MetaCase

Nokia Case Study

MetaEdit+ revolutionized the way Nokia develops mobile phone software

Effective development tools are vital for increasing productivity and improving the quality of products. When Nokia was searching for an effective modeling tool, the prime criteria were encapsulation of domain knowledge, flexible method support and code generation. After evaluating a number of off-the-shelf tools, they undertook the development of their own solution using the MetaEdit+ metaCASE tool from MetaCase. This decision has been shown to be wholly justified.

Mobile phones and mobile computing is one of the fastest growing markets today. The number of mobile phone users worldwide approaches the 1 billion mark, while the phones continuously include new innovations and features, thus changing the way people communicate and make business. At the same time the software inside the mobile phone has become the most critical part from the consumer's point of view: it determines the winners.

For the manufacturers, time to market and quality are critical: innovative features and problem-free operation guarantee the publicity and good reviews which lead to top sales figures. In this environment, where releasing a product a month before a competitor translates to millions of dollars, the productivity of the development process and tools is vital.

OBJECTIVES FOR DEVELOPMENT TOOLS

To compete more effectively, Nokia was looking for development tools that improve the productivity of development teams by an order of magnitude. These improvements were to be achieved by applying the following strategies:

- Working at higher abstraction levels so that designers don't have to know everything. They can focus on designs rather than on how to implement it in code.
- Encapsulate domain knowledge so that only the relevant part of mobile phone software is captured. This lets development teams focus on the characteristics of the needed functionality. The domain knowledge contained in the method also makes for a easier, faster learning curve for new employees or transferred personnel.
- Link designs to code generators so that designers are effectively 'writing code' as they design (but they don't realize it immediately!).
- Underpin the development process with a tool effective enough that no one will want to develop outside the tool.

THE TOOL SEARCH

In the beginning, a team in Nokia's Advanced Development Group evaluated prominent CASE tools. The result of the tool evaluation was a disappointment. The tools examined were found to be inflexible in terms of method extensions, code generators and process support. Most available tools support published methods that were not domain-oriented. They allow the description of almost any application, but only by often clumsy mappings from the application domain to the tool's own concepts, losing vital understanding of the domain in the process. "UML and other methods say nothing about mobile phones. We were looking for more", said David Narraway, Project Manager at Nokia.

Most importantly, Nokia realized that there was a need for tools that fitted the domain, rather than for tools which required the domain and organization to change. A team in the Advanced Development Group had already developed a phone architecture, components and code generators to partly automate creation of mobile phone user-interfaces. A graphical design tool was needed to obtain the full advantage from these other tools. Hence, they decided to undertake the development of their own CASE solution.

THE METACASE SOLUTION

Quite early on, Nokia chose a metaCASE approach. There was a strong need for method flexibility — as the domain evolves, so should the method — and for applying the new development objectives quickly. Among a number of metaCASE tools evaluated, MetaEdit+ was selected: "It was the most flexible, allowed us to define our own design syntax quickly, and test ideas quickly while developing the method," summarized David Narraway. Report generation capabilities and links to code generation were also highly valued.

The metaCASE architecture behind MetaEdit+ products is divided into two separate tools. Method Workbench supports method development and stores the method as metamodels in the repository. MetaEdit+ reads these metamodels and provides customised CASE functionality for modelling with that method. This is a groundbreaking feature of MetaEdit+: users can model directly with domain concepts and rules, no extra mapping from a domain to a modelling language is needed. This fundamentally changes the way models can support communication, problem-solving, designing and coding.

In Nokia, Method Workbench was used to model the concepts and rules of the mobile phone domain, its graphical notations, code generators and document generation templates. "We can do miracles with Method Workbench," summed up Software Technologist Jyrki Okkonen, one of the key persons behind Nokia's method.

BENEFITS ACHIEVED

By developing and implementing their own method in MetaEdit+ Nokia has achieved what they were seeking.

Productivity increased by a factor of 10

Order of magnitude productivity gains: even early phases of tool introduction showed dramatically shorter development times. "A module that was expected to take 2 weeks even with the new tool now took 1 day from the start of the design to the finished product", said David Narraway.

Focus on functionality not on implementation

The domain-oriented method performed well: It allowed developers to concentrate on the required phone functionality and shifted the focus from code to design. New design requirements were easy to incorporate into the method thanks to the flexibility of the metaCASE tool. Method changes in traditional design tools would require programming and sharing of new tool versions, whereas MetaEdit+ allowed quick changes to the method in its form-based tools, and even automatically updated models made with earlier method versions.

Full code generation from models

Results from code generation were astonishing. "In many cases we can generate 100% of the required code. This is a result of innovative method development: domain-oriented metamodels provide the ideal way to link designs to code and to software components. Because all the work is done in the design phase, traditional programming is not needed anymore," says David Narraway.

Improved documentation

The documentation generated by MetaEdit+ followed Nokia's standards. As such it was applicable for review meetings as well as for full product description. Automated document generation naturally saved time and improved consistency and standard compliance.

Support for learning and introducing new developers

Other improvements were related to training costs and introduction of new team members. Because the method fitted the domain, all the developers found it easy to adopt: it already included the concepts and vocabulary they were familiar with. The learning curve of newcomers went down from 6 months to 2 weeks, and training costs were cut to a fraction. The main reason was the domain-oriented method: new developers do not need to understand all the details (or even look at the code) as the method directs and the tool enforces the development at higher levels of abstraction.

CONCLUSION

In retrospect, all the objectives for an ideal design tool were satisfied by MetaEdit+. Nokia now has in use a method which fits their domain seamlessly. They have an environment that automates their design work, generates most of the product code, and structures the development process. All this has improved the productivity of development teams and given Nokia first mover advantage, allowing it to demonstrate full working prototypes of next-generation phones while its competitors can only show empty shells.

In the future, the flexibility of MetaEdit+ allows Nokia to improve their development process and methods while still supporting the development of current phone models. "By implementing our own methods into MetaEdit+ we have obtained a flexible development environment which fits our needs," concludes David Narraway.

YOUR NEXT STEP

Visit us at http://www.metacase.com and tell us how we can revolutionize your software development!

MetaCase
info@metacase.com
www.metacase.com