Capturing Industrial Information Models with Ontologies and Constraints

Evgeny Kharlamov
Senior Research Fellow
Department of Computer Science
University of Oxford

B. Cuenca Grau, E. Jimenez-Ruiz, S. Lamparter,
G. Mehdi, M. Ringsquandl, Y. Nenov, S. Grimm
M. Roshchin, Ian Horrocks

Semantic Tutorial. 2017. Oxford

Smart Factory

Automation

- of various individual processes
  - production
  - warehouse

(Enterprise-wide) integration

- of machines and processes
- factory as one organism

Control

- machines and processes
- monitoring, analytics, and diagnostics

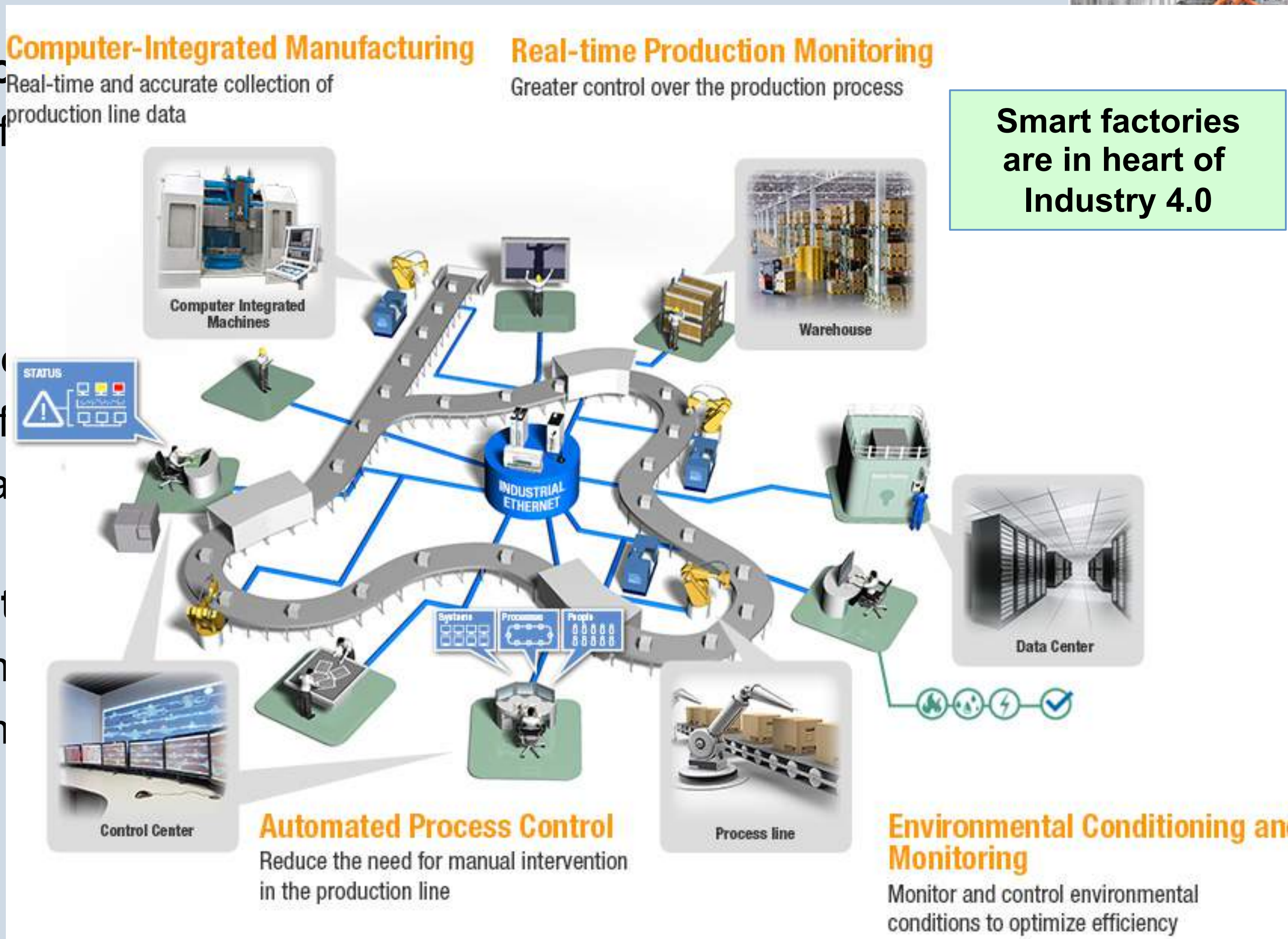
UNIVERSITY OF OXFORD

Smart Factory

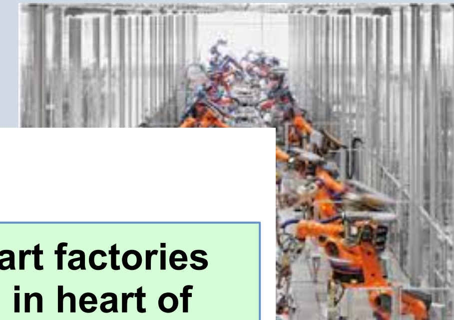
Auto
- of

(Ente
- of
- fa

Cont
- m
- m



Smart factories are in heart of Industry 4.0

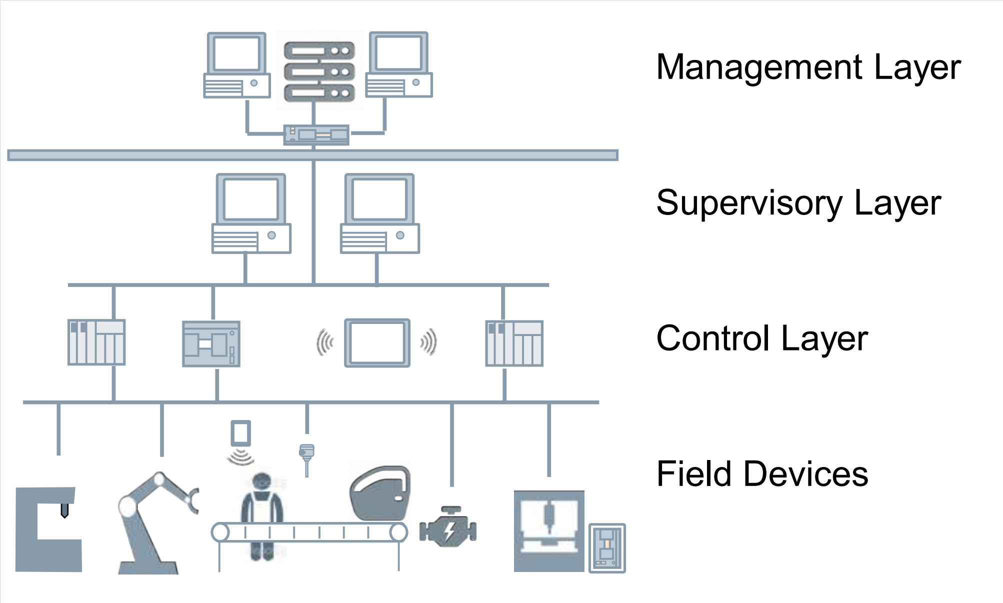


UNIVERSITY OF OXFORD

Software View on Smart Factories

Smart factory is

- fully computerized
- software-driven (system)

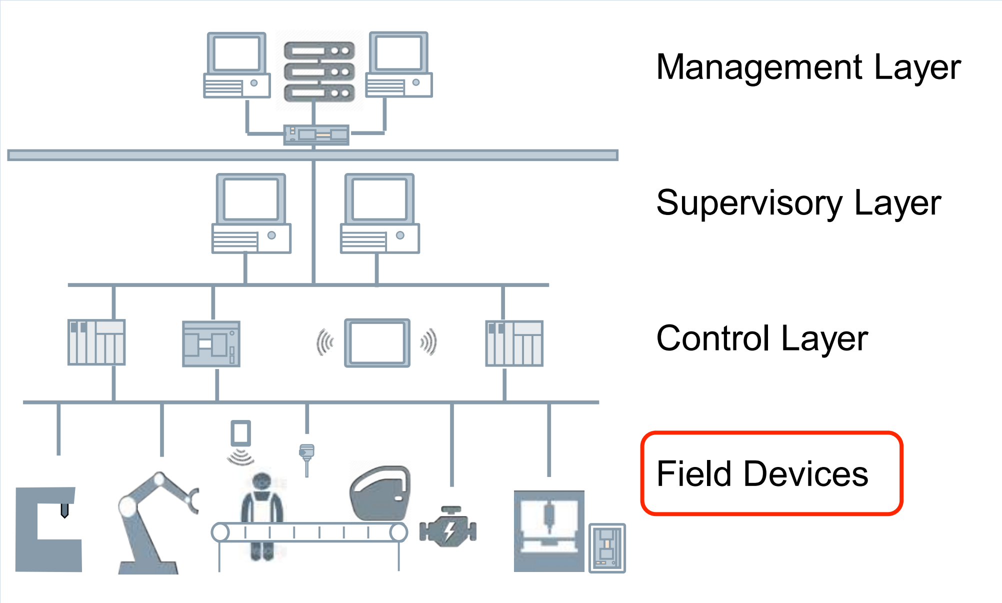


Software View on Smart Factories

Smart factory is
- fully computerized
- software-driven (system)

Software levels
- embedded in machines

Ex: Conveyor belt system
- simple controlling
  - positioning
  - speed
  - safety: emergency stop



Conveyor belt system

Software View on Smart Factories

Smart factory is

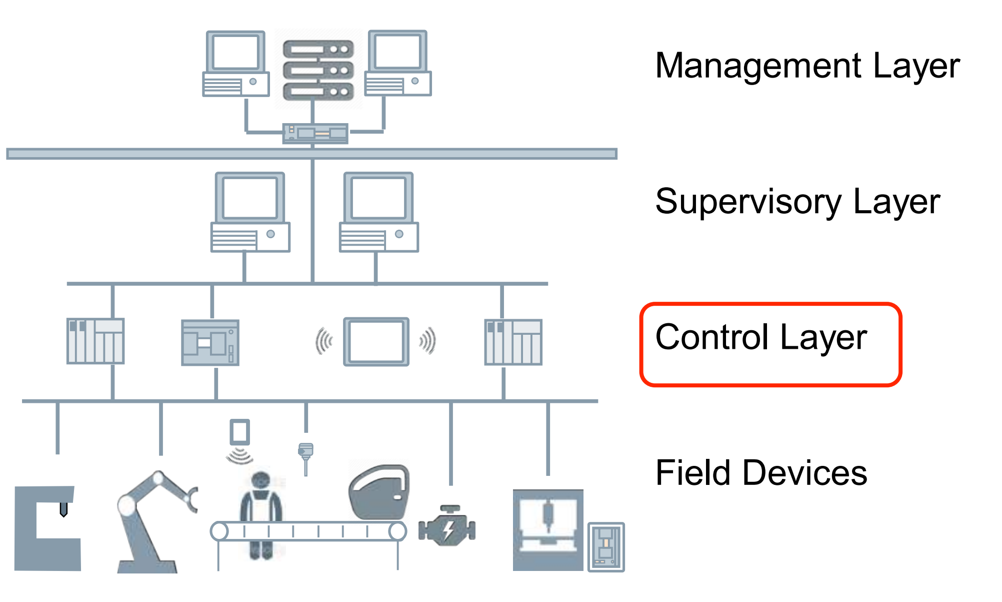
- fully computerized
- software-driven (system)

Software levels

- embedded in machines
- controlling several machines

Ex: Manufacturing conveying sub-system

- combines
  - Conveyer belt system
  - Routing system
  - Storage system
- orchestrated by complex controllers



Mechatronics Sub-System with Siemens PLC

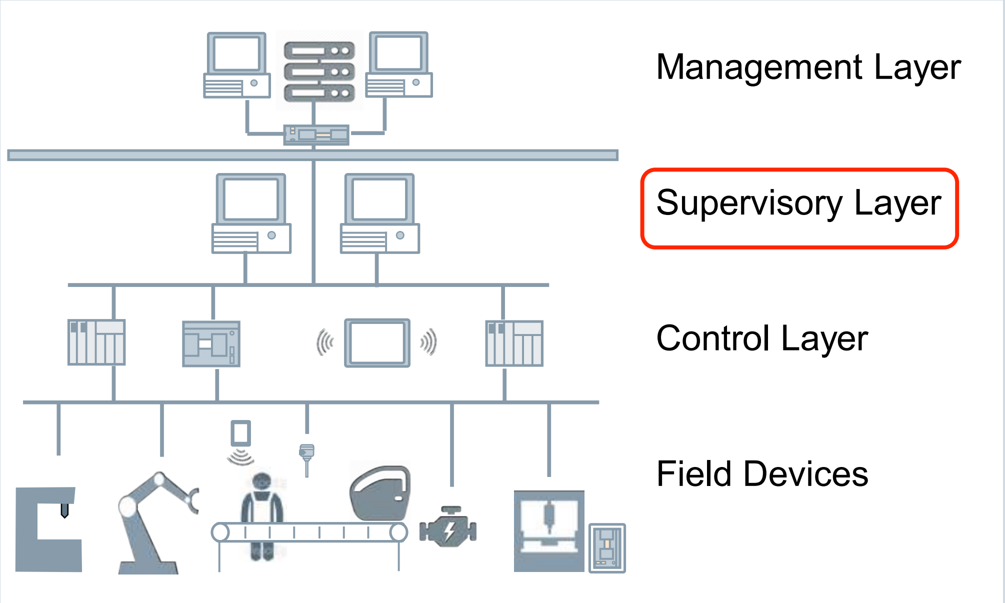
Software View on Smart Factories

Smart factory is

- fully computerized
- software-driven (system)

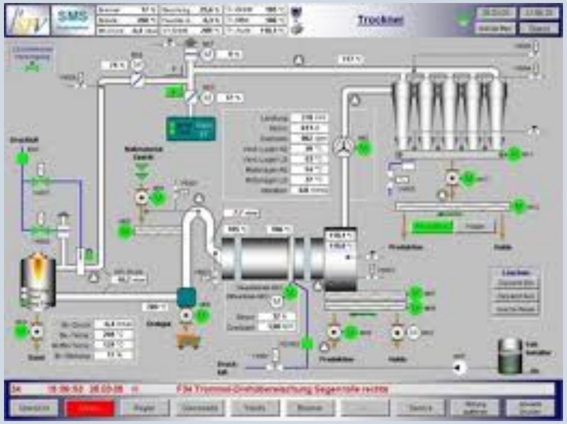
Software levels

- embedded in machines
- controlling several machines
- controlling the whole plant



Supervisory level

- plant-wide
  - integration
  - orchestration of processes
- plant-wide
  - monitoring
  - diagnostics of machines and processes



SCADA computer system

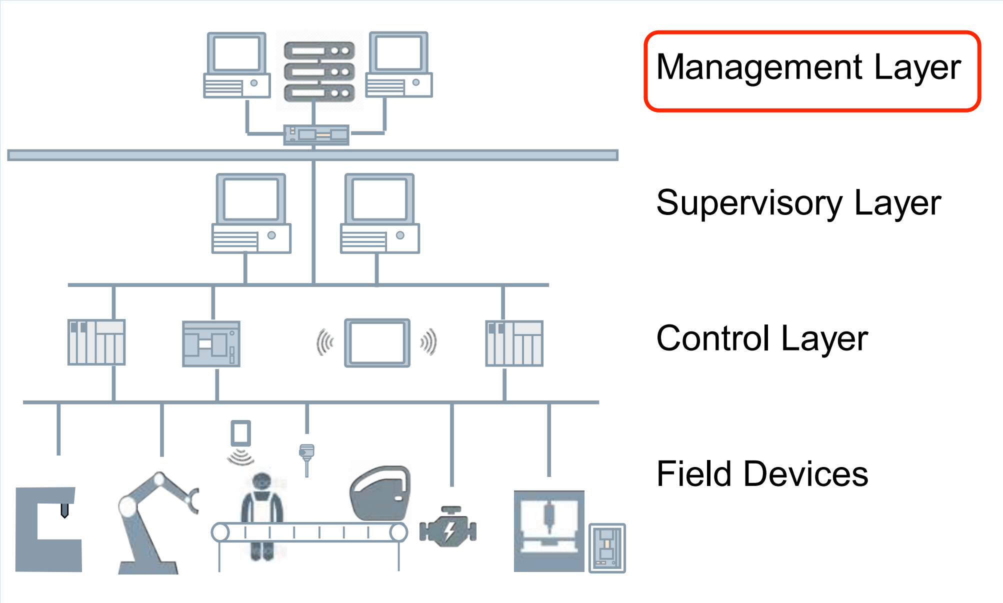
Software View on Smart Factories

Smart factory is

- fully computerized
- software-driven (system)

Software levels

- embedded in machines
- controlling several machines
- controlling the whole plant
- management level software
  - ERP
    - Manufacturing resource planning
    - Finance
    - Human resources



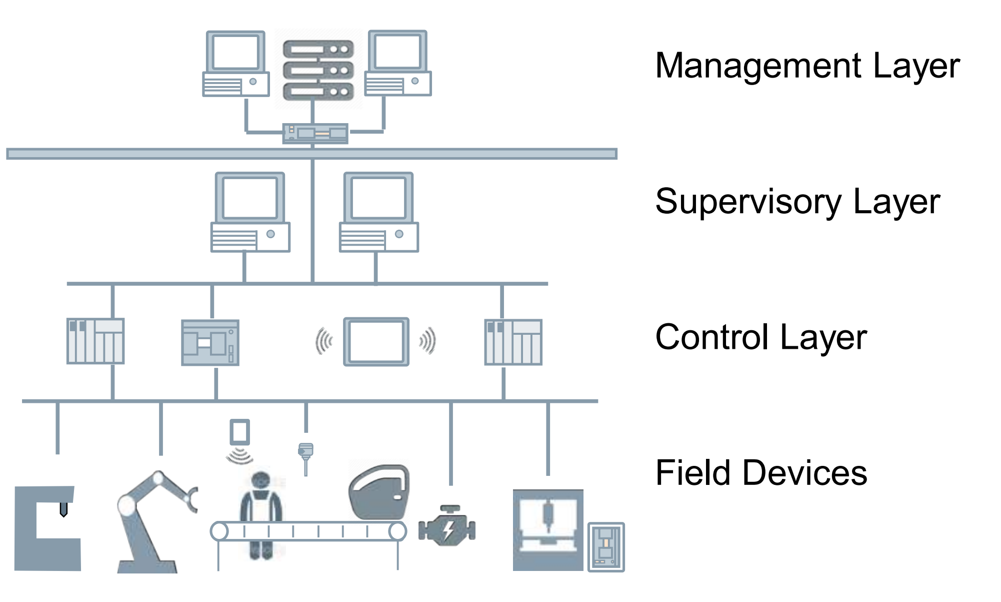
Software View on Smart Factories

Smart factory

- fully computerized
- software-driven (system)

Software levels

- embedded in machines
- controlling several machines
- controlling the whole plant
- management level software
  - ERP
    - Manufacturing resource planning
    - Finance
    - Human resources



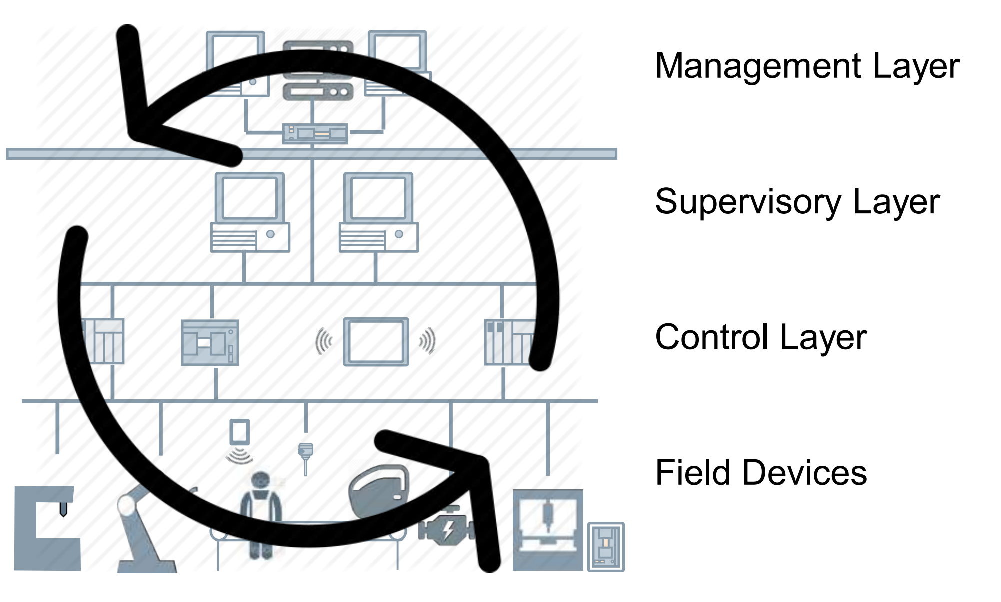
UNIVERSITY OF OXFORD

Software Challenges

Smart factory
fully computerized and software-driven

Challenges

- Software development
- Software integration



software development: ~40% of the price of manufacturing machines

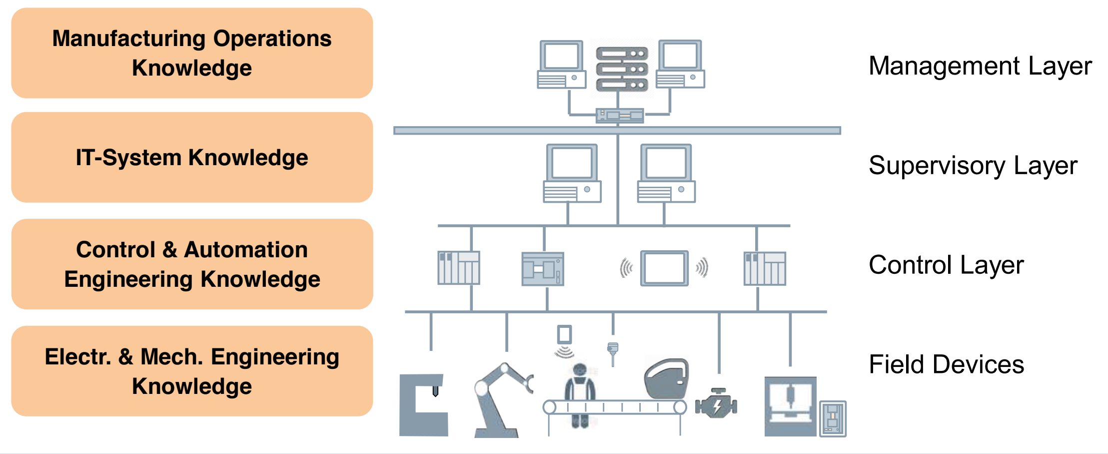
estimated by Mechanical Engineering Industry Association (VDMA) [2011]

UNIVERSITY OF OXFORD

Information Models for Smart Factories

Factory-wide info. models

- address challenges
  - SW development
  - SW integration
- capture knowledge on all SW levels

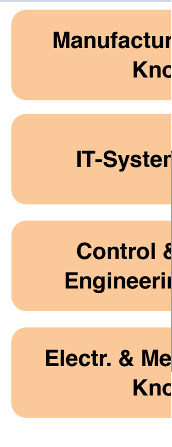


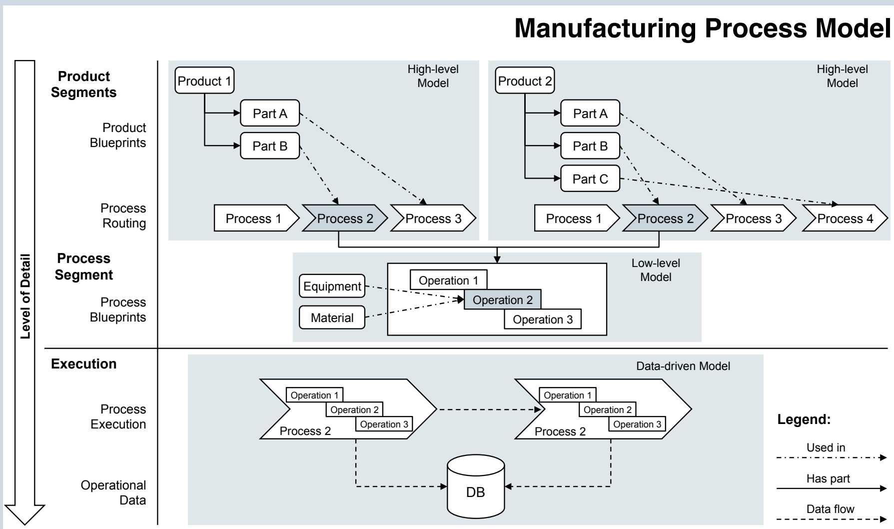
UNIVERSITY OF OXFORD

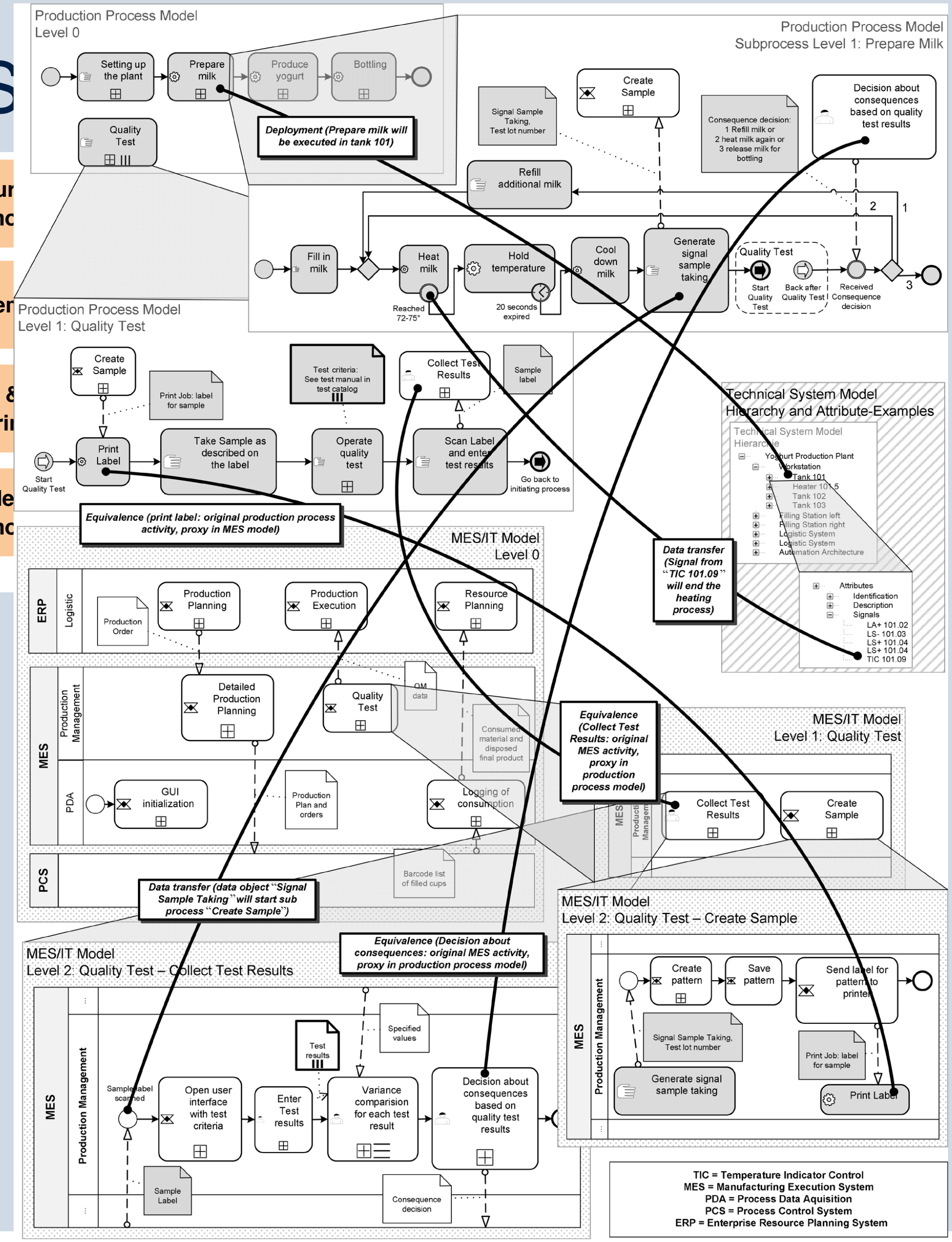
Information Models for S

Factory-wide info. models

- address challenges
  - SW development
  - SW integration
- capture knowledge on all SW levels





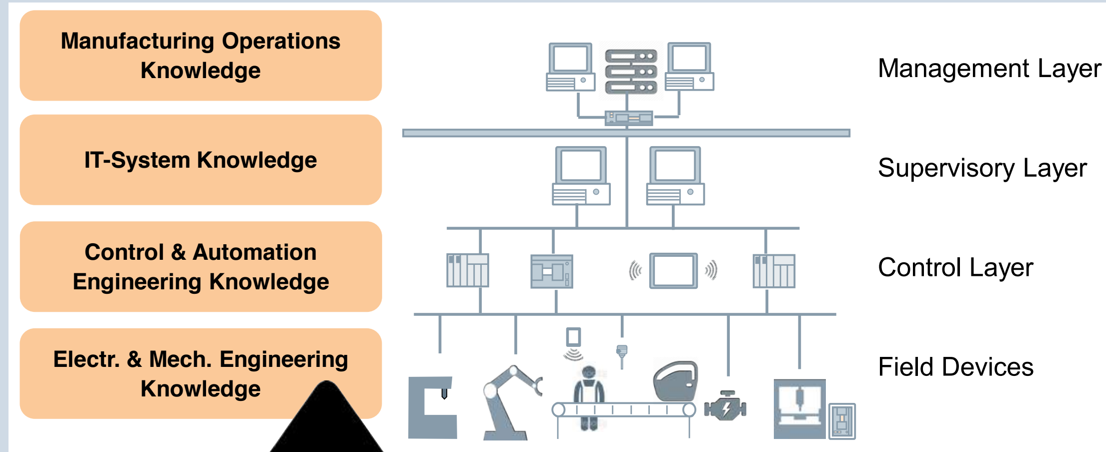


UNIVERSITY OF OXFORD

Information Models for Smart Factories

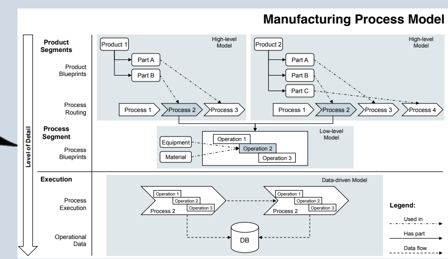
Factory-wide info. models

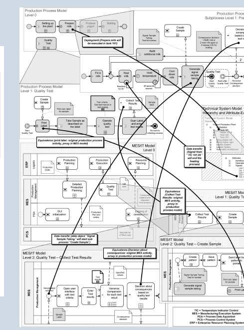
- address challenges
  - SW development
  - SW integration
- capture knowledge on all SW levels



Industrial standardisation is critical

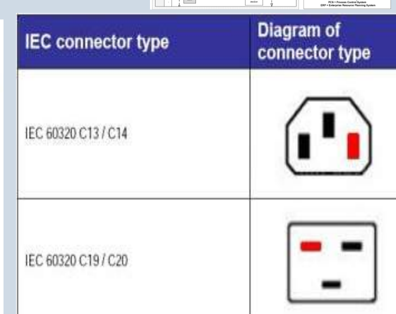
- ensures: safety, security, robustness, …
- sets "good practices" for industrial automation
- bases for industrial information models





International Organization for Standardization

INTERNATIONAL ELECTROTECHNICAL COMMISSION



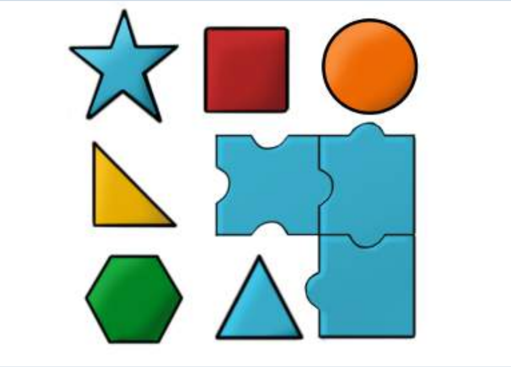
How well these models solve the problems?

UNIVERSITY OF OXFORD

Challenges with Existing Information Models

Reality of Information Models

- many types of models co-exist in one factory
- often incompatible models
  - independently developed
  - use different (often incompatible) formats
  - come from different types of proprietary software
  - may not come with a well-defined semantics
  - specification can be ambiguous



Consequences

- applications
  - ad hoc customization for various models
  - loosely integrated
- model management is a nightmare
  - development
  - maintenance
  - integration

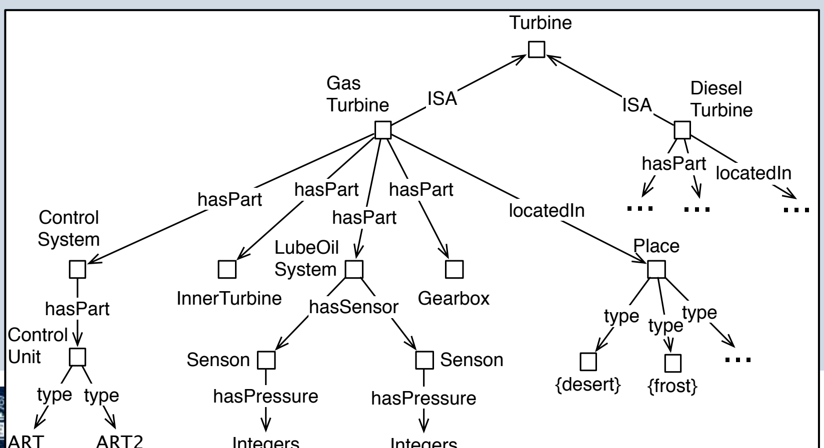
Can Semantic Technologies make life easier?

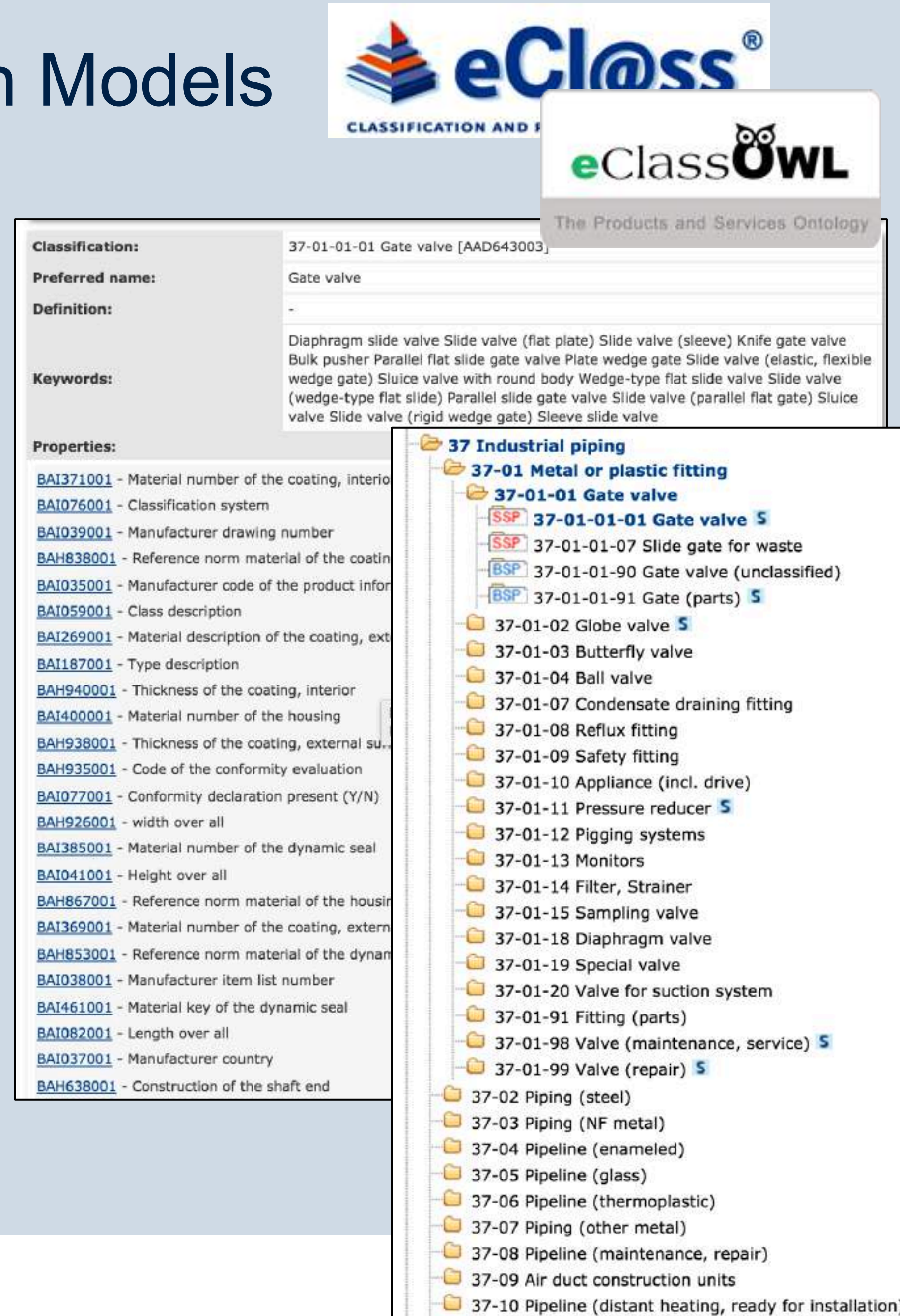
UNIVERSITY OF OXFORD

Ontologies as Information Models

Industrial Adoption of Sem. Tech.

- A lot of research
- Industry started adapting Sem Tech
  - Statoil, Aibel, Siemens
- OWL 2 and RDF Benefits
  - W3C standard
  - a lot of tooling
  - clear (machine process.) semantics
  - flex. data standard: storing, exch.





Outline

Intro

- Smart factories and the role of software
- Industrial information models to facilitate smart factories
- Ontologies as industrial information models

Our project

- goals
- achievements

Capturing
Industrial Information Models with
Ontologies and Constraints

Our Project Goals

Goals
1. Onto language for IIM
2. Concrete ontologies
3. Modelling methodology and tooling

1. Ontology language for industrial info. models

- better understanding
- set foundations for ontologies capturing
  - master data ~ industrial standards
  - domain specific model ~ concrete factories
- study
  - expressiveness
  - management tasks: ontology and data oriented
  - algorithms: to efficiently accomplish the tasks

2. Concrete ontologies

- to show modeling capabilities and practical benefits for industry

3. Modelling Methodology and Tooling

- cost efficient for creation & management of IIM – w/o SWeb background

Our Achievements

Goals
1. Onto language for IIM
2. Concrete ontologies
3. Modelling methodology and tooling

Ontology language for IIM
- expressiveness
- algorithms

Concrete ontologies
- 2 ontologies
- experiments

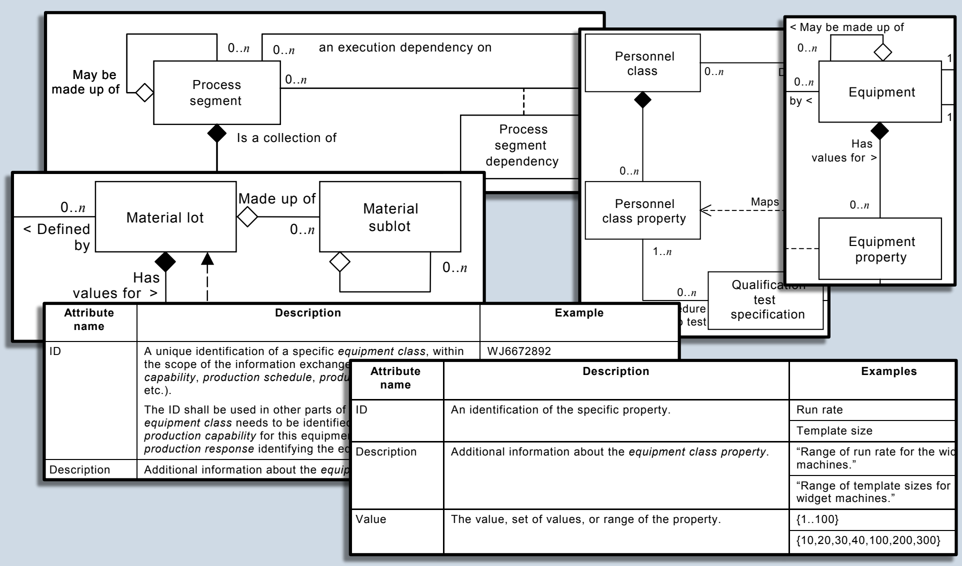
Modeling methodology and tooling
- SOMM systems

Ontology Language for Industrial Info Models

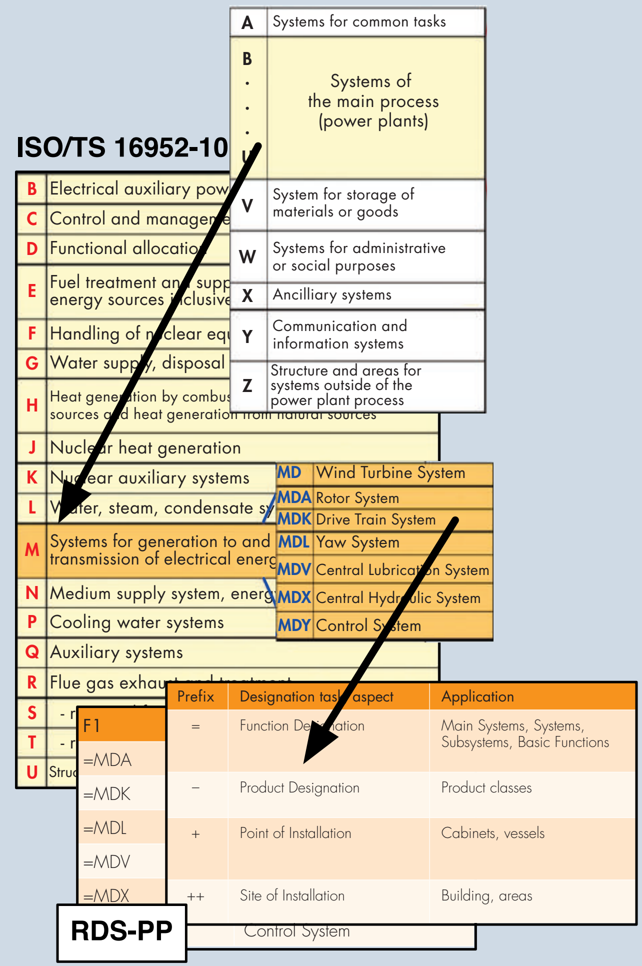
Analyzed two (sets of) industrial standards

- Manufacturing
  - IEC 62264 → ISA 88 and ISA 95
- Energy
  - IEC 81346 → ISO/TS 16952-10
    → RDS PP and KKS
- Consolidated modeling requirements

ISA 88/95

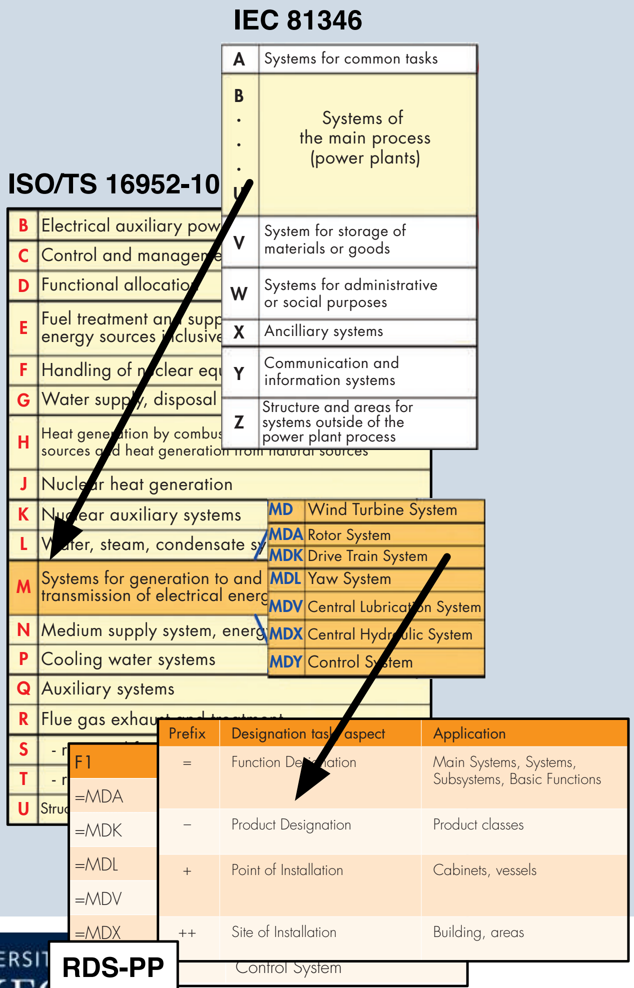


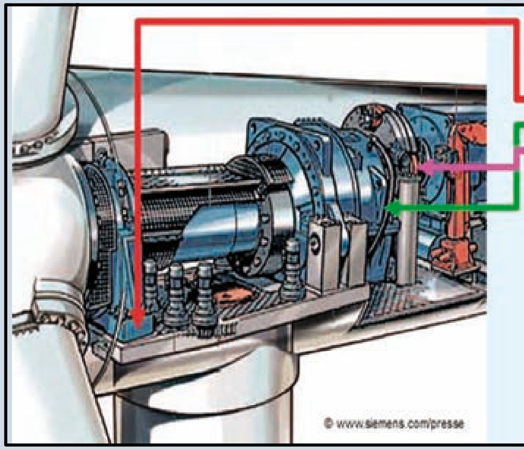
IEC 81346



Energy

IEC 81346
→ ISO/TS 16952-10
→ RDS PP and KKS

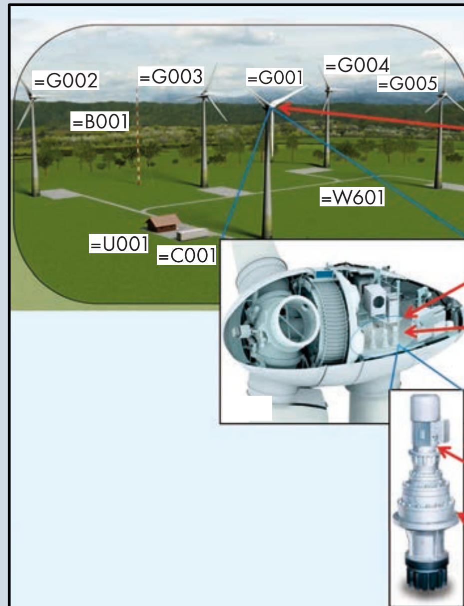




F1	Denomination
=MDK10	Rotor Bearing System
=MDK20	Speed Conversion System
=MDK30	Brake System Drive Train
=MDK40	Torque Transmission High Speed Shaft
=MDK50	Auxiliary Systems Drive Train
=MDK51	Main Gear Oil System
=MDK52	Offline Gear Oil System
=MDK53	Common Oil Lubrication System Drive Train
=MDK54	Rotor Lock Drive Train
=MDK55	Rotor Slewing Unit
=MDK56	Cooling System Drive Train

Wind Turbine Model

Wind Power Plan Model



Conjoint designation for Wind Power Plant: **#5154N00883E.DE_NW.ELI_1WN**

Main system designation e.g. for Wind Turbine Generator: **=G001**

System designation e.g. for Yaw System: **=G001 MDL**

Subsystem designation e.g. for Yaw Drive System: **=G001 MDL10**

Basic Function designation e.g. for Yaw Drive 1: **=G001 MDL10 MZ010**

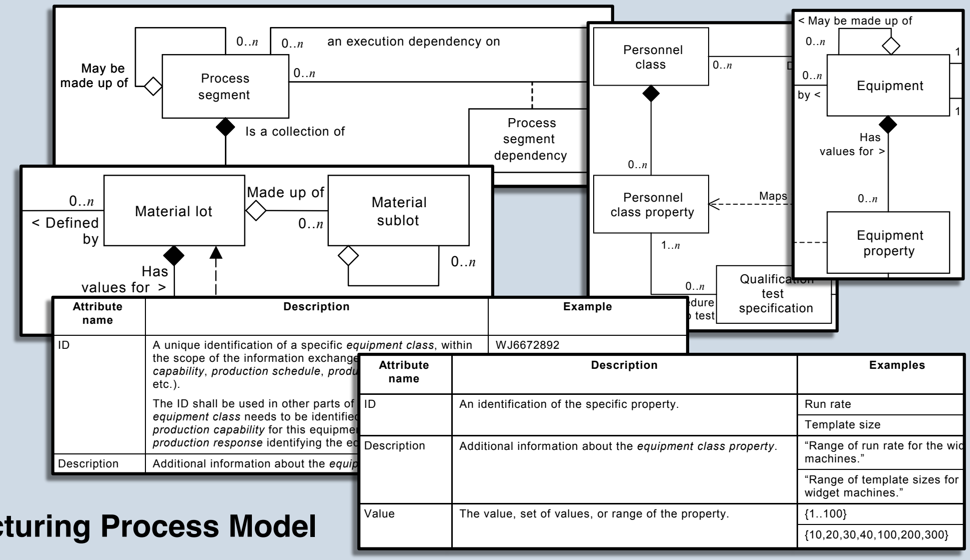
Product designation e.g. for Yaw Motor 1: **=G001 MDL10 MZ010–MA001**

Product designation e.g. for Yaw Gear 1: **=G001 MDL10 MZ010–TL001**

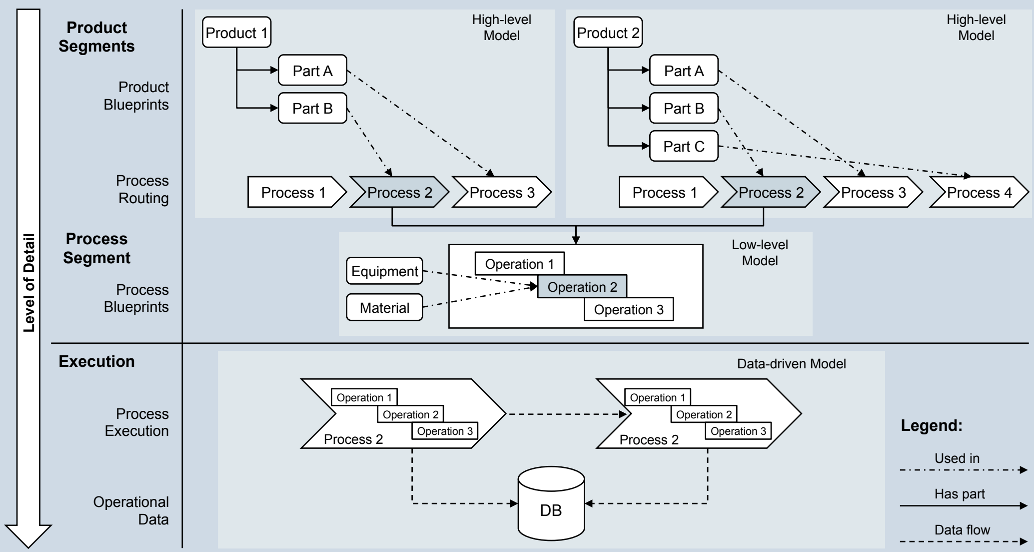
Manufacturing

IEC 62264
→ ISA 88 and ISA 95

ISA 88/95



Manufacturing Process Model

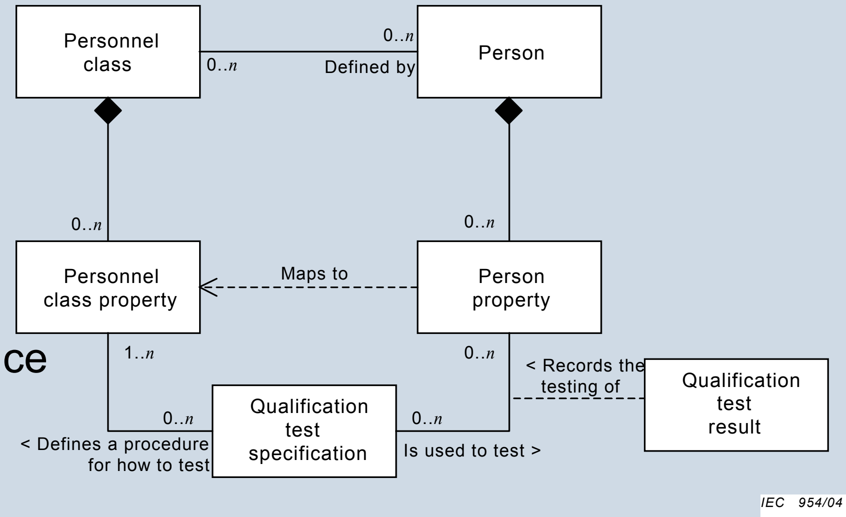


UNIVERSITY OF OXFORD

How to Turn an ISA standard into an Ontology?

Vocabulary

- Classes
  - Main class, subclasses,
    type of provenance, prov. Results
- Properties
  - Data, Object, Annotation, with provenance



Axioms

- Properties “attached” to classes with
  - Typing
  - Default values
  - Uniqueness of prop. values
  - “Required” property
  - Cardinality restriction (?)
- Disjoint classes
  - E.g. personnel & equipment

Table 5 – Attributes of person

Attribute name	Description	Examples
ID	A unique identification of a specific person, within the scope of the information exchanged (*production capability, production schedule, production performance*, etc.). The ID shall be used in other parts of the model when the *person* needs to be identified, such as the *production capability* for this person, or a *production response* identifying the person.	999-123-4567 Jane W Smith – #2 Employee 23
Description	Additional information about the resource.	“Person information” “Person information” “Person information”
Name	The name of the individual. This is meant as an additional identification of the resource, but only as information and not as a unique value.	Joe Smith III Jane Bubba

Someone has to make a design choice on how to interpret a standard

UNIVERSITY OF OXFORD

A Possible Interpretation of ISA 88-95

ISA 88-95 modules

- Person, Equipment, Material

Classes:

- Person, PersonClass, PersonProvenanceType PersonProvenanceTest
- Engineer, Plummer, etc

Properties

- ID, Description, Name
- DoB, Address

Attached properties for Person

- ID (Int): compulsory, unique
- Name (String)

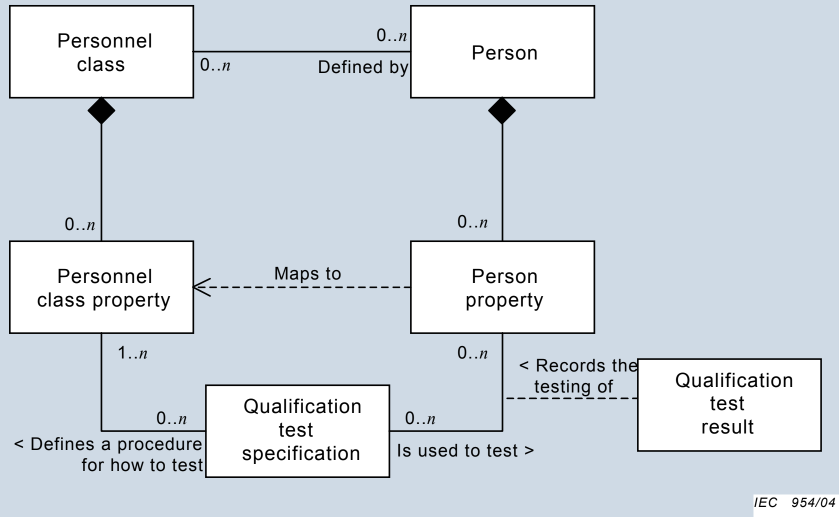


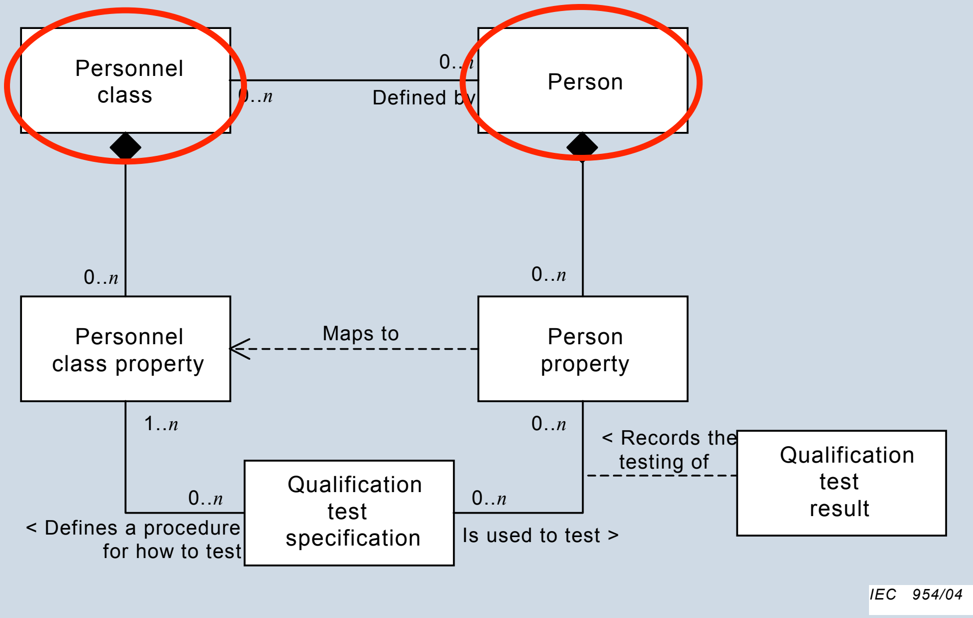
Table 5 – Attributes of person

Attribute name	Description	Examples
ID	A unique identification of a specific person, within the scope of the information exchanged (*production capability, production schedule, production performance*, etc.). The ID shall be used in other parts of the model when the *person* needs to be identified, such as the *production capability* for this person, or a *production response* identifying the person.	999-123-4567 Jane W Smith – #2 Employee 23
Description	Additional information about the resource.	"Person information" "Person information" "Person information"
Name	The name of the individual. This is meant as an additional identification of the resource, but only as information and not as a unique value.	Joe Smith III Jane Bubba

Someone has to make a design choice on how to attach properties to classes

UNIVERSITY OF OXFORD

Classes



Class

- “Person” = {bob, john, …}

Class of classes (?) or jobs (?)

- “Personnel class” = {Engineers, Pilots, …}

Modeling in OWL

Class: Person

Class: Engineer, Pilot
SubClassOf: Person

Class: Personnel

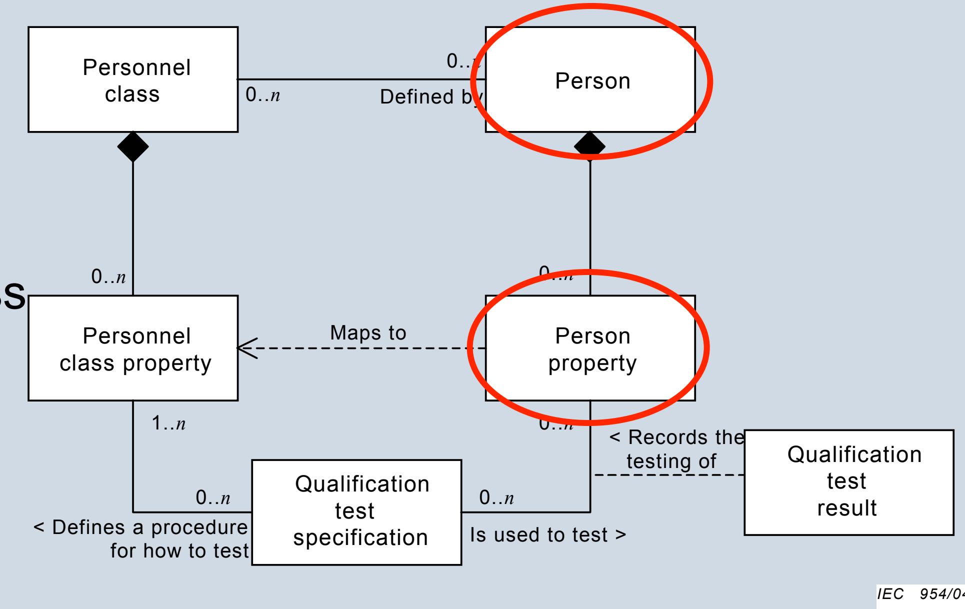
Individual: Engineer
Types: Personnel

Individual: Pilot
Types: Personnel

Design choice is not trivial

UNIVERSITY OF OXFORD

Properties of Objects



Attributes: for objects

- "Default" for objects of a given class
  - E.g.: *Bob* has ID, Desc, Name
- Extra properties of objects (for objects of a given class)
  - Person prop. = {age, friend-of}
  - Defined via "templates"
  - *Age* has ID, Desc, Value, V. unit of meas.

Table 5 – Attributes of person

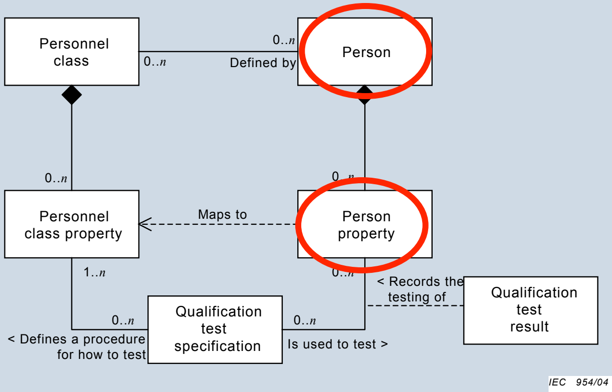
Attribute name	Description	Examples
ID	A unique identification of a specific person, within the scope of the information exchanged (*production capability, production schedule, production performance*, etc.). The ID shall be used in other parts of the model when the *person* needs to be identified, such as the *production capability* for this person, or a *production response* identifying the person.	999-123-4567 Jane W Smith – #2 Employee 23
Description	Additional information about the resource.	"Person information" "Person information" "Person information"
Name	The name of the individual. This is meant as an additional identification of the resource, but only as information and not as a unique value.	Joe Smith III Jane Bubba

Table 6 – Attributes of person property

Attribute name	Description	Examples
ID	An identification of the specific property.	Class 1 certified Exposure hours available Pager number
Description	Additional information about the *person property*.	"Indicates if the person is class 1 certified widget assembly operator" "Indicates number of exposure hours available this month" "Pager number"
Value	The value, set of values, or range of the property. The value(s) is assumed to be within the range or set of defined values for the related *personnel class property*.	True 4 800-555-1212
Value unit of measure	The unit of measure of the associated property value, if applicable.	Boolean h Phone number

UNIVERSITY OF OXFORD

Properties of Objects



Default

Class: Person
HasKey: ID
SubClassOf: Description <u>min</u> 1, description <u>only</u> string
SubClassOf: Name <u>exactly</u> 1, Name <u>only</u> string

Template

ObjectProperty: PagerNumber
Annotations: ID "pager number"
Description "descr of pager number"
Domain: Person
Range: PhoneNumber

AnnotationProperty: ID
Domain: PagerNumber, …
Range: integer[> 0]

Table 5 – Attributes of person

Attribute name	Description	Examples
ID	A unique identification of a specific person, within the scope of the information exchanged (*production capability, production schedule, production performance*, etc.). The ID shall be used in other parts of the model when the *person* needs to be identified, such as the *production capability* for this person, or a *production response* identifying the person.	999-123-4567 Jane W Smith – #2 Employee 23
Description	Additional information about the resource.	"Person information" "Person information" "Person information"
Name	The name of the individual. This is meant as an additional identification of the resource, but only as information and not as a unique value.	Joe Smith III Jane Bubba

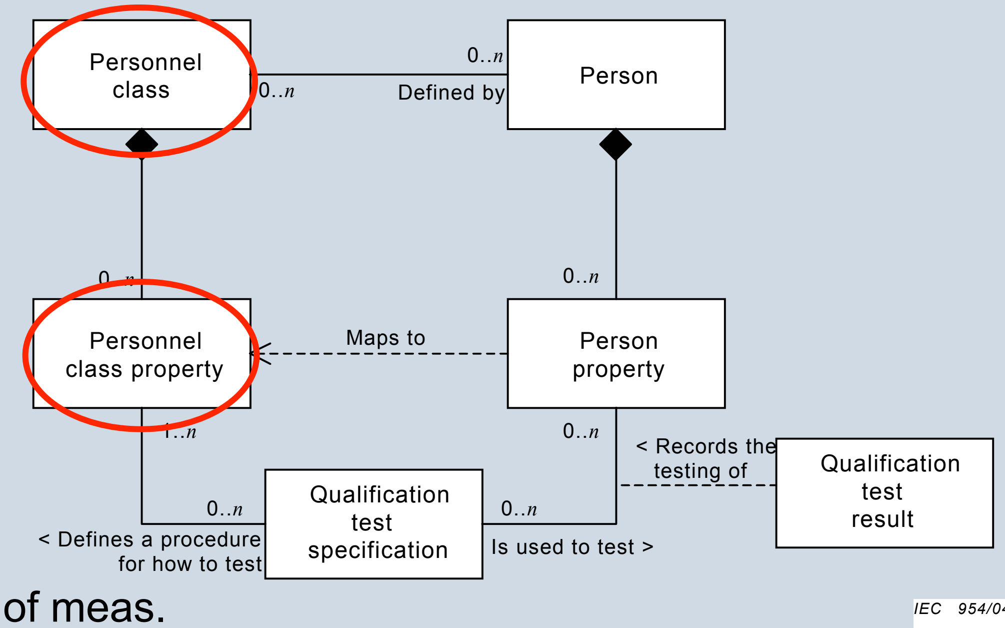
Table 6 – Attributes of person property

Attribute name	Description	Examples
ID	An identification of the specific property.	Class 1 certified Exposure hours available Pager number
Description	Additional information about the *person property*.	"Indicates if the person is class 1 certified widget assembly operator" "Indicates number of exposure hours available this month" "Pager number"
Value	The value, set of values, or range of the property. The value(s) is assumed to be within the range or set of defined values for the related *personnel class property*.	True 4 800-555-1212
Value unit of measure	The unit of measure of the associated property value, if applicable.	Boolean h Phone number

HasKey is not a constraint:
does not enforce explicit ID for data

UNIVERSITY OF OXFORD

Properties of Classes



Attributes: for lasses

- “Default” for specific class
- *Bob* has ID, Desc, Name

- Extra properties for classes
  - Person prop. = {age, friend-of}
  - Defined via “templates”
  - *Age* has ID, Desc, Value, V. unit of meas.

Table 5 – Attributes of person

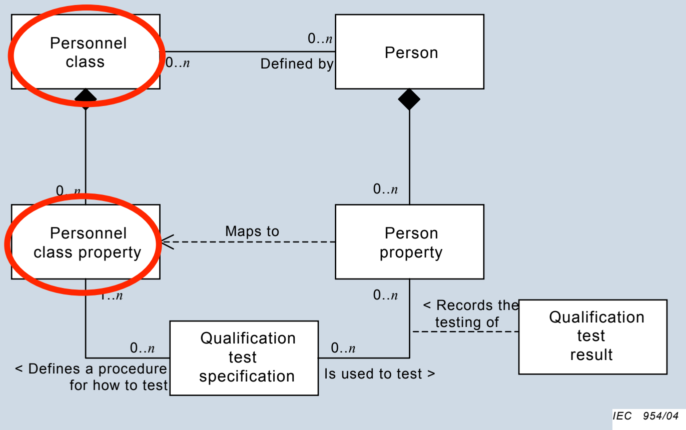
Attribute name	Description	Examples
ID	A unique identification of a specific person, within the scope of the information exchanged (*production capability, production schedule, production performance*, etc.). The ID shall be used in other parts of the model when the *person* needs to be identified, such as the *production capability* for this person, or a *production response* identifying the person.	999-123-4567 Jane W Smith – #2 Employee 23
Description	Additional information about the resource.	“Person information” “Person information” “Person information”
Name	The name of the individual. This is meant as an additional identification of the resource, but only as information and not as a unique value.	Joe Smith III Jane Bubba

Table 6 – Attributes of person property

Attribute name	Description	Examples
ID	An identification of the specific property.	Class 1 certified Exposure hours available Pager number
Description	Additional information about the *person property*.	“Indicates if the person is class 1 certified widget assembly operator” “Indicates number of exposure hours available this month” “Pager number”
Value	The value, set of values, or range of the property. The value(s) is assumed to be within the range or set of defined values for the related *personnel class property*.	True 4 800-555-1212
Value unit of measure	The unit of measure of the associated property value, if applicable.	Boolean h Phone number

UNIVERSITY OF OXFORD

Properties of Classes



Default

AnnotationProperty: ID
Domain: PersonnerlClass
Range: string

AnnotationProperty: Description
Domain: PersonnerlClass
Range: string

Template

DataProperty: ClassOneCertified
Annotations: ID "Class One Certified"
Description "Indicates the …"
Domain: Engineer
Range: Boolean

Class: Engineer
SubClassOf: ClassOneCertified exactly 1 and
ClassOneCertified exists {true, false}
ClassOneCertified only {true, false}

Table 4 – Attributes of personnel class property

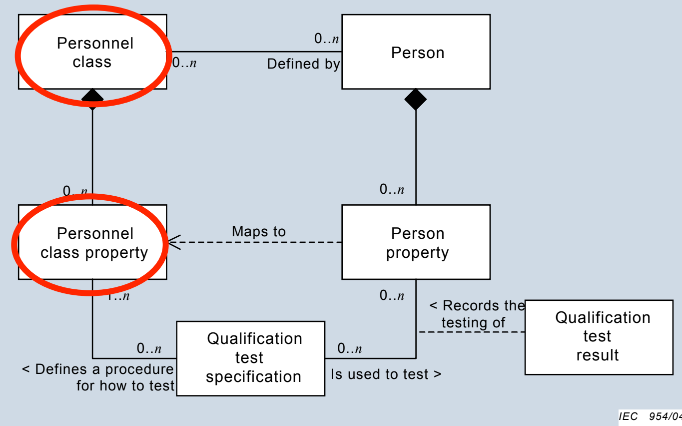
Attribute name	Description	Examples
ID	An identification of the specific property, unique under the scope of the parent *personnel class* object. For example, the property "*has class 1 safety training*" (with values of *yes* or *no*) may be defined under several different *personnel class* definitions, such as *fork lift operator* and *pipe fitter* classes, but has a different meaning for each class.	Class 1 certified Night shift available Monthly exposure hours maximum
Description	Additional information and description about the *personnel class property*.	"Indicates the certification level of the operator." "Indicates if operator is available for night shift." "Indicates the maximum monthly exposure hours that can be used."
Value	The value, set of values, or range of the property. This presents a range of possible numeric values, a list of possible values, or it may be empty if any value is valid.	{True, False} {True, False} [0..20]
Value unit of measure	The unit of measure of the associated property values, if applicable.	Boolean Boolean h

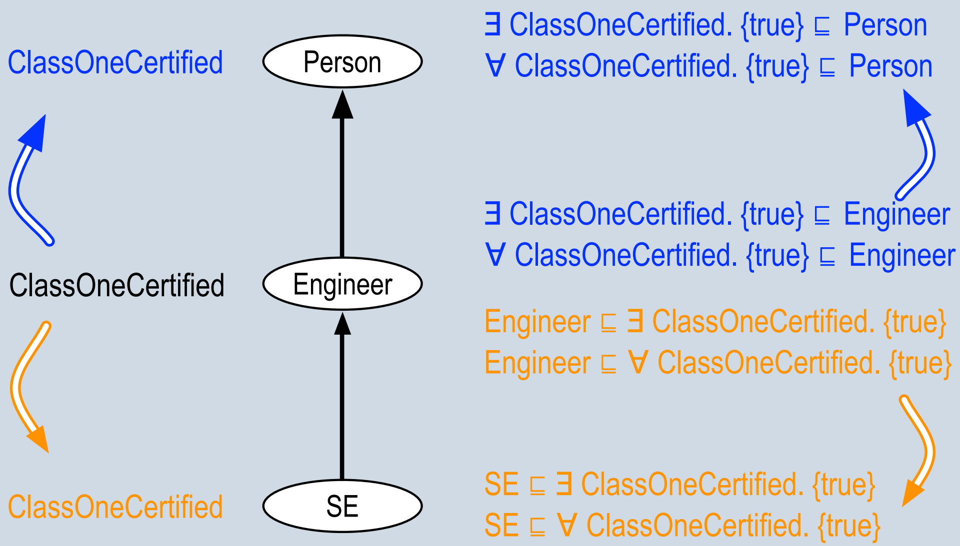
Table 3 – Attributes of personnel class

Attribute name	Description	Example
ID	A unique identification of a specific *personnel class*. These are not necessarily job titles, but identify classes that are referenced in other parts of the model.	Widget assembly operator
Description	Additional information and description about the *personnel class*.	"General information about widget assembly operators."

UNIVERSITY OF OXFORD

Inheritance of Properties





Q: What kind of inheritance do we need?

A: May make sense to allow different options

Table 4 – Attributes of personnel class property

Attribute name	Description	Examples
ID	An identification of the specific property, unique under the scope of the parent *personnel class* object. For example, the property "*has class 1 safety training*" (with values of *yes* or *no*) may be defined under several different *personnel class* definitions, such as *fork lift operator* and *pipe fitter* classes, but has a different meaning for each class.	Class 1 certified Night shift available Monthly exposure hours maximum
Description	Additional information and description about the *personnel class property*.	"Indicates the certification level of the operator." "Indicates if operator is available for night shift." "Indicates the maximum monthly exposure hours that can be used."
Value	The value, set of values, or range of the property. This presents a range of possible numeric values, a list of possible values, or it may be empty if any value is valid.	{True, False} {True, False} [0..20]
Value unit of measure	The unit of measure of the associated property values, if applicable.	Boolean Boolean h

Table 3 – Attributes of personnel class

Attribute name	Description	Example
ID	A unique identification of a specific *personnel class*. These are not necessarily job titles, but identify classes that are referenced in other parts of the model.	Widget assembly operator
Description	Additional information and description about the *personnel class*.	"General information about widget assembly operators."

Ontology Language for Industrial Info Models

Axioms

- assigning (relevant) properties to classes
  - If-Then by default (A-quantifier)
  - influence type of inheritance
  - domains and ranges of properties

SubClassOf($Turbine$ $Equipment$)
SubDataPropertyOf($hasRotorSpeed$ $hasSpeed$)
TransitiveObjectProperty($hasPart$)
InverseObjectProperties($hasPart$ $partOf$)

SubClassOf($Conveying$
ObjectAllValuesFrom($followedBy$ $Packaging$))

Data Constraints

- Compulsory and default values
- # of compulsory values
- functional properties
- encoded as annotated standard axioms

SubClassOf($Turbine$ SomeValuesFrom(R B))
SubClassOf(A HasValue(R b))
SubClassOf(A MaxCardinality(n R B))
SubClassOf(A MinCardinality(n R B))
FunctionalProperty(R)

SubClassOf($Turbine$ ObjectSomeValuesFrom($hasPart$ $Rotor$))
SubClassOf($TwoRotorTurbine$ ObjectMinCardinality(2 $hasPart$ $Rotor$))
SubClassOf($TwoRotorTurbine$ ObjectMaxCardinality(2 $hasPart$ $Rotor$))

Boris Motik, Ian Horrocks, Ulrike Sattler:
Bridging the gap between OWL and RDBs. JWS'09

Algorithms: Reasoning, Data Validation

Separate axioms and constr.

- using annotations
- axioms: reasoning
- constraints: data validation

Encode in Datalog

- gives a unified framework for axioms and constraints

Choose the right system

- triple store or rule inference system
- supporting
  - Datalog reasoning and
  - stratified negation-as-failure
- IRIS, RDFOx, etc

OWL 2 Axiom	Datalog Rules
SubClassOf$(A\ B)$	$B(?x) \leftarrow A(?x)$
SubPropertyOf$(P_1\ P_2)$	$P_2(?x,?y) \leftarrow P_1(?x,?y)$
TransitiveObjectProperty(P)	$P(?x,?z) \leftarrow P(?x,?y) \wedge P(?y,?z)$
InverseObjectProperties(P_1, P_2)	$P_2(?y,?x) \leftarrow P_1(?x,?y)$ and $P_1(?y,?x) \leftarrow P_2(?x,?y)$
SubClassOf$(A$ AllValuesFrom$(P\ B))$	$B(?y) \leftarrow P(?x,?y) \wedge A(?x)$

OWL Axiom	Datalog rules
SubClassOf$(A$ SomeValuesFrom$(R\ B))$	$R\_B(?x) \leftarrow R(?x,?y) \wedge B(?y)$ and $Violation(?x,\alpha) \leftarrow A(?x) \wedge$ **not** $R\_B(?x)$
SubClassOf$(A$ HasValue$(R\ b))$	$Violation(?x,\alpha) \leftarrow A(?x) \wedge$ **not** $R(?x,b)$
FunctionalProperty(R)	$R\_2(?x) \leftarrow R(?x,?y_1) \wedge R(?x,?y_2) \wedge$ **not** $owl{:}sameAs(?y_1,?y_2)$ and $Violation(?x,\alpha) \leftarrow R\_2(?x)$
SubClassOf$(A$ MaxCardinality$(n\ R\ B))$	$R\_(n{+}1)\_B(?x) \leftarrow \bigwedge_{1\leq i\leq n+1} (R(?x,?y_i) \wedge B(?y_i))$ $\bigwedge_{1\leq i<j\leq n+1}$ (**not** $owl{:}sameAs(?y_i,?y_j))$ and $Violation(?x,\alpha) \leftarrow A(?x) \wedge R\_(n{+}1)\_B(?x)$
SubClassOf$(A$ MinCardinality$(n\ R\ B))$	$R\_n\_B(?x) \leftarrow \bigwedge_{1\leq i\leq n} (R(?x,?y_i) \wedge B(?y_i))$ $\bigwedge_{1\leq i<j\leq n}$ (**not** $owl{:}sameAs(?y_i,?y_j))$ and $Violation(?x,\alpha) \leftarrow A(?x) \wedge$ **not** $R\_n\_B(?x)$

UNIVERSITY OF OXFORD

Our Achievements

Goals
1. Onto language for IIM
2. Concrete ontologies
3. Modelling methodology and tooling

Ontology language for IIM
- formalization
- algorithms

Concrete ontologies
- 2 ontologies
- experiments

Modeling methodology and tooling
- SOMM systems

Ontologies

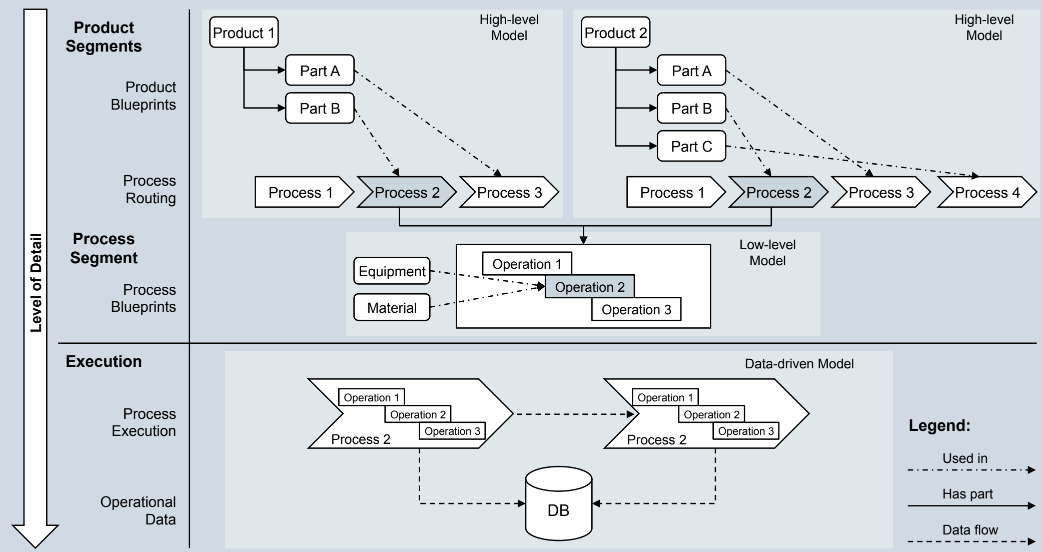
Manufacturing ontology

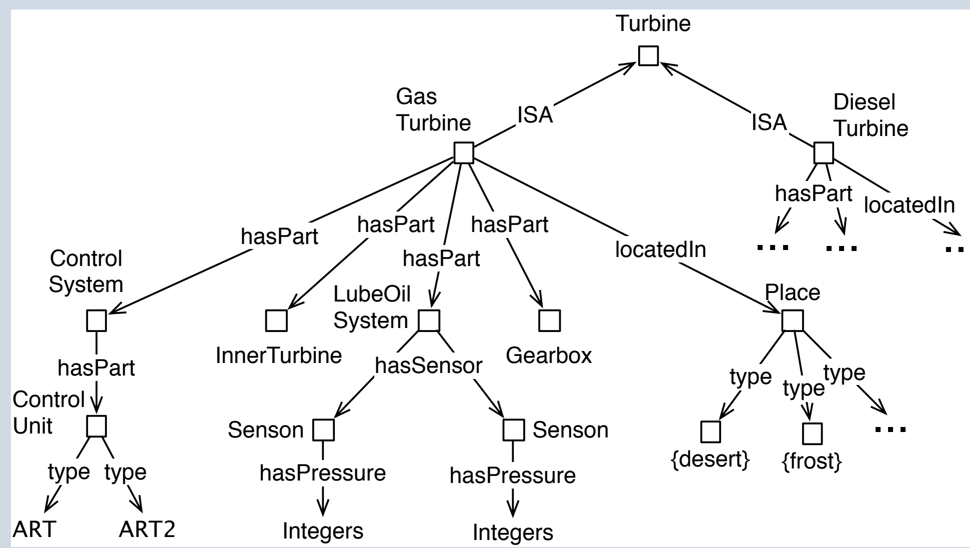
- based on IEC 62264
- 79 standard axioms
- 20 constraints

Turbine ontology

- based on IEC 81346
- 121 standard axioms
- 25 constraints

Manufacturing Process Model



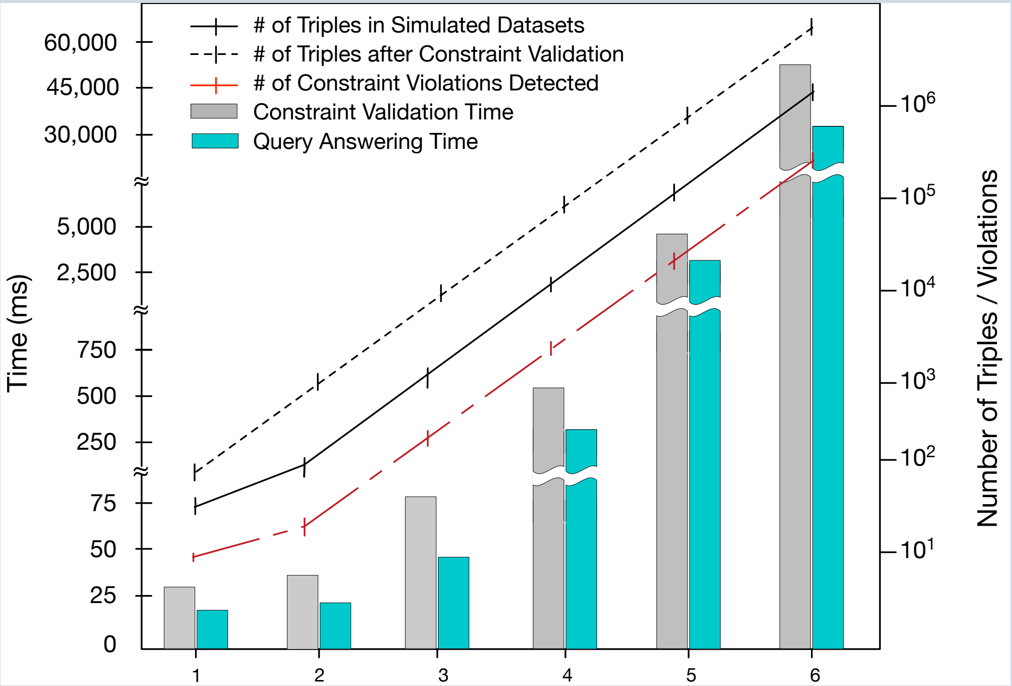


UNIVERSITY OF OXFORD

Manufacturing Experiment

Manufacturing data

- simulated by Siemens
- two types of products
- two configurations
  - manufacturing that violates the model specifications (too much material is used)
  - manufacturing according to specifications
- 6 data sets: 50 → 1x10^6



3 monitoring queries

- Q1: find all products that use material from a given lot
- Q2: find all material lots used in a given product
- Q3: find the total quantity of material in lots of a specific kind

Results

- C. validation, Q. answering is feasible on stock hardware: 87s over data datasets with ~1 million triples

UNIVERSITY OF OXFORD

Gas Turbine Experiment

Anonymized dataset

- from 800 real gas turbines
- sensor readings (temperature, pressure, rotor speed and position)
- associated processes (e.g., expansion, compression, start up, shut down)
- converted from a relational DB into RDF
- 25,090 triples over 4, 076 individuals.

3 monitoring queries

- Q1: find all core parts, equipment & current state of all turb. of a given type
- Q2: find all components involved in a compression process
- Q3: find temperature readings of turbines of a given type

Results

- Constraint checking and query answering: < 2s
- 1,582 constraint violations

Our Achievements

Goals

1. Onto language for IIM
2. Concrete ontologies
3. Modelling methodology and tooling

Ontology language for IIM

- formalization
- algorithms

Concrete ontologies

- 2 ontologies
- Experiments

Modeling methodology and tooling

- SOMM systems

See demo later today!

UNIVERSITY OF OXFORD

Summary

Goals
1. Onto language for IIM
2. Concrete ontologies
3. Modelling methodology and tooling

Use case analyses
- Smart factories and the role of info models
- Industrial standards
  - Manufacturing (IEC 62264), Energy (IEC 81346)

Foundations of ontology language to capture IIM
- Capturing with axioms and constraints
- Algorithms for constraint verification and query answering

Concrete ontologies
- 2 ontologies: Manufacturing, Energy
- experiments

Modeling methodology and tooling
- SOMM system