ForMetis
ENTERPRISE ENGINEERS

Enterprise engineering based IT systems for organizations

Realizing
Business - IT alignment
via
Modeling, instead of Programming,

Dr. ir. Steven J H van Kervel.
ForMetis BV, Netherlands.

Agenda

1. Who are we? Principles and goals
2. State of the art in enterprise IT systems
3. Enterprise Engineering - overview
4. Enterprise IT systems engineering
5. The enterprise operating system
6. A business case in full production
7. Future research topics
8. Discussion and acknowledgements

1 WHO are we? Principles and goals.

Who are we?

- Enterprise Engineers
- Adaptive case management systems since 1998
- Customers: Financial services, banking, and misc
- Focus op R&D - methodologies - software tools
- Bridging academic world & professional world

a small enterprise, dreaming big, customer is king

Our principles and goals

- Human-oriented methods (DEMO)
  human autonomy, truthfulness, competence, responsibility, accountability, authority, are all key notions

- Business is leading (not IT)

- Address the business - IT alignment challenge

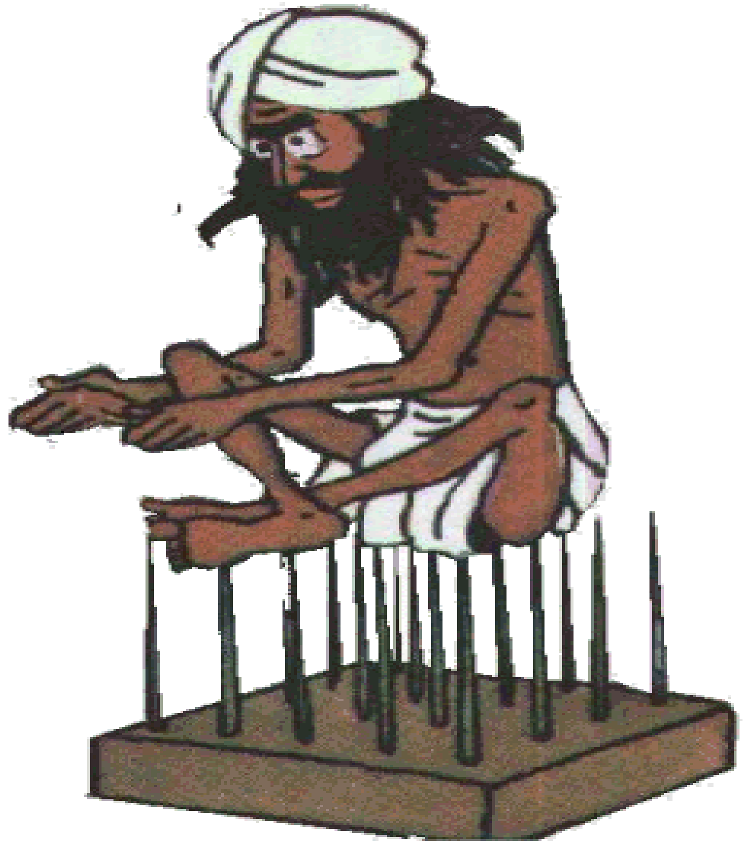
- Systematic engineering methods application
  High quality requirements and specifications;
  Modeling, verification, validation, model simulation;

- Support the agile enterprise → evolving models & software

'State of the art' methods are not good enough
Enterprise Engineering - DEMO is a promising young engineering discipline

2 State of the art in enterprise IT systems

Problem domain: 1- IT systems for enterprises



Life is full of suffering:

1 Business-IT alignment fails

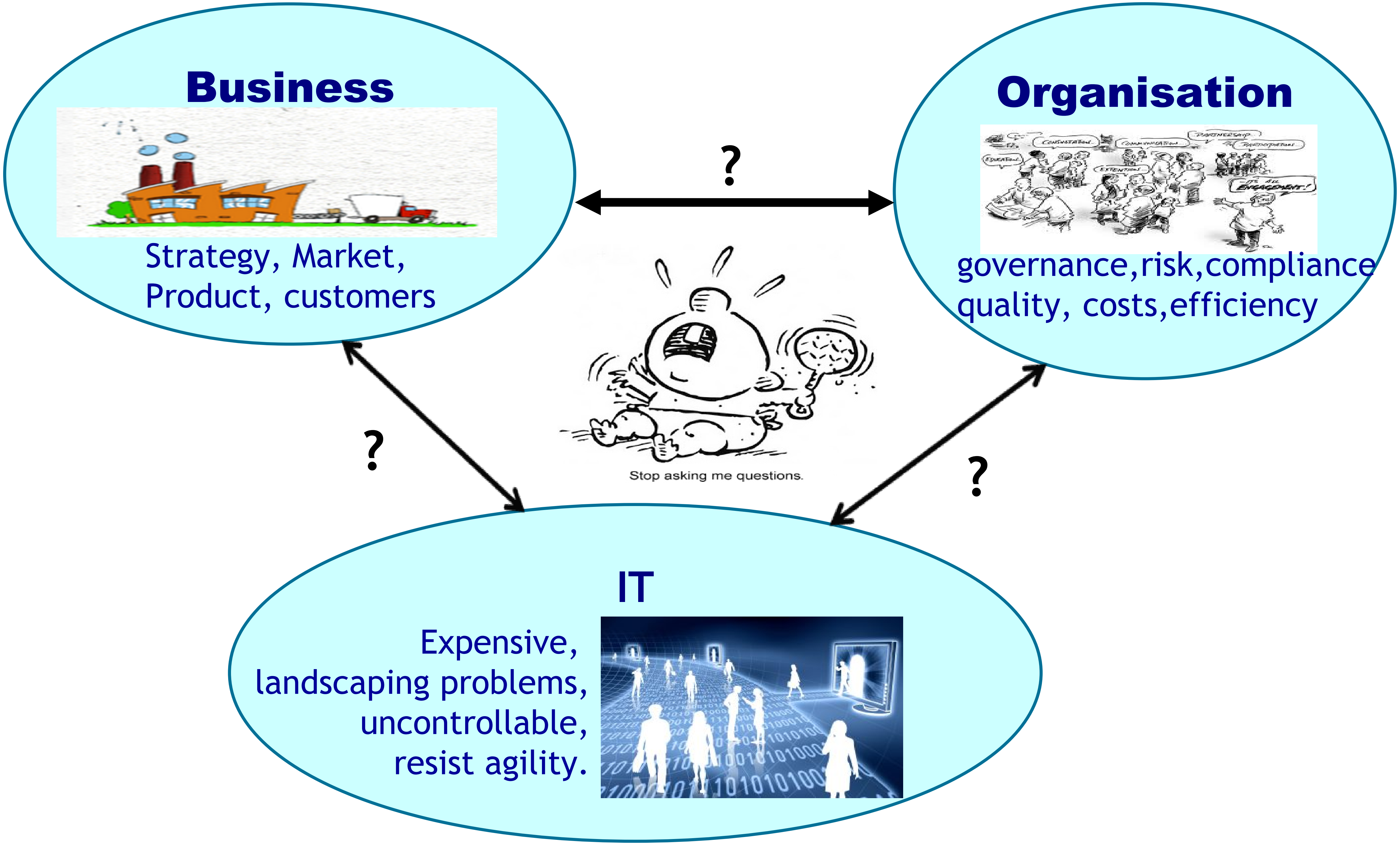
2 IT system engineering is a #@$$

3 IT systems resist evolution

IT state of the art fails: Zachman, BPMN, BPEL, OPC, TOGAF, Flow-charts
Or do software engineers fail?
Symptoms or causes or deeper causes?

<u>Questions:</u>
How to construct *better* information systems? -function
How to construct information systems in *a better way*? - construction

1- Business – IT challenge – a *functional* mismatch



Business

Strategy, Market,
Product, customers

Organisation

governance,risk,compliance
quality, costs,efficiency

IT

Expensive,
landscaping problems,
uncontrollable,
resist agility.

Unmanageable complexity, unclear domains, concepts and relations, Bad methods

Problem 1: Business IT alignment fails too often

Observations:
IT systems functionality does not meet expectations
Mismatch expectations & user functional specs
After acceptance tests there are major rebuild of IT systems
Causes further deterioration of IT system structure and complexity [NS]
Lack of truthfulness, appropriateness, comprehensiveness, consistency, coherence & conciseness (C4-ness) qualities of IT system specs

Some bad practises:
Functional requirements are not subjected to early validation
Validation is done at acceptance tests – far too late
Validation against user functional specs or user expectations?

We must better (C4-ness) understand Reality,
understand appropriateness and truthfulness qualities of specifications

Problem 2: IT system engineering fails too often

Observations:

1- Standish Group: 32% IT projects success, remainder fails
> 10 M $ project budget guarantee project failure

2- Budzier A, Oxford University & Tata Consultancy:
similar figures cijfers.

Successes of IT:

GPS systems, GSM systems, Medical imaging, Internet + much more

Conclusions:

If programmers gets high quality specs, they succeed.

For enterprise IT systems, programmers get no high quality specs

Task: To devise high quality specs of enterprises.

Getting high quality specifications ~ well understanding a domain
Programmers are NOT guilty

Problem 3: IT systems resist evolution

Observations:
Costs of enterprise IT system maintenance increases exponentially over time.
At some point in time further maintenance - adaptations - bug fixing becomes unmanageable in complexity and resources.
In many IT projects this point is achieved before delivery.

Causes are explosion of combinatorial effects [Normalized Systems theory]

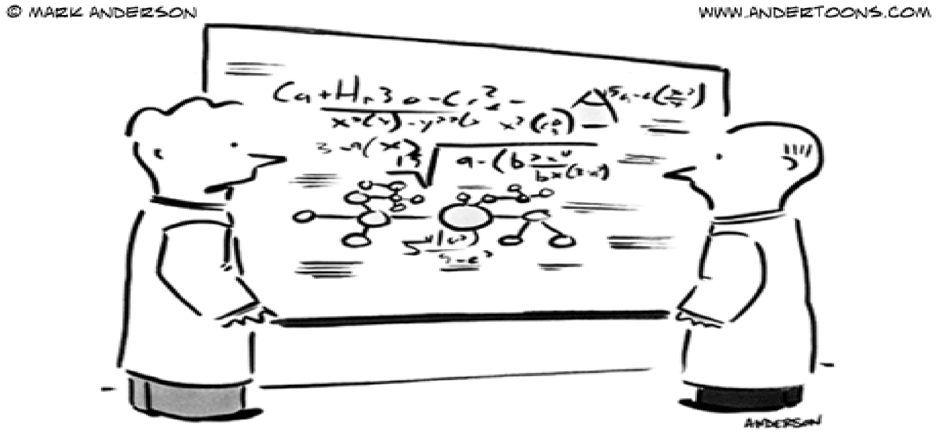
Two complementary approaches:
1- Apply NS compliant software development environments

2- Apply model (instance) driven software architectures
New requirements → new model → new IT system
(state of the art MDA - model transformations - considered not good enough for our domain)

Problem 4 State of the art modeling methods fail

Best practice methods:
"we don't understand why this works well, but we try to repeat this, while we don't know how".

Unsubstantiated promises followed by catastrophes;
Nobody questions the usual "management & sales BS"



"I don't know, it's a little formulaic."

Lack of scientific foundations,
Lack validation, verification,
Lack of formal rigor

Better than nothing - not good enough

3 Enterprise engineering - overview

Principles of Enterprise Engineering

Enterprises are social systems (general systems theory)

Founded on scientific theories (Φ, τ, Ψ); enterprise ontology

Systems engineering: Function – Construction; Black-White Box etc

DEMO methodology to construct DEMO models of enterprises

Models are high quality specifications, propositions in a formal ontological language, support model validation and verification.

Finite state automaton – DEMO engine – executes model propositions expressed in 4 DEMO aspect model languages.

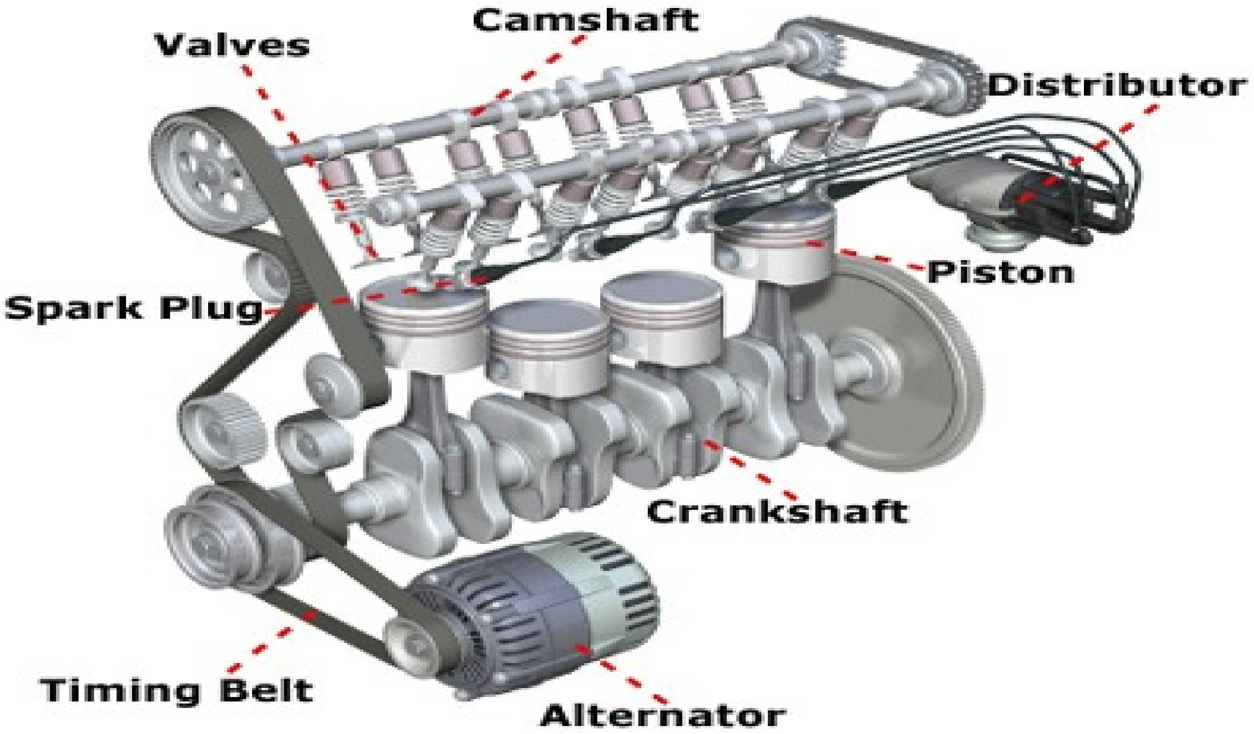
It's Physics (non-relativistic & non-quantum) for organizations.

Engineering -1-

Function and Construction

Two different perspectives on any engineering artifact

www.shutterstock.com · 19661566



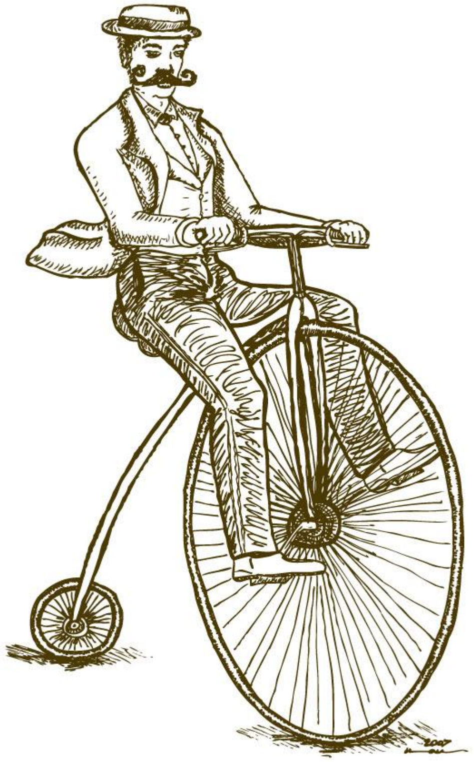
Function perspective
subjective,
"in the eye of the beholder"

Construction perspective,
objective,
"construction and operation"

Engineering -2-

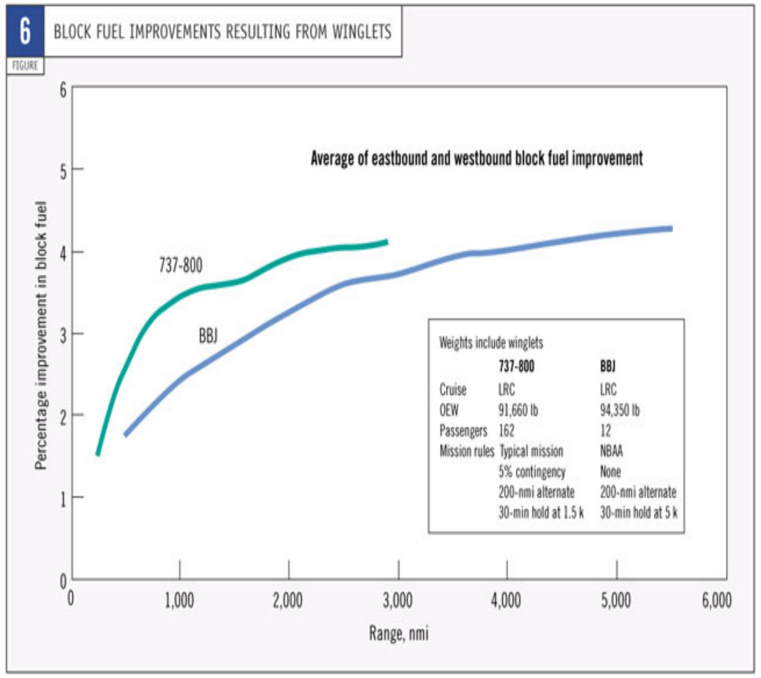
Function and Construction - engineering cycle

Incremental improvement of construction → incremental improvement of function



Engineering -3-

Example: Boeing improves efficiency by 5%.


6 FIGURE BLOCK FUEL IMPROVEMENTS RESULTING FROM WINGLETS

Engineering sciences, design science research, systems theory

Formal methods, languages, models, finite elements method

Empirical sciences: physics, gas laws, Navier-Stokes equations

Validation in wind tunnels

Engineering -4-

Example: Models are high quality specifications

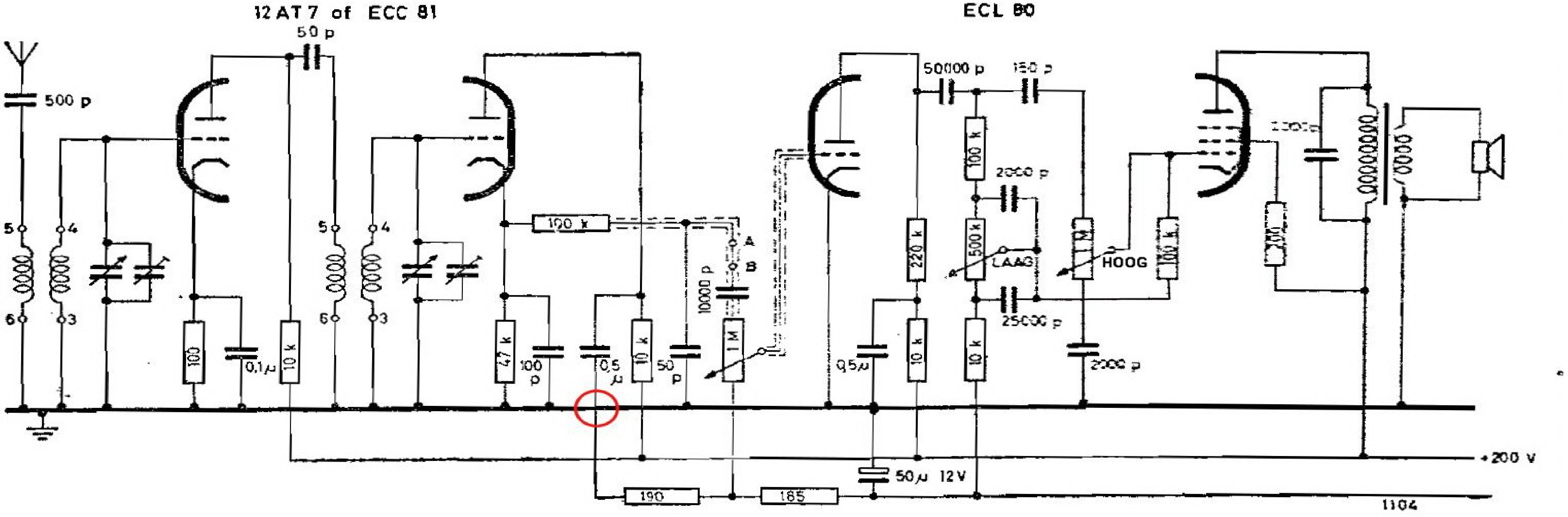


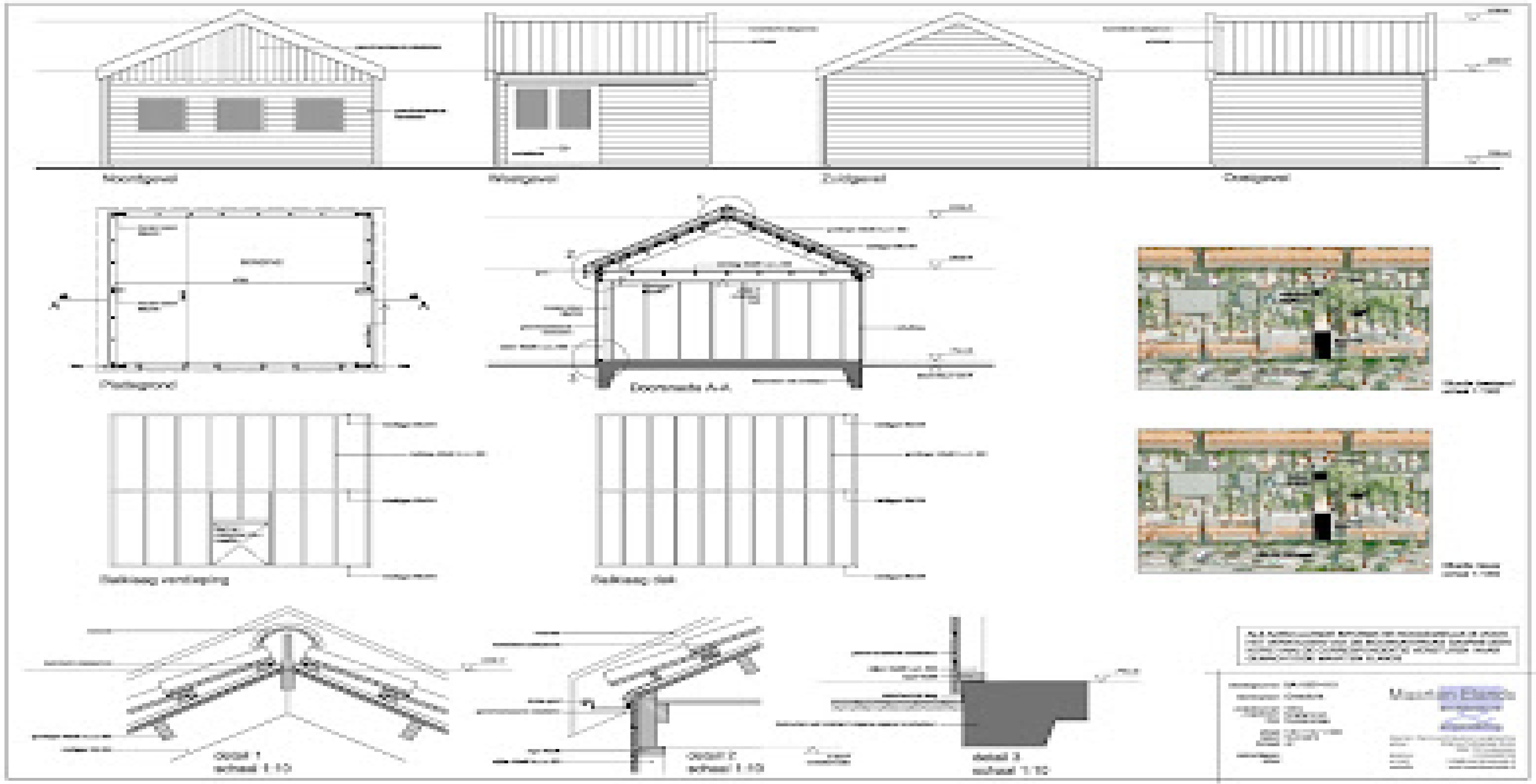
fig. 9

Any electronics amateur builds a working radio using this scheme

We business consultants, business architects, fail to deliver high quality specifications of enterprises and IT systems.

Engineering -5-

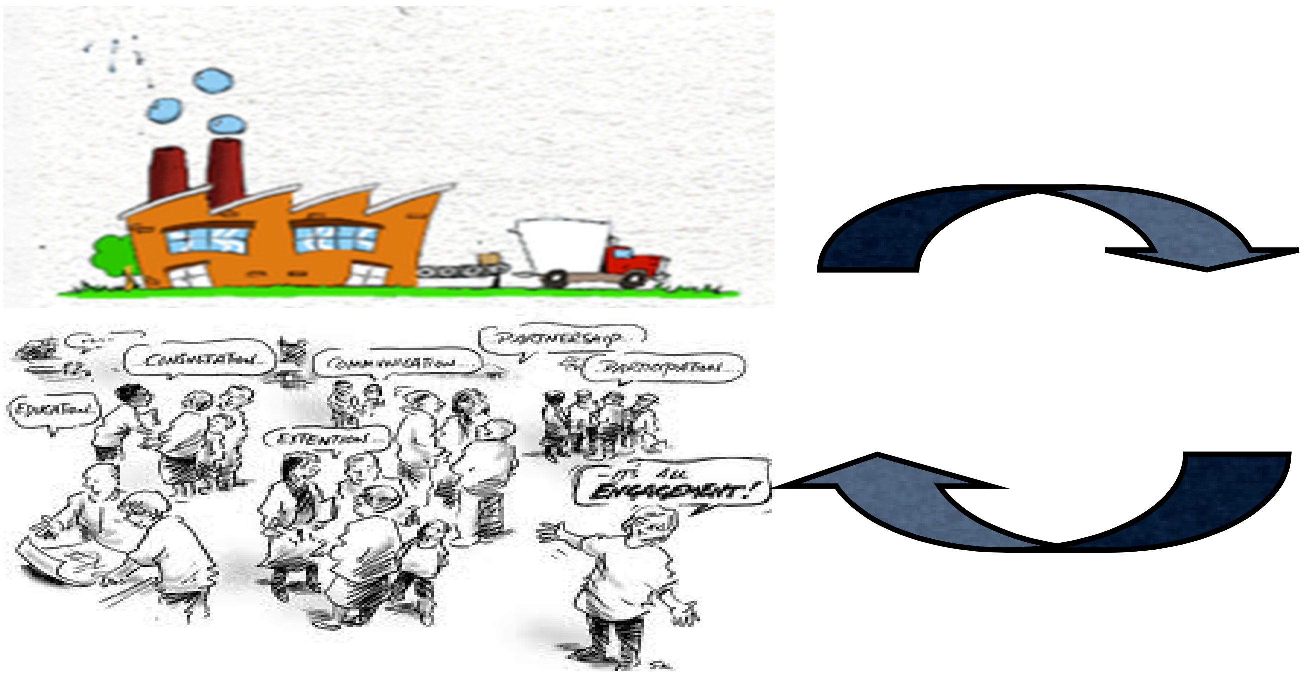
Example: models are specifications with multiple aspects

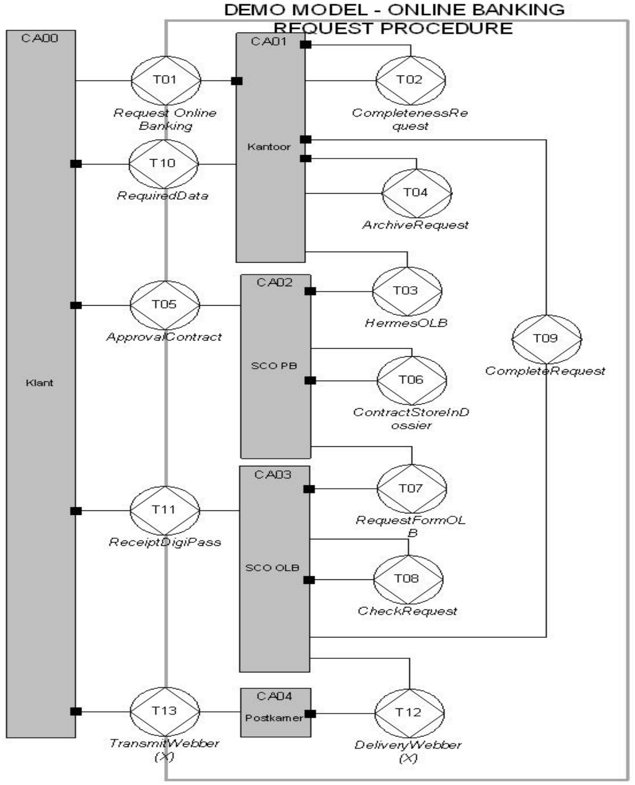


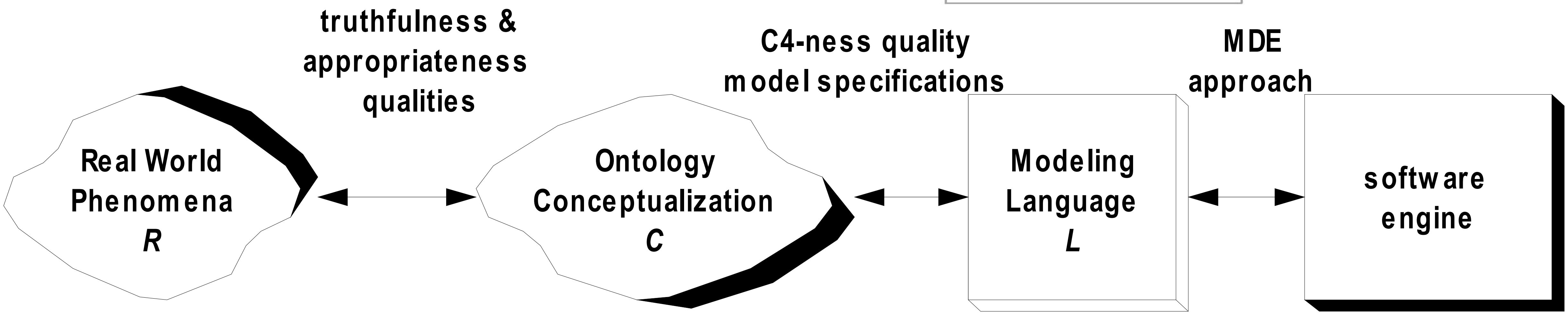
Any contractor builds this barn complete and correct with this model.

We business architects, consultants, analysts etc. fail to implement.

Ontology: a shared formal representation of Reality







truthfulness & appropriateness qualities

C4-ness quality model specifications

MDE approach

Real World Phenomena
R

Ontology Conceptualization
C

Modeling Language
L

software engine

Enterprises are social systems

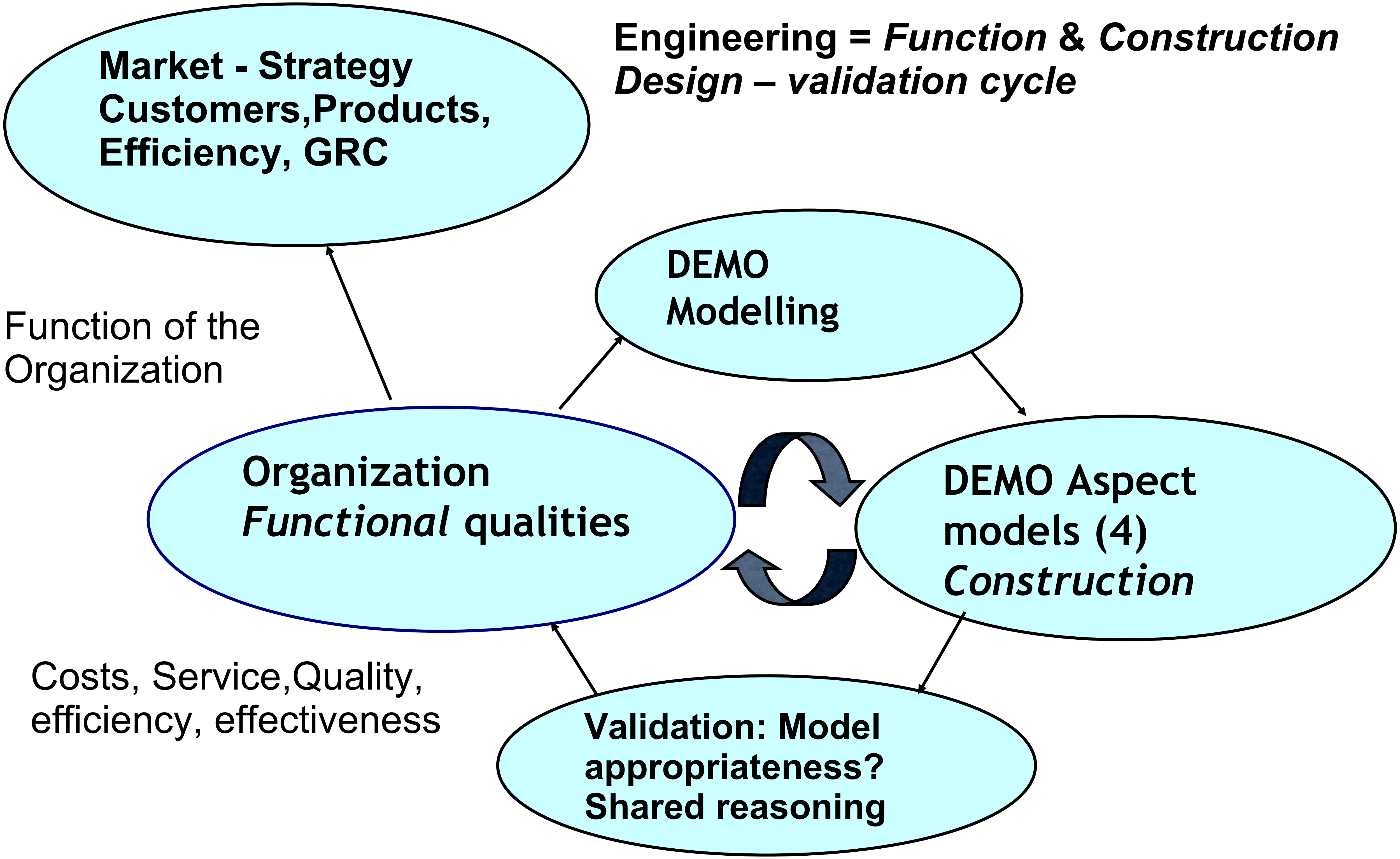
Enterprise Ontology: 4 axioms

4 DEMO Aspect models

DEMO engine executes (MDA) DEMO models

DEMO Modelling

Engineering = *Function & Construction*
Design – validation cycle



Market - Strategy
Customers,Products,
Efficiency, GRC

Function of the
Organization

DEMO
Modelling

Organization
***Functional* qualities**

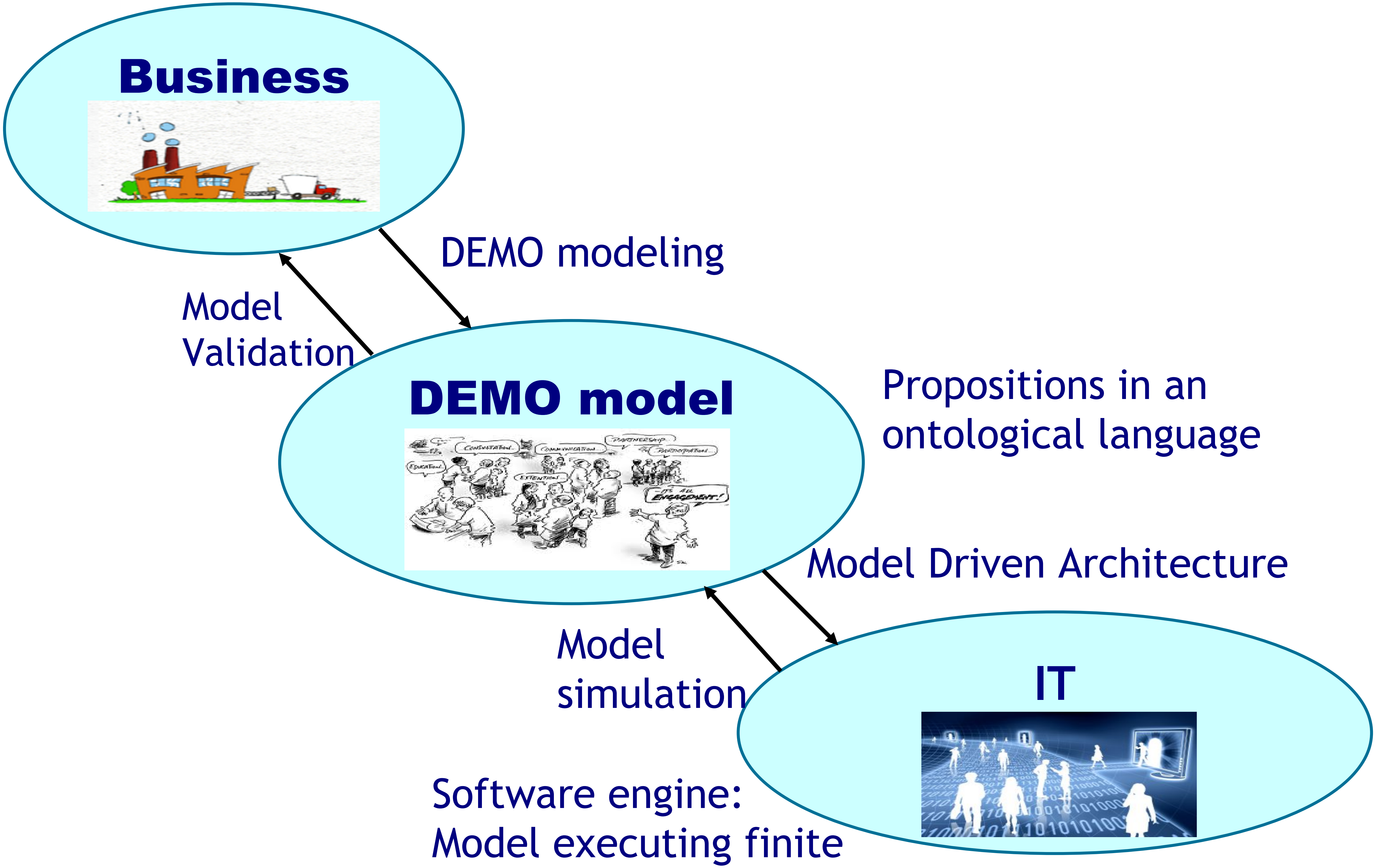
DEMO Aspect
models (4)
Construction

Costs, Service,Quality,
efficiency, effectiveness

Validation: Model
appropriateness?
Shared reasoning

4 Enterprise IT systems engineering

Enterprise Engineering – DEMO modeling



Business

DEMO modeling

Model Validation

DEMO model

Propositions in an ontological language

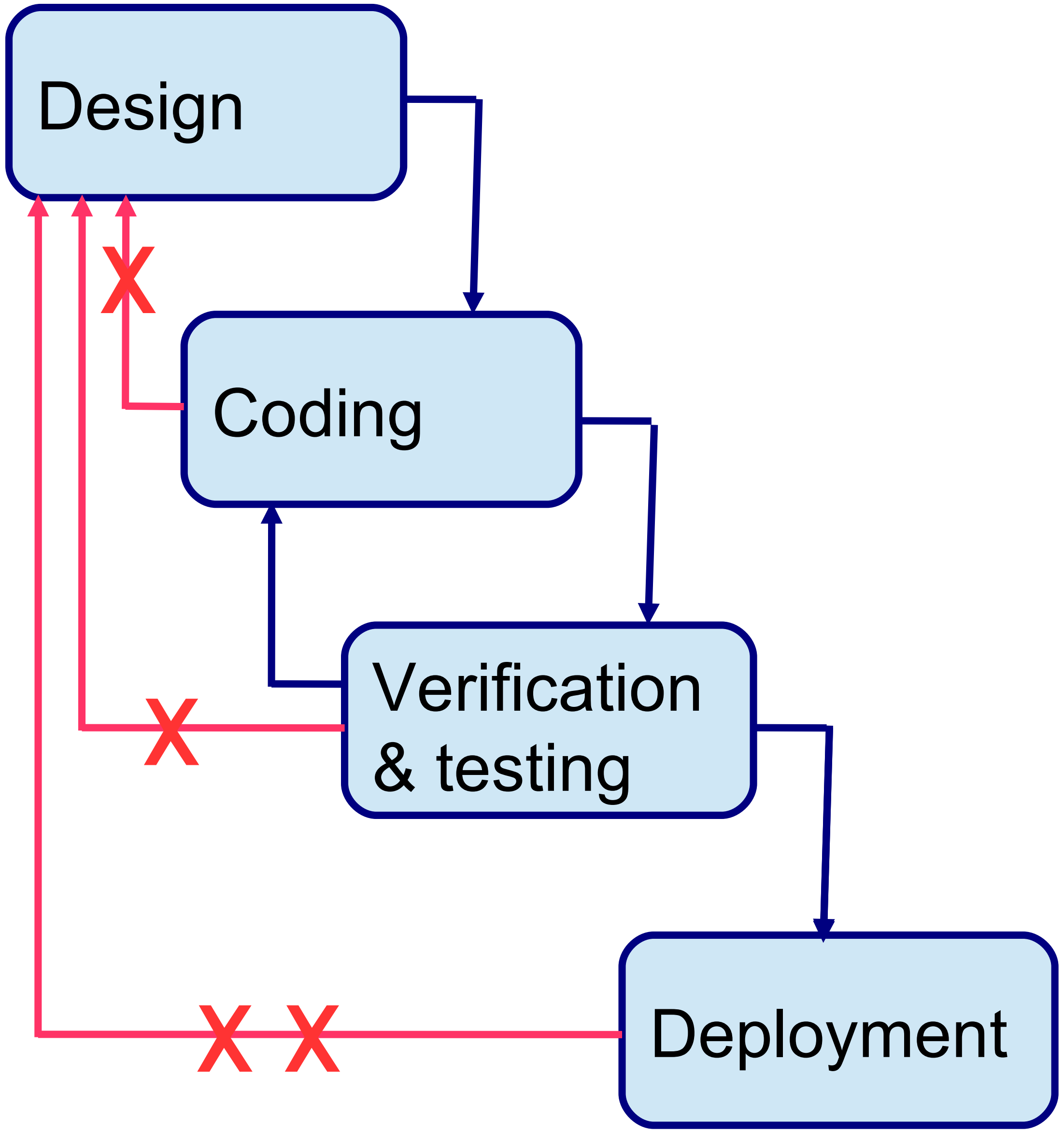
Model Driven Architecture

Model simulation

IT

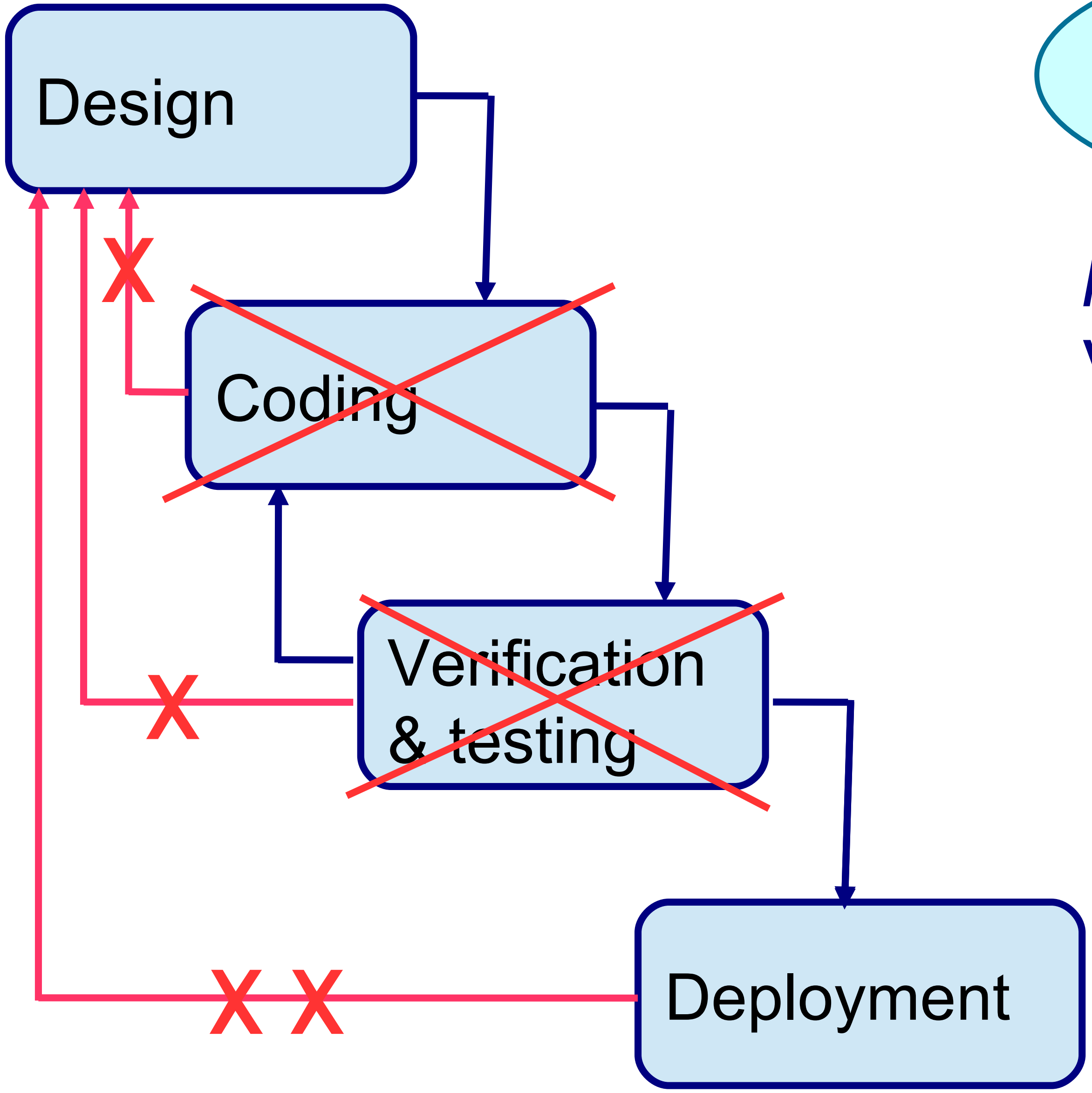
Software engine:
Model executing finite state automata

Enterprise Engineering - Methods



Traditional waterfall method - root of much evil

Enterprise Engineering – Model driven IT

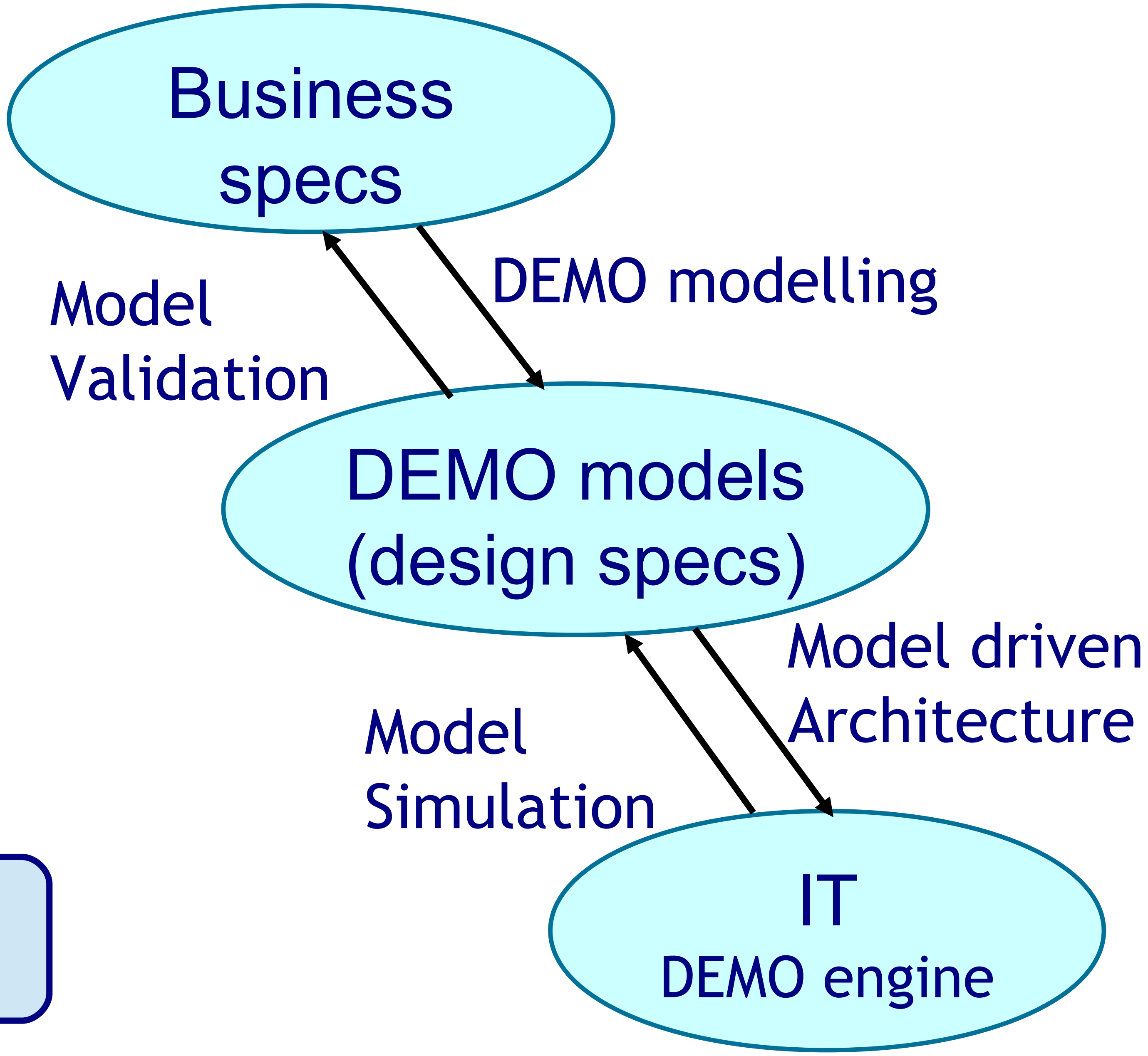


Design

~~Coding~~

~~Verification & testing~~

Deployment



Business specs

DEMO modelling

Model Validation

DEMO models (design specs)

Model driven Architecture

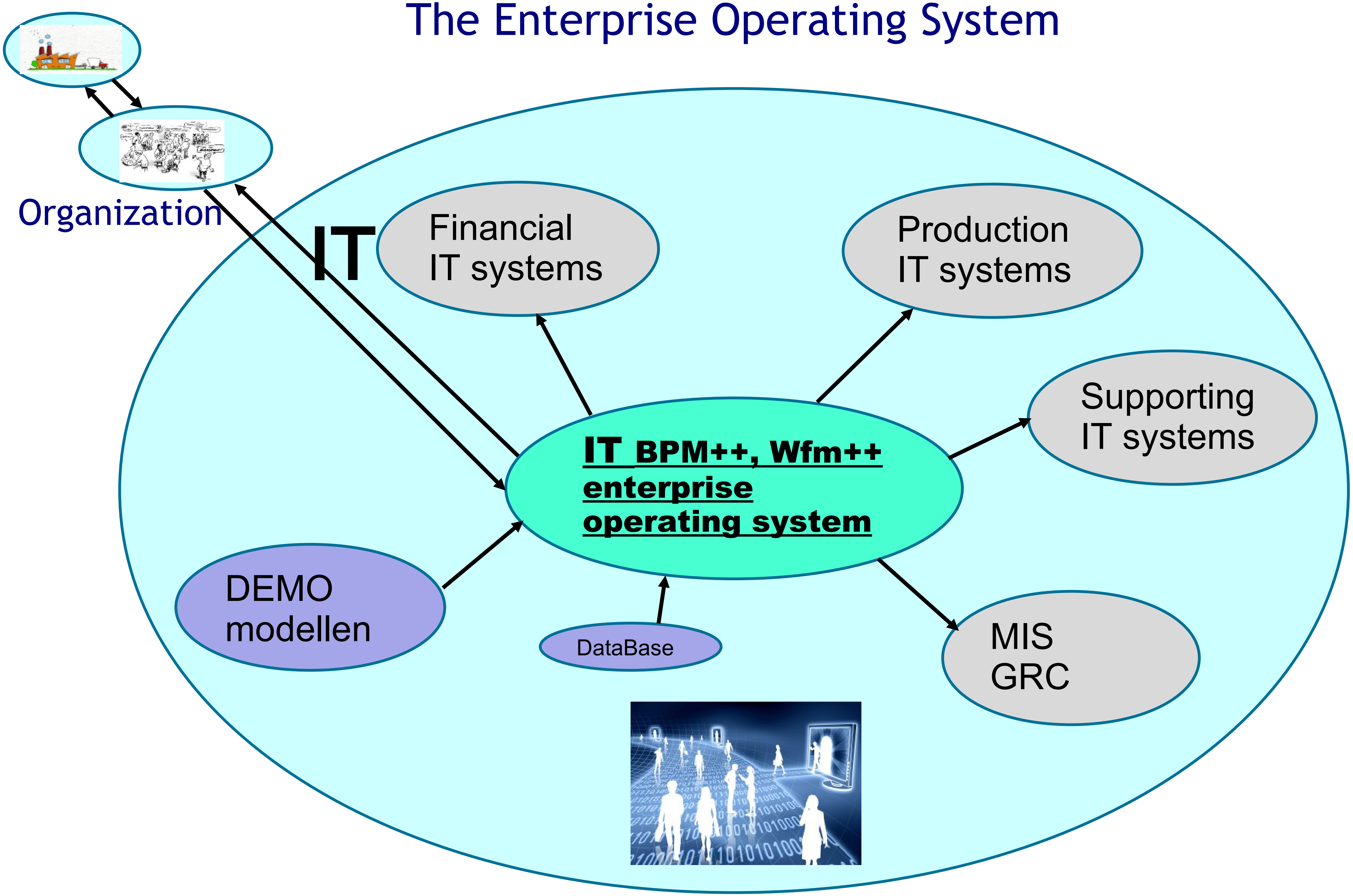
Model Simulation

IT

DEMO engine

5 The enterprise operating system

The Enterprise Operating System



Organization

IT

Financial IT systems

Production IT systems

Supporting IT systems

IT BPM++, Wfm++ enterprise operating system

DEMO modellen

DataBase

MIS GRC

6 A business case in full production

Enterprise Engineering - Practice

DEMO applied to:

1. Organizational (re)design

2. IT & Software Engineering (BPM++, Workflow++)
   High quality specifications

3. ITIL and COBIT

4. Governance, Risk & Compliance (GRC)

First professional DEMO engine project in 2012

Application domain: Adaptive case management systems.

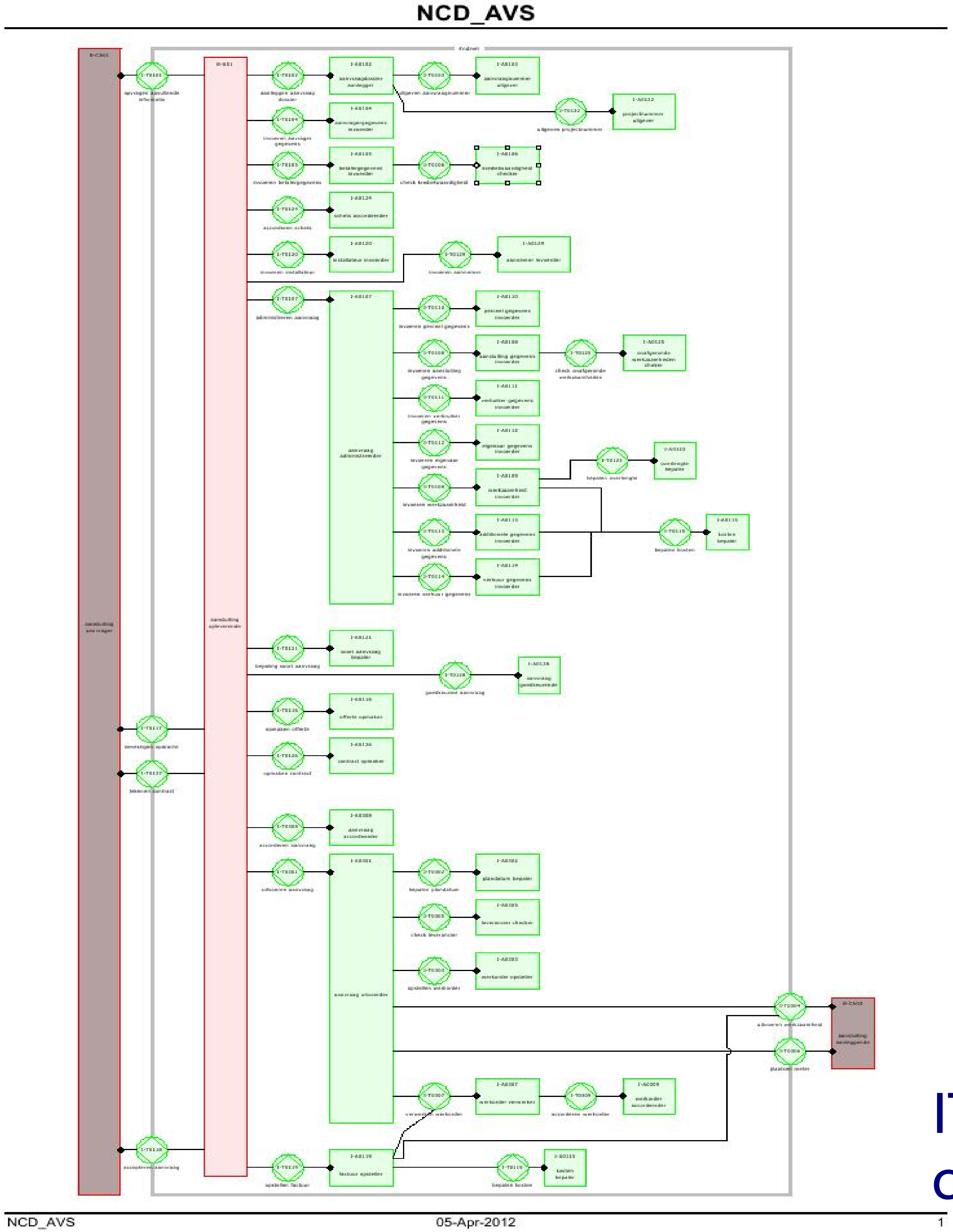
Complex tailor-made service delivered to a demanding customer where the customer is co-producer.

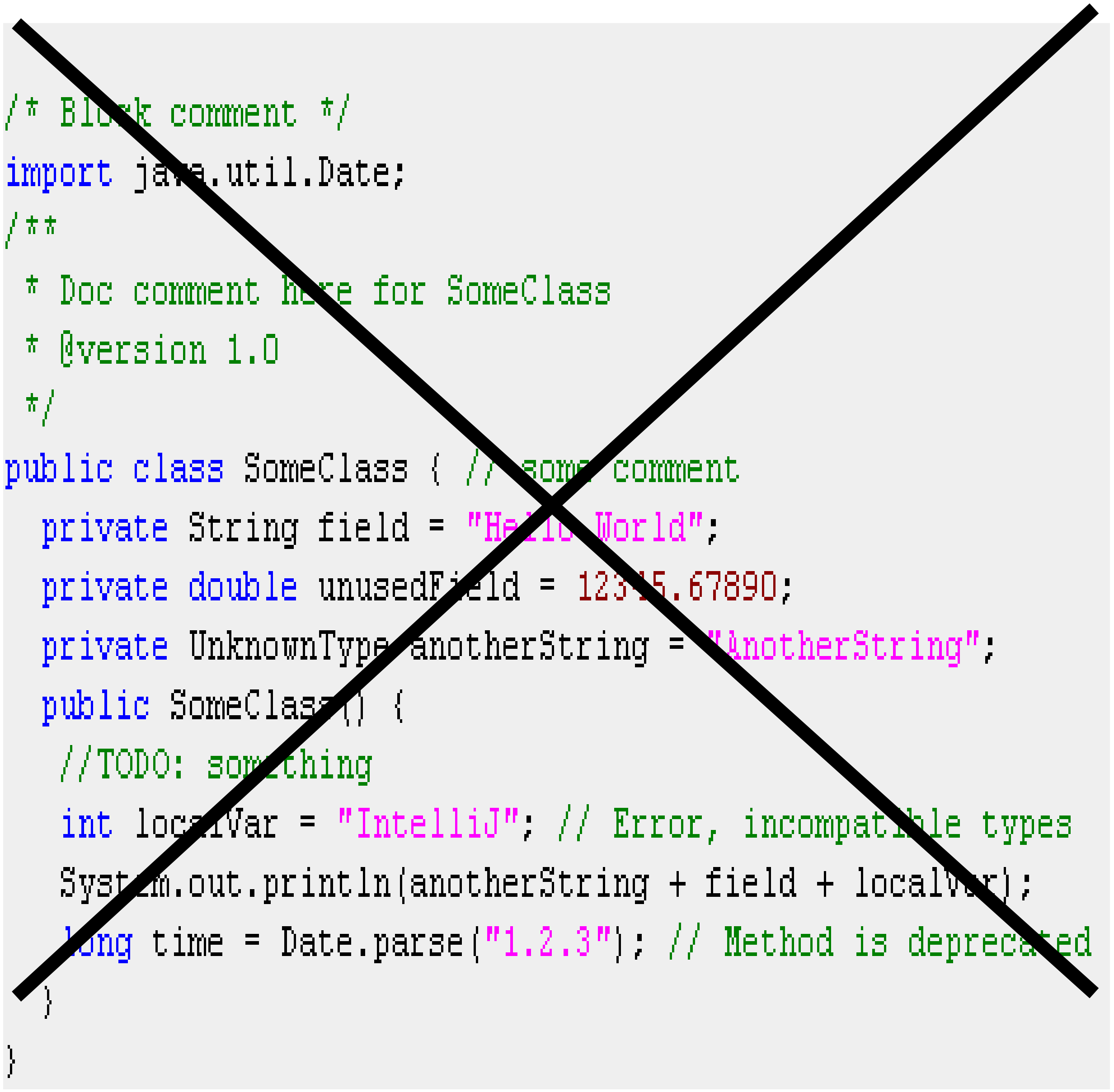
Ex: Government services, financial services, litigation & legal services, document-based services.

Production of a contract for energy services, subjected to business procedures, legal compliance, approval, document-oriented, produced interactively with the customer.

- Interfaces to legacy systems incl. external ERP system.
- Integrated with a (MDE) document production system (P-system).
- Dossier and document scanning, routing.
- Compliance to business procedures and avoiding anomalies, deadlocks, incomplete transactions and deadline control are achieved.

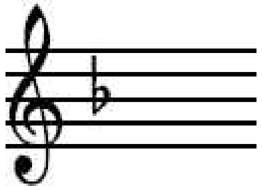
DEMO model based IT system in production





```
/* Block comment */
import java.util.Date;
/**
 * Doc comment here for SomeClass
 * @version 1.0
 */
public class SomeClass { // some comment
  private String field = "Hello World";
  private double unusedField = 12345.67890;
  private UnknownType anotherString = "AnotherString";
  public SomeClass() {
   //TODO: something
   int localVar = "IntelliJ"; // Error, incompatible types
   System.out.println(anotherString + field + localVar);
   long time = Date.parse("1.2.3"); // Method is deprecated
  }
}
```

IT system in production (2012)
ca 80% reduction of programming (# lines)



Some FUTURE Research Topics

Empirical evidence to support the appropriateness claims of EE, addressing the problems: Professional business cases, solving real life problems & devising metrics to assess case studies.

Devising theory and engineering methods to address the Governance, Risk and Compliance (GRC), ITIL, COBIT problems in (financial) services industry, based on enterprise ontology and DEMO.

Social engineering methods, knowledge engineering, shared reasoning and validation methods, organizational transformations and other soft skills.

Methods and tools for the systematic design and implementation of supporting adaptive case management systems (ACMS) that operate directly on the enterprise operating system.

Design and construction of enterprise IT systems operating on the enterprise operating system; e.g. financial & accounting IT systems, production systems, personnel management etc.

Conjecture: these systems can be constructed using a model driven engineering approach, based on ontologies, mapping of concepts, conceptual languages and model executing software engines.

A radical new approach for ERP systems, based on the enterprise operating system and ACMS.

Design and construction of BRE (Business Rules Engines), using a model driven engineering approach, operating on the enterprise operating system.

AMBITIOUS Future of enterprise IT systems

* Enterprise IT systems become a commodity

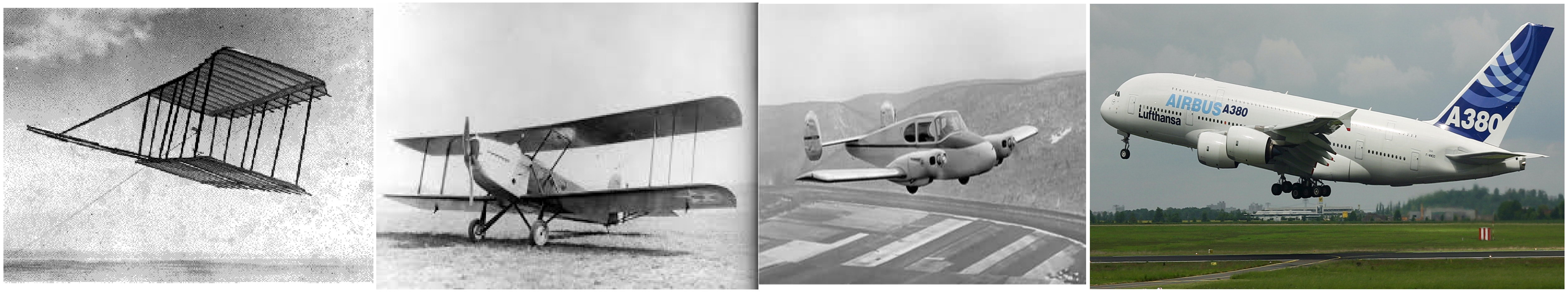
* Mass produced by an automated software factory,
using model driven architectures, conceptual languages, engines.

* The real challenge is modeling of enterprises,
In such a way that:
Governance, risk & compliance criteria are met;
Efficiency and effectiveness are optimized.

* Human beings work in a human way with others,
carry freedom, responsibility, competence, authority,
and develop their own enterprise models,
In order to support their customer in the best way.

Abandoning not human oriented Taylorian and Weberian organizations.

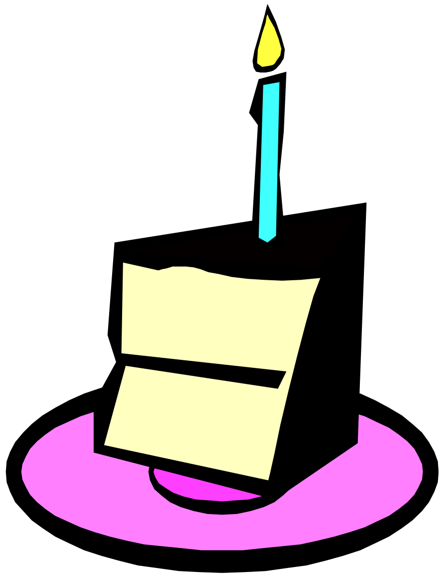
Motivation for research on enterprise engineering



* We are here

- There are real problems in society that need solutions.
- Problems are related to understanding humans and enterprises on one side, and engineering artifacts on the other side.
- Enterprise engineering is a solid engineering discipline - but very young and immature - with much to do and to achieve.
- There are some formidable theoretical challenges to solve.
- Students get an engineering education focused on solid engineering, formal methods etc, but also on human and social engineering topics.
- Benefits for Society, Students, University and Science.

Discussion & Acknowledgements



Ciao! Community, DEMO professionals, students.

We shall support them with their professional work.