Implementing Model Driven E-Business Architectures using UML and XMI

UML in the DOTCOM Enterprise

Sridhar Iyengar

November 6-9, 2000 : Palm Springs, CA

sridhar.iyengar2@unisys.com

www.unisys.com
Global services and technology company
  – 1999 Revenues : $7.54 Billion
  – www.unisys.com

Services, Systems and Software for the enterprise

Use of metadata, modeling and repositories for software and systems integration
  – www.unisys.com/marketplace/urep

Committed to specifying and implementing open standards for enterprise software integration
  – OMG MOF, UML, XML, LDAP, XMI, CORBA, COM, EJB...
Presentation Overview

- E-Business Integration Challenges
- Relevance to EAI and E-Business
- A model driven E-Business Integration Architecture
  - OMG Modeling and Metadata Architecture
  - Application to Enterprise Integration
- Overview of MOF, XMI, UML, CWM
- UML Profile for EAI overview
- Putting it all together
Scope of Presentation

What this presentation about

– A new architectural approach to software integration
– Relevance of metadata, modeling and architecture not only for building but integrating applications
– Some early work in model driven integration at Unisys object technology labs and related OMG work
– Reinforcing the fact that evolving and managing software is a bigger problem than building it initially

What this presentation is NOT about

– Specific products even though I will mention some
– A complete methodology: we are still in the early stages of understanding the dynamics of the problem
– XML, UML, EJB, CORBA, COM, XMI in detail, even though we show the relevance to the overall architecture
e-Business Impact = Massive Complexity!

Changing Business Models

New Business Strategies

Agile Organization

Revolutionary Marketing

Customer First

Processes-Systems

Data

Value Chain Collaboration

Workflow

Transactions

Legacy

Integrated eMarkets

Knowledge

New Fulfillment

24x7 Operations

eCustomer Service

Real e-Business Touches Everything
Enterprise Architecture OR Architecture for the Enterprise

- **Enterprise**
  - ‘Global 1000 company with a large IT investment that is expected to be used to gain business advantage’
  - Virtual community of companies that have increased complexity!

- **Enterprise Architecture**
  - Most vendors and architects at ‘large’ customers think they have one
  - Vendor architectures are usually ‘marketectures’ rationalizing the products/technologies they have
    - Microsoft DNA, DOT.NET, IBM E-Business App Framework, SUN J2EE...

- **Architecture for the Enterprise**
  - Typically home grown painfully by pragmatic architecture teams using a variety of middleware technologies
  - Sometimes a lot worse
E-Business Is Key Driver

Back Office

E-business Strategic — ERP — Custom

Web Front-Office

… That Complicates The Picture!

- Unprecedented competitive threats
  - Nimble startups with no legacy
  - Unforeseen global business challengers
  - The rise, fall and re-emergence of e-businesses
- New Business Opportunities
  - Brand Leverage
  - Partner Leverage
  - New business models

Business Reaction Time is the real issue
Traditional Integration Approaches

- **Batch**
  - Export
  - Import

- **Near-time**
  - Send
  - Receive

- **Real-time**
  - Sync
  - Commit
… Do Not Scale!

- Business semantics are buried in code and not surfaced
- Implementations can not be easily replicated
- Little or no flexibility in dealing with change in corp. business models
- Point-to-point connections can not be multiplexed
What Is Needed?

- Realization that multiple approaches are necessary
  - To integrate business processes
  - To integrate operational data
  - To integrate transactions

- Acknowledgement that there is no silver bullet
  - Need a flexible architecture
  - Access to domain experts
  - IT commitment to use the architecture

- Realize that the higher level abstractions are needed
  - UML not just for development but for integration
  - Business process model based integration
Enterprises Will Need Multiple Approaches

- SAP
- PeopleSoft
- Oracle Apps.
- In-house
- Siebel
- Custom
- Manugistics

*Data Integration Products*

*Functional Integration Products*

*Process Integration Products*
**Mediation of Integration Complexity**

Difficult to manage  
Hard to understand

Much simpler design  
Easy to communicate

**A unified architecture that supports integration of tools, processes, data and content.**

**Key benefit:** Less intrusive adoption of new business models, more rapid uptake of new partners, reduced impact of new technology on the business and better leverage of existing systems
Use A Flexible Architecture

- A model and metadata driven Architecture
  - Grounded in UML and MOF
  - Metadata driven transformations
- Robust transformation engine
  - Captures business rules for data transformations
- Abstraction of the interface from the source(s) and target(s)
  - Transformations and connection rules apply to a generic description
- More complex transactions
  - Long duration transactions (business transactions) : OMG ETS
  - Recovery control
  - Able to describe backout actions in case of source/target failures
- Operational control
  - Interfaces become part of the standard operational monitoring and notification system
Integrate the changing picture of the web based business

- Global customer interaction
- Emergence of de facto standards such as XML, XMI, UML, HTTP, SOAP, J2EE, ebXML, UDDI
  - The ‘marriage of the week’, ‘standard of the week’ syndrome
- Focus shifted from “back-office” applications (ERP) to “front-office” (CRM) and now Portals and Vortals
- The need for secure Internet transactions
- And ultimately to a new paradigm shift towards e-Business
XML as the e-Business Integration Enabler

- Excellent fit for exchanging data; messages come with the description of the content -- “Metadata”
- Already adopted by the development community -- host of products on the market
- Endorsed by major players -- Unisys, SAP, PeopleSoft, Oracle, Microsoft, IBM
- Let’s developers focus on the content not the mechanism

Good News: Almost every one agrees!
XML Business-to-Business Example

- Bid management and order fulfillment
- Secondary supply chain
- Materials Request
- Orders
- Oracle Order Entry
- I2
- Preferred supplier
- SAP MRP
- PeopleSoft HRMS
- Vantive SFA
- XML over Internet
- XML based Integration
- Business Portal
- Web-based reseller
- Retail

Proprietary XML-RPC, SOAP, SCOAP....
If XML is so great why DON’T we have smooth integration today

- Technology is still new and evolving
- Vendor architecture wars moving to a new level!
- But there is real lack of methodology and discipline in how to use XML
- Garage DTDs - duplicating the same problems
- Plethora of XML frameworks that lack a cohesive architecture
- Some of the XML frameworks are just a bunch of XML DTDs on a web site
  - If we built airlines and roads this way, there would be gridlock everywhere!

- So what is the solution
  - Back to basics: Architecture, discipline, software engineering, modeling and software integration engineering
Using XML is the first real opportunity to allow applications to be connected without excessive support from systems integrator and domain experts.

The biggest benefit is speed of delivery which is by far the critical issue for eBusiness enablement.
**Elements of XML Based Integration**

- Capture Message semantics and structure in XML formats without being tied to a platform or technology
  - But be able to manage mappings to different technologies for legacy integration (XSLT, …)
- Easily manage mappings and transformations between XML messages
- Message “enrichment”
- Support major connectivity (COM, EJB, CORBA..) options with security
- Provide a high-performance and scalable run-time environment
  - And a productive business model driven development environment
- Security requirements such as X.500 integration, PKI, SSL, ActiveDirectory, JNDI...
Based on Unisys R&D in Business eXchange Server: How to Integrate integration frameworks

**Model Based Integration**

- Semantics (Relationships, Rules)
- Transformations and Mappings
- Workflow & Rules
- Core Services

- Captures the essence of the Business Processes
- Enables rapid adaptation to changing business models
- Rapid integration with new partner operational systems
- Inherent versioning effectively manages change
- Minimizes impact on changes to back-end systems
- Business rules can be changed and dynamically invoked at run-time

**Example sources**
- SAP
- Oracle Apps.
- COMS/TIP
- WebLogic (EJB)...

**Example targets**
- PeopleSoft
- Siebel
- COM/MTS
- COMS/TIP
- CICS
- BizTalk Server
Now back to Understanding how XMI ‘architecture enables’ XML

- Component Architecture builds on object architecture (more constrained)
- Object Architectures and Frameworks
  - White box reuse (Inheritance, Polymorphism…)
  - Patterns and Frameworks
- Component Architectures
  - Black box reuse (Encapsulation)
  - Components (ActiveX, JavaBeans, EJBs, CORBA Components)
  - Component Assembly and Deployment Management
OMG E-Business Integration Vision
for components, data, processes, domains

Community & Enterprise Information Portals (KM…)

HealthCare

Financial

Manufacturing

Insurance...

E-Business Application Development
UML, SPE

E-Business Intelligence
CWM, PM

E-Business Application Integration
CCM, EJB, COM, XML

Information Models, Components and Metadata (XMI, MOF…)

Directory, Security, Database, Web, Transaction, Caching, Metadata, Services..

Distributed Runtime Middleware (IIOP, XML/Value, SOAP…)

Copyright 2000, UNISYS Corporation

(C) Iyengar 2000
The E-Business Application Life Cycle and the Process

Architecture Centric, Business driven, Iterative and Incremental

Models, Metadata, Components, Middleware

Discovery and Transformation

Modeling Architecture, Object, Data...

Add Business Logic

Build / Wrap Components

Assemble & Test Components

Configure & Deploy Components

Component Runtimes (EJB, COM+)

Manage Component

Acquired Components

Business Requirements Modeling

Rigorous

WebTime!

And do this with quality in a distributed environment
OMG Modeling and Metadata Framework

- UML: Model & Design
- XMI: MOF2XML DTD, MOF2XML Doc, MOF2XML Schema*
- IDL: MOF2IDL
- JMI*: MOF2Java
- Meta Manage

* Coming

Vertical Industry Specifications
Data Warehousing
B2B Application Integration
Model Driven App Development
Meta Data Management
And So On
OMG Metamodeling Layers

Defined By

OMG Terms

User Objects Layer (M0)

Model Layer (M1)

Metamodel Layer (M2)

Meta-Metamodel Layer (M3)

Sample Objects

<Acme_Software_Share 98789>, 654.56, sell_limit_order, <Stock Quote Svr 32123>

StockShare, askPrice, sellLimitOrder, StockQuoteServer [Models, Interfaces, BODs]

UML::Class, Attribute, CWM::Table, CWM::Cube [UML, CWM, EAI, IDL…]

MOF::Class, MOF::Attribute, MOF::Operation [MOF]

XML Terms

User Objects
XML Documents
Business data

Business Models, DTDs
App Schemas

CWM, EAI, IDL, UML, DTD/Schema
Middleware Schema

MOF DTD/Schema
Schema of Schema!!
There is more & more metadata lurking everywhere!

<table>
<thead>
<tr>
<th>Name</th>
<th>Char</th>
<th>Meta Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Marital Status</td>
<td>Enumeration (M, D, W, S)</td>
<td></td>
</tr>
<tr>
<td>Age</td>
<td>Integer</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
</tr>
<tr>
<td>Mary</td>
</tr>
<tr>
<td>Bill</td>
</tr>
<tr>
<td>George</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Meta Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Employee Record</td>
</tr>
<tr>
<td>Last Name</td>
</tr>
<tr>
<td>First Initial</td>
</tr>
<tr>
<td>Employee Number</td>
</tr>
</tbody>
</table>
**Metamodel Architecture**

Heart of Integrating Metadata

MOF

M3 Layer: Specifies meta-metaclasses for the UML metamodel

M2 Layer: Specifies metaclasses for the UML metamodel, such as Class

M1 Layer: Specifies classes for the UML user models, such as Passenger, Ticket, TravelAgency

M0 Layer: User objects that are instances of UML user model classes, such as instances of Passenger, Ticket, TravelAgency
**MOF Overview**

- Foundation for OMG metadata and modeling architecture
  - Model, design and implement metamodels
  - Provides 4 generic meta-object interfaces for introspection (Used by all MOF based Meta-models...)
  - Provides MOF-IDL mapping to automate generation of concrete object interfaces for specific metamodels
  - Provides MOF-XML mappings to automate generation of XML DTDs and Documents (XMI specification)
  - Can also be used with COM using COM/CORBA Interoperability software

- MOF uses UML for notation and design
Meta Object Facility (MOF) Architecture

Discover & Manipulate metadata

Find and Manage Metadata Registries, Servers

Model using UML Class Diagrams precisely

<<OMG Meta-metamodel>>
MOF Model

<<CORBA IDL Module>>
MOF 1.1 Reflective

MOF Facility
Reflective module of the MOF
(Meta Programming/Modeling)

- All meta-models based on the MOF inherit this package.

- This is the foundation for introspection in the meta-model architecture.
**XML Metadata Interchange (XMI)**

- Use W3C Extensible Markup Language (XML) for the transfer syntax and interchange format
  - Specify XML Document Type Definitions (DTD) to enable transfer and verification of
    - UML based models (using UML DTD)
    - Data Warehouse models (using CWM DTD)
    - Java and EJB artifacts (using Java and EJB DTDs)
    - MOF based metamodels and their instances (using MOF DTD)
- Specify a precise MOF to XML mapping
  - Allows interchange of any MOF based metamodel and corresponding models (MOF--> XML Stream)
  - Enables automatic generation of XML documents that can be validated with generated DTDs
- Use UML and MOF for metamodel design and implementation
  - Designed by Unisys, IBM, Oracle, DSTC et al
XML Streams (Models)
(Many - based on each metamodel DTD)

XML DTD (MetaModels)
(1 per metamodel used for validation)

Validate

XML 1.0, 1.1 Simplified
OMG Documents : ad/99-10-02, ad/99-10-03

XML Syntax and Encoding
MOF Metadata Definitions & Management
UML Metamodel Analysis & Design

UML Models
CWM Models
Travel Document

UML DTD
CWM DTD
Travel DTD

XMI 1.0 : ad/98-10-05, 98-10-06
**XMI/MOF Tool/Application/Data Interoperability Options**

- MOF Introspection
  - write
- Model Transfer
- DTD Generation
- Schema Generation *
- (XMI)
- Concrete Model/Metadata Access
- * Coming in XMI

Generated Interfaces for Meta(model)

Intermediate Stream/File (XML)

MOF::Reflective

XML

Tool/App

Tool/App

EAI
EC
CWM
UML

* Coming in XMI
### A Simple Model

Let us pretend this is a trivial (meta)model. MOF and XMI work for models and metamodels.

<table>
<thead>
<tr>
<th>Business</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer</td>
</tr>
<tr>
<td>id : CustId</td>
</tr>
<tr>
<td>update()</td>
</tr>
</tbody>
</table>
<?xml version="1.0" encoding="UTF-8"?>

<!-- Document Prologue, etc. -->
<Model xmi.id="a1" name = "Business" visibility = "public">
  <ownedElement>
    <Class xmi.id="a7" name = "Customer">
      <feature>
        <Attribute name = "id">
          <multiplicity>
            <Multiplicity lower = "1" upper = "1"/>
          </multiplicity>
          <type>
            <DataType href="|a247"/>
          </type>
        </Attribute>
        <Operation name = "update" scope = "instance"/>
      </feature>
    </Class>
  </ownedElement>
</Model>
How to use XMI

- Define the domain or technology specific model
  - Middleware models are usually called metamodels by OMG
  - Use UML - only knowledge of Class modeling needed to get started on the design

- Export model to XMI processor which generates
  - XML DTDs for the specific (meta) model
  - XML documents that conform to the DTD

- Manage the Models, DTDs and documents in a MOF/XMI compliant distributed repository

- Use XMI toolkits from IBM, Unisys and others
**XMI - Automobile Example**  
*(simplified)*

**UML Model**

```plaintext
Auto
- Color : String
- Door : Integer
- Engine : Integer
```

**XMI Document**

```xml
<Auto>
  <Color>Red</Color>
  <Door>4</Door>
  <Engine>2</Engine>
</Auto>
```

**IDL, Java...**

```java
interface Auto {
    public String color;
    public int Door;
    public int Engine;
}
```

**XMI DTD, Schema**

```xml
<!Element Auto (Color*, Door*, Engine*)>
```
Use XMI for Integration
Examples of Initiatives

- Component based development and deployment scenario
  - Or how do you make e-business development and deployment tools work together

- Application Integration Scenario
  - Or how do you integrate e-business applications

- Data Warehouse and Content Management Scenario
  - Or how do you integrate the data in databases, file systems and the internet
The E-Business Application Life Cycle and the Process

Architecture Centric, Business driven, Iterative and Incremental

Models, Metadata, Components, Middleware

Rigorous WebTime!

And do this with quality in a distributed environment
XMI/MOF for E-Business Application Development OMG November 98 Demo

- WebSphere
- Rose
- VA TC
- DTD Gen
- VA Java
- IBM VisualAge
- Oracle Repository
- Oracle Designer
- XMI
- MOF/XMI
- Unisys UREP
- Rational Rose
- MOF DTDGen
- Select
- Enterprise
- Non-XMI Repositories

Community & Enterprise Information Portals (KML,...)
HealthCare
Financial
Manufacturing
Insurance...

E-Business Application Development UML, CCM
E-Business Intelligence CWM, OIM
E-Business Application Integration CCM, EJB, COM+, XML

Information Models, Components and Metadata (XML, MOF,...)
Directory, Security, Database, Web, Transaction, Caching, Metadata, Services...

Distributed Runtime Middleware (IIOP, HTTP,...)
Unisys UREP/ICF
Component Management Framework

Business, Component and Data Modeling
Visual Development Environments & Legacy Wrapping
EJB/COM+ Generation & Deployment
Web App Srvr or COM+
Component Management

Traceability Links & Transformations

UREP
XML
IDL
MOF
RSM
EJB
CWM
VCS

XML
JDBC
Oracle
Native

Metadata, Models, Components, code...

Copyright 2000, UNISYS Corporation
Travel Business Service Scenario
An E-Business Community

Customer
• requests Itinerary

BXS Travel
• interacts with multiple systems
• responds with Itinerary

BXS Travel Service:
Make Travel Arrangements

Customer

Customer Repository

Flight Reservation

Hotel Reservation
Component Based Design for E-Business (Service = Interface)

- In B2B integration frameworks the use of Services (ebXML, UDDI, BizTalk…) are common
- These can be designed using UML which is very effective for component based development
  - Class diagrams and Collaborations to design components
  - Interaction Diagrams and State machines for designing behavior
  - Activity Diagrams for modeling business processes
    - Some issues with the tight coupling of Activity diagrams and state machines are being addressed in UML 1.4+
- Model the document flows among B2B XML frameworks
Fragments of the ‘Travel Business Model’

- <BxsDocument>
  ItineraryRequestDocument
  - custName : string
  - pin : short
  - departureDate : string
  - departureCity : string
  - arrivalDate : string
  - arrivalCity : string
  - arrivalCountry : string
  - economyClass : string
  - accommodationsAreOptional : string
  - numberOfDays : string

- <BxsDocument>
  CustomerDocument
  - name : string
  - custNum : short
  - address : string
  - telephone : string
  - e_mail : string
  - category : short

- <BxsDocument>
  AccomodationRequest
  - checkInDate : string
  - numberOfDays : short
  - city : string
  - country : string

- <BxsDocument>
  AccomodationDocument
  - hotel : string
  - type : short
  - station : string
  - telephone : string
  - url : string

- <BxsDocument>
  FlightRequestDocument
  - departureDate : string
  - departureCity : string
  - arrivalDate : string
  - departureCountry : string
  - arrivalCity : string
  - arrivalCountry : string

- <BxsDocument>
  FlightDocument
  - departureDate : string
  - arrivalDate : string
  - flightNumber : string
  - charge : short
  - economyClass : string

---

Copyright 2000, UNISYS Corporation
Building the Travel Community Model
Building the Community Model

Client Interface

- ItineraryRequest Document
  - GetCustInfo : Transform
    - CustomerInfoReq Document
      - CustomerDocument
        - Customer Repository Interface

- FlightInfo : Transform
  - FlightRequestDocument
    - FlightDocument
      - Flight Reservation Interface

- HotelInfo : Transform
  - HotelRequestDocument
    - HotelDocument
      - Hotel Reservation Interface

- ReturnInfo : Transform
  - TravelScheduleDocument
    - Process
      - MakeTravelArrangements
Business Process Meta Model (partial)

Flow of Control

Stimulus, in context of BP Instance
The Credit Check Process

UML State Machine, OAG XML DTD
XMI: Travel DTD

<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE DTD/>

<!-- MODEL CLASS: ItineraryRequestDocument -->

<element ItineraryRequestDocument.departureCountry (PCDATA) >
<element ItineraryRequestDocument.economyClass (PCDATA) >
<element ItineraryRequestDocument.accomodationsAreOptional (PCDATA) >
<element ItineraryRequestDocument.custName (PCDATA) >
<element ItineraryRequestDocument.departureCity (PCDATA) >
<element ItineraryRequestDocument.arrivalCountry (PCDATA) >
<element ItineraryRequestDocument.arrivalDate (PCDATA) >
<element ItineraryRequestDocument.arrivalCity (PCDATA) >
<element ItineraryRequestDocument.pin (PCDATA) >
<element ItineraryRequestDocument.departureDate (PCDATA) >
<element ItineraryRequestDocument.numberOfDays (PCDATA) >
<element ItineraryRequestDocument (ItineraryRequestDocument.departureCountry,
   ItineraryRequestDocument.economyClass,
   ItineraryRequestDocument.accomodationsAreOptional,
   ItineraryRequestDocument.custName,
   ItineraryRequestDocument.departureCity,
   ItineraryRequestDocument.arrivalCountry,
   ItineraryRequestDocument.arrivalDate,
   ItineraryRequestDocument.arrivalCity,
   ItineraryRequestDocument.pin,
   ItineraryRequestDocument.departureDate,
   ItineraryRequestDocument.numberOfDays,
   XML.extension) />

<attlist ItineraryRequestDocument
   %XML.element.att;>
<XMI version="1.0" encoding="utf-8">
<!DOCTYPE XMI (View source for full doctype... )>
  <!ENTITY XMI version="1.0"> 
  <!ENTITY XMI header> 
  <!ENTITY XMI model xmlns="Travel" xmi.version="1.3"> 
  <!ENTITY XMI metamodel xmlns="Uml" xmi.version="1.3"> 
  <!ENTITY XMI metamodel xmlns="model" xmi.version="1.3"> 
  <!ENTITY XMI content> 
  <!ENTITY itineraryRequestSignal xmlns="b.1"> 
  <!ENTITY itineraryRequestSignal id="0.1"> 
  <!ENTITY itineraryRequestDocument xmlns="a.1"> 
  <!ENTITY itineraryRequestDocument id="b.1"> 
  <!ENTITY itineraryRequestDocument departureCountry="USA"> 
  <!ENTITY itineraryRequestDocument departureCity> 
  <!ENTITY itineraryRequestDocument arrivalCountry="USA"> 
  <!ENTITY itineraryRequestDocument arrivalCity> 
  <!ENTITY itineraryRequestDocument economyClass="yes"> 
  <!ENTITY itineraryRequestDocument economyClass="yes"> 
  <!ENTITY itineraryRequestDocument accommodationsAreOptional="yes"> 
  <!ENTITY itineraryRequestDocument accommodationsAreOptional="yes"> 
  <!ENTITY itineraryRequestDocument customName="Mr. Cust"> 
  <!ENTITY itineraryRequestDocument customName="Mr. Cust"> 
  <!ENTITY itineraryRequestDocument departureCity="Miami"> 
  <!ENTITY itineraryRequestDocument departureCity="Miami"> 
  <!ENTITY itineraryRequestDocument arrivalCity="Las Vegas"> 
  <!ENTITY itineraryRequestDocument arrivalCity="Las Vegas"> 
  <!ENTITY itineraryRequestDocument pinn=1100> 
  <!ENTITY itineraryRequestDocument pin=1100> 
  <!ENTITY itineraryRequestDocument departureDate="000125"> 
  <!ENTITY itineraryRequestDocument departureDate="000125"> 
  <!ENTITY itineraryRequestDocument arrivalDate="000125"> 
  <!ENTITY itineraryRequestDocument arrivalDate="000125"> 
  <!ENTITY itineraryRequestDocument numberOfDays="1"> 
  <!ENTITY itineraryRequestDocument numberOfDays="1"> 
  <!ENTITY itineraryRequestRequestSignal id="0.1"> 
  <!ENTITY itineraryRequestRequestSignal id="0.1"> 
</XMI>
**XML Based e-Business EAI**

Based on Unisys R&D in Business eXchange Server: How to Integrate integration frameworks

---

**Model Based Integration**

- Captures the essence of the Business Processes
- Enables rapid adaptation to changing business models
- Rapid integration with new partner operational systems
- Inherent versioning effectively manages change
- Minimizes impact on changes to back-end systems
- Business rules can be changed and dynamically invoked at run-time

---

Example sources
- SAP
- Oracle Apps.
- COMS/TIP
- WebLogic (EJB) ...

Example targets
- PeopleSoft
- Siebel
- COM/MTS
- COMS/TIP
- CICS
- BizTalk Server

---

Source1

Source2

Target1

Target2

**Core Services**

- Semantics (Relationships, Rules)
- Transformations and Mappings
- Workflow & Rules

**Business Process Modeler**

**XML Handler**

**COM CORBA EJB TIP**

---

**Unisys**

Copyright 2000, UNISYS Corporation
Semantic integrity of BXS Repository:

- **Model Interchange Services:** Common services for Design tools
- **BXS Runtime Engine:** Dynamic execution of UML/BXS Model
- **XMI:** externalization of model elements, per xmi Standard
Integrating XML, COM, CORBA, EJB and Legacy: Adapters

BXS Runtime Engine

send(Signal, source, target)

Routes by Component type

EJB 1.1 Application Server
context: Repository

BXS DCOM CORBA etc OAG

Legacy Access
• Handles Component type
• Converts external representation to/from M0 XMI (Signal Instance)

Physical Implementation
• Determined by Instance Information
UML Profile for EAI

- OMG RFP for introducing architecture modeling discipline into EAI
- Initial submissions at www.omg.org
- Key players: IBM, Oracle, Unisys, Rational, Concept 5
- Focus: Event based application integration architectures
- UML profile and Metamodels for EAI
- Expected to be standardized 1Q2001
UML Profile for Enterprise Distributed Object Computing (EDOC)

- UML profile for CORBA: Completed
- UML profile for EJB: Final stages in Java Community Process
- UML profile for EDOC
  - Business model and process based integration that supports multiple middleware architectures
  - ‘Enterprise Collaboration Architecture’
  - To be completed 1H2001
- UML profiles for EDOC and EAI to be integrated
XML/XMI for Application Integration and Industry Specific Domains

- OMG XML/Value (2000)
- UML Profile for EAI, I-EAI (2000)
- Integration of OMG and OASIS Technologies/Frameworks
  - XML/XMI Registry Discussions underway with ebXML
  - UDDI is another factor
- Knowledge Management
- OMG Healthcare, Manufacturing, ECDTF…
- Most EAI vendors using/planning XML for B2B
OMG E-Business Integration Vision
for components, data, processes, domains

Community & Enterprise Information Portals (KM...)

HealthCare  Financial  Manufacturing  Insurance...

E-Business Application Development
UML, CCM

E-Business Intelligence
CWM

E-Business Application Integration
CCM, EJB, COM, XML

Information Models, Components and Metadata (XMI, MOF...)

Directory, Security, Database, Web, Transaction, Caching, Metadata, Services..

Distributed Runtime Middleware (IIOP, XML/Value, SOAP...)

Copyright 2000, UNISYS Corporation
(C) Iyengar 2000
Importance of (Meta)modeling using MOF, XMI and UML

- Provides a regular and rigorous infrastructure at a higher level of abstraction
- Furnishes an architectural basis for extensions and evolution of software
- Facilitates alignment with other standards that use a metamodel architecture
  - Potentially eliminate or evolve redundant standards (e.g., CDIF now endorses XMI)
- Supports interoperability and integration across domains at the semantic level
- Use UML to design metamodels and models, MOF to implement and manage them and XMI to interchange them over the Internet
Who is Implementing XMI, MOF, UML and CWM?

- IBM VisualAge for Java, WebSphere, Rose tool kit
- Unisys UREP, Component Management Server, Transform…
- IBM VisualWarehouse, Hyperion, Oracle, SAS
- Rational Software (Unisys XMI for Rose)
- DSTC, OMEX…
- Oracle Designer, Meta Integration, Together/J, Objecteering, ObjectsByDesign, Aonix…
- OMG standard metamodels and DTDs (MOF, UML, CCM, CWM*, Java*, EJB*…)
- Metadata Coalition voted overwhelmingly to build on OMG specifications for metadata (September 2000)
- Unisys and IBM and others for EAI and B2B Integration
- Java Community Process JSR-40 : Java Metadata Interface*

* In progress
OMG UML/XMI/MOF Usage

- Submissions
- EJB
- Java
- SPE
- UML Profile for EDOC...

- Evaluating
- MDC OIM
- CIAS
- Enterprise App Integration
- Document Management
- Etc.

- Standards
- MOF
- UML
- CCM
- IDL UML4CORBA
- CWM

- CORBA Med
- Life Sciences
- Electronic Commerce
Concluding Thoughts

- Ensuring a *unified model driven distributed architecture* is key to solving the heterogeneous integration problem.

- Most customers have and will continue to have components and information/data from multiple sources and formats that need integration.

- XML/HTTP, CORBA/EJB, DCOM/ActiveX, MOF/XMI/XML (Metadata, Services), UML (Modeling) and CWM (Data Warehousing) need to work together with the content (business models, BODS, HL7 RIM…)
  - UML4EAI is first step to standardizing ‘Integration by Modeling’

- Modeling and Metadata matters - Master it : Use it in for your middleware, use it for your domains

- Gain the benefits of component development and integration by building on UML, MOF and XMI
For More Information

- Unisys : www.unisys.com/marketplace/urep
- W3C : www.w3c.org
- DSTC : www.dstc.edu.au
- Sridhar : sridhar.iyengar2@unisys.com
- UML RTF : uml-rtf@omg.org
- MOF RTF : mof-rtf@omg.org
- XMI RTF : xmi-rtf@omg.org
- CWM RTF : cwm-rtf@omg.org