

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



UNIVERSITY OF
OXFORD

SIEMENS

Smart Factory

Automation

- of various individual processes
 - production
 - warehouse

(Enterprise-wide) integration

- of machines and processes
- factory as one organism

Control

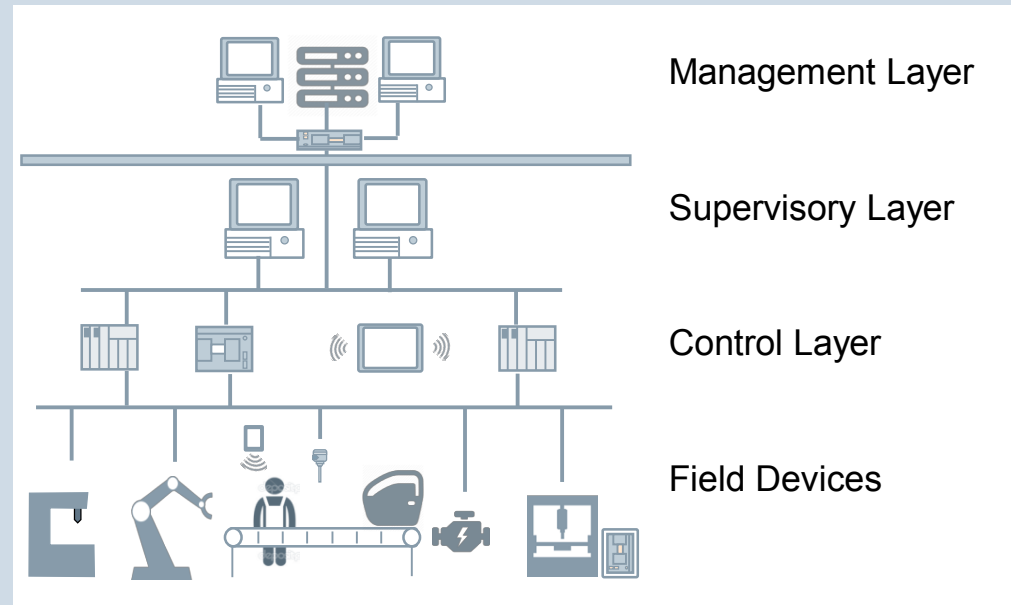
- machines and processes
- monitoring, analytics, and diagnostics



Software View on Smart Factories

Smart factory is

- fully computerized
- software-driven (system)



Software View on Smart Factories

Smart factory is

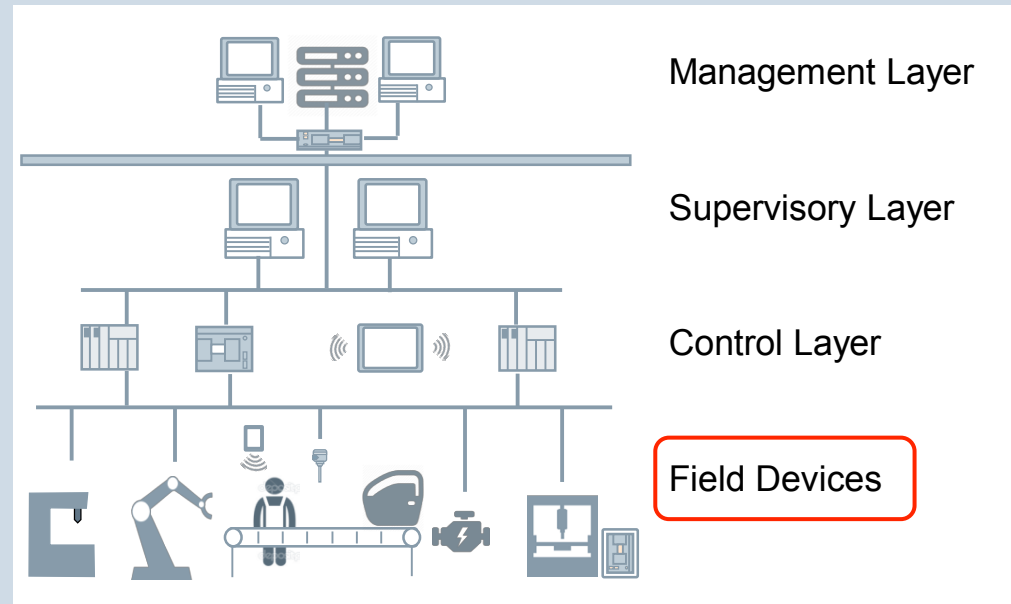
- fully computerized
- software-driven (system)

Software levels

- embedded in machines

Ex: Conveyor belt system

- simple controlling
 - positioning
 - speed
 - safety: emergency stop



Conveyor belt system

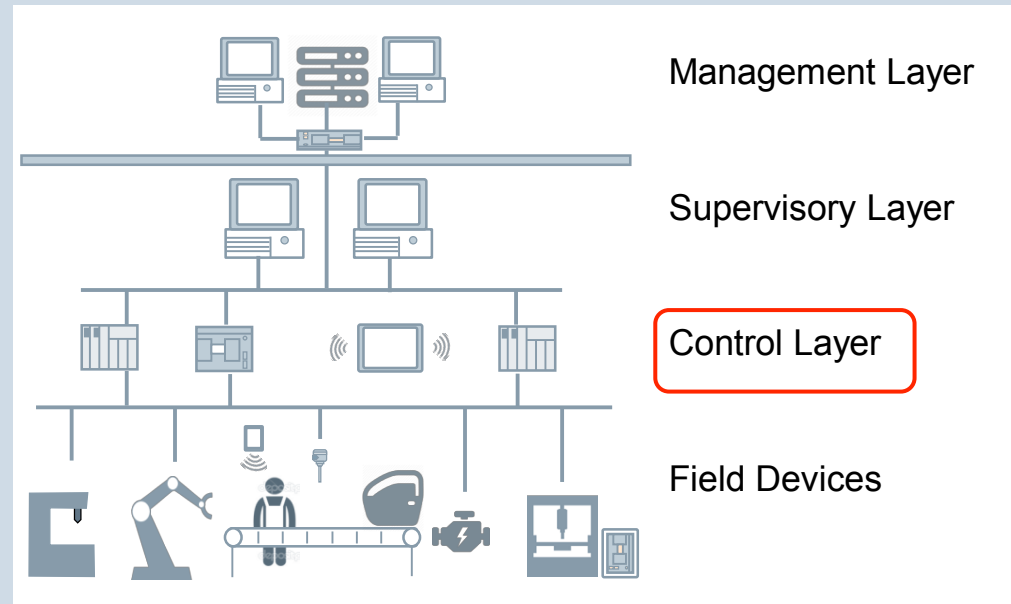
Software View on Smart Factories

Smart factory is

- fully computerized
- software-driven (system)

Software levels

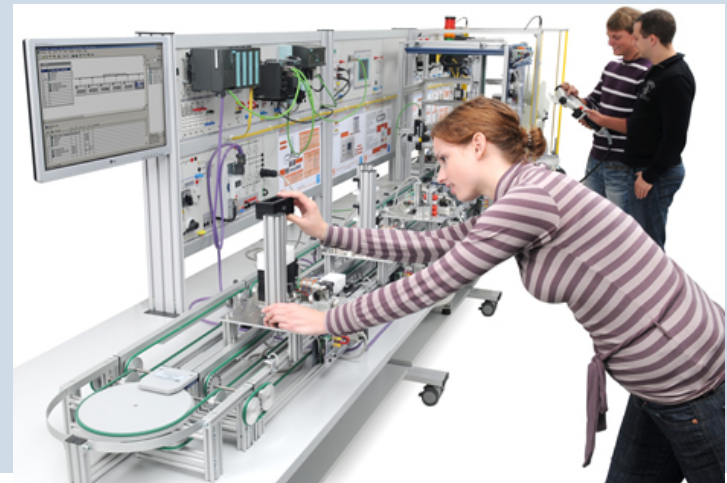
- embedded in machines
- controlling several machines



Ex: Manufacturing conveying sub-system

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

Mechatronics Sub-System with Siemens PLC



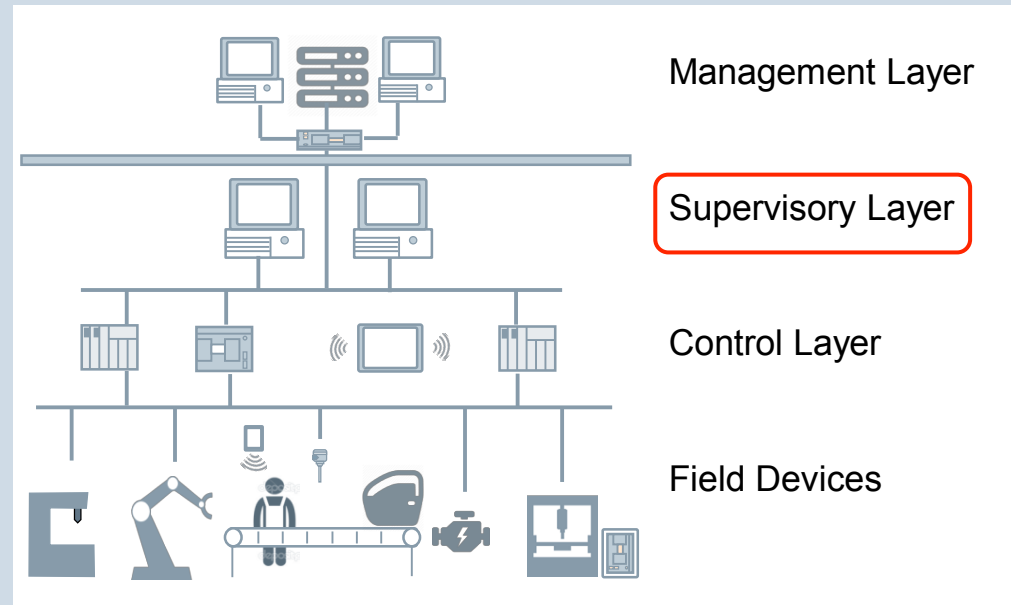
Software View on Smart Factories

Smart factory is

- fully computerized
- software-driven (system)

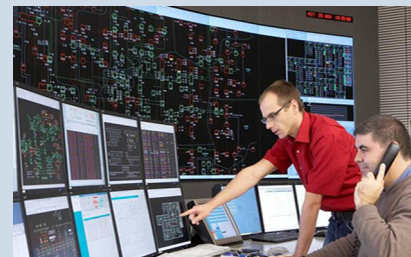
Software levels

- 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



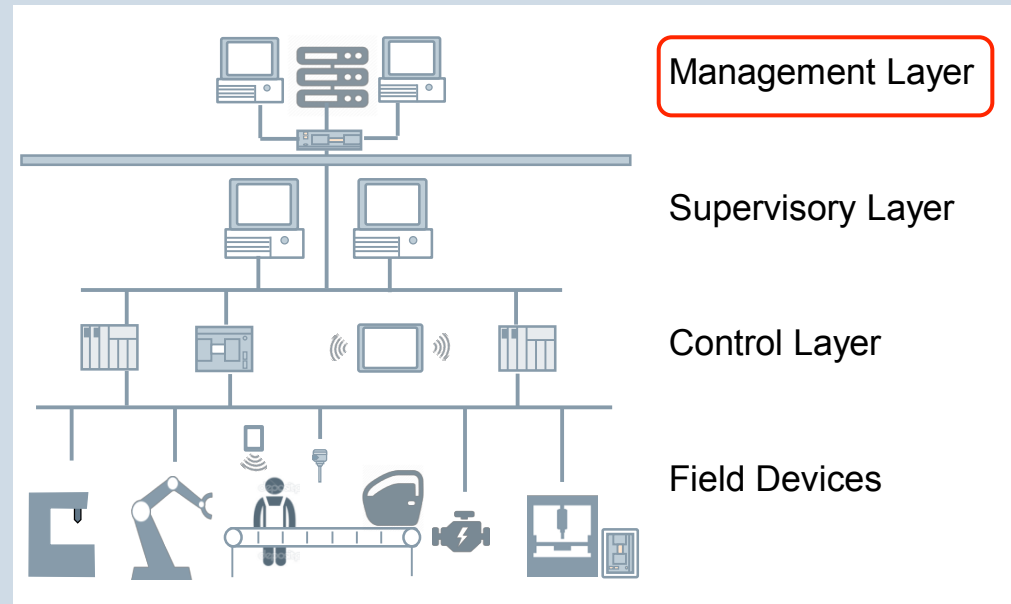
Software View on Smart Factories

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



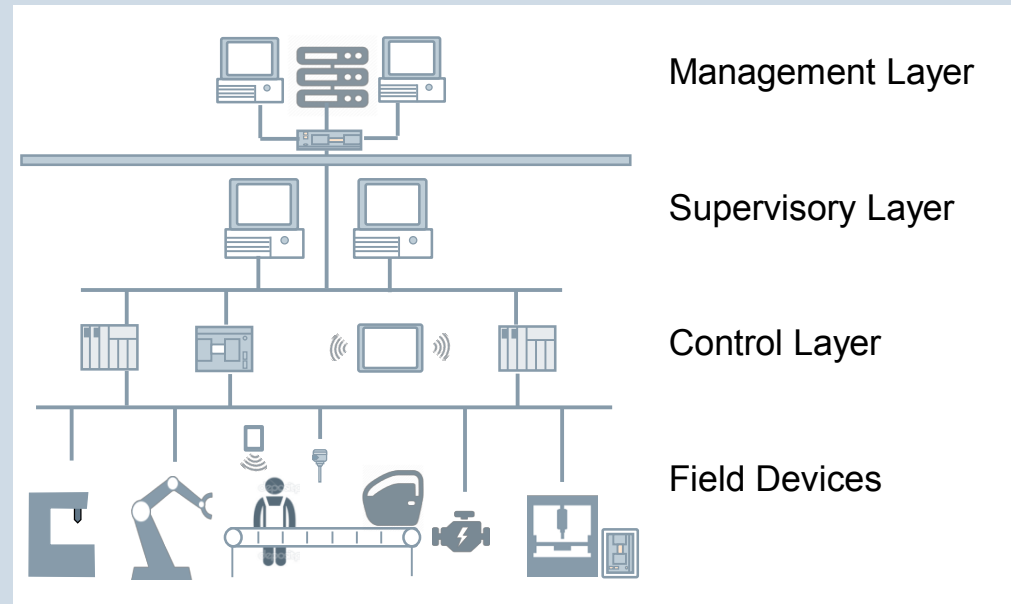
Software View on Smart Factories

Smart factory

- fully computerized
- software-driven (system)

Software levels

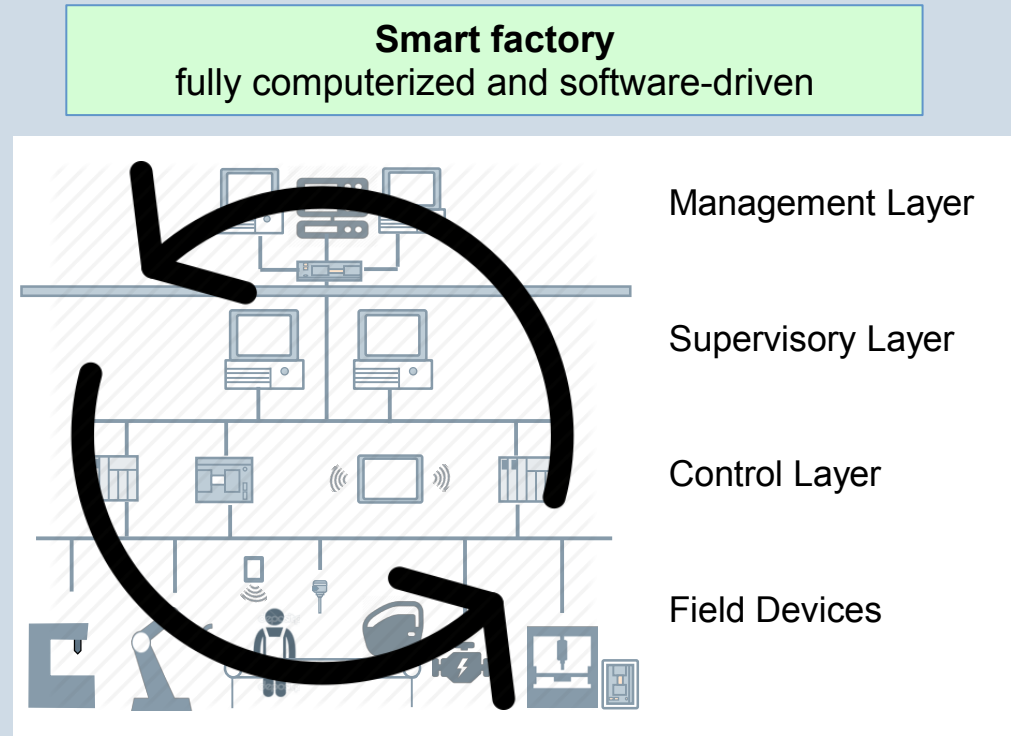
- embedded in machines
- controlling several machines
- controlling the whole plant
- management level software
 - ERP
 - Manufacturing resource planning
 - Finance
 - Human resources



Software Challenges

Challenges

- Software development
- Software integration



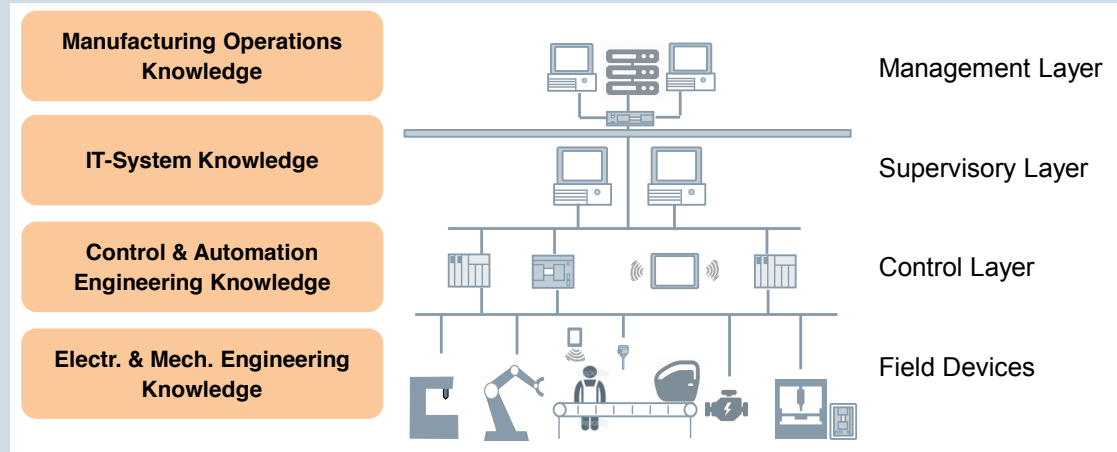
software development: ~40% of the price of manufacturing machines

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

Information Models for Smart Factories

Factory-wide info. models

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



Information Models for S

Factory-wide info. models

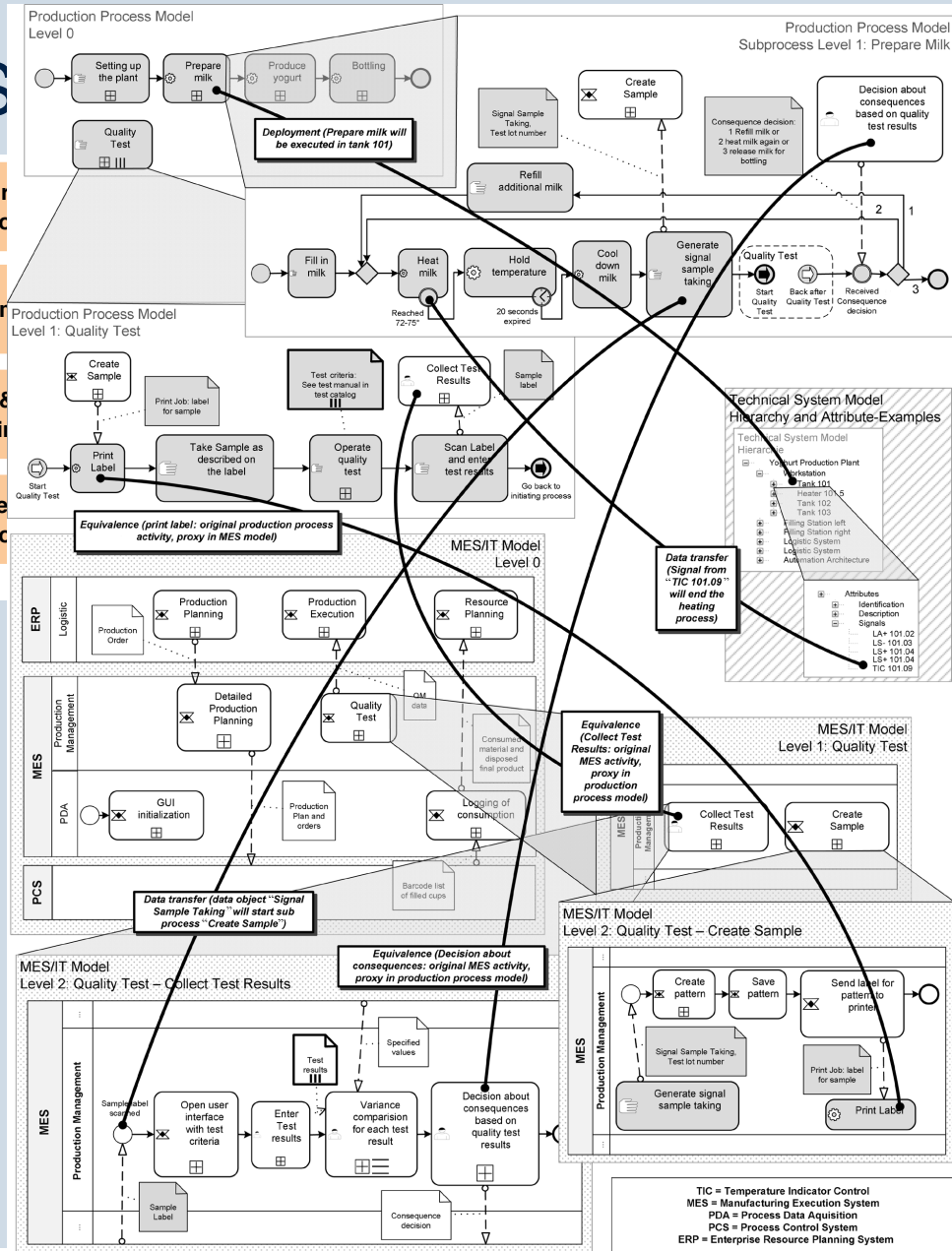
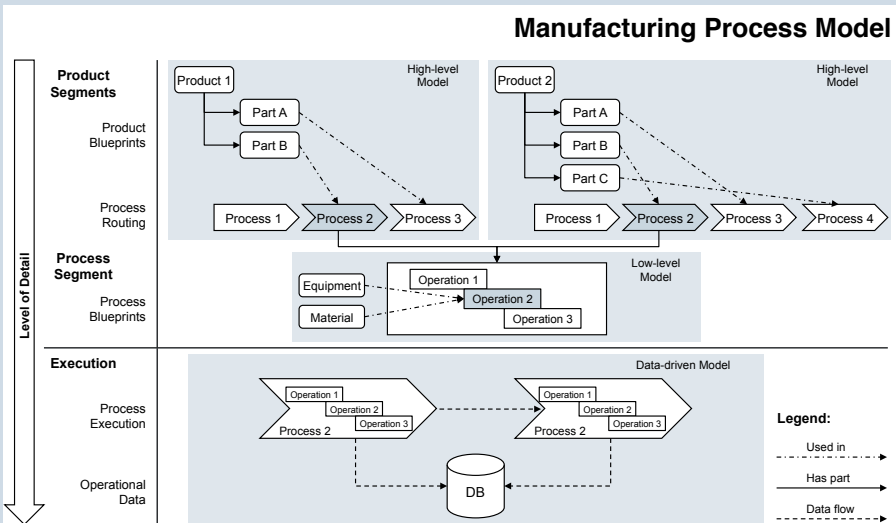
- address challenges
 - SW development
 - SW integration
- capture knowledge on all SW levels

Manufacturing Knowledge

IT-System

Control & Engineering

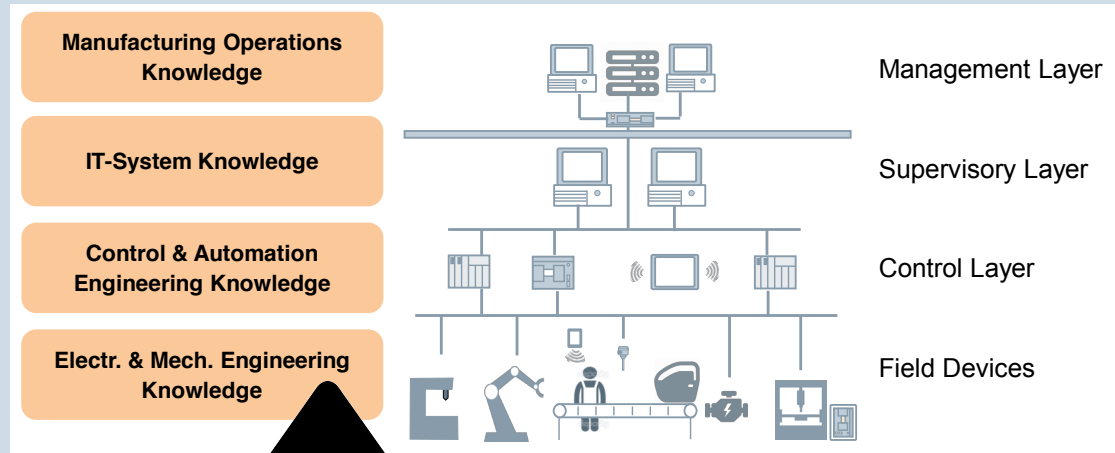
Electr. & Mechanical Knowledge



Information Models for Smart Factories

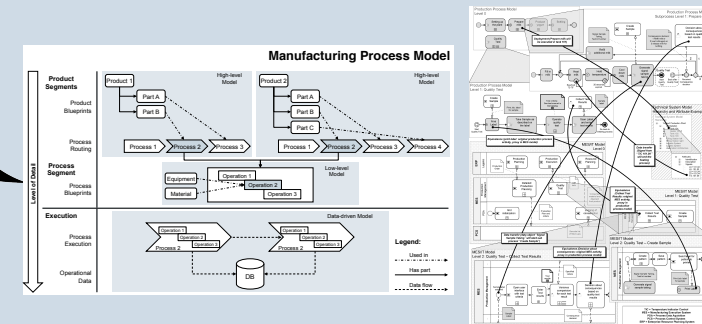
Factory-wide info. models

- 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



How well these models solve the problems?

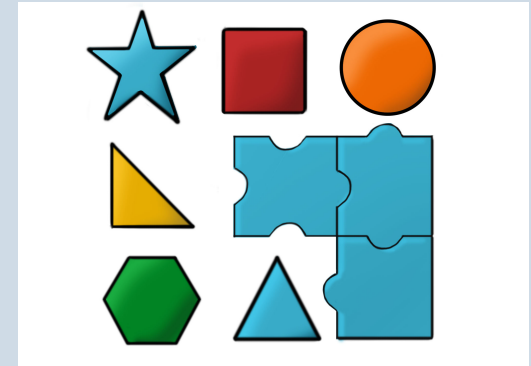


IEC connector type	Diagram of connector type
IEC 60320 C13 / C14	
IEC 60320 C19 / C20	

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?**



Ontologies as Information Models

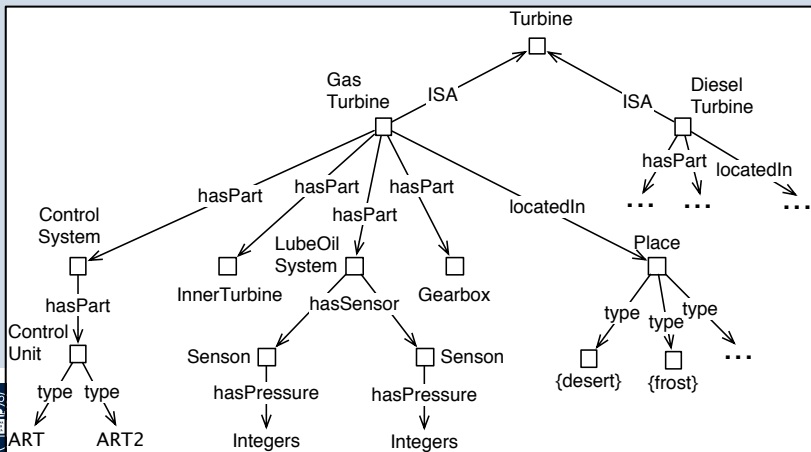


Industrial Adoption of Sem. Tech.

- 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.

Classification:	37-01-01-01 Gate valve [AAD643003]
Preferred name:	Gate valve
Definition:	-
Keywords:	Diaphragm slide valve Slide valve (flat plate) Slide valve (sleeve) Knife gate valve Bulk pusher Parallel flat slide gate valve Plate wedge gate Slide valve (elastic, flexible wedge gate) Sluice valve with round body Wedge-type flat slide valve Slide valve (wedge-type flat slide) Parallel slide gate valve Slide valve (parallel flat gate) Sluice valve Slide valve (rigid wedge gate) Sleeve slide valve
Properties:	<p>BAI371001 - Material number of the coating, interior</p> <p>BAI076001 - Classification system</p> <p>BAI039001 - Manufacturer drawing number</p> <p>BAH838001 - Reference norm material of the coating</p> <p>BAI035001 - Manufacturer code of the product information</p> <p>BAI059001 - Class description</p> <p>BAI269001 - Material description of the coating, external</p> <p>BAI187001 - Type description</p> <p>BAH940001 - Thickness of the coating, interior</p> <p>BAI400001 - Material number of the housing</p> <p>BAH938001 - Thickness of the coating, external surface</p> <p>BAH935001 - Code of the conformity evaluation</p> <p>BAI077001 - Conformity declaration present (Y/N)</p> <p>BAH926001 - width over all</p> <p>BAI385001 - Material number of the dynamic seal</p> <p>BAI041001 - Height over all</p> <p>BAH867001 - Reference norm material of the housing</p> <p>BAI369001 - Material number of the coating, external</p> <p>BAH853001 - Reference norm material of the dynamic seal</p> <p>BAI038001 - Manufacturer item list number</p> <p>BAI461001 - Material key of the dynamic seal</p> <p>BAI082001 - Length over all</p> <p>BAI037001 - Manufacturer country</p> <p>BAH638001 - Construction of the shaft end</p>

- 37 Industrial piping
 - 37-01 Metal or plastic fitting
 - 37-01-01 Gate valve
 - SSP 37-01-01-01 Gate valve S
 - SSP 37-01-01-07 Slide gate for waste
 - BSP 37-01-01-90 Gate valve (unclassified)
 - BSP 37-01-01-91 Gate (parts) S
 - 37-01-02 Globe valve S
 - 37-01-03 Butterfly valve
 - 37-01-04 Ball valve
 - 37-01-07 Condensate draining fitting
 - 37-01-08 Reflux fitting
 - 37-01-09 Safety fitting
 - 37-01-10 Appliance (incl. drive)
 - 37-01-11 Pressure reducer S
 - 37-01-12 Pigging systems
 - 37-01-13 Monitors
 - 37-01-14 Filter, Strainer
 - 37-01-15 Sampling valve
 - 37-01-18 Diaphragm valve
 - 37-01-19 Special valve
 - 37-01-20 Valve for suction system
 - 37-01-91 Fitting (parts)
 - 37-01-98 Valve (maintenance, service) S
 - 37-01-99 Valve (repair) S
 - 37-02 Piping (steel)
 - 37-03 Piping (NF metal)
 - 37-04 Pipeline (enameled)
 - 37-05 Pipeline (glass)
 - 37-06 Pipeline (thermoplastic)
 - 37-07 Piping (other metal)
 - 37-08 Pipeline (maintenance, repair)
 - 37-09 Air duct construction units
 - 37-10 Pipeline (distant heating, ready for installation)



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

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

Goals

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

Our Achievements

Ontology language for IIM

- expressiveness
- algorithms

Concrete ontologies

- 2 ontologies
- experiments

Modeling methodology and tooling

- SOMM systems

Goals

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



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

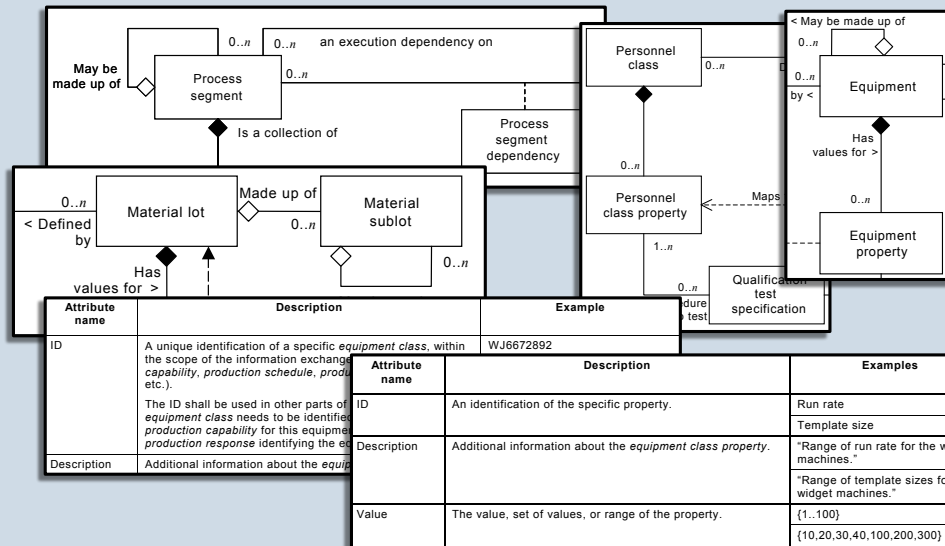
IEC 81346

A	Systems for common tasks
B	Systems of the main process (power plants)
C	System for storage of materials or goods
D	Systems for administrative or social purposes
E	Ancillary systems
F	Communication and information systems
G	Structure and areas for systems outside of the power plant process
H	Nuclear heat generation
J	Nuclear auxiliary systems
K	Water, steam, condensate systems
L	Systems for generation to and transmission of electrical energy
M	Medium supply system, energy
N	Cooling water systems
P	Auxiliary systems
Q	Flue gas exhaust systems
R	Control System

ISO/TS 16952-10

B	Electrical auxiliary power	MD	Wind Turbine System
C	Control and management	MDA	Rotor System
D	Functional allocation	MDK	Drive Train System
E	Fuel treatment and supply energy sources (inclusive)	MDL	Yaw System
F	Handling of nuclear equipment	MDV	Central Lubrication System
G	Water supply, disposal	MDX	Central Hydraulic System
H	Heat generation by combustion sources and heat generation from natural sources	MDY	Control System

ISA 88/95



Prefix	Designation task	Aspect	Application
F1	=	Function Designation	Main Systems, Systems, Subsystems, Basic Functions
=MDA	-	Product Designation	Product classes
=MDK	-	Product Designation	Product classes
=MDL	+	Point of Installation	Cabinets, vessels
=MDV	+	Point of Installation	Cabinets, vessels
=MDX	++	Site of Installation	Building, areas
RDS-PP		Control System	

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))
```

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($A \text{ AllValuesFrom}(P B)$)	$B(?y) \leftarrow P(?x, ?y) \wedge A(?x)$

OWL Axiom	Datalog rules
SubClassOf($A \text{ SomeValuesFrom}(R B)$)	$R.B(?x) \leftarrow R(?x, ?y) \wedge B(?y)$ and $Violation(?x, \alpha) \leftarrow A(?x) \wedge \text{not } R.B(?x)$
SubClassOf($A \text{ HasValue}(R b)$)	$Violation(?x, \alpha) \leftarrow A(?x) \wedge \text{not } R(?x, b)$
FunctionalProperty(R)	$R.2(?x) \leftarrow R(?x, ?y_1) \wedge R(?x, ?y_2) \wedge$ $\text{not owl:sameAs}(?y_1, ?y_2)$ and $Violation(?x, \alpha) \leftarrow R.2(?x)$
SubClassOf($A \text{ MaxCardinality}(n R B)$)	$R.(n+1).B(?x) \leftarrow \bigwedge_{1 \leq i \leq n+1} (R(?x, ?y_i) \wedge B(?y_i))$ $\bigwedge_{1 \leq i < j \leq n+1} (\text{not owl:sameAs}(?y_i, ?y_j))$ and $Violation(?x, \alpha) \leftarrow A(?x) \wedge R.(n+1).B(?x)$
SubClassOf($A \text{ MinCardinality}(n R B)$)	$R.n.B(?x) \leftarrow \bigwedge_{1 \leq i \leq n} (R(?x, ?y_i) \wedge B(?y_i))$ $\bigwedge_{1 \leq i < j \leq n} (\text{not owl:sameAs}(?y_i, ?y_j))$ and $Violation(?x, \alpha) \leftarrow A(?x) \wedge \text{not } R.n.B(?x)$

Our Achievements

Ontology language for IIM

- formalization
- algorithms

Concrete ontologies

- 2 ontologies
- experiments

Modeling methodology and tooling

- SOMM systems

Goals

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

Ontologies

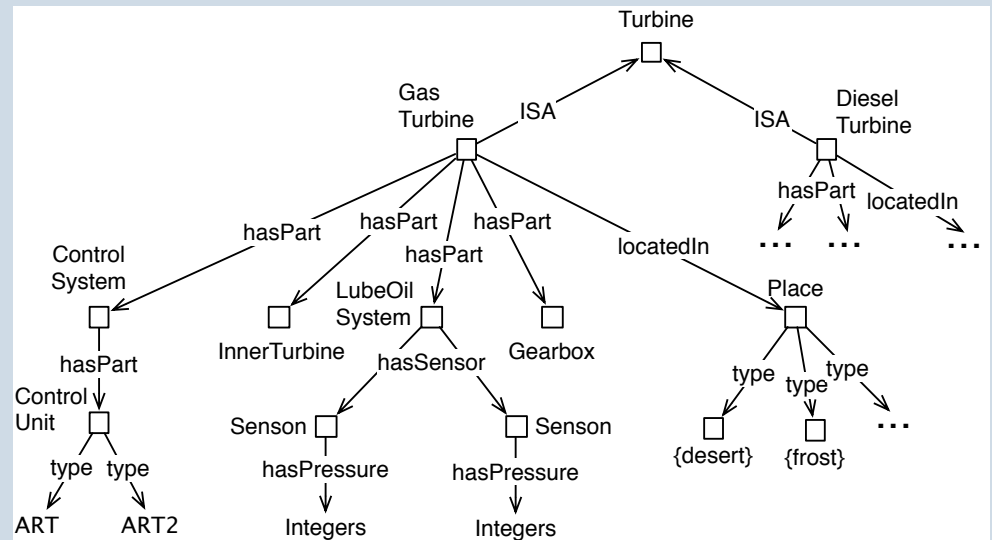
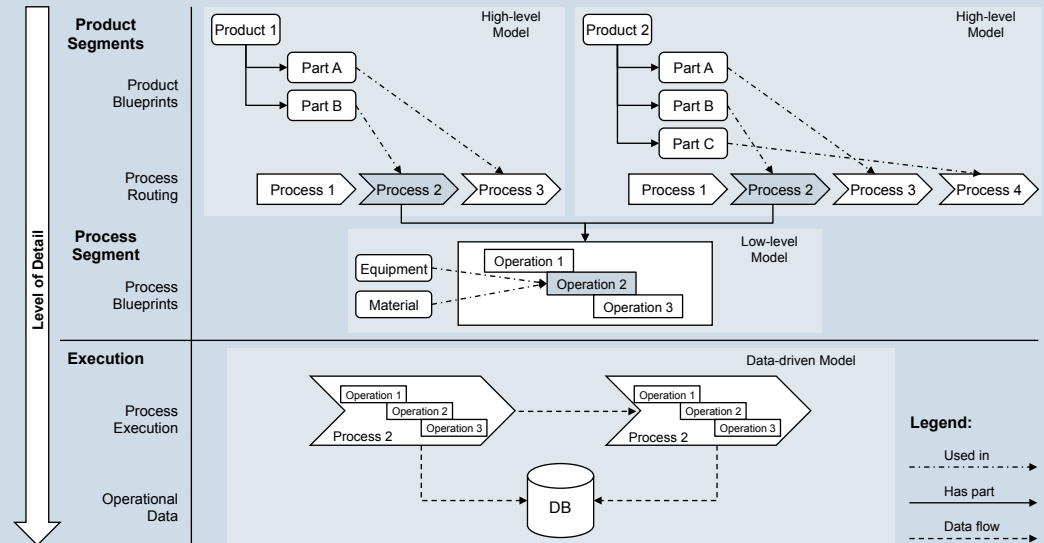
Manufacturing ontology

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

Turbine ontology

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

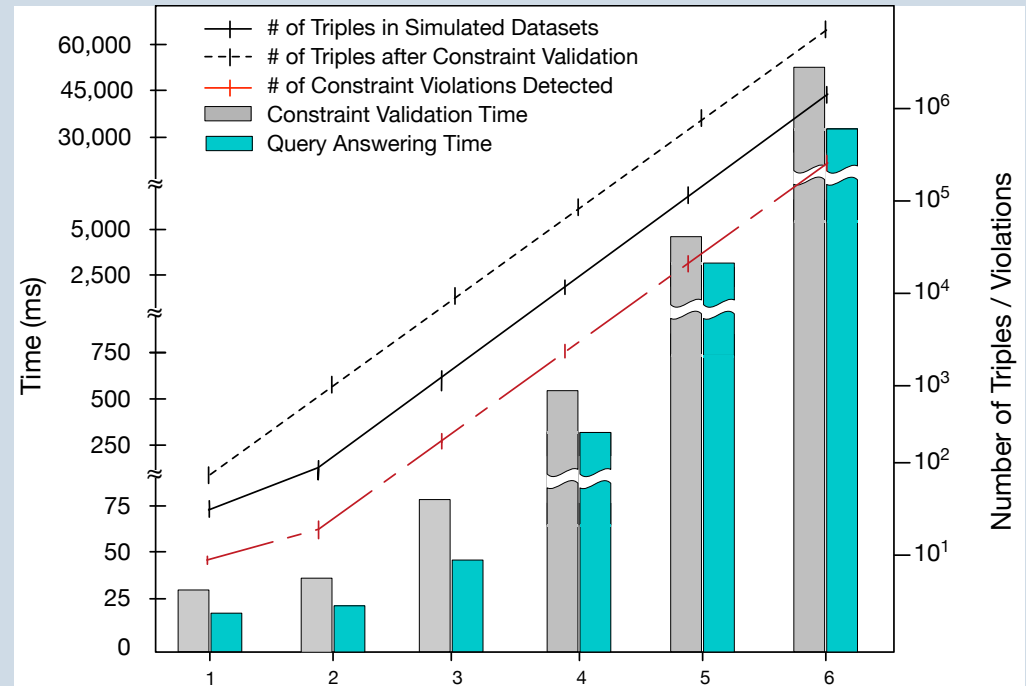
Manufacturing Process Model



Manufacturing Experiment

Manufacturing data

- simulated
- 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 → 1×10^6



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

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

Our Achievements

Ontology language for IIM

- formalization
- algorithms

Concrete ontologies

- 2 ontologies
- Experiments

Modeling methodology and tooling

- SOMM systems

Goals

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

SOMM: Siemens-Oxford Model Manager

Main Features

- WebProtege with new widgets
 - form-based widgets
- Axioms & constraints (A&C)
 - A&C on the same interface, encoded in Datalog
- Auto-generated data forms
 - from prop. assigned to classes
- Extended hierarchies
 - “partonomies” w/ any properties
- Ontology importing
- Reasoning
 - Hermit for ontology reasoning
- Constraint validation and Query answering
 - Datalog engine with stratified negation (IRIS)

Property (*)	Required?	Min.	Max.	Range (*)	Value
hasState	<input checked="" type="checkbox"/>	1	max	State	Enter individual
hasId	<input checked="" type="checkbox"/>	1	max	xsd:string	Enter literal value
hasConfig	<input checked="" type="checkbox"/>	1	3	SteamTurbineConfig	Enter individual
hasProductLine	<input type="checkbox"/>	min	max	ProductLine	Enter individual

SOMM Data Insertion - Details for 'steam_turbine_987'

hasState (*) Select a value

hasId (*) turbine_987

hasConfig (*) SteamTurbineConfigura

hasProductLine Select a value

SOMM Instance Hierarchy for Property 'follows'

- follows
 - Root
 - packaging123
 - testing123
 - conveying123
 - loading123

Summary

Use case analyses

- Smart factories and the role of info models
- Industrial standards: Manufacturing, Energy

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

Goals

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

