

# Une infrastructure pour (ré-)ingénierer des configurateurs web

## Les configurateurs web

Dans de nombreux domaines et industries, les utilisateurs peuvent commander et customiser une large quantité de produits, caractérisés par des centaines voire des milliers d'*options de configurations*.

En particulier, les *configurateurs web* abondent et offrent un environnement interactif pour aider les utilisateurs durant leur processus de décisions. Typiquement, un utilisateur sélectionne et désélectionne un ensemble d'options. Le configurateur s'assure que ces choix sont cohérents, puis éventuellement propage ces choix pour réduire l'effort de configuration. Les décisions conflictuelles peuvent être également détectées et expliquées à l'utilisateur.

Ce processus de configuration s'effectue généralement en plusieurs étapes jusqu'à la commande finale où toutes les décisions nécessaires à la commande d'un produit ont été réalisées.

Un premier exemple de configurateur web est donné ci-dessous. La capture d'écran est extraite du site d'Opel (<http://www.opel.ie/tools/model-selector/cars.html>, avril 2012).

The screenshot displays the Opel web configurator interface, divided into two main sections. The top section shows the '1. Trims/Series' step, where users can select from five trim levels: S, SC, SRI, SE, and Elite. Each trim level is accompanied by a radio button and a price starting from a certain amount (e.g., S from €19,495.00). Below this, a 'Standard Equipment Filter' dropdown menu is set to 'Heating/Ventilation'. The selected filter shows 'Air conditioning with particle filter, manual controls'. The bottom section shows the '4. Options' step, with a sub-menu for 'Safety/Security'. Under this category, four options are listed with checkboxes and prices: 'Emergency tire inflation kit' (Standard), 'Active front seat head restraints' (€103.00), 'Front and rear parking distance sensors' (€407.00), and 'Tyre Pressure Monitoring System' (€155.00).

1. Trims/Series	2. Engine/Transmission	3. Colour & Style	4. Options	5. Summary
<input type="radio"/> S				from € 19,495.00
<input checked="" type="radio"/> SC				from € 21,195.00
<input type="radio"/> SRI				from € 22,495.00
<input type="radio"/> SE				from € 23,495.00
<input type="radio"/> Elite				from € 25,495.00

Standard Equipment Filter: Heating/Ventilation

Heating/Ventilation

- Air conditioning with particle filter, manual controls

1. Trims/Series	2. Engine/Transmission	3. Colour & Style	4. Options	5. Summary	
Interior	Comfort/Convenience	Option Packs	<b>Safety/Security</b>	Seating	Audio/Comms/Nav
Heating/Ventilation	Mechanical	A-Z			

Safety/Security

<input checked="" type="checkbox"/>	Emergency tire inflation kit	Standard
<input checked="" type="checkbox"/>	Active front seat head restraints - also adds seat belt unfastened warning light for front passenger's seat	€ 103.00
<input type="checkbox"/>	Front and rear parking distance sensors	€ 407.00
<input checked="" type="checkbox"/>	Tyre Pressure Monitoring System	€ 155.00

Dans cet exemple, les options de configuration sont représentés via des *widgets* spécifiques (radio button, combo box, check box). Ces options peuvent être dans différents états:

- sélectionnées: c'est le cas de "Tyre Pressure Monitoring System" ou "SC", mais aussi de "Emergency tire inflation kit" (sélectionnées par défaut)
- désélectionnés: c'est le cas de "Active front seat head restraints"
- non sélectionnés: c'est le cas de "Front and rear parking distance sensors"

Les options de configurations sont *organisées* à travers des "tabs" (e.g., la tab "Options" est active) et différents sous-tabs ("Interior", "Safety/Security", etc.). Cela dénote la séparation du processus de configuration en plusieurs *étapes*.

Des *contraintes* plus ou moins complexes régissent la sélection et la désélection des configurations. Certains choix de l'utilisateur aboutissent à la sélection ou à la désélection automatique d'autres options de configuration (non nécessairement dans la même "sous-tab"). Enfin des informations descriptives sont généralement associées aux options de configurations (e.g., le prix).

Un deuxième exemple est donné ci-dessous, toujours sur le même configurateur web.



Exterior | Interior | Front | Side | Rear 360°

1. Trims/Series | 2. Engine/Transmission | 3. Colour & Style | 4. Options | 5. Summary

### Choose Your Options

Ext. Protection/Styling | Interior | Comfort/Convenience | Option Packs | Seating | **Audio/Comms/Nav**

Heating/Ventilation | Lighting | Mechanical | A-Z

Audio/Comms/Nav	
<input type="checkbox"/> <b>CD 30</b>	<b>Standard</b>
- CD player / MP3 functionality / stereo radio / seven speakers	
<input type="checkbox"/> <b>Mobile phone system with Bluetooth and graphic display</b>	<b>€510.00</b>
- (does not include phone or cradle)	
<input checked="" type="checkbox"/> <b>Touch and Connect</b>	<b>€706.00</b>
- Touch-screen satellite navigation / Mobile phone system with Bluetooth / USB connection with iPod control / Aux-in socket / CD/MP3 player / Stereo radio	

Ext. Protection/Styling | Interior | Comfort/Convenience | Option Packs | Seating | **Audio/Comms/Nav**

Heating/Ventilation | Lighting | Mechanical | A-Z

### Next Step: Accessories

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



**Corsa, SXi**

1.4i 16v VVT, Manual 5-speed

Sovereign Silver, Twist/Elba, charcoal/red

**Total** **€19,630.00**

[Next Step](#)

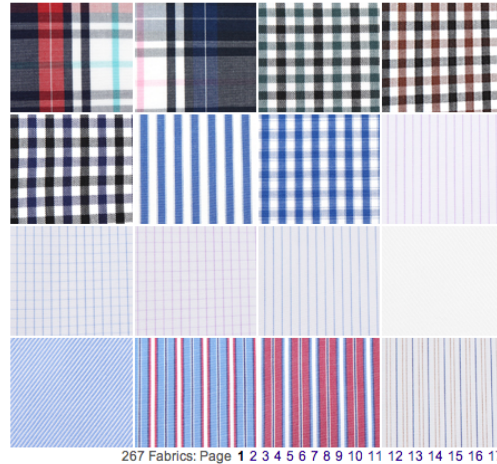
### Pricing Details

SXi	€17,455.00
1.4i 16v VVT, Manual 5-speed	
Sovereign Silver	€535.00
Twist/Elba, charcoal/red	€0.00
17-inch 5-twin-spoke alloy wheels with 215/45 R17 tyres	€762.00
<b>Options (3)</b>	
You selected:	
<input checked="" type="checkbox"/> Touch and Connect	€706.00
<input checked="" type="checkbox"/> DualFloor luggage compartment	€74.00
<input checked="" type="checkbox"/> ISOFIX child seat restraint system for front passenger's seat	€98.00
<b>Total</b>	<b>€19,630.00</b>

D'autres exemples existent, dans de nombreux domaines:

1. FABRIC 2. STYLE 3. PERSONALIZE 4. SIZE 5. SUMMARY

\$79



**NEXT STEP** →

**SAVE DESIGN** 📄

**FILTER FABRICS** SELECTED

- \$0-\$100 (basics)
- \$100-\$150 (really nice)
- \$150-\$200 (even nicer)
- \$200+ (as good as it gets)
- All

**Color**

- [Color swatches]
- All

**Pattern**

- Grid
- Stripes
- Gingham
- All
- Solid
- Plaid
- Other

**Wrinkle Free**

- Non-Iron Only

[More filters](#)

(<http://propercloth.com/custom-dress-shirts/>)

### Design your Dress

**Design Options**

Choose one option from each column to complete your custom design.

Style:	Neckline:	Sleeve:	Skirt:
markate	<input type="radio"/> mandarin	<input checked="" type="radio"/> sleeveless	<input checked="" type="radio"/> straight
amy	<input type="radio"/> jewel	<input type="radio"/> cap	<input type="radio"/> a-line
christine	<input type="radio"/> bateau	<input type="radio"/> short	
grace	<input type="radio"/> square	<input type="radio"/> elbow	<b>Length:</b>
kathleen	<input type="radio"/> deep square	<input type="radio"/> three-quarter	<input checked="" type="radio"/> knee
marilyn	<input type="radio"/> scoop	<input type="radio"/> long	<input type="radio"/> mid
	<input type="radio"/> deep scoop		<input type="radio"/> long
	<input type="radio"/> V		
	<input checked="" type="radio"/> low V		

**Fabrics**

Click on a fabric swatch to see how it looks on the dress form to the right. Filter results by type and search term below. Click on the info button next to the fabric name to see more information about the fabric.

Select a type: <All>

Search:

① textured jacquard dot silk chiffon - silver/soft gold	① stretch wool blend pinstripe - gray/white	① stretch wool blend pinstripe - black/tan	① Italian stretch wool satin - denim blue
① floral crepe -	① emerald green	① dior blue silk	① turquoise silk

(<http://www.dressbydesign.com/dress/design>)

Plus de 800 configurateurs web sont en effet répertoriés sur "Configurator Database": <http://www.configurator-database.com/>

# Une infrastructure pour (ré-)ingénierer des configurateurs web

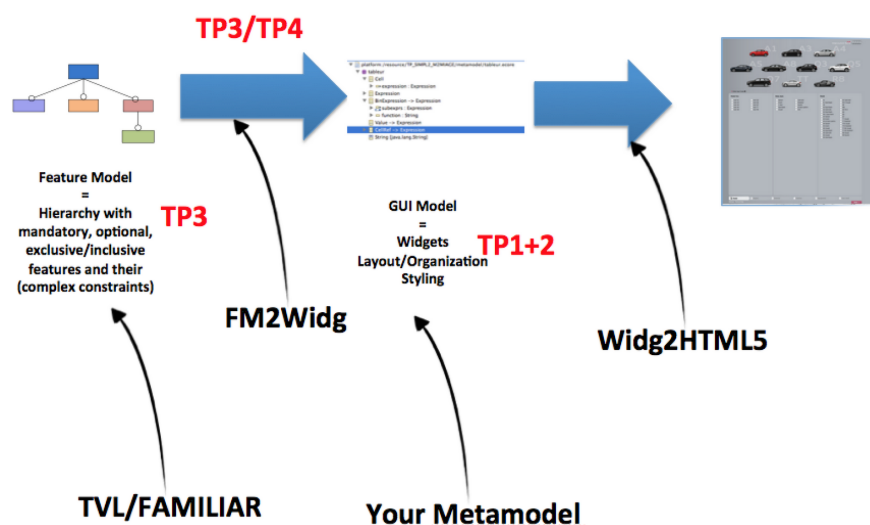
## Objectif général

Tous les configurateurs web ont un objectif similaire et des principes de fonctionnement communs. Ils ont également des caractéristiques propres sur l'aspect visuel et sur la manière de représenter ces configurations, sur l'assistance fournie à l'utilisateur lors de ces choix, sur la manière de séparer le processus de configurations, etc.

L'objectif de notre travail, réalisé à travers les différentes séances de TPs, est de développer une infrastructure complète (langages de modélisation et transformations, éditeurs textuels, et génération de code) pour ingénierer des configurateurs web.

Un scénario notamment visé est de ré-ingénierer des configurateurs web, démontrant ainsi la capacité de notre infrastructure à reproduire l'existant et même permettre de modifier certaines caractéristiques d'un configurateur web (e.g., sur l'aspect visuel).

L'approche retenue est de s'appuyer sur les techniques de l'ingénierie dirigée par les modèles pour parvenir à notre objectif. Plus spécifiquement, nous voulons fournir l'infrastructure suivante:



Nous nous appuyerons sur le formalisme des “feature models” pour modéliser et raisonner sur les relations logiques entre les options de configurations.

Un ensemble de transformations de modèle sera écrit pour transformer le “feature model” en un modèle “graphique” représentant les widgets, les informations de “style” et la disposition des options. Enfin il faudra fournir une transformation qui génère du code web (nous utiliserons HTML5). Cette étape est à réaliser hors séances de TP et est optionnelle.

## TP1

La première étape de notre travail consiste à analyser le domaine des configurateurs. Nous voulons comprendre les points communs et les différences entre ces configurateurs.

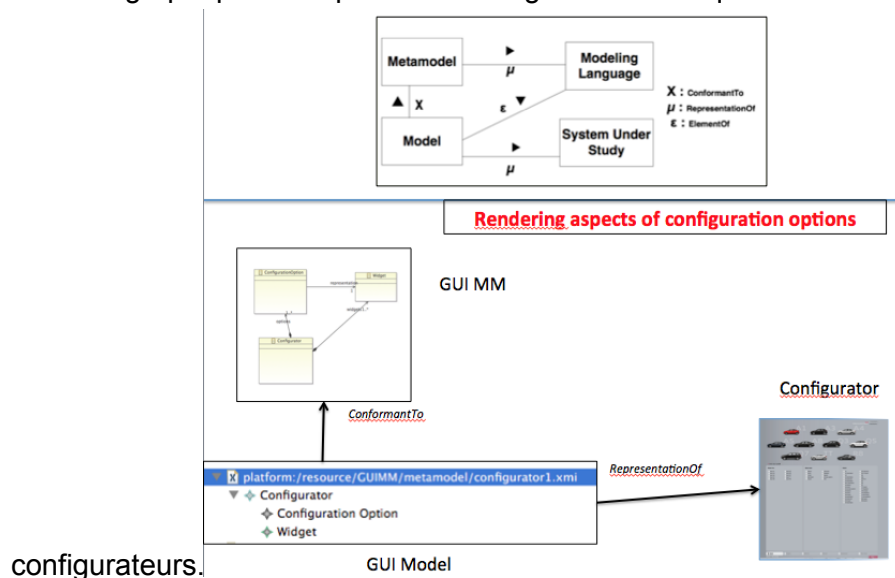
Nous nous concentrons sur la restitution graphique des options de configurations.

Pour ce faire, vous vous constituez en groupe de 2 (3 maximum) et vous sélectionnez 10 configurateurs parmi ceux proposés ici: <http://mathieuacher.com/teaching/SIMPLE-MIAGE2/projet/111configurators.csv>

- Vous produisez un métamodèle (avec Ecore et Eclipse) des “widgets” représentant les options de configurations et de la manière dont ils sont organisés (e.g., “tabs”)
- Vous produisez des modèles de widgets (avec Kermeta), montrant des “patterns” représentatifs de vos 10 configurateurs et bien sûr conforme au métamodèle

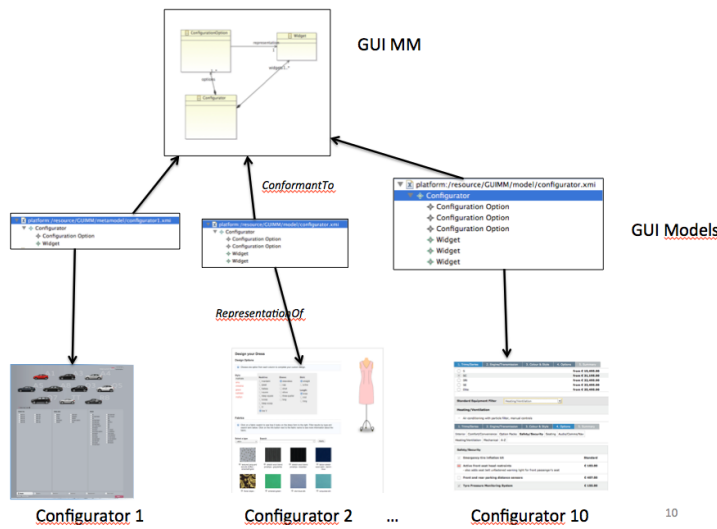
## TP 2

Vous poursuivez votre travail de métamodélisation où l’objectif est de proposer un langage abstrait<sup>1</sup> générique, sous la forme d’un métamodèle Ecore, capable d’exprimer le rendu graphique des options de configurations tels que rencontrés dans les



<sup>1</sup> On pourrait imaginer fournir une syntaxe *concrète* associée (textuelle ou graphique) pour faciliter la saisie de modèles

Le métamodèle doit fournir les constructions nécessaires pour modéliser les 10 configurateurs que vous avez sélectionnés.



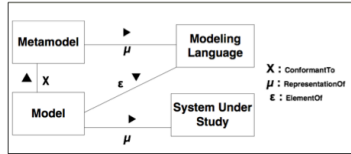
Bien entendu, les modèles spécifiés ne sont que des représentations *simplifiées* d'un configurateur, se concentrant sur leur *aspect* visuel.

### TP 3

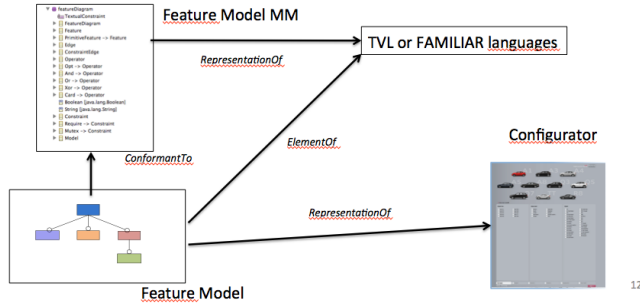
Un configurateur est bien plus qu'une interface graphique statique avec des "widgets": les choix de sélection et désélection sont vérifiés, l'utilisateur est aidé dans son processus de configuration par propagation de certaines décisions, et différentes étapes (e.g., avec ordre) composent le processus de configuration. Ces aspects du configurateur ne sont pour le moment pas modélisés.

En particulier, nous nous intéressons à l'aspect *logique* d'un configurateur, c'est-à-dire aux contraintes régissant les choix de configurations de l'utilisateur. A titre d'exemple, certaines options de configurations sont mutuellement exclusives: l'utilisateur peut sélectionner soit l'une soit l'autre, mais pas les deux.

Pour prendre en compte cet aspect logique, nous utilisons un autre formalisme de modélisation, plus adapté, largement étudié dans le monde académique et utilisé dans l'industrie, les "*feature models*"



**Logical aspects of configuration options**

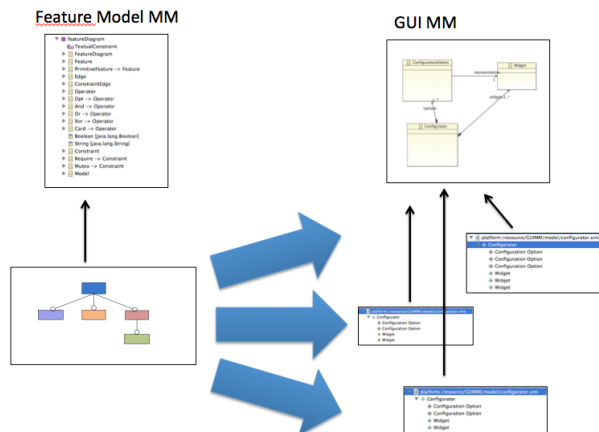


Nous avons vu en détail ce formalisme au cours d'un cours magistral. Un "annexe" vous est fourni pour en comprendre les principes généraux.

Le formalisme des feature models n'exprime pas les aspects graphiques d'un configurateur. Ce n'est pas ce pourquoi il a été conçu: il permet d'exprimer le statut optionnelle ou obligatoire d'une option, de spécifier les exclusions entre plusieurs features (groupe Xor), les implications, etc. mais en aucune manière ce formalisme ne dit comment doivent être restitués graphiquement les "features". Il faut donc combler ce "fossé". La problématique pour ce TP et le suivant est alors la suivante:

*Comment transformer un feature model en un modèle (aspect GUI) conforme au métamodèle que vous avez défini aux TPs 1 et 2?*

Ainsi, étant donné un feature model, nous pouvons obtenir des modèles (aspect GUI) de configurateur à l'aide de transformations.



Le travail à réaliser est le suivant:

- (T1) sélectionner un configurateur et élaborer le feature model correspondant
  - Télécharger un Eclipse avec Xtext: <http://www.eclipse.org/Xtext/>
  - Installer le plugin TVL update site dans Eclipse avec l'url suivante: <http://www.info.fundp.ac.be/~qbo/TVL/>.
  - Spécifier votre modèle TVL
- (T2) identifier des "patterns" de transformation (des slides sont en ligne pour vous aider) et notamment identifier comment les constructions offertes par le formalisme des feature models (e.g., hiérarchie, groupe Xor) peuvent être restituées graphiquement
- (T3) implémenter ces transformations en Kermeta, notamment sur le configurateur considéré
  - Vous avez besoin d'écrire votre transformation en considérant les métamodèles de TVL et votre métamodèle (GUI), le métamodèle TVL est disponible ici: <http://mathieuacher.com/teaching/SIMPLE-MIAGE2/projet/>
  - Pour convertir un modèle TVL (textuel) en XMI (exploitable par Kermeta), utilisez le converteur TVLExporter (cf adresse)

## TP 4

Vous continuez le travail réalisé précédemment.

Démontrer que votre solution est effective sur 5 configurateurs:

- fournir les feature models pour les 5 configurateurs
- écrire les spécifications de la transformation pour les 5 configurateurs et organiser vos transformations comme une librairie
- s'assurer que les résultats (i.e., modèles GUI générés par vos transformations) sont ceux attendus

## A rendre pour le 15 janvier 2013

- métamodèle (Ecore): TP1 et 2
- 5 modèles conformes à votre métamodèle (+ le code Kermeta pour construire ces modèles) pour 5 configurateurs de votre sélection: TP 1 et 2
- 5 feature models pour les 5 configurateurs de votre sélection: TP 3, 4 et 5
- librairie réutilisable de transformations en Kermeta: TP 3 et 4
- démonstration que votre librairie peut permettre de transformer vos 5 feature modèles et obtenir des modèles (GUI): TP 3 et 4
- (optionnel) écrire une transformation vers HTML 5