or 46 radionuclides.<table border=0><tr><td>Y</td><td> : Best Basis Constituent</td></tr><tr><td>N</td><td> : Supplemental Analyte</td></tr><tr><td>NULL</td><td> : Neither</td></tr></table>
	Primary_Valence	float	YES		Valence refers to the outer electronic configuration, or, the highest occupied electron energy level of the constituent. Primary_Valence is either the valence (whole number) of the constituent in its prevalent state, or, a weighted average of its valences. Primary_Valence is used in calculating the amount of hydroxide that must exist in a tank in order to get a net charge of zero. The Primary_Valence of radionuclides is equal to NULL in the BBIM since their concentrations are much lower than the non-radionuclides, hence their input to the charge balance is insignificant.
	Half_Life_Seconds	float	YES		The number of seconds that it takes for one half of the constituent (radionuclide) to decay into daughter products.
	Lambda	float	YES		Lambda is used in radionuclide decay factor calculations, and is equal to log (2)/Half_Life_Seconds*86400. The units of Lambda are in "days-1".
	Alpha_Emitter	smallint	YES		Flag indicating whether or not this constituent is an alpha emitting isotope. True is stored as -1 (the MS Access equivalent to true when attached to a checkbox control).
	Specific_Activity	float	YES		Of the radioisotope, measured in Ci/g. Used in the conversion of ug/g to uCi/g, and ug/mL to uCi/mL.
	Thermal_Neutron_Absorption	float	YES		The thermal neutron cross-section measures the probability of a neutron being absorbed by an atom. It is measured in barnes.
	Min_Subcritical_Ratio	float	YES		The Minimum SubCritical Ratio is the ratio of neutron absorbing atoms to fissile atoms necessary to ensure that waste remains subcritical.
	Critical_Absorber	char(1)	NO		Flag indicating whether this constituent is used in the <i>subcritical ratio for credited absorbers</i> calculation. The subcritical ratio for an absorber is the ratio of the mass of an absorber present in the waste to the mass of fissile material in the waste divided by the minimum subcritical ratio for that absorber. The 5 absorbers identified in the Criticality Specification for control of incoming waste into Tank Farms are called credited (or critical) absorbers.
	SOF_Constituent	char(1)	YES		Flag indicating whether this constituent is a member of the Sum of Fractions.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
Constituent_PK	Con_ID	ASC	Yes	NONCLUSTERED
i_con_id	Con_ID	ASC	Yes	NONCLUSTERED
i_con_name	Constituent_Name	ASC		NONCLUSTERED

dbo.Constituent_Dependents

Certain radionuclide inventories can be calculated based on another radionuclide inventory. The formula for calculating a dependent inventory is $isDependent\ Inventory = Source\ Inventory * Factor$. The dependents are

90Sr --> 90Y
137Cs --> 137mBa
243/244Cm --> 243Cm
243/244Cm --> 244Cm
90Sr --> Sr

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	Con_ID1	varchar(15)	NO		The Constituent identifier of the inventory that may be calculated.
	Con_ID2	varchar(15)	NO		The Constituent identifier of the inventory that may be used to calculate the inventory for Con_ID1.
	Factor	float	NO		A number used in the multiplication of the inventory of Con_ID2 to obtain the inventory for Con_ID1.

dbo.Constituent_Type

The type of constituent is either Chemical, Density, Radionuclide or Water.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			Type	varchar(12)	NO		Chemical, Radionuclide or Water.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
aaaaaConstituent_Type_PK	Type	ASC	Yes	NONCLUSTERED

Triggers

Trigger name:	dbo.Constituent_Type_DTrig	Created on:	27 Apr 2000
Trigger type:	DELETE	Trigger active:	Yes
Trigger name:	dbo.Constituent_Type_ITrig	Created on:	27 Apr 2000
Trigger type:	INSERT	Trigger active:	Yes
Trigger name:	dbo.Constituent_Type_UTrig	Created on:	27 Apr 2000
Trigger type:	UPDATE	Trigger active:	Yes

dbo.Debug_Setting

BBIM triggers and stored procedures "select" the on/off setting from the table Debug_Setting which enables them to turn on or off their debugging messages.

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	DS_ID	int	NO		Identity field, generated by the database.
	Type	varchar(4)	NO		The type of database object that can be debugged.
	Name	varchar(60)	YES		Name of the trigger or procedure in the BBIM database.
	Setting	smallint	NO		0 or -1, with 0 representing "turn OFF the debugging messages" and -1 representing "turn ON the debugging messages". -1 is used for "ON" because this corresponds to the MS Access checkbox in a form being "Checked".

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
_WA_Sys_Name_21B6055D	Name	ASC		NONCLUSTERED
U_Debug_Setting	Type	ASC	Yes	NONCLUSTERED
U_Debug_Setting	Name	ASC	Yes	NONCLUSTERED

Check constraints

<u>Check name</u>	<u>Column name</u>	<u>Expression</u>
CK_Type	Type	([Type] = 'PROC' or [Type] = 'TRIG')

dbo.Decay_Chain

Used by the *p_Decay_And_Distribute* procedure to calculate 2nd Order decay of radionuclides. These are:

93Zr --> 93mNb
231Pa --> 227Ac
232Th --> 228Ra
241Pu --> 241Am

I F P	Column name	Data type	Nulls	Default	Description
	Parent_Con_ID	varchar(15)	NO		The constituent identifier of the radionuclide which will undergo decay.
	Daughter_Con_ID	varchar(15)	NO		The constituent identifier of the Parent_Con_ID's daughter product.
	Percent_Decayed_Div100	float	NO		If a radionuclide decays to multiple daughters, then a percentage decays to one daughter product, and a percentage decays to another. The sum of the percentages must equal 100. Percent_Decayed_Div100 is the percent decayed to this particular daughter product from this particular parent, divided by 100.

Triggers

Trigger name:	dbo.Decay_Chain_DTrig	Created on:	26 Mar 2002
Trigger type:	DELETE	Trigger active:	Yes
Trigger name:	dbo.Decay_Chain_ITrig	Created on:	26 Mar 2002
Trigger type:	INSERT	Trigger active:	Yes
Trigger name:	dbo.Decay_Chain_UTrig	Created on:	26 Mar 2002
Trigger type:	UPDATE	Trigger active:	Yes

dbo.Densities

Densities contains Vector densities plus densities for the virtual "average vectors". This denormalized table is used to increase performance.

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	TWT_ID	int	NO		A computer generated unique identifier for the Tank_Waste_Component record. (TWT_ID is historical the table name was changed from Tank_Waste_Type to Tank_Waste_Component).
	Con_ID	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
	Type	varchar(16)	NO		The category or type of something.
	Name	varchar(60)	NO		Names of wastes found in the tanks. Some examples: <ul style="list-style-type: none"> Neutralized Current Acid Waste Complexant Concentrate Concentrated Phosphate Waste Dilute Complexed Dilute Non-Complexed SST interstitial liquid in SL/SC PUREX NCRW Sludge (TRU) Double-shell Slurry Feed PFP TRU Solids
	Density	float	YES		The density, in g/ml of the vector. A density is needed to calculate inventories when vector constituents are recorded as concentrations. If vector constituents values are recorded in units of inventory (kg or Ci), then Density is not needed.

dbo.Density_Rel_Std_Dev

Relative Standard Deviation for Density. Grouped by Shell Type {SST | DST} and Standard Waste Phase.

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	Shell_Type	char(10)	NO		The type of shell a waste tank has.
	Std_Waste_Phase	varchar(50)	NO		Baselined tanks have waste that is described by these phases: supernatant, sludge solid, sludge liquid, salt cake solid, salt cake liquid and retained gas.
	Density_RSD	float	NO		Relative Standard Deviation for Density, in units of percentage. Equal to Std_Dev / Density.

dbo.Inventory_Changes

Used by the *p_pop_Inv_Changes* stored procedure, which is called from the *configmgt.mdb* application. Holds information pertaining to the inventory changes that would occur if a tank were to be published.

I	F	P	Column name	Data type	Nulls	Default	Description
			Waste_Site_ID	varchar(15)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
			Con_ID	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
			Con_Name	varchar(15)	NO		Name of the constituent used for reporting purposes. Typically, the name comes from the Chemical Abstract Services (CAS) listing.
			RPD	float	YES		Relative percent difference
			Update_Mode	varchar(7)	YES		Can be one of three states: 'Locked' which means that this record will not be overwritten when data is 'published' from the BBIM (e.g. the database triggers and stored procedures will NOT overwrite this record); 'Force Update' which means that when data is 'published' from the BBIM, the triggers and stored procedures should overwrite this record; 'Default' which means that this record will only get overwritten by BBIM 'published' data if their RPD is < 1.
			Current_Inventory	float	YES		The Published Inventory.
			Pub_Inv4Display	varchar(12)	YES		Published Value of the Inventory for Display. Numeric value converted to a character string in a display format.
			Inventory	float	YES		The BBIM inventory value calculated for this constituent in the tank waste component.
			BB_Inv4Display	varchar(12)	YES		Formatted BBIM character version of a float value used for display purposes.
			Units	varchar(12)	YES		Concentration units of the reported value or the reporting limit.
			WU_ID	int	NO		Computer generated unique identifier for the Work_Unit record.
			Type	varchar(12)	NO		The Type of Event that occurred. (may be a useless field -- oversight on my part).
			Source	varchar(4)	NO		A code indicating the source from which waste was transferred
			Inventory_Effective_Date	datetime	NO		The date on which the published inventory for this tank is in effect. Events, such as tank transfers and evaporations may have occurred since this date. In this case, the inventory would not reflect these events (changes). This also includes events such as core samples.
			BB_Document	varchar(50)	YES		The document number of the Best Basis Document where the loading of data for this tank was obtained.
			Doc_Release_Date	datetime	YES		The date that the BB_Document was released.
			CCB_Log	varchar(40)	YES		The Log number of the Configuration Control Board's documentation that authorizes this work unit (change in BBIM data). Current examples are "CCB-001" up to "CCB-064".
			Sort_Order	int	YES		Used for sorting the records in this table or view.
			Basis	varchar(20)	YES		The basis or data source of the inventory calculation. Multiple basis (Tank_Inventory table only) are concatenated using the "/" character (e.g. S/E/T (241-BX-112)).
			Comment	varchar(255)	YES		Comment

dbo.Max_Liquid_Concentration

Contains the maximum liquid concentration of HYDROXIDE and NITRITE that can exist in each tank. When only one of these constituents (NITRITE) is known for any given tank, then the other (HYDROXIDE) can be calculated using the ratio of these maximum concentrations.

I	F	P	Column name	Data type	Nulls	Default	Description
			Waste_Site_ID	varchar(15)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
			Con_ID	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
			Max_Concentration	float	NO		The maximum concentration in ug/mL.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
pk_Max_Liq_Conc	Waste_Site_ID	ASC	Yes	CLUSTERED
pk_Max_Liq_Conc	Con_ID	ASC	Yes	CLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
fk_Max_Liq_Tank	Waste_Site_ID	aaaaaTank_PK (dbo.Tank)	
fk_Max_Liq_Con	Con_ID	Constituent_PK (dbo.Constituent)	

dbo.Phase_Sigma

A static lookup table which contains all the possible combinations of waste phases, and the sigma (variance) values for each phase. The Sigma values are used in the calculation of Tank Volume RSDs (Relative Standard Variation).

I F P	Column name	Data type	Nulls	Default	Description
	Super_Exists	char(1)	NO		Flag indicating whether or not the Supernatant waste phase exists.
	Salt_Exists	char(1)	NO		Flag indicating whether or not the Salt Cake waste phase exists.
	Sludge_Exists	char(1)	NO		Flag indicating whether or not the Sludge waste phase exists.
	Sigma_Inch_Super	float	NO		Variance in vertical inches of the Supernatant phase.
	Sigma_Inch_Salt	float	NO		Variance in vertical inches of the Saltcake phase.
	Sigma_Inch_Sludge	float	NO		Variance in vertical inches of the Sludge phase.

dbo.Representative_Vectors

Vectors are used in calculating Tank_Waste_Component inventories. Representative_Vectors provides the link between these two tables, and contains any additional data needed in these calculations.

I	F	P	Column name	Data type	Nulls	Default	Description
			RS_ID	int	NO		Computer generated unique identifier for the representative vector. (RS_ID is historicalthe table name was changed from Representative_Samples to Representative_Vectors).
			TWT_ID	int	NO		A computer generated unique identifier for the Tank_Waste_Component record. (TWT_ID is historicalthe table name was changed from Tank_Waste_Type to Tank_Waste_Component).
			V_ID	int	NO		Computer generated unique identifier for the vector.
			Include_In_Avg	varchar(1)	NO		Include in average flag. Indicates whether a vector within a group of vectors representing a tank waste component should be used in calculating inventories based on the average values of the vectors.
			Multiplier	float	NO		A numeric value, which is used to modify the concentration values of a vector when inventories are calculated. For example, a sample is taken from tank A and a vector stored in the database for this sample. Next, the waste is pumped into an evaporator and the volume reduced by one half. The waste is then pumped into tank B. The waste in tank B is now twice as concentrated as it was when the sample was taken in tank A. The sample vector can be used to represent the waste in tank B if a multiplier of 2 is factored in.
			Multiplier_Description	text	YES		If a multiplier other than 1.0 is used, then a reason should be documented as to why a multiplier is being applied to the data.
			Vector_Significance	smallint	YES		Indicates the relevance of the vector is in determining the Tank Waste Components inventory of constituents. This is used by program code to perform a best guess calculation of inventory. Within a Tank Waste Component, a vector with a significance of 1 would be selected for inventory calculations over a vector with a significance of 2. The virtual vectors, "Sample Average" has a significance of 10, that is, Sample Average vectors take precedence over single vectors with significance > 10.
			Skip_Utrig	int	YES		No longer used.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
I2_Rep_Vectors	V_ID	ASC		NONCLUSTERED
I3_Rep_Vectors	TWT_ID	ASC		NONCLUSTERED
RV_PK	RS_ID	ASC	Yes	NONCLUSTERED

Triggers

Trigger name:	dbo.Representative_Vectors_ITrig	Created on:	26 Mar 2002
Trigger type:	INSERT	Trigger active:	Yes
Trigger name:	dbo.Representative_Vectors_UTrig	Created on:	23 May 2002
Trigger type:	UPDATE	Trigger active:	Yes
Trigger name:	dbo.Representative_Vectors_DTrig	Created on:	26 Mar 2002
Trigger type:	DELETE	Trigger active:	Yes

dbo.Save_Vector_Locked_Status

Used by the `dbo.Standard_Decay_To_Date` Insert and Update trigger to record the Locked status of vectors. This table could be archived if this trigger is first modified.

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	V_ID	int	NO		Computer generated unique identifier for the vector.
	Locked	int	YES		Lock status of a vector in BBIM database.

dbo.Standard_Waste_Phase

Used to map waste phases to the standard BBI waste phases.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			Std_Waste_Phase	varchar(50)	NO		Baselined tanks have waste that is described by these phases: supernatant, sludge solid, sludge liquid, salt cake solid, salt cake liquid and retained gas.
			Waste_Phase	varchar(50)	NO		The name of the waste phase. In addition, a value of "Total" is used to define all of the waste phases in a tank.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK_Standrd_Waste_Phase	Std_Waste_Phase	ASC	Yes	NONCLUSTERED

dbo.Std_Unit_Conversion

Stores data used in converting various concentrations into ug/g and uCi/g. Formula for conversion is:

I	F	P	Column name	Data type	Nulls	Default	Description
			Units	varchar(12)	NO		Concentration units of the reported value or the reporting limit.
			Std_Units	varchar(6)	NO		Standard Units of the "standard concentration" value. The Std_Units are a subset of the Units_Of_Concentration values.
			Divide_By_Density	smallint	NO		Flag indicating whether division by r (density) is necessary in the conversion code.
			Factor	float	NO		A number used to multiply a initial value to get the converted concentration value.
			Inv_Units	char(2)	NO		The units that inventory is expressed in.
			Inv_Factor	float	NO		A number used to multiply a initial value to get the converted inventory value.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
PK__Std_Unit_Convers__797309D9	Units	ASC	Yes	CLUSTERED

dbo.Tank

The Tank table contains one record for each of the 177 HLW storage tanks at Hanford.

I F P	Column name	Data type	Nulls	Default	Description
	Waste_Site_ID	varchar(15)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	BB_Document	varchar(50)	YES		The document number of the Best Basis Document where the loading of data for this tank was obtained.
	Inventory_Effective_Date	datetime	YES		The date on which the published inventory for this tank is in effect. Events, such as tank transfers and evaporations may have occurred since this date. In this case, the inventory would not reflect these events (changes). This also includes events such as core samples.
	Doc_Release_Date	datetime	YES		The date that the BB_Document was released.
	Baselined	smallint	YES		This Waste_site_id baselined , if -1 then baselined to 2001.
	kL_Per_Inch	float	YES		Volume of waste that the tank holds per vertical inch.
	Sigma_Inch_Super	float	YES		Variance in vertical inches of the Supernatant phase.
	Sigma_Inch_Salt	float	YES		Variance in vertical inches of the Saltcake phase.
	Sigma_Inch_Sludge	float	YES		Variance in vertical inches of the Sludge phase.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
aaaaaTank_PK	Waste_Site_ID	ASC	Yes	NONCLUSTERED
Tank_ID	Waste_Site_ID	ASC		NONCLUSTERED

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.Max_Liquid_Concentration	fk_Max_Liq_Tank	aaaaaTank_PK
dbo.Tank_Waste_Component	FK_TWC_Waste_Site_ID	aaaaaTank_PK

Check constraints

Check name	Column name	Expression
CK_Baselined	Baselined	([Baselined] = 0 or [Baselined] = (-1))

Triggers

Trigger name:	dbo.Tank_DTrig	Created on:	27 Apr 2000
Trigger type:	DELETE	Trigger active:	Yes
Trigger name:	dbo.Tank_ITrig	Created on:	27 Apr 2000
Trigger type:	INSERT	Trigger active:	Yes
Trigger name:	dbo.Tank_UTrig	Created on:	27 Apr 2000
Trigger type:	UPDATE	Trigger active:	Yes

dbo.Tank_Inventory

The table where the calculated inventories for the constituents in each tank are stored. These data are an aggregate of the data found in the Tank_Waste_Inventory table, and are kept up-to-date by the Tank_Waste_Inventory triggers.

I	F	P	Column name	Data type	Nulls	Default	Description
			Waste_Site_ID	varchar(15)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
			Con_ID	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
			Inventory	float	NO		The inventory value calculated for this constituent in the tank waste component.
			Units	varchar(12)	NO		Concentration units of the reported value or the reporting limit.
			Basis	varchar(20)	YES		The basis or data source of the inventory calculation. Multiple basis (Tank_Inventory table only) are concatenated using the "/" character (e.g. S/E/T (241-BX-112)).
			Last_Updated_On	datetime	YES	(getdate())	Date and time that this record was last updated (update or insert).
			Allow_Auto_Update	int	YES	((-1))	A value of "-1" will allow the Tank_Waste_Inventory trigger to "automatically" update the record if a basis record is modified. Any other value "locks" the record (trigger has no effect on the value). Note that -1 is used because in the MS ACCESS front end, -1 is interpreted as an "X" in a bound checkbox field.
			Comment	varchar(255)	YES		Comment
			Alpha_Basis	smallint	YES		Used by the Tank_Waste_Inventory triggers to communicate to the Tank_Inventory triggers as to whether or not to perform Alpha adjustments. See the trigger code for details.
			Inv4Display	varchar(12)	YES	('N.NNE+00')	The inventory value, stored (converted) into scientific notation. Since this is a character field, it is only useful for display purposes. Examples are 1.23E+00; -5.09E99; 4.00E-31
			TI_Inv_Formula	varchar(60)	YES		Not currently used
			TI_Inv_Calc	varchar(60)	YES		Not currently used

Indexes

Index name	Column name	Sort direction	Is unique	Index type
Tank_Inventory_PK	Waste_Site_ID	ASC	Yes	CLUSTERED
Tank_Inventory_PK	Con_ID	ASC	Yes	CLUSTERED
Tank_Inventory_PK	Units	ASC	Yes	CLUSTERED

Triggers

Trigger name:	dbo.Tank_Inventory_UTrig	Created on:	06 Apr 2002
Trigger type:	UPDATE	Trigger active:	Yes
Trigger name:	dbo.Tank_Inventory_ITrig	Created on:	28 Apr 2000
Trigger type:	INSERT	Trigger active:	Yes
Trigger name:	dbo.Tank_Inventory_DTrig	Created on:	28 Apr 2000
Trigger type:	DELETE	Trigger active:	Yes

dbo.Tank_Waste_Component

In the best basis model, inventories are calculated down to the waste component in each tank. Waste components are usually equivalent to the waste phases in the tank. Sometimes waste phases are broken down further by defining waste types within a waste phase. A tank inventory can be calculated by summing the inventories of each tank waste component.

I F P	Column name	Data type	Nulls	Default	Description
PK	TWT_ID	int	NO		A computer generated unique identifier for the Tank_Waste_Component record. (TWT_ID is historical the table name was changed from Tank_Waste_Type to Tank_Waste_Component).
FK	Waste_Site_ID	varchar(15)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
FK	Waste_Phase	varchar(50)	NO		The name of the waste phase. In addition, a value of "Total" is used to define all of the waste phases in a tank.
	Waste_Type	varchar(60)	YES		Inventories are often calculated from process models. Waste types are used in these models to define the wastes produced from different processes. Waste_Type may be NULL (when sample based inventories are used).
	Volume	float	NO		The volume, usually in kL, of this tank waste component. If a tank waste component with a waste phase of "Total" exists, then the sum of the volumes for any other waste components for this tank should equal the "Total" volume.
	Units_Of_Volume	varchar(12)	YES		Every instance in the database is kL, however we may run across other units such as kgal.
	Vol_Reference	varchar(80)	YES		If the value for the volume was obtained from a document other than the best basis document for this tank (see tank.BB_Document), then the document name or document number from which the volume was obtained, otherwise NULL.
	Input_Data_Location	varchar(80)	YES		During the loading of data into the bbim database, tables are extracted from the electronic versions of best basis documents and saved for future use. Input_Data_Location is the path and filename where these tables are stored, relative to the \\TCD_DOCS\best_basis_baseline file share.
	Volume_kL	float	YES		The volume, in kL or kg, of this tank waste component. The Insert & Update triggers calculate this based upon the Volume & Units_of_Volume fields. If the volume is in kg and the Vector_Constituent "concentrations" are in units other than kg or wt%, inventories will not be calculated correctly.
	Average_Density	float	YES	(0)	If Representative_Vectors exist for this Tank_Waste_Component, the Average_Density equals the average of all Vector.Density values for which the Representative_Vectors.Include_In_Avg field is Y.
	Average_Vector_Multiplier	float	YES	(1)	No longer used. Individual Multipliers are applied to the Vectors.
	Bogus_For_Access_Update	char(1)	YES		No longer used.
	Distribution_V_ID	int	YES		Vector Identifier of the vector to be used to distribute the missing HDW and Alpha nuclides
	Volume_RSD	float	YES		Relative Standard Deviation for Volume, in units of percentage. Equal to Std_Dev / Volume
	Density_RSD	float	YES		Relative Standard Deviation for Volume, in units of percentage. Equal to Std_Dev / Volume
	Component_Effective_Date	datetime	YES		Date of the most recent update of component data (volume & inventory) due to the uploading of TXFR (Waste Transfer) data. When this date becomes more recent than the Tank.Inventory_Effective_Date, Tank.Inventory_Effective_Date will be set to this Component_Effective_Date.
	Density	float	YES		The density, in g/ml of the vector. A density is needed to calculate inventories when vector constituents are recorded as concentrations. If vector constituents values are recorded in units of inventory (kg or Ci), then Density is not needed.
	Weight_Percent_Water	float	YES		Weight percent water for this tank from BBI
	Oxalate_Fraction_Of_TOC	float	YES		Percentage of Oxalate in the Total Organic Carbon.
	Oxalate_Comment	varchar(160)	YES		Information about the fraction of Oxalate in the Total Organic Carbon.

Set_Vol_RSD	bit	YES	(0)	If set to 1, then the Volume_RSD is fixed (entered by the user); if set to 0 then calculate the Volume_RSD
Set_Den_RSD	bit	YES	(0)	If set to 1, then the Density_RSD is fixed (entered by the user); if set to 0 then calculate the Density_RSD
Reference_Document	varchar(120)	YES		Not used. There are only three records with non null reference document values in the table.
Density_Lower_95_Percent_C L	float	YES		Component Density Lower 95% Confidence Limit in units of g/mL. The 95% Two-Sided Confidence Intervals are calculated only for Double-Shell Tanks and only when sufficient sample determinations of the waste density are available.
Density_Upper_95_Percent_C L	float	YES		Component Density Upper 95% Confidence Limit in units of g/mL. The 95% Two-Sided Confidence Intervals are calculated only for Double-Shell Tanks and only when sufficient sample determinations of the waste density are available.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
I_TWC_TWT_ID	TWT_ID	ASC	Yes	NONCLUSTERED
I_TWC_Waste_Site_ID	Waste_Site_ID	ASC		CLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK_TWC_Waste_Phase	Waste_Phase	PK_Waste_Phase (dbo.Waste_Phase)	
FK_TWC_Waste_Site_ID	Waste_Site_ID	aaaaaTank_PK (dbo.Tank)	

Triggers

Trigger name:	dbo.Tank_Waste_Component_ITrig	Created on:	11 Jan 2003
Trigger type:	INSERT	Trigger active:	Yes
Trigger name:	dbo.Tank_Waste_Component_UTrig	Created on:	11 Jan 2003
Trigger type:	UPDATE	Trigger active:	Yes
Trigger name:	dbo.Tank_Waste_Component_DTrig	Created on:	11 Jan 2003
Trigger type:	DELETE	Trigger active:	Yes

dbo.Tank_Waste_Component_Hist

Historical data for the Tank_Waste_Component data. Use the Start_Of_Period and End_Of_Period fields as a date range to select a record for a given time. If the End_Of_Period is NULL, then this is the current Tank_Waste_Component data.

I	F	P	Column name	Data type	Nulls	Default	Description
			TWCH_ID	int	NO		Unique Identifier for the Tank_Waste_Component_Hist table.
			TWT_ID	int	NO		A computer generated unique identifier for the Tank_Waste_Component record. (TWT_ID is historical the table name was changed from Tank_Waste_Type to Tank_Waste_Component).
			Start_Of_Period	datetime	NO		The beginning data of a date range.
			End_Of_Period	datetime	YES		The ending date of a date range. If End_Of_Period is NULL, then the date range is OPEN (e.g. range is from Start_Of_Period to NOW).
			Waste_Site_ID	varchar(15)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
			Waste_Phase	varchar(50)	YES		The name of the waste phase. In addition, a value of "Total" is used to define all of the waste phases in a tank.
			Waste_Type	varchar(60)	YES		Inventories are often calculated from process models. Waste types are used in these models to define the wastes produced from different processes. Waste_Type may be NULL (when sample based inventories are used).
			Volume	float	YES		The volume, usually in kL, of this tank waste component. If a tank waste component with a waste phase of "Total" exists, then the sum of the volumes for any other waste components for this tank should equal the "Total" volume.
			Units_Of_Volume	varchar(12)	YES		Every instance in the database is kL, however we may run across other units such as kgal.
			Vol_Reference	varchar(80)	YES		If the value for the volume was obtained from a document other than the best basis document for this tank (see tank.BB_Document), then the document name or document number from which the volume was obtained, otherwise NULL.
			Input_Data_Location	varchar(80)	YES		During the loading of data into the bbim database, tables are extracted from the electronic versions of best basis documents and saved for future use. Input_Data_Location is the path and filename where these tables are stored, relative to the \\TCD_DOCS\best_basis_baseline file share.
			Volume_kL	float	YES		The volume, in kL or kg, of this tank waste component. The Insert & Update triggers calculate this based upon the Volume & Units_of_Volume fields. If the volume is in kg and the Vector_Constituent "concentrations" are in units other than kg or wt%, inventories will not be calculated correctly.
			Average_Density	float	YES		If Representative_Vectors exist for this Tank_Waste_Component, the Average_Density equals the average of all Vector.Density values for which the Representative_Vectors.Include_In_Avg field is "Y".
			Average_Vector_Multiplier	float	YES		No longer used. Individual Multipliers are applied to the Vectors.
			Bogus_For_Access_Update	char(1)	YES		Obsolete field. It was used by the old MS Access user interface.
			Distribution_V_ID	int	YES		Vector Identifier of the vector to be used to distribute the missing HDW and Alpha nuclides
			Volume_RSD	float	YES		Relative Standard Deviation for Volume, in units of percentage. Equal to Std_Dev / Volume
			Density_RSD	float	YES		Relative Standard Deviation for Density, in units of percentage. Equal to Std_Dev / Density.
			Component_Effective_Date	datetime	YES		Date of the most recent update of component data (volume & inventory) due to the uploading of TXFR (Waste Transfer) data. When this date becomes more recent than the Tank.Inventory_Effective_Date, Tank.Inventory_Effective_Date will be set to this Component_Effective_Date.

Density	float	YES	The density, in g/ml of the vector. A density is needed to calculate inventories when vector constituents are recorded as concentrations. If vector constituents values are recorded in units of inventory (kg or Ci), then Density is not needed.
Weight_Percent_Water	float	YES	Weight percent water for this tank from BBI
Oxalate_Fraction_Of_TOC	float	YES	Amount of Oxalate in the Total Organic Carbon.
Oxalate_Comment	varchar(160)	YES	User supplied information about the Oxalate in the waste.
Density_Lower_95_Percent_C L	float	YES	Component Density Lower 95% Confidence Limit in units of g/mL. The 95% Two-Sided Confidence Intervals are calculated only for Double-Shell Tanks and only when sufficient sample determinations of the waste density are available.
Density_Upper_95_Percent_C L	float	YES	Component Density Upper 95% Confidence Limit in units of g/mL. The 95% Two-Sided Confidence Intervals are calculated only for Double-Shell Tanks and only when sufficient sample determinations of the waste density are available.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
I_TWCH_TWT_ID	TWT_ID	ASC		NONCLUSTERED
I_TWCH_Waste_Site_ID	Waste_Site_ID	ASC		NONCLUSTERED
PK__Tank_Waste_Compo__1F198FD4	TWCH_ID	ASC	Yes	CLUSTERED

Triggers

Trigger name: dbo.Tank_waste_Component_HIST_INSERT
 Trigger type: INSERT
 Created on: 26 Mar 2002
 Trigger active: Yes

Trigger name: dbo.Tank_waste_Component_HIST_DELETE
 Trigger type: DELETE
 Created on: 26 Mar 2002
 Trigger active: Yes

Trigger name: dbo.Tank_Waste_Inventory_UTrig
Trigger type: UPDATE

Created on: 30 Aug 2002
Trigger active: Yes

Trigger name: dbo.Tank_Waste_Inventory_ITrig
Trigger type: INSERT

Created on: 30 Aug 2002
Trigger active: Yes

dbo.U_and_Pu_Dependents

U_and_Pu_Dependents records the associations of which Uranium and Plutonium isotopes are dependent on HDW Uranium and Plutonium isotopes for the adjusted isotopic inventory calculations. The *Vector_Constituent* update trigger uses these associations so that the appropriate (dependent) *Tank_Waste_Inventory* calculated isotopic inventories can be updated whenever a related HDW *Vector_Constituent* Uranium or Plutonium value is modified. The associations are static:

I F P	Column name	Data type	Nulls	Default	Description
	Inv_Con_ID	varchar(15)	NO		The affected Constituent identifier.
	HDW_Con_ID	varchar(15)	NO		The Constituent ID of the HDW <i>Vector_Constituent</i> which may have dependencies in the <i>Tank_Waste_Inventory</i> table.

dbo.Units_Of_Measure

Contains the units of measure used in the BBIM.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			Units	varchar(12)	NO		Concentration units of the reported value or the reporting limit.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
aaaaaUnits_Of_Measure_PK	Units	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.Vector_Constituent	FK_VC_Units	aaaaaUnits_Of_Measure_PK

Triggers

Trigger name:	dbo.Units_Of_Measure_DTrig	Created on:	27 Apr 2000
Trigger type:	DELETE	Trigger active:	Yes
Trigger name:	dbo.Units_Of_Measure_ITrig	Created on:	27 Apr 2000
Trigger type:	INSERT	Trigger active:	Yes
Trigger name:	dbo.Units_Of_Measure_UTrig	Created on:	27 Apr 2000
Trigger type:	UPDATE	Trigger active:	Yes

dbo.Vector

Vector refers to the sample or model vectors which contain a set of constituents and concentration values. Vectors are the data columns found in the best basis document tables. For example, table D3-2 below contains the SMM vector for the analytes in tank 241-AW-106.

I	F	P	Column name	Data type	Nulls	Default	Description
			V_ID	int	NO		Computer generated unique identifier for the vector.
			Vector_Type	varchar(20)	NO		In the case of tank inventories based on vectors, Vector_Type = Vector.Vector_Type. In the case of tank inventories based on other tank inventories, Vector_Type = "Tank Inventory". In the case of tank inventories based on vector averages, Vector_Type = "Sample Average".
			Template	varchar(15)	NO		A group of vectors is often used in inventory calculations for many tanks. A group is referred to as a Template. The Template field contains the name of the template.
			Waste_Site_ID	varchar(15)	YES		Tank Identifier, using the format of 241-Tank Farm-Tank #.
			Vector_Name	varchar(40)	NO		In the case of tank inventories based on vectors, Vector_Name = Vector.Vector_Name. In the case of tank inventories based on other tank inventories, Vector_Name = the Waste_Site_ID + the Waste_Type of the tank that the inventory is obtained from. In the case of tank inventories based on vector averages, Vector_Name = "(Calculated)".
			Density	float	YES		The density, in g/ml of the vector. A density is needed to calculate inventories when vector constituents are recorded as concentrations. If vector constituents values are recorded in units of inventory (kg or Ci), then Density is not needed.
			Sample_Date	datetime	YES		For vectors with a type of "Sample", Sample_Date contains the date that the sample was taken. Sometimes this data is not readily available.
			Decay_Date_Reported	datetime	YES	7/***** Object: Default dbo.Decay	The date that the vector's radionuclide values are valid. Decay calculations will be used to determine current and past radionuclide values.
			Ref_Document	varchar(80)	YES		The document number from which the vector data was obtained.
			Ref_Table_Name	varchar(150)	YES		The name of the table within the reference document from which the vector data was obtained.
			Ref_Table_No	varchar(15)	YES		The table number within the reference document from which the vector data was obtained.
			Ref_Table_Column_No	varchar(15)	YES		The column number within the reference table from which the vector data was obtained.
			Vector_Handle	varchar(15)	YES		A unique identifier that can be easily remembered by the users because it follows a naming convention. The Vector_Handle naming convention is First Letter of the Vector_Type + / + Waste_Site_ID, stripped of the 241 and dashes + / + integer, where integers are used to make the Vector_Handle unique for vectors that have the same Vector_Type and Waste_Site_ID. An example of a Vector Handles are {S/A101/1 S/A101/2}
			Bogus_Density	varchar(1)	YES		If the vector does not have density, then it may be necessary to enter a density in order for the inventory calculations to occur. In these cases, a value of "Y" would be entered in the Bogus_Density field to indicate that this density is NOT A MEASURED value.
			Density_Comment	varchar(80)	YES		Information about where the density came from, and or how it was derived.
			Locked	smallint	YES		Lock status of a vector in BBIM database. Not currently in use.
			Bogus_For_Access_Update	char(1)	YES		No longer in use as MS Access is no longer used for maintaining BBIM records.
			Intermediate_Decay_Date	datetime	YES		No Longer Used
			Skip_Utrig	int	YES		No longer used.
			AutoStatHeader_id	int	YES		Unique ID for AutoStatHeader
			AV_ID	int	YES		Unique Identifier for the tbl_New_Mean_Vector_Header

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
I1_Vector	Waste_Site_ID	ASC		NONCLUSTERED
I2_Vector	Vector_Type	ASC		NONCLUSTERED
ix_vector_handle	Vector_Handle	ASC		NONCLUSTERED
Vector_PK	V_ID	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.Vector_Constituent	FK_VC_V_ID	Vector_PK

Triggers

Trigger name:	dbo.Vector_UTrig	Created on:	23 May 2002
Trigger type:	UPDATE	Trigger active:	Yes
Trigger name:	dbo.Vector_ITrig	Created on:	23 May 2002
Trigger type:	INSERT	Trigger active:	Yes
Trigger name:	dbo.Vector_DTrig	Created on:	26 Mar 2002
Trigger type:	DELETE	Trigger active:	Yes

dbo.Vector_Constituent

The individual constituents that belong to a vector.

I	F	P	Column name	Data type	Nulls	Default	Description
			VC_ID	int	NO		Unique Identifier of the Vector_Constituent record. This is an identity column set by the SQL Server upon insert.
			V_ID	int	NO		Computer generated unique identifier for the vector.
			Con_ID	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
			Qualifier	varchar(3)	YES		Qualifies the concentration of the constituent.
			Reported_Concentration	float	YES		The concentration (activity for radionuclides) of this constituent, as entered by the user. Notesome of the best basis documents only report vector data in units of inventory. The original data for these vectors come from the Tank Characterization Report (TCR). Even though concentrations should be entered for vector constituents, we are allowing values with inventory units to be entered to ease the burden of data mining (going all the way back to the TCRs to obtain the data). At a later date, the "inventory" vector data should be updated with concentration data.
			Decayed_Concentration	float	YES		The calculated (adjusted) activity of this radionuclide based upon the Reported_Concentration, the Vector.Decay_Date_Reported, and the Standard_Decay_To_Date.Standard_Decay_Date. If the dates are the same, then the Decayed_Concentration equals the Reported_Concentration for this radionuclide. If this constituent is an analyte, Decayed_Concentration is NULL.
			Concentration	float	YES		Concentration is a redundant field, added as a single data point to use by database procedures and triggers. Concentration equals Decayed_Concentration if Decayed_Concentration is not null, otherwise Concentration equals Reported_Concentration.
			Units_of_Concentration	varchar(12)	YES		Units of the concentration values. Note that units may be inventory units too (refer to Vector_Constituent.Concentration).
			Std_Concentration	float	YES	(0)	The concentration of this constituent in "standard units". This value is calculated in the table update and insert triggers.
			Std_Units	varchar(12)	YES	('??')	Standard Units of the "standard concentration" value. The Std_Units are a subset of the Units_Of_Concentration values.
			Decay_Factor	float	YES		For radionuclides, Decayed_Concentration = Decay_Factor * Reported_Concentration. The Decay_Factor is calculated based upon the Constituent.Half_Life, Vector.Decay_Date_Reported, and Standard_Decay_To_Date.Standard_Decay_Date. For analytes, Decay_Factor is NULL.
			Comment	varchar(100)	YES		Comment
			Last_Updated_On	datetime	YES	(getdate())	Date and time that this record was last updated (update or insert).
			User_Comment	varchar(100)	YES		Information written by one or more users. The triggers should not update this field.
			Bogus_For_Access_Update	char(1)	YES		No Longer Used
			Parent_Decay_Factor	float	YES		No Longer Used
			Intermediate_Decayed_Conc	float	YES		No Longer Used
			Intermediate_Decay_Factor	float	YES		No Longer Used
			Intermediate_Parent_Decay_Factor	float	YES		No Longer Used
			Concentration_RSD	float	YES		Relative Standard Deviation for Concentration, in units of percentage. Equal to Std_Dev / Concentration.

basis	varchar(20)	YES	The basis or data source of the inventory calculation. Multiple basis (Tank_Inventory table only) are concatenated using the "/" character (e.g. S/E/T (241-BX-112)).
AutoStatResults_id	int	YES	Unique ID for AutoStatResult
NV_ID	int	YES	Link to the record in the tcd.dbo.tblNewVector_Choices1, which is generated by the AutoVector application.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
I2_Vector_Constituent	Con_ID	ASC		NONCLUSTERED
u_Vector_Constituent	V_ID	ASC	Yes	CLUSTERED
u_Vector_Constituent	Con_ID	ASC	Yes	CLUSTERED
VC_PK	VC_ID	ASC	Yes	NONCLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK_VC_Units	Units_of_Concentration	aaaaaUnits_Of_Measure_PK (dbo.Units_Of_Measure)	
FK_VC_Con_ID	Con_ID	Constituent_PK (dbo.Constituent)	
FK_VC_V_ID	V_ID	Vector_PK (dbo.Vector)	

Triggers

Trigger name:	dbo.Vector_Constituent_IUTrig	Created on:	24 Aug 2004
Trigger type:	INSERT, UPDATE	Trigger active:	Yes
Trigger name:	dbo.Vector_Constituent_DTrig	Created on:	26 Mar 2002
Trigger type:	DELETE	Trigger active:	Yes

dbo.Vector_Constituent_Excel_Data

Holds data that is copied from an Excel spreadsheet prior to being inserted into the Vector_Constituent table.

I	F	P	Column name	Data type	Nulls	Default	Description
			VCED_ID	int	NO		Computer generated unique identifier for the record.
			UserName	varchar(100)	NO		Name or identifier of person requesting an AutoTCR report.
			V_ID	int	NO		Computer generated unique identifier for the vector.
			Constituent_Name	varchar(50)	NO		Name of the constituent.
			Qualifier	varchar(3)	YES		Qualifies the concentration of the constituent.
			Reported_Concentration	float	YES		The concentration (activity for radionuclides) of this constituent, as entered by the user. Notesome of the best basis documents only report vector data in units of inventory. The original data for these vectors come from the Tank Characterization Report (TCR). Even though concentrations should be entered for vector constituents, we are allowing values with inventory units to be entered to ease the burden of data mining (going all the way back to the TCRs to obtain the data). At a later date, the "inventory" vector data should be updated with concentration data.
			Units_of_Concentration	varchar(12)	YES		Units of the concentration values. Note that units may be inventory units too (refer to Vector_Constituent.Concentration).
			Concentration_RSD	float	YES		Relative Standard Deviation for Concentration, in units of percentage. Equal to Std_Dev / Concentration.
			Last_Updated_On	datetime	YES	(getdate())	Date and time that this record was last updated (update or insert).

dbo.Waste_Phase

Waste in the Hanford HLW tanks separate out into different phases. The Waste_Phase table contains one record for each of these phases:

Retained Gas (Saltcake)
Retained Gas (Sludge)
Saltcake (Liquid & Solid)
Saltcake Interstitial Liquid
Saltcake Solid
Sludge (Liquid & Solid)
Sludge Interstitial Liquid
Sludge Solid
Supernatant

I	F	P	Column name	Data type	Nulls	Default	Description
			Waste_Phase	varchar(50)	NO		The name of the waste phase. In addition, a value of "Total" is used to define all of the waste phases in a tank.
			Physical_State	varchar(10)	NO		One of Gas, Liquid or Solid

Indexes

Index name	Column name	Sort direction	Is unique	Index type
PK_Waste_Phase	Waste_Phase	ASC	Yes	CLUSTERED

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.Tank_Waste_Component	FK_TWC_Waste_Phase	PK_Waste_Phase

Triggers

Trigger name:	dbo.Waste_Phase_DTrig	Created on:	27 Apr 2000
Trigger type:	DELETE	Trigger active:	Yes
Trigger name:	dbo.Waste_Phase_ITrig	Created on:	27 Apr 2000
Trigger type:	INSERT	Trigger active:	Yes
Trigger name:	dbo.Waste_Phase_UTrig	Created on:	27 Apr 2000
Trigger type:	UPDATE	Trigger active:	Yes

dbo.Work_Unit

When BBIM data is modified, it may affect other BBIM data (e.g. modifying a vector may cause a Tank's inventory to change). If the Configuration Control Board decides that BBIM data needs modification, then a "Work Unit" is created, such that the data to be modified, and all data that is dependent upon the data to be modified is placed in this unit.

I	F	P	Column name	Data type	Nulls	Default	Description
			WU_ID	int	NO		Computer generated unique identifier for the Work_Unit record.
			State	varchar(9)	NO		During the lifecycle of the work unit, it changes state. The normal lifecycle of a work unit is Unlocked à Pending à Published.
			Last_Updated_On	datetime	NO	(getdate())	Date and time that this record was last updated (update or insert).
			Last_Updated_By	varchar(40)	NO	(user_name(null))	The SQLServer user name of the person who performed the most recent update
			CCB_Log	varchar(40)	YES		The Log number of the Configuration Control Board's documentation that authorizes this work unit (change in BBIM data). Current examples are "CCB-001" up to "CCB-064".
			Title	varchar(40)	YES		The title of this work unit. Titles are used over and over again. Current examples are "New TCR", "Revised TCR", "Reconcile" and "Discrepancies".
			Description	varchar(160)	YES		Brief narrative describing a specific instance in this database table.
			Justification	varchar(160)	YES		Reason for the change.
			Impact	varchar(20)	YES		Short description of number of tanks impacted by this work unit. Current examples are "Single Tank", "Multiple Tanks" and "All Tanks".
			Comment	text	YES		Comment

Indexes

Index name	Column name	Sort direction	Is unique	Index type
pk_Work_Unit	WU_ID	ASC	Yes	CLUSTERED

Triggers

Trigger name:	dbo.Work_Unit_ITrig	Created on:	30 Mar 2001
Trigger type:	INSERT	Trigger active:	Yes
Trigger name:	dbo.Work_Unit_DTrig	Created on:	28 Oct 2002
Trigger type:	DELETE	Trigger active:	Yes
Trigger name:	dbo.Work_Unit_UTrig	Created on:	26 Mar 2002
Trigger type:	UPDATE	Trigger active:	Yes

dbo.Work_Unit_Archive

Contains an archive of all the Work_Unit records. Whenever a work unit's state is changed, the old record is inserted into the Work_Unit_Archive table. When a work unit is published, the Work_Unit record is deleted, along with its members.

I	F	P	Column name	Data type	Nulls	Default	Description
			WUA_ID	int	NO		Computer generated unique identifier for the Work_Unit_Archive record.
			WU_ID	int	NO		Computer generated unique identifier for the Work_Unit record.
			State	varchar(9)	NO		During the lifecycle of the work unit, it changes state. The normal lifecycle of a work unit is Unlocked à Pending à Published.
			Last_Updated_On	datetime	NO		Date and time that this record was last updated (update or insert).
			Last_Updated_By	varchar(40)	NO		The SQLServer user name of the person who performed the most recent update
			CCB_Log	varchar(40)	YES		The Log number of the Configuration Control Board's documentation that authorizes this work unit (change in BBIM data). Current examples are "CCB-001" up to "CCB-064".
			Title	varchar(40)	YES		The title of this work unit. Titles are used over and over again. Current examples are "New TCR", "Revised TCR", "Reconcile" and "Discrepancies".
			Description	varchar(160)	YES		Brief narrative describing a specific instance in this database table.
			Justification	varchar(160)	YES		Reason for the change.
			Impact	varchar(20)	YES		Short description of number of tanks impacted by this work unit. Current examples are "Single Tank", "Multiple Tanks" and "All Tanks".
			Comment	text	YES		Comment

Triggers

Trigger name: dbo.Work_Unit_Archive_ITrig
Trigger type: INSERT

Created on: 12 Jun 2002
Trigger active: Yes

dbo.Work_Unit_Member

Contains the individual members of a work unit. Members can only be tanks and vectors.

I	F	P	Column name	Data type	Nulls	Default	Description
			WUM_ID	int	NO		Computer generated unique identifier for the Work_Unit_Member record.
			WU_ID	int	NO		Computer generated unique identifier for the Work_Unit record.
			Member_Type	varchar(6)	NO		In the BBIM, Vectors and Tanks can be members of a work unit. Historical data may also be a member.
			Member_ID	varchar(15)	NO		The identifier of the member, or, for historical data, may contain a list of tanks, or tank farm(s), or text.
			Added_To_Unit_On	datetime	NO	(getdate())	The date and time that this member was added to the work unit
			Driver_Type	varchar(6)	YES		"Driver" refers to the object that caused this vector or tank to become a member. For example, if Vector S/A101/01 is used in the calculations of the inventory for tank 241-A-101, then S/A101/01 is the "Driver" for the inclusion of 241-A-101 into this work unit. In other words, if S/A101/01 is added to a work unit, then 241-A-101 MUST also be added. Driver_Type refers to the driver's type (tank or vector). If this member is the initial entry into the work unit, then Driver_Type would be NULL.
			Driver_ID	varchar(15)	YES		the identifier of the driver

Indexes

Index name	Column name	Sort direction	Is unique	Index type
I1_WUM_WU_ID	WU_ID	ASC		CLUSTERED
I2_WUM_WU_ID	Member_ID	ASC		NONCLUSTERED
I3_WUM_member_type	Member_Type	ASC		NONCLUSTERED
pk_Work_Unit_Member	WUM_ID	ASC	Yes	NONCLUSTERED

Triggers

Trigger name:	dbo.Work_Unit_Member_ITrig	Created on:	27 Apr 2000
Trigger type:	INSERT	Trigger active:	Yes
Trigger name:	dbo.Work_Unit_Member_UTrig	Created on:	27 Apr 2000
Trigger type:	UPDATE	Trigger active:	Yes

dbo.Work_Unit_Member_Archive

Contains an archive of all the Work_Unit records. Whenever the state of a work unit is changed, the old Work_Unit record is inserted into the Work_Unit_Archive, and its Work_Unit_Members are inserted into the Work_Unit_Archive table.

I	F	P	Column name	Data type	Nulls	Default	Description
			WUMA_ID	int	NO		Computer generated unique identifier for the Work_Unit_Member_Archive record.
			WUA_ID	int	NO		Computer generated unique identifier for the Work_Unit_Archive record.
			WUM_ID	int	NO		Computer generated unique identifier for the Work_Unit_Member record.
			Member_Type	varchar(6)	NO		In the BBIM, Vectors and Tanks can be members of a work unit. Historical data may also be a member.
			Member_ID	varchar(200)	NO		The identifier of the member, or, for historical data, may contain a list of tanks, or tank farm(s), or text.
			Added_To_Unit_On	datetime	NO		The date and time that this member was added to the work unit
			Driver_Type	varchar(6)	YES		"Driver" refers to the object that caused this vector or tank to become a member. For example, if Vector S/A101/01 is used in the calculations of the inventory for tank 241-A-101, then S/A101/01 is the "Driver" for the inclusion of 241-A-101 into this work unit. In other words, if S/A101/01 is added to a work unit, then 241-A-101 MUST also be added.
			Driver_ID	varchar(15)	YES		Driver_Type refers to the driver's type (tank or vector). If this member is the initial entry into the work unit, then Driver_Type would be NULL. the identifier of the driver
			inventory_effective_date	datetime	YES		The date on which the published inventory for this tank is in effect. Events, such as tank transfers and evaporations may have occurred since this date. In this case, the inventory would not reflect these events (changes). This also includes events such as core samples.

Indexes

	Index name	Column name	Sort direction	Is unique	Index type
	I1_WUMA_WUA_ID	WUA_ID	ASC		CLUSTERED

VIEWS

dbo.v_bbi_ccb_tracker

Created: 12 Jul 2002 Last updated: 12 Jul 2002

TWINS Query: Presents the current status of changes proposed to or approved by the Change Control Board. This query provides users with a summary of changes to the best-basis data set contained in the TCD. Data is obtained from the Work_Unit_Archive and Work_Unit_Archive_Member tables.

Resultset

Source	Column name	Data type	Nulls	Description
dbo.Work_Unit_Archive	ccb_log	varchar(40)	YES	A unique number assigned by the Change Control Board (CCB) to a set of best-basis inventory change records. This number is used to track the changes through the CCB approval and TCD data loading process.
	change_title	varchar(40)	YES	Indicates the particular activity or task that is the basis for initiating the change package.
dbo.Work_Unit_Archive	impact	varchar(20)	YES	Extent of the inventory change typically reported in terms of single tank or multiple tank impact.
	TCD_entry_date_time	datetime	NO	Date on which the corresponding change record was entered into the TCD. This same date is applied to each analyte changed as part of the respective change log. Once the data has been entered into TCD, the change is immediately visible via the queries available through the TWINS interface.
	tracker_change_description	varchar(160)	YES	Description of what is being changed.
	tracker_change_justification	varchar(160)	YES	Explanation of basis for the change.
dbo.Work_Unit_Member_Archive	waste_site_id_list	varchar(200)	NO	List of tanks or tank farms to which the inventory change records in the in the specific CCB Log pertain.
	inventory_effective_date	datetime	YES	Represents a data cut-off date. Indicates to users that newer sample data or transactions occurring after this date have not been considered as part of this best-basis assessment.
	published_on	datetime	NO	Date and time that this record was copied over from either the TCD database or the BBIM database (update or insert).

dbo.v_Calc_Detail_RDR

Created: 10 Jan 2007 Last updated: 12 Mar 2010

Used by the BBI ASP.NET application for closure tanks.

Resultset

Source	Column name	Data type	Nulls	Description
	Tank Name	varchar(15)	NO	
	Analyte	varchar(50)	NO	Standard best-basis analyte expressed as chemical symbol (reported without charge designation per the best-basis inventory convention).
	Waste Phase	varchar(50)	NO	The name of the waste phase. In addition, a value of "Total" is used to define all of the waste phases in a tank.
	Waste Type	varchar(60)	YES	Inventories are often calculated from process models. Waste types are used in these models to define the wastes produced from different processes. The waste type may be designated NA (Liquid), NA (Sludge) or NA (SlitCK) if the waste does not correspond to a defined waste type or if the waste layer is a mixture of waste types
dbo.Tank_Waste_Inventory	Qualifier	varchar(3)	NO	Indicates whether the value is less than < the stated value.
dbo.Tank_Waste_Inventory	Inventory	varchar(15)	YES	Total amount expressed in kg or Ci
	Inventory Units	varchar(12)	NO	Common units of measure: kilograms (kg) for chemicals and curies (Ci) for radionuclides.
	Inventory RSD	varchar(15)	NO	The standard deviation of the inventory, expressed as a fraction of the inventory. Inventory Unc = ((Density RSD)**2 + (Vol RSD)**2 + (Conc RSD)**2)**1/2
dbo.Tank_Inventory	Basis	varchar(20)	YES	The basis or data source of the inventory calculation. Multiple basis (Tank_Inventory table only) are concatenated using the "/" character (e.g. S/E/T (241-BX-112)).
	Inventory Formula	varchar(257)	YES	The mathematical expression for the calculation used in deriving inventories.
	Inventory Calculation	varchar(255)	YES	Detailed calculation showing the actual concentration, density, volume, and unit adjustment values used to derive the corresponding inventory value.
	Vector Density	varchar(15)	NO	The vector density is used to calculate inventory in the equation rho*vol*conc.
	Vector Density Units	varchar(4)	NO	Density units used in the calculation.
	Vector Density RSD	varchar(15)	YES	The density standard deviation, expressed as a fraction of the density.
	Volume	varchar(30)	YES	Volume used in the calculation
	Volume Units	varchar(2)	NO	Volume units used in the calculation.
	Volume RSD	varchar(15)	YES	The volume standard deviation, expressed as a fraction of the volume.
	Vector Concentration	varchar(15)	YES	The Vector Concentration (used with vector density, waste component volume and the multiplier to calculate Inventory).
	Vector Concentration Units	varchar(12)	NO	Concentration units used in the calculation.
dbo.Representative_Vectors	Multiplier	varchar(15)	YES	Multiplier used to adjust the concentration
	Component Density	varchar(15)	NO	The density value selected to best represent the current density of the waste layer. The component density is frequently the mean density of the preferred sample vector.
	Density Units	varchar(4)	NO	Units of Density
	Adjusted Concentration	varchar(15)	YES	The analyte concentration adjusted for changes in volume or weight percent water to reflect the current waste condition. Adjusted concentrations for liquids are reported on a volumetric basis (i.e. ug/mL or uCi/mL). Solid phase concentrations are calculated on a mass basis (i.e. ug/g or uCi/g) using the Component Density.
	Adjusted Concentration Units	varchar(6)	NO	Units used in the adjusted calculation.
	Decay To Date	varchar(30)	YES	The date to which this analyte is decayed.

Inventory Effective Date	datetime	YES	The date on which the published inventory for this tank is in effect. Events, such as tank transfers and evaporations may have occurred since this date. In this case, the inventory would not reflect these events (changes). This also includes events such as core samples.
Reference Document	varchar(120)	NO	Reference number for the related documentation.
Physical State	varchar(10)	YES	Liquid or Solid
Sort Order	int	NO	Values which when sorted on will arrange the query in the standard best-basis analyte order.
Concentration RSD	varchar(15)	YES	Relative Standard Deviation of the Concentration.

dbo.v_Component_List

Created: 18 Jul 2003 Last updated: 18 Jul 2003

Used by the Component Reports form in the BBI ASP.NET application.

Resultset

Source	Column name	Data type	Nulls	Description
dbo.Tank_Waste_Component	TWT_ID	int	NO	Unique identifier of this Tank Waste Component
	Waste_Component	varchar(112)	NO	Concatenation of the Tank Name, Waste Phase and Waste Type, for example 241-AN-106, Saltcake (Liquid & Solid), NA (Slitck)
dbo.Work_Unit	State	varchar(8)	NO	Indication of whether the record can be modified. Either Locked or Unlocked.

dbo.v_con_specific_activity

Created: 16 May 2003 Last updated: 16 May 2003

TWINS Query: This report is used to produce a listing of the specific_activity for the radionuclides

Resultset

Source	Column name	Data type	Nulls	Description
dbo.Constituent	Constituent_Name	varchar(50)	NO	Name of the chemical or radionulide.
dbo.Constituent	Specific_Activity	varchar(30)	YES	Of the radioisotope, measured in Ci/g. Used in the conversion of ug/g to uCi/g, and ug/mL to uCi/mL.
dbo.Constituent	sort_order	int	NO	An integer which dictates the sequence that each record should be displayed.

dbo.v_constituent_dependents

Created: 05 Apr 2002 Last updated: 05 Apr 2002

Certain radionuclide inventories can be calculated based on another radionuclide inventory. The formula for calculating a dependent inventory is $Dependent\ Inventory = Source\ Inventory * Factor$. The dependents are

- 90Sr --> 90Y
- 137Cs --> 137mBa
- 243/244Cm --> 243Cm
- 243/244Cm --> 244Cm
- 90Sr --> Sr

Resultset

Source	Column name	Data type	Nulls	Description
	con_name1	varchar(12)	YES	The Constituent name of the inventory that may be calculated.
	con_name2	varchar(12)	YES	The Constituent name of the inventory that may be used to calculate the inventory for Con_ID1.
dbo.Constituent_Dependents	factor	float	NO	A number used in the multiplication of the inventory of Con_ID2 to obtain the inventory for Con_ID1.
dbo.Constituent_Dependents	con_id1	varchar(15)	NO	The Constituent identifier of the inventory that may be calculated.
dbo.Constituent_Dependents	con_id2	varchar(15)	NO	The Constituent identifier of the inventory that may be used to calculate the inventory for Con_ID1.

dbo.v_Decay_Chain

Created: 27 Apr 2000 Last updated: 27 Apr 2000

Used by the `p_Decay_And_Distribute` procedure to calculate 2nd Order decay of radionuclides. These are:

- 93Zr --> 93mNb
- 231Pa --> 227Ac
- 232Th --> 228Ra
- 241Pu --> 241Am

Resultset

Source	Column name	Data type	Nulls	Description
	Parent	varchar(50)	NO	The constituent name of the radionuclide which will undergo decay.
	Daughter	varchar(50)	NO	The constituent name of the Parent_Con_IDs daughter product.
dbo.Decay_Chain	Percent_Decayed_Div100	float	NO	If a radionuclide decays to multiple daughters, then a percentage decays to one daughter product, and a percentage decays to another. The sum of the percentages must equal 100. Percent_Decayed_Div100 is the percent decayed to this particular daughter product from this particular parent, divided by 100.

dbo.v_Error_In_Con_Inventory

Created: 27 Apr 2000 Last updated: 27 Apr 2000

Shows if there is a difference in the sum of the Tank_Waste_Inventory and the Tank_Inventory inventories by constituent. A nice check to make sure there are no problems in the denormalization of the Tank_Inventory values.

Resultset

Source	Column name	Data type	Nulls	Description
dbo.Tank_Inventory	Waste_Site_ID	varchar(15)	NO	Tank Identifier, using the format of 241-Tank Farm-Tank #.
	Con_Name	varchar(15)	YES	Name of the constituent.
dbo.Tank_Inventory	Inventory	float	NO	Tank_Inventory value.
	SumOfConInventory	float	YES	Sum of the Tank_Waste_Inventory values.
dbo.Tank_Inventory	Units	varchar(12)	NO	Kg or Ci

dbo.v_RPD

Created: 27 Apr 2000 Last updated: 27 Apr 2000

Shows relative percent deviation between BBIM and BB_Published inventories for tanks currently in a Work Unit.

Resultset

Source	Column name	Data type	Nulls	Description
dbo.Work_Unit_Member	WU_ID	int	NO	Work Unit identifier
	Tank	varchar(15)	NO	Waste Site identifier
	RPD	float	YES	Relative Percent Difference: $(\text{abs}(x1 - x2) / ((x1 + x2)/2)) * 100$
	Con_Name	varchar(14)	YES	Name of the constituent.
	Update_Mode	varchar(7)	YES	Either NULL or Force. If Force, then the published inventory will be overwritten by the user supplied value.
	BB_Inv	float	NO	Inventory from the BBIM database.
	PUB_Inv	float	NO	Published Value of the Inventory
dbo.Tank_Inventory	Units	varchar(12)	NO	Kg or Ci
	BB_Basis	varchar(20)	YES	Basis from the BBIM database.
	PUB_Basis	varchar(20)	NO	Published Value of the Basis
	PUB_Source	varchar(4)	NO	Always BBIM. When the BBI was first created, some of the original data came from TCD. That is no longer the case.
dbo.Constituent	Con_ID	varchar(15)	NO	Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
dbo.Constituent	Type	varchar(12)	NO	Radionuclide or Chemical.
dbo.Constituent	Sort_Order	int	YES	An integer which dictates the sequence that each record should be displayed.

dbo.v_RPD4TanksInWorkUnit

Created: 27 Apr 2000 Last updated: 27 Apr 2000

Used by the PrepForPub form in the BBI ASP.NET application

Resultset

Source	Column name	Data type	Nulls	Description
dbo.RPD4TanksInWorkUnit	Waste_Site_ID	varchar(15)	NO	Tank Identifier, using the format of 241-Tank Farm-Tank #.
dbo.RPD4TanksInWorkUnit	Con_ID	varchar(15)	NO	Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
dbo.RPD4TanksInWorkUnit	Units	varchar(12)	NO	Kg or Ci
dbo.RPD4TanksInWorkUnit	WU_ID	int	NO	Work Unit identifier
dbo.RPD4TanksInWorkUnit	Con_Name	varchar(15)	NO	Name of the constituent.
dbo.RPD4TanksInWorkUnit	Type	varchar(12)	NO	Radionuclide or Chemical.
dbo.RPD4TanksInWorkUnit	State	varchar(9)	YES	Indication of whether the record can be modified. Either Locked or Unlocked.
dbo.RPD4TanksInWorkUnit	BB_CCB_Log	varchar(40)	YES	Configuration Control Board Log number.
dbo.RPD4TanksInWorkUnit	Sort_Order	int	YES	An integer which dictates the sequence that each record should be displayed.
dbo.RPD4TanksInWorkUnit	PUB_Source	varchar(4)	YES	Original source of the data. Always set to BBIM.
dbo.RPD4TanksInWorkUnit	Update_Mode	varchar(7)	YES	Either NULL or Force. If Force, then the published inventory will be overwritten by the user supplied value.
dbo.RPD4TanksInWorkUnit	Justification	varchar(80)	YES	Reason for the change.
dbo.RPD4TanksInWorkUnit	RPD	real	YES	Relative Percent Difference: $(\text{abs}(x1 - x2) / ((x1 + x2)/2)) * 100$
dbo.RPD4TanksInWorkUnit	BB_Inv	varchar(12)	YES	Inventory from the BBIM database.
dbo.RPD4TanksInWorkUnit	PUB_Inv	varchar(12)	YES	Published Value of the Inventory
dbo.RPD4TanksInWorkUnit	BB_Basis	varchar(20)	YES	Basis from the BBIM database.
dbo.RPD4TanksInWorkUnit	PUB_Basis	varchar(20)	YES	Published Value of the Basis
dbo.RPD4TanksInWorkUnit	BB_Comment	varchar(255)	YES	Comment from the BBIM database.
dbo.RPD4TanksInWorkUnit	PUB_Comment	varchar(255)	YES	Comment from the BB_Published database.
dbo.RPD4TanksInWorkUnit	BB_Inventory_Effective_Date	datetime	YES	Date that the inventory was calculated. Value is from the BBIM database.
dbo.RPD4TanksInWorkUnit	PUB_Inventory_Effective_Date	datetime	YES	Date that the inventory was calculated. Value is from the BB_Published database.
dbo.RPD4TanksInWorkUnit	BB_BB_Document	varchar(50)	YES	Best Basis Document number. Value is from the BBIM database.
dbo.RPD4TanksInWorkUnit	PUB_BB_Document	varchar(50)	YES	Best Basis Document number. Value is from the BB_Published database.
dbo.RPD4TanksInWorkUnit	BB_Doc_Release_Date	datetime	YES	Date document was released.
dbo.RPD4TanksInWorkUnit	PUB_Doc_Release_Date	datetime	YES	Published Value of the Doc_Release_Date
dbo.RPD4TanksInWorkUnit	User_Name	varchar(30)	NO	Name of the user who inserted or last modified this record.

dbo.v_Tank_Inventory

Created: 27 Apr 2000 Last updated: 27 Apr 2000

Retrieves data from the Tank_Inventory table along with its Constituent Name from the Constituent table

Resultset

Source	Column name	Data type	Nulls	Description
dbo.Tank_Inventory	Waste_Site_ID	varchar(15)	NO	Tank Identifier, using the format of 241-Tank Farm-Tank #.
dbo.Constituent	Con_ID	varchar(15)	NO	Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
dbo.Constituent	Constituent_Name	varchar(50)	NO	Name of the chemical or radionuclide.
dbo.Tank_Inventory	Inventory	float	NO	Total amount expressed in kg or Ci
dbo.Tank_Inventory	Units	varchar(12)	NO	Kg or Ci
dbo.Tank_Inventory	Basis	varchar(20)	YES	The basis or data source of the inventory calculation. Multiple basis (Tank_Inventory table only) are concatenated using the "/" character (e.g. S/E/T (241-BX-112)).
dbo.Tank_Inventory	Last_Updated_On	datetime	YES	Date and time that this record was last updated (update or insert).
dbo.Tank_Inventory	Allow_Auto_Update	int	YES	Not used. Original concept was to block updates to data at the constituent inventory level.
dbo.Tank_Inventory	Comment	varchar(255)	YES	Information about the data
dbo.Constituent	Type	varchar(12)	NO	Chemical or Radionuclide.
dbo.Constituent	Sort_Order	int	YES	An integer which dictates the sequence that each record should be displayed.

dbo.v_Tank_Inventory_Charge_Eq

Created: 27 Apr 2000 Last updated: 27 Apr 2000

Retrieves the Charge Equivalent ($\text{primary valence} * \text{Inventory} * 1000 / \text{Molecular Weight}$) from the Tank_Inventory table

Resultset

Source	Column name	Data type	Nulls	Description
dbo.Tank_Inventory	Waste_Site_ID	varchar(15)	NO	Tank Identifier, using the format of 241-Tank Farm-Tank #.
	Con_Name	varchar(12)	YES	Name of the constituent.
dbo.Constituent	Molecular_Weight	float	NO	The molecular weight of the constituent is needed for converting M (moles/L) to inventory units (kg).
dbo.Constituent	Primary_Valence	float	YES	Valence refers to the outer electronic configuration, or, the highest occupied electron energy level of the constituent. Primary_Valence is either the valence (whole number) of the constituent in its prevalent state, or, a weighted average of its valences. Primary_Valence is used in calculating the amount of hydroxide that must exist in a tank in order to get a net charge of zero. The Primary_
dbo.Tank_Inventory	Inventory	float	NO	Total amount expressed in kg or Ci
dbo.Tank_Inventory	Basis	varchar(20)	YES	The basis or data source of the inventory calculation. Multiple basis (Tank_Inventory table only) are concatenated using the "/" character (e.g. S/E/T (241-BX-112)).
	Charge_Equivalent	float	YES	$\text{primary valence} * \text{Inventory} * 1000 / \text{Molecular Weight}$
dbo.Constituent	Con_ID	varchar(15)	NO	Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.

dbo.v_U_and_Pu_Dependents

Created: 27 Apr 2000 Last updated: 27 Apr 2000

U_and_Pu_Dependents records the associations of which Uranium and Plutonium isotopes are dependent on HDW Uranium and Plutonium isotopes for the adjusted isotopic inventory calculations. The *Vector_Constituent* update trigger uses these associations so that the appropriate (dependent) *Tank_Waste_Inventory* calculated isotopic inventories can be updated whenever a related HDW *Vector_Constituent* Uranium or Plutonium value is modified. The associations are static:

Resultset

<u>Source</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Description</u>
	INV_Con_Name	varchar(12)	YES	The affected Constituent name.
	HDW_Con_Name	varchar(12)	YES	The Constituent Name of the HDW Vector_Constituent which may have dependencies in the Tank_Waste_Inventory table.
dbo.U_and_Pu_Dependents	Inv_Con_Id	varchar(15)	NO	The affected Constituent identifier.
dbo.U_and_Pu_Dependents	HDW_Con_ID	varchar(15)	NO	The Constituent ID of the HDW Vector_Constituent which may have dependencies in the Tank_Waste_Inventory table.

Stored procedures

dbo.p_bbim_login

Created: 30 Apr 2001 Last updated: 30 Apr 2001

Used by DTS packages to test whether a connection to the BBIM database can be made successfully.

dbo.p_Calc_Detail_RDR

Created: 10 Jan 2007 Last updated: 10 Jan 2007

Used by the Calculation Detail report for RDR tanks in the BBI ASP.NET application.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@Waste_Site_ID	varchar	IN	

dbo.p_Check_If_Con_Exists

Created: 21 May 2007 Last updated: 21 May 2007

Used by the BBI ASP.NET application to determine if TIC, SOF and PO4 exists for a given vector.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@V_ID	int	IN	
@TicExists	int	INOUT	
@PEExists	int	INOUT	
@SEExists	int	INOUT	

dbo.p_compute_sci_all

Created: 27 Apr 2000 Last updated: 27 Apr 2000

Converts a float into a value formatted in scientific notation.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@valin	float	IN	
@sciout	char	INOUT	
@decimalPlaces	int	IN	

dbo.p_Decay_And_Distribute

Created: 16 Jan 2003 Last updated: 13 May 2009

For this component, distribute the HDW isotopes from their parents (239/240Pu, UTOTAL, 239Pu, 241Am, 243/244Cm, 154Eu), distribute from TotalAlpha, and perform the necessary radioactive decay logic.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@TWT_ID	int	IN	

dbo.p_Decay_And_Distribute_DEBUG

Created: 11 Jun 2004 Last updated: 11 Jun 2004

Identical to the p_Decay_And_Distribute procedure except that it contains debugging statements.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@TWT_ID	int	IN	

dbo.p_Decay_And_Distribute_IDecay

Created: 13 Sep 2002 Last updated: 13 May 2009

Performs the intermediate decay and distribute calculations for the p_Decay_And_Distribute stored procedure.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@Con_ID	varchar	IN	
@Parent_Type	char	IN	
@Distribution_V_ID	int	IN	

dbo.p_Decay_And_Distribute_IDecay_DEBUG

Created: 31 Jul 2008 Last updated: 31 Jul 2008

Is identical to the p_Decay_And_Distribute_IDecay stored procedure except that it contains debugging statements.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@Con_ID	varchar	IN	
@Parent_Type	char	IN	
@Distribution_V_ID	int	IN	

dbo.p_Decay_And_Distribute_Pop_WTable

Created: 10 Sep 2002 Last updated: 13 May 2009

Populates a working table used by the p_Decay_And_Distribute procedure.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@TWT_ID	int	IN	
@Con_ID	varchar	IN	
@Distribution_V_ID	int	IN	
@Parent_Type	char	IN	

dbo.p_Decay_And_Distribute_Pop_WTable_DEBUG

Created: 30 Jul 2008 Last updated: 30 Jul 2008

Is identical to the p_Decay_And_Distribute_WTable stored procedure except that it contains debugging statements.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@TWT_ID	int	IN	
@Con_ID	varchar	IN	
@Distribution_V_ID	int	IN	
@Parent_Type	char	IN	

dbo.p_Dependent_Calcs

Created: 27 Apr 2000 Last updated: 27 Apr 2000

Update any dependent constituent inventories that are 'calculated' based upon the selected constituent.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@TWT_ID	int	IN	
@Con_ID	varchar	IN	

 **dbo.p_Inventory_Sum_Descrepancies**

Created: 25 Jan 2006 Last updated: 25 Jan 2006

Used by the Inventory Discrepancy report in the BBI ASP.NET application.

dbo.p_Oxalate_Update

Created: 25 May 2004 Last updated: 25 May 2004

If the Oxalate is missing and TOC is present, then insert an Oxalate Tank_Waste_Inventory record.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@TWT_ID	int	IN	

dbo.p_pop_RPD4Tanks

Created: 17 Jun 2002 Last updated: 17 Jun 2002

Used to populate the RPD4TanksInWorkUnit table shown in the BBI ASP.NET PrepForPub form.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@Waste_Site_ID	varchar	IN	
@UserName	varchar	IN	

dbo.p_RadioButton

Created: 08 Mar 2010 Last updated: 08 Mar 2010

Called by the RadioButton form in the BBI ASP.NET application

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@TWT_ID	int	IN	
@ShowStdUnits	bit	IN	

dbo.p_RadioButton_Plus

Created: 25 Jul 2003 Last updated: 25 Jul 2003

Return Lots of data about a give component.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@TWT_ID	int	IN	

dbo.p_Tank_Summary

Created: 10 Jan 2007 Last updated: 10 Jan 2007

Used by the Tank Summary report in the BBI ASP.NET application.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@Waste_Site_ID	varchar	IN	

dbo.p_TankCompareRpt

Created: 29 Aug 2003 Last updated: 29 Aug 2003

Used by the BBI ASP.NET application to compare the BBIM and published inventories for a given tank.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@Waste_Site_ID	varchar	IN	
@Mode	varchar	IN	

dbo.p_Total_OH

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Called by the dbo.p_Total_OH_Calc stored procedure

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@Waste_Site_ID	varchar	IN	
@Total_OH	float	INOUT	

dbo.p_Total_OH_Calc

Created: 29 Aug 2003 Last updated: 29 Aug 2003

Used by the TotalInvOH.aspx form in the BBI application which generates the Total OH Inventory as Calculated by Charge Balance

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@Waste_Site_ID	varchar	IN	

dbo.p_TWC_Add

Created: 18 Jul 2003 Last updated: 18 Jul 2003

Used by the TWC_Add.aspx form. Adds a new Tank_Waste_Component record.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@Waste_Site_ID	nvarchar	IN	
@Waste_Phase	nvarchar	IN	
@Waste_Type	nvarchar	IN	
@Volume	float	IN	
@Vol_Reference	nvarchar	IN	
@Doc_Reference	nvarchar	IN	
@Distribution_V_ID	float	IN	
@Density	float	IN	
@Density_Lower_95_Percent_CL	float	IN	
@Density_Upper_95_Percent_CL	float	IN	
@Weight_Percent_Water	float	IN	
@Component_Effective_Date	nvarchar	IN	
@Oxalate_Fraction_Of_TOC	float	IN	
@Oxalate_Comment	nvarchar	IN	
@TWT_ID	int	INOUT	

dbo.p_TWC_Select

Created: 18 Jul 2003 Last updated: 18 Jul 2003

Used by the TWC_Find.aspx form. Retrieves a Tank_Waste_Component record given its TWT_ID value.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@TWT_ID	int	IN	

dbo.p_TWC_Vectors_Select

Created: 18 Jul 2003 Last updated: 18 Jul 2003

Used by the TWC_Find.aspx form. Retrieves all the vectors associated with this Tank_Waste_Component record.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@TWT_ID	int	IN	

dbo.p_Update_Inventory_RSD

Created: 26 Mar 2002 Last updated: 26 Mar 2002

Updates the Inventory RSD for a Tank Waste Component when Density, Concentration and Volume RSDs are present.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@TWT_ID	int	IN	

dbo.p_Update_RadioButton

Created: 18 Jul 2003 Last updated: 18 Jul 2003

Using the selection from the BBI Radio Button form, update the tank's inventory based upon this selection.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@TWT_ID	int	IN	
@Con_ID	varchar	IN	
@VC_ID	int	IN	

dbo.p_Update_Tank_Inventory_Comment

Created: 27 Nov 2004 Last updated: 27 Nov 2004

Used by the BBI ASP.NET application to update the comment of a tank's inventory record.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@Waste_Site_ID	varchar	IN	
@Con_ID	varchar	IN	
@Comment	varchar	IN	

dbo.p_Vector_Add

Created: 18 Jul 2003 Last updated: 18 Jul 2003

Used in the *Vector_Add.aspx* form in the *BBI* application. Adds a new Vector record.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@Vector_Type	nvarchar	IN	
@Template	nvarchar	IN	
@Waste_Site_ID	nvarchar	IN	
@Vector_Name	nvarchar	IN	
@Density	float	IN	
@Sample_Date	nvarchar	IN	
@Decay_Date_Reported	nvarchar	IN	
@Ref_Document	nvarchar	IN	
@Ref_Table_Name	nvarchar	IN	
@Ref_Table_No	nvarchar	IN	
@Ref_Table_Column_No	nvarchar	IN	
@Density_Comment	nvarchar	IN	
@V_ID	int	INOUT	

dbo.p_Vector_Constituent_Add

Created: 18 Jul 2003 Last updated: 18 Jul 2003

Used in the Vector_Con_Add.aspx form in the BBI application. Adds a new Vector_Constituent record.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@V_ID	int	IN	
@Con_ID	nvarchar	IN	
@Qualifier	nvarchar	IN	
@Reported_Concentration	float	IN	
@Units_Of_Concentration	nvarchar	IN	
@User_Comment	nvarchar	IN	
@Basis	nvarchar	IN	
@VC_ID	int	INOUT	

dbo.p_Vector_Constituent_Import_Excel

Created: 27 Jan 2004 Last updated: 27 Apr 2010

Used in the BBI ASP.NET application to copy the data stored in the Vector_Constituent_Excel_Data table into the Vector_Constituent table.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@V_ID	int	IN	
@UserName	varchar	IN	

dbo.p_Vector_Delete

Created: 25 May 2004 Last updated: 25 May 2004

Used by the *BBI ASP.NET* application.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@V_ID	int	IN	

dbo.p_Vector_Has_Constituents

Created: 25 May 2004 Last updated: 25 May 2004

Used by the *BBI ASP.NET* application.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@V_ID	int	IN	

dbo.p_Vector_List

Created: 18 Jul 2003 Last updated: 18 Jul 2003

Retrieves a list of all vectors in the database.

 **dbo.p_Vector_List_With_Null**

Created: 29 Mar 2004 Last updated: 29 Mar 2004

Used the BBI ASP.NET application. Retrieves a list of all vectors in the database plus a blank record.

dbo.p_Vector_List1

Created: 18 Jul 2003 Last updated: 18 Jul 2003

Used by the BBI Vector_Add form. Retrieves a list of Vectors for this Tank_Waste_Component that are not currently used by the component.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@TWT_ID	int	IN	

dbo.p_Verification_Check

Created: 16 Nov 2004 Last updated: 03 Jun 2010

Used by the Inventory Check report in the BBI ASP.NET application.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@Waste_Site_ID	varchar	IN	

 **dbo.p_Verify_All_Tank_Mass_Balance**

Created: 14 Jan 2005 Last updated: 14 Jan 2005

Used by the Mass Balance Verification report in the BBI ASP.NET application.

dbo.p_Work_Unit_Add

Created: 18 Jul 2003 Last updated: 18 Jul 2003

Used by the BBI Work Unit Add form. Adds a Work Unit record.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@CCB_Log	nvarchar	IN	
@Title	nvarchar	IN	
@Description	nvarchar	IN	
@Justification	nvarchar	IN	
@Impact	nvarchar	IN	
@Comment	nvarchar	IN	
@Waste_Site_ID	nvarchar	IN	
@Last_Updated_By	nvarchar	IN	

dbo.p_Work_Unit_Delete

Created: 18 Jul 2003 Last updated: 18 Jul 2003

Used by the BBI Work Units form. Deletes a Work Unit record.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@WU_ID	int	IN	

dbo.p1_Dependent_Calcs

Created: 10 Jun 2002 Last updated: 12 Mar 2010

Called by the p_Dependent_Calcs stored procedure. Update any dependent constituent inventories that are 'calculated' based upon the selected constituent.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@TWT_ID	int	IN	
@Con_ID1	varchar	IN	
@Con_ID2	varchar	IN	
@Factor	float	IN	

dbo.sp_ColumnHeaders

Created: 19 Nov 2009 Last updated: 19 Nov 2009

Utility routine which output the column headers of the input table or view.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@TableName	varchar	IN	

dbo.spScriptInsert

Created: 29 Dec 2008 Last updated: 29 Dec 2008

Utility routine that creates a script for inserting data into the specified table.

Parameters

<u>PARAMETER NAME</u>	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@table	varchar	IN	
@condition	varchar	IN	

dbo.Tank_Waste_Inventory_Utrig_sp

Created: 30 Aug 2002 Last updated: 30 Aug 2002

Called by the Tank_Waste_Inventory Update trigger. Performes Upper Bounding Estimate calculations.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@Basis_Type	char	IN	

User defined functions

 dbo.Cs137Ci

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Return the Ci of the 137Cs for this Tank Waste Component.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	float	OUT	
@TWT_ID	int	IN	

dbo.GetInvAsCi

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Return the Inventory in Curies of the Constituent for this Tank Waste Component. Used the constituent's specific activity if the inventory happens to be recorded in kg.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	float	OUT	
@TWT_ID	int	IN	
@Con_ID	varchar	IN	

dbo.GetScientificNotation

Created: 10 Nov 2006 Last updated: 10 Nov 2006

Convert the float value in scientific notation using the specified number of decimal places.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	char	OUT	
@inputValue	float	IN	
@decimalPlaces	int	IN	

dbo.InvAsKg

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Return the Inventory in Kg of the Constituent for this Tank Waste Component. Used the constituent's specific activity if the inventory happens to be recorded in curies.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	float	OUT	
@TWT_ID	int	IN	
@Con_ID	varchar	IN	

dbo.mBa137Ci

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Return the Ci of the mBa137 for this Tank Waste Component.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	float	OUT	
@TWT_ID	int	IN	

dbo.OxalateKg

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Return the Kg of the Oxalate for this Tank Waste Component.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	float	OUT	
@TWT_ID	int	IN	

dbo.PlutoniumKg

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Return the Kg of the Plutonium (238Pu, 239Pu, 240Pu, 241Pu, 242Pu) for this Tank Waste Component.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	float	OUT	
@TWT_ID	int	IN	

dbo.Pu238Ci

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Return the Ci of the 238Pu for this Tank Waste Component.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	float	OUT	
@TWT_ID	int	IN	

dbo.Pu238Kg

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Return the Kg of the 238Pu for this Tank Waste Component.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	float	OUT	
@TWT_ID	int	IN	

dbo.Pu239Pu240Ci

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Return the Ci of the [239/240Pu] for this Tank Waste Component.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	float	OUT	
@TWT_ID	int	IN	

dbo.Pu239Pu240Kg

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Return the Kg of the [239/240Pu] for this Tank Waste Component.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	float	OUT	
@TWT_ID	int	IN	

dbo.Pu240Kg

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Return the Dg of the 240Pu for this Tank Waste Component.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	float	OUT	
@TWT_ID	int	IN	

dbo.spMVal

Created: 29 Dec 2008 Last updated: 29 Dec 2008

Required function, called by the spScriptInsert stored procedure.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@val	nvarchar money	OUT IN	

dbo.spVal

Created: 29 Dec 2008 Last updated: 29 Dec 2008

Required function, called by the spScriptInsert stored procedure.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	nvarchar	OUT	
@str	nvarchar	IN	

dbo.Sr90Ci

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Return the Ci of the 90Sr for this Tank Waste Component.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	float	OUT	
@TWT_ID	int	IN	

dbo.Th232Kg

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Return the Kg of the 232Th for this Tank Waste Component.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	float	OUT	
@TWT_ID	int	IN	

dbo.ThKg

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Return the Kg of the Th for this Tank Waste Component.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	float	OUT	
@TWT_ID	int	IN	

dbo.TOCKg

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Return the Kg of the TOC for this Tank Waste Component.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	float	OUT	
@TWT_ID	int	IN	

dbo.U235Kg

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Return the Kg of the 235U for this Tank Waste Component.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	float	OUT	
@TWT_ID	int	IN	

dbo.U238Kg

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Return the Kg of the 238U for this Tank Waste Component.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	float	OUT	
@TWT_ID	int	IN	

dbo.UraniumKg

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Return the Kg of the Uranium (232U, 233U, 234U, 235U, 236U, 238U) for this Tank Waste Component.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	float	OUT	
@TWT_ID	int	IN	

dbo.UTOTALKg

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Return the Kg of the UTOTAL for this Tank Waste Component.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	float	OUT	
@TWT_ID	int	IN	

dbo.VerifyCs137

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Returns a table row of (Result, Numerator, Demoninator, Criteria, Grade), with the results of whether the mBa137 / Cs137 Ci ratio for this Tank Waste Component is between 93.4 - 95.4%.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@TWT_ID	int	IN	
@IntermediateNum	float	IN	
@IntermediateDenom	float	IN	

dbo.VerifyEu154

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Returns a table row of (Result, Numerator, Demoninator, Criteria, Grade), with the results of whether the 154Eu / 155Eu Ci ratio for this Tank Waste Component is between 0.5 - 4.0.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@TWT_ID	int	IN	
@IntermediateNum	float	IN	
@IntermediateDenom	float	IN	

dbo.VerifyMassBalance

Created: 31 Jan 2005 Last updated: 31 Jan 2005

Returns a table row of (Result, MassBalance, Criteria, Grade), with the results of whether the Mass Balance for this Tank Waste Component is between 85 - 115%.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@TWT_ID	int	IN	
@Intermediate	float	IN	

dbo.VerifyOxalate

Created: 22 Nov 2004 Last updated: 22 Nov 2004

Returns a table row of (Result, Numerator, Demoninator, Criteria, Grade), with the results of whether the Oxalate / TOC Kg ratio for this Tank Waste Component is less than 101%.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@TWT_ID	int	IN	
@IntermediateNum	float	IN	
@IntermediateDenom	float	IN	

dbo.VerifyPlutonium

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Returns a table row of (Result, Numerator, Demoninator, Criteria, Grade), with the results of whether the 240Pu / Pu kg ratio for this Tank Waste Component is between 1 - 8%.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@TWT_ID	int	IN	
@IntermediateNum	float	IN	
@IntermediateDenom	float	IN	

dbo.VerifyPu238

Created: 22 Nov 2004 Last updated: 22 Nov 2004

Returns a table row of (Result, Numerator, Demoninator, Criteria, Grade), with the results of whether the 238Pu/ [239/240Pu] Ci ratio for this Tank Waste Component is between 0.3 - 10%.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@TWT_ID	int	IN	
@IntermediateNum	float	IN	
@IntermediateDenom	float	IN	

dbo.VerifyTh232

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Returns a table row of (Result, Numerator, Demoninator, Criteria, Grade), with the results of whether the Th / Th232 kg ratio for this Tank Waste Component is between 99 - 101%.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@TWT_ID	int	IN	
@IntermediateNum	float	IN	
@IntermediateDenom	float	IN	

dbo.VerifyU235

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Returns a table row of (Result, Numerator, Demoninator, Criteria, Grade), with the results of whether the 235U / U kg ratio for this Tank Waste Component is less than 1%.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@TWT_ID	int	IN	
@IntermediateNum	float	IN	
@IntermediateDenom	float	IN	

dbo.VerifyU238

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Returns a table row of (Result, Numerator, Demoninator, Criteria, Grade), with the results of whether the 238U / U kg ratio for this Tank Waste Component is between 99 - 100%.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@TWT_ID	int	IN	
@IntermediateNum	float	IN	
@IntermediateDenom	float	IN	

dbo.VerifyUTOTAL

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Returns a table row of (Result, Numerator, Demoninator, Criteria, Grade), with the results of whether the UTOTAL / U kg ratio for this Tank Waste Component is between 99 - 100%.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@TWT_ID	int	IN	
@IntermediateNum	float	IN	
@IntermediateDenom	float	IN	

dbo.VerifyY90

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Returns a table row of (Result, Numerator, Demoninator, Criteria, Grade), with the results of whether the 90Y / 90Sr Ci ratio for this Tank Waste Component is between 99 - 101%.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
@TWT_ID	int	IN	
@IntermediateNum	float	IN	
@IntermediateDenom	float	IN	

dbo.Y90Ci

Created: 16 Nov 2004 Last updated: 16 Nov 2004

Return the Ci of the Y90 for this Tank Waste Component.

Parameters

PARAMETER NAME	DATA TYPE	MODE	DESCRIPTION
	float	OUT	
@TWT_ID	int	IN	

CRTT Database Documentation

Server name: twins

Database name: CRTT

Documentation date: 7/29/2010

Database size: 46.25 MB

Database description:

Table summary

	Name	Data size	Index size	Row Count
	dbo.Bug_Resolution_Change_History	88 KB	104 KB	1412
	dbo.Change_Request_Status_Notes	896 KB	72 KB	328
	dbo.Change_Request_Tasks	1128 KB	592 KB	2371
	dbo.ChangeRequest	1040 KB	24 KB	411
	dbo.Clients	8 KB	24 KB	31
	dbo.CR_Resolution_Change_History	80 KB	88 KB	778
	dbo.Developers	8 KB	24 KB	19
	dbo.Project_Developers	16 KB	24 KB	16
	dbo.Projects	8 KB	24 KB	2
	dbo.ReportedBugs	1600 KB	104 KB	572
	dbo.ReportedBugs_Status_Notes	1520 KB	72 KB	560
	dbo.ReportedBugs_Tasks	2576 KB	552 KB	4825
	dbo.Required_Tasks	16 KB	40 KB	22
	dbo.ResolutionState	8 KB	40 KB	9
	dbo.Severity_Level	8 KB	24 KB	5

15 table(s)

TABLES

dbo.Bug_Resolution_Change_History

ever the resolution or state of a bug is changed, then its previous and new values are recorded here. Enables a search over time to see what has changed (been worked on).

I	F	P	Column name	Data type	Nulls	Default	Description
			BRCH_ID	int	NO		Unique Identifier of a Bug Resolution Change History record
			Bug_ID	int	NO		Unique identifier of the Bug.
			Old_Res_ID	int	YES		Link to the resolution state prior to the change.
			New_Res_ID	int	NO		Link to the resolution state after the change.
			Changed_On	datetime	NO		Date of the change in resolution or state.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
aaaaaBug_Resolution_Change_History_2_PK	BRCH_ID	ASC	Yes	NONCLUSTERED
i1_BRCH	BRCH_ID	ASC	Yes	NONCLUSTERED

dbo.Change_Request_Status_Notes

If a developer wishes to record any information about the change request, it is recorded here.

I	F	P	Column name	Data type	Nulls	Default	Description
			CRS_ID	int	NO		Unique Identifier for Change Request Status Notes
			CR_ID	int	YES		Unique Identifier for the Change Request record.
			Status_On	datetime	YES		Date that the record was entered into the database. Automatically generated by the database.
			Status_By	nvarchar(40)	YES		Who entered this status.
			Notes	ntext	YES		The notes recorded by the user.
			Link_To_Doc	nvarchar(255)	YES		Not used.

Indexes

	Index name	Column name	Sort direction	Is unique	Index type
aaaaa	Change_Request_Status_Notes_PK	CRS_ID	ASC	Yes	NONCLUSTERED
	Change_RequestChange_Request_Stat	CR_ID	ASC		NONCLUSTERED
	CR_ID	CR_ID	ASC		NONCLUSTERED
	CRS_ID	CRS_ID	ASC		NONCLUSTERED

Triggers

Trigger name:	dbo.Change_Request_Status_Notes_I	Created on:	06 Dec 2002
Trigger type:	INSERT	Trigger active:	Yes
Trigger name:	dbo.Change_Request_Status_Notes_U	Created on:	06 Dec 2002
Trigger type:	UPDATE	Trigger active:	Yes

Trigger name: dbo.ChangeRequest_ITrig
Trigger type: INSERT

Created on: 06 Dec 2002
Trigger active: Yes

Trigger name: dbo.ChangeRequest_UTrig
Trigger type: UPDATE

Created on: 06 Dec 2002
Trigger active: Yes

dbo.CR_Resolution_Change_History

Whenever the resolution or state of a change request is changed, then its previous and new values are recorded here. Enables a search over time to see what has changed (been worked on).

I	F	P	Column name	Data type	Nulls	Default	Description
			CRRCH_ID	int	NO		Unique Identifier for Change Request Resolution Change History
			CR_ID	int	NO		Unique Identifier for the Change Request record.
			Old_Res_ID	int	YES		Link to the resolution state prior to the change.
			New_Res_ID	int	NO		Link to the resolution state after he change.
			Changed_On	datetime	NO		Date that the resolution or state changes.

Indexes

	Index name	Column name	Sort direction	Is unique	Index type
	aaaaaCR_Resolution_Change_History2_PK	CRRCH_ID	ASC	Yes	NONCLUSTERED
	i1_CRRCH	CRRCH_ID	ASC	Yes	NONCLUSTERED

dbo.Project_Developers

Divides a developer into either the TWINS or DIALOG project, or into both.

I F P	Column name	Data type	Nulls	Default	Description
	PD_ID	int	NO		Unique Project Developer record identifier.
	Project_ID	int	NO		Unique identifier of the project (TWINS or DIALOG).
	Developer_Name	nvarchar(40)	NO		Name of the developer.
	IsSuperUser	char(1)	NO		Indicates whether this individual has Super User privileges. Super User privileges allow this user to edit or delete not only their own data, but data owned/assigned to others.
	IsProjMgr	char(1)	NO		Indicates whether this individual is a project manager. Only one record in the developers table can have its IsProjMgr field set to Y. This is controlled by the update/insert triggers. When new Change Requests or Bugs are created, the insert triggers assign Formal Testing Required (in the case of Change Requests) and Acceptance Testing Required (in the case of Bugs) tasks to the project manager.
	IsCurrent	char(1)	NO		Indicates whether or not the record is current. Useful for creating list boxes where only the current records should be displayed.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_id	PD_ID	ASC	Yes	NONCLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__Project_D__Devel__19DFD96B	Developer_Name	aaaaaDevelopers_PK (dbo.Developers)	
FK__Project_D__Proje__18EBB532	Project_ID	PK__Projects__17036CC0 (dbo.Projects)	

dbo.Projects

There are two projects that use the Change Request and Bug Tracking Tool: TWINS and DIALOG. This table enables the change requests and bugs to be separated into their own distinct project.

I	F	P	Column name	Data type	Nulls	Default	Description
			Project_ID	int	NO		Unique record identifier.
			Project_Acronym	varchar(10)	NO		Project acronym.
			Project_Name	varchar(100)	NO		Full name of the project.
			IsCurrent	char(1)	NO		Indicates whether or not the record is current. Useful for creating list boxes where only the current records should be displayed.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
PK__Projects__17036CC0	Project_ID	ASC	Yes	CLUSTERED

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.Project_Developers	FK__Project_D__Proje__18EBB532	PK__Projects__17036CC0

dbo.ReportedBugs

Contains information about any bugs, enhancements or glitches found for any TWINS application. Used as a project management tool to categorize, prioritize and assign responsibility and track the bug.

I	F	P	Column name	Data type	Nulls	Default	Description
			Bug_ID	int	NO		Unique record identifier.
			Title	nvarchar(255)	YES		Title of the bug.
			Description	ntext	YES		Information provided by the originator.
			Reported_By	nvarchar(50)	YES		Person who reported the bug.
			Reported_On	datetime	YES		Date that the bug was reported.
			Severity	nvarchar(12)	YES		Priority of the bug. High, Medium, Low.
			Assigned_To	nvarchar(30)	YES		The developer responsible for fixing the bug.
			Res_ID	smallint	YES	(14)	Resolution ID. Points to the current resolution or state of the bug.
			Close Date	datetime	YES		Completion date.
			upsized_ts	timestamp	YES		Not used.
			Due_Date	datetime	YES		Date that the bug should be fixed.
			Impact	varchar(6)	YES	('Low')	High, Medium or Low. How it affects the users.
			Risk	varchar(6)	YES	('Low')	High, Medium or Low. The risk of incorrectly fixing the bug due to its difficulty.
			Project_ID	int	NO		Which project this bug belongs to: TWINS or DIALOG.
			Reviewer	varchar(40)	YES		Person responsible for testing the changes made in fixing the bug.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
aaaaaReportedBugs_PK	Bug_ID	ASC	Yes	NONCLUSTERED
CR_ID1	Bug_ID	ASC		NONCLUSTERED
Res_ID	Res_ID	ASC		NONCLUSTERED

Check constraints

Check name	Column name	Expression
ck_BugImpact	Impact	([Impact] = 'Low' or [Impact] = 'High')
ck_BugRisk	Risk	([risk] = 'Low' or [risk] = 'High')

Triggers

Trigger name:	dbo.ReportedBugs_DTrig	Created on:	06 Dec 2002
Trigger type:	DELETE	Trigger active:	Yes
Trigger name:	dbo.ReportedBugs_ITrig	Created on:	06 Dec 2002
Trigger type:	INSERT	Trigger active:	Yes
Trigger name:	dbo.ReportedBugs_UTrig	Created on:	06 Dec 2002
Trigger type:	UPDATE	Trigger active:	Yes

dbo.ReportedBugs_Tasks

When a bug is first generated, the standard required tasks are created for the bug (refer to table Required_Tasks). These tasks can be deleted (they may not make sense), and/or, new tasks can be added.

I	F	P	Column name	Data type	Nulls	Default	Description
			BT_ID	int	NO		Unique Identifier of a Reported Bug Task
			Bug_ID	int	NO	(0)	Unique identifier of the bug.
			Req_ID	int	YES	(0)	If this was a required task that was created upon creation of the bug, this field points to the task record.
			Description	varchar(120)	YES		Description of the Task.
			Completed_On	datetime	YES		Date that the task was completed.
			Completion_By	nvarchar(30)	YES		Who completed the task.
			Not_Required_Reason	varchar(255)	YES		Text field that allows the user to explain why this task is not required for this bug.
			Comment	text	YES		Any information relating to this task.
			sort_order	int	YES		Used to sort the tasks within a bug.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
aaaaaReportedBugs_Tasks_PK	BT_ID	ASC	Yes	NONCLUSTERED
BT_ID	BT_ID	ASC		NONCLUSTERED
CR_ID	Bug_ID	ASC		NONCLUSTERED
Req_ID	Req_ID	ASC		NONCLUSTERED

dbo.Required_Tasks

Used by the ChangeRequest and ReportedBugs insert trigger to automatically create the tasks when a new record is created.

I	F	P	Column name	Data type	Nulls	Default	Description
			Req_ID	int	NO		Unique record identifier.
			Order_By_Field	smallint	YES (0)		Used to sort the required task records.
			Description	varchar(120)	YES		Description of the of the required task.
			Project_ID	int	NO		Which project this task belongs.
			Assign_To_Reviewer	char(1)	NO ('N')		Flag indicating whether this task is to be assigned to the initial reviewer.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
aaaaaRequired_Tasks_PK	Req_ID	ASC	Yes	NONCLUSTERED
Req_ID	Req_ID	ASC		NONCLUSTERED

dbo.ResolutionState

Contains all the possible resolutions and their associated states for a change request or reported bug.

I	F	P	Column name	Data type	Nulls	Default	Description
			Res_ID	int	NO		Unique Identifier of a Resolution State
			Resolution	nvarchar(50)	YES		On of Cancelled, Completed, Duplicate, Waiting on others, Ready to Close, On Hold, or Waiting on Client Acceptance Test.
			State	nvarchar(6)	YES		Either Open or Closed.
			Order_By_Field	smallint	YES	(0)	Used for sorting the resolutions.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
aaaaaResolutionState_PK	Res_ID	ASC	Yes	NONCLUSTERED
Res_ID	Res_ID	ASC		NONCLUSTERED

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.ChangeRequest	ChangeRequest_FK00	aaaaaResolutionState_PK

TCD Database Documentation

Server name: twins

Database name: TCD

Documentation date: 7/28/2010

Database size: 10499.4 MB

Database description:

Table summary

	Name	Data size	Index size	Row Count
	dbo.agg_level	8 KB	32 KB	37
	dbo.agg_level_definitions	8 KB	8 KB	25
	dbo.alpha_beta_gamma_emitter	8 KB	32 KB	3
	dbo.analysis_class	8 KB	32 KB	9
	dbo.analysis_method	144 KB	136 KB	647
	dbo.analysis_method_group	24 KB	32 KB	334
	dbo.analysis_result	318264 KB	87624 KB	1516283
	dbo.analysis_result_keys	143560 KB	640512 KB	1516291
	dbo.analysis_result_keys_multiple	0 KB	0 KB	0
	dbo.ANALYSIS_RESULT_MOD	548920 KB	102552 KB	352965
	dbo.analysis_result_to_load	952 KB	184 KB	0
	dbo.Archive_AutoStatAnalysisResultData	54608 KB	9576 KB	204198
	dbo.Archive_AutoStatHeader	96 KB	16 KB	403
	dbo.Archive_AutoStatHeaderOptions	2592 KB	16 KB	33588
	dbo.Archive_AutoStatResults	10168 KB	8 KB	27987
	dbo.Archive_AutoStatSpecialCases	9944 KB	8 KB	59684
	dbo.AutoStatHeader	64 KB	48 KB	169
	dbo.AutoStatHeaderOptions	1200 KB	1544 KB	15254
	dbo.AutoStatIDsProcessed	64 KB	8 KB	0
	dbo.AutoStatOverview	128 KB	8 KB	183
	dbo.AutoStatResults	4136 KB	1328 KB	9599
	dbo.AutoStatSpecialCases	4984 KB	712 KB	29075
	dbo.bottom_shape	8 KB	32 KB	2
	dbo.con_analysis_method_group	248 KB	240 KB	4516
	dbo.con_type	8 KB	32 KB	3
	dbo.constituent	216 KB	176 KB	1135
	dbo.constituent_group	8 KB	32 KB	22
	dbo.constituent_group_member	144 KB	16 KB	2173
	dbo.constituent_mod	712 KB	24 KB	2440
	dbo.constituent_synonym	408 KB	840 KB	4578
	dbo.constituent_synonym_mod	640 KB	16 KB	5089
	dbo.count_TVD_records_loaded	16 KB	8 KB	445
	dbo.cron_status	665976 KB	194744 KB	9696326
	dbo.data_status	8 KB	8 KB	4
	dbo.Debug_Setting	16 KB	88 KB	201
	dbo.device_type	8 KB	16 KB	10
	dbo.dqo_exceeding_limits	1048 KB	8 KB	912
	dbo.ESP_data	1896 KB	40 KB	19499
	dbo.ESP_metadata	16 KB	32 KB	177
	dbo.integrity	8 KB	32 KB	5
	dbo.isolation_status	8 KB	32 KB	8
	dbo.laboratory	8 KB	32 KB	7
	dbo.log_changes	640464 KB	252712 KB	2609962
	dbo.log_insert_counts	56 KB	8 KB	397
	dbo.means_and_variances_detail	29400 KB	488 KB	41221
	dbo.means_and_variances_header	40 KB	48 KB	561
	dbo.means_variances_detail_mod	47840 KB	152 KB	65227
	dbo.operator	8 KB	32 KB	5
	dbo.parent_table	8 KB	32 KB	4

dbo.phase	8 KB	32 KB	8
dbo.physical_state	8 KB	40 KB	3
dbo.privitization	8 KB	8 KB	3
dbo.project	8 KB	16 KB	17
dbo.project_type	8 KB	8 KB	3
dbo.qa_level	8 KB	24 KB	2
dbo.qa_type	8 KB	32 KB	14
dbo.qc_limits_am	112 KB	24 KB	913
dbo.qualifier_type	8 KB	8 KB	2
dbo.qualifiers	8 KB	8 KB	18
dbo.reporting_day	8 KB	32 KB	7
dbo.ResolveLog	7312 KB	1872 KB	33606
dbo.result_type	8 KB	32 KB	91
dbo.review_status	8 KB	32 KB	9
dbo.samp_attr	656 KB	128 KB	3340
dbo.samp_attr_lkup	8 KB	16 KB	19
dbo.sample	15016 KB	640 KB	32308
dbo.sample_analysis	8016 KB	11080 KB	57638
dbo.sample_analysis_to_load	96 KB	48 KB	0
dbo.sample_description	39800 KB	7376 KB	33139
dbo.sample_description_to_load	80 KB	40 KB	0
dbo.sample_no_proj_id	0 KB	0 KB	0
dbo.sample_number_master	3880 KB	1056 KB	59590
dbo.sample_portion	8 KB	32 KB	13
dbo.sample_relation	5384 KB	1344 KB	55747
dbo.sample_relation_to_load	72 KB	72 KB	0
dbo.sample_type	8 KB	32 KB	4
dbo.sampling_event	344 KB	280 KB	2001
dbo.sampling_event_denorm	8840 KB	24 KB	4060
dbo.sampling_event_to_load	56 KB	40 KB	0
dbo.source_sample	67768 KB	0 KB	60406
dbo.spike_recoveries_outofrange	4640 KB	21912 KB	22454
dbo.state	8 KB	32 KB	54
dbo.static_analysis_result_units	8 KB	16 KB	46
dbo.statistician_notify	0 KB	0 KB	0
dbo.statistician_notify_armod	0 KB	0 KB	0
dbo.status_flag	8 KB	32 KB	2
dbo.std_constituent_units	8 KB	16 KB	8
dbo.std_property_units	16 KB	16 KB	219
dbo.synonym_group_id	8 KB	32 KB	16
dbo.t_data_loaded_status	32 KB	16 KB	192
dbo.tank_core	24 KB	32 KB	317
dbo.tank_core_segment	800 KB	272 KB	2247
dbo.Tank_Farm	8 KB	16 KB	18
dbo.Tank_Farm_Area	8 KB	16 KB	2
dbo.tank_supernate_sample	504 KB	24 KB	1619
dbo.tank_surface_sample	72 KB	16 KB	154
dbo.tank_type	8 KB	16 KB	8
dbo.tank_use	8 KB	32 KB	5
dbo.tbl_SACS_waste_Site_id	24 KB	8 KB	177
dbo.tblLimits	88 KB	232 KB	487
dbo.tblNew_Mean_Vector_Header	304 KB	24 KB	303
dbo.tblNew_Vector_Choices	2592 KB	72 KB	12994

 dbo.tblNew_Vector_Choices1	736 KB	8 KB	7809
 dbo.TblNew_Vector_Counts	2496 KB	72 KB	13987
 dbo.TblNew_Vector_Status	8 KB	24 KB	4
 dbo.tblResolve_Tanks	64 KB	24 KB	46
 dbo.tblResolve_Users	16 KB	24 KB	36
 dbo.TblSendMailTo	8 KB	24 KB	13
 dbo.transaction_type	8 KB	32 KB	4
 dbo.transfers	1368 KB	48 KB	5818
 dbo.transfers_denorm	1760 KB	48 KB	5553
 dbo.unit_conversion	8 KB	16 KB	86
 dbo.units	8 KB	32 KB	102
 dbo.validation_status	8 KB	32 KB	2
 dbo.ventilation	8 KB	32 KB	2
 dbo.verification_process	8 KB	32 KB	2
 dbo.waste_tank	272 KB	32 KB	398
 dbo.waste_tank_group	8 KB	32 KB	2
 dbo.waste_tank_group_member	32 KB	32 KB	397
 dbo.WasteDescConstituents	8 KB	8 KB	45
 dbo.watch_list_status	8 KB	32 KB	9
 dbo.XREF_WASTETYPE_PHASEASSIGNMENT	16 KB	8 KB	157

122 table(s)

dbo.agg_level_definitions

Same as the `agg_level` table, but the definition is more detailed.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			agg_level_type	varchar(60)	NO		
			description	varchar(1000)	YES		Brief narrative describing a specific instance in this database table.
			sort_order	int	YES		Used for sorting the records in this table or view.
			agg_level_definitions_id	int	NO		

dbo.alpha_beta_gamma_emitter

Lookup table for alpha/beta/gamma emitters

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			alpha_beta_gamma_emitter	varchar(10)	NO		Type of emitter -- ALPHA, BETA, or GAMMA.
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_alpha_beta_gamma_emitter	alpha_beta_gamma_emitter	ASC	Yes	CLUSTERED
PK__alpha_beta_gamma__49515770	alpha_beta_gamma_emitter	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.constituent	FK__constitue__alpha__44579829	i_alpha_beta_gamma_emitter

dbo.analysis_class

Class of the analysis. Used by the TCDLoader2 ASP.NET application.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			analysis_class	varchar(10)	NO		A class or category of analysis.
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_analysis_class	analysis_class	ASC	Yes	CLUSTERED
PK__analysis_class__4A457BA9	analysis_class	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.analysis_method	FK__analysis__analy__22028025	i_analysis_class

dbo.analysis_method

Analysis methods used in the laboratories for waste tank data. Used by the TCDLoader2 ASP.NET application.

I	F	P	Column name	Data type	Nulls	Default	Description
			analysis_method_id	int	NO		Analysis method id includes those properties that identify the class, procedure, and preparation that describe how the results were obtained and analyzed for the sample. Analysis method IDs trace back to lab procedures such as LA-505-101 or PNL-ALO-103.
			method_group_id	varchar(30)	NO		Name assigned to the general group of laboratory analytical methods to which the current method belongs. Examples include: ICP:A and AA:F.
			method_name	varchar(100)	NO		Name of analysis method.
			lab_code	varchar(10)	NO		The name of the laboratory that analyzed the sample.
			analysis_class	varchar(10)	NO		A class or category of analysis.
			analysis_procedure	varchar(20)	NO		The identification assigned to a laboratory procedure.
			preparation_procedure	varchar(20)	NO		The code identifying a specific laboratory prep procedure.
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.
			recent_change	char(1)	YES		A flag indicating as recent change to records has taken place.

Indexes

	Index name	Column name	Sort direction	Is unique	Index type
	i1_analysis_method	analysis_method_id	ASC	Yes	CLUSTERED
	i2_analysis_method	lab_code	ASC	Yes	NONCLUSTERED
	i2_analysis_method	analysis_procedure	ASC	Yes	NONCLUSTERED
	i2_analysis_method	preparation_procedure	ASC	Yes	NONCLUSTERED
	i3_analysis_method	method_group_id	ASC		NONCLUSTERED
	PK_analysis_method__4B399FE2	analysis_method_id	ASC	Yes	NONCLUSTERED

Foreign keys

	Constraint name	Column name	Reference	Description
	FK_analysis__analy__22028025	analysis_class	i_analysis_class ()	
	FK_analysis__metho__23EAC897	method_group_id	i_analysis_method_group ()	
	FK_analysis__lab_c__22F6A45E	lab_code	i_laboratory ()	

Referencing tables

	Table name	Foreign key	Primary key or unique constraint
	dbo.sample_analysis	FK_sample_an__analy__64C467BB	i1_analysis_method

Triggers

Trigger name: dbo.t_analysis_method_del
 Trigger type: DELETE
 Created on: 14 Jul 2006
 Trigger active: Yes

Trigger name: dbo.t_analysis_method_ins_upd
 Trigger type: INSERT, UPDATE
 Created on: 14 Jul 2006
 Trigger active: Yes

dbo.analysis_result

Contains the measured values of tank waste sample data, and its associate meta data.

I F P	Column name	Data type	Nulls	Default	Description
	analysis_result_id	int	NO		Unique sequential number identifier to a single analytical record within the database.
	con_id	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
	analysis_date_time	datetime	YES		Analysis date and time
	result_type	varchar(20)	NO		Description of the type of results the analysis produced. Examples include: PRIMARY, PRIMARY#2
	rpt_value	float	YES		Reported value of the which may be recorded in different units than standard units.
	rpt_value_units	varchar(10)	NO		Units in which the value was originally reported, which may or may not be the same as the standard units.
	uncertainty_value	float	YES		Uncertainty value for the DQO criteria
	uncertainty_value_units	varchar(10)	YES		Units associated with the uncertainty value for the DQO criteria.
	std_value	float	YES		Inventory value reported in standard units. (The inventory value in standard units is reported in the second column of the best-basis inventory table.)
	std_value_units	varchar(10)	YES		Common units of measure: kilograms (kg) for chemicals and curies (Ci) for radionuclides.
	qualifiers	varchar(20)	YES		Indicators of the quality of the value. These codes can include data qualifiers, report flags, and validation flags. See Qualifier Definitions for more information.
	change_indicator	char(1)	YES		A flag to indicate that a change has occurred.
	verification_process	varchar(30)	YES		Identification assigned to the verification_process.
	validation_status	varchar(30)	YES		Validation status (unvalidated or validated) of an analysis result record.
	comment	varchar(255)	YES		Comment
	load_date_time	datetime	YES		The date and time that the record was loaded into the TCD.
	last_modified_date_time	datetime	YES		Date on which the corresponding change record was entered into the TCD. This same date is applied to each analyte changed as part of the respective change log. Once the data has been entered into TCD, the change is immediately visible via the queries available through the TWINS2 interface.
	access_level	tinyint	NO		Access control mechanism for data in database table.
	review_status	varchar(30)	YES		Review status. Current values are: REVIEWED, REVIEWED: SEE NOTE, CHECKING NOT APPLIED, DATA SUSPECT: SEE COMMENT, REVIEWED: NOT FOR INVENTORY and DATA RESOLUTION IN PROGRESS. See Review Status Caveats for more information.
	reviewer	varchar(20)	YES		Reviewer is the name of the individual who reviewed the data or "ComputerCheck" indicating data was compared against a defined set of criteria resulting in the data being flagged as possibly suspect.
	review_date	datetime	YES		Date on which the review was performed.
	review_comment	varchar(2000)	YES		Comment added by reviewer of this analysis result record.
	check_status	varchar(30)	YES		The status of checking applied to the record.
	check_comment	varchar(250)	YES		Comment about check status.
	check_date	datetime	YES		Date of checking.
	approval_date	datetime	YES		The date on which something was approved.
	sample_analysis_id	int	YES		Identity field for the sample_analysis table.
	qc_flag_computer_determined	varchar(20)	YES		QC flag as determined by computer review. Used for AutoTCR for data prior to lab implementation of QC flags.

	retention_time	float	YES
	qa_level	varchar(30)	YES
	sample_data_file_name	varchar(255)	YES

Used for TICs, display in minutes. This field is applicable to Primary, Duplicate, (TriPLICATE if available), and Blank results only.
Opportunistic or Requested

Indexes

Index name	Column name	Sort direction	Is unique	Index type
I_analysis_result	sample_analysis_id	ASC		NONCLUSTERED
i1_analysis_result	analysis_result_id	ASC	Yes	NONCLUSTERED
i2_analysis_result	con_id	ASC		NONCLUSTERED
i2_analysis_result	sample_analysis_id	ASC		NONCLUSTERED
i3_analysis_result	review_status	ASC		NONCLUSTERED
i4_analysis_result	con_id	ASC		NONCLUSTERED
i6_analysis_result	sample_analysis_id	ASC		NONCLUSTERED
i6_analysis_result	con_id	ASC		NONCLUSTERED
i6_analysis_result	result_type	ASC		NONCLUSTERED
i6_analysis_result	access_level	ASC		NONCLUSTERED
index_reviewer	reviewer	ASC		NONCLUSTERED
index_sample_analysis_id	sample_analysis_id	ASC		NONCLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
fk_uncertainty_value_units	uncertainty_value_units	i_units ()	
fk_qa_level	qa_level	PK__qa_level__2C00EBAB (dbo.qa_level)	
fk_sample_analysis_id	sample_analysis_id	i_sample_analysis ()	
fk_rpt_value_units	rpt_value_units	i_units ()	
fk_verification_process	verification_process	i_verification_process ()	
fk_review_status	review_status	i_review_status ()	
fk_constituent	con_id	i_constituent ()	
fk_result_type	result_type	i_result_type ()	
fk_validation_status	validation_status	i_validation_status ()	

Triggers

Trigger name:	dbo.t_analysis_result_ins_upd	Created on:	31 Aug 2006
Trigger type:	INSERT, UPDATE	Trigger active:	Yes
Trigger name:	dbo.t_analysis_result_del	Created on:	31 Aug 2006
Trigger type:	DELETE	Trigger active:	Yes

dbo.analysis_result_keys

Lookup table referenced by several TCD stored procedures.

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
 i	analysis_result_keys_id	int	NO		Unique record identifier.
 i	combo_key	varchar(600)	YES		A concatenation of samp_num, lab_analysis_procedure, primary_sample_Preperation, lab, con_id, result_type and comment
 i	source	varchar(40)	YES		A code indicating the source from which waste was transferred
	analysis_result_to_load_id	int	YES		

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_id	analysis_result_keys_id	ASC	Yes	NONCLUSTERED
i_key	combo_key	ASC		NONCLUSTERED
i_source	source	ASC		NONCLUSTERED

dbo.analysis_result_keys_multiple

Lookup table referenced by several TCD stored procedures.

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
 ix	combo_key	varchar(600)	YES		A concatenation of samp_num, lab_analysis_procedure, primary_sample_Preperation, lab, con_id, result_type and comment

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_combo	combo_key	ASC	Yes	NONCLUSTERED

dbo.ANALYSIS_RESULT_MOD

Records some of the data modifications performed on the analysis_result data. Look in the log_changes table for other modification to the data.

I F P	Column name	Data type	Nulls	Default	Description
	change_date_time	datetime	NO		The date and time that a change was made to the record.
	analysis_result_id	int	NO		Unique sequential number identifier to a single analytical record within the database.
	change_action	varchar(8)	NO		The type of change that was made to an analysis_result record - delete or update.
	change_reason	varchar(255)	NO		Field that allows the data entry person to enter a reason for the change that was made to the original record.
	change_user_name	varchar(8)	NO		The user_name of the person who changed the record.
	samp_num	varchar(12)	NO		Unique alpha-numeric ID assigned to identify a sample in the database.
	analysis_method_id	int	NO		Analysis method id includes those properties that identify the class, procedure, and preparation that describe how the results were obtained and analyzed for the sample. Analysis method IDs trace back to lab procedures such as LA-505-101 or PNL-ALO-103.
	lab_code	varchar(10)	NO		The name of the laboratory that analyzed the sample.
	con_id	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
	sample_dilution_key	tinyint	NO		Sample dilution key
	analysis_date_time	datetime	YES		Analysis date and time
	result_type	varchar(20)	NO		Description of the type of results the analysis produced. Examples include: PRIMARY, PRIMARY#2
	rpt_value	float	YES		Reported value of the which may be recorded in different units than standard units.
	rpt_value_units	varchar(10)	NO		Units in which the value was originally reported, which may or may not be the same as the standard units.
	uncertainty_value	float	YES		Uncertainty value for the DQO criteria
	uncertainty_value_units	varchar(10)	YES		Units associated with the uncertainty value for the DQO criteria.
	std_value	float	YES		Inventory value reported in standard units. (The inventory value in standard units is reported in the second column of the best-basis inventory table.)
	std_value_units	varchar(10)	YES		Common units of measure: kilograms (kg) for chemicals and curies (Ci) for radionuclides.
	qualifiers	varchar(20)	YES		Indicators of the quality of the value. These codes can include data qualifiers, report flags, and validation flags. See /dictionary/Qualifiers.asp for more information.
	change_indicator	char(1)	YES		A flag to indicate that a change has occurred.
	verification_process	varchar(30)	YES		Identification assigned to the verification_process.
	validation_status	varchar(30)	YES		Validation status (unvalidated or validated) of an analysis result record.
	comment	varchar(255)	YES		Comment
	load_date_time	datetime	YES		The date and time that the record was loaded into the TCD.
	last_modified_date_time	datetime	YES		Date on which the corresponding change record was entered into the TCD. This same date is applied to each analyte changed as part of the respective change log. Once the data has been entered into TCD, the change is immediately visible via the queries available through the TWINS2 interface.
	access_level	tinyint	NO		Access control mechanism for data in database table.

review_status	varchar(30)	YES	Review status. Current values are: REVIEWED, REVIEWED: SEE NOTE, CHECKING NOT APPLIED, DATA SUSPECT: SEE COMMENT, REVIEWED: NOT FOR INVENTORY and DATA RESOLUTION IN PROGRESS. See /dictionary/ReviewStatusCaveats.asp >Review Status Caveats for more information.
reviewer	varchar(50)	YES	Reviewer is the name of the individual who reviewed the data or "ComputerCheck" indicating data was compared against a defined set of criteria resulting in the data being flagged as possibly suspect.
review_date	datetime	YES	Date on which the review was performed.
review_comment	varchar(2000)	YES	Comment added by reviewer of this analysis result record.
 waste_site_id	varchar(30)	NO	Tank Identifier, using the format of 241-Tank Farm-Tank #.
method_group_id	varchar(30)	YES	Name assigned to the general group of laboratory analytical methods to which the current method belongs. Examples include: ICP:A and AA:F.
lab_samp_id	varchar(60)	YES	The Sample Number assigned by the lab.
dilut_factor	float	YES	Dilution factor entered as a single number when a sample has been diluted for analysis.. The number 100 is for a 1 to 100 dilution of the sample, a 0.1 is entered for a 10 to 1, and if a sample was not diluted, either the field is left blank or a 1 is entered.
batch_id	varchar(20)	YES	Identification of a group of analyses that were performed together and having something in common such as time of analysis or instrument.
con_name	varchar(60)	NO	Name of the constituent used for reporting purposes. Typically, the name comes from the Chemical Abstract Services (CAS) listing.
secondary_samp_prep_type	varchar(20)	YES	Secondary sample prep.
log_page	varchar(10)	YES	Log book page number containing specific sampling event information.
log_id	varchar(20)	YES	Identification number of the log book containing sampling event information.
reporting_day	varchar(15)	YES	Day in the analysis cycle when analysis was reported
analyst	varchar(20)	YES	Name or other identification of the technician who performed the analysis.
file_id	varchar(255)	YES	File ID used in the lab
sample_analysis_comment	varchar(255)	YES	Comment on the sample analysis.
post_tpa_flag	varchar(3)	YES	Data and information flagged TPA in this database are post-1989 analytical data and are recognized by the Washington State Department of Ecology for use in Tri- Party Agreement (TPA) and regulatory mandated documents. This data may be used for production of TPA deliverable documents.
sample_type	varchar(30)	NO	Type of sample. Current values are: CORE, SUPERNATE, and SURFACE.
agg_level	varchar(20)	NO	Label or descriptor applied to the whole, i.e. "aggregate", sample before separation into portions or fractions prior to analysis. See /dictionary/AggLevel.asp >Aggregation Level Definitions for more information.
device_type	varchar(30)	NO	Description of the type of container used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
sampling_event_name	varchar(20)	YES	Core ID, supernate sample ID, surface sample ID, tank composite name, or core composite name.
segment_name	varchar(20)	YES	Segment ID or core composite name.
sample_portion	varchar(20)	NO	A part of a sample that was taken.
tank_riser	varchar(12)	YES	Riser number on the waste tank.
sample_level	int	YES	Height at which the sample was taken, measured from the bottom of the waste site.
sample_level_units	varchar(10)	YES	Units of measure associated with the sample level.
sampling_begin_date	datetime	YES	Date the sampling started. Format is mm/dd/yyyy.
sampling_end_date	datetime	YES	Date the sampling was completed. Format is mm/dd/yyyy.
samp_date_time	datetime	YES	The date and time the sample was taken.
sampling_event_comment	varchar(255)	YES	A comment related to a sampling event.

recovery_percent	tinyint	YES	Percentage of the expected sample size that was recovered during removal from the tank.
appearance	varchar(255)	YES	Description of the physical appearance of the sample.
segment_comment	varchar(255)	YES	Further comments to clarify the information in the table.
parent_table	varchar(30)	NO	The name of the table in which the parent sample number is located.
physical_state	varchar(30)	YES	Liquid/Solid, the sample analysis state.
samp_prep_date_time	datetime	YES	Date and time the sample preparation took place. Format is mm/dd/yy hh:mm.
qa_type	varchar(20)	NO	Type of quality assurance sample, if any, represented by the current sample number. Current values include: DIW BLANK, FIELD BLANK, HOT CELL BLANK, LAB BLANK, TRIP BLANK, BLIND, HYDROSTATIC HEAD FLUID, LINER LIQUID, SERIAL DILUTION, STANDARD, SYNTHETIC, and NONE.
description	varchar(255)	YES	Brief narrative describing a specific instance in this database table.
doc_loc	varchar(150)	YES	Document in which analysis result is recorded.
lab_received_date	datetime	YES	Date sample was received at laboratory. Format is mm/dd/yy hh:mm.
sampler	varchar(20)	YES	Name of person who collected sample.
comp_name	varchar(255)	YES	Composite name
sample_comment	varchar(255)	YES	Comment about the sample.
transfer_since_sample	varchar(2)	YES	A waste transfer to or from this tank has occurred since this sampling_event.
sample_type_comment	varchar(255)	YES	Comment concerning the sample type.
preferred_method_group_id	varchar(30)	YES	The usual analysis method group for this constituent.
molecular_formula	varchar(50)	YES	Molecular formula for this constituent.
molecular_weight	float	YES	The molecular weight of the constituent is needed for converting M (moles/L) to inventory units (kg).
con_type	varchar(15)	NO	General type or classification: chemical, radiochemical, or physical property.
con_comment	varchar(255)	YES	A comment on the constituent.
half_life	float	YES	Half life of a chemical.
half_life_units	varchar(10)	YES	Units in which half life is reported.
heat_load_factor	float	YES	Heat load factor for this constituent
alpha_beta_gamma_emitter	varchar(10)	YES	Type of emitter -- ALPHA, BETA, or GAMMA.
valence	smallint	YES	The electronic charge of the waste constituent.
method_name	varchar(100)	YES	Name of analysis method.
analysis_class	varchar(10)	NO	A class or category of analysis.
analysis_procedure	varchar(20)	NO	The identification assigned to a laboratory procedure.
preparation_procedure	varchar(20)	NO	The code identifying a specific laboratory prep procedure.
method_description	varchar(255)	YES	
check_comment	varchar(255)	YES	Comment about check status.
check_date	datetime	YES	Date of checking.
check_status	varchar(60)	YES	The status of checking applied to the record.
Phase_assignment	varchar(30)	YES	Description which designates the phase assigned to the sample portion being measured or analyzed.
cas_number	varchar(15)	YES	Chemical Abstract Services (CAS) Number. These numbers were identified from sources such as ChemFinder, Aldrich catalog, the NIST database, and the STN database.
sample_analysis_id	int	YES	Identity field for the sample_analysis table.
retention_time	float	YES	
qa_level	varchar(30)	YES	
analysis_result_mod_id	int	NO	



Indexes

Index name	Column name	Sort direction	Is unique	Index type
------------	-------------	----------------	-----------	------------

i_id	analysis_result_mod_id	ASC		NONCLUSTERED
i1_analysis_result_mod	change_date_time	ASC	Yes	CLUSTERED
i1_analysis_result_mod	analysis_result_id	ASC	Yes	CLUSTERED
i2_analysis_result_mod	samp_num	ASC		NONCLUSTERED
i3_analysis_result_mod	waste_site_id	ASC		NONCLUSTERED
i4_analysis_result_mod	analysis_result_id	ASC		NONCLUSTERED

dbo.analysis_result_to_load

Holding table used by the TCDLoader2 ASP.NET application.

I F P	Column name	Data type	Nulls	Default	Description
	analysis_result_to_load_id	int	NO		
	con_name	varchar(255)	YES		Name of the constituent used for reporting purposes. Typically, the name comes from the Chemical Abstract Services (CAS) listing.
	con_id	varchar(255)	YES		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
	rpt_value	varchar(255)	YES		Reported value of the which may be recorded in different units than standard units.
	result_type	varchar(255)	YES		Description of the type of results the analysis produced. Examples include: PRIMARY, PRIMARY#2
	rpt_value_units	varchar(255)	YES		Units in which the value was originally reported, which may or may not be the same as the standard units.
	uncertainty_value	varchar(255)	YES		Uncertainty value for the DQO criteria
	uncertainty_value_units	varchar(255)	YES		Units associated with the uncertainty value for the DQO criteria.
	qualifiers	varchar(255)	YES		Indicators of the quality of the value. These codes can include data qualifiers, report flags, and validation flags. See "/dictionary/Qualifiers.asp" >Qualifier Definitions for more information.
	detection_limit	varchar(255)	YES		
	detection_limit_units	varchar(255)	YES		
	analysis_date_time	varchar(255)	YES		Analysis date and time
	comment	varchar(255)	YES		Comment
	data_type	varchar(255)	YES		In Master database tables.
	samp_num	varchar(255)	YES		Unique alpha-numeric ID assigned to identify a sample in the database.
	lab_analysis_procedure	varchar(255)	YES		
	primary_sample_preparation	varchar(255)	YES		
	sample_analysis_to_load_id	int	YES		
	load_date_time	datetime	YES		The date and time that the record was loaded into the TCD.
	ready_to_load	char(1)	YES		Flag indicating record is ready to be loaded
	sample_data_file_name	varchar(255)	YES		
	review_comment	varchar(6000)	YES		Comment added by reviewer of this analysis result record.
	is_preliminary_yn	char(1)	YES		
	review_yn	char(1)	YES		
	retention_time	varchar(255)	YES		
	qa_level	varchar(255)	YES		

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_con_name	con_name	ASC		NONCLUSTERED
i_id	analysis_result_to_load_id	ASC	Yes	NONCLUSTERED
i_sample_data_file_name	sample_data_file_name	ASC		NONCLUSTERED

dbo.Archive_AutoStatAnalysisResultData

The *dbo.p_autostat2* stored procedure saves a copy of the data that it is processing to this table.

I F P	Column name	Data type	Nulls	Default	Description
	autostatheader_id	int	NO		Unique ID for AutoStatHeader
	run_date	datetime	YES		Date AutoStat dataset was last processed
	waste_site_id	varchar(30)	YES		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	con_name	varchar(60)	YES		Name of the constituent used for reporting purposes. Typically, the name comes from the Chemical Abstract Services (CAS) listing.
	combined_method	varchar(30)	YES		Analytical method
	physical_state	varchar(30)	YES		Liquid/Solid, the sample analysis state.
	agg_level	varchar(20)	YES		Label or descriptor applied to the whole, i.e. "aggregate", sample before separation into portions or fractions prior to analysis. See /dictionary/AggLevel.asp >Aggregation Level Definitions for more information.
	std_value	float	YES		Inventory value reported in standard units. (The inventory value in standard units is reported in the second column of the best-basis inventory table.)
	std_value_units	varchar(10)	YES		Common units of measure: kilograms (kg) for chemicals and curies (Ci) for radionuclides.
	result_type	varchar(20)	YES		Description of the type of results the analysis produced. Examples include: PRIMARY, PRIMARY#2
	device_type	varchar(30)	YES		Description of the type of container used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
	phase_assignment	varchar(30)	YES		Description which designates the phase assigned to the sample portion being measured or analyzed.
	lab_samp_id	varchar(60)	YES		The Sample Number assigned by the lab.
	qualifiers	varchar(20)	YES		Indicators of the quality of the value. These codes can include data qualifiers, report flags, and validation flags. See /dictionary/Qualifiers.asp >Qualifier Definitions for more information.
	sampling_event_name	varchar(20)	YES		Core ID, supernate sample ID, surface sample ID, tank composite name, or core composite name.
	doc_loc	varchar(150)	YES		Document in which analysis result is recorded.
	segment_name	varchar(20)	YES		Segment ID or core composite name.
	sample_portion	varchar(20)	YES		A part of a sample that was taken.
	review_status	varchar(30)	YES		Review status. Current values are: REVIEWED, REVIEWED: SEE NOTE, CHECKING NOT APPLIED, DATA SUSPECT: SEE COMMENT, REVIEWED: NOT FOR INVENTORY and DATA RESOLUTION IN PROGRESS. See /dictionary/ReviewStatusCaveats.asp >Review Status Caveats for more information.
	sample_type	varchar(20)	YES		Type of sample. Current values are: CORE, SUPERNATE, and SURFACE.
	samp_num	varchar(12)	YES		Unique alpha-numeric ID assigned to identify a sample in the database.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_id	autostatheader_id	ASC		NONCLUSTERED
i_waste_site_id	waste_site_id	ASC		NONCLUSTERED

dbo.Archive_AutoStatHeader

After the `dbo.p_autostat2` stored procedure recalculates the auto stat data, the `dbo.p_AutoStatCleanup` stored proc records a snapshot of the new data in this table.

I F P	Column name	Data type	Nulls	Default	Description
	AutoStatHeader_id	int	NO		Unique ID for AutoStatHeader
	waste_site_id	varchar(30)	YES		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	reference_date	datetime	NO		Date of the reference document.
	physical_state	varchar(10)	YES		Liquid/Solid, the sample analysis state.
	phase_assignment	varchar(30)	YES		Description which designates the phase assigned to the sample portion being measured or analyzed.
	device_type	varchar(30)	YES		Description of the type of container used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
	needs_to_be_run	char(1)	YES		Y to run model
	last_run	datetime	YES		date of last run
	last_modified_date	datetime	YES		Date of last modification to data.
	last_options_update	datetime	YES		Data options were last updated
	Comments	varchar(3000)	YES		General comment by statisticians.
	autotcr_title	varchar(255)	YES		Title for the AutoStat dataset
	has_data_resolution_in_progress	char(1)	YES		Flag to indicate data has a resolution in progress in RESOLVE
	use_in_autotcr	char(1)	YES		Flag to indicate dataset is to be used in the AutoTCR
	recommended_grouping	varchar(60)	YES		Average grouping recommended by the system based on data composition
	requested_grouping	varchar(60)	YES		Average grouping requested by the user

dbo.Archive_AutoStatHeaderOptions

After the `dbo.p_autostat2` stored procedure recalculates the auto stat data, the `dbo.p_AutoStatCleanup` stored proc records a snapshot of the new data in this table.

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	AutoStatHeaderOptions_id	int	NO		Unique ID for AutoStatHeaderOptions
	AutoStatHeader_id	int	YES		Unique ID for AutoStatHeader
	field_name	varchar(30)	YES		The name of the table field where the data was modified.
	field_value	varchar(255)	YES		Value associated with field_name for this AutoStat Dataset
	use_value	char(1)	YES		Indicates whether record is included/excluded in criteria for dataset
	gray_out	char(1)	YES		Flag indicating that option is to be grayed out
	has_data_resolution_in_progress	char(1)	YES		Flag to indicate data has a resolution in progress in RESOLVE
	display_sequence	int	YES		The order in which something should be displayed.
	run_date	datetime	YES		Date AutoStat dataset was last processed

dbo.Archive_AutoStatResults

After the *dbo.p_autostat2* stored procedure recalculates the auto stat data, the *dbo.p_AutoStatCleanup* stored proc records a snapshot of the new data in this table.

I F P	Column name	Data type	Nulls	Default	Description
	AutoStatResults_id	int	NO		Unique ID for AutoStatResult
	AutoStatHeader_id	int	YES		Unique ID for AutoStatHeader
	waste_site_id	varchar(30)	YES		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	con_id	varchar(15)	YES		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
	molecular_formula	varchar(60)	YES		Molecular formula for this constituent.
	method_group_id	varchar(60)	YES		Name assigned to the general group of laboratory analytical methods to which the current method belongs. Examples include: ICP:A and AA:F.
	phase	varchar(10)	YES		Predominate solid waste phase in the tank as judged by the largest of the salt volume or sludge volume.
	overall_average	decimal(28.9)	YES		Overall average
	units	varchar(10)	YES		Concentration units of the reported value or the reporting limit.
	std_dev	decimal(28.9)	YES		Standard deviation for the mean.
	degrees_of_freedom	int	YES		Degrees of freedom.
	lower_limit_95	float	YES		Lower 95% confidence limit.
	upper_limit_95	float	YES		Upper 95% confidence limit.
	number_of_detected_values	int	YES		Number of measurements above detection limits
	number_of_non_detected_values	int	YES		Number of measurements below detection limits
	comments	varchar(3000)	YES		General comment by statisticians.
	statistical_model	varchar(30)	YES		Model that was fit to data.
	sampling_events	varchar(3000)	YES		List of sampling events covered by the means and variances data
	N	int	YES		Total number of measurements
	a	int	YES		Number of unique lab_samp_id
	S2	int	YES		Sum of the squares of the number of measurements in each level (constituent_name, combined_method, physical_state, std_value_units)
	T0	decimal(28.7)	YES		Sum of the squared std_value each level (constituent_name, combined_method, physical_state, std_value_units)
	Ta	decimal(28.7)	YES		Squared value of the sum of std_value each level (constituent_name, combined_method, physical_state, std_value_units) divided by the number of measurements
	Tu	decimal(28.7)	YES		Squared value of the sum of std_value each level (constituent_name, combined_method, physical_state, std_value_units) divided by the number of measurements (all lab_samp_id)
	T0Tiny	decimal(28.16)	YES		Sum of the squared std_value each level (constituent_name, combined_method, physical_state, std_value_units) -- when very small numbers are involved
	TaTiny	decimal(28.16)	YES		Same as Ta but for very small numbers
	TuTiny	decimal(28.16)	YES		Same as Tu but for very small numbers
	S2e	decimal(28.7)	YES		$(T0 - Ta)/(N-a)$
	S2a	decimal(28.7)	YES		$(Ta - Tu - (a - 1) * S2e)/(N - (S2/N))$
	S2eTiny	decimal(28.16)	YES		Same as S2e but for very small number
	S2aTiny	decimal(28.16)	YES		Same as S2a but for very small number
	has_data_resolution_in_progress	char(1)	YES		Flag to indicate data has a resolution in progress in RESOLVE
	run_date	datetime	YES		Date AutoStat dataset was last processed

dbo.Archive_AutoStatSpecialCases

After the `dbo.p_autostat2` stored procedure recalculates the auto stat data, the `dbo.p_AutoStatCleanup` stored proc records a snapshot of the new data in this table.

I F P	Column name	Data type	Nulls	Default	Description
	AutoStatSpecialCases_id	int	NO		Uniqued ID for AutoStatSpecialCases
	waste_site_id	varchar(30)	YES		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	AutoStatHeader_id	int	YES		Unique ID for AutoStatHeader
	doc_loc	varchar(150)	YES		Document in which analysis result is recorded.
	sample_portion	varchar(20)	YES		A part of a sample that was taken.
	sampling_event_name	varchar(20)	YES		Core ID, supernate sample ID, surface sample ID, tank composite name, or core composite name.
	segment_name	varchar(20)	YES		Segment ID or core composite name.
	agg_level	varchar(20)	YES		Label or descriptor applied to the whole, i.e. "aggregate", sample before separation into portions or fractions prior to analysis. See /dictionary/AggLevel.asp >Aggregation Level Definitions for more information.
	physical_state	varchar(10)	YES		Liquid/Solid, the sample analysis state.
	phase_assignment	varchar(30)	YES		Description which designates the phase assigned to the sample portion being measured or analyzed.
	device_type	varchar(30)	YES		Description of the type of container used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
	special_case_candidate	char(1)	YES		This record is a special case candidate
	use_special_case	char(1)	YES		Flag to indicate that record is to be used in the dataset
	has_data_resolution_in_progress	char(1)	YES		Flag to indicate data has a resolution in progress in RESOLVE
	run_date	datetime	YES		Date AutoStat dataset was last processed

dbo.AutoStatHeaderOptions

Referenced by the AutoStat ASP.NET application.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			AutoStatHeaderOptions_id	int	NO		Unique ID for AutoStatHeaderOptions
			AutoStatHeader_id	int	YES		Unique ID for AutoStatHeader
			field_name	varchar(30)	YES		The name of the table field where the data was modified.
			field_value	varchar(255)	YES		Value associated with field_name for this AutoStat Dataset
			use_value	char(1)	YES		Indicates whether record is included/excluded in criteria for dataset
			gray_out	char(1)	YES		Flag indicating that option is to be grayed out
			has_data_resolution_in_progress	char(1)	YES		Flag to indicate data has a resolution in progress in RESOLVE
			display_sequence	int	YES		The order in which something should be displayed.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_autostatheader_id	AutoStatHeader_id	ASC		NONCLUSTERED
i_autostatheaderoptions_id	AutoStatHeaderOptions_id	ASC		NONCLUSTERED
i_field_name	field_name	ASC		NONCLUSTERED

dbo.AutoStatIDsProcessed

This table has no records, therefore it is probably not used.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			autostatheader_id	int	YES		Unique ID for AutoStatHeader
			run_date	datetime	YES		Date AutoStat dataset was last processed

dbo.AutoStatOverview

Used by the AutoStat ASP.NET application, and populated by the dbo.p_autostatoverview_populate stored procedure.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			AutoStatHeader_id	int	YES		Unique ID for AutoStatHeader
			waste_site_id	varchar(30)	YES		Tank Identifier, using the format of 241-Tank Farm-Tank #.
			doc_loc	varchar(8000)	YES		Document in which analysis result is recorded.
			agg_level	varchar(8000)	YES		Label or descriptor applied to the whole, i.e. "aggregate", sample before separation into portions or fractions prior to analysis. See /dictionary/AggLevel.asp >Aggregation Level Definitions for more information.
			sampling_event_name	varchar(8000)	YES		Core ID, supernate sample ID, surface sample ID, tank composite name, or core composite name.
			segment_name	varchar(8000)	YES		Segment ID or core composite name.
			sample_portion	varchar(8000)	YES		A part of a sample that was taken.
			more_recent_data	char(1)	YES		Flag indicating more recent data is available

dbo.AutoStatResults

Referenced by the AutoStat ASP.NET application.

I F P	Column name	Data type	Nulls	Default	Description
	AutoStatResults_id	int	NO		Unique ID for AutoStatResult
	AutoStatHeader_id	int	YES		Unique ID for AutoStatHeader
	waste_site_id	varchar(30)	YES		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	con_id	varchar(15)	YES		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
	molecular_formula	varchar(60)	YES		Molecular formula for this constituent.
	method_group_id	varchar(60)	YES		Name assigned to the general group of laboratory analytical methods to which the current method belongs. Examples include: ICP:A and AA:F.
	phase	varchar(10)	YES		Predominate solid waste phase in the tank as judged by the largest of the salt volume or sludge volume.
	overall_average	decimal(28.9)	YES		Overall average
	units	varchar(10)	YES		Concentration units of the reported value or the reporting limit.
	std_dev	decimal(28.9)	YES		Standard deviation for the mean.
	degrees_of_freedom	int	YES		Degrees of freedom.
	lower_limit_95	float	YES		Lower 95% confidence limit.
	upper_limit_95	float	YES		Upper 95% confidence limit.
	number_of_detected_values	int	YES		Number of measurements above detection limits
	number_of_non_detected_values	int	YES		Number of measurements below detection limits
	comments	varchar(3000)	YES		General comment by statisticians.
	statistical_model	varchar(30)	YES		Model that was fit to data.
	sampling_events	varchar(3000)	YES		List of sampling events covered by the means and variances data
	N	int	YES		Total number of measurements
	a	int	YES		Number of unique lab_samp_id
	S2	int	YES		Sum of the squares of the number of measurements in each level (constituent_name, combined_method, physical_state, std_value_units)
	T0	decimal(28.7)	YES		Sum of the squared std_value each level (constituent_name, combined_method, physical_state, std_value_units)
	Ta	decimal(28.7)	YES		Squared value of the sum of std_value each level (constituent_name, combined_method, physical_state, std_value_units) divided by the number of measurements
	Tu	decimal(28.7)	YES		Squared value of the sum of std_value each level (constituent_name, combined_method, physical_state, std_value_units) divided by the number of measurements (all lab_samp_id)
	T0Tiny	decimal(28.16)	YES		Sum of the squared std_value each level (constituent_name, combined_method, physical_state, std_value_units) -- when very small numbers are involved
	TaTiny	decimal(28.16)	YES		Same as Ta but for very small numbers
	TuTiny	decimal(28.16)	YES		Same as Tu but for very small numbers
	S2e	decimal(28.7)	YES		$(T0 - Ta)/(N - a)$
	S2a	decimal(28.7)	YES		$(Ta - Tu - (a - 1) * S2e)/(N - (S2/N))$
	S2eTiny	decimal(28.16)	YES		Same as S2e but for very small number
	S2aTiny	decimal(28.16)	YES		Same as S2a but for very small number
	has_data_resolution_in_progress	char(1)	YES		Flag to indicate data has a resolution in progress in RESOLVE

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_autostatheader_id	AutoStatHeader_id	ASC		NONCLUSTERED

i_con_id	con_id	ASC	NONCLUSTERED
i_waste_site_id	waste_site_id	ASC	NONCLUSTERED

dbo.AutoStatSpecialCases

Referenced by the AutoStat ASP.NET application.

I	F	P	Column name	Data type	Nulls	Default	Description
			AutoStatSpecialCases_id	int	NO		Uniqued ID for AutoStatSpecialCases
			waste_site_id	varchar(30)	YES		Tank Identifier, using the format of 241-Tank Farm-Tank #.
			AutoStatHeader_id	int	YES		Unique ID for AutoStatHeader
			doc_loc	varchar(150)	YES		Document in which analysis result is recorded.
			sample_portion	varchar(20)	YES		A part of a sample that was taken.
			sampling_event_name	varchar(20)	YES		Core ID, supernate sample ID, surface sample ID, tank composite name, or core composite name.
			segment_name	varchar(20)	YES		Segment ID or core composite name.
			agg_level	varchar(20)	YES		Label or descriptor applied to the whole, i.e. "aggregate", sample before separation into portions or fractions prior to analysis. See /dictionary/AggLevel.asp >Aggregation Level Definitions for more information.
			physical_state	varchar(10)	YES		Liquid/Solid, the sample analysis state.
			phase_assignment	varchar(30)	YES		Description which designates the phase assigned to the sample portion being measured or analyzed.
			device_type	varchar(30)	YES		Description of the type of container used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
			special_case_candidate	char(1)	YES		This record is a special case candidate
			use_special_case	char(1)	YES		Flag to indicate that record is to be used in the dataset
			has_data_resolution_in_progress	char(1)	YES		Flag to indicate data has a resolution in progress in RESOLVE

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_autostatheader_id	AutoStatHeader_id	ASC		NONCLUSTERED

dbo.bottom_shape

Bottom shape of a tank

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			bottom_shape	varchar(20)	NO		The type of bottom shape that a waste tank has.
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_bottom_shape	bottom_shape	ASC	Yes	CLUSTERED
PK__bottom_shape__4D21E854	bottom_shape	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.waste_tank	FK__waste_tan__botto__1A2C5433	i_bottom_shape

dbo.con_analysis_method_group

Constituents can be analyzed by specific analysis method. This is a cross-reference list of methods to constituent groups. Used by the TCDLoader2 ASP.NET application.

I	F	P	Column name	Data type	Nulls	Default	Description
			con_analysis_method_group_id	int	NO		Unique record identifier.
			con_id	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
			method_group_id	varchar(30)	NO		Name assigned to the general group of laboratory analytical methods to which the current method belongs. Examples include: ICP:A and AA:F.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i1_con_analysis_method_group	con_id	ASC	Yes	CLUSTERED
i1_con_analysis_method_group	method_group_id	ASC	Yes	CLUSTERED
i2_con_analysis_method_group	method_group_id	ASC		NONCLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__con_analy__con_i__35155499	con_id	i_constituent ()	
FK__con_analy__metho__2D7432D1	method_group_id	i_analysis_method_group ()	
FK__con_analy__con_i__2C800E98	con_id	i_constituent ()	
FK__con_analy__metho__360978D2	method_group_id	i_analysis_method_group ()	

dbo.con_type

Constituent type

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			con_type	varchar(15)	NO		General type or classification: chemical, radiochemical, or physical property.
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_con_type	con_type	ASC	Yes	CLUSTERED
PK__con_type__4F0A30C6	con_type	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.std_constituent_units	FK__std_const__con_t__07197FBF	i_con_type
dbo.constituent	FK__constitue__con_t__454BBC62	i_con_type

dbo.constituent

Chemicals and radionuclides found in Hanford HLW tanks. There are currently 25 major chemicals, 5 minor chemicals, and 46 radionuclides that are reported in the best basis documents. These are the constituents that will have the largest effect on the waste retrieval and processing (vitrification).

I	F	P	Column name	Data type	Nulls	Default	Description
			con_id	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
			preferred_method_group_id	varchar(30)	YES		The usual analysis method group for this constituent.
			con_type	varchar(15)	NO		General type or classification: chemical, radiochemical, or physical property.
			con_name	varchar(60)	NO		Name of the constituent used for reporting purposes. Typically, the name comes from the Chemical Abstract Services (CAS) listing.
			molecular_formula	varchar(50)	YES		Molecular formula for this constituent.
			molecular_weight	float	YES		The molecular weight of the constituent is needed for converting M (moles/L) to inventory units (kg).
			comment	varchar(255)	YES		Comment
			half_life	float	YES		Half life of a chemical.
			half_life_units	varchar(10)	YES		Units in which half life is reported.
			heat_load_factor	float	YES		Heat load factor for this constituent
			alpha_beta_gamma_emitter	varchar(10)	YES		Type of emitter -- ALPHA, BETA, or GAMMA.
			valence	float	YES		The electronic charge of the waste constituent.
			analysis_use	char(1)	YES		Are there analysis results for this constituent or chemical in this database? Y or N.
			inventory_use	char(1)	YES		Is this constituent used in inventory
			historical_use	char(1)	YES		Is this constituent used in historical data
			baseline_use	char(1)	YES		This constituent used for baseline - Y or N.
			perform_review	char(1)	YES		Should computerized review be performed -- no longer used
			median_result	float	YES		Median result -- no longer used
			recent_change	char(1)	YES		A flag indicating as recent change to records has taken place. Used to trigger the update of AutoTCR tables.
			uncertainty_use	char(1)	YES		Flag indicating where a con_id is used.
			CAS_number	varchar(15)	YES		Chemical Abstract Services (CAS) Number. These numbers were identified from sources such as ChemFinder, Aldrich catalog, the NIST database, and the STN database.
			sounds_like	char(4)	YES		Soundex for the chemical name

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_constituent	con_id	ASC	Yes	CLUSTERED
i2_constituent	con_name	ASC	Yes	NONCLUSTERED
PK_constituent__4FFE54FF	con_id	ASC	Yes	NONCLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK_constitue__alpha__44579829	alpha_beta_gamma_emitter	i_alpha_beta_gamma_emitter ()	
FK_constitue__prefe__473404D4	preferred_method_group_id	i_analysis_method_group ()	
FK_constitue__half__463FE09B	half_life_units	i_units ()	
FK_constitue__con_t__454BBC62	con_type	i_con_type ()	

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.con_analysis_method_group	FK_con_analy__con_i__35155499	i_constituent

dbo.con_analysis_method_group	FK__con_analy__con_i__2C800E98	i_constituent
dbo.std_property_units	FK__std_prope__con_i__09F5EC6A	i_constituent
dbo.constituent_group_member	FK__constitue__con_i__30509F7C	i_constituent
dbo.analysis_result	fk_constituent	i_constituent
dbo.means_and_variances_detail	FK__means_and__con_i__5C2F21BA	i_constituent
dbo.constituent_synonym	FK__constitue__con_i__4CECDE2A	i_constituent
dbo.constituent_group_member	FK__constitue__con_i__491C4D46	i_constituent

Triggers

Trigger name:	dbo.t_constituent_del	Created on:	14 Jul 2006
Trigger type:	DELETE	Trigger active:	Yes
Trigger name:	dbo.t_constituent_ins_upd	Created on:	14 Jul 2006
Trigger type:	INSERT, UPDATE	Trigger active:	Yes

dbo.constituent_group

Groups of constituents such as Metals, PCBs, etc. Referemced by the CheckConstituentGroupMembers T2 and Populate Working_Contemp T2 DTS packages.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			con_group_id	varchar(20)	NO		Identifying name for a group of constituents.
			con_group_name	varchar(40)	NO		Constituent group name.
			comment	varchar(255)	YES		Comment
			for_v_tank_result	char(1)	YES		Is this constituent group used in the TWINS interface to identify constituent groups in a drop-down box

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_constituent_group	con_group_id	ASC	Yes	CLUSTERED
PK__constituent_grou__50F27938	con_group_id	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.constituent_group_member	FK__constitue__con_g__4828290D	i_constituent_group
dbo.constituent_group_member	FK__constitue__con_g__2F5C7B43	i_constituent_group

dbo.constituent_group_member

Identifies which group a constituent belongs in. Referenced by the CheckConstituentGroupMembers T2 and Populate Working_Contemp T2 DTS packages.

I	F	P	Column name	Data type	Nulls	Default	Description
			constituent_group_member_id	int	NO		Unique record identifier.
			con_group_id	varchar(20)	NO		Identifying name for a group of constituents.
			con_id	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_constituent_group_member	con_group_id	ASC	Yes	CLUSTERED
i_constituent_group_member	con_id	ASC	Yes	CLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__constitue__con_g__4828290D	con_group_id	i_constituent_group ()	
FK__constitue__con_g__2F5C7B43	con_group_id	i_constituent_group ()	
FK__constitue__con_i__30509F7C	con_id	i_constituent ()	
FK__constitue__con_i__491C4D46	con_id	i_constituent ()	

dbo.constituent_mod

Used by the Constituent update, insert and delete triggers to record changes in the Constituent table.

I	F	P	Column name	Data type	Nulls	Default	Description
			change_date_time	datetime	NO		The date and time that a change was made to the record.
			con_id	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
			 change_action	varchar(8)	NO		The type of change that was made to an analysis_result record - delete or update.
			change_reason	varchar(255)	NO		Field that allows the data entry person to enter a reason for the change that was made to the original record.
			change_user_name	varchar(30)	NO		The user_name of the person who changed the record.
			preferred_method_group_id	varchar(30)	YES		The usual analysis method group for this constituent.
			con_type	varchar(15)	NO		General type or classification: chemical, radiochemical, or physical property.
			con_name	varchar(60)	NO		Name of the constituent used for reporting purposes. Typically, the name comes from the Chemical Abstract Services (CAS) listing.
			molecular_formula	varchar(50)	YES		Molecular formula for this constituent.
			molecular_weight	float	YES		The molecular weight of the constituent is needed for converting M (moles/L) to inventory units (kg).
			comment	varchar(255)	YES		Comment
			half_life	float	YES		Half life of a chemical.
			half_life_units	varchar(10)	YES		Units in which half life is reported.
			heat_load_factor	float	YES		Heat load factor for this constituent
			alpha_beta_gamma_emitter	varchar(10)	YES		Type of emitter -- ALPHA, BETA, or GAMMA.
			valence	float	YES		The electronic charge of the waste constituent.
			analysis_use	char(1)	YES		Are there analysis results for this constituent or chemical in this database? Y or N.
			inventory_use	char(1)	YES		Is this constituent used in inventory
			historical_use	char(1)	YES		Is this constituent used in historical data
			baseline_use	char(1)	YES		This constituent used for baseline - Y or N.
			perform_review	char(1)	YES		Should computerized review be performed -- no longer used
			median_result	float	YES		Median result -- no longer used
			recent_change	char(1)	YES		A flag indicating as recent change to records has taken place.
			uncertainty_use	char(1)	YES		Flag indicating where a con_id is used.
			CAS_number	varchar(15)	YES		Chemical Abstract Services (CAS) Number. These numbers were identified from sources such as ChemFinder, Aldrich catalog, the NIST database, and the STN database.
			constituent_mod_id	int	NO		Unique record identifier.

Foreign keys

Constraint name	Column name	Reference	Description
FK_constitue__chang__3144C3B5	change_action	i_change_action ()	
FK_constitue__chang__4A10717F	change_action	i_change_action ()	

dbo.constituent_synonym

Synonym for a constituent name such as NO3 or Nitrate

I F P	Column name	Data type	Nulls	Default	Description
	synonym_name	varchar(100)	NO		Standard best-basis analyte expressed as chemical symbol (reported without charge designation per the best-basis inventory convention).
	synonym_group_id	varchar(40)	NO		ID for the synonym group
	con_id	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
	description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.
	recent_change	char(1)	YES		A flag indicating as recent change to records has taken place. Used to trigger the update of AutoTCR tables.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_constituent_synonym	synonym_name	ASC	Yes	CLUSTERED
i_constituent_synonym	synonym_group_id	ASC	Yes	CLUSTERED
i_constituent_synonym	con_id	ASC	Yes	CLUSTERED
i1_constituent_synonym	con_id	ASC		NONCLUSTERED
i2_constituent_synonym	synonym_name	ASC		NONCLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__constitue__synon__4DE10263	synonym_group_id	i_synonym_group_id ()	
FK__constitue__con_i__4CECD E2A	con_id	i_constituent ()	

Triggers

Trigger name: dbo.t_constituent_synonym_del
Trigger type: DELETE
Created on: 14 Jul 2006
Trigger active: Yes

Trigger name: dbo.t_constituent_synonym_ins_upd
Trigger type: INSERT, UPDATE
Created on: 14 Jul 2006
Trigger active: Yes

dbo.constituent_synonym_mod

Used by the constituent_synonym insert, update and delete triggers to record changes to the constituent_synonym table.

I	F	P	Column name	Data type	Nulls	Default	Description
			change_date_time	datetime	NO		The date and time that a change was made to the record.
			synonym_name	varchar(100)	NO		Standard best-basis analyte expressed as chemical symbol (reported without charge designation per the best-basis inventory convention).
			change_action	varchar(8)	NO		The type of change that was made to an analysis_result record - delete or update.
			change_reason	varchar(255)	NO		Field that allows the data entry person to enter a reason for the change that was made to the original record.
			change_user_name	varchar(30)	YES		The user_name of the person who changed the record.
			synonym_group_id	varchar(40)	NO		ID for the synonym group
			con_id	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.
			recent_change	char(1)	YES		A flag indicating as recent change to records has taken place.

Foreign keys

Constraint name	Column name	Reference	Description
FK_constitue__chang__4ED5269C	change_action	i_change_action ()	

dbo.count_TVD_records_loaded

I believe that the Caveat system is obsolete. Look at the comments in tcd.dbo.p_check_rules and Change Request 159.

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	year_loaded	int	NO		The year in which the record that is referred to was loaded into the database.
	month_loaded	int	NO		The month in which a record was loaded.
	waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	total_tank_records	int	NO		Total number of result records for a waste tank.

dbo.cron_status

Records the status of the nightly processing.

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	cron_status	varchar(80)	NO		A field for description messages about the status of night processing.
	cron_status_date	datetime	NO		The date and time that a night processing event occurred.
	cron_status_id	int	NO		Unique record identifier.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_cron_status	cron_status_date	ASC		NONCLUSTERED

dbo.data_status

Lookup table. Values indicate if the data is affected by waste transfers occurring after the analysis was performed.

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	data_status	varchar(30)	YES		Data is current (no transfers since sampled), pre-transfer (previous to a transfer), or useable even if prior to a transfer
	description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

dbo.Debug_Setting

BBIM triggers and stored procedures "select" and on/off setting from the table Debug_Setting which enables them to turn on or off their debugging messages.

I	F	P	Column name	Data type	Nulls	Default	Description
			DS_ID	int	NO		Identity field, generated by the database.
			Type	varchar(4)	NO		The type of database object that can be debugged.
			Name	varchar(60)	NO		Name of the trigger or procedure in the BBIM database.
			Setting	smallint	NO		0 or -1, with 0 representing "turn OFF the debugging messages" and -1 representing "turn ON the debugging messages". -1 is used for "ON" because this corresponds to the MS Access checkbox in a form being "Checked".

Indexes

Index name	Column name	Sort direction	Is unique	Index type
_WA_Sys_Name	Name	ASC		NONCLUSTERED
U_Debug_Setting	Type	ASC	Yes	NONCLUSTERED
U_Debug_Setting	Name	ASC	Yes	NONCLUSTERED

Check constraints

Check name	Column name	Expression
CK_Setting	Setting	([Setting] = 0 or [Setting] = (-1))
CK_Type	Type	([Type] = 'PROC' or [Type] = 'TRIG')

dbo.device_type

Device type used to obtain the sample

I	F	P	Column name	Data type	Nulls	Default	Description
			device_type_id	int	NO		Unique record identifier.
			device_type	varchar(30)	NO		Description of the type of container used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
			sample_type	varchar(30)	NO		Type of sample. Current values are: CORE, SUPERNATE, and SURFACE.
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
PK__device_type__1A966887	device_type	ASC	Yes	CLUSTERED
PK__device_type__1A966887	sample_type	ASC	Yes	CLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__device_ty__sampl__51B19347	sample_type	i_sample_type ()	

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.sampling_event	FK__sampling_event__043D1314	PK__device_type__1A966887
dbo.sampling_event	FK__sampling_event__7AB3A8DA	PK__device_type__1A966887

dbo.dqo_exceeding_limits

I F P	Column name	Data type	Nulls	Default	Description
	con_name	varchar(60)	YES		Name of the constituent used for reporting purposes. Typically, the name comes from the Chemical Abstract Services (CAS) listing.
	con_id	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
	dqo_type	varchar(20)	YES		Type of DQO.
	document_revision_number	varchar(40)	YES		Document revision number for the DQO.
	waste_site_id	varchar(30)	YES		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	operator	varchar(5)	YES		Limiting operator for the DQO criteria. The values are >, <, or =.
	analyzed_value	float	YES		Analyzed value which comes from the analytical results data and which exceeds (or is associated with other data which exceeds) established DQO limits.
	analyzed_value_units	varchar(10)	YES		Units associated with the analyzed value for the DQO criteria.
	title	varchar(255)	YES		The title of this work unit. Titles are used over and over again. Current examples are "New TCR", "Revised TCR", "Reconcile" and "Discrepancies".
	analyzed_uncertainty_value	float	YES		Analyzed uncertainty value which comes from the analytical results data and which exceeds (or is associated with other data which exceeds) established DQO limits.
	analyzed_uncertainty_units	varchar(10)	YES		Units associated with the analyzed uncertainty value for the DQO criteria.
	limit_value	float	YES		Limit value associated with the DQO criteria. This value is compared against the values obtained during the analytical process to identify those values exceeding DQO limits.
	limit_value_units	varchar(10)	YES		Units associated with the limit value for the DQO criteria.
	uncertainty_limit	float	YES		Uncertainty limit for the DQO criteria.
	uncertainty_limit_units	varchar(10)	YES		Units associated with the uncertainty limit for the DQO criteria.
	sample_description	varchar(255)	YES		Contains descriptive data about each sample consolidated into a single entry.
	dqo_description	varchar(100)	YES		Description of the DQO criteria which is exceeded by this analytical result.
	result_type	varchar(20)	YES		Description of the type of results the analysis produced. Examples include: PRIMARY, PRIMARY#2
	criteria_type	varchar(20)	YES		Criteria type indicating whether selection is part of a single or multiple criteria DQO.
	estimated_value	float	YES		Estimated value based upon an assumed density when concentration is Liters. It is 1.6 for solids, 1.2 for liquids.
	estimated_value_units	varchar(10)	YES		Units associated with the estimated value.
	out_of_bounds	varchar(3)	YES		Yes/No answer indicating the analyzed (or estimated) value is out of bounds relative to the DQO criteria.
	warning_note1	varchar(255)	YES		Explanation of the estimated value (part 1).
	warning_note2	varchar(255)	YES		Explanation of the estimated value (part 2).

dbo.ESP_data

Environmental Simulation Program data (Implemented in 1999).

I	F	P	Column name	Data type	Nulls	Default	Description
			waste_site_id	varchar(30)	YES		Tank Identifier, using the format of 241-Tank Farm-Tank #.
			synonym_name	varchar(40)	YES		Standard best-basis analyte expressed as chemical symbol (reported without charge designation per the best-basis inventory convention).
			aqueous_mol_hr	float	YES		Moles per hour in aqueous phase.
			aqueous_bulk_conc_mol_L	float	YES		The bulk concentration of the chemical species in solution reported in molarity (moles/L). The aqueous bulk concentration is the total moles of a species dissolved in the liquid phase divided by the total waste volume, regardless of whether the total waste volume is liquid, solid or a mixture of the two phases.
			solid_mol_hr	float	YES		Moles per hour in solid phase.
			solid_bulk_conc_mol_L	float	YES		The bulk concentration of the chemical species that is precipitated reported in molarity (moles/L). The solid bulk concentration is the total moles of a species in the solid phase (dry basis) divided by the total waste volume, regardless of whether the total waste volume is liquid, solid or a mixture of the two phases.
			ccb_log	varchar(10)	YES		The Log number of the Configuration Control Board's documentation that authorizes this work unit (change in BBIM data). Current examples are "CCB-001" up to "CCB-064".
			CCB_approval_date	datetime	YES		The date that the related change log was formally approved by the Best-Basis Inventory Change Control Board.
			load_date_time	datetime	YES		The date and time that the record was loaded into the TCD.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i1_ESP_data	waste_site_id	ASC	Yes	CLUSTERED
i1_ESP_data	synonym_name	ASC	Yes	CLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__ESP_data__waste___13B460CE	waste_site_id	i_waste_tank ()	

dbo.ESP_metadata

Environmental Simulation Program data (Implemented in 1999).

I F P	Column name	Data type	Nulls	Default	Description
	waste_site_id	varchar(30)	YES		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	ESP_version	varchar(10)	YES		The software version of the ESP used to perform the waste speciation. Output from a later ESP version may include chemical species that were not included in the data banks of earlier versions.
	date_ESP_run	datetime	YES		The date that the ESP simulation was performed. The Best-Basis Inventory in effect on that date was used in the chemical speciation. The initial data set is based primarily on the Best-Basis Inventories in effect as of October 1, 1997.
	tempC	float	YES		The temperature in degrees Celsius assumed for the ESP simulation. An approximate average of the actual waste temperature measurements was used when available.
	aqueous_density	float	YES		The ESP predicted density of the liquid or solid phase reported in g/mL. The density for the solid phase is reported on a dry basis, and neither the liquid density nor the solids density corresponds directly with the bulk chemical concentration data.
	solid_density	float	YES		The ESP predicted density of the liquid or solid phase reported in g/mL. The density for the solid phase is reported on a dry basis, and neither the liquid density nor the solids density corresponds directly with the bulk chemical concentration data.
	total_volumelnLiters	float	YES		The total volume measured in liters.
	total_waste_volume_liters	float	YES		The total waste volume assumed in the ESP simulation. Generally, this was the waste volume established by the Waste Tank Summary Report (HNF-EP-0182-XXX).
	load_date_time	datetime	YES		The date and time that the record was loaded into the TCD.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_ESP_metadata	waste_site_id	ASC	Yes	NONCLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__ESP_metad__waste__14A88507	waste_site_id	i_waste_tank ()	

dbo.integrity

Lookup table of the tank integrity acceptable values.

I	F	P	Column name	Data type	Nulls	Default	Description
			integrity	varchar(20)	NO		Assesment of the tank's ability to contain waste without leaking
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.
			abbrev	varchar(20)	YES		An abbreviation for another field in the table. The exact meaning depends on the table in which it is used.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_integrity	integrity	ASC	Yes	CLUSTERED
PK__integrity__55B72E55	integrity	ASC	Yes	NONCLUSTERED

dbo.isolation_status

Lookup table of the tankisolation status acceptable values.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			isolation_status	varchar(80)	NO		Condition of a tank's connection to external piping
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.
			abbrev	varchar(20)	YES		An abbreviation for another field in the table. The exact meaning depends on the table in which it is used.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_isolation_status	abbrev	ASC	Yes	NONCLUSTERED
PK__isolation_status__1B8A8CC0	isolation_status	ASC	Yes	CLUSTERED

dbo.log_changes

Records most data modifications including deletions of data for key tables in the TCD. Data is recorded via the target table's trigger code.

I F P	Column name	Data type	Nulls	Default	Description
 	log_changes_id	int	NO		Unique record identifier.
	TableName	varchar(30)	NO		Name of the table that was modified
	FieldName	varchar(60)	YES		Name of the field that was modified
	Key1	varchar(255)	NO		Fields that make the record unique
	Key1Value	varchar(255)	NO		Values in the fields making the record unique
	Action	varchar(10)	NO		Action taken (modified or deleted)
	OldValue	varchar(2000)	YES		Value before the modification or deletion occurred
	NewValue	varchar(2000)	YES		Present value (if modification occurred)
	ActionDate	datetime	NO		Date the action was taken
	ActionUser	varchar(30)	NO		Person who modified or deleted the record
	waste_site_id	varchar(30)	YES		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	con_name	varchar(60)	YES		Name of the constituent used for reporting purposes. Typically, the name comes from the Chemical Abstract Services (CAS) listing.
	con_id	varchar(15)	YES		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
	analysis_method_id	varchar(10)	YES		Analysis method id includes those properties that identify the class, procedure, and preparation that describe how the results were obtained and analyzed for the sample. Analysis method IDs trace back to lab procedures such as LA-505-101 or PNL-ALO-103.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_analysis_method_id	analysis_method_id	ASC		NONCLUSTERED
i_con_id	con_id	ASC		NONCLUSTERED
i_con_name	con_name	ASC		NONCLUSTERED
i_log_changes_id	log_changes_id	ASC	Yes	NONCLUSTERED
i_waste_site_id	waste_site_id	ASC		NONCLUSTERED
PK__log_changes__62EF9734	log_changes_id	ASC	Yes	CLUSTERED

dbo.log_insert_counts

Keeps a log of the number of records inserted into tables loaded from laboratory data.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			log_insert_counts_id	int	NO		Unique identified for the log_insert_counts table
			load_date	datetime	YES		Date record was loaded
			tank_dttm_id	varchar(100)	YES		Tank and date
			record_count	int	YES		Record count
			record_type	varchar(20)	YES		The name of the table that the records were inserted into.

dbo.means_and_variances_detail

Historical data loaded using the Statsprod MS Access application. No data has been loaded since 2006. The data is used by the AutoTCR Means and Confidence Interval report.

I F P	Column name	Data type	Nulls	Default	Description
 	waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
 	con_id	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
	method_group_id	varchar(30)	NO		Name assigned to the general group of laboratory analytical methods to which the current method belongs. Examples include: ICP:A and AA:F.
	phase	varchar(10)	YES		Predominate solid waste phase in the tank as judged by the largest of the salt volume or sludge volume.
	agg_level	varchar(20)	NO		Label or descriptor applied to the whole, i.e. "aggregate", sample before separation into portions or fractions prior to analysis. See /dictionary/AggLevel.asp >Aggregation Level Definitions for more information.
	mean	float	YES		Mean concentration for sample data for the tank.
	units	varchar(10)	NO		Concentration units of the reported value or the reporting limit.
	std_dev	float	YES		Standard deviation for the mean.
	degrees_of_freedom	int	YES		Degrees of freedom.
	lower_limit_95	float	YES		Lower 95% confidence limit.
	upper_limit_95	float	YES		Upper 95% confidence limit.
	reference	varchar(20)	YES		Reference document from which data was gathered.
 	reference_date	datetime	NO		Date of the reference document.
	number_of_detected_values	int	YES		Number of measurements above detection limits
	number_of_non_detected_values	int	YES		Number of measurements below detection limits
	comments	varchar(255)	YES		General comment by statisticians.
	statistical_model	varchar(60)	YES		Model that was fit to data.
	variance_riser	float	YES		Variance from riser effect.
	variance_sampling_event	float	YES		Variance from sampling event effect.
	variance_segment_name	float	YES		Variance from segment effect.
	variance_lab_samp_id	float	YES		Variance from lab sample effect.
	variance_residual	float	YES		Model error variance.
	record_id	int	NO		The value of the modified record's Key_Name.
	load_date_time	datetime	YES		The date and time that the record was loaded into the TCD.
	last_modified_date_time	datetime	YES		Date on which the corresponding change record was entered into the TCD. This same date is applied to each analyte changed as part of the respective change log. Once the data has been entered into TCD, the change is immediately visible via the queries available through the TWINS2 interface.
	temp_change_reason	varchar(255)	YES		Temporary change reason
	sampling_events	varchar(255)	YES		Triple pipe delimited list of sampling events covered by the means and variances data
	mv_id	int	NO		Identity field for means and variances detail record
	sampling_begin_date	datetime	YES		Date that the sampling began.
	sampling_end_date	datetime	YES		Date that the sampling ended.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_means_and_variances_detail	waste_site_id	ASC	Yes	CLUSTERED
i_means_and_variances_detail	con_id	ASC	Yes	CLUSTERED
i_means_and_variances_detail	method_group_id	ASC	Yes	CLUSTERED
i_means_and_variances_detail	phase	ASC	Yes	CLUSTERED

dbo.means_and_variances_header

Historical data loaded using the Statsprod MS Access application. No data has been loaded since 2006. The data is used by the AutoTCR Means and Confidence Interval report.

I	F	P	Column name	Data type	Nulls	Default	Description
			waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
			reference	varchar(20)	YES		Reference document from which data was gathered.
			reference_date	datetime	NO		Date of the reference document.
			posted_to_tcd_date	datetime	YES		Date means and variances data was posted to TCD
			comment	varchar(255)	YES		Comment

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_means_and_variances_header	waste_site_id	ASC	Yes	CLUSTERED
i_means_and_variances_header	reference_date	ASC	Yes	CLUSTERED
PK__means_and_varian__579F76C7	waste_site_id	ASC	Yes	NONCLUSTERED
PK__means_and_varian__579F76C7	reference_date	ASC	Yes	NONCLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__means_and__waste__5F0B8E65	waste_site_id	i_waste_tank ()	

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.means_and_variances_detail	FK__means_and_varian__5E176A2C	i_means_and_variances_header

dbo.means_variances_detail_mod

I F P	Column name	Data type	Nulls	Default	Description
	waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	con_id	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
	con_name	varchar(60)	NO		Name of the constituent used for reporting purposes. Typically, the name comes from the Chemical Abstract Services (CAS) listing.
	molecular_formula	varchar(60)	YES		Molecular formula for this constituent.
	method_group_id	varchar(30)	NO		Name assigned to the general group of laboratory analytical methods to which the current method belongs. Examples include: ICP:A and AA:F.
	phase	varchar(10)	YES		Predominate solid waste phase in the tank as judged by the largest of the salt volume or sludge volume.
	agg_level	varchar(20)	NO		Label or descriptor applied to the whole, i.e. "aggregate", sample before separation into portions or fractions prior to analysis. See /dictionary/AggLevel.asp >Aggregation Level Definitions for more information.
	mean	float	YES		Mean concentration for sample data for the tank.
	units	varchar(10)	NO		Concentration units of the reported value or the reporting limit.
	std_dev	float	YES		Standard deviation for the mean.
	degrees_of_freedom	int	YES		Degrees of freedom.
	lower_limit_95	float	YES		Lower 95% confidence limit.
	upper_limit_95	float	YES		Upper 95% confidence limit.
	reference	varchar(20)	YES		Reference document from which data was gathered.
	reference_date	datetime	YES		Date of the reference document.
	number_of_detected_values	int	YES		Number of measurements above detection limits
	number_of_non_detected_values	int	YES		Number of measurements below detection limits
	comments	varchar(255)	YES		General comment by statisticians.
	statistical_model	varchar(60)	YES		Model that was fit to data.
	variance_riser	float	YES		Variance from riser effect.
	variance_sampling_event	float	YES		Variance from sampling event effect.
	variance_segment_name	float	YES		Variance from segment effect.
	variance_lab_samp_id	float	YES		Variance from lab sample effect.
	variance_residual	float	YES		Model error variance.
	record_id	int	NO		The value of the modified record's Key_Name.
	load_date_time	datetime	YES		The date and time that the record was loaded into the TCD.
	last_modified_date_time	datetime	YES		Date on which the corresponding change record was entered into the TCD. This same date is applied to each analyte changed as part of the respective change log. Once the data has been entered into TCD, the change is immediately visible via the queries available through the TWINS2 interface.
	change_reason	varchar(255)	NO		Field that allows the data entry person to enter a reason for the change that was made to the original record.
	change_user_name	varchar(20)	NO		The user_name of the person who changed the record.
	change_date_time	datetime	NO		The date and time that a change was made to the record.
	change_action	varchar(8)	NO		The type of change that was made to an analysis_result record - delete or update.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
------------	-------------	----------------	-----------	------------

i_means_variances_detail_mod	record_id	ASC	Yes	CLUSTERED
i_means_variances_detail_mod	change_date_time	ASC	Yes	CLUSTERED

 **Foreign keys**

<u>Constraint name</u>	<u>Column name</u>	<u>Reference</u>	<u>Description</u>
FK__means_var__chang__5FFB29E	change_action	i_change_action ()	

dbo.operator

Mathematical operator used in unit conversion

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			operator	varchar(10)	NO		Limiting operator for the DQO criteria. The values are >, <, or =.
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_operator	operator	ASC	Yes	CLUSTERED
PK__operator__58939B00	operator	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.unit_conversion	FK__unit_conv__opera__19382FFA	i_operator

dbo.parent_table

Parent table lookup for samples

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			parent_table	varchar(30)	NO		The name of the table in which the parent sample number is located.
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_parent_table	parent_table	ASC	Yes	CLUSTERED
PK__parent_table__5987BF39	parent_table	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.sample	fk_parent_table	i_parent_table

dbo.phase

Physical state

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			phase_assignment	varchar(30)	NO		Description which designates the phase assigned to the sample portion being measured or analyzed.
			phase_description	varchar(255)	YES		Description of the phase

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_phase_assignment	phase_assignment	ASC	Yes	CLUSTERED
PK_phase	phase_assignment	ASC	Yes	NONCLUSTERED

dbo.physical_state

Physical state such as liquid or solid

I	F	P	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			physical_state	varchar(30)	NO		Liquid/Solid, the sample analysis state.
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_physical_state	physical_state	ASC	Yes	NONCLUSTERED
PK__physical_state__5B7007AB	physical_state	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.std_constituent_units	FK__std_const__physi__080DA3F8	PK__physical_state__5B7007AB
dbo.sample	fk_physical_state	PK__physical_state__5B7007AB
dbo.qc_limits_am	FK__qc_limits__phase__60F3D6D7	PK__physical_state__5B7007AB

dbo.privitization

Used by the *TCDLoader2 ASP.NET application*.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			privitization	varchar(20)	NO		Privitization status
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

dbo.project

Project information

I	F	P	Column name	Data type	Nulls	Default	Description
			proj_id	varchar(40)	NO		Identifies the project that authorized the data to be taken
			proj_type	varchar(40)	YES		The project type of a sample.
			proj_name_short	varchar(40)	YES		Name of project that caused the sample to be analyzed
			proj_name_long	varchar(60)	YES		Project name
			proj_doc_name_short	varchar(40)	YES		Short name/description for a project document
			proj_doc_name_long	varchar(60)	YES		Long name/description for a project document
			proj_doc_date	datetime	YES		Project date
			proj_load_date_time	datetime	YES		Project load date
			proj_modified_date_time	datetime	YES		Project modified date
			proj_sponsor	varchar(40)	YES		Sponsor of the project for which this sample was taken..

Indexes

Index name	Column name	Sort direction	Is unique	Index type
PK__project__1D72D532	proj_id	ASC	Yes	CLUSTERED

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.sample	fk_proj_id	PK__project__1D72D532

dbo.project_type

Lookup list of project types.

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	proj_type	varchar(40)	YES		The project type of a sample.

dbo.qa_level

Lookup list of QA levels.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			qa_level_id	int	NO		Unique record identifier.
			qa_level	varchar(30)	NO		Either Opportunistic or Requested.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK__qa_level__2C00EBAB	qa_level	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.analysis_result	fk_qa_level	PK__qa_level__2C00EBAB

dbo.qa_type

Lookup table for QA types

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			qa_type	varchar(20)	NO		Type of quality assurance sample, if any, represented by the current sample number. Current values include: DIW BLANK, FIELD BLANK, HOT CELL BLANK, LAB BLANK, TRIP BLANK, BLIND, HYDROSTATIC HEAD FLUID, LINER LIQUID, SERIAL DILUTION, STANDARD, SYNTHETIC, and NONE.
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_qa_type	qa_type	ASC	Yes	CLUSTERED
PK__qa_type__5C642BE4	qa_type	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.sample	fk_qa_type	i_qa_type

dbo.qc_limits_am

Upper and lower limits of analytical methods.

I	F	P	Column name	Data type	Nulls	Default	Description
			method_group_id	varchar(30)	NO		Name assigned to the general group of laboratory analytical methods to which the current method belongs. Examples include: ICP:A and AA:F.
			standard_rec_lower_bound	float	YES		The standard recovery lower bound of quality control limits.
			standard_rec_upper_bound	float	YES		The standard recovery upper bound of quality control limits.
			spike_rec_lower_bound	float	YES		The spike recovery lower bound of quality control limits.
			spike_rec_upper_bound	float	YES		The spike recovery upper bound of quality control limits.
			rpd_limit	float	YES		Reporting limit of a relative percent difference value.
			phase	varchar(30)	NO		Predominate solid waste phase in the tank as judged by the largest of the salt volume or sludge volume.
			qc_source	varchar(20)	NO		
			qc_limits_am_id	int	NO		

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_qc_limits_am	method_group_id	ASC	Yes	CLUSTERED
i_qc_limits_am	phase	ASC	Yes	CLUSTERED
i_qc_limits_am	qc_source	ASC	Yes	CLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__qc_limits__phase__60F3D6D7	phase	PK__physical_state__5B7007AB (dbo.physical_state)	

Triggers

Trigger name:	dbo.t_qc_limits_am_del	Created on:	27 Apr 2000
Trigger type:	DELETE	Trigger active:	Yes
Trigger name:	dbo.t_qc_limits_am_ins_upd	Created on:	12 Apr 2001
Trigger type:	INSERT, UPDATE	Trigger active:	Yes

dbo.qualifier_type

Current or Legacy.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			qualifier_type	varchar(20)	NO		Current or Legacy

dbo.qualifiers

Lookup table of valid qualifiers.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			qualifier	varchar(10)	YES		Qualifies the concentration of the constituent.
			description	varchar(1000)	YES		Brief narrative describing a specific instance in this database table.
			qualifier_type	varchar(20)	YES		Current or Legacy values.
			qualifier_id	int	NO		Unique record identifier.
			sort_order	int	YES		Used for sorting the records in this table or view.

dbo.reporting_day

Lookup table for reporting day

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			reporting_day	varchar(15)	NO		Day in the analysis cycle when analysis was reported
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_reporting_day	reporting_day	ASC	Yes	CLUSTERED
PK__reporting_day__5D58501D	reporting_day	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.sample	fk_reporting_day	i_reporting_day

dbo.ResolveLog

Used by the Resolve ASP application. Log of the changes to the analysis_result review_status values.

I	F	P	Column name	Data type	Nulls	Default	Description
			waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
			analysis_result_id	int	NO		Unique sequential number identifier to a single analytical record within the database.
			old_reviewer	varchar(60)	YES		Value of the reviewer prior to the update.
			new_reviewer	varchar(60)	YES		Value of the reviewer after the update was performed.
			old_review_status	varchar(60)	YES		Value of the status prior to the update.
			new_review_status	varchar(60)	YES		Value of the status after the update was performed.
			old_review_comment	varchar(2000)	YES		Value of the comment prior to the update.
			new_review_comment	varchar(2000)	YES		Value of the comment after the update was performed.
			old_review_date	datetime	YES		Value of the date prior to the update.
			new_review_date	datetime	YES		Value of the date after the update was performed.
			review_event	varchar(60)	YES		Indication of what process changed the review status.
			id	int	NO		Unique identifier of the set

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_id	id	ASC	Yes	NONCLUSTERED
i_waste_site_id	waste_site_id	ASC		NONCLUSTERED

dbo.result_type

Lookup table for result types. Used by the TCDLoader2 ASP.NET application.

I	F	P	Column name	Data type	Nulls	Default	Description
			result_type	varchar(20)	NO		Description of the type of results the analysis produced. Examples include: PRIMARY, PRIMARY#2
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_result_type	result_type	ASC	Yes	CLUSTERED
PK__result_type__5E4C7456	result_type	ASC	Yes	NONCLUSTERED

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.static_analysis_result_units	FK__static_an__resul__0531374D	i_result_type
dbo.analysis_result	fk_result_type	i_result_type
dbo.static_analysis_result_units	FK__static_an__resul__7BA7CD13	i_result_type

dbo.samp_attr

Sample attributes

I	F	P	Column name	Data type	Nulls	Default	Description
			samp_attr_identity	int	NO		Unique record identity.
			samp_attr_id	varchar(40)	NO		Name of sample attribute or name of constituent.
			samp_num	varchar(12)	NO		Unique alpha-numeric ID assigned to identify a sample in the database.
			samp_attr_text	varchar(255)	YES		Text of attribute
			samp_attr_num	float	YES		Value of sample attribute (if any)
			samp_attr_units	varchar(10)	YES		Units of attribute value (if any)
			samp_attr_load_date_time	datetime	YES		The date and time sample attributes were loaded into the database.
			samp_attr_modified_date_time	datetime	YES		The date and time at which sample attribute record was modified.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_samp_attr	samp_num	ASC		NONCLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__samp_attr__samp___62DC1F49	samp_attr_units	i_units ()	
FK__samp_attr__samp___61E7FB10	samp_attr_id	PK__samp_attr_lkup___1E66F96B (dbo.samp_attr_lkup)	
FK__samp_attr__samp___194BA7E5	samp_num	i_sample ()	

dbo.samp_attr_ikup

Lookup table for sample attributes

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			samp_attr_id	varchar(40)	NO		Name of sample attribute or name of constituent.
			samp_attr_short_name	varchar(40)	NO		Sample attribute name
			samp_attr_long_name	varchar(60)	NO		Description name for an attribute associated with a sample.
			samp_attr_default_units	varchar(10)	NO		Default units for the sample attribute
			samp_attr_type	varchar(10)	NO		Sample attribute type

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK__samp_attr_ikup__1E66F96B	samp_attr_id	ASC	Yes	CLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.samp_attr	FK__samp_attr__samp__61E7FB10	PK__samp_attr_ikup__1E66F96B

dbo.sample

TCD sample

I	F	P	Column name	Data type	Nulls	Default	Description
			samp_num	varchar(12)	NO		Unique alpha-numeric ID assigned to identify a sample in the database.
			sample_portion	varchar(20)	NO		A part of a sample that was taken.
			parent_table	varchar(30)	NO		The name of the table in which the parent sample number is located.
			agg_level	varchar(20)	NO		Label or descriptor applied to the whole, i.e. "aggregate", sample before separation into portions or fractions prior to analysis. See /dictionary/AggLevel.asp >Aggregation Level Definitions for more information.
			samp_date_time	datetime	YES		The date and time the sample was taken.
			physical_state	varchar(30)	NO		Liquid/Solid, the sample analysis state.
			sampler	varchar(20)	YES		Name of person who collected sample.
			lab_received_date	datetime	YES		Date sample was received at laboratory. Format is mm/dd/yy hh:mm.
			log_page	varchar(10)	YES		Log book page number containing specific sampling event information.
			log_id	varchar(20)	YES		Identification number of the log book containing sampling event information.
			doc_loc	varchar(150)	NO		Document in which analysis result is recorded.
			reporting_day	varchar(15)	YES		Day in the analysis cycle when analysis was reported
			qa_type	varchar(20)	NO		Type of quality assurance sample, if any, represented by the current sample number. Current values include: DIW BLANK, FIELD BLANK, HOT CELL BLANK, LAB BLANK, TRIP BLANK, BLIND, HYDROSTATIC HEAD FLUID, LINER LIQUID, SERIAL DILUTION, STANDARD, SYNTHETIC, and NONE.
			comp_name	varchar(255)	YES		Composite name
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.
			comment	varchar(255)	YES		Comment
			load_date_time	datetime	YES		The date and time that the record was loaded into the TCD.
			last_modified_date_time	datetime	YES		Date on which the corresponding change record was entered into the TCD. This same date is applied to each analyte changed as part of the respective change log. Once the data has been entered into TCD, the change is immediately visible via the queries available through the TWINS2 interface.
			access_level	tinyint	NO	/***** Object: Default dbo.d_ac	Access control mechanism for data in database table.
			proj_id	varchar(40)	YES		Identifies the project that authorized the data to be taken
			phase_assignment	varchar(30)	YES		Description which designates the phase assigned to the sample portion being measured or analyzed.
			data_status	varchar(30)	YES	('current')	Data is current (no transfers since sampled), pre-transfer (previous to a transfer), or useable even if prior to a transfer.
			sample_data_file_name	varchar(255)	YES		Status of the data defined as follows: 1) Pre-transfer -- analytical data that is associated with a sampling event that precedes a waste transfer into the tank or a complete transfer out of the tank; 2) Utilized/Pre-transfer -- sampling event that precedes a waste transfer but is still a component of the waste. Some or all data associated with this event are utilized in the BBI to estimate tank composition. 3) Current -- analytical data from samples taken after a transfer or containing waste unaffected by waste transfers. Origin of the data.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
------------	-------------	----------------	-----------	------------

i_sample	samp_num	ASC	Yes	CLUSTERED
PK__sample__47477CBF	samp_num	ASC	Yes	NONCLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
fk_proj_id	proj_id	PK__project__1D72D532 (dbo.project)	
fk_physical_state	physical_state	PK__physical_state__5B7007AB (dbo.physical_state)	
fk_sample_portion	sample_portion	i_sample_portion ()	
fk_parent_table	parent_table	i_parent_table ()	
fk_samp_num	samp_num	i_sample_number_master ()	
fk_agg_level	agg_level	i_agg_level ()	
fk_qa_type	qa_type	i_qa_type ()	
fk_reporting_day	reporting_day	i_reporting_day ()	

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.sample_analysis	FK__sample_an__samp__1A3FCC1E	i_sample
dbo.sample_relation	FK__sample_re__samp__17635F73	i_sample
dbo.samp_attr	FK__samp_attr__samp__194BA7E5	i_sample
dbo.sample_relation	FK__sample_re__relat__185783AC	i_sample

Triggers

Trigger name:	dbo.t_sample_del	Created on:	14 Jul 2006
Trigger type:	DELETE	Trigger active:	Yes
Trigger name:	dbo.t_sample_ins_upd	Created on:	14 Jul 2006
Trigger type:	INSERT, UPDATE	Trigger active:	Yes

dbo.sample_analysis

Analysis performed. Used by the TCDLoader2 ASP.NET application.

I	F	P	Column name	Data type	Nulls	Default	Description
			samp_num	varchar(12)	NO		Unique alpha-numeric ID assigned to identify a sample in the database.
			analysis_method_id	int	NO		Analysis method id includes those properties that identify the class, procedure, and preparation that describe how the results were obtained and analyzed for the sample. Analysis method IDs trace back to lab procedures such as LA-505-101 or PNL-ALO-103.
			lab_code	varchar(10)	NO		The name of the laboratory that analyzed the sample.
			sample_dilution_key	tinyint	NO		Sample dilution key
			dilut_factor	float	YES		Dilution factor entered as a single number when a sample has been diluted for analysis.. The number 100 is for a 1 to 100 dilution of the sample, a 0.1 is entered for a 10 to 1, and if a sample was not diluted, either the field is left blank or a 1 is entered.
			lab_samp_id	varchar(60)	YES		The Sample Number assigned by the lab.
			samp_prep_date_time	datetime	YES		Date and time the sample preparation took place. Format is mm/dd/yy hh:mm.
			secondary_samp_prep_type	varchar(20)	YES		Secondary sample prep.
			batch_id	varchar(20)	YES		Identification of a group of analyses that were performed together and having something in common such as time of analysis or instrument.
			analyst	varchar(20)	YES		Name or other identification of the technician who performed the analysis.
			file_id	varchar(255)	YES		File ID used in the lab
			comment	varchar(255)	YES		Comment
			load_date_time	datetime	YES		The date and time that the record was loaded into the TCD.
			last_modified_date_time	datetime	YES		Date on which the corresponding change record was entered into the TCD. This same date is applied to each analyte changed as part of the respective change log. Once the data has been entered into TCD, the change is immediately visible via the queries available through the TWINS2 interface.
			access_level	tinyint	NO	/****** Object: Default dbo.d_ac	Access control mechanism for data in database table.
			sample_analysis_id	int	NO		Identity field for the sample_analysis table.
			project_id	int	YES		Project ID
			sample_analysis_to_load_id	int	YES		Unique record identity.
			sample_data_file_name	varchar(255)	YES		Origin of the data.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_sample_analysis	sample_analysis_id	ASC	Yes	NONCLUSTERED
i1_sample_analysis	samp_num	ASC	Yes	CLUSTERED
i1_sample_analysis	analysis_method_id	ASC	Yes	CLUSTERED
i1_sample_analysis	lab_code	ASC	Yes	CLUSTERED
i1_sample_analysis	sample_dilution_key	ASC	Yes	CLUSTERED
i2_sample_analysis	samp_num	ASC		NONCLUSTERED
i2_sample_analysis	analysis_method_id	ASC		NONCLUSTERED
i2_sample_analysis	dilut_factor	ASC		NONCLUSTERED
i3_sample_analysis	samp_num	ASC		NONCLUSTERED
i3_sample_analysis	analysis_method_id	ASC		NONCLUSTERED
i4_sample_analysis	samp_num	ASC		NONCLUSTERED
i5_sample_analysis	analysis_method_id	ASC		NONCLUSTERED
PK__sample_analysis__621D053A	samp_num	ASC	Yes	NONCLUSTERED
PK__sample_analysis__621D053A	analysis_method_id	ASC	Yes	NONCLUSTERED

PK_sample_analysis__621D053A	lab_code	ASC	Yes	NONCLUSTERED
PK_sample_analysis__621D053A	sample_dilution_key	ASC	Yes	NONCLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK_sample_an_samp__1A3 FCC1E	samp_num	i_sample ()	
FK_sample_an_lab_c__65B8 8BF4	lab_code	i_laboratory ()	
FK_sample_an_analy__64C4 67BB	analysis_method_id	i1_analysis_method ()	

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.analysis_result	fk_sample_analysis_id	i_sample_analysis

Triggers

Trigger name:	dbo.t_sample_analysis_del	Created on:	14 Jul 2006
Trigger type:	DELETE	Trigger active:	Yes
Trigger name:	dbo.t_sample_analysis_ins_upd	Created on:	14 Jul 2006
Trigger type:	INSERT, UPDATE	Trigger active:	Yes

dbo.sample_analysis_to_load

Holding table used by the TCDLoader2 ASP.NET application.

I F P	Column name	Data type	Nulls	Default	Description
	sample_analysis_to_load_id	int	NO		Unique record indentify.
	lab_samp_id	varchar(255)	YES		The Sample Number assigned by the lab.
	dilut_factor	varchar(255)	YES		Dilution factor entered as a single number when a sample has been diluted for analysis.. The number 100 is for a 1 to 100 dilution of the sample, a 0.1 is entered for a 10 to 1, and if a sample was not diluted, either the field is left blank or a 1 is en
	lab_analysis_procedure	varchar(255)	YES		
	primary_sample_preparation	varchar(255)	YES		
	secondary_samp_prep_type	varchar(255)	YES		Secondary sample prep.
	samp_prep_date_time	varchar(255)	YES		Date and time the sample preparation took place. Format is mm/dd/yy hh:mm.
	analyst	varchar(255)	YES		Name or other identification of the technician who preformed the analysis.
	batch_id	varchar(255)	YES		Identification of a group of analyses that were performed together and having something in common such as time of analysis or instrument.
	reference	varchar(255)	YES		Reference document from which data was gathered.
	file_id	varchar(255)	YES		File ID used in the lab
	blank	varchar(255)	YES		
	comment	varchar(255)	YES		Comment
	analysis_method_id	varchar(255)	YES		Analysis method id includes those properties that identify the class, procedure, and preparation that describe how the results were obtained and analyzed for the sample. Analysis method IDs trace back to lab procedures such as LA-505-101 or PNL-ALO-103.
	samp_num	varchar(255)	YES		Unique alpha-numeric ID assigned to identify a sample in the database.
	load_date_time	datetime	YES		The date and time that the record was loaded into the TCD.
	ready_to_load	char(1)	YES		Flag indicating record is ready to be loaded
	sample_data_file_name	varchar(255)	YES		Origin of the data.
	review_comment	varchar(6000)	YES		Comment added by reviewer of this analysis result record.
	is_preliminary_yn	char(1)	YES		
	review_yn	char(1)	YES		
	analysis_results_have_errors	char(1)	YES		
	sample_dilution_key	int	YES		Sample dilution key

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_id	sample_analysis_to_load_id	ASC	Yes	NONCLUSTERED
i_sample_data_file_name	sample_data_file_name	ASC		NONCLUSTERED

dbo.sample_description

Contains descriptive data about each sample consolidated into a single entry, regardless of whether the sample is a supernate, surface or core samples Used by the TCDLoader2 ASP.NET application.

I F P	Column name	Data type	Nulls	Default	Description
	samp_num	varchar(12)	NO		Unique alpha-numeric ID assigned to identify a sample in the database.
	waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	sample_type	varchar(30)	YES		Type of sample. Current values are: CORE, SUPERNATE, and SURFACE.
	agg_level	varchar(20)	NO		Label or descriptor applied to the whole, i.e. "aggregate", sample before separation into portions or fractions prior to analysis. See /dictionary/AggLevel.asp >Aggregation Level Definitions for more information.
	device_type	varchar(30)	NO		Description of the type of container used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
	sampling_event_name	varchar(20)	NO		Core ID, supernate sample ID, surface sample ID, tank composite name, or core composite name.
	segment_name	varchar(20)	NO		Segment ID or core composite name.
	sample_portion	varchar(20)	NO		A part of a sample that was taken.
	tank_riser	varchar(12)	YES		Riser number on the waste tank.
	sample_level	int	YES		Height at which the sample was taken, measured from the bottom of the waste site.
	sample_level_units	varchar(10)	YES		Units of measure associated with the sample level.
	sampling_begin_date	datetime	YES		Date the sampling started. Format is mm/dd/yyyy.
	sampling_end_date	datetime	YES		Date the sampling was completed. Format is mm/dd/yyyy.
	transfer_since_sample	varchar(2)	YES		A waste transfer to or from this tank has occurred since this sampling_event.
	appearance	varchar(255)	YES		Description of the physical appearance of the sample.
	recovery_percent	tinyint	YES		Percentage of the expected sample size that was recovered during removal from the tank.
	sampling_event_comment	varchar(255)	YES		A comment related to a sampling event.
	sample_type_comment	varchar(255)	YES		Comment concerning the sample type.
	sample_date_time	datetime	YES		Date and time the sample was collected or created. Format is mm/dd/yy hh:mm.
	physical_state	varchar(30)	NO		Liquid/Solid, the sample analysis state.
	qa_type	varchar(20)	YES		Type of quality assurance sample, if any, represented by the current sample number. Current values include: DIW BLANK, FIELD BLANK, HOT CELL BLANK, LAB BLANK, TRIP BLANK, BLIND, HYDROSTATIC HEAD FLUID, LINER LIQUID, SERIAL DILUTION, STANDARD, SYNTHETIC, and NONE.
	description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.
	doc_loc	varchar(150)	YES		Document in which analysis result is recorded.
	lab_received_date	datetime	YES		Date sample was received at laboratory. Format is mm/dd/yy hh:mm.
	post_tpa_flag	varchar(3)	YES		Data and information flagged TPA in this database are post-1989 analytical data and are recognized by the Washington State Department of Ecology for use in Tri- Party Agreement (TPA) and regulatory mandated documents. This data may be used for production of TPA deliverable documents.
	access_level	tinyint	NO		Access control mechanism for data in database table.
	privitization	varchar(4)	YES		Privitization status
	proj_id	varchar(40)	YES		Identifies the project that authorized the data to be taken
	proj_type	varchar(40)	YES		The project type of a sample.
	proj_name_short	varchar(40)	YES		Name of project that caused the sample to be analyzed
	proj_doc_name_short	varchar(40)	YES		Short name/description for a project document
	proj_sponsor	varchar(40)	YES		Sponsor of the project for which this sample was taken..

volume_in_ml	float	YES	Volume of a sample in milliliters. Usually is type float, but appears as a varchar in some tables.
weight_in_grams	float	YES	The weight in grams -- This field is type float in most tables but is type char in some.
formatted_sample_location	varchar(255)	YES	The sample location formatted for display.
formatted_sample_event	varchar(255)	YES	The sampling event name formatted for display.
phase_assignment	varchar(30)	YES	Description which designates the phase assigned to the sample portion being measured or analyzed.
in_current_bbi	varchar(30)	YES	Is this sampling event current in BBI
data_status	varchar(30)	YES	Data is current (no transfers since sampled), pre-transfer (previous to a transfer), or useable even if prior to a transfer

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_sample_description	samp_num	ASC	Yes	NONCLUSTERED
i1_sample_description	waste_site_id	ASC		NONCLUSTERED
PK_sample_description	samp_num	ASC	Yes	NONCLUSTERED

dbo.sample_description_to_load

Holding table used by the TCDLoader2 ASP.NET application.

I F P	Column name	Data type	Nulls	Default	Description
	sample_description_to_load_id	int	NO		Unique record identifier.
	record_type	varchar(255)	YES		Record type
	samp_num	varchar(255)	YES		Unique alpha-numeric ID assigned to identify a sample in the database.
	physical_state	varchar(255)	YES		Liquid/Solid, the sample analysis state.
	sample_portion	varchar(255)	YES		A part of a sample that was taken.
	description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.
	parent_table	varchar(255)	YES		The name of the table in which the parent sample number is located.
	samp_date_time	varchar(255)	YES		The date and time the sample was taken.
	lab_received_date	varchar(255)	YES		Date sample was received at laboratory. Format is mm/dd/yy hh:mm.
	log_page	varchar(255)	YES		Log book page number containing specific sampling event information.
	log_id	varchar(255)	YES		Identification number of the log book containing sampling event information.
	sampler	varchar(255)	YES		Name of person who collected sample.
	doc_loc	varchar(255)	YES		Document in which analysis result is recorded.
	comment	varchar(255)	YES		Comment
	reporting_day	varchar(255)	YES		Day in the analysis cycle when analysis was reported
	agg_level	varchar(255)	YES		Label or descriptor applied to the whole, i.e. "aggregate", sample before separation into portions or fractions prior to analysis. See /dictionary/AggLevel.asp for more information.
	qa_type	varchar(255)	YES		Type of quality assurance sample, if any, represented by the current sample number. Current values include: DIW BLANK, FIELD BLANK, HOT CELL BLANK, LAB BLANK, TRIP BLANK, BLIND, HYDROSTATIC HEAD FLUID, LINER LIQUID, SERIAL DILUTION, STANDARD, SYNTHETIC, and NONE.
	comp_name	varchar(255)	YES		Composite name
	load_date_time	datetime	YES		The date and time that the record was loaded into the TCD.
	ready_to_load	char(1)	YES		Flag indicating record is ready to be loaded
	sample_data_file_name	varchar(255)	YES		Origin of the data.
	review_comment	varchar(6000)	YES		Comment added by reviewer of this analysis result record.
	is_preliminary_yn	char(1)	YES		
	review_yn	char(1)	YES		
	sample_relations_have_errors	char(1)	YES		

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_id	sample_description_to_load_id	ASC	Yes	NONCLUSTERED
i_sample_data_file_name	sample_data_file_name	ASC		NONCLUSTERED

dbo.sample_no_proj_id

The *p_rep_tcd_denorm_root_samps* stored procedure stores sample data that has no project_id values.

I	F	P	Column name	Data type	Nulls	Default	Description
			samp_num	varchar(12)	NO		Unique alpha-numeric ID assigned to identify a sample in the database.
			sample_portion	varchar(20)	NO		A part of a sample that was taken.
			parent_table	varchar(30)	NO		The name of the table in which the parent sample number is located.
			agg_level	varchar(20)	NO		Label or descriptor applied to the whole, i.e. "aggregate", sample before separation into portions or fractions prior to analysis. See /dictionary/AggLevel.asp >Aggregation Level Definitions for more information.
			samp_date_time	datetime	YES		The date and time the sample was taken.
			physical_state	varchar(30)	YES		Liquid/Solid, the sample analysis state.
			sampler	varchar(20)	YES		Name of person who collected sample.
			lab_received_date	datetime	YES		Date sample was received at laboratory. Format is mm/dd/yy hh:mm.
			log_page	varchar(10)	YES		Log book page number containing specific sampling event information.
			log_id	varchar(20)	YES		Identification number of the log book containing sampling event information.
			doc_loc	varchar(150)	NO		Document in which analysis result is recorded.
			reporting_day	varchar(15)	YES		Day in the analysis cycle when analysis was reported
			qa_type	varchar(20)	NO		Type of quality assurance sample, if any, represented by the current sample number. Current values include: DIW BLANK, FIELD BLANK, HOT CELL BLANK, LAB BLANK, TRIP BLANK, BLIND, HYDROSTATIC HEAD FLUID, LINER LIQUID, SERIAL DILUTION, STANDARD, SYNTHETIC, and NONE.
			comp_name	varchar(255)	YES		Composite name
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.
			comment	varchar(255)	YES		Comment
			load_date_time	datetime	YES		The date and time that the record was loaded into the TCD.
			last_modified_date_time	datetime	YES		Date on which the corresponding change record was entered into the TCD. This same date is applied to each analyte changed as part of the respective change log. Once the data has been entered into TCD, the change is immediately visible via the queries available through the TWINS2 interface.
			access_level	tinyint	NO		Access control mechanism for data in database table.
			proj_id	varchar(40)	YES		Identifies the project that authorized the data to be taken
			phase_assignment	varchar(30)	YES		Description which designates the phase assigned to the sample portion being measured or analyzed.
			data_status	varchar(30)	YES		Data is current (no transfers since sampled), pre-transfer (previous to a transfer), or useable even if prior to a transfer
			sample_data_file_name	varchar(255)	YES		Origin of the data.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
PK_sample_no_proj_id	samp_num	ASC	Yes	CLUSTERED

dbo.sample_portion

Lookup list of valid parts of a sample (Filtered Solids, Settled Solids, ...)

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			sample_portion	varchar(20)	NO		A part of a sample that was taken.
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_sample_portion	sample_portion	ASC	Yes	CLUSTERED
PK__sample_portion__00A18C5A	sample_portion	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.sample	fk_sample_portion	i_sample_portion

dbo.sample_relation

Relationship between sampling events and samples/sample analysis

I F P	Column name	Data type	Nulls	Default	Description
	sample_relation_id	int	NO		Unique record identifier.
	samp_num	varchar(12)	NO		Unique alpha-numeric ID assigned to identify a sample in the database.
	related_samp_num	varchar(12)	NO		The code identifying a sample that consists at least in part of material from another sample identified in the record.
	load_date_time	datetime	YES		The date and time that the record was loaded into the TCD.
	last_modified_date_time	datetime	YES		Date on which the corresponding change record was entered into the TCD. This same date is applied to each analyte changed as part of the respective change log. Once the data has been entered into TCD, the change is immediately visible via the queries available through the TWINS2 interface.
	access_level	tinyint	NO		Access control mechanism for data in database table.
	parent_amount	float	YES		The amount of a sample that came from a parent sample.
	parent_amount_units	varchar(10)	YES		The units in which amount of the parent sample is reported.
	sample_data_file_name	varchar(255)	YES		Origin of the data.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i1_sample_relation	samp_num	ASC	Yes	CLUSTERED
i1_sample_relation	related_samp_num	ASC	Yes	CLUSTERED
i2_sample_relation	related_samp_num	ASC		NONCLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__sample_re__paren__76E317F6	parent_amount_units	i_units ()	
FK__sample_re__samp__17635F73	samp_num	i_sample ()	
FK__sample_re__relat__185783AC	related_samp_num	i_sample ()	

Triggers

Trigger name:	dbo.t_sample_relation_ins_upd	Created on:	14 Jul 2006
Trigger type:	INSERT, UPDATE	Trigger active:	Yes
Trigger name:	dbo.t_sample_relation_del	Created on:	14 Jul 2006
Trigger type:	DELETE	Trigger active:	Yes

dbo.sample_relation_to_load

Holding table used by the TCDLoader2 ASP.NET application.

I	F	P	Column name	Data type	Nulls	Default	Description
			sample_relation_to_load_id	int	NO		Unique record identifier.
			record_type	varchar(255)	YES		Record type
			samp_num	varchar(255)	YES		Unique alpha-numeric ID assigned to identify a sample in the database.
			related_samp_num	varchar(255)	YES		The code identifying a sample that consists at least in part of material from another sample identified in the record.
			load_date_time	datetime	YES		The date and time that the record was loaded into the TCD.
			ready_to_load	char(1)	YES		Flag indicating record is ready to be loaded
			sample_data_file_name	varchar(255)	YES		Origin of the data.
			review_comment	varchar(6000)	YES		Comment added by reviewer of this analysis result record.
			is_preliminary_yn	char(1)	YES		
			review_yn	char(1)	YES		

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_id	sample_relation_to_load_id	ASC	Yes	NONCLUSTERED
i_related_samp_num	related_samp_num	ASC		NONCLUSTERED
i_samp_num	samp_num	ASC		NONCLUSTERED
i_sample_data_file_name	sample_data_file_name	ASC		NONCLUSTERED

dbo.sample_type

Type of sample such as supernate or core

I	F	P	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			sample_type	varchar(30)	NO		Type of sample. Current values are: CORE, SUPERNATE, and SURFACE.
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_sample_type	sample_type	ASC	Yes	CLUSTERED
PK__sample_type__64F971E5	sample_type	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.device_type	FK__device_ty__sampl__51B19347	i_sample_type

dbo.sampling_event

Information on how the sample was taken in the field. Used by the TCDLoader2 ASP.NET application.

I	F	P	Column name	Data type	Nulls	Default	Description
			sampling_event_identity	int	NO		Unique record identifier.
			sampling_event_id	varchar(20)	NO		The identification number of a sampling event collected from a waste site. For cores, this will be a core number, for supernate samples, this will be a supernate sample number, and for surface samples, this will be a surface sample number.
			waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
			sample_type	varchar(30)	NO		Type of sample. Current values are: CORE, SUPERNATE, and SURFACE.
			sampling_begin_date	datetime	YES		Date the sampling started. Format is mm/dd/yyyy.
			sampling_end_date	datetime	YES		Date the sampling was completed. Format is mm/dd/yyyy.
			transfer_since_sample	varchar(3)	NO		A waste transfer to or from this tank has occurred since this sampling_event.
			device_type	varchar(30)	NO		Description of the type of container used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
			tank_riser	varchar(12)	YES		Riser number on the waste tank.
			comment	varchar(255)	YES		Comment
			access_level	tinyint	NO		Access control mechanism for data in database table.
			privitization	varchar(4)	YES		Privitization status
			sampling_begin_date_range	datetime	YES		Begin date range; used to override actual sampling_begin_date in the AutoTCR in order to group similar events together. Maintained by TCs.
			sampling_end_date_range	datetime	YES		End date range; used to override actual sampling_end_date in the AutoTCR in order to group similar events together. Maintained by TCs.
			transfer_since_sample_override	varchar(2)	YES		Override for the transfer since sample flag set up the computer. No longer used.
			override_reason	varchar(255)	YES		The reason that some value in the record has been manually overridden.
			in_current_bbi	varchar(30)	YES		Is this sampling event current in BBI

Indexes

	Index name	Column name	Sort direction	Is unique	Index type
	i_sampling_event	waste_site_id	ASC	Yes	NONCLUSTERED
	i_sampling_event	sampling_event_id	ASC	Yes	NONCLUSTERED
	PK__sampling_event__65ED961E	sampling_event_id	ASC	Yes	NONCLUSTERED
	PK__sampling_event__65ED961E	waste_site_id	ASC	Yes	NONCLUSTERED

Foreign keys

	Constraint name	Column name	Reference	Description
	FK__sampling_event__043D1314	device_type	PK__device_type__1A966887 (dbo.device_type)	
	FK__sampling_event__043D1314	sample_type	PK__device_type__1A966887 (dbo.device_type)	
	FK__sampling_event__7AB3A8DA	device_type	PK__device_type__1A966887 (dbo.device_type)	
	FK__sampling_event__7AB3A8DA	sample_type	PK__device_type__1A966887 (dbo.device_type)	
	FK__sampling__waste__0348EDB	waste_site_id	i_waste_tank ()	
	FK__sampling__waste__79BF84A1	waste_site_id	i_waste_tank ()	

Referencing tables

	Table name	Foreign key	Primary key or unique constraint
	dbo.tank_supernate_sample	FK__tank_supernate_s__6F0CEC04	PK__sampling_event__65ED961E
	dbo.tank_surface_sample	FK__tank_surface_sam__6D24A392	PK__sampling_event__65ED961E

dbo.tank_core

FK__tank_core__7001103D

PK__sampling_event__65ED961E

Triggers

Trigger name: dbo.t_sampling_event_ins_upd
Trigger type: INSERT, UPDATE

Created on: 14 Jul 2006
Trigger active: Yes

Trigger name: dbo.t_sampling_event_del
Trigger type: DELETE

Created on: 13 Jul 2006
Trigger active: Yes

dbo.sampling_event_denorm

Sampling Event data plus Core, Supernate and Surface sample data.

I	F	P	Column name	Data type	Nulls	Default	Description
			sampling_event_id	varchar(20)	NO		The identification number of a sampling event collected from a waste site. For cores, this will be a core number, for supernate samples, this will be a supernate sample number, and for surface samples, this will be a surface sample number.
			waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
			sample_type	varchar(30)	NO		Type of sample. Current values are: CORE, SUPERNATE, and SURFACE.
			sampling_begin_date	datetime	YES		Date the sampling started. Format is mm/dd/yyyy.
			sampling_end_date	datetime	YES		Date the sampling was completed. Format is mm/dd/yyyy.
			transfer_since_sample	varchar(2)	NO		A waste transfer to or from this tank has occurred since this sampling_event.
			device_type	varchar(30)	NO		Description of the type of container used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
			tank_riser	varchar(12)	YES		Riser number on the waste tank.
			se_comment	varchar(255)	YES		Further information to clarify the information in the table.
			tank_serial_num	int	YES		The serial number of a waste tank.
			segment_id	varchar(12)	YES		Unique identifier for a segment of a core.
			segment_serial_num	varchar(7)	YES		Log number of the waste site core from which the sample was obtained.
			sample_level	int	YES		Height at which the sample was taken, measured from the bottom of the waste site.
			sample_level_units	varchar(10)	YES		Units of measure associated with the sample level.
			recovery_percent	tinyint	YES		Percentage of the expected sample size that was recovered during removal from the tank.
			appearance	varchar(255)	YES		Description of the physical appearance of the sample.
			segment_comment	varchar(255)	YES		Further comments to clarify the information in the table.
			samp_num	varchar(12)	NO		Unique alpha-numeric ID assigned to identify a sample in the database.

dbo.sampling_event_to_load

Used by the TCDLoader2 ASP.NET application.

I	F	P	Column name	Data type	Nulls	Default	Description
			sampling_event_to_load_id	int	NO		
			record_type	varchar(255)	YES		Record type
			tank_farm_id	varchar(255)	YES		
			tank_id	varchar(255)	YES		Code that identifies each tank in a tank farm system.
			sampling_event_id	varchar(255)	YES		The identification number of a sampling event collected from a waste site. For cores, this will be a core number, for supernate samples, this will be a supernate sample number, and for surface samples, this will be a surface sample number.
			samp_num	varchar(255)	YES		Unique alpha-numeric ID assigned to identify a sample in the database.
			segment_id	varchar(255)	YES		Unique identifier for a segment of a core.
			appearance	varchar(255)	YES		Description of the physical appearance of the sample.
			load_date_time	datetime	YES		The date and time that the record was loaded into the TCD.
			ready_to_load	char(1)	YES		Flag indicating record is ready to be loaded
			sample_data_file_name	varchar(255)	YES		
			review_comment	varchar(6000)	YES		Comment added by reviewer of this analysis result record.
			is_preliminary_yn	char(1)	YES		
			review_yn	char(1)	YES		

Indexes

	Index name	Column name	Sort direction	Is unique	Index type
i_id		sampling_event_to_load_id	ASC	Yes	NONCLUSTERED
i_sample_data_file_name		sample_data_file_name	ASC		NONCLUSTERED

dbo.source_sample

Contains descriptive data about all samples and the waste site samples to which they relate, consolidated into a single table, regardless of whether the sample is a supernate, surface or core sample.

I F P	Column name	Data type	Nulls	Default	Description
	source_samp_num	varchar(12)	NO		
	samp_num	varchar(12)	NO		Unique alpha-numeric ID assigned to identify a sample in the database.
	waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	sample_type	varchar(20)	YES		Type of sample. Current values are: CORE, SUPERNATE, and SURFACE.
	agg_level	varchar(20)	NO		Label or descriptor applied to the whole, i.e. "aggregate", sample before separation into portions or fractions prior to analysis. See "dictionary/AggLevel.asp" >Aggregation Level Definitions for more information.
	device_type	varchar(30)	NO		Description of the type of container used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
	sampling_event_id	varchar(20)	NO		The identification number of a sampling event collected from a waste site. For cores, this will be a core number, for supernate samples, this will be a supernate sample number, and for surface samples, this will be a surface sample number .
	segment_id	varchar(12)	YES		Unique identifier for a segment of a core.
	sample_portion	varchar(20)	NO		A part of a sample that was taken.
	tank_riser	varchar(12)	YES		Riser number on the waste tank.
	sample_level	int	YES		Height at which the sample was taken, measured from the bottom of the waste site.
	sample_level_units	varchar(10)	YES		Units of measure associated with the sample level.
	sampling_begin_date	datetime	YES		Date the sampling started. Format is mm/dd/yyyy.
	sampling_end_date	datetime	YES		Date the sampling was completed. Format is mm/dd/yyyy.
	transfer_since_sample	varchar(2)	NO		A waste transfer to or from this tank has occurred since this sampling_event.
	appearance	varchar(255)	YES		Description of the physical appearance of the sample.
	recovery_percent	tinyint	YES		Percentage of the expected sample size that was recovered during removal from the tank.
	sampling_event_comment	varchar(255)	YES		A comment related to a sampling event.
	sample_type_comment	varchar(255)	YES		Comment concerning the sample type.
	sample_date_time	datetime	YES		Date and time the sample was collected or created. Format is mm/dd/yy hh:mm.
	physical_state	varchar(30)	NO		Liquid/Solid, the sample analysis state.
	qa_type	varchar(20)	YES		Type of quality assurance sample, if any, represented by the current sample number. Current values include: DIW BLANK, FIELD BLANK, HOT CELL BLANK, LAB BLANK, TRIP BLANK, BLIND, HYDROSTATIC HEAD FLUID, LINER LIQUID, SERIAL DILUTION, STANDARD, SYNTHETIC, and NONE.
	comp_name	varchar(255)	YES		Composite name
	description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.
	doc_loc	varchar(150)	YES		Document in which analysis result is recorded.
	lab_received_date	datetime	YES		Date sample was received at laboratory. Format is mm/dd/yy hh:mm.
	access_level	tinyint	NO		Access control mechanism for data in database table.
	privitization	varchar(4)	YES		Privitization status
	proj_id	varchar(40)	YES		Identifies the project that authorized the data to be taken
	proj_type	varchar(40)	YES		The project type of a sample.
	proj_name_short	varchar(40)	YES		Name of project that caused the sample to be analyzed
	proj_doc_name_short	varchar(40)	YES		Short name/description for a project document
	proj_sponsor	varchar(40)	YES		Sponsor of the project for which this sample was taken..

volume_in_ml	float	YES	Volume of a sample in milliliters. Usually is type float, but appears as a varchar in some tables.
weight_in_grams	float	YES	The weight in grams -- This field is type float in most tables but is type char in some.
phase_assignment	varchar(30)	YES	Description which designates the phase assigned to the sample portion being measured or analyzed.
in_current_bbi	varchar(30)	YES	Is this sampling event current in BBI
data_status	varchar(30)	YES	Data is current (no transfers since sampled), pre-transfer (previous to a transfer), or useable even if prior to a transfer

dbo.spike_recoveries_outofrange

The Populate Spike Recoveries DTS package first deletes all records, then inserts values where the result_type is like spike and the result_value is less than 75% or greater than 125%

I F P	Column name	Data type	Nulls	Default	Description
	waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	con_id	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
	method_group_id	varchar(30)	NO		Name assigned to the general group of laboratory analytical methods to which the current method belongs. Examples include: ICP:A and AA:F.
	physical_state	varchar(6)	NO		Liquid/Solid, the sample analysis state.
	agg_level	varchar(20)	NO		Label or descriptor applied to the whole, i.e. "aggregate", sample before separation into portions or fractions prior to analysis. See /dictionary/AggLevel.asp for more information.
	hits	int	YES		Count of the spike recoveries which are outside 100% +/- 25% for this analyte, method, phase, and aggregation
	tot	int	NO		Count of all spike recoveries for this analyte, method, phase, and aggregation

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_spike_recoveries_outofrange	waste_site_id	ASC		NONCLUSTERED

dbo.state

Lookup table of the 50 states plus the America Samoa, District of Columbia, Puerto Rico and the Virgin Islands.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			state	varchar(2)	NO		State code for states (or territories) of the United States
			state_name	varchar(25)	YES		The non-abbreviated name of a state such as 'Texas'.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_state	state	ASC	Yes	CLUSTERED
PK__state__66E1BA57	state	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.laboratory	FK__laborator__state__5B3AFD81	i_state

dbo.statistician_notify

Referenced by the *CheckStatisticiansChecksT2 DTS* package.

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	field_checked	varchar(20)	NO		The name of the field that the p_stat_notify stored procedure is checking.
	waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	current_contents	varchar(30)	NO		The value of the field being checked.
	comments	varchar(255)	NO		General comment by statisticians.

dbo.statistician_notify_armod

Contains data relating to the modification of analysis_result records

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	field_checked	varchar(20)	NO		The name of the field that was checked.
	waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	con_name	varchar(60)	NO		Name of the constituent used for reporting purposes. Typically, the name comes from the Chemical Abstract Services (CAS) listing.
	current_armod	varchar(30)	NO		The current analysis result modification record.
	current_ar	varchar(30)	NO		The analysis_result_id of the record that was modified.
	comments	varchar(255)	NO		General comment by statisticians.

dbo.status_flag

Lookup table containing flags indicating that the data is waiting to be loaded, or used.

I	F	P	Column name	Data type	Nulls	Default	Description
			status_flag	varchar(10)	NO		L or U.
			description	varchar(255)	YES		Waiting to Load, or Used.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_status_flag	status_flag	ASC	Yes	CLUSTERED
PK__status_flag__67D5DE90	status_flag	ASC	Yes	NONCLUSTERED

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.sample_number_master	FK__sample_nu__statu__006C8230	i_status_flag
dbo.sample_number_master	FK__sample_nu__statu__75EEF3BD	i_status_flag
dbo.sample_number_master	FK__sample_nu__statu__67A0D466	i_status_flag

Triggers

Trigger name: dbo.t_status_flag_del
Trigger type: DELETE

Created on: 27 Apr 2000
Trigger active: Yes

dbo.std_constituent_units

Lookup table of units used for specific chemical constituents

<u>I</u> <u>F</u> <u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
 	con_type	varchar(15)	NO		General type or classification: chemical, radiochemical, or physical property.
	unit_category	varchar(15)	NO		The category of the units of measurement -- bulk, concentration, or physical.
 	physical_state	varchar(30)	NO		Liquid/Solid, the sample analysis state.
 	std_units	varchar(10)	NO		Standard Units of the "standard concentration" value. The Std_Units are a subset of the Units_Of_Concentration values.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_std_constituent_units	con_type	ASC	Yes	CLUSTERED
i_std_constituent_units	unit_category	ASC	Yes	CLUSTERED
i_std_constituent_units	physical_state	ASC	Yes	CLUSTERED

Foreign keys

<u>Constraint name</u>	<u>Column name</u>	<u>Reference</u>	<u>Description</u>
FK_std_const_physi__080DA3F8	physical_state	PK__physical_state__5B7007AB (dbo.physical_state)	
FK_std_const_con_t__07197FBF	con_type	i_con_type ()	
FK_std_const_std_u__0901C831	std_units	i_units ()	

dbo.std_property_units

Lookup table of units used for specific physical constituents

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			con_id	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
			std_units	varchar(10)	NO		Standard Units of the "standard concentration" value. The Std_Units are a subset of the Units_Of_Concentration values.

Foreign keys

<u>Constraint name</u>	<u>Column name</u>	<u>Reference</u>	<u>Description</u>
FK__std_prope__std_u__0AEA10A3	std_units	i_units ()	
FK__std_prope__con_i__09F5EC6A	con_id	i_constituent ()	

dbo.synonym_group_id

Used to group the constituent_synonym records.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			synonym_group_id	varchar(40)	NO		ID for the synonym group
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_synonym_group_id	synonym_group_id	ASC	Yes	CLUSTERED
PK__synonym_group_id__68CA02C9	synonym_group_id	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.constituent_synonym	FK__constitue__synon__4DE10263	i_synonym_group_id

dbo.t_data_loaded_status

The Populate TCD T2 Job runs the Populate Data Entry Status T2 DTS package, which executes the tcd.dbo.p_destatus stored procedure. This procedure populates the t_data_loaded_status with latest dates for activities that occur for each tank.

I F P	Column name	Data type	Nulls	Default	Description
	waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	core_date	datetime	YES		Date of the latest core sampling event for this tank.
	surface_date	datetime	YES		Date of the latest surface sampling event for the tank.
	supernate_date	datetime	YES		Date of the latest supernate sampling event for the tank.
	historical_flag	varchar(3)	YES		YES or null indicating existence of historical data for the tank.
	tcr_date	datetime	YES		Date of the latest TCR report for the tank posted into the DSA system (not including preliminary reports).
	htce_date	datetime	YES		Date of the latest HTCE data for the tank.
	best_basis_date	datetime	YES		Date of the latest Best Basis Inventory data for the tank.
	as_of_date	datetime	YES		Date on which the data in this Data Loading Summary was assembled (should be previous evening)

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_t_data_loaded_status	waste_site_id	ASC	Yes	CLUSTERED

dbo.tank_core

Information concerning the tank core. Used by the TCDLoader2 ASP.NET application.

I F P	Column name	Data type	Nulls	Default	Description
	tank_core_id	int	NO		Unique record identifier.
 	sampling_event_id	varchar(20)	NO		The identification number of a sampling event collected from a waste site. For cores, this will be a core number, for supernate samples, this will be a supernate sample number, and for surface samples, this will be a surface sample number.
 	waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	serial_num	int	YES		Serial number identifying a tank core sample.
	access_level	tinyint	NO		Access control mechanism for data in database table.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_tank_core	sampling_event_id	ASC	Yes	CLUSTERED
i_tank_core	waste_site_id	ASC	Yes	CLUSTERED
PK__tank_core__69BE2702	sampling_event_id	ASC	Yes	NONCLUSTERED
PK__tank_core__69BE2702	waste_site_id	ASC	Yes	NONCLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__tank_core__7001103D	sampling_event_id	PK__sampling_event__65ED961E (dbo.sampling_event)	
FK__tank_core__7001103D	waste_site_id	PK__sampling_event__65ED961E (dbo.sampling_event)	

Triggers

Trigger name:	dbo.t_tank_core_ins_upd	Created on:	14 Jul 2006
Trigger type:	INSERT, UPDATE	Trigger active:	Yes
Trigger name:	dbo.t_tank_core_del	Created on:	14 Jul 2006
Trigger type:	DELETE	Trigger active:	Yes

dbo.tank_core_segment

Information concerning the coring segment. Used by the TCDLoader2 ASP.NET application.

I	F	P	Column name	Data type	Nulls	Default	Description
			tank_core_segment_id	int	NO		
			samp_num	varchar(12)	NO		Unique alpha-numeric ID assigned to identify a sample in the database.
			sampling_event_id	varchar(20)	NO		The identification number of a sampling event collected from a waste site. For cores, this will be a core number, for supernate samples, this will be a supernate sample number, and for surface samples, this will be a surface sample number.
			waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
			segment_id	varchar(12)	NO		Unique identifier for a segment of a core.
			serial_num	varchar(7)	YES		Serial number identifying a tank core sample.
			sample_level	int	YES		Height at which the sample was taken, measured from the bottom of the waste site.
			sample_level_units	varchar(10)	YES		Units of measure associated with the sample level.
			recovery_percent	tinyint	YES		Percentage of the expected sample size that was recovered during removal from the tank.
			appearance	varchar(255)	YES		Description of the physical appearance of the sample.
			comment	varchar(255)	YES		Comment
			access_level	tinyint	NO		Access control mechanism for data in database table.
			weight_in_grams	float	YES		The weight in grams -- This field is type float in most tables but is type char in some.
			last_modified_date_time	datetime	YES		Date on which the corresponding change record was entered into the TCD. This same date is applied to each analyte changed as part of the respective change log. Once the data has been entered into TCD, the change is immediately visible via the queries available through the TWINS2 interface.
			modified_by	varchar(25)	YES		The person who modified a record.
			modification_reason	varchar(255)	YES		The reason a record was modified.
			volume_in_ml	float	YES		Volume of a sample in milliliters. Usually is type float, but appears as a varchar in some tables.
			sample_data_file_name	varchar(255)	YES		

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i1_tank_core_segment	samp_num	ASC		CLUSTERED
i2_tank_core_segment	waste_site_id	ASC	Yes	NONCLUSTERED
i2_tank_core_segment	sampling_event_id	ASC	Yes	NONCLUSTERED
i2_tank_core_segment	segment_id	ASC	Yes	NONCLUSTERED
PK_tank_core_segment	sampling_event_id	ASC	Yes	NONCLUSTERED
PK_tank_core_segment	waste_site_id	ASC	Yes	NONCLUSTERED
PK_tank_core_segment	segment_id	ASC	Yes	NONCLUSTERED

Triggers

Trigger name: dbo.t_tank_core_segment_del Created on: 14 Jul 2006
 Trigger type: DELETE Trigger active: Yes

Trigger name: dbo.t_tank_core_segment_ins_upd Created on: 14 Jul 2006
 Trigger type: INSERT, UPDATE Trigger active: Yes

dbo.Tank_Farm

A group of tanks.

I	F	P	Column name	Data type	Nulls	Default	Description
			Tank_Farm	varchar(8)	NO		Letter code identifying a tank farm such as 'SY' or 'U'
			Area	varchar(8)	YES		200 East or West.
			Construction_Period	varchar(8)	NO		The date range that the tanks were built in this farm.
			Description1	varchar(60)	NO		Description.
			Description2	varchar(60)	YES		Description.
			Shell_Type	varchar(5)	NO	('SST')	The type of shell a waste tank has.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
PK_TF_Tank_Farm	Tank_Farm	ASC	Yes	CLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK_TF_Area	Area	PK_TFA_Tank_Farm (dbo.Tank_Farm_Area)	

dbo.Tank_Farm_Area

200 East or West areas.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			Area	varchar(8)	NO		200 East or West.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK_TFA_Tank_Farm	Area	ASC	Yes	CLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.Tank_Farm	FK_TF_Area	PK_TFA_Tank_Farm

dbo.tank_supernate_sample

Information concerning the supernate sampling event. Used by the TCDLoader2 ASP.NET application.

I F P	Column name	Data type	Nulls	Default	Description
	tank_supernate_sample_id	int	NO		Unique record identifier.
	samp_num	varchar(12)	NO		Unique alpha-numeric ID assigned to identify a sample in the database.
	sampling_event_id	varchar(20)	NO		The identification number of a sampling event collected from a waste site. For cores, this will be a core number, for supernate samples, this will be a supernate sample number, and for surface samples, this will be a surface sample number.
	waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	sample_level	int	YES		Height at which the sample was taken, measured from the bottom of the waste site.
	sample_level_units	varchar(10)	YES		Units of measure associated with the sample level.
	recovery_percent	tinyint	YES		Percentage of the expected sample size that was recovered during removal from the tank.
	appearance	varchar(255)	YES		Description of the physical appearance of the sample.
	access_level	tinyint	NO		Access control mechanism for data in database table.
	weight_in_grams	float	YES		The weight in grams -- This field is type float in most tables but is type char in some.
	last_modified_date_time	datetime	YES		Date on which the corresponding change record was entered into the TCD. This same date is applied to each analyte changed as part of the respective change log. Once the data has been entered into TCD, the change is immediately visible via the queries available through the TWINS2 interface.
	modified_by	varchar(25)	YES		The person who modified a record.
	modification_reason	varchar(255)	YES		The reason a record was modified.
	volume_in_ml	float	YES		Volume of a sample in milliliters. Usually is type float, but appears as a varchar in some tables.
	sample_data_file_name	varchar(255)	YES		Origin of the data.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i1_tank_supernate_sample	samp_num	ASC	Yes	CLUSTERED
i1_tank_supernate_sample	sampling_event_id	ASC	Yes	CLUSTERED
i1_tank_supernate_sample	waste_site_id	ASC	Yes	CLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__tank_supernate_s__6FOCE C04	sampling_event_id	PK__sampling_event__65ED961E (dbo.sampling_event)	
FK__tank_supernate_s__6FOCE C04	waste_site_id	PK__sampling_event__65ED961E (dbo.sampling_event)	
FK__tank_supe__sampl__OCD2 5915	sample_level_units	i_units ()	

Triggers

Trigger name:	dbo.t_tank_supernate_sample_del	Created on:	14 Jul 2006
Trigger type:	DELETE	Trigger active:	Yes
Trigger name:	dbo.t_tank_supernate_samp_ins_upd	Created on:	14 Jul 2006
Trigger type:	INSERT, UPDATE	Trigger active:	Yes

dbo.tank_surface_sample

Information concerning the surface sampling event. Used by the TCDLoader2 ASP.NET application.

I F P	Column name	Data type	Nulls	Default	Description
	tank_surface_sample_id	int	NO		Unique record identifier.
	samp_num	varchar(12)	NO		Unique alpha-numeric ID assigned to identify a sample in the database.
	sampling_event_id	varchar(20)	NO		The identification number of a sampling event collected from a waste site. For cores, this will be a core number, for supernate samples, this will be a supernate sample number, and for surface samples, this will be a surface sample number.
	waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	sample_level	int	YES		Height at which the sample was taken, measured from the bottom of the waste site.
	sample_level_units	varchar(10)	YES		Units of measure associated with the sample level.
	recovery_percent	tinyint	YES		Percentage of the expected sample size that was recovered during removal from the tank.
	appearance	varchar(255)	YES		Description of the physical appearance of the sample.
	access_level	tinyint	NO		Access control mechanism for data in database table.
	weight_in_grams	float	YES		The weight in grams -- This field is type float in most tables but is type char in some.
	last_modified_date_time	datetime	YES		Date on which the corresponding change record was entered into the TCD. This same date is applied to each analyte changed as part of the respective change log. Once the data has been entered into TCD, the change is immediately visible via the queries available through the TWINS2 interface.
	modified_by	varchar(25)	YES		The person who modified a record.
	modification_reason	varchar(255)	YES		The reason a record was modified.
	volume_in_ml	float	YES		Volume of a sample in milliliters. Usually is type float, but appears as a varchar in some tables.
	sample_data_file_name	varchar(255)	YES		Origin of the data.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i1_tank_surface_sample	samp_num	ASC	Yes	CLUSTERED
i1_tank_surface_sample	sampling_event_id	ASC	Yes	CLUSTERED
i1_tank_surface_sample	waste_site_id	ASC	Yes	CLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__tank_surf__sampl__6E18C7CB	sample_level_units	i_units ()	
FK__tank_surface_sam__6D24A392	sampling_event_id	PK__sampling_event__65ED961E (dbo.sampling_event)	
FK__tank_surface_sam__6D24A392	waste_site_id	PK__sampling_event__65ED961E (dbo.sampling_event)	

Triggers

Trigger name:	dbo.t_tank_surface_sample_ins_upd	Created on:	14 Jul 2006
Trigger type:	INSERT, UPDATE	Trigger active:	Yes
Trigger name:	dbo.t_tank_surface_sample_del	Created on:	14 Jul 2006
Trigger type:	DELETE	Trigger active:	Yes

dbo.tank_type

Lookup of tank types

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			tank_type	varchar(5)	NO		Type of Tank
			shell_type	varchar(10)	NO		The type of shell a waste tank has.
			description	varchar(255)	NO		Brief narrative describing a specific instance in this database table.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK__tank_type__204F41DD	tank_type	ASC	Yes	CLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.waste_tank	FK__waste_tan__tank___1EF10950	PK__tank_type__204F41DD

dbo.tank_use

Lookup table of the possible uses for a double shell tank.

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	tank_use	varchar(60)	NO		Tank use for double-shell tanks only
	description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.
	abbrev	varchar(20)	YES		An abbreviation for another field in the table. The exact meaning depends on the table in which it is used.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_tank_use	abbrev	ASC	Yes	NONCLUSTERED

dbo.tbl_SACS_waste_Site_id

Maps the tank names in the SACS database to the tank names used throughout TWINS.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			tank	varchar(10)	YES		The unique identifier of the tank in the SACS database.
			waste_site_id	varchar(12)	YES		Tank Identifier, using the format of 241-Tank Farm-Tank #.
			sensor_type_id	varchar(2)	YES		Always equal to AE.
			kgal_per_inch	float	YES		Not used, equal to NULL.
			gauge_offset	float	YES	(0.0)	Not used, equal to NULL.
			user_offset	float	YES	(0.0)	Not used, equal to NULL.
			total_offset	float	YES	(0.0)	Not used, equal to NULL.

dbo.tblLimits

Contains the parameters and limits used in the RESOLVE ASP.NET application and the p_check_rules stored procedure.

I F P	Column name	Data type	Nulls	Default	Description
	con_id	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
	physical_state	varchar(30)	NO		Liquid/Solid, the sample analysis state.
	std_value_units	varchar(10)	NO		Common units of measure: kilograms (kg) for chemicals and curies (Ci) for radionuclides.
	test_type	varchar(20)	NO		Either greater than or less than test.
	float_value	float	YES		A value expressed as a float number.
	string_value	varchar(20)	YES		Not used, always equal to NULL.
	review_status	varchar(30)	YES		Review status. Current values are: REVIEWED, REVIEWED: SEE NOTE, CHECKING NOT APPLIED, DATA SUSPECT: SEE COMMENT, REVIEWED: NOT FOR INVENTORY and DATA RESOLUTION IN PROGRESS. See /dictionary/ReviewStatusCaveats.asp for more information.
	review_comment	varchar(255)	YES		Comment added by reviewer of this analysis result record.
	date_active	datetime	NO		Date that the limit goes into effect.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
_WA_Sys_con_id_3BF5C5A4	con_id	ASC		NONCLUSTERED
_WA_Sys_float_value_58920452	float_value	ASC		NONCLUSTERED
_WA_Sys_physical_state_58920452	physical_state	ASC		NONCLUSTERED
_WA_Sys_review_comment_58920452	review_comment	ASC		NONCLUSTERED
_WA_Sys_std_value_units_58920452	std_value_units	ASC		NONCLUSTERED
_WA_Sys_test_type_58920452	test_type	ASC		NONCLUSTERED

dbo.tblNew_Mean_Vector_Header

Used in the AutoVector ASP.NET application. [View the ERD](ERD/autovector.jpg) and notice the joined fields.

IFP	Column name	Data type	Nulls	Default	Description
	AV_ID	int	NO		Unique Identifier for the tbl_New_Mean_Vector_Header
	Means_Data_Name	nvarchar(50)	NO		
	Means_Waste_Site_ID	nvarchar(15)	YES		
	Means_Field_Sample_Name	nvarchar(255)	YES		
	Means_Phase	nvarchar(50)	YES		
	Means_Date	datetime	YES		
	Vector_Type	nvarchar(18)	NO	('Sample')	In the case of tank inventories based on vectors, Vector_Type = Vector.Vector_Type. In the case of tank inventories based on other tank inventories, Vector_Type = "Tank Inventory". In the case of tank inventories based on vector averages, Vector_Type = "Sample Average".
	Template	nvarchar(15)	YES	('NA')	A group of vectors is often used in inventory calculations for many tanks. A group is referred to as a Template. The Template field contains the name of the template.
	Vector_Name	nvarchar(40)	YES		In the case of tank inventories based on vectors, Vector_Name = Vector.Vector_Name. In the case of tank inventories based on other tank inventories, Vector_Name = the Waste_Site_ID + the Waste_Type of the tank that the inventory is obtained from. In the case of tank inventories based on vector averages, Vector_Name = "(Calculated)".
	Density	float	YES	(1)	The density, in g/ml of the vector. A density is needed to calculate inventories when vector constituents are recorded as concentrations. If vector constituents values are recorded in units of inventory (kg or Ci), then Density is not needed.
	Density_Comment	nvarchar(80)	YES		Information about where the density came from, and or how it was derived.
	Sample_Date	datetime	YES		For vectors with a type of "Sample", Sample_Date contains the date that the sample was taken. Sometimes this data is not readily available.
	Analysis_Date	datetime	YES		Date of sample analysis. Format is mm/dd/yy.
	Reference_Document	nvarchar(80)	YES		
	Reference_Table_Name	nvarchar(150)	YES		
	Reference_Table_Number	nvarchar(15)	YES	('NA')	
	Reference_Table_Column	nvarchar(15)	YES	('3')	
	Comp_Waste_Site_ID	nvarchar(15)	YES		
	Comp_Waste_Phase	nvarchar(50)	YES		
	Comp_Waste_Type	nvarchar(50)	YES		
	Comp_Inventory_Date	datetime	YES		
	Comp_Volume	float	YES	(0)	
	Comp_Volume_Units	nvarchar(50)	YES		
	Comp_Volume_Reference	nvarchar(80)	YES		
	Rep_Include_In_Average	bit	NO	(0)	
	Rep_Multiplier	float	YES	(1)	
	Rep_Multiplier_Description	nvarchar(240)	YES		
	Is_New	bit	NO	(1)	
	author	nvarchar(50)	YES		
	Tank_Coordinator	nvarchar(50)	YES		
	Is_Saved	bit	NO	(0)	
	New_Vector_Status	nvarchar(20)	YES	('New')	The status of this particular vector.
	Create_Date	datetime	YES	(getdate())	Creation date for the HTWOS Overview record
	AutoStatHeader_id	int	YES		Unique ID for AutoStatHeader
	V_ID	int	YES		Computer generated unique identifier for the vector.

dbo.tblNew_Vector_Choices

Referenced by the AutoStat ASP.NET application. [View the ERD](ERD/autovector.jpg) and notice the joined fields.

I F P	Column name	Data type	Nulls	Default	Description
	Means_Data_Name	varchar(50)	YES		
	Con_ID	varchar(15)	YES		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
	Method_Group_Name	varchar(30)	YES		
	Con_Name	varchar(50)	YES		Name of the constituent used for reporting purposes. Typically, the name comes from the Chemical Abstract Services (CAS) listing.
	Choice_Select	bit	NO	(0)	
	Mean	float	YES	(0)	Mean concentration for sample data for the tank.
	Mean_Units	varchar(10)	YES		
	Less_Than	varchar(1)	YES	(")	
	User_Comment	varchar(100)	YES		Information written by one or more users. The triggers should not update this field.
	Concentration_RSD	float	YES		Relative Standard Deviation for Concentration, in units of percentage. Equal to Std_Dev / Concentration.
	AV_ID	int	YES		Unique Identifier for the tbl_New_Mean_Vector_Header

Indexes

Index name	Column name	Sort direction	Is unique	Index type
PK_tblnew_vector_choices	Means_Data_Name	ASC	Yes	CLUSTERED
PK_tblnew_vector_choices	Con_ID	ASC	Yes	CLUSTERED
PK_tblnew_vector_choices	Method_Group_Name	ASC	Yes	CLUSTERED
PK_tblnew_vector_choices	Con_Name	ASC	Yes	CLUSTERED

Triggers

Trigger name: dbo.tblNew_Vector_Choices_Itrig
Trigger type: INSERT

Created on: 19 Dec 2002
Trigger active: Yes

dbo.tblNew_Vector_Choices1

Used by the ASP.NET AutoVector application.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			NV_ID	int	NO		
			AV_ID	int	NO		Unique Identifier for the tbl_New_Mean_Vector_Header
			Con_ID	varchar(15)	YES		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
			Method_Group_Name	varchar(30)	YES		
			Con_Name	varchar(50)	YES		Name of the constituent used for reporting purposes. Typically, the name comes from the Chemical Abstract Services (CAS) listing.
			Choice_Select	bit	NO	(0)	
			Mean	float	YES	(0)	Mean concentration for sample data for the tank.
			Mean_Units	varchar(10)	YES		
			Less_Than	varchar(1)	YES	(")	
			User_Comment	varchar(100)	YES		Information written by one or more users. The triggers should not update this field.
			Concentration_RSD	float	YES		Relative Standard Deviation for Concentration, in units of percentage. Equal to Std_Dev / Concentration.
			Is_Selected	bit	NO	(1)	
			AutoStatResults_id	int	YES		Unique ID for AutoStatResult

dbo.TblNew_Vector_Counts

Used in the Vector2000_Prod.mdb application. [View the ERD](#) and notice the joined fields.

I F P	Column name	Data type	Nulls	Default	Description
	Means_Data_Name	nvarchar(50)	NO		
	Con_ID	nvarchar(50)	YES		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
	Con_Name	nvarchar(50)	YES		Name of the constituent used for reporting purposes. Typically, the name comes from the Chemical Abstract Services (CAS) listing.
	Con_Type	nvarchar(50)	YES		General type or classification: chemical, radiochemical, or physical property.
	My_Count	int	YES		
	Is_Selected	bit	NO		

Indexes

Index name	Column name	Sort direction	Is unique	Index type
pk_tblnew_vector_counts	Means_Data_Name	ASC	Yes	CLUSTERED
pk_tblnew_vector_counts	Con_ID	ASC	Yes	CLUSTERED

dbo.TblNew_Vector_Status

Referenced by the AutoStat ASP.NET application. Used in the Vector2000_Prod.mdb application

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			New_Vector_Status	nvarchar(20)	NO		The status of this particular vector.
			New_Vector_Status_Show	nvarchar(20)	NO		The status as shown to the user.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
pk_tblnew_vector_status	New_Vector_Status	ASC	Yes	CLUSTERED

dbo.tblResolve_Tanks

A list of tanks that need to have their review status resolved. Set by the `p_Resolve_Tanks` stored procedure.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK_tblResolve_Tanks	waste_site_id	ASC	Yes	NONCLUSTERED

dbo.tblResolve_Users

Used by the *Resolve ASP* application.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			Login_ID	varchar(50)	NO		The identification of a person accessing the system or database.
			Best_Role	varchar(50)	YES		A role in the SQL database in which a user is a member.
			User_Name	varchar(20)	YES		Given name of the user

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK_tblResolve_Users	Login_ID	ASC	Yes	NONCLUSTERED

dbo.TblSendMailTo

Used in the Vector2000_Prod.mdb application

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			MailName	nvarchar(50)	YES		Email address.
			SendYN	bit	NO		

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
pk_tblsendmailto	MailName	ASC	Yes	CLUSTERED

dbo.transaction_type

Lookup list of valid codes representing gains or losses of waste from a tank.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			transaction_type	varchar(2)	NO		Two character code for the type of transfer transaction - Loss, Gain, evaporation and etc..
			description	varchar(255)	NO		Brief narrative describing a specific instance in this database table.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_transaction_type	transaction_type	ASC	Yes	CLUSTERED
PK__transaction_type__0B1F1ACD	transaction_type	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.transfers	FK__transfers__trans__14737ADD	i_transaction_type

dbo.transfers

Tank waste transfers from January 1985 to September 2001.

I F P	Column name	Data type	Nulls	Default	Description									
FK	waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.									
FK	transaction_type	varchar(2)	NO		Two character code for the type of transfer transaction - Loss, Gain, evaporation and etc..									
FK	waste_source	varchar(15)	NO		The source of the waste that was transferred. If the transaction is an evaporation, the source will be a code like DN500, which means that an evaporation takes place on the waste of the DN waste type and evaporates it down to 50% of its current volume, and the concentrated waste is transferred to the tank listed in the destination column. The code given for evaporations is determined by the two letters of the final waste type and the percent the volume is to be evaporated. An important number to be used in this position is 100. The code DN500 means a 50% evaporation, but the exception to this convention occurs if DN100 is used. The 100 means no evaporation, rather than the 10% evaporation that would otherwise be assumed. This exception allows the evaporation entry to be used to change waste types in a tank.									
FK	waste_destination	varchar(15)	NO		The destination of the waste that was transferred.									
FK	transfer_begin_date	datetime	NO		The date the transfer began.									
FK	transfer_end_date	datetime	NO		The date the transfer ended									
FK	waste_type_start_volume	float	NO		The original volume of the given waste type in the tank before the transfer.									
FK	start_volume_units	varchar(10)	NO		The units used to measure the waste type start volume.									
FK	waste_type_end_volume	float	NO		The volume of the given waste type in the tank after the transfer.									
FK	end_volume_units	varchar(10)	NO		The units used to measure the waste type end volume.									
FK	waste_type	varchar(2)	NO		Abbreviation for the name of the waste. Some Examples: <table border="0"> <tr> <td>AW</td> <td>CC</td> <td>CP</td> </tr> <tr> <td>DC</td> <td>DN</td> <td>IL</td> </tr> <tr> <td>SLUDGE</td> <td>SOLID_CAKE</td> <td>SUPER</td> </tr> </table>	AW	CC	CP	DC	DN	IL	SLUDGE	SOLID_CAKE	SUPER
AW	CC	CP												
DC	DN	IL												
SLUDGE	SOLID_CAKE	SUPER												
FK	transfer_volume	float	NO		The volume of the waste that was transferred.									
FK	transfer_volume_units	varchar(10)	NO		The units used to measure the transfer volume.									
FK	tank_volume	float	NO		The volume of waste in the tank after the transfer.									
FK	tank_volume_units	varchar(10)	NO		The units used to measure the tank volume.									
	load_date_time	datetime	NO		The date and time that the record was loaded into the TCD.									
	last_modified_date_time	datetime	YES		Date on which the corresponding change record was entered into the TCD. This same date is applied to each analyte changed as part of the respective change log. Once the data has been entered into TCD, the change is immediately visible via the queries available through the TWINS2 interface.									
	access_level	tinyint	NO		Access control mechanism for data in database table.									

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_transfers	waste_site_id	ASC	Yes	CLUSTERED
i_transfers	transaction_type	ASC	Yes	CLUSTERED
i_transfers	waste_source	ASC	Yes	CLUSTERED
i_transfers	waste_destination	ASC	Yes	CLUSTERED
i_transfers	transfer_begin_date	ASC	Yes	CLUSTERED
i_transfers	transfer_end_date	ASC	Yes	CLUSTERED
i_transfers	waste_type_start_volume	ASC	Yes	CLUSTERED

i_transfers	waste_type_end_volume	ASC	Yes	CLUSTERED
i_transfers	waste_type	ASC	Yes	CLUSTERED
i_transfers	transfer_volume	ASC	Yes	CLUSTERED
i_transfers	tank_volume	ASC	Yes	CLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__transfers__trans__137F56A4	transfer_volume_units	i_units ()	
FK__transfers__tank__128B326B	tank_volume_units	i_units ()	
FK__transfers__waste__165BC34F	waste_type	i_waste_type ()	
FK__transfers__waste__15679F16	waste_site_id	i_waste_tank ()	
FK__transfers__start__11970E32	start_volume_units	i_units ()	
FK__transfers__end_v__10A2E9F9	end_volume_units	i_units ()	
FK__transfers__trans__14737ADD	transaction_type	i_transaction_type ()	

Triggers

Trigger name:	dbo.t_transfers_ins_upd	Created on:	22 Mar 2001
Trigger type:	INSERT, UPDATE	Trigger active:	Yes

dbo.transfers_denorm

TWINS Query: Contains data about tank gains, transfers, losses, or evaporations beginning in January 1985.

I F P	Column name	Data type	Nulls	Default	Description
	waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	transaction_type	varchar(2)	NO		Two character code for the type of transfer transaction - Loss, Gain, evaporation and etc..
	transaction_type_description	varchar(255)	YES		The type of trasaction (Loss, Gain, Transfer, or Evaporation).
	waste_source	varchar(15)	NO		The source of the waste that was transferred. If the transaction is an evaporation, the source will be a code like DN500, which means that an evaporation takes place on the waste of the DN waste type and evaporates it down to 50% of its current volume, and the concentrated waste is transferred to the tank listed in the destination column. The code given for evaporations is determined by the two letters of the final waste type and the percent the volume is to be evaporated. An important number to be used in this position is 100. The code DN500 means a 50% evaporation, but the exception to this convention occurs if DN100 is used. The 100 means no evaporation, rather than the 10% evaporation that would otherwise be assumed. This exception allows the evaporation entry to be used to change waste types in a tank.
	waste_source_description	varchar(255)	YES		The description of the waste source.
	waste_destination	varchar(15)	NO		The destination of the waste that was transferred.
	waste_destination_description	varchar(255)	YES		The description of the waste destination.
	transfer_begin_date	datetime	NO		The date the transfer began.
	transfer_end_date	datetime	NO		The date the transfer ended
	waste_type_start_volume	float	NO		The original volume of the given waste type in the tank before the transfer.
	start_volume_units	varchar(10)	NO		The units used to measure the waste type start volume.
	waste_type_end_volume	float	NO		The volume of the given waste type in the tank after the transfer.
	end_volume_units	varchar(10)	NO		The units used to measure the waste type end volume.
	waste_type	varchar(2)	NO		Abbeviation for the name of the waste. Some Examples:<table border=0><tr><td>AW</td><td>CC</td><td>CP</td></tr><tr><td>DC</td><td>DN</td><td>IL</td></tr><tr><td>SLUDGE</td><td>SOLID_CAKE</td><td>SUPER</td></tr></table>
	waste_type_description	varchar(255)	YES		The description of the type of waste that was transferred.
	transfer_volume	float	NO		The volume of the waste that was transferred.
	transfer_volume_units	varchar(10)	NO		The units used to measure the transfer volume.
	tank_volume	float	NO		The volume of waste in the tank after the transfer.
	tank_volume_units	varchar(10)	NO		The units used to measure the tank volume.
	load_date_time	datetime	NO		The date and time that the record was loaded into the TCD.
	last_modified_date_time	datetime	YES		Date on which the corresponding change record was entered into the TCD. This same date is applied to each analyte changed as part of the respective change log. Once the data has been entered into TCD, the change is immediately visible via the queries available through the TWINS2 interface.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_transfers_denorm	waste_site_id	ASC	Yes	CLUSTERED
i_transfers_denorm	transaction_type	ASC	Yes	CLUSTERED
i_transfers_denorm	waste_source	ASC	Yes	CLUSTERED
i_transfers_denorm	waste_destination	ASC	Yes	CLUSTERED

i_transfers_denorm	transfer_begin_date	ASC	Yes	CLUSTERED
i_transfers_denorm	transfer_end_date	ASC	Yes	CLUSTERED
i_transfers_denorm	waste_type_start_volume	ASC	Yes	CLUSTERED
i_transfers_denorm	waste_type_end_volume	ASC	Yes	CLUSTERED
i_transfers_denorm	waste_type	ASC	Yes	CLUSTERED
i_transfers_denorm	transfer_volume	ASC	Yes	CLUSTERED
i_transfers_denorm	tank_volume	ASC	Yes	CLUSTERED

dbo.unit_conversion

Table for cross-referencing conversion of units from 1 measurement to a standard unit of measure

I F P	Column name	Data type	Nulls	Default	Description
	unit_conversion_id	int	NO		Unique record identifier.
	convert_units_from	varchar(10)	NO		The units of measurement to which a conversion formula should be applied.
	convert_units_to	varchar(10)	NO		The units of measurement to which a formula should convert.
	operator_order	tinyint	NO		Used to indicate the order in which mathematical operations should be done.
	operator	varchar(10)	NO		Limiting operator for the DQO criteria. The values are >, <, or =.
	factor	float	NO		A number used in the multiplication of the inventory of Con_ID2 to obtain the inventory for Con_ID1.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_unit_conversion	convert_units_from	ASC	Yes	CLUSTERED
i_unit_conversion	convert_units_to	ASC	Yes	CLUSTERED
i_unit_conversion	operator_order	ASC	Yes	CLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__unit_conv__conve__174FE788	convert_units_from	i_units ()	
FK__unit_conv__opera__19382FFA	operator	i_operator ()	
FK__unit_conv__conve__18440BC1	convert_units_to	i_units ()	

Triggers

Trigger name:	dbo.t_unit_conversion_ins_upd	Created on:	13 Jul 2006
Trigger type:	INSERT, UPDATE	Trigger active:	Yes

dbo.units

Lookup table for units. Used by the TCDLoader2 ASP.NET application.

I	F	P	Column name	Data type	Nulls	Default	Description
			units	varchar(10)	NO		Concentration units of the reported value or the reporting limit.
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_units	units	ASC	Yes	CLUSTERED
PK__units__0C133F06	units	ASC	Yes	NONCLUSTERED

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.unit_conversion	FK__unit_conv__conve__174FE788	i_units
dbo.transfers	FK__transfers__trans__137F56A4	i_units
dbo.samp_attr	FK__samp_attr__samp__62DC1F49	i_units
dbo.waste_tank	FK__waste_tan__capac__1B20786C	i_units
dbo.analysis_result	fk_uncertainty_value_units	i_units
dbo.sample_relation	FK__sample_re__paren__76E317F6	i_units
dbo.transfers	FK__transfers__tank__128B326B	i_units
dbo.std_property_units	FK__std_prope__std_u__0AEA10A3	i_units
dbo.waste_tank	FK__waste_tan__max_o__1DFCE517	i_units
dbo.analysis_result	fk_rpt_value_units	i_units
dbo.tank_surface_sample	FK__tank_surf__sampl__6E18C7CB	i_units
dbo.means_and_variances_detail	FK__means_and__units__5D2345F3	i_units
dbo.waste_tank	FK__waste_tan__eleva__1D08C0DE	i_units
dbo.static_analysis_result_units	FK__static_an__std_u__06255B86	i_units
dbo.transfers	FK__transfers__start__11970E32	i_units
dbo.constituent	FK__constitue__half__463FE09B	i_units
dbo.static_analysis_result_units	FK__static_an__std_u__7C9BF14C	i_units
dbo.transfers	FK__transfers__end_v__10A2E9F9	i_units
dbo.waste_tank	FK__waste_tan__diame__1C149CA5	i_units
dbo.tank_supernate_sample	FK__tank_supe__sampl__0CD25915	i_units
dbo.std_constituent_units	FK__std_const__std_u__0901C831	i_units
dbo.unit_conversion	FK__unit_conv__conve__18440BC1	i_units

dbo.validation_status

Lookup table for validation status

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			validation_status	varchar(30)	NO		Validation status (unvalidated or validated) of an analysis result record.
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_validation_status	validation_status	ASC	Yes	CLUSTERED
PK__validation_statu__0D07633F	validation_status	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.analysis_result	fk_validation_status	i_validation_status

dbo.ventilation

Lookup table for ventilation

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			ventilation	varchar(30)	NO		Type of ventilation a waste tank has.
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_ventilation	ventilation	ASC	Yes	CLUSTERED
PK__ventilation__0DFB8778	ventilation	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.waste_tank	FK__waste_tan__venti__1FE52D89	i_ventilation

dbo.verification_process

Lookup table for verification process

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			verification_process	varchar(30)	NO		Identification assigned to the verification_process.
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_verification_process	verification_process	ASC	Yes	CLUSTERED
PK__verification_pro__0EEFABB1	verification_process	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.analysis_result	fk_verification_process	i_verification_process

dbo.waste_tank

Characteristics of the Hanford underground waste tanks. Note that the Tank table in the BBIM database also contains a similar set of these characteristics. However, the bbim..Tank table contains information on only the 177 HLW tanks and a few DCRT tanks while the tcd..Waste_Tank table contains information on all Hanford tanks.

I F P	Column name	Data type	Nulls	Default	Description
	waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	tank_type	varchar(5)	YES		Type of Tank
	capacity	float	YES		The amount of material a waste tank can hold.
	capacity_units	varchar(10)	YES		The units in which the capacity of a waste tank was reported.
	diameter	float	YES		Diameter of a waste tank.
	diameter_units	varchar(10)	YES		The units the diameter of waste tank is reported in.
	elevation	float	YES		The height above sea level of a waste tank.
	elevation_units	varchar(10)	YES		The units in which elevation is reported.
	construction_date	varchar(30)	YES		The date a waste tank was constructed.
	comment	varchar(255)	YES		Comment
	access_level	tinyint	NO		Access control mechanism for data in database table.
	max_operating_depth	float	YES		Maximum operating depth of a waste tank.
	max_operating_depth_units	varchar(10)	YES		Units in which maximum depth of a waste tank is reported.
	bottom_shape	varchar(20)	YES		The type of bottom shape that a waste tank has.
	ventilation	varchar(30)	YES		Type of ventilation a waste tank has.
	in_service_date	varchar(30)	YES		Date tank began receiving waste.
	recent_analysis_result_change	char(1)	YES		A flag indicating that there has been a recent change to analysis result data for a waste tank.
	recent_inventory_change	char(1)	YES		A flag indicating a recent change to tank inventory.
	recent_all_hanlon_change	char(1)	YES		A flag indicating there has been a recent change to Hanlon data for a tank. Flag indicating that the AutoTCR stdrpts_executive_summary table has been updated (set to null after the update).
	declared_inactive	varchar(20)	YES		The date a waste tank was declared inactive.
	interim_stabilization	varchar(20)	YES		A tank that contains less than 50 Kgallons of drainable interstitial liquid and less than 5 Kgallons of supernatant. If the tank was jet pumped to achieve interim stabilization, then the jet pump flow or saltwell screen inflow must also have been at or
	intrusion_prevention	varchar(20)	YES		The designation reflecting the completion of physical efforts required to minimize the addition of liquids into an inactive storage tank, process vault, sump, catch tank or diversion box.
	watch_list_status	varchar(30)	YES		The watch list status of a waste tank.
	transfer_lock_out_date	datetime	YES		Date that transfers to this waste tank were ended.
	used_in_sampling_event	char(1)	YES		This waste tank does or does not appear in a sampling event.
	last_modified_date_time	datetime	YES		Date on which the corresponding change record was entered into the TCD. This same date is applied to each analyte changed as part of the respective change log. Once the data has been entered into TCD, the change is immediately visible via the queries available through the TWINS2 interface.
	modified_by	varchar(25)	YES		The person who modified a record.
	flammable_gas_facility_group	varchar(100)	YES		Flammable gas facility group for this tank
	modification_reason	varchar(255)	YES		The reason a record was modified.
	surface_level_type	varchar(20)	YES		Surface level type for the tank
	operation_capacity_kgal	float	YES		How much waste the tank can hold for operational purposes. This may vary from tank to tank even when the tank types are the same because of corrosion in the tank wall.
	tank_status	varchar(20)	YES		Abbreviation of the status of the stabilization of the waste within the tank. Relevant only to single shell tanks (SST). Waste can be Partially Interim Stabilized (PS), Interim Stabilized (IS) and Controlled Clean and Stable (CCS).

DIL_gal	int	YES	The amount of Drainable Interstitial Liquid remaining in the tank. Measured in gallons.
PLR_gal	int	YES	The amount of Pumpable Liquid remaining in the tank. Measured in gallons.
show_DIL_PLR_SU	int	YES	Flag that indicates whether or not to display the values for DIL_gal, PLR_gal and solids_volume_update in the WTSR (Hanlon) report.
unpumpable_volume_kgal	float	YES	Waste that cannot be pumped out that resides in the bottom 18 inches of the tank. Volume varies depending upon the tank diameter, whether it has a dish bottom or not, and the shape of the dish bottom.
interstitial_liquid_level_inches	float	YES	The measure in inches above the bottom of the tank of the Interstitial Liquid Level (ILL) in a tank. If this level is not null, then use it to calculate the Drainable Interstitial Liquid Level (DIL_gal) for the WTSR (Hanlon) report. Note that there is NO liquids above this value in the saltcake or sludge. CURRENTLY NOT USED DUE TO INCOMPLETE SPECIFICATIONS FOR THE DIL CALCULATIONS. FIELD IS KEPT FOR POSSIBLE FUTURE USE.
tank_dish_height_inches	float	YES	The height in inches of the dish bottom for this tank. If the tank has a flat bottom, then this value equals zero. Used in calculating the Drainable Interstitial Liquid in the WTSR (Hanlon) report. CURRENTLY NOT USED DUE TO INCOMPLETE SPECIFICATIONS FOR THE DIL CALCULATIONS. FIELD IS KEPT FOR POSSIBLE FUTURE USE.
dish_volume_kgal	float	YES	Volume, in kgal of the dish bottom portion of the tank. If the tank bottom is flat, then dish_volume_kgal is equal to 0. Used in the calculation of the DIL in the WTSR (Hanlon) report. CURRENTLY NOT USED DUE TO INCOMPLETE SPECIFICATIONS FOR THE DIL CALCULATIONS. FIELD IS KEPT FOR POSSIBLE FUTURE USE.
tank_use_abbrev	varchar(20)	YES	Used in the WTSR (Hanlon) report for double shelled tanks. See /DevTools/DevFieldHelp.asp?dt_id=766 >tc.d..tank_use
solids_volume_updated_on	datetime	YES	Date that the volume of solids within the tank was last updated.
pumped_1980_2001_kgal_to_DST	float	YES	Amount of waste pumped in kgal from Jan 1, 1980 to Jan 1, 2001 into any double shelled tank. The TXFR application begins recording data starting on Jan 1, 2001, so this field is needed to calculate waste pumped from Jan 1, 1980 to the present (required for the Hanlon Report).
tank_integrity	varchar(20)	YES	Whether or not the tank is leaking.
saltcake_porosity	float	YES	Porosity of the Salt Cake in a tank. Used to calculate the DIL in the WTSR (Hanlon) report. CURRENTLY NOT USED DUE TO INCOMPLETE SPECIFICATIONS FOR THE DIL CALCULATIONS. FIELD IS KEPT FOR POSSIBLE FUTURE USE.
sludge_porosity	float	YES	Porosity of the Sludge in a tank. Used to calculate the DIL in the WTSR (Hanlon) report. CURRENTLY NOT USED DUE TO INCOMPLETE SPECIFICATIONS FOR THE DIL CALCULATIONS. FIELD IS KEPT FOR POSSIBLE FUTURE USE.
WTSR_data_modified_on	datetime	YES	The last date that any WTSR (Hanlon) data has been modified for this record.
notes_for_description_of_tank_report	varchar(1000)	YES	Notes added by TC for use in the AutoTCR Description of Tank report
DLR_gal	float	YES	
stabilization_method	varchar(6)	YES	
stabilization_method_footnote	varchar(20)	YES	

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_waste_tank	waste_site_id	ASC	Yes	CLUSTERED
PK__waste_tank__10D7F423	waste_site_id	ASC	Yes	NONCLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__waste_tan__tank___1EF10950	tank_type	PK__tank_type__204F41DD (dbo.tank_type)	
FK__waste_tan__capac__1B20786C	capacity_units	i_units ()	
FK__waste_tan__botto__1A2C5433	bottom_shape	i_bottom_shape ()	
FK__waste_tan__max_o__1DFCE517	max_operating_depth _units	i_units ()	
FK__waste_tan__eleva__1D08C0DE	elevation_units	i_units ()	
FK__waste_tan__watch__20D951C2	watch_list_status	i_watch_list_status ()	
FK__waste_tan__diame__1C149CA5	diameter_units	i_units ()	
FK__waste_tan__venti__1FE52D89	ventilation	i_ventilation ()	

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.means_and_variances_header	FK__means_and__waste__5F0B8E65	i_waste_tank
dbo.waste_tank_group_member	FK__waste_tan__waste__21CD75FB	i_waste_tank
dbo.sampling_event	FK__sampling__waste__0348EEDB	i_waste_tank
dbo.sampling_event	FK__sampling__waste__79BF84A1	i_waste_tank
dbo.transfers	FK__transfers__waste__15679F16	i_waste_tank
dbo.ESP_metadata	FK__ESP_metad__waste__14A88507	i_waste_tank
dbo.ESP_data	FK__ESP_data__waste__13B460CE	i_waste_tank

Triggers

Trigger name:	dbo.t_waste_tank_ins_upd	Created on:	19 May 2003
Trigger type:	INSERT, UPDATE	Trigger active:	Yes
Trigger name:	dbo.t_waste_tank_del	Created on:	19 May 2003
Trigger type:	DELETE	Trigger active:	Yes

dbo.waste_tank_group

IMUST or Underground Storage tank groups.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			waste_tank_group_id	varchar(30)	NO		The type of storage tank such as IMUST or Underground Storage Tank.
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_waste_tank_group	waste_tank_group_id	ASC	Yes	NONCLUSTERED
PK__waste_tank_group__26081B33	waste_tank_group_id	ASC	Yes	CLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.waste_tank_group_member	FK__waste_tan__waste__22C19A34	PK__waste_tank_group__26081B33

dbo.waste_tank_group_member

Groups tanks into one of the two valid tank groups (IMUST or Underground Storage).

I	F	P	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			waste_tank_group_id	varchar(30)	NO		The type of storage tank such as IMUST or Underground Storage Tank.
			waste_site_id	varchar(30)	YES		Tank Identifier, using the format of 241-Tank Farm-Tank #.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_waste_tank_group_member	waste_tank_group_id	ASC	Yes	NONCLUSTERED
i_waste_tank_group_member	waste_site_id	ASC	Yes	NONCLUSTERED

Foreign keys

<u>Constraint name</u>	<u>Column name</u>	<u>Reference</u>	<u>Description</u>
FK__waste_tan__waste__22C19A34	waste_tank_group_id	PK__waste_tank_group__26081B33 (dbo.waste_tank_group)	
FK__waste_tan__waste__21CD75FB	waste_site_id	i_waste_tank ()	

dbo.WasteDescConstituents

Used by the TWINS Waste Description query. Contains the constituents that are reported by this query. Query needed to assist the Waste Management Group designate waste according to WAC 173-303-9904 Dangerous waste sources list (Change Request 290).

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	con_id	varchar(20)	YES		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
	con_name	varchar(100)	YES		Name of the constituent used for reporting purposes. Typically, the name comes from the Chemical Abstract Services (CAS) listing.
	con_type	varchar(20)	YES		General type or classification: chemical, radiochemical, or physical property.

dbo.watch_list_status

Used by the DSAExperiment (DTCROVERRIDE) ASP application. Lookup table for watch list status

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			watch_list_status	varchar(30)	NO		The watch list status of a waste tank.
			description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_watch_list_status	watch_list_status	ASC	Yes	CLUSTERED
PK__watch_list_statu__12C03C95	watch_list_status	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.waste_tank	FK__waste_tan__watch__20D951C2	i_watch_list_status

dbo.XREF_WASTETYPE_PHASEASSIGNMENT

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>									
	waste_type	varchar(255)	YES		Abbeviation for the name of the waste. Some Examples: <table border="0"><tr><td>AW</td><td>CC</td><td>CP</td></tr><tr><td>DC</td><td>DN</td><td>IL</td></tr><tr><td>SLUDGE</td><td>SOLID_CAKE</td><td>SUPER</td></tr></table>	AW	CC	CP	DC	DN	IL	SLUDGE	SOLID_CAKE	SUPER
AW	CC	CP												
DC	DN	IL												
SLUDGE	SOLID_CAKE	SUPER												
	phase_assignment	varchar(255)	YES		Description which designates the phase assigned to the sample portion being measured or analyzed.									
	waste_site_id	varchar(30)	YES		Tank Identifier, using the format of 241-Tank Farm-Tank #.									
	xref_id	int	NO											

TWINS_CATALOG Database Documentation

Server name: twins

Database name: TCD

Documentation date: 7/28/2010

Database size: 332.44 MB

Database description:

Table summary

	<u>Name</u>	<u>Data size</u>	<u>Index size</u>	<u>Row Count</u>
	dbo.data_sources	200 KB	24 KB	113
	dbo.input_params	88 KB	24 KB	217
	dbo.output_fields	1152 KB	32 KB	2764
	dbo.SACS_ILL_Tanks	8 KB	16 KB	67
	dbo.sacs_sl_and_tptr_tanks	8 KB	8 KB	0
	dbo.tank_aliases	8 KB	16 KB	177
	dbo.user_input_params_and_values	1128 KB	48 KB	1
	dbo.What_Is_New	376 KB	24 KB	3162
	dbo.WIN_Category	8 KB	24 KB	10

9 table(s)

TABLES

dbo.data_sources

Used by the TWINS Data queries. Contains information about each data query.

I F P	Column name	Data type	Nulls	Default	Description
	subject_area_name	varchar(40)	NO		One of Best Basis Inventory, Internal Testing, Measurements Safety, Sample Analysis or Vapor. Corresponds to the subject areas in the TWINS data source selection. Field is informational only. It is not picked up by the UI code.
	source_name	varchar(80)	NO		The name of the data source, which can be a table, view or stored procedure callable by TWINS. Source_name is fully qualified as dbname.owner.object, for example: SACS.dbo.TWINS_IIL.V.
	server_name	varchar(30)	NO		The name of a computer used as a file server.
	descr_name	varchar(255)	YES		The name of the source or query that is displayed by the data query web form.
	type	varchar(5)	NO		Either proc or table. Tells the query whether to build a select statement (for tables) or to execute a stored procedure (for procs).
	source_description	text	NO		The description of the query.
	mime_type	varchar(40)	NO		Set to application/vnd.ms-excel. When the mime type is set to this value, the browser (Internet Explorer) knows to open the data set in Excel.
	owner_name	varchar(32)	YES		Not used.
	db_name	varchar(32)	YES		Not used. The database name is embedded in the source_name.
	keywords	varchar(255)	YES		Not used. Intent was to place in the Meta tag section of the web page.
	important_info	varchar(1000)	YES		Additional information about the data.
	important_info_title	varchar(30)	YES		Title that is displayed with the important_info text.

Indexes

	Index name	Column name	Sort direction	Is unique	Index type
i_data_sources		subject_area_name	ASC	Yes	CLUSTERED
i_data_sources		source_name	ASC	Yes	CLUSTERED

dbo.input_params

Each data source may have input parameters that are used to narrow down the TWINS data query. This input_param meta-data is by the TWINS query to build the input form.

I F P	Column name	Data type	Nulls	Default	Description
	source_name	varchar(80)	NO		The name of the data source, which can be a table, view or stored procedure callable by TWINS. Source_name is fully qualified as dbname.owner.object, for example: SACS.dbo.TWINS_IIL.V.
	parameter_name	varchar(30)	NO		Name of an input parameter which can be an attribute name if the source is a table or view, or an input parameter if the source is a stored procedure.
	descr_name	varchar(40)	NO		The displayable name of the parameter shown on the web page.
	control_type	varchar(10)	NO		Fixed or Date. Fixed parameter values are displayed in dropdown lists, while date values are shown in custom date range fields.
	call_sequence	int	NO		The order in which the parameters are displayed on the form.
	lookup_table_name	varchar(80)	YES		For fixed data types, the name of the table or view used to populate the dropdown list.
	lookup_field_name	varchar(30)	YES		The name of the field in the lookup table or view which is used to get data values for the dropdown list.
	select_all_allowed	varchar(1)	YES		A flag indicating whether or not this parameter can be ignored when retrieving data.
	param_description	varchar(255)	NO		A description of the input parameter.
	type	varchar(5)	NO	('table')	Indicates whether the lookup_table_name contains the name of a table (or view) or the name of a stored procedure.

Indexes

	Index name	Column name	Sort direction	Is unique	Index type
	i_input_params	source_name	ASC	Yes	CLUSTERED
	i_input_params	parameter_name	ASC	Yes	CLUSTERED

dbo.output_fields

Meta data describing the result set of the TWINS data query. Used by the TWINS query to retrieve and output the results.

I F P	Column name	Data type	Nulls	Default	Description
	source_name	varchar(80)	NO		The name of the data source, which can be a table, view or stored procedure callable by TWINS. Source_name is fully qualified as dbname.owner.object, for example: SACS.dbo.TWINS_IIL.V.
	field_name	varchar(60)	NO		The name of the table field or stored procedure output field.
	control_type	varchar(10)	NO		Fixed, numeric or date.
	descr_name	varchar(65)	YES		The name of the field that will be output as the column heading.
	display_sequence	int	NO		The order in which the field should be displayed in relation to the other fields with the same source_name.
	field_description	text	NO		Used by the data field definitions page to describe the meaning of the output field.
	comment	varchar(1000)	YES		Not used. Has the same value as the field_description.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_output_fields	source_name	ASC	Yes	CLUSTERED
i_output_fields	field_name	ASC	Yes	CLUSTERED

dbo.SACS_ILL_Tanks

A lookup list of the valid tank names found in the SACS interstitial liquid table.

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	tank	varchar(15)	NO		The unique identifier of the tank.

dbo.sacs_sl_and_tptr_tanks

A lookup list of the valid tank names found in the SACS temperature data table.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			tank	varchar(15)	NO		The unique identifier of the tank.

dbo.tank_aliases

Cross reference mapping of SACS and TWINS tank names.

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	waste_site_id	varchar(15)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	alias_name	varchar(15)	NO		An alternative version of the waste_site_id used for a tank such as 'A101' for 241-A-101.

dbo.user_input_params_and_values

Used to pass query parameters and values between the TWINSDATA applications *BuildQuery.aspx* and *ExtractData.aspx* forms.

I F P	Column name	Data type	Nulls	Default	Description
	user_ip_addr	varchar(15)	NO		Ip address of user recorded by the BuildQuery form, later used by the ExtractData form. This value identifies the query parameters.
	parameter_name	varchar(32)	NO		Name of an input parameter in Twins_catalog database.
	char_value	varchar(255)	YES		IF the parameter is a character type, then the character value for this parameter is stored in this field.
	int_value	int	YES		IF the parameter is an integer type, then the integer value for this parameter is stored in this field.
	date_value	datetime	YES		IF the parameter is a date type, then the date value for this parameter is stored in this field.
	entered_dttm	datetime	NO		The date and time that was entered by the user.
	source_name	varchar(80)	YES		Identifies the data source, which can be the name of a stored procedure, table or view.
	field_name	varchar(30)	YES		The name of the data source field.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_ip_addr	user_ip_addr	ASC		NONCLUSTERED

dbo.What_Is_New

Holds the What Is New information for the TWINS web interface. wwwcgi and public do NOT have select privileges on this table. They must select the data using the v_What_Is_New view, which selects records that can be displayed (See Display_Record field).

I	F	P	Column name	Data type	Nulls	Default	Description
			WIN_ID	int	NO		What Is New (WIN) Identifier
			Effective_On	datetime	NO		Date that this took place, or was in effect.
			WINC_ID	int	NO		What Is New Category (WINC) Identifier
			Description	varchar(800)	YES		Brief narrative describing a specific instance in this database table.
			Display_Record	char(1)	YES	('N')	If set to Y, then this record can be displayed for the end user, otherwise it is hidden.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
PK__What_Is_New__74794A92	WIN_ID	ASC	Yes	CLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__What_Is_N__WINC___756D6ECB	WINC_ID	PK__WIN_Category__72910220 (dbo.WIN_Category)	

dbo.WIN_Category

Different categories for the What Is New web page.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			WINC_ID	int	NO		What Is New Category (WINC) Identifier
			Category	varchar(30)	NO		Used to separate the different types.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK__WIN_Category__72910220	WINC_ID	ASC	Yes	CLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.What_Is_New	FK__What_Is_N__WINC__756D6ECB	PK__WIN_Category__72910220

TWINS_SUPPORT Database Documentation

Server name: twins

Database name: TWINS_SUPPORT

Documentation date: 7/28/2010

Database size: 207.25 MB

Database description:

Table summary

	Name	Data size	Index size	Row Count
	dbo.CurrentUser	16 KB	8 KB	6
	dbo.Dot_Net_Error_Log	1632 KB	8 KB	1065
	dbo.NavTreeNodes	40 KB	24 KB	252
	dbo.Roles	8 KB	24 KB	54
	dbo.TWINS_application_types	8 KB	8 KB	6
	dbo.TWINS_Applications	16 KB	40 KB	48
	dbo.TWINS_Applications_database_roles	16 KB	24 KB	46
	dbo.TWINS_Applications_db_permissions_xref	16 KB	24 KB	43
	dbo.TWINS_Applications_other_setup_xref	16 KB	24 KB	42
	dbo.TWINS_primary_backup_lkup	8 KB	8 KB	2
	dbo.TWINS_roles	8 KB	40 KB	17
	dbo.TWINS_staff	8 KB	40 KB	10
	dbo.TWINS_staff_roles	16 KB	24 KB	22
	dbo.UserRoles	8 KB	24 KB	335
	dbo.Users	8 KB	24 KB	71
	dbo.Users_of_AutoVector	8 KB	8 KB	0
	dbo.Users_of_CRTT	8 KB	8 KB	0

17 table(s)

TABLES

dbo.CurrentUser

Used by the TCROverride ASP application. Link between users and applications for getting correct modification name

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			UID	int	NO		User ID
			TWINS_Applications_id	int	NO		ID for records in the TWINS_Applications table
			CurrentUser_yn	char(1)	YES		Is this the current user

dbo.Dot_Net_Error_Log

Used by the (PNNL) STAFF ASP.NET application. Each exception thrown by a TWINS ASP.NET is logged in this table.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			DN_ID	int	NO		Unique record identifier.
			UserName	varchar(30)	YES		Login name of the person who experience the error while performing some task.
			Displayable_Name	varchar(30)	YES		Person's name.
			App_Directory	varchar(30)	YES		Directory where the application aspx code is stored
			URL	nvarchar(512)	YES		URL for the web page
			Method	nvarchar(100)	YES		Thrown exceptions method.
			Message	nvarchar(512)	YES		Message from the thrown exception.
			Type	nvarchar(512)	YES		Thrown exception type.
			Source	nvarchar(512)	YES		Pointer to the source (code) of the error.
			Stack_Trace	text	YES		Stack trace from the thrown exception.
			User_Comment	nvarchar(512)	YES		After the exception is caught, the user has a chance to record what he/she was doing when the error occurred. This information is stored (if provided).
			Developer_Comment	nvarchar(512)	YES		Any comment by the developer relating to this error.
			Bug_ID	int	YES		Link to any bug that was created in response to this error occurring.
			State	nvarchar(6)	YES	('OPEN')	OPEN or CLOSED.
			Occurred_On	datetime	YES	(getdate())	Date and time that the error occurred

dbo.NavTreeNodes

Referenced by the Menu Control ASP.NET application. Information is used to create the left navigation menu of the different ASP.NET applications.

I	F	P	Column name	Data type	Nulls	Default	Description
			NodeID	int	NO		Unique record identifier.
			RID	int	NO		Role ID
			ParentNodeID	int	YES		The identifier of this nodes parent. If there is no parent, then the value is NULL.
			Title	varchar(80)	NO		The text displayed as a menu item (node).
			URL	varchar(120)	YES		The link to the web page for this menu item (node).
			Description	varchar(255)	YES		Displayed when the user hovers over the menu link. Gives detailed information about the menu item.
			DisplayOnMenu	bit	YES (1)		Should this line item be displayed on the menu
			OrderBy	int	NO (0)		Order in which this item should be displayed on the menu.
			LineFeed	bit	NO (0)		Indicates whether an extra line feed should be used after the menu item.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
PK_NavTreeNodes_NodeID	NodeID	ASC	Yes	CLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK_NavTreeNodes_RID	RID	PK_Roles_RID (dbo.Roles)	

dbo.Roles

ASPX roles. Referenced by the Menu Control ASP.NET application.

I	F	P	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			RID	int	NO		Role ID
			TWINS_Applications_id	int	NO		ID for records in the TWINS_Applications table
			RoleName	varchar(50)	NO		Role name for aspx purposes
			Description	varchar(255)	YES		Brief narrative describing a specific instance in this database table.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK_Roles_RID	RID	ASC	Yes	CLUSTERED

Foreign keys

<u>Constraint name</u>	<u>Column name</u>	<u>Reference</u>	<u>Description</u>
FK_Roles_TWINS_App_ID	TWINS_Applications_id	pk_TWINS_Applications_id (dbo.TWINS_Applications)	

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.NavTreeNodes	FK_NavTreeNodes_RID	PK_Roles_RID
dbo.UserRoles	FK_UserRoles_RID	PK_Roles_RID

dbo.TWINS_application_types

Application types in TWINS

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			application_type	varchar(60)	YES		Type of user interface application. Examples are Access or WebForms

dbo.TWINS_Applications

List of user interface TWINS applications. Referenced by the Menu Control ASP.NET application.

I	F	P	Column name	Data type	Nulls	Default	Description
			TWINS_Applications_id	int	NO		ID for records in the TWINS_Applications table
			application_name	varchar(60)	NO		Name of the application that permissions are being established for
			application_type	varchar(30)	NO		Type of user interface application. Examples are Access or WebForms
			author_id	int	NO		Author ID
			placed_in_production	datetime	YES		Date application was placed in production
			custodian_id	int	YES		Staff ID of the application custodian
			modifier_id	int	YES		Staff ID of last person to modify the application
			last_modified_date	datetime	YES		Date of last modification to data.
			comment	varchar(3000)	YES		Comment
			description	varchar(3000)	YES		Brief narrative describing a specific instance in this database table.
			key_contacts	varchar(200)	YES		Key contacts for this application
			last_modification_id	int	YES		Bug ID or Change Request ID of last modification
			last_modification_type	char(3)	YES		Was the last modification a Bug or Change Request
			app_directory	varchar(50)	YES		Directory where the application aspx code is stored

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_name	application_name	ASC	Yes	NONCLUSTERED
pk_TWINS_Applications_id	TWINS_Applications_id	ASC	Yes	CLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
fk_author	author_id	pk_TWINS_staff_id (dbo.TWINS_staff)	
fk_custodian_id	custodian_id	pk_TWINS_staff_id (dbo.TWINS_staff)	
fk_modifier_id	modifier_id	pk_TWINS_staff_id (dbo.TWINS_staff)	

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.Roles	FK_Roles_TWINS_App_ID	pk_TWINS_Applications_id
dbo.TWINS_Applications_db_permissions_xref	fk_TWINS_Applications_id	pk_TWINS_Applications_id
dbo.TWINS_Applications_other_setup_xref	fk_TWINS_Applications_id2	pk_TWINS_Applications_id

dbo.TWINS_Applications_database_roles

List of database roles in TWINS

I	F	P	Column name	Data type	Nulls	Default	Description
			TWINS_Applications_database_roles_id	int	NO		ID for records in the TWINS_Applications_database_roles table
			database_role	varchar(60)	NO		Role in the database application to this application
			description	varchar(3000)	YES		Brief narrative describing a specific instance in this database table.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_database_role	database_role	ASC	Yes	NONCLUSTERED

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.TWINS_Applications_db_permissions_xref	fk_database_role	i_database_role

dbo.TWINS_Applications_db_permissions_xref

Database permissions needed for TWINS applications

I	F	P	Column name	Data type	Nulls	Default	Description
			TWINS_Applications_db_permissions_xref_id	int	NO		ID for records in the TWINS_Applications_db_permissions_xref table
			TWINS_Applications_id	int	NO		ID for records in the TWINS_Applications table
			database_name	varchar(60)	NO		The name of a database.
			database_role	varchar(60)	NO		Role in the database needed for this application to work properly.
			applies_to	varchar(255)	YES		Users or groups within the database for which this role needs to be applied.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_db_xref	TWINS_Applications_id	ASC	Yes	NONCLUSTERED
i_db_xref	database_name	ASC	Yes	NONCLUSTERED
i_db_xref	database_role	ASC	Yes	NONCLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
fk_database_role	database_role	i_database_role ()	
fk_TWINS_Applications_id	TWINS_Applications_id	pk_TWINS_Applications_id (dbo.TWINS_Applications)	

dbo.TWINS_Applications_other_setup_xref

Other permissions needed for TWINS applications

I	F	P	Column name	Data type	Nulls	Default	Description
			TWINS_Applications_other_setup_xref_id	int	NO		ID for records in the TWINS_Applications_other_setup_xref table
			TWINS_Applications_id	int	NO		ID for records in the TWINS_Applications table
			table_name	varchar(60)	YES		The name of the BBIM table in which the user modified a record.
			location_type	varchar(30)	YES		Permissions need to be set elsewhere than database -- this field holds the of object (example -- DTS, Access table) that must be modified
			location_name	varchar(60)	YES		Permissions need to be set elsewhere than database -- this field holds the name of the table (or other object) that must be modified
			setup_comment	varchar(4000)	YES		Addition comment about setting up permissions for this application
			applies_to	varchar(255)	YES		Users or groups within the database for which these instructions need to be applied.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_setup_xref	TWINS_Applications_id	ASC		NONCLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
fk_TWINS_Applications_id2	TWINS_Applications_id	pk_TWINS_Applications_id (dbo.TWINS_Applications)	

dbo.TWINS_primary_backup_ikup

Lookup table for TWINS role clarification

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	primary_backup	varchar(60)	YES		Is this user the primary or backup person for the role

dbo.TWINS_roles

Roles users can assume in TWINS. Referenced by the Menu Control ASP.NET application.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			TWINS_roles_id	int	NO		ID for records in the TWINS_Roles table
			TWINS_role	varchar(60)	NO		Role user plays in TWINS

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_role	TWINS_role	ASC	Yes	NONCLUSTERED
pk_TWINS_roles_id	TWINS_roles_id	ASC	Yes	CLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.TWINS_staff_roles	fk_TWINS_roles_id	pk_TWINS_roles_id

dbo.TWINS_staff_roles

Roles of the TWINS staff

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			TWINS_staff_roles_id	int	NO		ID for records in the TWINS_Staff_Roles table
			TWINS_staff_id	int	NO		ID for records in the TWINS_Staff table
			TWINS_roles_id	int	NO		ID for records in the TWINS_Roles table
			primary_backup	varchar(10)	YES		Is this user the primary or backup person for the role

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_staff_roles	TWINS_staff_id	ASC	Yes	NONCLUSTERED
i_staff_roles	TWINS_roles_id	ASC	Yes	NONCLUSTERED

Foreign keys

<u>Constraint name</u>	<u>Column name</u>	<u>Reference</u>	<u>Description</u>
fk_TWINS_roles_id	TWINS_roles_id	pk_TWINS_roles_id (dbo.TWINS_roles)	
fk_TWINS_staff_id	TWINS_staff_id	pk_TWINS_staff_id (dbo.TWINS_staff)	

dbo.UserRoles

Link between users and roles for ASPX. Referenced by the Menu Control ASP.NET application.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			UID	int	NO		User ID
			RID	int	NO		Role ID

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK_UserRoles	UID	ASC	Yes	CLUSTERED
PK_UserRoles	RID	ASC	Yes	CLUSTERED

Foreign keys

<u>Constraint name</u>	<u>Column name</u>	<u>Reference</u>	<u>Description</u>
FK_UserRoles_RID	RID	PK_Roles_RID (dbo.Roles)	
FK_UserRoles_UID	UID	PK_Users_UID (dbo.Users)	

dbo.Users

ASPX users. Referenced by the Menu Control ASP.NET application.

I	F	P	Column name	Data type	Nulls	Default	Description
			UID	int	NO		User ID
			UserName	varchar(30)	NO		Name or identifier of person requesting an AutoTCR report.
			Password	varchar(255)	YES		User password
			FirstName	varchar(30)	NO		First Name of the person.
			LastName	varchar(30)	NO		Last Name of the person.
			EmailAddr	varchar(50)	NO		User email address
			displayable_name	varchar(30)	NO		Name desired to display in modified by fields

Indexes

Index name	Column name	Sort direction	Is unique	Index type
PK_Users_UID	UID	ASC	Yes	CLUSTERED

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.UserRoles	FK_UserRoles_UID	PK_Users_UID

dbo.Users_of_AutoVector

Specific information needed only for AutoVector users

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	UID	int	NO		User ID
	SendMail_yn	char(1)	NO		Send mail yes/no
	Comment	varchar(255)	YES		Comment

dbo.Users_of_CRTT

Specific information needed only for CRTT users

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	UID	int	NO		User ID
	isCurrent_yn	char(1)	NO		Is current staff for CRTT
	isSuperUser_yn	char(1)	NO		Is super user for CRTT
	isProjectManager_yn	char(1)	NO		Is project manager for CRTT

TXFR Database Documentation

Server name: twins

Database name: TXFR

Documentation date: 7/28/2010

Database size: 700.00 MB

Database description:

Table summary

	Name	Data size	Index size	Row Count
	dbo.Adjustment	8 KB	32 KB	9
	dbo.Generator	8 KB	16 KB	10
	dbo.Generator_Stream	8 KB	16 KB	18
	dbo.Generator_Stream_Phase	8 KB	16 KB	11
	dbo.Generator_Transfer	16 KB	16 KB	506
	dbo.Help	8 KB	16 KB	14
	dbo.Phase_Volume	568 KB	16 KB	11185
	dbo.Sink	8 KB	16 KB	1
	dbo.Sink_Transfer	8 KB	16 KB	1
	dbo.Stream_Constituent	8 KB	16 KB	12
	dbo.Tank_Transfer	56 KB	8 KB	506
	dbo.Tank_Volume	376 KB	224 KB	3839
	dbo.Volume_Adjustment	40 KB	16 KB	2317

13 table(s)

TABLES

dbo.Adjustment

Defines a type of change in volume that may occur to waste in a tank.

I	F	P	Column name	Data type	Nulls	Default	Description
			Adjustment_ID	int	NO		Unique identifier of the record.
			Adjustment_Name	varchar(40)	NO		Name of the adjustment: Analysis (+/-) Condensate (+/-) Flush Water Gas (+/-) General Variance (+/-) Instrumentation (+/-) Rebaseline Stabilization Water Waste Volume Reduction
			BBI_Inv_Affected	bit	NO		A flag indicating whether the inventory of the BBI would be affected if a Volume_Adjustment is uploaded to the BBI. The only options are that inventory is affected (changed) and concentrations remain the same, or visa versa. Analysis (+/-) and Rebaseline are the only Adjustments that affect inventory.
			Description	varchar(255)	YES		A general description of the adjustment.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
PK__Adjustment__22401542	Adjustment_ID	ASC	Yes	CLUSTERED
UQ__Adjustment__2DB1C7EE	Adjustment_Name	ASC	Yes	NONCLUSTERED

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.Volume_Adjustment	FK__Volume_Ad__Adjus__382F5661	PK__Adjustment__22401542

dbo.Generator

Any facility or operations that generates High Level Radioactive Waste (HLW), or waste that will end up in HLW storage tanks.

I	F	P	Column name	Data type	Nulls	Default	Description
			Generator_ID	int	NO		Unique identifier of the record.
			Generator_Name	varchar(40)	NO		The name of the generator of waste: 151-AZ 222S Labs 242A Evaporator Corrosion Control ER-311 Other PFP Retrieval Tank Farms WESF
			Description	varchar(255)	YES		General description of this generator of waste.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
PK__Generator__2334397B	Generator_ID	ASC	Yes	CLUSTERED

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.Generator_Stream	FK__Generator__Gener__2EA5EC27	PK__Generator__2334397B

dbo.Generator_Stream

Characterizes the waste stream which is produced by a generating facility or operations.

I	F	P	Column name	Data type	Nulls	Default	Description
			GS_ID	int	NO		Unique identifier of the record.
			Generator_ID	int	YES		Unique (foreign key) ID of the Generator.
			Stream_Name	varchar(60)	NO		The name of the waste stream. Note that there may be several instances of Generator_Stream records that have the same Stream_Names. For example, there may be a couple of different 'Oxalic Acid' waste streams, but if they have different concentrations of Oxalic Acid, they have to be distinct. Some examples: 151-AZ waste 222S Waste ER-311 waste Evaporator Water Flush Water NaNO2 NaOH Non-Defined Waste Stream Oxalic Acid PFP Waste Raw Water Waste Volume Reduction WESF Waste
			Start_Date	datetime	YES		Information to further identify/describe this Generator Waste Stream. Not sure if this field is necessary.
			End_Date	datetime	YES		Information to further identify/describe this Generator Waste Stream. Not sure if this field is necessary.
			Description	varchar(255)	YES		A general description of this Generator Waste Stream.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
PK__Generator_Stream__24285DB4	GS_ID	ASC	Yes	CLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__Generator__Gener__2EA5EC27	Generator_ID	PK__Generator__2334397B (dbo.Generator)	

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.Generator_Stream_Phase	FK__Generator__GS_ID__2F9A1060	PK__Generator_Stream__24285DB4
dbo.Generator_Transfer	FK__Generator__GS_ID__308E3499	PK__Generator_Stream__24285DB4

dbo.Generator_Stream_Phase

The waste phase portion of the generator stream.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			GSP_ID	int	NO		Unique record identifier.
			GS_ID	int	YES		Link to the Generator Stream record.
			Waste_Phase	varchar(50)	NO		The name of the waste phase. In addition, a value of "Total" is used to define all of the waste phases in a tank.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK_Generator_Stream__251C81ED	GSP_ID	ASC	Yes	CLUSTERED

Foreign keys

<u>Constraint name</u>	<u>Column name</u>	<u>Reference</u>	<u>Description</u>
FK_Generator__GS_ID__2F9A1060	GS_ID	PK_Generator_Stream__24285DB4 (dbo.Generator_Stream)	

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.Stream_Constituent	FK_Stream_Co__GSP_I__3552E9B6	PK_Generator_Stream__251C81ED

dbo.Generator_Transfer

A transfer of waste from a Generator to a Tank is one of the ways that a Tank_Volume may change and is a sub-type of the super-type Tank_Volume.

I	F	P	Column name	Data type	Nulls	Default	Description
			TV_ID	int	NO		Unique Identifier of Generator_Transfer. Since Generator_Transfer is a sub-type of Tank_Volume, Tank_Volume has a corresponding TV_ID value.
			GS_ID	int	YES		Identifier (foreign key) of the Generator Stream.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
PK__Generator_Transf__2610A626	TV_ID	ASC	Yes	CLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__Generator__GS_ID__308E3499	GS_ID	PK__Generator_Stream__24285DB4 (dbo.Generator_Stream)	
FK__Generator__TV_ID__318258D2	TV_ID	PK__Tank_Volume__2BC97F7C (dbo.Tank_Volume)	

dbo.Help

Contains the information displayed when the help icon is clicked for each page in the TXFR .NET application.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			HID	int	NO		Unique record identifier. The ASP.NET web page uses this identifier to retrieve the appropriate help record.
			Title	varchar(100)	NO		Shown as the title of the help page.
			Body	varchar(4000)	YES		The help text displayed on the help page.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK__Help__2704CA5F	HID	ASC	Yes	CLUSTERED

dbo.Sink

A facility that receives waste. All tracking of the waste (by TWINS applications) ends after it is received by the sink.

I	F	P	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			Sink_ID	int	NO		Unique identity of the record.
			Sink_Name	varchar(60)	NO		The name of the Sink. The only current Sink in the system is 6AW.
			Description	varchar(255)	YES		A general description of the sink.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK__Sink__28ED12D1	Sink_ID	ASC	Yes	CLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.Sink_Transfer	FK__Sink_Trans__Sink___336AA144	PK__Sink__28ED12D1

dbo.Sink_Transfer

A transfer of waste from a tank to a sink is one of the ways that a Tank_Volume may change and is a sub-type of the super-type Tank_Volume.

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			TV_ID	int	NO		Unique Identifier of Sink_Transfer. Since Sink_Transfer is a sub-type of Tank_Volume, Tank_Volume has a corresponding TV_ID value.
			Sink_ID	int	YES		Identifier (foreign key) of the Sink.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK__Sink_Transfer__29E1370A	TV_ID	ASC	Yes	CLUSTERED

Foreign keys

<u>Constraint name</u>	<u>Column name</u>	<u>Reference</u>	<u>Description</u>
FK__Sink_Tran__Sink__336AA144	Sink_ID	PK__Sink__28ED12D1 (dbo.Sink)	
FK__Sink_Tran__TV_ID__345EC57D	TV_ID	PK__Tank_Volume__2BC97F7C (dbo.Tank_Volume)	

dbo.Stream_Constituent

Stream_Constituent contains the concentration of each constituent in a given Generator_Stream. It is important that a Generator_Stream be characterized accurately prior to uploading an associated Generator_Transfer record into the BBI database. This is imperative if the BBI is to maintain an accurate inventory.

I	F	P	Column name	Data type	Nulls	Default	Description
			SC_ID	int	NO		Unique identifier of the record.
			GSP_ID	int	NO		Generator Stream ID (foreign key).
			Con_ID	varchar(15)	NO		Unique identifier of the constituent. Same as the CAS (Chemical Abstract Services) number, if a CAS number exists for this constituent.
			Concentration	float	NO		The concentration of this constituent in the stream.
			Units	varchar(12)	NO		The allowable units are: ug/mL uCi/mL wt% Enforcement of these is controlled by the .NET application.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
PK__Stream_Constitue__2AD55B43	SC_ID	ASC	Yes	CLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__Stream_Co__GSP_I__3552E9B6	GSP_ID	PK__Generator_Stream__251C81E D (dbo.Generator_Stream_Phase)	

dbo.Tank_Transfer

A transfer of waste from a source Tank to a sink Tank is one of the ways that a Tank_Volume may change and is a sub-type of the super-type Tank_Volume.

I	F	P	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			Source_TV_ID	int	YES		The Unique Identifier of Source Tank_Volume. Since Tank_Transfer is a sub-type of Tank_Volume, Tank_Volume has a corresponding TV_ID value.
			Sink_TV_ID	int	YES		The Unique Identifier of Sink Tank_Volume. Since Tank_Transfer is a sub-type of Tank_Volume, Tank_Volume has a corresponding TV_ID value.

Foreign keys

<u>Constraint name</u>	<u>Column name</u>	<u>Reference</u>	<u>Description</u>
FK__Tank_Tran__Sink___3647 0DEF	Sink_TV_ID	PK__Tank_Volume__2BC97F7C (dbo.Tank_Volume)	
FK__Tank_Tran__Sourc___373B 3228	Source_TV_ID	PK__Tank_Volume__2BC97F7C (dbo.Tank_Volume)	

dbo.Tank_Volume

Tank_Volume captures the change in the volume of the tank at a given time. Tank_Volume is a "super-type" entity in that it contains the common data (fields) needed for recording transfers. The unique fields needed for recording transfers are found in the "sub-type" entities: Volume_Adjustment; Tank_Transfer; Sink_Transfer and Generator_Transfer. Note that the volume is calculated by summing the Phase_Volume.Vol_kL values.

I	F	P	Column name	Data type	Nulls	Default	Description
			TV_ID	int	NO		Unique record identifier.
			Waste_Site_ID	varchar(15)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
			Start_Date	datetime	NO		The date that the change in volume event began.
			Sort_Order	int	NO		If multiple Tank_Volume records for a given tank have the same Start_Date, then Sort_Order is used to determine the sequence of the changes. Records with greater Sort_Order within a Waste_Site_ID and Start_Date are more recent.
			End_Date	datetime	YES		Used for changes in volume that occur over a period of days.
			Comment	varchar(255)	YES		Free format comment for the user.
			txfr_event_id	int	YES		Used only to assist in copying data from the previous version of the TXFR application. Links to the tcd.dbo.TXFR_Event table.
			txfr_type	varchar(4)	YES		Indicates the type of transfer that caused the change in volume: {sink adj tank gen}
			Locked	bit	NO		The user has the option of locking this Tank's volume (associated Phase_Volume records) when rippling of volume changes occurs.
			Upload_To_BBI	bit	NO		The Uploaded_To_BBI, Uploaded_To_BBI_On and Upload_Start_Date dictate the State of the data as follows: Uploaded_To_BBI, Upload_Start_Date, Uploade_To_BBI_On : State 0, null, null : Event Not Uploaded To BB 1, null, null : Event in Queue to be Uploaded 1, datetime, null : Event being uploaded 1, datetime, datetime : Event Uploaded to BBI 0, datetime, datetime : Event in Queue to be Unloaded from BBI 0, null, datetime : Event being Unloaded See Uploade_To_BBI defintion
			Uploaded_To_BBI_On	datetime	YES		See Uploade_To_BBI defintion
			Upload_Start_Date	datetime	YES		See Uploade_To_BBI defintion

Indexes

Index name	Column name	Sort direction	Is unique	Index type
PK__Tank_Volume__2BC97F7C	TV_ID	ASC	Yes	CLUSTERED
UNQ__Tank_Volume	Waste_Site_ID	ASC	Yes	NONCLUSTERED
UNQ__Tank_Volume	Start_Date	ASC	Yes	NONCLUSTERED
UNQ__Tank_Volume	Sort_Order	ASC	Yes	NONCLUSTERED

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.Generator_Transfer	FK__Generator__TV_ID__318258D2	PK__Tank_Volume__2BC97F7C
dbo.Phase_Volume	FK__Phase_Vol__TV_ID__32767D0B	PK__Tank_Volume__2BC97F7C
dbo.Sink_Transfer	FK__Sink_Tran__TV_ID__345EC57D	PK__Tank_Volume__2BC97F7C
dbo.Tank_Transfer	FK__Tank_Tran__Sink__36470DEF	PK__Tank_Volume__2BC97F7C
dbo.Tank_Transfer	FK__Tank_Tran__Sourc__373B3228	PK__Tank_Volume__2BC97F7C
dbo.Volume_Adjustment	FK__Volume_Ad__TV_ID__39237A9A	PK__Tank_Volume__2BC97F7C

dbo.Volume_Adjustment

Associates a specific Adjustment with a change in Tank_Volume. It is one of the ways that a Tank_Volume may change and is a sub-type of the super-type Tank_Volume.

I	F	P	Column name	Data type	Nulls	Default	Description
			TV_ID	int	NO		Unique Identifier of Volume_Adjustment. Since Volume_Adjustment is a sub-type of Tank_Volume, Tank_Volume has a corresponding TV_ID value.
			Adjustment_ID	int	YES		Adjustment Identifier (foreign key).

Indexes

Index name	Column name	Sort direction	Is unique	Index type
PK__Volume_Adjustmen__2CBDA3B5	TV_ID	ASC	Yes	CLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
FK__Volume_Ad__Adjus__382F5661	Adjustment_ID	PK__Adjustment__22401542 (dbo.Adjustment)	
FK__Volume_Ad__TV_ID__39237A9A	TV_ID	PK__Tank_Volume__2BC97F7C (dbo.Tank_Volume)	

VAPOR Database Documentation

Server name: twins

Database name: VAPOR

Documentation date: 7/28/2010

Database size: 909.19 MB

Database description:

Table summary

	Name	Data size	Index size	Row Count
	dbo.agg_level	8 KB	24 KB	3
	dbo.analysis_method_desc	8 KB	40 KB	30
	dbo.analysis_results	38760 KB	10792 KB	119469
	dbo.analysis_results_denormalized	55120 KB	6256 KB	119052
	dbo.analysis_results_to_load	488 KB	24 KB	149
	dbo.avrgd_rprtd_values	7448 KB	7768 KB	29693
	dbo.chem_name_descWEB	560 KB	1000 KB	8508
	dbo.chemical_class_desc	840 KB	16 KB	6655
	dbo.chemical_id_desc	96 KB	80 KB	2052
	dbo.chemical_name_desc	648 KB	1656 KB	8508
	dbo.chemical_sort_order	8 KB	16 KB	9
	dbo.DatabaseVersion	8 KB	24 KB	1
	dbo.flammability_data	8 KB	24 KB	5
	dbo.flammability_results	288 KB	216 KB	1480
	dbo.industrial_hygiene_data	408 KB	40 KB	2340
	dbo.industrial_hygiene_results	1392 KB	56 KB	9005
	dbo.laboratory_desc	8 KB	32 KB	13
	dbo.log_analysis_results	43096 KB	12256 KB	124061
	dbo.log_analysis_results_to_load	52416 KB	8 KB	125583
	dbo.log_insert_counts	72 KB	8 KB	333
	dbo.log_reference_info	328 KB	8 KB	1062
	dbo.log_sample_data	1072 KB	160 KB	3459
	dbo.log_sample_reference_relation	360 KB	8 KB	6604
	dbo.max_lengths	88 KB	8 KB	133
	dbo.monitoring_data	8 KB	24 KB	100
	dbo.monitoring_results	56 KB	80 KB	369
	dbo.project_type	8 KB	24 KB	3
	dbo.qa_samp_type	8 KB	24 KB	7
	dbo.qualifier_desc	8 KB	16 KB	14
	dbo.radiological_analysis_results	368 KB	8 KB	1414
	dbo.radiological_sample_data	24 KB	8 KB	127
	dbo.reference_info	152 KB	128 KB	509
	dbo.reference_type	8 KB	24 KB	13
	dbo.result_type	8 KB	24 KB	7
	dbo.sample_data	736 KB	1192 KB	3295
	dbo.sample_data_to_load	56 KB	24 KB	4
	dbo.sample_device_desc	8 KB	24 KB	29
	dbo.sample_method_desc	8 KB	24 KB	10
	dbo.sample_reference_relation	168 KB	256 KB	6604
	dbo.SHMS_grab_sample_info	160 KB	32 KB	887
	dbo.SHMS_grab_sample_results	1352 KB	40 KB	9927
	dbo.tankfarm_use	8 KB	16 KB	24
	dbo.units	8 KB	24 KB	15
	dbo.us_states	8 KB	24 KB	51
	dbo.waste_tank	16 KB	96 KB	397

45 table(s)

TABLES

dbo.agg_level

Aggregation Level

I	F	P	Column name	Data type	Nulls	Default	Description
			agg_level	varchar(30)	NO		Label or descriptor applied to the whole, i.e. "aggregate", sample before separation into portions or fractions prior to analysis. See /dictionary/AggLevel.asp >Aggregation Level Definitions for more information.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
PK__agg_level__18B6AB08	agg_level	ASC	Yes	NONCLUSTERED

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.sample_data	fk_agg_level	PK__agg_level__18B6AB08

dbo.analysis_method_desc

Analysis methods used in the laboratories for waste tank data

I	F	P	Column name	Data type	Nulls	Default	Description
			analysis_method	varchar(50)	NO		Primary instrumentation used to analyze a sample (a sample is defined as a discrete quantity of material collected for analysis).
			analysis_method_name	varchar(255)	NO		The complete , non-abbreviated, description of an analysis method.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_analysis_method_desc	analysis_method	ASC	Yes	CLUSTERED
PK__analysis_method___4C6B5938	analysis_method	ASC	Yes	NONCLUSTERED

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.analysis_results	fk_analysis_method	i_analysis_method_desc

dbo.analysis_results

Analysis results

I F P	Column name	Data type	Nulls	Default	Description
PK	result_id	int	NO		Integer ID field in AutoTCR, TCD, and TCD_READ databases. Char(12) ID field created from date and time in Vapor database.
FK	collection_id	int	NO		Unique identifier created from date and time.
FK	chemical_id	varchar(15)	NO		ID of the constituent used for reporting purposes. Typically, the id is the Chemical Abstract Services (CAS) listing.
	reported_value	float	YES		Estimated or quantified concentration of a chemical as reported by the analytical lab. Mass per unit volume reported values are at the Reference Temperature and 1.01325 bar pressure.
	reporting_limit	char(20)	YES		A lower limit below which the concentration of a constituent is not reported. The concentration of the constituent either cannot be measured or is below an instrument detection limit or below a contract required quantitation limit. This is a text field and cannot be included in numerical calculations. For more information on reporting limits, see the corresponding references
FK	units	varchar(10)	NO		Concentration units of the reported value or the reporting limit.
	ref_temperature	float	NO		Temperature for which mass per unit volume concentration results are given. Units are degrees C.
	conc_mgm3_25C	char(20)	YES		Estimated or quantified concentration of a chemical in units of mg/m3 at a reference temperature of 25C and reference pressure of 1.01325.
	analysis_date	datetime	YES		Date of sample analysis. Format is mm/dd/yy.
	elapsed_hold_time	int	YES		Allowed time in days between sample collection and sample analysis. Value given is the shorter of any administratively specified or published laboratory analytical holding time study criteria.
	holding_time	int	YES		Allowed time in days between sample collection and sample analysis. Value given is the shorter of any administratively specified or published laboratory analytical holding time study criteria.
	basis_hold_time	varchar(100)	YES		Basis of specified allowed holding time. Specifies whether allowed holding time was based on administrative requirements or on laboratory validated holding times.
	data_qual_code	char(10)	YES		Single-letter flags that indicate noteworthy aspects of sample result. See Vapor Qualifier Definitions for more information.
FK	analysis_method	varchar(50)	NO		Primary instrumentation used to analyze a sample (a sample is defined as a discrete quantity of material collected for analysis).
FK	lab_abbrev	varchar(6)	NO		Acronym of laboratory performing analyses.
	comments_results	varchar(255)	YES		Any concerns or unusual information about the results.
FK	reference_id	int	NO		ID field created from date and time in the Vapor database.
	page_num	char(15)	YES		Page number or range of pages where the information was cited
	TPA_Flag	varchar(3)	NO		Data and information flagged 'TPA' in this database are post-1989 analytical data and are recognized by the Washington State Department of Ecology for use in Tri- Party Agreement (TPA) and regulatory mandated documents. This data may be used for production of TPA deliverable documents.
	load_date_time	datetime	YES	(getdate())	The date and time that the record was loaded into the TCD.
	access_level	tinyint	NO	(1)	Access control mechanism for data in database table.
FK	result_type	varchar(50)	NO		Description of the type of results the analysis produced. Examples include: PRIMARY, PRIMARY#2
	lab_sample_id	varchar(30)	YES		Identifier code assigned to a sample by the laboratory.

review_status	varchar(30)	YES	Review status. Current values are: REVIEWED, REVIEWED: SEE NOTE, CHECKING NOT APPLIED, DATA SUSPECT: SEE COMMENT , REVIEWED: NOT FOR INVENTORY and DATA RESOLUTION IN PROGRESS. See Review Status Caveats for more information.
review_date	datetime	YES	Date on which the review was performed.
reviewer	varchar(30)	YES	Reviewer is the name of the individual who reviewed the data or "ComputerCheck" indicating data was compared against a defined set of criteria resulting in the data being flagged as possibly suspect.
review_comment	varchar(4000)	YES	Comment added by reviewer of this analysis result record.
analysis_results_file_name	varchar(255)	YES	File name saying where the data came from

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i1_analysis_results	chemical_id	ASC		CLUSTERED
i2_analysis_results	result_id	ASC		NONCLUSTERED
i2_analysis_results	collection_id	ASC		NONCLUSTERED
i3_analysis_results	collection_id	ASC		NONCLUSTERED
PK__analysis_results__498EEC8D	result_id	ASC	Yes	NONCLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
fk_lab_abbrev	lab_abbrev	i_laboratory_desc ()	
fk_collection_id	collection_id	PK__sample_data__671F4F74 (dbo.sample_data)	
fk_units	units	PK__units__634EBE90 (dbo.units)	
fk_chemical_id	chemical_id	i_chem_id_desc ()	
fk_result_type	result_type	PK__result_type__793DFFAF (dbo.result_type)	
fk_reference_id	reference_id	i_reference_info ()	
fk_analysis_method	analysis_method	i_analysis_method_desc ()	

Triggers

Trigger name:	dbo.t_analysis_results_del	Created on:	05 Jan 2005
Trigger type:	DELETE	Trigger active:	Yes
Trigger name:	dbo.t_analysis_results_ins_upd	Created on:	05 Jan 2005
Trigger type:	INSERT, UPDATE	Trigger active:	Yes

dbo.analysis_results_denormalized

Snapshot of analysis_results combined with sample_data and chemical_name_desc – created to improve UI performance

I F P	Column name	Data type	Nulls	Default	Description
	waste_site_id	varchar(30)	NO		Tank Identifier, using the format of 241-Tank Farm-Tank #.
	chemical_name	varchar(240)	NO		The name of the constituent used for reporting purposes. Typically, the name comes from the Chemical Abstract Services (CAS) listing.
	molecular_weight	float	YES		The molecular weight of the constituent is needed for converting M (moles/L) to inventory units (kg).
	chemical_id	varchar(15)	NO		ID of the constituent used for reporting purposes. Typically, the id is the Chemical Abstract Services (CAS) listing.
	agg_level	varchar(30)	NO		Label or descriptor applied to the whole, i.e. "aggregate", sample before separation into portions or fractions prior to analysis. See /dictionary/AggLevel.asp >Aggregation Level Definitions for more information.
	QA_samp_type	varchar(30)	NO		QA type of sample.
	reported_value	float	YES		Estimated or quantified concentration of a chemical as reported by the analytical lab. Mass per unit volume reported values are at the Reference Temperature and 1.01325 bar pressure.
	reporting_limit	char(20)	YES		A lower limit below which the concentration of a constituent is not reported. The concentration of the constituent either cannot be measured or is below an instrument detection limit or below a contract required quantitation limit. This is a text field and cannot be included in numerical calculations. For more information on reporting limits, see the corresponding references
	units	varchar(10)	NO		Concentration units of the reported value or the reporting limit.
	ref_temperature	float	NO		Temperature for which mass per unit volume concentration results are given. Units are degrees C.
	conc_mgm3_25C	char(20)	YES		Estimated or quantified concentration of a chemical in units of mg/m3 at a reference temperature of 25C and reference pressure of 1.01325.
	ppmv	varchar(22)	YES		Parts per million.
	data_qual_code	char(10)	YES		Single-letter flags that indicate noteworthy aspects of sample result. See /dictionary/VaporQualifiers.asp >Vapor Qualifier Definitions for more information.
	sample_device	varchar(30)	YES		Description of the type of container used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
	sampling_method	varchar(20)	NO		Description of the type of general method used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
	analysis_method	varchar(50)	NO		Primary instrumentation used to analyze a sample (a sample is defined as a discrete quantity of material collected for analysis).
	lab_abbrev	varchar(6)	NO		Acronym of laboratory performing analyses.
	field_sample_id	varchar(45)	NO		Code assigned to a sample by the collecting organization (a sample is defined as a discrete quantity of material collected for analysis).
	lab_sample_id	varchar(30)	YES		Identifier code assigned to a sample by the laboratory.
	dtm_sample	datetime	NO		Date and time the sample was taken (a sample is defined as a discrete quantity of material collected for analysis).
	analysis_date	datetime	YES		Date of sample analysis. Format is mm/dd/yy.
	elapsed_hold_time	int	YES		Allowed time in days between sample collection and sample analysis. Value given is the shorter of any administratively specified or published laboratory analytical holding time study criteria.
	holding_time	int	YES		Allowed time in days between sample collection and sample analysis. Value given is the shorter of any administratively specified or published laboratory analytical holding time study criteria.

basis_hold_time	varchar(100)	YES	Basis of specified allowed holding time. Specifies whether allowed holding time was based on administrative requirements or on laboratory validated holding times.
document_num	varchar(30)	YES	Code assigned by government agencies or government contractors to documents and correspondence they publish; the code is used for tracking purposes. This code includes revision numbers. This code does not include the UC category, which is used by the Office of Scientific and Technical Information.
reference_title	varchar(255)	NO	Title of the reference as it appears on the title page. Subtitles are entered after a colon. If one volume of a set is referenced, then the volume title and volume number should be reported as part of the title
comments_results	varchar(255)	YES	Any concerns or unusual information about the results.
TPA_flag	varchar(10)	YES	Data and information flagged 'TPA' in this database are post-1989 analytical data and are recognized by the Washington State Department of Ecology for use in Tri- Party Agreement (TPA) and regulatory mandated documents. This data may be used for production of TPA deliverable documents.
access_level	tinyint	NO	Access control mechanism for data in database table.
result_type	varchar(50)	NO	Description of the type of results the analysis produced. Examples include: PRIMARY, PRIMARY#2
 result_id	int	YES	Integer ID field in AutoTCR, TCD, and TCD_READ databases.
table_name	varchar(60)	YES	Char(12) ID field created from date and time in Vapor database. The name of the BBIM table in which the user modified a record.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_id	result_id	ASC		NONCLUSTERED
i_tank_chemical	waste_site_id	ASC		NONCLUSTERED
i_tank_chemical	chemical_id	ASC		NONCLUSTERED

dbo.analysis_results_to_load

Analysis results awaiting review and approval to be loaded into the analysis_results table

I F P	Column name	Data type	Nulls	Default	Description
	analysis_results_to_load_id	int	NO		Unique ID for the analysis_results_to_load table
	field_sample_id	varchar(255)	YES		Code assigned to a sample by the collecting organization (a sample is defined as a discrete quantity of material collected for analysis).
	lab_sample_id	varchar(255)	YES		Identifier code assigned to a sample by the laboratory.
	chemical_name	varchar(255)	YES		The name of the constituent used for reporting purposes. Typically, the name comes from the Chemical Abstract Services (CAS) listing.
	chemical_id	varchar(255)	YES		ID of the constituent used for reporting purposes. Typically, the id is the Chemical Abstract Services (CAS) listing.
	molecular_weight	varchar(255)	YES		The molecular weight of the constituent is needed for converting M (moles/L) to inventory units (kg).
	ref_temperature	varchar(255)	YES		Temperature for which mass per unit volume concentration results are given. Units are degrees C.
	analysis_date	varchar(255)	YES		Date of sample analysis. Format is mm/dd/yy.
	result_type	varchar(255)	YES		Description of the type of results the analysis produced. Examples include: PRIMARY, PRIMARY#2
	reported_value	varchar(255)	YES		Estimated or quantified concentration of a chemical as reported by the analytical lab. Mass per unit volume reported values are at the Reference Temperature and 1.01325 bar pressure.
	units	varchar(255)	YES		Concentration units of the reported value or the reporting limit.
	reporting_limit	varchar(255)	YES		A lower limit below which the concentration of a constituent is not reported. The concentration of the constituent either cannot be measured or is below an instrument detection limit or below a contract required quantitation limit. This is a text field and cannot be included in numerical calculations. For more information on reporting limits, see the corresponding references
	data_qual_code	varchar(255)	YES		Single-letter flags that indicate noteworthy aspects of sample result. See Vapor Qualifier Definitions for more information.
	holding_time	varchar(255)	YES		Allowed time in days between sample collection and sample analysis. Value given is the shorter of any administratively specified or published laboratory analytical holding time study criteria.
	basis_hold_time	varchar(255)	YES		Basis of specified allowed holding time. Specifies whether allowed holding time was based on administrative requirements or on laboratory validated holding times.
	analysis_method	varchar(255)	YES		Primary instrumentation used to analyze a sample (a sample is defined as a discrete quantity of material collected for analysis).
	analysis_method_name	varchar(255)	YES		The complete, non-abbreviated, description of an analysis method.
	lab_abbrev	varchar(255)	YES		Acronym of laboratory performing analyses.
	comments_results	varchar(255)	YES		Any concerns or unusual information about the results.
	load_date_time	datetime	YES		The date and time that the record was loaded into the TCD.
	ready_to_load	char(1)	YES		Flag indicating record is ready to be loaded
	analysis_results_file_name	varchar(255)	YES		File name saying where the data came from
	review_comment	varchar(6000)	YES		Comment added by reviewer of this analysis result record.

Indexes

	Index name	Column name	Sort direction	Is unique	Index type
i_id		analysis_results_to_load_id	ASC	Yes	NONCLUSTERED

Triggers

Trigger name:	dbo.t_analysis_results_to_load_del	Created on:	03 Jan 2005
Trigger type:	DELETE	Trigger active:	Yes
Trigger name:	dbo.t_analysis_results_to_load_ins_upd	Created on:	03 Jan 2005
Trigger type:	INSERT, UPDATE	Trigger active:	Yes

dbo.avrgd_rprtd_values

Contains simple statistical values for reported analytical results which are not flagged as suspect ("S" in data qualifier field). Statistics are derived from reported values displayed in the Analysis Results table.

I F P	Column name	Data type	Nulls	Default	Description
	result_average_id	int	NO		ID field created from date and time in the Vapor database.
	tank_id	varchar(30)	NO		Code that identifies each tank in a tank farm system.
	sample_job_num	varchar(20)	YES		Reference number of sampling job assigned by sample collecting organization. This number is the first 5 characters of the Sample Id.
	sample_date	varchar(10)	YES		For vectors with a type of "Sample", Sample_Date contains the date that the sample was taken. Sometimes this data is not readily available.
	chemical_id	varchar(15)	NO		ID of the constituent used for reporting purposes. Typically, the id is the Chemical Abstract Services (CAS) listing.
	result_average	float	YES		Arithmetic mean of all non-zero results for an identified analyte which are not flagged as 'suspect' in the data qualifier code field. (Replicate analysis results are not included in the calculation.) The result average is specific for the tank, sample job #, sample device, and analytical laboratory and is derived from the Analysis Results table.
	result_st_dev	float	YES		Standard deviation of all nonsuspect results for identified analyte.
	reporting_limit	char(20)	YES		A lower limit below which the concentration of a constituent is not reported. The concentration of the constituent either cannot be measured or is below an instrument detection limit or below a contract required quantitation limit. This is a text field and cannot be included in numerical calculations. For more information on reporting limits, see the corresponding references
	units	varchar(10)	YES		Concentration units of the reported value or the reporting limit.
	ref_temperature	float	NO		Temperature for which mass per unit volume concentration results are given. Units are degrees C.
	num_sample_results	int	YES		Number of sample results used in "result average".
	data_qual_code	char(10)	YES		Single-letter flags that indicate noteworthy aspects of sample result. See /dictionary/VaporQualifiers.asp for more information.
	sample_device	varchar(30)	YES		Description of the type of container used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
	lab_abbrev	varchar(6)	NO		Acronym of laboratory performing analyses.
	new_record	char(1)	YES		Flag indicating a new record in the Vapor database.
	sampling_method	char(20)	YES		Description of the type of general method used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
	chemical_name	varchar(240)	YES		The name of the constituent used for reporting purposes. Typically, the name comes from the Chemical Abstract Services (CAS) listing.
	analysis_method	varchar(50)	YES		Primary instrumentation used to analyze a sample (a sample is defined as a discrete quantity of material collected for analysis).
	molecular_weight	float	YES		The molecular weight of the constituent is needed for converting M (moles/L) to inventory units (kg).
	mgm3	float	YES		Milligrams per meter cubed.
	ppmv	float	YES		Parts per million.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i1_avrgd_rprtd_values	chemical_id	ASC		CLUSTERED
i2_avrgd_rprtd_values	tank_id	ASC		NONCLUSTERED
i2_avrgd_rprtd_values	sample_job_num	ASC		NONCLUSTERED
i2_avrgd_rprtd_values	chemical_id	ASC		NONCLUSTERED

i2_avrgd_rprtd_values	lab_abbrev	ASC	NONCLUSTERED
i3_avrgd_rprtd_values	sample_job_num	ASC	NONCLUSTERED

dbo.chem_name_descWEB

Chemical Synonyms used by the TWINS views

I F P	Column name	Data type	Nulls	Default	Description
	chemical_name_descWEB_id	int	NO		Unique identified for the chemical_name_descWEB table
	chemical_id	varchar(15)	NO		ID of the constituent used for reporting purposes. Typically, the id is the Chemical Abstract Services (CAS) listing.
	chemical_name	varchar(240)	NO		The name of the constituent used for reporting purposes. Typically, the name comes from the Chemical Abstract Services (CAS) listing.
	frequency_used	char(15)	NO		Whether a chemical name is the standard or a synonym in the Vapor database
	analysis_use	char(1)	YES		Are there analysis results for this constituent or chemical in this database? Y or N.
	average_use	char(1)	YES		Is/is not in the averages table.
	test_data_use	char(1)	YES		This chemical_id appears in results from 'Test Data' project type in the Vapor database. Y or N.
	IMUST_data_use	char(1)	YES		Indicates if chemical exists in IMUST data in the Vapor database.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i1_chem_name_descWEB	chemical_id	ASC	Yes	CLUSTERED
i1_chem_name_descWEB	chemical_name	ASC	Yes	CLUSTERED
i2_chem_name_descWEB	chemical_name	ASC		NONCLUSTERED
i3_chem_name_descWEB	chemical_id	ASC		NONCLUSTERED

dbo.chemical_class_desc

Chemical Class

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			chemical_class_desc_id	int	NO		Unique identified for the chemical_class_desc table
			chemical_id	varchar(15)	NO		ID of the constituent used for reporting purposes. Typically, the id is the Chemical Abstract Services (CAS) listing.
			chemical_class	varchar(50)	NO		Name of the element or functionality group of the vapor chemicals.
			comments_classif	varchar(255)	YES		Comments on the particular classification of the chemical.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_chem_class_desc	chemical_id	ASC	Yes	CLUSTERED
i_chem_class_desc	chemical_class	ASC	Yes	CLUSTERED

dbo.chemical_id_desc

Chemicals and radionuclides found in vapor area of the Hanford tanks.

I	F	P	Column name	Data type	Nulls	Default	Description
			chemical_id	varchar(15)	NO		ID of the constituent used for reporting purposes. Typically, the id is the Chemical Abstract Services (CAS) listing.
			molecular_weight	float	YES		The molecular weight of the constituent is needed for converting M (moles/L) to inventory units (kg).
			structure_formula	varchar(255)	YES		The chemical formula of an analyte in the vapor database.
			odor	varchar(255)	YES		Description of odor of a chemical in vapor database.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_chem_id_desc	chemical_id	ASC	Yes	CLUSTERED
PK__chemical_id_desc__4B7734FF	chemical_id	ASC	Yes	NONCLUSTERED

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.analysis_results	fk_chemical_id	i_chem_id_desc

dbo.chemical_sort_order

Chemical Sort Order

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			chemical_id	char(15)	NO		ID of the constituent used for reporting purposes. Typically, the id is the Chemical Abstract Services (CAS) listing.
			derived_flamm_sort	int	YES		Sort order for derived flammability
			direct_flamm_sort	int	YES		Sort order for direct flammability

dbo.DatabaseVersion

Database Version

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			Version_number	numeric(18,0)	NO		A number indicating the version of a database or software. Appears as both type float and char.
			Version_date	datetime	NO		Database version date
			Version_description	varchar(255)	YES		Database version description

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_Version_number	Version_number	ASC	Yes	CLUSTERED

dbo.flammability_data

Sampling event data for the flammability results

I F P	Column name	Data type	Nulls	Default	Description
	chemical_id	char(15)	NO		ID of the constituent used for reporting purposes. Typically, the id is the Chemical Abstract Services (CAS) listing.
	lower_flamm_limit	int	NO		Lower flammability limit of identified analytes in air at standard temperature and pressure.
	lfl_units	char(10)	NO		Units of lower flammability limit.
	reference_id	int	YES		ID field created from date and time in the Vapor database.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_flamm_data	chemical_id	ASC		CLUSTERED
i_flamm_data	lower_flamm_limit	ASC		CLUSTERED

dbo.flammability_results

Contains headspace flammability data, expressed as percent of Lower Flammability Limit (LFL), for selected analytes for the specified tanks, by sample date.

I F P	Column name	Data type	Nulls	Default	Description
	tank_id	varchar(30)	YES		Code that identifies each tank in a tank farm system.
	sample_job_num	varchar(20)	NO		Reference number of sampling job assigned by sample collecting organization. This number is the first 5 characters of the Sample Id.
	sample_date	char(10)	NO		For vectors with a type of "Sample", Sample_Date contains the date that the sample was taken. Sometimes this data is not readily available.
	chemical_name	varchar(20)	NO		The name of the constituent used for reporting purposes. Typically, the name comes from the Chemical Abstract Services (CAS) listing.
	lower_flamm_limit	int	YES		Lower flammability limit of identified analytes in air at standard temperature and pressure.
	lfl_units	char(10)	YES		Units of lower flammability limit.
	avg_concentrations	char(15)	YES		Arithmetic mean of all non-zero results for an identified analyte which are not flagged as "suspect" in the data qualifier code field. (Replicate analysis results are not included in the calculation.) The result average is specific for the tank, sample job #, sample device, and analytical laboratory. If no results are above reporting limits, then reporting limit is given. These data are derived from the Result Average table.
	units	char(10)	YES		Concentration units of the reported value or the reporting limit.
	percent_lfl	char(15)	NO		Result average times 100 divided by the lower flammability limit of the analyte.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i1_flamm_results	tank_id	ASC		CLUSTERED
i1_flamm_results	sample_job_num	ASC		CLUSTERED
i2_flamm_results	tank_id	ASC		NONCLUSTERED
i2_flamm_results	sample_job_num	ASC		NONCLUSTERED
i2_flamm_results	sample_date	ASC		NONCLUSTERED
i2_flamm_results	chemical_name	ASC		NONCLUSTERED

dbo.industrial_hygiene_data

Used by the TCROverride ASP application. Sampling event data for the industrial hygiene analysis results

I	F	P	Column name	Data type	Nulls	Default	Description
			ih_data_id	char(12)	NO		ID value created from date and time in the Vapor database.
			tank_id	char(15)	NO		Code that identifies each tank in a tank farm system.
			survey_date_time	datetime	NO		Date and time that the sampling was performed.
			survey_id	char(20)	YES		Unique identifier of the industrial hygiene survey.
			tank_riser	char(5)	YES		Riser number on the waste tank.
			probe_depth	float	YES		Distance in meters from top face of the riser flange to the instrument inlet.
			location	char(20)	YES		Location in the tank of the industrial hygiene sample.
			job_activity	varchar(30)	YES		Job activity occurring when the industrial hygiene sample was collected.
			load_date_time	datetime	YES		The date and time that the record was loaded into the TCD.
			last_modified_date_time	datetime	YES		Date on which the corresponding change record was entered into the TCD. This same date is applied to each analyte changed as part of the respective change log. Once the data has been entered into TCD, the change is immediately visible via the queries available through the TWINS2 interface.
			comments_ih_data	varchar(255)	YES		Comments or any concerns or unusual information about the industrial hygiene data.
			match_analysis_sample	char(1)	YES		This industrial hygiene sample has a matching sample in the analysis samples.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_industrial_hygiene_data	tank_id	ASC		CLUSTERED
i_industrial_hygiene_data	survey_date_time	ASC		CLUSTERED
i_industrial_hygiene_data	tank_riser	ASC		CLUSTERED
i2_industrial_hygiene_data	ih_data_id	ASC		NONCLUSTERED
PK_ih_data_id	ih_data_id	ASC	Yes	NONCLUSTERED

Triggers

Trigger name: dbo.t_indust_hygiene_data_ins_upd
 Trigger type: INSERT, UPDATE

Created on: 27 Apr 2000
 Trigger active: Yes

dbo.industrial_hygiene_results

Contains industrial hygiene filed monitoring results for personal and area measurements.

I	F	P	Column name	Data type	Nulls	Default	Description
			ih_result_id	char(12)	NO		ID value created from date and time in the Vapor database.
			ih_data_id	char(12)	NO		ID value created from date and time in the Vapor database.
			measurement_type	varchar(45)	NO		Type of measurement reported.
			result_value	float	YES		Estimated or quantified concentration of a chemical.
			reporting_limit	char(20)	YES		A lower limit below which the concentration of a constituent is not reported. The concentration of the constituent either cannot be measured or is below an instrument detection limit or below a contract required quantitation limit. This is a text field and cannot be included in numerical calculations. For more information on reporting limits, see the corresponding references
			units	char(10)	NO		Concentration units of the reported value or the reporting limit.
			data_qual_code	char(10)	YES		Single-letter flags that indicate noteworthy aspects of sample result. See Vapor Qualifier Definitions for more information.
			analysis_method	varchar(50)	YES		Primary instrumentation used to analyze a sample (a sample is defined as a discrete quantity of material collected for analysis).
			monit_instru_model	char(20)	YES		Model designation of monitoring instrument.
			monit_instru_calib	varchar(45)	YES		Method used to calibrate monitoring instrument.
			load_date_time	datetime	YES		The date and time that the record was loaded into the TCD.
			last_modified_date_time	datetime	YES		Date on which the corresponding change record was entered into the TCD. This same date is applied to each analyte changed as part of the respective change log. Once the data has been entered into TCD, the change is immediately visible via the queries available through the TWINS2 interface.
			comments_ih_results	varchar(255)	YES		Comment for industrial hygiene result in Vapor and Vapor_stage databases.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i1_industrial_hygiene_results	measurement_type	ASC		CLUSTERED
i2_industrial_hygiene_results	ih_result_id	ASC		NONCLUSTERED
i2_industrial_hygiene_results	ih_data_id	ASC		NONCLUSTERED
i3_industrial_hygiene_results	ih_data_id	ASC		NONCLUSTERED
PK_industrial_hygiene_results	ih_result_id	ASC	Yes	NONCLUSTERED

Triggers

Trigger name: dbo.t_indust_hy_results_ins_upd
 Trigger type: INSERT, UPDATE

Created on: 27 Apr 2000
 Trigger active: Yes

dbo.laboratory_desc

Laboratory Description

I	F	P	Column name	Data type	Nulls	Default	Description
			lab_abbrev	varchar(6)	NO		Acronym of laboratory performing analyses.
			lab_name	varchar(255)	NO		Name of laboratory in Vapor database.
			lab_city	varchar(30)	YES		City of laboratory address in Vapor database.
			lab_state	varchar(2)	YES		State of address of laboratory in Vapor database.
			lab_contact	char(14)	YES		Name of contact person at laboratory in Vapor database.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_laboratory_desc	lab_abbrev	ASC	Yes	CLUSTERED
PK__laboratory_desc__4A8310C6	lab_abbrev	ASC	Yes	NONCLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
fk_lab_state	lab_state	PK__us_states__625A9A57 (dbo.us_states)	

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.analysis_results	fk_lab_abbrev	i_laboratory_desc

dbo.log_analysis_results

Log for changes to the analysis_results data

I F P	Column name	Data type	Nulls	Default	Description
	log_analysis_results_id	int	NO		Unique ID for the log_analysis_results table
	change_date_time	datetime	YES		The date and time that a change was made to the record.
	change_status	varchar(20)	YES		Change status (either modified or deleted)
	result_id	int	NO		Integer ID field in AutoTCR, TCD, and TCD_READ databases.
	collection_id	int	YES		Char(12) ID field created from date and time in Vapor database.
	chemical_id	varchar(15)	YES		Unique identifier created from date and time.
	reported_value	float	YES		ID of the constituent used for reporting purposes. Typically, the id is the Chemical Abstract Services (CAS) listing.
	reporting_limit	char(20)	YES		Estimated or quantified concentration of a chemical as reported by the analytical lab. Mass per unit volume reported values are at the Reference Temperature and 1.01325 bar pressure.
	units	varchar(10)	YES		A lower limit below which the concentration of a constituent is not reported. The concentration of the constituent either cannot be measured or is below an instrument detection limit or below a contract required quantitation limit. This is a text field and cannot be included in numerical calculations. For more information on reporting limits, see the corresponding references
	ref_temperature	float	NO		Concentration units of the reported value or the reporting limit.
	conc_mgm3_25C	char(20)	YES		Temperature for which mass per unit volume concentration results are given. Units are degrees C.
	analysis_date	datetime	YES		Estimated or quantified concentration of a chemical in units of mg/m3 at a reference temperature of 25C and reference pressure of 1.01325.
	elapsed_hold_time	int	YES		Date of sample analysis. Format is mm/dd/yy.
	holding_time	int	YES		Allowed time in days between sample collection and sample analysis. Value given is the shorter of any administratively specified or published laboratory analytical holding time study criteria.
	basis_hold_time	varchar(100)	YES		Allowed time in days between sample collection and sample analysis. Value given is the shorter of any administratively specified or published laboratory analytical holding time study criteria.
	data_qual_code	char(10)	YES		Basis of specified allowed holding time. Specifies whether allowed holding time was based on administrative requirements or on laboratory validated holding times.
	analysis_method	varchar(50)	NO		Single-letter flags that indicate noteworthy aspects of sample result. See "/dictionary/VaporQualifiers.asp" for more information.
	lab_abbrev	varchar(6)	YES		Primary instrumentation used to analyze a sample (a sample is defined as a discrete quantity of material collected for analysis).
	comments_results	varchar(255)	YES		Acronym of laboratory performing analyses.
	reference_id	int	YES		Any concerns or unusual information about the results.
	page_num	char(15)	YES		ID field created from date and time in the Vapor database.
	TPA_Flag	varchar(3)	YES		Page number or range of pages where the information was cited
	load_date_time	datetime	YES		Data and information flagged 'TPA' in this database are post-1989 analytical data and are recognized by the Washington State Department of Ecology for use in Tri- Party Agreement (TPA) and regulatory mandated documents. This data may be used for production of TPA deliverable documents.
	access_level	tinyint	YES		The date and time that the record was loaded into the TCD.
					Access control mechanism for data in database table.

result_type	varchar(50)	YES	Description of the type of results the analysis produced. Examples include: PRIMARY, PRIMARY#2
lab_sample_id	varchar(30)	YES	Identifier code assigned to a sample by the laboratory.
review_status	varchar(30)	YES	Review status. Current values are: REVIEWED, REVIEWED: SEE NOTE, CHECKING NOT APPLIED, DATA SUSPECT: SEE COMMENT , REVIEWED: NOT FOR INVENTORY and DATA RESOLUTION IN PROGRESS. See /dictionary/ReviewStatusCaveats.asp >Review Status Caveats for more information.
review_date	datetime	YES	Date on which the review was performed.
reviewer	varchar(30)	YES	Reviewer is the name of the individual who reviewed the data or "ComputerCheck" indicating data was compared against a defined set of criteria resulting in the data being flagged as possibly suspect.
review_comment	varchar(4000)	YES	Comment added by reviewer of this analysis result record.
analysis_results_file_name	varchar(255)	YES	File name saying where the data came from
chemical_name	varchar(240)	YES	The name of the constituent used for reporting purposes. Typically, the name comes from the Chemical Abstract Services (CAS) listing.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_chemical_id	chemical_id	ASC		NONCLUSTERED
i_collection_id	collection_id	ASC		NONCLUSTERED
i_id	collection_id	ASC		NONCLUSTERED
i_id	change_date_time	ASC		NONCLUSTERED

dbo.log_analysis_results_to_load

Log of analysis_results_to_load

I F P	Column name	Data type	Nulls	Default	Description
	log_analysis_results_to_load_id	int	NO		Unique identified for the log_analysis_results_to_load table
	change_date_time	datetime	YES		The date and time that a change was made to the record.
	change_status	varchar(10)	YES		Change status (either modified or deleted)
	analysis_results_to_load_id	int	YES		Unique ID for the analysis_results_to_load table
	field_sample_id	varchar(255)	YES		Code assigned to a sample by the collecting organization (a sample is defined as a discrete quantity of material collected for analysis).
	lab_sample_id	varchar(255)	YES		Identifier code assigned to a sample by the laboratory.
	chemical_name	varchar(255)	YES		The name of the constituent used for reporting purposes. Typically, the name comes from the Chemical Abstract Services (CAS) listing.
	chemical_id	varchar(255)	YES		ID of the constituent used for reporting purposes. Typically, the id is the Chemical Abstract Services (CAS) listing.
	molecular_weight	varchar(255)	YES		The molecular weight of the constituent is needed for converting M (moles/L) to inventory units (kg).
	ref_temperature	varchar(255)	YES		Temperature for which mass per unit volume concentration results are given. Units are degrees C.
	analysis_date	varchar(255)	YES		Date of sample analysis. Format is mm/dd/yy.
	result_type	varchar(255)	YES		Description of the type of results the analysis produced. Examples include: PRIMARY, PRIMARY#2
	reported_value	varchar(255)	YES		Estimated or quantified concentration of a chemical as reported by the analytical lab. Mass per unit volume reported values are at the Reference Temperature and 1.01325 bar pressure.
	units	varchar(255)	YES		Concentration units of the reported value or the reporting limit.
	reporting_limit	varchar(255)	YES		A lower limit below which the concentration of a constituent is not reported. The concentration of the constituent either cannot be measured or is below an instrument detection limit or below a contract required quantitation limit. This is a text field and cannot be included in numerical calculations. For more information on reporting limits, see the corresponding references
	data_qual_code	varchar(255)	YES		Single-letter flags that indicate noteworthy aspects of sample result. See Vapor Qualifier Definitions for more information.
	holding_time	varchar(255)	YES		Allowed time in days between sample collection and sample analysis. Value given is the shorter of any administratively specified or published laboratory analytical holding time study criteria.
	basis_hold_time	varchar(255)	YES		Basis of specified allowed holding time. Specifies whether allowed holding time was based on administrative requirements or on laboratory validated holding times.
	analysis_method	varchar(255)	YES		Primary instrumentation used to analyze a sample (a sample is defined as a discrete quantity of material collected for analysis).
	analysis_method_name	varchar(255)	YES		The complete, non-abbreviated, description of an analysis method.
	lab_abbrev	varchar(255)	YES		Acronym of laboratory performing analyses.
	comments_results	varchar(255)	YES		Any concerns or unusual information about the results.
	load_date_time	datetime	YES		The date and time that the record was loaded into the TCD.
	ready_to_load	char(1)	YES		Flag indicating record is ready to be loaded
	analysis_results_file_name	varchar(255)	YES		File name saying where the data came from
	review_comment	varchar(6000)	YES		Comment added by reviewer of this analysis result record.

dbo.log_insert_counts

Log of records inserted -- used by UI

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			log_insert_counts_id	int	NO		Unique identified for the log_insert_counts table
			load_date	datetime	YES		Date record was loaded
			tank_dttm_id	varchar(100)	YES		Tank and date
			record_count	int	YES		Record count
			record_type	varchar(20)	YES		Record type

dbo.log_reference_info

Log for changes to the reference_info data

I	F	P	Column name	Data type	Nulls	Default	Description
			log_reference_info_id	int	NO		Unique ID for the log_reference_info table
			change_date_time	datetime	YES		The date and time that a change was made to the record.
			change_status	varchar(20)	YES		Change status (either modified or deleted)
			reference_id	int	NO		ID field created from date and time in the Vapor database.
			reference_type	varchar(20)	YES		The type of reference.
			access_public	char(1)	YES		The document is available to the public Y or N.
			authors	varchar(255)	NO		Names of authors of report or similar.
			year_published	varchar(4)	YES		The year in which the report or other document was published.
			reference_title	varchar(255)	NO		Title of the reference as it appears on the title page. Subtitles are entered after a colon. If one volume of a set is referenced, then the volume title and volume number should be reported as part of the title
			document_num	varchar(30)	YES		Code assigned by government agencies or government contractors to documents and correspondence they publish; the code is used for tracking purposes. This code includes revision numbers. This code does not include the UC category, which is used by the Office of Scientific and Technical Information.
			publisher	varchar(255)	YES		Publisher of a report or other reference.
			publisher_city	varchar(30)	YES		City part of address of a publisher of a report or other reference.
			publisher_state	char(2)	YES		State part of address of a publisher of a report or other reference.
			publisher_country	varchar(30)	YES		Country part of address of publisher of a report or other reference.
			comments_reference	varchar(255)	YES		Comment on reference information in Vapor and Vapor_stage databases.

dbo.log_sample_data

Log for changes to the sample_data data

I	F	P	Column name	Data type	Nulls	Default	Description
			log_sample_data_id	int	NO		Unique ID for the log_sample_data table
			change_date_time	datetime	YES		The date and time that a change was made to the record.
			change_status	varchar(20)	YES		Change status (either modified or deleted)
			collection_id	int	NO		Unique identifier created from date and time.
			field_sample_id	varchar(45)	YES		Code assigned to a sample by the collecting organization (a sample is defined as a discrete quantity of material collected for analysis).
			tank_id	varchar(30)	YES		Code that identifies each tank in a tank farm system.
			sampling_access	char(20)	YES		Sampling port, access, or riser from which the sample was taken (a sample is defined as a discrete quantity of material collected for analysis).
			tank_temp	float	YES		Temperature inside the waste tank when the sample was taken (units are °C) (a sample is defined as a discrete quantity of material collected for analysis).
			barometric_pressure	float	YES		Barometric pressure (units are torr) measured at the start of the sample collection. When the VSS was used to collect samples (see Sampling Method) measurements were made inside the headspace. When the ISVS or ISS was used to collect samples, ambient pressure at time of sampling was used as reported by the Hanford Meteorological Station.
			dtm_sample	datetime	NO		Date and time the sample was taken (a sample is defined as a discrete quantity of material collected for analysis).
			sample_device	varchar(20)	YES		Description of the type of container used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
			sampling_method	varchar(20)	YES		Description of the type of general method used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
			comments_sampling	varchar(255)	YES		Any concerns or unusual information about collecting the sample (a sample is defined as a discrete quantity of material collected for analysis).
			project_type	char(40)	YES		The project type of a sample.
			agg_level	char(30)	YES		Label or descriptor applied to the whole, i.e. "aggregate", sample before separation into portions or fractions prior to analysis. See /dictionary/AggLevel.asp for more information.
			QA_samp_type	char(30)	YES		QA type of sample.
			sample_job_num	varchar(20)	YES		Reference number of sampling job assigned by sample collecting organization. This number is the first 5 characters of the Sample Id.
			work_package	varchar(30)	YES		Work Package Number associated with the sampling event
			load_date_time	datetime	YES		The date and time that the record was loaded into the TCD.
			ih_survey_number	varchar(30)	YES		Industrial Hygiene survey number
			volume_l	float	YES		Volume, measured in liters.

Indexes

	Index name	Column name	Sort direction	Is unique	Index type
i_id		collection_id	ASC		NONCLUSTERED
i_id		change_date_time	ASC		NONCLUSTERED

dbo.log_sample_reference_relation

Log for changes to the sample_reference_relation data

I	F	P	Column name	Data type	Nulls	Default	Description
			log_sample_reference_relation_id	int	NO		Unique ID for the log_sample_reference_relation table
			change_date_time	datetime	YES		The date and time that a change was made to the record.
			change_status	varchar(20)	YES		Change status (either modified or deleted)
			sample_reference_relation_id	int	YES		Unique identified for the sample_reference_relation table
			collection_id	int	YES		Unique identifier created from date and time.
			reference_id	int	NO		ID field created from date and time in the Vapor database.

dbo.max_lengths

Maximum length of data in each field -- used by UI

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			max_lengths_id	int	NO		Unique identified for the max_lengths table
			table_name	varchar(100)	YES		The name of the BBIM table in which the user modified a record.
			field_name	varchar(100)	YES		The name of the table field where the data was modified.
			max_length	int	YES		Maximum length of data

dbo.monitoring_data

Monitoring Data

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			monit_data_id	char(12)	NO		ID field created from date and time in the Vapor database.
			tank_id	char(15)	NO		Code that identifies each tank in a tank farm system.
			tank_riser	char(5)	NO		Riser number on the waste tank.
			probe_depth	float	NO		Distance in meters from top face of the riser flange to the instrument inlet.
			dttm_monitoring	datetime	NO		Date and time that the monitoring was performed.
			comments_monitoring	varchar(255)	YES		Any concerns or unusual information about the monitoring event.
			reference_id	int	YES		ID field created from date and time in the Vapor database.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
i_monitoring_data	tank_id	ASC		CLUSTERED
i_monitoring_data	tank_riser	ASC		CLUSTERED
i_monitoring_data	probe_depth	ASC		CLUSTERED
i_monitoring_data	dttm_monitoring	ASC		CLUSTERED
i2_monitoring_data	monit_data_id	ASC		NONCLUSTERED

dbo.monitoring_results

Monitoring Results

I	F	P	Column name	Data type	Nulls	Default	Description
			monit_result_id	char(12)	NO		ID field created from date and time in the Vapor database.
			monit_data_id	char(12)	NO		ID field created from date and time in the Vapor database.
			measurement_type	varchar(45)	NO		Type of measurement reported.
			result_value	float	YES		Estimated or quantified concentration of a chemical.
			reporting_limit	char(20)	YES		A lower limit below which the concentration of a constituent is not reported. The concentration of the constituent either cannot be measured or is below an instrument detection limit or below a contract required quantitation limit. This is a text field and cannot be included in numerical calculations. For more information on reporting limits, see the corresponding references
			units	char(10)	NO		Concentration units of the reported value or the reporting limit.
			data_qual_code	char(10)	YES		Single-letter flags that indicate noteworthy aspects of sample result. See Vapor Qualifier Definitions for more information.
			analysis_method	varchar(50)	NO		Primary instrumentation used to analyze a sample (a sample is defined as a discrete quantity of material collected for analysis).
			monit_instru_model	char(20)	YES		Model designation of monitoring instrument.
			monit_instru_calib	varchar(45)	YES		Method used to calibrate monitoring instrument.
			comments_monit_results	varchar(255)	YES		Any concerns or unusual information about the results.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i1_monitoring_results	measurement_type	ASC		CLUSTERED
i2_monitoring_results	monit_result_id	ASC		NONCLUSTERED
i2_monitoring_results	monit_data_id	ASC		NONCLUSTERED
i3_monitoring_results	monit_data_id	ASC		NONCLUSTERED

dbo.project_type

Project type

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			project_type	varchar(40)	NO		The project type of a sample.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK__project_type__15DA3E5D	project_type	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.sample_data	fk_project_type	PK__project_type__15DA3E5D

dbo.qa_samp_type

QA Sample Type

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			qa_samp_type	varchar(30)	NO		QA type of sample.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK__qa_samp_type__1C873BEC	qa_samp_type	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.sample_data	fk_qa_samp_type	PK__qa_samp_type__1C873BEC

dbo.qualifier_desc

TWINS Query: Qualifiers used in the vapor database

I F P	Column name	Data type	Nulls	Default	Description
	data_qual_code	varchar(1)	YES		<p>Single-letter flags that indicate noteworthy aspects of sample result. See /dictionary/VaporQualifiers.asp for more information.</p> <p>B Compound found in associated laboratory blank as well as sample C Based on EPA contract laboratory protocol D Target analyte reported is the result of a secondary dilution E Reported concentration was above the instrumental calibration range F Target analyte reported in sample was also found in field blank above action limit G Chosen by Numatec Hanford Corporation - Special Analytical Support H Analysis was performed after allowed analytical holding time had elapsed or analysis date is not available J Chosen by the tank vapor database (TVD) staff unique to TVD K Reported concentration was estimated L Based on EPA contract laboratory protocol M Target analyte was absent (not detected) from sample N Chosen by Numatec Hanford Corporation - Special Analytical Support O Compound is a tentatively identified compound. Includes both chemical library matches, chemist identified compounds, and unknowns. P Based on EPA contract laboratory protocol Q Target analyte was detected, but at concentrations less than Vapor Program Required Quantitation Limits R Chosen by Numatec Hanford Corporation - Special Analytical Support S Result suspect - see comment field T Chosen by the tank vapor database (TVD) staff unique to TVD U Target analyte reported in sample was also found in trip blank above action limit V Chosen by Numatec Hanford Corporation - Special Analytical Support W Compound was analyzed for but not detected above reporting value X Based on EPA contract laboratory protocol Y Analytical laboratory did not have approved quality assurance documentation, or that a significant quality assurance deficiency was associated with reported result Z Chosen by the tank vapor database (TVD) staff unique to TVD AA Laboratory defined flag. This flag indicates that remarks pertinent to data quality are in the data report narrative or comment. AB Chosen by the tank vapor database (TVD) staff. Unique to TVD.</p>

qualifier_descript	varchar(255)	NO	Definition for the data qualifier.
qualifier_comments	varchar(255)	YES	Basis of qualifier.

dbo.radiological_analysis_results

Contains radiological analysis results for vapor and particulate samples.

I F P	Column name	Data type	Nulls	Default	Description
	result_id	int	NO		Integer ID field in AutoTCR, TCD, and TCD_READ databases. Char(12) ID field created from date and time in Vapor database.
	collection_id	int	YES		Unique identifier created from date and time.
	chemical_id	char(15)	NO		ID of the constituent used for reporting purposes. Typically, the id is the Chemical Abstract Services (CAS) listing.
	chemical_name	varchar(255)	NO		The name of the constituent used for reporting purposes. Typically, the name comes from the Chemical Abstract Services (CAS) listing.
	result_type	char(15)	NO		Description of the type of results the analysis produced. Examples include: PRIMARY, PRIMARY#2
	reported_value	float	YES		Estimated or quantified concentration of a chemical as reported by the analytical lab. Mass per unit volume reported values are at the Reference Temperature and 1.01325 bar pressure.
	reporting_limit	char(20)	YES		A lower limit below which the concentration of a constituent is not reported. The concentration of the constituent either cannot be measured or is below an instrument detection limit or below a contract required quantitation limit. This is a text field and cannot be included in numerical calculations. For more information on reporting limits, see the corresponding references
	MDL	float	YES		Lower method detection limit.
	units	char(10)	NO		Concentration units of the reported value or the reporting limit.
	data_qual_code	char(10)	YES		Single-letter flags that indicate noteworthy aspects of sample result. See Vapor Qualifier Definitions for more information.
	count_error	float	YES		Error in radiological analysis.
	count_error_units	char(10)	YES		Units of "count error".
	count_times	int	YES		Duration of radiological count.
	count_times_units	char(10)	YES		Units of "count times".
	lab_sample_id	varchar(45)	YES		Identifier code assigned to a sample by the laboratory.
	batch_id	char(15)	YES		Identification of a group of analyses that were performed together and having something in common such as time of analysis or instrument.
	analysis_date	datetime	YES		Date of sample analysis. Format is mm/dd/yy.
	lab_received_date	datetime	YES		Date sample was received at laboratory. Format is mm/dd/yy hh:mm.
	analytical_proc	char(15)	NO		Procedure that describe how the results were obtained and analyzed for the sample.
	analysis_method_group	varchar(30)	NO		Analysis Method Group
	lab_abbrev	char(6)	NO		Acronym of laboratory performing analyses.
	reference_id	int	YES		ID field created from date and time in the Vapor database.
	comments_results	varchar(255)	YES		Any concerns or unusual information about the results.
	TPA_Flag	char(3)	NO		Data and information flagged 'TPA' in this database are post-1989 analytical data and are recognized by the Washington State Department of Ecology for use in Tri- Party Agreement (TPA) and regulatory mandated documents. This data may be used for production of TPA deliverable documents.

dbo.radiological_sample_data

Contains information about the conditions under which the sample was collected and the type of sample that was taken.

I	F	P	Column name	Data type	Nulls	Default	Description
			collection_id	int	NO		Unique identifier created from date and time.
			sample_number	varchar(45)	YES		Unique alpha-numeric ID assigned to identify a sample.
			tank_id	char(15)	NO		Code that identifies each tank in a tank farm system.
			sample_description	varchar(50)	YES		Contains descriptive data about each sample consolidated into a single entry.
			tank_headspace_temp	float	YES		Temperature of the headspace.
			headspace_temp_units	char(10)	YES		Headspace temperature units. F = Fahrenheit.
			atmospheric_temp	float	YES		Ambient air temperature.
			atmospheric_temp_units	char(10)	YES		Ambient temperature units. F =Fahrenheit.
			barometric_pressure	float	YES		Barometric pressure (units are torr) measured at the start of the sample collection. When the VSS was used to collect samples (see Sampling Method) measurements were made inside the headspace. When the ISVS or ISS was used to collect samples, ambient pressure at time of sampling was used as reported by the Hanford Meteorological Station.
			sampling_begin_dttm	datetime	NO		Date the sampling started.
			sampling_end_dttm	datetime	NO		Date the sampling was completed.
			total_vol	float	YES		Total sample volume collected.
			units_volume	char(10)	YES		Units for volume.
			device_type	char(25)	YES		Description of the type of container used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
			comments_sampling	varchar(255)	YES		Any concerns or unusual information about collecting the sample (a sample is defined as a discrete quantity of material collected for analysis).

dbo.reference_info

Reference for the analysis results data

I F P	Column name	Data type	Nulls	Default	Description
	reference_id	int	NO		ID field created from date and time in the Vapor database.
	reference_type	varchar(20)	NO		The type of reference.
	access_public	char(1)	NO		The document is available to the public Y or N.
	authors	varchar(255)	NO		Names of authors of report or similar.
	year_published	varchar(4)	YES		The year in which the report or other document was published.
	reference_title	varchar(255)	NO		Title of the reference as it appears on the title page. Subtitles are entered after a colon. If one volume of a set is referenced, then the volume title and volume number should be reported as part of the title
	document_num	varchar(30)	YES		Code assigned by government agencies or government contractors to documents and correspondence they publish; the code is used for tracking purposes. This code includes revision numbers. This code does not include the UC category, which is used by the Office of Scientific and Technical Information.
	publisher	varchar(255)	YES		Publisher of a report or other reference.
	publisher_city	varchar(30)	YES		City part of address of a publisher of a report or other reference.
	publisher_state	varchar(2)	YES		State part of address of a publisher of a report or other reference.
	publisher_country	varchar(30)	YES		Country part of address of publisher of a report or other reference.
	comments_reference	varchar(255)	YES		Comment on reference information in Vapor and Vapor_stage databases.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_prim	reference_title	ASC	Yes	NONCLUSTERED
i_prim	document_num	ASC	Yes	NONCLUSTERED
i_prim	comments_reference	ASC	Yes	NONCLUSTERED
i_reference_info	reference_id	ASC	Yes	CLUSTERED
PK_reference_info__681373AD	reference_id	ASC	Yes	NONCLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
fk_publisher_state	publisher_state	PK_us_states__625A9A57 (dbo.us_states)	
fk_reference_type	reference_type	PK_reference_type__6442E2C9 (dbo.reference_type)	

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.analysis_results	fk_reference_id	i_reference_info

Triggers

Trigger name:	dbo.t_reference_info_del	Created on:	03 Jan 2005
Trigger type:	DELETE	Trigger active:	Yes
Trigger name:	dbo.t_reference_info_ins_upd	Created on:	03 Jan 2005
Trigger type:	INSERT, UPDATE	Trigger active:	Yes

dbo.reference_type

Reference Type

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			reference_type	varchar(20)	NO		The type of reference.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK__reference_type__6442E2C9	reference_type	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.reference_info	fk_reference_type	PK__reference_type__6442E2C9

dbo.result_type

Result type for the analysis

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			result_type	varchar(50)	NO		Description of the type of results the analysis produced. Examples include: PRIMARY, PRIMARY#2

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK__result_type__793DFFAF	result_type	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.analysis_results	fk_result_type	PK__result_type__793DFFAF

dbo.sample_data

Information about the sampling event

I F P	Column name	Data type	Nulls	Default	Description
	collection_id	int	NO		Unique identifier created from date and time.
	field_sample_id	varchar(45)	NO		Code assigned to a sample by the collecting organization (a sample is defined as a discrete quantity of material collected for analysis).
	tank_id	varchar(30)	NO		Code that identifies each tank in a tank farm system.
	sampling_access	char(20)	YES		Sampling port, access, or riser from which the sample was taken (a sample is defined as a discrete quantity of material collected for analysis).
	tank_temp	float	YES		Temperature inside the waste tank when the sample was taken (units are °C) (a sample is defined as a discrete quantity of material collected for analysis).
	barometric_pressure	float	YES		Barometric pressure (units are torr) measured at the start of the sample collection. When the VSS was used to collect samples (see Sampling Method) measurements were made inside the headspace. When the ISVS or ISS was used to collect samples, ambient pressure at time of sampling was used as reported by the Hanford Meteorological Station.
	dtm_sample	datetime	NO		Date and time the sample was taken (a sample is defined as a discrete quantity of material collected for analysis).
	sample_device	varchar(20)	NO		Description of the type of container used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
	sampling_method	varchar(20)	NO		Description of the type of general method used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
	comments_sampling	varchar(255)	YES		Any concerns or unusual information about collecting the sample (a sample is defined as a discrete quantity of material collected for analysis).
	project_type	varchar(40)	NO		The project type of a sample.
	agg_level	varchar(30)	NO		Label or descriptor applied to the whole, i.e. "aggregate", sample before separation into portions or fractions prior to analysis. See /dictionary/AggLevel.asp for more information.
	QA_samp_type	varchar(30)	NO		QA type of sample.
	sample_job_num	varchar(20)	NO		Reference number of sampling job assigned by sample collecting organization. This number is the first 5 characters of the Sample Id.
	work_package	varchar(30)	YES		Work Package Number associated with the sampling event
	load_date_time	datetime	YES	(getdate())	The date and time that the record was loaded into the TCD.
	ih_survey_number	varchar(30)	YES		Industrial Hygiene survey number
	volume_l	float	YES		

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_field_sample_id	field_sample_id	ASC		NONCLUSTERED
i1_sample_data	sampling_access	ASC		NONCLUSTERED
i1_sample_data	dtm_sample	ASC		NONCLUSTERED
i2_sample_data	collection_id	ASC		NONCLUSTERED
i3_sample_data	project_type	ASC		CLUSTERED
i4_sample_data	field_sample_id	ASC		NONCLUSTERED
PK__sample_data__671F4F74	collection_id	ASC	Yes	NONCLUSTERED

Foreign keys

Constraint name	Column name	Reference	Description
fk_qa_samp_type	QA_samp_type	PK__qa_samp_type__1C873BEC (dbo.qa_samp_type)	

fk_tank_id	tank_id	PK__waste_tank__6442E2C9 (dbo.waste_tank)
fk_project_type	project_type	PK__project_type__15DA3E5D (dbo.project_type)
fk_agg_level	agg_level	PK__agg_level__18B6AB08 (dbo.agg_level)
fk_sampling_method	sampling_method	PK__sample_method_de__662B2B 3B (dbo.sample_method_desc)
fk_sample_device	sample_device	PK__sample_device_de__65370702 (dbo.sample_device_desc)

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.analysis_results	fk_collection_id	PK__sample_data__671F4F74

Triggers

Trigger name:	dbo.t_sample_data_del	Created on:	03 Jan 2005
Trigger type:	DELETE	Trigger active:	Yes

Trigger name:	dbo.t_sample_data_ins_upd	Created on:	03 Jan 2005
Trigger type:	INSERT, UPDATE	Trigger active:	Yes

dbo.sample_data_to_load

Sample data awaiting review and approval to be loaded into sample data

I F P	Column name	Data type	Nulls	Default	Description
	sample_data_to_load_id	int	NO		Unique ID for the sample_data_to_load table
	field_sample_id	varchar(255)	YES		Code assigned to a sample by the collecting organization (a sample is defined as a discrete quantity of material collected for analysis).
	tank_id	varchar(255)	YES		Code that identifies each tank in a tank farm system.
	sampling_access	varchar(255)	YES		Sampling port, access, or riser from which the sample was taken (a sample is defined as a discrete quantity of material collected for analysis).
	tank_temp	float	YES		Temperature inside the waste tank when the sample was taken (units are °C) (a sample is defined as a discrete quantity of material collected for analysis).
	barometric_pressure	float	YES		Barometric pressure (units are torr) measured at the start of the sample collection. When the VSS was used to collect samples (see Sampling Method) measurements were made inside the headspace. When the ISVS or ISS was used to collect samples, ambient pressure at time of sampling was used as reported by the Hanford Meteorological Station.
	dtm_sample	datetime	YES		Date and time the sample was taken (a sample is defined as a discrete quantity of material collected for analysis).
	sample_device	varchar(255)	YES		Description of the type of container used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
	sampling_method	varchar(255)	YES		Description of the type of general method used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
	comments_sampling	varchar(255)	YES		Any concerns or unusual information about collecting the sample (a sample is defined as a discrete quantity of material collected for analysis).
	project_type	varchar(255)	YES		The project type of a sample.
	agg_level	varchar(255)	YES		Label or descriptor applied to the whole, i.e. "aggregate", sample before separation into portions or fractions prior to analysis. See "/dictionary/AggLevel.asp" Aggregation Level Definitions for more information.
	QA_samp_type	varchar(255)	YES		QA type of sample.
	sample_job_num	varchar(255)	YES		Reference number of sampling job assigned by sample collecting organization. This number is the first 5 characters of the Sample Id.
	work_package	varchar(255)	YES		Work Package Number associated with the sampling event
	ready_to_load	char(1)	YES		Flag indicating record is ready to be loaded
	review_comment	varchar(6000)	YES		Comment added by reviewer of this analysis result record.
	reference_title	varchar(255)	YES		Title of the reference as it appears on the title page. Subtitles are entered after a colon. If one volume of a set is referenced, then the volume title and volume number should be reported as part of the title
	ih_survey_number	varchar(30)	YES		Industrial Hygiene survey number
	temperature_units	char(1)	YES		Temperature units
	volume_l	float	YES		

Indexes

	Index name	Column name	Sort direction	Is unique	Index type
i_id		sample_data_to_load_id	ASC	Yes	NONCLUSTERED

dbo.sample_device_desc

Sample device description

I	F	P	Column name	Data type	Nulls	Default	Description
			sample_device	varchar(20)	NO		Description of the type of container used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
			sample_device_desc	varchar(255)	NO		Description of a device for collecting samples in the Vapor database.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
PK__sample_device_de__65370702	sample_device	ASC	Yes	NONCLUSTERED

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.sample_data	fk_sample_device	PK__sample_device_de__65370702

dbo.sample_method_desc

Sample method description

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			sampling_method	varchar(20)	NO		Description of the type of general method used to collect a sample (a sample is defined as a discrete quantity of material collected for analysis).
			sample_method_desc	varchar(7000)	NO		Description of the general method used to collect a sample.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK__sample_method_de__662B2B3B	sampling_method	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.sample_data	fk_sampling_method	PK__sample_method_de__662B2B3B

dbo.sample_reference_relation

Cross-reference between sample_data and reference_info

I	F	P	Column name	Data type	Nulls	Default	Description
			sample_reference_relation_id	int	NO		Unique identified for the sample_reference_relation table
			collection_id	int	NO		Unique identifier created from date and time.
			reference_id	int	NO		ID field created from date and time in the Vapor database.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i_sample_ref_relation	collection_id	ASC	Yes	NONCLUSTERED
i_sample_ref_relation	reference_id	ASC	Yes	NONCLUSTERED
i_sample_reference_relation	collection_id	ASC	Yes	CLUSTERED
i_sample_reference_relation	reference_id	ASC	Yes	CLUSTERED
PK__sample_reference__4E53A1AA	collection_id	ASC	Yes	NONCLUSTERED
PK__sample_reference__4E53A1AA	reference_id	ASC	Yes	NONCLUSTERED

Triggers

Trigger name: dbo.t_sample_reference_relation_del
Trigger type: DELETE
Created on: 03 Jan 2005
Trigger active: Yes

Trigger name: dbo.t_sample_reference_relation_ins_upd
Trigger type: INSERT, UPDATE
Created on: 03 Jan 2005
Trigger active: Yes

dbo.SHMS_grab_sample_info

SHMS Sample Data

I F P	Column name	Data type	Nulls	Default	Description
	grab_information_id	char(12)	NO		ID field generated and from date and time in the Vapor database.
	sample_id	varchar(45)	NO		Code assigned to a sample by the collecting organization (a sample is defined as a discrete quantity of material collected for analysis).
	dtm_sample	datetime	NO		Date and time the sample was taken (a sample is defined as a discrete quantity of material collected for analysis).
	analysis_date	datetime	NO		Date of sample analysis. Format is mm/dd/yy.
	login_number	char(10)	YES		Number assigned to a Standard Hydrogen Monitoring Systems sample run recorded in the laboratory record book.
	tank_id	char(15)	NO		Code that identifies each tank in a tank farm system.
	report_date	datetime	YES		Date of report.
	lab_record_book	char(15)	YES		Name or number of the laboratory record book.
	page_num	char(15)	YES		Page number or range of pages where the information was cited
	comments_SHMS_info	varchar(255)	YES		Any concerns or unusual information about the SHMS sample.
	analytical_procedure	char(20)	YES		Code assigned by the laboratory to analytical procedures they use in analytical work. This code includes revision numbers.
	lab_abbrev	char(6)	NO		Acronym of laboratory performing analyses.
	analysis_method	varchar(50)	NO		Primary instrumentation used to analyze a sample (a sample is defined as a discrete quantity of material collected for analysis).

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i1_SHMS_grab_sample_info	dtm_sample	ASC		CLUSTERED
i1_SHMS_grab_sample_info	tank_id	ASC		CLUSTERED
i2_SHMS_grab_sample_info	grab_information_id	ASC		NONCLUSTERED

Triggers

Trigger name: dbo.t_SHMS_grab_samp_info_ins_upd
Trigger type: INSERT, UPDATE

Created on: 27 Apr 2000
Trigger active: Yes

dbo.SHMS_grab_sample_results

SHMS Results

I F P	Column name	Data type	Nulls	Default	Description
	grab_results_id	char(12)	NO		ID value created from date and time in the Vapor database.
	grab_information_id	char(12)	NO		ID field generated and from date and time in the Vapor database.
	chemical_id	char(15)	NO		ID of the constituent used for reporting purposes. Typically, the id is the Chemical Abstract Services (CAS) listing.
	reported_value	float	YES		Estimated or quantified concentration of a chemical as reported by the analytical lab. Mass per unit volume reported values are at the Reference Temperature and 1.01325 bar pressure.
	reporting_limit	char(20)	YES		A lower limit below which the concentration of a constituent is not reported. The concentration of the constituent either cannot be measured or is below an instrument detection limit or below a contract required quantitation limit. This is a text field and cannot be included in numerical calculations. For more information on reporting limits, see the corresponding references
	units	char(10)	NO		Concentration units of the reported value or the reporting limit.
	conc_ppmv	char(20)	YES		The estimated or quantified concentration of a chemical in units of ppmv.
	data_qual_code	char(10)	YES		Single-letter flags that indicate noteworthy aspects of sample result. See /dictionary/VaporQualifiers.asp >Vapor Qualifier Definitions for more information.
	comments_SHMS_results	varchar(255)	YES		Any concerns or unusual information about the SHMS results.
	conditions	varchar(30)	YES		Tank conditions other than regular SHMS such as Gas Release Event.

Indexes

Index name	Column name	Sort direction	Is unique	Index type
i1_SHMS_grab_sample_results	grab_information_id	ASC		CLUSTERED
i2_SHMS_grab_sample_results	grab_results_id	ASC		NONCLUSTERED
i2_SHMS_grab_sample_results	grab_information_id	ASC		NONCLUSTERED
i3_SHMS_grab_sample_results	chemical_id	ASC		NONCLUSTERED

dbo.tankfarm_use

List of tankfarms

<u>I F P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
	tankfarm	char(2)	NO		The letters identifying a tank farm such as 'SY' or 'U'.
	SHMS_composite	char(1)	YES		Is this a Standard Hydrogen Monitoring System composite?

dbo.units

Units

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			units	varchar(10)	NO		Concentration units of the reported value or the reporting limit.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK__units__634EBE90	units	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.analysis_results	fk_units	PK__units__634EBE90

dbo.us_states

US States

<u>I</u>	<u>F</u>	<u>P</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
			state_abbrev	varchar(2)	NO		US state postal abbreviation
			state_name	varchar(30)	NO		The non-abbreviated name of a state such as 'Texas'.

Indexes

<u>Index name</u>	<u>Column name</u>	<u>Sort direction</u>	<u>Is unique</u>	<u>Index type</u>
PK__us_states__625A9A57	state_abbrev	ASC	Yes	NONCLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	<u>Primary key or unique constraint</u>
dbo.laboratory_desc	fk_lab_state	PK__us_states__625A9A57
dbo.reference_info	fk_publisher_state	PK__us_states__625A9A57

dbo.waste_tank

Characteristics of the Hanford underground waste tanks. Note that the Tank table in the BBIM database also contains a similar set of these characteristics. However, the bbim..Tank table contains information on only the 177 HLW tanks and a few DCRT tanks while the tcd..Waste_Tank table contains information on all Hanford tanks.

I	F	P	Column name	Data type	Nulls	Default	Description
			tank_id	varchar(30)	NO		Code that identifies each tank in a tank farm system.
			tank_type	varchar(5)	YES		Type of Tank

Indexes

Index name	Column name	Sort direction	Is unique	Index type
PK__waste_tank__6442E2C9	tank_id	ASC	Yes	NONCLUSTERED

Referencing tables

Table name	Foreign key	Primary key or unique constraint
dbo.sample_data	fk_tank_id	PK__waste_tank__6442E2C9