

# Model-based Software Product Lines Overview and Principles

Mathieu Acher  
Maître de Conférences  
[mathieu.acher@irisa.fr](mailto:mathieu.acher@irisa.fr)

# Material

[https://github.com/acherm/  
teaching-MDE-IL18189](https://github.com/acherm/teaching-MDE-IL18189)

<http://teaching.variability.io>  
<http://familiar.variability.io>



**Generator**  
**~ composition of**  
**video sequences**



**video  
variants**



**Generator**  
~ composition of  
video sequences

**video  
variants**

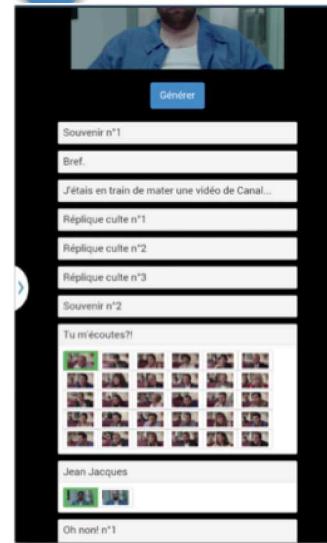
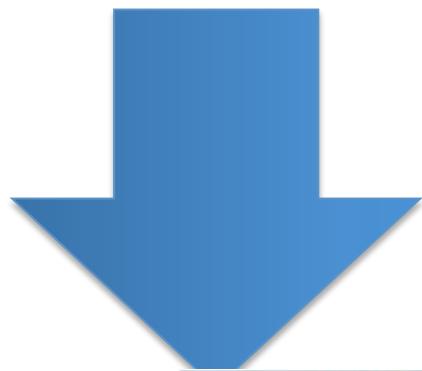




```
foo1.videogen ✘

mandatory videoseq v1 "https://www.youtube.com/watch?v=PjNi1uYhV5w"
optional videoseq v2 "v2/folder/v2.mp4"
alternatives v3 {
    videoseq v31 "v3/seq1.mp4"
    videoseq v32 "v3/seq1.mp4"
    videoseq v33 "v3/seq1.mp4"
}

alternatives v4 {
    videoseq v41 "v4/seq1.mp4"
    videoseq v42 "v4/seq1.mp4"
}
mandatory videoseq v5 "https://www.youtube.com/watch?v=ezKx-S0LiNQ"
```



- ## Website/online
- Random generation
  - Configurator
  - Game
  - ...

foo1.videogen

```
mandatory videoseq v1 "https://www.youtube.com/watch?v=PJNi1uYhV5w"
optional videoseq v2 "v2folder/v2.mp4"
@alternatives v3 {
    videoseq v31 "v3/seq1.mp4"
    videoseq v32 "v3/seq1.mp4"
    videoseq v33 "v3/seq1.mp4"
}

@alternatives v4 {
    videoseq v41 "v4/seq1.mp4"
    videoseq v42 "v4/seq1.mp4"
}
mandatory videoseq v5 "https://www.youtube.com/watch?v=ezKx-S0LiNQ"
```

**Feature model:** another model for modeling “features” of your Web site (eg ability to save the video; mode=generation with frequencies)

**configurable**  
**generator**  
**of video generator**

#### Website/online

- Random generation
- Configurator
- Game
- ...



foo1.videogen ✘

```
mandatory videoseq v1 "https://www.youtube.com/watch?v=PjNi1uYhV5w"
optional videoseq v2 "v2folder/v2.mp4"
alternatives v3 {
    videoseq v31 "v3/seq1.mp4"
    videoseq v32 "v3/seq1.mp4"
    videoseq v33 "v3/seq1.mp4"
}
alternatives v4 {
    videoseq v41 "v4/seq1.mp4"
    videoseq v42 "v4/seq1.mp4"
}
mandatory videoseq v5 "https://www.youtube.com/watch?v=ezKx-S0LiNQ"
```

#1 How to design,  
create, and support  
dedicated languages  
(DSLs)?

#2 How to transform  
models/programs?



#3 How to manage  
variability/variants?

#4 How do frameworks  
internally work?

# Plan

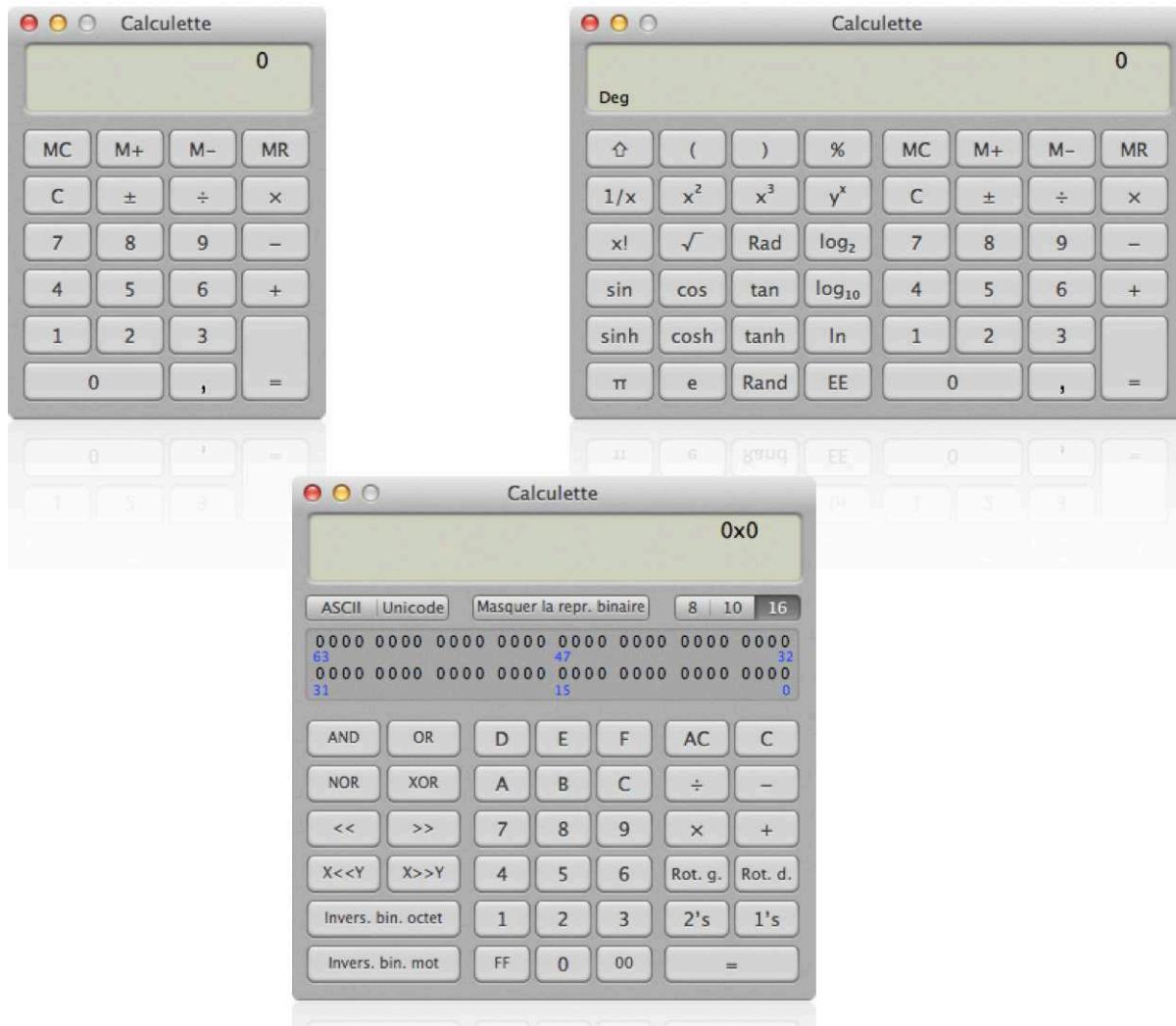
- Challenges and Overview
  - Developing billions of software product is hard but now a common practice
- Implementing Variability
  - Revisit of existing techniques and curriculum
- Specificity of Product Line Engineering
  - Process, methods
- Feature Models
  - Defacto standard for modeling product lines and variability

# Contract

- The idea of software product lines and variability
  - You will be able to recognize this class of systems
  - Aware of the complexity
  - Aware of the specific development process
  - Aware of existing techniques
- Feature modeling
  - A widely used formalism for modeling product lines and configurable systems in a broad sense

# Software Product Line and Variability Engineering

## Challenges and Overview





```
macher-wifi:getting-started macher1$ yo jhipster
```

I'm all done. Running `npm install & bower install` for you to install the required dependencies.

```
[1/15] [P] Generating application configuration  
[2/15] [P] Generating application entities  
[3/15] [P] Generating application services  
[4/15] [P] Generating application controllers  
[5/15] [P] Generating application entities  
[6/15] [P] Generating application services  
[7/15] [P] Generating application controllers  
[8/15] [P] Generating application entities  
[9/15] [P] Generating application services  
[10/15] [P] Generating application controllers  
[11/15] [P] Generating application entities  
[12/15] [P] Generating application services  
[13/15] [P] Generating application controllers  
[14/15] [P] Generating application entities  
[15/15] [P] Generating application services
```

Welcome to the JHipster Generator v2.17.0

```
? (1/15) What is the base name of your application? jhipster  
? (2/15) What is your default Java package name? com.mycompany.myapp  
? (3/15) Do you want to use Java 8? Yes (use Java 8)  
? (4/15) Which *type* of authentication would you like to use? (Use arrow keys)  
> HTTP Session Authentication (stateful, default Spring Security mechanism)  
  OAuth2 Authentication (stateless, with an OAuth2 server implementation)  
  Token-based authentication (stateless, with a token)
```

« 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 »



aka Variability

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

**Starter****Home Premium Upgrade****Professional Upgrade****Ultimate Upgrade**

\$119.99\*

Buy

\$199.99\*

Buy

\$219.99\*

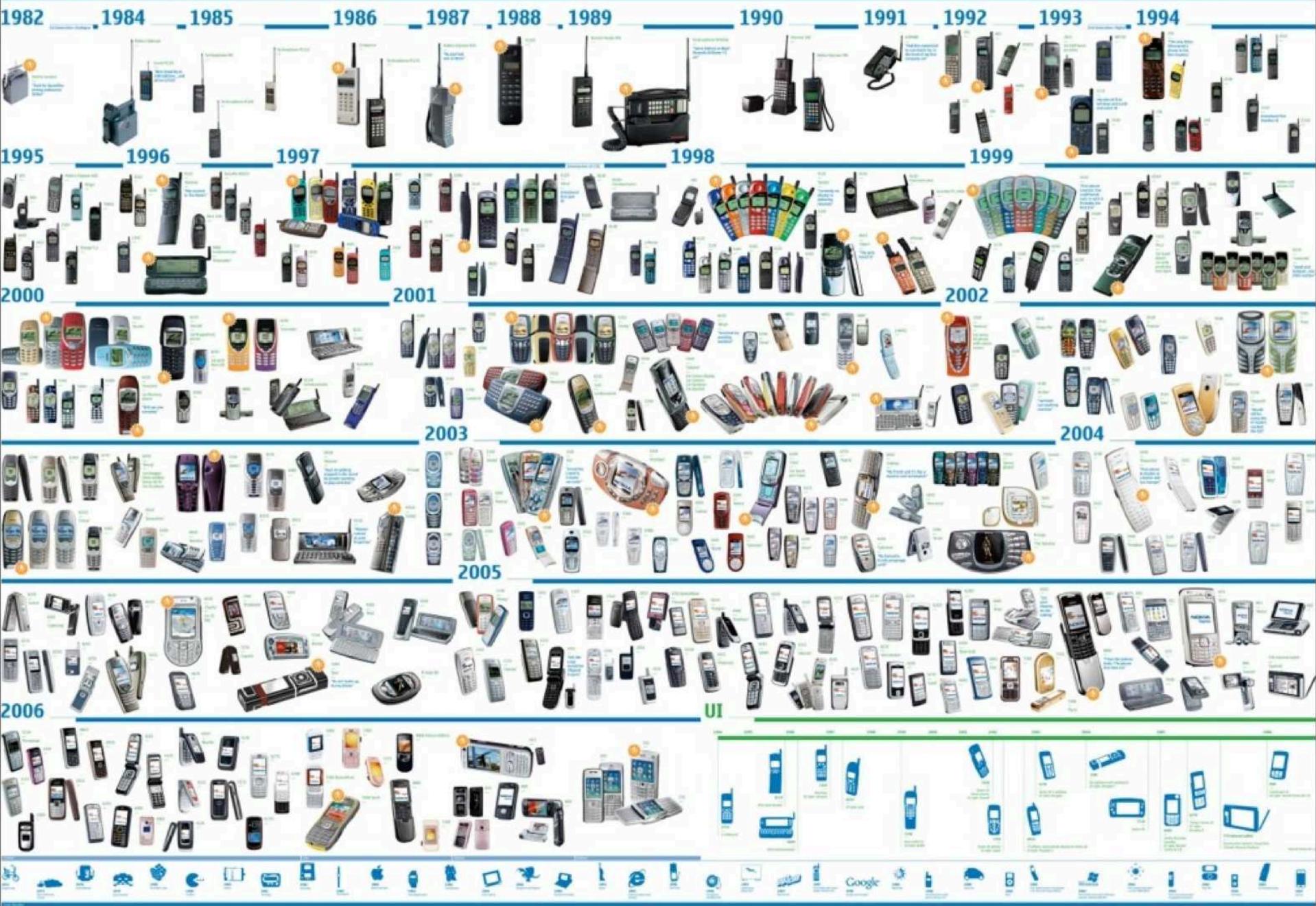
Buy

**Communication**

Bluetooth support	✓	✓	✓	✓
Join a homegroup	✓	✓	✓	✓
Internet Explorer 8	✓	✓	✓	✓
View Available Networks	✓	✓	✓	✓
Windows Connect Now (WCN)	✓	✓	✓	✓
Create a homegroup		✓	✓	✓
Location and other sensors support		✓	✓	✓
Support for joining domains			✓	✓

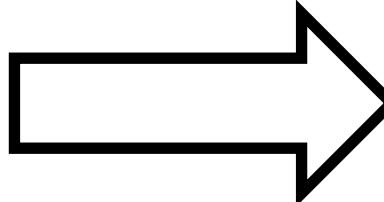
**Entertainment**

DirectX 11	✓	✓	✓	✓
Gadgets	✓	✓	✓	✓
Games Explorer	✓	✓	✓	✓
Play To	✓	✓	✓	✓
Windows Media Player 12	✓	✓	✓	✓
Create and play DVDs	✓	✓	✓	✓
Internet TV	✓	✓	✓	✓





# Software-intensive systems

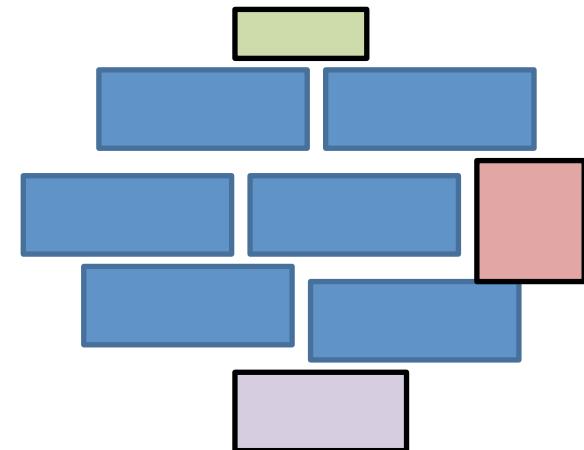


come in many variants

# Software Product Line Engineering

Factoring out **commonalities**

for **Reuse** [Krueger et al., 1992] [Jacobson et al., 1997]



Managing **variabilities**

for Software **Mass Customization** [Bass et al., 1998] [Krueger et al., 2001], [Pohl et al., 2005]



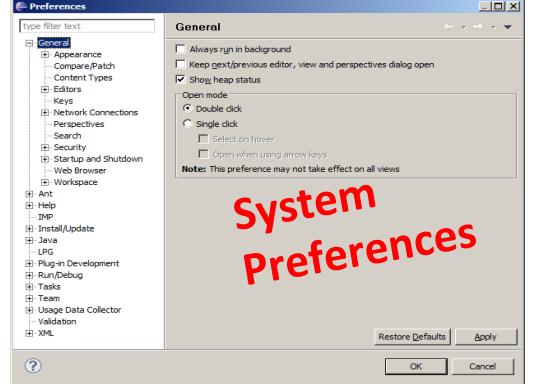
# Variability

“the ability of a system to be efficiently extended, changed, customized or configured for use in a particular context”

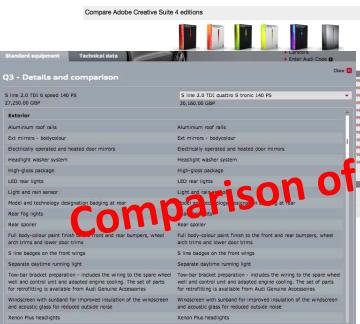
*Mikael Svahnberg, Jilles van Gurp, and Jan Bosch (2005)*



NEW KANGOO VAN RANGE  
Configurators



System Preferences



Comparison of \*

[httpd.conf](http://httpd.conf) -- win32 Apache  
Building a Web Server, for Windows

```
Listen 80
ServerRoot "/www/Apache2"
DocumentRoot "/www/webroot"

ServerName localhost:80
ServerAdmin admin@localhost

ServerSignature On
ServerTokens Full

DefaultType text/plain
AddDefaultCharset ISO-8859-1

UseCanonicalName Off

HostnameLookups Off

ErrorLog logs/error.log
LogLevel error

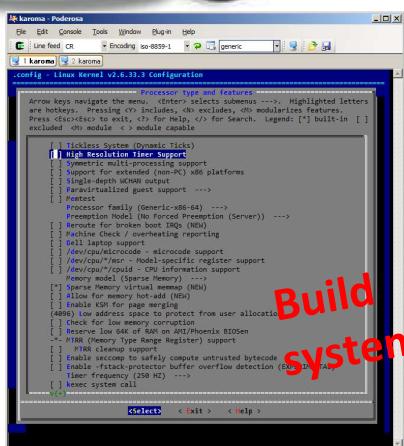
PidFile logs/httpd.pid

Timeout 300

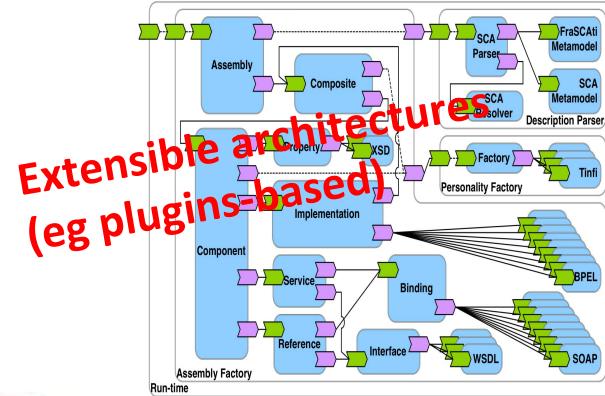
KeepAlive On
MaxKeepAliveRequests 100
KeepAliveTimeout 15

<IfModule mpm_winnt.c>
    ThreadsPerChild 250
    MaxRequestsPerChild 0
</IfModule>
```

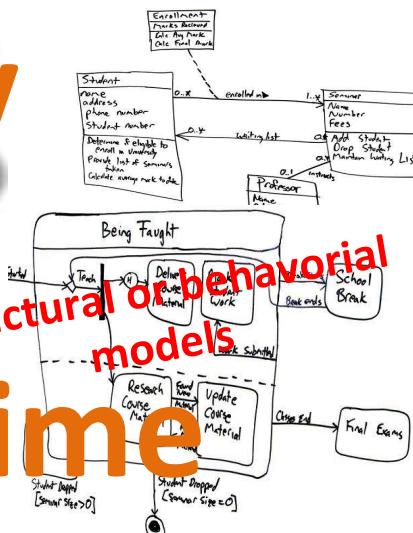
Configuration files



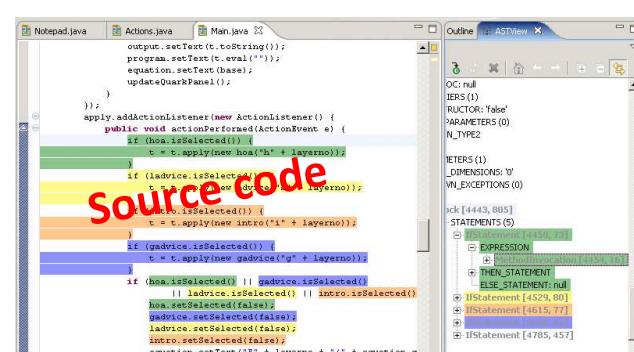
Build systems



Extensible architectures  
(eg plugins-based)



Structural or behavioral models



<b>Developer Tools</b>
<b>Development</b>
<b>Drivers</b>
<b>DTP/Prepress</b>
<b>Educational</b>
<b>Finance</b>
<b>Font Tools</b>
<b>Games</b>
<b>Graphics</b>
<b>HTML Tools</b>
<b>Internet Utilities</b>
<b>iPhone Applications</b>
<b>iPod Tools</b>
<b>Math/Scientific</b>
<b>Multimedia</b>
<b>Network/Admin</b>
<b>Screensavers</b>
<b>Security</b>
<b>Spotlight Plugins</b>
<b>Utilities</b>
<b>System Utilities</b>
<b>Video</b>
<b>Word Processing</b>
<b>GLOBAL PAGES &gt;&gt;</b>
<b>NEWS ARCHIVE &gt;&gt;</b>
<b>DOFTPEDIA REVIEWS &gt;&gt;</b>
<b>MEET THE EDITORS &gt;&gt;</b>

variability

**Power Matte 2.1.3 update**



Adobe Photoshop plugin that can extract a subject in an image

[read more >]

**Grid 1.1 update**



Helps you generate perspective grids

[read more >]

**Picture Frame 2.2 update**



Quickly generate multi-frame photos using your Mac

[read more >]

**FashionLab Studio 1.1 update**



Makes it easy to design your own T-shirt using a Mac

[read more >]

**Size:** 13.20 MB

**Platform:** Mac OS X 10.5 or later

**License:** Trial

**Rating:** Good (3.0/5)

**Downloads:** 1,504

**Updated:** June 20th, 08:21 UTC



**Size:** 102 KB

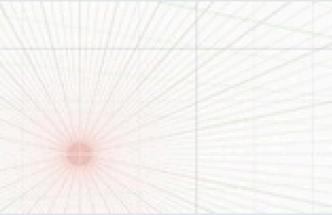
**Platform:** Mac OS X 10.8 or later

**License:** Commercialware

**Rating:** NOT RATED

**Downloads:** 21

**Updated:** June 20th, 07:56 UTC



**Size:** 716 KB

**Platform:** Mac OS X 10.6.6 or later

**License:** Commercialware

**Rating:** Excellent (5.0/5)

**Downloads:** 297

**Updated:** June 20th, 07:53 UTC



**Size:** 3.10 MB

**Platform:** Mac OS X 10.6.6 or later

**License:** Commercialware

**Rating:** NOT RATED

**Downloads:** 3

**Updated:** June 20th, 07:49 UTC





# RENAULT VANS



CARS | VANS | ELECTRIC VEHICLES | RENAULT BUSINESS | USED CARS | OWNER SERVICES | ABOUT RENAULT | RENAULT SHOP **NEW**

Renault UK > Renault Vans > New Kangoo Van Range > Kangoo Van > Build your own Kangoo Van > Select Options

## NEW KANGOO VAN RANGE

01 Preferences

02 Version

03 Equipment & options

< Previous

Next >

### OPTIONS

#### > COMFORT

- |   |        |
|---|--------|
| <input checked="" type="checkbox"/> Central storage console & armrest between seats | £50.00 |
|---|--------|

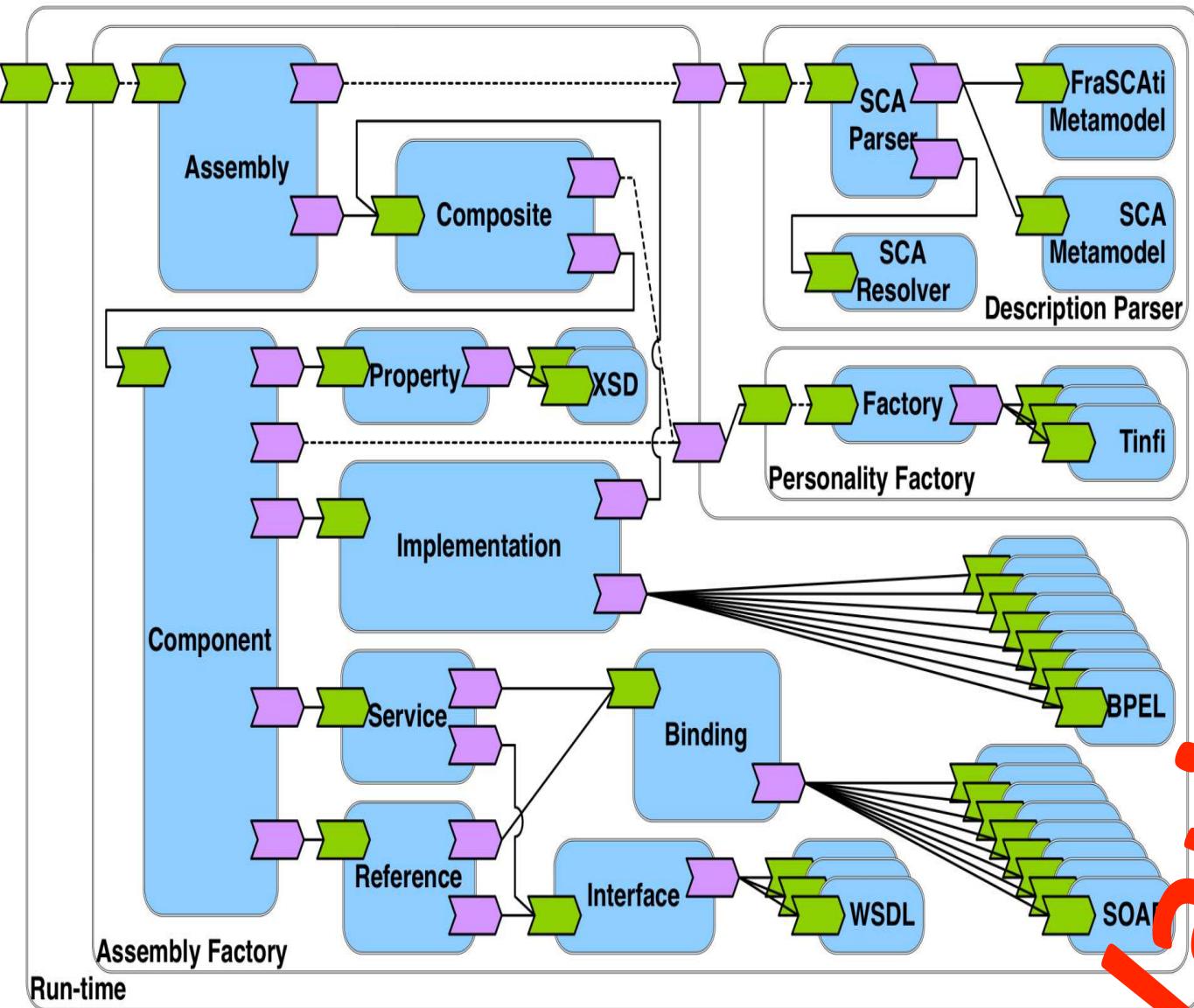
#### > DRIVING

- |  |       |
|--|-------|
| <input type="checkbox"/> Electric door mirrors | £0.00 |
|--|-------|

#### > SAFETY & SECURITY

- |   |         |
|---|---------|
| <input checked="" type="checkbox"/> ESC (Electronic Stability Control) with traction and understeer control | £200.00 |
|---|---------|





**Variability**

A large red arrow points diagonally upwards and to the right, highlighting the concept of variability in the system.



(a) Variant #1 of video sequence



(b) Variant #2 of video sequence



(c) Variant #3 of video sequence



(d) Variant #4 of video sequence



(e) Variant #5 of video sequence



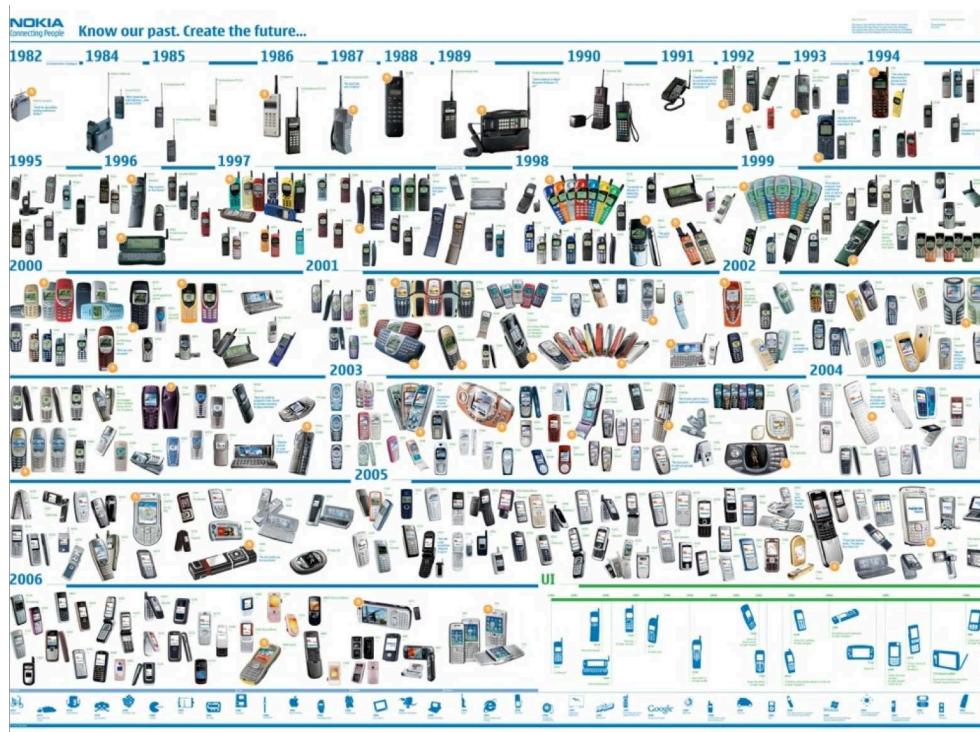
(f) Variant #6 of video sequence

Figure 1: Six variants of video sequences synthesized with ViViD



# Variability in time vs in space

- **Variability in Time (releases)**
  - the existence of different **versions** of an artifact that are valid at different times
- **Variability in Space (variants)**
  - the existence of an artifact in different **shapes** at the same time



# Benefits

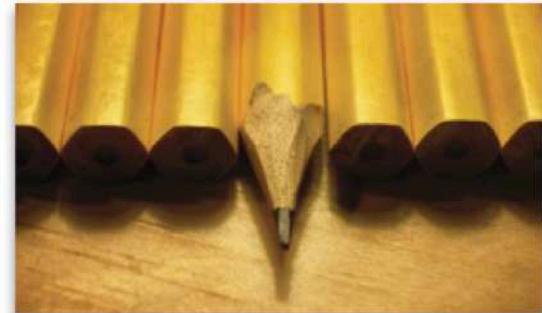
Improve product reliability



Improve usability



Improve consistency across products...



# Benefits

Reduce production costs



Reduce certification costs



Shorten time-to-market



# Hall of Fame

[splc.net/fame.html](http://splc.net/fame.html)



**BOSCH**

Invented for life



**PHILIPS**



**NOKIA**  
Connecting People

**CelsiusTech**

**ERICSSON**



**Lucent Technologies**  
Bell Labs Innovations



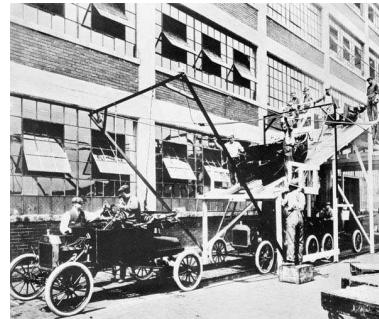


# Printer Firmware

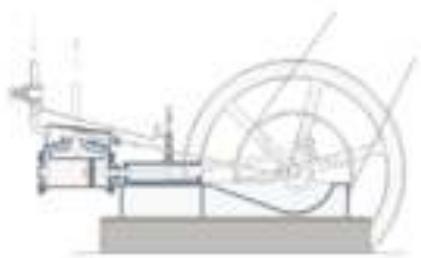
- Production cost reduced by 75%
- Development time reduced by 33%
- Reported defects reduced by 96%



# A Bit of History: Industrial Revolution



1901  
Henry Ford



1698  
Thomas Savery



1980s

# Nowaday: Product Lines Everywhere



# LE PLIAGE PERSONNALISÉ

MODÈLES

COULEUR  
RECTO

COULEUR  
VERSO

BOUCLERIE

RESET



LE PLIAGE CUIR

LE PLIAGE TOILE



## VOTRE PERSONNALISATION

Porte-monnaie Toile : 9 x 7 x 5 cm  
Couleur recto : Garance  
Couleur verso : Malabar  
Bouclerie : Bronze

35,00 € AJOUTER AU PANIER

Infos   Partager   J'aime

# Product Lines of Cars



This image may contain optional equipment. [360°](#)

**Agila, Club**  
1.2i 16v, 5 Speed  
Blaze Red, Melt / Elba Charcoal  
**Total** € 15,684.00

[Exterior](#) | [Interior](#)      [Side](#) | [Front](#) | [Rear](#)      [360°](#)

[1. Trims/Series](#) | [2. Engine/Transmission](#) | [3. Colour & Style](#) | [4. Options](#) | [5. Summary](#)      [Next Step](#)

**Choose Your Options**

<input type="checkbox"/> CD 30	Standard	- MP3 CD player with MP3 format, stereo radio, steering wheel mounted audio controls
<input checked="" type="checkbox"/> Air conditioning		€ 923.00
<input checked="" type="checkbox"/> Electronic Stability Programme (ESP)		€ 411.00
<input checked="" type="checkbox"/> Emergency tyre inflation kit in lieu of space-saver spare wheel and tyre	Standard	

[Audio/Comms/Nav](#) | [Heating/Ventilation](#) | [Mechanical](#) | [Safety/Security](#) | [A-Z](#)

**Pricing Details**

Club	€ 14,350.00
1.2i 16v, 5 Speed	
Blaze Red	€ 0.00
Melt / Elba Charcoal	€ 0.00
15-inch steel wheels with 185/60 R 15 tyres and flush wheel covers	€ 0.00
<b>Options (2)</b>	
You selected:	
<input checked="" type="checkbox"/> Air conditioning	€ 923.00
<input checked="" type="checkbox"/> Electronic Stability Programme (ESP)	€ 411.00
<b>Total</b>	<b>€ 15,684.00</b>

[Next Step: Summary](#)

**Legend**

- Selected Option
- Selectable Option
- Option contained in an option pack
- Option contained in an option pack or standard equipment which has been replaced by another option
- Option that is only selectable together with another option. Please click for details

**Willkommen bei selve - the shoe individualizer**

http://www.selve.net/index\_js.html

KOLLEKTION FÜSSTYP HYSELVE INFO HOME

MODELLE  
LOOKBOOK

SELVE\_ID  
PASSWORD  
>>ANMELDEN

selve Kollektion -> Style: casuals -> Modell: Opal

modell-details >> hier clicken

>> SELVE SCHUHREGAL Inhalt:0  
>> SHOPPING BAG Inhalt:0

A. Erstes Oberleder  
Veloursleder Sand

B. Veloursleder Bordeaux  
Veloursleder Cognac

C. Futterleder  
Beige

D. Absatz  
Hufeisen Braun

E. Sohle  
Gummisohle

>>ÄNDERN  
>>ZURÜCKLEGEN

Müsli individuell online mixen! Bio-Müsli. - Mozilla Firefox

File Edit View History Bookmarks Tools Help

m http://www.mymuesli.com/muesli/index.php?vw=mixer&ec=step1&mnid=1&mnpt=1&type=t0 softwareproduktlinien ABP S

Müsli individuell online mixen! Bio-M... +

my muesli custom-mixed cereals

muesli mixer blog fragen about us

Müslibasis Basis verfeinern Früchte Nüsse & Kerne Extras

**Früchte**

Köstliche Bio-Trockenfrüchte, müsigerecht aufbereitet. Du kannst eine Frucht auch mehrmals auswählen, um deren Anteil zu steigern.

**Ananas**  
lecker, exotisch und wunderbar | 0.65€ (30g)  
[mehr Infos](#)

**Apfelstücke**  
Ohne Worte weil Klassiker | 0.45€ (25g)  
[mehr Infos](#)

**Aprikosen**

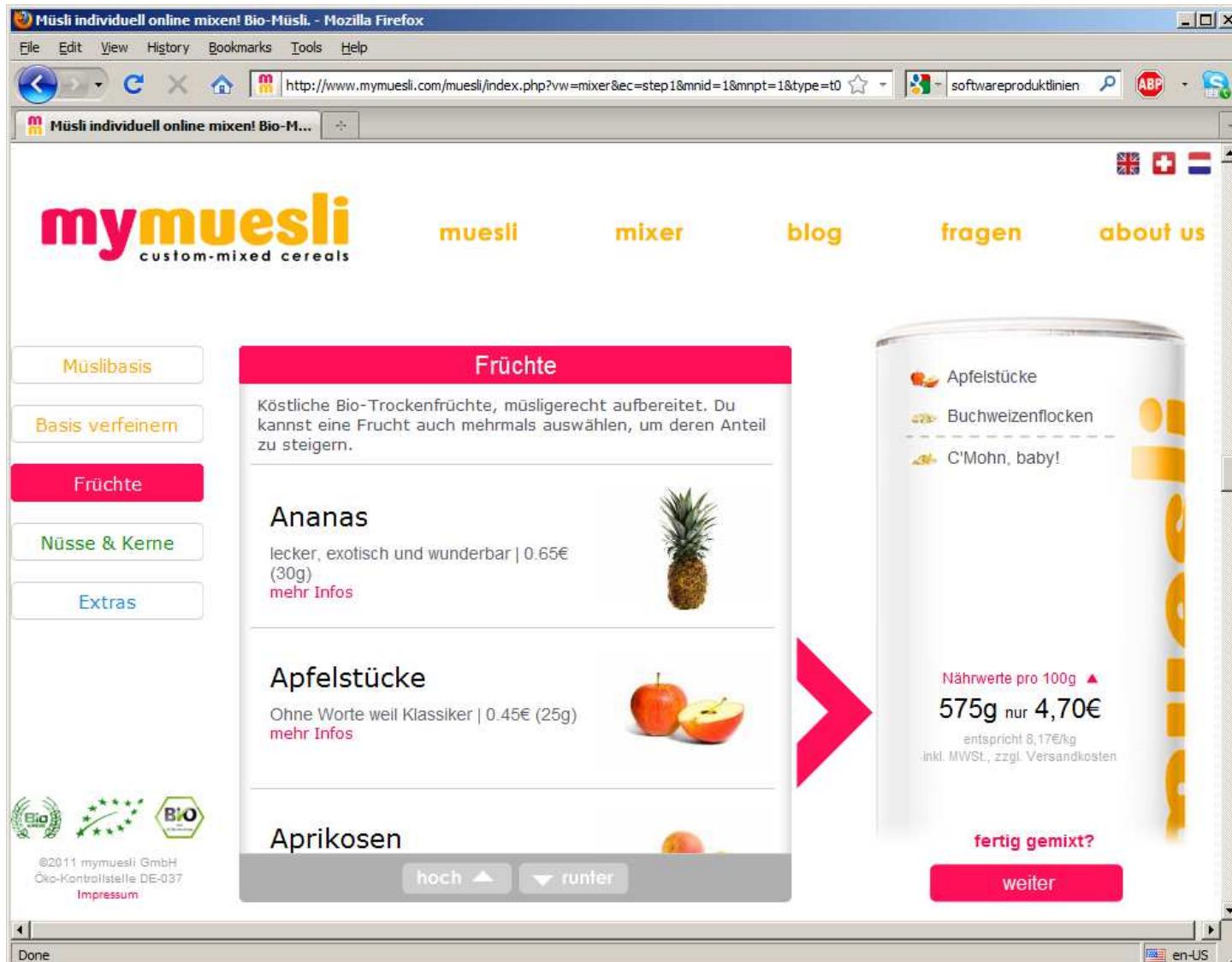
hoch ▲ ▼ runter

Apfelstücke  
Buchweizenflocken  
C'Mohn, baby!

Nährwerte pro 100g ▲  
**575g nur 4,70€**  
entspricht 8.17€/kg  
inkl. MWSt., zzgl. Versandkosten

fertig gemixt?  
weiter

Done en-US



Der Dell Online-Shop: Stellen Sie Ihr eigenes System zusammen - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

Getting Started Latest Headlines

<http://configure2.euro.dell.com/dellstore/config.aspx?c=de&cs=dedhs1&kc=3058j=de&oc=W06390xp&s=dhs&sbc=pr>

Bestellen Sie online oder wählen Sie 0800 533 55 40 03(gebührenfrei)

**DELL** Produkte Service Support Einkaufsunterstützung Suche

Dell empfiehlt Windows Vista™ Home Premium.

Sie befinden sich hier: Deutschland > PRIVATANWENDER

**1 Meinen Dell konfigurieren** **2 Zubehör auswählen** **3 Elektronik** **4 Software & Service** **5 Bestätigen & zum Warenkorb hinzufügen**

Als Symbol anzeigen

128 MB nVidia NVS285 DVI/VGA-Grafikkarte

**Grafikkarte**

Auswahlhilfe

- 256 MB ATI Fire GL V7200-Grafikkarte [plus 416,50 € oder 13 €/Monat<sup>1</sup>]
- 128 MB nVidia Quadro FX550-Grafikkarte [plus 69,02 € oder 2 €/Monat<sup>1</sup>]
- 256 MB nVidia Quadro FX3450-Grafikkarte [plus 547,40 € oder 17 €/Monat<sup>1</sup>]
- 128 MB nVidia NVS285 DVI/VGA-Grafikkarte [Im Preis enthalten]
- Grafikkarte PCIe x16 (DVI/VGA) Matrox QID LP PCIe, 128 MB, DVI- oder VGA-Grafikkarte für 4 Monitore [plus 630,70 € oder 20 €/Monat<sup>1</sup>]
- 128 MB ATI Fire GL V3400-Grafikkarte [plus 44,03 € oder 1 €/Monat<sup>1</sup>]

**Festplatte**

Auswahlhilfe

- 80 GB Serial ATA-II-Festplatte (7.200 U/min) mit NCQ [Im Preis enthalten]
- 160 GB Serial ATA-II-Festplatte (7.200 U/min) mit NCQ [plus 16,66 €]
- 80 GB Serial ATA-II-Festplatte (7.200 U/min) mit NCQ [Im Preis enthalten]

**Sicher Einkaufen mit Trusted Shops und Old-zurück-Garantie.**

**Dell Precision™ 390 Essential (W06390xp)**

inkl. MwSt., zzgl. 19,04 € Versand  
\*\*Ermäßiger Sonderpreis\*\*  
Es gelten keine zusätzlichen Preisnachlässe.  
Das Angebot gilt für maximal 5 Systeme

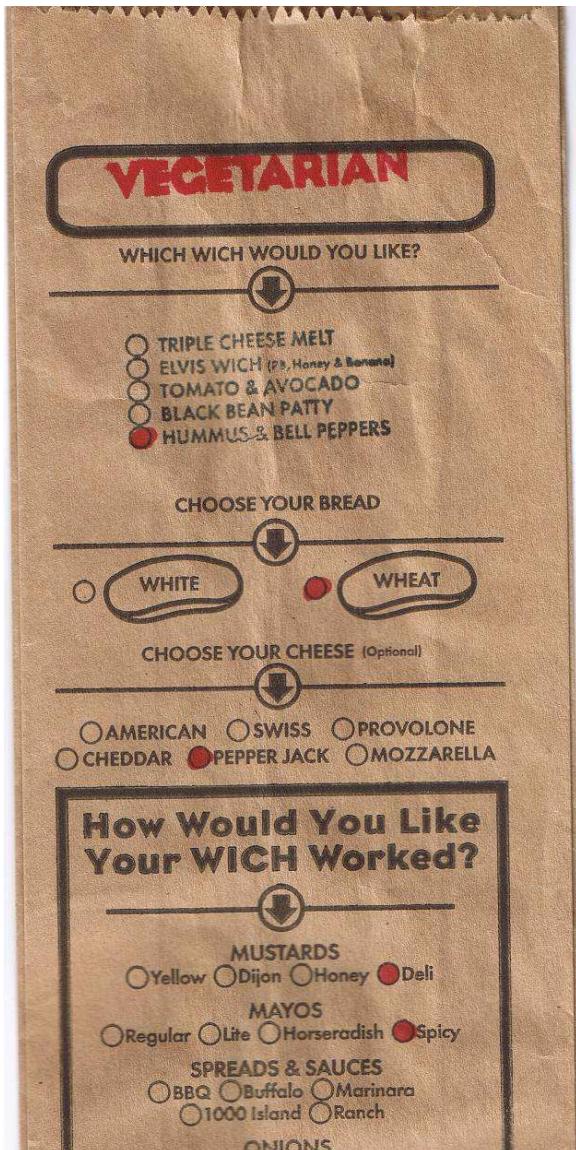
**913,92 €**

Finanzierung ab **30 €/mtl.**  
Jetzt finanzieren - erst ab Januar 2008 zahlen!  
Weitere Informationen zur Ratenfinanzierung

Für einen noch umfassenderen Schutz Ihres Systems beinhaltet der oben erwähnte Preis ein Upgrade Service Paket. Um auf den beworbenen Preis zu kommen, entmarkieren Sie die Kategorie "Business Support".

Transferring data from i.dell.com...

# Food? Product lines!







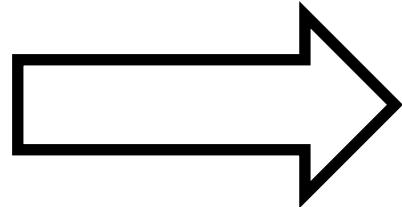
# Mass production

What about  
software?

**Product lines of  
software intensive systems**

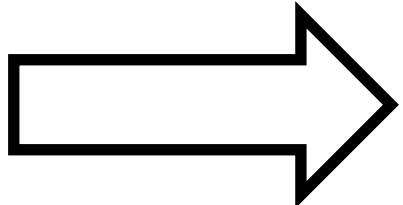
# Software intensive systems

are declined in many variants



# Software intensive systems

are declined in many variants



# Software Product Lines

A graphic featuring a large hamburger on the right and a paper menu on the left. The menu has sections for "VEGETARIAN" sandwiches, bread choices, cheese options, and toppings like mustard, lettuce, and onions.

**VEGETARIAN**

WHICH WICH WOULD YOU LIKE?

- TRIPLE CHEESE MELT
- ELVIS (Chicken, Bacon & Honey)
- TOMATO & AVOCADO
- BLACK BEAN PATTY
- HUMMUS & BELL PEPPERS

CHOOSE YOUR BREAD

- WHITE
- WHEAT

CHOOSE YOUR CHEESE (Optional)

- AMERICAN
- SWISS
- PROVOLONE
- CHEDDAR
- PEPPER JACK
- MOZZARELLA

**How Would You Like Your WICH Worked?**

MUSTARDS

- Yellow
- Dijon
- Honey
- Deli

MAYOS

- Regular
- Old fashioned
- Ranch
- 1000 Island
- Spicy

SPREADS & SAUCES

- Mayo
- Buffalo
- Marinara
- Cole slaw
- Bell Peppers

ONIONS

- Red
- Grilled
- Crispy Strings

VEGGIES

- Lettuce
- Tomato
- Pickles
- Jalapenos
- Olive Salad
- Mushrooms
- Sauerkraut
- Cole slaw
- Bell Peppers

OILS & SPICES

- Salt
- Pepper
- Oregano
- Parmesan

EXTRAS (.75¢ Each)

- Bacon
- Avocado
- Pickle (Whole)
- More Meat
- More Cheese

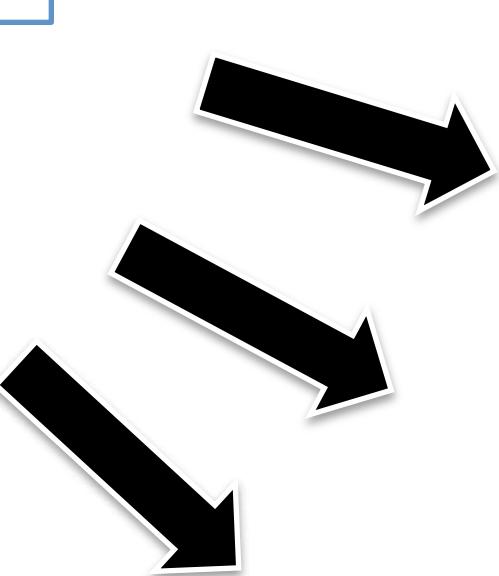
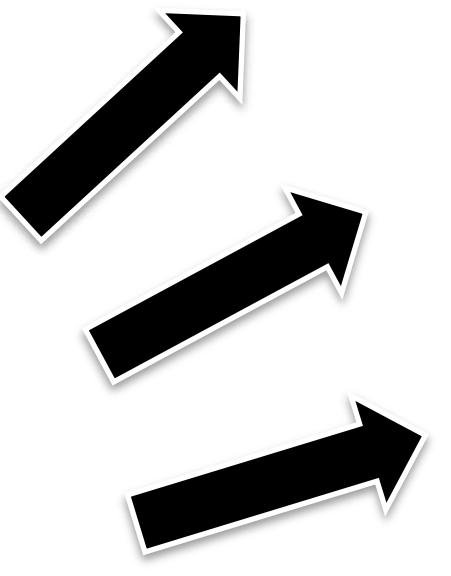
WRITE YOUR NAME HERE



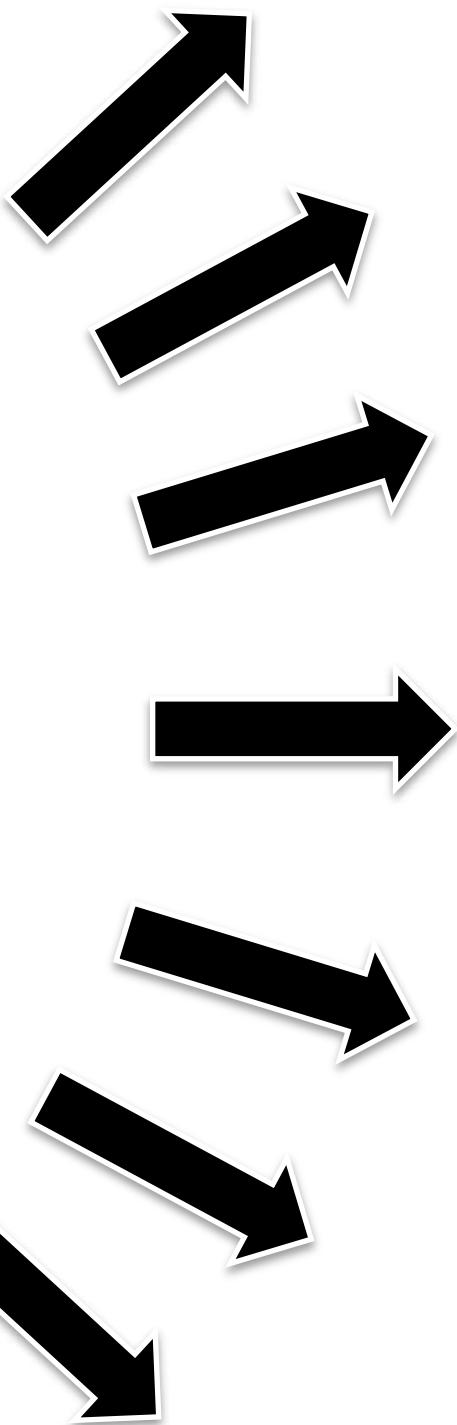
```
01011011  
11011110  
00110110  
11001101  
10001111  
10100110  
10001010  
10101011  
00001110  
11010101  
11011101  
01100100  
01010101  
11010110  
10101101
```



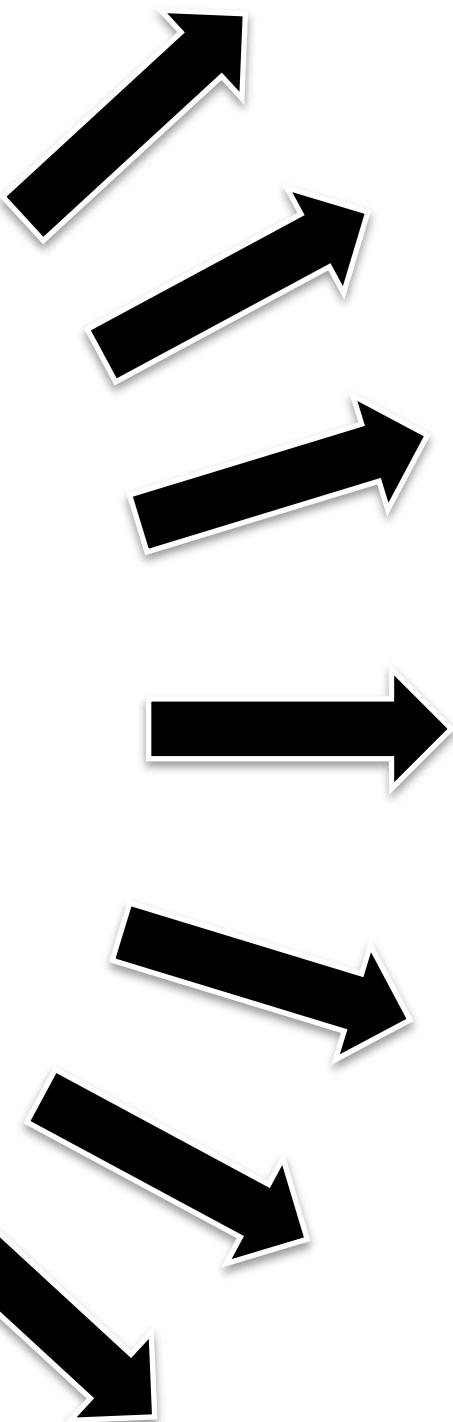
Car



# Database Engine

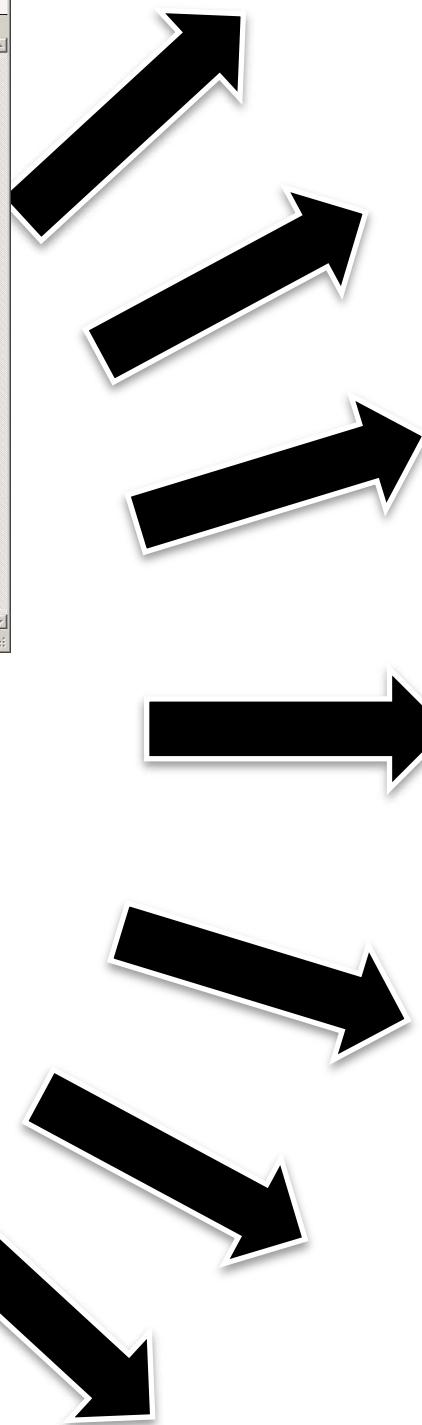


# Printer Firmware



# Linux Kernel

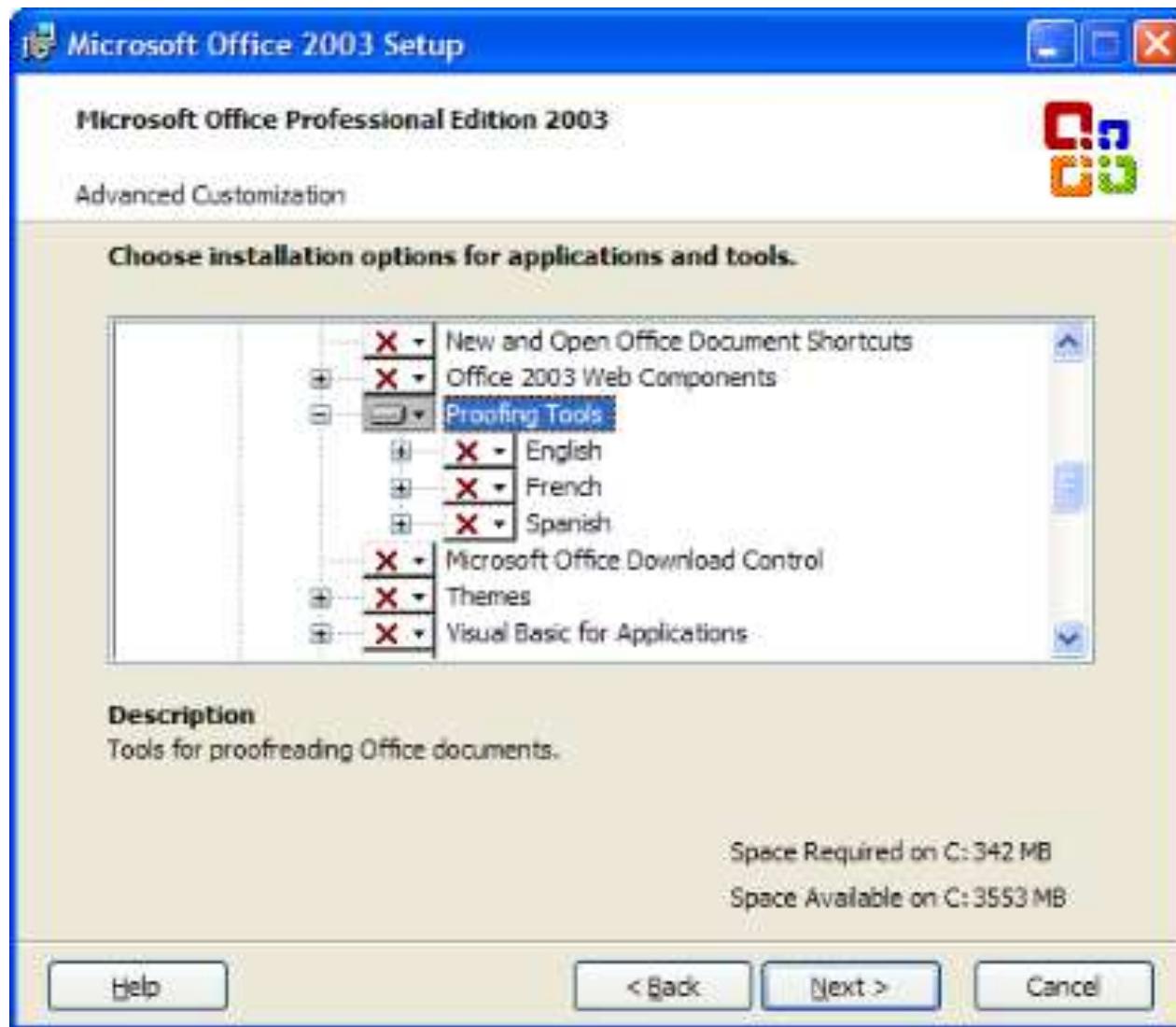
```
Line feed: CR Encoding: iso-8859-1 generic .config - Linux Kernel v2.6.33.3 Configuration Processor type and features Arrow keys navigate the menu. <Enter> selects submenus -->. Highlighted letters are hotkeys. Pressing <> includes, <> excludes, <> modularizes features. Press <Esc><Esc> to exit, <> for Help, <> for Search. Legend: [*] built-in [ ] excluded [<> module] <> module capable [ ] Tickless System (Dynamic Ticks) [ ] High Resolution Timer Support [ ] Symmetric Multi-processing support [ ] Support for extended (non-PC) x86 platforms [ ] Single-depth KCHAN output [ ] Paravirtualized guest support ... Memtest [ ] Processor family (Generic-x86-64) ---> [ ] Preemption Model (No Forced Preemption (Server)) ---> [ ] Renote for broken boot IRQ (NEW) [ ] Machine Check / overheating reporting [ ] Dell laptop support [ ] /dev/cpu/microcode - microcode support [ ] /dev/cpu/*msr - Model-specific register support [ ] /dev/cpu/*cpuid - CPU information support [ ] Memory model (Sparse Memory) ---> [*] Sparse Memory Virtual memmap (NEW) [ ] Allow for memory hot-add (NEW) [ ] Enable KSM for page merging (4096) Low address space to protect from user allocation [ ] Check for low memory corruption [ ] Reserve low 64M of RAM on AMI/Phoenix BIOSes [ ] MTTR (Memory Type Range Register) support [ ] MTTR cleanup support [ ] Enable seccomp to safely compute untrusted bytecode [ ] Enable -fstack-protector buffer overflow detection (EXPERIMENTAL) [ ] Timer frequency (250 Hz) ---> [ ] kexec system call <(<> Select> < Exit > < Help >
```



# Linux-Kernel



# Features in Microsoft Office



Bref

bref.  
CANAL à 30 ans.

ETAPE 1 : DONNE TON PRENOM

MATHIEU

→ OK

# Online Generator

bref30ans.canalplus.fr/#c

## **ÉTAPE 2 : CHOISIS 3 BONS SOUVENIRS**



# Variant



# Question

Give three examples of software product lines  
(also called configurable systems or variability-intensive systems)

# Variability

“the ability of a system to be efficiently extended, changed, customized or configured for use in a particular context”

*Mikael Svahnberg, Jilles van Gurp, and Jan Bosch (2005)*



A large, intricate 3D white maze is set against a light gray background. The maze is composed of many interconnected paths and dead ends, creating a complex and challenging structure.

**Variability = Complexity**

optional, independent

# 33 features



a unique variant for every  
person on this planet

320<sup>optional, independent</sup>  
features

more variants than estimated  
atoms in the universe



2000 features

10000  
features



The development of a  
**family** of software systems

differs from the development of  
a **single** software system

**THANKS CAPTAIN  
OBVIOUS**



« The development of a  
**family** of software systems  
differs from the development of  
a **single** software system »

# Reuse

# *Commonality*

# Customization

# *Variability*

# Automation

A photograph of a car assembly line. In the foreground, a worker wearing a white shirt and red overalls is leaning into the open driver-side door of a silver car, working on the interior. Behind him, several other cars are lined up along the assembly line. The background shows the industrial interior of a factory with various equipment and a digital display showing numbers.

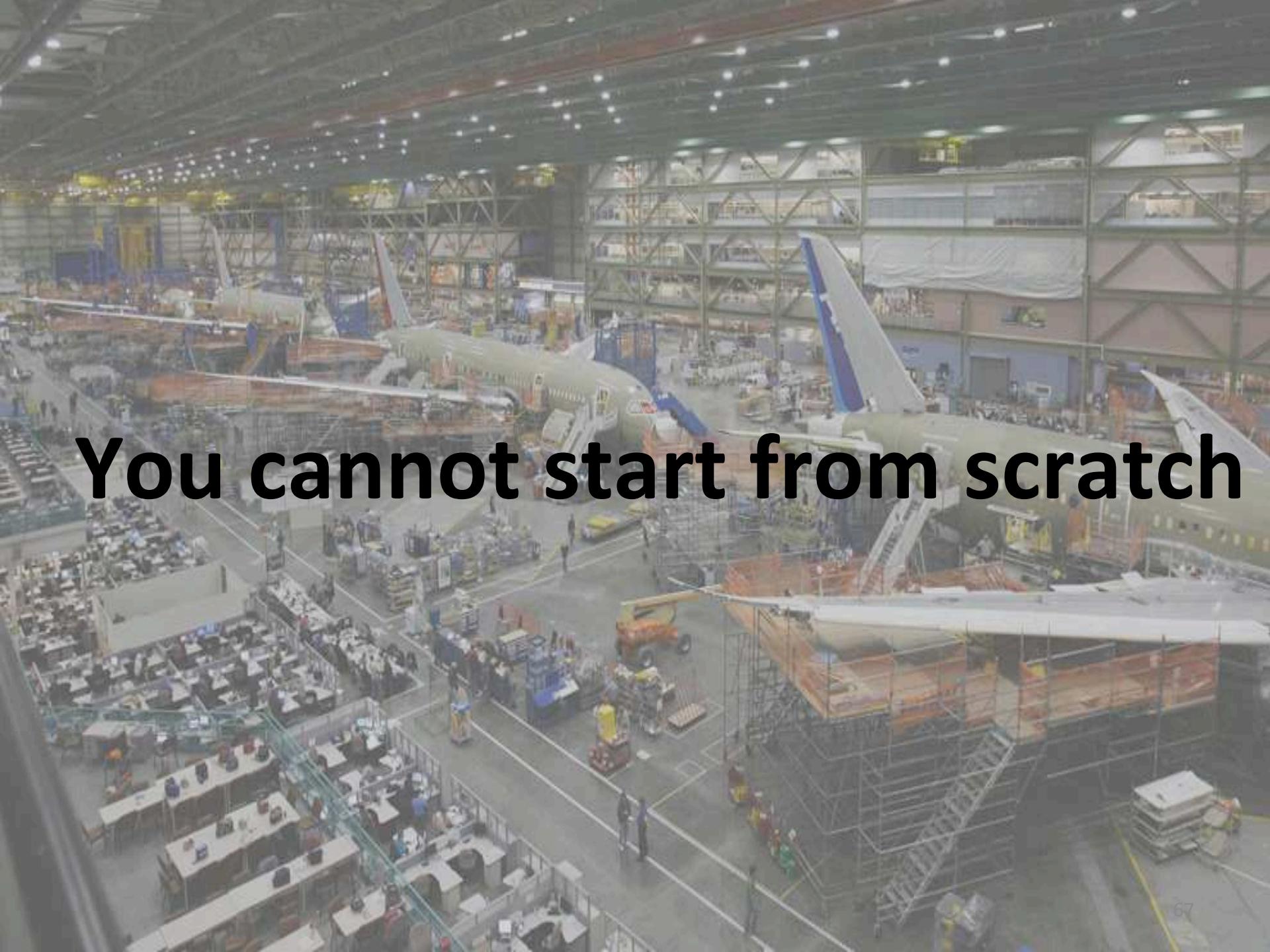
# Assembly Line and Mass Customization



# **Reuse and Mass Customization**



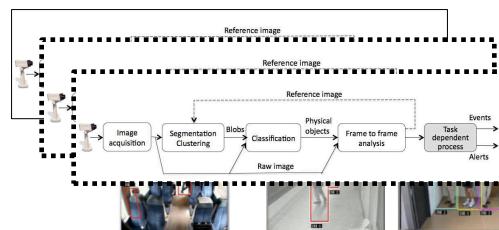
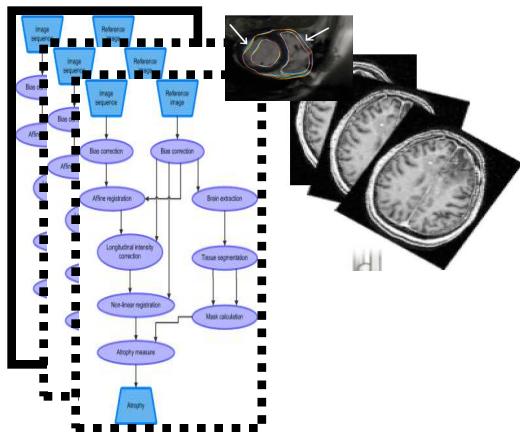
# Starting from scratch?

An aerial photograph of a massive industrial hangar, likely at an aerospace manufacturing facility. The hangar is filled with the skeletal frames of several large aircraft, possibly Boeing 787s, which are characterized by their distinctive white fuselage and blue-tipped wings. The floor of the hangar is a complex network of walkways, scaffolding, and equipment. In the foreground, there are several rows of small, enclosed workstations or offices. The overall scene conveys a sense of a major engineering and manufacturing operation.

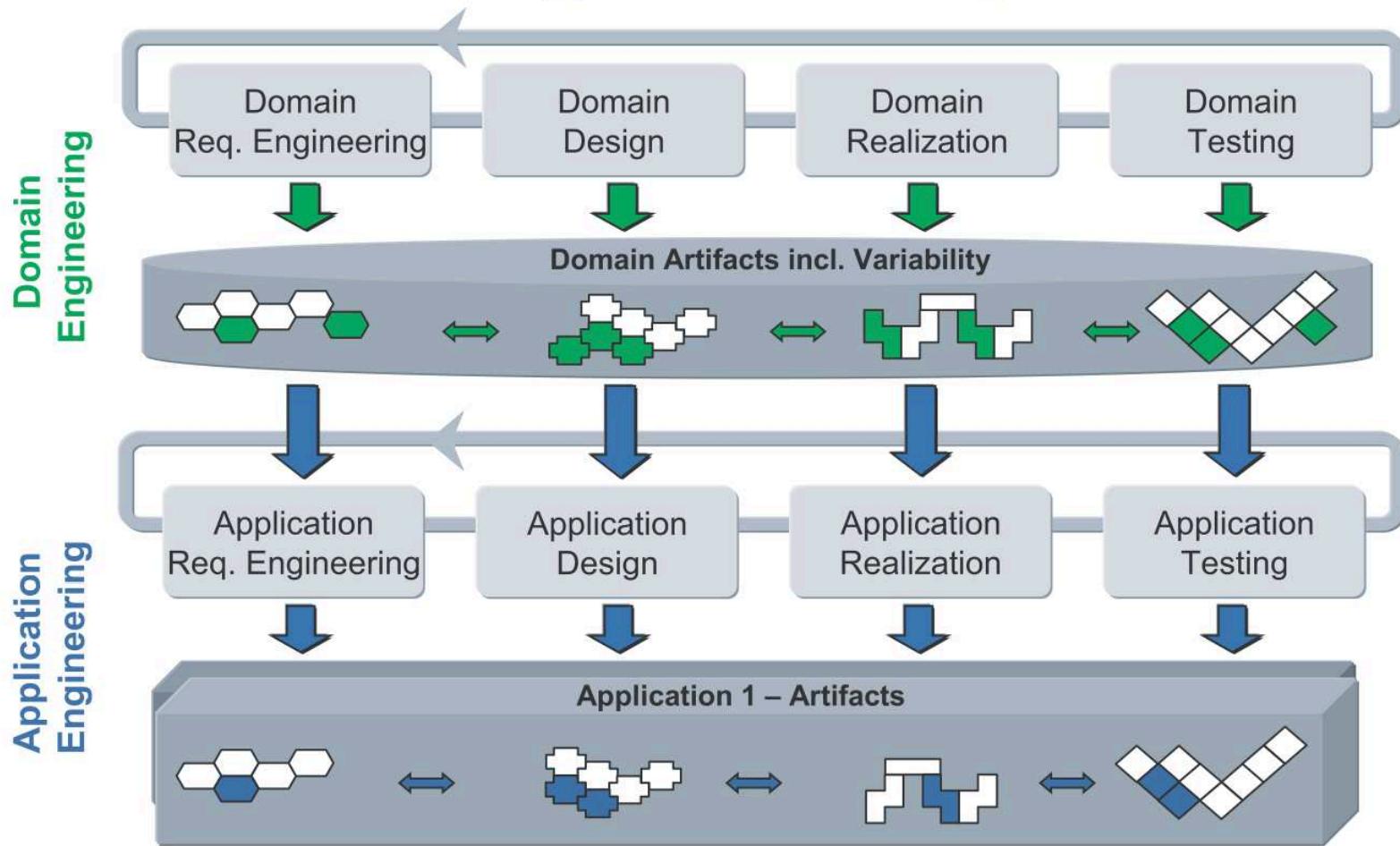
**You cannot start from scratch**

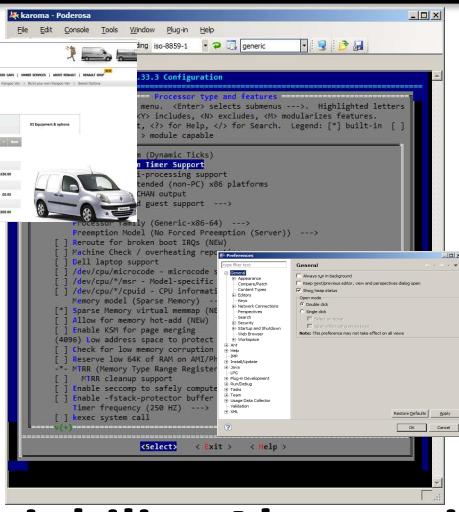
*“a set of software- intensive systems that share a common, managed set of features satisfying the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way” [Clements et al., 2001]*

# Software Product Lines

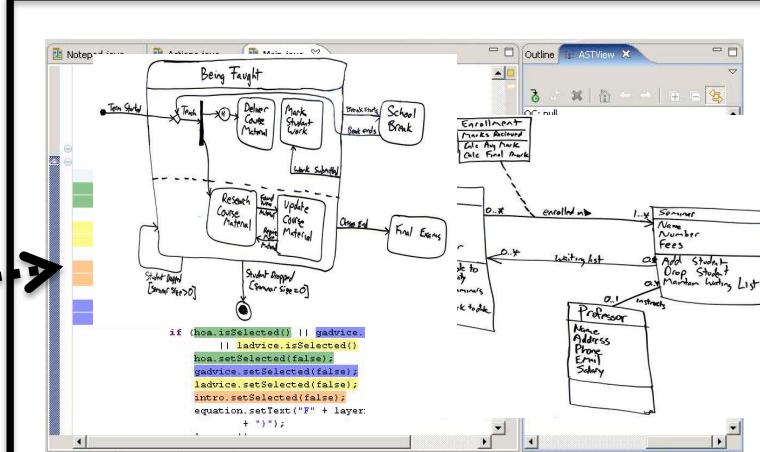


# Software Product-Line Engineering





**Variability Abstraction Model (VAM)**



**Variability Realization Model (VRM)**

**Domain Artefacts (e.g., models)**



**Configuration  
(resolution model)**



**Software Generator  
(derivation engine)**



# Mapping: an example

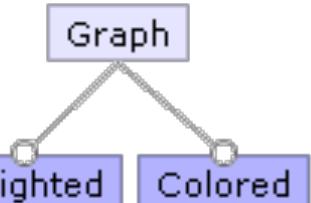
```
class Graph {  
    Vector nv = new Vector(); Vector ev = new Vector();  
    Edge add(Node n, Node m) {  
        Edge e = new Edge(n, m);  
        nv.add(n); nv.add(m); ev.add(e);  
        e.weight = new Weight();  
        return e;  
    }  
    Edge add(Node n, Node m, Weight w)  
    Edge e = new Edge(n, m);  
    nv.add(n); nv.add(m); ev.add(e);  
    e.weight = w; return e;  
}  
void print() {  
    for(int i = 0; i < ev.size(); i++) {  
        ((Edge)ev.get(i)).print();  
    }  
}
```

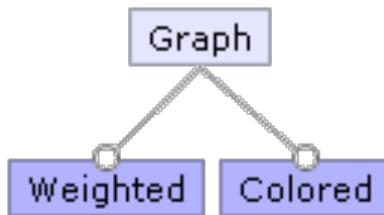
```
class Node {  
    int id = 0;  
    Color color = new Color();  
    void print() {  
        Color.setDisplayColor(color);  
        System.out.print(id);  
    }  
}
```

```
class Edge {  
    Node a, b;  
    Color color = new Color();  
    Weight weight = new Weight();  
    Edge(Node _a, Node _b) { a = _a; b = _b; }  
    void print() {  
        Color.setDisplayColor(color);  
        a.print(); b.print();  
        weight.print();  
    }  
}
```

```
class Color {  
    static void setDisplayColor(Color c) { ... }  
}
```

```
class Weight { void print() { ... } }
```





```

class Graph {
    Vector nv = new Vector(); Vector ev = new Vector();
    Edge add(Node n, Node m) {
        Edge e = new Edge(n, m);
        nv.add(n); nv.add(m); ev.add(e);
        /*if[WEIGHT]*/
        e.weight = new Weight();
        /*end[WEIGHT]*/
        return e;
    }
    /*if[WEIGHT]*/
    Edge add(Node n, Node m, Weight w)
        Edge e = new Edge(n, m);
        nv.add(n); nv.add(m); ev.add(e);
        e.weight = w; return e;
    }
    /*end[WEIGHT]*/
    void print() {
        for(int i = 0; i < ev.size(); i++) {
            ((Edge)ev.get(i)).print();
        }
    }
}

/*if[WEIGHT]*/
class Weight { void print() { ... } }
/*end[WEIGHT]*/

```

```

class Edge {
    Node a, b;
    /*if[COLOR]*/
    Color color = new Color();
    /*end[COLOR]*/
    /*if[WEIGHT]*/
    Weight weight;
    /*end[WEIGHT]*/
    Edge(Node _a, Node _b) { a = _a; b = _b; }
    void print() {
        /*if[COLOR]*/
        Color.setDisplayColor(color);
        /*end[COLOR]*/
        a.print(); b.print();
        /*if[WEIGHT]*/
        weight.print();
        /*end[WEIGHT]*/
    }
}

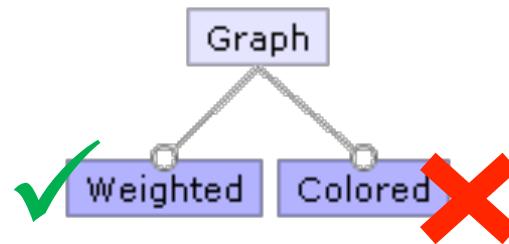
/*if[COLOR]*/
class Color {
    static void setDisplayColor(Color c) { ... }
}
/*end[COLOR]*/

```

```

class Node {
    int id = 0;
    /*if[COLOR]*/
}

```



```

class Graph {
    Vector nv = new Vector(); Vector ev = new Vector();
    Edge add(Node n, Node m) {
        Edge e = new Edge(n, m);
        nv.add(n); nv.add(m); ev.add(e);
        e.weight = new Weight();
        return e;
    }
    Edge add(Node n, Node m, Weight w)
        Edge e = new Edge(n, m);
        nv.add(n); nv.add(m); ev.add(e);
        e.weight = w; return e;
    }
    void print() {
        for(int i = 0; i < ev.size(); i++) {
            ((Edge)ev.get(i)).print();
        }
    }
}
  
```

```

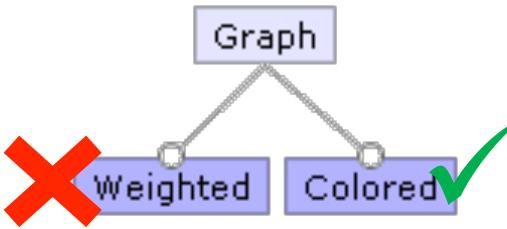
class Edge {
    Node a, b;
    Weight weight;
    Edge(Node _a, Node _b) { a = _a; b = _b; }
    void print() {
        a.print(); b.print();
        weight.print();
    }
}
  
```

```

class Node {
    int id = 0;
    void print() {
        System.out.print(id);
    }
}
  
```

```

class Weight { void print() { ... } }
  
```



```

class Graph {
    Vector nv = new Vector(); Vector ev = new Vector();
    Edge add(Node n, Node m) {
        Edge e = new Edge(n, m);
        nv.add(n); nv.add(m); ev.add(e);
    return e;
    }
    void print() {
        for(int i = 0; i < ev.size(); i++) {
            ((Edge)ev.get(i)).print();
        }
    }
}

```

```

class Edge {
    Node a, b;
    Color color = new Color();
    Edge(Node _a, Node _b) { a = _a; b = _b; }
    void print() {
        Color.setDisplayColor(color);
        a.print(); b.print();
    }
}

```

```

class Color {
    static void setDisplayColor(Color c) { ... }
}

```

```

class Node {
    int id = 0;
    Color color = new Color();
    void print() {
        Color.setDisplayColor(color);
        System.out.print(id);
    }
}

```

[generator-jhipster / app / templates / src / main / java / package / config / \\_DatabaseConfiguration.java](#) **jdubois** 2 days ago Use Spring Boot's configuration meta-data9 contributors 

184 lines (165 sloc) | 9.69 KB

[Raw](#) [Blame](#) [History](#)   

```
1 package <%=packageName%>.config;
2 <% if (databaseType == 'sql') { %>
3 import <%=packageName%>.config.liquibase.AsyncSpringLiquibase;
4 import com.codahale.metrics.MetricRegistry;
5 import com.fasterxml.jackson.datatype.hibernate4.Hibernate4Module;
6 import com.zaxxer.hikari.HikariConfig;
7 import com.zaxxer.hikari.HikariDataSource;
8 import liquibase.integration.spring.SpringLiquibase;<% } %><% if (databaseType == 'mongodb' && authenticationType == 'oauth2') { %>
9 import <%=packageName%>.config.oauth2.OAuth2AuthenticationReadConverter;<% } %><% if (databaseType == 'mongodb') { %>
10 import com.mongodb.Mongo;
11 import org.mongeez.Mongeez;<% } %>
12 import org.slf4j.Logger;
13 import org.slf4j.LoggerFactory;<% if (databaseType == 'sql') { %><% if (hibernateCache == 'hazelcast') { %>
14 import org.springframework.cache.CacheManager;<% } %>
15 import org.springframework.beans.factory.annotation.Autowired;
16 import org.springframework.boot.autoconfigure.condition.ConditionalOnExpression;<% } %><% if (databaseType == 'mongodb') { %>
17 import org.springframework.boot.autoconfigure.mongo.MongoAutoConfiguration;
18 import org.springframework.boot.autoconfigure.mongo.MongoProperties;<% } %><% if (databaseType == 'sql') { %>
19 import org.springframework.boot.autoconfigure.jdbc.DataSourceProperties;
20 import org.springframework.boot.autoconfigure.liquibase.LiquibaseProperties;
21 import org.springframework.context.ApplicationContextException;<% } %>
22 import org.springframework.context.annotation.Bean;
23 import org.springframework.context.annotation.Configuration;
24 import org.springframework.context.annotation.Profile;<% if (databaseType == 'mongodb') { %>
25 import org.springframework.context.annotation.Import;<% } %><% if (databaseType == 'sql') { %>
26 import org.springframework.core.env.Environment;<% } %><% if (databaseType == 'mongodb' && authenticationType == 'oauth2') { %>
27 import org.springframework.core.convert.converter.Converter;<% } %><% if (databaseType == 'mongodb') { %>
28 import org.springframework.core.io.ClassPathResource;<% } %><% if (searchEngine == 'elasticsearch') { %>
29 import org.springframework.data.elasticsearch.repository.config.EnableElasticsearchRepositories;<% } %><% if (databaseType == 'mon
30 import org.springframework.data.mongodb.config.AbstractMongoConfiguration;
31 import org.springframework.data.mongodb.config.EnableMongoAuditing;<% } %><% if (databaseType == 'mongodb' && authenticationType =
32 import org.springframework.data.mongodb.core.convert.CustomConversions;<% } %><% if (databaseType == 'mongodb') { %>
33 import org.springframework.data.mongodb.core.mapping.event.ValidatingMongoEventListener;
34 import org.springframework.data.mongodb.repository.config.EnableMongoRepositories;
35 import org.springframework.validation.beanvalidation.LocalValidatorFactoryBean;<% } %><% if (databaseType == 'sql') { %>
```

macher-wifi:getting-started macher1\$ yo jhipster

I'm all done. Running `npm install & bower install` for you to install the required dependencies.

# JHIPSTER STARER IF YOU DON'T KNOW

Welcome to the JHipster Generator v2.17.0

- ? (1/15) What is the base name of your application? **jhipster**
- ? (2/15) What is your default Java package name? **com.mycompany.myapp**
- ? (3/15) Do you want to use Java 8? Yes (use Java 8)
- ? (4/15) Which type of authentication would you like to use? (Use arrow keys)
  - > **HTTP Session Authentication (stateful, default Spring Security mechanism)**
  - OAuth2 Authentication (stateless, with an OAuth2 server implementation)
  - Token-based authentication (stateless, with a token)

## Variability Model



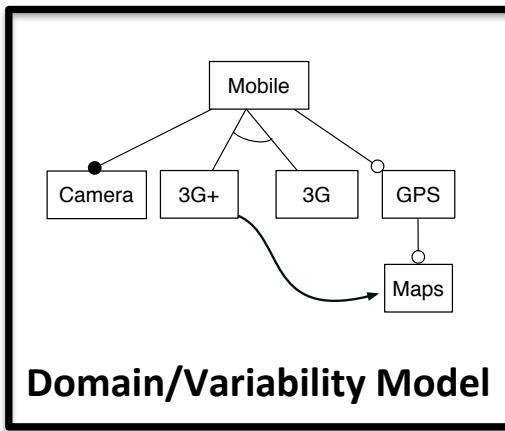
mapping

```
1 package io.jhipster.generator.config;
2
3 import com.codahale.metrics.MetricRegistry;
4 import com.fasterxml.jackson.databind.ObjectMapper;
5 import com.zaxxer.hikari.HikariConfig;
6 import com.zaxxer.hikari.HikariDataSource;
7 import org.hibernate.cache.CacheManager;
8 import org.hibernate.cache.hazelcast.HazelcastCacheManager;
9 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory;
10 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.Autowired;
11 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
12 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
13 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
14 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
15 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
16 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
17 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
18 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
19 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
20 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
21 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
22 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
23 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
24 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
25 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
26 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
27 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
28 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
29 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
30 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
31 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
32 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
33 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
34 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
35 import org.hibernate.cache.hazelcast.HazelcastCacheManagerFactory.AutoWired;
```

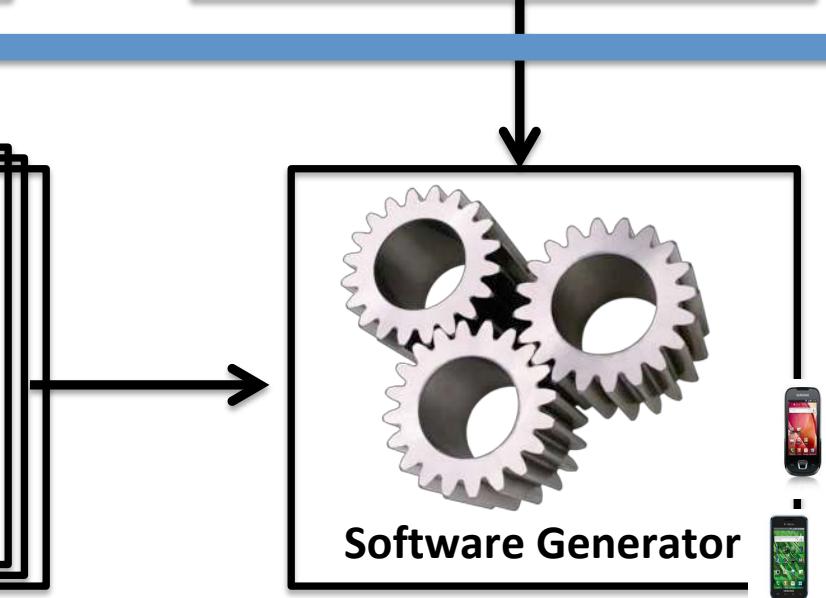
## Base Artefacts

## Software Generator (derivation engine)

# Domain Engineering



A screenshot of a software interface showing code in Notepad.java and an AST View panel. The code includes logic for "apply" and "intro" methods, involving variables like "t", "device", "advice", "layerno", and "selected". The AST View shows nodes for EXPRESSION, THEN\_STATEMENT, ELSE\_STATEMENT, and IFStatement.



# Application Engineering

« the investments required to develop the reusable artifacts during **domain engineering**, are outweighed by the benefits of deriving the individual products during **application engineering** »

# Software Product Line and Variability Engineering

A revisit of your cursus

# What is new?

**Family vs single systems**

**Focus on reuse**

**Domain engineering**

**Factoring out commonality**

**Managing variability**

« variability »

Is it really new?

# Parameter

```
Administrator: C:\Windows\system32\cmd.exe
C:\Users\kaestner.INFORMATIK.000>dir /?
Displays a list of files and subdirectories in a directory.

DIR [drive:][path][filename] [/A[[:l]attributes] [/B] [/C] [/D] [/L] [/N]
  [/O[[:l]sortorder]] [/P] [/Q] [/R] [/S] [/T[[:l]timefield]] [/W] [/X] [/4]

[drive:][path][filename]
      Specifies drive, directory, and/or files to list.

/A          Displays files with specified attributes.
attributes   D  Directories                  R  Read-only files
              H  Hidden files                A  Files ready for archiving
              S  System files                I  Not content indexed files
              L  Reparse Points             -  Prefix meaning not
/B          Uses bare format (no heading information or summary).
/C          Display the thousand separator in file sizes. This is the
            default. Use /-C to disable display of separator.
/D          Same as wide but files are list sorted by column.
/L          Uses lowercase.
/N          New long list format where filenames are on the far right.
/O          List by files in sorted order.
sortorder    N  By name (alphabetic)        S  By size (smallest first)
              E  By extension (alphabetic)  D  By date/time (oldest first)
              G  Group directories first   -  Prefix to reverse order
/P          Pauses after each screenful of information.
```

# Parameter -i in grep

```
1 int match_icase;
2
3 int main (int argc, char **argv)
4 {
5     [...]
6     while ((opt = get_nondigit_option (argc, argv, &default_color))
7         switch (opt)
8         {
9             [...]
10            case 'i':
11                match_icase = 1;
12                break;
13            }
14        }
15
16
17 static const char *
18 print_line_middle (const char *beg, const char *lim,
19                     const char *line_color, const char *match_color)
20 {
21     [...]
22     if (match_icase)
23     {
24         ibeg = buf = (char *) xmalloc(i);
25         while (--i >= 0)
26             buf[i] = tolower(beg[i]);
27     }
}
```

# Global configuration

```
class Config {  
    public static boolean isLogging = false;  
    public static boolean isWindows = false;  
    public static boolean isLinux = true;  
}  
class Main {  
    public void foo() {  
        if (isLogging)  
            log(„running foo()“);  
        if (isWindows)  
            callWindowsMethod();  
        else if (isLinux)  
            callLinuxMethod();  
        else  
            throw RuntimeException();  
    }  
}
```

# Configuration

## httpd.conf -- win32 Apache Building a Web Server, for Windows

```
Listen 80
ServerRoot "/www/Apache2"
DocumentRoot "/www/webroot"
```

```
ServerName localhost:80
ServerAdmin admin@localhost
```

```
ServerSignature On
ServerTokens Full
```

```
DefaultType text/plain
AddDefaultCharset ISO-8859-1
```

```
UseCanonicalName Off
```

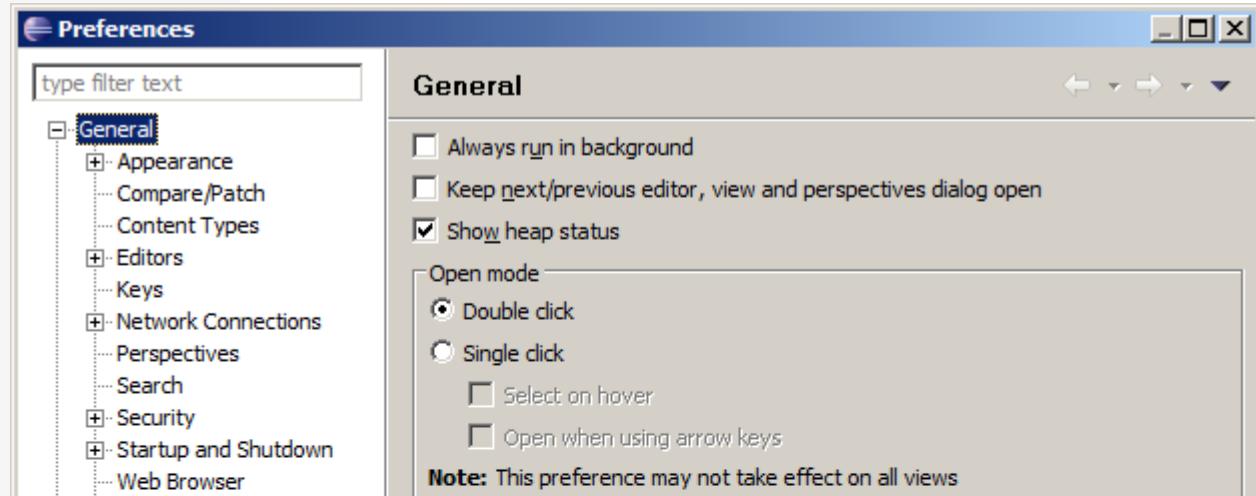
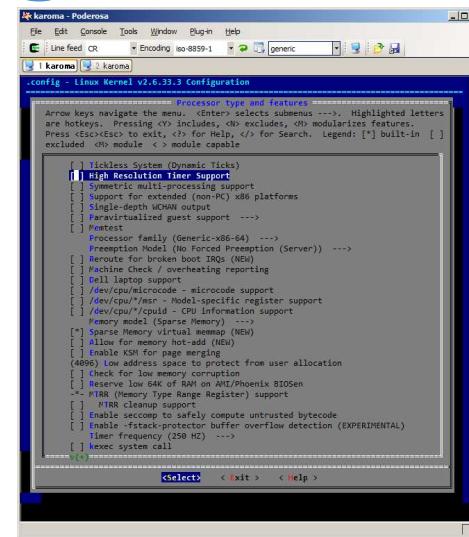
```
HostnameLookups Off
```

```
ErrorLog logs/error.log
LogLevel error
```

```
PidFile logs/httpd.pid
```

```
Timeout 300
```

```
KeepAlive On
MaxKeepAliveRequests 100
```



# Conditional compilation

## #ifdef (Berkeley DB)

```
static int __rep_queue_filedone(dbenv, rep, rfp)
    DB_ENV *dbenv;
    REP *rep;
    __rep_fileinfo_args *rfp; {
#ifndef HAVE_QUEUE
    COMPQUIET(rep, NULL);
    COMPQUIET(rfp, NULL);
    return (__db_no_queue_am(dbenv));
#else
    db_pgno_t first, last;
    u_int32_t flags;
    int empty, ret, t_ret;
#endif
#ifdef DIAGNOSTIC
    DB_MSGBUF mb;
#endif
    // over 100 lines of additional code
}
#endif
```

# Intentional Code Cloning

~ Copy & Paste

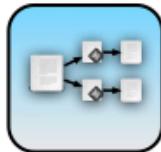
# Code Cloning (example, Linux driver)

cyberstorm.c

```
....  
static void dma_dump_state(struct NCR_ESP *esp)  
{  
    ESPLOG("esp%d: dma -- cond_reg<%02x>\n",  
           esp->esp_id, ((struct cyber_dma_registers *)  
                           (esp->dregs))->cond_reg);  
    ESPLOG("intreq:<%04x>, intena:<%04x>\n",  
           custom.intreq, custom.intenar));  
}  
  
static void dma_init_read(struct NCR_ESP *esp, __u32 addr, int  
length)  
{  
    struct cyber_dma_registers *dregs =  
        (struct cyber_dma_registers *) esp->dregs;  
  
    cache_clear(addr, length);  
  
    addr &= ~(1);  
    dregs->dma_addr0 = (addr >> 24) & 0xff;  
    dregs->dma_addr1 = (addr >> 16) & 0xff;  
    dregs->dma_addr2 = (addr >> 8) & 0xff;  
    dregs->dma_addr3 = (addr ) & 0xff;  
    ctrl_data &= ~(CYBER_DMA_WRITE);  
}.....
```

cyberstormll.c

```
....  
static void dma_dump_state(struct NCR_ESP *esp)  
{  
    ESPLOG("esp%d: dma -- cond_reg<%02x>\n",  
           esp->esp_id, ((struct cyberll_dma_registers *)  
                           (esp->dregs))->cond_reg));  
    ESPLOG("intreq:<%04x>, intena:<%04x>\n",  
           custom.intreq, custom.intenar));  
}  
  
static void dma_init_read(struct NCR_ESP *esp, __u32 addr, int  
length)  
{  
    struct cyberll_dma_registers *dregs =  
        (struct cyberll_dma_registers *) esp->dregs;  
  
    cache_clear(addr, length);  
  
    addr &= ~(1);  
    dregs->dma_addr0 = (addr >> 24) & 0xff;  
    dregs->dma_addr1 = (addr >> 16) & 0xff;  
    dregs->dma_addr2 = (addr >> 8) & 0xff;  
    dregs->dma_addr3 = (addr ) & 0xff;  
}  
.....
```

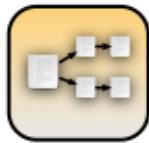


# Replicate & Specialize

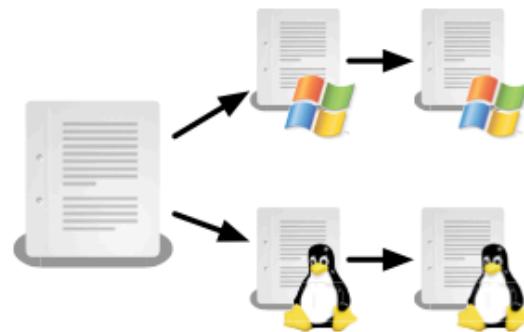


## Clone to reuse and adapt existing solutions

- + Less effort needed
- Long-term cost outweighs short-term benefit
- ~ Cost of refactoring rises over time

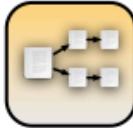


# Platform Variations

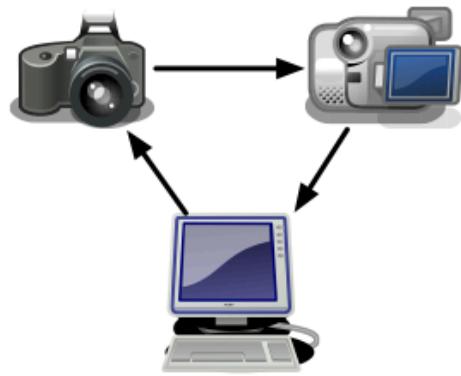


**Clone existing code and fix  
low level platform interaction**

- + Avoid complexity of virtualization layer
- Hard to propagate bug fixes
- ~ Ensure consistent behavior of all clones



# Hardware Variations



## Clone existing driver

- + No risk of changing existing driver
- Code growth
- ~ Dead code can creep into system

# Inheritance (OOP)

Base Class encapsulate commonalities

Derive classes specialize peculiarities

# Generic Programming

## C++ template

```
template <typename T>
T max(T x, T y)
{
    return x < y ? y : x;
}
```

## Generics in Java

```
public interface List<E> {
    void add(E x);
    Iterator<E> iterator();
}
public interface Iterator<E> {
    E next();
    boolean hasNext();
}
```

# Design Patterns

Template Method

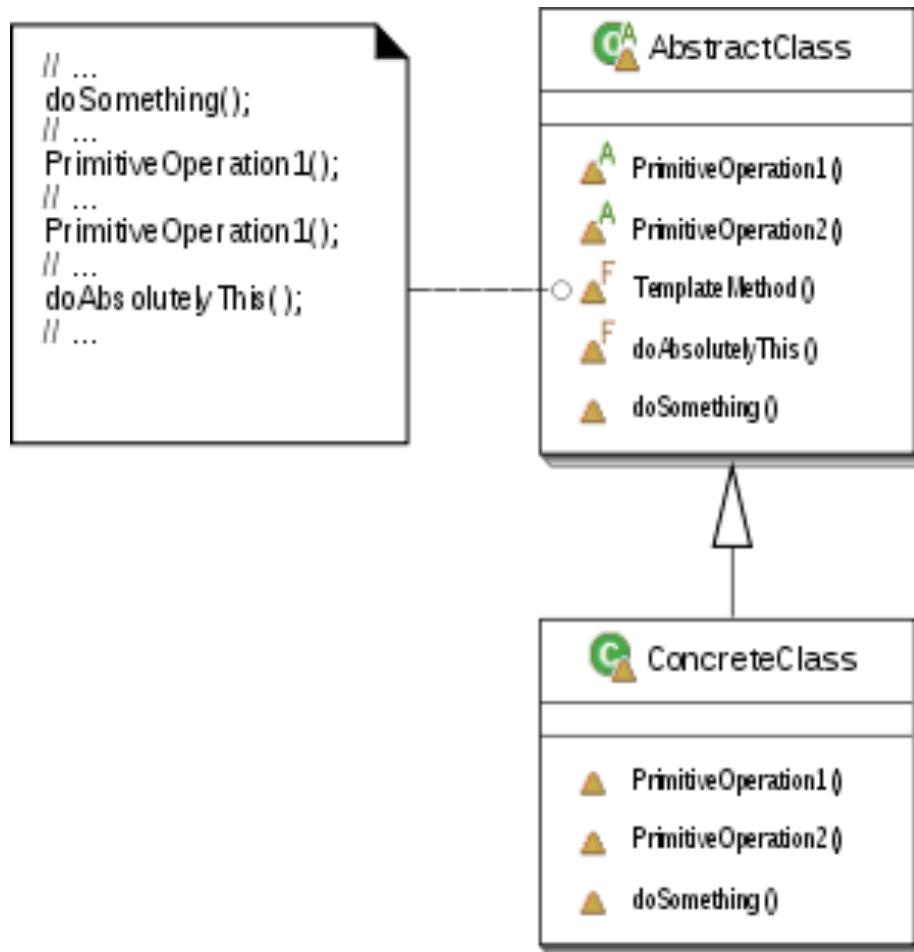
Factory

Strategy

Decorator

....

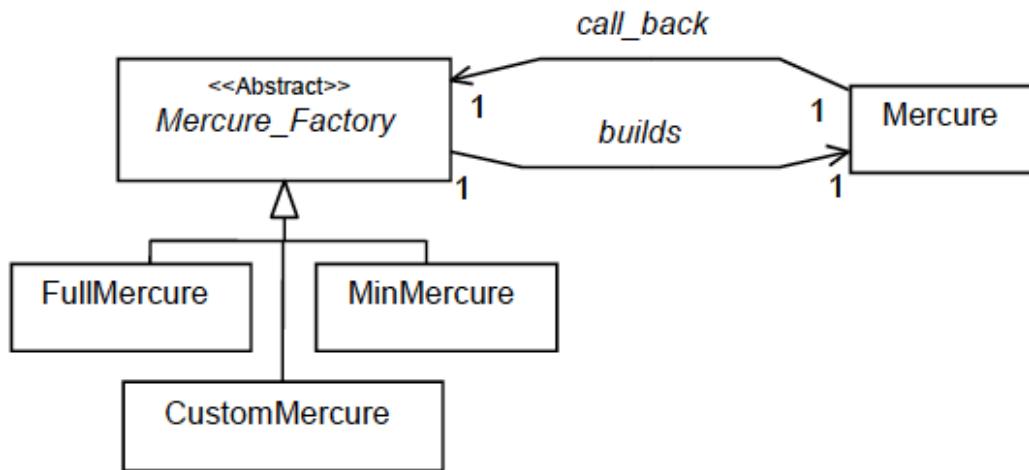
# Template Method



# The decision model

## ■ The Abstract Factory Design Pattern – [Gamma et al 95]

Mercure_Factory
new_gui() : GUI
new_language() : Language
new_network_manager() : Manager
new_netdriver() : Net Driver
new_engine() : Engine



CustomMercure
<<GUI1>> <<GUI2>> new_gui() : GUI
<<<Language2-1>> new_language() : Language
<<Manager1>> new_network_manager() : Manager
<<NetDriver1>> <<NetDriver2>> new_netdriver() : Net Driver
<<Engine1>> new_engine() : Engine

# API Framework

# Plugin-based systems

# (Active) Annotations

can have parameters

# Metamodeling and Domain-Specific Languages

# httpd.conf -- win32 Apache

## Building a Web Server, for Windows

```
Listen 80
ServerRoot "/www/Apache2"
DocumentRoot "/www/webroot"

ServerName localhost:80
ServerAdmin admin@localhost

ServerSignature On
ServerTokens Full

DefaultType text/plain
AddDefaultCharset ISO-8859-1

UseCanonicalName Off

HostnameLookups Off

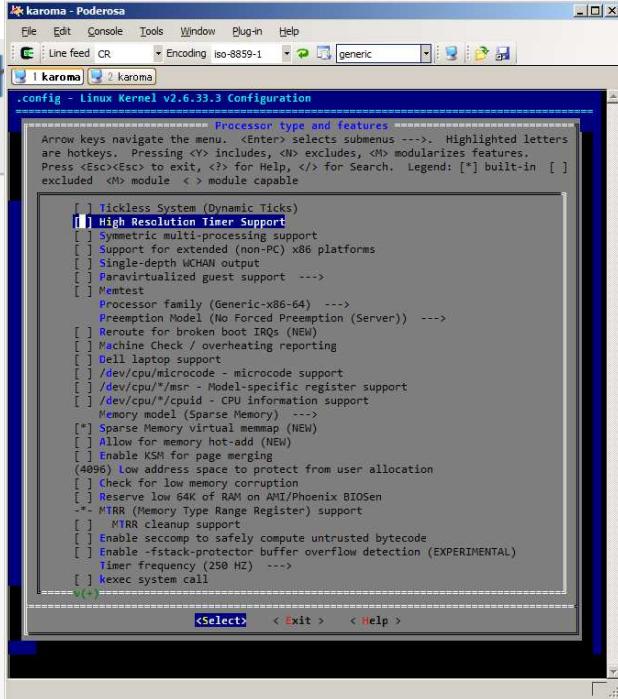
ErrorLog logs/error.log
LogLevel error

PidFile logs/httpd.pid

Timeout 300

KeepAlive On
MaxKeepAliveRequests 100
KeepAliveTimeout 15

<IfModule mpm_winnt.c>
    ThreadsPerChild 250
    MaxRequestsPerChild 0
</IfModule>
```



RENAULT VANS

CARS | VANS | ELECTRIC VEHICLES | RENAULT BUSINESS | USED CARS | OWNER SERVICES | ABOUT RENAULT | RENAULT SHOP

Renault UK > Renault Vans > New Kangoo Van Range > Kangoo Van > Build your own Kangoo Van > Selected Options

NEW KANGOO VAN RANGE

01 Preferences    02 Version    03 Equipment & options

< Previous    > Next

**OPTIONS**

> COMFORT

Central storage console & armrest between seats £50.00

> DRIVING

Electric door mirrors £0.00

> SAFETY & SECURITY

ESC (Electronic Stability Control) with traction and understeer control £200.00

Preferences

type filter text

**General**

Always run in background

Keep next/previous editor, view and perspectives dialog open

Show heap status

Open mode

Double click

Single click

Select on hotkey

Open when

**Note:** This preference is shared with the Java perspective.

Notepad.java    Actions.java    Main.java

```
output.setText(t.toString());
program.setText(t.eval(""));
equation.setText(base);
updateQuarkPanel();
}

add.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        if (hoa.isSelected()) {
            t = t.apply(new hoa("h" + layerno));
        }
        if (ladvice.isSelected()) {
            t = t.apply(new advice("a" + layerno));
        }
        if (intro.isSelected()) {
            t = t.apply(new intro("i" + layerno));
        }
        if (gadvice.isSelected()) {
            t = t.apply(new gadvice("g" + layerno));
        }
    }
});

if (hoa.isSelected() || gadvice.isSelected() || ladvice.isSelected() || intro.isSelected()) {
    hoa.setSelected(false);
    gadvice.setSelected(false);
    ladvice.setSelected(false);
    intro.setSelected(false);
    equation.setText("F" + layerno + "(" + equation.g + ")");
}
```

Outline    ASTView

OC: null  
IERS: 1  
RUCTOR: false  
PARAMETERS: 0  
N\_TYPE2  
ITERES: 1  
DIMENSIONS: 0  
VN\_EXCEPTIONS: 0  
node[4443, 805]  
STATEMENTS: 5  
EStatement[4450, 191]  
+ EXPRESSION  
+ MethodInvocation  
+ THEN\_STATEMENT  
+ ELSE\_STATEMENT: null  
+ IfStatement[4529, 80]  
+ IfStatement[4615, 77]  
+ IfStatement[4785, 457]

A photograph of a heavily rusted, vintage-style pickup truck. The truck is positioned diagonally, facing towards the top left. It is situated in a field of tall, dry grass and weeds. The body of the truck is a light green color, which has faded significantly due to rust. The front grille and headlight area are missing, and the side door is open, revealing the interior frame and some debris. The bed of the truck is visible, showing signs of decay. The overall condition of the vehicle is poor, suggesting it has been abandoned for a long time.

Unused flexibility



# Illegal variant

# Feature Models

A screenshot of the Karmic configuration tool interface. It displays a list of kernel configuration options under the heading "Processor Type and Features". The options include "High Resolution Timer Support", "Symmetric multi-processing support", "Single-depth MMU output", "Paravirtualized guest support", "Processor family (Generic<-x86-64)", "Preemption Model (No Forced Preemption (Server))", "Reserve first boot IRQ (NEW)", "Machine Check Exception reporting", "Bell laptop support", "Microcode support", "/dev/cpu/microcode", "Model-specific register support", "/dev/cpu/\*cpuuid - CPU Information support", "Memory model (Sparse Memory)", "Allow memory hot-add (NEW)", "Enable KSM for page merging", "Report low memory corruption to user application", "Check for low memory corruption", "Reserve low 64K of RAM on AMI/Phoenix BIOSen", "Support for TBR (Type Range Register) support", "Enable seccomp to safely compute untrusted bytecode", "Enable fstack-protector buffer overflow detection (EXPERIMENTAL)", "File fragmentation (256 KB)", and "[ ] kexec system call". At the bottom, there are buttons for "<Select> < Exit > < Help >".

Variability Model

A screenshot of an IDE showing a Java code editor with several annotations overlaid. The code is part of a class named Notepad.java and includes imports for Actions.java and Main.java. Annotations are color-coded: green for 'OCT: null', yellow for 'IERS(1)', orange for 'RUCTOR: 'False'', blue for 'PARAMETERS(0)', red for 'N\_TYPE2', purple for 'LETERS(1)', light blue for '\_DIMENSIONS: 0', and grey for 'VN\_EXCEPTIONS(0)'. A large black arrow points from the Variability Model section to this IDE screen.

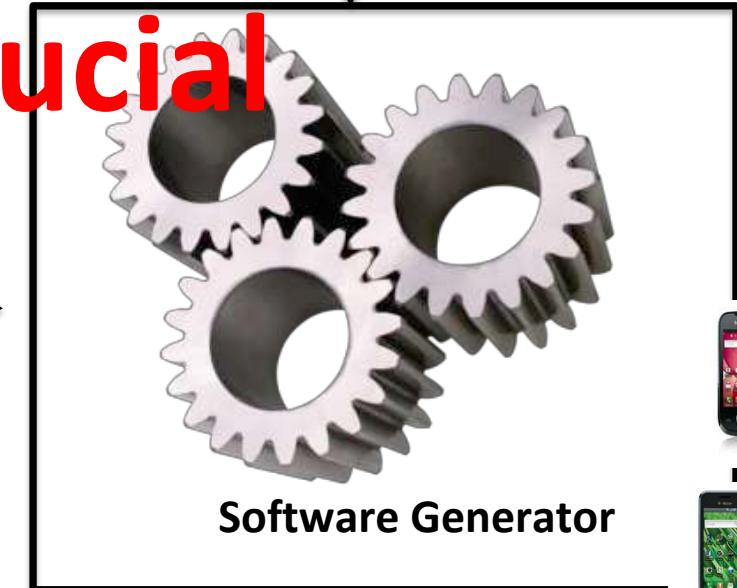
Modeling variability in main artifacts (e.g., source code)



Configuration



is crucial



Software Generator

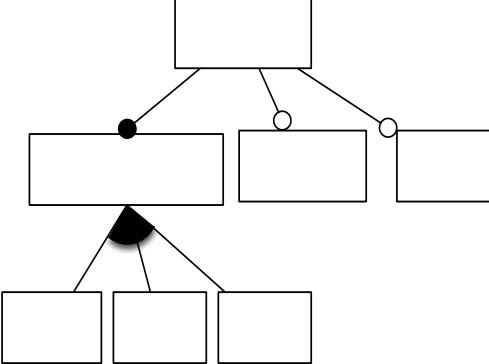
# Unused flexibility





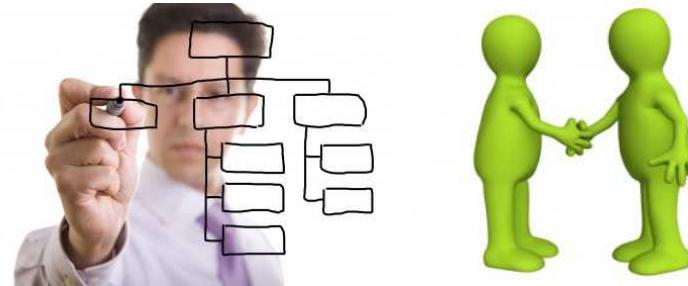
# Illegal variant

# Feature Model



not, and, or, implies

## Communicative



## Analytic



## Generative





## R8 Spyder 5.2 FSI quattro R-tronic

### Prix total

**171.216,00 EUR**

### Prix de base

**170.490,00 EUR**

### Equipements optionnels

**726,00 EUR**

- ▶ Informations détaillées
- ▶ Entrez l'Audi Code ↗
- ▶ Générer un PDF
- ▶ Nouvelle configuration

[+] Plein écran / Dimensions

▶ Fermer la capote

Habitacle

Tableau de bord

### Packs

Aucun pack n'est proposé pour ce modèle.

### Couleurs

Blanc Ibis

Noir

Prix: 0,00 EUR



Couleurs métallisées à partir de 0,00 EUR



Couleurs à effet perlé à partir de 0,00 EUR



Couleurs personnalisées Audi exclusive



### Couleur capote

Noir



### Jantes

4 Jantes alu 5 BRANCHES ROTOR finition titane 8,5 x 19 à l'avant, 11 x 19 à l'arrière. Pneus 235/35 R19 à l'avant et 305 /30 R19 à l'arrière

Prix: 726,00 EUR

19" à partir de 0,00 EUR





## R8 Spyder

5.2 FSI quattro R tronic

Prix total

185.899,35 EUR

Prix de base

170.490,00 EUR

Equipements optionnels

15.409,35 EUR

▶ Informations détaillées

▶ Entrez l'Audi Code

▶ Générer un PDF

▶ Nouvelle configuration

[+ Plein écran / Dimensions

[+ Vue extérieure

[+ Tableau de bord

▶ Packs d'équipements

▶ Extérieur

▶ Jantes & pneumatiques

▶ Intérieur

▶ Volants

▶ Sièges

### Sécurité & technique

▶ Infotainment

▶ Châssis

▶ Freins

### Systèmes d'assistance

▶ Autres

Régulateur de vitesse

320,65 EUR

Système d'aide au stationnement APS avant / arrière

931,70 EUR



Système d'aide au stationnement APS avant / arrière avec affichage dans l'écran MMI

1.373,35 EUR

Système d'aide au stationnement Advanced : APS avant et arrière et caméra arrière

1.790,80 EUR



Audi hill assist : assistance au démarrage en côte

Série

Réinitialiser la sélection

excludes



#### Attention:

Le prix peut varier en fonction du choix de moteur et des équipements.

#### Un aperç des équipements:

Mode expert



## A5 Sportback 3.0 TDI quattro S tronic

### Prix total

**54.450,15 EUR**

Prix de base

**50.570,00 EUR**

Equipements optionnels

**3.890,15 EUR**

▶ Informations détaillées

▶ Entrez l'Audi Code

▶ Nouvelle configuration

### Vérification de votre sélection

#### Cet équipement nécessite un équipement complémentaire:

- GPS Plus avec disque dur 2.934,25 EUR

#### Voici les équipements complémentaires possibles:

- Ordinateur de bord en couleur avec programme efficiency 181,50 EUR  
Remarque: uniquement sur les modèles avec système Start-Stop et uniquement disponible en combinaison avec l'autoradio Concert, l'autoradio Symphony ou un système de navigation
- Pack Intenso Plus 3.100,00 EUR  
Sans appareil de navigation

[+] Plein écran / Dimensions

- ▶ Extérieur
- ▶ Jantes & pneumatiques
- ▶ Intérieur
- ▶ Volants
- ▶ Sièges
- ▶ Sécurité & technique
- Infotainment**

#### Attention:

Le prix peut varier en fonction du choix de moteur et des équipements.

#### Un aperç des équipements:

Mode expert

Réinitialiser la sélection

**1 Modèle**

**2 Moteur**

**3 Extérieur**

**4 Intérieur**

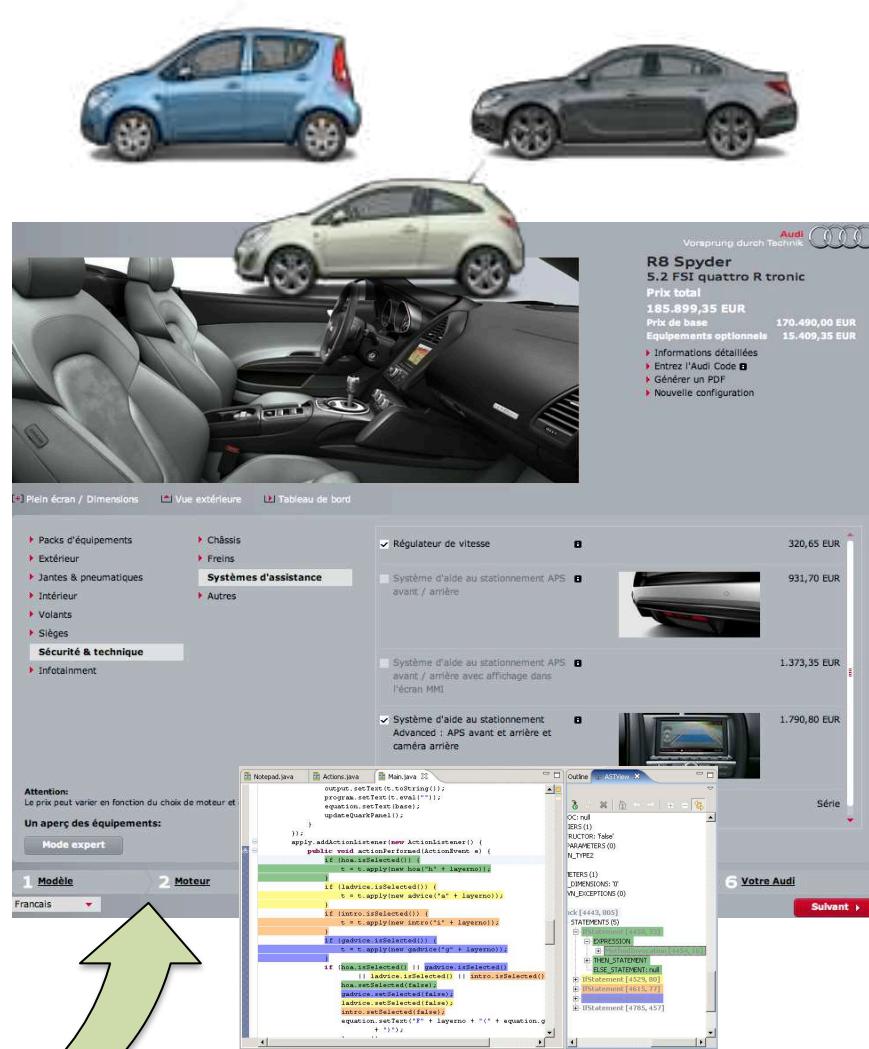
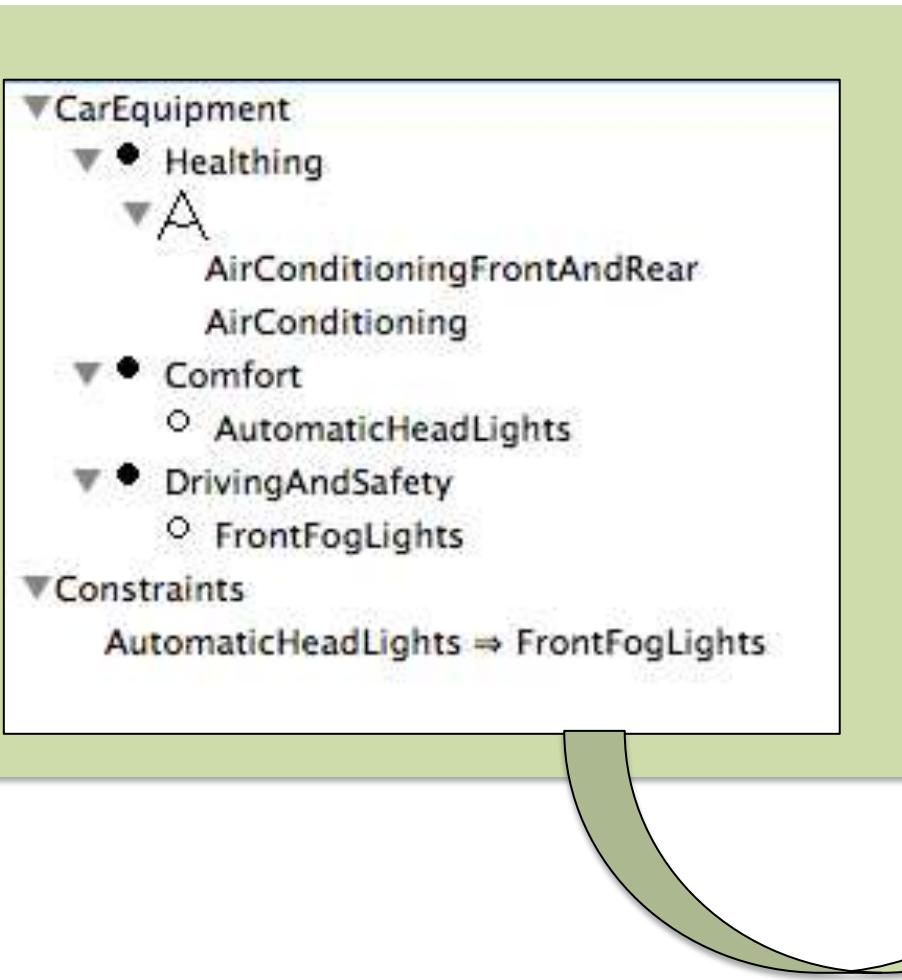
**5 Option**

**6 Votre Audi**

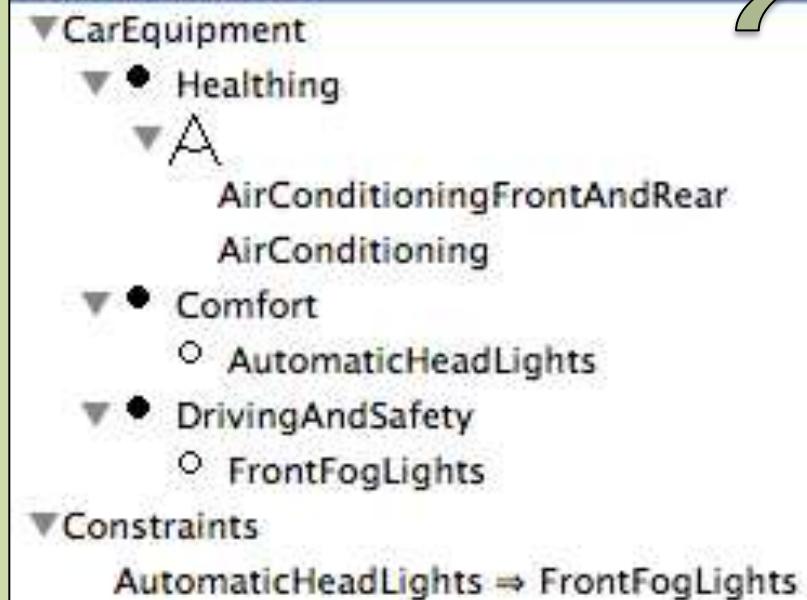
Français

Suivant ▶

# Feature Models



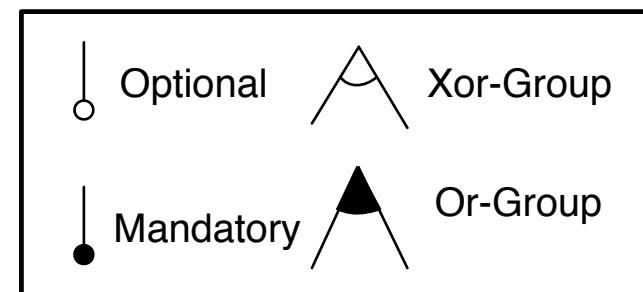
# Feature Models (Background)

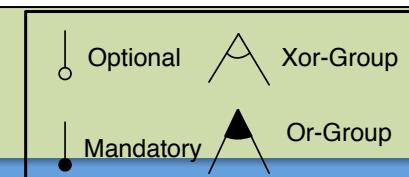
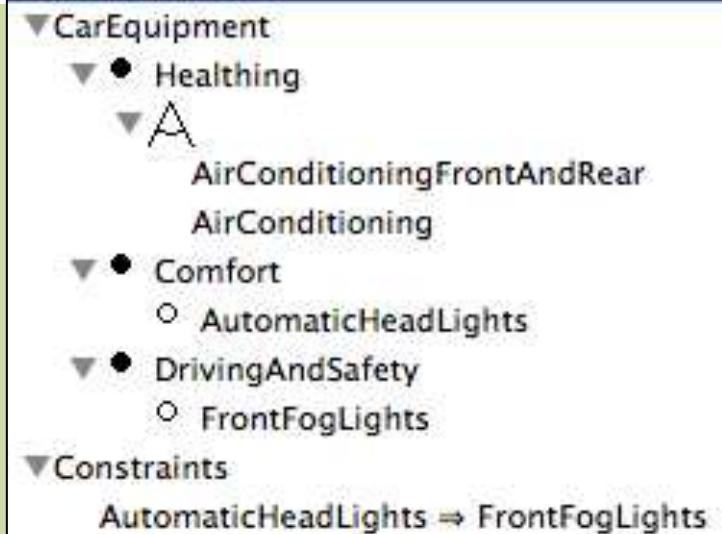
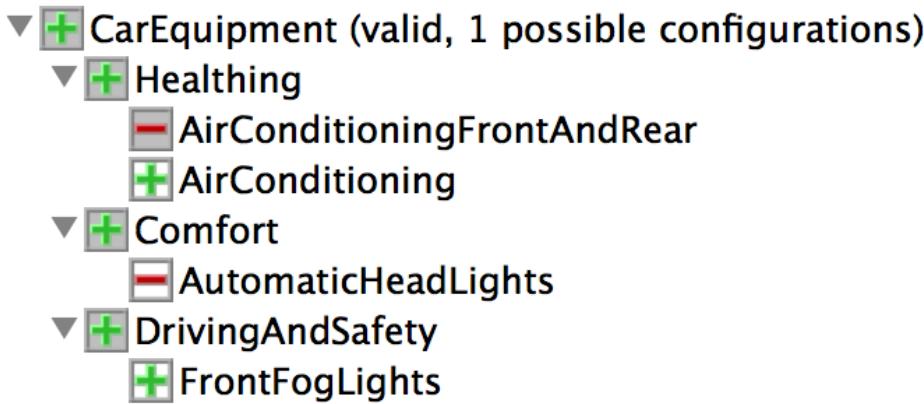


**Hierarchy:** rooted tree

**Variability:**

- mandatory,
- optional,
- Groups: exclusive or inclusive features
- Cross-tree constraints





# Hierarchy + Variability

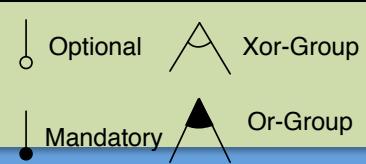
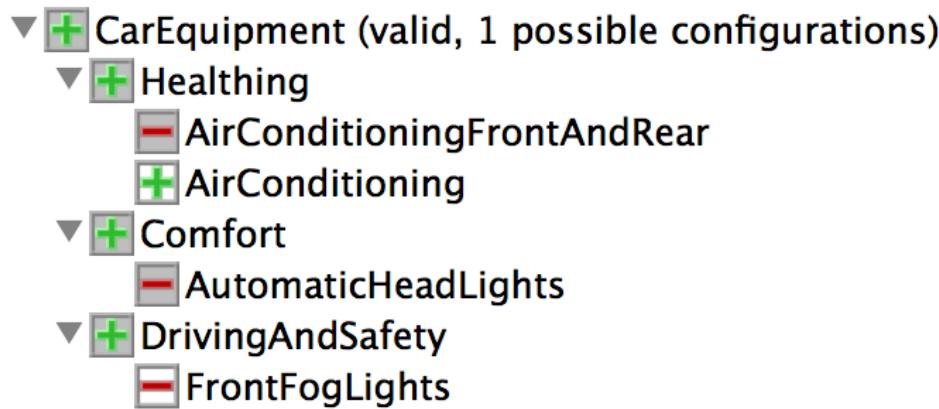
=

## set of valid configurations

**configuration = set of features selected**

{CarEquipment, Comfort, DrivingAndSafety, Healthing, AirConditioning, FrontFogLights}





# Hierarchy + Variability

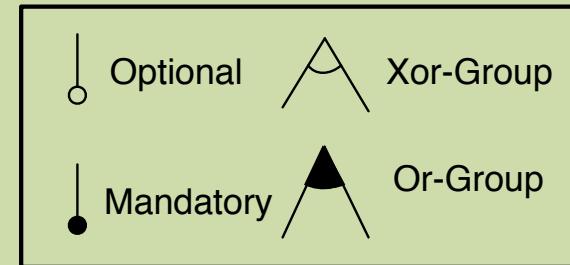
=

## set of valid configurations

**configuration = set of features selected**

{CarEquipment, Comfort, DrivingAndSafety, Healthing, AirConditioning}





## Hierarchy + Variability

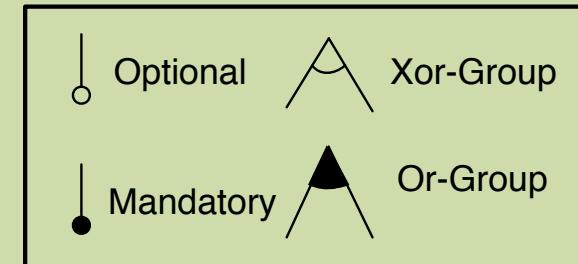
=

## set of valid configurations

**configuration = set of features selected**

{CarEquipment, Comfort, DrivingAndSafety, Healthing, AirConditioning, AutomaticHeadLights}





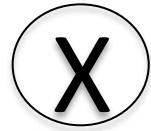
## Hierarchy + Variability

=

## set of valid configurations



{CarEquipment, Comfort,  
DrivingAndSafety,  
Healthing}



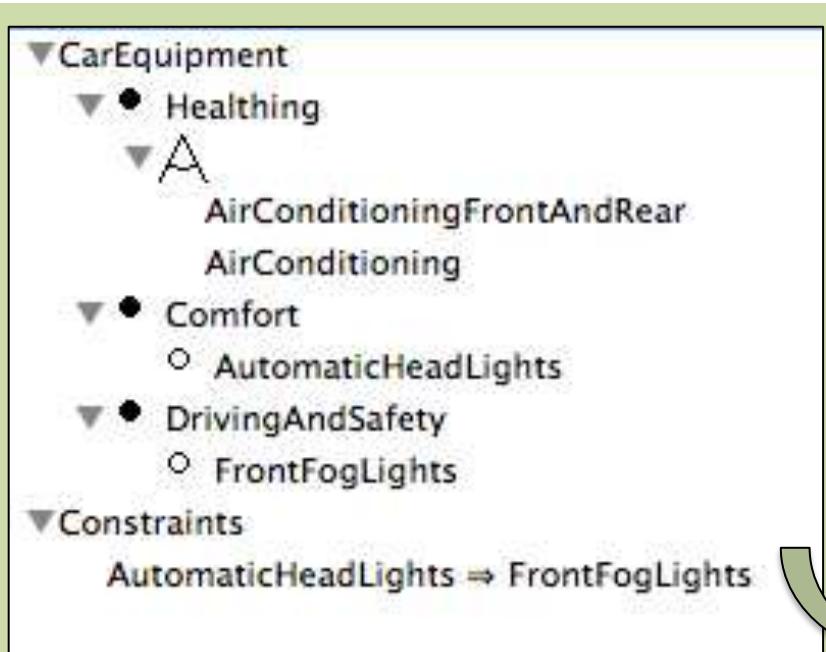
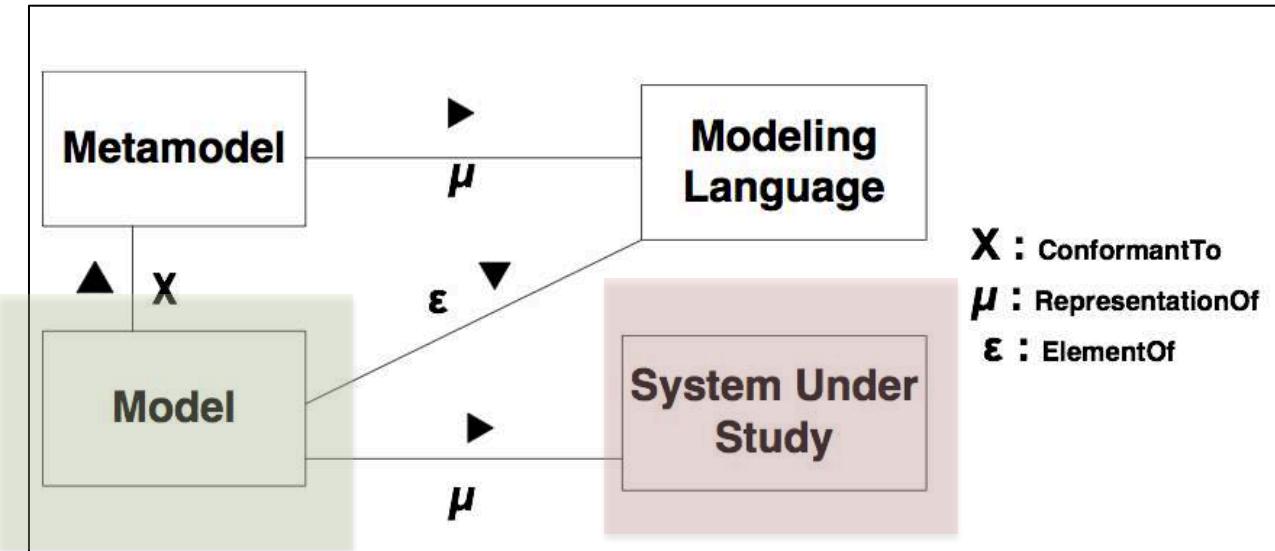
- {AirConditioning, FrontFogLights}
- {AutomaticHeadLights, AirConditioning, FrontFogLights}
- {AutomaticHeadLights, FrontFogLights, AirConditioningFrontAndRear}
- {AirConditioningFrontAndRear}
- {AirConditioning}
- {AirConditioningFrontAndRear, FrontFogLights}

<http://familiar.variability.io/>

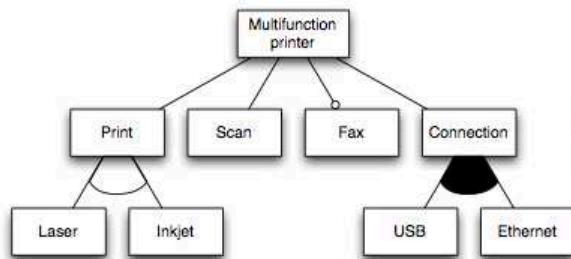
<http://familiar.variability.io/ide/familiar>



# Feature Models



# Typical implementations



result

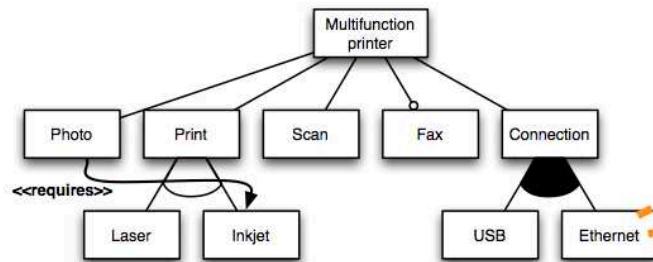


Z3

# Product Derivation

feature model

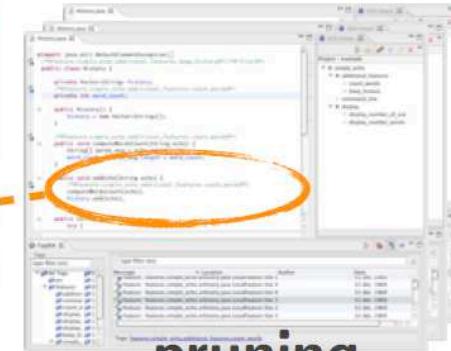
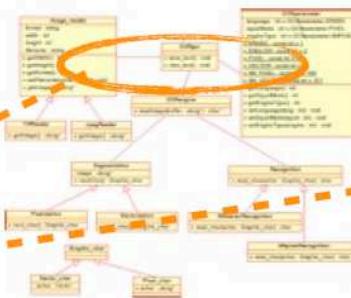
variable model and code assets



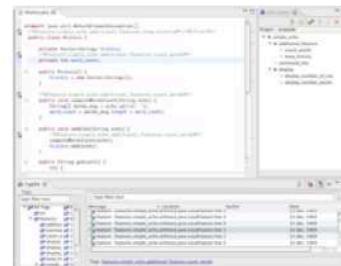
configuration

{ MP, Photo, Print, Inkjet, Scan,  
Fax, Connection, USB, Ethernet }

product spec



pruning,  
composition,  
weaving,  
transformation



product

# Exercise

- JHipster (<http://www.jhipster.tech/>), let us consider the last version (November, 2017)
  1. List all options of Jhipster
  2. Propose some options that Jhipster could propose/support
  3. Describe some combinations of options of Jhipster that are not possible; justify why!

Look at homepage, documentation, and... source code!

Bref  
as a product line

## ETAPE 2 : CHOISIS 3 BONS SOUVENIRS



## ETAPE 3 : JE REGARDE MON EPISODE UNIQUE

DEJÀ 1 796 108 ÉPISODES GENÉRÉS.

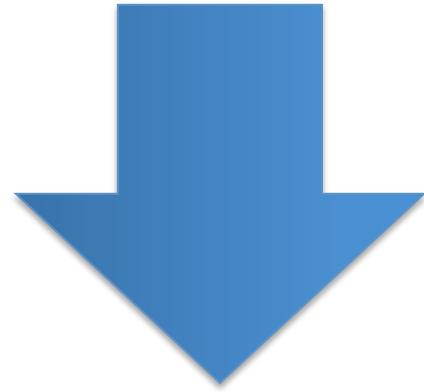




```
foo1.videogen ✘

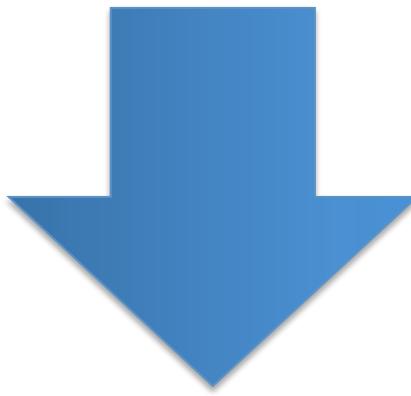
mandatory videoseq v1 "https://www.youtube.com/watch?v=PjNi1uYhV5w"
optional videoseq v2 "v2folder/v2.mp4"
alternatives v3 {
    videoseq v31 "v3/seq1.mp4"
    videoseq v32 "v3/seq1.mp4"
    videoseq v33 "v3/seq1.mp4"
}

alternatives v4 {
    videoseq v41 "v4/seq1.mp4"
    videoseq v42 "v4/seq1.mp4"
}
mandatory videoseq v5 "https://www.youtube.com/watch?v=ezKx-S0LiNQ"
```



 FFmpeg

```
VideoGen {  
    mandatory videoseq v1 "V1/v1.mp4"  
    optional videoseq v2 "v2folder/v2.mp4"  
    alternatives v3 {  
        videoseq v31 "v31.mp4"  
        videoseq v32 "v32.mp4"  
    }  
}
```



model-to-text

```
fmVideoGen = FM (VideoGen: v1  
[v2] v3; v3: (v31|v32); )
```

FAMILIAR

# Product line and variability everywhere

- Your video generator is a product line
  - Generalization of Bref generator
  - It is even a configurable generator
  - There is an hidden feature model in every VideoGen specification
- Xtext is a configurable generator (see MWE2)
- JHipster is a product line
- ffmpeg is a product line

# How MDE (IDM) can help

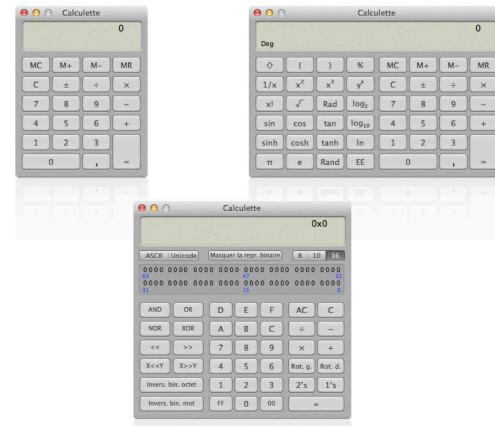
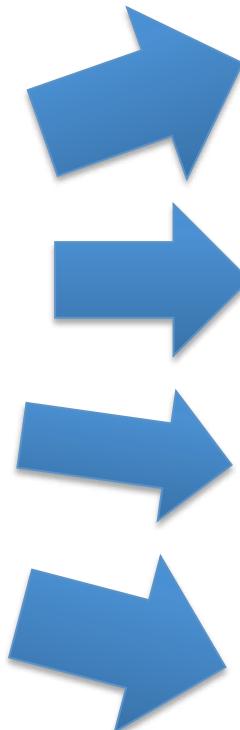
## Software Product Line Engineering

# Generative approach

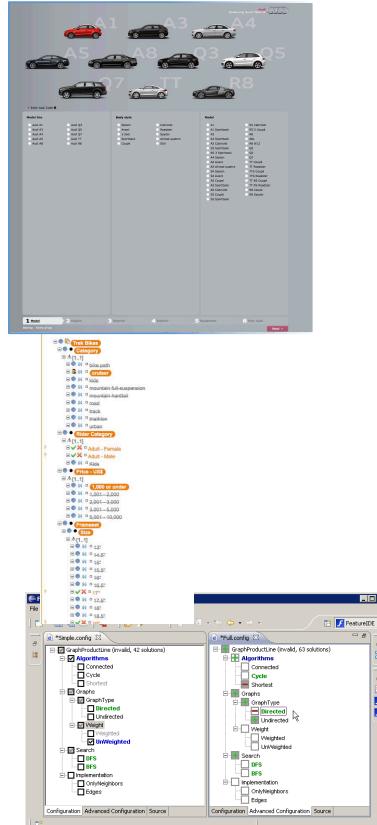
- Programming the generation of programs
  - Very old practice
  - Metaprogramming: generative language and target language are the same
    - Reflection capabilities
- Generalization of this idea:
  - from a specification written in one or more textual or graphical domain-specific languages
  - you generate **customized variants**

Modeling and implementing system families such that a desired system can be automatically generated from a specification written in one or more textual or graphical domain-specific languages.

Models  
And  
Languages



# Models And Languages



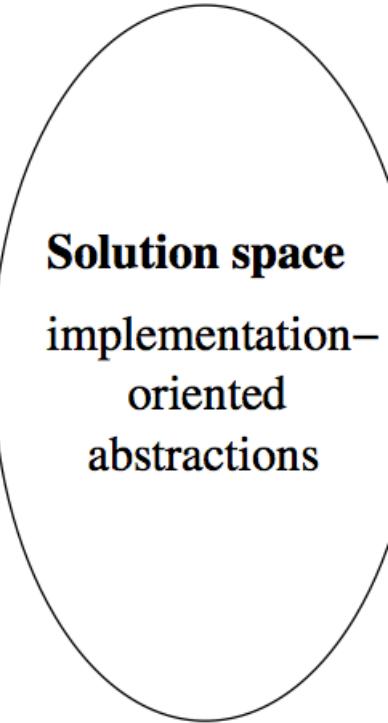
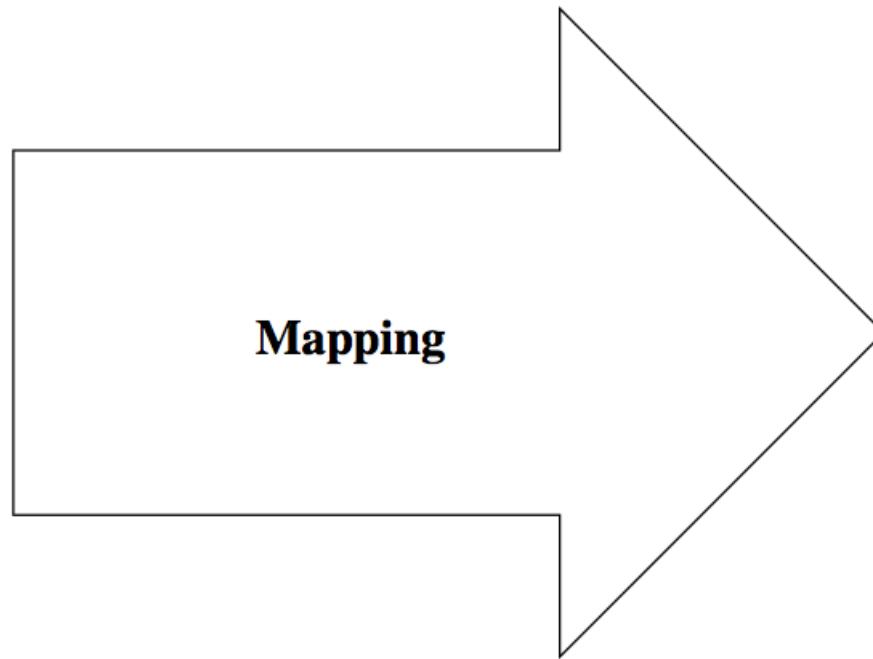
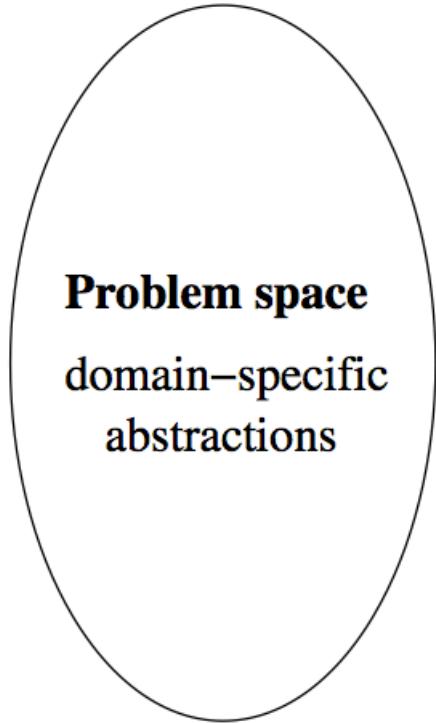
1. Trim/Series	2. Engine/Transmission	3. Colour & Style	4. Options	5. Summary
<input checked="" type="radio"/> SC				from € 14,950,-
<input type="radio"/> SRI				from € 21,195,00
<input type="radio"/> SE				from € 22,495,-
<input type="radio"/> Elite				from € 23,495,-
				from € 25,495,-

**Standard Equipment Filter** Heating/Ventilation

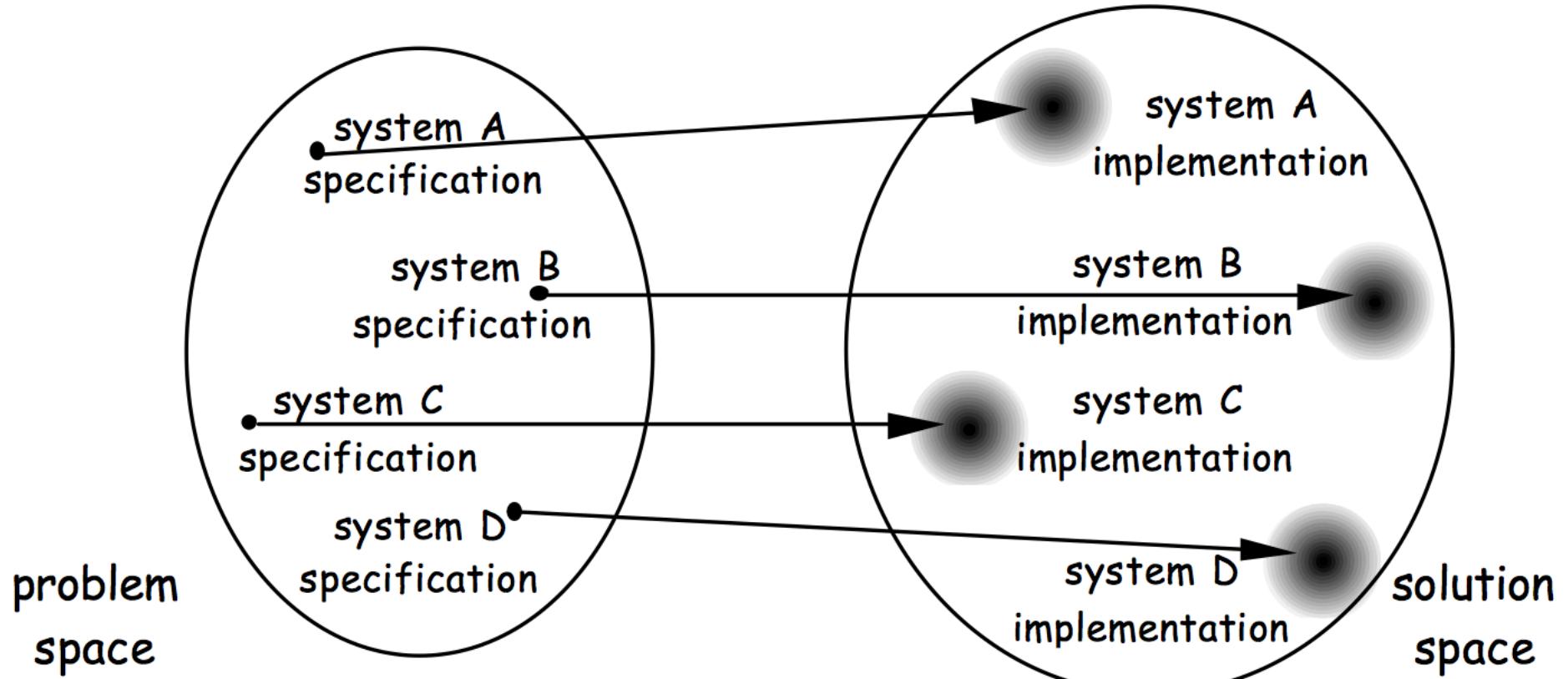
**Heating/Ventilation**

- Air conditioning with particle filter, manual controls

1. Trim/Spec	2. Engine/Transmission	3. Colour / Style	4. Options	5. Summary
Interior Confort/Convenience	Options packs	Safety/Security	Seating	Audi/Comms/Nav
Heating/Ventilation	Mechanical	A-Z		
<b>Safety/Security</b>				
<input checked="" type="checkbox"/> Emergency tire inflation kit				Standard
<input checked="" type="checkbox"/> Active front seat head restraints				<b>€ 103.00</b>
				- also adds side air ventilation serving light for front passenger's seat.
<input type="checkbox"/> Front and rear parking distance sensors				<b>€ 407.00</b>
<input checked="" type="checkbox"/> Tyre Pressure Monitoring System				<b>€ 155.00</b>
<input type="checkbox"/> Remote control ultrasonic security alarm system				<b>€ 357.00</b>



[Czarnecki and Eisenecker 2000]



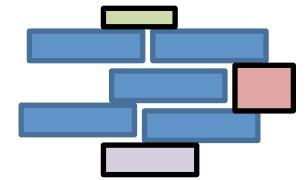
[Czarnecki, PhD thesis]

# Developing Product Lines

## Metamodels, DSLs, and Transformations to the rescue

- Domain Engineering
  - Domain Models
  - Level of abstraction
  - Domain-specific modeling languages
    - (visual or textual) syntax, precise semantics
    - analyzed (verification)
  - Traceability between the artefacts
- Application Engineering
  - Model transformations (automation)
- Reduce the gap

# Summary

- **Software product line engineering**
    - Mass customization
    - Family of software intensive systems
    - Systematic reuse
    - Domain engineering
    - Variability management
  - **Variability** everywhere
    - Applied and applicable to many industries and domains
  - **Modeling and implementing** variability: an overview
- 
- 

# Other references

- Krzysztof Czarnecki and Ulrich Eisenecker "Generative Programming: Methods, Tools, and Applications"
- S. Apel, D. Batory, C. Kästner, and G. Saake. Feature-Oriented Software Product Lines: Concepts and Implementation. Berlin/Heidelberg: Springer-Verlag, 2013.
- Cory Kapser, Michael W. Godfrey: "Cloning considered harmful" considered harmful: patterns of cloning in software. Empirical Software Engineering 13(6): 645-692 (2008)
- C. Kästner. Virtual Separation of Concerns: Toward Preprocessors 2.0. PhD thesis, 2010
- Klaus Pohl, Günter Böckle, Frank van der Linden: Software Product Line Engineering - Foundations, Principles, and Techniques. Springer 2005

# Other references

- Krzysztof Czarnecki, Krzysztof Pietroszek: Verifying feature-based model templates against well-formedness OCL constraints. GPCE 2006: 211-220
- José A. Galindo, Mauricio Alferez, Mathieu Acher, Benoit Baudry, and David Benavides. A Variability-based Testing Approach for Synthesizing Video Sequences (2014). In ISSTA'14
- Sarkar, A., J. Guo, N. Siegmund, S. Apel, and K. Czarnecki, "Cost-Efficient Sampling for Performance Prediction of Configurable Systems" In ASE'2015
- Mathieu Acher, Guillaume Bécan, Benoit Combemale, Benoit Baudry, and Jean-Marc Jézéquel. Product lines can jeopardize their trade secrets (2015). In ESEC/FSE'15