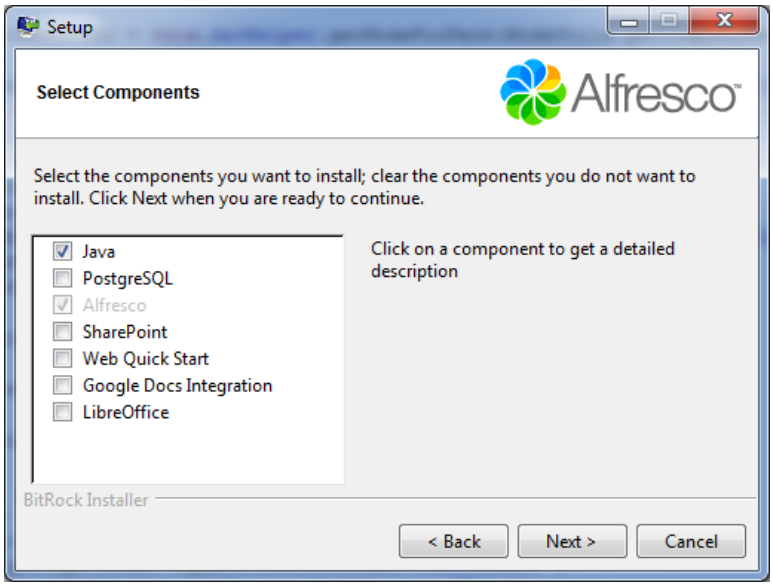
Velo Server Installation Guide

1) **Install Alfresco.** Run alfresco 4.2.c installer. Select the "advanced" install so you can tell it to not install postgres. If you select the "easy" install, it will also install postgres db as a service, but that's ok. Service start is manual, so we don't need to use it. Uncheck all components except Java (default installation is to use the alfresco bundled java). Remove share.war or rename it to share.war.bak so that tomcat will start up faster. Also remove alfresco.war – it will be replaced with a custom war file later. At the very end of the installer, it'll ask if you want to start "Alfresco"/"Share" – select NO. Download the alfresco installer from here: https://wiki.alfresco.com/wiki/Community_file_list_4.2.c



2) **Install MySQL 5.5.** Alfresco 4.2.c is "blessed" with MySQL 5.5. It appears that Alfresco 4.2.c works fine with the older 5.1 version (so this is probably fine for dev machines), but for production servers, it would be best to use the recommended version. Installers can be found in http://dev.mysql.com/downloads/mysql/5.5.html

3) Create Alfresco database and give alfresco database user permissions where $ALFRESCO_DB_PASSWORD below is the value in the db.password property in $ALF_HOME/tomcat/shared/classes/alfresco-global.properties:

```
mysql -u root -p
create database alfresco;
grant all on alfresco.* to 'alfresco'@'localhost' identified by
'$ALFRESCO_DB_PASSWORD' with grant option;
grant all on alfresco.* to 'alfresco'@'localhost.localdomain' identified by
'$ALFRESCO_DB_PASSWORD' with grant option;
```

4) **Backup SOLR.** Copy $ALF_HOME/alf_data/solr folder to solr_bak or something (so we can recover a pristine copy if needed – I learned the hard way)

5) **Configure alfresco-global.properties.** Due to changes in property scope in the way subsystem properties loaded, it's cleanest to set all of our custom alfresco config properties in one place. Add/override these properties in $ALF_HOME/tomcat/shared/classes/alfresco-global.properties. Make sure the values match your deployment.

```
### Solr indexing ###
index.subsystem.name=solr
dir.keystore=${dir.root}/keystore
solr.host=localhost
solr.port=8080
solr.port.ssl=9443

### database connection properties ###
db.name=alfresco
db.driver=org.gjt.mm.mysql.Driver
db.url=jdbc:mysql://localhost/alfresco
hibernate.dialect=org.hibernate.dialect.MySQLInnoDBDialect

#
# Transaction retry parameters used by
# Alfresco's RetryingTransactionHelper
# We overload the default # of retries because
# it's now set to 40!  That is reallllllly long -
# we want something smaller.
#
server.transaction.max-retries=10

# email properties
# No SMTP auth required
mail.protocol=smtp
mail.host=smtpmail1.pnl.gov
mail.port=25
mail.encoding=UTF-8
mail.smtp.auth=false
mail.from.default=cats.meow@pnl.gov
mail.smtp.debug=false
mail.smtp.timeout=5000
mail.smtp.starttls.enable=false
mail.smtps.auth=false
mail.smtps.starttls.enable=false

# ECM content usages/quotas
system.usages.enabled=false
system.spaces.usages.enabled=false

#
# Determine if modification timestamp propagation from child to parent nodes is
respected or not.
# Even if 'true', the functionality is only supported for child associations
that declare the
# 'propagateTimestamps' element in the dictionary definition.
# Velo is setting this to FALSE because we were getting all kinds of
concurrency problems with folders
# because of this
```

```
system.enableTimestampPropagation=false
```

6) **Add MySQL JDBC Driver**. Download the mysql connector j jar file ("mysql-connector-java-5.1.22-bin.jar") from http://dev.mysql.com/downloads/connector/j/ and copy to $ALF_HOME/tomcat/lib

7) **Create SOLR Tomcat Server.** Install solr in separate tomcat (default is to run it in same tomcat as alfresco, but that causes sync problems with Velo bootstrap code and also it is bad for production). Download a Tomcat installer from here: https://tomcat.apache.org/download-70.cgi

   a. **Move solr.xml file.** Move $ALF_HOME/tomcat/conf/Catalina/localhost/solr.xml to $SOLR_TOMCAT/conf/Catalina/localhost
   b. **Change ports.** Change ports in SOLR tomcat server.xml file so there are no clashes with alfresco tomcat. Also, you MUST define an SSL connector pointing to the keystore files that reside in $ALF_HOME/alf_data/keystore. Basically you can copy the connector from the alfresco server.xml file except change the port.

```
<Connector port="9443" protocol="org.apache.coyote.http11.Http11Protocol"
SSLEnabled="true" maxThreads="150" scheme="https"
keystoreFile="C:\Alfresco\alfresco-community-
4.2.c/alf_data/keystore/ssl.keystore"
keystorePass="kT9X6oe68t" keystoreType="JCEKS"
 secure="true" connectionTimeout="240000"
truststoreFile="C:\Alfresco\alfresco-community-
4.2.c/alf_data/keystore/ssl.truststore"
truststorePass="kT9X6oe68t" truststoreType="JCEKS"
clientAuth="false" sslProtocol="TLS"
allowUnsafeLegacyRenegotiation="true" maxSavePostSize="-1" />
```

   c. **Update alfresco-global.properties.** Solr properties need to point to SOLR TOMCAT:

```
solr.port=8080
solr.port.ssl=9443
```

   d. **Update solrcore.properties.** Edit $ALF_HOME/alf_data/solr/workspace-SpacesStore/conf/solrcore.properties and $ALF_HOME/alf_data/solr/archive-SpacesStore/conf/solrcore.properties (*NOTE there are two solrcore.properties, one for each store*) to make sure the alfresco port points to the right alfresco server:

```
alfresco.host=localhost
alfresco.port=8082
alfresco.port.ssl=8443
alfresco.baseUrl=/alfresco
```

   e. IN $solr_tomcat/bin create a new file named setenv.sh. Set the contents of the file to the following, replacing $ALF_HOME with opt/alfresco-4.2.c

```
export JAVA_HOME=$ALF_HOME/java
export JAVA_OPTS="$JAVA_OPTS -Xms4G -Xmx8G -XX:NewSize=3G
-XX:MaxPermSize=256M -XX:+UseConcMarkSweepGC
-XX:+CMSIncrementalMode -XX:CMSInitiatingOccupancyFraction=80"
```

f. **Update tomcat-users.xml.** Modify solr-tomcat/conf/tomcat-users.xml to add this line

<user username="CN=Alfresco Repository, OU=Unknown, O=Alfresco Software Ltd., L=Maidenhead, ST=UK, C=GB" roles="repository" password="null"/>

8) **Start SOLR Server.** SOLR server must always be started before Alfresco. It will continuously poll for updates from alfresco. Before alfresco starts up, you will see connection refused errors in the solr log, but that's ok. As soon as alfresco starts, it will be fine.

9) **Deploy CAT and Velo.**

a. Manual deploy: Download the catAlfresco.war file from <public svn mirror URL to catAlfresco.war> , unwar the war to $ALF_HOME/tomcat/webapps/. You should now have $ALF_HOME/tomcat/webapps/alfresco
b. Build and Deploy from source: Check out in the same directory <svn root> the following folders (each are eclipse projects):
   i. CATClientWebServices
   ii. CATServerCommon
   iii. CATServerCore
   iv. CATServerWebServices
   v. gov.pnnl.velo.commons
   vi. SDKAlfrescoEmbedded
   vii. SDKAlfrescoRemote
   viii. velo
   ix. VeloBuild
   x. VeloDeploy

   Create a build.<username>.properties file under VeloBuild, start by copying another build.*.properties file and modifying the values to use paths on your maching

   Run the default ant target of VeloBuild/build.xml ("deploy-cat-server-and-modules-incremental")

10) **Install Activemq (5.9.0 – can be downloaded from http://activemq.apache.org/activemq-590-release.html) and have it run as a service. Once its installed, copy cat-core.jar from $ALF_HOME/tomcat/webapps/alfresco/WEB-INF/lib to activemq's lib folder. Restart the service.**

11) **Set JVM Arguments for Alfresco Tomcat.** You can set these in $ALF_HOME/tomcat/bin/setenv.sh(or .bat). This is the bare minimum. You only need to set the jmx properties if you want to profile your server (recommended only on dev servers, NOT production). Replace the line → JAVA_OPTS="$JAVA_OPTS" with the following code:

Make sure this is implemented as one line of code.

```
JAVA_OPTS="$JAVA_OPTS -Xms1G -Xmx2G -XX:MaxPermSize=256m -XX:NewSize=512m -
XX:+UseConcMarkSweepGC -XX:CMSInitiatingOccupancyFraction=80 -
XX:+CMSIncrementalMode -Dcom.sun.management.jmxremote -
Dcom.sun.management.jmxremote.port=8999 -
Dcom.sun.management.jmxremote.authenticate='false' -
Dcom.sun.management.jmxremote.ssl='false'"
```

12) **Start Alfresco Server.** The bundled Alfresco tomcat is now Tomcat 7, and the bundled Alfresco Java is Java 7. **Alfresco 4 will not run with Java 6.** If you are starting it from Eclipse, you need to get the latest version of the Sysdeo Tomcat plugin that supports Tomcat 7. Download the Sysdeo Eclipse Tomcat Launcher plugin from here http://www.eclipsetotale.com/tomcatPlugin.html and copy it into your eclipse dropins folder. You will need to configure the tomcat plugin to run with Java 7, so you need to import the Java 7 JDK from $ALF_HOME/java. You will also need to set the JVM args defined above.