Ligne de Produits Logiciels

Vue générale, principes

Mathieu Acher
Maître de Conférences
mathieu.acher@irisa.fr

Material

http://mathieuacher.com/teaching/MDI/

Plan

- Challenges and Overview
  - Developping 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

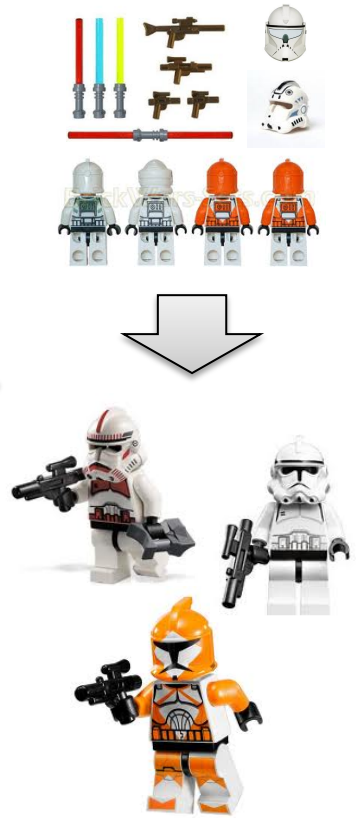
Challenges and Overview

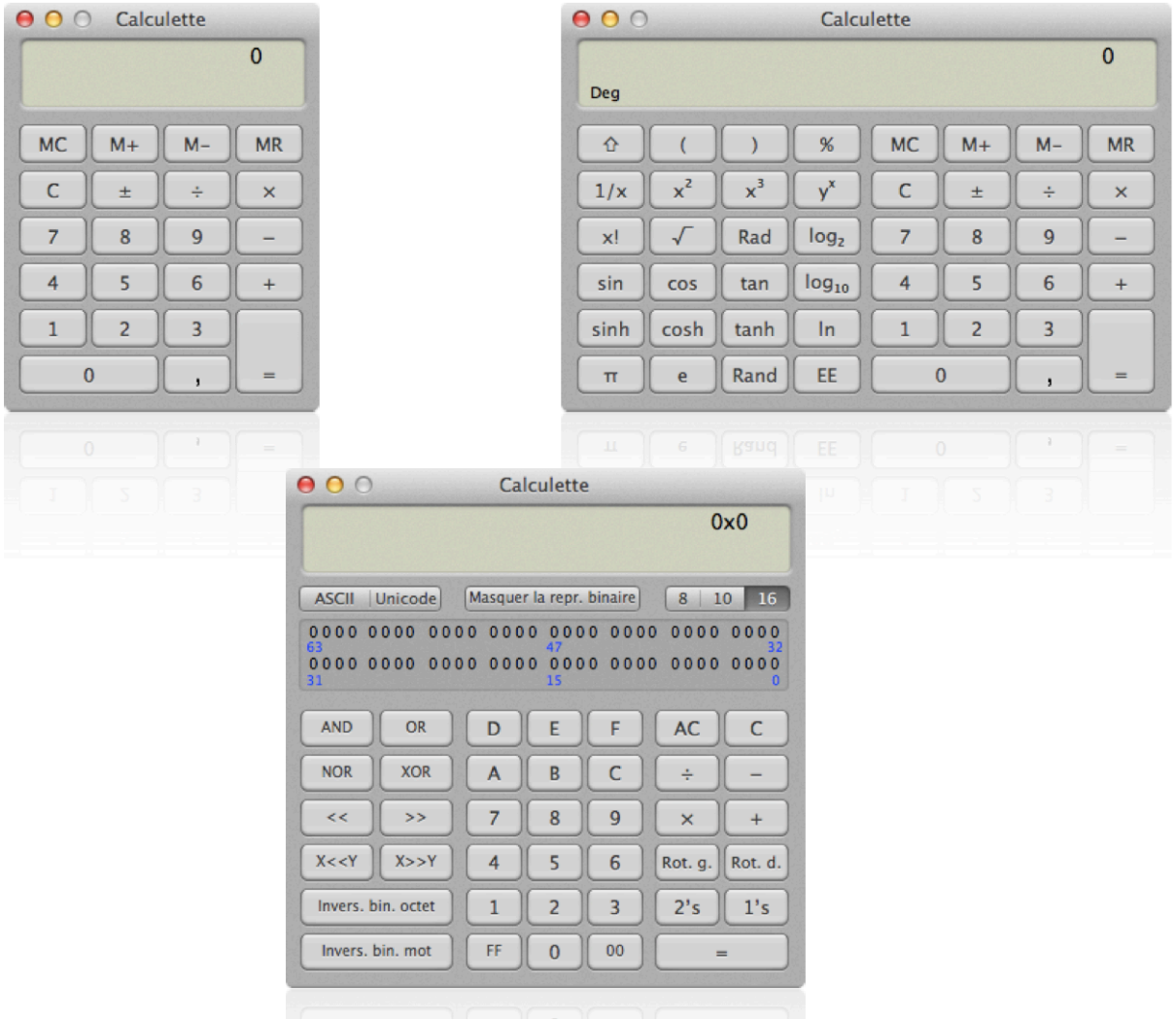
Trois manières de construire un produit (logiciel)

Indépendamment | „Clone & Own“ | **Assets „partagés“**

Lignes de produits logiciels

Configuration de produits
Modélisation de la variabilité
Composants
Langages dédiés
Générateurs
Préprocesseurs
Patrons de conceptions
…




Calculette
0
MC M+ M− MR
C ± ÷ ×
7 8 9 −
4 5 6 +
1 2 3 =
0 ,
Calculette
0
Deg
⇧ ( ) % MC M+ M− MR
1/x x² x³ yˣ C ± ÷ ×
x! √ Rad log₂ 7 8 9 −
sin cos tan log₁₀ 4 5 6 +
sinh cosh tanh ln 1 2 3 =
π e Rand EE 0 ,
Calculette
0x0
ASCII Unicode
Masquer la repr. binaire
8 10 16
0000 0000 0000 0000 0000 0000 0000 0000
63 47 32
0000 0000 0000 0000 0000 0000 0000 0000
31 15 0
AND OR D E F AC C
NOR XOR A B C ÷ −
<< >> 7 8 9 × +
X<<Y X>>Y 4 5 6 Rot. g. Rot. d.
Invers. bin. octet 1 2 3 2's 1's
Invers. bin. mot FF 0 00 =

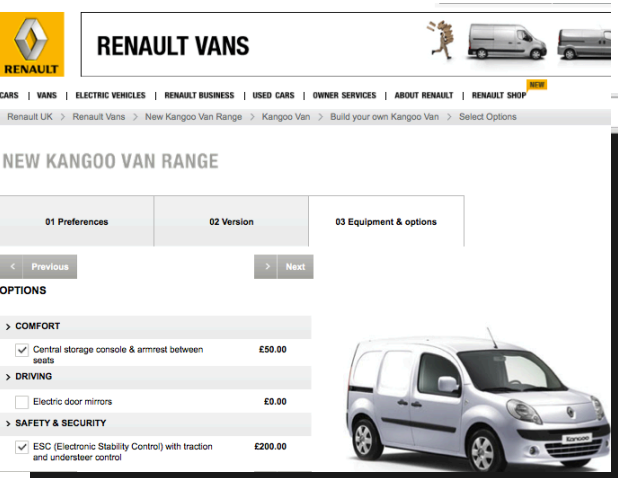
« A set of programs is considered to constitute a **family**, whenever it is worthwhile to study programs from the set by **first studying the common properties** of the set and then determining the **special properties** of the individual family members »

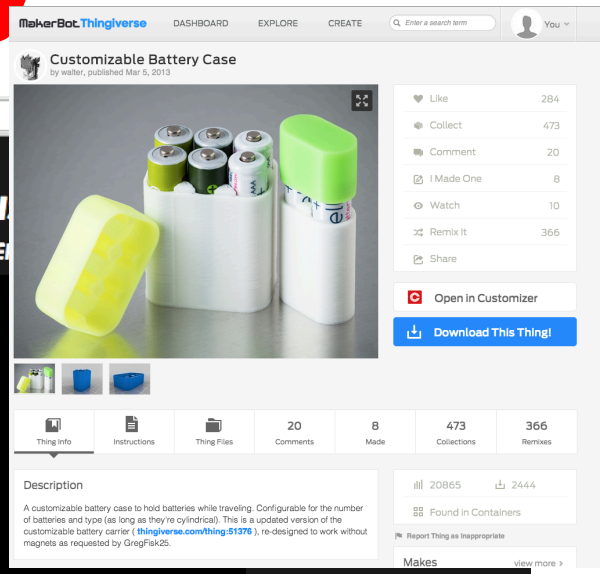
aka **Variability**

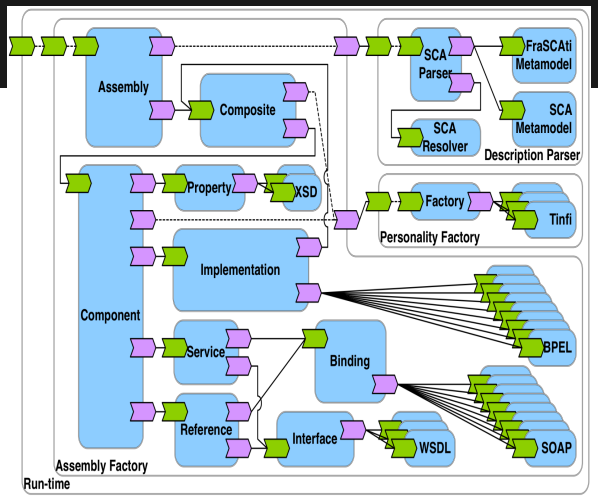
David L. Parnas — "On the design and development of program families" in Transactions on Software Engineering, SE-2(1):1–9, 1976

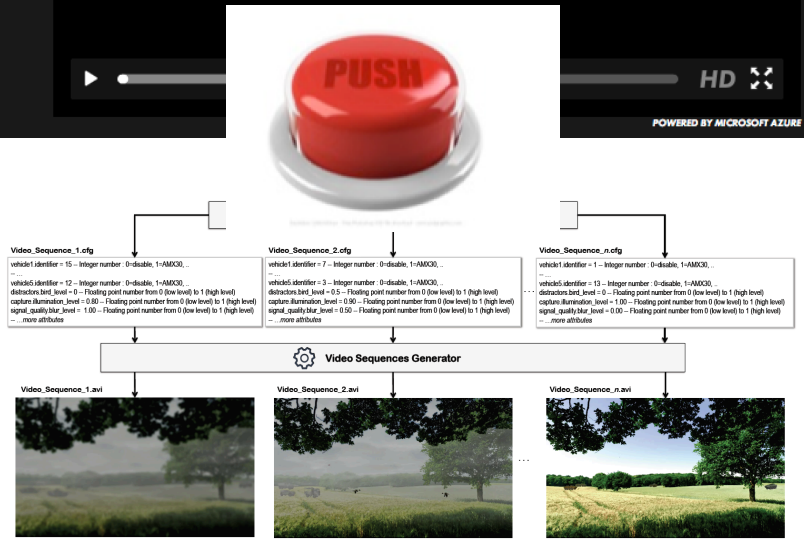
Product Lines

Vari-ability









	 Starter	 Home Premium Upgrade $119.99* Buy	 Professional Upgrade $199.99* Buy	 Ultimate Upgrade $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		✓	✓	✓

NOKIA
Connecting People

Know our past. Create the future...

1982 1984 1985 1986 1987 1988 1989 1990 1991 1992 1993 1994

1995 1996 1997 1998 1999

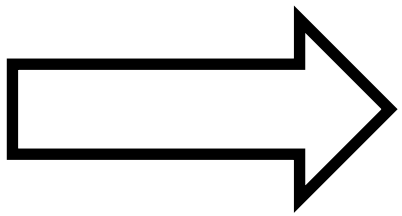
2000 2001 2002

2003 2004

2005

2006 UI

Software-intensive systems

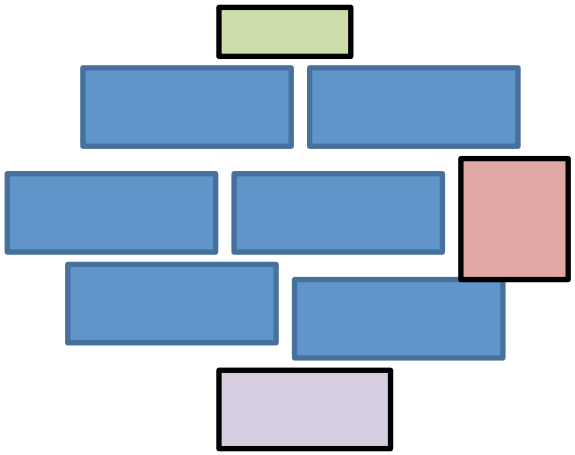


come in many **variants**

Software Product Line Engineering

Factoring out **commonalities**

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



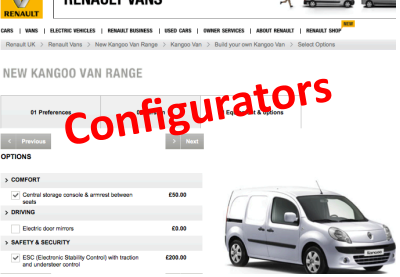
Managing **variabilities**

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

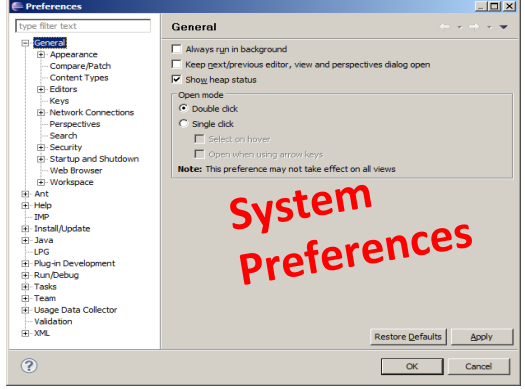
Variability

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

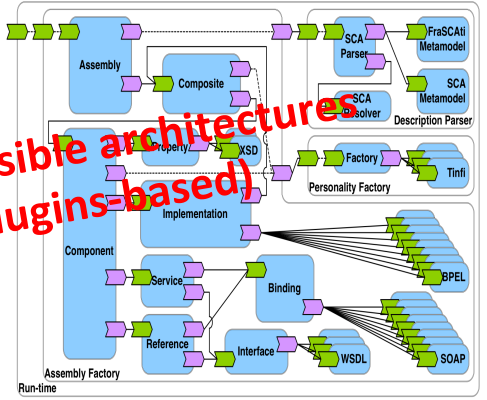
Mikael Svahnberg, Jilles van Gurp, and Jan Bosch (2005)



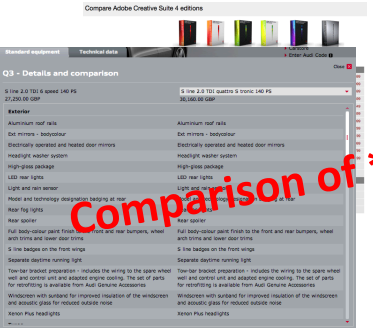
Configurators



System Preferences

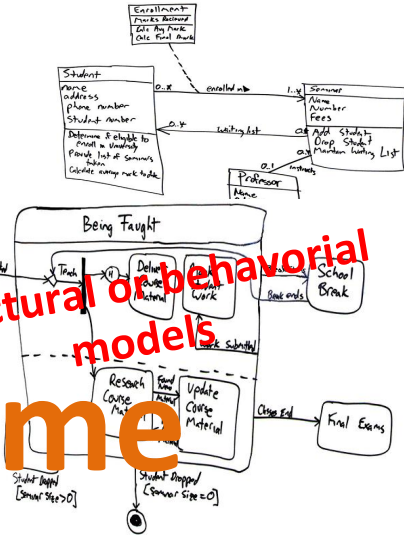


Extensible architectures (eg plugins-based)

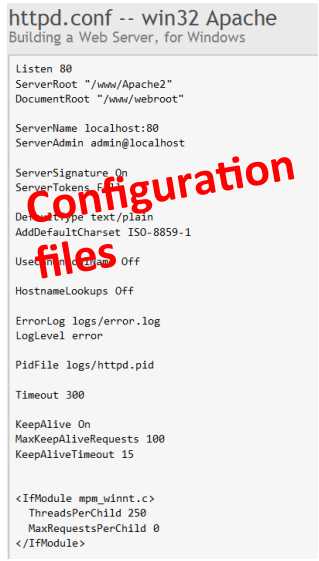


Comparison of *

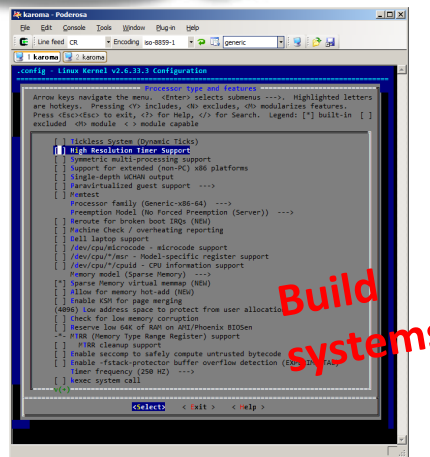
External Variability
Internal Variability
Variability @ run.time



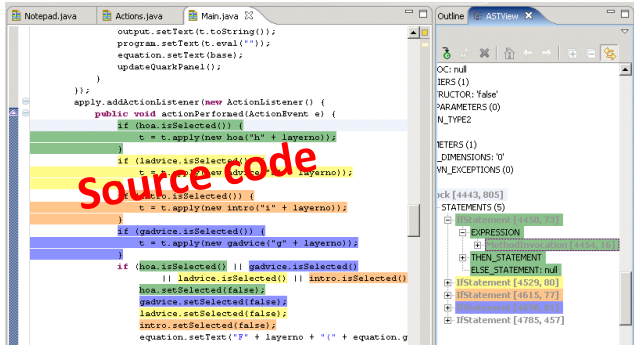
Structural or behavorial models



Configuration files



Build systems



Source code

Developer Tools
Development
Drivers
DTP/Prepress
Educational
Finance
Font Tools
Games
Graphics
HTML Tools
Internet Utilities
iPhone Applications
iPod Tools
Math/Scientific
Multimedia
Network/Admin
Screensave
Security
Spot t Plug Utilities
System ilities
Video
Word Processing

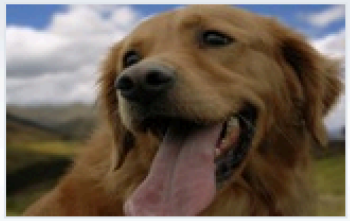
LOBAL PAGES >>
EWS ARCHIVE >>
OFTPEDIA REVIEWS >>
EET THE EDITORS >>

Power Matte 2 1.3 update

Adobe Ar plugin that can extract a ect in an i e

[read

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



Grid 1.1 update

Helps you generate perspective grids

[read more >]

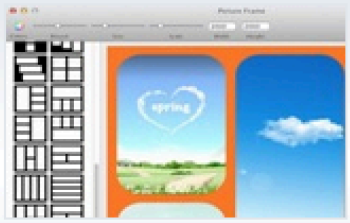
Size:	102 KB
Platform:	Mac OS X 10.8 or later
License:	Commercialware
Rating:	NOT RATED
Downloads:	21
Updated:	June 20th, 07:56 UTC

Picture Frame 2.2 update

Quickly generate multi-frame photos using your Mac

[read more >]

Size:	716 KB
Platform:	Mac OS X 10.6.6 or l...
License:	Commercialware
Rating:	Excellent (5.0/5)
Downloads:	297
Updated:	June 20th, 07:53 UTC

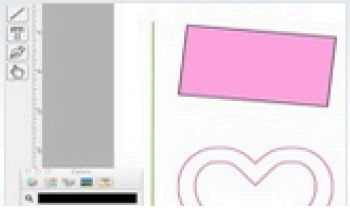


FashionLab Studio 1.1 update

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

[read more >]

Size:	3.10 MB
Platform:	Mac OS X 10.6.6 or l...
License:	Commercialware
Rating:	NOT RATED
Downloads:	3
Updated:	June 20th, 07:49 UTC



Variability

Variability

RENAULT

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

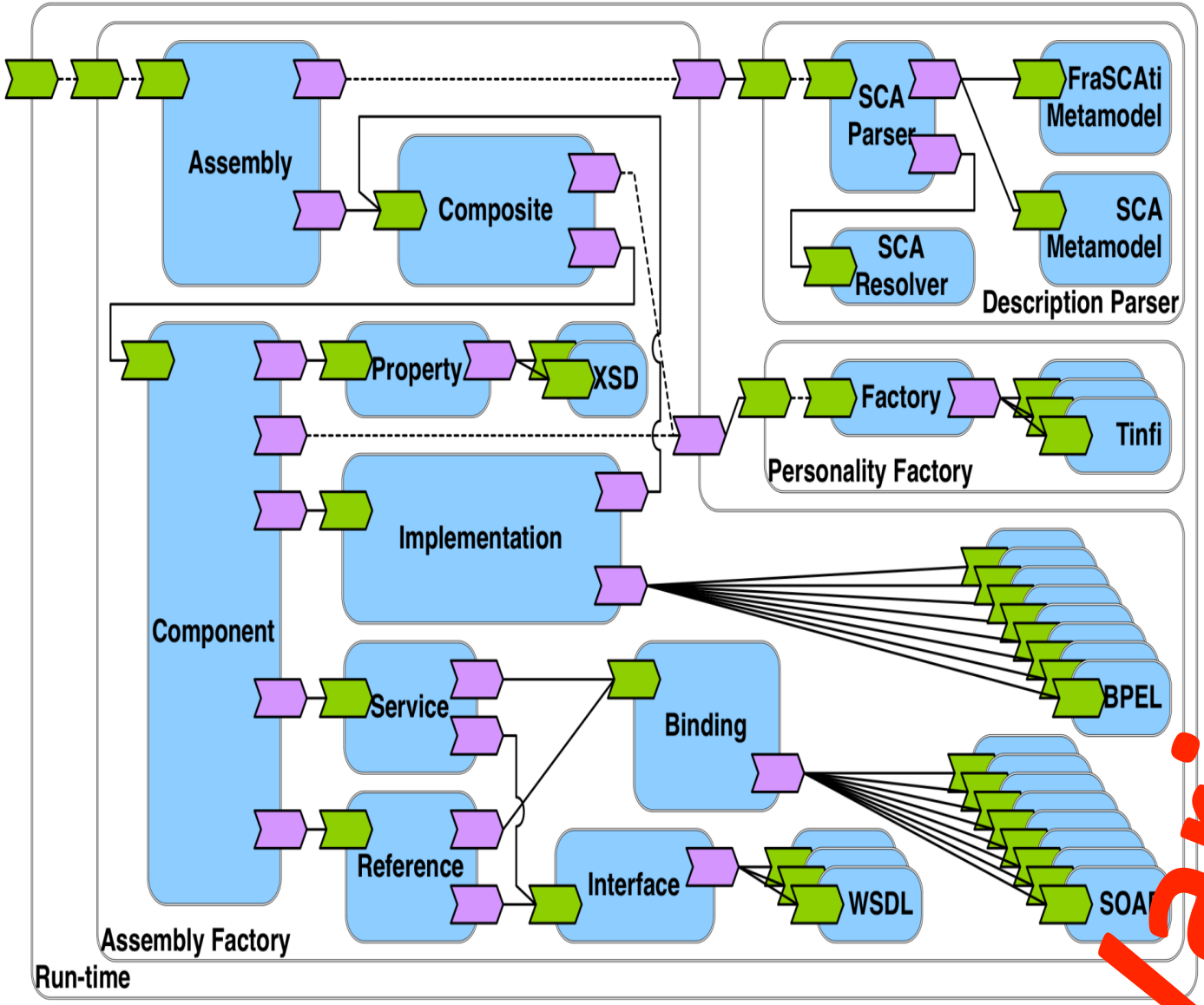
01 Preferences	02 Version	03 Equipment & options

Previous Next

OPTIONS

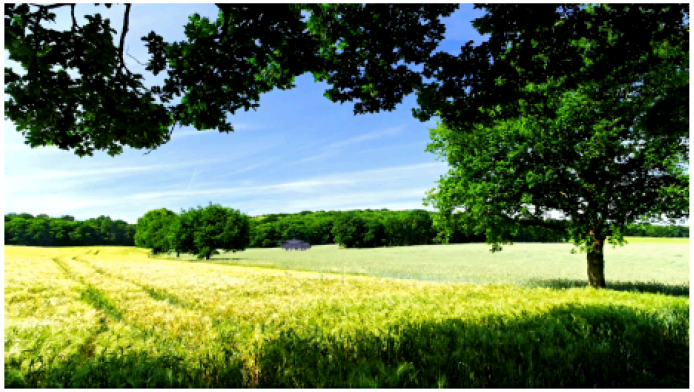
> COMFORT	
☑ Central storage console & armrest between seats	£50.00
> DRIVING	
☐ Electric door mirrors	£0.00
> SAFETY & SECURITY	
☑ ESC (Electronic Stability Control) with traction and understeer control	£200.00

Variability



Variability

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

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

HIPSTER STACK
FOR
JAVA DEVS

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) 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

Video_Sequence_1.cfg

vehicle1.identifier = 15 -- Integer number : 0=disable, 1=AMX30, ..
-- ...
vehicle5.identifier = 12 -- Integer number : 0=disable, 1=AMX30, ..
distractors.bird_level = 0 -- Floating point number from 0 (low level) to 1 (high level)
capture.illumination_level = 0.80 -- Floating point number from 0 (low level) to 1 (high level)
signal_quality.blur_level = 1.00 -- Floating point number from 0 (low level) to 1 (high level)
-- *...more attributes*

Video_Sequence_2.cfg

vehicle1.identifier = 7 -- Integer number : 0=disable, 1=AMX30, ..
-- ...
vehicle5.identifier = 3 -- Integer number : 0=disable, 1=AMX30, ..
distractors.bird_level = 0.5 -- Floating point number from 0 (low level) to 1 (high level)
capture.illumination_level = 0.90 -- Floating point number from 0 (low level) to 1 (high level)
signal_quality.blur_level = 0.50 -- Floating point number from 0 (low level) to 1 (high level)
-- *...more attributes*

. . .

Video_Sequence_*n*.cfg

vehicle1.identifier = 1 -- Integer number : 0=disable, 1=AMX30, ..
-- ...
vehicle5.identifier = 13 -- Integer number : 0=disable, 1=AMX30, ..
distractors.bird_level = 0 -- Floating point number from 0 (low level) to 1 (high level)
capture.illumination_level = 1.00 -- Floating point number from 0 (low level) to 1 (high level)
signal_quality.blur_level = 0.00 -- Floating point number from 0 (low level) to 1 (high level)
-- *...more attributes*

Video Sequences Generator

Video_Sequence_1.avi

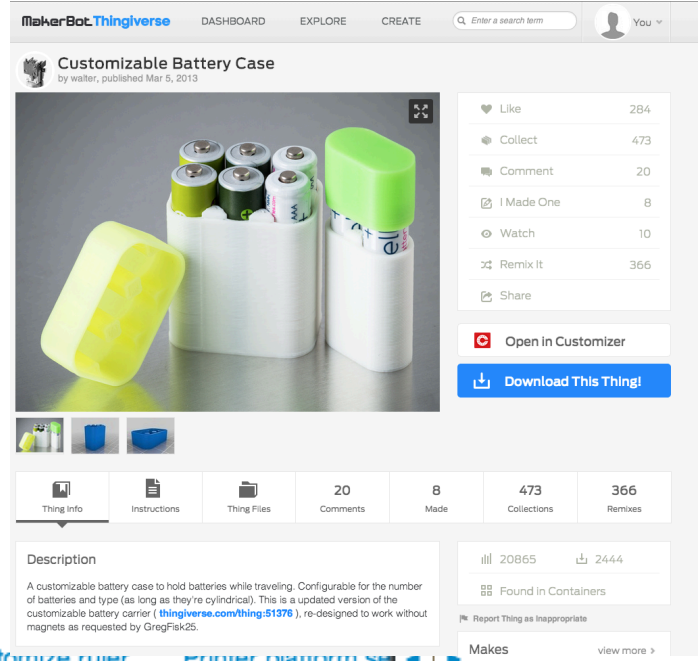
Video_Sequence_2.avi

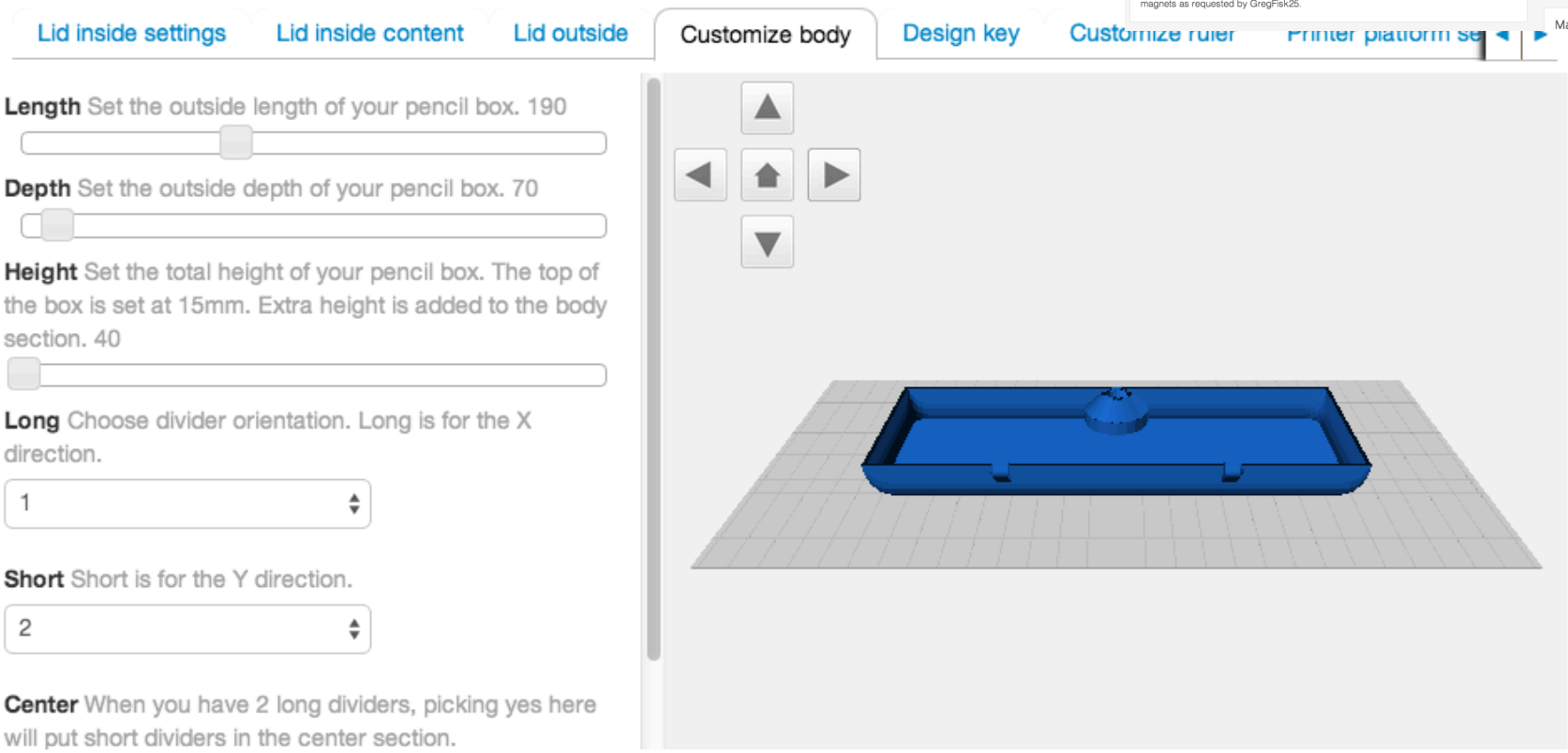
. . .

Video_Sequence_*n*.avi

```
/* [Customize body] */

//Set the outside length of your pencil box.
length=190;//[70:400]
//Set the outside depth of your pencil box.
depth=70;//[50:400]
//Set the total height of your pencil box.  The top of the box is set at 15mm.
//Extra height is added to the body section.
height=40;//[40:150]

//Choose divider orientation.  Long is for the X direction.
long = 1;//[0,1,2]
//Short is for the Y direction.
short = 2;//[0,1,2,3]
//When you have 2 long dividers,
// picking yes here will put short dividers in the center section.
center = 0;//[1:Yes,0:No]
```

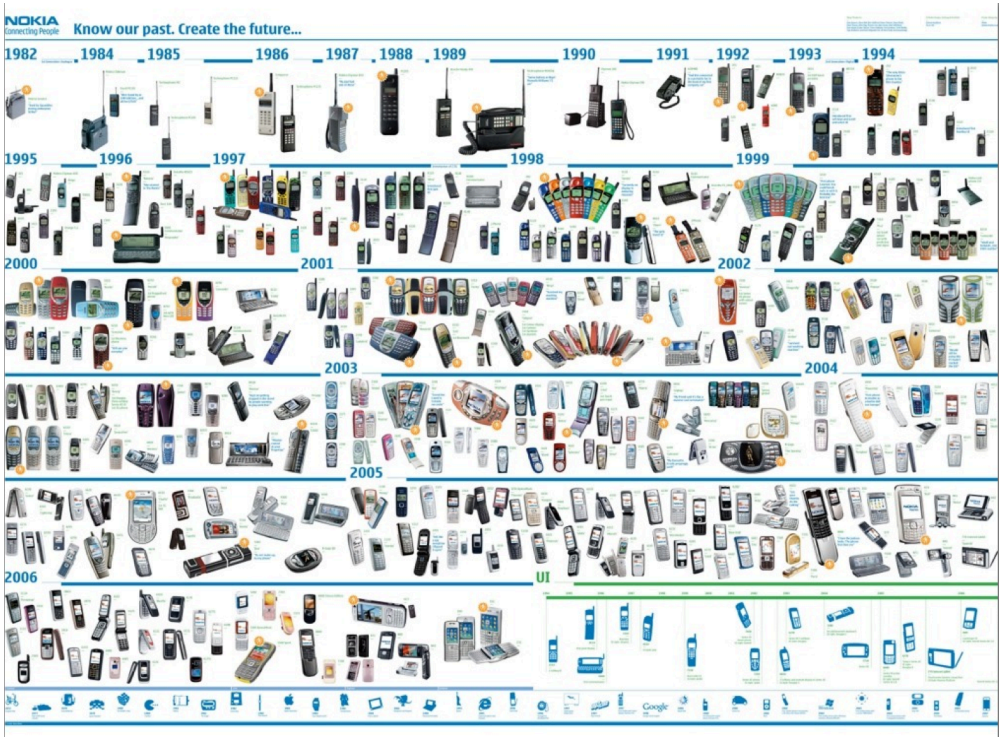
1





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**

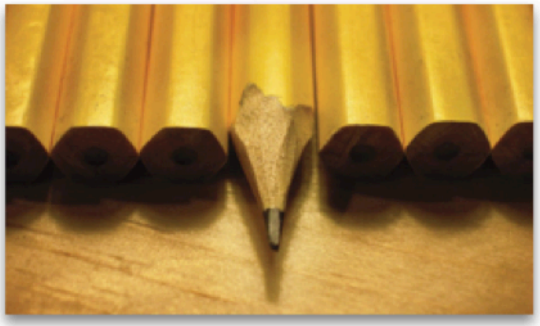


Benefits

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

BOSCH
Invented for life

PHILIPS

ERICSSON

NOKIA
Connecting People

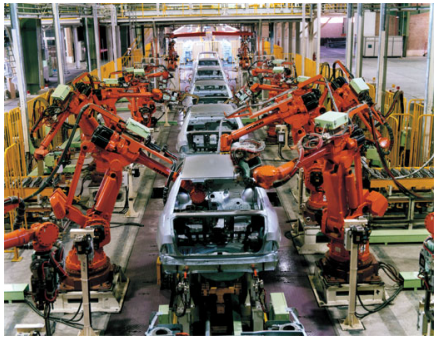
CelsiusTech

Lucent Technologies
Bell Labs Innovations

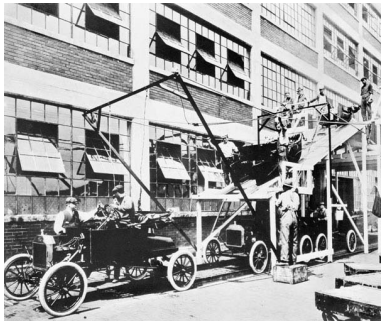
Printer Firmware

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

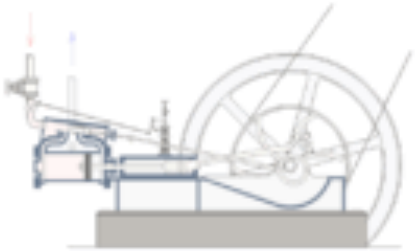
A Bit of History: Industrial Revolution



1980s



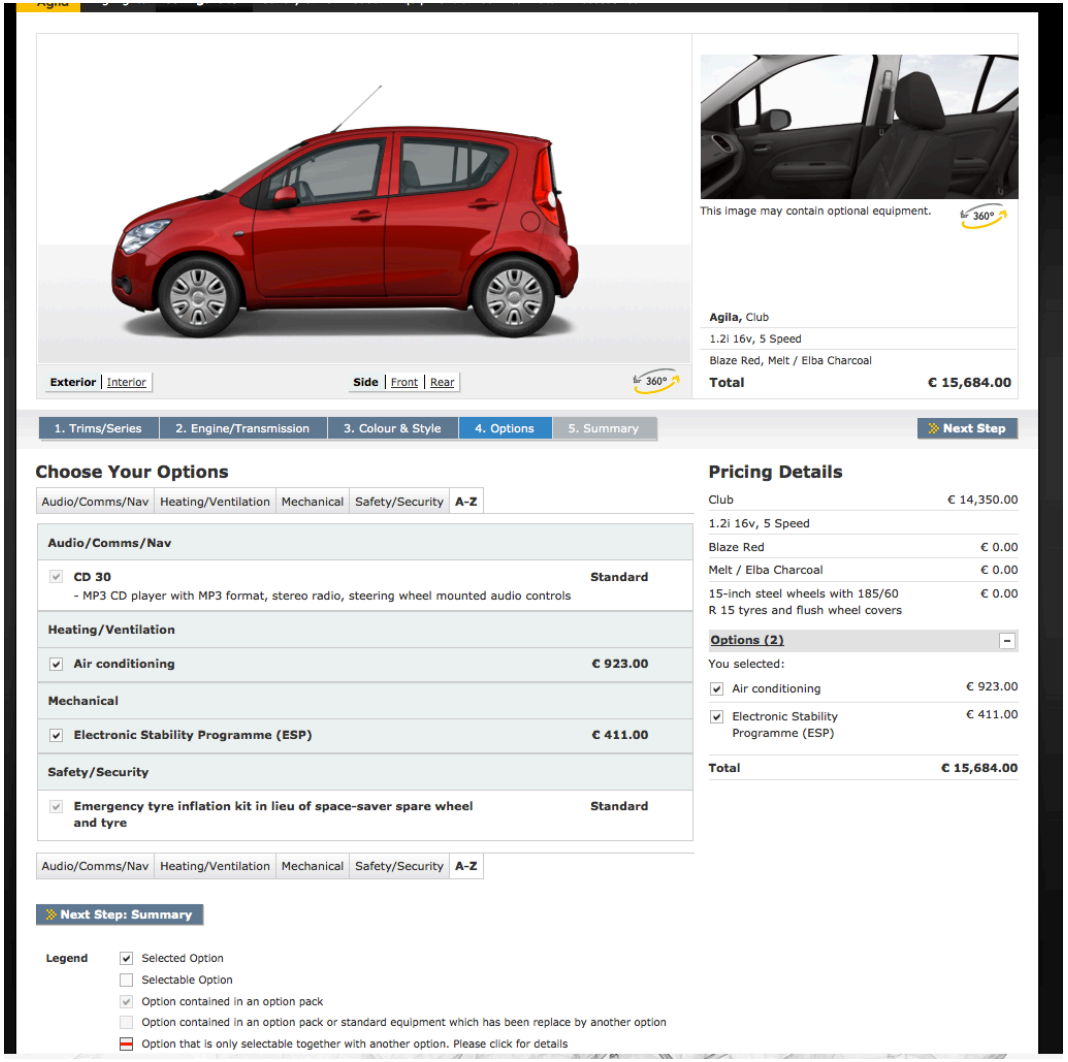
1901
Henry Ford



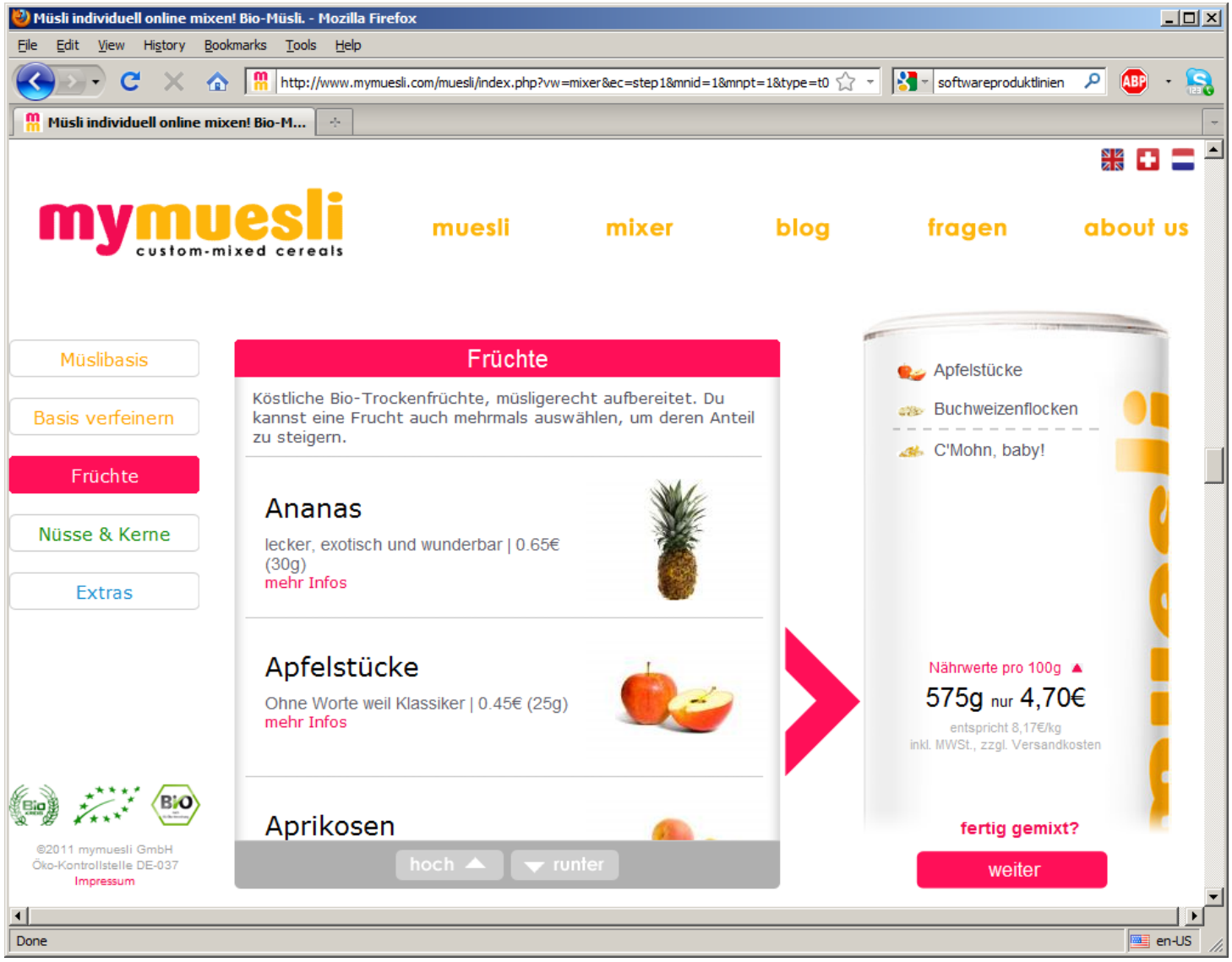
1698
Thomas Savery

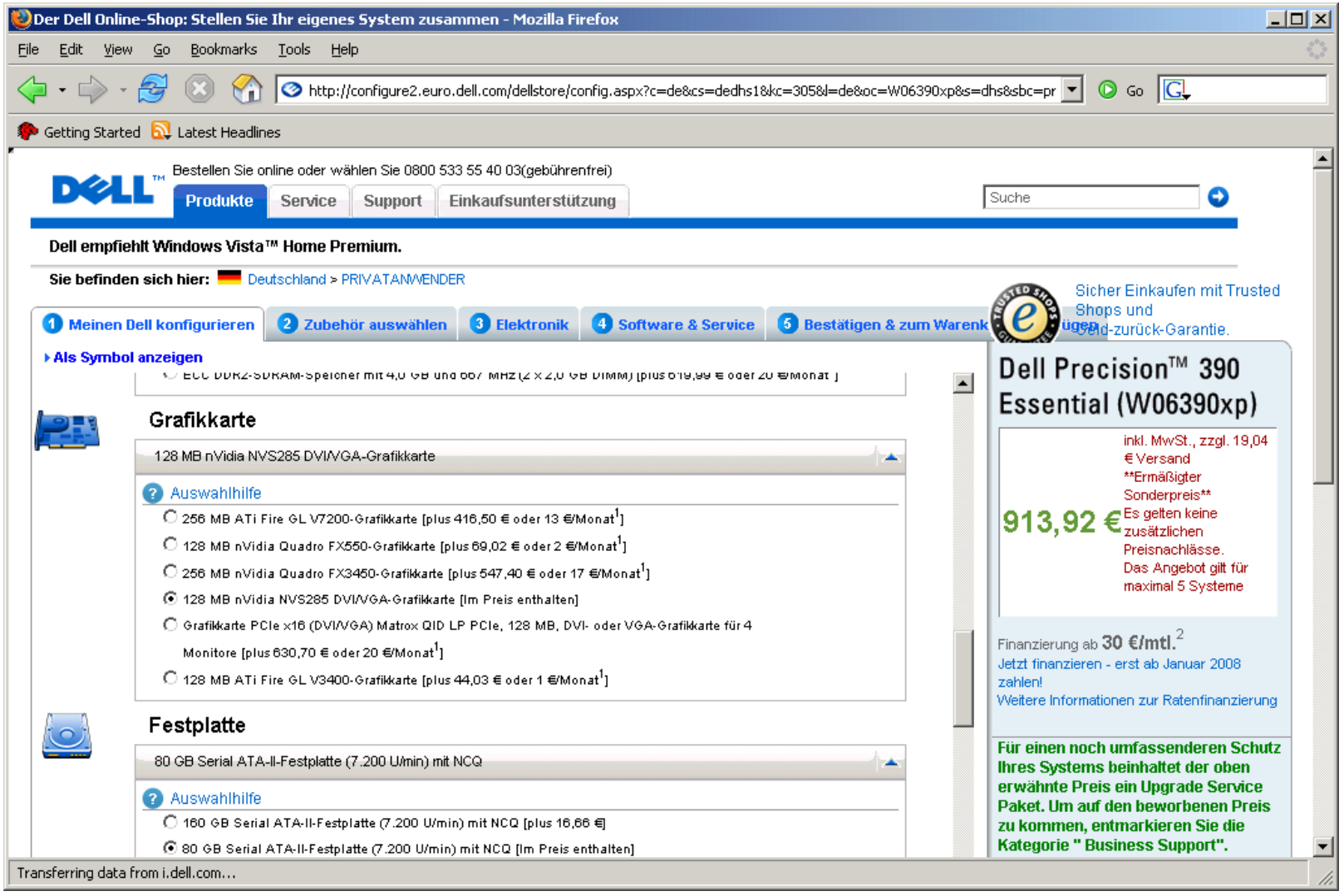
Nowaday: Product Lines Everywhere

Product Lines of Cars

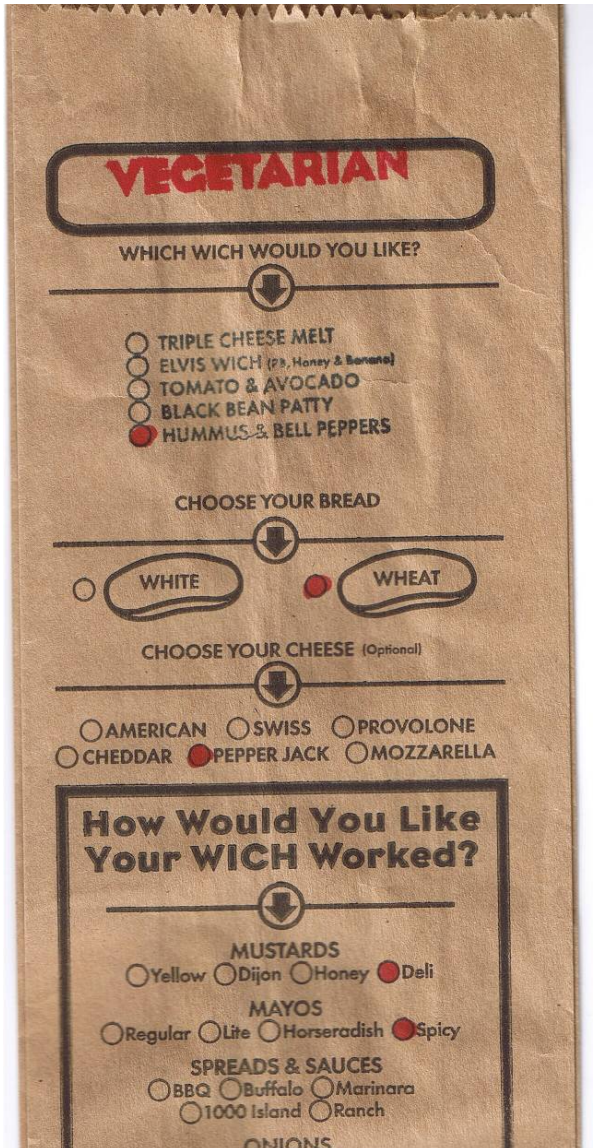



Willkommen bei selve - the shoe individualizer
http://www.selve.net/index_js.html
KOLLEKTION
FUSSTYP
MYSELVE
INFO
HOME
MODELLE
LOOKBOOK
SELVE ID
PASSWORT
>>ANMELDEN
selve
selve Kollektion -> Style: casuals -> Modell: Opal
modell-details
>>hier clicken
>>SELVE SCHUHREGAL
inhalt: 0
>>SHOPPING BAG
inhalt : 0
A. Erstes Oberleder
Velours leder Sand
Velours leder Bordeaux
Velours leder Cognac
Velours leder Sand
B.
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
http://www.mymuesli.com/muesli/index.php?vw=mixer&ec=step1&mnid=1&mnpt=1&type=t0
softwareproduktlinien
Müsli individuell online mixen! Bio-M...
mymuesli
custom-mixed cereals
muesli
mixer
blog
fragen
about us
Müslibasis
Basis verfeinern
Früchte
Nüsse & Kerne
Extras
Früchte
Köstliche Bio-Trockenfrüchte, müsligerecht 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
©2011 mymuesli GmbH
Öko-Kontrollstelle DE-037
Impressum
Done
en-US


Der Dell Online-Shop: Stellen Sie Ihr eigenes System zusammen - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://configure2.euro.dell.com/dellstore/config.aspx?c=de&cs=dedhs1&kc=305&l=de&oc=W06390xp&s=dhs&sbc=pr
Go
Getting Started
Latest Headlines
Bestellen Sie online oder wählen Sie 0800 533 55 40 03(gebührenfrei)
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 Warenk
Sicher Einkaufen mit Trusted Shops und Geld-zurück-Garantie.
Als Symbol anzeigen
Grafikkarte
128 MB nVidia NVS285 DVI/VGA-Grafikkarte
Auswahlhilfe
256 MB ATi Fire GL V7200-Grafikkarte [plus 416,50 € oder 13 €/Monat1]
128 MB nVidia Quadro FX550-Grafikkarte [plus 69,02 € oder 2 €/Monat1]
256 MB nVidia Quadro FX3450-Grafikkarte [plus 547,40 € oder 17 €/Monat1]
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 €/Monat1]
128 MB ATi Fire GL V3400-Grafikkarte [plus 44,03 € oder 1 €/Monat1]
Festplatte
80 GB Serial ATA-II-Festplatte (7.200 U/min) mit NCQ
Auswahlhilfe
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]
Dell Precision™ 390 Essential (W06390xp)
913,92 €
inkl. MwSt., zzgl. 19,04 € Versand
Ermäßigter Sonderpreis
Es gelten keine zusätzlichen Preisnachlässe.
Das Angebot gilt für maximal 5 Systeme
Finanzierung ab 30 €/mtl.2
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...

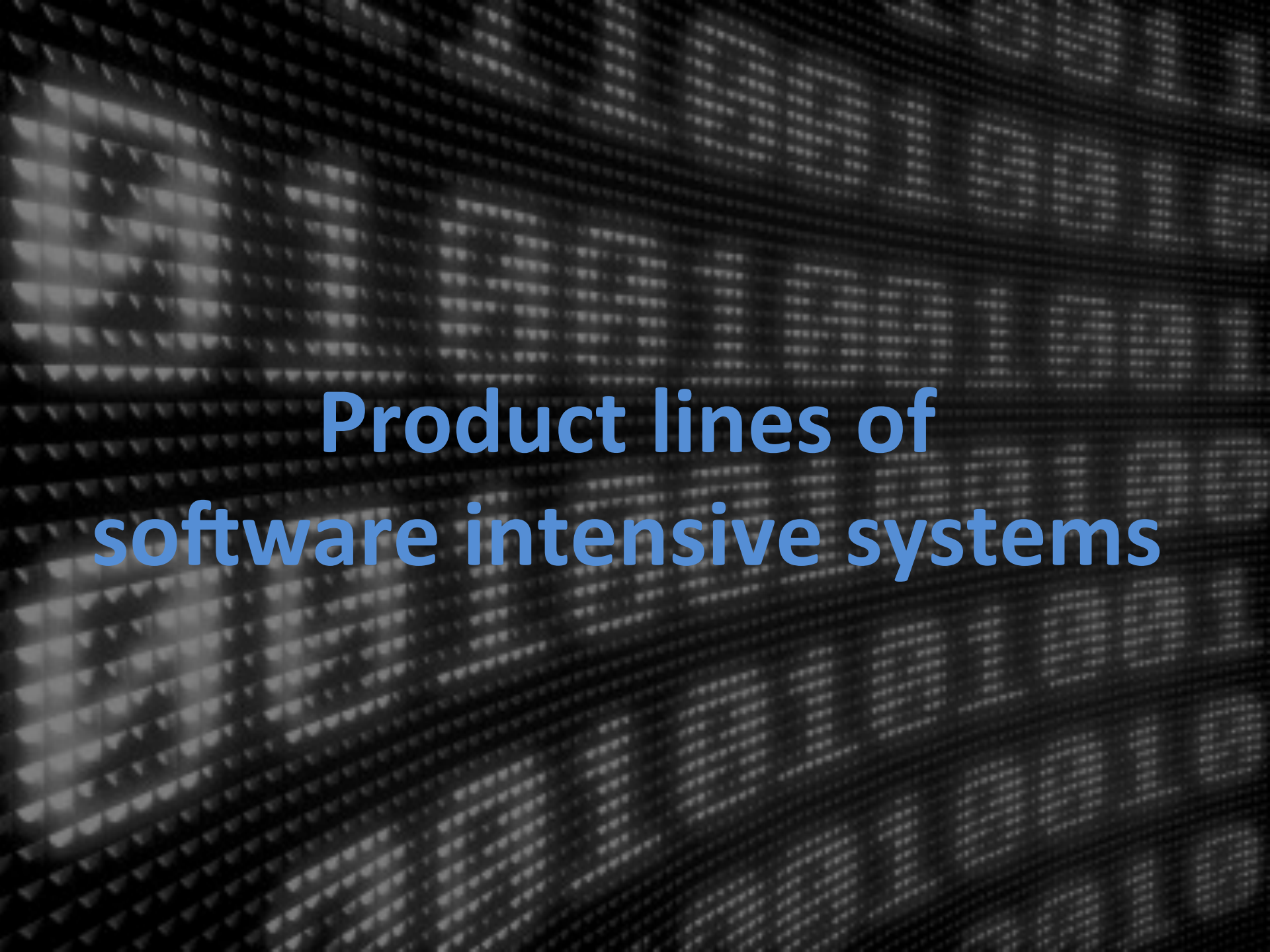
Food? Product lines!




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

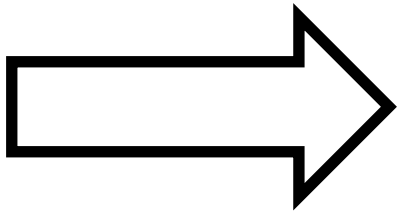
Mass production

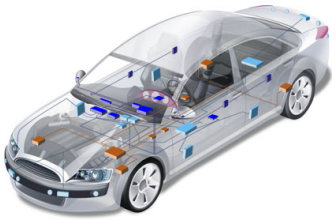

What about
software?


Product lines of
software intensive systems

Software intensive systems

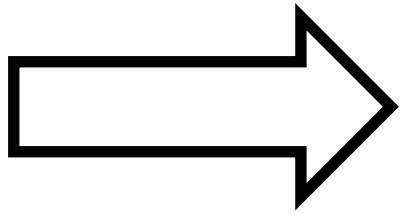
are declined in many **variants**





Software intensive systems

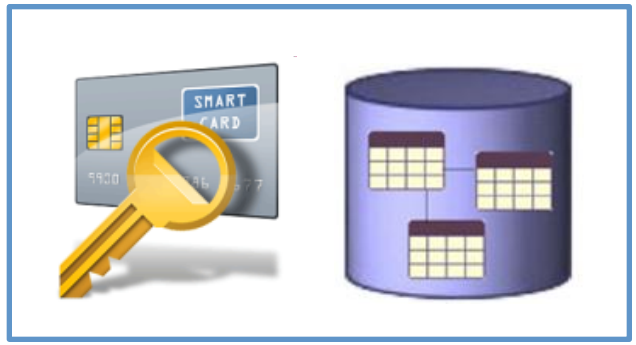
are declined in many **variants**



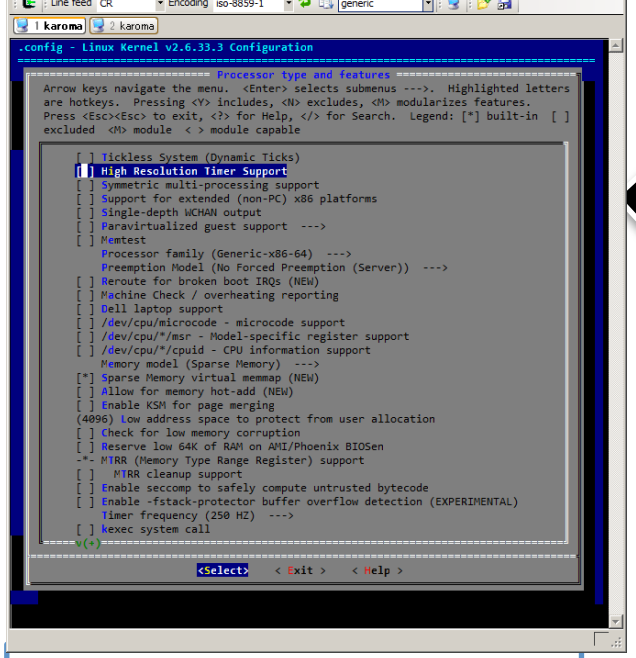
Software Product Lines

Car

Database Engine

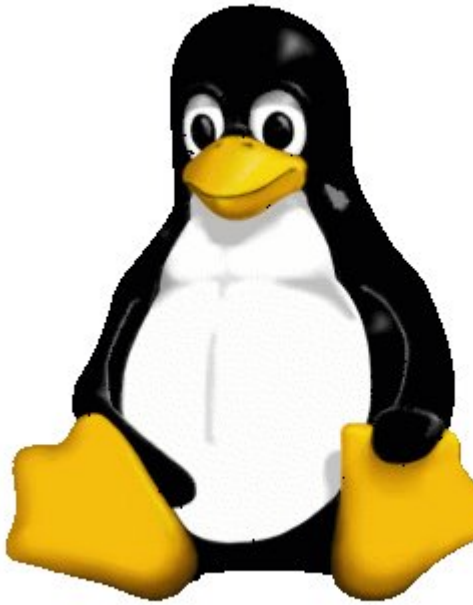


Printer Firmware

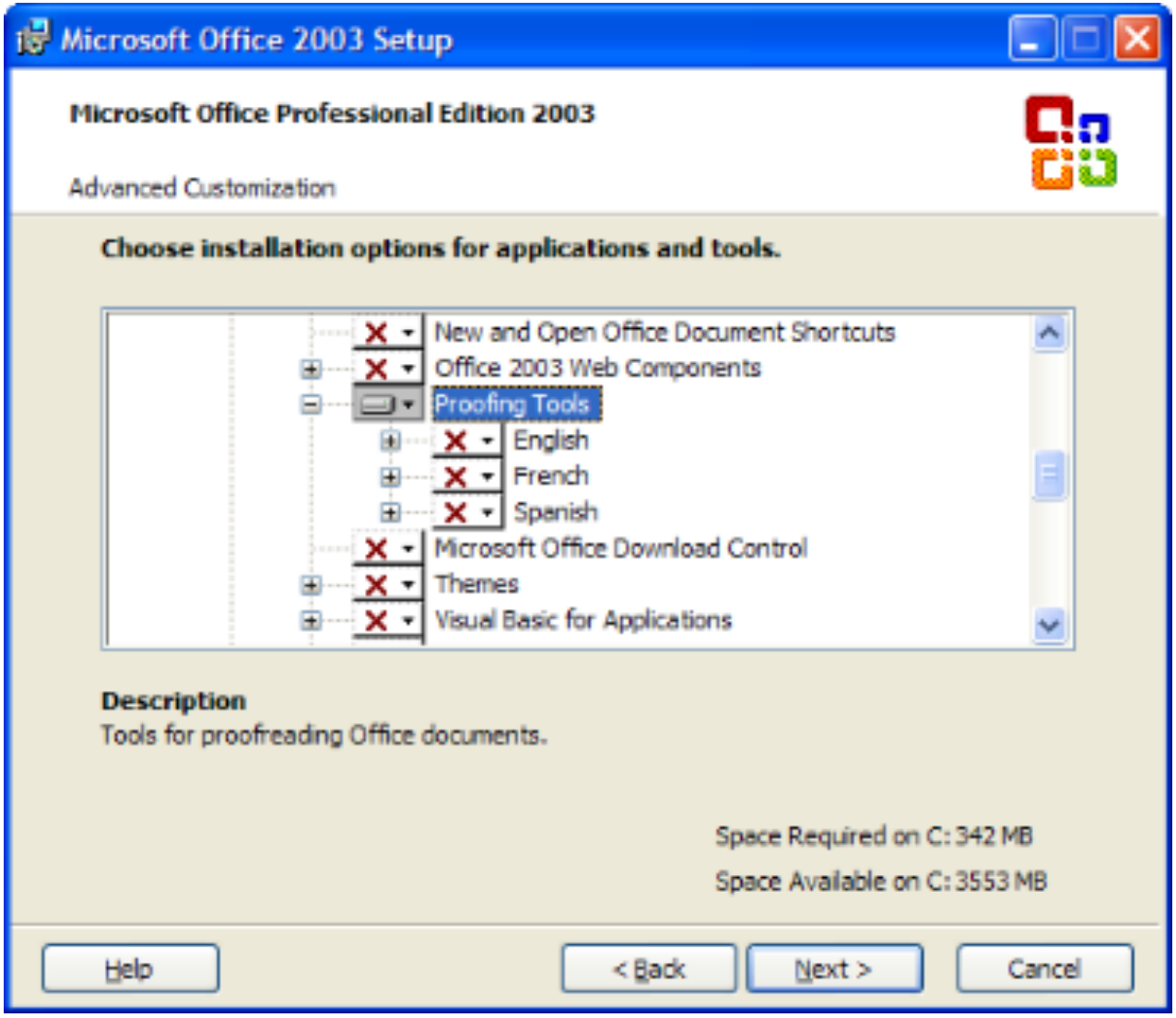


Linux Kernel

Linux-Kernel



Features in Microsoft Office



Bref

bref30ans.canalplus.fr

CANAL+ 30ANS

BREF | PROGRAMMES | SURPRISES | EXPO

bref.
CANAL a 30 ans.

ETAPE 1 : DONNE TON PRENOM

MATHIEU → OK

Online Generator

bref30ans.canalplus.fr/#c

ETAPE 2 : CHOISIS 3 BONS SOUVENIRS

Variant


bref30ans.canalplus.fr/#video
ETAPE 3 : JE REGARDE MON EPISODE UNIQUE
DEJÀ 761 545 EPISODES GENERES.
MATHIEU+
présente
HD
POWERED BY MICROSOFT AZURE

Quizz Time

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"

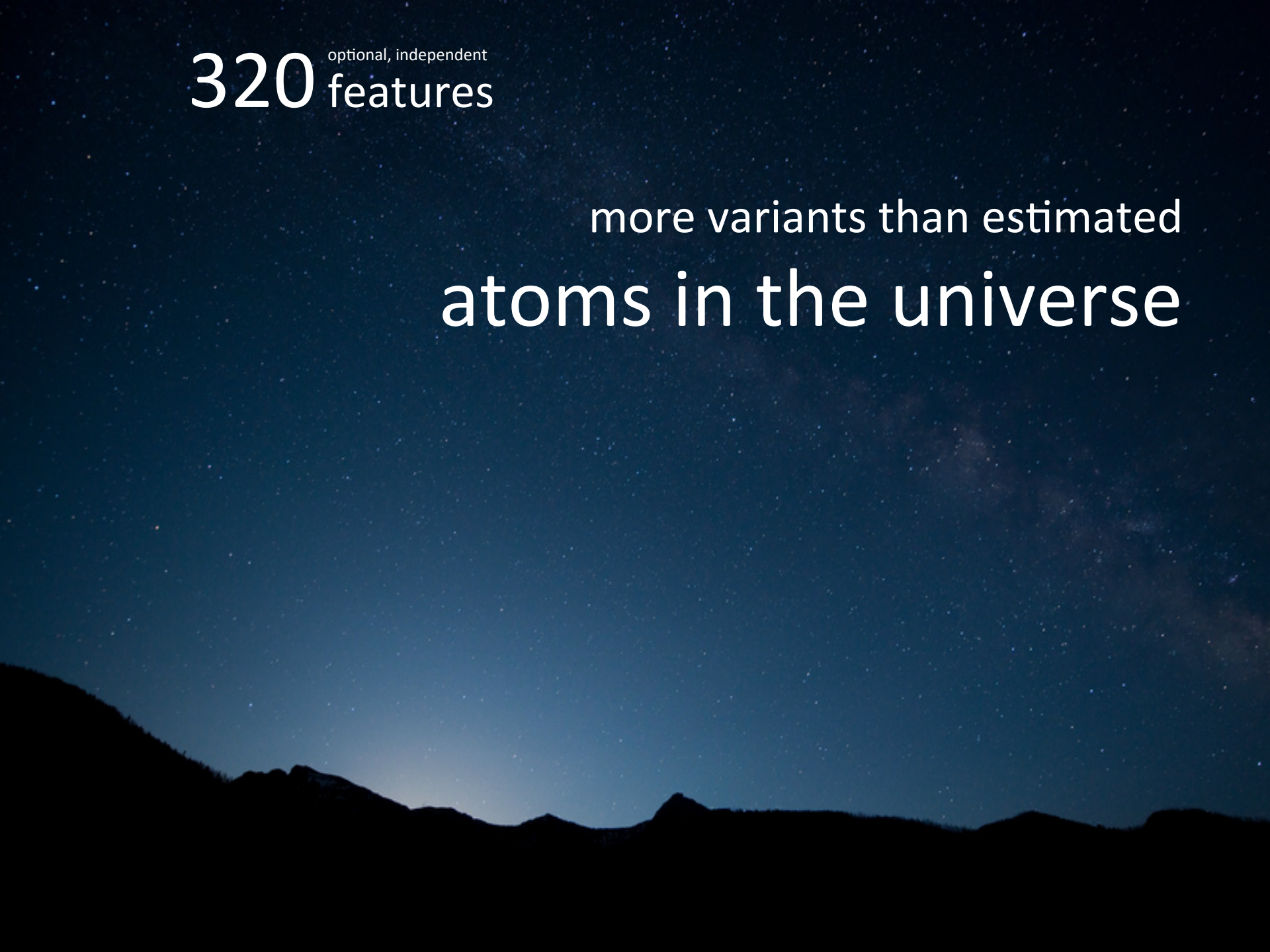
Mikael Svahnberg, Jilles van Gurp, and Jan Bosch (2005)


Variability = Complexity

33 optional, independent features

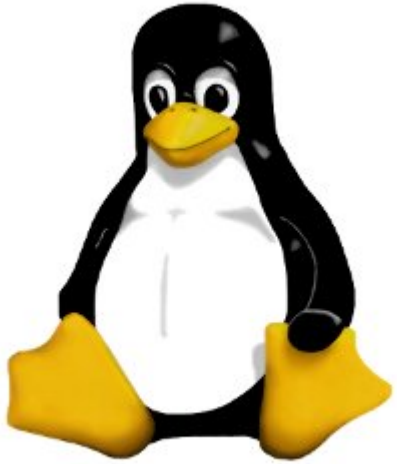
a unique variant for every

person on this planet

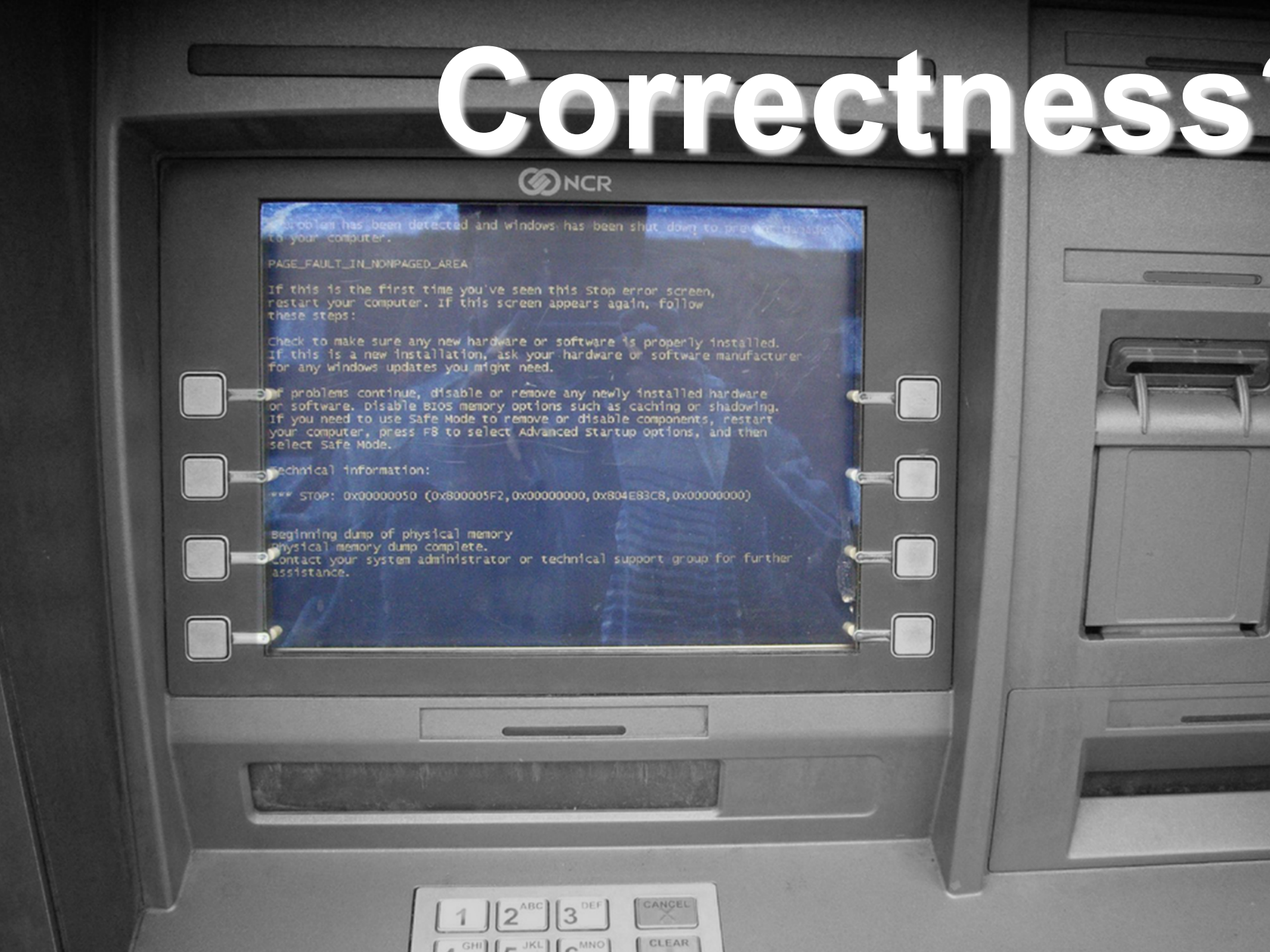

320 optional, independent features
more variants than estimated
atoms in the universe

2000 features

10000 features




Automation?
Avoid solving the same problem!
2, 3…n times


Correctness
NCR
PAGE_FAULT_IN_NONPAGED_AREA
If this is the first time you've seen this Stop error screen,
restart your computer. If this screen appears again, follow
these steps:
Check to make sure any new hardware or software is properly installed.
If this is a new installation, ask your hardware or software manufacturer
for any windows updates you might need.
If problems continue, disable or remove any newly installed hardware
or software. Disable BIOS memory options such as caching or shadowing.
If you need to use Safe Mode to remove or disable components, restart
your computer, press F8 to select Advanced Startup Options, and then
select Safe Mode.
Technical information:
*** STOP: 0x00000050 (0x800005F2,0x00000000,0x804E83C8,0x00000000)
Beginning dump of physical memory
Physical memory dump complete.
Contact your system administrator or technical support group for further
assistance.
1
2 ABC
3 DEF
CANCEL
CLEAR

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

Customization

Automation

Commonality

Variability

Assembly Line and Mass Customization

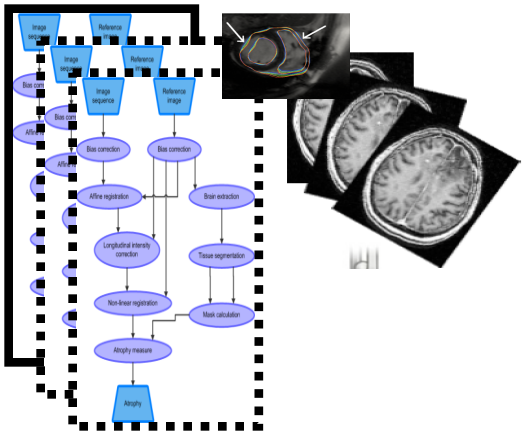
Reuse and Mass Customization

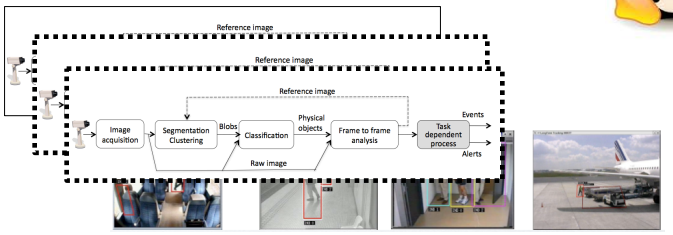

Starting from scratch?


You cannot start from scratch

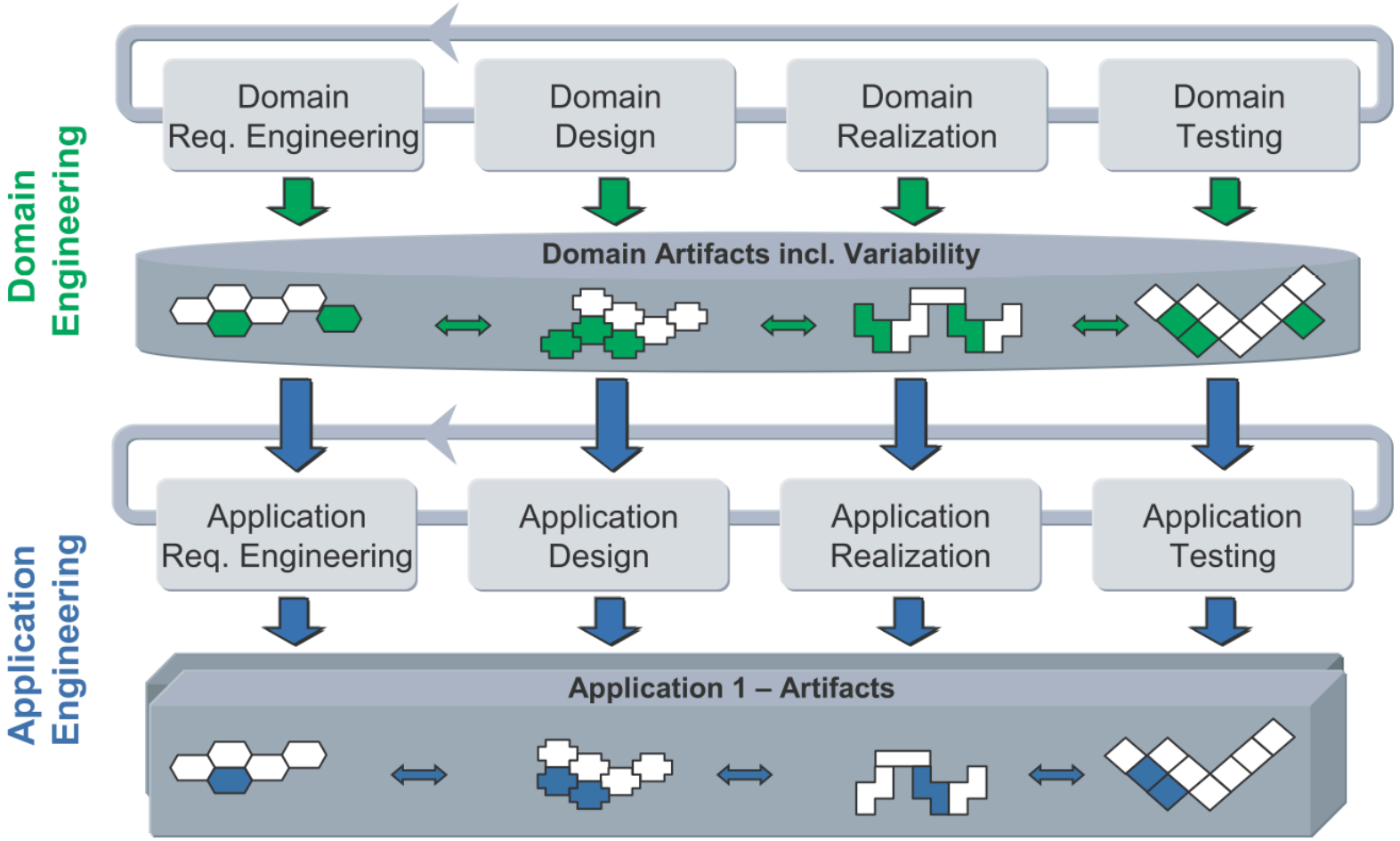
"a set of software- intensive systems that share a common, managed set of features satisfying the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way" **[Clements et al., 2001]**

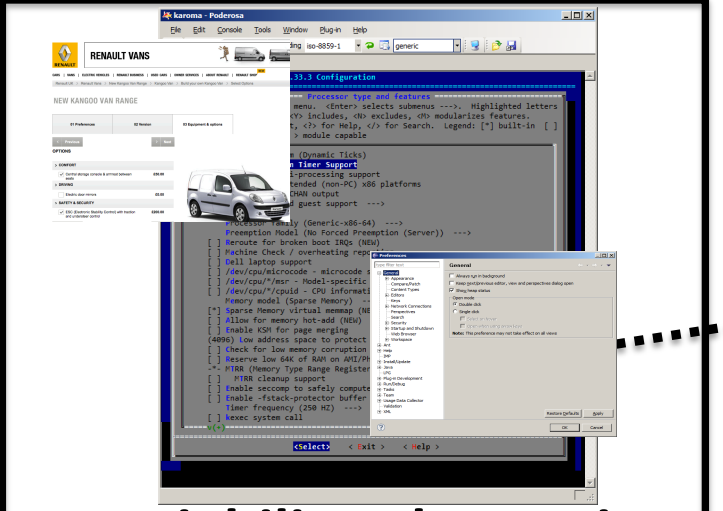
Software Product Lines





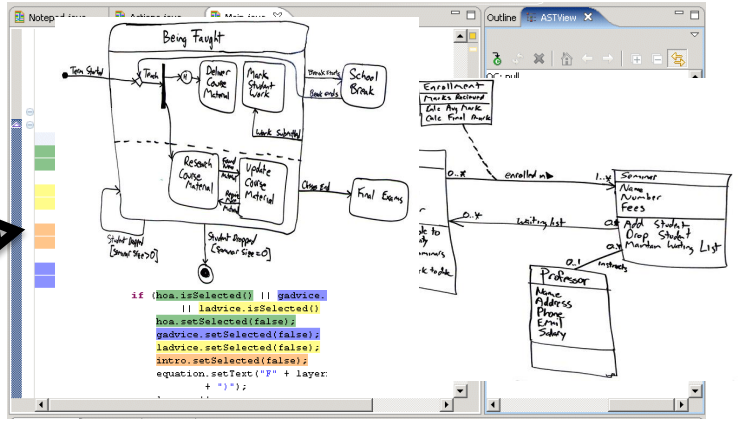
Software Product-Line Engineering



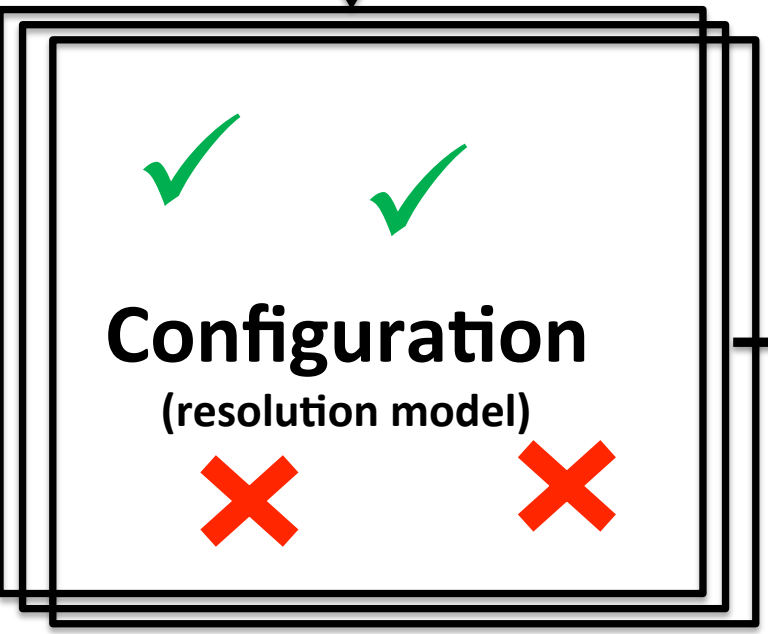


Variability Abstraction Model (VAM)

Variability Realization Model (VRM)



Domain Artefacts (e.g., models)



Configuration

(resolution model)

Software Generator

(derivation engine)

Mapping: an example

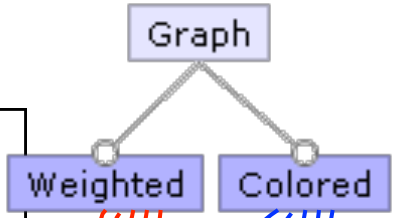
```
class Graph {
  Vector nv = new Vector(); Vector ev = new Vector();
  Edge add(Node n, Node m) {
    Edge e = new Edge(n, m);
    nv.add(n); nv.add(m); ev.add(e);
    e.weight = new Weight();
    return e;
  }
  Edge add(Node n, Node m, Weight w)
    Edge e = new Edge(n, m);
    nv.add(n); nv.add(m);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

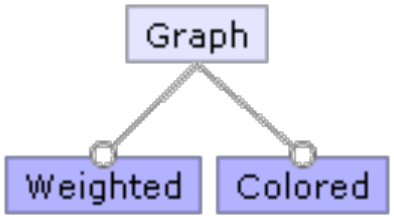
```
class Node {
  int id = 0;
  Color color = new Color();
  void print() {
    Color.setDisplayColor(color);
    System.out.print(id);
  }
}
```

```
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() { ... } }
```





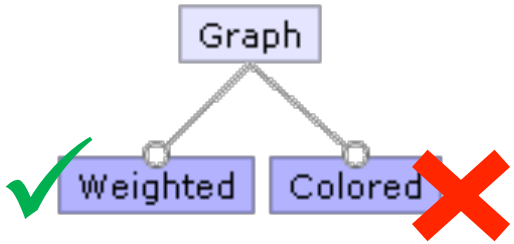

```
class Graph {
  Vector nv = new Vector(); Vector ev = new Vector();
  Edge add(Node n, Node m) {
    Edge e = new Edge(n, m);
    nv.add(n); nv.add(m); ev.add(e);
    /*if[WEIGHT]*/
    e.weight = new Weight();
    /*end[WEIGHT]*/
    return e;
  }
  /*if[WEIGHT]*/
  Edge add(Node n, Node m, Weight 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]*/
```

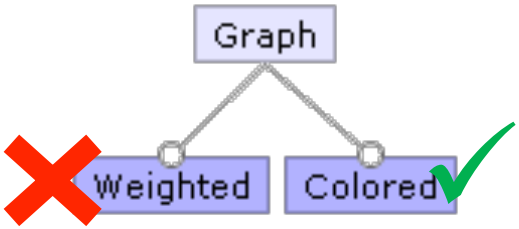



```
class Graph {
  Vector nv = new Vector(); Vector ev = new Vector();
  Edge add(Node n, Node m) {
    Edge e = new Edge(n, m);
    nv.add(n); nv.add(m); ev.add(e);
e.weight = new Weight();
return e;
  }
Edge add(Node n, Node m, Weight w)
    Edge e = new Edge(n, m);
    nv.add(n); nv.add(m); ev.add(e);
    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() { ... } }
```

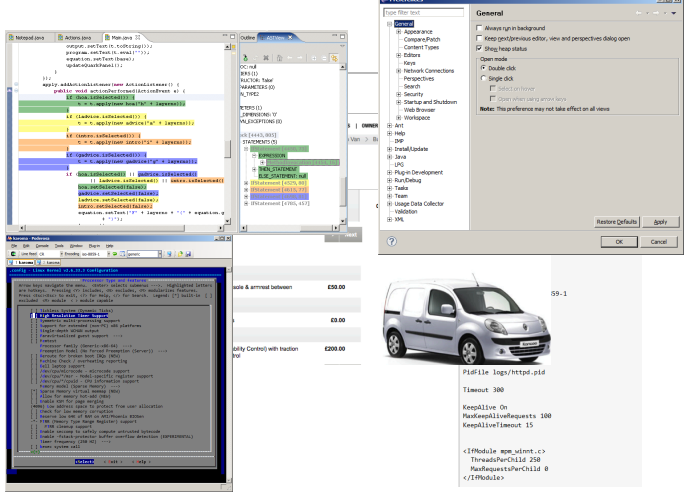



```
class Graph {
  Vector nv = new Vector(); Vector ev = new Vector();
  Edge add(Node n, Node m) {
    Edge e = new Edge(n, m);
    nv.add(n); nv.add(m); ev.add(e);
return e;
  }
void print() {
    for(int i = 0; i < ev.size(); i++) {
      ((Edge)ev.get(i)).print();
    }
  }
}
```

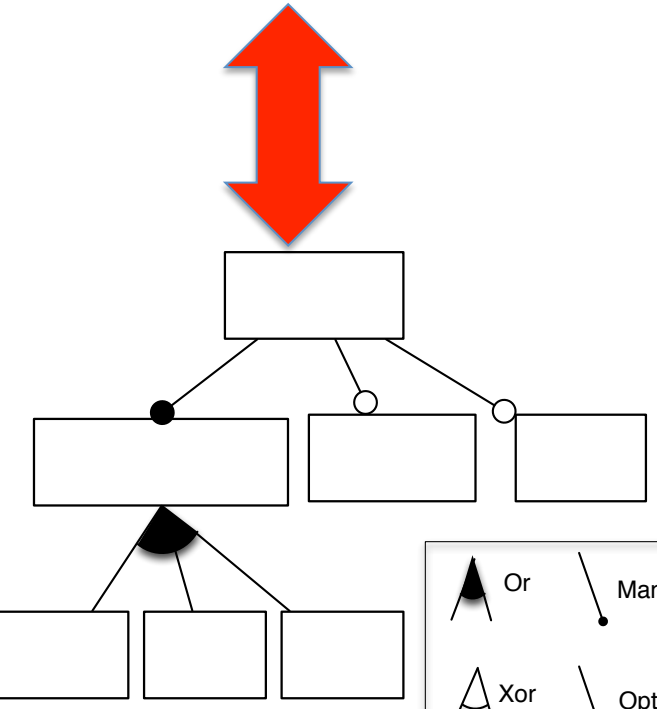
```
class Edge {
  Node a, b;
  Color color = new Color();
Edge(Node _a, Node _b) { a = _a; b = _b; }
  void print() {
Color.setDisplayColor(color);
a.print(); b.print();
}
}
```

```
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);
  }
}
```



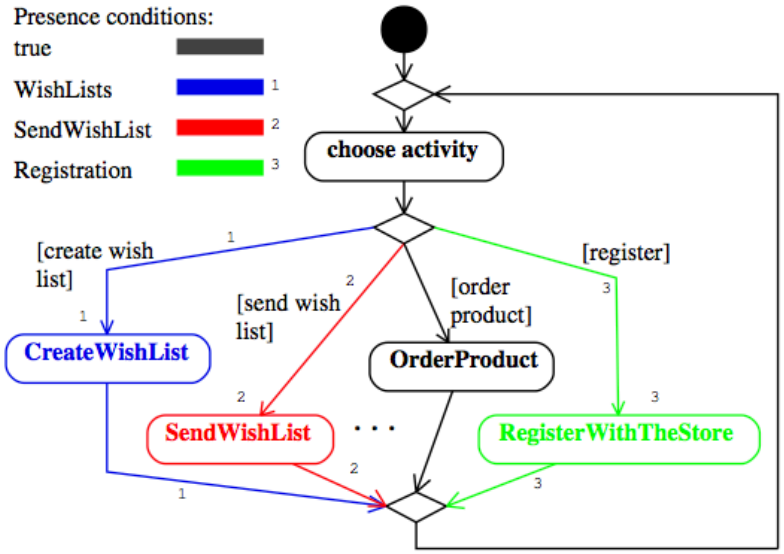
variantes de code (e.g., Java ou C)
variantes d'interface utilisateur
variantes de séquence vidéo
variantes de langages
variantes de **modèles**
...



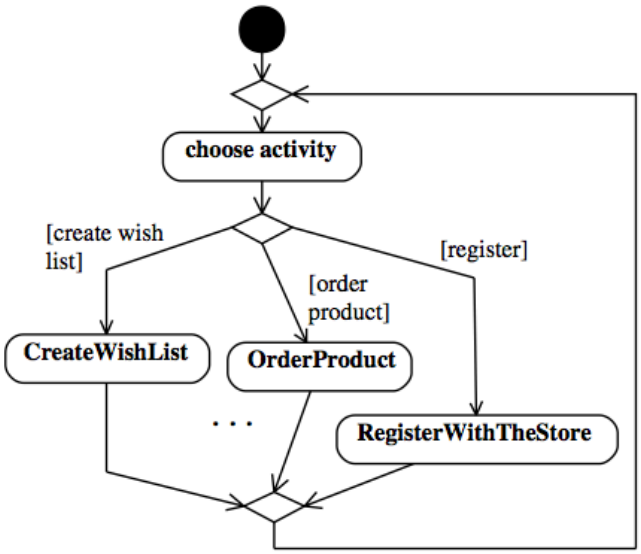
Feature Models

[Czarnecki et al. (GPCE'05)]

Feature-based Model Templates



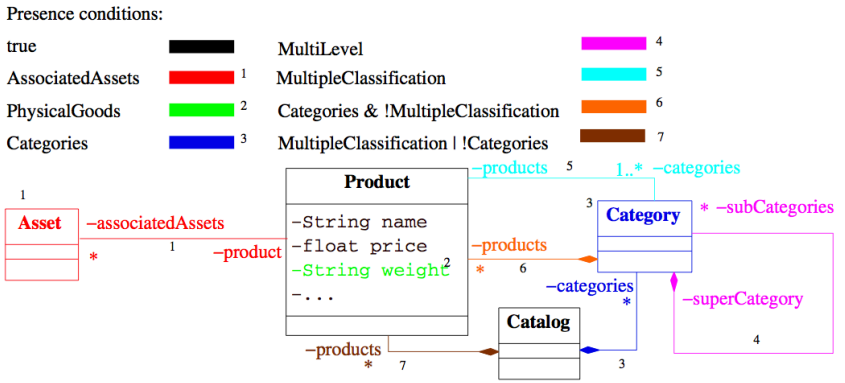
(a) Storefront template



(b) Storefront instance

[Czarnecki et al. (GPCE'05)]

Approach



Refers to features through annotations

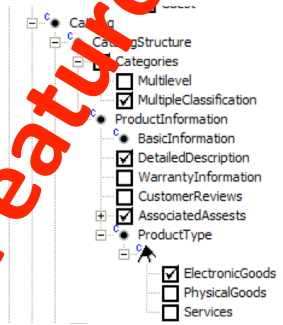
Feature model

Model template expressed in target notation and annotated with presence conditions and meta–expressions

Manual configuration process

Feature configuration

Features/Options

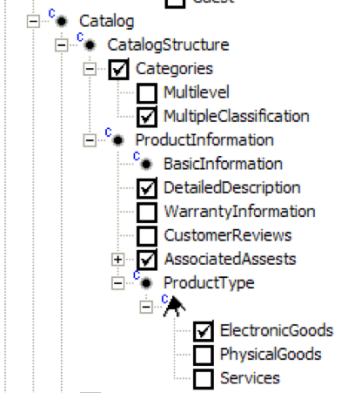


Automatic template instantiation
- Evaluation of presence conditions and meta–expressions
- Element removal
- Post–processing

Template instance

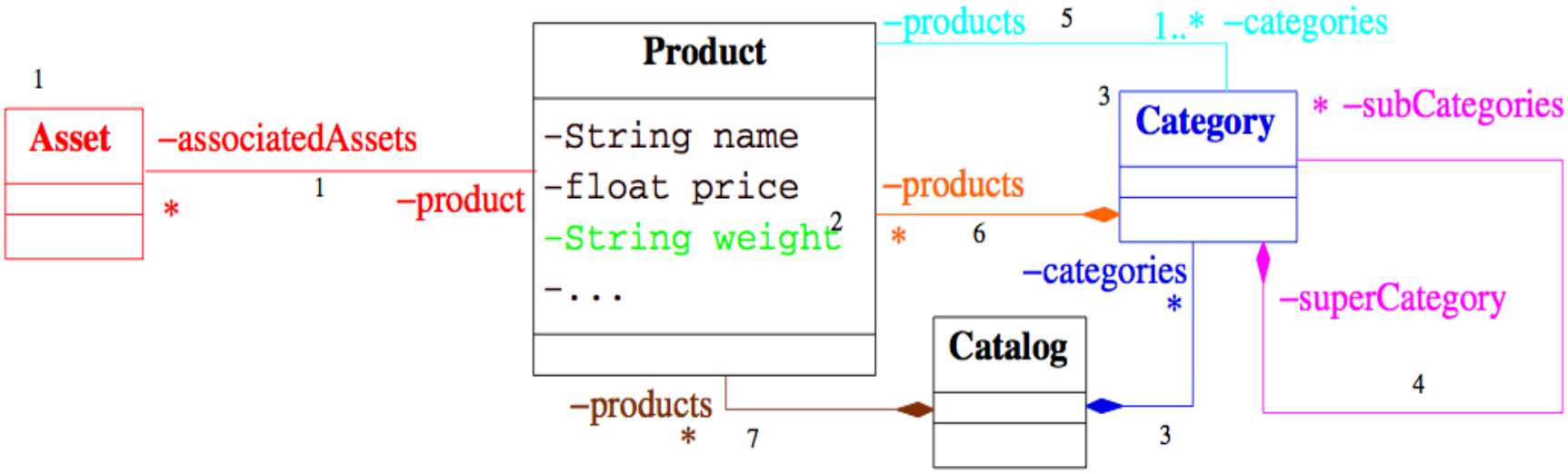
Product Model

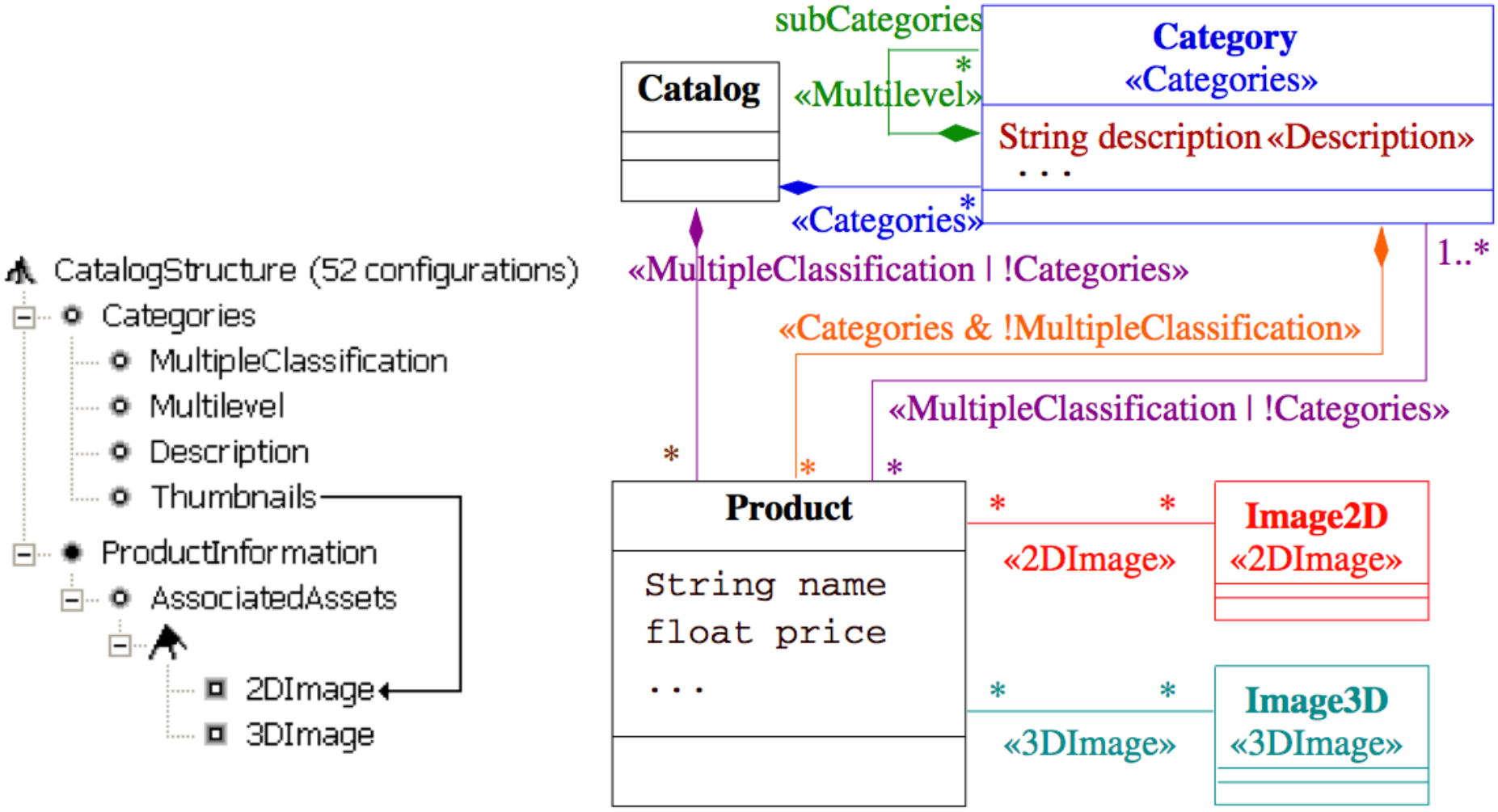
[Czarnecki et al. (GPCE'05)]

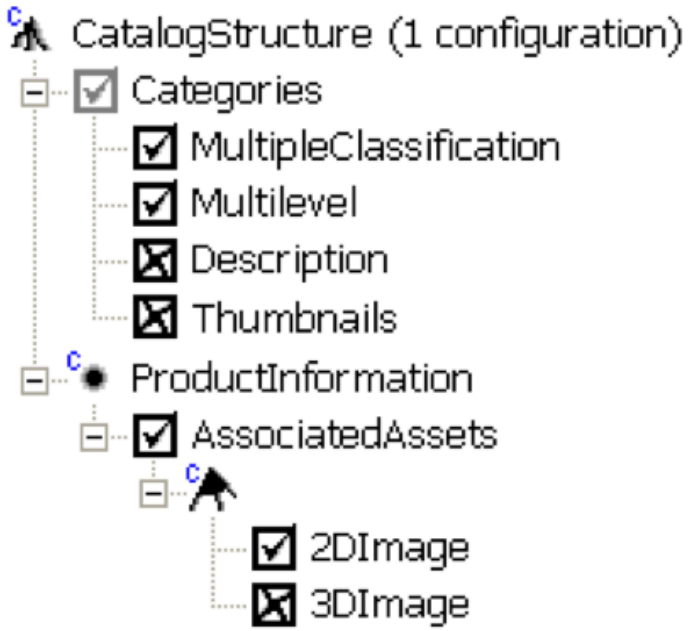


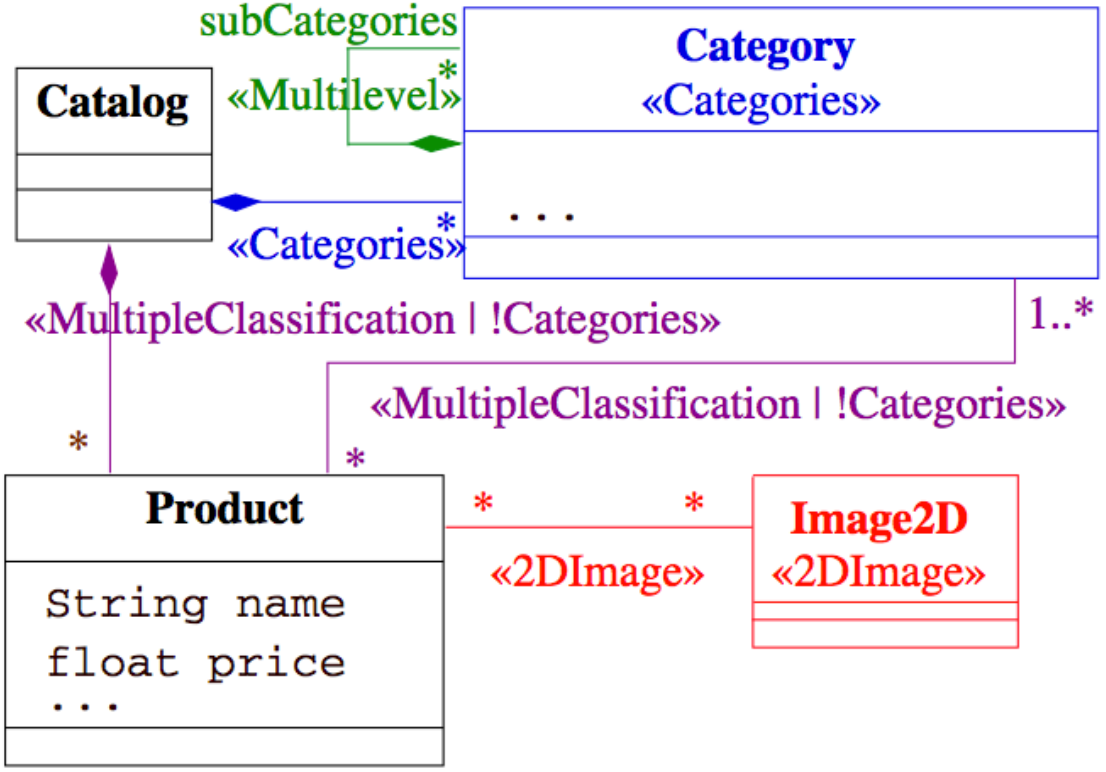
Presence conditions:

true		MultiLevel	4
AssociatedAssets	1	MultipleClassification	5
PhysicalGoods	2	Categories & !MultipleClassification	6
Categories	3	MultipleClassification | !Categories	7

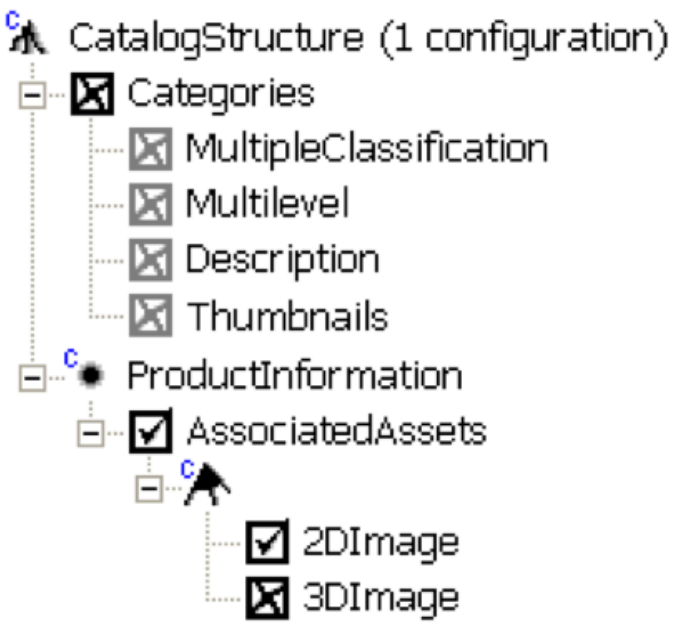


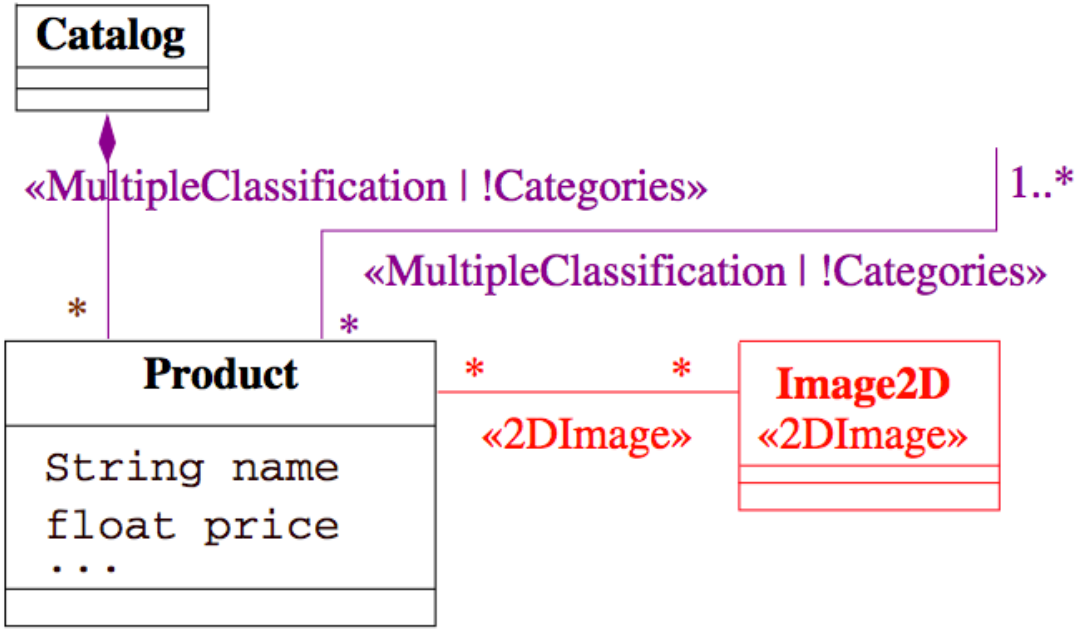

subCategories
Category
«Categories»
Catalog
«Multilevel»
String description «Description»
. . .
«Categories»
«MultipleClassification | !Categories»
1..*
«Categories & !MultipleClassification»
«MultipleClassification | !Categories»
Product
String name
float price
. . .
«2DImage»
Image2D
«2DImage»
«3DImage»
Image3D
«3DImage»
CatalogStructure (52 configurations)
Categories
MultipleClassification
Multilevel
Description
Thumbnails
ProductInformation
AssociatedAssets
2DImage
3DImage


CatalogStructure (1 configuration)
Categories
MultipleClassification
Multilevel
Description
Thumbnails
ProductInformation
AssociatedAssets
2DImage
3DImage

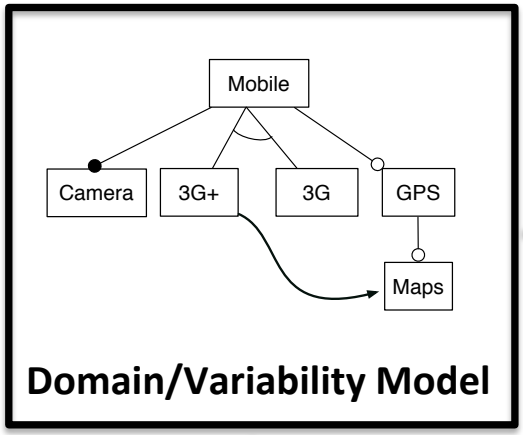

subCategories
Catalog
«Multilevel»
Category
«Categories»
...
«Categories»
«MultipleClassification | !Categories»
1..*
«MultipleClassification | !Categories»
Product
String name
float price
...
«2DImage»
Image2D
«2DImage»

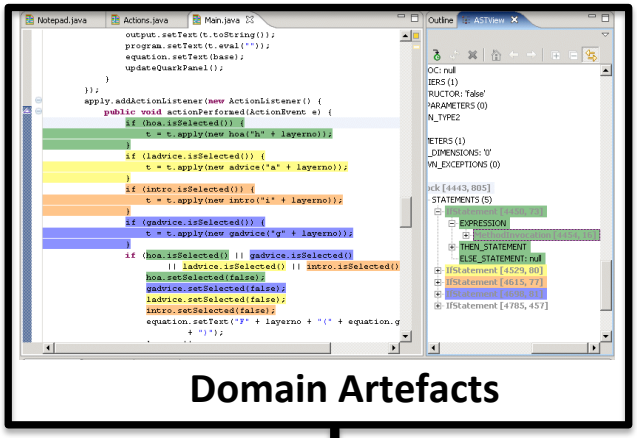
Ooops


CatalogStructure (1 configuration)
Categories
MultipleClassification
Multilevel
Description
Thumbnails
ProductInformation
AssociatedAssets
2DImage
3DImage

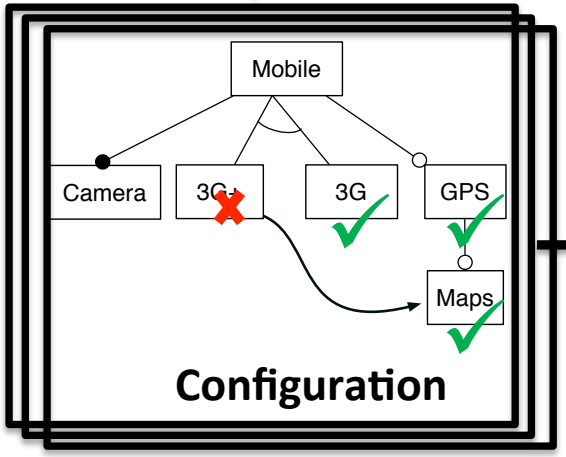

Catalog
«MultipleClassification | !Categories»
1..*
«MultipleClassification | !Categories»
*
*
Product
String name
float price
...
*
*
«2DImage»
Image2D
«2DImage»

Domain Engineering



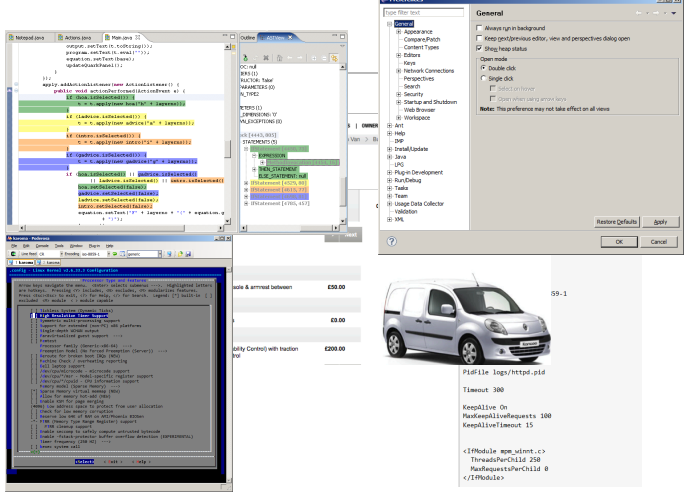


Application Engineering

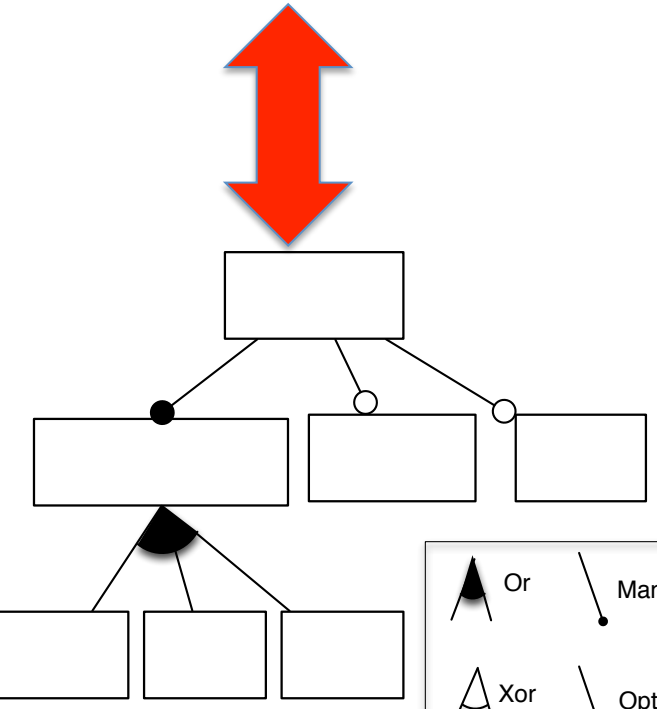


*« the investments required to develop the reusable artifacts during **domain engineering**, are outweighed by the benefits of deriving the individual products during **application engineering** »*

Jan Bosch et al. (2004)



variantes de code (e.g., Java ou C)
variantes d'interface utilisateur
variantes de séquence vidéo
variantes de langages
variantes de modèles
…



not, and, or, implies

Feature Models

Software Product Line and Variability Engineering

A revisit of your cursus

What is new?

Family vs single systems

Focus on reuse

Domain engineering

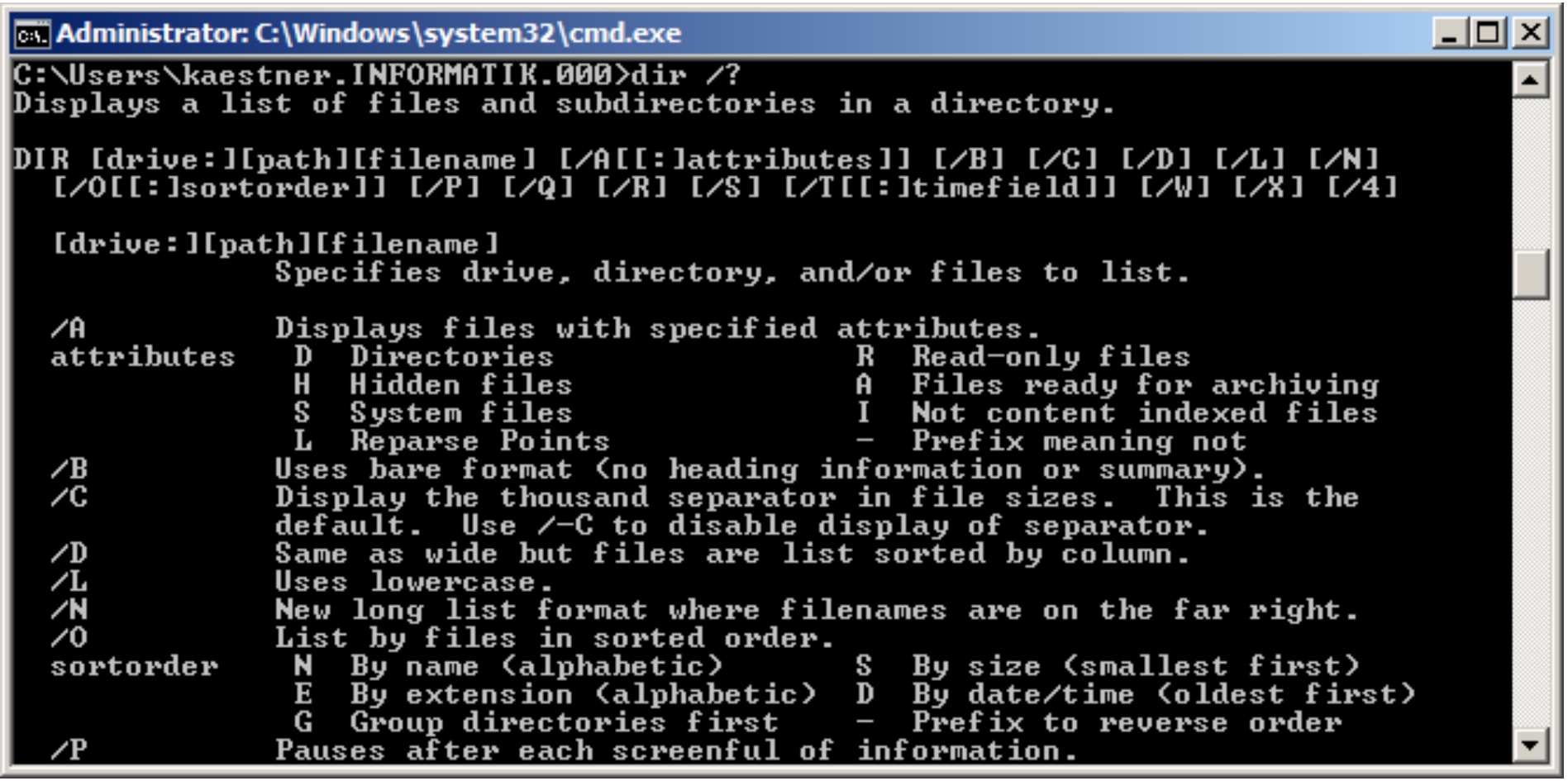
Factoring out commonality

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[[:]attributes]] [/B] [/C] [/D] [/L] [/N]
[/O[[:]sortorder]] [/P] [/Q] [/R] [/S] [/T[[:]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_c
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     }
```

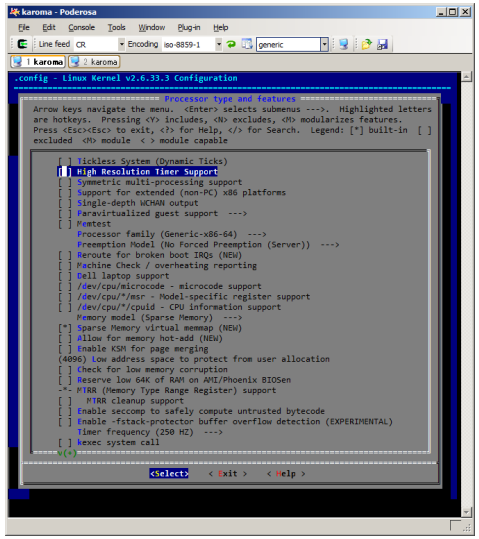
Global configuration

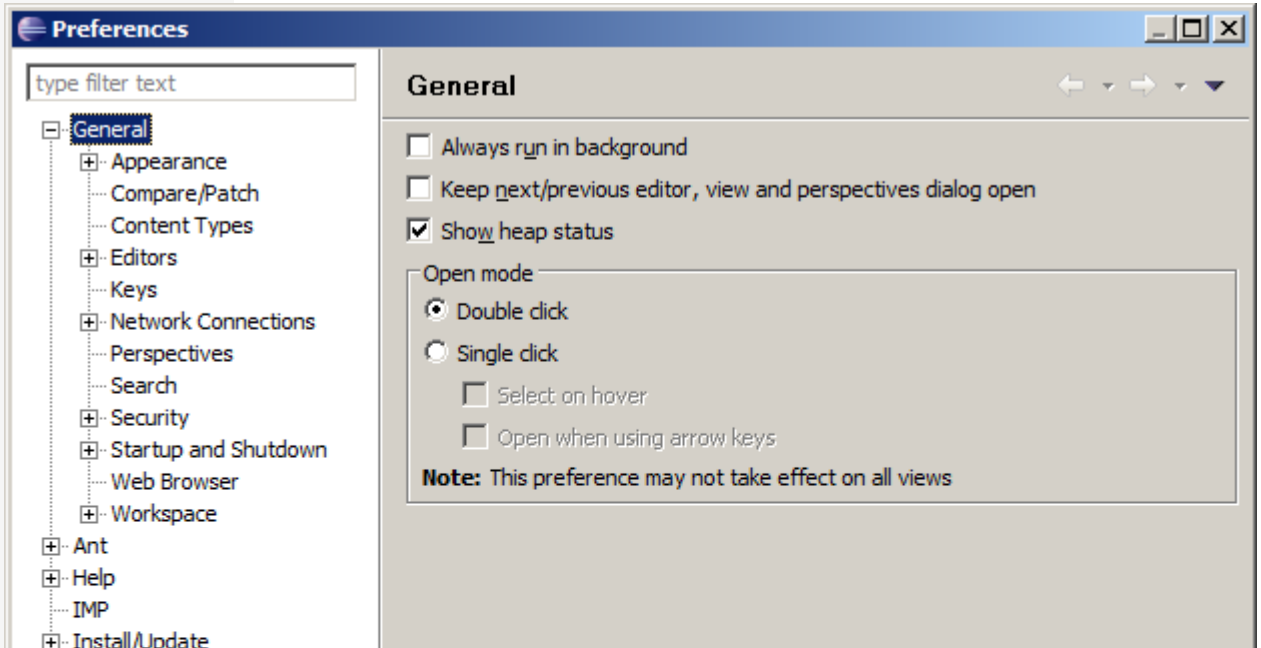
```
class Config {
  public static boolean isLogging = false;
  public static boolean isWindows = false;
  public static boolean isLinux = true;
}
class Main {
  public void foo() {
    if (isLogging)
      log(„running foo()“);
    if (isWindows)
      callWindowsMethod();
    else if (isLinux)
      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;
#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.intreqr, 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);
.......
```

cyberstormII.c

```
....
static void dma_dump_state(struct NCR_ESP *esp)
{
        ESPLOG(("esp%d: dma -- cond_reg<%02x>\n",
                esp->esp_id, ((struct cyberII_dma_registers *)
                                        (esp->dregs))->cond_reg));
        ESPLOG(("intreq:<%04x>, intena:<%04x>\n",
                custom.intreqr, custom.intenar));
}

static void dma_init_read(struct NCR_ESP *esp, __u32 addr, int
        length)
{
        struct cyberII_dma_registers *dregs =
                (struct cyberII_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

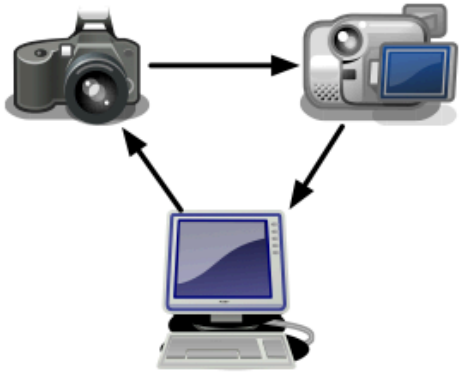
~ Cost of refactoring rises over time

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
(more details in the next course)

Template Method

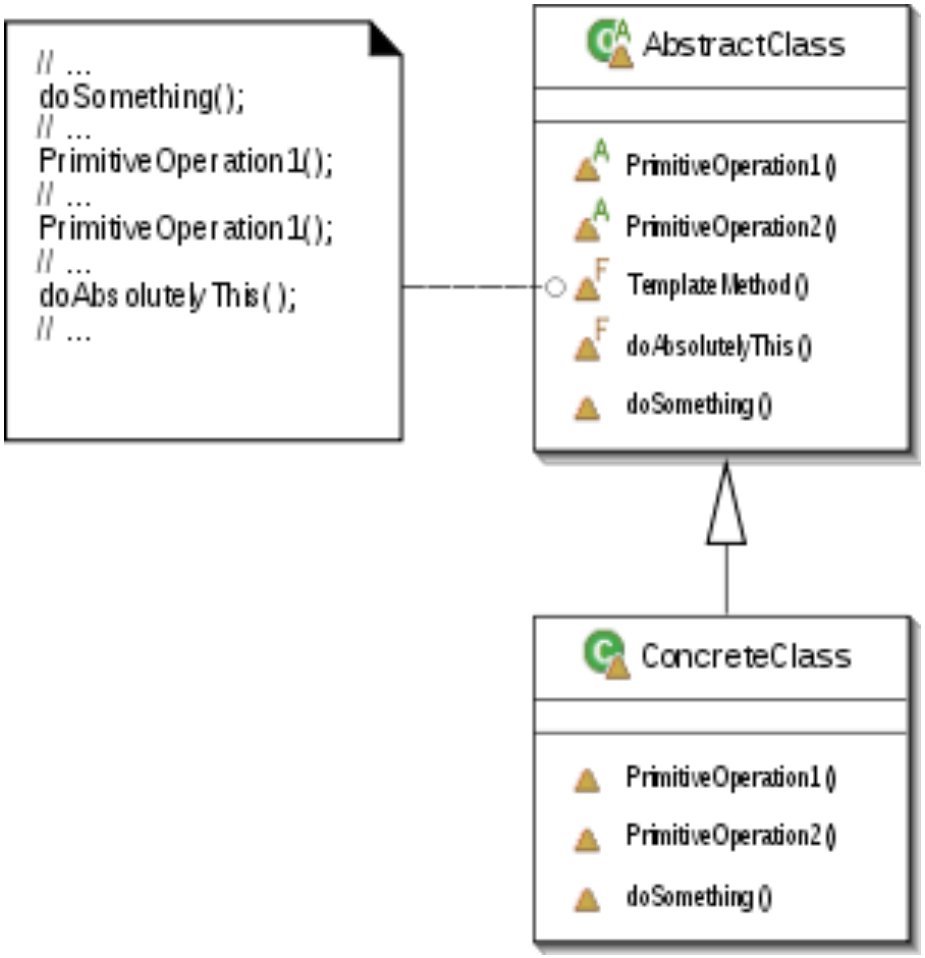
Factory

Strategy

Decorator

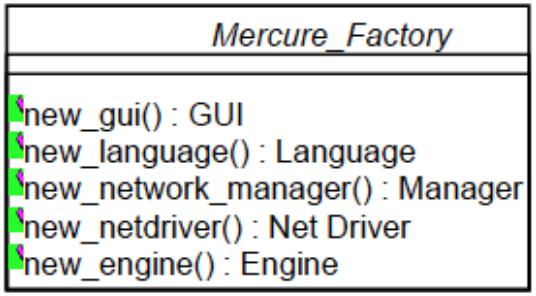
....

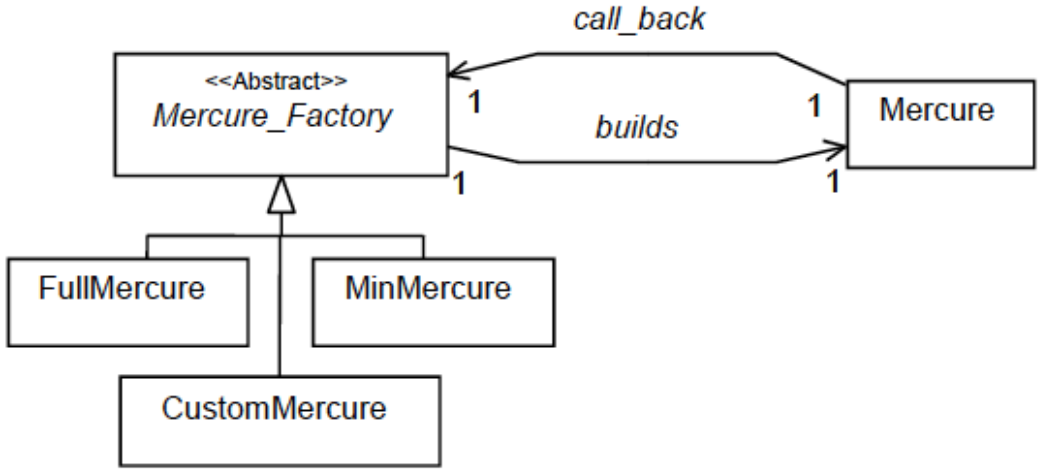
Template Method

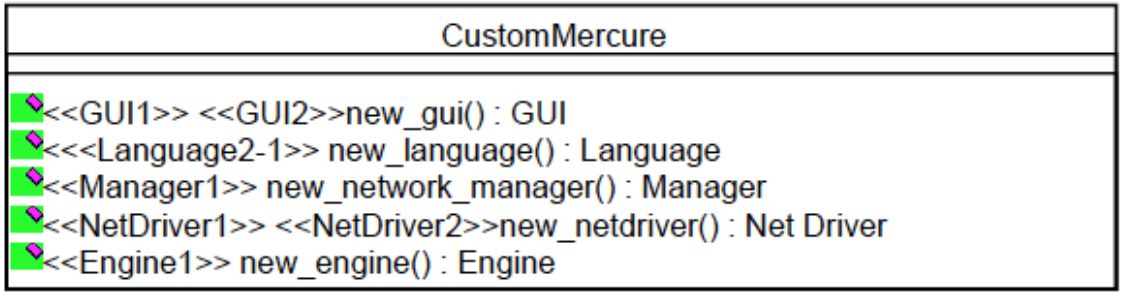


The decision model

- The Abstract Factory Design Pattern
  - [Gamma et al 95]







API

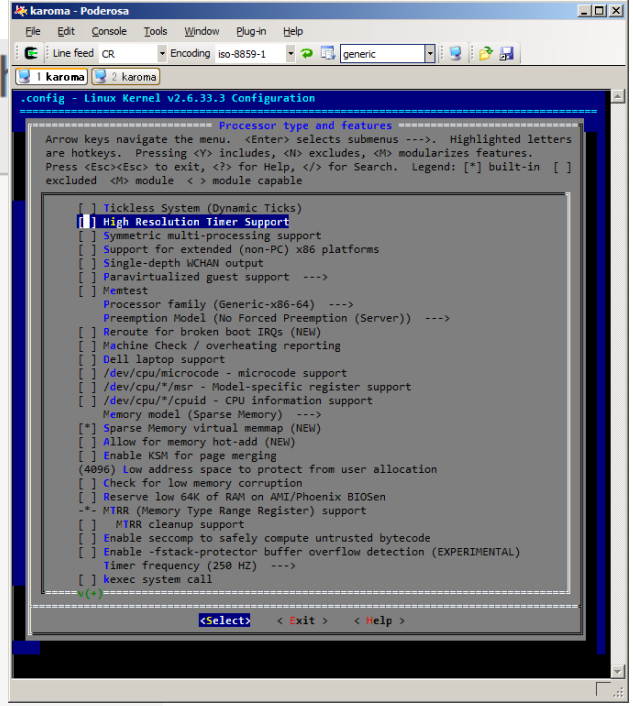
Framework

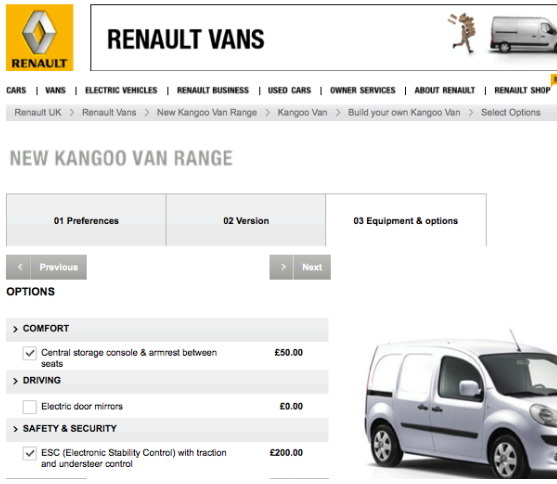
Plugin-based systems

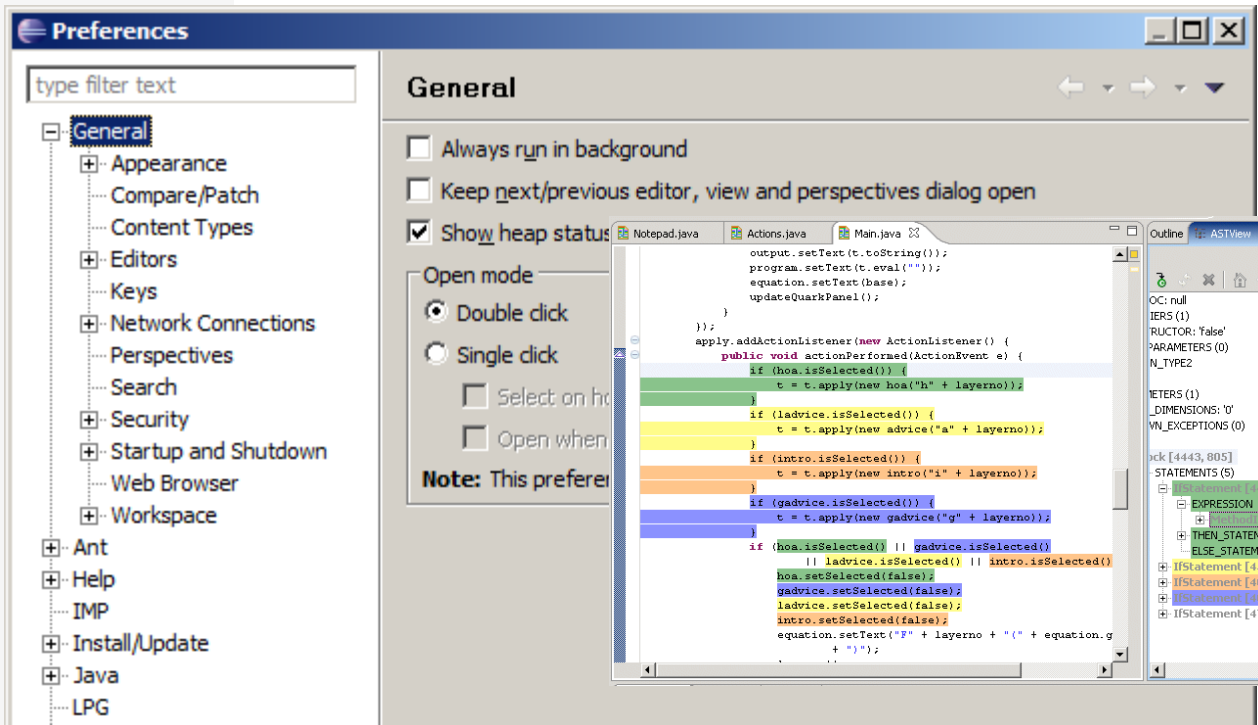
httpd.conf -- win32 Apach

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>
```





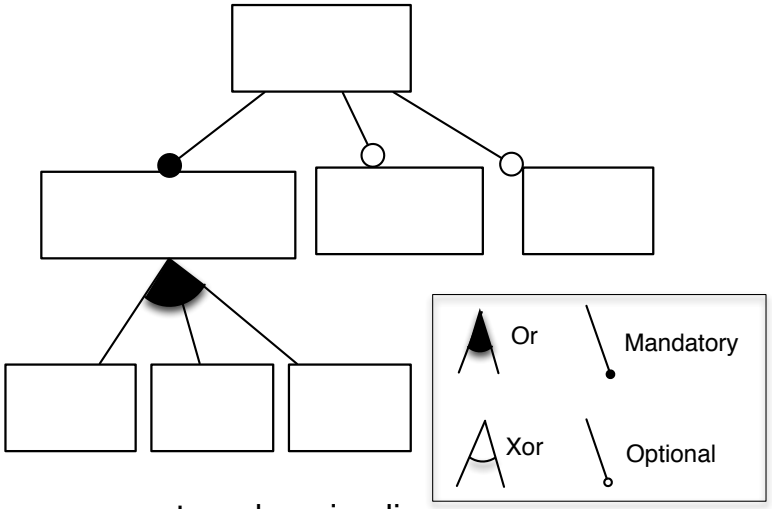


Variability in the Video Domain

What are the differences?

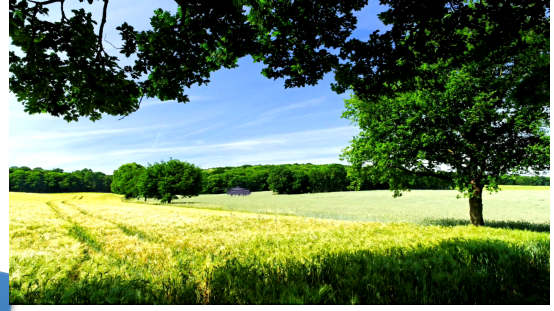
aka what is the variability of a video?

We synthesize video sequence variants with variability techniques



not, and, or, implies

Variability Models (feature models)



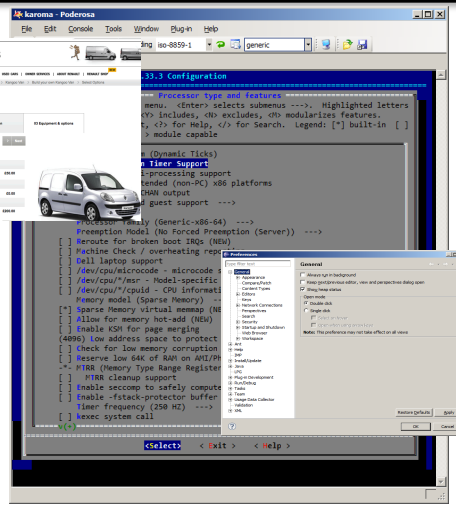
Why?

Industrial needs:
consumer and provider of video algorithms have severe difficulties to test their algorithms on different kinds of inputs.

Why?

Problem: collecting videos is a key economic problem.

Solution: hundreds of video sequences with different characteristics.

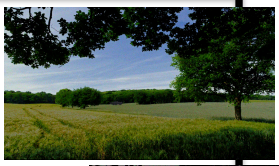


Feature Model

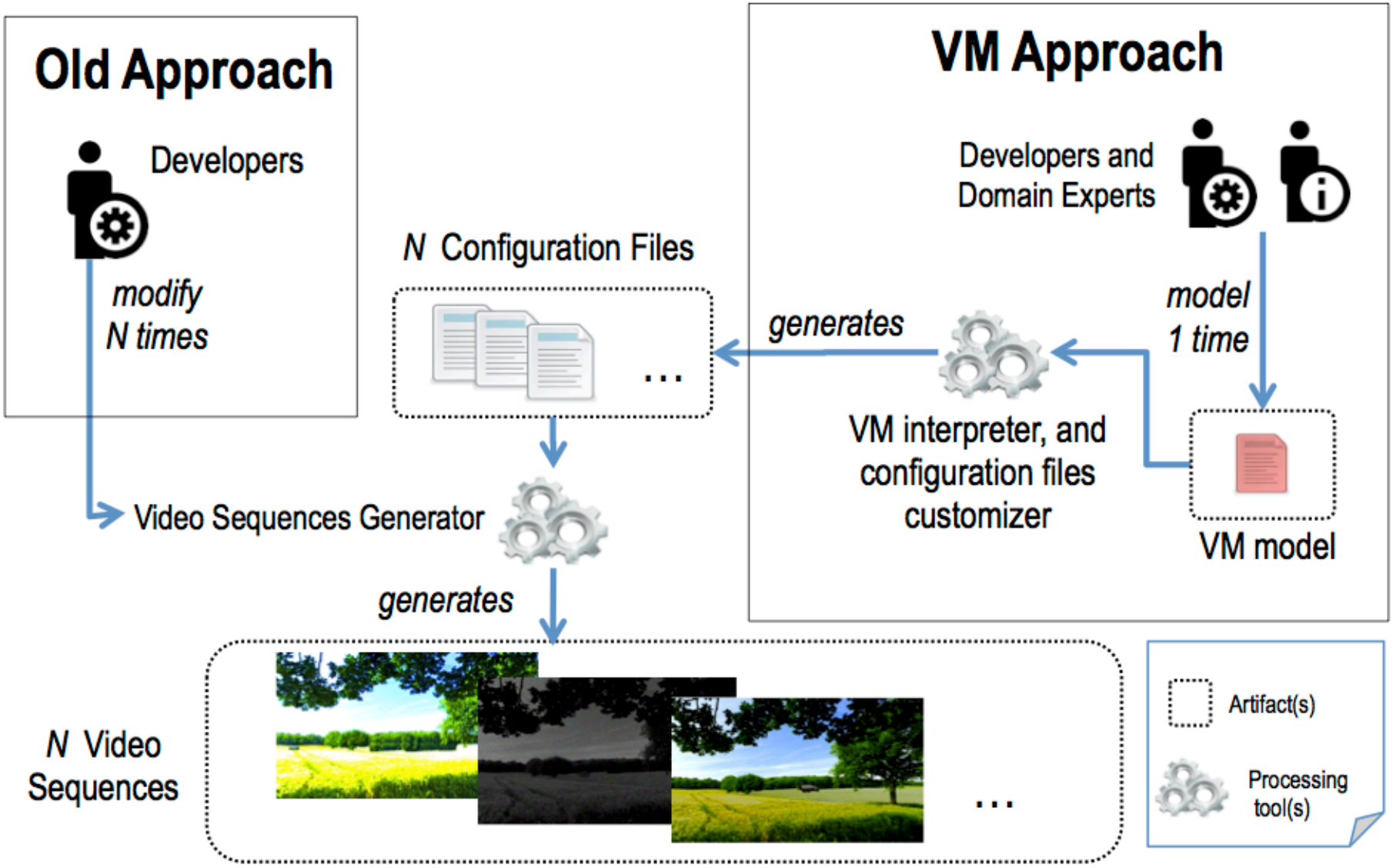
mapping

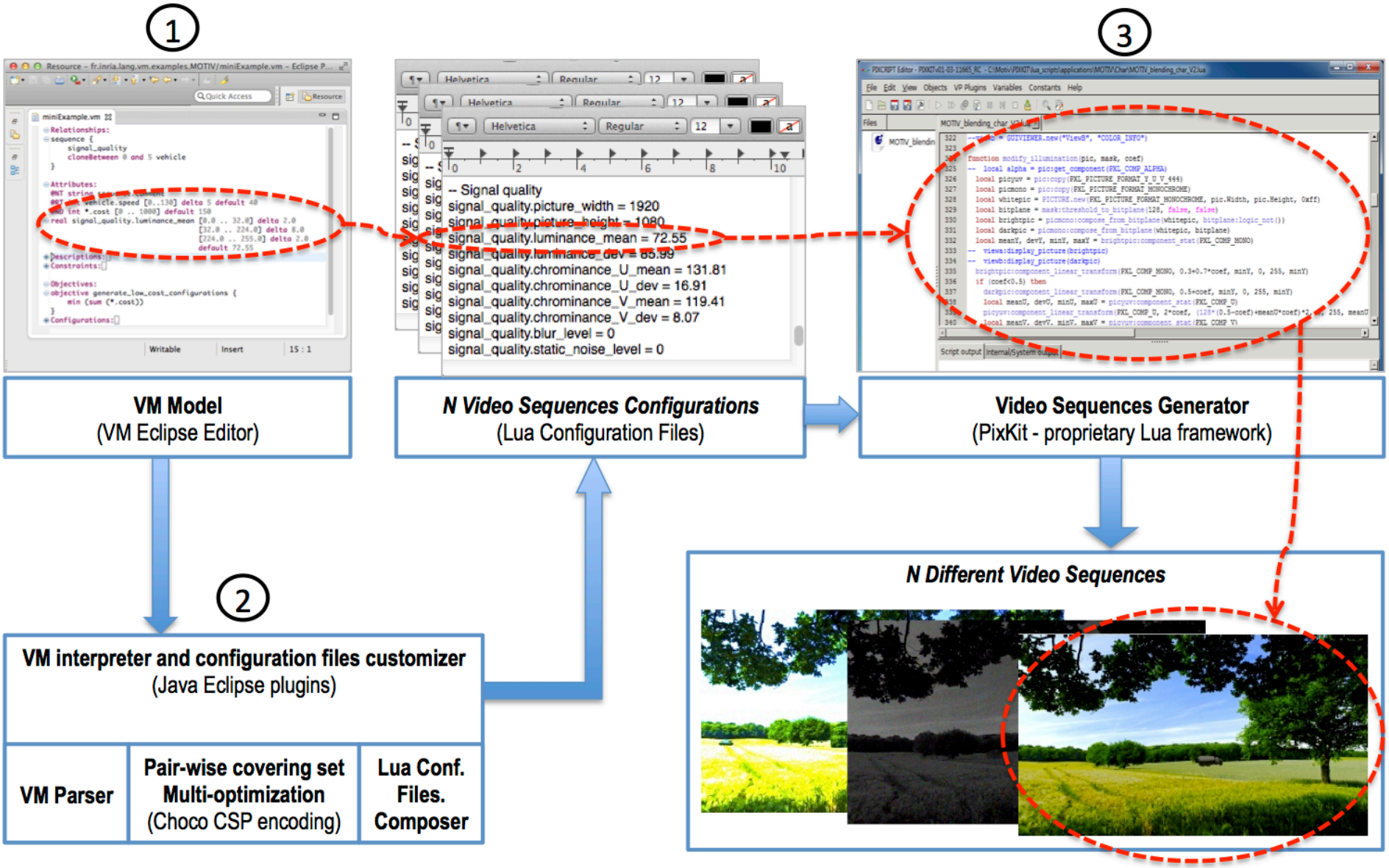
```
print("->Step9")
  if (CFG.distractors.close_moving_vegetation~=0) then
    windvect5, precwindvect5, newwindvect5 =
generate_wind_vector_field2(workwidth, workheight, 256, 1, 1, 35, picnum,
precwindvect5, newwindvect5)
    windvectmul =
windvect5:mul(24*CFG.distractors.close_moving_vegetation)
    globalvect = compose_vect(masque_feuilles_sombres, windvectmul, globalvect
    hfvx, hfvy =
windvect0:mul(6*CFG.distractors.close_moving_vegetation):to_matrix()
    hfvx = MATRIX.multerm(hfvx, invdepthmat)
    hfvy = MATRIX.multerm(hfvy, invdepthmat)
    lfvect = windvect2:resize_bilinear(windvect2.Width, windvect2.Height/16)
    lfvect = lfvect:resize_bilinear(windvect2.Width, windvect2.Height)
    lfvx, lfvy = lfvect:mul(-
12*CFG.distractors.close_moving_vegetation):to_matrix()
    lfvx = MATRIX.multerm(lfvx, depthmat)
    lfvy = MATRIX.multerm(lfvy, depthmat*0.1)
    windvectcomp = VECT2D.new_from_matrices(MATRIX.addterm(lfvx, hfvx),
MATRIX.addterm(lfvy, hfvy))
    globalvect = compose_vect(masque_ble, windvectcomp, globalvect)
    globalvect = compose_vect(masque_orge, windvectcomp, globalvect)
    globalvect = compose_vect(masque_ble_fond, windvectcomp, globalvect)
  end

print("->Step10")
```

```
-- Distractors
distractors.butterfly_level = 0.2 -- Floating point number from 0
(low level) to 1 (high level)
distractors.bird_level = 0.3 -- Floating point number from 0 (low
level) to 1 (high level)
distractors.far_moving_vegetation = 0.2 -- Floating point number
from 0 (low level) to 1 (high level)
distractors.close_moving_vegetation = 0.2 -- Floating point number
from 0 (low level) to 1 (high level)
distractors.light_reflection = 0 -- Floating point number from 0
(low level) to 1 (high level)
distractors.blinking_light = 0 -- Floating point number from 0
(low level) to 1 (high level)
```



Software Generator
(derivation engine)


Old Approach
Developers
modify
N times
Video Sequences Generator
N Configuration Files
...
generates
N Video
Sequences
...
VM Approach
Developers and
Domain Experts
model
1 time
generates
VM interpreter, and
configuration files
customizer
VM model
Artifact(s)
Processing
tool(s)


1
3
Relationships:
sequence {
signal_quality
cloneBetween 0 and 5 vehicle
}
Attributes:
real signal_quality.luminance_mean [0.0 .. 32.0] delta 2.0
[32.0 .. 224.0] delta 8.0
[224.0 .. 255.0] delta 2.0
default 72.55
Descriptions:
Constraints:
Objectives:
objective generate_low_cost_configurations {
min (sum (*.cost))
}
Configurations:
-- Signal quality
signal_quality.picture_width = 1920
signal_quality.picture_height = 1080
signal_quality.luminance_mean = 72.55
signal_quality.luminance_dev = 65.99
signal_quality.chrominance_U_mean = 131.81
signal_quality.chrominance_U_dev = 16.91
signal_quality.chrominance_V_mean = 119.41
signal_quality.chrominance_V_dev = 8.07
signal_quality.blur_level = 0
signal_quality.static_noise_level = 0
VM Model
(VM Eclipse Editor)
N Video Sequences Configurations
(Lua Configuration Files)
Video Sequences Generator
(PixKit - proprietary Lua framework)
2
VM interpreter and configuration files customizer
(Java Eclipse plugins)
VM Parser
Pair-wise covering set Multi-optimization
(Choco CSP encoding)
Lua Conf. Files. Composer
N Different Video Sequences

(configuration file)

```
-- Distractors
distractors.butterfly_level = 0.2 -- Floating point number from 0
(low level) to 1 (high level)
distractors.bird_level = 0.3 -- Floating point number from 0 (low
level) to 1 (high level)
distractors.far_moving_vegetation = 0.2 -- Floating point number
from 0 (low level) to 1 (high level)
distractors.close_moving_vegetation = 0.2 -- Floating point number
from 0 (low level) to 1 (high level)
distractors.light_reflection = 0 -- Floating point number from 0
(low level) to 1 (high level)
distractors.blinking_light = 0 -- Floating point number from 0
(low level) to 1 (high level)
```

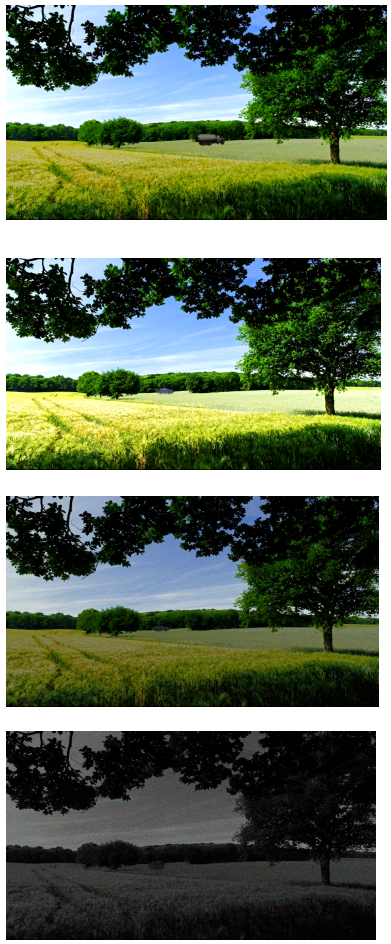
```
print("->Step9")
  if (CFG.distractors.close_moving_vegetation~=0) then
    windvect5, precwindvect5, newwindvect5 =
generate_wind_vector_field2(workwidth, workheight, 256, 1, 1, 35, picnum,
precwindvect5, newwindvect5)
    windvectmul =
windvect5:mul(24*CFG.distractors.close_moving_vegetation)
    globalvect = compose_vect(masque_feuilles_sombres, windvectmul, globalvect
    hfvx, hfvy =
windvect0:mul(6*CFG.distractors.close_moving_vegetation):to_matrix()
    hfvx = MATRIX.multerm(hfvx, invdepthmat)
    hfvy = MATRIX.multerm(hfvy, invdepthmat)
    lfvect = windvect2:resize_bilinear(windvect2.Width, windvect2.Height/16)
    lfvect = lfvect:resize_bilinear(windvect2.Width, windvect2.Height)
    lfvx, lfvy = lfvect:mul(-
12*CFG.distractors.close_moving_vegetation):to_matrix()
    lfvx = MATRIX.multerm(lfvx, depthmat)
    lfvy = MATRIX.multerm(lfvy, depthmat*0.1)
    windvectcomp = VECT2D.new_from_matrices(MATRIX.addterm(lfvx, hfvx),
MATRIX.addterm(lfvy, hfvy))
    globalvect = compose_vect(masque_ble, windvectcomp, globalvect)
    globalvect = compose_vect(masque_orge, windvectcomp, globalvect)
    globalvect = compose_vect(masque_ble_fond, windvectcomp, globalvect)
  end

print("->Step10")
```

(Lua code)

Variability and Machine Learning

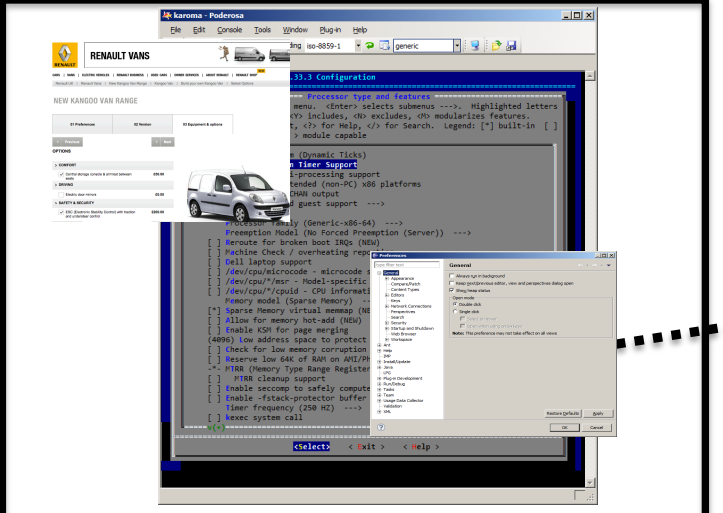
(contact me for internships)

Defects detection
Benchmarking
Incremental design
Performance prediction



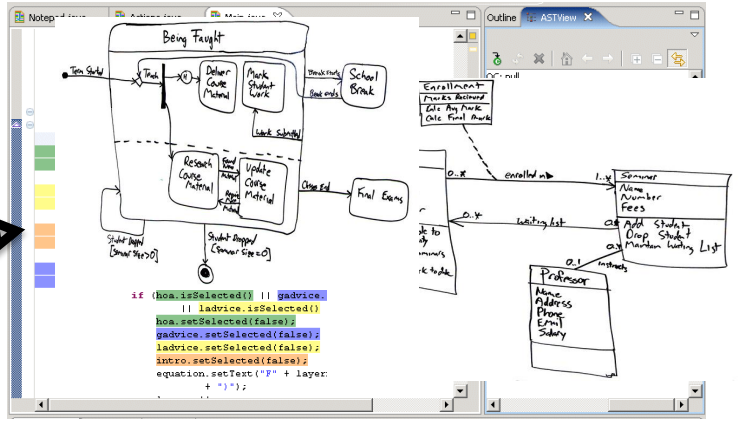
Algorithm 1	Algorithm 2	Algorithm 3
0.63	0.93	0.82
0.81	0.92	0.81
0.43	0.3	0.8
0.39	0.03	0.01

Feature Models

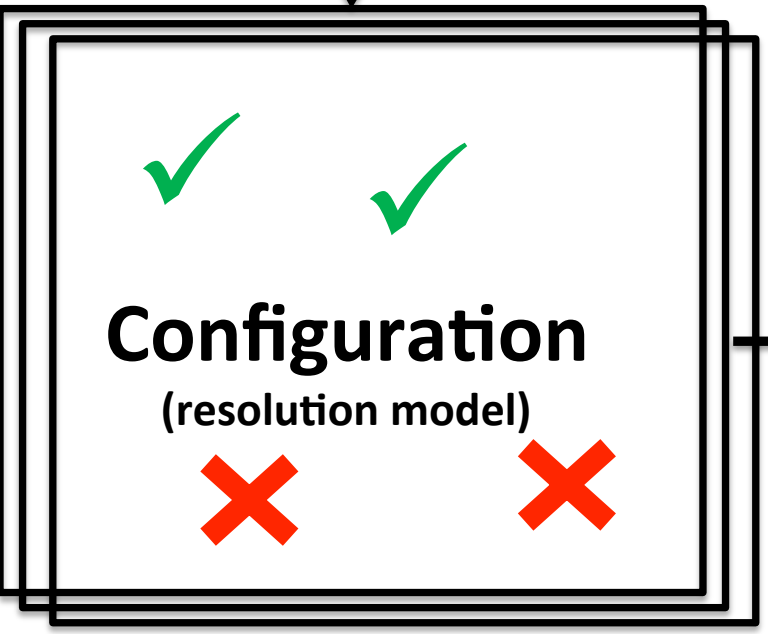


Feature Model

Variability Realization Model (VRM)



Base Artefacts (e.g., models)



Configuration

(resolution model)

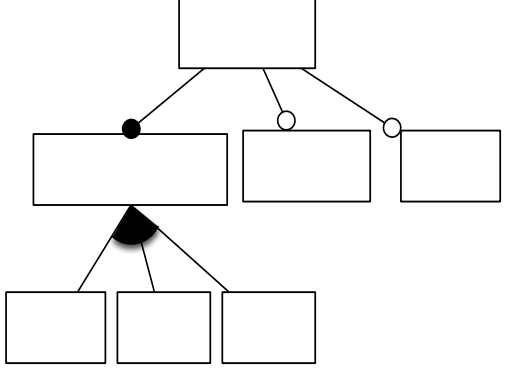
Software Generator

(derivation engine)


Unused flexibility

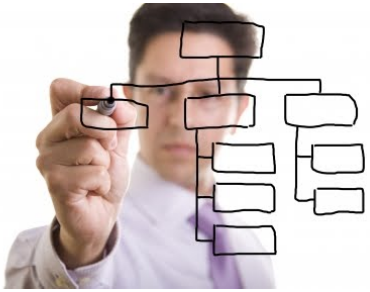

Illegal variant

Feature Model

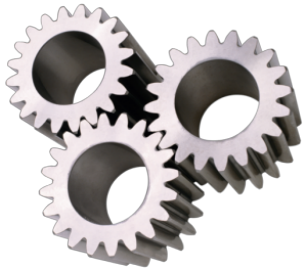


not, and, or, implies

Communicative



Analytic



Generative

Audi
Vorsprung durch Technik

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

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

1 Modèle | 2 Moteur | 3 Extérieur | 4 Intérieur | 5 Option | 6 Votre Audi

Audi
Vorsprung durch Technik

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

excludes

☑ 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

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

Audi
Vorsprung durch Technik

A5 Sportback
3.0 TDI quattro S tronic

Prix total
54.460,15 EUR

Prix de base	50.570,00 EUR
Equipements optionnels	3.890,15 EUR

- Informations détaillées
- Entrez l'Audi Code
- Nouvelle configuration

[+] Plein écran / Dimensions

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 **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		181,50 EUR
○ Pack Intenso Plus		3.100,00 EUR

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

2.934,25 EUR

● Sans appareil de navigation — Série

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

Un aperç des équipements:

Mode expert

Réinitialiser la sélection

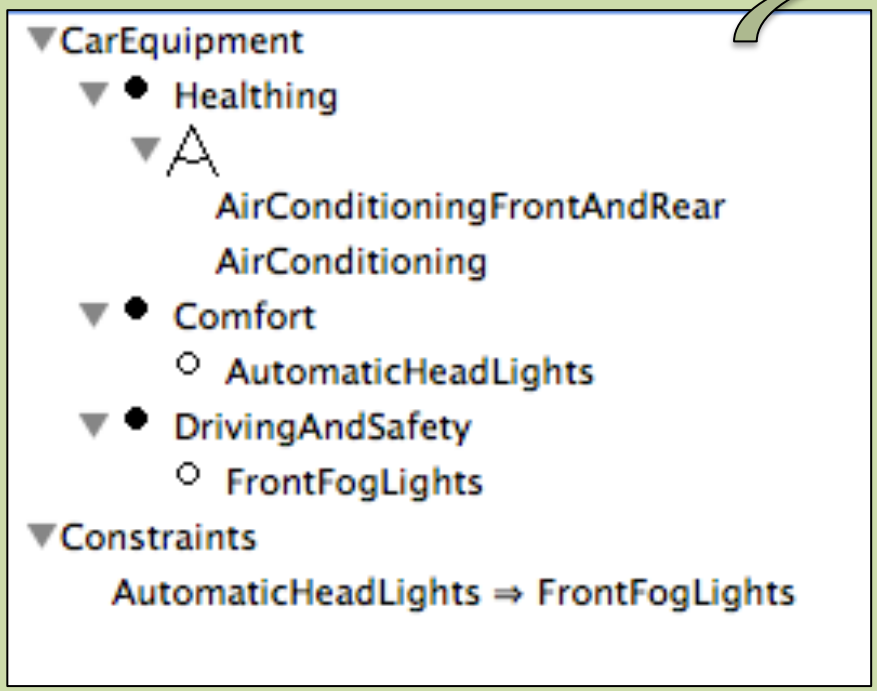
1 Modèle | 2 Moteur | 3 Extérieur | 4 Intérieur | 5 Option | 6 Votre Audi

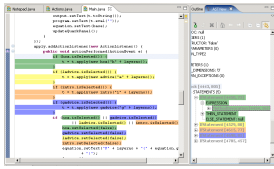
Francais

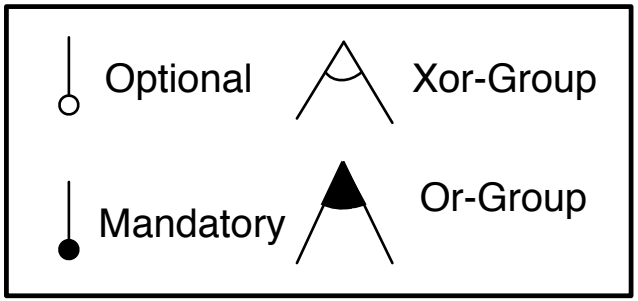
Suivant

Feature Models

(defacto standard for modeling variability)







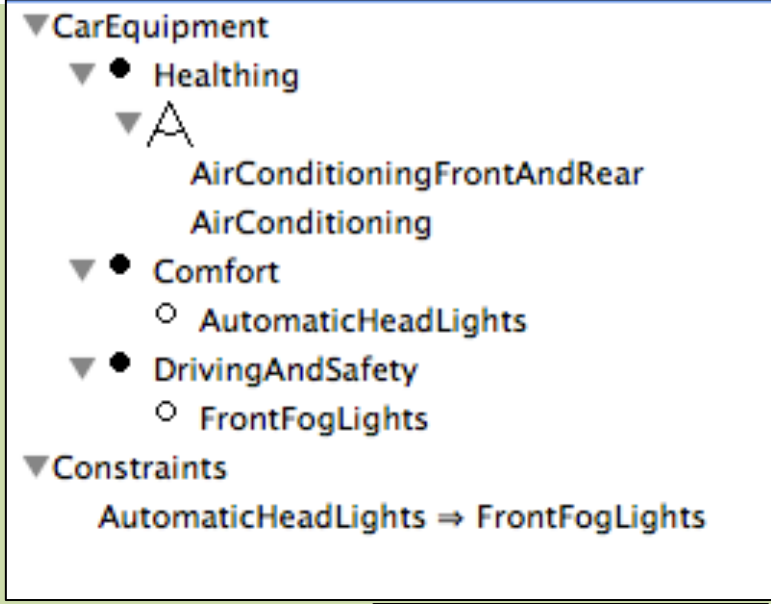
Hierarchy: rooted tree

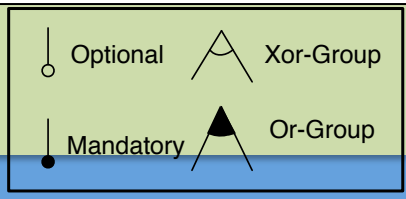
Variability:

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

SPLC12Scripts Model | SPLC12Scripts.config

- CarEquipment (valid, 1 possible configurations)
  - Healthing
    - AirConditioningFrontAndRear
    - AirConditioning
  - Comfort
    - AutomaticHeadLights
  - DrivingAndSafety
    - FrontFogLights

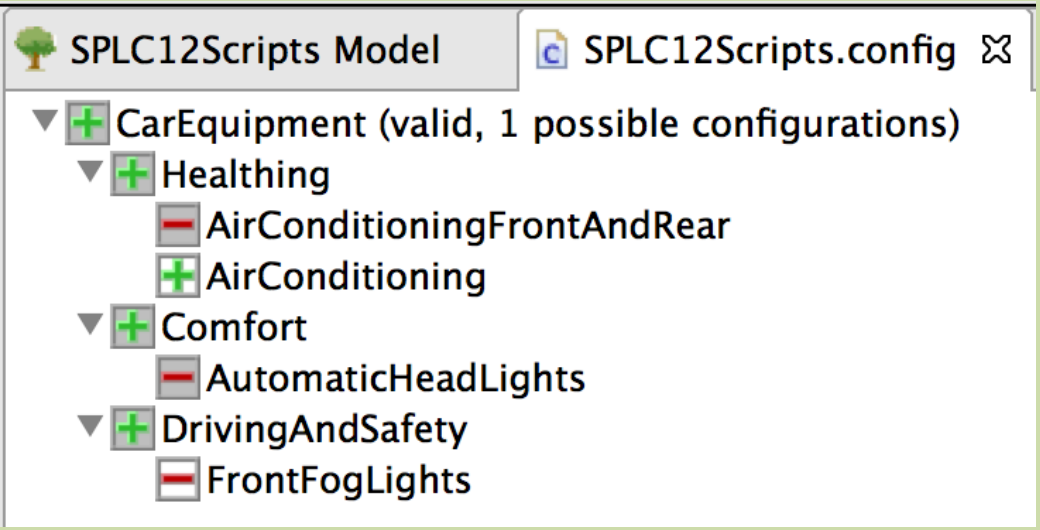


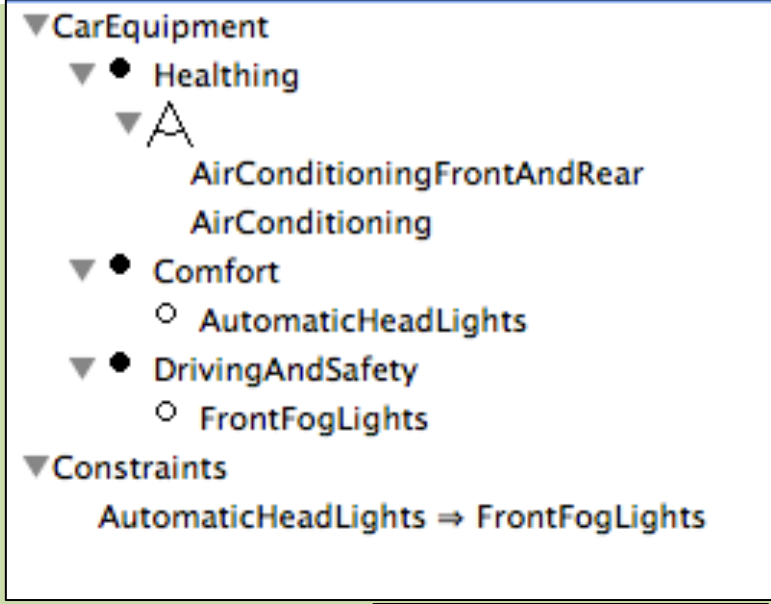


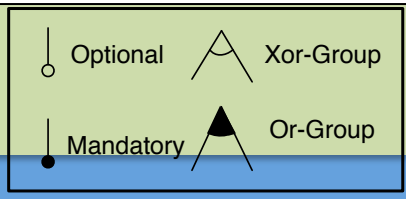
Hierarchy + Variability
=
set of valid configurations

configuration = set of features selected

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



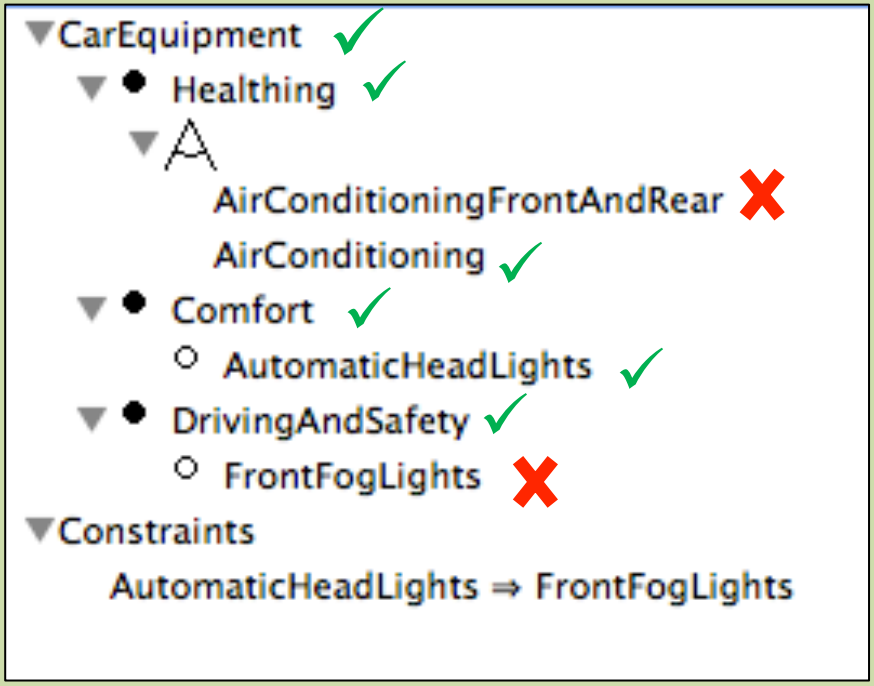


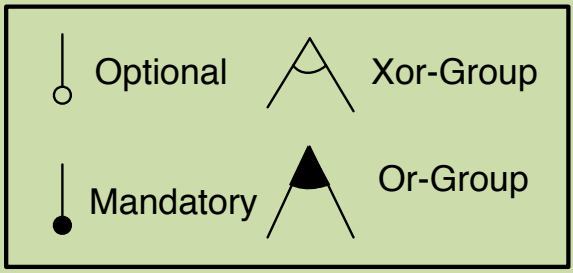


Hierarchy + Variability
=
set of valid configurations

configuration = set of features selected

{CarEquipment, Comfort, DrivingAndSafety, Healthing, AirConditioning}

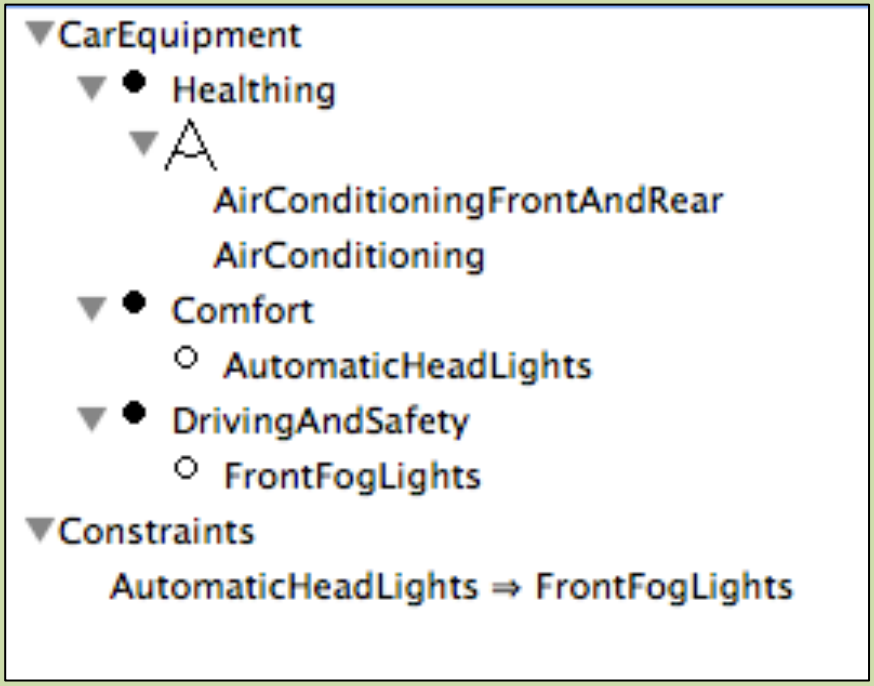


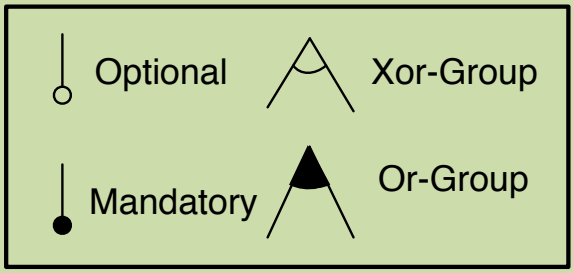


Hierarchy + Variability
=
set of valid configurations

configuration = set of features selected

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



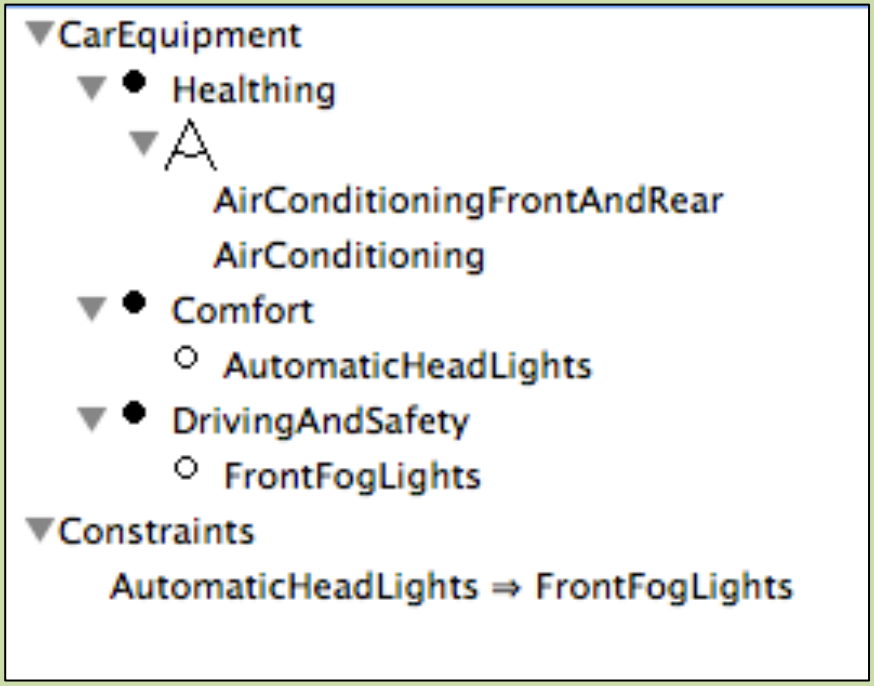


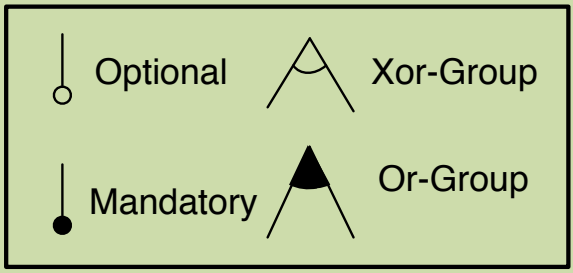
Hierarchy + Variability
=
set of valid configurations

{CarEquipment, Comfort, DrivingAndSafety, Healthing}

X

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

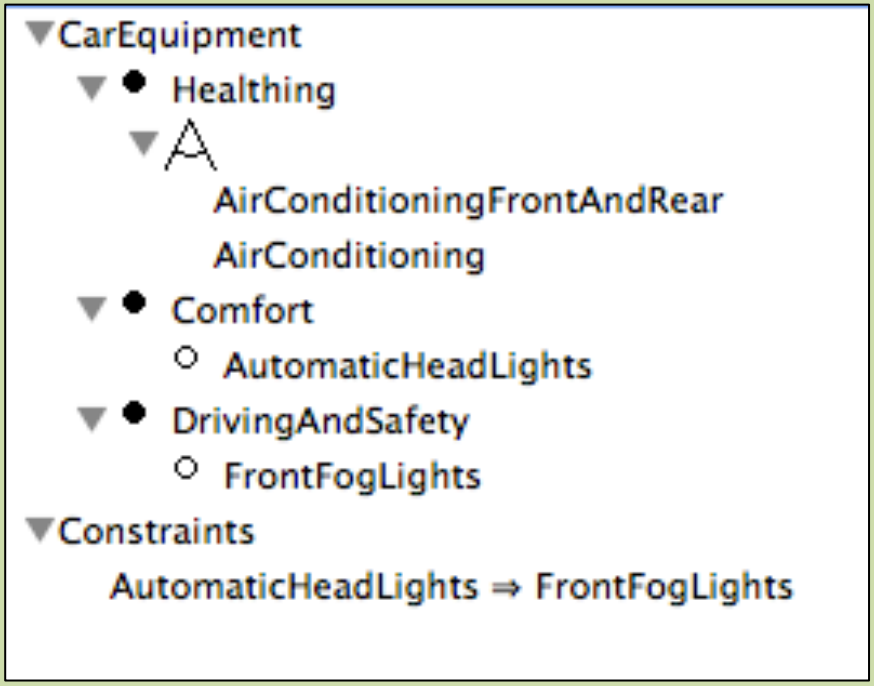


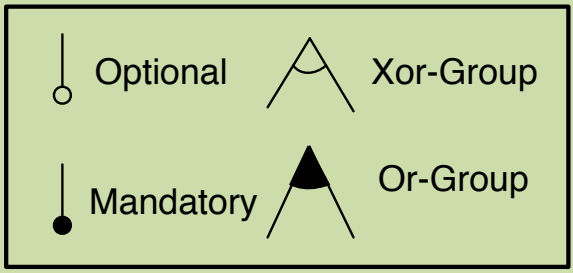


Hierarchy + Variability
=
set of valid configurations

Configuration set (from a basic feature model of car)

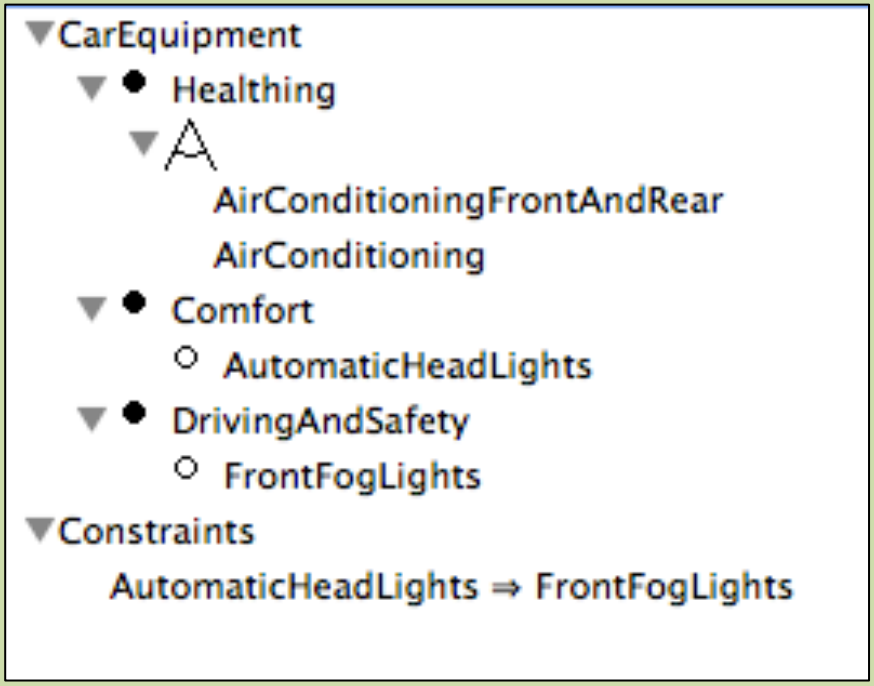
	CarEquipment	Comfort	DrivingAndSafety	Healting	AirConditioning	FrontFogLights	AutomaticHeadLights	AirConditioningFrontAndRear
Car2	yes	yes	yes	yes	yes	yes	yes	no
Car6	yes	yes	yes	yes	no	yes	no	yes
Car1	yes	yes	yes	yes	yes	yes	no	no
Car4	yes	yes	yes	yes	no	no	no	yes
Car5	yes	yes	yes	yes	yes	no	no	no
Car3	yes	yes	yes	yes	no	yes	yes	yes

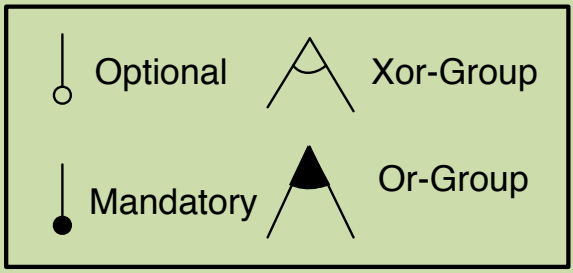




Hierarchy + Variability
=
set of valid configurations

Product	CarEquipment	Comfort	DrivingAndSafety	Healting	AirConditioning	FrontFogLights	AutomaticHeadLights	AirConditioningFrontAndRear
Find	Yes / No	Yes / No	Yes / No	Yes / No	Yes / No	Yes / No	Yes / No	Yes / No
Car1	yes	yes	yes	yes	yes	yes	no	no
Car2	yes	yes	yes	yes	yes	yes	yes	no
Car3	yes	yes	yes	yes	no	yes	yes	yes
Car4	yes	yes	yes	yes	no	no	no	yes
Car5	yes	yes	yes	yes	yes	no	no	no
Car6	yes	yes	yes	yes	no	yes	no	yes



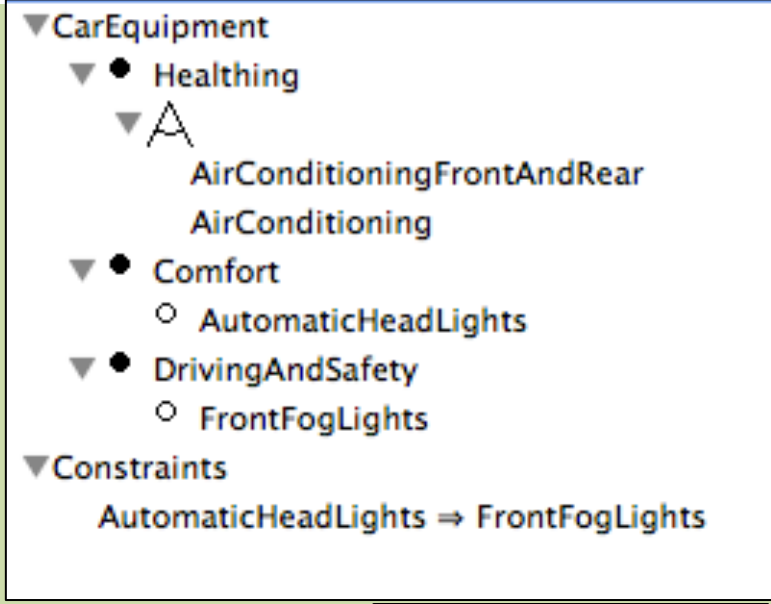


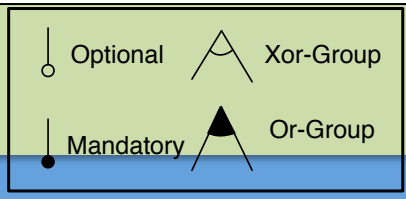
Hierarchy + Variability
=
set of valid configurations

Product					AirConditioning	FrontFogLights	AutomaticHeadLights	AirConditioningFrontAndRear
Find					Yes No	Yes No	Yes No	Yes No
Car1					yes	yes	no	no
Car2					yes	yes	yes	no
Car3					no	yes	yes	yes
Car4					no	no	no	yes
Car5					yes	no	no	no
Car6					no	yes	no	yes

SPLC12Scripts Model | SPLC12Scripts.config

- CarEquipment (valid, 1 possible configurations)
  - Healthing
    - AirConditioningFrontAndRear
    - AirConditioning
  - Comfort
    - AutomaticHeadLights
  - DrivingAndSafety
    - FrontFogLights



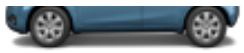


Hierarchy + Variability
=
set of valid configurations

configuration = set of features selected

{CarEquipment, Comfort, DrivingAndSafety, Healthing, AirConditioning}

Product					AirConditioning	FrontFogLights	AutomaticHeadLights	AirConditioningFrontAndRear
Find					Yes ☑ No ☐	Yes ☐ No ☑	Yes ☐ No ☑	Yes ☐ No ☑
Car5					yes	no	no	no



FAMILIAR

(FeAture Model scrIpt Language for manIpulation and Automatic Reasoning)

http://familiar-project.github.com/

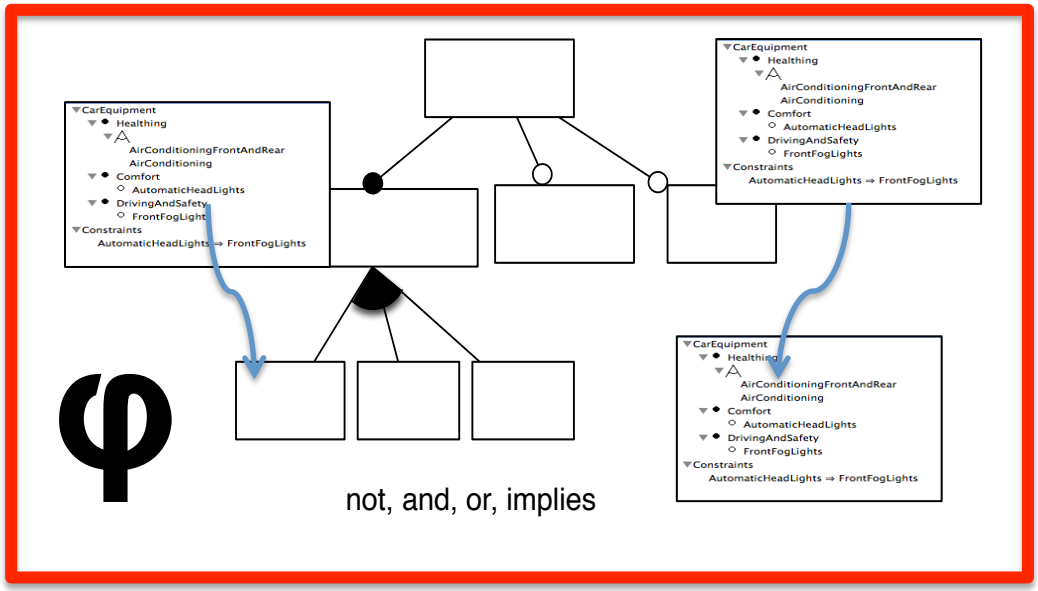
COMMON VARIABILITY LANGUAGE

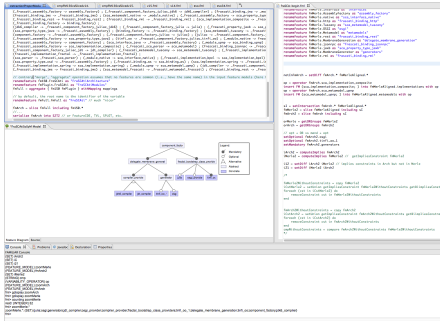
S.P.L.O.T. Software Product Lines Online Tools

IDE Feature

TVL

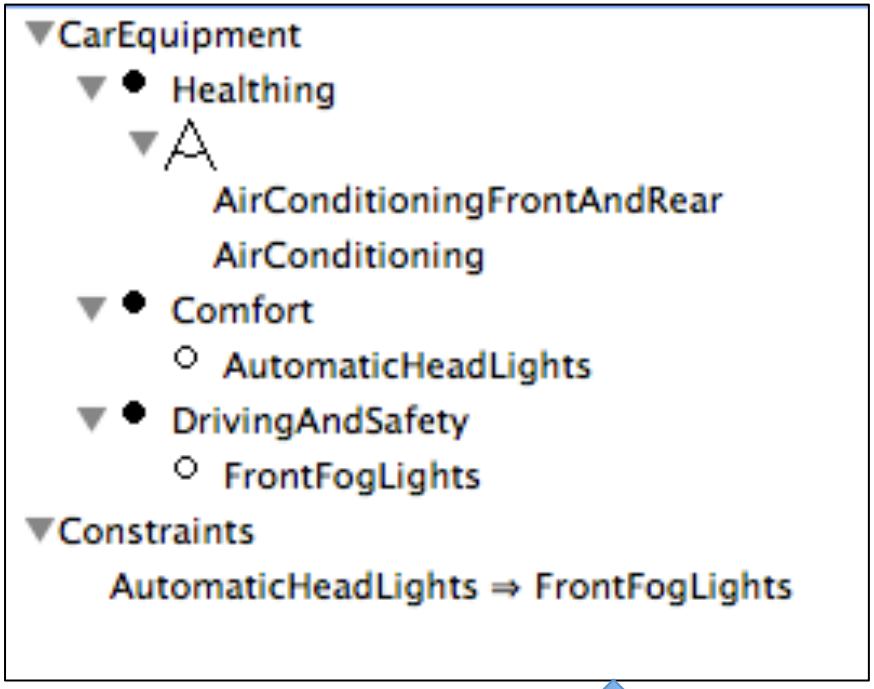
DIMACS





importing, exporting, composing, decomposing, editing, configuring, reverse engineering, computing "diffs", refactoring, testing, and reasoning about (multiple) variability models

Mathieu Acher, Philippe Collet, Philippe Lahire, Robert B. France « A Domain-Specific Language for Large-Scale Management of Feature Models » Science of Computer Programming (SCP), 2013



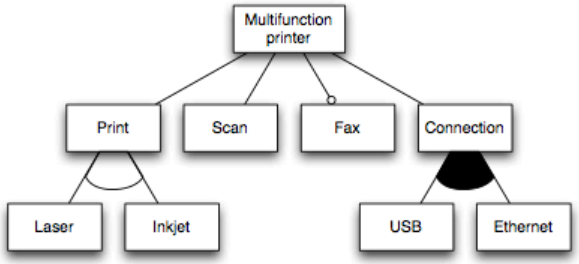
(Boolean) Feature Models

(Boolean) Formula φ

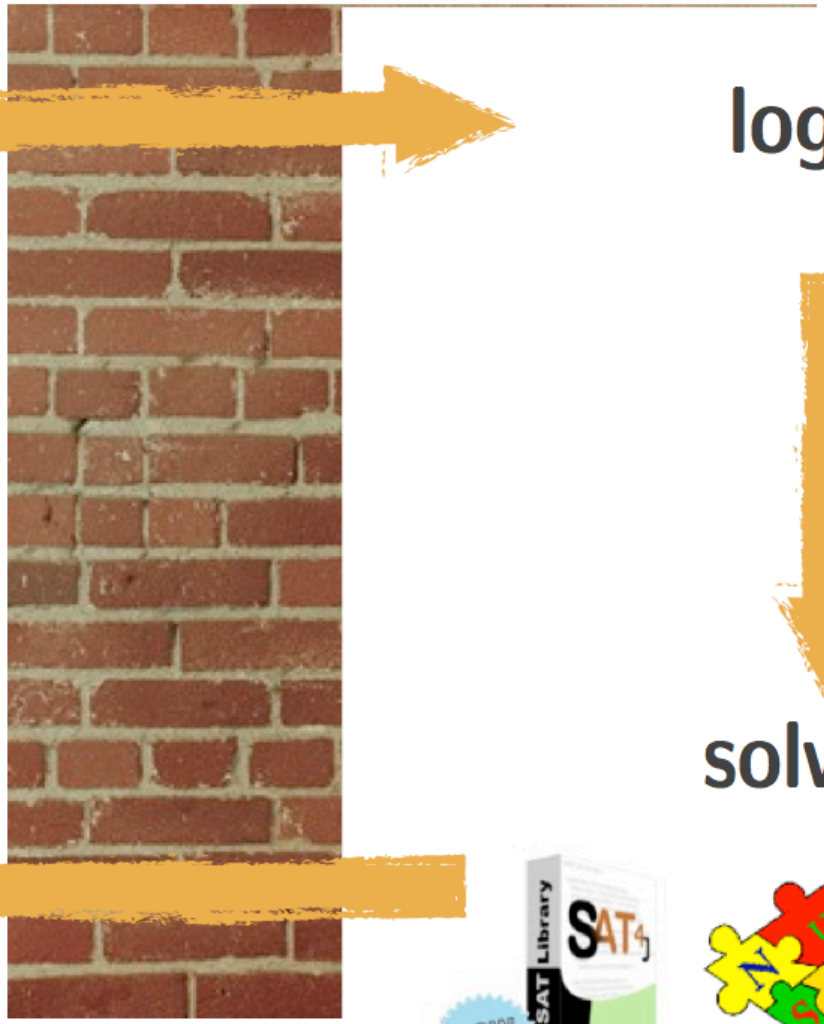
Product	CarEquipment	Comfort	DrivingAndSafety	Healting	AirConditioning	FrontFogLights	AutomaticHeadLights	AirConditioningFrontAndRear
Find	Yes / No	Yes / No	Yes / No	Yes / No	Yes / No	Yes / No	Yes / No	Yes / No
Car1	yes	yes	yes	yes	yes	yes	no	no
Car2	yes	yes	yes	yes	yes	yes	yes	no
Car3	yes	yes	yes	yes	no	yes	yes	yes
Car4	yes	yes	yes	yes	no	no	no	yes
Car5	yes	yes	yes	yes	yes	no	no	no
Car6	yes	yes	yes	yes	no	yes	no	yes

(Boolean) Product Comparison Matrix

Typical implementations



result

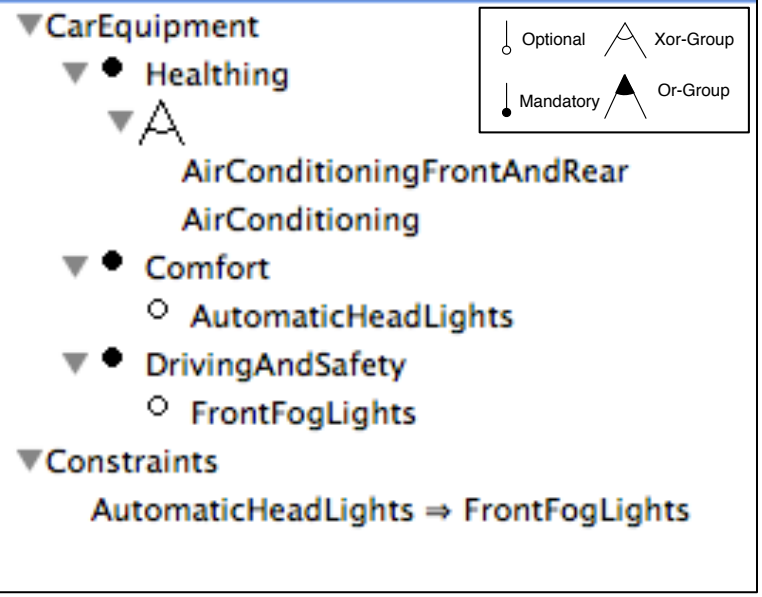


logics

solvers

Learning Feature Models with

(a.k.a implementing the introductory example)



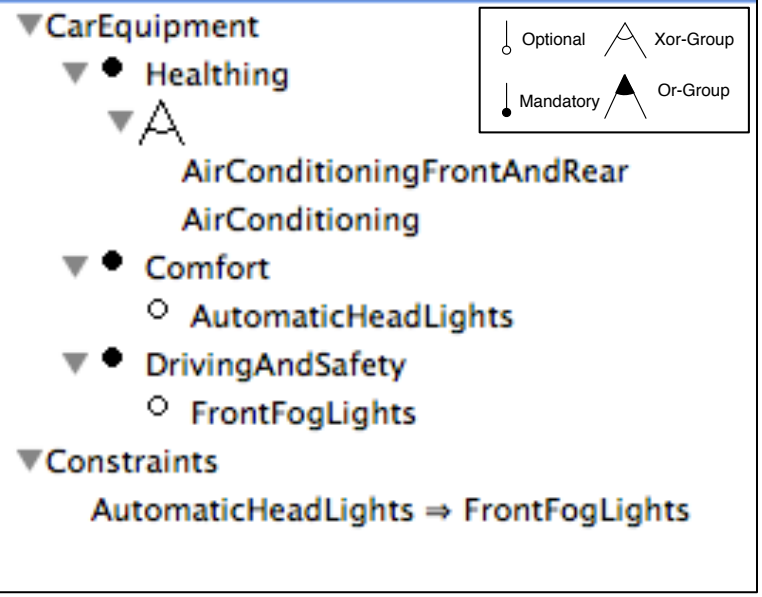

```
fml> convert fmCarEquipment into SPLOT
res1: (STRING) <feature_model name="fmCarEquipment">
<meta>
<data name="description"/>
<data name="creator"/>
<data name="address"/>
<data name="email"/>
<data name="phone"/>
<data name="website"/>
<data name="organization"/>
<data name="department"/>
<data name="date"/>
<data name="reference"/>
</meta>
<feature_tree>
:r CarEquipment(_r0)
        :m Healthing(_r1)
                :g [1,1]
                        : AirConditioningFrontAndRear(_r2)
                        : AirConditioning(_r3)
        :m DrivingAndSafety(_r4)
                :o FrontFogLights(_r5)
        :m Comfort(_r6)
                :o AutomaticHeadLights(_r7)
</feature_tree>
<constraints>
C0:~_r7 or _r5
</constraints>
</feature_model>
```

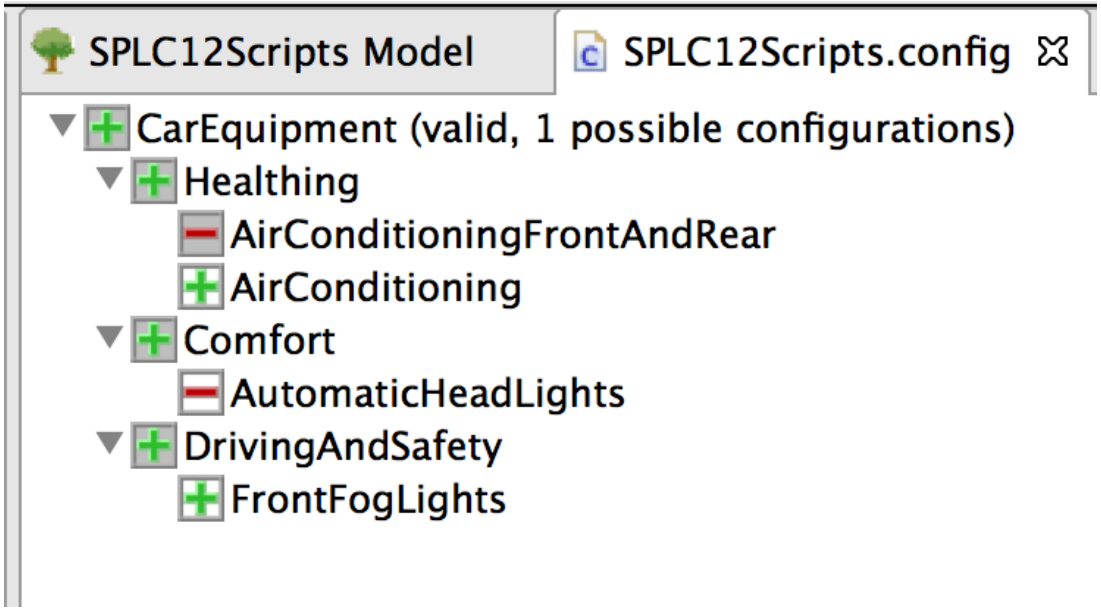
S.P.L.O.T.
Software Product Lines Online Tools

IDE
eature

```
fmCarEquipment = FM (CarEquipment : Healthing DrivingAndSafety Comfort ; // 3 mandatory features
                     Healthing : (AirConditioning|AirConditioningFrontAndRear) ; // Xor
                     DrivingAndSafety : [FrontFogLights] ; // optional
                     Comfort : [AutomaticHeadLights] ; // optional
                     // cross-tree constraints
                     AutomaticHeadLights -> FrontFogLights ; )
```

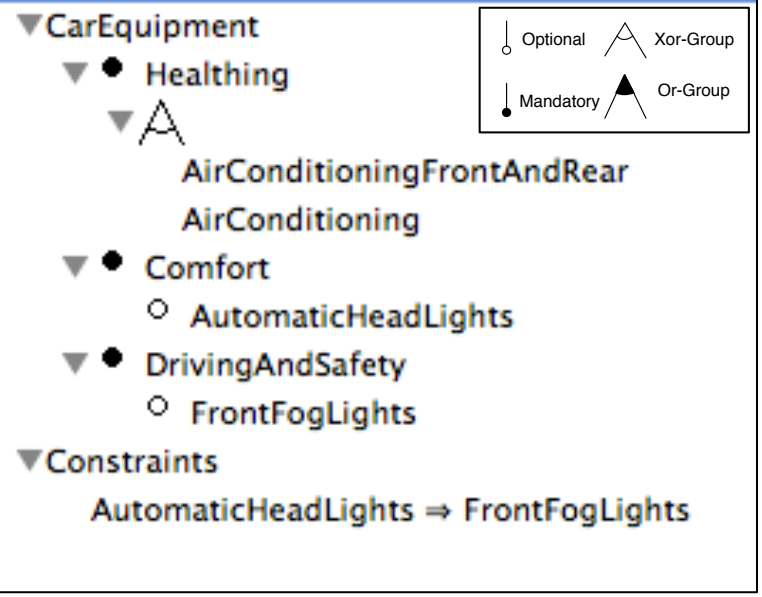
```
MacBook-Pro-de-Mathieu-3:Documents macher1$ java -Xmx1024M -jar FML-basic-1.1.jar FMLTestRepository/carEquipTuto.fml
FAMILIAR (for FeAture Model scrIpt Language for manIpulation and Automatic Reasoning)  version 1.1 (beta)
http://familiar-project.github.com/
fml> ls
(FEATURE_MODEL) fmCarEquipment
fml>
```






```
fmCarEquipment = FM (CarEquipment : Healthing DrivingAndSafety Comfort ; // 3 mandatory features
                                    Healthing : (AirConditioning|AirConditioningFrontAndRear) ; // Xor
                                    DrivingAndSafety : [FrontFogLights] ; // optional
                                    Comfort : [AutomaticHeadLights] ; // optional
                                    // cross-tree constraints
                                    AutomaticHeadLights -> FrontFogLights ; )
```

```
fml> c1 = configuration fmCarEquipment
c1: (CONFIGURATION) selected: [Healthing, CarEquipment, DrivingAndSafety, Comfort]       deselected: []
fml> select AirConditioning FrontFogLights in c1
res2: (BOOLEAN) true
fml> deselect AutomaticHeadLights in c1
res3: (BOOLEAN) true
fml> selectedF c1
res4: (SET) {Comfort;CarEquipment;Healthing;AirConditioning;DrivingAndSafety;FrontFogLights}
```



{CarEquipment, Comfort,
DrivingAndSafety,
Healthing}

X

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

```
fmCarEquipment = FM (CarEquipment : Healthing DrivingAndSafety Comfort ; // 3 mandatory features
                                    Healthing : (AirConditioning|AirConditioningFrontAndRear) ; // Xor
                                    DrivingAndSafety : [FrontFogLights] ; // optional
                                    Comfort : [AutomaticHeadLights] ; // optional
                                    // cross-tree constraints
                                    AutomaticHeadLights -> FrontFogLights ; )
```

```
fml> co = cores fmCarEquipment
co: (SET) {CarEquipment;Healthing;DrivingAndSafety;Comfort}
fml> fmCarEquipment.*
res6: (SET) {DrivingAndSafety;AirConditioningFrontAndRear;Comfort;Healthing;FrontFogLights;AirConditioning;AutomaticHeadLights;CarEquipment}
fml> setDiff fmCarEquipment.* co
res7: (SET) {AutomaticHeadLights;FrontFogLights;AirConditioning;AirConditioningFrontAndRear}
fml>
```

```
res1: (DOUBLE) 6.0
```

```
// carEquipTuto.fml
fmCarEquipment = FM ("carEquip.fml")

c1 = configuration fmCarEquipment
select AirConditioning FrontFogLights in c1
deselect AutomaticHeadLights in c1
s0 = selectedF c1


s1 = configs fmCarEquipment // configuration set
n1 = size s1
n2 = counting fmCarEquipment // efficient way, se
assert (n2 eq n1)

co = cores fmCarEquipment // features necessarily
vp = setDiff fmCarEquipment.* co
//
```

Much more than that!

Let us have a deeper look

FAMiliAR

(FeAture Model scrIpt Language for manIpulation and Automatic Reasoning)

http://familiar-project.github.com/

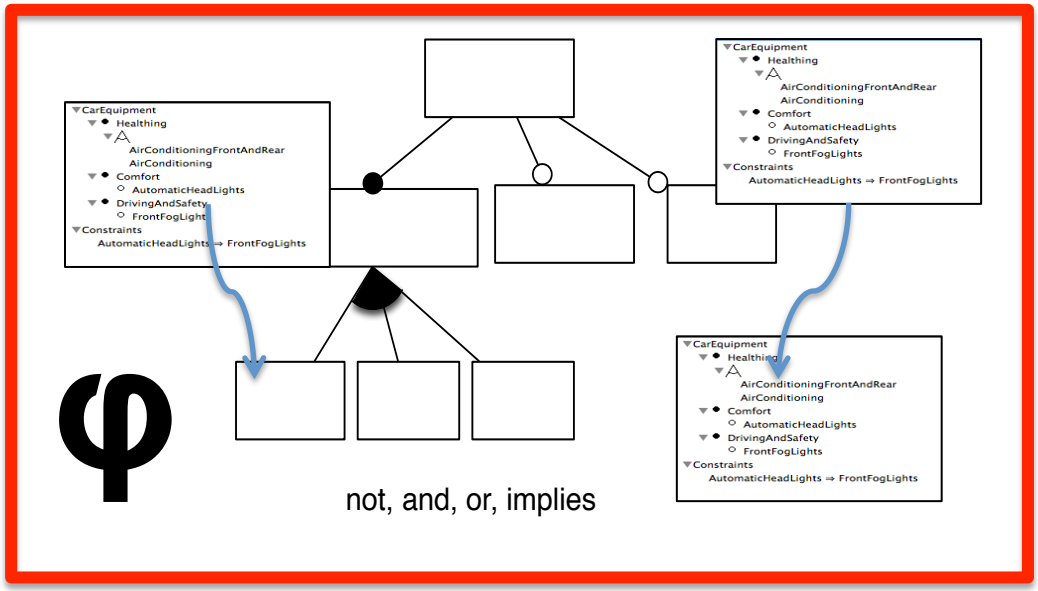
COMMON VARIABILITY LANGUAGE

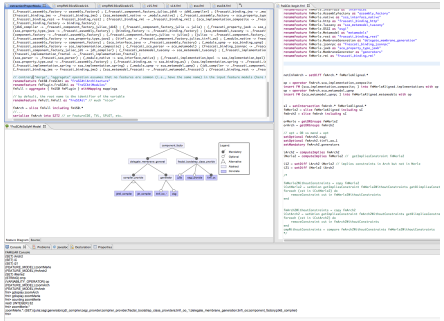
S.P.L.O.T. Software Product Lines Online Tools

IDE Feature

TVL

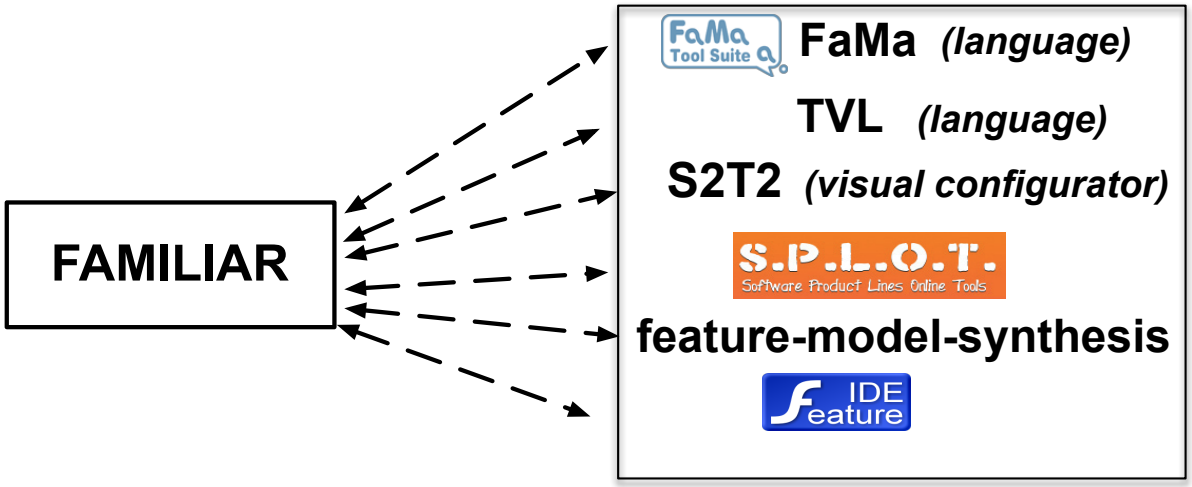
DIMACS





importing, exporting, composing, decomposing, editing, configuring, reverse engineering, computing "diffs", refactoring, testing, and reasoning about (multiple) variability models

Importing/Exporting feature models

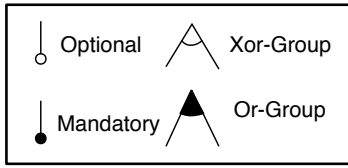


Internal notation *or* by "filename extensions"

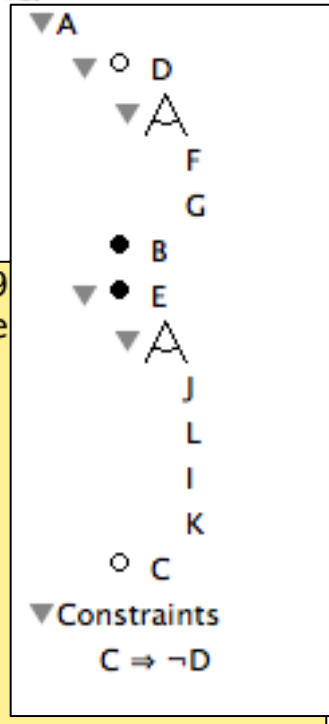
```
fm1 = FM ("foo1.tvl")
fm2 = FM (A : [B] [C] D ; )
fm3 = FM ("foo2.m")
serialize fm2 into SPLOT // export
```

conf.fml

Configuration



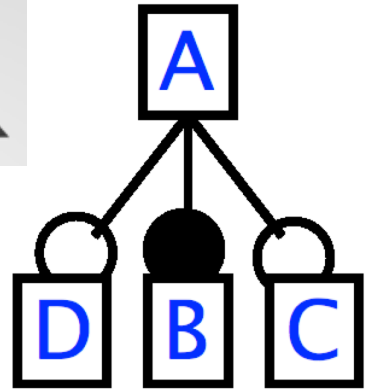

```
fm1 = FM (A: B [C] [D] E; D : (F|G) ; E : (I|J|K|L) ; C -> !D ; )
c1 = configuration fm1
select C in c1
sc1 = selectedF c1 // accessors
cFM1 = asFM c1 // configuration and FM: back!
```



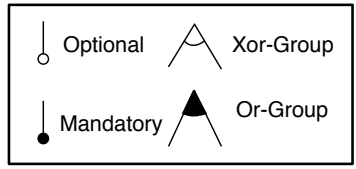

```
MacBook-Pro-de-Mathieu-2:FML-scripts macher$ java -jar -Xmx1024M ../FML-0.9.9
FAMILIAR (for FeAture Model scrIpt Language for manIpulation and Automatic Re
University of Nice Sophia Antipolis, UMR CNRS 6070, I3S Laboratory
https://nyx.unice.fr/projects/familiar/
fml> cFM1
cFM1: (FEATURE_MODEL) A: B E C ;
E: (J|L|I|K) ;
E;
A;
B;
C;
fml> fm1
fm1: (FEATURE_MODEL) A: [D] B E [C] ;
D: (F|G) ;
E: (J|L|I|K) ;
(C -> !D);
```

A ^
A ⇔ B ^
C => A ^
D => A

FAMiliAR



FM



φ

```
fml> fm1 = FM ("output/fm1.tvl")
root A {
    group [ 3..3 ] {
        opt D {
        },
        B {
        },
        opt C {
        }
    }
}
fm1: (FEATURE_MODEL) A: [D] B [C] ;
```

```
fm1bis = FM ("foo3.dimacs")
fm1bisbis = FM ("foo3.constraints")
```

```
fml> c1 = cores fm1
c1: (SET) {B;A}
fml> c1bis = cores fm1bis
```

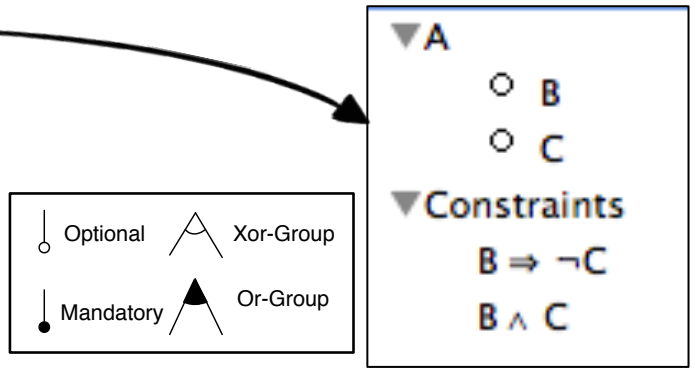
```
fml> compare fm1 fm1bis
res7: (STRING) REFACTORING
fml> compare fm1bis fm1bisbis
res8: (STRING) REFACTORING
```

```
fml> c1 eq c1bisbis
res6: (BOOLEAN) true
```

operatorsFM.fml

Operations for Feature Models (1)

```
fm1 = FM (A : [B] [C] ; B -> !C ; B and C ; )
b1 = isValid fm1
```

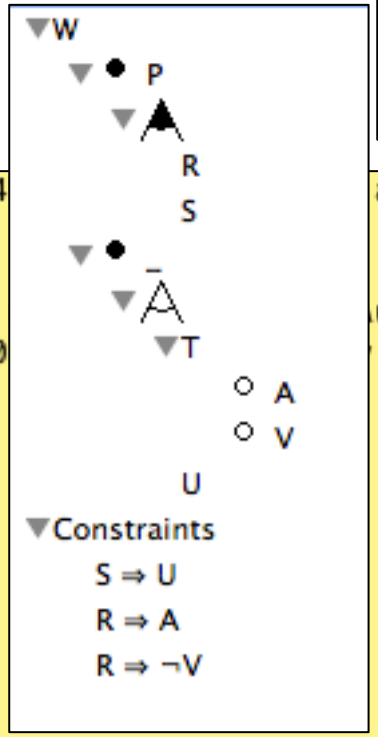


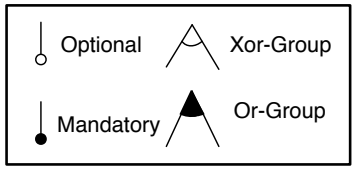

```
acher-scr:FML-scripts macher$ java -jar -Xmx1024M ../FML-0.9.9.5.jar operatorsFM.f
FAMILIAR (for FeAture Model scrIpt Language for manIpulation and Automatic Reasoni
University of Nice Sophia Antipolis, UMR CNRS 6070, I3S Laboratory
https://nyx.unice.fr/projects/familiar/
fml> ls
(FEATURE_MODEL) fm1
(BOOLEAN) b1
fml> b1
b1: (BOOLEAN) false
fml> configs fm1
res0: (SET) {}
```

operatorsFM2.fml

Operations for Feature Models (2)

```
1  fm1 = FM (W : P (T|U); P : (R|S)+ ; T : [V] [A] ; R -> !V ; S -> U ; R -> A ; )
2  b1 = isValid fm1
3  s1 = configs fm1
4  c1 = counting fm1
5  dfm1 = deads fm1
6  println "cores: ", cores fm1
7  fo1 = falseOptionals fm1
```



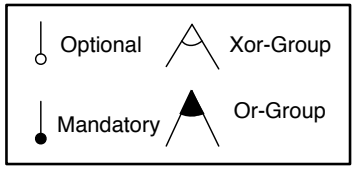



```
acher-scr:FML-scripts macher$ java -jar -Xmx1024                    ar operatorsFM2.fml
cores: {P;W}

FAMILIAR (for FeAture Model scrIpt Language for                     utomatic Reasoning)
University of Nice Sophia Antipolis, UMR CNRS 60
https://nyx.unice.fr/projects/familiar/
fml> ls
(SET) fo1
(SET) dfm1
(SET) s1
(DOUBLE) c1
(BOOLEAN) b1
(FEATURE_MODEL) fm1
fml> c1
c1: (DOUBLE) 2.0
fml> fo1
fo1: (SET) {A}
fml> dfm1
dfm1: (SET) {V}
```

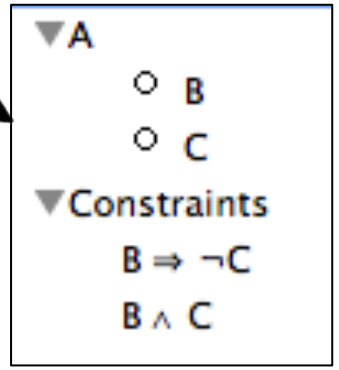
φ

operatorsFM3.fml

Operations for Feature Models (3)



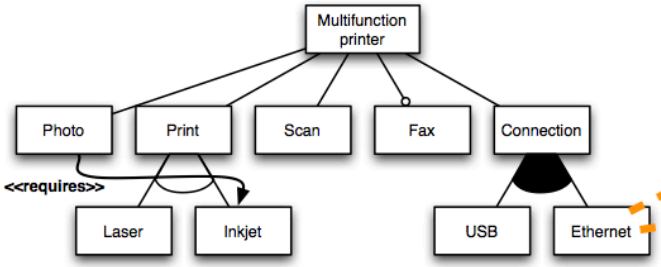

```
fm1 = FM (A : [B] [C] ; B -> !C ; B and C ; )
b1 = isValid fm1

csts1 = constraints fm1
foreach (cst in csts1) do
        println "removing constraint... ", cst
        removeConstraint cst in fm1
        c = counting fm1
        println "now the number of valid configurations is... ", c
end
```



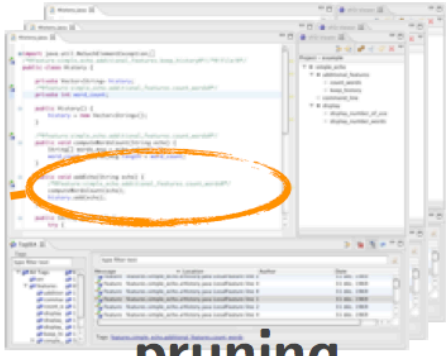

```
MacBook-Pro-de-Mathieu-2:FML-scripts macher$ java -jar -Xmx1024M ../FML-0.9.9.5.jar operatorsFM3.fml
removing constraint... (B & C)

now the number of valid configurations is... 3.0

removing constraint... (B -> !C)

now the number of valid configurations is... 4.0
```

Product Derivation

feature model



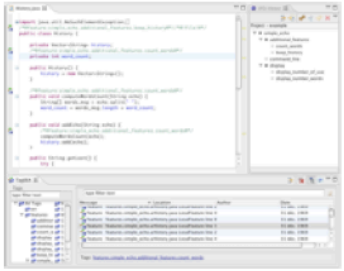
variable model and code assets



pruning,
composition,
weaving,
transformation
...

configuration

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

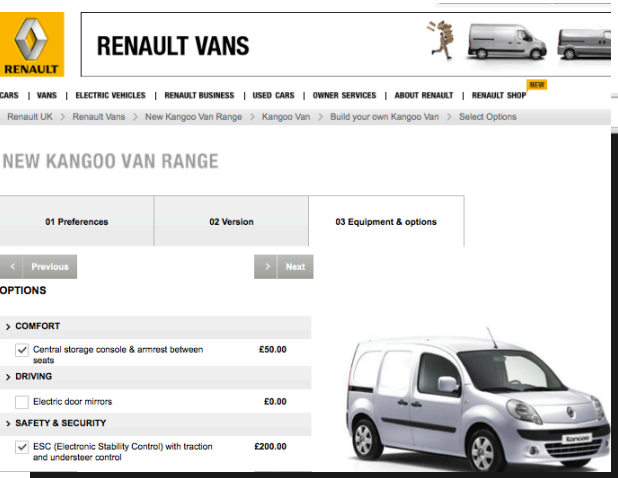


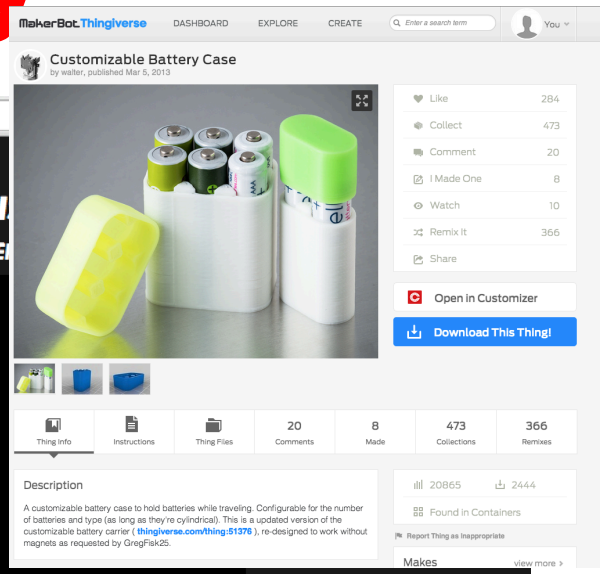
product spec

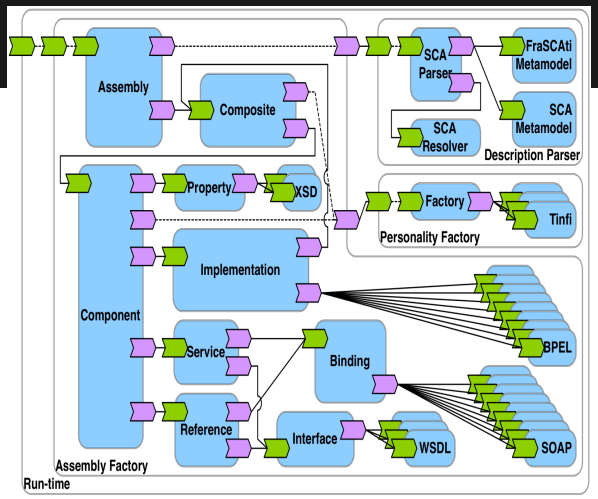
product

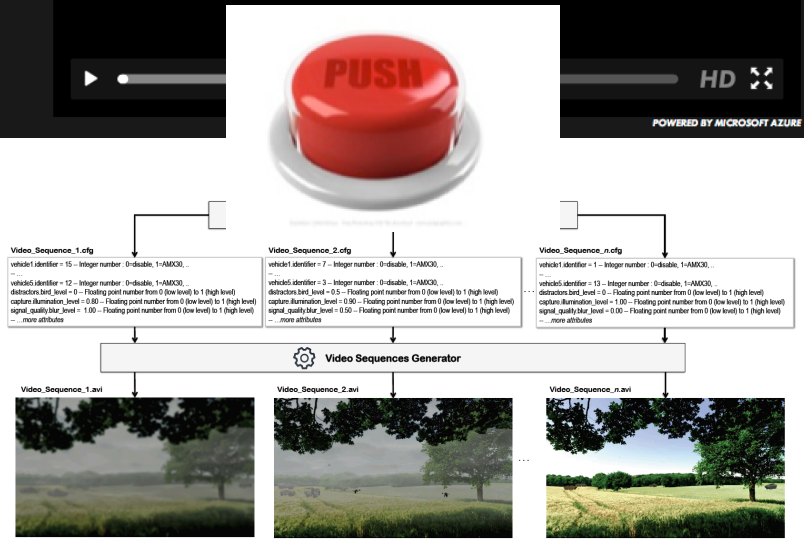
Product Lines

Vari-ability









Video_Sequence_1.cfg

vehicle1.identifier = 15 -- Integer number : 0=disable, 1=AMX30, ..
-- ...
vehicle5.identifier = 12 -- Integer number : 0=disable, 1=AMX30, ..
distractors.bird_level = 0 -- Floating point number from 0 (low level) to 1 (high level)
capture.illumination_level = 0.80 -- Floating point number from 0 (low level) to 1 (high level)
signal_quality.blur_level = 1.00 -- Floating point number from 0 (low level) to 1 (high level)
-- *...more attributes*

Video_Sequence_2.cfg

vehicle1.identifier = 7 -- Integer number : 0=disable, 1=AMX30, ..
-- ...
vehicle5.identifier = 3 -- Integer number : 0=disable, 1=AMX30, ..
distractors.bird_level = 0.5 -- Floating point number from 0 (low level) to 1 (high level)
capture.illumination_level = 0.90 -- Floating point number from 0 (low level) to 1 (high level)
signal_quality.blur_level = 0.50 -- Floating point number from 0 (low level) to 1 (high level)
-- *...more attributes*

. . .

Video_Sequence_*n*.cfg

vehicle1.identifier = 1 -- Integer number : 0=disable, 1=AMX30, ..
-- ...
vehicle5.identifier = 13 -- Integer number : 0=disable, 1=AMX30, ..
distractors.bird_level = 0 -- Floating point number from 0 (low level) to 1 (high level)
capture.illumination_level = 1.00 -- Floating point number from 0 (low level) to 1 (high level)
signal_quality.blur_level = 0.00 -- Floating point number from 0 (low level) to 1 (high level)
-- *...more attributes*

Video Sequences Generator

Video_Sequence_1.avi

Video_Sequence_2.avi

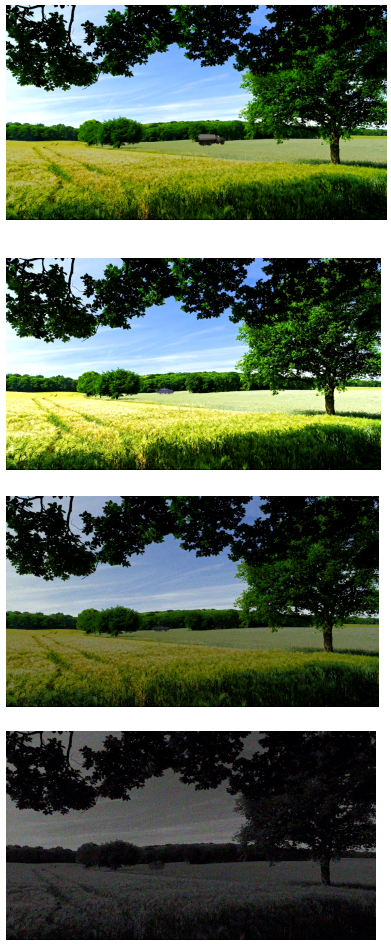
. . .

Video_Sequence_*n*.avi

Variability and Machine Learning

(contact me for internships)

Defects detection
Benchmarking
Incremental design
Performance prediction



Algorithm 1	Algorithm 2	Algorithm 3
0.63	0.93	0.82
0.81	0.92	0.81
0.43	0.3	0.8
0.39	0.03	0.01