

# Your presenter...Baba Piprani, SICOM Canada

---

- Senior IT Consultant with over 30 yrs standardization experience...Computer Languages, SQL, Conceptual Schema, Data Modelling, IRDS, Metadata Registry, MOF...
- Developed award winning implementations of standards-based Data Quality Firewalls with advanced generation architecture data warehouses and Web based applications using SBVR, ORM, NIAM, Master Data Management, Metadata Repositories/Registries using SQL DBMSs...
- Clients: Canadian Government departments Transport, Foreign Affairs, Defence, Superintendent of Financial Systems, Public Works...including private sector, and CNIS (China National Institute for Standardization)
- Working with Donald Chapin, John Hall and Sjir Nijssen in the progression and advancement of SBVR

**A Common  
Terminology, Semantic Metadata & Data Model Framework  
for Relating  
SBVR\*, ISO 704 & 1087 to ISO/IEC 19763 & 11179  
*Open Forum 2008***

Donald Chapin

Business Semantics Ltd

United Kingdom

Co-Chair, OMG SBVR  
Revision Task Force

[Donald.Chapin@BusinessSemantics.com](mailto:Donald.Chapin@BusinessSemantics.com)

John Hall

Model Systems Ltd

United Kingdom

Co-Chair, OMG  
Regulatory Compliance  
SIG

[john.hall@modelsys.com](mailto:john.hall@modelsys.com)

Sjir Nijssen

PNA Group

The Netherlands

Member OMG SBVR  
Revision Task Force

[sjir.nijssen@pna-group.nl](mailto:sjir.nijssen@pna-group.nl)

Baba Piprani

SICOM Canada

ISO IEC/ JTC1 SC32  
WG2 Metadata and WG3  
Database Languages

[babap@attglobal.net](mailto:babap@attglobal.net)

**\*The Object Management Groups's new "Semantics of Business Vocabulary & Business Rules" specification**

# Context and Scope of This Presentation

---

- This presentation is about the **bridge** between:
  - the business model – especially the special purpose language (*terminology*) that the people who run organizations
    - use when they think, talk and write about
      - what they have to deal with during the course of their work, -- and
  - implemented information systems
- Semantic communities addressed are those centred around organizations
- This presentation is focused quite narrowly on only one:
  - use of terminology –
    - for specifying business data/rules requirements and designing relational databases
  - aspect of bridging from the business model to an information system –
    - the WHAT, some HOW and WHEN (*but not focusing on the WHERE, WHO or WHY*)
  - kind of metadata –
    - **semantic metadata**: data about the **meaning** of data

**NOTE:** For all uses of terminology and SBVR see “SBVR: What is Now Possible and Why?” (<http://www.brcommunity.com/a2007/b407.html>)

# TOPICS

---

1. Need for Common Understanding
2. Terminology Science (*Terminology*) & Information Science (*Metadata & Data Models*)
3. Architecture for Transforming Terminology to Semantic Metadata & Data Models
  - based on the OMG's new MDA Foundation Model as applied to both the Organization and its Business Application Software
4. Transforming Terminology to Semantic Metadata & Data Models
  1. Business Ontology (A)  $\leftrightarrow$  Consolidated Data & Rules Requirements (B)
  2. Business Requirements for Software (B)  $\leftrightarrow$  Class-of-Platform Independent Data Model (C)
  3. Class-of-Platform Independent Data Model (C)  $\leftrightarrow$  Class-of-Platform Specific Data Model (D)
  4. Class-of-Platform Specific Data Model (D)  $\leftrightarrow$  Vendor Platform Specific Data Model (E)
5. Using SBVR to Provide Business Semantics for Semantic Metadata & Data (ISO 11179 & 19763)

# 1. Problem Statement: Lack of Common Understanding

---

- There is widespread confusion about the differences between
  - **terminology** as understood in terminology science, and
  - **semantic metadata** and **data modelling** as understood in information science.
- This is particularly relevant to the OMG because the vocabulary part of SBVR is a terminology standard made rigorous in formal logic by fact-oriented modeling.
- The confusion is seen in observed problems:
  - The disagreements in the ISO community over how ISO TC 37 Terminology standards relate to ISO/IEC JTC1 SC32 WG2 Metadata standards.
  - Attempts to integrate SBVR with data modeling specifications (*e.g. E/R, OMG CWM, OMG IMM, etc.*) without the necessary transforms.
  - Questions about how SBVR relates to:
    - Fact-oriented Modelling (*e.g. ORM, CogNIAM, etc.*)
    - UML
    - W3C's RDF and OWL
  - Direct reverse engineering of data models into the SBVR structures without adding in the SBVR semantics not found in data models;
    - i.e. not collecting or documenting the meanings of the terms as used by the people who run the organization to communicate, verbally and in writing, and not providing the transform mappings between the data structures and the terminology concepts.

# Keys to a Common Understanding

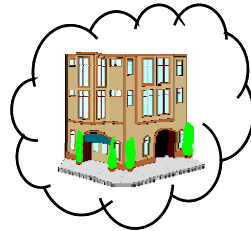
---

- Understanding the ***differences*** and ***inter-connections*** between:
  - Terminology
    - as documentation of ***concepts*** shared among the minds of the people in some semantic community and used in their communication
  - and
  - Semantic Metadata & Data Models
    - as documentation of ***facts and rules about***
      - » ***Data***, and
      - » ***Structures for recording/storing data***, respectively
  - Models of the organization (*a social system*)
  - and
  - Models of the IT system (*a mechanical system*)
  - the referenced standards, the purpose and strength of each of, and how their metamodel elements correlate
- Understanding:
  - the different kinds of models that bridge from Terminology to Implemented Databases
  - the transforms required to get from one to the other

## 2. Terminology Science and Information Science – Different Kinds of Documentation Artefacts

### Terminology

car, business unit  
rental car: car that is owned by EU-Rent and is used for rentals  
branch: business unit that stores cars and rents cars  
rental car is available  
branch has location  
car is stored at branch  
car is of model



### Semantic Metadata (as in ISO 19763 & 11179 with business semantics supplied by SBVR)

#### By Car Movement

Car Movement ID  
 (movement-id)

Car Group Name

(group name that designates car group that is specified by car movement that is of movement-id)

#### By Branch

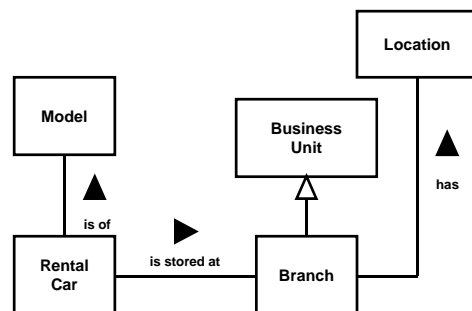
Sending Branch ID

(role of rental organization unit id that is of 'rental organization unit id designates sending branch')

Receiving Branch ID

(role of rental organization unit id that is of 'rental organization unit id designates receiving branch')

### Data Model (a model of structures for recording/storing data)



**Model**  
 Model\_id: char (8)  
 model\_name: char (24)  
 manufacturer: char (30)  
 body\_style: char (12)  
 fuel\_type: char (12)  
 engine\_size: int  
 ...

**Rental Car**  
 VIN: char (36)  
 purchase: date  
 mileage: int  
 last\_service: date  
 colour: char(12)  
 status: char (12)  
 ...

### Recorded Data (using a modeled structure for data)

RENTAL CAR					
Vehicle Identification Number	Purchase Date	Mileage Driven	Last Service Date	External Colour	Rental Status
123456	2007-11-3	13,450	2008-03-30	Tan	Ready to Rent
etc.					

## 2. Terminology Science and Information Science – Different Kinds of Documentation Artefacts

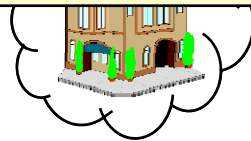
### Terminology

car, business unit  
rental car: car that is owned by  
 EU-Rent and is used for  
 rentals  
branch: business unit that stores



## Terminology Science

car is stored at branch  
car is of model



**Semantic Metadata** (as in ISO 19763 & 11179 with  
 business semantics supplied by SBVR)

### By Car Movement

Car Movement ID  
 (movement-id)

Car Group Name

(group name that designates car group that is  
 specified by car movement that is of movement-id)

### By Branch

Sending Branch ID

(role of rental organization unit id that is of 'rental  
 organization unit id designates sending branch' )

Receiving Branch ID

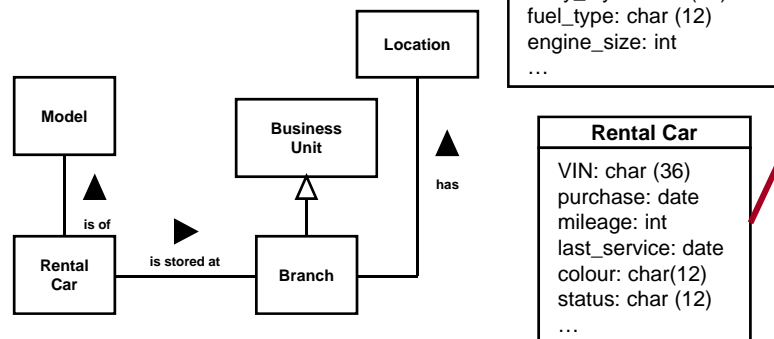
(role of rental organization unit id that is of 'rental

## Information Science

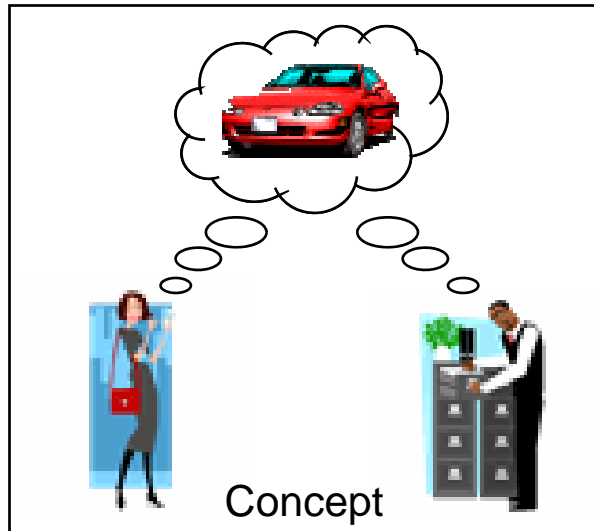
**Recorded Data** (using a modeled structure for data)

RENTAL CAR					
Vehicle Identification Number	Purchase Date	Mileage Driven	Last Service Date	External Colour	Rental Status
123456	2007-11-3	13,450	2008-03-30	Tan	Ready to Rent
etc.					

**Data Model** (a model of structures  
 for recording/storing data)



# Concepts, Representations, Data, Metadata & Semantic Metadata



“a wheeled automotive vehicle designed for passenger transportation on streets and roadways and commonly propelled by an internal-combustion engine using a volatile fuel (as gasoline); especially a private passenger vehicle as distinguished from a bus or truck” [MWUD]

Representation of concept (US English)

1FMZU73E04ZA69101, Explorer, 20071013, 6854

Data about the thing denoted by individual concept

Metadata

CAR			
VIN	model	purchase_date	mileage

VIN (Vehicle Identification Number) is fixed length 17 characters, drawn from the English alphabet (excluding I, 0 and Q) and digits 0 through 9

Semantic Metadata

VIN encoding is specified in ISO 3779 and 3780

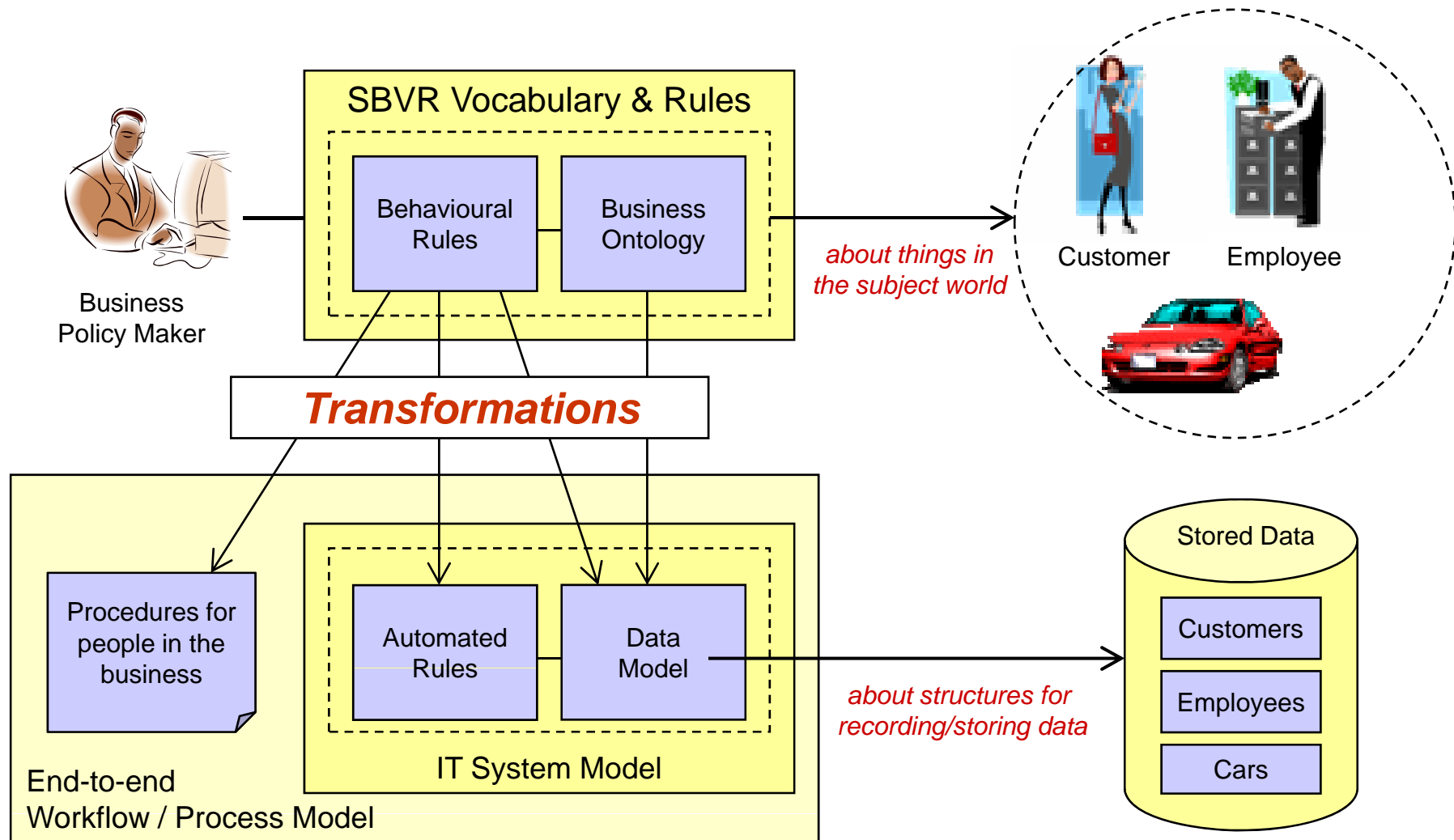
**VIN:** *role of VIN that is of 'VIN designates car'* (data element definition in terms of SBVR)

Although the ISO VIN format is used, the USA and the EU have adopted different encodings for VINs issued in their jurisdictions

# Contents of Terminology, Semantic Metadata, and Data Models – Each Documents Different Things

- Things documented by the contents of:
  - **Terminology:** Concepts and Representations (*of real things*) ONLY
    - i.e. Concepts and representations in minds of, and shared among, people in a subject field, discipline, profession, community, or organization unit
      - e.g. ‘car’ → concept: “wheeled automotive vehicle designed for passenger transportation on roads ...”
  - **Semantic Metadata:** Meaning of Data ONLY (*usually in terms of data elements and data values*)
    - e.g. ‘car’ → *role of VIN that is of ‘VIN designates car’* [VIN = Vehicle Identification Number, ISO 3779, 3780]
  - **Data Models:** Structures for Recording/Storing Data ONLY (*part of the information system design*)
    - e.g. ‘car’ → information system structure for recording/storing data about cars
- What Semantic Metadata and Data Models Don’t Talk About
  - Since Concepts are ***units of thought***
    - in the minds of members of a semantic community and shared among them,
      - concepts are therefore **not *in themselves* data**
        - Data is ***propositions*** taken to be true expressed by ***designations*** for concepts & by ***identifiers*** for things
        - Concepts are ***communicated*** by using terms, names, definitions and examples
        - ***Documentation*** of concepts in a ***terminological dictionary or database*** is a special kind of data
    - Therefore, since **semantic metadata** is “data about data” and **data models** are “data about structures for recording/storing data”,
      - semantic metadata and data models ***do not document concepts.***
    - Concepts can ***and should*** be related to **semantic metadata** and **data models** by their representations, but they are not part of either – they are in people’s minds

# Terminology Science and Information Science – Are About Different Kinds of Things



# Common Understanding Opens Valuable New Opportunities

---

- Understanding:
  - the differences between, and inter-connections among:
    - Terminology (ISO 704 & 1087)
    - Semantic Metadata (ISO/IEC 11179 & 19763)
    - Data Models
    - Recorded Data

is the basis for the bridge from the language of business people to implemented information systems.

- End to end traceability goes all the way from business definitions to the implement system –
  - not, as in the past, just from IS requirements to the implemented system.

- Understanding that:
  - concepts exist in the minds of people; data (recorded) exist on a medium
  - things in the subject and data about them are two very different things
  - terminologies are ***not metadata***; they are ***dictionaries***

is the basis for usefully inter-connecting terminology with semantic metadata and data models, ***and standards for them***

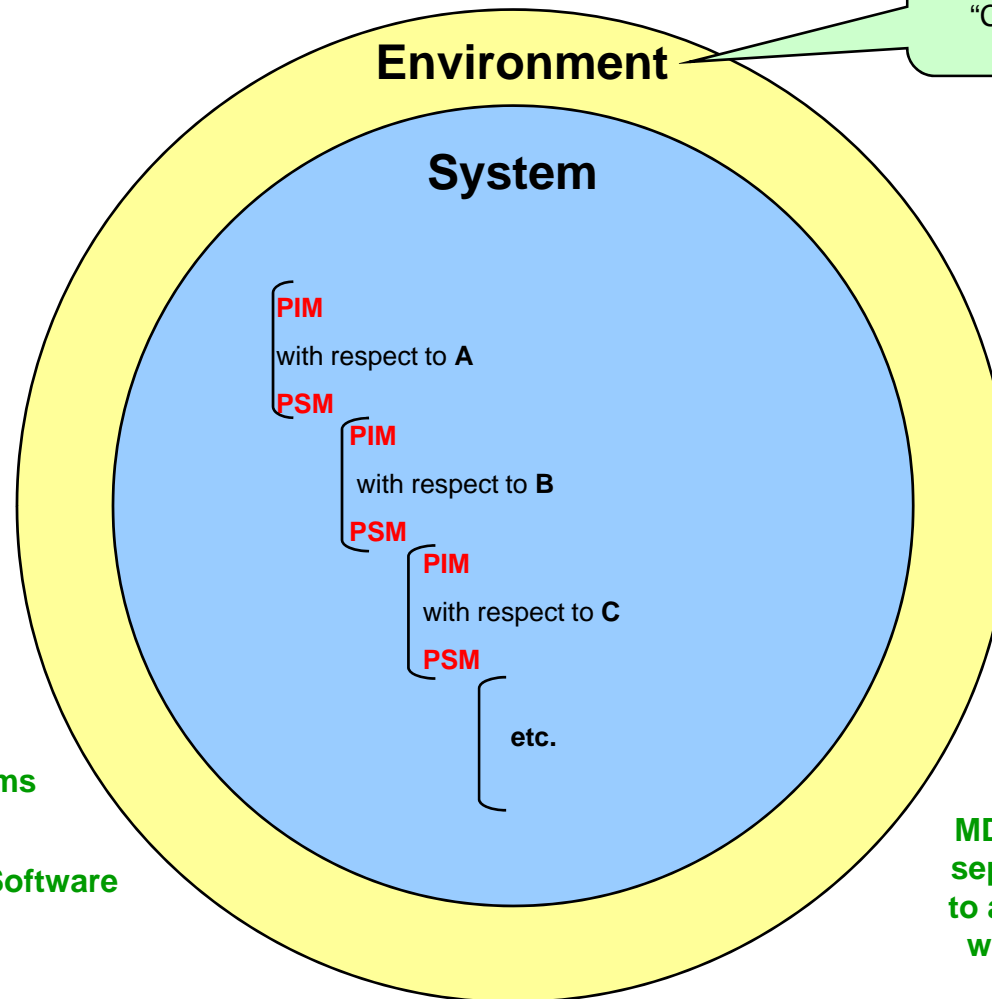
# **3. Architecture for Transforming Terminology to Semantic Metadata & Data Models**

BASED ON

the OMG's new MDA Foundation Model  
as applied to both  
the Organization and  
its Business Application Software  
(See Reference Appendix A)

# MDA Foundation Model: Applicable to ANY Kind of System

Before MDA was generalized what is now called "Environment" was called "CIM – Computation Independent Model"

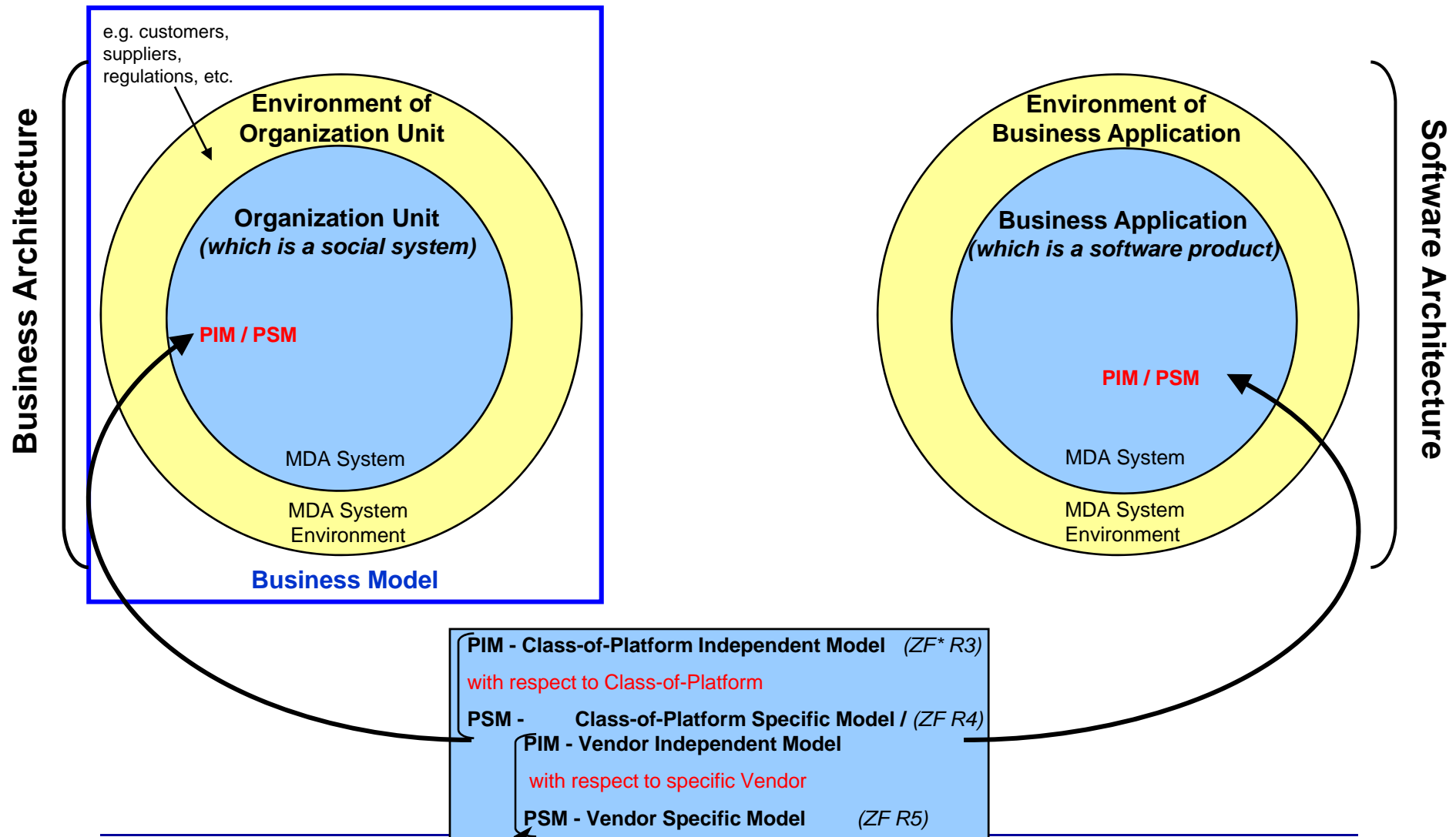


## Applicable to:

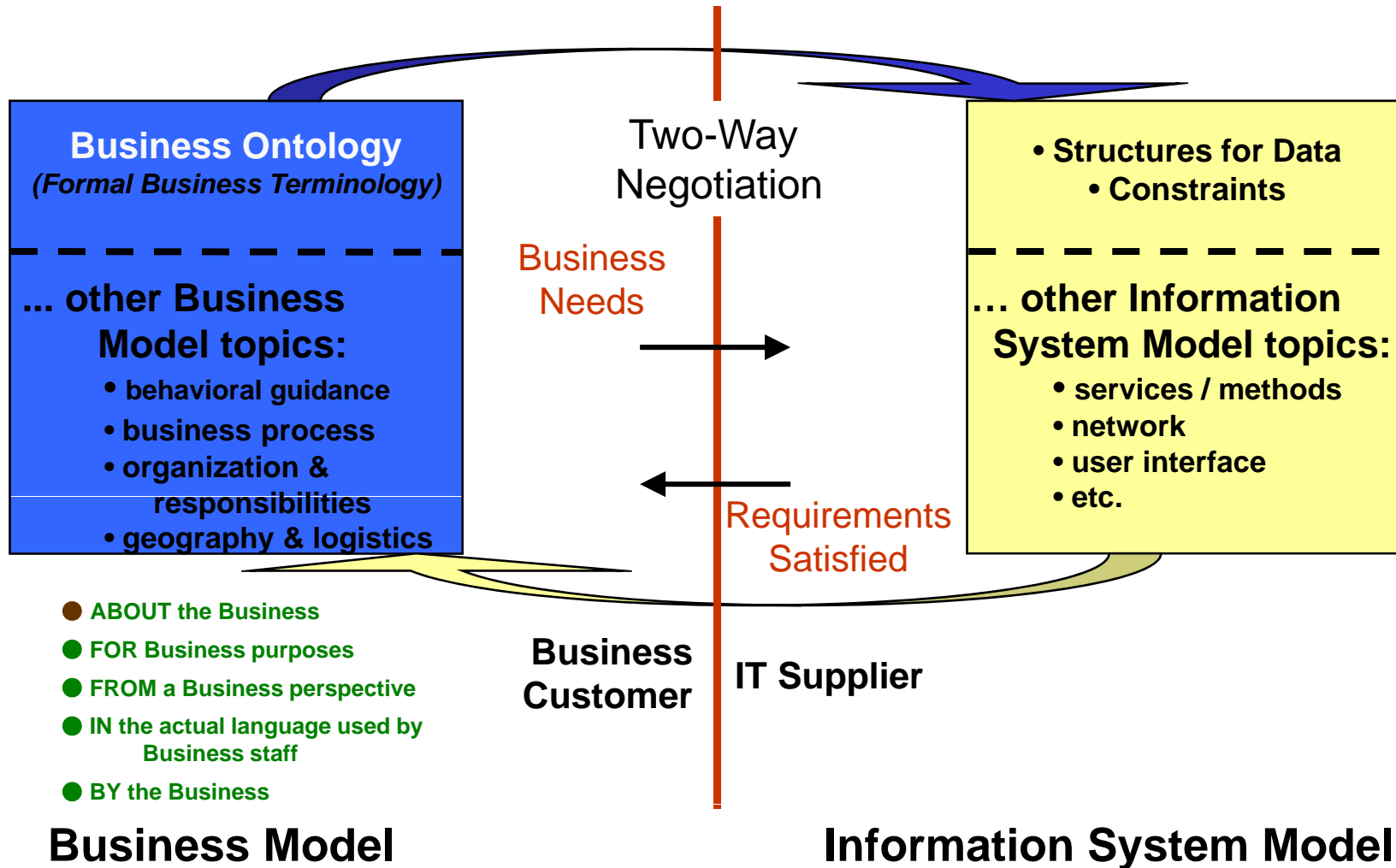
- Operating Systems
- Realtime Systems
- Process Control Systems
- Database Systems
- Business Application Software
- Organizations
- ... etc.

In addition to MDA Foundation Specification, separate documents applying it to a specific classes of systems will be published by the OMG

# MDA Foundation Model Applied: Business Architecture vs. Software Architecture



# Business Model vs. Information System Model



# Essential Business Model

---

## essential business model

**class-of-platform independent model** of the organization

- “Essential” (as in *of the essence*, not as in *necessary*)
- Independent of whether people or automated equipment will do the work
- Used to make these decisions
- Independent of any technologies
- Used to choose technologies and design the class-of-platform specific models

## essential business process model

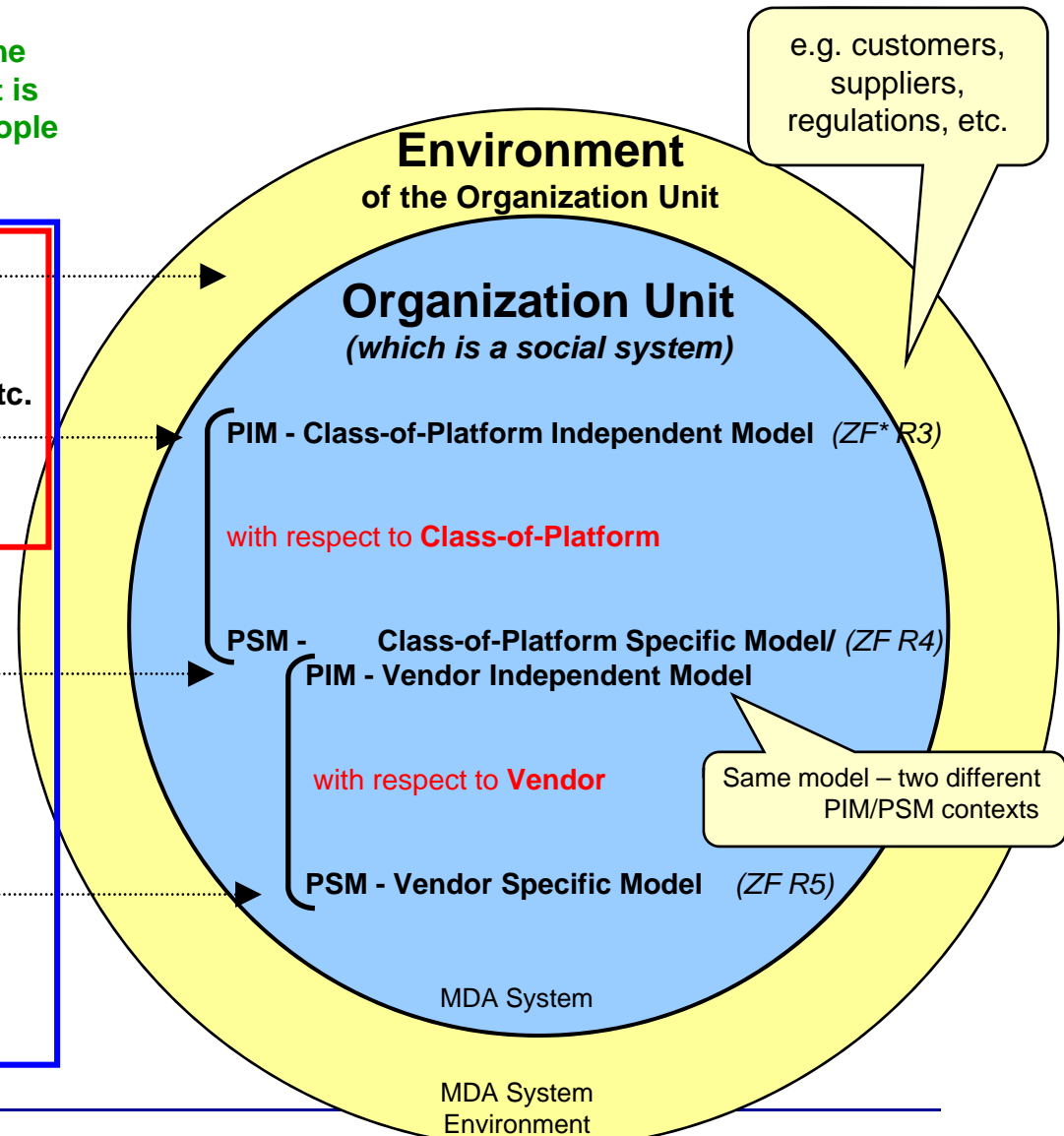
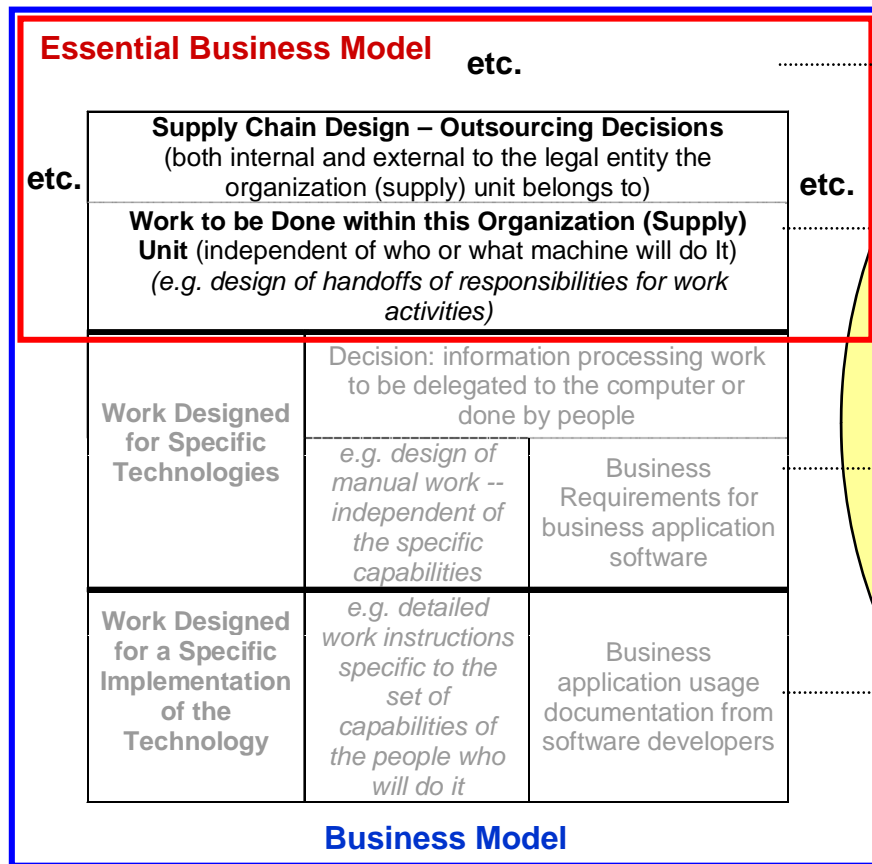
**essential business model** that models business processes of the organization

- Focuses on “bringing about new, original things, directly or indirectly by communication”
  - All tasks are about decisions, judgements, and engaging in commitments only
  - Included are the “**NEED to KNOW & NEED to REMEMBER**” requirements of bottom level **tasks** for **kinds of business facts** as defined in the **business ontology** (formal business terminology)
- Excludes all (*manual and automated*) information processing considerations
  - Excludes all information processing actions, transforms, movement and information storage
  - Does not include document, data or information artefacts flowing through the tasks of the process

\*see “performa” on pages 105-106, 144-148 in Dietz, Jan L.G. *Enterprise Ontology: Theory and Methodology*. Springer, 2006.

# MDA Foundation Model Applied: ... to Organizations

**Note: The Essential Business Model stops with the Class-of-Platform Independent Model as that is the basis for choosing technologies; e.g. people or computers for information processing**



# Architecture for Making the Bridge Between Organizations and Software Real

---

- An architecture for Business Models and Business Application Software Models and how they relate,

- based on the new generalization of the OMG's Model Driven Architecture,

provides an objective frame of reference for defining the:

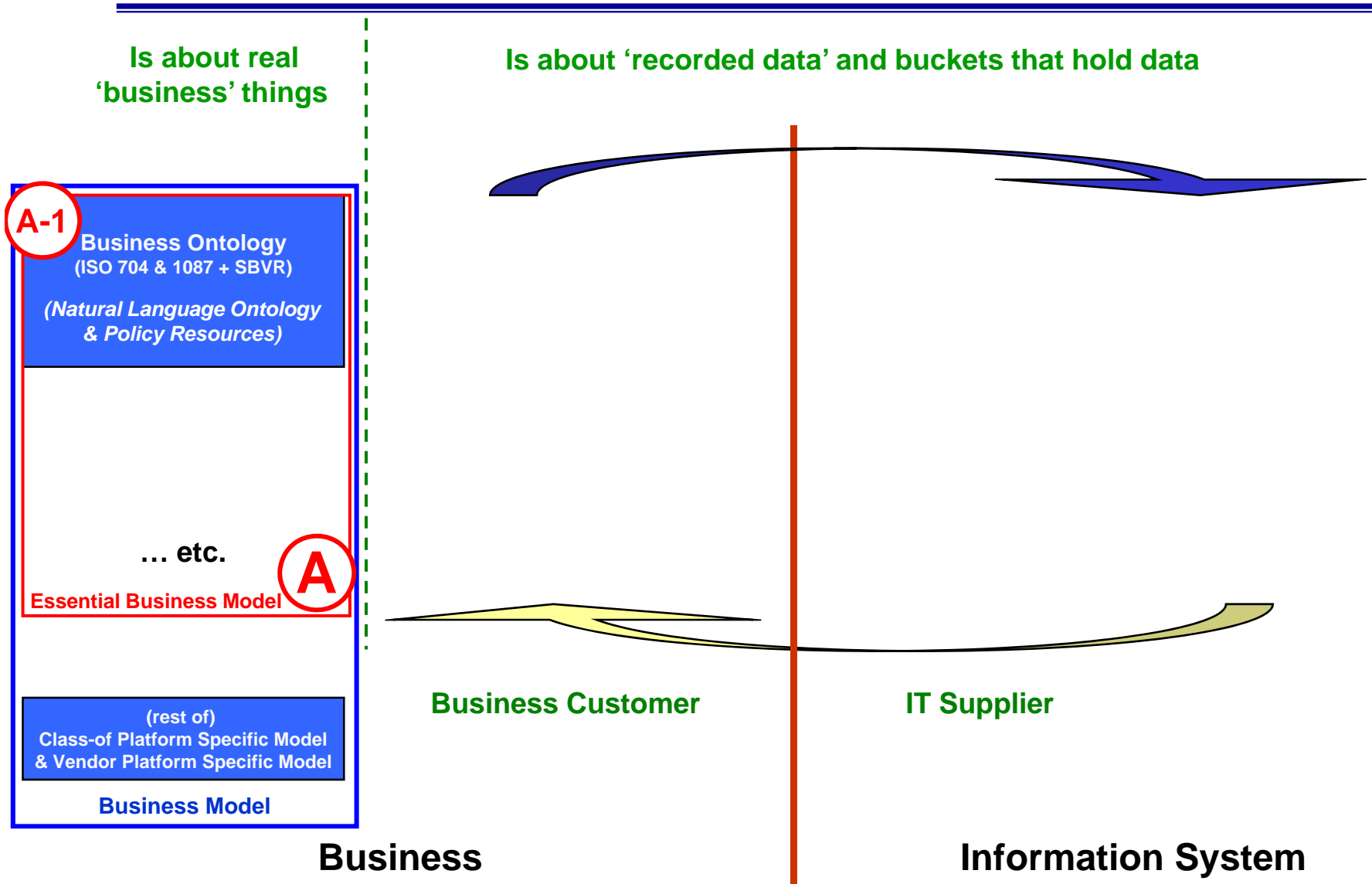
- key stages between the language of business people and implemented business application software
- transforms between stages to connect the two

- Modeling only the “essentials” of business processes:

- Reduces the work of modeling “business” processes
  - by eliminating information system processing actions from business consideration
- Enables the business to concentrate on
  - what it is trying to accomplish by in the business process, and
  - what has to be done from a business point of view

## **4. Transforming Terminology to Semantic Metadata & Data Models**

# How to Recognize a Business Ontology (A-1)



# How to Recognize a Business Ontology (A-1)

---

## car movement

Definition: planned movement of some [rental car](#)

Note: may be accomplished by a rental or by a car transfer made by a EU-Rent driver

## car movement specifies car group

Necessity: Each [car movement](#) *specifies exactly one* [car group](#)

## car movement has sending branch

Necessity: Each [car movement](#) *has exactly one* [sending branch](#)

## car movement has receiving branch

Necessity: Each [car movement](#) *has exactly one* [receiving branch](#)

## rental

Definition: contract *that is for use of some car and is for some rental period*

## rental has renter

Necessity: Each [rental](#) *has exactly one* [renter](#)

## rental includes car movement

Necessity: Each [rental](#) *includes exactly one* [car movement](#)

## advance rental

Definition: [rental](#) *that is contracted on or earlier than the day before the* [scheduled pickup date/time of the rental](#)

## walk-in rental

Definition: [rental](#) *that is contracted on the day that the car is picked up*

## cash rental

Definition: [rental](#) *that is paid for with money*

Note: Despite the name, cash rentals are usually paid with credit cards.

## points rental

Definition: [advance rental](#) *that is paid for with* [loyalty club points](#)

Examples taken from EU Rent Case Study in Annex E of  
SBVR 1.0 Specification  
([http://www.omg.org/technology/  
documents/br\\_pm\\_spec\\_catalog.htm](http://www.omg.org/technology/documents/br_pm_spec_catalog.htm))

# How to Recognize Behavioural Business Rules (A-1)

---

It is obligatory that a renter *has possession of at most one rental car*

It is prohibited that a barred driver *is a driver of a rental*

It is obligatory that a renter *who has possession of a rental car has a valid driver licence*

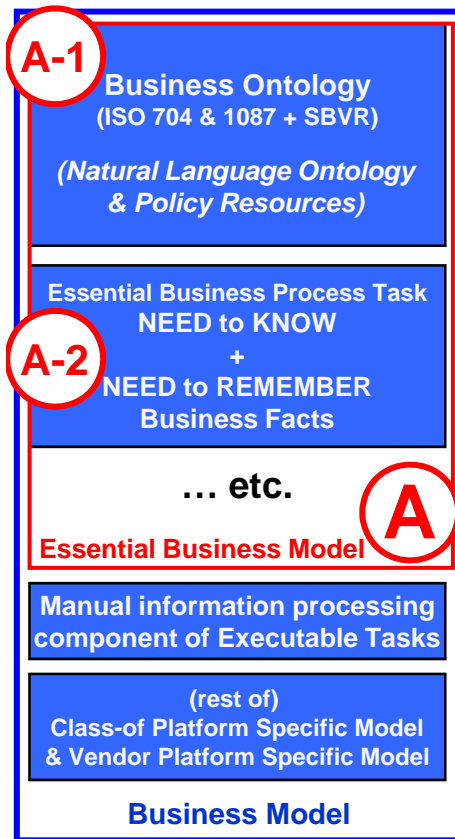
It is obligatory that a rental *is guaranteed by an acceptable credit card that is held by the renter of the rental*

It is obligatory that a renter *who has possession of a rental car is capable of driving the car safely*

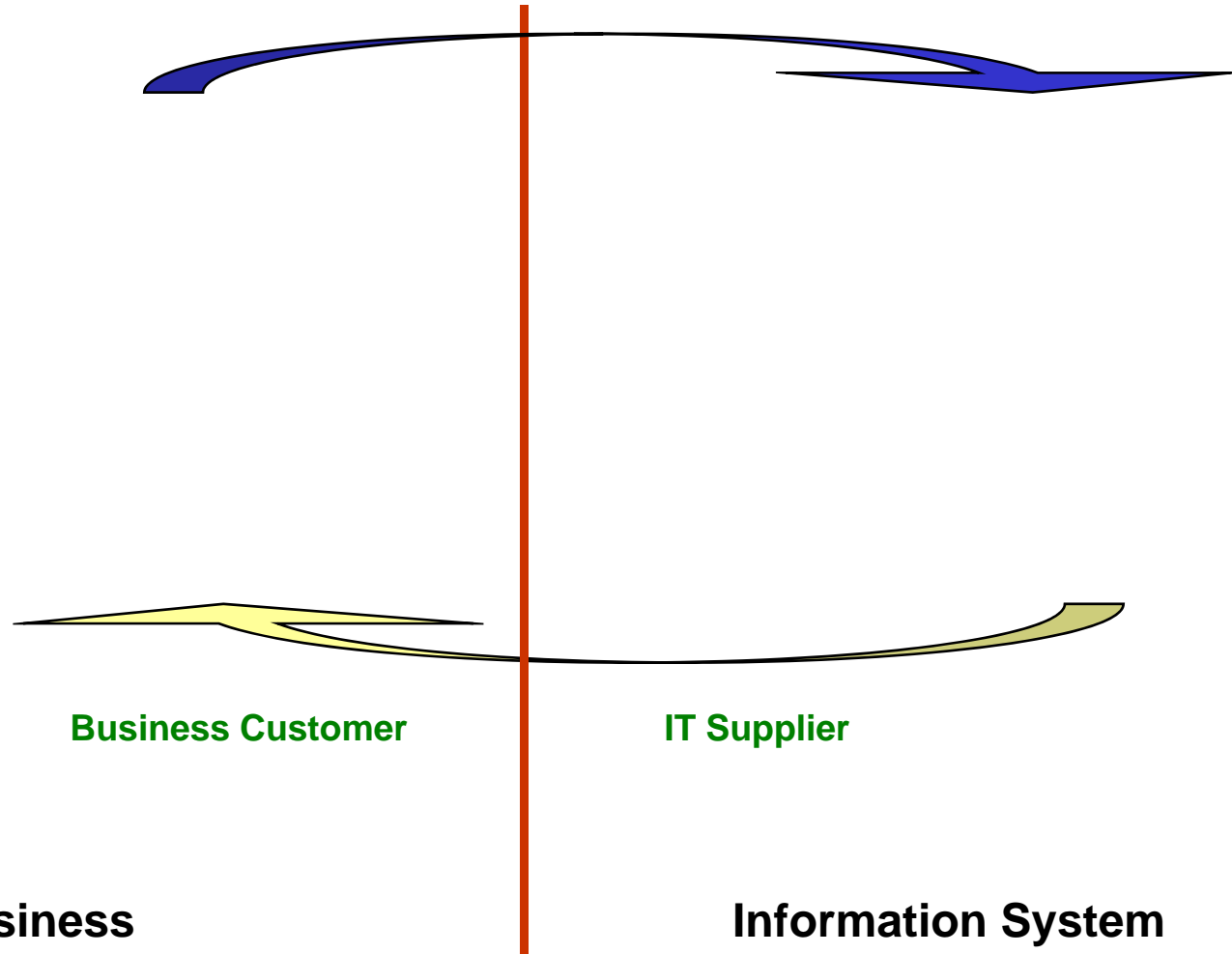
# How to Recognize Data Requirements of Essential Business Process Tasks (A-2)

Is about real 'business' things

Is about 'recorded data' and buckets that hold data

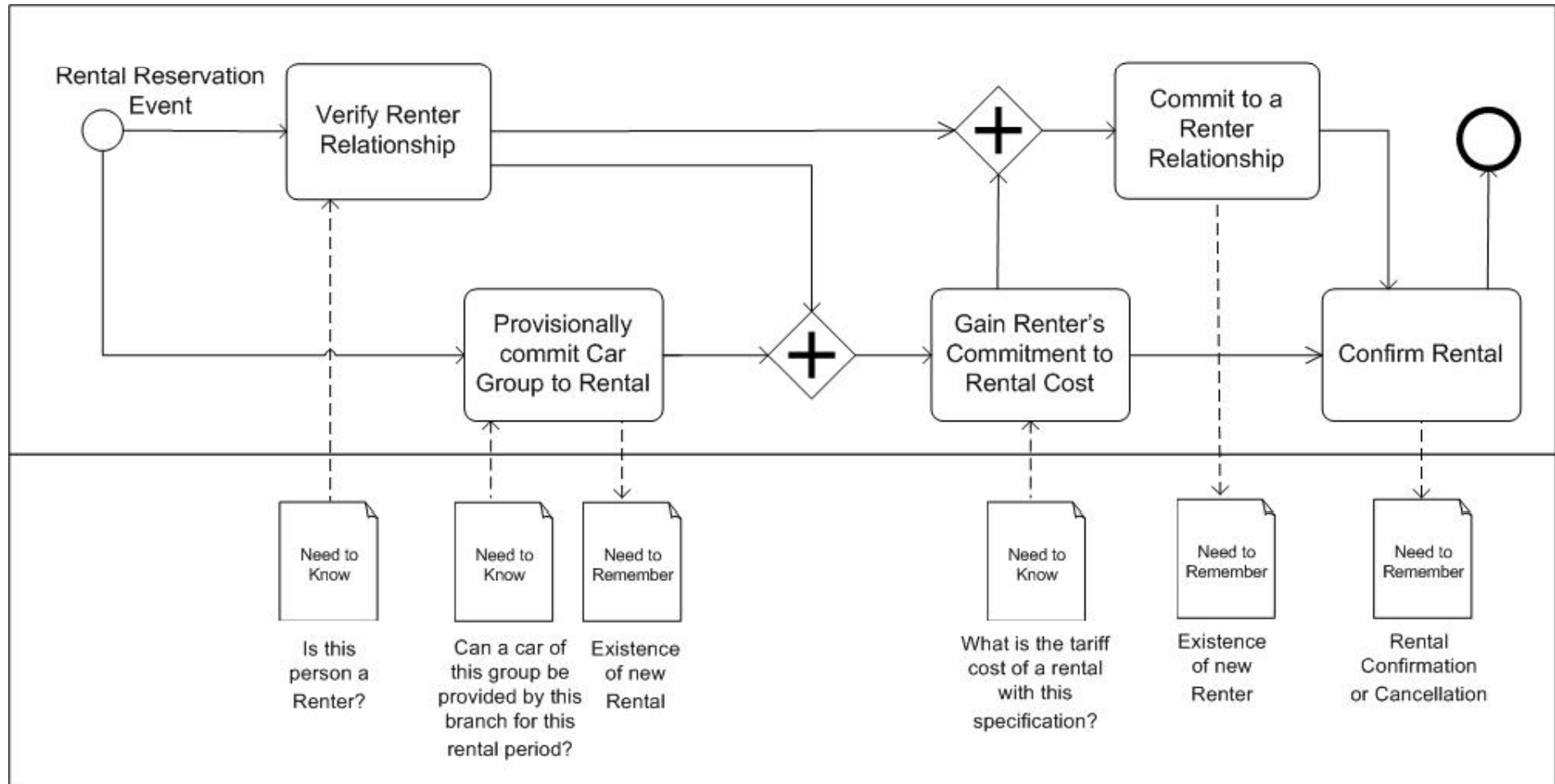


**Business**



**Information System**

# How to Recognize Data Requirements of Essential Business Process Tasks (A-2)



**NOTE: This is an Essential Business Process Model (see slide 16).**

**At this stage we do not know what part will be done by IT and what part by people as that is done in TRANSFORM 1.**

# Business Models Enable Better Software Requirements for Less

---

- Business Ontologies are natural language business terminologies that document the language of the business people
  - in a way that is understood in formal logic so much of the requirements for and design of application software can be automated
- Data requirements of Tasks in Essential Business Process Models are formulated with respect to doing the Task as:
  - NEED to KNOW: Questions and required kinds of business facts that answer them
  - NEED to REMEMBER: Business facts coming into existence as a result
- The NEED to KNOW / NEED to REMEMBER requirements + their supporting Business Ontology
  - provide the business input for specifying the “Consolidated Data & Rules Requirements” for new / improved application software
    - This business input will already exist in organizations that regularly document how they run their organization in Business Models, saving requirements costs

**4.1**

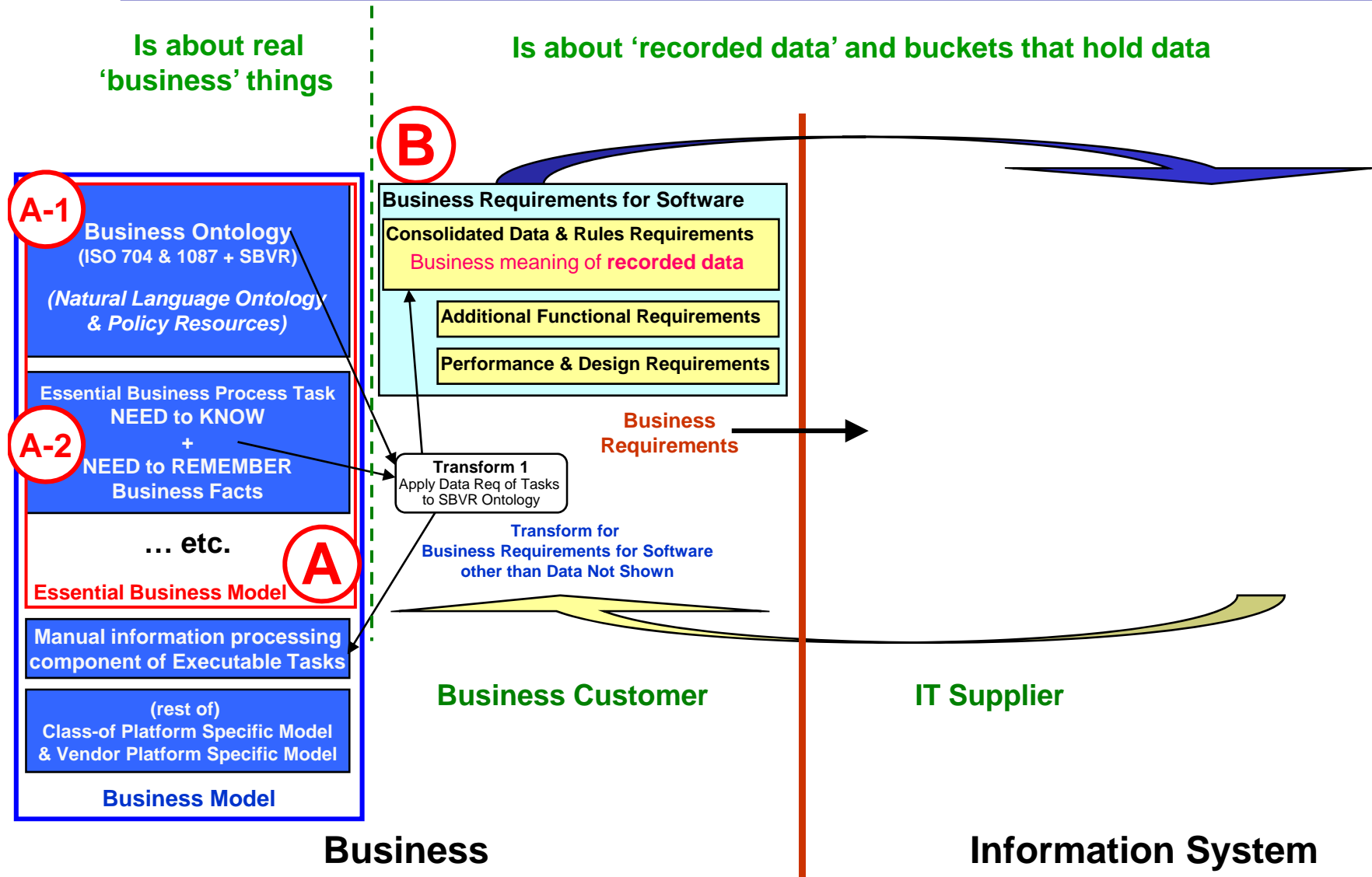
**Business Ontology (A)**



**Consolidated Data & Rules Requirements (B)**

Done by the Business as Part of  
Business Requirements for Software Specification

# How to Recognize Consolidated Data & Rules Requirements (B)



## Transform 1: Business Ontology (A) $\leftrightarrow$ Consolidated Data & Rules Requirements (B)

- Establish Scope
  - Identify the Essential Business Processes within scope
  - Identify the subset of Business Ontology that supports the selected Essential Business Processes
- Specify requirements for data support of business processes by IS
  - Questions to which processes require answers (includes reports, prompts, etc. as well as queries)
  - New knowledge created in business processes that has to be remembered
- Decide what information process will be manual and what will be automated
- Identify Behavioural Business Rules that govern Business Processes
  - To be automated in software
  - To be applied by people:
    - Requirements for recording actions
    - Questions needed to support decisions
- Specify performance requirements:
  - Volumes, speed of delivery, data quality and currency, integrity, protection, archiving, etc.
  - Reporting requirements for data management; e.g. number of errors, volume of change

# How to Recognize Consolidated Data & Rules Requirements (B)

---

- Accept Rental Booking

Questions (IS must be able to answer):

- Is this person already a EU-Rent renter?
- Can a car of this car group be provided by this pick-up branch for this rental period?
- What is the tariff cost of a rental with this specification?

New facts created (IS must be able to store and retrieve):

- (possibly) Existence of new renter
- Existence of new rental
- rental *is responsibility of* renter
- rental *specifies* car group
- rental *has* pick-up branch
- rental *has* scheduled pick-up date-time
- rental *has* return branch
- rental *has* scheduled return date-time
- (possibly) rental *requests* car model

# How to Recognize Consolidated Data & Rules Requirements (B)

---

## Behavioural Rules

- The information system must support all process decisions that are based on behavioural rules
- The business may have decided that some behavioural rules will be enforced by software. Rules in this category may be specified as constraints, e.g.
  - A renter can have only one rental car at a time (the rental period of a rental must not overlap the rental period of any other rental of the same renter)
  - No barred driver may be a driver of a rental
- Some behavioural rules may be over-ridden by staff with appropriate authority, e.g.
  - A branch manager may waive a rental return penalty

The data model may need to support recording of the justification, e.g. attributes 'waived status' and 'reason for waiver' on 'rental extra charge'

- Some behavioural rules have to be enforced by people, e.g.
  - At rental pick-up time, each driver must present an original, valid driver's licence  
The data model may need to support recording of confirmation that branch staff have checked driver licences
  - No EU-Rent car may be handed over to a person who appears incapable of driving the car safely  
The data model may need to accommodate decision support information for branch staff, e.g. symptoms of alcohol or drug influence or illness, limits on height, girth and physical disabilities

## Examples for Transform 1: Business Ontology (A) $\leftrightarrow$ Consolidated Data & Rules Requirements (B)

CASE 1: Scope - Rentals and Service Scheduling

Business Ontology (A)	Consolidated Data & Rules Requ. (B)
Relevant subset of Business Ontology	Essential business processes for Rentals and Service Scheduling: Rental Reservation, Rental Pick-up, Rental Return, Car Transfer, Service Scheduling ...

CASE 2: Business process requirements for data

Business Ontology (A)	Consolidated Data & Rules Requ. (B)
renter, person, driver, loyalty club member car group, RTU, operating country, rental rate rental, pick-up branch, drop-off branch, pick-up date-time, drop-off date time, car group, car model, renter	Question: Is this person already a renter? Question: What is the basic tariff cost of this rental? New fact: new rental Constraint: rental can have at most 3 additional drivers

CASE 3: Automated and manual processes

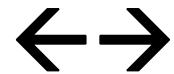
Business Ontology (A)	Consolidated Data & Rules Requ. (B)
	Automated: assign cars to day's rentals Manual: rental pick-up Both: accept rental reservation

CASE 4: Operative Business Rules

Business Ontology (A)	Consolidated Data & Rules Requ. (B)
renter can have at most one car at a time	rental periods for two rentals of the same renter must not overlap

**4.2**

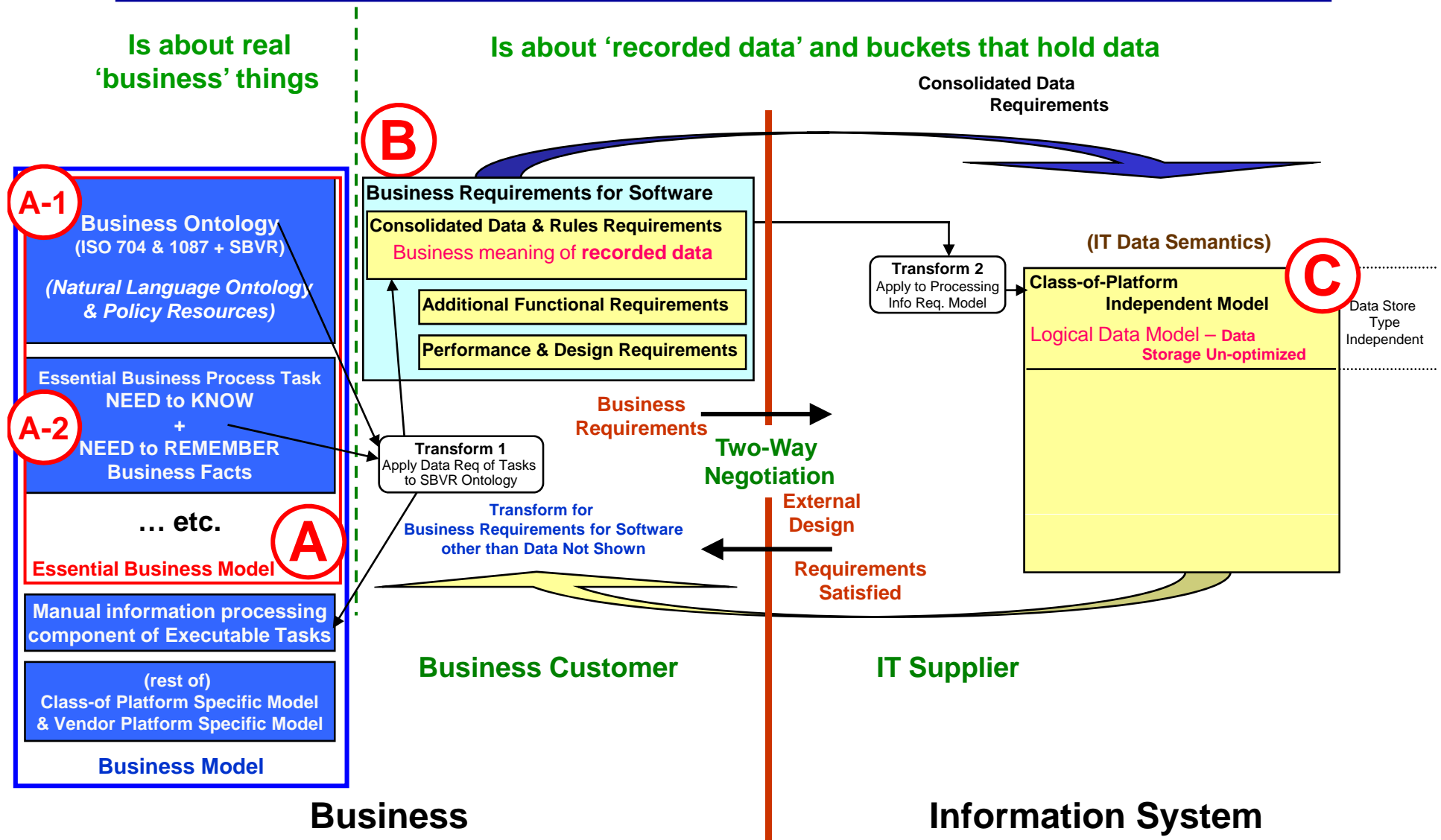
**Business Requirements for Software (B)**



**Class-of-Platform Independent Data Model (C)**

Done by IT specialists as Part of Systems Design

# How to Recognize a Class-of-Platform Independent Data Model (C)



## Transform 2: Business Requirements for Software (B) ↔ Class-of-Platform Independent Data Model (C)

---

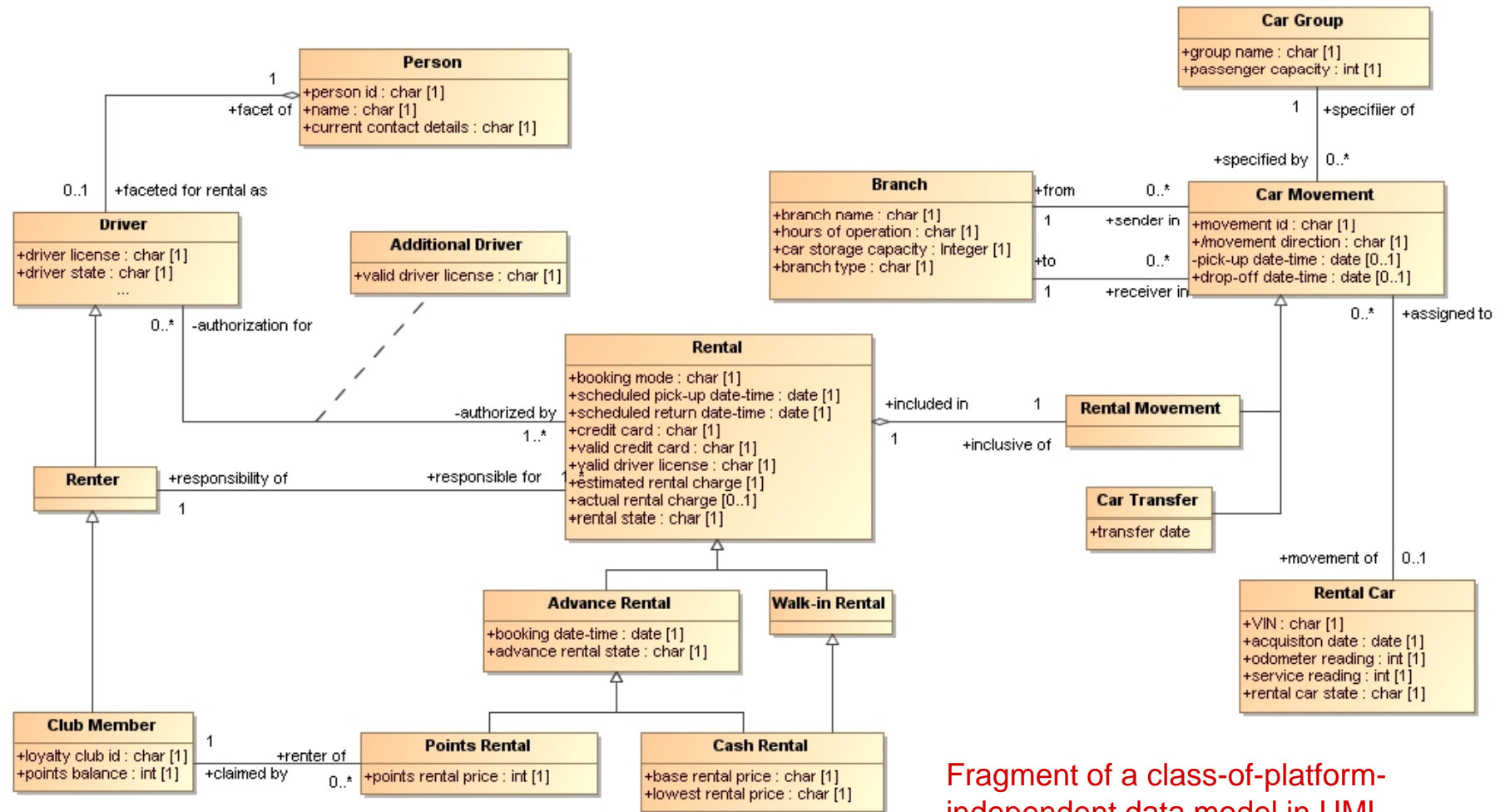
- Transform the scoped subset of the Business Ontology to data model:
  - Noun concepts to classes/entities
    - Definitional rules may be transformed into constraints on data
    - Some constraints may be placed on item representation – length, value ranges ...
    - Some synonyms may be required so that different types of user can retrieve data in their own terms
  - Fact types to associations/relationships

Validate the data model for support of ‘need to know’ and ‘need to remember’ requirements of Essential Business Processes:

- Start point – identified by data provided by the business process
- ‘Navigable path’ through the data model, accessing all required data items
- Ensure support for Behavioural Rules
  - Rules that are to be automated:
    - Strictly-enforced - may become specifications of constraints on data
    - Not strictly enforced – IS will need to request a user decision, and perhaps record reason for level of enforcement applied
  - Rules that are to be applied by people:
    - May need decision support information, and to record violations detected

Grouping noun concepts into entities/classes and attributes may be done in this transformation, or later. For example, in CogNIAM and ORM, this transformation could produce a scoped fact model to meet requirements, and a later transformation would produce a data model with attributes).

# How to Recognize a Class-of-Platform Independent Data Model (C)

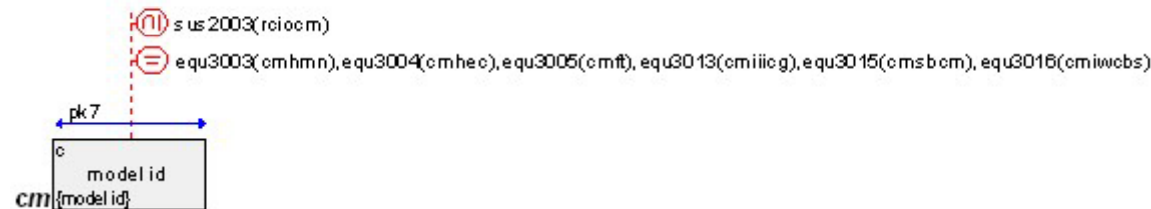


Fragment of a class-of-platform-independent data model in UML notation

# Class-of-Platform Independent Model (C) – alternative target: Scoped Fact Model

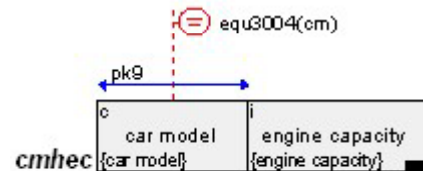
Fragment of a fact model in  
CogNIAM notation

car model {car model}



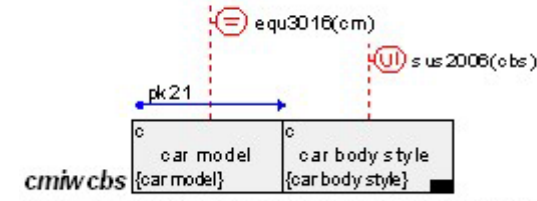
7: [model id] <model id> identifies a specific [car model]

car model has engine capacity {car model has engine capacity}



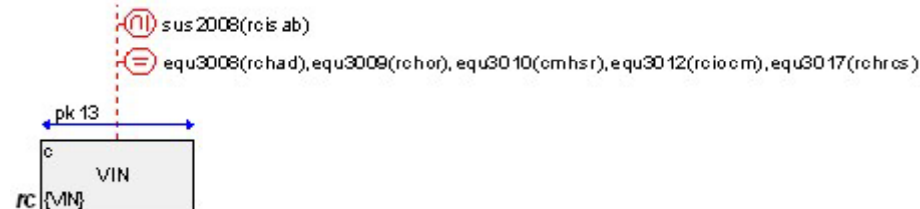
9: [car model] <car model> has [engine capacity] <engine capacity>

car model is with car body style {car model is with car body style}



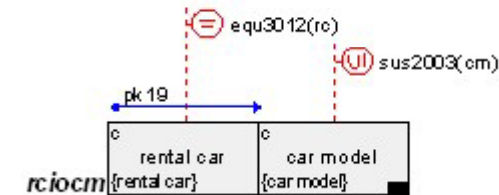
24: [car model] <car model> is with [car body style] <car body style>  
25: [car body style] <car body style> is of [car model] <car model>

rental car {rental car}



16: [VIN] <VIN> identifies a specific [rental car]

rental car is of car model {rental car is of car model}



22: [rental car] <rental car> is of [car model] <car model>  
26: car model <car model> is model of [rental car] <rental car>

# Examples for Transform 2:

## Business Requirements for Software (B) ↔ Class-of-Platform Independent Data Model (C)

CASE 1: Business Ontology to Data Model

Business Requirements for Software (B)	Class-of-Platform Independent Data Model (C)
rental, car movement .. booking date-time, sending branch, receiving branch ...	Classes, with association 'rental includes car movement' Attributes

CASE 2: Validate support for 'Need to Know'

Business Requirements for Software (B)	Class-of-Platform Independent Data Model (C)
What is the tariff charge for a rental with this specification?	Derivation: decompose rental period into RTUs Path 1: branch – site – operating country - currency Path 2 car group - rental rate for country and RTU

CASE 3: Validate support for 'Need to Remember'

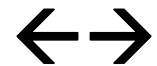
Business Requirements for Software (B)	Class-of-Platform Independent Data Model (C)
New rental	car movement (car group, sending and receiving branch) rental (booking date-time, scheduled pick-up and return date-times, state = provisional)

CASE 4: Support for Behavioural Rules

Business Requirements for Software (B)	Class-of-Platform Independent Data Model (C)
Automated: renter has at most one car at a time Manual: driver must have original valid licence	rental – renter – rental Additional attribute of rental: "licence checked"

**4.3**

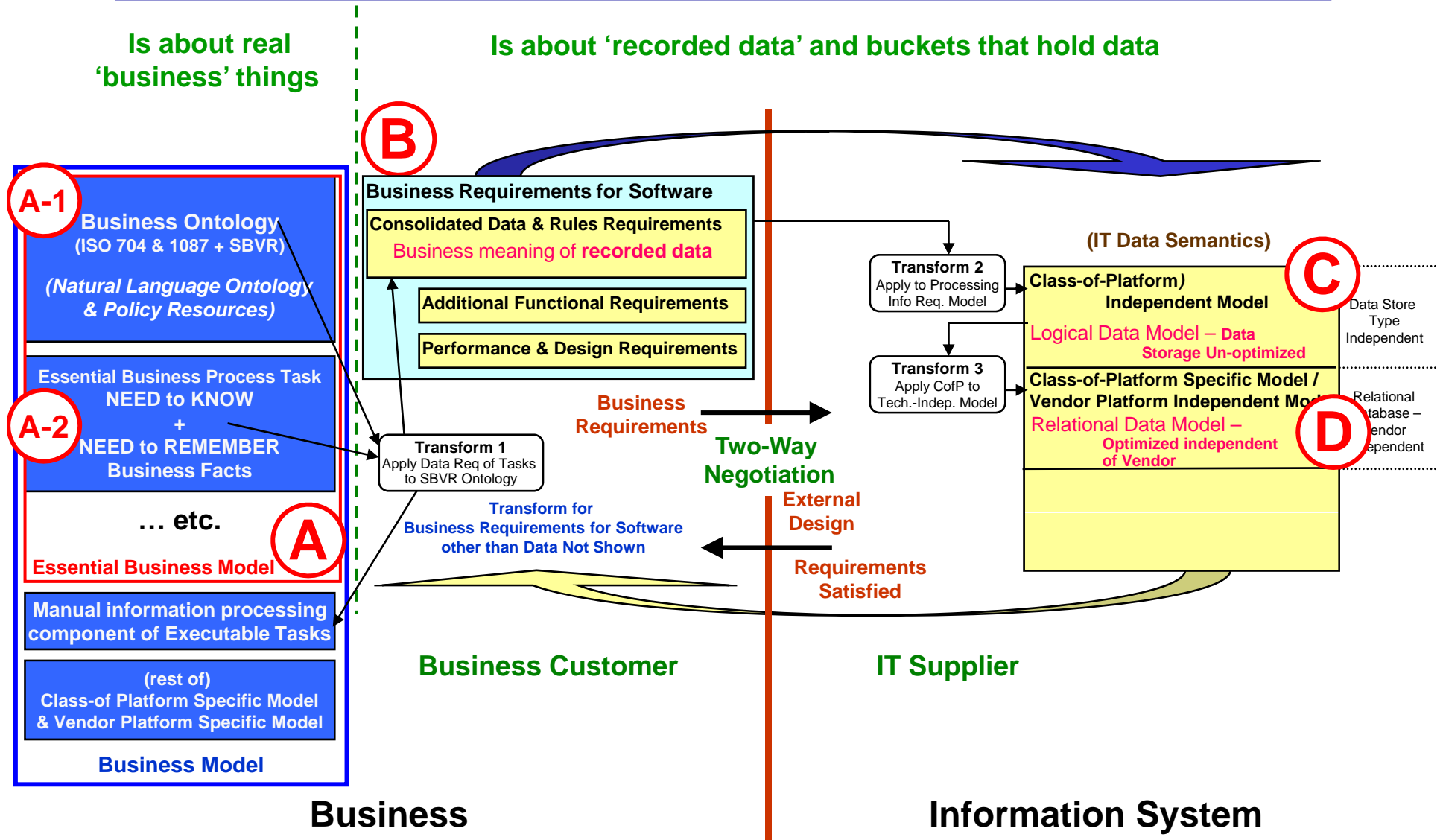
**Class-of-Platform Independent Data Model (C)**



**Class-of-Platform Specific Data Model (D)**

Done by IT specialists as Part of Systems Design

# How to Recognize a Class-of-Platform Specific Data Model (D)



## Transform 3: Class-of-Platform-Independent Data Model (C) $\leftrightarrow$ Class-of-Platform-Specific Data Model (D)

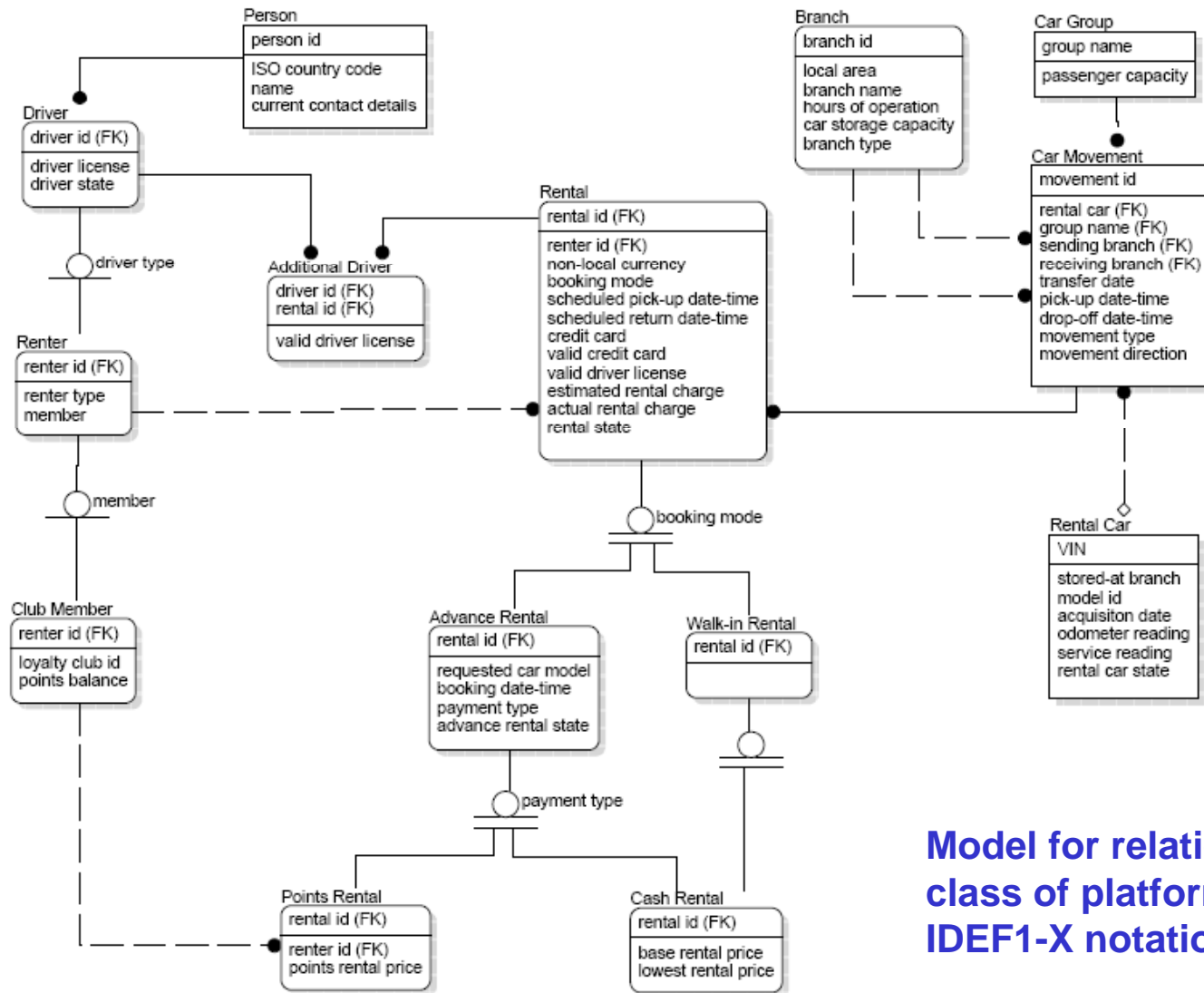
---

The detail of the transform will vary, depending on the target class of platform

Example – relational class of platform

- If attributes were not grouped into entities in transformation 2, do so
- Resolve many-to-many associations with associative entities
- Designate or create primary key(s) for each entity
- Ensure that each association is supported with keys
  - Introduce foreign keys into ‘child’ entities as necessary
- Ensure that all constraints are mapped from previously defined definitions and business rules and are supported:
  - Document any that need to be applied by processes

# How to Recognize a Class-of-Platform Specific Data Model (D)



**Model for relational class of platform in IDEF1-X notation**

# Examples for Transform 3:

## Class-of-Platform Independent Data Model (C) $\leftrightarrow$ Class-of-Platform Specific Data Model (D)

CASE 1: Direct correspondence

Class-of-Platform Independent Data Model (C)	Class-of-Platform Specific Data Model (D)
Classes and non-key attributes correspond directly, except for subtypes of 'car movement'	There is little risk in a relational design in merging 'car movement' and its simple subtypes. A 'movement type' attribute is introduced to distinguish instances.

CASE 2: Primary keys

Class-of-Platform Independent Data Model (C)	Class-of-Platform Specific Data Model (D)
Some business identifiers are suitable as primary keys Primary keys have to be defined for all entities	E.g. VIN, movement id, group name Specified, or inherited via defining relationships, e.g. 'movement id' is used as primary key for rental – with role name 'rental id'.

CASE 3: Associations

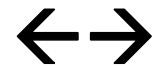
Class-of-Platform Independent Data Model (C)	Class-of-Platform Specific Data Model (D)
Associations are modelled with foreign keys	The primary key of the 'parent' entity is used as a foreign key in the 'child' entity, e.g. 'renter id' in 'rental'

CASE 4: Specialization

Class-of-Platform Independent Data Model (C)	Class-of-Platform Specific Data Model (D)
Type indicators have been added to supertypes, to distinguish subtypes ( common IDEF1-X practice),	E.g. 'driver type' in 'driver', 'payment type' in 'advance rental'

**4.4**

**Class-of-Platform Specific Data Model (D)**



**Vendor Platform Specific Data Model (E)**

Done by IT specialists as part of Systems Design



## Transform 4: Class-of-Platform-Specific Data Model (D) $\leftrightarrow$ Vendor Platform Specific Data Model (E)

---

The detail of the transform will vary, depending on the target platform

- Example – relational platform
  - Usually, the modelling tool can provide an automated “first-cut” design
  - The DBA can then optimize for performance:
    - Merging subtypes
    - Denormalizing
    - Changing placement options
    - Deciding on indexes
    - Etc ...

# How to Recognize a Vendor Platform Specific Data Model (E)

PERSON
person-id (pk) person_name contact_details ISO_country (fk) renter_indicator driver_indicator driver_license driver_state

Indexed on:  
person\_id  
ISO\_country

CLUB_MEMBER
person-id (pk) loyalty_club_id (ak) points_balance

Indexed on:  
person\_id  
loyalty\_club\_id

RENTAL
movement_id (pk) person_id (fk) non-local currency (fk) booking_mode_indicator booking_date-time scheduled_pick_up_date-time scheduled_return_date-time requested_car_model payment_type_indicator basic_rental_price lowest_rental_price loyalty_club_id (fk) points_rental_price credit_card valid_cc_flag valid_license_flag estimated_rental_charge actual_rental_charge points_flag points_rental_price rental_state

Indexed on:  
movement\_id  
person\_id

CAR_MOVEMENT
movement_id (pk) group_name (fk) rental_car_VIN sending_branch ISO_country receiving_branch scheduled_date movement_type_indicator movement_direction_indicator

Indexed on:  
movement\_id  
group\_name  
rental\_car\_VIN  
ISO\_country  
scheduled\_date  
sending\_branch  
receiving\_branch

CAR_MOVEMENT_TIMES
movement_id (pk) pick_up_date-time drop-off_date-time

Indexed on movement\_id

# Examples for Transform 4:

## Class-of-Platform Specific Data Model (D) ↔ Vendor Platform Specific Data Model (E)

CASE 1: One-to-one correspondence

Class-of-Platform Specific Data Model (D)	Vendor Platform Specific Data Model (E)
Entity 'club member' and its attributes Note: 'renter id' is a role name of 'person id'	Table CLUB_MEMBER, with person_id as primary key

CASE 2: Subtypes and supertype merged

Class-of-Platform Specific Data Model (D)	Vendor Platform Specific Data Model (E)
'person', 'driver' and 'renter' and their attributes 'rental' and its subtypes and their attributes	Table PERSON and table RENTAL with 'advance rental state' merged into rental_state Indicators are added to distinguish subtypes Null values are permitted for subtypes' attributes

CASE 3: Denormalization for performance (1)

Class-of-Platform Specific Data Model (D)	Vendor Platform Specific Data Model (E)
'country of rental' is 'country of site of sending branch of car movement included in rental'	ISO_country is included in table CAR_MOVEMENT to partition rentals by country

CASE 4: Denormalization for performance (2)

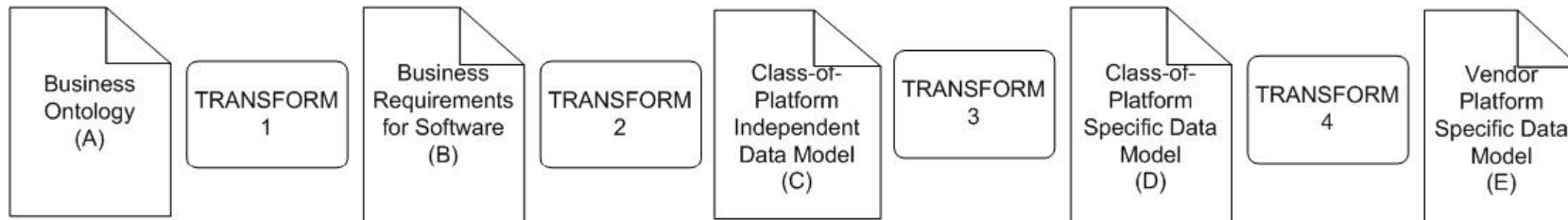
Class-of-Platform Specific Data Model (D)	Vendor Platform Specific Data Model (E)
Attributes 'pick-up date-time' and 'drop-off date-time' of car movement are actual values recorded after the events	They are stored in a separate table to reduce the size of the heavily-indexed CAR_MOVEMENT table.

# **4.5**

## **Optimization and Trade-offs**

Transformation Short Cuts

# Transform Shortcuts for Specific Situations



## Shortcuts that Continue to Bridge the Way Business People Think, Talk & Write and Implemented Software

Combine Transforms 3 & 4:

Ability to more easily implement the Class-of-Platform Specific Model with a different vendors product will be significant reduced

Combine Transforms 2 & 3:

Ability to more easily implement the information system architecture using a different class-of-platform will be significant reduced

Combine Transforms 2, 3 & 4:

Implementing with a different vendor or class-of platform and vendor will require an almost complete redesign of the information processing system

Combine Transforms 1 & 2:

Ability to trace and deal with all the impacts of changes in the essential business process and its data needs will be greatly reduced

## Shortcuts the Diminish the Software's Connectedness to the Way Business People Think, Talk & Write

Omitting or Inadequately Performing Transform 1:

- Not keeping and managing a business ontology (A) as a language asset for the organization
- Not using and staying connected to an active business ontology (A) while creating Consolidated Data & Rules Requirements Specifications

Omitting or Inadequately Performing Transform 2:

- Not creating a Consolidated Data & Rules Requirements (B) based on data originations and uses in the business of the organization
- Not using and staying connected with an Consolidated Data & Rules Requirements (B) or the logical data model (class-of-platform independent data model (C))

Only Maintaining the Physical Data Model:

- Not Maintaining the logical data model (C) or staying connected to the business ontology (A) in place of that

# Architecture the Bridge Between Organizations and Software – Best of Both Worlds

---

- With the upcoming OMG MDA Architecture for Organizations and Business Application Software, it will be possible to create a design:
  - for application software that:
    - communicates with business users in their own language, and
    - implements their business policies
    - enforces the data constraints implicit in Business Ontology concept definitions
  - appropriate to strategic needs
    - portability:
      - across technologies (*classes-of-platform*), and
      - across vendors
- or**
- low cost and time to completion
  - designed most quickly for least cost for a single technology (*class-of-platform*), or vendor where that is all that is needed strategically

# 5. Using SBVR to Provide Business Semantics for Semantic Metadata & Data:

*When I Look at a Piece of Data, How Do I Know What It Means?*

SBVR  $\leftrightarrow$  Data Elements & Data Categories  
(+ ISO 704 & 1087) (ISO/IEC 11179) (ISO 12620)

# SBVR is a Synthesis from Four Disciplines

---

## 1. TERMINOLOGY & VOCABULARY:

- **The foundation for SBVR** is ISO TC 37 (Terminology and Language & other Content Resources) terminology science standards ISO 704 and 1087
  - About human communication using special purpose language in the context of natural language

## 2. FACT-ORIENTED MODELING with interpretation in FORMAL LOGIC:

- **The precision of formal logic** was added to ISO 1087-1 concepts, designations, and concept relations by fact-oriented modeling\*
  - Precise meanings for SBVR Vocabulary and Behavioral Guidance enables them to be transformed into IT system designs without losing or changing the business semantics.

\* ISO Technical Report TR 9007:1987, "Concepts and Terminology for the Conceptual Schema and the Information Base", and  
"A Logical Analysis of Information Systems: Static Aspects of the Data Oriented Perspective"  
([http://www.orm.net/Halpin\\_PhDthesis.pdf](http://www.orm.net/Halpin_PhDthesis.pdf))

**SBVR 1.0 Specification** ([http://www.omg.org/technology/documents/br\\_pm\\_spec\\_catalog.htm](http://www.omg.org/technology/documents/br_pm_spec_catalog.htm))

# SBVR is a Synthesis from Four Disciplines

---

## 3. LINGUISTICS & LINGUISTIC ANNOTATION OF NATURAL LANGUAGE GRAMMAR

- **Target natural language grammar structures** (*external to SBVR*) were provided by:
  - linguistics,
  - ISO TC 37/SC 4 “Linguistic Representation” standards, and
  - De-facto industry standards

as input to the design of SBVR **semantic formulations** so that they would both:

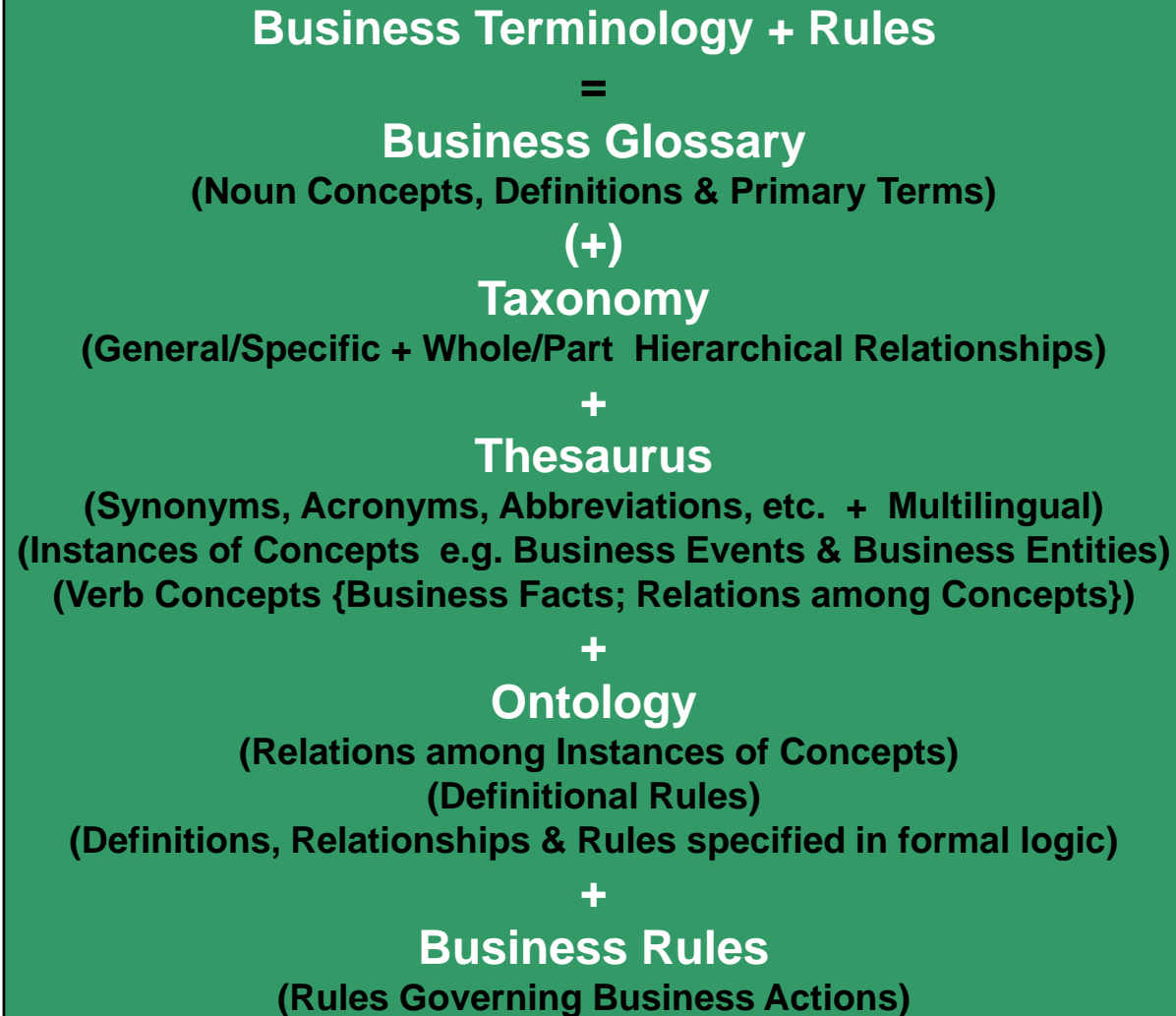
- adequately formulate in logic with a formal interpretation the most complicated **definitions and logic statements** expressed using selected natural language grammar features, and
  - adequately connect these **definitions and logic statements** to the underlying SBVR vocabulary of concepts and representations via verb concepts (*ISO TC 37 concept relations made formal by fact-oriented modeling*)
- Provided the basis for a future rich multilingual natural language notation for SBVR

## 4. BUSINESS PRACTICE of VOCABULARY & BUSINESS RULES:

- **Practical applicability of SBVR in Organizations** was provided by hundreds of collective man-years experience in business consultancy applying vocabulary and business rule approaches to the needs of organizations

# How SBVR Integrates Several Approaches

## How SBVR Relates to Existing Language Resources



# Ways SBVR Relates to ISO/IEC 19763 Metamodel Framework

---

- SBVR is a natural language (linguistics-based) ontology that:
  - is ISO 704 & 1087-1 turned into an ontology by the integration of fact-oriented modelling and its interpretation in formal logic
  - Should be added to the ontology language list in Annex C of ISO/IEC 19763-3
  - serves as one of the ontologies in figure 3 of ISO/IEC 19763-1
- The “Architecture for Transforming Terminology to Semantic Metadata & Data Models” is one architecture for transform mappings between ontologies and metadata registries.
- SBVR vocabulary (*aka terminology*) can provide the business meanings for data in metadata registries via ISO/IEC 11179
  - *For details see “When I Look at a Piece of Data, How Do I Know What It Means?” in this presentation*

# Challenge: Combine Data, Metadata & Concept Systems

## Inference Search Query:

*“find water bodies downstream from Fletcher Creek where chemical contamination was over 10 micrograms per liter between December 2001 and March 2003”*

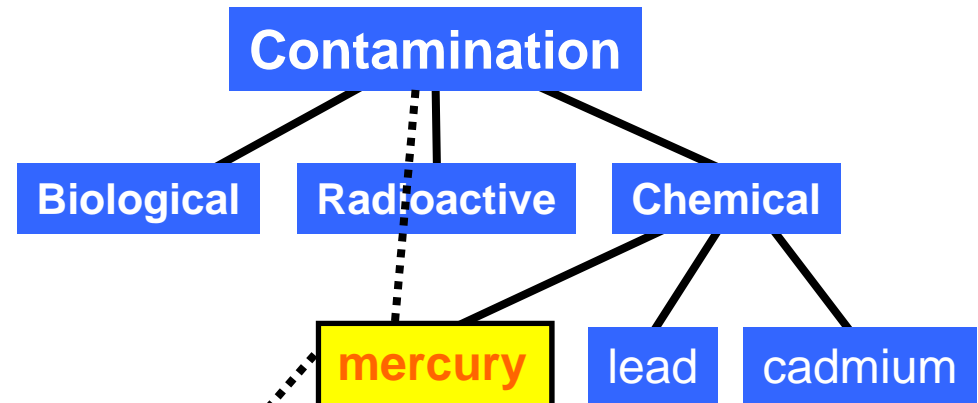
## Data:

<u>ID</u>	<u>Date</u>	<u>Temp</u>	<u>Hg</u>
A	06-09-13	4.4	4
B	06-09-13	9.3	2
X	06-09-13	6.7	78

## Metadata:

<u>Name</u>	<u>Datatype</u>	<u>Definition</u>	<u>Units</u>
<b>ID</b>	text	<b>Monitoring Station Identifier</b>	not applicable
<b>Date</b>	date	<b>Date</b>	yy-mm-dd
<b>Temp</b>	number	<b>Temperature (to 0.1 degree C)</b>	degrees Celcius
<b>Hg</b>	number	<b>Mercury contamination</b>	micrograms per liter

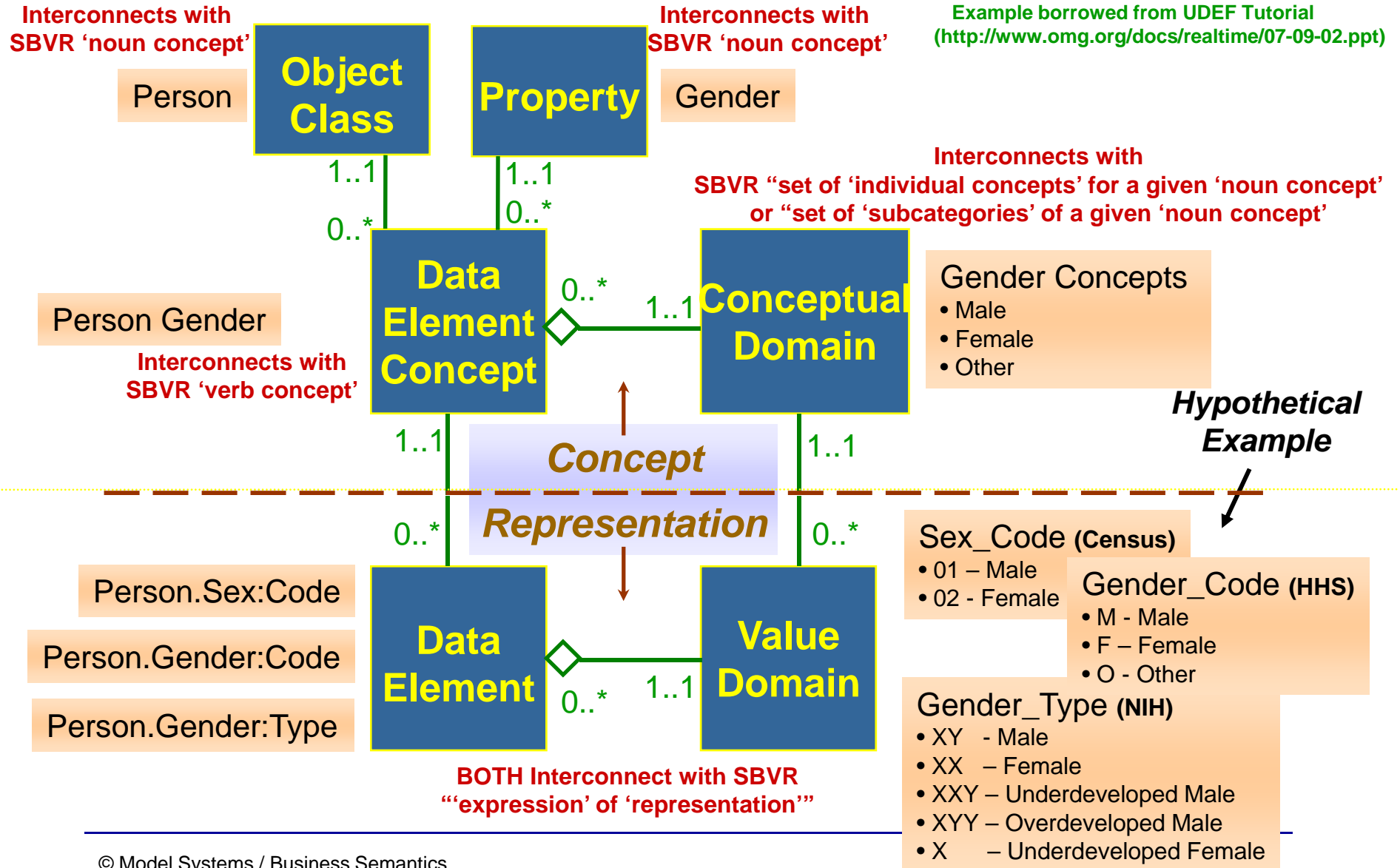
## Concept system:



From ISO/IEC 11179 presentation by Bruce Bargmeyer  
<http://hpcrd.lbl.gov/SDM/XMDR/presentations/Bruce%20Bargmeyer-XMDR%20&%20Ecoinformatics-V10%20for%20Plenary-2006-10-224.ppt>

# SBVR Model Elements Interconnected with ISO/IEC 11179 Model Elements

Example borrowed from UDEF Tutorial (<http://www.omg.org/docs/realtime/07-09-02.ppt>)



# Using SBVR Structured English Definitions to Give Loosely Coupled Business Semantics to ISO/IEC 11179 Data Elements

---

- Interconnect DATA ELEMENT & DATA VALUE to SBVR “‘Expression’ of ‘Representation’”
  - **Car Movement ID:** ([movement-id](#))
  - **VIN:** ([role of VIN that is of ‘VIN designates car’](#))  
[VIN = Vehicle Identification Number, ISO 3779, 3780]
  - **Car Group Name:** ([group name that designates car group that is specified by car movement that is of movement-id](#))
- Interconnect DATA ELEMENTS CONCEPT to SBVR ‘Verb Concept’
  - ([car movement specifies car group](#))
- Interconnect CONCEPTUAL DOMAIN to SBVR:
  - “set of ‘individual concepts’ for a given ‘noun concept’, or
  - “set of ‘subcategories’ of a given ‘noun concept’
  - Individual Concepts comprising the conceptual domain of [car group](#):
    - ([Economy](#))
    - ([Compact](#))
    - ([Intermediate](#))
    - ([Full Sized](#))
- Interconnect OBJECT CLASS & PROPERTY to SBVR ‘Noun Concept’
  - ([car group](#))
  - ([car movement](#))
  - ([transfer date](#))

Examples taken from EU Rent Case Study in Annex E of  
SBVR 1.0 Specification  
([http://www.omg.org/technology/  
documents/br\\_pm\\_spec\\_catalog.htm](http://www.omg.org/technology/documents/br_pm_spec_catalog.htm))

# SBVR Business Ontologies Connect Semantic Metadata to the Actual Language Business People Use

---

- The organization is free to document, maintain and manage its language resources in Business Ontologies
  - in a way that is consistent with **all** of its intended uses
- IT is free to document, maintain and manage its data as an IT Metadata Registry
  - In a way that is best given its infrastructure, platforms, vendors and capabilities
- The two can be maintained independently
  - with requests in either way between the business and IT for additions and changed connections between them
- The meaning of data is specified by real business semantics –
  - the concept definitions business people use to think, talk and write

# References in this Presentation

---

1. “Semantics of Business Vocabulary and Business Rules (SBVR) 1.0 Specification  
<http://www.omg.org/cgi-bin/doc?formal/08-01-02.pdf>
2. “Comparison of Many Aspects of Terminology, Semantic Metadata, Data Models & Data” Chart  
[www.BusinessSemantics.com/Resources/terminology\\_comparison.pdf](http://www.BusinessSemantics.com/Resources/terminology_comparison.pdf)
3. OMG’s “The MDA Foundation Model”  
<http://www.omg.org/cgi-bin/doc?ormsc/07-06-03.pdf>
4. “MDA Foundational Model applied to both the Organization and Business Application Software” Diagram v1-0  
<http://www.omg.org/docs/bmi/08-03-19.pdf>
5. “Business Architecture as the Application of the MDA Foundation Model to ‘Organizations’”  
Presentation to Open Group Business Architecture Working Group, Glasgow, April 23, 2008  
<http://www.omg.org/docs/bmi/08-05-02.pdf>
6. The Deep Structure of Business Processes, Jan L.G. Dietz  
[http://www.demo.nl/option,com\\_docman/task,doc\\_download/gid,1/Itemid,81/](http://www.demo.nl/option,com_docman/task,doc_download/gid,1/Itemid,81/)
7. “Approximate Traceability from Terminology/SBVR to Data Models” Chart  
[www.BusinessSemantics.com/Resources/terminology\\_traceability.pdf](http://www.BusinessSemantics.com/Resources/terminology_traceability.pdf)

**END**

# **Appendix A.**

## Terminology Used

# Definitions Based on MDA Foundation Model and ISO 1087-1

---

## **model** (*generalized from MDA model*)

set of all the **expressions** of a single **system definition** of a **set of interrelated concepts** that are in the minds of, and shared among, the members of a semantic community, that are conceptualized for a specific purpose of a given subject field

## **system model (syn: MDA model)**

- More General Concept
  - **model**
- Delimiting Characteristics
  - the semantic community is the community of developers of the system
  - the given subject field is “developing some concrete or abstract thing of interest as a system”
  - the interrelated concepts are related to the thing by an explicit or implicit isomorphism

## **expression**

something that manifests a system definition in (a) specified notation(s) and is recorded in a specified medium

## **system definition**

formulation into an abstract syntax of the meaning of a **set of interrelated concepts** that are conceptualized for a specific purpose using given language specification; e.g. UML, OWL, ORM, etc.

## **set of interrelated concepts**

synonyms:

**concepts system** [ISO 1087-1 (3.2.11)]

**body of shared concepts** [SBVR]

**NOTE:** These definitions are based on “The MDA Foundation Model” Section 2. “Models”  
([HTTP://www.omg.org/docs/ormsc/07-06-03.pdf](http://www.omg.org/docs/ormsc/07-06-03.pdf))

# Definitions Based on MDA Foundation Model and ISO 1087-1

---

## terminological dictionary

- collection of **terminological entries** (3.8.2) presenting information related to **concepts** (3.2.1) or **designations** (3.4.1) from one or more specific **subject fields** (3.1.2) [ISO 1087-1 (3.7.1)]

### Equivalent Definition from MDA: (syn: model)

set of all the **expressions** of a single **system definition** of a **set of interrelated concepts** that are in the minds of, and shared among, the members of a semantic community, that are conceptualized for a specific purpose of a given subject field

- Additional characteristics true of all ISO 407 / 1087-1, 1087-2 **terminological dictionaries**:
  - always documents the special purpose language for the subject field
  - **designations** (3.4.1) always talk about (*denote*) things (**objects** 3.1.1) in the subject world and NOT anything else in the terminological data collection
  - meaning (**concept** 3.2.1) separated from expression (**designation** 3.4.1)
  - words comprising terms are in a natural language and chosen to be the best natural language equivalent of the meaning of the concept (*see ISO 704:2000 section 7.3 “Term Formation”*)
- Additional characteristics true of all SBVR **terminological dictionaries**:
  - always documents the special purpose language of the semantic community
  - semantic community is identified and unambiguously defined, often identified by reference to a discipline, subject field, or one or more organization units

**NOTE:** all term references taken from ISO 1087-1:2000 or ISO 1087-2:2000

# Definitions Based on MDA Foundation Model and ISO 1087-1

---

## **ontology**

description of a universe of discourse in a language that a computer can process  
[ISO/IEC 19763-3 (4.2.2)]

## **natural language ontology (syn: formal terminology)**

- More General Concept
  - **ontology**
- Delimiting Characteristics
  - conceptualized in a same way the semantic community thinks & communicates about it
  - with terms and names formed from natural language

### Equivalent Definition from ISO 1087-1:

- More General Concept
  - **terminological dictionary**
- Delimiting Characteristics
  - defined in a way that gives it an adequate interpretation in formal logic

## **business ontology (syn: formal business terminology)**

- More General Concept
  - **natural language ontology**
- Delimiting Characteristics
  - subject world of the natural language ontology is any aspect of the context, design, management or operation of an organization and its products/services, except for Business Requirements for Software specification and software development

# Business Requirements for Software Terms

---

## specification

a detailed precise presentation of something or of a plan or proposal for something [Merriam-Webster Collegiate Dictionary 2a]

## requirement

- something that is needed for a particular purpose (*often used in the plural*) [Encarta]

## business requirements for software specification (syn: business requirements for software)

- More General Concept
  - specification
  - requirement
- Delimiting Characteristics
  - The thing for which requirements are being specified is a software system
  - the particular purpose for the requirements is the automation of selected information processing and business decision-making by the organization for which the software is a business resource
  - the content of the requirements specification includes:
    - Software capability required
      - » That for which the software should do something
      - » What the software should do with that
    - Software quality required
      - » How well the software should be materialized
      - » How well the software should do what it does

# Business Requirements for Software Terms

---

## consolidated data & rules requirements

part of a **business requirements for software specification** that contains only *that for which the software should do something*, including:

- Kinds of Business Facts with Reference Schemes and their requirement context, e.g. bottom level activity for essential business processes
  - NEED to KNOW
  - NEED to REMEMBER
  - NEED in OFFICIAL/LEGAL DOCUMENT
  - NEED for AUDIT (*intermediate data between system inputs and outputs*)
- Terminological entries which provide the business semantics for the business facts, including concepts covering events and states
- Structural Rules made explicit from intensional definitions and modified for recording requirements
- Time for which business facts must be true and timeliness (*recording lag from occurrence*)
- Data integrity points required; e.g. *month-end closing, input batches, etc.*
- Behavioural rules

# Some Key MDA Terms (1)

---

## **class-of-platform independent model:**

model that is independent of any particular class of platform

e.g. specific to document, relational database, object-oriented database, messaging system, etc.

## **class-of-platform specific model (syn: vendor platform independent model):**

model that is optimized for with a given class of platform, but independent of any specific vendor platform;

e.g. specific to relational database, but independent of Oracle, IBM, Microsoft, etc.

## **vendor platform specific model:**

model that is optimized for a single vendor's platform of a given class of platform;

e.g. HP Phototsmart 3310 All-in-One; SQL Server; Versant; MQ Series

## Some Key MDA Terms (2)

---

### **software model**

- More General Concept
  - **system model** that models the class of system: software

### **class-of-platform independent data model**

- More General Concepts
  - **software model** that is a **class-of-platform independent model** that contains only structures for recording/storing data

## Some Key MDA Terms (3)

---

### **class-of-platform specific data model**

**software model** that is a **class-of-platform specific model**  
and contains only structures for recording/storing data

### **vendor platform specific data model**

**software model** that is a **vendor platform specific model**  
and contains only structures for recording/storing data

# Essential Business Model

---

## essential business model

**class-of-platform independent model** of the organization

- “Essential” (as in *of the essence*, not as in *necessary*)
- Independent of whether people or automated equipment will do the work
- Used to make these decisions
- Independent of any technologies
- Used to choose technologies and design the class-of-platform specific models

## essential business process model

**essential business model** that models business processes of the organization

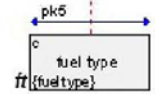
- Focuses on “bringing about new, original things, directly or indirectly by communication”\*
  - All tasks are about decisions, judgements, and engaging in commitments only
  - Included are the “**NEED to KNOW & NEED to REMEMBER**” requirements of bottom level **tasks** for **kinds of business facts** as defined in the **business ontology** (formal business terminology)
- Excludes all (*manual and automated*) information processing considerations
  - Excludes all information processing actions, transforms, movement and information storage
  - Does not include document, data or information artefacts flowing through the tasks of the process

\*see “performa” on pages 105-106, 144-148 in Dietz, Jan L.G. *Enterprise Ontology: Theory and Methodology*. Springer, 2006.

# **Appendix B.**

CogNIAM Notation Example

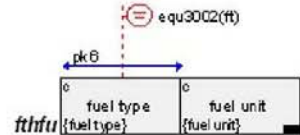
LOT fuel type {fuel type}



5: [fuel type] <fuel type> identifies a specific [fuel type]

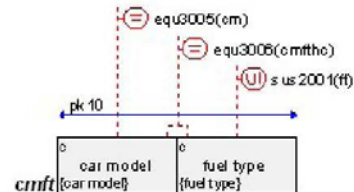
val5002: {gasoline, diesel, LPG, electricity}

fuel type has fuel unit {fuel type has fuel unit}



6: [fuel type] <fuel type> has [fuel unit] <fuel unit>

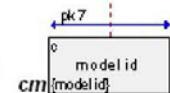
car model fuel type {car model fuel type}



10: [car model] <car model> is user of [fuel type] <fuel type>

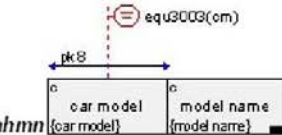
12: [fuel type] <fuel type> is used by [car model] <car model>

car model {car model}



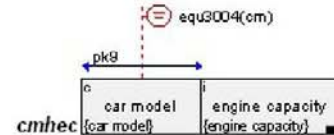
7: [model id] <model id> identifies a specific [car model]

car model has engine capacity {car model has engine capacity}

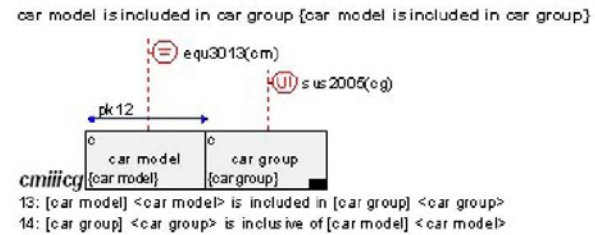
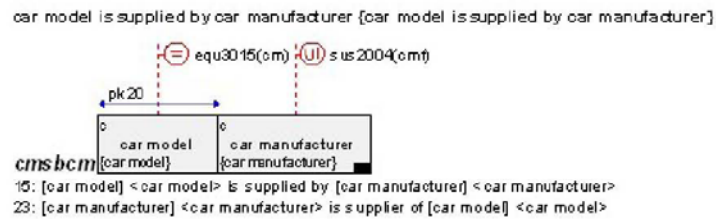
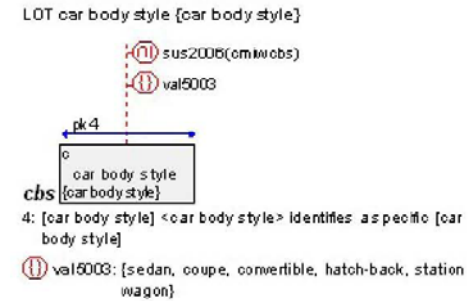
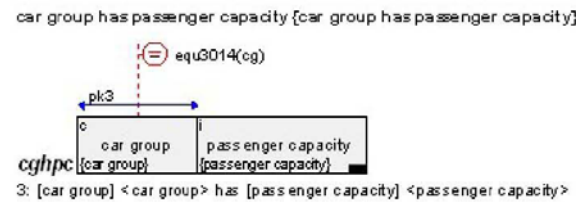
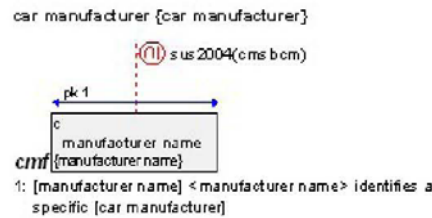
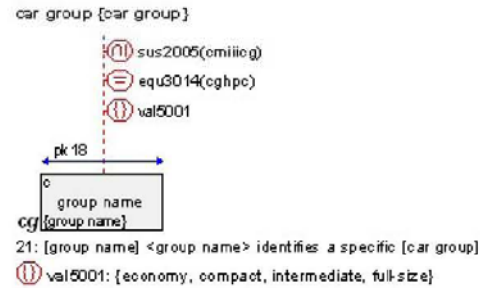


8: [car model] <car model> has [model name] <model name>

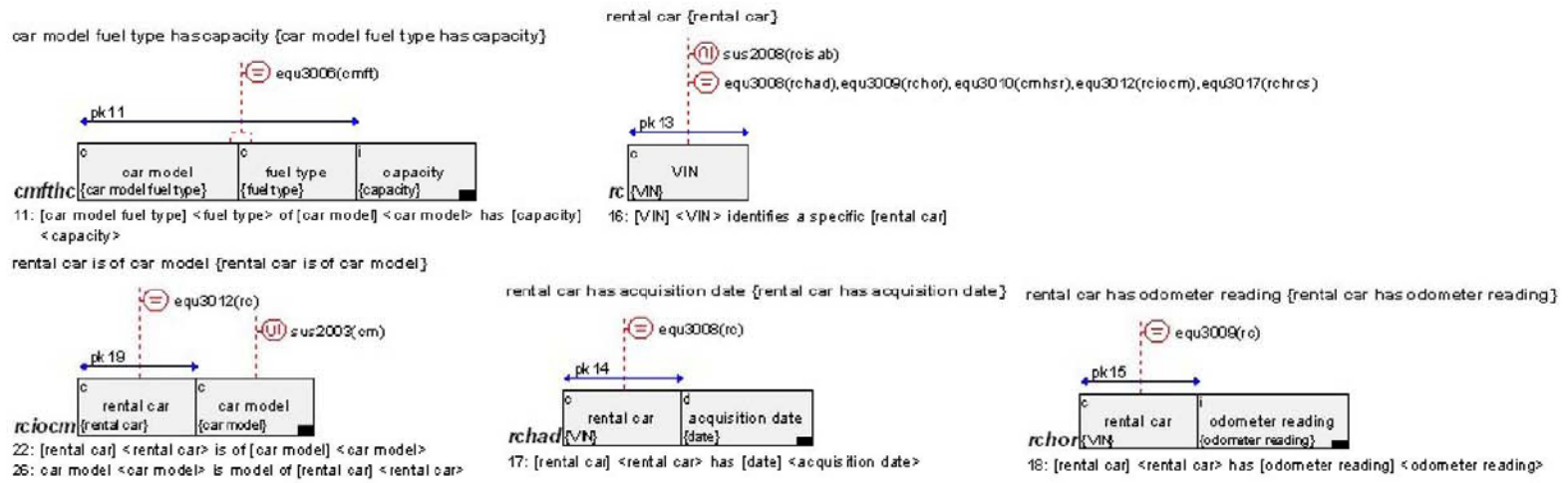
car model has engine capacity {car model has engine capacity}



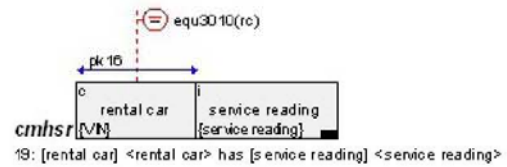
9: [car model] <car model> has [engine capacity] <engine capacity>



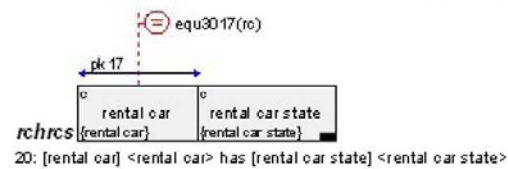




rental car has service reading {rental car has service reading}



rental car has rental car state {rental car has rental car state}



rental car is stored at branch {rental car is stored at branch}

