

How to add Named Printers support to a report by using triggers in Microsoft Dynamics GP

CONFIDENTIAL ARTICLE

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

INTRODUCTION

This article describes how to add Named Printers support to a report by using triggers in Microsoft Dynamics GP.

The Named Printers feature in Microsoft Dynamics GP lets you route print output to a particular printer. Starting with Great Plains 6.0 Dynamics and Great Plains 6.0 eEnterprise, the Named Printers feature has been part of the Microsoft Dynamics GP core dictionary (Dynamics.dic). Most of the common reports and documents in Microsoft Dynamics GP include code to support Named Printers functionality. However, some reports in the Microsoft Dynamics GP core dictionary or in third-party products do not contain Named Printers support.

Note The easiest way to add Named Printers support is to have the product developer add the appropriate code to the report. For information about how to do this, see the "Named Printers Interface" document in the Microsoft Dynamics GP Software Development Kit (SDK).

This article contains a method that lets you add Named Printers support to a report in any dictionary by using triggers. Because you cannot add the printer clause to an existing **run report** command, you must change the application's default printer before the report runs. Then, restore the default printer after the report is finished running.

Note For this method to work, you must have a system default printer defined. You may also have an optional company default printer defined. These default printers are used to restore the application default printer after the report is finished printing.

MORE INFORMATION

To add Named Printer support to a report by using triggers, follow these steps:

1. Add the printer task to the Assign Named Printers window. To do this, call the following **ST_Set_Printer_Task** function from the initialization script of the product that you are running.


```
ST_Set_Printer_Task(IN_Printer_Series, IN_Printer_Task, IN_Printer_Description);
```
2. Add the following two helper procedures to your code so that the code can call the Named Printers functionality to change the default printer.

Procedure 1: ST_Printer_Interface_Change

```
{ Global Procedure: ST_Printer_Interface_Change }

in integer IN_Printer_Series;
in string IN_Printer_Task;

local 'Printer Settings' PrinterSettings;

PrinterSettings = ST_Set_To_Default_Printer(IN_Printer_Series, IN_Printer_Task);
if not empty(PrinterSettings) then
    Printer_SetDestination(PrinterSettings);
end if;
```

Procedure 2: ST_Printer_Interface_Default

```
{ Global Procedure: ST_Printer_Interface_Default }

local 'Printer Settings' PrinterSettings;

PrinterSettings = ST_Set_To_Default_Printer(7,ST_DEFAULT); {System}
if not empty(PrinterSettings) then
    Printer_SetDestination(PrinterSettings);
end if;
PrinterSettings = ST_Set_To_Default_Printer(8,ST_DEFAULT); {Company}
if not empty(PrinterSettings) then
```

Partner Only Article

Article ID : 921636
Last Review :
Revision : 1.1

```
Printer_SetDestination(PrinterSettings);  
end if;
```

3. Determine whether the script that runs the report runs in the foreground or in the background.

Note The triggers that you add must run before the script that contains the **run report** command runs. Also, these triggers must run after the script that contains the **run report** command runs.

To determine whether the script that runs the report runs in the foreground or in the background, suspend background processing, and then print the report. To suspend background processing by using Process Monitor, click **Process Monitor** on the **File** menu, and then click **Suspend**.

After you suspend the background processing, print the report. When you print the report, the report appears in the background queue as expected. However, if a procedure also appears in the background queue, you can verify that the processing occurs in the background.

4. Identify where to put the triggers. Use the information from step 3 together with the Dexterity Script Log files to identify the last script that ran before the report runs. This is the script on which you must trigger.

Note You may have to perform tests to identify the correct script.

5. Add triggers to call the helper procedures. Before the script from step 4 runs, add a trigger, and then call the **ST_Printer_Interface_Change** procedure. Then, add a trigger after the same script to call the **ST_Printer_Interface_Default** procedure. If you determined that the script runs in the background, you must use a background call instead of a typical to call both helper procedures.

APPLIES TO

- Dexterity, when used with:
 - Microsoft Dynamics GP 9.0
 - Microsoft Business Solutions–Great Plains 8.0
 - Microsoft Business Solutions–Great Plains 7.5

Keywords: kbexpertiseadvanced kbmbspartner kbhowto kbinfo kbmbsmigrate KB921636