IBM® Tivoli® Software

Maximo Asset Management – Version 7.5 Releases

QBR (Ad Hoc) Reporting and Report Object Structures

Document Version 8

Pam Denny
Maximo Report Designer/Architect

CONTENTS

REVISION HISTORY

Date	Version	Revised By	Comments
April 2014	8	PD	1. Updated to note that 'Use Where Clause' not enabled for QBR reports on page 70
August 2013	7	PD	1. Updated and reformatted content in sections 323 (ROS Best Practices) and 327 (Maxrelationship Business Rules) 2. Added content in business rule section that non-persistent attributes in maxrelationships not supported
May 2013	6	PD	1. Corrected parameter reference on page 48 to match screenshot (From Installation Date to Site) 2. Added 2.10 section for QBR Creation Limit Feature
January 2013	5	PD	1. Added note on errors resulting if MasterPage removed or renamed in Section 2.10 2. Updated formatting
October 2012	4	PD	1. Clarified that users can only select filters (Parameters) from selected, parent table attributes on page 31.
September 2012	3	PD	1. Added best practice information on 'QBR Security Access – Best Practices' 2. Added error message text, in addition to screen shots, for improved searching 3. Updated and combined Format notes on page 35
August 2012	2	PD	1. Expanded information on Public Security access, starting on page 33, including information on REPORTADHOCLOC Cron Task and SYNCREPORTLABELS maxvar 2. Updated report reference materials
September 2011	1	PD	Included details on 'Utilizing Report Object Structures in Application export functionality'

Document Overview

This document provides details on the Functionality and Business Logic used for the Release 7.5 Ad Hoc Reporting Capability known as Query Based Reporting (QBR). This includes information on creating and utilizing Report Object Structures (ROS) which are the basis for Ad Hoc Reporting.

This document first reviews the QBR functionality and its process flows. It then delves into the details of how you can create and execute Ad Hoc Reports. It then concludes with information on the setup work involved in enabling Ad Hoc Reports, including Security, Administration and how to enable Report Object Structures. QBR's are often referred to as Ad Hoc Reports, and the terminology is used interchangeably in this document.

Finally, please note that this document is based on the Maximo ® Base Services 7.5.x Release or later versions. For information on earlier 7.1x versions of QBR, reference earlier versions of this document, noted at the end of this document.

1 QBR Overview

Query Based Reports (QBR) are enabled to meet the ad hoc reporting needs of your users. QBR's enable users to create their own unique reports by selecting columns, specifying sorting and grouping, and the query the report will run against. Once the user saves the report, this information is fed through an API to create an xml report design file, which is then executed thru the report engine.

This functionality is intended for users to create their own custom reports. This is critical for many reasons, including

- Enables users to quickly create their own reports for their unique business needs
- Does not require technical, development skills, like Java, to create the report
- Does not consume development hours in creating and maintaining large numbers of unique reports, which may only be used by a very small number of users

QBR's are different than traditional, enterprise reports for the reasons listed below.

How they are developed
Enterprise reports are developed by a developer in the Design Tool, versus QBR reports are initiated by a user and created On the Fly within the applications.

Who uses them
Enterprise reports are developed for a large number of users, and are enabled for quick access. QBR reports are developed for a single user or business case, and are not designed for mass distribution.

Who has the ability to delete them
Enterprise reports can only be deleted by an Administrator who has access to the Report Administration application, or the database. QBRs can be deleted either by the User who created the Report, or by the administrator in the Report Administration application.

The Type of Security they have
Enterprise reports have group level security, where groups can be granted access at either an individual report level, or an application level. QBRs however are unique because they may be for a user's individual business needs, which may not want to be shared with others. Because of this requirement, QBR's utilize user level report security, which can be expanded to enable public and/or group security access.

*Note: For best practice information on enabling QBR report access for your users, see the section titled 'Best Practices - QBR Security Access' below.

Both Enterprise reports and Ad Hoc reports have their unique, individual business needs that they must meet. Enterprise reports are intended for access by high volumes of users, and their focus is on heavily formatted, detailed reports with complex sql code optimized for performance. Conversely, Ad Hoc reports are intended for access by a small number of users to meet their individual business needs. These use cases are often referred to as 'one off' business needs. Additionally, Ad Hoc reports are not optimized for formatting, as the assumption is that the user will determine the formatting of his report through individualized grouping and sorting, or exporting of the data to excel.

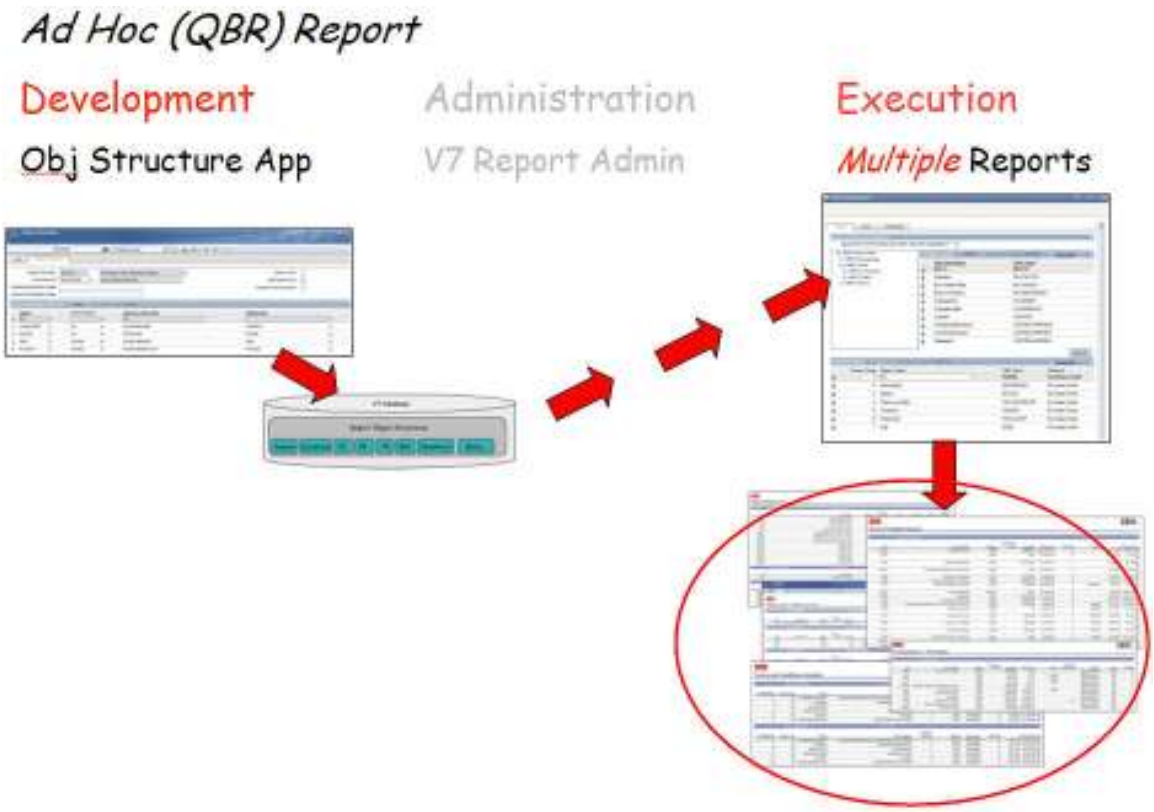
Because of their unique business needs, the process to develop an Enterprise report from an Ad Hoc Report differs significantly.

The diagram below shows the stages that an Enterprise report goes through from Report Development, Administration and then Execution. (Each of these steps is detailed in the Report Feature Guide.)

The Report Design, along with its properties and xml file, are created in the Report Designer by a report developer. These files are then imported into the Database. An administrator then utilizes the Report Administration application to set security access, and modify any property or application settings. Then, in the last stage, the user executes the report, and a single, highly formatted report is generated.

Ad Hoc reports have a significantly different reporting process. The goal of Ad Hoc reporting is to enable the user to create his own report – meaning he determines what fields/sorting/grouping etc are included in his report. This frees up significant development time as the developer no longer is required to produce each individual report requirement.

The diagram below shows the stages that an Ad Hoc or QBR report goes through from Report Development and then Creation and Execution by a user. Note that no Report Administration is required in these stages – the user simply creates his own individual reports on the fly.

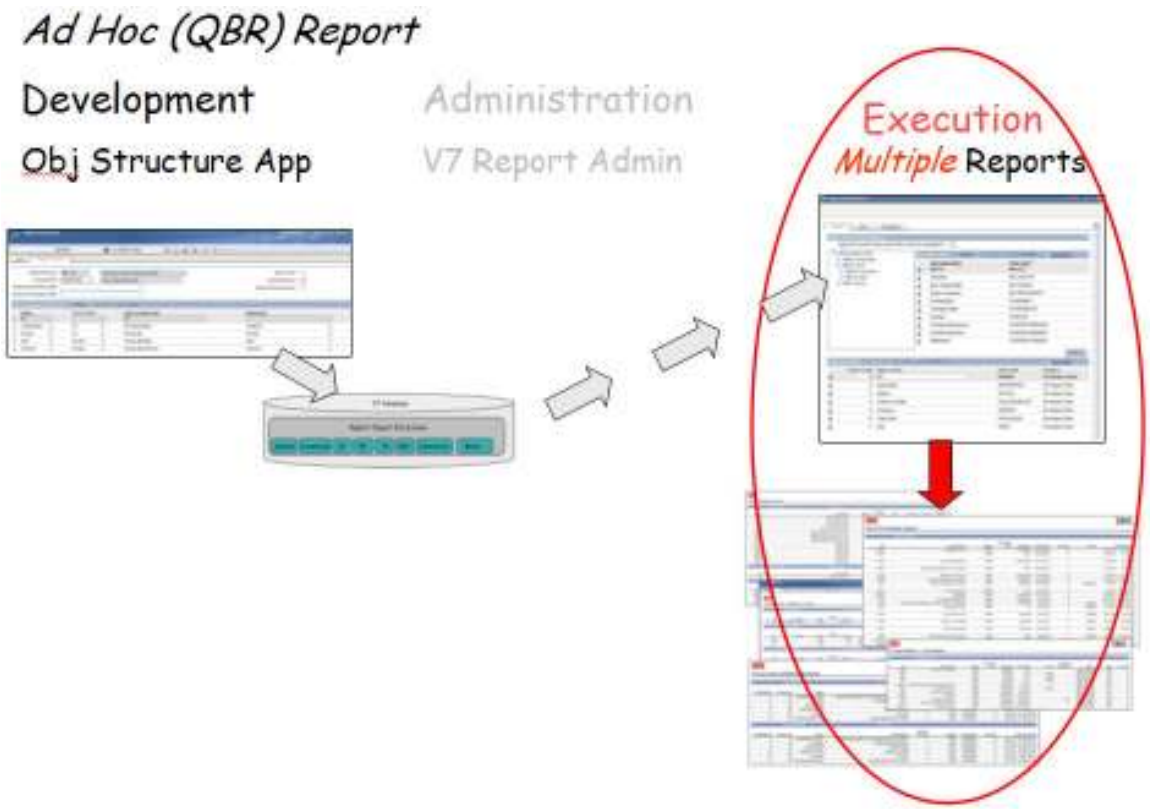


The development base of QBR reports is Report Object Structures (ROS). ROS contain the pre-joined tables of data that the user can select his report fields from. ROS are created in the Object Structure application, and stored in the database. Since formatting of the Ad Hoc reports is left up to the individual user, ROS are focused only on joining related tables. This saves significant development, administration and maintenance time. When a user wants to create his own report, he inputs his requirements, and any number of different reports can be generated based on the application's report object structure(s).

2 QBR – Creation/Execution Process

This document will begin by detailing how a user can create and execute Ad Hoc reports. This is the functionality highlighted in the circled portion of the diagram below.

The document will also review security best practices and functionality. It concludes with details on the setup work involved in enabling Ad Hoc reports, including Administration and how to enable Report Object Structures. This last portion is intended only for developers and/or administrators.



2.1 Create QBR: Access

First, creating QBR reports will be discussed. QBR reports can be created by an individual user for his unique business need, or by a supervisor or manager, who wants to create reports to share with a number of different individuals. The individual who creates the QBR report is identified as the owner of the QBR. These next pages detail the steps involved in creating a QBR report.

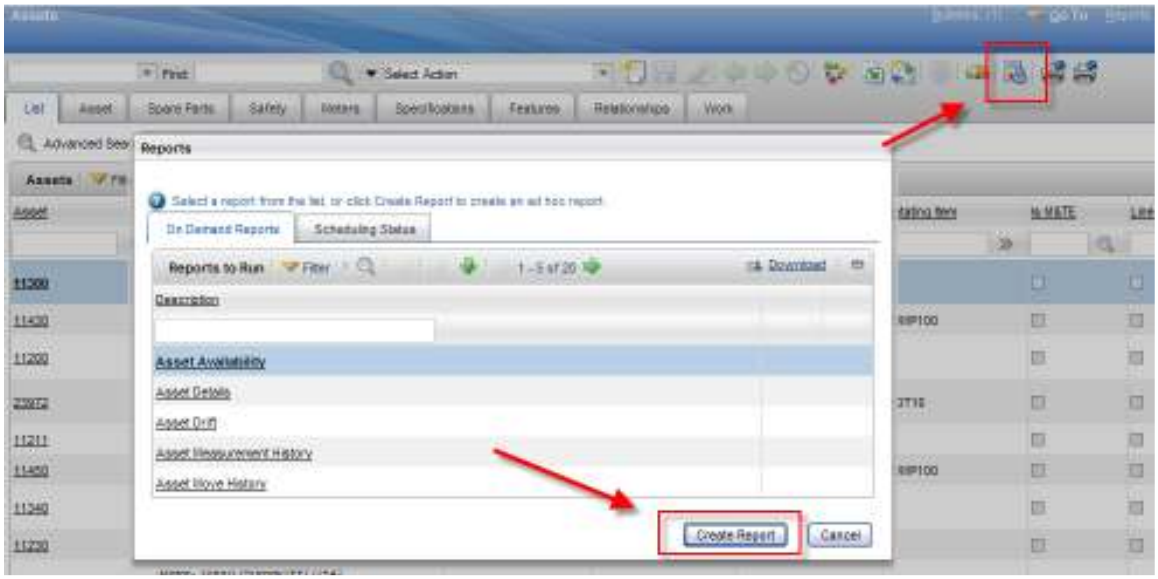
Users can go to a number of different places to access QBR, including

(1) Create Report icon in the toolbar menu within an application

(2) Create Report button in the bottom section of the Reports window.

This Reports window can be accessed within an application or from the Report Link on the top right side of the Application Menu.

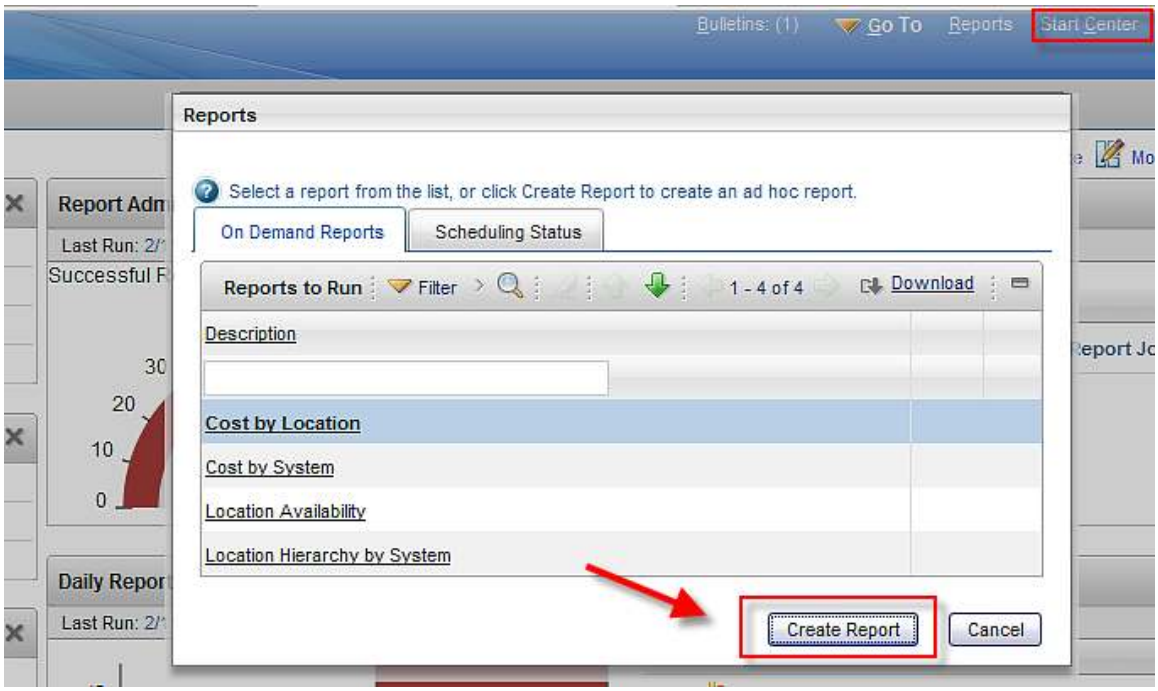
Examples of these locations are highlighted below.



Notes on Accessing QBR Functionality:

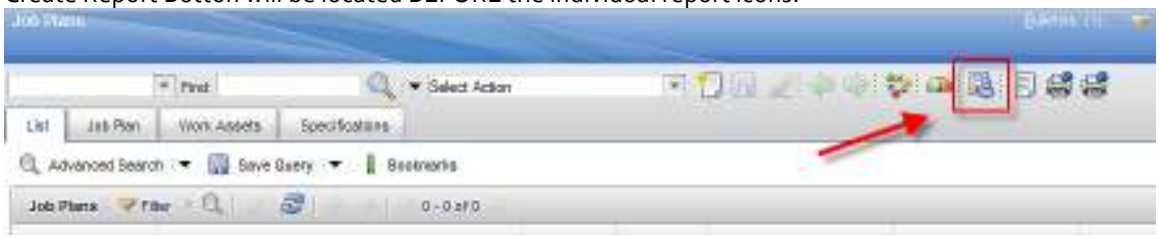
1. The ability to Create New QBRs outside of an application is available. This means the user can access the functionality from the Reports Menu on the Start Center, or access it for an application external to his current application. For example, if a User was in the Asset application, and from the top Report Menu, selected Reports > Work Orders, the Create Report button in the Reports Window for Work Order would be available.

- This access is seen with in the screen shot below from the Start Center.
- If the user accesses the Create Report Functionality outside of an application, there will be no default values in the Selected Field Section of the QBR's Select Tab. This occurs because the list of persistent fields for the application are not available outside of the application.

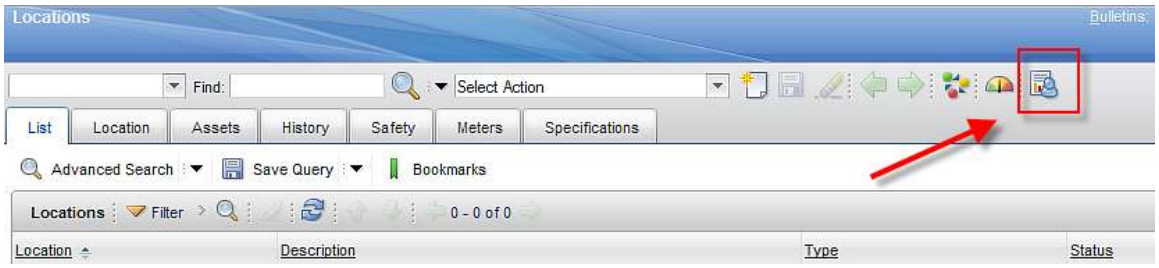


2. The Create Report Icon will vary in its exact location on the application's toolbar – depending on the application it is accessed from via the business rules below.

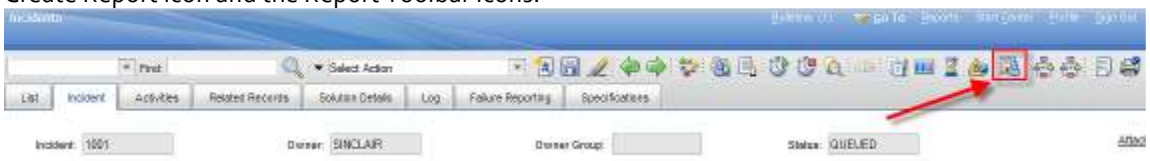
A. If an application has individual report icons configured (ex. Direct Print or Browser View) the Create Report Button will be located BEFORE the individual report icons.



B. If no Report Icons are configured, the 'Create Report' Icon will display as the last of the application tool bar icons.



C. However, if an application has Workflow options enabled, the workflow icons will reside between the Create Report Icon and the Report Toolbar Icons.

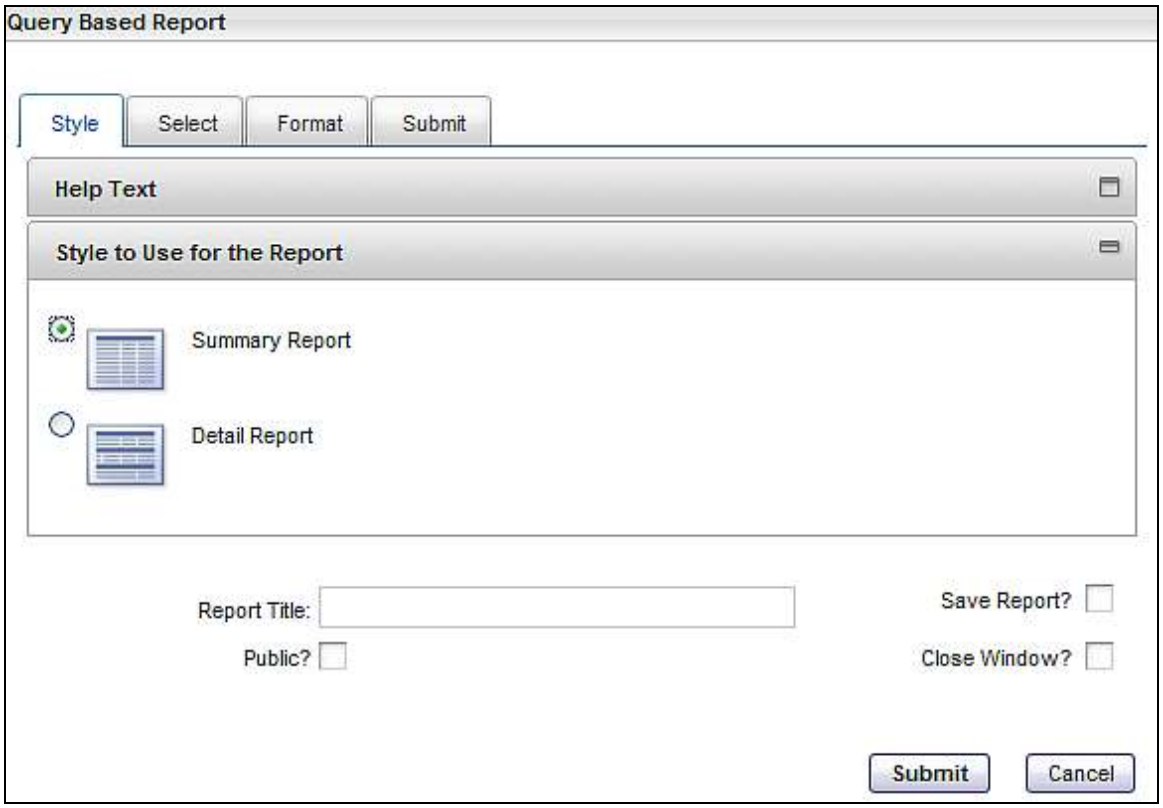


3. The Create Report Icon is available on any of the tabs within the applications.

2.2 Create QBR: Style Tab

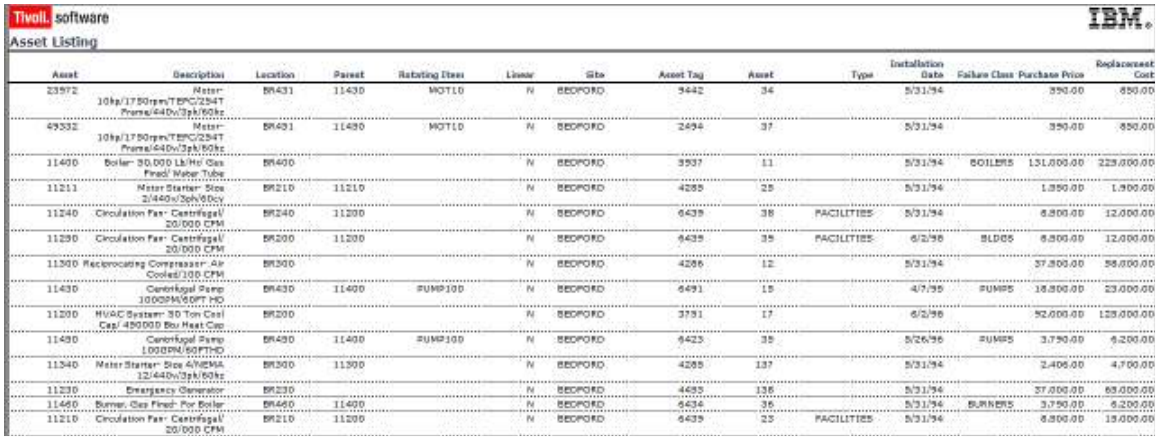
When the user clicks on the Create Button, a new window displays. This is the window the user will utilize to create his unique report, and is often referred to as the QBR wizard or QBR window. The QBR window contains four subtabs – Style, Select, Format and Submit. It opens to the first tab, Style.

On the Style Tab, the user can select from two types of reports: A Summary report, or a Detail report. A visual representation of the two different reports is displayed on this tab.



2.2.1 Style Tab: Summary Report

A Summary report lists data which can be grouped and sorted. Its primary purpose is often to export the data to Microsoft Excel for additional analysis.



Tivoli software

IBM.

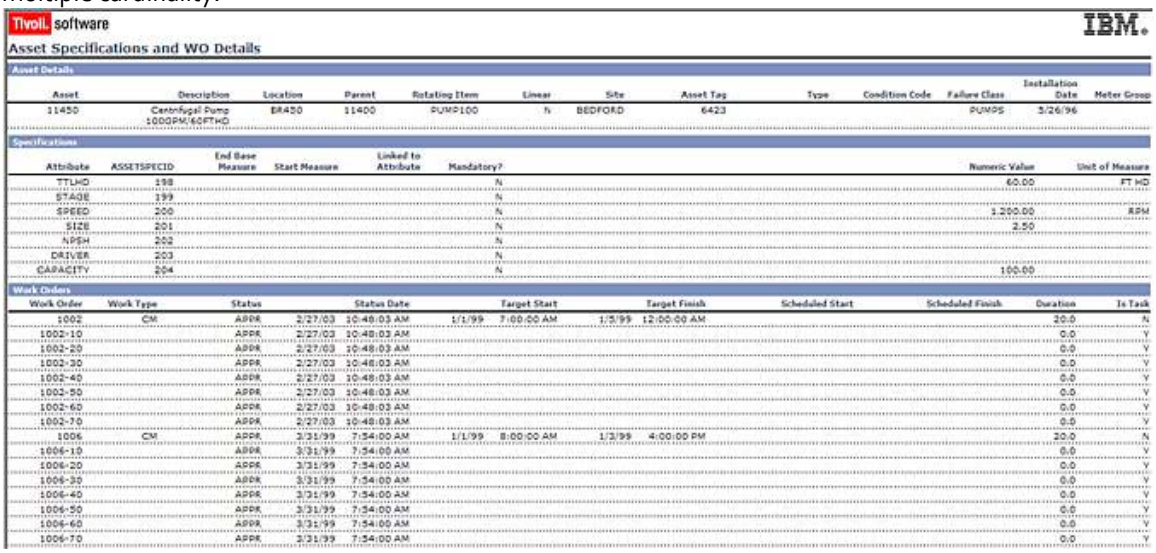
Asset Listing

Asset	Description	Location	Parent	Rotating Item	Linear	Site	Asset Tag	Asset	Type	Installation Date	Failure Class	Purchase Price	Replacement Cost
23972	Motor-10hp/1750rpm/TEFC/254T Frame/440v/3ph/60hz	BR431	11430	MOT10	N	BEDFORD	9442	34		5/31/94		350.00	650.00
49332	Motor-10hp/1750rpm/TEFC/254T Frame/440v/3ph/60hz	BR431	11430	MOT10	N	BEDFORD	2494	37		5/31/94		350.00	650.00
11400	Boiler- 30,000 Lb/Hr Gas Fired/ Water Tube	BR400			N	BEDFORD	3937	11		5/31/94	BOILERS	131,000.00	225,000.00
11211	Motor Starter- Size 2/440v/3ph/60cy	BR210	11210		N	BEDFORD	4265	25		5/31/94		1,350.00	1,900.00
11240	Circulation Fan- Centrifugal/ 20/000 CFM	BR240	11200		N	BEDFORD	6439	38	FACILITIES	5/31/94		8,800.00	12,000.00
11200	Circulation Fan- Centrifugal/ 20/000 CFM	BR200	11200		N	BEDFORD	6439	39	FACILITIES	6/2/96	BLDGS	8,800.00	12,000.00
11300	Reciprocating Compressor- Air Cooled/100 CFM	BR300			N	BEDFORD	4266	12		5/31/94		37,500.00	58,000.00
11430	Centrifugal Pump 1000PM/60FT HD	BR430	11400	PUMP100	N	BEDFORD	6491	15		4/7/95	PUMPS	18,800.00	23,000.00
11200	HVAC System- 30 Ton Cool Cap/ 450000 Btu Heat Cap	BR200			N	BEDFORD	3731	17		6/2/96		92,000.00	125,000.00
11430	Centrifugal Pump 1000PM/60FTHD	BR430	11400	PUMP100	N	BEDFORD	6423	35		5/26/96	PUMPS	3,750.00	6,200.00
11340	Motor Starter- Size 4/NEMA 12/440v/3ph/60hz	BR300	11300		N	BEDFORD	4265	137		5/31/94		2,406.00	4,700.00
11230	Emergency Generator	BR230			N	BEDFORD	4455	136		5/31/94		37,000.00	65,000.00
11460	Burner, Gas Fired- For Boiler	BR460	11400		N	BEDFORD	6434	36		5/31/94	BURNERS	3,750.00	6,200.00
11210	Circulation Fan- Centrifugal/ 20/000 CFM	BR210	11200		N	BEDFORD	6439	23	FACILITIES	5/31/94		8,800.00	15,000.00

With a Summary report, users can define up to 3 Filters (Parameters), along with 3 grouping and 3 Sorting Sections.

2.2.2 Style Tab: Detail Report

A Detail report contains details on records, with page breaks separating the record's details. These types of reports are more complex in nature, and enable the use of multiple 1:N (One to Many) relationships, or multiple cardinality.



Tivoli. software

IBM.

Asset Specifications and WO Details

Asset Details

Asset	Description	Location	Parent	Rotating Item	Linear	Site	Asset Tag	Type	Condition Code	Failure Class	Installation Date	Meter Group
11450	Centrifugal Pump 1000GPM/60FTHD	BR450	11400	PUMP100	N	BEDFORD	6423			PUMPS	3/26/96	

Specifications

Attribute	ASSETSPECID	End Base Measure	Start Measure	Linked to Attribute	Mandatory?	Numeric Value	Unit of Measure
TTLHD	198				N	60.00	FT HD
STAGE	199				N		
SPEED	200				N	1,200.00	RPM
SIZE	201				N	2.50	
NPSH	202				N		
DRIVER	203				N		
CAPACITY	204				N	100.00	

Work Orders

Work Order	Work Type	Status	Status Date	Target Start	Target Finish	Scheduled Start	Scheduled Finish	Duration	Is Task
1002	CM	APPR	2/27/03 10:48:03 AM	1/1/99 7:00:00 AM	1/5/99 12:00:00 AM			20.0	N
1002-10		APPR	2/27/03 10:48:03 AM					0.0	Y
1002-20		APPR	2/27/03 10:48:03 AM					0.0	Y
1002-30		APPR	2/27/03 10:48:03 AM					0.0	Y
1002-40		APPR	2/27/03 10:48:03 AM					0.0	Y
1002-50		APPR	2/27/03 10:48:03 AM					0.0	Y
1002-60		APPR	2/27/03 10:48:03 AM					0.0	Y
1002-70		APPR	2/27/03 10:48:03 AM					0.0	Y
1006	CM	APPR	3/31/99 7:54:00 AM	1/1/99 8:00:00 AM	1/3/99 4:00:00 PM			20.0	N
1006-10		APPR	3/31/99 7:54:00 AM					0.0	Y
1006-20		APPR	3/31/99 7:54:00 AM					0.0	Y
1006-30		APPR	3/31/99 7:54:00 AM					0.0	Y
1006-40		APPR	3/31/99 7:54:00 AM					0.0	Y
1006-50		APPR	3/31/99 7:54:00 AM					0.0	Y
1006-60		APPR	3/31/99 7:54:00 AM					0.0	Y
1006-70		APPR	3/31/99 7:54:00 AM					0.0	Y

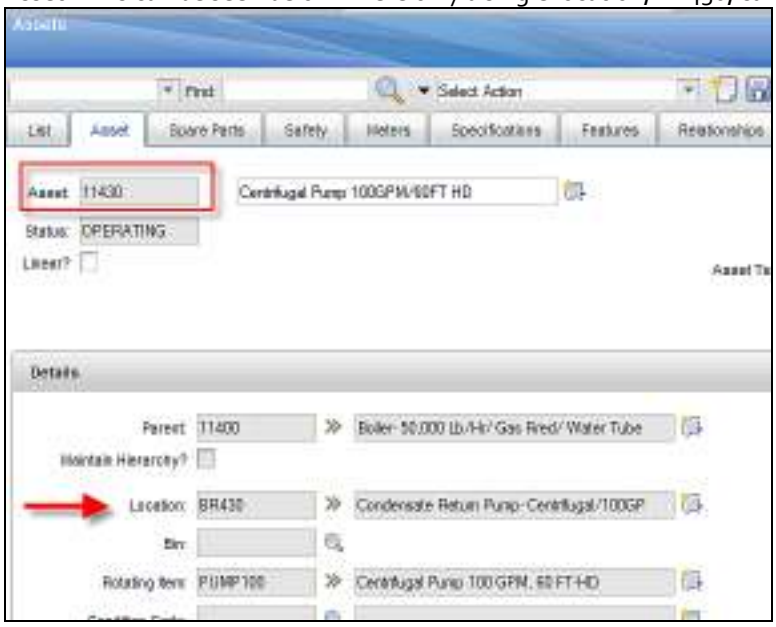
The key difference between a Summary and Detail report is the number of multiple relationships you can select in your report. There are no limits to the number of multiple relationships you can select in a Detail report – whereas a Summary report restricts you to one or less.

Additionally a user could select a Detail report, and only select fields from the Parent Category or Main table of the application. In this case, the user has no categories with multiple cardinality and all fields will display in the header section.

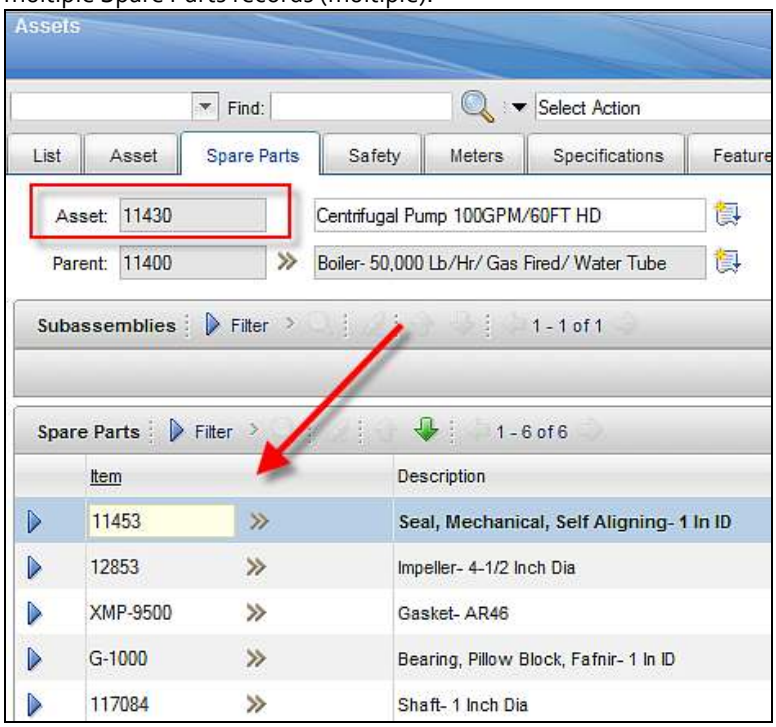
If you are unsure as to the type of report you want to create, you may want to select the Detail report type as it enables the greatest field selection in your report design.

Before continuing, single and multiple relationships will be defined. Cardinality is a database term defining the type of relationship. In the case of QBR reporting, there are 2 types: One-to-one (1: 1) or One-to-many (1:N). One-to-one relationships are known as Single Relationships, whereas One-to-many are Multiple Relationships.

An example of a 1:1, single relationship is Asset-Location as only a single location can be associated to an Asset. This can be seen below where only a single location, BR430, can be associated to Asset 11430.



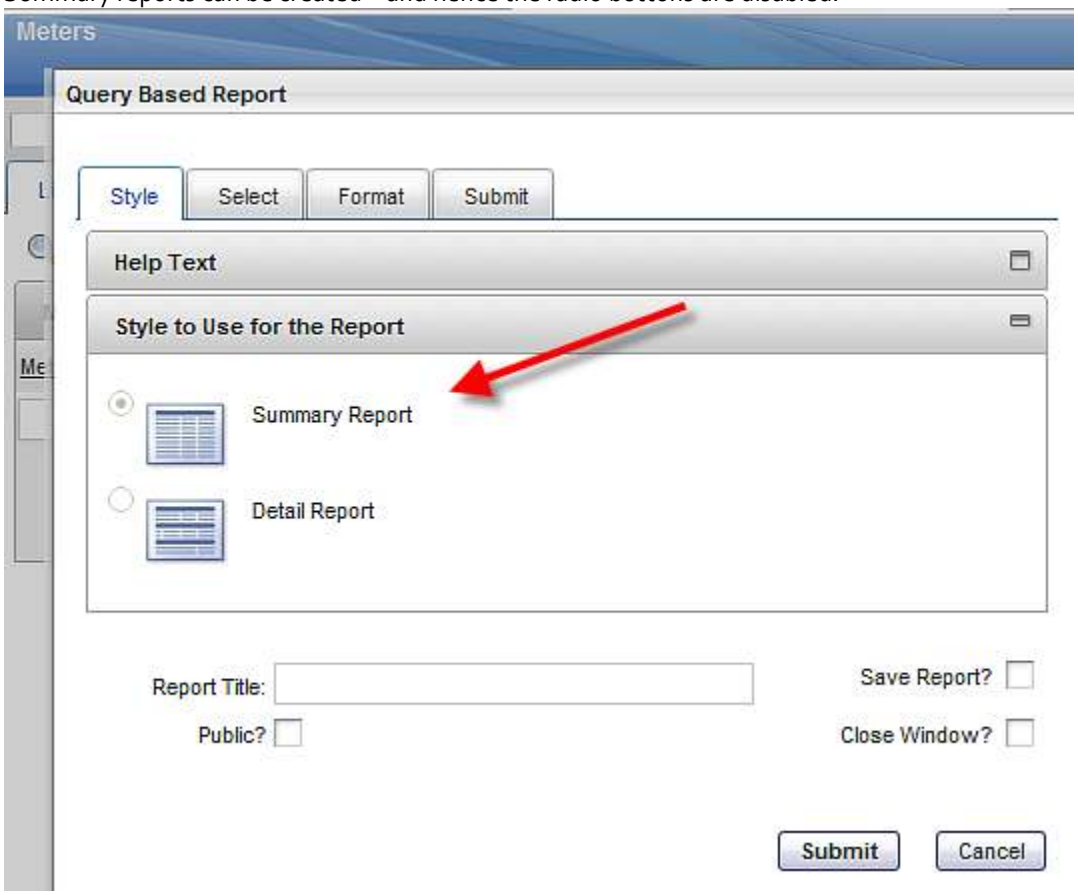
An example of a 1:N, multiple relationship is Asset-Spare Parts. For each asset (single) there can be multiple Spare Parts records (multiple).



More details on cardinality are contained in the section MAXRELATIONSHIP.CARDINALITY at the end of this document.

Style Tab Notes

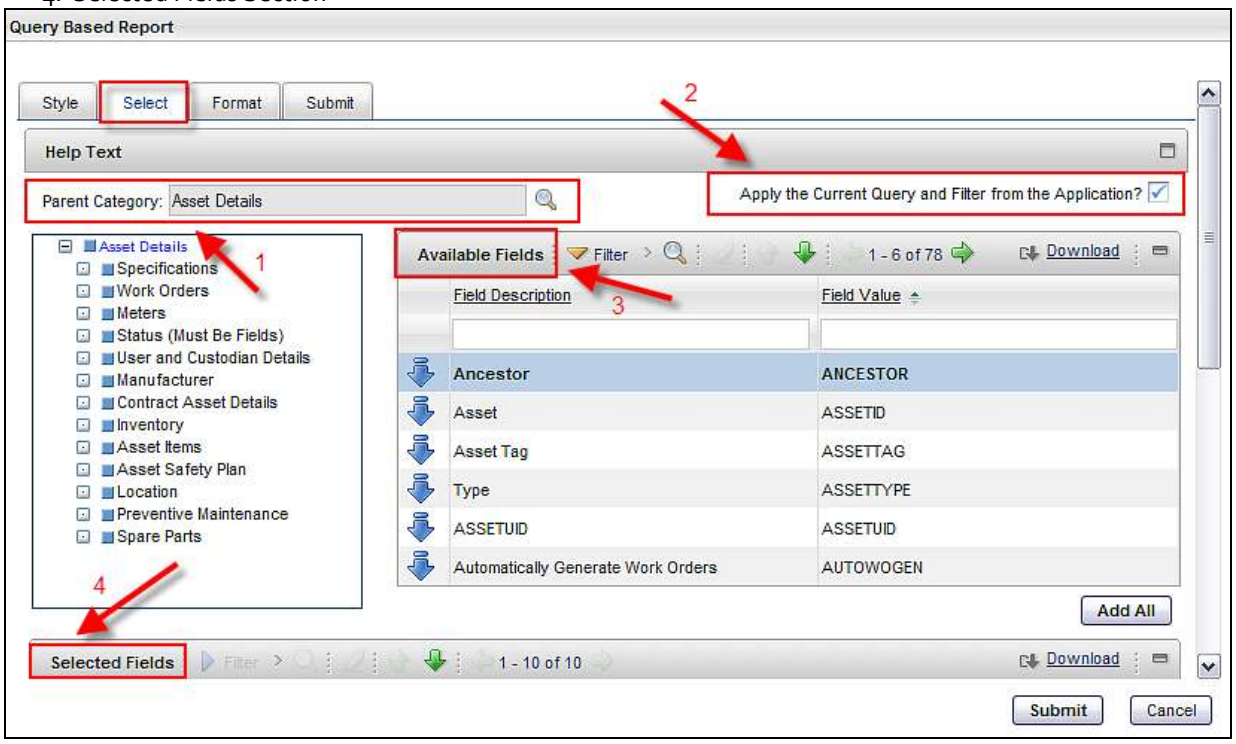
1. You may access the QBR Functionality for an application, and notice that its Radio Buttons are disabled on the Style Tab. This occurs because there is no ROS for the application, or the user does not have rights to access any existing ROS for the application. When there is no available ROS for the application, detail reports can not be created because there are no child tables. In this case, only Summary reports can be created – and hence the radio buttons are disabled.



2.3 Create QBR: Select Tab

The select tab of QBR is where the user will spend the most amount of time. It has four critical sections, which are

1. Parent Category
2. Query to Use in the Report
3. Available Fields Section, Including Drill Down Tree
4. Selected Fields Section



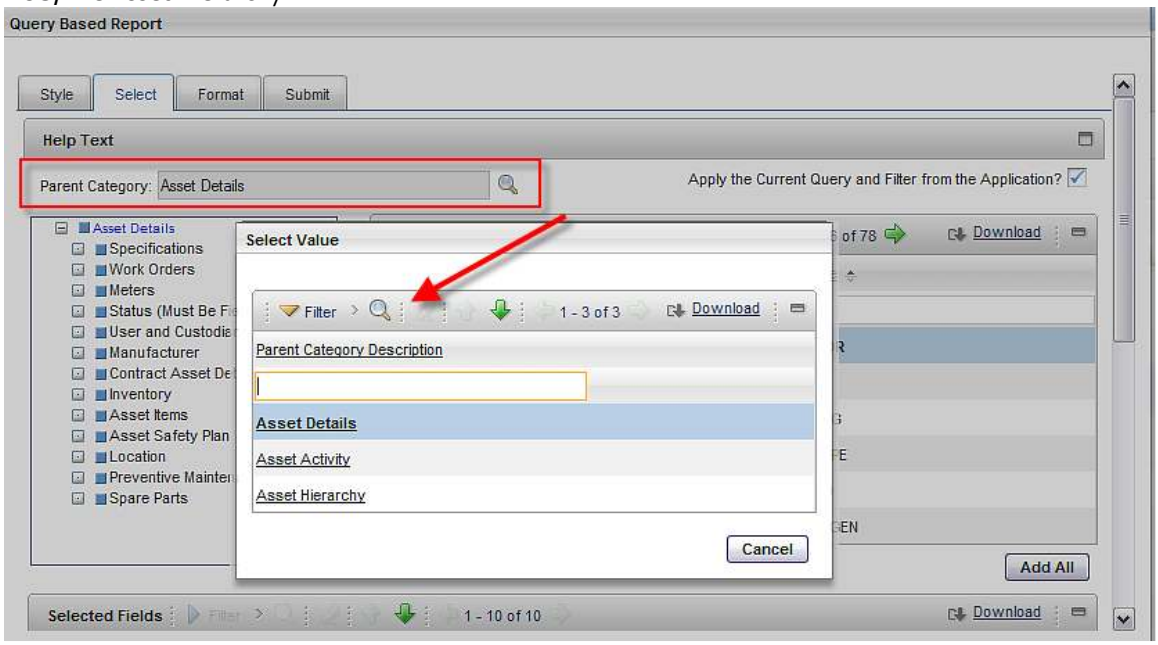
2.3.1 Select Tab: Parent Category

The Parent Category enables a user to select from any number of report object structures (ROS) associated to an application. Parent Category is simply a user friendly term for Report Object Structure (ROS).

When the user enters the Select Tab, a default parent category Value will display. The default value will be

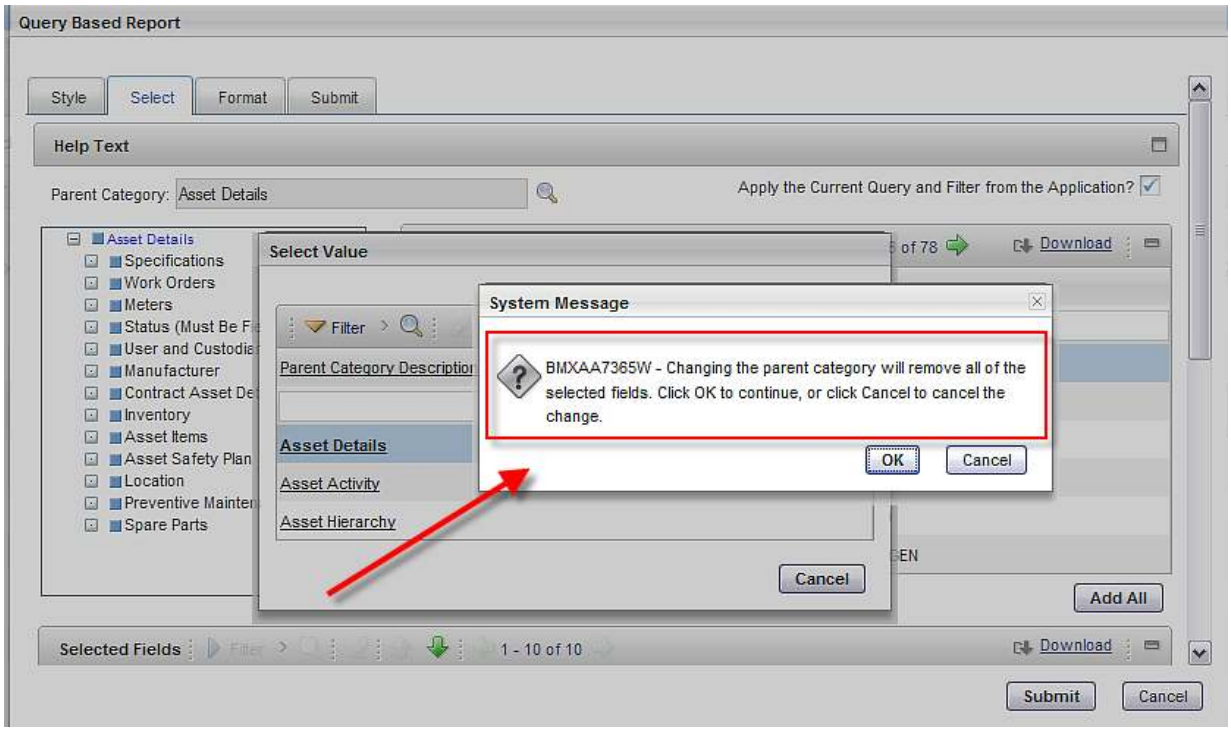
(1)If no ROS are defined for an app, the parent category will display as 'Application Name' Detail
(2) If only ROS is defined for an app, the single ROS will be the default value
(3) If multiple ROS are defined for an app, the first ROS entered in the Object Structure app will display

The screen shot below shows the Asset application that has three parent categories, or three ROS. The user can choose to build his QBR report from the default 'Asset Details' ROS, or switch to a different ROS, like Asset Hierarchy.

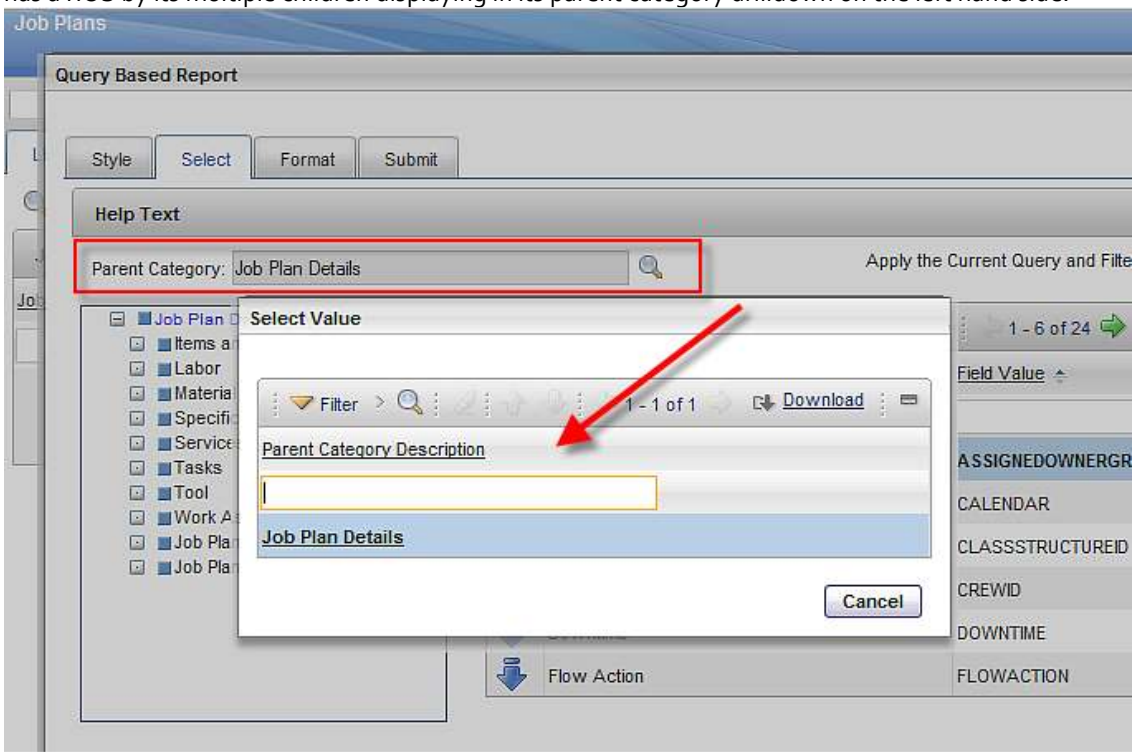


*NOTE: The user will only see the ROS that he has access to. Details on granting security access to ROS are noted in the Security Access Section later in this guide.

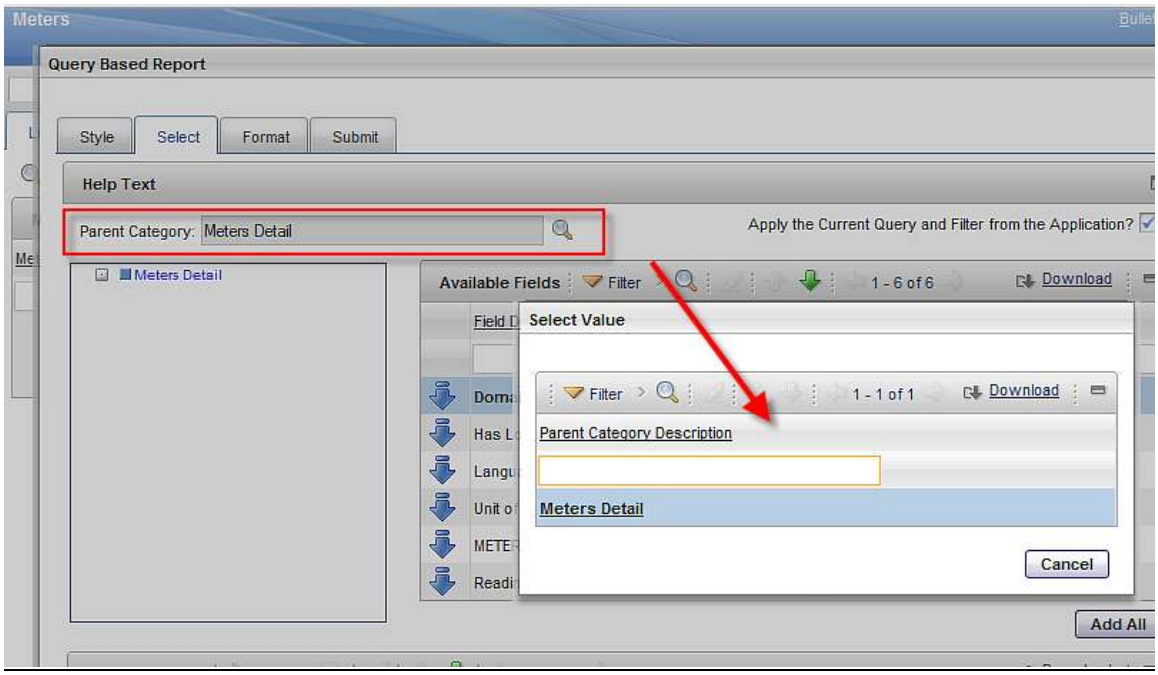
If the user switches ROS, they will lose any selected fields they may already have added as shown below.



Here is a screen shot of an application that only has a single ROS available to the user. You can identify it has a ROS by its multiple children displaying in its parent category drilldown on the left hand side.

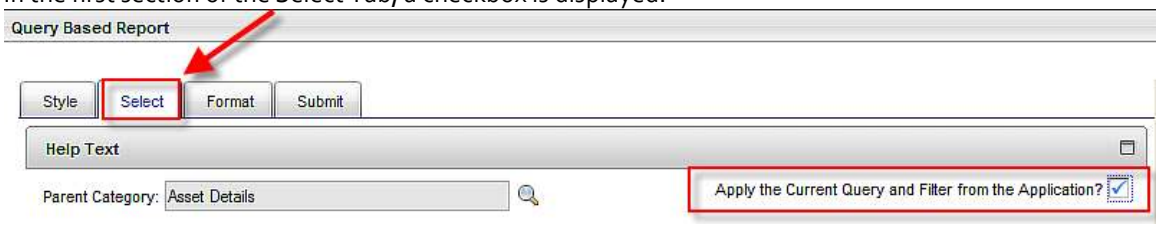


Here is a screen shot of an application that does not have a ROS available to the user. You can identify it does not have a ROS because it has no children under the parent category drilldown - Only Meters Details is displayed.



2.3.2 Select Tab: Query to use in the Report Section

In the first section of the Select Tab, a checkbox is displayed.



This checkbox enables the user the option to apply their existing query and filters from an application or not.

- If the user enables the checkbox, the application's current query and filter will be saved to the report. Whenever the report executes, it will always execute with that applied query. This is the default value.
- If the user does not enable the checkbox, the application's current query and filter will *Not* be saved to the report.

Enabling the Query Flag saves the query as the 'Saved Where Clause'. More details on the display of this to the user is detailed beginning on page 45.

Additionally, this functionality can be described with the example below of enabling the Query Flag in the Asset application.

A. User Bob goes to the ASSET application, and opens to his saved query named BEDFORD.. The BEDFORD query has a where clause of *…where (siteid like '%BEDFORD%')*

B. Bob then applies a filter of BR in the location field on the Asset list tab. His application's sql where clause is updated to
….where (BEDFORD Query) + *(Application Filter)*
Which is also described as
((siteid like '%BEDFORD%')) and *((location like '%BR%'))*

C. Bob creates a QBR report and wants it to ALWAYS execute against both the saved query + filter. To do this, he

1. Enables the 'Apply the Current Query and Filter from the Application'.
2. Saves his report to something he will quickly recognize as 'Bedford Assets in BR Location'.

When the report is saved, the application's current where clause (query + filter) is saved in the sql of the report design as shown below.
((siteid like '%BEDFORD%')) and *((location like '%BR%'))*

Whenever Bob executes his 'Bedford Assets in BR Location' Report in the future, it will ALWAYS apply the saved where clause *((siteid like '%BEDFORD%'))* and *((location like '%BR%'))* to the report's results.

Next, we'll describe what will happen when the Query Flag is NOT enabled.

A. Bob is in the Asset application, and opens to his saved query BEDFORD. He applies the same filter of BR in the location field.

B. Bob creates a new QBR report, but does ***NOT*** want it to ALWAYS execute against the combination of his current query + filter. Instead, he wants the report to dynamically pick up the active query and filter from the application at the time he executes the report. Therefore, he….

1. Does NOT Enable the 'Apply the Current Query and Filter from the Application'
2. Saves the report as 'Bob's Asset Report'.

When the report is saved, the application's current where clause (query + filter) is ***NOT*** saved in the sql of the report design. Instead, only Bob's Field Selection is saved, along with any database joins, sorting and grouping he selected.

C. Two days later, Bob goes back to the Asset application. He opens up to one of his saved application's queries of ASSETDOWN, which has a where clause of…
…..where isrunning = '0'

D. Bob filters on '11' in the asset field on the list tab and his sql where clause becomes
…..where (isrunning = '0) and (asset like %11%)

E. Bob then wants to run his 'Bob's Asset Report' against this current record set. So, he executes the report, and since the 'Save Application's Current Query' Flag WAS not enabled (applied) to the saved report, the report accepts the current, dynamic filter from the application which is
…..where (isrunning = '0') and (asset like %11%)

Each of these two options supports multiple use cases, so it will be up to the user creating each individual QBR if he wants to apply the Application Current Query and Filter – or not.

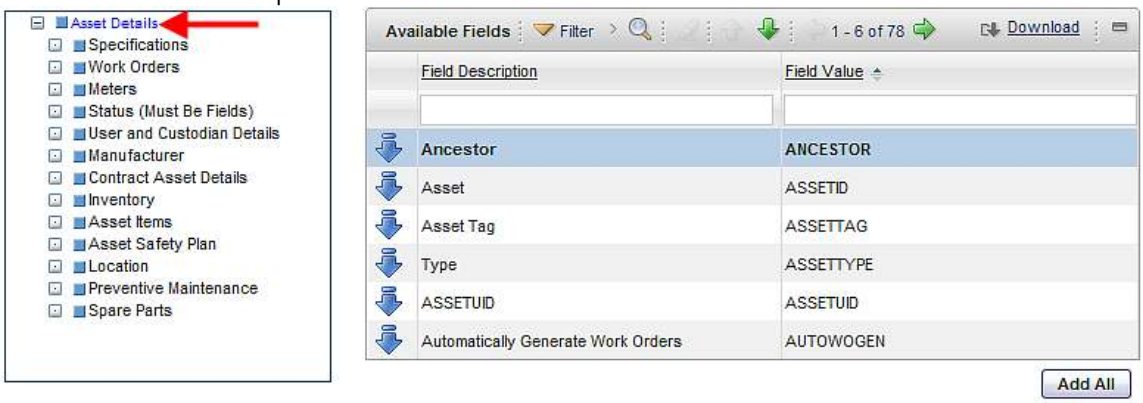
Additionally, users can also define run time filters, or parameters that can be applied to the QBR at run time. More details on this functionality is in the Format Tab below.

2.3.3 Select Tab: Available Fields Section

The third section of the Select Tab is the Available fields section. On the right side, the available fields a user can add to his report display. On the left, a category tree drilldown displays. Categories are groupings of tables (objects) that are related via max relationships. These categories are configurable, and will vary significantly by application.

Categories enable the user to select fields for his report from multiple tables or objects. They are designed to enable the user to visualize the categories and their parent/child relationships. These report categories are defined in the Integration – Object Structure application as Report Object Structures (ROS). More details on creating these are contained at the end of this document.

Continuing with our example, the parent category is Asset. This parent has multiple children (Specifications, Work Orders, Status…). The user can scroll through these categories to add the fields he wants for his individual report.



When the user clicks on the parent level category, it expands to show all of its children. If a child does not have any children, no further drilldowns will be available. If a child has additional children, parent's grandchildren, they will visually display with the + icon. Multiple children-grandchildren relationships can be configured.

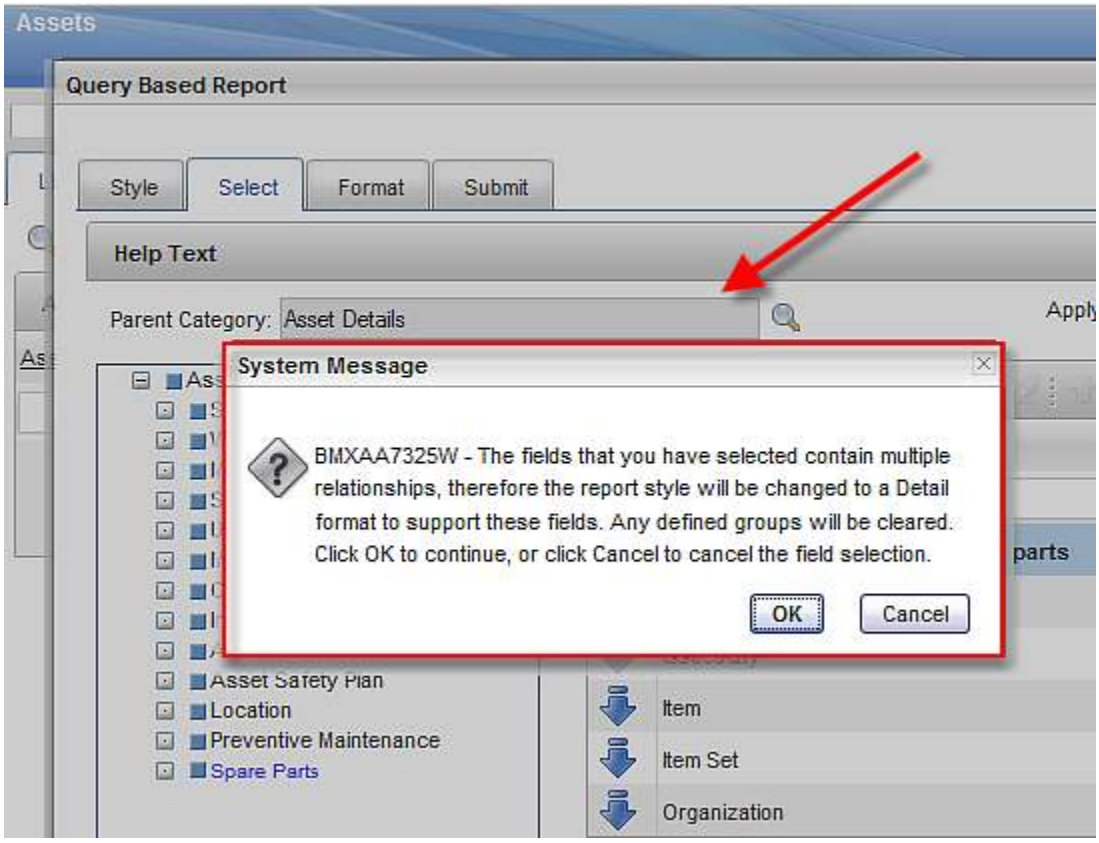
The user can add fields to his report by using the arrow icon next to each field. Additionally, he can 'Add All' of the fields at once to add fields in mass.

Standard filtering capability is enabled on (1) Field Description or (2) Field Value to help the user quickly find his values. Also note that the 'Available Fields' and 'Selected Fields' section scroll independently of each other. For example, if the user is scrolling through the selected fields section, the Available fields section stays as is.

During the report creation process, as the user adds fields to the Selected Field section, their category relationships are evaluated to determine if they are single or multiple relationships. Any number of categories with Single Relationships can be added to the QBR.

However, when categories have Multiple Relationships, different business rules are invoked.
If the user has selected a Summary report, only 1 Category with a Multiple Cardinality can be included. If the user tries to select fields from multiple categories with 1:N relationships, he will be given the choice to either change his style type to Detail, or to remove the field selection. This is shown with the message below

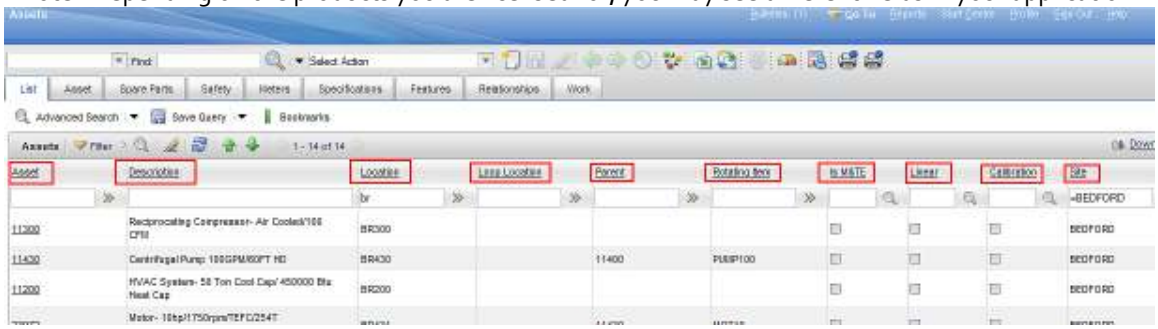
BMXAA7325W: The fields you have selected contain multiple relationships, therefore the report style will be changed to a Detail format to support these fields. Any defined groups will be cleared. Click O to continue, or click Cancel to cancel the field selection.



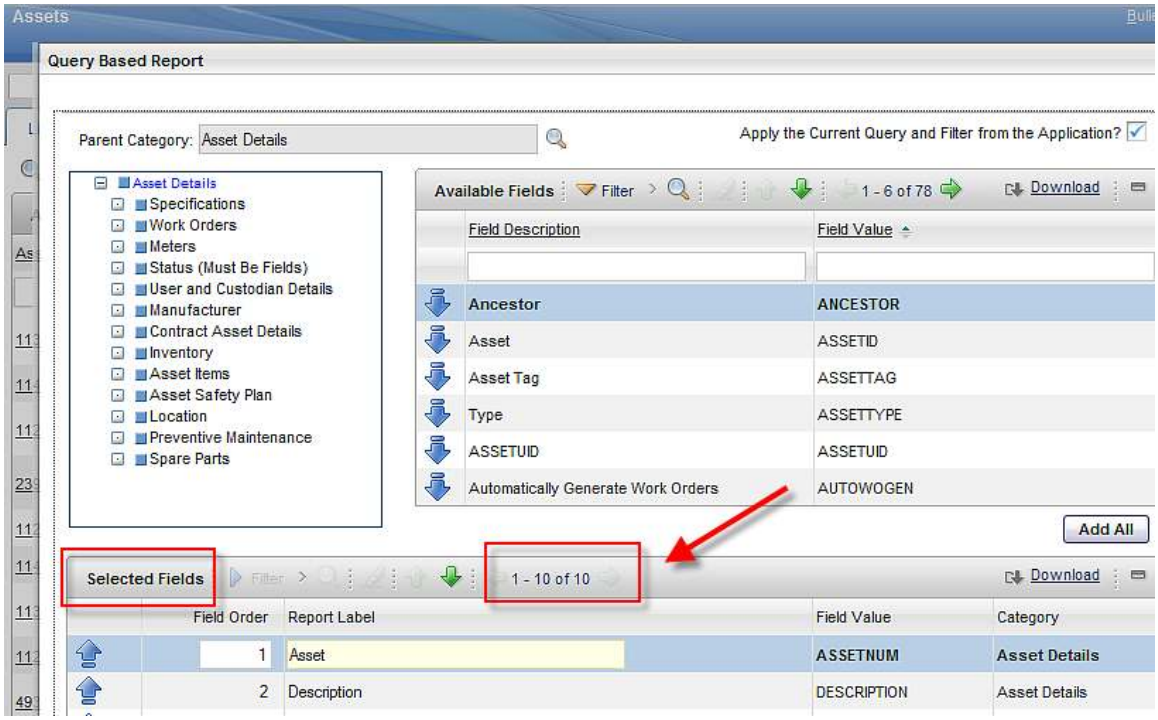
2.3.4 Select Tab: Selected Fields Section

To save the user time and clicks, all of the persistent fields from the Application's List page display in the selected field section by default. Because these are the most frequently used fields in an application, this saves the user time from having to add these fields each time he wants to create a report.

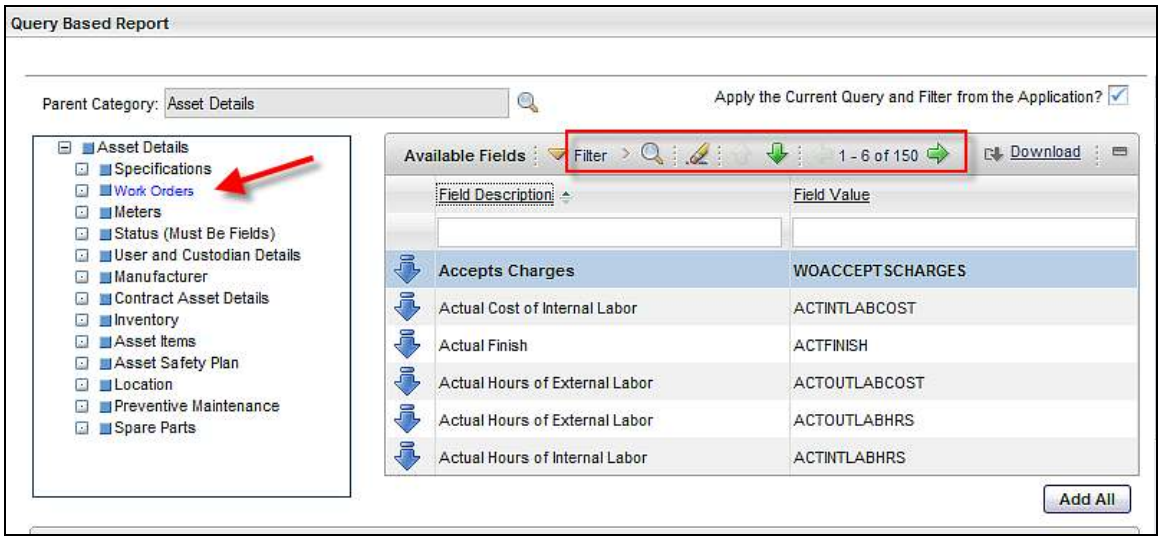
In the Asset application below, there are 10 persistent fields on the list tab as highlighted by the red blocks.
*Note: Depending on the products you are licensed for, you may see different fields in your application.



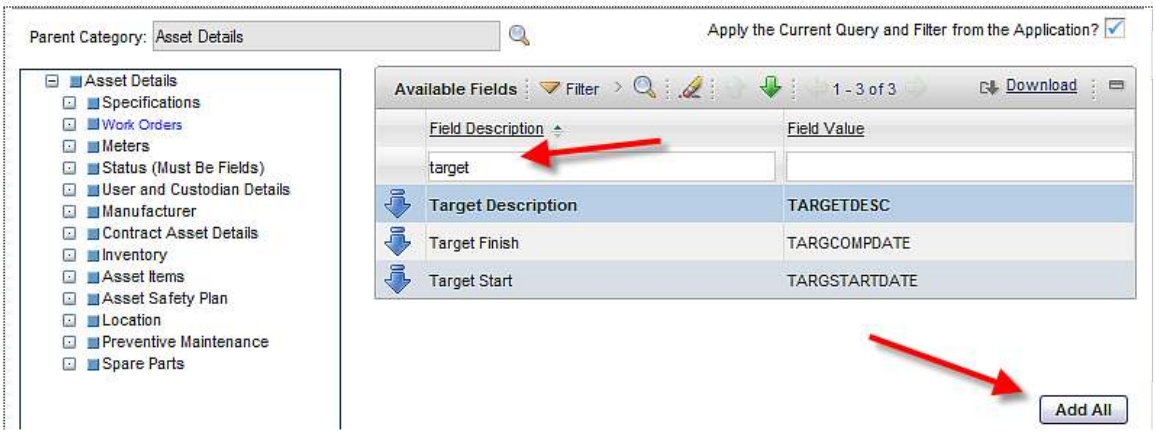
When the user opens up the Query Window, each of these persistent fields has been automatically added to the Selected Field section for the user.



To locate fields to add to the user's selected field section, the user can quickly navigate through the Available fields section to see what other fields are available to add. He can navigate through the Drilldown Category on the left, and select any children or grandchildren categories. When selecting a new category, the Available fields window will refresh with the fields from the new category. Additionally, the highlighted category will be visualized by a blue text color in the drilldown window.

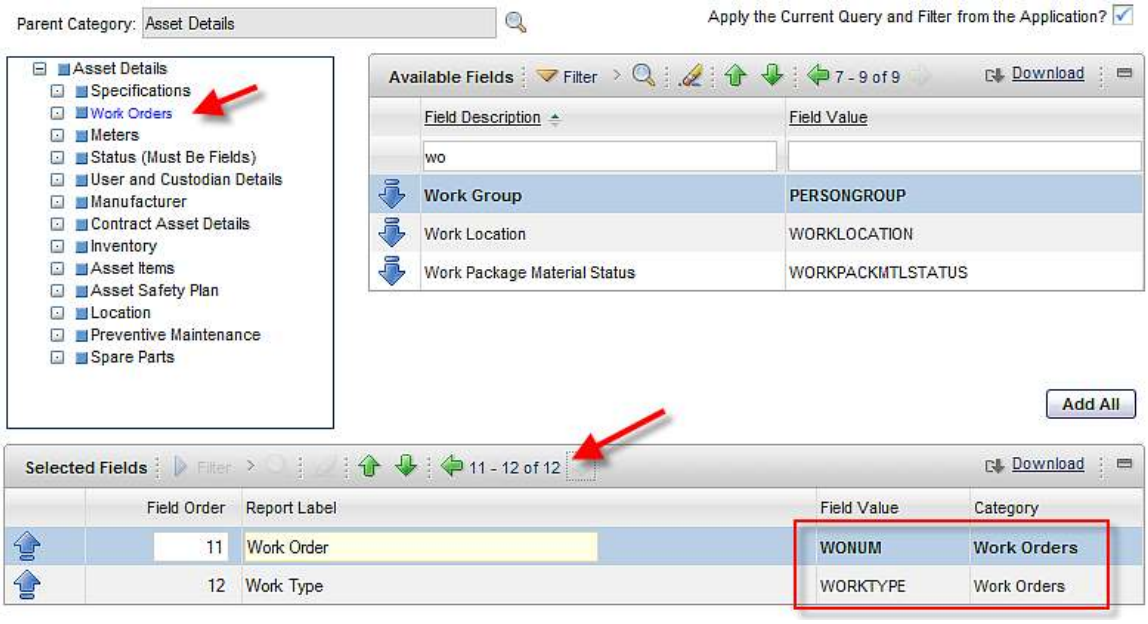


Additionally, the user can filter for specific text. When the results display, the user can add individual fields by using the down arrows next to the Field description, or use the 'Add All' button to add all of the fields from his Filtered Query to the selected field section below.

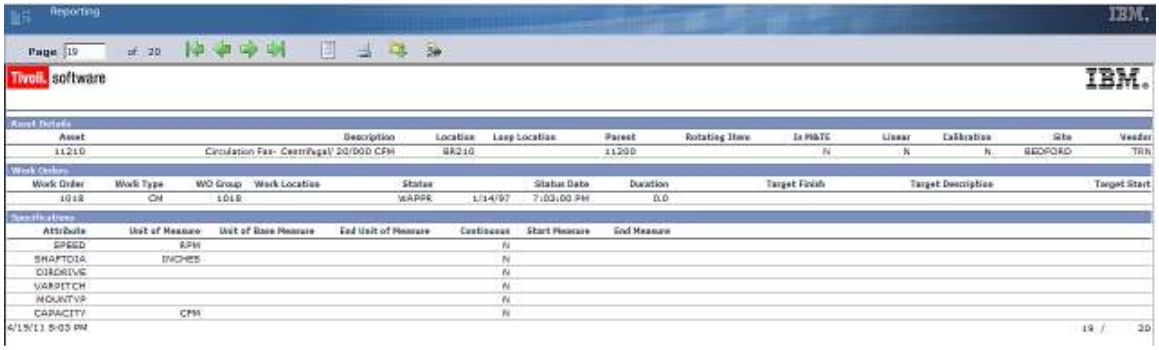


In this example, the user selected the Work Order category and filtered on 'wo'. He then added the Work Order and Work Type fields to the Selected Fields. This action causes the followings

A. Adds the new field(s) to the end of the Selected field list.
B. Displays the field's category on the right of the Selected field section
C. Removes the selected field from the Available field section.



The user can continue to add more fields to his QBR report until he has the complete listing he needs. In this example, multiple fields were added from the Work Order and Specifications Categories as shown below.



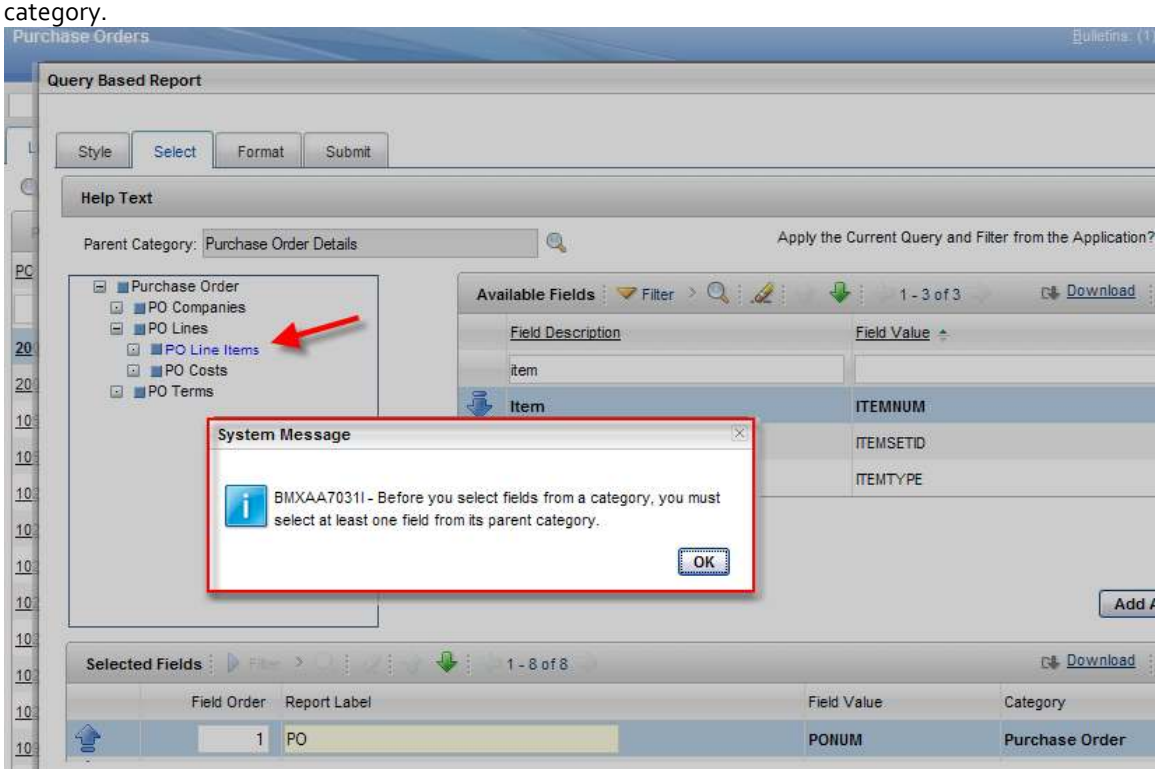
Additionally, as the user builds the report he can change the Column Order and/or Report Label. To do this, the user types in the new value directly in the applicable text box. After changes are made, the user must click refresh in order for any column order changes to take effect. The order that the columns are displayed after the refresh will always be in ascending order of the Column Order field.

An example of why a user may want to change the Report Label is when fields from different categories have the same Field Value. (For example, Description exists in both Asset and Work Order). The user creating the report may want to distinguish these two fields by over-riding the two Default Description Values with something like Asset Description and/or WO Description.

Notes on Selecting Fields:
1. When adding fields to the Selected Field category, the user must follow the hierarchical structure of the ROS. The user has to follow the relationships of selecting fields from the parent, the child and then the grandchild. A user can not add fields from the parent and then the grandchild as there is no direct relationship between these two.

This is shown in an example using the PO application. In this case, the Parent is PO, child is PO Lines, and grand child is PO Line Items. The user tried to directly select fields from the grandchild before selecting any from the child, and received the error below

BMXAA70311: Before you select fields from a category, you must select at least one field from its parent category.



2. This same logic applies for removing fields from the Selected Field section. Using the example above, if fields are added to the Selected Field section from the

PO Parent
PO Lines Child
PO Line Items Grandchild

The user can *not* remove fields from the PO Lines Child and only leave fields from PO Parent and PO Line Items Grandchild.

PO Parent
~~PO Lines Child~~
PO Line Items Grandchild

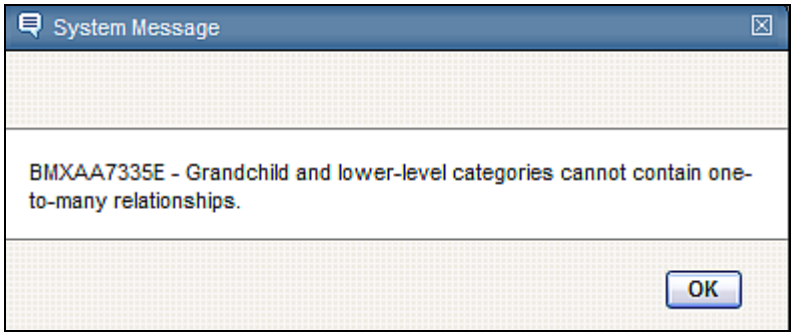
Deleting the parent from the report causes an error because there is no direct relationship between the PO Parent and the PO Line Items Grandchild. At least one field from the PO Lines Child must be maintained in the report to build the table relationships correctly.

3. Within a Child, Grandchild or lower level Category, any numbers of 1:1 Relationships can be chosen. For example, within the PO application, the user selects the PO Lines Child Category. Within the PO Lines, the user can select from as many grandchildren categories that he chooses (like PO Line Items) that he likes as long as they are 1:1 relationships.

However, within a lower level category, no 1:N Relationships can be chosen. Using the same example as above, the user selects the PO Lines Child Category. Within the PO Lines Child Category, the user can not select grandchildren categories with 1:N Relationships – like PO Line Receipts or PO Line Returns.

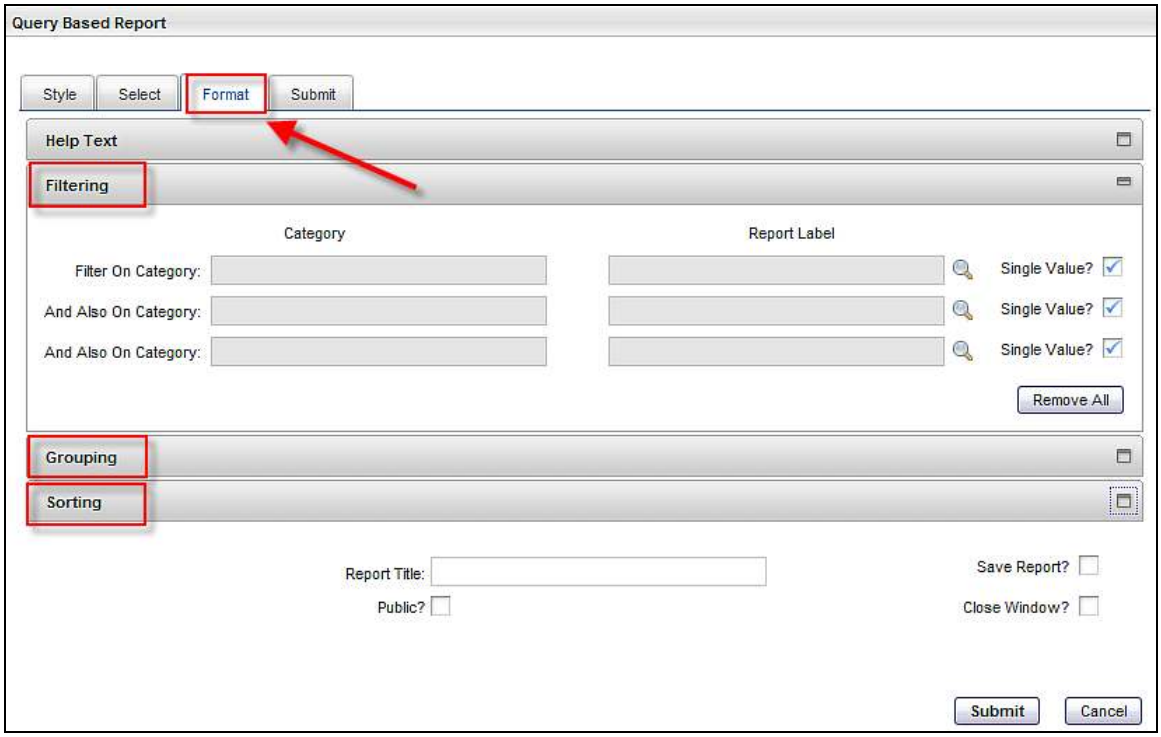
If the user tries to add fields from categories with 1:N Relationships at a Grand child level, the error message shown below will display.

BMXAA7335E – Grandchild and lower-level categories cannot contain one-to-many relationships.



2.4 Create QBR: Format Tab

Once the user has all the fields he wants in his Ad Hoc Report, he may go to the third tab, Format, shown below.



*Note: In this screenshot, the grouping and sorting sections are collapsed.

The format tab is optional. There is no requirement that this functionality be used in a QBR reports. Additionally, there are no requirements that Filtering, Grouping and Sorting must be done together.

- The user can choose to Group his report only, or Sort his report only.
- If the user chooses both Grouping and Sorting, either function can be defined first. For example, Grouping does not have to be defined before Sorting.

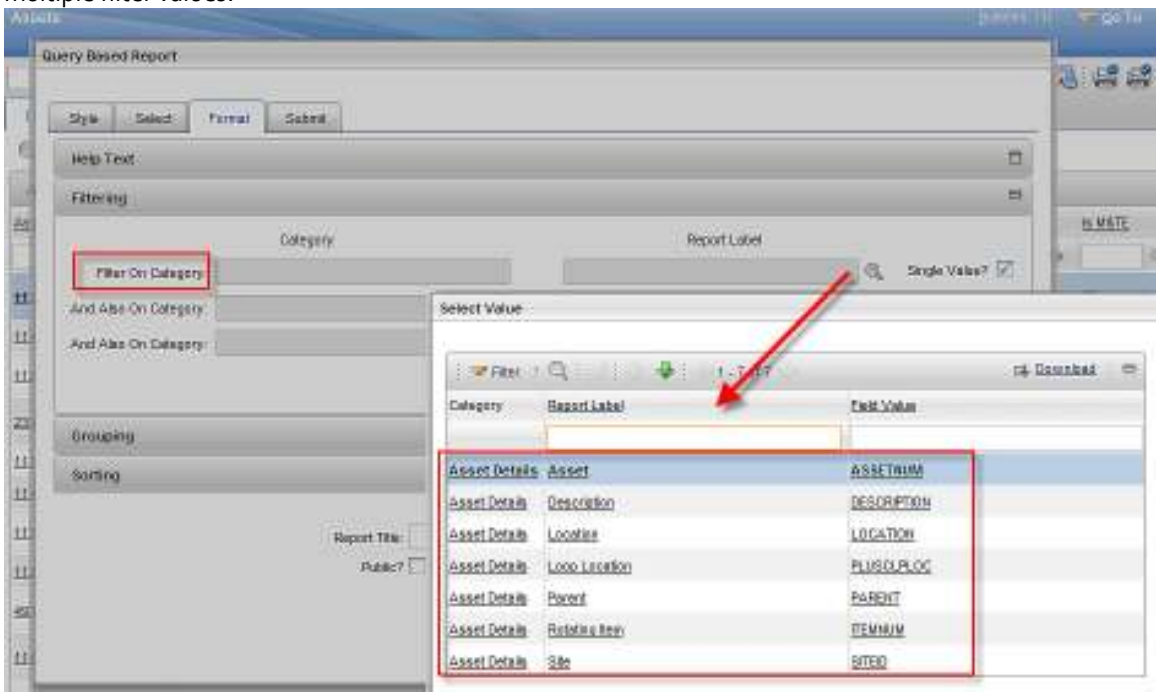
To enable filtering, grouping and sorting however, the user must first have added columns to his QBR report in the Select Tab. If the user tries to filter, sort or group before adding any fields in the Select Tab, no fields will be available to him to choose from.

2.4.1 Format Tab: Filters

The first section under the Format tab is filters. This enables users to specify filters or parameter values that can be dynamically applied at run time. The terminology 'filters' is used as client feedback indicated more users were familiar with filters than parameters.

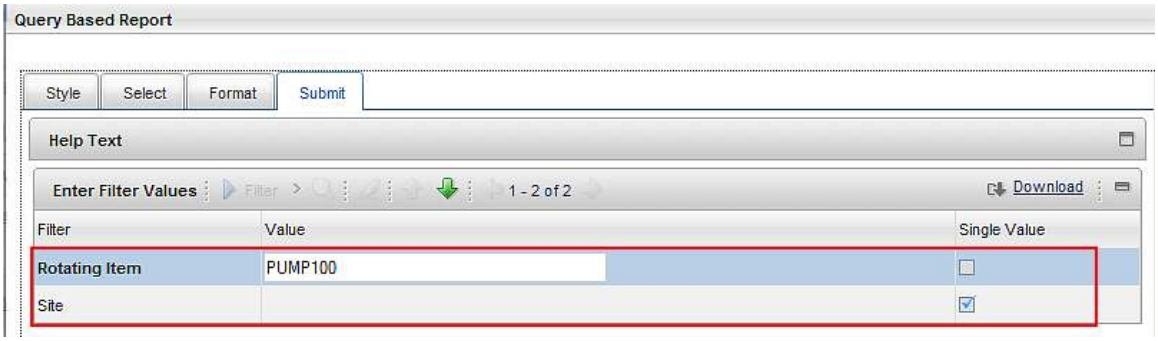
Filters can be added on any selected field from the application's main table, which the user has chosen via the Select' QBR Tab. This insures that the selected filters are bound parameters.
Note: Fields from other child tables cannot be selected as filters. This restriction prevents unexpected results from occurring due to the binding of a child parameter via its parent-child maxrelationship.

To specify a filter, the user clicks on the lookup, and a window displays listing all of the available fields that the user can specify as a filter. Three filters can be selected, and the user can specify single or multiple filter values.



The user selects Rotating Item and Site as filters as shown here.

To verify if these are the correct filters for this report, the user can click on the Submit tab. He can then optionally enter filter values, and execute the Ad Hoc report. If these are the correct values – and there are no other changes he wants to make to the report – he can save the report. If the user still wants to make some modifications, he simply goes back to the Style, Select or Format Tabs to continue building his report.

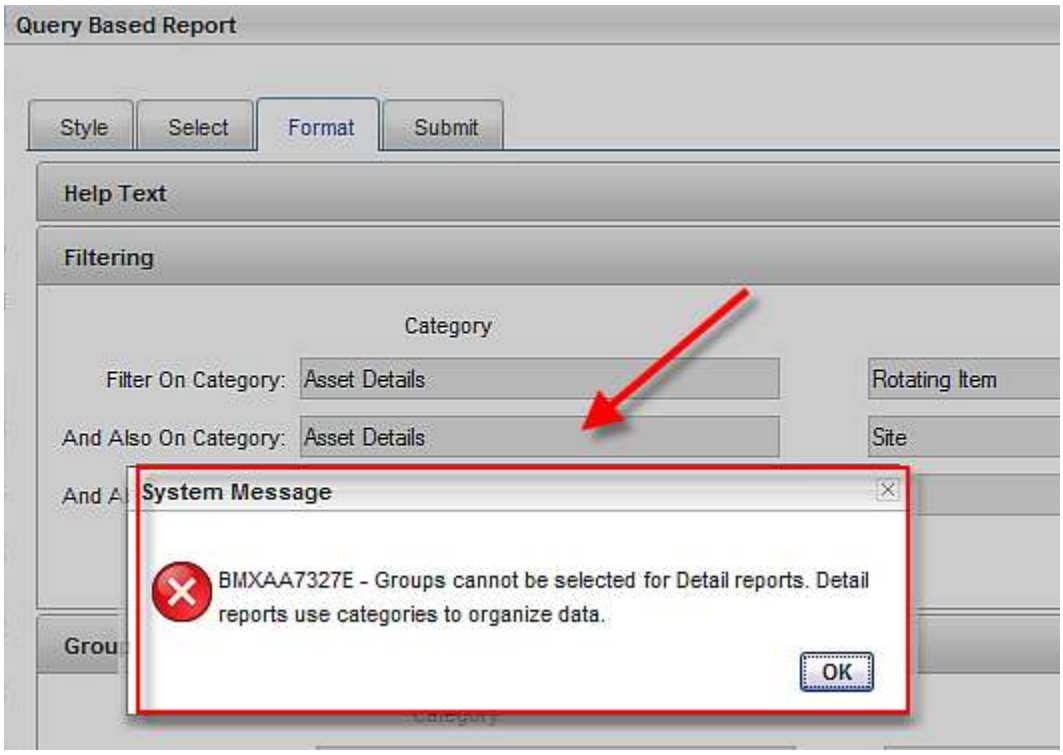


2.4.2 Format Tab: Grouping

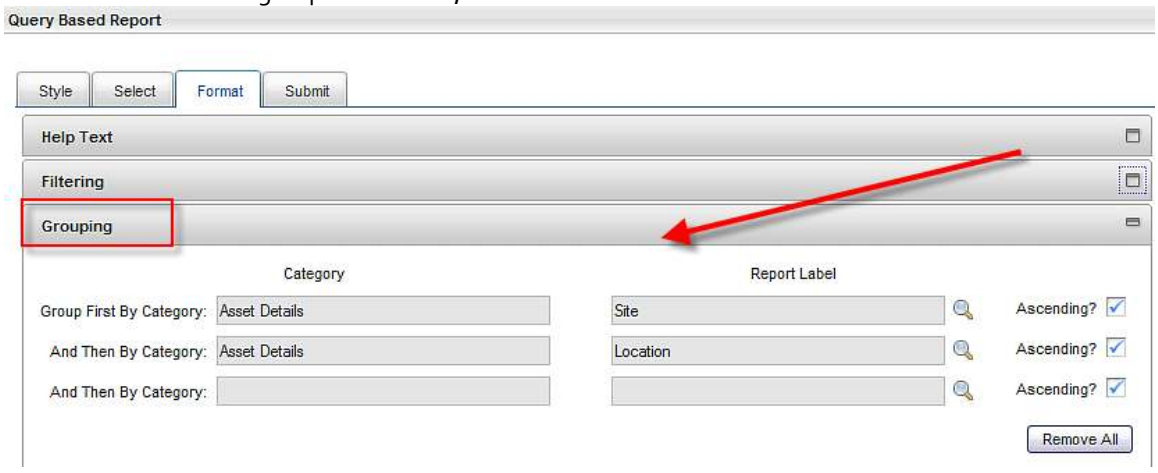
This second section in the Sort Tab is Grouping. Within this section, the user is able to select three Groupings. These are all optional and are defined as Null by Default.

Grouping is only enabled for Summary reports. If the Report type is Detail, user selected grouping is not enabled. Instead, groupings are defined for Detail reports via their selected categories. If you try to enable grouping for a Detail report, the message below will display.

BMXAA7237E – Groups cannot be selected for Detail reports. Detail reports use categories to organize data.



Groupings for Summary reports are defined via the Grouping Lookup. This is a dynamic lookup which only displays the user's Selected Fields. In the example below, a new QBR Summary report is initiated. And the user selects to group first on Site, and then location.



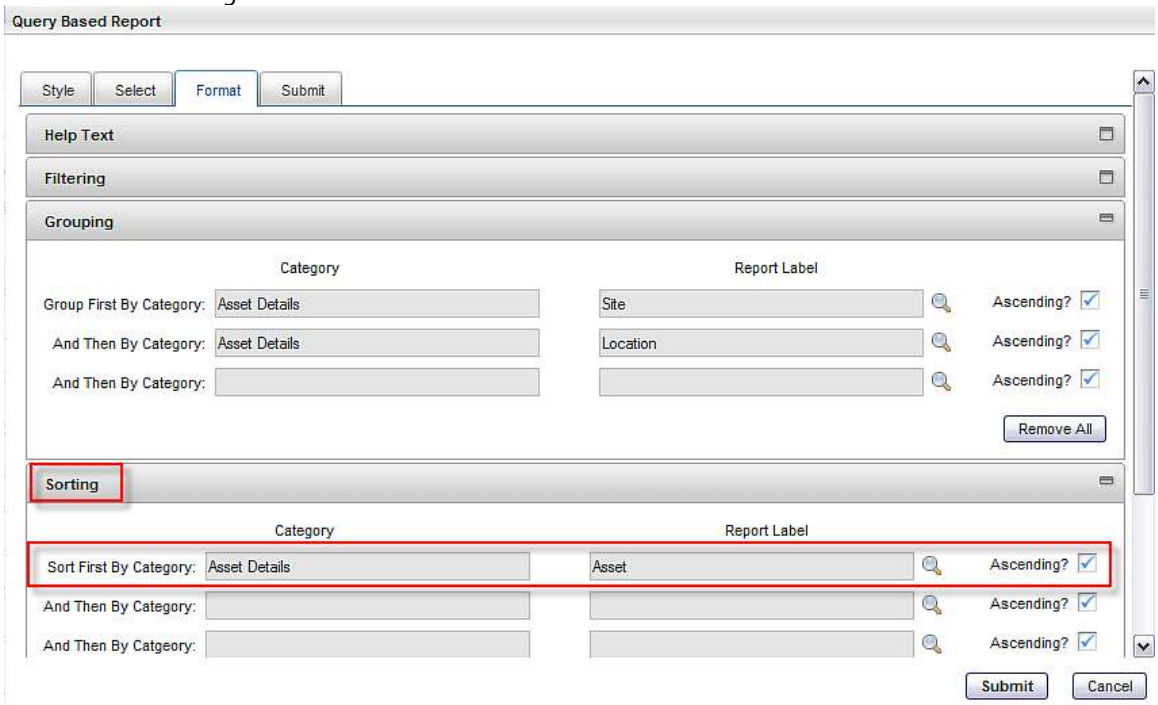
2.4.3 Format Tab: Sorting

The third section in the Format Tab is Sorting. Sorting will follow the same process as grouping, meaning that the fields to be sorted will be selected off the lookup. If two fields need to be sorted, the user will access the lookup twice.

The Label lookup that is used in the Sorting and Grouping section is the same.

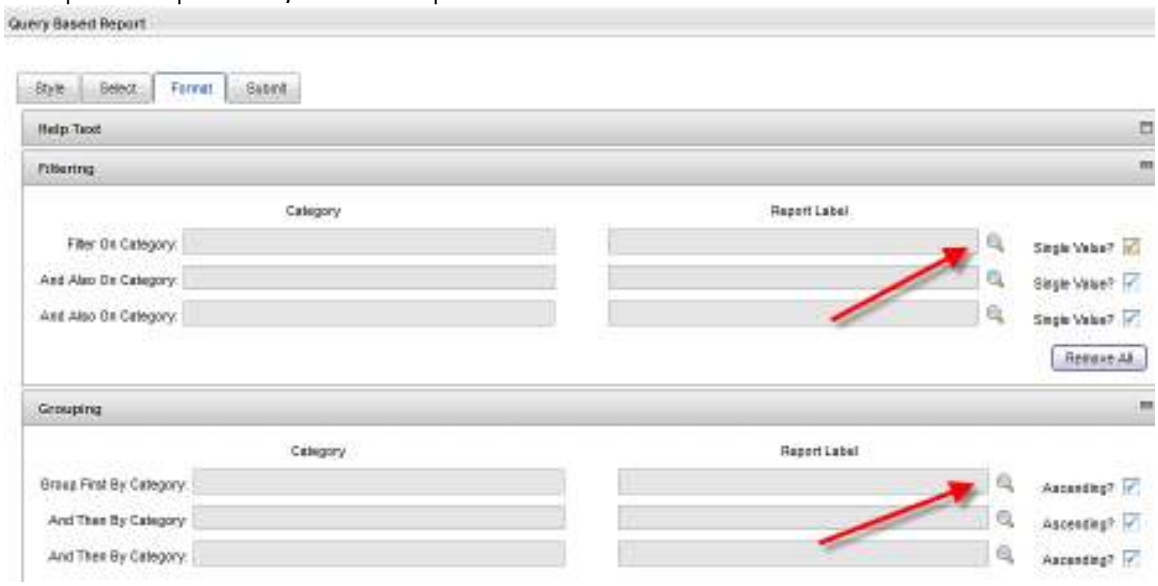
> The first sorting must be defined before the second. To enable this, the second sorting will be disabled until the first grouping is enabled.
>
> Users can add the same field in the Sorting section that is in the Grouping section. If the user specifies a different sort order than the grouping order specifies, the grouping order will take precedence.

In the example below, after the grouping was defined, the user selected to sort the data by Asset Number in ascending order.



2.4.4 Format Tab: Miscellaneous Notes

1. The format lookups are derived dynamically for each unique QBR. Additionally, the fields in the lookup are non-persistent, so the lookup and values are not sort-able or filter-able.



2. When accessing the lookup, you can select a single value for filtering or grouping only.

3. Three filtering, grouping and sorting selections are enabled out of the box. This was determined after analyzing the numbers of groupings and sorting on hundreds of reports, along with balancing client input in minimizing the complexity of the QBR functionality. If this does not meet your individual needs, the code can be customized.

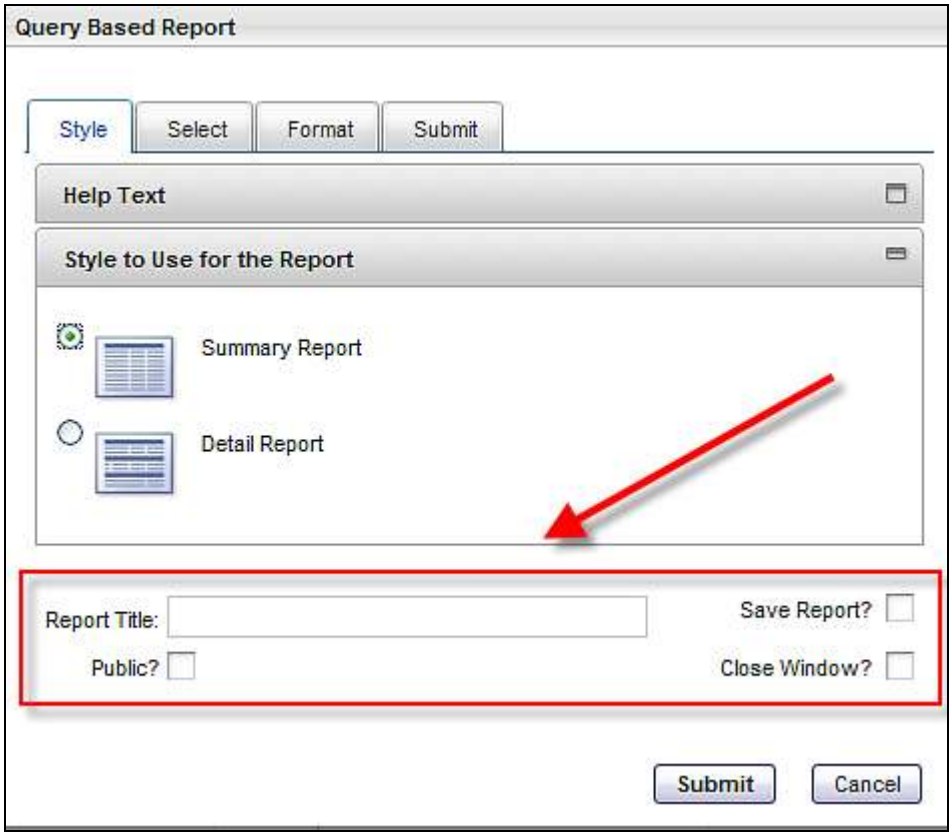
4. When defining groupings, the user must first define the first grouping before the second one can be defined. To enable this, the second grouping field is disabled until the first grouping is applied.
   - The same process applies for the third grouping, meaning the first two groupings must be defined before defining the third.

5. Grouping of report data is an extremely valuable concept which enables the user to view data quickly and concisely in multiple categories. However, certain data types lend themselves more to groupings than others. For example, data fields with time values may be very difficult to group on due to the wide variation in time values.
   a. However, to enable flexibility, no limitations have been placed on the data types that can be grouped.

6. The sort order from an application is not applied to the QBR report. If the user wants to apply the same sort order that was enabled in the application, he must explicitly specify it on the Format tab within the QBR window.

2.5 Common Fields – All Four Tabs

Before reviewing the Schedule tab, it is important to review the bottom fields that are displayed on all four Tabs. These fields are optional and can be used if a user wants to save the QBR report for execution in the future.

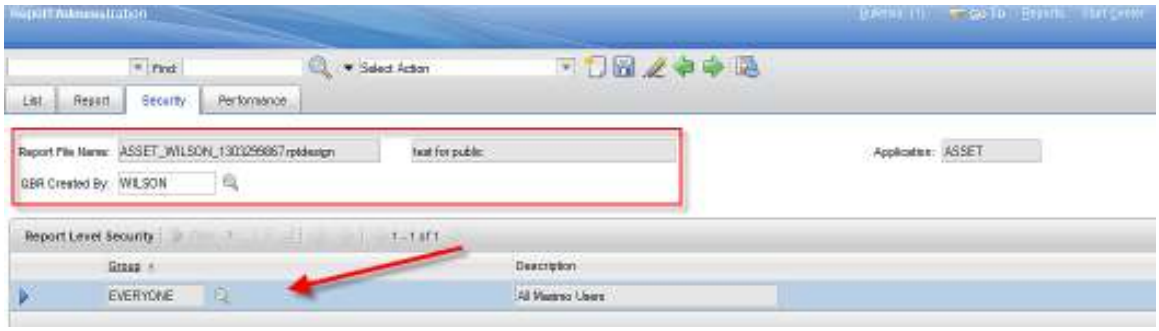


1. Report Title: This is the Report Title that appears at the top of the report. It is also the text displayed to the user when he clicks 'Run Reports'. It is typically a value that is meaningful to the user, and enables him to quickly identify the report in the future.

- By default, this value is null.
- It is identified in the database as REPORT.DESCRIPTION.
- If the user enables the Save Report? Field, the Report Title is required.

2. Public? The public field enables the user to grant other users access to execute the Ad Hoc report they have created. It empowers the QBR creator to share his report with others – and he does not have to request the administrator to grant security access to other groups.

The default value for Public is no or de-selected. If the user enables the Public? Field, the Save Report? Field must be enabled. Enabling the Public? Field creates report access in the REPORTAUTH Table for the EVERYONE Security Group.



If the Public Access is given, members of the EVERYONE Security group will be able to execute the QBR report. They can not delete the Ad Hoc report. The only users who can delete the Ad Hoc report are (1) The user who created it and (2) The administrator

Important Notes on Public Functionality

A. You may not want to grant each security group access to the Public Field for their QBR reports. Therefore, a Security Action is available, to specify which Security Groups can Enable Create Report Public Access. More details on this can be found in the security section titled 'QBR Security Access – Functionality'.

B. Beginning with Version 7.5.0.1 release, the REPORTADHOCLOC cron task was introduced for multiple language environments only. In these multiple language environments, when you save a public QBR, the SYNCHREPORTLABELS maxvar is set to true.

Then, when the REPORTADHOCLOC cron task is executed, it will check the SYNCHREPORTLABELS maxvar, synchronize the labels and set the maxvar flag back to false on completion.

3. Save Report? This field enables the user to save his unique report inputs as a report design file. A user may need to execute his unique report on a regular basis, or may want to share it with others. In these cases, the user should save his report. Saving the report creates multiple database table entries including:

1. REPORT, for report details including the userid of the person creating the QBR
2. REPORTDESIGN for the created XML Report Design File
3. REPORTLABEL with the Report File Name and Column Descriptions
4. REPORTDEPEND for the Maximo System Libraries
5. REPORTLOOKUP if the report contains filters or parameter values
6. REPORTADHOC to enable future editing of the QBR
7. REPORTADHOCFIELD to enable future editing of the QBR selected fields

In other cases however, a user may only need a report for a single instance. In this case, the user will choose to not Save their report.

4. Close Window?: The Close Window? field enables the user to close all report dialogs after executing his report.

- If the user enables the Close Window? field, after the report executes in the browser, the user will return to his application and all reporting windows will be closed.
- If the user does NOT enable the Close Window? field after the report executes in the browser, the user will return to his application and the QBR window and report window will still be open.

The default for Close Window is (1) Enabled if the User saves his Report (2) Not-Enabled if the User does not save his report.

Buttons:

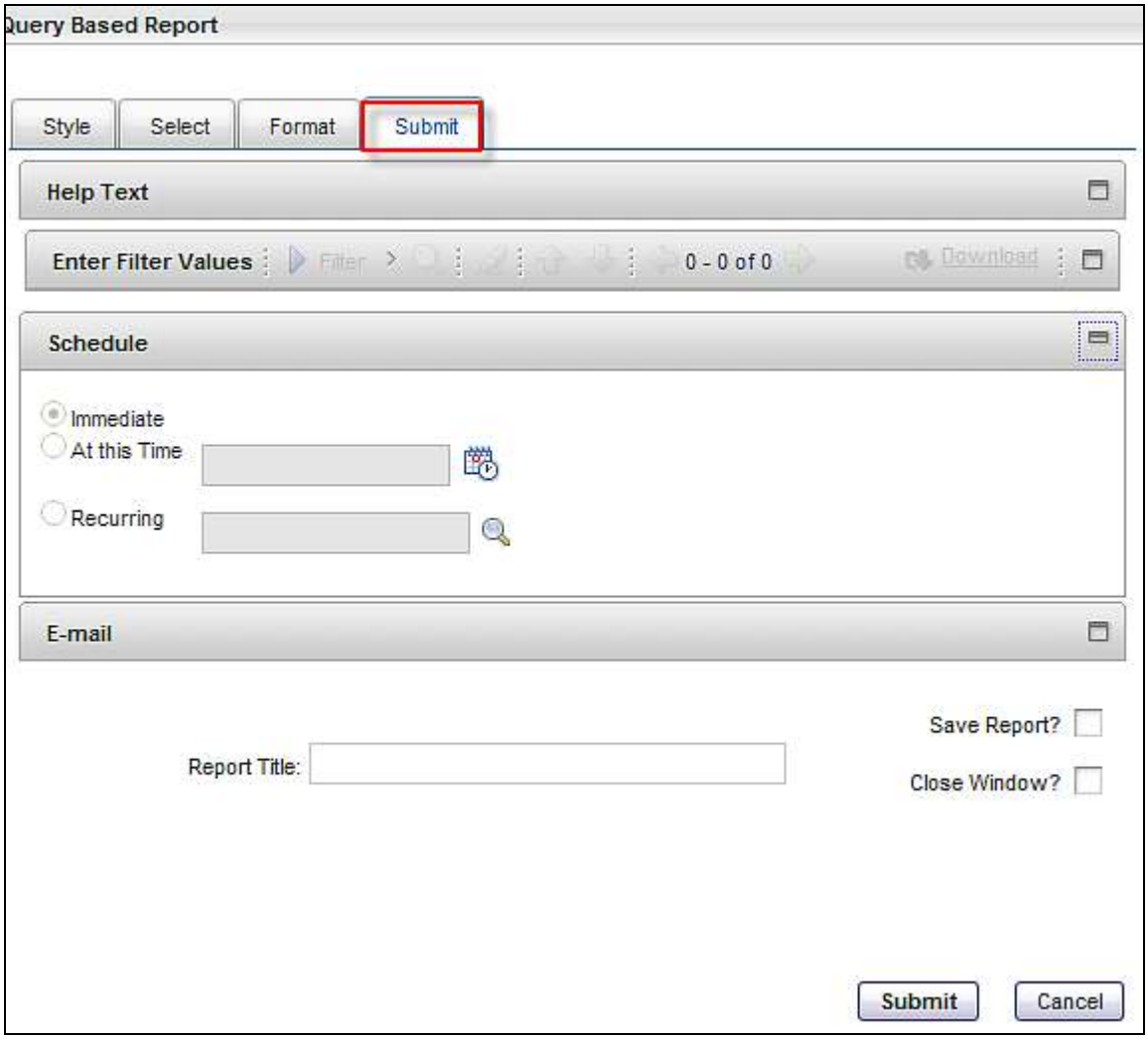
At the very bottom of each page are two buttons: Cancel and Submit. At any time during the QBR creation process, the user can select either of these buttons.

CANCEL: Cancels the report creation process and closes the QBR window.

SUBMIT: Takes the report inputs and passes them to the report engine. The report will be generated and displayed to the user in HTML.

2.6 Create QBR: Submit Tab

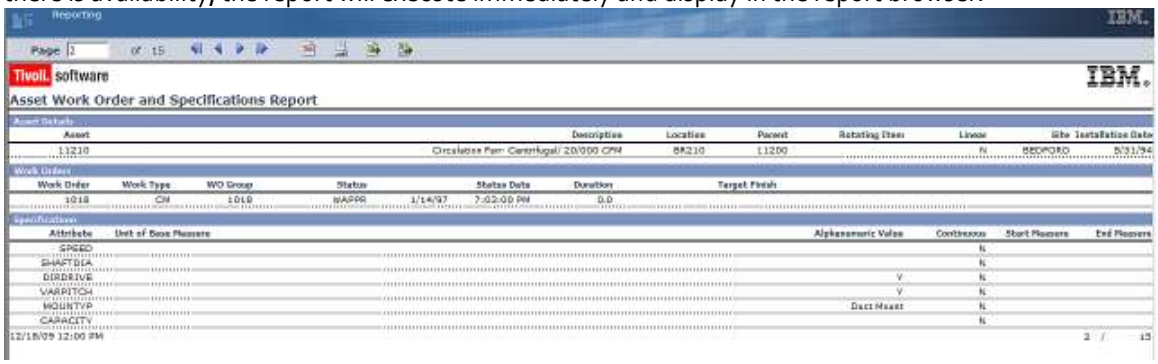
The last tab in the QBR window is the Submit Tab. It contains any defined filters (parameters), along with scheduling and emailing options. This is essentially a Report's Request Page – as it collects the user's run time inputs before executing the report. The immediate Run Option is displayed by default.



2.6.1 Schedule Tab: Run Report Immediately

Continuing with our example, the user names his report 'Asset Work Order and Specifications Report', and saves the report.

He chooses to run the report immediately and the report queue is checked to see if there is availability. If there is availability, the report will execute immediately and display in the report browser.



Note: For more information on Report Queuing, reference the Report Feature Guide.

When the report is displayed in the browser, standard reporting functionality is available. This includes converting to pdf for saving or printing, downloading to other file formats or navigating through the pages of results. If the report is converted to pdf, it displays as follows:

Tivoli. software IBM.

Asset Work Order and Specifications Report

Asset Details

Asset	Description	Location	Parent	Rotating Item	Linear	Site	Installation Date
11210	Circulation Fan- Centrifugal/ 20/000 CFM	BR210	11200		N	BEDFORD	5/31/94

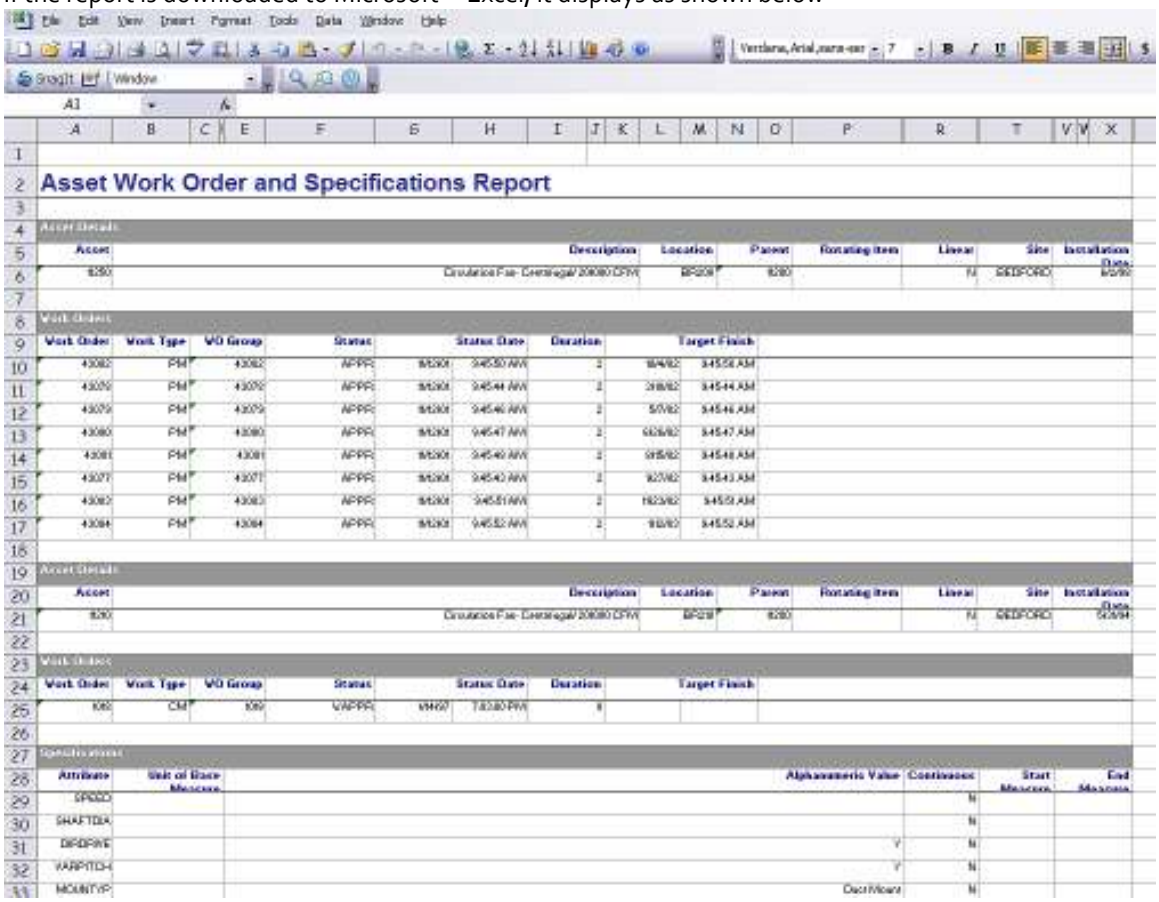
Work Orders

Work Order	Work Type	WO Group	Status	Status Date	Duration	Target Finish
1018	CM	1018	WAPPR	1/14/97 7:03:00 PM	0.0	

Specifications

Attribute	Unit of Base Measure	Alphanumeric Value	Continuous	Start Measure	End Measure
SPEED			N		
SHAFTDIA			N		
DIRDRIVE		Y	N		
VARPITCH		Y	N		
MOUNTYP		Duct Mount	N		
CAPACITY			N		

If the report is downloaded to Microsoft® Excel, it displays as shown below



Asset Work Order and Specifications Report

Asset	Description	Location	Parent	Rotating Item	Linear	Site	Installation Date
8280	Circulation Fan- Centrifugal 20000 CFM	BP208	8280		N	BEDFORD	6/5/96

Work Order	Work Type	WO Group	Status	Status Date	Duration	Target Finish
43082	PM	43082	APPR	8/1/08	1	8/4/08
43076	PM	43076	APPR	8/1/08	1	8/4/08
43079	PM	43079	APPR	8/1/08	1	8/7/08
43080	PM	43080	APPR	8/1/08	1	8/26/08
43081	PM	43081	APPR	8/1/08	1	8/8/08
43077	PM	43077	APPR	8/1/08	1	8/27/08
43083	PM	43083	APPR	8/1/08	1	8/23/08
43084	PM	43084	APPR	8/1/08	1	8/8/08

Asset	Description	Location	Parent	Rotating Item	Linear	Site	Installation Date
8280	Circulation Fan- Centrifugal 20000 CFM	BP208	8280		N	BEDFORD	6/5/96

Work Order	Work Type	WO Group	Status	Status Date	Duration	Target Finish
8280	CM	8280	UNPPR	8/4/07	8	

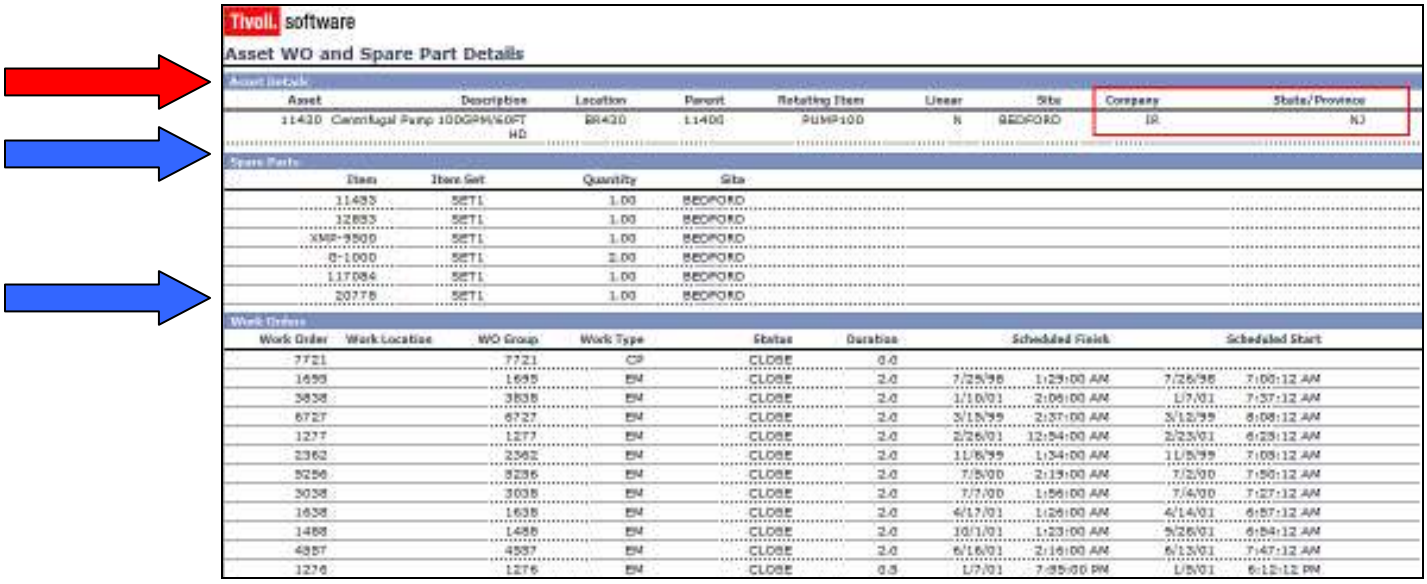
Attribute	Unit of Base Measure	Alphanumeric Value	Continuous	Start Measure	End Measure
SPEED			N		
SHAFTDIA			N		
DRIVEWE		Y	N		
FANPITCH		Y	N		
MOUNTYP		DuctMount	N		

QBR report Display: Detail Reports

If a detail report is selected, it includes a header section containing (1) selected fields from the Parent Category for the individual record and (2) selected fields from categories with Single Relationships.

After the header section, the Category Sections display. These are the categories having Multiple Relationships. Their subheader text label is its Category Description or Application Name.

In the example below, an Asset WO and Spare Part Details report is created.



Tivoli. software

Asset WO and Spare Part Details

Asset	Description	Location	Parent	Rotating Item	Linear	Site	Company	State/Province
11410	Centrifugal Pump 100GPM/60FT HD	BR410	11400	PUMP100	N	BEDFORD	IR	NJ

Item	Item Set	Quantity	Site
11483	SET1	1.00	BEDFORD
12883	SET1	1.00	BEDFORD
XMP-9500	SET1	1.00	BEDFORD
B-1000	SET1	2.00	BEDFORD
117084	SET1	1.00	BEDFORD
30778	SET1	1.00	BEDFORD

Work Order	Work Location	WO Group	Work Type	Status	Duration	Scheduled Finish		Scheduled Start	
7721		7721	CP	CLOSE	0.0				
1695		1695	EM	CLOSE	2.0	7/25/98	1:23:00 AM	7/25/98	7:00:12 AM
3838		3838	EM	CLOSE	2.0	1/10/01	2:06:00 AM	1/7/01	7:57:12 AM
6727		6727	EM	CLOSE	2.0	3/15/99	2:37:00 AM	3/12/99	8:08:12 AM
1277		1277	EM	CLOSE	2.0	2/26/01	12:54:00 AM	2/23/01	6:23:12 AM
2362		2362	EM	CLOSE	2.0	11/8/99	1:54:00 AM	11/5/99	7:05:12 AM
5256		5256	EM	CLOSE	2.0	7/5/00	2:19:00 AM	7/2/00	7:50:12 AM
3038		3038	EM	CLOSE	2.0	7/7/00	1:56:00 AM	7/4/00	7:27:12 AM
1638		1638	EM	CLOSE	2.0	4/17/01	1:26:00 AM	4/14/01	6:57:12 AM
1488		1488	EM	CLOSE	2.0	10/1/01	1:23:00 AM	9/28/01	6:54:12 AM
4587		4587	EM	CLOSE	2.0	6/16/01	2:16:00 AM	6/13/01	7:47:12 AM
1276		1276	EM	CLOSE	0.5	1/7/01	7:35:00 PM	1/5/01	6:12:12 PM

The header section is highlighted by the red arrow. It contains selected fields from the parent Asset table. It also contains selected fields a single relationship table, Company, highlighted within the red box.

The category sections are displayed next highlighted by the blue arrows. These contain the selected fields from other multiple relationship categories.

If the QBR has no data within a category section (subheader), it will not display that category. For Asset 11250 below, it does not have any spare parts associated with it, so the spare parts section is not displayed.

Tivoli. software

Asset WO and Spare Part Details

Asset Details

Asset	Description	Location	Parent	Rotating Item	Linear	Site	Company	State/Province
11250	Circulation Fan- Centrifugal/ 20/000 CFM	BR200	11200		N	BEDFORD	TRN	WI

Work Orders

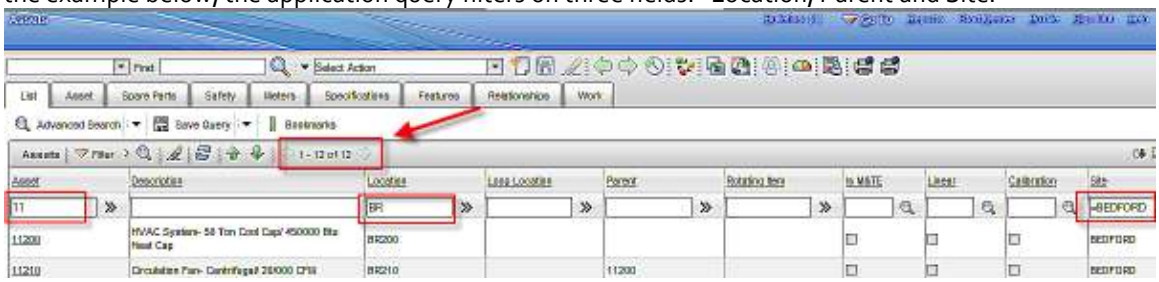
Work Order	Work Location	WO Group	Work Type	Status	Duration	Scheduled Finish		Scheduled Start	
43082		43082	PM	APPR	2.0	10/4/02	9:45:50 AM	10/3/02	9:45:50 AM
43078		43078	PM	APPR	2.0	3/18/02	9:45:44 AM	3/17/02	9:45:44 AM
43079		43079	PM	APPR	2.0	5/7/02	9:45:46 AM	5/6/02	9:45:46 AM
43080		43080	PM	APPR	2.0	6/26/02	9:45:47 AM	6/25/02	9:45:47 AM
43081		43081	PM	APPR	2.0	8/15/02	9:45:48 AM	8/14/02	9:45:48 AM
43077		43077	PM	APPR	2.0	1/27/02	9:45:43 AM	1/26/02	9:45:43 AM
43083		43083	PM	APPR	2.0	11/23/02	9:45:51 AM	11/22/02	9:45:51 AM
43084		43084	PM	APPR	2.0	1/12/03	9:45:52 AM	1/11/03	9:45:52 AM

12/11/09 12:01 PM

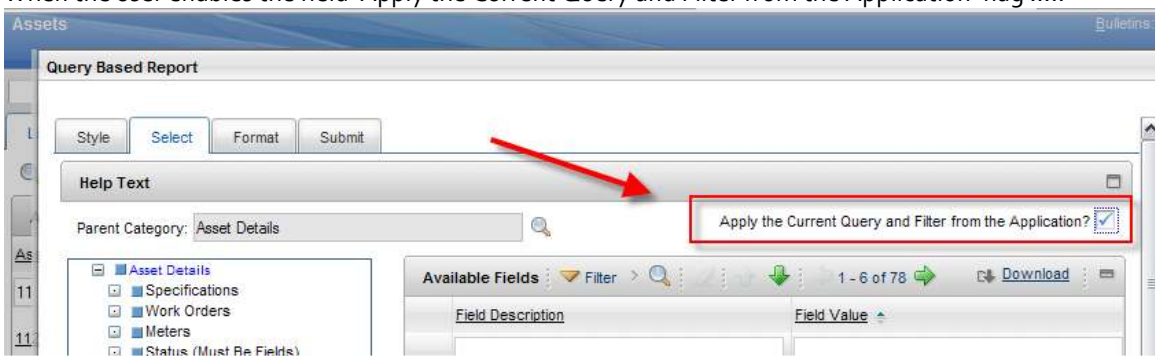
2.6.2 QBR report Display: Saved and Dynamic Where Clauses

To enable visibility on the query that the Ad Hoc report executes against, the report's saved and dynamic query filters are displayed on the last page of the QBR report. This display is made up of two parts: (1) Saved Where Clause and (2) Dynamic Where Clause.

The Saved Where Clause contains the application's query and/or filter that was applied to the report. In the example below, the application query filters on three fields: Location, Parent and Site.



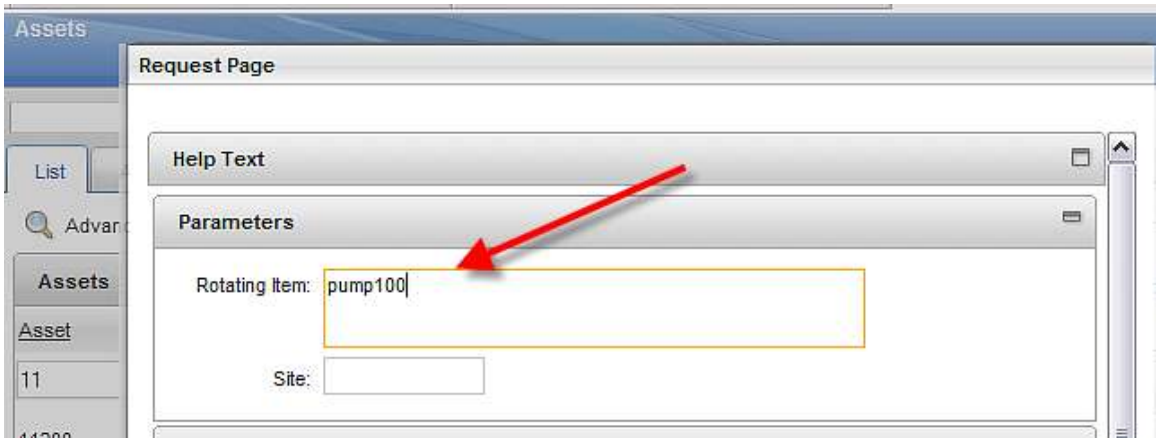
When the user enables the field 'Apply the Current Query and Filter from the Application' flag .....



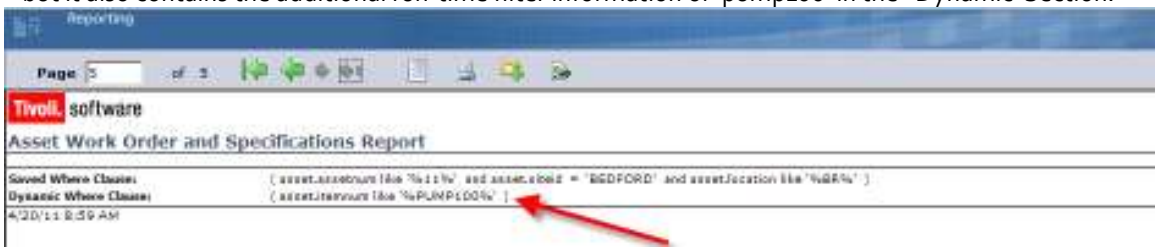
The application query is applied to the report and the saved report will always execute against that query. This saved query is displayed in the 'Saved Where Clause'.

The Dynamic Where Clause displays any additional filters that are displayed at run time. These filters can come from User Inputted Filters defined for the Report on the Format Tab. Site or Data Restrictions can also be included in the report's dynamic where clause, but these are not displayed on the report.

Continuing with this example, the QBR report that was created had two filters: Rotating Item and Site. When the report is executed from the Action Menu, the user inputs a parameter value of 'pump100' at run time.



Then, when the report is executed, it displays the same information in the 'Saved Where Clause' Section – but it also contains the additional run-time filter information of 'pump100' in the "Dynamic' Section.



2.6.3 QBR report Display Notes

1. When a QBR report is created, the user's individual report inputs are taken and converted to an xml design file via an API. This on-the-fly report design file utilizes the standard report formatting, including colors, fonts, logos and page formatting. This information is applied to the report design file via the Maximo System Library.

2. If a QBR report is converted to pdf, it will display in landscape format, like the other Enterprise Out of the Box reports.

3. As noted earlier, there are no limitations to the number of fields a user can add to the report. A user can choose to only include a single data field, or a very large number like 50 fields.

If the user adds a very large number of database fields to their QBR report, this will require the user to scroll through the Report Browser to view all of their fields.

Additionally, if the user prints or exports the report with a very large number of fields to Adobe, all of the report fields will *not* display due to the fixed page size of Adobe.

Also, if the user prints the report to HTML, the user will need to scroll through the Report Browser to view all of their fields – but only a portion of those fields will print in HTML due to the fixed page size.

However, these constraints should not limit users to a minimum number of fields because many users want a large number of fields to export to .csv or .xls for additional data analysis. Since the export of data to .csv/.xls is a primary requirement of this functionality, the number of fields the user can utilize in his QBR report will not be limited by scrolling or incomplete display of data in pdf.

QBR report Design File Notes

When the user creates a Detail Ad Hoc Report, an .rptdesign file is created. If the user saves the report, the design file is saved to the database in the tables including REPORT, REPORTLOOKUP, REPORTDESIGN, REPORTLABEL and REPORTDEPEND.

If the user does not save the report, the design file is still generated, but discarded and removed from the application's temporary file location when the user logs out, or his session expires.

When the .rptdesign file is created, it contains information including
1. The maxrelationships used in the Report Object Structure to join objects together. These maxrelationships are converted to sql and used within the .rptdesign file.

2. All fields or objects referenced in the relationship are used in the .rptdesign file even if the user does not select each field in his Ad Hoc report.

To explain this, use an example of an Ad Hoc report for Work Order. In this example, the user selects fields from the Work Order Parent, and a few fields from the Multiple Assets, Locations and CIs child category. The maxrelationship that joins Work Order to Multiple Assets, Locations and CI is MULTIASSETLOCCI. The sql for this relationships is shown below.

ALLMULTIASSETLOCCI	recordkey=:wonum and recordclass=:woclass and worksiteid=:siteid

What this means is that the user does not have to select each field defined in the relationship (like woclass) to his Ad Hoc report to make the report execute properly. These fields are referenced in the .rptdesign file regardless if the user explicitly selects them or not.

2.6.4 QBR: Miscellaneous Report Title and Saving Notes

1. You can create multiple QBR reports with the same title, as long as they are created from different applications. You can not create multiple QBR reports with the same name in the same application.

For example, Bob creates a QBR report with the title 'Bob's Report' in the Asset application. He can create a different QBR report and save it with the same report title 'Bob's Report' as long as it is created from a different application.

If you try to create multiple QBR reports with the same title in the same application, you will receive the error message below.

BMXAA7044W – You are the owner of another Query Based Report in this application with the same Report title. Click OK to overwrite the existing report that has this title. Click CANCEL to exit this action and enter a new Report Title for the current report.

Since you are the owner of the original QBR report, you can choose to over-write the QBR report, or Cancel and enter a different report name.

2. QBR report titles must be also be unique with any other reports within the application.

For example, you have a report called 'Vendor Analysis' in the Purchasing application. If you then try to create a Ad Hoc report with the Report Title of 'Vendor Analysis' in the Purchasing application, you will receive the error below.

BMXAA7045E – Another report in this application has this Report Title. Specify a new Report title.

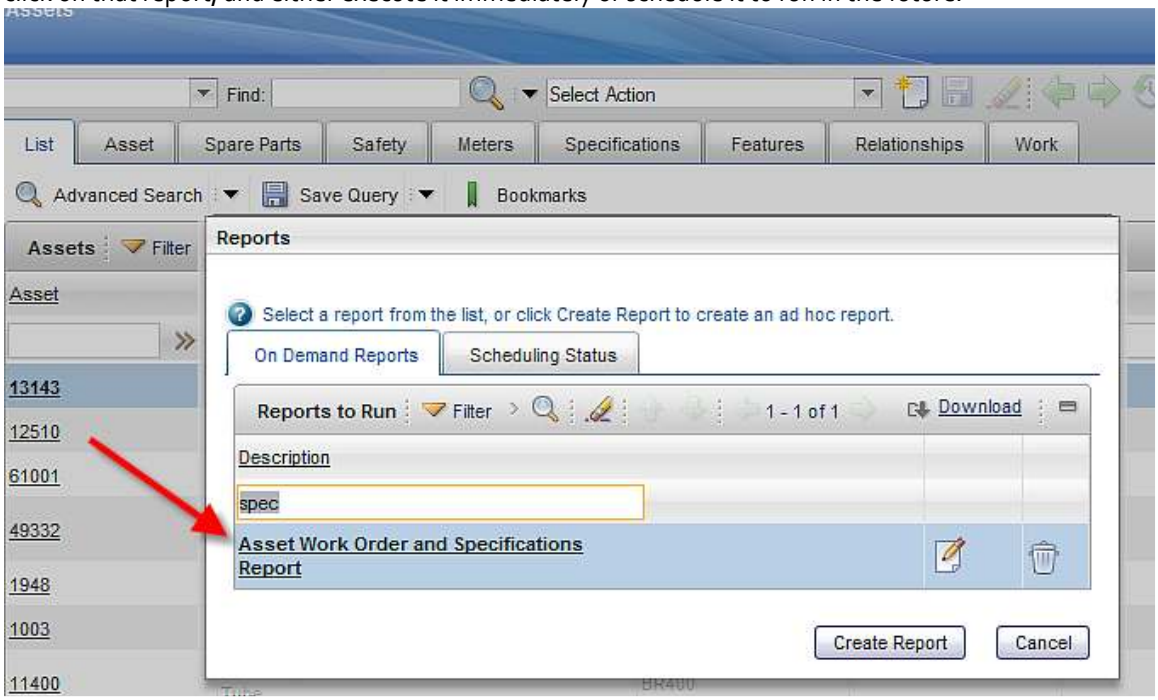
2.7 QBR report Execution

After the report is created, it can be executed immediately or scheduled to execute in the future. If the QBR report is saved, a number of different actions can occur including:

Executing from the Run Report Window
Enabling other security groups to access
Editing the QBR Report
Deleting the QBR when it is no longer required

2.7.1 Executing from the Run Report Window

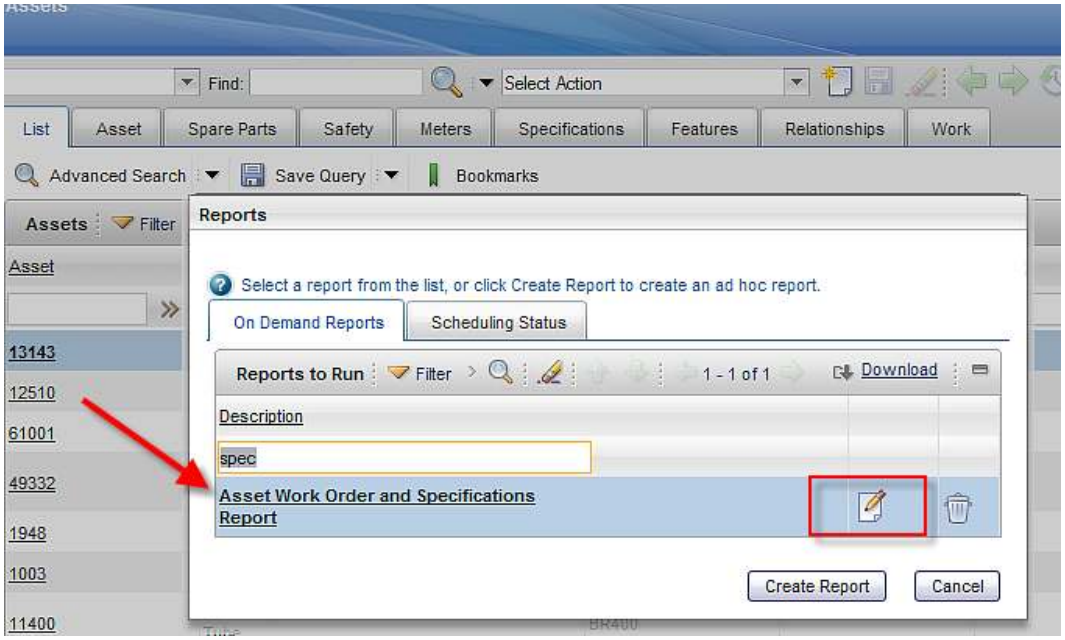
After the user creates and saves a QBR report, it is accessible to him from the Reports Window. This is shown below as the new QBR report 'Asset Work Order and Specification Report'. The user can then click on that report, and either execute it immediately or schedule it to run in the future.



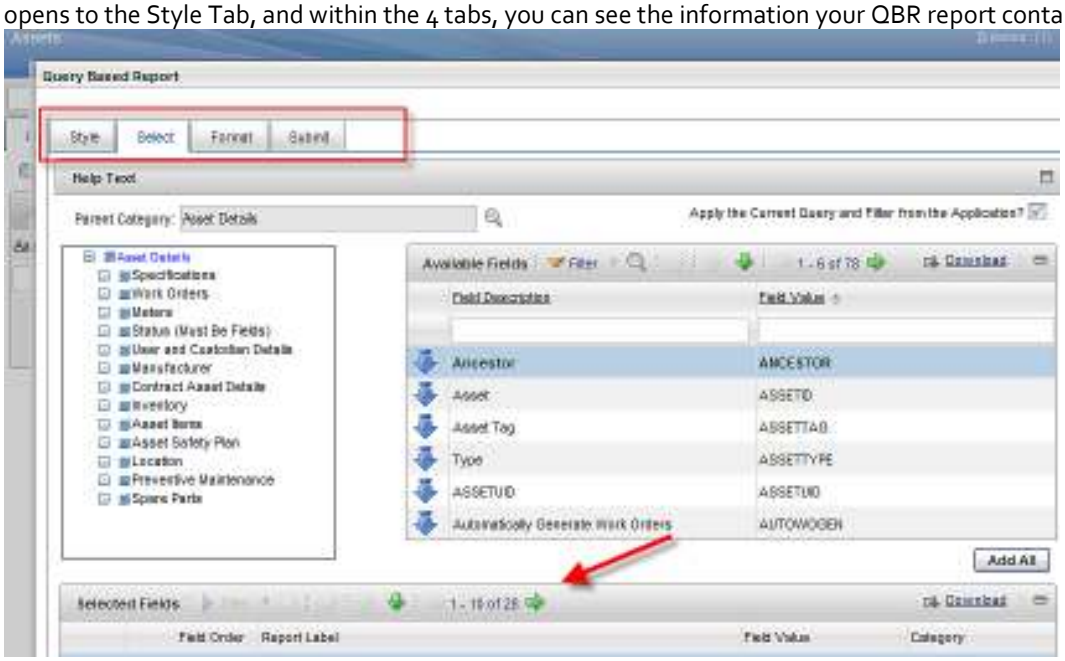
Additionally, since this user is the creator of the QBR report, it displays with both the Edit and Delete icons for the user also.

2.8 Editing the QBR Report

After you have created and saved your QBR report, you have the option to edit it by clicking on the Edit Pencil Icon next to the report name on the Run Report dialog. Only the user who created the ad hoc report has the ability to edit it.



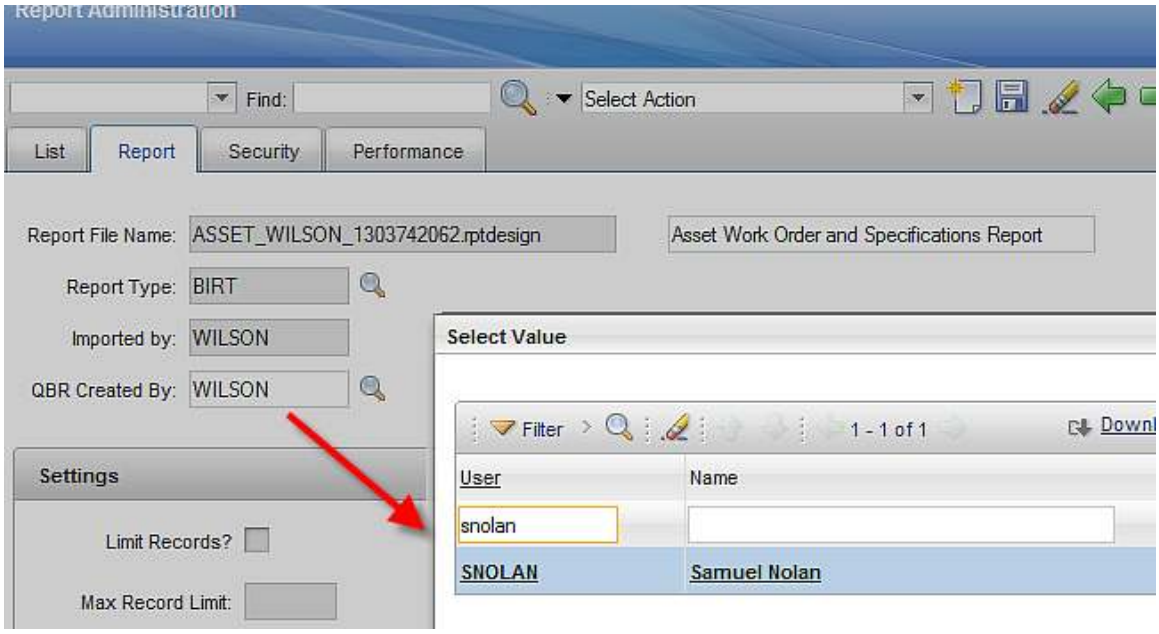
If you click on the Edit icon, the QBR report will open in the same dialog that you used to create it. It opens to the Style Tab, and within the 4 tabs, you can see the information your QBR report contains.



2.8.1 Who Can Edit QBR Report

As mentioned above, only the user who created the QBR can edit the QBR. No additional security access needs to be granted to the owner of the QBR report for editing.

The administrator cannot edit it – he can only delete it. However, if the QBR owner changes jobs or leaves the company, the administrator can reassign the owner of the QBR to another user. This can be done in the report administration application by changing the 'QBR Created By' field to the new owner.

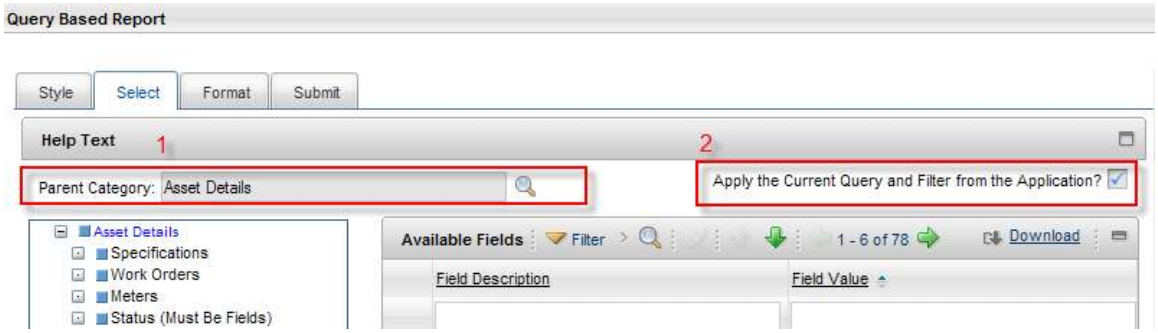


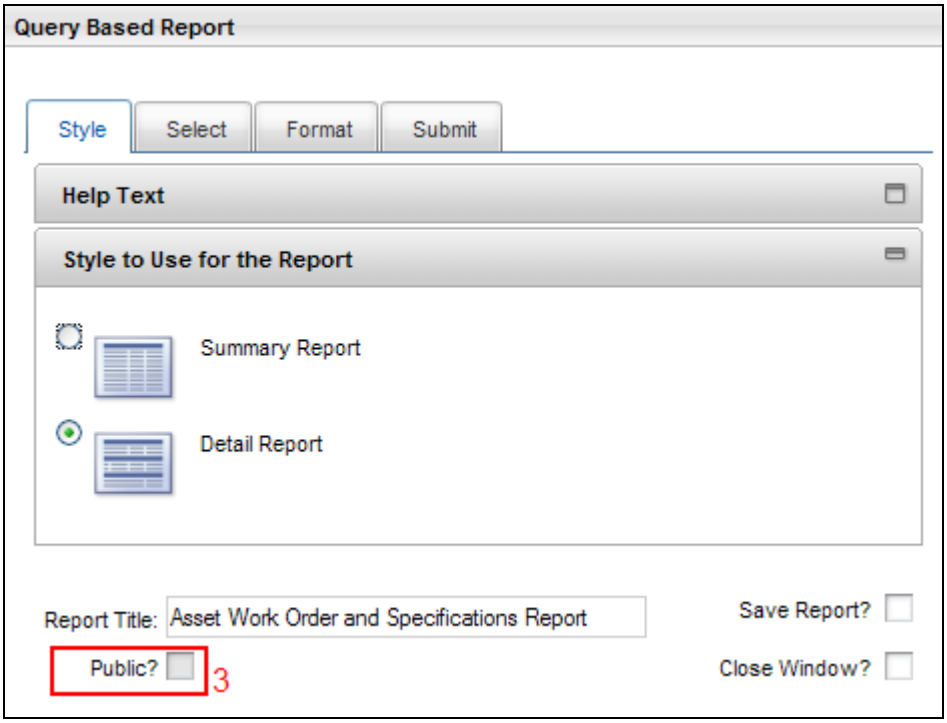
Fields Not Enabled for Editing

When you edit your report, you can change the style, add or remove fields, and modify the filters, sorting and grouping. However, there are 3 items you can not edit which are:

1. The Parent Category, or Report Object Structure of the QBR.
2. The selection of including or not including the Application Query
3. The selection of enabling or not enabling Public Security Access

These fields are not enabled for editing, and are grayed out as shown in the screen shots below.





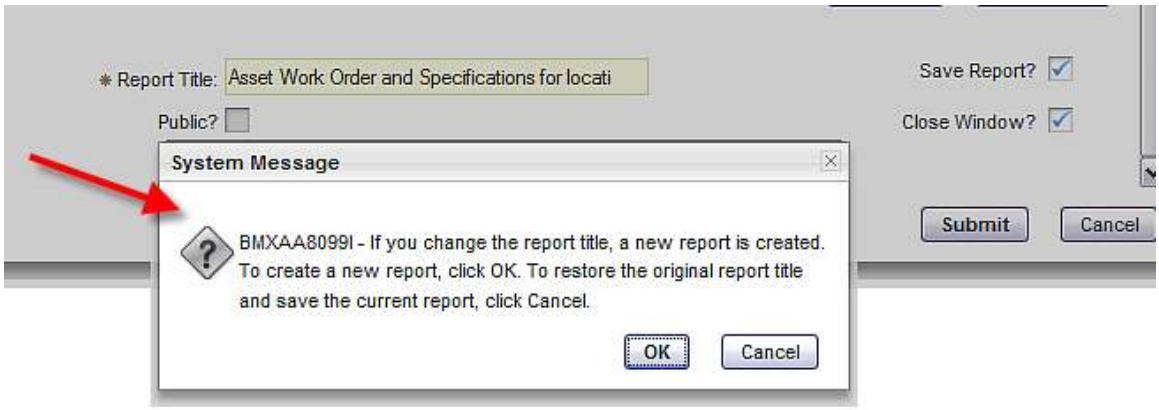
2.8.2 Saving QBR Edits

When you edit your QBR report, the Save Report functions the same as when you create a QBR report. This means that the Save Report? Functionality is not enabled by default. You can make your edit changes, view them in the browser first to verify them, and then go back to the QBR dialog to save them.

If you select 'Save Report?' and

Do not change the Report Title, the original QBR report will be overwritten with the new values.

Do change the Report Title, you will first receive a message that a new QBR report will be created. You then have the option to continue, or to cancel and utilize the original title.



Scheduling and Report Object Structure Impact

If you want to edit your QBR report, you may be impacted by business rules for scheduling and Report Object Structure Changes noted below.

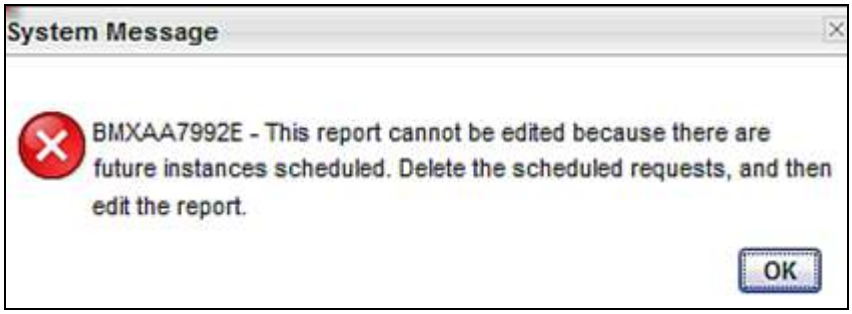
Scheduling:

You can edit the QBR report if there are no future scheduling instances of it.

If there are future scheduling instances of the QBR report, you can not edit it.

If you try to edit it, you will receive the message shown below.

BMXAA7992E - This report cannot be edited because there are future instances scheduled. Delete the scheduled request and then edit the report.



Report Object Structure Changes:

You can edit the QBR report if the ROS that it uses has not been changed since it was created.

If the ROS has been changed since the QBR Creation date, the QBR cannot be edited. This occurs because the QBR is dependant on the ROS.

To illustrate this, two scenarios are given below. In the first scenario, the ROS was last updated on January 1, 2011, and the QBR which utilizes it was last updated on April 15, 2011. If the user tries to edit the QBR, he can because the ROS Change date < QBR Change date.

However, in scenario two, the ROS was updated on May 1, 2011 after the QBR's change date of April 15, 2011. In this case, if the user tries to edit the QBR report on May 15, 2011, he will be unable to because the ROS Change date > QBR Change date.

	Scenario 1	Scenario 2
ROS Change Date	January 1, 2011	May 1, 2011
QBR Change Date	April 15, 2011	April 15, 2011
Today's Date	May 15, 2011	May 15, 2011
Can edit QBR?	Yes	No

2.8.3 Key Property File for QBR Editing

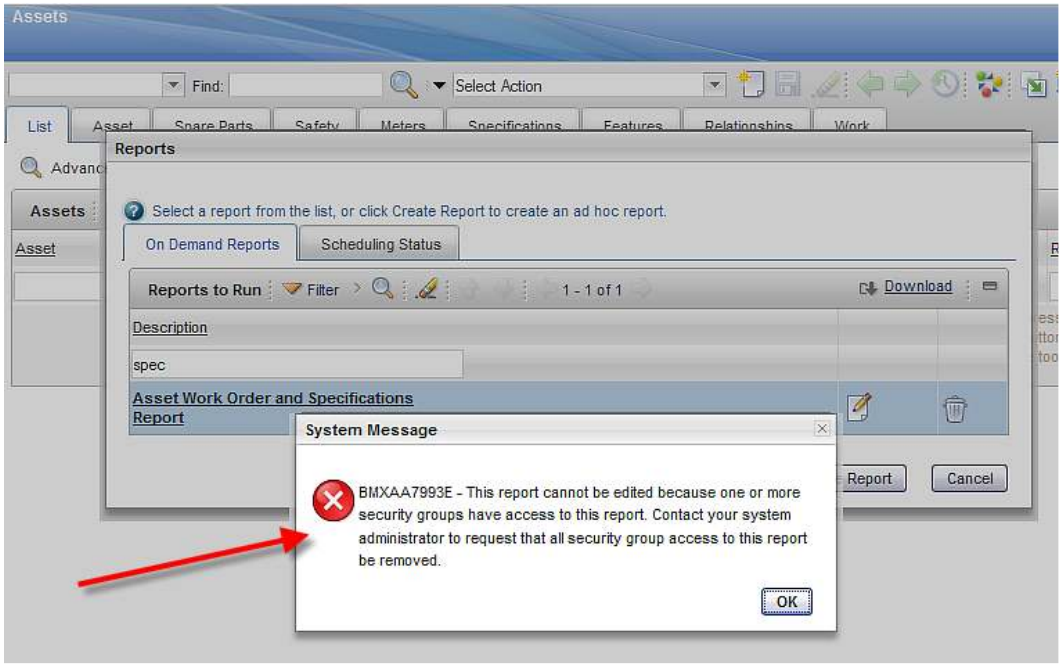
There is a key property file that determines if your users can edit their QBR reports if they have granted other users access to them. This includes the 'Public' security access granted to the EVERYONE security group. The property file is

mxe.report.adhoc.editWithGroupAccess

By default, the setting is set to 0 or no.

This setting can be explained best with an example. Imagine you created a QBR report, and enabled the 'Public' Security field. This gives the EVERYONE security group access to run the QBR report you created. If this setting was left at its default value of 0, and you tried to edit your report, you would receive the message below.

BMXAA7993E – This report cannot be edited because one or more security groups have access to this report. Contact your system administrator to request that all security group access to this report be removed.



This is set as a configurable value because you may not want your users modifying QBR reports that other users may rely on. On the other hand, you may want the QBR owner to have the flexibility to modify the report they create.

In the case where you want a user to modify his report, but want to leave the property setting at its default value, you could duplicate the QBR report in report administration, and not assign any security groups access to it.

Notes on QBR Editing

1. QBR reports created prior to the 7.5 release are not available for editing.

2. QBR reports created prior to the 7.5 release, and duplicated in 7.5, can also not bed edited.

QBR – Security Access

Before detailing how the security functionality is enabled for QBR reports, a best practices overview of security for QBR reports will be detailed.

2.9 Best Practices - QBR Security Access

Each of your Version 7 users has a broad range of skills sets and job requirements. Because of this diversity, you should carefully evaluate report security – and which users have access to report execution and creation.

Using your users varying skill sets and job requirements, you can place them into different reporting categories which could include

1. Non Report Users: Users who utilize the system for viewing data, querying or status input only. They either do not require the ability to execute reports for their job, or they require a very structured functionality process, which does not include report execution.

2. Individual Report Users: Users with specific job requirements, requiring limited access to a small set of reports.

3. Application Report Users: Users with a broader range of job requirements, and who work extensively in a specific set of Maximo applications.

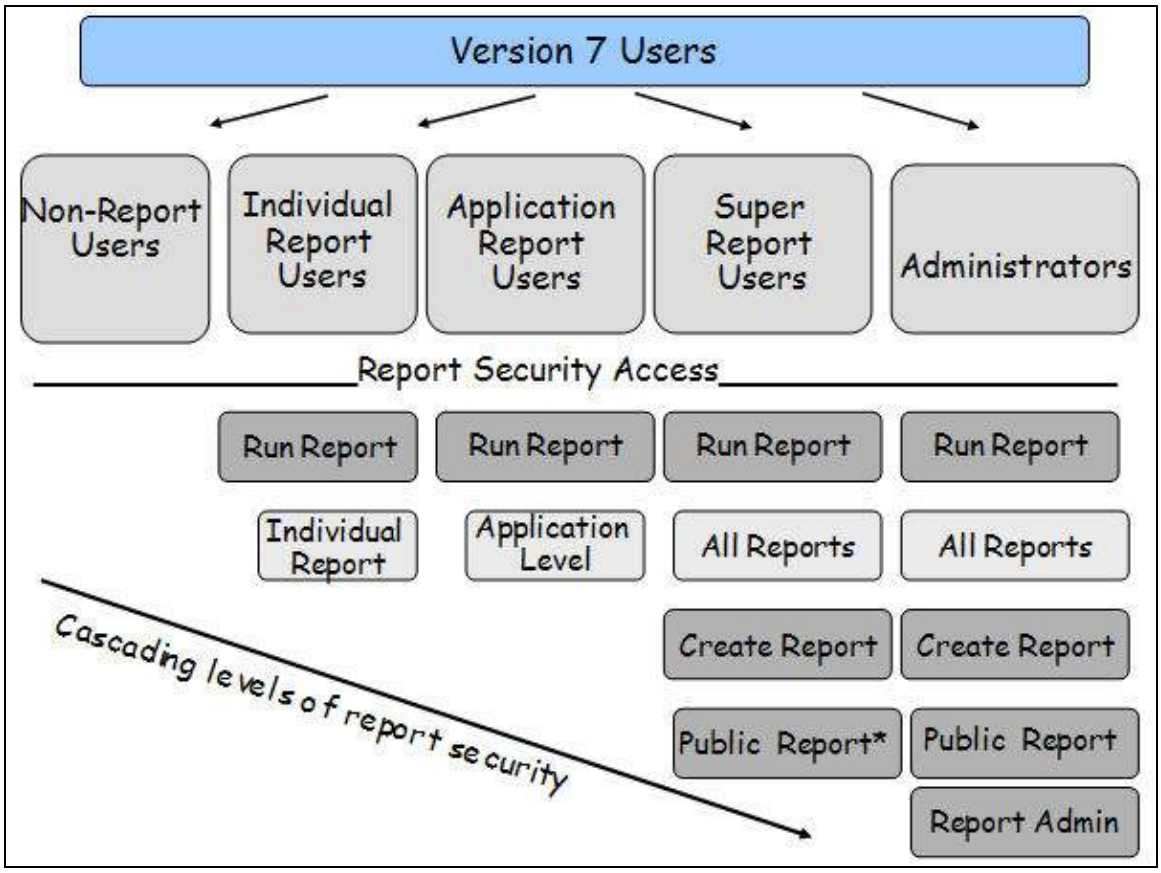
4. Super Report Users: Users whose primary focus is on analyzing data to improve business processes, or to provide data to supervisors/managers. These users are very familiar with the Maximo database and a variety of software tools.

5. Administrators: Users who administer and maintain all aspects of reporting, including security, repository, usage and performance enablers. This individual has the greatest level of report security.

Each of these groups of users should be granted varying levels of report security. This will insure they have the functionality they need to perform their job, while also making sure they are not performing actions which could potentially negatively impact the system or other users.

For example, the 'Individual Report Users' who has very specific job requirements, and hence limited access to a small number of applications within Maximo – would only require access to a small number of reports. Therefore these users should not be granted 'All Report Security Access'. Instead, they should only be granted security access to run a small, selected number of reports, and should not be granted access to create QBR reports.

This compares to the Super Report User, who has access to a large number of applications within Maximo, is familiar with the Maximo database, and spends a large percentage of his day using a variety of software tools. This user should have access to run all reports, and also create his own reports. Depending on the unique skill sets of this user, you may also want to grant them access to create reports for others via the Public report security access.



*More details on the Public Report Functionality can be found in the section titled 'Common Fields – All Four Tabs'

*More details on the varying levels of report security – Individual, Application and All Reports – can be found in the V7 Report Feature guides available here http://ibm.co/niokIv

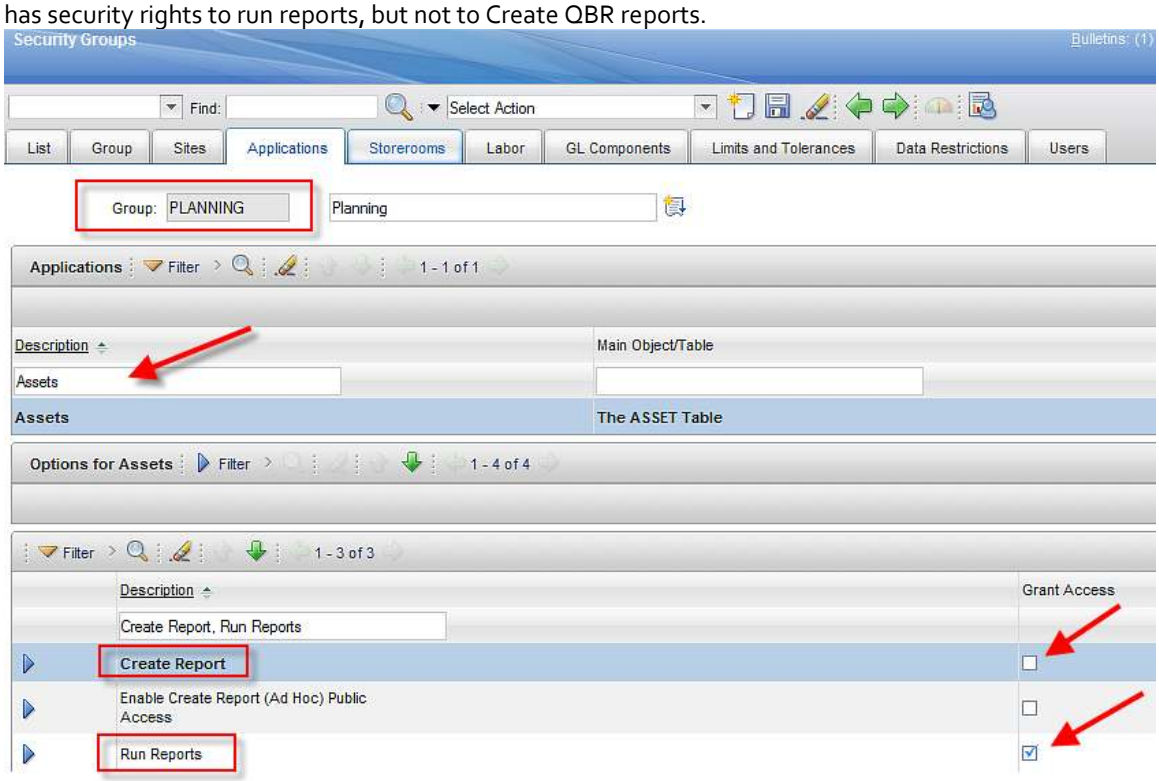
2.9.1 Functionality - QBR Security Access

As noted above, it is recommended that you restrict the number of users who have Create QBR report access. Create QBR report access should only be given to users trained in the functionality, and those with a top level understanding of the Maximo database structure and applications. Restricting QBR security access will help in preventing all users from creating large numbers of reports that may contain very large record sets and that may not be needed.

Therefore, the ability to 'Create' a QBR report must be explicitly granted from the Security Group application for each application. The security group must first have 'Run Report' Access, and then the additional 'Create' QBR Functionality can be given.

Security groups can have 'Create' QBR access for some applications and not others. This enables you to manage the security groups creating reports. This security restriction can prevent users from creating unnecessary reports, which can consume premium database and application server resources.

The Create QBR functionality is available for any application that has the 'Run Report' Action available. An example of this is the Asset application shown below. In this case, the Security Group 'PLANNING' has security rights to run reports, but not to Create QBR reports.



If an application does not have the 'Run Report' Application available, the Create QBR functionality will not be available. There is a dependency on the Create Report functionality on RUNREPORT. An example of this is the BBOARD (Bulletin Board) application.

These security group authorizations are held in the APPLICATIONAUTH table (select * from APPLICATIONAUTH where optionname = 'CREATEREPT').

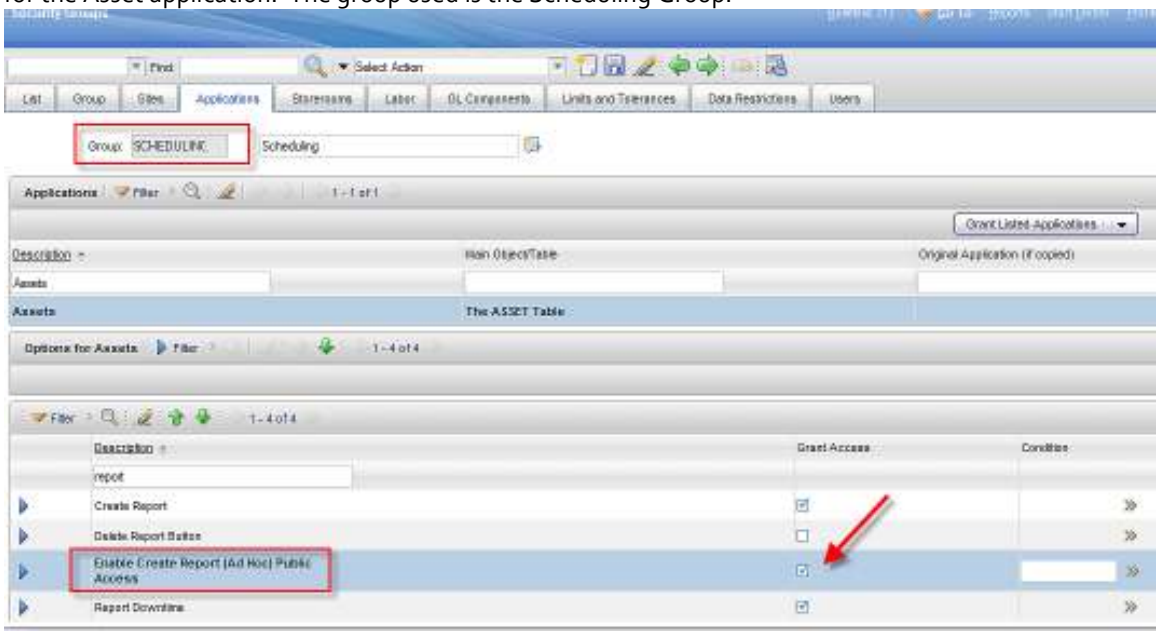
For demo/testing purposes, the maxdemo database has enabled the 'Create QBR' functionality for the MAXADMIN Security Group.

2.9.2 Public QBR Security Access

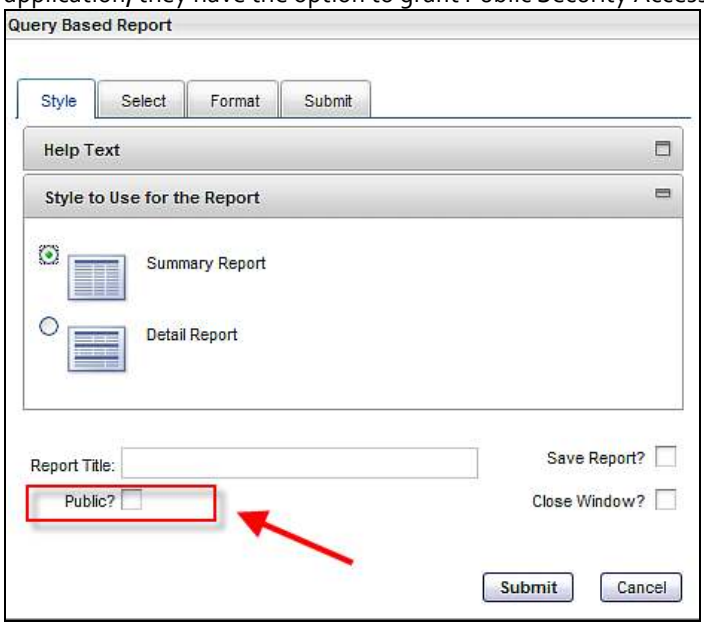
As noted earlier, and in the best practices section, you should evaluate which users are granted access to Public QBR security. This security access enables users to grant EVERYONE security access to the QBR reports they create. This is described in the scenarios below.

Scenario One – Public QBR Security Access Enabled

This first scenario shows what happens when a security group does have Public Access for QBR reports for the Asset application. The group used is the Scheduling Group.

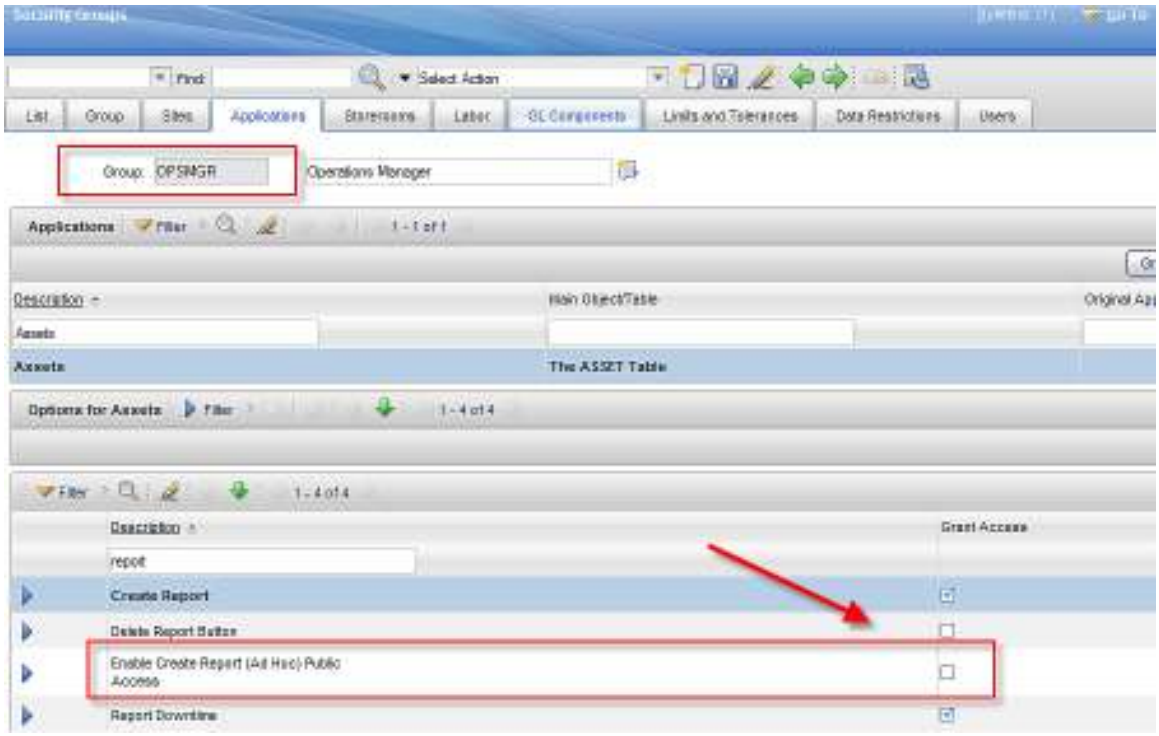


When a member of the SCHEDULING security group accesses the QBR Functionality for the Asset application, they have the option to grant Public Security Access to their individual QBR report – or not.

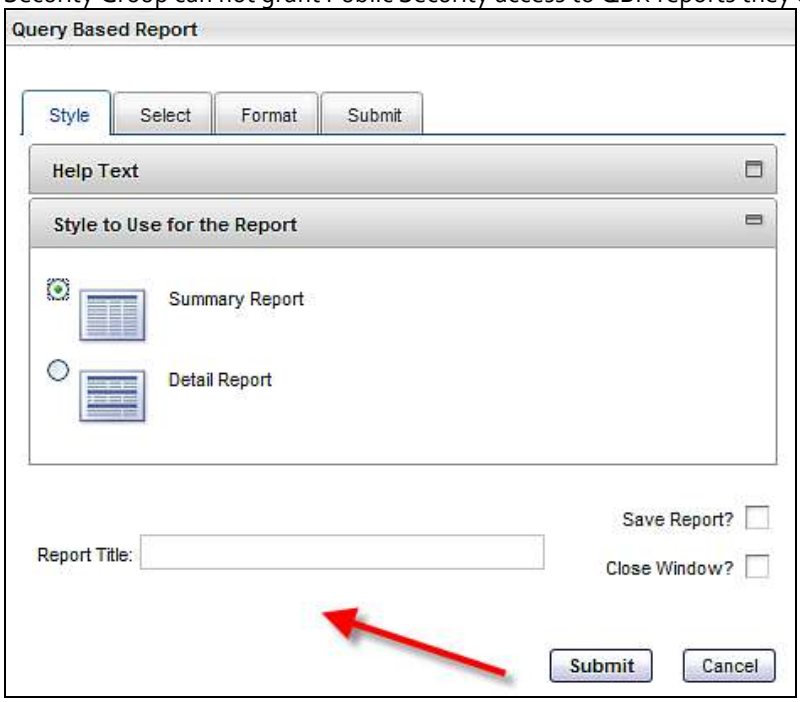


Scenario Two - Public QBR Security Access *NOT* Enabled

In this second scenario, the security group *does not have* Public Access for QBR reports for the Asset application. The group used is the OPSMGR Group.

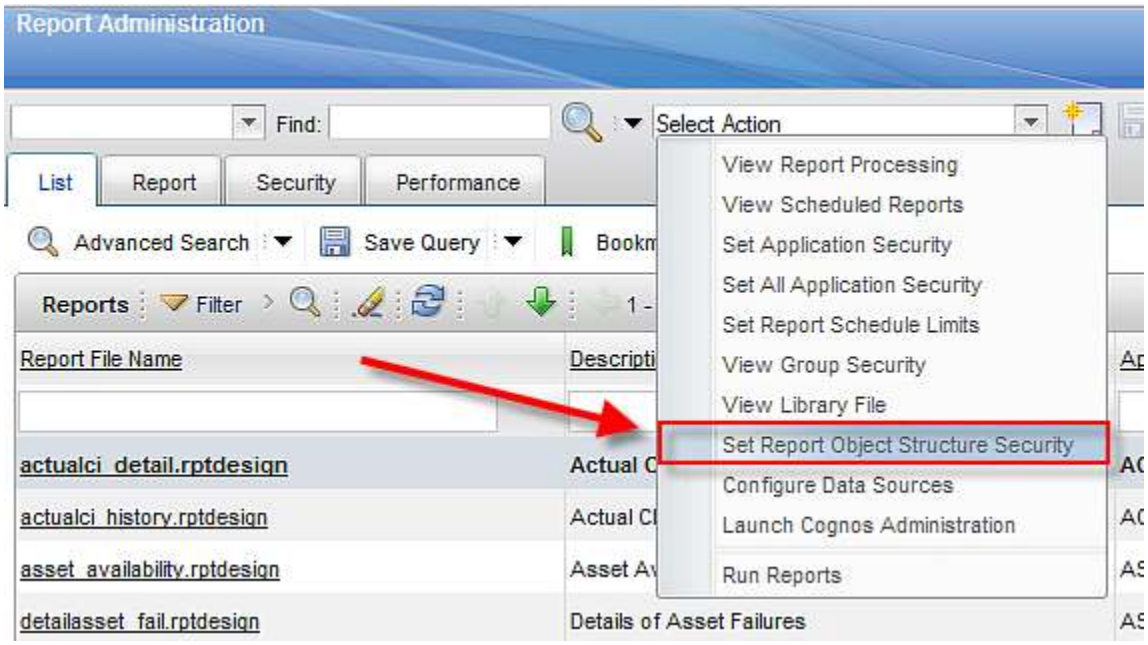


When a member of the OPSMGR Security Group accesses the QBR Functionality for the Asset application, the Public Field is not visible to them in the QBR window. Members of this OPSMGR Security Group can not grant Public Security access to QBR reports they create.

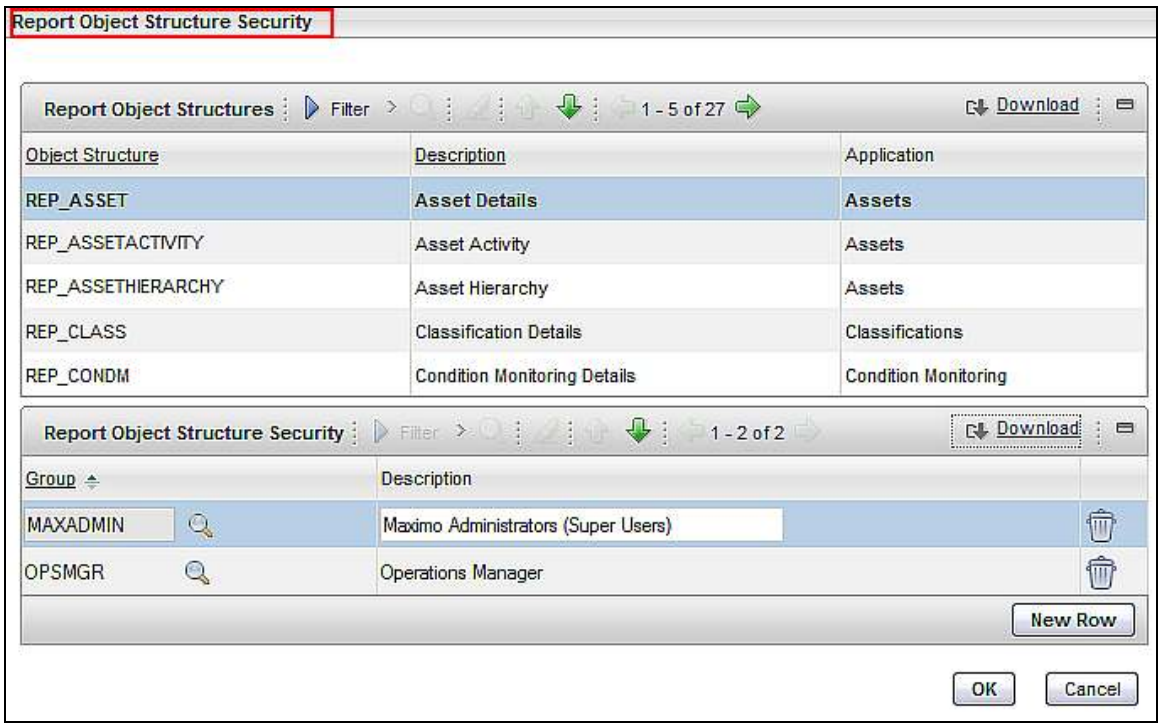


2.9.3 Report Object Structure Security Access

You can specify which security groups have access to which Report Object Structures within the Report administration application. To define this, select the 'Set Report Object Structure Security' from the action menu. This action is modeled after the 'Set Application Security' also available in Report Administration.



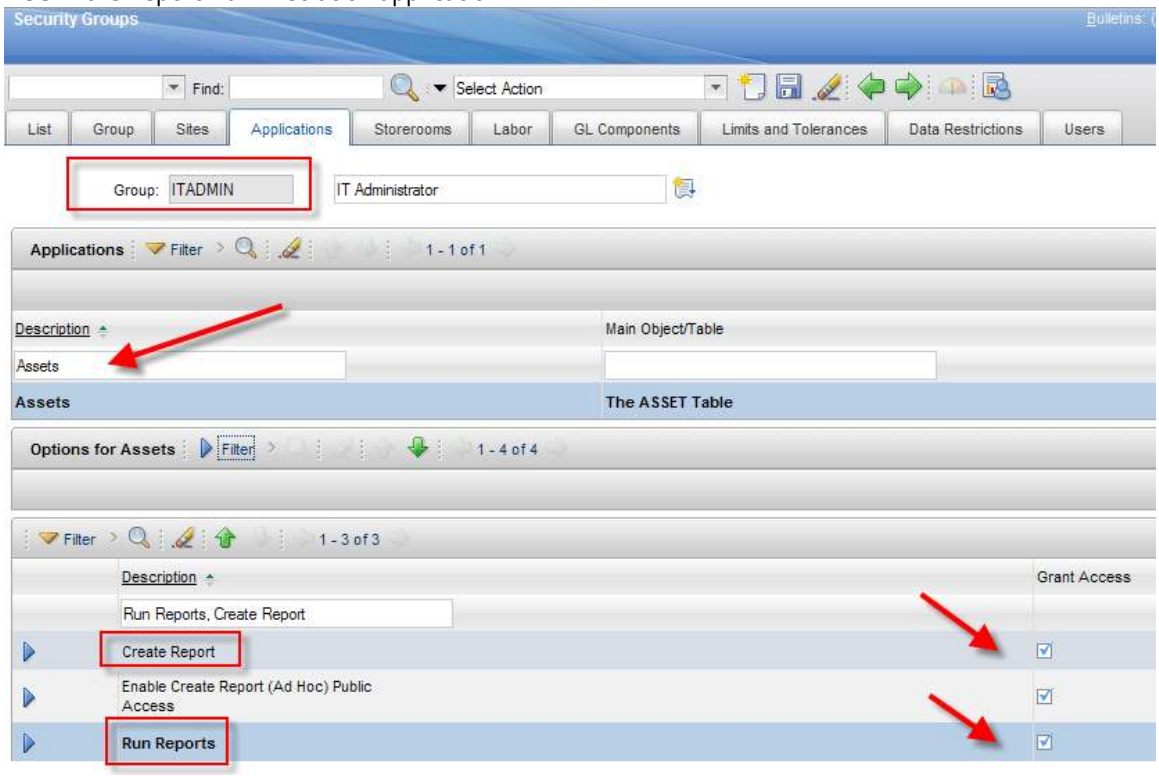
The Report Object Structure Security action has two sections: (1) The top section displays all ROS, and (2) the bottom section shows the security groups who have been granted access to the ROS.



To grant security access, the Administrator highlights the Report Object Structure, like Asset Details. He then clicks New Row, and follows standard functionality for adding new Security Group access. A lookup is available on the Group Field to quickly identify Security Groups.

The list of groups that an administrator can grant ROS Security access to are only those security groups that have 'Create Report' access to the application. This simplifies managing ROS Security for the administrator.

An example of this is shown below for the ITADMIN Security Group. This group has both Run Reports and Create Report access for the Asset application, so they can be granted access to any of the Asset ROS in the Report Administration application.



Note: Security Group authorizations to the Report Object Structures are held in the REPORTOSAUTH table.

2.10 QBR Creation – Record Limits

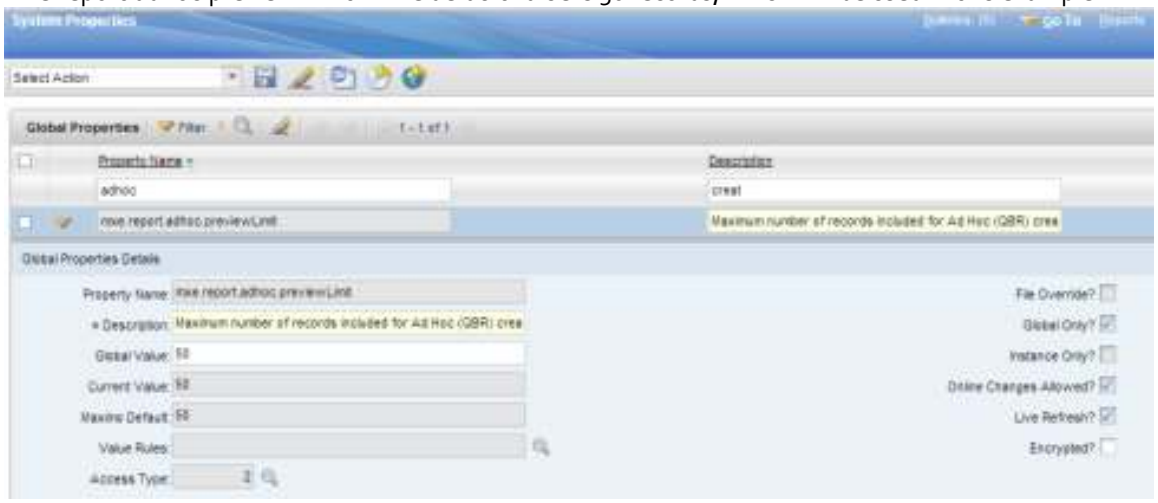
When your users create QBR Reports, you may want to prevent them from executing these reports against large database record sets. This can prevent your users from accidentally creating QBR reports on large sums of records, which can cause negative performance implications.

To enable this, a property setting was introduced in the 7.5.0.3 Fixpack level in April 2013 called mxe.report.adhoc.previewLimit.

With this property setting, you define a maximum record limit that a QBR report can execute against when it is created. If the records in the QBR query exceed that limit, the QBR preview report will not display.

A scenario detailing this functionality is provided below.

1. Access the property application, and filter for the property setting mxe.report.adhoc.previewLimit. The default value is 50 records, which will be used in this example.

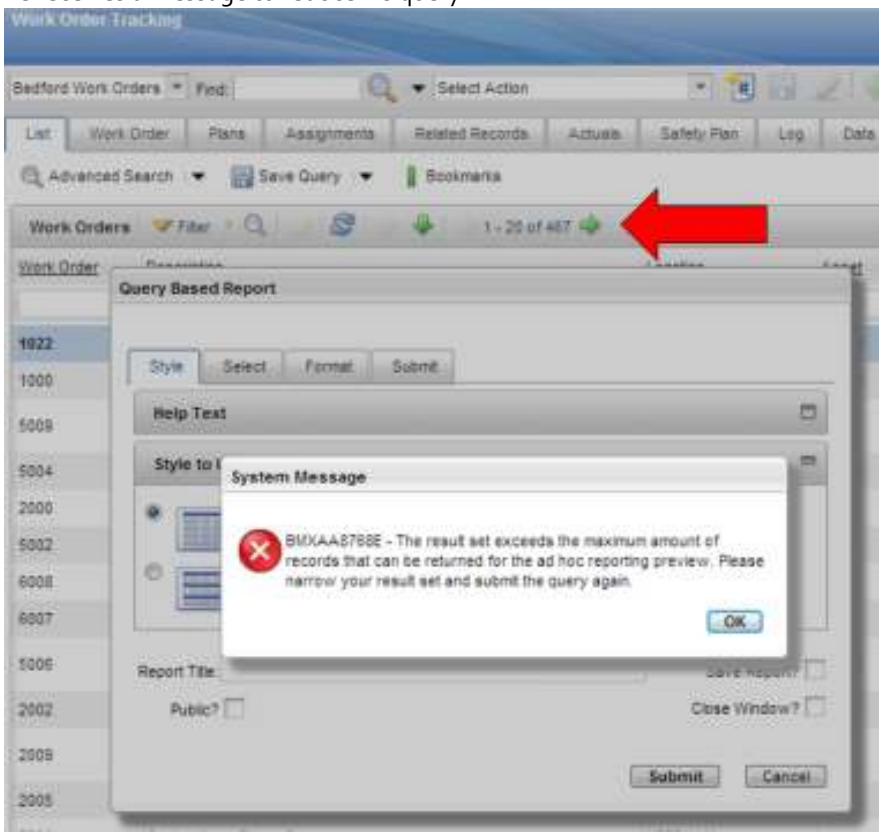


2. Next, one of your users begins creating a QBR report within the Work Order Tracking application. He is unaware of the large amount of records in his query, which is 467 records as shown in the screen shot below. When the user clicks submit – an evaluation of the application record set and any filters the user may have added in his Ad hoc report – is made against the property setting value.

   If the (application record set + report filters) > property setting value, the report preview will not occur. The user will be prompted with a message to reduce his application query as shown in the screen shot below.

   If the (application record set + report filters) < property setting value, the report preview will display in the report viewer.

In this case, his application query of 467 records is greater than the 50 records in the property setting. So, he receives a message to reduce his query.



Notes:
1. If the user had clicked 'Save' in the QBR dialog, and his record set > property value, the report design file will be created and saved in the database – but it will not display in the report viewer.

2. This setting will only apply to only those QBR reports that are being created (not yet saved)

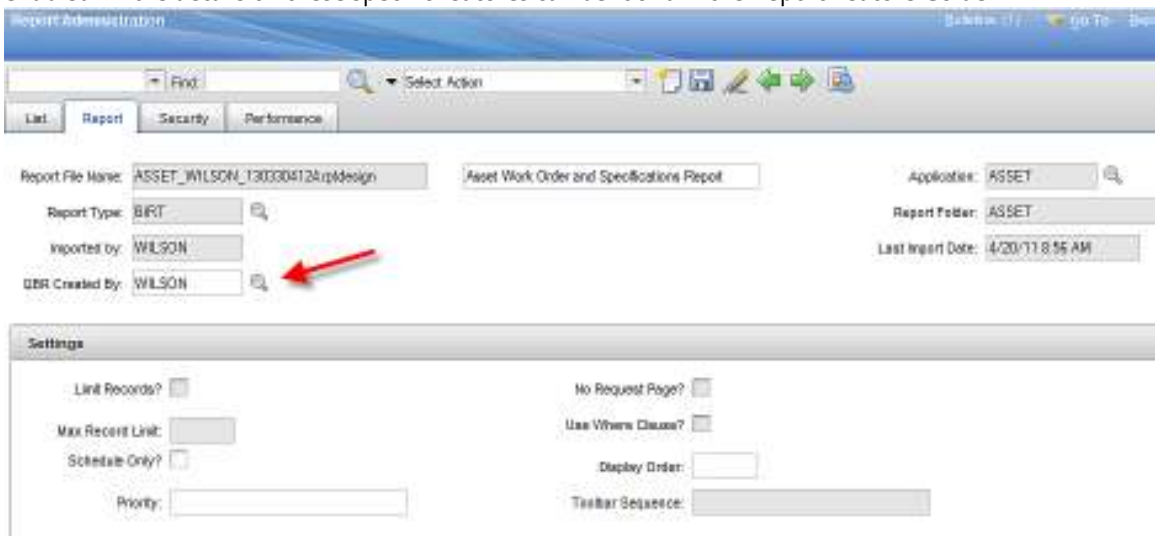
3. If a user accesses the 'Create Report' functionality outside of the application (ex. Start Center) and this property setting is enabled, they will receive the same warning message as noted above and the report will not execute.

2.11 Key Administration Settings and Notes for QBR reports

1. The QBR functionality requires the names "maximoLandscape" and "maximoDataSource" for the appropriate library components. If you rename or replace either of these components, QBR reports will not execute.

> If you need to modify the master page "maximoLandscape" in the library, be sure to keep the original name of 'maximoLandscape" after your modifications. Additionally, be sure to review this file in any future fixpacks or version updates, as you may need to re-apply your customizations.

2. Report Settings: Report administrators can extend QBR reports by enabling settings as Schedule Only, Priority and Display Orders. Additionally, depending on whether or not the QBR has parameters defined, settings like Browser View, Direct Print and Direct Print with Attachments can potentially be enabled. More details on these specific features can be found in the Report Feature Guide.

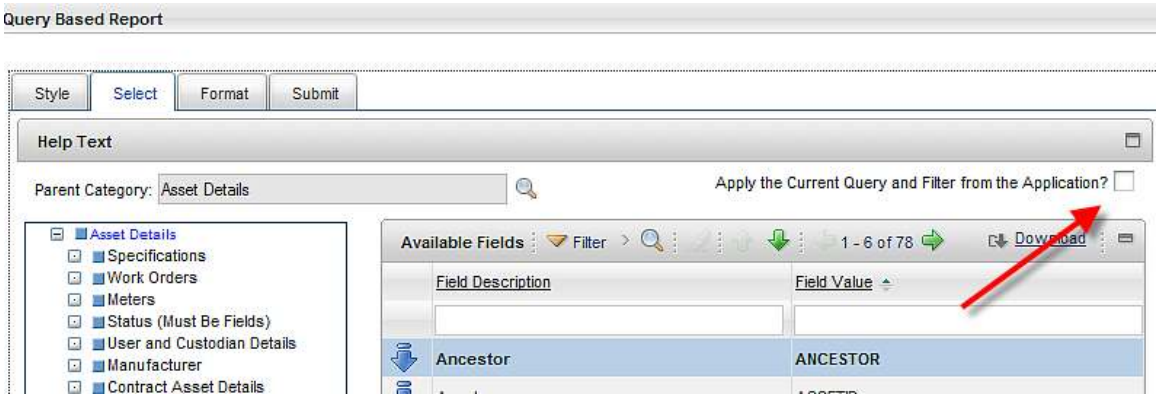


3. Parameters can be added to QBR reports during their creation. As noted earlier, these parameters can only be specified on the main table of the application, and are bound parameters.

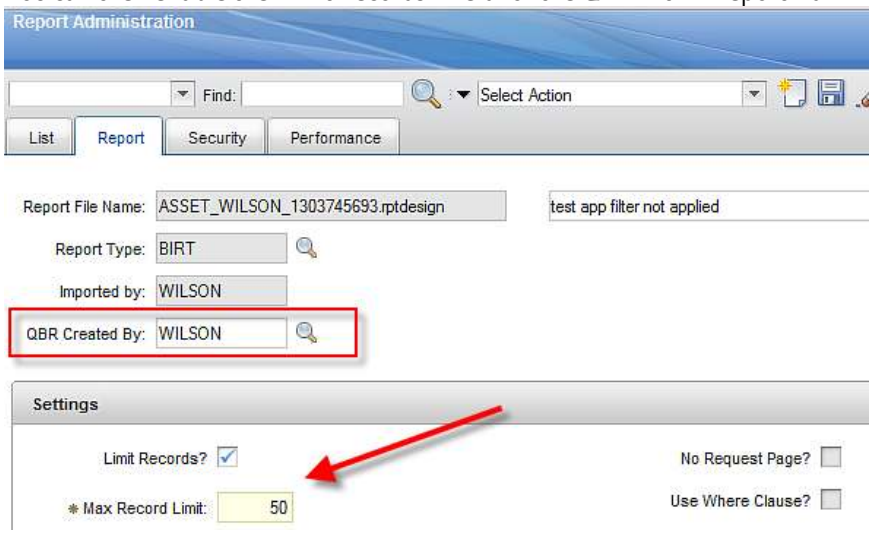
- If the bound parameter is a date, the Date Lookup will applied by default.
- If the bound parameter is not a date, no Lookup will be applied by default. However, the administrator can access the QBR's record in the Report Administration application, and manually apply a lookup if needed.

4. As highlighted in the screen shot above, when saving an Ad Hoc report, the field 'Use Where Clause' in Report Administration is disabled. This is a non-editable field that is not available to saved Ad Hoc reports..

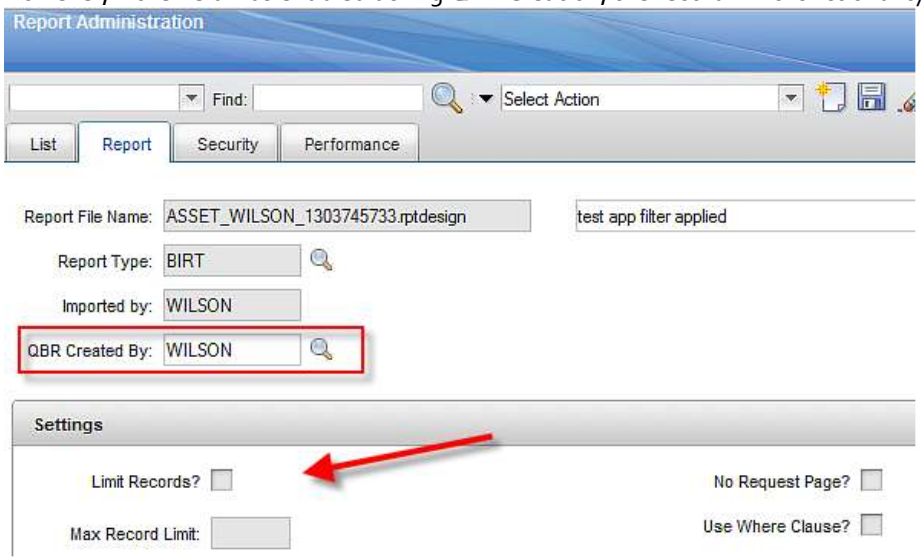
5. If you did not enable the flag 'Apply the Current Query and Filter from the application' during QBR Creation as shown here -



You can then enable the Limit Records? Field for the QBR within Report Administration.



However, if the field was enabled during QBR Creation, the record limit functionality is not available.



2.12 Extending Ad Hoc Reporting Functionality

The features listed below are a few items that you may wish to utilize to extend or customize the Ad Hoc reporting functionality.

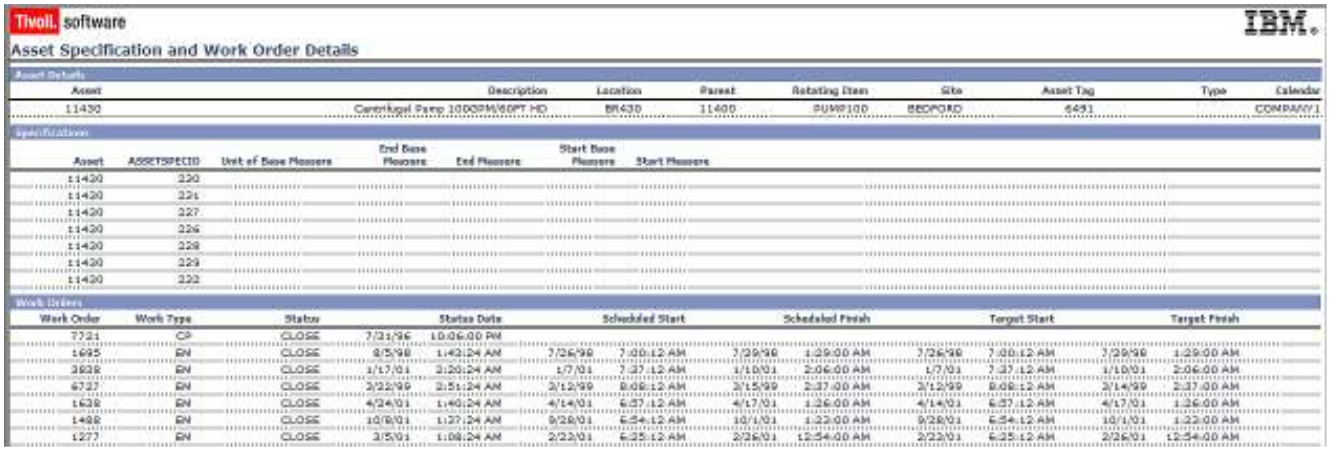
2.12.1 Use Ad Hoc Reporting as a base for Custom Report Development

When Ad Hoc Report is created and saved, its resulting design file (.rptdesign) that was created on the fly is saved to the database. This enables the report to be accessed in the future for immediate run access or scheduling and emailing.

Additionally, once the report is saved in the database, it can be extended within the Report Designer tool. By simply exporting the report and opening it in the design tool, you can quickly build upon the report design by adding calculations, graphs or additional features.

This can become a huge time saving feature for your custom report development because ad hoc reports can be created with complex sql from multiple tables, filters and application queries. Instead of having the developer create all this information – you can let the Maximo framework perform this work, and then build upon the ad hoc report in the designer.

To show how this can be accomplished, an example below is given. First, the Ad Hoc report is created and saved. In this case, the report is created in the Asset application, and called 'Asset Specifications and Work Order Details.'


Tivoli. software

IBM.

Asset Specification and Work Order Details

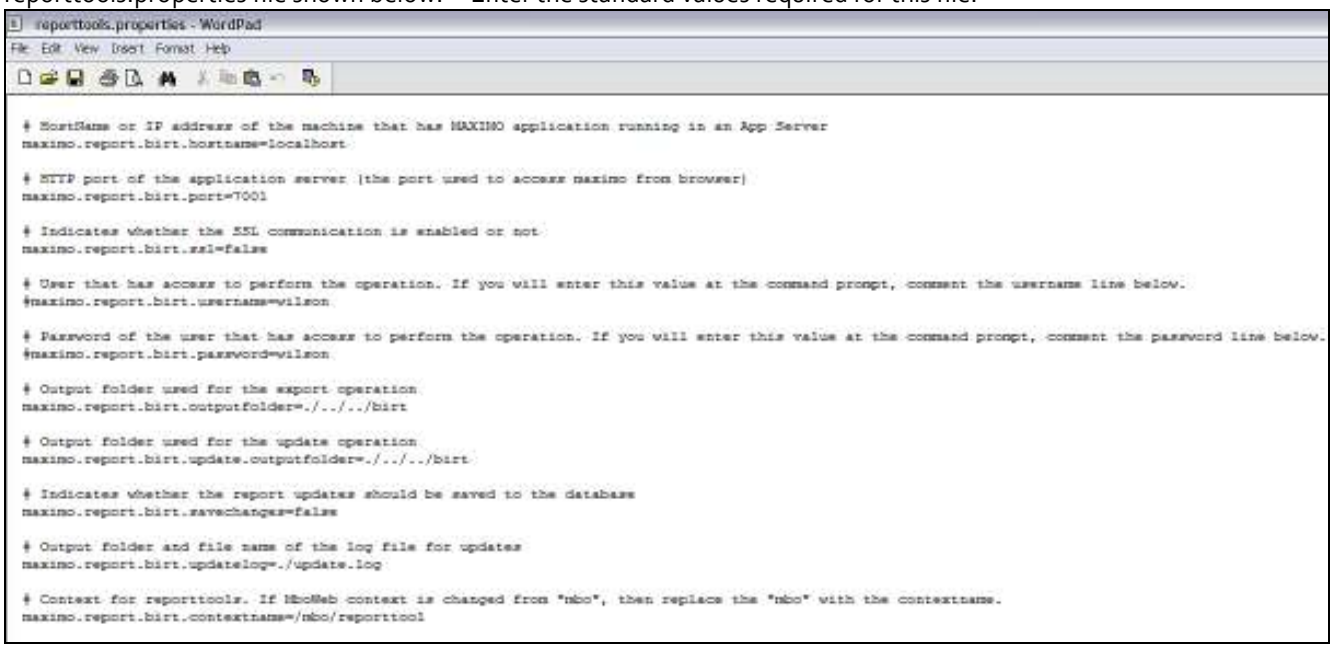
Asset	Description	Location	Parent	Rotating Item	Site	Asset Tag	Type	Calendar
11430	Centrifugal Pump 1000GPM/60PT HD	BR430	11400	PUMP100	BEDFORD	6491		COMPANY1

Asset	ASSETSPECID	Unit of Base Measure	End Base Measure	End Measure	Start Base Measure	Start Measure
11430	220					
11430	221					
11430	227					
11430	226					
11430	228					
11430	229					
11430	222					

Work Order	Work Type	Status	Status Date	Scheduled Start	Scheduled Finish	Target Start	Target Finish
7721	CP	CLOSE	7/31/96 10:06:00 PM				
1695	EN	CLOSE	8/5/98 1:43:24 AM	7/26/98 7:00:12 AM	7/29/98 1:29:00 AM	7/26/98 7:00:12 AM	7/29/98 1:29:00 AM
3838	EN	CLOSE	1/17/01 2:20:24 AM	1/7/01 7:37:12 AM	1/10/01 2:06:00 AM	1/7/01 7:37:12 AM	1/10/01 2:06:00 AM
6727	EN	CLOSE	2/22/99 2:51:24 AM	2/12/99 8:08:12 AM	2/15/99 2:37:00 AM	2/12/99 8:08:12 AM	2/14/99 2:37:00 AM
1628	EN	CLOSE	4/24/01 1:40:24 AM	4/14/01 6:57:12 AM	4/17/01 1:26:00 AM	4/14/01 6:57:12 AM	4/17/01 1:26:00 AM
1498	EN	CLOSE	10/8/01 1:27:24 AM	9/28/01 6:54:12 AM	10/1/01 1:22:00 AM	9/28/01 6:54:12 AM	10/1/01 1:22:00 AM
1277	EN	CLOSE	3/5/01 1:08:24 AM	2/22/01 6:25:12 AM	2/26/01 12:54:00 AM	2/22/01 6:25:12 AM	2/26/01 12:54:00 AM

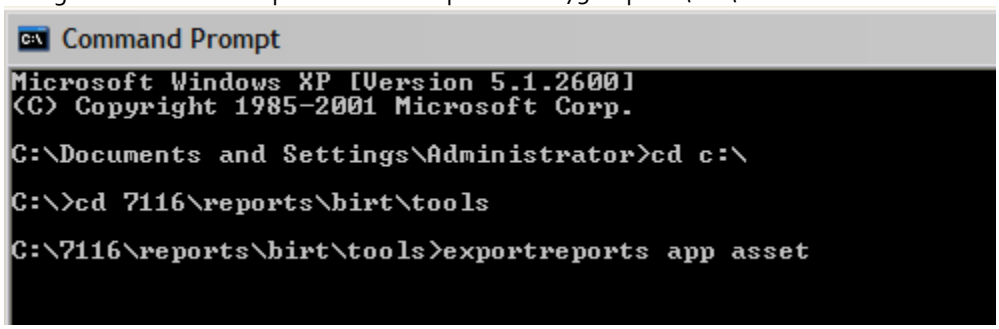
Then, the report is exported for its repository in the database to a local file system. This is done via a command utility. Enabling the command utility is a properties file which first must be configured. The property file is: reporttools.properties file.

To configure this file, browse to the tool location ...<V75> \reports\birt\tools. Locate and open the reporttools.properties file shown below. Enter the standard values required for this file.

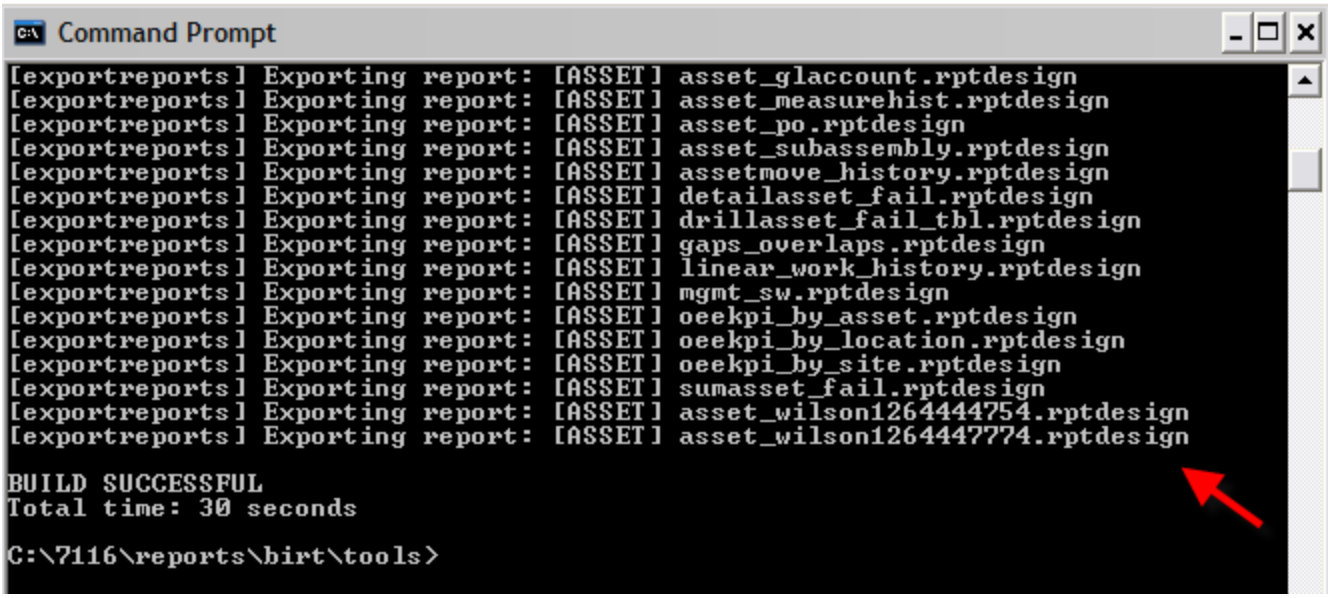


*NOTE: For more details on configuring this file and the variations that can be enabled, reference the 'V75 Report Developer's Guide' noted at the end of this document.

After configuring the properties file, you can export the Ad Hoc Report by opening a command window. Navigate down to the exportcommands path.... <V75>reports\birt\tools

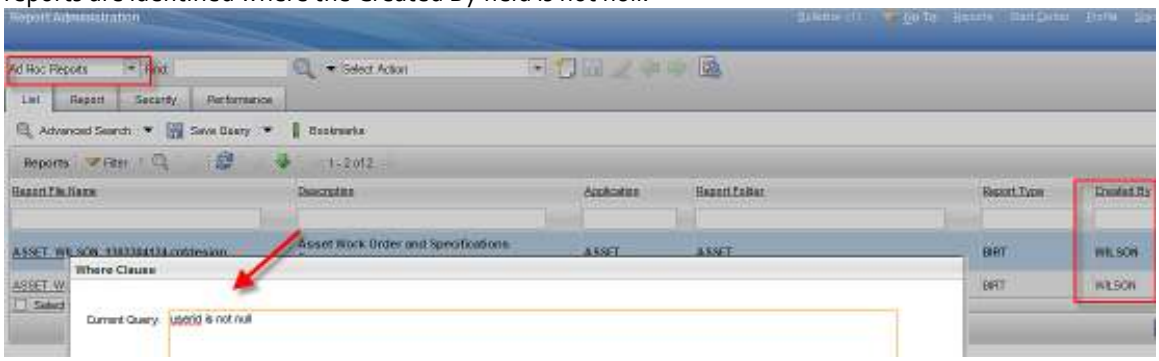


Then, export all of the reports from the asset application by:
exportreports app asset



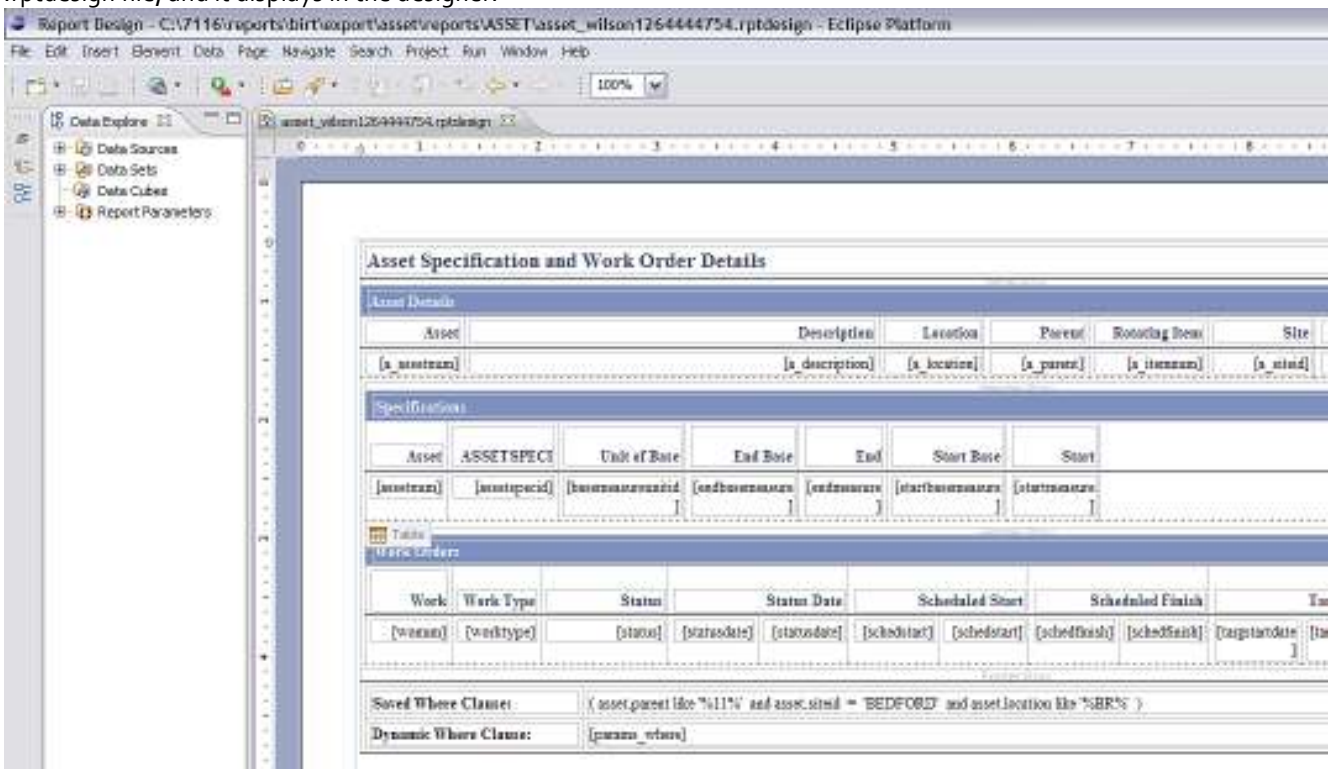
*Note: You can also export only a single report by using the command:
exportreport report <application> <reportfilename>

However, all command files work with file names – not the report descriptions that users assign during Ad Hoc Report Creation. So to use this command, you first must know the exact name of the QBR report. To find the report file name, access the Report Administration application. You can either filter on the 'Created By' field in the list tab, or utilize a query to quickly locate your Ad hoc reports. Ad Hoc reports are identified where the Created By field is not null.

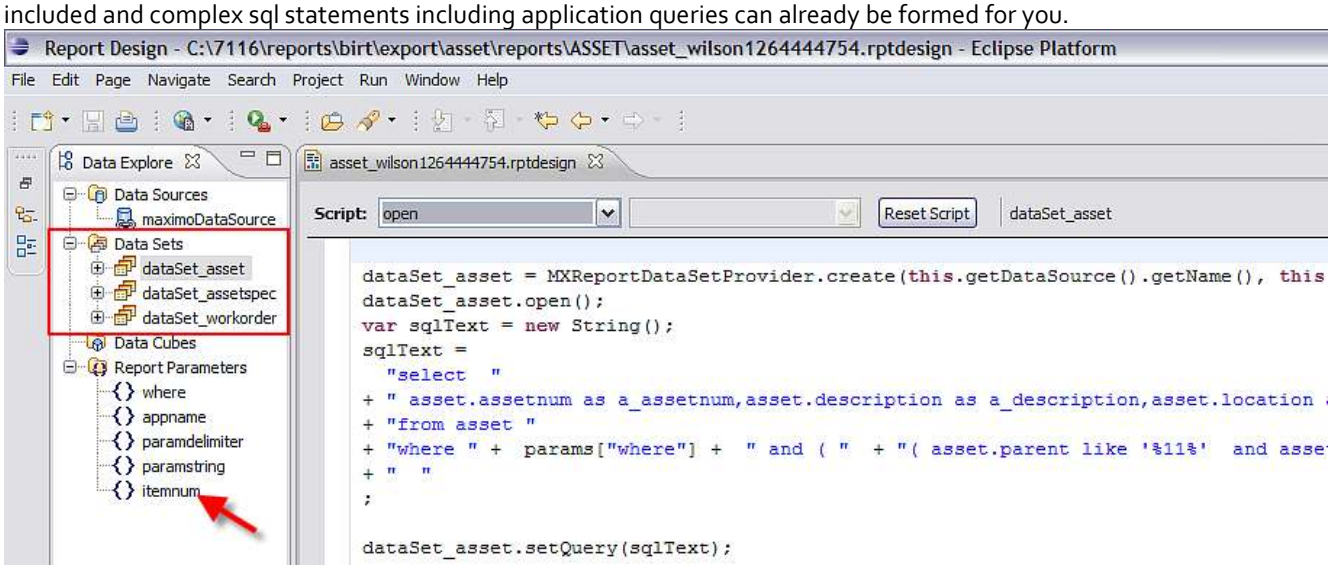


Then execute the command to export a single report

Once you have exported the ad hoc report, open up the Report Designer tool. From the menu, click File – Open File and navigate to the directory where you exported your reports. Select the Ad Hoc Report's .rptdesign file, and it displays in the designer.



You can immediately see that you have an excellent beginning to extend this report further for any other customizations you may need. Multiple data sets (subreports) can be already populated, parameters included and complex sql statements including application queries can already be formed for you.



This can become an excellent starting point for your report developer.
Note: If the developer chooses to modify the design file, it is recommended that the report file name be modified to identify it from the original file. If the developer plans on utilizing this report as an Enterprise

Report, he would need to create and/or append the reports.xml and properties file for the new enterprise report. Additionally, because it is now an enterprise report, it would need to be imported through the reports import command, or the UI utility in the Report Administration application.

2.12.2 Number of Rows in Select Tab of QBR Window

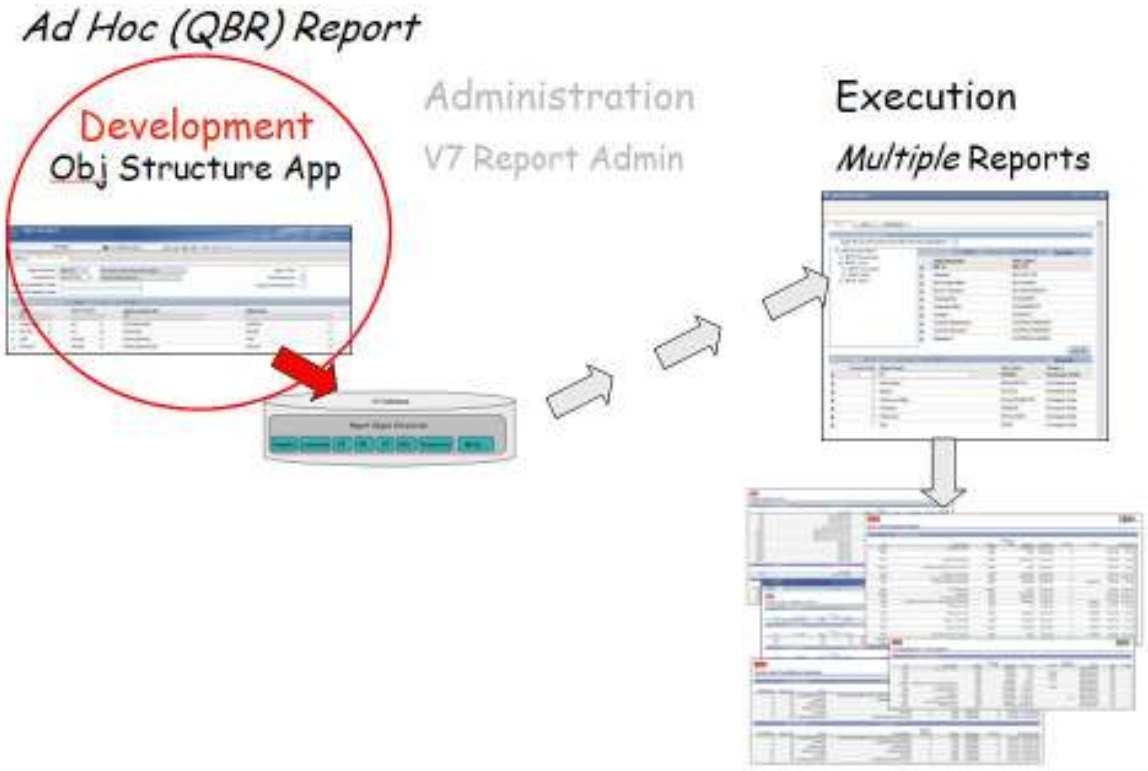
10 rows display by default in the 'Available Fields' and 'Selected Fields' sections of the Select QBR Tab. This number can be configured in the library.xml file located in <V75>\resources\presentations\system in the code shown below.

```
<table id="available_fields_table" label="Available Fields" inputmode="readonly" relationship="MAXATTRIBUTE" orderby="attributename"
  <tablebody id="available_fields_table_body" displayrowsperpage="10" filterexpanded="false" filterable="true"  >
    <tablecol id="available_fields_table_col_1" type="event" mxevent="addAttribute" mxevent_icon="btn_moveToSelected.gif" mxevent_de
    <tablecol id="available_fields_table_col_2" label="Field Description" dataattribute="title" sortable="true" showfilterfield="tru
    <tablecol id="available_fields_table_col_3" label="Field Value" dataattribute="attributename" sortable="true" showfilterfield="t
  </tablebody>
</table>
```

If you want to modify the default value, it should be set in both the 'Available Fields' and 'Selected Fields' Section. The updated xml file should then be reimported via the Application Designer.

3 QBR – Development Process

Next, the document details the setup work involved in enabling Ad Hoc reports, including Administration and how to enable Report Object Structures. This last portion is intended only for developers and/or administrators.

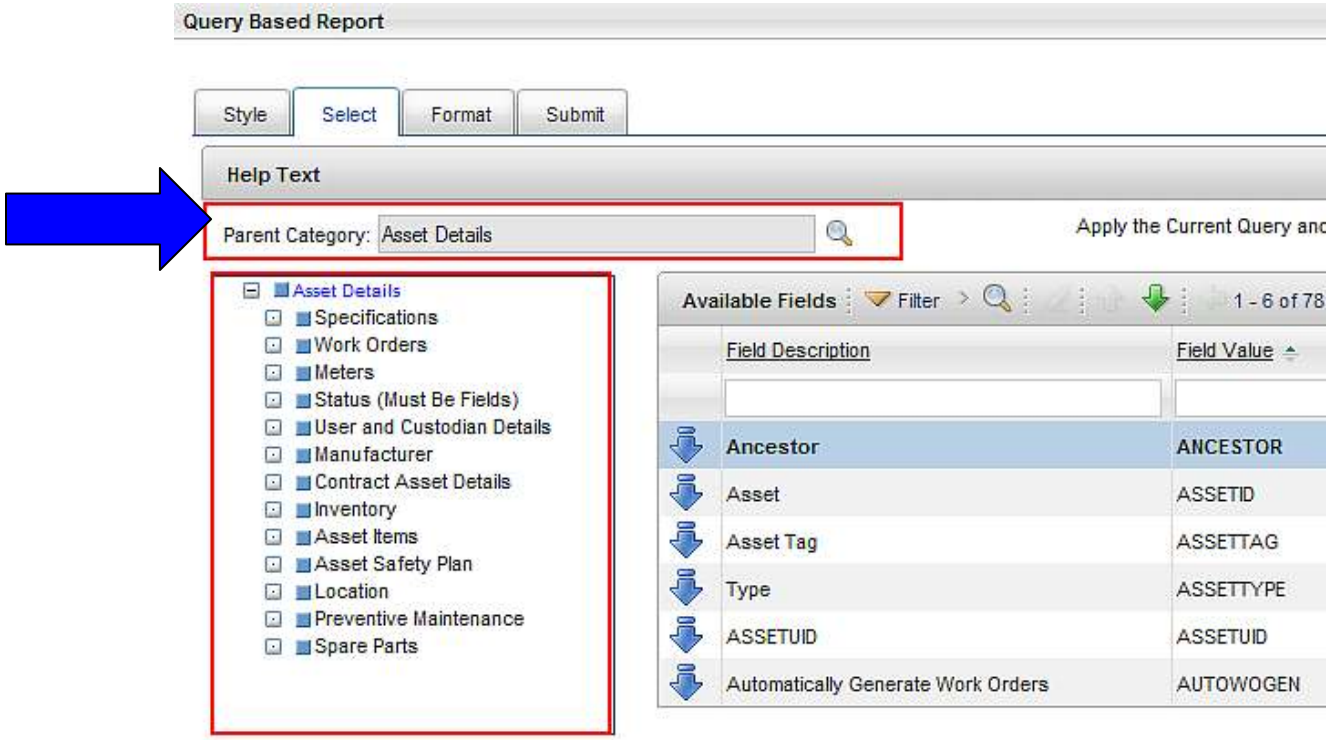


3.1 Overview of ROS

Report Object Structures (ROS) are used within the Ad Hoc, or Query Based Reporting (QBR) Functionality. ROS enable you to select fields from multiple categories (objects) for their ad hoc reports. Additionally, ROS are designed to enable you to visualize application categories and their parent/child relationships. These categories are displayed via a category tree drilldown in the Select Tab of the QBR window in the top left hand side.

When you access a QBR window, the Report Object Structures are displayed via user friendly terminology – Parent Category. Parent Categories are ROS that you have access to.

The example below shows a QBR window for the Asset application. The ROS that you have security rights to are accessible via the lookup highlighted by the red arrow. The blue arrow highlights the selected ROS with its parent and child objects.



In this example, the Parent Category is Asset Details. This parent has multiple children (Specifications, Work Orders, Meters etc). The children categories are Source Objects created for the Application's Main Object, Asset, through maxrelationships.

If a child does not have any children, no further drilldowns are available. If a child has additional children, (main object's grandchildren), they are displayed visually with the + icon. Multiple children-grandchildren relationships can be configured with ROS.

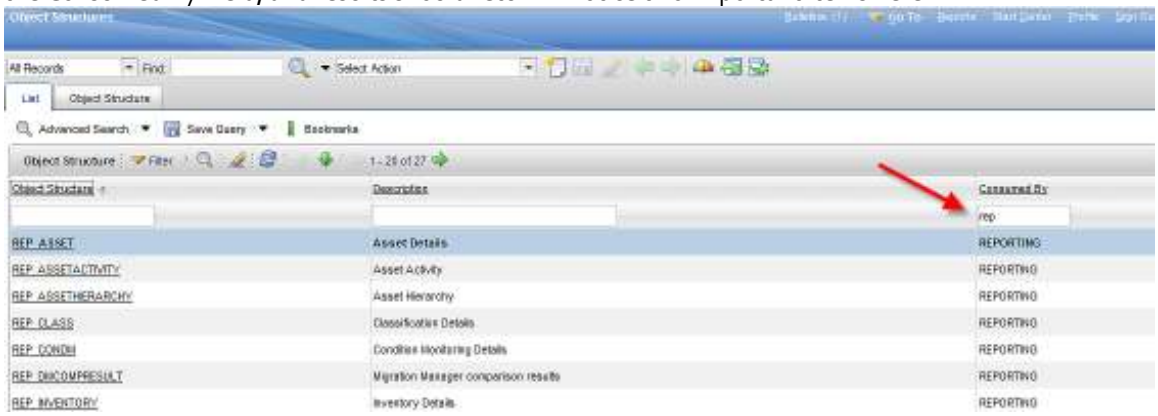
3.2 Creating Report Object Structures

To enable field selection from multiple tables, setup or administrative work needs to be done to enable the ROS. This is only required if you want the QBR report to pull from multiple tables.

It is very important that this setup work be done by a developer or administrator who is very familiar with the Maximo database and database concepts. Without this knowledge, the ROS can be setup incorrectly, leading to bad or misleading data in the Ad Hoc reports.

The steps below detail how to setup a ROS for the Purchase Requisition application and the business rules involved. This example is being used as a ROS for this application is not delivered out of the box.

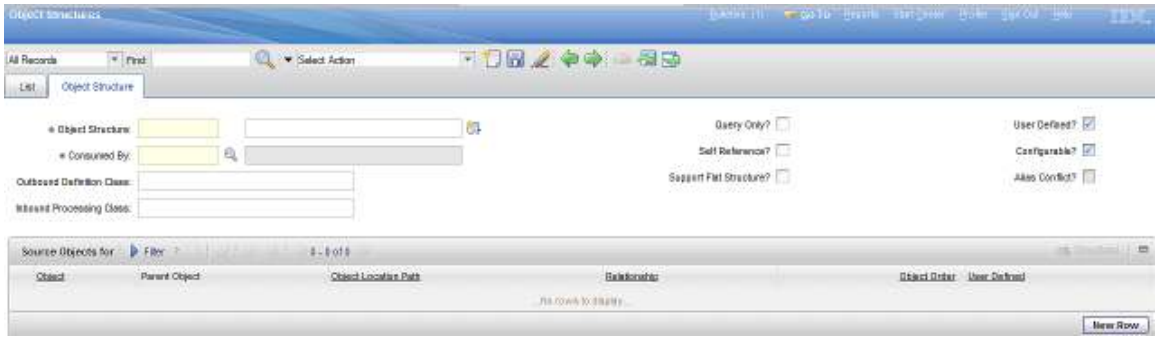
1. Go to the Object Structure Application by accessing Integration -> Object Structures. Filter on rep in the Consumed By Field, and results should return. Notice two important items here -



A. First, Report Object Structures are identified with the unique attribute of consumed by of 'REPORTING'. If this is not done, the ROS will not be enabled for QBR reporting. To enable this, a Value of REPORTING with a Description of 'Reporting Application' is used in the SYNONYMDOMAIN Table for DOMAINID = 'INTUSEWITH'.

B. Secondly, it is it is recommended that you use a naming convention to quickly identify the ROS. The ROS delivered use the naming convention of 'REP_Application Name'. (For example, REP_PR). You can choose to either continue with this naming convention, or utilize one that may be more applicable to your environment.

2. Next, we will begin building a new ROS for the Purchase Requisition (PR) application. This process is very similar to other parent – child relationship building. To start, insert a new record.

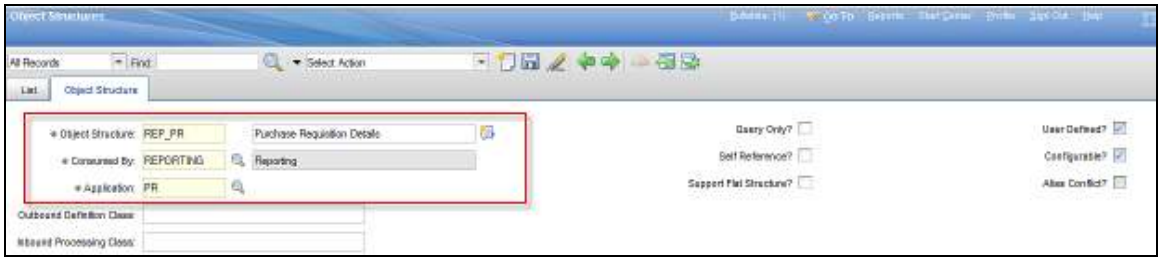


3. The top level of a ROS identifies the application where it will be used. Since this ROS is for the PR application, whose main table is PR (MAXAPPS.MAINTBNAME), an Object Structure called REP_PR will be created. The standard naming convention will be utilized, so the following values will be entered:

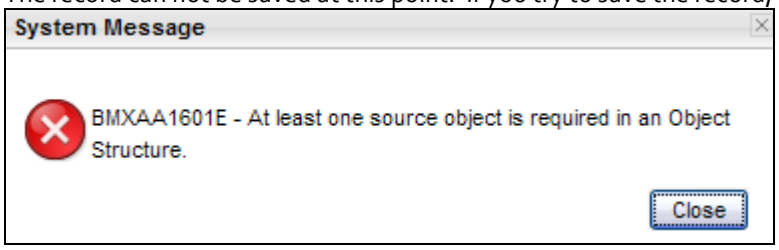
Object Structure: REP_PR
Description: Purchase Requisition Details
Consumed By: REPORTING (Selected from Value List)
Application: PR

*Note: When you select the Consumed By of Reporting, the Application Field conditionally appears. This is required as this associates the ROS to an application.

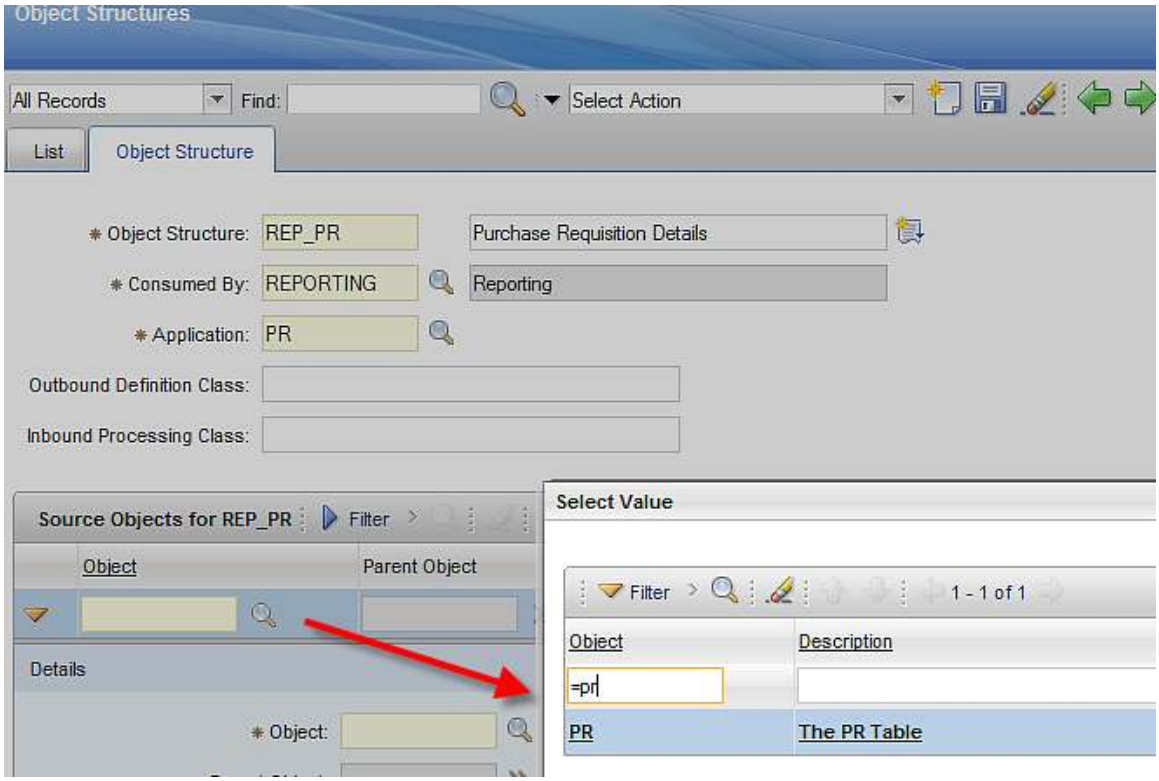
All other header fields should be null by default.



The record can not be saved at this point. If you try to save the record, the following error will appear:

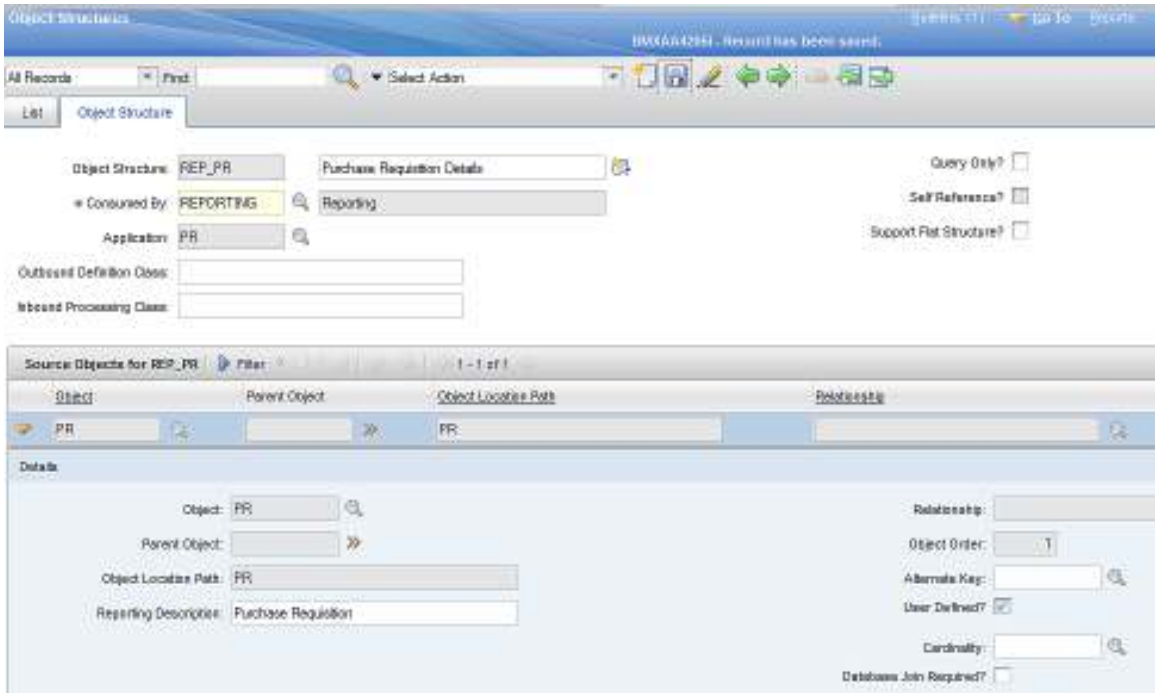


4. Next, you must define at least one source object for the ROS. The Main Object (Table) for the application will be the Parent Source Object. Again, this will be the main table of the application (MAXAPPS.MAINTBNAME). Therefore, the Object of PR will be selected from the Object Lookup.


Object Structures
All Records
Find:
Select Action
List
Object Structure
* Object Structure:
REP_PR
Purchase Requisition Details
* Consumed By:
REPORTING
Reporting
* Application:
PR
Outbound Definition Class:
Inbound Processing Class:
Source Objects for REP_PR
Filter
Object
Parent Object
Details
* Object:
Select Value
Filter
1 - 1 of 1
Object
Description
=pr
PR
The PR Table

5. The Parent Source Object (PR) must (1) have a Parent Object of null and (2) have an Object Order of 1.

Also, note the Reporting Description field. The text value entered will display in the Create QBR window. Therefore, enter a text value that will immediately be recognized. After entering this information, save the record.



Next, any child or grandchildren categories must be defined as Source Objects to the parent. These Source Objects are also known as the Categories for the Application's QBR. They are the database tables (objects) related to the parent via maxrelationships.

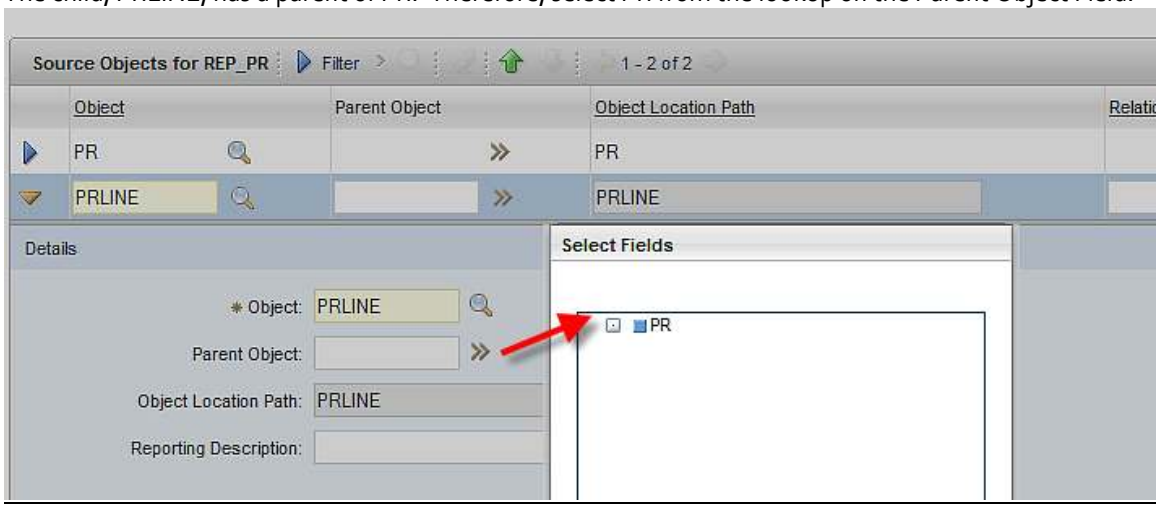
Three children will be created in our example: PR Lines, Companies, Terms and Conditions.

PR

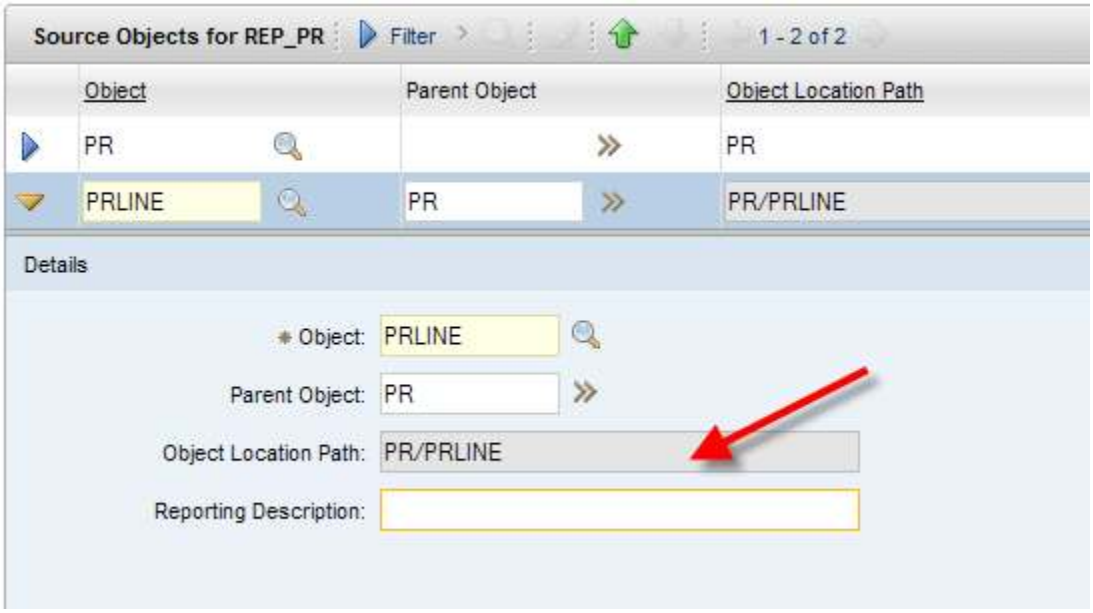
- PR Lines
- Companies
- Terms and Conditions

6. To do this, click New Row in the Source Object Section. In the object field, click on the lookup and filter on prline. Select the PRLINE value.

The child, PRLINE, has a parent of PR. Therefore, select PR from the lookup on the Parent Object Field.



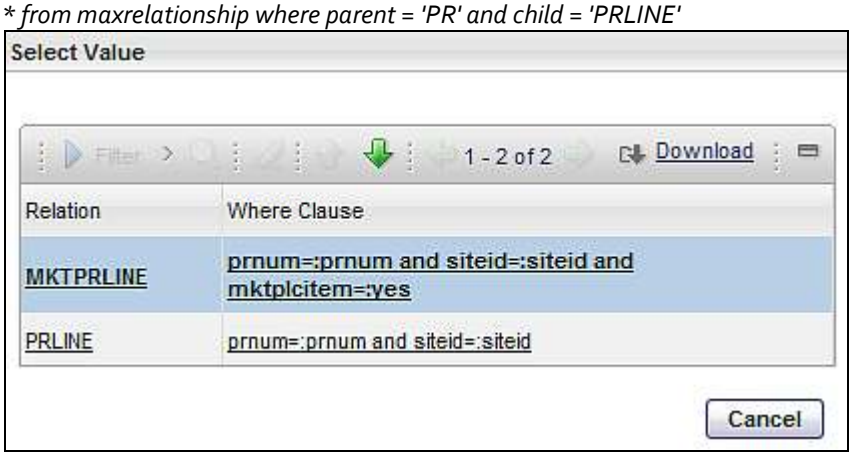
After selecting the Parent Object of PR, the read only field of Object Location path is updated to be 'PR/PRLINE' to indicate the hierarchy path.



7. Enter a Reporting Description for this category that users will quickly identify in the Create Window UI. This example will use a description of 'PR Line Details'.

8. Next, select the Relationship between the Parent PR and the Child PRLINE Table by clicking on the relationship lookup. This lookup displays values from the maxrelationship table.

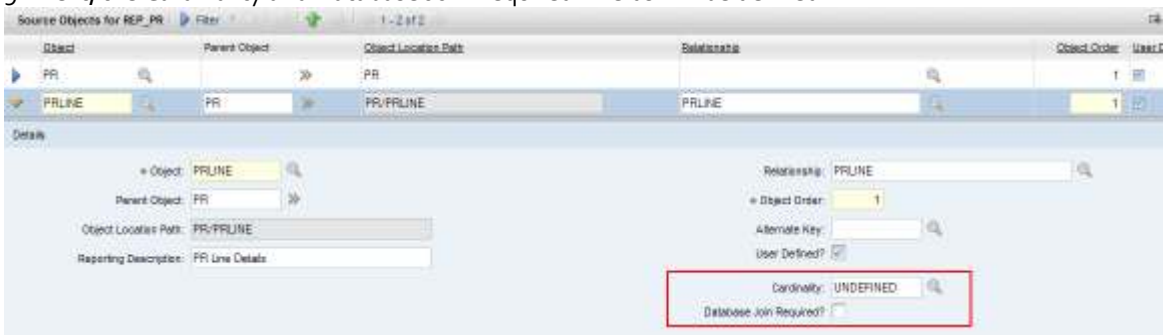
*select * from maxrelationship where parent = 'PR' and child = 'PRLINE'*



Select the correct relationship for the ROS you are creating.

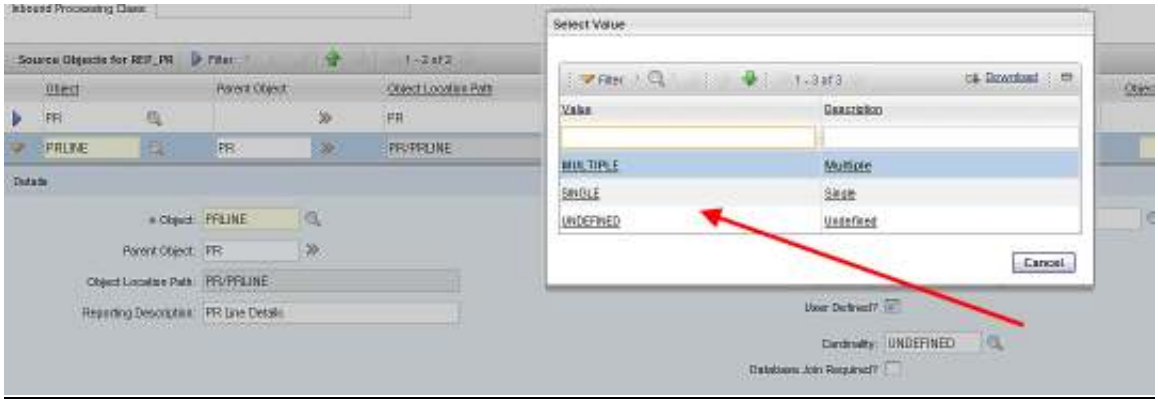
Notes: Best practices for selecting the correct relationship are contained at the end of this section.

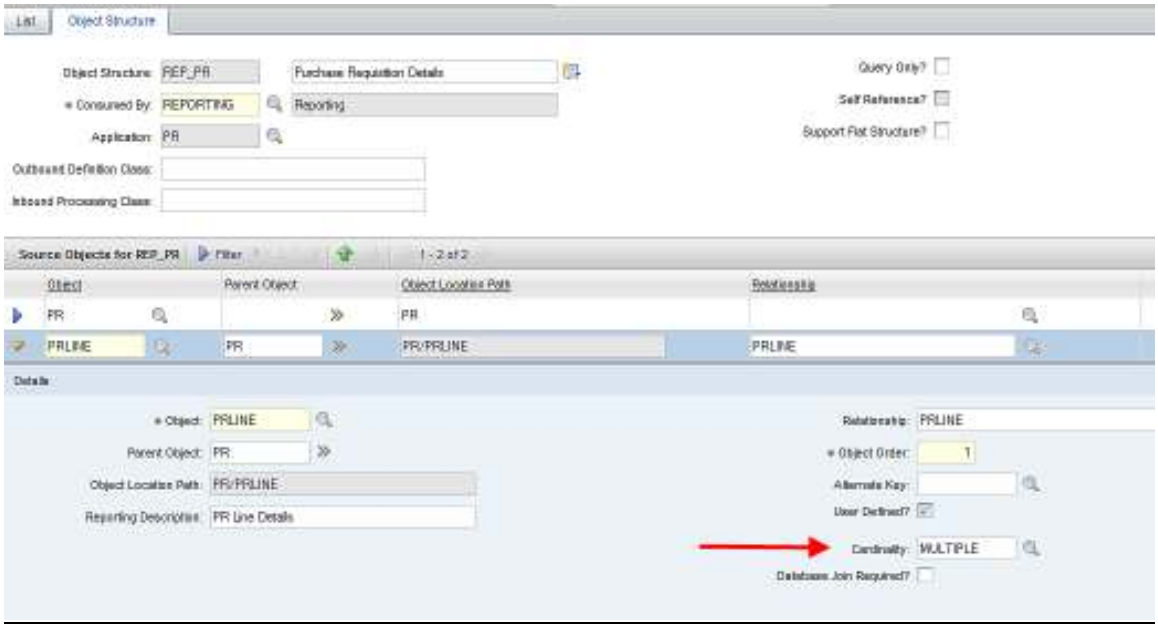
9. Next, the Cardinality and Database Join Required? fields will be defined.



The cardinality value defaults to UNDEFINED, which is assumes a multiple relationship. Select the relationship type that is applicable. In this example, Multiple is selected because multiple PR Lines can be associated to a single PR. (One to Many, or 1:N Relationship)

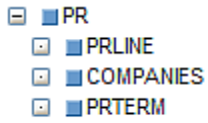
Enable the 'Database Join Required Field' if for every PR record there is a corresponding record in the PR Line table.



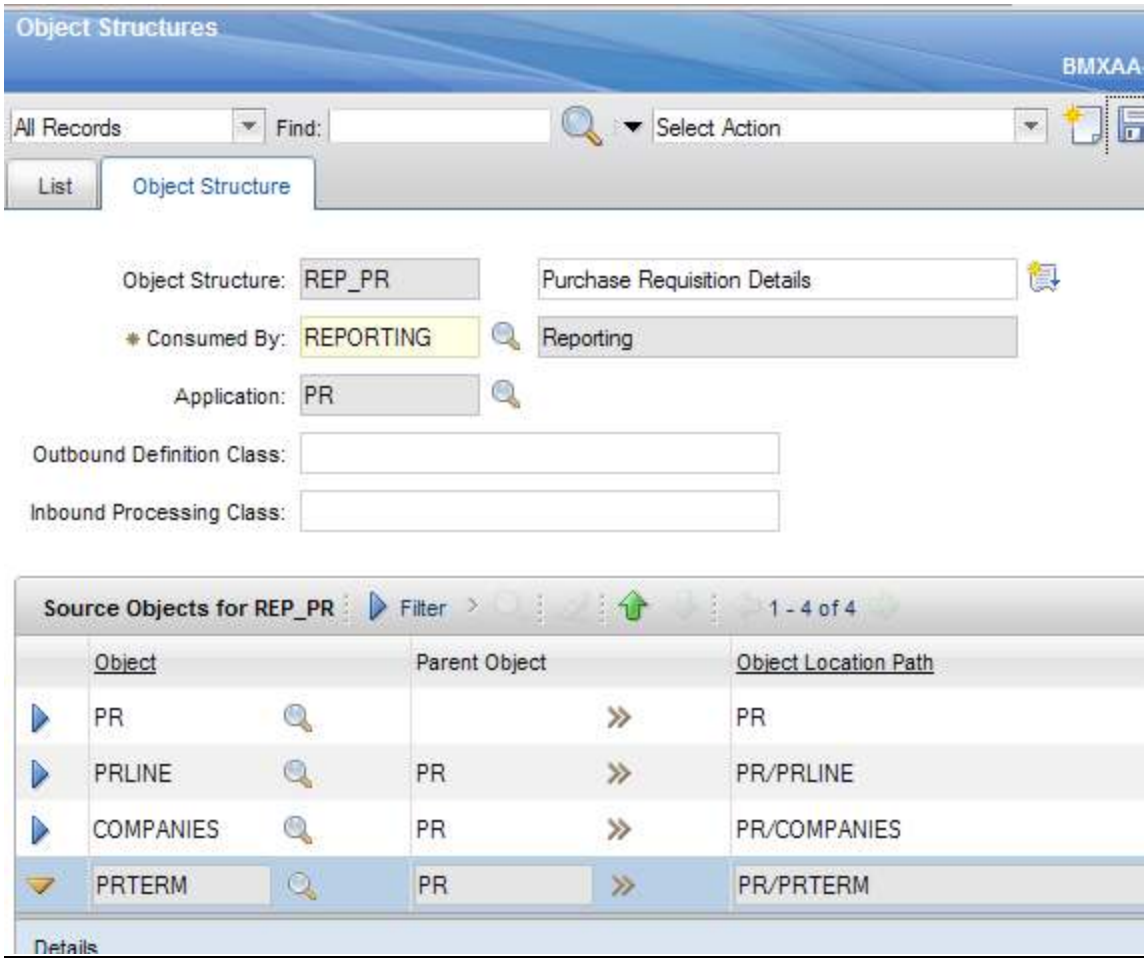


*NOTE: When you define the Cardinality and Database Join Required fields, these settings apply to the relationship, not to the object structure. So once they are set they apply to any ROS using that relationship. If you change them in one ROS definition, they will be changed for the others too.

10. After saving, Click New Row again and repeat this process of adding a Source Object for the Child PR Companies and PR Terms. Notice that when you click on the Parent Object lookup this time, you can start to view how the ROS hierarchy is being built.



11. Save the ROS with the new Companies and PRTERM Child objects.



3.2.1 Excluding Persistent Fields from Source Objects

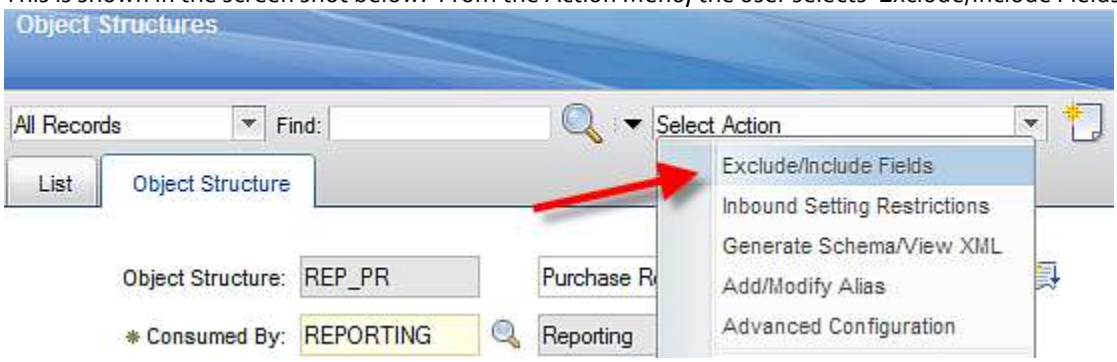
This next section is optional. However, it is highly recommended that you review its functionality as it will improve the Usability features of Ad Hoc reporting for your users.

By default, when a ROS is created, the fields displayed in each Category are the Persistent Fields of the Object. Any non-persistent object fields will not be available in the ROS and Ad Hoc reporting.

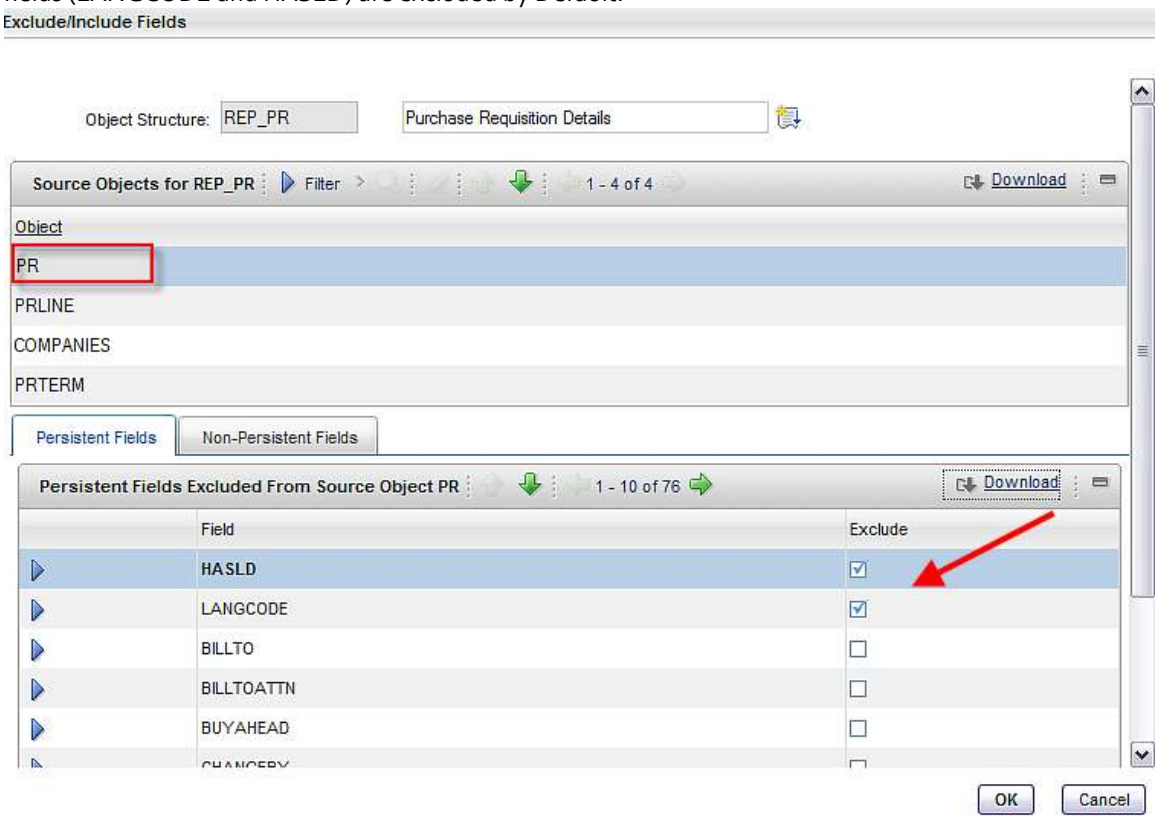
Please note however, that the number of fields for each category could be very large. Additionally, some fields may repeat themselves between the parent and the child. (ex: PR Number exists in both the Parent PR Object and the Child PR Line Object.)

Therefore, to reduce the potentially high amounts of fields that the user has to filter through, the Object Structure Action of Excluding/Including fields can be used. This functionality enables the administrator to refine the field selection for each Source Object used by reporting. Without this functionality, the vast amounts of data available to the user creating his QBR report could become un-manageable.

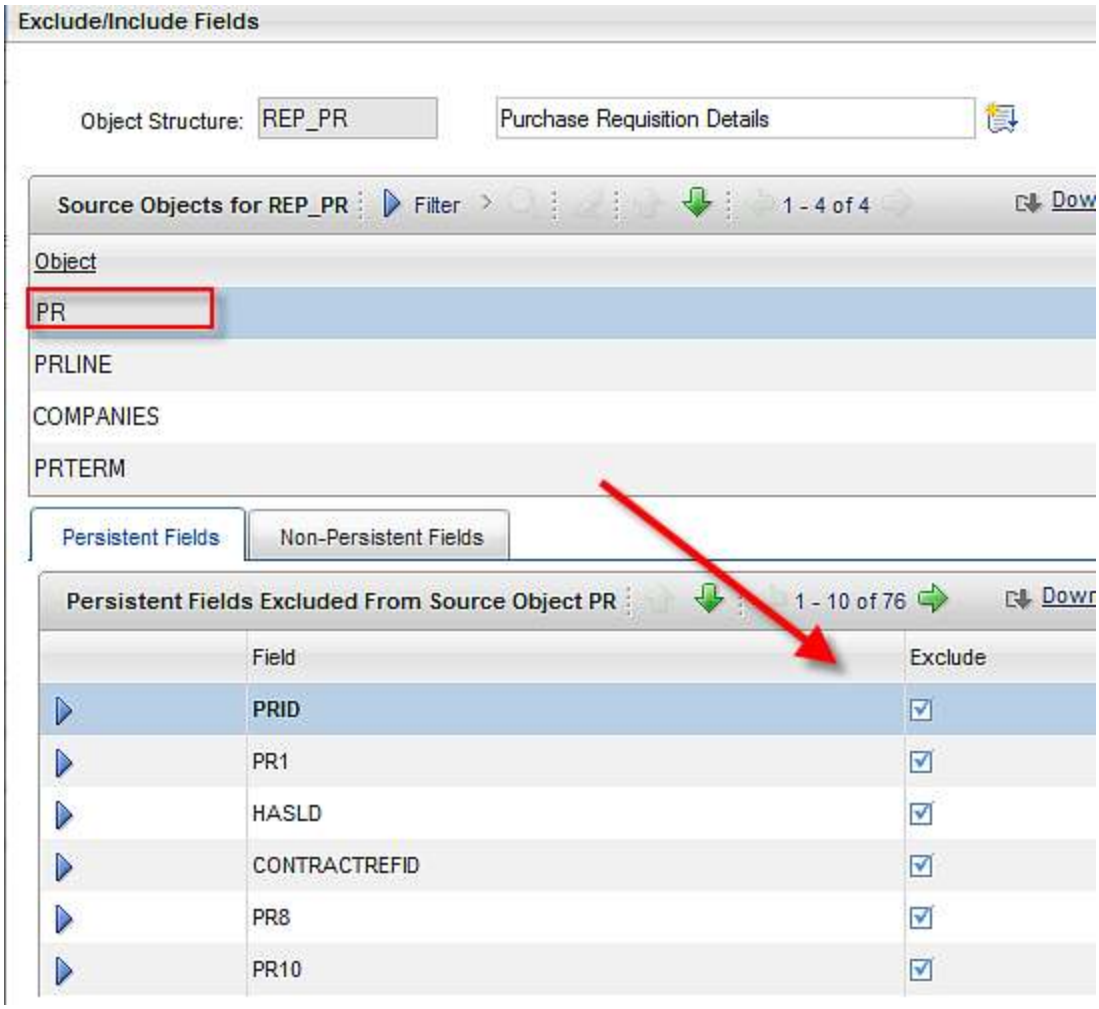
This is shown in the screen shot below. From the Action Menu, the user selects 'Exclude/Include Fields'.



For each of the Source Objects, the Administrator can enable the check box to Exclude all the fields he does not want to display for the Source Object. The screen shot below shows the Parent PR Object, two fields (LANGCODE and HASLD) are excluded by Default.



The administrator can then choose to exclude any other PR Fields that would not be used by the user creating an Ad Hoc report. An updated example is shown below.



Again, this is an optional mechanism that can be utilized to reduce the potentially overwhelming number of persistent database fields displayed to the user during Ad Hoc report creation.

Any excluded fields are held in the MAXINTOBJCOLS table. Therefore, only the *persistent* fields that are NOT contained in the MAXINTOBJCOLS table display as available Category Fields.

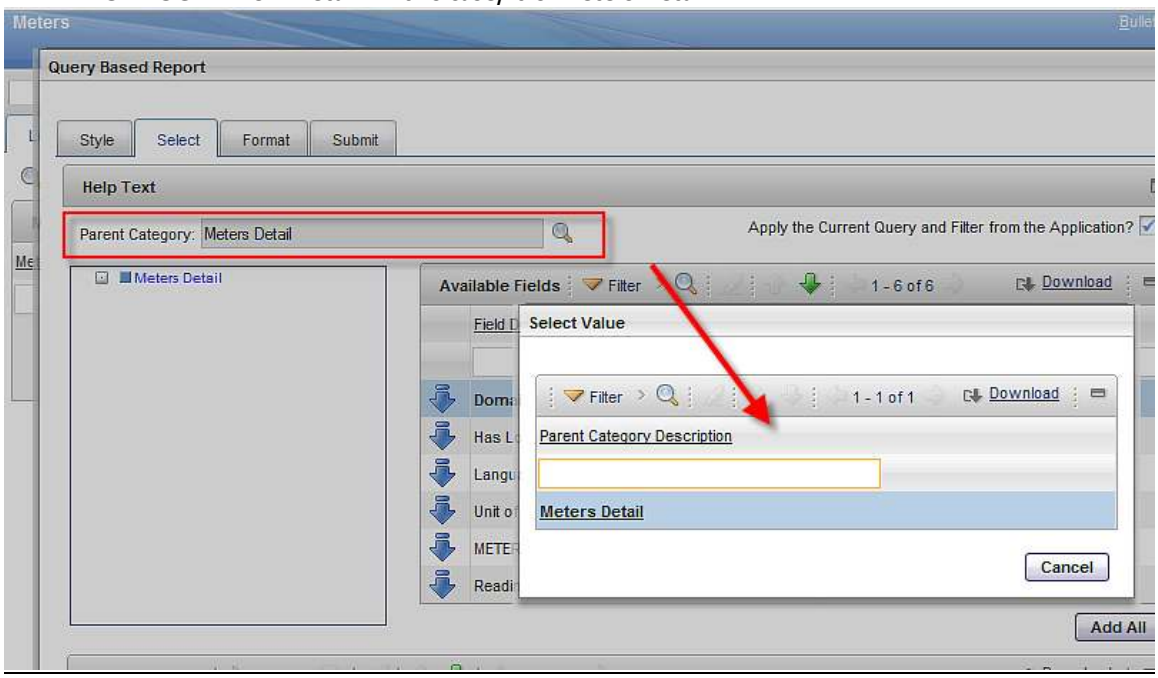
Note that even though Non-Persistent fields are displayed in this dialog in the Object Structure Application, they can not be used for Reporting Objects. This is because reports execute directly against the database (persistent fields). Reports do not execute against the maximo business objects, therefore, non-persistent fields are not used in Ad Hoc reporting.

*See Section 3.2.7 below for details on maxrelationships and non-persistent fields.

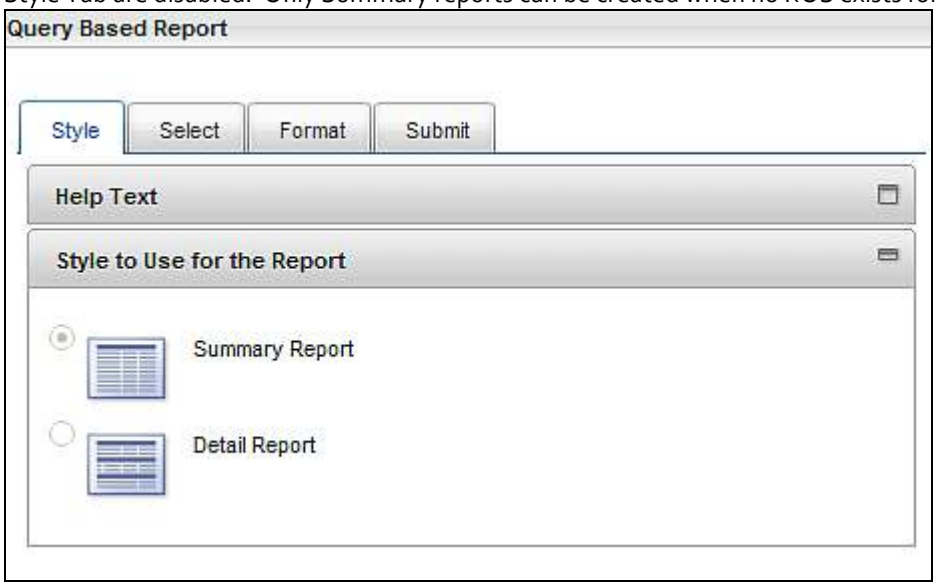
3.2.2 Can Ad Hoc Reports be created if a ROS for an App is not available?

Applications do not need to have Reporting Object Structures (ROS) defined to enable Ad Hoc reporting for that application.

Below is an example of what displays when *no* Report Object Structure is defined for the application. The category drilldown only displays a single level. No children or grandchildren categories are enabled. Additionally, the category description shown on the left hand side, uses a default value of MAXAPPS.DESCRIPTION Detail. In this case, it is 'Meters Detail'.



Also, to recap, when no ROS is available for an application, the Summary and Detail Radio Buttons on the Style Tab are disabled. Only Summary reports can be created when no ROS exists for an application.



3.2.3 Best Practices on Creating Report Object Structures

1. Individuals creating Report Object Structures must be very familiar with the Maximo Database Structure and Maxrelationships. Additional information on the Database Structure can be obtained from the Entity Relationship Diagram (ERD). This diagram can be obtained at the URL below.

https://www-304.ibm.com/software/brandcatalog/ismlibrary/details?catalog.label=1TW10MA25

2. There are no limits to the numbers of child objects that you can configure for a ROS. As long as each Child Object has a valid maxrelationship, it can be utilized.

3. ROS can be created against database views along with database tables, as long as the database views are valid Maximo Objects. An example of a Parent Object that is a database view is TOOLITEM for the Tools application.

4. In releases prior to 7.5, you were unable to add the same object multiple times in a single Report Object Structure due to QBR Business Rules. This functionality has been enhanced in the 7.5 release, and you can include the same object multiple times as shown below with the Labor Child Object

Work Order
- Assets
- Locations
- Planned Labor
  - Labor
- Actual Labor
  - Labor

5. When the ROS is being created, and the definitions of the maxrelationships, cardinality and other fields are being defined, you may want to execute QBR reports from the Maximo Browser to insure the expected results return. However, if a change is made to these fields in the Object Structure application, you must close down any open QBR window in order for the changes to take affect. Additionally, if you change these fields via backend database scripts, you must restart the application server for the changes to take affect.

This occurs because the QBR Functionality caches queries so it doesn't need to re-fetch data that it expects hasn't changed (ex. data from MAXRELATIONSHIP)

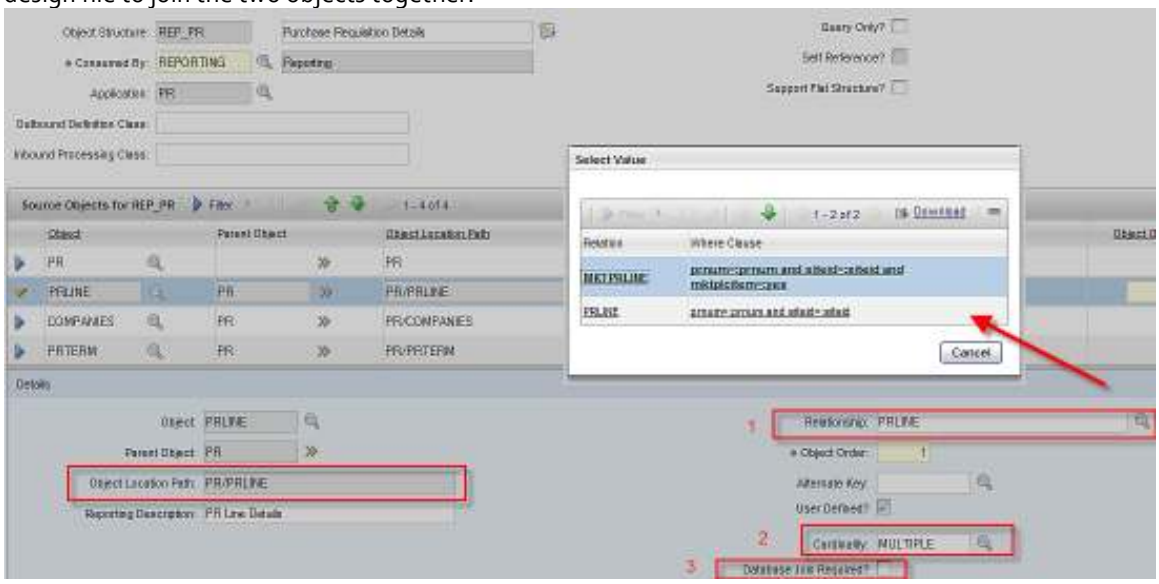
3.2.4 Understanding Database Joins and Cardinality

The Report Object Structures create a hierarchy of related categories for an application. The ROS utilize three critical pieces of information which becomes the basis of the report sql query for the Ad Hoc report. These three components are:

1. How are the objects joined together?
2. What type of Cardinality does the joined objects use?
3. Is the Database Join between the 2 Objects required?

Each of these is very critical as they determine what data is returned in the Ad Hoc report. If the values are set incorrectly, the data returned could either be incorrect or un-usable.

The three areas are highlighted below. The first component – how are the objects joined together – is defined via the selected maxrelationship. The maxrelationship sql is used within the Ad Hoc report's design file to join the two objects together.

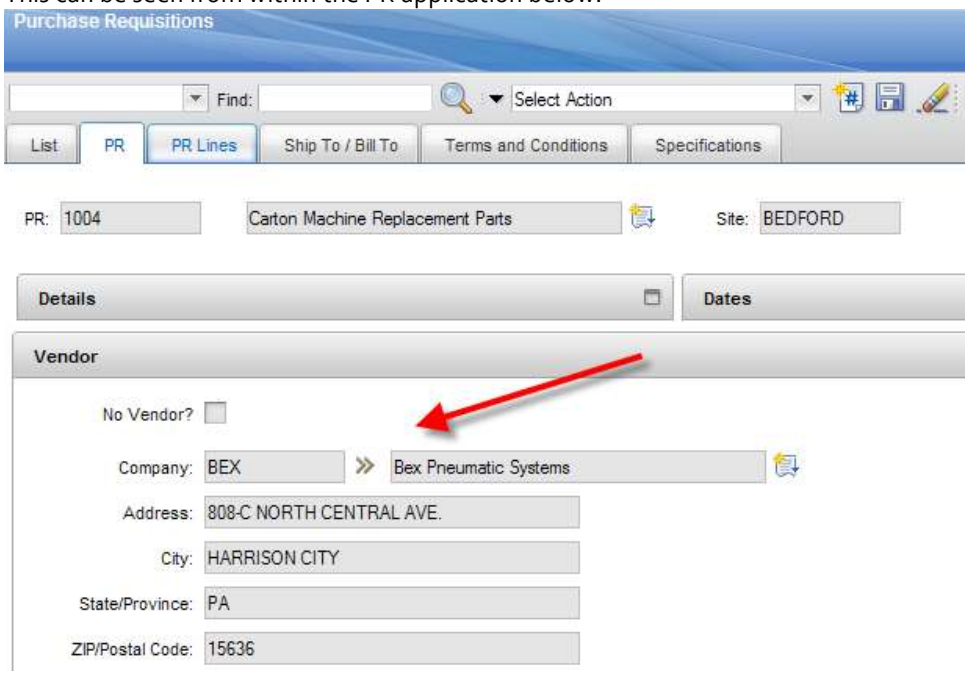


The other two fields – Cardinality and Database Join Required – are unique to ROS. The values for these fields are held in MAXRELATIONSHIPS, and more details on each of these two fields is below.

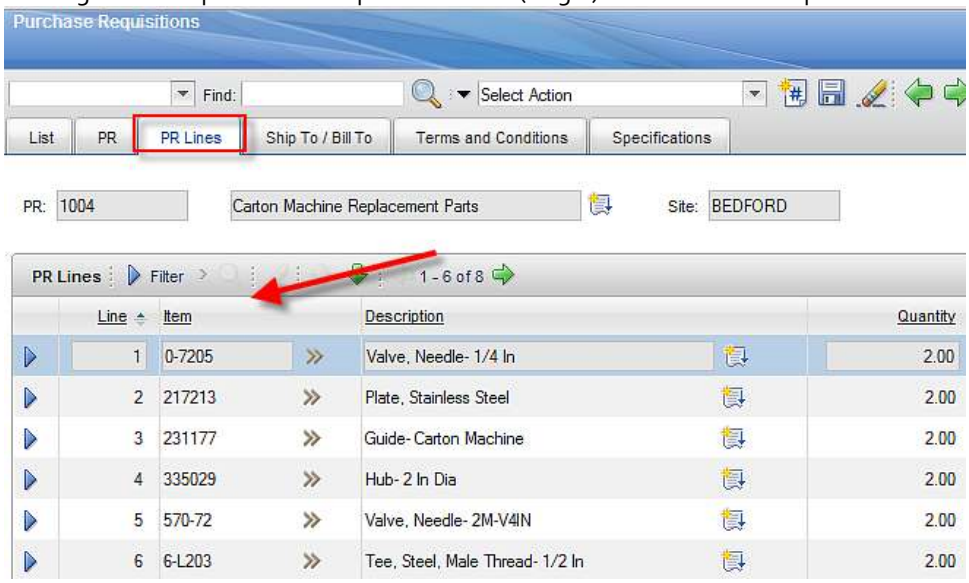
3.2.5 MAXRELATIONSHIP.CARDINALITY

Cardinality is a database term defining the type of relationship. In the case of reporting, there are two types: One-to-one or One-to-many. One-to-one relationships are known as Single Relationships, whereas One-to-many are Multiple Relationships.

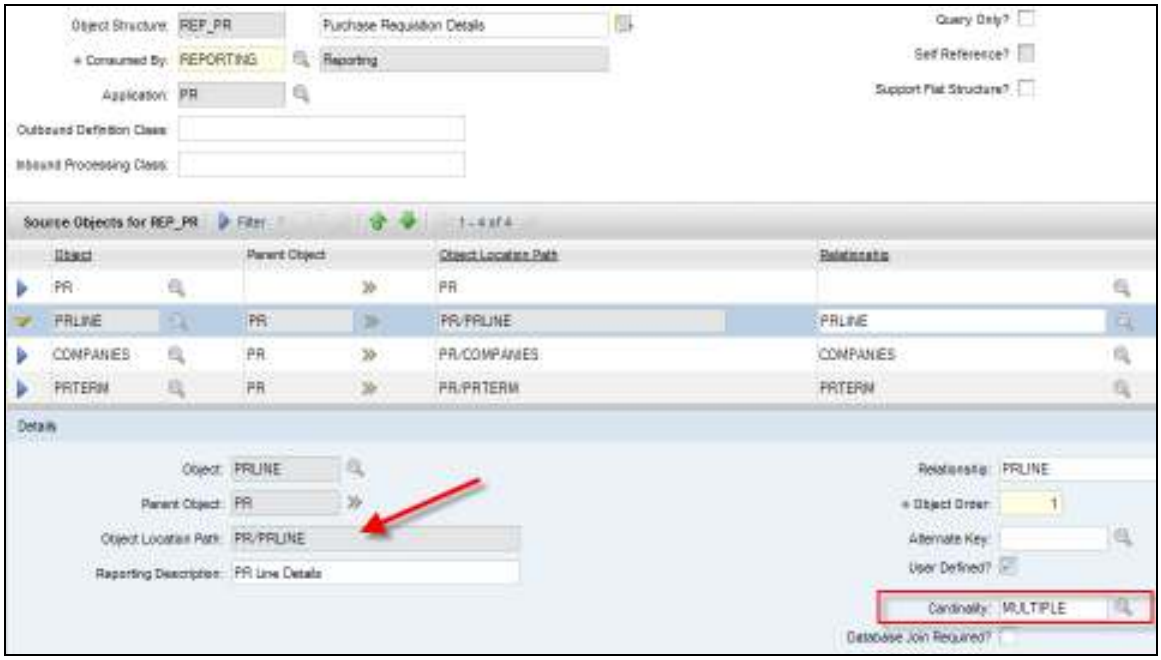
If it is a single relationship, it means it is a one to one relationship (1:1). Using our example, the PR-Companies relationship is a single relationship as only a single Company can be associated to a single PR. This can be seen from within the PR application below.



If it is a multiple relationship, it means it is a one to many relationship. (1:N) The PR-PR Lines Relationship is a single to multiple relationship. For each PR (single) there can be multiple PR Line records (multiple).



Once the relationships between objects are understood, the cardinality can be set properly within the Object Structure Application. This screen shot shows the Multiple Cardinality, or 1:N Relationship for PR to PRLINE.



The MAXRELATIONSHIP.CARDINALITY field is a domain field with three values: Single, Multiple or undefined.

1. The default value is UNDEFINED.
2. Values of Undefined, Null and Multiple are all treated as Multiple relationships.

During the report creation process, as the user adds fields to the Selected Field Section, their relationships are evaluated to determine if they are single or multiple relationships

Any number of categories with Single Relationships can be used in Summary or Detail reports. However, for Multiple Relationship Categories, Summary reports can only use one while Detail reports can utilize any number.

3.2.6 MAXRELATIONSHIP.DBJOINREQUIRED

As mentioned earlier, the type of database join between two tables determines the data that is returned in a report. It is probably one of the most important aspects in defining the report relationships.

There are two types of joins. A required database join, and an optional, or left outer join. The type of join used in the maxrelationship will determine the type of data returned in the Ad Hoc report.

If the join is defined as required, or an inner join, a '1' will be used in the MAXRELATIONSHIP.DBJOINREQUIRED field. This type of join will only return results that have corresponding data in both tables. These inner joins limit the data that is returned.

If the join is defined as NOT required, or an outer join, its value will be '0'. This is the default value. This type of join will return results that may or may not have corresponding data.

Looking further at the relationship between the PR to PR Line table -
If the MAXRELATIONSHIP.DBJOINREQUIRED = '1', only PRs with entries in the PRLINE table will be returned in the report.

If the MAXRELATIONSHIP.DBJOINREQUIRED = '0', any PR – with or without entries in the PRLINE table - will be returned in the report.

In this case, you can create and save a PR without defining any PRLINES, so the DBJOINREQUIRED = 0.

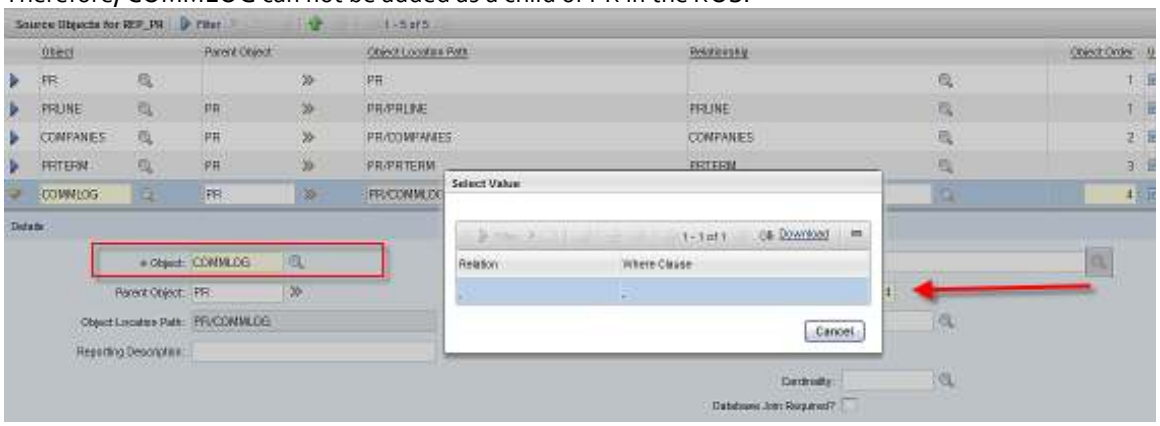
Note: For Detail reports, the DB Join Required field does not apply to the relationship between the Category sections and the Header section. For example, if PR is in the header and PRLINE is in a separate Category section, PRs will show in the header even when there are no related PRLINES . This is because separate SQL queries are executed to retrieve the data for these sections.

3.2.7 Business Rules for ROS maxrelationships

Review the key Business Rules below to create ROS that can be utilized for QBR Reporting

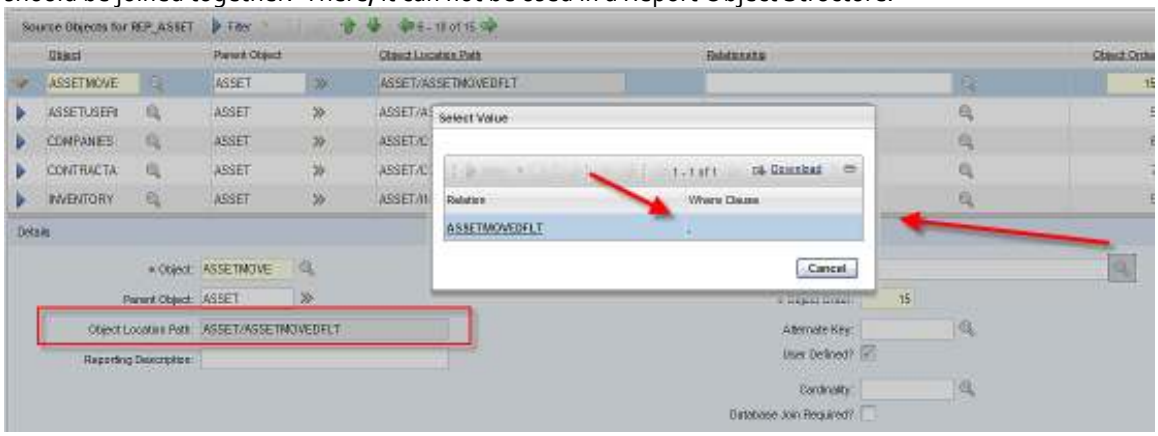
1. Each Child, or Source Object, must have a relationship to the Parent Source Object to be valid. Without the relationship, the tables can not be joined, causing the QBR report to error out.

This is shown in the PR ROS below. The COMMLOG Object is added as a child of PR. However, COMMLOG has no relationship to the PR parent as shown by the null values in the relationship lookup. Therefore, COMMLOG can not be added as a child of PR in the ROS.



2. Each child, or source object, must be joined to its parent with relationships where the database sql is defined. If the source object has a null database sql, it can not be used as it will give invalid results.

This is shown below using an Asset ROS example. A Child Object of ASSETMOVEDFLT is added to the Parent Asset Object. When clicking on the relationship lookup, notice that its database sql is null. Even though a maxrelationships exists between the two tables, there is no database sql on how those two tables should be joined together. There, it can not be used in a Report Object Structure.



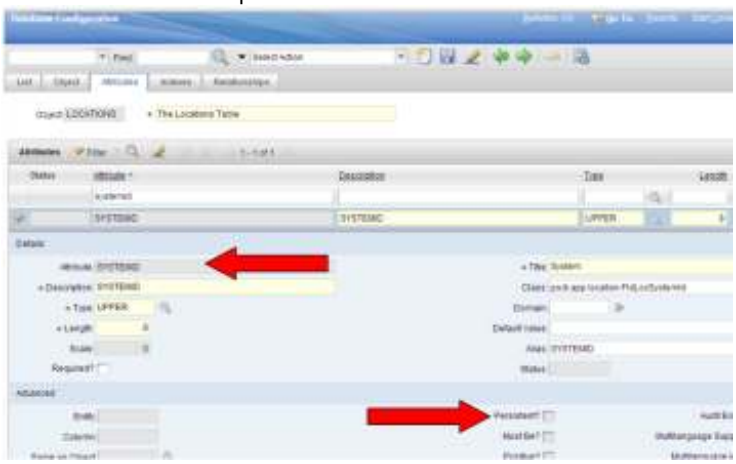
3. The relationship used between a child and a parent must not contain any non-persistent fields. Only persistent attributes can be used in Ad Hoc reporting.

An example that highlights this is with the REP_LOCATION ROS. The ROS was extended to include the child object, LOCHIERARCHY to the parent object, LOCATIONS. The relationship INVSYSTEM_PARENT was selected, which is defined as

location=:parent and systemid=:systemid and siteid=:siteid

However, because systemid is a non-persistent field, the relationship cannot be used within the ROS to create a report.

If you are unsure if a field is persistent or not, go to the Database Configuration application. Select the Object, and then go to the Attributes tab. Search on the selected attribute, and under its Advanced section, see if the Persistent? Field is checked. If it is not , the field is non persistent and cannot be used in a ROS maxrelationship.



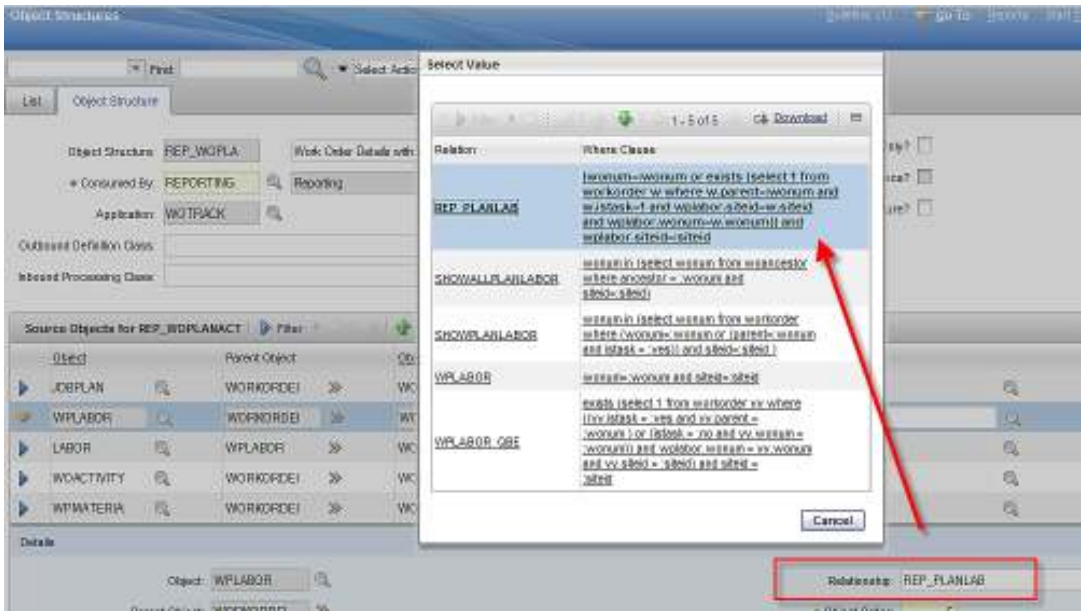
4. Additional restrictions exist on the format of the where clause used in a QBR relationship:

a. Dates in bind variables are not supported. For example, the following relationship from CALENDAR to WORKPERIOD may not be used:
calnum=:calnum and workdate between :startdate and :enddate and orgid=:orgid

b. Replacement variables such as :&username& are not supported. The variables :yes and :no can be replaced with 1 and 0 for use in QBR relationships

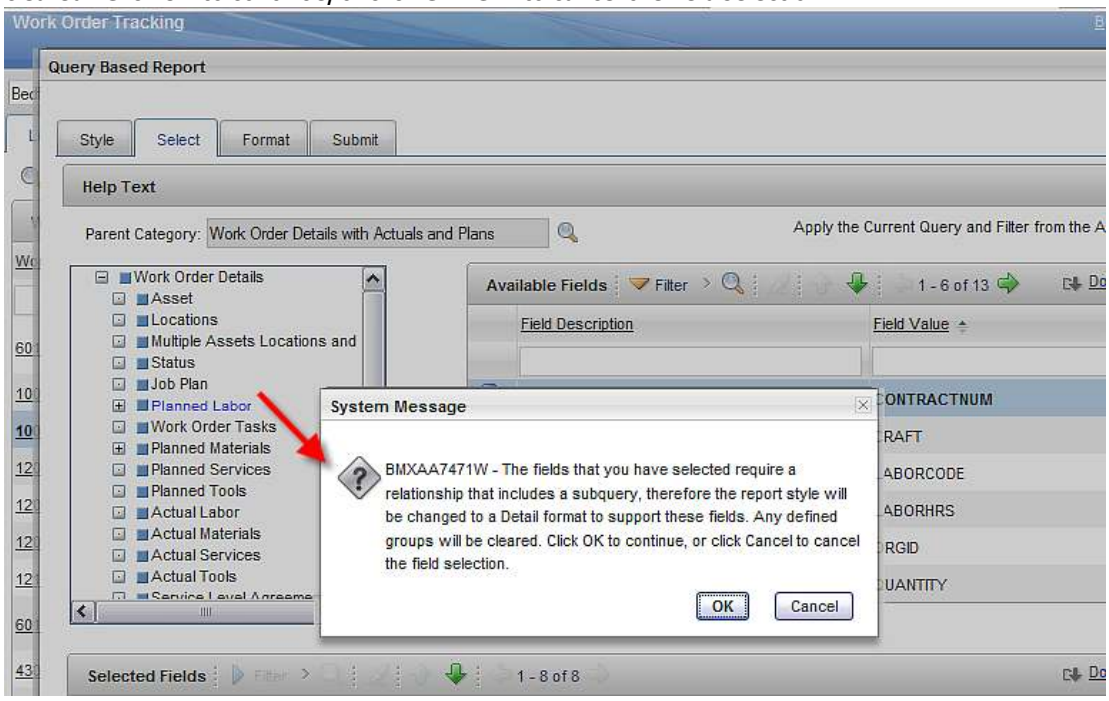
5. If a maxrelationship contains a subquery, it can only be used in Detail reports, not Summary Reports. An example of a maxrelationship containing a subquery is in the out of the box ROS called REP_WOPLANACT. One of its children, WPLABOR has a maxrelationship with a subquery shown below.

(wonum=:wonum or exists (select 1 from workorder w where w.parent=:wonum and w.istask=1 and wplabor.siteid=w.siteid and wplabor.wonum=w.wonum)) and wplabor.siteid=:siteid



If you select a Summary report, and try to select fields from the WPLABOR Child, a message will display that the report type needs to be converted to a Detail report.

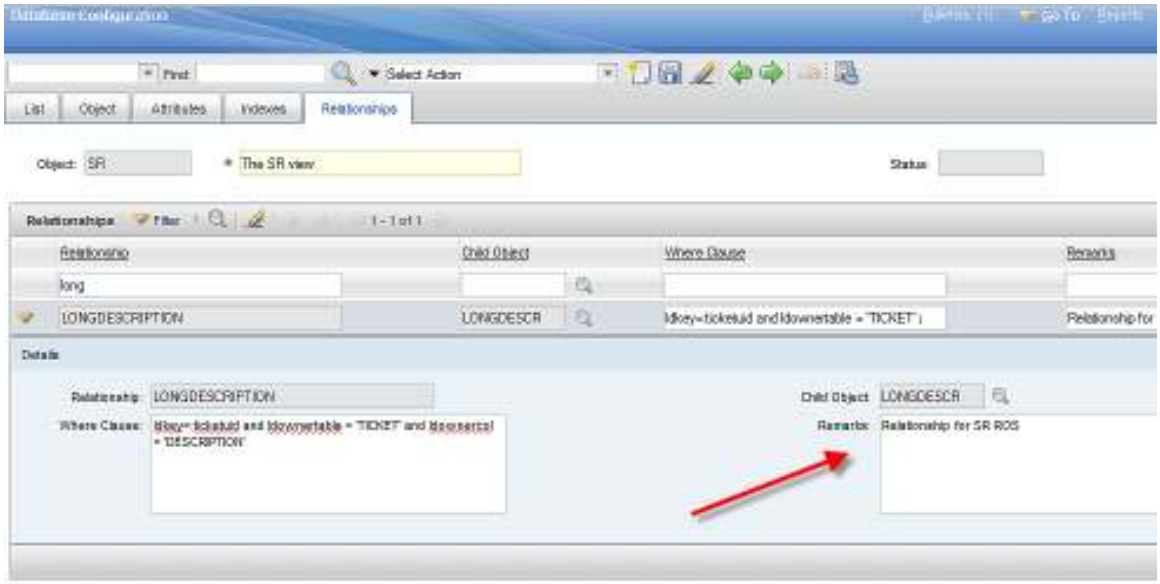
BMXAA7471W - The fields that you have selected require a relationship that includes a subquery, therefore the report style will be changed to a Detail format to support these fields. Any defined groups will be cleared. Click OK to continue, or click CANCEL to cancel the field selection.



Additionally, maxrelationships containing subqueries are not allowed at the grandchild level because nested Detail reports are not supported. If you try to do this, the message below will display.

> This relationship includes one or more subqueries. A report can only contain this type of relationship between the main category and its children.

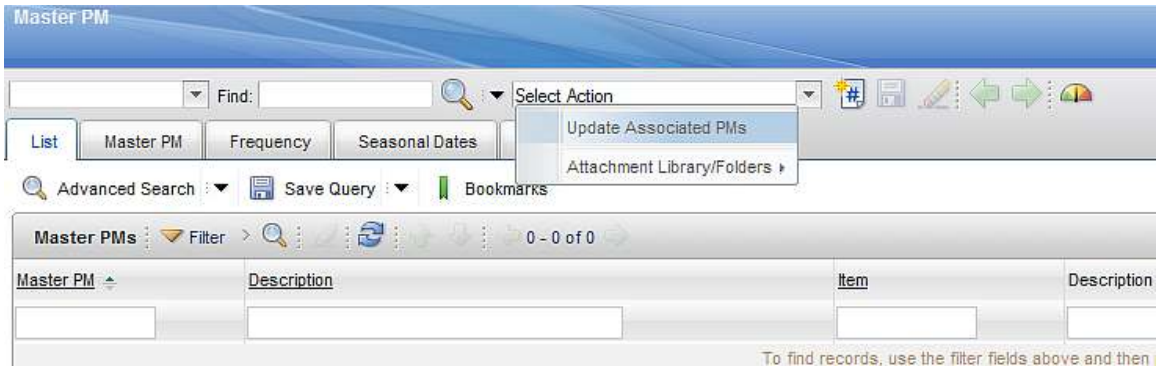
6. Also, you may find it useful to create your own unique maxrelationships to be used in the ROS. This can help you better manage and understand the relationships used in your QBR reports. An example of this is a Long Description relationship added below for the SR application – note that its description highlights that it was created exclusively for the SR ROS.



3.2.8 Adding QBR Functionality to New or Cloned Applications

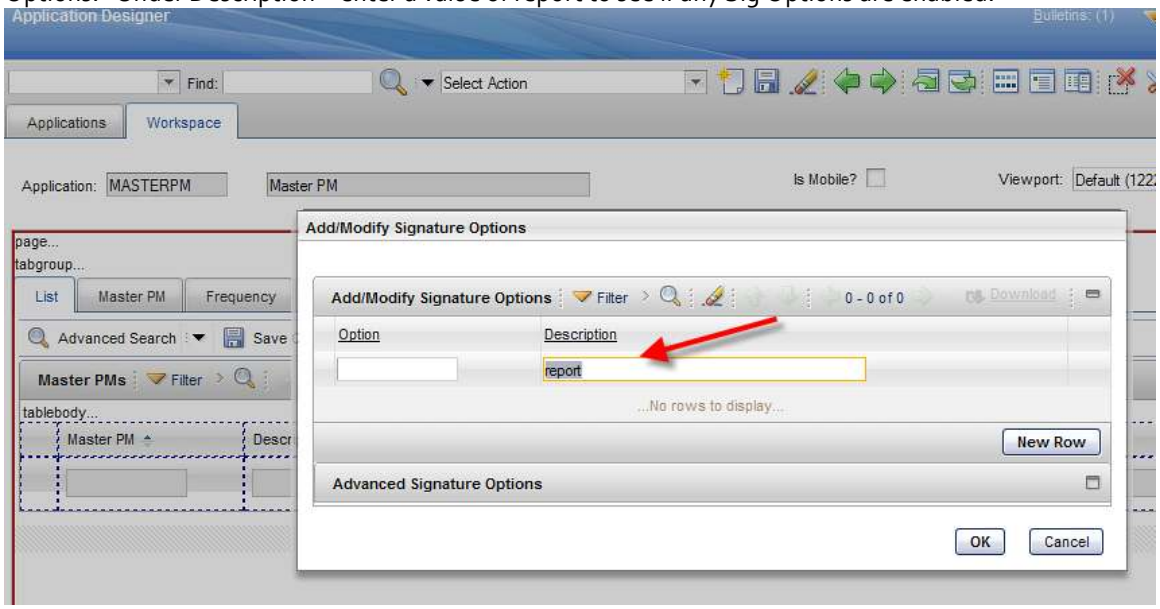
Some applications may not be enabled for Ad Hoc reporting out of the box because they may be administrative intensive, a non-standard application or may utilize complex methods for determining its main table. Examples of some of the applications not enabled for Ad Hoc reporting include Sets, Currency Codes, Exchange Rates, and Self Service applications.

However, you may want to enable QBR Functionality for one of these applications, or for a new or cloned application that you have created. The steps below detail an example of how to do this using the Master PM application as an example.



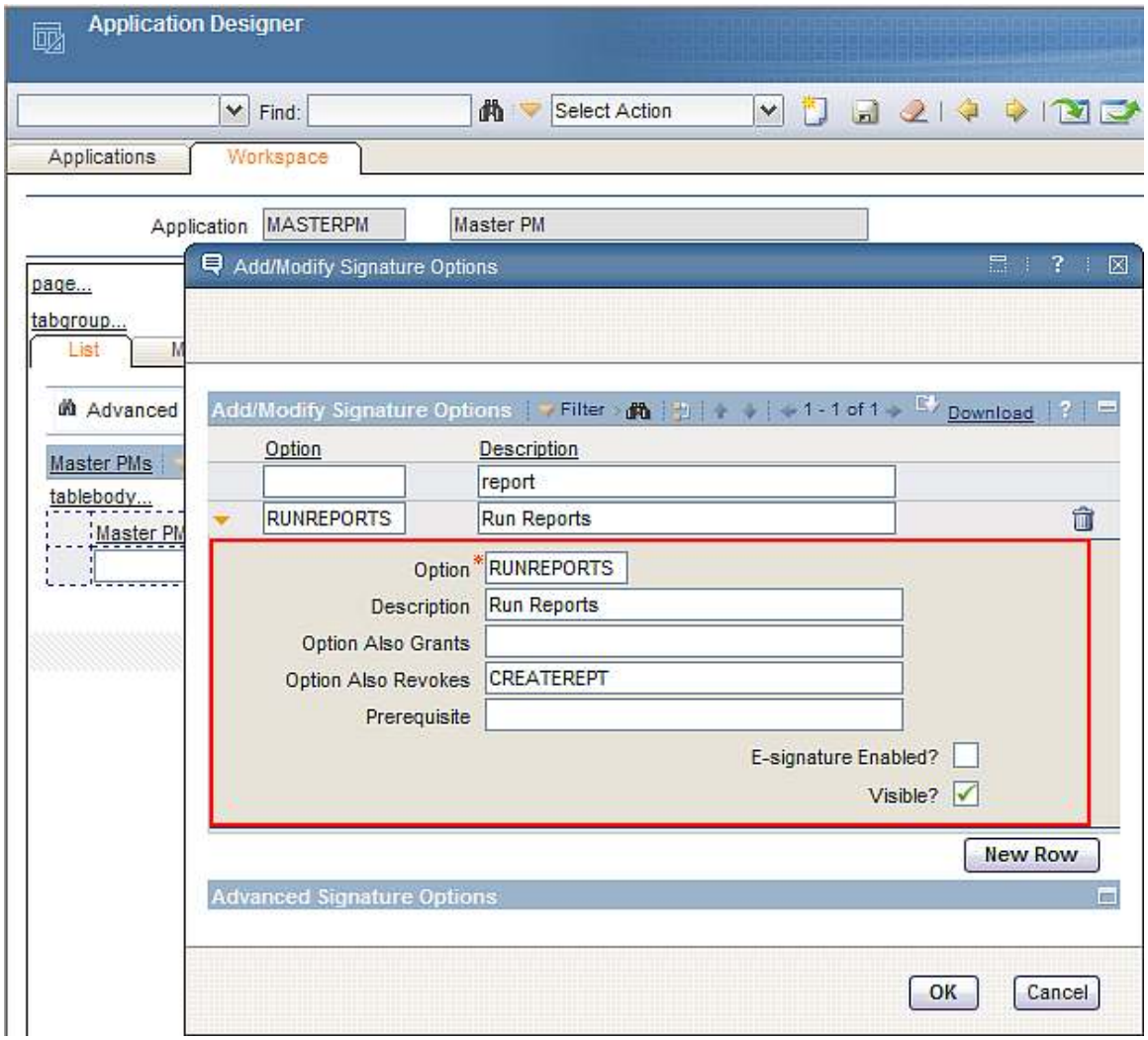
Note: In order to do this, the application must have the RUNREPORTS Security option enabled. If this security option is not available, it must first be added before adding the Create Reports Option.

First, verify that RUNREPORTS Security Option is enabled for the application. To do this, access Application Designer and filter for the application. From the action menu, select Add/Modify Signature Options. Under Description – enter a value of report to see if any Sig Options are enabled.

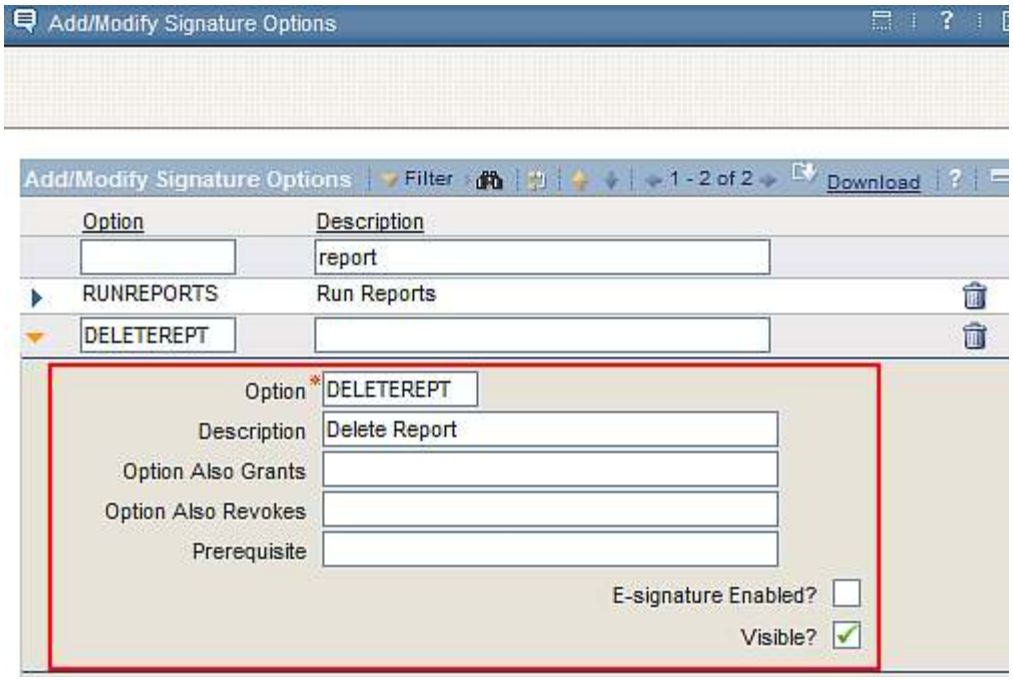


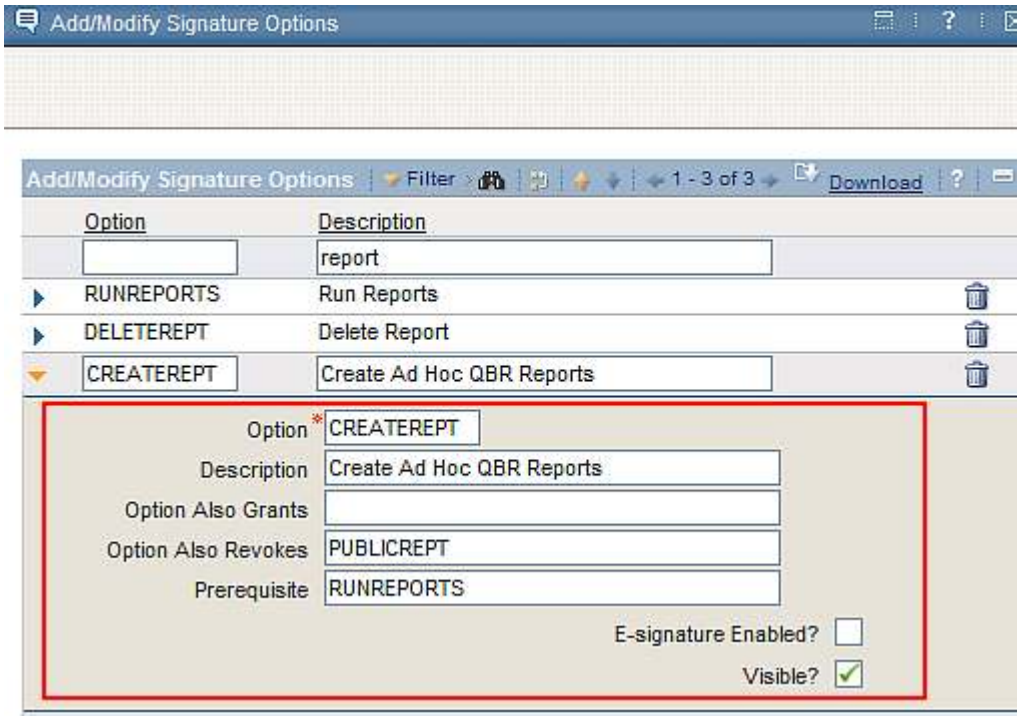
In this case, no sig options are enabled, so they all will be created – including the Prerequisite – Run Reports.

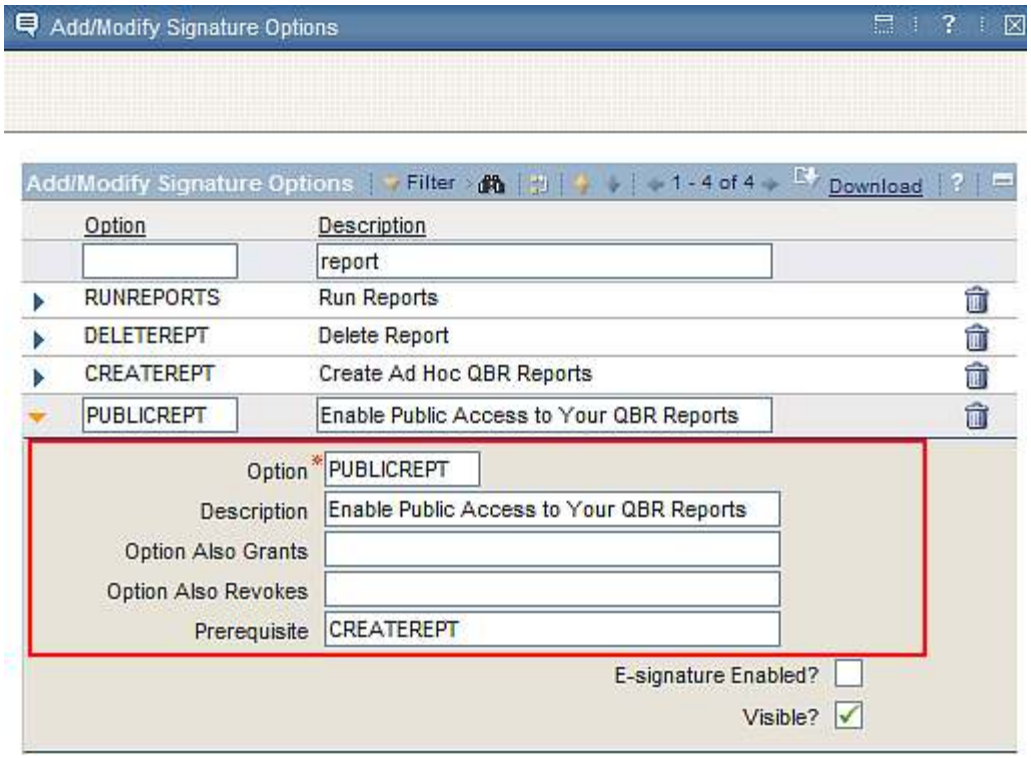
Click New Row, and add the first sig option as shown below for: RUNREPORTS



Repeat this process for the remaining three other sig options: (1) DELETEREPT (2) CREATEREPT and (3) PUBLICREPT

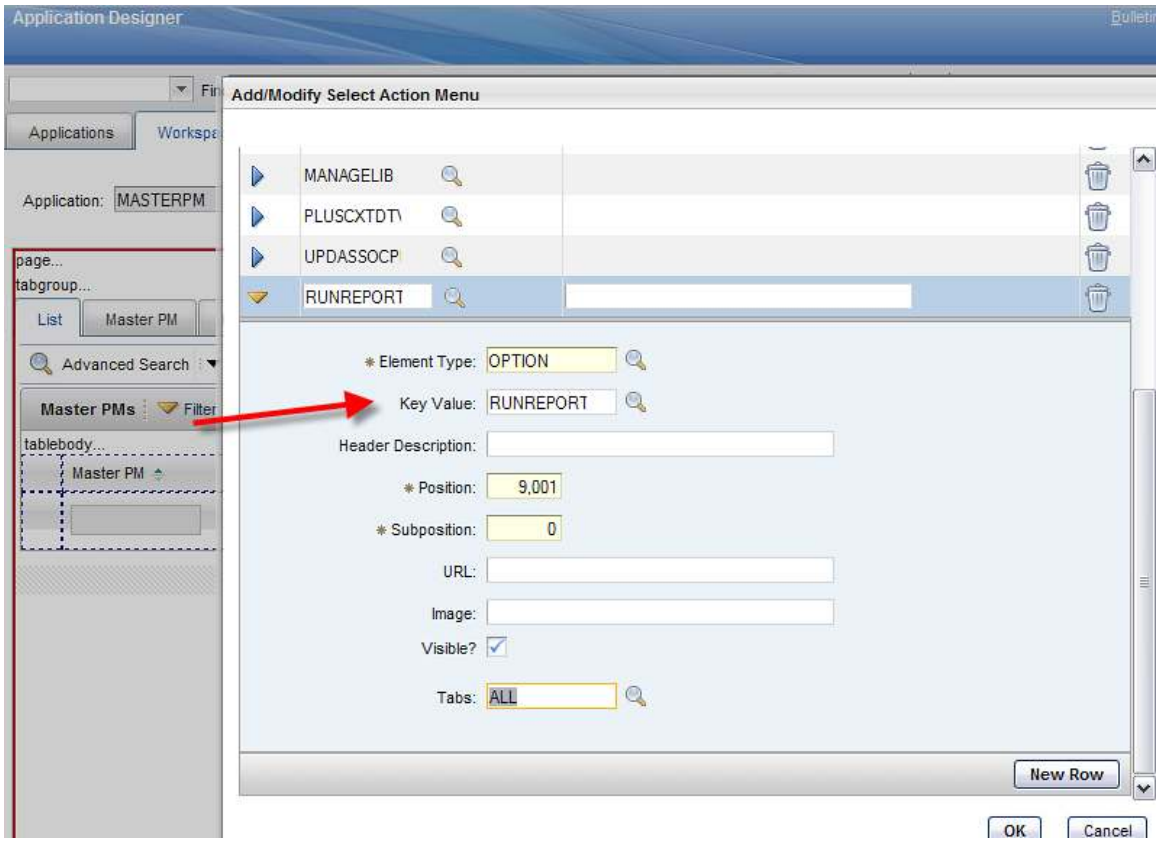

Add/Modify Signature Options
Add/Modify Signature Options
Filter
1 - 2 of 2
Download
Option
Description
report
RUNREPORTS
Run Reports
DELETEREPT
Option *
DELETEREPT
Description
Delete Report
Option Also Grants
Option Also Revokes
Prerequisite
E-signature Enabled?
Visible?


Add/Modify Signature Options
Add/Modify Signature Options
Filter
1 - 3 of 3
Download
Option
Description
report
RUNREPORTS
Run Reports
DELETEREPT
Delete Report
CREATEREPT
Create Ad Hoc QBR Reports
Option *
CREATEREPT
Description
Create Ad Hoc QBR Reports
Option Also Grants
Option Also Revokes
PUBLICREPT
Prerequisite
RUNREPORTS
E-signature Enabled?
Visible?

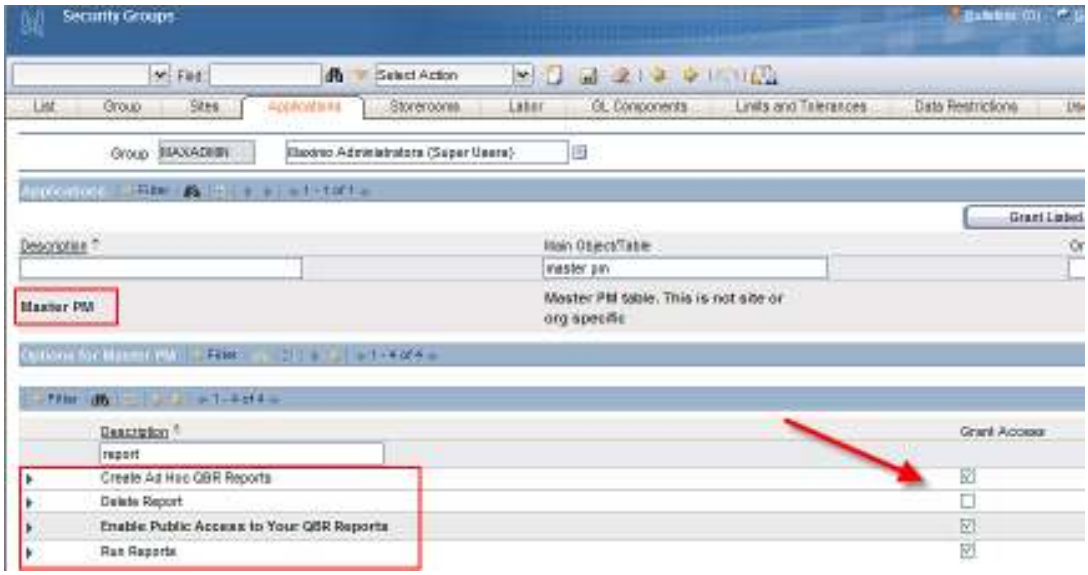


Click OK and save the changes.

Next, add the Run Report Action to the Action menu for the Master PM application. Select ''Add/Modify Select Action Menu'. Input values as shown below and save.



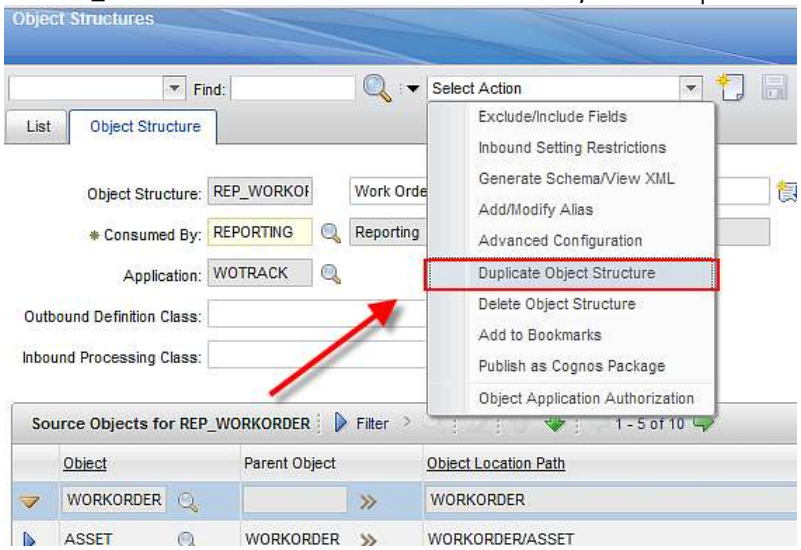
Once the new report security options and menu option have been enabled, follow the standard Application Designer processes for enabling the application updates. You can then grant security privileges to your selected Security Groups via the Security Group application.



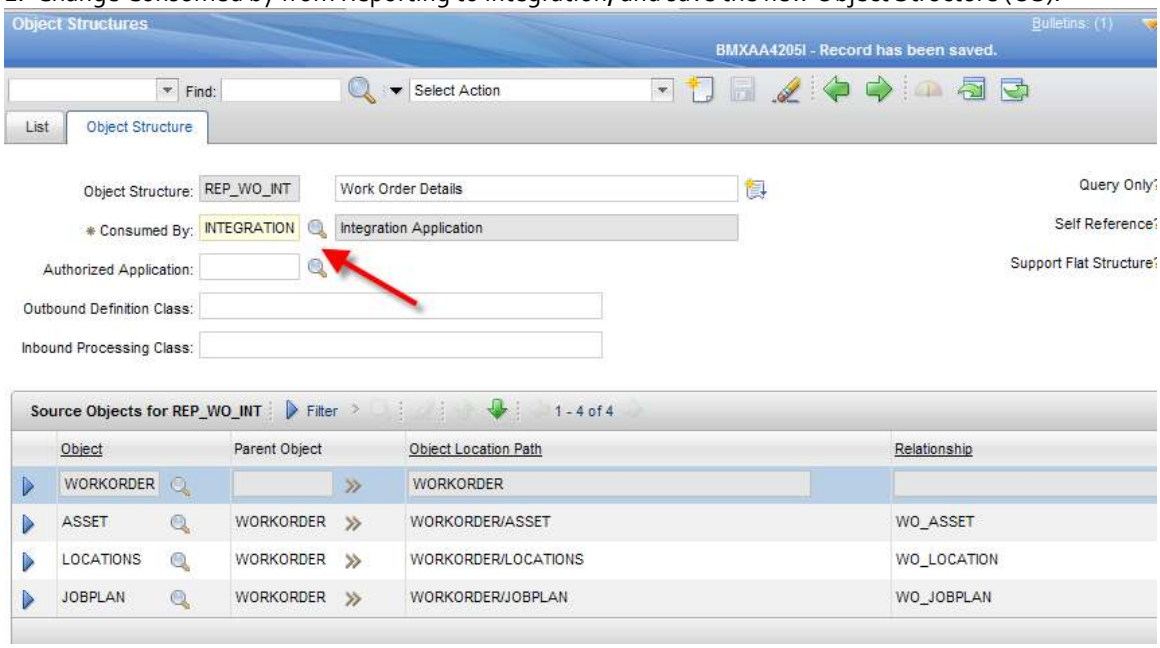
4 Utilizing Report Object Structures in Application export functionality

You can also use your Report Object Structures (ROS) for the Application export functionality. This functionality enables a user to export multiple database object and attribute fields that are defined in object Structures – and enables you to apply the power and flexibility of your ROS in multiple ways. The steps below detail how you can enable this.

1. Access Object Structure application, and select a Report Object Structure. The example below shows the REP_WORKORDER ROS. From the action menu, select 'Duplicate Object Structure'.



2. Change Consumed by from Reporting to Integration, and save the new Object Structure (OS).

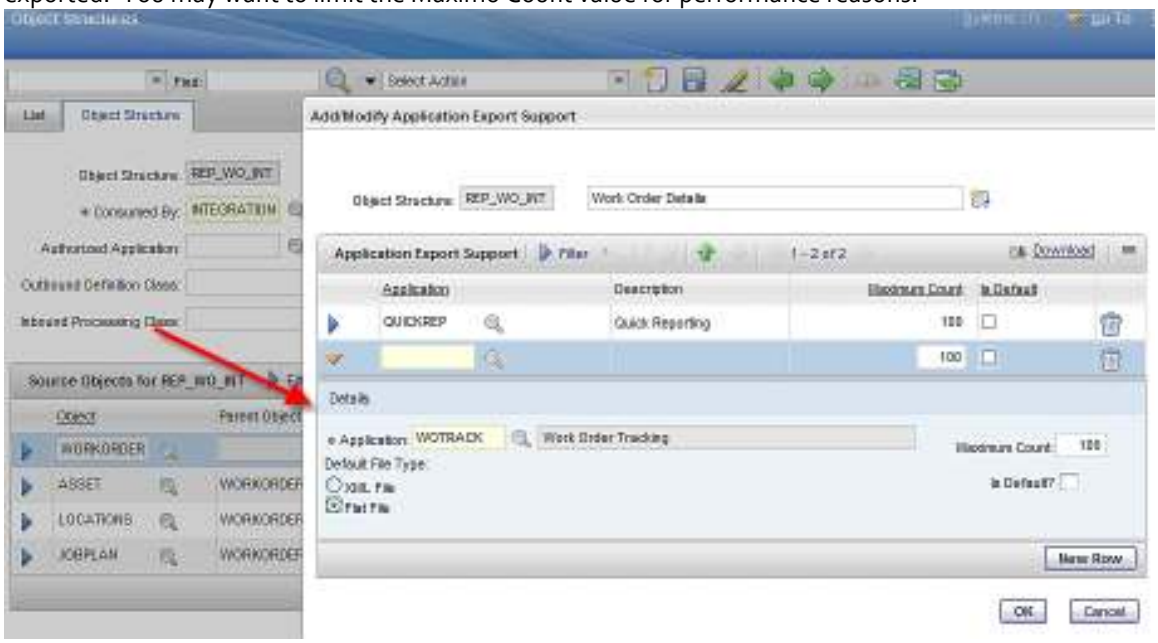


*Note: In this example, a number of child objects have been removed from the duplicated object structure. This was done only for purposes of this example.

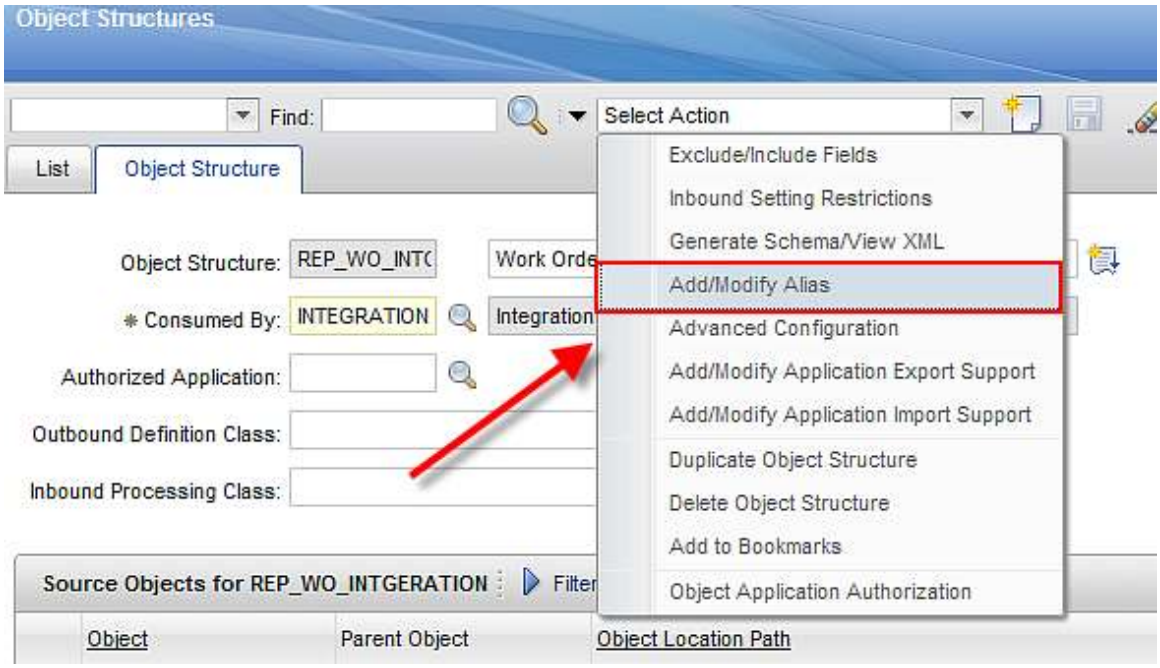
3. Next, from the action menu, select 'Add/Modify Application Export Support'. Additionally, if you want your users to export to xls, enable the 'Support Flat Structure' field.

Select the application where you want the application export functionality to be enabled. You can enable any application whose main object is the same as the parent object of the OS.

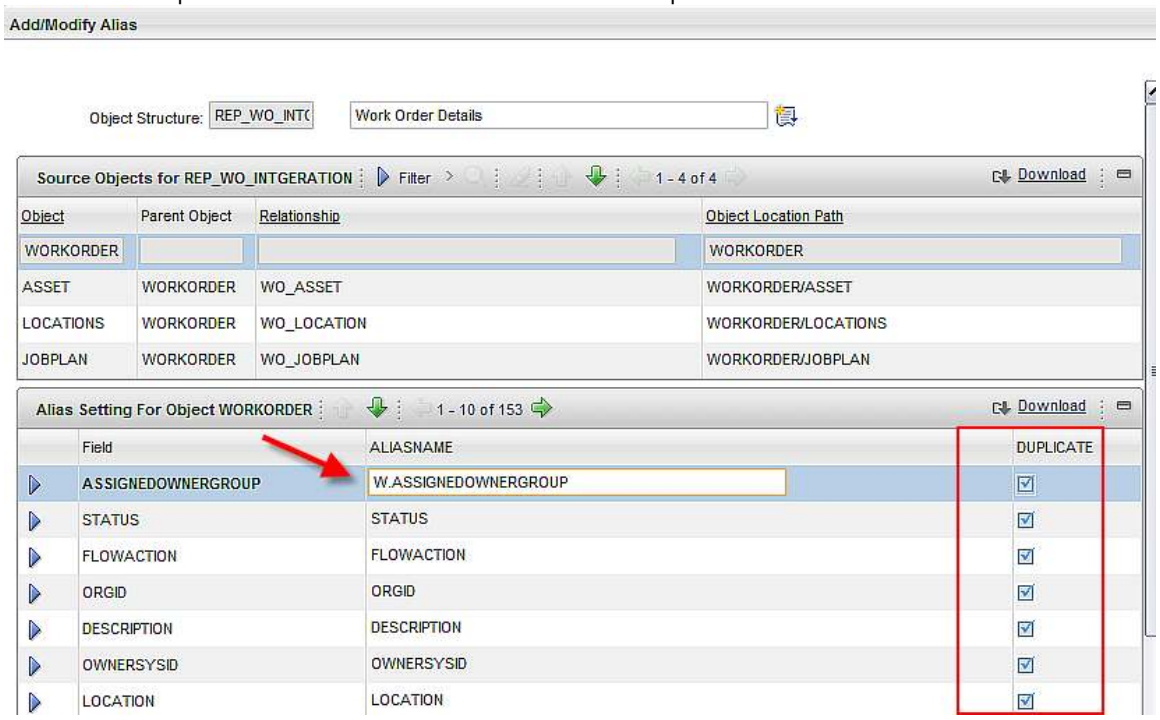
Additionally, select the default format of the exported file, and the maximum number of rows that can be exported. You may want to limit the Maximo Count value for performance reasons.



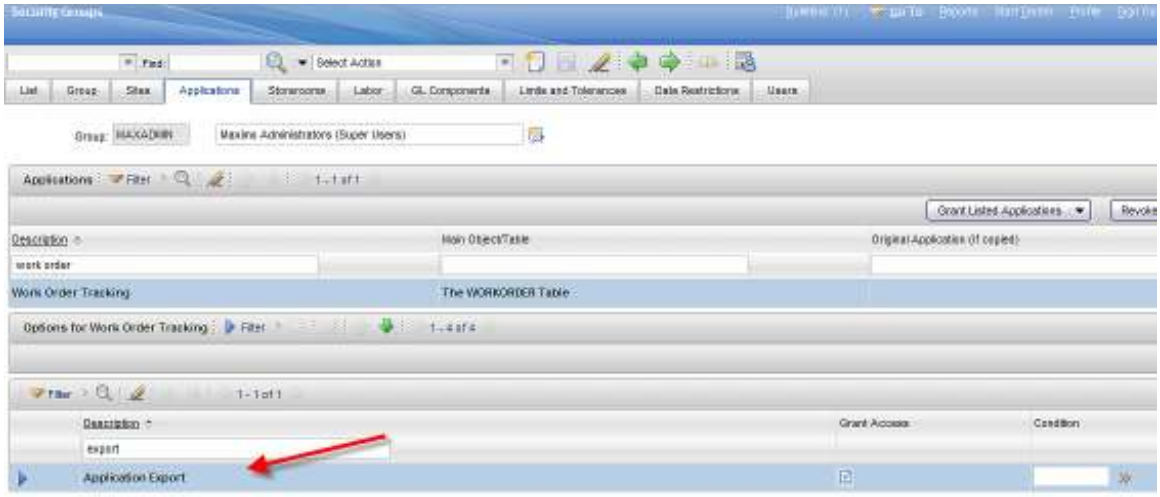
4. From the action menu, select 'Add/Modify Alias'. This action is required to make sure each exported field is uniquely identified.



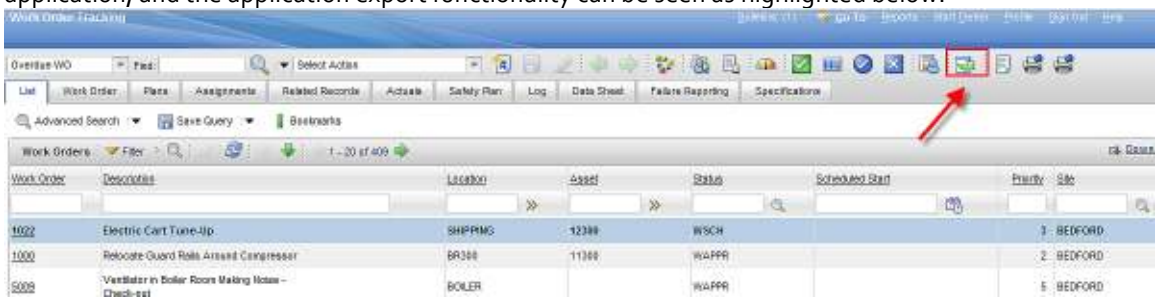
For each field that has the duplicate field enabled, assign it a unique identifier. For example, a W for Workorder was placed before the attribute value in the example below.



5. After insuring unique values are available for each field, next, access the Security Group application. This will enable you to define which security groups have access to the application export functionality for the application.

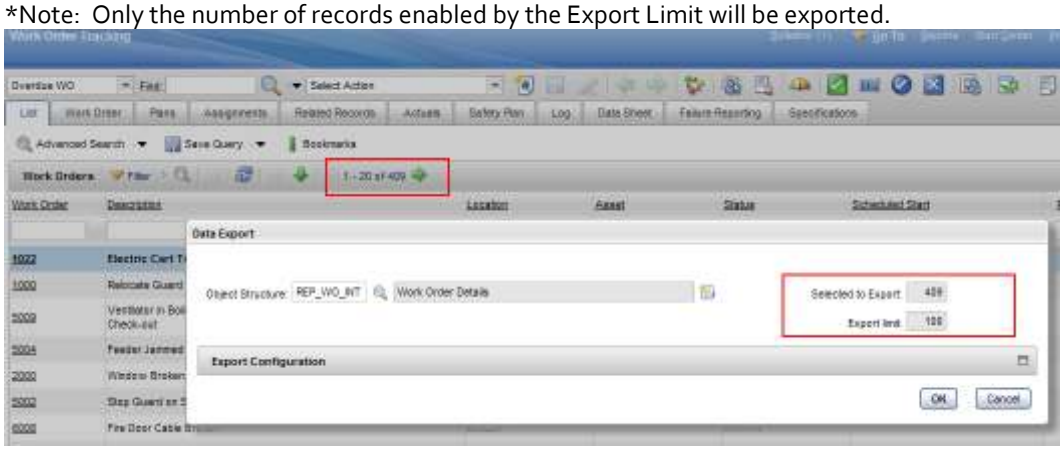


6. To enable the security access to take affect, close the browser and open a new one. Access the selected application, and the application export functionality can be seen as highlighted below.

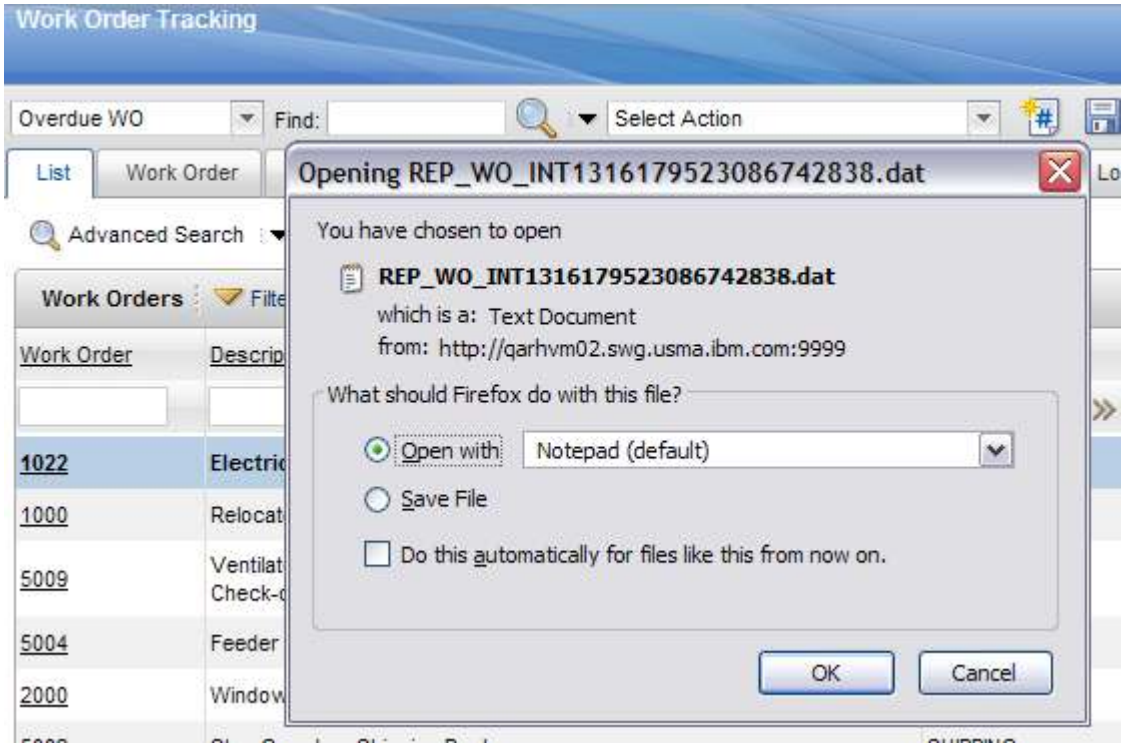


7. Next, select a query from the application, and select the Application Export icon. A dialog displays where you can select the Object Structure to use for the export. Once you have selected it, the maximum export limit for that OS is displayed along with the number of records in your query.

*Note: Only the number of records enabled by the Export Limit will be exported.

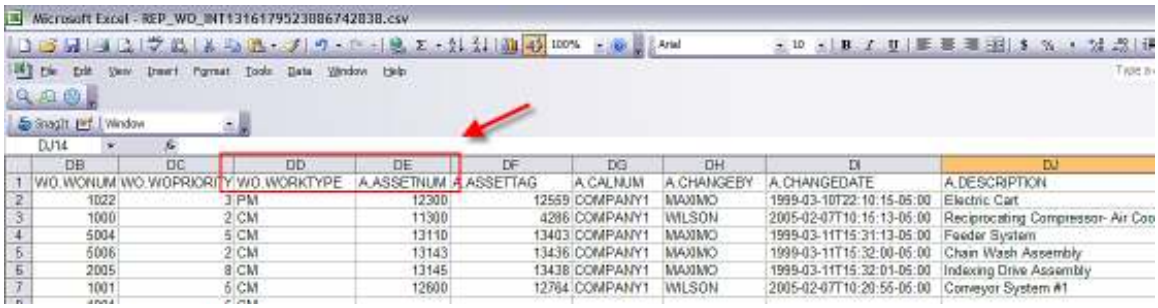


8. A dialog then displays with the exported file name.



The exported file has a .dat file extension. To work with this file in Microsoft Excel, you need to change it from .dat to .csv. (Note: Do not remain to xls or data will be placed in all one cell)

9. Open the exported results. The exported fields include each of the fields defined in the Object Structure, from multiple database objects. This is highlighted as shown below with the Worktype field from the Work Order Object, and the Asset number field from the Asset Object.



	DB	DC	DD	DE	DF	DG	DH	DI	DJ
1	WO.WONUM	WO.WOPRIORITY	WO.WORKTYPE	A.ASSETNUM	A.ASSETTAG	A.CALNUM	A.CHANGEBY	A.CHANGEDATE	A.DESCRIPTION
2	1022	3	PM	12300	12559	COMPANY1	MAXIMO	1999-03-10T22:10:15-05:00	Electric Cart
3	1000	2	CM	11300	4286	COMPANY1	WILSON	2005-02-07T10:16:13-05:00	Reciprocating Compressor- Air Co
4	5004	5	CM	13110	13403	COMPANY1	MAXIMO	1999-03-11T15:31:13-05:00	Feeder System
5	5006	2	CM	13143	13436	COMPANY1	MAXIMO	1999-03-11T15:32:00-05:00	Chain Wash Assembly
6	2005	8	CM	13145	13438	COMPANY1	MAXIMO	1999-03-11T15:32:01-05:00	Indexing Drive Assembly
7	1001	5	CM	12600	12764	COMPANY1	WILSON	2005-02-07T10:20:55-05:00	Conveyor System #1

REFERENCE MATERIALS

The following lists additional references available at the time this guide was prepared. To locate these documents, access this IBM Maximo Wiki Page:

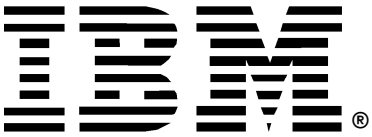
https://www.ibm.com/developerworks/community/wikis/home?lang=en#!/wiki/IBM%20Maximo%20Asset%20Management/page/Reporting%20Documentation

or its shortened url of http://t.co/3qgidEXj6L

This page contains the latest listing of report reference materials, including description, revision levels and hyperlinks to the documentation

Report Detail Guides

Document Name	Description	Version	Reference Number	Revision #	Last Posting Date	BLog* Posting
V75 ReportBooklet	Contains listings, file names, descriptions, details on parameters, formatting (grouping, sorting) and a pdf copy of each of the OOB (Out of the Box) Delivered Reports.	7.5	1497942		4/26/2011	Delivered Reports
V71 ReportBooklet	Contains listings, file names, descriptions, details on parameters, formatting (grouping, sorting) and a pdf copy of each of the OOB (Out of the Box) Delivered Reports.	7.1	1305005	4	12/28/2009	
V7 Delivered Report Comparison	Comparison of the delivered reports from the Maximo 6x, 7.1 and 7.5 releases	7.1, 7.5	1566746		10/3/2011	What's in and Out
V75 Report Feature Guide	Details how the embedded report engine is utilized, including a review of the file structure, installation and database structure. Includes information on Security, Scheduling, Administration, Queuing, and Property Files.	7.5	1498433	3	7/20/2012	
V71 Report Feature Guide	Details how the embedded report engine is utilized, including a review of the file structure, installation and database structure. Includes information on Security, Scheduling, Administration, Queuing, and Property Files.	7.1.x	1305020	10	3/15/2012	
V75 QBR Ad Hoc Reporting	Details how users can Create and Execute Ad Hoc Reports, and the Administrative setup work involved, including how to create Report Object Structures.	7.5	1498285	5	1/4/2013	
V71+ QBR Ad Hoc Reporting	etails how users can Create and Execute Ad Hoc Reports, and the Administrative setup work involved, including how to create Report Object Structures.	7.1.1.6+	1417417	7	1/14/2013	



IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PAPER "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes may be made periodically to the information herein; these changes may be incorporated in subsequent versions of the paper. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this paper at any time without notice.

Any references in this document to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
4205 South Miami Boulevard
Research Triangle Park, NC 27709 U.S.A.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml.

==> Include the following company attribution statements *only* if you mention any of them in your paper:

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Linear Tape-Open, LTO, the LTO Logo, Ultrium, and the Ultrium logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

Other company, product, or service names may be trademarks or service marks of others.