

ORACLE **White Paper**

# **Designer/2000**

## **Release 2.0 Product Overview**

**ACHIEVING 100% GENERATION**

**VERSION 1.0: MARCH, 1996**

March 1996

Authors: Simon Stow, Mark Pirie

Contributors:

Copyright © 1996 Oracle Corporation.

**All rights reserved. Printed in the U.S.A.**

**This document is provided for informational purposes only and the information herein is subject to change without notice. Please report any errors herein to Oracle Corporation. Oracle Corporation does not provide any warrants covering and specifically disclaims any liability in connection with this document.**

---

## Introduction

Designer/2000 Release 2.0 will provide previously unobtainable levels of productivity in the design and generation of client/server systems. Release 2.0 offers dramatically increased generation capabilities, both in terms of the targets for generation and in terms of what is generated for each target. This paper introduces the new server generators (for Oracle Web Server, Oracle Rdb, Oracle Lite, DB2/2, Microsoft SQL Server, Sybase, Informix, ANSI and ODBC databases) and the major new features of client generation: application logic and reusable module components. The new features will enable the 100% generation of sophisticated client server applications and provide many options for deployment.

Designer/2000 R2.0 provides a single, comprehensive management environment for the design and generation of client/server systems. Key is the introduction of *application logic* custom code which may be recorded as part of the system design in the Designer/2000 repository. Generators then incorporate the application logic into the generated system, thereby removing the need to modify generated programs other than through Designer/2000.

Release 2.0 offers yet another quantum leap in productivity and usability. *Module components* (parts of program specifications) can be reused and shared between modules: generators then create reusable 'objects' as appropriate to the target environment(s). A new *Design Editor* provides a single, fully integrated design and generation environment for both client and server. The Design Editor has many new user interface features to maximise ease-of-use and to minimise the 'learning curve'.

### Designer/2000 Release 2.0 Overview

Release 2.0 of Designer/2000 has a number of major new features:

1. Extended Server Generation
  - Support for Oracle7, Oracle Lite, Oracle Rdb, Oracle Web Server
  - Support for non-Oracle Databases: DB2/2, Microsoft SQL Server, Sybase, Informix
  - Support for ANSI SQL DDL, ODBC Databases
  - Generation of Oracle7 database triggers to implement domain and arc validation, column value derivation, journaling, etc.
  - API generation of insert, update, delete and lock procedures for tables
  - Full support for Oracle7 symmetric replication: the design and generation of distributed, replicated servers
2. New Client Generation Features
  - Application Logic - storing PL/SQL, Visual Basic, Oracle Basic and Java code
  - Design Reuse - Module components shareable between module definitions
  - Co-operative Generators - Allowing seamless generation of modules integrating Forms, Reports and Graphics components.
  - Power Objects Generation
  - Web Applets Generation
3. Design Editor
  - A fully integrated Design and Generation environment providing a single User Interface for integrated design of client and server
  - Usability features include: wizards, guides, drag and drop, edit-in-place, synchronised updates and right-mouse menus.

In addition, Designer/2000 Release 2.0 will offer full support for, and compatibility with, Developer/2000 Release 2.0 as well as upgrades for all the existing process modelling, systems modelling, design and generation tools available in Designer/2000 Release 1.3.

---

## Key New Features

### Open Server Generation

Designer/2000 Release 2.0 will fully support the definition, generation and reverse engineering of Oracle 7.3, Oracle Lite, Oracle Rdb and Oracle Web Servers.

In addition, Release 2.0 will allow users to define and generate database designs for DB2/2, Microsoft SQL Server, Sybase and Informix. The design of existing databases created within these database systems can also be reversed engineered into the Designer/2000 R2.0 Repository.

Support is also provided for other database types via the generation of ANSI-compliant SQL Data Definition Language (DDL) files. Release 2.0 will also be able to open any ANSI-compliant SQL DDL file and capture the database design direct from the DDL commands. Release 2.0 can also generate and reverse engineer ODBC-compliant database designs via ODBC connections.

These new open server generation and reverse engineering option allow Designer/2000 R2.0 to be used as the single definition and unifying management control for systems implemented across many databases from many vendors. Further, existing systems can be rapidly and reliably re-engineered using the Designer/2000 R2.0 Server Generation capabilities.

### Server API Generation

We will be adding the ability to generate an API for a table. This will result in a server-side package being generated which contains procedures for insert, update, delete and lock. In Developer/2000 it will be possible to base a forms block on different procedures for query, insert, update and delete and this enhancement will allow those procedures to be generated. This feature will also provide an API which can be used by other programs for functions packaged application integration.

### Application Partitioning

Previous versions of Designer/2000 and Oracle CASE have supported the ability to define and generate database triggers. In Release 2.0 of Designer/2000 Oracle 7 generation has been extended to include the generation of database triggers based upon declarative design information in the Repository. For example, a primary key column may derive its value from an Oracle 7 sequence. In Release 2.0 the system designer will be able to record whether that derivation is to occur at the client or at the server. If at the client, then a Forms ON-INSERT trigger may be created in the appropriate block; if at the server then a PRE-INSERT database trigger will be generated for the appropriate table.

Database triggers may be generated to implement the following functionality:

- Column derivation (e.g. from sequences, auditing, or from other column values)
- Domain validation
- Journalling
- Cascade update on foreign keys
- Arc validation
- Simple denormalisation
- Foreign key ranges

At Release 2.0, all declaratively defined business rules can be marked for implementation at the client, the server or both. The Oracle7 Server Generator and the client generators (e.g. the Forms Generator) will generate code appropriately, and provide user-friendly error messages.

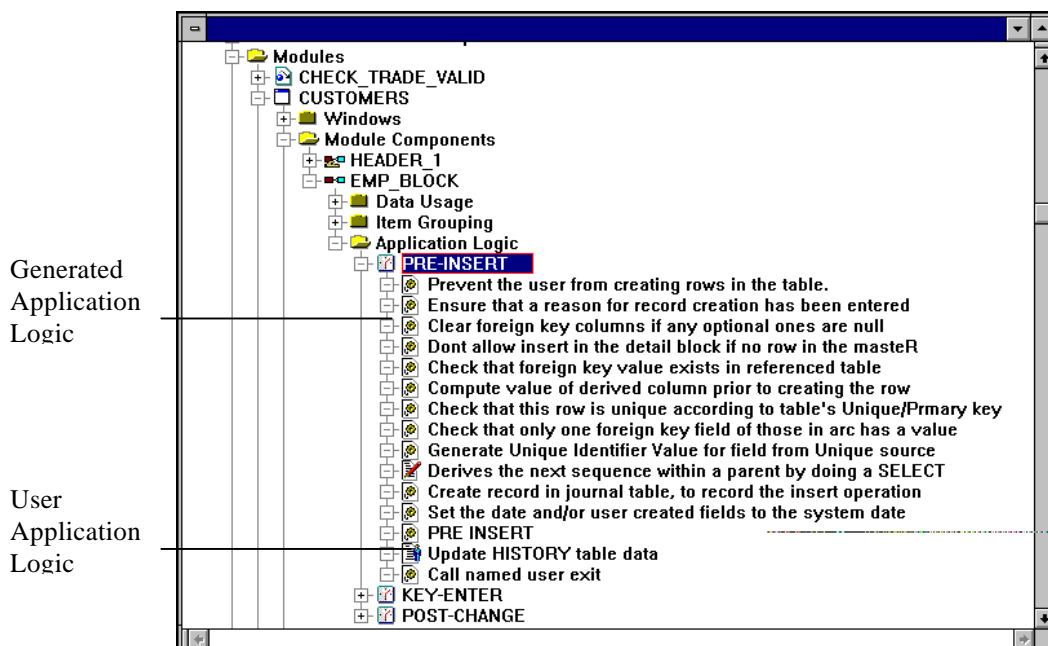
Application partitioning becomes a simple task and functionality can be moved quickly and simply from client to server or vice versa.

## Application Logic

A major new feature of Designer/2000 Release 2.0 is the ability to record application logic or 'custom code' within the repository, and have the generators embed that code within the generated applications. This reduces the need for Regeneration options or post-generation alterations to generated code and greatly improves the manageability and maintainability of systems developed using Designer/2000.

The reverse engineering and design capture capabilities of Designer/2000 have also been extended to provide 100% reverse engineering of application logic as well as 100% generation.

The repository will allow the storage and definition of application logic by holding the event model for each target language. For the Oracle Forms Generator this will mean that logic can be defined at form, block and item level for event points such as WHEN-VALIDATE-ITEM and PRE-INSERT. Each event point may have several pieces of code associated with it, some user defined - application logic, and some generated from declarative definitions - generated logic.



**Figure 1** An example of a PRE-INSERT trigger on the EMP\_BLOCK where the 'Update History table data' user application logic is positioned between the generated application logic.

This simple example demonstrates that designers will be able to see the code that is to be generated from repository definitions and sequence their own application logic within the generated code 'chunks' as required. If required, it will also be possible to replace generated application logic with the users own.

This concept is extended to the server generator for the definition of application logic in database triggers and commit time server-side procedures.

Application logic can be recorded against modules of all types and will be fully supported by the generators for :

- Developer/2000 Forms, Reports and Graphics
- Power Objects
- Web Applications
- Visual Basic

In order to be able to support this new functionality the Module Logic Navigator has been extended to support the syntax of PL/SQL v2, Oracle Basic, Visual Basic and Java.

## **Reuse - Module Components**

Further improvements in productivity can be gained by exploiting the concept of reusable module components in Designer/2000 Release 2.0. The module component concept maps to a block in Forms, a group in Reports, or a Zone in Visual Basic. In Designer/2000 terms it typically consists of a base table and its lookup tables; it provides a logical 'view' to the end user.

Once a module is defined, its components may be shared with other modules. For example, a Forms module to maintain DEPARTMENTS will have a DEPARTMENT block. There may be a requirement to create a master-detail DEPARTMENTS-EMPLOYEES Forms module. In such a case the DEPARTMENT module component maybe shared with the other module.

This greatly reduces the time taken to define modules, and ensures that end-users see consistent views of the same data: the DEPARTMENT block always looks the same. The generators will also exploit the component nature of the definition to generate reusable components in the target language, be it Forms, Visual Basic, Power Objects etc.

Primary and Secondary Column Usages have been remodelled to allow items to be defined as bound to a column or unbound. We are extending the model so that a Module Component can consist wholly of unbound items, thus allowing the generation of control blocks.

## **Co-operative Generators**

A Developer/2000 module may be made up of components of different languages. For example, a Forms module may have an embedded Chart. In Designer/2000 Release 2.0 the generators will co-operate in the generation of this module; generation will be a single, one-step process: the generators will pass control among themselves as appropriate to generate both the Forms and the Chart components, together with the required navigation, data passing or data sharing, and so on.

## **Transaction Navigation**

The design level model has been extended to allow the definition of buttons - 'Action Items' for a given window. Typically the buttons are used to:

- Navigate to another window
- Navigate to another module
- To initiate a process

These can be graphically manipulated on the Module Data Diagram thus providing a mechanism for defining the navigation for a transaction independently of the data model for the modules.

## **Design Editor**

The design and generation capabilities of Designer/2000 Release 2.0 have been combined to provide a fully integrated Design and Generation Environment. The Design Editor provides a single user interface for the application design of both client and server, and a framework in which the co-operative generators share and synchronously maintain design information.

The Design Editor consists of an object navigator window which displays all the design level objects, such as table, columns, modules, reusable module components, etc. The navigator allows all the design objects to be viewed with standard object navigator functionality to drill down to detail information. The action of selecting an object such as a module and dragging it onto the main Design Editor window opens the Module Data Diagrammer for the selected module. This same drag and drop technique can be used to open the Data Diagrammer and the Module Logic Diagrammer if the appropriate design object is selected. The Object Navigator and the other diagrammers will be designed to work together allowing objects to selected in one and dropped in another. For example, a table and some columns maybe selected in the Object Navigator and dropped on an open module diagram in the Module Data Diagrammer and this will automatically create usages of the those tables and columns by that module.

Another key change to Module Data Diagrammer will be the ability to show two views of a module. The first will be the existing module data usage which shows which tables and columns are used by the module and how those tables usages are linked together. The second view gives an indication of what the module will look like when it is generated. This view will display the items, item groups, tab controls, navigation buttons, etc.

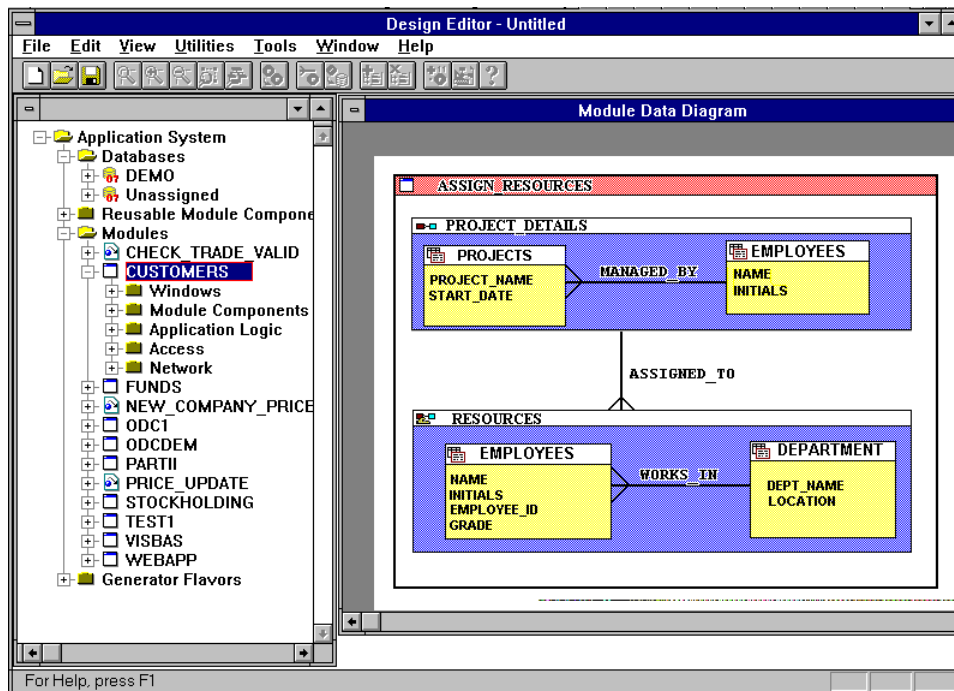


Figure 2 The Module Data Usage View

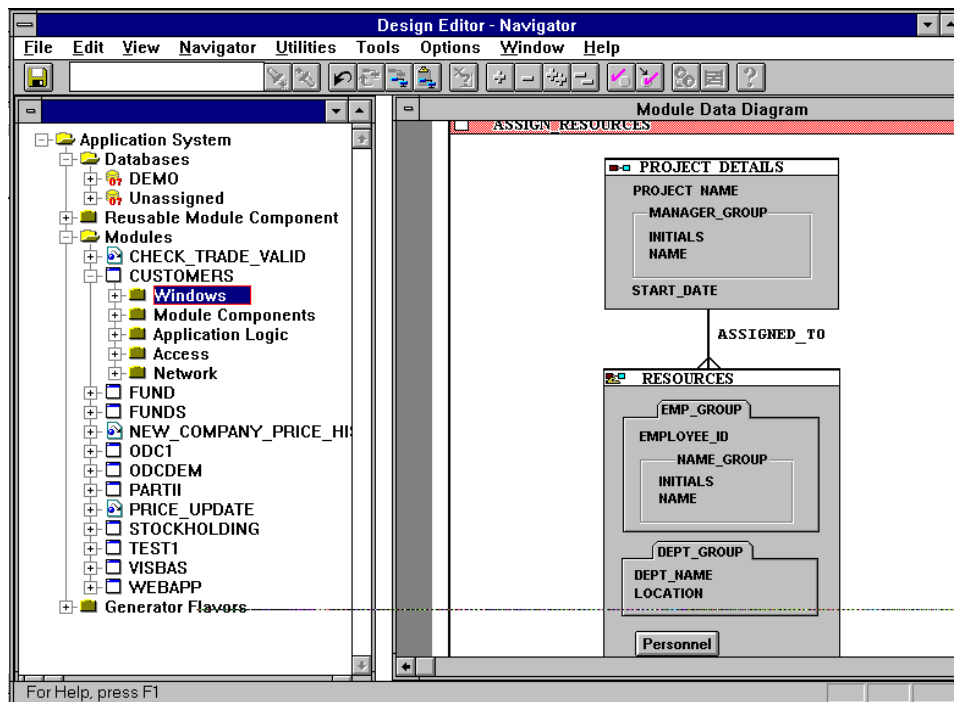


Figure 3 The Module Layout View

The Design Editor offers many improvements in ease-of-use and ease-of learning. Some features are:

- Re-entrant wizards, e.g. for table definition, including columns, keys, indexes etc.
- Guides stepping inexperienced users through complex processes
- Drag and Drop
- Edit in Place, e.g. allowing column names to be changed on the drawing surface without the need to bring up a dialogue box.
- Synchronisation of Updates - if one representation of an element is updated then all representations will be simultaneously updated.
- Revised Documentation and On-line Help
- Right Mouse Menus - providing context sensitive lists of relevant operations when the right mouse button is clicked over an object in the Design Editor.
- The design editor and the generators will run of the same memory structure which allows improved performance and tighter integration and interoperability.

## **Conclusion**

Designer/2000 Release 2.0 achieves 100% Generation by the provision of Application Logic which enables designers to record code specific to their requirements and which cannot otherwise be defined declaratively and generated automatically. Many CASE tools have long had that capability, at least in the arena of COBOL programming.

However, Designer/2000 Release 2.0 goes far beyond that simple definition of 100% Generation: in terms of the number of different targets can be generated, both client and server side, in terms of the ability to define and generate integrated client and server systems, and in terms of providing a single management environment for systems development. Furthermore, all of these capabilities are provided by a toolset which offers extremely high levels of usability and productivity.