

METACONTENT MANAGEMENT WITH ORACLE DESIGNER AND WEBDB

Andrew Faulkner, BC Ministry of Environment, Lands & Parks

INTRODUCTION

This paper presents a Web based approach to accessing application systems metacontent within the Oracle Designer Repository using Oracle WebDB. Also outlined is the information necessary to sufficiently document Application Systems for use during the maintenance phase of the system's life. The requirements for Applications Systems metacontent are incorporated from the various people in the business and technical roles that interact with it.

The repository in Oracle Designer can be populated during the application development life cycle or through the reverse engineering of existing schemas such as those found in the data warehouse. A consistent problem is the lack of use made of the metacontent elements during the maintenance phase.

Metacontent is often referred to simply as metadata. Unfortunately, usage of the term metadata has become ambiguous and means different things in different contexts. In this context, *metacontent* is data that describes both the application's organization and the application's use.

BUSINESS PROBLEM

- It is difficult for the organization to list and report on current Application Systems and their state.
- Several recent surveys and initiatives have captured metacontent about application systems but this resides in various forms (MSWord, Excel, MS Access, paper) and has been collected to different standards.
- There is insufficient metacontent in the repository to allow appropriate decisions as to inclusion or deletion of Application Systems while conducting "house cleaning" as part of the upgrade version 1.3.2 to version 6.0 of Designer.
- There is a requirement for different properties and views of the Application System metacontent. Application Systems information is of interest to people in a number of different roles such as Business Analyst, Application Manager, Systems Management, Risk Management, Operations Planners, and many others who each bring a different perspective.
- Although they meet our published minimum standards, in most instances, Application Systems are currently not very well documented in the CASE delivery provided by our developers into the Oracle Designer repository.

OPPORTUNITY

- Standardize field usage in Designer and update CASE standards as part of the migration to Oracle Designer 6.0 version.
- Provide a living repository for metacontent collected from recent organizational initiatives:
 1. Year 2000 (Y2k) inventory of corporate applications by *Y2K Project Management Office*;
 2. List of ongoing projects maintained by *Applications & Data Management Support*;
 3. Business Continuation Planning (BCP) process by *Systems Support and Operations*;
 4. Freedom of Information/Protection of Privacy inventory by *Information and Privacy Section*;
 5. Oracle CASE upgrade inventory by *Applications & Data Management Support* ;
- Foster ownership by Application Managers for the Application System metacontent.
- Provide access to this metacontent to the Application Managers and other interested parties via a low cost Web interface promoting communication and allowing summarization and statistical reporting.

SCOPE

The Application Systems of concern here are those corporate computer programs that perform the functions of selecting, updating, inserting, and/or deleting structured data that is part of a business process. This discussion is not concerned with software, such as word processing programs that manipulate unstructured data.

APPLICATION SYSTEMS METACONTENT

The information within an Application System is the business content. The content is data like *permit number, amount of nitrate or well location*. The information about the Application System that is running and interacting with the business content is considered the system metacontent. The metacontent is data like *application manager name, system version number, and number of users*.

Application System metacontent is dynamic throughout the system development life cycle (SDLC). At various phases different people require access to and contribute to this information by adding to or refining it. The introduction of SDLC standards has provided a process framework and a document delivery standard. Typically within the organization our grasp of the metacontent diminishes dramatically as the system enters the Maintenance phase.

Reporting on metacontent information across many production Applications can be a frustrating and time consuming exercise. A sample of recent questions include:

- How many applications currently have some sort of Web interface?
- How many 16 bit Oracle Forms applications are there and who are the Application Managers?
- What applications are without maintenance contracts?
- How many applications does company XYZ maintain?
- What is the user population of the current 32bit Applications going on the NT Terminal Server?
- How many VAX Applications are there?
- What Applications have a data warehousing component?
- How do the Applications map onto business priorities?
- What organizational units are responsible for what Applications?
- What is the current CASE version in production? What are the differences between CASE and database changes made since delivery?
- Who are the Custodians of all the Applications in the CASE repository and who are the Application Managers?

It is important to differentiate between the Application as a computer system, used within the organization, and the Application System element in the repository. There may be Applications within the organization that are in use, but have not been developed in house or as an Oracle Application (i.e. legacy VAX application). These will not have a corresponding Application System in the repository. Similarly there may be one or more Application Systems in the repository that are unfinished, and have been abandoned, and therefore do not have a corresponding Application in the organization.

METACONTENT STRUCTURE

A brief survey was conducted to ascertain the current form and structure of Application Systems metacontent in use within the organization and in other government agencies. The survey included the perspectives of Systems Operations, Business Analysis, Data Administration, and Repository Management. Analysis resulted in the 22 attributes listed below in

Table 1.

Attribute	Description
Short Name	The acronym or mnemonic by which this application system is known.
Long Name	The display title of the application system that is to be reported on screens and reports.
Application Description	A brief description of the application system from the business perspective.
Application Custodian	Ultimately refers to the Branch Director, however, sections can be listed to indicate delegated responsibility. Enter an Organizational Unit only (not someone's name).
Application Manager or Client Contact	Who to call for support. Enter a staff person's name as written in the Government Directory.
Application Manger Userid	Use the GEMS userid format where possible.
Application Version	Build-to version number. (Not CASE version number.)
Production Date	Production date of first live usage of this version.
Contractor/Developer	Company Name/Proj Mgr Name/Phone Number/Contract File Number/Start Date/End Date/
Contract Amount	Dollar amount of Development contract or current Maintenance agreement.
# users	Number of users who are trained to access or expected to access the system.
disk requirements	Estimate or calculation from Size Estimate reports of disk usage.
Application Size	Comments on #users and disk requirements. Summaries from FPA, Area Metric, and Database Table and Size Estimate reports.
Total Cost	Estimate or calculation of total development cost to date.
BA	Business Analyst - ISB liaison
Implementation Status	Status as to SDLC phase - (some standard coding to facilitate reporting)
Objectives	The objectives of the application system within the context of the enterprise.
Purpose	A summary of the purpose of the application system within the context of the business unit.
Constraints	The constraints on this application system. Note interfaces to other corporate systems.
Priorities	The priorities of this application system.
Data warehouse	Indicates whether or not this feeds the data warehouse.
Notes	Notes on the application system.

Table 1 Recommended metacontent

REPOSITORY BASED APPROACH TO METACONTENTS

Using the attributes recommended above (

Table 1 Recommended metacontent) a repository based approach is seen as the most effective means of management. The repository is the cornerstone of an Application Systems development project. The goal of the repository is to provide a single point of access and control, eliminating, or significantly reducing support, synchronization, training, maintenance, and learning curves. By incorporating the identified Application System metacontent into the repository many benefits are gained in efficiency and management as well as leveraging the existing metacontent gathering effort of the earlier stages of the SDLC that are also captured there. The standard CASE tool within the organization is Oracle Designer. For the rest of this paper, references to the repository means the Oracle Designer Repository.

PRINCIPLES

The approach taken to the implementation of Application Systems metacontent management must be rooted in the organizational culture. This section lists the relevant beliefs and values of the organization.

1. The organization embraces a standard SDLC with a CASE development approach in order to minimize the risks, enhance the quality, and reduce costs of implementing Application Systems.
2. Maintenance is typically the most expensive and longest phase of the Application System life cycle and tracking and documenting the system status and modifications is equal to that of any other phase in the SDLC where our standards require substantial effort to be expended.
3. Applications Systems do not exist in isolation but are integrated into a corporate organization and systems architecture in which their attributes must be accurately and completely presented to all.
4. A multi-user common database of Application System information is as important to the production application in the Maintenance phase as it was to the developing application in any other phase.

OBJECTIVES

The following objectives should be established for maintaining the metacontent information in the repository:

- Known who is responsible for the Application in the CASE repository.
- Known who is responsible for the Application within the organization.
- Known what the maintenance status is for the Application (e.g., current contract, company, and contact).
- Know the current status of the Application and its data modeled in the repository.
- Know the extent to which the Application impacts the organization.
- Centralize the storage and reporting of Application metrics such as size and cost.
- Maintain a complete profile of Application characteristics as to type, platform, and implementation profile.

ROLES

Identified here are a number of roles and responsibilities that have an interest in the metacontent repository.

APPLICATION MANAGERS

- Provide up-to-date information on their respective applications.

CORPORATE DATA ADMINISTRATOR

- Act as the Data Manager for the repository information.
- Act as the Application Manager for the purposes of Application Systems metacontent access.

CORPORATE DBA

- Act as the repository manager, load all CASE deliveries, safe guard the contents.

SYSTEM DEVELOPERS, BUSINESS ANALYSTS, DATA ADMINISTRATORS & DBAs

- Contribute and use the information from the repository.

ADAPTING THE REPOSITORY

In exploiting the Oracle Repository it must be adapted to satisfy the objectives listed above. In addition standards must be put in place so that the metacontent has common meaning for cross-application reporting. There are several strategies available.

EXTEND THE DESIGNER META-MODEL

It is possible to extend the Designer meta-model to accommodate the desired objectives and metacontent functionality by adding new elements and associations. This would basically be building new functionality into the Designer model. Although this provides the most comprehensive way of achieving the desired repository content it requires detailed analysis and planning and intensifies the level of repository maintenance that must be regularly carried out. Historically meta-model extension in Oracle Designer/2000 has led to some issues with repository management, loss of support, and limitation of upgrade ability.

USE AVAILABLE PROPERTIES

By using available properties of elements that already exist in the meta-model the new metacontent attributes can be stored. The user updatable properties of Application System in Designer 6.0, and their definitions are shown Table 2. This is based on the single database view CI_APPLICATION_SYSTEMS. Note that there 20 definable properties (labeled USER_DEFINED_PROPERTY_0 to USER_DEFINED_PROPERTY_19) available. These user definition properties require a slight intervention in the meta-model to properly register them for use.

APPLICATION_SYSTEM_OWNER	VARCHAR2(30)	The name of the user which owns this application system
AUTHORITY	VARCHAR2(30)	The name of the person or organization authorizing this application system.
CONSTRAINTS	VARCHAR2(240)	The constraints on the application system.
DATAWAREHOUSE_FLAG	VARCHAR2(1)	Indicates whether or not this is a data warehousing application
DISPLAY_TITLE	VARCHAR2(240)	The display title of the application system that is to be reported on screens and reports.
NAME	VARCHAR2(100)	The name of the application system.
PRIORITIES	VARCHAR2(240)	The priorities of the application system.
REMARK	VARCHAR2(2000)	A comment about the application system.
USER_DEFINED_PROPERTY_0	VARCHAR2(240)	user extensibility
USER_DEFINED_PROPERTY_1	VARCHAR2(240)	user extensibility
USER_DEFINED_PROPERTY_2	VARCHAR2(240)	user extensibility
...
USER_DEFINED_PROPERTY_19	VARCHAR2(240)	user extensibility
DESCRIPTION	*multi-line text*	CDIDSC - A brief description of the application system.
NOTES	*multi-line text*	CDINOT - Notes on the application system.
OBJECTIVES	*multi-line text*	CDIOBJ - The objectives of the application system.
SUMMARY	*multi-line text*	CDISUM - A summary of the purpose of the application system.

Table 2 User updatable properties for CI_APPLICATION_SYSEMS in Designer 6.0

OVERLOAD THE MULTI-LINE TEXT FIELDS

A large variety of information can be stored by overloading the multi-line text fields with information in a defined format that is parsable and easily recovered. For instance, where there is a requirement for an individual's Name and Phone Number to be stored, it could all be in one character field as long as data entry contained leading tags as to the nature of the following characters. My entry would look like:

NM:Andrew Faulkner PH:387-8100

PROPOSED REPOSITORY METACONTENT

A combination of using available properties and overloading the multi-line text fields is proposed. This is sufficient to meet immediate objectives and in the future may lead to the development of repository extensions that more comprehensively meet the needs to manage Application Systems. The recommended attributes mapped to existing repository properties are shown in Table 3.

The standard properties of an Application System as recorded in the repository include some that may be confusing. *OWNER* refers to the user who is responsible for the Application's metacontent as loaded into the repository and is updated automatically by the Designer system. The organization's standard is that this be the DBA as master user SYSCASE. *VERSION* refers to the update of the metacontent through version control which again Designer automatically updates. This is very probably different from the version number scheme being used in reference to the actual deployed application.

Other attributes identified such as *Platform*, *Database*, *Computer*, and *Operating System* already exist as part of the Designer Repository metacontent but have not been utilized as part of the organization's CASE standard. A review of this standard may be useful in light of the new features of Designer 6.

IMPLEMENTING METACONTENT MANAGEMENT

The ministry has installed the Oracle product WebDB that facilitates access to data using Netscape or MSIE Web browsers. Using this tool our Application Systems metacontent is made available for viewing directly from the Designer Repository. The following is a step-by-step guide to implementing access to the repository using WebDB.

SYSTEM CONFIGURATION

- Oracle database version 7.3.4 or above
- Oracle WebDB 2.0 or above

Oracle Designer 6.0

- Both WebDB and Designer in same instance

The restriction of maintaining WebDB and Designer in the same instance is removed with WebDB version 2.1 in which the Database Link works correctly between instances.

ACCESSING THE ORACLE DESIGNER SCHEMA FROM WEBDB

The first part of the process is to make the application system metacontent data selectable by WebDB users. We then create a user in WebDB and build the report. After the report is built, it is "published" by making the component public.

Provide select privilege to the metacontent data:

1. Logon to Repository instance as Repository owner (SYSCASE) in SQLPLUS
2. Grant select on CI_APPLICATION_SYSTEMS to public

Create a schema to hold the WebDB objects:

1. Logon to WebDB as the Administrator (usually user WEBDB)

Create a userid "metacontent"

Attribute	Repository Property	Field Type	Description
Short Name	NAME	VARCHAR2(100)	The name of the application system.
Long Name	DISPLAY_TITLE	VARCHAR2(240)	The display title of the application system that is to be reported on screens and reports.
Application Description	DESCRIPTION	*multi-line text* CDIDSC	A brief description of the application system.
Application Custodian	AUTHORITY	VARCHAR2(30)	Ultimately refers to the Branch Director, however, sections can be listed to indicate delegated responsibility. Enter an Organization Unit only (not someone's name).
Application Manager or Client Contact	USER_DEFINED_PROPERTY_	VARCHAR2(60)	Who to call for support. Enter a staff person's name as written in the Government Directory.
Application Manager Userid	USER_DEFINED_PROPERTY_	VARCHAR2(50)	Use the GEMS userid format where possible.
Application Version	USER_DEFINED_PROPERTY_	VARCHAR2(100)	Build-to version number. (Not CASE version number.)
Production Date	USER_DEFINED_PROPERTY_	DATE	Production date of first live usage.
Contractor/ Developer	USER_DEFINED_PROPERTY_	VARCHAR2(240)	Developer: Company Name/Proj Mgr Name/Phone Number/Contract File Number/Start Date/End Date/
Contract Amount	USER_DEFINED_PROPERTY_	NUMBER(10)	Dollar amount of current Dev contract or Maintenance agreement
# users	USER_DEFINED_PROPERTY_	NUMBER(5)	Number of users who are trained to access or expected to access the system.
disk requirements	USER_DEFINED_PROPERTY_	NUMBER(5)	Estimate or calculation of from Size Estimate reports of disk usage.
Application Size	USER_DEFINED_PROPERTY_	VARCHAR2(240)	Comments on #users and disk requirements. Summaries from FPA, Area Metric, and Database Table and Size Estimate reports.
Total Cost	USER_DEFINED_PROPERTY_	NUMBER(10)	Estimate or calculation of total system cost to date.
BA	USER_DEFINED_PROPERTY_	VARCHAR2(30)	Business Analyst - ISB liaison
Implementation Status	USER_DEFINED_PROPERTY_	VARCHAR2(30)	Status as to SDLC phase - (some standard coding to facilitate reporting)
Objectives	OBJECTIVES	*multi-line text* - CDIOBJ	The objectives of the application system.
Purpose	SUMMARY	*multi-line text* - CDISUM	A summary of the purpose of the application system.
Constraints	CONSTRAINTS	VARCHAR2(240)	The constraints on the application system.
Priorities	PRIORITIES	VARCHAR2(240)	The priorities of the application system.
Data warehouse	DATAWAREHOUSE_FLAG	VARCHAR2(1)	Indicates whether or not this feeds the data warehouse.
Notes	NOTES	*multi-line text* - CDINOT	Notes on the application system.

Table 3 Proposed Repository Metacontent

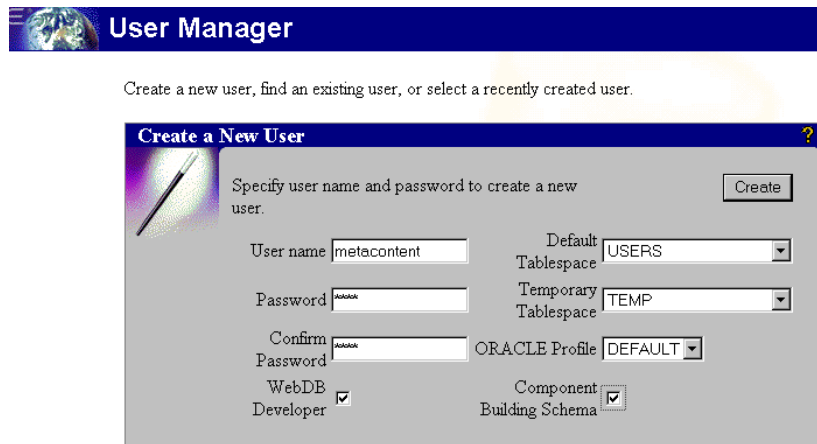


Figure 1: Create a new user

- Grant METACONTENT developer privilege
- Grant Browse Privilege in schemas SYSCASE, METACONTENT

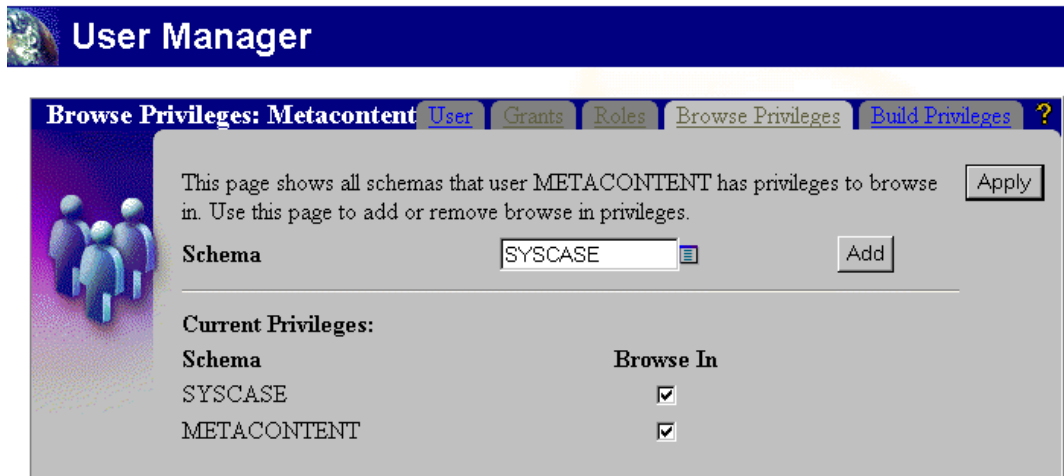


Figure 2: Grant browse privilege on the SYSCASE schema

- Grant Build Privilege in schema METACONTENT

Verify your access and create a WebDB report:

1. Logon to WebDB as METACONTENT
2. You should be able to browse SYSCASE.CI_APPLICATION_SYSTEMS
3. Activate the Report Wizard and create METACONTENT_REPORT_1
4. Choose the view SYSCASE.CI_APPLICATION_SYSTEMS
5. Select the columns of interest (NAME, DISPLAY_TITLE, AUTHORITY, USER_DEFINED_PROPERTY_0
6. In the "Column Conditions" frame set LATEST_VERSION_FLAG = Y
7. (note: there are no quotes around the value)
8. Click finish (the checkered flag) and then Run
9. Go back and under Privileges grant to PUBLIC

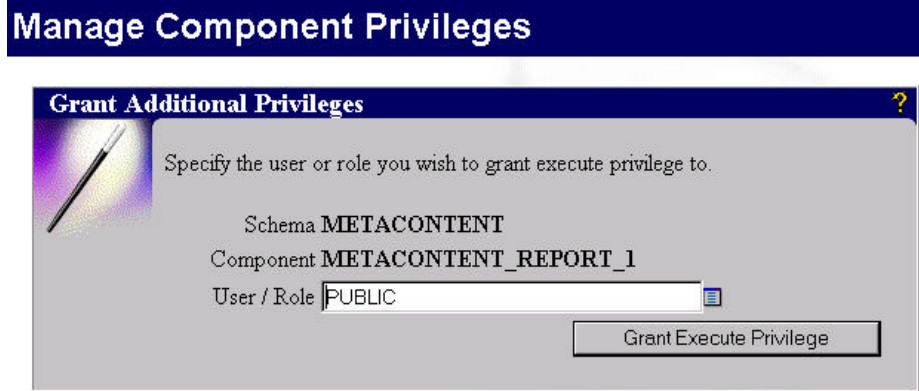


Figure 3: Make the report viewable by all

Create a WebDB menu to generate the report:

1. Activate the Menu Wizard and create METACONTENT_MENU_1
2. Enter a good title “Application Systems Metacontent” and some welcoming text
3. Enter a menu item entry “Most Recent Versions Report” and for the URL “METACONTENT.METACONTENT_REPORT_1.SHOW”
4. Click finish (the checkered flag) and then Run
5. Run the report by select the “Most Recent Versions Report” menu choice link
6. Go back and under Privileges grant to PUBLIC

At this point you now have viewable WebDB report. Any user with an Oracle ID can now run this report. Things to do to improve usability:

- Publish the URL http://yourbiz.com:9000/WebDB/METACONTENT.METACONTENT_MENU_1.show
- Create a corporate template and apply it to the components
- Navigation buttons in footers so you can always get back to the menu
- Add parameter reports

EXTENDING DESIGNER REPOSITORY

The previous report just accessed the repository meta-model as delivered out of the box. To fully capture the required metacontent a number of available property fields need to be modified. Below is an example of one such minor extension to the Designer model.

Modify USER_DEFINED_PROPERTY_0 to be “Application Manager”:

1. Logon to Repository Administration Utility
2. Click “Maintain User Extensions”
3. Explode Element Types/System Defined Elements/APP Application System/Properties
4. Highlight “Usrx0” and click Edit
5. Modify the properties to present the “Application Manager” prompt
6. Click OK and click Publish

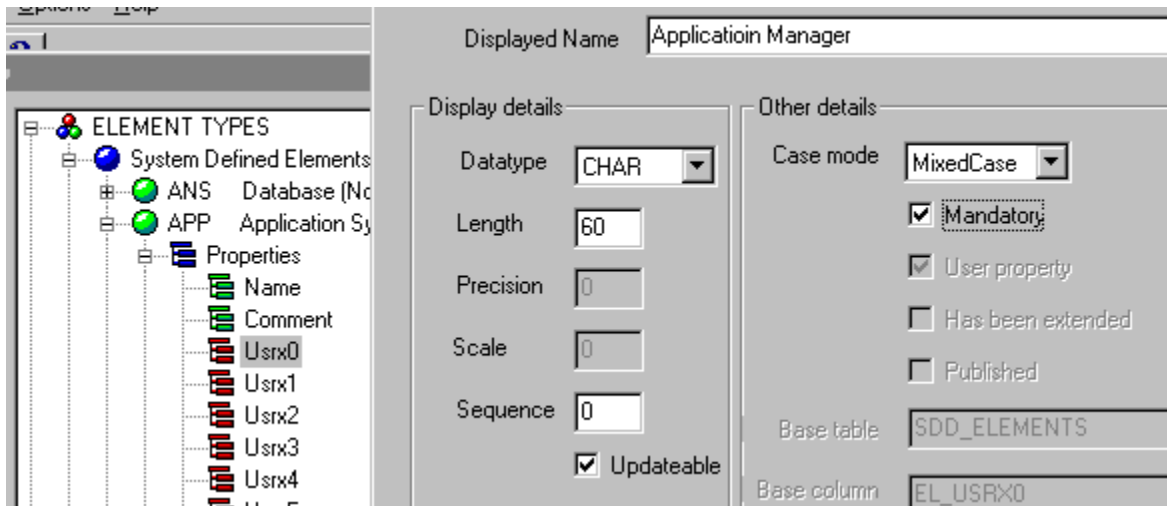


Figure 4: Extend meta-model

Open an Application System in the Designer Object Repository Object Navigator and see new property listed under UE (user extension).

Modify METACONTENT_REPORT_1 to reflect new data:

1. Logon to WebDB
2. Go to Report Building
3. Start edit report wizard by clicking on METACONTENT_REPORT_1 in the section of Recently Edited Reports
4. Click Edit
5. Click Column Formatting tab
6. Edit column heading to be “Application Manager”
7. Click Finish and run the report

Report Results

Name	Display Title	Authority	Application Manager
AGMS	Angling Guide Management System	FISH	(null)
AIR	Air Quality (Auto Polling)	AIR	Ernie Tradewell
ARMS	Automated Records Management	ISB	Drew Smyth
CRS FAMILY	Parent Application for PDS Related	(null)	Corey Bell

Row(s) 1 - 4

Figure 5: Example report

The report builder produces a very readable report but is not very flexible. Very creative reports can be built as Dynamic components by embedding PL/SQL into HTML. A dynamic report (Figure 6) was used to create a Web lookup from the Application Manager’s name into the corporate Web based phone directory.

Report Results

Angelo Facchin	BATH	Bathymetric Mapping	FISH	27-SEP-99
Corey Bell	CRS FAMILY	Parent Application for PDS Related	PPWM	27-SEP-99
Corey Bell	CRISP	Comprehensive Records & Information Syst	PPWM	27-SEP-99



Figure 6: Dynamic report linking Name to Phone Directory

UPDATING REPOSITORY DATA WITH WEBDB

It is quite possible to update the metacontent with WebDB, however it is very important never to directly edit the Oracle Designer tables. Changes to the metacontent must always be done with the repository API. Failure to do so may result in a corrupted repository and loss of data.

By using the “before” and “after” triggers on the WebDB Forms, the repository metacontent for a particular Application System can be copied to a temporary user table, edited, and then an API update script generated. This script can either be executed by the user, if they have the right permissions, or forwarded to the repository administrator to execute.

SUMMARY

Better understanding of the characteristics of Application Systems and attention to tracking them through the production/maintenance phase is highly desirable. There are several opportunities at this juncture related to the upgrade of Oracle Designer to version 6.0 and the comprehensive metacontent available as a result of Year 2000 and other initiatives.

Establishing new standards for metacontent management in Designer and providing Web based access is procedurally and technically feasible.

ACKNOWLEDGMENTS

I wish to extend my appreciation for the discussion and feedback provided by colleagues within the Ministry and by the Data Administrator’s Forum within the BC Government.

ABOUT THE AUTHOR

Andrew Faulkner
 Data Warehouse Technical Architect
 Ministry of Environment Lands & Parks
 P.O. Box 9346 Stn Prov Govt
 VICTORIA BC V8W 9M1
 Canada
 andrew.faulkner@gems5.gov.bc.ca
 +1 (250) 387-8100