Tamino

X-Node: Mapping to External Databases

Version 9.7

April 2015

WEBMETHODS

This document applies to Tamino Version 9.7.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Document ID: INS-XNODE-97-20160318

Table of Contents

Preface

This document describes the X-Node feature of Tamino. It is intended for database administrators and application programmers.

The X-Node feature enables the Tamino X-Machine to communicate with non-Tamino databases.

This information is structured into the following sections:

1 X-Node Access to Adabas

This section gives an overview of accessing an Adabas database from Tamino via X-Node, and provides various examples.

The following section is structured as follows:

General

Network setup

If Tamino and Adabas are running on the same machine, the X-Node link between them is available automatically when Adabas version 3.3 or later is running, therefore you do not have to configure the software. This is because these Adabas versions include the ADALNKX DLL (Windows) or shared library (UNIX) that builds the connection automatically.

If Tamino and Adabas version 3.2 are running on the same machine, the ADALNKX DLL (Windows) or shared library (UNIX) is required in order to build an X-Node link. You can install ADALNKX by installing either the Entire Net-Work Client software or the full Entire Net-Work package.

Access to a remote Adabas installation requires the use of the XTS internal product, which is included in the Tamino kit.

Using Natural date and time fields

Natural stores date and time values into Adabas fields with format "P", i.e., as integer numbers. If such a Natural date field has the datatype `xs:date` in the Tamino schema, then the internal value representation will be converted to a readable date value according to the XML Schema specification. Similarly, a Natural time field is specified as `xs:dateTime`. Also, when storing an XML document into an Adabas X-Node mapped doctype, `xs:date` and `xs:dateTime` fields will be converted into the Natural internal format.

Natural date and time values have the following restrictions:

- Dates range from a hypothetical "Jan 1, 0000" to "Dec 31, 2699".
- A Natural time has a precision of 0.1 seconds, whereas XML Schema allows arbitrary fractions of a second.

As a consequence, when storing XML documents into an Adabas X-Node mapped doctype, dates outside the Natural date range will be rejected, and seconds will be rounded to a precision of 0.1 seconds.

Tamino Access to Adabas on Windows

The following section describes how to configure Tamino to access Adabas, and also includes examples of read, write and delete operations.

Reading from an Adabas Database

Perform the following steps:

1. Using the Tamino Manager, create and start a Tamino database (or choose an existing one);
2. The file called *ada_empl.tsd* in the directory *Documentation/tsl* under the Tamino installation directory contains a TSD schema for the example described in the section *Example Schema for Adabas Mapping* in the Tamino Schema Definition Language documentation.

   In this file, change the database number (to `211` in the example) and the file number. The element `tsd:subTreeAdabas` has two attributes `dbid` and `fnr` that contain this information;
3. In the Tamino Interactive Interface, define the collection by performing the following steps:
   - In the field **Database URL** of the **Define** tab, enter the name of the Tamino database (replace *`mydb`* with the name of your database);
   - Enter the location of *ada_empl.tsd* in the **Schema file** field and choose the **Define** button.

     The message

     ```
     <ino:messageline>_DEFINE: schema ada_empl in collection ada_empl ↩
     defined</ino:messageline>
     ```

     shows that the definition was successful;
4. Access the desired data on Adabas:
   - In the field **Collection** of the **X-Query** tab, enter the name of the collection, namely "ada_empl".
   - In the Query field, enter a query expression and choose the **Query** button.

     Examples:

```
employee[name/surname='ADAM']
employee[address/city='ATLANTA']
```

Writing to and Deleting from an Adabas Database

You can also write data into an Adabas database. To do this, perform these steps:

1. Create an XML instance of the schema that you defined in the previous step. For example, you can copy the sample XML instance that defines an employee "James Bond" from the documentation section *Example Schema for Adabas Mapping* mentioned above. In the following example, the XML instance file is called *bond.xml*.
2. Specify the location of the *bond.xml* file in the **Load file** field of the **Load** tab of the Tamino Interactive Interface, then choose the **Load** button. Tamino should generate the following response:

   ```
   <ino:messageline>document processing started</ino:messageline>
   </ino:message>
   <ino:object ino:collection="ada_empl" ino:doctype="employee" ino:id="1" />
   <ino:message ino:returnvalue="0">
   <ino:messageline>document processing ended</ino:messageline>
   ```

   Note that you can only write at most one “James Bond” instance to the Adabas database, because of unique key requirements.
3. Specify the following query in the **X-Query** field of the **X-Query** tab:

   ```
   employee[name/surname='Bond']
   ```

   then click on the **Query** button. Note that this example assumes that you have used the sample data from the Tamino user documentation. If you created other data, replace the query shown above by an appropriate query for your data. Note also that the query is case-sensitive, so in this example, you must specify "Bond" and not "BOND".

To delete this element, do the following:

1. Enter the following query in the **Delete Query** field of the **Delete** tab:

   ```
   employee[name/surname='Bond']
   ```

   then click on the **Delete** button.

   If the deletion is successful, Tamino returns the following response:

```
  <ino:messageline>_DELETE: document(s) deleted</ino:messageline> ↩
```

Tamino Access to Adabas on UNIX

The following section describes how to configure Tamino to access Adabas and gives examples of read, write and delete operations.

- Reading From an Adabas Database
- Writing To and Deleting From an Adabas Database

Reading From an Adabas Database

The instructions for reading from an Adabas database on UNIX are the same as for reading from an Adabas database on Windows, with one restriction on the use of the Mozilla browser, as described below. See the section ***Reading from an Adabas Database*** in the Windows section above for the general instructions on reading from an Adabas database.

The restriction when using Mozilla is as follows: When the Tamino Interactive Interface is running in the Mozilla browser, the extension *.tsd* is not recognized, therefore it is necessary to assign a different extension such as *.xml* to the schema file. You can do this by executing a copy command in the *Documentation/tsl* directory:

```
cd $INODIR/%INOVERS/Documentation/tsl
cp ada_empl.tsd ada_empl.xml
```

Writing To and Deleting From an Adabas Database

The instructions for writing to and deleting from an Adabas database on UNIX are the same as for writing to and deleting from an Adabas database on Windows. See the section ***Writing To and Deleting From an Adabas Database*** in the Windows section above for general instructions on writing to and deleting from an Adabas database.

The restriction on the use of the Mozilla browser described in the previous section also applies when writing and deleting.

2 X-Node Mapping Examples

Examples of how to specify the formal relationship between a Tamino schema and an external database table are provided in the External Mapping section in the chapter *Tamino-Specific Extensions to the Physical Schema* of the documentation for Tamino XML schema language.

Note: If you intend to use X-Node to connect to a read-only or replication (simultaneous query) database of a relational database system, all primary keys must be mapped.

Index