INFINIAM™ ANALYTICAL SERVICES

G4 Platform Technology

- Complete source code decomposition into a Hyper Model.
- Detailed software analysis and software quality measurement.
- Automated conversion of large and complex software systems.

Introduction

The G4 platform is a proven software technology platform specifically developed for software analysis, software quality measurement and automated application migration of large and complex software and database systems consisting of millions of lines of source code in different programming languages.

The G4 platform combined with our unique G4 project and G4 testing strategies allows for large and complex applications to be migrated quickly and efficiently by means of a well managed process to new modern "cloud compatible" platforms that are based on the latest programming languages, databases, architecture and frameworks in accordance with the clients specifications.

The migration is done without interruption of ongoing maintenance and orders of magnitude more cost effective, faster and without risk compared with current mostly manual methods. Even complex migration projects can be offered on a fixed price and fixed time engagement.

After project completion a complete set of working, tested and maintainable source code is delivered that is compatible with the selected platform specifications and that is functionally equivalent to the tested and proven source application. The delivered source code can be further maintained by the client without any further dependencies on Cornerstone Technology using the clients in-house Integrated Development Environment (IDE).

1. Replatforming technology

1.1 Premises

The G4 replatforming technology is based on the following premises:

- Full source code decomposition to meta information in the form of the so called Hyper Model. The meta information is uploaded in the G4 Repository for detailed analysis. The Hyper Model is the input for all further processing.
- Automated conversion of business logic, screens, jobs, databases and data.
- The functionality of the application after conversion running on the new platform equals the functionality before conversion. Business rules are for 100% retained and transferred to the new platform
- The conversion is implemented through so called conversion rules, each translating a concept: a statement, a data type or a construct. Typically there are a few dozen conversion rules implemented in a convertor.

1.2 Source code decomposition

Every project starts with source code decomposition, whether the goal is software conversion, software analysis or software quality measurement. Software is written in a certain language according to the grammar of that language. In most application environments multiple languages are used. For example a typical mainframe business application can be developed in COBOL, CICS, IDMS, JCL, Assembler, SAS and REXX.

Decomposition involves processing the source code through a standard process chain as shown in the figure below.

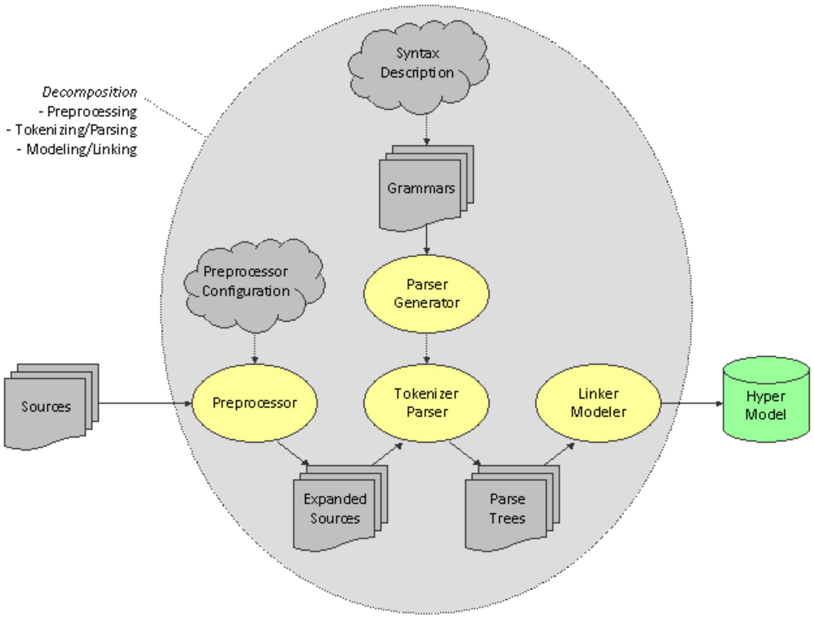

Figure 1: Source Code Decomposition

The processing steps are: pre-processing, tokenizing and parsing, linking and modeling. The necessary components for decomposition are all part of the G4 Platform. In fact, they are considered the kernel of the G4 Platform.

Per language the decomposition process can be configured. For instance, by describing the grammar of a language, a parser for that language can be generated dynamically. For configuration and verification purposes, process steps can be executed separately. Eventually, the decomposition processes are incorporated in converters, loaders, etc.

The goal of decomposition is retrieving as much meta information as possible from all types of source code of an application. Syntactic information, code structure, relations, including relations between different types of sources, and much more is stored in the Hyper Model.

1.3 Further processing

The Hyper Model is input for further processing, see the figure below:

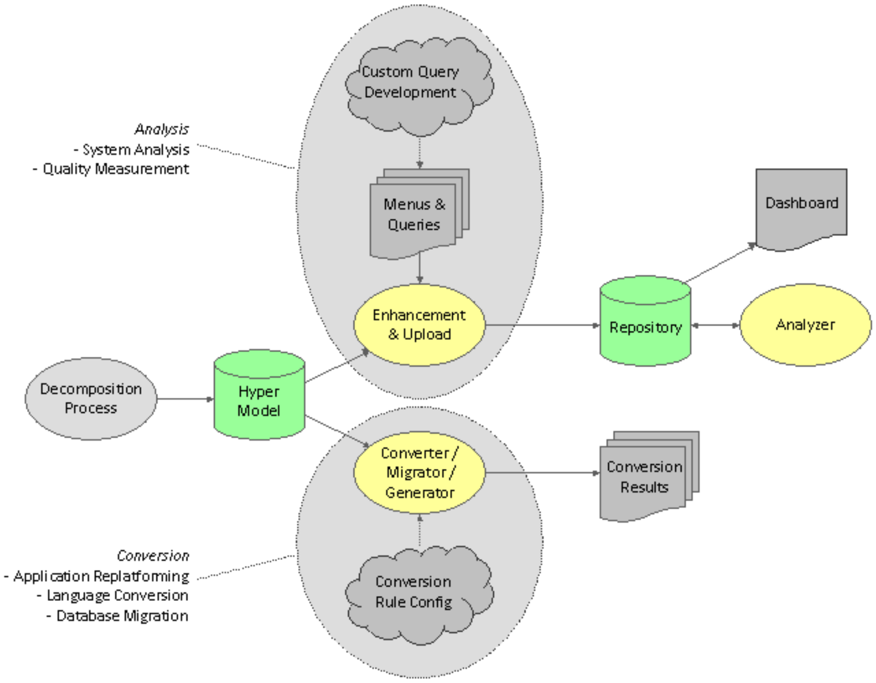

Figure 2: Processing After Decomposition

Further processing can be analysis: enhancement of information and upload in the G4 Repository. The G4 Repository is used for interactive analysis with the G4 Analyzer. An integral part of the repository are the predefined queries and the menu system. Through the queries, information can be brought up about where something is used, what are the system wide dependencies, and much more. The G4 Repository is also used to generate a G4 Dashboard, which holds many business and technical quality metrics including ISO software quality standards for analyzability, changeability, stability and testability.

Beside the fact that the G4 Repository, G4 Analyzer, and G4 Dashboard are products, they are essential for replatforming support. Especially the analyzer will be used to analyze the original application. For example the CRUD matrix provides information for project clustering and modularizing the new application. The use of specific constructs can be looked up for the reference architecture or for prioritizing conversion rule implementation. The already extensive set of queries can easily be expanded, if necessary.

Further processing can also be conversion. With conversion is meant here: any automatic transformation of source code, whether it is language conversion, (database) migration, beautification, restructuring, or as part of replatforming.

Conversion leads to conversion results, which is nothing more than idiomatic source code, but of course different from the input source code that is compatible with the new platform.

Conversion is driven by so called *Conversion Rules*. Each conversion rule converts a certain concept or construct of the source language to an equivalent concept of the target language. Examples of concepts are: structure, statements, or combinations of statements, expressions or computations, identifier names, data types, database access. For each of these concepts there will be a conversion rule translating it. The translation of some concepts may be combined in one conversion rule, or a very complex concept may be divided across several conversion rules.

It is important to notice that the total set of conversion rules is limited: several dozens to a little over a hundred, because of the fact that conversion rules are generic. Which means that a single conversion rule converts all occurrences of a construct. This is many orders of magnitude less than the number of lines of source code to be converted.

1.4 Replatforming example

The functionality and behavior of a computer application is determined by programs that are written in one or more programming languages. Typically these programs make use of functionality that is provided by the platform, available frameworks and possible third party software. All the necessary information describing the functionality of the application is available in de source code of the application including all dependencies.

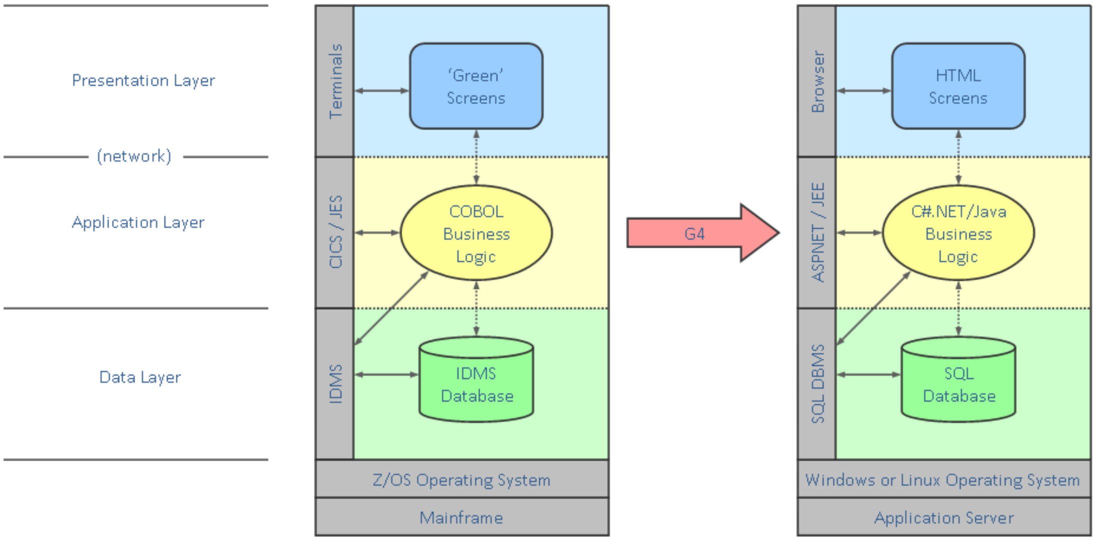

Figure 3: Example of a legacy Replatforming business case

Every migration project starts with creating a G4 Repository of the system. Using G4 Analyzer we get detailed facts on the complete system quickly and accurately, even for very large and complex systems containing many millions lines of source code in different languages. With our detailed platform knowledge we identify any dependencies and define with you the correct and preferred mappings between the platforms using automation. With our G4 platform we can address effectively the 3 possible mapping situations such as:

One to one mapping is possible.

Only a partial mapping is possible

There is no similar functionality on the new platform

The image below describes the different decomposition processes and the different conversions and migrations in a single schema.

For each language there is a separate decomposition process leading to a single hyper model. For each conversion/migration there is a separate process or converter/migrator. A single G4 Repository is made which is not shown in the picture below.

In figure 4, the input types of source code are shown: COBOL, including Copybooks, JCL jobs and procedures, BMS maps, and IDMS schema. Also the output types of source code are shown: VB.NET/ASP.NET, create script, and schema XML.

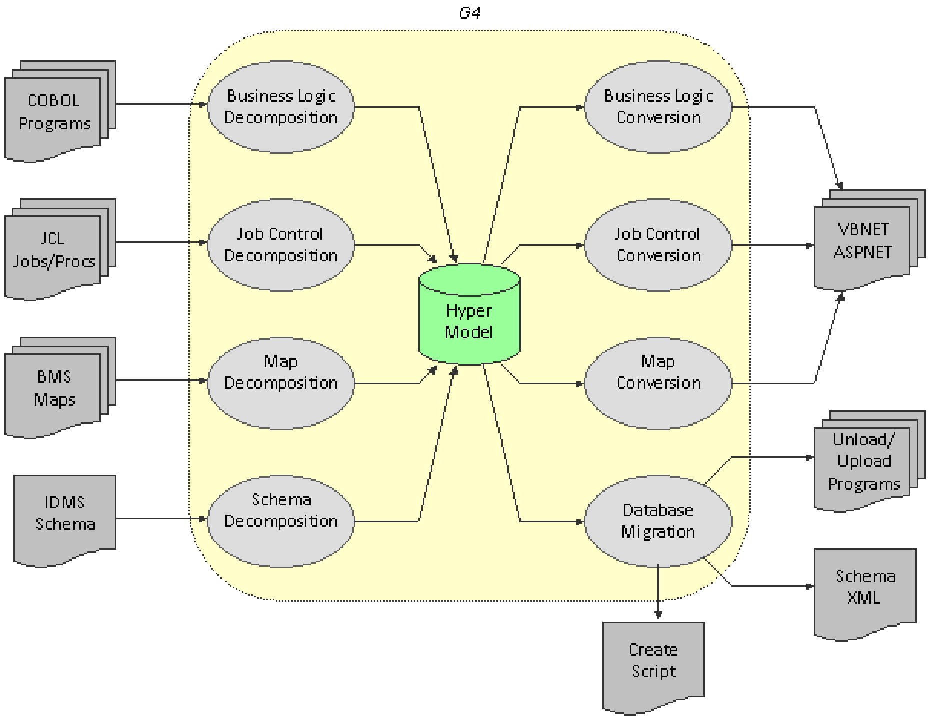

Figure 4: Sample Replatforming Processes.

After conversion a complete set of tested, maintainable and running source code is delivered that works perfectly with the new platform environment. The new source code provides the same proven and tested functionality as the original application running on the original platform.

In addition the convertor automatically creates the correct upload programs, unload programs and create scripts for the data migration.

The delivered source code can be further maintained by the client without any further dependencies on Cornerstone Technology using the clients in-house Integrated Development Environment (IDE).

2. G4 platform supported languages and databases

The G4 platform already supports over 96 different languages and database management systems. Adding G4 platform support for a new language or database can be done in a matter of days.

2.1 What defines a computer language

Applications are written in one or more programming languages that control the behavior of in particular computers while data for business applications are typically organized and stored by means of a database. A typical legacy business application is normally developed using multiple languages, for example COBOL, JCL, IMS and Assembler. For Cornerstone a language means any formal constructed language that controls the behavior of a computer system.

Over time many languages, already over 1500, have been developed and new languages are still being introduced. Many languages have over the years gone out of fashion and are no longer kept up to date, supported and taught. However there are still many, sometimes essential and critical, applications in daily use that have been developed using these now obsolete languages.

2.2 What G4 platform language support means

Adding G4 support for a new language means that the source code written in that language is made compatible with our proprietary Hyper Model through a process of decomposition. See section 1.2 for more information about the decomposition process. Once a new language is compatible with our Hyper Model all our available G4 platform analysis, reporting and conversion tools such as G4 Analyzer, G4 Dashboard and G4 Convertor can immediately be used enabling a high level of re-use.

2.3 Overview of G4 platform supported languages

We have G4 platform support for many program languages, screens, data definition languages (DDL) and job control languages (JCL) from different vendors like; IBM, Honeywell/Bull, Unisys, Tandem, Siemens, Fujitsu, Oracle, CA Technologies, HP, Cincom Systems, Progress Software, Gupta Technologies, Micro Focus, Microsoft.

2.3.1 Programming languages

ABAP (SAP AG)
ALGOL
AppBuilder (Blue Phoenix)
ASSEMBLER / HLASM (IBM)
C# C#.NET (Micosoft)
C C++ (Microsoft, HP, IBM)
CA Cool:Gen (CA Technologies)
CA Easytrieve (CA Technologies)
CA Ideal (CA Technologies)
CA VISION Builder (CA Technologies)

Clipper (Nantucket)
CLIST (IBM)
COBOL COBOL-74 COBOL-78 COBOL-85 (IBM, ICL, Honeywell, Bull, Siemens)
CSP (IBM)
MFCOBOL (Micro Focus)
DATABUS (Datapoint)
Delphi (Borland)
DSL (Customer Specifc)
EGL (IBM)
Fortran (HP)
Java SE / Java EE (Oracle)
JSP (Oracle)
MAGIC (Magic Software Enterprises)
MAGNUM (Tymshare, Allshare)
Mantis (Cincom Systems)
NATURAL (Software AG)
Oracle Forms (Oracle)
Pascal (HP)
Pick/BASIC
PL/1 (IBM)
PL/SQL (Oracle)
PowerHouse (Cognos Inc.)
Progress (Progress Software)
RPG (IBM, Unisys)
SAS (SAS Institute)
SCL (ICL, Fujitsu)
SQR (Oracle)
Team Developer (Centura, Gupta Technologies)
Topend
T-SQL (Microsoft, Sybase)
Uniface (Uniface)
VB / VB.NET (Microsoft)
VPLUS (HP)

2.3.2 Job Control Languages (JCL)

JCL (IBM, Bull)
JCL 7 (Bull)
CL (IBM)
SJCL (Siemens)
ZEKE (Altai)

2.3.3 Databases (DDL)

ADABAS (Software AG)
BDAM (IBM)
CODASYL
Datacom/DB (CA Technologies)
DB2 (IBM)
DB4 (IBM)
DMSII (Unisys)
DMS1100 (Unisys)
IDMS (CA-Technologies)
IDS/2 (Honeywell)
IMS (IBM)
Informix (IBM)
ISAM (IBM)
MySQL (Oracle)
NonStop SQL (Tandem, HP)
OracleDB (Oracle)
SQL Server (Microsoft)
Supra (Cincom Systems)
TOTAL (Cincom Systems)
TurboIMAGE (HP)
UDS (Siemens)
VSAM (IBM)

2.3.4 Online Transaction Processors

CICS (IBM)
Tuxedo (Oracle)
TP8 (Bull)
UTM (Siemens)