Infor Enterprise Server Plug-in for Microsoft SQL Server Reporting Services Administration Guide

Release 10.7.x

Contents

About this guide

To use the SQL Server Reporting Services for Infor LN Reports, you require the Infor ES plug-in for Microsoft SQL Server Reporting Services. This plug-in enables the Report Server to process and render reports that are sent from LN.

Intended audience

This document is intended for administrators and explains the installation and configuration of the Infor ES Reporting plug-in, at the LN node.

The Infor ES plug-in for Microsoft SQL Server Reporting Services is referred to as Infor ES Reporting plug-in.

Related documents

You can find the documents in the product documentation section of the Infor Support Portal, as described in "Contacting Infor".

• *Infor Enterprise Server Administration Guide* (U8854 US)
• *Infor LN Specific Installation Guide - Updates* (U9497 US)
• *Infor Enterprise Server Plug-in for Microsoft SQL Server Reporting Services - Development Guide* (U9657 US)

The LN reports that can be rendered by the Report Server must be designed according to several rules and restrictions that are described in the *Infor Enterprise Server Plug-in for Microsoft SQL Server Reporting Services - Development Guide* (U9657 US).

Contacting Infor

If you have questions about Infor products, go to Infor Concierge at https://concierge.infor.com/ and create a support incident.

The latest documentation is available from docs.infor.com or from the Infor Support Portal. To access documentation on the Infor Support Portal, select **Search > Browse Documentation**. We recommend that you check this portal periodically for updated documentation.

If you have comments about Infor documentation, contact documentation@infor.com.

Chapter 1: Introduction

Microsoft SQL Server 2012 Reporting Services provides a complete, server-based platform. This platform is designed to support a wide variety of reporting needs, enabling organizations to deliver information across their entire enterprise. Infor also supports SQL Server 2016.

The integration is built to use the power of SSRS in an LN environment. The LN session reports can be redesigned as an SSRS report design, which provides a modern layout with all types of features such as images, indicators, and graphs. For an administrator, the following tasks apply:

- Installation

  Install the Infor ES Reporting plug-in in the SSRS environment. This plug-in makes the output of the LN sessions available as data sources for the SSRS report design. The installation is described in Installation on page 7.

- Configuration

  Define the SSRS server(s) in the LN environment. You can send the output of the session reports to an SSRS server through an LN device. For details, see Configuration on the LN Server on page 9.

- Distribution

  Distribute the SSRS designs from the LN dictionary to the SSRS server(s). For details, see Distribute MS Report Designs on page 16.

Chapter 2: Installation

This chapter describes how to install the Infor ES Reporting plug-in, which has been added to SSRS to handle LN reports.

Prerequisites

Before you can start the installation, ensure that you have access to either a Microsoft SQL Server 2012 or a Microsoft SQL Server 2016 environment. SQL Server is usually installed on a central machine.

The Report Server supports Native or SharePoint Integrated mode with Windows authentication. In future releases, claims-based authentication, for example, AD FS, will also be supported.

If you want to use direct printing in SSRS you must connect a printer which supports direct printing of PDF files. This means that the printer must be able to handle a byte stream with PDF data which will be send from LN.

Client tools for SQL Server 2016

If you use Microsoft SQL Server 2016, you must install these client tools:

1   Visual Studio 2015
2   SQL Server Data Tools (SSDT) 2015
3   Visual Studio extensibility tools update 3
4   Infor ES plug-in for Microsoft SQL Server Reporting Services

    The Infor ES plug-in for Microsoft SQL Server Reporting Services has been validated until Windows Server 2016.

You must install the tools in the indicated order.

Note: Follow the sizing guidelines for using Infor LN with the Microsoft Reporting Solution. See the In for LN Sizing Guidelines Microsoft Reporting Solution.

The installer

The installer is used for Microsoft SQL Server 2012 and for Microsoft SQL Server 2016.

The software contains:

- LN Data Extension

  This is an extension made for SSRS that enables the integration with LN.

- LN Report Configurator

  With this tool you can automatically design Microsoft SSRS reports for LN.

Install the Data Extension only on your runtime environment, that is, the Report Server. You are not required to install the development environment.

During the installation process, several dialog boxes are displayed. The dialog boxes are self-explanatory, but ensure that you select the **Custom install** in the **Setup Type** dialog box.

Click **Next** to display the Custom Setup screen.

Select the Runtime Environment. The Data Extension is installed on your runtime environment, that is, the Report Server.

If you select **Development Environment**, the Data Extension and the Report Configurator are installed on your development environment.

To start the installer:

1   Download the Infor ES Reporting plug-in installer.
2   Unzip the file and double-click the setup.exe file in the Start folder.
3   The Infor Installation wizard will start and the Welcome screen is displayed. Click **Next**.

After the installation has finished successfully, you can proceed with

Chapter 3: Configuration on the LN Server

To send the output of the session reports to an SSRS server through an LN device, you must complete the configuration steps that are provided in this chapter.

Define the Report Server

The **Report Servers (ttrpi1500m000)** session must contain information to be able to communicate with the Report Server.

Provide this information:

1   Use the classic tools or Web UI on your LN system. To start the **Report Servers (ttrpi1500m000)** session.
2   Specify this information:
    • The name and description of the Report Server.
    • The Service Account to use for the webservice calls. This account is a Windows authentication account that has been set up at the Report Server and configured with privileges that allow webservice calls. The specified user must have access to the Report Server with the "Content Manager" role. All other users who want to view reports must have the "Browser" role. See the Microsoft documentation for more information on those roles.
    • The Report Folder, which is the folder in which the reports must be stored. The following default string is supplied:

    ```
    "%2fInfor%20Reporting%2fLN%2fSessionReports%2f$hostname%2f$pacc"
    ```

    With this string, the Report Folder will be automatically filled at runtime with the hostname and package combination of the user. The folder must start with "%2f" and can contain the macros $hostname and $pacc. You can also specify your own folder.

    • The Web Services URL, which is the URL that is used to access the Report Server.
    • The Report Viewer URL, which is the URL that is used to access the report on the Report Server. A default string is supplied. $project and $report will be substituted at runtime as the Package Combination and the code of the 4GL report. By default, the $folder directory is generated. You must replace this with $project. The Report Folder can also contain $folder. At runtime, $folder is substituted with the path that is defined in the Report Folder.

Create a device

To print the output of the session reports, you must create an LN device. With this device you can:

• Render reports by using the SSRS.
• Print preview XML files.
• View log files after you have imported or exported MS reports.

You must create two devices: one for normal report rendering and one for creating XML files that are used to design reports in SSDT.

For each device that you create to render a report on the Report Server, you must define which server it will use. You can define only one server per device, although multiple devices can use the same server.

Create a device for normal report rendering

Use the classic tools or Web UI on your LN system.

To create a device:

1 Start the **Device Data (ttaad3500m000)** session.
2 Add a device by entering a device name, for example, MS_SSRS1.
3 In the **Device Type** field, select **Microsoft Reporting Services**.
4 To start the Report Viewer that renders the report, specify this command:

```
-server <your server as defined in ttrpi1500m000>
```

To specify other arguments in the command, see the online help.

Create a device for creating XML files

Before you can create reports, you must add a device to be able to create a preview XML file. Start the classic tools or Web UI on your LN system.

To create a device:

1 Start the **Device Data (ttaad3500m000)** session.
2 Add a device by entering a device name, for example, IER_DESIGN.
3 In the **Device Type** field, select **Microsoft Reporting Services**.
4 In the **Argument** field, specify the command -DESIGNER.

The output files named ${BSE_TMP}\ier_[your name]_[report name].xml are created. These XML files are used for the report metadata and runtime data. The files differ from the normal XML files that are used for rendering reports at runtime.

For more information about device management, see the *Infor Enterprise Server Administration Guide* (U8854 US).

Direct printing

To send a report directly to the printer without viewing it on screen, you can use direct printing. You can use direct printing for:

- Batch printing
- During print jobs

For direct printing you must create a Microsoft device that has additional arguments.

To create this device:

1   Use the classic tools or Web UI on your LN system. To start the **Device Data (ttaad3500m000)** session.
2   Add a device by entering a device name, for example, `MR_DP`.
3   In the **Device Type** field, select **Microsoft Reporting Services**.
4   Specify a different argument, for example:

```
-server jd01 -printer \\srv-nld-fp0\nlbap024
```

For more options, see this table:

Device argument	Result	Conditions
-printer	Prints the report to the default printer of the user.	Only possible when there is a UI; not possible when in job mode.
-printer <printer>	Prints the report to the specified printer	Not possible on UNIX in job mode.
-device <windows printer device>	Prints the report to the specified Windows Printer device. This argument uses the Device Queue of the Windows Printer device.	Not possible on UNIX in job mode.
-device <windows -server printer device>	Prints the report to the specified Windows Server Printer Device. This argument uses the Device Queue of the Windows Server Printer device. In the argument of the Windows Server Printer device, the server and port number to connect to the Infor ES Reporting Service must be specified.	

Device argument	Result	Conditions
-server <your server as defined in ttrpi1500m000>	Prints the report to a printer on the specified server.	

For details, see the online help of the **Device Data (ttaad3100s000)** session.

View reports in specific formats

By default, the reports are shown in the Report Viewer in HTML format. To have the reports rendered directly in another format, you must define a separate device for each format.

In the device settings you require the -format argument. You can use each rendering format that your report server supports. Which values are supported depends on the version of SQL Server Reporting Services.

This table shows the formats:

Format	Description	SSRS 2012	SSRS2016
XML	XML file with report data	Yes	Yes
CSV	CSV (comma delimited)	Yes	Yes
PDF	PDF	Yes	Yes
MHTML	MHTML (web archive)	Yes	Yes
EXCEL	Excel 2003 (xls)	Yes	Yes
EXCELOPENXML	Excel 2007 (xlsx)	Yes	Yes
IMAGE	TIFF file	Yes	Yes
WORD	Word 2003 (doc)	Yes	Yes
WORDOPENXML	Word 2007 (docx)	Yes	Yes

When you send the report to a device for which the -format argument is set, the specified format will be passed in the render request. After rendering the report, you will be asked whether you want to save or open the file.

Store Reports

You can store the rendered reports in a specific folder with the -path argument. Within the -path argument, you can use these placeholders:

- `$date`
- `$time`
- `$datetime`

- `$user`
- `$report`
- `$format`
- `$file`

For example:

`-path "f:\reports\$user\$report_$datetime"`

If the format is Excel, this will result in:

`f:\reports\johndoe\tdsls440101000_20101129102030.xls`

Note: The -path argument always refers to a directory that must be present on the LN server. You cannot specify a directory on the client PC. To access the files from the client PCs, you must connect to the directory by defining it as a network share.

To set a filename for the file to be stored on the file system you must assign a value to the predefined variable `spool.fileout`. If you specify the `$file` placeholder in the `-path` option of the Device argument, the value of `spool.fileout` is used to create the file. Note that `spool.fileout` must contain the full absolute path.

After `spool.close(), rprt_close(), brp.close()` you can implement your own processing for the file.

Device Argument Constraints

The following table shows the possible combinations of the arguments that can be used in the definition of a Microsoft Reporting Server device:

-printer -device	-format	-path	Action
✓	✓ PDF	✓	Output is printed on specified printer/device and also stored at the supplied location.
✓	✓ not PDF	✓	Not possible.
✓	✓ PDF	⊖	Output is printed on specified printer/device.
✓	✓ not PDF	⊖	Not possible.
✓	⊖	✓	Output is printed on specified printer/device and also stored at the supplied location in PDF format.
✓	⊖	⊖	Output is printed on specified printer/device.
⊖	✓	✓	Output is stored at the specified location in the specified format. This device can also be used for jobs although no printer is entered.
⊖	✓	⊖	Direct viewing in the specified format.

-printer -device	-format	-path	Action
⛔	⛔		Output is stored at the specified location in PDF.
⛔	⛔	⛔	Default direct viewing.

✅ = specified

⛔ = not specified

The -server argument must be specified for all Microsoft Reporting Server devices.

Advanced file processing

You can use External Reporting for a standard session. When you cannot assign a value to `spool.fileout`, use the **4GL Program** field of the **Device specification (ttaad3500m000/ttaad3100s000)** session. The specified value in this field is an own customized session. This session will be started as soon as the file is created. Several arguments are passed to this session.

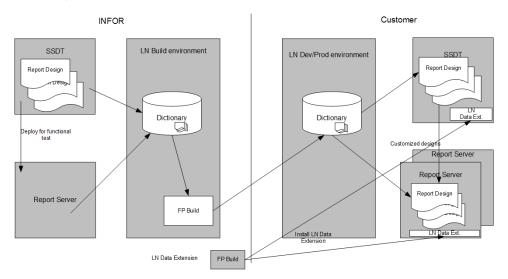| Argument	Description	Domain	Remark
1	Report File	string 256 MB	The file that has been created by External Reporting. It is the result of the substituted -path option of the device or a temporary file if the -path option was not specified.
2	Format	string 20	The file format, for example PDF, EXCEL. It is the format specified with the -format option. When the -format option is not specified, the format will be PDF.
3	Language	string 1	The language in which the report is created.
4	Device	string 14	The original External Reporting device to which the report was sent.
5	Process ID	long	The process id of the original session that produced the report. This process id can be used to retrieve variable values from the print session. Note that you must use the get.var() function. You cannot use the import() function, because the original session is not the parent session of the current session.

An example of the source code:

```
#pragma used dll ottdllbw
function main()
```

```
{
        string      report.file(256) mb
        string      report.format(20)
        string      report.language(1)
        string      device(20)
        string      printer(30)
        long        session.id
        long        exitcode
        long        ret

        report.file = argv$(1)
        report.format = argv$(2)
        report.language = argv$(3)
        device = argv$(4)
        session.id = lval(argv$(5))

        |* Use get.var(session.id, "<variable in session>", <your vari
able>) to retrieve variables from print session

        on case device
        case "PR1":
                printer = "\\prserver\p001"
                break
        case "PR2":
                printer = "\\prserver\p002"
                break
        default:
                printer = "\\prserver\p003"
                break
        endcase

        if report.language = "h" then
                printer = printer & "h"
        endif

        on case report.format
                case "PDF":    |* Print PDF file to printer
                        ret = server2client(report.file,
"${BSE_TMP}/temp.pdf", false)
                        ret = start.application.local(
                                sprintf$("C:\Program Files
(x86)\Adobe\Reader 10.0\Reader\AcroRd32.exe /t %s %s",
                                        "${BSE_TMP}/temp.pdf", printer),
                                true,
                                exitcode)
                        break
                case "EXCEL":  |* Show Excel file
                        ret = server2client(report.file,
"${BSE_TMP}/temp.xls", false)
                        ret = start.application.local(
"${BSE_TMP}/temp.xls", false, exitcode)
                        break
                default:
                        break
        endcase
}
```

Chapter 4: Distribute MS Report Designs

To render an LN report on the Report Server, a Microsoft report design for that report must exist on the Report Server. There are various ways of deploying such reports to the Report Server and also retrieving them for further customization.

This diagram shows how this process works:

At Infor, the report designs are created with the SQL Server Data Tools (SSDT). SSDT is a Visual Studio extension that is part of SQL Server. For more information about designing the reports, see the *Infor Enterprise Server Plug-in for Microsoft SQL Server Reporting Services - Development Guide* (U9657 US).

These report designs are deployed from SSDT to the Report Server to perform functional tests. These tests can go through multiple iterations. After testing is finished, you can import the report designs from the Report Server into the LN Dictionary of the Build Environment by using the **Import SSRS Report Design (ttrpi1200m100)** session.

All report designs that are imported will be included in a feature pack build, which can be delivered to customers and partners. Installing the feature pack makes the report designs available in the dictionary of that LN environment.

The next step is to export these reports to one of the following:

- A report design file to customize the reports further in SSDT
- Directly to the Report Server

Run the export process with the **Export SSRS Report Designs (ttrpi1200m100)** session. Customized reports can be deployed from SSDT to the Report Server. When deploying to a Report Server, the Report Server must be defined in the **Report Servers (ttrpi1500m000)** session as described in Configuration on the LN Server on page 9.

To redistribute the report designs, you can include them in a feature pack or a PMC solution.

With the **Import SSRS Report Design (ttrpi1200m100)** session, you can import the customized reports back into the LN environment from either the Report Server or a report design file.

To customize reports in SSDT and to render the reports created by the Report Server, the Infor ES Reporting plug-in must be installed.

These sessions are described in the next sections:

- **Import SSRS Report Designs (ttrpi1200m100)**
- **Export SSRS Report Designs (ttrpi1200m000)**

Import SSRS Report Designs (ttrpi1200m100)

The LN environment contains a dictionary in which report designs are stored. The designs in the dictionary are only used for storage and redistribution purposes, for example, between a development environment and a production environment.

The reports can be imported from the Report Server or from a report design file that is created by SSDT.

To import the MS report designs:

1   Use the classic tools or Web UI on your LN system. To start the **Import SSRS Report Designs (ttrpi1200m100)** session.
2   Specify this information in the **Selection Range** section:
- **Package**: the range of packages for which to import MS Report Designs stored as additional files.
- **Report**: the range of LN reports for which to import the MS Report Designs.
- **Version, Release, Customer**: the VRC of the LN reports for which to import the MS Report Designs.
3   In the **Options** section, you can optionally select the check boxes:
- **Overwrite**: Selecting this check box indicates whether an MS report is imported when its corresponding MS Report Design already exists. If this check box is selected, the MS Report Design is overwritten. If this check box is cleared, the MS Report Design is skipped.
- **Print Import Log**: indicates that the session shows/prints a log of all MS reports within the range. The log contains information about the import's success or failure and, if necessary, why it failed.

If the **Print Import Log** check box is selected, a **Select Device** dialog box is displayed. Select the device on which the log will be printed after the import process is completed. The selected device must be a Microsoft Report Server because the Import Log is an MS Report. This MS Report is automatically deployed to the Report Services if the report is not yet available.

4 In the **Import from** section, enter a value in one of these fields:

- **Server**: Enter the code of the Report Server from which to extract the MS Report Designs. The report server description and the folder in which the reports are gathered are shown. The report folder automatically contains the hostname and package combination of the user that you supplied in the **Report Servers (ttrpi1500m000)** session.
- **Client**: supply the folder on the client PC from which to import the MS Report Designs.

5 Click **Import** to import the files to the data dictionary.

To distribute the MS reports, you can include the report designs in a feature pack or a PMC solution.

Export SSRS Report Designs (ttrpi1200m000)

After installing a feature pack or a PMC solution, the report designs are placed in the LN Data Dictionary. To customize the reports in SSDT, these report designs can be exported to a file. The report designs can also be deployed on the Report Server.

To export the MS reports:

1 Use the classic tools or Web UI on your LN system. To start the **Export SSRS Report Designs (ttrpi1200m000)** session.

2 Specify this information in the **Selection Range** section:

- **Package**: the range of packages for which to export MS Report Designs stored as additional files.
- **Report**: the range of LN reports for which to export the MS Report Designs.
- **Version, Release, Customer**: the VRC of the LN reports for which to export the MS Report Designs.

3 Specify this information in the **Options** section:

- **Multi-level**: select this check box to export MS Report Designs that are not in the supplied Version, Release, or Customer, but are in an ancestor VRC.
- **VRC Depth**: The depth indicates up to and including which ancestor VRC the session will process for the export.
- **Overwrite**: Select this check box to indicate whether an MS report is exported when its corresponding MS Report Design exists at the export destination. If this check box is selected, the MS Report Design is overwritten. If the check box is cleared, the MS Report Design is skipped.
- **Print Export Log**: Select this check box to indicate that the session shows/prints a log of all the MS reports within the range. The log contains information about the export's success or failure and, if necessary, why it failed.

If the **Print Export Log** check box is selected, a **Select Device** dialog box is displayed. Select the device on which the log will be printed after the export process is completed. The selected device must be a Microsoft Report Server because the Export Log is an MS Report. This MS Report is automatically deployed to the Report Services if the report is not yet available.

4 In the **Export to** section, specify a value in one of these fields:

- **Server**: Specify the code of the Report Server to which the MS Report Designs are exported. The report server description and the folder in which the reports are gathered are shown. The report folder automatically contains the hostname and package combination of the user that you supplied in the **Report Servers (ttrpi1500m000)** session.
- **Client**: Specify the folder on the client PC to which MS Report Designs are exported.

5 Click **Export** to export the files.

Chapter 5: Troubleshooting

Logging

There are several ways to enable logging for the Microsoft Reporting for LN solution. You can use log files to find and solve issues.

Logging on the report server

Logging on the report server is performed by Apache log4net. Warnings, errors, and fatal errors are logged by default. The messages are logged to a file called `LNReporting.log`.

After a default installation, this file is located in:

```
C:\ProgramData\Infor\MSRS\LNReporting.log
```

The logging is configured by a configuration file called `log4net4ln.config`.

By default, this file is located in:

```
C:\ProgramData\Infor\MSRS\log4net4ln.config
```

The Infor ES Reporting plug-in installer installs the `log4net dll` file. The Infor ES plug-in will create a default `log4net4ln.config` file when it is run for the first time.

```xml
<?xml version="1.0" encoding="utf-8" ?>
<log4net>
  <appender name="RollingFile" type="log4net.Appender.RollingFileAppender">

    <file value="C:\ProgramData\Infor\MSRS\LNReporting.log" />
    <appendToFile value="true" />
    <maximumFileSize value="1MB" />
    <maxSizeRollBackups value="10" />

    <layout type="log4net.Layout.PatternLayout">
      <conversionPattern value="%level %date %logger - %message%newline"
/>
    </layout>
  </appender>
```

```
  <root>
    <level value="WARN" />
    <appender-ref ref="RollingFile" />
  </root>
</log4net>
```

To see all logging, change the <level value="WARN" /> line to:

<level value="DEBUG" />

Changes to the log4net4ln.config file are effective immediately. A restart of the Report Server is not required.

Logging on LN backend

Logging on the backend can be enabled by setting these environment variables.

- SSRS_SERVER_LOGGING
- SOAP_TRACE

SSRS_SERVER_LOGGING

This logging shows the communication between LN and the Report Server on a high level.

For Worktop you must change the BWC file to set this variable.

For Web UI you must change the Web UI profile. Click **Configure** to supply the command line with the environment variable.

After the change you must restart the running bshell. The messages will be logged in the file log.ssrs . You can find this file in the $BSE/log directory.

SOAP_TRACE

This logging supplies detailed information about the communication between LN and the Report Server.

The communication between LN and the Microsoft Reporting webservices is done through SOAP. To view the SOAP communication, add the following to the command path of the BWC file being used:

-set SOAP_TRACE=1

After this change you must restart the running bshell. The messages will be logged in several files. You can find these files in the $BSE/log directory. All file names start with soap. The last part of the name depends on the session using a SOAP request. For example:

- soap.ttstpsplclose
- soap.ttrpims.rs.rp