
Représentation déclarative et transformations/manipulations des modèles

[Séparation en modélisation déclarative entre la représentation déclarative des modèles et toutes les transformations/manipulations des modèles qui s'ensuivent](#)

[Illustration : un exemple de mise en pratique de l'approche déclarative](#)

[De la spécification déclarative à une plate-forme de modélisation cible](#)

Séparation en modélisation déclarative entre la représentation déclarative des modèles et toutes les transformations/manipulations des modèles qui s'ensuivent ^(^)

Ce paragraphe s'appuie sur le document « [Position Paper on Declarative Modelling in Ecological and Environmental Research](#) ». Il est repris ici essentiellement des éléments du paragraphe « 1. Introduction » de ce document, à consulter directement pour plus d'informations.

Références du document source :

Document « **Position Paper on Declarative Modelling in Ecological and Environmental Research** », de Robert Muetzelfeldt (Centre for Intelligent Systems and their Applications, School of Informatics, The University of Edinburgh), vu le 04/09/06 à l'URL <http://www.decmod.org/documents/dmeer.pdf> .

Le [dossier modelia](#) associé à ce document est : « [Le quoi, le pourquoi et le comment de la modélisation déclarative](#) ».

Séparation en modélisation déclarative entre la représentation déclarative des modèles et toutes les transformations/manipulations des modèles qui s'ensuivent

L'approche déclarative considère la modélisation en tant que processus de conception. Un aspect essentiel de la modélisation déclarative est la séparation entre, d'un côté la représentation des modèles (a), et de l'autre côté tous les outils (b) développés afin de manipuler les modèles. Il est à noter qu'exécuter le modèle (lancer une simulation) constitue une des manipulations du modèle mais n'est pas la seule envisageable.

Par exemple pour un modèle donné en écologie :

- (a) Ce modèle serait sauvegardé/enregistré en langage déclaratif dans un fichier spécifiant les objets, les variables et les équations du modèle.
- (b) Pour ce modèle, il existerait par exemple un outil permettant d'afficher la structure du modèle selon un type de représentation donné, un autre outil permettant de simuler le comportement du modèle, un autre outil permettant de comparer deux versions du modèle.

Voir aussi le schéma ci-dessous.

En modélisation déclarative, séparation entre (a) la représentation des modèles et (b) toutes les transformations/manipulations des modèles qui s'ensuivent :

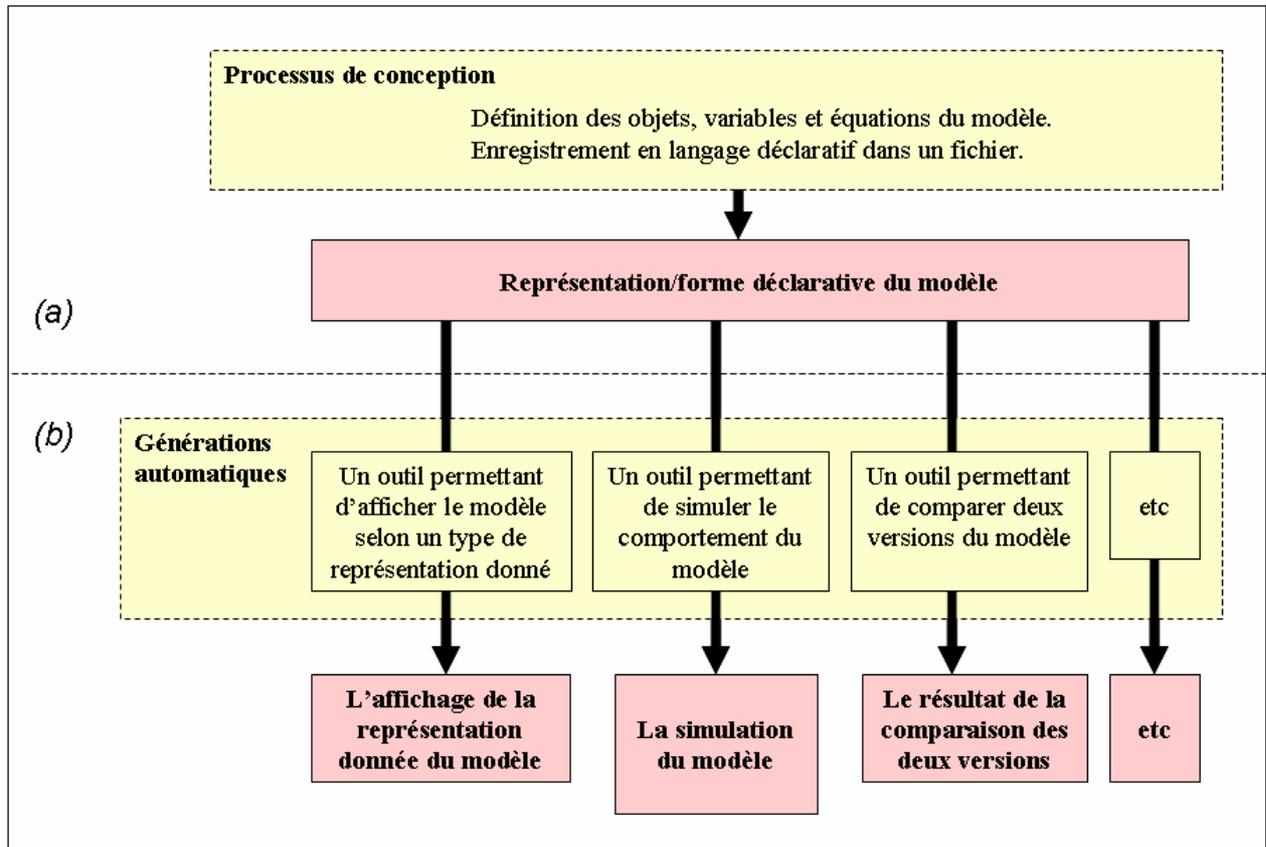


Illustration : un exemple de mise en pratique de l'approche déclarative ^(^)

Démonstration « MultiGuise »

Références : la démonstration « MultiGuise » est en ligne à l'URL <http://www.decmod.org/multiguise> (vue le 15/09/06).

Présentation de la démonstration :

Cette démonstration met en pratique la modélisation selon l'approche déclarative. Dans le processus ici illustré, tout d'abord la conception du modèle est faite sous forme déclarative. Ensuite, **à partir de la forme déclarative** (représentation XML), **toutes sortes de résultats** sont obtenues automatiquement au moyen d'outils de transformation appropriés.

Description de la démonstration :

La démonstration propose plusieurs **modèles** : « Simple ecosystem », « McMurtrie vegetation », « DaisyWorld », « Century soil », « Predator-prey », « Individual-based trees ».

La démonstration propose d'appliquer à chacun de ces modèles plusieurs **outils de transformations** : « Model summary », « Summary statistics », « MathML display », « Javascript generator », « MODCOM code generator », « Simulator », « Raw XML », « Model diagram (SVG) », « TIME code generator », « MODCOM C# generator ».

Les résultats obtenus en appliquant un des outils à un des modèles sont générés de manière dynamique, à partir de la forme déclarative du modèle (sa représentation XML) qui est l'unique forme stockée du modèle.

Simulation du modèle : Un des résultats qu'il est possible de générer donne moyen de simuler le comportement du modèle ; c'est le bouton « run » contenu dans le résultat obtenu avec la transformation « Simulator ».

Autre explication :

Voir la présentation « **Environmental modelling and Web 2.0 : Using Connotea to share XML-represented information** » faite par **Robert Muetzelfeldt** lors de la conférence « **Web 2.0 conference** » organisée le 24 novembre 2006 par l'université d'Edinburgh (« Star Gazing Group », <http://www.itfutures.ed.ac.uk>) vu le 01/12/06 à l'URL http://www.itfutures.ed.ac.uk/robert_muetzelfeldt_synopsis.htm .

De la spécification déclarative à une plate-forme de modélisation cible ^(^)

Voir le dossier modelia « [Modélisation déclarative, ontologies, génération de code, modèles indépendants des plates-formes](#) » qui repose sur l'article « Declarative modelling for architecture independence and data/model integration : a case study » de Ferdinando Villa (University of Vermont, USA), Marcello Donatelli (ISCI, Bologna, Italy), Andrea Rizzoli (IDSIA, Lugano, Switzerland), Peter Krause & Sven Kralisch (University of Jena, Jena, Germany), Frits K. van Evert (PRI, Wageningen, The Netherlands), vu le 04/09/06 à l'URL <http://www.iemss.org/iemss2006>.

Voir en particulier dans ce dossier le paragraphe « [De la base de données des connaissances au code ; de la spécification déclarative à une plate-forme de modélisation cible](#) ».

[La page au format pdf](#) (01/12/06)

- Dernière mise à jour le 01/12/06, mise en ligne le 27/09/06 –

Plate-forme INRA-ACTA-ICTA, Modelia <http://www.modelia.org>
