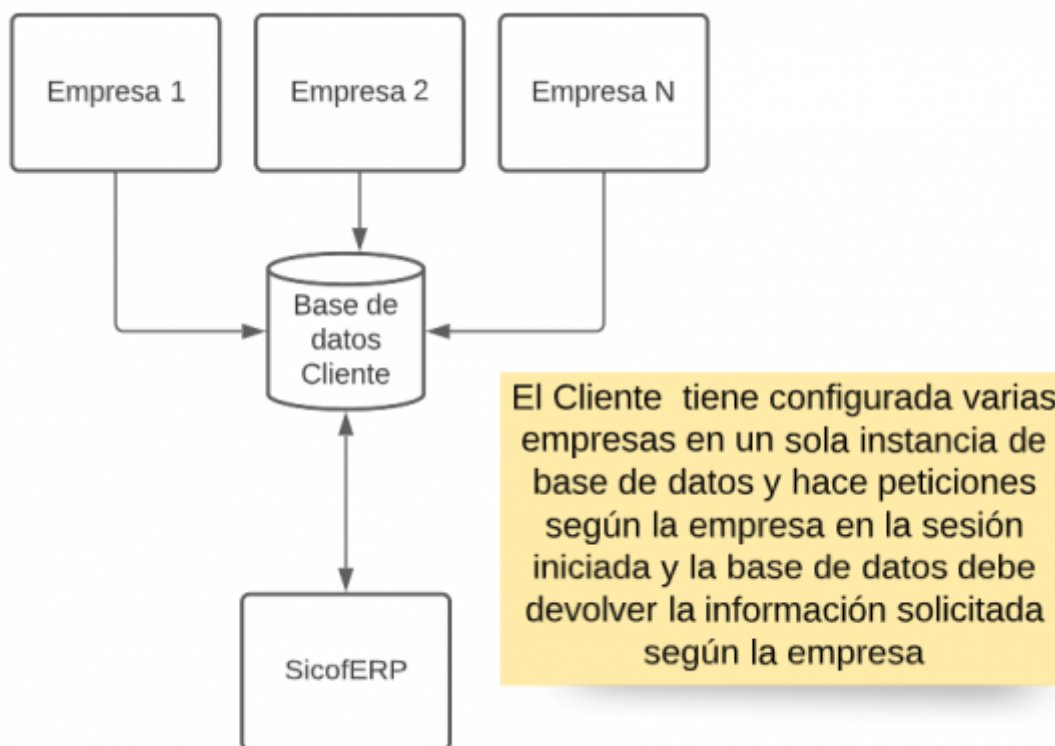


Base de Datos - Multiempresa: Fondos (Fosemed)

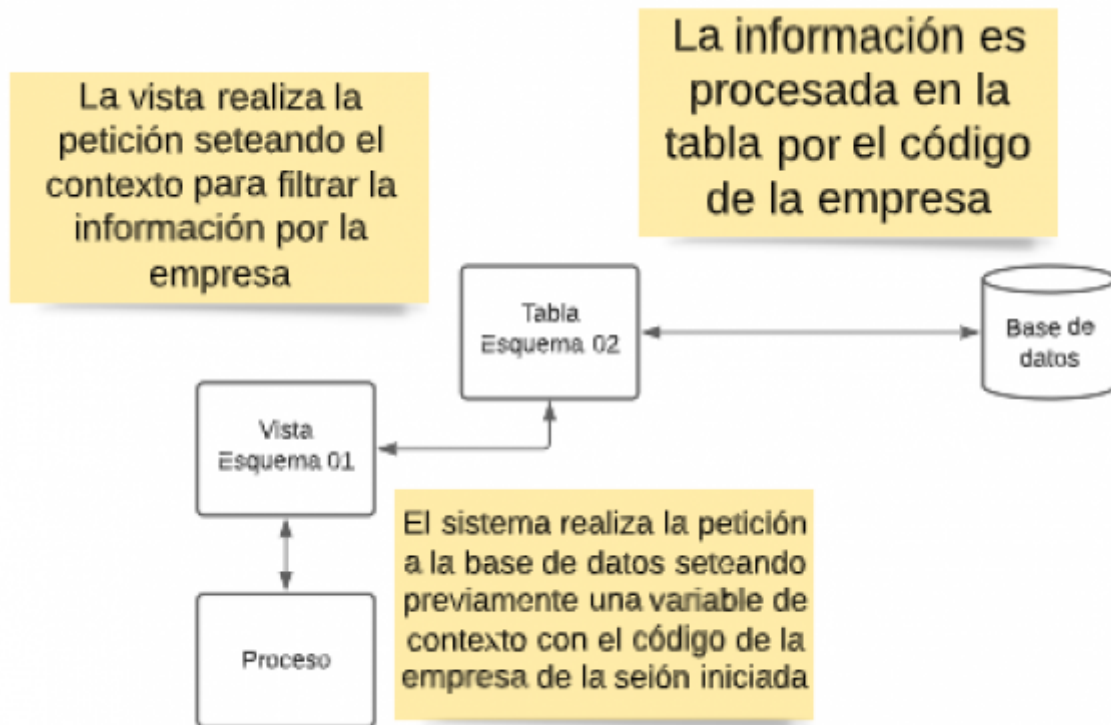
Esta sección esta dedicada a los procesos de actualización de script en los clientes multiempresa. Esta documentación aplica para el cliente Fosemed y aquellos que apliquen configuración multiempresa en la versión Oracle Database versión 11gR2 y posteriores donde el cliente se identifique por contexto en vistas.

Problema



Implementar multiempresa en una sola instancia de base de datos.

Solución



Implementar un modelo de vistas que extraigan información de tablas aplicando un filtrado por contexto.

Implementación Modelo

- Crear contexto.
- Crear paquete donde se utilice el contexto por medio de un procedimiento almacenado.
- Crear Esquema donde se definan las tablas.
- Crear Esquema donde se definan vistas a las tablas con todas las columnas y aplica un filtro con el contexto definido.

Reglas del modelo

- Las tablas van en un esquema diferente al esquema utilizado en los modelos uniempresa. Ejemplo: si el modelo uniempresa para presupuesto es PRESUP01 se debe crear un esquema PRESUP02 donde se almacenarán las tablas.
- Las vistas se crean con la propiedad de actualización en el esquema utilizado en los modelos uniempresa. Ejemplo si modelo uniempresa para presupuesto es PRESUP01 se debe crear en ese esquema las vistas.
- Los trigger se crean en el esquema de las vistas apuntando a las tablas del esquema.
- Los paquetes se crean en el esquema de las vistas apuntando a las tablas del esquema.
- Los procedimientos y funciones se crean en el esquema de las vistas apuntando a las tablas del esquema.
- Las secuencias se crean en el esquema de las vistas apuntando a las tablas del esquema.
- El código PLSQL no se alterará y seguirá la lógica definida en el modelo uniempresa. Esto para evitar versiones de código diferentes.

Modelo de actualización de Scripts

A continuación se define el proceso de actualización según los siguientes tipos:

Actualización de estructuras DDL/PLSQL

Toda estructura DDL menos la modificación/agregación de columnas en tablas se actualizan según las [Reglas del modelo](#)

Actualización de estructuras DML

Toda estructura DML (Insert, Update, Select, Delete) se ejecuta igual que los modelos uniempresa, sin embargo se debe ejecutar previamente el contexto según el siguiente script y empresa a afectar.

```
EXECUTE sicof.pkg_ctx_sicof.sp_nombre_variable('CODIGO_MEMPRESA', '[Código de la Empresa]');
```

Actualización de estructuras DDL (Columnas en Tablas)

Las actualización de tablas requiere que se ejecuten los siguientes pasos:

- Ejecutar el alter en la tabla.
- Ejecutar un script que genera un script de la vista que representa la tabla actualizada.
- Ejecutar el script que regenera la vista.

Ejemplo

Se comparte el siguiente ejemplo de una actualización real del módulo de presupuesto.

```
--
=====
==
-- Fecha: 3:27 p. m. domingo, 24 de julio de 2022 - carlos.torres@ada.co
-- Paso 1: Crear las columnas en la tabla
--
=====
==
ALTER TABLE PRESUP02.FE_TIPO_FACTURA ADD (FECHA_FINAL_RESOLUCION DATE);
ALTER TABLE PRESUP02.FE_TIPO_FACTURA ADD (CLAVE_TECNICA VARCHAR2(256));
--
=====
==
-- Fecha: 3:27 p. m. domingo, 24 de julio de 2022 - carlos.torres@ada.co
-- Paso 2: Ejecutar el script que genera un script de la vista actualizada
```

de la tabla

```
--
=====

==
SELECT  'CREATE OR REPLACE FORCE VIEW ' || REPLACE(X.OWNER, '02', '01') ||
        '.' || X.TABLE_NAME || ' AS SELECT * ' ||
        'FROM ' || X.OWNER || '.' || X.TABLE_NAME || ' T ' ||
        'WHERE t.codigo_empresa IN (DECODE (SYS_CONTEXT (''CTX_SICOF'',
        ''CODIGO_EMPRESA''), ''999999999'', ''-1'', SYS_CONTEXT (''CTX_SICOF'',
        ''CODIGO_EMPRESA''), ''999999999'') OR SYS_CONTEXT (''CTX_SICOF'',
        ''CODIGO_EMPRESA'') = ''0'');' AS NEW_SYNTAX_SQL
FROM    SYS.ALL_ALL_TABLES X
WHERE   X.OWNER = 'PRESUP02'--aqui va el esquema de la tabla
AND     X.TABLE_NAME = 'FE_TIPO_FACTURA ';--Aqui va el nombre de la tabla

--
=====

==
-- Fecha: 3:28 p. m. domingo, 24 de julio de 2022 - carlos.torres@ada.co
-- Paso 3: Ejecutar el script de la vista actualizada contenido en la
columna NEW_SYNTAX_SQL
-- del paso 2
--
=====

==
CREATE OR REPLACE FORCE VIEW PRESUP01.FE_TIPO_FACTURA
AS
    SELECT *
    FROM PRESUP02.FE_TIPO_FACTURA T
    WHERE t.codigo_empresa IN
        (DECODE (SYS_CONTEXT (''CTX_SICOF'', ''CODIGO_EMPRESA''),
        ''999999999'', ''-1'',
        SYS_CONTEXT (''CTX_SICOF'', ''CODIGO_EMPRESA''),
        ''999999999'')
        OR SYS_CONTEXT (''CTX_SICOF'', ''CODIGO_EMPRESA'') = ''0');
```

Notas del Proceso

- La [Actualización de estructuras DDL \(Columnas en Tablas\)](#) sólo aplica para tablas que tienen el campo código_empresa y tienen representación en tabla/vista.
- Toda sintaxis DML debe ejecutar previamente el seteo contexto.
- Si el seteo de contexto se ejecuta con el código_empresa = 0 se aplicará la para todas las empresas, esto puede generar errores ya que los cambios aplicarán para todas la empresas generando configuraciones o información duplicada.

[←Volver atras](#)

From:

<http://wiki.adacsc.co/> - **Wiki**

Permanent link:

<http://wiki.adacsc.co/doku.php?id=ada:howto:sicoferp:database:oracle:multiempresafondosfosemed> 

Last update: **2022/07/24 20:42**