11. Information Integration and Interoperability

INFO 202 - 6 October 2008

Bob Glushko

Plan for INFO Lecture #11

Understanding the "Integration Problem"

"When Models Don't Match"

Enterprise Data Integration

Business Intelligence and "Dashboards"

Enterprise Information & Data Management Goals

- Run the business more efficiently through greater automation and "straight through" or "end-to-end" processing of the information that it creates and receives
- Consolidation of data from multiple business units to create a unified view of the {customer, supply chain, etc}
- Get end-to-end visibility of business processes
- Take different perspectives (from high level aggregation to resolving individual data anomalies or inconsistency)
- Make better decisions more quickly

Enterprise (and Inter-enterprise) Information & Data Management Challenges

- Internal to a firm, application "silos" or "stovepipes" may have been created over time and not have been designed to share information with each other
- Each of these systems has a specific purpose and a data model customized for that purpose - so these models may be incomplete or incompatible with respect to each other
- This causes problems when business processes can span multiple departments, business applications, or even multiple firms
- These problems are greater when information or data comes from outside the firm

The Integration Requirement

- Companies have so many internal (with employees) and external (with customers and suppliers) interactions that they must automate as many as possible
- This requires INTEGRATION -- the controlled and automated sharing of content, data and business processes among any services, applications, or information sources, intra- or inter-company
- Integration has long been a substantial portion of the IT activities in many companies

Integration Is NOT Interoperation

- Integration means that one application can extract or obtain information from another one
- It doesn't mean that the information will work "as is" for the receiving application or service
- Different systems may use different formats for nominally the same data items (September 11, 2001, 9/11/2001 and 9-11-01; 11/09/01 in Europe)
- Furthermore, there may be significant semantic differences between data items with the same name

Syntactic, Structural, & Semantic Interoperability

Syntactic interoperability is just the ability to exchange information. It requires agreement or compatibility at the transport and application layers of the communications protocol stack and with the messaging protocol and encoding format

Structural interoperability means that all of the expected information components are present with the same arrangement and granularity

Semantic interoperability requires that the content of the message be understood by the recipient application or process

Interoperability isn't All or None

Some interoperability problems can be detected and resolved by completely automated mechanisms

Other problems can be detected and resolved with some human intervention

Other problems can be detected but not resolved

Some problems can go undetected

Transformation in Semantic Integration

Semantic integration is the process by which a common semantic "data model" or "object model" is created through transformation

- Easy transformations include field length, data type or unit conversion
- Product attributes can be extracted from text descriptions
- Classification into product categories (e.g., UN SPSC) can be performed using some form of "nearest neighbor" or principle components analysis

What's the most powerful semantic integration processor?

Who does the transformation - the sender or the receiver?

Catalog Integration

We've talked about the need to combine information sources with different data/document models and semantics in many previous lectures

Combining "catalog information" is a challenge e-businesses (Stonebraker & Hellerstein) but also for digital libraries

The goal for both is to create catalogs that describe actual and "virtual" resources that they can provide access to without needing to collect or control them directly

Technical Challenges Shared by Multi-Enterprise Content / Digital Libraries

Information comes from owners with varying relationships with the integrator

Information comes in different formats and semantics

Information requires multiple schemas and multiple taxonomies

Technical Challenges that Contrast Multi-Enterprise Content / Digital Libraries

Digital libraries are integrating metadata, while e-business applications are integrating both metadata and content from hybrid document types

Standards for library content and metadata are better established than those for enterprise content

Information is syndicated / personalized much more for enterprise content

Enterprise information is operational and volatile

Non-Technical Challenges

Who makes decisions about destination formats?

Whose responsibility is it to create the destination format?

Who owns the new information?

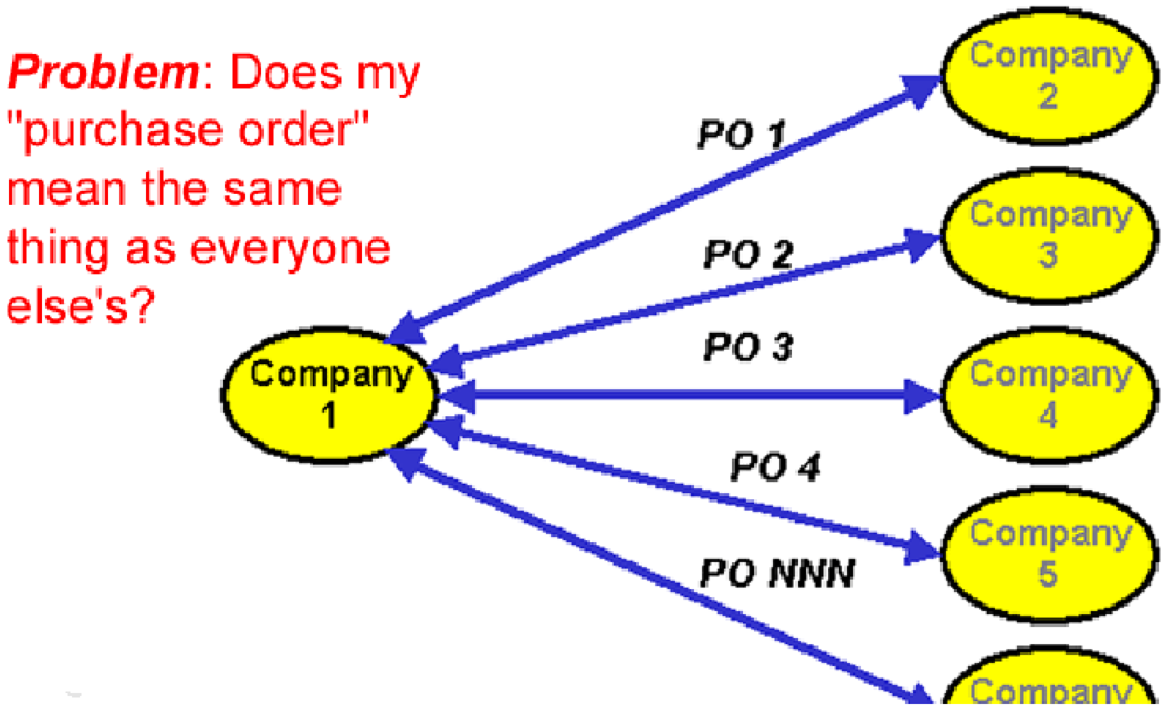
When Models Don't Match

Suppose you publish your web service interface description and tell the world "my ordering service requires a purchase order that conforms to this schema"

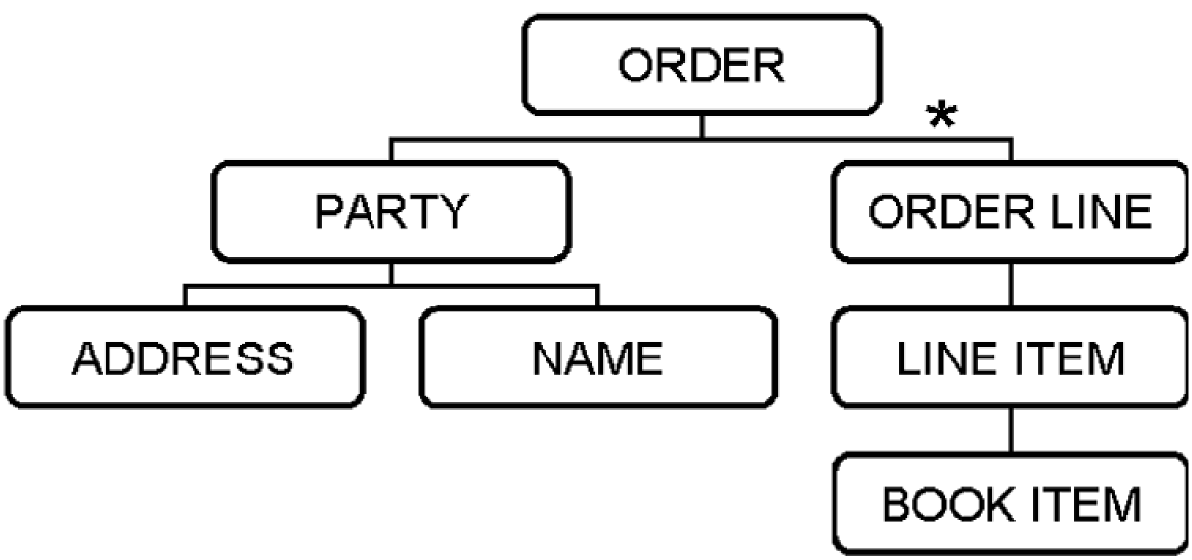
This says "send me MY purchase order" not "send me YOUR purchase order"

How likely is it that the purchase orders being used by other firms will be able to meet your interface requirement, either directly or after being transformed?

To Interoperate, or not to Interoperate?



The Target Model For The Interoperability Scenarios



The XSD Schema for the Expected Order [1]

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
 elementFormDefault="qualified">
<xs:element name="Order" type="OrderType"/>
<xs:complexType name="OrderType">
  <xs:sequence>
    <xs:element name="BuyersID" type="xs:string"/>
    <xs:element name="BuyerParty" type="PartyType"/>
    <xs:element name="OrderLine"  type="OrderLineType"
          maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="PartyType">
<xs:sequence>
    <xs:element name="ID" type="xs:string"/>
    <xs:element name="PartyName" type="PartyNameType"/>
    <xs:element name="Address" type="AddressType"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="PartyNameType">
<xs:sequence>
    <xs:element name="Name" type="xs:string" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
```

The XSD Schema for the Expected Order [2]

```
<xs:complexType name="AddressType">
<xs:sequence>
    <xs:element name="Room" type="xs:string"/>
    <xs:element name="BuildingNumber" type="xs:string"/>
    <xs:element name="StreetName" type="xs:string"/>
    <xs:element name="CityName" type="xs:string"/>
    <xs:element name="PostalZone" type="xs:string"/>
    <xs:element name="CountrySubentity" type="xs:string"/>
    <xs:element name="Country" type="xs:string"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="OrderLineType">
<xs:sequence>
    <xs:element name="LineItem" type="LineItemType"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="LineItemType">
<xs:sequence>
    <xs:element name="BookItem" type="BookItemType"/>
    <xs:element name="BasePrice" type="xs:decimal"/>
    <xs:element name="Quantity" type="xs:int"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="BookItemType">
<xs:sequence>
    <xs:element name="Title" type="xs:string"/>
    <xs:element name="Author" type="xs:string"/>
    <xs:element name="ISBN" type="xs:string"/>
</xs:sequence>
```

The Expected Instance

```
<Order>
<BuyersID>91604</BuyersID>
<BuyerParty>
<ID>KEEN</ID>
<PartyName>
   <Name>Maynard James Keenan</Name>
</PartyName>
<Address>
   <Room>505</Room>
   <BuildingNumber>11271</BuildingNumber>
   <StreetName>Ventura Blvd.</StreetName>
   <CityName>Studio City</CityName>
   <PostalZone>91604</PostalZone>
   <CountrySubentity>California</CountrySubentity>
   <Country>USA</Country>
</Address>
</BuyerParty>
<OrderLine>
<LineItem>
   <BookItem>
     <Title>Foucault's Pendulum</Title>
     <Author>Umberto Eco</Author>
     <ISBN>0345368754</ISBN>
     </BookItem>
     <BasePrice>7.99</BasePrice>
     <Quantity>1</Quantity>
</LineItem>
</OrderLine>
</Order>
```

Identical Model with Different Tag Names [1]

```
<Customer>
<Number>KEEN</Number>
<Name>
   <BusinessName>Maynard James Keenan</BusinessName>
</Name>

<Location>
   <Unit>505</Unit>
   <StreetNumber>11271</StreetNumber>
   <Street>Ventura Blvd.</Street>
   <City>Studio City</City>
   <ZipCode>91604</ZipCode>
   <State>California</State>
   <Country>USA</Country>
</Location>
</Customer>
```

Identical Model with Different Tag Names [2]

```
<Acheteur>
<ID>KEEN</ID>
<Nom>
   <NomCommercial>Maynard James Keenan</NomCommercial>
</Nom>
<Addresse>
   <Appartment>505</Appartment>
   <Bâtiment>11271</Bâtiment>
   <Rue>Ventura Blvd.</Rue>
   <Ville>Studio City</Ville>
   <CodePostal>91604</CodePostal>
   <Etat>California</Etat>
   <Pays>USA</Pays>
</Addresse>
</Acheteur>
```

Same Model, Attributes Instead of Elements

```
<BuyerParty
 ID="KEEN"
 Name="Maynard James Keenan"
 Room="505" BuildingNumber="11271"
 StreetName="Ventura Blvd."
 City="Studio City"
 State="California"
 PostalCode="91604"
 >
```

Granularity Conflicts

```
<Address>
  <StreetAddress>11271 Ventura Blvd. #505</StreetAddress>
  <City>Studio City 91604</City>
  <CountrySubentity>California</CountrySubentity>
  <Country>USA</Country>
</Address>

<PartyName>
  <FamilyName>Keenan</FamilyName>
  <MiddleName>James</MiddleName>
  <FirstName>Maynard</FirstName>
</PartyName>
```

Assembly Mismatch - Separate Customer and Order Documents [1]

```
<BuyerParty>
<ID>KEEN</ID>
<PartyName>
   <Name>Maynard James Keenan</Name>
</PartyName>
<Address>
   <Room>505</Room>
   <BuildingNumber>11271</BuildingNumber>
   <StreetName>Ventura Blvd.</StreetName>
   <CityName>Studio City</CityName>
   <PostalZone>91604</PostalZone>
   <CountrySubentity>California</CountrySubentity>
   <Country>USA</Country>
</Address>
</BuyerParty>
```

Assembly Mismatch - Separate Customer and Order Documents [2]

```
<Order>
<BuyersID>91604</BuyersID>
<BuyerParty>
   <ID>KEEN</ID>
</BuyerParty>
<OrderLine>
<LineItem>
   <BookItem>
     <Title>Foucault's Pendulum</Title>
     <Author>Umberto Eco</Author>
     <ISBN>0345368754</ISBN>
     </BookItem>
     <BasePrice>7.99</BasePrice>
     <Quantity>1</Quantity>
</LineItem>
</OrderLine>
</Order>
```

Conceptual Incompatibility

```
<Address>
   <Latitude direction="N">37.871</Latitude>
   <Longitude direction="W">-122.271</Longitude>
</Address>
```

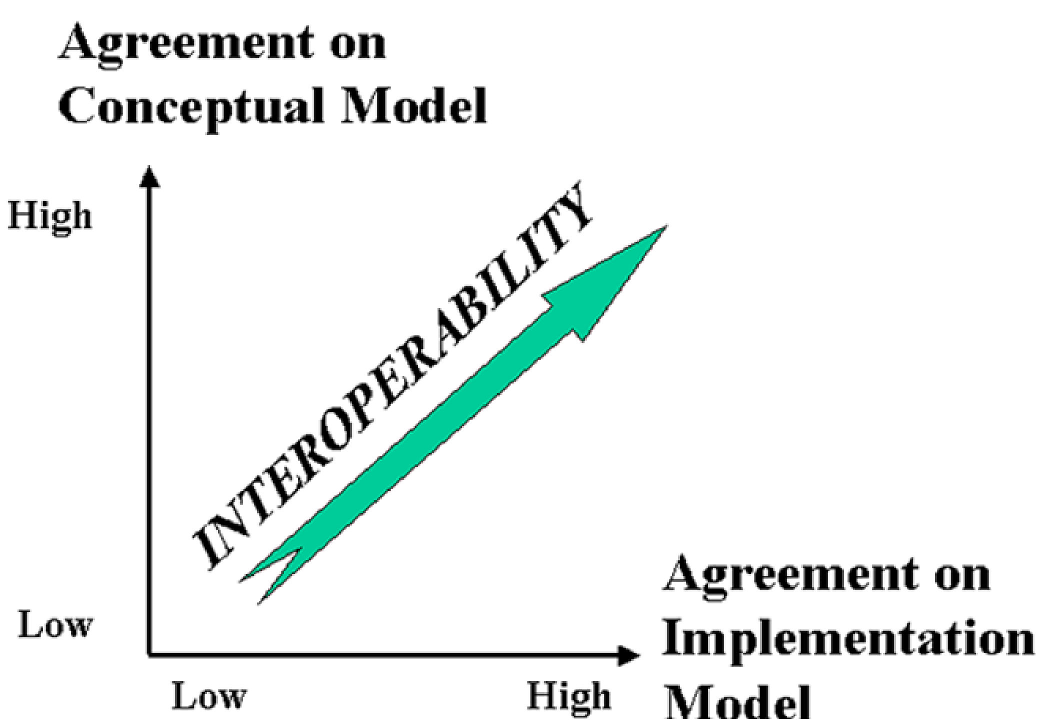
Lessons from the Interoperability Examples

There are a large number of ways that two implementation models that are supposed to be equivalent can fail that test

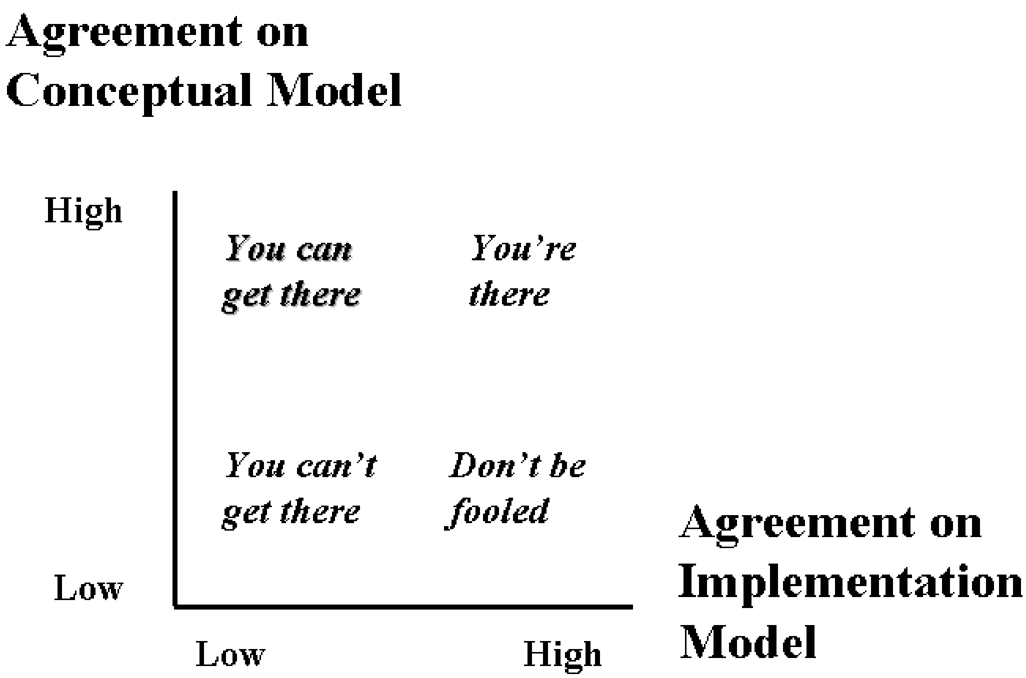
But no matter how different they look, with different syntaxes, tag names, or assembly models, if their conceptual model is the same, it is possible to transform one implementation model to another

Validation is not sufficient to guarantee complete interoperability

The Dimensions of Interoperability



The Dimensions of Interoperability



An Enterprise Information Integration Scenario

An existing customer calls a service representative to increase an order

The service representative must:

- locate information about the customer
- locate the existing order
- determine if the order can be changed or whether a new order must be created
- determine whether to accept the order based on the customer's payment history and credit

What information sources or applications must the service rep consult? How can it be done?

Integration "By Eye"

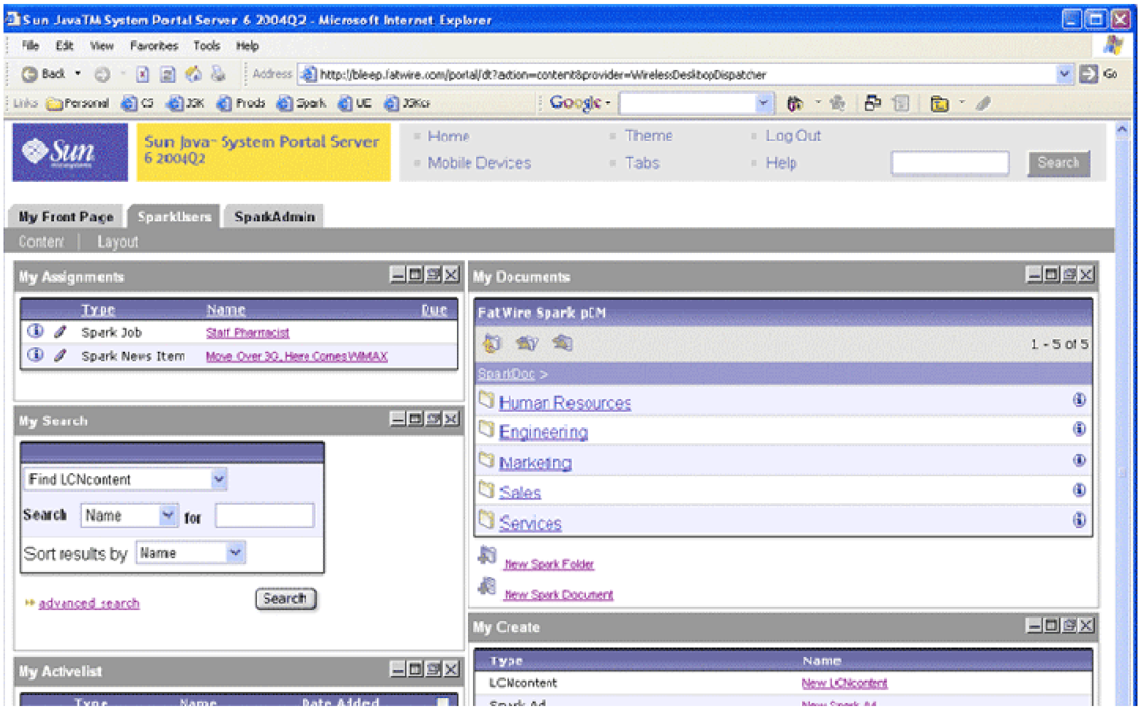
Swivel-Chair Integration

The need to consult multiple unintegrated applications to locate information to complete a business process

Recent study by Corizon:

- 66% of call center agents use three applications or more to serve customers on a typical call
- 27% use five or more
- 71% claim time is wasted on or after a call because of switching between different applications
- 53% admit that errors creep in when entering data into multiple systems

Enterprise Portals



Enterprise Portal Applications

Portal applications replace the different interfaces to multiple systems with a single, user-friendly screen that accesses only the parts of a back-end system that the employee needs

Purpose is to create a unified experience with a "single sign-on"

You can think of this trying to recreate something like Yahoo for the enterprise (Intranet)

Nearly every major software vendor has created an enterprise portal solution that is an attempt to "up-sell" from the application server platform

Enterprise Information Integration

Integration "by eye" is inadequate in situations with high transaction rates or complex data, and it is necessary for the applications to share data without human intervention

This requires true semantic unification of the underlying logic and content models, which may or may not be presented to the user as a single "composite application"

For large firms, and increasingly for medium-sized ones, a solution is to implement an ERP system that integrates all of the operational data

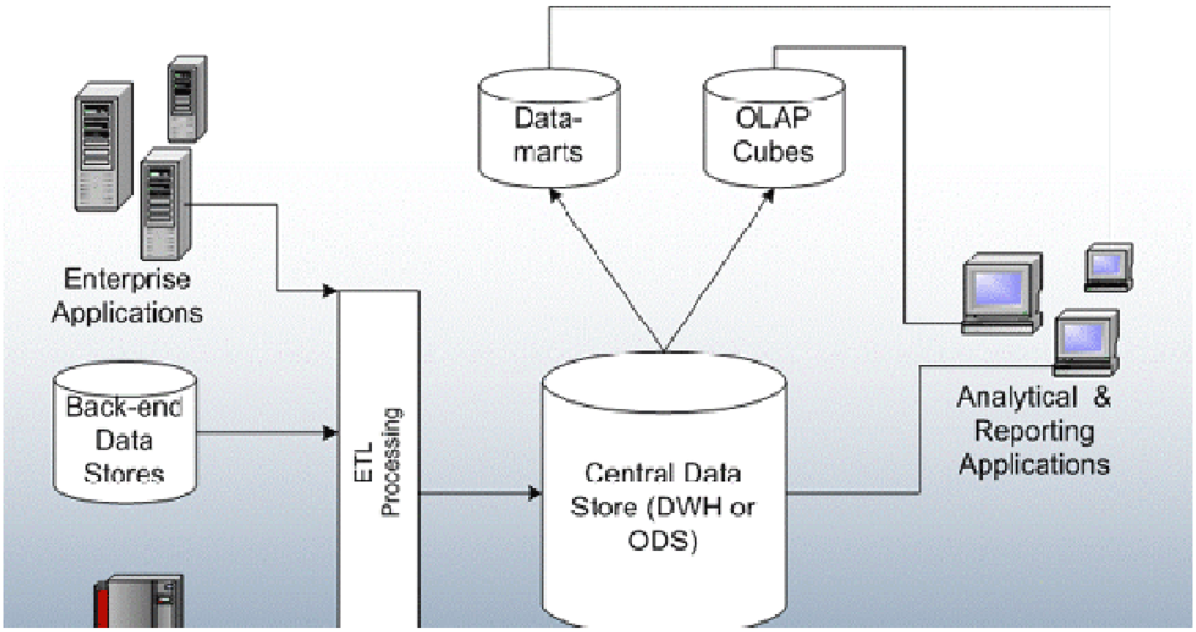
Business Intelligence

ERP and other enterprise systems contain the very granular and "live" operational data of the enterprise

ERP systems generate historical reports that are useful for long-term decision making, but don't enable ad hoc analysis of operations needed to make tactical decisions

So you need another set of your enterprise data organized in a data model optimized for asking questions rather than running your business

Generic Enterprise Information Integration Architecture (Gantz, 2004)



Data Warehouses

A data warehouse is a "subject-oriented, integrated, time-varying, non-volatile collection of data used in organizational decision making"

Data warehouses extract data from ERP systems and other related business software applications into a separate repository

It is common practice to "stage" data prior to merging it into a data warehouse with an "Extract, Transform, and Load" (ETL) application

Since the information won't change, denormalization to improve query performance is a common ETL process

The data model for the warehouse, designed to enable efficient ad hoc data analysis and reporting, is sometimes called a "hypercube"

A common term for the analysis done in a warehouse is online analytical processing or OLAP

The Virtual Warehouse

A virtual warehouse is created "on demand" by centralizing and normalizing metadata about the data sources rather than the data itself.

The data is left in its original location and extracted only when needed, which makes more "real time" analysis and "business intelligence"

Bad UI for Business Intelligence

Date	Ordered	Budget	%MoBud	Shipped	Prodn	%Scrap	Bk Orders	Finish Gds	%SNS	%Stk Opn	Ord
04/12 Tue	9117	40000	64.3	5639	27535	9.09	9289	626094	38.13	56.2	981
04/11 Mon	29364	40000	75.4	7762	17543	92.48	9500	761418	24.14	3.57	314
04/10 Sun			66.3		52931	52.18	103	4486	57.88	46.98	66
04/09 Sat			66.3		33443	38.16	4353	43611	38.72	90.43	54
04/08 Fri	17560	40000	66.3	11727	76227	7.9	2617	256052	16.78	61.48	963
04/07 Thu	121454	40000	72.3	5681	47345	39.77	4472	493547	38.09	83.78	162
04/06 Wed	350	40000	85.3	729	96389	21.16	2509	73723	48.35	85.44	479
	177845	200000		33558	351413	37.25					

---------------------------------------West 07:05am on 13 Apr---------------------------------------

Date	Ordered	Budget	%MoBud	Shipped	Prodn	%Scrap	Bk Orders	Finish Gds	%SNS	%Stk Opn	Ord
04/12 Tue	2031	40000	64.3	7613	65227	4.67	7939	397196	60.7	11.21	553
04/11 Mon	199078	40000	75.4	24353	17491	22.88	80	491770	9.75	68.48	290
04/10 Sun			66.3		68668	26.63	6777	296368	28.01	81.6	82
04/09 Sat			66.3		77862	98.02	1220	614643	92.86	80.06	62
04/08 Fri	30614	40000	66.3	65615	44575	16.01	4085	876569	28.58	32.7	689
04/07 Thu	5707	40000	72.3	3931	66256	26.75	3386	835051	71.4	42.76	485
04/06 Wed	1274	40000	85.3	9203	29315	46.78	2636	44358	21.27	42.93	541
	238704	200000		118715	369384	34.53					

---------------------------------------South 09:01am on 13 Apr---------------------------------------

Date	Ordered	Budget	%MoBud	Shipped	Prodn	%Scrap	Bk Orders	Finish Gds	%SNS	%Stk Opn	Ord
04/12 Tue	9086	40000	64.3	2548	74901	51.04	5151	126651	19.31	87.64	265
04/11 Mon	88882	40000	75.4	77567	29842	46.3	9469	415786	22.49	49.13	390
04/10 Sun			66.3		27473	30.65	2628	839236	71.95	70.9	567
04/09 Sat			66.3		99504	83.87	3583	482674	22.55	11.16	268
04/08 Fri	50899	40000	66.3	53323	44674	16.44	381	392549	32.68	88.93	115
04/07 Thu	3124	40000	72.3	7839	34419	63.59	9163	994794	77.96	35.38	630
04/06 Wed	9469	40000	85.3	8872	63609	92.83	3467	958648	2.57	74.38	993
	161460	200000		155369	374422	54.96					

Driving Your Business

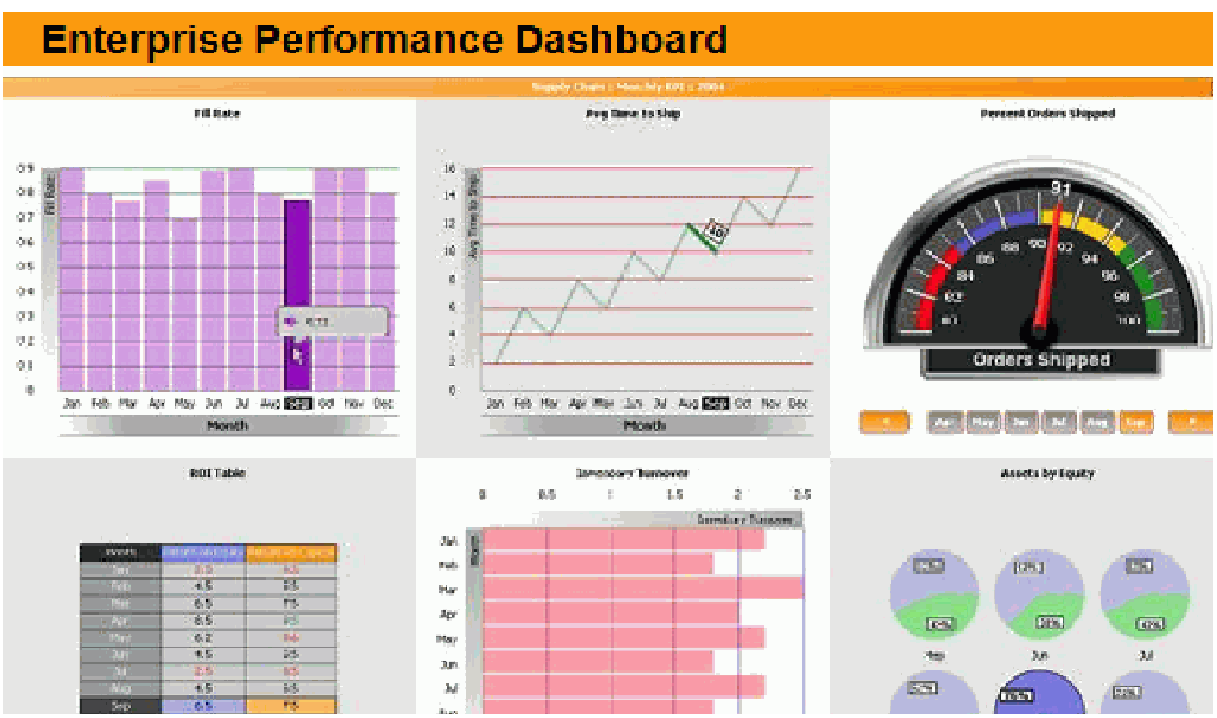
The best data warehouse design and the most clever OLAP won't help the business if the analysis can't be understood by the decision makers

"Dashboards" combine information integration with information visualization to enhance the usability of business intelligence

A dashboard provides hierarchical views appropriate to different management levels and the means to "drill down" to find details

See idashboards.com or demo.visualmining.com

Dashboard UI for Business Intelligence



Readings for INFO Lecture #12

M. Brun, J. Brown and R. Lohde, "Adoption of UBL in Denmark: Business cases and experiences"

Smita Brunnermeier and Sheila Martin, "Interoperability Costs in the US Automotive Supply Chain"

Arnon Rosenthal, Len Seligman, and Scott Renner. "From Semantic Integration to Semantics Management: Case Studies and a Way Forward"