
SESAM
Nov 2008

Projekthantering vid MDE

Konfigurations och projekt/linjestyrning med Model-Driven Engineering

Göran Calås

Project Manager
Systems Architect

Nov 20, 2008
goran.calas@SAABGroup.com
+46 768 967167

Göran Calås

- Saab AB:
  - Technology Transfer Group – Model Driven Engineering
  - Patent Manger for BU Saab Training
- Saab Training Systems:
  - Software Architect
  - Project Manager
  - Requirements Management Databases
  - Model-based software engineering
- Research: Robustness Analysis and Technology Forecasting method
- Teacher: Design Patterns, Systems Architectures, …

AGENDA

- Configuration Management
- Project/Line Management

Definitions

- QVT (Query / View / Transformation)
- MDA[OMG] (Model-Driven Architecture) for SW design. Supports MDE
- MOF[OMG] (Meta-Object Facility), a formally defined UML
- MDE (Model-Driven Engineering)
- MBSE (Model Based Software Engineering)

Configuration Management in MDE

Configuration Management
Traditional SW Engineering vs. MDE

Traditional SW ENG	MDE
• One release at the time	• Work with multiple releases, e.g. a Change should be active in the following releases, and variants
• Few product variants	• New product variants are composed of existing model assets

Configuration Management
Traditional SW Engineering vs. MDE

Traditional SW ENG

- Config Item = File, or directory

MDE

- Config Item =
- Model, Feature, Use Case, Requirement, Test results, Release, Iteration, Group work, Aspect (AP), Model configuration, Code generator, Model transform, MDE Tool environment, My-sandbox, Test model, model of metrics, Usage model, MoDAF artifacts, Domain specific models, Quality status e.g. reviewed/tested…

Line Manager and Project Manager – New skills demanded

- Advanced Configuration Management skills
- MDE / MBSE modeling and analysis skills
- Tool environment awareness
- MDE process knowledge

- If everything is inside the model:
  - It is stupid to rely on progress status rumors
  - When the live up-to-date status exists inside the model

- Models are everywhere, not just stored as UML, e.g. MoDAF, parametric models, domain specific models.

The Project Manager / Architect, defines the software engineering project as a build model

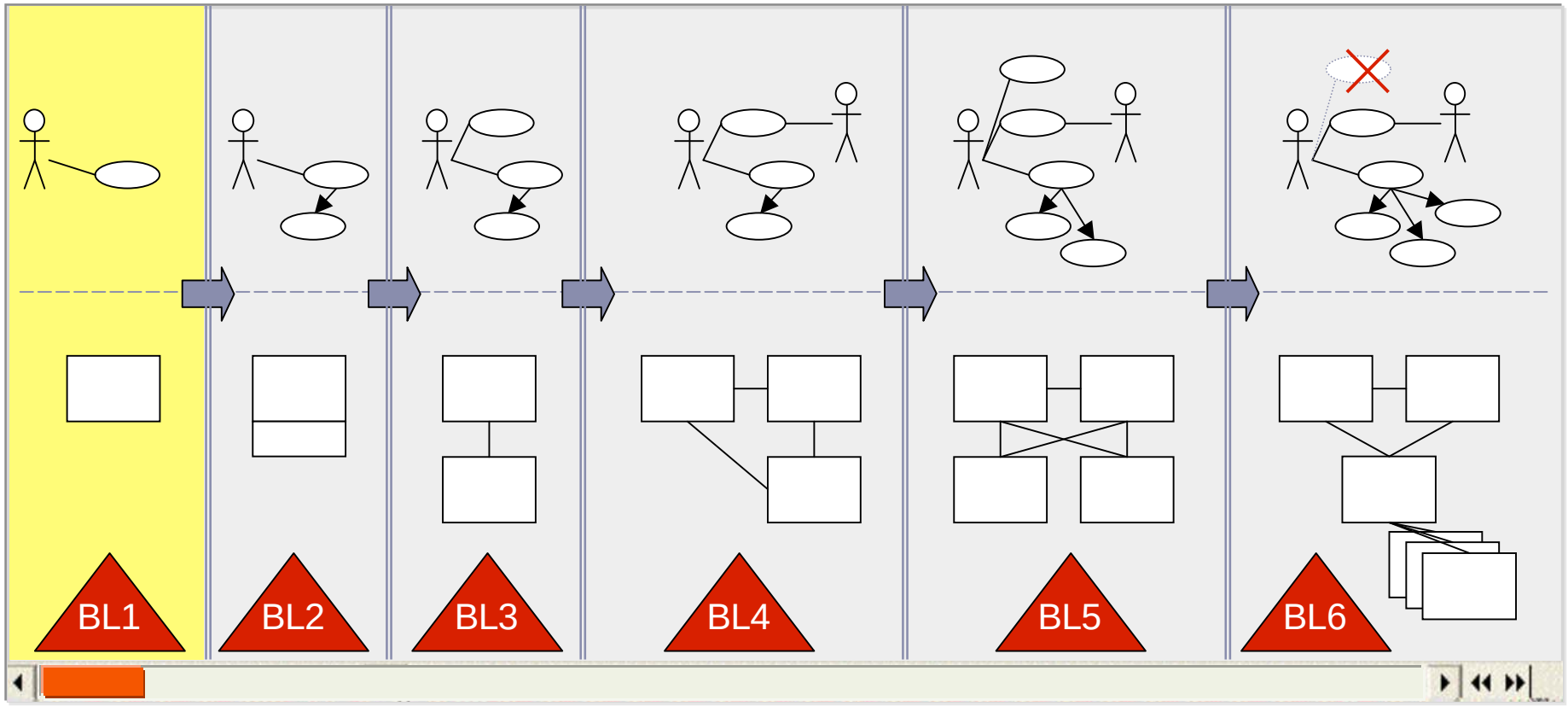
- A project is to satisfy certain goals, e.g defined by S.M.A.R.T.
- The project goal is broken down to work packages
  - Each work package is to bring new value, and changes to the environment
  - If building a house, or software an incremental plan, or model is described
  - By stating economical, and project management properties for each model item, it is possible to calculate estimated project costs, as well as following up project work.
  - Features, High level requirements, Use Cases, package diagrams, component diagrams are usually good stand points for project planning and visual follow up.
  - By visualizing progress, it is possible to use color coded use cases, and baseline plans, showing expected progress and real progress.

Configuration Management in MDE vs traditional SWE → Consequences

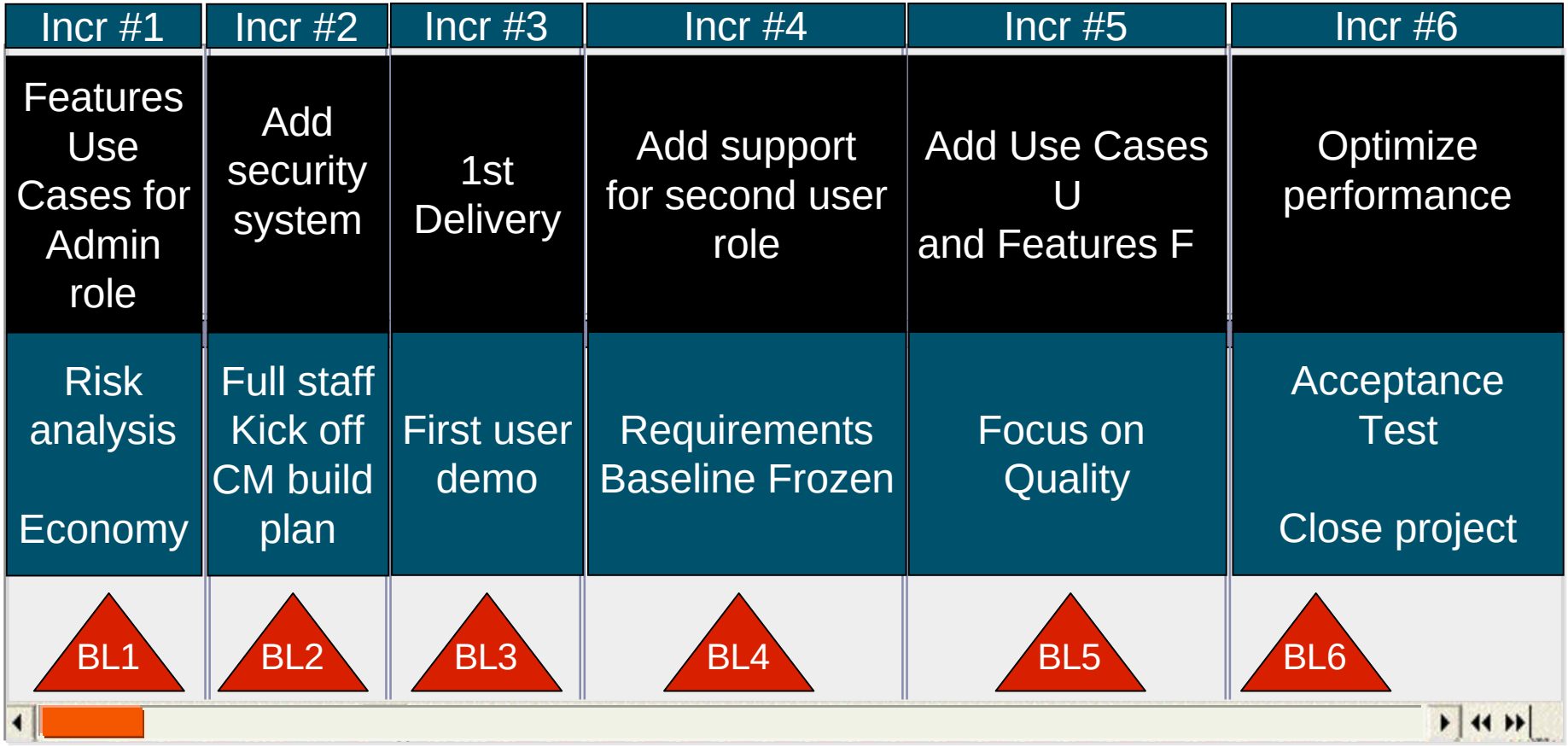
- The model is mapped to files constituting configuration items
- Change management
  - Current practice: ClearCase UCM Activities, are a good base for gathering configuration items, related to useful tasks performed.
  - Better solution: Track Change Management objects and Activities inside model
  - Even better solution: Meta-model manages configuration items and activities transforming configuration items included in old baseline, into new configuration items included in new baseline.
- Similar problems was solved in .NET / CLR & Java environments, using manifest file and version concepts.
  - See DLL-HELL in Windows; and JAR-HELL.

- How would this look?

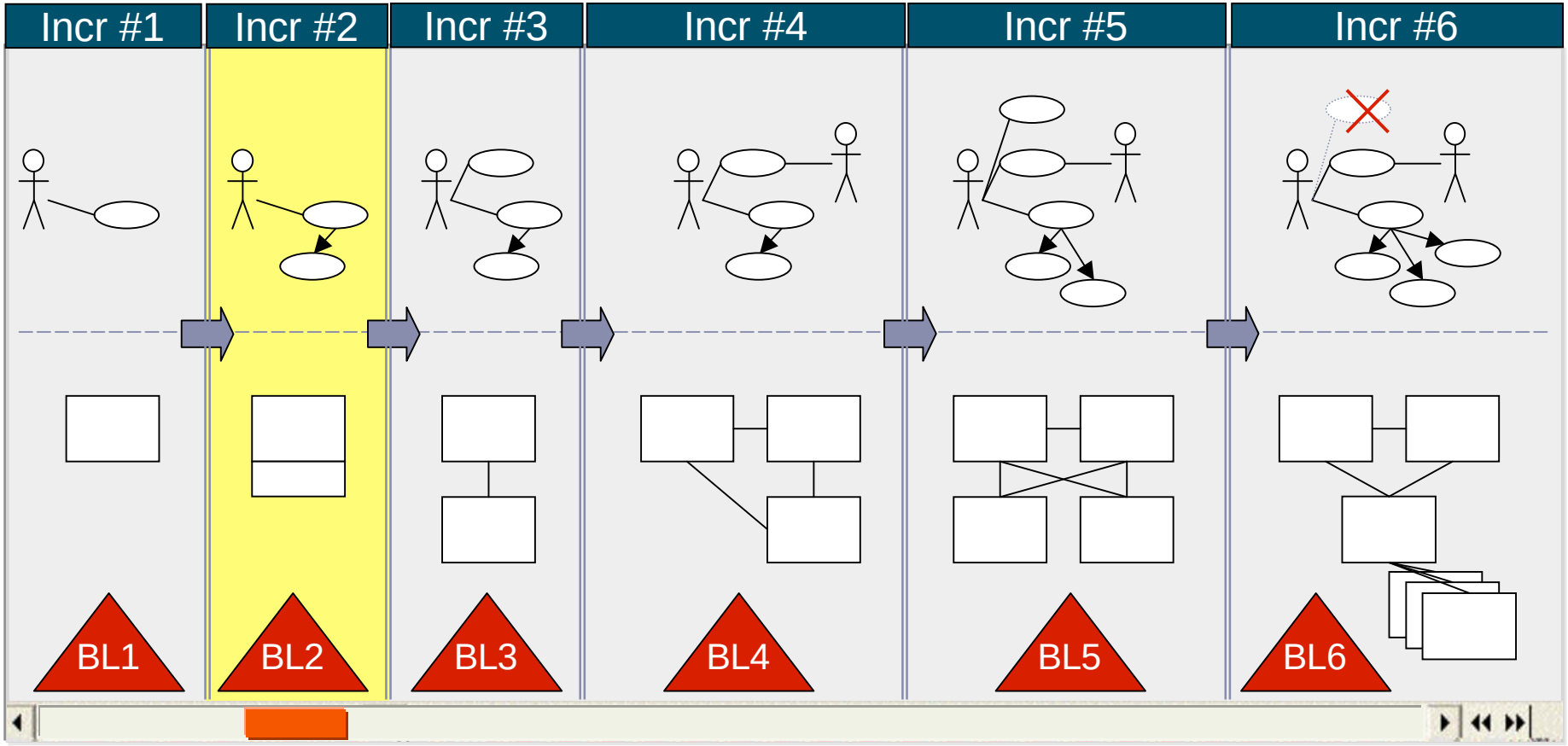
CM in a MDE life-cycle perspective
A time machine…



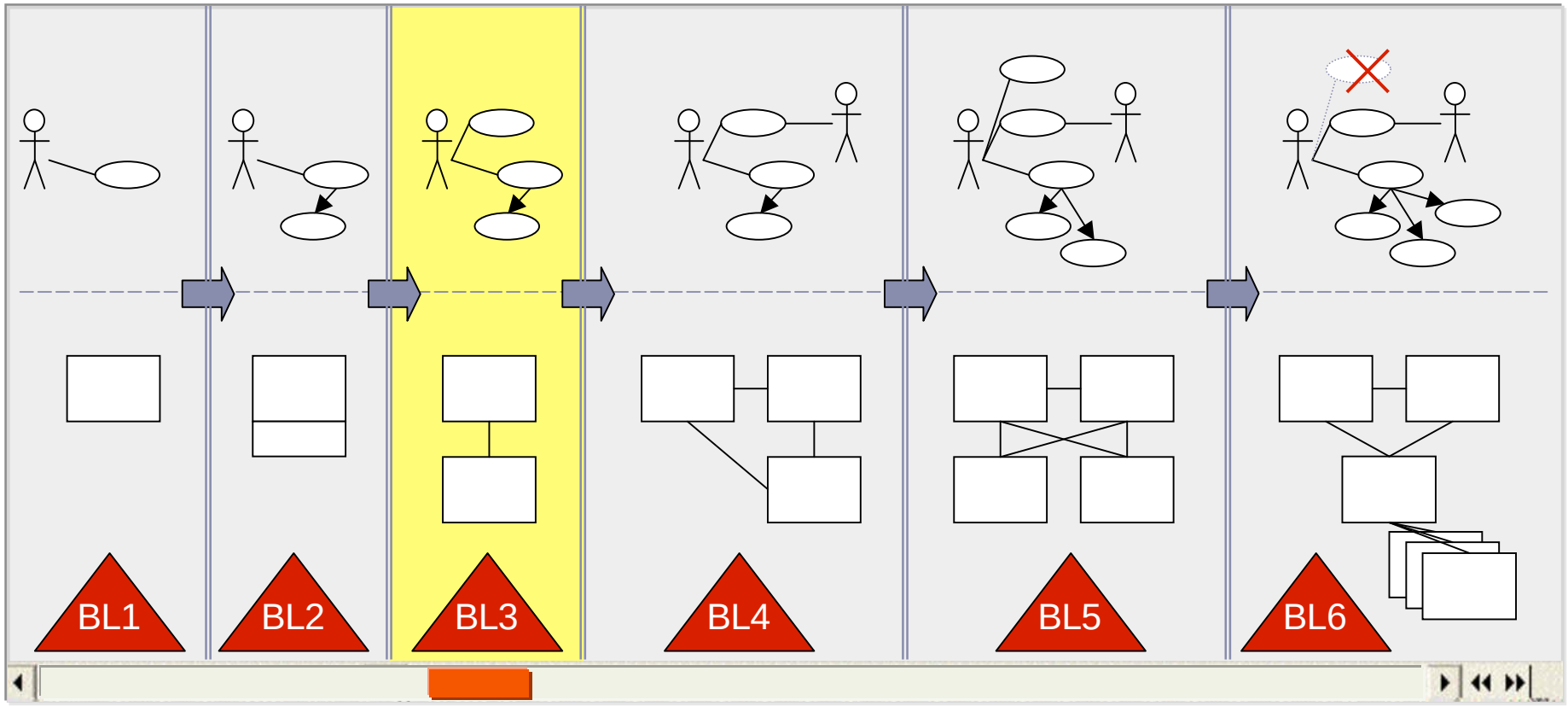
CM in a MDE life-cycle perspective
Create a project, and build plan…



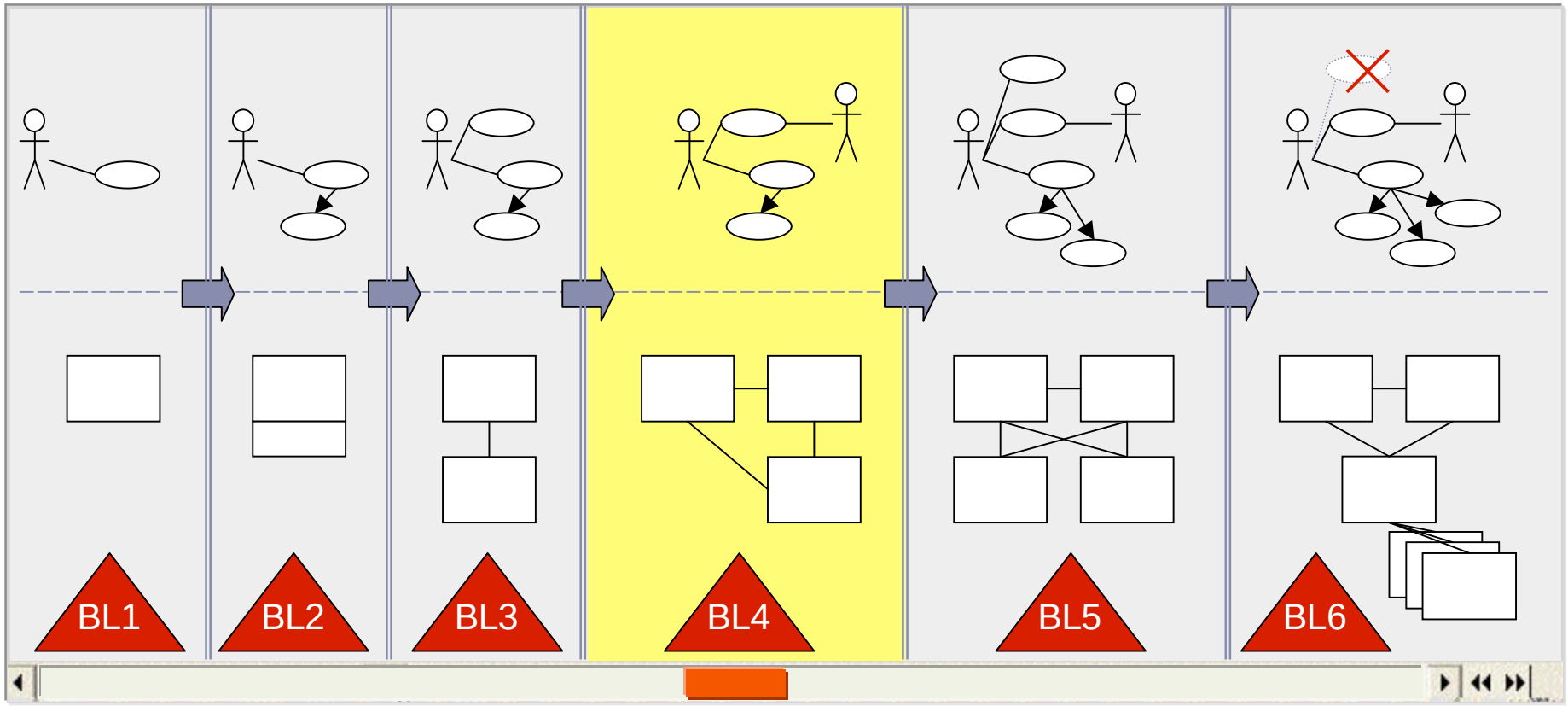
CM in a MDE life-cycle perspective
A time machine…



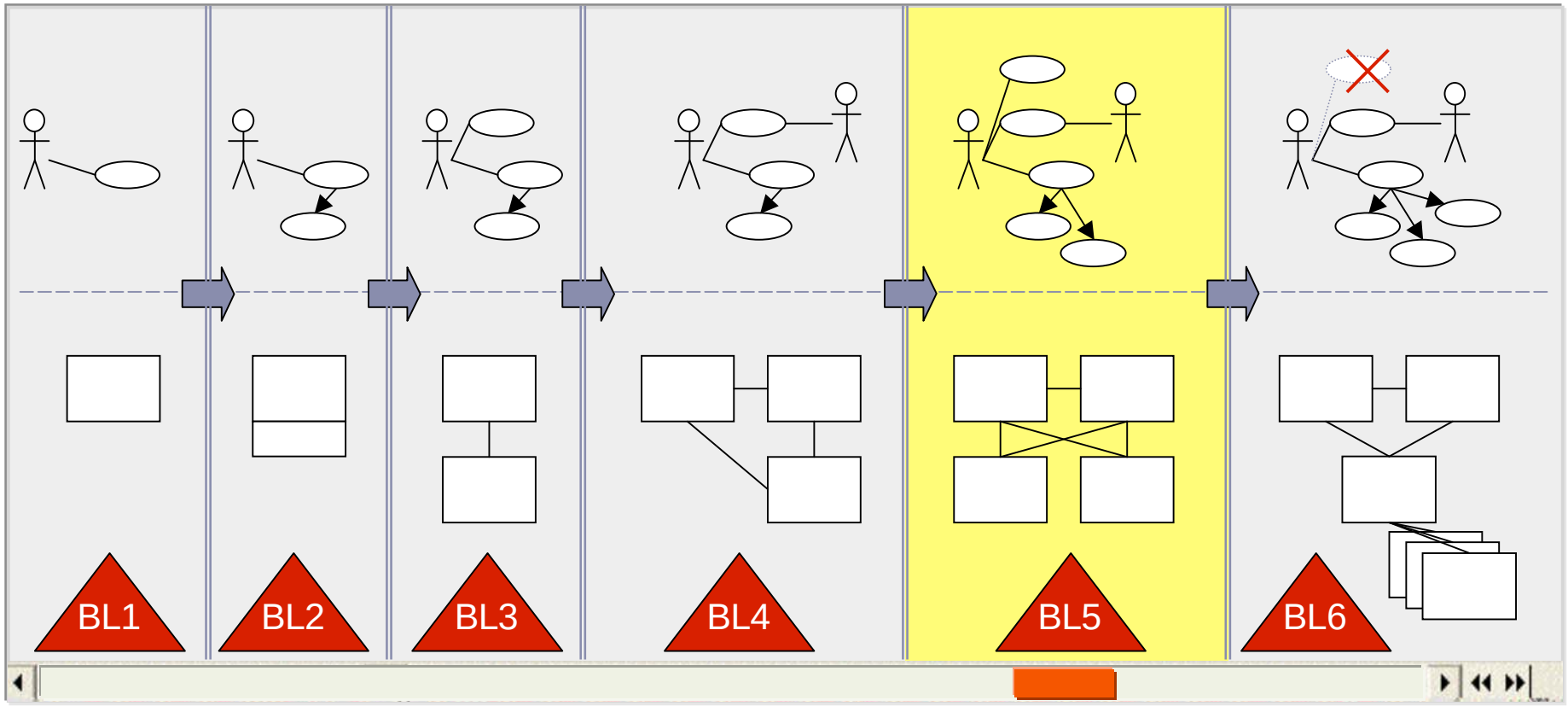
CM in a MDE life-cycle perspective
A time machine…



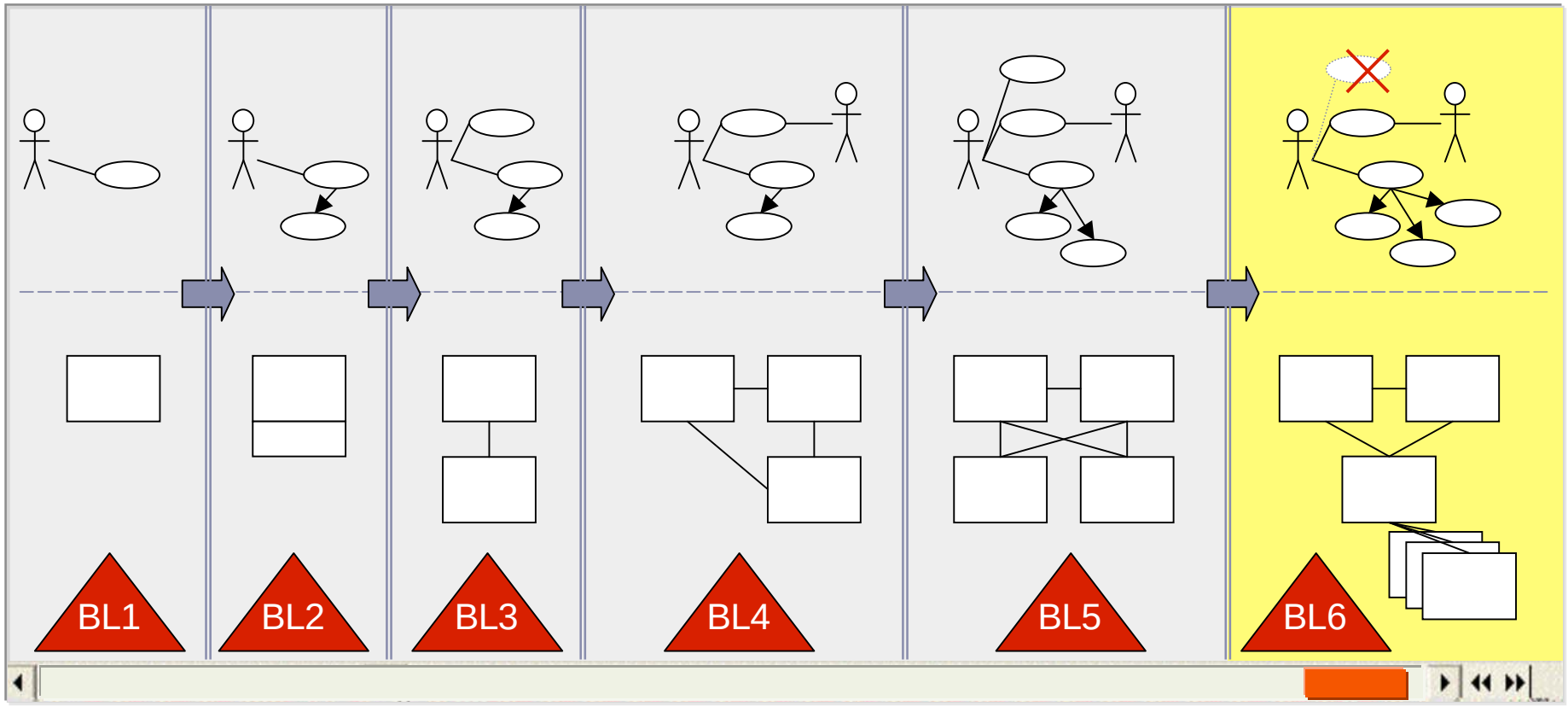
CM in a MDE life-cycle perspective
A time machine…



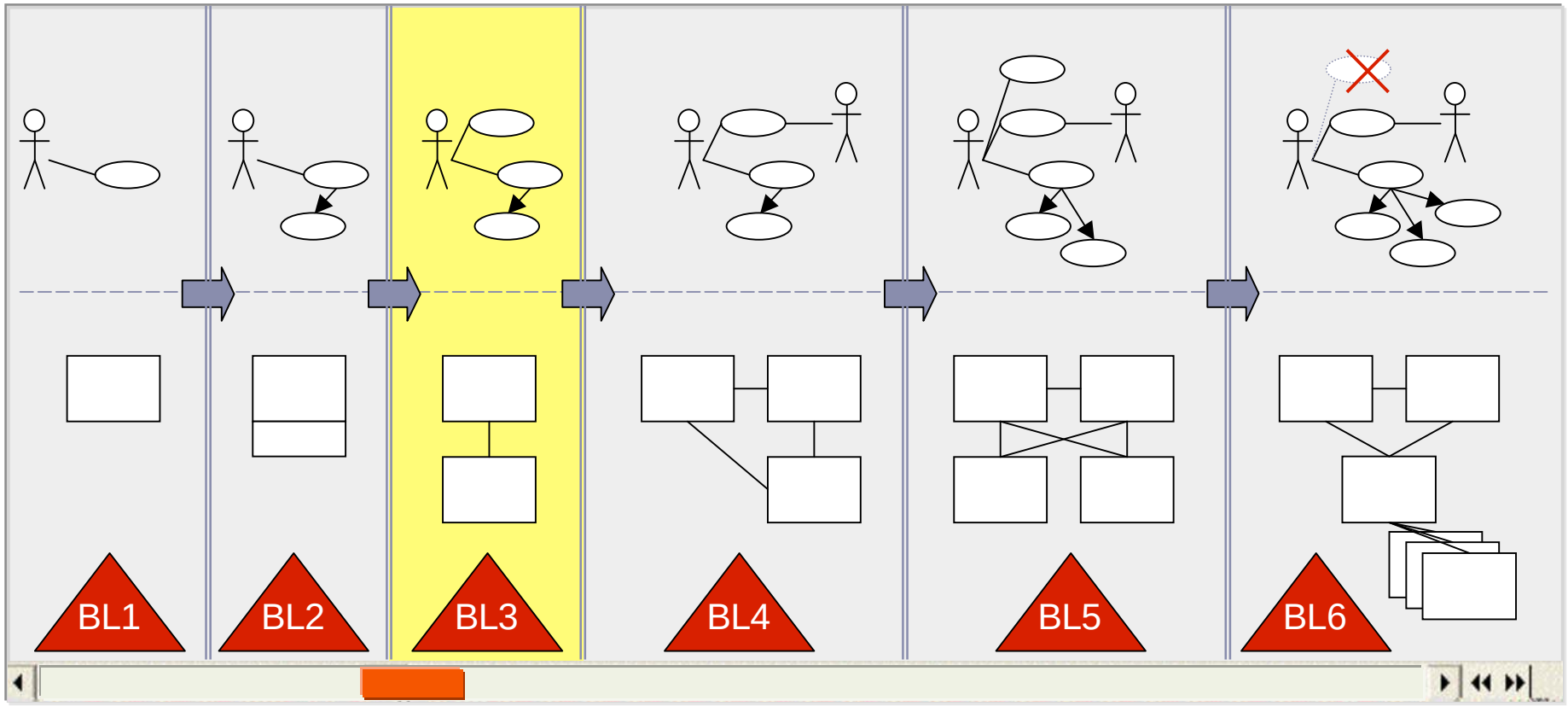
CM in a MDE life-cycle perspective
A time machine…



CM in a MDE life-cycle perspective
A time machine…



CM in a MDE life-cycle perspective
A time machine…

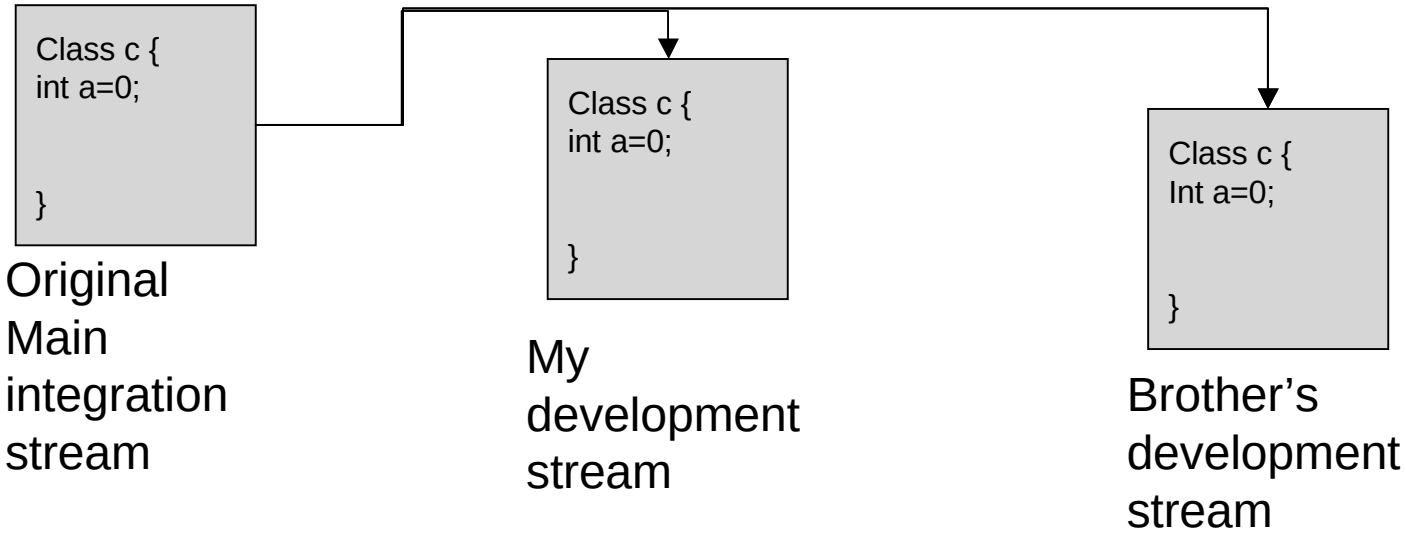


CM in a MDE life-cycle perspective
What you really gets, today…

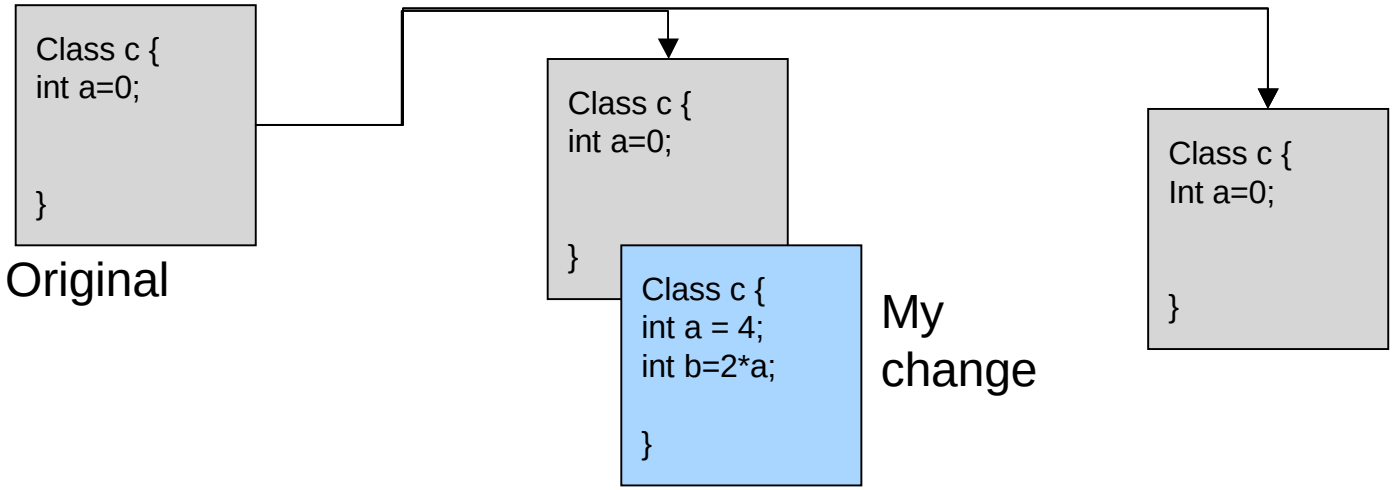
- Ability to
  - Map model elements to files on a directory structure
  - Map model elements to XMI files
  - Create a baseline based on multiple files

- Problems when
  - New Modeling Tools are released twice a year
  - You want to work on multiple baselines, e.g. baseline comparison, metrics comparison, roll-back of old features, etc.
  - Merging changes three model versions using Base Aware Merge

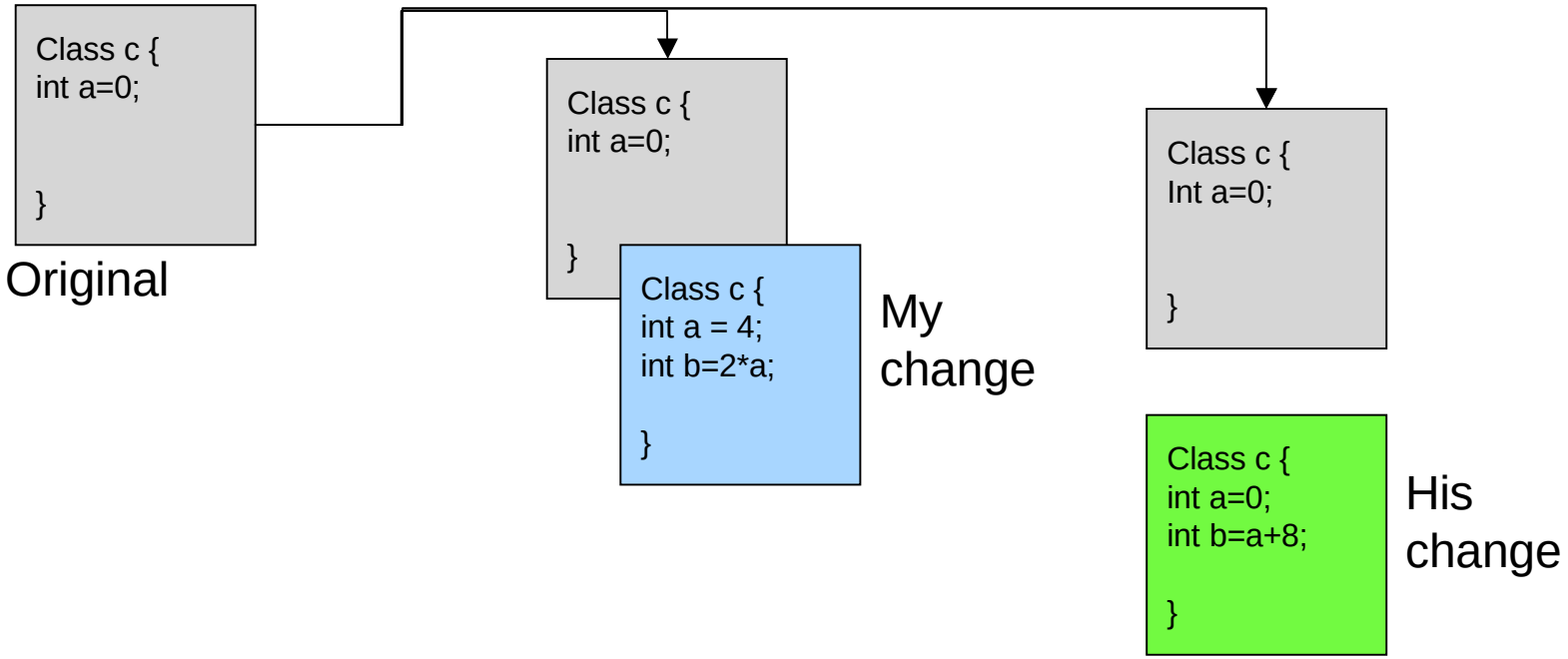
Base-aware Merge → 3-way-merge from hell



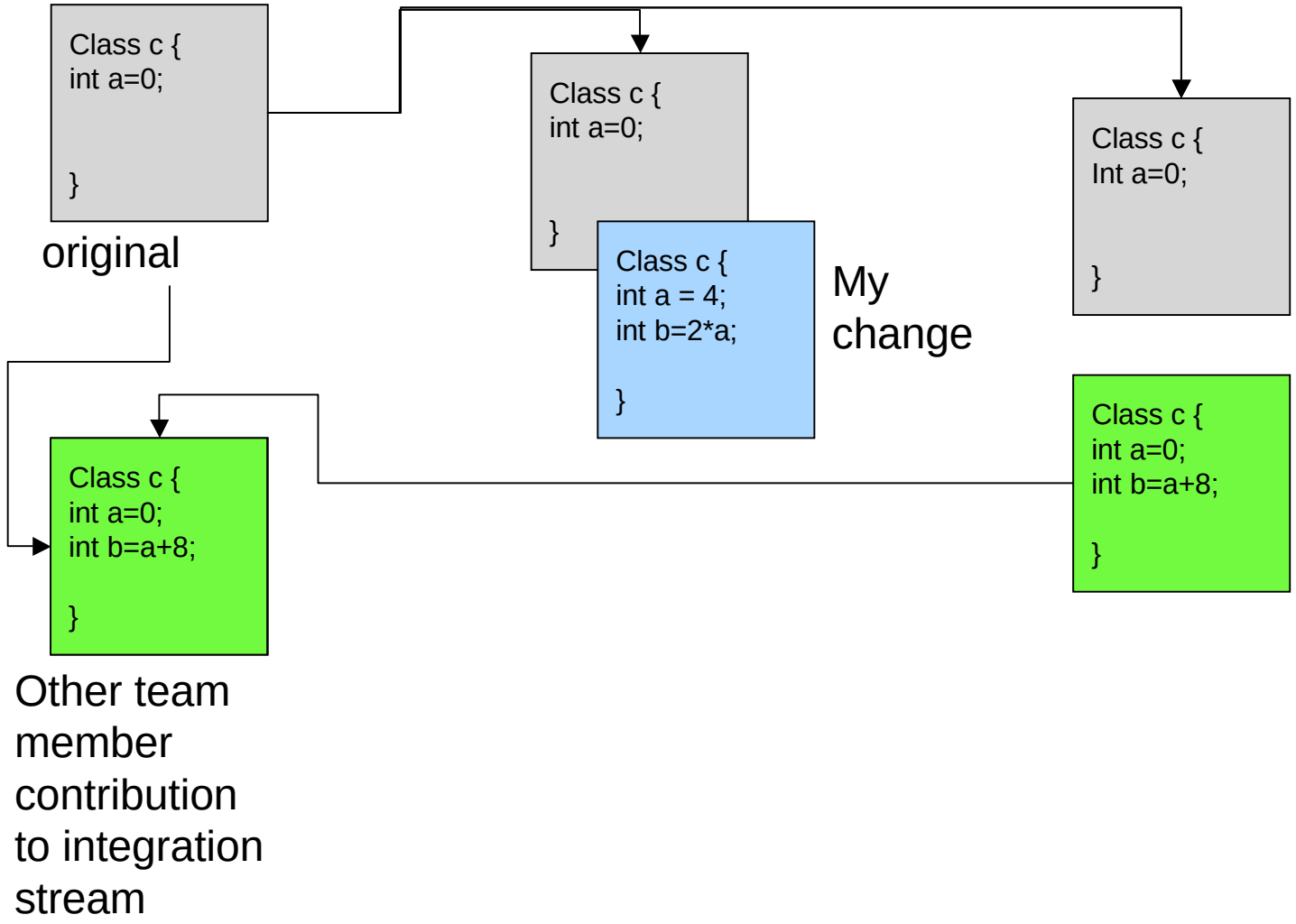
Base-aware Merge → 3-way-merge from hell



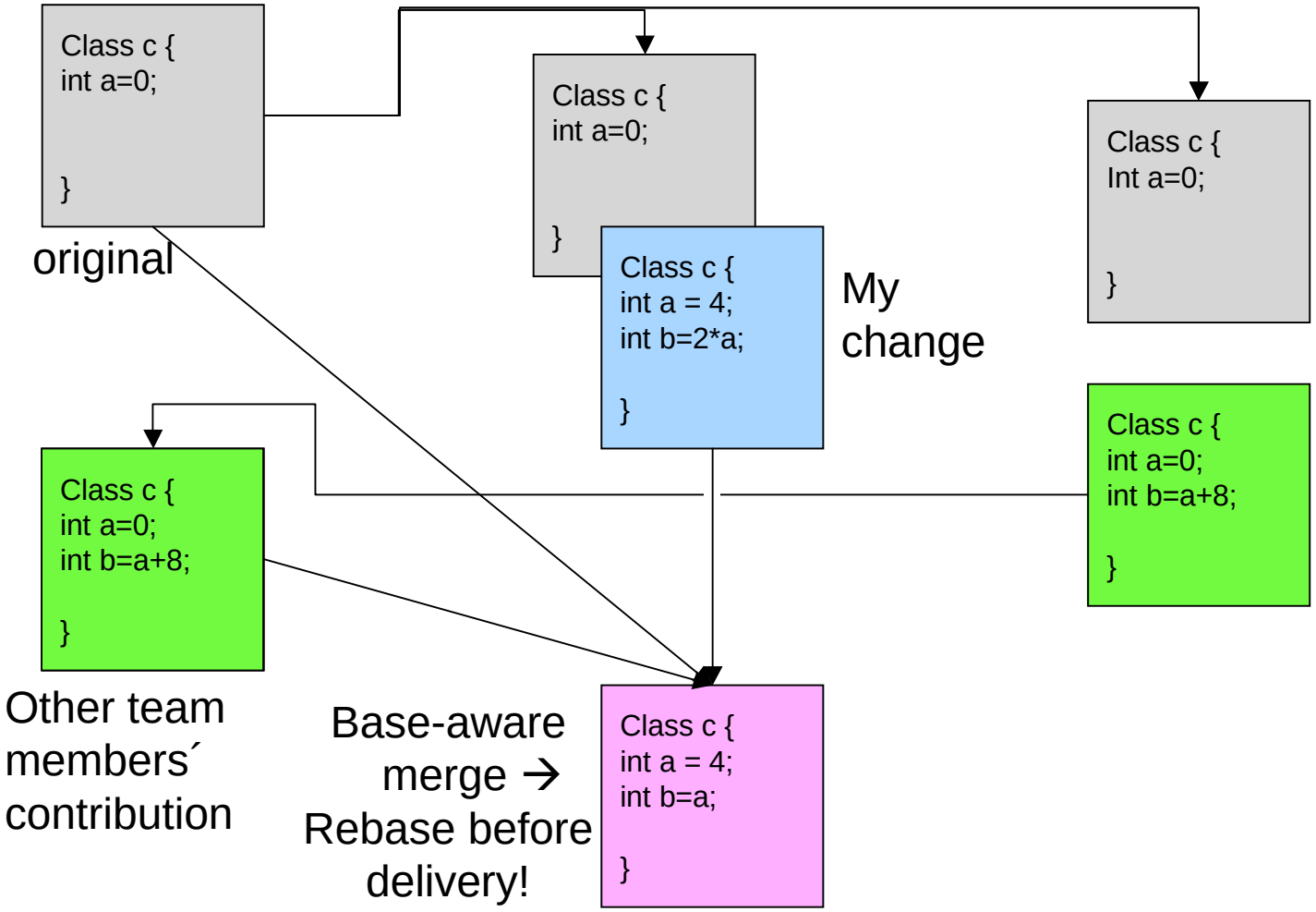
Base-aware Merge → 3-way-merge from hell



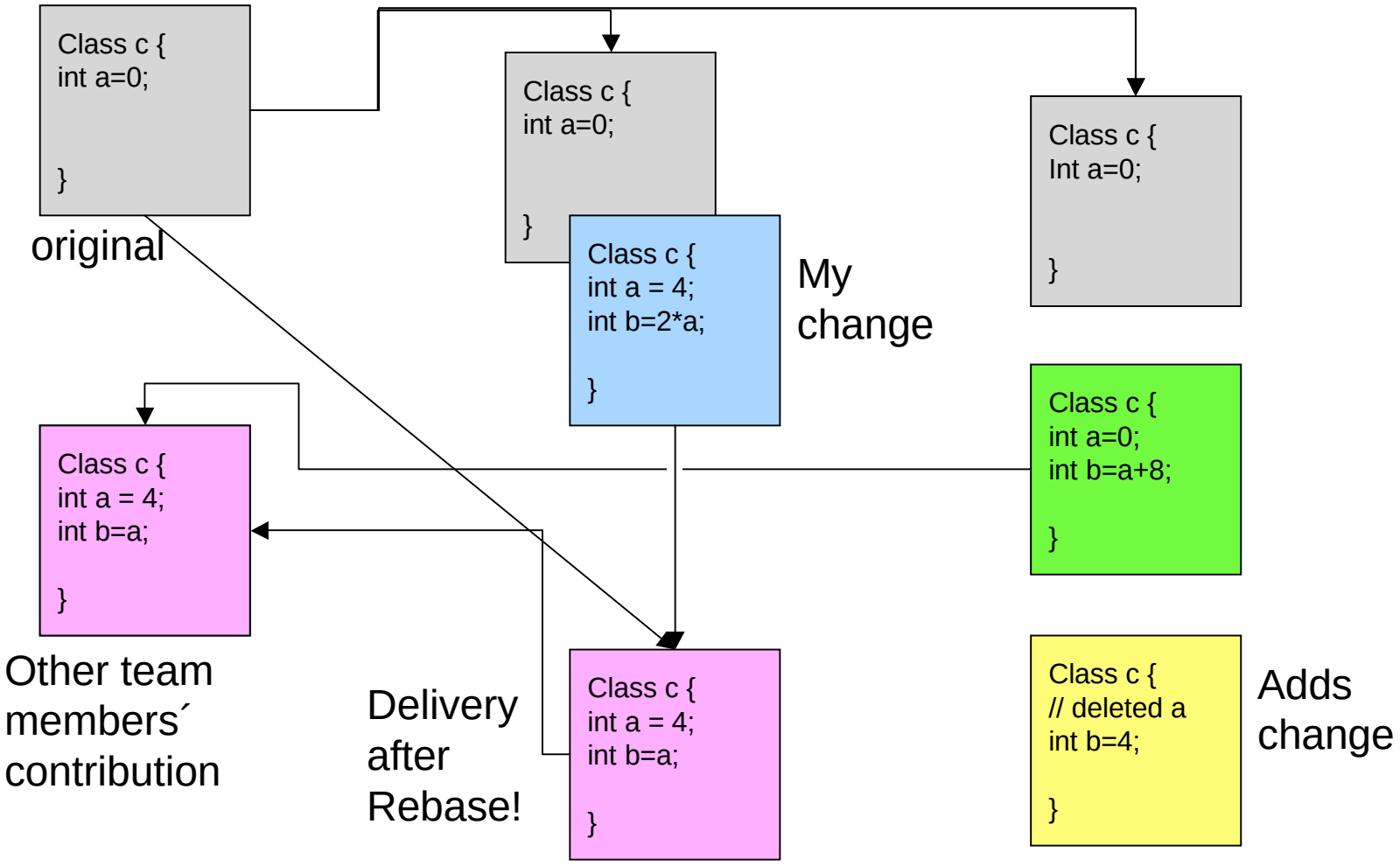
Base-aware Merge → 3-way-merge from hell



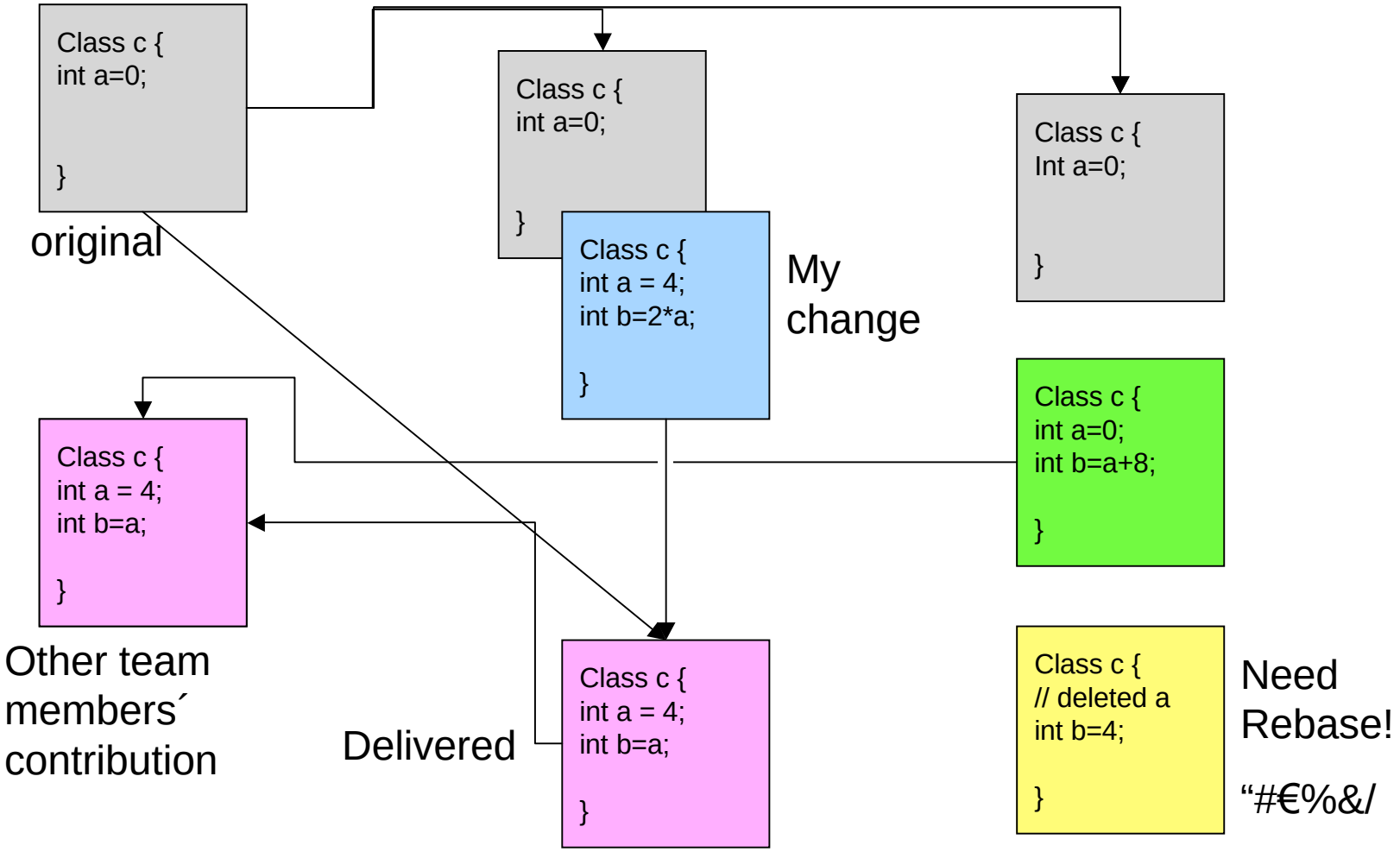
Base-aware Merge → 3-way-merge from hell



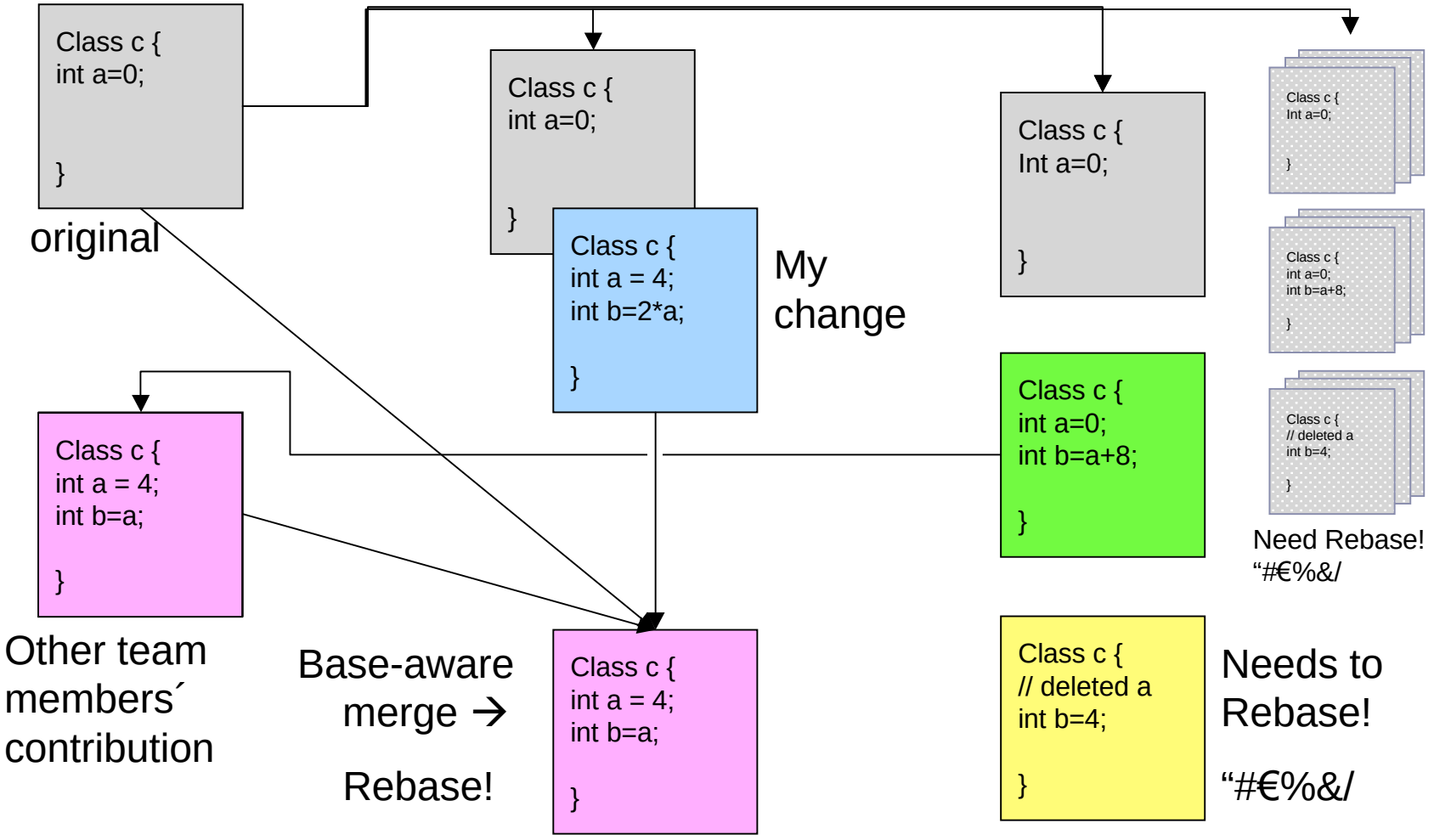
Base-aware Merge → 3-way-merge from hell



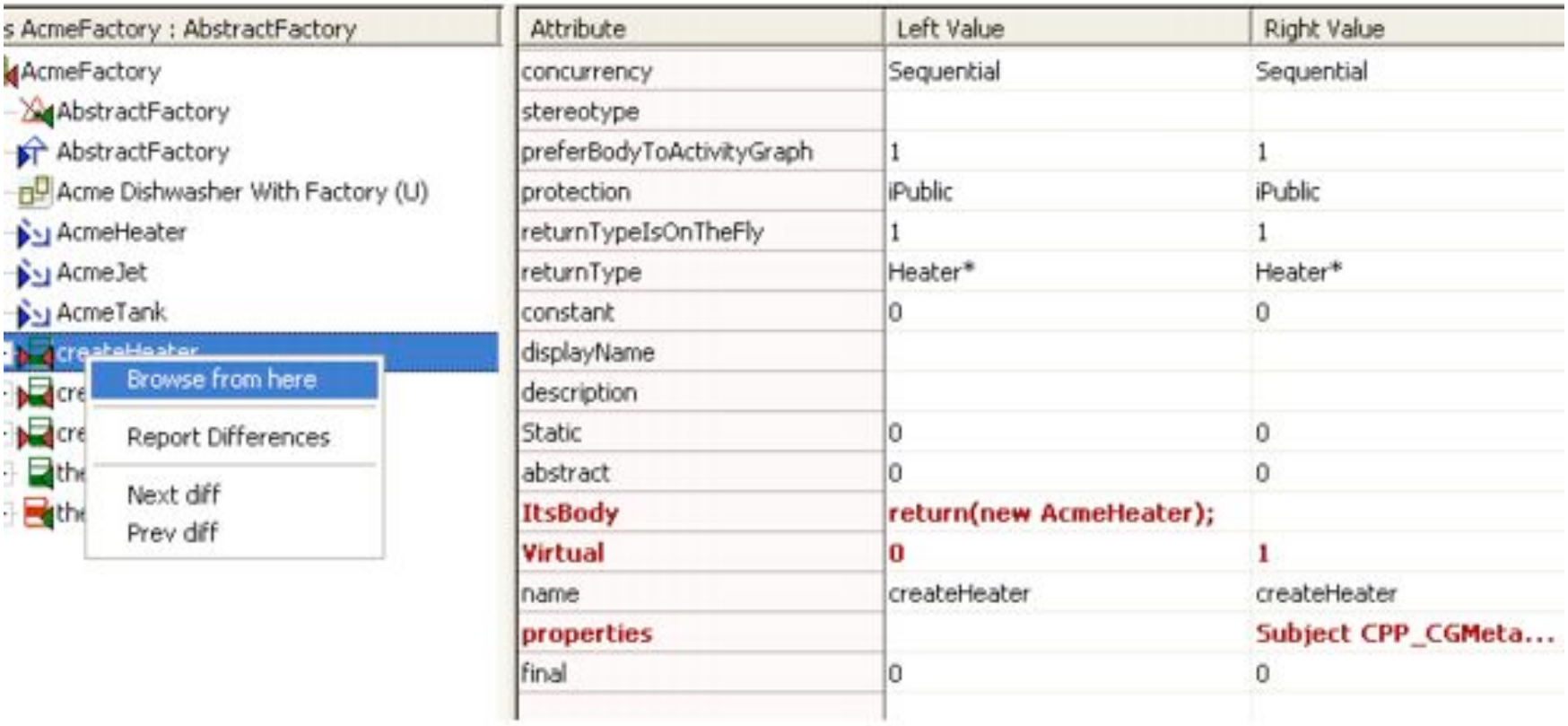
Base-aware Merge → 3-way-merge from hell



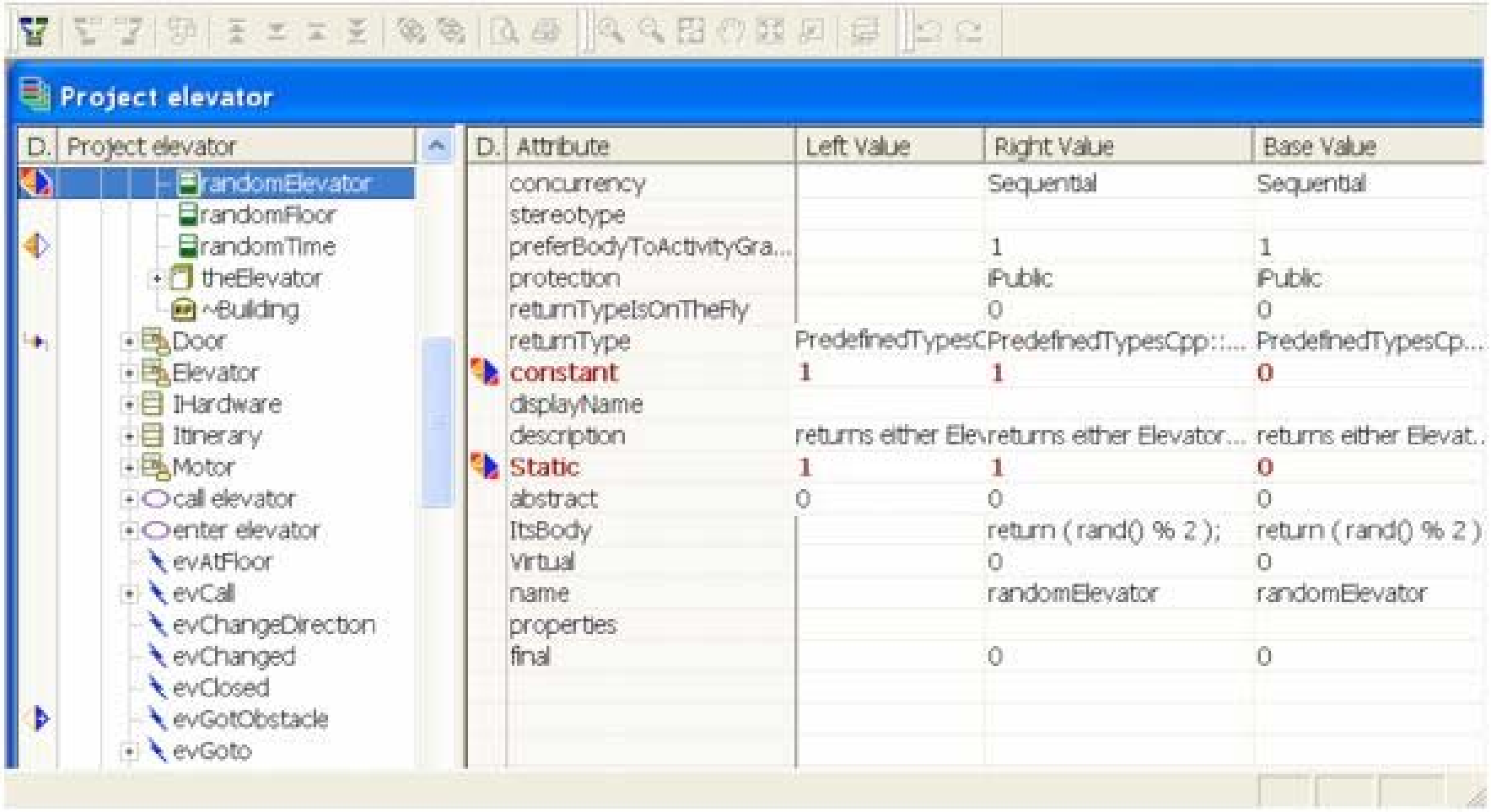
Base-aware Merge → 3-way-merge from hell



Two way merge…



Three way merge…



New possibilities using MDE extended with configuration management awareness

- Project management and increment planning
  - E.g the real-time picture of the model state is a true viable progress indicator
- Travel in time and space
- See transformation events, morphing software model
- Analysis of stinker code, and change prone software
- Analysis of robustness
- Analysis of cost of programming and modelling
  - Taboo?
- Tool for software productivity prediction and analysis
- Technology forecasting
- Re-design patterns, re-factoring, identification of anti-patterns

What is a software engineering

- **Software engineering** is the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software.[1] It encompasses techniques and procedures, often regulated by a software development process, with the purpose of improving the reliability and maintainability of software systems.[2] The effort is necessitated by the potential complexity of those systems, which may contain millions of lines of code.[3]
- A.k.a. transform a software system SSa into a better software system SSb, using understood requirements UR, past individual experiences IE, and individual software transformation skills (programming).
- A.k.a. Programming

Project Management in MDE

Big Bang… or “Extreme Makeover”…

Långsammare i början

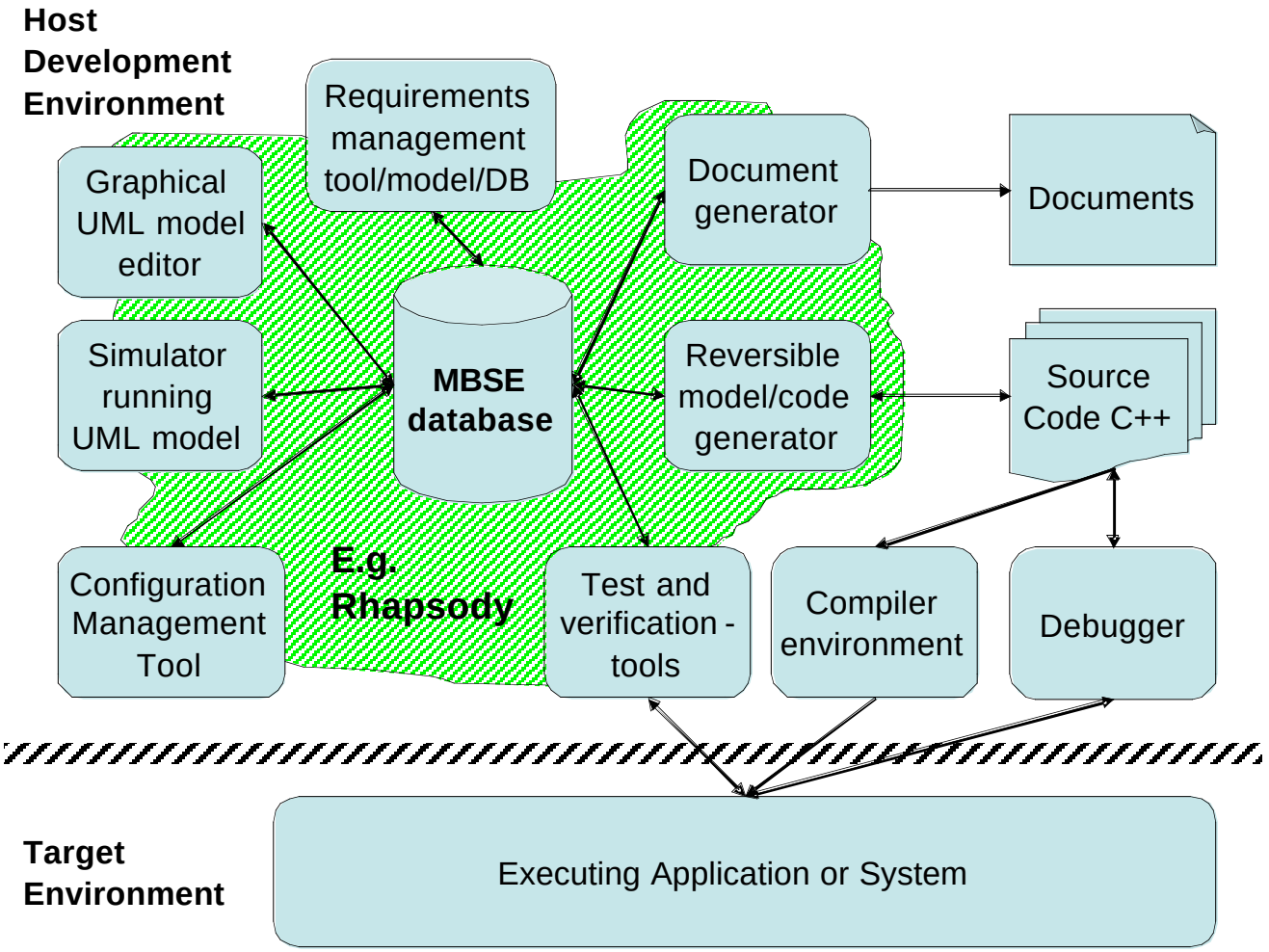
Nu ska man inte tro att modellbaserad utveckling omedelbart löser alla problem som finns i den traditionella utvecklingen och direkt leder till högre produktivitet. I början kan det rent av vara tvärtom.

Ett alternativ hade varit att bara jobba med C++ och det är kanske så att vi hade kommit snabbare fram i just det här projektet, men arbetet med modellbaserad utveckling måste ses som en långsiktig investering, säger Göran Calås ansvarig för införandet av modellbaserad utveckling på Saab Training System.

En anledning att det kan ta lite extra tid med modellbaserad utveckling i början är att utvecklarna ska lära sig nya arbetsmetoder och ett nytt verktyg. Det kan vara fruktansvärt frustrerande för erfararna utvecklare att upptäcka att de har svårt att åstadkomma de enklaste saker för att de inte är bekanta med verktyget. På samma gång kan det vara något som håller utvecklarnas intresse uppe för den nya metoden.

Det är ett väldigt intensivt lärande hela tiden och det är väldigt stimulerande för utvecklarna även om det kan vara frustrerande ibland, säger Niclas Vilsek.

Big bang theory – A case study



Big bang vs Nice introduction of MDE / MBSE

Big bang

- Quick
- Early pay off
- High initial cost
- Resistance is futile among team members
- Requires extensive training
- High pressure
- High risks → Quick return on investments
- People cannot be expected to understand potential problems that has never occurred, hence change may not be permanent.
- Requires a champion / mentor
- Requires teambuilding and team room

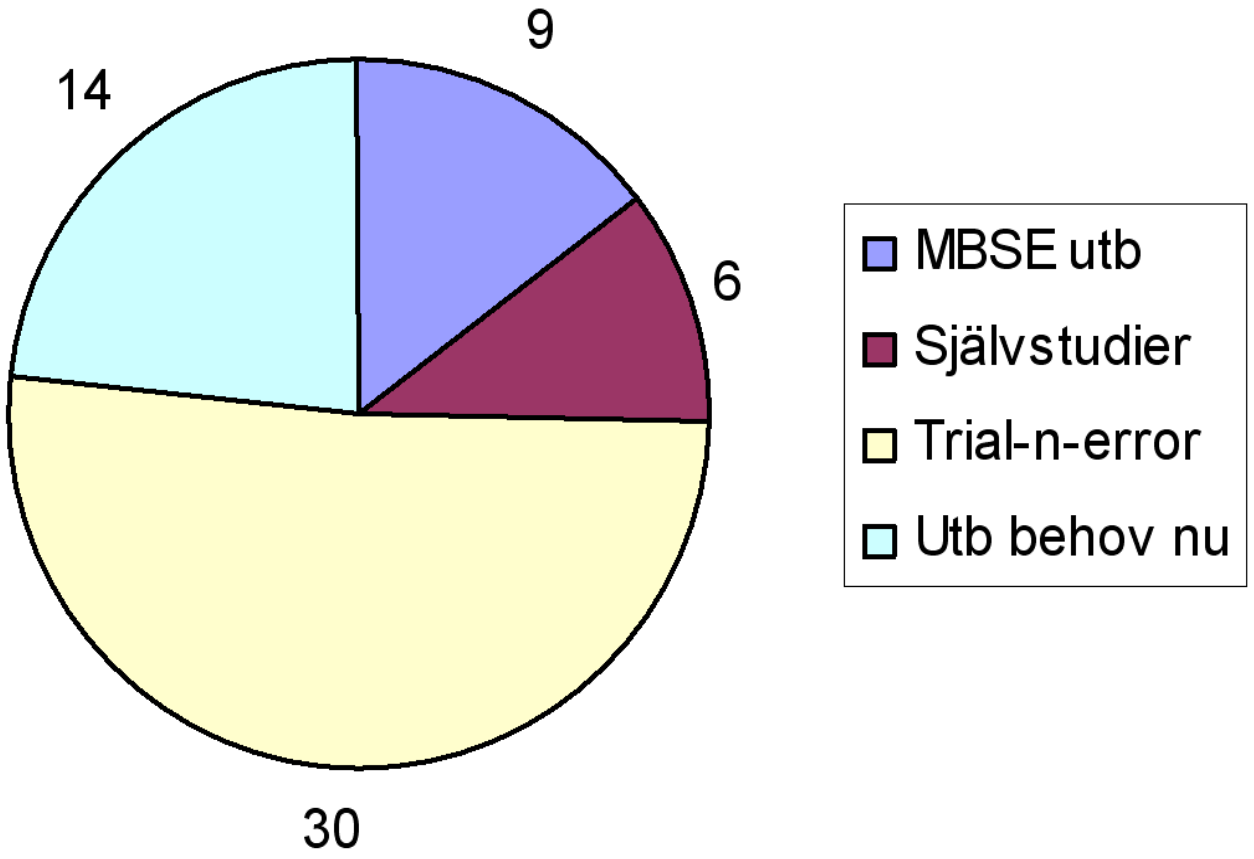
Nice

- Takes some time
- Late pay off
- Incremental investment
- Resistance may occur, but can be negotiated.
- Everyone has a fair chance to understand the motivation behind the changes.

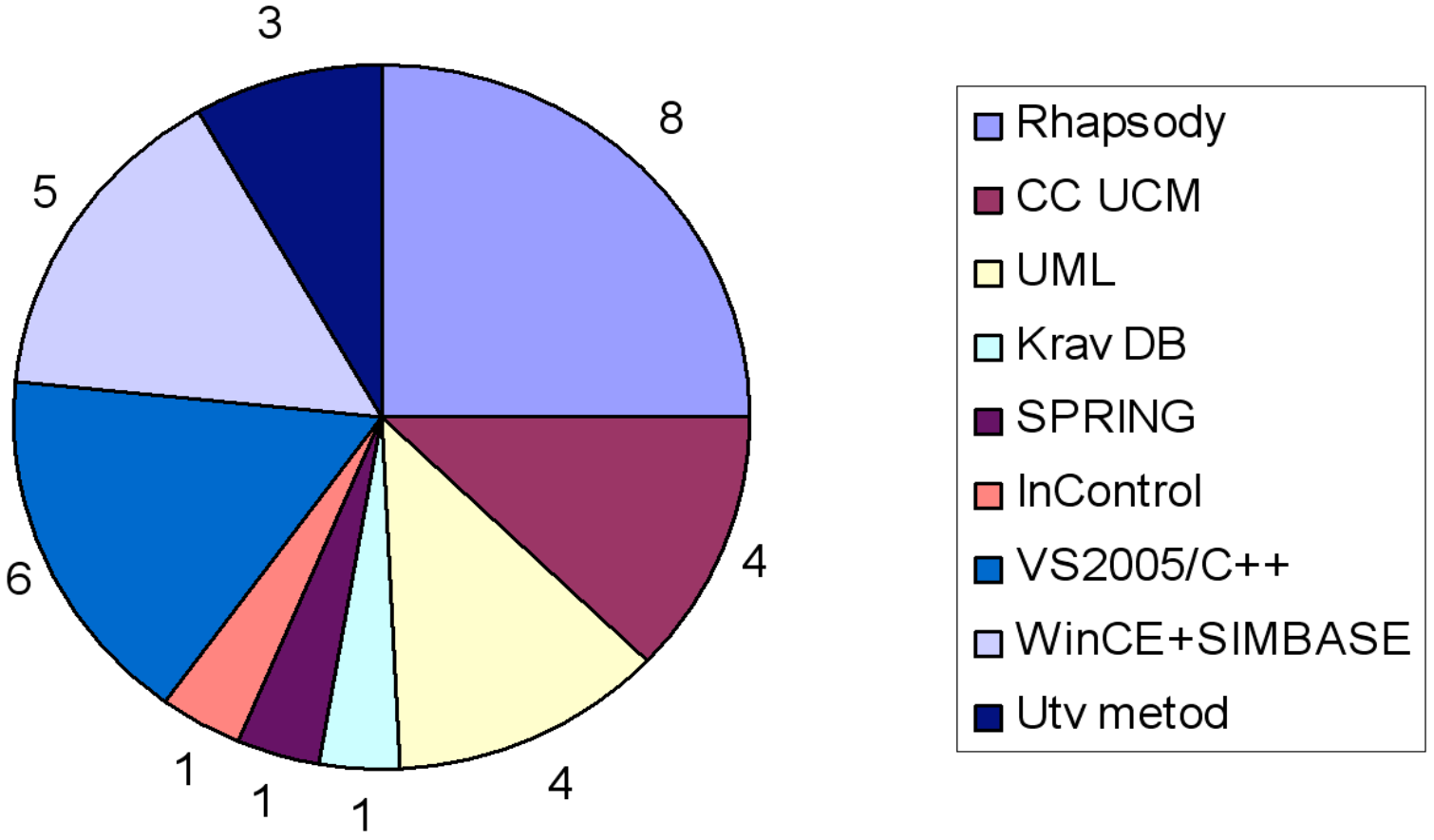
Do you believe that we (will) earn money using MBSE? (Asked after 18 months operation)

- In the future 50%
- Yes, now 20%
- No 20%
- Do not know 10%

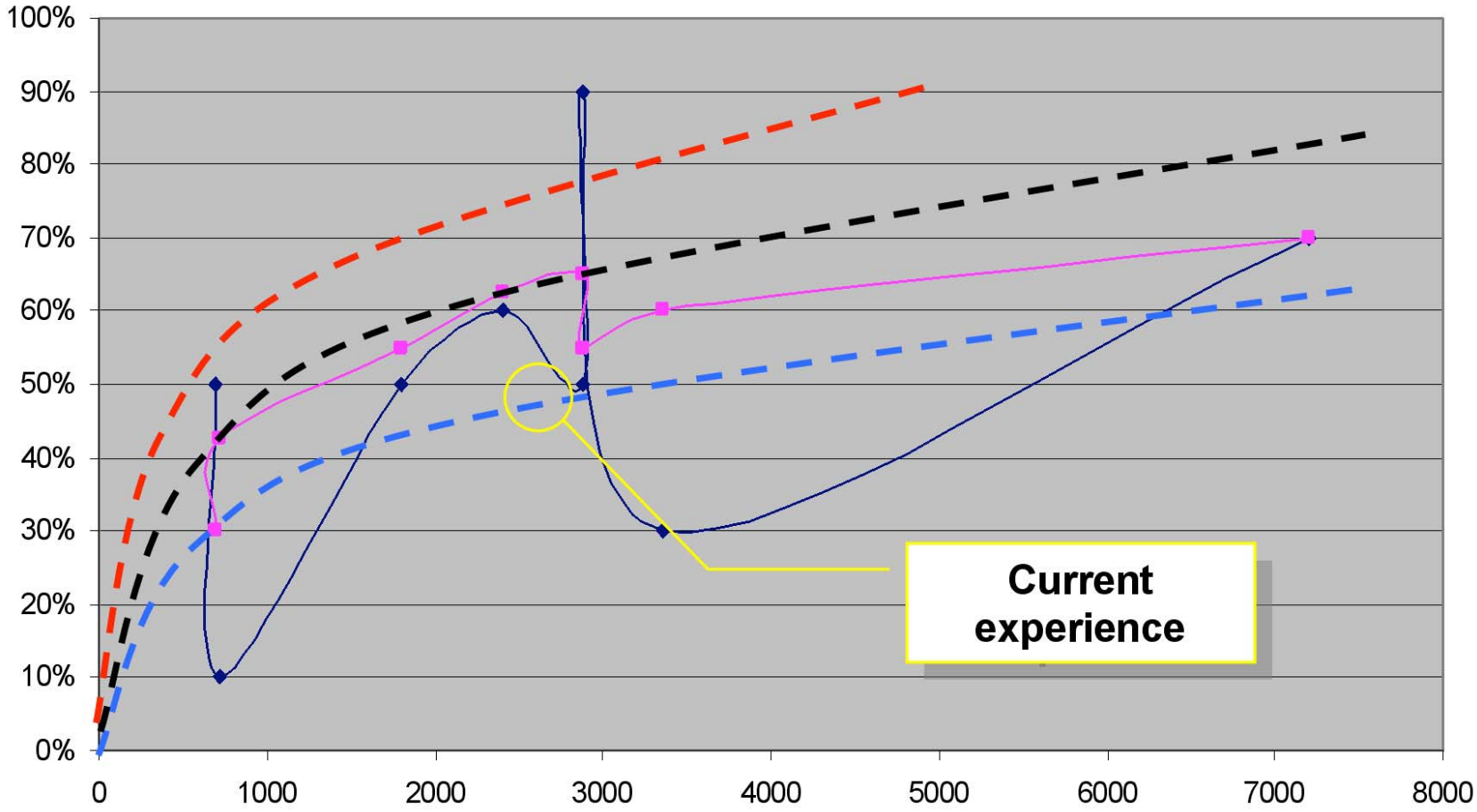
How did you got needed competence?



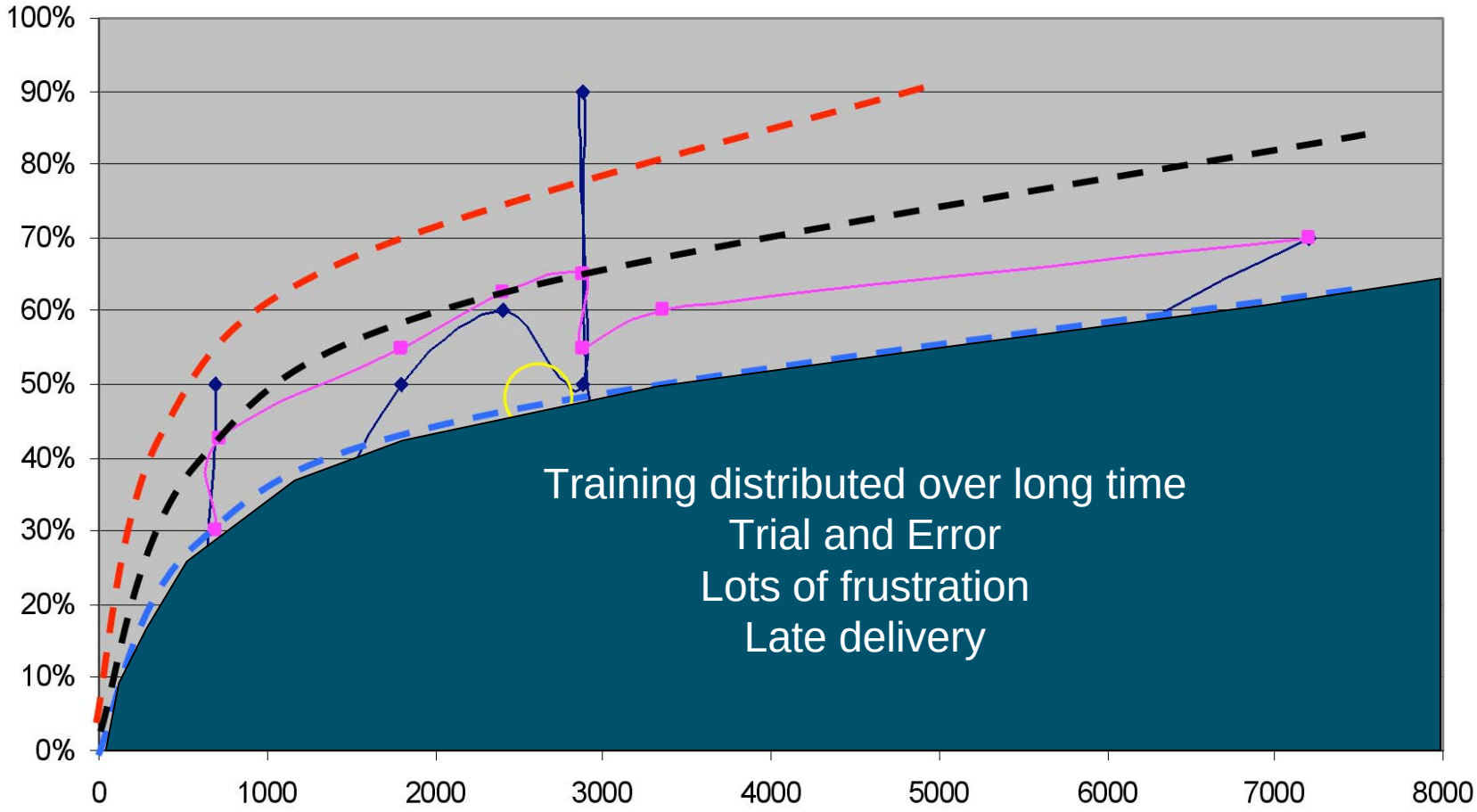
Estimated training needs for each project member
Less than 25% was provided by management
...extended with ”skunk” training



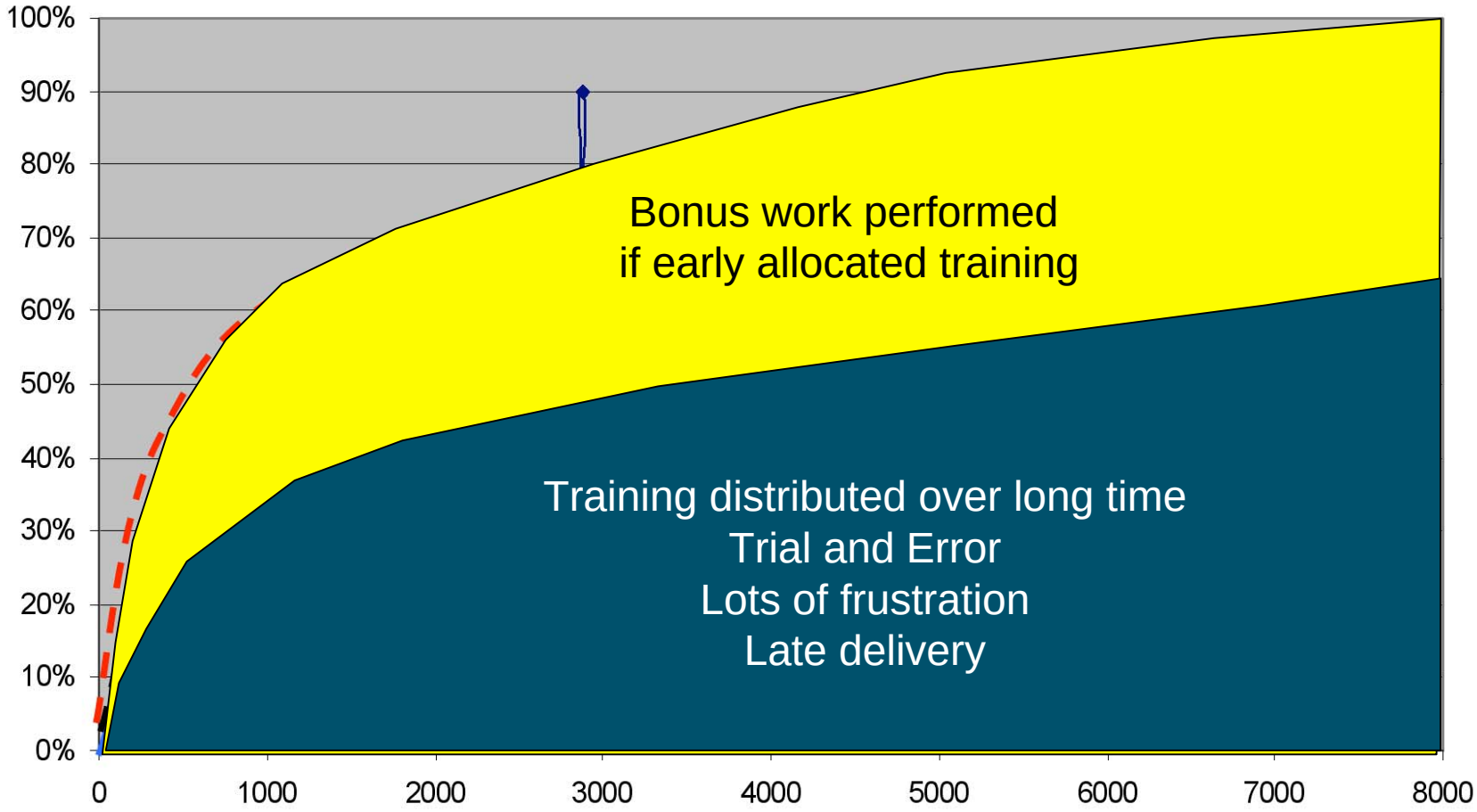
Individual perceived experience level, related to man hour spent using MDE tools



Impact of weak early training
– Area correpsonds to work performed



Impact of early early training – Area correpsonds to work performed



Lessons learnt

- Early training is key to early efficiency when introducing MBSE and MDE tool environments, in team projects with average complexity.
- A MDE champion/mentor with a training objective is essential for on the job learning. This is not a natural behavior for all consultants.
- The Project Manager needs essential skills in MDE, CM, and release planning.
- It is possible to introduce MDE and MBSE as a big bang effort.
  - Keep focus on essential aspects, thus avoiding non productive work.
    This is a difficult balance between autocracy and team involvement.
    As the team skill matures, team autonomy is to increase. Situational Leadership!
  - Provide as much as digestible, early and efficient, on-the job training. A key factor!
  - It is essential to keep everyone up-to-date with a master-plan for MDE introduction and project results.

SUMMARY

CM & PM in MDE

Summary

- CM gets more complicated in MDE projects, as more CM aspects are possible
- CM is still immature. CM modelling will be integrated in MDE UML modelling environment, making CM a part of the UML MOF.
- CM aware MDE brings valuable new capabilities.
- By introducing build and CM models, describing project activities in early; MDE becomes second nature for project managers.
- By managing PM models in synch with other MDE artifacts a live project status is known
- Big Bang MDE introduction is possible, risky, requires strong leadership, and early training
- Early training is key to early return on investments in Big Bang MDE
- MDE Project and Line Managers need MDE skills... Live status in the model!


SESAM
Nov 2008

Göran Calås

Saab Training Systems AB
SE-561 85 HUSKVARNA, SWEDEN
Phone: +46 768 967 167 , Mobile: +46 768 967167

Email: Goran.Calas@SAABGroup.Com