Form Follows Function
Model-Driven Engineering for Clinical Trials

Jim Davies¹, Jeremy Gibbons¹, Radu Calinescu², Charles Crichton¹, Steve Harris¹, and Andrew Tsui¹

¹ Department of Computer Science, University of Oxford, Wolfson Building, Parks Road, Oxford OX1 3QD, UK
http://www.cs.ox.ac.uk/firstname.lastname/
² Computer Science Research Group, Aston University, Aston Triangle, Birmingham B4 7ET, UK
http://www-users.aston.ac.uk/~calinerc/

Abstract. We argue that, for certain constrained domains, elaborate model transformation technologies—implemented from scratch in general-purpose programming languages—are unnecessary for model-driven engineering; instead, lightweight configuration of commercial off-the-shelf productivity tools suffices. In particular, in the CancerGrid project, we have been developing model-driven techniques for the generation of software tools to support clinical trials. A domain metamodel captures the community’s best practice in trial design. A scientist authors a trial protocol, modelling their trial by instantiating the metamodel; customized software artifacts to support trial execution are generated automatically from the scientist’s model. The metamodel is expressed as an XML Schema, in such a way that it can be instantiated by completing a form to generate a conformant XML document. The same process works at a second level for trial execution: among the artifacts generated from the protocol are models of the data to be collected, and the clinician conducting the trial instantiates such models in reporting observations—again by completing a form to create a conformant XML document, representing the data gathered during that observation. Simple standard form management tools are all that is needed. Our approach is applicable to a wide variety of information-modelling domains: not just clinical trials, but also electronic public sector computing, customer relationship management, document workflow, and so on.

It is the pervading law of all things organic and inorganic,
Of all things physical and metaphysical,
Of all things human and all things super-human,
Of all true manifestations of the head,
Of the heart, of the soul,
That the life is recognizable in its expression,
That form ever follows function. This is the law.

Louis Sullivan

© Springer-Verlag Berlin Heidelberg 2012
1 Introduction

Randomized controlled trials are considered to be the ‘gold standard’ for experiments in medicine. They provide the most reliable evidence supporting or refuting a scientific hypothesis, such as that ‘treatment $X$ cures more patients suffering from disease $D$ than does treatment $Y$’. An experiment is designed: treatment regimes $X$ and $Y$ will be specified; patients suffering from disease $D$ will be recruited; recruits will be stratified into groups with similar relevant characteristics, based on factors such as age, gender and lifestyle; patients within each group will be allocated at random to treatment $X$ or treatment $Y$; the results will be analyzed to determine whether or not the difference in effect of the treatments is statistically significant. So far, so good.

From a software point of view, a clinical trial is largely an exercise in data management: observations have to be specified, collected, recorded, integrated, and analyzed. But the software engineering aspects of setting up and running a clinical trial are not trivial. Two particular problems that we will address involve data integration and tool generation.

The data integration problem occurs because medical researchers want to be able to combine the results of multiple trials, a process known as meta-analysis. It is often the case that a single trial in isolation does not have adequate statistical power to yield a robustly significant conclusion. Nevertheless, if sufficiently many trials have been conducted, investigating sufficiently similar hypotheses and collecting sufficiently similar data, it may be possible to pool the results to greater effect: “...the drug Tamoxifen—an oestrogen blocker that may prevent breast cancer cells growing—was the object of forty-two studies world-wide, of which only four or five had shown significant benefits. But this did not mean that Tamoxifen did not protect against breast cancer. When we put all the studies together it was blindingly obvious that it does...” [34]. In other situations, the meta-analysis aims at evaluating new hypotheses that are formulated long after the completion of the trials that originally collected the data involved—in this case, data from trials investigating quite different hypotheses may be integrated.

Either way, for meta-analysis to be possible, it is necessary to capture and curate metadata expressing the ‘semantics’ of the data; only then is it possible to determine whether data collected in different trials are commensurate, and if so, how to relate them. For example, when measuring blood pressure, it is not enough to record a pair of numbers, or a pair of pressures, or a pair of measurements in mmHg, or even to indicate that these represent systolic and diastolic pressure; it is also necessary to know how that data was collected (at rest, or after five minutes on a treadmill?), and maybe factors such as who collected it (in the clinic by a professional, or at home by the patient?) and how recent and reliable it is. This semantic metadata is an essential part of the context of the data, and logically forms part of the model of the trial—alongside more syntactic metadata such as the name of the trial and the types of data items.

As for tool development, standard practice in clinical trials management these days is to pass a textual document containing the trial protocol over to database programmers in the clinical trials unit, or to consultants from a trials management