

How to create SQL tables for an integrating Dexterity program

CONFIDENTIAL ARTICLE

(The information in this article is provided to you in accordance with your Confidentiality Agreement)

TechKnowledge Content

SUMMARY

This article discusses how to create SQL tables for Microsoft Business Solutions - Great Plains Dexterity based programs.

MORE INFORMATION

Use the following steps to create SQL tables that reside in an integrating third party program written in Dexterity. These steps also cover how to grant SQL permission to those tables.

Note This method replaces the amAutoGrant method that is described in Chapter 41 of the Dexterity Programmers Guide, Volume 1.

1. Create a global procedure named **Startup** if one does not already exist in your third party program. This script runs when you start Great Plains and is typically where triggers are registered.
2. In the **Startup** script, create a procedure trigger on the Add_Successful_Login_Record procedure by using the following code.

```
{Name: Startup}
local integer l_result;
l_result = Trigger_RegisterProcedure(script Add_Successful_Login_Record, TRIGGER_AFTER_ORIGINAL, script gp_create_tables);
if l_result <> SY_NOERR then
warning "The Add_Successful_Login_Record trigger is not registered.";
end if;
```

3. The registered trigger created in step 2 will call the global procedure **gp_create_tables** whenever the user logs in to Microsoft Dynamics GP. This procedure is run when a Great Plains company is initially opened or when the user or company is switched.

4. Create the global procedure named **gp_create_tables**. This script will create the tables for the third party program in the correct SQL Server database. It will also create the table generated stored procedures (zDP procs) and grant SQL permissions to the table and procedures. Use the following code.

```
{Name: gp_create_tables}
local boolean result,l_result,OUT_Access;
{if logged in as sa, let them create the tables}
if 'SQLSaUser' of globals then
    OUT_Access = true;
else
    {This else statement will work only on 8.0. If logged in as a user other than sa,
    but they have table access permissions, let them create the tables}
```

Partner Only Article

Article ID : 874554
Last Review : N/A
Revision : 3.0

```
        if syUserInRole('User ID' of globals, ROLE_SYSADMIN) or (syUserIsDBO ('User ID' of globals, 'Intercompany ID' of globals)
and syUserIsDBO ('User ID' of globals, SQL_SYSTEM_DBNAME)) then
            OUT_Access = true;
        end if;
end if;

if 'SQL Server' of globals > 0 and OUT_Access then
    {enable table creation mode}
    result = Table_SetCreateMode(true);
    {Do not display any table errors to the user.}
    result = Table_DisableErrorChecks(true);
    {accessing the table creates it, list all your tables here, make sure to close the tables when done}
    get first table GPSetup; {Purchasing series table}
    close table GPSetup;
    get first table GPSetup2; {System series table}
    close table GPSetup2; {now set permissions, call once for the table and once for the stored procs}
    {GPSetup is a purchasing series table so that will be in the company dbo}
    l_result = GrantAccess(physicalname(table GPSetup),false,"DYNGRP",'Intercompany ID' of globals)
        of form 'SQL Maintenance';
    l_result = GrantAccess(physicalname(table GPSetup),true,"DYNGRP",'Intercompany ID' of globals)
        of form 'SQL Maintenance';
    {GPSetup2 is a system series table so that will be in the DYNAMICS database}
    l_result = GrantAccess(physicalname(table GPSetup2),false,"DYNGRP","DYNAMICS") of form 'SQL Maintenance';
    l_result = GrantAccess(physicalname(table GPSetup2),true,"DYNGRP","DYNAMICS") of form 'SQL Maintenance';
    {Turn off automatic table creation.}
    result = Table_SetCreateMode(false);
    {Turn table error reporting back on.}
    result = Table_DisableErrorChecks(false);
end if;
```

5. The tables and table generated stored procedures will be created in SQL Server and SQL permissions will be granted. This script will only run for the SA or DYNSA users and after the tables are created it is okay to run this script again.

This article was TechKnowledge Document ID: 33429

APPLIES TO

- Microsoft Business Solutions–Great Plains Dexterity, when used with:
Microsoft Business Solutions–Great Plains 8.0

Keywords: kbmbspartner kbmbsmigrate KB874554