



How do I add Exaquantum trend or tag data to a Batch Report?

This depends on the type of report required. If the report is for a single batch then a single data sheet can be used to obtain all the data and then the data here can be reference from anywhere within the report itself.

This can also be the case when the number of batches is always a fixed quantity for example when two batches are being compared as again it is a simple matter to provide two data sheets one for each batch.

The problem arises when the number of batches is unknown or variable in for example a daily or weekly report.

In this case a programmatic approach is required in order to obtain the required information.

There are a number of techniques that can be employed to obtain Exaquantum Data into an excel sheet and these techniques can be extended for use in reports. These are as follows:

1. Use the Exaquantum/Batch Add-in. This is the simplest way of getting batch related data including trend data into a report. This does however require a separate data sheet for each batch to be included in the report and is based on the internal unique ID assigned to the batch when the record is added to the database. To use the Batch Add-in in reports you first populate the data sheet with the required data using any batch and then replace the Batch ID with a cell reference. The Batch Help file has instructions on how to do this (see the topic on Procedure for Reworking as a Report Template). If more than one batch is involved then the Batch Ids will need to come from other queries. This will only work for Batch Trend Data from tags configured in the Equipment Configuration Tool. This is also not very useful when the number of batches for which data is required is not a fixed number.
2. A more flexible mechanism that can be used is the Exaquantum Query Wizard. It is possible to create a query and then change this programmatically to obtain any available data. In particular the time or times used in the query would need to be modified whenever the template is used. This basically means that it will be necessary to code the query explicitly as in the example below.

```
Public Sub GetTagTrendData()
```

```
Dim dtStartSub As Date
```

```
Dim iRow As Integer
```

```
Dim sQuery As String
```

```
On Error GoTo Error
```

```
,
```

```
' Generate the Query as follows
```



```

'
' We require it in the order Timestamp and value
'
' sQuery = "QTrendData N'Timestamp, Value', N'"
'
' Start and End Times from the summary sheet
'
sQuery = sQuery & Format(Worksheets("Summary").Cells(21, _
                                     3).Text, "yyyy-mm-dd hh:mm:ss")
sQuery = sQuery & ", N'" & Format(Worksheets("Summary").Cells(22, 3).Text, _
                                     "yyyy-mm-dd hh:mm:ss")
'
' Every Second and with the bounding values
'
sQuery = sQuery & ", N'00:00:01', N'IncludeBounding', N'"
'
' Tag list from the Trend Tag Sheet
'

iRow = 2
While Worksheets("Trend Tags").Cells(iRow, 4).Text <> ""
    sQuery = sQuery & Worksheets("Trend _
                                Tags").Cells(iRow, 4).Text
    iRow = iRow + 1
    If Worksheets("Trend Tags").Cells(iRow, 4).Text _
                                                <> "" Then
        sQuery = sQuery & ", "
    End If
Wend
sQuery = sQuery & ""

dtStartSub = Now
'
' Note in this example it would be necessary to have made a query using MSQuery
' already in order to get an entry in the QueryTables collection.
' What we are doing is replacing the CommandText property of this Query with the
' one that has been constructed.
'
With Worksheets("Trend Data").QueryTables("Trend Data")
    ,

```



```

' Wait for for a previous refresh to complete.
' Time out after 30 seconds
'
Do While .Refreshing = True
    If Now < DateAdd("s", 30, dtStartSub) Then Exit Do
    DoEvents
Loop
If .Refreshing Then .CancelRefresh
.CommandType = xlCmdSql
.CommandText = sQuery
.Refresh (False)
End With
Exit Sub
Error:
' Put Error Handling Here
End Sub

```

3. You can also use the API directly. Examples of using the API in reports can be found in the Advanced group of templates provided with the product (in particular see the Batch Trend Data 1 Report).

It is also possible to use the Add method of the QueryTable object to create queries on the fly or use an existing QueryTable entry as a template to create a new one.

There is one other method which would seem intuitive to use and that is the Exaquantum/Explorer Excel Add-in. Most people would be familiar with this and therefore it would seem the most obvious thing to use. It returns data to Excel asynchronously, however and there is no way to tell the Report Manager Service when all the data has been acquired. This will result in reports being printed with some data missing (#REGISTER). This is therefore not a good method to use and is not recommended.

Exaquantum /Batch Releases Affected
All