simufact

Simulating Manufacturing

MSC Software Company

Best practice in information exchange in process scale modelling

2nd International Workshop on Software Solutions for ICME

Barcelona, 12th – 15th April 2016

Dr.- Ing. Ralph Bernhardt
Simufact Engineering GmbH

simufact
Simulating Manufacturing
MSC Software Company

Motivation / Thesis

Why are open interface standards so fundamentally important to us?

Process Simulation and Modelling tools are essential for getting process designs and products quickly to market ... the process industries remain extraordinarily complex. ***No one software vendor, asset owner, licensor, engineering company or academic has all the answers.***

… open process simulation software standards are designed to allow seamless communication between various pieces of software from different sources. By employing open simulation tools as the backbone, it is possible to ***assemble*** customized ***computational tools with a minimum of effort***.

Not only … open standards enable lower-cost simulation due to ease of re-use of simulation and modelling components, they will also drive innovation across the process industries. ***Novel and new solutions*** to previously unsolvable problems will be realized ***through increased plug-and-play simulation interoperability***, rapid integration of new ideas, and proliferation ***of 3rd party technology*** developers.

Source: http://www.aspentech.com/engineering/interfaces.aspx

Virtual process chains

- The 7 main groups of virtual process tools (own systematic)

Solid state forming

Solid state surface treatment

Primary shaping

Mechanical and thermal joining

Crash, fatigue, structural analysis

Heat treatment

Machining

- CFD – casting
- Powder metallurgy

(implicit & explicit codes)

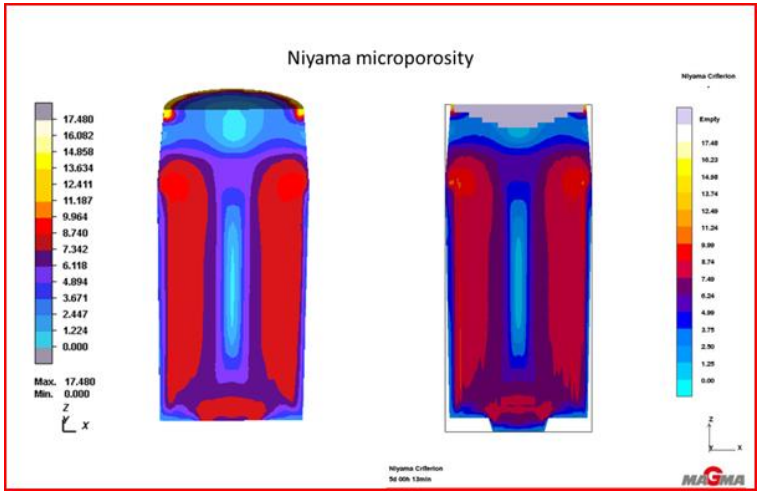
- Explicit codes (crash)
- Implicit & explicit codes for fatigue, structural analysis

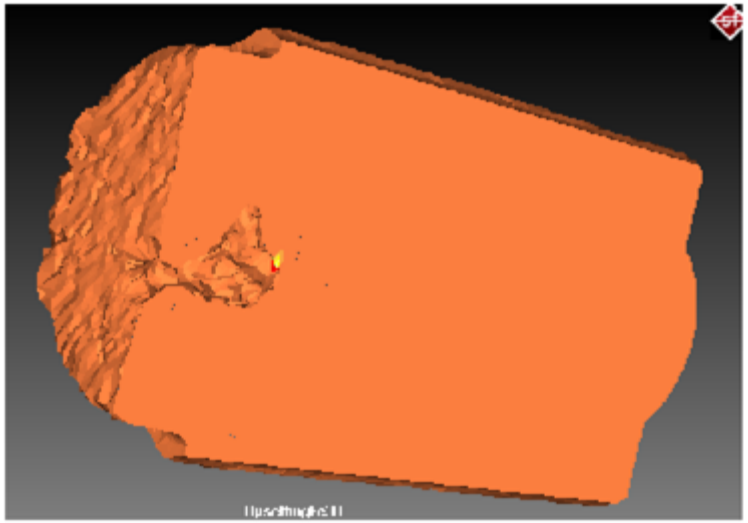
- implicit & explicit codes

FEM, FVM, FDM, BEM, others

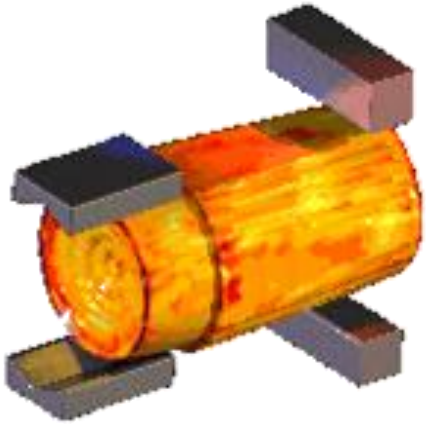
Example: Casted microstructure

- The main result values relevant for the subsequent forging steps are:
  - Macroscopic blowholes
  - Microscopic blowholes (porosity)





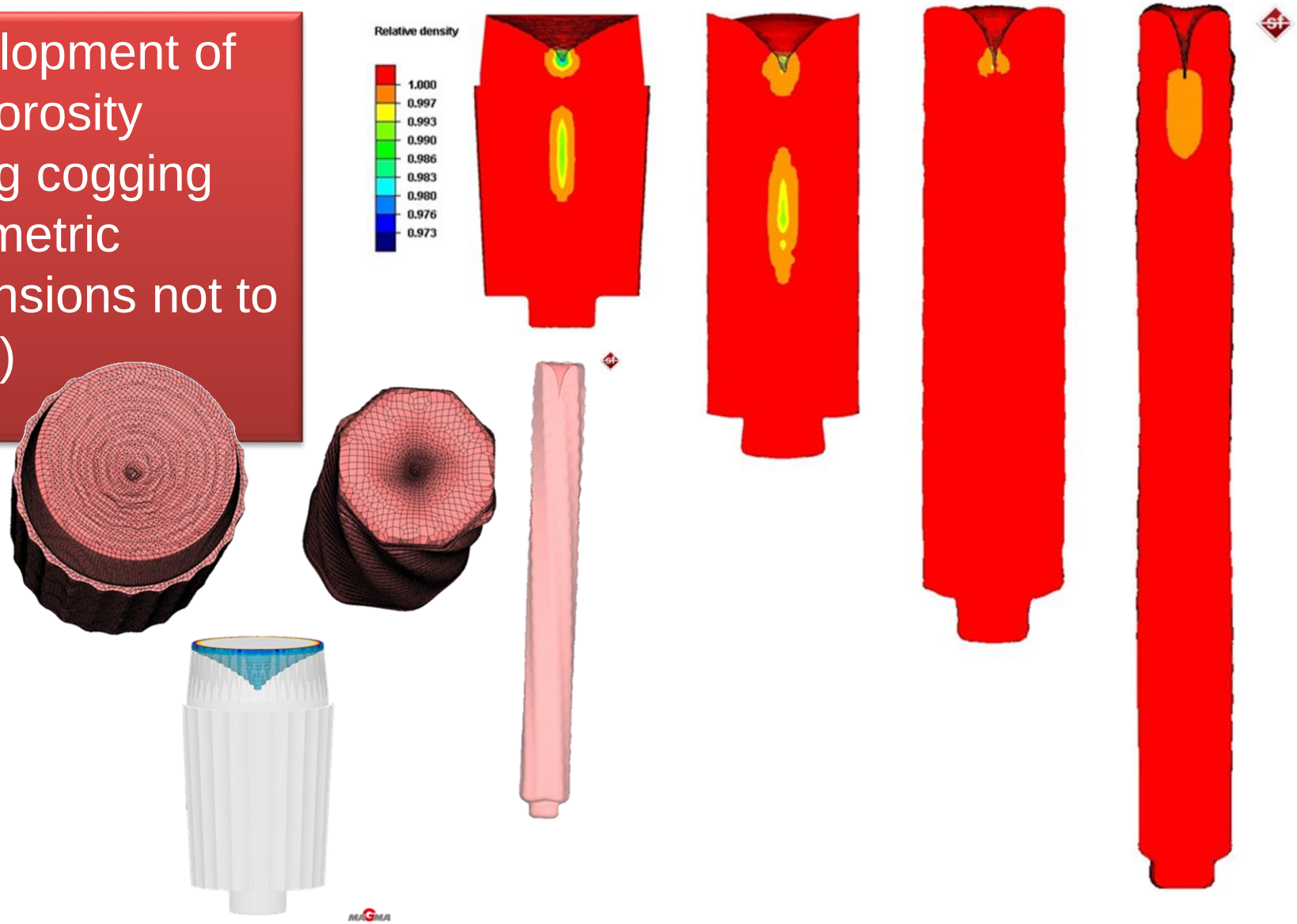
Explicit FV solver for casting



Implicit FE solver for cogging

Shrinkage and porosity

Development of the porosity during cogging (geometric dimensions not to scale)



Which level of interfacing?

Composability: the consistent representation of truth regarding the same objects as represented in the participating systems!

Modeling / Abstraction

Interoperability: the ability to exchange usable data between two systems and to invoke their services using the appropriate input parameters!

Simulation / Implementation

Integretability

Network / Connectivity

Level 6
Conceptual Interoperability

Level 5
Dynamic Interoperability

Level 4
Pragmatic Interoperability

Level 3
Semantic Interoperability

Level 2
Syntactic Interoperability

Level 1
Technical Interoperability

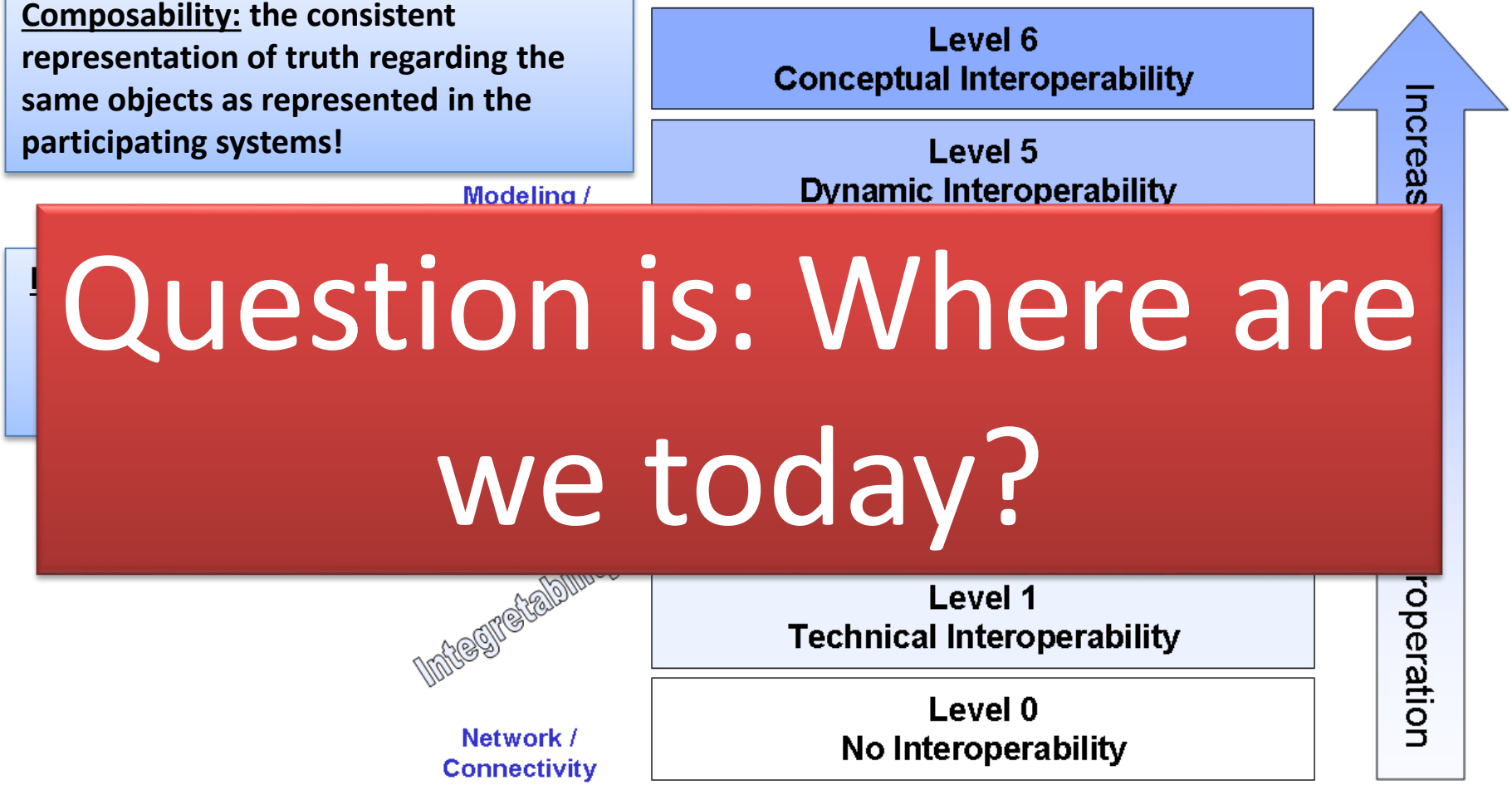
Level 0
No Interoperability

Increasing Capability for Interoperation

Source: http://en.wikipedia.org/wiki/Conceptual_interoperability

Which level of interfacing?

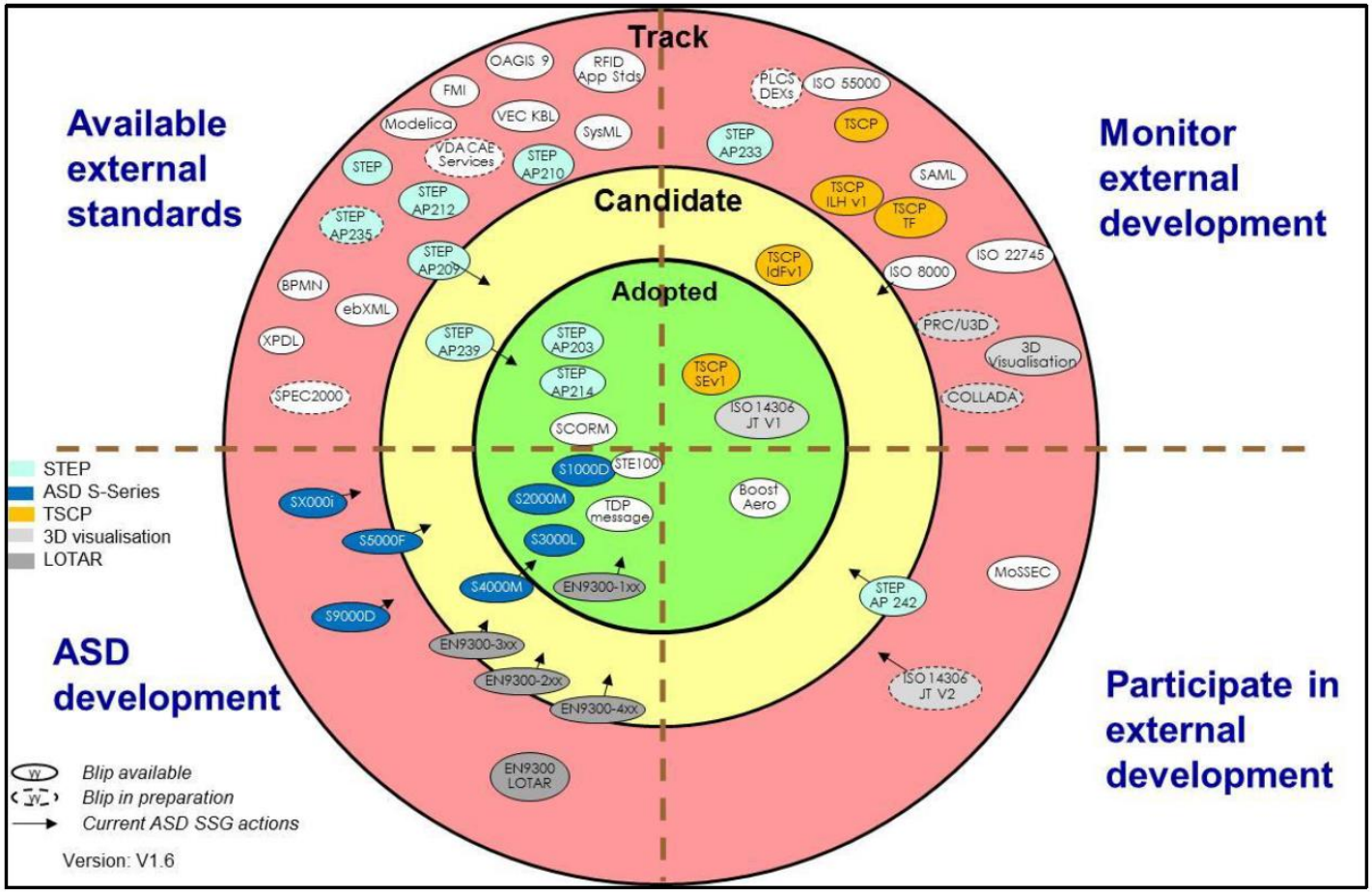
Composability: the consistent representation of truth regarding the same objects as represented in the participating systems!



Question is: Where are we today?

Source: http://en.wikipedia.org/wiki/Conceptual_interoperability

Map of standards in Aerospace



Source:
ASD STRATEGIC STANDARDISATION GROUP
Through Life Cycle Interoperability - A critical strategic lever for competitiveness
Rev 1.0, 9/05/2014; Ref: ASD/TLCI/01

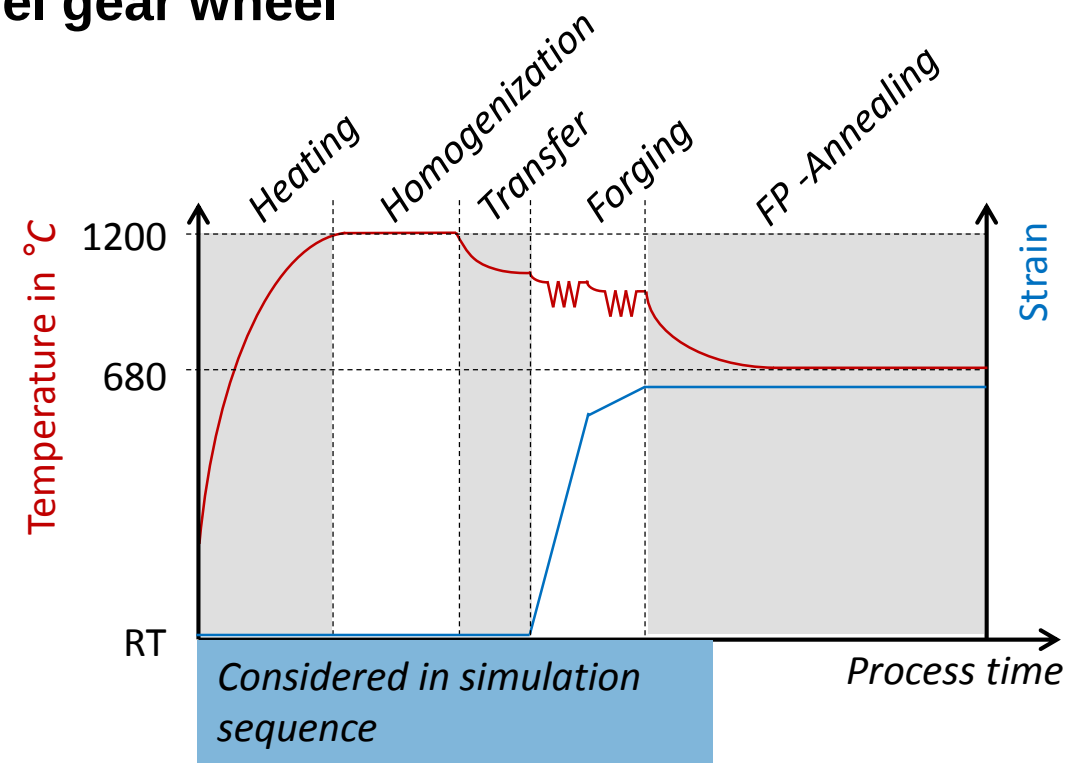
- Todays academic standard -

What does academic standard means?

- Phenomenon is known and can be modelled (at least partial)
- Needs special knowledge for preprocessing
- Mostly using several platforms / software solutions / special tools
- Many interfaces using different programming languages and are adapted to the specific project/platform
- Often time consuming and „off“-line → postvariables

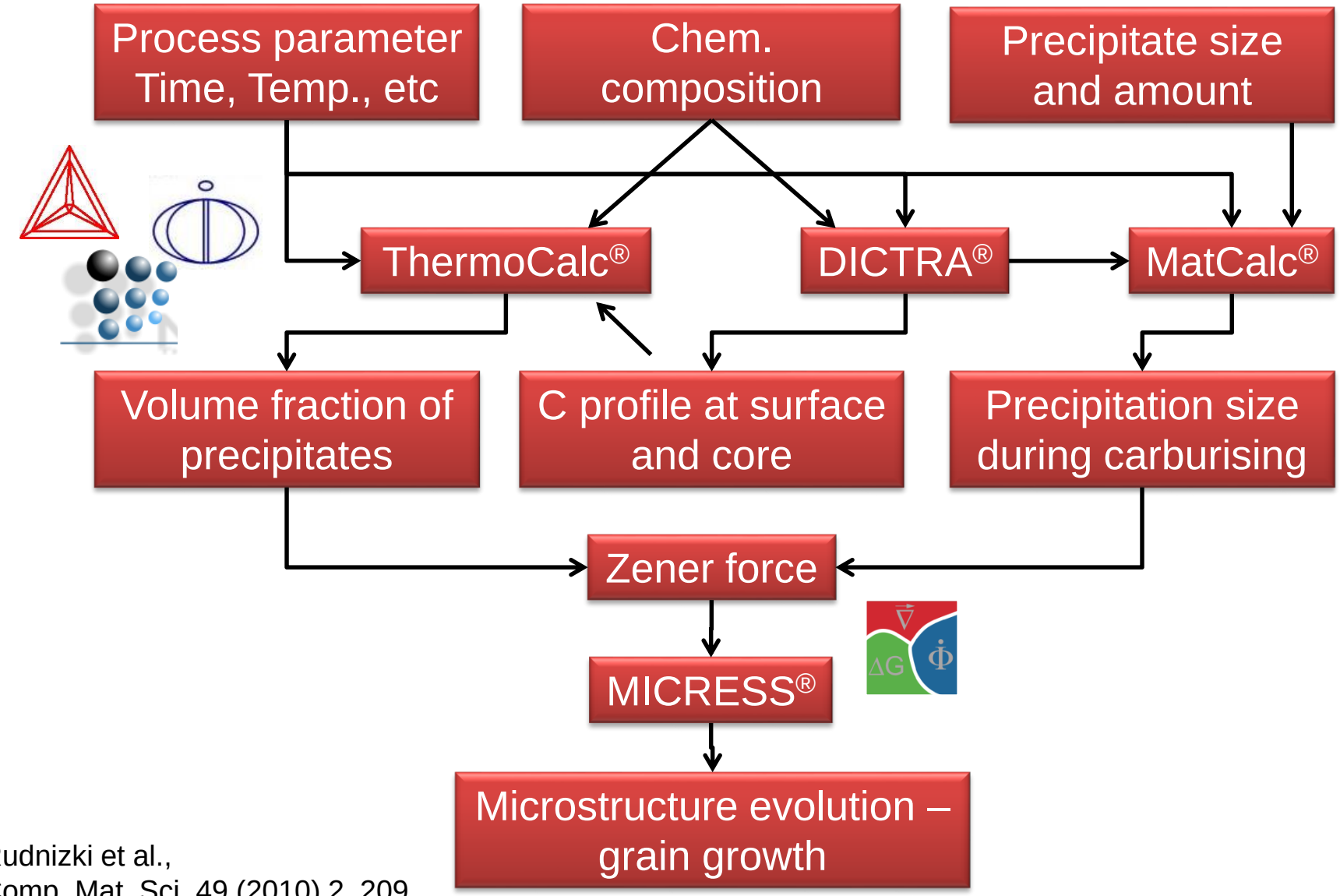
Process chain

- **Process design for the bevel gear wheel**
  - Forming strategy
  - Die design
  - Fine and homogeneous Microstructure (xRX, GG)



Source: IBF, RWTH Aachen

ICME modeling tools



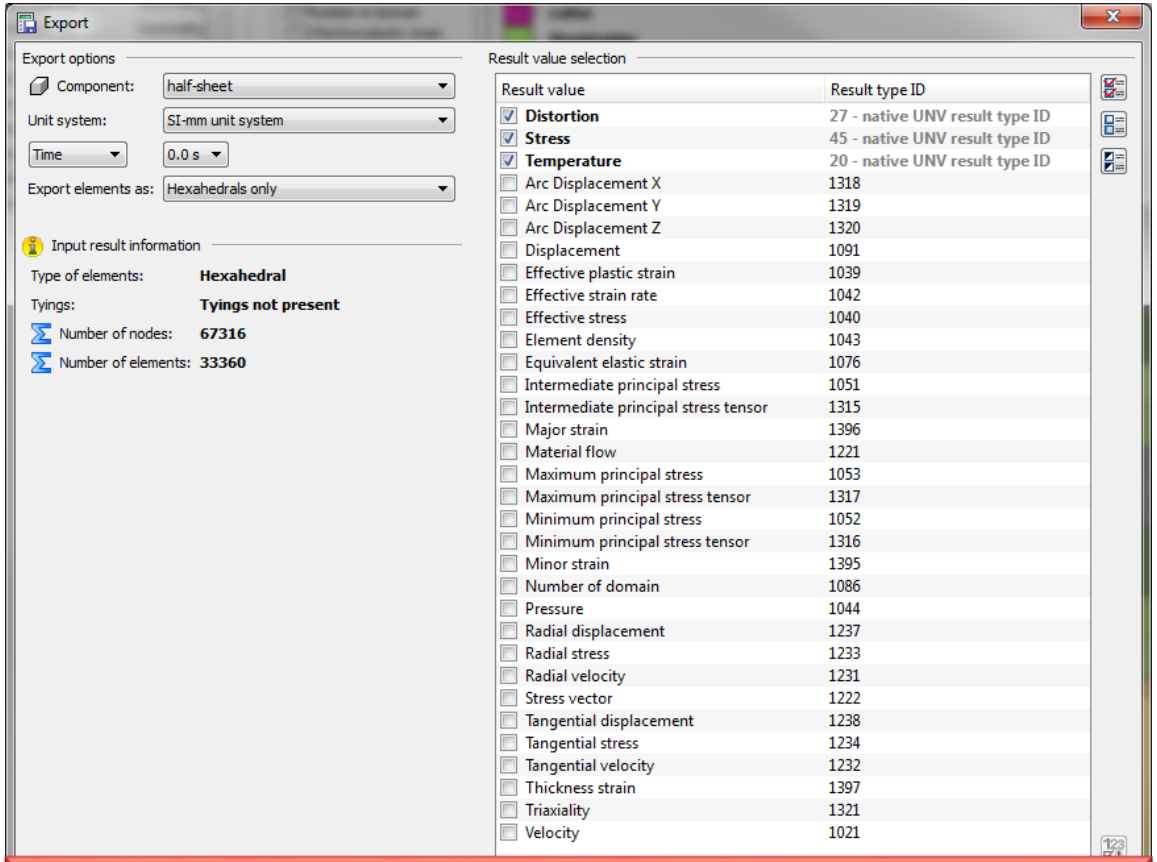
Rudnizki et al.,
Comp. Mat. Sci. 49 (2010) 2, 209

– Todays industrial standard – (wishlist … since it is not existing so far)

What should industrial standards provide?

- Easy-to-set-up
- No special programming know-how or additional editing
- One platform – not single solutions – uses standard tools and interfaces
- Flexible and exchangable (annual lease basis)
- „push-the-button“ solution
- No significant increase of pre- & post-processing

Data export interface (example)



UNV- standard – looks easy, right? Can be enhanced as well … the only problem seems to be the harmonization of terms (semantic level)

Typically such export interfaces provide geometry and local distribution of scalar values for one selected increment

Additionally so called tracking points can be used in order to save and export as well the history of selected data (i.e. strain vs. time history) → the only problem here is allocation of required memory which can be easily reach some TB

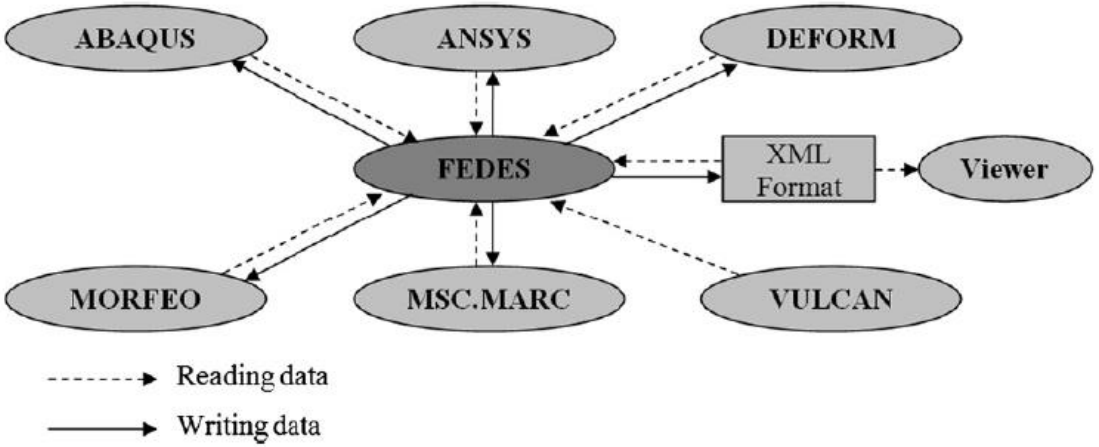
Solver interfacing

CAE Solver Interfacing

Solution supports a host of different solver formats for both import and export. Along with fully supported solvers, Solution provides a completely tailored environment (user profile) for each supported solver. It also provides the flexibility to support additional solvers through a unique and straightforward interfacing language

S.M. Afazov et al. / Advances in Engineering Software 47 (2012) 104–113

Or this:



Obviously there is more than just a data interface
But why we need this and is reality really that easy? → *"… unique and straightforward interfacing language"*

Interoperability of simulation tools along a process chain is way more than exchanging data only

Problems (inter)Facing in Reality

- Shell for crash is different from shell for forming
- Shell elements vs. solid elements

Different element formulation

Missing mapping technology

- No general algorithms available
- No mapping between shell and solid

Simulation-Systems are typically not compatible

Different mesh topology

Different implementation of material formulations

- Mesh for forming is not suitable for joining
- Different adaptive meshing concepts

Current trends are driven by property-oriented design – prediction of fatigue properties is/will become essential for the future !

- Different failure criteria
- Different “depth of physics”
- Different validity ranges (forming, welding)

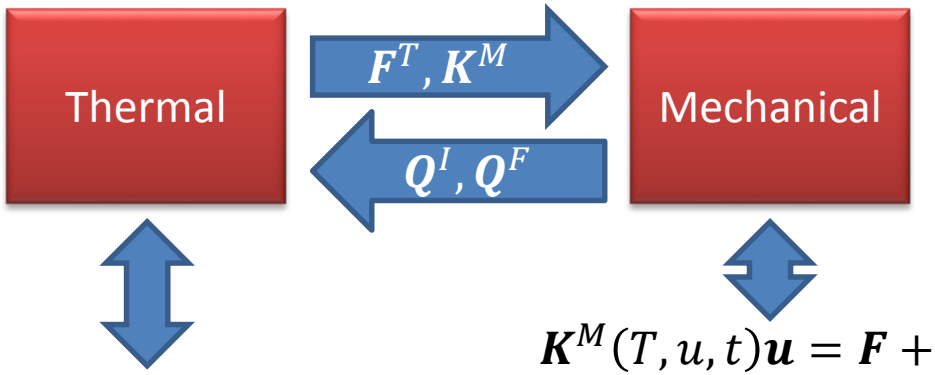
Main issue

- Constitutive material laws

… describe the response of materials to external forces/stresses (displacements)

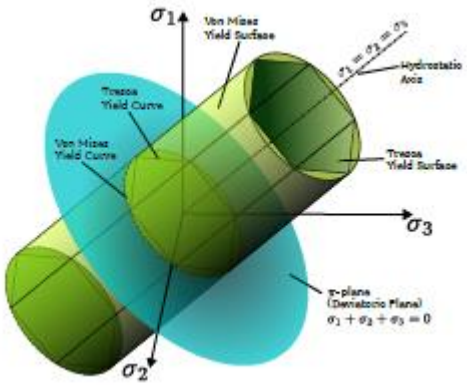
… leads to strain, strain rates that are used to calculate stress tensors

… flow rule is taken to determine whether the response is elastic or plastic



$$C^T(T)\dot{T} + K^T(T)T = Q + Q^I + Q^F$$

$$K^M(T,u,t)u = F + F^T$$

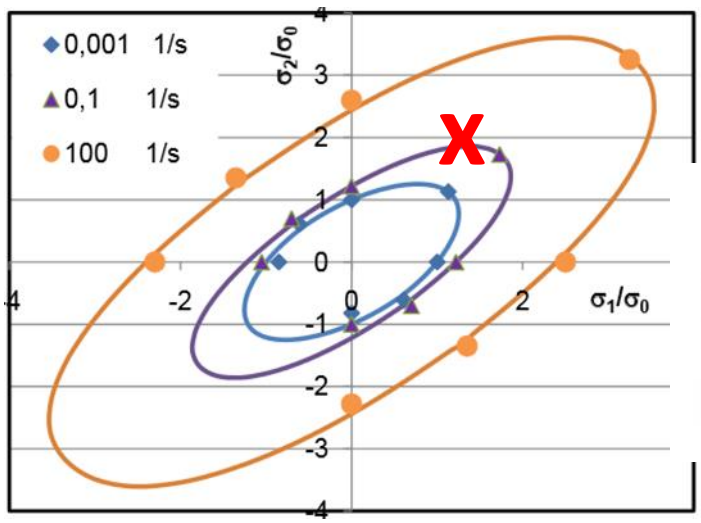


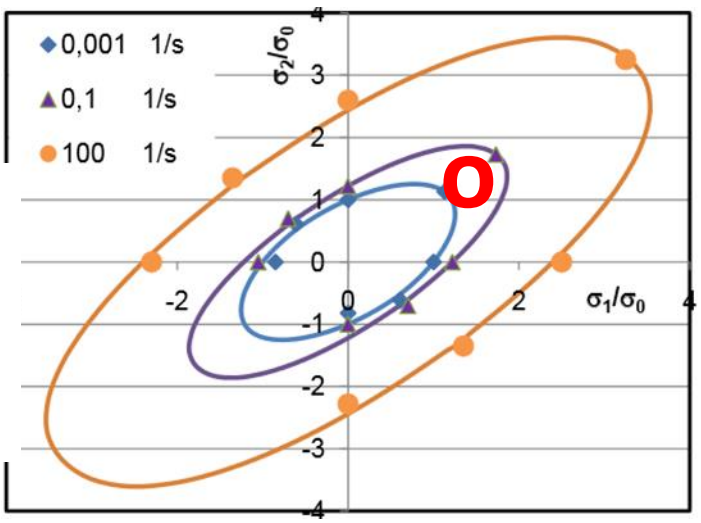
Mismatch of resultant data

Material data are organized in libraries and can be called to be an asset of a software – the cost for validated and well referenced material data are remarkable

System A uses yield locus model A

System B uses yield locus model B





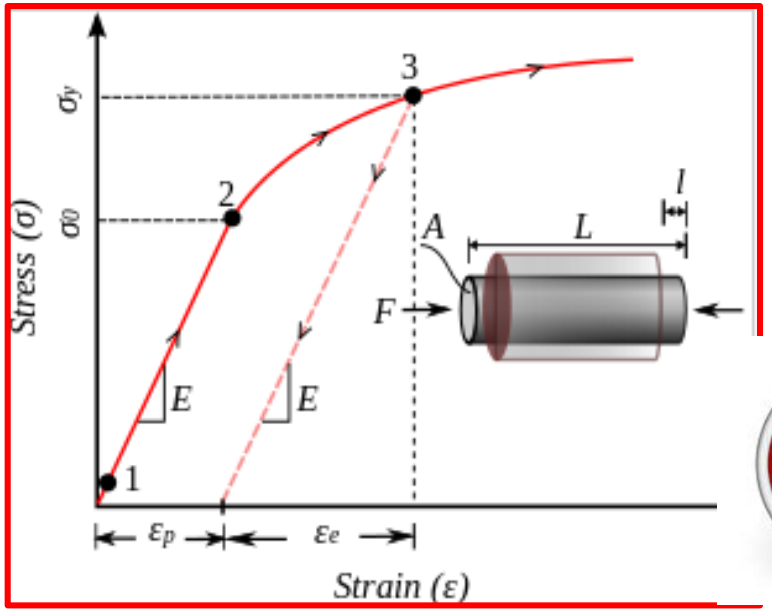
For a given external load and appropriate straining this models provides a flow stress of **“x” in plastic state**

… or vice versa

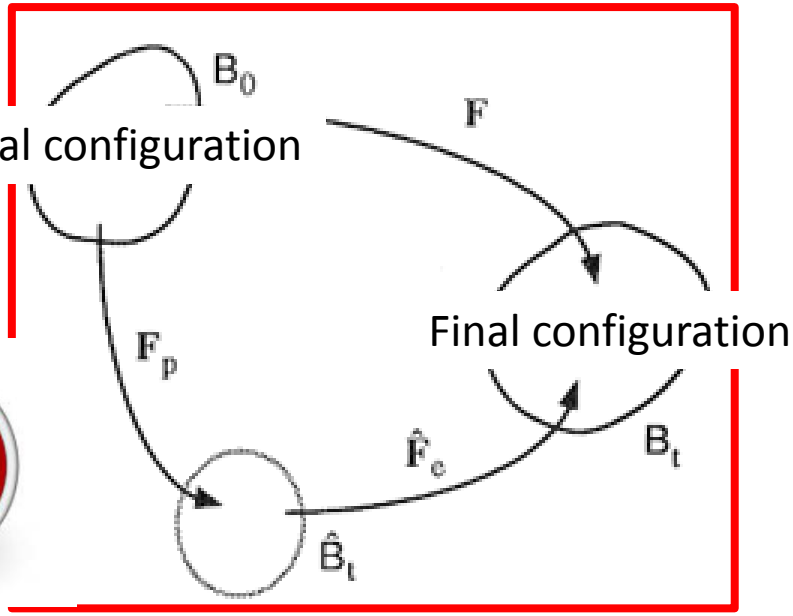
Exporting the data from A to B and put the strain/strain rate field into the yield locus model of B gives you flow stress **“o” → hence elastic state**

Incompatability I

Tool A uses “additive plasticity”



Tool B uses “multiplicative plasticity”



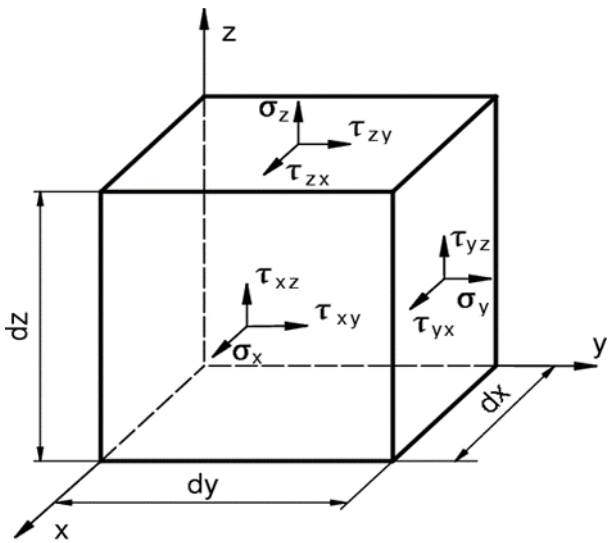
… requires deviatoric strains only

… requires full strain tensor and strain gradients

Incompatability II

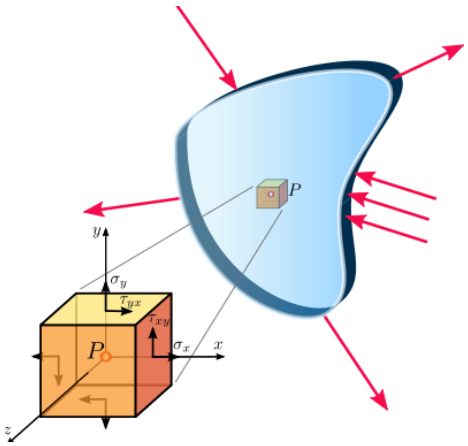
Tool A uses solid elemets

$$\sigma_{ij} = \begin{pmatrix} \sigma_x & \tau_{xy} & \tau_{xz} \\ \tau_{yx} & \sigma_y & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \sigma_z \end{pmatrix} \quad with \quad \begin{matrix} i = x, y, z \\ j = x, y, z \end{matrix}$$



Tool B uses shell elements

$$\sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} & 0 \\ \sigma_{21} & \sigma_{22} & 0 \\ 0 & 0 & 0 \end{bmatrix} \equiv \begin{bmatrix} \sigma_x & \tau_{xy} & 0 \\ \tau_{yx} & \sigma_y & 0 \\ 0 & 0 & 0 \end{bmatrix}$$



… there are mapping algorithms to convert A to B but: error propagation!!!

Conclusions

Shown examples refer to the 3rd and 4th level of interoperability

3rd level

Beyond the ability of two or more computer systems to exchange information, **semantic interoperability** is the ability to **automatically interpret the information exchanged meaningfully and accurately** in order to produce useful results as defined by the end users of both systems. To achieve semantic interoperability, both sides must refer to a common information exchange reference model. The content of the information exchange requests are unambiguously defined: what is sent is the same as what is understood.

4thlevel

Pragmatic Interoperability is reached when the interoperating systems **are aware** of the **methods and procedures** that each system is employing. In other words, the use of the data – or the context of its application – is understood by the participating systems; the context in which the information is exchanged is unambiguously defined. This layer puts the (word) meaning into context.

Source: http://en.wikipedia.org/wiki/Conceptual_interoperability

Which level of interfacing?

Composability: the consistent representation of truth regarding the same objects as represented in the participating systems!

Modeling / Abstraction

Interoperability: the ability to exchange usable data between two systems and to invoke their services using the appropriate input parameters!

Simulation / Implementation

Integretability

Network / Connectivity

Level 6
Conceptual Interoperability

Level 5
Dynamic Interoperability

Level 4
Pragmatic Interoperability

Level 3
Semantic Interoperability

Level 2
Syntactic Interoperability

Level 1
Technical Interoperability

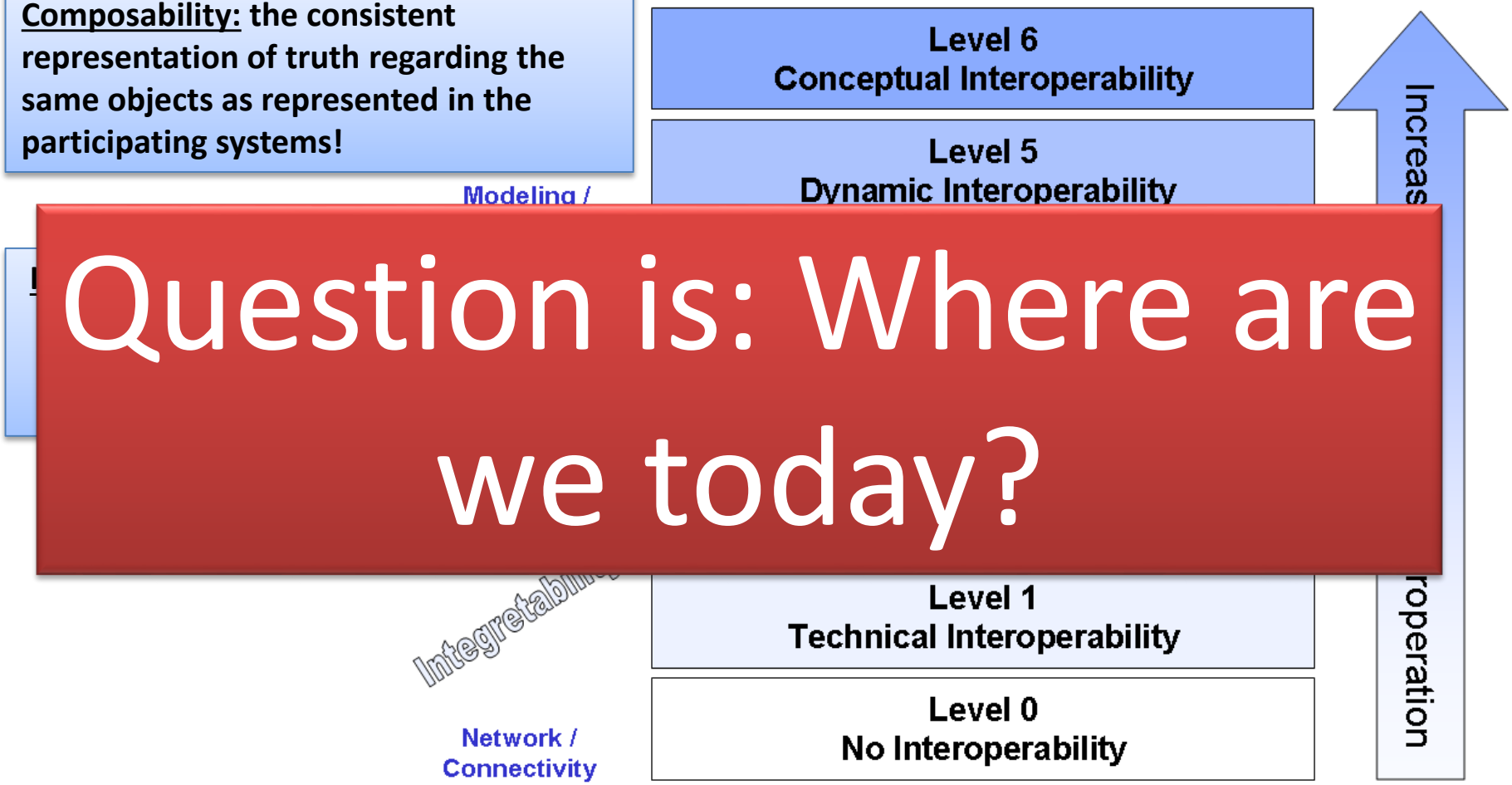
Level 0
No Interoperability

Increasing Capability for Interoperation

Source: http://en.wikipedia.org/wiki/Conceptual_interoperability

Which level of interfacing?

Composability: the consistent representation of truth regarding the same objects as represented in the participating systems!



Question is: Where are we today?

Source: http://en.wikipedia.org/wiki/Conceptual_interoperability

We are here …

Composability: the consistent representation of truth regarding the same objects as represented in the participating systems!

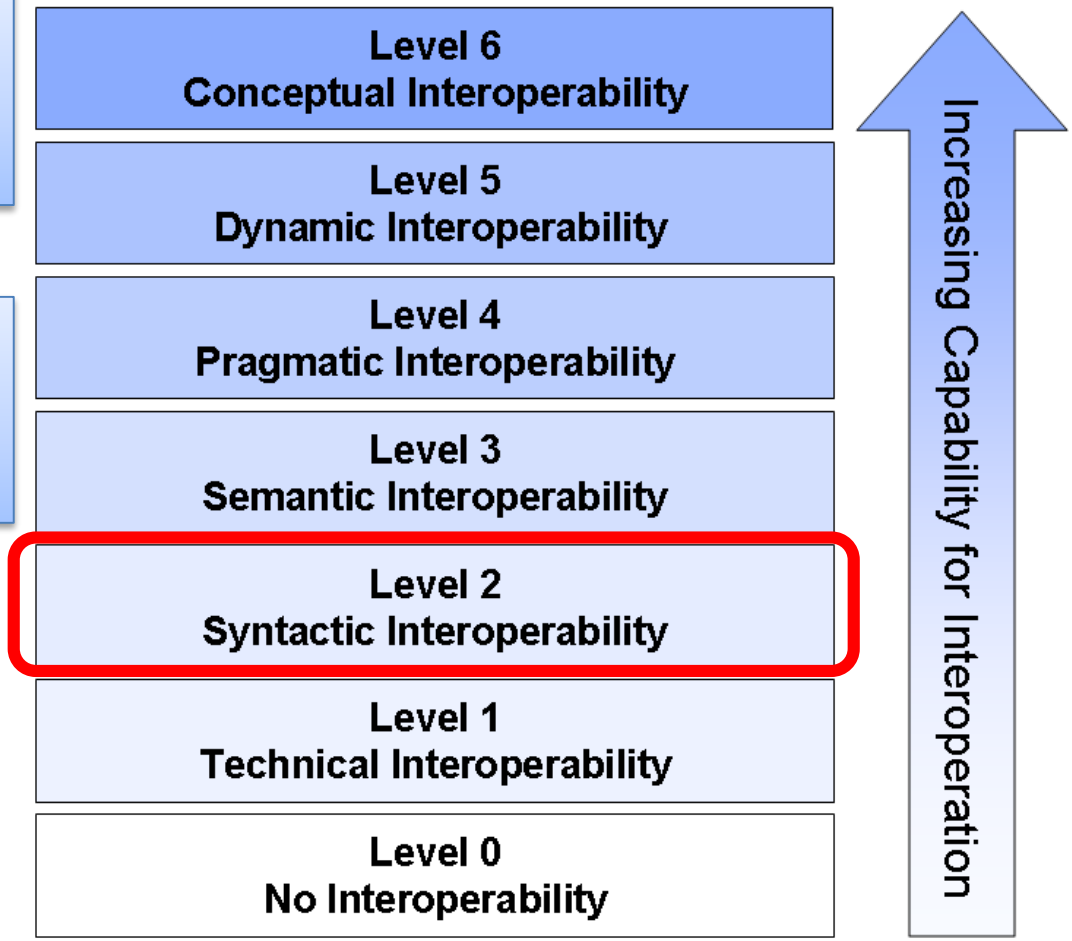
Modeling / Abstraction

Interoperability: the ability to exchange usable data between two systems and to invoke their services using the appropriate input parameters!

Simulation / Implementation

Integretability

Network / Connectivity



Source: http://en.wikipedia.org/wiki/Conceptual_interoperability

Still the main issues

Solver type

Data format

Data accuracy

Operating system

Scope of information

Element type

Exchange format (risk anlysis)

resolution

Data/ interface viewer?

geometry

Philosophy (of interfacing)

GUI driven inputs

mapping

ASCII or BIN?

Where to define data?

Which effects?

Historical paths (i.e straining)

Tables or equations

units

Parsing?

Thank you!

simufact engineering gmbh
Tempowerkring 19
21079 Hamburg

Tel.: 040 790 128 - 000
Fax: 040 790 128 - 199

info@simufact.de
www.simufact.com

simufact
Simulating Manufacturing
MSC Software Company