Exercise Academic Skills for Software Engineers

Software Ecosystems

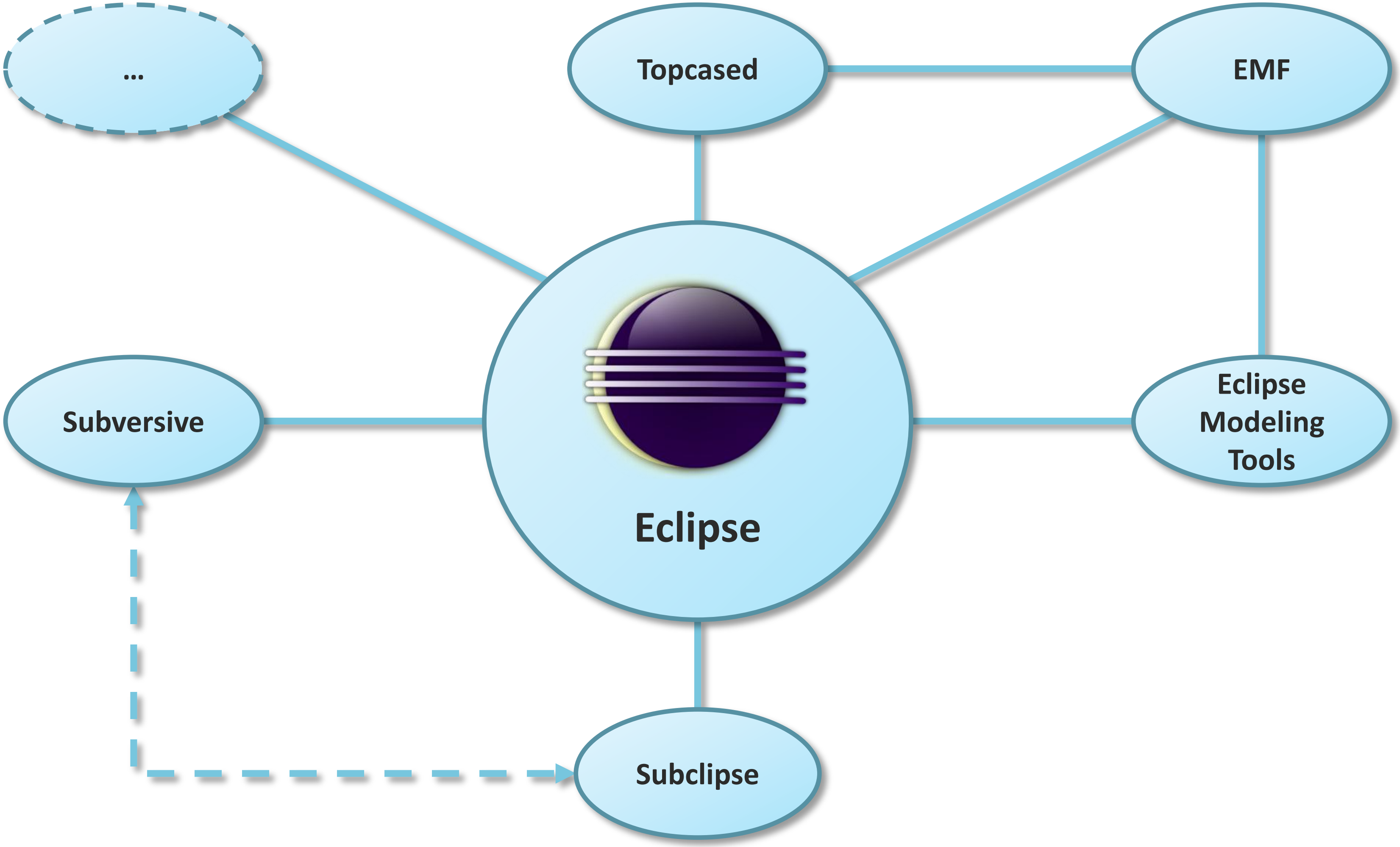
Opening up Software Product Lines

Christoph Seidl

Europa fördert Sachsen. ESF Europäischer Sozialfonds SACHSEN

TECHNISCHE UNIVERSITÄT DRESDEN

Example: Eclipse Software Ecosystem



> Definition: Software Ecosystem

- Definition: Software Ecosystem
  - “… the *set of software solutions* that enable, support and automate the activities and transactions by the actors in the *associated social or business ecosystem* and the *organizations that provide* these solutions.” [Bos09]
  - “... a *software platform*, a set of internal and external *developers and a community* of domain experts ...” [BB09]
  - “… a *collection of software products* that have … *symbiotic relationships* [and] may interact through non-software elements, such as *customers, users, developers, and markets*.” [YRB07]
  - “… the *complete set of entities* with which [an organization] interacts to satisfy its goals.“ [Gre09b]

- Uniform definition has yet to be established!

Similarities with Software Product Lines

- Software products
  - From one family
  - Highly configurable

- Managing variability
  - Shared technological platform/core
  - Different individual variants

- Configuration knowledge about...
  - Dependencies
  - Exclusions
  - Alternatives

Differences to Software Product Lines

	SPL	SECO
Vendors	one (maybe subcontractors)	multiple
Variant Space	closed	open
All Products Known in Advance	yes	no (usually)
Variability Mechanism	positive -or- negative	positive
Extension Mechanism	implicit -or- explicit	explicit
Modularity for Users	invisible	visible
Development	local -or- distributed	distributed
Configuration Knowledge	centralized and explicit	distributed and implicit (usually)

Open Challenges

- **Definition**
  - How to distinguish different types of SECOs or views on them?
  - How to establish a concrete definition for (technical) SECOs?
- **Evolution**
  - How to deal with constant evolution/lack of a particular fixed state of the SECO?
- **Multiple Vendors**
  - How to deal with multiple vendors contributing to the variability model to still maintain configurability?
- **System Properties**
  - How to maintain system properties such as safety under open world assumption?
- **Artifact Heterogeneity**
  - How to deal with a largely heterogeneous set of assets required for configuration?
    - Software: apps, libraries, process parts etc.
    - Hardware: sensors, actuators etc.
- **Cluttering**
  - How to prevent concrete systems from getting cluttered through no longer required artifacts after upgrades?

> Literature

- **[BB09]** Bosch, Bosch-Sijtsema: *From Integration to Composition: On the Impact of Software Product Lines, Global Development and Ecosystems* (2009)
- **[BJB09]** Boucharas, Jansen, Brinkkemper: *Formalizing Software Ecosystem Modeling* (2009)
- **[Bos09]** Bosch: *From Software Product Lines to Software Ecosystems* (2009)
- **[Gre09a]** McGregor: *Ecosystems* (2009)
- **[Gre09b]** McGregor: *Ecosystems, continued* (2009)
- **[Han10]** Hanssen: *Opening up Software Product Line Engineering* (2010)
- **[JFB09]** Jansen, Finkelstein, Brinkkemper - *A Sense of Community - A Research Agenda for Software Ecosystems* (2009)
- **[YRB07]** Yu, Ramaswamy, Bush: *Software Evolvability: An Ecosystem Point of View* (2007)