Sage 300 2023

Customizing Printed Forms

August 2022

Contents

Customizing Printed Forms and Reports with the SAP Crystal Reports Designer

This document describes how to use SAP Crystal Reports® for Sage 300 to modify the reports and forms that come with Sage 300, such as Accounts Receivable statements, Accounts Payable checks, Order Entry invoices, and Inventory Control stock labels.

All reports and business forms in Sage 300 *except G/L financial statements* are generated using Crystal Reports.

The chapter assumes that you are familiar with Crystal's capabilities.

Sage 300 Reports and Forms

Reports provided with Sage 300 fall into two main types:

- **Standard business forms.** You can use Crystal Reports to customize the following forms:

    - Statements
    - Checks
    - Credit notes
    - Invoices
    - Quotes
    - Order confirmations
    - Picking slips
    - Mailing, inventory, and shipping labels
    - Purchase requisitions
    - Purchase Orders

    This document provides some background on how datapipes work and how you use them with custom reports.

- **Standard accounting reports.** You can also use Crystal Reports to customize the standard reports that come with Sage 300, or to create new reports for particular reporting purposes.

Choosing Between Datapiped and ODBC Forms

Sage 300 provides a large number of business forms that can be used "as is" or be customized to suit the requirements of your company. Typically, each business form comes in two versions, one for use with preprinted forms, and one that prints a complete form on plain paper, with boxes, lines, etc.

However, some business forms provide more alternatives. For example, in Order Entry, you can customize the following invoice forms:

Datepiped forms
- OEINV01.RPT and OEINV02.RPT are datapiped reports, which means that they use a program to read information from the Sage 300 database and sort it for printing. Datapiped reports are faster, but not as easily customizable as ODBC-based reports.

  OEINV01.RPT is designed for printing on plain paper using a laser or ink-jet printer, whereas OEINV02.RPT is designed for preprinted forms.

ODBC forms
- OEINV04.RPT and OEINV05.RPT are ODBC-based versions of OEINV01.RPT and OEINV02.RPT. They read data directly from the Sage 300 database using standard database access methods.

  Note: By default, Order Entry prints the datapiped versions of these forms.

What Is a Datapipe?

A *datapipe* is a dynamic link library (DLL) program that performs database queries for reports or forms.

Because datapipes are compiled and optimized for specific reports, they can provide significant performance advantages over reports that use standard database access techniques (ODBC access).

The disadvantage of datapipes is that they provide access to a fixed group of fields. You cannot customize a datapiped report as much as you can customize non-datapiped one.

Modifying Forms Using Crystal Reports

You can edit ODBC-based reports and datapiped reports in Sage 300 using version 11 or higher of Crystal Reports.

For example, you can:

- Add graphic images.

- Add report titles.

- Add sections to the report to display subtotals and grand totals.

- Add text to a report, or edit existing report text.

- Change fonts and styles and format fields.

- Move fields around the report.

- Insert the following types of fields in ODBC reports:

  - **Database fields.** Database fields are the fields defined in the Sage 300 database and accessible using the existing database query or by modifying the database query in Crystal Reports.

    - To insert database fields, choose Field Object from the Insert menu, and select the database field from the Field Explorer window.

    - To expand the database query,

  - **Computed fields.** Computed fields are fields you can create to contain the results of computations on other fields on the report. You can use computed fields to calculate data, such as totaling numbers in a column or computing an average.

- Insert the following types of fields in datapiped reports:

  - **Datapipe fields.** Datapipe fields are the fields defined by the datapipe dynamic link library file. To insert datapipe fields, choose Field Object from the Insert menu, and select the database field from the Field Explorer window.

  - **Computed fields.** Computed fields are fields you can create to contain the results of computations on other fields on the report. You can use computed fields to calculate data, such as totaling numbers in a column or computing an average.

**Note:** You cannot add database fields to a datapipe report definition.

Review Existing Reports and Forms Before You Start

Print copies of the forms from Sage 300 or refer to the Reports chapters in each of the application *User Guides*. In most cases, you must have data before you can print the forms.

Editing an Existing ODBC Report with Crystal Reports

If you are editing an existing ODBC report, you must set the datasource name for the report so it matches the datasource on your computer.

1. Copy and rename the Crystal report that you want to change.

2. Open the renamed report, and choose Database, Set Datasource Location... from the Crystal Reports menu. The Set Datasource Location dialog box will appear.
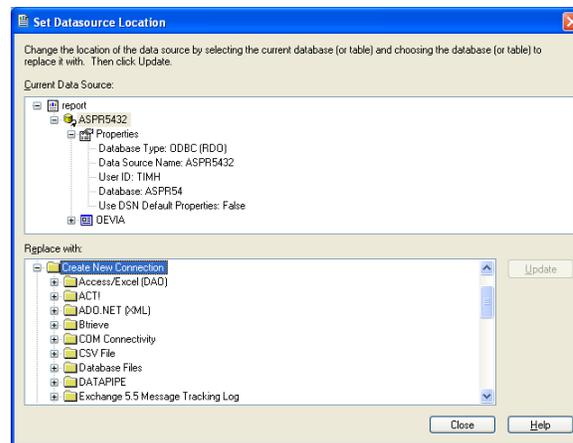
3. To change the current data source, expand Create New Connection in the Replace With window, scroll down to ODBC, and expand again.

4. Choose the ODBC data source for the report data, and click the Next button.
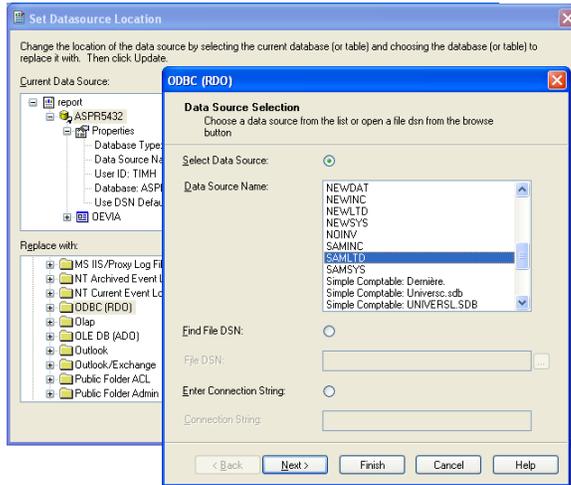
5. Enter the connection information for the ODBC connection and click the Finish button.

6. Select a table in the Current Data Source and one in the Replace With data source, and click the Update button.
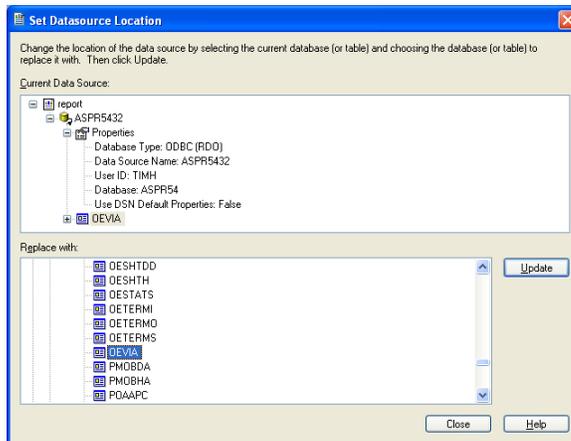
7.   Continue to update each table in the report.

8.   Click the close button when you are finished.

> **Note:** If the report has subreports, you must also set the data sources for each of the tables in each of the subreports.

Creating an ODBC Report with Crystal Reports

To create a new report with ODBC access to a Sage 300 database, follow these general steps.

1.   Open Crystal Reports and choose File, New, Standard Report to create a new blank report using Crystal's Standard Report Creation Wizard. The Data page will appear.
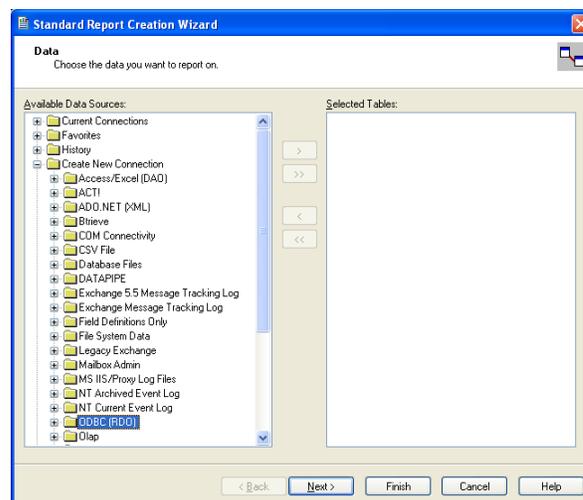
2.   Expand the Create New Connection folder in the Data window, and then expand ODBC to open a dialog from which you can select the data source that you want to use—in this case, SAMLTD.

acutedata
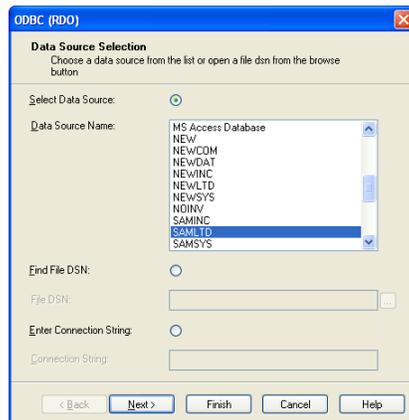
3. Choose the Next button after choosing the data source to move on to the connection information screen, where you enter log on information for designing the report.
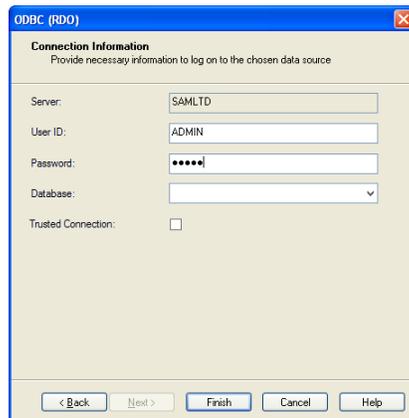
4. Click Finish.

5. Select the database tables that you need for the report.
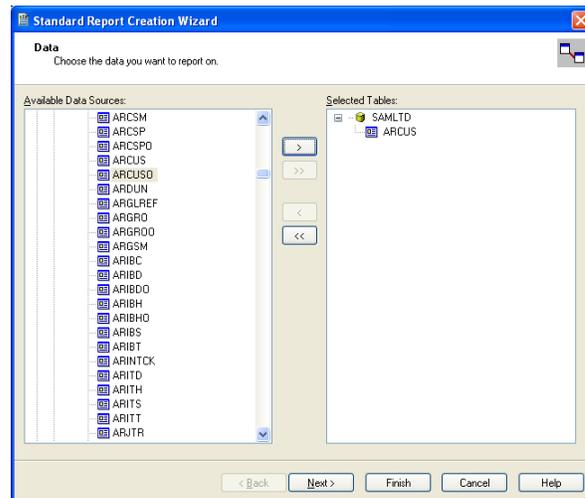
6. Save the report in the ENG directory for the application. (For example, AR61A\ENG.)

7. Add the optional parameter fields defined in the *xx*rpt.ini file. The name of the Crystal report parameter field should be the same as the corresponding optional parameter field defined in the *xx*rpt.ini file.

8. Add groups and formula fields, as needed.

9. Complete the report design (define sections, order, and so forth).

Creating a Datapipe Report with Crystal Reports

A datapipe is a dynamic link library (DLL). These DLLs are programs that perform database queries for some Sage 300 reports. These programs can provide significant performance advantages because you compile and optimize them for specific reports.

The datapipe DLL returns records to Crystal, one at a time.

Crystal reports that use a datapipe DLL also have a DATAPIPE.INI text file in the same directory where the DLL resides.

The text file includes one section for each datapipe DLL. Each DLL section contains a query entry listing, containing the following information:

• The query name and a list of the default values for the datapipe parameters, separated by "\n".

Note: These parameters have a one-to-one correspondence with the datapipe query entry of the report section in the *xx*rpt.ini file.

- Parameter values P1 through P4, which specify the default database from which Crystal will read when designing the report form.

If a datapipe contains multiple queries, it will list all of the queries in the datapipe.ini file. However, when you create a new Crystal datapipe report, all queries, except for the one required for the report, must be commented out in the datapipe.ini file.

Example:

This example shows the datapipe DLL section in the DATAPIPE.INI which is required to create the Crystal datapipe report, GLPJCON1.rpt.

```
[GLPJCON.DLL]
query="GLPJCON1\n0000\n999999\n \n0\n "
'query="GLPJCON3\n0000\n999999\n \n0\n "
'query="GLPJCON4\n0000\n999999\n \n0\n "
'PARAMETER        3       4           12 13
P1=SAMLTD
P2=ADMIN
P3=
P4=ENG
```

Parameters Used in Crystal Datapipe Reports

Crystal datapipe reports use the following parameter types:

- Datapipe design time parameters.

- Datapipe query (optional) parameters.

Datapipe Design Time Parameters

These parameters are only defined in the datapipe.ini file as they are only required during report development.

When a user starts a datapipe report from the Sage 300 desktop, the desktop passes information about the user's system to the datapipe. During development, you supply this information to Crystal in the datapipe.ini file.

- P1—This parameter specifies the database ID defined in Sage 300 Database Setup, for example, SAMINC.

- P2—This parameter specifies the user ID, for example, ADMIN).

- P3— Leave this parameter value blank.

- P4— This parameter specifies the language code for the report, for example, ENG.

Example:

Here is a sample entry from a datapipe.ini file, which shows the entry of the design time parameters.

```
[GLPPERR.DLL]
query="GLPPJER"
P1=SAMLTD
P2=ADMIN
P3=
P4=ENG
```

In this example, Crystal is directed to use the SAMINC database (as defined in Sage 300 Database Setup), and to logon to the database as user ADMIN. The report will be generated as an English-language report; therefore, it should reside in the application's ENG subdirectory.

Datapipe Query (Optional) Parameters

Datapipe query parameters (defined as optional parameters in the *xx*rpt.ini file) are the parameters passed from the report to the datapipe DLL. You enter these parameters in the report query.

These parameters may include:

- Key ranges.

- Possible sort orders.

- Data that must be passed in, such as a flag to select multicurrency.

The Report Initialization (*xx*rpt.ini) File

The report initialization file, *xx*rpt.ini, defines the reports for an application. It is divided into report name sections—one section for each application report.

For a report that uses datapipes, you must add the following three entries to its report name section in the *xx*rpt.ini file.

- type=datapipe

  This entry defines the report as a datapipe report.

- datapipe path=..\

  This entry tells the API where the report's datapipe is located. As reports are language-specific, they are stored in the application's language resource directory, for example GL61A\ENG.

Datapipes are stored in the application's parent directory, for example, GL61A, as they are not language specific. Datapipes only contain data retrieval and calculation functionality, and the same datapipe can be used for different language versions of the same report.

- datapipe query="*<Datapipe DLL>=<Query Name>*\n*<parameter1>*\n … *<parameterm>*\n"

This entry specifies the datapipe DLL name, and the query (report or subreport) name and parameter values used by a datapipe for the report. Parameter values are preceded by "\n".

> **Note:** Use the ^ symbol to separate datapipe subreport queries. For an example of a datapipe query which includes datapipe subreport entries, see the datapipe query line in the [GLPJ01] section of the GLRPT.INI file.

Example:

The following example shows the [GLBCH01] report section in the GLRPT.INI file for the GLBCH01 Crystal datapipe report. The corresponding datapipe DLL entry in the DATAPIPE.INI file follows.

```
[GLBCHL01]
heading=24020
crystal=glbchl01
orientation=landscape
paper size=1
optparams=21
type=datapipe
2=FRBATCH          STRING
3=TOBATCH          STRING
4=FRLDGR           STRING
5=TOLDGR           STRING
6=FRDATE           STRING
7=TODATE           STRING
8=TYPES            STRING
9=STATUS           STRING
10=FCURNDEC        STRING
11=QTYDEC          STRING
12=QTYHDG          STRING
13=UNITHDG         STRING
14=PATH            COMPANY
15=QUERY           STRING
16=EXT             FILEEXT
17=PRINTED         STRING
18=USEGS           STRING
19=USER            STRING
20=OPTIONALFIELDS  STRING
21=ICTINSTALLED    STRING
22=ICTMULTI        STRING
datapipe path=..\
datapipe
query="glbchl.dll=GLBCHL01\n%FRBATCH%\n%TOBATCH%\n%FRLDGR%\n%TOLDGR%\n
```

```
%FRDATE%\n%TODATE%\n%QUERY%\n%PRINTED%\n%USEGS%\n%USER%\n%OPTIONALFIEL
DS%\n%ICTINSTALLED%\n%ICTMULTI%"
```

In the query line for this example, the datapipe DLL is identified as glbch1.dll, the query name (report name) is GLBCHL01, and 13 of the 21 parameters defined for the report will be passed to the datapipe. Parameter names begin and end with the "%" character, and are separated by a preceding "\n".

The parameter names are the same as those listed in the GLRPT.INI file for the respective report. For example, %TOBATCH% substitutes the current value of the report's TOBATCH parameter.

If the report has datapiped subreports, then the datapipe query string contains multiple sections separated by the "^" symbol. The datapipe DLL name and query name in the query entry line correspond to the DLL section name ([GLBCHL.DLL]) and the query name (GLBCHL01) defined in the DATAPIPE.INI file, as noted below:

```
[GLBCHL.DLL]
query="GLBCHL01\n0\n99999\n \nZZ\n20050101\n99991231\n0\n \n0\n \n1
\n0 \n0"
P1=SAMLTD
P2=ADMIN
P3=
P4=ENG
```

Creating the Datapiped Crystal Report Specification

The following example creates a standard report in Crystal Reports version 11.

1. Open Crystal Reports and choose File, New, Standard Report to create a new blank report using Crystal's Standard Report Creation Wizard. The Data page will appear.
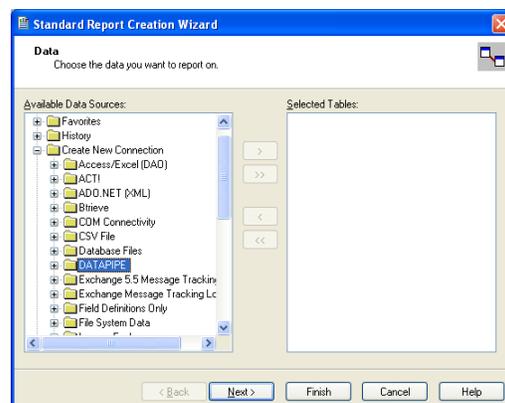
2. Expand the Create New Connection folder in the Data window, and then expand Database Files to open a dialog from which you can search for the datapipe DLL file.

3. Navigate to the directory that contains the datapipe DLL file (this directory must also contain the datapipe.ini file).

4. Click the "Files of type" drop-down list options, and select the "All Files" option.
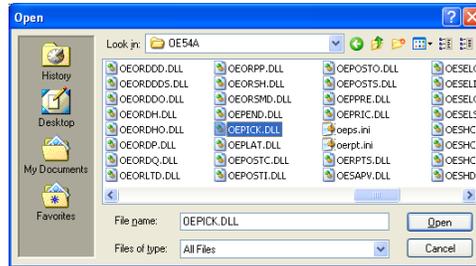
5. Double-click the datapipe DLL file to add it to the Database Files folder in the Data Explorer window.

   In the example below, we added the OEPICK datapipe to the report.
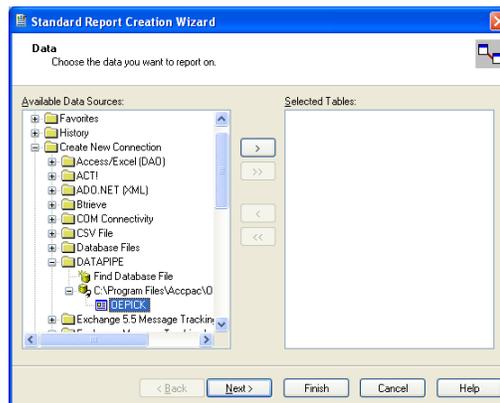
6. Click the Finish button to add the DLL to the report.

7. Save the report in the ENG directory for the application. (In our example, OE61A\ENG.)

8. Change the database location as follows:

   a. Select "Set Location" from the Database menu.

   b. In the File location field, delete all the text after the datapipe DLL name and precede the datapipe DLL name with "..\". For example, the text in the File location field for the datapipe OEPICK.DLL should be ..\OEPICK.DLL.
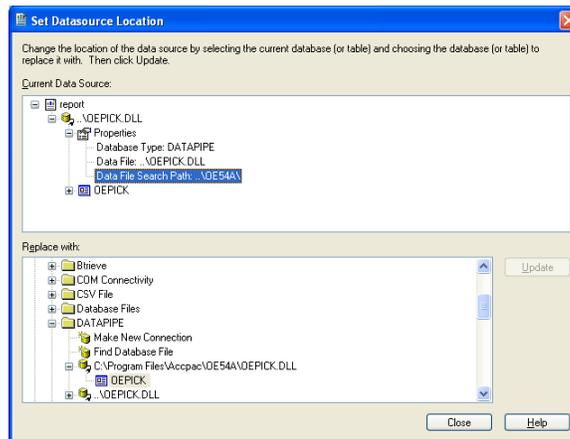
c. Click the Close button to save your changes and close the dialog.

9. Save the report again.

10. Add the optional parameter fields defined in the *xx*rpt.ini file. The name of the Crystal report parameter field should be the same as the corresponding optional parameter field defined in the *xx*rpt.ini file.

11. Add groups and formula fields, as needed.

12. Complete the report design (define sections, order, and so forth).

Integrating Crystal Reports into Sage 300

There are several ways to integrate a custom Crystal report into an existing application, you can:

- Add a report to the Sage 300 Desktop as a new report object.

- Record a macro and add it to the Sage 300 Desktop (so the user can more easily enter criteria).

- You can replace an existing report and use the existing report UI.

If you use a macro to start a new report, you must define your report in the Sage 300 application's report initialization file, *xx*rpt.ini. For more information, see the following section, "The Report Descriptor File."

If you are replacing a standard Sage 300 report with a new one, you may have to change *xx*rpt.ini.

To run a macro which displays a Crystal report from the Sage 300 Web Desktop, the Crystal ActiveX Report Viewer must have been previously downloaded and installed on the client machine. If this viewer is not currently installed on the client, the simplest way to initiate the download and install process is to run a Crystal report from the Sage 300 Web Desktop.

The Report Descriptor File

The report descriptor file is an ASCII file that lists the names and parameters of all the reports in a Sage 300 application.

The report descriptor file is called *xxvvv\xx*RPT.INI, where *xx* is the two-letter application id, and *vvv* is the program version. For example, the report descriptor file for General Ledger version 6.2A would be GL62A\GLRPT.INI.

Standard File Format

Each report is described in a separate section. Each section begins with the report name, and descriptor file statements follow the report name. For example, the following section describes a picking slip in Order Entry:

```
[OEPICK01]
type=DATAPIPE
heading=28030
crystal=OEPICK01
orientation=portrait
paper size=1
optparams=12
2=SELECTBY         STRING  28038
3=SORTBY           STRING  28038
4=FROMSELECT       STRING  28035
5=TOSELECT         STRING  28036
6=FROMLOC          STRING  28261
7=TOLOC            STRING  28262
8=REPRINT          STRING  28037
9=QTYDEC           STRING  28260
10=COMPLETED       STRING
11=PRINTKIT        STRING
12=PRINTBOM        STRING
13=SESHNDL         STRING
datapipe path=..\
datapipe
query="oepick.dll=OEPICK01\n%SELECTBY%\n%FROMSELECT%\n%TOSELECT%\n%FRO
MLOC%\n%TOLOC%\n%REPRINT%\n%QTYDEC%\n%COMPLETED%\n0\n0\n0\n0\n%SESHNDL
%"
```

The descriptor file statements are explained in the following paragraphs:

type

For a datapipe report, include a type=datapipe line. Do not include in a non-datapiped report.

heading

heading=*string resource number* | "*heading string*"

This is the heading of the report window. It can be either the resource number of a string in the application's language-specific resource DLL, or a string in quotes.

crystal

crystal=*crystal file name*

This required statement designates a Crystal Report. The file extension, .rpt, is not included. The default directory for this file is the application's language-specific subdirectory. The default extension is .RPT. You can include a complete path name if you want.

orientation

orientation=*orientation*

This specifies the default orientation for this report. It is either portrait or landscape.

paper size

paper size=*paper size*

This specifies the default paper size for this report. It is a number from PRINT.H in the Windows SDK INCLUDE directory. (For example, 1 = letter, 2 = legal, and so on.)

optparams

optparams=*number of optional parameters*

This optional statement identifies the number of optional parameters. (If you omit this statement, zero is assumed.)

[number]

[number]=*option name   option type   list   description*

These statements describe the name and format of each optional parameter. There must be the same number of optional parameters as specified in the optparams statement.

- **number** is the number of the optional parameter. You must include a specification for each optional parameter. These were originally designed to be the caret parameter number.

- *option name* is the parameter name.

- *option type* is the parameter type. Valid parameter types are STRING, LIST, COMPANY, SYSTEM, SITE, and FILEEXT. These types are described later.

acutedata

- *[list]* is additional descriptive information in the list, given as a list of integer values and symbolic strings, where the parameter type is LIST.

  The values and strings are separated by commas; spaces are not allowed. The symbolic strings can be either string resource numbers for the language-specific resource DLL, or quoted strings.

- *description* is a text resource string number or string that is used to document the parameter during macro recording.

datapipe path

This entry tells the API where the report's datapipe is located. Since datapipes are not language-specific, they are placed in the application's parent directory. Reports, on the other hand, are language-specific, and so, are placed in the language subdirectory of the application's parent directory. Set this statement to datapipe path=..\ to instruct the API to look for the datapipe DLL in the application's parent directory.

Do not include in a non-datapiped report.

datapipe query

This entry specifies the datapipe DLL name, and the query name and parameter values (separated by \n) used by a datapipe for the report.

Do not include in a non-datapiped report.

> **Note:** Use the ^ symbol to separate datapipe subreport queries.

Possible parameter types are:

Parameter Type	Description
STRING	For a simple string. If the parameter type is STRING, then no additional descriptive information is given.
LIST	For mapping a list of values to string resources. The list consists of a set of pairs of values and string resources.
COMPANY	For setting the table path. No translation is necessary, because the database engine uses the given table name (for example, GLASV) to figure out how to access the actual data.
SYSTEM	For setting the table path to the company's system directory. No translation is necessary, because the database engine uses the given table name (for example, GLASV) to figure out how to access the actual data.

Datapipe reports have the following additional statements:

type

For a datapipe report, include a type=datapipe line.

datapipe path

This entry tells the API where the report's datapipe is located. Since datapipes are not language-specific, they are placed in the application's parent directory. Reports, on the other hand, are language-specific, and so, are placed in the language subdirectory of the application's parent directory. Set this statement to datapipe path=..\ to instruct the API to look for the datapipe DLL in the application's parent directory.

datapipe query

This entry specifies the datapipe DLL name, and the query name and parameter values (separated by \n) used by a datapipe for the report.

Note: Use the ^ symbol to separate datapipe subreport queries.

Report Style Guidelines

Sage 300 uses set of report style guidelines to ensure a level of consistency in all Sage 300 reports. The guidelines cover both external and internal issues. Internal consistency is helpful if users or programmers need to change report definitions.

The guidelines are:

- Standard font and point size for reports is Arial 8.

- The first line of the page heading contains the following three entries:

    – The first entry is the date and time, left-justified. The date is in "Long Date" format. The time is formatted as *h:mm* AM/PM.

    – The second entry is the company name, centered. Size the field to accommodate the full length of the company name.

    – The third entry is the page number, right-justified.

- Column titles are underlined and bold.

- Company name, report headings, and labels should be bold. Other text should be regular.

- The first report parameter is CMPNAME—the company name.

- The first letter of each word is capitalized in the report name and column headings.

- Report options should only be printed on the first page. The Crystal report filename (without the .rpt extension) is always printed in parentheses to the right of the report title.

- The first letter of each sentence is capitalized in field labels for report options, footers, and report body text.

- Database field names are all upper case, to differentiate them from calculated fields. Calculated field names are mixed upper and lower case.

- For portrait-oriented reports, margins are:

  – Left, .75
  – Right, at least .35
  – Top, .5
  – Bottom, .5

- For landscape-oriented reports, margins are:

  – Left, at least .35
  – Right, at least .35
  – Top, .75
  – Bottom, .5

acutedata