
Reladomo Database Definition Generator

20 June 2006

Table of Contents

1. Introduction	1
2. Usage	1
2.1. DDL Generation from Ant	1
2.2. Direct DDL Generation	2
3. Internals	2
3.1. Relevant Hierarchical Information	2
3.2. Caveats	2

1. Introduction

Database definition language (DDL) file generation is an optional feature in Reladomo that allows users to generate scripts to create tables, indices and foreign keys from the Reladomo object definition XML files. These scripts are stored in .ddl, .idx and .fk files, respectively. To date, there is support for Sybase and UDB databases.

2. Usage

DDL generation may be run through direct interaction with the `MithraDbDefinitionGenerator` class or, more simply, from an Ant task.

2.1. DDL Generation from Ant

An Ant task provides users with a simple way to generate database files for a large set of Reladomo object definition XML files. Consider the following example as it points to a number of XML elements used by `MithraDbDefinitionGenerator`:

```
<target name="generate-example-mithra-db"
        depends="compile-mithra-generator">
  <taskdef name="mithra-gen"
           classname
           = "com.gs.fw.common.mithra.generator.dbgenerator.MithraDbDefinitionGenerator"
           loaderRef="mithraGenerator">
    <classpath refid="mithragen.classpath"/>
  </taskdef>
  <mithra-gen xml="${example.home}/xml/mithra/example/
ExampleClassList.xml"
             generatedDir="${generated.src.dir}/db"
             buildList="Product, ProductPrice"
             databaseType="sybase" />
</target>
```

Of interest are the elements for the `mithra-gen` tag above.

The `xml` element specifies the location of the Reladomo class list -- a list of all of the classes to be generated by Reladomo. By default, the DDL generator will create database definition files for all classes. If `buildList` is provided, the generator will only produce definition files for the classes listed. Keep in mind that the include list is a comma-separated list of the Reladomo classes associated with tables that one wishes to generate.

The database type must be specified using the `databaseType` element. Acceptable values for the type are "sybase" and "udb82".

2.2. Direct DDL Generation

To run DDL generation, one creates an instance of `com.gs.fw.common.mithra.generator.dbgenerator.MithraDbDefinitionGenerator` and makes a call to `MithraDbDefinitionGenerator.execute()`. Before calling `execute()`, one must set the destination directory for the generated database scripts, the location of the Reladomo object definition XML files and the database type with calls to `MithraDbDefinitionGenerator.setGeneratedDir()`, `setXml()` and `setDatabaseType()`, respectively.

The generator currently accepts "sybase" and "udb82" as possible database types. If no database type is specified or an invalid type is provided, the generator throws a `BuildException`.

3. Internals

3.1. Relevant Hierarchical Information

The `MithraDbDefinitionGenerator` class implements the abstract class `AbstractMithraGenerator` and makes extensive use of the `com.gs.fw.common.mithra.generator.MithraObjectTypeWrapper` and `com.gs.fw.common.mithra.databasetype.AbstractDatabaseType` classes.

A `MithraObjectTypeWrapper` object maps to a database table. As such, the generator uses it to gather information about tables, their columns and their indexing.

The SQL data types used in the `.ddl` files are retrieved from the database-appropriate child of `AbstractDatabaseType`. For example, if one specifies Sybase as the database type, the generator will determine the SQL data types from the `SybaseDatabaseType` class.

3.2. Caveats

In general, SQL data types match up reasonably well with Java data types. In cases of multiple options or ambiguity, the developers were forced to choose specific or peculiar mappings. Where the mapping is not intuitive, we have provided details:

Table 1.

Java Type	Sybase	UDB
long	numeric(18,0)	bigint
String	varchar()	varchar()

It is also worth noting that unless `maxLength` is specified for a `String` attribute in its Reladomo object definition XML, the generator will create a column of type `varchar(255)`.