

FORM Release Notes

Release R1.1 [980802]



ChiMu Corporation

1220 N. Fair Oaks Ave, #1314
Sunnyvale, CA 94089

Phone: 408 734-9068

Email: info@chimu.com

www.chimu.com

Release Notes

This document contains release notes for FORM.

Release R1.1 [980802]

The following enhancements have been made in the releases since R1.0.2

1. All the packages use a lowercase 'com' as a prefix to match the now-JavaSoft-approved (and ChiMu's original) logical domain name prefix.
2. The FORM engine and Database Descriptions understand schemes and catalogs. A default prefix (e.g. 'Scheme.Catalog.') can be added to all tables or individual tables can be assigned different schemes and catalogs.
3. Examples now work off textual scheme descriptions so they can be modified to try out different database variations.
4. Statements are now explicitly closed whenever exceptions are thrown by the database/JDBC layer.
5. The Kernel Frameworks are integrated in with the JDK 1.2 collection classes' protocol. When JDK 1.2 is finally released, FORM will convert over to full collection support.
6. The Kernel Frameworks look for the lowercase prefixed version of JGL (i.e. 'com.objectspace.'), to see if JGL is included on this system.
7. Retrieval-on-demand has been enhanced. FORM has always supported:
 - a) Retrieving Forward Associations on-demand
 - b) Retrieving to-many Reverse and External Associations on-demandFORM now supports
 - c) Retrieving to-one Reverse and External Associations on-demandThis is made possible by using an ObjectHolder for the instance variable. The ObjectHolder will evaluate a Functor to fetch the desired object the first time the ObjectHolder's value is asked for.
8. Some caching fixes were improved so if an object is ever read in its entirety (and not just by the Stub itself), the existing Stub is inflated with the available information.
9. You can now provide custom overriding of the Form Preprocessor to generate different FormInfo files and Binding sections. All binding init and extract statements can be overridden to call conversion functions. FormInfo files can be generated with standard "name-mangled" column names so hand mapping is not required.

Release R1.0.2 [971203]

The DB2 database description was added and control over the length of generated query variable names is available through QueryPack.

Release R1.0.1 [971106]

The examples were cleaned up and made more general. They now rarely look for specific data that was placed by the population scripts, so they now function when the database state has been changed (e.g. objects deleted).

Release R1.0

All the previous release notes (for during final beta) were removed. To summarize some of the final enhancements or notable functionality:

1. Extensive control over refreshing and caching have been made public to FORM clients.
2. WeakReferences are available for both the Sun and MS VM. WeakReferences are not part of the current Java specification, so our implementation must rely on VM specific functionality. We call this "110% Pure Java": the code is not VM dependent (i.e. it works on any VM) but we take

advantage of VM specific features if a particular VM is present. FORM also comes in a 100% Pure Java version that does not include any VM specific functionality¹.

3. FORM completely avoids using JDBC drivers to handle datatype conversions during writes. Too many (i.e. almost all) JDBC drivers were failing to properly convert data types so the FORM Database abstraction layer now handles the conversions itself.

¹ The 100% Java JAR is not part of the standard distribution but is available upon request.