Win-Win with Embedded Software Components as a Product

Joachim Langenwalter, Mentor Graphics GmbH

Dr. Emmerich Fuchs, DECOMSYS GmbH

Stefan Duda, Elektrobit

Markus Maier, ETAS GmbH

Dr. Helmut Schelling, Vector Informatik GmbH

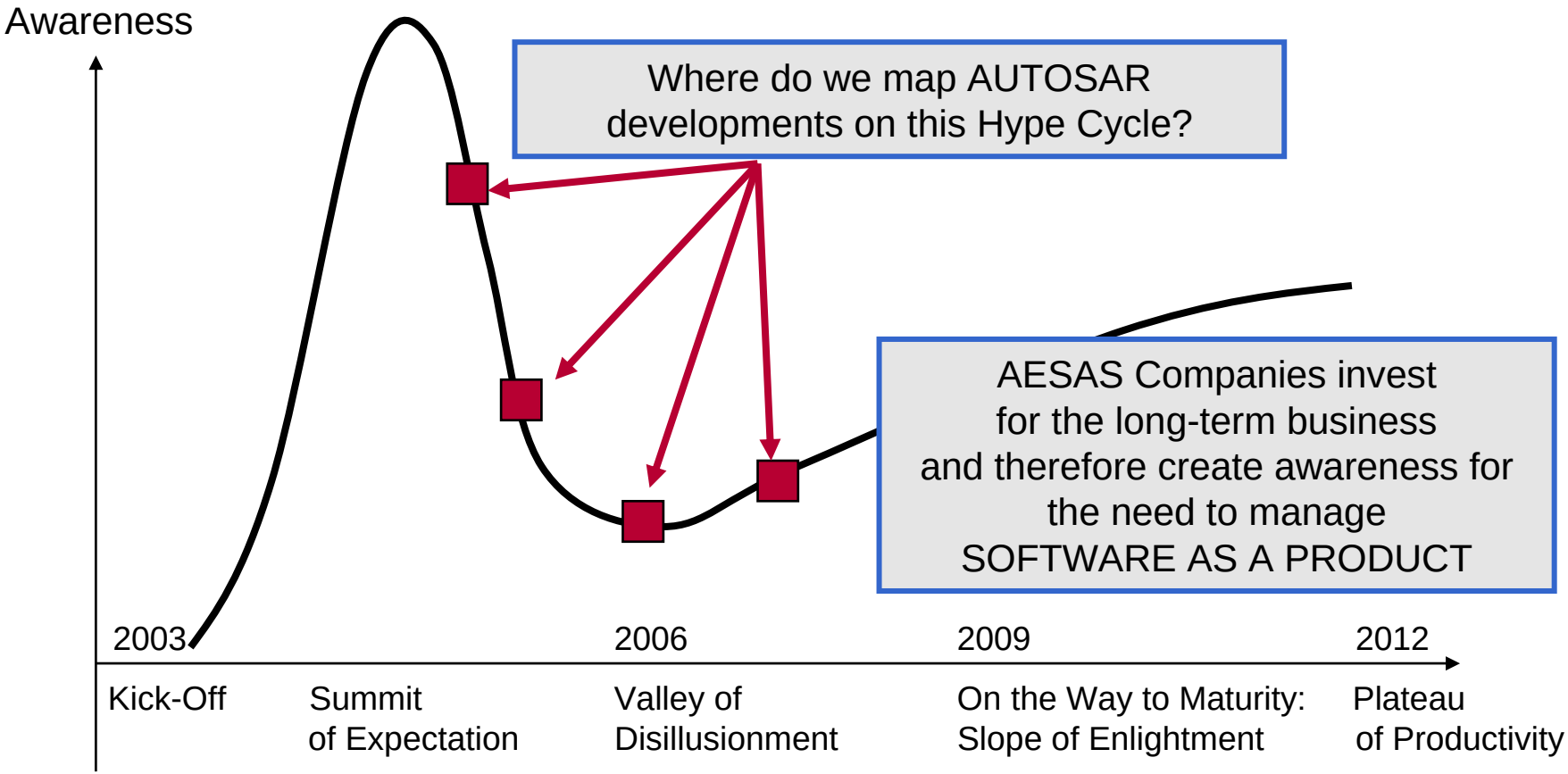
Dr. Michael Siedentop, Softing AG

Georg Kopetz, TTTech Computertechnik AG

Dr. Meinhard Erben, AESAS

AESAS
Association of European Suppliers for Automotive Software
at the AUTOSAR Premium Member
Conference in Brussels, December 12th, 2006

HypeCycle of New Technologies and AUTOSAR™



HypeCylce by Gartner Group

Agenda

- Update on AESAS
- Changes for software companies with AUTOSAR
- Embedded software as a product
- Source code vs. object code
- Summary and next steps

AESAS Summary

Association of European Suppliers for Automotive Software

- AESAS stands for **A**ssociation of **E**uropean **S**uppliers for **A**utomotive **S**oftware
- AESAS is an industry association of medium-sized companies that develop and distribute software and related services to the automotive industry

AESAS Members Update

Association of European Suppliers for Automotive Software

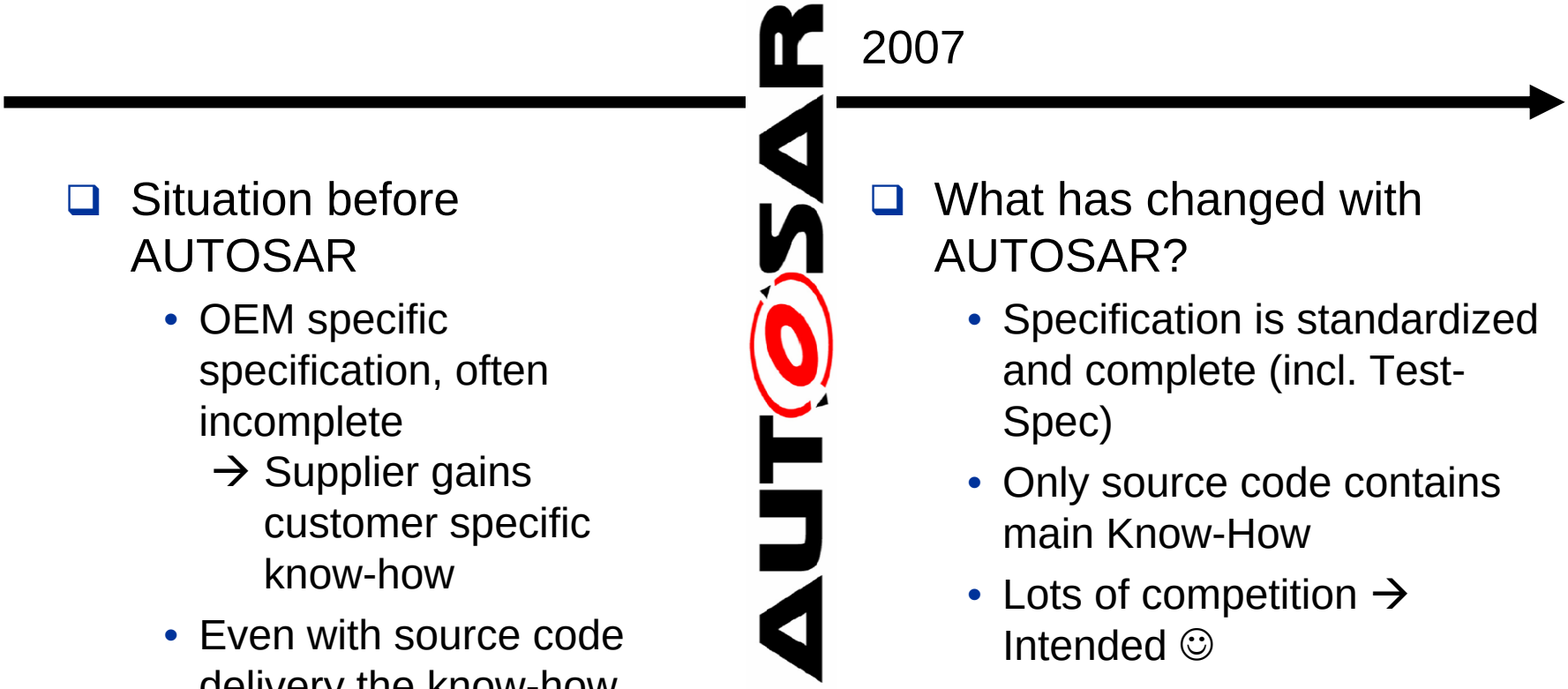
Working Groups and Activities

- Business Models
  - Assessment of current business models
  - Issues with current models
  - Recommendations for business models
  - Ready in H1 2007 to present results
- Source Code
  - Content of this presentation
- Image
  - Create awareness for the values provided by automotive software companies beyond the program code.

Agenda

- Update on AESAS
- Changes for software companies with AUTOSAR
- Embedded software as a product
- Source code vs. object code
- Summary and next steps

Why now ?



- Situation before AUTOSAR
  - OEM specific specification, often incomplete
    → Supplier gains customer specific know-how
  - Even with source code delivery the know-how advantage was maintained

- What has changed with AUTOSAR?
  - Specification is standardized and complete (incl. Test-Spec)
  - Only source code contains main Know-How
  - Lots of competition → Intended ☺
  - Big Specification requires major investments

Agenda

- Update on AESAS
- Changes for software companies with AUTOSAR
- Embedded software as a product
- Source code vs. object code
- Summary and next steps

Properties of Automotive Embedded SW

- Functional Requirements / Properties
  - Embedded software. The software is an integral component of an electronic system (close proximity to hardware).
  - Distributed software. Logical software entity is distributed over several ECUs.
  - Great demands on safety. Verification of functional safety is a prerequisite for type approval for use in public traffic.
  - Great demands on real-time properties. The required time span between the occurrence of an external event and its treatment in the ECU is very short
  - High demands on availability. A vehicle function is not permitted to fail.
  - High cost pressure on hardware. Great demands on software optimization.

Source: “Prototyping”, M. Maier, Nikkei Automotive Technology Days 2005, Tokio

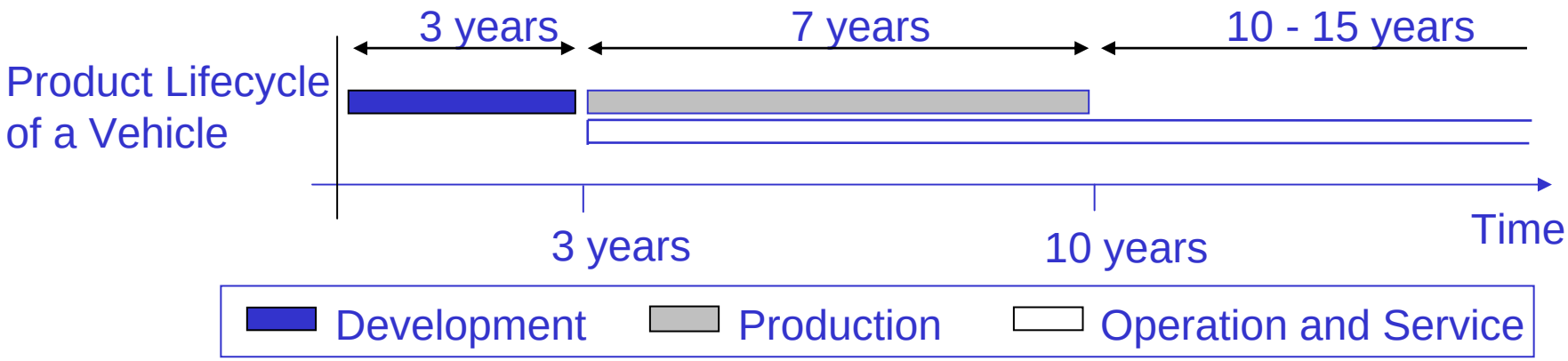
Properties of Automotive Embedded SW

- Non-Functional Requirements / Properties
  - Long-Life Cycle. Due to the nature of the Automotive Business, the Embedded Software must remain active (in maintenance) for a long period of time. Typically from start of development over production to service (10 to 15 years)

Source: “Prototyping”, M. Maier, Nikkei Automotive Technology Days 2005, Tokio

Lifecycle of Automotive Embedded SW

Source: “Automotive Software Engineering, Dr. Th. Zurawka, Jörg Schäuffele



Additional Challenge: Complex Development Partnerships

- OEM, Tier-1, Tier-2 and BSW Module Supplier need to be integrated
  - Different ‘modes’ of operation are common in the Automotive Industry
  - Each Development Partner has unique Know-How that contributes to the overall success of the project

Automotive Embedded SW as a Product

Source: “Automotive Software Engineering, Dr. Th. Zurawka, Jörg Schäuffele

- Result: Automotive Embedded Software is a Product
  - Applies to Basic Software Modules especially
  - Development, Timing, Budget, Conformance and **Service** must be guaranteed over the life cycle
- Win-Win with Automotive Embedded Software
  - In order to address the unique challenges of Automotive Embedded Software, a win-win situation for all development partners is the key for success

→ Business Models of Development Partners need to ensure the Win-Win situation in order to guarantee the long-term success

Different Types of License Models

Examples for License Models:

- Service Contract
- Contract for work and labor
- Buy-Out
- License agreement (Usage rights)
- Mix of above

- AESAS defines, based on the unique experience of its members, recommendations for business models
  - Focus is to ensure the long-term success of Automotive Embedded Software and AUTOSAR

Type of Contract

Practically all usual types of contracts:

- Service Contract
- Contract for work and labor
- Buy-Out
- License agreement (Usage rights)
- Mix of above

Agenda

- Update on AESAS
- Changes for software companies with AUTOSAR
- Embedded software as a product
- Source code vs. object code
- Summary and next steps

Customer Statements (I)

- “We need source code (SC) for review, to judge the quality”
- “We need SC to debug (in laboratory and field test) to analyze problems”
- “OEM: We want the Tier to take the responsibility (liability) for one ECU and therefore we want to give him access to SC”
- “Why changing now if we always got source code?”
- “We want the buyout to develop and maintain the code further”
- “We need SC for fast and cost efficient porting”

Customer Statements(II)

- Tier 1 and Software Supplier as an integrator:
  - “Our (T1 customer) concern is with seeing the SC, that the originator may claim, we violated the copyright with our own implementation (this would require a clean room implementation)”
  - “The integrator of the AUTOSAR validator (seeing SC) agreed not to create BSWC within the next years”
- “Ensures maintenance and support in case of relationship problems with supplier (insolvence, etc.)”
- “SC in connection with configuration enables more efficient solutions (RAM, ROM, Speed)”
- “In case of safety critical applications we prefer object code (OC)”
- Tier 1: “We give our developers for the drivers only object code, so they cannot modify the SC”

Product Updates

How to do product updates with source code delivery?

- Will the SW-modules (only) be maintained by the customer?
- Will the SW-vendor get feedback about improvements or bug fixes etc.?
- How will the changes of customer and vendor be synchronized?

Liability and Warranty

- If the BSWC supplier delivers the software in source code, the need for warranty and liability restrictions for the BSWC supplier is even much higher: Since the OEM is then able to modify and/or extend the software, he may create bugs in the software for which the BSWC must not be held liable.
- The warranty obligations of the BSWC supplier must be limited to the software provided by the BSWC supplier to the OEM.

Copyright and Property Rights

Three legal areas to consider:

- Copyright (object and source code)
- IP protection (especially source code)
- Patent rights

How to protect the immaterial rights with source code?

- Act in good faith (only with NDA)
- Escrow
- Digital signature
- Generated source (know-how embedded in generator tool)

IP Protection

Different levels of potential damages:

- Medium (User):
  - Development department at OEM, Tier1 or Tier2
- High (Competitor):
  - Integrator/competitor who is/has developing /-ed the same modules
  - T1/T2-department, who is developing the same modules
- Very High (Competitor):
  - Companies in areas with limited attention to copyrights

Possible Solutions

Recommendations to limit the risk:

- Deliver only object code
- Turn around principle
  - Not standard to deliver source code
  - In case of SC delivery only with limitation of user community (e.g. company name or department)
- Source code for review only (OEM and ECU supplier)
- SC for OEM only (no right to pass on further)
- SC for T1 (no right to pass on and specification of user community within company)
- SC to integration company (Competitor with specific source code NDA between vendor and integrator)

Compatibility

- Software interface specifications and certification:
  - AESAS-members would like to certify the compatibility of the different modules (within AUTOSAR and outside)
- Development of AESAS recommended practices including appropriate certifications:
  - AESAS would like to establish recommendations for the development with high quality and introduce a quality standard to guarantee a high quality level

Win-Win with Software

- User (OEM, T1)
  - Standard BSW enable vendor (IC, T1 and SW vendor) independent implementation of customer feature
  - “Commodity-zation” of BSW
- SW Vendor
  - Has an interest in IP protection to ensure long term business, which leads to long term support of the customer
  - Differentiation through design and add-on tools

Next steps

- Show possible recommendations to establish a successful business in automotive software
- Solve contradiction: Object code only vs. source code
- Establish a common vocabulary of concepts/terms between members and customers
- Establish „AESAS-Compliance“ quality argument/level

→ **To join AESAS please send email to contact@aesas.org**