Apelon DTS

Distributed TerminologySystem

DTS 3.1 Quick Start Guide

Table of Contents

Install Apelon DTS **3**

Create DTS 3.1 User **4**

Create Database User and Tablespace (Oracle) 4

Create SQL Server Database File (Microsoft SQL Server) 5

Create and Prepare the Knowledgebase **6**

Create DTS 3.1 Knowledgebase 6

Populate DTS 3.1 Knowledgebase Tables 7

Populate New Tables Through “Full” Subscription Import 7

Populate New Tables Through Data Migration 9

Run Knowledgebase Admin 12

Configure Apelon DTS Server **15**

Start Servers and DTS Browser **18**

Start Apelon DTS Server 18

Start Tomcat Server 18

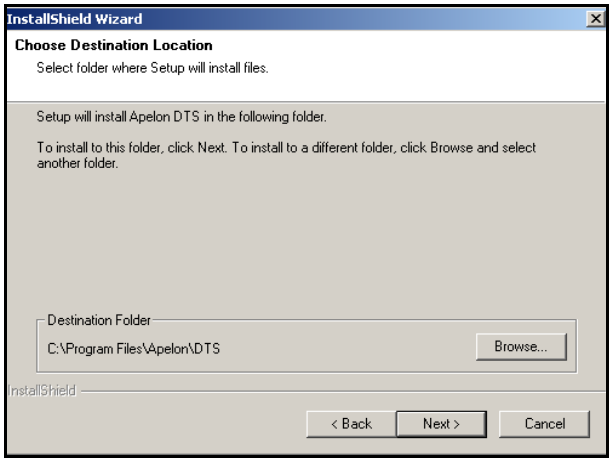
Start Apelon DTS Browser 19

Copyright © 1999-2004 Apelon, Inc. All Rights Reserved.

Install Apelon DTS

Refer to the *DTS Installation Guide* for more information on installing DTS 3.1.

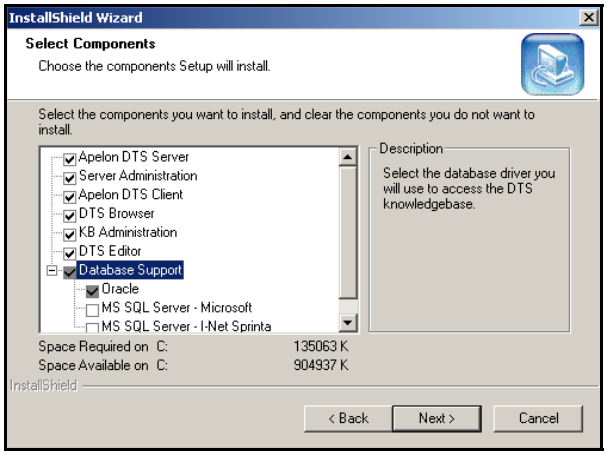
1. Execute the Apelon DTS 3.1 exe.
2. When the InstallShield Wizard *Welcome Screen* displays, click **Next** to continue.
3. When the License Agreement displays, click **Yes** to accept the stated terms.
4. When the DTS 3.1 **Readme** text displays, read the text and click **Next** to continue.
5. When the *Destination Location* window displays, click **Browse** to select the installation location, then click **Next**.



6. On the *Component Selection* window, choose which of the Apelon DTS components (**Apelon DTS Server**, **Server Administration**, **Apelon DTS Client**, **DTS Browser**, **KB Administration**, and **DTS Editor**) you want installed. The default is to install all of the DTS components; click the corresponding checkbox to deselect undesired components.

   You can install the DTS components in various configurations, on the same machine, or each on separate machines (as long as all the machines containing the DTS components are connected by a network). The default is to install all six components on the machine on which the installer is run.

Copyright © 1999-2004 Apelon, Inc. All Rights Reserved.



7. You then must specify the database on which you will maintain your DTS knowledgebase, and the driver through which the DTS Server and DTS Editor will access the data. The appropriate DTS configuration files are updated to reflect this selection.

   - **Oracle** (the default) applies if you have installed either **Oracle 8i** or **Oracle 9i** to maintain your knowledgebase.
   - Click **MS SQL Server – Microsoft** if you will maintain the knowledgebase in a *Microsoft* SQL 2000 Server database, and use the *Microsoft* JDBC driver.
     - A **SQL Server 2000 Type 4 JDBC** driver is required.
     - The default *Microsoft* driver is installed automatically.
   - Click **MS SQL Server – I-Net Sprinta** if you will maintain the knowledgebase in a *Microsoft* SQL 2000 Server database, and use the I-Net Sprinta JDBC driver.

8. Click **Next** to begin installing the selected components in the location you specified. When the *Install Complete* window displays, click **Finish**.

Create DTS 3.1 User

Create Database User and Tablespace (Oracle)

1. If Oracle is to be used, you must create a user and define a tablespace before the DTS 3.1 tables can be created. Navigate to the **DTS\scripts\Oracle** directory.

2. Locate the file **userCreate.sql**., and open it (the file is illustrated).

Copyright © 1999-2004 Apelon, Inc. All Rights Reserved.

```
SPOOL userCreate.log

DROP USER dts3 CASCADE;

DROP TABLESPACE dts3 INCLUDING CONTENTS;

CREATE TABLESPACE dts3
  DATAFILE 'd:\Oracle\oradata\<host>\dts3.dbf'
  --REUSE
  SIZE 25M
  --AUTOEXTEND OFF
 AUTOEXTEND ON NEXT 100M MAXSIZE 10000M;

CREATE USER dts3 IDENTIFIED BY dts3
  DEFAULT TABLESPACE dts3;

GRANT connect TO dts3;
GRANT resource TO dts3;
GRANT dba TO dts3;

REVOKE UNLIMITED TABLESPACE FROM dts3;

--ALTER USER dts3 QUOTA 0 ON USER_DATA;

ALTER USER dts3 QUOTA 10000M ON dts3;

QUIT;
```

Note the highlighted line.

DATAFILE 'd:\Oracle\oradata\<host>\dts3.dbf'

3. Change the "**d:**" to the letter of the local drive where your Oracle databases are installed, then change **<host>** to the name of the subdirectory where your Oracle data files are located. Note the following example.

   DATAFILE '**{local drive}:\Oracle\oradata\{host name}\dts3.dbf**'

4. Save the **userCreate.sql** file.
5. Run the file **userCreate.bat**. This will create a user named **dts3** and the user data file **dts3.dbf** in the (DATA) subdirectory where your Oracle data files are located.

Create SQL Server Database File (Microsoft SQL Server)

1. If Microsoft SQL Server is to be used, you must make the following edits in the **drivers.xml** file (**DTS\bin**):
   - Uncomment the **driver_count** property and set the value to **2**.
   - Uncomment the **display_name** value and set the value to **2**.
2. A user and database must be defined and created before the knowledgebase can be created. Navigate to the **DTS\scripts\SQL2K** directory. Locate the file **userCreate.sql**., and open it (a portion of the file is illustrated).

Copyright © 1999-2004 Apelon, Inc. All Rights Reserved.

```
IF EXISTS (SELECT name FROM master.dbo.sysdatabases WHERE name = N'dts30')

       DROP DATABASE [dts30]

GO

CREATE DATABASE [dts30]  ON (NAME = N'dts30_Data', FILENAME = N'd:\Program
Files\Microsoft SQL Server\MSSQL\data\dts30_Data.MDF' , SIZE = 2,
FILEGROWTH = 10%) LOG ON (NAME = N'dts30_Log', FILENAME = N'd:\Program
Files\Microsoft SQL Server\MSSQL\data\dts30_Log.LDF' , SIZE = 1,
FILEGROWTH = 10%)

 COLLATE SQL_Latin1_General_CP1_CI_AS

GO

exec sp_dboption N'dts30', N'autoclose', N'false'

GO
```

Note the highlighted section.

d:\Program Files\Microsoft SQL Server\MSSQL\data\dts30

3. Change the path to reflect the location where your SQL Server databases are installed.
4. Save the **userCreate.sql** file.
5. Run the file **userCreate.bat**. This will create a user named **dts3** and the user data file **dts3.dbf** in the DATA directory.

Create and Prepare the Knowledgebase

Create DTS 3.1 Knowledgebase

Before you run the **Knowledgebase Create** utility to create the DTS 3.1 schema, you must follow this procedure to configure the parameters by which the schema will be created, and by which a connection will be made to the DTS 3.1 database.

1. Open the **kbcreate.xml** file (**DTS\bin**) where you set the values for **either** an Oracle **or** a SQL 2000 connection. Note the illustration.

Copyright © 1999-2004 Apelon, Inc. All Rights Reserved.

```
<?xml version="1.0" ?>

<!DOCTYPE DBConfig SYSTEM "http://apelon.com/dtd/util/db/dbconfig.dtd">
 <DBConfig>

<!-- ORACLE CONNECTION -->
        <connection>
                <property name="type" value="oracle" />
                <property name="user" value="dts3" />
                <property name="pass" value="dts3" />
                <property name="host" value="localhost" />
                <property name="databaseName" value="ORCL" />
                <property name="databasePort" value="1521" />
                <property name="jdbcDriver" value="oracle.jdbc.driver.OracleDriver"
/>
                <property name="url_template"
value="jdbc:oracle:thin:@[HOST]:[PORT]:[DATABASE]" />
        </connection>

<!-- SQL2000 CONNECTION -->
<!--
        <connection>
                <property name="direction" value="target" />
                <property name="type" value="sql2k" />
                <property name="user" value="sa" />
                <property name="pass" value="" />
                <property name="host" value="localhost" />
                <property name="databaseName" value="dts30" />
                <property name="databasePort" value="1433" />
                <property name="jdbcDriver" value="com.inet.tds.TdsDriver" />
                <property name="url_template"
value="jdbc:inetdae:[HOST]:[PORT]?database=[DATABASE]" />
        </connection>
-->
</DBConfig>
```

2. Edit and save the file according to the procedure in the *Create DTS Knowledgebase Schema* discussion in the *Knowledgebase Administrators Guide*.

3. Run the file **kbcreate.bat** (**DTS\bin**) to start the Knowledgebase Create utility.

Populate DTS 3.1 Knowledgebase Tables

After you create the DTS 3.1 tables using the Knowledgebase Create utility, you can populate the new tables with data. If you intend to populate your DTS 3.1 knowledgebase with new subscription data, refer to the *Populate New Tables Through "Full" Subscription Import* discussion. If you intend to populate the tables with migrated data from your existing knowledgebase, refer to the *Populate New Tables Through Data Migration* discussion.

Populate New Tables Through "Full" Subscription Import

The **Client Import** utility allows you to populate your DTS 3.1 tables through a **full import** of initial subscription data. Apelon exports initial subscription data to you in a .zip file that includes data from all of the vocabularies to which you subscribe.

1. Unzip the **SOURCE-VERSION-FULL.zip** import subscription data file provided by Apelon into the **DTS/bin** directory. A **data** subdirectory is created in

Copyright © 1999-2004 Apelon, Inc. All Rights Reserved.

DTS/bin.
Under the new **data** directory, a separate subdirectory is created for each vocabulary source (i.e., namespace) in your knowledgebase. Note the illustration.

2. Open the **kbcontent-import.xml** file (**DTS\bin**) and specify the database connection parameters. A portion of the **kbcontent-import.xml** file is illustrated.

```
<!--
        ##################################
        #  Client Database Connection    #
        ##################################
      -->

      <!-- Oracle Connection Parameters -->
      <connection>
            <property name="direction" value="target" />
            <property name="type" value="oracle" />
            <property name="user" value="dts3" />
            <property name="pass" value="dts3" />
            <property name="host" value="localhost" />
            <property name="databaseName" value="ORCL" />
            <property name="databasePort" value="1521" />
            <property name="jdbcDriver" value="oracle.jdbc.driver.OracleDriver" />
            <property name="url_template" value="jdbc:oracle:thin:@[HOST]:[PORT]:[DATABASE]"
/>
            <property name="blockSize" value="512" />
      </connection>

            <!--
             ###################################
             #  Database Connection (sql2k)    #
             ###################################
            -->
            <!-- Microsoft Driver -->
            <!--
            <connection>
                        <property name="direction" value="target" />
              <property name="type" value="sql2k" />
              <property name="user" value="sa" />
              <property name="pass" value="" />
              <property name="host" value="localhost" />
              <property name="databaseName" value="dts3" />
              <property name="databasePort" value="1433" />
              <property name="jdbcDriver" value="com.microsoft.jdbc.sqlserver.SQLServerDriver" />
              <property name="url_template"
value="jdbc:microsoft:sqlserver://[HOST]:[PORT];databasename=[DATABASE]" />
   </connection>
```

Connection parameters for **SQL2000** (for both the *Microsoft* Driver and the I-Net Sprinta Driver) are commented out as the default. For a **SQL2000** connection,

Copyright © 1999-2004 Apelon, Inc. All Rights Reserved.

uncomment out the appropriate **SQL2000 CONNECTION PARAMETERS** section, then comment out the **ORACLE CONNECTION** section.

3. The highlighted values in the **kbcontent-import.xml** file will vary depending on the content and version being imported. These values are supplied by Apelon.

```
<!-- Content Type to import (full or diff) -->
        <property name="contentType" value="full" />

        <!-- Namespace(s) to import (Client KB) -->
        <namespace name="ICD-9-CM" >
                <property name="add" value="true" />
        </namespace>

        <!-- New Version In (present in CDF) -->
        <property name="newVersionIn" value="1.0.0.0" />

        <!-- Version Out (Client KB) -->
        <property name="versionOut" value="NOT_RETIRED" />
```

4. Run the **kbcontent-import.bat** process to import initial subscription content into your knowledgebase. Each vocabulary source coded in the **kbcontent-import.xml** configuration file is loaded with data from its corresponding subscription vocabulary subdirectory under **DTS\bin\data**.

Populate New Tables Through Data Migration

If you intend to populate your new DTS 3.1 tables with migrated data, you must run the **Knowledgebase Load** utility to migrate knowledgebase data from DTS 2.x or TDE data sources. The procedures referenced in this discussion represent a "**standard migration**" of TDE or DTS 2.x data into DTS 3.1.

1. Each TDE or DTS 2.x source vocabulary you wish to migrate to DTS 3.1 must be listed in the **sources.id** file in the **DTS\bin** directory. A portion of the file that is included with the DTS 3.1 installation is illustrated.

```
######################
##                  ##
## Ontylog Sources ##
##                  ##
######################

### ICD (International Classification of Diseases, Ninth Revision,
Clinical Modification)
ICD-9-CM=10

### CPT (Physicians' Current Procedural Terminology)
CPT=20

### SNOMEDCT
SNOMED CT=30

### SNOMED Historical Relationships (Inactives)
SNOMED CT Inactive=35
```

If you want to migrate a local source vocabulary that is not listed in the **sources.id**

Copyright © 1999-2004 Apelon, Inc. All Rights Reserved.

file, you must add the source vocabulary with a unique ID number. The starting ID number for assignment is listed near the end of the file (e.g., **value >= 32768**).

2. To configure the connection between the source database and target database (i.e., the load destination) open the **kbload.xml** file (**DTS\bin**).

```
<?xml version="1.0" ?>

<!DOCTYPE DBConfig SYSTEM "http://apelon.com/dtd/util/db/dbconfig.dtd">
 <DBConfig>
        <!-- The authority for the namespace -->
        <property name="authority" value="Apelon3" />

        <!-- The version label for content -->
        <property name="version" value="1.0.0.0" />

        <!-- The version number for content (numerals only) -->
        <property name="versionNumber" value="1" />

        <!-- The content type to be loaded ('ontylog' or 'thesaurus') -->
        <property name="conceptType" value="ontylog" />

        <!-- Access mode for the constent ('readwrite' or 'readonly') -->
        <property name="accessMode" value="readonly" />

        <!-- Namespace type ('local' or 'nonlocal' )-->
        <property name="namespaceType" value="nonlocal" />

        <!-- Namespace has semantic type properties ('true' or 'false') -->
        <property name="hasSemanticType" value="false" />

        <!-- Migrating from database -->
        <connection>
                <property name="direction" value="source" />
                <property name="type" value="oracle" />
                <property name="user" value="icd" />
                <property name="pass" value="icd" />
                <property name="host" value="localhost" />
                <property name="databaseName" value="ORCL" />
                <property name="databasePort" value="1521" />
                <property name="jdbcDriver" value="oracle.jdbc.driver.OracleDriver"
/>
                                <property name="url_template"
value="jdbc:oracle:thin:@[HOST]:[PORT]:[DATABASE]" />
                <property name="blockSize" value="256" />
        </connection>
        <!-- Migrating to database (DTS30) -->
        <connection>
        <property name="direction" value="target" />
                <property name="type" value="oracle" />
                <property name="user" value="dts3" />
                <property name="pass" value="dts3" />
                <property name="host" value="localhost" />
                <property name="databaseName" value="ORCL" />
                <property name="databasePort" value="1521" />
                <property name="jdbcDriver" value="oracle.jdbc.driver.OracleDriver"
/>
                <property name="url_template"
value="jdbc:oracle:thin:@[HOST]:[PORT]:[DATABASE]" />
                <property name="blockSize" value="256" />
        </connection>
</DBConfig>
```

Copyright © 1999-2004 Apelon, Inc. All Rights Reserved.

Note the following property values in the file:

- The value for the property **authority** is the authority name for this source.
- The value for the property **version** is the version name (i.e., label) for this vocabulary; the initial (full) vocabulary version for a first-time migration may be, for example, **1.0.0.0.** You can replace this value with any alpha/numeric version identifier.
- The value for the **versionNumber** property must increase for each subsequent vocabulary load (e.g., start with **1,** followed by **2**, **3**, etc.).
- The value for the property **conceptType** is **ontylog** for TDE and DTS 2.x data migrations.
- The value for the property **accessMode** is **readonly** for loading standard terminologies, and **readwrite** for local namespaces to make them editable in DTS.

  > Apelon suggests that you set the **accessMode** to **readonly** for the load of a TDE (i.e., ontylog) terminology, and then use the DTS Editor to create a separate (editable) local namespace to which new and modified content can be written. This approach prevents the inadvertent overwrite of any new local data (entered using the DTS Editor) if updated TDE content is migrated into the DTS 3.1 knowledgebase at a later time.

- The value for the property **namespaceType** is **nonlocal** for subscription vocabularies; for **non-subscription vocabularies**, or for custom TDE migrations, the value is **local**.
- The value for the property **hasSemanticType** is **true** for UMLS sources only.

3. In the database **source** parameters in the upper portion of the file, modify the source property values (e.g., **user**, **pass**, **host**, etc.) for either the DTS 2.x or TDE knowledgebase, as necessary.
4. In the database **target** parameters in the lower portion of the file, modify the target property values (e.g., **user**, **pass**, **host**, etc.) for the DTS 3.1 knowledgebase, as necessary.
5. Save the **kbload.xml** file.

Copyright © 1999-2004 Apelon, Inc. All Rights Reserved.

6. Run the file **kbload.bat** (**DTS\bin**) to start the Knowledgebase Load utility. The utility opens the source and target database connections specified in the **kbload.xml** file, and migrates data from the source knowledgebase to the target knowledgebase.

Run Knowledgebase Admin

If you are migrating local namespace content from a DTS 2.x or TDE database to DTS 3.1 using the **Knowledgebase Load** utility (see *Knowledgebase Load Procedures*) you must run the Knowledgebase Admin utility to prepare the namespace in the DTS 3.1 knowledgebase for access by the DTS server and clients. If you imported **only** subscription namespace content by performing the full subscription load procedure (see *Full Subscription Load Procedures*) there is no need to run the Knowledgebase Admin utility.

The Knowledgebase Admin utility also allows you to create synonymous terms, and generate mappings between concepts with corresponding property values in any two namespaces. Mappings (associations) are generated based on the configuration you specify.

1. In order to run the Knowledgebase Admin utility, you must establish the connection against which the utility will run in the **kbadmin.xml** file. A sample **kbadmin.xml** file is provided in **DTS\bin** as a template for the creation of synonymous terms and associations. Note the **kbadmin.xml** file illustration.

   This file needs to be edited to run against **local** namespaces. The process should not be run against **subscription** namespaces (it has been run already by Apelon).

Copyright © 1999-2004 Apelon, Inc. All Rights Reserved.

```
<?xml version="1.0" ?>

<!DOCTYPE DTSDbConfig SYSTEM "http://apelon.com/dtd/dts/dtsconfig.dtd">

<DTSDbConfig>
        <!--
                This is an example configuration file illustrating
                the various parameters needed to run KBAdmin process.
                Users are required to modify it according to their needs
                and run KBAdmin process.
                Note: The kbadmin should be run only on a local or
                        non-subscription source
                        (e.g., on a local namespace migrated from TDE)
        -->
        <!--
            Specifiy the connection paramters to connect
            to a DTS30 knowledgebase.
        -->
        <connection>
                <property name="type" value="oracle" />
                <property name="user" value="dts3" />
                <property name="pass" value="dts3" />
                <property name="host" value="localhost" />
                <property name="databaseName" value="ORCL" />
                <property name="databasePort" value="1521" />
                <property name="jdbcDriver" value="oracle.jdbc.driver.OracleDriver" />
                <property name="url_template" value="jdbc:oracle:thin:@[HOST]:[PORT]:[DATABASE]" />
        </connection>
        <!--
            This is the name of the namespace for which
            synonyms terms and/or associations with other
            namespace are to be created.
        -->
        <property name="namespace" value="ICD-9-CM" />

        <!-- Build synonymous term from a given property

           - The first property element name attribute has the
             property defintion name which is to be converted
             to a synonymous term. The "value" field attribute
             states whether this synonymous term is a preferred
             term or not ('true' for preferred, 'false' if not).

          - The second property element allows user to
            add a property to the synonymous term created above.
             The value of the term property is specified in the
             'value' attribute of this element.

            Example below attaches a 'Term Type' property
            with a value 'Preferred Term' to all the
            synonymous terms created from the property
            'Preferred Name' present in ICD-9-CM namespace.

            Note:
            - For a given namespace, only one type
              of synonymous term can be a preferred term.
            - Once the property is translated to synonymous
              term corresponding property type definition
              and its instances are deleted from the namespace.
        -->
<!-- Example to create synonymous term for ICD-9-CM -->
        <!--
        <synonym nameSpace="ICD-9-CM" >
                <property name="Preferred Name" value="true" />
                <property name="Term Type" value="Preferred Term" />
        </synonym>
```

Synonymous Terms

Copyright © 1999-2004 Apelon, Inc. All Rights Reserved.

```
                <synonym nameSpace="ICD-9-CM">
                <property name="Entry Term" value="false" />
                <property name="Term Type" value="Entry Term" />
        </synonym>
        -->

        <!-- Example to create synonymous term for CPT -->
        <!--
        <synonym nameSpace="CPT" >
                <property name="Preferred Name" value="true" />
                <property name="Term Type" value="Long Descriptor" />
        </synonym>
        <synonym nameSpace="CPT" >
                <property name="Synonym" value="false" />
                <property name="Term Type" value="Medium Descriptor" />
        </synonym>
        -->
        <!-- Build associations (or mappings) between two concepts
           having the same property values in two different namespaces.

           - Mapping element attribute 'sourceNameSpace' specifies
             the source namespace name.

           - Mapping element attribute 'targetNameSpace' specifies
             the target namespace name.

           - Property element 'SOURCE_PROPERTY_NAME' is the property
             definition used to create the source end of association.

           - Property element 'SOURCE_PROPERTY_NAME' is the property
             definition used to create the source end of association.

           - Property element 'DELETE_SOURCE_PROPERTY_POST_MAP' deletes
             instances of any property in source namespace which are
             used to create associations if the value is set to 'true'
             or if the value is set to 'false' does not delete them.

           - Property element 'DELETE_TARGET_PROPERTY_POST_MAP' deletes
             instances of any property in target namespace which are
             used to create associations if the value is set to 'true'
             or if the value is set to 'false' does not delete them.

           Example below shows that (source) concepts with property
           'SNOMED_ICD9_MAP' in SNOMED-CT and (target) concepts with
           property 'Code in Source' in ICD-9-CM will be mapped (associations
           if they have the same property value (codes in this case).
           After associations are created, delete all property instances
           for 'SNOMED_ICD9_MAP' property definitions in the SNOMED CT namespace
           and do not delete all property instances for 'Code in Source' property
           definitions in the 'ICD-9-CM' namespace.

           Note:
           - Both source and target namespace is required to be present
             before creating the associations.
           - Both properties specified in the 'SOURCE_PROPERTY_NAME' and
             'TARGET_PROPERTY_NAME' must be present to successfully create
             associations.
        -->
        <!--
        <mapping sourceNameSpace="SNOMED CT" targetNameSpace="ICD-9-CM" >
                <property name="SOURCE_PROPERTY_NAME" value="SNOMED_ICD9_MAP" />
                <property name="TARGET_PROPERTY_NAME" value="Code in Source" />
                <property name="DELETE_SOURCE_PROPERTY_POST_MAP" value="true" />
                <property name="DELETE_TARGET_PROPERTY_POST_MAP" value="false" />
        </mapping>
        -->
</DTSDbConfig>
```

Mappings

Copyright © 1999-2004 Apelon, Inc. All Rights Reserved.

2. Modify the connection property values (e.g., **user**, **pass**, **host**, etc.) as necessary.

   By default, the **Synonymous Term** block in the kbadmin.xml file is commented out. To convert properties into synonymous terms, uncomment out that block, then modify the elements appropriately. Note that after converting to synonymous terms, the property definition is deleted from the knowledgebase.

   The **Mappings** block in the kbadmin.xml file also is commented out. To map similar property values, uncomment out that block, then modify the elements appropriately.

3. Save the **kbadmin.xml** file.
4. Run the file **kbadmin.bat** (**DTS\bin**) to start the Knowledgebase Admin utility. The utility checks if the specified synonym names, namespaces, and mapping names and namespaces exist. If they do not, Knowledgebase Admin logs the error and stops.
5. Refer to the comments in the **kbadmin.xml** file for additional information regarding the **synonymnous terms** and **mappings** procedures.

Configure Apelon DTS Server

Access to the Apelon DTS Server is based on the configurations you define in the **apelonserverprops.xml** file (**DTS\bin**). Within this file are the properties and configurable values for accessing a database using Apelon DTS. If any of your connection parameters differ from the values established in this file, you must edit the appropriate properties before starting the Apelon DTS server.

If you are using the Apelon DTS Server in secure mode, you can establish **Authentication** to require that users log in (with a password) to a secure socket connection. Set the **authentication** property value to **true** to require login; set to **false** (the default) if the user is not required to log into the server. Both client and server should set the property to **true** to enable a secure connection.

After the database is loaded, use the User Manager utility (**UserManager.bat**) in **DTS\bin** to create a users table and add users.

Copyright © 1999-2004 Apelon, Inc. All Rights Reserved.

apelonserverprops.xml

```
<?xml version="1.0" ?><!DOCTYPE apelonserverconfig SYSTEM "http://apelon.com/dtd/properties/apelonserverconfig.dtd">
<!-- ApelProps-->

<apelonserverconfig>
	<property name="port" value="6666" />
	<property name="log" value="apelonserverlog.xml" />
	<property name="validate_parser" value="true" />
	<property name="authentication" value="false" />
	<property name="user_admin" value="com.apelon.dts.server.DTSUsers" />

	<connection default="true">
		<property name="type" value="oracle" />
		<property name="jdbcDriver" value="oracle.jdbc.driver.OracleDriver" />
		<property name="url_template" value="jdbc:oracle:thin:@[HOST]:[PORT]:[DATABASE]" />
		<property name="user" value="dts3" />
		<property name="pass" value="dts3" />
		<property name="host" value="localhost" />
		<property name="database_name" value="ORCL" />
		<property name="database_port" value="1521" />
	</connection>

	<queryserver>
		<property name="qs" value="com.apelon.dts.server.QueryServer" />
		<property name="maxDbConns" value="15" />
		<property name="header" value="DTS:001" />
	</queryserver>

	<queryserver>
		<property name="qs" value="com.apelon.dts.server.MatchServer" />
		<property name="maxDbConns" value="15" />
		<property name="header" value="DTS:004" />
	</queryserver>

	<queryserver>
		<property name="qs" value="com.apelon.dts.server.SubConceptServer" />
		<property name="maxDbConns" value="15" />
		<property name="header" value="DTS:005" />
	</queryserver>
```

Copyright © 1999-2004 Apelon, Inc. All Rights Reserved.

apelonserverprops.xml, continued

```
        <queryserver>
                <property name="qs" value="com.apelon.dts.server.NavQueryServer" />
                <property name="maxDbConns" value="15" />
                <property name="header" value="DTS:006" />
        </queryserver>

        <queryserver>
                <property name="qs" value="com.apelon.dts.server.SearchQueryServer" />
                <property name="maxDbConns" value="15" />
                <property name="header" value="DTS:007" />
        </queryserver>

        <queryserver>
                <property name="qs" value="com.apelon.dts.server.TranslateQueryServer" />
                <property name="maxDbConns" value="15" />
                <property name="header" value="DTS:008" />
        </queryserver>

        <queryserver>
                <property name="qs" value="com.apelon.dts.server.ThesaurusConceptServer" />
                <property name="maxDbConns" value="15" />
                <property name="header" value="DTS:009" />
        </queryserver>

        <queryserver>
                <property name="qs" value="com.apelon.dts.server.AssociationServer" />
                <property name="maxDbConns" value="15" />
                <property name="header" value="DTS:010" />
        </queryserver>

        <queryserver>
                <property name="qs" value="com.apelon.dts.server.NamespaceServer" />
                <property name="maxDbConns" value="15" />
                <property name="header" value="DTS:011" />
        </queryserver>

        <queryserver>
                <property name="qs" value="com.apelon.dts.server.TermServer" />
                <property name="maxDbConns" value="15" />
                <property name="header" value="DTS:012" />
        </queryserver>

        <queryserver>
                <property name="qs" value="com.apelon.dts.server.DTSLicenseServer" />
                <property name="maxDbConns" value="15" />
                <property name="header" value="DTS:013" />
        </queryserver>

        <queryserver>
                <property name="qs" value="com.apelon.dts.server.DTSConceptServer" />
                <property name="maxDbConns" value="15" />
                <property name="header" value="DTS:014" />
        </queryserver>

        <queryserver>
                <property name="qs" value="com.apelon.dts.server.OntylogConceptServer" />
                <property name="maxDbConns" value="15" />
                <property name="header" value="DTS:015" />
        </queryserver>

</apelonserverconfig>
```

Copyright © 1999-2004 Apelon, Inc. All Rights Reserved.

Start Servers and DTS Browser

Start Apelon DTS Server

When you start the Apelon DTS Server (**Start>Programs>Apelon>Apelon DTS>Start Apelon DTS Server**) the window that displays inicludes the following to indicate that the server started successfully.

```
Starting Apelon Server on 192.168.1.35:6666
Initializing socketServer
The validation of XML parser is on
RSS: Listening on port: 6666
Initializing adminServer
Starting socketServer
```

Start Tomcat Server

To run the Apelon DTS Browser, you first must start the Tomcat Server on the machine where the DTS Browser component is installed. When you start the Tomcat Server (**Start>Programs>Apelon>Apelon DTS>Start Tomcat Server**) this window displays to indicate the server started successfully.

```
Start Tomcat
Using CATALINA_BASE:   ..
Using CATALINA_HOME:   ..
Using CATALINA_TMPDIR: ..\temp
Using JAVA_HOME:       ..\..\java\jdk
Starting service Tomcat-Standalone
Apache Tomcat/4.1.27
```

NOTE: If you access the DTS Browser across the Internet through a "firewall," you may need to change the Tomcat connection port. If this is the case, open the **server.xml** file (**DTS\tomcat\conf**) and change the default connection port from **8081** to a port that is open through the firewall (at some installations, port **80** may be the only one available). Note the highlighted port designation in the following illustration.

```
<Connector className="org.apache.catalina.connector.http.HttpConnector"

            port="8081"
           minProcessors="5"
           maxProcessors="75"
           enableLookups="true"
           redirectPort="8443"
           acceptCount="10"
           debug="0"
           connectionTimeout="60000"/>
```

Copyright © 1999-2004 Apelon, Inc. All Rights Reserved.

Start Apelon DTS Browser

When you start the Apelon DTS Browser (**Start>Programs>Apelon>Apelon DTS> Apelon DTS Browser**) a Browser window will open and you will be prompted to enter a host and port. If you are hitting the local machine, just enter **localhost / 6666**.

If you receive a "page cannot be displayed error" because you changed the default connection port from **8081** in the **server.xml** file, you must change the address (URL) to reflect the modified port. For example, change http://localhost:**8081**/dtstreebrowser/index.jsp to http://localhost:**80**/dtstreebrowser/index.jsp. You also can edit the properties on the Apelon DTS Browser link off the **Start** menu so that the proper connection will be used from that point forward.

Back to Top

Copyright © 1999-2004 Apelon, Inc. All Rights Reserved.