



Model-based Variability Engineering

Dr. Mathieu Acher
Associate Professor

<http://www.mathieuacher.com>



Course given for Johannes
Kepler Universität (JKU) in Linz

(9th april 2014)

Variability and Software Product Lines

Perhaps, you ignore the names of something omnipresent in numerous contexts

The three ways to build a (software) product

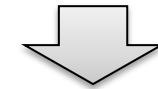
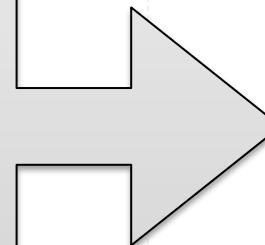
Independently

„Clone & Own“

„Shared“ (reusable) Assets

Software Product Lines

Product Configuration
Variability Modeling
Components
Domain-specific Languages
Generators
Preprocessors
Design Patterns
...

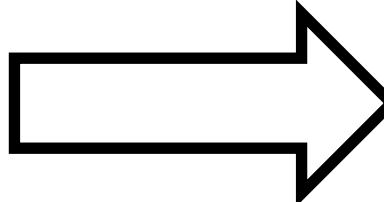


(credits: Thorsten Berger's slide)



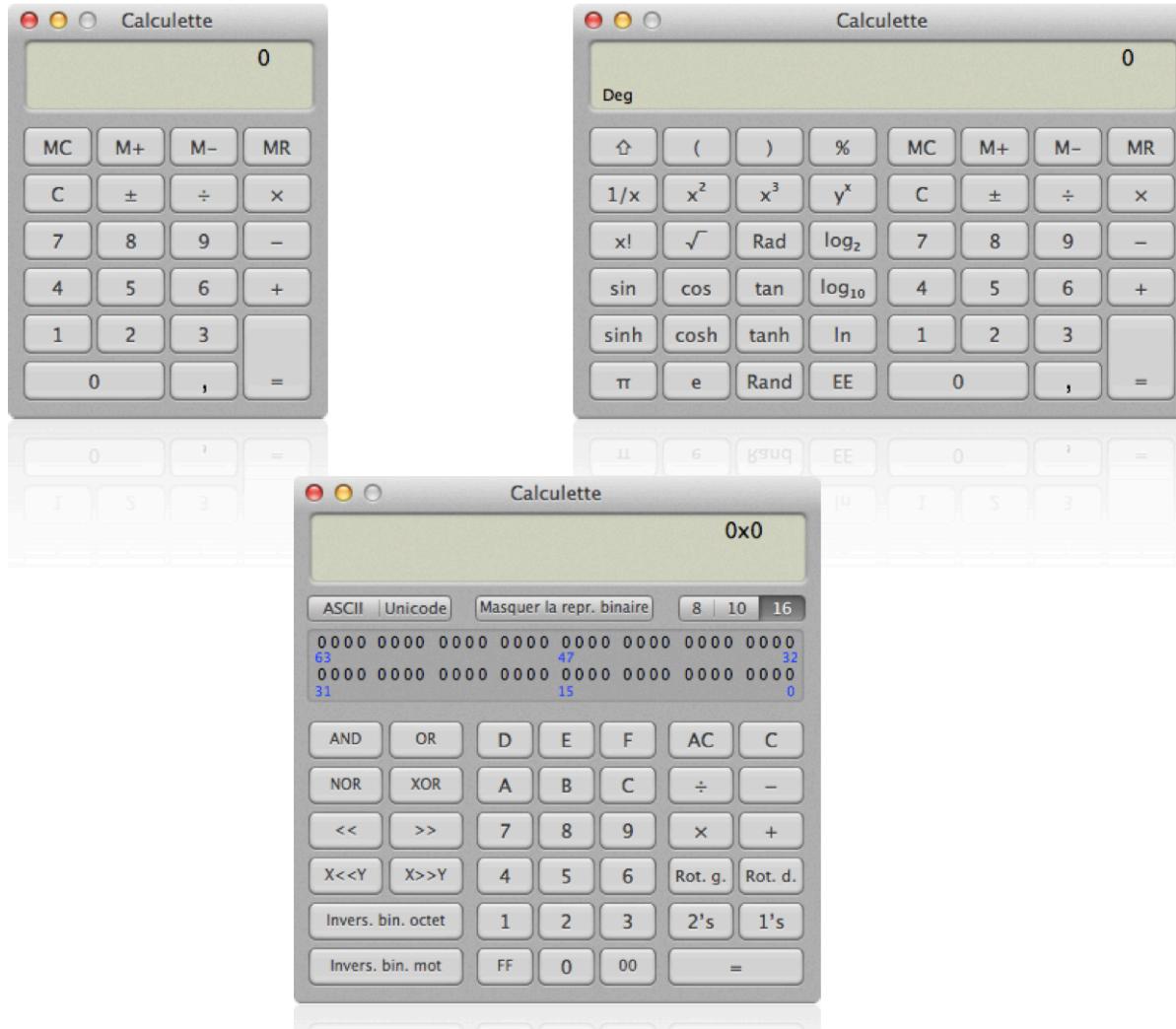


Software-intensive systems



come in many variants





« 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, 1976⁸

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)



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
Screensavers
Security
Spotlight Plugins
Utilities
System Utilities
Video
Word Processing
 GLOBAL PAGES >>
NEWS ARCHIVE >>
DAFTOPEDIA REVIEWS >>
MEET THE EDITORS >>

variability

Power Matte 2.1.3 update

 Adobe After Effects plugin that can extract a subject in an image

[\[read more >\]](#)

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

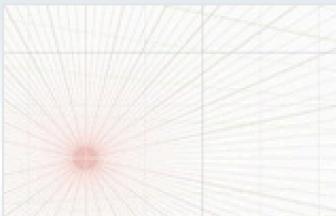


Grids 1.1 update

 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 later
License: Commercialware
Rating: Excellent (5.0/5)
Downloads: 297
Updated: June 20th, 07:53 UTC

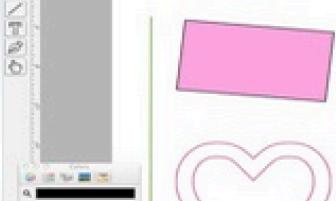


FashionLab Studio 1.1 update

 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 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

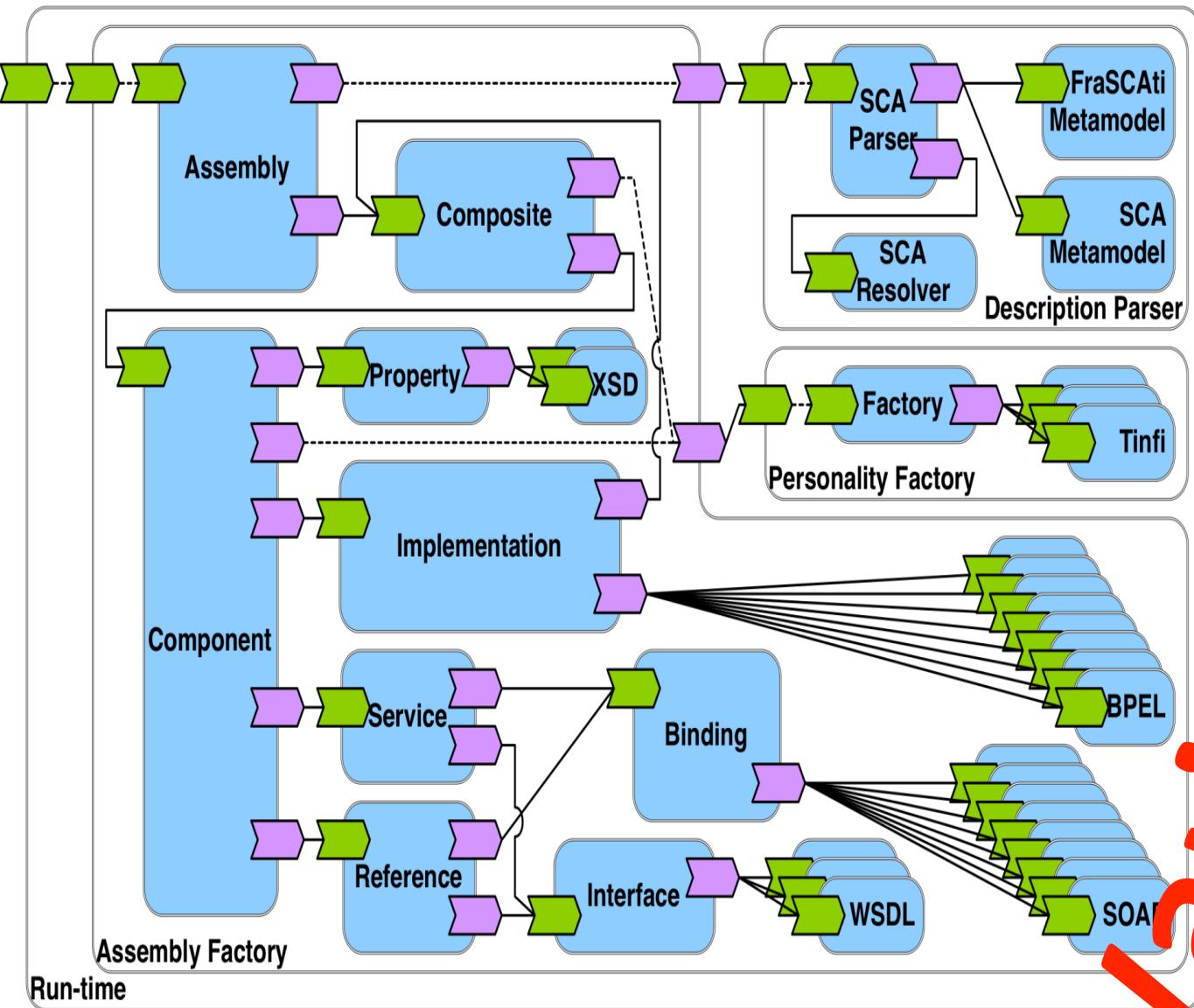
- | | |
|--|-------|
| <input type="checkbox"/> Electric door mirrors | £0.00 |
|--|-------|

> SAFETY & SECURITY

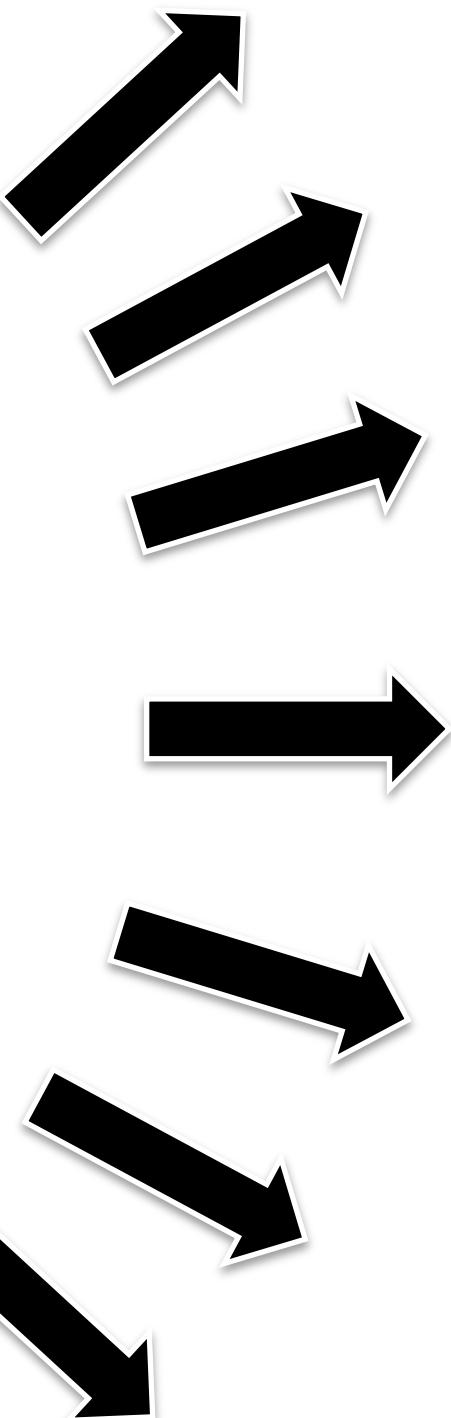
- | | |
|---|---------|
| <input checked="" type="checkbox"/> ESC (Electronic Stability Control) with traction and understeer control | £200.00 |
|---|---------|



Variability

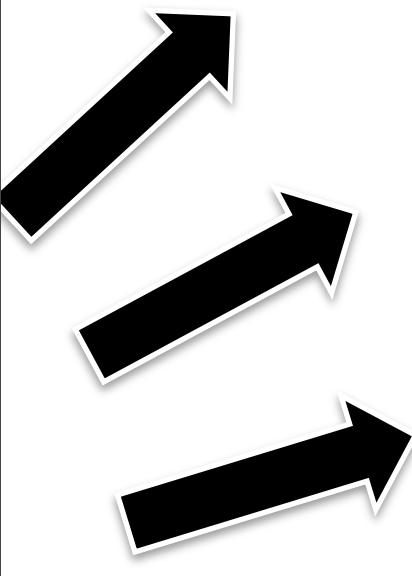


Printer Firmware



```
1 karmad 2 karmas 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, <M> 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 [ ] Support for extended (non-PC) x86 platforms [ ] Single-depth ICHAN output [ ] Paravirtualized guest support ... [ ] Memtest [ ] Processor family (Generic-x86-64) --- [ ] Preemption Model (No Forced Preemption (Server)) --- [ ] Renote for broken boot IRQs (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 [ ] 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 BIOSen [ ] MTRR (Memory Type Range Register) support [ ] MTRR cleanup support [ ] Enable seccomp to safely compute untrusted bytecode [ ] Enable -fstack-protector buffer overflow detection (EXPERIMENTAL) [ ] Timer frequency (250 Hz) --- [ ] kexec system call v(<) <Select> < Exit > < Help >
```

Linux Kernel



Variability Patterns

1. Boolean yes/no answers
2. Partial/constrained yes/no answers
3. Single-value answers
4. Multiple values answers
5. “Unknown” answers
6. Empty cells
7. Inconsistent cells
8. Additional / Extra information

Service name	Automatic forwarding	E-mail client access ¹⁴	client E-mail for other server	Integration with IM service	Domain Name customization	Interface script technique
AOL Mail	No	Yes (POP3, IMAP, SMTP)	Yes ⁰	AOL Instant Messenger	No ¹	JavaScript/ Ajax ⁴
Bigfoot Communications	Premium account only	Yes (POP3, IMAP, SMTP)	Yes (POP3 only)	XMPP ^③	Yes	HTML/ JavaScript/ CSS/Ajax
FastMail.FM	Paid accounts only	Yes (IMAP) ⁷	Paid accounts (POP3, Hotmail) ²	XMPP	Enhanced and group (Business/ Family) accounts	HTML/ JavaScript/ CSS/Ajax (Optional user supplied custom css+JavaScript)
Gmail	Yes	Yes (POP3, IMAP) SSL/TLS supported SMTP restricted ¹⁸	Yes (POP3 only)	Google Talk ^{beta} (XMPP), AOL Instant Messenger	Yes (Google Apps \$5.00 monthly/ \$50.00 annually)	HTML/ JavaScript/ Ajax ²
GMX Mail	No	Yes (POP3, IMAP ¹⁷ , SMTP) SSL/TLS supported	Yes (POP3 only)	XMPP	Yes	HTML/ JavaScript/ Ajax
Hushmail	No	Extra cost ⁸	?	No	\$1.99/\$3.99 monthly through Hushmail Business	Java or HTML
Mail.com	No	Yes (POP3, IMAP, SMTP) SSL/TLS supported	Yes (POP3 only)	Google Talk (XMPP)	No	HTML/ JavaScript/ Ajax ²
Mail.ru	Yes	Yes (POP3, IMAP)	Yes (POP3 only)	custom ^⑦	?	HTML/ Ajax (Beta)
rediff	No	Plus members only	?	Rediff Bot	Yes ^①	JavaScript/ Ajax ²
Runbox	Yes	Yes (IMAP, POP, SMTP) SSL/TLS supported	Yes (POP3, Hotmail, Gmail) SSL/TLS supported	XMPP, Google Talk, AOL Instant Messenger, MSN, ICQ, IRC ^[41]	Yes	HTML/ JavaScript/ CSS/Ajax
Seznam.cz	Yes	Yes (POP3, IMAP, SMTP) SSL/TLS supported	Yes (POP3 only)	No	No	HTML/ JavaScript
Windows Live Hotmail	Yes	Partial (POP3, SMTP) ³	Yes (POP3 only)	Windows Live Messenger	Yes ⁴	HTML/ JavaScript/ CSS/Ajax
Yahoo! Mail	Plus accounts only	Yes (POP3-Plus members only, but free in some countries, IMAP) SSL/TLS supported	?	Yes (POP3 only)	Yahoo! Messenger, Windows Live Messenger	\$35 yearly ^⑦
Yandex Mail	Yes	Yes (POP3, IMAP, SMTP, SSL)	Yes (POP3 only)	Ya Online, any XMPP IM	Yes (Free, Yandex PDD supports up to 1000 mailboxes without verification of legal use)	HTML/ JavaScript/ CSS/Ajax



WIKIPEDIA
The Free Encyclopedia

Variability

```

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();
    }
}

```

```

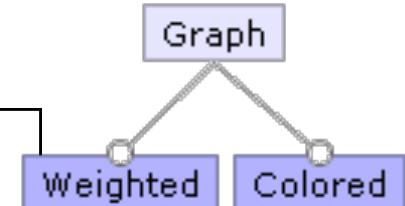
class Color {
    static void setDisplayColor(Color c) { ... }
}

```

```

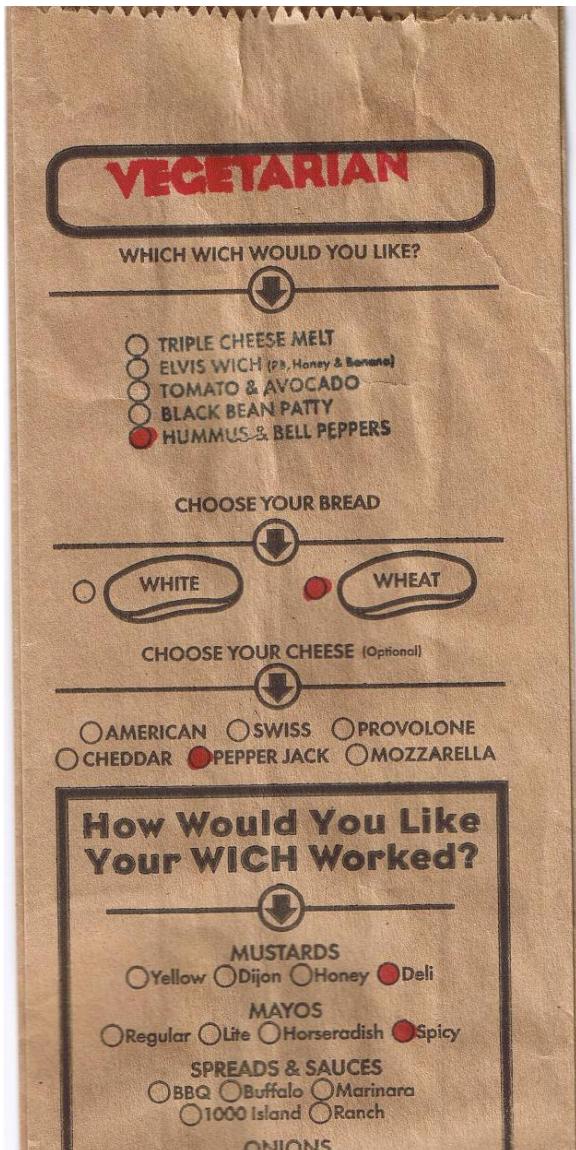
class Weight { void print() { ... } }

```



(credits: Christian Kaestner's slide)

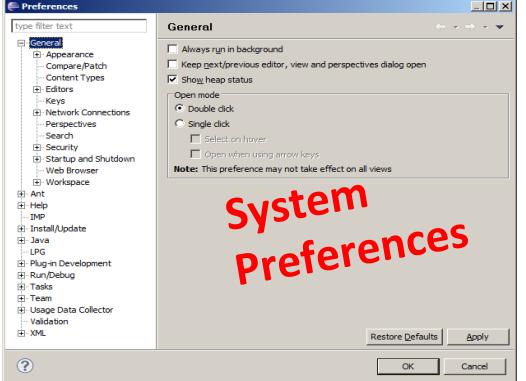
Food? Product lines!



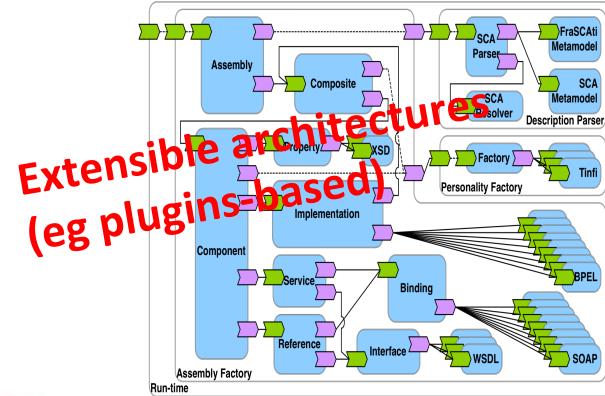


NEW KANGOO VAN RANGE

Configurators



System
Preferences



Extensible architectures
(eg plugins-based)



Product Lines and Variability

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

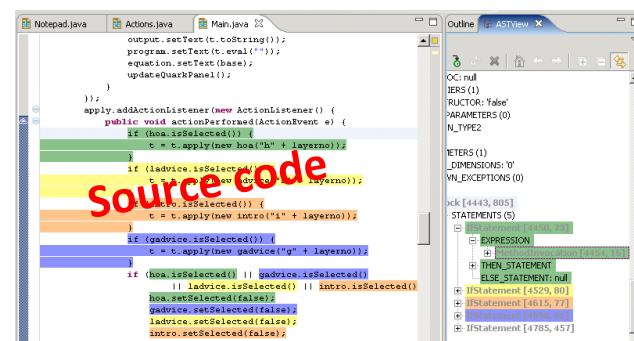
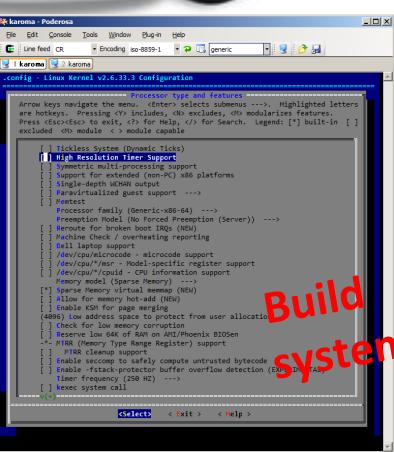
PidFile logs/httpd.pid

Timeout 300

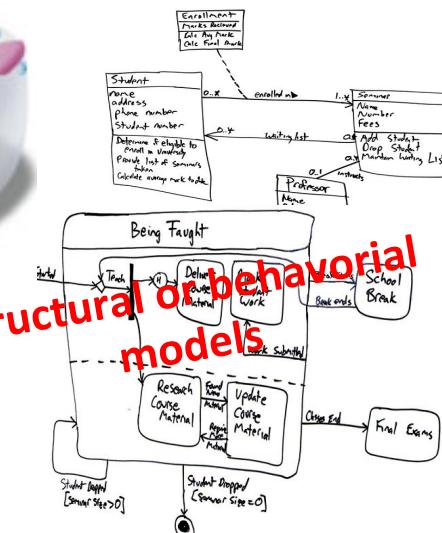
KeepAlive On
MaxKeepAliveRequests 100
KeepAliveTimeout 15

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

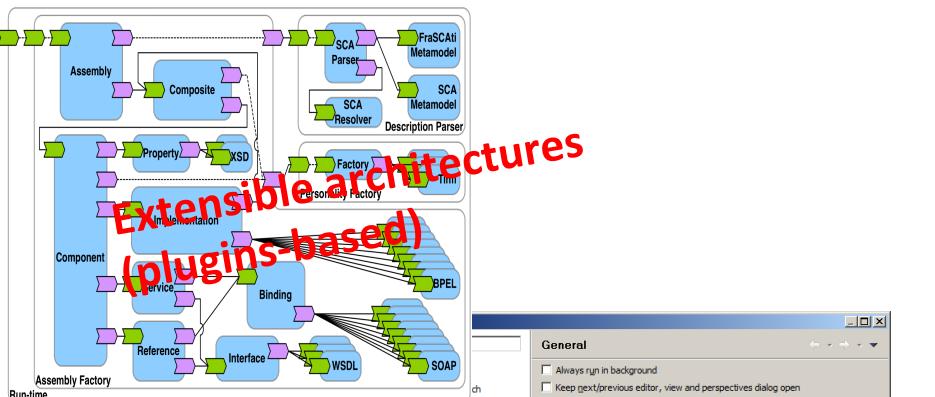
Configuration
files



Source
code



Structural or behavioral
models



<http://httpd.conf -- win32 App 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

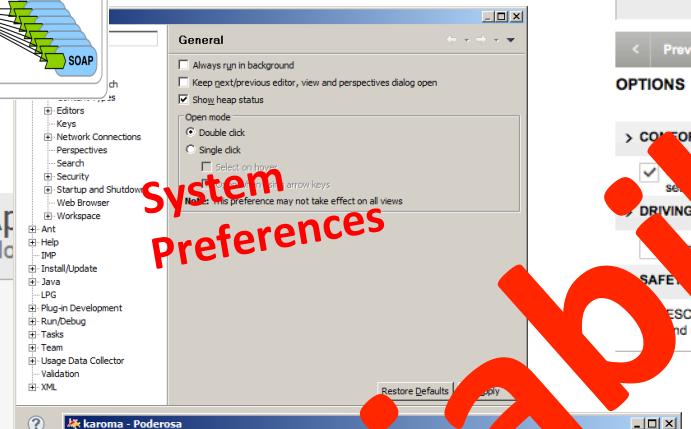
Timeout 300

KeepAlive On
MaxKeepAliveRequests 100
KeepAliveTimeout 15

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

```

System Preferences



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 > Select Options

NEW KANGOO VAN RANGE

01 Preferences 02 Version 03 Equipment & options

OPTIONS

> COMFORT
✓ Electric storage, table & armrest between seats
£50.00

DRIVING
✓ Electronic stability control (ESC) with traction and understeer control

SAFETY & SECURITY

Check the box beside the products you want which edition they're in.

COMPONENTS	Design Premium	Design Standard	Web Premium	Web Standard	Production Premium	Production Standard	Master Collection
Adobe Photoshop CS4	•	•	•	•	•	•	•
Adobe Illustrator CS4	•	•	•	•	•	•	•
Adobe Acrobat Pro CS4	•	•	•	•	•	•	•
Adobe Dreamweaver CS4	•	•	•	•	•	•	•
Adobe Fireworks CS4	•	•	•	•	•	•	•
Adobe Contributed CS4	•	•	•	•	•	•	•
Adobe After Effects CS4	•	•	•	•	•	•	•
Adobe Premiere Pro CS4*	•	•	•	•	•	•	•
Adobe Soundbooth CS4	•	•	•	•	•	•	•
Adobe OnLocation™ CS4*	•	•	•	•	•	•	•
Adobe Encore® CS4	•	•	•	•	•	•	•
SHARED FEATURES, SERVICES, AND APPLICATIONS	•	•	•	•	•	•	•
Adobe Bridge CS4	•	•	•	•	•	•	•
Adobe Device Central CS4	•	•	•	•	•	•	•
Adobe Dynamic Link	•	•	•	•	•	•	•
Adobe Version Cue® CS4	•	•	•	•	•	•	•

Compare Adobe Creative Suite 4 editions

Check the box beside the products you want which edition they're in.

COMPONENTS	Design Premium	Design Standard	Web Premium	Web Standard	Production Premium	Production Standard	Master Collection
Adobe Photoshop CS4	•	•	•	•	•	•	•
Adobe Illustrator CS4	•	•	•	•	•	•	•
Adobe Acrobat Pro CS4	•	•	•	•	•	•	•
Adobe Dreamweaver CS4	•	•	•	•	•	•	•
Adobe Fireworks CS4	•	•	•	•	•	•	•
Adobe Contributed CS4	•	•	•	•	•	•	•
Adobe After Effects CS4	•	•	•	•	•	•	•
Adobe Premiere Pro CS4*	•	•	•	•	•	•	•
Adobe Soundbooth CS4	•	•	•	•	•	•	•
Adobe OnLocation™ CS4*	•	•	•	•	•	•	•
Adobe Encore® CS4	•	•	•	•	•	•	•
SHARED FEATURES, SERVICES, AND APPLICATIONS	•	•	•	•	•	•	•
Adobe Bridge CS4	•	•	•	•	•	•	•
Adobe Device Central CS4	•	•	•	•	•	•	•
Adobe Dynamic Link	•	•	•	•	•	•	•
Adobe Version Cue® CS4	•	•	•	•	•	•	•

Notepad.java Actions.java Main.java ASTView

```

    output.setText(t.toString());
    program.setText(t.eval(""));
    equation.setText("base");
    updateQuarkPanel();
  }
)
apply.addActionListener(new ActionListener() {
  public void actionPerformed(ActionEvent e) {
    if (hoa.isSelected()) {
      t = t.apply("hoa" + layerno);
    }
    if (ladvice.isSelected()) {
      t = t.apply(new ladvice("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.get
      + ")");
    }
  }
}
)
  
```

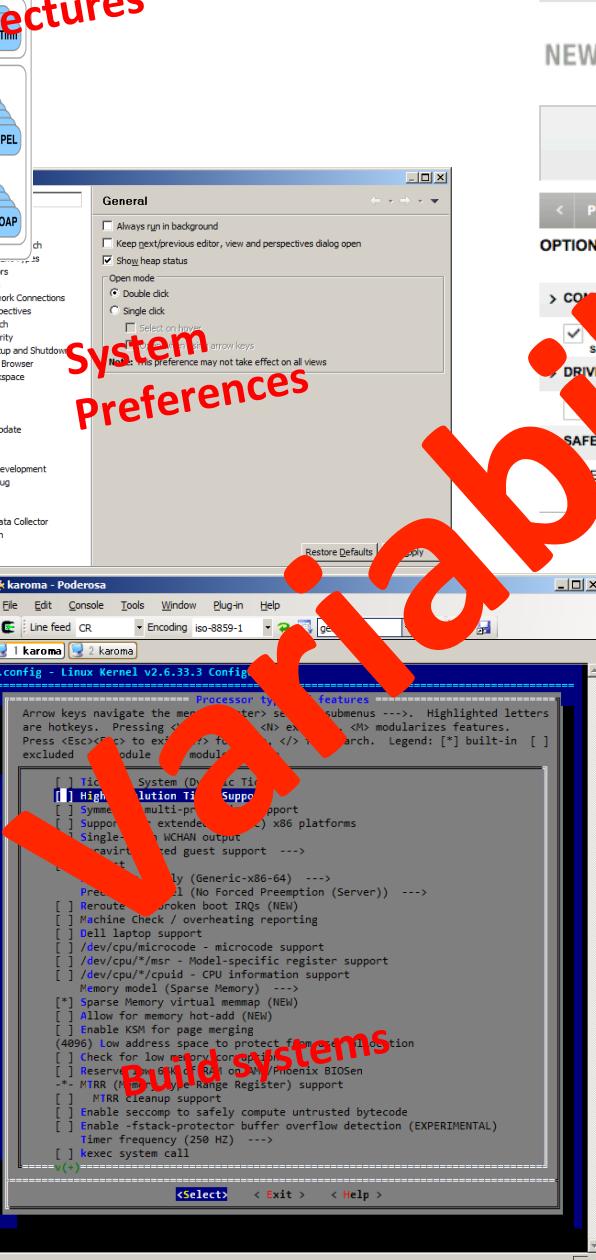
OC: null
IERS (1)
RUCTOR: 'false'
PARAMETERS (0)
N_TYPE2
IETERS (1)
DIMENSIONS: '0'
VN_EXCEPTIONS (0)

lock [4443, 805]
STATEMENTS (5)

- + EXPRESSION
- + THEN_STATEMENT
- + ELSE_STATEMENT: null
- + IfStatement [4529, 80]
- + IfStatement [4615, 77]
- + IfStatement [4593, 81]
- + IfStatement [4785, 457]

Build systems

Configurators



Comparison of Product

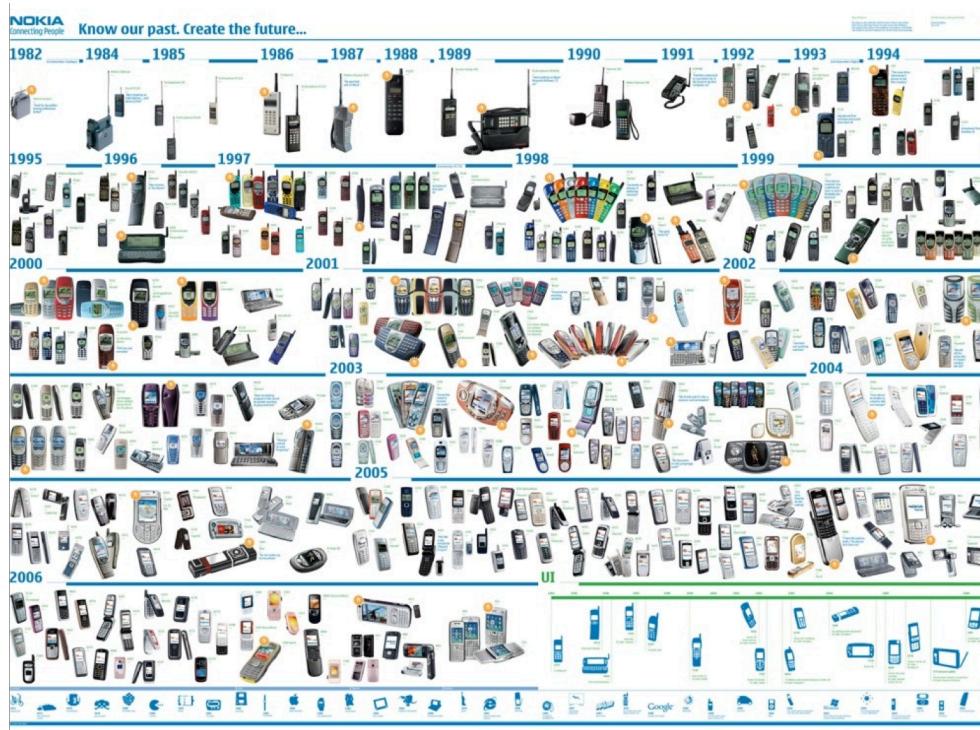
source code

Variability Engineering

Challenges, Promises, and
Benefits

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



NOKIA
Connecting People

CelsiusTech

ERICSSON



Lucent Technologies
Bell Labs Innovations



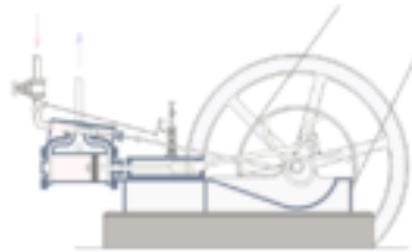


Printer Firmware

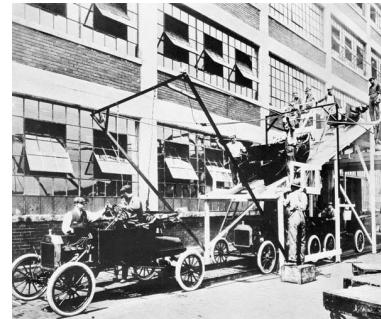
- Production cost reduced by 75%
- Development time reduced by 33%
- Reported defects reduced by 96%



A Bit of History: Industrial Revolution



1698
Thomas Savery



1901
Henry Ford



1980s

Nowaday: Product Lines Everywhere



A large, intricate 3D white maze is set against a light gray background. The maze consists of many interconnected paths and dead ends, creating a complex network of levels and corners. It occupies the entire frame, from the top left to the bottom right.

Variability = Complexity

(credits: Christian Kaestner's slide)

33 features

optional, independent



a unique variant for every
person on this planet

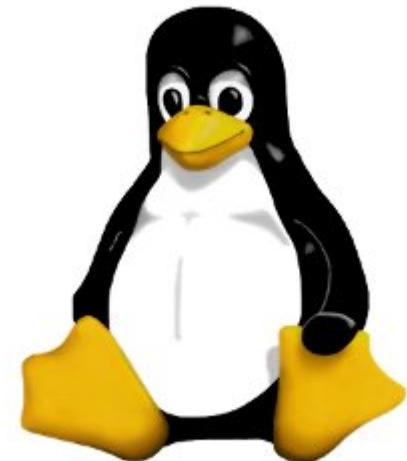
320^{optional, independent}
features

more variants than estimated
atoms in the universe



2000 features

10000
features



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

Intentional Code Cloning

~ Copy & Paste

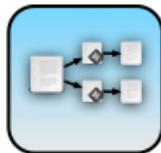
Code Cloning (example, Linux driver)

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;  
}  
.....
```

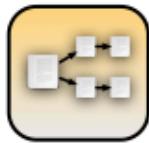


Replicate & Specialize

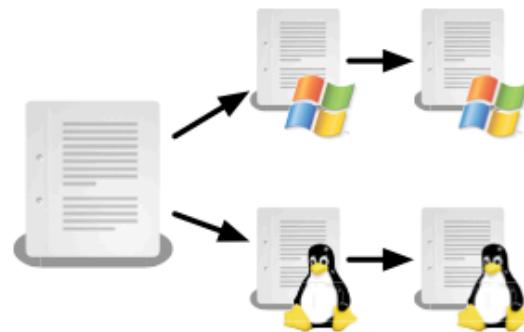


Clone to reuse and adapt existing solutions

- + 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

« 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
```

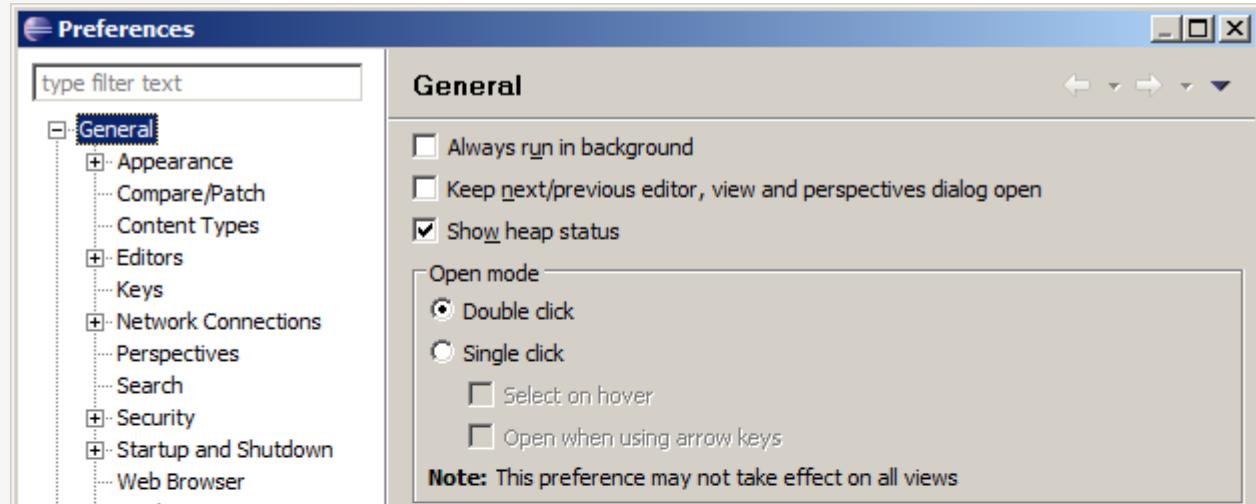
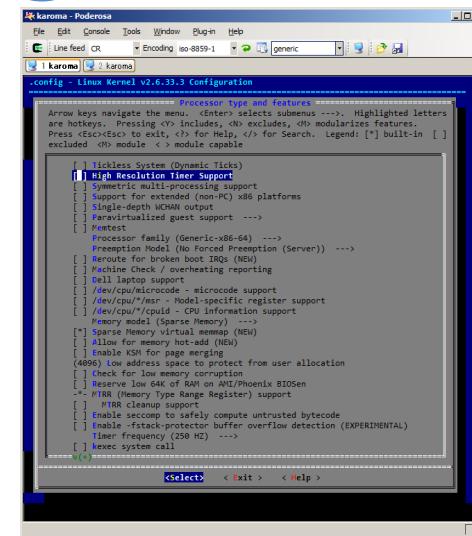
```
HostnameLookups Off
```

```
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
```

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

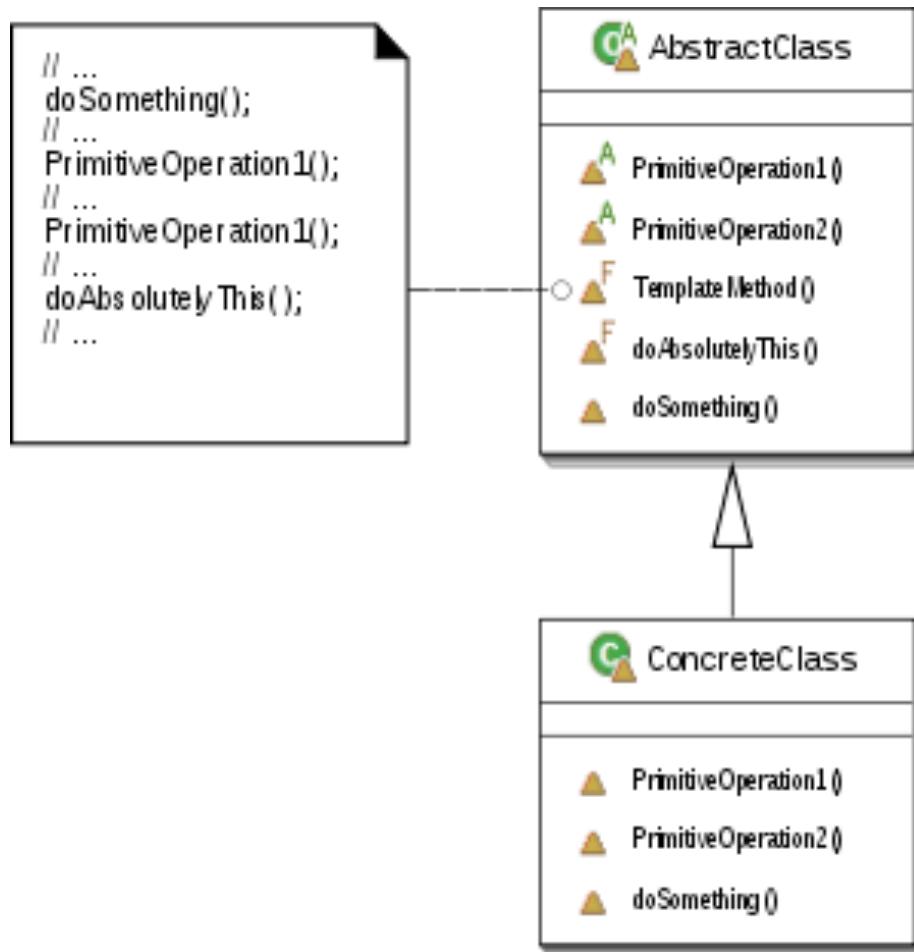
Factory

Strategy

Decorator

....

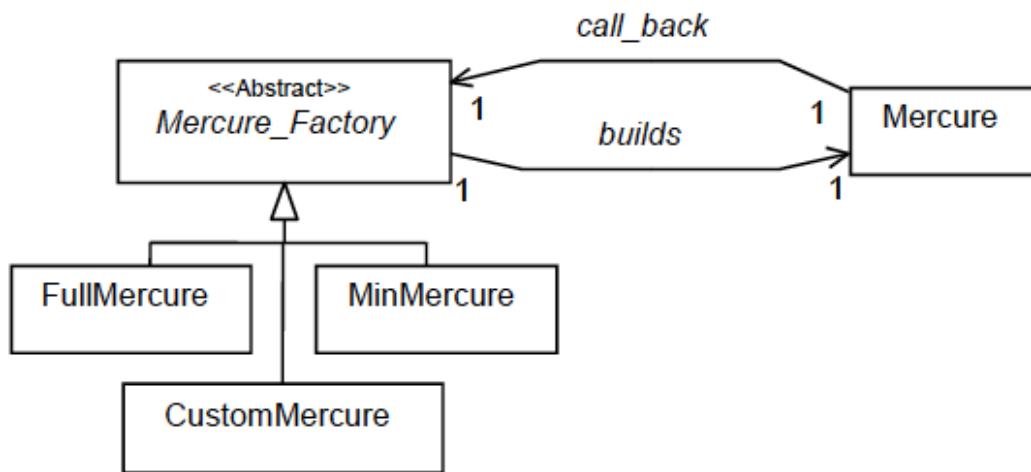
Template Method



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

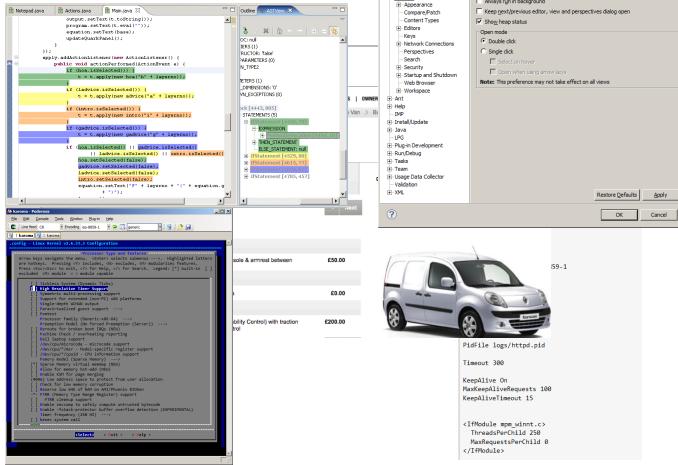
Plugin-based systems

(Active) Annotations

can have parameters

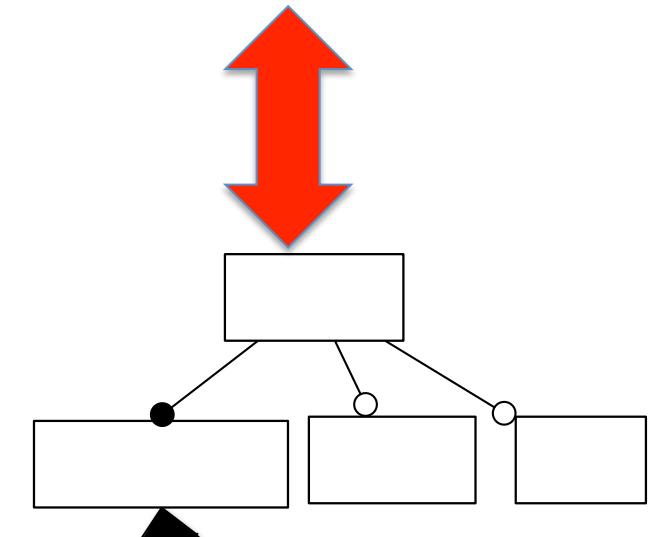
Metamodeling and Domain-Specific Languages

Variability
(general approach)

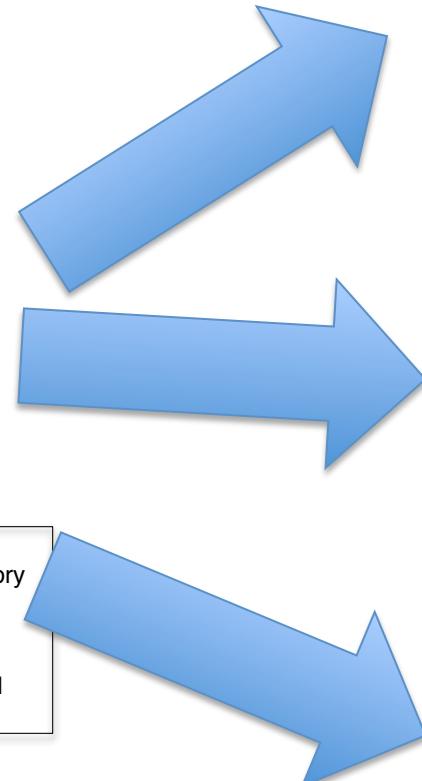


Variants of code (e.g., Java ou C)
 Variants of user interfaces
 Variants of video sequences
 Variants of models (e.g., UML or SysML)

...

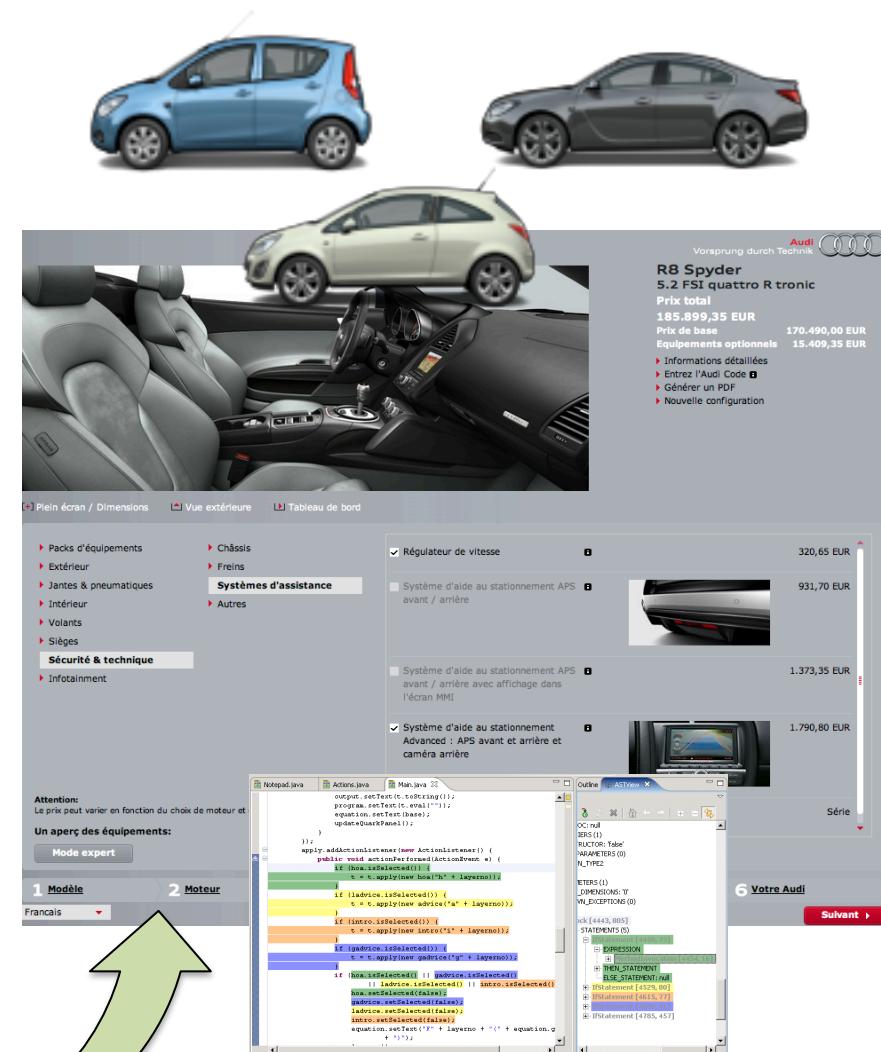
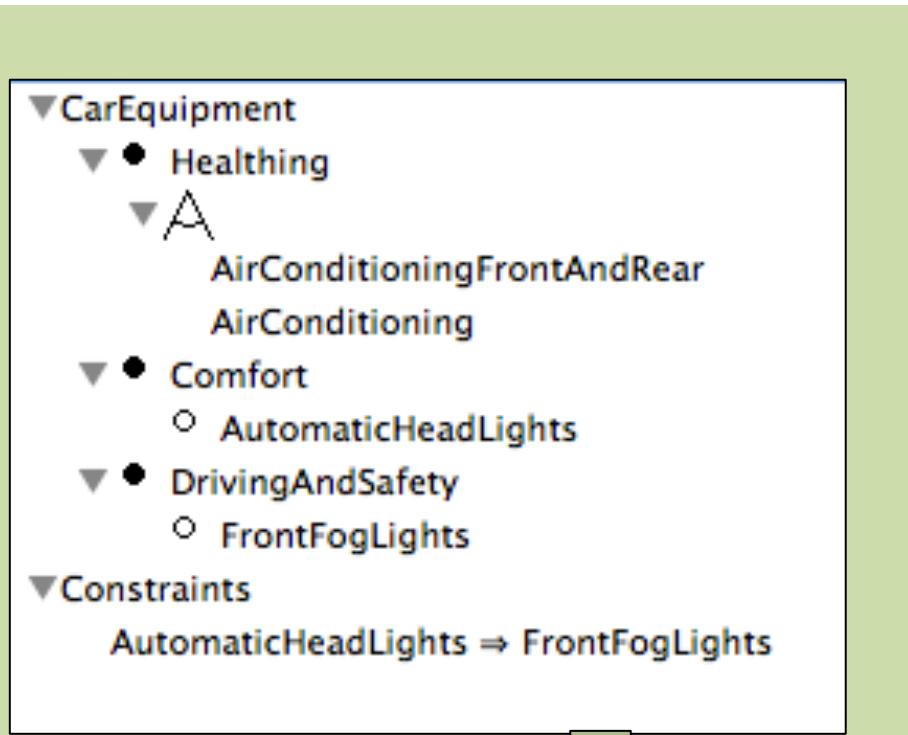


not, and, or, implies

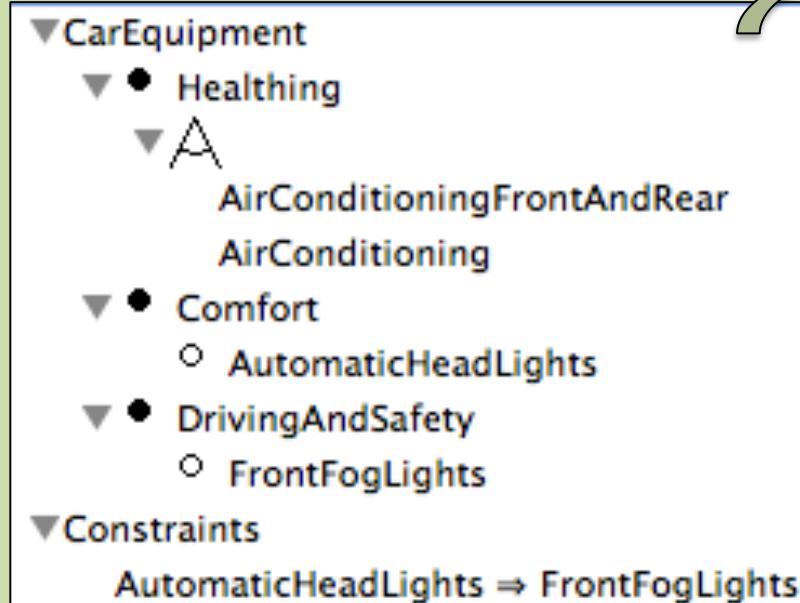


Variability Models (feature models)

Feature Models



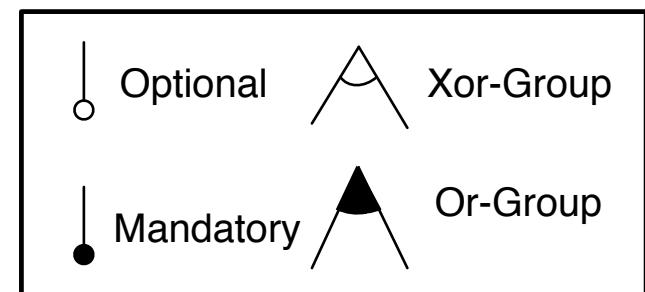
Feature Models (Background)

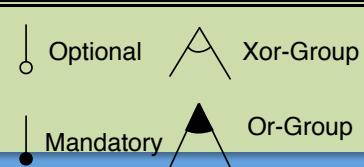
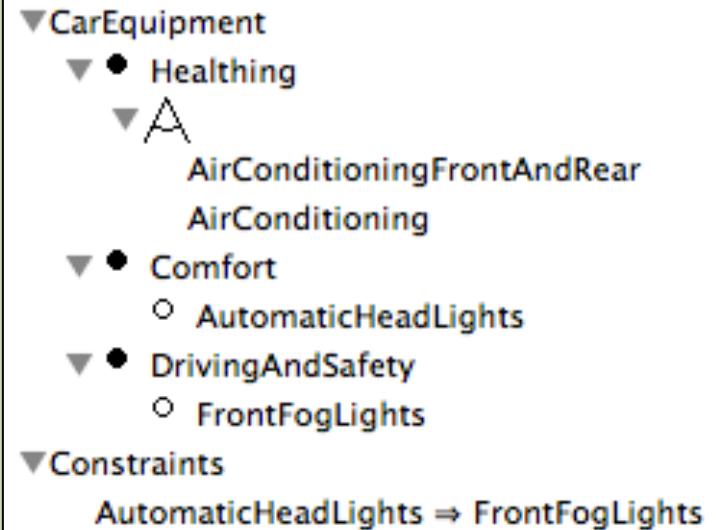
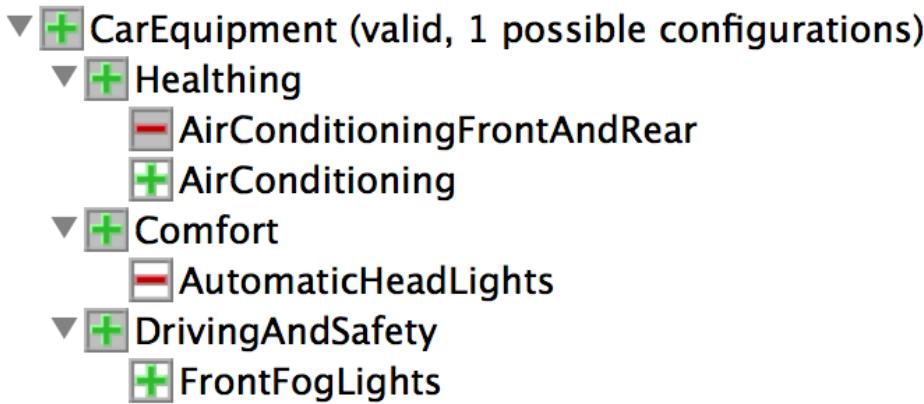


Hierarchy: rooted tree

Variability:

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





Hierarchy + Variability

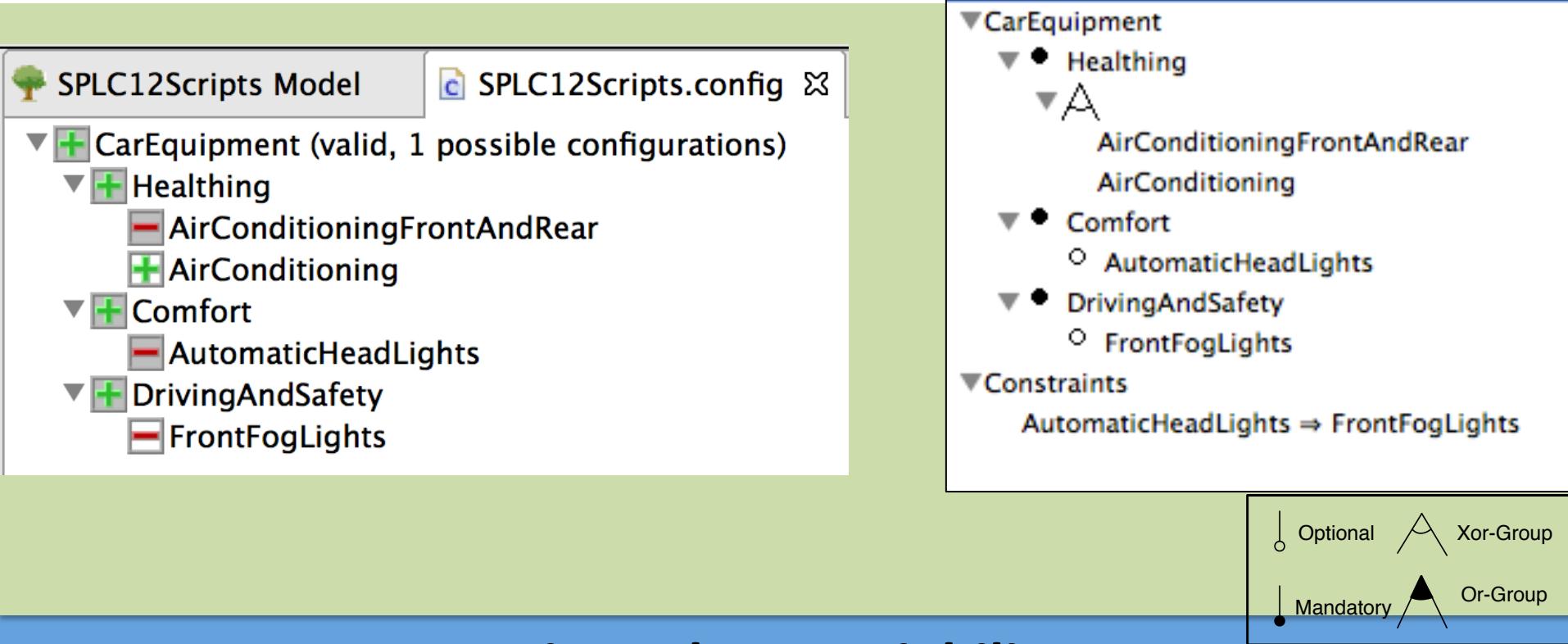
=

set of valid configurations

configuration = set of features selected

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





Hierarchy + Variability

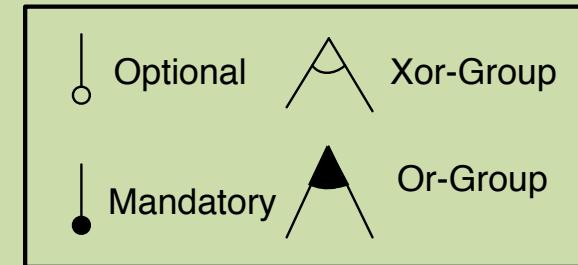
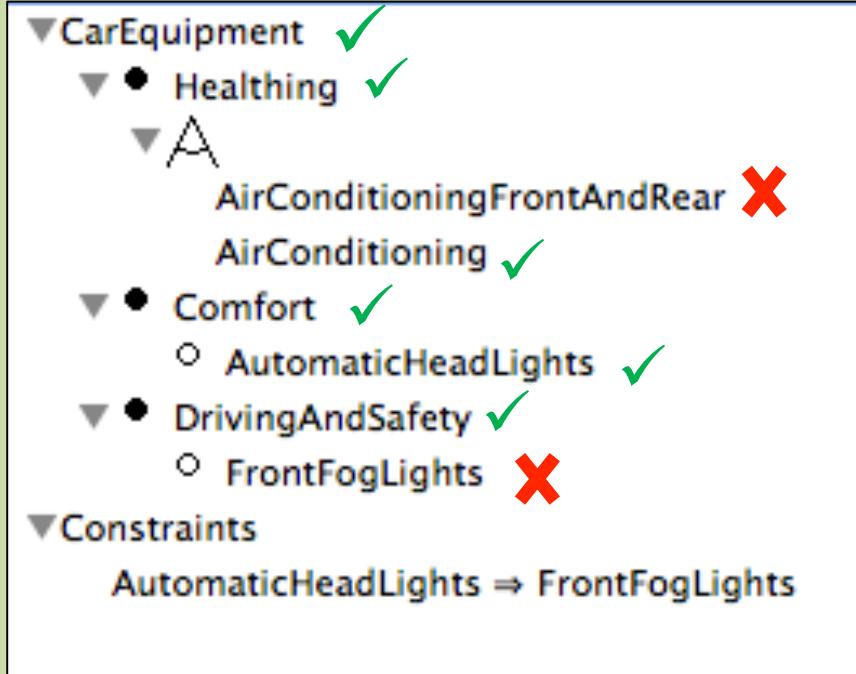
=

set of valid configurations

configuration = set of features selected

{CarEquipment, Comfort, DrivingAndSafety, Healthing, AirConditioning}





Hierarchy + Variability

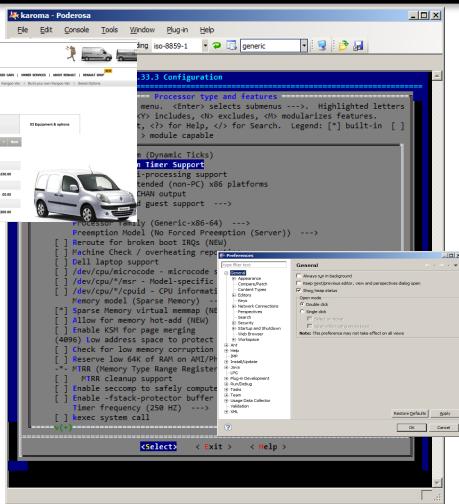
=

set of valid configurations

configuration = set of features selected

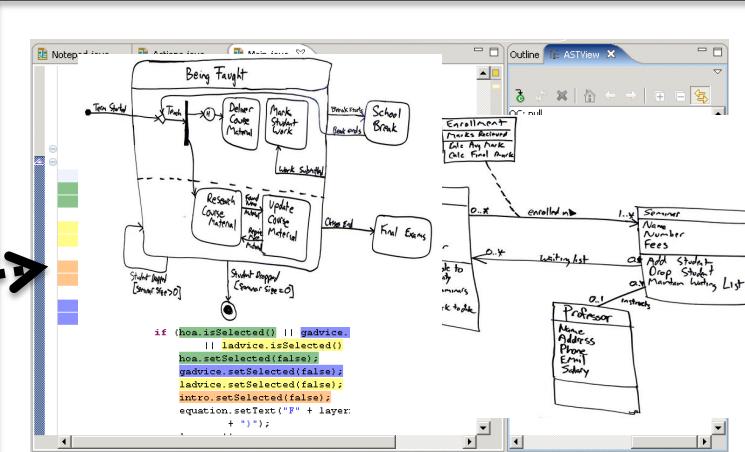
{CarEquipment, Comfort, DrivingAndSafety, Healthing, AirConditioning, AutomaticHeadLights}





Feature Model

Variability Realization Model (VRM)



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



Configuration
(resolution model)



Software Generator
(derivation engine)



A photograph of an old, green-painted pickup truck that has been left to decay in a field. The truck is heavily rusted, particularly on the body and the front fenders. The driver's side door is open, revealing the interior frame and some remaining mechanical components. The truck is positioned in front of a dense thicket of green bushes and tall grass. In the bottom left corner, there are some wooden planks and metal debris, suggesting a construction or demolition site nearby.

Unused flexibility

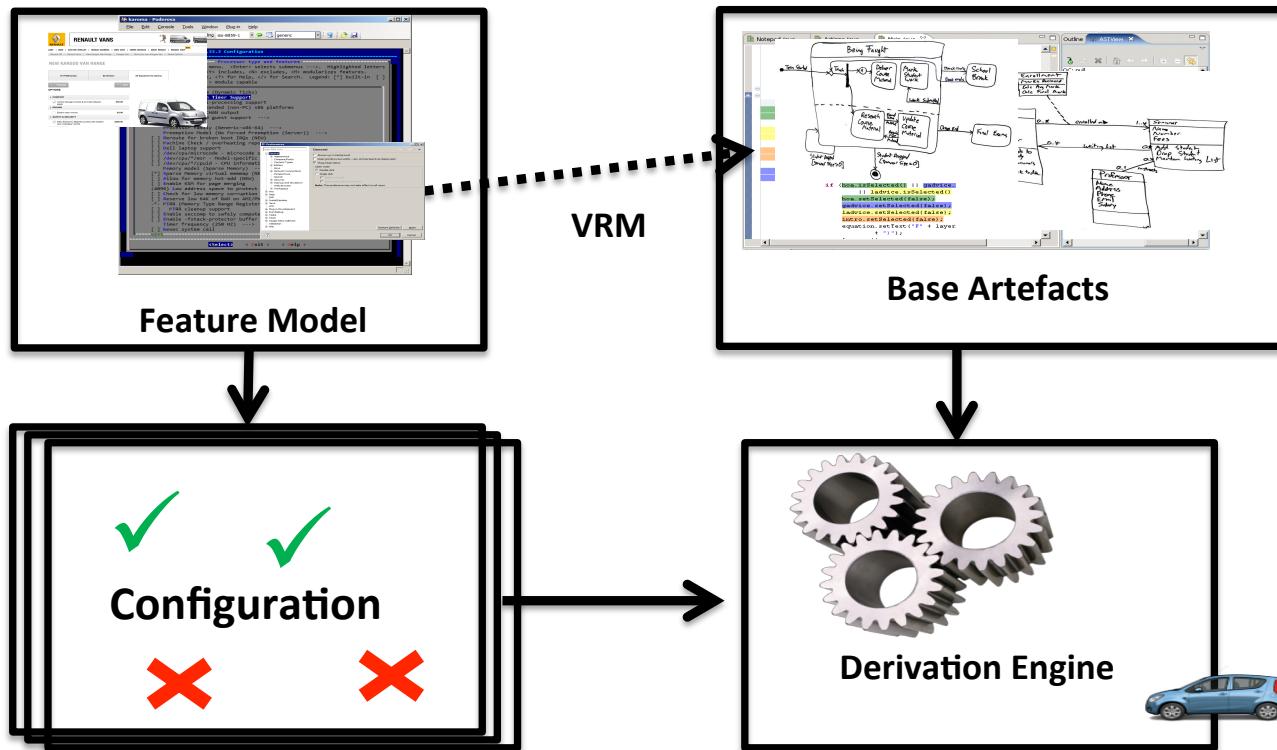


Illegal variant

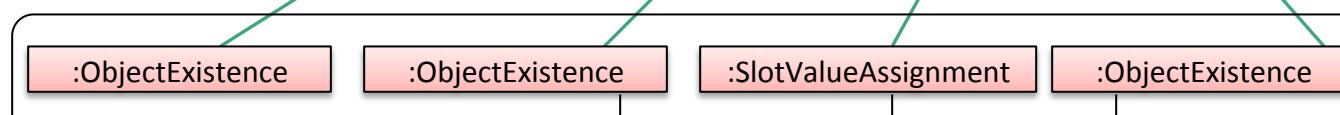
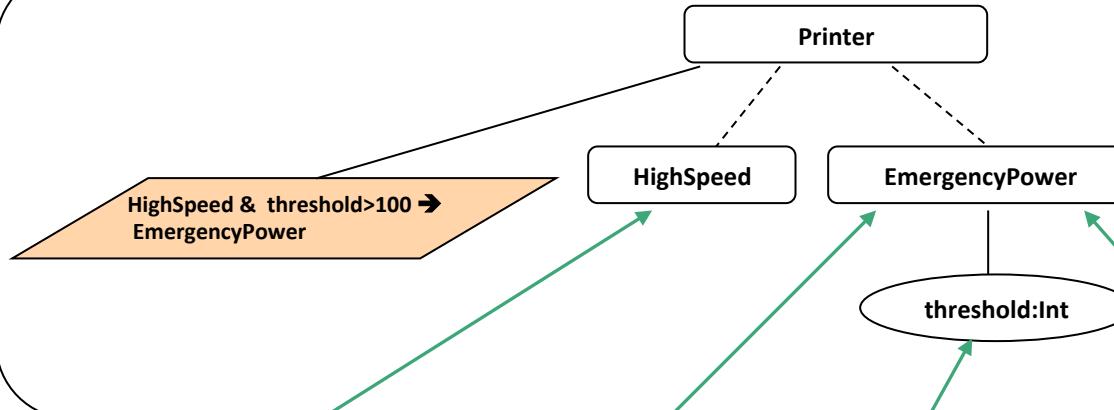
Variability and Standardization

Three pillars of CVL

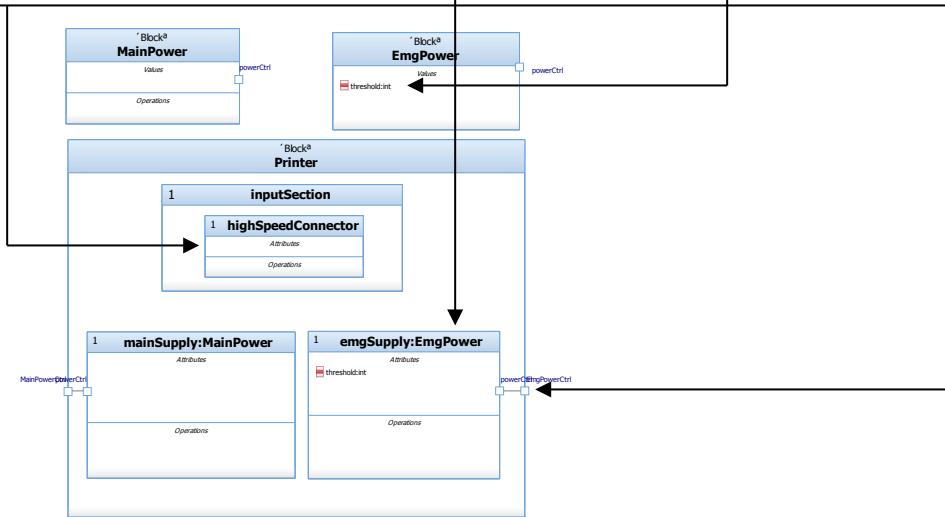
- Feature Models with attributes and multi-features
- Base "model" (eg UML or SysML model)
- Variability Realization Model (VRM)
 - add, remove, substitute, parameterization



(Attributed) Feature Model

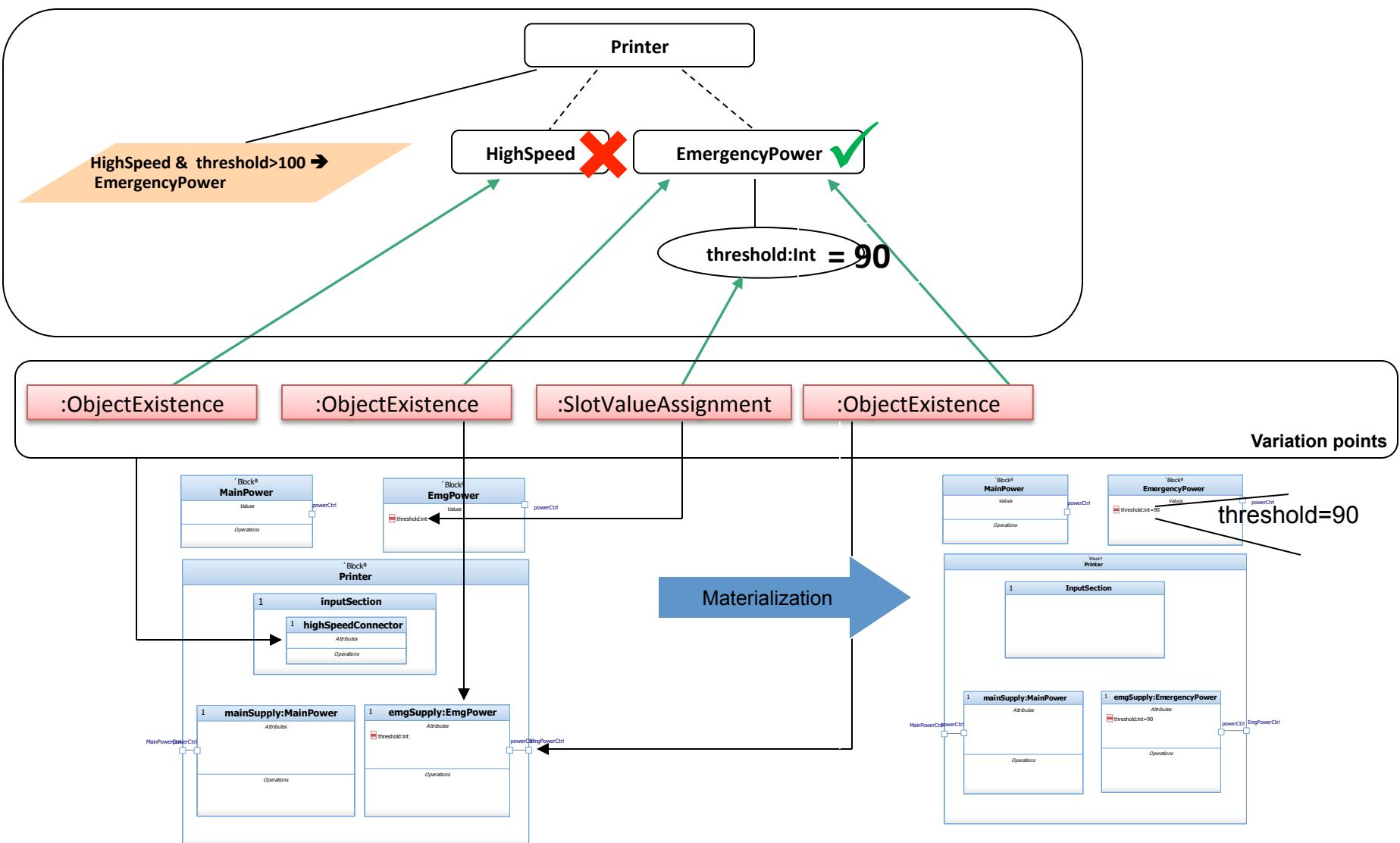


Variation
points
(VRM)



Based Model

Configuration and Product Derivation



Web Configurators (visualization aspect)

How to render a product visualization given a set of options selected?

Configuration steps

Choose a fabric

Choose your shirt's fabric using the drop-down menu.
Then click on a fabric swatch to see it on the shirt designer.

Collection & Price:

All

Work Shirt

Autograph Design Exeter

Exeter is a Easy-Iron.Easy-Care, 100% Cotton fabric from the Autograph Design collection.
This Stitch Stripe Poplin fabric has a Black mix colour.

Density: 40 / 1 * 40 / 1

Weight: 128 g/m²

Configuration option

49 £

Product visualisation

The image shows a screenshot of the Marks & Spencer web configurator. At the top, a horizontal navigation bar lists steps: Fabric, Design, Monogram, Sizing, Extra, and Review. The 'Fabric' step is highlighted with a yellow background. Below the navigation, a section titled 'Choose a fabric' contains instructions and a dropdown menu labeled 'Collection & Price: All'. A grid of fabric swatches is displayed, with one specific swatch in the first row, first column highlighted by a green border. To the right of the swatches, a 'Work Shirt' is shown with its details: 'Autograph Design Exeter', 'Exeter is a Easy-Iron.Easy-Care, 100% Cotton fabric from the Autograph Design collection.', 'This Stitch Stripe Poplin fabric has a Black mix colour.', 'Density: 40 / 1 * 40 / 1', and 'Weight: 128 g/m²'. An orange box labeled 'Configuration option' points to the fabric selection area. To the right of the shirt, a price of '49 £' is displayed above a large image of the shirt. An orange box labeled 'Product visualisation' points to this image. Further to the right, there are several smaller images showing different views of the shirt (front, back, close-ups) and a zoomed-in view of the fabric texture.

Marks & Spencer web configurator

Fabric	Design
Work Shirt Exter Autograph Design (C49) Elastan Easy-Care 100% Cotton Stitch Stripe Poplin Black mix	Collar: Classic Point Sleeve: cuff 2 Buttons Pocket: No Pocket Placket & Buttons: Real front White Pearl Buttons: Matching Stitching Base Item: Curved Contrasting: Extra:

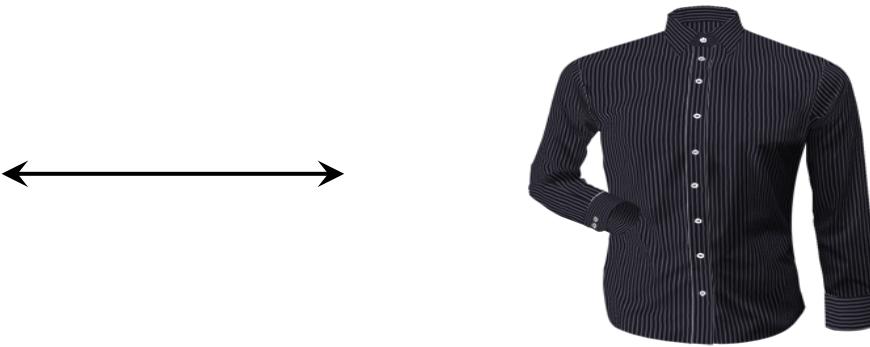
Change

Sizing

Person's name: Him
Height: 4 feet 8.0 inch
Collar Size: 14.00 inch
Weight: 99 lbs/stone
Age: 16
Fit: Regular Fit
Change

Monogram

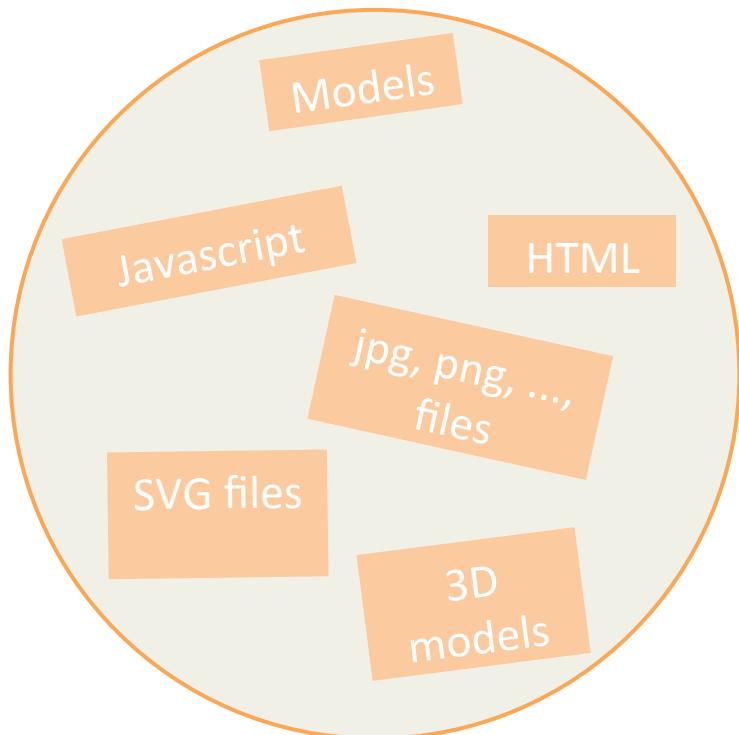
Text:
Colour:
Font:
Position:
Change



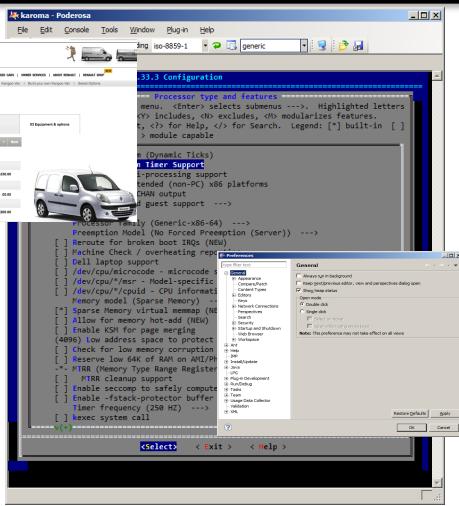
PRODUCT CONFIGURATION



VISUAL REPRESENTATION

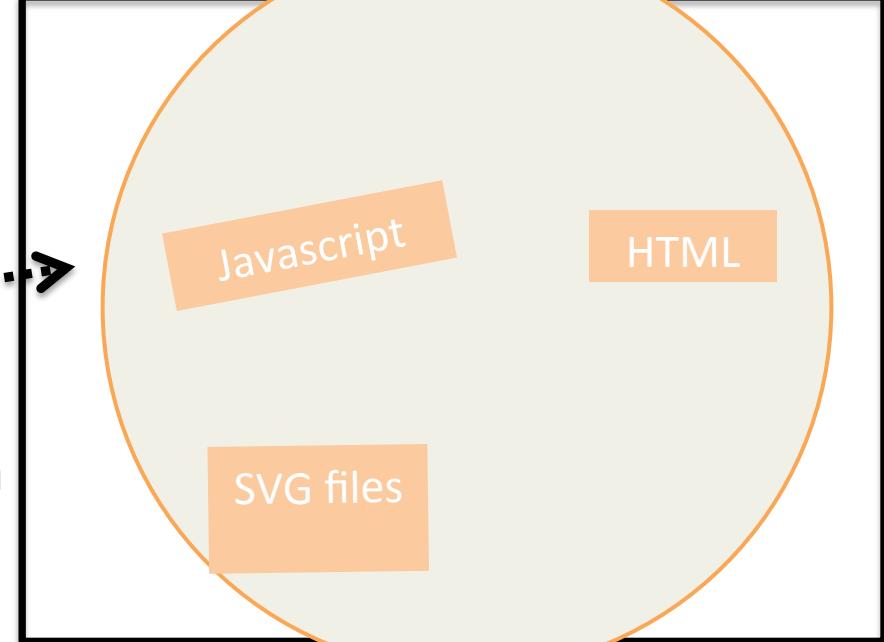


Feature
models



Feature Model

Variability Realization Model (VRM)



✓ ✓
Configuration
(resolution model)
✗ ✗



Variability in the Video Domain



What are the differences?

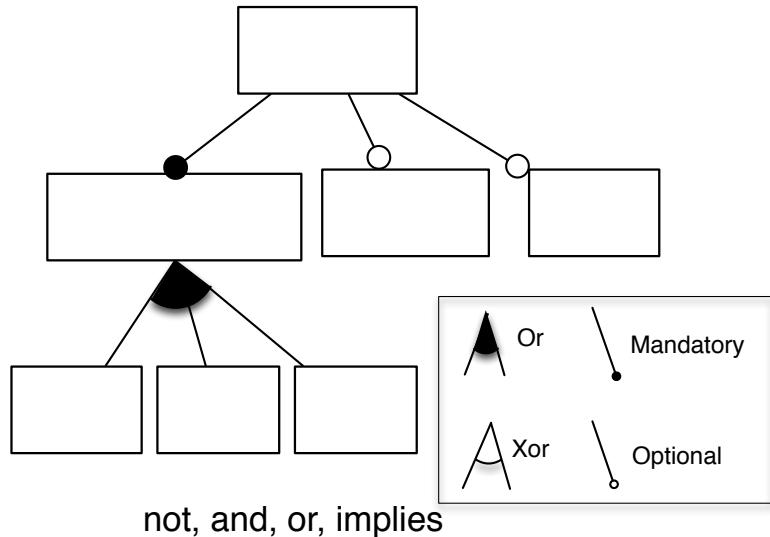




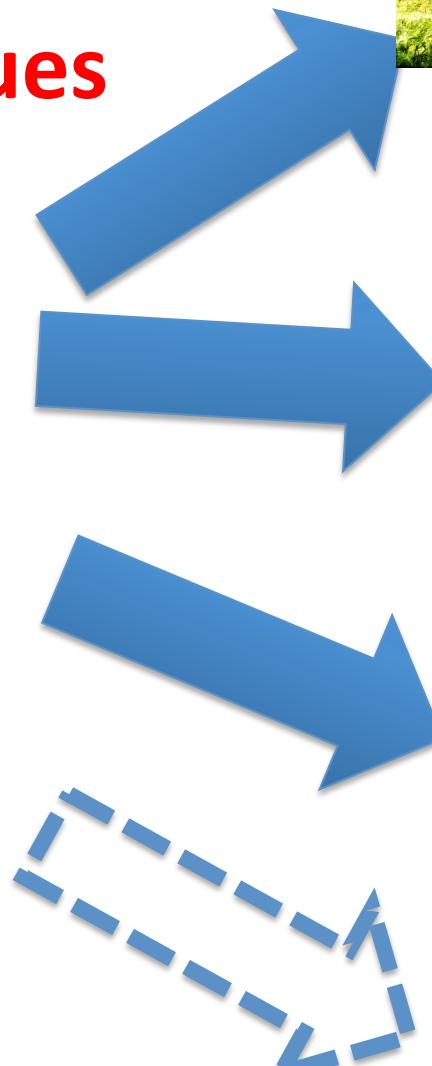
aka what is the variability of a video?



We synthesize video sequence variants with variability techniques



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.**

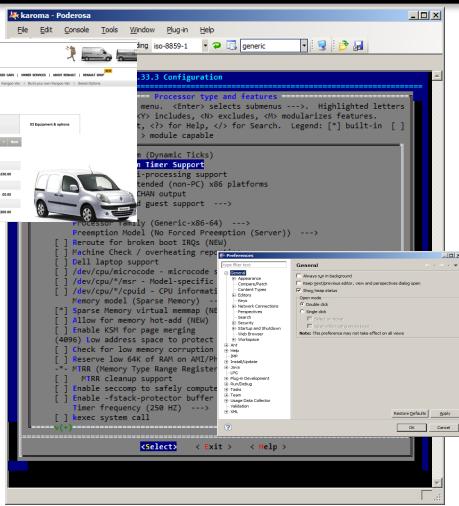


Why?

Problem: collecting videos is a key economic problem.

Solution: hundreds of video sequences with different characteristics.





Feature Model

Variability Realization Model (VRM)



✓ ✓
Configuration
(resolution model)
✗ ✗

Software Generator
(derivation engine)

A large graphic of three interlocking gears, rendered in a light purple color. To the right of the gears, there are three smaller versions of the same landscape image, each with a different color filter applied: a normal color image, a black and white (grayscale) image, and a sepia-toned image.

Old Approach



Developers

*modify
 N times*

N Configuration Files



Video Sequences Generator

generates



N Video Sequences



VM Approach

Developers and Domain Experts



*model
1 time*

VM interpreter, and
configuration files
customizer



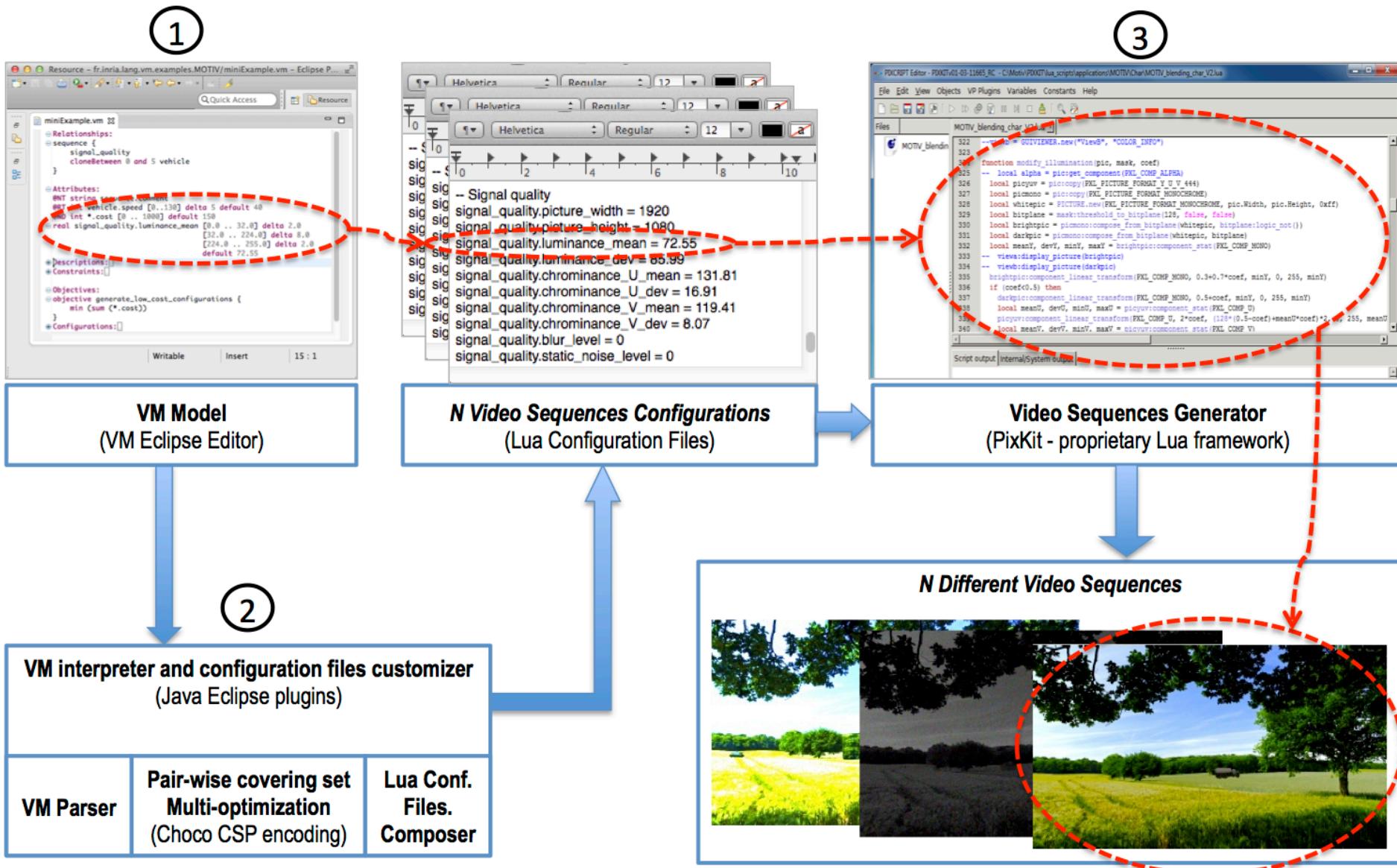
generates



VM model

Artifact(s)

Processing
tool(s)



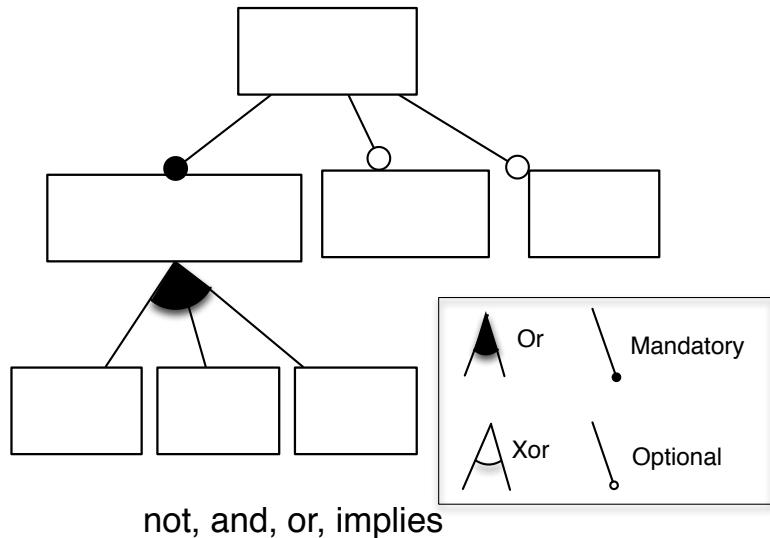
(configuration file)

```
-- 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)
```

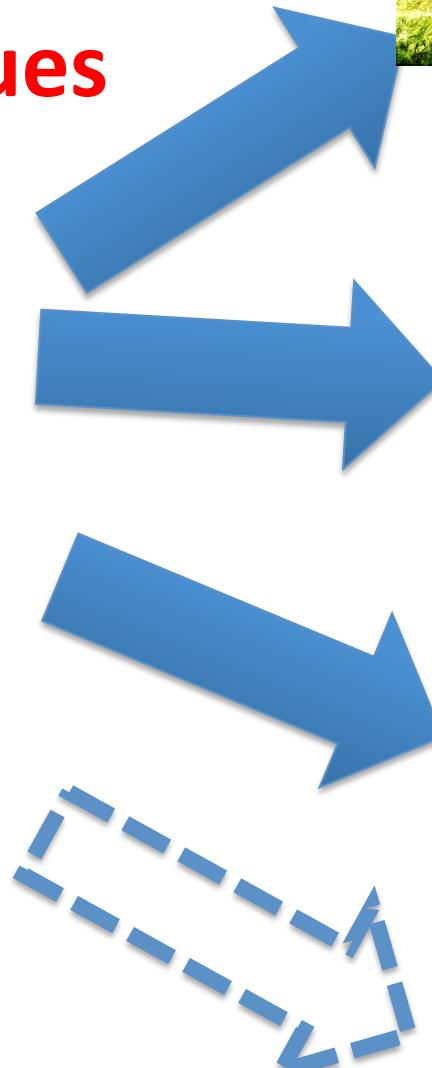
```
print("->Step9")
if (CFG.distractors.close_moving_vegetation~=0) then
    windvect5, precwindvect5, newwindvect5 =
generate_wind_vector_field2(workwidth, workheight, 256, 1, 1, 35, picnum,
precwindvect5, newwindvect5)
    windvectmul =
windvect5:mul(24*CFG.distractors.close_moving_vegetation)
    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(lfyv, depthmat*0.1)
    windvectcomp = VECT2D.new_from_matrices(MATRIX.addterm(lfvx, hfvx),
MATRIX.addterm(lfyv, 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)

We synthesize video sequence variants with variability techniques



Variability Models (feature models)



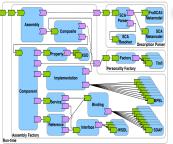
Take Away Messages

- Variability is everywhere
 - Like software is everywhere
 - Numerous artefacts, domains, kinds of systems are subject to customization
- Engineering variability can be hard
 - Exponential number of products
 - Basic or sophisticated techniques (than span your entire cursus) exist
- You're now aware of that!
 - #e1 Recognize variability
 - #e2 Re-visit your cursus and body of knowledge under the angle of variability
 - #e3 Apply state-of-the-art techniques

Reverse Engineering

Compose/Decompose
Configure
Analyse
Generate

Component Models



Dependencies Files



Source Code



Product descriptions



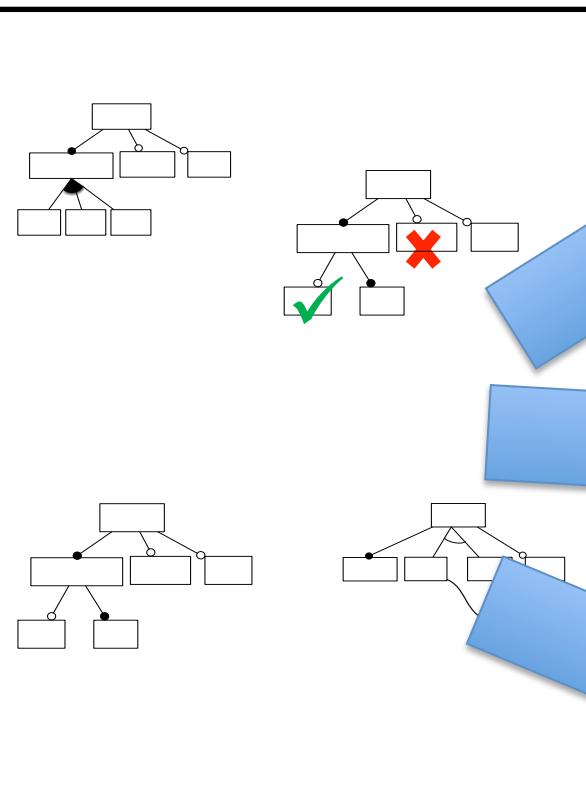
Regulatory Requirements



Web Configurators

FAMILIAR

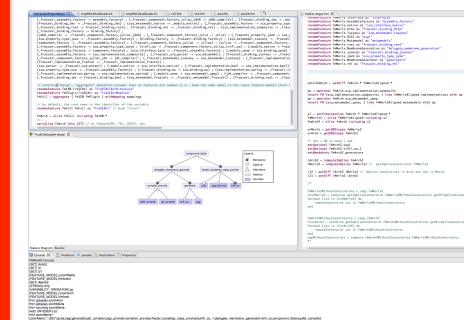
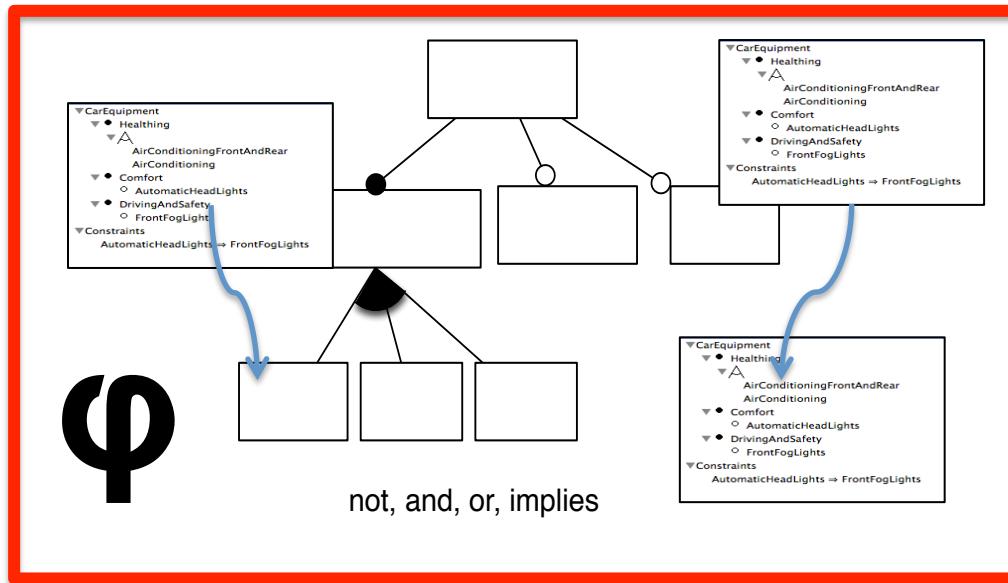
Applications: Softpedia, 100+ web configurators, video analysis, video protection, medical imaging, Wikipedia, plugin architectures, Linux, etc.



FAMILIAR

(FeAture Model script Language for manipulation and Automatic Reasoning)

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



« Fluent » API (Java, for an integration into JVM-based applications)
Dedicated language and environment (Web, Eclipse)

Applications: Softpedia, 100+ web configurators, video analysis, video protection, medical imaging, Wikipedia, architectures à plugins, Linux, etc.