

# **Software Product Lines**

## **- Basics -**

# Product line



# Product line



Lounge



Pop



ID



By Diesel



Abarth



By Gucci



America



Sport



Edizione  
Maserati



Tributo  
Ferrari

Black  
Jack



Barbie



# Software product line: principles

- ▶ A product(-ion) line applied to software products

The image shows a screenshot of Mozilla's Firefox product line landing page. It features three main sections: "Firefox for desktop » Trusted. Flexible. Fast." with a screenshot of the desktop browser; "Firefox for iOS » Fast. Smart. Yours." with a screenshot of an iPhone displaying the app interface; and "Firefox for Android » Make it uniquely yours." with a screenshot of an Android smartphone displaying the app interface. Below these sections is a horizontal menu with download links for different operating systems and language versions:

Windows	Windows 64-bit	OS X	Linux	Linux 64-bit
Français  Download	 Download	 Download	 Download	 Download

A large red arrow points from the "Windows" section of the menu down towards the "Français" download link.

# Examples



Preferences

**Text Editors**

See '[Colors and Fonts](#)' to configure the font.

Undo history size:

Displayed tab width:

Insert spaces for tabs

Highlight current line

Show print margin

Print margin column:

Show line numbers

Show range indicator

Show whitespace characters ([configure visibility](#))

Show affordance in hover on how to make it sticky

When mouse moved into hover:

Enable drag and drop of text

Warn before editing a derived file

Smart caret positioning at line start and end

Appearance color options:

Line number foreground  
Current line highlight  
Print margin  
Find scope  
Selection foreground color  
Selection background color  
Background color  
Foreground color  
Hyperlink

Color:

More colors can be configured on the '[Colors and Fonts](#)' preference page.

# Examples

---

## Compare Windows 10 editions

	Home	Pro	Enterprise	Education	Mobile	Mobile Enterprise
Cortana <sup>1</sup>	✓	✓	✓		✓	✓
Windows Ink <sup>3</sup>	✓	✓	✓	✓		
Continuum for Phones <sup>4</sup>	✓	✓	✓	✓	✓	✓
Start Menu and Live Tiles	✓	✓	✓	✓	✓	
Tablet Mode	✓	✓	✓	✓	✓	
Pen <sup>5</sup>	✓	✓	✓	✓		
Voice, touch and gesture	✓	✓	✓	✓	✓	✓
Microsoft Edge (+ reading view PDF reader)	✓	✓	✓	✓	✓	✓
Remote Desktop	✓	✓	✓	✓	✓	✓

# Main characteristics

---

## **Mass customization**

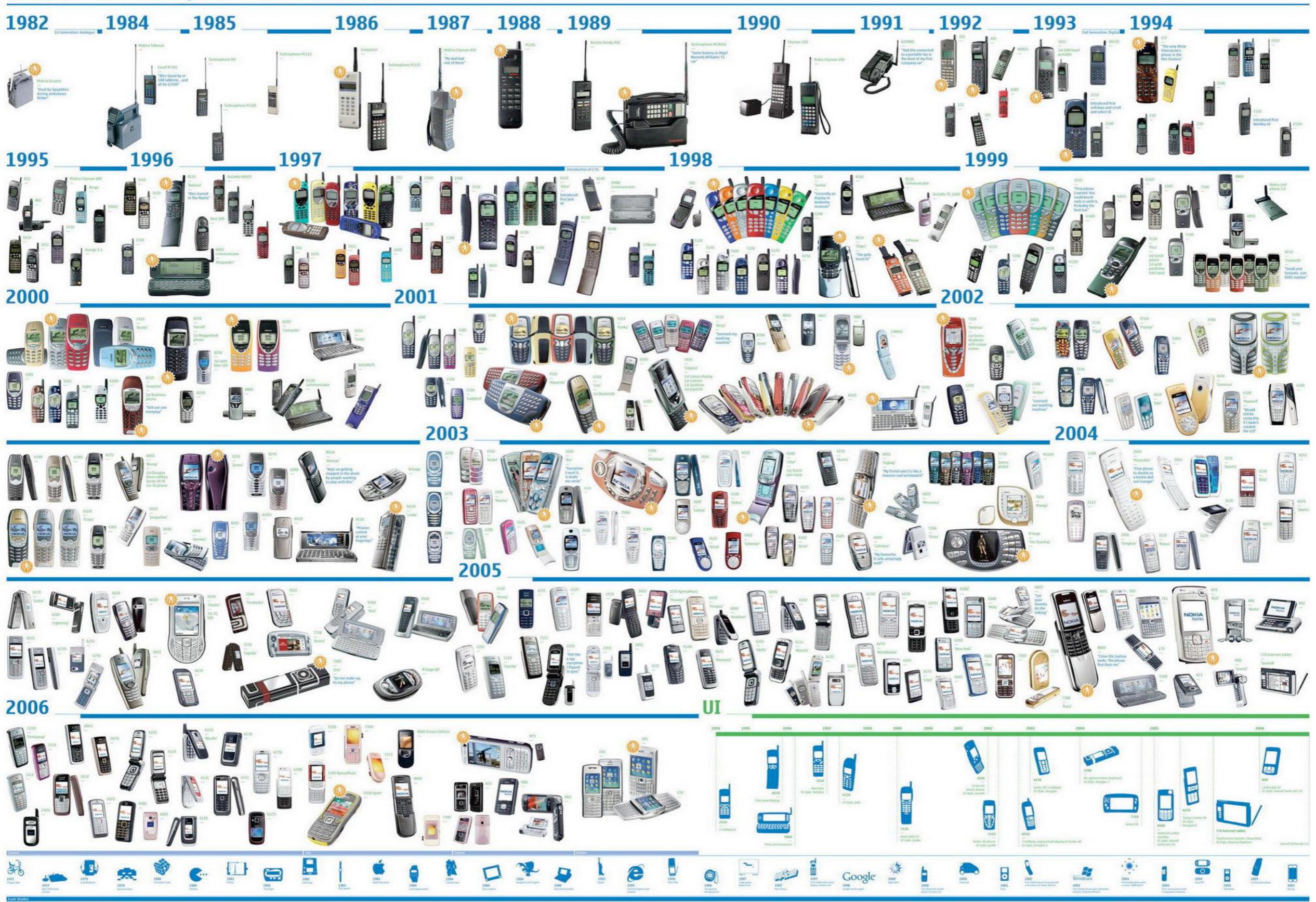
Large-scale production of goods tailored to individual customers' needs

## **Software reuse**

# Mass customization

 **Huawei** Connecting People

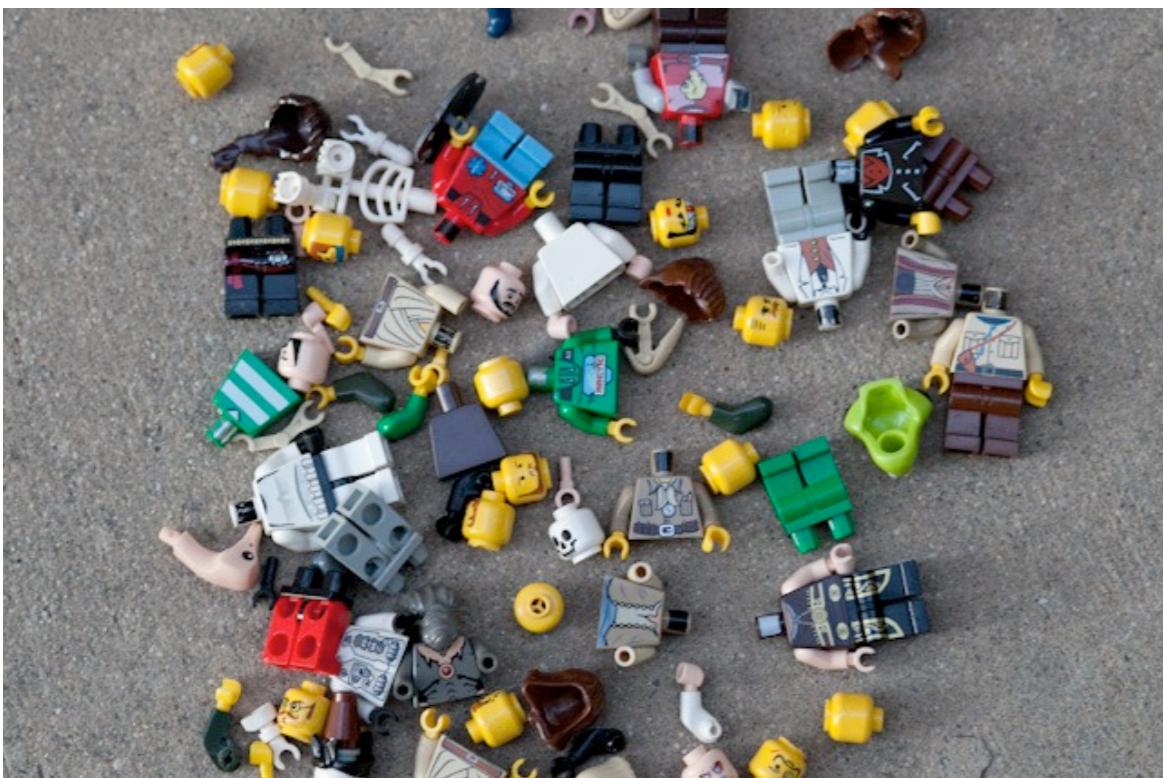
**Know our past. Create the future...**



# Software reuse

---

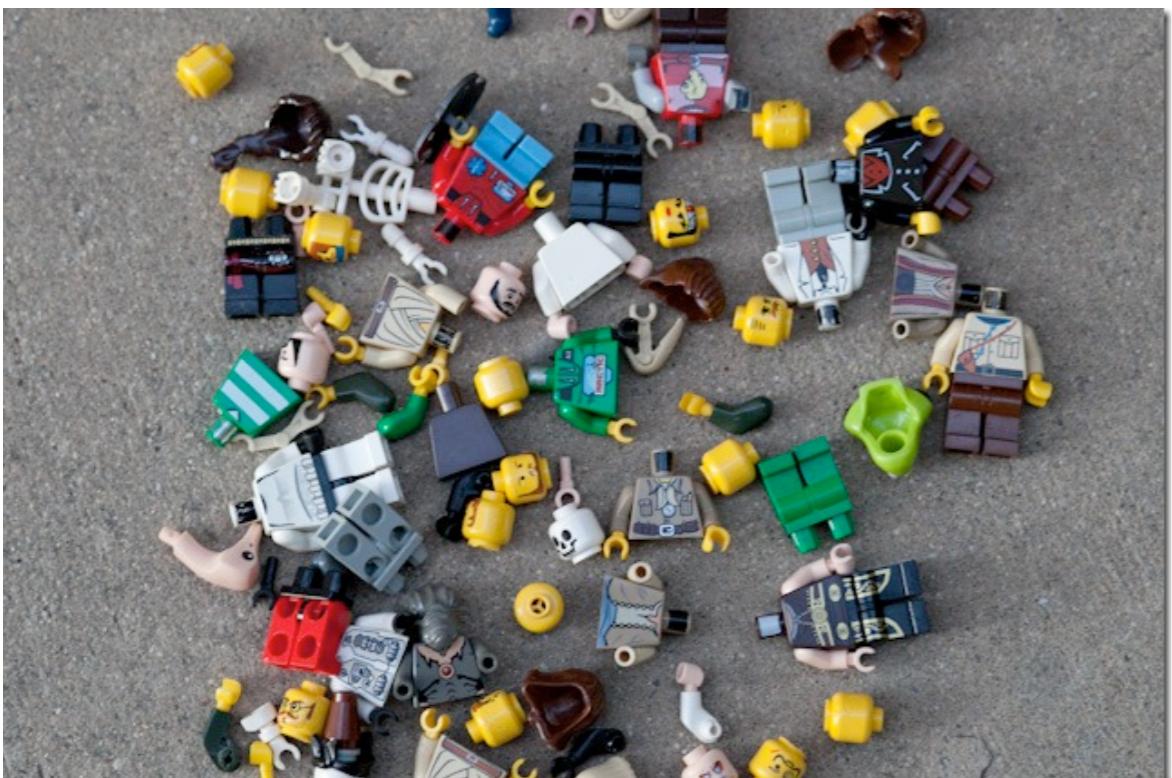
- ▶ Creating software systems from existing software rather than building software systems from scratch



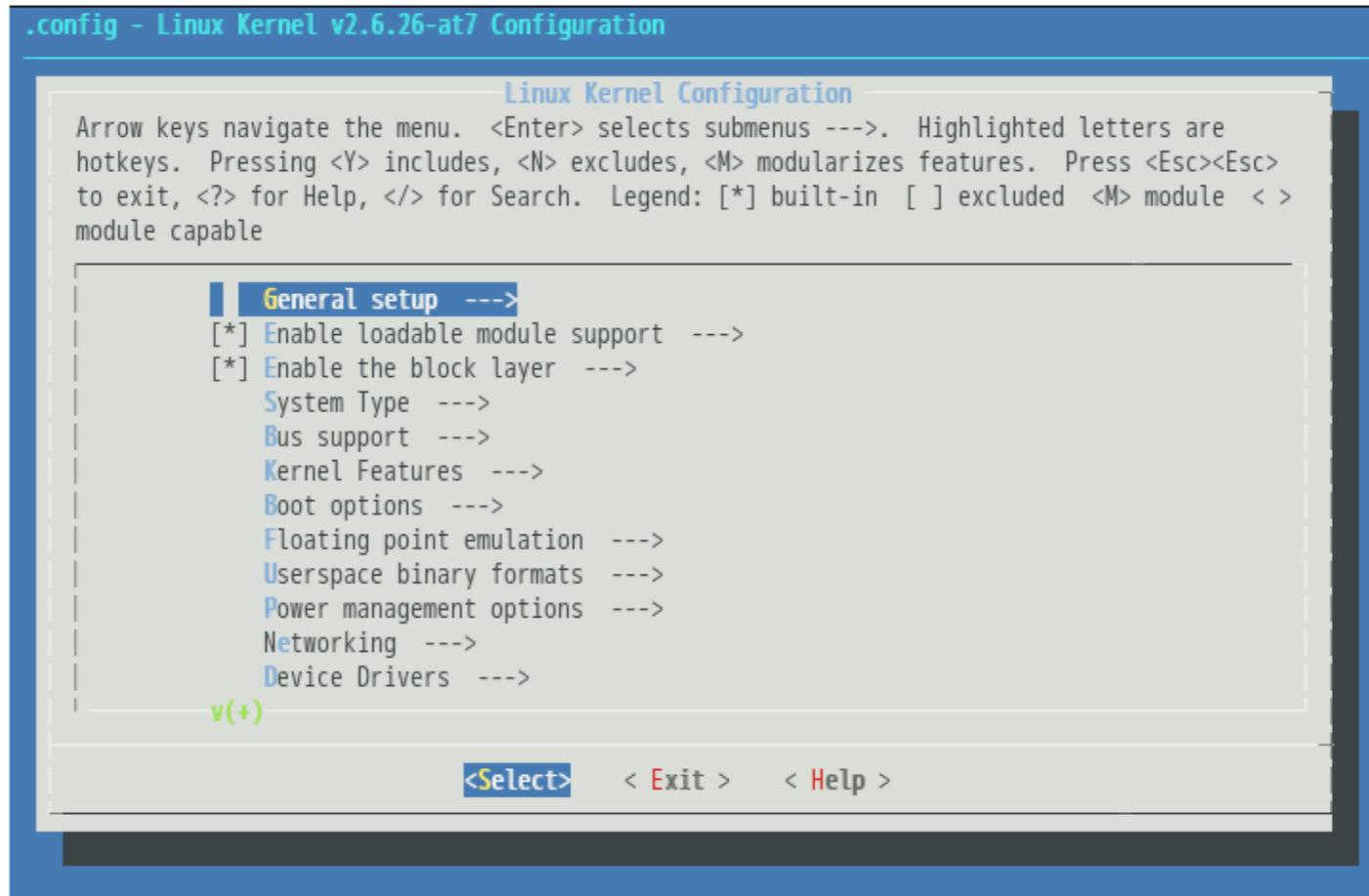
# Software reuse

---

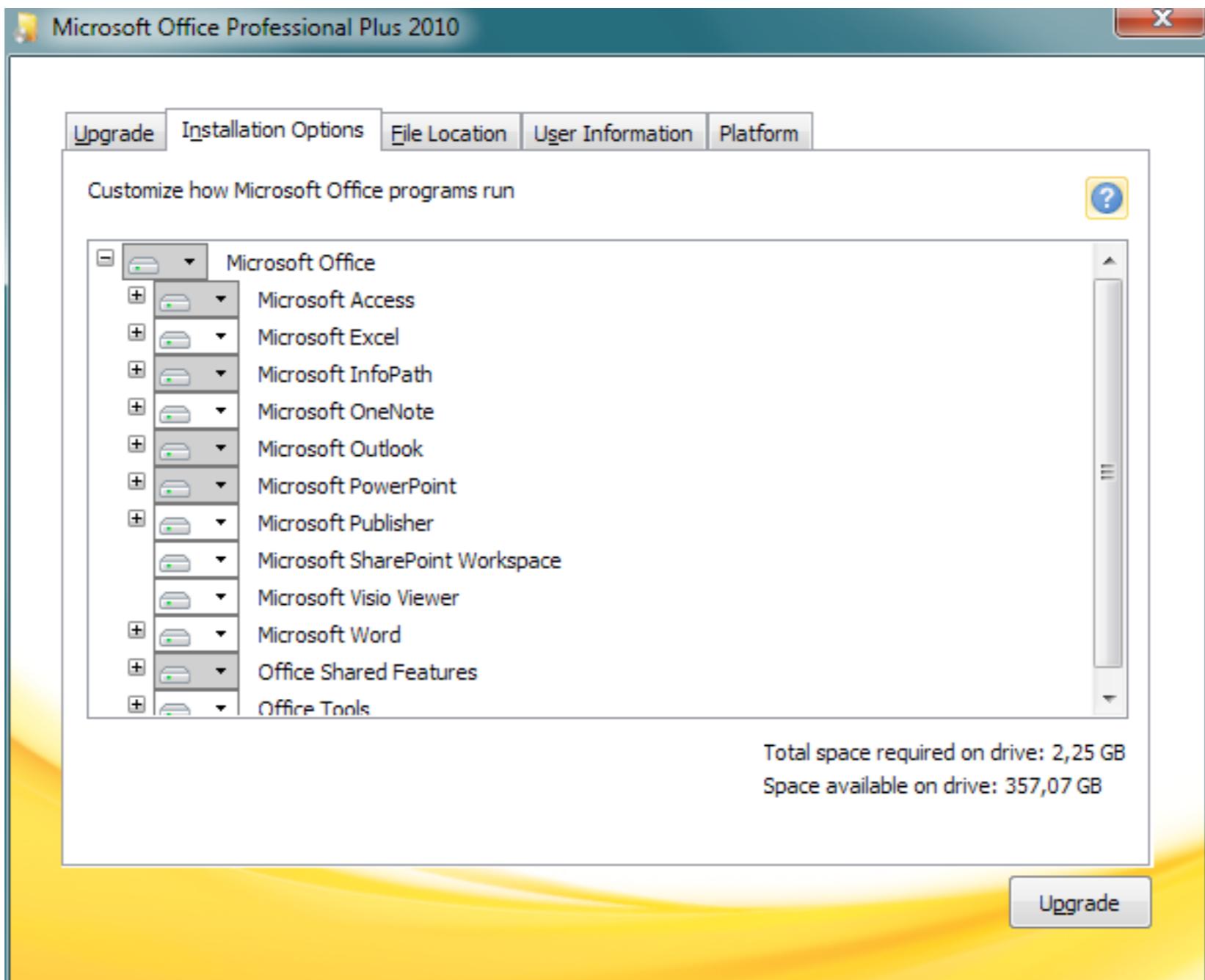
- ▶ Software product lines make software reuse easier since developed products are from the **same family**



# Examples



# Examples



FAMILY!

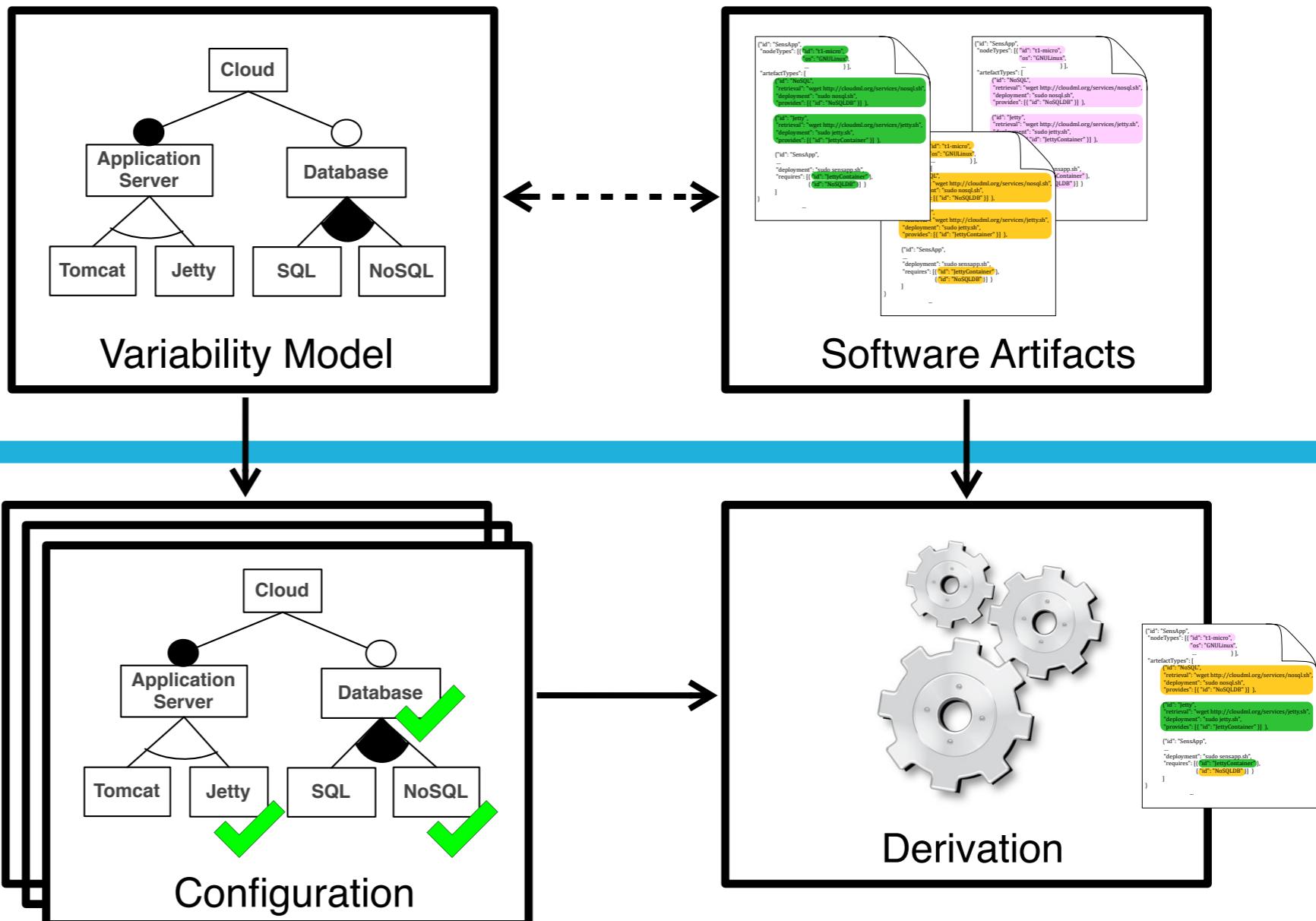
“A set of programs is considered to constitute a **family**, whenever it is worthwhile to study programs from the set by first studying the **common properties** of the set and then determining the **special properties** of the individual family members”

David L. Parnas, On the design and development of program families  
in Transactions on Software Engineering, SE-2(1):1–9, 1976

- ▶ Software product line engineering
  - Factoring out **commonalities** for **reuse**
  - Managing **variability** for software **mass customization**

# Software product line engineering

Domain  
Engineering

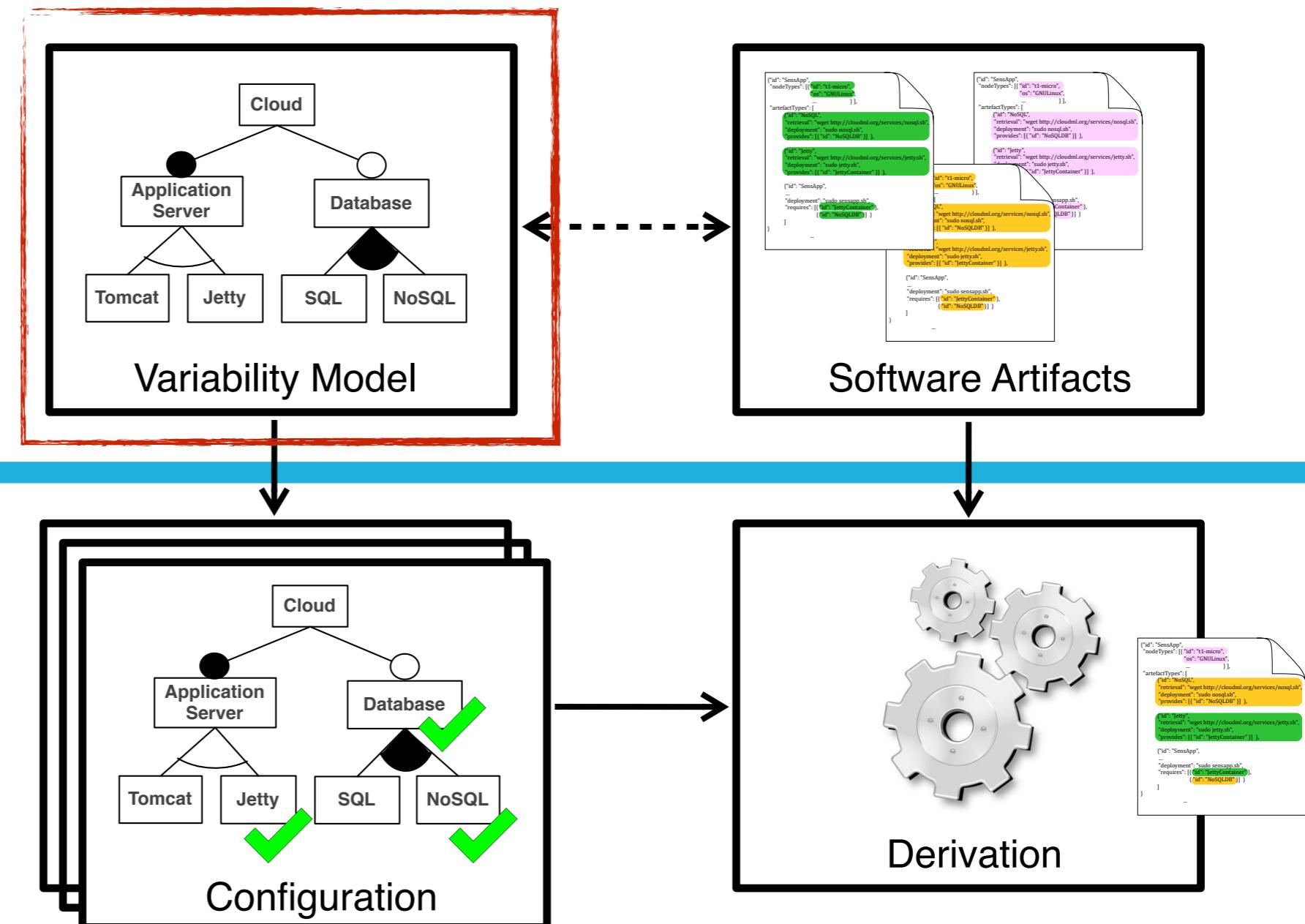


“The investments required to develop the reusable artefacts during **domain engineering**, are outweighed by the benefits of deriving the individual products during **application engineering**”

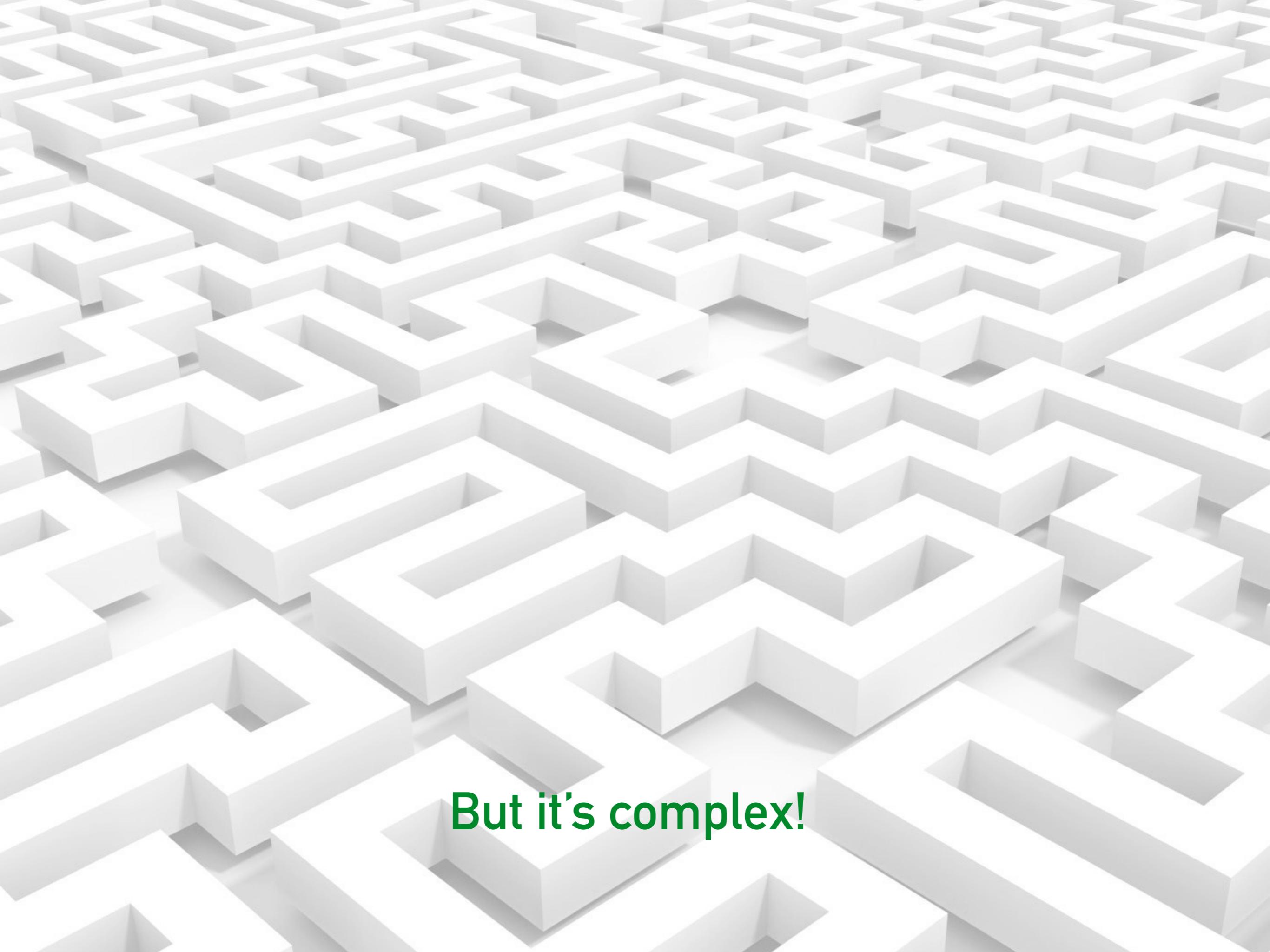
Jan Bosch et al. 2004

# Software product line engineering

Domain  
Engineering



Managing variability is the key concern in SPLE

A large, intricate 3D white maze is shown from a top-down perspective, filling the entire frame. The maze consists of many interconnected paths and dead ends, creating a complex and challenging-looking puzzle.

**But it's complex!**

# 33 optional, independent features



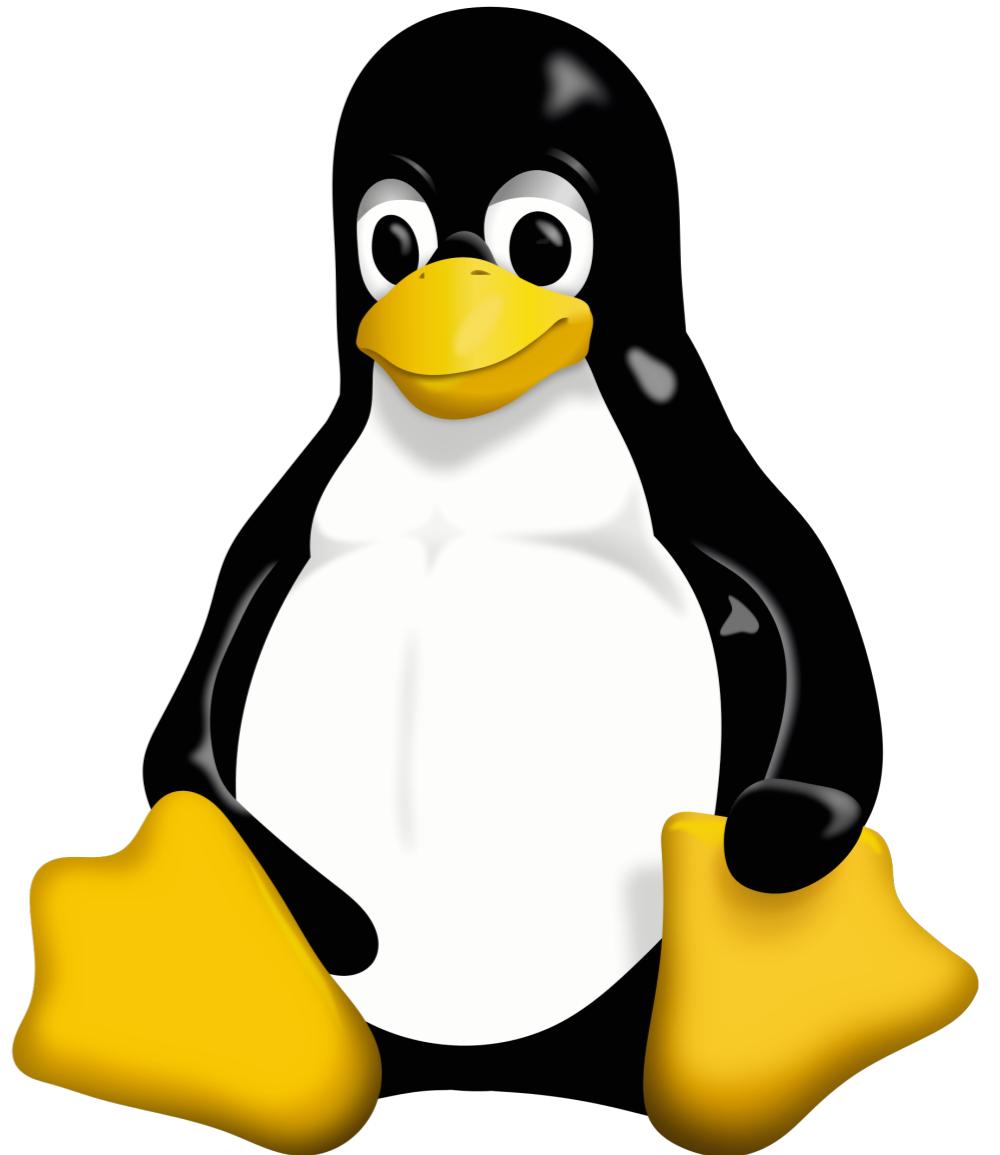
a unique variant for every  
person on this planet



## Printer firmware



2 000 features



12 000+ features

# Variability Management

---

- ▶ Variability management is crucial
- ▶ Need for
  - Clear syntax and semantics
  - Support for automated reasoning
- Feature models
  - *De facto* standard for describing the variability of a domain
  - Meet the requirements above

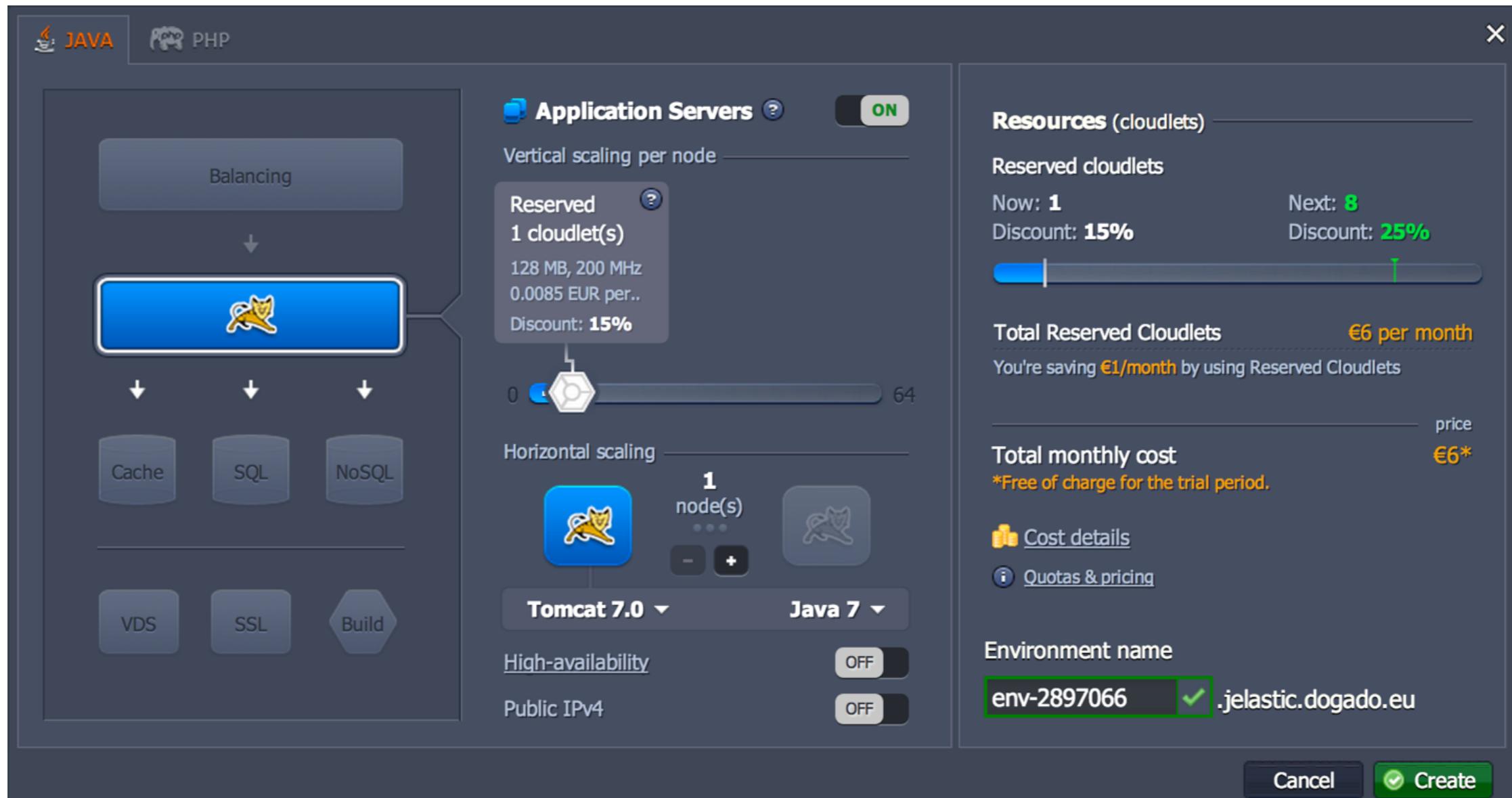
# Features

---

- ▶ What is a feature?
  - End-user visible behavior or property of a system
  - That may be optional or may have alternatives
  
- ▶ Features represent commonalities and variabilities of software systems



# Feature modeling: example



# Feature modeling: example

The screenshot shows the Jelastic application builder interface for a Java application. The left sidebar has tabs for JAVA and PHP, with JAVA selected. The main area displays a feature model diagram:

- A central node labeled with a cat icon is connected to three nodes below it: Cache, SQL, and NoSQL.
- These three nodes are connected to a single node at the bottom labeled VDS.
- From the VDS node, arrows point to three nodes: SSL, Build, and another VDS node.
- An arrow also points from the original VDS node to the bottom-right corner of the screen.

On the right side, the configuration panel includes the following sections:

- Application Servers**: A switch is set to **ON**. Below it, "Vertical scaling per node" is set to **1 cloudlet(s)**, with details: **128 MB, 200 MHz**, **0.0085 EUR per..**, and a **Discount: 15%**. A slider shows **0** to **64**.
- Horizontal scaling**: Shows **1 node(s)** with a minus and plus button, and a preview icon of a cat.
- Tomcat 7.0** and **Java 7** dropdown menus.
- High-availability** and **Public IPv4** toggle switches, both set to **OFF**.
- Resources (cloudlets)**: Shows **Reserved** **1 cloudlet(s)** with a **Discount: 15%** applied.
- Total Reserved Cloudlets**: Shows **Now: 1** and **Discount: 15%**.
- Total monthly cost**: Shows **You're saving €1/month b...** and **\*Free of charge for the first month**.
- Cost details** and **Quotas & pricing** links.
- Environment name**: Set to **env-2897066**.



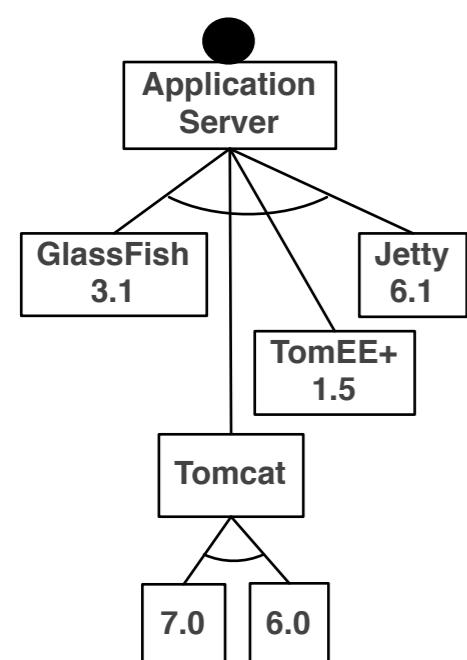
# Feature modeling: example

The screenshot shows the Jelastic cloud application builder interface. On the left, a sidebar has a highlighted 'Application Server' section. The main area displays a Java application stack diagram:

- Java tab:** JAVA icon.
- PHP tab:** PHP icon.
- Application Servers:** A blue box containing a cat icon, labeled 'Balancing' above it. It is connected via arrows to three smaller boxes below: 'Cache', 'SQL', and 'NoSQL'. To the right of this box are settings for 'Vertical scaling per node':
  - Reserved:** 1 cloudlet(s), 128 MB, 200 MHz, 0.0085 EUR per.., Discount: 15%.
  - A slider bar set at 0, ranging from 0 to 64.
- Horizontal scaling:** Shows a blue box with a cat icon and a grey box with a cat icon, both connected by a line. Below them are dropdown menus for 'Tomcat 7.0' and 'Java 7'.
- Cloudlet features:** High-availability (OFF) and Public IPv4 (OFF).
- Resources (cloudlets):** Reserved cloudlets: Now: 1, Discount: 15%.
- Total Reserved Cloud:** You're saving €1/month.
- Total monthly cost:** \*Free of charge for the trial.
- Cost details:** Link to cost details.
- Quotas & pricing:** Link to quotas & pricing.
- Environment name:** env-2897066.

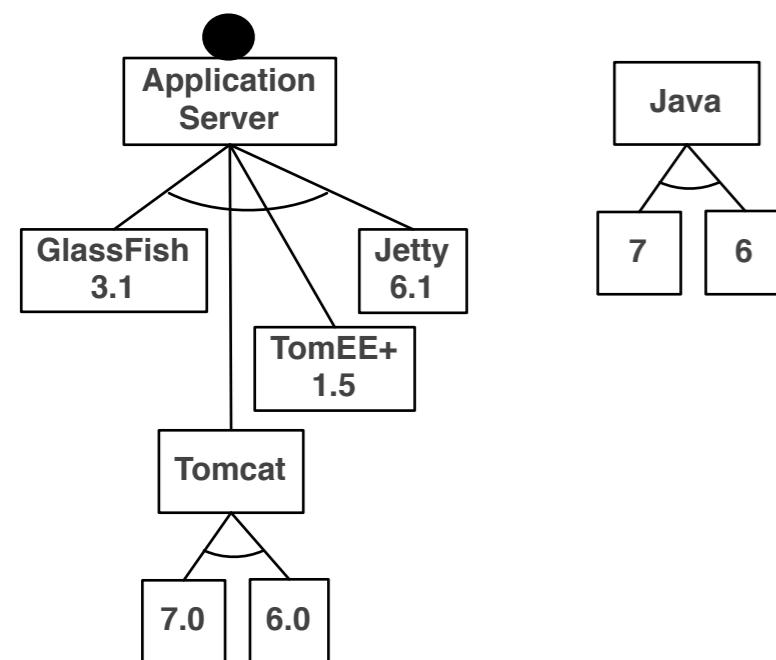


# Feature modeling: example



The screenshot shows the Jelastic cloud interface for configuring an application server. On the left, there are tabs for "JAVA" (selected) and "PHP". In the center, a large box contains a "Balancing" section with a dropdown menu showing "Tomcat 7.0" and "Java 7". Below this are several other options: "GlassFish 3.1", "Jetty 6.1", "TomEE+ 1.5", "Tomcat 6.0", and "Tomcat 7.0". At the bottom of this central box are buttons for "VDS", "SSL", and "Build". To the right of the central box, there are sections for "Application Servers" (with a switch set to "ON"), "Vertical scaling per node" (showing 1 reserved cloudlet at 128 MB, 200 MHz, 0.0085 EUR per month, and a 15% discount), and "Horizontal scaling" (showing 1 node selected). At the bottom right are buttons for "High-availability" and "Public IPv4", both currently set to "OFF". On the far right, there are sections for "Resources (cloudlets)", "Reserved cloudlets" (1 reserved, 15% discount), "Total Reserved Cloud", "Total monthly cost" (\*Free of charge for the trial), "Cost details", "Quotas & pricing", and "Environment name" (set to "env-2897066").

# Feature modeling: example



The screenshot shows the Jelastic cloud management interface. On the left, there are tabs for 'JAVA' and 'PHP'. The 'JAVA' tab is selected. In the center, there's a 'Balancing' section with a dropdown menu set to 'Tomcat 7.0' and 'Java 7'. This section lists several Java application servers: GlassFish 3.1, Jetty 6.1, TomEE+ 1.5, Tomcat 6.0, and Tomcat 7.0. Below the servers are buttons for 'VDS', 'SSL', and 'Build'. To the right, there's a 'Application Servers' section with a switch labeled 'ON'. It shows 'Vertical scaling per node' with '1 cloudlet(s)', '128 MB, 200 MHz', '0.0085 EUR per..', and a 'Discount: 15%'. It also shows 'Horizontal scaling' with '1 node(s)' and a '+' button. At the bottom, there are switches for 'High-availability' and 'Public IPv4', both set to 'OFF'. On the far right, there are sections for 'Resources (cloudlets)', 'Reserved cloudlets', 'Total Reserved Cloud...', 'Total monthly cost', 'Cost details', 'Quotas & pricing', and 'Environment name' (set to 'env-2897066').

# Feature modeling: example

The screenshot shows the Jelastic cloud platform interface for managing Java application servers. The top navigation bar has tabs for JAVA and PHP, with JAVA selected. On the left, there's a sidebar with a 'Balancing' section and a main panel for 'Application Servers'. The main panel shows a summary of reserved resources: 1 cloudlet(s) with 128 MB, 200 MHz, and a 0.0085 EUR per month discount of 15%. It also shows horizontal scaling options with 1 node(s). Below this, there are sections for 'High-availability' and 'Public IPv4', both currently set to OFF. On the right, there are sections for 'Resources (cloudlets)', 'Reserved cloudlets', and 'Cost details'. The environment name is listed as 'env-2897066'.

**Application Servers**  **ON**

Vertical scaling per node

Reserved  1 cloudlet(s)  
128 MB, 200 MHz  
0.0085 EUR per..  
Discount: 15%

Horizontal scaling

1 node(s)

High-availability  OFF

Public IPv4  OFF

**Resources (cloudlets)**

Reserved cloudlets

Now: 1  
Discount: 15%

Total Reserved Cloudlets

You're saving €1/month b...

Total monthly cost  
\*Free of charge for the first 100 hours

[Cost details](#)

[Quotas & pricing](#)

Environment name

env-2897066



# Feature modeling: example

The screenshot shows the Jelastic application builder interface for Java applications. The left sidebar has tabs for JAVA and PHP, with JAVA selected. The main area displays a stack diagram for a Java application:

- Balancing:** A blue box containing a cat icon.
- Application Servers:** A section with a switch labeled "ON". It shows "Vertical scaling per node" with 1 reserved cloudlet (128 MB, 200 MHz, 0.0085 EUR per..) and a discount of 15%. A slider is set at 0, and a progress bar shows 64.
- Horizontal scaling:** Shows 1 node(s) with a cat icon and a plus/minus button.
- Tomcat 7.0** and **Java 7** dropdown menus.
- High-availability:** OFF switch.
- Public IPv4:** OFF switch.

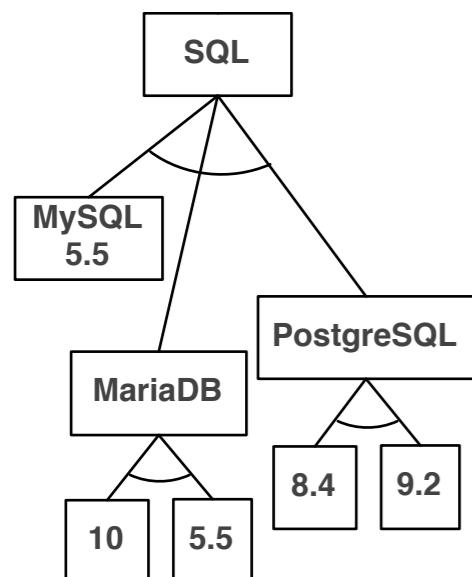
The stack diagram includes the following components:

- Balancing:** A blue box with a downward arrow pointing to the Application Server.
- Application Server:** A blue box with a cat icon, connected to the Balancing layer.
- Database:** Three cylinders labeled Cache, SQL, and NoSQL. The SQL cylinder is highlighted with a red border.
- Supporting Services:** VDS, SSL, and Build.

On the right side of the interface, there are sections for **Resources (cloudlets)**, **Reserved cloudlets** (Now: 1, Discount: 15%), **Total Reserved Cloudlets**, **Total monthly cost** (\*Free of charge for the trial), **Cost details**, **Quotas & pricing**, and **Environment name** (env-2897066).

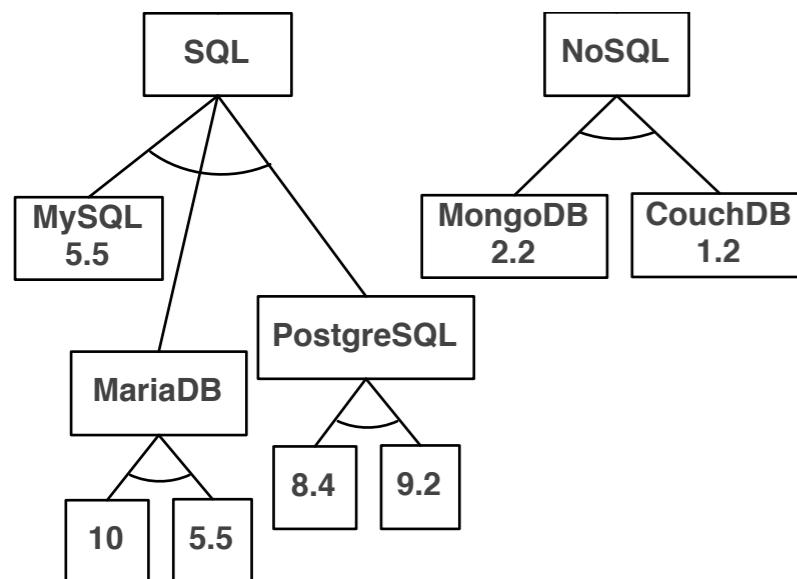


# Feature modeling: example



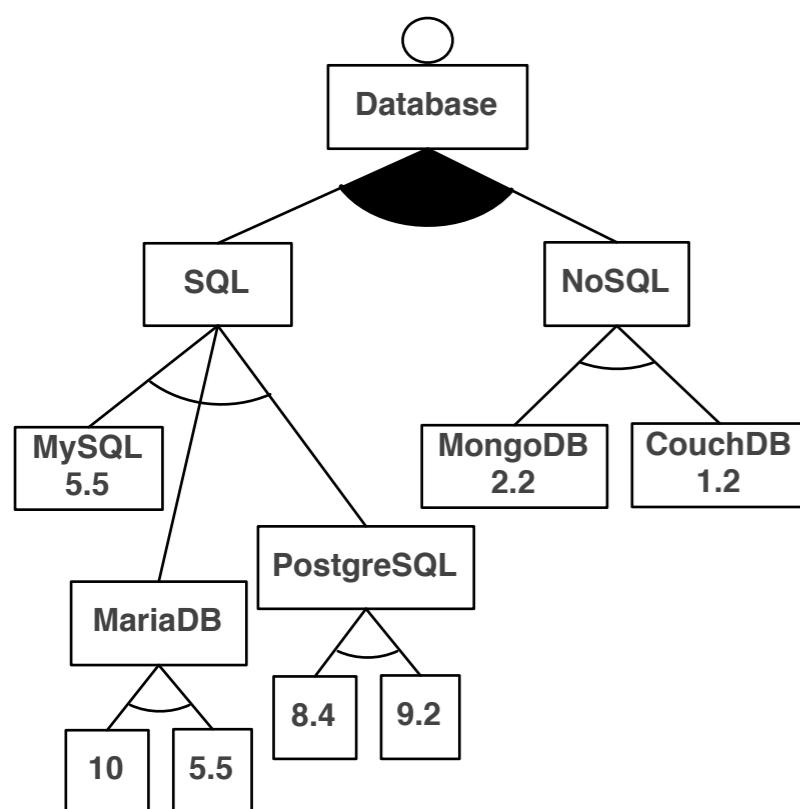
The screenshot shows the Jelastic cloud interface. On the left, there's a navigation bar with "JAVA" and "PHP" tabs. The main area displays a configuration for an "Application Server". A central icon is a yellow cat-like logo. Below it are three smaller icons: "Cache", "NoSQL", and "VDS". To the right of the cat icon is a dropdown menu for "PostgreSQL 8.4" which lists "MariaDB 10.0", "MariaDB 5.5", "MySQL 5.5", "PostgreSQL 8.4", and "PostgreSQL 9.2". Above the cat icon is a "Balancing" section. To the right of the cat icon is a "Resources (cloudlets)" panel showing "Reserved cloudlets: Now: 1, Discount: 15%", a progress bar, and "Total Reserved Cloudlets: You're saving €1/month". Below the resources panel are sections for "Horizontal scaling" (with a "1 node(s)" button), "High-availability" (with an "OFF" switch), and "Public IPv4" (with an "OFF" switch). On the far right, there are links for "Cost details", "Quotas & pricing", and an "Environment name" field containing "env-2897066".

# Feature modeling: example



The screenshot shows the Jelastic cloud management interface. At the top, there are tabs for 'JAVA' and 'PHP'. Below the tabs, there's a 'Balancing' section with a 'Cat' icon. To the right, under 'Application Servers', it says 'Vertical scaling per node' is turned 'ON'. It shows '1 cloudlet(s)' reserved, with details: '128 MB, 200 MHz, 0.0085 EUR per..', 'Discount: 15%', and a slider set at 0 out of 64. Below this, 'Horizontal scaling' is shown with '1 node(s)'. Under 'MongoDB 2.2', there are icons for 'Cache' and 'VDS'. At the bottom, there are sections for 'High-availability' (OFF) and 'Public IPv4' (OFF). On the right side, there are sections for 'Resources (cloudlets)', 'Reserved cloudlets', 'Total Reserved Cloud...', 'Total monthly cost', 'Cost details', 'Quotas & pricing', and 'Environment name' (set to 'env-2897066').

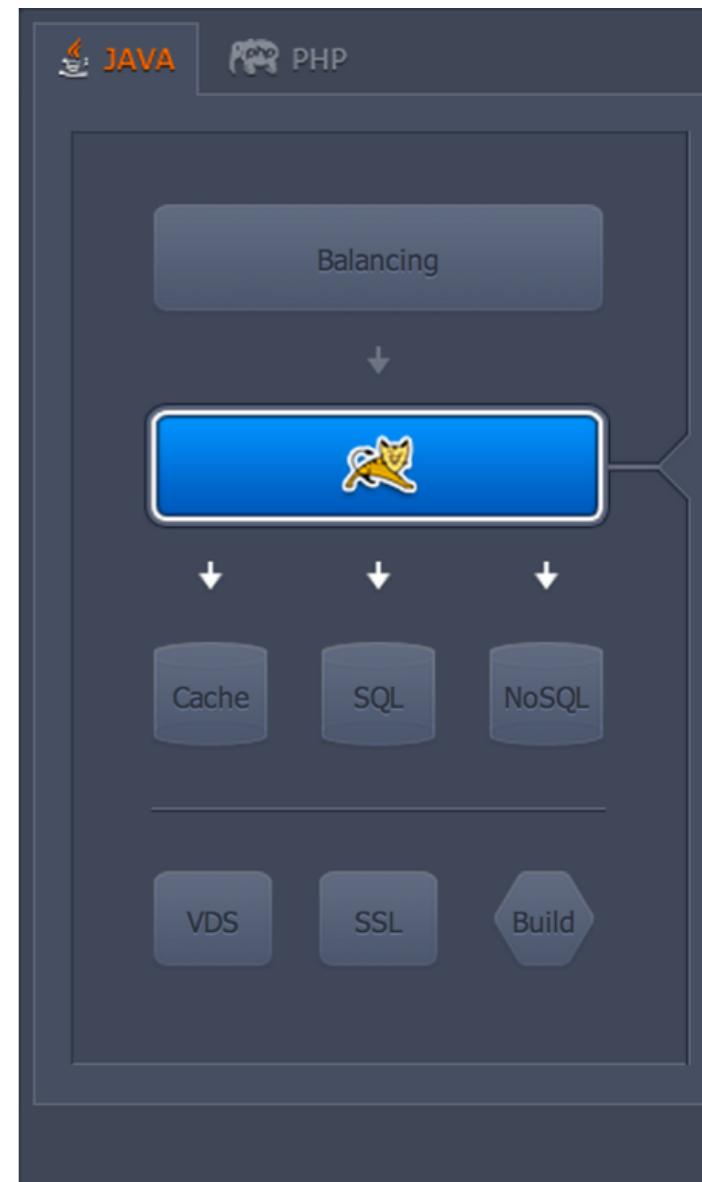
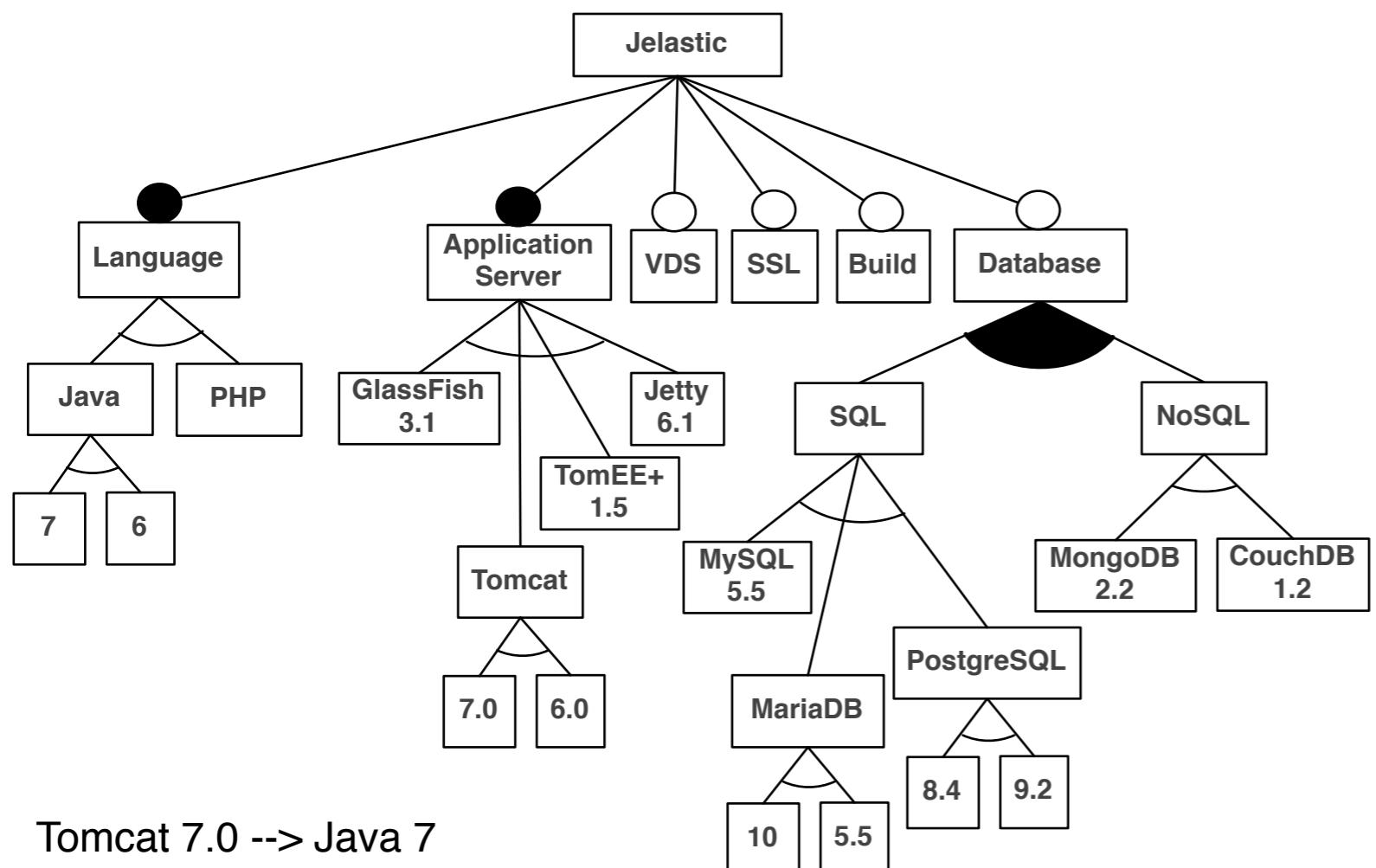
# Feature modeling: example



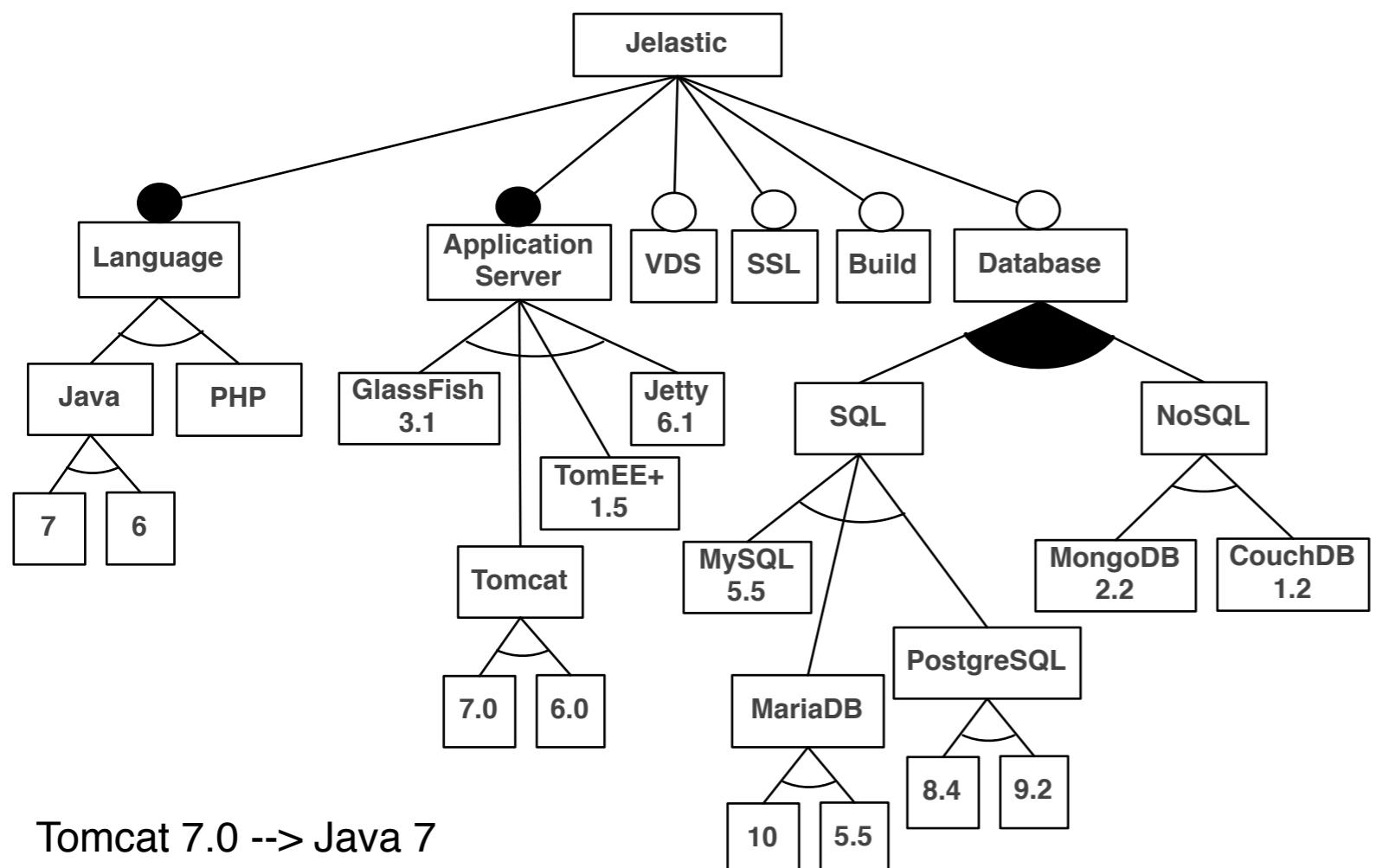
The screenshot shows the Jelastic application configuration interface. It includes:

- Java tab:** Shows Java and PHP application icons.
- Application Servers:** A section with a switch labeled 'ON'. It shows 'Vertical scaling per node' with 1 reserved cloudlet(s) at 128 MB, 200 MHz, 0.0085 EUR per.., and a 15% discount. It also shows 'Horizontal scaling' with 1 node(s) and a Tomcat 7.0 and Java 7 dropdown.
- Resources (cloudlets):** Shows 'Reserved cloudlets' at 1, 'Discount: 15%', and a progress bar indicating saving €1/month.
- Total Reserved Cloud:** Shows a total of 64.
- Total monthly cost:** Shows a cost of \*Free of charge for the trial.
- Cost details:** A link to view costs.
- Quotas & pricing:** A link to view quotas and pricing.
- Environment name:** A field containing 'env-2897066'.

# Feature modeling: example

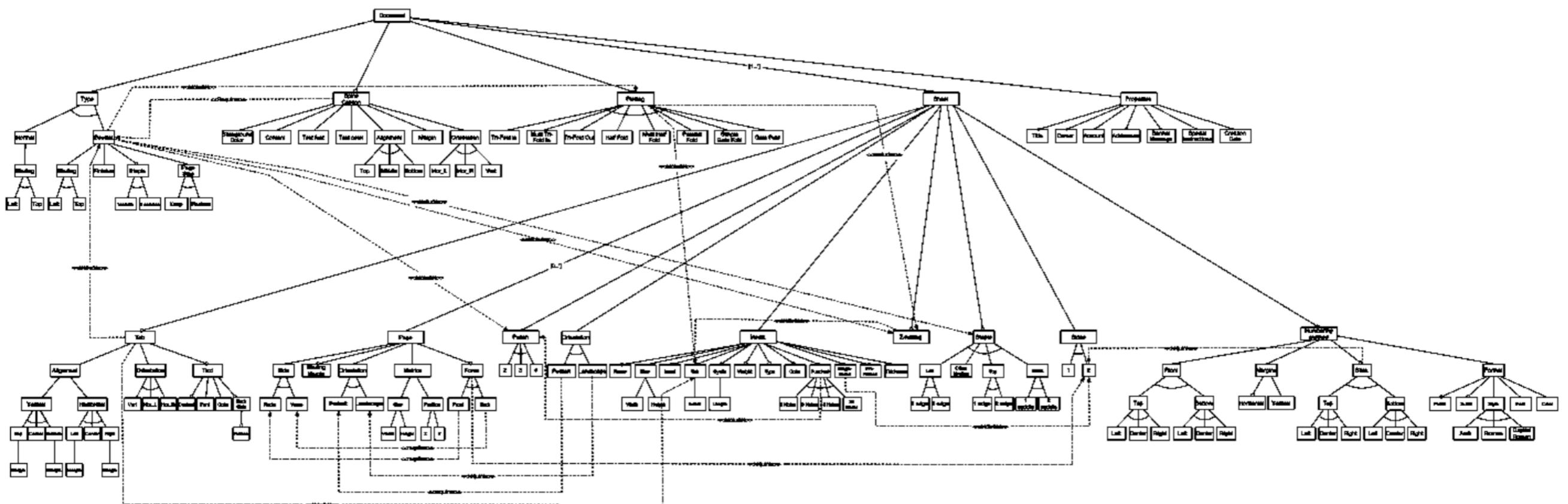


# Feature modeling: example



# Feature models

- ▶ Suggest a visual representation
  - ▶ But...
    - they are not mandatory
    - they do not necessarily scale
    - or respond to user's concerns



# Textual variability languages

---

- ▶ TVL, Clafer, FAMILIAR...

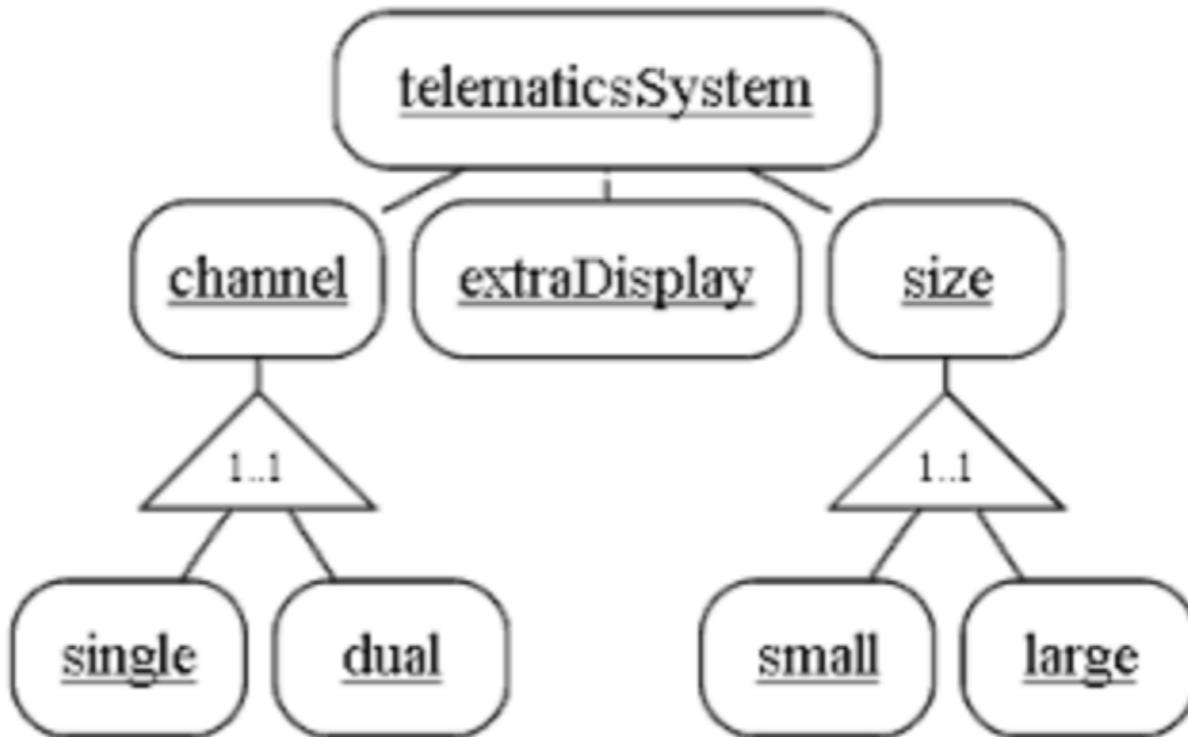
**Clafer model**

```
telematicsSystem
  xor channel
    single
    dual

extraDisplay ?

xor size
  small
  large
```

**Visualization (CVL notation)**



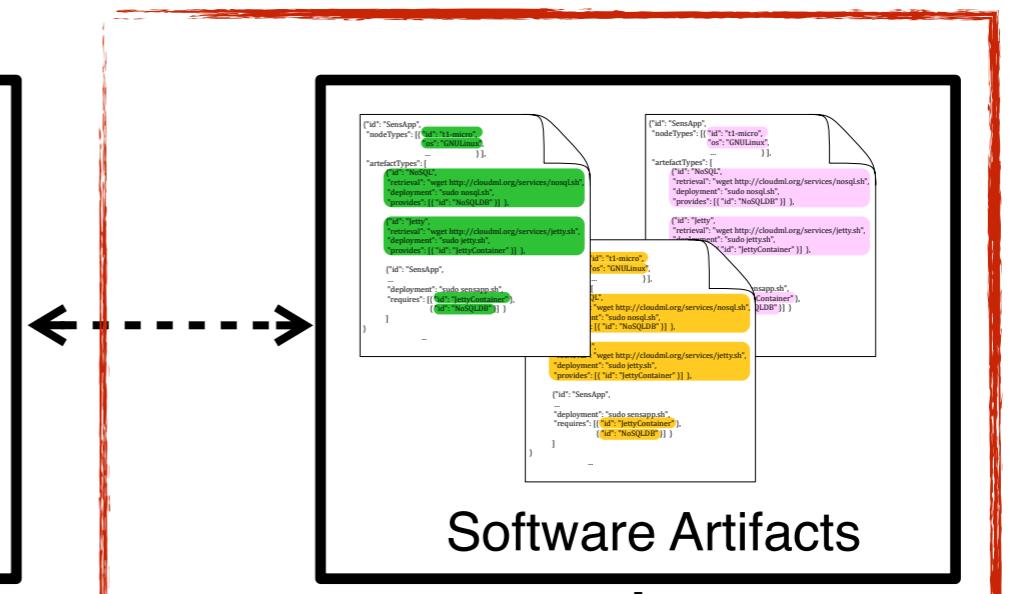
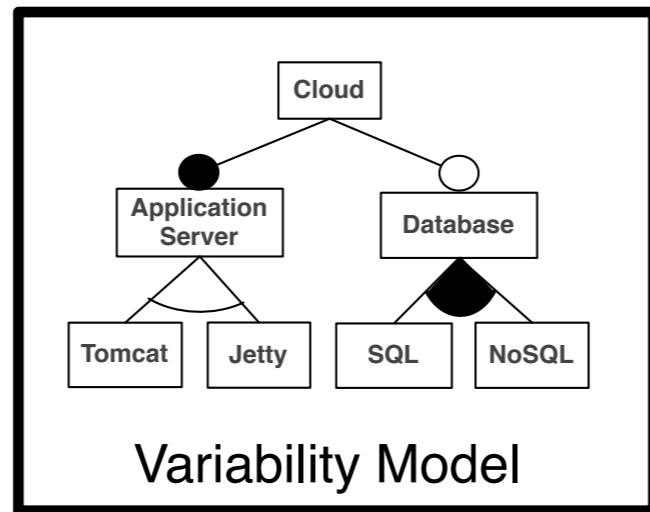
# Time vs Space

---

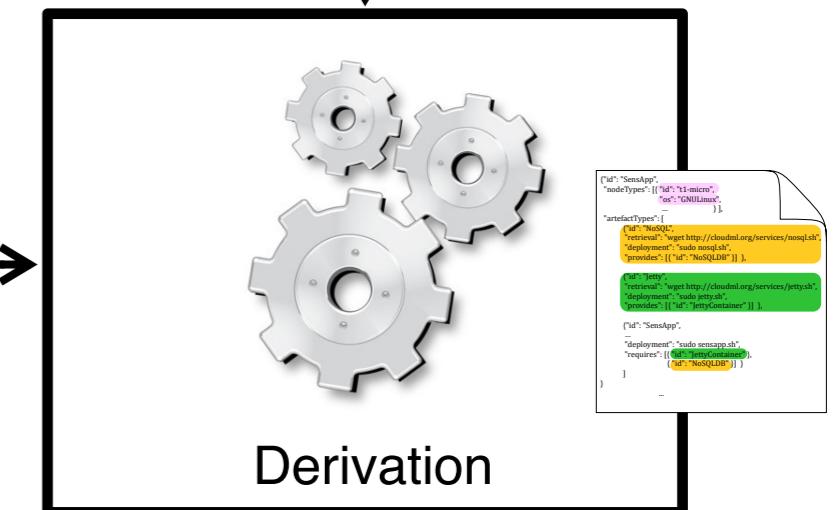
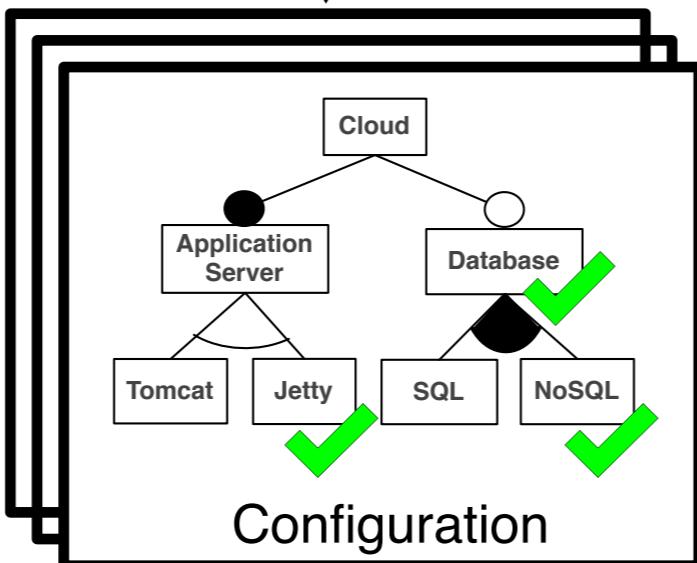
- ▶ Variability in time (releases)
  - different versions of an artifact that are valid at different periods of time
  
- ▶ Variability in space (variants)
  - existence of an artifact in different shapes at the same time

# Software product line engineering

Domain  
Engineering



Application  
Engineering



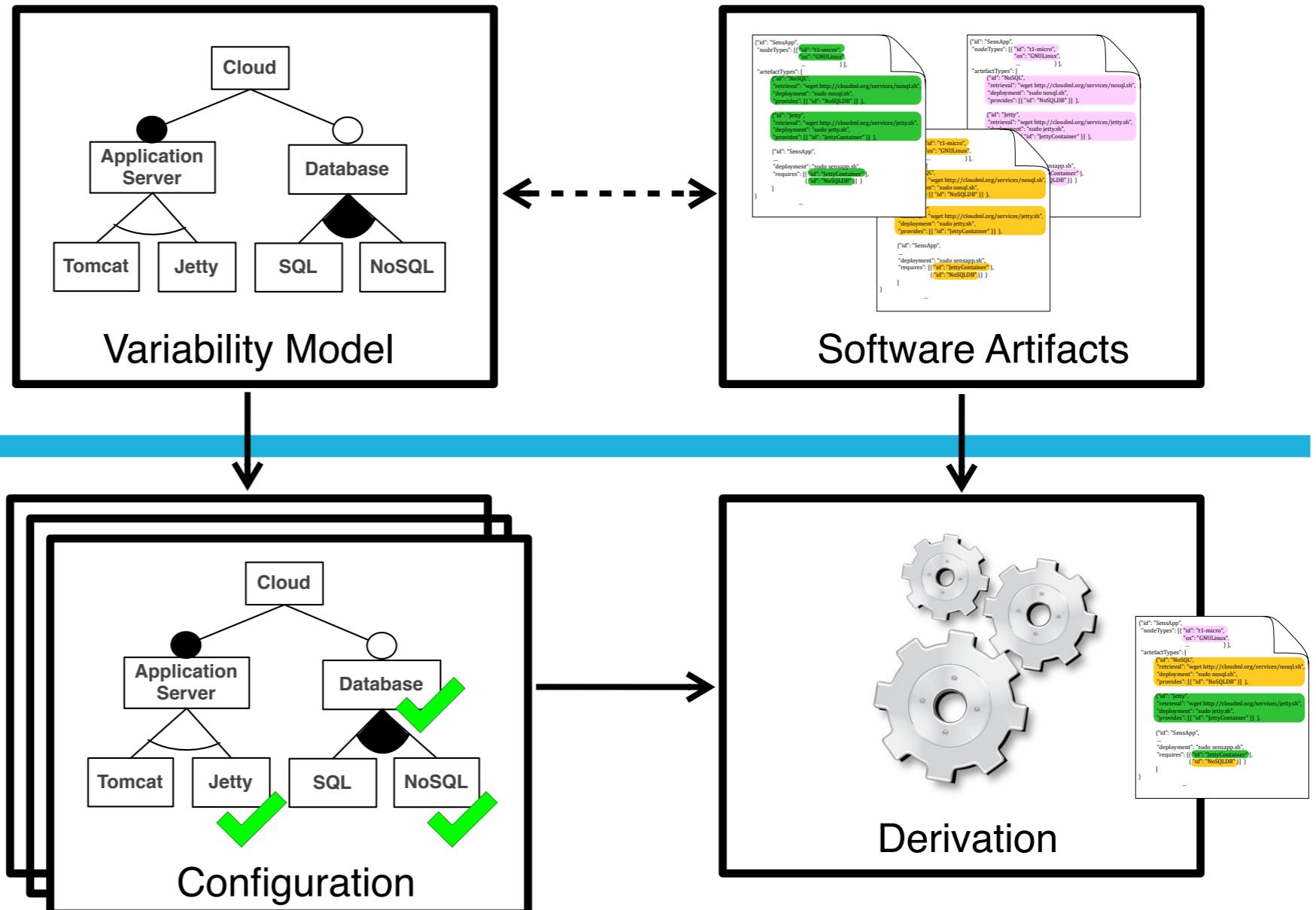
# Software artifacts

---

- ▶ Any assets the domain requires
  - Code snippet
  - Model fragments
  - Documentation
  - ...
  
- ▶ Mapping between feature model and assets
  - one-to-one
  - one-to-many

# Software product line engineering

Domain  
Engineering

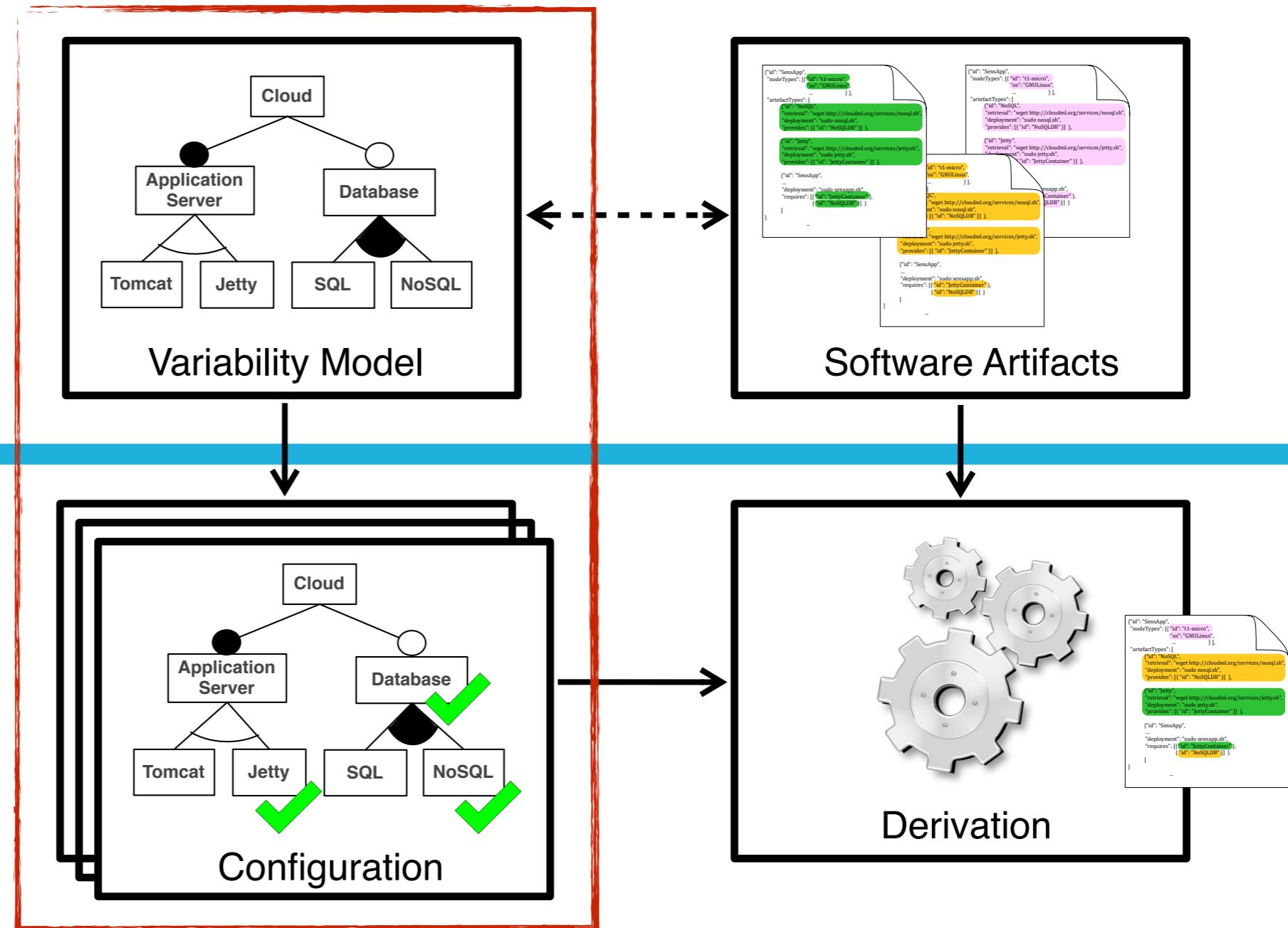


Domain Engineering = development for reuse

Application Engineering = development by reuse

# Software product line engineering

Domain  
Engineering

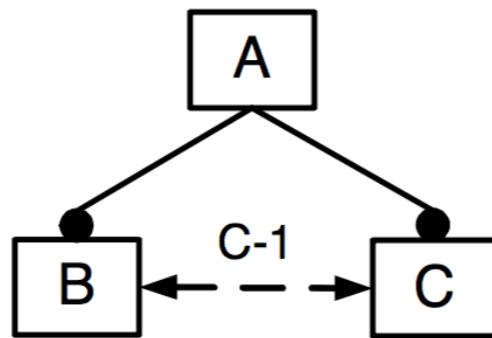


Domain Engineering = development for reuse

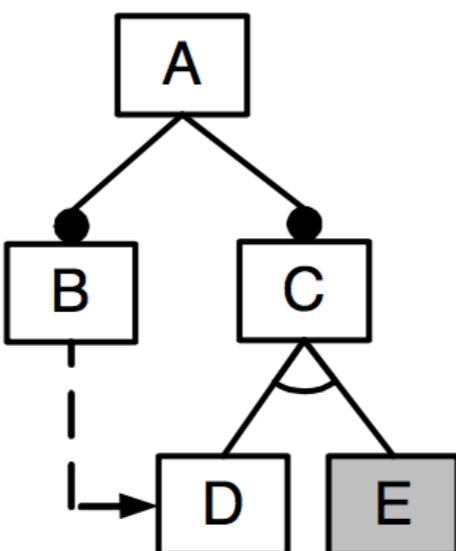
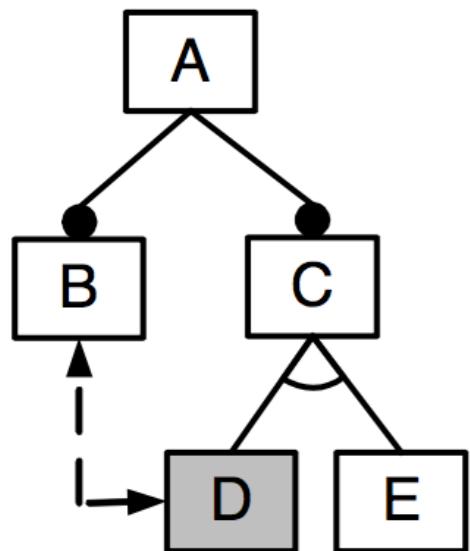
Application Engineering = development by reuse

# Feature modeling: automated analysis

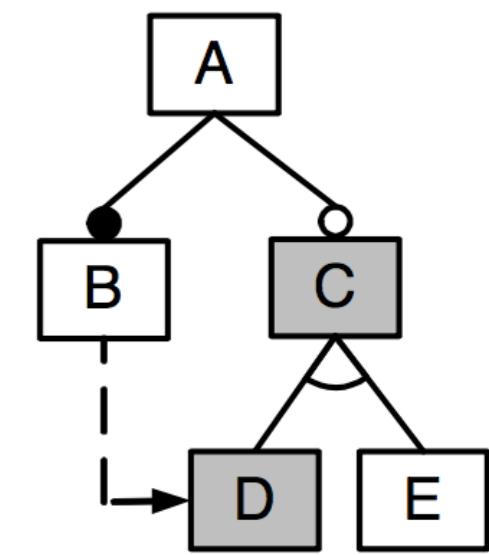
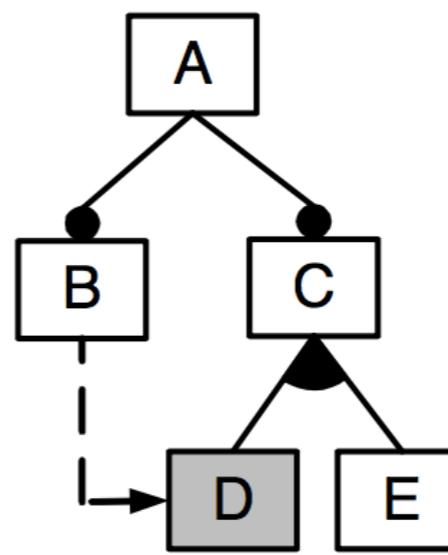
- ▶ Void FM



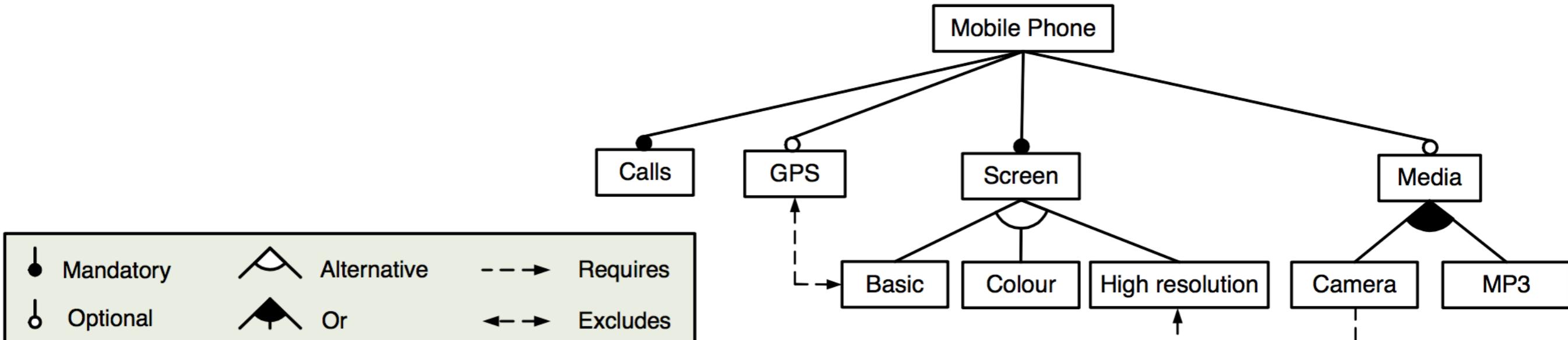
- ▶ Dead feature



- ▶ False optional



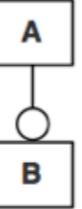
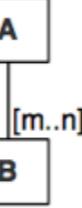
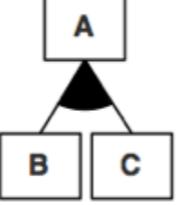
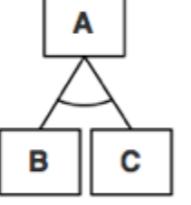
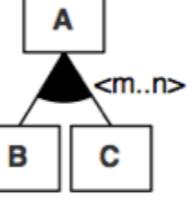
# Feature modeling: automated analysis



## All products, Number of products

```
P1 = {MobilePhone,Calls,Screen,Basic}
P2 = {MobilePhone,Calls,Screen,Basic,Media,MP3}
P3 = {MobilePhone,Calls,Screen,Colour}
P4 = {MobilePhone,Calls,Screen,Colour,GPS}
P5 = {MobilePhone,Calls,Screen,Colour,Media,MP3}
P6 = {MobilePhone,Calls,Screen,Colour,Media,MP3,GPS}
P7 = {MobilePhone,Calls,Screen,High resolution}
P8 = {MobilePhone,Calls,Screen,High resolution,Media,MP3}
P9 = {MobilePhone,Calls,Screen,High resolution,Media,MP3, Camera}
P10 = {MobilePhone,Calls,Screen,High resolution,Media, Camera}
P11 = {MobilePhone,Calls,Screen,High resolution, GPS}
P12 = {MobilePhone,Calls,Screen,High resolution,Media,MP3, GPS}
P13 = {MobilePhone,Calls,Screen,High resolution,Media, Camera, GPS}
P14 = {MobilePhone,Calls,Screen,High resolution,Media, Camera, MP3, GPS}
```

# Feature modeling: reasoning

Relation	Feature model notation	Constraint
Mandatory		$B = A$
Optional		$\text{ifThen}(A = 0; B = 0; )$
Feature Cardinality		$\text{ifThenElse}(A = 0; B = 0; B \in \{m, n\})$
Or-group		$\text{ifThenElse}(A > 0; \text{sum}(B, C) \geq 1; \text{sum}(B, C) = 0; )$
Alternative-group		$\text{ifThenElse}(A > 0; \text{sum}(B, C) = 1; \text{sum}(B, C) = 0; )$
Group cardinality		$\text{ifThenElse}(A > 0; \text{sum}(B, C) \in \{m, n\}; \text{sum}(B, C) = 0; )$
Implies	$A \rightarrow B$	$\text{ifThen}(A > 0; B > 0; )$
Excludes	$A \rightarrow \neg B$	$\text{ifThen}(A > 0; B = 0; )$

# Feature modeling: reasoning

Import

**Install Details**

The operation cannot be completed. See the details.

Name	Version	Id
<input checked="" type="checkbox"/> EMF Edit Data Binding	1.2.0.v20110606-0...	org.eclipse.emf.databinding.edit.feature....
<input checked="" type="checkbox"/> EMF Model Transaction Workbench Integration	1.5.1.v20110823-1...	org.eclipse.emf.workspace.feature.group
<input checked="" type="checkbox"/> EMF Validation Framework	1.5.0.v20110502-1...	org.eclipse.emf.validation.feature.group
<input checked="" type="checkbox"/> Mylyn Commons	3.6.0.v20110608-1...	org.eclipse.mylyn.commons.feature.group
<input checked="" type="checkbox"/> Mylyn Task-Focused Interface	3.6.0.v20110608-1...	org.eclipse.mylyn.context_feature.feature...

Select All   Deselect All

**Details**

Cannot complete the install because of a conflicting dependency.  
Software being installed: EMF Model Transaction Workbench Integration 1.5.1.v20110823-1800-37708s734D3C7D3CCCC1  
(org.eclipse.emf.workspace.feature.group 1.5.1.v20110823-1800-37708s734D3C7D3CCCC1)

< Back   Next >   Finish   Cancel



# Feature modeling: examples

---



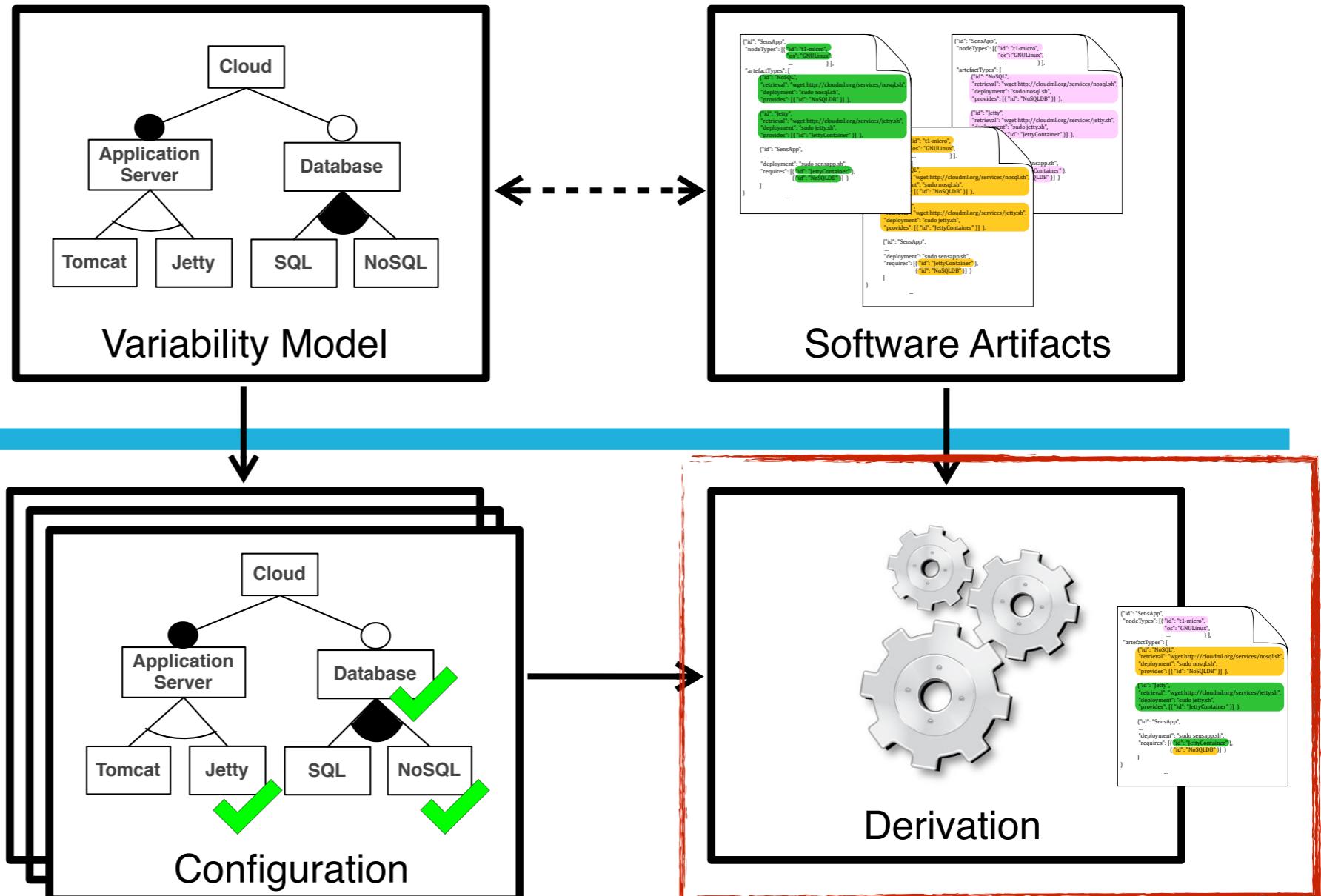
**S.P.L.O.T.**  
Software Product Lines Online Tools

Marcilio Mendonca, Moises Branco, Donald Cowan: S.P.L.O.T. - Software Product Lines Online Tools.  
In Companion to the 24th ACM SIGPLAN International Conference on Object-Oriented  
Programming, Systems, Languages, and Applications, OOPSLA 2009, October 2009,  
Orlando, Florida, USA.



# Software product line engineering

Domain  
Engineering



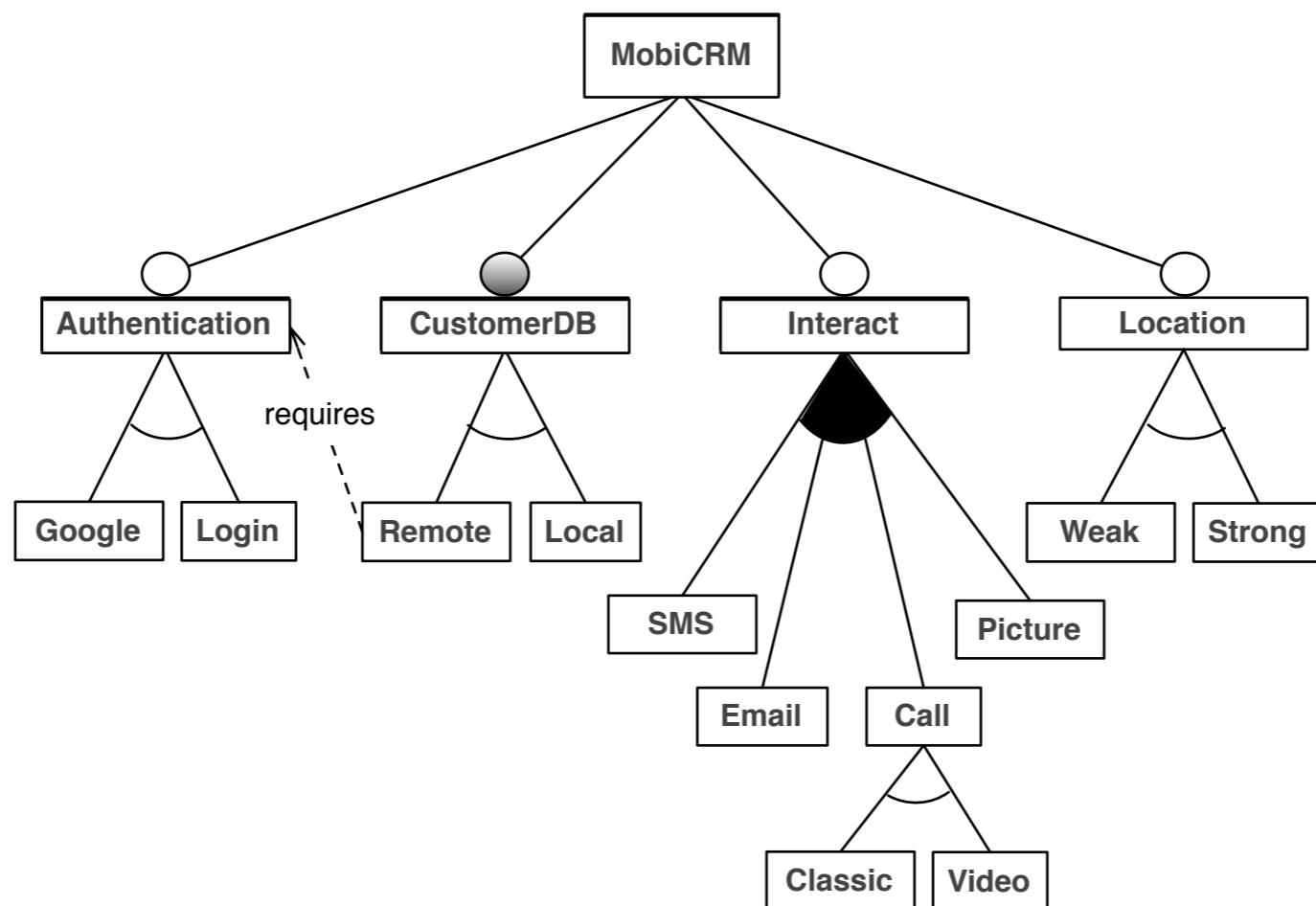
# Derivation

---

- ▶ If assets are
  - Code snippet
  - Model fragments
  - Aspects
  - ...
  
- ▶ Derivation can be
  - Code generation
  - Merge
  - Weaving
  - ...

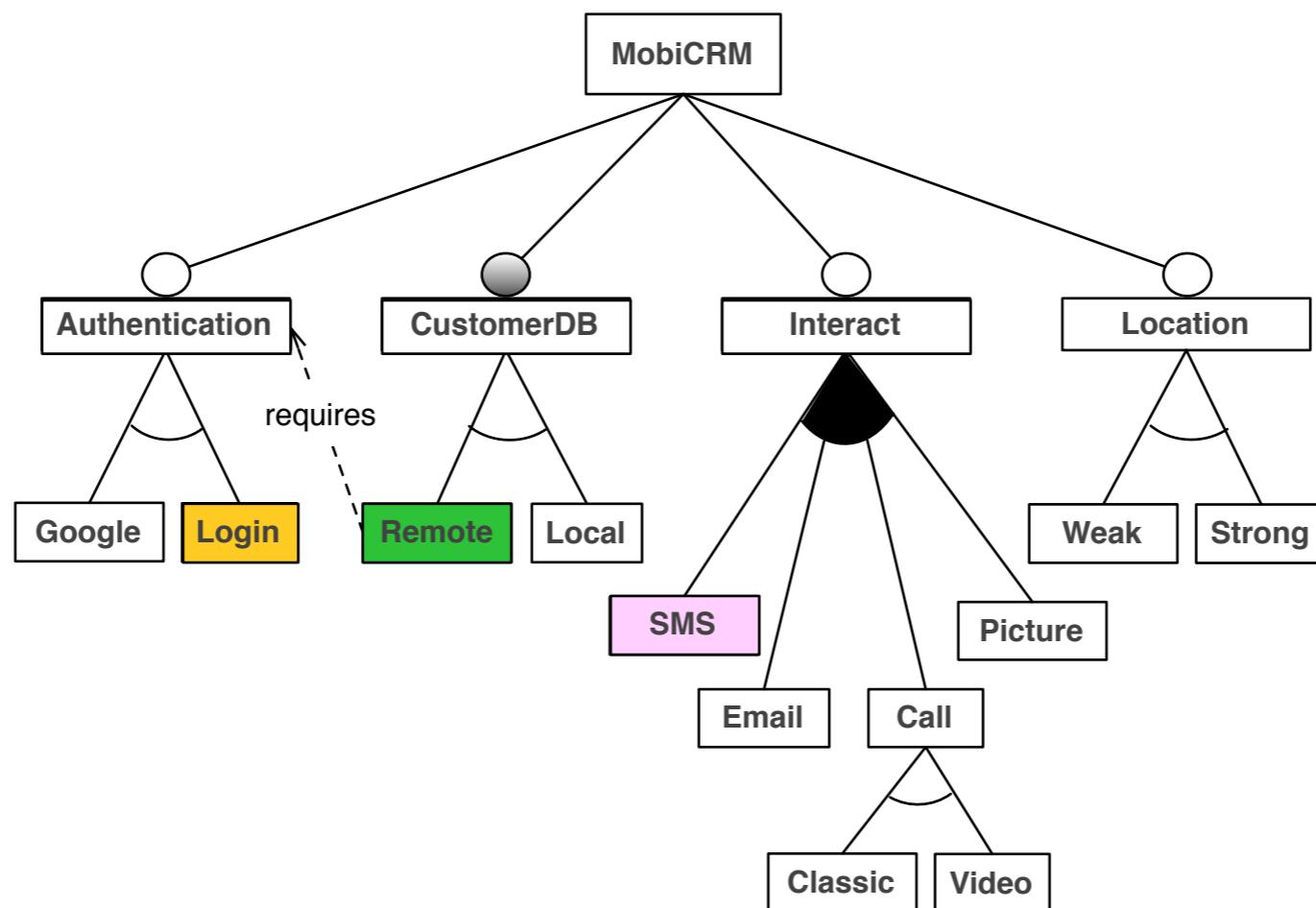
# Derivation: examples

---



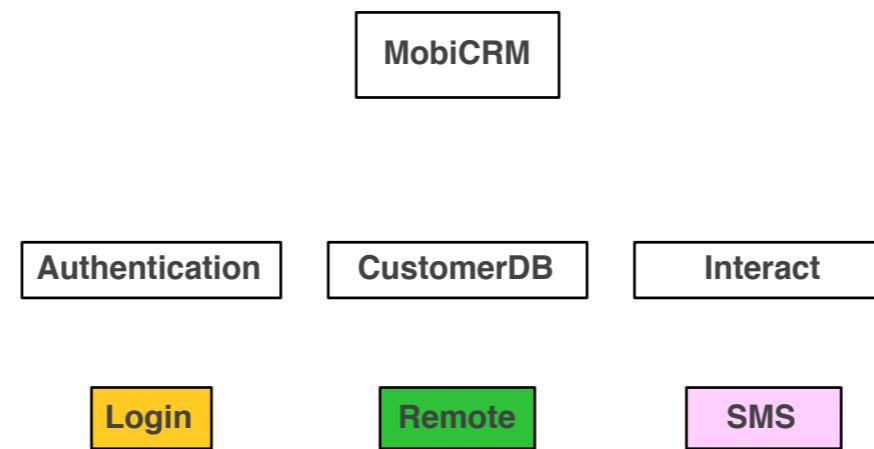
# Derivation: examples

---



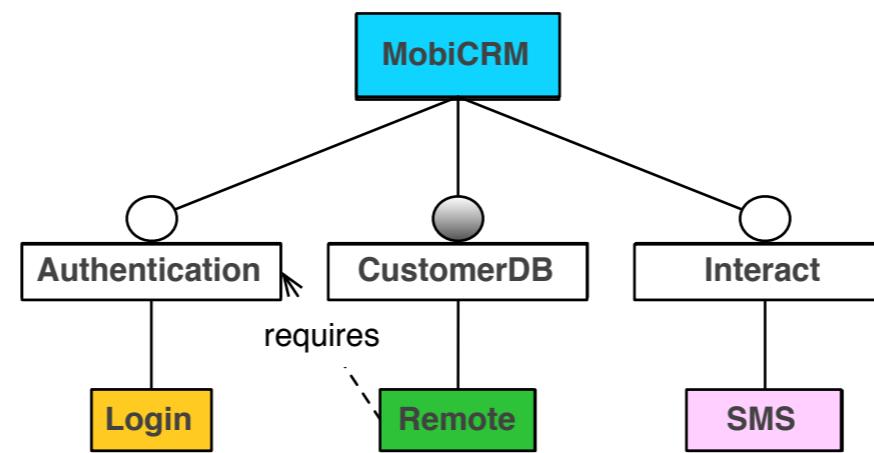
# Derivation: examples

---

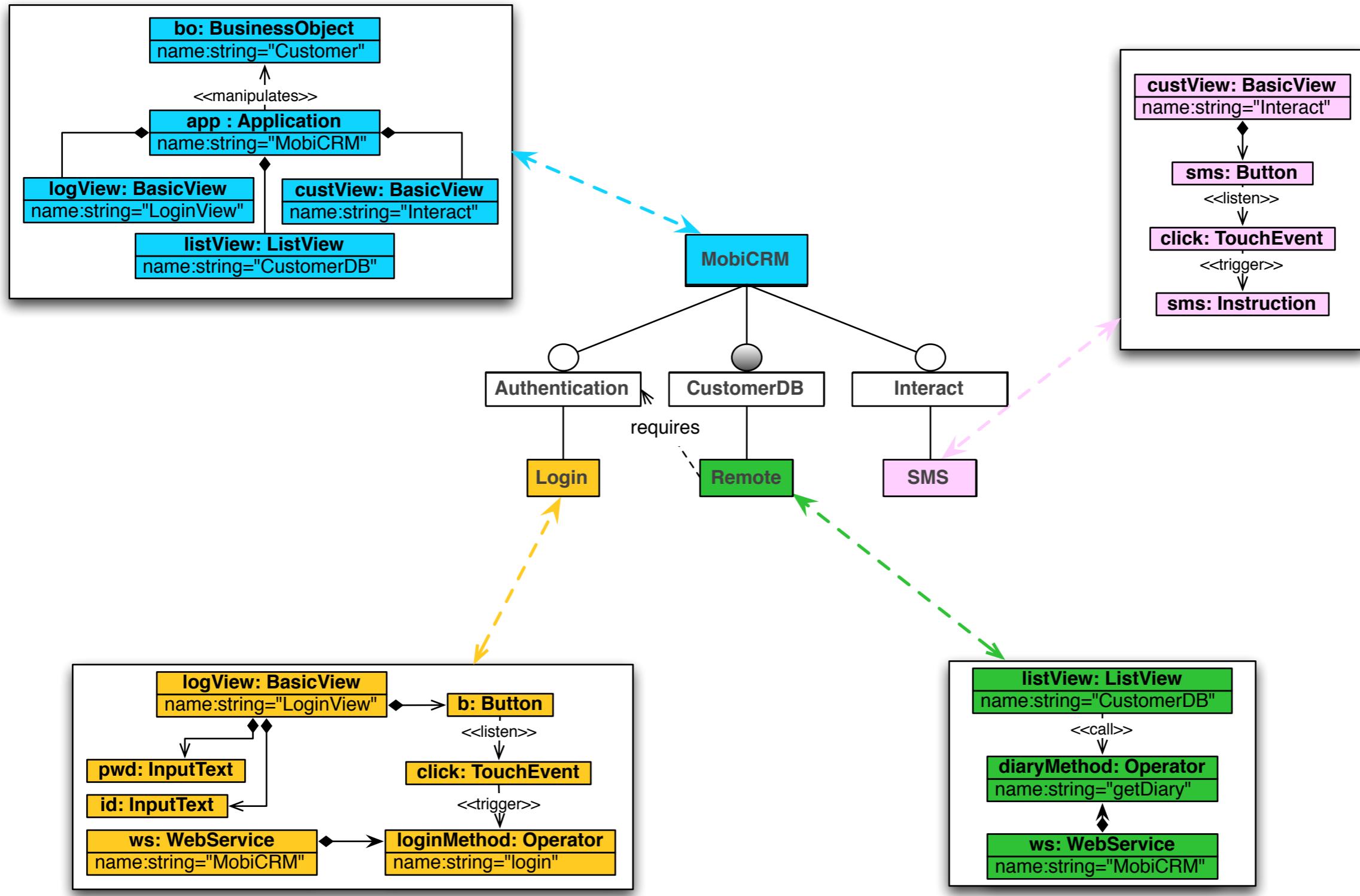


# Derivation: examples

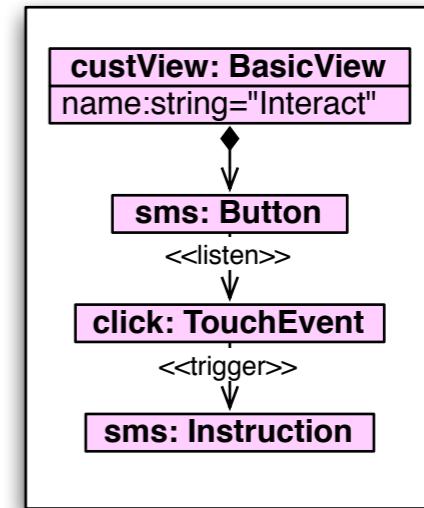
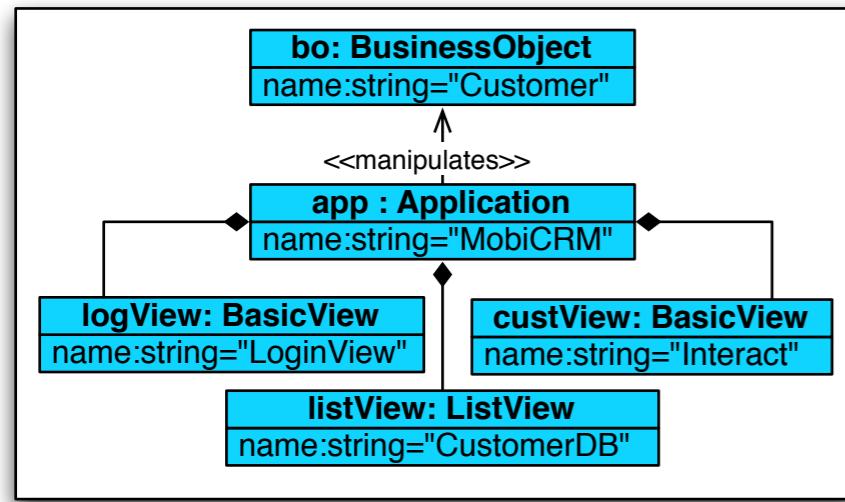
---



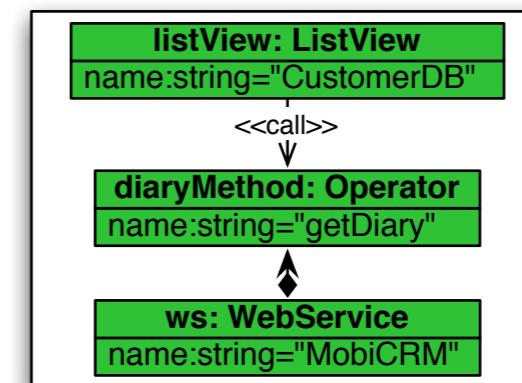
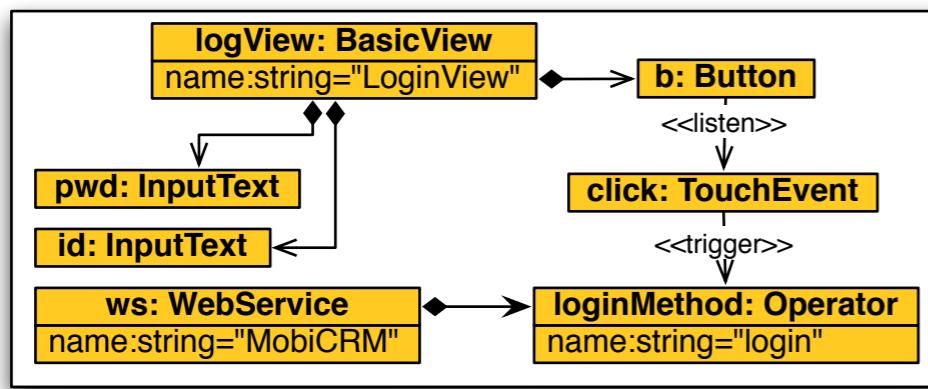
# Derivation: examples



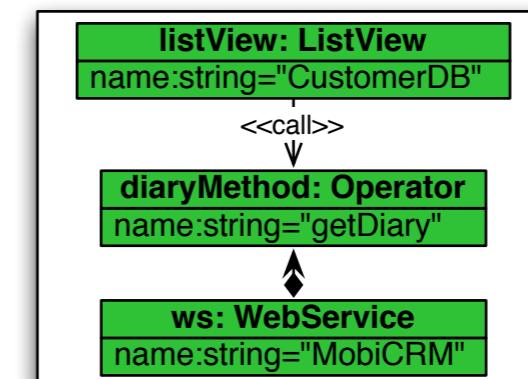
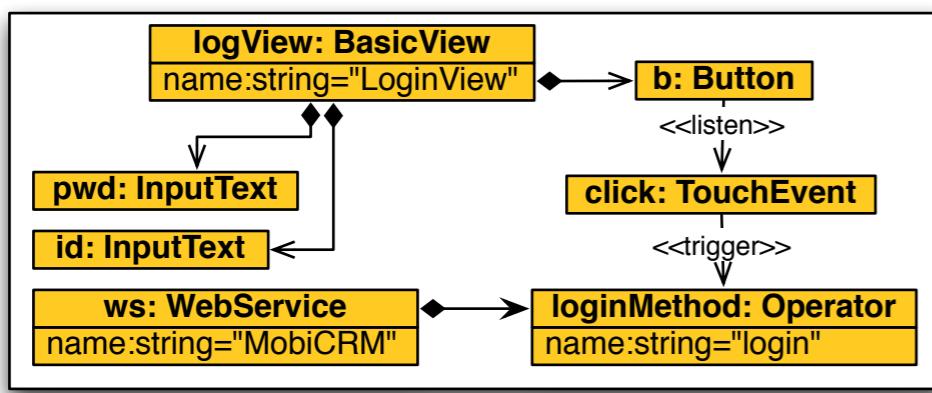
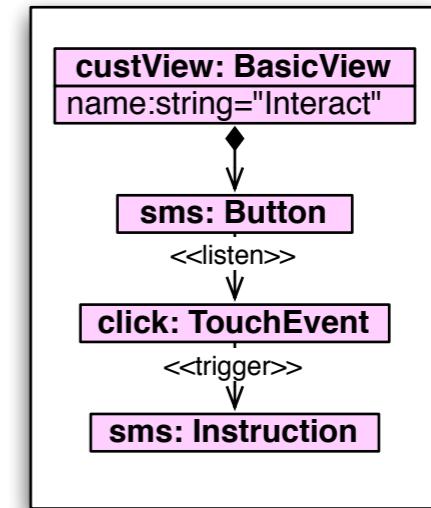
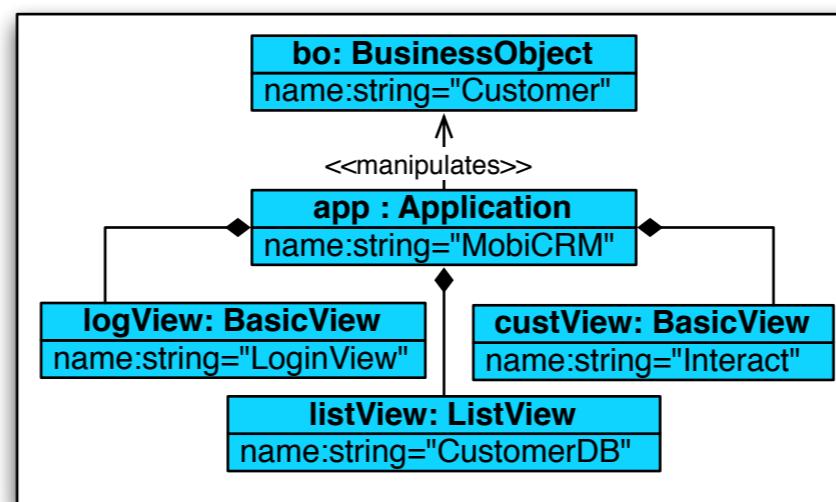
# Derivation: examples



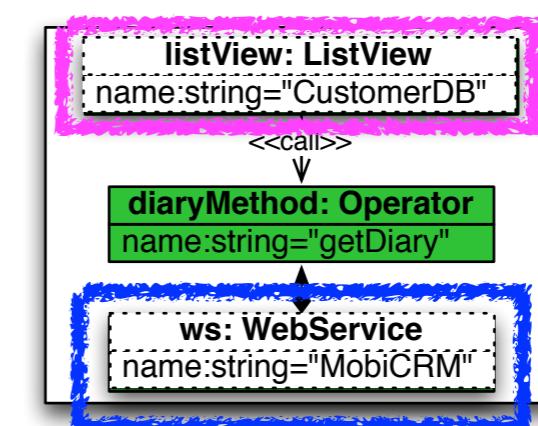
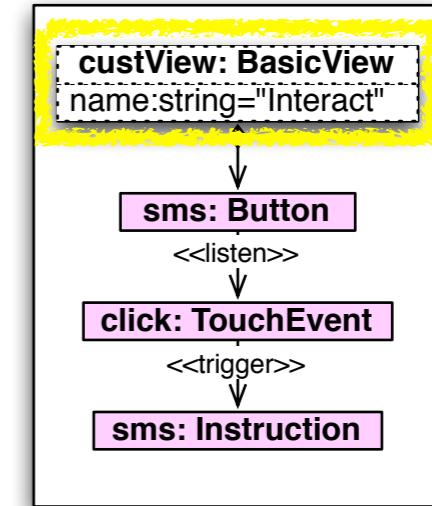
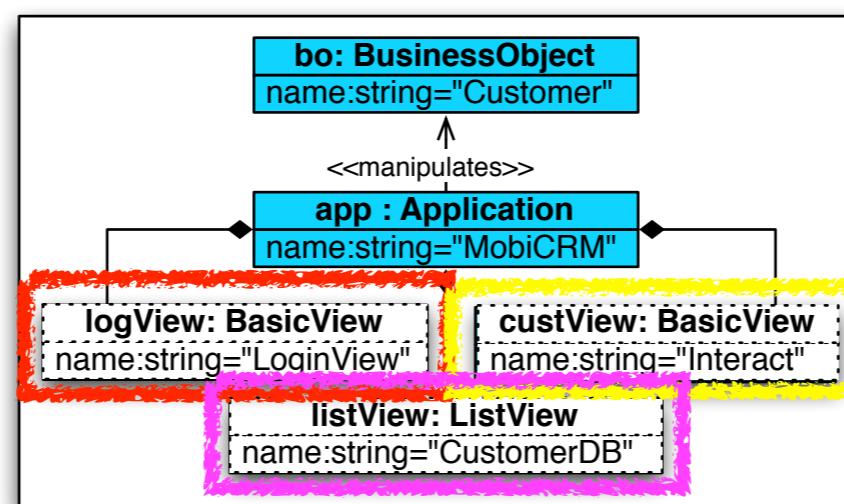
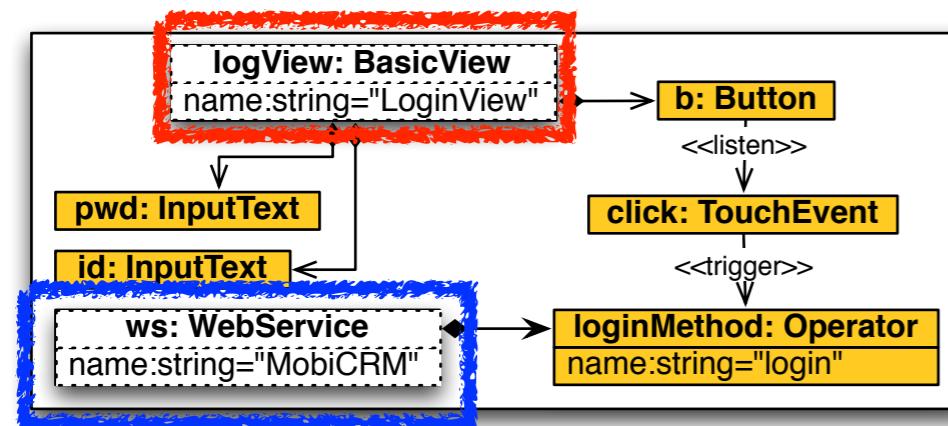
Merge



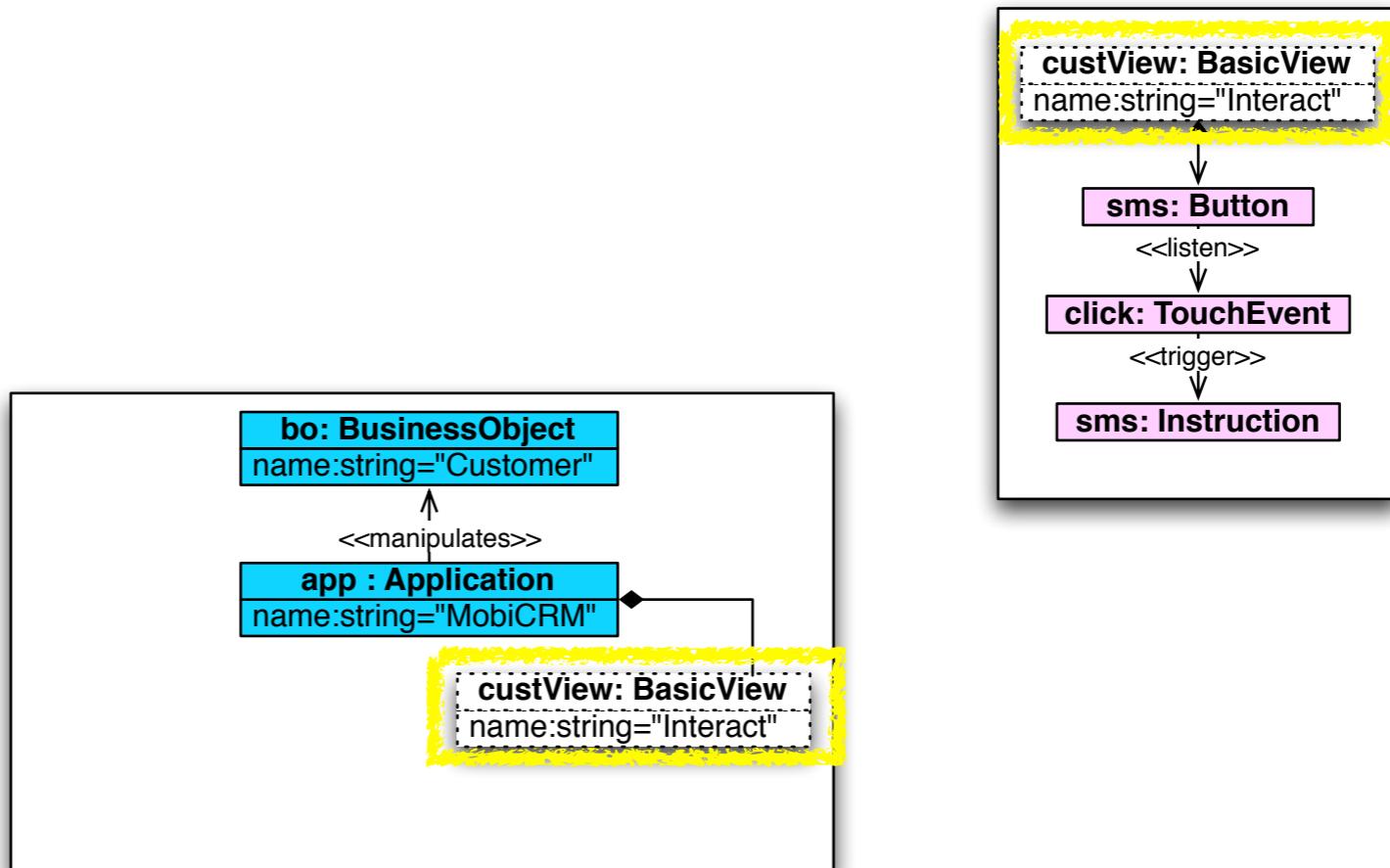
# Derivation: examples



# Derivation: examples

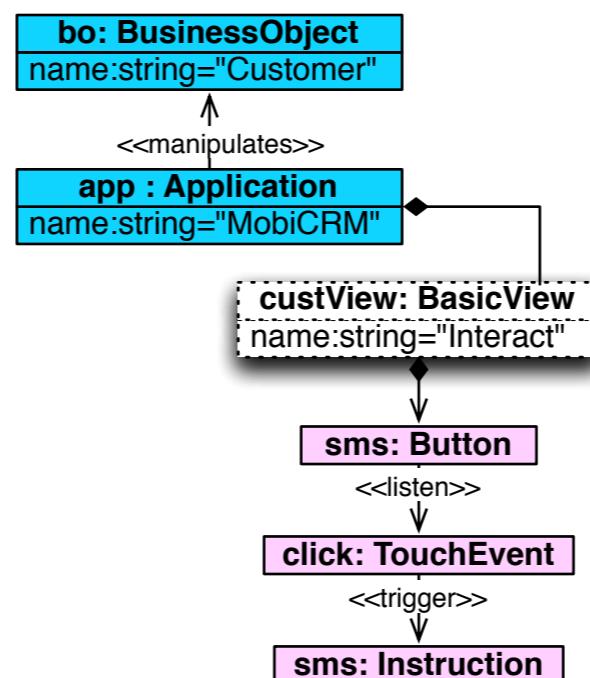


# Derivation: examples

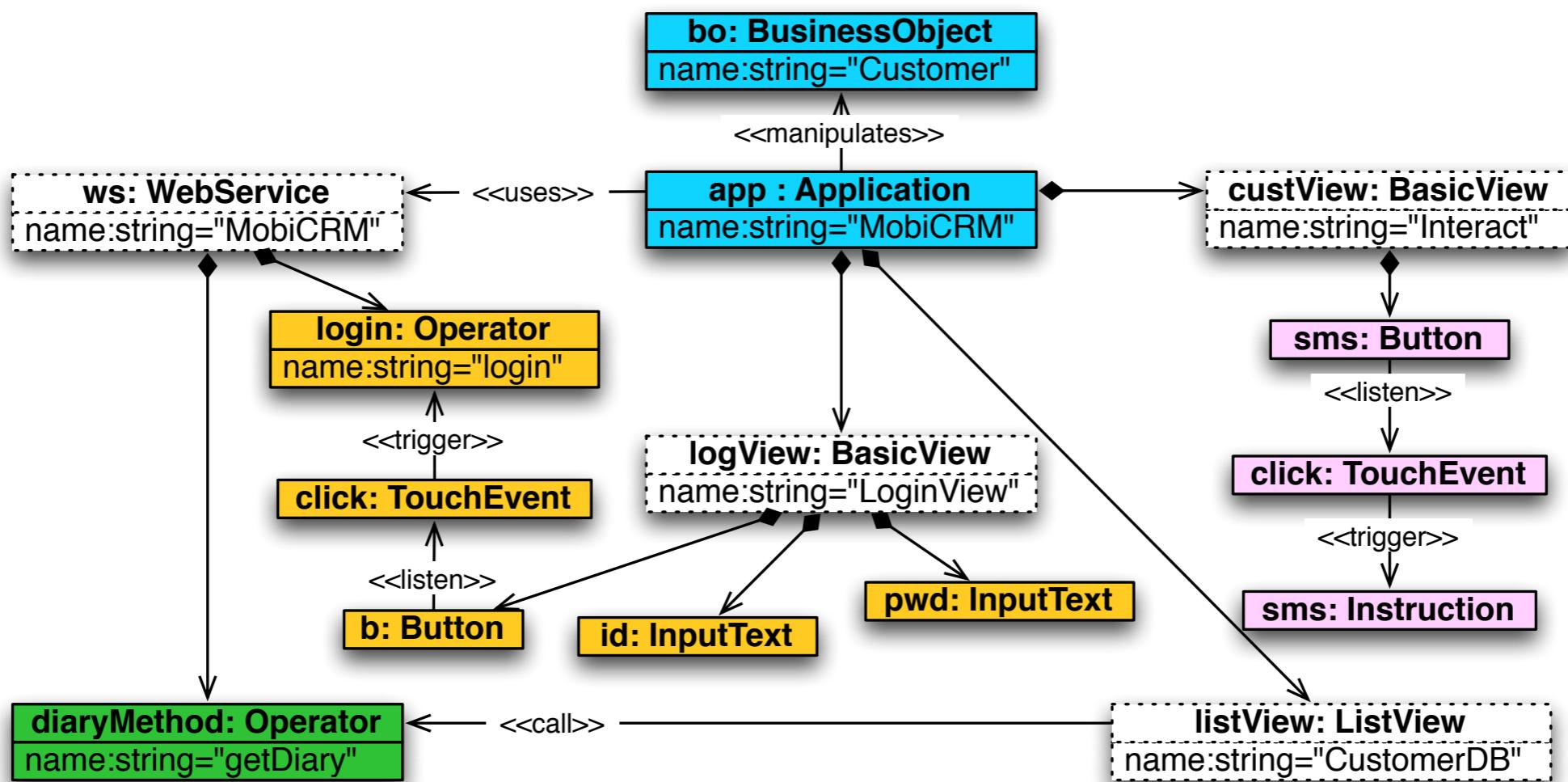


# Derivation: examples

---



# Derivation: examples



# Derivation: examples (cloud)

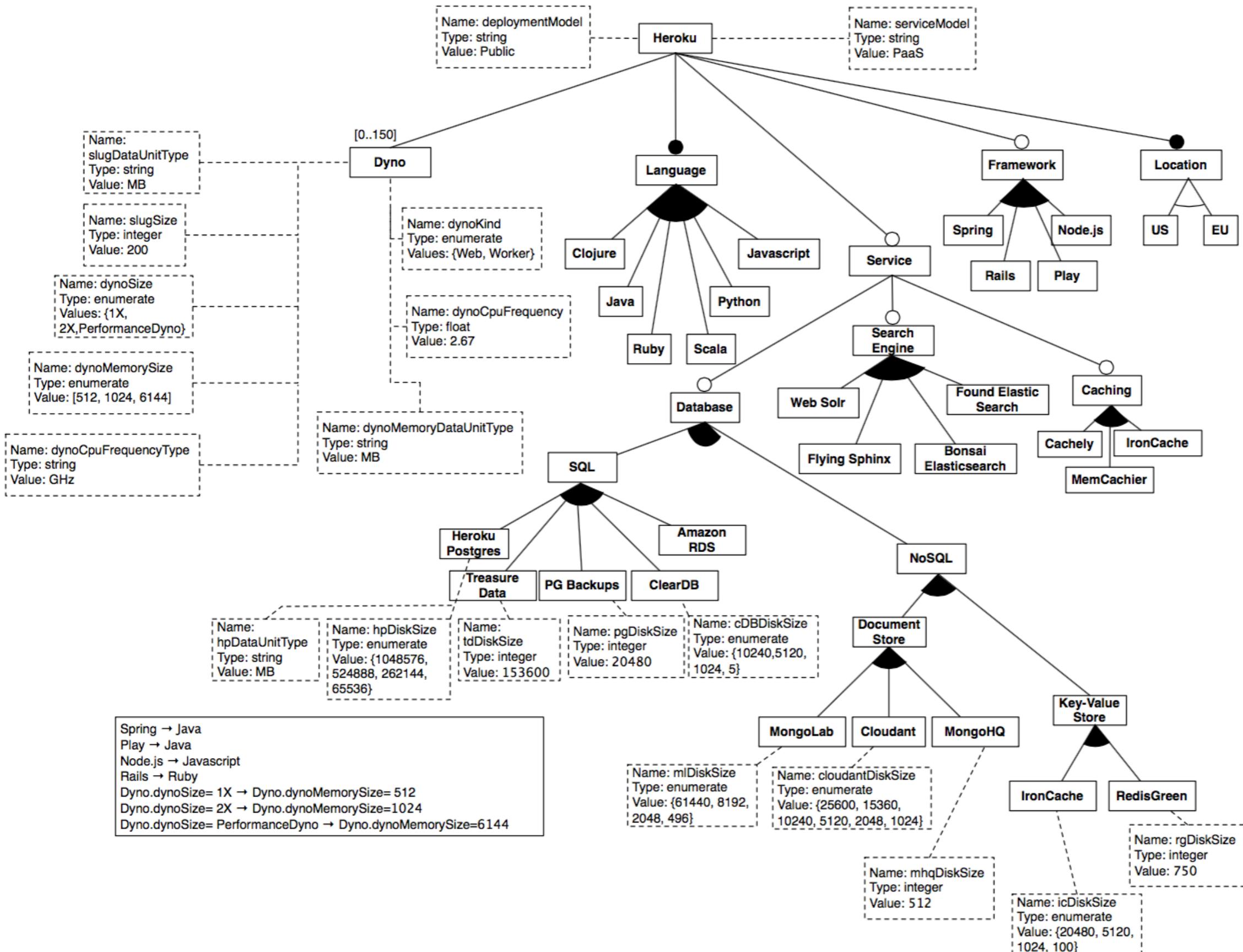
---

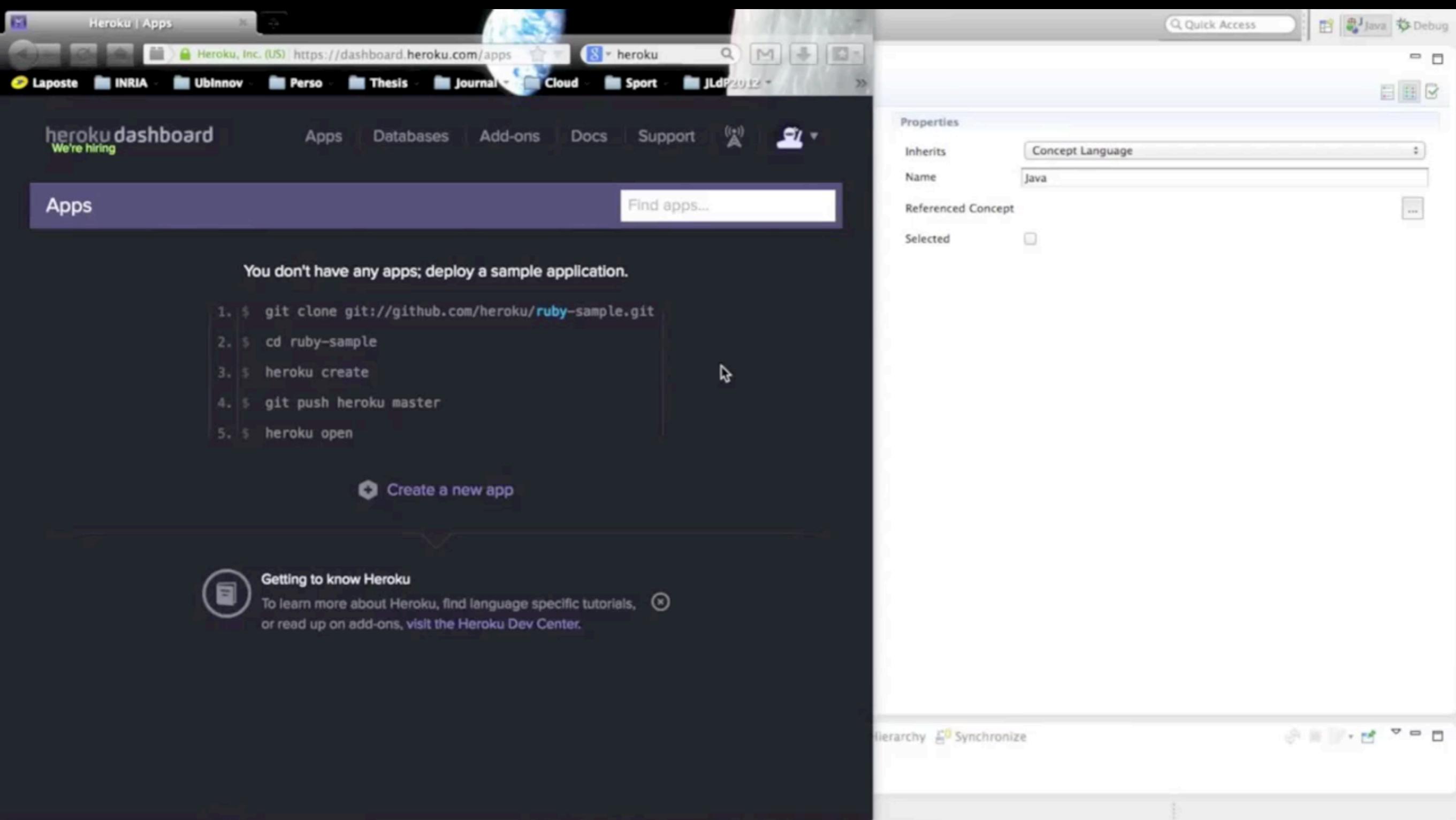


# Derivation: examples (cloud)



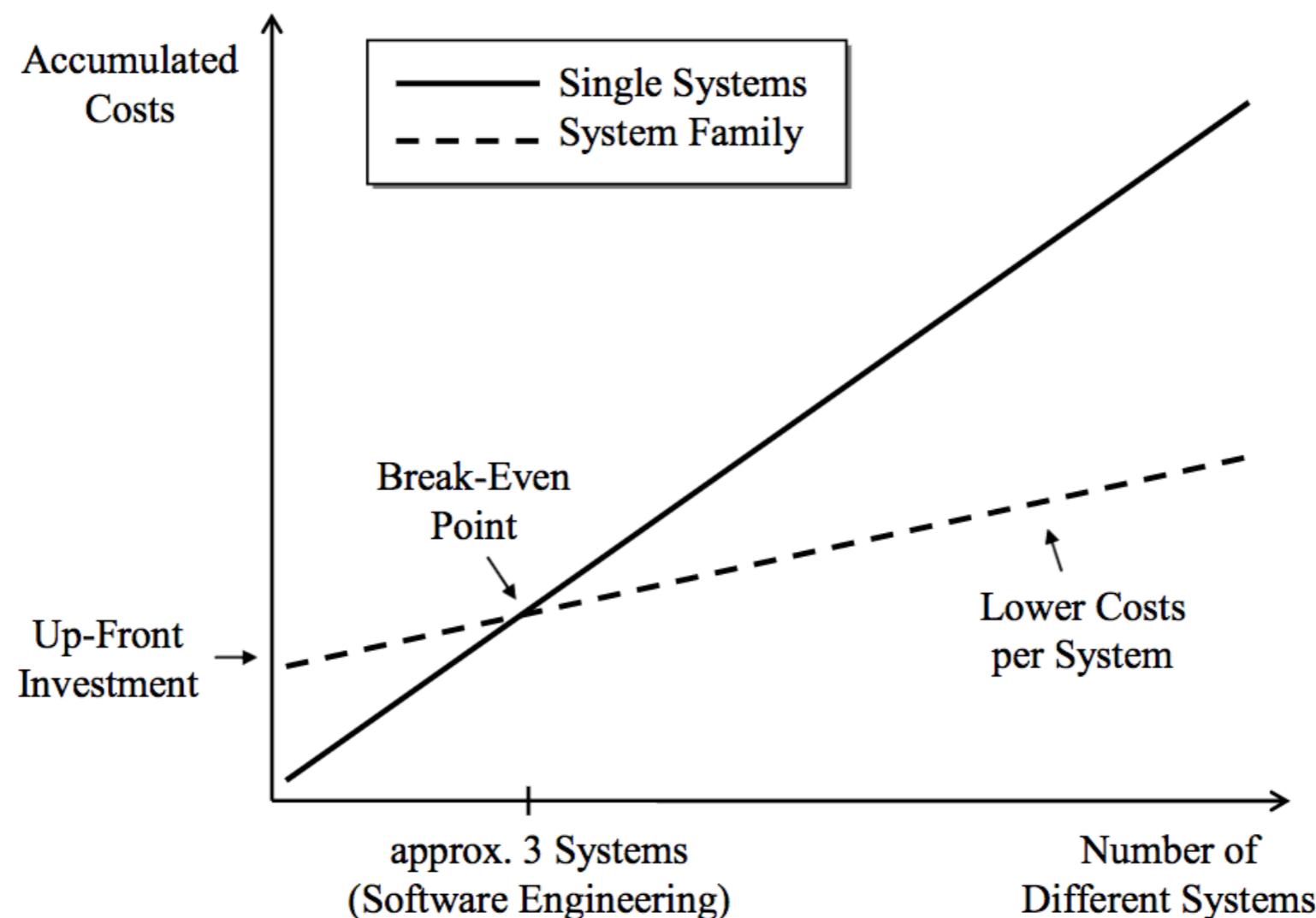
# Derivation: examples (cloud)





# SPL: main benefits

- ▶ Shorten time-to-market
- ▶ Reduce production costs



- ▶ Improve product reliability

# Example

---



## Printer firmware



- ▶ Production cost reduced by 75%
- ▶ Development time reduced by 33%
- ▶ Reported defects reduced by 96%

# To conclude

---

- ▶ SPL engineering has been successful ([splc.net/fame.html](http://splc.net/fame.html))



**BOSCH**

Invented for life



**PHILIPS**

**ERICSSON**

**SALiON™**  
TARGET. WIN. DELIVER.™



**NOKIA**  
Connecting People

**CelsiusTech**



**Lucent Technologies**  
Bell Labs Innovations



# To conclude

---

## ► What about the future?

- Cyber-physical systems
- Massively distributed
- Complex et variable



- ▶ Deux exercices indépendants
  - ▶ SPL
  - ▶ Variability modeling
- ▶ En binôme
- ▶ Lisez bien (et respectez) les consignes

- ▶ Deux exercices indépendants
  - ▶ SPL
  - ▶ Variability modeling
- ▶ En binôme
- ▶ Lisez bien (et respectez) les consignes

**[http://researchers.lille.inria.fr/cquinton/SPL/TP1\\_sujet.pdf](http://researchers.lille.inria.fr/cquinton/SPL/TP1_sujet.pdf)**