

**SAVANCE**  
EIOBOARD

## **SQL Report Builder**

**Savance**

**Phone: 248-478-2555 | Fax: 248-478-3270**

**[www.eioboard.com](http://www.eioboard.com) | [support@eioboard.com](mailto:support@eioboard.com) | [www.savance.com](http://www.savance.com)**

© 2015



## Table of Contents

<b>Overview</b>	<b>3</b>
<b>Start a New Report</b>	<b>3</b>
<b>Run or Export a Report</b>	<b>9</b>
<b>Editing a Report</b>	<b>10</b>
1 Editing Columns.....	10
2 Filters.....	11
Dynamic Option List .....	14
3 Displaying Filters and Other Fields.....	15
<b>Sample Report Query</b>	<b>15</b>

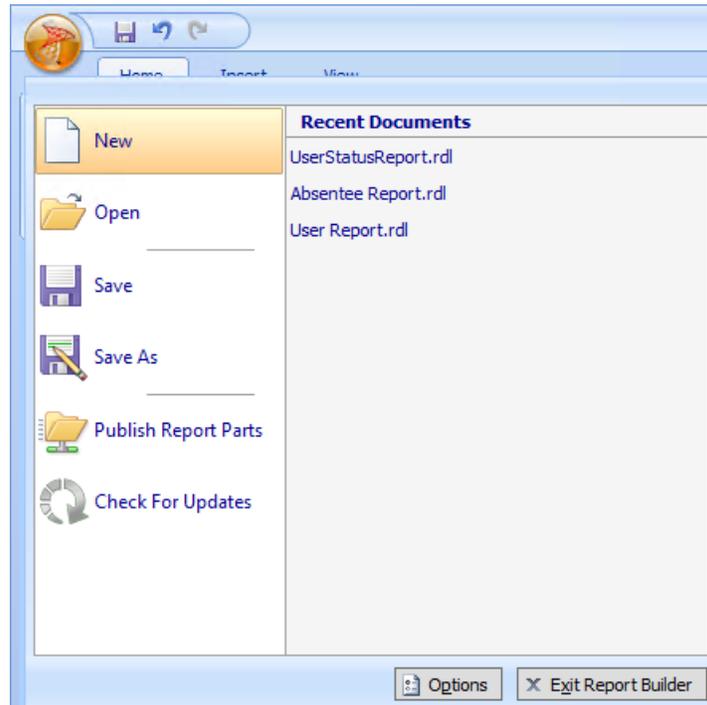


## Overview

This document will cover how to use Microsoft SQL Server Report Builder in order to create custom reports for EIOBoard. By default, EIOBoard includes some built-in reports (simply click on **Reports** from the main toolbar). However, SQL Report Builder allows you to create reports that are specific to your organization. Please note that basic prior knowledge of SQL tables and queries is required in order to use SQL Report Builder.

## Start a New Report

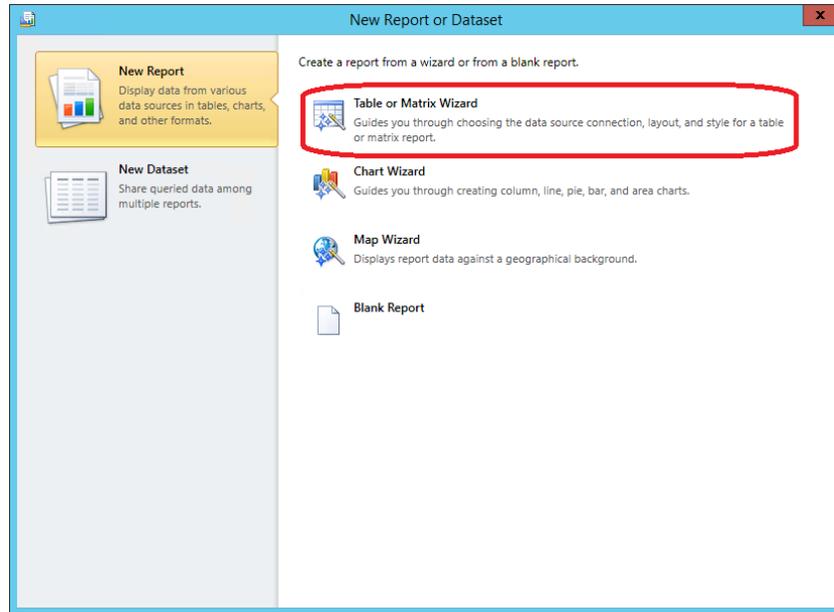
To get started, open up SQL Report Builder and start a new report. You can start a new report by clicking on the menu orb (at the top left of the screen) and clicking **New**.



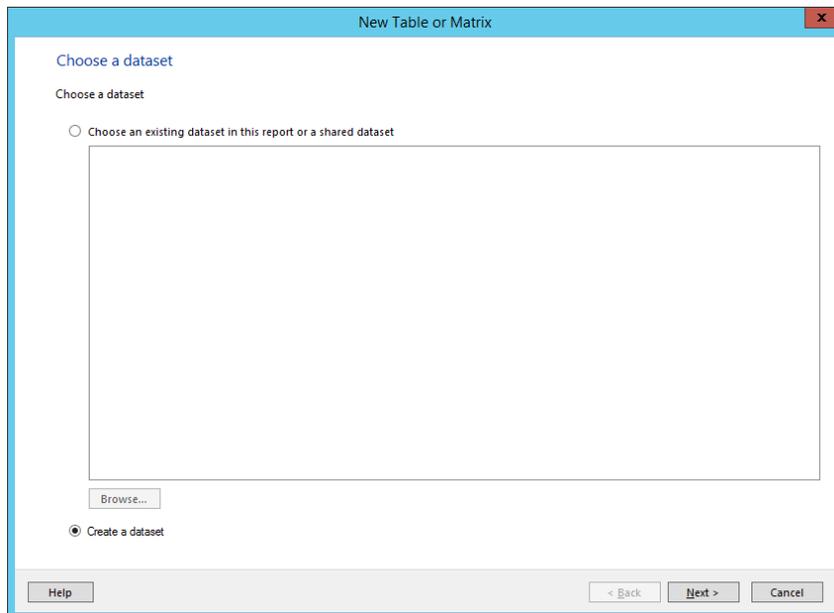
First of all, select **New Report** and then click **Table or Matrix Wizard**, as seen below.



## SQL Report Builder



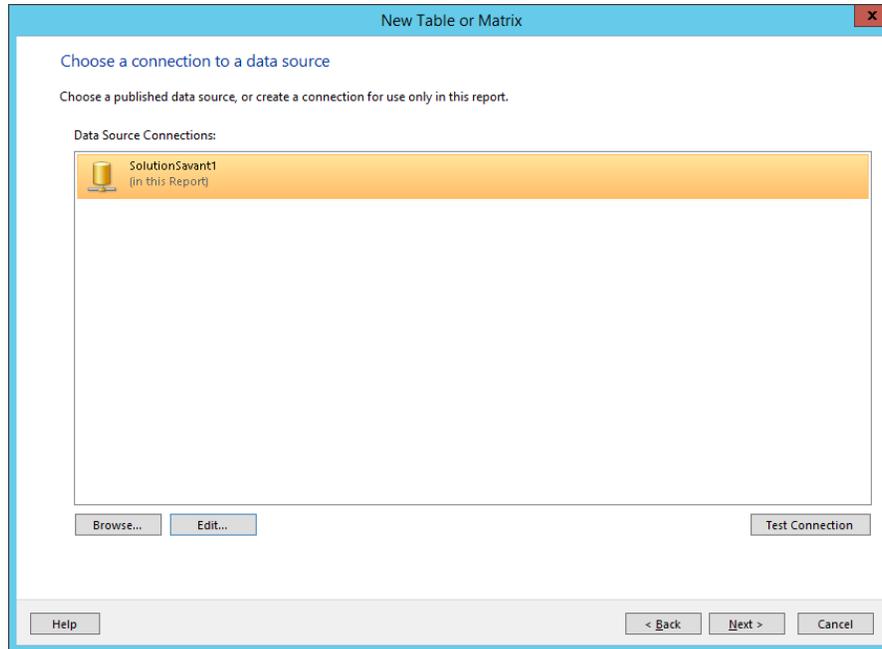
The **New Table or Matrix** wizard will appear and walk you through the creation of your report. First of all, choose **Create a dataset** (we will soon create a dataset that connects to the EIOBoard database).



Next you can choose the datasource, which determines the connection to the database. If you have done this before, you will see your data source listed in the connections. Click on it and click next. If you need to add your EIOBoard database, click **New**.



## SQL Report Builder



If you are adding a new database connection, the following window will come up. In the **General** tab, you will need to give your connection a name. Make sure the connection type is *Microsoft SQL Server*. Then you can add your connection string if you already have one, or you can use the following template:

*Data Source=[ServerName];Initial Catalog=[DatabaseName]*

In the **Credentials** tab, you can either save the username and password for your database, or you can prompt for credentials each time a connection to the database is made (ie whenever the report is run).

Once you are done, you can click **Test Connection** in the **General** tab in order to make sure you typed in everything correctly. Then you can click **OK** to save the connection.



## SQL Report Builder

**Data Source Properties**

Change name, type, and connection options.

**General**

Name:

Use a shared connection or report model  
 Use a connection embedded in my report

Select connection type:

Connection string:

Use single transaction when processing the queries

**Data Source Properties**

Change the credentials used to connect to the data source.

**General**

**Credentials**

**Warning:** This information is only stored when you save the report to a report server.

Use current Windows user. Kerberos delegation might be required.  
 Use this user name and password

User name:

Password:

Use as Windows credentials  
 Impersonate the authenticated user after connecting

Prompt for credentials

Enter prompt text:

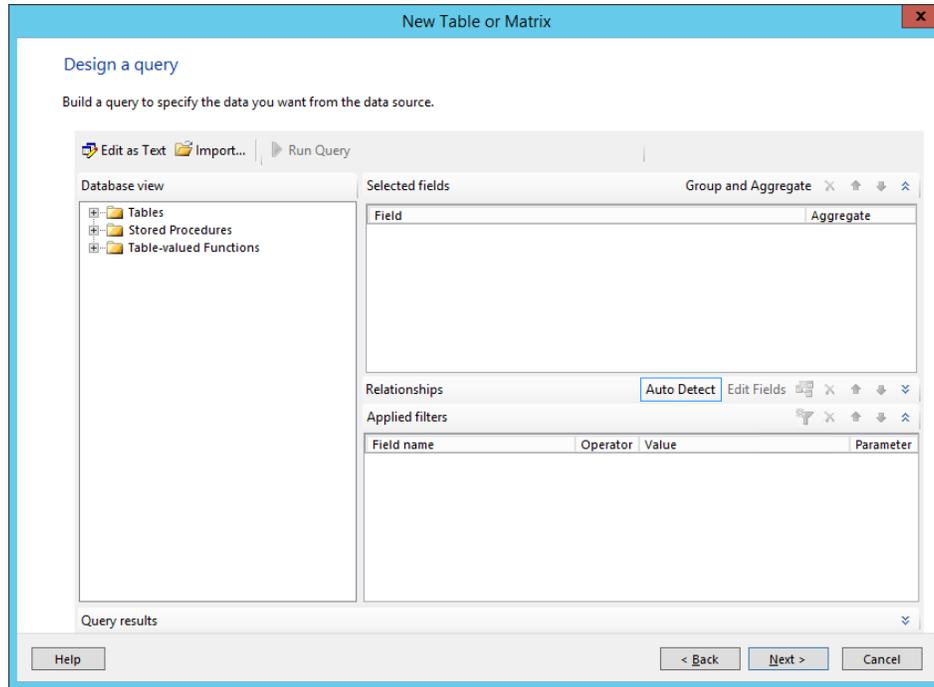
Use as Windows credentials

Do not use credentials

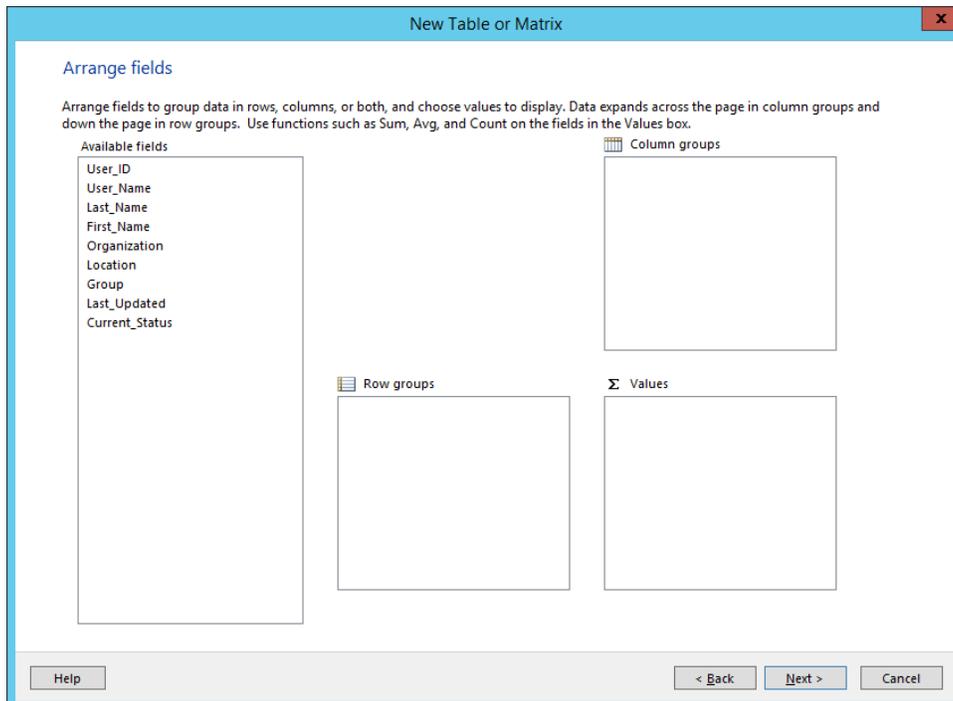
The next window that comes up allows you to choose what data to pull from the database. You can pull values directly from a table, you can run a stored procedure that exists in the database, or you can create your own query (by clicking **Edit as Text**).



## SQL Report Builder



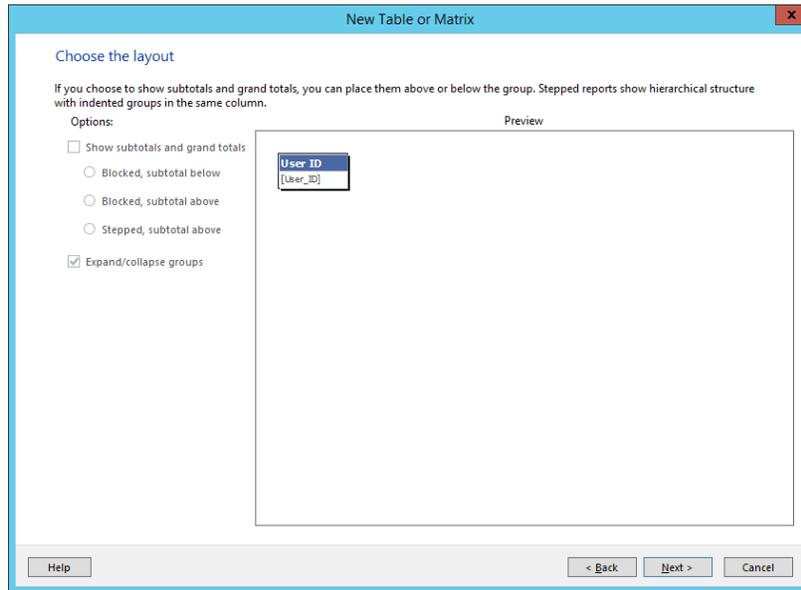
After selecting where to pull the data from, you will have a list of available fields (columns) that will be returned. Any of these fields you wish to show on the report should be dragged and dropped into the **Values** box in the order you would like them to appear. This can be changed later if necessary from the report designer.



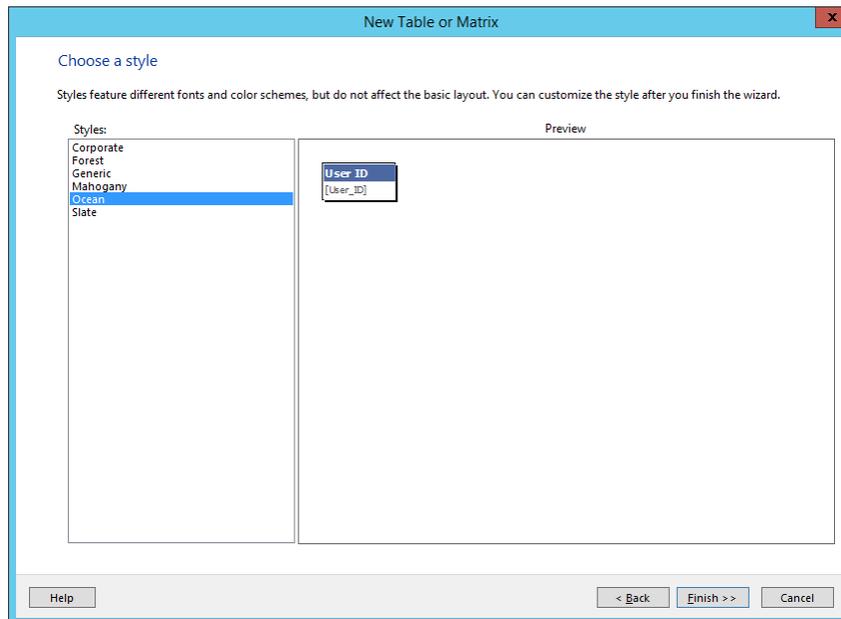


## SQL Report Builder

The next window simply shows you what columns you selected to show and how they are organized. Nothing needs to be done here. Again, the organization of the fields can be changed later.



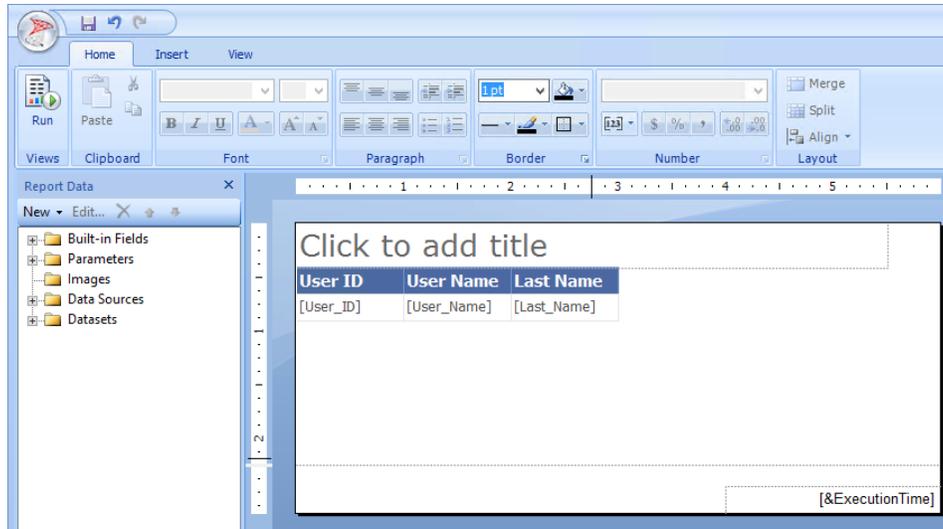
Lastly, you can choose a style for the report. The default is fine, you can pick a new style if you wish. This does not affect the data that is shown in the report.



Click Finish to finalize all the initial settings for the report! A window similar to the one below will come up, showing the columns you chose. It is recommended that you save shortly after the creation of your report in order to not lose all your hard work!



## SQL Report Builder

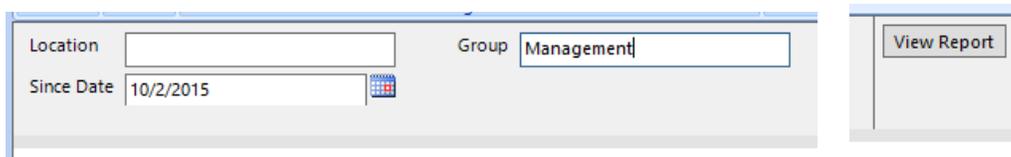


### Run or Export a Report

To run your report, simply click the **Run** button on the top-left of the **Home** tab.



If there are any filters, you can specify those parameters and then click **View Report**.



Once you run the report, you can see the results below the parameters.

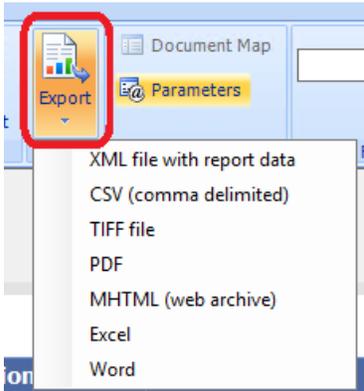
User Report								
User ID	User Name	Last Name	First Name	Organization	Location	Group	Last Updated	Current Status
100000	SavTest	SavTest	SavTest	Savance Intranet Test	*** No Location ***	*** No Group ***	Aug 31 2015 11:45AM	Lunch
100001	Test	Test	Test	Savance Intranet Test	*** No Location ***	*** No Group ***	Sep 20 2015 4:44PM	Unavailable

9/20/2015 4:45:34 PM



## SQL Report Builder

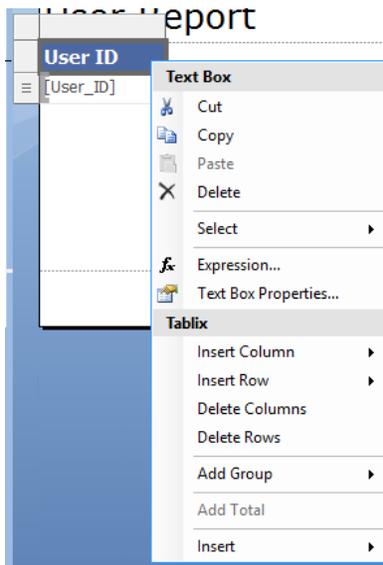
If you would like to export your report to CSV, Excel, PDF, etc, you can do so by clicking **Export** on the top of the page. You can then choose a location to save the exported report!



## Editing a Report

### 4.1 Editing Columns

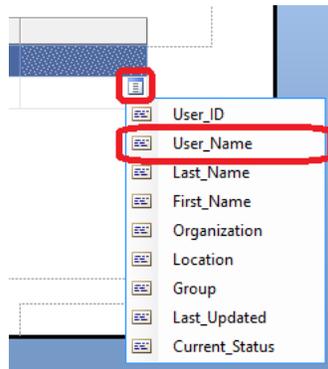
You can add, remove, and rename columns by right clicking on them, as seen below. For example, if you want to add a column, you can click on **Insert Column**.





## SQL Report Builder

If you are inserting a column, you can specify which database field it is bound to by clicking on the data icon (as circle below) and then specifying which database field you would like to show up in this column.



## 4.2 Filters

A filter in a report is a query parameter that the user can specify upon running the report. For example, perhaps I only want to run the report for a single group of users rather than all users. This feature allows you to do this.

In order for you to have a filter, you must specify the parameter inside your query, preceded by the '@' symbol. For example, let's say we have created a report that returns all users and the groups they are in. In this case, *@Group* might be used to show only users within the selected group. So if you wanted to allow the user to filter on groups, you could add this line to the WHERE clause of your SQL Select statement:

*AND Contacts.GroupList LIKE @Group*

When the report is run, only users within this selected group will show up. Of course, you can add any number of parameters that you wish.

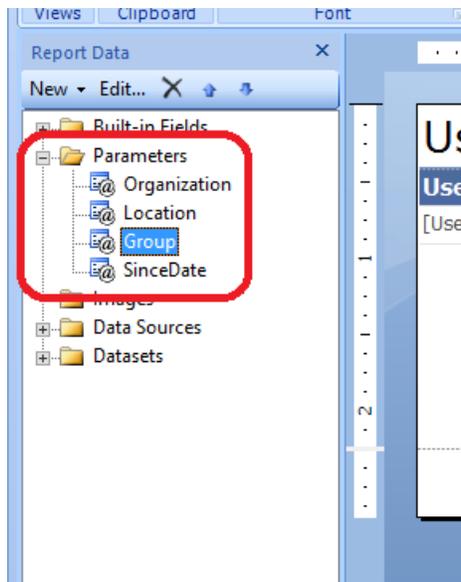
A regular filter allows the user to manually type in a value. For example, if you are filtering on groups, the user can manually type in "Management" for all users in the Management group to show up.

On the other hand, a dynamic option filter is a drop down menu that changes based on what's in your database. Again, let's say we are filtering on groups. In this case, you may want to create a drop down filter for each of the possible groups. But what happens when someone adds a new group in EIOBoard? A dynamic option list will check all the current groups in the database and add them to the list of filters, even when someone adds, removes, or renames a list in EIOBoard.

To create a new filter, double-click on one of the parameters in the **Parameters** folder on the left.



## SQL Report Builder



This will bring up the following window. In the **General** tab, you can specify the name, prompt (what will be displayed to the user), and the Data Type. Here is a brief description of each data type, which determines what :

- **Text:** allows the user to type in any value (such as if the user types in "Management" to search for users in the Management group)
- **Boolean:** 1 or 0 (this is equivalent to a check box, which is either checked or unchecked)
- **Date / Time:** a date value and/or a time value (this is good for date fields, like when the user last updated their status)
- **Integer:** a non-decimal number
- **Float:** a decimal number

Depending on which of these you pick, you will have a couple extra options below it:

- **Allow blank value (""):** allows the user to pass in a blank value (ie not typing anything into a search textbox)
- **Allow null value:** allows the user to check a box labeled NULL to pass in a NULL value for this parameter
- **Allow multiple values:** allows the user to pass in more than one value (not necessary for what we are discussing)



## SQL Report Builder

Report Parameter Properties

Change name, data type, and other options.

**General**

Available Values

Default Values

Advanced

Name: Group

Prompt: Group

Data type: Text

Allow blank value ("")

Allow null value

Allow multiple values

Select parameter visibility:

Visible

Hidden

Internal

Help OK Cancel

In the **Available Values** tab, you can determine what values the user can specify. Below are the possible options:

- **None:** the user can specify anything that matches the data type; in the case of the Text data type, this is the equivalent of having a search text box
- **Specify values:** this allows you to create static values for a drop down box
- **Get values from a query:** this allows you to create a dynamic options list (see next section for details)

Once you have finished, click OK. Now if you run the query, you can see all the filters you created!

Location   NULL

Group Management   NULL

Since Date 9/30/2015

Please note that if you have any hidden fields, you will have to specify a default for the field (in the case that it is required by the query or SP). You can also specify a default for any visible fields, such as setting a default date.



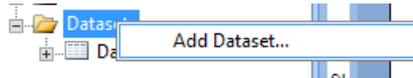
## SQL Report Builder

### 4.2.1 Dynamic Option List

If you choose "Get values from a query" when choosing the Available Values for a filter, you can create a dynamic option list. You can then choose the following values:

- **Dataset:** This is the query you are grabbing the information from. This will be described in further detail in a moment.
- **Value field:** This is the value that will be searched on, such as the database ID value of a Group
- **Label field:** This is the value that will show up for the user (not necessarily what is being searched on), such as the display name of a Group

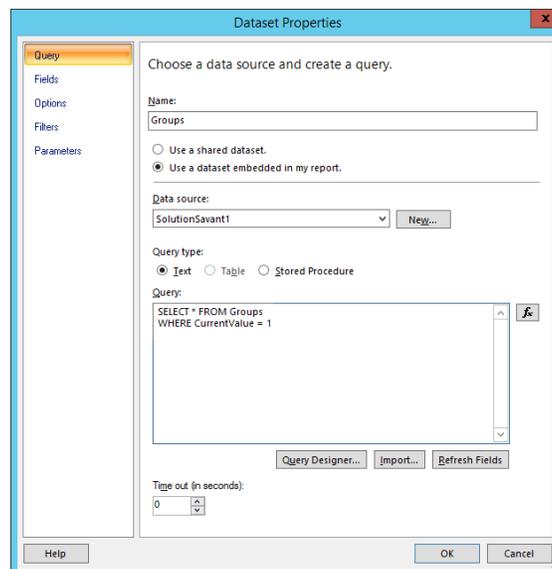
If you have not already done so, you will need to first create the dataset you are searching. To do so, right-click on the Datasets folder and click **Add Dataset...**



The following window will come up. First give the data set a name, and then select *Use a dataset embedded in my report*. Change the data source to the one connected to your EIOBoard database. Then you have the option to choose *Text* (a manually typed query) or *Stored Procedure*. In our case, we will most often use text. You can usually use the following syntax to get all active values from a table (check each table to be sure before writing the query):

```
SELECT * FROM [TableName]  
WHERE CurrentValue = 1
```

If you click **Refresh Fields** and no error is thrown, then you're good to go! Click OK to save the new dataset.





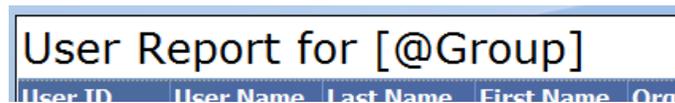
## SQL Report Builder

Once you've created the dataset, you can now use it for your dynamic option list. For example, let's say you wanted to show a drop down for all the groups in the database. In this case, the value field (what is actually searched on) would be the GroupSys, and the display field (what the user sees) would be the GroupName.

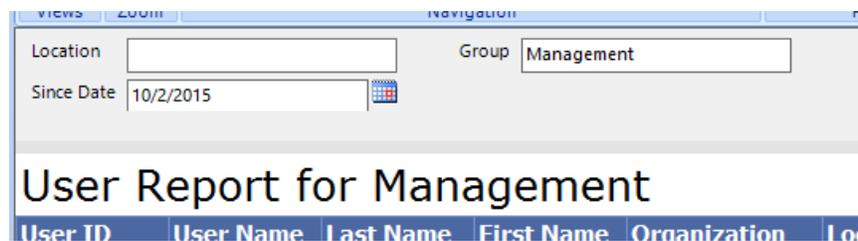
### 4.3 Displaying Filters and Other Fields

SQL Report Builder allows you to display certain variables directly on the report, which is useful after exporting the report. For example, you can show the filter parameters on the report by dragging the parameter into a text field in the report designer.

In the example below, I've added the Group filter to the header of the report. You can either drag the parameter into this textbox, or you can type in `[@Group]` manually.



When the report is run, the group I chose can be seen directly on the report so that another user can see what I chose for this filter.



On top of showing filters, you can also show some built in parameters. Each of these can be found in the **Built-in Fields** folder (above **Parameters**) on the far left. The most useful parameters are mostly like Execution Time (when the report was run), Page Number, and Overall Page Number. For example, you can type in `[&ExecutionTime]` to show the execution time directly on the report.

## Sample Report Query

Of course, there are thousands of different custom reports you could create using SELECT statements or Stored Procedures. Below is a very simply example.

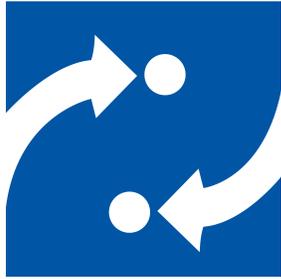


## SQL Report Builder

```
SELECT
    COALESCE(u.UserID, '') AS 'User ID',
    COALESCE(u.UserName, '') AS 'User Name',
    COALESCE(c.LastName, '') AS 'Last Name',
    COALESCE(c.FirstName, '') AS 'First Name',
    COALESCE(o.Name, '') AS 'Organization',
    COALESCE(ol.LocationName, '*** No Location ***') AS 'Location',
    COALESCE(c.GroupList, '*** No Group ***') AS 'Group',
    CONVERT(VARCHAR(30), COALESCE(cs.DateUpdated, '')) AS 'Last Updated',
    COALESCE(s.Status, '') AS 'Current Status'
FROM CurrentStatus cs
    INNER JOIN Contacts c ON c.ContactSys = cs.ContactSys
    INNER JOIN Users u ON u.ContactSys = c.ContactSys AND u.Guest = 0
    INNER JOIN Statuses s ON s.StatusSys = cs.StatusSys
    INNER JOIN Organizations o ON o.OrganizationSys = c.OrganizationSys
    LEFT JOIN OrganizationLocations ol ON c.OrganizationLocationSys = ol.OrganizationLocationSys
WHERE c.CurrentValue = 1
    AND COALESCE(o.Name, '') LIKE @Organization
    AND COALESCE(ol.LocationName, '*** No Location ***') LIKE @Location
    AND COALESCE(c.GroupList, '*** No Group ***') LIKE @Groups
    AND cs.DateUpdated NOT BETWEEN @DateUpdated AND GETDATE()
ORDER BY cs.DateUpdated ASC
```

This is a basic SELECT statement that gets all active users and their current statuses. It gets fields such as their ID, name, username, group, location, etc. First of all, take a look at the WHERE clause near the end. You'll notice that there are some parameters, such as *@Location* and *@Groups*. These are the parameters that are available to create filters in SQL Report Builder. In this case, the user can manually type in the Groups or select the groups from a dynamic option list.

Again, there are many more reports that can be made for users, status history, current statuses, timecards, daily usage, phone activity, and much more!



**SAVANCE**  
EIOBOARD

**Thank you for choosing Savance!**

**Savance**

**Phone: 248-478-2555 | Fax: 248-478-3270**

**[www.eioboard.com](http://www.eioboard.com) | [support@eioboard.com](mailto:support@eioboard.com) | [www.savance.com](http://www.savance.com)**

**EB-HG-SQLReports  
1.0.0**