

Rearchitecting by ReARC Software GmbH

Transforming a Legacy Application into a New Platform Independent Technology like SAP R/3

2. Workshop Software-Reengineering

Short Abstract

Roland Bechtel, Chief Architect (Compiler)

Roland.Becht@ReARC.de

ReARC specializes in the automated transformation from COBOL mainframe applications into the SAP R/3 system. This presentation should give you an insight of the methodology and ideas behind the transformation and the ReARC tools to automate this transformation in a high percentage.

If needed: short introduction into the SAP R/3 system, architecture and main methodology.

The complete process of a transformation, starting at the first analysis and ending with the new system:

The first step in realizing a transformation project is to get a very good understanding of the legacy system. To get this information we usually have to do the following steps:

- A ReARC Analyser tool shows us the program structure (call tree) of the COBOL programs, for each single COBOL program we get the GOTO graph, before and after the automated elimination, and we get a statistics about all the keywords used. Effort so far: some hours (no tools have to be adjusted)
- A solutions workshop together with the original programmers (if still available) determines the exact grammar used, the bottlenecks, a rough time frame

After the analysis the real parsing of the COBOL files can begin. This step is divided into several parts:

- Precompilation of the COBOL sources to get an almost LR(k) COBOL grammar
- Interactively adjust the Scanner and the Grammar Rules (about 1500) with the ReARC BNF grammar editor
- Generate an executable Lexer and Parser (with Flex and Bison)
- Parse the COBOL sources and generate a ReARC special intermediate format

In parallel to the tools adjustment the Data Migration is prepared:

- First out of the old (often hierarchical or even network) data structure the new relational data structure has to be developed
- The old data has to be migrated into the new relational data base
- The data manipulation commands in the COBOL source code have to be translated into relational queries for the ABAP sources

The next step is to massage and restructure the syntax free intermediate code. This is done by the ReARC Rearchitecting Editor as GUI for the Term Processor Kimwitu, a list based term manipulating tool developed by the University of Twente in Enschede, Netherlands.

Those Rearchitecting Rules include:

- Goto Elimination
- Data Base Calls
- Variable Transformations
- Global Include Files
- Screen Layout and processing Screens (DynPros)
- Report Generation
- Message Handling
- Unparsing into ABAP syntax

The result of the rearchitecting process is a SAP R/3 near structure of the former COBOL program that can be saved into the DB.

Unfortunately the last transformation step is not perfect, there are still parts of the new program you have to change manually. The ReARC Workbench for SAP R/3, our next tool, was implemented to give the programmer the opportunity to intervene and to change the ABAP target code or to change complete development elements. Those changes are mainly:

- Change the generated screen layout
- Change the flow of PBO and PAI (the screen flow)
- Tune the DB calls
- Insert Menues and Message types
- ...

So far we haven't touched the SAP system at all. The SAP workbench, however, was not built to add new modules from outside, like you can do with API calls or other interfaces. Therefore we had to implement a SAP Generator generating ABAP code into the R/3 system, either transactionally with feedback of syntax errors or synchronously.

This Generator works hand in hand with a Test and Trace Monitor. Although SAP already provides a Trace Monitor, the ReARC Monitor does not only test more events but provides also a graphical output of the trace.

This Monitor was necessary to assure the quality of the transformation step. As a high percentage of the transformed legacy application could be rearchitected in an automated fashion the quality assurance concentrates on proofing that the tools worked well, not that the resulting program works well (it is more or less a one to one transformation, errors in the source program will result in errors in the target program).

So far the legacy application was only rearchitected into the SAP system, but not reengineered. Of course we have the opportunities of adding new SAP features like Workplace Integration, Backoffice Integration or Internet Connection.