

Product Derivation



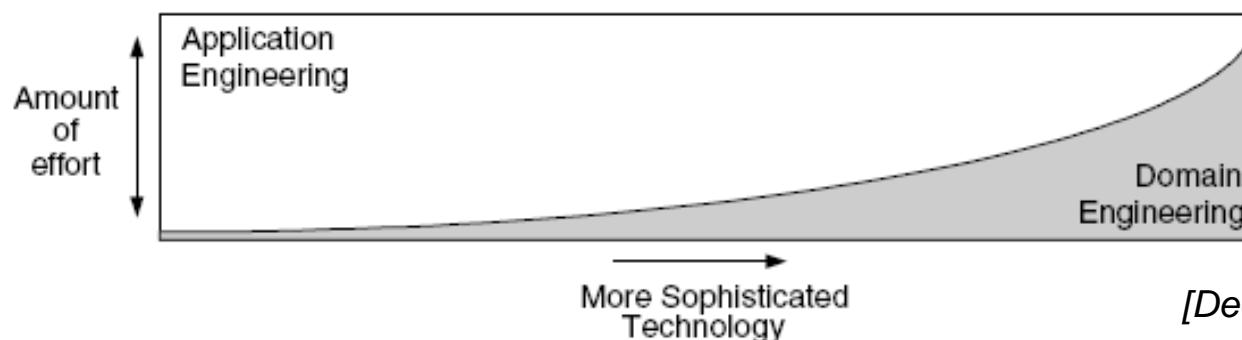
KV Product Line Engineering (343.354)

Dr. Roberto Lopez-Herrejon

Dr. Rick Rabiser

Product Derivation

Meeting a particular **customer's needs** by making decisions to **select and configure assets** from a product line



[Deelstra et al. 2005]

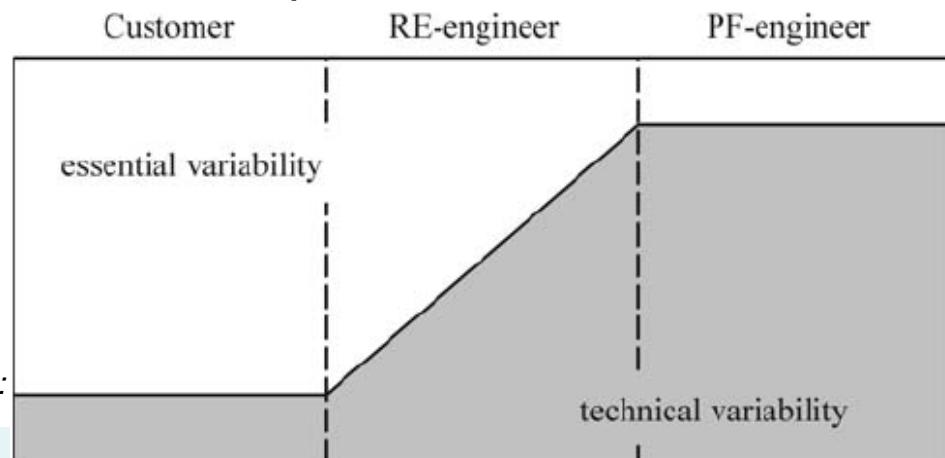
Outline

- ▶ Motivation
- ▶ The Application Engineering Process
- ▶ Communicating Variability
- ▶ Requirements for Product Derivation
- ▶ Product Derivation with DOPLER^{UCon}
- ▶ Derivation in other Tools
- ▶ What now?
 - The requirements engineering problem
 - Visualization community
 - Product configurators

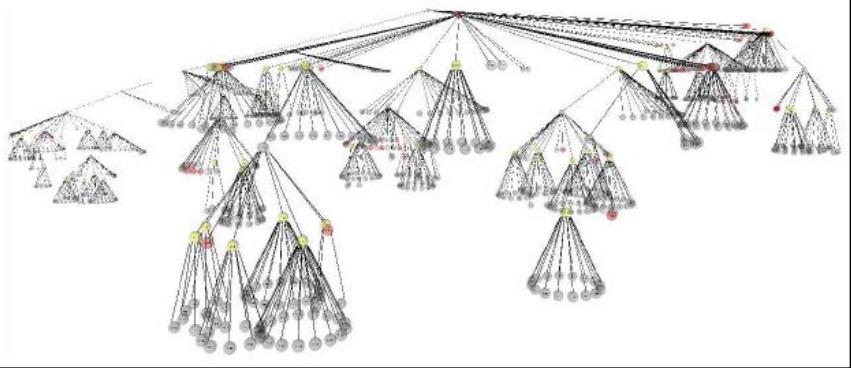
Motivation [1/2]

- ▶ Investments for defining the product line must be outweighed by the benefits of being able to derive products!
- ▶ „*deriving individual products from shared software assets is a **time-consuming and expensive** activity*“ [Deelstra et al. 2005]
- ▶ **Variability** provided by a PL has to be **communicated**
 - to the customers in order to elicit their requirements
 - to other stakeholders, e.g.,
 - Sales/Project managers
 - Developers
 - Testers
 - etc.

[Halman and Pohl 2004]:



Motivation [2/2]



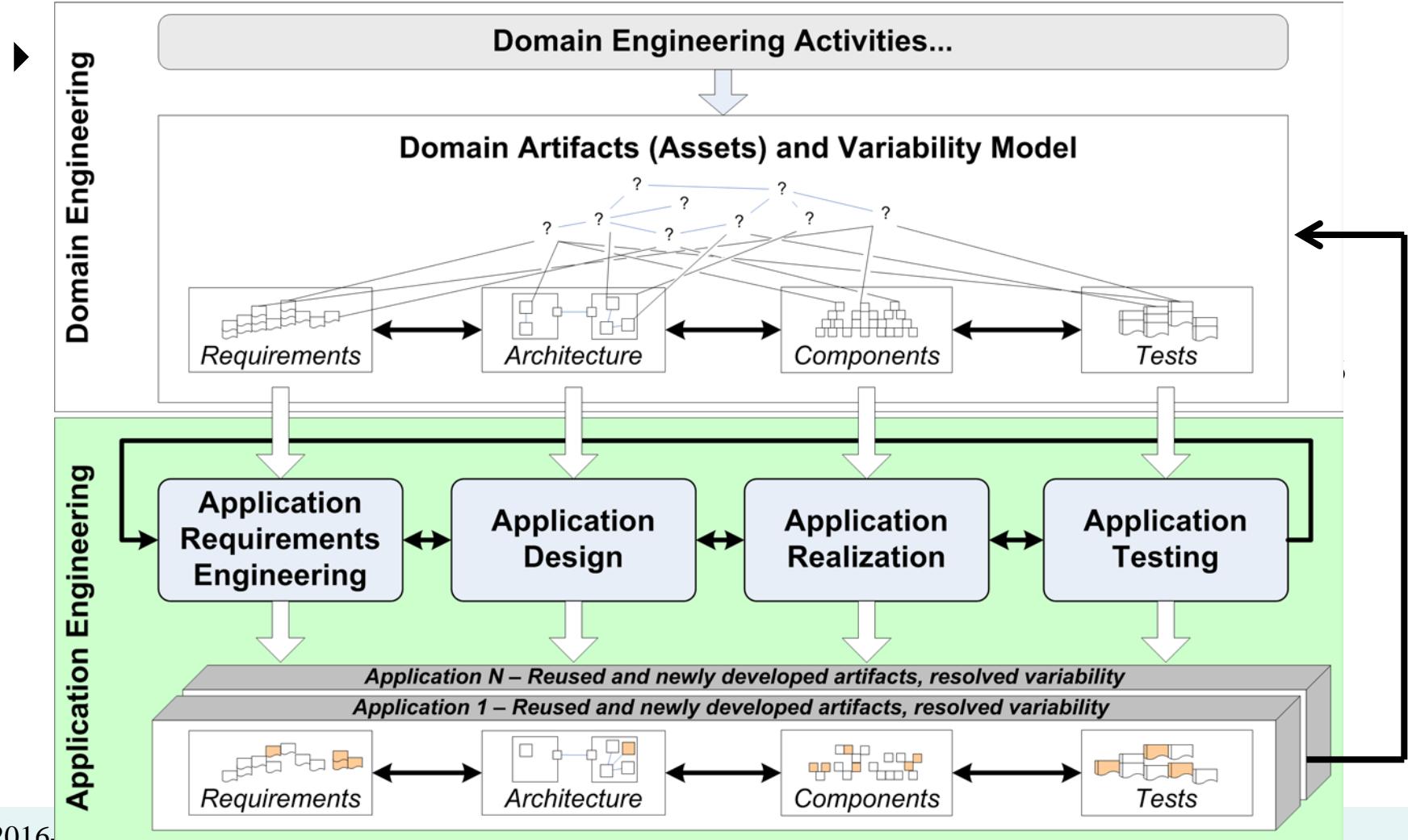
- ▶ Derivation is hard because:
 - Variability models typically are large and complex
 - 1000s of elements with 1000s of (non-trivial) dependencies
 - Diverse quite different people have to understand (different parts) of the provided variability
 - Derivation is a project
 - possibly long duration
 - user, roles, tasks need to be managed
 - The „blue-sky“ derivation scenario (pure selection and configuration of existing assets) simply does not hold in practice
 - More realistic scenario: Customers have special wishes and requirements not covered by the PL
 - How to capture, negotiate, document, validate, manage these rqts?
 - Input to additional development and product line evolution

Confusing Terms...

- ▶ Diverse terms are often used for describing the same process
 - Product Derivation
 - Product Customization
 - Product Configuration
 - Application Engineering
 - Product Production
 - etc.
- ▶ Our **opinion**
 - Product configuration is synonym for product derivation
 - Derivation -> selection
 - Configuration -> customization
 - Application Engineering -> (software) engineering process perspective
 - Product Production -> business (production planning) process perspective

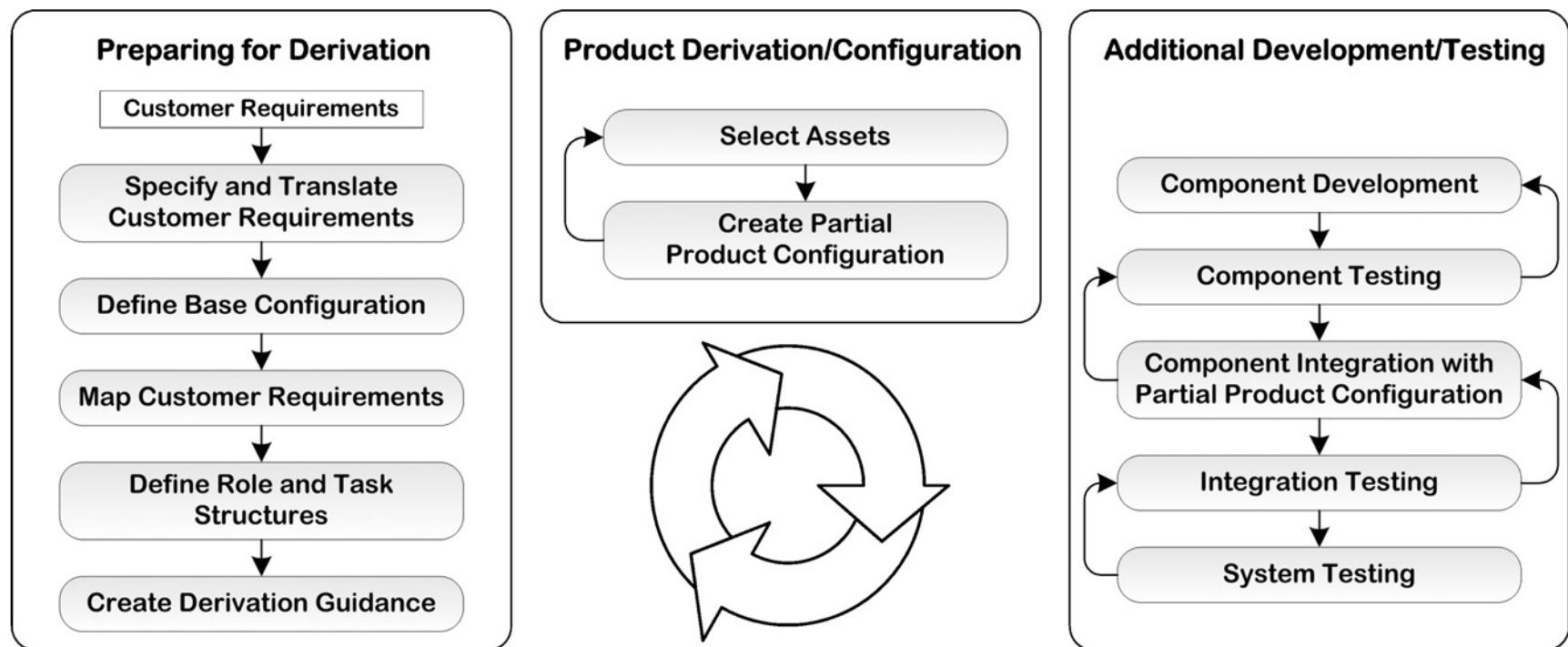


Application Engineering Process



Key Activities of Product Derivation

- compared two product derivation approaches developed in two different, independent research projects
- analyzed three existing product derivation approaches for their support for these activities

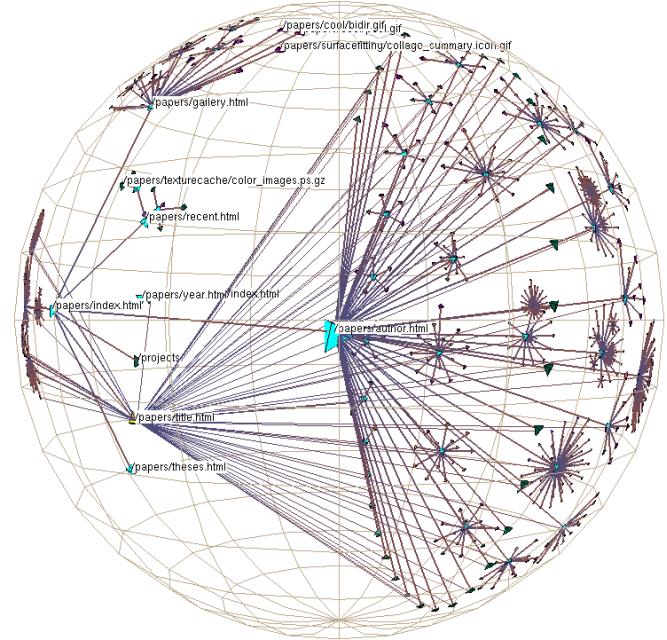


How to Communicate Variability?

- ▶ Textual
- ▶ Tabular
- ▶ Tree-based
- ▶ Graph-based

- ▶ 2D vs. 3D

- ▶ Static (just displaying) vs. Dynamic (allowing interactions) visualization

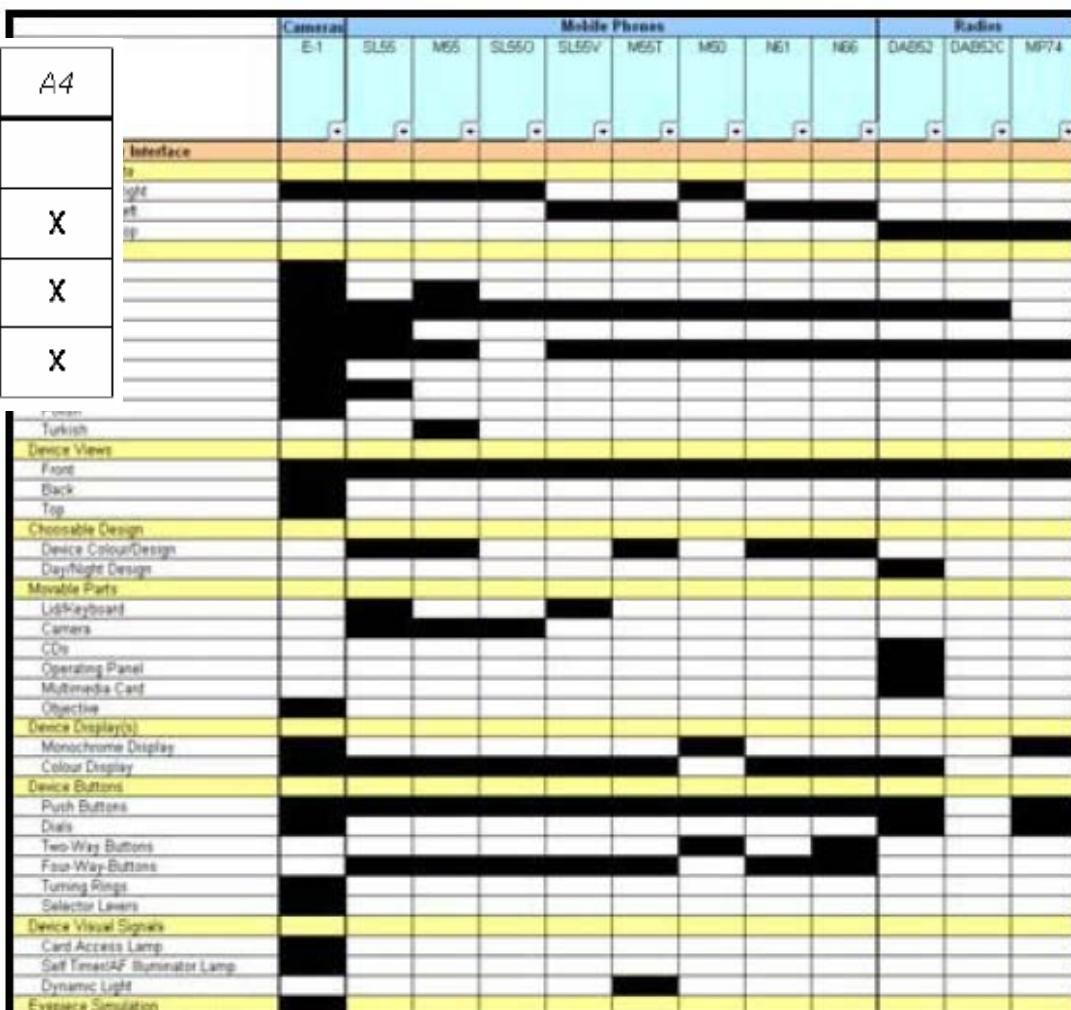


Communication using Matrices

Application-Rqts Matrix

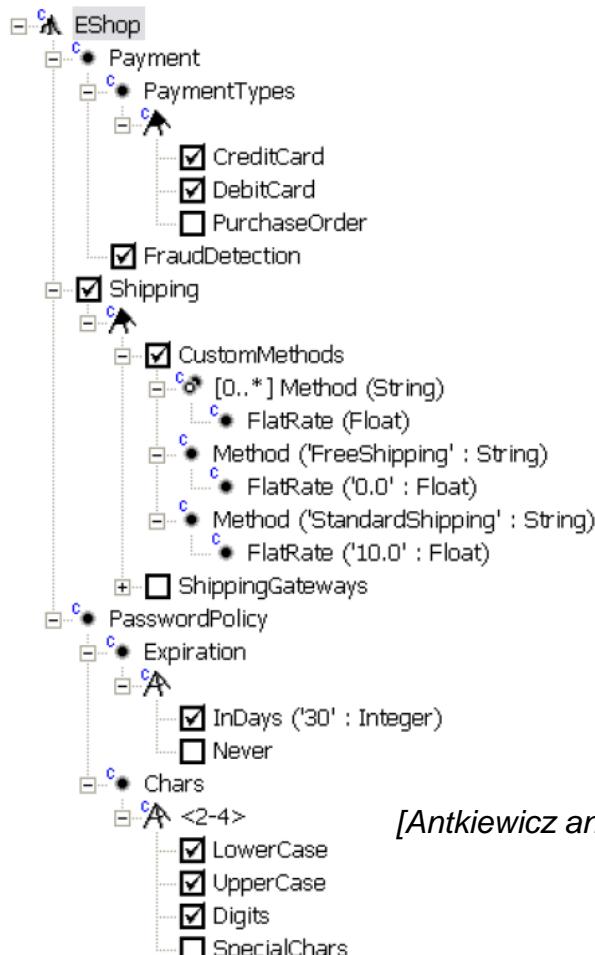
Req / Applications	A1	A2	A3	A4
<i>Manual Transmission</i>	X		X	
<i>Automatic Transmission</i>		X		X
<i>Radio Entertainment</i>	X	X		X
<i>CD Entertainment</i>	X		X	X

Application-Feature Matrix

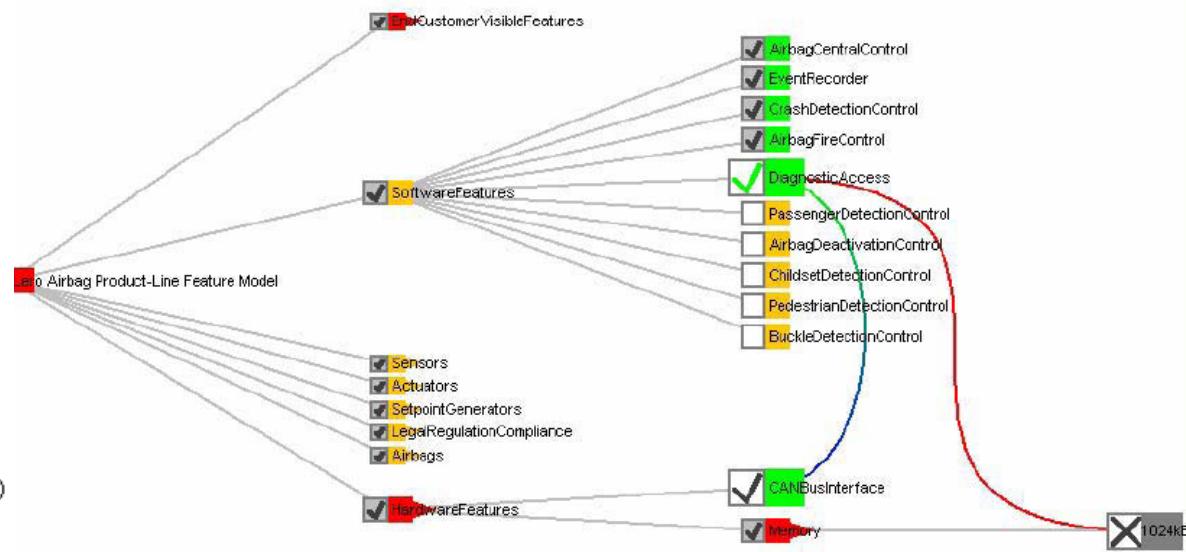


Communication using (Feature) Trees

Textual



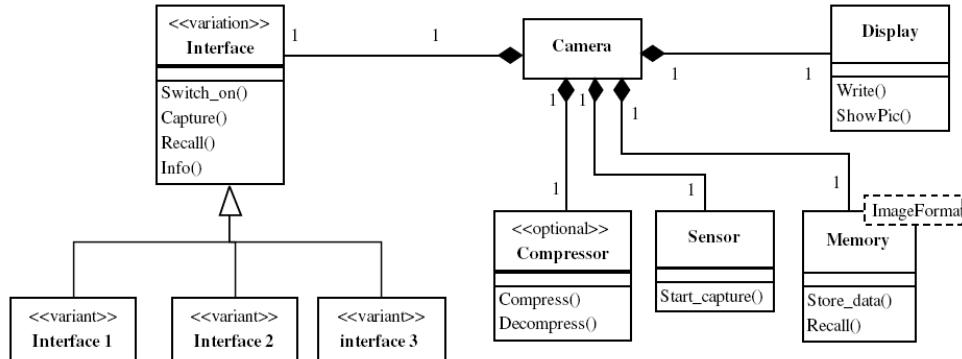
Graphical



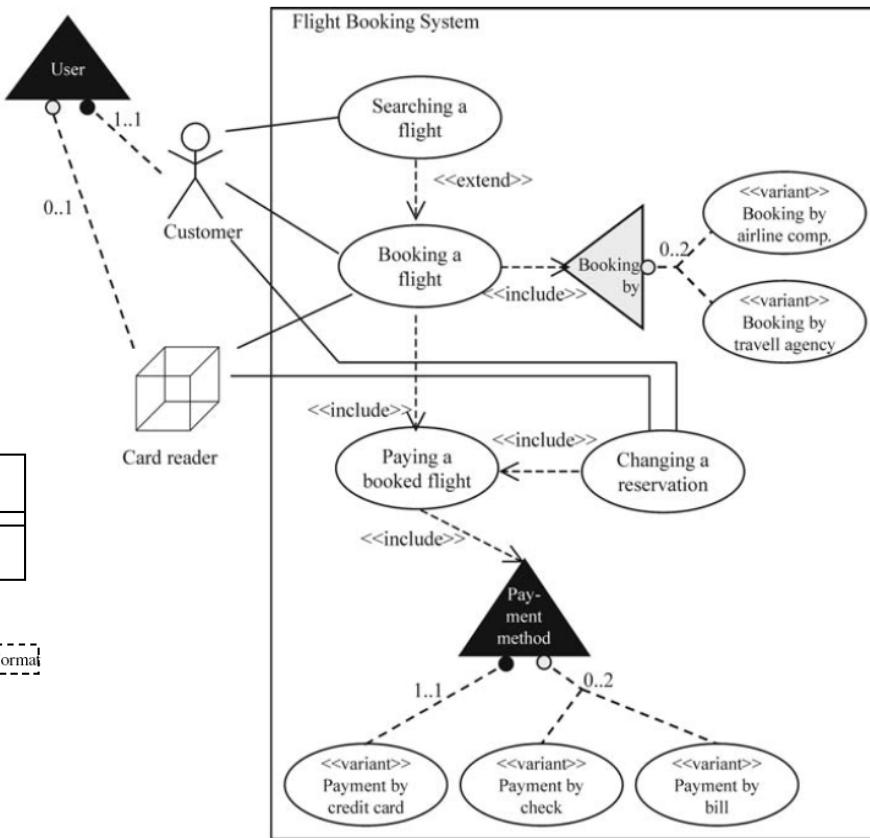
Communication using (extended) UML notations

Use Case diagrams

Class Diagrams



[Ziadi et al. 2003]



[Halmans and Pohl 2004]

Communication using Decisions

Textual

- ② Are multiple steelgrades supported?
- ② Which defects shall be tracked?
- ② Is the mechanic provided by Siemens VAI?
- ② Shall Tundish Level1 Preheating signals be provided cyclically?
- ✓ What is the scope of the product to be delivered?
- ② Which product types shall be casted?
- ② Can the mold width be changed?
- ② How is the strand fed into the caster?
- ② Which casting modes are supported?
- ② How many strands does the caster have?
- ② Which models shall be delivered?**
- ② Are setpoints sent as a list?
- ② Who provided the locking strategy?
- ② Who generates the product ID?
- ② Does the L1 Interface conform to the VAI Standard?
- ② Enter the reference-position of the Torch Cutting Machine
- ② Enter the distance to the reference-position of the Torch Cutting Machine
- ② Shall the 3D thermal tracking model be used?
- ② Is adjusting the nozzle spray width controlled by the cooling model?
- ② Which cooling model do you want to apply?
- ② Which Dynagap strategies shall be used?

[Rabiser 2009]

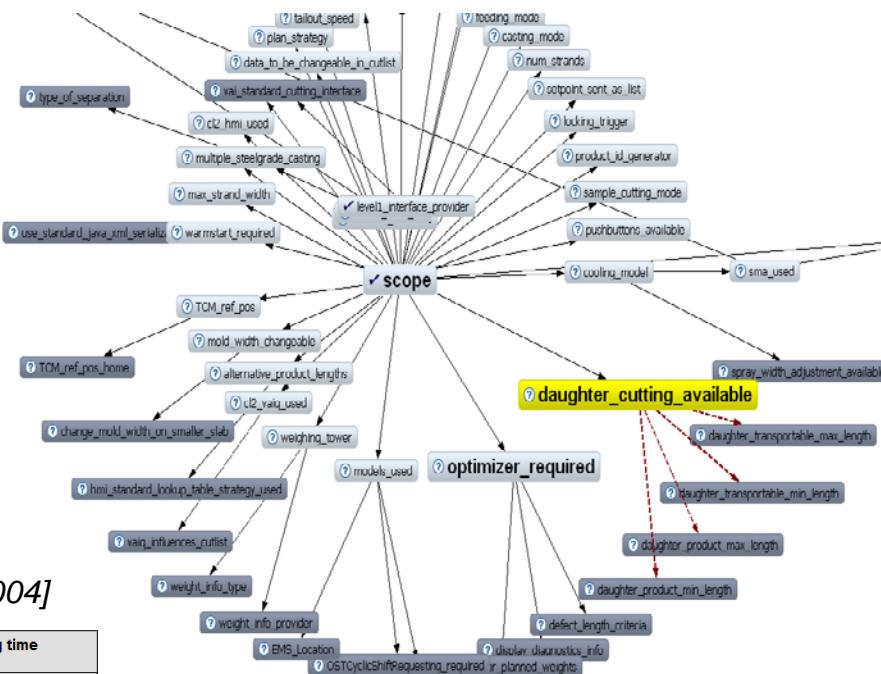
basis

choose

DPT
Dynaflex
EMS
MCO
Mold Level
NMI
NZC

ok

Graphical



Tabular

[Schmid and John, 2004]

Name	Visibility	Description	Range	Constraints	Binding time
infoSource		What kind of information source does the Web platform have?	Static Dynamic External		Runtime
userManager		Which kind of user management will be applied?	User, Group Hierarchy	Hierarchy => Group	Compile time
DatabaseSupport	infoSource == Dynamic	What concrete database support will be used?	Access MySQL Oracle		Install time / System initialization

What support needed for Product Derivation?

- ▶ Elicited with surveys at VaMoS Workshop 2008 and IEEE International Software Product Line Conference 2008
 - Surveys developed based on systematic literature review
- ▶ 55 questionnaires distributed, 39 returned, 29 analyzed based on experience threshold
- ▶ Result: 6 (High-Level) Requirements
 - Examples of how to realize these

Product Derivation: Basic Support

Requirement	Implementation examples
Basic support	
Automated and interactive variability resolution	<ul style="list-style-type: none"> Tool support for making decisions (e.g., selecting features) Engines for resolving constraints and dependencies
Adaptability and extensibility	<ul style="list-style-type: none"> Meta-modeling support Extension points for the integration of domain-specific generators
Application requirements management support	<ul style="list-style-type: none"> Requirements management capabilities in derivation
Flexible and user-specific visualizations of variability	<ul style="list-style-type: none"> Support for filtering, sorting, searching the decision-space
End-user guidance	<ul style="list-style-type: none"> Background information and rationale on choices
Project management support	<ul style="list-style-type: none"> Tracking project-related information

Product Derivation: Advanced Support

Requirement	Advanced support
Automated and interactive variability resolution	<ul style="list-style-type: none">Support for the automatic inclusion and customization of assetsImmediate visual feedback when making choicesAPIs for interacting with existing tools
Adaptability and extensibility	
Application requirements management support	<ul style="list-style-type: none">Support for relating product-specific requirements with existing variabilityPlanning support for evolution
Flexible and user-specific visualizations of variability	<ul style="list-style-type: none">Role- and user-specific visualizationsTask-specific visualizationsModifiable visualizations
End-user guidance	<ul style="list-style-type: none">Hints and recommendationsInterface to simulation support
Project management support	<ul style="list-style-type: none">User/role/task management support

Comparison of PD Tools

Regarding Requirements for PD

(--no support, - some support, + supported, ++ very well supported).

Capability/Tool	COVAMOF-VS	FMP	Gears	Kumbang	<i>pure::variants</i>	V-Manage	ViSiT-FC	XFeature	DOPLER ^{ucon}
Automated & Interactive Variability Resolution	-	+	-	-	++	+	++	+	++
Adaptability and Extensibility	-	--	-	-	++	--	--	-	++
Integrated Application Requirements Management	--	--	--	--	+	--	--	--	++
Flexible and User-specific Visualization of Variability	-	+	-	+	+	-	++	+	++
End-user Guidance for Decision-making	+	-	-	--	+	-	+	--	++
Project Management Support	--	--	-	--	+	--	--	--	+

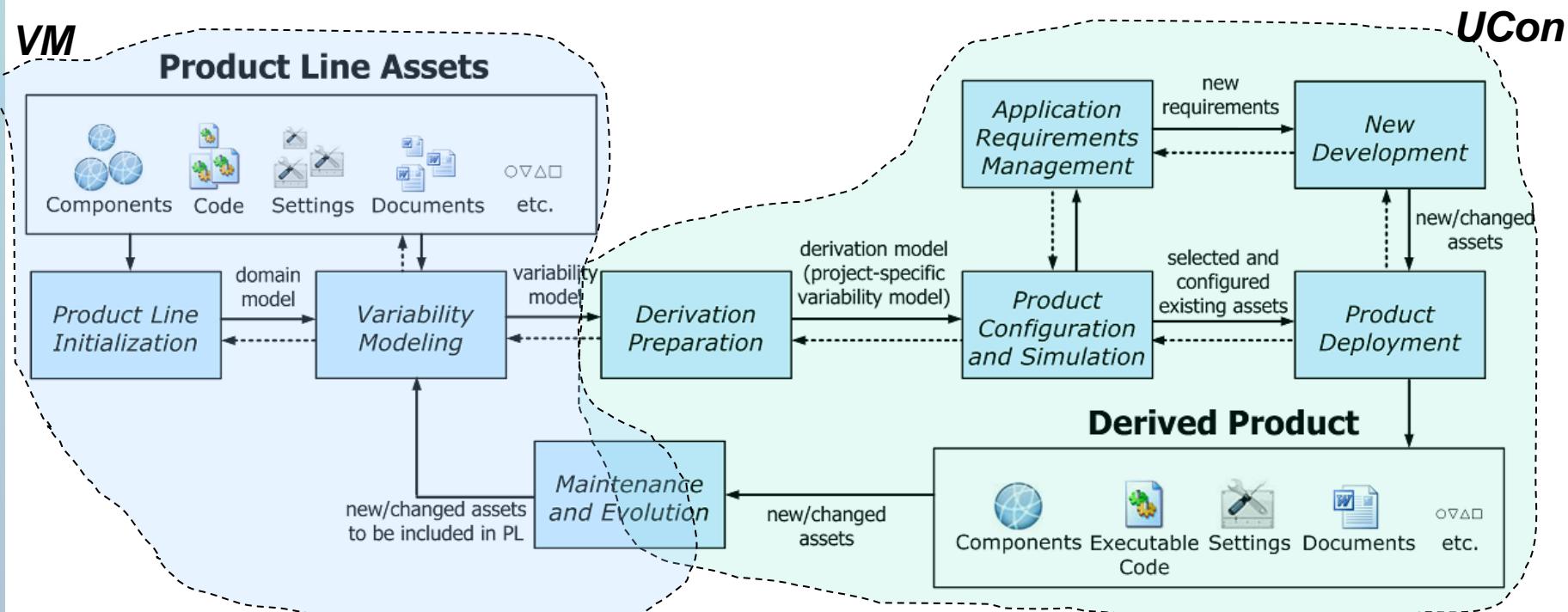
Product Derivation with



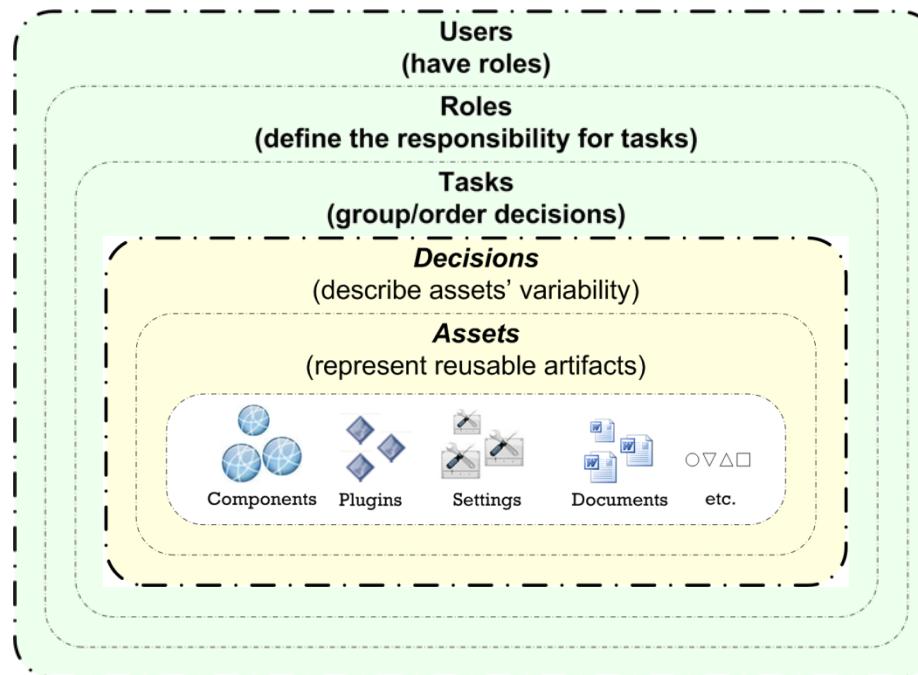
© CDLab for ASE, JKU
<http://ase.jku.at/dopler>

[Rabiser et al. 2006-2011]

- ▶ DOPLER comprises
 - DOPLER^{VM} (variability modeling and management)
 - DOPLER^{UCon} (product configuration and appl. reqts. mgmt.)



Users/Roles/Tasks/Decisions

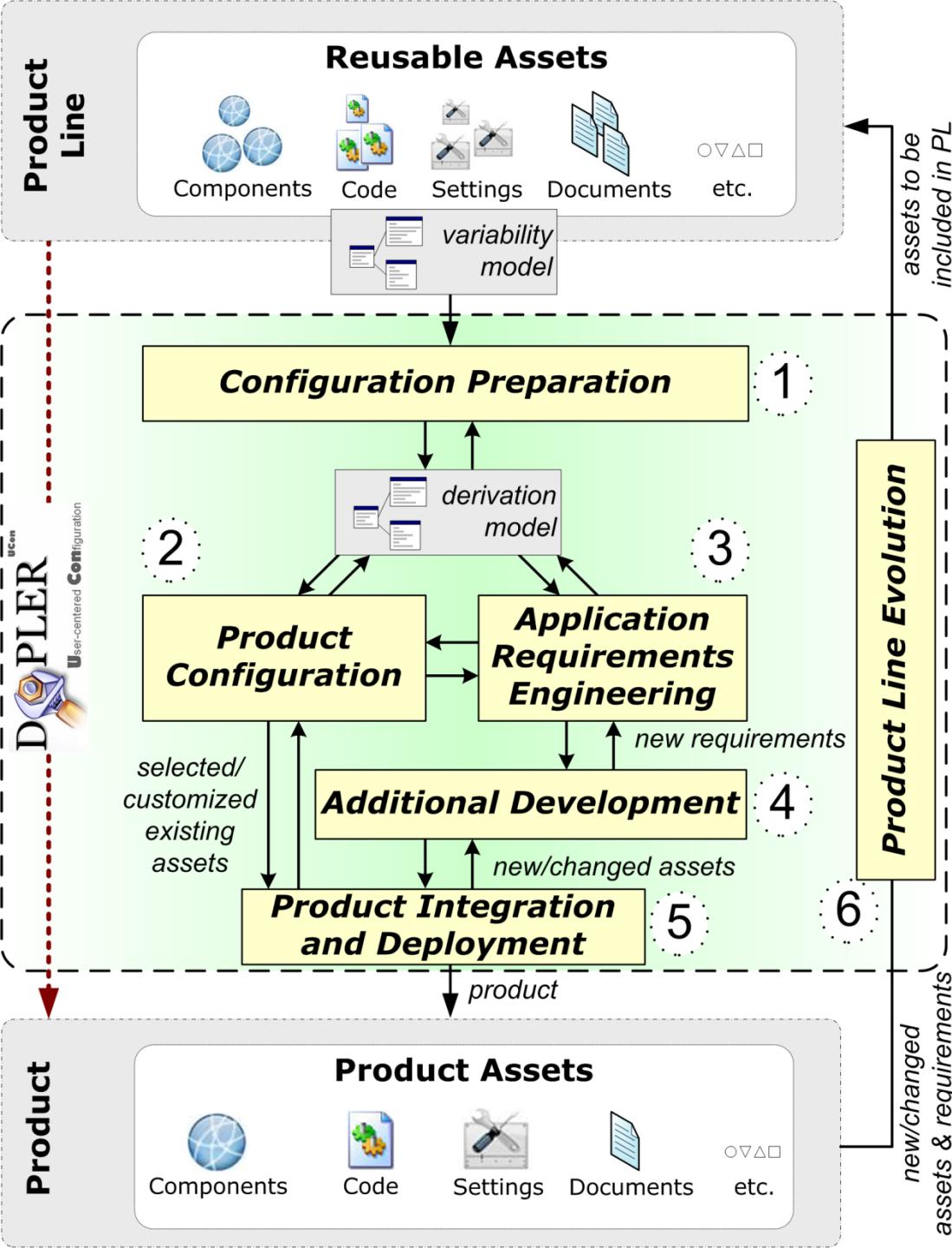


Process

Rabiser: A User-Centered Approach to Product Configuration in Software Product Line Engineering, PhD Thesis, JKU, Feb 2009.

Rabiser and Dhungana: Integrated Support for Product Configuration and Requirements Engineering in Product Derivation, EUROMICRO SEAA 2007.

Rabiser et al.: Supporting Product Derivation by Adapting and Augmenting Variability Models, SPLC 2007.

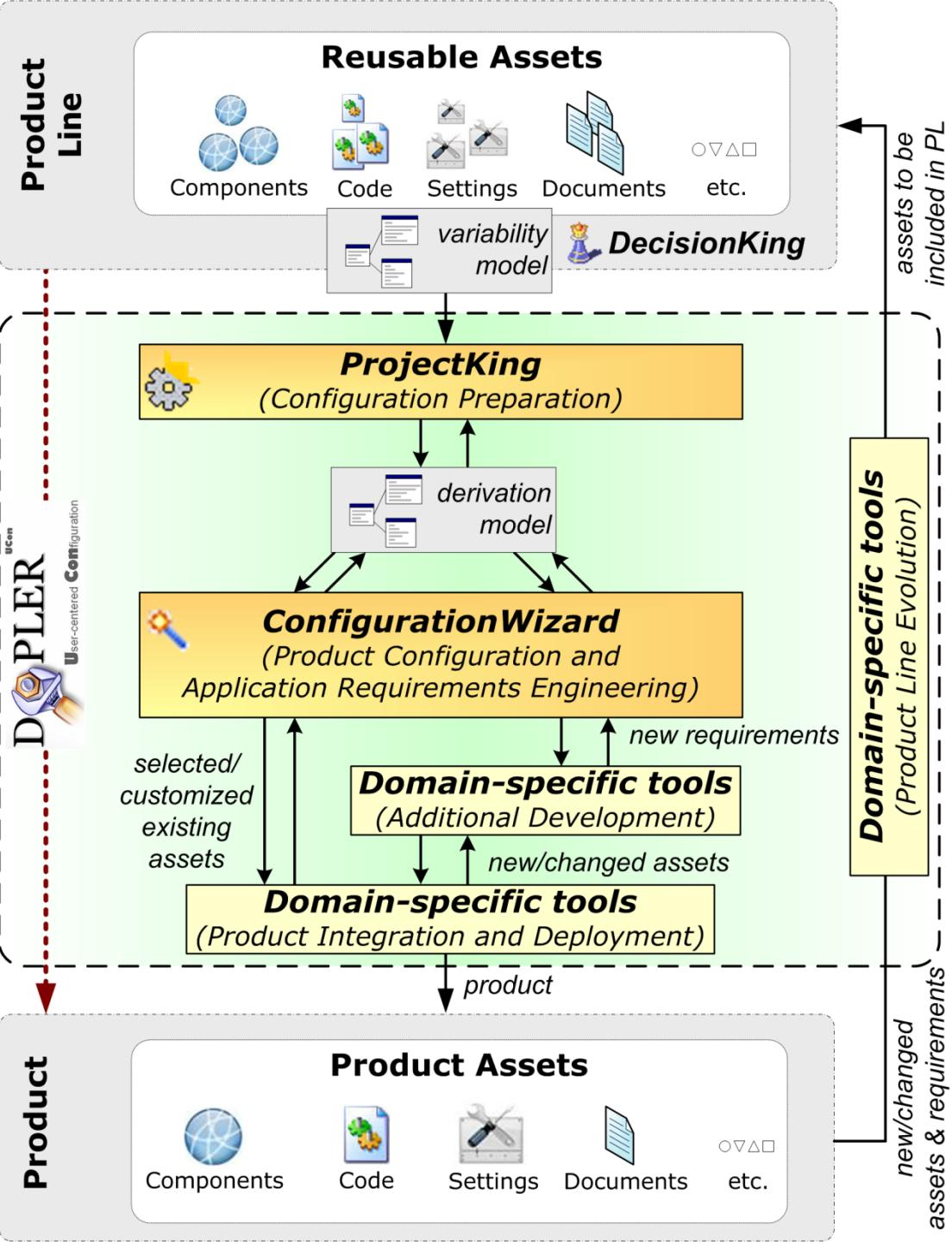


Tool Support

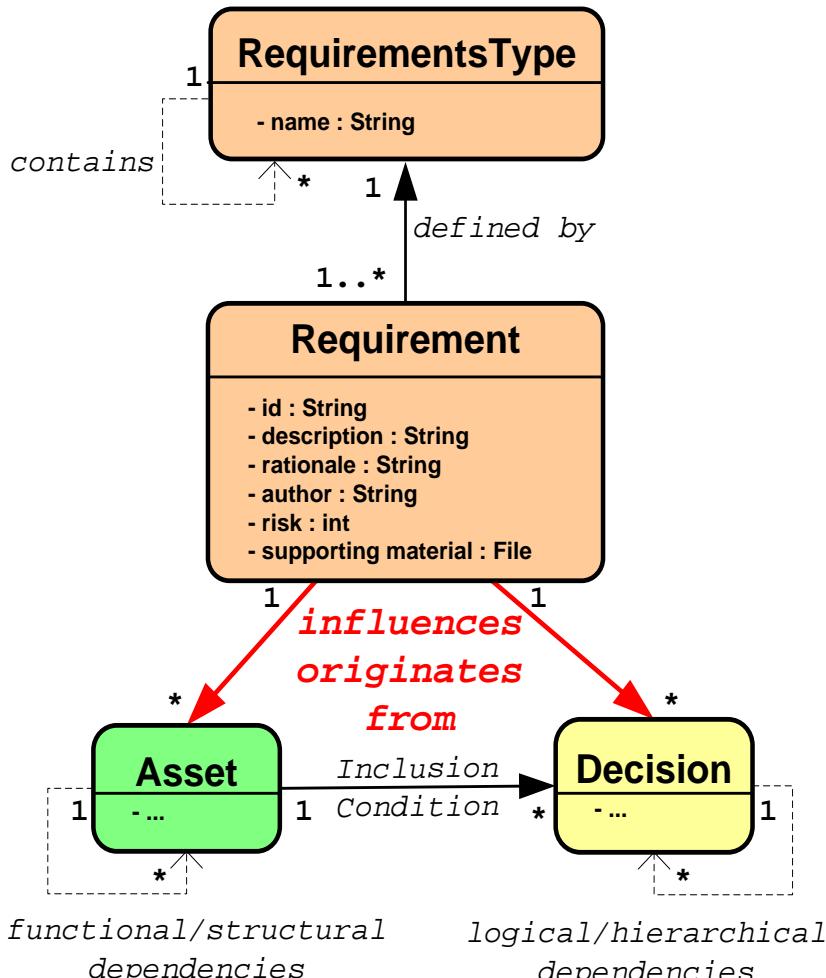
Rabiser: A User-Centered Approach to Product Configuration in Software Product Line Engineering, PhD Thesis, JKU, Feb 2009.

Rabiser et al.: Involving Non-Technicians in Product Derivation and Requirements Engineering: A Tool Suite for Product Line Engineering, Tool Demo, RE 2007.

Rabiser: Flexible and User-Centered Visualization Support for Product Derivation, ViSPL 2008, SPLC 2008.



Relating Requirements with Existing Assets and Variability



► Origins

- Assets/Decisions that gave rise to a requirement

► Influences

- Possible impact of requirement on existing Assets/Decisions

► Based on Volere Template [\[www.volere.co.uk\]](http://www.volere.co.uk)

Demo: ConfigurationWizard

- ▶ Default visualization support
 - textual
 - tabular
 - tree-based
 - graph-based

- ▶ Flexible/Extensible
 - Arbitrary Visualizations by developing your own visualizer extension

Study on Capabilities for End-User Guidance [ASE 2012]

- ▶ 9 business-oriented Siemens stakeholders
 - sales manager (SM)
 - two product owners (PO)
 - a marketing manager (MM)
 - a product lifecycle manager (PLM)
 - a team manager (TM)
 - a developer (D)
 - a project manager (PM)
 - and a sales person (S)
- ▶ Worked with the CW Tool and performed pre-specified tasks
- ▶ Think-aloud, logging, screen-casting, usability questionnaire, structured interviews

Study System and Configuration Tasks

Selected System

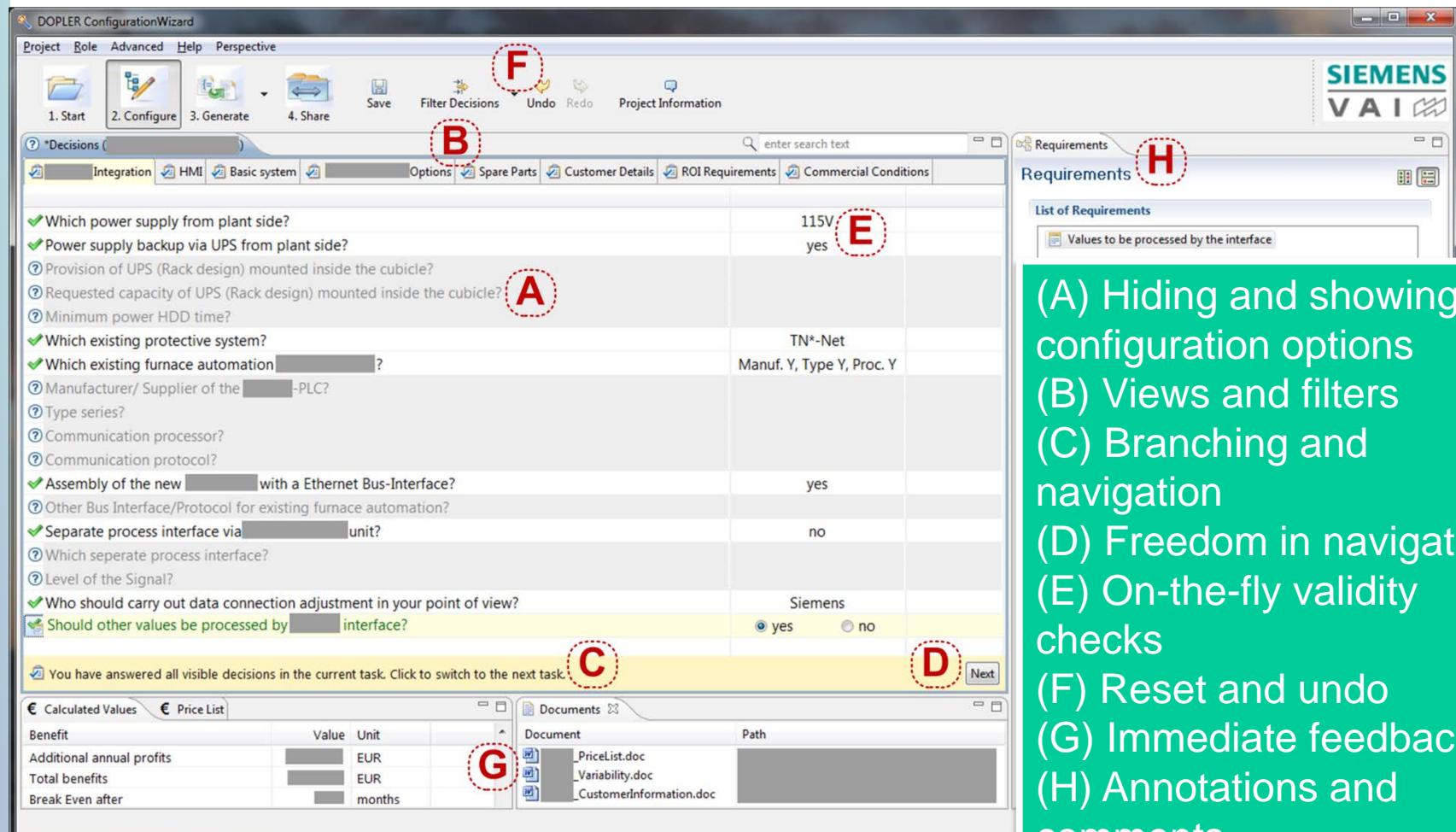
- Siemens Electrode Control Systems for electric arc furnaces
- Configuration of approved sales offers (10-100+ page documents)

Study Tasks

- Configure
 - Customer details and ROI requirements
 - Basic system and spare parts
 - Human-machine interface
 - Commercial conditions
- Generate sales documents
- View and check business values and price list
- Capture customer-specific requirements



Configuration Wizard (CW) and Capabilities



- (A) Hiding and showing configuration options
- (B) Views and filters
- (C) Branching and navigation
- (D) Freedom in navigation
- (E) On-the-fly validity checks
- (F) Reset and undo
- (G) Immediate feedback
- (H) Annotations and comments

(A) Hiding and showing configuration options

The screenshot displays two windows side-by-side. On the left is the 'DOPLER ConfigurationWizard' application. The main window shows a grid of configuration decisions. Several questions are marked with green checkmarks, indicating they have been answered. A question in the first column is circled with a red dotted line and labeled 'A'. Another question in the second column is circled with a red dotted line and labeled 'B'. A question in the third column is circled with a red dotted line and labeled 'C'. A question in the fourth column is circled with a red dotted line and labeled 'D'. A question in the fifth column is circled with a red dotted line and labeled 'E'. A question in the sixth column is circled with a red dotted line and labeled 'F'. A question in the seventh column is circled with a red dotted line and labeled 'G'. A question in the eighth column is circled with a red dotted line and labeled 'H'. Below the grid, a message says 'You have answered all visible decisions in the current task. Click to switch to the next task.' To the right of the main window is a 'Requirements' dialog box with fields for 'List of Requirements', 'Details', 'Description', 'Justification', 'Origin', 'Risk level', and 'Priority'. The Siemens logo is visible in the background.

- (+) Reduces complexity
- (+) Provides immediate feedback
- (-) Graying out unanswerable questions can cause confusion

→ Allow users to turn-off this capability

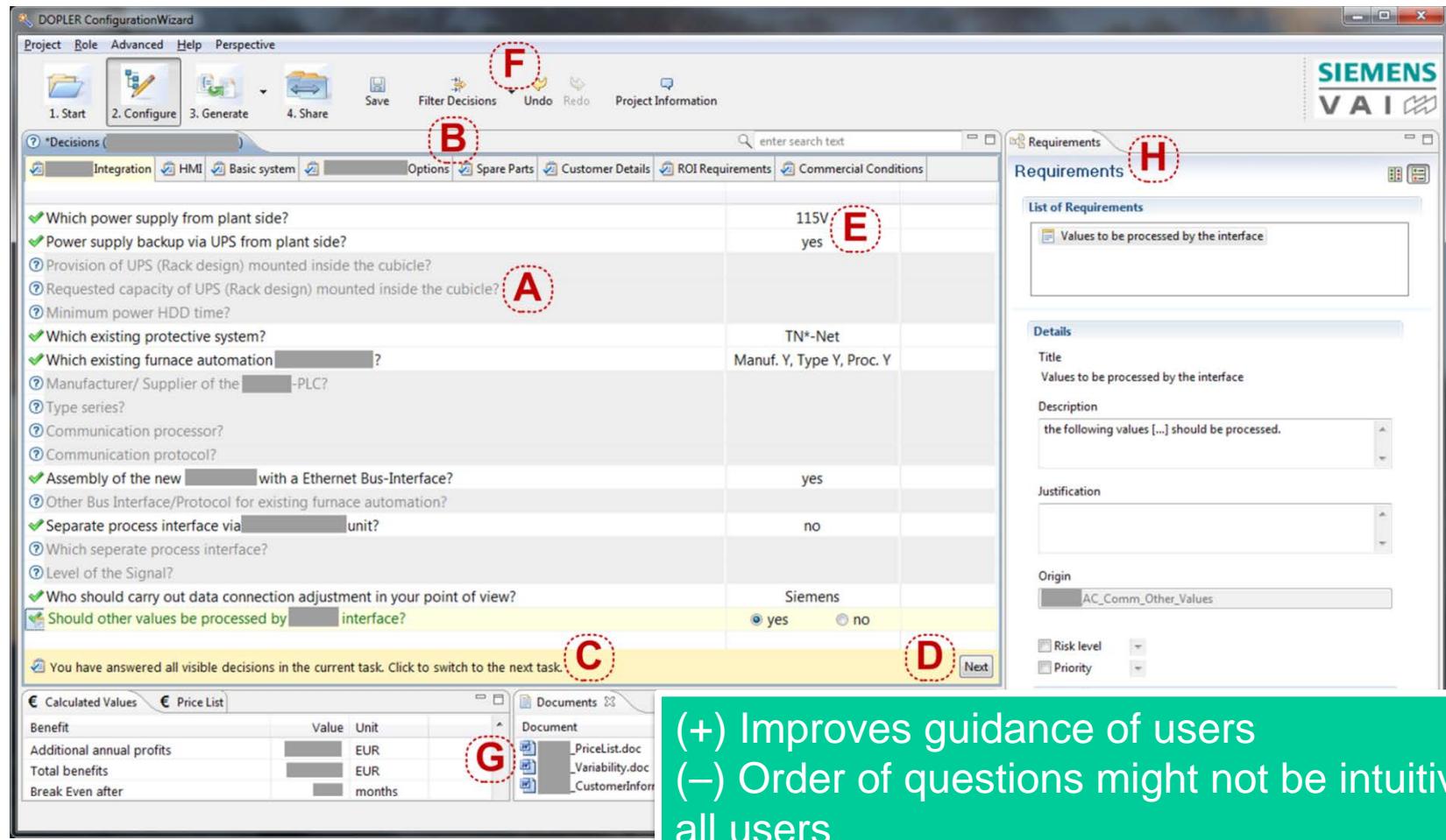
(B) Views and filters

The screenshot displays the DOPLER Configuration Wizard application window. The interface includes:

- Toolbar:** Project, Role, Advanced, Help, Perspective, 1. Start, 2. Configure, 3. Generate, 4. Share, Save, Filter Decisions (circled in red), Undo, Redo, Project Information.
- Main Area:** A grid-based configuration table with rows and columns of questions and answers. Key entries include:
 - Row 1: Which power supply from plant side? (yes) circled in red as point E.
 - Row 2: Provision of UPS (Rack design) mounted inside the cubicle? (no) circled in red as point A.
 - Row 3: Requested capacity of UPS (Rack design) mounted inside the cubicle? (TN*-Net)
 - Row 4: Minimum power HDD time? (Manuf. Y, Type Y, Proc. Y)
 - Row 5: Which existing protective system?
 - Row 6: Which existing furnace automation? (no)
 - Row 7: Manufacturer/ Supplier of the -PLC? (Siemens)
 - Row 8: Type series?
 - Row 9: Communication processor?
 - Row 10: Communication protocol?
 - Row 11: Assembly of the new [redacted] with a Ethernet Bus-Interface? (yes)
 - Row 12: Other Bus Interface/Protocol for existing furnace automation? (no)
 - Row 13: Separate process interface via [redacted] unit? (no)
 - Row 14: Which separate process interface? (no)
 - Row 15: Level of the Signal? (no)
 - Row 16: Who should carry out data connection adjustment in your point of view? (Siemens)
 - Row 17: Should other values be processed by [redacted] interface? (yes) circled in red as point D.
- Task Bar:** You have answered all visible decisions in the current task. Click to switch to the next task. (circled in red as point C).
- Bottom Left:** Calculated Values and Price List tabs, showing benefit calculations (Additional annual profits, Total benefits, Break Even after) and currency conversion (Value: EUR, Unit: months).
- Bottom Right:** Documents pane showing files: _PriceList.doc, _Variability.doc, _CustomerInformation.doc.
- Right Panel:** Requirements window titled Requirements (circled in red as point H). It contains sections for List of Requirements, Details, Description, Justification, Origin, Risk level, and Priority.

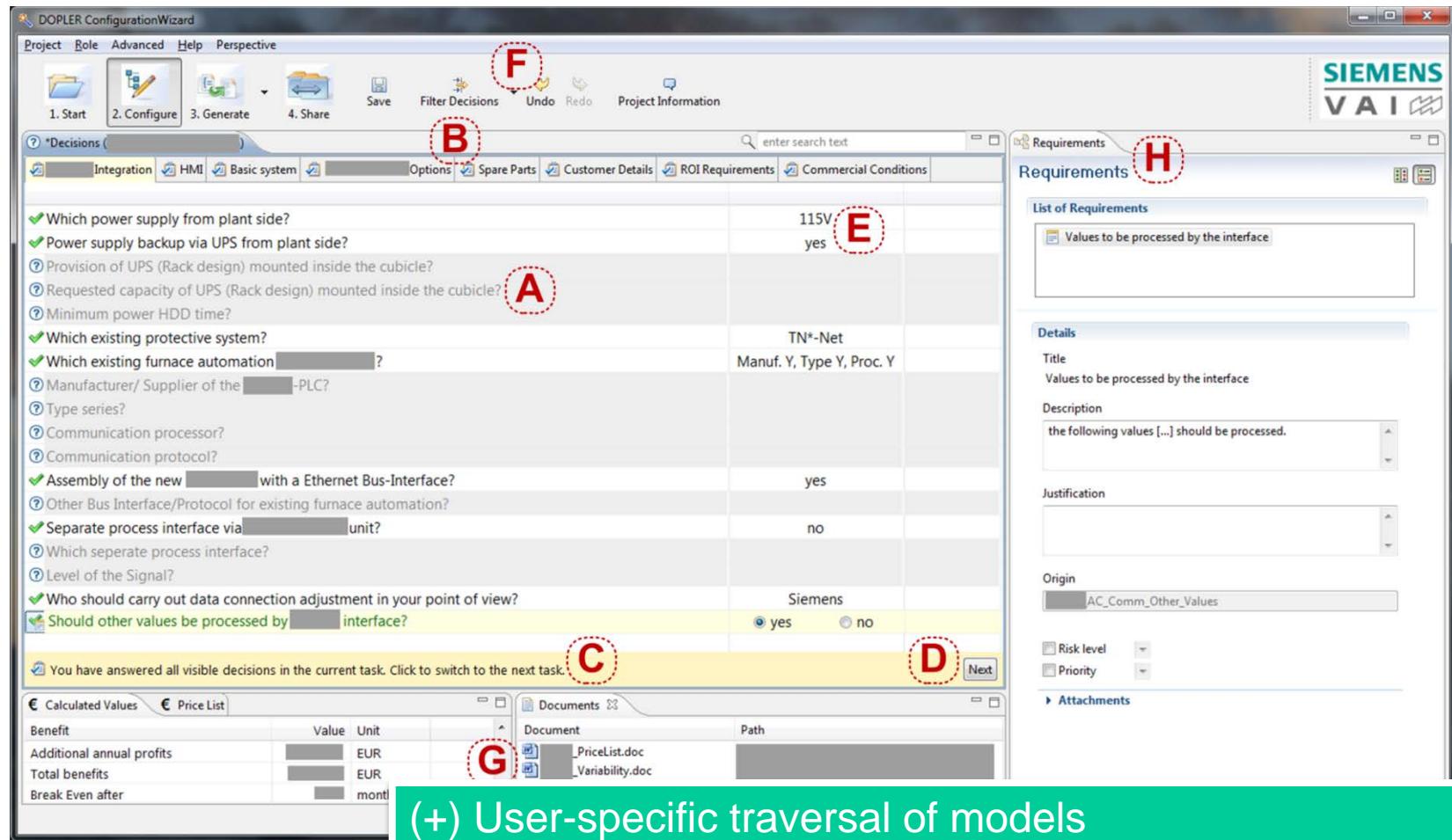
(+) Essential for large and complex models
 (-) Dependencies might not be obvious
 → Balance the number of views and elements
 → Visualize dependencies on demand

(C) Branching and navigation



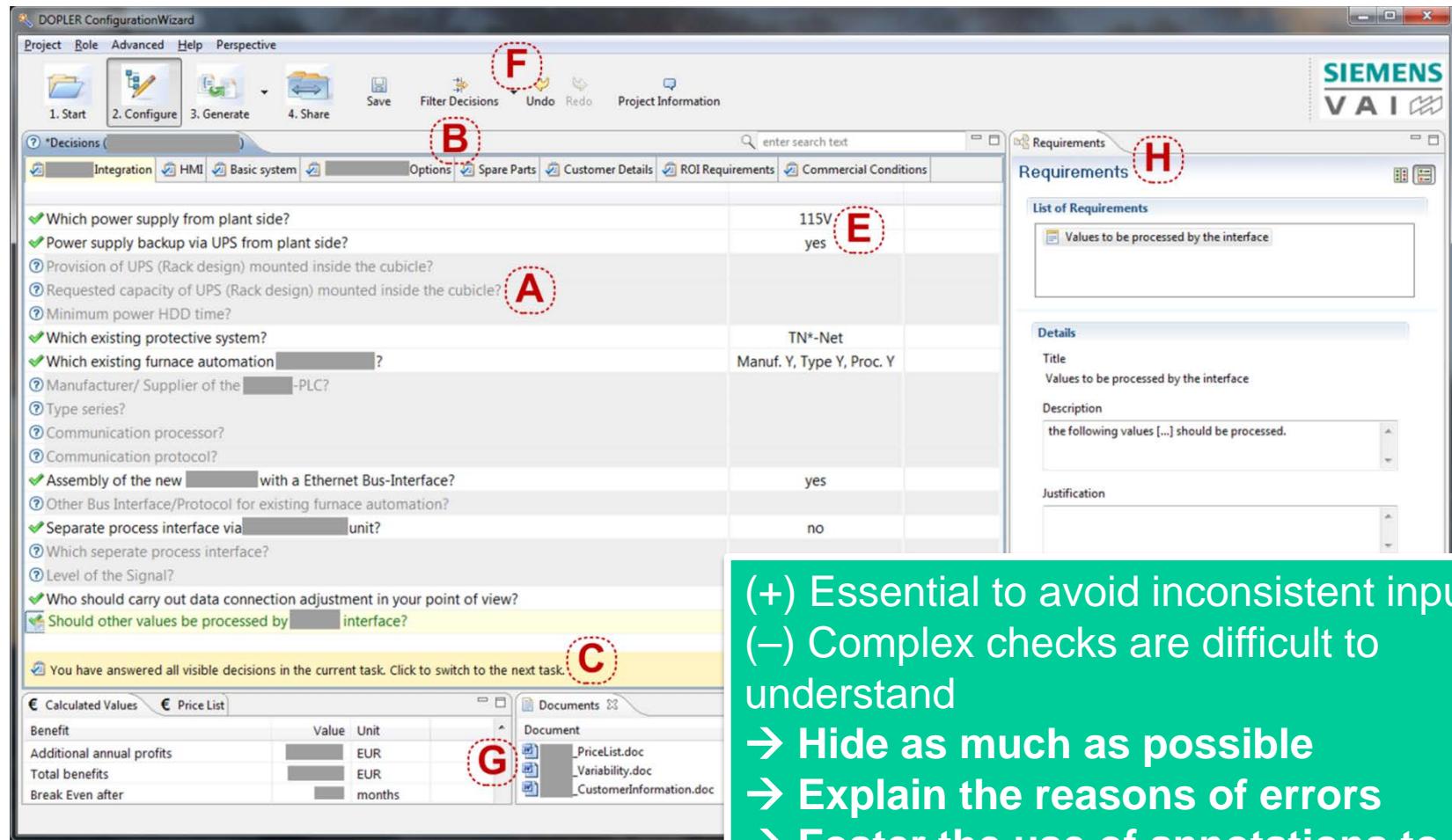
- (+) Improves guidance of users
 - (-) Order of questions might not be intuitive for all users
- Combine with freedom in navigation
- Provide user-specific navigation paths

(D) Freedom in navigation



(+) User-specific traversal of models
 (-) Lack of guidance if not combined with other capabilities
 → Combine with branching and navigation

(E) On-the-fly validity checks



The screenshot shows the DOPLER ConfigurationWizard application interface. Key elements include:

- Toolbar:** Project, Role, Advanced, Help, Perspective, Save, Filter Decisions, Undo, Redo, Project Information. Annotations **F** and **B** are circled around the 'Filter Decisions' and 'Save' buttons respectively.
- Decisions List:** A table listing requirements. Annotations **A** and **E** are circled around specific rows. Annotation **C** points to a message at the bottom of the list.
- Calculated Values:** A table showing benefit calculations. Annotation **G** points to the 'Price List' tab.
- Documents:** A sidebar showing document files like '_PriceList.doc', '_Variability.doc', and '_CustomerInformation.doc'.
- Requirements Window:** A separate window titled 'Requirements' with sections for 'List of Requirements' (containing 'Values to be processed by the interface'), 'Details' (Title: 'Values to be processed by the interface', Description: 'the following values [...] should be processed.', Justification), and 'SIEMENS VAI' branding.

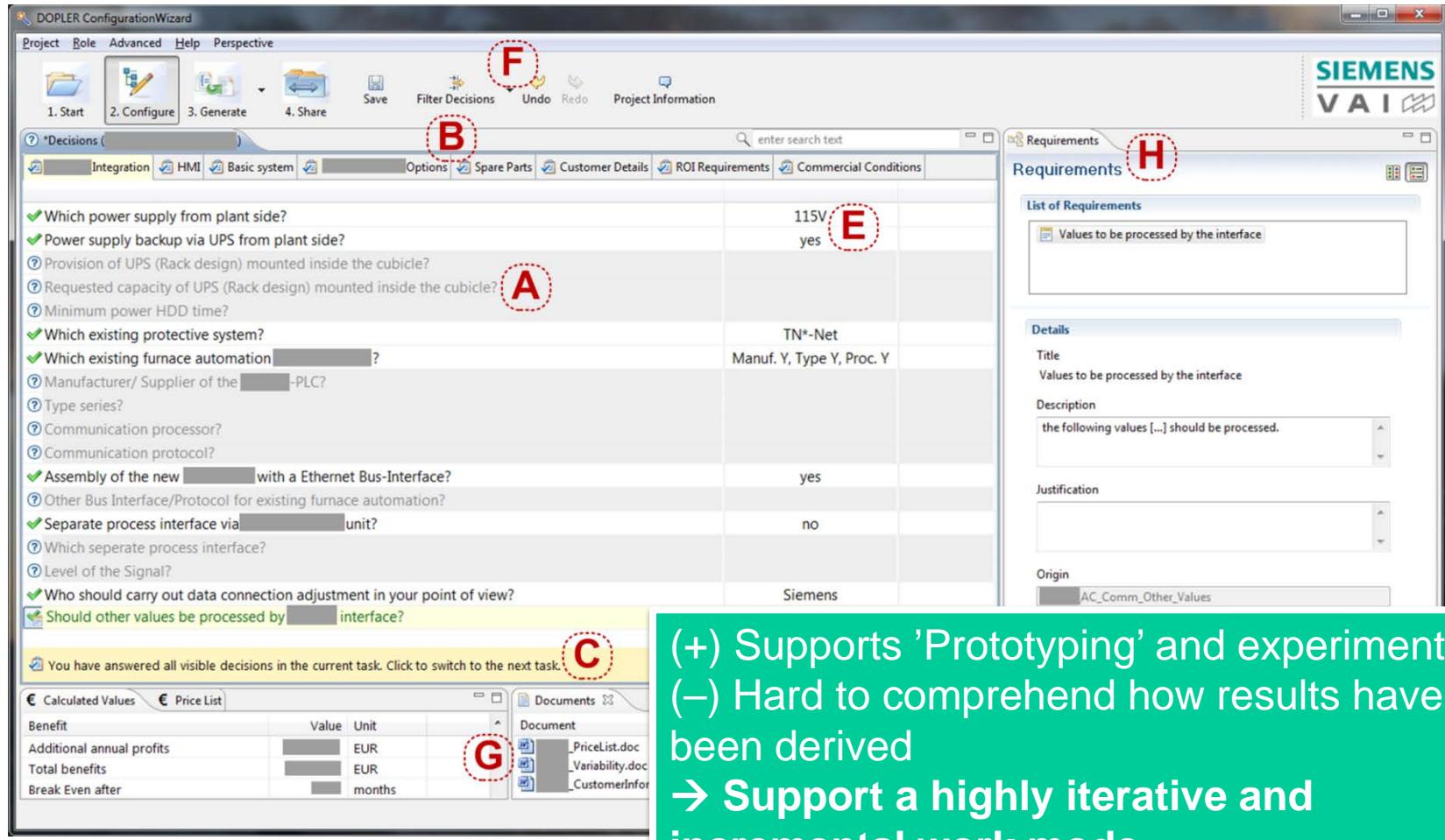
(+) Essential to avoid inconsistent input
 (-) Complex checks are difficult to understand
 → Hide as much as possible
 → Explain the reasons of errors
 → Foster the use of annotations to record difficulties with checks

(F) Reset and undo

The screenshot shows the DOPLER Configuration Wizard application window. The interface includes a top menu bar with Project, Role, Advanced, Help, and Perspective. Below the menu is a toolbar with icons for Start, Configure, Generate, Share, Save, Filter Decisions, Undo (circled F), Redo, and Project Information. The main workspace displays a list of decisions (A) and their answers (B). A status bar at the bottom indicates 'You have answered all visible decisions in the current task. Click to switch to the next task.' (C). On the right, there's a Requirements editor window (D) with sections for List of Requirements, Details, Description, Justification, Origin, Risk level, Priority, and Attachments. A document browser (E) shows files like '_PriceList.doc', '_Variability.doc', and '_CustomerInformation.doc'. A summary table (F) at the bottom left shows calculated values for Benefit, Value, Unit, and Path.

(+) Allows to experiment rapidly with different choices
→ Combine with branching and navigation

(G) Immediate feedback



The screenshot illustrates a software interface for configuration, specifically the DOPLER Configuration Wizard. The interface is divided into several panels:

- Top Bar:** Project, Role, Advanced, Help, Perspective. Buttons include 1. Start, 2. Configure (highlighted with a red circle), 3. Generate, 4. Share, Save, Filter Decisions (highlighted with a red circle), Undo, Redo, and Project Information.
- Decisions Panel (B):** Shows a list of questions with checkboxes. Some are checked (e.g., "Which power supply from plant side?") and others are unanswered (e.g., "Provision of UPS (Rack design) mounted inside the cubicle?"). A red circle highlights the question "Provision of UPS (Rack design) mounted inside the cubicle?" which is circled with a red dotted line.
- Results Panel (E):** Displays the answer "yes" next to the question "115V". A red circle highlights the "yes" response.
- Requirements Panel (H):** Shows a requirement entry form. A red circle highlights the requirement title "Values to be processed by the interface".
- Calculated Values Panel (C):** Shows a table of calculated values. A red circle highlights the status message "You have answered all visible decisions in the current task. Click to switch to the next task." Below the table is a red circle highlighting the "Calculated Values" tab.
- Calculated Values Table:**

Benefit	Value	Unit
Additional annual profits		EUR
Total benefits		EUR
Break Even after		months
- Documents Panel (G):** Shows a list of documents: _PriceList.doc, _Variability.doc, and _CustomerInfor. A red circle highlights the document list.

(+) Supports 'Prototyping' and experimentation
 (-) Hard to comprehend how results have been derived
 → Support a highly iterative and incremental work mode
 → Explain how automated processes work

(H) Annotations and comments

The screenshot shows the DOPLER Configuration Wizard application window. The interface includes a toolbar with icons for Start, Configure, Generate, Share, Save, Filter Decisions, Undo, Redo, and Project Information. A menu bar with Project, Role, Advanced, Help, and Perspective is visible. A search bar at the top right says "enter search text". Below the toolbar is a navigation bar with tabs: Integration, HMI, Basic system, Options (highlighted with red circle B), Spare Parts, Customer Details, ROI Requirements, and Commercial Conditions.

The main area contains a table with rows of questions and their answers. Annotations are circled in red:

- A**: Circles the question "Provision of UPS (Rack design) mounted inside the cubicle?"
- B**: Circles the "Decisions" tab in the navigation bar.
- C**: Circles the status bar message "You have answered all visible decisions in the current task. Click to switch to the next task."
- D**: Circles the "Next" button in the bottom right corner of the main window.
- E**: Circles the value "yes" in the answer column for the question "Power supply backup via UPS from plant side?"
- F**: Circles the "Filter Decisions" icon in the toolbar.
- G**: Circles the "Calculated Values" tab in the bottom left corner.
- H**: Circles the "Requirements" tab in the navigation bar on the right.

On the right side of the screen, there is a separate "Requirements" window open. It has sections for "List of Requirements" (containing "Values to be processed by the interface"), "Details" (with fields for Title, Description, Justification, and Origin), and "Attachments".

(+) Allows dealing with deviations and exceptions
 (-) Purpose of collected annotations and comments unclear
 → Explain how annotations and comments are processed

Derivation in other Tools

▶ Feature-oriented Tools

- Pure::Variants
- Gears
- SPLOT
- Kumbang
- FeatureIDE
- COVAMOF
- VISIT-FC

▶ Workflow-oriented Tools

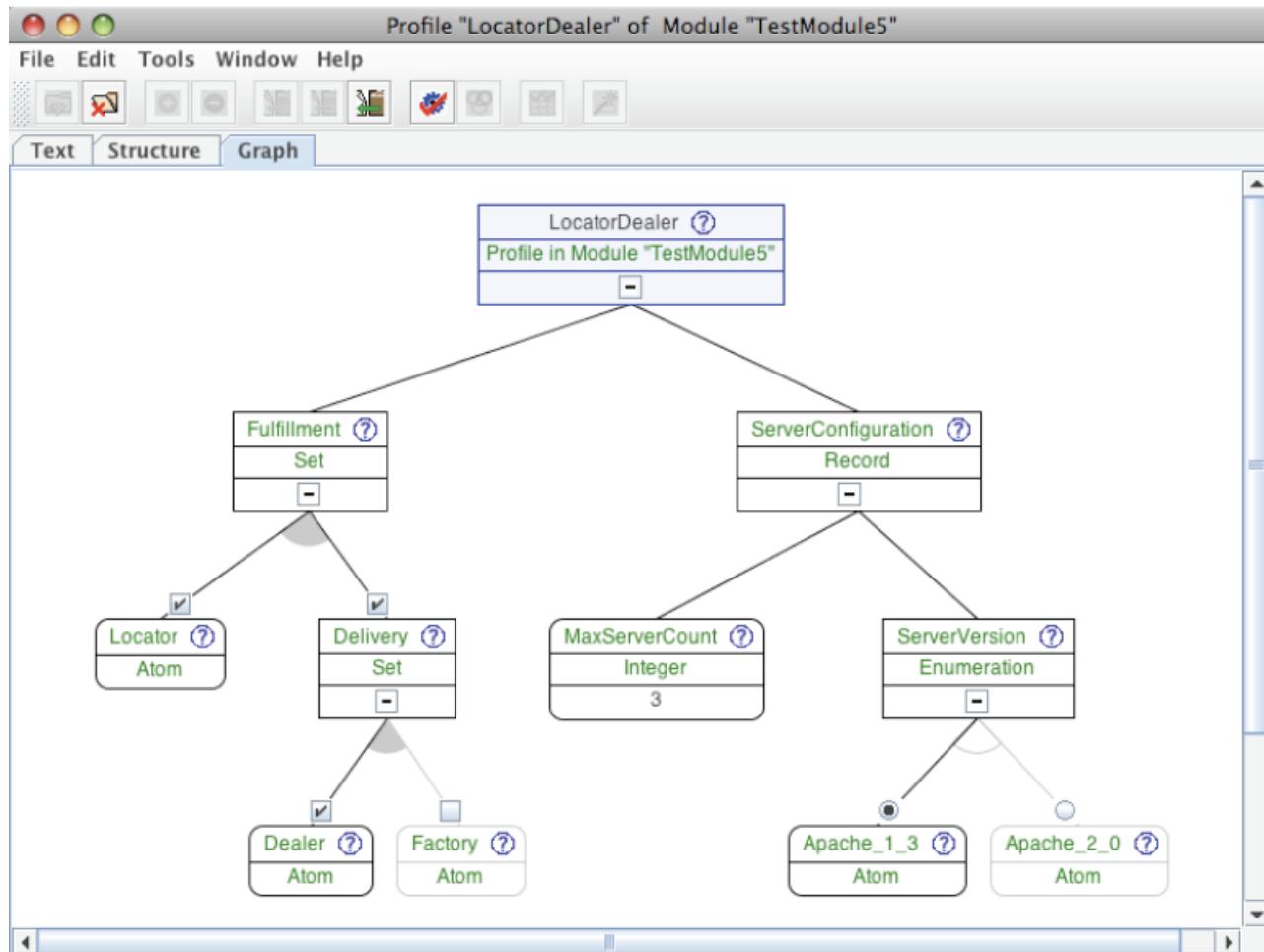
- DOPLER CW
- C2O
- Quaestio
- V-Manage (Resolve)

▶ Knowledge-based Tools

- SAP Configurator
- ILOG Configurator
- Configit Configurator
- SalesPLUS
- ...

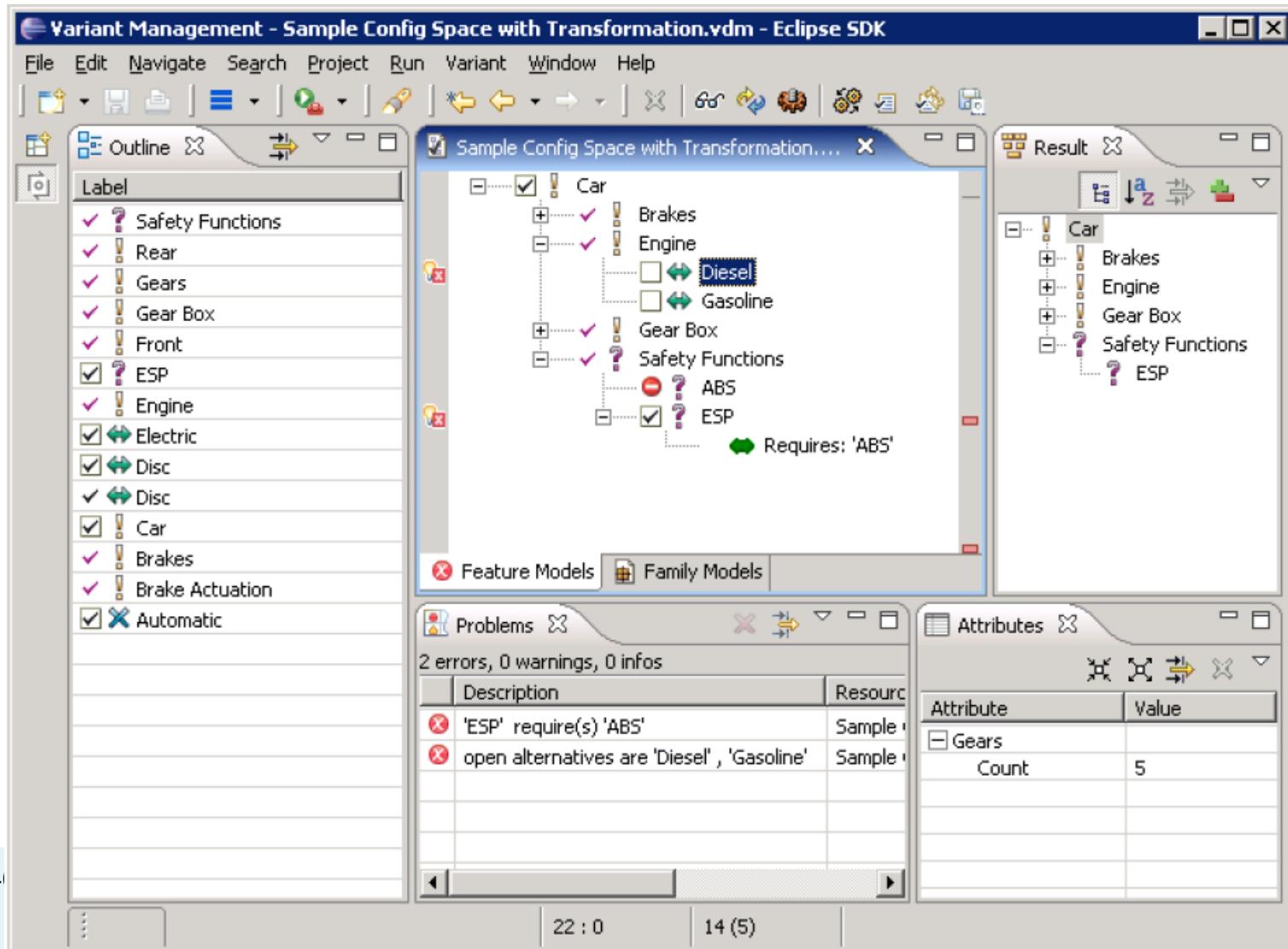
Example: Derivation in GEARS

[Krueger 2007]

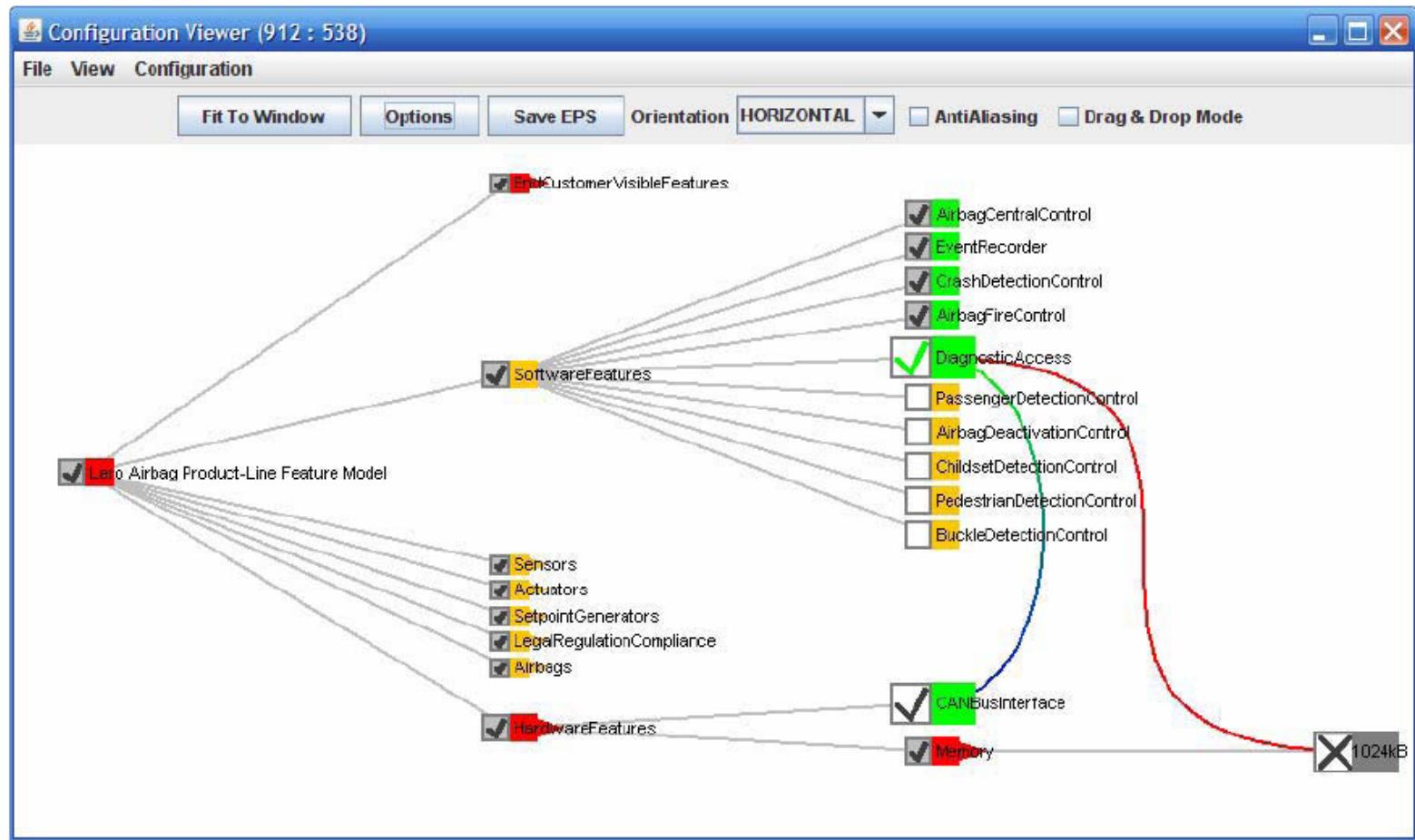


Example: Derivation in pure::variants

© pure systems GmbH (<http://www.pure-systems.com/>)



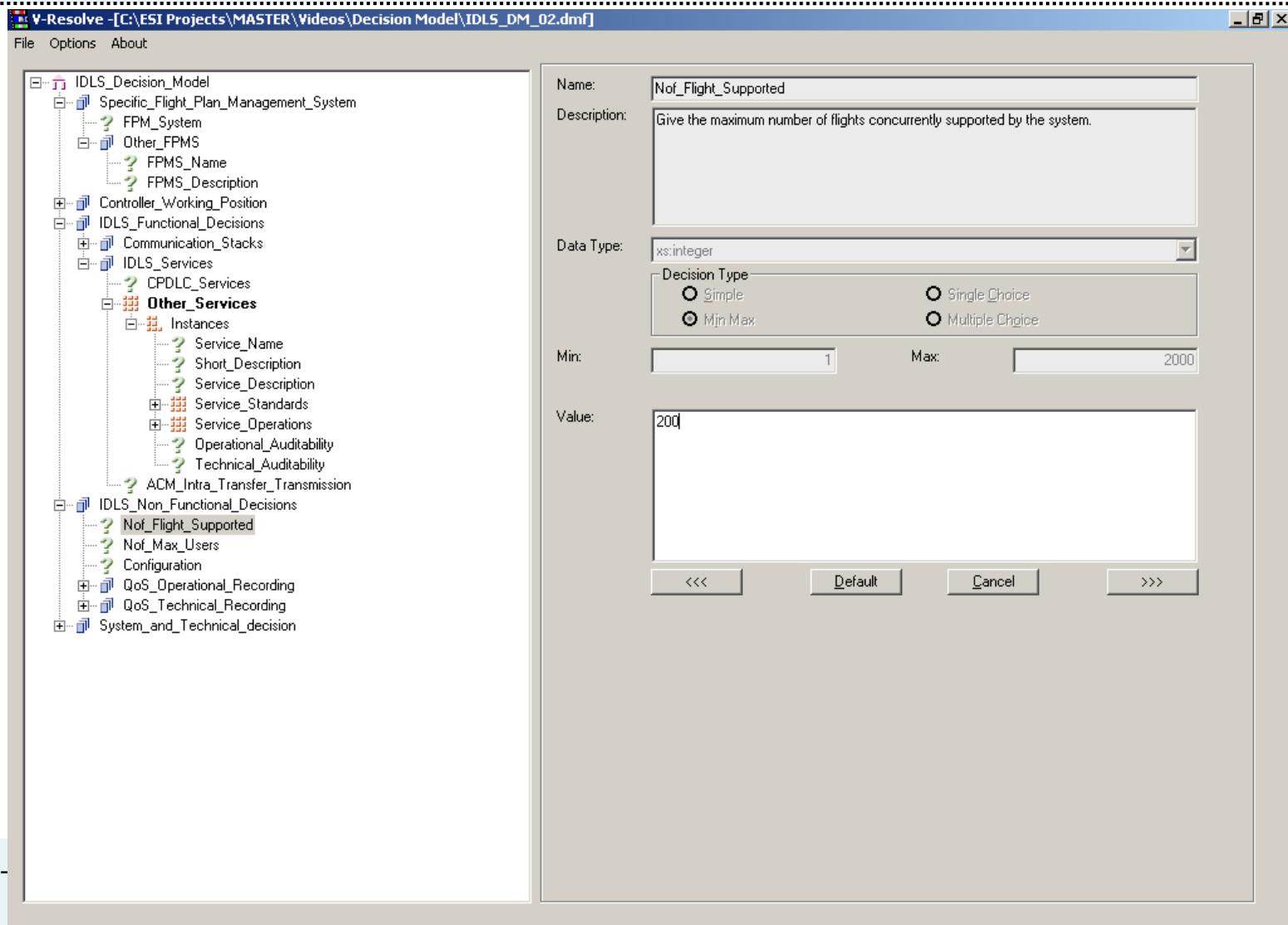
Example: Derivation in Visit-FC



© lero (<http://www.lero.ie>)

Example: Derivation in V-Resolve

© ESI, European Software Institute (<http://modeldrivenarchitecture.esi.es/>)



Exercise

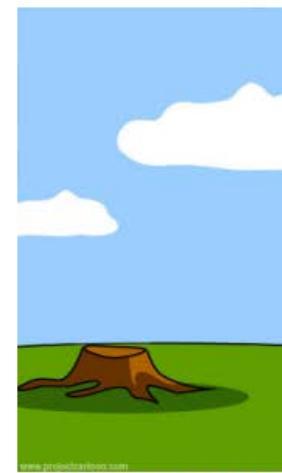
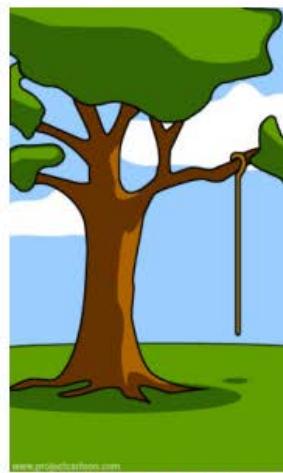
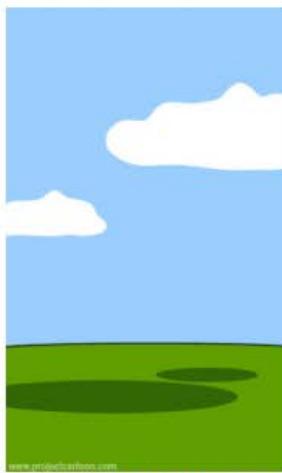
- ▶ Try to represent the full variability described in the Table for 3 products that can be derived in a fictitious mobile phone product line
 - Feature Diagram OR
 - Decision Graph OR
 - Whatever you like
- ▶ Must fit in the space provided on the hand-out (the box)

Functionality	PM_1	PM_2	PM_3
F1: Make and receive calls using GSM 900	X	X	X
F2: Make and receive calls using GSM 1800	X	X	X
F3: Make and receive calls using GSM 1900		X	X
F4: Hold and swap a call	X	X	X
F5: Receive and update voice mail	X	X	X
F6: Display and update time and date	X	X	X
F7: Set alarm and time	X	X	X
F8: Record, display, and manipulate call logs	X	X	X
F9: Play games	X	X	X
F10: Update calendar	X	X	X
F11: Add, delete, and update preferences	X	X	X
F12: Add, delete, and update contacts	X	X	X
F13: Include calculator	X	X	X
F14: Take photos using VGA camera	X		
F15: Take photos using VGA camera with 2x digital zoom		X	
F16: FM radio			X
F17: Email system using SMTP, POP3, or IMPA4	X	X	X
F18: Hand-free speaker	X		X
F19: Send and receive text messages	X	X	X
F20: Send and receive multimedia message	X	X	X
F21: Play RealOne format tunes and video		X	
F22: Play and record MP3 format tunes			X
F23: Record and update video (clips)	X		
F24: Play 3GPP video format		X	X
F25: Play Real Video format		X	
F26: Access Internet using WAP 1.2.1	X		X
F27: Access Internet using WAP 2.0		X	
F28: Access Internet using WAP XHTML		X	X
F29: Connect via Bluetooth transfer data	X	X	X
F30: Connect via Infrared transfer data	X	X	
F31: Connect via USB			X
F32: Play MIDI formatted tunes	X	X	X
F33: Play AMR formatted tunes		X	X
F34: Play AAC formatted tunes			X
F35: Play MP3 formatted tunes			X
F36: Play WAV formatted tunes			X
F37: Play True Tones formatted tunes		X	
F38: Compose and play MIDI formatted ring tones		X	X
F39: Record and update voice messages	X	X	X
F40: Transfer data via SyncML	X		X
F41: Transfer data via SyncML and TCP/IP			X
F42: Support CLDC Java technology	X	X	X
F43: Support MIDP Java technology	X	X	X
F44: Support Wireless messaging API Java technology		X	X
F45: Support Mobile media API Java technology		X	X

What now?

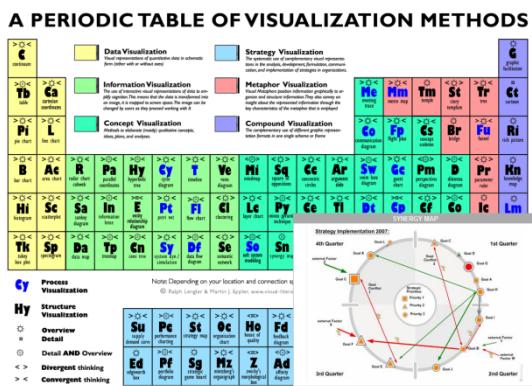
- ▶ Learn from visualization community
- ▶ Learn from product configurators
- ▶ Key issue – mapping requirements to variability
 - will remain → being a good application engineer requires being a good requirements engineer!

Key issue: Product Derivation = Mapping (Customer) Requirements to Variability



PLE vs. Visualization Community

- ▶ Powerful visualizations could improve product derivation – only a few are used currently
- ▶ See for instance:
 - <http://d3js.org/>
 - http://www.visual-literacy.org/periodic_table/



Configurators

- ▶ <http://www.dell.com/us/p/desktops?~ck=mn>
- ▶ <http://www.bmw.com/com/en/general/carconfigurator/content.html>
- ▶ <https://www.volkswagen.co.uk/owners/my/configurations/car-select>
- ▶ <http://configurator.kraftstoff-bikes.com/bikes/at>

- ▶ We should learn from them!

Summary

- ▶ Do not underestimate Product Derivation!
- ▶ Approaches/Tools must be focused on their users and their roles/tasks and must be flexible/adaptable
- ▶ Automate tedious tasks
- ▶ Reduce complexity
- ▶ „Take the user by the hand“, be a requirements engineer
- ▶ Focus on real-world problems

- ▶ *Overall, product line engineering should make **building products easier, not harder!** ;-)*

Next Week (27.4.)

- ▶ PL Tool Support

Used/Useful References

- [Deelstra et al. 2005]
S. Deelstra, M. Sinnema, and J. Bosch, "Product derivation in software product families: a case study," *Journal of Systems and Software*, vol. 74(2), pp. 173-194, 2005.
- [O'Leary 2009]
P. O'Leary, R. Rabiser, I. Richardson, and S. Thiel, "Important Issues and Key Activities in Product Derivation: Experiences from Two Independent Research Projects," *Proc. of the 13th International Software Product Line Conference (SPLC 2009)*, San Francisco, CA, USA, Software Engineering Institute, Carnegie Mellon, 2009, pp. 121-130.
- [Halmans and Pohl 2004]
G. Halmans and K. Pohl, "Communicating the Variability of a Software-Product Family to Customers," *Informatik - Forschung und Entwicklung*, vol. 18(3-4), pp. 113-131, 2004.
- [Nestor et al. 2007]
D. Nestor, L. O'Malley, A. Quigley, E. Sikora, and S. Thiel, "Visualisation of Variability in Software Product Line Engineering," *Proc. of the First International Workshop on Variability Modelling of Software-intensive Systems*, Limerick, Ireland, Lero - Technical Report 2007-01, 2007, pp. 71-78.
- [Antkiewicz and Czarnecki 2004]
M. Antkiewicz and K. Czarnecki, "FeaturePlugin: Feature Modeling Plug-In for Eclipse," *Proc. of the 2004 OOPSLA Workshop on Eclipse Technology eXchange*, Vancouver, Canada, ACM, 2004, pp. 67-72.
- [Cawley et al. 2008]
C. Cawley, D. Nestor, A. Preußner, G. Botterweck, and S. Thiel, "Interactive Visualisation to Support Product Configuration in Software Product Lines," *Proc. of the Second International Workshop on Variability Modeling of Software-Intensive Systems*, Essen, Germany, ICB-Research Report, 2008, pp. 7-16.
- [Krueger 2007]
C. W. Krueger, "The 3-Tiered Methodology: Pragmatic Insights from New Generation Software Product Lines," *Proc. of the 11th International Software Product Line Conference (SPLC 2007)*, Kyoto, Japan, IEEE Computer Society, 2007, pp. 97-106.
- [Diaz et al. 2005]
O. Díaz, S. Trujillo, and F. I. Anfurrutia, "Supporting Production Strategies as Refinements of the Production Process," *Proc. of the 9th International Software Product Line Conference (SPLC 2005)*, Rennes, France, Springer Heidelberg / Berlin, 2005, pp. 210-221.
- [Schmid and John, 2004]
K. Schmid and I. John. *A Customizable Approach to Full-Life Cycle Variability Management*. Journal of the Science of Computer Programming, Special Issue on Variability Management, 53(3), pp. 259-284. 2004.

Used/Useful References

- ▶ [Sinnema and Deelstra 2008]
M. Sinnema and S. Deelstra, "Industrial Validation of COVAMOF," *Journal of Systems and Software*, 2008.
- ▶ [Atkinson et al. 2002]
C. Atkinson, J. Bayer, C. Bunse, E. Kamsties, O. Laitenberger, R. Laqua, D. Muthig, B. Paech, J. Wüst, and J. Zettel, *Component-Based Product Line Engineering with UML*: Addison-Wesley, 2002.
- ▶ [Asikainen et al. 2003]
T. Asikainen, T. Soininen, and T. Männistö, "A Koala-Based Approach for Modelling and Deploying Configurable Software Product Families," *Proc. of the 5th International Workshop on Product-Family Engineering (PFE 2003)*, Siena, Italy, Springer Berlin Heidelberg, 2003, pp. 225-249.
- ▶ [Bayer et al. 2000]
J. Bayer, C. Gacek, D. Muthig, and T. Widen, "PuLSE-I: Deriving Instances from a Product Line Infrastructure," *Proc. of the 7th IEEE International Conference and Workshop on the Engineering of Computer Based Systems (ECBS)*, Edinburgh, Scotland, UK, IEEE Computer Society, 2000, pp. 237-245.
- ▶ [Czarnecki et al. 2004]
K. Czarnecki, S. Helton, and U. W. Eisenecker, "Staged configuration using feature models," *Proc. of the 3rd International Software Product Line Conference (SPLC 2004)*, Boston, MA, USA, Springer Berlin Heidelberg, 2004, pp. 266-283.
- ▶ [O'Leary et al. 2007]
P. O'Leary, M. Ali Babar, S. Thiel, and I. Richardson, "Towards Agile Product Derivation in Software Product Line Engineering," *Proc. of the 4th International Workshop on Rapid Integration of Software Engineering techniques (RISE)*, Luxembourg, 2007.
- ▶ [Rabiser et al. 2007]
R. Rabiser, P. Grünbacher, and D. Dhungana, "Supporting Product Derivation by Adapting and Augmenting Variability Models," *11th International Software Product Line Conference (SPLC 2007)*, Kyoto, Japan, 2007.
- ▶ [Rabiser, 2009]
R. Rabiser. *A User-Centered Approach to Product Configuration in Software Product Line Engineering*. PhD Thesis, Institute for Systems Engineering and Automation, Christian Doppler Laboratory for Automated Software Engineering, Johannes Kepler University, Linz, 2009.
- ▶ [Rabiser, 2010]
R. Rabiser, P. Grünbacher, and D. Dhungana, "Requirements for Product Derivation Support: Results from a Systematic Literature Review and an Expert Survey," *Information and Software Technology*, 52(3), 2010, pp. 324-346.
- ▶ [Ziadi et al. 2003]
T. Ziadi, L. Helouet, and J. M. Jezequel, "Towards a UML Profile for Software Product Lines," *5th International Workshop on Software Product-Family Engineering (PFE 2003)*, Siena, Italy, 2003.