

# Integrated Requirements, Change and Configuration-Management for Bosch-Automotive

07.10.2008

by Ulrich Schopf, Stefan Joos; Robert Bosch GmbH



## Project Mission

### Task:

**Standardize Software -Processes, -Methods and -Tools to guarantee high efficient software exchange**

### Project Phases:

- |   |                 |
|---|-----------------|
| <b>1. Concept</b> development by Task-Force                       | Jan to Mar 2006 |
| <b>2. Implementation</b> (def. of rules and methods, development) | Dec 2008        |
| <b>3. Rollout</b> to the automotive business units projects       | Jan 2009        |

### Prerequisites:

- 1. No** evaluation of new tools, use existent tools.
- 2. No** envelope approach, focus on the essentials.
- 3. Use** best practices from all business units.



## Concept in a Nutshell

1. Component based SW-Development.
2. UBK wide SCM<sup>1)</sup> and REQM -Tool .
3. MSR<sup>3)</sup> for SW-Component Description (Documentation, Data specification, SW-Architecture).
4. AUTOSAR-Conformity is ensured by conversion
5. UBK wide standardised
  - modelling- and programming-guidelines,
  - naming conventions,
  - variant concept...
6. Integrated Development Environment (IDE) as universal development environment for manual and model based code generation (ASCET)

1)SCM & REQM = Software Configuration Management & Requirements Management 3) MSR = Manufacturer Supplier Relationship, standardised in ASAM



**BOSCH**

## Requirements Engineering in V-Model Context

### → Requirements Management (REQM)

Management and tracing of issues (requirements, defects, change requests)

### → Requirements Development (RD)

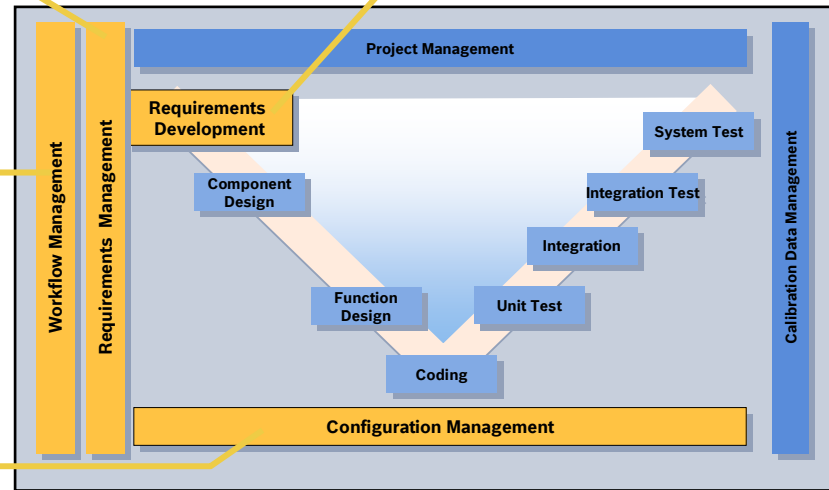
- Structured development of requirements,
- Manage reuse of requirements

### → Workflow Management (WFM)

Supports processes through implemented workflows

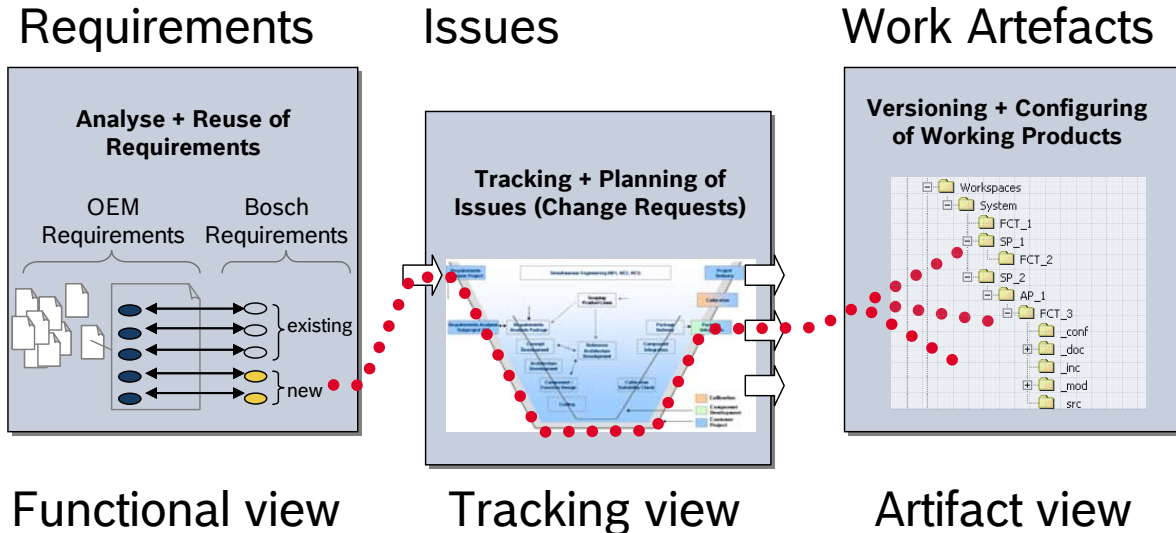
### → Software Configuration Mgmt (SCM)

Storage of results



## Key Aspects of the integrated method

- In an **integrated REQM-, RD- and SCM method**, requirements, issues and artifacts are managed in a single task.
- **Forward Tracing**
- **Backward Tracing**



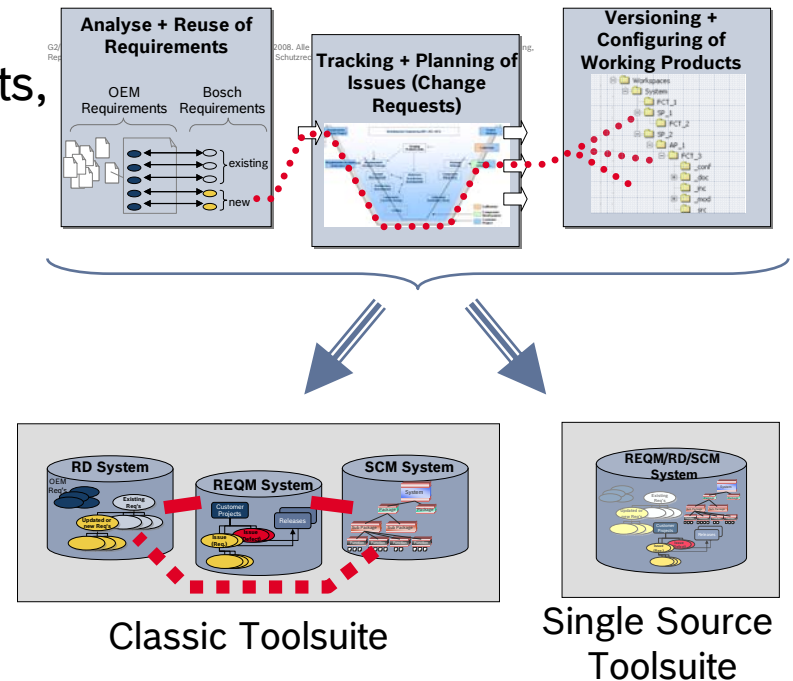
## Classic - versus Single Source-Tool Suites

### → “Classic” Tool Suite

- References between Requirements, issues, artefacts via different data models
- Usage of references are complex, high maintenance costs, risk of inconsistency

### → Single Source Tool Suite

- References are implemented with one single data model
  - easy to maintain, more data consistency
- ⇒ more flexibility to enhance and maintain



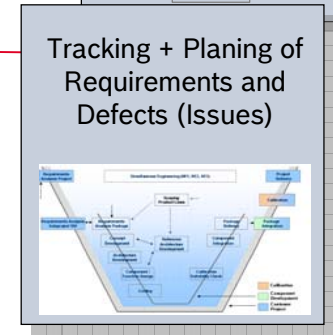
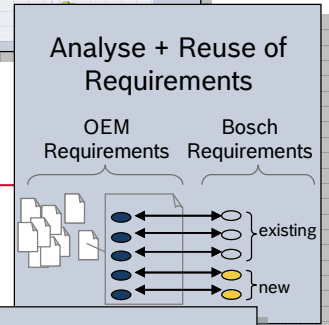
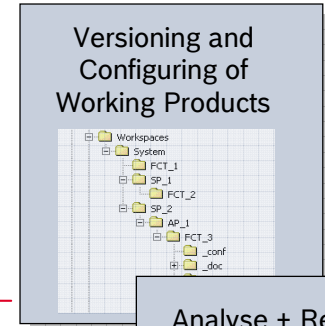
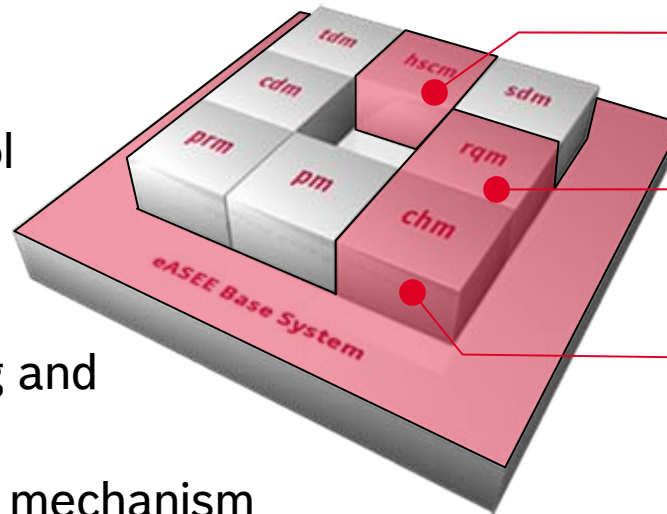
## Single Source Tool Suite Realization with eASEE modules

### → Necessary Modules

- hscm → SCM tool
- rqm → RD tool
- chm → REQM tool
- base system

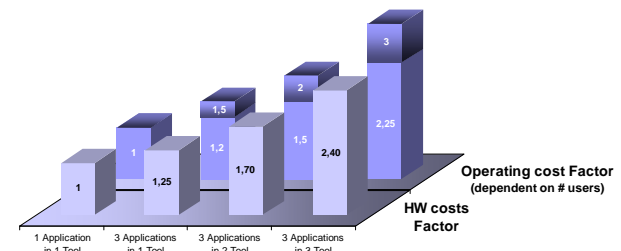
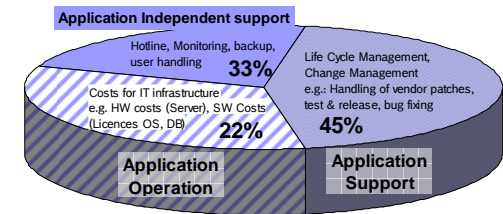
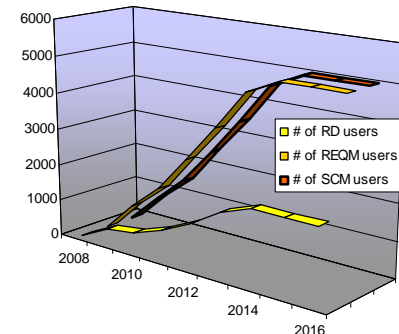
### → Benefits

- common versioning and base lining
- same query/search mechanism
- common reporting
- same look & feel
- common user management



## Classic - versus Single Source-Tool Suites Cost Model Prerequisites:

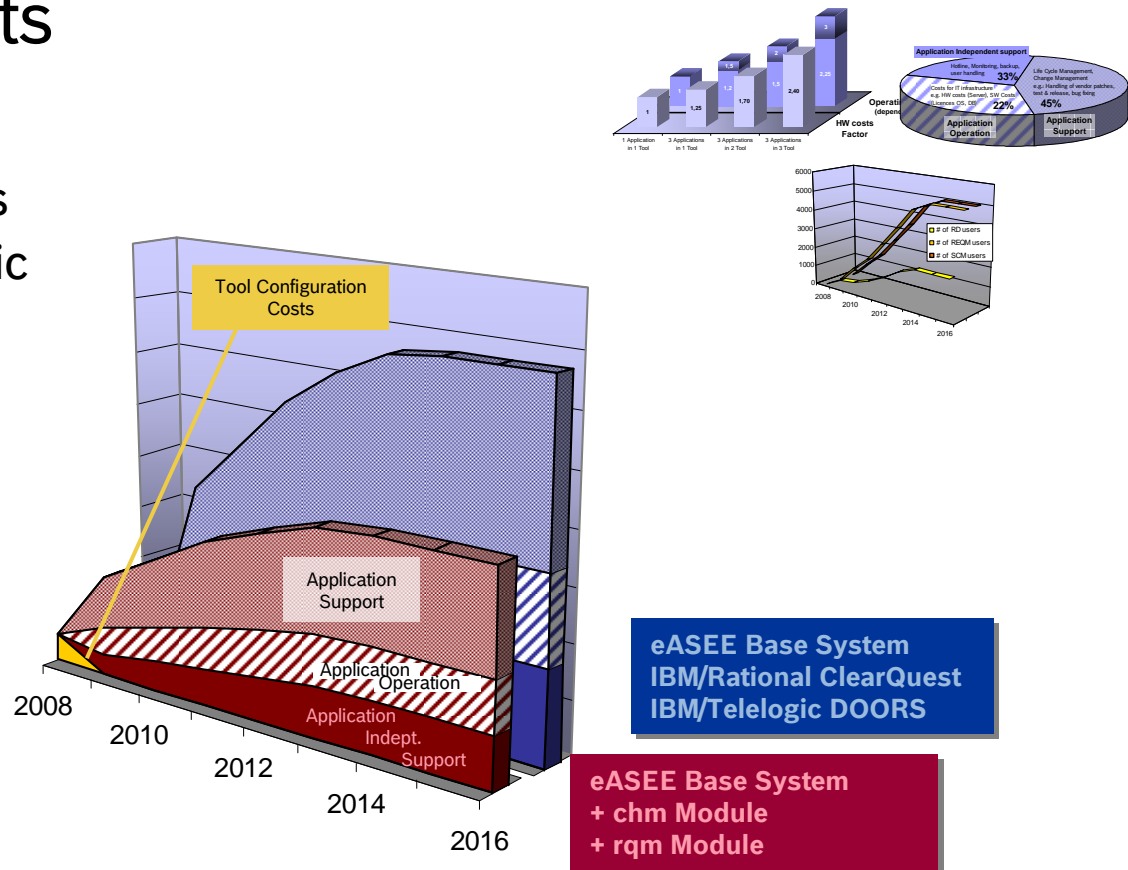
- User Ramp Up Model up to 5000 users
- Application Operation (Hardware)
- Application dependent (User support, updating)
- Application independent costs (Hotline, monitoring, backup, user handling)
- Tool Operating Factors  
1 Tool, 2 Tools, 3 Tools for 3 Applications



## Classic - versus Single Source-Tool Suites Operating Costs

→ Single source tool suite approach less than half to classic tool suite approach

→ No significant impact of tool configuration



## Resume

- An integrated method for Requirements-, Change- and Configuration-Management is needed, because it
  - manages requirements, issues and artifacts in a single task (single source improves data quality)
  - allows process analysis like reuse metrics
  - supports development efficiency
- “Single Source” Tool Suites like eASEE supports the integrated method “Classic” Tool Suites not.
- Market situation is mostly with Classic Tool Suite approach
- Bosch Automotive decided to use the eASEE Tool Suite

