

**Technical  
White  
Paper**



**Automating the Generation and  
Secure Distribution of Excel Reports**

# Table of Contents

- Introduction..... 3
- Creating Spreadsheet Reports: A Cumbersome and Manual Process..... 3
- Distributing Spreadsheet Reports: An Administrative Burden With Security Risks ..... 5
- Repetition and Errors: Practice Does Not Always Make Perfect..... 6
- Requirements for Improving Efficiency and Minimizing Errors ..... 6
- Using Spreadsheet Report Templates and a Report Server to Automate Excel Reporting ..... 8
- Creating Templates for Repeatable and Reliable Spreadsheet Reporting..... 9
- Server-based Generation, Management and Distribution of Excel Reports ..... 12
- Additional Resources..... 16

## Introduction

Over 250 million people know how to use Microsoft Excel. Because so many end users are comfortable with using Excel for reviewing data and for conducting “what-if” analysis, delivering reports in spreadsheet format is a common requirement for reporting projects.

Many existing reporting tools that are designed to deliver reports in Web or PDF format offer the option to save those reports in a CSV or XLS format. The drawback with these solutions is that the resulting spreadsheet reports typically lack visual formatting (in the case of CSV files), live Excel formulas for computed values, charts that are linked to the relevant underlying data, and other Excel visualization features that users expect. As a result, users are unable to perform “what-if” analyses without manually recreating formulas and charts every time they get a new report – a task that is both time consuming and error prone.

In response to high end user demand for reports in Excel format, many organizations have assigned individuals in the IT department with the task of producing spreadsheet reports that contain the required data and formatting. While this centralizes the task of creating reports and adding the needed Excel functionality to the reports, the process still remains a manual one, with considerable scope for introducing errors when putting the reports together, and creating an additional burden of report distribution.

Once the reports are created, these individuals typically email the reports out. As the numbers of reports or users receiving reports grow, this manual distribution quickly becomes an administrative time sink. In addition, the risk of compromising data security is increased as reports may inadvertently be emailed to the wrong people.

This paper outlines the reasons why generating and distributing spreadsheet reports is so tedious and error prone by examining the typical steps in the process. By reviewing these steps in detail, we can develop a clear understanding of the technical requirements for an automated solution. Finally, the paper introduces a solution for automating the generation and secure distribution of spreadsheet reports based on Actuate’s e.Spreadsheet Designer and iServer Express.

## Creating Spreadsheet Reports: A Cumbersome and Manual Process

The creation of reports containing the desired Excel formatting involves a sequence of manual steps, from querying the underlying data sources, importing the results of queries into spreadsheets, formatting the data and adding the required Excel features.

## Manual Data Pulls

Excel reports may be used for regular review of operational metrics (e.g., weekly reporting on Key Performance Indicators) or for on-demand, ad-hoc analysis of information. In both of these cases, users need to be presented with data that is up-to-date. Preparing the data for presenting in these Excel reports is typically time-consuming and repetitious.

An analyst or administrator who has access to the relevant underlying databases needs to invoke one or more SQL scripts every time a report with updated data is required. For regular operational reporting, this manual execution is necessary on a periodic basis (daily, weekly, etc.) For ad-hoc analysis requests, the analyst needs to interrupt their regular workflow in order to produce the updated data. Running these scripts may also require the installation and configuration of database client software on computers.

Typically, these SQL scripts produce comma delimited data files that can then be opened in Excel as CSV files. There are Excel plug-in technologies from business intelligence software vendors that can help eliminate some of the manual running of SQL scripts. These solutions, however, are helpful only in the case where all the needed data is available through the semantic layer provided by the business intelligence tool.

## Repetitive Formatting of Data

Once the right data is pulled via a SQL script, the Excel report needs to be manually updated with the formatting and calculations that users require. For example, fonts may need to be changed on data values, column or row heading labels may need to be added, width and spacing of cells, rows and columns may need to be adjusted. If users require summary and drilldown capabilities while viewing data, Excel outlining features that support this summarization must be manually added to the appropriate rows and columns.

## Adding Necessary Formulas and Charts for “What-If” Analysis

In addition to these visual changes, an analyst would also need to edit the spreadsheet to enable “what-if” analysis by users. Spreadsheets are especially valuable to business users because users can change data values in specific cells and view the impact of the change on other metrics in the spreadsheet. For example, in a sales reporting situation, a sales manager might want to assess whether improving orders from a specific customer by 25% would be enough to achieve annual sales targets. In this case, they could simply increase the value in the cell that shows a customer’s order total by 25% and see if the cell that shows percentage of target achieved shows a value greater than 100%.

For this type of what-if analysis to work, it’s essential that the cell containing the customer’s order total is referenced in the formula in the cell for overall sales total, and that the cell containing this overall sales total is referenced in the formula for the cell for sales target achievement. Without these interdependent formulas, changing the value of the customer order total by 25% would have no impact on the percentage of target achieved.

In order for an Excel report to support this widely used “what-if” analysis ability, any formulas in the report would need to be added by hand to the Excel report that’s being constructed from the manual data pull. As the number of cells and formulas in a report grows, this task becomes more time-consuming, and it must be repeated every time a new report with up-to-date data is required.

The same manual process of adding functionality to the spreadsheet is required if charts or Pivot Tables are needed in the report, and these charts and Pivot Tables would also need to be re-constructed every time the report is run.

Without many of these useful features like cells with formulas, charts and Pivot Tables, delivering the data in spreadsheet format is of limited value.

## **Distributing Spreadsheet Reports: An Administrative Burden With Security Risks**

### **Email Distribution**

Once the appropriate formatting, formulas, charts and other Excel features have been added to the spreadsheet reports, they are ready for distributing to users. Typically, this entails having an administrator email these spreadsheets as attachments to users or groups that need them.

The person responsible for this task needs to keep track of which individuals need which report, and take precautions not to accidentally send reports containing sensitive data to unauthorized users. As the number of reports, recipients and frequency of reporting increases, this task can balloon into a major time-sink. In addition, organizations that depend on such report distribution processes need to ensure that more than one person knows the process so that there is no interruption in the delivery of regular reports.

### **File or Web Server-Based Storage**

Another alternative to distributing reports is to store them on a central file server or web server and email users a link when new reports are available. This cuts down on email bandwidth by not sending out potentially large reports as attachments but creates other administrative burdens.

In order to organize and secure reports, administrators need to set up an appropriate folder structure to organize the reports, and then set and maintain permissions on folders to prevent unauthorized access to reports. These steps are necessary in any scenario where reports need to be organized and secured – the unnecessary, additional burden is that administrators will need to take care that they put finished reports in the right place in this hierarchy. A mistake could mean that an unauthorized user gets access to data. Since every new report must be manually filed, every new report creates the risk of a security compromise.

## Repetition and Errors: Practice Does Not Always Make Perfect

As noted above, the typical process for creating and distributing spreadsheet reports within an organization is an intensely manual and repetitive process, from pulling the relevant data, to formatting reports, to securing and distributing them.

Not only are these repetitive processes a significant resource drain, consuming multiple person-hours of time on a daily or weekly basis, but they increase the risk of compromised data security or inaccurate data.

The process of manually creating spreadsheet reports has multiple points in which errors can be introduced. For example:

- A SQL query that has a different “where” clause every time a report is run (say for different time periods) may be edited with an incorrect data value
- Formulas that are added back into spreadsheets may reference the wrong cells, causing incorrect totals or calculated values
- Charts that are manually created may reference the wrong data range on a worksheet

The risk of making such errors exists every time the process is repeated, for each version of a report that’s needed.

## Requirements for Improving Efficiency and Minimizing Errors

In order to save time, reduce the risk of errors in reports, and reduce the risk of unauthorized access to information in reports, an automated solution for generating and securely distributing Excel reports is essential.

An automated solution must address the following requirements:

### **Refreshing the spreadsheet with data from the right data sources**

Once data sources and queries have been identified, the solution should support the definition of a query or multiple queries for a report that will consistently execute each time the report is run and automatically refresh the data in the report.

## **Laying out the data across rows, columns, worksheets**

Once the underlying data for a report is ready, the solution should lay out the data automatically within a single worksheet or multiple worksheets. The layout should be consistently applied every time the report is run.

## **Automatically Including Excel Formatting and Calculations**

In addition to laying out the data on the correct rows, columns and worksheets, the solution should also automatically generate the appropriate Excel formatting on the report. Fonts for labels and data values should be of the correct type, size, color and formatting. Cells containing calculated values should contain formulas referencing the correct cells and charts that reference the correct data ranges should be automatically generated. If the report requires Pivot Tables or Outlining, these features should be automatically generated as well.

## **Notifying Users When New Reports Are Available**

The solution should automatically email users when reports with new data are available, either by sending a link to the report, or sending a message with the report attached.

## **Self-Service Report Generation**

The solution should allow users to request, run and view reports in a self-service fashion. Users should also be able to define schedules which specify how frequently they want any given report to be run.

## **Managing Large Numbers of Reports**

The solution should be able to organize and manage multiple reports and multiple versions of each report effectively. As new versions of reports are run, they should be automatically filed in appropriate folders. As the number of reports grows over time, the solution should provide for automatic archival and deletion of reports based on defined policies.

## **Securing Reports**

The solution should support the definition of access rules which specify the users or groups of users that have permissions to access reports in specific folders and permissions to run or view individual reports.

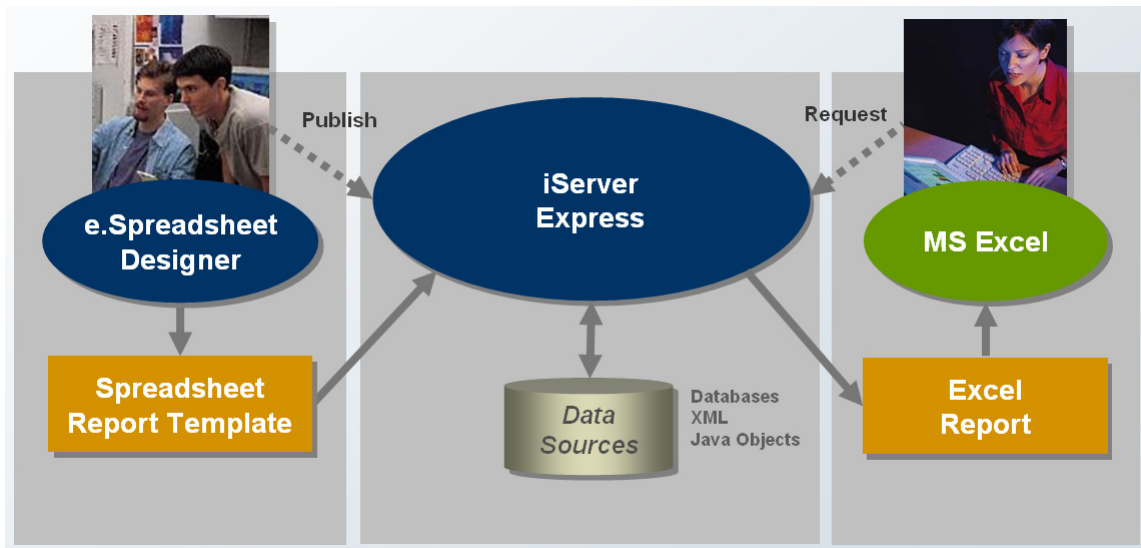
## Using Spreadsheet Report Templates and a Report Server to Automate Excel Reporting

The combination of e.Spreadsheet Designer and iServer Express effectively address the key requirements for generating and securely distributing spreadsheet reports in an automated fashion. Together, they establish a paradigm that supports repeatable and reliable spreadsheet reporting processes.

e.Spreadsheet Designer is a free, easy-to-use report writer for creating authentic Excel files and reports that retain formulas, formatting, live charts and data connections.

Using e.Spreadsheet Designer, IT developers or analysts create a spreadsheet report template for each report. A template specifies queries that return the relevant data sets, how the data should be laid out on a worksheet or across multiple worksheets, and the calculations, charts and other Excel features that need to be applied to the data.

Once these spreadsheet report templates have been created, they are published to a secure, central report server, the iServer Express. iServer Express deploys, manages, schedules, secures, runs and distributes e.Spreadsheet reports. Users can access iServer Express via a web-based interface to browse reports, run them on-demand as well as schedule them to run on a regular basis.



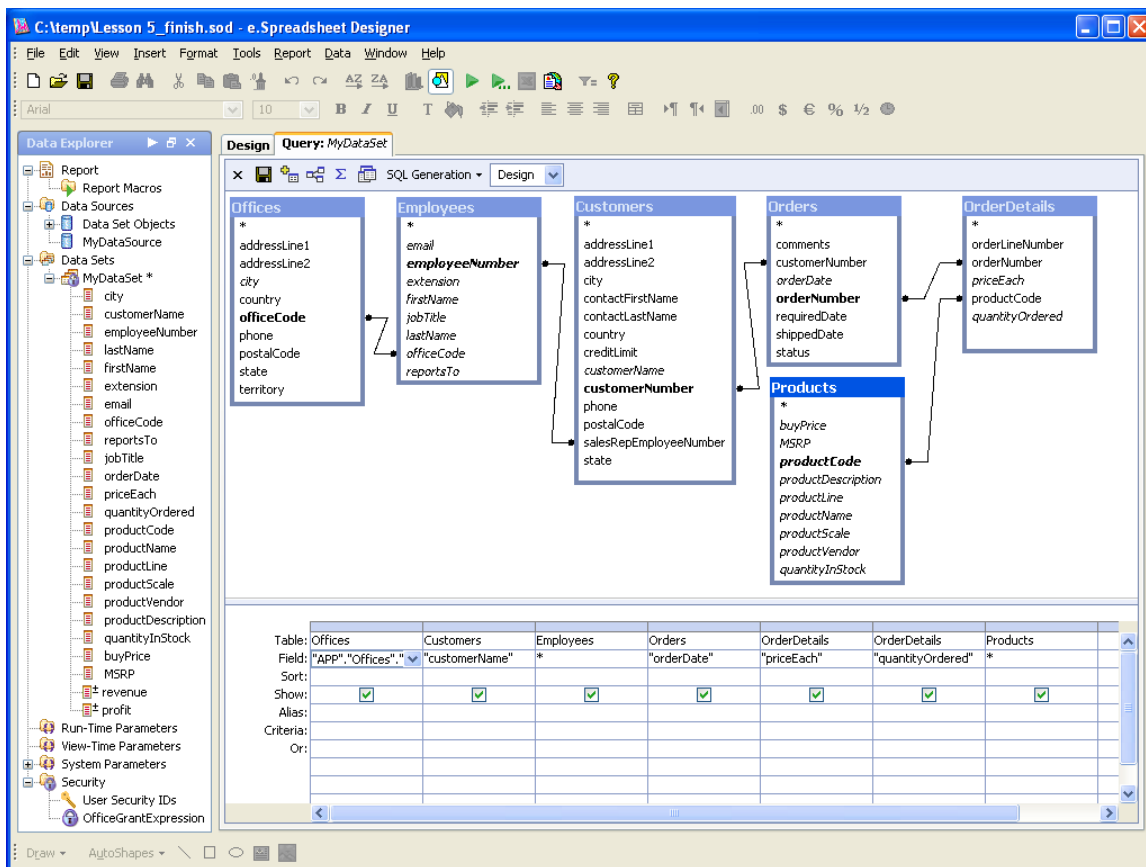


# Creating Templates for Repeatable and Reliable Spreadsheet Reporting

e.Spreadsheet Designer provides an easy-to-use Excel-like environment where IT developers or analysts can define all the elements of a given Excel report: the sources of data, the queries, the layout of the data on the spreadsheet and the Excel features that are needed. These definitions are saved in a spreadsheet report design file that serves as the “blueprint” for generating Excel reports for users.

## Specifying the Source Data

Spreadsheet reports created with the e.Spreadsheet Designer can access a variety of data sources. Out-of-the-box, the tool provides access to databases supporting JDBC, XML documents, Web Services and to flat file data sources. In addition, data from multiple data sources can be combined into one spreadsheet report. Optionally, a query can be defined against the results of other queries to conveniently join multiple data sets.



Multiple wizards within e.Spreadsheet Designer help simplify the specification of data sources. For example, a Data Connection Wizard displays the accessible data sources and guides the creation of new data source connections, while an available Visual Query Wizard guides the building of a query. A Data Preview feature verifies that the data connections are defined correctly and helps determine that the correct tables were chosen for display.

## Specifying the Layout

e.Spreadsheet Designer allows developers and analysts to specify how the data should be laid out within a spreadsheet. The layout definitions automatically take into account that the queries for a report may return different volumes of data each time the report is run. e.Spreadsheet Designer will automatically adjust the layout as dictated by the data.

For example, columns and rows may expand across the page with outlining. Data can dynamically burst down rows, across columns and across worksheets within a report.

e.Spreadsheet Designer also automatically groups data into dates; for example, days, weeks, months and quarters.

## Specifying Excel Formulas, Charts and Special Formatting

Using e.Spreadsheet Designer, a developer or analyst can define cells containing calculations and totals that use dynamic cell references. If the data in the report is burst across multiple spreadsheets, it is still possible to reference data across different sheets. This supports the effective creation of reports with summary and detail worksheets, with a summary worksheet that subtotals the values on other worksheets.

The screenshot displays the e.Spreadsheet Designer interface. The main window shows a spreadsheet with the following data:

|    | A                              | B                                 | C                | D             | E | F | G | H | I |
|----|--------------------------------|-----------------------------------|------------------|---------------|---|---|---|---|---|
| 1  | Company-Wide Metrics           |                                   |                  |               |   |   |   |   |   |
| 2  |                                |                                   |                  |               |   |   |   |   |   |
| 3  |                                |                                   |                  |               |   |   |   |   |   |
| 4  | Company-Wide Customer Averages |                                   |                  |               |   |   |   |   |   |
| 5  |                                | Profit                            | Revenue          | Profit Margin |   |   |   |   |   |
| 6  | All Customers                  | $\#sum(Profit)$                   | $\#sum(Revenue)$ | $=B6/C6$      |   |   |   |   |   |
| 7  | Average per Customer           | $=B6/numcust$                     | $=C6/numcust$    |               |   |   |   |   |   |
| 8  |                                |                                   |                  |               |   |   |   |   |   |
| 9  | Number Customers               | $\#count(distinct(customerName))$ |                  |               |   |   |   |   |   |
| 10 |                                |                                   |                  |               |   |   |   |   |   |
| 11 | Company-Wide Thresholds        |                                   |                  |               |   |   |   |   |   |
| 12 |                                | Profit                            | Revenue          | Profit Margin |   |   |   |   |   |
| 13 | low threshold                  | $=B7/2$                           | $=C7/2$          | 38.00%        |   |   |   |   |   |
| 14 | high threshold                 | $=B7*2$                           | $=C7*2$          | 42.00%        |   |   |   |   |   |
| 15 |                                |                                   |                  |               |   |   |   |   |   |
| 16 |                                |                                   |                  |               |   |   |   |   |   |
| 17 |                                |                                   |                  |               |   |   |   |   |   |
| 18 |                                |                                   |                  |               |   |   |   |   |   |

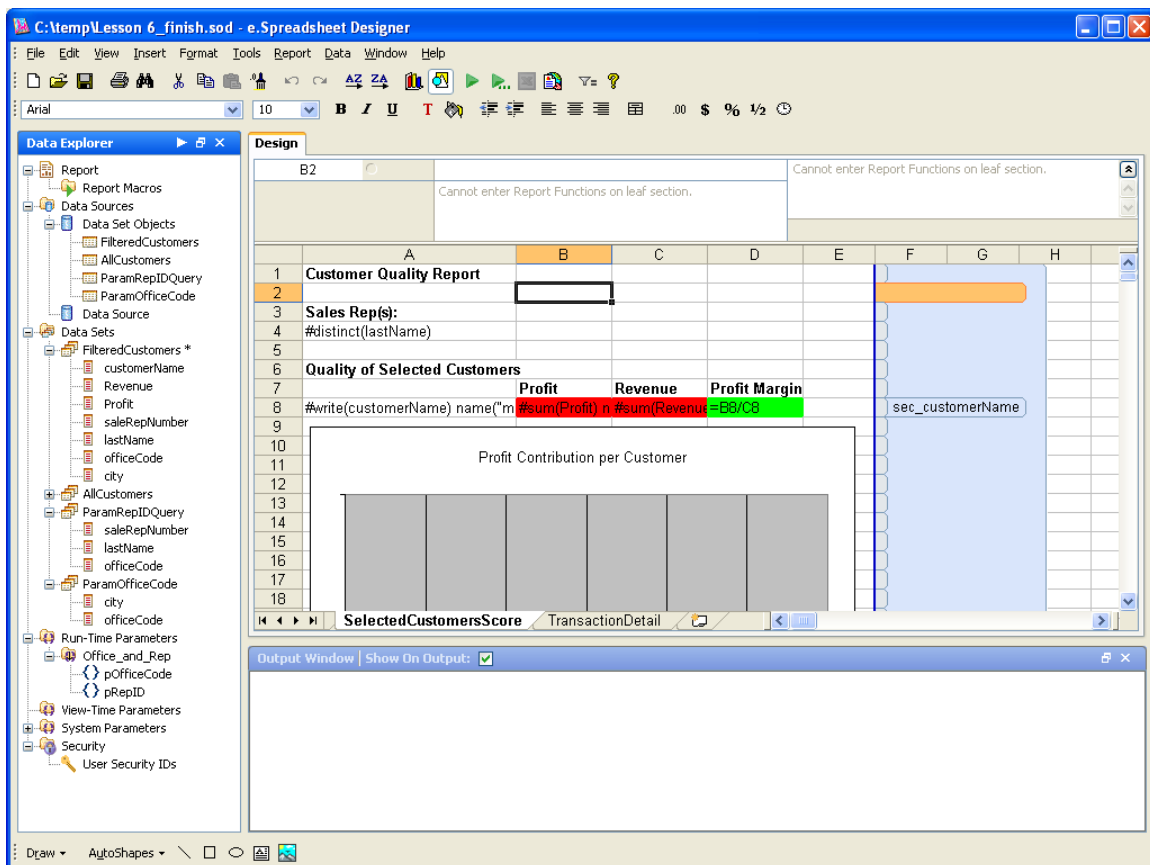
The interface includes a Data Explorer on the left, a Design View at the top, and an Output Window at the bottom. The spreadsheet shows formulas for calculating averages and thresholds based on profit and revenue data.

The dynamic reference to cell values in formulas permits the display of the appropriate totals and calculated values in a spreadsheet report. More significantly, it supports effective “what-if” analysis for users viewing the spreadsheet. Changing any of the values in cells that are referenced in formulas will change calculated values.

Similarly, charts are also defined based on dynamic cell references, and changing values of cells that are referenced in a chart will change the appearance of the chart.

e.Spreadsheet Designer supports a variety of 2D and 3D Excel charts including column, line, area, step, bar, combination, pie, doughnut, XY (scatter), stock and bubble.

In addition to formulas and charts, e.Spreadsheet Designer enables the use of other Excel formatting features. For example, it supports defining reports that use all the Excel fonts, types and sheet formatting options for cells, rows, columns and headers – including borders and merged cells. Excel reports with the needed data sorting and automatic data filtering options can be defined. e.Spreadsheet Designer can also generate spreadsheets using the Excel outlining feature with row and column groupings, so details can be shown or hidden as needed. Finally, e.Spreadsheet Designer can display data in the form of Excel Pivot Tables, so that users can easily sort, sum and flip data on their own.



With e.Spreadsheet Designer’s extensive support for Excel formatting features, it’s possible to generate Excel reports that contain all the rich features that users expect from an Excel spreadsheet in a reliable and repeated manner.

## Server-based Generation, Management and Distribution of Excel Reports

Spreadsheet report templates created with the e.Spreadsheet Designer are the “blueprints” for generating Excel reports: specifying the data, layout and formatting for a finished report. The iServer Express is the server that securely stores these “blueprints” and enables users and administrators to use them as the basis for generating fully functional Excel spreadsheets.

iServer Express incorporates a spreadsheet reporting engine that:

- Reads a given spreadsheet report template
- Connects to the data sources specified in the template and executes queries
- Generates an Excel spreadsheet that contains the relevant formatted data, complete with formulas, functions and charts

The combination of the spreadsheet report templates created with the e.Spreadsheet Designer and the automated spreadsheet reporting engine in the iServer Express provides degrees of consistency and efficiency for spreadsheet report generation that is simply not possible with manual processes. The consistency helps reduce errors in spreadsheet reports and the efficiency means that hours of manual labor in creating reports can be eliminated.

In addition to providing an automated solution for generating spreadsheet reports, the iServer Express supports the efficient management, securing and distribution of reports.

### Organizing Templates and Reports

Most organizations typically have multiple spreadsheet reports that are used to support the analysis needs of users. As the number of reports grows, facilities to effectively organize and manage these reports become more important.

iServer Express enables administrators to set up a hierarchical folder structure to help organize both the spreadsheet report templates that are the basis for generating reports and finished spreadsheet reports themselves. Administrators can also specify archiving and versioning policies for reports to help keep a manageable number of reports available for online access and archive the necessary reports for safe keeping.

### Securing Spreadsheets

The iServer Express offers hierarchal role-based security with multi-level access permissions which, when used together, provide multiple approaches for granular control over user access to reports, down to the level of individual sheets, rows or columns within a spreadsheet report.

## Securing Spreadsheets: Users and Roles to Manage Access

While the iServer Express can be configured to allow anonymous access to reports, most applications will take advantage of the server's built-in security mechanism. Access to reports can be user-based, where access levels are determined by unique privileges assigned to an individual, or role-based where access levels are determined by the functional or hierarchical role a person has in the organization, such as their specific job or management level.

iServer Express security can be integrated with external directory services such as LDAP and Active Directory in order to leverage existing security directories. Security can be integrated such that user names and roles are stored in iServer Express, with invisible log-in to iServer Express handled by external single sign-on, or so that user names and roles are not stored on iServer Express and authorization and authentication are both handled by the external directory.

## Securing Spreadsheets: Fine Grained Control over Access

iServer Express allows administrators to specify granular access to reports, functions and data for any given user or role. Permissions can specify which reports users are able to access and they can also determine whether users can view reports, generate them, or both.

QGSVolume > Employees (Folder) : Properties

The screenshot shows the 'Privileges' tab for the 'Employees' folder. The 'Share' option is selected. The 'Available' list includes roles like 'Employee' and 'Customer', and users like 'DianeMoore' and 'Sue Jackson'. The 'Selected' list shows 'All' and several users with their permissions. The 'Filter' section is set to 'Roles'. The 'Apply these privilege settings to the contents of the folder' checkbox is checked.

| Selected:   | Visible (V)                         | Secure Read (S)                     | Read (R)                            | Execute (E)                         | Write (W)                           | Delete (D)                          | Grant (G)                           |
|-------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| All         | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            |
| DianeMoore  | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/>            |
| Sue Jackson | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            |
| Jane Yancy  | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/>            |
| MaryQuinn   | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |

iServer Express also supports the ability to control end user access to individual rows, columns and cells when used in conjunction with the SmartSheet Security option.

# Scheduling Spreadsheet Report Execution

Spreadsheet reports are commonly used to monitor metrics that need to be reviewed on a regular basis. In order to support these applications, the iServer Express allows administrators to specify schedules that determine when a given report is executed. Schedules can be calendar-based, event-based, where report execution is triggered by events occurring within the report server or in external applications, or business rules-based, where report execution is triggered when a combination of data thresholds is met or when a specific period ends.

QGSVolume > Customers > DanMorris > BudgetRiskReport (RPTDESIGN) (Version 1) : Schedule x

**Schedule** Parameters Output Privileges Channels Notification

Job name: BudgetRiskReport \*

TimeZone: GMT-8:00 Pacific Standard Time

Run job:

- Right now
- Once: date [ ] # time [ ] (M/d/yyyy h:mm a)
- Recurring: Last day of every month [v] time 11:59 PM (h:mm a)
- Advanced: Edit Schedule...
- Wait for event: Job Event [v] Event name: Expense Report  
Job name: [ ] Select Job Lag time (min): 60  
 Wait for success OR  Wait for failure

Priority:  Low (200)  Medium (500)  Other (1 - 500): 500 (1)

Executable version:  Always use latest version of BudgetRiskReport.rptdesign  
 Always use version 1 of BudgetRiskReport.rptdesign

Retry failed jobs:  Use volume default  
 Retry [ ] times; wait [ ] hours [ ] minutes between attempts  
 Do not retry

(1) This job will use the lower priority of this setting and the one assigned to you in your user profile.

OK Cancel

## Distributing Spreadsheet Reports to Users

Administrators can configure iServer Express to automatically send emails to specific users when a report is ready to be viewed. The report can also be attached to the email.

[QGSVolume](#) > [Customers](#) > [DanMorris](#) > [BudgetRiskReport \(RPTDESIGN\) \(Version 1\)](#) : Schedule

The screenshot shows the 'Notification' tab of a configuration page. At the top, there are tabs for 'Schedule', 'Parameters', 'Output', 'Privileges', 'Channels', and 'Notification'. The 'Notification' tab is active. Below the tabs, there are several checkboxes and options:

- Override user preferences (For all notified users)
- If job succeeds:
  - Send e-mail notification
  - Attach document
  - Create completion notices in user's personal channel
- If job fails:
  - Send e-mail notification
  - Create completion notices in user's personal channel

Below these options is a dropdown menu labeled 'Format for attached report:' with a downward arrow. At the bottom, there are two lists: 'Available:' and 'Selected:'. The 'Available:' list contains 'Customer', 'Employee', 'Intranet', 'Sales', and 'Vendor'. The 'Selected:' list contains 'Sales', 'MaryQuinn', 'BillAbner', and 'DianeMoore'. Between the two lists are two buttons: a right-pointing arrow and a left-pointing arrow.

In addition to this “push” notification via email, iServer Express presents a web-based interface to users through which they can browse reports and report templates, run them on-demand and view them. In this way, users can meet their reporting requirements in a self-service fashion, without requesting that an administrator run a report and mail it to them.

When running reports on-demand, the iServer Express can automatically present users with dynamically generated parameter pages to help users create reports for specific data ranges. Based on the report that a given user selects, users are prompted to specify the appropriate parameters for a report. Reports may have dynamic cascading prompts, so that different lists of values for a parameter are presented to users based on the selection they’ve made for the values of other parameters.

Finally, users can also specify their own calendar-based schedules for running reports, providing yet another degree of self-service access.

## Additional Resources

- [\*e.Spreadsheet Designer Information\*](#)
- [\*iServer Express Information\*](#)
- [\*Free Download of e.Spreadsheet Designer\*](#)
- [\*Free 90 Day Trial Download of iServer Express for Windows\*](#)
- [\*Free 90 Day Trial Download of iServer Express for Linux\*](#)