**Case Study:** Calling SSRS Report with Parameters in SSIS Packages (On demand/Schedule)

Consider a scenario where the organization needs is generate a large volume of parameterized reports and email the reports to different users. The reports need to be generated and emailed on-demand. As this information is generated every month, the reports should contain only data that is related to the details of the month for that user.

Using SSRS alone to generate and email these reports has limitations. A data-driven subscription is required to email the reports to the users using E-Mail Delivery extension. The data-driven subscription requires a fixed schedule that triggers the delivery of the reports, in contrast to the requirement where report generation and delivery needs to be done on demand.

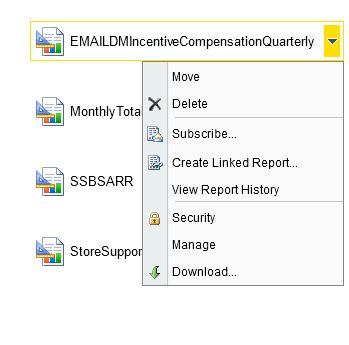
Using the SSRS and SSIS solution, a subscription and schedule are not required. The user can generate and email the parameterized reports on-demand.

**Data Driven Subscription in SSRS:**

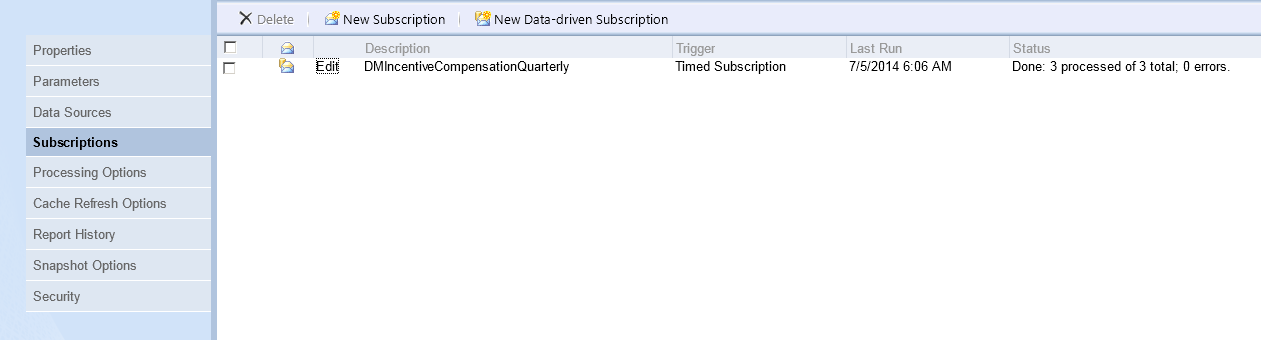
When you setup a subscription for the report , report server processes the subscription, generates the reports and delivers it to the intended audience via email delivery or file share delivery method on the defined schedule.

Getting Started with Report Subscription

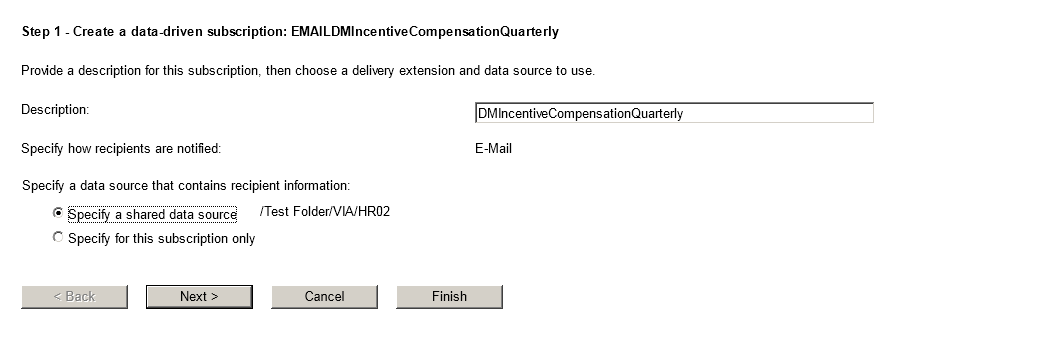
To setup data-driven subscription for a report, go to Report Manager, navigate to the folder that contains the report, point to the report for which you want to set up a subscription, click on the down arrow and click on the Manage menu item as shown below:



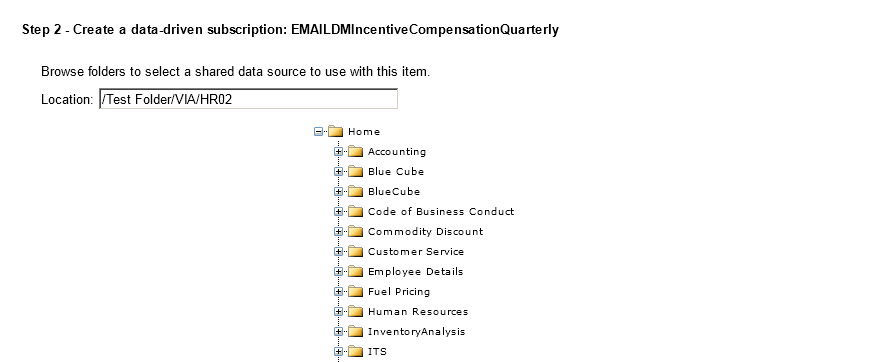
On the manage report page, click on the Subscription option and then click on the New Data-driven Subscription button as shown below:



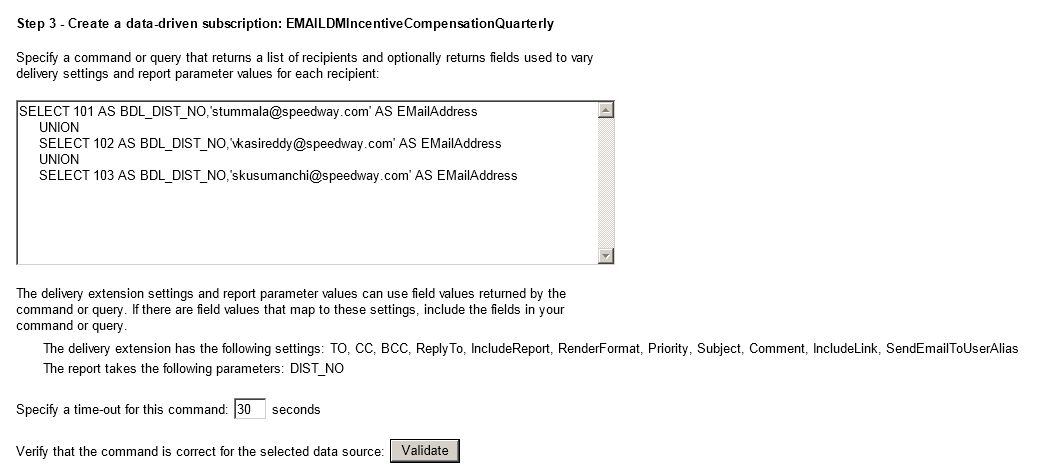
Clicking on the New Data-driven Subscription will launch a wizard to create the data-driven subscription. On the first step of the wizard, you need to provide a description for the subscription, delivery mode and the data source to use to get information about the subscriber and its parameter values:



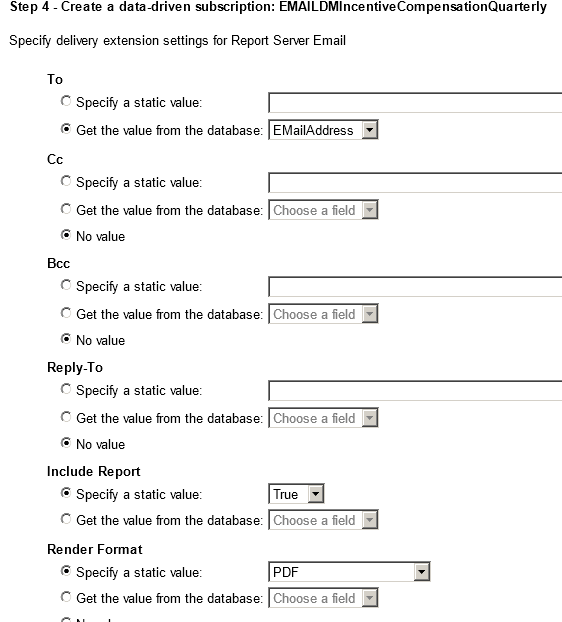
On the second step of the wizard, you need to specify the data source details. You can either use a predefined share data source or create a new one:



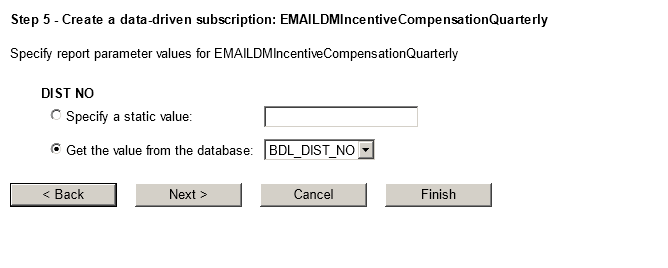
On the third step of the wizard, you need to specify the query to fetch the list of recipients, delivery settings and parameter values, if any, from the data source chosen in the last step. You can click on the Validate button to get your query validated by data source:



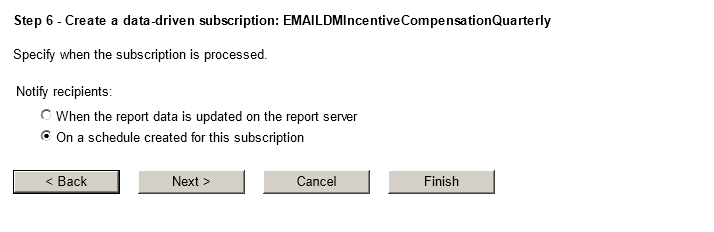
On the fourth step of the wizard, you need to specify delivery extension settings for the delivery mode you chose on the second step. You can specify different values for report execution, which will either come from the database or be static values; for example in the case of file share more, you can specify path, file name, extension, write mode and user name/password (either to come from database or to specify the static value) to be used by report server to put the file at the specified location:



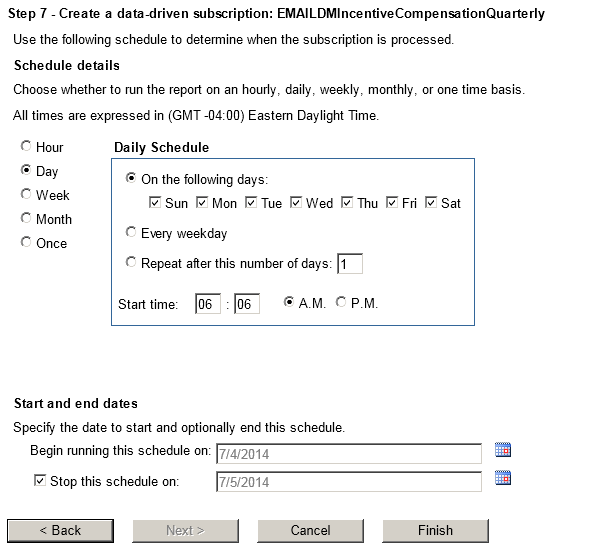
On the fifth step of the wizard, you need to specify report parameter values. You can specify different values for report parameters, which will either come from the database or use default of the parameter or use static values; for example in screenshot below, you can see I am taking the value for DIST NO from the database:



On the sixth step of the wizard, you need to specify when the subscription will be processed. You can choose any of these three options as shown below:



If you chose "On a schedule created for this subscription" option on the last step, then on the step 7 you need to define the schedule for subscription processing.



**Solution Approach:**

To develop a solution prototype, we need two artifacts for our solution. First, we need a SSRS report that displays data based on some parameter(s) and is used as a model to generate different reports. Second, we need to create a SSIS package that generates the reports, saves reports in multiple formats, and emails the reports using the Send Mail task.

SSRS 2008 R2 provides access to the full functionality of the report server through the Report Server Web service. The Web service provides two endpoints, for native mode and SharePoint integrated mode. In native mode, we use the ReportExecution2005 endpoint and the ReportService2005 endpoint.

The ReportExecution2005 endpoint contains features that facilitate report processing and rendering, and the ReportService2005 endpoint manages objects on a report server. Our intention and requirement is to just execute any report that would be deployed on report server, and save that report in the format of choice. So we only need to use the ReportExecution2005 endpoint.

**Part I: Parameterized Report Development**

Create a parameterized report that is used as a model to generate different report outputs based on the “Continent” parameter, from SSIS.

1. Create a parameterized report in SSRS
2. Deploy the report into report server

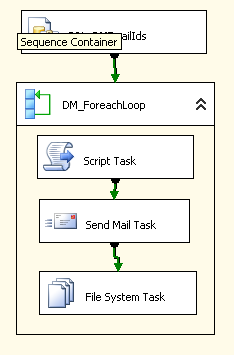
Part II: SSIS Package Development

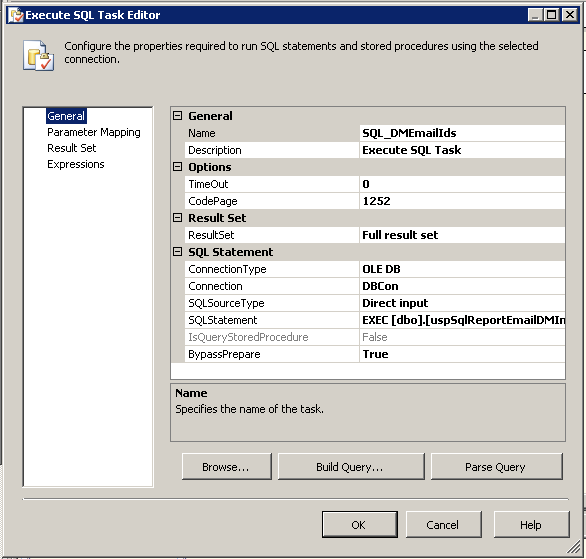
Automate the SSRS Reports email subscription through SSIS.

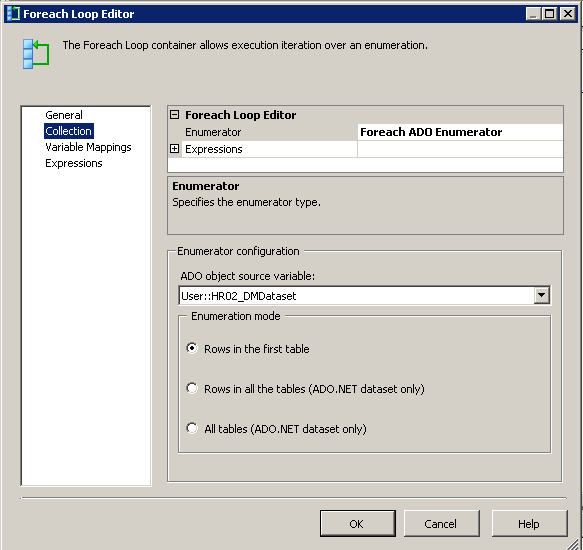
You can use SSRS subscription to send the report.

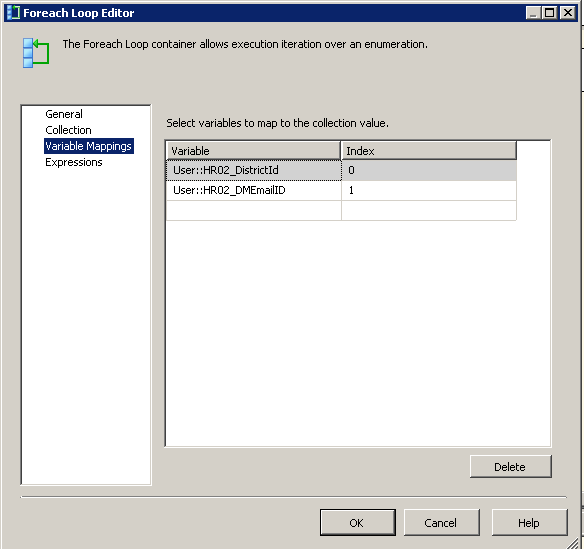
If you really want the SSIS to send the report. You can do the following steps,

1. Create the SSIS package using BIDs.
2. Add Execute SQL Task
3. Add Foreach Loop Container.
4. Drag Script task, Send Mail Task and File System Task into the Container.









Add <http://localhost/ReportServer/reportexecution2005.asmx> as Web Reference and name it rs.

we only make a call to the endpoint to execute the report after setting the parameter value.

ReportExecutionService rs = new ReportExecutionService();

Byte[] results;

string encoding = string.Empty;

string mimetype = string.Empty;

string extension = string.Empty;

Warning[] warnings = null;

string[] streamId = null;

string deviceInfo = null;

string ssrsConnection = “http://localhost/ReportServer/reportexecution2005.asmx”

[Location of your report services];

rs.Credentials = System.Net.CredentialCache.DefaultCredentials;

rs.Url = ssrsConnection;

try

{

var reportpath = string.Format("/{0}/{1}", ssrsFolderName, ssrsReportName);

rs.LoadReport(reportpath, null);

//Adding Parameters

ParameterValue[] paramValues = new ParameterValue[2];

ParameterValue paramValue = new ParameterValue();

paramValue.Name = "ReportParamName";

paramValue.Value = "Value";

paramValues[0] = paramValue;

rs.SetExecutionParameters(paramValues, "en-US");

results = rs.Render(“PDF”, deviceInfo, out extension, out mimetype, out encoding, out warnings, out streamId);

string path = string.Format(@"{0}\{1}.{2}", “Location to write the file”, “filename”, “pdf”);

using (FileStream stream = File.OpenWrite(path))

{

stream.Write(results, 0, results.Length);

}

}

catch (Exception ex)

{

MessageBox.Show(ex.StackTrace);

}