HAL
open science

AppliDE : modélisation et génération d'applications pour smartphones

Clément Quinton, Christophe Demarey, Nicolas Dolet, Laurence Duchien

To cite this version:

Clément Quinton, Christophe Demarey, Nicolas Dolet, Laurence Duchien. AppliDE : modélisation et génération d'applications pour smartphones. Journées sur l'Ingénierie Dirigée par les Modèles (IDM'11), Jun 2011, Lille, France. pp.41-45. inria-00617565

HAL Id: inria-00617565

https://inria.hal.science/inria-00617565v1

Submitted on 5 Sep 2011

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.

AppliDE : modélisation et génération d'applications pour smartphones

Principes d'une Ligne de Produits Logiciels pour applications mobiles

Clément Quinton — Christophe Demarey — Nicolas Dolet — Laurence Duchien

Equipe-Projet INRIA ADAM, Université Lille 1, LIFL UMR CNRS 8022
INRIA Lille - Nord Europe
40 Avenue Halley - Park Plaza - Bât A
59 650 Villeneuve d'Ascq

prenom.nom@inria.fr

RÉSUMÉ. *Le développement d'applications pour téléphones intelligents s'est complexifié avec l'arrivée de nouvelles plates-formes. Cet article explique comment réaliser une application pour terminaux mobiles en utilisant des Lignes de Produits Logiciels (LPL). Les LPL permettent de concevoir et développer une famille de produits logiciels qui prend en compte des facteurs de variation tout en minimisant les coûts et les temps de réalisation. Nous introduisons le cadre logiciel* APPLIDE*, qui réalise des LPL pour téléphones intelligents avant d'exposer son fonctionnement au cours d'une démonstration.*

ABSTRACT. *The availability of new mobile platforms has made the development of smartphone applications more complicated. This paper explains how to develop an application for mobile devices using Software Product Lines (SPL). Considering variation factors, SPL allows the conception and the development of a software products family minimizing realization cost and time. We present the* APPLIDE *framework, which provides SPL for smartphones, and we show how it works with a short demonstration.*

MOTS-CLÉS : *Ligne de Produits Logiciels, Ingénierie Dirigée par les Modèles, métamodèle, gabarits, génération de code, téléphones intelligents.*

KEYWORDS: *Software Product Line, Model-Driven Engineering, meta-model, templates, code generation, smartphones.*

1. Introduction

L'arrivée des téléphones intelligents (en anglais *smartphones*) sur le marché de la téléphonie mobile a ouvert de nouvelles perspectives à leurs utilisateurs. En effet, ces smartphones doivent leur succès aux nombreuses applications qu'ils proposent. A l'origine réservées au seul iPhone, elles sont aujourd'hui accessibles sur l'ensemble de ces terminaux mobiles via des places de marché. Pour toucher un plus large public, l'application développée doit être disponible au plus tôt pour un maximum de smartphones, quel que soit leur système d'exploitation (iPhone, Android, Windows Phone 7 ou autre). Mais comment développer et déployer rapidement une application sur des environnements aussi hétérogènes ? Cet article propose une solution qui repose à la fois sur les principes des Lignes de Produits Logiciels (LPL) et sur l'Ingénierie Dirigée par les Modèles (IDM) et qui est réalisée dans le cadre logiciel APPLIDE.

2. AppliDE

Les LPL permettent de définir des produits logiciels à partir de diagrammes de caractéristiques (en anglais *features*). La sélection des *features* permet de dériver des produits ayant des caractéristiques communes et appartenant à un même domaine (Kang *et al.*, 1990, Pohl *et al.*, 2005). Les produits obtenus sont sensiblement différents les uns des autres : on parle alors de variabilité. Notre proposition, APPLIDE, permet d'automatiser la dérivation d'applications pour différentes plates-formes cibles à partir d'un même modèle. La variabilité se retrouve dans les exigences fonctionnelles et non fonctionnelles requises par l'application mais également au niveau de l'environnement technique de celle-ci, chaque plate-forme possédant son propre langage de développement[1]. Pour le déployer sur différentes plates-formes d'exécutions, nous nous baserons sur les techniques de l'IDM en proposant des générateurs personnalisés vers ces plates-formes. Le métamodèle d'APPLIDE capture les concepts permettant de choisir *(i)* les fonctionnalités souhaitées au sein de l'application (envoi de mail, de SMS, activation du GPS), *(ii)* les différentes sources de données requises (fichier, Web Service ou base de données) et *(iii)* les éléments graphiques utilisés (vues sous forme de listes, boutons, labels, champs de saisie). Le développeur sélectionne les éléments souhaités et modélise donc l'application sans se soucier de la plate-forme sur laquelle elle sera déployée. Les fichiers sources de l'application finale, propres à chaque plate-forme cible, sont ensuite obtenus en exécutant les générateurs de code d'APPLIDE. Cas d'utilisation typique de l'IDM et des transformations, APPLIDE peut facilement être étendu afin de supporter une nouvelle plate-forme cible, en ajoutant de nouveaux générateurs. Sous forme de gabarits de génération, ceux-ci ont un faible coût de mise en place et de maintenance. L'utilisation d'APPLIDE permet ainsi de minimiser les coûts et les temps de réalisation. A partir d'un même modèle, on obtient le code source natif des *n* plates-formes cibles, là où un processus de développement classique requiert *n* développements dans *n* langages différents.

1. L'extraction d'un diagramme de *features* à partir de l'implémentation d'APPLIDE est en cours pour supporter de futurs raisonnements.

3. Travaux connexes

T. Young, quant à lui, a développé dans ses travaux une LPL pour mobiles J2ME en utilisant les techniques de la Programmation Orientée Aspects (Young, 2005). Dans le contexte industriel, les sociétés Mobile Distillery[2] ou Neomades[3] proposent chacune un outil permettant le portage d'applications mobiles développées en JavaME vers d'autres plates-formes. D'autres cadres logiciels, comme par exemple Appcelerator[4], Tino[5] ou PhoneGap[6], traduisent du code HTML et JavaScript en code source iPhone ou Android. Par notre approche, APPLIDE est indépendant du langage de programmation mais également de la plate-forme cible.

4. Conclusion

Dans cet article, nous avons montré comment l'Ingénierie Dirigée par les Modèles peut supporter l'implémentation d'une Ligne de Produits Logiciels dédiée au développement d'applications pour *smartphones*. Dans la continuité de ce travail, la prise en compte des informations proposées par les capteurs de ces *smartphones* permet d'envisager une adaptation à l'exécution de ces applications. Les travaux de l'équipe ADAM avec la thèse de Carlos Parra (Parra, 2011) vont dans ce sens avec en particulier le projet FUI CAPPUCINO[7] qui vise à concevoir, déployer et exécuter des applications mobiles sensibles au contexte et capables de s'adapter dynamiquement en fonction des services disponibles.

5. Bibliographie

Kang K. C., Cohen S. G., Hess J. A., Novak W. E., Peterson A. S., Feature-Oriented Domain Analysis (FODA) Feasibility Study, Technical report, Carnegie-Mellon University Software Engineering Institute, November, 1990.

Parra C., Towards Dynamic Software Product Lines : Unifying Design and Runtime Adaptations, Thèse de doctorat, INRIA - Université Lille 1, France, 2011.

Pohl K., Böckle G., Linden F. J. v. d., *Software Product Line Engineering : Foundations, Principles and Techniques*, Springer-Verlag, 2005.

Young T., Using AspectJ to Build a Software Product Line for Mobile Devices, Master's thesis, University of British Columbia, Vancouver, Canada, 2005.

2. Mobile Distillery : `http ://www.mobile-distillery.com`
3. Neomades : `http ://www.neomades.com`
4. Appcelerator : `http ://www.appcelerator.com`
5. Tino : `http ://www.tinomobile.com/index.php`
6. PhoneGap : `http ://www.phonegap.com`
7. CAPPUCINO : `http ://www.cappucino.fr`

Remerciements

Ce travail a été soutenu par le Ministère Français de l'Education Nationale et de la Recherche, le Conseil Régional du Nord-Pas de Calais et le FEDER via le *Contrat de Projets Etat Region* (CPER) 2007-2013.

APPENDIX

A. Démonstration

Afin d'illustrer notre approche, nous avons réalisé une étude de cas en développant une application mobile pour iPhone et Android. Cette application, inspirée des outils de Gestion de la Relation Client (GRC), offre les fonctionnalités suivantes : *(i)* connexion au serveur, *(ii)* affichage de la liste des clients en fonction du profil identifié, *(iii)* détail d'un client et *(iv)* possibilité de le contacter par mail, sms ou appel et de le localiser sur une carte. La démonstration est décomposée en trois phases : une phase de sélection et de modélisation, une phase de génération de code et une phase de déploiement (Figure 1).

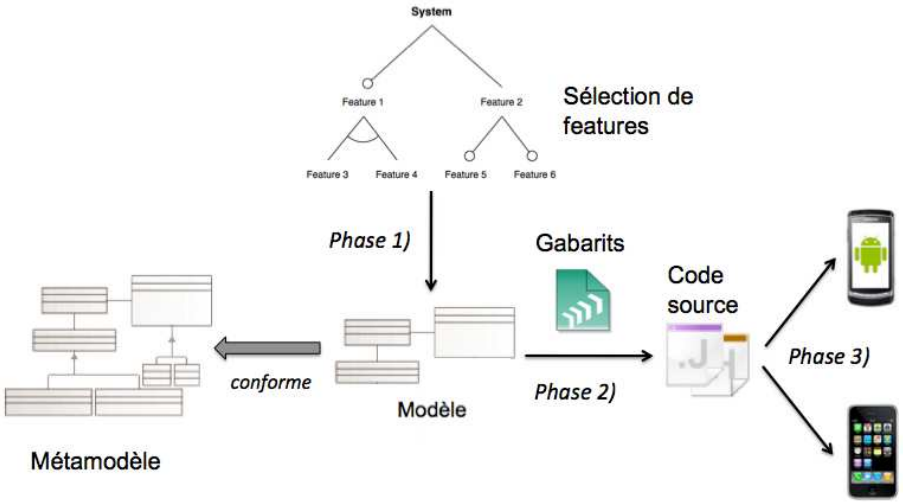

Figure 1. *Processus de développement avec* APPLIDE.

Phase 1, sélection des *features*, composition et modélisation de l'application. En utilisant les outils d'Eclipse Modeling Framework (EMF), nous créons un modèle auquel nous ajoutons tous les éléments nécessaires à son bon fonctionnement : source de données requise (un Web Service fournissant les opérations de connexion et de récupération de la liste des clients), objets métiers manipulés (objet Client) et les différentes vues (écran de connexion, liste des clients, détail d'un client, établissement d'un devis) de l'interface graphique (Figure 2).

Figure 2. *Application GRC, écran de connexion sur Android et iPhone.*

Phase 2, génération de code. Les générateurs de code ont été développés sous la forme de gabarits Acceleo. Dans le cadre de cette démonstration, ce sont près de 1500 lignes de code qui sont générées. Après l'avoir modélisée et générée, l'application est entièrement fonctionnelle et n'a nécessité aucun ajout en termes de développement.

Phase 3, déploiement. Les fichiers source natifs générés (Objective-C pour iPhone, Java pour Android) sont exportés afin d'être utilisés dans leur projet respectif. Chaque projet est ensuite compilé et exécuté dans son environnement de développement dédié. Les applications peuvent enfin être déployées sur les terminaux mobiles.

L'application a également été testée sur iPad et fonctionne parfaitement. Avec la LPL d'APPLIDE, elle pourrait facilement être portée sur d'autres plates-formes (Windows Mobile 7, Bada, Palm) à condition d'incorporer les générateurs de code correspondant dans la LPL.