

Work smarter not harder!

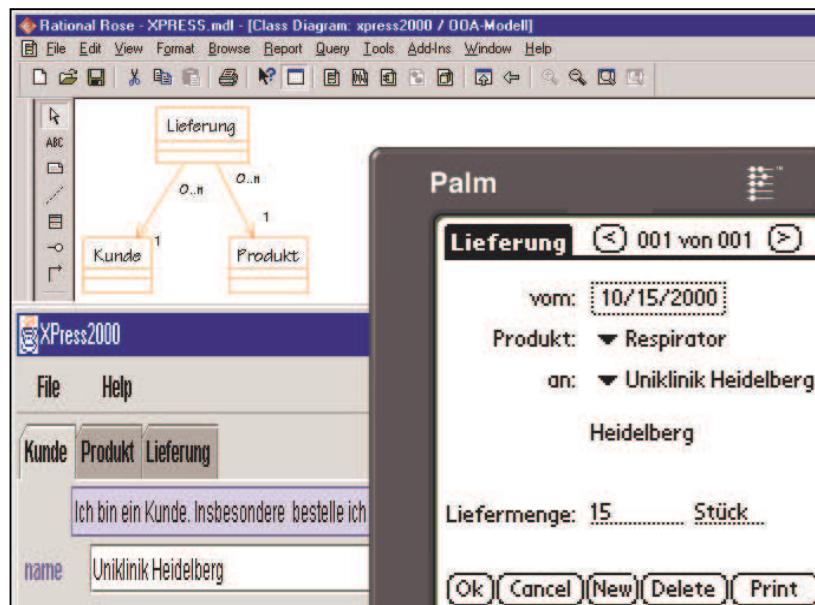
- Save up to 90% of your encoding-effort
- Let the computer finish boring work at high speed
- Optimize dates, costs and time simultaneously

Reduction of manual development-work

With smartGENERATOR you get the suitable generation-program for your UML-based software-plans (independent of your platform). Your UML-tool is connected through XML.

Executable application at the press of a button

You obtain a complete set of source code files respecting your pre-set rules. The application can be compiled, is executable and already possesses all required interfaces. You merely programme the functions as usual. By separating content and technical architecture you succeed in creating a user-specific, executable and solid application at the press of a button. You decide what is generated and how.



Pic.: Applicationframework for Palm/C and Internet/Java made with smartGENERATOR

Quality of development

- Every generator run underlies the "transaction-principle": All source-files are modified or none (if an error occurs)
- High maintainability of generated generators, as they exist as java-code
- Independent maintenance of smartGENERATOR is possible
- Support of separation between architecture and content, support of architecture-centreing development

Acceleration of development

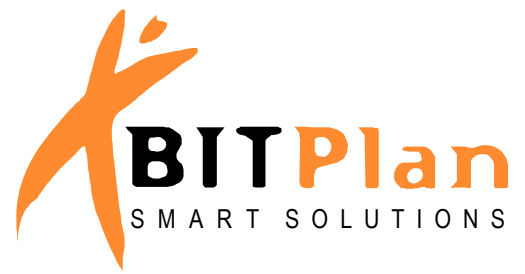
Adaptation and modification

Through templates the generator is flexible to your demands, your environment and particularly customizable to your software-architecture.

- Templates can be created for any architectures, even for IMS
- One-time production of templates, reproductions of unchanging quality as many as you like
- Reproductions everywhere, similarly and in high speed

smartGenerator

The professional solution for software development



Flexibility of development

- Second generation's generator, generic production of Java generators via BITPlan smartGENERATOR
- Acceleration of software development by reduction of encoding-effort
- Creation of complete patterns of architecture, Source Code
e.g. for Java, C++ or COBOL, types of data bank or HTML documentation and more

Examples for supportable architectures

- Web
- Dot.net
- EJB
- Mainframe/COBOL
- CORBA
- Palm OS
-

Integrated tools

- Rational Rose
- Poseidon Gentleware
- microTOOL objectivF
- Together Soft Together/J oder C++
- Select
- Object Domain of Object Domain Systems
- JDBC- sources and other Repositories

Economisation of development

- Created code will be licence-free
- Independence of tools, no expensive adaptations necessary if tools have to be changed
- Reduction of development-effort

Examples for supported languages

- Java
- C#
- C++
- C
- Delphi-Pascal
- COBOL
- CORBA-IDL
- HTML
- XML
- SQL-DDL
-

The introduction of tools for professional software development has three aspects:

- the strategic decision for a certain technology
- the costs
- the time

At none of these aspects you will have to abandon tool-support in your projects.

