MDE Adoption in Industry: Challenges and Success Criteria

Parastoo Mohagheghi (SINTEF),
Miguel Fernandez & Juan Martell (Telefónica),
Mathias Fritzsche & Wasif Gilani (SAP)

Motivation

- Model-Driven Engineering (MDE) has been promoted as a solution to handle increased complexity of software development.
- MDE is characterized by:
  - Raising the abstraction level and hiding platform details → improved communication, portability of solutions
  - Models in all phases → improved communication
  - Domain-specific languages and platforms → improved communication, easier modelling, improved productivity
  - Generation of artefacts from models → remove manual work, increased productivity, improved quality and traceability
- But:
  - So far not widely adopted by industry, as also verified by a review of literature.

Motivation

- Model-Driven Engineering (MDE) has been promoted as a solution to handle increased complexity of software development.
- MDE is characterized by:
  - Raising the abstra… …orm details → improved communica…
  - Models in all ph… …n
  - Domain-sp… …mproved communication, easier modelli…
  - Generation of arte… …models → remove manual work, increased productivity, improved quality and traceability
- But:
  - So far not widely adopted by industry, as also verified by a review of literature.

Context

- We are involved in the FP6 MODELPLEX project (EU IP, 2006-2010):
  - The goal of MODELPLEX is to develop solutions for applying MDE in complex software system development.
    - Composition and weaving, traceability, language engineering, model discovery and understanding, simulation and model-based testing, and model-based system management.
  - We plan to evaluate the MODELPLEX solutions empirically in the context of the four use case providers; i.e., Telefónica, SAP, Thales and WesternGeCo.
  - There is a varying degree of earlier experience with MDE in these companies and they focus on different aspects; for example some more on Domain-Specific Languages, while others more on testing.

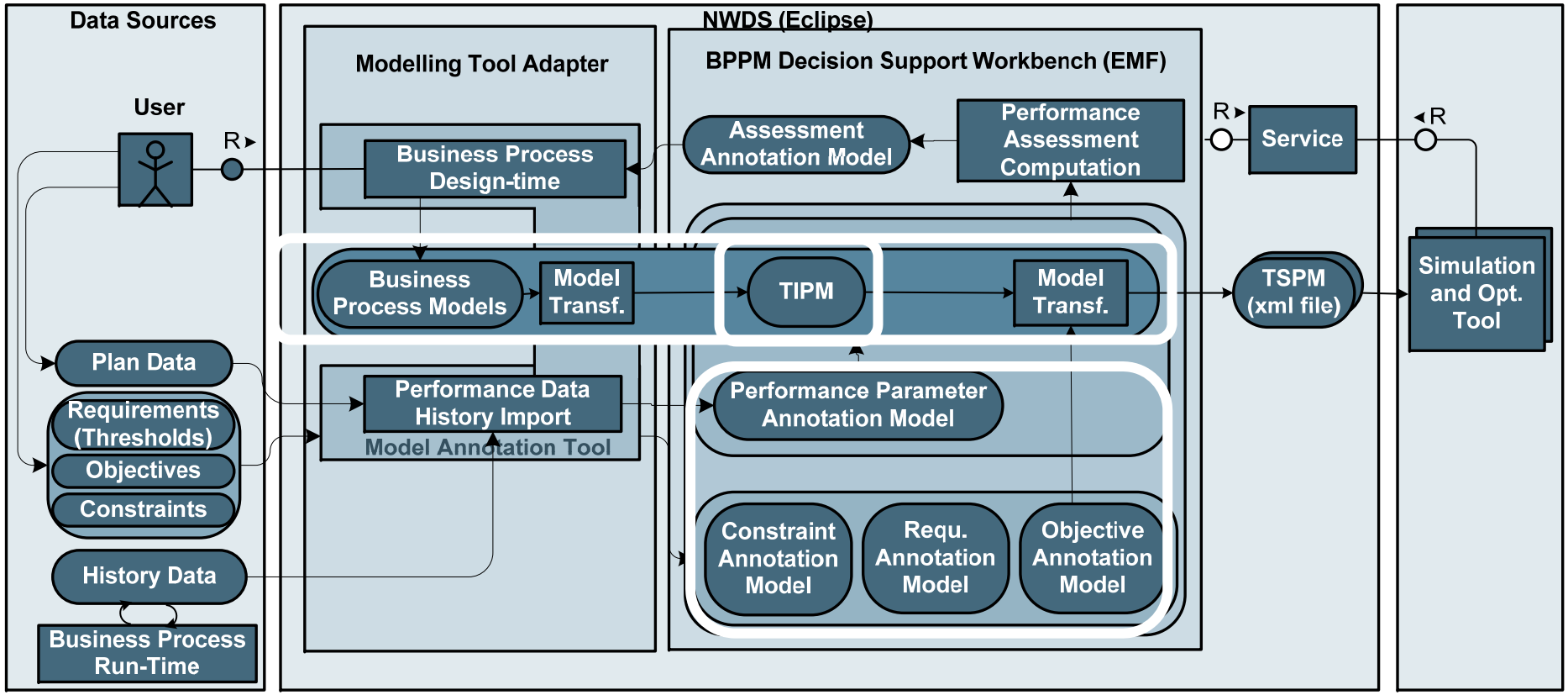
SAP- Motivation

- Developing an environment for Composite applications; i.e., applications that combine loosely coupled services with their own business logic.
- Need to involve domain experts in the software development process.
  - E.g. business domain experts, consultants, etc., having no programming skills, need to be able to model and deploy customized applications suited for specific business requirements.
- Need of supporting non-technical users with regards to non-functional requirements
  - E.g. Impact of design decisions on performance.

SAP-Current Work (I)

- Extension of currently used business process modelling tools with performance related decision support (what-if simulations and optimizations)
  - Utilization of MDE artefacts at business process run-time to optimize resource usage.
  - Decision support for the modifications and extensions of productive business processes.
  - Decision support for the definition of new business processes.

SAP-Current Work (II)



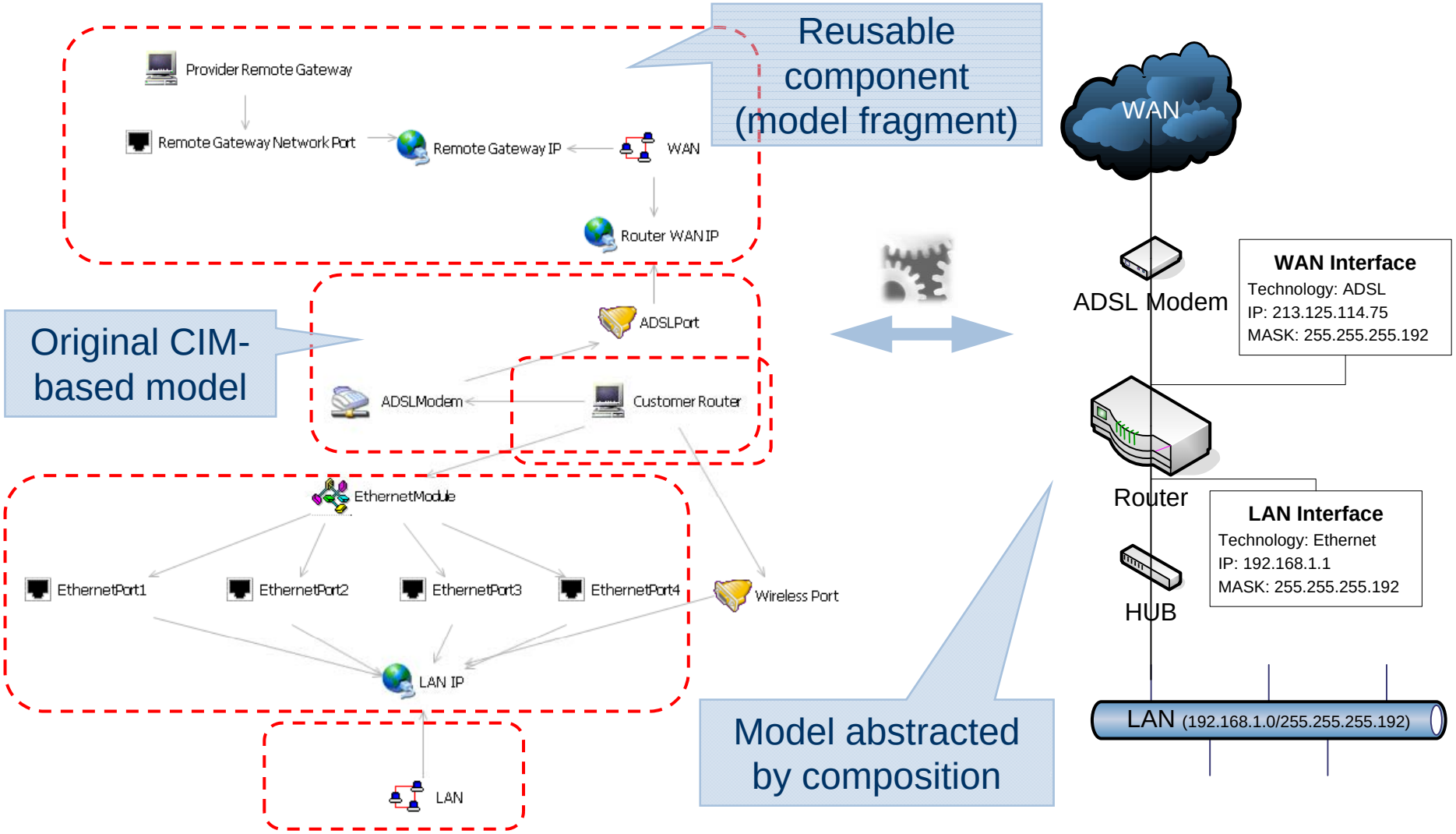
SAP- Challenges

- Tool Support to define and maintain long model transformation chains.
- Traceability through long model transformation chains.
- Dealing with numerous model/meta-model artefacts.
  - Mega-Modelling?
- Extension of models/meta-models, which might be provided by 3rd parties, without polluting them.
  - Model weaving?

Telefónica - Motivation

- Earlier attempts with UML failed due to lack of proper tools and inability to maintain code and models in-synch.
- Earlier experience with domain-specific frameworks steered us towards Domain-Specific Modelling.
- MODELPLEX work focused on the following areas:
  - DSM approach in the telecommunications domain.
  - Modelling in different levels of abstraction.
  - Model verification techniques.
  - Composition / weaving of models at design time and run time:
    - Service composability.
    - Model-based system management.

Telefónica - Current Work



Telefónica - Challenges

- Creating a DSM solution requires meta-modelling and tool development skills and also:
  - Getting the people in the domain to agree upon a standard syntax.
  - Making the DSL interact properly with anything outside.
- Model synchronization, roundtrip engineering and management of large sets of models are not solved yet.
- Too many languages and tools need to be used:
  - Are the techniques for handling complexity in danger of making the software engineering process itself too complex?
- The main question that an organization might be asking itself right now is “do I really need MDE?”

Challenge #1: Solutions for the Domain

- DSM requires expertise in:
  - Meta-modelling: identifying right concepts without redefining UML
  - Meta-modelling environments
  - Integration of languages
  - Maintenance and updating languages
  - Integration of tools
- Possible long-term solutions:
  - More domain-specific profiles available
  - Improvement of metamodeling tools

Challenge #2: Complexity in the Development Process

- We have all activities required before (requirement management, analysis, design, implementation, testing, configuration etc.) but with
  - Developing languages, editors and generators
  - Adapting tools
  - Integrating tools
- Does it mean increased complexity of software development at the moment or forever?
- Possible long-term solutions:
  - Tool chains

Challenge #3: Expertise and Training

- Both Challenge #1 (domain appropriate solutions) and Challenge #2 (complexity of software development) requires higher expertise:
  - Do we have access to these experts?
  - What do we do for training future ones?
- Possible long-term solutions:
  - Incremental adoption of MDE to allow learning
  - Courses and training
  - Better documentation of tools

Conclusions

- We have earlier experience with all aspects of software development but MDE introduces new challenges.
- What are the gains?
  - Productivity increase if we manage to get things right (and in the long term, not in single projects);
  - Improved quality if the approach allows prevention of defects or earlier detection;
  - Better communication if we achieve domain appropriate solutions.
- Main challenges:
  - Managing complexity of MDE,
  - Moving from small cases to large industrial applications,
  - Expertise, expertise, expertise!

Thanks! Any questions?