Centrallix Reporting System Reference

The Centrallix reporting system provides the ability to generate reports in a variety of formats from ObjectSystem data and a report object. This document describes the report object structure in detail and how to use the various report components to build a report.

Reports can have parameters - see the 'report' object for more details. Parameters are passed to a report via HTTP URL parameters, and are defined in the report by top-level attributes other than 'title' and 'titlebar'.

Sample code is generally given in "structure file" format, which is the normal format for the building of applications. However, other suitable object-structured data formats can be used, including XML.

Where a 'moneyformat' is specified, it is a text string in which the following special characters are recognized (the default money format is "$0.00"):

#	- Optional digit unless after the decimal point or the digit immediately before the decimal point.
0	- Mandatory digit, no leading zero suppression.
,	- Insert a comma, suppressed if no digits around it.
.	- Decimal point (only one allowed).
$	- Dollar sign.
+	- Sign, + if positive or zero, - if negative.
–	- Sign, blank if positive or zero, - if negative.
()	- Surround number with () if it is negative.
[]	- Surround number with () if it is positive or zero.
	- (space) optional digit, but put a space in its place if suppressing leading 0's.

Where a 'dateformat' is specified, it is a text string with the following character sequences recognized (default is "dd MMM yyyy HH:mm"):

dd	- Two digit day of month (01 through 31).
ddd	- Day of month plus cardinality (1st through 31st).
MMMM	- Full (long) month name (January through December).
MMM	- Three-letter month abbreviation (Jan through Dec).
MM	- Two digit month (01 through 12).
yy	- Two digit year (00 through 99).
yyyy	- Four digit year (0000 through 9999).
HH	- Hour in 24-hour format (00 through 23).
hh	- Hour in 12-hour format (00 AM through 11 PM).
mm	- Minutes (00 through 59).
ss	- Seconds (00 through 59).
I	- At the beginning of the format, indicates that the date is in international (dd/mm/yy) order rather than U.S. (mm/dd/yy) order. Used mainly when a date is being input rather than when it is being generated.

Copyright (c) 1998-2003 LightSys Technology Services, Inc.

Documentation on the following report components is available:

report	The top-level report object
comment	A textual comment to be printed on the report
column	Prints a single column value from the source query

data	A data component in the report consisting of an arbitrary expression
section	An overall section of the report, possibly multicolumnar
form	A freeform layout container for displaying query results
table	A tabular presentation of report data
query	A definition of a SQL query to use elsewhere in the report

report

type: **system/report**
visual: **yes**
container: **yes**

Overview:

The report object is the top-level container for a report, and is the first object that should be listed in the structure file. It basically embodies the report itself.

Although there are only two properties listed for this component, others can be specified. If they are, they become report parameters which can be referenced later on in the report and can be modified by the user running the report.

Usage:

This component should only be used at the top level, and should never be used inside another container. It may contain other report components such as queries, forms, sections, tables, etc.

Properties:

Property	Type	Description
title	string	The title for the report.
titlebar	yes/no	Whether the report will contain a title section at the top giving the title, requesting user, date, and time.

Child Properties:

none currently available

Sample Code:

```
$Version=2$
myreport "system/report"
        {
        minsize = 1;  // a parameter.
        title = "This is a hello-world report";

        comment "report/comment"
                {
                text = "Hello World";
                }
        }
```

comment

type: **report/comment**
visual: **yes**

container: **no**

Overview:

The comment report component simply specifies a static string of text to be placed on the report with a given style at a given location.

Usage:

This component can be used within a section, form, or at the top level within the report object. It cannot contain any components.

Properties:

Property	Type	Description
autonewline	yes/no	Whether to automatically emit a newline at the end of text. Defaults to "yes".
font	string	The font to use. Available fonts are "times", "helvetica", and "courier".
relypos	integer	The vertical position relative to the current section, in lines (6ths of an inch).
style	stringvec	A list of styles, separated by commas. Values include "bold", "italic", "underline", "compressed", "barcode", "expanded", and "center". The "barcode" style converts a number into a USPS barcode for mailings.
text	string	The text to print on the report.
xpos	integer	The horizontal position, in character cells (10ths of an inch).
ypos	integer	The vertical position relative to the top of the page, in lines (6ths of an inch).

Child Properties:

none currently available

Sample Code:

```
$Version=2$
myComment "report/comment"
        {
        xpos = 10;
        text = "This is printed starting at column number 10.";
        }
```

column

type: **report/column**
visual: **yes**
container: **no**

Overview:

The column component allows the value of a source query's data field to be printed.

Usage:

This component can be used within a section, form, or at the top level within the report object. It cannot contain any components.

Properties:

Property	Type	Description
autonewline	yes/no	Whether to automatically emit a newline at the end of text. Defaults to "yes".
dateformat	string	If the field has a date/time data type, this allows the format to be specified. See the introduction to the reporting system documentation for date/time format information.
font	string	The font to use. Available fonts are "times", "helvetica", and "courier".

moneyformat	string	If the field has a money data type, this allows the format to be specified. See the introduction to the reporting system documentation for money format information.
nullformat	string	Sets the text to use if the data is NULL. Normally "NULL" is displayed.
relypos	integer	The vertical position relative to the current section, in lines (6ths of an inch).
source	stringvec	The source query to use, and the column name in the source query, separated by a comma.
style	stringvec	A list of styles, separated by commas. Values include "bold", "italic", "underline", "compressed", "barcode", "expanded", and "center". The "barcode" style converts a number into a USPS barcode for mailings.
xpos	integer	The horizontal position, in character cells (10ths of an inch).
ypos	integer	The vertical position relative to the top of the page, in lines (6ths of an inch).

Child Properties:

none currently available

Sample Code:

```
$Version=2$
myColumn "report/column"
        {
        relypos = 5; // five lines down from top of current section
        style = bold,italic;
        source = myQuery,modified_date;
        dateformat = "MMM ddd, yyyy";
        }
```

data

type: **report/data**
visual: **yes**
container: **no**

Overview:

The data component allows the value of an arbitrary expression to be printed. The expression can contain references to report parameters (via the object 'this') or to query fields.

Usage:

This component can be used within a section, form, or at the top level within the report object. It cannot contain any components.

Properties:

Property	Type	Description
autonewline	yes/no	Whether to automatically emit a newline at the end of text. Defaults to "no".
dateformat	string	If the field has a date/time data type, this allows the format to be specified. See the introduction to the reporting system documentation for date/time format information.
font	string	The font to use. Available fonts are "times", "helvetica", and "courier".
moneyformat	string	If the field has a money data type, this allows the format to be specified. See the introduction to the reporting system documentation for money format information.
nullformat	string	Sets the text to use if the data is NULL. Normally "NULL" is displayed.
relypos	integer	The vertical position relative to the current section, in lines (6ths of an inch).
style	stringvec	A list of styles, separated by commas. Values include "bold", "italic", "underline", "compressed", "barcode", "expanded", and "center". The "barcode" style converts a number into a USPS barcode for mailings.
value	string	A quoted expression (inside a string); if string literals are provided inside the expression, they have to be quoted again, as in "'one' + 'two'".
xpos	integer	The horizontal position, in character cells (10ths of an inch).

ypos	integer	The vertical position relative to the top of the page, in lines (6ths of an inch).

Child Properties:

none currently available

Sample Code:

```
$Version=2$
myData "report/data"
        {
        value = ":myQuery:string1 + ', ' + :myQuery:string2 + '.'";
        }
```

section

type: **report/section**
visual: **yes**
container: **yes**

Overview:

The section component of a report basically defines a general subpart of the report, possibly having multiple columns (newspaper-style) plus margin settings and more. It is also useful for defining a part of the report, perhaps just inside of a form, in which you need to position text or data relative not to the page as a whole but rather to a chunk of information or data.

Sections may span multiple pages.

Usage:

Sections can be used inside of components which are containers. They can contain other components, except for report/query and system/report components, and cannot contain other section components.

Properties:

Property	Type	Description
colsep	integer	The separation between columns, if more than one was specified, in 10ths of an inch.
columns	integer	The number of columns for the section (newspaper style). Default is 1.
dateformat	string	Sets the default date/time format. See the introduction to the reporting system documentation for date/time format information.
font	string	The font to use. Available fonts are "times", "helvetica", and "courier".
linespacing	integer	Sets the number of lines per inch. Default is 6 lines per inch.
margins	intvec	The left and right margins for the section, relative to the page. Defaults to 0,0 (no margins).
moneyformat	string	Sets the default money format. See the introduction to the reporting system documentation for money format information.
nullformat	string	Sets the default text to use when data is NULL. Normally "NULL" is displayed.
page	integer	Sets the starting page number for the section.
style	stringvec	A list of styles, separated by commas. Values include "bold", "italic", "underline", "compressed", "barcode", "expanded", and "center". The "barcode" style converts a number into a USPS barcode for mailings.
title	string	A title for the section. If provided, it will be centered above the section.

Child Properties:

none currently available

Sample Code:

```
$Version=2$
mySection "report/section"
        {
        title = "The Next Section";
        columns = 2;
        colsep = 2;          // 39 char wide cols on an 80 col page
        nullformat = "";     // display nothing if data is null instead of displaying 'NULL'

        //
        // more components go here
        //
        }
```

form

type: **report/form**
visual: **yes**
container: **yes**

Overview:

The form provides a way to display query results in a non-tabular fashion. Each individual piece of data has to be manually positioned, requiring more effort but giving more flexibility. The form will essentially iterate through its contents for each record it retrieves from a query.

Forms can have one of three modes: normal, outer, and inner. A 'normal' form simply starts the query, retrieves all of the results from the query (up to the reclimit maximum), iterating over its content once for each record, and then ends the query, discarding any data remaining. An 'outer' form starts the query and then iterates over its content while more results are available, but does not actually retrieve the results itself. An 'inner' form works only (directly or indirectly) inside an 'outer' form, and retrieves records (up to the reclimit maximum) but does not start or end the query.

The typical reason to use 'outer' and 'inner' mode forms is to group query results in chunks of a certain size, perhaps in a part of a multipage preprinted form such as a receipt or invoice. The 'outer' mode form is used to surround the multipage preprinted forms as a whole, and the 'inner' mode form is used to generate the section of the report that contains the list that carries over from page to page but in an area not the size of the entire form (and possibly having other data above and/or below it).

Forms can also be used with multiple queries simultaneously. They support running these queries in one of several fashions: 'nested', 'parallel', and 'multinested'. In the first, 'nested', if two queries were specified, the second query would be run once in its entirety for each record returned from the first - this is identical to nesting multiple forms inside of one another, each referencing only one query.

1,2,3	a,b
1	a
1	b
2	a
2	b
3	a
3	b

The second, 'parallel', runs the queries simultaneously and independently of one another - the results are combined together on the output - and continues until no query has more records.

1,2,3	a,b
1	a
2	b
3	NULL

The third, 'multinested', is more interesting - here the form iterates once through each query before the nested queries are run, giving each individual record its own unique line.

1,2,3	a,b

```
1    NULL
1    a
1    b
2    NULL
2    a
2    b
3    NULL
3    a
3    b
```

Usage:

May be used inside any visual container, and may contain any visual component or container.

Properties:

Property	Type	Description
dateformat	string	Sets the default date/time format. See the introduction to the reporting system documentation for date/time format information.
ffsep	yes/no	Whether to do a page break between successive records. Default 'no'.
font	string	The font to use. Available fonts are "times", "helvetica", and "courier".
mode	string	The mode of the form (see overview). 'normal' is the default.
moneyformat	string	Sets the default money format. See the introduction to the reporting system documentation for money format information.
multimode	string	How the form handles multiple queries (see overview). 'nested' is the default.
nullformat	string	Sets the default text to use when data is NULL. Normally "NULL" is displayed.
page	integer	Sets the starting page number for the form. Used sometimes if the page number has to be reset to 1, as in reports containing a series of multipage parts (such as multi-page invoices).
reclimit	integer	Sets a limit on the number of records the form is to print.
relylimit	integer	Sets a limit on the height of the form's contents inside a section or page. The form will continue to print records until this number of lines on the report have been used since the top of the current section.
rulesep	yes/no	Whether to emit a horizontal line between successive records. Default 'no'.
source	stringvec	A list of one or more query(ies) to run for this form. If more than one is specified, use the 'multimode' to determine how they are combined.
style	stringvec	A list of styles, separated by commas. Values include "bold", "italic", "underline", "compressed", "barcode", "expanded", and "center". The "barcode" style converts a number into a USPS barcode for mailings.

Child Properties:

none currently available

Sample Code:

```
$Version=2$
normalForm "report/form"
        {
        source=myQuery;
        rulesep=yes;

        //
        // query data is printed here with report/data, report/column, etc.
        //
        }
outerForm "report/form"
        {
        source=myQuery;
        mode=outer;
        comment "report/comment" { text="Here are up to six of the records:"; }
        innerForm "report/form"
                {
                source=myQuery;
                mode=inner;
                reclimit=6;

                //
                // display query data here with report/data, report/column, etc.
```

```
                    //
                }
        }
```

table

type: **report/table**
visual: **yes**
container: **no**

Overview:

The table component is used to present data in an orderly, tabular fashion, in rows and columns. While less flexible than the form, it is much easier to use.

Tables, like forms, can have more than one 'mode'. However, tables only support 'normal' and 'inner'; 'outer' is not supported since tables do not contain other components. See the form documentation for more on the mode.

Tables also can handle multiple queries using a 'multimode' - see the form for more information.

Usage:

Tables can be used inside any visual container, but cannot contain other objects

Properties:

Property	Type	Description
align	stringvec	A list of strings defining how the data in each column should be aligned to its cell. Values are 'left', 'right', and 'center'.
colsep	integer	Sets the separation between the table's columns, in 10ths of an inch.
columns	stringvec	A list of field names from the query source (expressions 'no') or expressions to evaluate for column information (expressions 'yes').
colnames	stringvec	A list of column header titles.
dateformat	string	Sets the default date/time format. See the introduction to the reporting system documentation for date/time format information.
expressions	yes/no	Determines whether the information given for columns in the 'columns' property are column names or full Centrallix expressions. Default is 'no'. You will want to set this to 'yes' if the 'source' property contains more than one query.
font	string	The font to use. Available fonts are "times", "helvetica", and "courier".
headeralign	stringvec	A list of strings defining how the header names in each column should be aligned. Values are 'left', 'right', and 'center'.
lowersep	yes/no	Whether to display a separator line at the bottom of the table in addition to the one drawn between the table and its header (titlebar). Default is 'no'.
mode	string	The mode of the table (see overview). 'normal' is the default.
moneyformat	string	Sets the default money format. See the introduction to the reporting system documentation for money format information.
multimode	string	How the table handles multiple queries (see overview). 'nested' is the default.
nodatamsg	yes/no	Whether to display the '(no data returned)' message below the table if no records were returned from the query source. Default is 'yes' to display the no-data message.
nullformat	string	Sets the default text to use when data is NULL. Normally "NULL" is displayed.
reclimit	integer	Sets a limit on the number of records the table is to print.
source	stringvec	A list of one or more query(ies) to run for this form. If more than one is specified, use the 'multimode' to determine how they are combined.
style	stringvec	A list of styles, separated by commas. Values include "bold", "italic", "underline", "compressed", "barcode", "expanded", and "center". The "barcode" style converts a number into a USPS barcode for mailings.
titlebar	yes/no	Whether to display a title bar on the table with column header names. Default is 'yes' to display it.
widthmode	string	How the table computes column widths. 'relative' means column widths are relative to the size of the font, whereas 'absolute' means tenths of an inch.

widths	intvec	A list of numbers giving the widths of the columns.

Child Properties:

none currently available

Sample Code:

```
$Version=2$
myTable "report/table"
        {
        source=myQuery;
        expressions=yes;
        colnames='Name','Size';
        columns=':myQuery:name', 'condition(:myQuery:size == 0, \'empty\', :myQuery:size)';
        align=left,right;
        headeralign=left,left;
        titlebar=yes;
        }
```

query

type: **report/query**
visual: **no**
container: **no**

Overview:

The query is used to define a SQL statement that will be used to obtain data from the ObjectSystem. It in and of itself does not actually run the query, but must be invoked by a table or form elsewhere in the report.

Queries can be linked to each other in order to depend on each others' data results.

Queries can also contain aggregate definitions, which are similar to 'COMPUTE BY' statements in other SQL languages, and calculate aggregate values as a the report progresses. These aggregates are normally used to provide summary data at the end of the query or at the end of a part of the query.

Usage:

Queries can only be contained within a 'system/report'. They can only contain 'report/aggregate' components.

Properties:

Property	Type	Description
sql	string	A SQL statement. It can include report parameter substitution by embedding '&xyz' in the report to reference the parameter 'xyz'. To referenced linked data from other queries, use '&1' and '&2' (and so forth) to reference the first and second links, respectively.
link	stringvec	Two strings, one specifying the query to link to, and the second specifying the name of the column/field to reference in that query.

Child Properties:

(of report/aggregate child components)

Property	Type	Description
compute	string	An expression specifying the computation to be performed. Normally this should include one or more aggregate functions (sum, count, avg, min, and max) in some form or fashion.
where	string	A conditional expression specifying which rows should be included in the aggregate computation.

Sample Code:

```
$Version=2$
myQuery "report/query"
```

```
    {
    sql = "select :name, :size from /samples where :size >= &minsize";
    totalsize "report/aggregate" { compute = "sum(:size)"; }
    bigfilesizetotal "report/aggregate" { compute = "sum(:size)"; where=":size > 10000"; }
    }
```