Session ID #A44

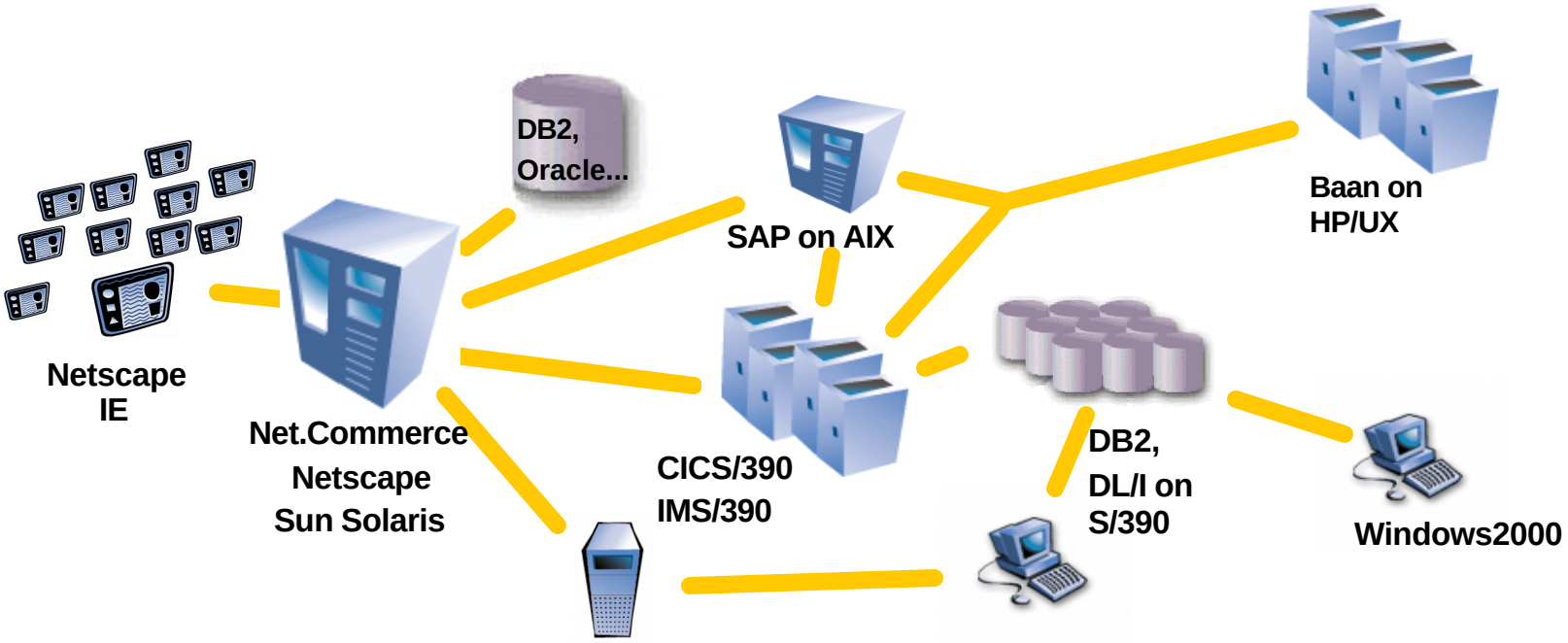
IBM's XML Solution for Enterprise Application Tools and Web Services

Shyh-Mei F. Ho
IMS e-business
shyhmei@us.ibm.com

Miami Beach, FL October 22-25, 2001

RETURN TO INDEX

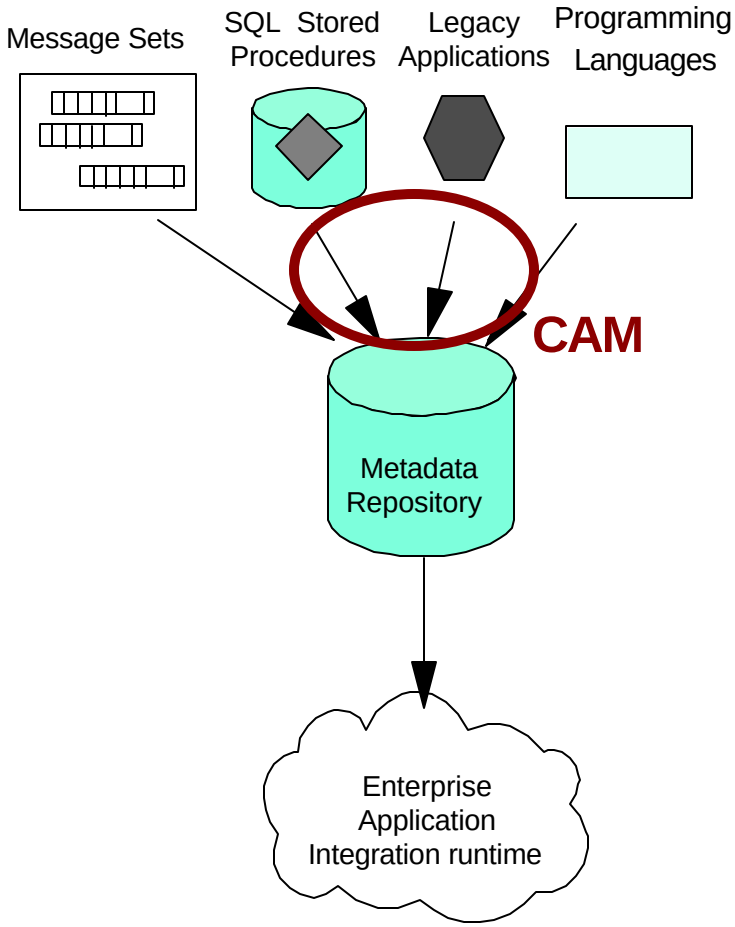
e-Business Application and Problem



- **Where is the application in this picture?**
  - Everywhere!
- **How is the application assembled?**
  - With Connectors!
- **It is not easy to integrate or e-business enable the applications**
  - No program accessible interface

Common Application Metamodel (CAM)

- **Solution: CAM defines a standard for accessing enterprise applications defined in proprietary structures, including 3270 screen formats**
  - OMG standardization final submissions
  - CAM allows to develop tools that analyze existing enterprise applications and describe applications as metadata in a format that can be exchanged among tools
- **CAM simplifies application integration for invoking and translating application information**
- **CAM enables Web services for enterprise applications**

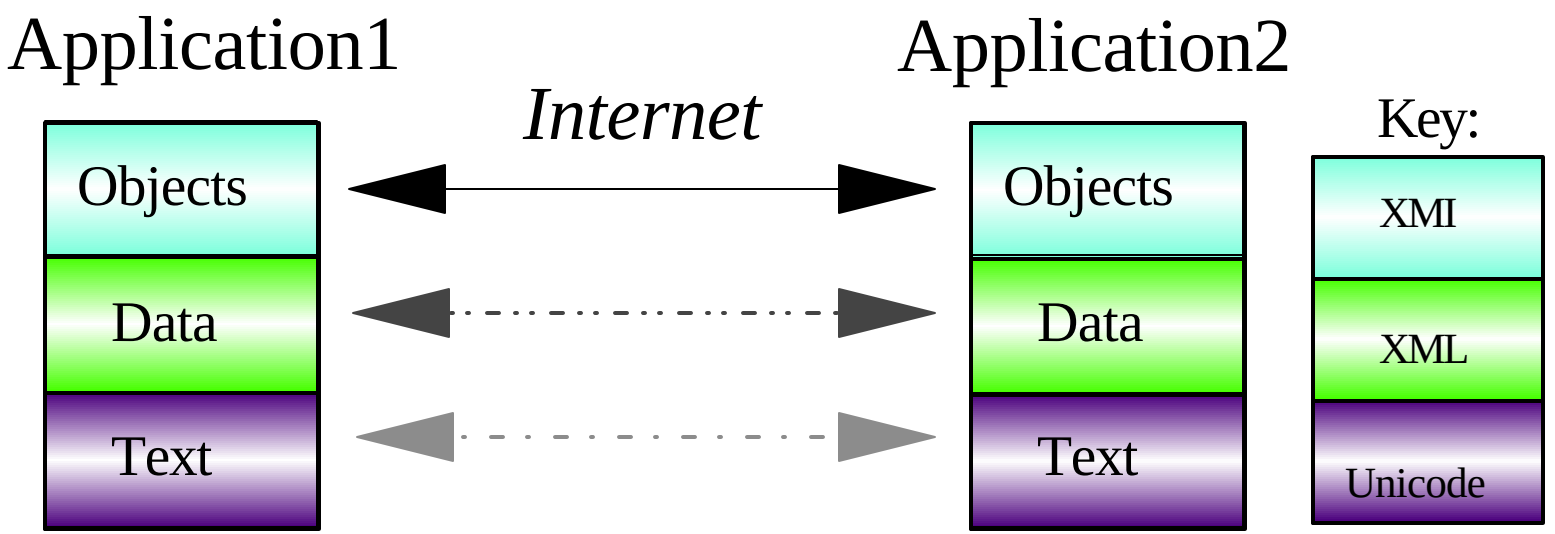


XML & XMI in a Nutshell

- **XML (eXtended Markup Language)**
  - an interchange standard for B2B application and workflow integration
  - user defined tags
- **XMI (XML Metadata Interchange)**
  - object-oriented modeling using UML (Unified Modeling Language)
  - based on object model to generate DTD (Document Type Definition) or schema which defines consistent structure and arrangement of tags in an XML document
  - XMI leadership in OMG (Object Management Group)
- **XML document based on XMI DTD or schema**
  - a document that contains XML tags formatted according to the rules described by the XMI DTD or schema

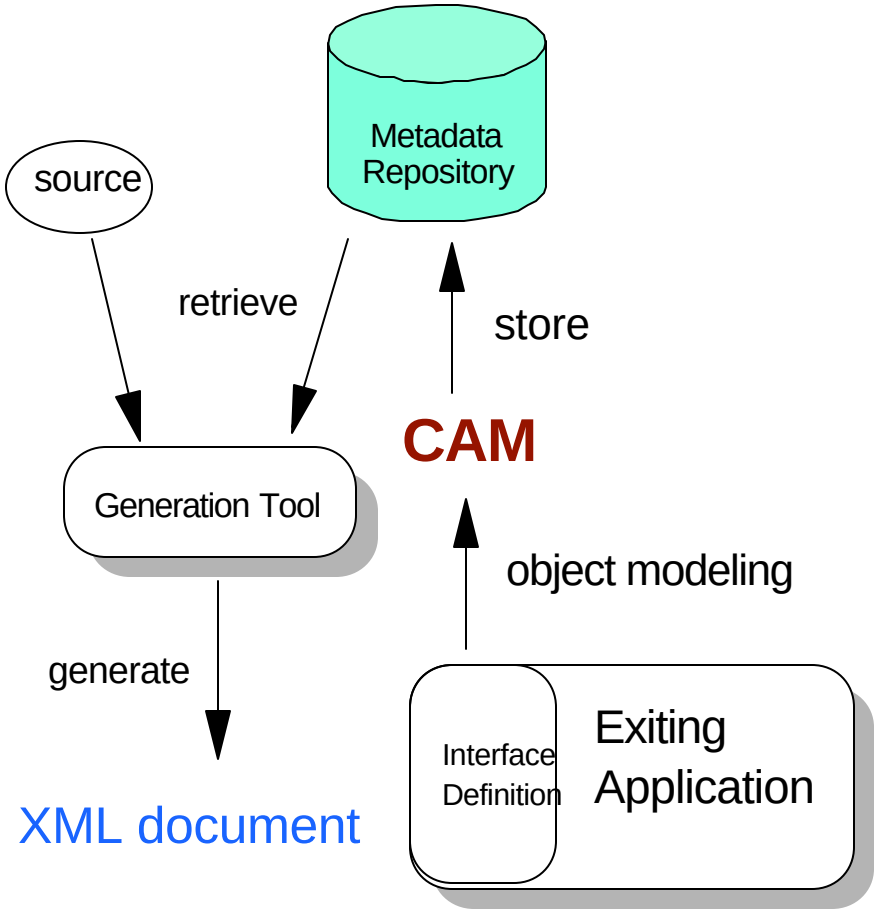
Sharing Objects

- XML - Sharing Data
- XMI - Sharing Objects
  - Creates custom XML DTDs/Schemas and XML documents matching Object Designs.



Modeling for CAM

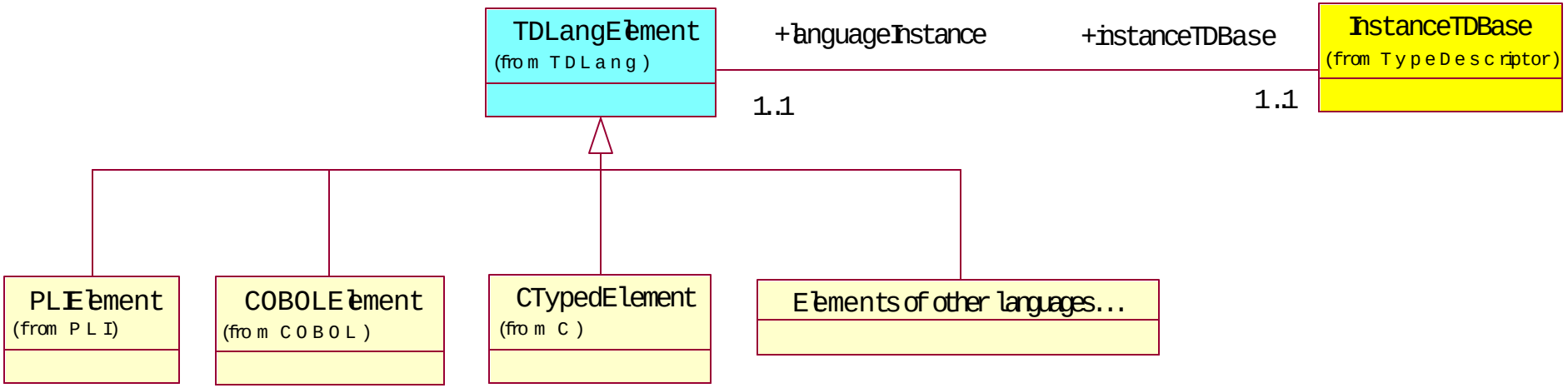
- **Start with object modeling**
  - Don't design or manually code XML DTD or schema directly
  - Instead, use UML (Unified Modeling Language) class design mechanism to model application interface definitions
- **Use tool to generate XML interchangeable DTD files or schemas for application interface metamodels**
- **Then apply mapping**



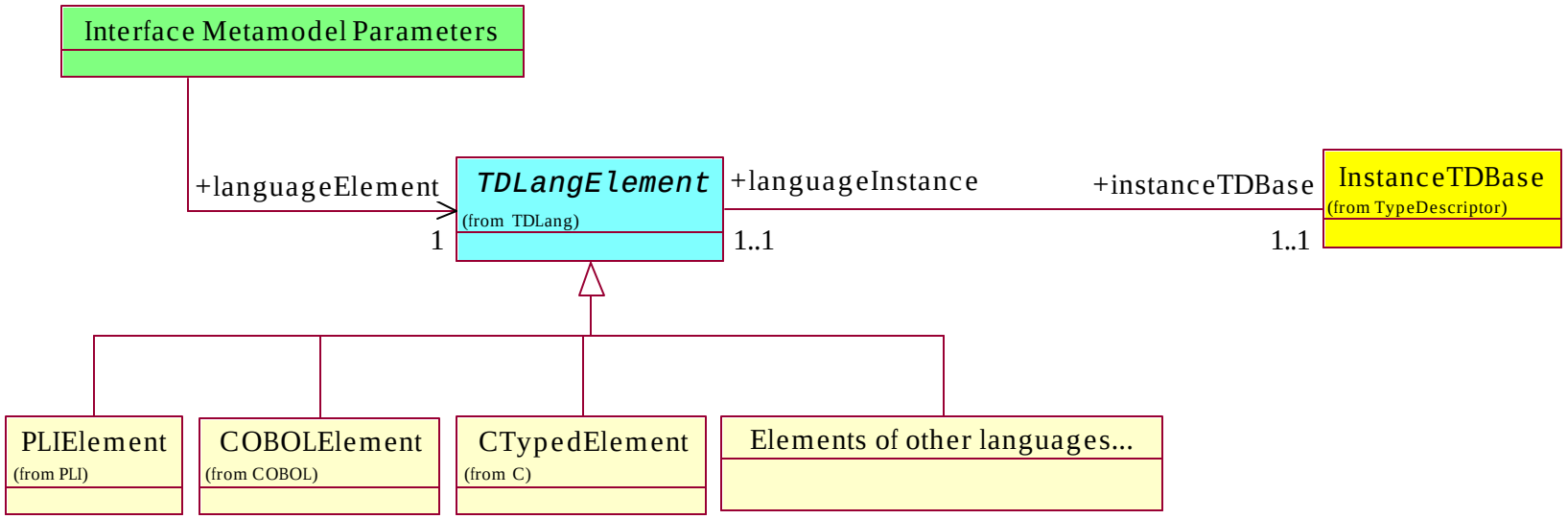
Language and Physical Representation Models

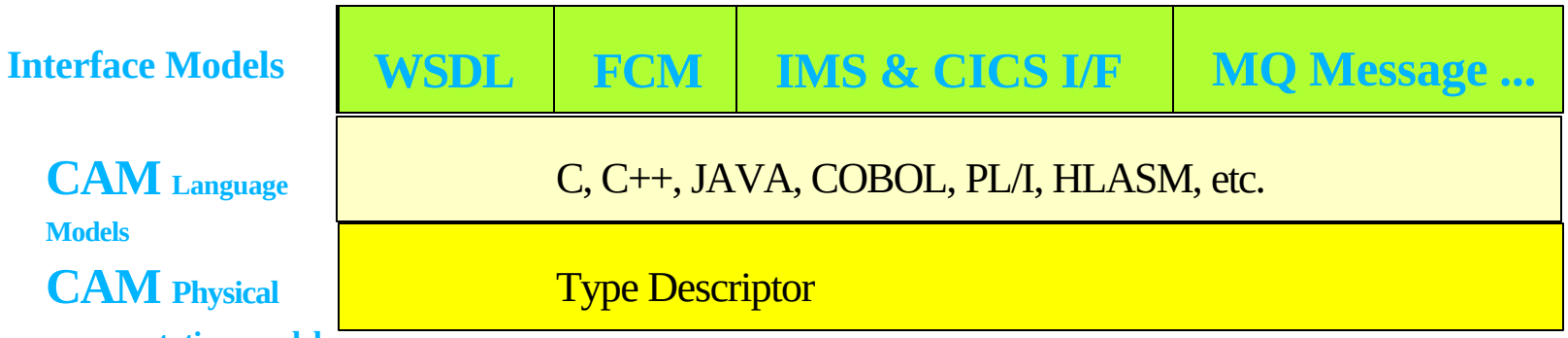
- **Language metamodels**
  - Programming languages used by enterprise application programs to define data structures which represent interfaces for connectors
- **TypeDescriptor metamodel**
  - presents a language and platform independent way of describing implementation types, including arrays and structured types
  - this information is needed for marshaling and for connectors, which have to transform data from one language and platform domain into another
  - an instance of TypeDescriptor describes the physical representation of a specific data type for a particular platform and compiler
- **TDLang metamodel**
  - base classes to language models
  - type target for assoication from TypeDescriptor to language metamodels

Language and Physical Representation Models



CAM and Interface Models

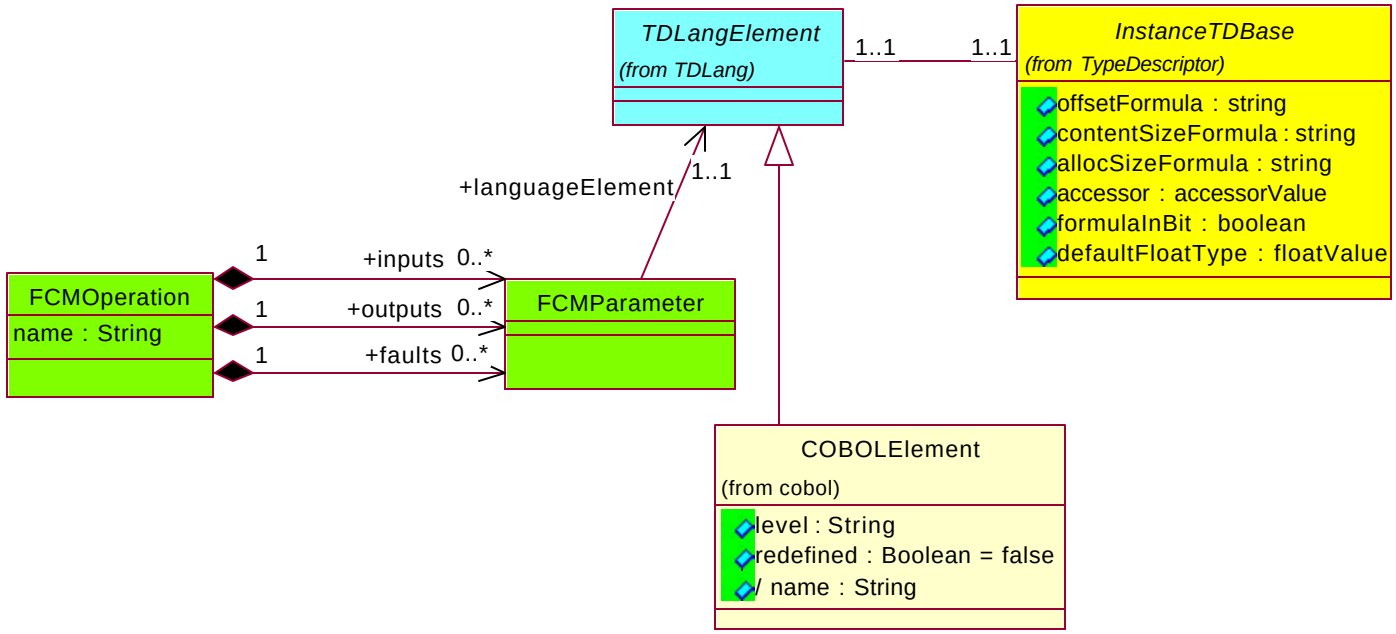


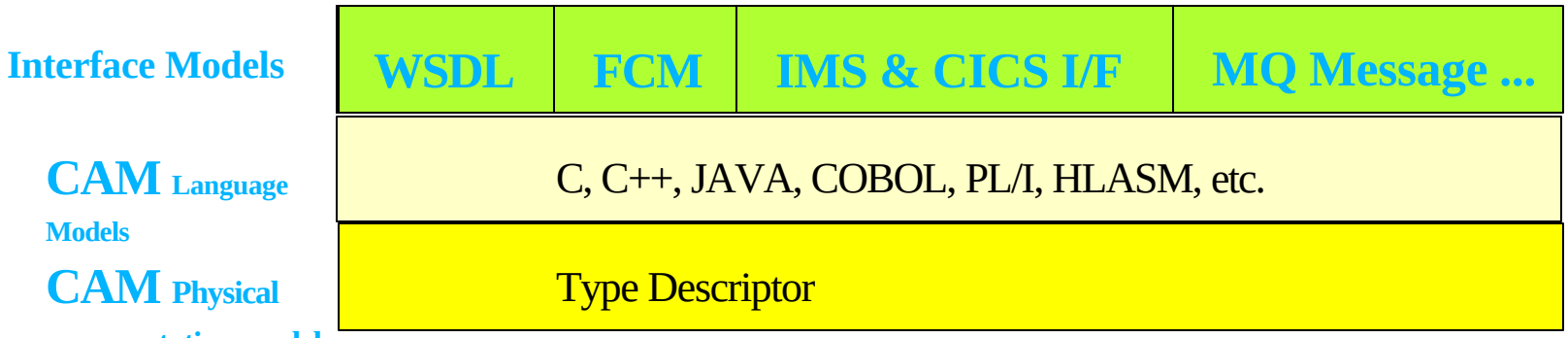


Flow Composition Model (FCM)

- **FCM from the UML for EDOC (ad/2001-06-09) describes the interactions and flows of information between application components**
  - enable complex actions to be broken down into simple "flow components" or
  - alternatively, enable simple entities to be composed into higher level "flow models"
  - can be deployed into a variety of runtime environments
- **FCM composes complex flows based on invoking operations on components**
- **FCM needs CAM to define the signature (input and output) of aplications to provide language specific and wire formats for component inputs and outputs, and enables connectors and transformers to be defined and built so that data can flow properly between components**

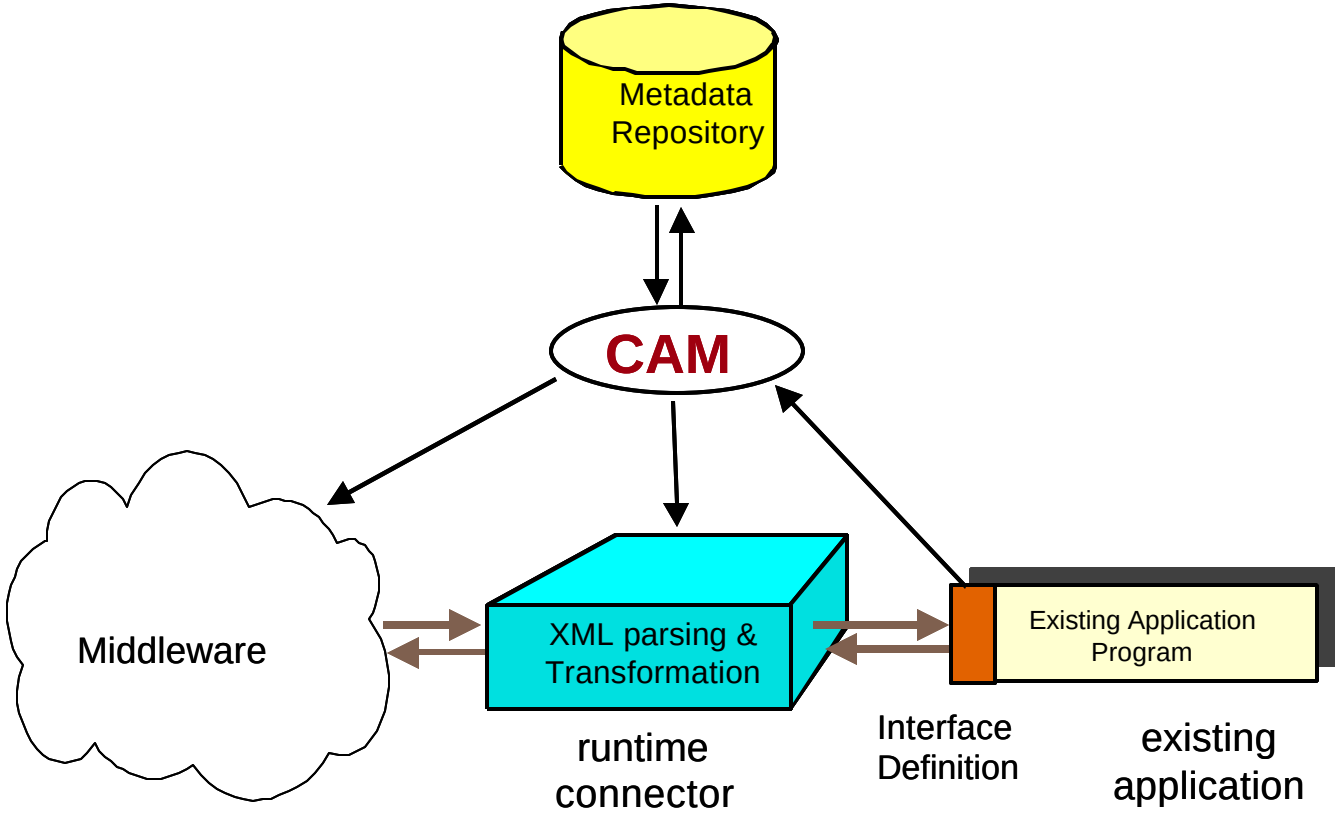
CAM and FCM Model



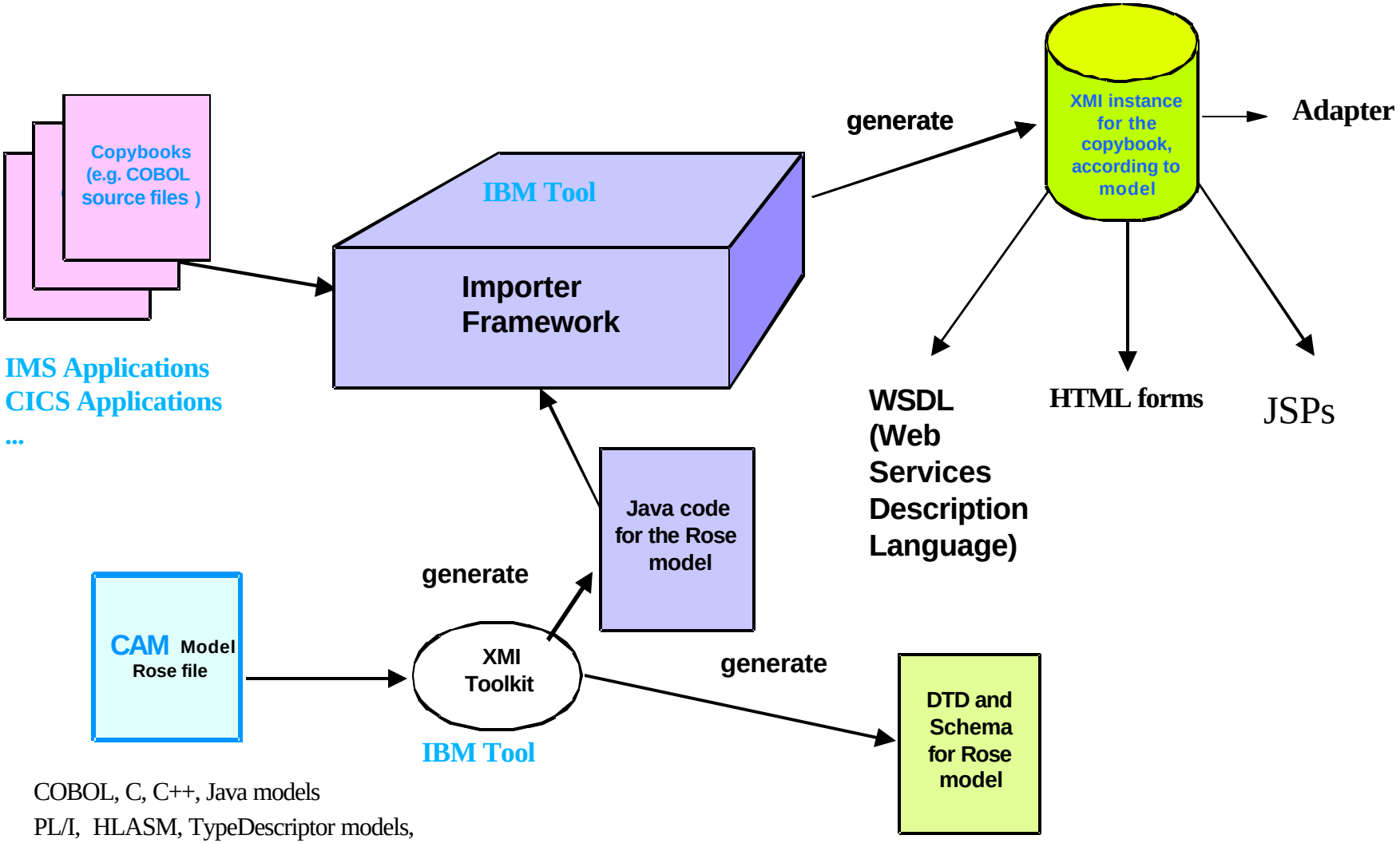


CAM Deployment

- **CAM is needed for data transformation in an enterprise application integration environment**
  - Provides data type mapping between mixed languages
  - Facilitates data translations from one language and platform domain into another



CAM and IBM Tools Support (directions)



COBOL, C, C++, Java models
PL/I, HLASM, TypeDescriptor models,
MFS , BMS models

CAM and Tools Support (directions)

- **CAM provides the underpinning for enterprise tools and connectors/adapters that perform connections and transformations**
  - importers
    - COBOL, C, PL/I, and MFS etc.
  - adapters
    - COBOL, C, PL/I, and MFS etc.
    - adapters for Java environment, i.e. WAS J2EE
    - non-Java adapters for IMS and CICS applications or gateway
  - microflows
    - MQ workflows
    - IMS and CICS conversational and pseudo-conversational transactions
  - Web services tool for enterprise applications

CAM and OMG Standardization

- **Industry standards for metadata is critical**
- **OMG is a recognized industry standards organization**
- **IBM is leading the CAM efforts in the OMG Enterprise Application Integration (EAI) submission (ad/2001-08-02)**
  - Joint submitters:
    - IBM, DSTC, Hitachi, Rational, Oracle, Unisys
  - Supporters:
    - Charles Schwab, Data Access, Consortium for Business Object Promotion (CBOP), and IONA
  - Assessors
    - Boeing, John Deere, InteliData, and Price Waterhouse Coopers
  - Continue aligning with other customers and vendors
  - The initial submission was done August 21, 2000
  - Revised document for review February, 2001
  - Final submission August 20, 2001

CAM and OMG Standardization ...

- **OMG's Enterprise Distributed Object Computing (EDOC) standard (ad/2001-06-09) is also based on CAM (9/13/01)**
  - CAM is part of the EDOC final submission (06/01)
    - Java metamodel (part of CAM)
    - EDOC adopted IBM's Flow Composition Model (FCM); and FCM's interface parameters use CAM
  - Joint submitters:
    - IBM, CBOP, Data Access Technologies, DSTC, EDS, Fujitsu, IONA, Open-IT, Sun Microsystems, and Unisys
  - Supporters:
    - Hitachi, SINTEF, NetAccount

CAM and Web Services Demo

Transform Existing Enterprise Applications into Web Services

Web Services

- **Web Services are the next step in the evolution of the WWW and allow programmable elements to be placed on web sites where other can access in distributed behaviors**
  - A provider of information or capabilities exposed on a network through a consistent set of interfaces and protocols
  - support heterogeneous environment seamlessly
- **Connect applications to applications in other businesses quickly and easily**
  - Focus on automation of development and deployment
- **Establish interaction with marketplaces more efficiently**
- **Deliver business functions to a broader set of customers and partners**
- **Pursue new business models by combining applications in new dynamic ways**

What is UDDI?

- **Universal Description, Discovery, and Integration**
- **A project to speed interoperability and adoption for web services**
  - Standards-based specifications for service description and discovery
  - UDDI registry implementation
    - public Web service registry and development resources
  - Partnership among more than 280 industry and business leaders, i.e. IBM, Microsoft, HP, Oracle and Sun Microsystems

Problems UDDI Solves

Broader B2B

A mid-sized manufacturer needs to create 400 online relationships with customers, each with their own set of standard and protocols

Smarter Search

A flower shop in Taiwan wants to be “plugged in” to every marketplace in the world, but doesn’t know how

Easier Aggregation

A B2B marketplace cannot get catalog data for relevant suppliers in its industry, along with connections to shippers, insurers, etc.

Describe Services

Discover Services

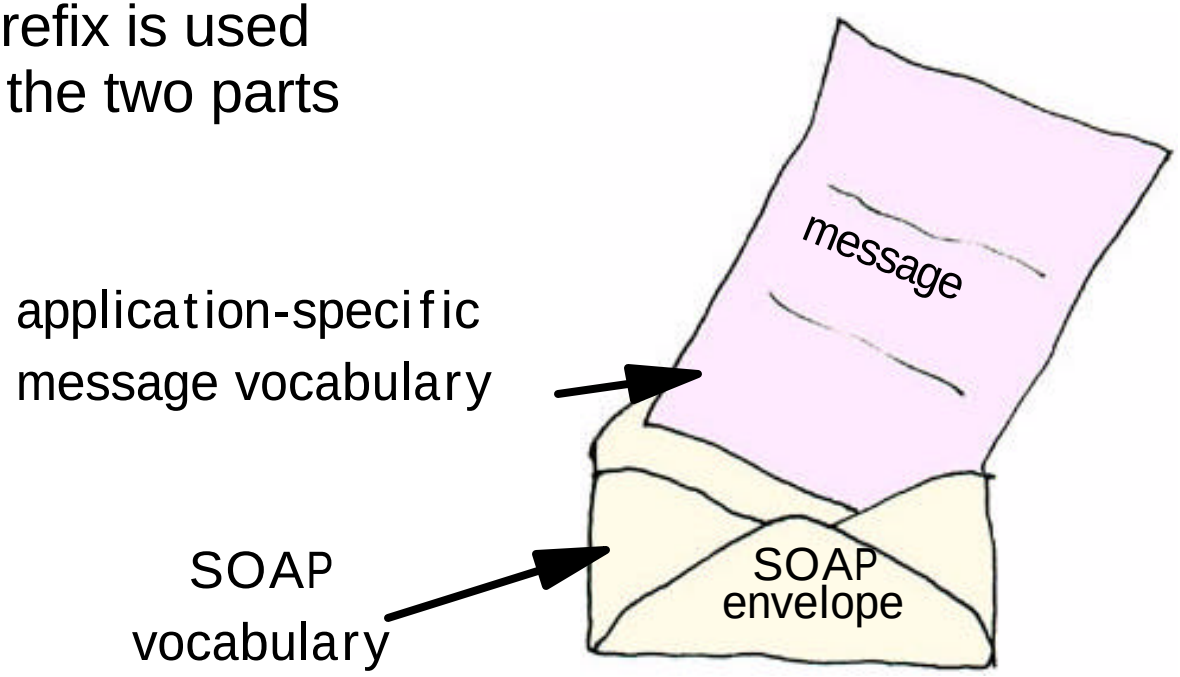
Integrate them Together

Simple Object Access Protocol (SOAP)

- **A SOAP message consists of**
  - Envelop - provides a frame around Header and the message body
  - Header - contains contextual information about messages
  - Body - contains the payload data
- **SOAP is based on one-way message transfer; XML envelope for requests and responses**
  - can be used for Remote Procedure Call
- **XML encoding for messages**
- **Binding to HTTP as the first transport layer protocol**
  - other transports are allowed
- **Loose coupling with endpoints**
  - no restrictions on the endpoint implementation technology choice
- **Submitted to W3C for consideration as a standard**
  - XML messaging protocol

SOAP Message Structure

- **Request and Response messages**
  - Request invokes a method on a remote object
  - Response returns result of running the method
- **SOAP defines an "envelope"**
  - "envelope" wraps the message itself
  - message is a different vocabulary
  - namespace prefix is used to distinguish the two parts



A SOAP Request Message

```
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://{soaporg}/envelope/"
  SOAP-ENV:encodingStyle=
       "http://{soaporg}/encoding/">

   <SOAP-ENV:Body>
        <m:GetLastTradePrice xmlns:m="Some-URI">
            <symbol>DIS</symbol>
        </m:GetLastTradePrice>
   </SOAP-ENV:Body>

</SOAP-ENV:Envelope>
```

message

SOAP envelope

A SOAP Response Message

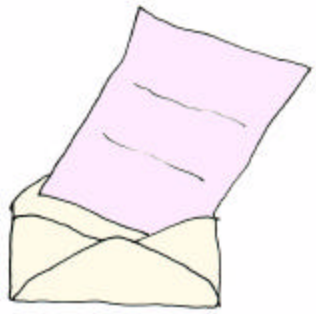
```
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://{soaporg}/
  SOAP-ENV:encodingStyle=
       "http://{soaporg}/encoding/">

   <SOAP-ENV:Body>
       <m:GetLastTradePriceResponse
                        xmlns:m="Some-URI">
          <Price>34.5</Price>
       </m:GetLastTradePriceResponse>
   </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

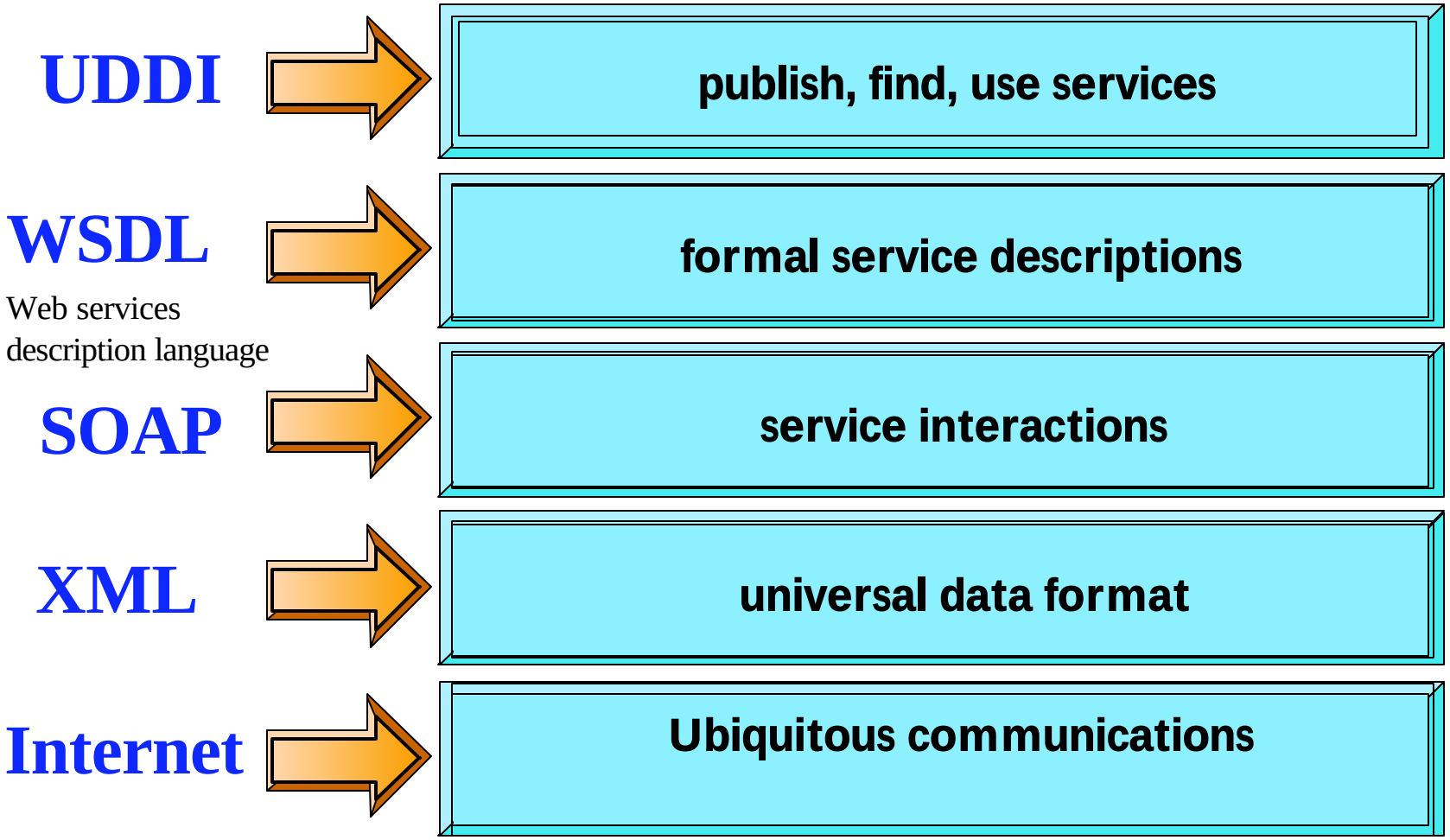
Result returned in Body

message

SOAP envelope



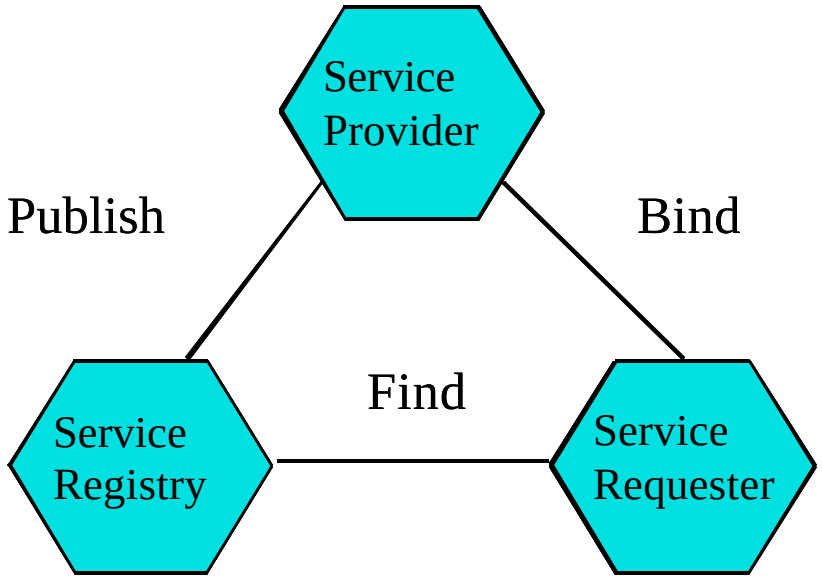
Web Services Fundamentals



Simple, Open, Broad Internet Support

Web Services Architecture

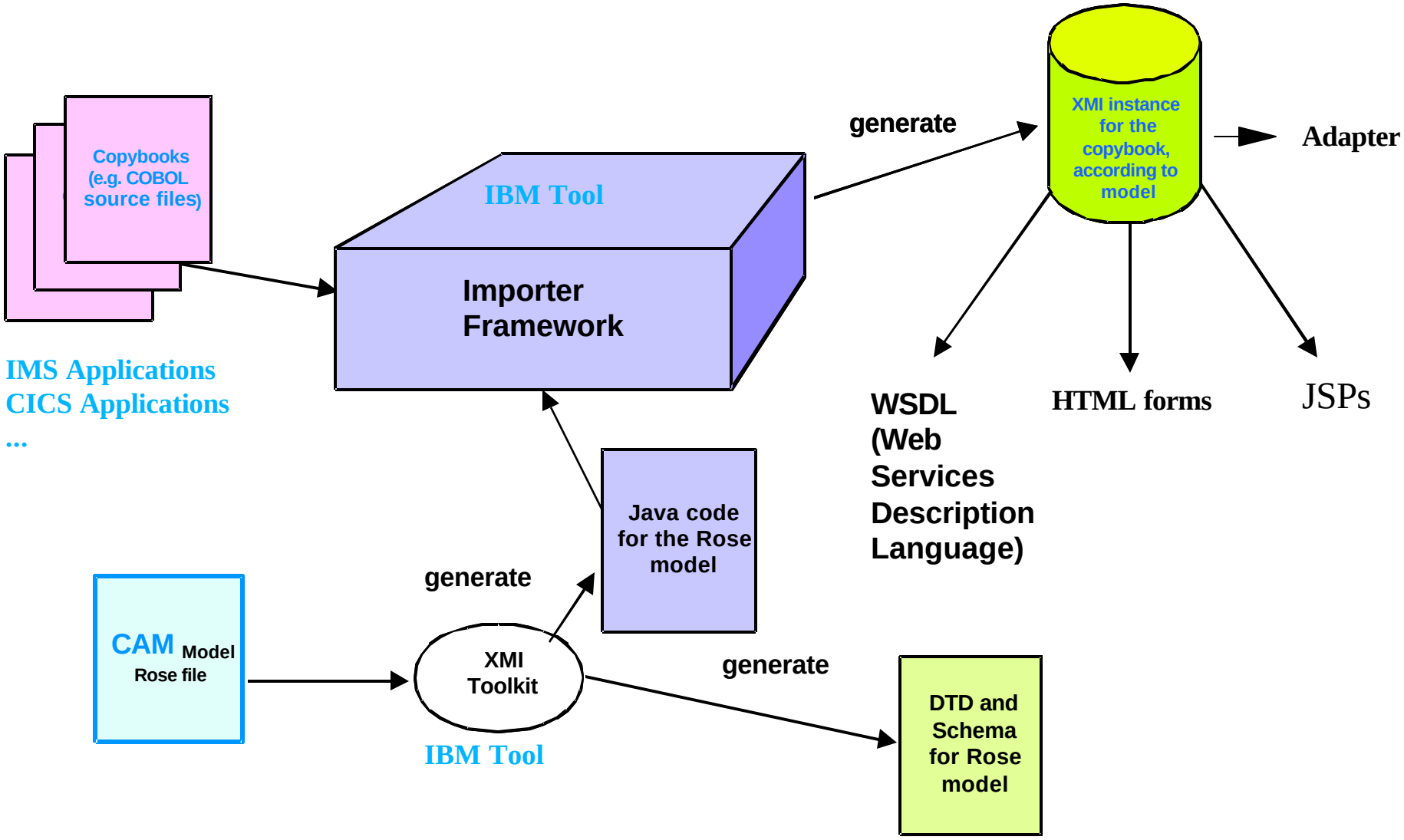
- Use WSDL (Web Services Description Language) to describe a service, i.e. business function
- Publish a service by registering its description with a registry, i.e. UDDI registry
- Find the business functions by sending queries to the registry and receiving binding information
- Invoke (Bind) the service using the binding information



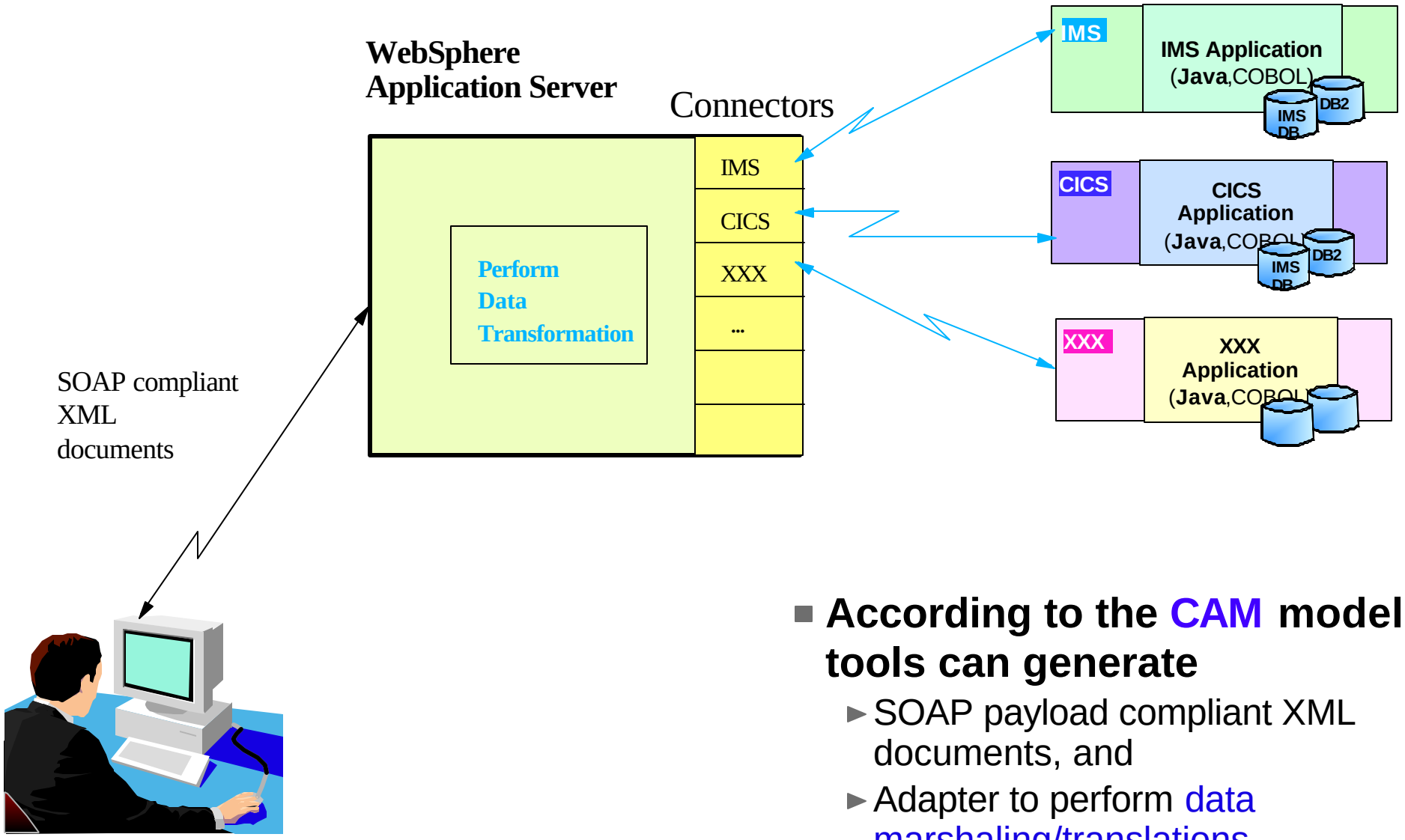
Examples of Web Services

- weather reports
- news feed
- airline schedules
- airline reservations
- rental car agreements
- credit check
- credit card validation
- request for quote
- supply chain management
- purchase order

Transform Existing Applications to Web Services (Development)



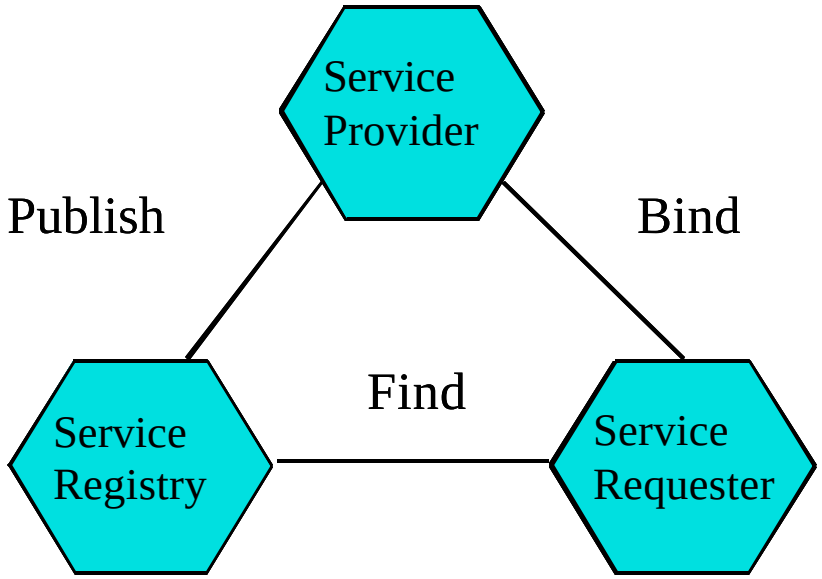
Transform Existing Applications to Web Services (Runtime)



- **According to the CAM model, tools can generate**
  - SOAP payload compliant XML documents, and
  - Adapter to perform data marshaling/translations

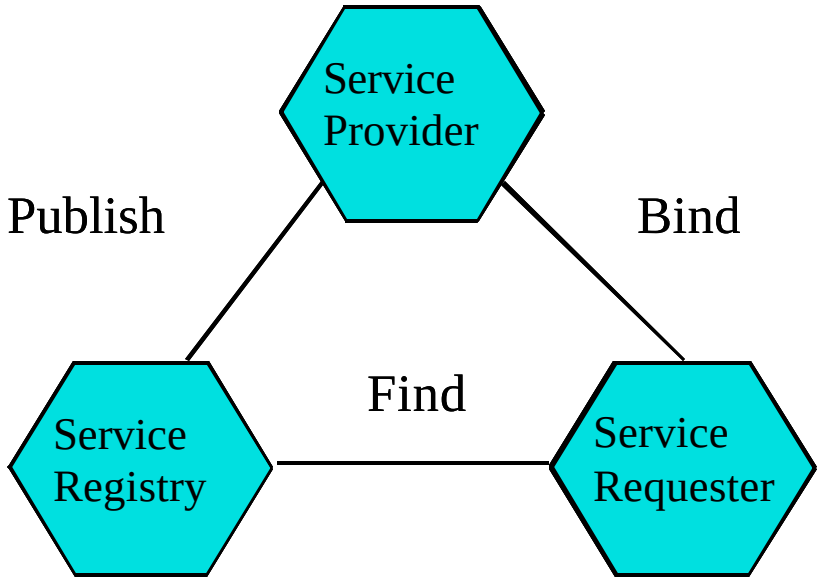
CAM Demo Page

- **A new IBM CAM web site home page is under construction which consists of a set of Web services**
  - PhoneBook
    - IMS HLASM transaction program
  - TraderDemo
    - CICS COBOL transaction program
  - StockQuote
    - non-IBM program



CAM Demo Page

- **A new IBM CAM web site home page is under construction which consists of a set of Web services**
  - PhoneBook
    - IMS HLASM transaction program
  - TraderDemo ⇨ published
    - CICS COBOL transaction program
  - StockQuote ⇨ published
    - non-IBM program



Summary

- **CAM is an IBM cross division project**
- **CAM provides a standard way to interchange information about enterprise applications**
  - OMG EAI and EDOC standardization final submissions
- **CAM is highly reusable and independent of any particular tool or middleware**
  - IBM and non-IBM tools can easily access enterprise applications (e.g. CICS and IMS)
  - IBM and non-IBM tools can access any CAM enabled applications
- **CAM is an incentive to connector suppliers**
- **IBM next generation tools framework is using CAM as its basic metamodel for enterprise applications and programming languages**
- **CAM is needed for data transformation in an enterprise application integration environment**
- **IMS XML transaction support is based on CAM**
- **CAM enables Web services for enterprise applications**

RETURN TO INDEX