Incremental Evaluation of Model Queries over EMF Models: A Tutorial on EMF-IncQuery*

Gábor Bergmann, Ákos Horváth,
István Ráth, and Dániel Varró

Budapest University of Technology and Economics,
Department of Measurement and Information Systems,
H-1117 Magyar tudósok krt. 2, Budapest, Hungary
{bergmann,ahorvath,rath,varro}@mit.bme.hu

Keywords: incremental pattern matching, EMF, model query.

1 Introduction

Model driven development platforms such as the industry leader Eclipse Modeling Framework (EMF) greatly benefit from pattern matching, as it supports various usecases including model validation, model transformation, code generation and domain specific behaviour simulation. Pattern matching is a search for model elements conforming to a given pattern that describes their arrangement and properties, e.g. finding a violation of a complex well-formedness constraint of a domain specific modeling language.

Two major issues arise in pattern matching: (i) it can have significant impact on runtime performance and scalability; and (ii) it is often tedious and time consuming to (efficiently) implement manually on a case-by-case basis. The latter is typically addressed by a declarative query language (e.g., EMF Query, OCL) processed by a general-purpose pattern matching engine.

2 EMF-IncQuery

The current tutorial introduces a declarative model query framework over EMF called EMF-IncQuery [1], using the graph pattern formalism (from the theory of graph transformations) as its query language and relying on incremental pattern matching for improved performance. Graph patterns represent conditions (or constraints) that have to be fulfilled by a part of the instance model. A basic graph pattern consists of *structural constraints* prescribing the existence of nodes (EObjects) and edges (EReference and EAttribute instances) of a given type. Additional features include pattern composition, negation, and attribute constraints.

The advantage of declarative query specification is that it achieves (effective) pattern matching with much less time-consuming, manual coding effort than

* This work was partially supported by the SecureChange (ICT-FET-231101) European Research Project.

R.B. France et al. (Eds.): ECMFA 2011, LNCS 6698, pp. 389–390, 2011.

ad-hoc model traversal. While EMF-IncQuery is not the only technology providing declarative queries over EMF (think of EMF Query or MDT-OCL), it has a distinguishing feature, namely *incremental pattern matching*.

In case of incremental pattern matching, matches of a pattern are explicitly stored and remain available for immediate retrieval throughout the lifetime of the EMF ResourceSet. Even when the EMF model is modified, these caches are continuously and automatically kept up-to-date using the EMF Notification API. This maintenance happens without additional coding, and works regardless how the model was modified (graphical editor, programmatic manipulation, loading a new EMF resource, etc.). In many scenarios this technique provides significant speed-up at the cost of increased memory consumption.

Additionally, some shortcomings of EMF are mitigated by capabilities of EMF-IncQuery, such as cheap enumeration of all instances of a type regardless where they are located in the resource tree. Another such use is navigation of EReferences in the opposite direction, without having to augment the metamodel with an EOpposite, which is problematic if the metamodel is beyond our control.

While EMF-IncQuery might not be the tool best suited for every single model query problem, it offers some great and unique features in a range of use cases, some of which will be demonstrated in the tutorial.

3 Tutorial

In this tutorial, we give an overview of the EMF-IncQuery system, demonstrating how the technology can be applied, and discuss gains and trade-offs. We will show how cheap pattern matching can have significant performance advantages in a number of scenarios, such as model validation (model editors can continuously evaluate complex well-formedness constraints and give efficient, immediate feedback), model transformation (determining the applicability of declarative transformation rules), and simulation of dynamic domain-specific models (identifying the possible model evolutions). These will be illustrated using a case study of on-the-fly well-formedness constraint evaluation in UML models.

Our target audience includes experts already working with EMF based query or model transformation technologies like EMF Query or ATL and programmers/educators who wish to learn about a new EMF based query technology. The tutorial will build on a basic understanding of EMF and graph patterns to explain these technicalities and will focus on a software engineer's viewpoint on using our framework.

Reference

1. Bergmann, G., Horváth, A., Ráth, I., Varró, D.: Incremental evaluation of model queries over EMF models. In: Petriu, D., Rouquette, N., Haugen, O. (eds.) MODELS 2010. LNCS, vol. 6394, pp. 76–90. Springer, Heidelberg (2010), `http://viatra.inf.mit.bme.hu/incquery`