FAST TOPHOO COOKING WITH PAPYRUS
POLARSYS DAY
Avionics software engineering @ AIRBUS

• Avionics SW Dept
• Safety critical: constrained by DO178/ED12
• Software architecture representation
• Graphical/textual model
• Using subsets of
  • UML for 10 years
  • HOOD for 30 years
• But HOOD tools are obsolescent and UML is too complex
• Idea: deploy a new tool implementing HOOD subset based on UML infrastructure
• Target audience: developers and architects
Tophoo

- HOOD subset as UML profile
- Hierarchical and graphical
- Requirement management
- Team work
- Seamless integration to in-house Version Control System
- Model verification
  - Dozens of design rules
  - DO178/330 qualification
- Generation of architecture document
  - Reference for DO178 certification
- Generation of code skeleton (C, ASM)
  - Shall support incremental generation
TOPHOOO
System engineering @ AIRBUS

• Systems design office @ AIRBUS

• Graphical models introduced in the last decade
  • Vitech CORE for operational scenarii
  • SysML for functional breakdown and simulation

• But
  • Vitech CORE uses a proprietary data format making interoperability problematic
  • SysML is too complex and its concepts not matching system engineers usages (Block instead of Functions, etc.).

• Why not reapplying the recipe of TOPHOO?
• Target audience: system engineers
FAST

- SART-like concepts as SysML profile
- Hierarchical and graphical
- Team work
- Seamless integration with usual Version Control Systems (subversion, git)
- Model verification
- Intuitive interface
- Highly customizable
TOPHOO/FAST architecture
TOPHOO/FAST architecture
TOPHOO/FAST architecture

Model Explorer  Policies  Property View  Diagrams & palettes

Customization  Customization  Customization  Customization
Facet  GMF  XWT  - UML component diagram

VCS Module  Acceleo Templates  Gendoc Template
Airbus VCS  Acceleo  Gendoc
Next technical advances of Papyrus

Outline of planned incoming works (2013-2014)

- Follow-up of stability and correction effort.
- Follow-up of documentation and ergonomy improvement
- Improve internal practices applying MDE paradigms for designing Papyrus
  - Model-based management of the Papyrus architecture.
  - Automatic model-driven generation of unitary and functional tests.
- Conformity to OMG standards (to most up-to-date versions)
  - UML2.5, ALF, fUML, MARTE 1.2.
- Improve CDO and Xtext integration
- Improve Alf and OCL support
- New features:
  - Initial support for ISO42010 (Architectural Framework Designer: View&Viewpoint)
  - Support for layers within Diagram
  - Customization features (cont’d)

Papyrus 1.0 scheduled for Eclipse Luna (07/2014)
Papyrus 1.0 Customizations: Advanced stylesheet support
Papyrus 1.0 Customizations: Generalization of the edition tools

- Model-based definition of the palette elements
- Custom creation tools shared between palette and model explorer
- Runtime definition and deployment of palettes, menus
Papyrus 1.0 Customizations: Generalization of the edition tools
To summarize...

- Strengths and weakness of the approach
  - + Both tools well accepted by the users
  - + UML/SysML compatibility and standard data format
  - + Much more efficient than GMF based development
    - Specific Code base / std code base between 2% and 5%
  - + Good trade-off regarding Openness, flexibility and reuse
  - - Dependency to not completely stabilized Papyrus API on branch 0.8.2
  - - OCL queries more difficult to write for UML profile than meta-models

- Perspectives
  - TOPHOO: add real time concepts from MARTE, support formal behavioural specification, proposal for PolarSys
  - FAST: add requirements management, interoperability with Scade Systems, proposal for PolarSys
Questions ?
Evaluate This Session

1. Sign-in: www.eclipsecon.org

2. Select session from schedule

3. Evaluate: +1  0  -1