Software Product Lines, Variability, and Configurations

Overview and Principles

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

UNIVERSITÉ DE RENNES 1

istic Informatique Électronique

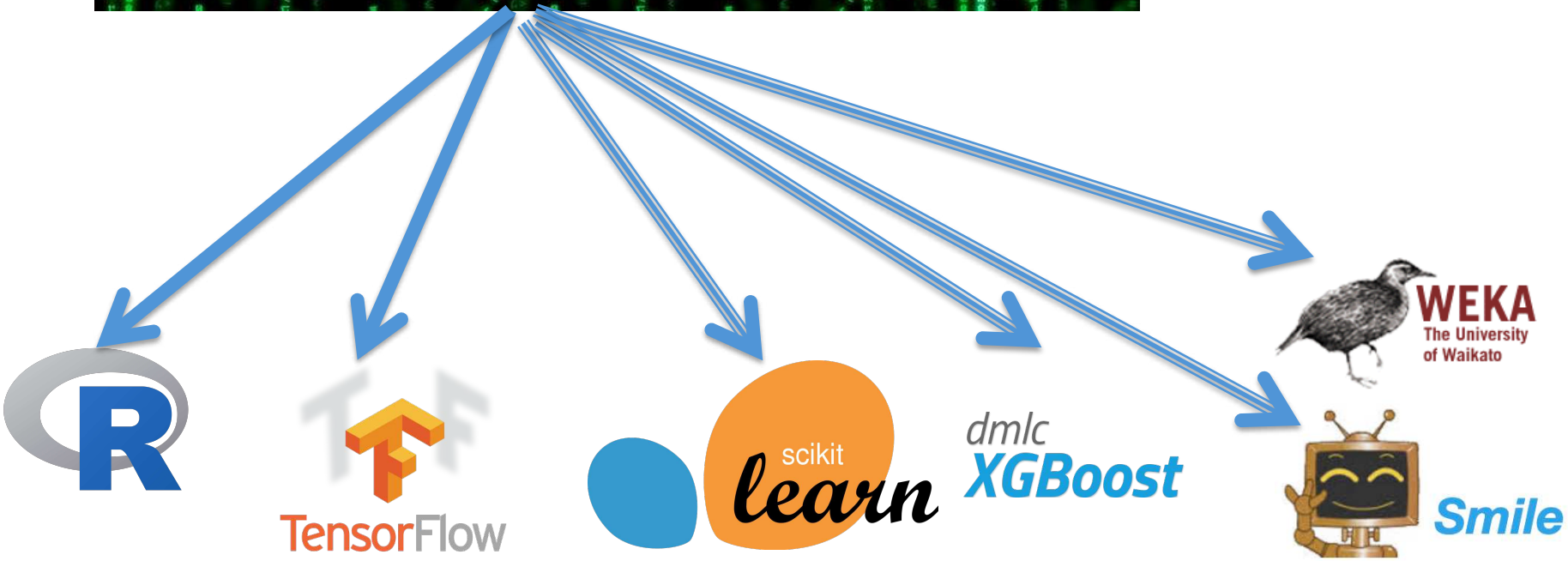
Material

https://github.com/acherm/teaching-MDE1920

http://teaching.variability.io

http://familiar.variability.io

MML Language



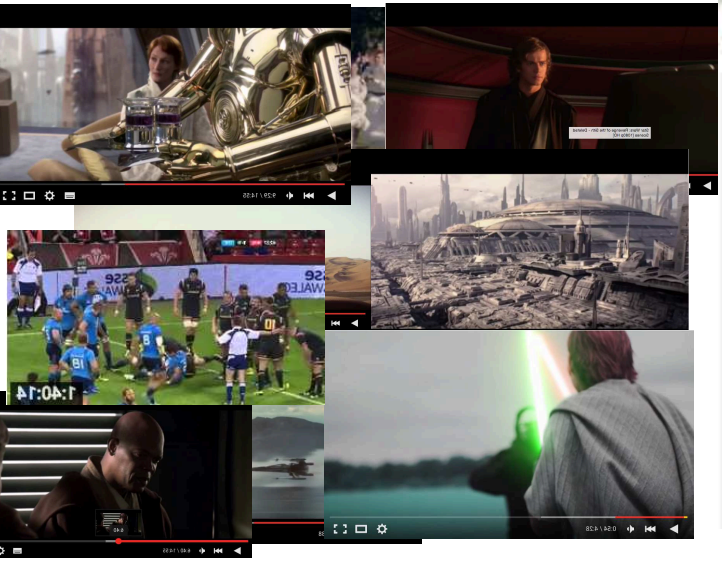
Variants and configurations

Generator
~ composition of video sequences

video
variants

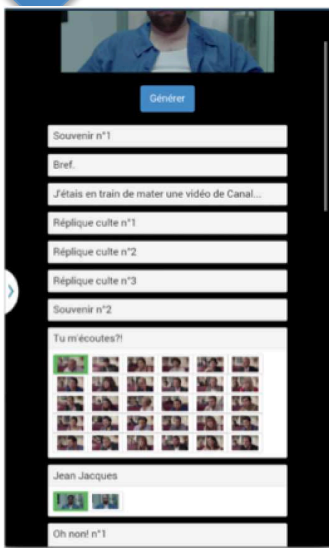
Generator
~ composition of video sequences

video variants



foo1.videogen

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

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

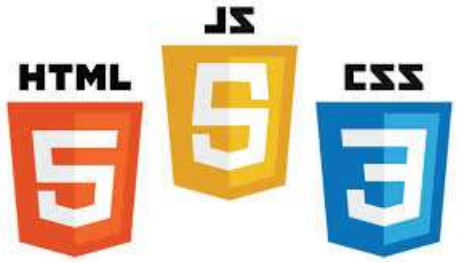


Website/online

- Random generation
- Configurator
- Game
- …

foo1.videogen

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

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



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

#2 How to transform models/programs?

#3 How to manage variability/variants?

#4 How do frameworks internally work?

Plan

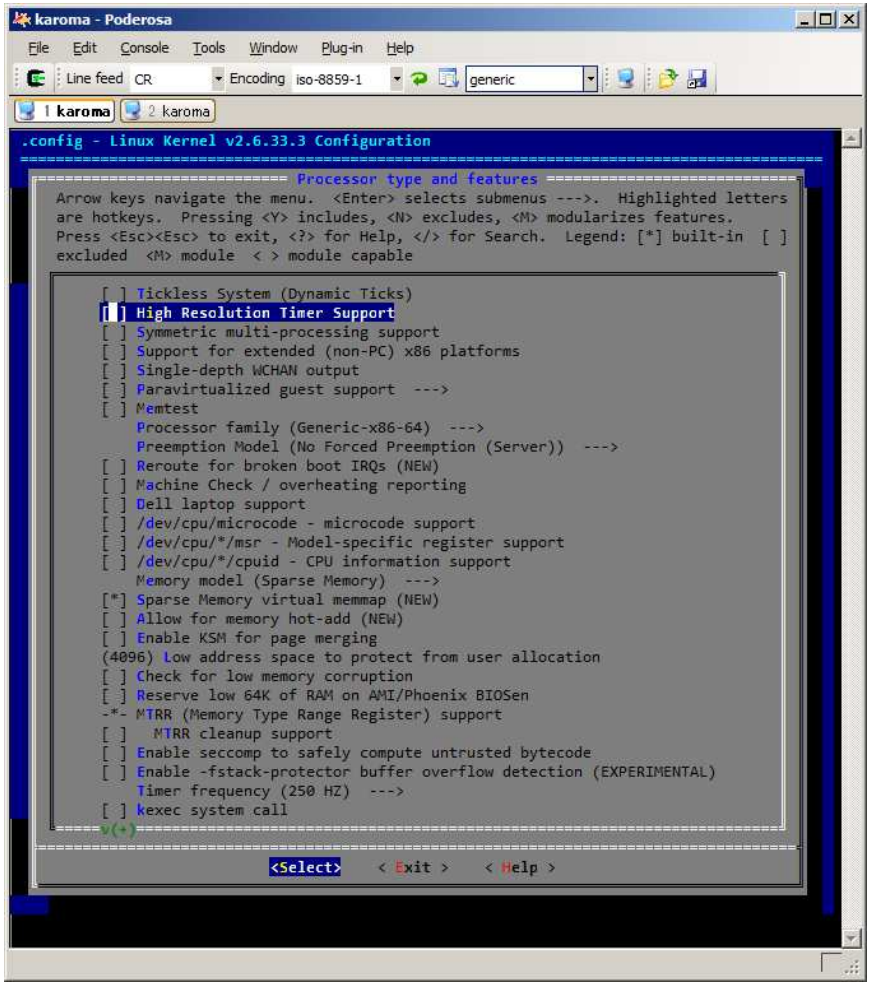
- Software product lines, configurable systems, and generators in the real-world
- From copy-and-paste to metamodeling
- The Jhipster case
  - An example of a real-world highly configurable system: how it is implemented and how we can model/test JHipster

Contract

- The idea of software product lines and variability
- Variability modeling
- Case studies

Software Product Line and Variability Engineering

Kernel Linux



Linux everywhere since **highly configurable**

```
config X86_X2APIC
        bool "Support x2apic"
        depends on X86_LOCAL_APIC && X86_64 && (IRQ_REMAP || HYPERVISOR_GUEST)
        ---help---
          This enables x2apic support on CPUs that have this feature.

          This allows 32-bit apic IDs (so it can support very large systems),
```

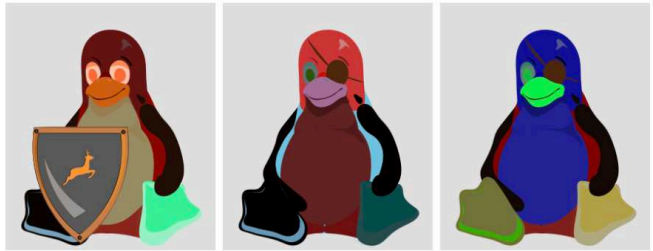
```
config IOSF_MBI
        tristate "Intel SoC IOSF Sideband support for SoC platforms"
        depends on PCI
        ---help---
          This option enables sideband register access support for Intel SoC
          platforms. On these platforms the IOSF sideband is used in lieu of
          MSR's for some register accesses, mostly but not limited to thermal
          and power. Drivers may query the availability of this device to
          determine if they need the sideband in order to work on these
          platforms. The sideband is available on the following SoC products.
```

Kconfig files/doc

```
#
# Processor type and features
#
# CONFIG_ZONE_DMA is not set
# CONFIG_SMP is not set
# CONFIG_X86_FEATURE_NAMES is not set
# CONFIG_X86_FAST_FEATURE_TESTS is not set
CONFIG_X86_X2APIC=y
CONFIG_X86_MPPARSE=y
CONFIG_GOLDFISH=y
# CONFIG_INTEL_RDT_A is not set
# CONFIG_X86_EXTENDED_PLATFORM is not set
CONFIG_IOSF_MBI=m
CONFIG_IOSF_MBI_DEBUG=y
CONFIG_X86_SUPPORTS_MEMORY_FAILURE=y
# CONFIG_SCHED_OMIT_FRAME_POINTER is not set
```

.config

https://github.com/diverse-project/tuxart



You Retweeted

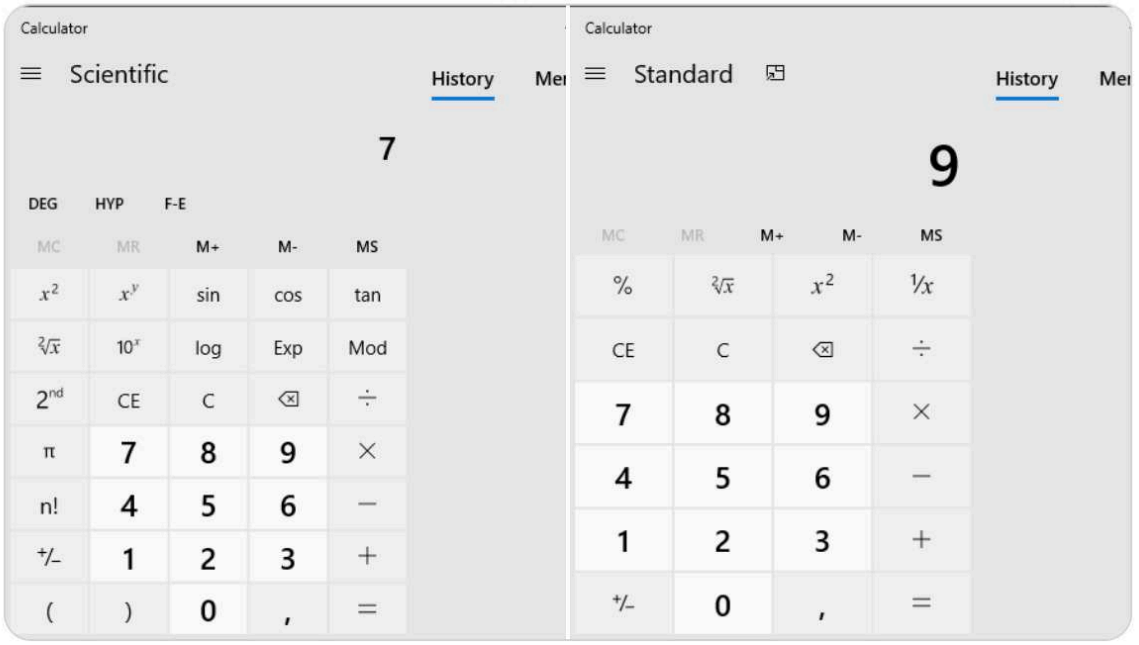
Thomas Thüm
@ThomasThuem

How much is 1 + 2 * 3?

According to @Windows 10, the result depends on the mode that you have chosen!!!!!

The Calculator app gives the correct result in Scientific mode and a wrong result with the default mode called Standard.

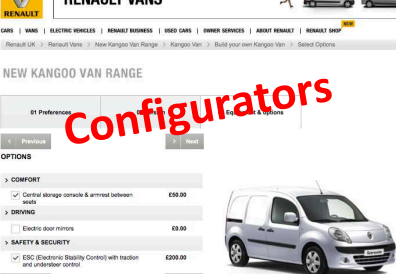
I bet this caused thousands of wrong calculations!



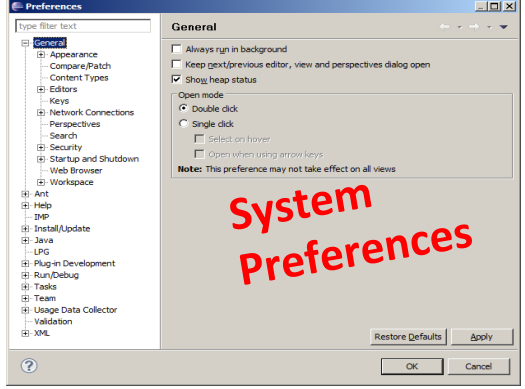
11:46 PM · Oct 9, 2019 · Twitter Web App

```
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)
```

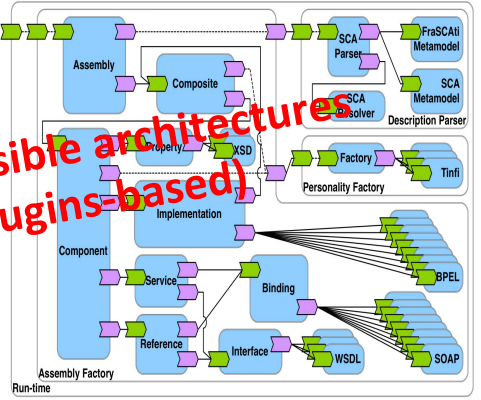
	 Starter	 Home Premium Upgrade $119.99* Buy	 Professional Upgrade $199.99* Buy	 Ultimate Upgrade $219.99* Buy
Communication				
Bluetooth support	✓	✓	✓	✓
Join a homegroup	✓	✓	✓	✓
Internet Explorer 8	✓	✓	✓	✓
View Available Networks	✓	✓	✓	✓
Windows Connect Now (WCN)	✓	✓	✓	✓
Create a homegroup		✓	✓	✓
Location 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		✓	✓	✓



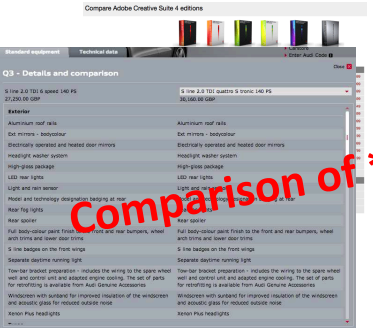
Configurators



System Preferences

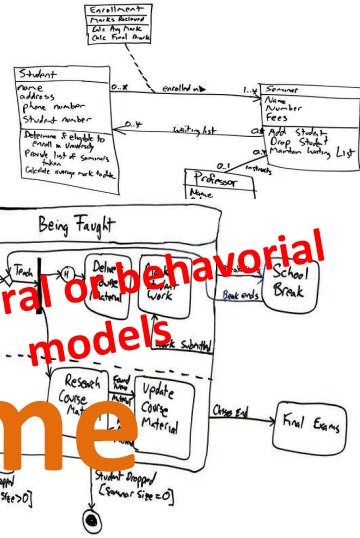


Extensible architectures (eg plugins-based)

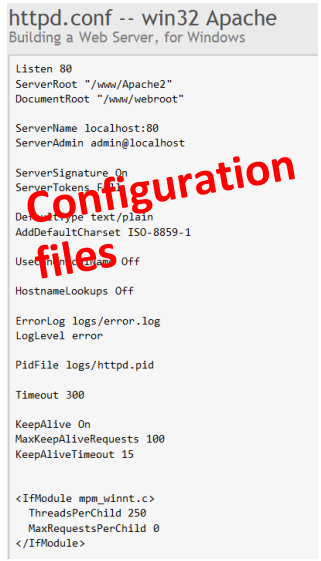


Comparison of *

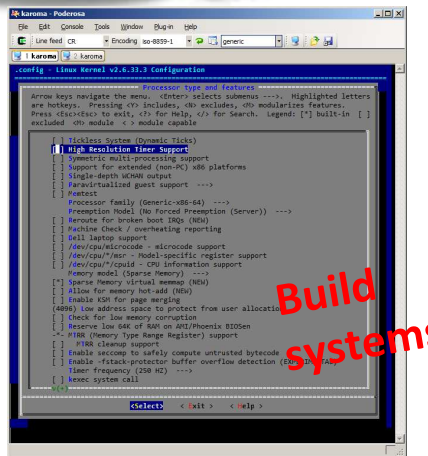
External Variability
Internal Variability
Variability @ run.time



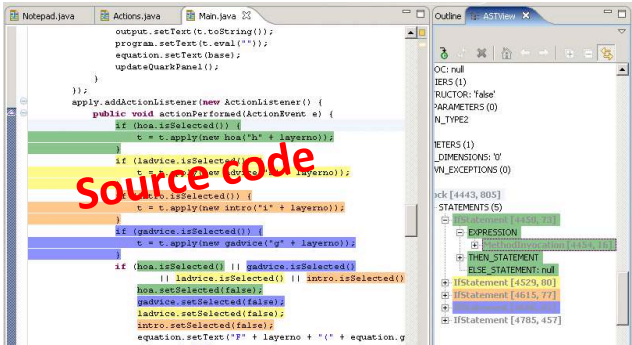
Structural or behavorial models



Configuration files



Build systems



Source code

(a) Variant #1 of video sequence

(b) Variant #2 of video sequence

(c) Variant #3 of video sequence

(d) Variant #4 of video sequence

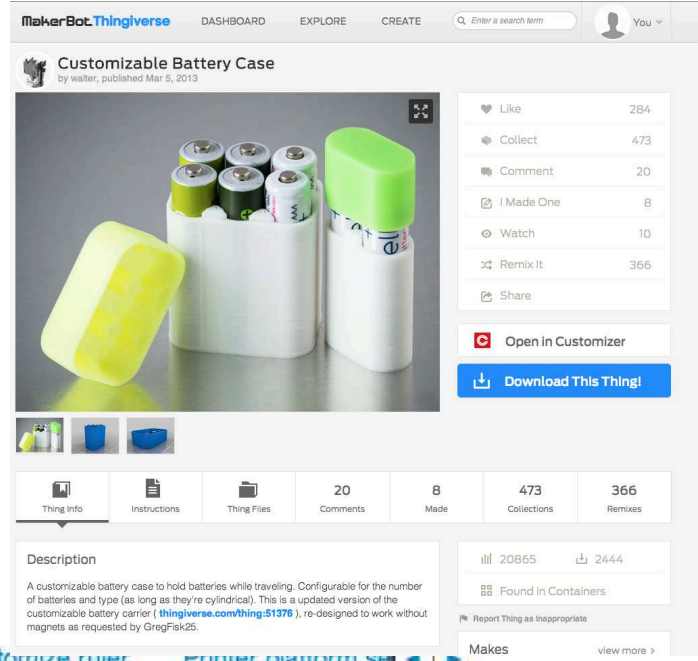
(e) Variant #5 of video sequence

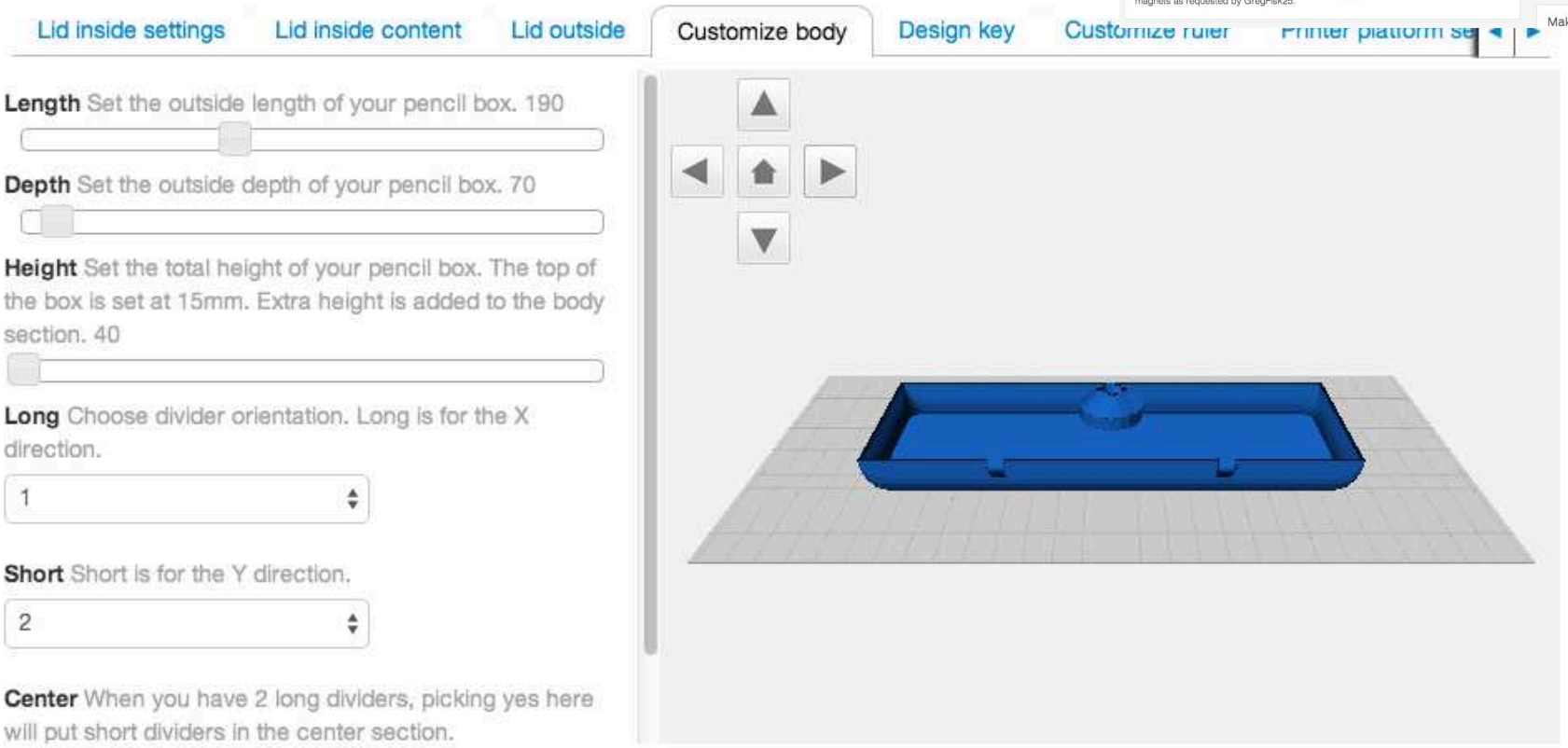
(f) Variant #6 of video sequence

Figure 1: Six variants of video sequences synthesized with ViViD

```
/* [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





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

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

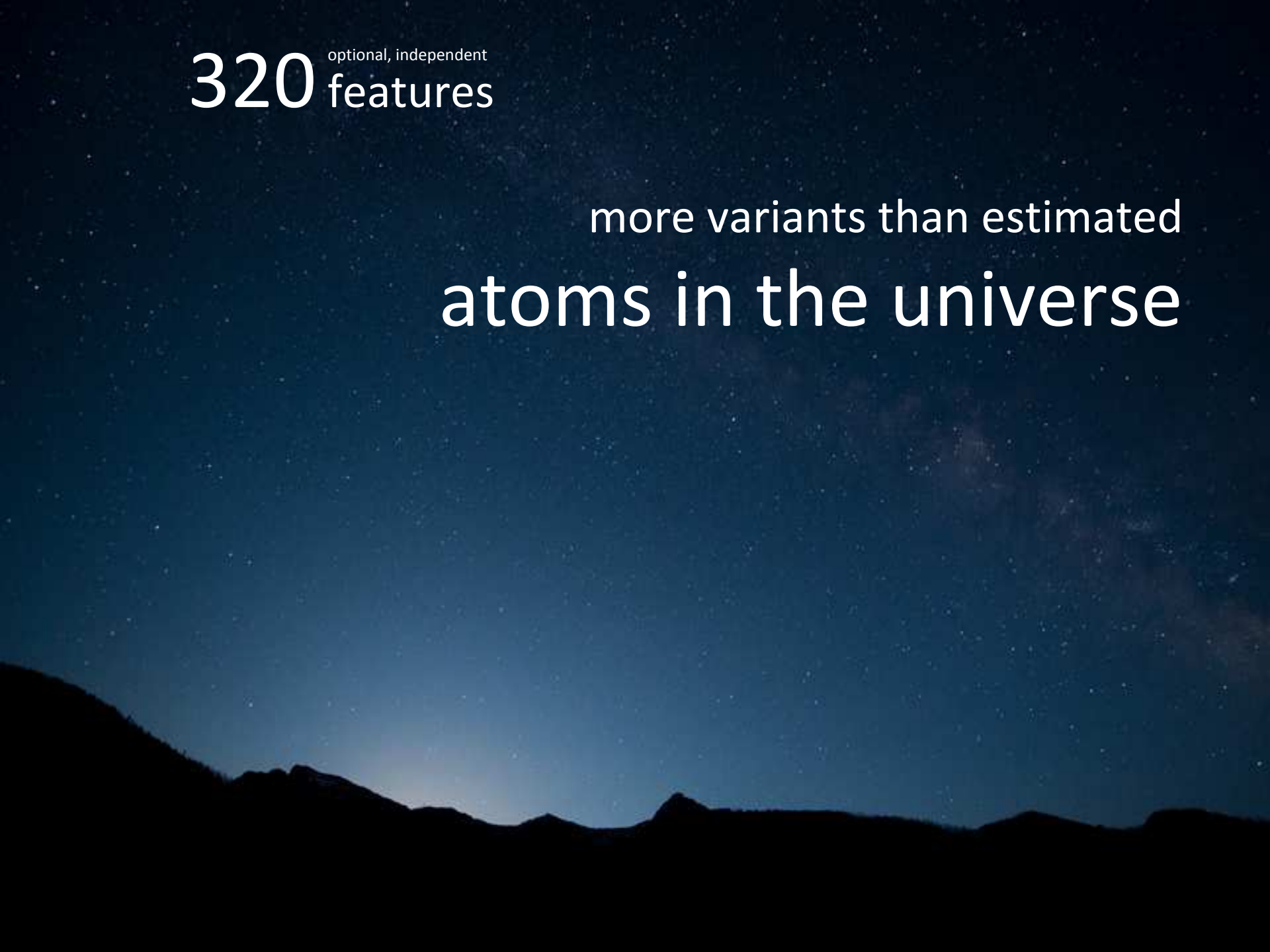
Give three examples of software product lines (also called configurable systems or variability-intensive systems)


Variability = Complexity

33 optional, independent features

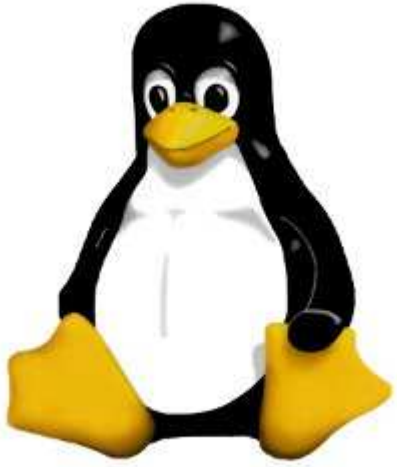
a unique variant for every

person on this planet


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

2000 features

16000 features

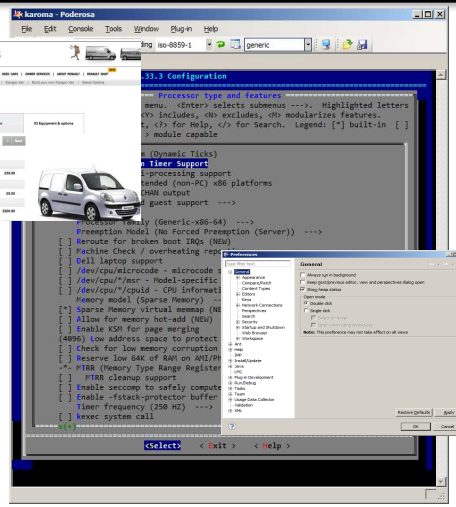


Linux Kernel

$\approx 10^{6000}$ configurations

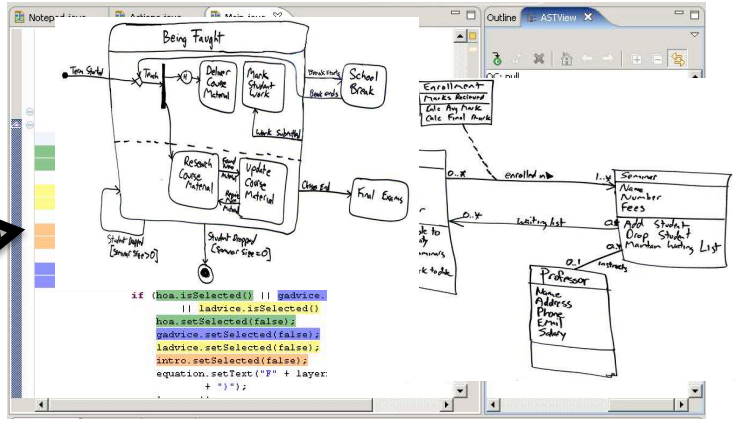
$\approx 10^{80}$ is the estimated number of atoms in the universe

$\approx 10^{40}$ is the estimated number of possible chess positions

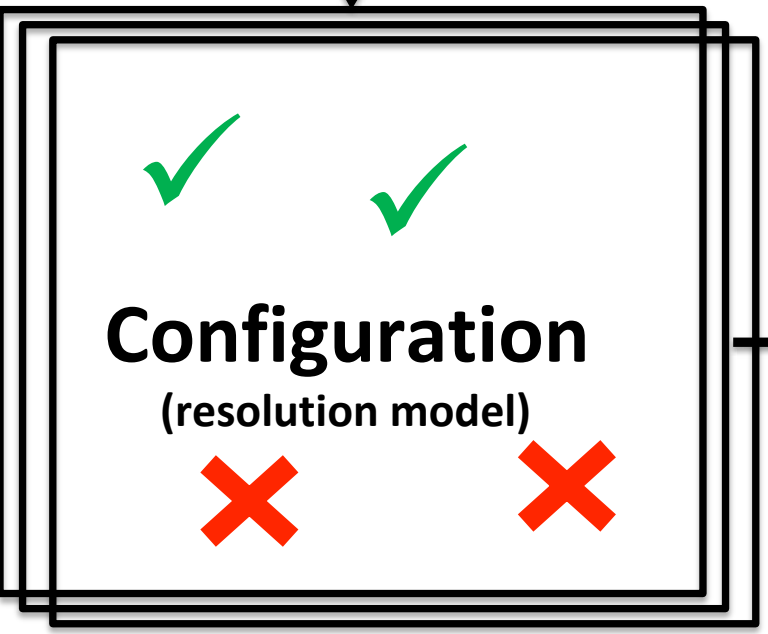


Variability Abstraction Model (VAM)

Variability Realization Model (VRM)



Domain Artefacts (e.g., models)



Configuration

(resolution model)

Software Generator

(derivation engine)

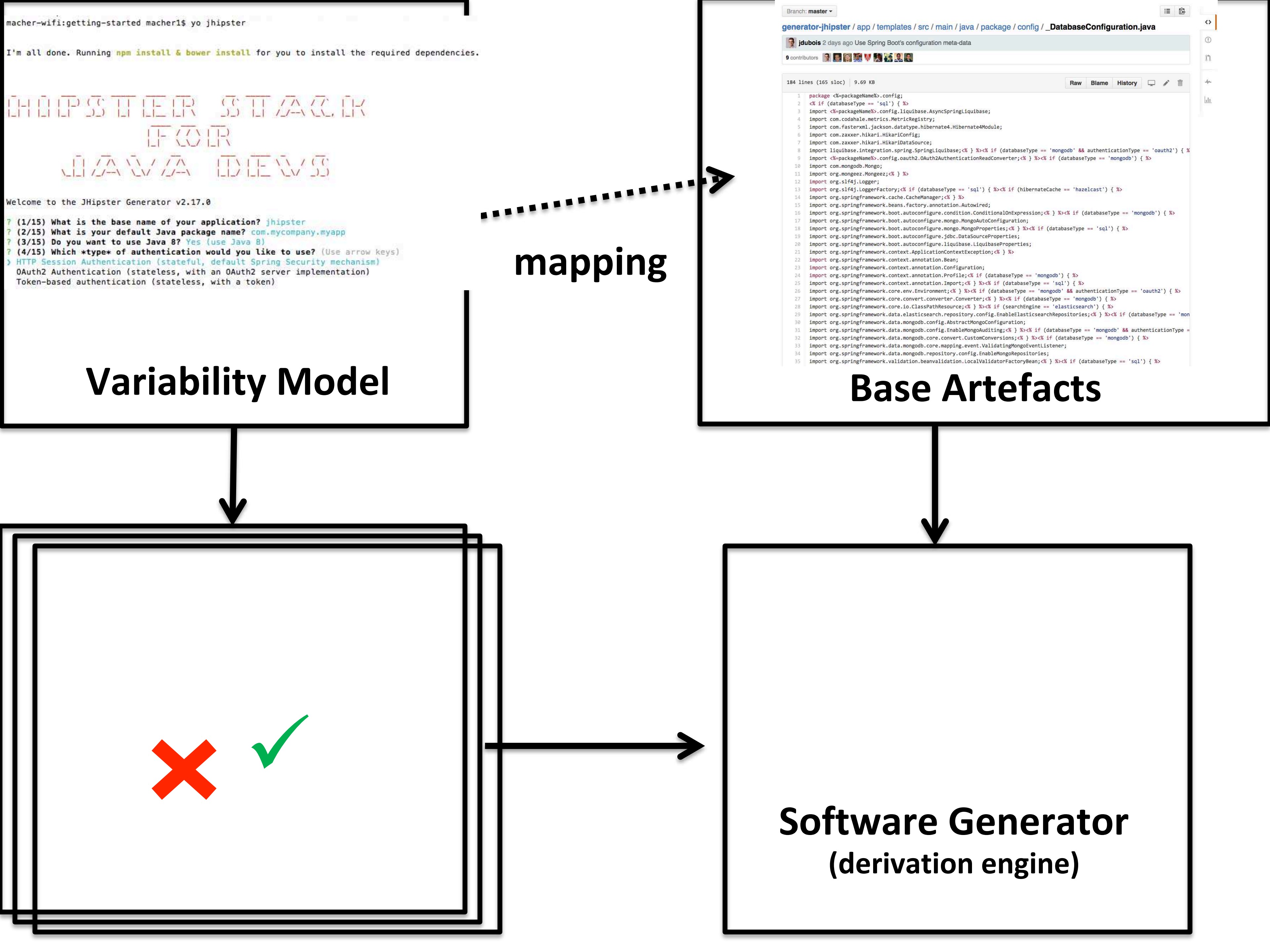
jdubois 2 days ago Use Spring Boot's configuration meta-data

9 contributors 

184 lines (165 sloc) | 9.69 KB

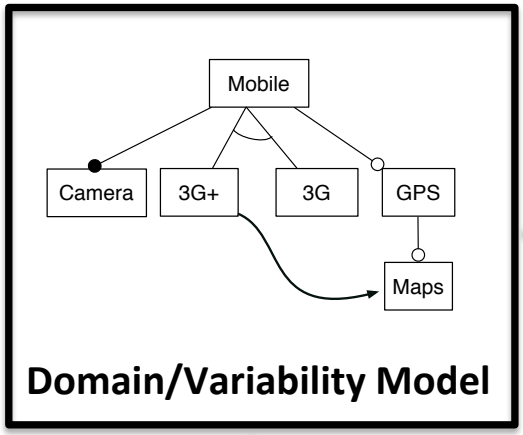
Raw | Blame | History

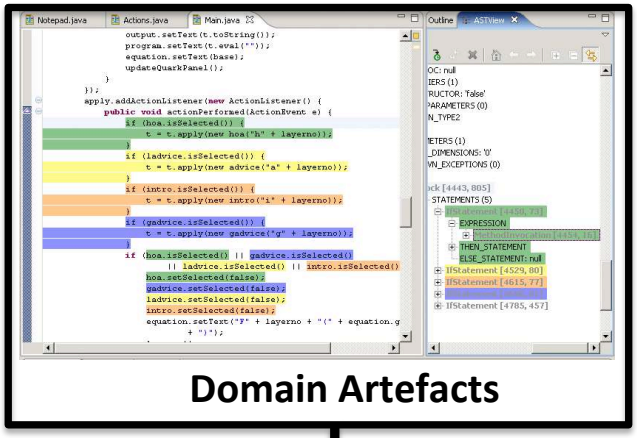
```
1  package <%=packageName%>.config;
2  <% if (databaseType == 'sql') { %>
3  import <%=packageName%>.config.liquibase.AsyncSpringLiquibase;
4  import com.codahale.metrics.MetricRegistry;
5  import com.fasterxml.jackson.datatype.hibernate4.Hibernate4Module;
6  import com.zaxxer.hikari.HikariConfig;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



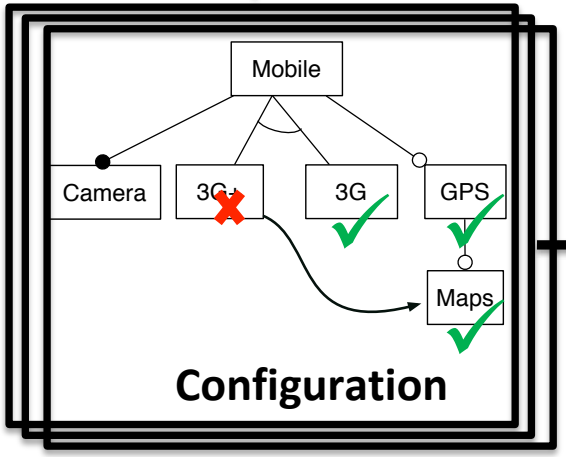
macher-wifi:getting-started macher1$ yo jhipster
I'm all done. Running npm install & bower install for you to install the required dependencies.
JHIPSTER STACK
FOR
JAVA DEVS
Welcome to the JHipster Generator v2.17.0
? (1/15) What is the base name of your application? jhipster
? (2/15) What is your default 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)
Variability Model
mapping
Branch: master
generator-jhipster / app / templates / src / main / java / package / config / _DatabaseConfiguration.java
jdubois 2 days ago Use Spring Boot's configuration meta-data
9 contributors
184 lines (165 sloc) 9.69 KB
Raw
Blame
History
Base Artefacts
Software Generator
(derivation engine)

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)

« variability »

Is it really new?

Command Line Options

```
x264 --quiet
     --no-progress
     --no-asm
     --rc-lookahead 60
     --ref 9
     -o trailer_480p24.x264
     trailer_2k_480p24.y4m
```

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     }
```

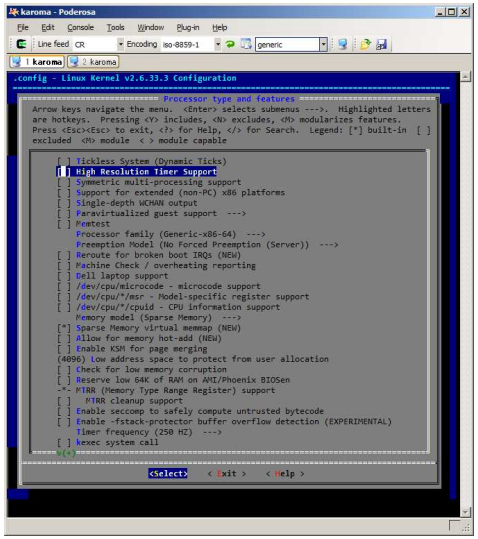
Global configuration

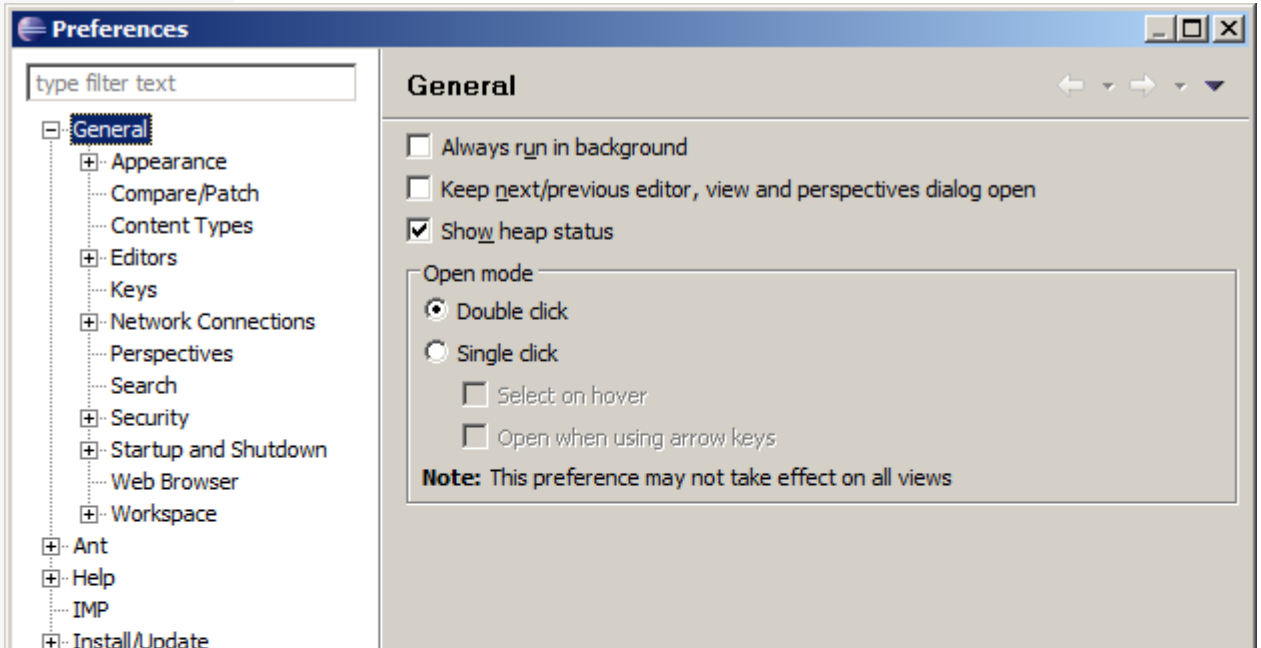
```
class Config {
  public static boolean isLogging = false;
  public static boolean isWindows = false;
  public static boolean isLinux = true;
}
class Main {
  public void foo() {
    if (isLogging)
      log(„running foo()“);
    if (isWindows)
      callWindowsMethod();
    else if (isLinux)
   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

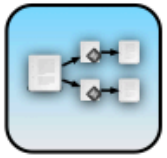
Code Cloning (example, Linux driver)

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

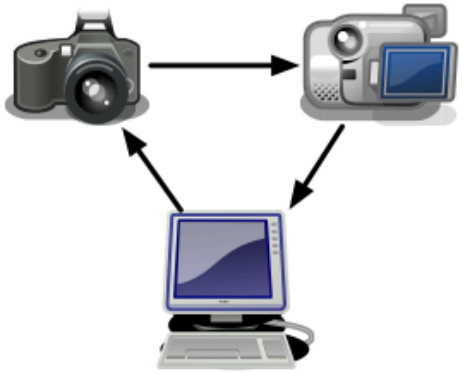
Clone existing code and fix low level platform interaction

+ 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

Template Method

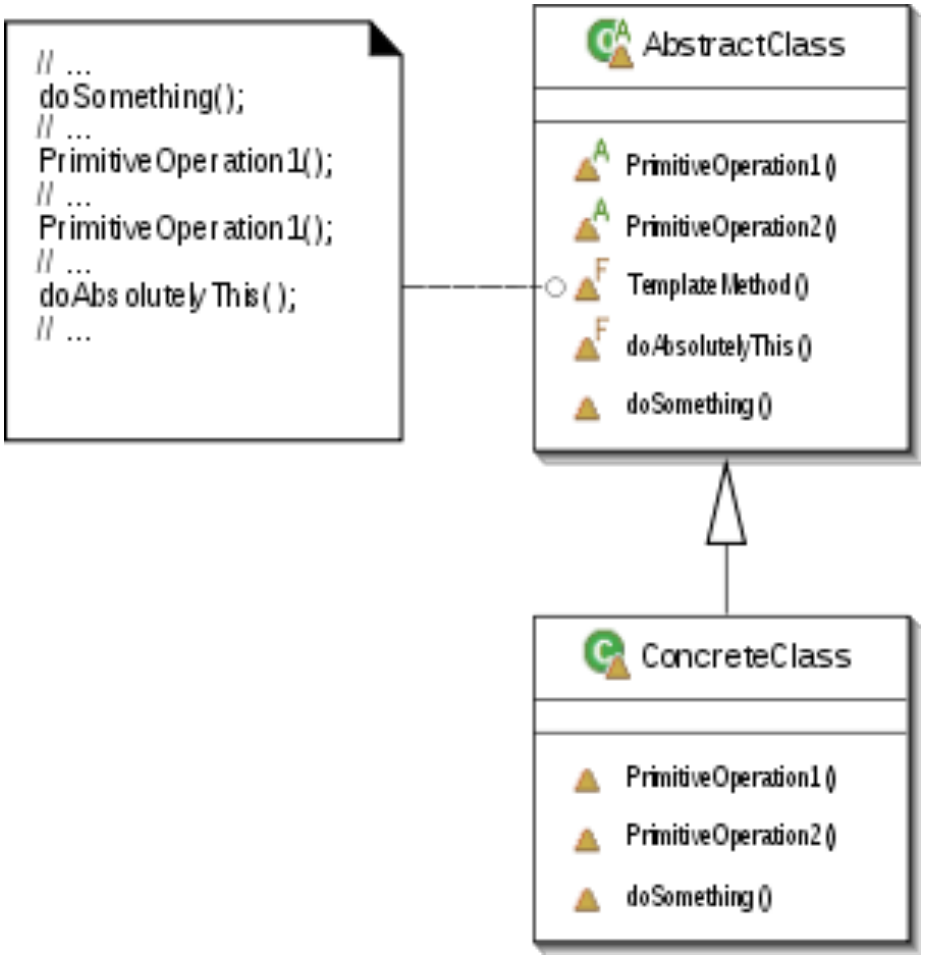
Factory

Strategy

Decorator

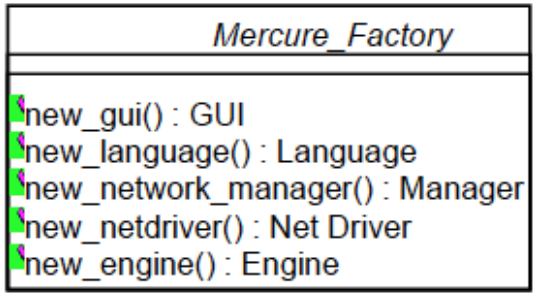
....

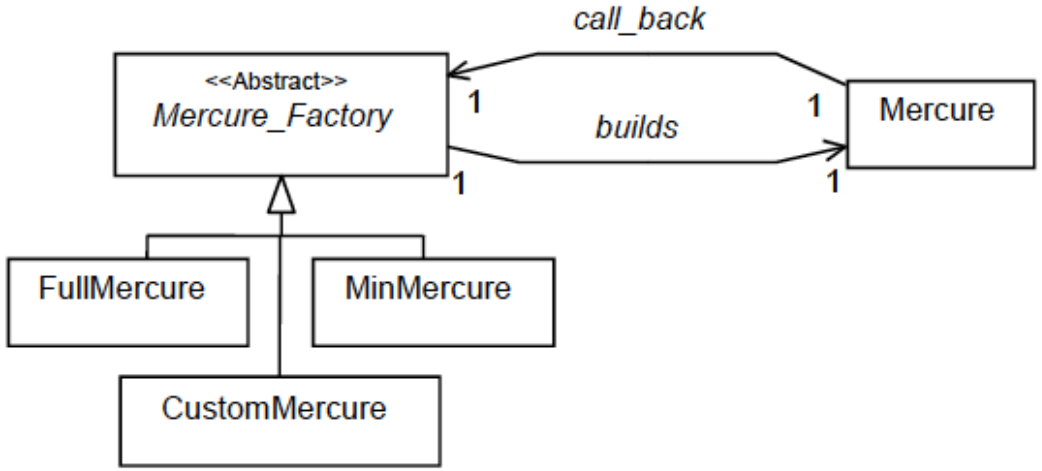
Template Method

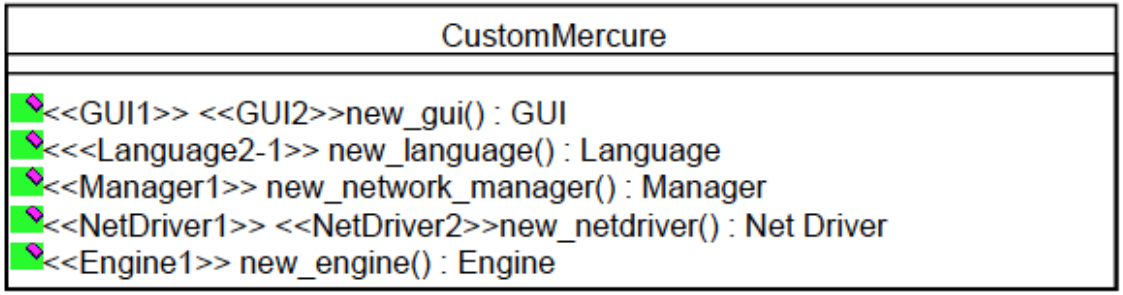


The decision model

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







API

Framework

Plugin-based systems

(Active) Annotations

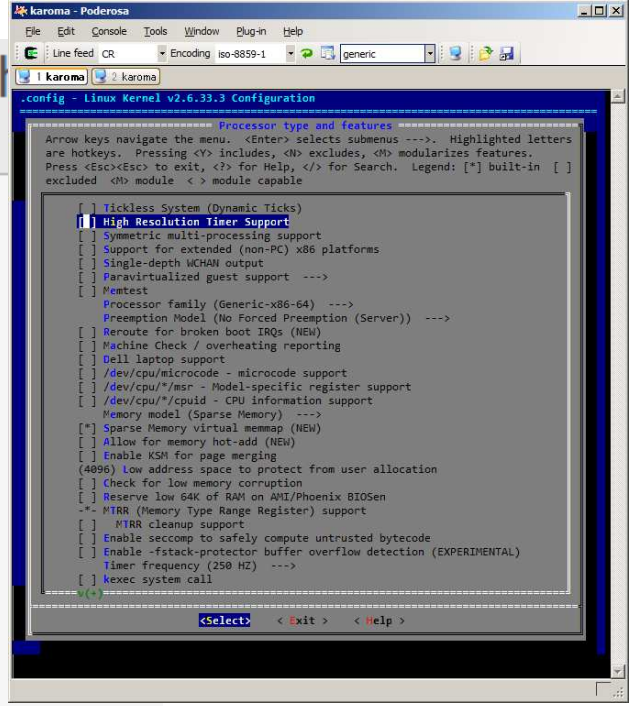
can have parameters

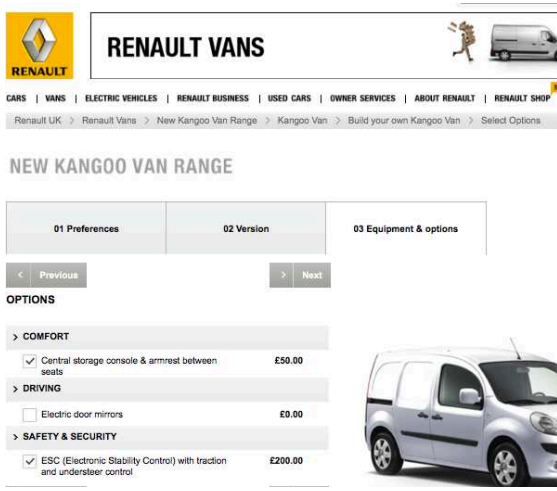
Metamodeling and Domain-Specific Languages

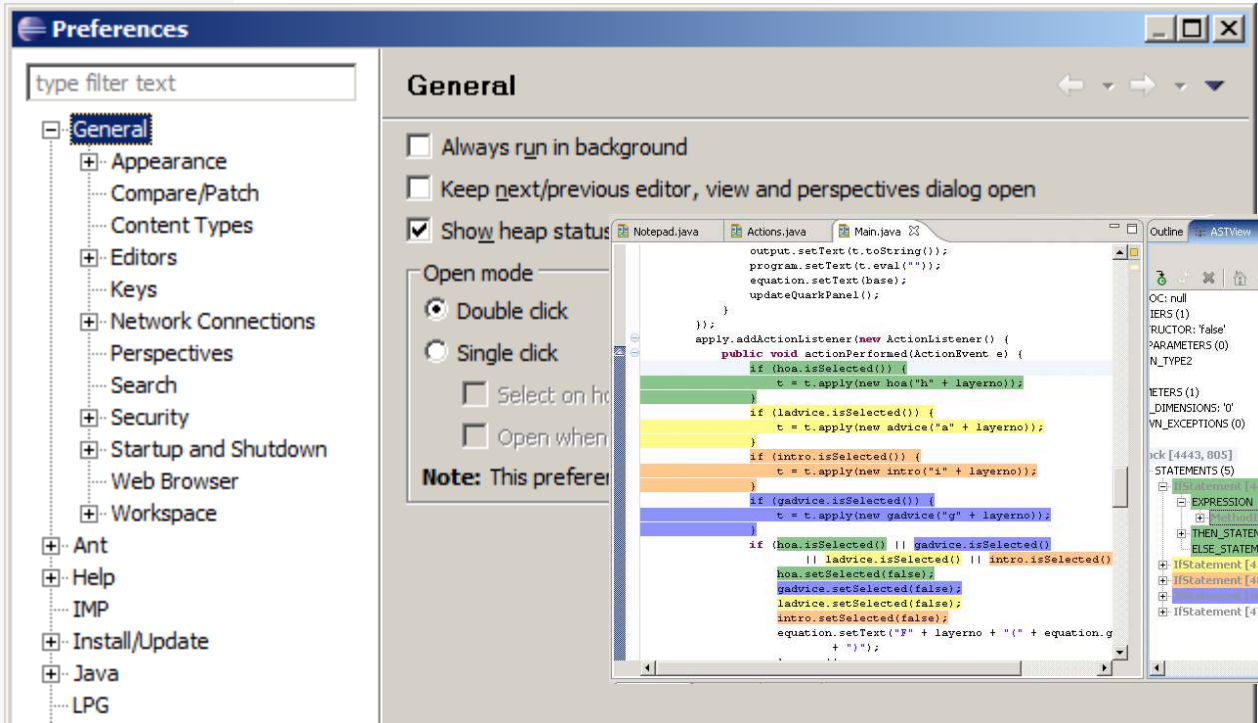
httpd.conf -- win32 Apach

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>
```



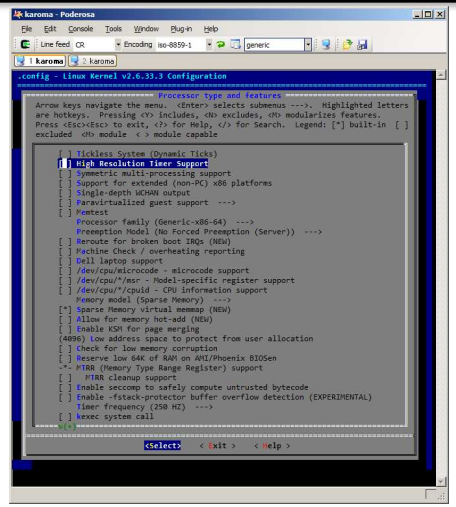




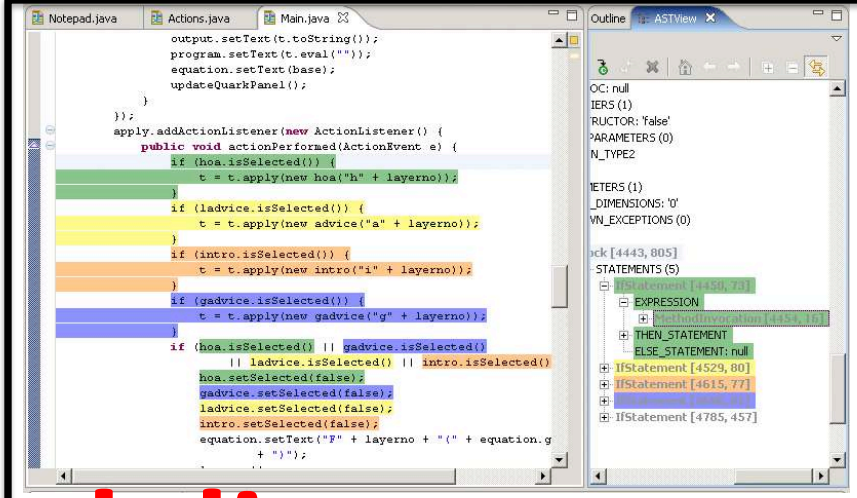

Unused flexibility


Illegal variant

Feature Models



Variability Model



Domain Artefacts (e.g., source code)

Modeling variability is crucial

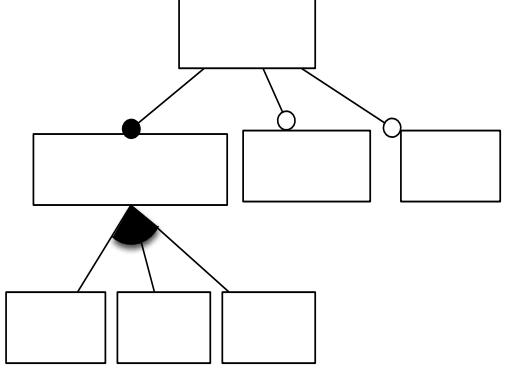
Configuration

Software Generator


Unused flexibility

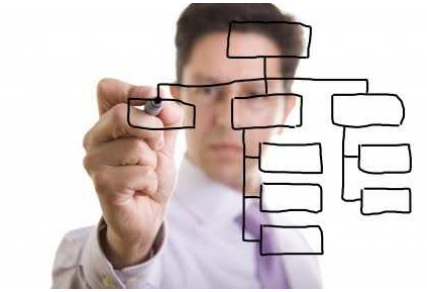

Illegal variant

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

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

- Packs d'équipements
- 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

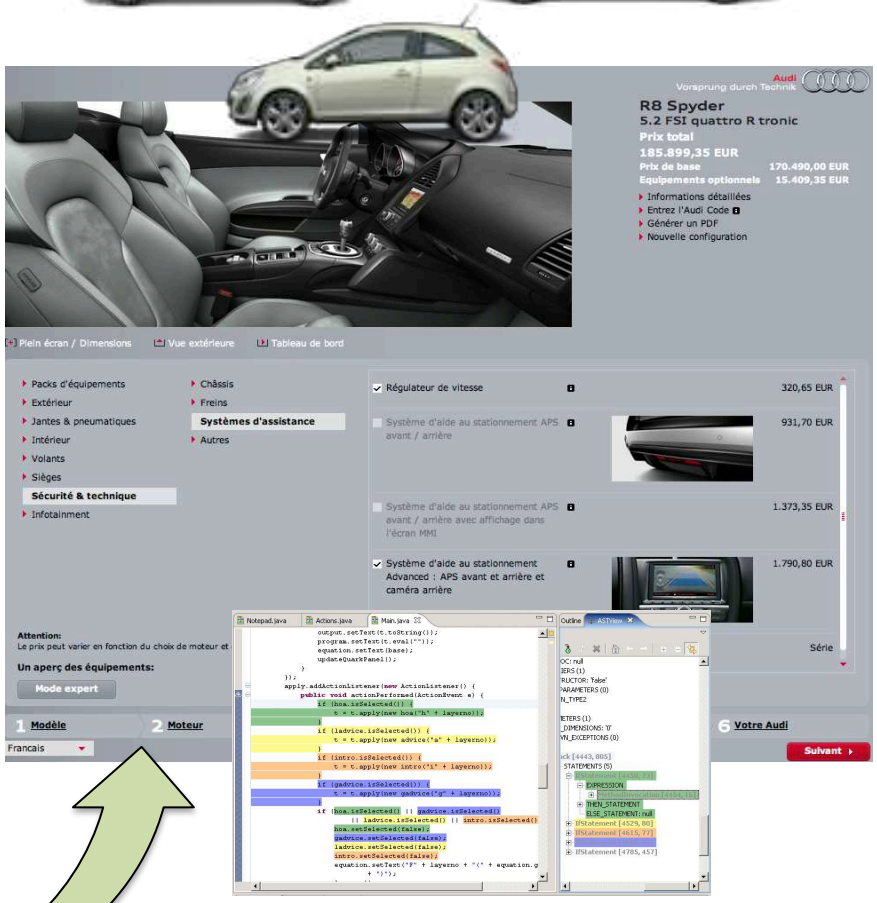
Réinitialiser la sélection

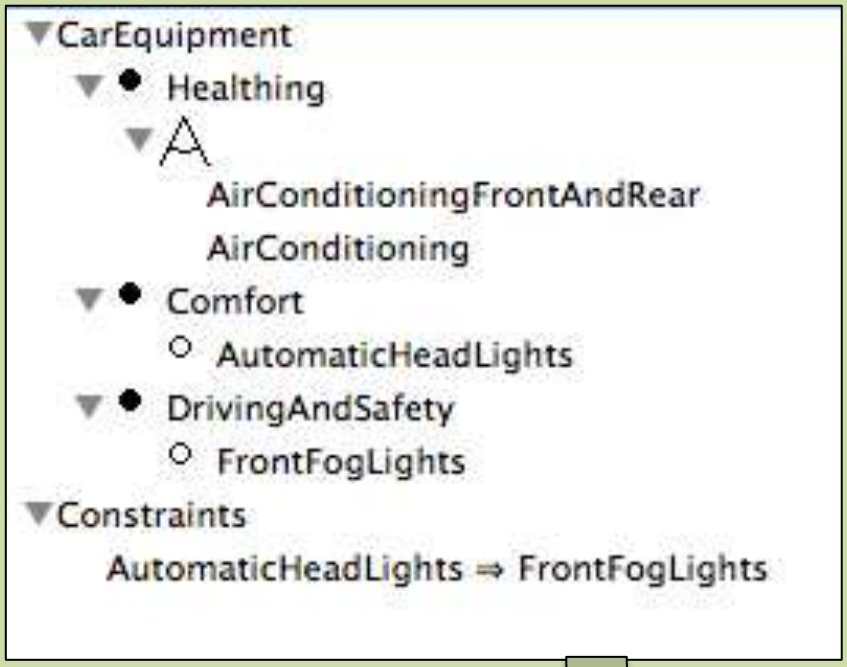
1 Modèle | 2 Moteur | 3 Extérieur | 4 Intérieur | 5 Option | 6 Votre Audi

Francais

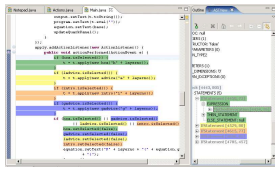
Suivant

Feature Models





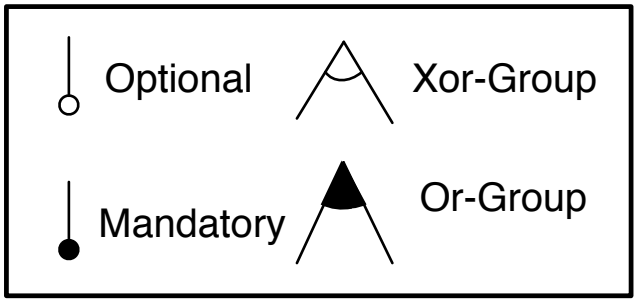
Feature Models (Background)



Hierarchy: rooted tree

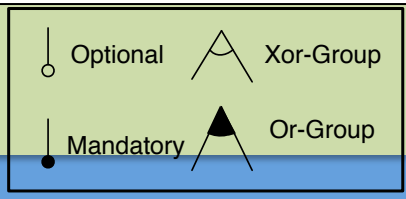
Variability:

- 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



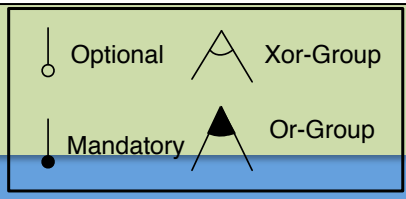
Hierarchy + Variability
=
set of valid configurations

configuration = set of features selected

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

SPLC12Scripts Model | SPLC12Scripts.config

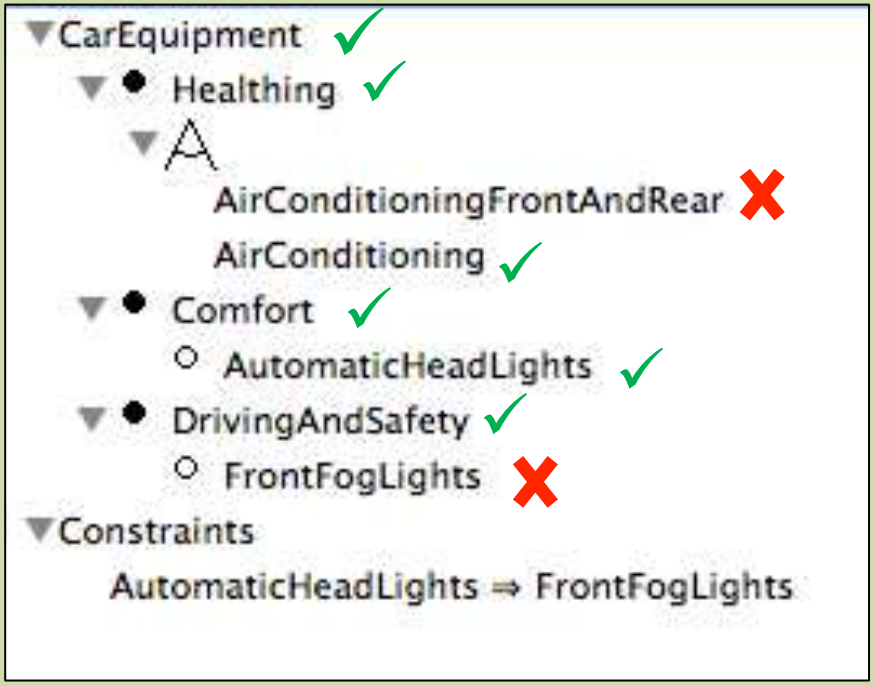
- CarEquipment (valid, 1 possible configurations)
  - Healthing
    - AirConditioningFrontAndRear
    - AirConditioning
  - Comfort
    - AutomaticHeadLights
  - DrivingAndSafety
    - FrontFogLights

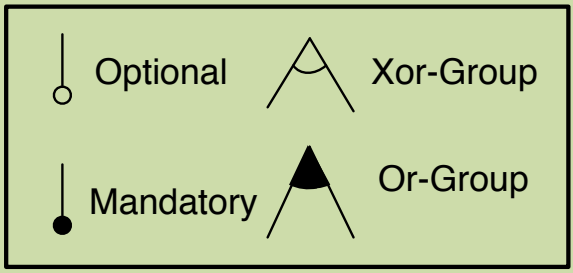


Hierarchy + Variability
=
set of valid configurations

configuration = set of features selected

{CarEquipment, Comfort, DrivingAndSafety, Healthing, AirConditioning}

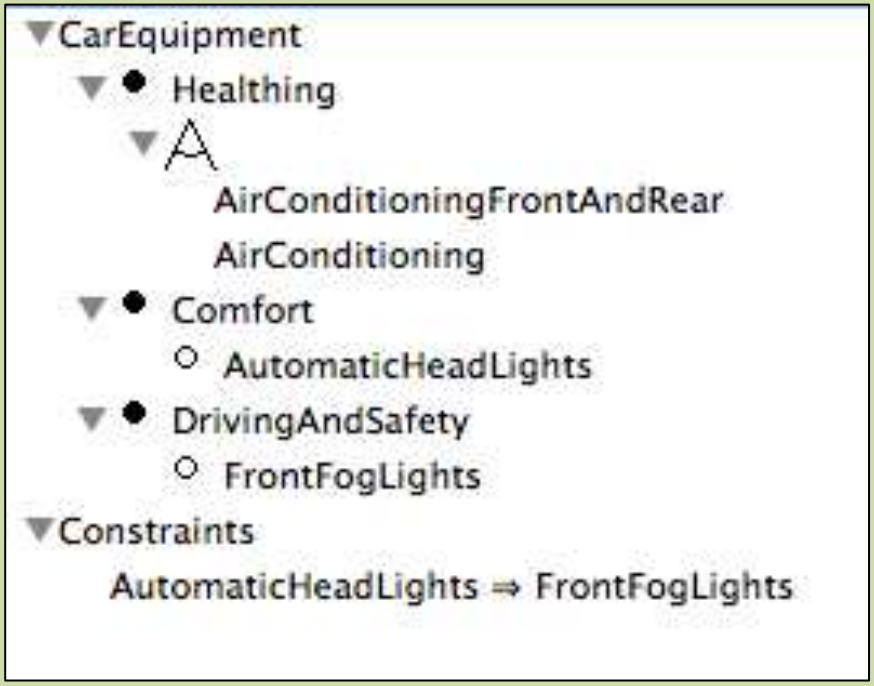


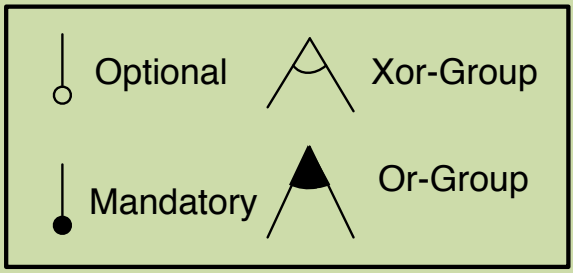


Hierarchy + Variability
=
set of valid configurations

configuration = set of features selected

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



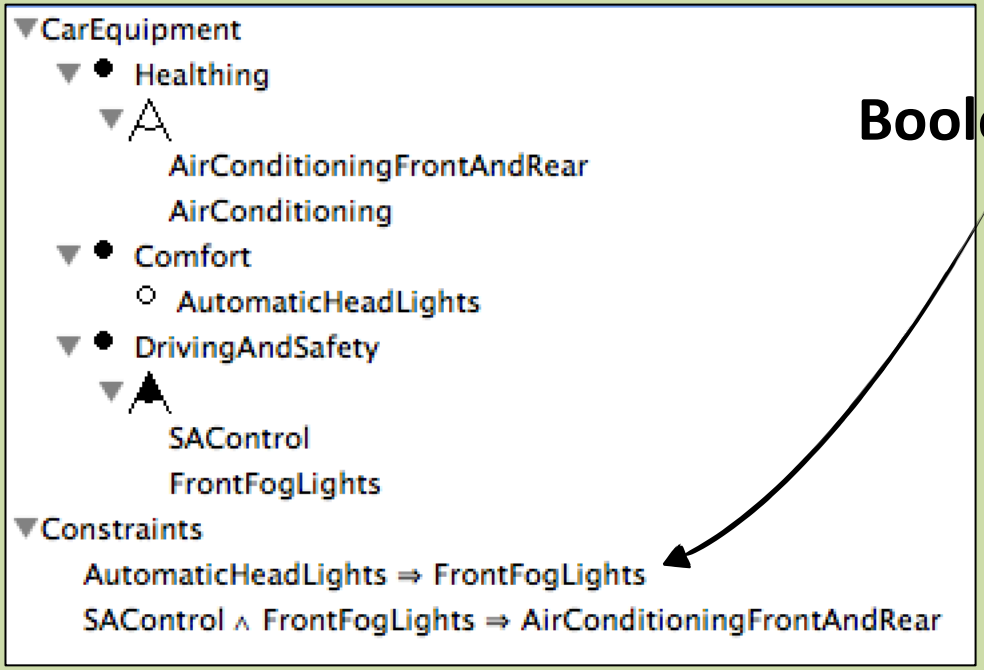


Hierarchy + Variability
=
set of valid configurations

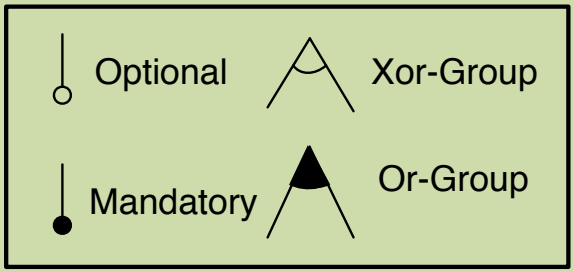
{CarEquipment, Comfort, DrivingAndSafety, Healthing}

X

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



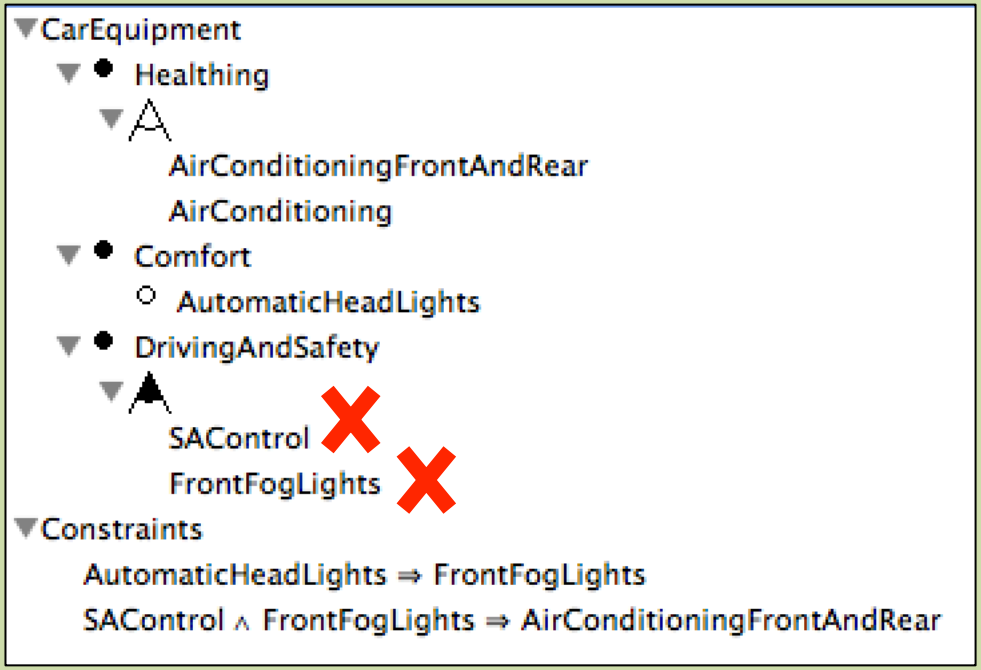
Boolean logic: ^, v, not, implies

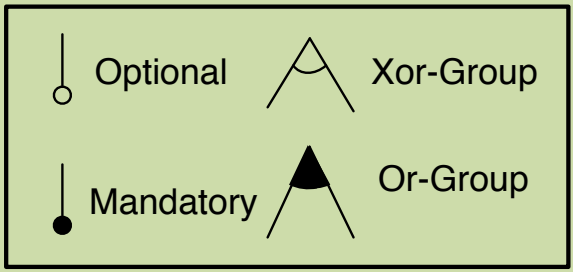


Hierarchy + Variability

=

set of valid configurations





Hierarchy + Variability

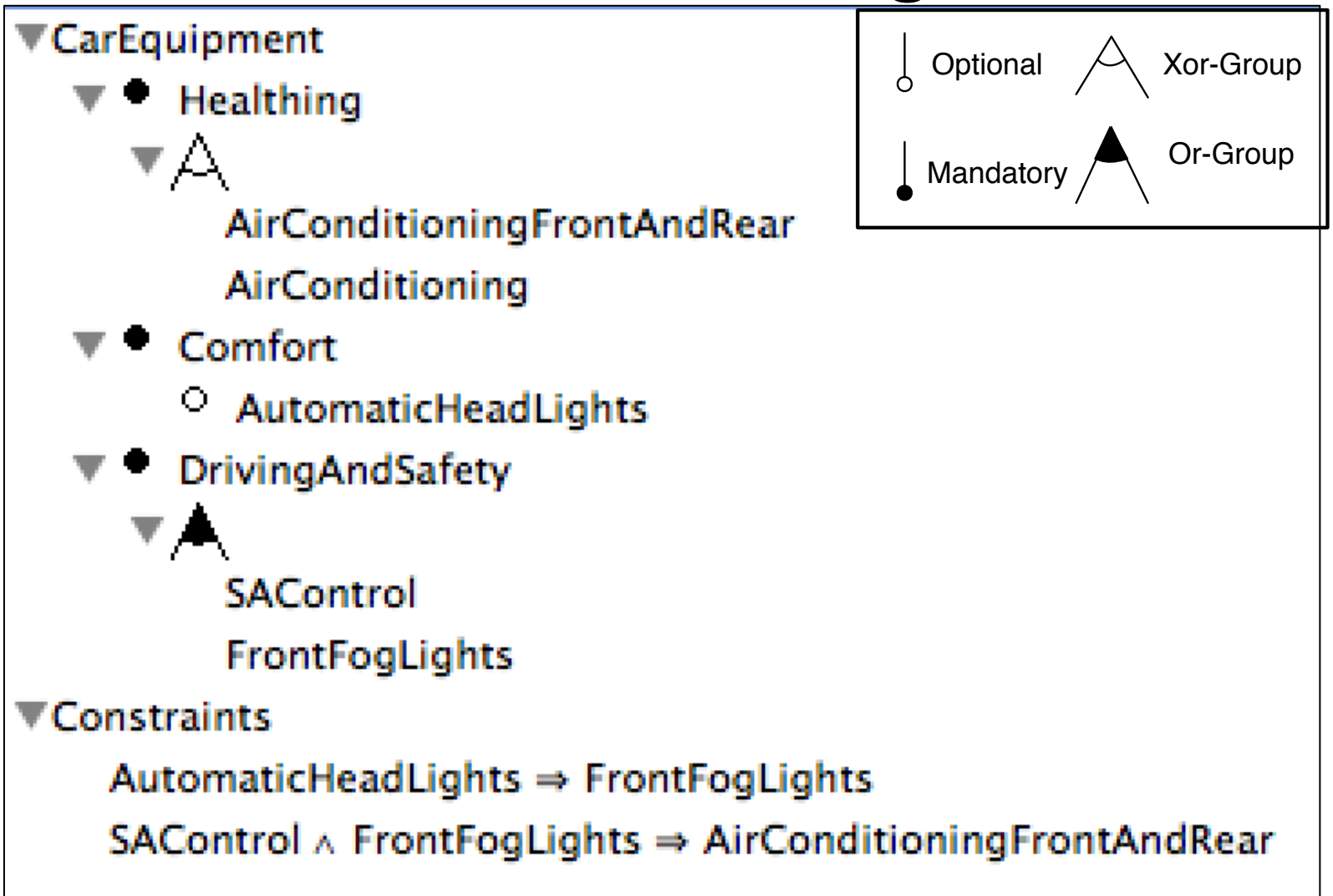
=

set of valid configurations

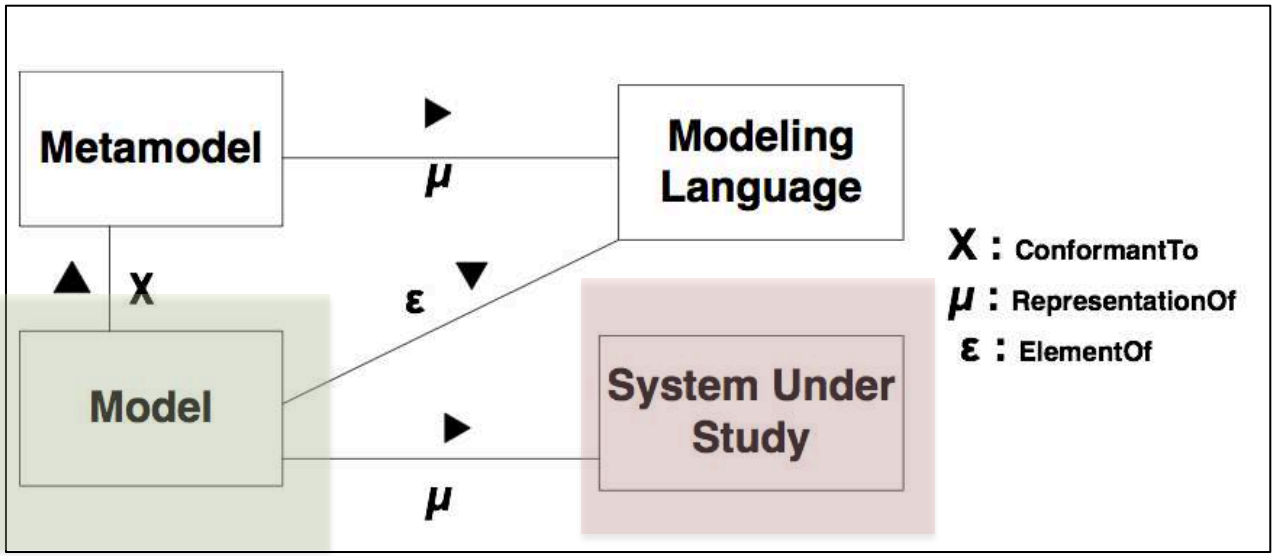
Or-group: at least one!

Quizz Time

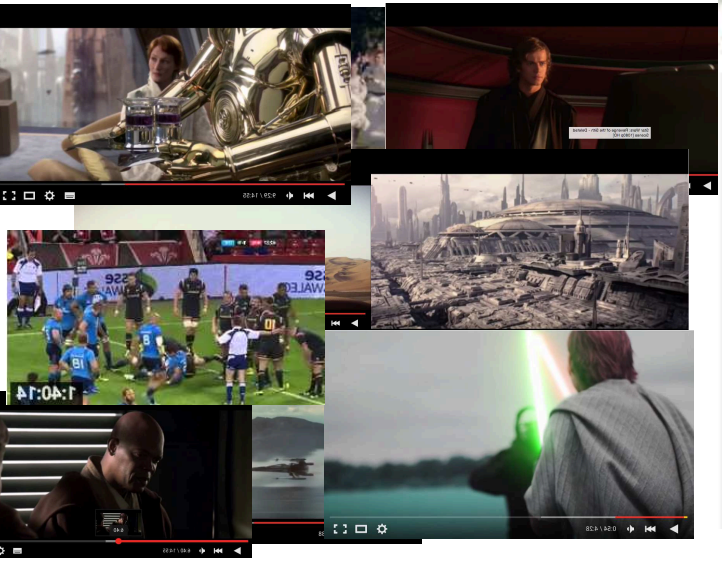
Enumerate all configurations of…



Feature Models

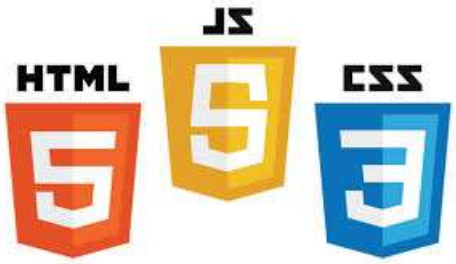


- CarEquipment
  - Healthing
    - AirConditioningFrontAndRear
    - AirConditioning
  - Comfort
    - AutomaticHeadLights
  - DrivingAndSafety
    - FrontFogLights
- Constraints
  - AutomaticHeadLights ⇒ FrontFogLights



foo1.videogen

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

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



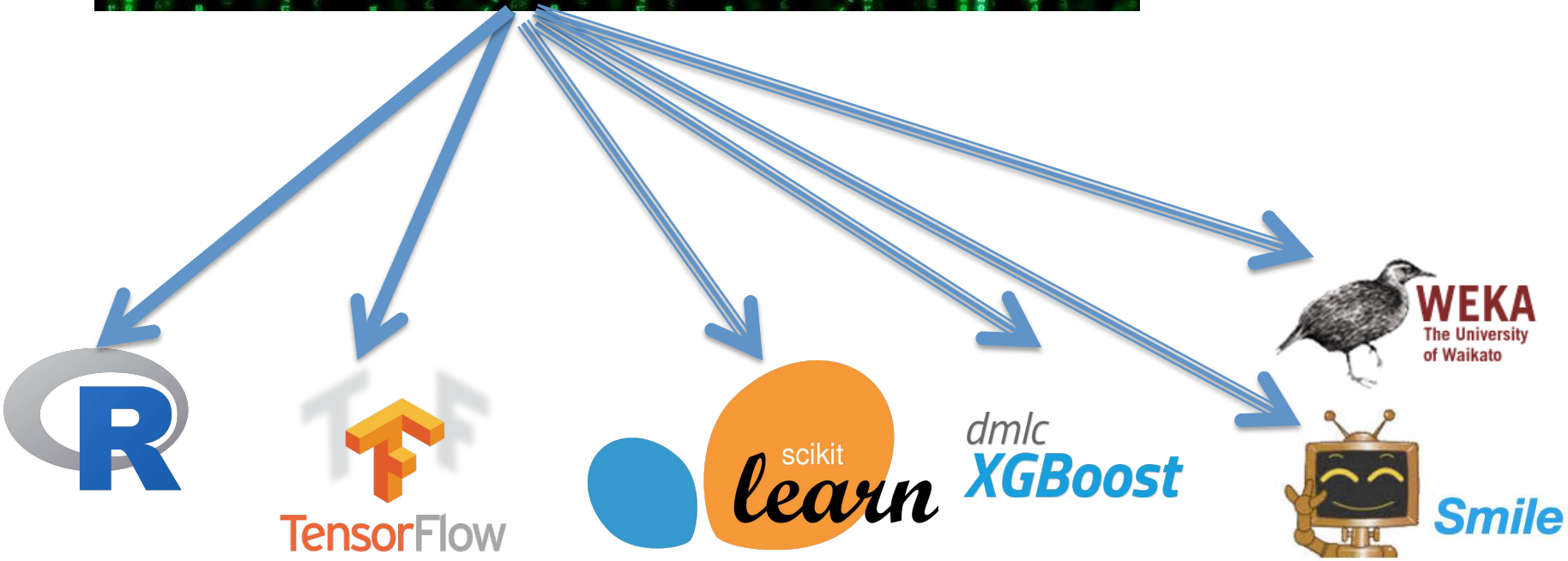
Product line and variability everywhere

- A video generator is a product line
  - Generalization of Bref generator
  - It is even a configurable generator
- Xtext is a configurable generator (see MWE2)
- JHipster is a product line
- ffmpeg is a product line
- MML language is a product line

JHipster

Case Study

MML Language



Variants and configurations

Other references

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

Other references

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