OMONDO®

THE LIVE UML COMPANY

OMG Workshop

Modeling and IDEs

Orlando, FL USA

Speakers:

Mr Xavier Maysonnave – Founder of Omondo (Database Lead Developer)

xavier@omondo.com

The History of Omondo

- Company creation with headquarters in Paris in September 2002
- Research and coding of EclipseUML started in 2001
- EclipseUML Free Edition launched in September 2002 *(more than 300,000 regular users)*
- EclipseUML Studio launched in June 2004
- Native EclipseUML2 metamodel and EclipseUML Studio integration

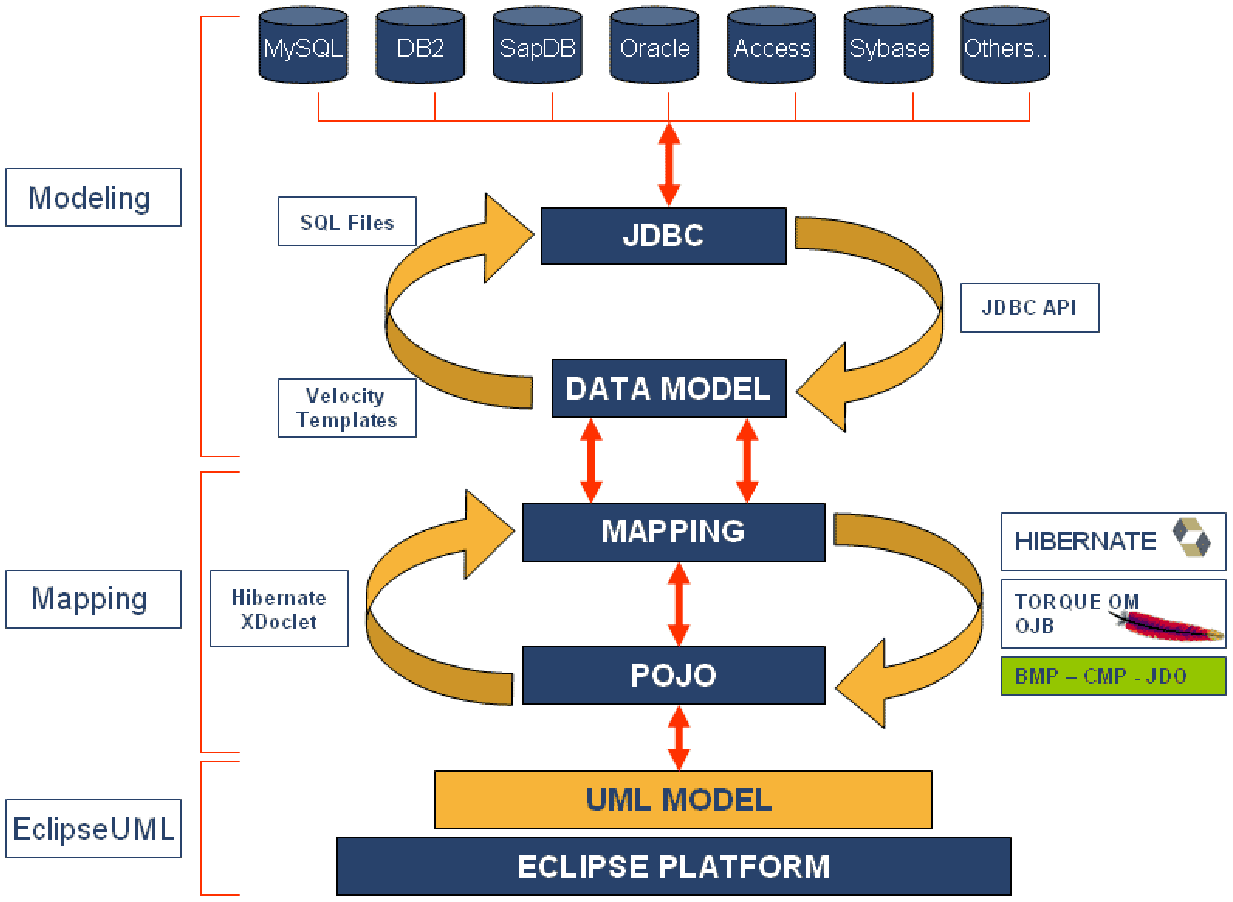
  in February 2005.

OMONDO®
THE LIVE UML COMPANY

Agenda

- **Overview**
- **Database Cycle**
  - Reverse Engineering
  - Forward Engineering
- **Code Cycle**
  - Apache Torque and Apache OJB
  - Hibernate
  - UML Profile
- **What’s Next**
  - More Tools
  - JSR-220 Annotation Support

OMONDO® THE LIVE UML COMPANY



Database Cycle

- **Reverse Engineering** is **JDBC** based
- Local **Schemas** are XML files
- Standard **DTD**s are provided
  - Apache Turbine
  - Apache Torque
  - Local and Remote DTD Validation
- Direct XML edition is always possible
  - Fast and easy for small updates
  - No Schema Validation

Database Cycle

XML Configuration File

```
<?xml version="1.0" encoding="UTF-8"?>
<connection xmlns="http://www.omondo.com/database/connection">
  <database name="firebird"/>
  <user name="sysdba" password="qlIQiSncAdvwrDgkK9+1rQ=="/>
  <jdbc url="jdbc:firebirdsql://localhost/employee"/>
  <classpath value="file:OMONDO_DATABASE_HOME/misc/FireBird/1.0.1/firebirdsql-full.jar"/>
  <classpath value="file:OMONDO_DATABASE_HOME/misc/MySQL/3.1.7/mysql-connector-java-3.1.7-bin.jar"/>
  <classpath value="file:OMONDO_DATABASE_HOME/misc/PostgreSQL/postgresql-8.0-310.jdbc3.jar"/>
  <schema value="file:./schema.ods.xml" validate="false" filterView="true" filterIndex="false" filterForeignKey="false" filterPrimary="true«  filterForeign="false" />
  <diagram value="file:./diagram.odd"/>
</connection>
```

XML Schema File

```
 <table name="PROJ_DEPT_BUDGET" idMethod="none">
    <column name="FISCAL_YEAR" type="INTEGER" primaryKey="true" position="1" required="true" default="0" javaName="fiscalYear" javaType="primitive"/>
    <column name="PROJ_ID" type="VARCHAR" size="5" primaryKey="true" position="2" required="true" javaName="projId" javaType="primitive"/>
    <column name="DEPT_NO" type="CHAR" size="3" primaryKey="true" position="3" required="true" javaName="deptNo" javaType="primitive"/>
    <column name="QUART_HEAD_CNT" type="INTEGER" javaType="primitive"/>
    <column name="PROJECTED_BUDGET" type="DECIMAL" size="12" precision="5" javaType="primitive"/>
    <foreign-key name="INTEG_48" foreignTable="PROJECT" minCardinality="0" maxCardinality="*" onUpdate="none" onDelete="none">
      <reference local="PROJ_ID" foreign="PROJ_ID" position="1"/>
    </foreign-key>
    <foreign-key name="INTEG_47" foreignTable="DEPARTMENT" minCardinality="0" maxCardinality="*" onUpdate="none" onDelete="none">
      <reference local="DEPT_NO" foreign="DEPT_NO" position="1"/>
    </foreign-key>
    <unique name="DEPT_NO">
      <unique-column name="DEPT_NO" position="1" direction="A"/>
    </unique>
    <unique name="PROJ_ID">
      <unique-column name="PROJ_ID" position="1" direction="A"/>
    </unique>
  </table>
```

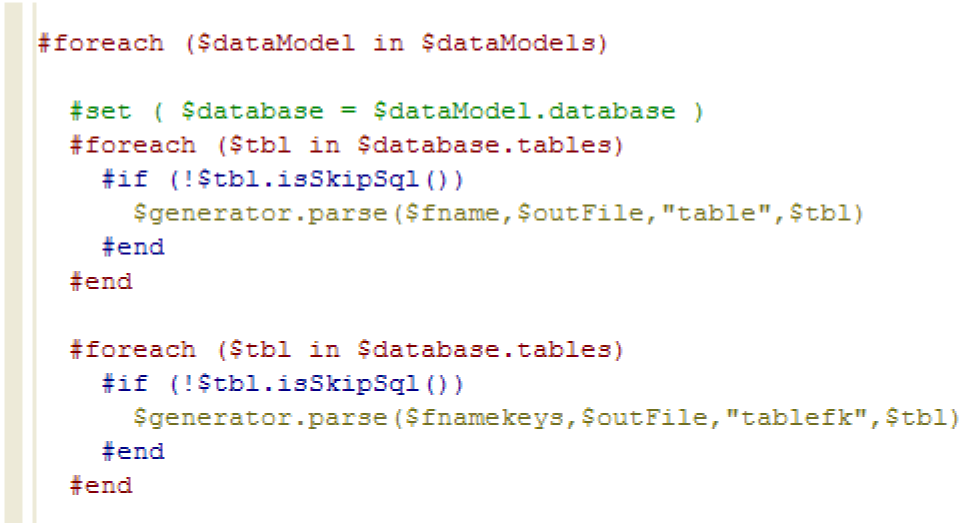
Database Cycle

- **Database Diagram**
  - Model Validation
- **SQL 92** Import
- Data Import and Export
- **SQL Generation** is **Velocity** based
- **Forward Engineering**
- Torque Templates
  - The benefit of a community
  - Torque Model

THE LIVE UML COMPANY

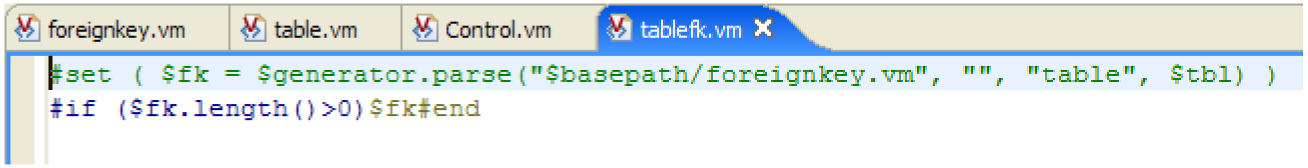
Database Cycle

Control.vm



```
#foreach ($dataModel in $dataModels)

  #set ( $database = $dataModel.database )
  #foreach ($tbl in $database.tables)
    #if (!$tbl.isSkipSql())
      $generator.parse($fname,$outFile,"table",$tbl)
    #end
  #end

  #foreach ($tbl in $database.tables)
    #if (!$tbl.isSkipSql())
      $generator.parse($fnamekeys,$outFile,"tablefk",$tbl)
    #end
  #end
```

tablefk.vm

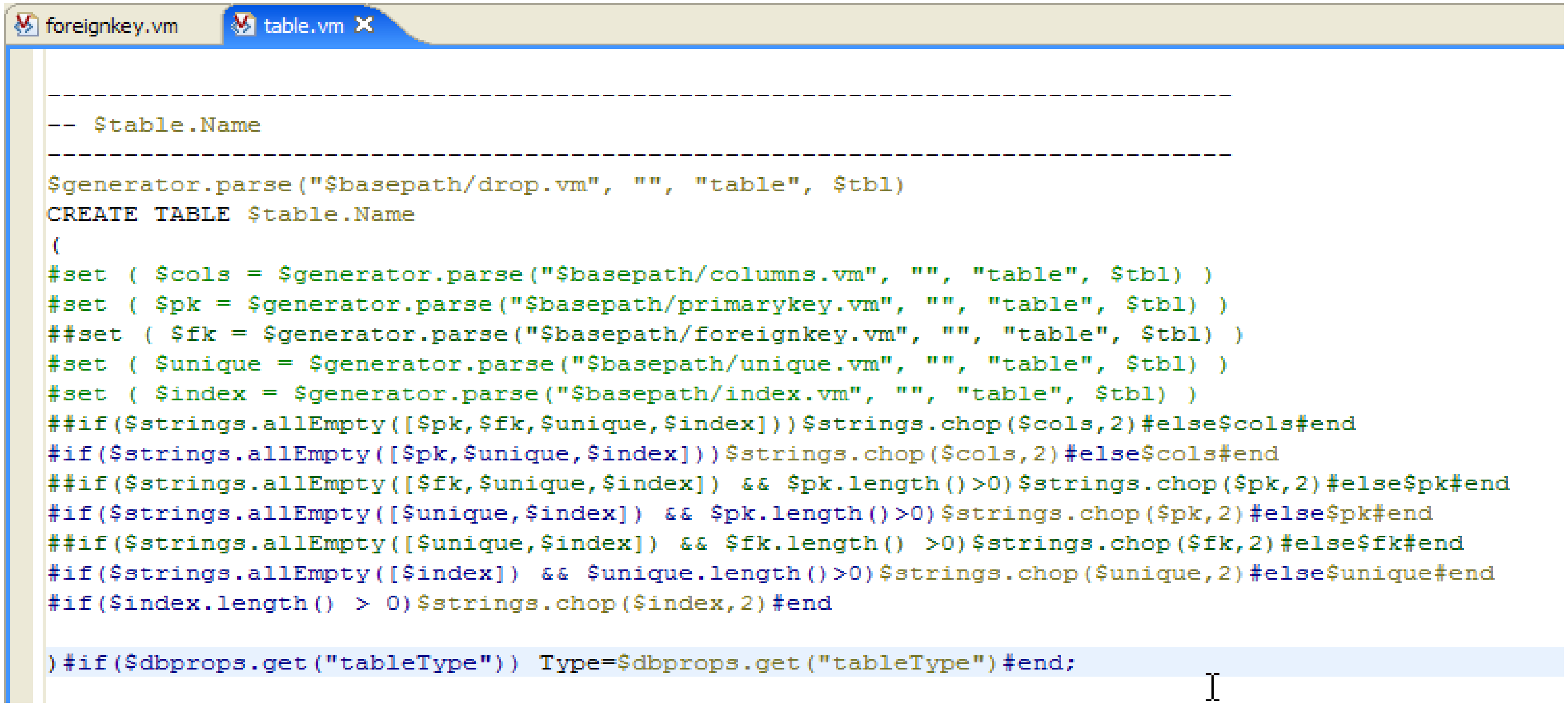


```
#set ( $fk = $generator.parse("$basepath/foreignkey.vm", "", "table", $tbl) )
#if ($fk.length()>0)$fk#end
```

THE LIVE UML COMPANY

Database Cycle

table.vm

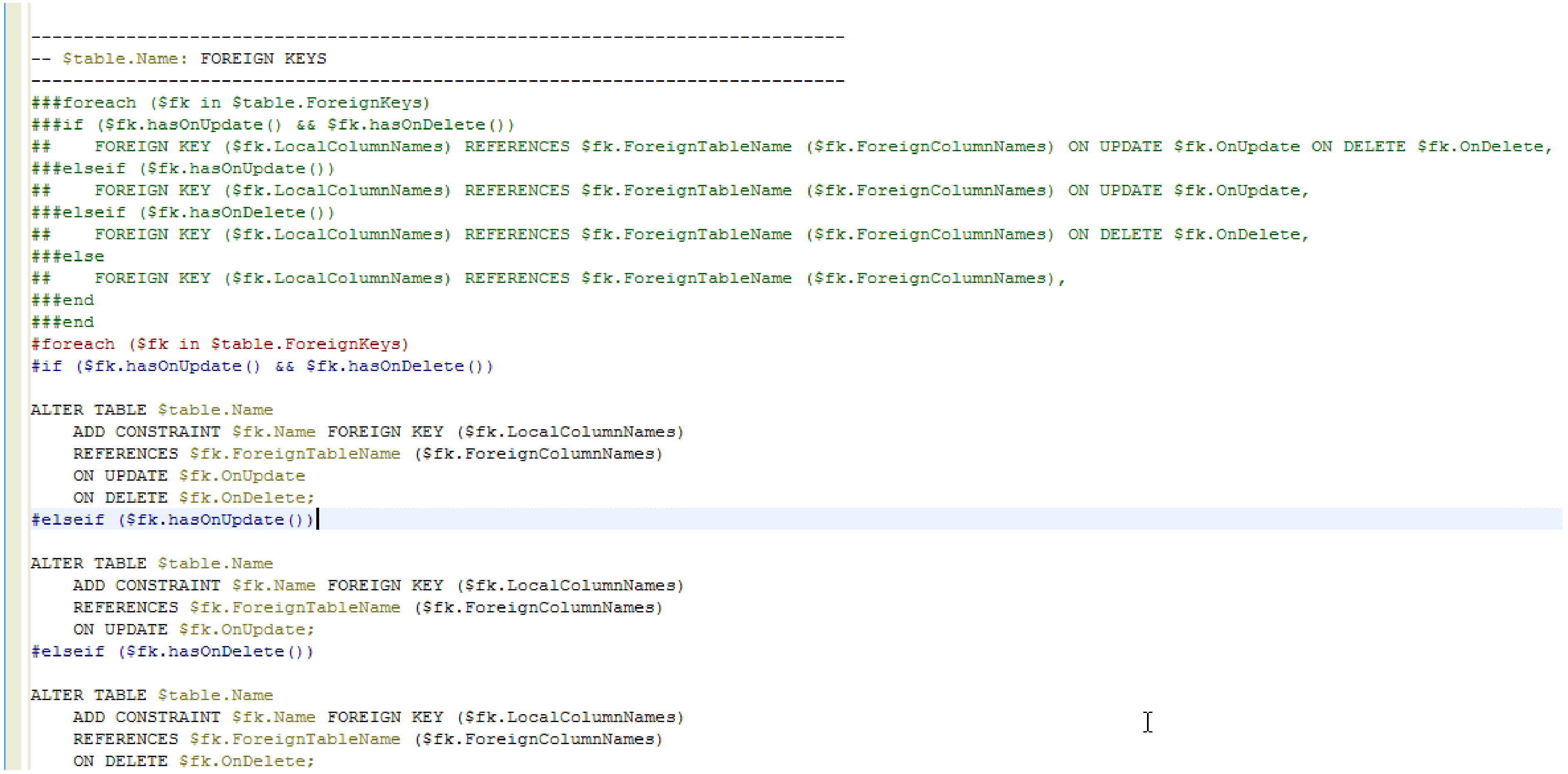


foreignkey.vm | table.vm

```
-------------------------------------------------------------------------------
-- $table.Name
-------------------------------------------------------------------------------
$generator.parse("$basepath/drop.vm", "", "table", $tbl)
CREATE TABLE $table.Name
(
#set ( $cols = $generator.parse("$basepath/columns.vm", "", "table", $tbl) )
#set ( $pk = $generator.parse("$basepath/primarykey.vm", "", "table", $tbl) )
##set ( $fk = $generator.parse("$basepath/foreignkey.vm", "", "table", $tbl) )
#set ( $unique = $generator.parse("$basepath/unique.vm", "", "table", $tbl) )
#set ( $index = $generator.parse("$basepath/index.vm", "", "table", $tbl) )
##if($strings.allEmpty([$pk,$fk,$unique,$index]))$strings.chop($cols,2)#else$cols#end
#if($strings.allEmpty([$pk,$unique,$index]))$strings.chop($cols,2)#else$cols#end
##if($strings.allEmpty([$fk,$unique,$index]) && $pk.length()>0)$strings.chop($pk,2)#else$pk#end
#if($strings.allEmpty([$unique,$index]) && $pk.length()>0)$strings.chop($pk,2)#else$pk#end
##if($strings.allEmpty([$unique,$index]) && $fk.length() >0)$strings.chop($fk,2)#else$fk#end
#if($strings.allEmpty([$index]) && $unique.length()>0)$strings.chop($unique,2)#else$unique#end
#if($index.length() > 0)$strings.chop($index,2)#end

)#if($dbprops.get("tableType")) Type=$dbprops.get("tableType")#end;
```

THE LIVE UML COMPANY

Database Cycle



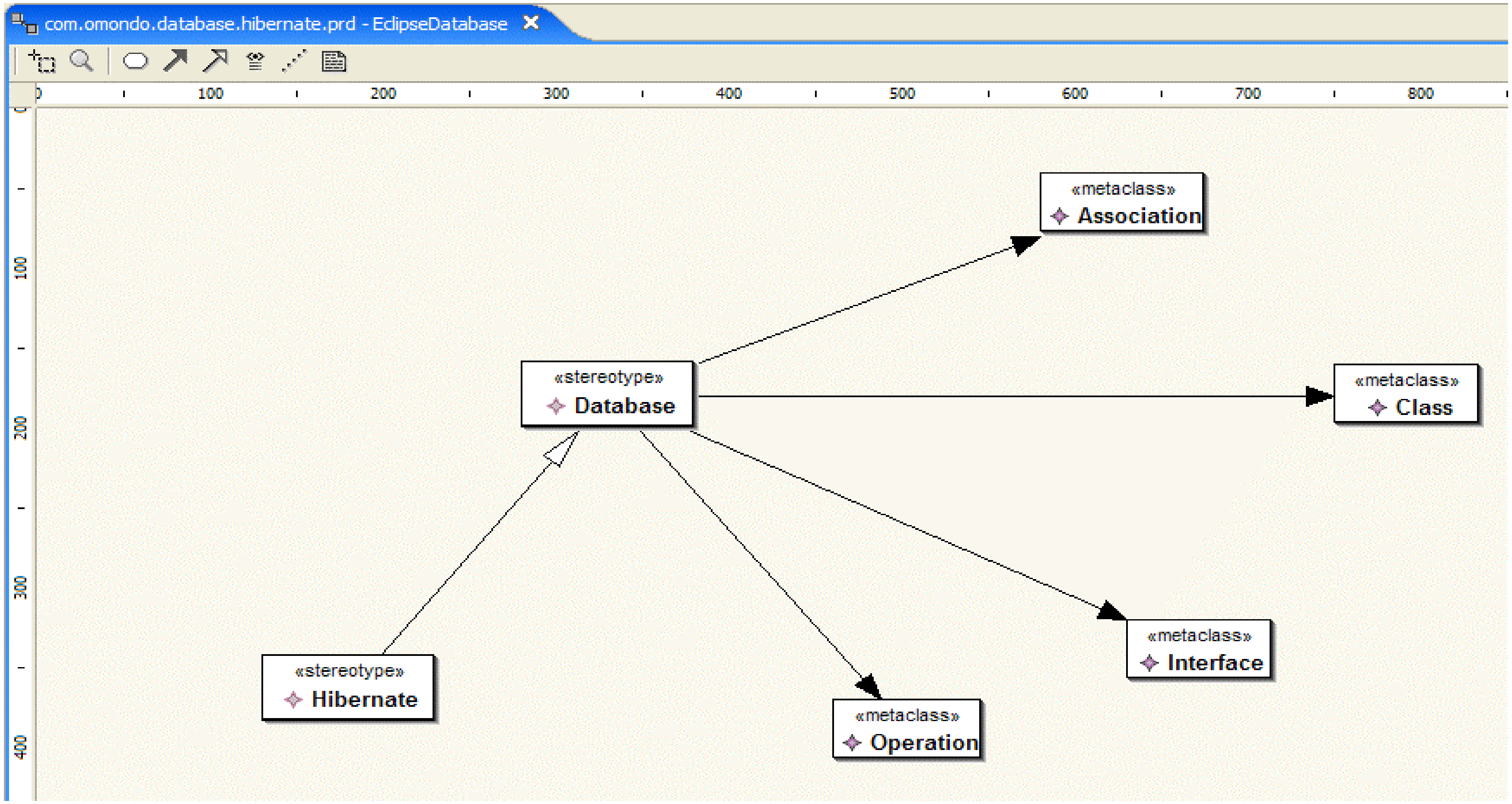
```
-----------------------------------------------------------------------------
-- $table.Name: FOREIGN KEYS
-----------------------------------------------------------------------------
###foreach ($fk in $table.ForeignKeys)
###if ($fk.hasOnUpdate() && $fk.hasOnDelete())
##     FOREIGN KEY ($fk.LocalColumnNames) REFERENCES $fk.ForeignTableName ($fk.ForeignColumnNames) ON UPDATE $fk.OnUpdate ON DELETE $fk.OnDelete,
###elseif ($fk.hasOnUpdate())
##     FOREIGN KEY ($fk.LocalColumnNames) REFERENCES $fk.ForeignTableName ($fk.ForeignColumnNames) ON UPDATE $fk.OnUpdate,
###elseif ($fk.hasOnDelete())
##     FOREIGN KEY ($fk.LocalColumnNames) REFERENCES $fk.ForeignTableName ($fk.ForeignColumnNames) ON DELETE $fk.OnDelete,
###else
##     FOREIGN KEY ($fk.LocalColumnNames) REFERENCES $fk.ForeignTableName ($fk.ForeignColumnNames),
###end
###end
#foreach ($fk in $table.ForeignKeys)
#if ($fk.hasOnUpdate() && $fk.hasOnDelete())

ALTER TABLE $table.Name
    ADD CONSTRAINT $fk.Name FOREIGN KEY ($fk.LocalColumnNames)
    REFERENCES $fk.ForeignTableName ($fk.ForeignColumnNames)
    ON UPDATE $fk.OnUpdate
    ON DELETE $fk.OnDelete;
#elseif ($fk.hasOnUpdate())

ALTER TABLE $table.Name
    ADD CONSTRAINT $fk.Name FOREIGN KEY ($fk.LocalColumnNames)
    REFERENCES $fk.ForeignTableName ($fk.ForeignColumnNames)
    ON UPDATE $fk.OnUpdate;
#elseif ($fk.hasOnDelete())

ALTER TABLE $table.Name
    ADD CONSTRAINT $fk.Name FOREIGN KEY ($fk.LocalColumnNames)
    REFERENCES $fk.ForeignTableName ($fk.ForeignColumnNames)
    ON DELETE $fk.OnDelete;
```

Code Cycle

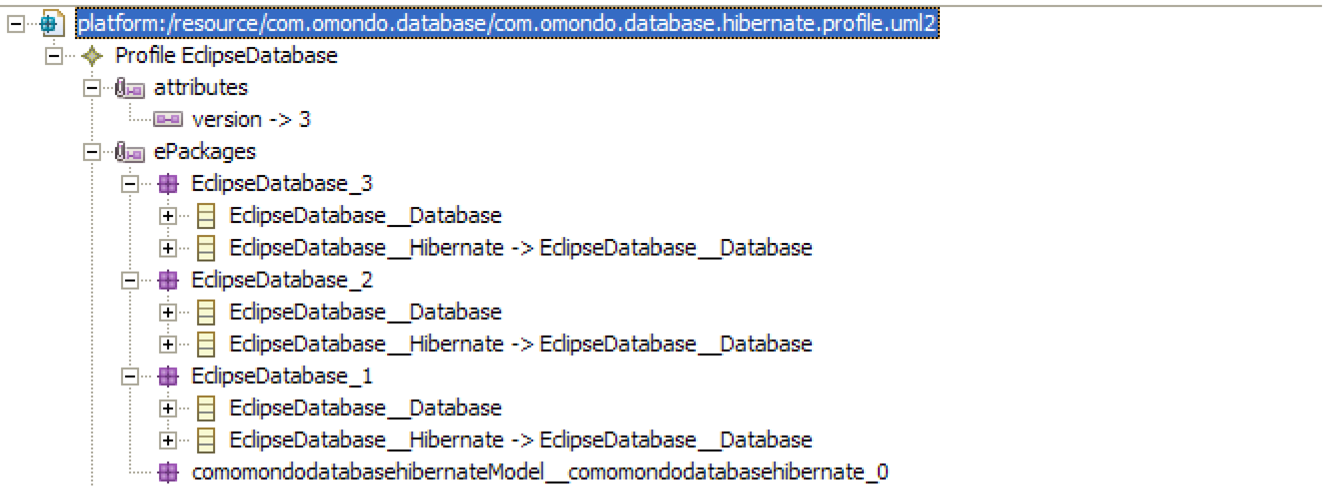
- **Apache Torque**
- **Apache OJB**
- **Hibernate**
    - **XDoclet** Support
    - **Configuration** and **Mapping** File
    - **hbm2java** a Standard Code Generator
    - Import and Export
    - **UML Profile**

THE LIVE UML COMPANY

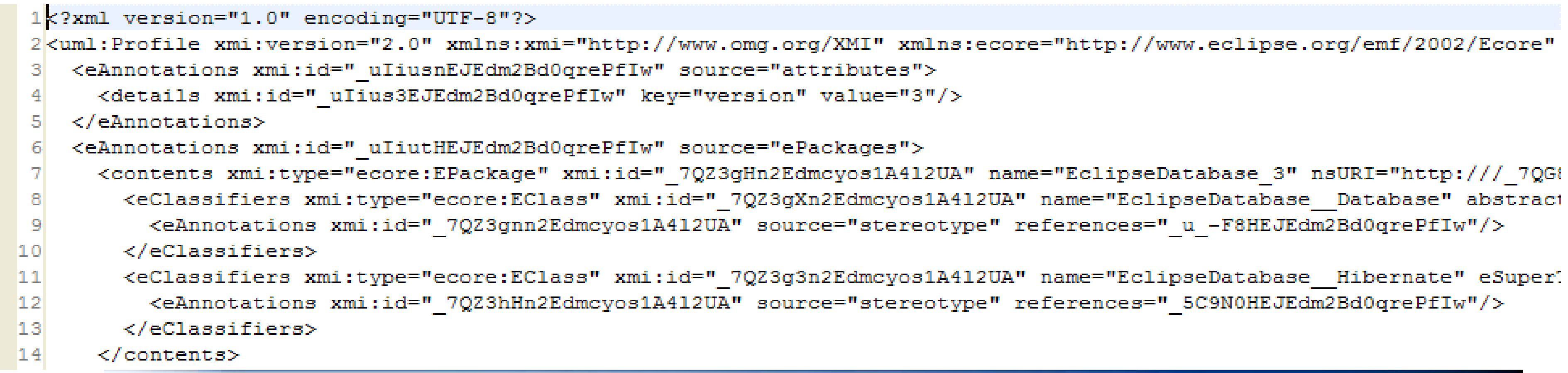
Code Cycle



Code Cycle



- platform:/resource/com.omondo.database/com.omondo.database.hibernate.profile.uml2
  - Profile EclipseDatabase
    - attributes
      - version -> 3
    - ePackages
      - EclipseDatabase_3
        - EclipseDatabase__Database
        - EclipseDatabase__Hibernate -> EclipseDatabase__Database
      - EclipseDatabase_2
        - EclipseDatabase__Database
        - EclipseDatabase__Hibernate -> EclipseDatabase__Database
      - EclipseDatabase_1
        - EclipseDatabase__Database
        - EclipseDatabase__Hibernate -> EclipseDatabase__Database
      - comomondodatabasehibernateModel__comomondodatabasehibernate_0



```
<?xml version="1.0" encoding="UTF-8"?>
<uml:Profile xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI" xmlns:ecore="http://www.eclipse.org/emf/2002/Ecore"
  <eAnnotations xmi:id="_uIiusnEJEdm2Bd0qrePfIw" source="attributes">
    <details xmi:id="_uIius3EJEdm2Bd0qrePfIw" key="version" value="3"/>
  </eAnnotations>
  <eAnnotations xmi:id="_uIiutHEJEdm2Bd0qrePfIw" source="ePackages">
    <contents xmi:type="ecore:EPackage" xmi:id="_7QZ3gHn2Edmcyos1A412UA" name="EclipseDatabase_3" nsURI="http:///_7QG
      <eClassifiers xmi:type="ecore:EClass" xmi:id="_7QZ3gXn2Edmcyos1A412UA" name="EclipseDatabase__Database" abstract
        <eAnnotations xmi:id="_7QZ3gnn2Edmcyos1A412UA" source="stereotype" references="_u_-F8HEJEdm2Bd0qrePfIw"/>
      </eClassifiers>
      <eClassifiers xmi:type="ecore:EClass" xmi:id="_7QZ3g3n2Edmcyos1A412UA" name="EclipseDatabase__Hibernate" eSuper
        <eAnnotations xmi:id="_7QZ3hHn2Edmcyos1A412UA" source="stereotype" references="_5C9N0HEJEdm2Bd0qrePfIw"/>
      </eClassifiers>
    </contents>
```

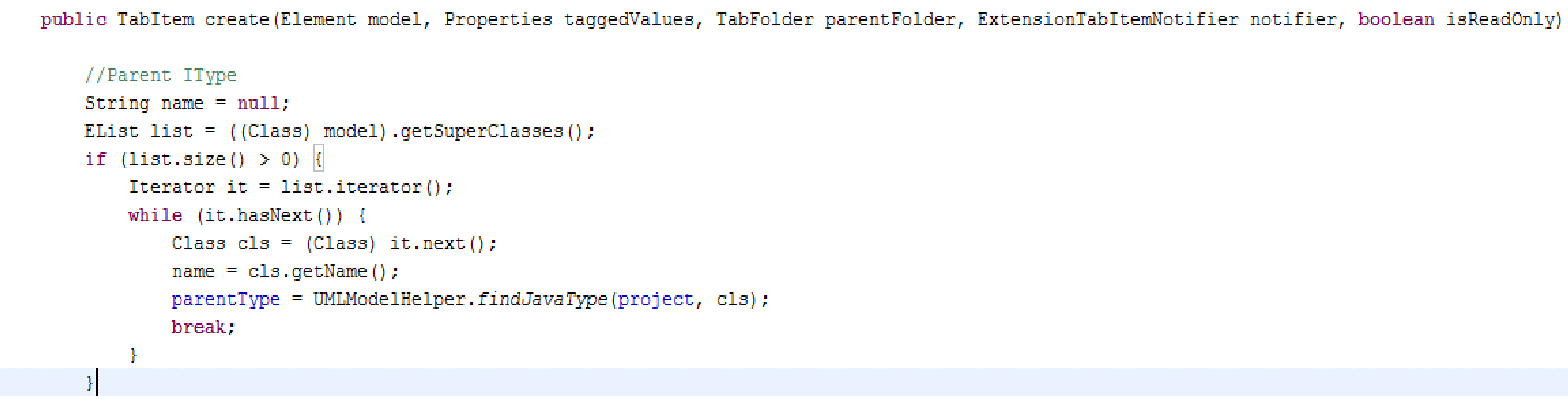
Code Cycle

- **UML Profile**
  - Diagram enhancement with your own logic
  - Interoperability based on XMI 2.0
- **Hibernate UML Profile**
  - **Class Diagram XDoclet** integration
  - **Mapping** File generation

Code Cycle

```
<extension
    point="com.omondo.uml.std.profiles"
>
    <profile
        name="%databaseHibernateProfile"
        class="com.omondo.database.external.profile.HibernateProfile"
        id="com.omondo.database.external.profile.HibernateProfile.id"
        model="com.omondo.database.hibernate.profile.uml2"
    >
        <fragment
            class="com.omondo.database.external.profile.HibernateClassFragment"
            diagramType="classDiagram"
            elementType="class"
            stereotype="EclipseDatabase::Hibernate"
        />
        <fragment
            class="com.omondo.database.external.profile.HibernateMethodFragment"
            diagramType="classDiagram"
            elementType="operation,interface"
            stereotype="EclipseDatabase::Hibernate"
        />
    </profile>
</extension>
```

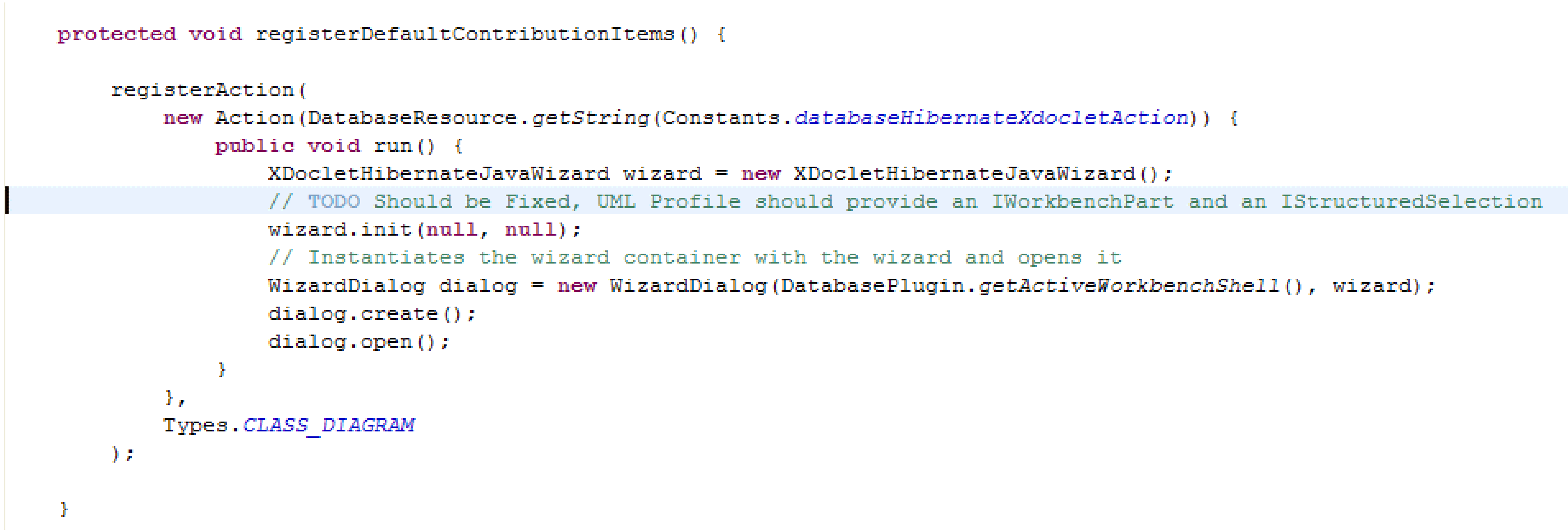
Code Cycle



```
public TabItem create(Element model, Properties taggedValues, TabFolder parentFolder, ExtensionTabItemNotifier notifier, boolean isReadOnly)

    //Parent IType
    String name = null;
    EList list = ((Class) model).getSuperClasses();
    if (list.size() > 0) {
        Iterator it = list.iterator();
        while (it.hasNext()) {
            Class cls = (Class) it.next();
            name = cls.getName();
            parentType = UMLModelHelper.findJavaType(project, cls);
            break;
        }
    }
```

THE LIVE UML COMPANY

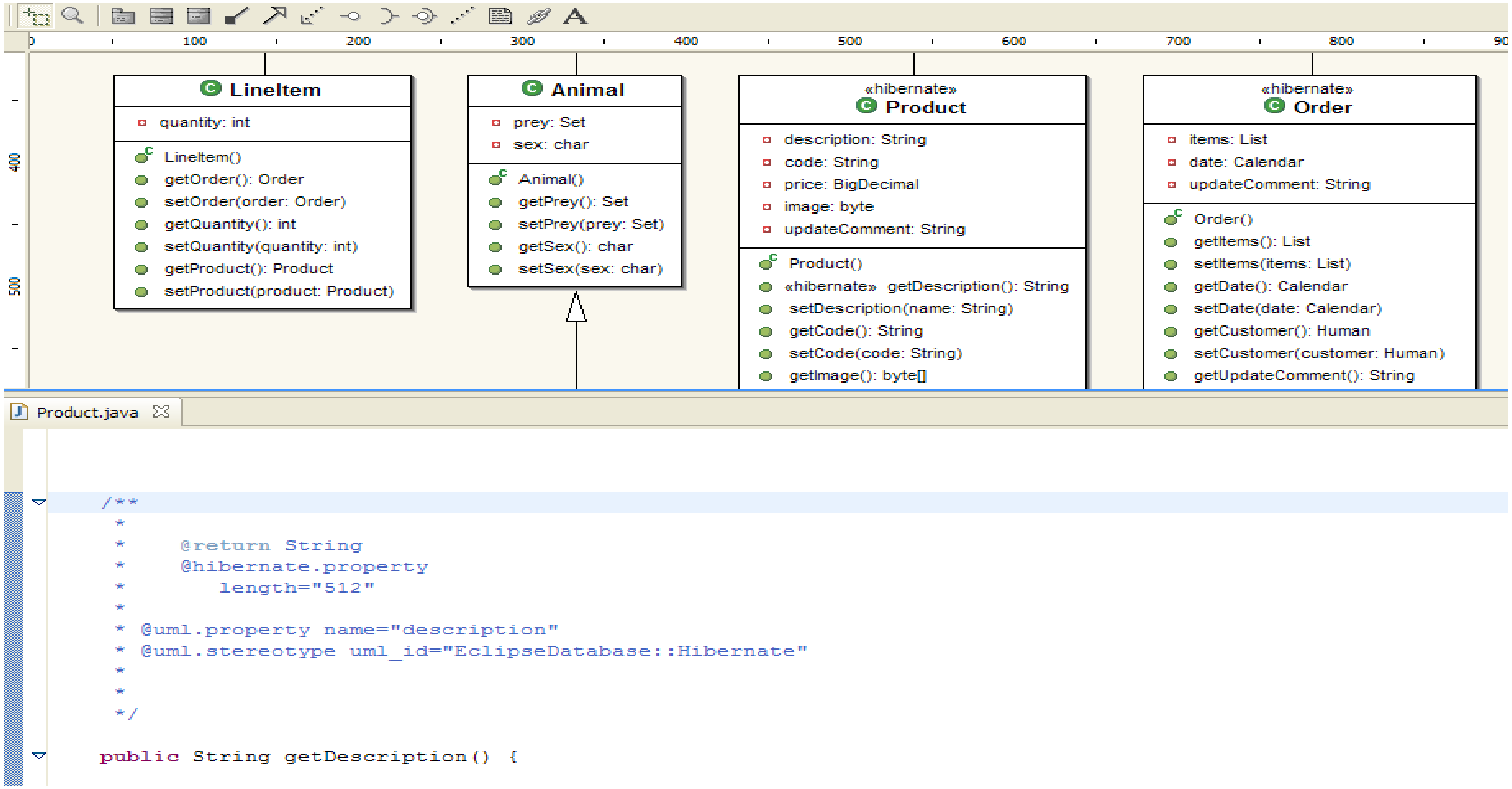
Code Cycle



```java
protected void registerDefaultContributionItems() {

    registerAction(
        new Action(DatabaseResource.getString(Constants.databaseHibernateXdocletAction)) {
            public void run() {
                XDocletHibernateJavaWizard wizard = new XDocletHibernateJavaWizard();
                // TODO Should be Fixed, UML Profile should provide an IWorkbenchPart and an IStructuredSelection
                wizard.init(null, null);
                // Instantiates the wizard container with the wizard and opens it
                WizardDialog dialog = new WizardDialog(DatabasePlugin.getActiveWorkbenchShell(), wizard);
                dialog.create();
                dialog.open();
            }
        },
        Types.CLASS_DIAGRAM
    );

}
```

THE LIVE UML COMPANY

Code Cycle



```
/**
 *
 *    @return String
 *    @hibernate.property
 *       length="512"
 *
 * @uml.property name="description"
 * @uml.stereotype uml_id="EclipseDatabase::Hibernate"
 *
 *
 */
public String getDescription() {
```

What's Next

- **More Tools**
  - Merge and Compare
  - Reverse Filtering
  - Alter Table
  - Stored procedures, Views
- **More UML Profiles**
  - EJB
  - JDO
  - …

What’s Next

- **JSR-220 (EJB 3.0)** defines annotations for ORM
  - JDK 1.5 based
  - No more mapping file
- **Two levels**
  - **Logical Level**
    - **@Entity, @ManyToOne, @OneToOne**

```
@Entity
public class Customer implements Serializable {
```

  - **Schema Level**
    - **@Table, @Column, @JoinColumn**

```
@Entity
@Table(name="CUSTOMER")
@SecondaryTable(name="CUST_DETAIL")
@JoinColumn(name="CUST_ID")
public class Customer { ... }
```

OMONDO® THE LIVE UML COMPANY

Reference

- http://db.apache.org/torque
- http://db.apache.org/ojb
- http://www.hibernate.org
- http://jakarta.apache.org/velocity
- http://www.eclipse.org
- http://www.eclipse.org/uml2
- http://www.eclipse.org/emf
- http://jcp.org/en/jsr/detail?id=220