



# Web Work Queries

## Table of Contents

- 17.1 INTRODUCTION TO QUERIES..... 2**
- 17.1.1 QUERY BY EXAMPLE..... 3
- 17.1.2 COMPARISON OPERATORS ..... 5
- 17.1.3 GREATER/LESS THAN vs. GREATER/LESS THAN AND EQUALS ..... 6
- 17.1.4 BETWEEN OPERATOR..... 6
- 17.1.5 NULL CONDITION ..... 6
- 17.1.6 WILDCARD ..... 7
- 17.1.7 SAVING AND RE-RUNNING QUERIES (WORKORDER MODULE ONLY !!) ..... 9
- 17.1.8 RUNNING A SAVED QUERY ..... 10
- 17.1.9 EDITING A SAVED QUERY ..... 10
- 17.1.10 COPYING/PERSONALIZING A SAVED QUERY ..... 11

## 17.1 Introduction to Queries

A “query”, in normal everyday life, is a question. In “computer talk” a query is the equivalent of asking the computer a question. For example, your question might be “show me all of the work orders that have been done in the past year on HVAC unit One, in Building One”. Posed correctly, in computer language, the computer will return to you a listing of all of the matching work orders.

Web Work uses “Structured Query Language” or SQL as the database query engine. This should not be confused with Microsoft SQL server – SQL is a language, Microsoft SQL Server is a database application – a program that organizes the storage of data in a computer, and retrieves that data based on receiving QUERIES from applications, using the SQL “language”. Structured Query Language works across many companies database engines, including Microsoft, Gupta, Oracle, mySQL, etc., and is based on a common language syntax that allows programs to be “portable” and utilize different back end database solutions.

In our example above, the common English request of “show me all work orders that have been done in the past year on the HVAC unit in Building One”, would be translated by the Web Work application into a SQL syntax query, something like this:

```
SELECT * FROM WORKORDERS WHERE LOCATIONID='BLDGONE' AND OPENDATE>=01/01/2003  
AND OPENDATE<=12/31/2003 AND EQUIPMENTID='HVACONE'
```

Obviously, every day users are not expected to be familiar with the SQL “language” syntax, which can be highly technical, and even the smallest error in logic, can lead to huge discrepancies in results.

Below we will explore a much easier method of querying records know as “query by example.”

**17.1.1 Query By Example**

Because very few people are well versed enough in SQL syntax to write their own queries, Tero has developed the Web Work application to use "Query By Example". Simply put, Query By Example means that you will show the program, using the standard forms, what data you wish to have retrieved, and the computer will translate your criteria into a SQL syntax request and process this request against the database.

So, in order to execute even VERY complex queries against the database, the Web Work program requires nothing more from the user than to "show" it what you want, and it will do all of the formatting of the query for you automatically.

The screen shot below shows the query example from above, in the Web Work – Work Order screen. In order to perform this query, you would login to Web Work, and open the Work Order module by clicking on the Work Order module button. On entering the module, the default mode is "QUERY" mode, so the system is ready for you to enter your Query, by example.

We want to find the work orders for HVAC Unit One (HVACONE) in Building One (BLDGONE) which was done in the calendar year 2003, so we will enter:

Location: B1 (Building 1)  
 Equipment: H1 (HVAC Unit 1)  
 Open Date: 1/1/2003,12/31/2003 (the , indicates between – see wildcards below)

The screenshot shows the 'Work Order Module' interface. The form includes the following fields and values:

- W/O Number: [Empty]
- Procedure: [Empty]
- Priority: [Empty]
- Location ID: B1
- Equipment ID: H1
- Meter Reading: [Empty]
- Contact: [Empty]
- Contact Phone: [Empty]
- Account: [Empty]
- Est Cost: [Empty]
- Department: [Empty]
- Note1: [Empty]
- Note4: [Empty]
- Data Division: [Empty]
- Fault Code 1: [Empty]
- Work Type: [Empty]
- PM Number: [Empty]
- Status: [Empty]
- Room: [Empty]
- Down Time: [Empty]
- Eqp Operator: [Empty]
- Requester: [Empty]
- Open Date: 1/1/2003,12/31/2003
- Target Date: [Empty]
- Comp Date: [Empty]
- Duration (time): [Empty]
- Project ID: [Empty]
- Note2: [Empty]
- Note5: [Empty]
- Comp Remark Code: [Empty]
- Craft: [Empty]
- Manager: [Empty]
- Crew: [Empty]
- Employee: [Empty]
- Phase: [Empty]

- 
- The default mode for all modules is the “query” mode. This can be changed under Group Defaults in the Admin module which is described in detail in the Admin Module Chapter.
- To enter “query” mode from any other mode, simply select OPERATIONS from the Module Options Menu and select New (Work Order, PM, Equipment etc.) Query or click the  new query icon at the bottom right corner of the screen.
- While in Query Mode in any module, the user can select the field(s) they want to use in their query.
- Type the information you wish the query to contain in any field(s) and then click on the  lookup icon at the bottom of the screen to display a list of the records in table format, which match the criteria selected for the query.



You can also retrieve all records specific to your criteria by entering the selection criteria in the appropriate field(s) and then pressing the ENTER key on your keyboard. This will open all records, which match your criteria with the first one selected on the main screen. To move from record to record click on the arrow button at the bottom left hand side of the screen.

### 17.1.2 Comparison Operators

In the preceding example, we queried for an exact match on a location and equipment, and used a date range on the open date. There are several 'comparison' operators that you can use on the fields using the query by example method. By default, the operator is "=", or equals, so if we simply put H1 in the equipment field, the resulting query will say "where equipment EQUALS H1".

We could also use other operators:

- > (greater than) – "open date is greater than 1/1/2003"
- >= (greater than or equals) – "open date is greater than or equal 1/1/2003"
- < (less than) – "completion date is less than 12/31/2003"
- <= (less than or equals) – "completion date is less than or equal 12/31/2003"
- <> (not equals) – "building is NOT EQUAL to B1"
- , (between) – "open date is BETWEEN 1/1/2003 and 12/31/2003"

NOTE: These operators do NOT work in the DESCRIPTION field of any module, as that field is a MEMO style field and uses an 'implied-LIKE' wildcard (see below).

#### Examples:

In the work order screen, we query for LOCATION=BUILDING 1.

- Work Order module, Query mode, enter B1 in the Location field

Location ID  

- Click on the  lookup icon. The query results would be displayed:

Operations Resources Tools Help				<i>Work Order Module</i>		
Select all Unselect all Retrieve Records				Order by: WoNum		
	WO Number	Request	Location ID	Equipment ID	Status	Open Date
<input type="checkbox"/>	00003304	Temperature is fluctuating in Cafeteria.	B1		NEWWO	7/4/2005
<input type="checkbox"/>	00003305	Replace Filter on HVAC Unit	B1	H1	INPRG	7/4/2005
<input type="checkbox"/>	00003306	Replace Broken Belt on HVAC Unit	B1	H1	COMP	7/4/2005
<input type="checkbox"/>	00003307	Replace HVAC Filter	B1	H1	NEWWO	7/4/2005

- Notice that all of the work orders have a location id of B1.

Conversely, if we wanted everything EXCEPT BUILDING ONE, we would query for LOCATION <> BUILDING 1.

- Work order module, Query mode, enter <>B1 in the Location field

Location ID  

- Click on the lookup button. The query results would be displayed:

Operations Resources Tools Help				<i>Work Order Module</i>		
Select all Unselect all Retrieve Records				Order by: WoNum		
WO Number	Request	Location ID	Equipment ID	Status	Open Date	
<input type="checkbox"/> 00003308	Sink in womens washroom is leaking into the cupboard below.	B2		NEWREQ	7/4/2005	
<input type="checkbox"/> 00003309	Graffiti on the west wall.	B3		NEWWO	7/4/2005	

- Notice now that the location id in the result set are B2 and B3, and that B1 is NOT included in the results.

### 17.1.3 GREATER/LESS THAN vs. GREATER/LESS THAN AND EQUALS

The difference between > (greater than) and >= (greater than and equals) is that the >= will INCLUDE the date (or number) specified in your query condition, whereas the > will not. For example, if I entered >1/1/2003 in the open date field in work orders, the results would include all work orders with an open date GREATER than (after) 1/1/2003, so from 1/2/2003 until today. If I used >=1/1/2003, the results would include all work orders with an open date of 1/1/2003 and greater (including any opened on 1/1/2003).

This is a common source of confusion for people doing date queries!

### 17.1.4 BETWEEN OPERATOR

The between operator can be used in webwork ONLY on date fields, and the query result will INCLUDE the dates on each end of the between condition – for example, entering 1/1/2003,12/31/2003 in the open date field on work orders, will return work orders where the open date is greater than or equals 1/1/2003 and less than or equals 12/31/2003, so work orders opened on 1/1/2003 and 12/31/2003 (and everything in between) would be included in the results.

### 17.1.5 NULL CONDITION

The word NULL is a 'reserved word' in SQL syntax, and it is used to describe a field with NO data entered into it.

If, for example, you wish to retrieve all work orders that do not have an employee assigned, you could simply enter the word null in the employee field and click lookup – the results will be all work orders without an employee code in the employee field.

You can also use the <>null condition, which would return results where the field is NOT empty – this being the opposite of the example above, would show only work orders that HAD an employee code in the field.

### 17.1.6 Wildcard

Wildcards are very useful in running queries where you may not want, or need, an exact match, but you may be looking for 'partial' matches. This is particularly useful in running queries on 'free form' text fields, such as the work request description, as you could very easily have two essentially identical work orders entered by two different people that are described very differently.

For example, if you wanted to find all of the work orders for replacing filters in HVAC units in August 2002, you could enter a query where: Completion Date is between 8/1/2002 and 8/31/2002 AND Work Request Description is "replace filter in HVAC".

The problem with this query is that on one work order, the description might be "replace filter in HVAC", on another it might be "replace HVAC filter", on another maybe "HVAC filter replace" – if you ran the query for an exact match of "replace filter in HVAC" you would only see ONE of these 3 work orders.

The solution, is rather than searching on an 'exact' match in your query, you can search for a 'partial' match, using the sql "LIKE" condition. In our example, rather than looking for work orders where the request description is "replace filter in HVAC", we could look for work orders where the request is "LIKE" "replace filter". The "LIKE" condition is represented by the "wildcard" character % (percent sign). In Webwork, the LIKE condition is AUTOMATICALLY applied when searching on the DESCRIPTION fields in every module – so if you query the work orders by putting the word "REPLACE FILTER" in the request description box, Webwork will translate your query automatically, to be " LIKE %REPLACE FILTER% ".

The screenshot shows the 'Work Order Module' search interface. At the top, there are navigation links: 'Operations', 'Resources', 'Tools', and 'Help'. The main search area has a large text input field for 'w/o Number' containing the word 'REPLACE'. Below this are several smaller input fields: 'Procedure', 'Priority', 'Fault Code 1', 'Work Type', 'PM Number', and 'Status'. Each of these smaller fields has a magnifying glass icon next to it, indicating a search function. The interface is styled with a light blue header and a white background for the form fields.

The example above is the work order screen, in query mode, with the word REPLACE entered in the work request description field. The results of the query (clicking the lookup button) are shown below:

The screenshot shows the 'Work Order Module' results screen. At the top, there are navigation links: 'Operations', 'Resources', 'Tools', and 'Help'. Below the navigation is a table of work orders. The table has columns for 'WO Number', 'Request', 'Location ID', 'Equipment ID', 'Status', and 'Open Date'. There are three rows of data, each with a checkbox in the first column. The 'Request' column contains the words 'Replace Filter on HVAC Unit', 'Replace Broken Belt on HVAC Unit', and 'Replace HVAC Filter'. The 'Status' column contains 'INPRG', 'COMP', and 'NEWWO'. The 'Open Date' column contains '7/4/2005' for all three rows. At the bottom of the table, there is a text label: 'SELECT FROM WORK ORDER WHERE (Request Like '%REPLACE%') (and in our example, we also have limited the selection to only work orders for division 3 – see divisions elsewhere in this manual for details)'. The interface is styled with a light blue header and a white background for the table.

WO Number	Request	Location ID	Equipment ID	Status	Open Date
<input type="checkbox"/> 00003305	Replace Filter on HVAC Unit	B1	H1	INPRG	7/4/2005
<input type="checkbox"/> 00003306	Replace Broken Belt on HVAC Unit	B1	H1	COMP	7/4/2005
<input type="checkbox"/> 00003307	Replace HVAC Filter	B1	H1	NEWWO	7/4/2005

Notice that the query results include three work orders with the word REPLACE in the work order request description, and you can see the resulting SQL query syntax at the bottom of the result set (SELECT FROM WORK ORDER WHERE (Request Like '%REPLACE%') (and in our example, we also have limited the selection to only work orders for division 3 – see divisions elsewhere in this manual for details)

The % sign represents ANY characters and any NUMBER of characters – so the condition “LIKE %REPLACE FILTER%” would return any work orders that have the text “REPLACE FILTER” anywhere in the description. This would include the following, for example:

- REPLACE FILTER ON THE MOTOR
- REPLACE FILTER
- NEED TO REPLACE FILTER
- FORGOT TO REPLACE FILTER ON THE HVAC UNIT
- Etc.

In this example, you will notice however, that the string WITHIN the % signs (REPLACE FILTER) must be an exact match – so using REPLACE FILTER in the work order description box would NOT match with a work order having a description of: “REPLACE HVAC FILTER” because the word HVAC is BETWEEN the words REPLACE and FILTER. In order to then ‘widen’ the query to include text that has a word, words, or characters between REPLACE and FILTER, you would have to enter: REPLACE%FILTER in the description box. It is important to note that the words must still be in the ORDER that you have entered in your example, so you would not see “NEED TO DO A FILTER REPLACE” in your query results under these circumstances.

Wildcard example:

Work Order #	Description
1	REPLACE HVAC FILTER
2	NEED TO REPLACE FILTER
3	FILTER REPLACE ON HVAC
4	HVAC FILTER REPLACE
5	DIRTY HVAC FILTER NEEDS REPLACEMENT
6	NEED TO REPLACE HVAC FILTER

Using the above work order examples, the following queries would return the results as indicated:

Text entered in description field	Like Condition	Results (work orders)
REPLACE HVAC	%REPLACE HVAC%	1, 6
REPLACE FILTER	%REPLACE FILTER%	2
REPLACE%FILTER	%REPLACE%FILTER%	1, 2, 6
HVAC%FILTER	%HVAC%FILTER%	1,4,5,6
REPLACE	%REPLACE%	1, 2, 3, 4, 5, 6

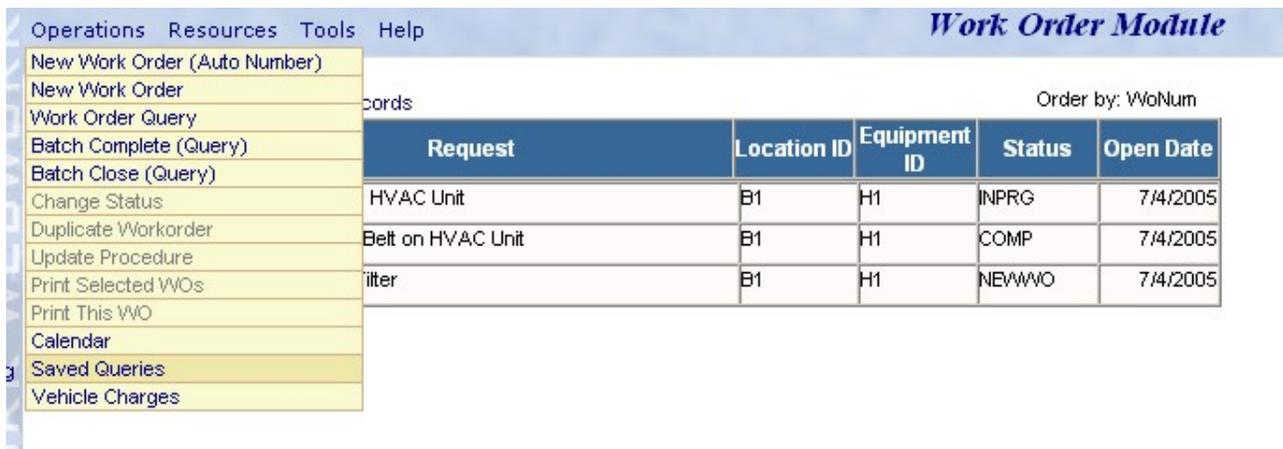
As you can see in the last example, the % wildcard will also find the matching string if it is PART of the word.

For example, if you wanted to find all of the work orders with the word PART in the description, you would enter PART in the work order description field, and the resulting query would be LIKE %PART%. While this would give you work orders such as “NEED PART FOR HVAC”, “REPLACE PART ON FAN”, “BROKEN PART ON CAR”, it would also return things such as “BEARING IS BROKEN APART”, “RENEW PARTNER PROGRAM LICENSE”, “GET PARTRIDGE OUT OF PEAR TREE”, etc ... because the query is not specifying the word PART as such, but is just looking for the characters P, A, R and T, in sequence, somewhere within the description field.

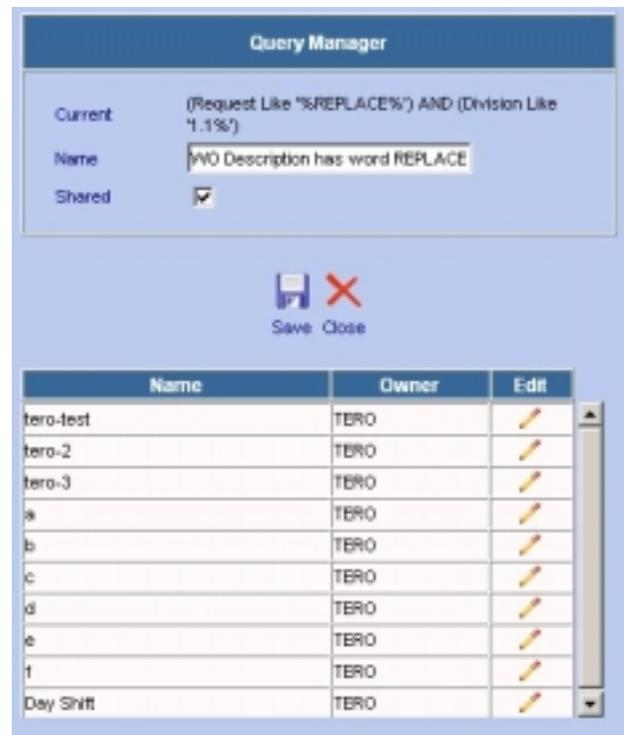
**17.1.7 Saving and Re-running Queries (WORKORDER MODULE ONLY !!)**

Once you have learned how to create a query you may want to store it so you can run it again in the future. This can be particularly useful if you have a very complex query, a query you wish to run over and over again, or if you wish to share your query with a fellow employee. To save a query:

- Open a module and enter your selection criteria as described above
- Click the Operations menu and click Saved Queries



- This will launch the Saved Query Manager as shown to the right.
- You can see the SQL syntax of your query shown (Request Like %REPLACE%) AND (Division Like '1.1%')
- Type the name you want to save the query as in the Name field. (WO Description has word REPLACE)
- To make the query accessible to other Web Work users, click the Shared box so it has a check mark in it. Please note that this will also allow others to edit your query.
- To save the query, click the  button.



### 17.1.8 Running a Saved Query

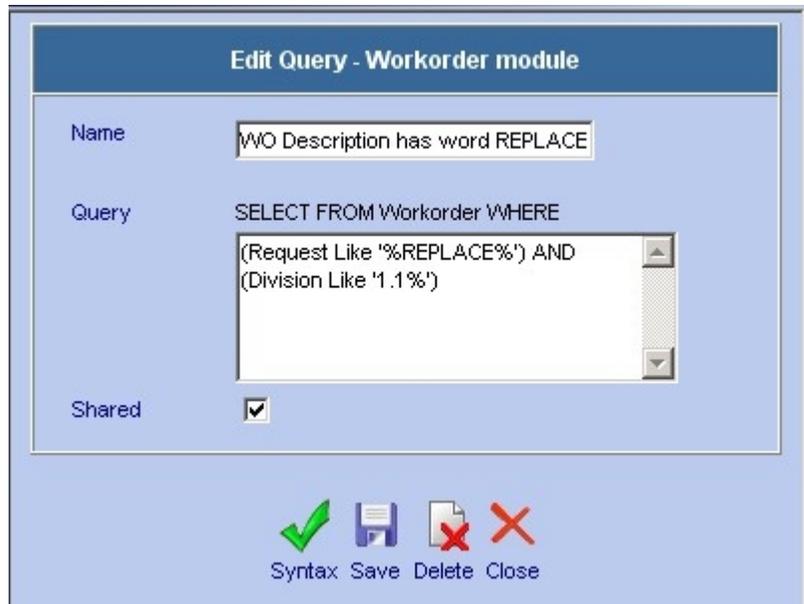
Once you have saved your query, it is written to the database for you (or others if you have shared the query) to run again later. To run a saved query:

- Select [Saved Queries](#) from the operations menu in the work order module
- **CLICK** on the query name in the list shown. (**Note:** This will erase anything you have in progress on the work order screen and return the result set, so you should finish any editing/modifying you are doing before running a saved query.)

### 17.1.9 Editing a Saved Query

You may wish to change the criteria in your saved query, and you can do this by **EDITING** the saved query.

- From the work order module, select [Saved Queries](#) from the Operations menu, and click on the [Edit Icon](#)  in the rightmost column of the query you wish to edit.
- The Edit Query screen, as shown, will popup and you can manually edit the sql syntax of your query. Notice that there is a [CHECK SYNTAX](#) button for you to check the integrity of your sql syntax before saving the query.



As mentioned previously SQL SYNTAX is critical, and even small mistakes can cause dramatically different results!

You could use the edit query function on a saved query to 'personalize' a query already created by someone else.

For example, if BOB saves a query that shows all NEW, EMERGENCY, PLUMBING work orders that are assigned to him, the query would end up being:

```
SELECT FROM WORK ORDER WHERE
(wotype='EM') AND (wostatus='NEW') AND (craft='PLUMB') AND (employee='BOB')
```

Bob then saves that query and names it "Bobs New Plumbing Emerg WOs", and he SHARES it.

Mary could then go into the Work Order module, and **EDIT** that work order to show HER work orders rather than Bob's!

**NOTE:** If Bob shares his query, then anyone, including Mary, can CHANGE his query and save it – so Mary could change the employee='BOB' to employee='MARY' and save it and now 'Bobs New Plumbing Emerg WOs" would actually be Mary's work orders !!!

#### **17.1.10 Copying/Personalizing a Saved Query**

If you are 'personalizing' a query, you should make your own 'copy' of the query:

- Work Order Module, Operations, Saved Queries
- Click on the query you wish to 'personalize' to run it and return the results (i.e.: Bobs New Plumbing Emerg WOs)
- Once you have the results on the screen, go back to Work Order Module, Operations, Saved Queries, and save the query under a new name (i.e.: Marys New Plumbing Emerg WOs)
- Edit the new query to change the condition (change employee='BOB' to employee='MARY')
- Save the query.