



Découpez vos Modèles avec Kompren : une Démonstration

Arnaud Blouin, Benoit Combemale, Benoit Baudry

► **To cite this version:**

Arnaud Blouin, Benoit Combemale, Benoit Baudry. Découpez vos Modèles avec Kompren : une
Démonstration. 2012. hal-00696149

HAL Id: hal-00696149

<https://hal.inria.fr/hal-00696149>

Submitted on 11 May 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Découpez vos Modèles avec *Kompren* : une Démonstration

Arnaud Blouin, Benoit Combemale, and Benoit Baudry

Triskell, INRIA/IRISA, Rennes

{ablouin,bcombemale}@irisa.fr bbaudry@inria.fr

1 Introduction

Le découpage de modèles (*model slicing*) est une opération qui extrait un sous-ensemble d'un modèle dans un but précis, tel que la compréhension de modèles et l'amélioration de performances. Cependant, les approches actuelles de découpage de modèles sont dédiées à un domaine de modélisation (métamodèle) particulier. L'apparition régulière de nouveaux domaines nécessite alors la conception et l'implantation de nouvelles fonctionnalités de découpage de modèles. Dans nos récents travaux, nous avons proposé *Kompren*, un environnement pour définir et générer des découpeurs de modèles (*model slicers*) pour tout type de domaine de modélisation [1]. Cette démonstration présente les différents outils de *Kompren*¹ et trois cas d'utilisation illustrant l'expressivité du langage.

2 Présentation de *Kompren*

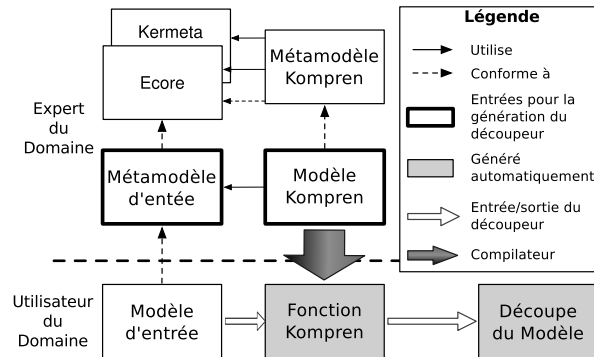


FIGURE 1. Vue d'ensemble du processus d'utilisation de *Kompren*

La figure 1 présente une vue d'ensemble de l'utilisation de *Kompren*. Tous les concepts et relations de *Kompren* sont définis dans le métamodèle de *Kompren*.

1. http://people.irisa.fr/Arnaud.Blouin/software_kompren.html

Un modèle *Kompren* décrit un découpeur de modèles en sélectionnant dans le métamodèle d'entrée les classes et les propriétés d'intérêts qui seront extraites lors de l'exécution, le métamodèle d'entrée étant un modèle Ecore. Une fois un modèle *Kompren* défini (une syntaxe textuelle et son éditeur sont fournis), il peut être compilé en une fonction *Kompren* consistant en du byte-code Scala/Java exécutable. Cette fonction prend en entrée un modèle à découper, conforme au métamodèle d'entrée, et produit en sortie une découpe de ce modèle.

Par défaut, l'exécution d'un découpeur (fonction *Kompren*) sur un modèle d'entrée assure la conformité du modèle de sortie (*slice*) en extrayant les éléments nécessaires à la préservation de cette conformité même s'ils ne sont pas explicitement sélectionnés dans le modèle *Kompren*. Cependant, la conformité n'est pas requise dans certains cas ; c'est pourquoi *Kompren* offre des fonctionnalités relâchant cette conformité afin de permettre aux développeurs de paramétrer les découpeurs de modèles, notamment à l'aide du langage d'action Kermeta [2].

3 Description de la Démonstration

La démonstration vise à expliquer comment *Kompren* fonctionne et les différentes utilisations qu'il est possible d'en faire. Pour cela, cette démonstration se compose des scénarios suivants :

- **Compréhension de modèles.** Le zoom sémantique et le filtrage dynamique sont des techniques de visualisation dont le but est d'afficher uniquement les informations relatives à une préoccupation donnée, ce qui consiste donc en une forme de découpage de modèles. Nous montrons comment *Kompren* peut être utilisé pour définir de telles techniques de visualisation puis intégré dans un visualiseur de modèles.
- **Optimisation.** Étant donné un point d'entrée dans un modèle, le découpage de modèles peut être utilisé pour supprimer les éléments non pertinents relativement à ce point d'entrée. Nous illustrons ce principe en expliquant comment *Kompren* est utilisé dans l'environnement Kermeta pour réduire le temps de compilation des programmes Kermeta 2.
- **Analyse statique.** Étant donnée une opération *op* dans un modèle exécutable, le découpage de modèles peut être utilisé pour identifier les variables non utilisées déclarées dans *op*. Nous présentons comment *Kompren* permet de définir une telle analyse pour des opérations Kermeta.

Cette démonstration s'effectue à l'aide des outils de développement de *Kompren*, notamment son éditeur textuel, son compilateur et Kermeta 2.

Références

1. Arnaud Blouin, Benoît Combemale, Benoit Baudry, and Olivier Beaudoux. Modeling model slicers. In *Proceedings of MODELS'11*, pages 62–76, 2011.
2. Pierre-Alain Muller, Franck Fleurey, and Jean-Marc Jézéquel. Weaving executability into object-oriented meta-languages. In *Proc. of MODELS'05*, pages 264–278, 2005.