Open Toolkit for UML Model
Driven Code Generation for the Enterprise

Andy Farrar
farrarm@saic.com
Problem Area

• **Enterprise systems are hard to build**
  – Too much time is spent building the middle-tier
  – Developers want to focus on user functionality
  – Experts are needed for design and implementation

• **Technology evolution makes selection of middleware difficult and costly**
  – Market moves to “newer” technology, leaves developers and clients with stagnate and under supported technology
  – Difficult to make cost effective use of the latest advances in technology
Problem Area

• **Support for Frameworks and Architectures**
  – Need for implementation of supporting methods

• **No easy way to experiment with new technology**
  – Enterprise Java Beans (EJBs)
  – CORBA services
  – COM components
  – Web services
  – Too much work required of developers to make use of services
Focus on capturing the maximum amount of metadata about the system within a UML meta-model and then USE it

- *Use all available metadata for code generation*
- *Don’t invent a new generation language*
- *Allow code generation to be open and extensible*
- *Use existing UML modeling tools*
  - *But don’t tie developers into a proprietary solution*
Components - Scenario

 Provides the Information Infrastructure

**LEGEND**

Our Component =  
COTS Component =  
Hand-Coded Component =  

**Components**

- Client
- Business Application Server
- Developer’s Source Files
- Generated Source Files
- Templates
- Metadata
- Code Generator
- Schema Server
- Model Editor
- Repository Adapters
- Metadata
- IDL
- XMI
- Java
- Information Repositories
  - Oracle, Sybase, etc
<table>
<thead>
<tr>
<th>Case</th>
<th>Number of Classes</th>
<th>Average Attributes per Class</th>
<th>Average Operations per Class</th>
<th>Lines of Code Generated</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>7</td>
<td>30</td>
<td>24</td>
<td>8,000</td>
</tr>
<tr>
<td>B</td>
<td>120</td>
<td>22</td>
<td>6</td>
<td>257,000</td>
</tr>
<tr>
<td>C</td>
<td>321</td>
<td>12</td>
<td>8</td>
<td>750,000</td>
</tr>
</tbody>
</table>

Note: Lines of code vary with the number of services enabled for that run of code generation.
Examples

• Generation of J2EE architecture class
  – Bean managed persistence to mapped tables
  – Generation of standard J2EE classes
  – Generation of deployment files

• Generation of CORBA infrastructures
  – Generation of methods required by many of the OMG services
  – Persistence to both object and relational databases

• Generation of Web Service and XML files
  – WSDL, Web Service connector code
  – XML Schema
  – Model to Model mapping generation

• Basically anything that can be defined by metadata
  – Design Patterns
Questions?