DOCUMENT ENGINEERING SERVICES
supporting universal business

COMPOSING MEANINGFUL DOCUMENTS WITH UBL

Tim McGrath

Open Standards 2008: Composability within SOA Symposium

Date: 28 April – 1 May

Location: Santa Clara Marriott

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

COMPOSING MEANINGFUL DOCUMENTS WITH UBL

- an approach to design and customization of UBL to satisfy specific requirements

UBL is establishing itself as the language of choice for a variety of electronic commerce applications. It is particularly well suited for use in business web services and service oriented architectures. The use of UBL for such environments ensures that services remain meaningful and worthy in a variety of different compositions. Inevitably, these applications may require some refinement of the standard structures.

This presentation explains some of the design principles behind UBL and how both the standard components and the principles behind them can be adapted to suit the requirements of different business processes. We will explore a case study involving the Danish Government's OIOUBL and Service Oriented Infrastructure project.

Composing different services into a meaningful business process requires interoperability of the information contained in the documents exchanged. This tutorial will discuss the opportunities for utilizing the OASIS Universal Business Language (UBL) as the glue to bind these services together at a business data level.

This tutorial explains the design principles behind UBL and how it can be adapted to suit the requirements of different business processes. These composite applications will require some customization of the standard structures and the creation of new document types.

The objective is to enable participants to understand the issues involved in customizing UBL and how to develop specific implementations that suit the requirements of their context of use. The target audience is system analysts and developers responsible for the implementation of document interfaces.

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Outline

- Background to Document Design
- UBL Design
- UBL Specification
- Case Study Scenario
- Designing Customizations
- Specifying Customizations
- Validating Customizations
- Summary and Feedback

© Document Engineering Services 2008

Introductions

Who are we and why are we here?

© Document Engineering Services 2008

Background to Document Design

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

What are Documents?

- For nearly three thousand years information has been organized in purposeful and self-contained packages.
  - We call them documents.
- Using documents for exchanging business information is an old idea.
  - The technology for encoding and exchanging documents has profoundly changed

  but
  - The concept of a document has remained surprisingly stable.

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Using documents for exchanging business information is natural and intuitive.

Administrative documents
Early Dynastic III, about 2500 BC
Probably from Shuruppak (Fara)

Left: the governor receives 14 shekels of silver from 5 persons, and 46 shekels remain unpaid by 13 other persons. The total amount of silver is 1 mina.

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

XML Document

```
<cac:TaxTotal>
  <cbc:TaxAmount currencyID="SHEKEL">60.00</cbc:TaxAmount>
  <cac:TaxSubtotal>
    <cbc:TaxableAmount currencyID="SHEKEL">100.00</cbc:TaxableAmount>
    <cbc:TaxAmount currencyID="SHEKEL">14.00</cbc:TaxAmount>
    <cac:TaxCategory>
      <cac:TaxScheme>
        <cbc:ID>ASSYRIAN GOVERNOR TAX</cbc:ID>
      </cac:TaxScheme>
    </cac:TaxCategory>
  </cac:TaxSubtotal>
</cac:TaxTotal>
```

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

How Documents Work

- Documents are self-contained collections of information.
  - Interfaces for people.
  - Interfaces to business processes.
- To be meaningful documents must be defined using a common language.
  - Not for information content

  but
  - For describing the meaning (semantics) of the information content.
- The Universal Business Language is a semantic document description language.

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Document Exchange

- A document exchange consists of both processes and the documents (information) they produce and consume.
- By understanding the processes
  - we learn what kinds of information is needed.
- By understanding the document
  - we learn what kinds of processes are possible.

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Interoperability

- A basic requirement for two businesses to conduct business is that their business systems interoperate.
- Easy to express but hard to achieve.
- Variations in strategies, technology platforms, legacy applications, business processes, and terminology.
- Interoperability doesn't require that two systems be identical.
  - Just they have common understanding of the meanings of things (semantics)

© Document Engineering Services 2008

UBL Design

© Document Engineering Services 2008

UBL Design

ebXML Core Components

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

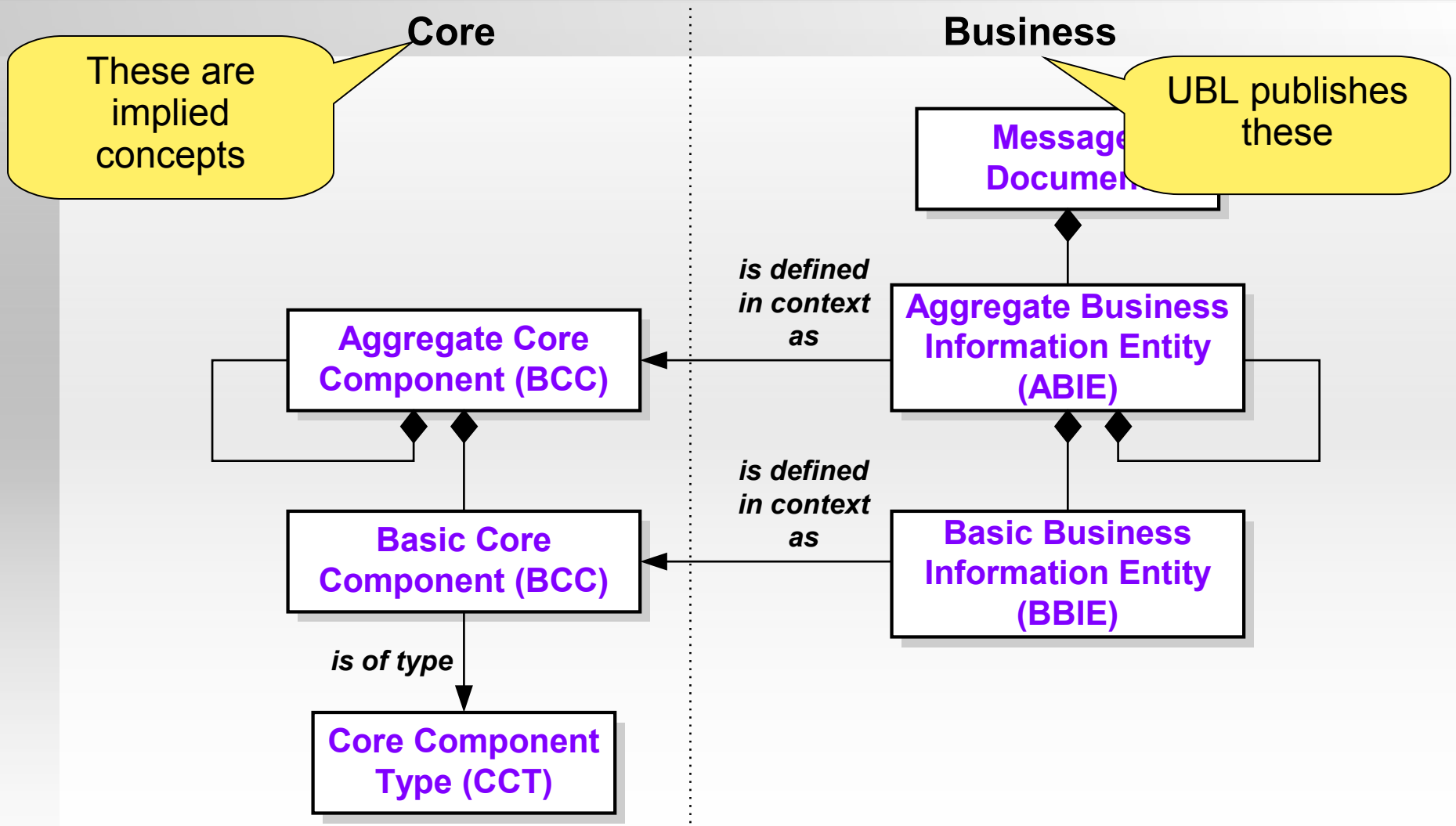
ebXML CCTS

- ebXML Core Components Technical Specification:
  - ISO 15000-5.
  - UN/CEFACT
- How to describe core components of business documents in a syntax independent manner.
- UBL is the first set of integrated XML document specifications based on CCTS.
  - UN/CEFACT recognizes UBL as appropriate first-generation XML documents for eBusiness.

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

CCTS Principles



© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

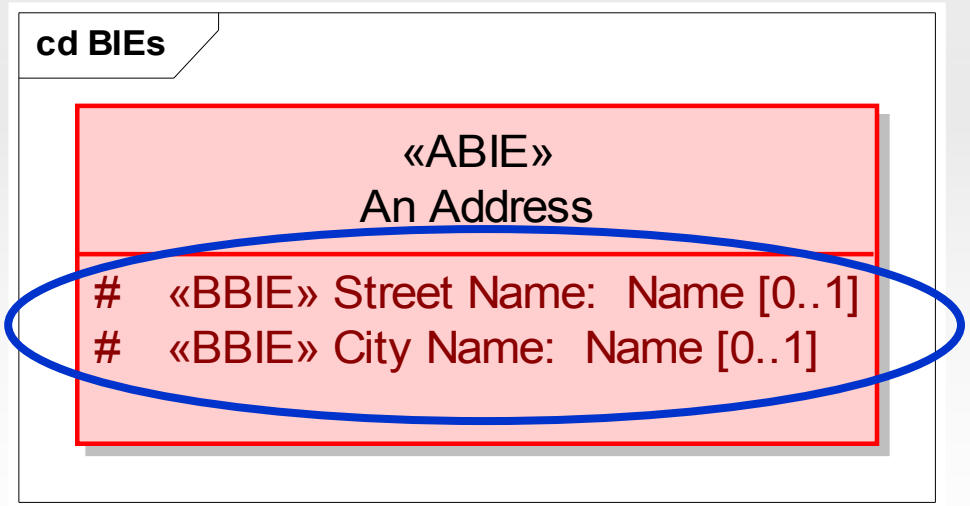
Business Information Entities

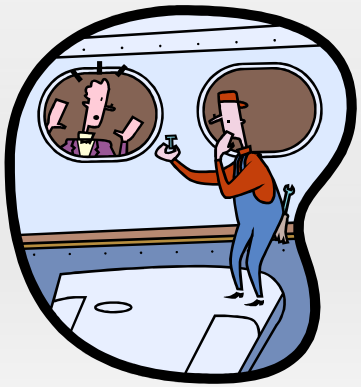
- Reusable building blocks for the exchange of information ("Core Components") in a specific Business Context.
- Aggregate BIE
  - A collection of related pieces of business information that together convey a distinct business meaning.
- Association BIE
  - Is associated to an Aggregate Business Information Entity, which describes its structure.
- Basic BIE
  - A singular business characteristic of a specific aggregation.

© Document Engineering Services 2008

Basic Business Information Entities

- A singular business characteristic in a specific business context.
- Individual pieces of information.



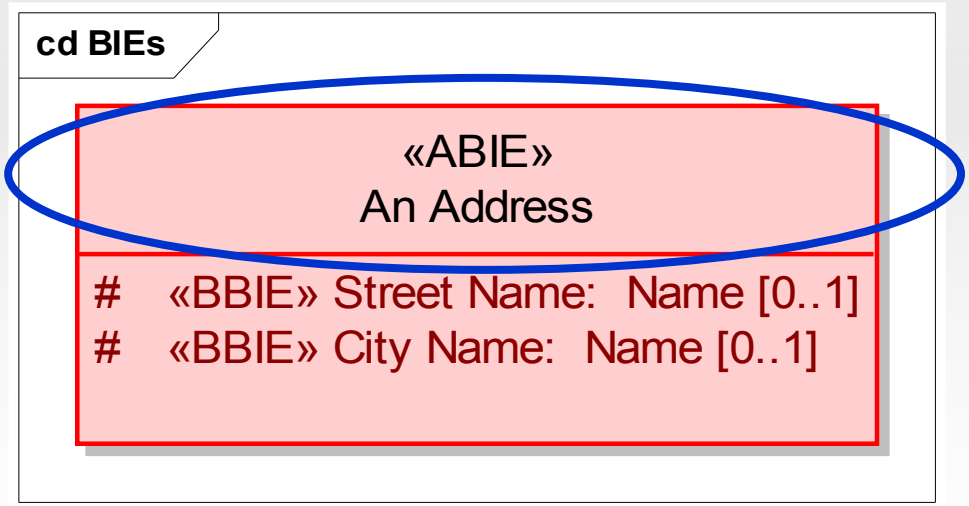


© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Aggregate Business Information Entities

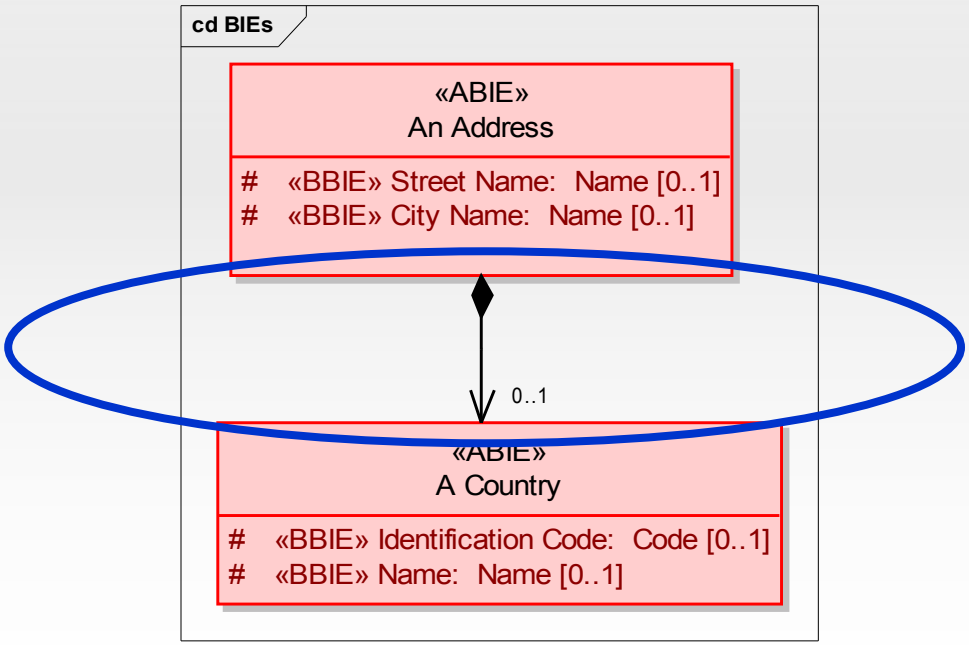
- A collection of related pieces of business information.
- A container of BBIEs and ASBIEs to other ABIEs.



© Document Engineering Services 2008

Association
Business Information Entities

- Represents an ABIE in a specific business context.
- An ABIE contained within another ABIE.



© Document Engineering Services 2008

UBL Design

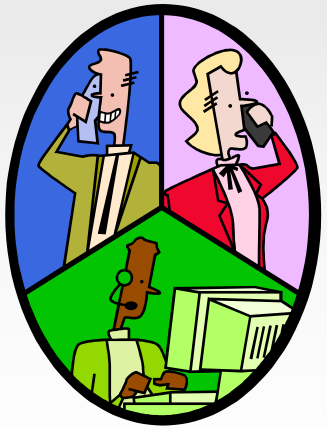
Naming Information Entities

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Names

- Meaningful Information Entity names.
  - promote a common understanding.
  - encourage reuse.
- May need different names in different modeling artifacts.
  - XML Tag may not be the modeling or the presentation name.



© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Choosing the Right Names

- Selecting meaningful terminology for names is a craft not a science.
  - Think about spoken languages.
  - No two designers will choose the same terms.
- Levels of discipline:
  - Dictionaries.
  - Controlled vocabularies.
  - Formal ontologies.

- We need more information than just names.
- We also need to understand the context of use.

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Rules for Consistent Naming

- Names should be grammatically consistent.
  - Singular not plurals
- Names should not be ambiguous.
  - Is "Supplier's ID" the same as "ID of Supplier"?
- Check names are not semantically misleading.
  - "Product Number" may not be a number, so "Product Identifier" is better.
- Qualify names to clarify the context of use.
  - "Customer Party" is more meaningful than "Customer".

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

ebXML Naming Rules

- ebXML Core Components naming rules are an application of ISO 11179.
- All Information Entities get a Dictionary Entry Name (DEN).
- Dictionary Entry Names are globally unique.
- They are the 'key' or unique identifier for any Information Entity.

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Dictionary Entry Names

{qualifier} ***Object Class Term***, ". ",
{qualifier} ***Property Term***, ". ",
{qualifier} ***Representation Term***

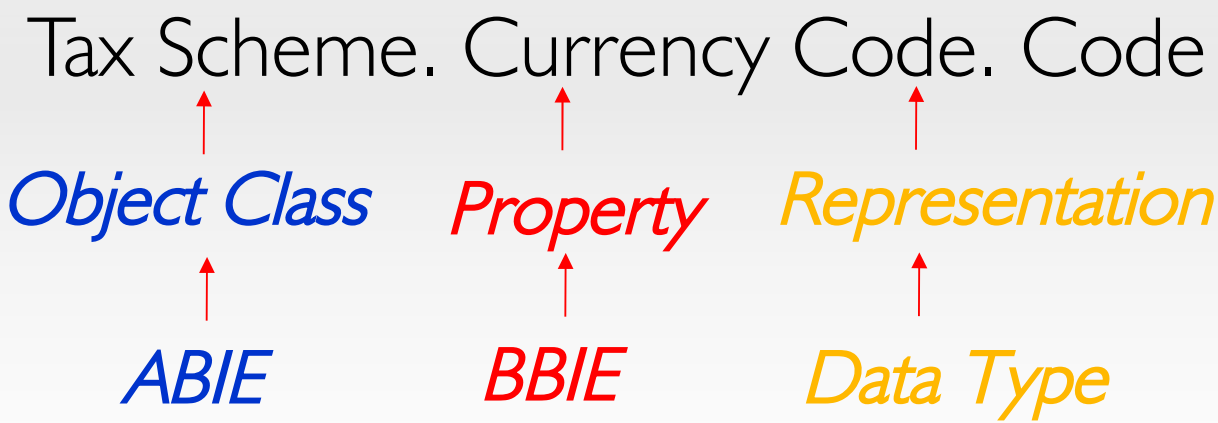
- Object Class Term
  - The object class to which it belongs.
- Property Term
  - Reflecting its function as a property or distinguishing characteristic of the object class.
- Representation Term
  - How it is represented.

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Examples of CCTS Names

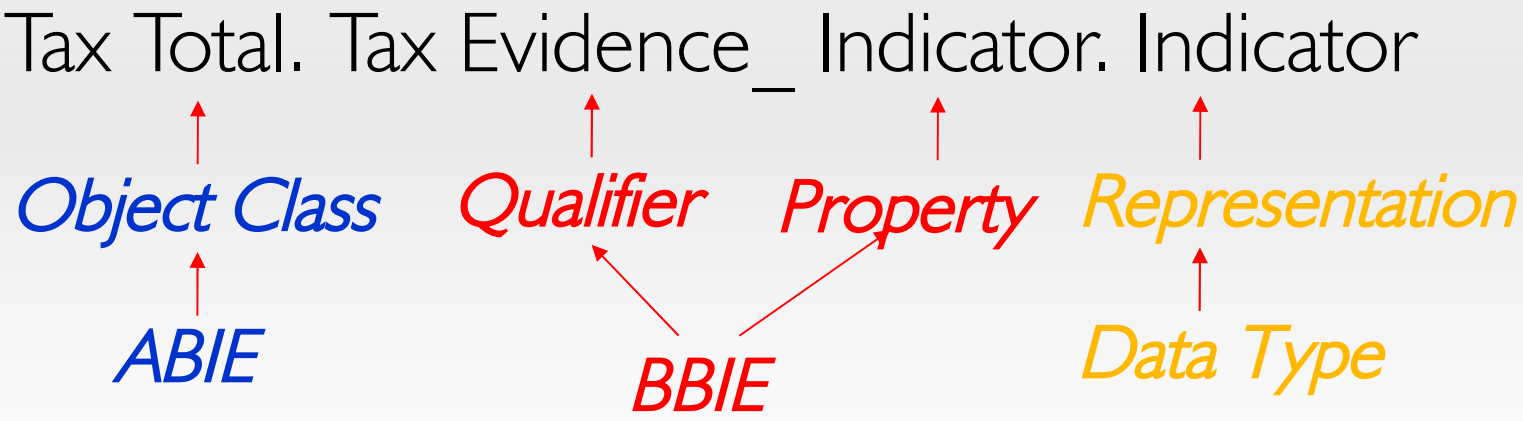
- The name for '*The currency in which the tax is collected and reported, expressed as a code*':



© Document Engineering Services 2008

Qualified CCTS Names

- The name for '*An indicator as to whether these totals are recognized as legal evidence for taxation purposes*':



© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

CCTS Truncation Rule

- The name for '*Identifies the tax scheme*':

  Tax Scheme. Identifier. Identifier

- Can be truncated to remove duplicate terms:

  Tax Scheme. Identifier

© Document Engineering Services 2008

UBL Design

Representation Terms

© Document Engineering Services 2008

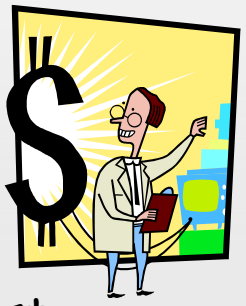
Amount

Content

A number of monetary units specified in a currency where the unit of currency is explicit or implied.

Amount. Currency. Identifier

The currency of the amount.



Amount. Currency. Code List Version Identifier

The version of the UN/ECE rec. 9 code list.

```
<cbc:TaxAmount currencyID="DKK">1262.50</cbc:TaxAmount>
```

© Document Engineering Services 2008

Binary Object

(Graphic, Picture, Sound, Video)

Binary Object. Content	A set of finite-length sequences of binary octets.
Binary Object. Format. Text	The format of the binary content.
Binary Object. Mime. Code	The mime type of the binary object.
Binary Object. Encoding. Code	Specifies the decoding algorithm of the binary object.
Binary Object. Character Set. Code	The character set of the binary object if the mime type is text.
Binary Object. Uniform Resource. Identifier	The Uniform Resource Identifier that identifies where the binary object is located.
Binary Object. Filename. Text	The filename of the binary object.

```
<cbc:EmbeddedDocumentBinaryObject mimeCode="application/pdf"
encodingCode="Base64" characterSetCode="UTF-8">
brIG0Ng53KHKaoOQNQYcpyFHKa9UhkgzjkPBTaFJDIBUasfdgartujjuoprr
wer34578deewcwetyuuutyur643432THyB4KY5SYM5BZcpMkshU0UzJoL
HSDEWFdreefssd///
</cbc:EmbeddedDocumentBinaryObject>
```

DOCUMENT ENGINEERING SERVICES
supporting universal business

Code

Code. Content	A character string (letters, figures or symbols) that for brevity and/or language independence may be used to represent or replace a definitive value or text of an attribute.
Code List. Identifier	The identification of a list of codes.
Code List. Agency. Identifier	An agency that maintains one or more code lists.
Code List. Agency Name. Text	The name of the agency that maintains the code list.

```
<cbc:ItemClassificationCode listID="UNSPSC"  listVersionID="7.0401">
43191501
</cbc:ItemClassificationCode>
```

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Code (contd.)

Code List. Name. Text	The name of a list of codes.
Code List. Version. Identifier	The version of the code list.
Code. Name. Text	The textual equivalent of the code content.
Language. Identifier	The identifier of the language used in the corresponding text string.
Code List. Uniform Resource. Identifier	The Uniform Resource Identifier that identifies where the code list is located.
Code List Scheme. Uniform Resource. Identifier	The Uniform Resource identifier that identifies where the code list scheme is located.

```
<cbc:ItemClassificationCode listID="UNSPSC"  listVersionID="7.0401">
43191501
</cbc:ItemClassificationCode>
```

© Document Engineering Services 2008

Date Time
(Date, Time)

Date Time. Content — The particular point in the progression of time.

Date Time. Format. Text — The format of the date/time content.

<cbc:StartDate>2007-01-25</cbc:StartDate>

© Document Engineering Services 2008

Identifier

Identifier. Content

A character string used to identify and distinguish uniquely, one instance of an object in an identification scheme from all other objects within the same scheme.

Identification Scheme. Identifier

The identification of the identification scheme.

Identification Scheme. Name. Text

The name of the identification scheme.

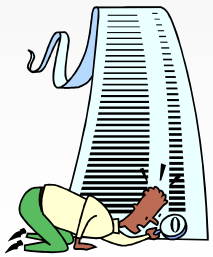
```
<cbc:ID schemeID="GTIN">5712345780121</cbc:ID>
```

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Identifier (contd.)

Identification Scheme Agency. Identifier	The identification of the agency that maintains the identification scheme.
Identification Scheme. Agency Name. Text	The name of the agency that maintains the identification scheme.
Identification Scheme. Version. Identifier	The version of the identification scheme.
Identification Scheme Data. Uniform Resource. Identifier	The Uniform Resource Identifier that identifies where the identification scheme data is located.
Identification Scheme. Uniform Resource. Identifier	The Uniform Resource Identifier that identifies where the identification scheme is located.



© Document Engineering Services 2008

Indicator

Indicator. Content	The value of the indicator.
Indicator. Format. Text	Whether the indicator is numeric, textual or binary.

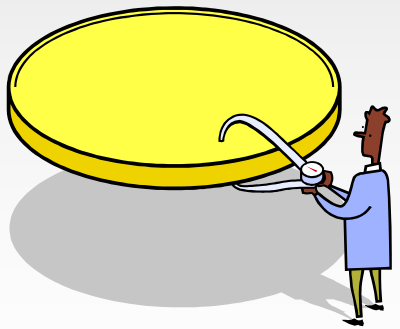
<cbc:ChargeIndicator>false</cbc:ChargeIndicator>

© Document Engineering Services 2008

Measure

Measure. Content	The numeric value determined by measuring an object.
Measure Unit. Code	The type of unit of measure.
Measure Unit. Code List Version. Identifier	The version of the measure unit code list.

```
<cbc:GrossWeightMeasure unitCode="KGS">
21408.79
</cbc:GrossWeightMeasure>
```



© Document Engineering Services 2008

Numeric
(Value, Rate, Percent)

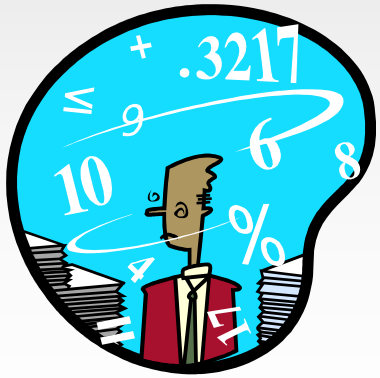
Numeric. Content

Numeric information that is assigned or is determined by calculation, counting or sequencing.

Numeric. Format.
Text

Whether the number is an integer, decimal, real number or percentage.

<cbc:PackSizeNumeric> | </cbc:PackSizeNumeric>



© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Quantity

Quantity. Content	A counted number of non-monetary units possibly including fractions.
Quantity. Unit. Code	The unit of the quantity.
Quantity Unit. Code List. Identifier	The quantity unit code list.
Quantity Unit. Code List Agency. Identifier	The identification of the agency that maintains the quantity unit code list.
Quantity Unit. Code List Agency Name. Text	The name of the agency which maintains the quantity unit code list.

<cbc:PackQuantity unitCode="EA">1</cbc:PackQuantity>

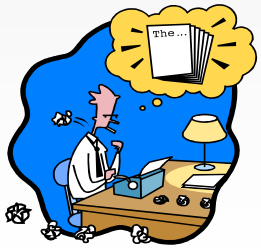
© Document Engineering Services 2008

Text
(Name)

Text. Content	A character string (I.e. a finite set of characters) generally in the form of words of a language.
Language. Identifier	The identifier of the language used in the corresponding text string.
Language. Locale. Identifier	The identification of the locale of the language.

<cbc:Description>Nokia Mobile telephone –Type ABC</cbc:Description>

Note: The Representation term "Text" is not part of the tag name (UBL naming rule)



© Document Engineering Services 2008

UBL Design

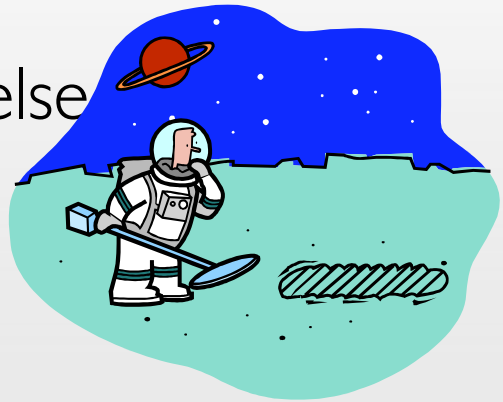
Codes and Identifiers

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Sets of Codes

- A special set of possible values.
- The code value refers to something else
- May be symbolic representations:
  - Initialisms ("EU"),
  - Acronyms ("SARS"),
  - Apocopations ("Reefer") or
  - Arbitrary abbreviations ("XML").
- But maybe not...
  - Both ISO code "CNY" and "156" identify the Chinese Yuan Renminbi.



© Document Engineering Services 2008

Codes and Values

AV	Capsule
BA	Barrel
BB	Bobbin
BC	Bottlecrate / bottlerack
BF	Balloon, non-protected
BG	Bag
BH	Bunch
BI	Bin
BJ	Bucket
BK	Basket

Refers to

Types of packaging codes

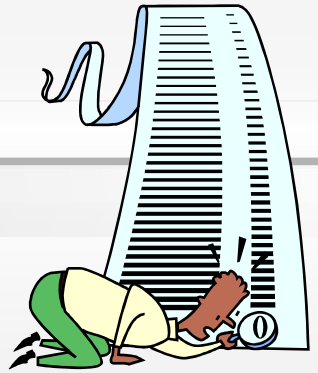
KMT	kilometre
C45	nanometre
C52	picometre
A71	femtometre
A45	decametre
NMI	nautical mile
A11	
A12	
C63	parsec
AK	fathom
X1	chain
INH	inch
M7	micro-inch
FOT	foot
YRD	yard

Refers to

Units of Measurement codes

© Document Engineering Services 2008

Identifiers



- Have meaning in their own right.
  - Don't have to reference anything.
- May be a constrained set of values:
  - Set of Driver's License numbers issued.
- May be symbolic representations (like codes).
- Or (effectively) not…
  - The values for identifying a phone number.
- We use codes as identifiers:
  - Digits in phone numbers are codes for regions.
- But not identifiers as codes:
  - What does your Driver's License number refer to?

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

What are these?

ABCUIJKL123456XYZ
ABCU123456789000
ABCU123456789001

Waybill Numbers

And these?

Shipping Container Numbers

OCLU1234567
NHKL7654321
HPGL8877665

© Document Engineering Services 2008

UBL Specification

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

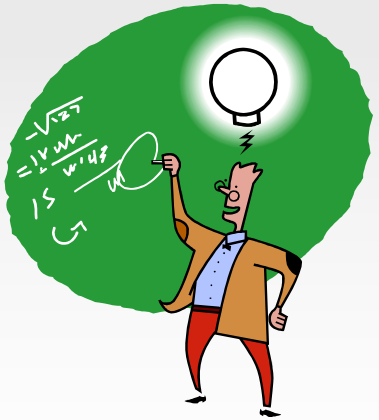
Specifying Documents

- We need a range of metadata to support the explanation of meanings.
  - Names are not enough.
- We define these in models.

© Document Engineering Services 2008

UBL Specification

Models and Modeling



© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

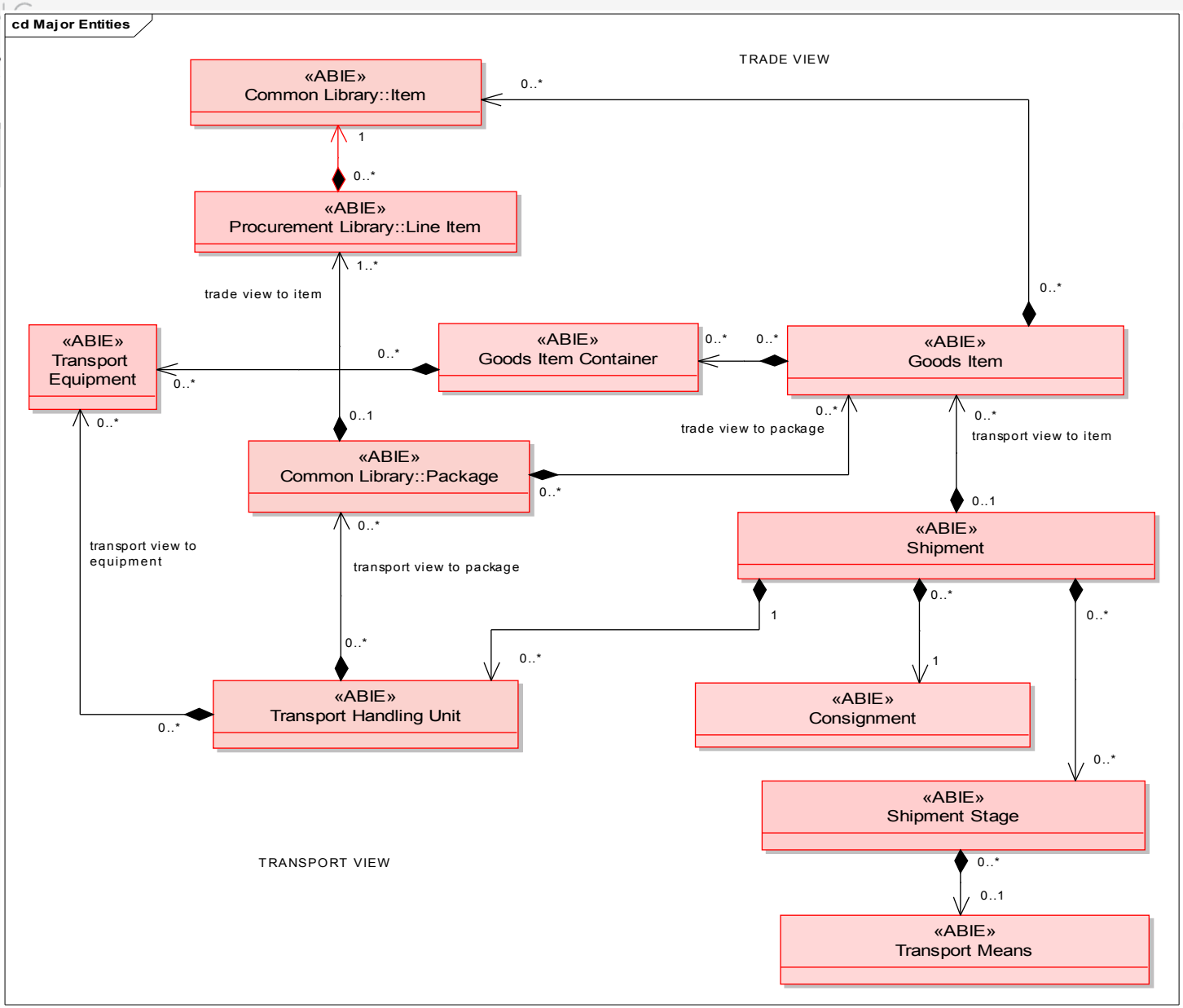
Component Models

- A generalized, conceptual model.
  - "webs" or networks.
  - Domain Model, Enterprise Data Model.
- Not a model of one document…

  but,
- A model of Information Entities usable in all documents.
  - and databases, etc.
- Complexity depends on the number of the business rules.

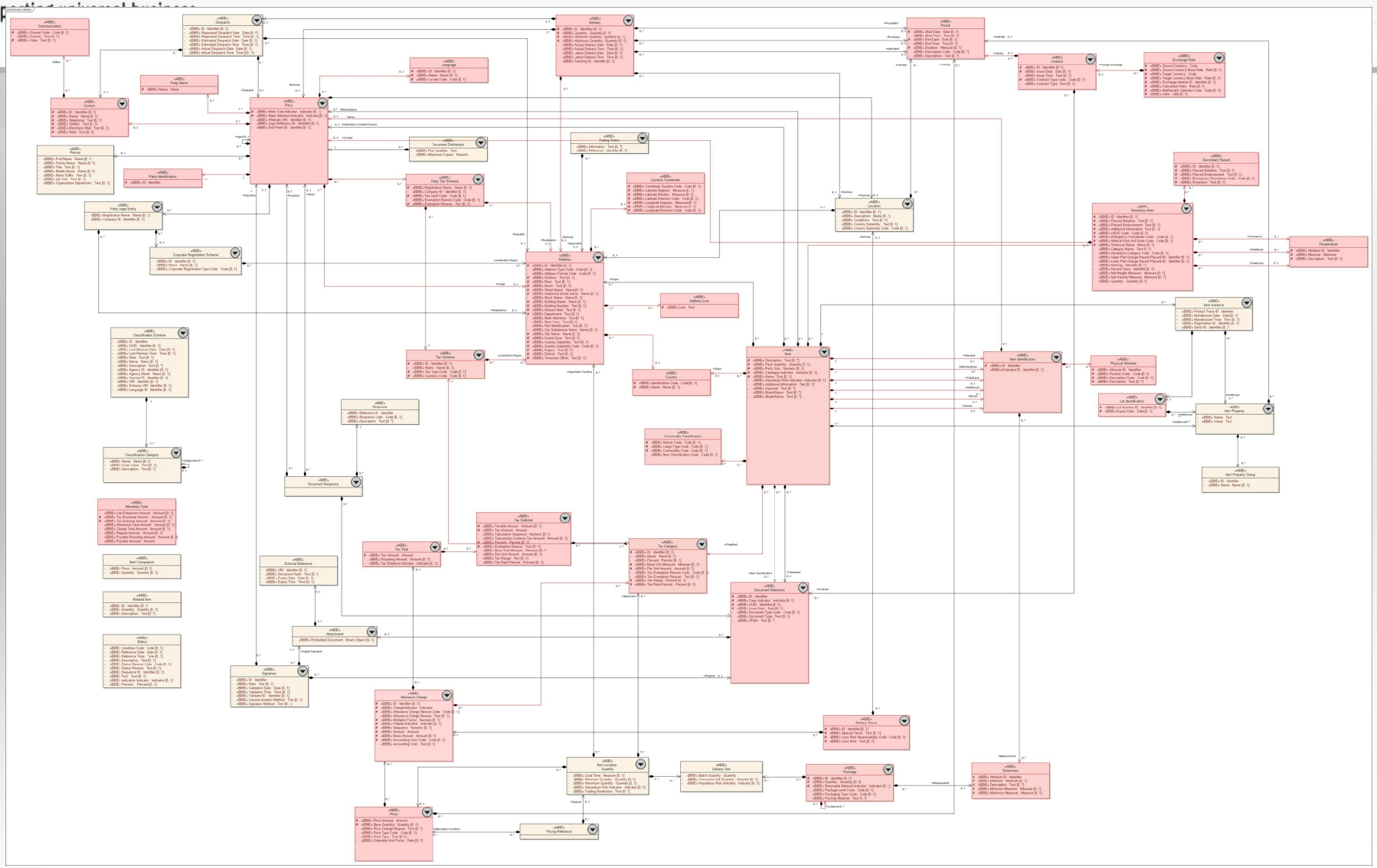
© Document Engineering Services 2008

A Component Model



DOCUMENT ENGINEERING SERVICES

UBL's Common Components



© Document Engineering Services 2008

UBL Specification

Assembling Components into Documents

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

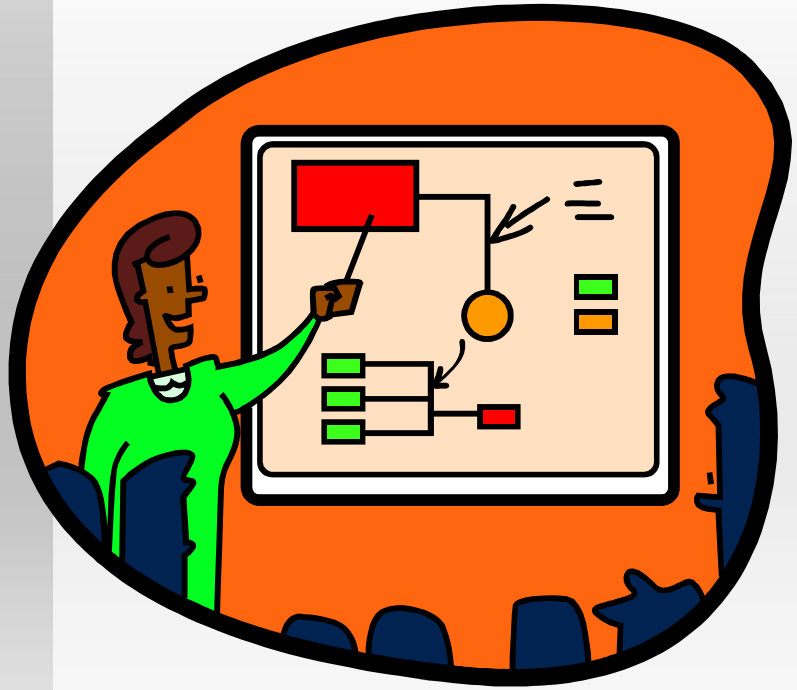
Documents are Hierarchical

- Inverse tree structure.
- Can be represented in a 2D space.
  - That's why markup languages need just <start> and </end> tags.
- Nesting imposes meaning.
  - When "Contact" contains "Name" we know that the name is that of the contact.
  - Parent (genealogy) gives contact of use.

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

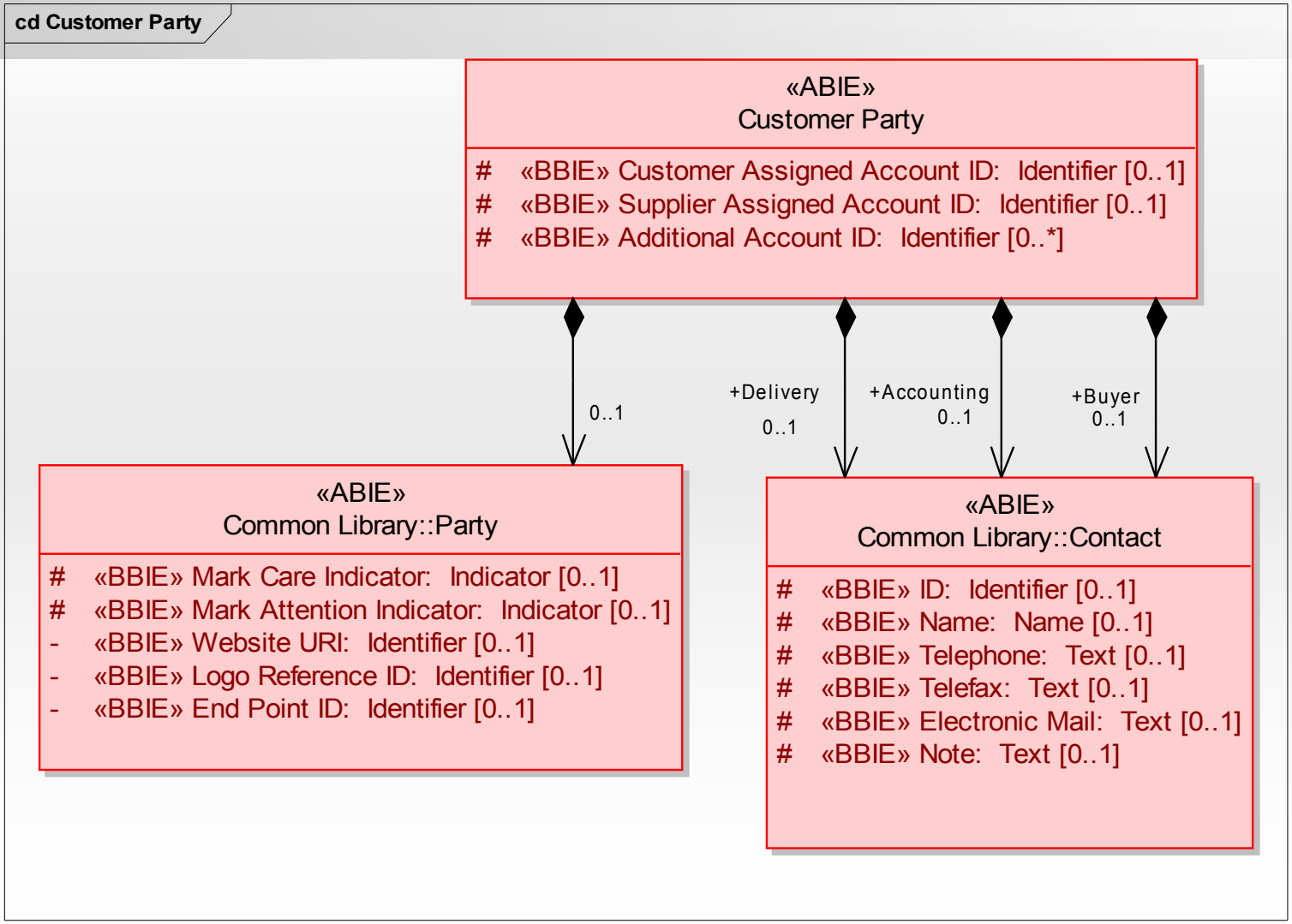
Hierarchy not Network



NOT

© Document Engineering Services 2008

A Document Assembly Model



© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Another Document Assembly Model

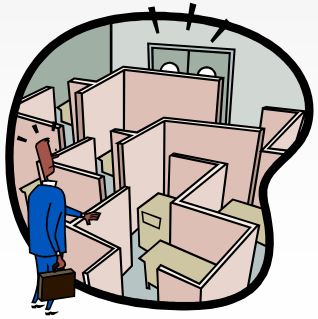
UBL Name	Dictionary Entry Name	Cardinality	Definition
CustomerParty	Customer Party. Details		Information about the Customer Party.
CustomerAssignedAccountID	Customer Party. Customer Assigned_ Account Identifier. Identifier	0..1	An identifier for the Customer's account, assigned by the Customer itself.
SupplierAssignedAccountID	Customer Party. Supplier Assigned_ Account Identifier. Identifier	0..1	An identifier for the Customer's account, assigned by the Supplier.
AdditionalAccountID	Customer Party. Additional_ Account Identifier. Identifier	0..n	An identifier for the Customer's account, assigned by a third party.
Party	Customer Party. Party	0..1	An association to Party.
DeliveryContact	Customer Party. Delivery_ Contact. Contact	0..1	An association to Delivery Contact.
AccountingContact	Customer Party. Accounting_ Contact. Contact	0..1	An association to Accounting Contact (Customer).
BuyerContact	Customer Party. Buyer_ Contact. Contact	0..1	An association to Buyer Contact.

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Creating a Document Assembly

- Each type of document usually requires its own document assembly model.
- We create a document assembly model by:
  1. Choosing an "entry point" into the Document Component Model.
     - The entry point is a structure that will form the root of the document tree.
  2. Following a pathway through the component model
     - guided by business rules for the document required.

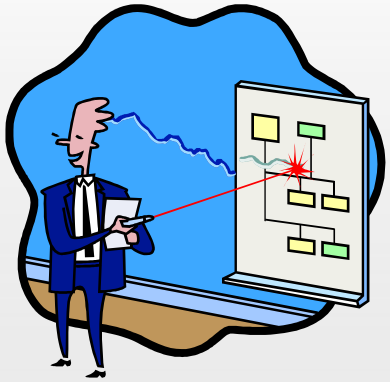


© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Assembling Associations

- If an association is mandatory:
  - must follow the path
  - the associated ABIE must be in the document.
- If an association is optional:
  - may follow the path
  - the associated ABIE can be in the document if required by business rules.
- Cardinality of associations controls the depth of the document hierarchy.



© Document Engineering Services 2008

UBL Specification

Assembly Patterns

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

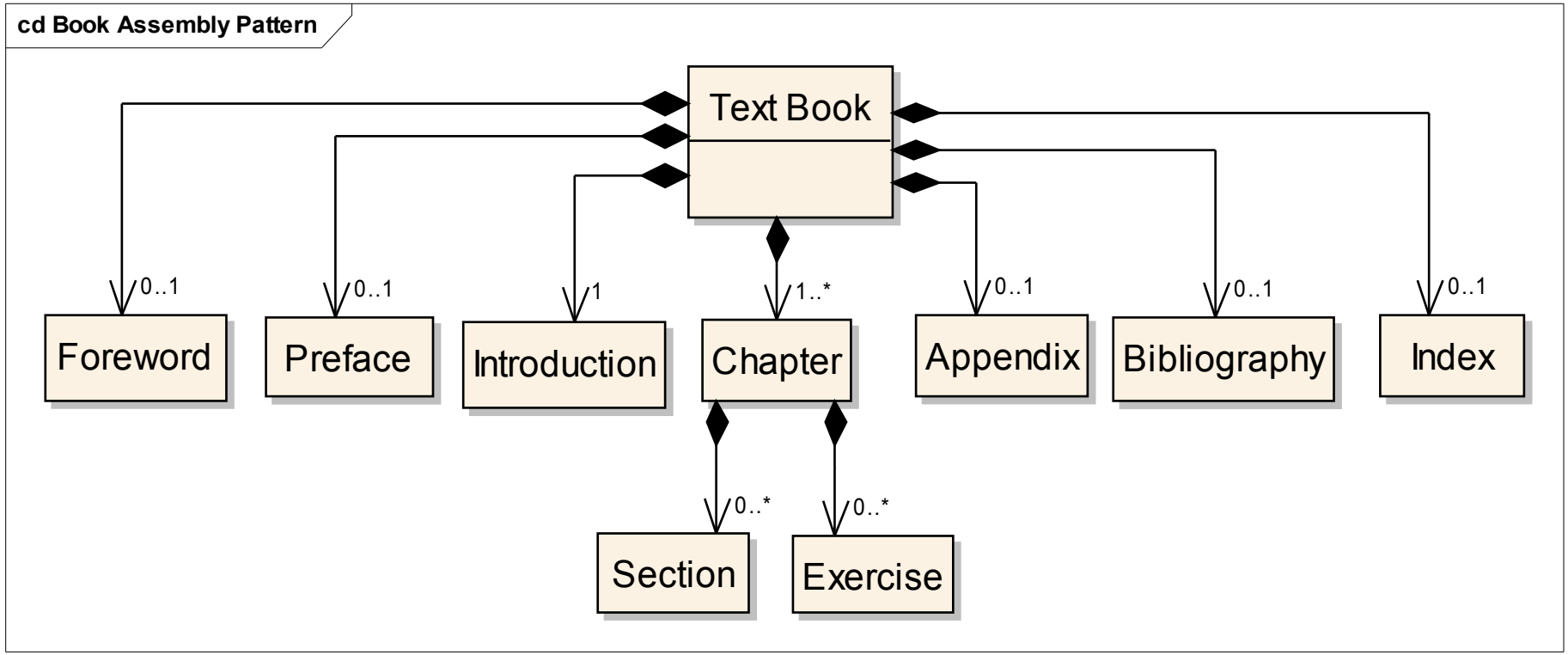
Patterns for Document Assembly

- Document assembly models express business rules based on the context of use.
- Many assemblies share common structures.
  - There are patterns of assemblies.
  - a Book, a Contract, an Order.
- Good designers employ these patterns…
  - Because processes will find their documents familiar.

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

The "Book" Assembly Pattern



© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Reusing Assembly Patterns

- Assembly patterns encourages the use of common libraries of components.
  - Reusing patterns from elsewhere in the model
  - Adopting patterns from outside
- The challenge is customizing these patterns to suit specific requirements.

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

COMPOSING MEANINGFUL DOCUMENTS WITH UBL

Tim McGrath

Open Standards 2008: Composability within SOA Symposium
Date: 28 April – 1 May
Location: Santa Clara Marriott

04/04/08

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

COMPOSING MEANINGFUL DOCUMENTS WITH UBL

- an approach to design and customization of UBL to satisfy specific requirements

UBL is establishing itself as the language of choice for a variety of electronic commerce applications. It is particularly well suited for use in business web services and service oriented architectures. The use of UBL for such environments ensures that services remain meaningful and worthy in a variety of different compositions. Inevitably, these applications may require some refinement of the standard structures.

This presentation explains some of the design principles behind UBL and how both the standard components and the principles behind them can be adapted to suit the requirements of different business processes. We will explore a case study involving the Danish Government's OIOUBL and Service Oriented Infrastructure project.

Composing different services into a meaningful business process requires interoperability of the information contained in the documents exchanged. This tutorial will discuss the opportunities for utilizing the OASIS Universal Business Language (UBL) as the glue to bind these services together at a business data level.

This tutorial explains the design principles behind UBL and how it can be adapted to suit the requirements of different business processes. These composite applications will require some customization of the standard structures and the creation of new document types.

The objective is to enable participants to understand the issues involved in customizing UBL and how to develop specific implementations that suit the requirements of their context of use. The target audience is system analysts and developers responsible for the implementation of document interfaces.

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Outline

- Background to Document Design
- UBL Design
- UBL Specification
- Case Study Scenario
- Designing Customizations
- Specifying Customizations
- Validating Customizations
- Summary and Feedback

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Introductions

Who are we and why are we here?

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Background to Document Design

04/04/08

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

What are Documents?

- For nearly three thousand years information has been organized in purposeful and self-contained packages.
  - We call them documents.
- Using documents for exchanging business information is an old idea.
  - The technology for encoding and exchanging documents has profoundly changed

  but
  - The concept of a document has remained surprisingly stable.

© Document Engineering Services 2008

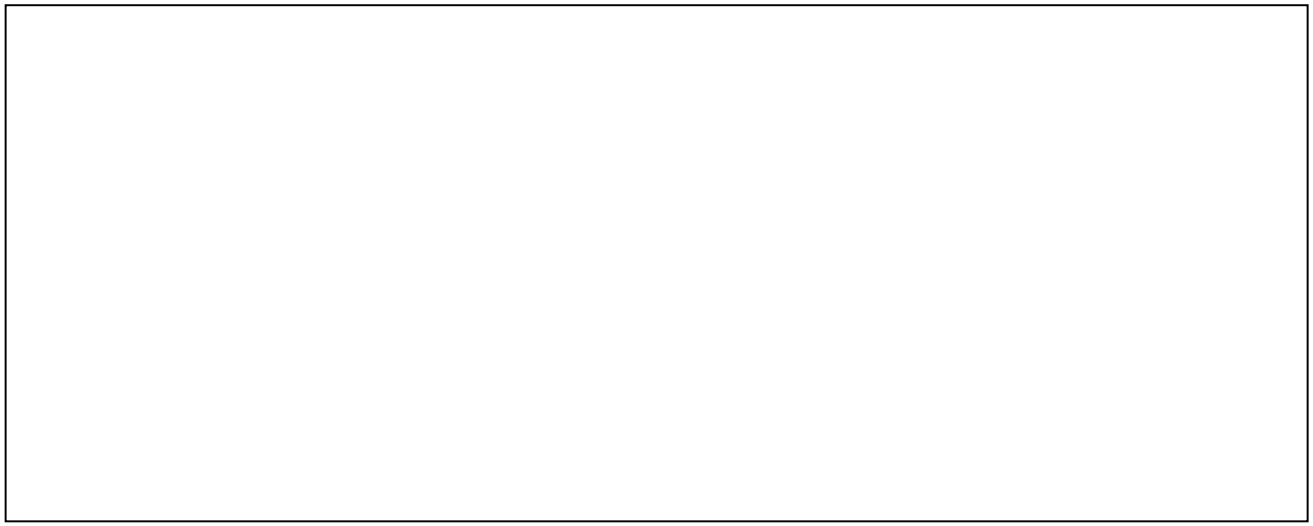
DOCUMENT ENGINEERING SERVICES
supporting universal business

Using documents for exchanging business information is natural and intuitive.

Administrative documents
Early Dynastic III, about 2500 BC
Probably from Shuruppak (Fara)

Left: the governor receives 14 shekels of silver from 5 persons, and 46 shekels remain unpaid by 13 other persons. The total amount of silver is 1 mina.

© Document Engineering Services 2008



@ Port Community Systems 2007

DOCUMENT ENGINEERING SERVICES
supporting universal business

How Documents Work

- Documents are self-contained collections of information.
  - Interfaces for people.
  - Interfaces to business processes.
- To be meaningful documents must be defined using a common language.
  - Not for information content

  but
  - For describing the meaning (semantics) of the information content.
- The Universal Business Language is a semantic document description language.

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Document Exchange

- A document exchange consists of both processes and the documents (information) they produce and consume.
- By understanding the processes
  - we learn what kinds of information is needed.
- By understanding the document
  - we learn what kinds of processes are possible.

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Interoperability

- A basic requirement for two businesses to conduct business is that their business systems interoperate.
- Easy to express but hard to achieve.
- Variations in strategies, technology platforms, legacy applications, business processes, and terminology.
- Interoperability doesn't require that two systems be identical.
  - Just they have common understanding of the meanings of things (semantics)

© Document Engineering Services 2008

UBL Design

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

UBL Design
ebXML Core Components

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

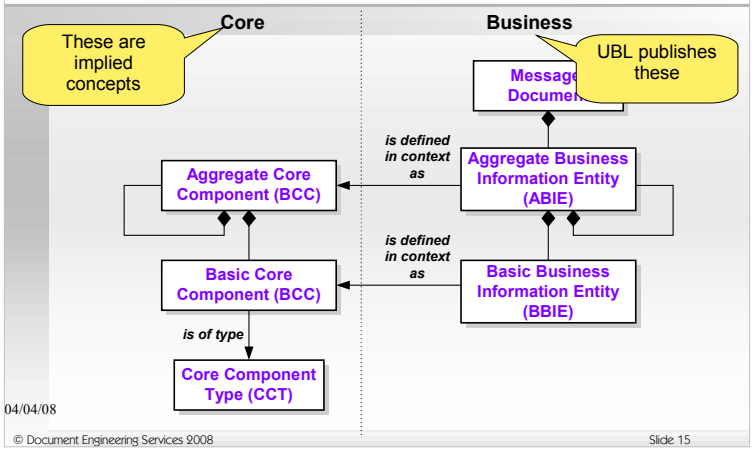
ebXML CCTS

- ebXML Core Components Technical Specification:
  - ISO 15000-5.
  - UN/CEFACT
- How to describe core components of business documents in a syntax independent manner.
- UBL is the first set of integrated XML document specifications based on CCTS.
  - UN/CEFACT recognizes UBL as appropriate first-generation XML documents for eBusiness.

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

CCTS Principles



© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

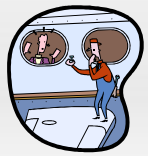
Business Information Entities

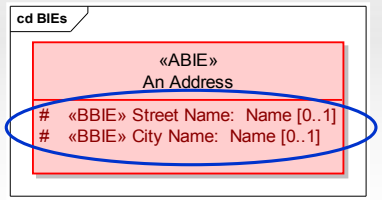
- Reusable building blocks for the exchange of information ("Core Components") in a specific Business Context.
- Aggregate BIE
  - A collection of related pieces of business information that together convey a distinct business meaning.
- Association BIE
  - Is associated to an Aggregate Business Information Entity, which describes its structure.
- Basic BIE
  - A singular business characteristic of a specific aggregation.

© Document Engineering Services 2008

Basic
Business Information Entities

- A singular business characteristic in a specific business context.
- Individual pieces of information.



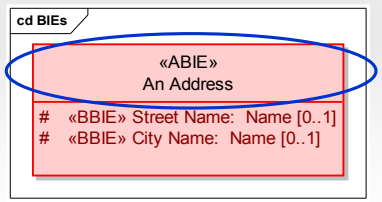


© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Aggregate Business Information Entities

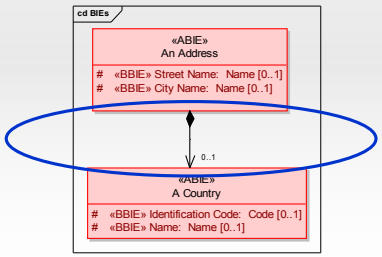
- A collection of related pieces of business information.
- A container of BBIEs and ASBIEs to other ABIEs.



© Document Engineering Services 2008

Association
Business Information Entities

- Represents an ABIE in a specific business context.
- An ABIE contained within another ABIE.



© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

UBL Design

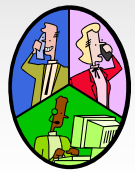
Naming Information Entities

© Document Engineering Services 2008

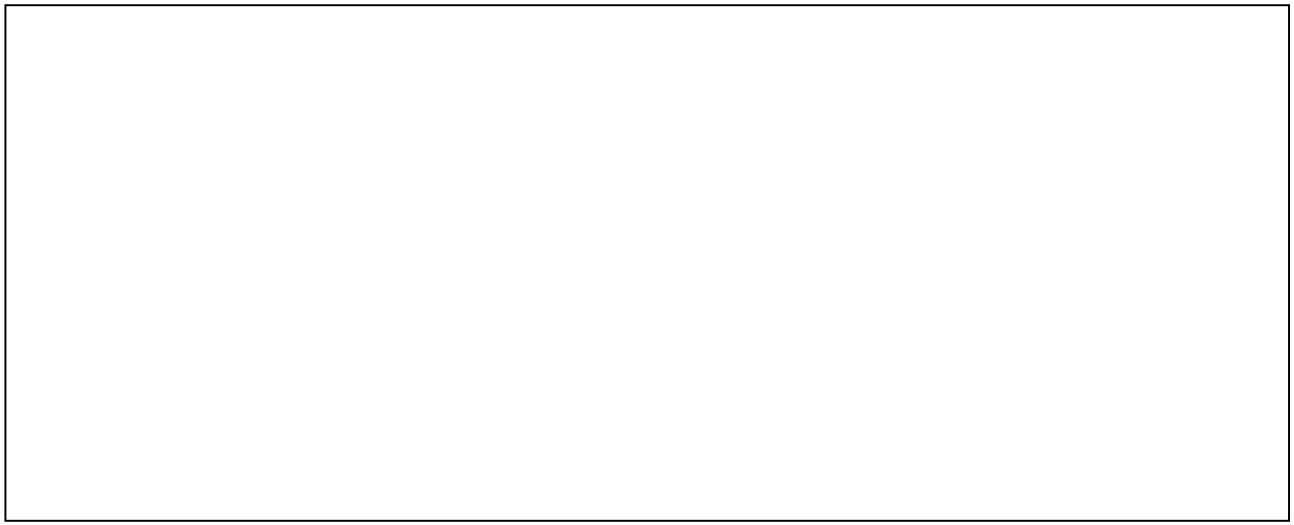
DOCUMENT ENGINEERING SERVICES
supporting universal business

Names

- Meaningful Information Entity names.
  - promote a common understanding.
  - encourage reuse.
- May need different names in different modeling artifacts.
  - XML Tag may not be the modeling or the presentation name.



© Document Engineering Services 2008



@ Port Community Systems 2007

DOCUMENT ENGINEERING SERVICES
supporting universal business

Rules for Consistent Naming

- Names should be grammatically consistent.
  - Singular not plurals
- Names should not be ambiguous.
  - Is "Supplier's ID" the same as "ID of Supplier"?
- Check names are not semantically misleading.
  - "Product Number" may not be a number, so "Product Identifier" is better.
- Qualify names to clarify the context of use.
  - "Customer Party" is more meaningful than "Customer".

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

ebXML Naming Rules

- ebXML Core Components naming rules are an application of ISO 11179.
- All Information Entities get a Dictionary Entry Name (DEN).
- Dictionary Entry Names are globally unique.
- They are the 'key' or unique identifier for any Information Entity.

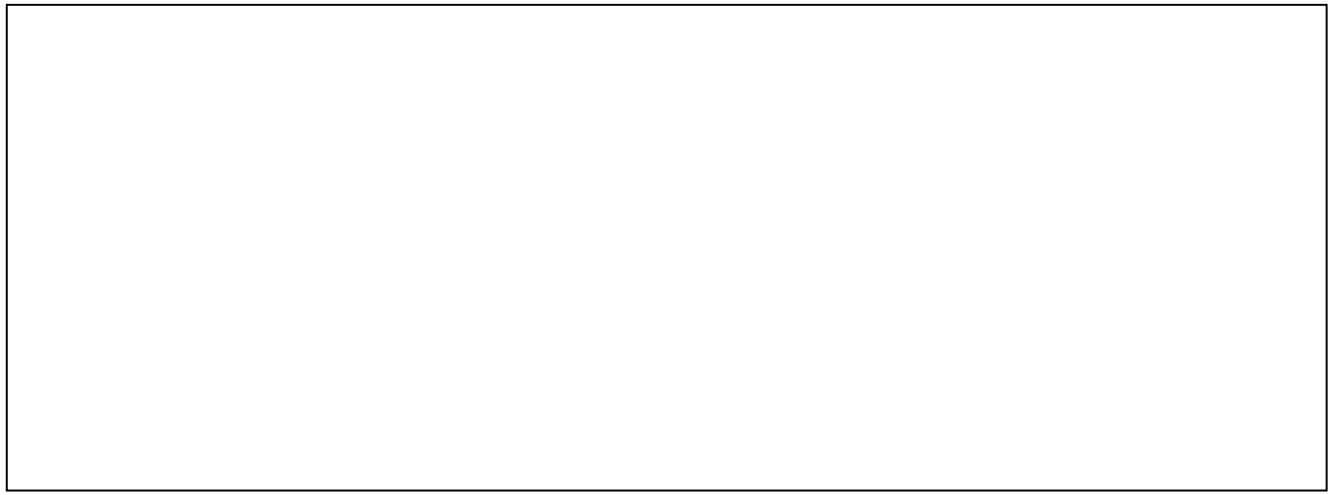
© Document Engineering Services 2008

Dictionary Entry Names

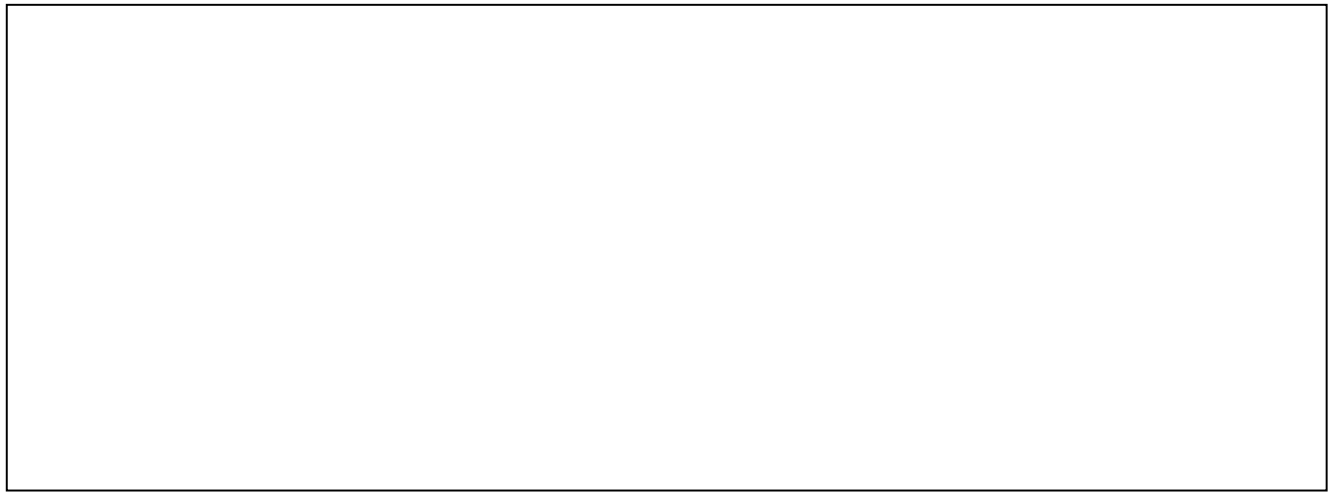
{qualifier} ***Object Class Term***, ". ",
{qualifier} ***Property Term***, ". ",
{qualifier} ***Representation Term***

- Object Class Term
  - The object class to which it belongs.
- Property Term
  - Reflecting its function as a property or distinguishing characteristic of the object class.
- Representation Term
  - How it is represented.

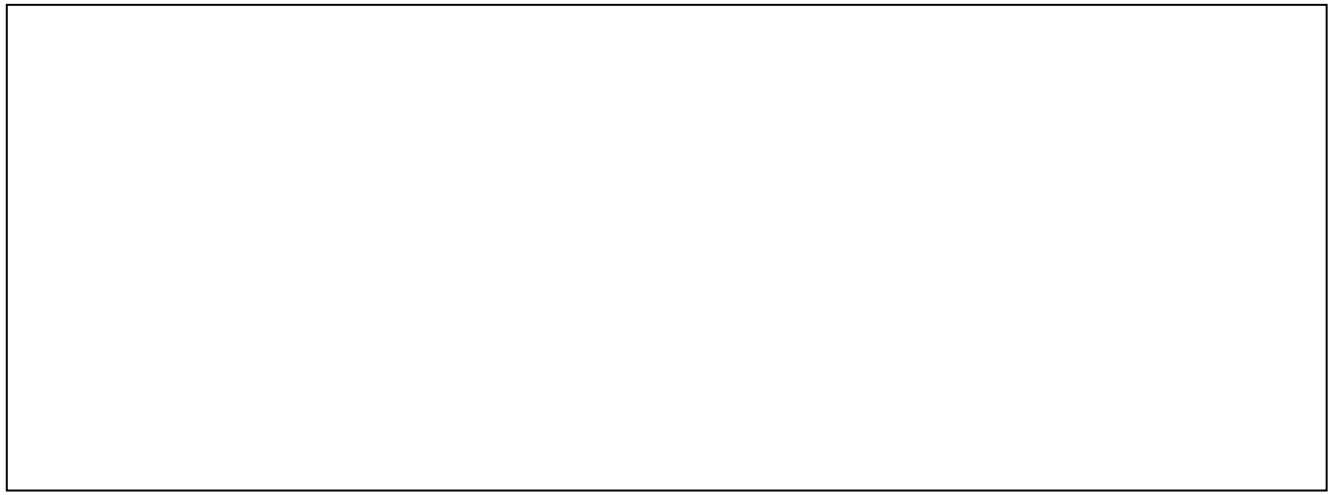
© Document Engineering Services 2008



@ Port Community Systems 2007



@ Port Community Systems 2007



@ Port Community Systems 2007

DOCUMENT ENGINEERING SERVICES
supporting universal business

UBL Design
Representation Terms

© Document Engineering Services 2008

Amount

Content

A number of monetary units specified in a currency where the unit of currency is explicit or implied.

Amount. Currency. Identifier

The currency of the amount.

Amount. Currency. Code List Version Identifier

The version of the UN/ECE rec. 9 code list.

```
<cbc:TaxAmount currencyID="DKK">1262.50</cbc:TaxAmount>
```

© Document Engineering Services 2008

Binary Object

(Graphic, Picture, Sound, Video)

Binary Object. Content	A set of finite-length sequences of binary octets.
Binary Object. Format. Text	The format of the binary content.
Binary Object. Mime. Code	The mime type of the binary object.
Binary Object. Encoding. Code	Specifies the decoding algorithm of the binary object.
Binary Object. Character Set. Code	The character set of the binary object if the mime type is text.
Binary Object. Uniform Resource. Identifier	The Uniform Resource Identifier that identifies where the binary object is located.
Binary Object. Filename. Text	The filename of the binary object.

```
<cbc:EmbeddedDocumentBinaryObject mimeCode="application/pdf"
encodingCode="Base64" characterSetCode="UTF-8">
brIG0Ng53KHKaoOQNQYcpyFHKa9UhkgzjkPBTaFJDIBUasfdgartujjuoprr
wer34578deewcwetyuuutyur643432THyB4KY5SYM5BZcpMkshU0UzJoL
HSDEWFdreefssd///
</cbc:EmbeddedDocumentBinaryObject>
```

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Code

Code. Content	A character string (letters, figures or symbols) that for brevity and/or language independence may be used to represent or replace a definitive value or text of an attribute.
Code List. Identifier	The identification of a list of codes.
Code List. Agency. Identifier	An agency that maintains one or more code lists.
Code List. Agency Name. Text	The name of the agency that maintains the code list.

```
<cbc:ItemClassificationCode listID="UNSPSC"  listVersionID="7.0401">
43191501
</cbc:ItemClassificationCode>
```

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Code (contd.)

Code List. Name. Text	The name of a list of codes.
Code List. Version. Identifier	The version of the code list.
Code. Name. Text	The textual equivalent of the code content.
Language. Identifier	The identifier of the language used in the corresponding text string.
Code List. Uniform Resource. Identifier	The Uniform Resource Identifier that identifies where the code list is located.
Code List Scheme. Uniform Resource. Identifier	The Uniform Resource identifier that identifies where the code list scheme is located.

```
<cbc:ItemClassificationCode listID="UNSPSC"  listVersionID="7.0401">
43191501
</cbc:ItemClassificationCode>
```

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Date Time
(Date, Time)

Date Time. Content	The particular point in the progression of time.
Date Time. Format. Text	The format of the date/time content.

<cbc:StartDate>2007-01-25</cbc:StartDate>

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Identifier

Identifier. Content	A character string used to identify and distinguish uniquely, one instance of an object in an identification scheme from all other objects within the same scheme.
Identification Scheme. Identifier	The identification of the identification scheme.
Identification Scheme. Name. Text	The name of the identification scheme.

```
<cbc:ID schemeID="GTIN">5712345780121</cbc:ID>
```

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Identifier (contd.)

Identification Scheme Agency. Identifier	The identification of the agency that maintains the identification scheme.
Identification Scheme. Agency Name. Text	The name of the agency that maintains the identification scheme.
Identification Scheme. Version. Identifier	The version of the identification scheme.
Identification Scheme Data. Uniform Resource. Identifier	The Uniform Resource Identifier that identifies where the identification scheme data is located.
Identification Scheme. Uniform Resource. Identifier	The Uniform Resource Identifier that identifies where the identification scheme is located.

Indicator

Indicator. Content	The value of the indicator.
Indicator. Format. Text	Whether the indicator is numeric, textual or binary.

<cbc:ChargeIndicator>false</cbc:ChargeIndicator>

© Document Engineering Services 2008

Measure

Measure. Content	The numeric value determined by measuring an object.
Measure Unit. Code	The type of unit of measure.
Measure Unit. Code List Version. Identifier	The version of the measure unit code list.

```
<cbc:GrossWeightMeasure unitCode="KGS">
21408.79
</cbc:GrossWeightMeasure>
```

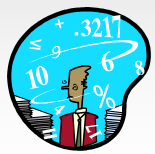
© Document Engineering Services 2008

Numeric
(Value, Rate, Percent)

Numeric. Content — Numeric information that is assigned or is determined by calculation, counting or sequencing.

Numeric. Format: Text — Whether the number is an integer, decimal, real number or percentage.

<cbc:PackSizeNumeric> | </cbc:PackSizeNumeric>



© Document Engineering Services 2008

Quantity

Quantity. Content	A counted number of non-monetary units possibly including fractions.
Quantity. Unit. Code	The unit of the quantity.
Quantity Unit. Code List. Identifier	The quantity unit code list.
Quantity Unit. Code List Agency. Identifier	The identification of the agency that maintains the quantity unit code list.
Quantity Unit. Code List Agency Name. Text	The name of the agency which maintains the quantity unit code list.

<cbc:PackQuantity unitCode="EA">1</cbc:PackQuantity>

© Document Engineering Services 2008

Text
(Name)

Text. Content	A character string (I.e. a finite set of characters) generally in the form of words of a language.
Language. Identifier	The identifier of the language used in the corresponding text string.
Language. Locale. Identifier	The identification of the locale of the language.

<cbc:Description>Nokia Mobile telephone –Type ABC</cbc:Description>

Note: The Representation term "Text" is not part of the tag name (UBL naming rule)

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

UBL Design
Codes and Identifiers

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Sets of Codes

- A special set of possible values.
- The code value refers to something else
- May be symbolic representations:
  - Initialisms ("EU"),
  - Acronyms ("SARS"),
  - Apocopations ("Reefer") or
  - Arbitrary abbreviations ("XML").
- But maybe not…
  - Both ISO code "CNY" and "156" identify the Chinese Yuan Renminbi.

© Document Engineering Services 2008

Codes and Values

DOCUMENT ENGINEERING SERVICES
supporting universal business

AV	Capsule
BA	Barrel
BB	Bobbin
BC	Bottlecrate / bottlerack
BF	Balloon, non-protected
BG	Bag
BH	Bunch
BI	Bin
BJ	Bucket
BK	Basket

Refers to

Types of packaging codes

KMT	kilometre
C45	nanometre
C52	picometre
A71	femtometre
A45	decametre
NMI	nautical mile
A11	
A12	
C63	parsec
AK	fathom
X1	chain
INH	inch
M7	micro-inch
FOT	foot
YRD	yard

Refers to

Units of Measurement codes

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Identifiers

- Have meaning in their own right.
  - Don't have to reference anything.
- May be a constrained set of values:
  - Set of Driver's License numbers issued.
- May be symbolic representations (like codes).
- Or (effectively) not…
  - The values for identifying a phone number.
- We use codes as identifiers:
  - Digits in phone numbers are codes for regions.
- But not identifiers as codes:
  - What does your Driver's License number refer to?

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

What are these?

ABCUIJKL123456XYZ
ABCU123456789000
ABCU123456789001

Waybill Numbers

And these?

Shipping Container Numbers

OCLU1234567
NHKL7654321
HPGL8877665

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

UBL Specification

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Specifying Documents

- We need a range of metadata to support the explanation of meanings.
  - Names are not enough.
- We define these in models.

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

UBL Specification

Models and Modeling

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

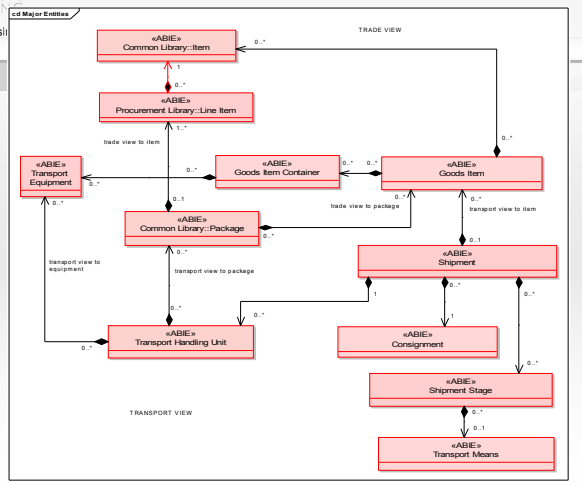
Component Models

- A generalized, conceptual model.
  - "webs" or networks.
  - Domain Model, Enterprise Data Model.
- Not a model of one document…

  but,
- A model of Information Entities usable in all documents.
  - and databases, etc.
- Complexity depends on the number of the business rules.

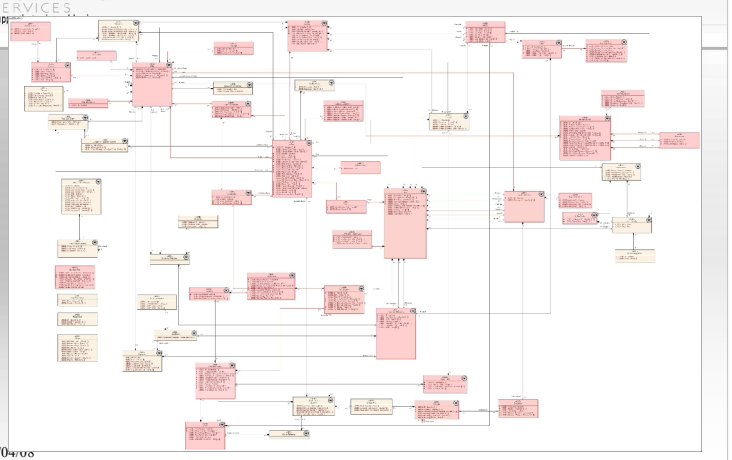
© Document Engineering Services 2008

A Component Model



© Document Engineering Services 2008

UBL's Common Components



© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

UBL Specification

Assembling Components into Documents

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

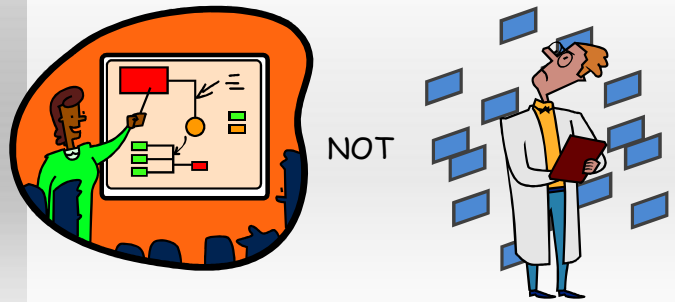
Documents are Hierarchical

- Inverse tree structure.
- Can be represented in a 2D space.
  - That's why markup languages need just <start> and </end> tags.
- Nesting imposes meaning.
  - When "Contact" contains "Name" we know that the name is that of the contact.
  - Parent (genealogy) gives contact of use.

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

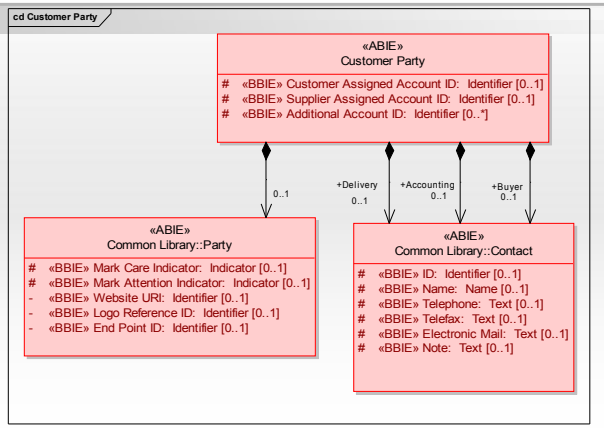
Hierarchy not Network



04/04/08

© Document Engineering Services 2008

A Document Assembly Model



© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Another Document Assembly Model

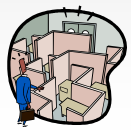
UBL Name	Dictionary Entry Name	Cardinality	Definition
CustomerParty	Customer Party. Details		Information about the Customer Party.
CustomerAssignedAccountID	Customer Party. Customer Assigned_ Account Identifier. Identifier	0..1	An identifier for the Customer's account, assigned by the Customer itself.
SupplierAssignedAccountID	Customer Party. Supplier Assigned_ Account Identifier. Identifier	0..1	An identifier for the Customer's account, assigned by the Supplier.
AdditionalAccountID	Customer Party. Additional_ Account Identifier. Identifier	0..n	An identifier for the Customer's account, assigned by a third party.
Party	Customer Party. Party	0..1	An association to Party.
DeliveryContact	Customer Party. Delivery_ Contact. Contact	0..1	An association to Delivery Contact.
AccountingContact	Customer Party. Accounting_ Contact. Contact	0..1	An association to Accounting Contact (Customer).
BuyerContact	Customer Party. Buyer_ Contact. Contact	0..1	An association to Buyer Contact.

© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Creating a Document Assembly

- Each type of document usually requires its own document assembly model.
- We create a document assembly model by:
  1. Choosing an "entry point" into the Document Component Model.
     - The entry point is a structure that will form the root of the document tree.
  2. Following a pathway through the component model
     - guided by business rules for the document required.

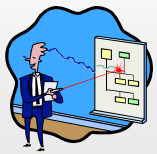


© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Assembling Associations

- If an association is mandatory:
  - must follow the path
  - the associated ABIE must be in the document.
- If an association is optional:
  - may follow the path
  - the associated ABIE can be in the document if required by business rules.
- Cardinality of associations controls the depth of the document hierarchy.



© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

UBL Specification
Assembly Patterns

© Document Engineering Services 2008

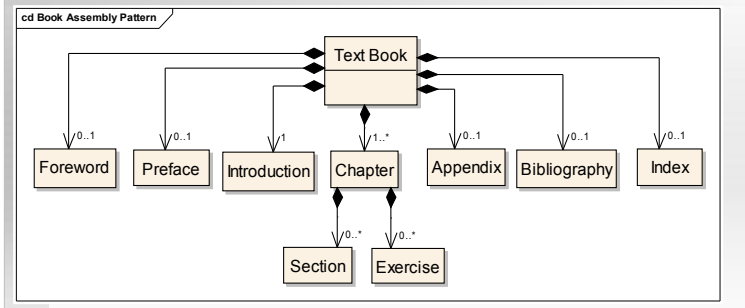
DOCUMENT ENGINEERING SERVICES
supporting universal business

Patterns for Document Assembly

- Document assembly models express business rules based on the context of use.
- Many assemblies share common structures.
  - There are patterns of assemblies.
  - a Book, a Contract, an Order.
- Good designers employ these patterns…
  - Because processes will find their documents familiar.

© Document Engineering Services 2008

The “Book” Assembly Pattern



© Document Engineering Services 2008

DOCUMENT ENGINEERING SERVICES
supporting universal business

Reusing Assembly Patterns

- Assembly patterns encourages the use of common libraries of components.
  - Reusing patterns from elsewhere in the model
  - Adopting patterns from outside
- The challenge is customizing these patterns to suit specific requirements.

© Document Engineering Services 2008