

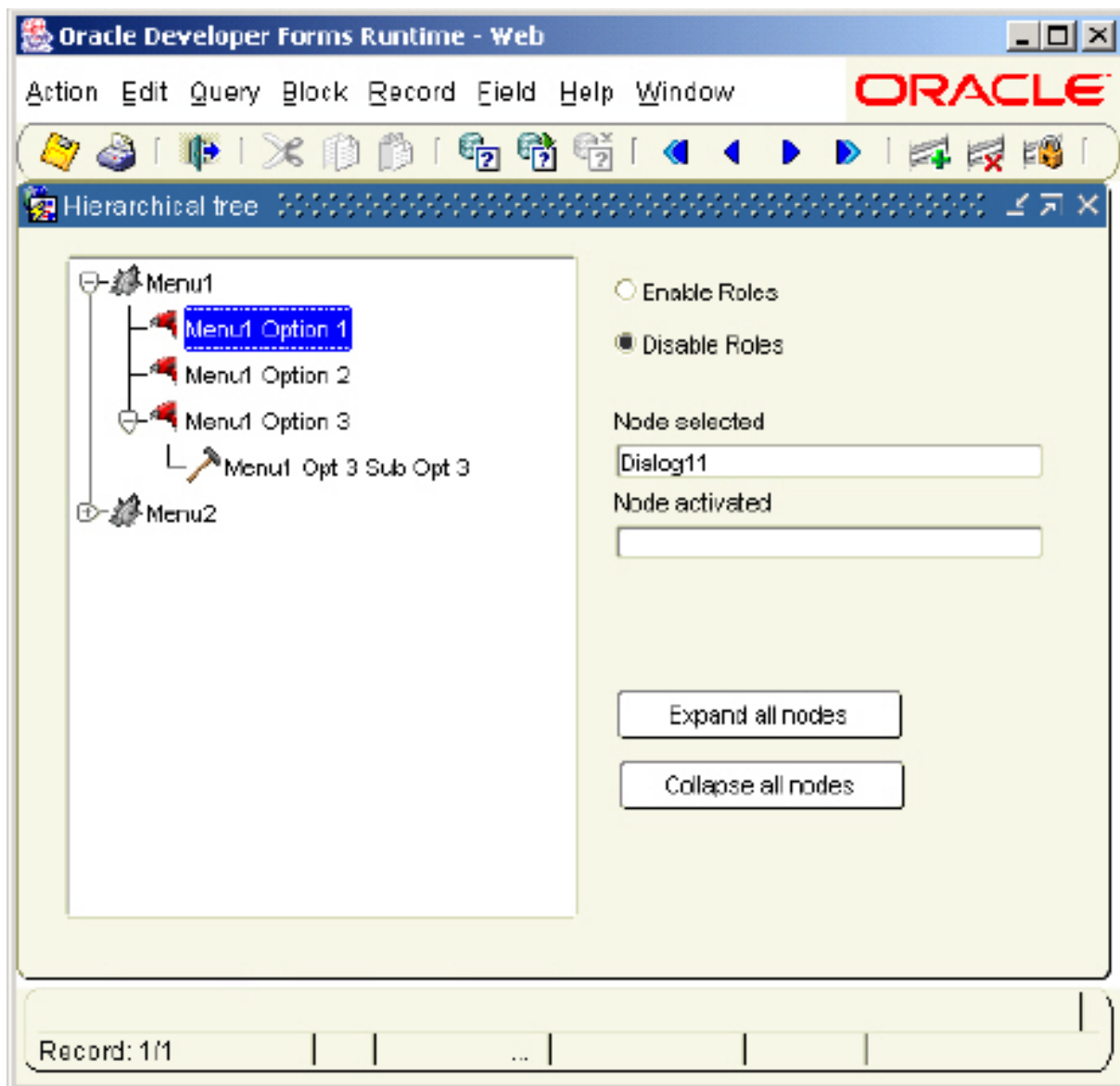
Oracle Forms

Handle hierarchical trees

[Home page](#)

1. Purpose

This is an article to show how to handle a hierarchical tree in an Oracle Forms application.



The goal is to build a tree menu with the possibility of enabling the database roles.

There are 2 ways to populate a hierarchical tree in Forms:

- Using a record group / query
- Building the tree with the `Add_Tree_Node()` built-in

Because we want to handle a database stored menu, this sample use the first option with a record group populated by a database table.
This table contains the corresponding menu options.

2. The database objects

This example needs 2 tables to handle the menu options and their corresponding roles:

1. The table that contains the menu options : **MENU**

```
CREATE TABLE MENU
(
  ID          NUMBER(5),           -- Unique identifiant
  LABEL      VARCHAR2(128 BYTE),  -- Tree label
  ICON       VARCHAR2(40 BYTE),   -- Icon name
  MASTER     NUMBER(5),           -- Parent ID
  STATUS     NUMBER(1)           DEFAULT 1, -- Initial status of the node
  VALUE      VARCHAR2(128 BYTE)   -- Dialog name to call
)
/
```

ICON contains the icon name (without extension) attached to the node.

STATUS can take one of the three possible values:

- 0 normal (Not expandable)
- 1 expanded
- -1 collapsed

2. The table that contains the corresponding roles : **MENU_ROLES**

```
CREATE TABLE MENU_ROLES
(
  ID          NUMBER(5),           -- Menu identifiant
  ROLE       VARCHAR2(30 BYTE)    -- Role name
)
/
```

If you want to enable the database roles in the tree menu, you have to execute the following steps :

- Create the necessary roles in the database

```
CREATE ROLE ROLE_MENU1
CREATE ROLE ROLE_MENU2
...
```

- Grant the roles to the users

```
GRANT ROLE_MENU1 to user1
GRANT ROLE_MENU1 to user2
...
```

This sample needs the **ROLE_MENU1** and **ROLE_MENU2** roles enabled.

Here is the content of the tables after the provided script is executed:

Table : MENU

ID	LABEL	ICON	MASTER	STATUS	VALUE
1	Menu1	mainmenu		1	
2	Menu1 Option 1	optionmenu	1	1	Dialog11
3	Menu1 Option 2	optionmenu	1	1	Dialog12
4	Menu1 Option 3	optionmenu	1	1	
5	Menu1 Opt 3 Sub Opt 3	suboptionmenu	4	1	Dialog131
6	Menu2	mainmenu		-1	
7	Menu2 Option1	optionmenu	6	1	Dialog21

Table MENU_ROLES :

ID	ROLE
1	ROLE_MENU1
2	ROLE_MENU1
3	ROLE_MENU1
4	ROLE_MENU1
5	ROLE_MENU1
6	ROLE_MENU2
7	ROLE_MENU2

3. How to populate the hierarchical tree

The Forms tree needs five column to work correctly:

```
SELECT STATUS, LEVEL, LABEL, ICON, VALUE
FROM MENU
```

Four of them come from the table:

STATUS that indicate the initiate status of the node
LABEL that is the visible label of the node
ICON that contains the (16x16 pixels) icon name of the node (can be NULL)
VALUE that contains the value of the node

LEVEL, wich is a "CONNECT BY" specific pseudo-column specifies the depth at which the individual node appears.

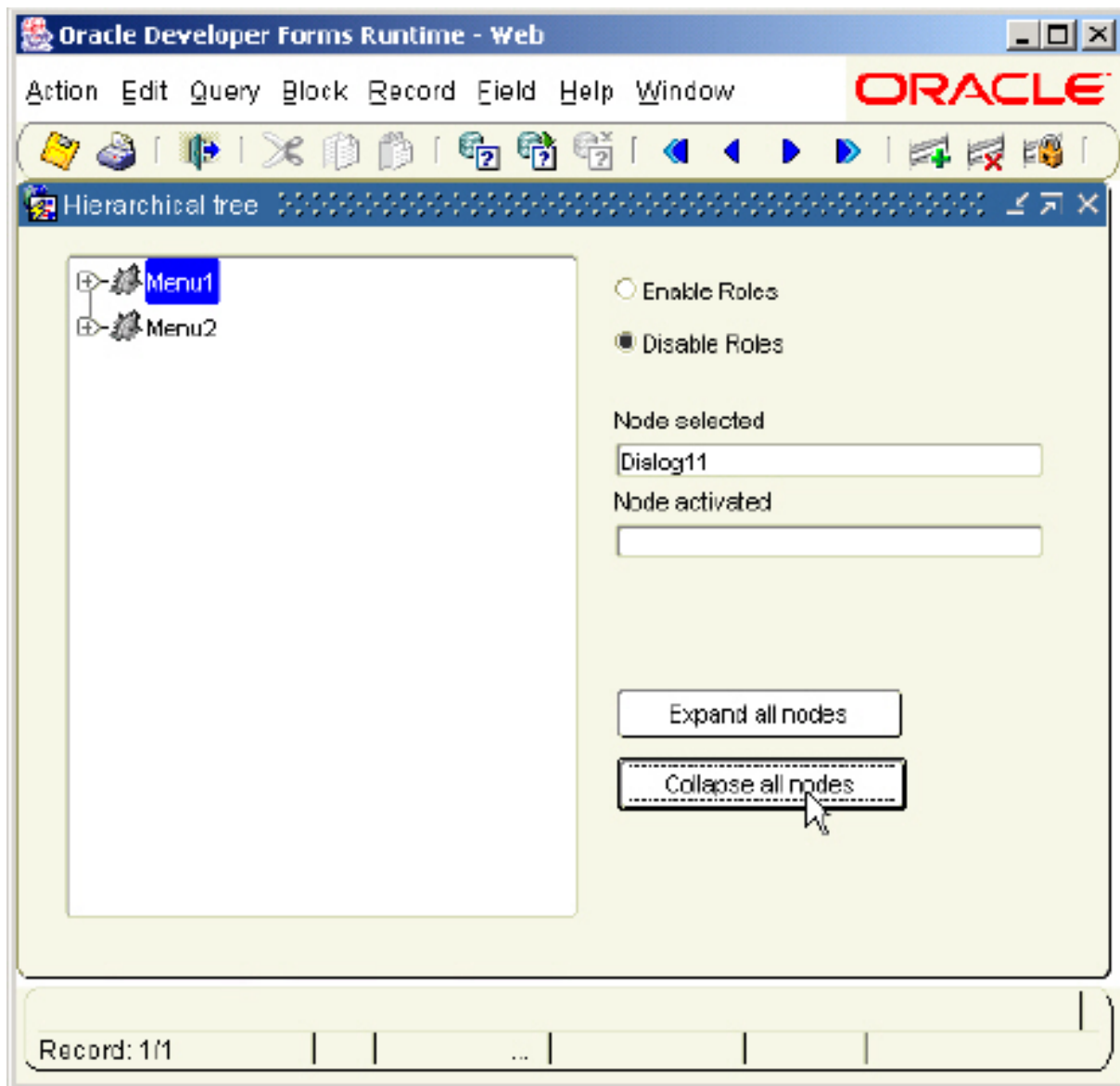
The tree is populated with a record group...

```
HTREE := FIND_ITEM('BL_TREE.MENU');
V_IGNORE := POPULATE_GROUP('RG_TREE');
FTREE.SET_TREE_PROPERTY(HTREE, FTREE.RECORD_GROUP, 'RG_TREE');
```

... , itself populated with a CONNECT BY query.

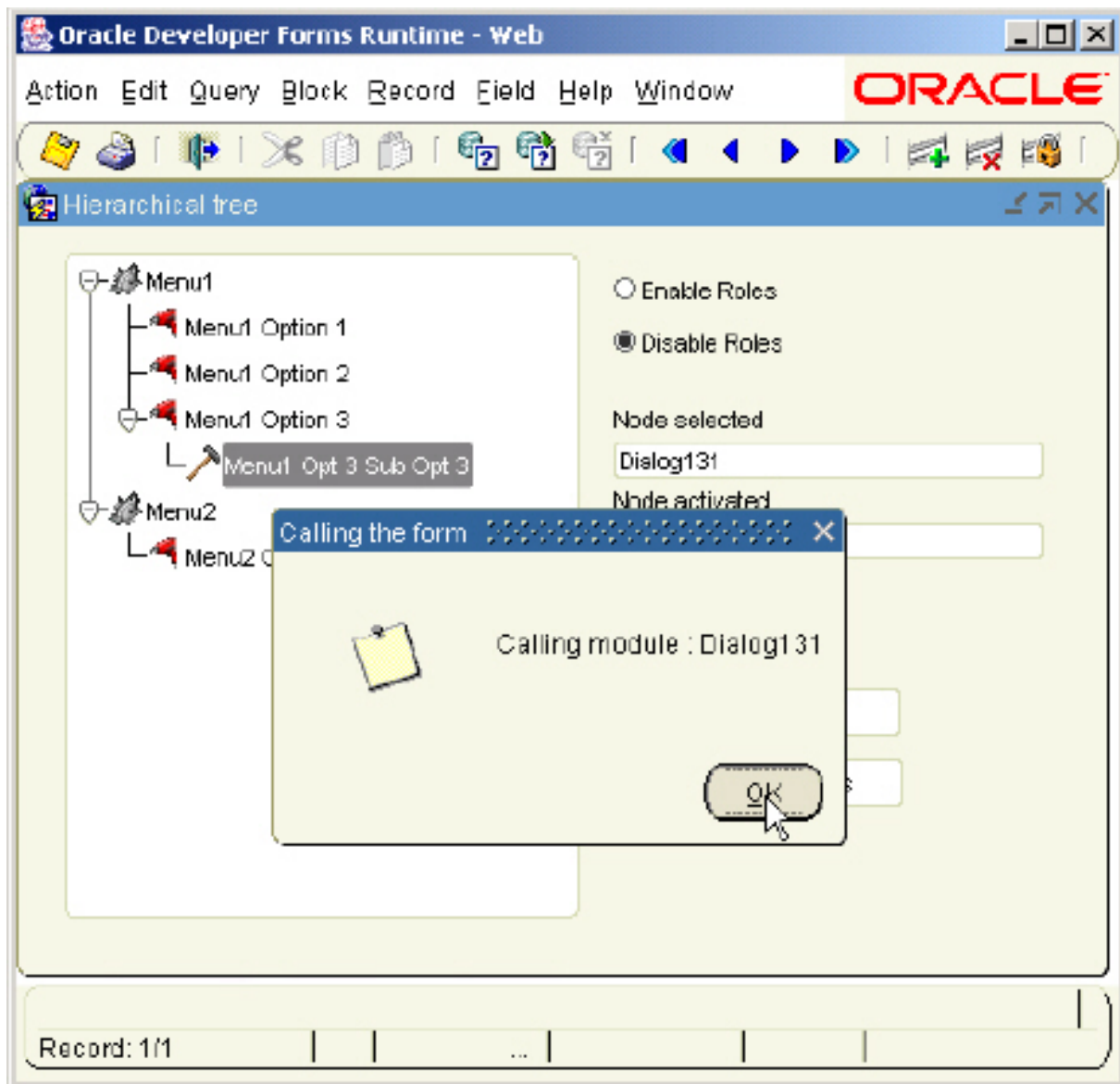
```
SELECT STATUS, LEVEL, LABEL, ICON, VALUE
FROM MENU
CONNECT BY PRIOR ID = MASTER
START WITH MASTER IS NULL
```

Two buttons are added to show how to **expand** or **collapse** all the nodes of the tree:



A single click populate the "Node selected" display item.

A double click populate the "Node activated" display item and call the corresponding screen:



A radio button allows to take into account the database roles.

For this purpose, the tree use another record group (RG_TREE_ROLES) that filters the query.

```

SELECT m.STATUS, LEVEL, m.LABEL, m.ICON, m.VALUE
FROM MENU m
WHERE EXISTS
(
  Select ROLE From MENU_ROLES
  Where ID = m.ID
  And ROLE IN (Select granted_role From user_role_privs)
)
CONNECT BY PRIOR ID = MASTER
START WITH MASTER IS NULL

```

4. The sample dialog

- [Download the tree.zip file](#)
- Unzip the **tree.zip** file

- Run the **tree_install.sql** script under Sql*Plus or Sql Developer.
- Create the 2 roles **ROLE_MENU1** and **ROLE_MENU2**.
- Assign **ROLE_MENU1** to one user and both roles to another user.
- Copy the 4 icons of the **/icons** directory in your icons directory
- Open the **tree.fmb** module (Oracle Forms 10.1.2)
- Compile all and run the module