Document Engineering: Designing Documents for Transactions and Web Services

Bob Glushko (glushko@sims.berkeley.edu)

OASIS Symposium - 24 April 2005

1. Who Are We, And Why Are We Here?

- Who Am I?
- Who Are You?

2. Outline for the Tutorial

- Motivating Document Engineering
- XML By Itself Does Nothing
- The Document Engineering Approach
  - Modeling Business Processes
  - Modeling Document Exchanges
  - Distinguishing Content, Structure, and Presentation
  - Harvesting and Consolidating Components
  - Assembling Document Models from Content Components
  - Encoding Models in XML
- Example 1: Event Calendar Network
- Example 2: Composite Travel Service
- We WON'T talk much about XML and implementation because (a) that's what you already know and (b) the analysis and modeling activities are bigger determinants of project success and failure

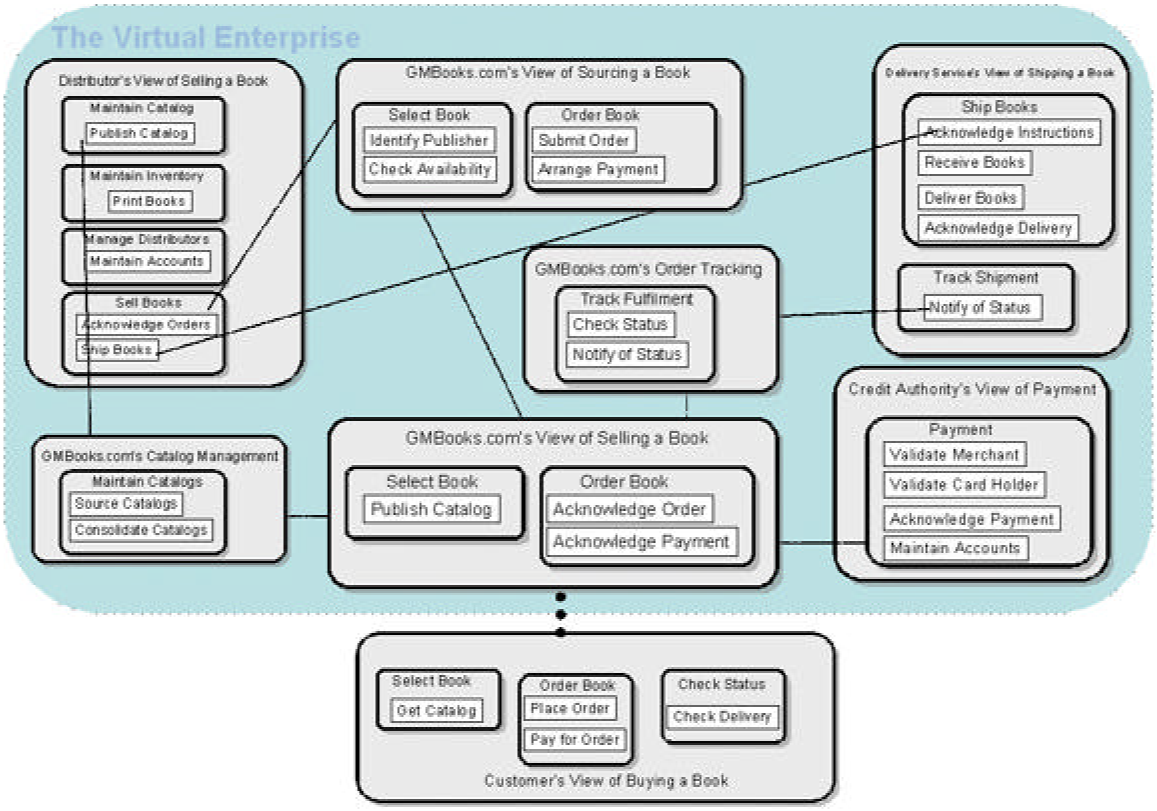
3. What is Document Engineering?

- A new discipline for specifying, designing, and implementing the electronic documents that request or provide interfaces to business processes, often via Web-based services
- A synthesis of information and systems analysis, business process modeling, electronic publishing, and distributed computing
- A set of courses taught at UC Berkeley
- An upcoming book (co-authored with Tim McGrath, MIT Press, 2005)

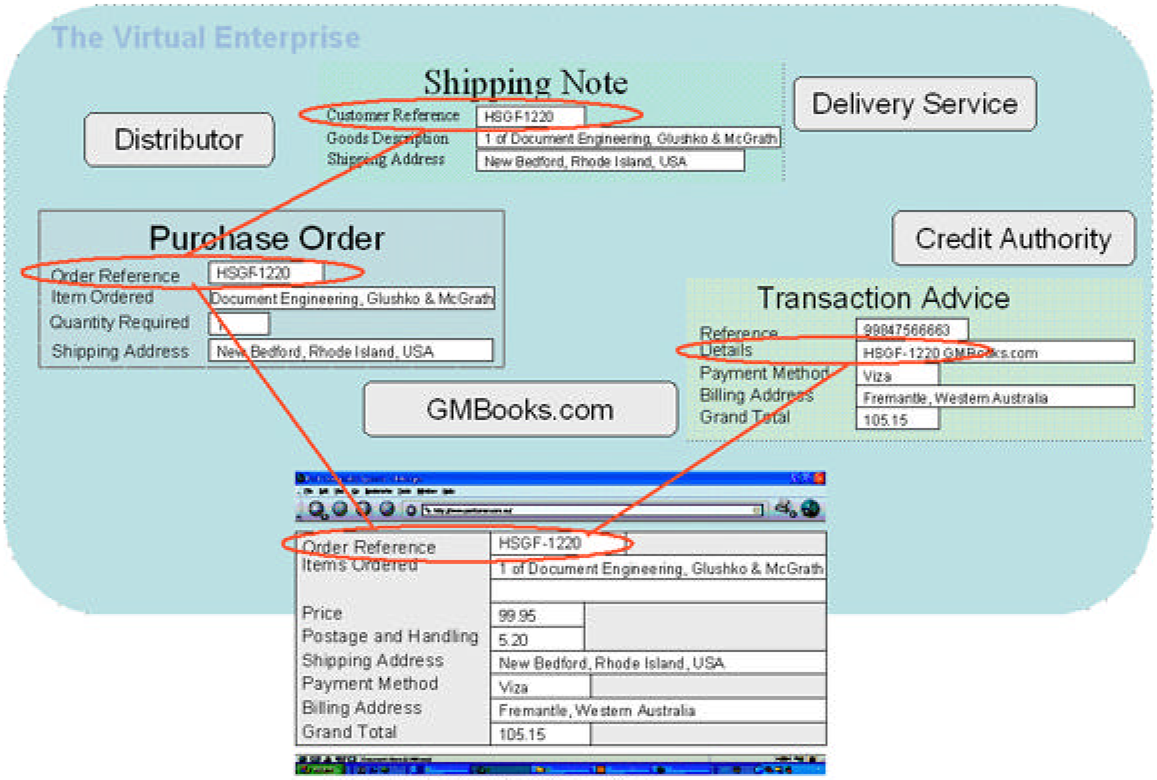
4. Motivating "Document Engineering"

- Scenario:
  - Customer selects book from catalog on an online bookstore
  - Customer pays with credit card
  - Book arrives via express shipper two days later
- From the customer's perspective there is only one "transaction"
- But the bookstore is a virtual enterprise that follows the drop shipment pattern to coordinate the activities of 4 different service providers transacting with each other
- This coordination - or choreography - is carried out with document exchanges

5. The Virtual Bookstore



6. Overlapping Information Models in the Virtual Bookstore



Shipping Note

Customer Reference	HSGF1220
Goods Description	1 of Document Engineering, Glushko & McGrath
Shipping Address	New Bedford, Rhode Island, USA

Purchase Order

Order Reference	HSGF1220
Item Ordered	Document Engineering, Glushko & McGrath
Quantity Required	1
Shipping Address	New Bedford, Rhode Island, USA

Transaction Advice

Reference	99847566663
Details	HSGF-1220 GMBooks.com
Payment Method	Viza
Billing Address	Fremantle, Western Australia
Grand Total	105.15

Order Reference	HSGF-1220
Items Ordered	1 of Document Engineering, Glushko & McGrath
Price	99.95
Postage and Handling	5.20
Shipping Address	New Bedford, Rhode Island, USA
Payment Method	Viza
Billing Address	Fremantle, Western Australia
Grand Total	105.15

Customer's View of Buying a Book

7. Questions for Document Engineering

- What documents are being exchanged?
- What does each document mean?
- Will the recipient process the document the way its sender expects?
- Can each service provider preserve its investments in older technologies for document exchange while taking advantage of new ones?
- Can each service provider preserve its investments in business processes and relationships while creating new ones?

8. The Document Exchange Pattern

- Businesses have long dealt with each other by exchanging documents
- Halfat's clay pot receipt for taxes is certainly one of the oldest documents that record a business transaction (355 BCE)

On the 16th of Tammuz, year 4 of Artaxerces, Halfat brought barley: 1 kor, 12 seah, 3 qab; wheat: 1 kor, 5 seah, 4 qab.

9. The Document Exchange Pattern (continued)

- Very natural thing to do
  - the simplest case is "here's my catalog, do you want to buy anything" and the exchanged document being "here's my order"
- We use concepts like "supply chains" and "distribution channels" as metaphors for the coordinated or choreographed flow of information and materials/products between businesses
- These are complex patterns composed from the document exchange pattern

10. The Evolution of "Business Architectures" for Document Exchange

- The technology for business documents has changed throughout history
- But the basic idea of document exchange has changed relatively little
- For over two thousand years the "business architecture" around the exchange of documents was non-proprietary and loosely-coupled
- Neither party to the exchange needed to know how the other produced or understood the documents – the sender made no assumptions about the technology at the other end
- For a very short time period (evolutionarily speaking) from about 1950-1995 we suffered through a period of proprietary and tightly-coupled business architecture
- Fortunately the tightly coupled approach is turning out to be an evolutionary dead end for document exchange

- We're back to a non-proprietary and loosely-coupled architecture for the exchange of business documents – now using the Internet, XML, Web services

11. Web Services

- *Web services* is today's biggest buzzword
- The idea is simple – encapsulate or "wrap" some specific and discrete unit of functionality to hide its implementation and make it reusable by sending it an XML message, to which it replies with an XML message
- *"What the Web did for program-to-user interactions, Web Services are poised to do for program-to-program interactions"* – typical hype about Web services
- Business models like Drop Shipment are a natural fit for web services

12. Document Engineering as the Methodology for Exploiting the Internet as a Business Application Platform

- But being non-proprietary and loosely-coupled isn't sufficient for a successful document exchange – exchanging information does no good if the information can’t be understood by the parties (or applications) doing the exchanging.
- The Web services "standards" not only don't solve this problem – they completely ignore it
- Document Engineering will ensure that the documents can be understood

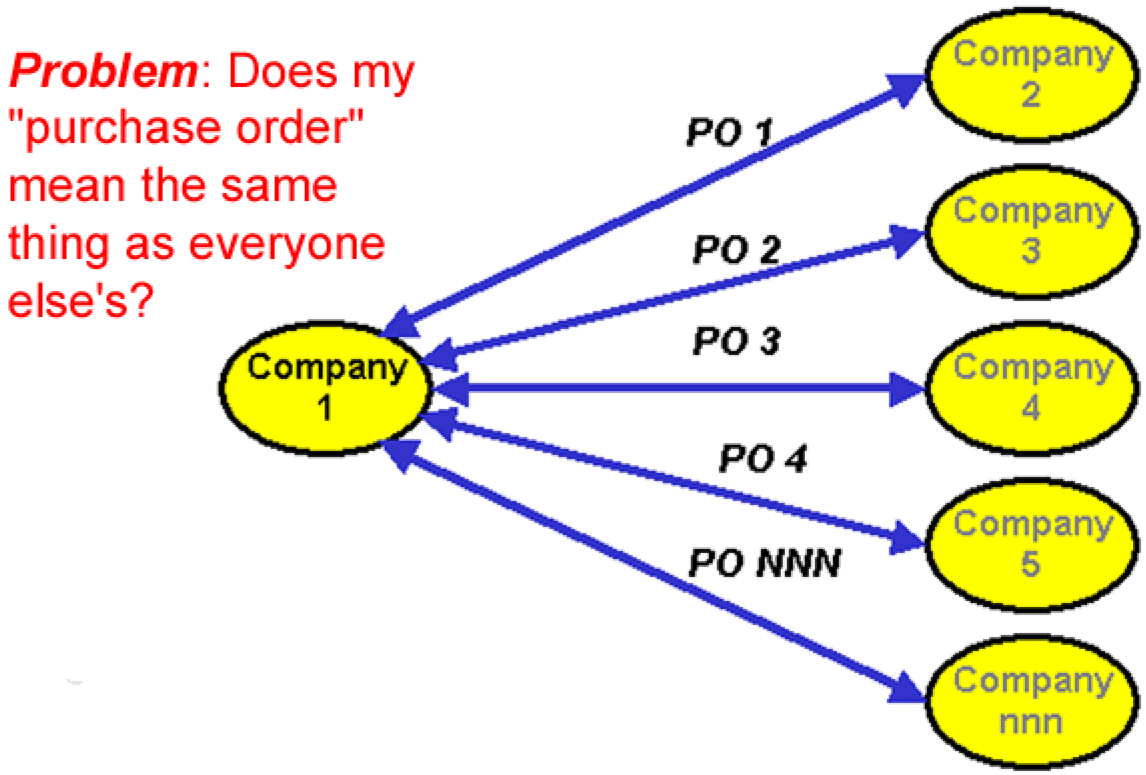
13. Document Engineering Isn't Just About XML

- XML is a useful technology for Document Engineering, but using XML doesn't make you a document engineer
- The *best* thing about XML is the ease with which you can create a new vocabulary for a particular type of document
- XML is just the syntax in which we encode document models... what really matters is how we modeled the documents

14. Creating Models is Easy, But Creating GOOD Models is Hard

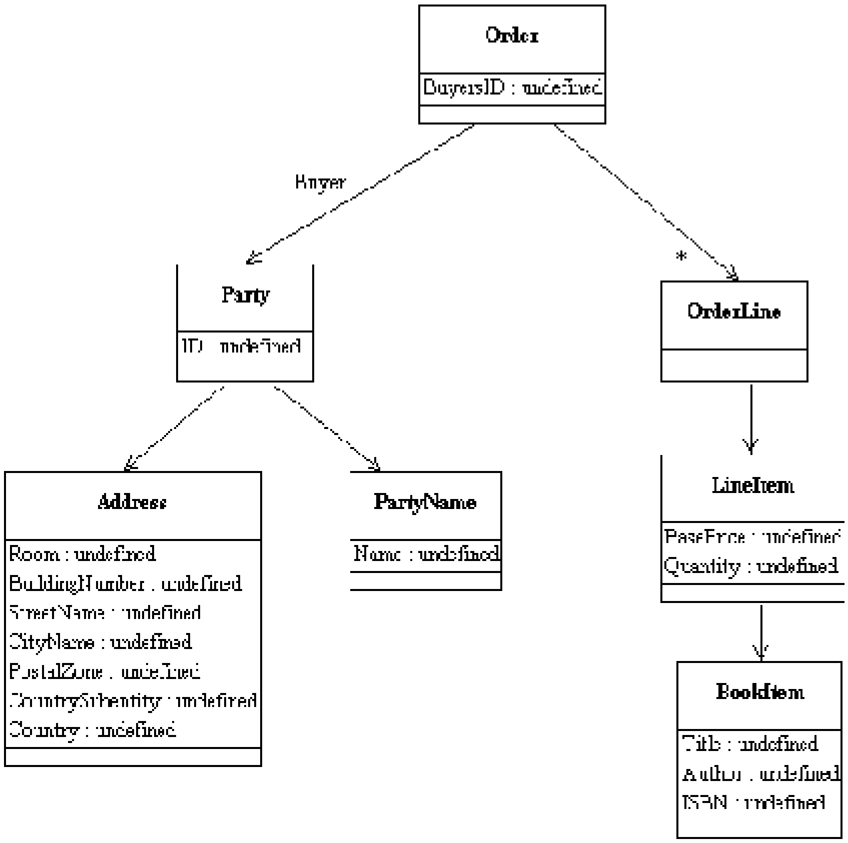
- The *worst* thing about XML is the same as the best thing – the ease with which you can create a new vocabulary
- No way around the classical problems of classification and naming we know from philosophy, linguistics, cognitive psychology, and information science
- XML is NOT "self-describing"
    - The same content will inevitably be described using different names, and different content will be given the same names
- There are often multiple vocabularies for the same or related domains and especially for the common information models that are used in more than one domain

15. The Equivalence Problem



Problem: Does my "purchase order" mean the same thing as everyone else's?

16. The Target Model For The Interoperability Scenarios



17. The XSD Schema for the Expected Order

```
<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>

<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>
</xs:complexType>
</xs:schema>
```

18. 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>
```

19. 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>
```

20. 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>
```

21. 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"
 >
```

22. 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>
```

23. Assembly Mismatch - Separate Customer and Order Documents

```
<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>

<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>
```

24. Conceptual Incompatibility

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

25. The "Not So Fast" Cases that Might Even Validate

- The names are the same but the semantics aren't

```
<BuyerParty>
<ID>555-22-1234</ID>

<Address>
   <Room>505</Room>
   <BuildingNumber>11271</BuildingNumber>
   <StreetName>Ventura Blvd.</StreetName>
   <CityName>Studio City</CityName>
   <PostalZone>91604-3136</PostalZone>
   <CountrySubentity>California</CountrySubentity>
   <Country>USA</Country>
</Address>
```

26. Validation Does Not Imply Interoperability

- After all these cases where interoperability may or may not possible because the conceptual or implementation models differ we need to talk about the "easy" case ... and make sure you recognize that it might not be

- Suppose the document validates against the recipient's schema

    - The semantics can still be different in important ways (the ID SSN example) – the strongest level of validation can fall short of establishing that the "same tags" have exactly the "same meaning" to the sender and recipient

    - Furthermore, the recipient may not be able to validate all of the business rules that are important

    - This is a good argument for industry standards / reference models / in your conceptual models or using XML vocabularies that represent them in authoritative ways

27. The Big Ideas of Document Engineering

- Doing business with documents implies an agreement about what the documents mean and about how the recipient will process and reply them – we need formal (and computer readable) models for both the documents and the processes in which they participate

- Doing business requires both "publication-like" document types like brochures and technical manuals and "transactional" documents like purchase orders and invoices – so we need analysis and design methods that work for both ends of this "Document Type Spectrum"

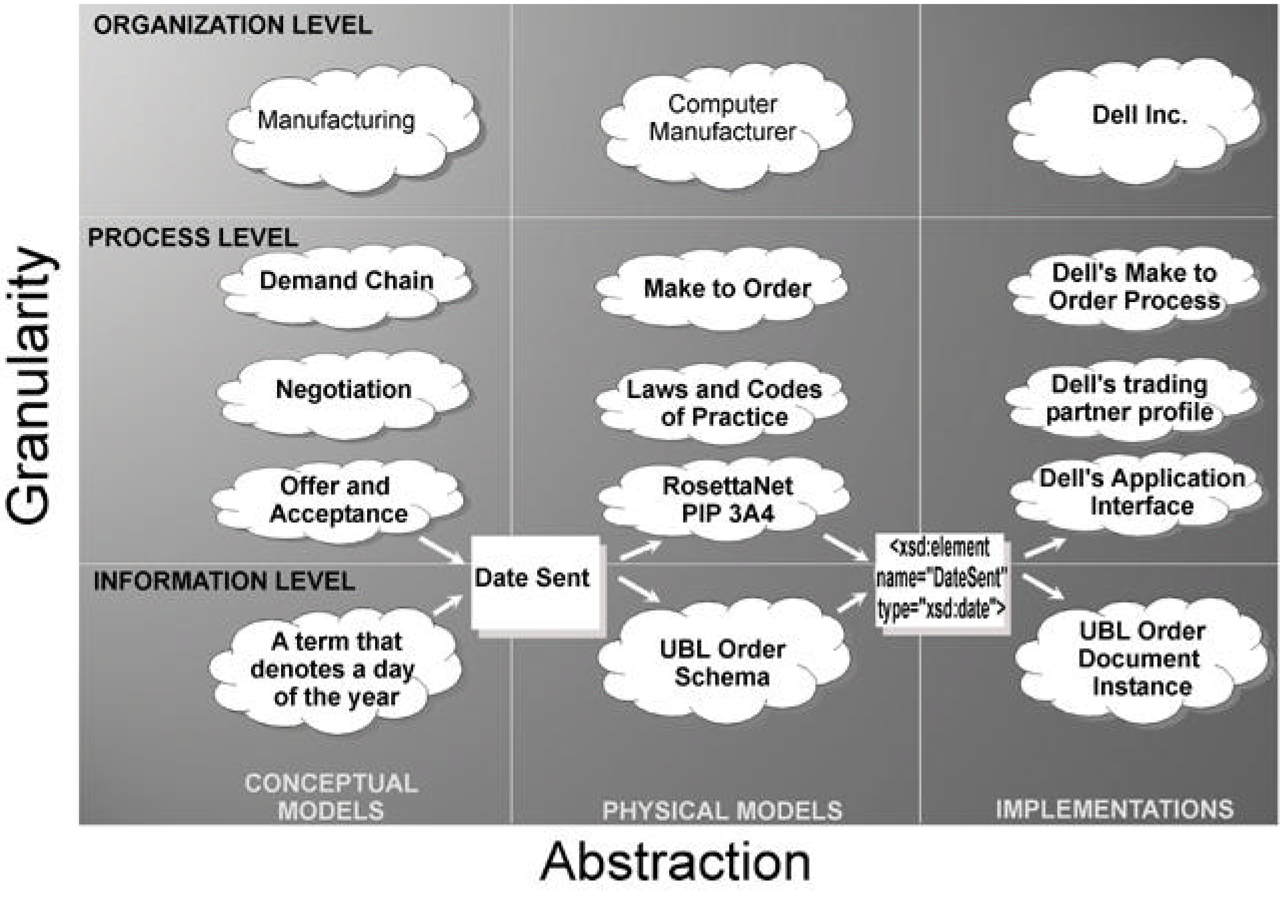
- Businesses try to differentiate themselves but tend to do things in similar ways – and the fundamental requirement for mutual intelligibility leads to regular patterns in documents and business processes

- Using XML schemas to encode the models of both the documents and their associated processes is attractive for both theoretical and practical reasons

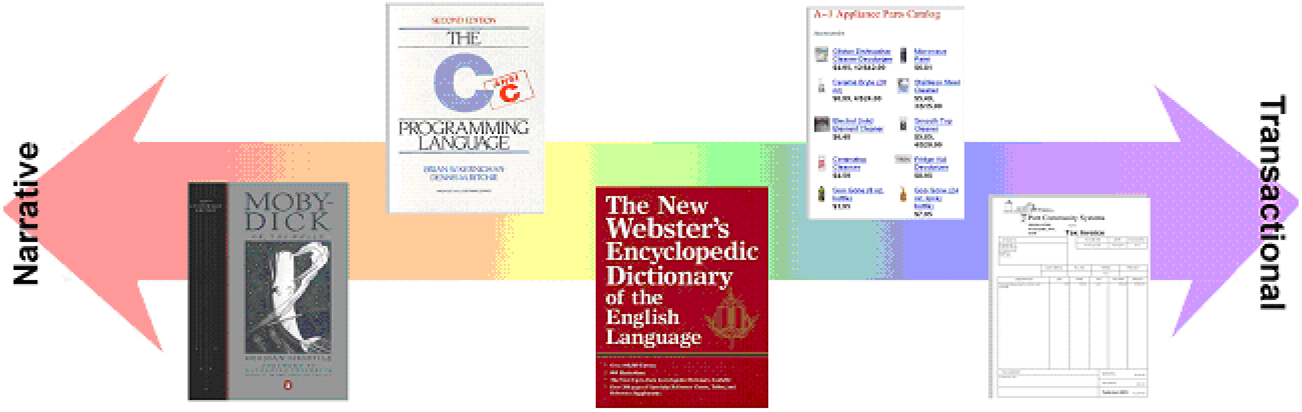
28. Document Exchange is the Mother of All Patterns

- Document exchange is the "mother of all patterns" for business models, business processes, and business information
    - *Business model or organizational* patterns: marketplace, auction, supply chain, build to order, drop shipment, vendor managed inventory, etc.
    - *Business process* patterns: procurement, payment, shipment, reconciliation, etc.
    - *Business information* patterns: catalog, purchase order, invoice, etc. and the components they contain for party, time, location, measurement, etc.

29. The Model Matrix



30. The Document Type Spectrum


Narrative
MOBY-DICK
THE C PROGRAMMING LANGUAGE
The New Webster's Encyclopedic Dictionary of the English Language
A-1 Appliance Parts Catalog
Tax Invoice
Transactional

- "Publications" or "Narrative" Document Types
    - Examples: Brochures, user guides, technical manuals, ...
- "Transactional" Document Types
    - Examples: Purchase orders, invoices, payment instructions, ...
- "Hybrids"
    - In between these endpoints are documents that exhibit more regularity in data content and structure than pure narrative types but for which presentation remains important because they are targeted for use by people
    - Examples: Catalogs, encyclopedias, ...

31. The Document Type Spectrum – "Publications"

- Authored by people
- Highly designed, with rich presentational characteristics correlated with semantics and structure
- Heterogeneous in structure and content
- Weakly datatyped – "just text"

32. The Document Type Spectrum – "Transactional Documents"

- Created mechanically
- Few and somewhat arbitrary presentational characteristics
- Homogeneous in structure and content
- Strongly datatyped

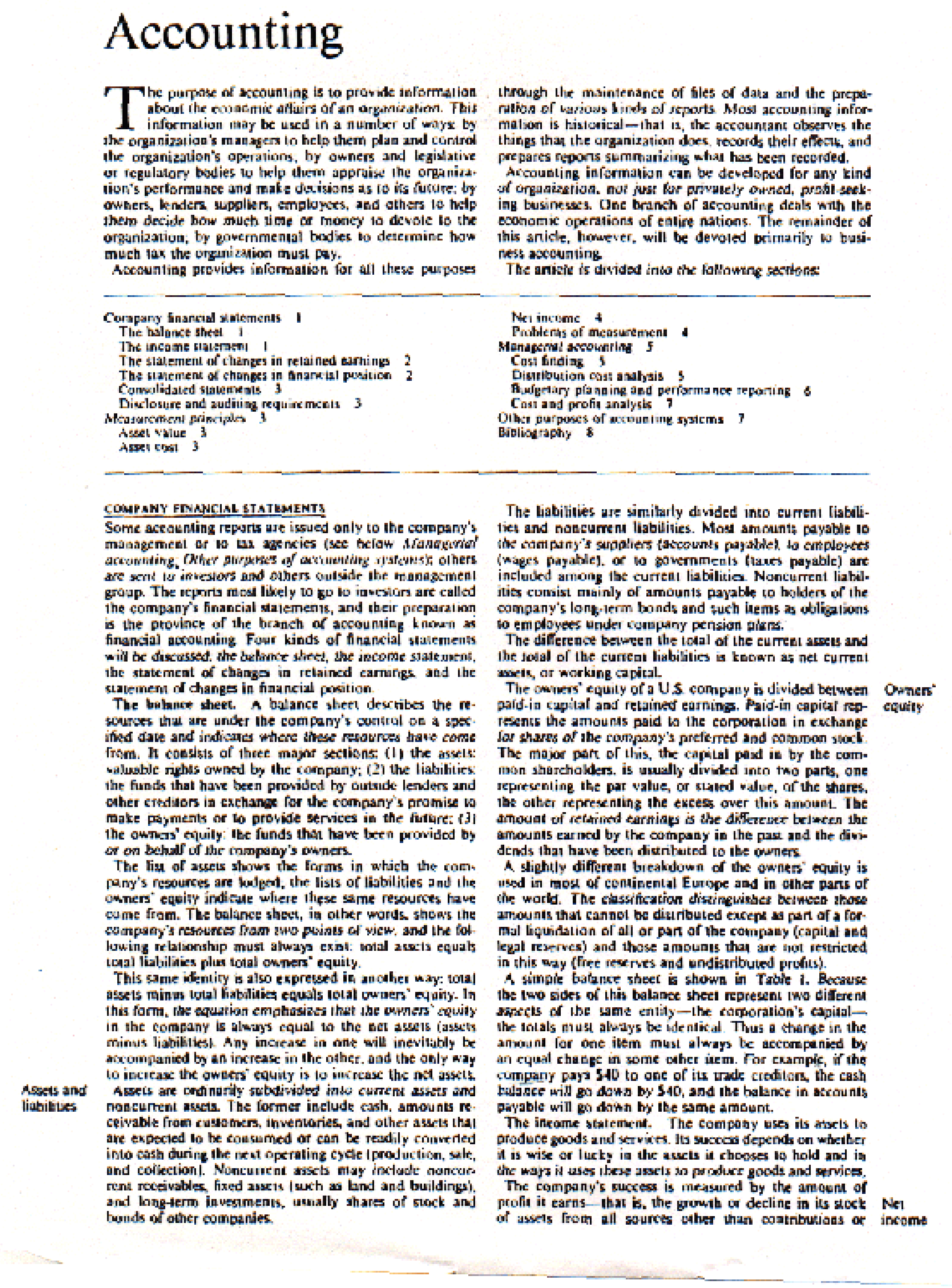
33. Systematic Variation in Document Types Across the Spectrum

- Instances more heterogeneous on narrative end
- Types are "broader" and more descriptive, less prescriptive on narrative end
- The set of content types within a document type is much greater on the transactional end because the leaves aren't "just text"
- More need for "metadata" augmentation of documents on narrative end, because on transactional end what would be metadata is more likely to be explicitly contained in the content already
- Presentational information more likely to be correlated with content and structure on narrative end

34. Dictionaries, Encyclopedias, and Reference Books

- Usually very carefully designed, with regular structure that is exploited in information access and navigation features to enhance usability
- Often have rich repertoire of content component types (pictures, maps, charts, formulas, tables)
- *Mixed content* in paragraphs or other text blocks will contain numerous content types

35. Encyclopedia Entry



Accounting

The purpose of accounting is to provide information about the economic affairs of an organization. This information may be used in a number of ways: by the organization's managers to help them plan and control the organization's operations, by owners and legislative or regulatory bodies to help them appraise the organization's performance and make decisions as to its future; by owners, lenders, suppliers, employees, and others to help them decide how much time or money to devote to the organization; by governmental bodies to determine how much tax the organization must pay.

Accounting provides information for all these purposes through the maintenance of files of data and the preparation of various kinds of reports. Most accounting information is historical—that is, the accountant observes the things that the organization does, records their effects, and prepares reports summarizing what has been recorded.

Accounting information can be developed for any kind of organization, not just for privately owned, profit-seeking businesses. One branch of accounting deals with the economic operations of entire nations. The remainder of this article, however, will be devoted primarily to business accounting.

The article is divided into the following sections:

Company financial statements 1
The balance sheet 1
The income statement 1
The statement of changes in retained earnings 2
The statement of changes in financial position 2
Consolidated statements 3
Disclosure and auditing requirements 3
Measurement principles 3
Asset value 3
Asset cost 3
Net income 4
Problems of measurement 4
Managerial accounting 5
Cost finding 5
Distribution cost analysis 5
Budgetary planning and performance reporting 6
Cost and profit analysis 7
Other purposes of accounting systems 7
Bibliography 8

COMPANY FINANCIAL STATEMENTS

Some accounting reports are issued only to the company's management or to tax agencies (see below *Managerial accounting*, *Other purposes of accounting systems*); others are sent to investors and others outside the management group. The reports most likely to go to investors are called the company's financial statements, and their preparation is the province of the branch of accounting known as financial accounting. Four kinds of financial statements will be discussed: the balance sheet, the income statement, the statement of changes in retained earnings, and the statement of changes in financial position.

The balance sheet. A balance sheet describes the resources that are under the company's control on a specified date and indicates where these resources have come from. It consists of three major sections: (1) the assets: valuable rights owned by the company; (2) the liabilities: the funds that have been provided by outside lenders and other creditors in exchange for the company's promise to make payments or to provide services in the future; (3) the owners' equity: the funds that have been provided by or on behalf of the company's owners.

The list of assets shows the forms in which the company's resources are lodged; the lists of liabilities and the owners' equity indicate where these same resources have come from. The balance sheet, in other words, shows the company's resources from two points of view, and the following relationship must always exist: total assets equals total liabilities plus total owners' equity.

This same identity is also expressed in another way: total assets minus total liabilities equals total owners' equity. In this form, the equation emphasizes that the owners' equity in the company is always equal to the net assets (assets minus liabilities). Any increase in one will inevitably be accompanied by an increase in the other, and the only way to increase the owners' equity is to increase the net assets.

Assets and liabilities

Assets are ordinarily subdivided into current assets and noncurrent assets. The former include cash, amounts receivable from customers, inventories, and other assets that are expected to be consumed or can be readily converted into cash during the next operating cycle (production, sale, and collection). Noncurrent assets may include noncurrent receivables, fixed assets (such as land and buildings), and long-term investments, usually shares of stock and bonds of other companies.

The liabilities are similarly divided into current liabilities and noncurrent liabilities. Most amounts payable to the company's suppliers (accounts payable), to employees (wages payable), or to governments (taxes payable) are included among the current liabilities. Noncurrent liabilities consist mainly of amounts payable to holders of the company's long-term bonds and such items as obligations to employees under company pension plans.

The difference between the total of the current assets and the total of the current liabilities is known as net current assets, or working capital.

Owners' equity

The owners' equity of a U.S. company is divided between paid-in capital and retained earnings. Paid-in capital represents the amounts paid to the corporation in exchange for shares of the company's preferred and common stock. The major part of this, the capital paid in by the common shareholders, is usually divided into two parts, one representing the par value, or stated value, of the shares, the other representing the excess over this amount. The amount of retained earnings is the difference between the amounts earned by the company in the past and the dividends that have been distributed to the owners.

A slightly different breakdown of the owners' equity is used in most of continental Europe and in other parts of the world. The classification distinguishes between those amounts that cannot be distributed except as part of a formal liquidation of all or part of the company (capital and legal reserves) and those amounts that are not restricted in this way (free reserves and undistributed profits).

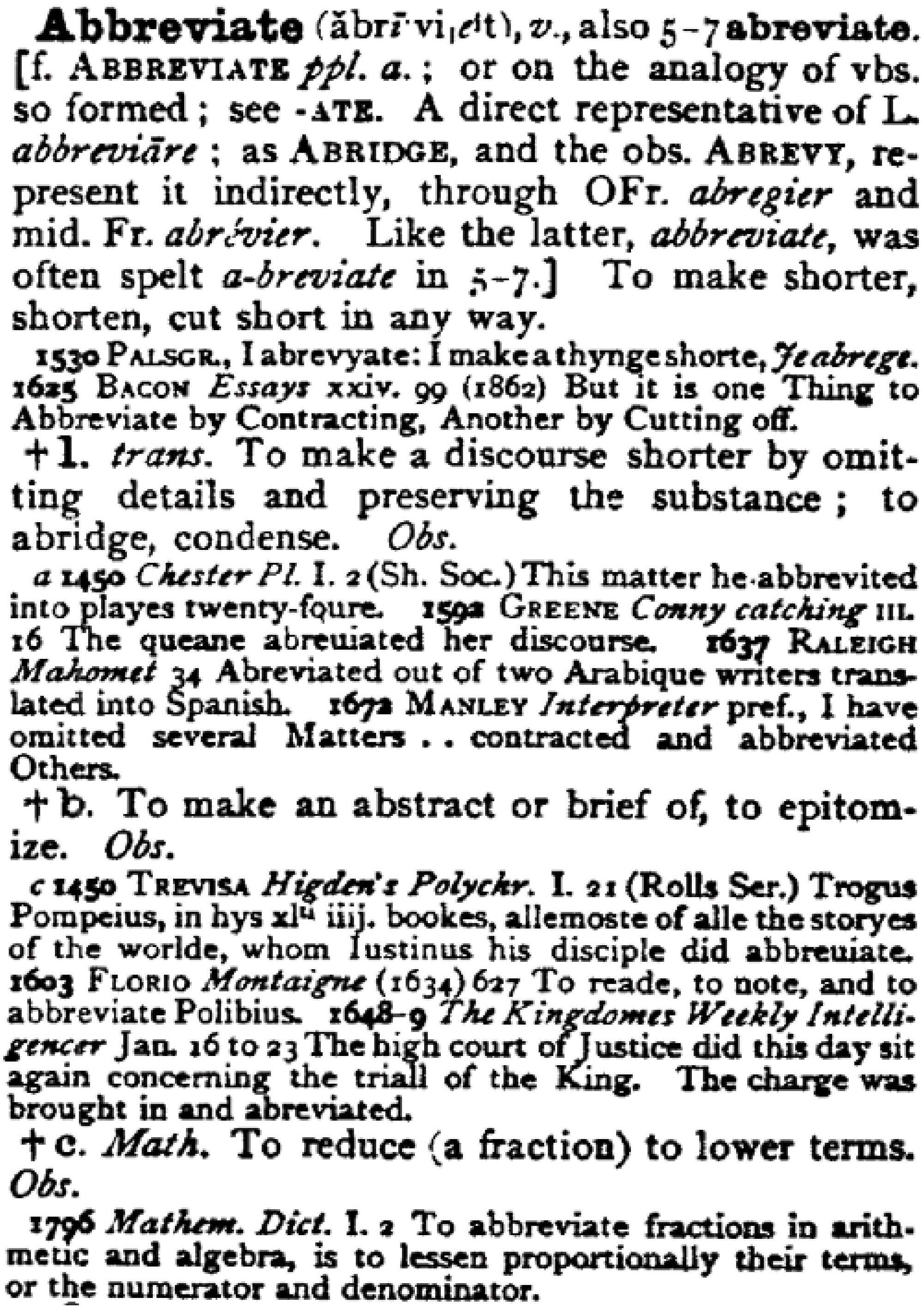
A simple balance sheet is shown in Table 1. Because the two sides of this balance sheet represent two different aspects of the same entity—the corporation's capital—the totals must always be identical. Thus a change in the amount for one item must always be accompanied by an equal change in some other item. For example, if the company pays $40 to one of its trade creditors, the cash balance will go down by $40, and the balance in accounts payable will go down by the same amount.

The income statement. The company uses its assets to produce goods and services. Its success depends on whether it is wise or lucky in the assets it chooses to hold and in the ways it uses these assets to produce goods and services.

Net income

The company's success is measured by the amount of profit it earns—that is, the growth or decline in its stock of assets from all sources other than contributions or

36. Oxford English Dictionary – Typical Entry



Abbreviate (ăbrī·viˌeɪt), *v.*, also 5–7 **abreviate**. [f. ABBREVIATE *ppl. a.*; or on the analogy of vbs. so formed; see -ATE. A direct representative of L. *abbreviāre*; as ABRIDGE, and the obs. ABREVY, represent it indirectly, through OFr. *abregier* and mid. Fr. *abrévier*. Like the latter, *abbreviate*, was often spelt *a-breviate* in 5–7.] To make shorter, shorten, cut short in any way.

1530 PALSGR., I abrevyate: I make a thynge shorte, *Je abrege*. 1625 BACON *Essays* xxiv. 99 (1862) But it is one Thing to Abbreviate by Contracting, Another by Cutting off.

†1. *trans.* To make a discourse shorter by omitting details and preserving the substance; to abridge, condense. *Obs.*

a 1450 *Chester Pl.* I. 2 (Sh. Soc.) This matter he·abbrevited into playes twenty-foure. 1592 GREENE *Conny catching* III. 16 The queane abreuiated her discourse. 1637 RALEIGH *Mahomet* 34 Abreviated out of two Arabique writers translated into Spanish. 1672 MANLEY *Interpreter* pref., I have omitted several Matters . . contracted and abbreviated Others.

†**b.** To make an abstract or brief of, to epitomize. *Obs.*

c 1450 TREVISA *Higden's Polychr.* I. 21 (Rolls Ser.) Trogus Pompeius, in hys xl^{u} iiij. bookes, allemoste of alle the storyes of the worlde, whom Iustinus his disciple did abbreuiate. 1603 FLORIO *Montaigne* (1634) 627 To reade, to note, and to abbreviate Polibius. 1648–9 *The Kingdomes Weekly Intelligencer* Jan. 16 to 23 The high court of Justice did this day sit again concerning the triall of the King. The charge was brought in and abreviated.

†**c.** *Math.* To reduce (a fraction) to lower terms. *Obs.*

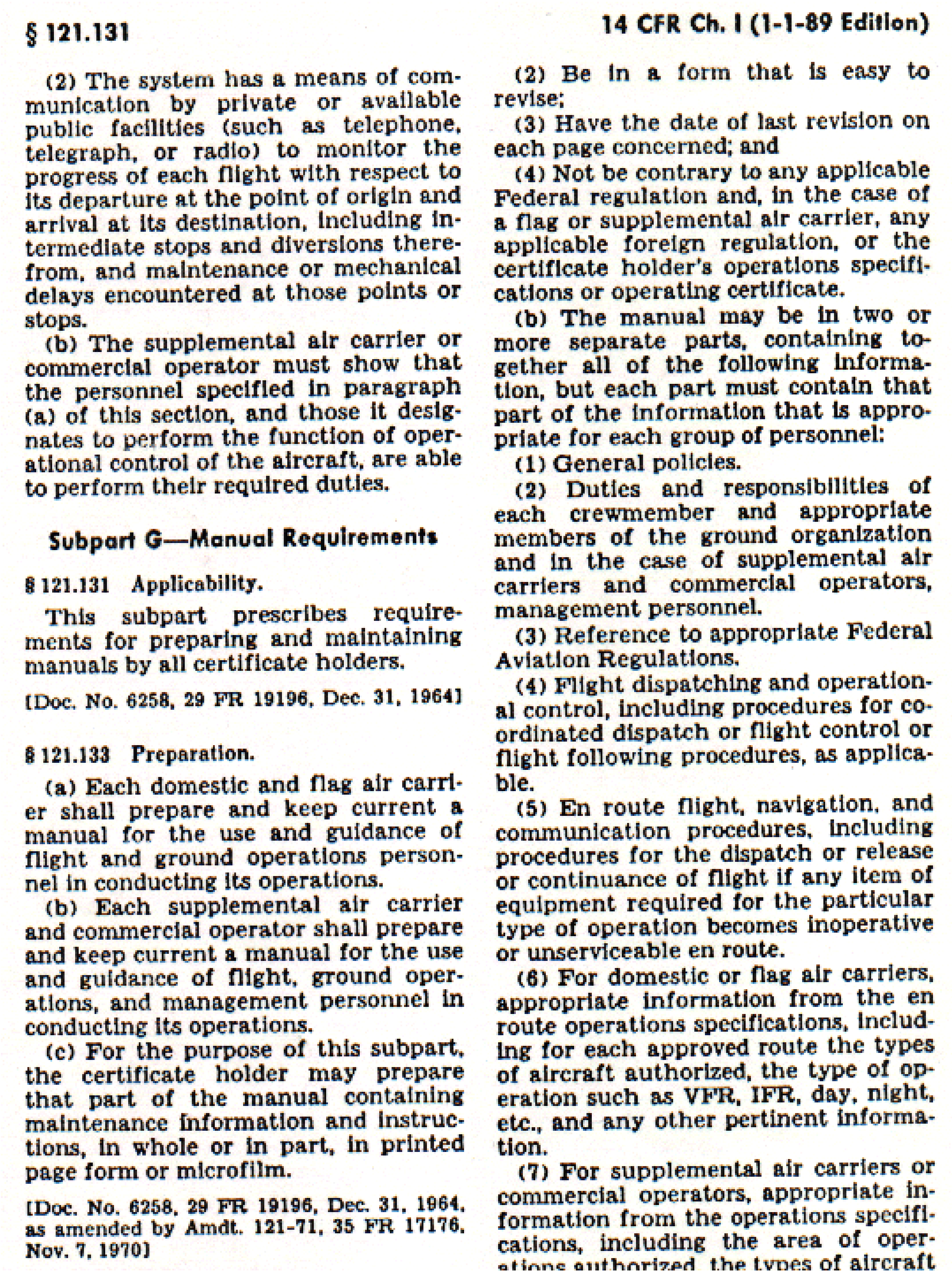
1796 *Mathem. Dict.* I. 2 To abbreviate fractions in arithmetic and algebra, is to lessen proportionally their terms, or the numerator and denominator.

37. Procedures, Policies, Laws, and Regulations

- Usually mostly text, created and used by people
- Information that is often extremely important to companies and highly-paid professionals because the cost of finding (or not finding) information can be high
- Often has high "intrinsic hypertext" character with many explicit and implicit links between content components
- Often follow structural conventions and standards with regular numbering and naming schemes

- Versioning and configuration requirements can pose problems
- Making this type of content computable or executable is a huge R&D area (XML standards like XACML, policy engines and wizards, expert systems)

38. Code of Federal Regulations



§ 121.131

14 CFR Ch. I (1-1-89 Edition)

(2) The system has a means of communication by private or available public facilities (such as telephone, telegraph, or radio) to monitor the progress of each flight with respect to its departure at the point of origin and arrival at its destination, including intermediate stops and diversions therefrom, and maintenance or mechanical delays encountered at those points or stops.

(b) The supplemental air carrier or commercial operator must show that the personnel specified in paragraph (a) of this section, and those it designates to perform the function of operational control of the aircraft, are able to perform their required duties.

Subpart G—Manual Requirements

§ 121.131 Applicability.

This subpart prescribes requirements for preparing and maintaining manuals by all certificate holders.

[Doc. No. 6258, 29 FR 19196, Dec. 31, 1964]

§ 121.133 Preparation.

(a) Each domestic and flag air carrier shall prepare and keep current a manual for the use and guidance of flight and ground operations personnel in conducting its operations.

(b) Each supplemental air carrier and commercial operator shall prepare and keep current a manual for the use and guidance of flight, ground operations, and management personnel in conducting its operations.

(c) For the purpose of this subpart, the certificate holder may prepare that part of the manual containing maintenance information and instructions, in whole or in part, in printed page form or microfilm.

[Doc. No. 6258, 29 FR 19196, Dec. 31, 1964, as amended by Amdt. 121-71, 35 FR 17176, Nov. 7, 1970]

(2) Be in a form that is easy to revise;

(3) Have the date of last revision on each page concerned; and

(4) Not be contrary to any applicable Federal regulation and, in the case of a flag or supplemental air carrier, any applicable foreign regulation, or the certificate holder's operations specifications or operating certificate.

(b) The manual may be in two or more separate parts, containing together all of the following information, but each part must contain that part of the information that is appropriate for each group of personnel:

(1) General policies.

(2) Duties and responsibilities of each crewmember and appropriate members of the ground organization and in the case of supplemental air carriers and commercial operators, management personnel.

(3) Reference to appropriate Federal Aviation Regulations.

(4) Flight dispatching and operational control, including procedures for coordinated dispatch or flight control or flight following procedures, as applicable.

(5) En route flight, navigation, and communication procedures, including procedures for the dispatch or release or continuance of flight if any item of equipment required for the particular type of operation becomes inoperative or unserviceable en route.

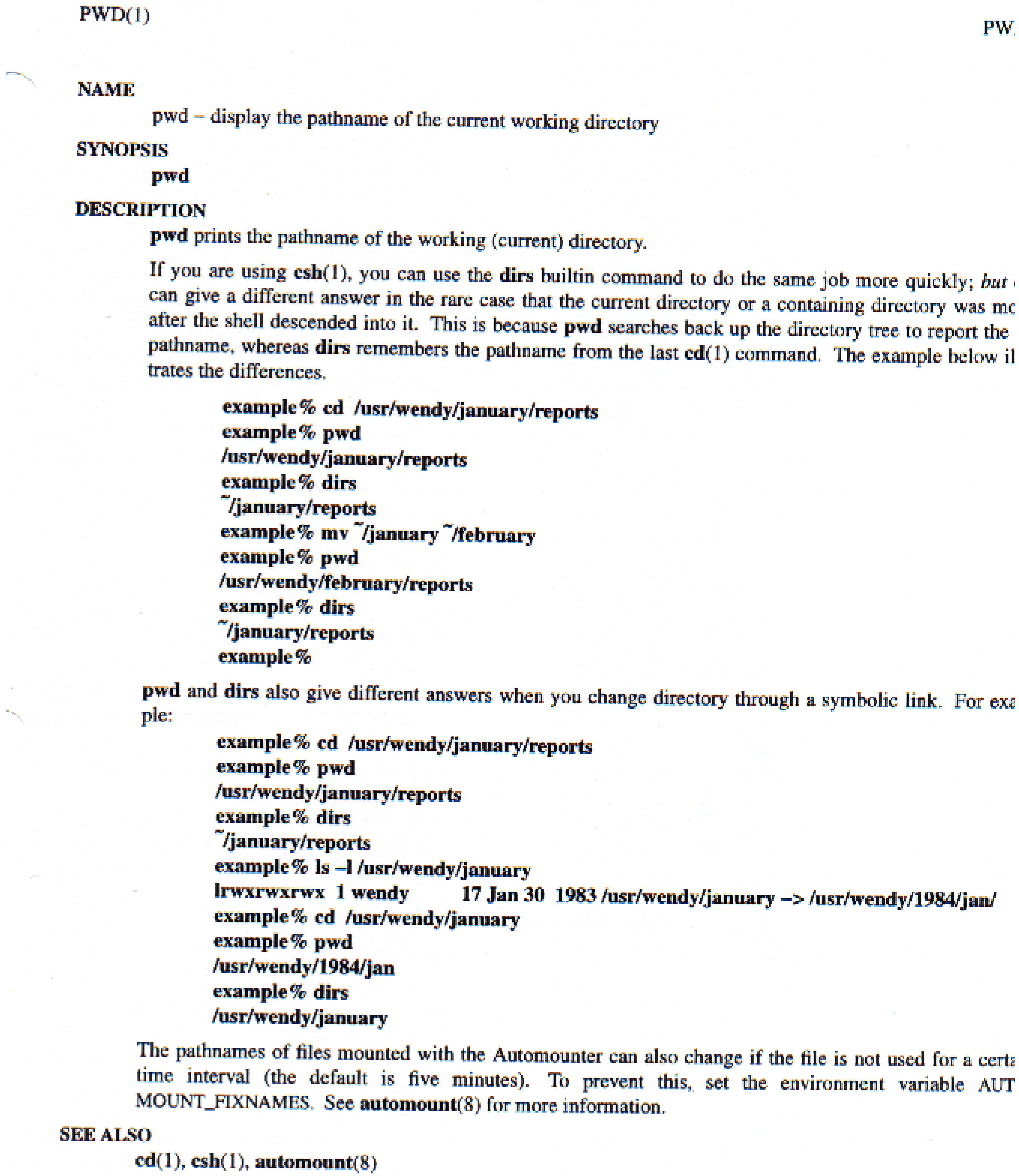
(6) For domestic or flag air carriers, appropriate information from the en route operations specifications, including for each approved route the types of aircraft authorized, the type of operation such as VFR, IFR, day, night, etc., and any other pertinent information.

(7) For supplemental air carriers or commercial operators, appropriate information from the operations specifications, including the area of operations authorized, the types of aircraft

39. Catalogs

- Many different types
- Some are extracted from ERP system or product database
- Often contain a mixture of structured and unstructured content
- Often a challenge to match the user's vocabulary and ontology for a product domain

40. Software "Man Page"



PWD(1) PW

NAME

pwd – display the pathname of the current working directory

SYNOPSIS

pwd

DESCRIPTION

pwd prints the pathname of the working (current) directory.

If you are using **csh**(1), you can use the **dirs** builtin command to do the same job more quickly; *but* can give a different answer in the rare case that the current directory or a containing directory was mc after the shell descended into it. This is because **pwd** searches back up the directory tree to report the pathname, whereas **dirs** remembers the pathname from the last **cd**(1) command. The example below il trates the differences.

```
example% cd  /usr/wendy/january/reports
example% pwd
/usr/wendy/january/reports
example% dirs
~/january/reports
example% mv ~/january ~/february
example% pwd
/usr/wendy/february/reports
example% dirs
~/january/reports
example%
```

pwd and **dirs** also give different answers when you change directory through a symbolic link. For exa ple:

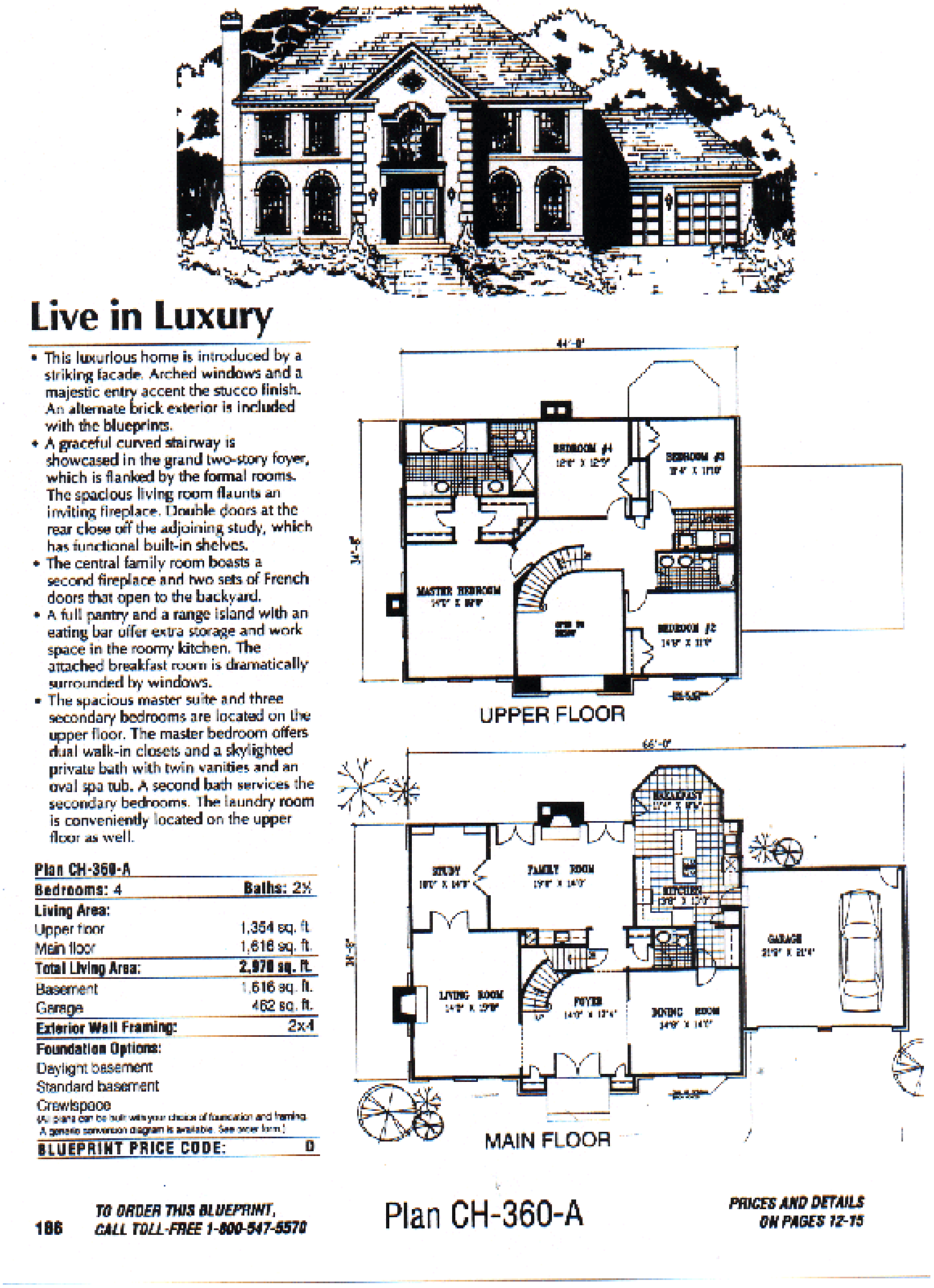
```
example% cd  /usr/wendy/january/reports
example% pwd
/usr/wendy/january/reports
example% dirs
~/january/reports
example% ls –l /usr/wendy/january
lrwxrwxrwx  1 wendy       17 Jan 30  1983 /usr/wendy/january –> /usr/wendy/1984/jan/
example% cd  /usr/wendy/january
example% pwd
/usr/wendy/1984/jan
example% dirs
/usr/wendy/january
```

The pathnames of files mounted with the Automounter can also change if the file is not used for a certa time interval (the default is five minutes). To prevent this, set the environment variable AUT MOUNT_FIXNAMES. See **automount**(8) for more information.

SEE ALSO

cd(1), **csh**(1), **automount**(8)

41. Home Blueprint


44'-0"
BEDROOM #4
BEDROOM #3
MASTER BEDROOM
OPEN TO BELOW
BEDROOM #2
UPPER FLOOR
66'-0"
BREAKFAST
STUDY
FAMILY ROOM
KITCHEN
GARAGE
LIVING ROOM
FOYER
DINING ROOM
MAIN FLOOR

Live in Luxury

- This luxurious home is introduced by a striking facade. Arched windows and a majestic entry accent the stucco finish. An alternate brick exterior is included with the blueprints.
- A graceful curved stairway is showcased in the grand two-story foyer, which is flanked by the formal rooms. The spacious living room flaunts an inviting fireplace. Double doors at the rear close off the adjoining study, which has functional built-in shelves.
- The central family room boasts a second fireplace and two sets of French doors that open to the backyard.
- A full pantry and a range island with an eating bar offer extra storage and work space in the roomy kitchen. The attached breakfast room is dramatically surrounded by windows.
- The spacious master suite and three secondary bedrooms are located on the upper floor. The master bedroom offers dual walk-in closets and a skylighted private bath with twin vanities and an oval spa tub. A second bath services the secondary bedrooms. The laundry room is conveniently located on the upper floor as well.

Plan CH-360-A	
Bedrooms: 4	**Baths:** 2½
Living Area:	
Upper floor	1,354 sq. ft.
Main floor	1,616 sq. ft.
Total Living Area:	**2,970 sq. ft.**
Basement	1,616 sq. ft.
Garage	462 sq. ft.
Exterior Wall Framing:	2x4
Foundation Options:	
Daylight basement	
Standard basement	
Crawlspace	

(All plans can be built with your choice of foundation and framing. A generic conversion diagram is available. See order form.)

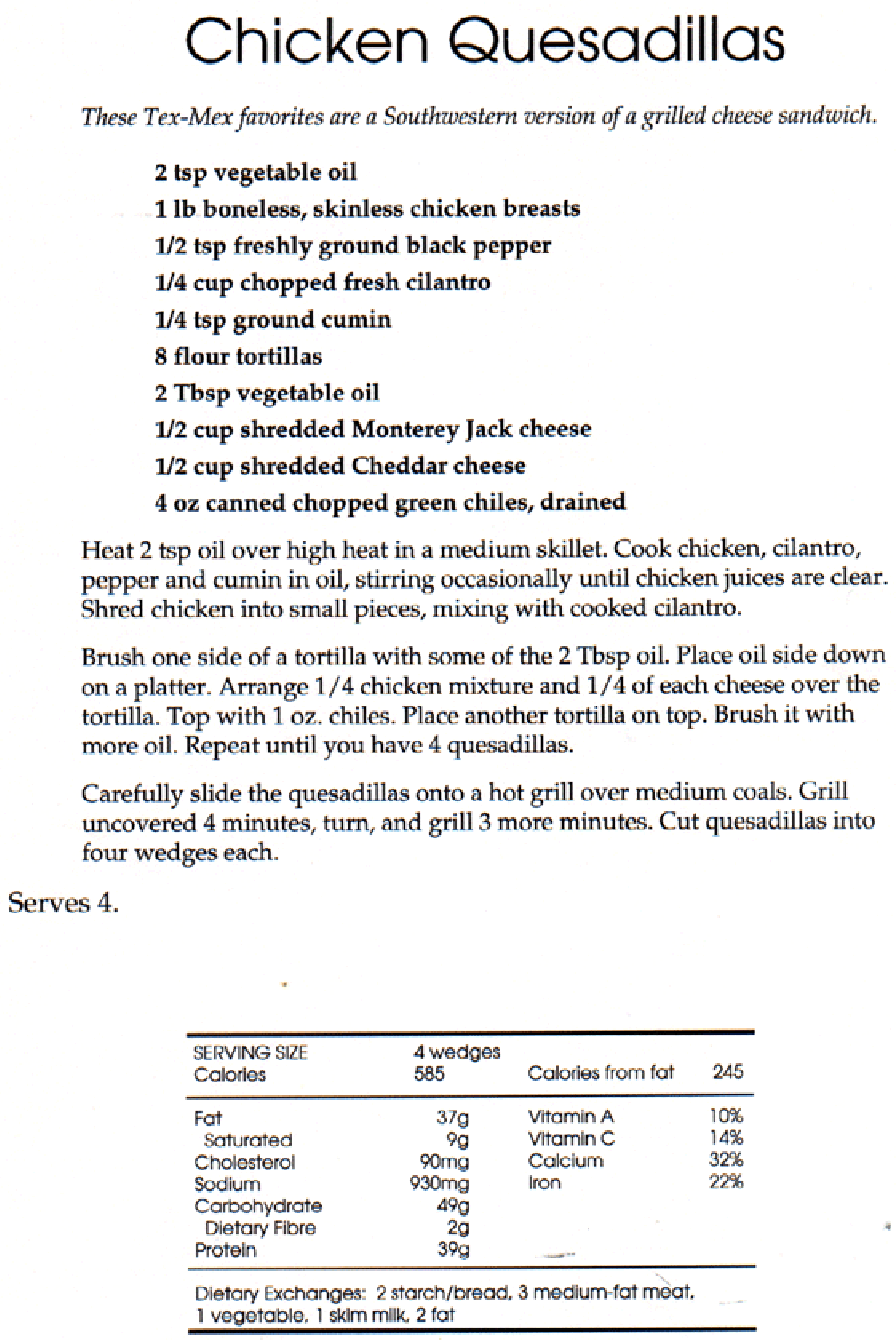
BLUEPRINT PRICE CODE: D

186 ***TO ORDER THIS BLUEPRINT, CALL TOLL-FREE 1-800-547-5570***

Plan CH-360-A

PRICES AND DETAILS ON PAGES 12-15

42. Recipe



Chicken Quesadillas

These Tex-Mex favorites are a Southwestern version of a grilled cheese sandwich.

2 tsp vegetable oil
1 lb boneless, skinless chicken breasts
1/2 tsp freshly ground black pepper
1/4 cup chopped fresh cilantro
1/4 tsp ground cumin
8 flour tortillas
2 Tbsp vegetable oil
1/2 cup shredded Monterey Jack cheese
1/2 cup shredded Cheddar cheese
4 oz canned chopped green chiles, drained

Heat 2 tsp oil over high heat in a medium skillet. Cook chicken, cilantro, pepper and cumin in oil, stirring occasionally until chicken juices are clear. Shred chicken into small pieces, mixing with cooked cilantro.

Brush one side of a tortilla with some of the 2 Tbsp oil. Place oil side down on a platter. Arrange 1/4 chicken mixture and 1/4 of each cheese over the tortilla. Top with 1 oz. chiles. Place another tortilla on top. Brush it with more oil. Repeat until you have 4 quesadillas.

Carefully slide the quesadillas onto a hot grill over medium coals. Grill uncovered 4 minutes, turn, and grill 3 more minutes. Cut quesadillas into four wedges each.

Serves 4.

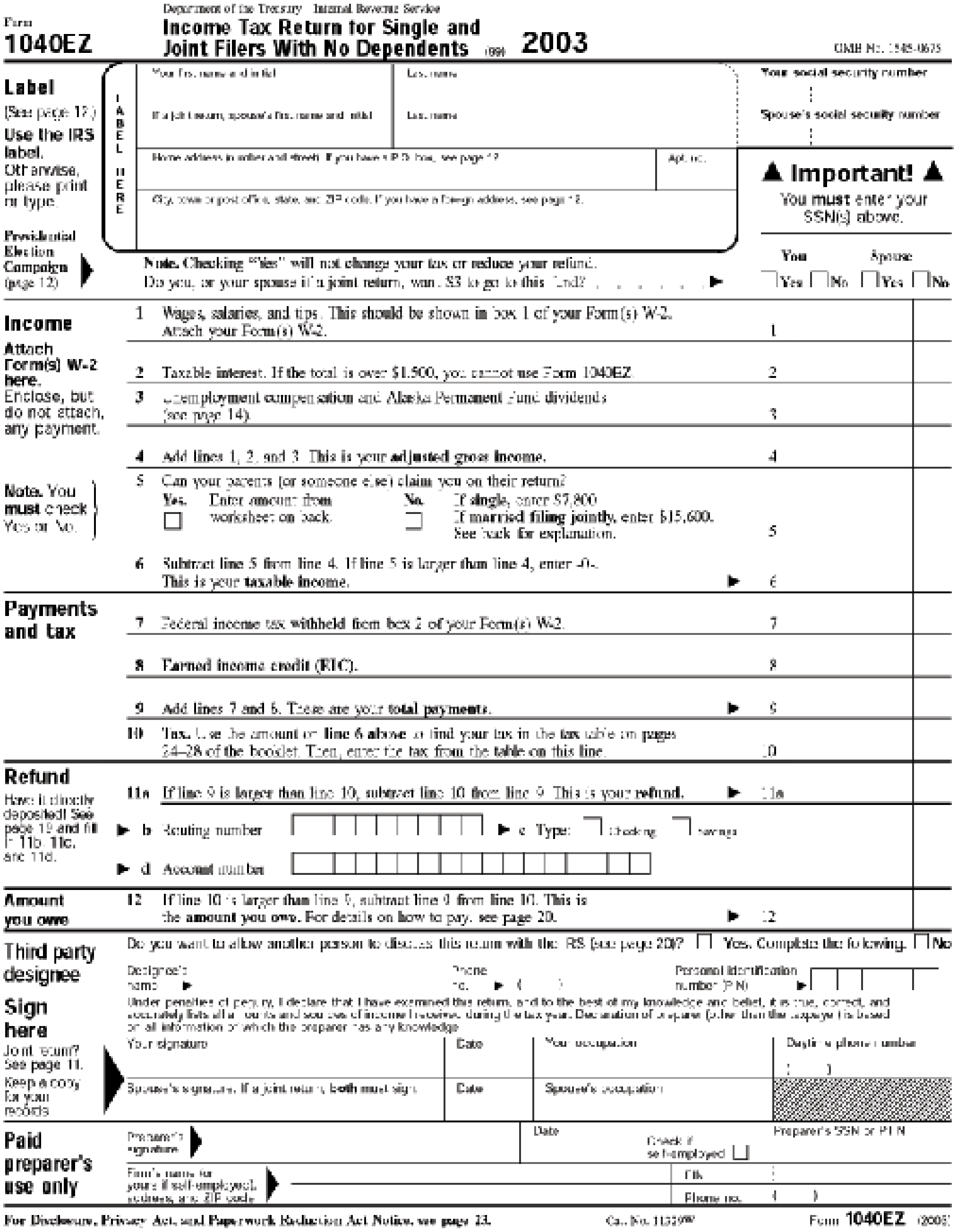
SERVING SIZE	4 wedges		
Calories	585	Calories from fat	245
Fat	37g	Vitamin A	10%
Saturated	9g	Vitamin C	14%
Cholesterol	90mg	Calcium	32%
Sodium	930mg	Iron	22%
Carbohydrate	49g		
Dietary Fibre	2g		
Protein	39g		

Dietary Exchanges: 2 starch/bread, 3 medium-fat meat, 1 vegetable, 1 skim milk, 2 fat

43. Transaction Documents

- Printed or electronic forms
- Data-intensive, designed to capture and present small information components
- Inputs and outputs of business processes and often created and consumed by computers
- Few and somewhat arbitrary presentational characteristics
- Strongly datatyped with field length, range and value, other restrictions

44. Tax Form



Form **1040EZ**

Department of the Treasury—Internal Revenue Service

Income Tax Return for Single and Joint Filers With No Dependents (99) **2003**

OMB No. 1545-0675

Label (See page 12.) **Use the IRS label.** Otherwise, please print or type.

LABEL HERE

Your first name and initial | Last name | Your social security number

If a joint return, spouse's first name and initial | Last name | Spouse's social security number

Home address (number and street). If you have a P.O. box, see page 12. | Apt. no.

City, town or post office, state, and ZIP code. If you have a foreign address, see page 12.

▲ **Important!** ▲ You **must** enter your SSN(s) above.

Presidential Election Campaign (page 12)

Note. Checking "Yes" will not change your tax or reduce your refund.

Do you, or your spouse if a joint return, want $3 to go to this fund? ▶ You □ Yes □ No Spouse □ Yes □ No

Income

Attach Form(s) W-2 here. Enclose, but do not attach, any payment.

1 Wages, salaries, and tips. This should be shown in box 1 of your Form(s) W-2. Attach your Form(s) W-2. 1

2 Taxable interest. If the total is over $1,500, you cannot use Form 1040EZ. 2

3 Unemployment compensation and Alaska Permanent Fund dividends (see page 14). 3

4 Add lines 1, 2, and 3. This is your **adjusted gross income**. 4

Note. You **must** check Yes or No.

5 Can your parents (or someone else) claim you on their return?
Yes. Enter amount from worksheet on back. □
No. If **single**, enter $7,800. If **married filing jointly**, enter $15,600. See back for explanation. □ 5

6 Subtract line 5 from line 4. If line 5 is larger than line 4, enter -0-. This is your **taxable income**. ▶ 6

Payments and tax

7 Federal income tax **withheld** from box 2 of your Form(s) W-2. 7

8 Earned income credit (EIC). 8

9 Add lines 7 and 8. These are your **total payments**. ▶ 9

10 Tax. Use the amount on **line 6 above** to find your tax in the tax table on pages 24–28 of the booklet. Then, enter the tax from the table on this line. 10

Refund

Have it directly deposited! See page 19 and fill in 11b, 11c, and 11d.

11a If line 9 is larger than line 10, subtract line 10 from line 9. This is your **refund**. ▶ 11a

▶ b Routing number □□□□□□□□□ ▶ c Type: □ Checking □ Savings

▶ d Account number □□□□□□□□□□□□□□□□□

Amount you owe

12 If line 10 is larger than line 9, subtract line 9 from line 10. This is the **amount you owe**. For details on how to pay, see page 20. ▶ 12

Third party designee

Do you want to allow another person to discuss this return with the IRS (see page 20)? □ **Yes.** Complete the following. □ **No**

Designee's name ▶ Phone no. ▶ ( ) Personal identification number (PIN) ▶ □□□□□

Sign here

Joint return? See page 11. Keep a copy for your records.

Under penalties of perjury, I declare that I have examined this return, and to the best of my knowledge and belief, it is true, correct, and accurately lists all amounts and sources of income I received during the tax year. Declaration of preparer (other than the taxpayer) is based on all information of which the preparer has any knowledge.

Your signature | Date | Your occupation | Daytime phone number ( )

Spouse's signature. If a joint return, **both** must sign. | Date | Spouse's occupation

Paid preparer's use only

Preparer's signature ▶ | Date | Check if self-employed □ | Preparer's SSN or PTIN

Firm's name (or yours if self-employed), address, and ZIP code ▶ | EIN | Phone no. ( )

For Disclosure, Privacy Act, and Paperwork Reduction Act Notice, see page 23. Cat. No. 11329W Form **1040EZ** (2003)

45. The Data/Document Chasm

- It's obviously a continuum... but transactional and narrative documents have traditionally been analyzed with different disciplines and use different tools, terminology, and techniques – little intersection
- Very different intellectual and domain roots
  - "Document analysis" for narrative documents: literary criticism, graphical design; electronic publishing
  - "Data modeling" for transactional documents: philosophy, linguistics, systems analysis; data automation

46. "Document Analysis" Methodology Summary

- Scope: One document type at a time
- Reuse focus: identify "boilerplate" content and repeating structural elements
- Heuristic rather than formal techniques
- Descriptive "text encoding" to capture idiosyncratic aspects of instances
- Typical textbook: Maler and Andaloussi. *Developing SGML DTDs: From Text to Model to Markup* (1996)

47. "Data Modeling" Methodology Summary

- Scope: multiple interrelated document types
- Reuse focus: identify the overlapping content in transformationally-related document types or in "base" and "extended" document components
- Prescriptive approach – design schemas to optimize processing in some information system (databases or W3C XML Schema)
- Formal techniques (e.g., schema normalization)
- Typical textbook: Carlis and Maguire. *Mastering Data Modeling.* (2001).

48. Document Analysis {and,or,vs} Data Modeling: Crossing the Data/Document Chasm

- Document Engineering harmonizes the terminology and emphasizes what they have in common rather than highlighting their differences
  - Identifying the presentational, content, and structural components and defining their relationships to each other
  - Identifying "good" content components
  - Designing, describing, and organizing components to facilitate their reuse
  - Assembling hierarchical document models that organize components according to the requirements of a specific context for information exchange

49. {and,or,vs} Business Process Analysis

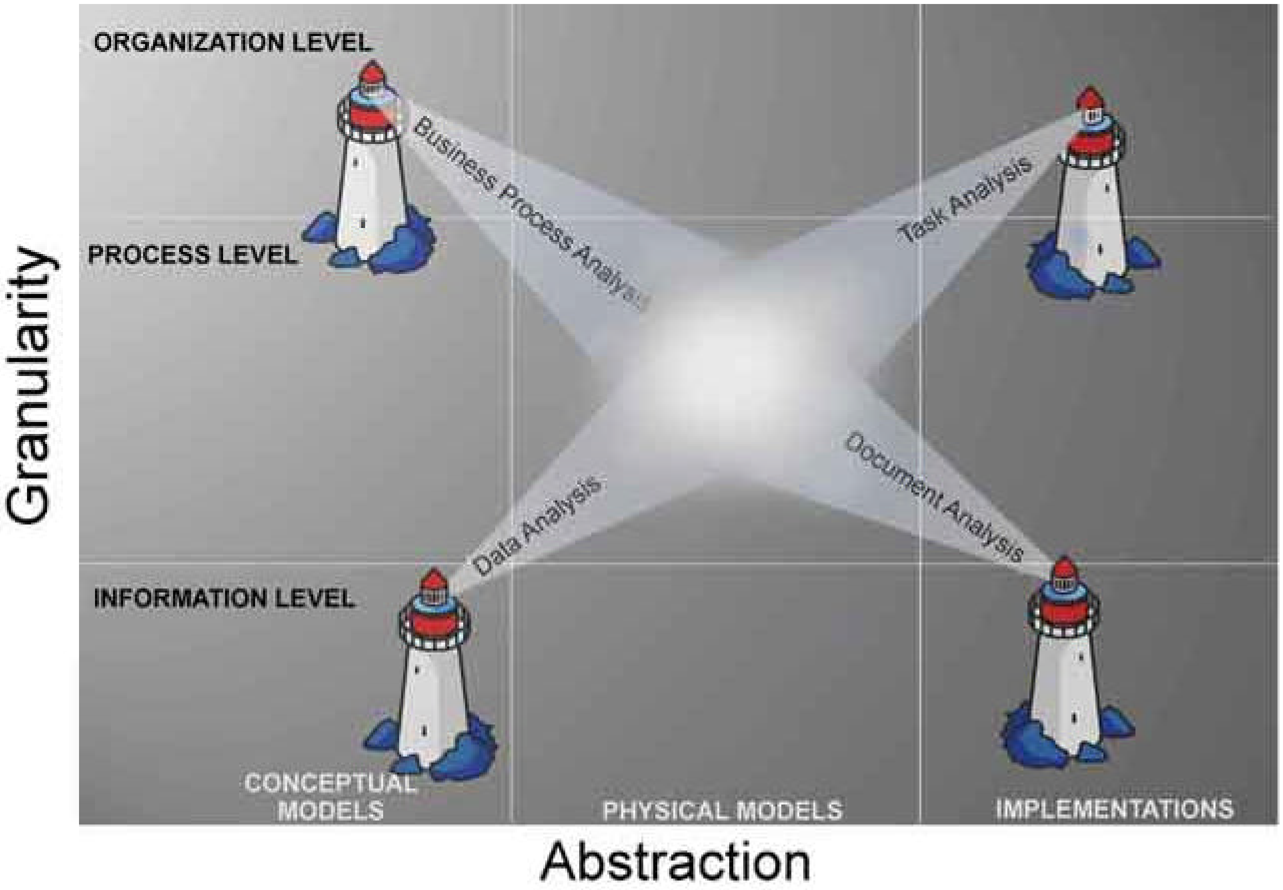
- Business process analysis begins with an abstract or broadly scoped perspective on business activities
- Emphasizes "Does this work from a business perspective?"
- Inherently a "top down" approach that starts with business models and processes and gets to the "document payloads" only at the end
- In contrast, the document analysis and data modeling approaches focus from the beginning on the structure and content of the "document payload" that will be exchanged – a "bottom up" approach that emphasizes "Does this work from a technical perspective?"

50. {and,or,vs} User Task Analysis

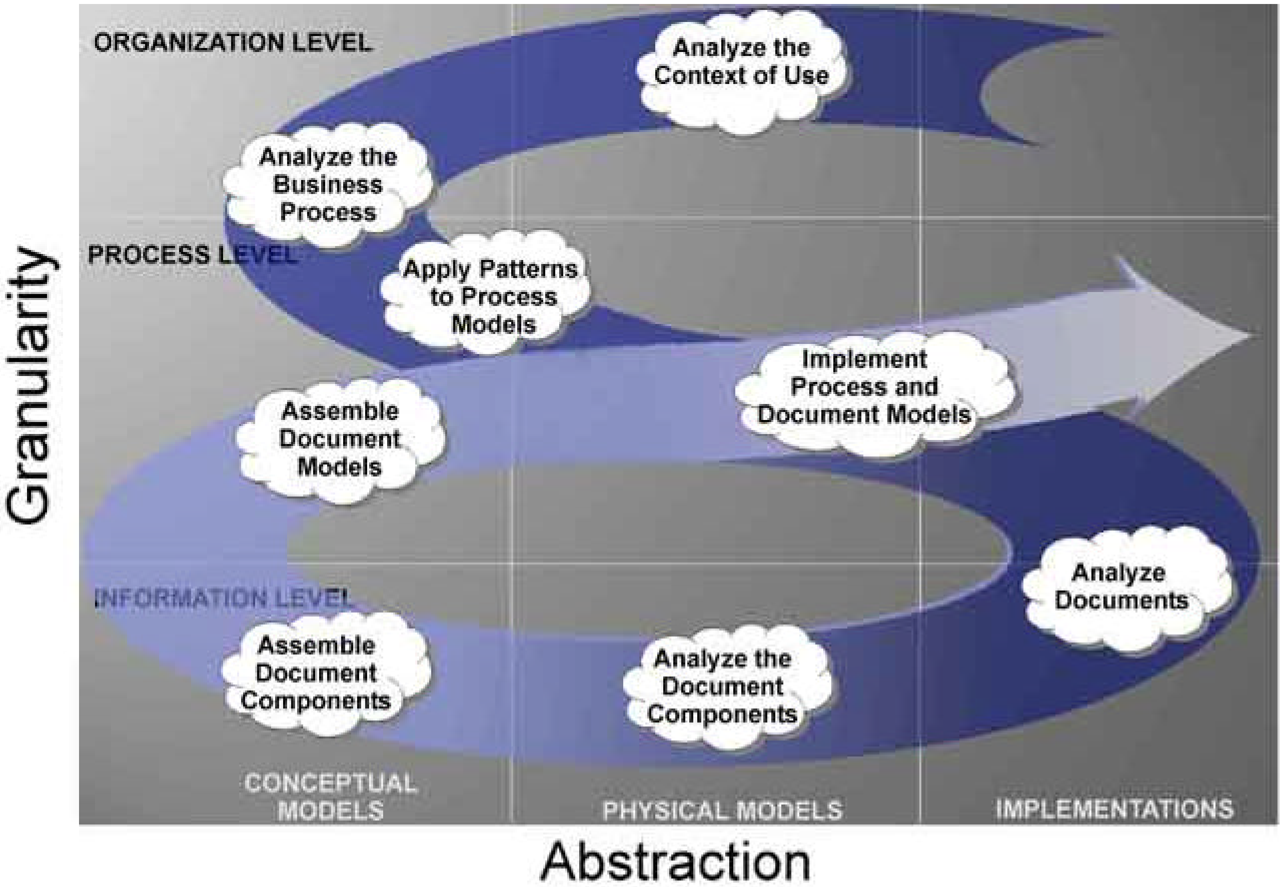
- Task analysis (or user analysis) is the observation of people performing the tasks or use cases when the application or system must support human interfaces and not just other applications
- Task analysis and document analysis are closely related; document analysis reveals candidate information components and task analysis reveals rules about their intent and usage.
- Task analysis is especially important when few documents or information sources exist because human problems or errors can suggest that important information is missing

51. A Unified View of Analysis and Modeling: Meeting in the Middle

- Document Engineering unifies four different disciplines or methods of analysis that until now have had little intersection
- We need to achieve both business and technical interoperability – the former is necessary but insufficient for the latter
- We need models of the desired business processes and the documents that they will produce and consume at the same level of detail and implementability
- This is represented in the Model Matrix as "meeting in the middle"
- Document Engineering is a systematic approach for "getting to the middle"



52. The Document Engineering Approach


ORGANIZATION LEVEL
Analyze the Context of Use
Analyze the Business Process
PROCESS LEVEL
Apply Patterns to Process Models
Implement Process and Document Models
Assemble Document Models
INFORMATION LEVEL
Analyze Documents
Assemble Document Components
Analyze the Document Components
CONCEPTUAL MODELS
PHYSICAL MODELS
IMPLEMENTATIONS
Granularity
Abstraction

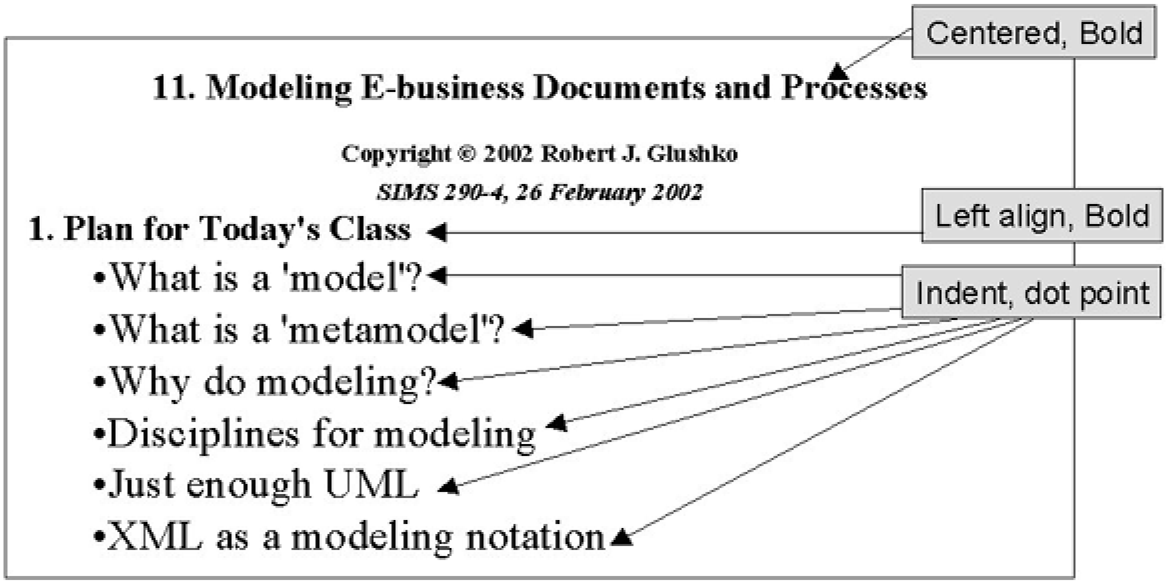
53. Three Types of Information In Documents

- We need a vocabulary to classify different kinds of information that we find in documents and sets of data
  - *Content* – "what does it mean" information
  - *Structure* – "where is it" or "how it is organized or assembled" information
  - *Presentation* – "how does it look" or "how is it displayed" information

54. Presentation Information

- Human-oriented attributes for visual (or other sensory) differentiation (type font, type size, color, background, indentation, pitch, ...)
- Implementation specific (e.g. Web vs print vs auditory)
- Good user interface design correlates this with structural or content information, increasingly so as we move away from transactional end of spectrum

55. Presentation View of a Lecture Slide



11. Modeling E-business Documents and Processes

Copyright © 2002 Robert J. Glushko

SIMS 290-4, 26 February 2002

1. Plan for Today's Class

•What is a 'model'?

•What is a 'metamodel'?

•Why do modeling?

•Disciplines for modeling

•Just enough UML

•XML as a modeling notation

- Presentation can indicate structural or content information:
    - "Copyright © 2002 Robert J. Glushko" has different formatting than the title, indicating that it is another candidate component for analysis
- But can also conceal it:
    - The presentation does not differentiate between the topic number (11.) and the Topic Title "Modeling E-business Documents and Processes", but we can deduce that they are two separate components by their functional independence

56. Extracting Presentation Rules

- Presentation affects structure and content by applying transformation rules to them
- To understand the structure and content we must identify and record what the rules of the transformation were
- Explicit transform rules can be encoded in templates, stylesheets or source code

57. But Sometimes Rules Can't be Extracted

- No access to source formats or source code
- Rules may be inaccessible in source formats ("override" formatting in word processors instead of style tags)
- Rules don't exist or are inconsistently followed (author has "fontitis" with "ransom note" presentation style)

58. Correlations or Conventions with Presentation Information

- Color, pitch, other perceptual dimensions can be correlated with semantic distinctions
- Type size is usually correlated with the structural hierarchy

- Content types can have characteristic layouts or text attributes
- Adjacency can suggest a semantic relationship, like that between figure and caption
- Presentation order is sometimes semantically significant

59. Presentations that Mask Content Components

- A form may ask you to enter your address this way

```
Address:
        Line 1: _________________
        Line 2: _________________
        City: ____________  State: ________  ZipCode: _________
```

- But "line 1" and "line 2" are presentation labels that are not useful for any purpose other than printing out an address label
- They are not candidate content components
- They are masking content components like "number," "street," etc.

60. Generated or Derived Components

- "Table of Contents," "Permuted Index," and list of figures, tables, or other types of components can usually be generated or derived from other components and are not components in their own right
- Similarly, if "ExtendedPrice" is "Quantity" x "UnitPrice" we might only want the latter two components in our model since collecting that first one separately could lead to data integrity problems

61. Tables [1]

Document Engineering	Glushko	MW 2:00-3:30	South Hall 202
XML Technologies	Milowski	TTh 2:00-3:30	South Hall 110
Secrets of Consulting	Downes	Th 12:00-2:00	South Hall 205

62. Tables [2]

- A table is a systematic pattern of relationships among content, structure, and presentation information, typically represented in a set of embedded rectangular grids
- A table presents information by organizing some set of meaningful elements to emphasize the relationships between the elements and the manner in which combinations of elements interact
- Most tables (90% of them?) follow regular matrix or structural patterns in which the organization of information (and the presentation applied to it) is consistent with (or reinforces) the relationships between the content that is contained in the cells or regions defined by the matrix
- A transactional document type is often little more than a table (of items ordered, purchased, shipped, etc.) with some additional information about the parties to the transaction
- A table embedded in another document might be best understood as a "mini-document type" of its own, especially

when the "containing" document type is more narrative than transactional

63. Analyzing Tables

- The nature of these relationships is often explicitly represented in the headings for rows, columns, or other structural elements

- When the relationships are not explicit, they can often be determined by analyzing the datatypes and content of the cells or the manner in which the content varies from cell to cell

- The mere existence or non-existence of values within the cells can have semantic significance.

64. The Trouble With Tables

- The obvious and optimal analysis of the information in a table is in terms of these content relationships:

```
<Courses>
  <Course>
    <Title>Document Engineering</Title>
    <Instructor>Glushko</Instructor>
    <Schedule>MW 2:00-3:30</Schedule>
    <Location>South Hall 202</Location>
  </Course>
  <Course>
    <Title>XML Technologies</Title>
    <Instructor>Milowski</Instructor>
    <Schedule>TTh 2:00-3:30</Schedule>
    <Location>South Hall 110</Location>
    </Course>
  <Course>
    <Title>Secrets of Consulting</Title>
    <Instructor>Downes</Instructor>
    <Schedule>Th 12:00-2:00</Schedule>
    <Location>South Hall 205</Location>
    </Course>
</Courses>
```

65. The Trouble With Tables [2]

- Unfortunately, the predictable geometry for organizing their content has led to tables being analyzed and implemented in terms of the structure of their presentation rather than a set of content relationships

- Put another way, this means that document types which are inherently a set of content relationships are usually analyzed and implemented in terms of their presentational features (rows, columns, headings, spans, etc.)

- So what you typically get is something like:

```
<table>
<row>
    <cell>Document Engineering</cell>
    <cell>Glushko</cell>
    <cell>MW 2:00-3:30</cell>
    <cell>South Hall 202</cell>
</row>
<row>
    <cell>XML Technologies</cell>
    <cell>Milowski</cell>
    <cell>TTh 2:00-3:30</cell>
    <cell>South Hall 110</cell>
```

```
</row>
<row>
    <cell>Secrets of Consulting</cell>
    <cell>Downes</cell>
    <cell>Th 12:00-2:00</cell>
    <cell>South Hall 205</cell>
</row>
</table>
```

66. The Trouble With Tables [3]

- And in addition to the problem that most tables aren't represented in ways that capture their "tablehood" essence, it has been estimated that 95% of the information marked up as <TABLE> on the web is not really a table

- And some small percentage of things that are tables according to our definition defy content encoding because they combine content, structure, and presentation in ways that are often impossible to untangle or that are highly idiosyncratic but conventional

    - Periodic Table (http://pearl1.lanl.gov/periodic/default.htm)

67. Structural Information

- The "where is it ..." components

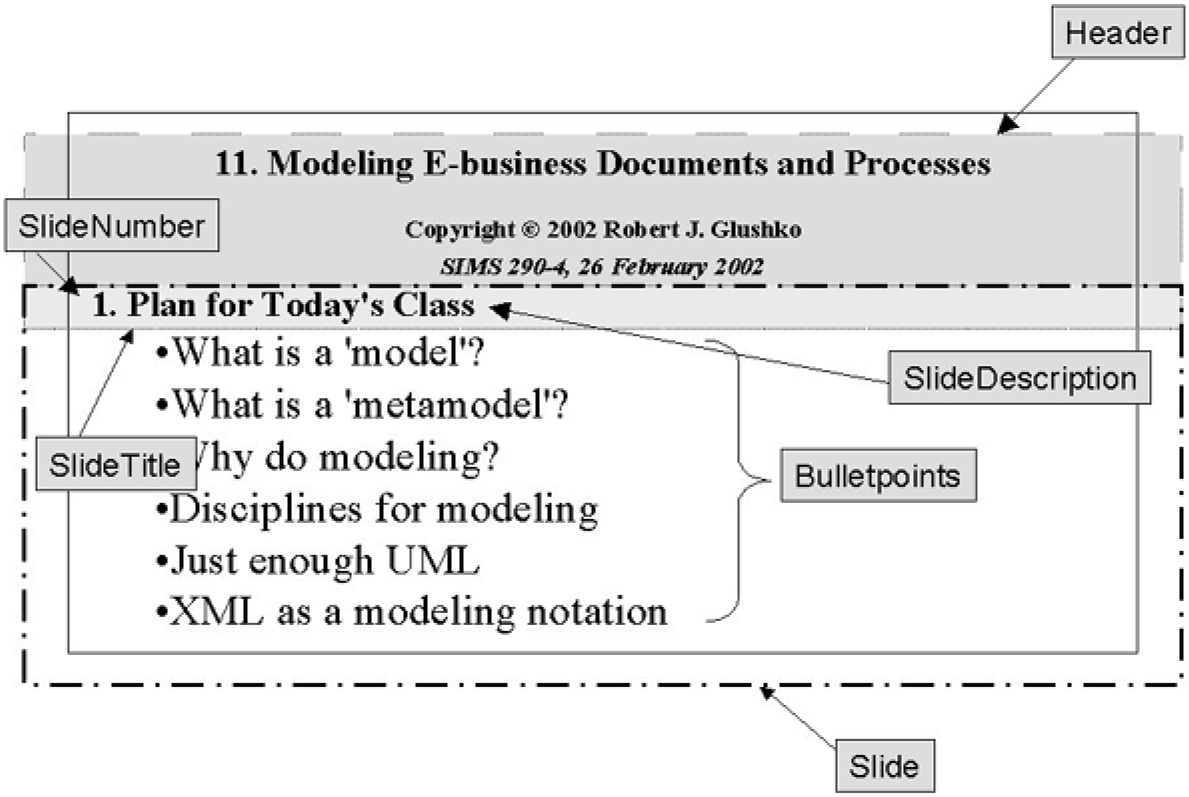
- Physical piece of a document (e.g. table, section, title, header, footer)

- The structural components provide the hierarchical "skeleton" or "scaffold" into which the content components are arranged

- The structure also provides a framework for presentation

- Structures are often hierarchical - one component can contain others

68. Structural View of a Lecture Slide



11. Modeling E-business Documents and Processes

Copyright © 2002 Robert J. Glushko

SIMS 290-4, 26 February 2002

1. Plan for Today's Class

- •What is a 'model'?
- •What is a 'metamodel'?
- •Why do modeling?
- •Disciplines for modeling
- •Just enough UML
- •XML as a modeling notation

- Each lecture has a similar structure: slides, each containing bullet points, each of which contains the content, which is different each time

- The structure of each slide is the same: a Header, containing metadata about the lecture and the topic of the slide, and the Body, which contains the content. Knowing this structure does not tell us what the lecture may be about – it is not the content.

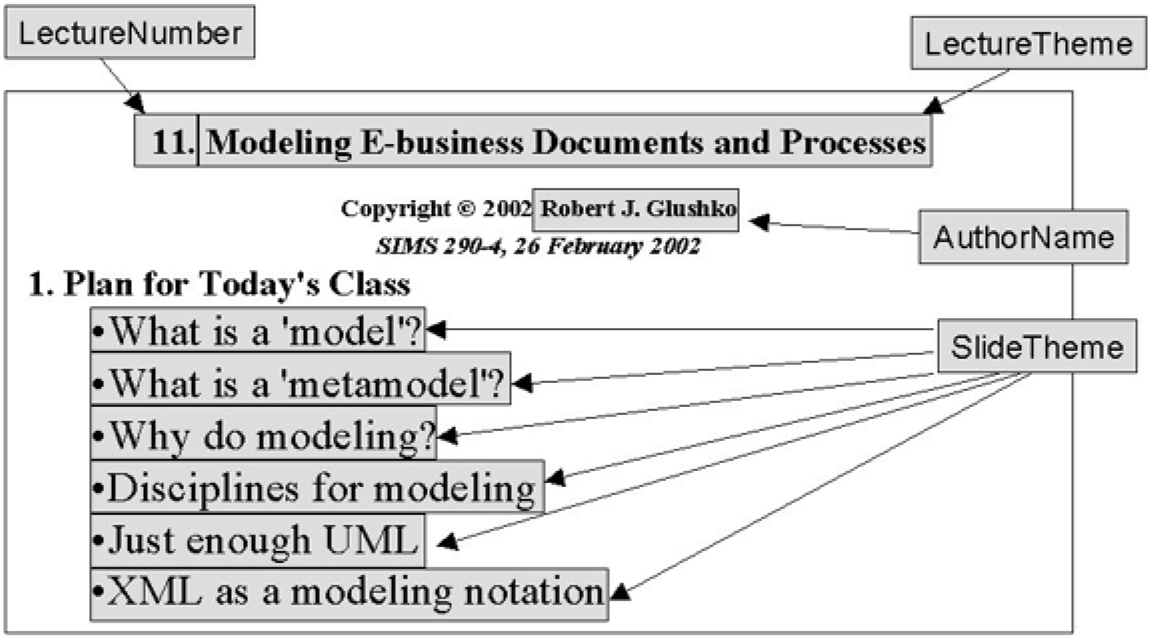
- The sequence of the bullet points corresponds to the sequence of the content in the lecture, so it is also structurally significant.

69. Content Components

- Content components are the "nouns" in our documents or sets of data – things like "topic," "summary," "name," "address," "price"

- These are the "what is it..." components

70. Content View of a Lecture Slide



- Concentrate of the meaning of the numbers and text that is left after we've isolated them from the presentation and structure

71. Analyzing "Possible Values"

- It is critical to capture any rules governing the possible values for a component
- Sometimes possible values are conventional, fixed, and span the entire semantic range for some domain (days of week, AM/PM)
- Determine who can control the value sets (internal [Manufacturer part #s] vs external [Bar codes])
- Patterns like regular expressions are often useful but not sufficient for validation
- And if the set of possible values isn't well motivated, fix it in your component design

72. Code Sets

- Code sets are constrained sets of values that are often completely arbitrary
- The ISO code sets for countries (3166), currencies (4217), quantities and units of measure (31) are the bedrock ones that you should generally defer to without question

73. The Simplest Model

- The simplest or minimal information component model is a glossary – a list of the words used to describe or name the "things of significance" and what they mean
- This simple data model is augmented as attributes or characteristics of the significant things are identified and recorded
- The model is further developed as relationships or associations or links between the "significant things" are identified

and recorded

74. The Modeling Artifacts

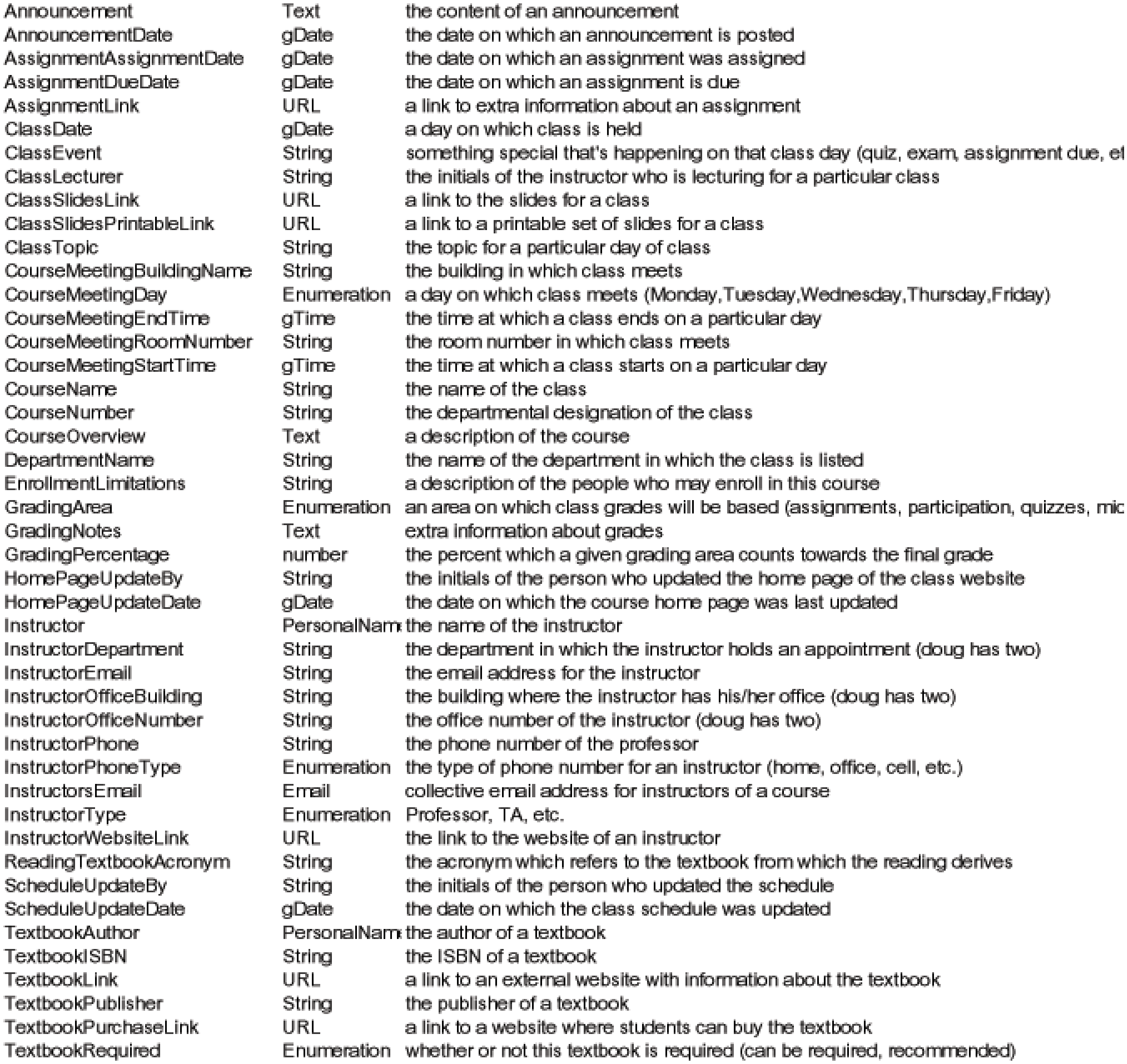
- We've chosen or developed a set of recommended modeling artifacts for each phase of the Document Engineering approach
- There is a natural progression that yields some overlap or correlation between them as later artifacts refine or consolidate earlier ones
- These artifacts have evolved to optimize the "step size" and to encourage more systematic, traceable, and predictable efforts

75. Harvesting Components

- As we identify candidate content components, we need to record its properties (or attributes or behaviors) that let us understand it and distinguish it from other ones
- A practical way to do this for each document or information source being analyzed, create a table or spreadsheet containing the candidate component and the metadata useful in understanding and distinguishing it from other ones

76. Example Harvest

- Example harvest from Course Syllabus project at UC Berkeley (Cracraft and de Larios-Heiman, 2005)



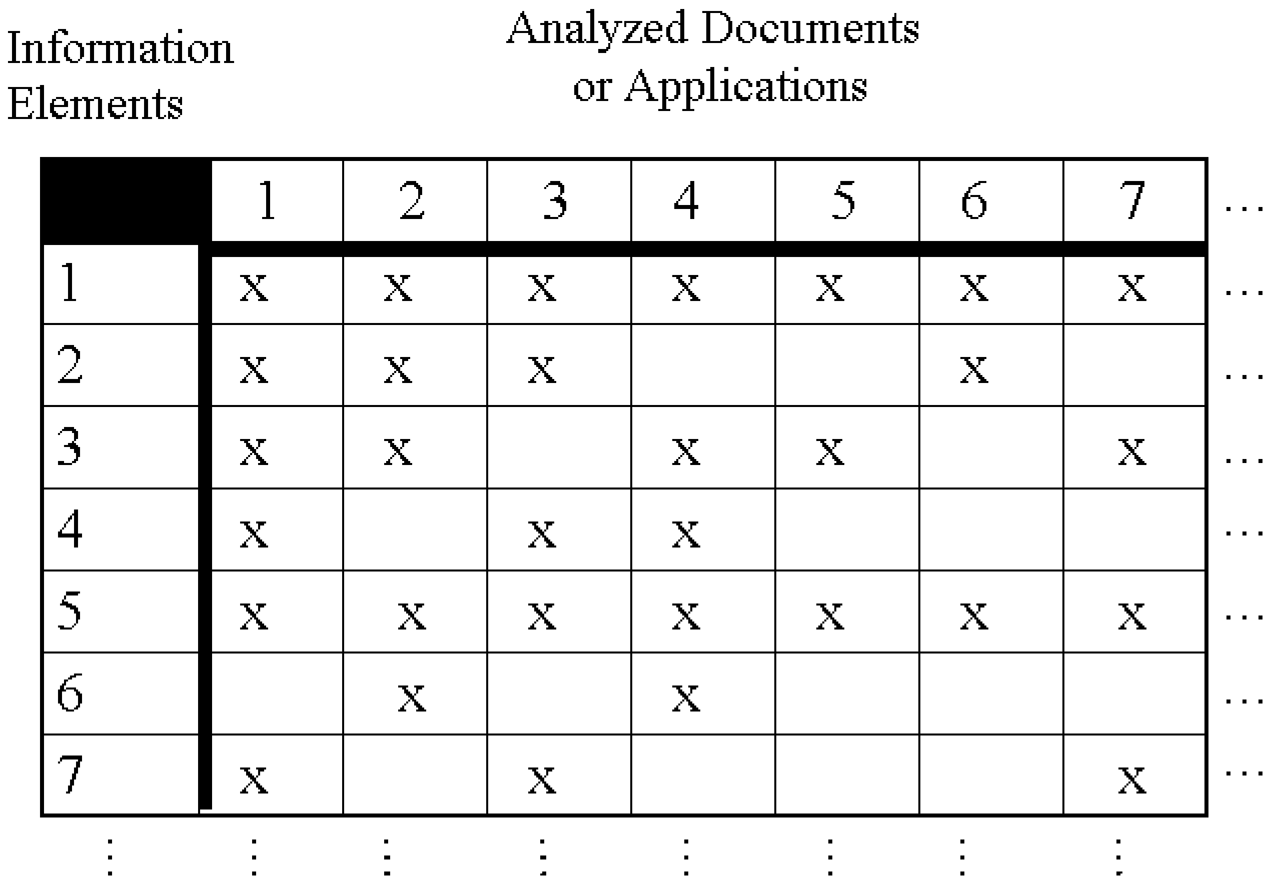
Announcement	Text	the content of an announcement
AnnouncementDate	gDate	the date on which an announcement is posted
AssignmentAssignmentDate	gDate	the date on which an assignment was assigned
AssignmentDueDate	gDate	the date on which an assignment is due
AssignmentLink	URL	a link to extra information about an assignment
ClassDate	gDate	a day on which class is held
ClassEvent	String	something special that's happening on that class day (quiz, exam, assignment due, et
ClassLecturer	String	the initials of the instructor who is lecturing for a particular class
ClassSlidesLink	URL	a link to the slides for a class
ClassSlidesPrintableLink	URL	a link to a printable set of slides for a class
ClassTopic	String	the topic for a particular day of class
CourseMeetingBuildingName	String	the building in which class meets
CourseMeetingDay	Enumeration	a day on which class meets (Monday,Tuesday,Wednesday,Thursday,Friday)
CourseMeetingEndTime	gTime	the time at which a class ends on a particular day
CourseMeetingRoomNumber	String	the room number in which class meets
CourseMeetingStartTime	gTime	the time at which a class starts on a particular day
CourseName	String	the name of the class
CourseNumber	String	the departmental designation of the class
CourseOverview	Text	a description of the course
DepartmentName	String	the name of the department in which the class is listed
EnrollmentLimitations	String	a description of the people who may enroll in this course
GradingArea	Enumeration	an area on which class grades will be based (assignments, participation, quizzes, mic
GradingNotes	Text	extra information about grades
GradingPercentage	number	the percent which a given grading area counts towards the final grade
HomePageUpdateBy	String	the initials of the person who updated the home page of the class website
HomePageUpdateDate	gDate	the date on which the course home page was last updated
Instructor	PersonalNam	the name of the instructor
InstructorDepartment	String	the department in which the instructor holds an appointment (doug has two)
InstructorEmail	String	the email address for the instructor
InstructorOfficeBuilding	String	the building where the instructor has his/her office (doug has two)
InstructorOfficeNumber	String	the office number of the instructor (doug has two)
InstructorPhone	String	the phone number of the professor
InstructorPhoneType	Enumeration	the type of phone number for an instructor (home, office, cell, etc.)
InstructorsEmail	Email	collective email address for instructors of a course
InstructorType	Enumeration	Professor, TA, etc.
InstructorWebsiteLink	URL	the link to the website of an instructor
ReadingTextbookAcronym	String	the acronym which refers to the textbook from which the reading derives
ScheduleUpdateBy	String	the initials of the person who updated the schedule
ScheduleUpdateDate	gDate	the date on which the class schedule was updated
TextbookAuthor	PersonalNam	the author of a textbook
TextbookISBN	String	the ISBN of a textbook
TextbookLink	URL	a link to an external website with information about the textbook
TextbookPublisher	String	the publisher of a textbook
TextbookPurchaseLink	URL	a link to a website where students can buy the textbook
TextbookRequired	Enumeration	whether or not this textbook is required (can be required, recommended)

77. What Metadata to Record About Candidate Components

- What attributes about each type of content might we record in our analysis?
  - Names/synonyms/homonyms (what it is called)
  - Definition (what it "means")
  - Identifiers
  - Cardinality/Optionality (occurrence rules)
  - Restricted values, code sets, defaults

- Data Type (text, numbers, date, video)
- Relationships/Associations (participation in structures)
- Origin (Is this new information, or from some other source? Who maintains it?)
- Access (who is allowed to view/change/copy/etc. it)
- Permanence (is it static or dynamic? how often does it change?)
- Business processes in which it participates

78. Consolidating The Harvest



Information Elements

Analyzed Documents or Applications

	1	2	3	4	5	6	7	…
1	x	x	x	x	x	x	x	…
2	x	x	x			x		…
3	x	x		x	x		x	…
4	x		x	x				…
5	x	x	x	x	x	x	x	…
6		x		x				…
7	x		x				x	…
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	

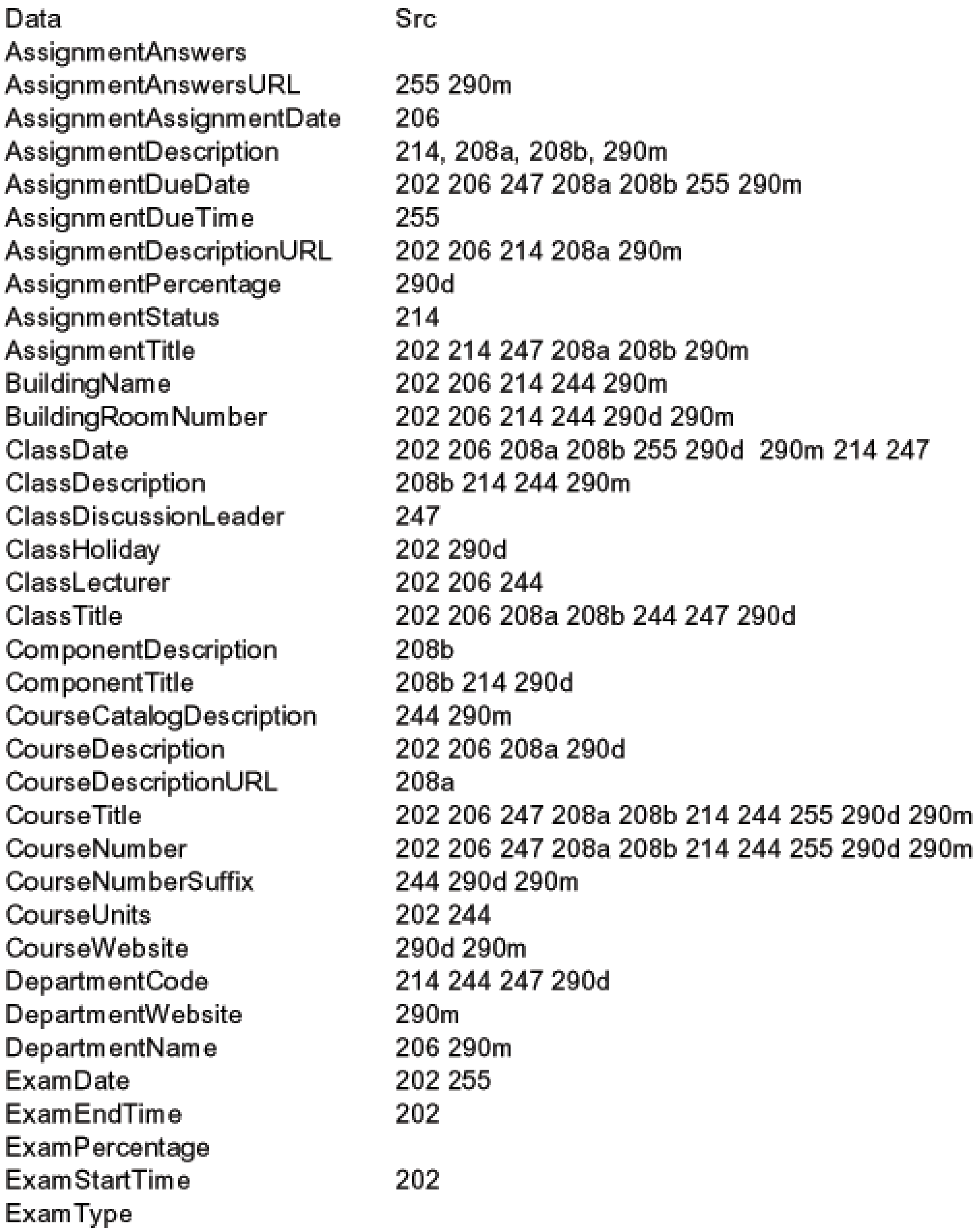
- We can begin our consolidation with the candidate components from any of the information sources, but we recommend using the one you believe is the most authoritative or that yielded the most components
- The goal is to combine components that are synonyms (different names for the same meaning) and to distinguish any homonyms (same names for different meanings)

- It is desirable for a set of components to enable one and only one way to describe something because duplication or redundancy implies choices that could lead to inconsistent models and non-interoperable schemas
- Some guidelines or questions to help minimize duplicate components, especially when they are proposed from documents / sources / applications from different authors or organizations:
    - Are the differences between the proposed components substantive or stylistic? (writing or encoding style)
    - Are the differences "real" but "unimportant" to users or applications? (spurious precision)

79. Seek Semantic Clarity and Precision

- It seems obvious that we need "good names" and "good definitions" for the components we identify and design but what does that mean?
- In an article titled "What's in a Name?" (http://www.vertaasis.com/articles/whats_in_a_name.htm) Farish recommends three "levels" of models (or names) that line up nicely with our three stages of analysis, design, and encoding
    - *Business names* – a format that lets the requirement or semantics be easily readable and verifiable by a business person (not a modeling or XML expert). This should use familiar words and be completely technology-independent
    - *Logical names* – a format optimized for the expression of the design or model; essential that they are expressive enough to reflect the relationships between model components. Logical names might follow precise rules to ensure that they can be reliably stored and located in a data dictionary; ("qualified names" specialize general terms to convey the context of use)
    - *Physical names* – the format required by the implementation technology for the model
- "The expense of resolving ambiguous business terms over and over on a daily basis pales in comparison with the expense of *NOT* realizing that there is an ambiguity in the term"
- Farish's example: "Shipping Container"

80. Consolidation Example

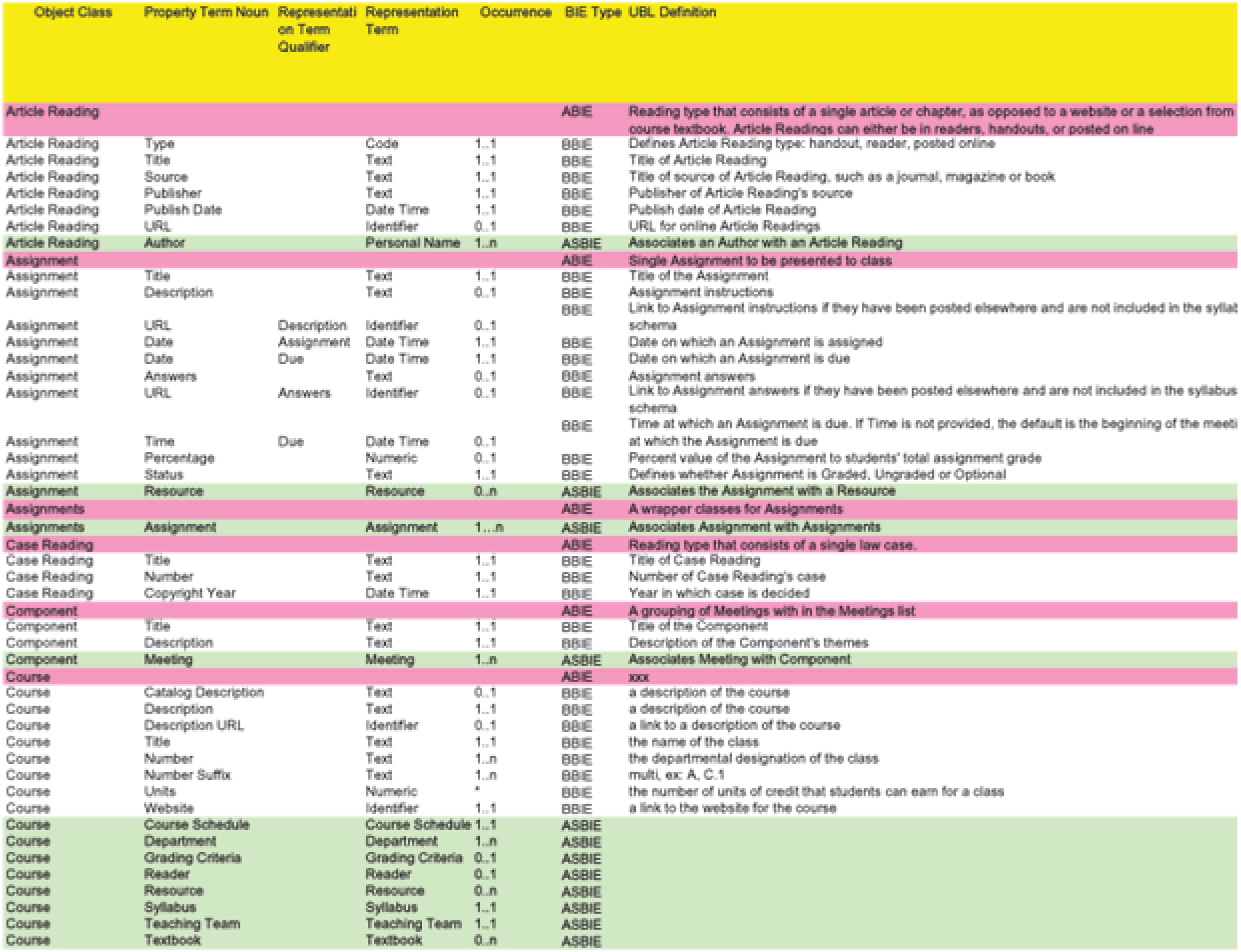


Data	Src
AssignmentAnswers	
AssignmentAnswersURL	255 290m
AssignmentAssignmentDate	206
AssignmentDescription	214, 208a, 208b, 290m
AssignmentDueDate	202 206 247 208a 208b 255 290m
AssignmentDueTime	255
AssignmentDescriptionURL	202 206 214 208a 290m
AssignmentPercentage	290d
AssignmentStatus	214
AssignmentTitle	202 214 247 208a 208b 290m
BuildingName	202 206 214 244 290m
BuildingRoomNumber	202 206 214 244 290d 290m
ClassDate	202 206 208a 208b 255 290d 290m 214 247
ClassDescription	208b 214 244 290m
ClassDiscussionLeader	247
ClassHoliday	202 290d
ClassLecturer	202 206 244
ClassTitle	202 206 208a 208b 244 247 290d
ComponentDescription	208b
ComponentTitle	208b 214 290d
CourseCatalogDescription	244 290m
CourseDescription	202 206 208a 290d
CourseDescriptionURL	208a
CourseTitle	202 206 247 208a 208b 214 244 255 290d 290m
CourseNumber	202 206 247 208a 208b 214 244 255 290d 290m
CourseNumberSuffix	244 290d 290m
CourseUnits	202 244
CourseWebsite	290d 290m
DepartmentCode	214 244 247 290d
DepartmentWebsite	290m
DepartmentName	206 290m
ExamDate	202 255
ExamEndTime	202
ExamPercentage	
ExamStartTime	202
ExamType	

81. Normalization

- *Normalization*– Applying techniques for reducing redundancy and increasing integrity in information models
    - The consolidated list of unique candidate components is equivalent to 1NF in relational theory
    - Data normalization techniques can be applied to further refine the set of candidate components (if used sensibly)
    - Components that are functionally independent of each other are separated and their bi-directional relationships are recorded

82. Nornalization Example



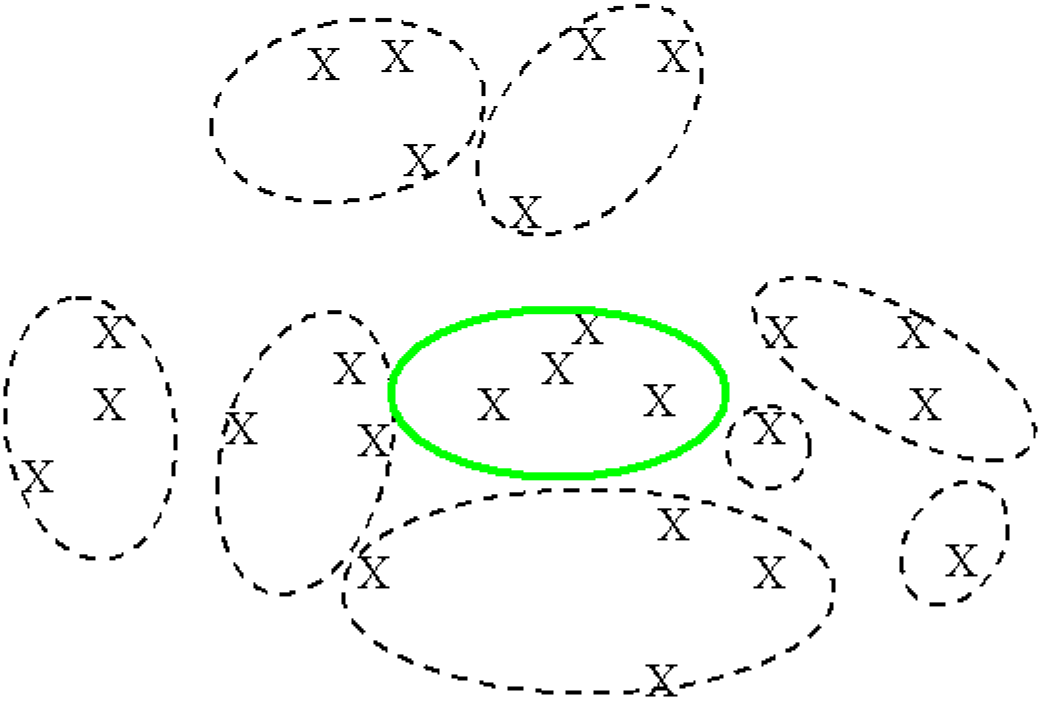
Object Class	Property Term Noun	Representation Term Qualifier	Representation Term	Occurence	BIE Type	UBL Definition
Article Reading					ABIE	Reading type that consists of a single article or chapter, as opposed to a website or a selection from course textbook. Article Readings can either be in readers, handouts, or posted on line
Article Reading	Type		Code	1..1	BBIE	Defines Article Reading type: handout, reader, posted online
Article Reading	Title		Text	1..1	BBIE	Title of Article Reading
Article Reading	Source		Text	1..1	BBIE	Title of source of Article Reading, such as a journal, magazine or book
Article Reading	Publisher		Text	1..1	BBIE	Publisher of Article Reading's source
Article Reading	Publish Date		Date Time	1..1	BBIE	Publish date of Article Reading
Article Reading	URL		Identifier	0..1	BBIE	URL for online Article Readings
Article Reading	Author		Personal Name	1..n	ASBIE	Associates an Author with an Article Reading
Assignment					ABIE	Single Assignment to be presented to class
Assignment	Title		Text	1..1	BBIE	Title of the Assignment
Assignment	Description		Text	0..1	BBIE	Assignment instructions
Assignment	URL	Description	Identifier	0..1	BBIE	Link to Assignment instructions if they have been posted elsewhere and are not included in the syllat schema
Assignment	Date	Assignment	Date Time	1..1	BBIE	Date on which an Assignment is assigned
Assignment	Date	Due	Date Time	1..1	BBIE	Date on which an Assignment is due
Assignment	Answers		Text	0..1	BBIE	Assignment answers
Assignment	URL	Answers	Identifier	0..1	BBIE	Link to Assignment answers if they have been posted elsewhere and are not included in the syllabus schema
Assignment	Time	Due	Date Time	0..1	BBIE	Time at which an Assignment is due. If Time is not provided, the default is the beginning of the meeti at which the Assignment is due
Assignment	Percentage		Numeric	0..1	BBIE	Percent value of the Assignment to students' total assignment grade
Assignment	Status		Text	1..1	BBIE	Defines whether Assignment is Graded, Ungraded or Optional
Assignment	Resource		Resource	0..n	ASBIE	Associates the Assignment with a Resource
Assignments					ABIE	A wrapper classes for Assignments
Assignments	Assignment		Assignment	1...n	ASBIE	Associates Assignment with Assignments
Case Reading					ABIE	Reading type that consists of a single law case.
Case Reading	Title		Text	1..1	BBIE	Title of Case Reading
Case Reading	Number		Text	1..1	BBIE	Number of Case Reading's case
Case Reading	Copyright Year		Date Time	1..1	BBIE	Year in which case is decided
Component					ABIE	A grouping of Meetings with in the Meetings list
Component	Title		Text	1..1	BBIE	Title of the Component
Component	Description		Text	1..1	BBIE	Description of the Component's themes
Component	Meeting		Meeting	1..n	ASBIE	Associates Meeting with Component
Course					ABIE	xxx
Course	Catalog Description		Text	0..1	BBIE	a description of the course
Course	Description		Text	1..1	BBIE	a description of the course
Course	Description URL		Identifier	0..1	BBIE	a link to a description of the course
Course	Title		Text	1..1	BBIE	the name of the class
Course	Number		Text	1..n	BBIE	the departmental designation of the class
Course	Number Suffix		Text	1..n	BBIE	multi, ex: A, C.1
Course	Units		Numeric	*	BBIE	the number of units of credit that students can earn for a class
Course	Website		Identifier	1..1	BBIE	a link to the website for the course
Course	Course Schedule		Course Schedule	1..1	ASBIE	
Course	Department		Department	1..n	ASBIE	
Course	Grading Criteria		Grading Criteria	0..1	ASBIE	
Course	Reader		Reader	0..1	ASBIE	
Course	Resource		Resource	0..n	ASBIE	
Course	Syllabus		Syllabus	1..1	ASBIE	
Course	Teaching Team		Teaching Team	1..1	ASBIE	
Course	Textbook		Textbook	0..n	ASBIE	

83. Informal Normalization with "Core and Contexts"

- When we use the "Address" component in the context of domestic mail it will not contain the same components as when we use "Address" in the context of international shipments
- When we customize a "context-free" or "reusable" or "base" component for a specific context, some information may need to be added, and some information may need to be deleted
- Can we organize our components to package the information needed for different contexts in aggregates so they can be reused efficiently?

84. Core and Contexts [1]

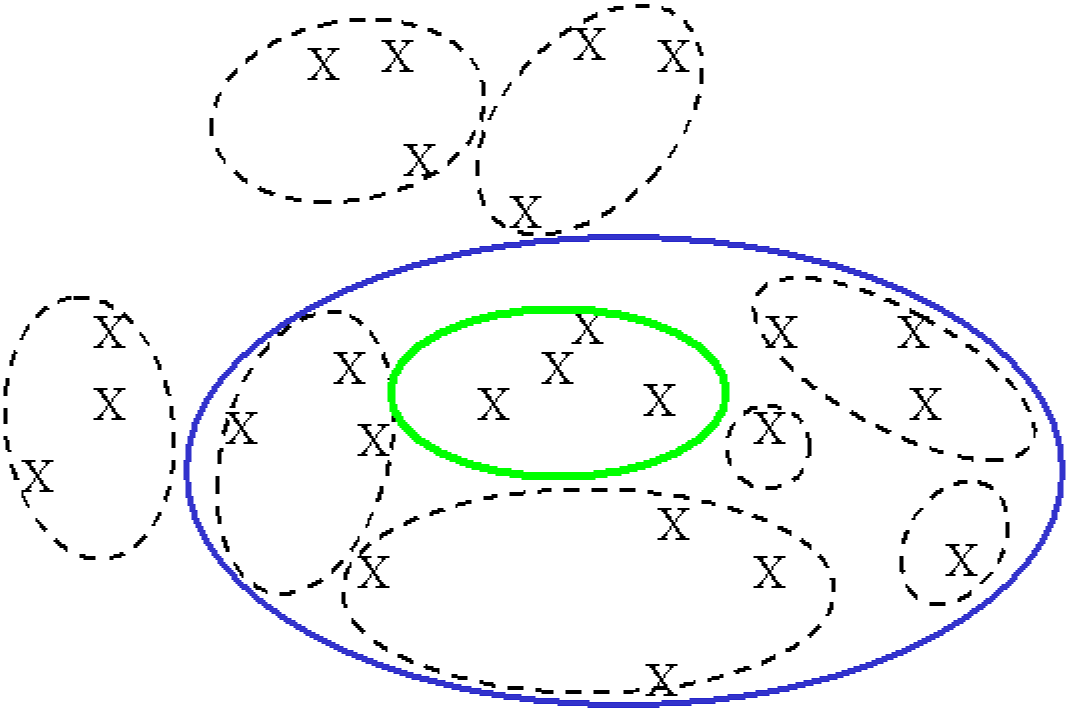
- We create a set of components, each of which contains the information needed for a more specialized context (but not the core itself)



Component Modeling Using Contexts

85. Core and Contexts [2]

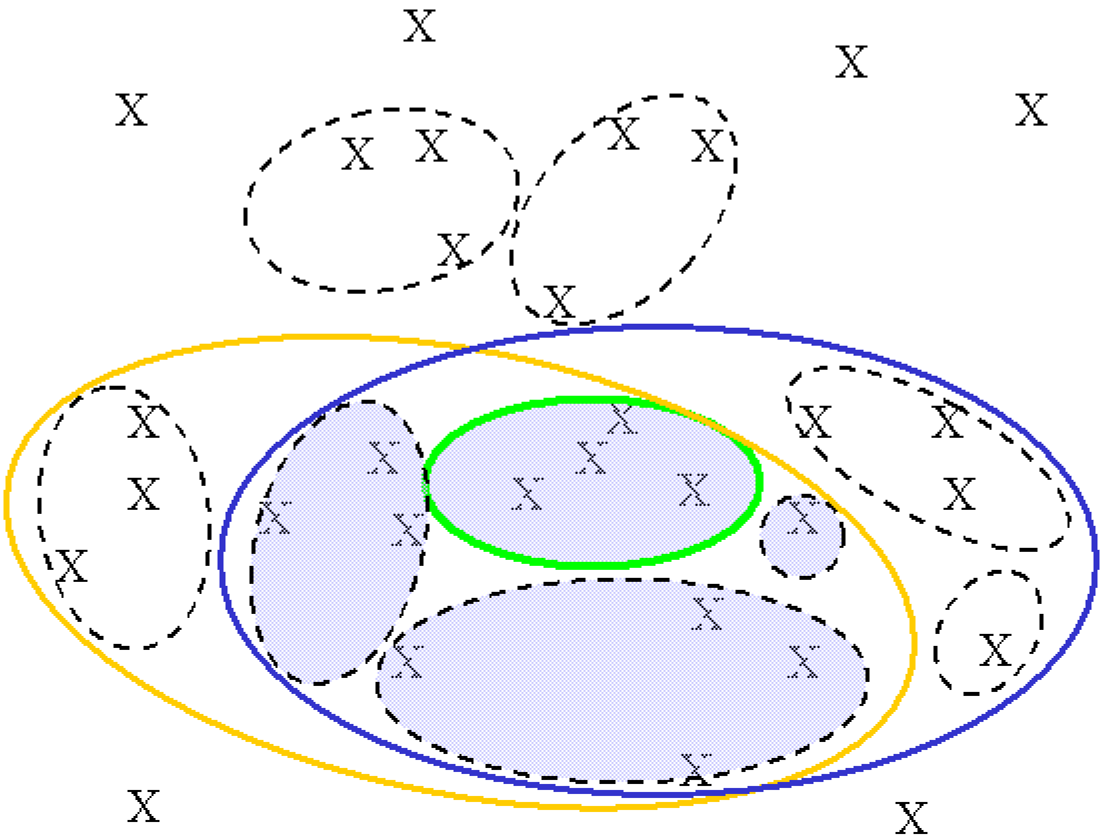
- These additional components can be assembled with the core, alone or in combinations, to create the customized model that we need in any context



Component Modeling in “Contexts” Styl
To Represent Specialized Components

86. Core and Contexts [3]

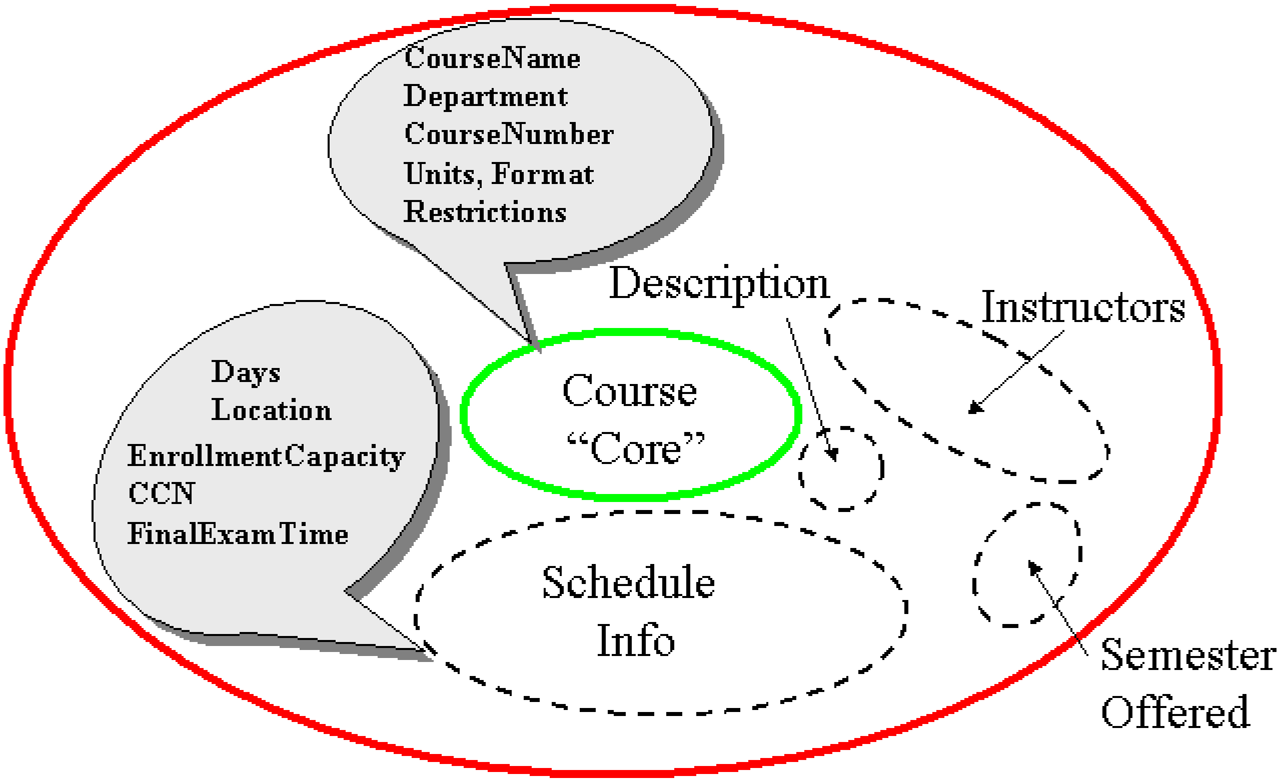
- This approach captures both the core semantic overlap as well as a lot more without bringing in any unnecessary information



Components Modeled in “Contexts” Styl
Can Capture Common Semantics

87. Modeling with Contexts in the Course "ecosystem" (Garvey et al 2003) [1]

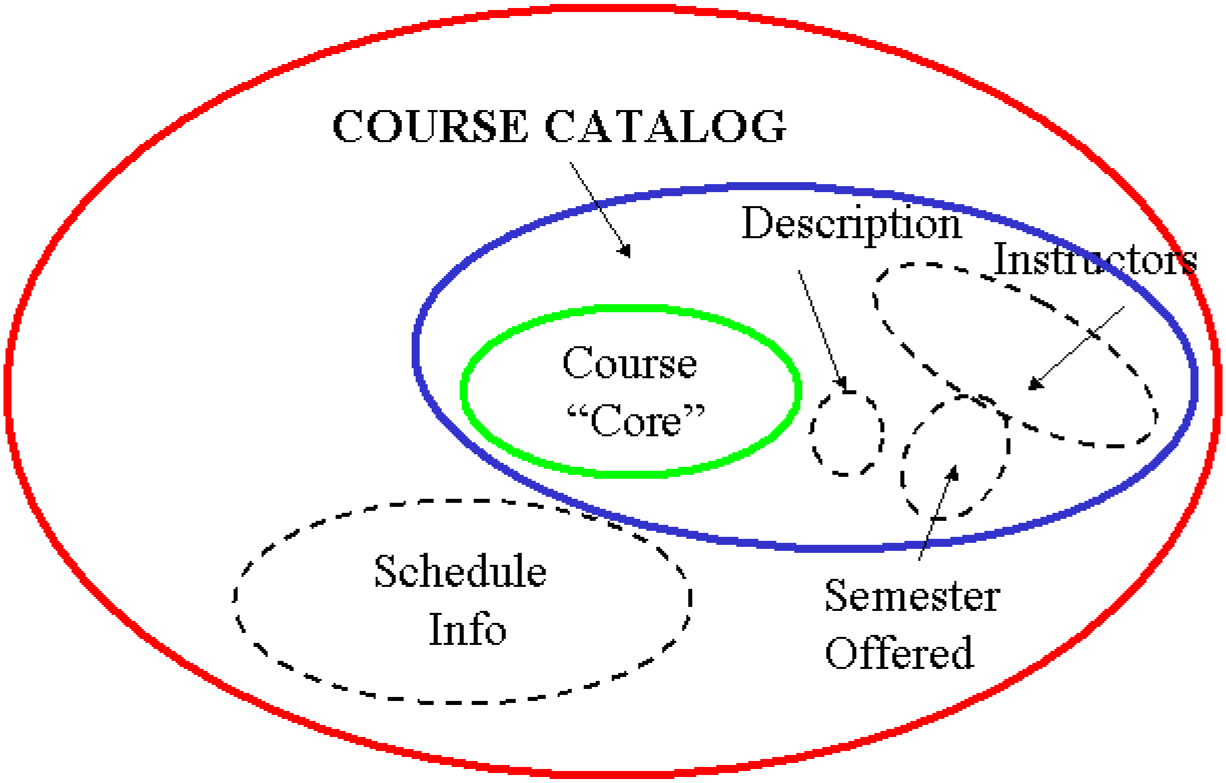
- Simplified depiction of "core" course and its contexts



Partial Model of a “Course” and its Contexts

88. Modeling with Contexts in the Course "ecosystem" (Garvey et al 2003) [2]

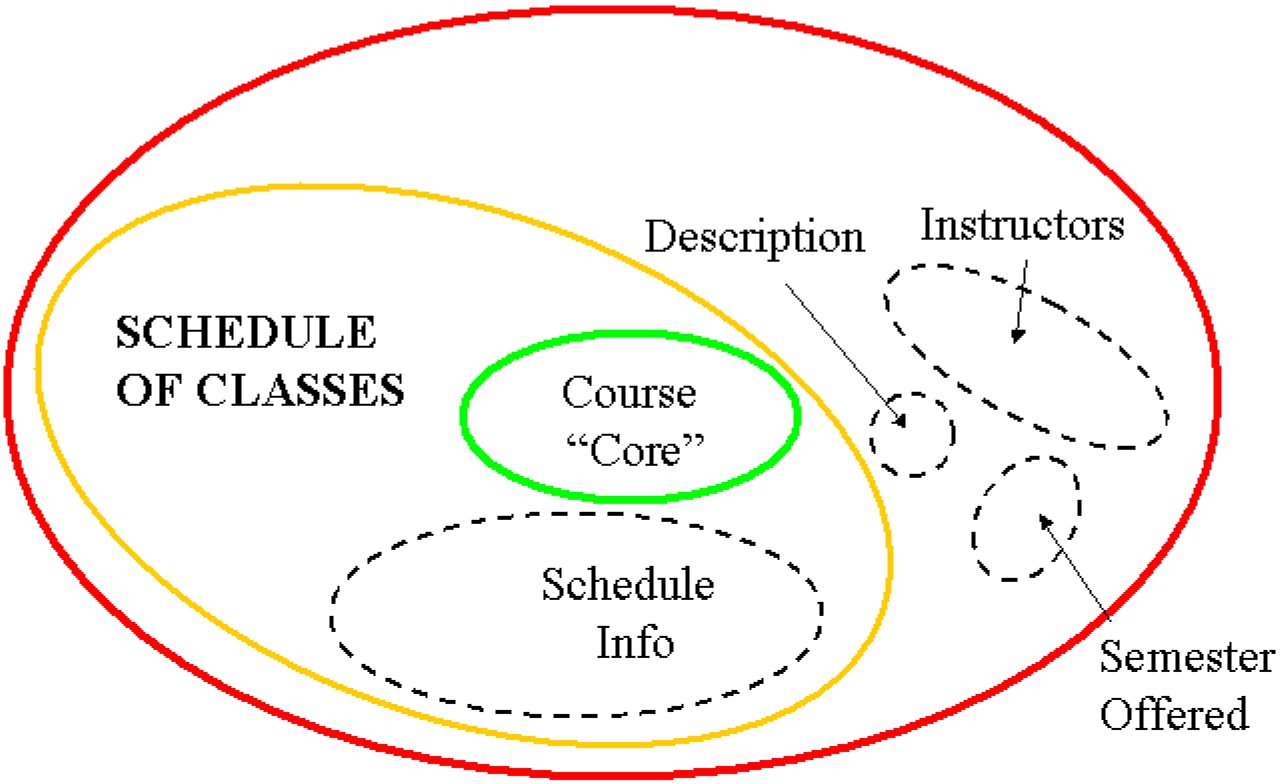
- Course catalog assembled from core + catalog context



Component Assembly for
"COURSE CATALOG"

89. Modeling with Contexts in the Course "ecosystem" (Garvey et al 2003) [3]

- Schedule of classes assembled from core + schedule context



Component Assembly for “SCHEDULE OF CLASSES”

90. Document Assembly

- Document assembly is the process of creating a model of a document type – hierarchical and nested – by drawing on the "pool" or library of content and structural components

- Assembly involves designing (or selecting a pattern for) the top level structure as an entry point and then navigating through the relationships in the conceptual model collecting the components in the order that best satisfies your requirements

- Assembly order can differ whenever there is a bi-directional relationship between components – whenever two components are functionally independent, an assembly order chooses one of the relationships to enforce an interpretation on the assembled document

- Most assembly pathways have a dominant hierarchical character because most documents have a strong structural hierarchy or clear divisions based on standard presentational patterns

- The navigational route taken through the conceptual model is determined by the processing requirements or context of the document's use

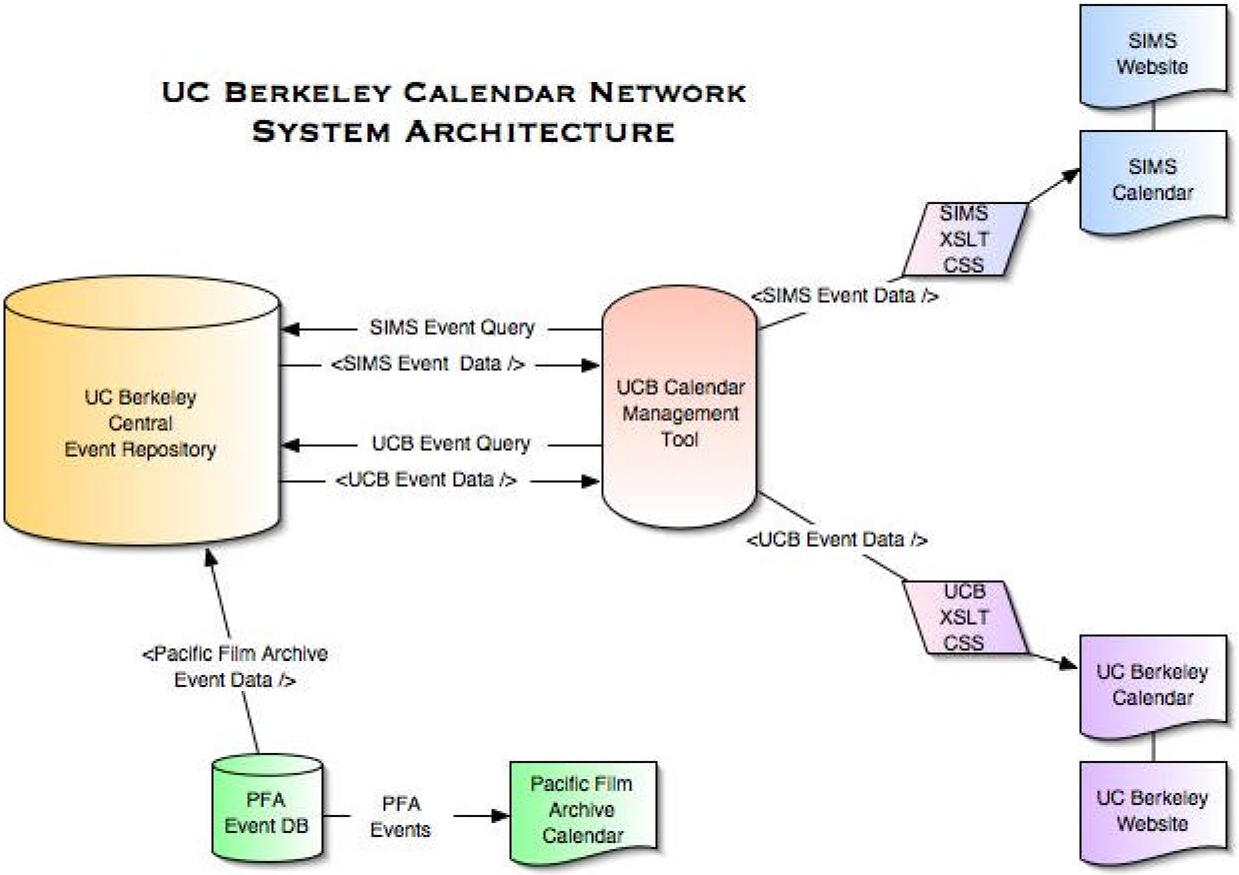
91. Example 1:The Berkeley Event Calendar Network

- Bloodworth & Glushko, *Model-driven application design for a campus calendar network*, XML 2004

- Document engineering case study whose "snapshots" illustrate the analysis, modeling, and schema encoding approach

- The problem - scores of calendars on berkeley.edu with overlapping coverage and audiences but incompatible data models
- No automated reuse of information; you need to submit events to multiple calendars or copy events from them
- Each calendar has a different event submission form and a different model of an event
- Typical of problems that occur in every large organization with timesheets, expense reimbursement, registration, etc. and also representative of B2B interactions with incompatible catalogs, orders, etc.

92. Event Calendar Network: Conceptual Architecture

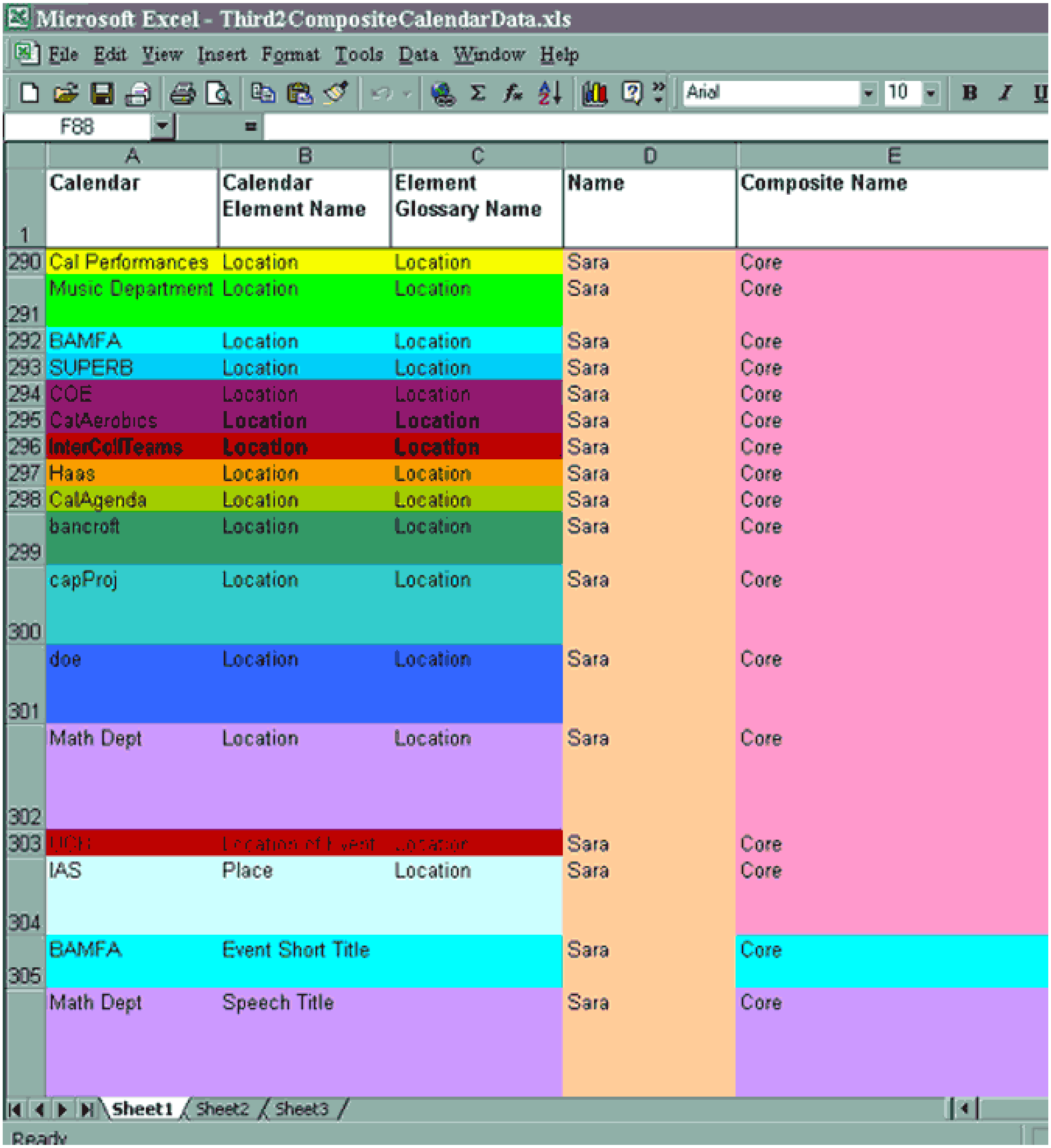


93. Event Calendars: Harvesting and Consolidating Components

- Don't be fooled by collection of presentation info in Business School event creation form accessed from here
- Synonyms:
  - Start Date
  - Commencement
- Homonyms:
  - Contact (person submitting an event)

- ○ Contact (person to contact about an event)
- ○ Category / Type (disjoint domains: events, attendees)

94. Event Calendars Consolidated Table of Candidate Components

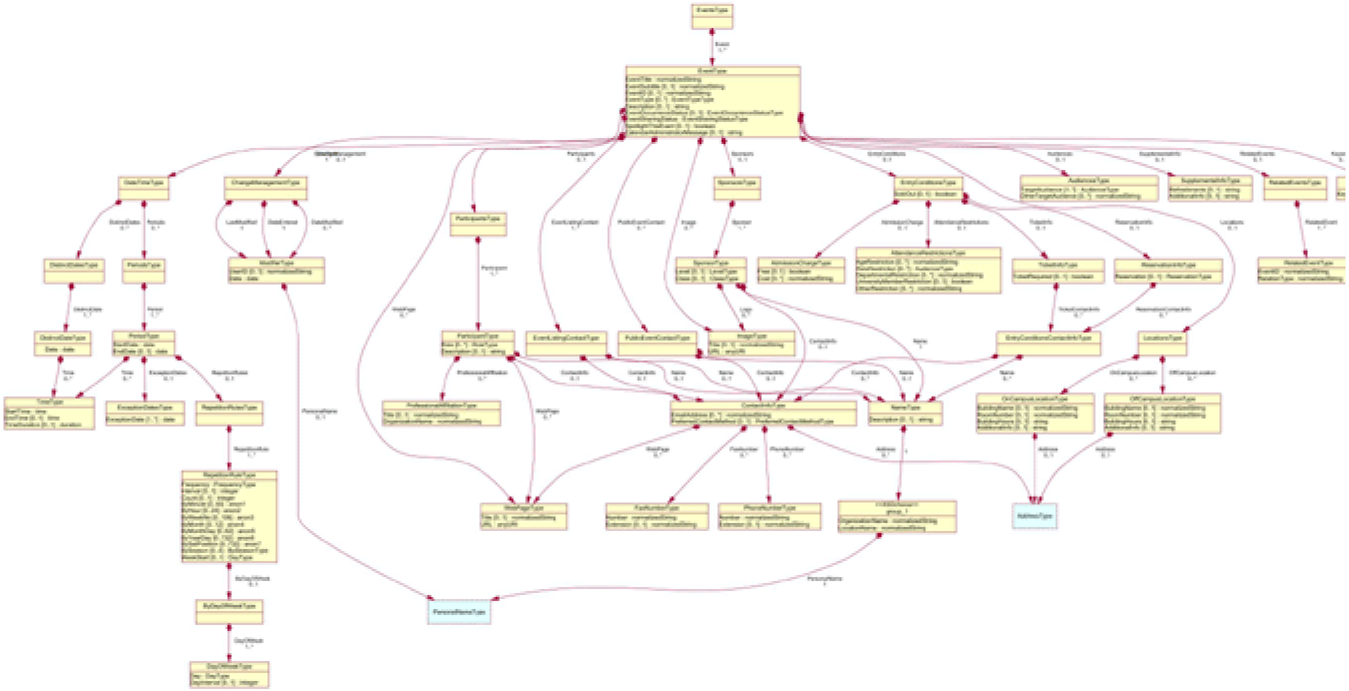

Microsoft Excel - Third2CompositeCalendarData.xls
File Edit View Insert Format Tools Data Window Help
Arial
10
B I U
F88
=
Sheet1
Sheet2
Sheet3
Ready

	A	B	C	D	E
1	Calendar	Calendar Element Name	Element Glossary Name	Name	Composite Name
290	Cal Performances	Location	Location	Sara	Core
291	Music Department	Location	Location	Sara	Core
292	BAMFA	Location	Location	Sara	Core
293	SUPERB	Location	Location	Sara	Core
294	COE	Location	Location	Sara	Core
295	CalAerobics	Location	Location	Sara	Core
296	InterCollTeams	Location	Location	Sara	Core
297	Haas	Location	Location	Sara	Core
298	CalAgenda	Location	Location	Sara	Core
299	bancroft	Location	Location	Sara	Core
300	capProj	Location	Location	Sara	Core
301	doe	Location	Location	Sara	Core
302	Math Dept	Location	Location	Sara	Core
303	UCH	Location of Event	Location	Sara	Core
304	IAS	Place	Location	Sara	Core
305	BAMFA	Event Short Title		Sara	Core
	Math Dept	Speech Title		Sara	Core

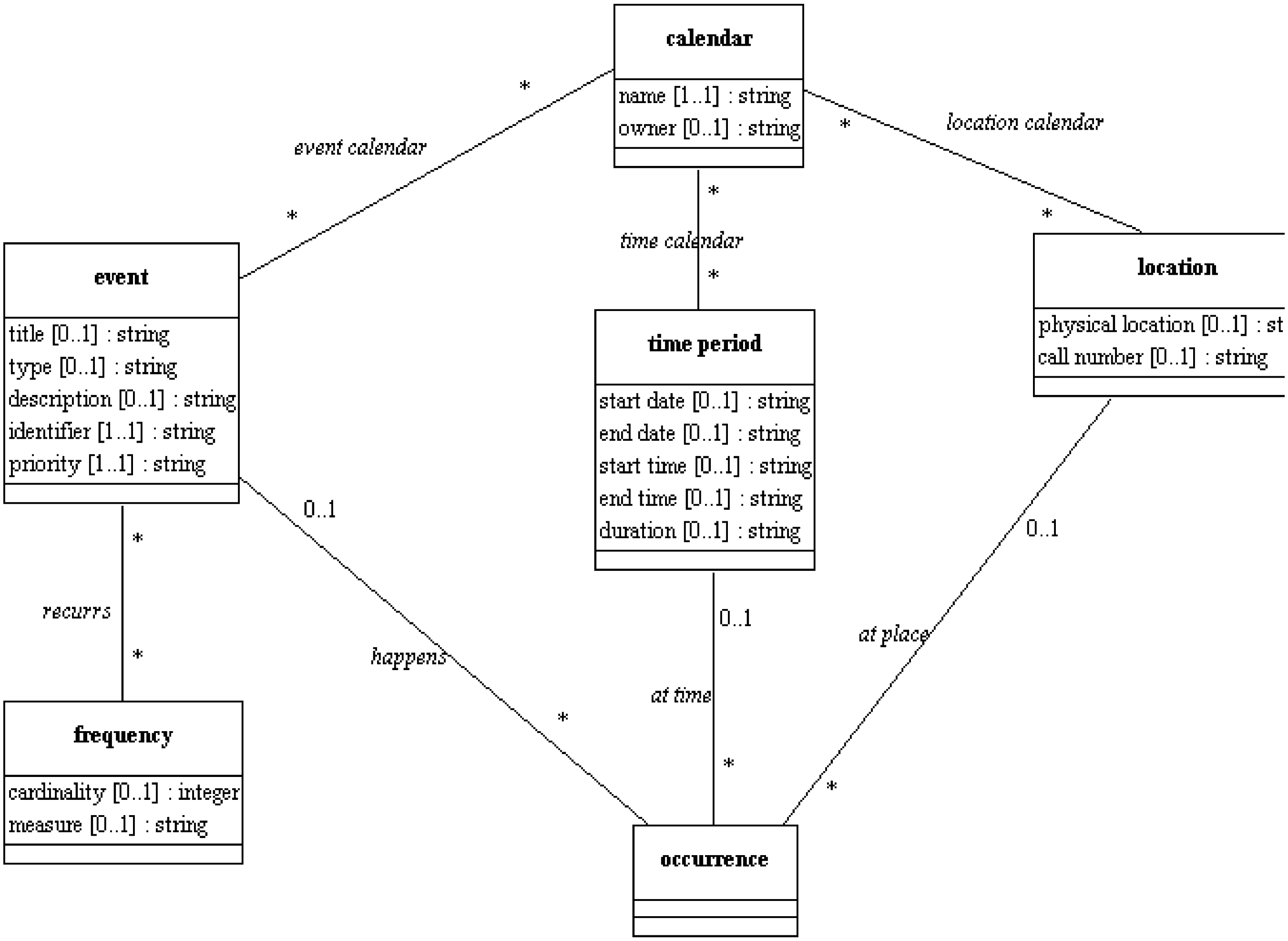
95. Event Calendars: The Conceptual Model

- When we've analyzed all of the candidate components for dependencies, we've created a conceptual model for event calendars

- From this model we can assemble any of a set of related document types for different varieties of event calendars

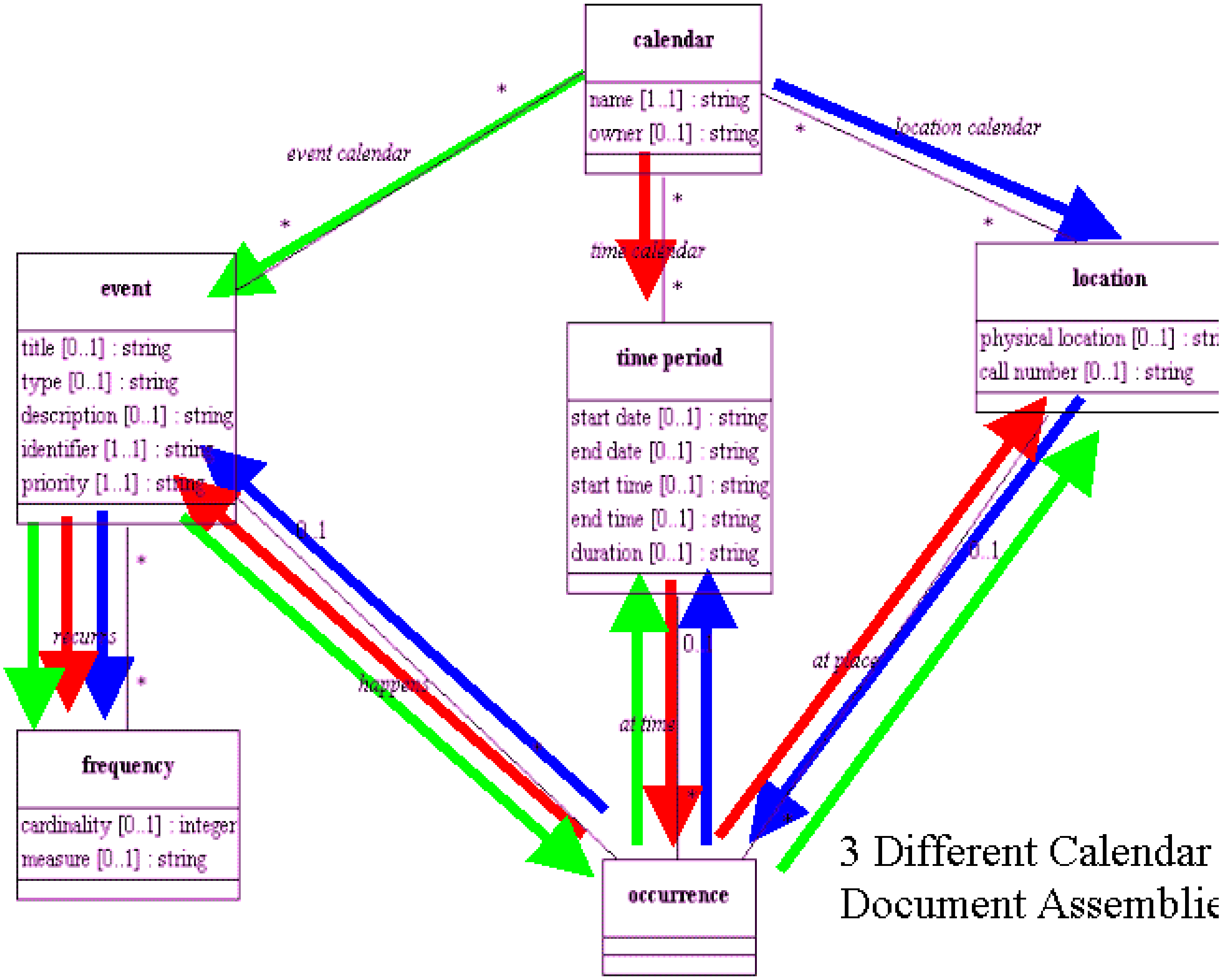


96. Event Calendars: The Conceptual Model (Simplified for the Tutorial)



97. Event Calendars: Assembly

- Different types of calendars can be assembled from the same conceptual model by starting at different entry points and following the associations



98. Implementing Models in Document Engineering

- Many of the important patterns for Document Engineering are used when encoding assembled document models as XML schemas
- This is a two-stage process: encoding assembled document models as physical ones, and then applying formatting or style transformations to create instances with desired properties
    - A rigorous format for the document model that specifies how to name components and describe their relationships can support the automated generation of the physical ones
- When instances implemented in different technologies are generated or re-generated from models, they can more readily interoperate because of their common conceptual components

99. Schema Encoding Rules Might Cover...

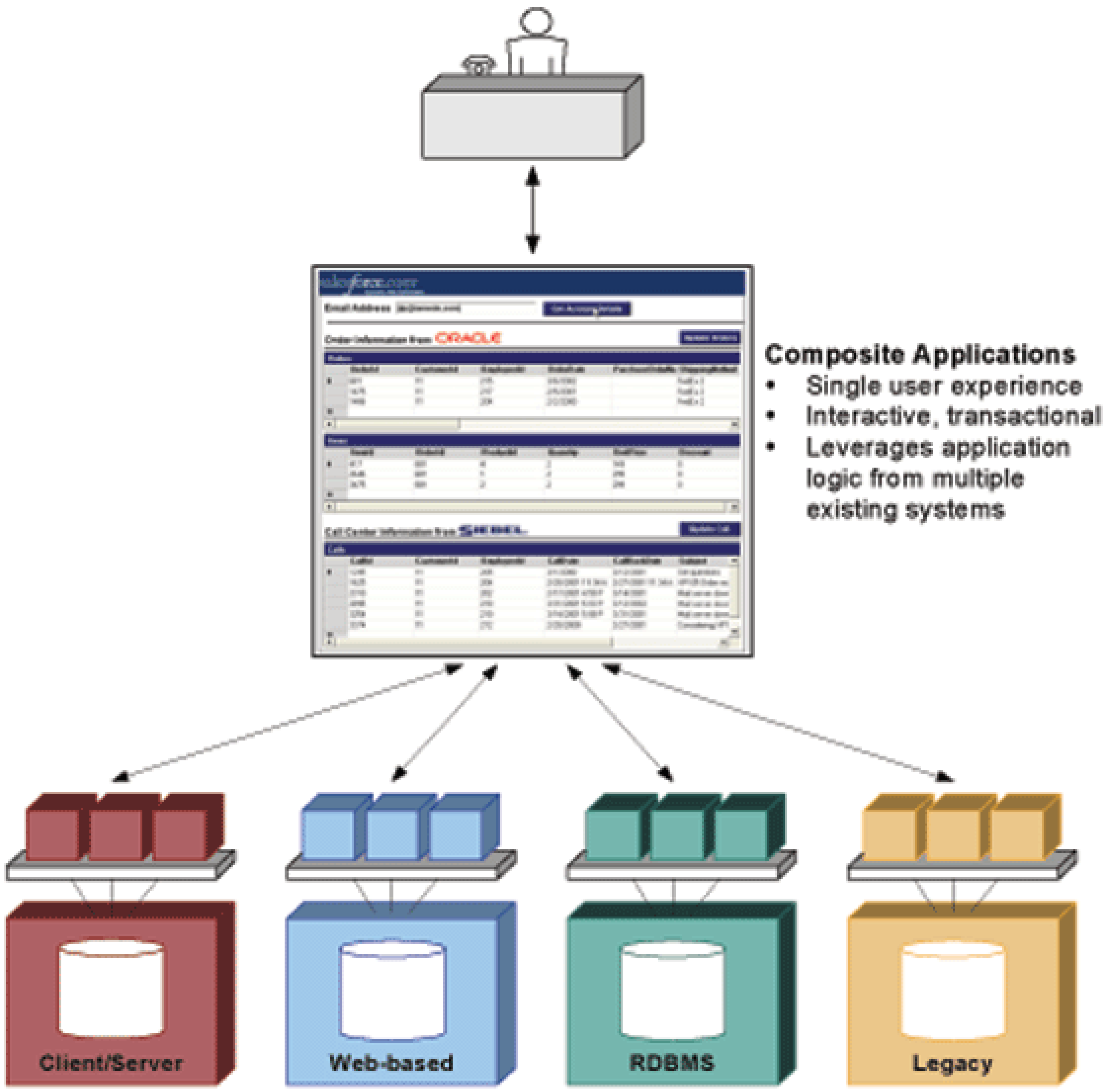
- The choice of normative schema language
- Naming and construction of elements, attributes, and types

- Modularity, namespaces, and versioning
- Embedded schema documentation
- Handling codes/enumerated lists

100. An Event Calendar Instance

```
<Calendar>
<EventOccurrenceLocation>
<FrequencyNumeric>1</FrequencyNumeric>
<FrequencyMeasure unitCode="days">1</FrequencyMeasure>
<Event>
  <Identifier>Tutorial #3</Identifier>
  <Title>Document Engineering: Designing Documents for Transactions and Web Services</Ti
  <Type>Tutorial</Type>
  <Description>Bob Glushko's Tutorial at OASIS Symposium.</Description>
  <Priority>High</Priority>
  <Participation>
    <TicketRequiredIndicator>true</TicketRequiredIndicator>
    <ChargeStructure>
      <Fee>included in registration</Fee>
      <Description>Depends on the flavor of registration</Description>
    </ChargeStructure>
  </Participation>
  <Speaker>
    <Name>Robert J. Glushko</Name>
    <Affiliation>University of California, Berkeley</Affiliation>
  </Speaker>
</Event>
<Location>
  <PhysicalLocation>New Orleans Marriott</PhysicalLocation>
  <Website>
    <URL>http://www.oasis-open.org/events/symposium_2005/</URL>
  </Website>
</Location>
<Occurrence>
  <StartDate>2005-04-24</StartDate>
  <StartTime>13:30:00</StartTime>
  <EndDate>2005-04-24</EndDate>
  <EndTime>17:00</EndTime>
</Occurrence>
</EventOccurrenceLocation>
</Calendar>
```

101. Example 2: Composite Travel Service


salesforce.com
Order Information from ORACLE
Call Center Information from SIEBEL
Composite Applications
• Single user experience
• Interactive, transactional
• Leverages application logic from multiple existing systems
Client/Server
Web-based
RDBMS
Legacy

- Document Engineering is essential in the design of composite applications and services
  - What patterns of service combination are required to meet our business objectives?
  - Are there constraints on "composed services" based on extent of semantic overlap?
  - What reusable information components do we need to make services interoperate?

102. Composite Travel Reservation Service Scenario [1]

1. CUSTOMER registers with TRAVEL SERVICE by providing information about name, credit card, and some travel preferences and constraints.
   - The credit card information includes a card number, a billing address, and an expiration date.
   - Travel preferences should handle dietary preferences or restrictions adequate for airline meals and restaurants, quality rating thresholds for air (class of service), hotel (rating) and restaurants (rating), and some notion of convenience based on the distance between the hotel and restaurant.

103. Composite Travel Reservation Service Scenario [2]

2. CUSTOMER interacts with TRAVEL SERVICE to request a flight and a hotel, providing information about departure and arrival location and the desired departure time or arrival time for the outbound and return trip.

- The CUSTOMER also requests a restaurant reservation that meets his dietary preferences at a restaurant that is convenient to the hotel at a time between two and three hours after the flight arrives
- Assume round trip travel only, no airline preference, and no cost restrictions

104. Composite Travel Reservation Service Scenario [3]

3. TRAVEL SERVICE composes the appropriate message to the relevant service providers.

- At some appropriate point the TRAVEL SERVICE proposes a list of feasible flight and hotel combinations to the CUSTOMER.

4. CUSTOMER selects one of the flight and hotel combinations.

105. Composite Travel Reservation Service Scenario [4]

5. TRAVEL SERVICE composes the appropriate message to the relevant service providers.

- At some appropriate point the TRAVEL SERVICE proposes a list of feasible restaurants to the CUSTOMER that meet the distance constraints between the selected hotel and the candidate restaurants.

6. CUSTOMER selects a restaurant.

106. Composite Travel Reservation Service Scenario [5]

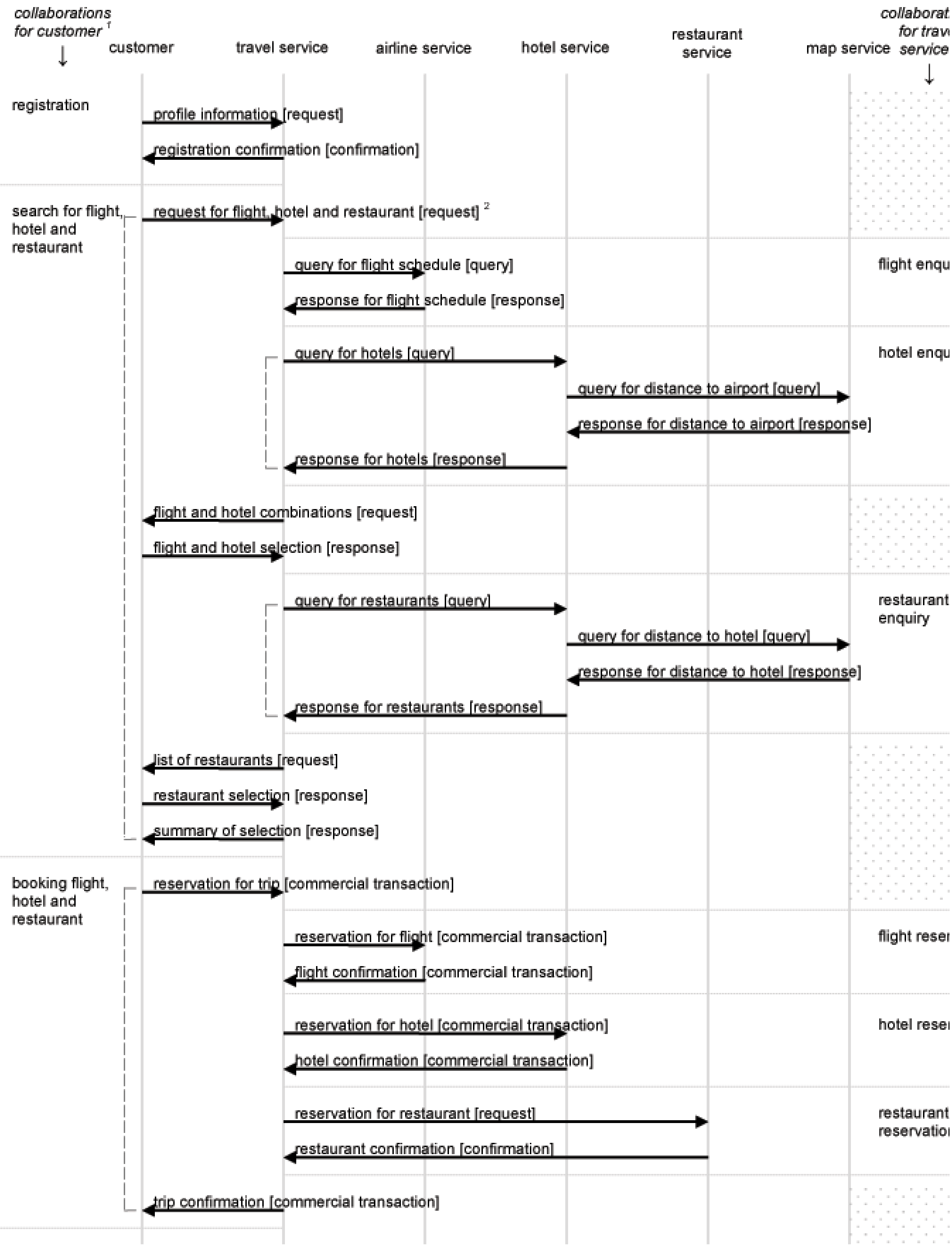
7. TRAVEL SERVICE composes the appropriate message to the relevant service providers to confirm the CUSTOMER's reservations

- Provides the CUSTOMER's credit card information to the services that need it.

8. At the appropriate time the TRAVEL SERVICE sends a single message to the CUSTOMER confirming all the parts of the composite reservation.

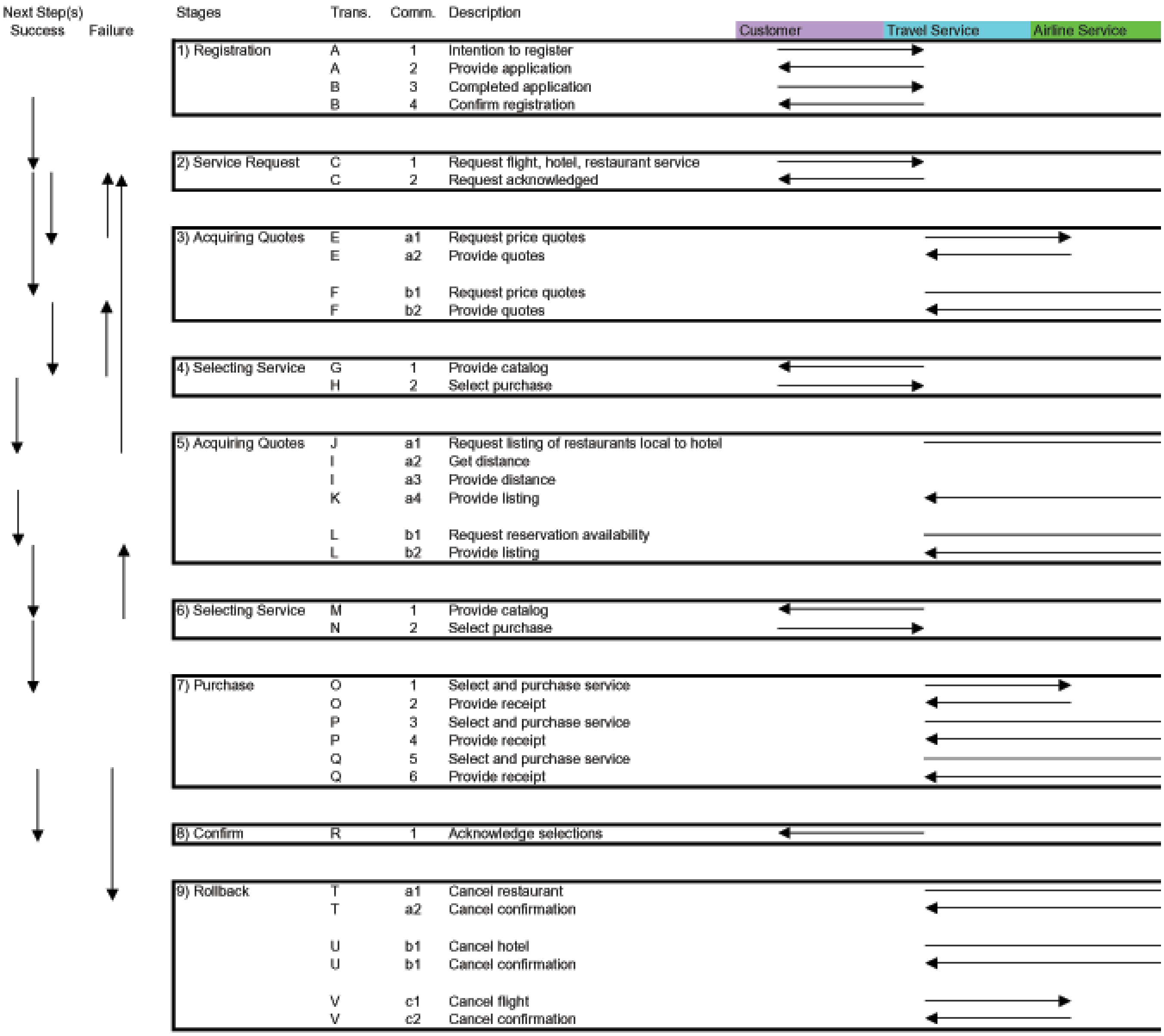
107. Collaborations and Transaction Diagrams [1]

Sequence Diagram for Composite Travel Service


collaborations for customer [1]
customer
travel service
airline service
hotel service
restaurant service
map service
collaborat for trav service
registration
profile information [request]
registration confirmation [confirmation]
search for flight, hotel and restaurant
request for flight, hotel and restaurant [request] [2]
query for flight schedule [query]
response for flight schedule [response]
flight enqu
query for hotels [query]
hotel enqu
query for distance to airport [query]
response for distance to airport [response]
response for hotels [response]
flight and hotel combinations [request]
flight and hotel selection [response]
query for restaurants [query]
restaurant enquiry
query for distance to hotel [query]
response for distance to hotel [response]
response for restaurants [response]
list of restaurants [request]
restaurant selection [response]
summary of selection [response]
booking flight, hotel and restaurant
reservation for trip [commercial transaction]
reservation for flight [commercial transaction]
flight reser
flight confirmation [commercial transaction]
reservation for hotel [commercial transaction]
hotel reser
hotel confirmation [commercial transaction]
reservation for restaurant [request]
restaurant reservatio
restaurant confirmation [confirmation]
trip confirmation [commercial transaction]

- Named collaborations and transactions
- Arrows on transactions to indicate direction of document exchange
- Transaction pattern type as annotation in brackets.

108. Collaborations and Transaction Diagrams [2]


Next Step(s)
Success
Failure
Customer
Travel Service
Airline Service

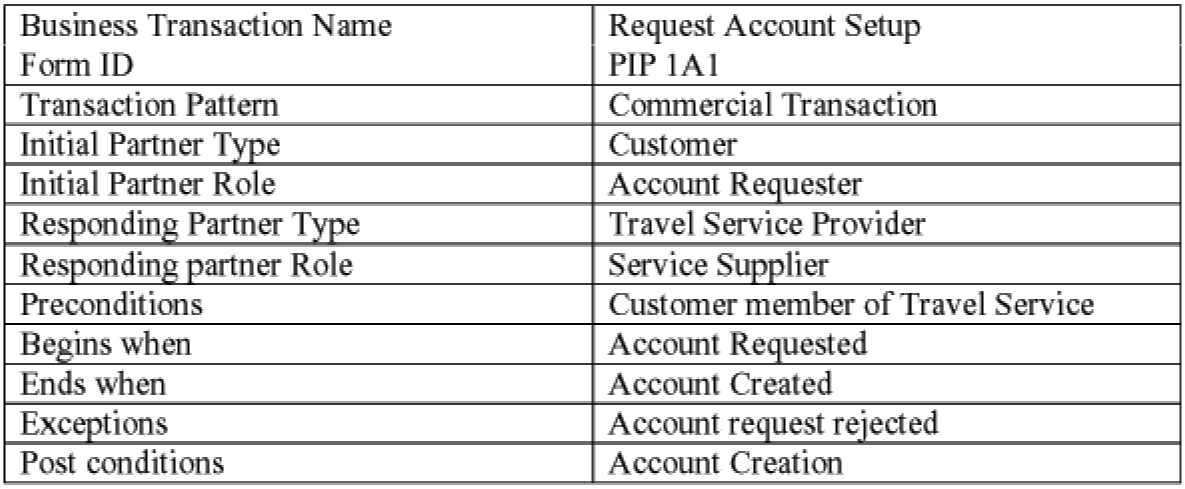
Stages	Trans.	Comm.	Description
1) Registration	A	1	Intention to register
	A	2	Provide application
	B	3	Completed application
	B	4	Confirm registration
2) Service Request	C	1	Request flight, hotel, restaurant service
	C	2	Request acknowledged
3) Acquiring Quotes	E	a1	Request price quotes
	E	a2	Provide quotes
	F	b1	Request price quotes
	F	b2	Provide quotes
4) Selecting Service	G	1	Provide catalog
	H	2	Select purchase
5) Acquiring Quotes	J	a1	Request listing of restaurants local to hotel
	I	a2	Get distance
	I	a3	Provide distance
	K	a4	Provide listing
	L	b1	Request reservation availability
	L	b2	Provide listing
6) Selecting Service	M	1	Provide catalog
	N	2	Select purchase
7) Purchase	O	1	Select and purchase service
	O	2	Provide receipt
	P	3	Select and purchase service
	P	4	Provide receipt
	Q	5	Select and purchase service
	Q	6	Provide receipt
8) Confirm	R	1	Acknowledge selections
9) Rollback	T	a1	Cancel restaurant
	T	a2	Cancel confirmation
	U	b1	Cancel hotel
	U	b1	Cancel confirmation
	V	c1	Cancel flight
	V	c2	Cancel confirmation

- Detailed numbering scheme to make it very useful as index to later work products
- Color coding of actors (columns)
- Transition logic

109. The Models

- Some teams missed some key requirement or instruction
- Other teams proposed redundant transactions with individual airlines, hotels, and restaurants
- Some teams went beyond the stated requirements and proposed new ones along with transactions that could handle them (most common example was "compute distance from airport to hotel")
- Some teams proposed inefficient ways of meeting the requirements; most common examples involved the transactions with the mapping service, where GPS was used in overly clever ways

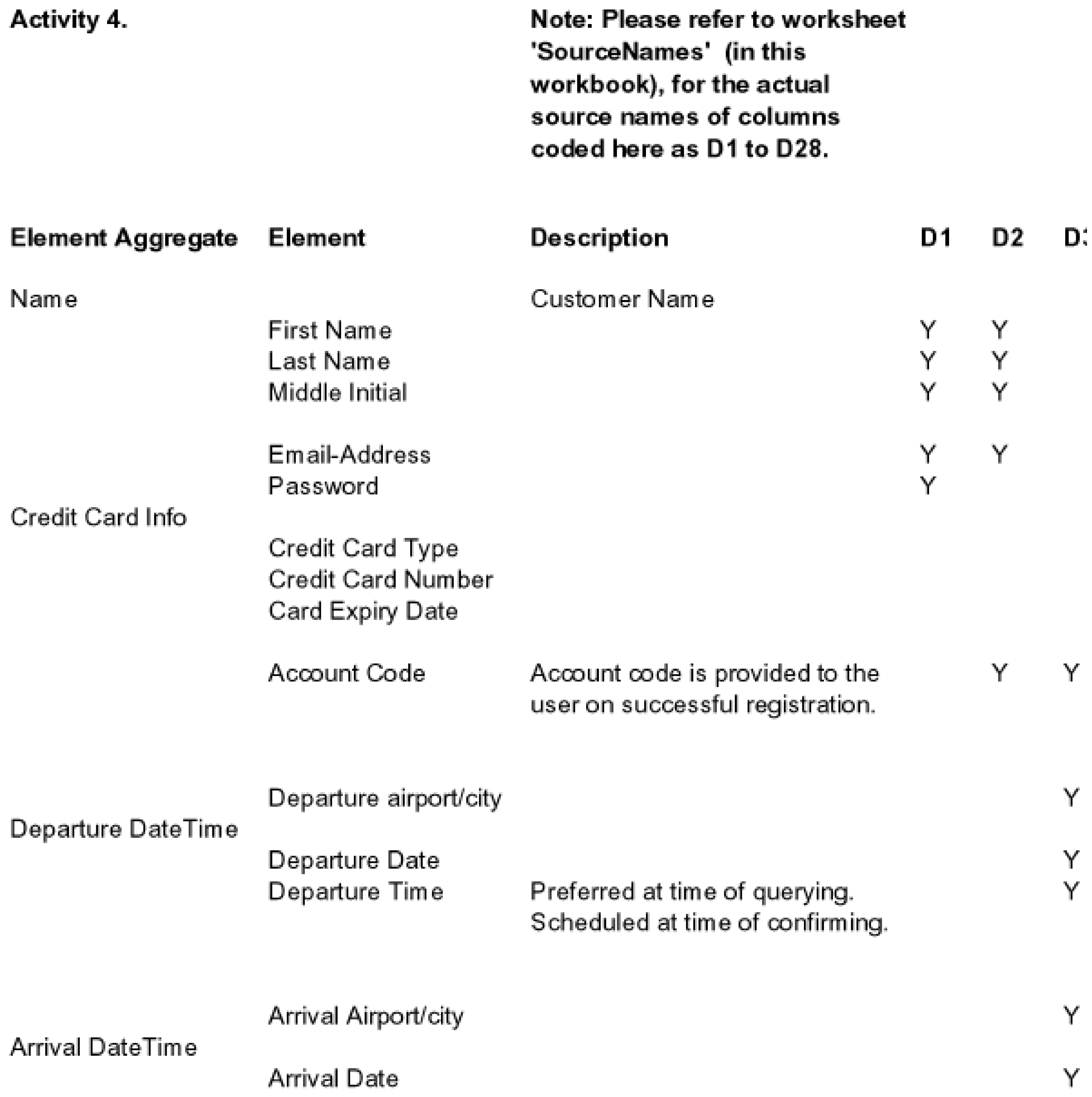
110. Transaction Worksheets



Business Transaction Name	Request Account Setup
Form ID	PIP 1A1
Transaction Pattern	Commercial Transaction
Initial Partner Type	Customer
Initial Partner Role	Account Requester
Responding Partner Type	Travel Service Provider
Responding partner Role	Service Supplier
Preconditions	Customer member of Travel Service
Begins when	Account Requested
Ends when	Account Created
Exceptions	Account request rejected
Post conditions	Account Creation

- Uses RosettaNet PIPs (standard supply chain processes) as metamodel
- "The Query/Response process encapsulated in PIP 3A2 correspond to transactions 3.1, 3.2, 5.1 and 3.7, 3.8, 5.4. Each of these Service-Service transactions begins with a query (3.1, 3.2, 5.1) and end with a response (3.7, 3.8, 5.4)."
- "The Request/Response process encapsulated in PIP 3A1 correspond to transactions 3.3, 3.4, 3.5, 3.6, 5.2 and 5.3. Each of these transactions is a Service-Service transaction, the only difference being the Initial Partner Type, and the Responding Partner Type."

111. Consolidating Candidate Components [1]

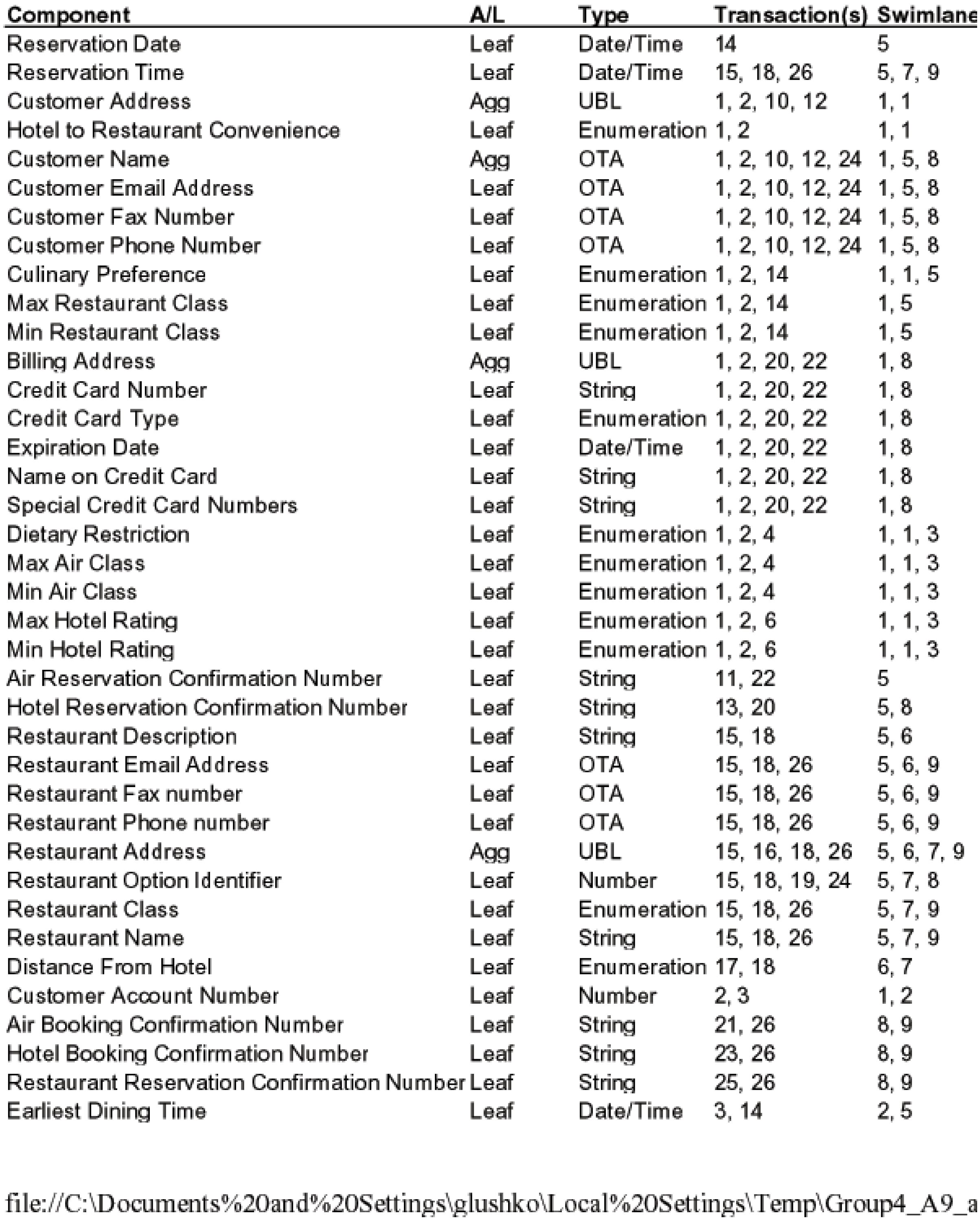


Activity 4.

Note: Please refer to worksheet 'SourceNames' (in this workbook), for the actual source names of columns coded here as D1 to D28.

Element Aggregate	Element	Description	D1	D2	D3
Name		Customer Name			
	First Name		Y	Y	
	Last Name		Y	Y	
	Middle Initial		Y	Y	
	Email-Address		Y	Y	
	Password		Y		
Credit Card Info					
	Credit Card Type				
	Credit Card Number				
	Card Expiry Date				
	Account Code	Account code is provided to the user on successful registration.		Y	Y
	Departure airport/city				Y
Departure DateTime					
	Departure Date				Y
	Departure Time	Preferred at time of querying. Scheduled at time of confirming.			Y
	Arrival Airport/city				Y
Arrival DateTime					
	Arrival Date				Y

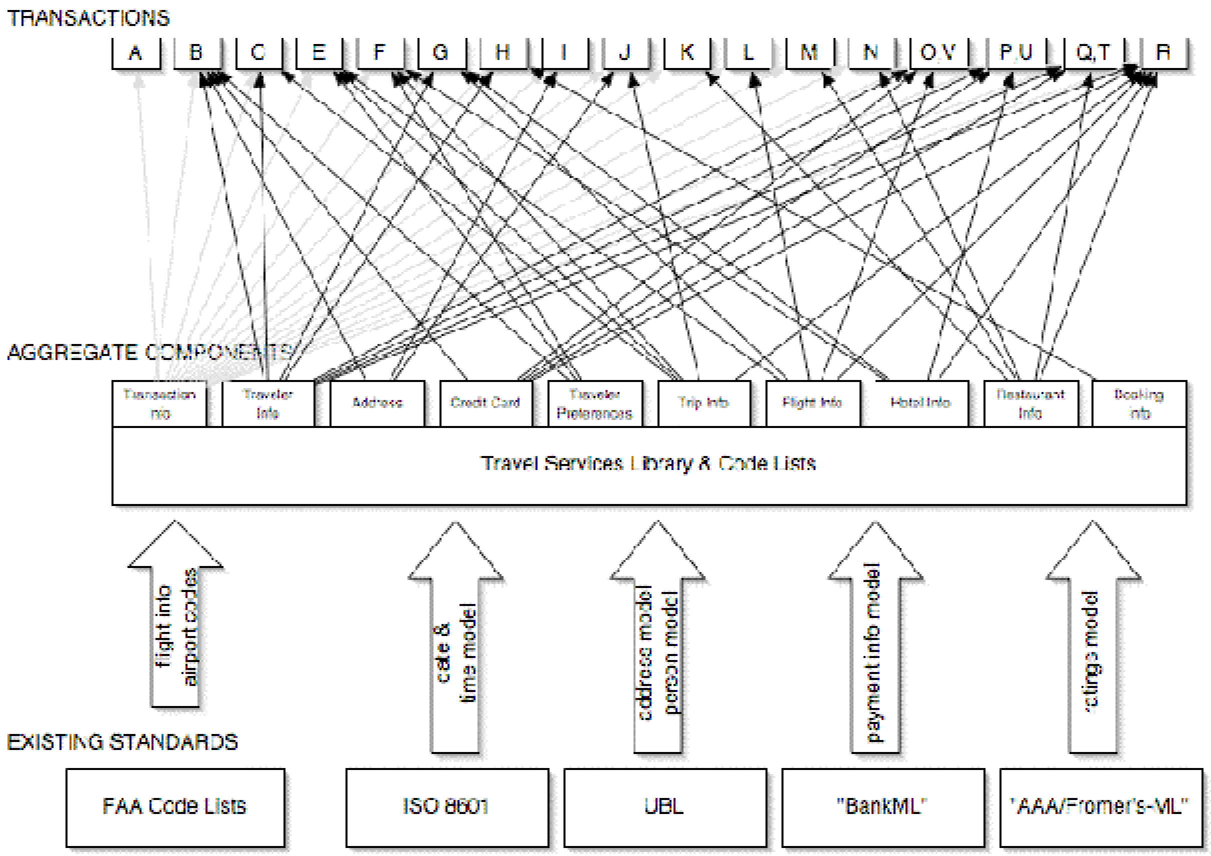
112. Consolidating Candidate Components [1]



Component	A/L	Type	Transaction(s)	Swimlane
Reservation Date	Leaf	Date/Time	14	5
Reservation Time	Leaf	Date/Time	15, 18, 26	5, 7, 9
Customer Address	Agg	UBL	1, 2, 10, 12	1, 1
Hotel to Restaurant Convenience	Leaf	Enumeration	1, 2	1, 1
Customer Name	Agg	OTA	1, 2, 10, 12, 24	1, 5, 8
Customer Email Address	Leaf	OTA	1, 2, 10, 12, 24	1, 5, 8
Customer Fax Number	Leaf	OTA	1, 2, 10, 12, 24	1, 5, 8
Customer Phone Number	Leaf	OTA	1, 2, 10, 12, 24	1, 5, 8
Culinary Preference	Leaf	Enumeration	1, 2, 14	1, 1, 5
Max Restaurant Class	Leaf	Enumeration	1, 2, 14	1, 5
Min Restaurant Class	Leaf	Enumeration	1, 2, 14	1, 5
Billing Address	Agg	UBL	1, 2, 20, 22	1, 8
Credit Card Number	Leaf	String	1, 2, 20, 22	1, 8
Credit Card Type	Leaf	Enumeration	1, 2, 20, 22	1, 8
Expiration Date	Leaf	Date/Time	1, 2, 20, 22	1, 8
Name on Credit Card	Leaf	String	1, 2, 20, 22	1, 8
Special Credit Card Numbers	Leaf	String	1, 2, 20, 22	1, 8
Dietary Restriction	Leaf	Enumeration	1, 2, 4	1, 1, 3
Max Air Class	Leaf	Enumeration	1, 2, 4	1, 1, 3
Min Air Class	Leaf	Enumeration	1, 2, 4	1, 1, 3
Max Hotel Rating	Leaf	Enumeration	1, 2, 6	1, 1, 3
Min Hotel Rating	Leaf	Enumeration	1, 2, 6	1, 1, 3
Air Reservation Confirmation Number	Leaf	String	11, 22	5
Hotel Reservation Confirmation Number	Leaf	String	13, 20	5, 8
Restaurant Description	Leaf	String	15, 18	5, 6
Restaurant Email Address	Leaf	OTA	15, 18, 26	5, 6, 9
Restaurant Fax number	Leaf	OTA	15, 18, 26	5, 6, 9
Restaurant Phone number	Leaf	OTA	15, 18, 26	5, 6, 9
Restaurant Address	Agg	UBL	15, 16, 18, 26	5, 6, 7, 9
Restaurant Option Identifier	Leaf	Number	15, 18, 19, 24	5, 7, 8
Restaurant Class	Leaf	Enumeration	15, 18, 26	5, 7, 9
Restaurant Name	Leaf	String	15, 18, 26	5, 7, 9
Distance From Hotel	Leaf	Enumeration	17, 18	6, 7
Customer Account Number	Leaf	Number	2, 3	1, 2
Air Booking Confirmation Number	Leaf	String	21, 26	8, 9
Hotel Booking Confirmation Number	Leaf	String	23, 26	8, 9
Restaurant Reservation Confirmation Number	Leaf	String	25, 26	8, 9
Earliest Dining Time	Leaf	Date/Time	3, 14	2, 5

file://C:\Documents%20and%20Settings\glushko\Local%20Settings\Temp\Group4_A9_a

113. Document Model Assembly



114. Summary

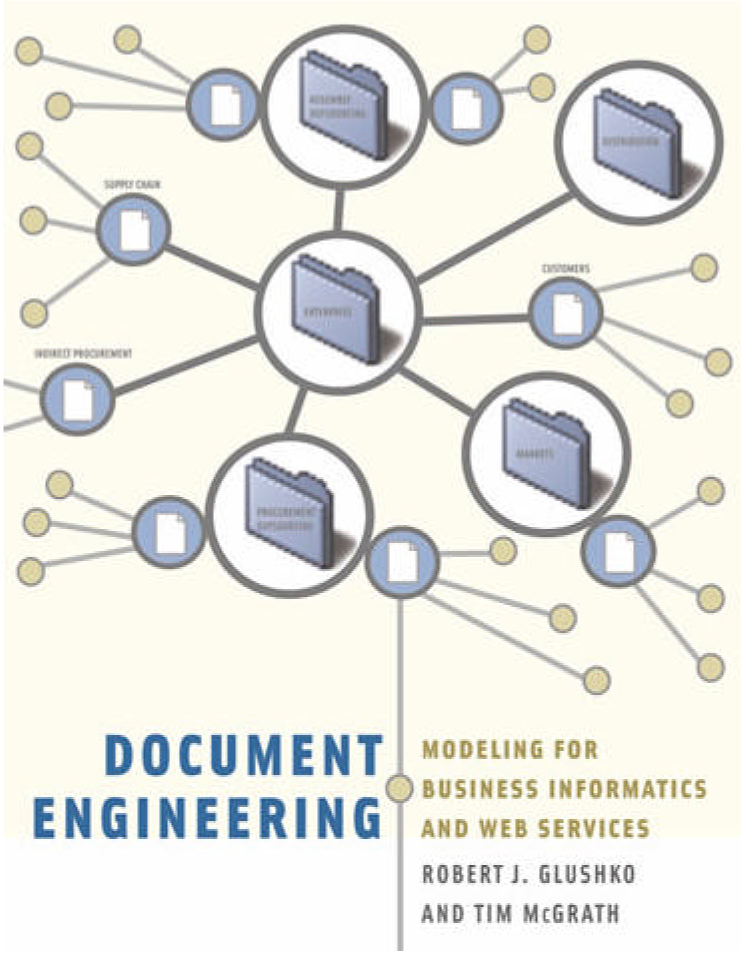
- "Document Engineering" is evolving as a new discipline for specifying, designing, and implementing the electronic documents that request or provide interfaces to business processes via Web-based services
- Best practices in Document Engineering require and reinforce the identification and reuse of patterns of information exchange from the perspective of business models, business processes, and document exchanges
- Doing business requires both "publication-like" document types like brochures and technical manuals and "transactional" documents like purchase orders and invoices – so we need analysis and design methods that work for both ends of this "Document Type Spectrum"
- Document Engineering can emphasize what these analysis and design approaches have in common rather than highlighting their differences
  - Designing, describing, and organizing components to facilitate their reuse
  - Assembling hierarchical document models that organize components according to the requirements of a specific context for information exchange
- The methodology we've systematized for Document Engineering seems to be interesting, learnable, and usable

115. Acknowledgments

- Much of this material comes from a book called *Document Engineering: Modeling for Business Informatics and Web Services* by Robert J. Glushko and Tim McGrath to be published in 2005 by MIT Press

- Three years of students at the University of California, Berkeley have contributed to its development through courses and research projects with the first author
- The methodology has been significantly refined through its use by the library content team of the Universal Business Language initiative, led by the second author

116. Document Engineering: The Book


SUPPLY CHAIN
CUSTOMERS
INDIRECT PROCUREMENT
DOCUMENT ENGINEERING
MODELING FOR BUSINESS INFORMATICS AND WEB SERVICES
ROBERT J. GLUSHKO AND TIM McGRATH