

CCTS – Semantic Data Modeling Within and Across The Firewall

Mark Crawford
Standards Architect
SAP Labs mark.crawford@sap.com

Gunther Stuhec
Standards Architect
SAP AG gunther.stuhec@sap.com

The Data Interoperability Dilemma

ISO 15000-5 Core Components

Process Models, CCTS, and Syntax

Concepts In Action

CCTS and The Semantic Web

CCTS in SAP

The Data Interoperability Dilemma

Users

Higher Education
& Research



- Integrate with multiple processes and industries



Mining

Healthcare



- Do not want to learn, deploy and maintain different integration approaches



Pharma

Financial
Service Provider



Industry Communities

- Want complete, interoperable specifications
- Want to avoid endless reinvention



Oil & Gas



Insurance

High
Tech



Internal Applications

- Support data across multiple business functional areas
- Support interfaces with different databases and systems



Aerospace
& Defense

Telco



Media

Chemicals



Application Providers

- Support every industry and process
- Want easier, cheaper interoperability approaches to reduce TCO



Consumer
Products

Automotive



Mill Products



Retail

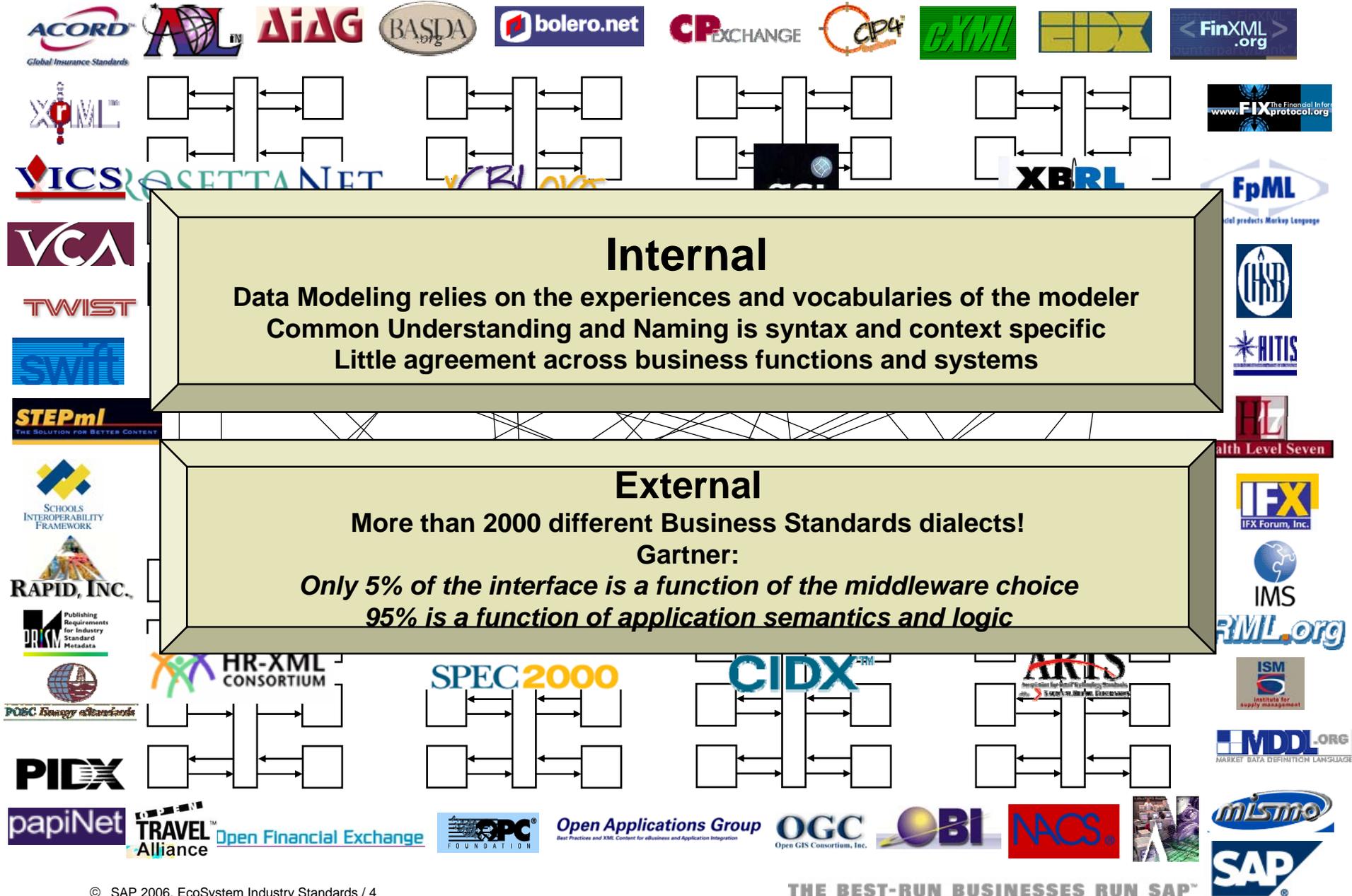
Engineering &
Construction



Banking



How Serious is the Problem?

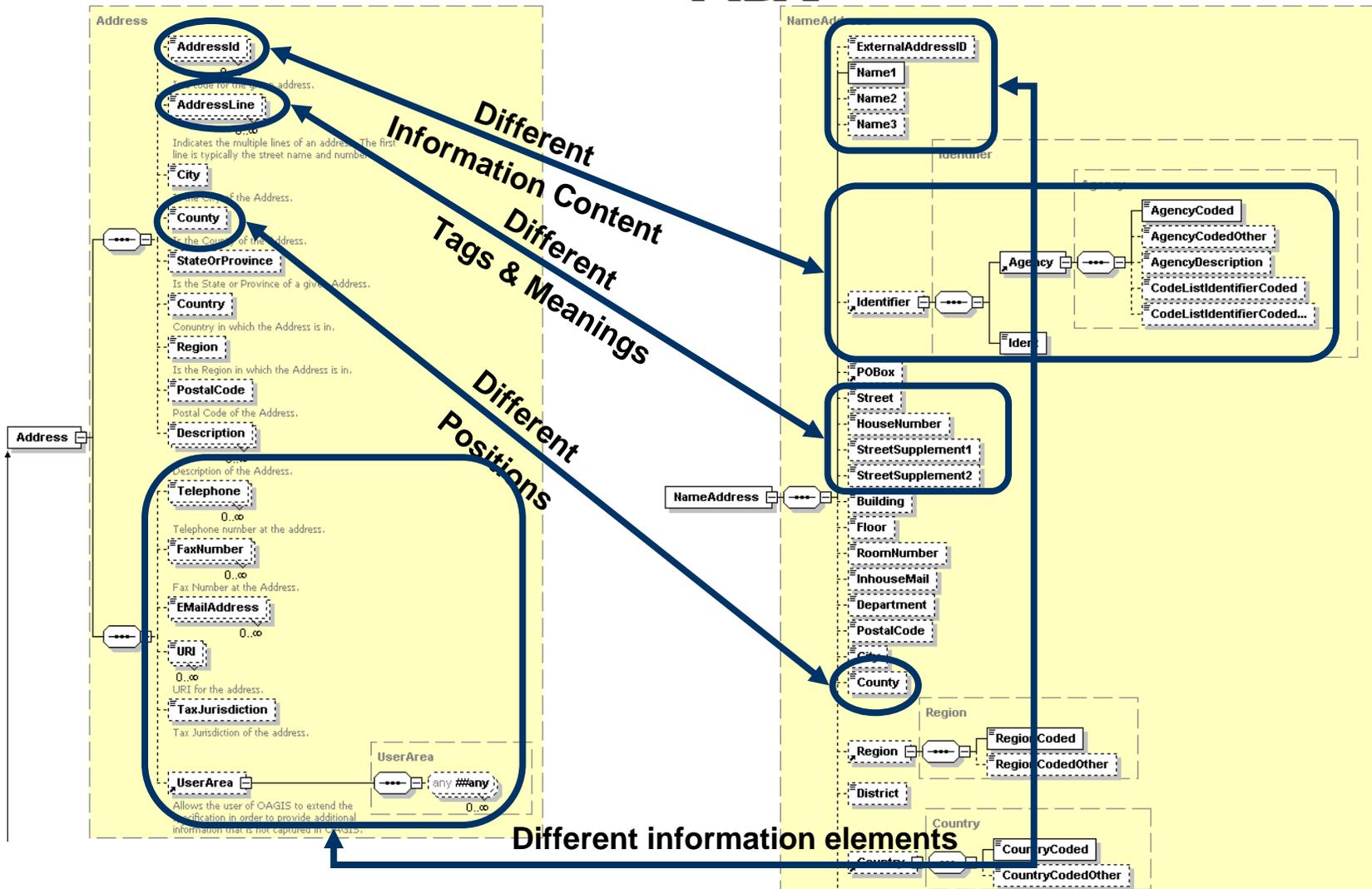


Example Please!

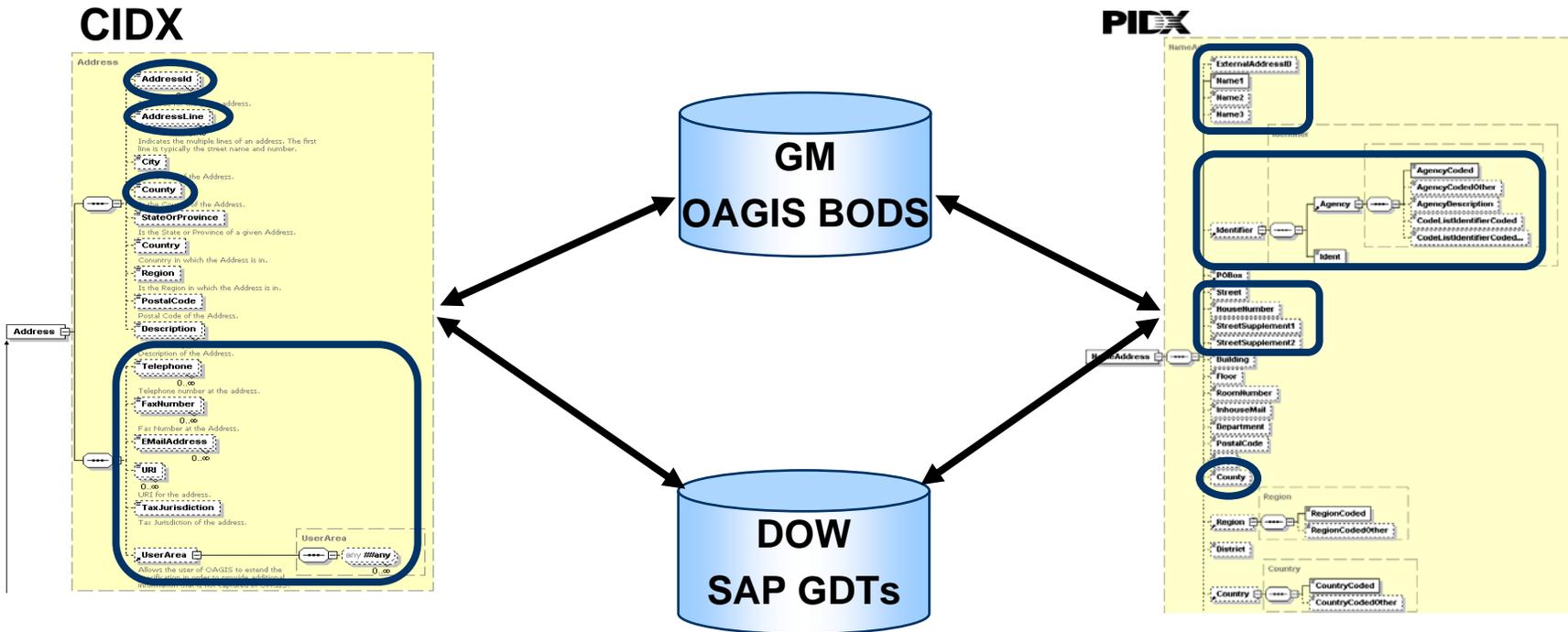
Barriers to Semantic Interoperability

CIDX

PIDX



Impact of Barriers to Semantic Interoperability



Different Information Content + Different information Elements + Different Tags & Meanings = Increased \$ TCO \$

The Data Interoperability Dilemma

ISO 15000-5 Core Components

Process Models, CCTS, and Syntax

Concepts In Action

CCTS and The Semantic Web

CCTS in SAP

Semantic Data Modeling Solution – ISO 15000-5 Core Components

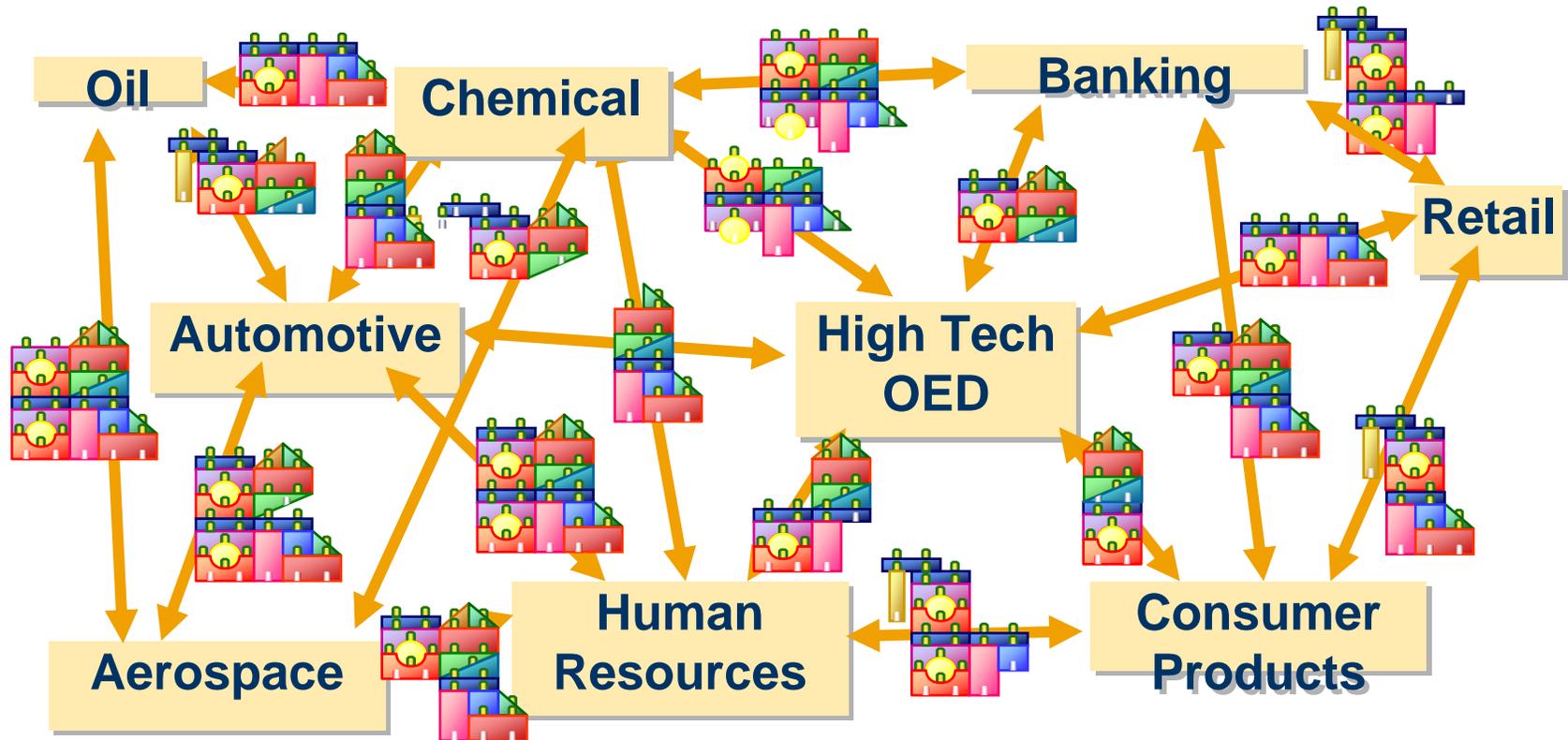
■ What:

- **A methodology for developing semantic data models**
- **The integration mechanism for coordination between Architectures, Process Models, Data Models, and Syntax Expressions**
- **A way to identify, capture and maximize the re-use of business information to support and enhance information interoperability across multiple business situations**

■ How:

- **Syntax and Context Neutral conceptual models**
- **Context specific physical/logical models**
- **Transformation to syntax specific information exchanges**

Semantic Data Modeling Solution - ISO 15000-5 Core Components



Integration mechanism for Process Models, Databases, Applications, Services, and Transactions

A way to identify, capture and maximize the re-use of business information to support and enhance information interoperability across multiple business situations

Semantics at the Data Layer

- Common Naming Convention and Structuring
- Syntax and Context Neutral Conceptual Data Models
- Context Specific Physical/ Logical Data Models
- Consistency in design and use all the syntax layer

SAP GDTs are based on ISO 15000-5 CCTS

Who is working with Core Components

Industry independent standards groups

- ISO TC154, the current group responsible for EDIFACT and UNTDED – complete alignment to Core Components
- OAGi (Open Applications Group) – OAGIS Version 9.0 and later
- OASIS UBL (Universal Business Language)

Industry dependent standards groups

- ACORD for Insurance Data - evaluating
- AIA (Aerospace Industry Association)
- AIAG, ODETTE and JAMA/JAPIA – for all current and future projects in the international automotive sector.
- BoostAero for its Supply Chain Project in the Aerospace and Defense Industry
- CIDX in the chemical industry for its Material Safety Project
- CIDX (chemical), PIDX (oil) and RAPID (agriculture) – evaluating a move to core component compliance
- EUDIN for European Waste Movement Project
- Eurofer for Messages in European Steel Industry
- HL7 for future Healthcare Data
- IATA (International Air Transport Association)
- GS.1 for Supply Chain Messages and Cataloging
- OTA (Open Travel Alliance) - evaluating for Business Data in Travel-Industry
- SWIFT for Messages in Financial Business Processes
- SITPRO and EUROPRO for Sanitary & Phytosanitary Certificates
- XBRL - evaluating for future Accounting & Audit data and collaboration with UN/CEFACT

Governments groups

- Department of Navy – DONXML V2.0
- DCMA (Defense Contract Management Agency)
- NDIA (National Defense Industrial Association)
- US Federal CIO Council

International groups

- eBes – (eBusiness Board for European Standardization) - for its Invoice Processing and Remittance Advice
- GJXDM (Global Justice XML Data Model)
- WCO (World Customs Organization)

National Governments

- Australian Government
- Canadian Government
- Danish Government
- German Government – formal announcement identifying CCTS as future data standard for domestic affairs
- Hong Kong Government
- Korean Government
- Sweden and Norway are evaluating Core components

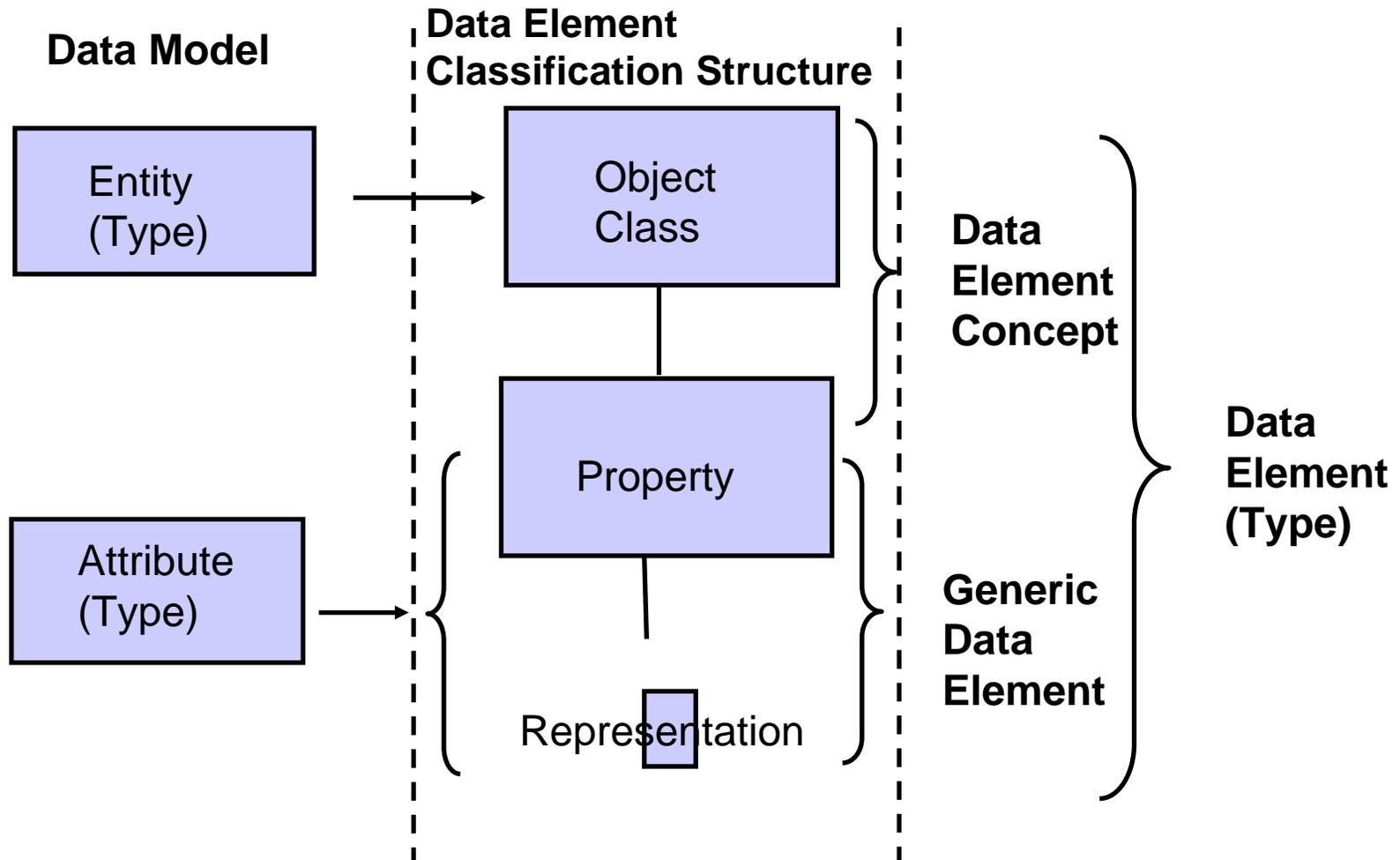
Software Vendors

- SAP
- Oracle (through OAGIS)

ISO 11179 –

- **Describes the standardizing and registering of data elements to make data understandable and shareable**
- **Provides concrete guidance on the formulation and maintenance of discrete data element descriptions and semantic content – Identifiers, Definitions, & Classification Categories**
- **Used to formulate data elements in a consistent, standard manner**
- **Provides guidance for establishing a data element registry**

ISO 11179 Data Element Constructs



Semantic rules – enable meaning to be conveyed

- Example: Components consist of discrete terms:
 - ◆ Object class terms, property terms, representation terms, qualifier terms

Syntactic rules – relate items in a consistent, specified order

- Example: a rule might require the property term is always the second component in the name

Lexical rules – (word form and vocabulary) rules reduce redundancy and increase precision

- Example: Nouns are used in the singular only

Uniqueness rule – ensures names are unique within a context.

- Example: Homonyms shall not occur in a namespace
 - ◆ bank (embankment)
 - ◆ bank (place where money is kept)

11179

Object class

Property 1: representation 1
Property 2: representation 2
Property 3: representation 3
Property 4: representation 4

15000-5

Address. Details

Address. Street: text
Address. Post code: text
Address. Town: text
Address. Country: identifier

This is basic OO Good Stuff!

- **Enables interoperability among different industries and applications**
- **Holds related information together and avoids semantic fragmentation**
- **Facilitates multilingual support**
- **Syntax neutral**
 - ◆ **Models can be readily expressed in XML Schema, UML diagrams, JAVA classes, SQL based relational data bases, etc.**
- **Guarantees semantic understanding in any technical implementation and interface**
- **Does not require complicated and expensive mappings between interfaces.**
- **Significantly reduces data total cost of ownership**

**United Nations Centre for Trade Facilitation and Electronic
Business**

**International Organization for Standardization – Technical
Committee 154 – Processes, Data Elements & Documents in
Commerce, Industry & Administration**

**ISO\DTS 15000-5: 2006 Core Components
Technical Specification 2nd Edition**

UN/CEFACT Version 2.2

Working Draft

**Draft B
31 March 2006**

ISO15000-5 Overview

Creates common re-usable building blocks

- Template Data Constructs – Core Components – e.g. Party, Address, Organization
- Reusable logical/physical Data Constructs – called BIEs – e.g. Buyer_Party, Seller_Party
- Core Data Types, i.e. for “Amount, Code, Measure, Name, and Quantity”
- Business Messages, e.g. Order, Invoice, Financial Reporting

Based on Semantic Definitions

- Clear rules on how to define semantics to explain what items mean

Uses a Context Mechanism that controls how data constructs vary depending on the context

- e.g. By business process, business process role, industry, country/region, etc.

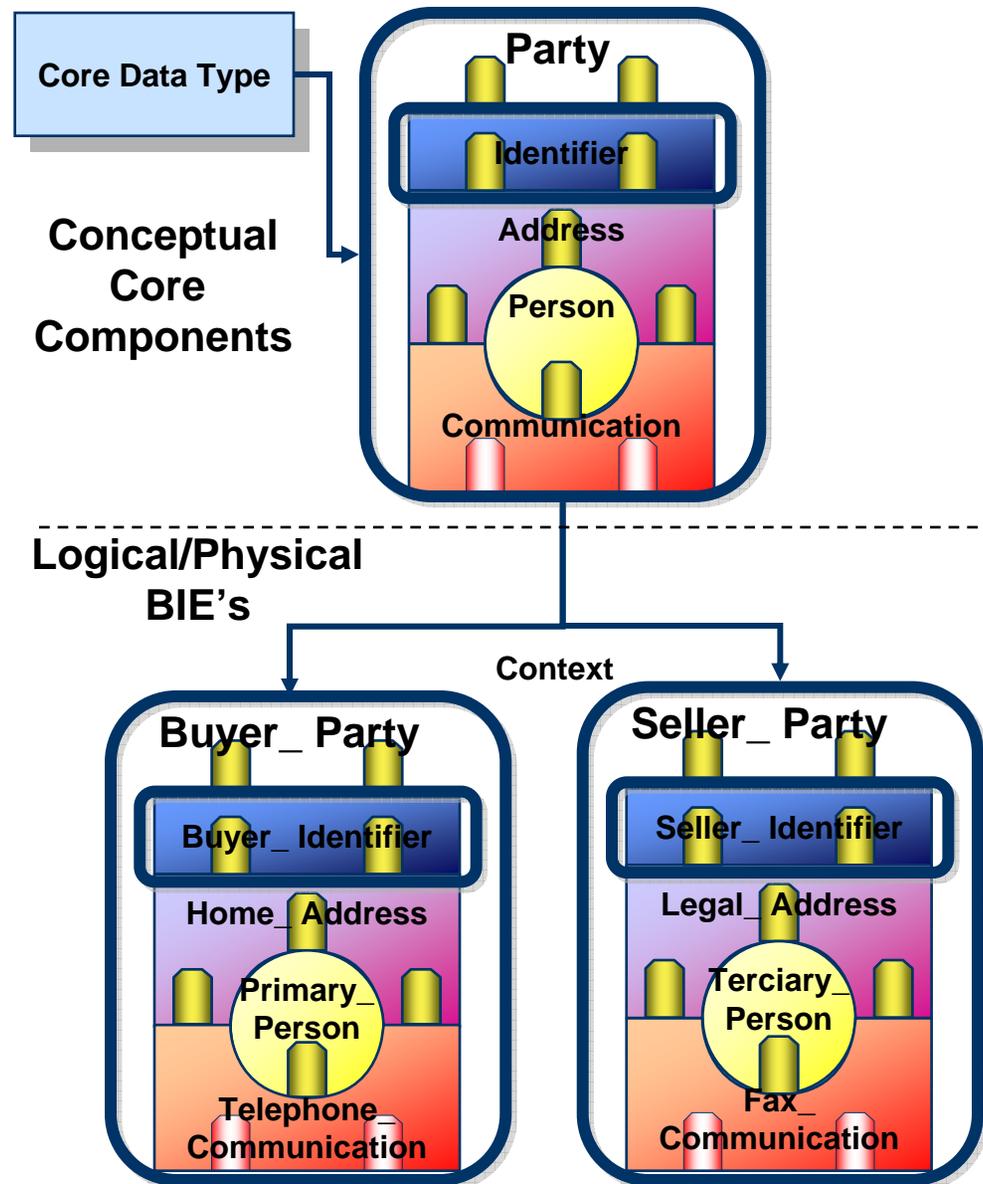
Syntax neutral

- Can be used to define business documents OR business objects/databases

Provides the heavy lifting for syntax specific representations

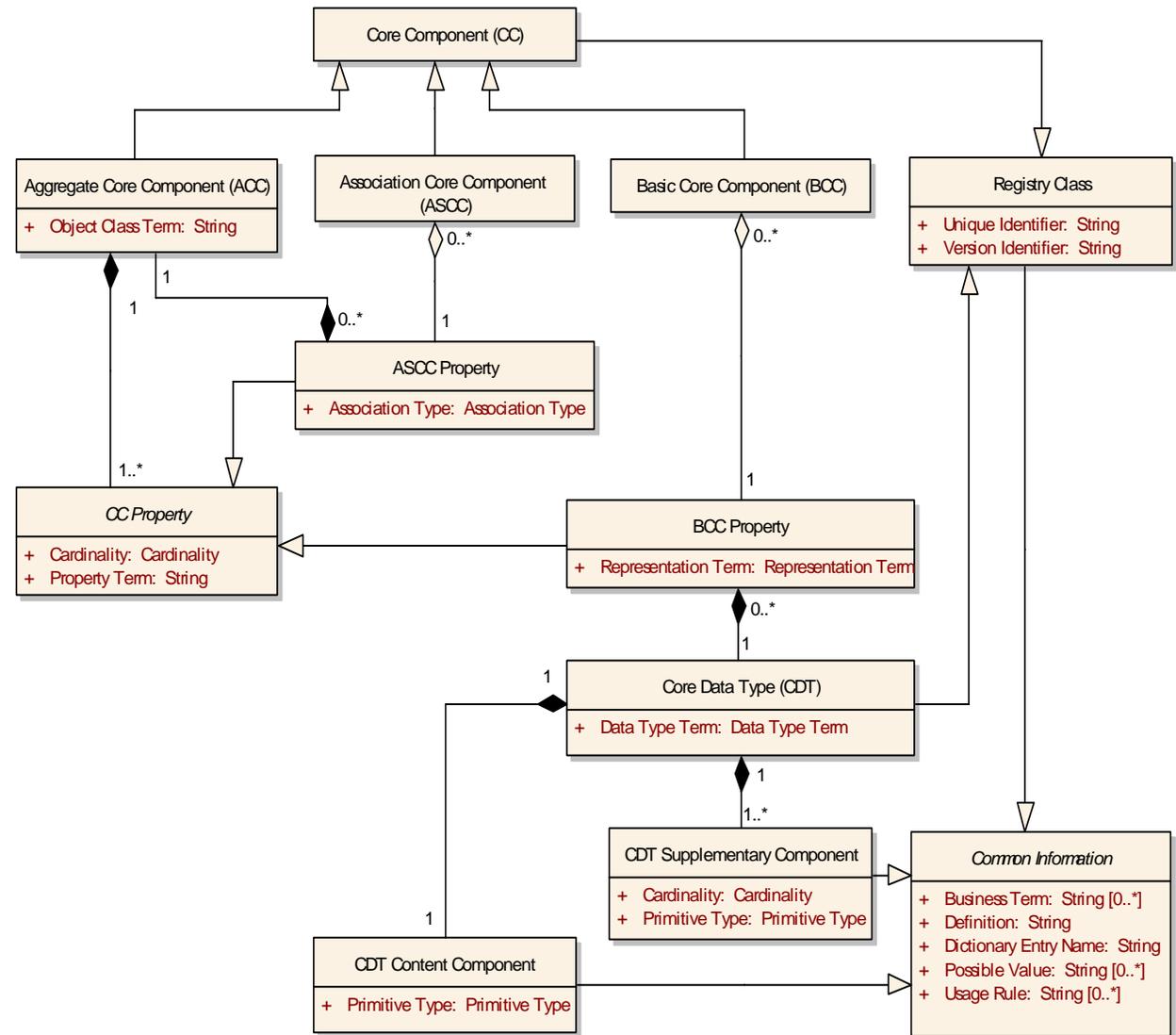
- XML Naming and Design Rules define how to map to XML Schema
- XMI Profile for future UML2XML
- UML2EDIFACT for traditional EDI

Definitions stored in a repository



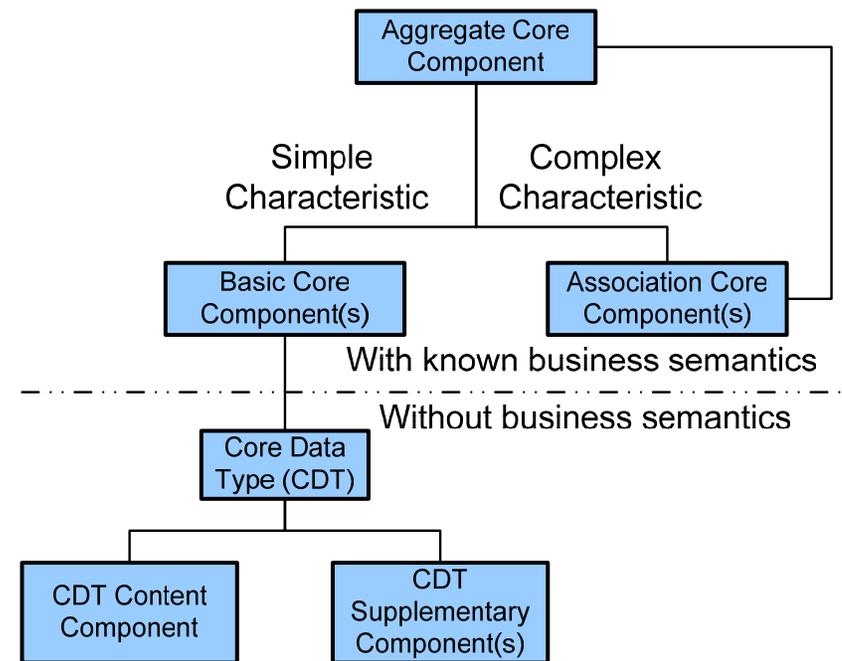
The Core Component Model

- Core Components are the centerpiece of the approach
- Conceptual in nature
- Used as reference only
- Never instantiated
- Intended to have single global reference model



Can You Simplify That Please?

- **Core Components are reusable building blocks that equate to simple and complex data elements**
- **Core Components consist of:**
 - **Basic Core Components**
 - **Aggregate Core Components**
 - **Association Core Components**
- **Basic Core Components are supported by Core Data Types that define their value space**



So What Does A Core Component Look Like

Financial Account. Details

Financial Account. Type. Code

Financial Account. Name

Financial Account. Currency. Code

Financial Account. Owner. Party

Financial Account. Servicer. Party

Financial Account. Information Recipient. Party

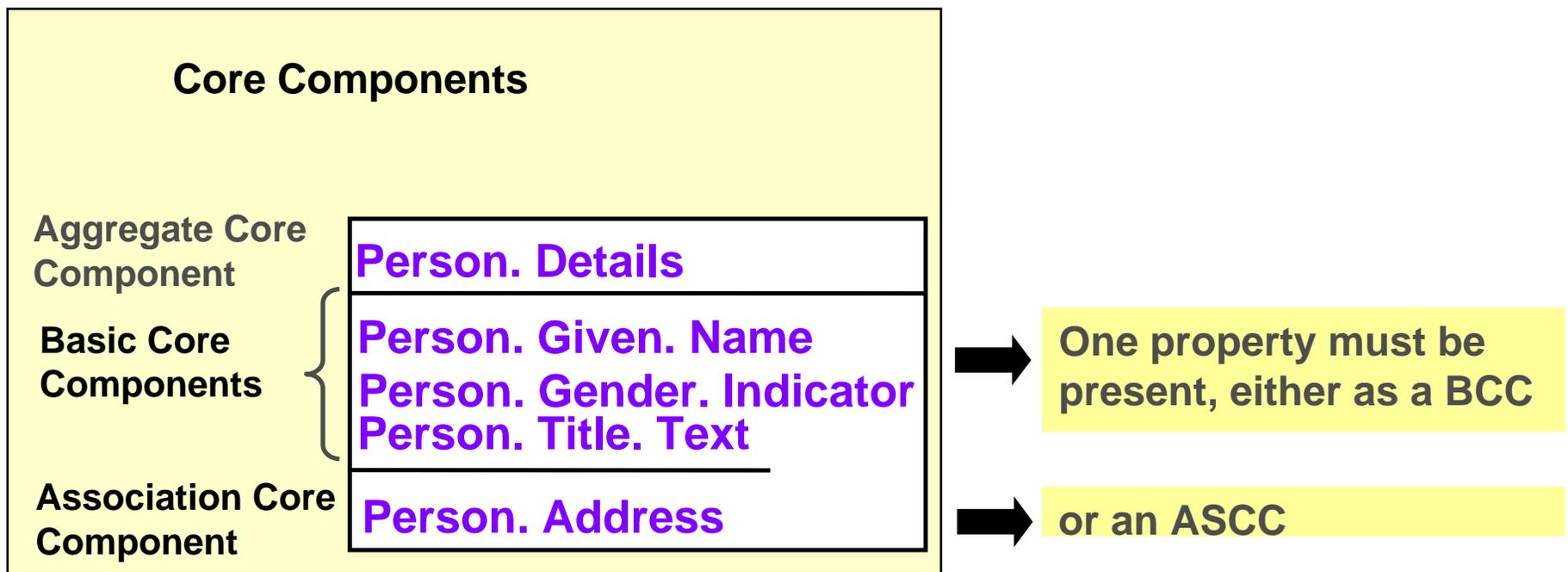
Financial Account. Agent. Party

Financial Account - A specific business arrangement whereby debits and/or credits arising from transactions are recorded, such as, a financial account with a bank, a financial account with a trading partner

Relate that to the Model Please - Aggregate Core Component (ACC)

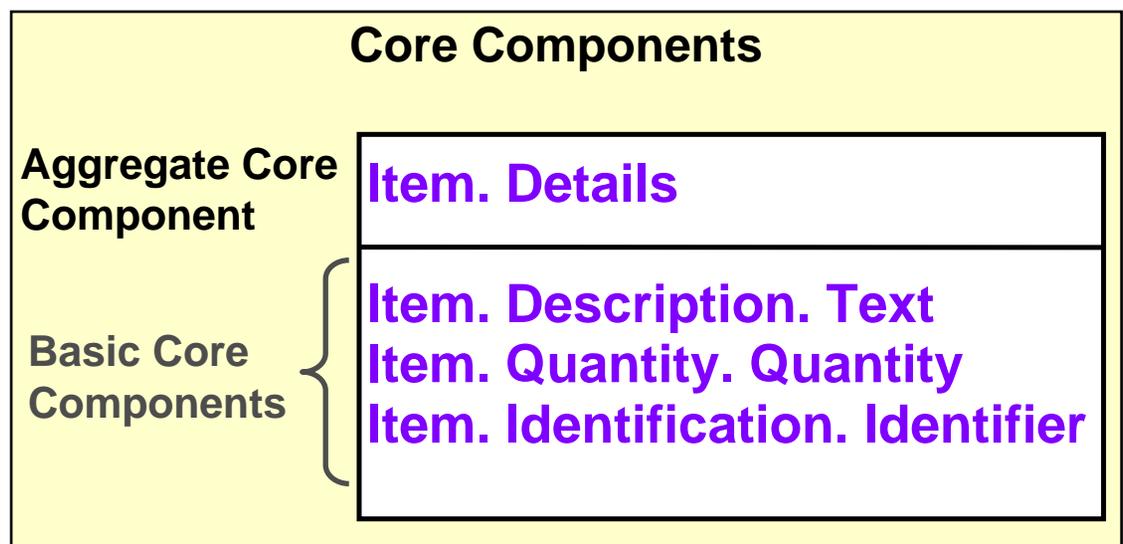
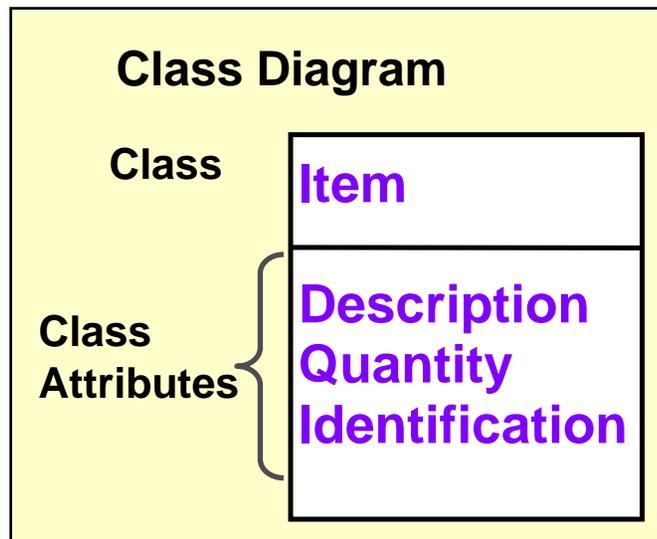
A collection of related pieces of business information that together convey a distinct business meaning, independent of any specific *Business Context*

Expressed in modelling terms, it is the representation of an *Object Class*, independent of any specific *Business Context*

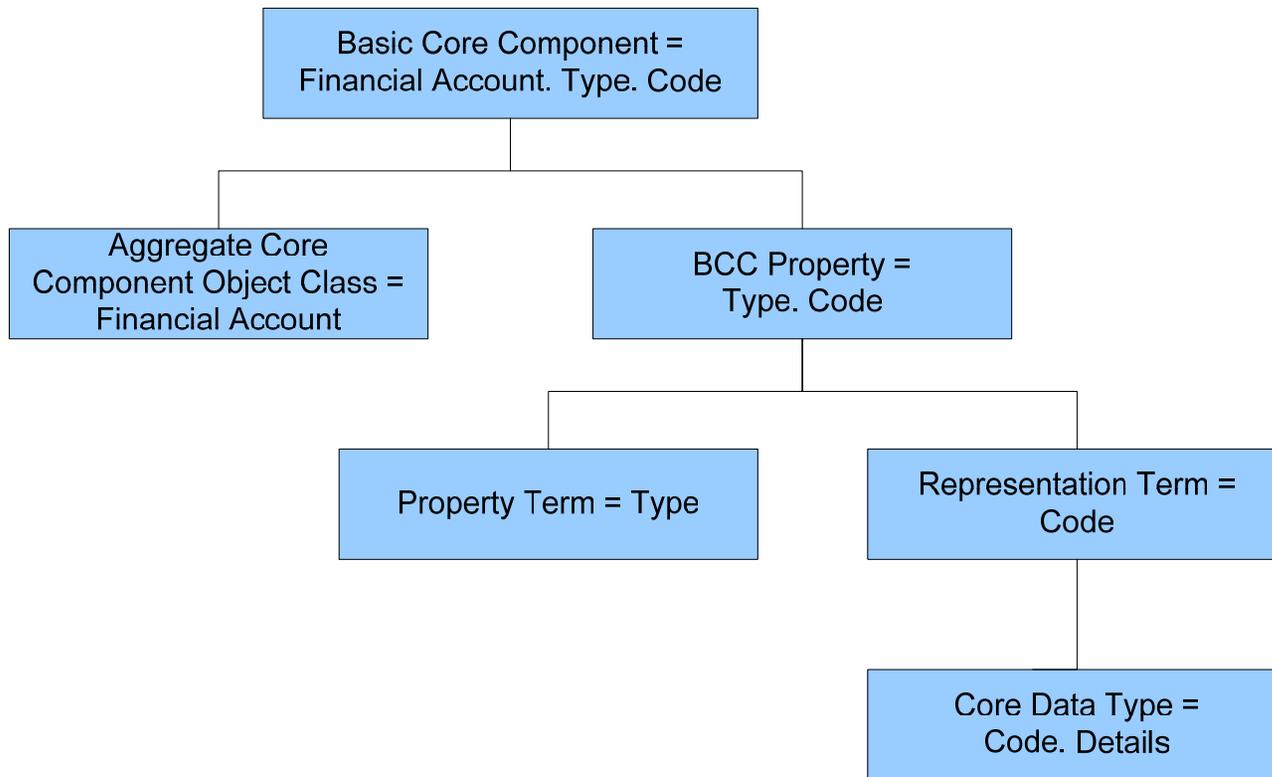


Basic Core Component (BCC)

- Simple properties
- Equivalent to attribute of a class
- Key feature is consistent Tri-partite Naming
 - Object:Property:Representation



Basic Core Component Parts



Entity

Financial Account

Attribute

**Type
Code**

Class

Financial Account

Property/Attribute

Type:code

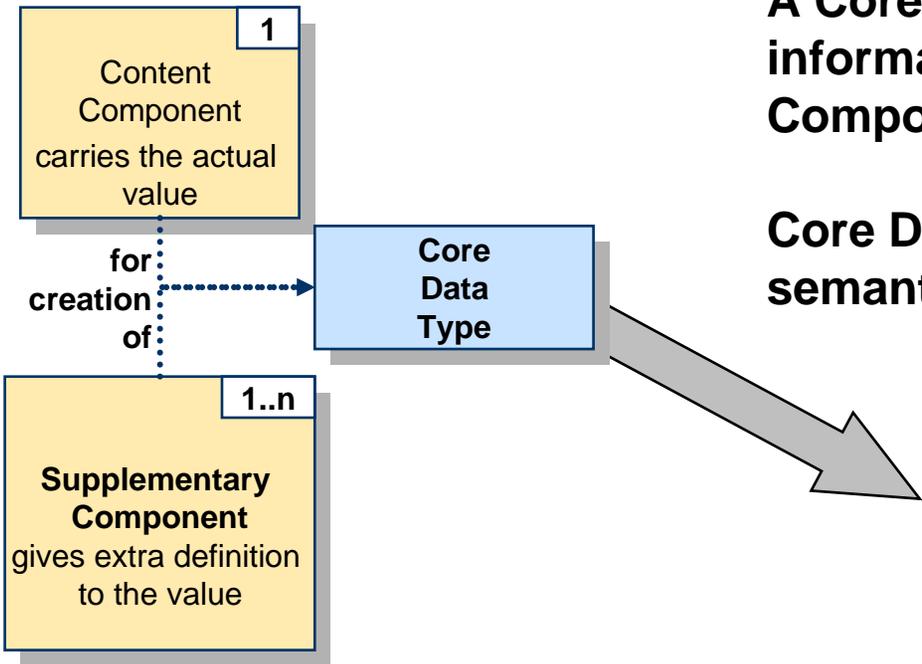
Core Component

Financial Account. Details

BCC

Financial Account. Type.

Core Data Type

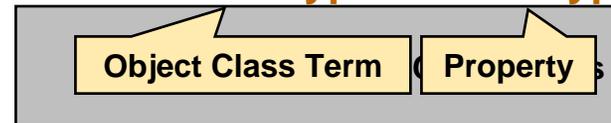


A **Core Data Type** signifies the “type” of information represented by a **Basic Core Component**.

Core Data Types do not contain business semantics – they are “semantic-neutral”

Dictionary Entry Name
Unqualified Data Types (DTs + Secondary Representation Terms):

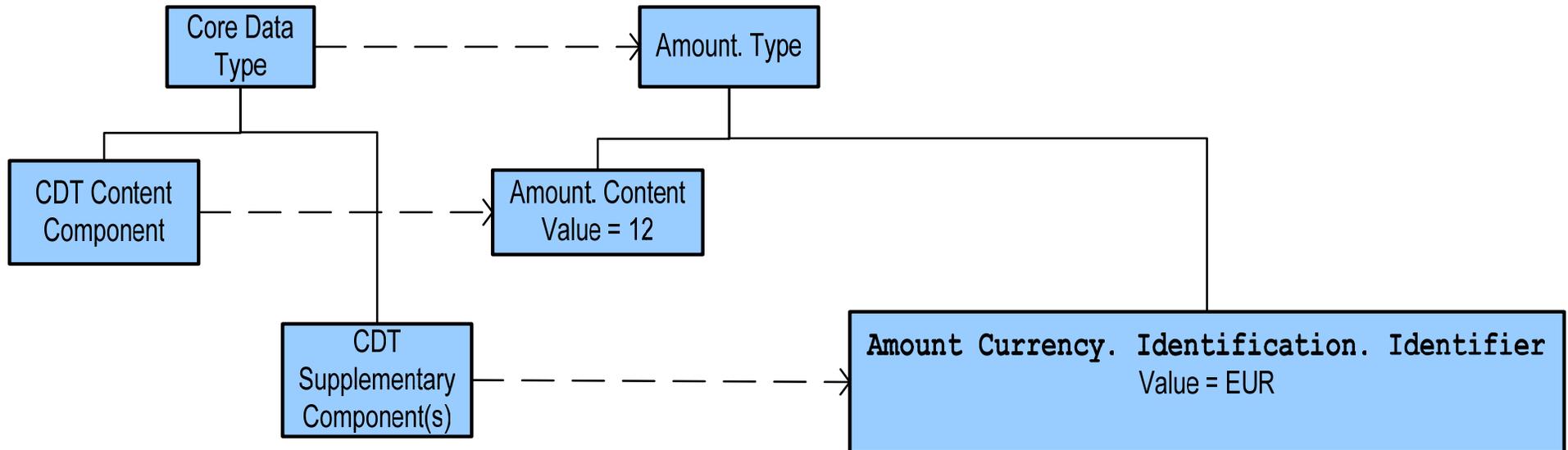
<Data Type Term>. Type



22 Defined Data types



Example Please



- **CDT Amount. Type- primitive is decimal.**
- **Amount. Content carries the value of 12**
 - This value has no semantic meaning on its own
- **Supplementary Component is Amount Currency. Identification. Identifier**
 - gives essential extra definition to the CDT content component
 - ◆ EUR represents the Euro currency,

Representation Terms and Core Data Types

• Representation Term	Definition	Related <i>Core Data Type</i>
• Amount	<ul style="list-style-type: none"> •A number of monetary units specified in a currency where the unit of currency is explicit or implied. 	• Amount. Type
• Binary Object	<ul style="list-style-type: none"> •A set of finite-length sequences of binary octets. •[Note: This <i>Representation Term</i> shall also be used for <i>Data Types</i> representing graphics (i.e. diagram, graph, mathematical curves, or similar representation), pictures (i.e. visual representation of a person, object, or scene), sound, video, etc.] 	• Binary Object. Type
• Code	<ul style="list-style-type: none"> •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 a <i>Property</i>. •[Note: The term 'Code' should not be used if the character string identifies an instance of an <i>Object Class</i> or an object in the real world, in which case the <i>Representation Term</i> identifier should be used.] 	• Code. Type
• Date	<ul style="list-style-type: none"> •A particular point in the progression of dates (ISO 8601) 	• Date. Type
• Date Time	<ul style="list-style-type: none"> •A particular point in the progression of time (ISO 8601). 	• Date Time. Type
• Duration	<ul style="list-style-type: none"> •A Duration. Type is a period of time of a particular length without a fixed start or end time. This period of time is expressed in years, months, days, hours, minutes, seconds, and fractions of a second. 	• Duration. Type
• Graphic	<ul style="list-style-type: none"> •A finite data stream of diagram, graph, mathematical curves, or similar vector based representation in a specific notation, which is expressed in base 64 encoding. 	• Graphic. Type
• Identifier	<ul style="list-style-type: none"> •A character string used to establish the identity of, and distinguish uniquely, one instance of an object within an identification scheme from all other objects within the same scheme. 	• Identifier. Type

Representation Terms and Core Data Types

• Representation Term	Definition	Related Core Data Type
• Indicator	<ul style="list-style-type: none"> •A list of exactly two mutually exclusive Boolean values that express the only possible states of a <i>Property</i>. •[Note: Values typically indicate a condition such as on/off; true/false etc.] 	• Indicator. Type
• Measure	<ul style="list-style-type: none"> •A numeric value determined by measuring an object. Measures are specified with a unit of measure. The applicable unit of measure is taken from UN/ECE Rec. 20. •[Note: This <i>Representation Term</i> shall also be used for measured coefficients (e.g. m/s).] 	• Measure. Type
• Name	<ul style="list-style-type: none"> •A word or phrase that constitutes the distinctive designation of a person, place, thing, or concept 	• Name. Type
• Numeric	<ul style="list-style-type: none"> •Numeric information that is assigned or is determined by calculation, counting or sequencing. It does not require a unit of quantity or a unit of measure. •[Note: This <i>Representation Term</i> shall also be used for <i>Data Types</i> representing Ratios (i.e. rates where the two units are not included or where they are the same), Percentages, etc.) 	• Numeric. Type
• Percent	<ul style="list-style-type: none"> •A number that relates to the comparison figure 100. 	• Percent. Type
• Picture	<ul style="list-style-type: none"> •A visual representation of a person, object, or scene in binary notation (octets) 	• Picture. Type
• Quantity	<ul style="list-style-type: none"> •A counted number of non-monetary units. Quantities need to be specified with a unit of quantity. •[Note: This <i>Representation Term</i> shall also be used for counted coefficients (e.g. flowers/m²).] 	• Quantity. Type
• Rate	<ul style="list-style-type: none"> •a fraction whose numerator and denominator are quantities, values, or dimensionless factors, independent from each other. 	• Rate. Type
• Ratio	<ul style="list-style-type: none"> •A fixed ratio between two values, like charge, payment, scale. 	• Ratio. Type

Representation Terms and Core Data Types

• Representation Term	Definition	Related <i>Core Data Type</i>
• Sound	•Any form of audio file. This includes files such as audio recordings in binary notation (octets).	• Sound. Type
• Text	•A character string (i.e. a finite set of characters) generally in the form of words of a language. •[Note: This <i>Representation Term</i> shall also be used for names (i.e. word or phrase that constitutes the distinctive designation of a person, place, thing or concept).]	• Text. Type
• Time	•A particular point in the progression of time. (ISO 8601)	• Time. Type
• Value	•Expresses the concept of numeric worth in general.	• Value. Type
• Video	•The recording, reproducing or broadcasting of visual images on magnetic tape or digitally in binary notation (octets)	• Video. Type

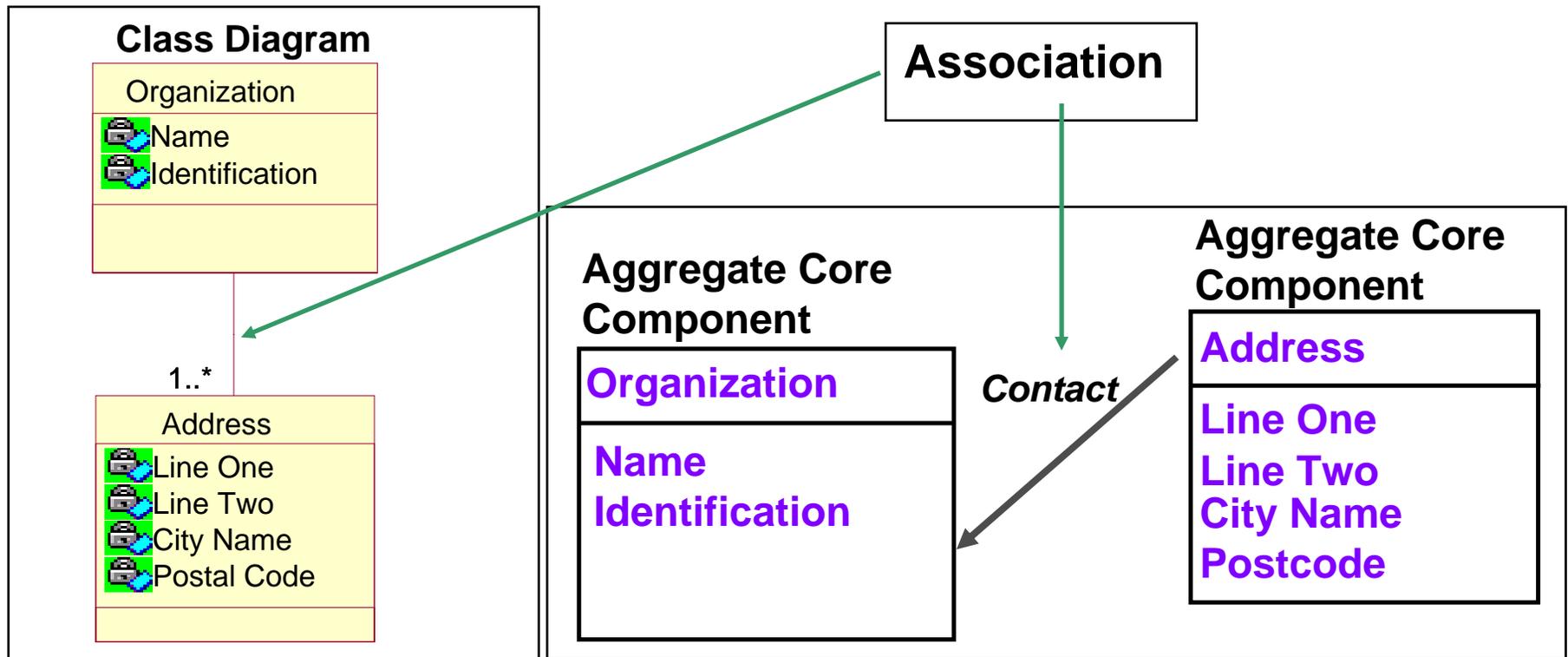
Core Data Type – Amount. Type

•Dictionary Entry Name	•Object Class Term	•Property Term	•Representation Term	•Suffix	•Primitive Type	•Card.	•Restriction	•Definition	•Remarks	•Default
•Amount. Type	•Amount			•Type						
•Amount. Content	•Amount			•Content	•Decimal	•1..1	<ul style="list-style-type: none"> •Total Digits •Fractional Digits •Min. Inclusive •Max. Inclusive •Min. Exclusive •Max. Exclusive 	•A number of monetary units		
•Amount Currency. Identification. Code	•Amount Currency	•Identification	•Code		•String	•1..1	<ul style="list-style-type: none"> •Expression •Length •Min. Length •Max. Length •Enumeration 	•The currency of the amount	•ISO 4217 alpha codes are recommended, and these are also published free of charge as UN/ECE Recommendation 9	
•Amount Currency. Code List Identification. Identifier	•Amount Currency	•Code List Identification	•Identifier		•String	•0..1	<ul style="list-style-type: none"> •Expression •Length •Min. Length •Max. Length •Enumeration 	•The identification of the currency code list		•ISO 4217
•Amount Currency. Code List Agency. Identifier	•Amount Currency	•Code List Agency	•Identifier		•String	•0..1	<ul style="list-style-type: none"> •Expression •Length •Min. Length •Max. Length •Enumeration 	•The agency that maintains the currency code list	•UN/EDIFACT 3055 is recommended.	
•Amount Currency. Code List Agency Name. Text	•Amount Currency	•Code List Agency	•Name		•String	•0..1	<ul style="list-style-type: none"> •Expression •Length •Min. Length •Max. Length •Enumeration 	•The name of the agency that maintains the code list	•Use if entry not found in Amount Currency. Code List Agency. Identifier	
•Amount Currency. Code List Version. Identifier	•Amount Currency	•Code List Version	•Identifier		•String	•0..1	<ul style="list-style-type: none"> •Expression •Length •Min. Length •Max. Length •Enumeration 	•The version of the code list being used		

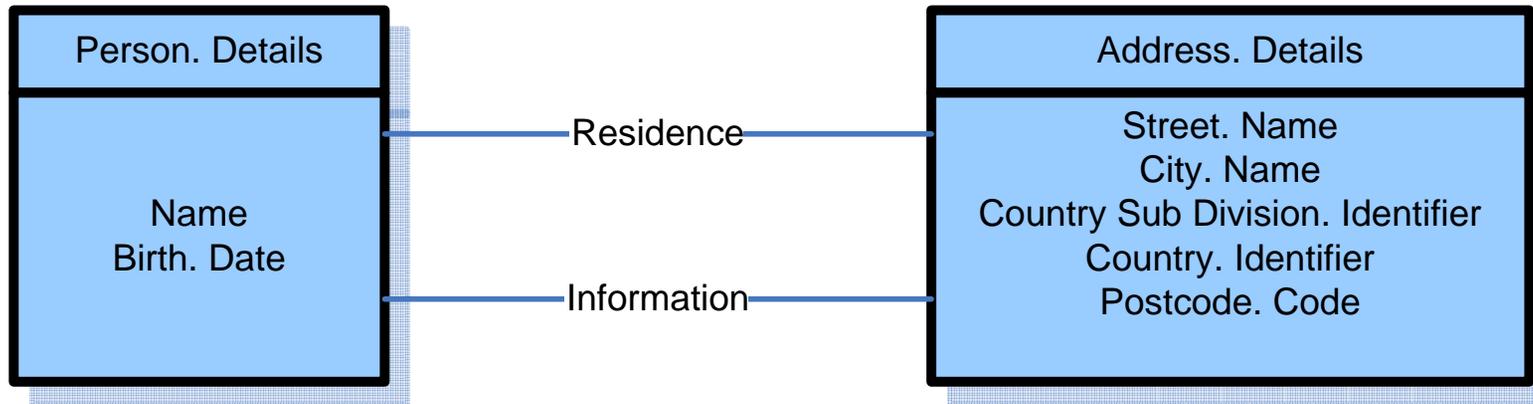
Association Core Component (ASCC)

- **Complex Property**
- **Naming mechanism for expressing the relationship between two object classes**
- **Expresses the nature of the association**

Organization. Contact. Address



Lets Look At That Again



The ACC Person. Details has four properties:

- **Two BCC properties –**
 - **Person. Name**
 - **Person. Birth. Date**
- **Two ASCC properties –**
 - **Person. Residence. Address**
 - **Person. Information. Address.**



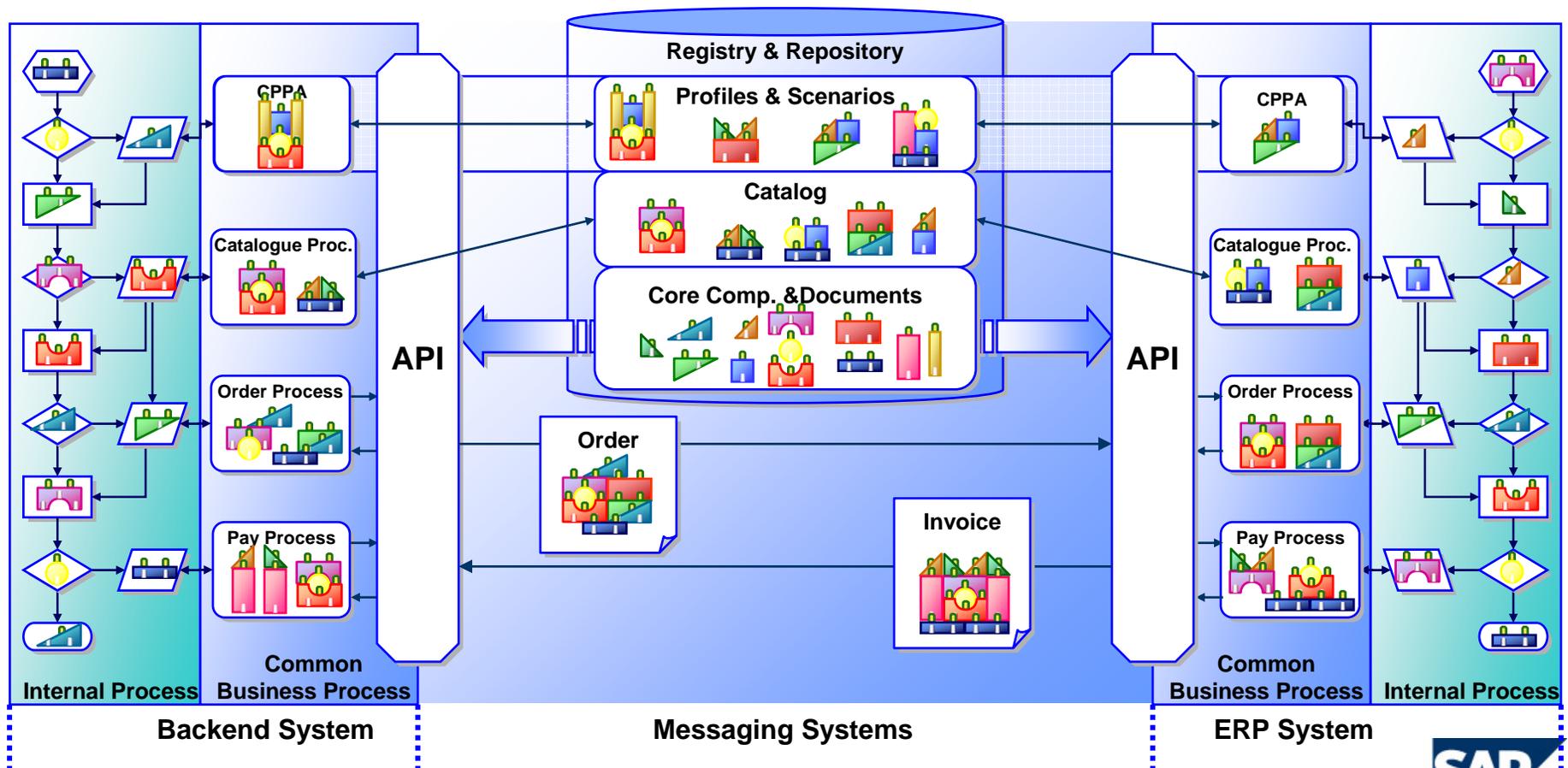
From Conceptual to Real World

The Advantage of Conceptual and Context Specific Semantics

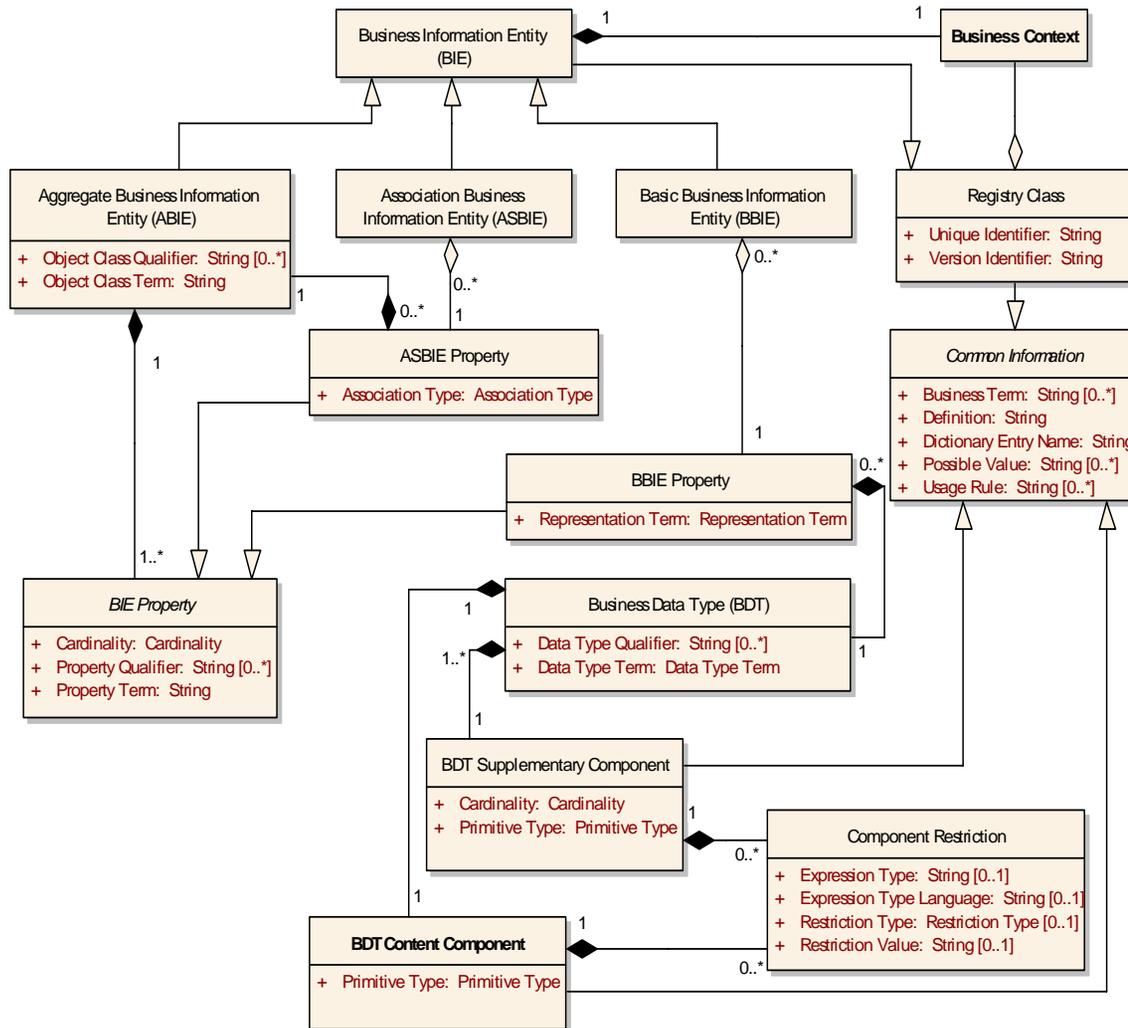
Unambiguous understanding of business information in any industry (semantic) and application (technique)

Same use of business information internally (applications) and externally (between business partners)

Less effort for internal/external interoperability



Context Specific Artifacts - Business Information Entities



■ BIEs are built from their corresponding CCs through the application of qualifiers and context

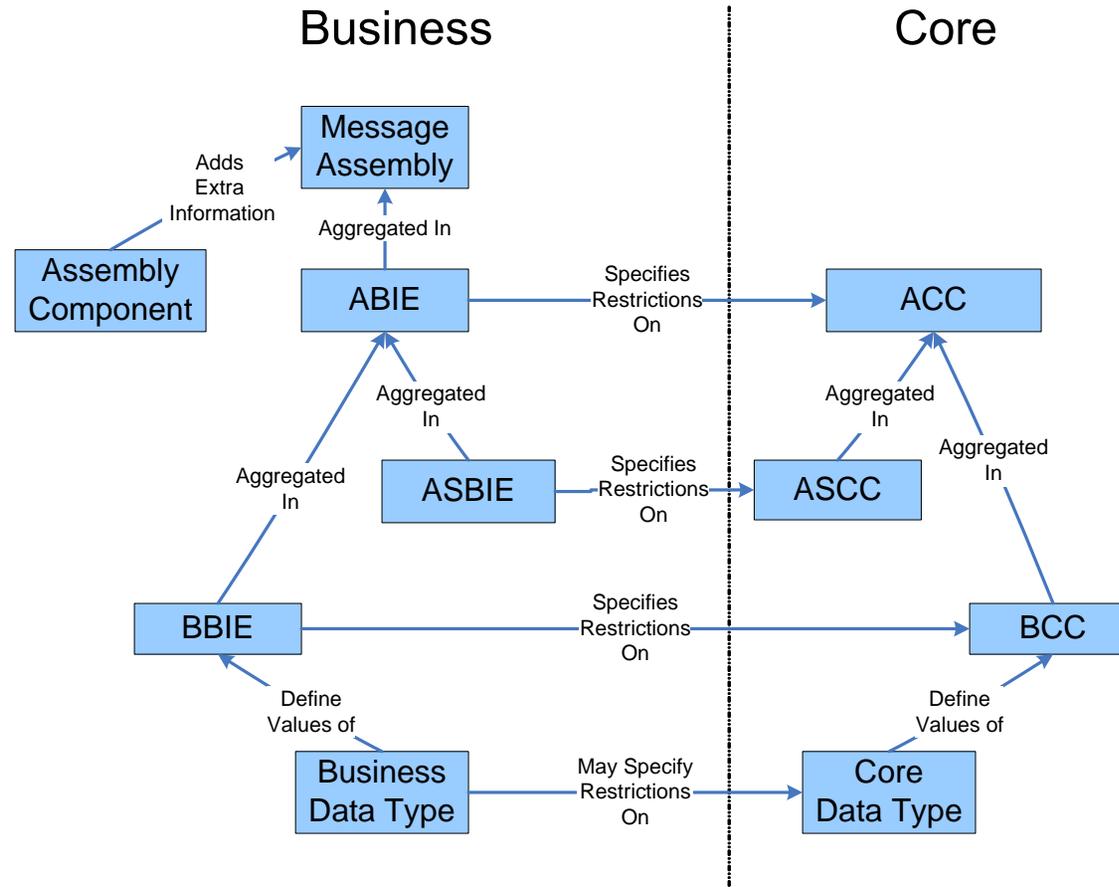
■ BIE composites are suitable for reuse with the same context values – as well as different context values

■ BIEs aggregate up to and including the information exchange level

■ BIEs specify restrictive data typing as appropriate for context specific instantiations

■ BIE models can be global, federated, or stand alone

Relationship Between BIEs and CC's



Differentiating Context Through Qualifiers

Qualifiers are used when the three data element name parts described earlier are not sufficient to uniquely identify a data element

Qualifiers can be added in front of an Object Class, a Property Term, or both

Examples:

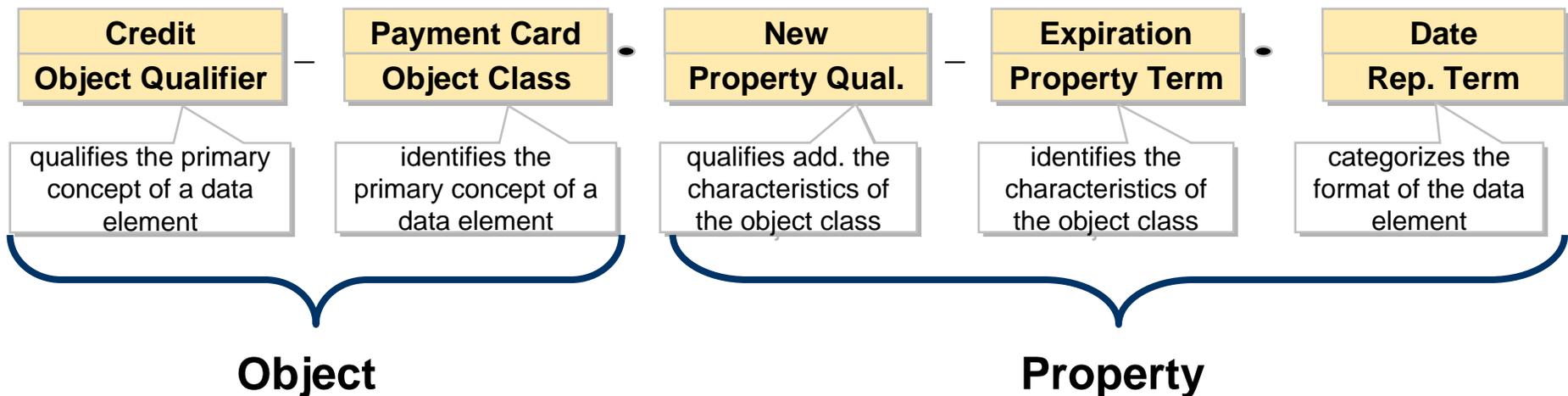
■Office_ Address. City. Name

- ◆ “Office” is a Qualifier for the “Address” Object Class
- ◆ Differentiates this data element from one named Home_ Address. City. Name

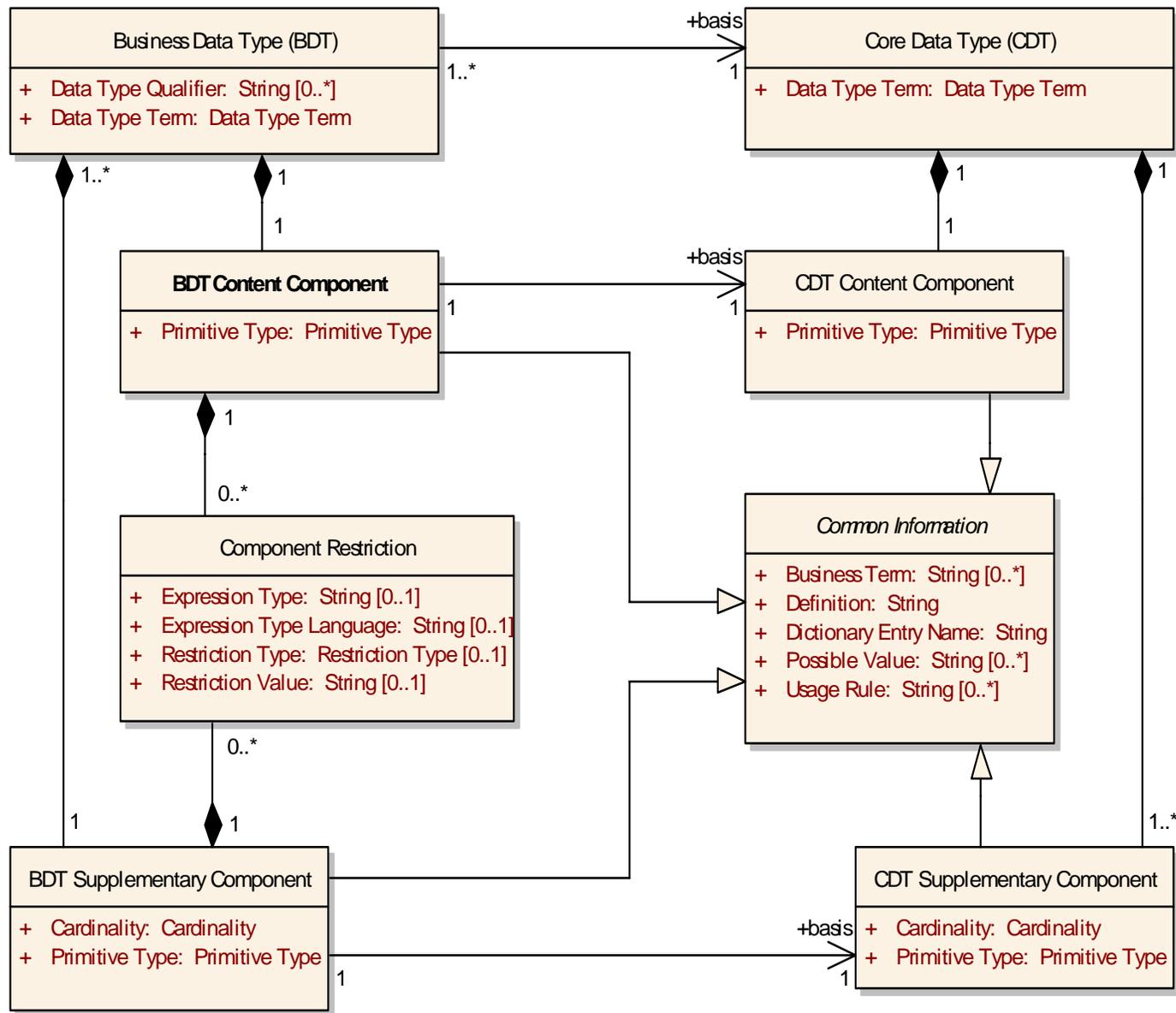
■Organization. Primary_ Contact. Name

- ◆ “Primary” is a Qualifier for the “Contact” Property Term
- ◆ Differentiates this data element from one named Organization. Secondary_ Contact. Name

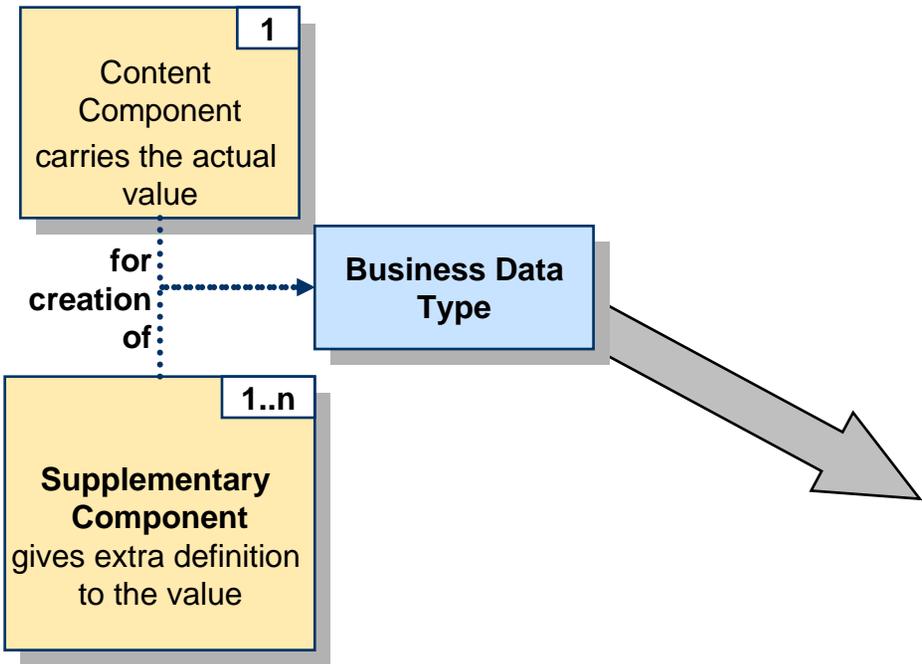
Basic Business Information Entity



Business Data Types



Business Data Type

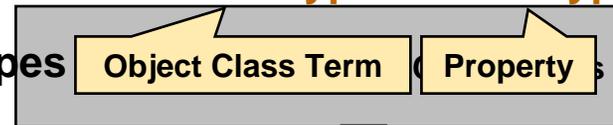


- A Business Data Type signifies the “type” of information represented by a Basic Business Information Entity
- Unqualified BDT’s are direct manifestations of their source CDT
- Business Data Types do not contain business semantics – they are “semantic-neutral”
- Business Data Type qualifiers do contain business semantics – they are used for specific contextualizations

Dictionary Entry Name
 Unqualified Data Types (DTs + Secondary Representation Terms):

<Data Type Term>. Type

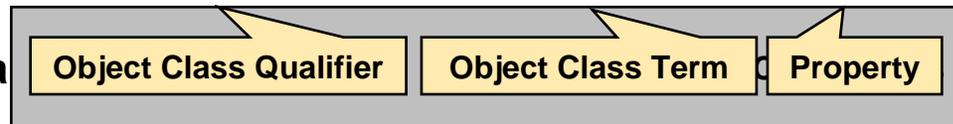
22 Defined Business Data Types



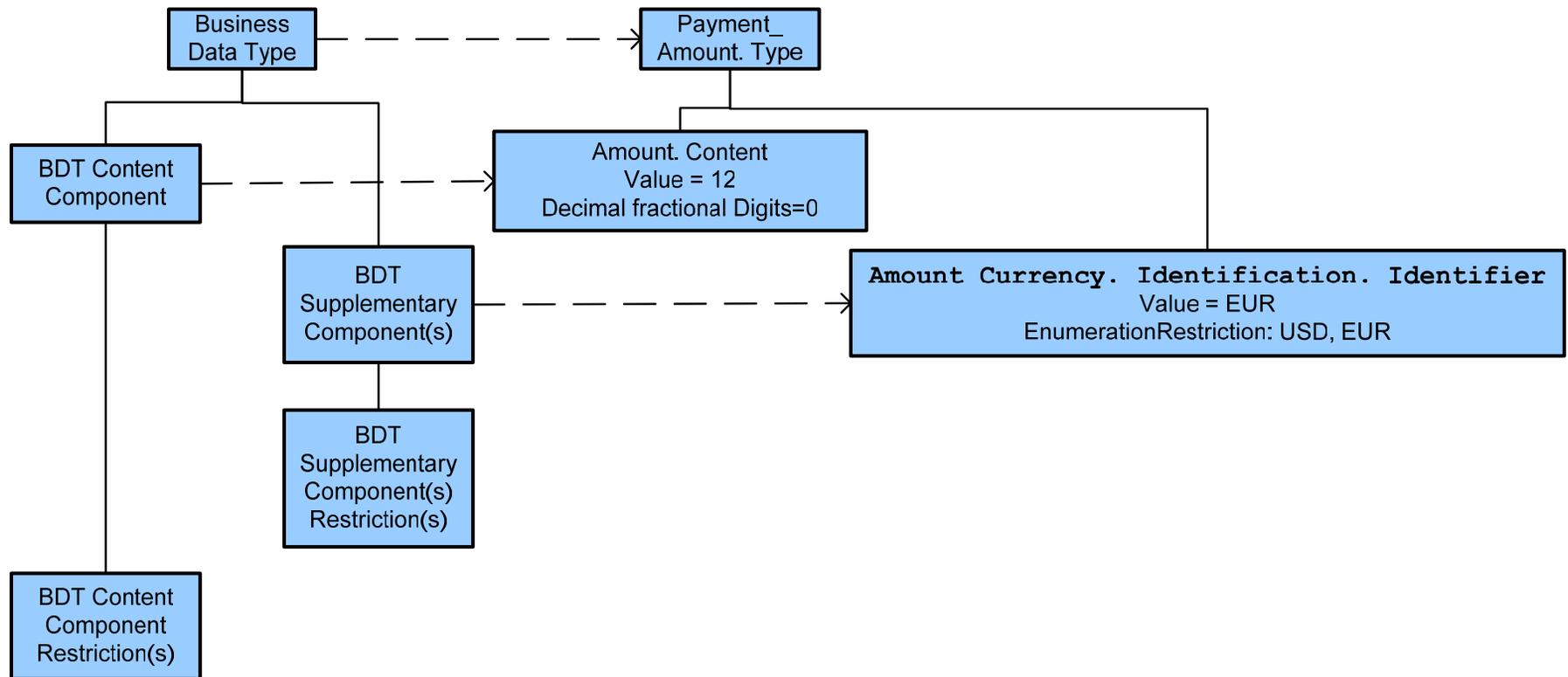
Qualified Data Types:

[?<Data Term Qualifier>_*] <Data Type Term>. Type

Infinite number of Related Data Types can be created



Differentiating Context Through Business Data Types



Differentiating Context Through Business Data Types

•Primitive Type	•Format Restriction	•Definition
String	Expression	Defines the set of characters that can be used at a particular position in a string.
String	Length	Defines the required length of the string.
String	Minimum Length	Defines the minimum length of the string. [Note] This format restriction shall not be used in combination with the Length format restriction .
String	Maximum Length	Defines the maximum length of the string. [Note] This format restriction shall not be used in combination with the Length format restriction .
String	Enumeration	Defines the exhaustive list of allowed values.
Decimal, Integer	Total Digits	Defines the maximum number of digits to be used.
Decimal	Fractional Digits	Defines the maximum number of fractional digits to be used.
Decimal, Integer	Minimum Inclusive	Defines the lower limit of the range of allowed values. The lower limit is also an allowed value.
Decimal, Integer	Maximum Inclusive	Defines the upper limit of the range of allowed values. The upper limit is also an allowed value.
Decimal, Integer	Minimum Exclusive	Defines the lower limit of the range of allowed values. The lower limit is no allowed value. [Note] This format restriction shall not be used in combination with the Minimum Inclusive format restriction .

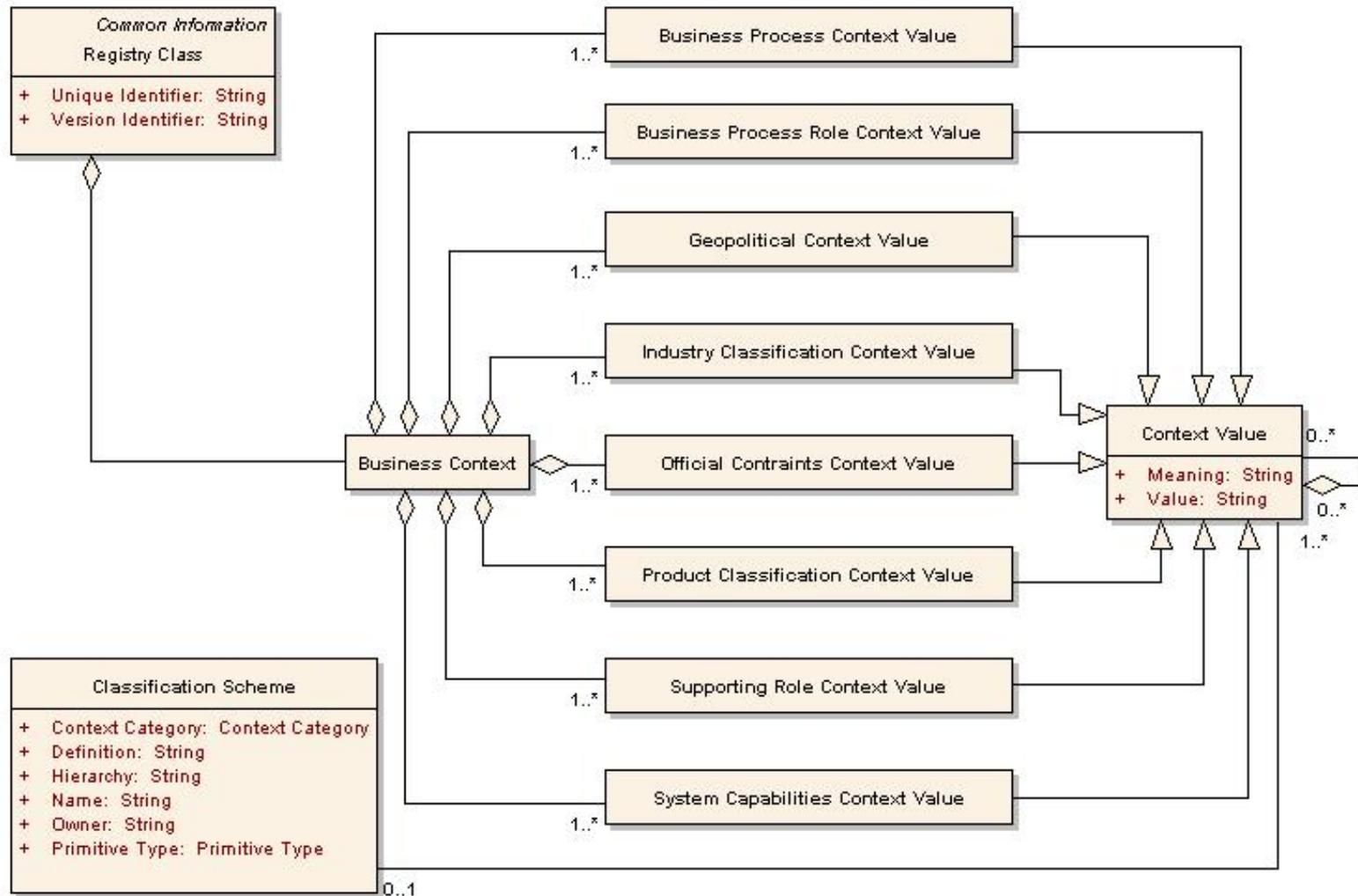
Differentiating Context Through Business Data Types

•Primitive Type	•Format Restriction	•Definition
Decimal, Integer	Maximum Exclusive	Defines the upper limit of the range of allowed values. The upper limit is no allowed value. [Note] This format restriction shall not be used in combination with the Maximum Inclusive format restriction.
Date	Minimum Inclusive	Defines the lower limit of the range of allowed dates. The lower limit is also an allowed date.
Date	Maximum Inclusive	Defines the upper limit of the range of allowed dates. The upper limit is also an allowed date.
Date	Minimum Exclusive	Defines the lower limit of the range of allowed dates. The lower limit is no allowed date. [Note] This format restriction shall not be used in combination with the Minimum Inclusive format restriction.
Date	Maximum Exclusive	Defines the upper limit of the range of allowed dates. The upper limit is no allowed date. [Note] This format restriction shall not be used in combination with the Maximum Inclusive format restriction.

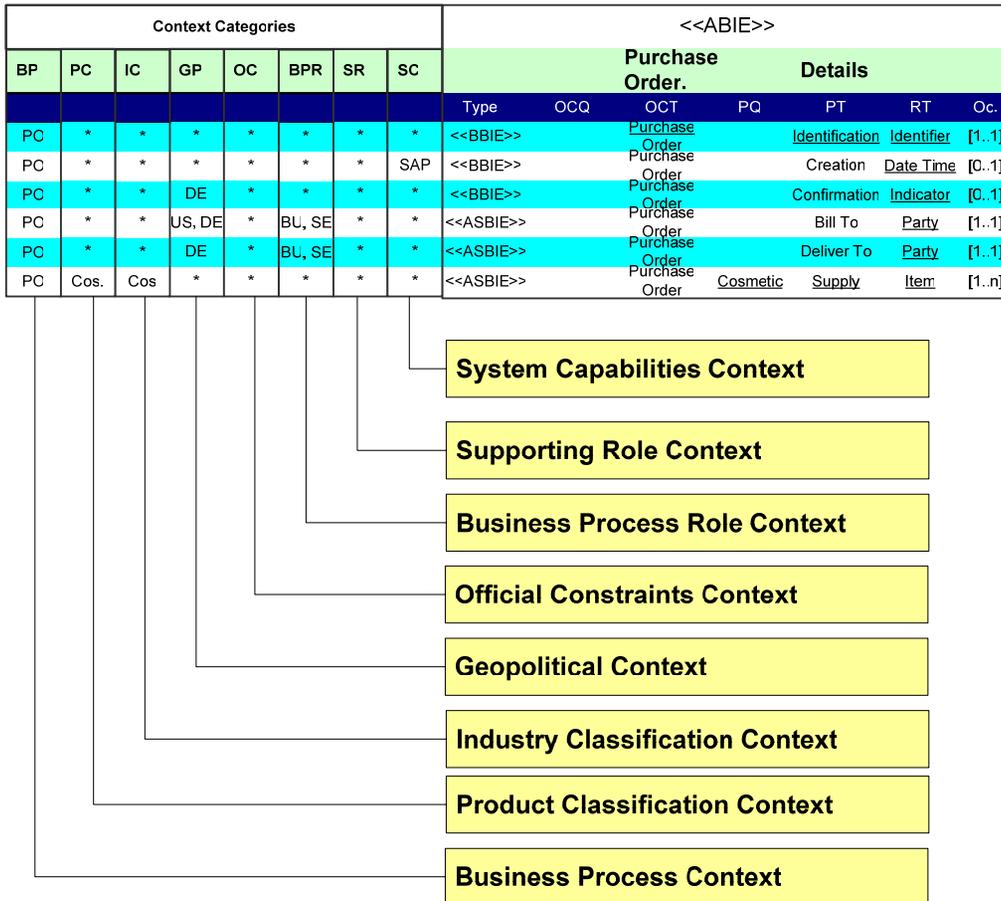
Differentiating Context Through Business Data Types

•Primitive Type	•Format Restriction	•Definition
Binary	Length	Defines the required length of the binary object.
Binary	Minimum Length	Defines the minimum length of the binary object. [Note] This format restriction shall not be used in combination with the Length format restriction .
Binary	Maximum Length	Defines the maximum length of the binary object. [Note] This format restriction shall not be used in combination with the Length format restriction .
Binary	White Space	Defines the allowed white space for the binary object

Differentiating Context Through 8 Space Values

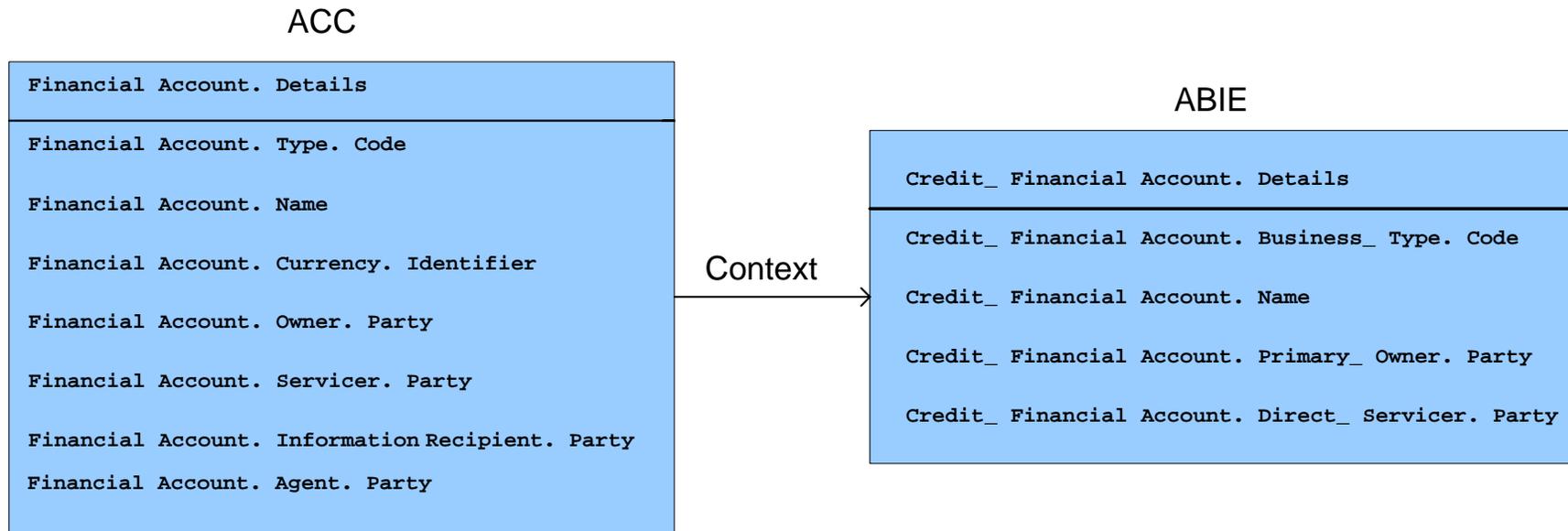


Context Driver Principle



Context Category	Value List
Business Process Context	UN/CEFACT Catalogue of Common Business Processes
Product Classification Context	UNSPSC
Industry Classification Context	UNSPSC
Geopolitical Context	ISO 3166.1 ISO 3166.2
Official Constraints Context	?
Business Process Role Context	UN/EDIFACT DE 3055 (Roles)
Supporting Role Context	?
System Capabilities Context	?

Example - Aggregate Business Information Entity

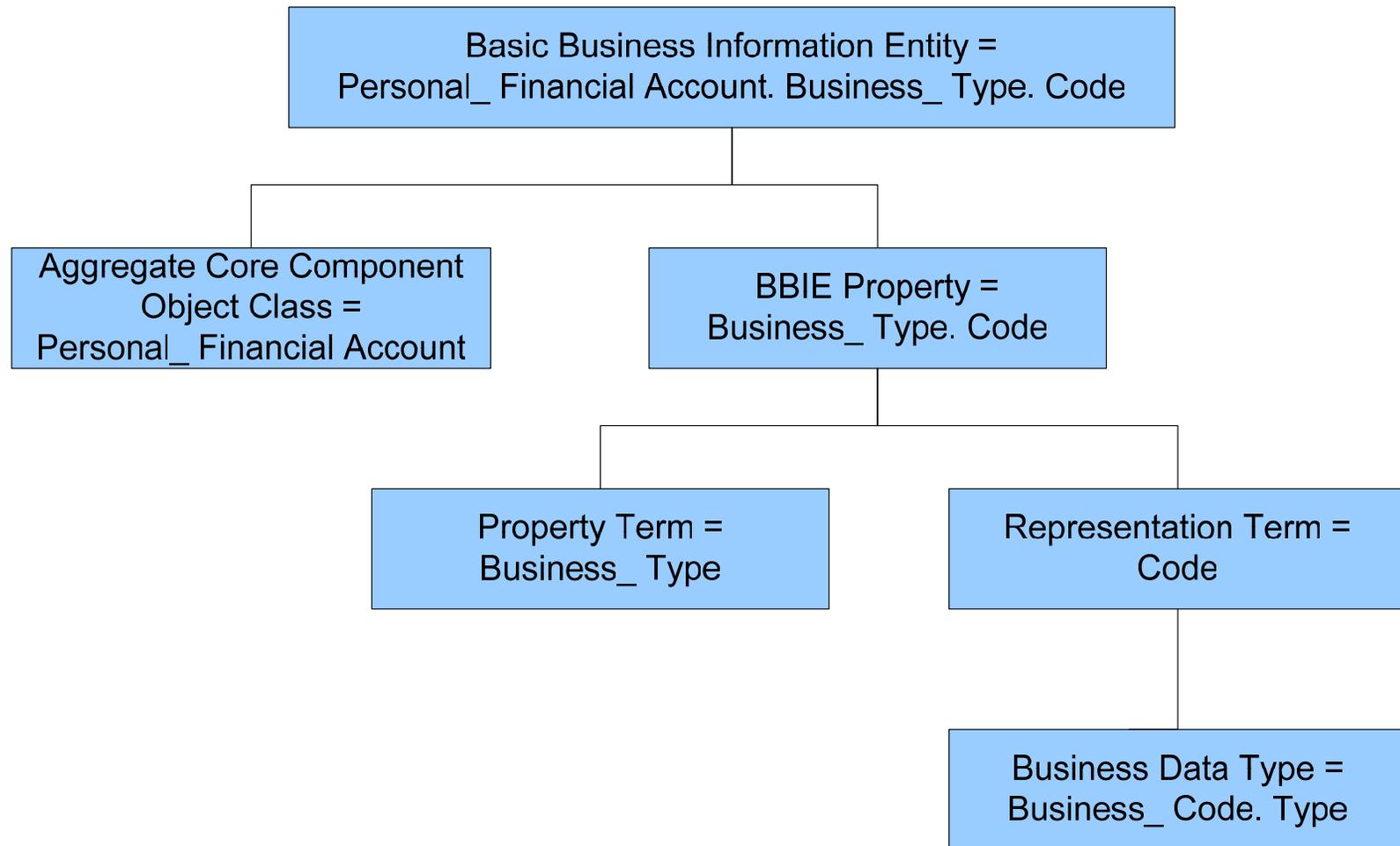


■ For the ABIE Credit_ Financial Account. Details, context has been applied to the ACC of Financial Account. Details.

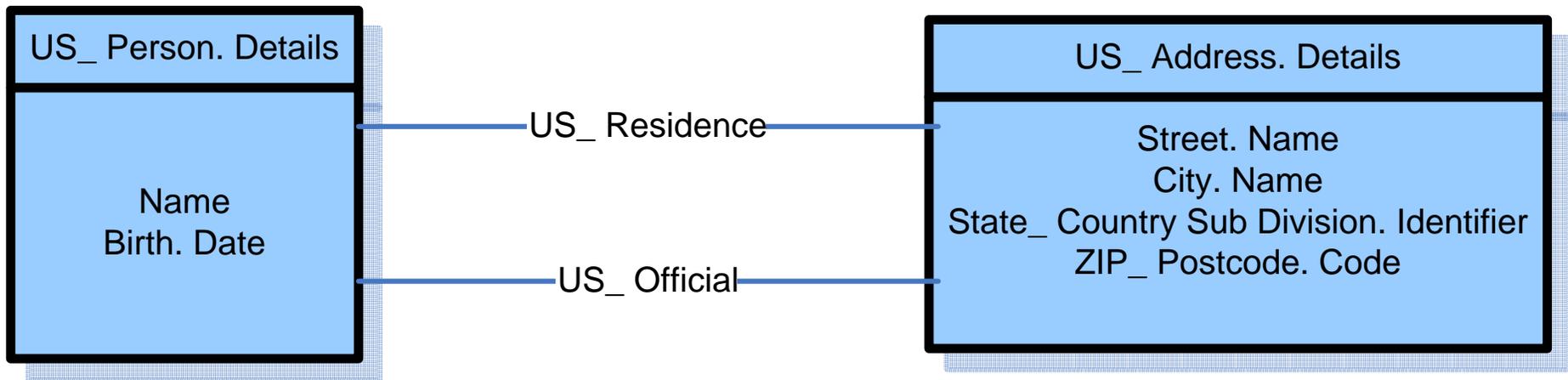
■ This context has resulted in:

- qualification of the object class
- qualification of selected property terms
- restriction on the content model

Example - Basic Business Information Entity



Example - Association Business Information Entity



- **US_Person.Details has two simple properties:**
 - Name. Name
 - Birth. Date
- **It also has two complex properties:**
 - US_Residence. US_Address
 - US_Official. US_Address.

Lets Summarize What We Have Covered So Far

CC's create common re-usable building blocks

- **Conceptual Data Constructs – Core Components – e.g. Party, Address, Organization**
- **Reusable physical/logical Data Constructs – called BIEs – e.g. Buyer_Party, Seller_Party**
- **Core Data Types, i.e. for “Amount, Code, Measure, Name, and Quantity”**
- **Business Data Types for instantiation**

Syntax neutral

- **Can be used to define business documents OR business objects/databases**

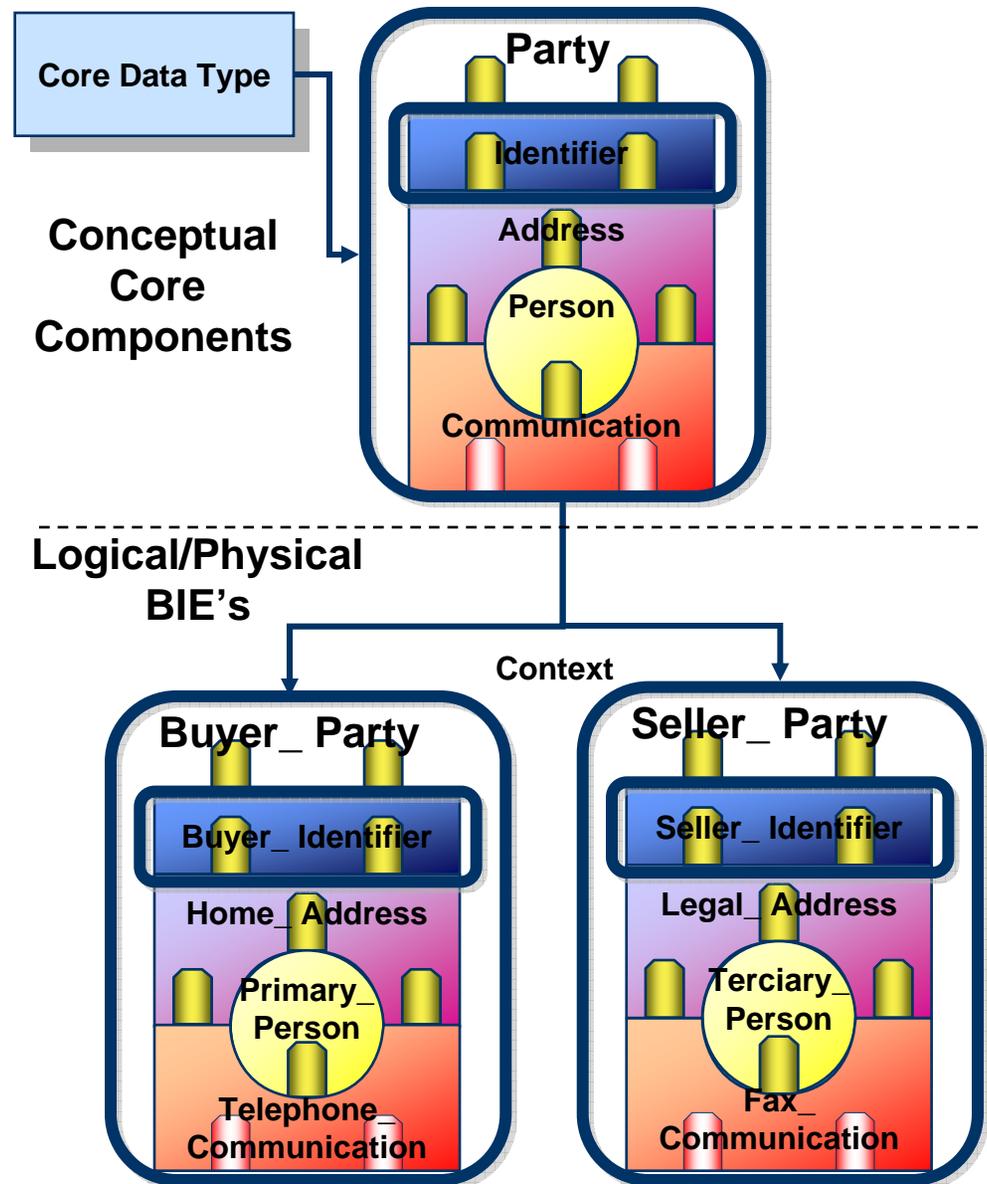
Based on Semantic Definitions

- **Clear rules on how to define semantics to explain what items mean**

Uses a Context Mechanism that controls how data constructs vary depending on the context

- **e.g. By business process, business process role, industry, country/region, etc.**

Definitions stored in a repository



The Data Interoperability Dilemma

ISO 15000-5 Core Components

Process Models, CCTS, and Syntax

Concepts In Action

CCTS and The Semantic Web

CCTS in SAP

Coordinated Standards Are Key

Stages of development over time

Enterprise Archicture &
Business Process Analysis

UML / UMM

Information Analysis/Data Modeling

ISO 11179 / ISO 15000-5

XML Development

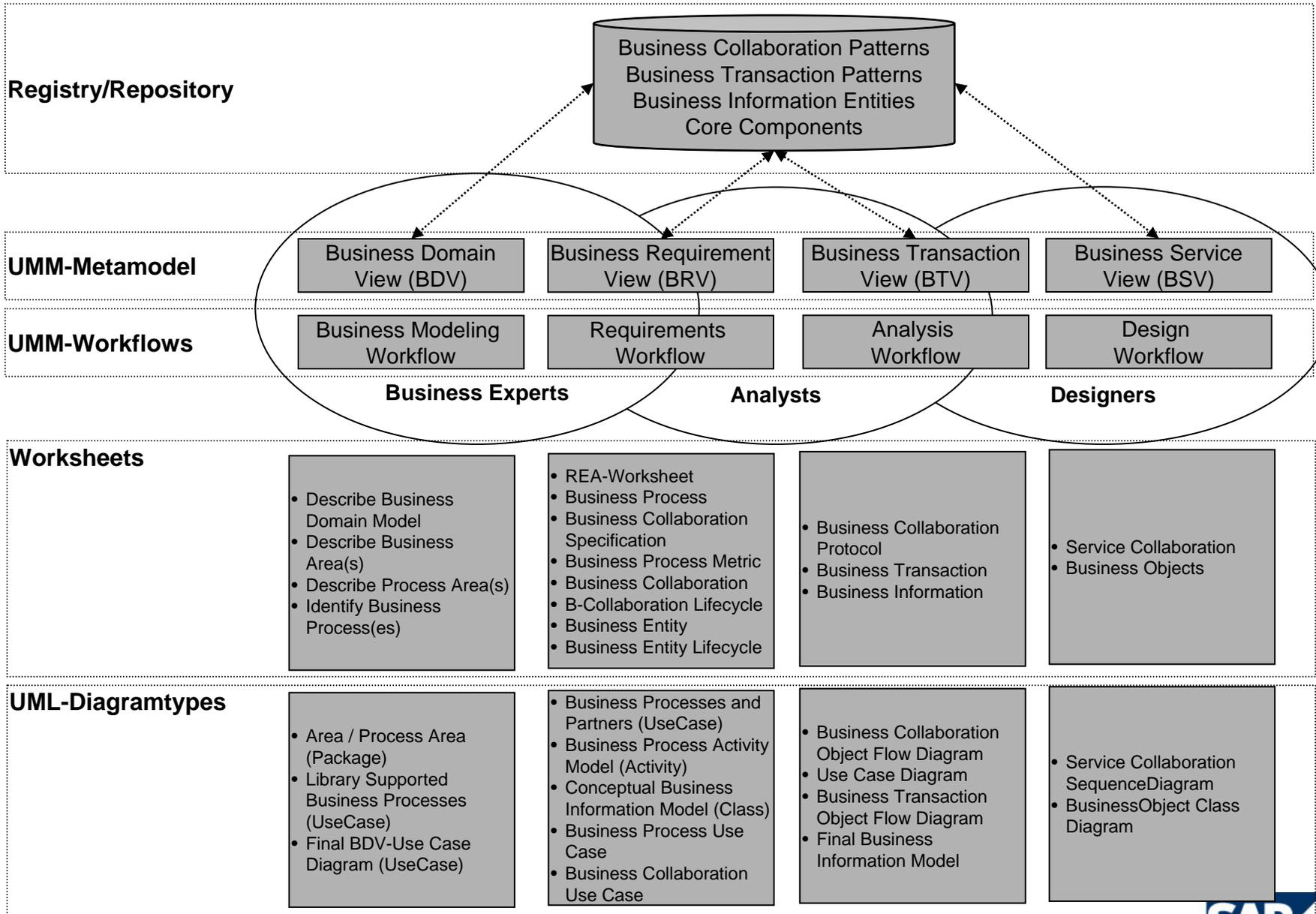
XML / XSD / NDR

- Component Reuse
- Requirements
- Process Modeling
- Data Analysis and Design
- Syntax Expression

Overlaps in standards

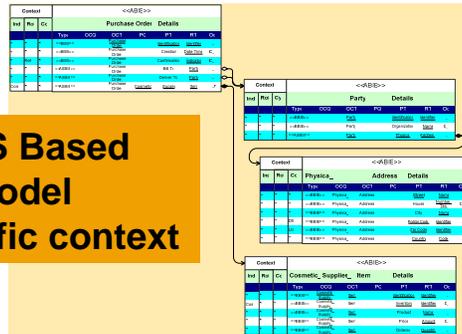
UN/CEFACT Modelling Methodology (UMM)

UN/CEFACT Modeling Methodology (UMM)



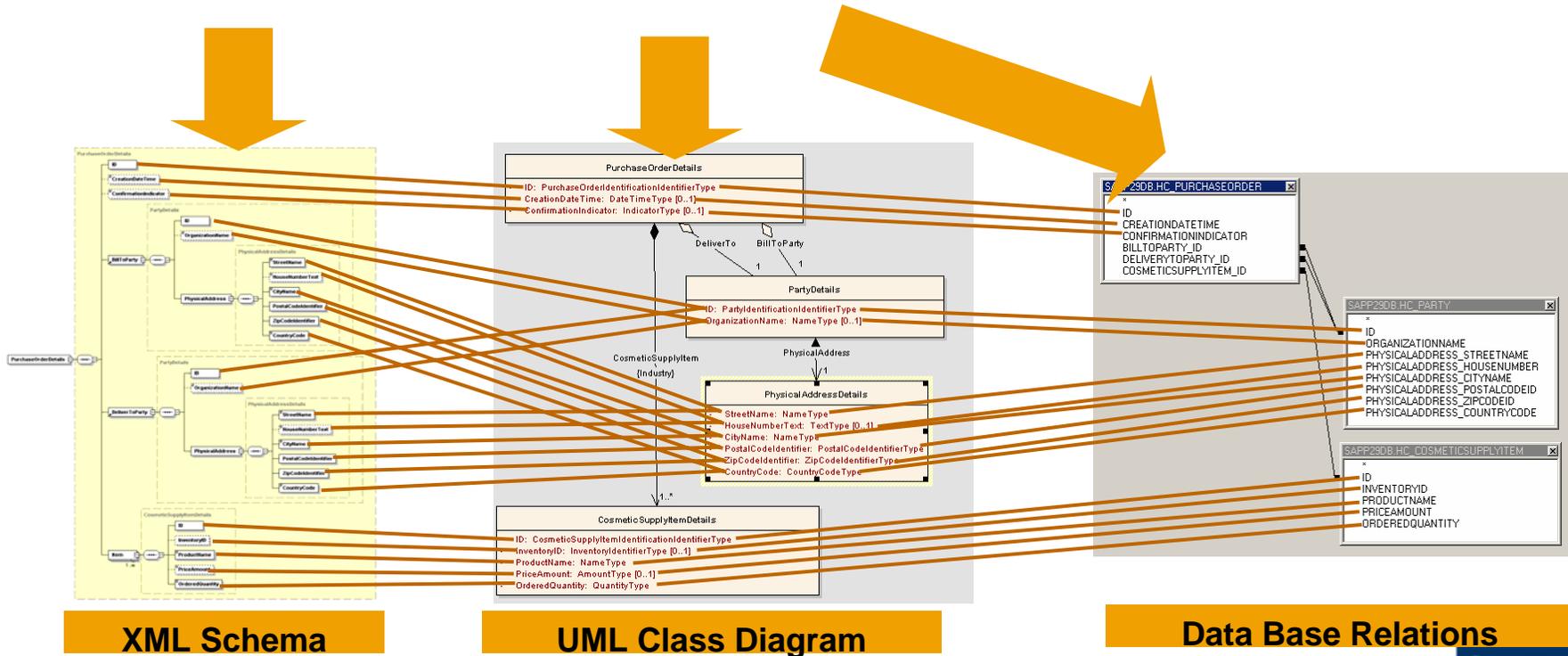
Consistent Rules for Transformation in different Syntax

CCTS Based Model in specific context



Consistency in design and use at the syntax layer:

- XML NDR for Schema Development
- EDI Transformation Rules
- UML2XML through CC Drive XMI Profiles
- UML2EDIFACT through CCTS based transformation rules



XML Schema

UML Class Diagram

Data Base Relations



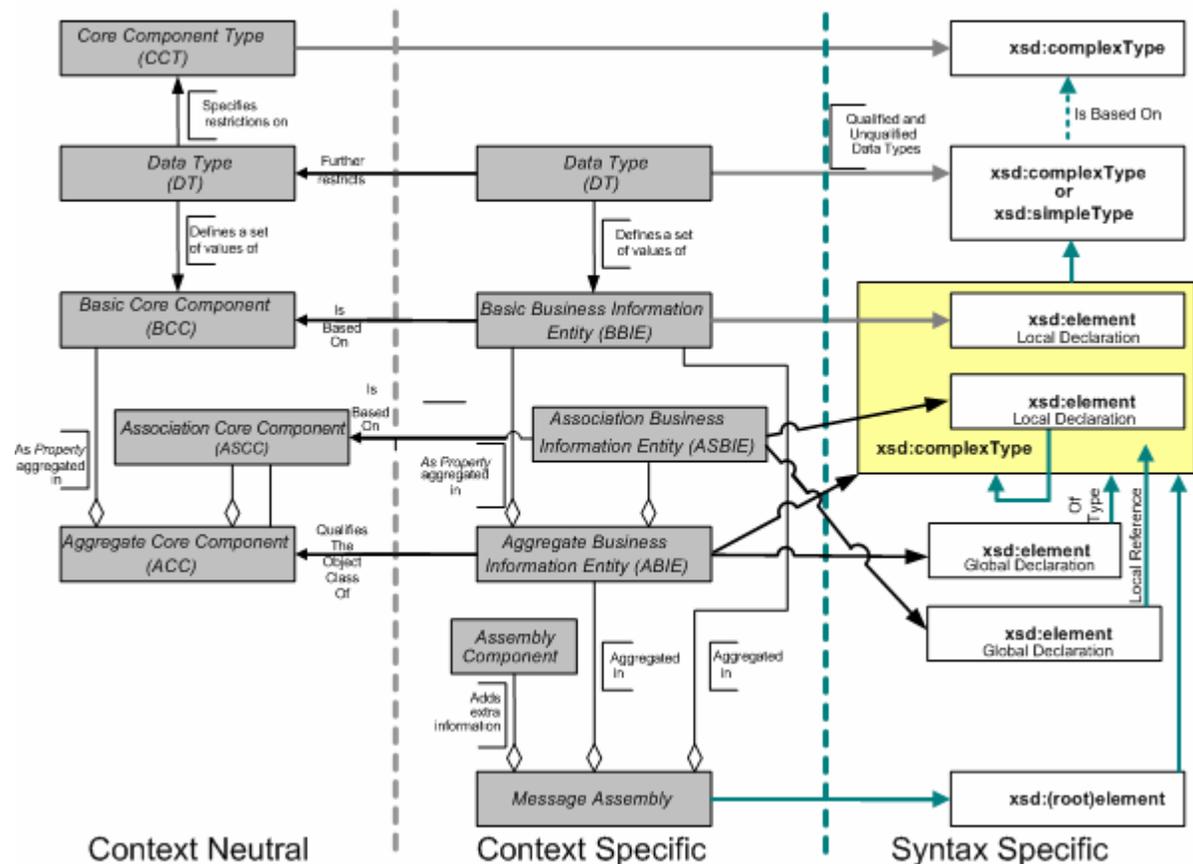
Creating the Syntax – Optimized XSD

■ **Context Neutral Core Components provide the conceptual model**

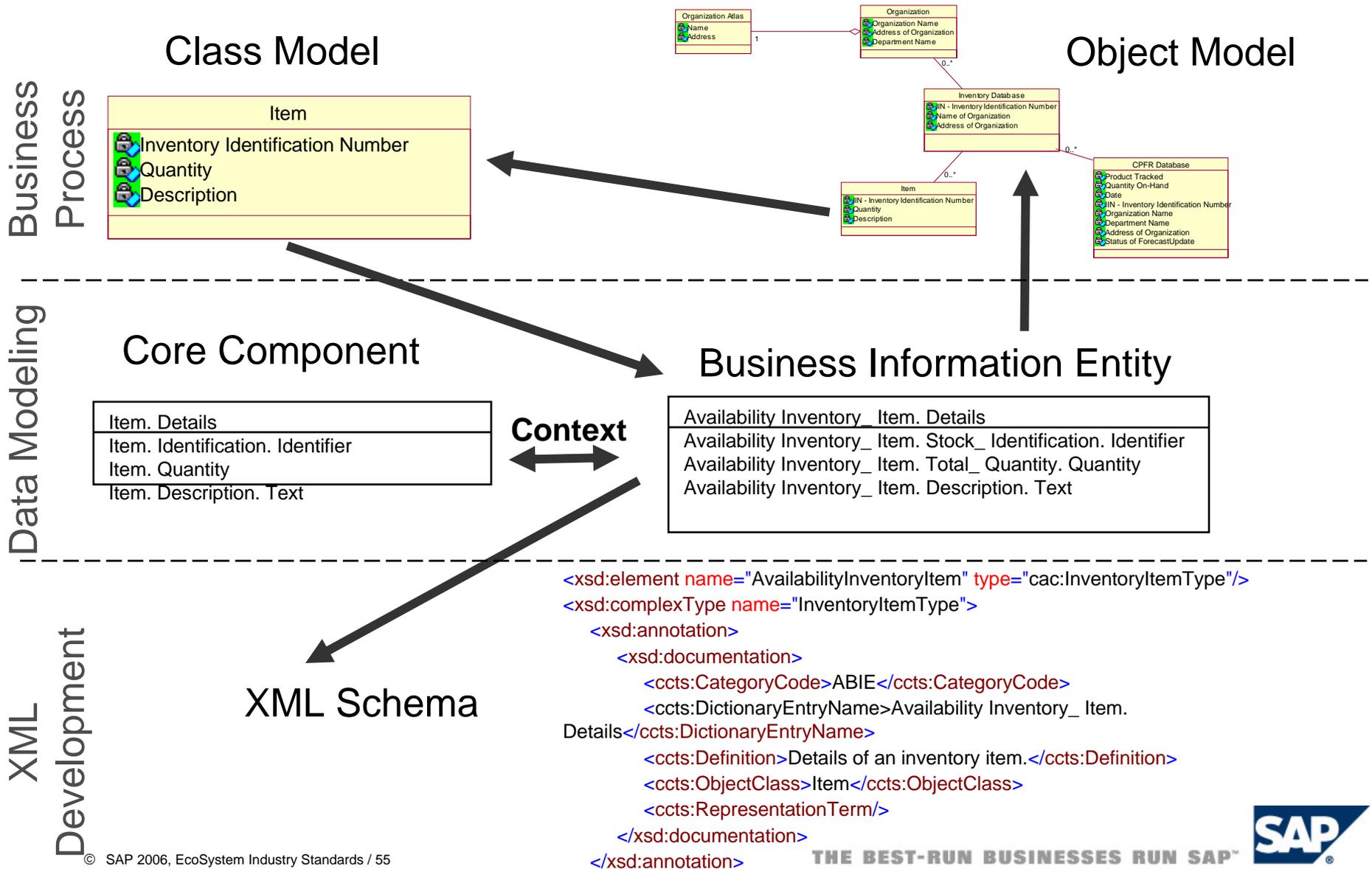
■ **Context specific BIEs provide the physical model**

■ **XSD expressions following optimized Naming and Design Rules provide the syntax specific instantiations**

■ **Tight integration between all three layers maximizes interoperability within and across database and industry use**



Integration and Interoperability in Phases of Design



The Data Interoperability Dilemma

ISO 15000-5 Core Components

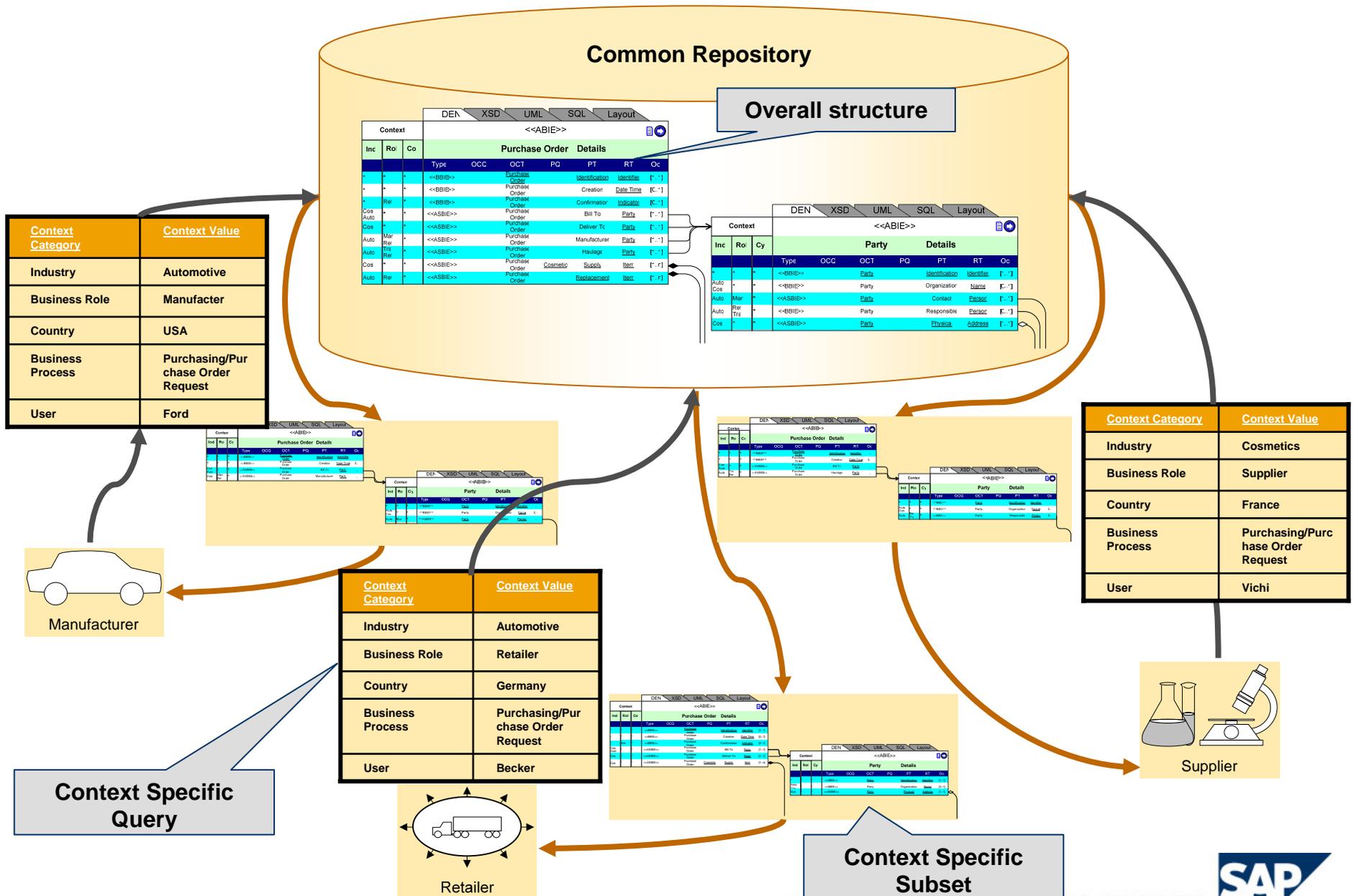
Process Models, CCTS, and Syntax

Concepts In Action

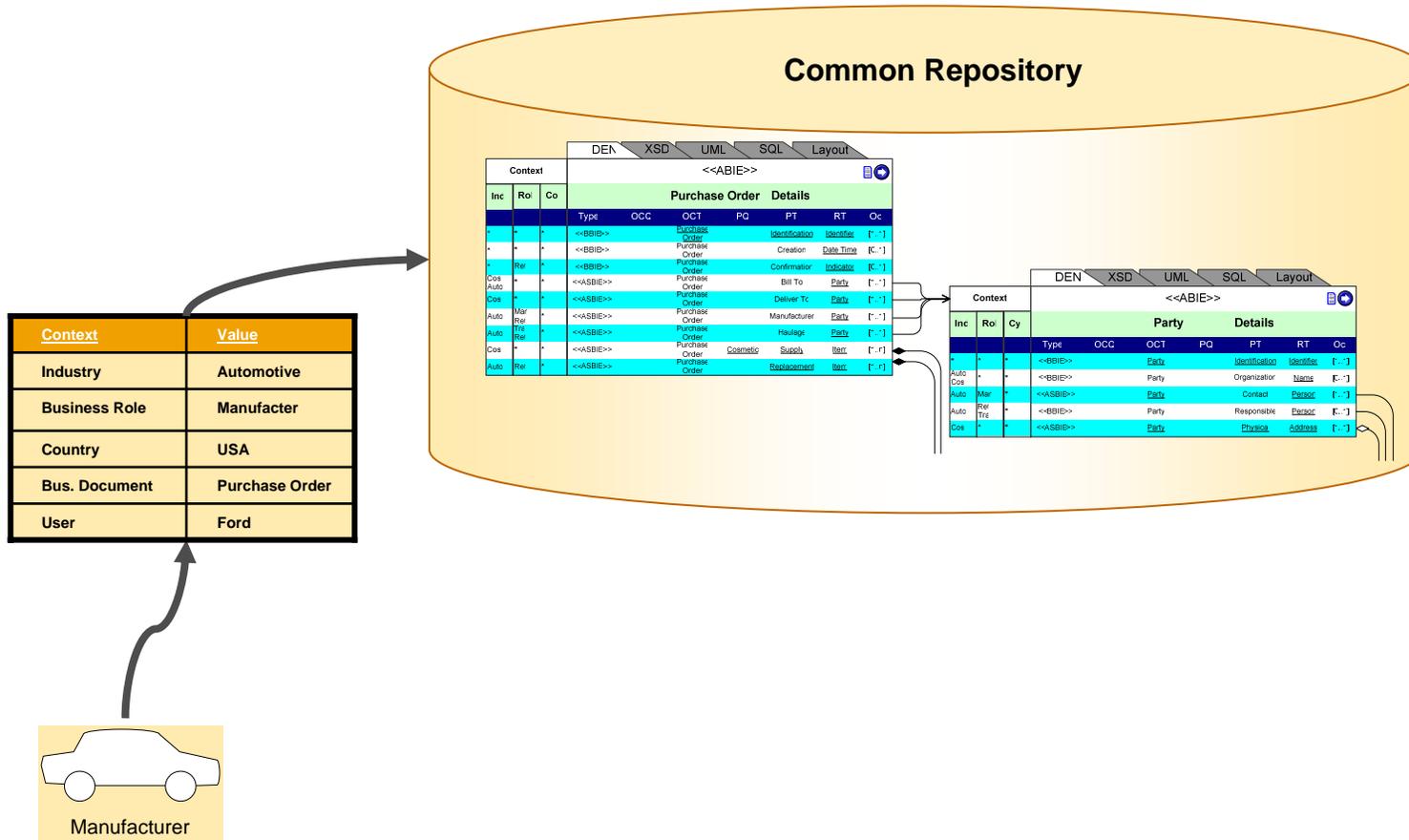
CCTS and The Semantic Web

CCTS in SAP

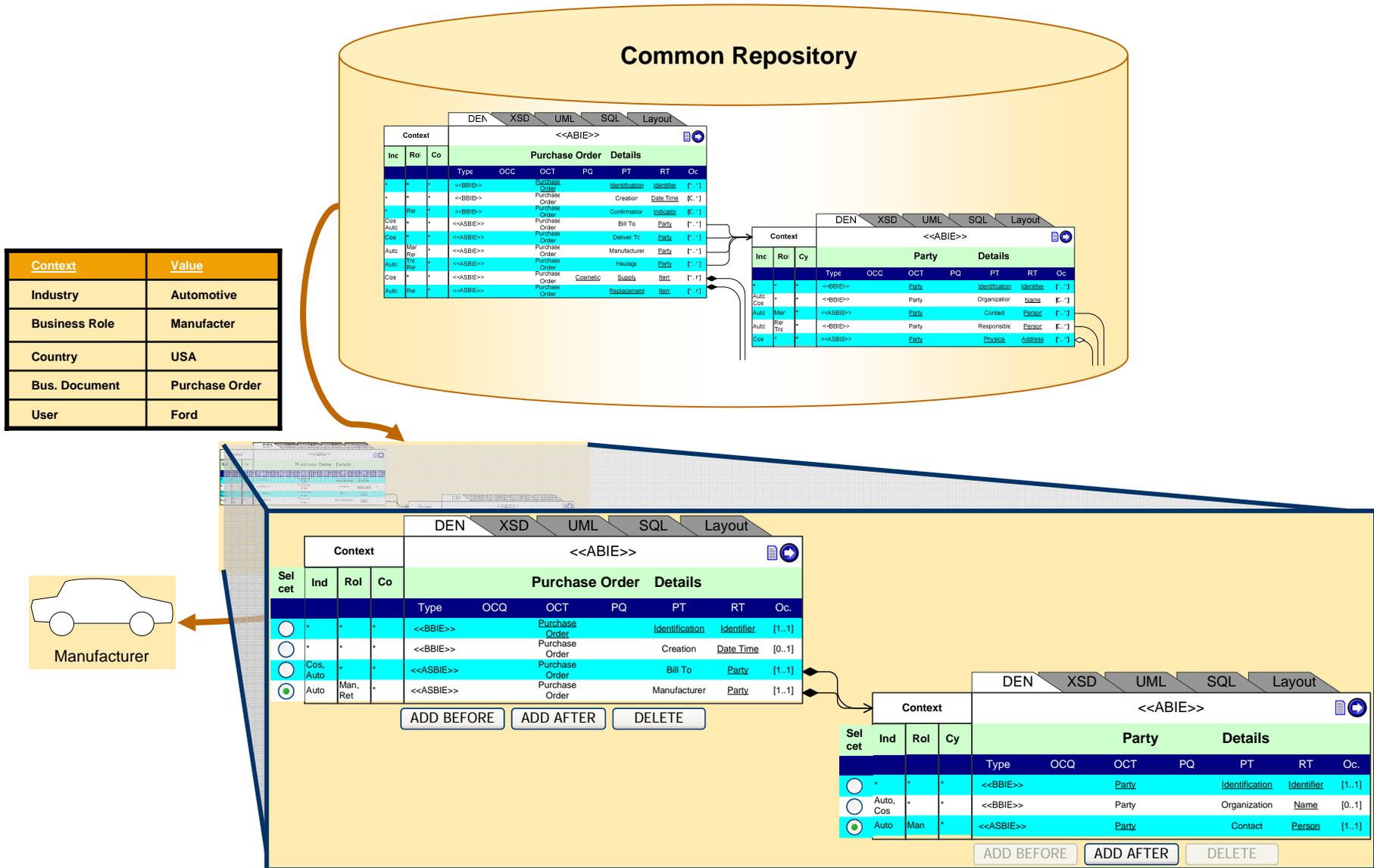
Collaborative Modeling Environment



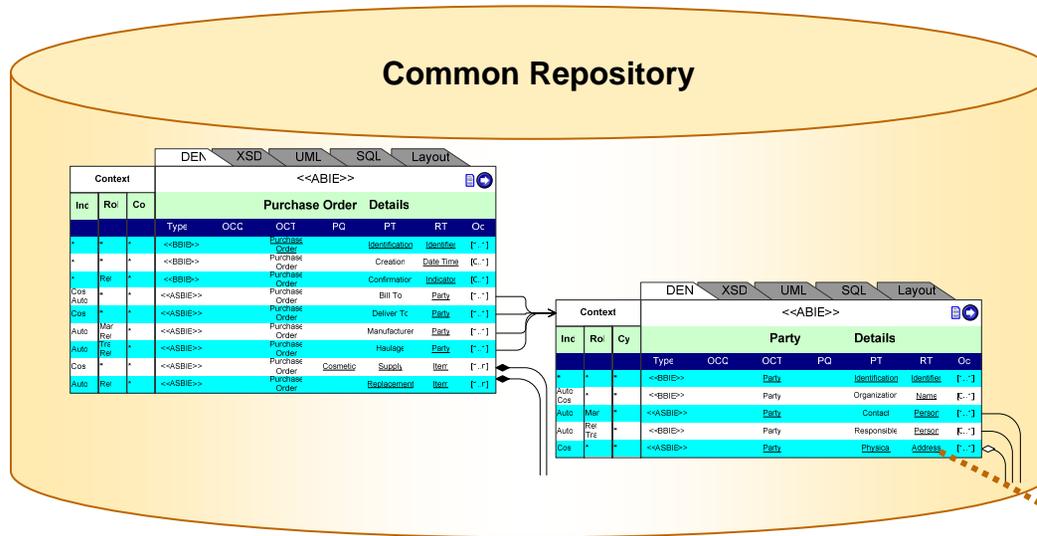
Query Subset Based on Context



Get Subset

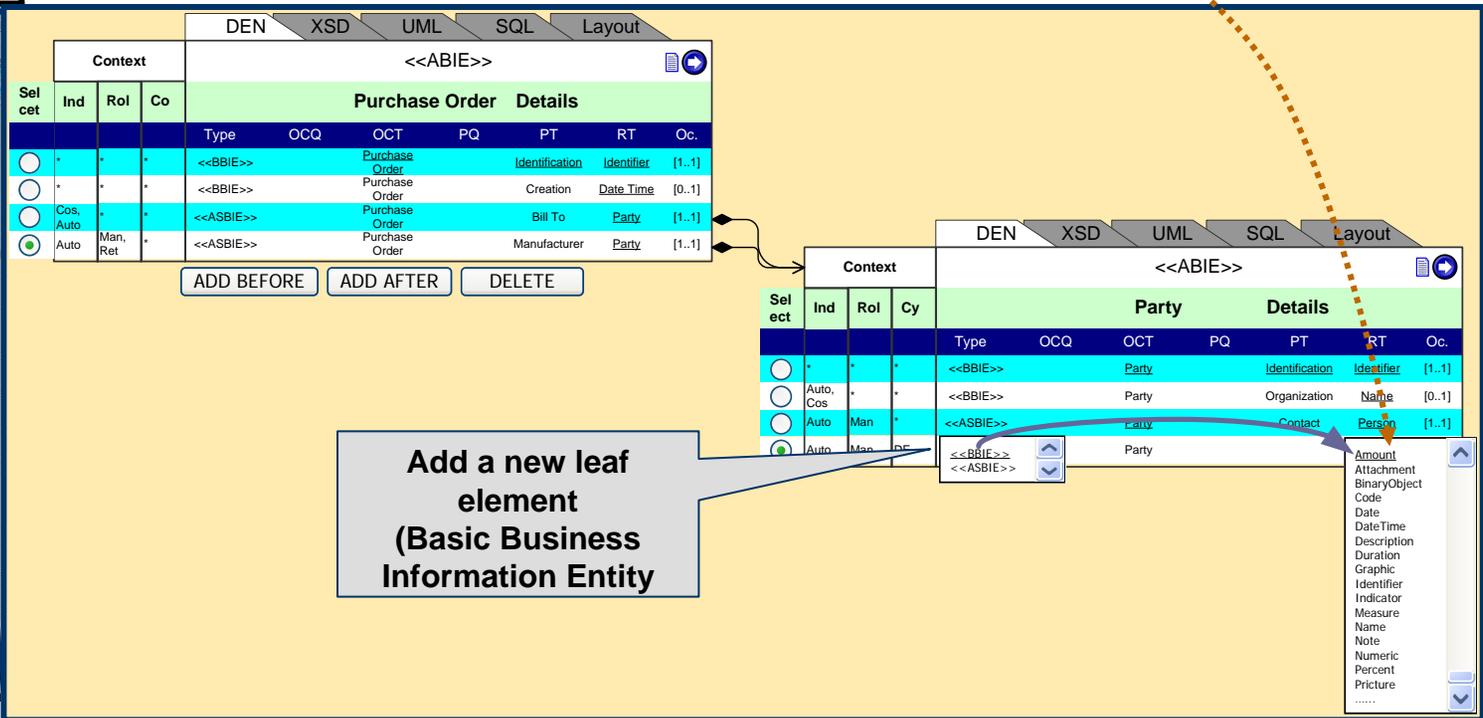
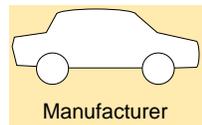


Add Customizations (In Own Context)

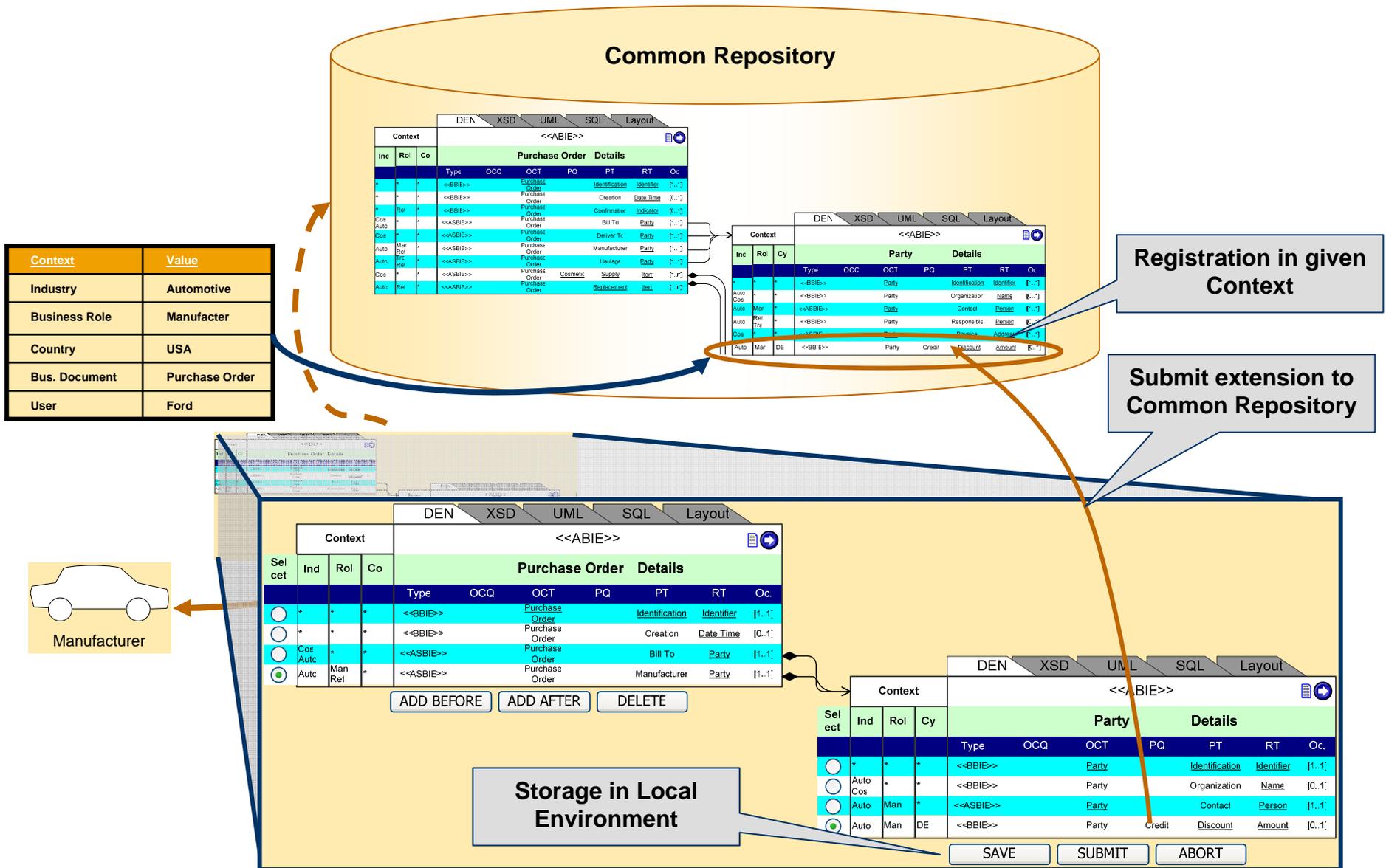


Context	Value
Industry	Automotive
Business Role	Manufacturer
Country	USA
Bus. Document	Purchase Order
User	Ford

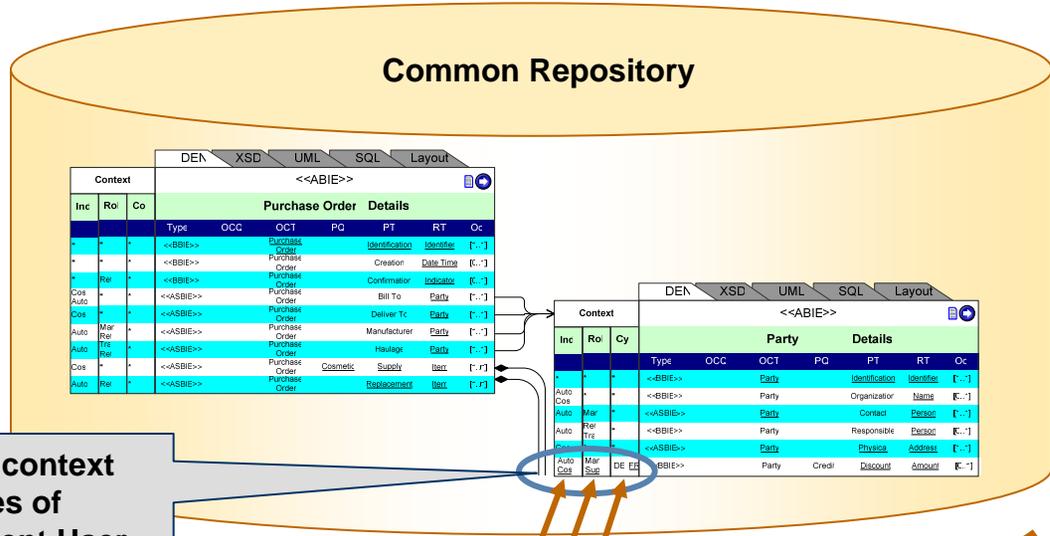
List of all possible Representation Terms



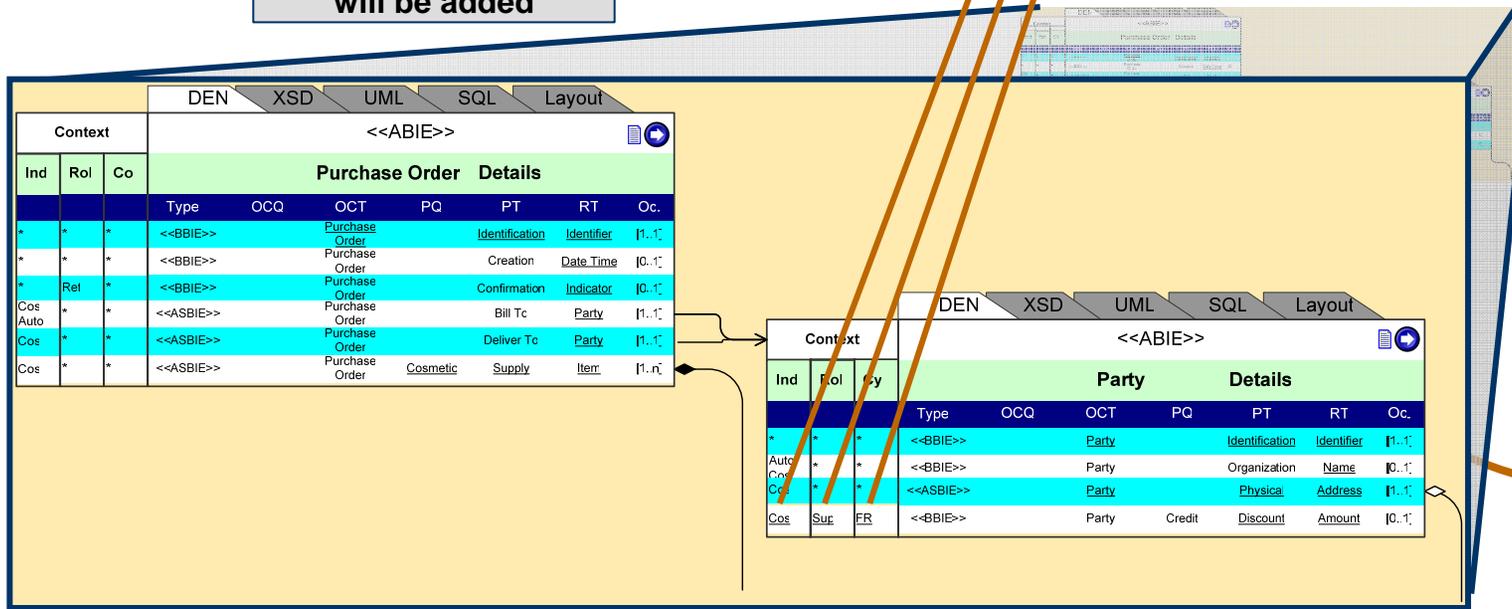
Registration in Common Repository



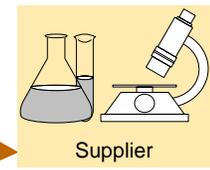
Independent Extension of Source Business Entity



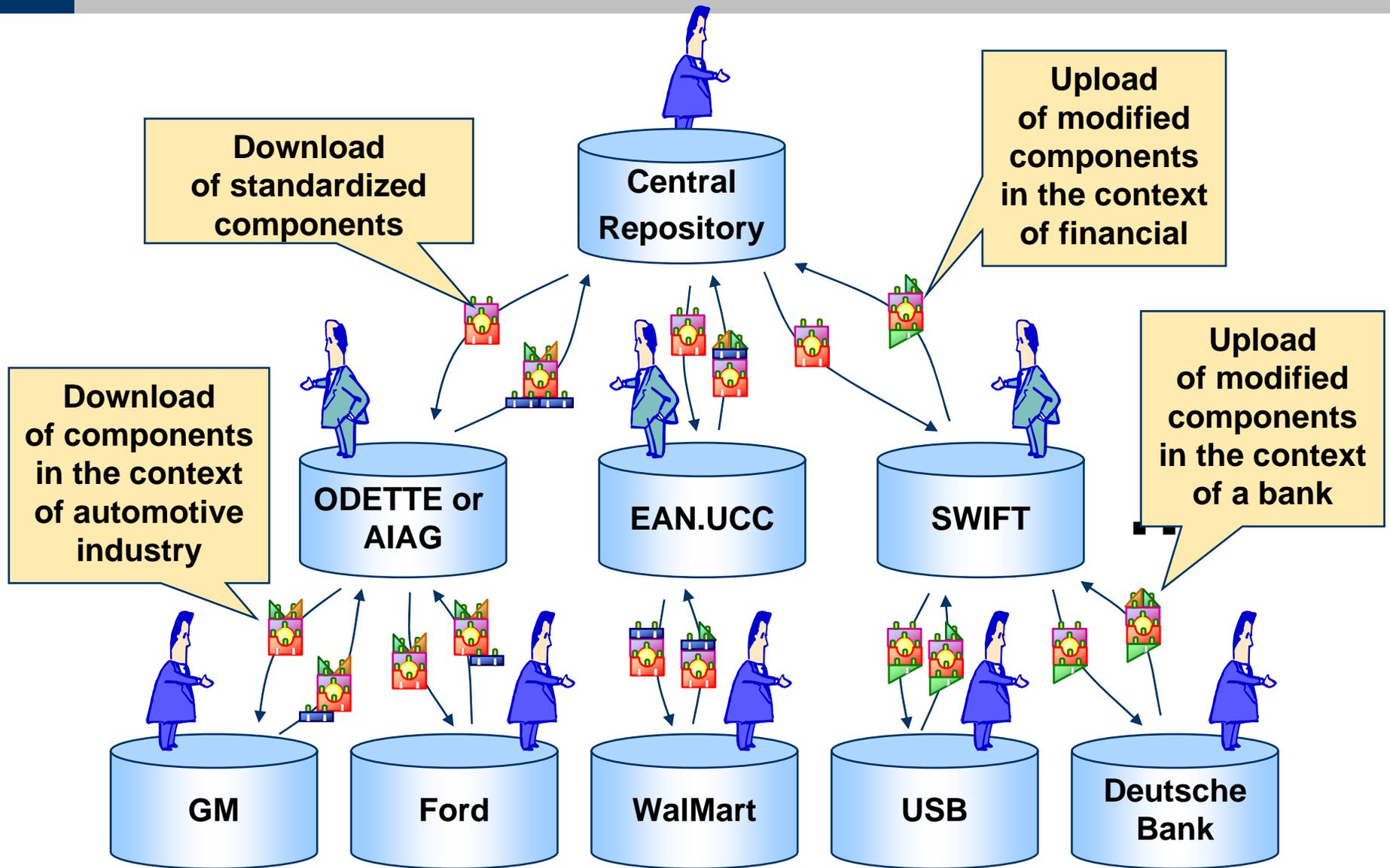
Specific context values of independent User will be added



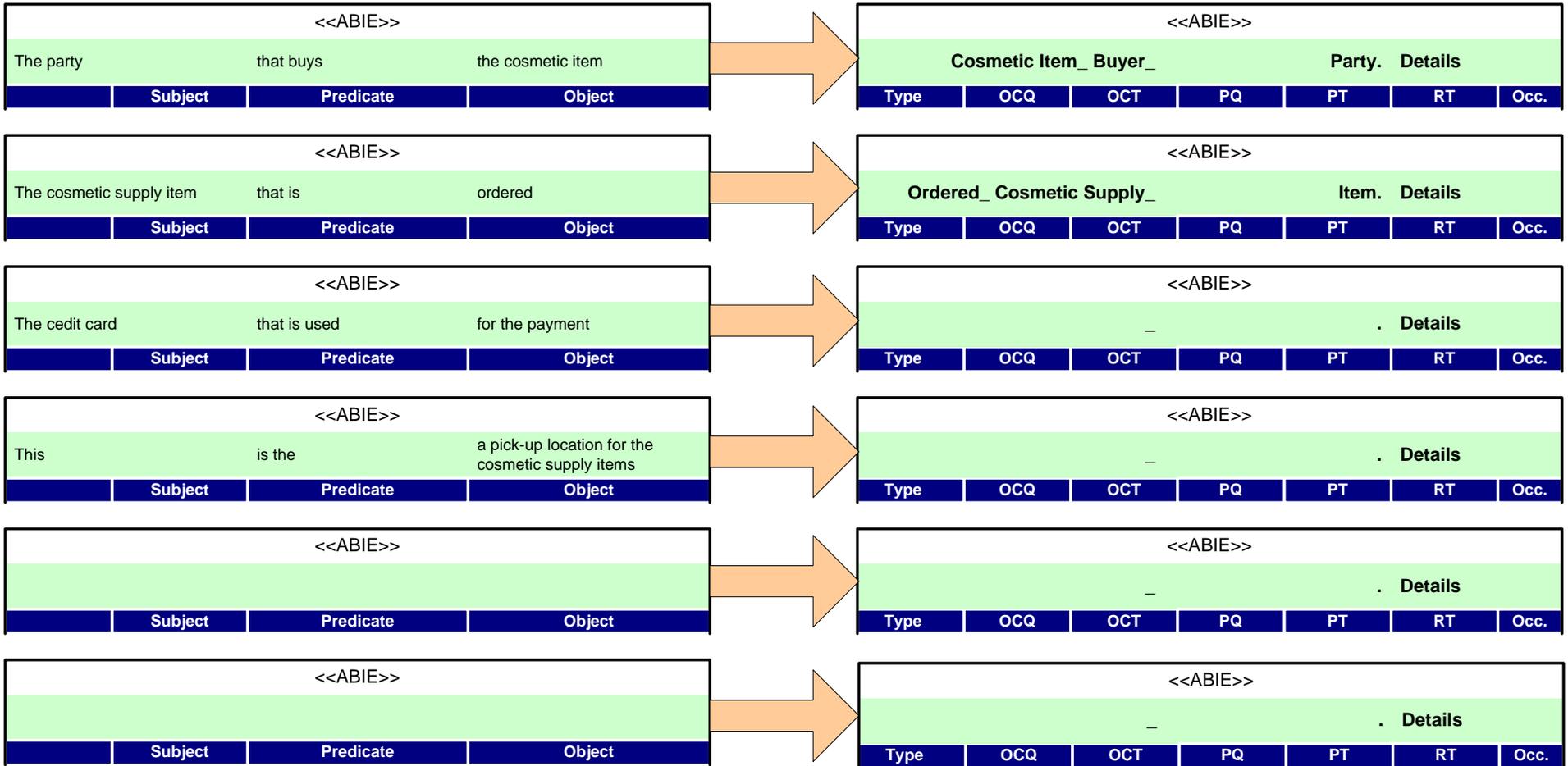
Context	Value
Industry	Cosmetics
Business Role	Supplier
Country	France
Bus. Document	Purchase Order
User	Vichi



Distribution and Harmonization using same Procedure

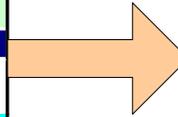


Identify ABIEs



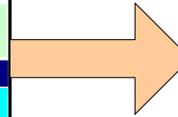
Identify BBIEs

<<ABIE>>						
This details defines the purchase order of cosmetics that is used for a request						
Type	Subject		Predicate		Object	
<<BBIE>>					This date time appoints the creation of requesting purchase order of cosmetics.	
<<BBIE>>					This indicator confirms the requesting purchase order of cosmetics.	



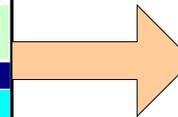
<<ABIE>>						
Requesting_Cosmetics_ Purchase Order. Details						
Type	OCQ	OCT	PQ	PT	RT	Occ.
<<BBIE>>	Requesting_	Purchase Order		Ccreation.	Date Time	[1..1]
<<BBIE>>	Requesting_	Purchase Order		Confirmation.	Indicator	[1..1]

<<ABIE>>						
This details defines the item of a cosmetic supply.						
Type	Subject		Predicate		Object	
<<BBIE>>					This name labels the item of a cosmetic supply.	
<<BBIE>>					This amount is the net price of a cosmetic supply item.	
<<BBIE>>					This amount is the gross price of a cosmetic supply item.	
<<BBIE>>					This amount is the total price of a cosmetic supply item.	



<<ABIE>>						
. Details						
Type	OCQ	OCT	PQ	PT	RT	Occ.
<<BBIE>>						[..]
<<BBIE>>						[..]
<<BBIE>>						[..]
<<BBIE>>						[..]

<<ABIE>>						
This details defines the item of a cosmetic supply.						
Type	Subject		Predicate		Object	
<<BBIE>>						
<<BBIE>>						
<<BBIE>>						
<<BBIE>>						

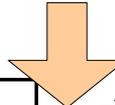


<<ABIE>>						
. Details						
Type	OCQ	OCT	PQ	PT	RT	Occ.
<<BBIE>>						[..]
<<BBIE>>						[..]
<<BBIE>>						[..]
<<BBIE>>						[..]

Identify ASBIEs I

<<ABIE>>						
This details defines the purchase order of cosmetics that is used for a request						
Type	Subject	Predicate	Object	Occ.	RT	PT
<<BBIE>>	This identifier identifies the purchase order of cosmetics that is used for a request.					
<<BBIE>>	This date time appoints the creation of requesting purchase order of cosmetics.					
<<BBIE>>	This indicator confirms the requesting purchase order of cosmetics.					
<<ASBIE>>	This is the bill to party that is defined in the requesting purchase order of cosmetics.					
<<ASBIE>>	This is the deliver to party that is defined in the requesting purchase order of cosmetics.					
<<ASBIE>>	This item is the cosmetic supply of requesting purchase order of cosmetics.					

<<ABIE>>						
This details defines the item of a cosmetic supply.						
Type	Subject	Predicate	Object	Occ.	RT	PT
<<BBIE>>	This name labels the item of a cosmetic supply.					
<<BBIE>>	This amount is the net price of a cosmetic supply item.					
<<BBIE>>	This amount is the gross price of a cosmetic supply item.					
<<BBIE>>	This amount is the total price of a cosmetic supply item.					



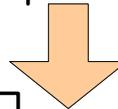
<<ABIE>>						
Requesting_Cosmetics_ Purchase Order. Details						
Type	OCQ	OCT	PQ	PT	RT	Occ.
<<BBIE>>	Requesting_Cosmetics_	Purchase Order.		Identification.	<u>Identifier</u>	[1..1]
<<BBIE>>	Requesting_Cosmetics_	Purchase Order.		Creation.	<u>Date Time</u>	[0..1]
<<BBIE>>	Requesting_Cosmetics_	Purchase Order.		Confirmation.	<u>Indicator</u>	[0..1]
<<ASBIE>>						[1..1]
<<ASBIE>>						[1..1]
<<ASBIE>>	Requesting_Cosmetics_	Purchase Order.	<u>Cosmetic_</u>	<u>Supply_</u>	<u>Item</u>	[1..n]

<<ABIE>>						
Cosmetic_Supply_ Item. Details						
Type	OCQ	OCT	PQ	PT	RT	Occ.
<<BBIE>>	Cosmetic_Supply_	Item.		Label.	<u>Name</u>	[1..1]
<<BBIE>>	Cosmetic_Supply_	Item.	Net_	Price.	<u>Amount</u>	[0..1]
<<BBIE>>	Cosmetic_Supply_	Item.	Gross_	Price.	<u>Amount</u>	[n..n]
<<BBIE>>	Cosmetic_Supply_	Item.	Total_	Price.	<u>Amount</u>	[n..n]

Identify ASBIEs II

<<ABIE>>						
This details defines the purchase order of cosmetics that is used for a request						
Type	Subject	Predicate	Object	Occ.	RT	OCQ
<<BBIE>>	This identifier identifies the purchase order of cosmetics that is used for a request.					
<<BBIE>>	This date time appoints the creation of requesting purchase order of cosmetics.					
<<BBIE>>	This indicator confirms the requesting purchase order of cosmetics.					
<<ASBIE>>	This is the bill to party that is defined in the requesting purchase order of cosmetics.					
<<ASBIE>>	This is the deliver to party that is defined in the requesting purchase order of cosmetics.					
<<ASBIE>>	This item is the cosmetic supply of requesting purchase order of cosmetics.					

<<ABIE>>						
This detail defines the party						
Type	Subject	Predicate	Object	Occ.	RT	OCQ
<<BBIE>>	This identifier identifies the party.					
<<BBIE>>	This name represents the organization					
<<ASBIE>>						



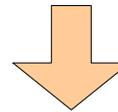
<<ABIE>>						
Requesting_Cosmetics_ Purchase Order. Details						
Type	OCQ	OCT	PQ	PT	RT	Occ.
<<BBIE>>	Requesting_Cosmetics_	Purchase Order.		Identification.	<u>Identifier</u>	[1..1]
<<BBIE>>	Requesting_Cosmetics_	Purchase Order.		Creation.	<u>Date Time</u>	[0..1]
<<BBIE>>	Requesting_Cosmetics_	Purchase Order.		Confirmation.	<u>Indicator</u>	[0..1]
<<ASBIE>>						[1..1]
<<ASBIE>>						[1..1]
<<ASBIE>>	Requesting_Cosmetics_	Purchase Order.	<u>Cosmetic_</u>	<u>Supply.</u>	<u>Item</u>	[1..n]

<<ABIE>>						
Party. Details						
Type	OCQ	OCT	PQ	PT	RT	Occ.
<<BBIE>>						[1..1]
<<BBIE>>						[0..1]
<<ASBIE>>						[n..n]

Identify ASBIEs III

<<ABIE>>						
This detail defines the party						
Type	Subject	Predicate	Object	Occ.		
<<BBIE>>	This identifier identifies the party.					
<<BBIE>>	This name represents the organization					
<<ASBIE>>						

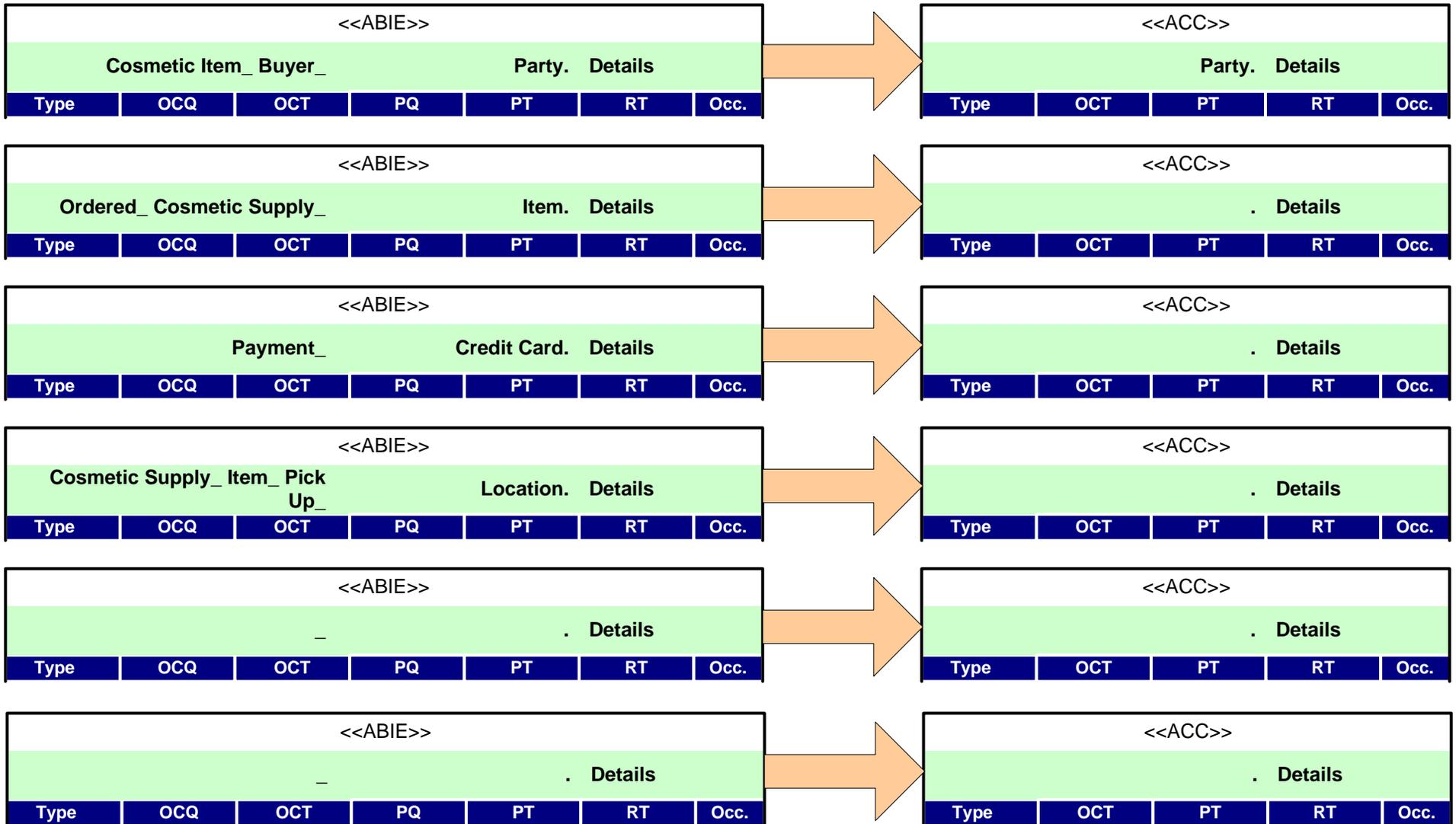
<<ABIE>>						
Type	Subject	Predicate	Object	Occ.		
<<BBIE>>						
<<BBIE>>						
<<ASBIE>>						



<<ABIE>>						
Party. Details						
Type	OCQ	OCT	PQ	PT	RT	Occ.
<<BBIE>>		Party.		Identification.	<u>Identifier</u>	[1..1]
<<BBIE>>		Party.		Organization.	<u>Name</u>	[0..1]
<<ASBIE>>						[..]

<<ABIE>>						
Details						
Type	OCQ	OCT	PQ	PT	RT	Occ.
<<BBIE>>						[..]
<<BBIE>>						[..]
<<ASBIE>>						[..]

Derivation of ACCs by ABIEs.



Derivation of BCCs by BBIEs

<<ABIE>>						
Requesting_Cosmetics_		Purchase Order. Details				
Type	OCQ	OCT	PQ	PT	RT	Occ.
<<BBIE>>	Requesting_Cosmetics_	Purchase Order		Creation.	Date Time	[1..1]
<<BBIE>>	Requesting_Cosmetics_	Purchase Order		Confirmation.	Indicator	[1..1]

<<ACC>>				
Purchase Order. Details				
Type	OCT	PT	RT	Occ.
<<BCC>>	Purchase Order.	Creation.	Date Time	[1..1]
<<BCC>>	Purchase Order.	Confirmation.	Indicator	[1..1]

<<ABIE>>						
Cosmetic Supply_		Item. Details				
Type	OCQ	OCT	PQ	PT	RT	Occ.
<<BBIE>>	Cosmetic_Supply_	Item.		Label.	<u>Name</u>	[1..1]
<<BBIE>>	Cosmetic_Supply_	Item.	Net_	Price.	<u>Amount</u>	[0..1]
<<BBIE>>	Cosmetic_Supply.	Item.	Gross_	Price.	<u>Amount</u>	[n..n]
<<ASBIE>>	Cosmetic_Supply.	Item.	Total_	Price.	<u>Amount</u>	[n..n]

<<ACC>>				
. Details				
Type	OCT	PT	RT	Occ.
<<BCC>>				[n..n]
<<BCC>>				[n..n]

<<ABIE>>						
. Details						
Type	OCQ	OCT	PQ	PT	RT	Occ.
<<BBIE>>						[..]
<<BBIE>>						[..]
<<BBIE>>						[..]
<<BBIE>>						[..]

<<ACC>>				
. Details				
Type	OCT	PT	RT	Occ.
<<BCC>>				[n..n]

Derivation of ASCCs by ASBIEs I

Requesting_Cosmetics_ Purchase Order. Details						
Type	OCQ	OCT	PQ	PT	RT	Occ.
<<BBIE>>	Requesting_Cosmetics_	Purchase Order.		Identification.	Identifier	[1..1]
<<BBIE>>	Requesting_Cosmetics_	Purchase Order.		Creation.	Date Time	[0..1]
<<BBIE>>	Requesting_Cosmetics_	Purchase Order.		Confirmation.	Indicator	[0..1]
<<ASBIE>>	Requesting_Cosmetics_	Purchase Order.		Bill To.	Party	[1..1]
<<ASBIE>>	Requesting_Cosmetics_	Purchase Order.		Deliver To.	Party	[1..1]
<<ASBIE>>	Requesting_Cosmetics_	Purchase Order.	Cosmetic_	Supply.	Item	[1..n]

Cosmetic_Supply_ Item. Details						
Type	OCQ	OCT	PQ	PT	RT	Occ.
<<BBIE>>	Cosmetic_Supply_	Item.		Label.	Name	[1..1]
<<BBIE>>	Cosmetic_Supply_	Item.	Net_	Price.	Amount	[0..1]
<<BBIE>>	Cosmetic_Supply_	Item.	Gross_	Price.	Amount	[n..n]
<<BBIE>>	Cosmetic_Supply_	Item.	Total_	Price.	Amount	[n..n]

Purchase Order. Details				
Type	OCT	PT	RT	Occ.
<<BCC>>	Purchase Order.	Identification.	Identifier	[1..1]
<<BCC>>	Purchase Order.	Creation.	Date Time	[0..1]
<<BCC>>	Purchase Order.	Identification.	Identifier	[0..1]
<<BCC>>				[1..1]
<<BCC>>				[1..1]
<<ASCC>>	Purchase Order.	Supply.	Item	[n..n]

Item. Details				
Type	OCT	PT	RT	Occ.
<<BCC>>	Item.	Label.	Name	[1..1]
<<BCC>>	Item.	Price.	Amount	[0..1]

Derivation of ASCCs by ASBIEs II

<<ABIE>>						
Requesting_Cosmetics_			Purchase Order. Details			
Type	OCQ	OCT	PQ	PT	RT	Occ.
<<BBIE>>	Requesting_Cosmetics_	Purchase Order.		Identification.	<u>Identifier</u>	[1..1]
<<BBIE>>	Requesting_Cosmetics_	Purchase Order.		Creation.	<u>Date Time</u>	[0..1]
<<BBIE>>	Requesting_Cosmetics_	Purchase Order.		Confirmation.	<u>Indicator</u>	[0..1]
<<ASBIE>>	Requesting_Cosmetics_	Purchase Order.		Bill To.	<u>Party</u>	[0..1]
<<ASBIE>>	Requesting_Cosmetics_	Purchase Order.		Deliver To.	<u>Party</u>	[0..1]
<<ASBIE>>	Requesting_Cosmetics_	Purchase Order.	<u>Cosmetic_</u>	<u>Supply_</u>	<u>Item</u>	[1..n]

<<ABIE>>						
Cosmetic_Supply_			Item. Details			
Type	OCQ	OCT	PQ	PT	RT	Occ.
<<BBIE>>	Cosmetic_Supply_	Item.		Label.	<u>Name</u>	[1..1]
<<BBIE>>	Cosmetic_Supply_	Item.	Net_	Price.	<u>Amount</u>	[0..1]
<<BBIE>>	Cosmetic_Supply_	Item.	Gross_	Price.	<u>Amount</u>	[n..n]
<<BBIE>>	Cosmetic_Supply_	Item.	Total_	Price.	<u>Amount</u>	[n..n]

<<ACC>>				
Purchase Order. Details				
Type	OCT	PT	RT	Occ.
<<BCC>>	Purchase Order.	Identification.	Identifier	[1..1]
<<BCC>>	Purchase Order.	Creation.	Date Time	[0..1]
<<BCC>>	Purchase Order.	Identification.	Identifier	[0..1]
<<BCC>>				[0..1]
<<BCC>>				[0..1]
<<ASCC>>	Purchase Order.	Supply.	Item	[n..n]

<<ACC>>				
Item. Details				
Type	OCT	PT	RT	Occ.
<<BCC>>	Item.	Label.	Name	[1..1]
<<BCC>>	Item.	Price.	Amount	[0..1]

Derivation of ASCCs by ASBIEs III

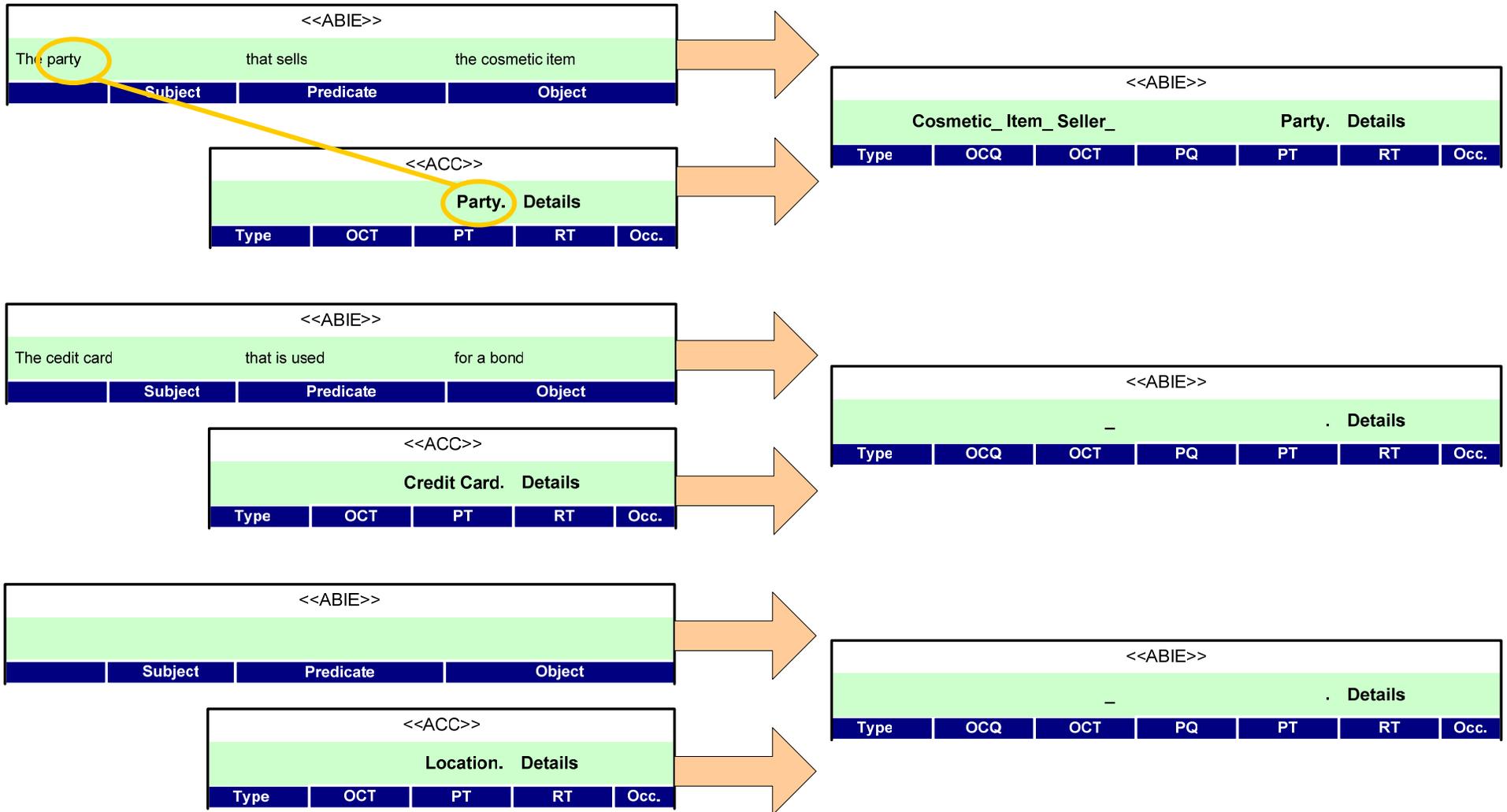
<<ABIE>>						
Party. Details						
Type	OCQ	OCT	PQ	PT	RT	Occ.
<<BBIE>>		Party.		Identification.	<u>Identifier</u>	[1..1]
<<BBIE>>		Party.		Organization.	<u>Name</u>	[0..1]
<<ASBIE>>						[..]

<<ABIE>>						
Details						
Type	OCQ	OCT	PQ	PT	RT	Occ.
<<BBIE>>						[..]
<<BBIE>>						[..]
<<ASBIE>>						[..]

<<ACC>>				
Party. Details				
Type	OCT	PT	RT	Occ.
<<BCC>>	Party.	Identification.	Identifier	[1..1]
<<BCC>>	Party.	Organization.	Name	[0..1]
<<ASCC>>				[..]

<<ACC>>				
. Details				
Type	OCT	PT	RT	Occ.
<<BCC>>				[..]
<<BCC>>				[..]
<<ASCC>>				[..]

Define ABIEs by ACCs



Define BBIEs by BCCs

«ABIE»					
This details defines the purchase order of spare parts that is used for a response					
Type	Subject	Predicate	Object		
«BBIE»	This date time appoints the creation of responding purchase order of spare parts.				
«BBIE»	This indicator confirms the responding purchase order of spare parts.				

«ACC»				
Purchase Order. Details				
Type	OCT	PT	RT	Occ.
«BCC»	Purchase Order.	Creation.	Date Time	[1..1]
«BCC»	Purchase Order.	Confirmation.	Indicator	[1..1]

«ABIE»						
Responding_ Spare Parts_ Purchase Order. Details						
Type	OCQ	OCT	PQ	PT	RT	Occ.
«BBIE»	Responding_ Space Parts_	Purchase Order		Creation.	Date Time	[1..1]
«BBIE»	Responding_ Spare Parts_	Purchase Order		Confirmation.	Indicator	[1..1]

«ABIE»					
This details defines the item of a spare part..					
Type	Subject	Predicate	Object		
«BBIE»	This name labels the item of a spare part.				
«BBIE»	This amount is the calculated price of a spare part item.				
«BBIE»	This amount is the net price of a spare part item.				
«BBIE»	This amount is the booked price of a spare part item.				

«ACC»				
Item. Details				
Type	OCT	PT	RT	Occ.
«BCC»	Item.	Label.	Name	[1..1]
«BCC»	Item.	Price.	Amount	[0..1]

«ABIE»						
Details						
Type	OCQ	OCT	PQ	PT	RT	Occ.
«BBIE»						[1..1]
«BBIE»						[..]
«BBIE»						[..]
«ASBIE»						[..]

Define ASBIEs by ASCCs I

<<ABIE>>

This details defines the purchase order of spare parts that is used for a response

Type	Subject	Predicate	Object
<<BBIE>>	This identifier identifies the purchase order of spare part that is used in a response.		
<<BBIE>>	This date time appoints the creation of responding purchase order of spare parts.		
<<BBIE>>	This indicator confirms the responding purchase order of spare parts.		
<<ASBIE>>	This is the changed deliver to party that is defined in the responding purchase order of spare parts.		
<<ASBIE>>	This item the rejected spare part of a responding purchase order of spare parts.		

<<ABIE>>

This details defines the item of a spare part.

Type	Subject	Predicate	Type	Object
<<BBIE>>	This name labels the item of a spare part.			
<<BBIE>>	This amount is the calculated price of a spare part item.			
<<BBIE>>	This amount is the net price of a spare part item.			
<<BBIE>>	This amount is the booked price of a spare part item.			

<<ACC>>

Purchase Order. Details

Type	OCT	PT	RT	Occ.
<<BCC>>	Purchase Order.	Identification.	Identifier	[1..1]
<<BCC>>	Purchase Order.	Creation.	Date Time	[0..1]
<<BCC>>	Purchase Order.	Identification.	Identifier	[0..1]
<<BCC>>	Purchase Order.	Bill To.	Party	[0..1]
<<BCC>>	Purchase Oder.	Deliver To.	Pary	[0..1]
<<ASCC>>	Purchase Order.	Supply.	Item	[0..n]
<<ASCC>>	Purchase Order.	Spare Part.	Item	[0..n]

<<ACC>>

Item. Details

Type	OCT	PT	RT	Occ.
<<BCC>>	Item.	Label.	Name	[1..1]
<<BCC>>	Item.	Price.	Amount	[0..1]

<<ABIE>>

Responding_Spare Parts_ Purchase Order. Details

Type	OCQ	OCT	PQ	PT	RT	Occ.
<<BBIE>>	Responding_Spare Parts_	Purchase Order.	Identification.	Identifier		[1..1]
<<BBIE>>	Responding_Spare Parts_	Purchase Order.	Creation.	Date Time		[0..1]
<<BBIE>>	Responding_Spare Parts_	Purchase Order.	Confirmation.	Indicator		[0..1]
<<ASBIE>>						[1..1]
<<ASBIE>>	Responding_Spare Parts_	Purchase Order.	Rejected_Spare Part.	Item		[1..n]

<<ABIE>>

Spare Part_Supply_ Item. Details

Type	OCQ	OCT	PQ	PT	RT	Occ.
<<BBIE>>	Spare Part_Supply_	Item.	Label.	Name		[1..1]
<<BBIE>>	Cosmetic_Supply_	Item.	Calculated_	Price.	Amount	[0..1]
<<BBIE>>	Cosmetic_Supply_	Item.	Net_	Price.	Amount	[n..n]
<<BBIE>>	Cosmetic_Supply_	Item.	Booked_	Price.	Amount	[n..n]



Define ASBIEs by ASCCs II

<<ABIE>>

This details defines the purchase order of cosmetics that is used for a request

Type	Subject	Predicate	Object
<<BBIE>>	This identifier identifies the purchase order of cosmetics that is used for a request.		
<<BBIE>>	This date time appoints the creation of requesting purchase order of cosmetics.		
<<BBIE>>	This indicator confirms the requesting purchase order of cosmetics.		
<<ASBIE>>	This is the bill to party that is defined in the requesting purchase order of cosmetics.		
<<ASBIE>>	This is the deliver to party that is defined in the requesting purchase order of cosmetics.		
<<ASBIE>>	This item is the cosmetic supply of requesting purchase order of cosmetics.		

<<ABIE>>

This detail defines the party

Type	Subject	Predicate	Object	Occ.
<<BBIE>>	This identifier identifies the party.			
<<BBIE>>	This name represents the organization			
<<ASBIE>>	This address is the business address.			

<<ACC>>

Purchase Order. Details

Type	OCT	PT	RT	Occ.
<<BCC>>	Purchase Order.	Identification.	Identifier	[1..1]
<<BCC>>	Purchase Order.	Creation.	Date Time	[0..1]
<<BCC>>	Purchase Order.	Identification.	Identifier	[0..1]
<<BCC>>	Purchase Order.	Bill To.	Party	[0..1]
<<BCC>>	Purchase Oder.	Deliver To.	Pary	[0..1]
<<ASCC>>	Purchase Order.	Supply.	Item	[0..n]
<<ASCC>>	Purchase Order.	Spare Part.	Item	[0..n]

<<ACC>>

Party. Details

Type	OCT	PT	RT	Occ.
<<BCC>>	Party.	Identification.	Identifier	[1..1]
<<BCC>>	Party.	Organization.	Name	[0..1]
<<ASCC>>	Party.	Business.	Address	[0..1]

<<ABIE>>

Responding_Spare Parts_ Purchase Order. Details

Type	OCQ	OCT	PQ	PT	RT	Occ.
<<BBIE>>	Responding_Spare Parts_	Purchase Order.	Identification.		Identifier	[1..1]
<<BBIE>>	Responding_Spare Parts_	Purchase Order.		Creation.	Date Time	[0..1]
<<BBIE>>	Responding_Spare Parts_	Purchase Order.	Confirmation.		Indicator	[0..1]
<<ASBIE>>						[1..1]
<<ASBIE>>	Responding_Spare Parts_	Purchase Order.	Rejected_ Spare Part.		Item	[1..n]

<<ABIE>>

. Details

Type	OCQ	OCT	PQ	PT	RT	Occ.
<<BBIE>>						[1..1]
<<BBIE>>						[0..1]
<<ASBIE>>						[..]

The Data Interoperability Dilemma

ISO 15000-5 Core Components

Process Models, CCTS, and Syntax

Concepts In Action

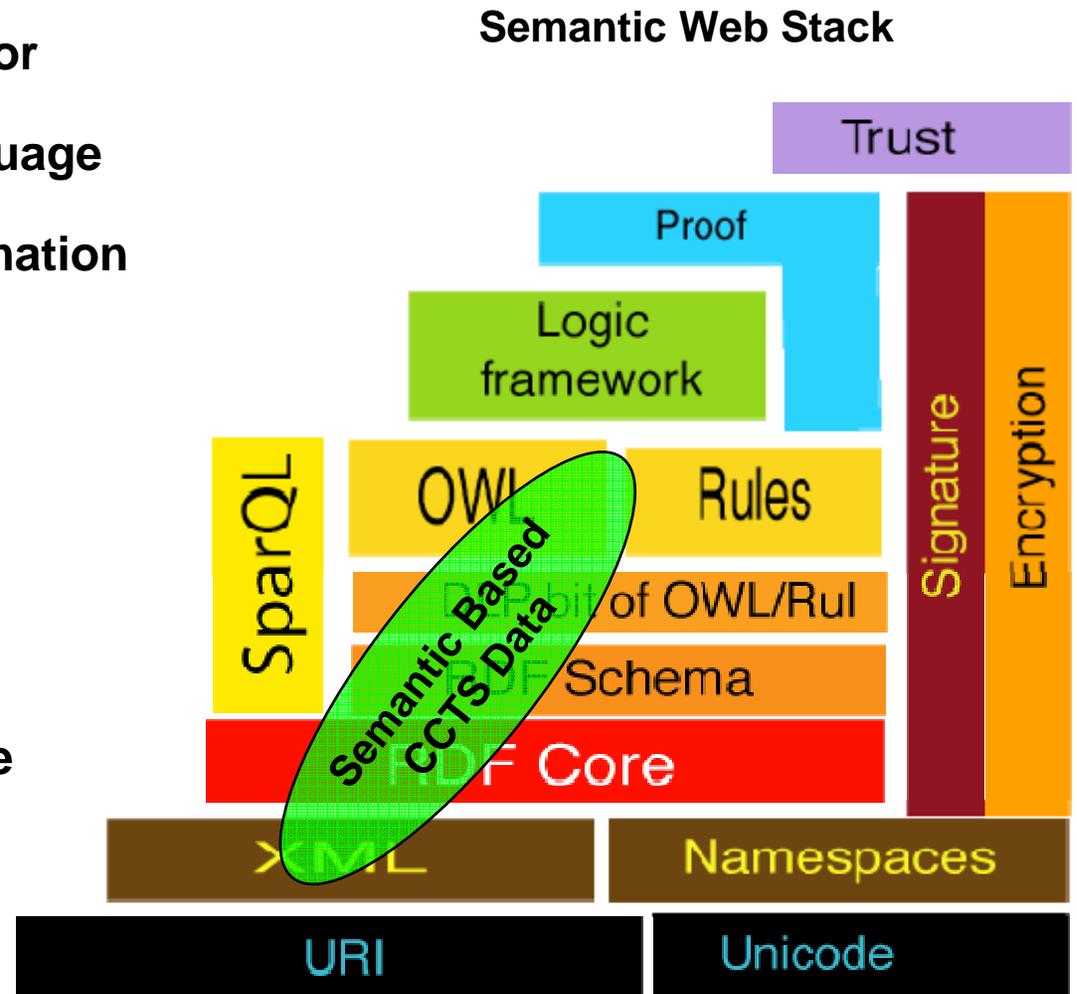
CCTS and The Semantic Web

CCTS in SAP

Synergy Between Core Components, OWL, RDF and Semantic Web

- RDF is Semantic Web locator
- OWL is Semantic Web language
- XML is expression of information

- CCTS can be heavy lifter at database level for easing burdens on XML/RDF/OWL
- Consistency to data structures, meaning and use
- Consistency in metadata



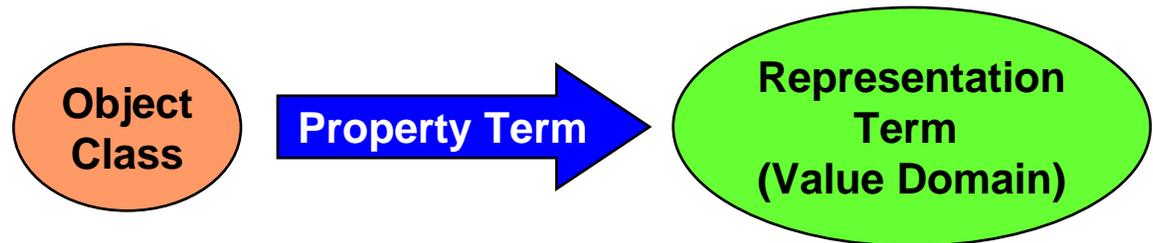
Courtesy W3C

Core Components & RDF Alignment

■ RDF Triple



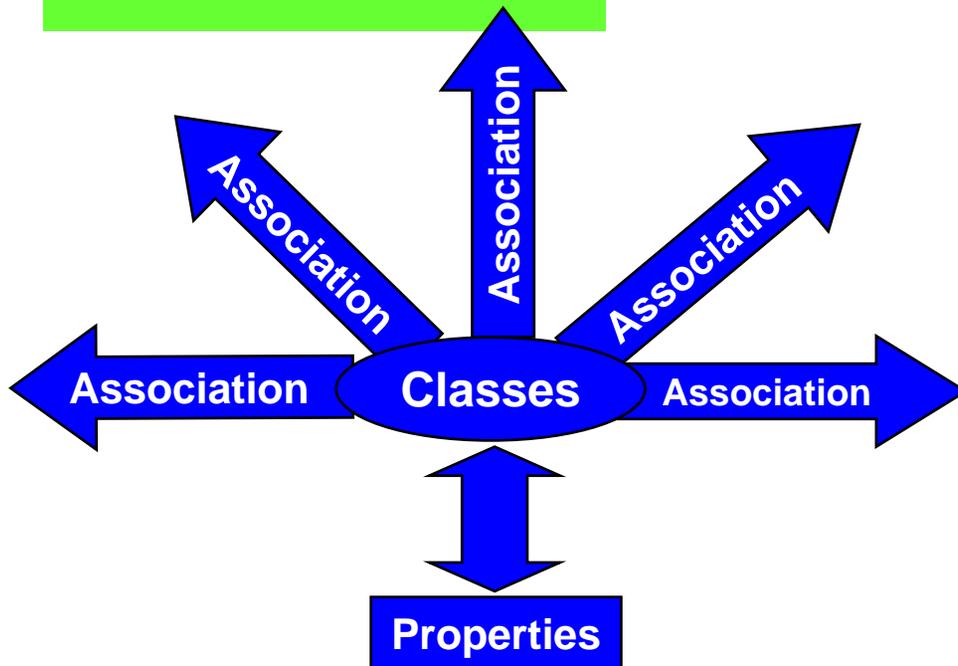
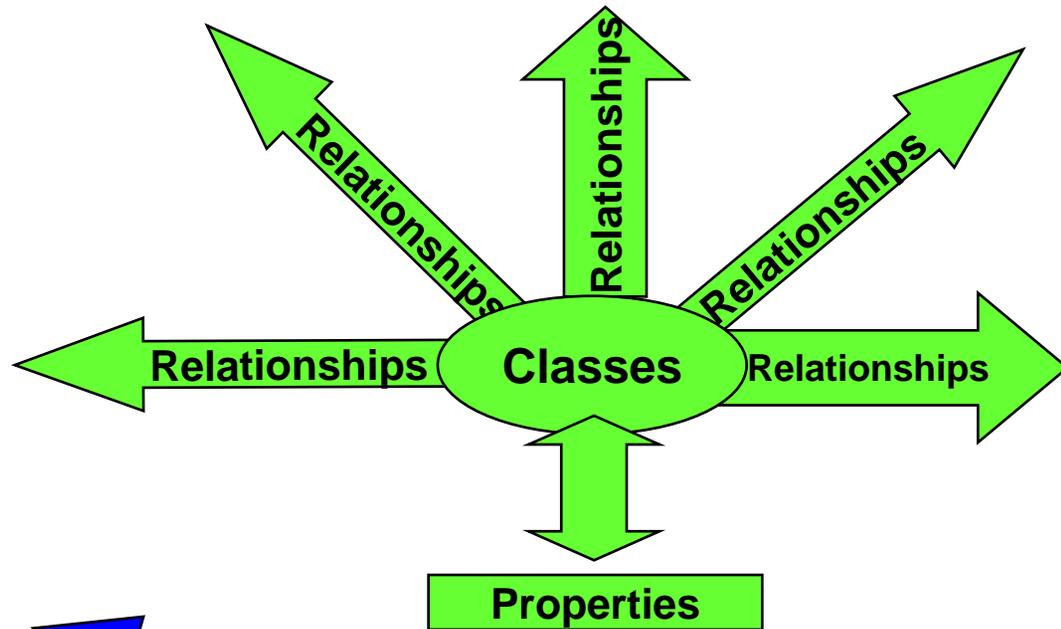
■ Core Component Triple



Core Components and OWL Alignment

OWL Triples

- **Classes** (general things) in the many domains of interest
- The relationships that can exist among things
- The properties (or attributes) those things may have



CC Triples

- **Object Class**
- **Associations**
- **Properties (Property Term + Representation Term)**

The Data Interoperability Dilemma

ISO 15000-5 Core Components

Process Models, CCTS, and Syntax

Concepts In Action

CCTS and The Semantic Web

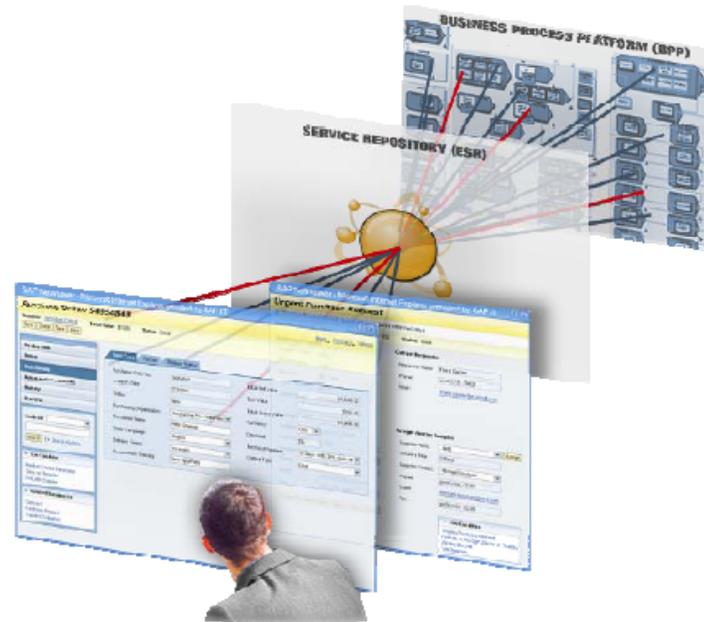
CCTS in SAP

Take The Business Perspective

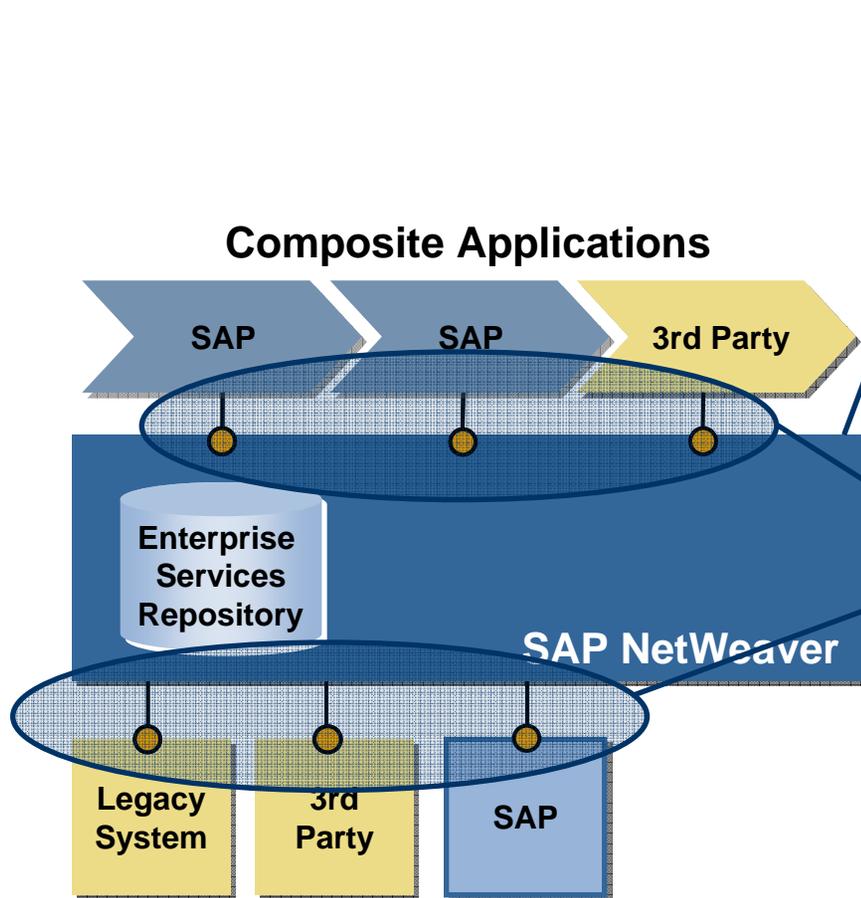
SOA IS REQUIRED BUT NOT ENOUGH

Common Business Semantics are required

$$\begin{array}{r} \text{SOA} \\ + \\ \text{ES} \\ \hline \text{ESA} \end{array}$$



Enterprise Services (Platform View)



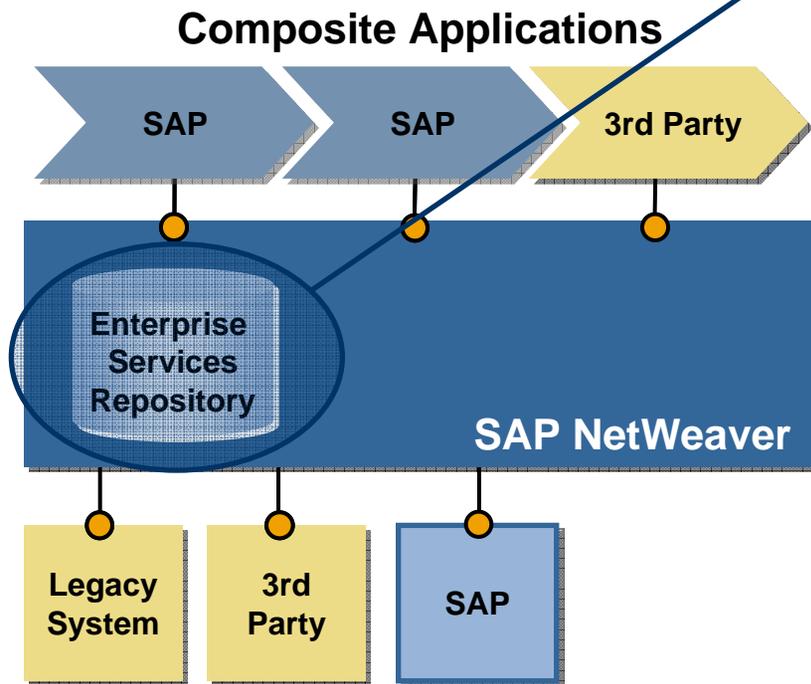
Enterprise Standards

- Reliability, scalability, performance, security
- Driven by business and legal requirements
- Mostly user-specific
- Examples: throughput of 100.000 invoices per hour, Section 508 accessibility

Web Services Standards

- SOA using Internet standards (XML, HTTP, etc.)
- Driven by interoperability needs with partner platforms (e.g. IBM WebSphere, Microsoft .NET)
- Covered by W3C, OASIS, WS-I
- Examples: WSDL, SOAP, WS-Security, WS-ReliableMessaging

Introducing Standards for Standards



Development Methodology Standards

- Modeling and design rules
- Driven by development and integration efficiency and quality needs
- Covered by OMG, UN/CEFACT
- Examples: UML, XMI, CCTS, XML NDR

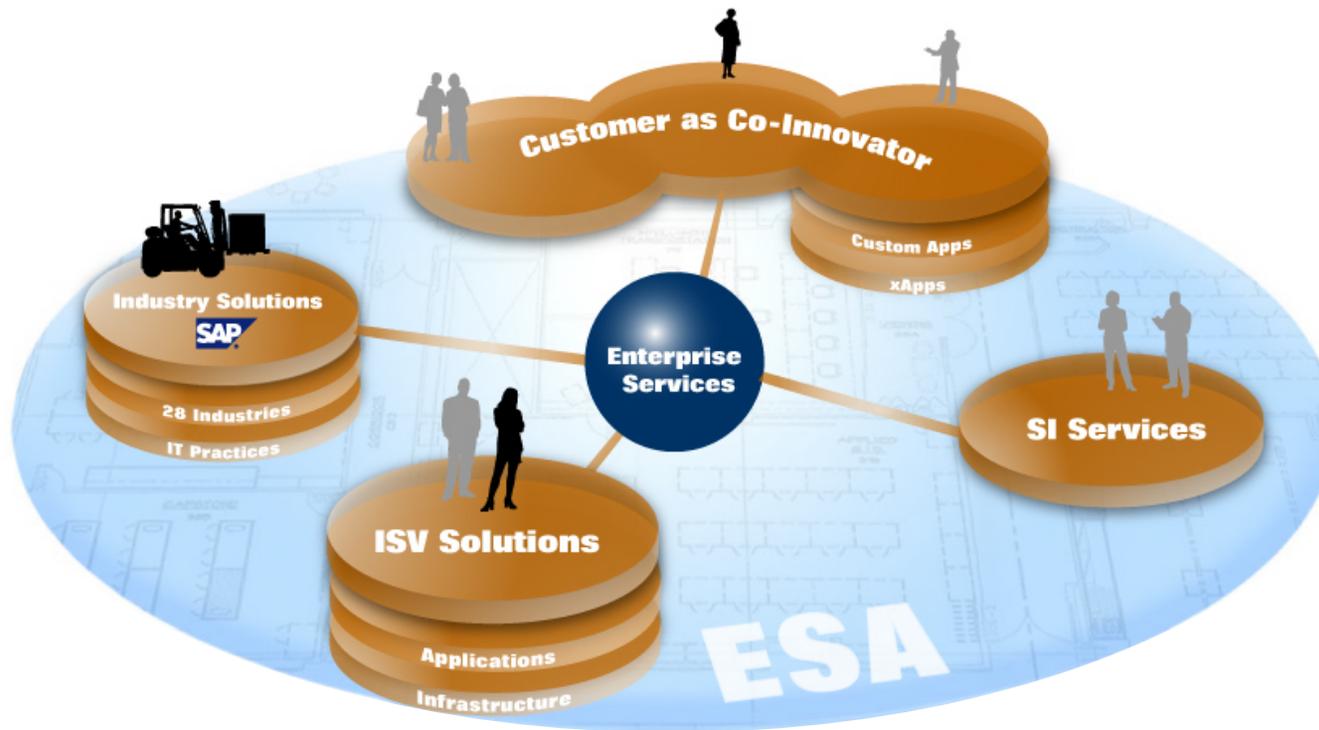
→ Effectively enhance reuse and address the Business Standards Dilemma

→ SAP is leveraging ISO 15000-5 CCTS and XML NDR for the semantic-based definition of Enterprise Services

→ CCTS For Global Data Types

→ XML NDR for XML internal and External expressions

Platform Ecosystem



Collective Innovation

- Customers, Partners, ISVs & SAP
- Develop and prioritize Enterprise Service Definitions

Based on Open Standards

- Interoperability: Web Services
- Methodology: CCTS, etc.



Questions?

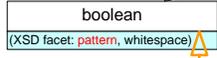




UN/CEFACT DataType system

Author: Anders W. Tell, Financial Toolsmiths AB, Version: 0.5

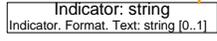
Primitive types



Enumeration types



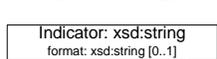
Structured types



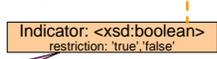
Core Components Technical Specification

Technology Neutral W3C XML Schema

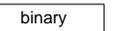
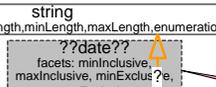
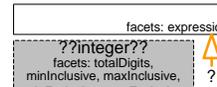
Core Component Types



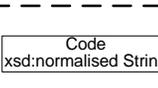
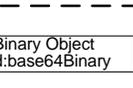
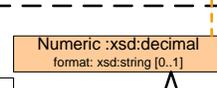
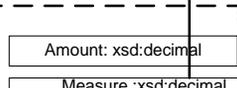
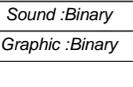
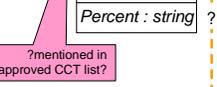
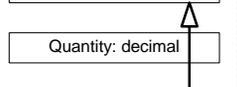
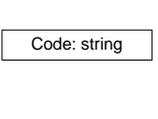
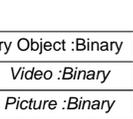
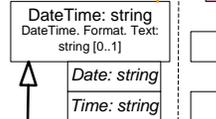
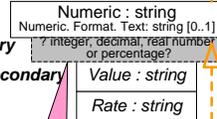
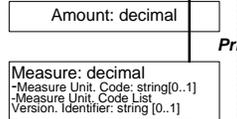
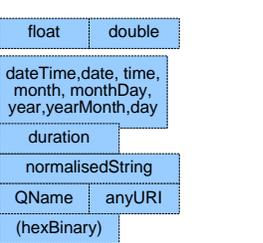
Unqualified DataTypes



Qualified DataTypes



Proposed additions



W3C XSD

XSD Primitive datatypes (19):
 string, boolean, decimal, float, double, duration, dateTime, time, date, gYearMonth, gYear, gMonthDay, gDay, gMonth, hexBinary, base64Binary, anyURI, QName, NOTATION

XSD Derived datatypes (25):

<string>
 normalisedString, token, language, NMTOKEN, NMTOKENS, Name, NCName, ID, IDREF, IDREFS, ENTITY, ENTITIES

<decimal>
 integer, nonPositiveInteger, negativeInteger, long, int, short, byte, nonNegativeInteger, unsignedLong, unsignedInt, unsignedShort, unsignedByte, positiveInteger

Fundamental Facets (5):

equal, ordered, bounded, cardinality, numeric

Constraining Facets (12):
 length, minLength, maxLength, pattern, enumeration, whiteSpace, maxInclusive, maxExclusive, minInclusive, minExclusive, totalDigits, fractionDigits

?applied format/pattern restriction?
 ?reverse algorithm to find correct supplementary component?
 ? Change in attribute Datatype?
 ? Change in cardinality?
 ?missing attribute?

!! Add downcast restriction in order to be precise and allow mapping from XML to CCTS!!



- No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.
- Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.
- Microsoft, Windows, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.
- IBM, DB2, DB2 Universal Database, OS/2, Parallel Sysplex, MVS/ESA, AIX, S/390, AS/400, OS/390, OS/400, iSeries, pSeries, xSeries, zSeries, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, and Informix are trademarks or registered trademarks of IBM Corporation in the United States and/or other countries.
- Oracle is a registered trademark of Oracle Corporation.
- UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.
- Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.
- HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.
- Java is a registered trademark of Sun Microsystems, Inc.
- JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.
- MaxDB is a trademark of MySQL AB, Sweden.
- SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.
- These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.