



---

## Migration Re-Defined

ORMIT is clearly **The Tool of Choice for Forms Developers** looking to migrate their forms & reports applications from Oracle Forms version V2.x/3.0/4.5/5.0/6x to 9i/10g web-enabled Form applications that can be deployed on Oracle 9i/10g Application Server.

### ORMIT vs. Manual migrations

ORMIT is a very powerful yet easy to use tool. ORMIT helps save migration, enhancement & reengineering related time and costs significantly. Organizations can procure ORMIT and do their migrations themselves with no dependency on external consultants if they so desire.

Here are some benefits of using ORMIT that sets ORMIT apart:

**Easy to Use:** The intuitive and easy-to-use UI provides for a low learning curve. In fact, anyone who has used Forms Builder could easily start using ORMIT in just a few hours.

**Fully Automated migration:** The migration process using ORMIT is fully automated. The postmigration enhancements, re-engineering and maintenance could also be automated to a greater degree.

**Clean-Compliant Code:** The entire migration process using ORMIT strictly adheres to Oracle recommended best practices and processes. ORMIT does not introduce or encourage any proprietary code.

**Preserve Business Rules/Logic:** ORMIT allows Developers to migrate their Forms to new versions without changing any business rules or logic so Developers do not need documentation for the application prior to the migration.

**Oracle Validated:** ORMIT is a product that has been tried & tested and validated by Oracle.

**Language Independent Migration:** ORMIT allows for multi-language migrations. It has been tested on many different languages including Japanese, Arabic among others.

**Compare different versions of forms:** This powerful utility provides an option to compare and extract the differences between older version code and migrated Form code in Triggers and Procedures, thus saving the development team the effort to look for differences long after the migrations are done.

**Address Multiple Forms:** ORMIT allows Developers to address multiple forms for migration in one single step. Post-migration fixes can also be made to all or a selected set of forms simultaneously, resulting in significant time savings.

**Simulated Migration:** A simple configuration option allows the user to simulate the migration process and generate a pre-migration check report. This report allows tweaking the actual migration and increase migration automation level.

**Documentation:** A migration report is generated at each step of the migration. This documentation allows users to visit the migration changes long after the migration is done. In addition, ORMIT places a date & time stamp for code level changes made using ORMIT (replacing/adding strings, attaching libraries etc.)

Over and above some of these, ORMIT users derive direct tangible benefits to their migration efforts at every stage of migration.



## VALUE OF ORMIT DURING DIFFERENT STAGES OF MIGRATION

### Pre-Migration/Analysis Stage

Analyzing a migration is an important task that needs to be performed mandatory. Depending on when the forms applications were written, there may be situations where the original set of developers may not be around in the organization anymore. Even if the migration team consists of developers with the functional knowledge of the forms, manual analysis of the forms is a task that can take 2-3 weeks of effort.

Using ORMIT pre-migration utilities, this task can easily be cut down to a maximum of 1 week, while generating tangible reports helpful for planning the migration stage well.

Some of these include:

**Simulated Migration:** A simple configuration option allows the user to simulate the migration process and generate a pre-migration check report. This report allows tweaking the actual migration and increase migration automation level.

**Pre-migration Report:** The pre-migration report is generated in Excel. The issues could easily be sorted for similarity such as obsolete calls, host calls, user exits, scroll bar, mouse navigation, run-report objects to name a few. The migration can then be planned keeping in mind common items and how to address them during the process of migration and during post-migration. Based on this report the modules could be divided among team members while also providing them with specific tasks/functions along with the documentation.

For eg. Applications developed in v6.0 or prior in character mode, it is recommended to simulate the migration process to ensure visual attributes such as font, size, column width can be addressed during migration itself.

**Configuration options:** ORMIT provides users with options to configure migration settings based on the pre-migration report and the simulated migrations performed. For eg. the length of fields could be increased automatically during the migration process. Configuration options can be set for each of the migration stages of versions 3.0 to 4.5, 4.5 to 6i or 6i to 9i.

**Re-apply migration settings:** ORMIT allows users to save migrated forms into separate folders than the original forms, so the original forms are never over-written. Migration setting configured during the simulated migration can be re-used over and over again to avoid errors, as well as help get better results during each iteration of migration, if so desired.

**Determine the order of migration:** Forms based applications are typically developed using references and dependencies on other Forms, PLLs and objects. ORMIT provides an analysis report where users can gather at a single glance the dependencies and references each form has on an associated object. Users can then use this analysis report and approach migration with the specific priority.

	Manual	Using Ormit
<b>Description</b>	per app	per app
<b>Planning/ Analysis</b>		
Forms and Report Migration	55 Days	55 Days

*This figure shows comparative time for Manual vs. Using ORMIT for 400 Forms, 150 reports.*

**DIRECT IMPACT: TIME & COST SAVINGS**



## Migration Stage

The **Migration Wizard** is a very powerful feature of ORMIT. Developers are able to use this wizard to migrate older versions of Forms to the next version based on user preferences and inputs.

The migration can be performed on several forms at once or one at a time. This can all be done without opening a single form. This in itself is a significant time saving as compared to the process of opening the forms in Forms Builder.

**Some of the other features within the migration wizard which could be applied in one step include:**

- Migrating from character based forms
- Converting V2 trigger to V3 \ PL/SQL.
- Replace PL/SQL and Oracle reserve words
- Address all issues related to built-ins, system variables and compatibility
- Takes care of Window titles and Root Window problems
- Modifies old pop-ups into stacked views
- Existing PL/SQL library can be used during migration
- Facility to transfer messages to alerts
- Supports Menu and PL/SQL library migrations
- ORMIT handles all type of canvases (stacked, tabbed etc.)
- Handles Multi-Lingual forms

Log Files: Detailed Log files are generated at every step of migration, which records all the activity during migration. These serve as documentation for the future.

Above all these migration features, the migration process adheres to Oracle's recommended best practices. Since ORMIT does not change any business logic, there is no requirement for functional documentation to perform migration.

	Manual	Using Ormit
Description	<u>per app</u>	<u>per app</u>
FORMS Migration	240 Days	4-5 Days

*This figure shows comparative time for Manual vs. Using ORMIT for 400 Forms, 150 Reports.*

### **DIRECT IMPACT: TIME SAVINGS**

## Post-Migration Fixes

Using built-in wizards, Developers can manipulate Form Triggers, Objects and Properties and apply them across selected forms or all forms at once.

**Property Wizard** assists Developers in modifying the property of any Form Objects based on user defined input or templates. It has the advantage of handling multiple forms at the same time. One click is sufficient to change the Form Objects property on single or multiple form(s).



**Trigger Wizard** provides Developers with options to Add/ Replace/ Append to Existing /Delete a String in a trigger. It also provides Developers with the ability to create a new trigger at Form, Block or Item level or Replace/Delete an existing trigger.

**Object Wizard** is used for sub-classing or removing the sub -class from any Form Objects for a given Form(s). It simplifies the task against a manual sub-classing effort.

**Comparison Wizard** provides an option to compare and extract the differences between older version code and migrated Form code in Triggers and Procedures.

**Find and Replace** is one of ORMIT’s most powerful features. Using Find & Replace, Developers can search for single or multiple strings within Forms and then choose to replace them with a customer specified string. Find and Replace also provides a search report which proves useful for developers searching for a specific application’s specific text/strings/calls. The Find relationship option allows Developers to search for dependencies and references within Forms along with providing detailed reports.

**Mouse Navigation:** A PLL along with ORMIT is provided to customers who desire the need to implement Mouse navigation. This PLL can be further customized by the customer or be used to re-create a PLL that is best suited for their application. Using ORMIT applying mouse navigation to multiple forms, blocks & items becomes a breeze.

Using ORMIT Find utility, users can quickly create reports for Property attributes, as well as perform a Find & Replace similar to Microsoft Word, for properties such as color, font-size, width etc. at the item property level. This coupled with the fact that it could be done on multiple forms at a time, reduces manual effort tremendously in the post-migration stage.

There are several other possibilities that make life a lot easier for developers in the post-migration stage such as Scroll Bar, Attach/Detach libraries/packages/objects, applying multiple changes to multiple forms at once, Dynamic Screen resolution, Stacked Canvas Re-positioning.

In all these cosmetic changes can prove to be very tedious if applied manually to each and every form. ORMIT not only makes it easier for developers, but also provides the organizations with a ready documentation so as to enable the corporation to re-visit the migration at a later time.

Description	Manual	Using Ormit
<b>FORMS</b>	<u>per app</u>	<u>per app</u>
<b>Re - Engineering/Post-migration Enhancements</b>  Attach/Detach libraries, references, Objects,obsolete Call, Prompt Scroll Bars, Mouse Navigation Screen/Canvas Resolution, fonts,	320 days	8 days
<b>Migration related documents</b>	40 days	-

*This figure shows comparative time for Manual vs. Using ORMIT for 400 Forms.*

**DIRECT IMPACT: TIME & COST SAVINGS**

## Reports Migration

ORMIT migrates Oracle SQL Writer Reports version 1.1 or Developer Reports 2.5 to 6i and 9i /10g that can be deployed on Oracle 9i /10g Application Server. All the well-known issues are addressed by ORMIT during this migration process.

ORMIT performs the migration steps in a user-friendly batch mode when compared against manual conversion process.

	Manual	Using Ormit
<b>Description</b>	<u>per app</u>	<u>per app</u>
<b>Reports</b>		
<b>Migration</b>	68 Days	2 Days
<b>Post Migration Fixes</b>	100 Days	6 Days
character to web, deployment Run_Report calls		

*This figure shows comparative time for Manual vs. Using ORMIT for 150 Reports.*

ORMIT Advantages:

- Direct web deployment
- Maintains existing behavior and functionality
- Complete XML metadata support
- Refined reports with complete GUI and Internet features
- Descriptive log files
- 

ORMIT also provides specific options that enable users to rapidly address RUN\_REPORT Calls, in the post migration stage. Using ORMIT, developers can easily identify the instances of these calls and apply the new modifications easily while saving time.

### **DIRECT IMPACT: TIME & COST SAVINGS**

## Testing Stage

The final step before QA/pre-production deployment is the testing stage. Due to the fact that ORMIT does not change any business logic, ORMIT reduces the time required for unit/functional testing. All changes/modifications made using ORMIT are date & time stamped. Conversion wizards allow view of changes between before & after modes. An organized and easily sort able documentation in XLS format allows testers to rapidly test and address any issues at this level easily.

	Manual	Using Ormit
<b>Description</b>	<u>per app</u>	<u>per app</u>
<b>Testing</b>		
<b>Developer/Unit Testing</b>	55 Days	11 Days
<b>Functional Testing</b>	30 Days	30 Days

*This figure shows comparative time for Manual vs. Using ORMIT for 400 Forms and 150 Reports.*

### **DIRECT IMPACT: TIME SAVINGS**