



15070 Royal Ave.
 White Rock, BC V4B 1L9
 Canada

Tel: (604) 538-5094
fbdris1@attglobal.net
www.fbda.ca

Natural/Adabas Transformation Matrix

Introduction

FBD Associates Inc. provides a variety of solutions for re-platforming Natural/Adabas applications. Table 1 provides a matrix of the transformations applied to each component of a Natural/Adabas application when implementing either a JavNat/DB2 transformation or an EGL/DB2 transformation, to generate an application that is functionally equivalent to the original Natural/Adabas application.

After the migration is complete the generated application can be then deployed on multiple hardware platforms (Java), or to a mainframe Cobol environment if EGL is the migration target. Table 1 also provides a indication of the extent of manual editing of the generated components that is required to complete the transformation subsequent to JavNat processing. Comments regarding the design features, testing and documentation of the transformed application are provided.

Natural/Adabas	JavNat Equivalent	EGL Equivalent
FDT (File Definition Table – physical file description)	DDL for target RDBMS. Fully automated – no editing. Customer selects table/column names.	DDL for target RDBMS. Fully automated – no editing. Customer selects table/column names.
DDM (Data Definition Module – logical file descriptions)	Java class on a one to one basis. Fully automated – no manual editing.	EGL sqlRecord on a one to one basis. Fully automated – no editing.
GDA, PDA, LDA (General, Parameter and Local Data Areas)	Java class on a one to one basis. Fully automated – no editing.	EGL BasicRecord on a one to one basis. Fully automated – no editing.
MAP (User presentation)	Java class on a one to one basis. Fully automated – no editing.	EGL TextForm on a one to one basis. Fully automated – no editing.
Program, SubProgram, SubRoutine (Logic)	Java class on a one to one basis. Fully automated – very limited manual	EGL BasicProgram on a one to one basis. Fully automated – very limited manual editing

Natural/Adabas	JavNat Equivalent	EGL Equivalent
modules)	editing of the generated Java code - principally to workaround flagged statements where direct translation not possible such as the Natural REINPUT statement.	of the generated EGL code - principally to workaround flagged statements where direct translation is not possible such as the Natural REINPUT statement. When INPUT USING MAP or inline INPUT statements are present, an EGL formGroup is also created, incorporating all the maps/data elements used in the module. Fully automated – no editing.
Copycode	Merged inline prior to transform.	Merged inline prior to conversion.
HelpRoutine	Java class on a one to one basis. Fully automated – limited editing of the generated Java.	EGL BasicProgram on a one to one basis. Fully automated – limited manual editing of the generated EGL code.
JCL	JCXML module on a one to one basis. JCXML is an XML file that specifies the same batch steps as the original JCL and is interpreted by a custom Java class to perform batch operations. Certain batch operations such as SORTs may be replaced by a Java equivalent.	JCL or JCXML depending on the EGL generated code target platform.
3GL Modules such as Cobol or Assembler	Replaced on a one to one basis with a Java class. In some cases the functions provided by the 3GL module may no longer be required.	3GL modules will be called directly from EGL modules.
Other	Base Classes – custom Java classes to provide syntactical behavior identical to Natural. Source code license provided as part of the transformation project.	No proprietary base classes required. Some EGL libraries will be included to support certain Natural functionality.
Testing	Testing strategy involves side-by-side screen comparisons and diffs of reports and generated work files.	Generally testing strategy involves side-by-side functional screen and report comparisons and diffs of generated work files. The EGL IDE provides a high productivity debug environment to minimize requirements for debug and test in the deployment environment.
Code Simplification	The Java code is generated on a statement to statement and module to module basis and thereby preserves the “look and feel” of the Natural application. No attempt is made to simplify the generated Java source code.	The EGL code is generated on a statement to statement and module to module basis preserving the procedural flow of the Natural application.

Natural/Adabas	JavNat Equivalent	EGL Equivalent
Business Rules	Preserves the business rules implemented in the Natural application un-modified.	Preserves the business rules implemented in the Natural application un-modified.
Application Resource Consolidation	Preserves the error messages and similar text information as they existed in the Natural source code – i.e. they appear in the Java code in the same location as in the Natural application.	Preserves the error messages and similar text information as they existed in the Natural source code – i.e. they appear in the EGL code in the same location as in the Natural application.
Scalability	Tuning and code enhancements may be required to allow some applications to scale up.	Since one of the targets of EGL is CICS/COBOL, no degradation in performance would be expected.
Maintainability	<p>The architecture as well as the look and feel of the legacy Natural application are preserved thereby facilitating on-going maintenance of the JavNat application by the customer's existing development staff with minimal Java development experience.</p> <p>The design history and documentation of the Natural application are preserved and can be useful in maintaining the JavNat application.</p>	<p>The structure and flow of the original application is maintained, the variable and functions names are retained, and the EGL syntax is comparable to Natural.</p> <p>The IBM Rational Software Development platform is a robust IDE which offers debugging, syntax prompting and error flagging.</p>
Source Configuration	Jave IDE project source files.	EGL source files.
Documentation	JavaDocs to include preservation of comments from Natural source.	To be determined after discussion with IBM.
Open Source Support	The JavNat base classes are proprietary but the source code files for these classes are delivered and can be enhanced as required. Alternatively software maintenance support can be acquired from FBDA.	N/A
Complexity	As for the Natural application.	As for the Natural application.
Security	Preserves the security rules of the Natural application un-modified.	Preserves the security rules of the Natural application un-modified.
Comments:	<p>Provides the lowest cost, least risk transformation.</p> <p>Retains the architecture as well as the "look and feel" of the Natural application; facilitating on-going maintenance by existing staff.</p>	<p>Provides an intermediate cost, low risk conversion option.</p> <p>Through its generation options, provides cross-platform solutions. EGL can generate both Java and Cobol run time code from the same EGL code base. The ability to generate</p>

Natural/Adabas	JavNat Equivalent	EGL Equivalent
		<p>Cobol greatly facilitates a migration in place to allow execution on the existing mainframe platform where that is the most economical choice.</p> <p>Offers a modern web development environment and allows for easy development of Web Services.</p>