# OA Report Writer

| | |
|---|---|
| Prepared by: | Documentation Team |
| Prepared for: | Learning Resources |
| Module: | Business Intelligence |
| Date: | 2016 |
| Document Ref: | LMDSY0049 |
| Version: | 11.05 |

Construction Industry Solutions Ltd.
11 St. Laurence Way
SL1 2EA

Construction Industry Solutions Ltd.
11 St. Laurence Way
SL1 2EA

**THIS USER GUIDE WAS CREATED USING MADCAP FLARE.**

# CONTENTS

# 1 OA Report Writer

The Report Writer is a tool which can be used by a broad cross-section of COINS users to produce powerful column based reports across the COINS system.

## 1.1 Key features

- Can be used 'out of the box', with over 1200 default queries, through the COINS Business Logic, providing access to every table in the system.
- Fully controllable using COINS function access so Users will only ever have access to the appropriate data.
- Suited to both managerial and operational end users of COINS.
- Users have ownership of their own reports, which can be published to COINS User Groups using the Report Runner functionality.
- Allows for quick and easy ad-hoc reporting across the COINS system.
- Supported by a documented view of the database (Database Enquiry).
- Can be a partner for, or a first step towards using the BI Designer toolset.

## 1.2 Key Functionality:

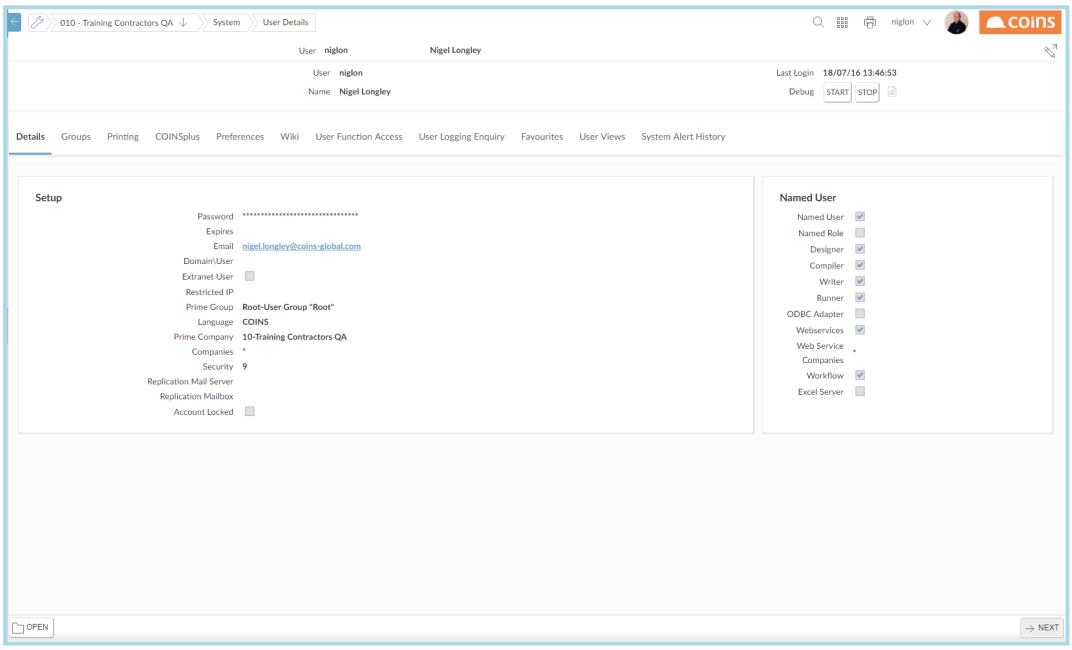
Whilst being an entry level reporting tool aimed at any COINS user, the Report Writer also provides extensive functionality to allow the production of powerful reports :

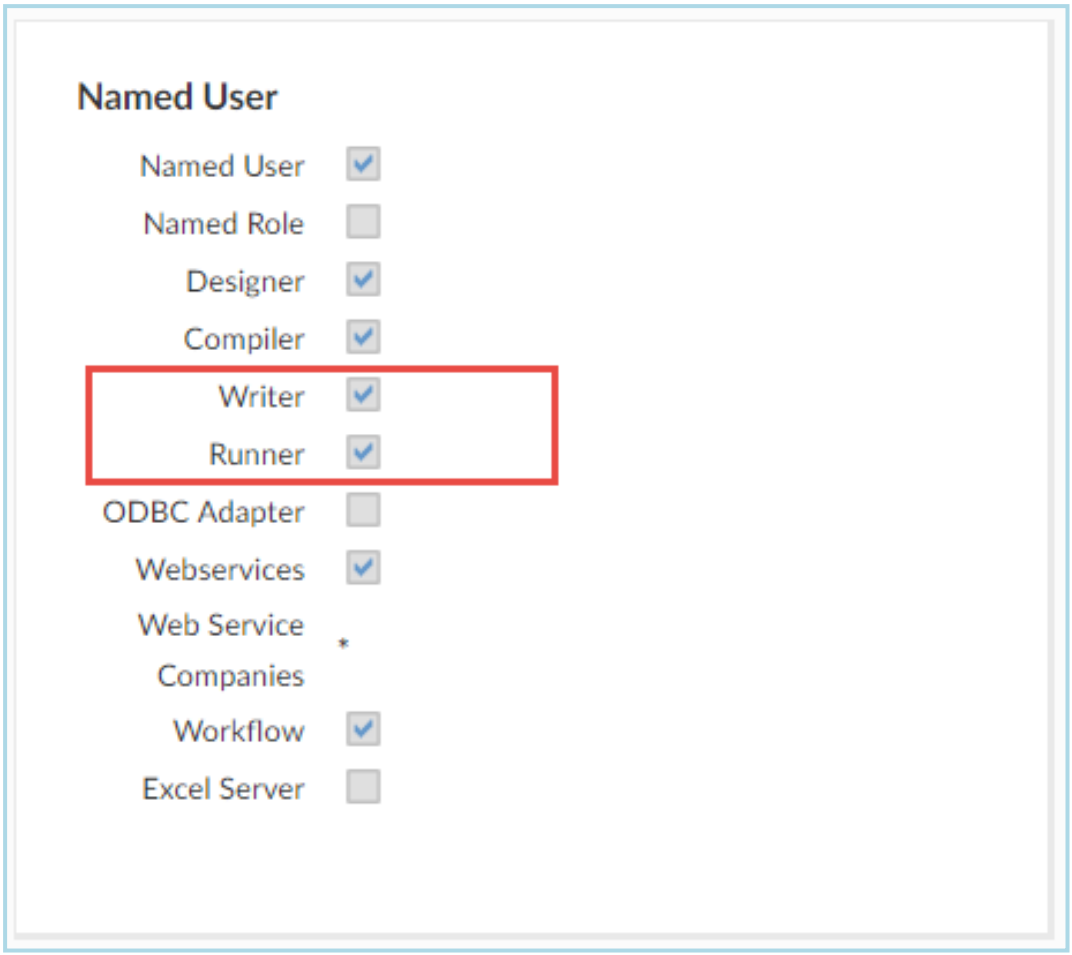
- Totalling methods.
- Dynamic sorting and summarising of the data.
- Using calculations to manipulate the data
- Displaying the information graphically using charts.

## 1.3 Granting Access to Report Writer and Report Runner

Report Writer licences are based on a Named Used licence. To assign a licence to a User, (once the appropriate module has been licensed via the branding screen) access the User record and click the appropriate licence.

Report Runner licences are based on a Named Used licence and can be assigned in the same manner as the Report Writer licence




Named User
Named User
Named Role
Designer
Compiler
Writer
Runner
ODBC Adapter
Webservices
Web Service Companies
Workflow
Excel Server

### 1.3.1 Function Security

As well as having Report Writer licence ticked against their user profile, function security to the Report Writer will also need to be assigned to a user profile/groups:

The top level menu (for OA Reporting and BI) is %WOAREPBI

Below that they will need the OA Utilities/Database Enquiry:

%WOAREPU

%WSY5000WXXX

%WSY5001WXXX

Report writer consists of:

%WOAREP1

%WSY2400BRTN

%WSY2410BSTE

%WSY2400BRFD

%WSY2400BRFDA

%WSY2400BRFDB

%WSY2400BRFDD

%WSY2400BRFDU

%WSY2400BRFDX

%WSY2400BRTNA

%WSY2400BRTNB

%WSY2400BRTND

%WSY2400BRTNU

%WSY2400BRTNX

%WSY2400BXXX

%WSY2400BXXXX

%WSY2400FRTN

%WSY2400SRTN

%WSY2400SRTNO

%WSY2400SRTNT1

%WSY2400SRTNT2

%WSY2400SRTNT3

%WSY2400SRTNT4

%WSY2400SRTNT5

%WSY2400SRTNT6

%WSY2400SRTNT7

%WSY2400SRTNU

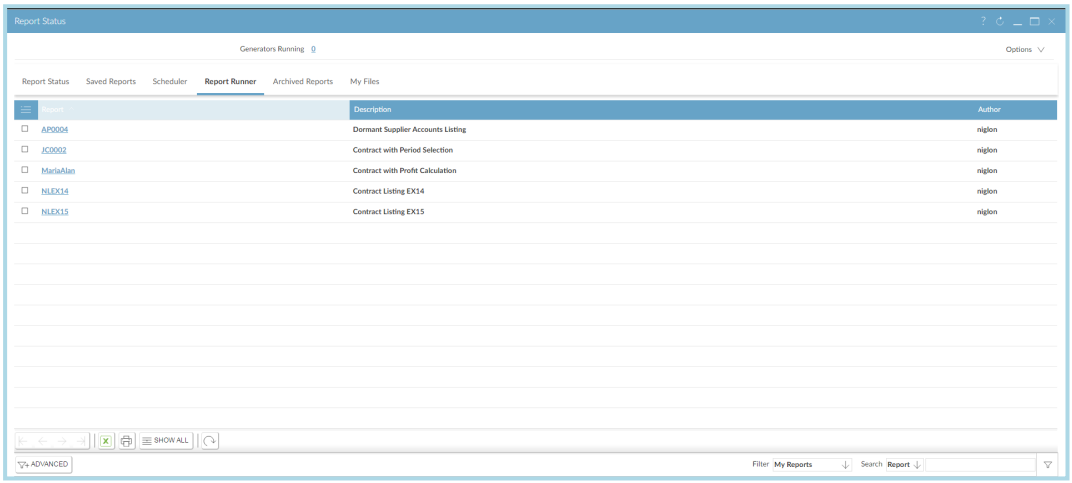
%WSY2401FRTN

%WSY2403FRTN

%WSY2405FRTN

#### 1.3.1.1 Report Runner

For clients using Report Runner, in addition to the above, it will be necessary to grant the appropriate access to the Report Runner tab functions in the Report Status Workbench (%WSY2430BRTN)



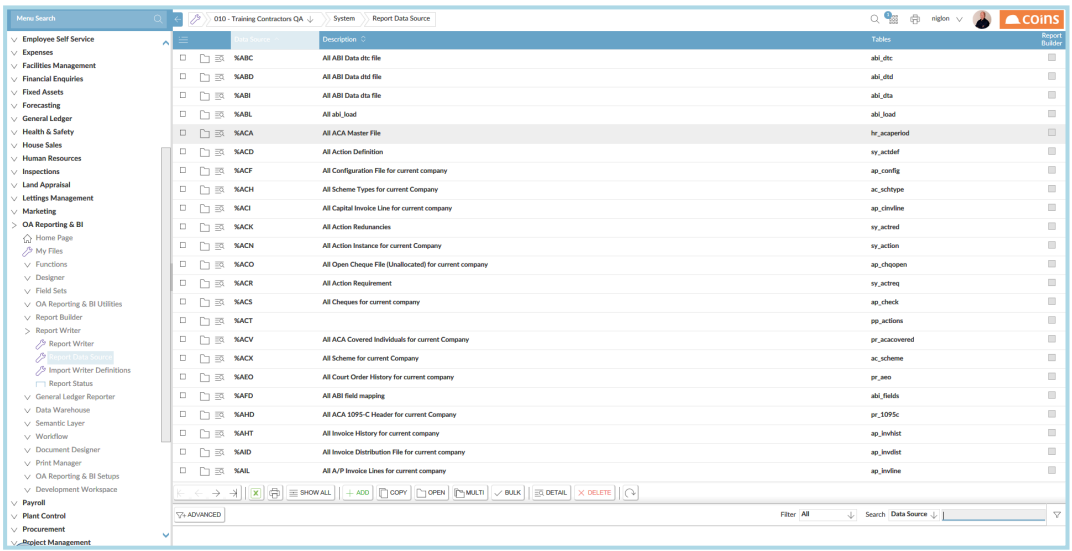
# 2 Report Data Sources

Report Data Source allows you to set up data sources for reports created using Report Writer or Report Builder. A large number of pre-defined queries exist to allow Report Writer to access the majority of the tables in COINS without the user having to know or use the database query language.

In addition to the standard set of Report Writer queries provided by COINS it is possible to for System Administrators, with a reasonable knowledge of the query language, to build user specific queries to meet business requirements not provided by the standard set.

The list of available Report Data Sources can be accessed by navigating to:

**OA Reporting and BI > Report Writer > Report Data Source**




| Field | Description |
|---|---|
| Data Source | The code of the data source. |
| Description | The description of the data source. |
| Tables | Tables in the query |
| Report Builder | Whether this data source can be used in Report Builder |

## 2.1 Report Data Source Definitions

Information from RDBMS systems is retrieved using query languages. Progress RDBMS (The database used by coins) uses Progress 4GL query language. In response to a query, the database returns a result set, which is just a list of rows containing the answers. The simplest query is just to return all the rows from a table, but more often, the rows are filtered in some way to return just the answer wanted.

Example Query on the coins database to retrieve all contracts (jc_job) that belong to company 1:

```
FOR EACH jc_job WHERE jc_job .kco = 1
```

Often, data from multiple tables gets combined into one, by doing a join. Conceptually, this is done by taking all possible combinations of rows (the "cross-product"), and then filtering out everything except the answer.

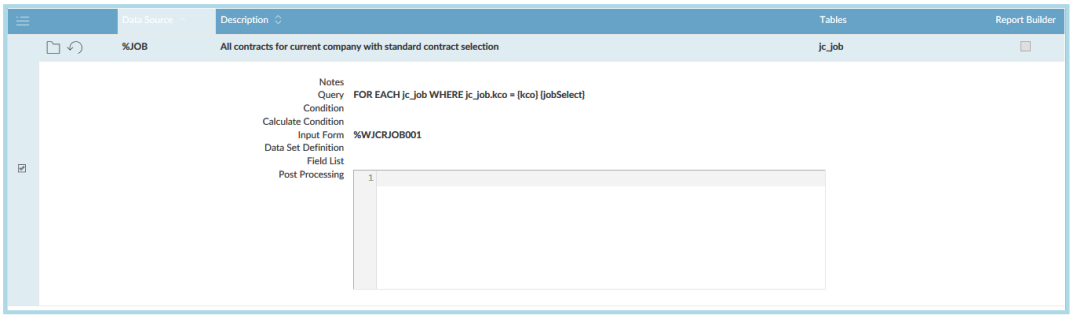
Example Query on the coins database to retrieve all costheads (jc_costcode) that belong to contracts (jc_job) that belong to company 1:

```
FOR EACH jc_job WHERE jc_job .kco = 1,
EACH jc_costcode OF jc_job
```

The flexibility of relational databases allows programmers to write queries that were not anticipated by the database designers. As a result, relational databases can be used by multiple applications in ways the original designers did not foresee, which is especially important for databases that might be used for decades. This has made the idea and implementation of relational databases very popular with businesses.

COINS OA uses a simplified version of the 4GL query language in combination with the RSP's to extract the data for reports and enquiries (for further information on RSP's – Record Service Procedures – see the relevant COINS BI documentation).

COINS OA uses the query to decide which records are accessed from the coins database from the database. The Page Design (Report Design) will determine which fields from these records are displayed (either on screen or in a report).




| Field | Description |
|---|---|
| Notes | Notes or technical information about the query. |
| Query | 4GL code that will select the appropriate records in a query. |
| Condition | Query condition method for the query. The method specified is called and if the returned logical value is false then the record is excluded from the query.<br><br>Conditions provide a means of giving additional functionality to the Query, but these must be pre-defined by COINS in order to be used and are stored in certain Record Service Procedures. If you have a requirement to access information in the data tables that does not appear to be straight-forward to put into standard Progress 4GL, please contact COINS for advice on possible Conditions that may be available. |
| Calculate Condition | A calculate condition is used to limit the range of data returned to a report. The calculation is performed on each line of the report to determine whether the record should be included or not. The calculation should return either a 1 for true (i.e. display the line) or a 0 for false (do not display the line). The calculation should be a valid calculation on fields available to the report. |
| Input Form | The input form used to prompt for run-time selection criteria for this report query. This is optional.<br><br>It is possible to associate a Report Writer query with selections criteria, either standard or user defined. User Defined forms are created using the Page Designer Tool.The query must include the appropriate {tttSelect} token. |

| Field | Description |
|---|---|
| Data Set Definition | The data set definition(s) to be used for this query.<br><br>A comma separated list of data set definition codes. |
| Field List | The list of fields that will be visible to the user of this query from the available fields in the query.<br><br>You can enter a CAN-DO list of fields to be shown. |
| Post Processing | |

# 3 OA and BI Utilities

To assist developers in creating and testing OA Queries and calculations, a number of utilities are available with the OA & BI Reporting Module.

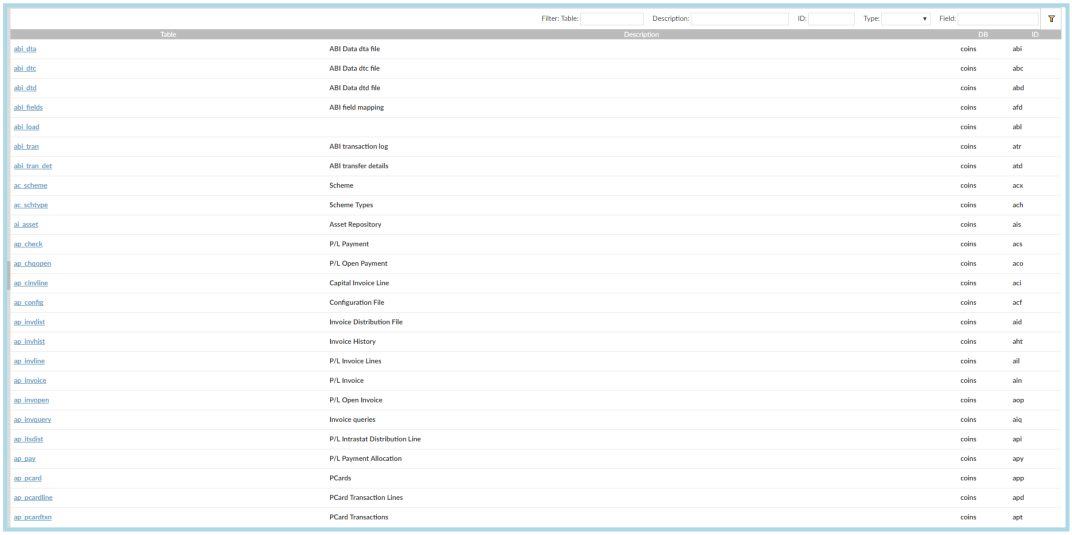
The commonly used utilities are:

- Database Enquiry
- Query Editor
- Calculation Editor
- Object Enquiry

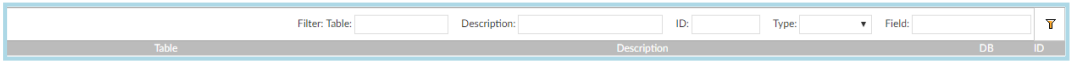
## 3.1 Database Enquiry

To assist users in Open Architecture to understand and exploit the coins database schema when creating enquiries and reports a powerful tool has been developed which provides information on all coins tables, fields and formats.

The Database Enquiry provides detailed information on the structure of every table in the coins database.




Filters at the top of the page allow you to search for specific tables, fields and table types.




It also provides field information, descriptions, formats as well as documentation supplied by the coins Development Team to support users in creating their queries. This includes all calculated fields available via the RSP's (Record Service Procedures).

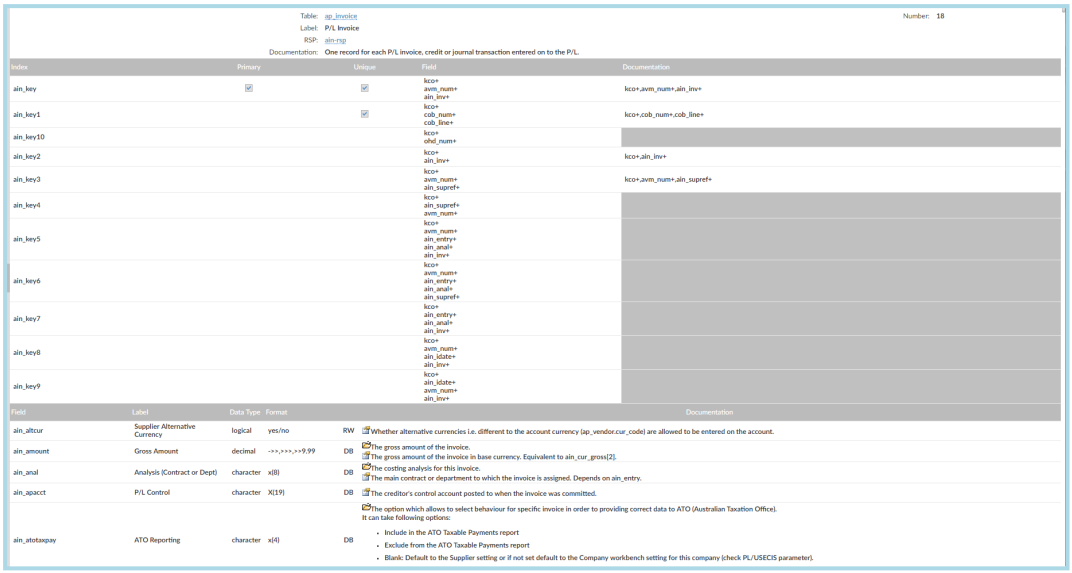


Table: ap_invoice Number: 18
Label: P/L Invoice
RSP: ain-rsp
Documentation: One record for each P/L invoice, credit or journal transaction entered on to the P/L.

| Index | Primary | Unique | Field | Documentation |
|---|---|---|---|---|
| ain_key | ☑ | ☑ | kco+ avm_num+ ain_inv+ | kco+,avm_num+,ain_inv+ |
| ain_key1 | | ☑ | kco+ cob_num+ cob_line+ | kco+,cob_num+,cob_line+ |
| ain_key10 | | | kco+ ohd_num+ | |
| ain_key2 | | | kco+ ain_inv+ | kco+,ain_inv+ |
| ain_key3 | | | kco+ avm_num+ ain_supref+ | kco+,avm_num+,ain_supref+ |
| ain_key4 | | | kco+ ain_supref+ avm_num+ | |
| ain_key5 | | | kco+ avm_num+ ain_entry+ ain_anal+ ain_inv+ | |
| ain_key6 | | | kco+ avm_num+ ain_entry+ ain_anal+ ain_supref+ | |
| ain_key7 | | | kco+ ain_entry+ ain_anal+ ain_inv+ | |
| ain_key8 | | | kco+ avm_num+ ain_idate+ ain_inv+ | |
| ain_key9 | | | kco+ ain_idate+ avm_num+ ain_inv+ | |

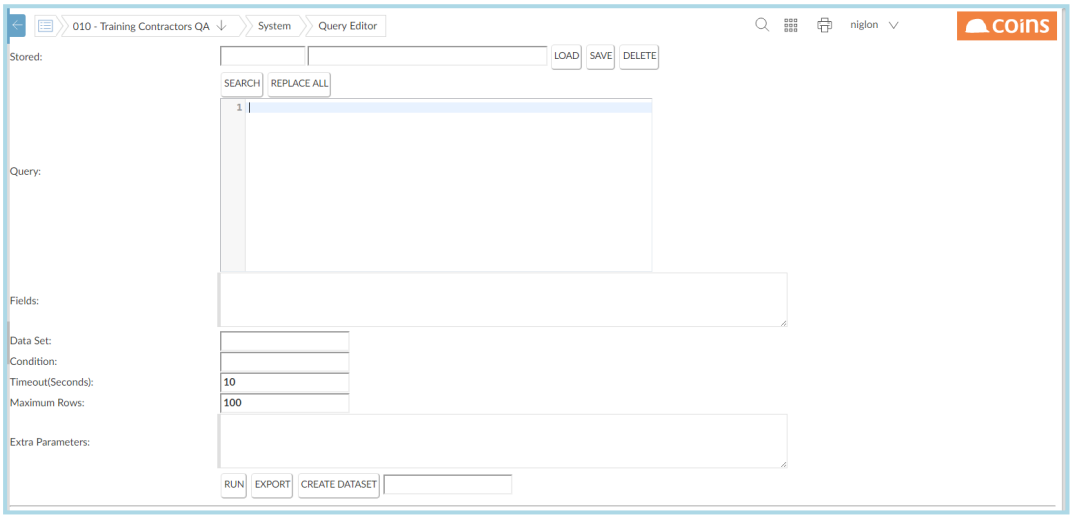
| Field | Label | Data Type | Format | | Documentation |
|---|---|---|---|---|---|
| ain_altcur | Supplier Alternative Currency | logical | yes/no | RW | Whether alternative currencies i.e. different to the account currency (ap_vendor.cur_code) are allowed to be entered on the account. |
| ain_amount | Gross Amount | decimal | ->>,>>>,>>9.99 | DB | The gross amount of the invoice.<br>The gross amount of the invoice in base currency. Equivalent to ain_cur_gross[2]. |
| ain_anal | Analysis (Contract or Dept) | character | x(8) | DB | The costing analysis for this invoice.<br>The main contract or department to which the invoice is assigned. Depends on ain_entry. |
| ain_apacct | P/L Control | character | X(19) | DB | The creditor's control account posted to when the invoice was committed. |
| ain_atotaxpay | ATO Reporting | character | x(4) | DB | The option which allows to select behaviour for specific invoice in order to providing correct data to ATO (Australian Taxation Office).<br>It can take following options:<br>• Include in the ATO Taxable Payments report<br>• Exclude from the ATO Taxable Payments report<br>• Blank: Default to the Supplier setting or if not set default to the Company workbench setting for this company (check PL/USECIS parameter). |

In addition the Database Enquiry will provide the links available to associated tables and also provide the syntax required to build a query to create these links within Page and Report Designer.

| From | To | Join To | Documentation | Code |
|---|---|---|---|---|
| 1 | * | ap_cinvline | ap_cinvline OF ap_invoice | ap_cinvline.kco=ap_invoice.kco<br>AND ap_cinvline.avm_num=ap_invoice.avm_num<br>AND ap_cinvline.ain_inv=ap_invoice.ain_inv |
| 1 | * | ap_invdist | ap_invdist OF ap_invoice | ap_invdist.kco=ap_invoice.kco<br>AND ap_invdist.avm_num=ap_invoice.avm_num<br>AND ap_invdist.ain_inv=ap_invoice.ain_inv |
| 1 | * | ap_invline | ap_invline OF ap_invoice | ap_invline.kco=ap_invoice.kco<br>AND ap_invline.avm_num=ap_invoice.avm_num<br>AND ap_invline.ain_inv=ap_invoice.ain_inv |
| 1 | 1 | ap_invopen | ap_invopen OF ap_invoice | ap_invopen.kco=ap_invoice.kco<br>AND ap_invopen.avm_num=ap_invoice.avm_num<br>AND ap_invopen.ain_inv=ap_invoice.ain_inv |
| 1 | 1 | ap_invquery | ap_invquery OF ap_invoice | ap_invquery.kco=ap_invoice.kco<br>AND ap_invquery.avm_num=ap_invoice.avm_num<br>AND ap_invquery.ain_inv=ap_invoice.ain_inv |
| 1 | * | ap_itsdist | ap_itsdist OF ap_invoice | ap_itsdist.kco=ap_invoice.kco<br>AND ap_itsdist.avm_num=ap_invoice.avm_num<br>AND ap_itsdist.ain_inv=ap_invoice.ain_inv |
| 1 | * | ap_vatdist | ap_vatdist OF ap_invoice | ap_vatdist.kco=ap_invoice.kco<br>AND ap_vatdist.avm_num=ap_invoice.avm_num<br>AND ap_vatdist.ain_inv=ap_invoice.ain_inv |
| * | 1 | ap_vendor | ap_vendor OF ap_invoice | ap_vendor.kco=ap_invoice.kco<br>AND ap_vendor.avm_num=ap_invoice.avm_num |
| * | 1 | ap_vendsum | ap_vendsum OF ap_invoice | ap_vendsum.kco=ap_invoice.kco<br>AND ap_vendsum.avm_num=ap_invoice.avm_num |
| * | 1 | ar_invoice | ar_invoice OF ap_invoice | ar_invoice.kco=ap_invoice.kco<br>AND ar_invoice.cob_num=ap_invoice.cob_num<br>AND ar_invoice.cob_line=ap_invoice.cob_line |
| * | 1 | cb_tdet | cb_tdet OF ap_invoice | cb_tdet.kco=ap_invoice.kco<br>AND cb_tdet.cob_num=ap_invoice.cob_num<br>AND cb_tdet.cob_line=ap_invoice.cob_line |
| * | 1 | cb_topen | cb_topen OF ap_invoice | cb_topen.kco=ap_invoice.kco<br>AND cb_topen.cob_num=ap_invoice.cob_num<br>AND cb_topen.cob_line=ap_invoice.cob_line |
| * | 1 | co_currency | co_currency OF ap_invoice | co_currency.kco=ap_invoice.kco<br>AND co_currency.cur_code=ap_invoice.cur_code |
| * | 1 | co_vat | co_vat OF ap_invoice | co_vat.kco=ap_invoice.kco<br>AND co_vat.vat_code=ap_invoice.vat_code |
| * | 1 | cs_certificate | cs_certificate OF ap_invoice | cs_certificate.kco=ap_invoice.kco<br>AND cs_certificate.cob_num=ap_invoice.cob_num<br>AND cs_certificate.cob_line=ap_invoice.cob_line |
| * | 1 | fa_invoice | fa_invoice OF ap_invoice | fa_invoice.kco=ap_invoice.kco<br>AND fa_invoice.ain_inv=ap_invoice.ain_inv<br>AND fa_invoice.avm_num=ap_invoice.avm_num |
| * | 1 | hs_devsum | hs_devsum OF ap_invoice | hs_devsum.kco=ap_invoice.kco<br>AND hs_devsum.job_num=ap_invoice.job_num<br>AND hs_devsum.jph_phase=ap_invoice.jph_phase |
| * | 1 | jc_job | jc_job OF ap_invoice | jc_job.kco=ap_invoice.kco<br>AND jc_job.job_num=ap_invoice.job_num |
| * | 1 | jc_jobsum | jc_jobsum OF ap_invoice | jc_jobsum.kco=ap_invoice.kco<br>AND jc_jobsum.job_num=ap_invoice.job_num |
| * | 1 | jc_phase | jc_phase OF ap_invoice | jc_phase.kco=ap_invoice.kco<br>AND jc_phase.job_num=ap_invoice.job_num<br>AND jc_phase.jph_phase=ap_invoice.jph_phase |
| * | 1 | jc_phssum | jc_phssum OF ap_invoice | jc_phssum.kco=ap_invoice.kco<br>AND jc_phssum.job_num=ap_invoice.job_num<br>AND jc_phssum.ioh_phase=ap_invoice.ioh_phase |

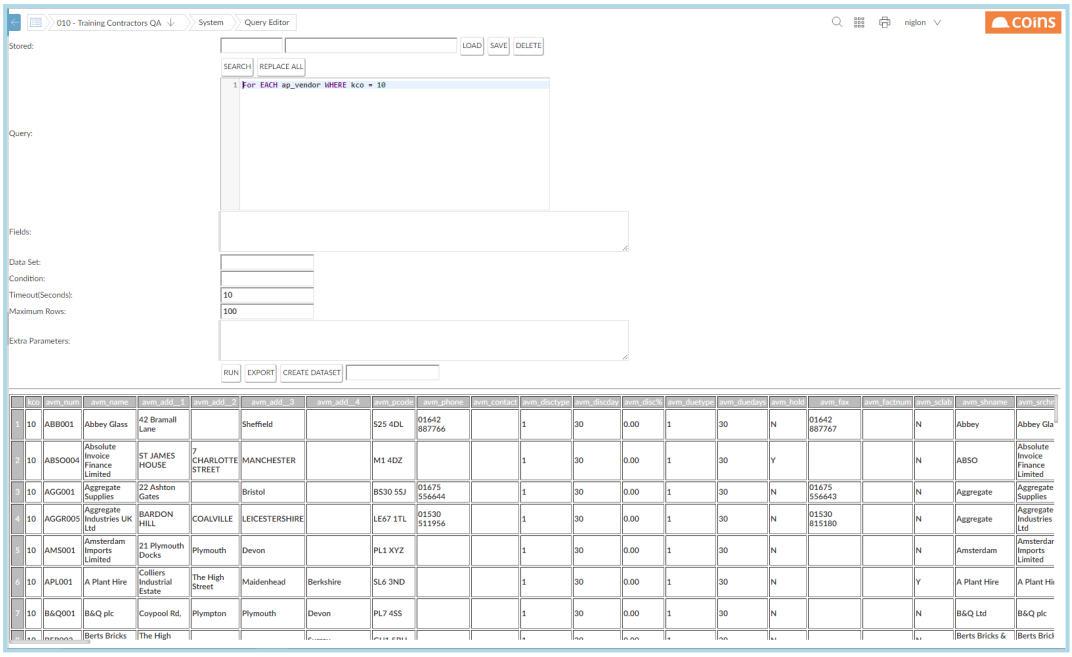
## 3.2 Query Editor

The Query Editor allows you to try out 4GL queries against the COINS database and sample the data returned. This function can be useful to test queries before being used in reports or enquiries.




To use the query editor simply enter the query, and (optionally) any fields required - space separated - and click RUN. The system will return an error if any part of the query is incorrect – and a sample of data if the query compiles OK.

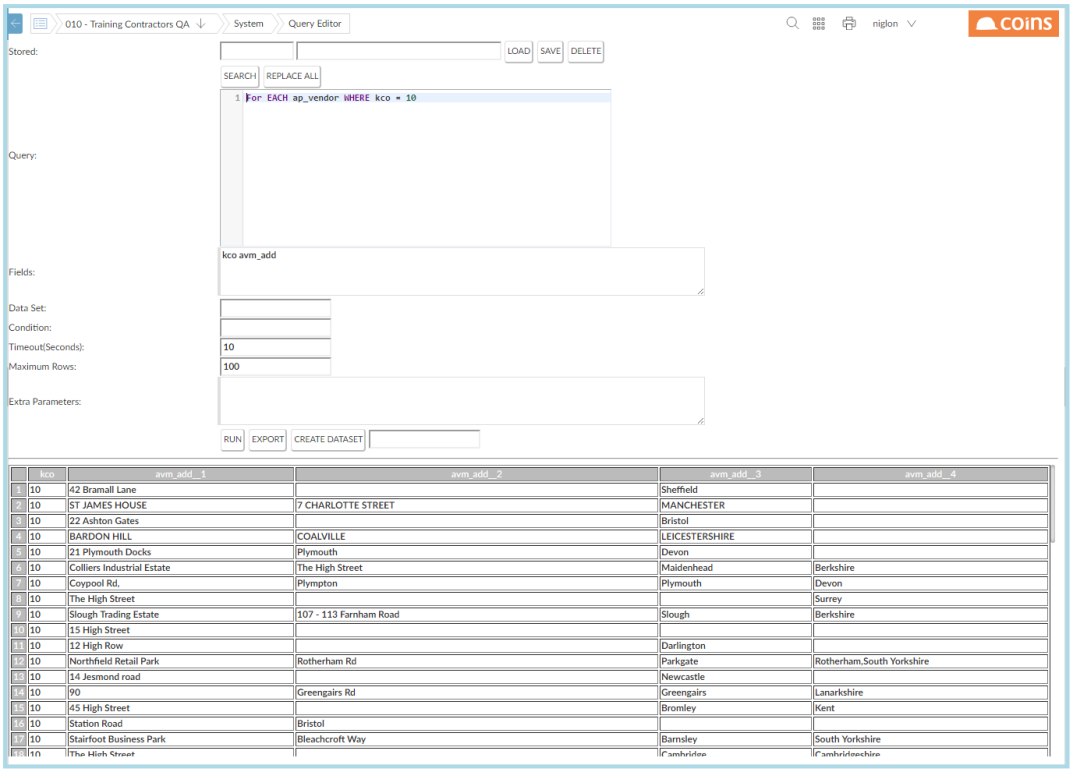
> **Note:** **If the fields section was left blank, all fields will be displayed. If field names were specified, only those fields will be shown.**




| Field | Descrription |
|---|---|
| Data Set | A Data Set definition can be entered here to display the information created in the data set (No query or fields are required for this). |
| Condition Field | A function that determines whether a record should be included or not. The function returns a logical value: yes to include the record, no to exclude it. |
| Maximum Rows | Allows the query to fun faster by only displaying a maximum number of rows per query. 10.23 onwards, this defaults to 10 |

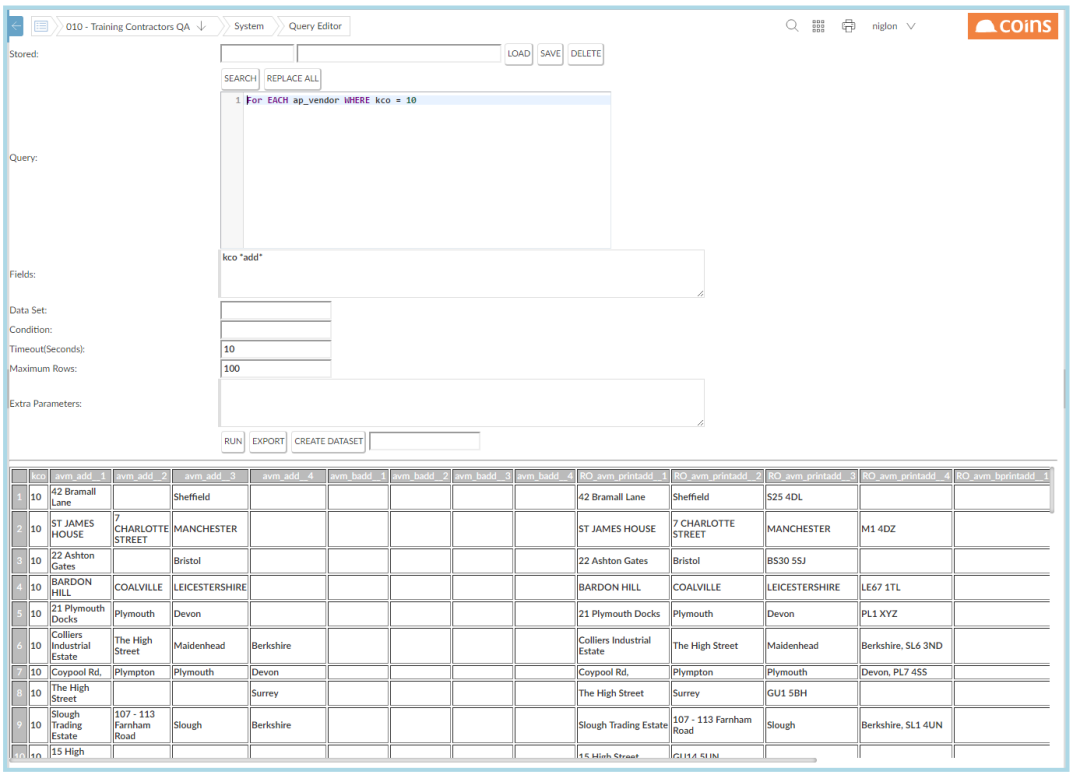
| Field | Descrription |
|---|---|
| Extra Parameters (10.23 onwards) | Where a dataset has been specified, this field allows entry of of parameters (URL) that are needed by the dataset query.<br><br>e.g.<br><br>The parameterised fields are so that you don't have to 'hard code' queries in the dataset to get it to run in the query editor – particularly if there are date replacements etc with fields like {RS_glp_fdate__2}. Or another useful reason for using these parameters is so that you can test results in a efficient way for instance: if you have a query on the dataset which reads:<br><br>FOR EACH jc_job WHERE jc_job.kco = {kco} {jobSelect}<br><br>You could call the dataset from within the query editor and in the parameters say jobSelect=and jc_job.job_num = 'XXXX' (where XXXX is a vailid contract number).<br><br>That way the dataset would run but for only contract XXXX - This is good to save time in checking the validation of fields in the dataset as you don't have to wait till the whole dataset evaluates prior to getting a response back.<br><br>If you have more than one {} replacement in your dataset then you would separate the parameters with a & symbol Eg: Dataset query might read :<br><br>FOR EACH jc_job WHERE jc_job.kco = {kco} {jobSelect},<br><br>EACH jc_costcode of jc_job WHERE TRUE {jccSelect}<br><br>You could call the dataset from within the query editor and in the parameters say jobSelect=and jc_job.job_num = 'XXXX' &jccSelect= and jc_costcode.jcc_cc = 'YYYY' (where XXXX is a vailid contract number and YYYY is a valid Costcode). |

DB Fields defined as array in the Database Enquiry such as avm_add[4] can be entered without the index of elements to display all elements (previously the element would need to be specified such as avm_add__1, avm_add__2 etc.)




| | kco | avm_add__1 | avm_add__2 | avm_add__3 | avm_add__4 |
|---|---|---|---|---|---|
| 1 | 10 | 42 Bramall Lane | | Sheffield | |
| 2 | 10 | ST JAMES HOUSE | 7 CHARLOTTE STREET | MANCHESTER | |
| 3 | 10 | 22 Ashton Gates | | Bristol | |
| 4 | 10 | BARDON HILL | COALVILLE | LEICESTERSHIRE | |
| 5 | 10 | 21 Plymouth Docks | Plymouth | Devon | |
| 6 | 10 | Colliers Industrial Estate | The High Street | Maidenhead | Berkshire |
| 7 | 10 | Coypool Rd, | Plympton | Plymouth | Devon |
| 8 | 10 | The High Street | | | Surrey |
| 9 | 10 | Slough Trading Estate | 107 - 113 Farnham Road | Slough | Berkshire |
| 10 | 10 | 15 High Street | | | |
| 11 | 10 | 12 High Row | | Darlington | |
| 12 | 10 | Northfield Retail Park | Rotherham Rd | Parkgate | Rotherham,South Yorkshire |
| 13 | 10 | 14 Jesmond road | | Newcastle | |
| 14 | 10 | 90 | Greengairs Rd | Greengairs | Lanarkshire |
| 15 | 10 | 45 High Street | | Bromley | Kent |
| 16 | 10 | Station Road | Bristol | | |
| 17 | 10 | Stairfoot Business Park | Bleachcroft Way | Barnsley | South Yorkshire |
| 18 | 10 | The High Street | | Cambridge | Cambridgeshire |

Wildcards can be used in “Fields” on Query Editor screen to mask fields




| | kco | avm_add__1 | avm_add__2 | avm_add__3 | avm_add__4 | avm_badd__1 | avm_badd__2 | avm_badd__3 | avm_badd__4 | RO_avm_printadd__1 | RO_avm_printadd__2 | RO_avm_printadd__3 | RO_avm_printadd__4 | RO_avm_bprintadd__1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 10 | 42 Bramall Lane | | Sheffield | | | | | | 42 Bramall Lane | Sheffield | S25 4DL | | |
| 2 | 10 | ST JAMES HOUSE | 7 CHARLOTTE STREET | MANCHESTER | | | | | | ST JAMES HOUSE | 7 CHARLOTTE STREET | MANCHESTER | M1 4DZ | |
| 3 | 10 | 22 Ashton Gates | | Bristol | | | | | | 22 Ashton Gates | Bristol | BS30 5SJ | | |
| 4 | 10 | BARDON HILL | COALVILLE | LEICESTERSHIRE | | | | | | BARDON HILL | COALVILLE | LEICESTERSHIRE | LE67 1TL | |
| 5 | 10 | 21 Plymouth Docks | Plymouth | Devon | | | | | | 21 Plymouth Docks | Plymouth | Devon | PL1 XYZ | |
| 6 | 10 | Colliers Industrial Estate | The High Street | Maidenhead | Berkshire | | | | | Colliers Industrial Estate | The High Street | Maidenhead | Berkshire, SL6 3ND | |
| 7 | 10 | Coypool Rd, | Plympton | Plymouth | Devon | | | | | Coypool Rd, | Plympton | Plymouth | Devon, PL7 4SS | |
| 8 | 10 | The High Street | | | Surrey | | | | | The High Street | Surrey | GU1 5BH | | |
| 9 | 10 | Slough Trading Estate | 107 - 113 Farnham Road | Slough | Berkshire | | | | | Slough Trading Estate | 107 - 113 Farnham Road | Slough | Berkshire, SL1 4UN | |
| 10 | 10 | 15 High | | | | | | | | 15 High Street | GU14 5UN | | | |

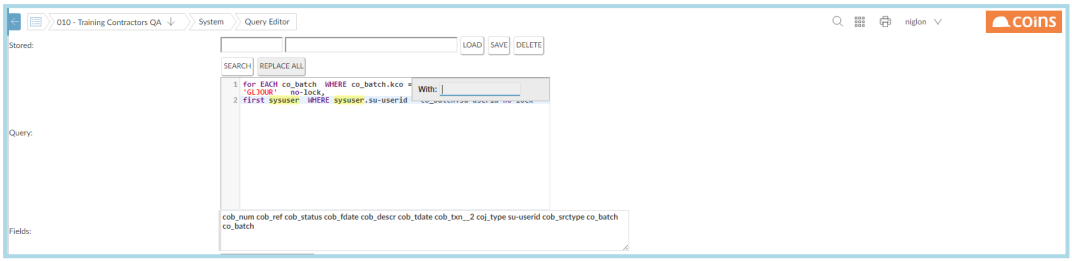
### 3.2.1 Search and Replace

Clicking Search allows you to search for a character string within the Query. This is particularly useful for large queries.

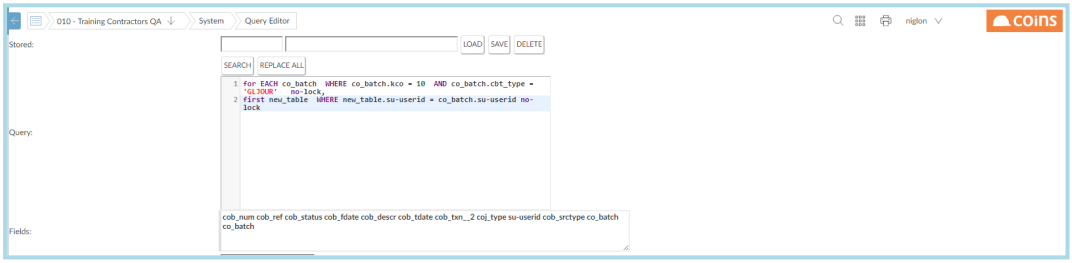

If found, the string will be highlighted within the query.


Clicking Replace All will prompt for the search string to be replaced


Enter the required string and click Enter. The system will then prompt for the replacement string.




Enter the replace ment string and press Enter. The system will then replace all instances of original text with the replacement string.




### 3.2.2 Exporting from the Query Editor

To export the query and associated results to excel simply click Export . The query editor will then open the data set in a new screen. This information can then be exported to Excel by right-clicking anywhere in the data table.

The spread sheet created will contain the appropriate query and links to coins so that the data can be refreshed at any point.

### 3.2.3 Saving Queries in Query Editor (10.27)

To prevent the need to keep re-typing regularly used queries, save and load options are available as of v10.27 and are located at the top of the Query Editor page.

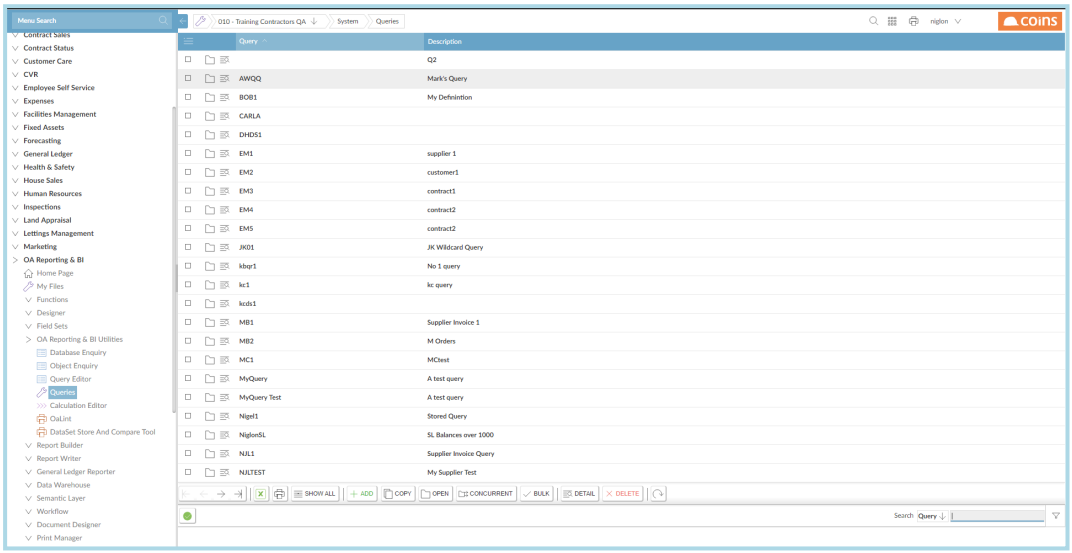
Once you have set up your query and selected the required fields, in the Stored fields, specify a name for the query and a description to further identify it.

Click Save

The Query will now be stored and can be retrieved at any time by running Query Editor, specifying the Query name and clicking Load button.

To delete a saved query, specify the name and select the Delete button.

The full list of stored queries may be viewed from the new Queries option from the OA Reporting and BI Utilities Menu



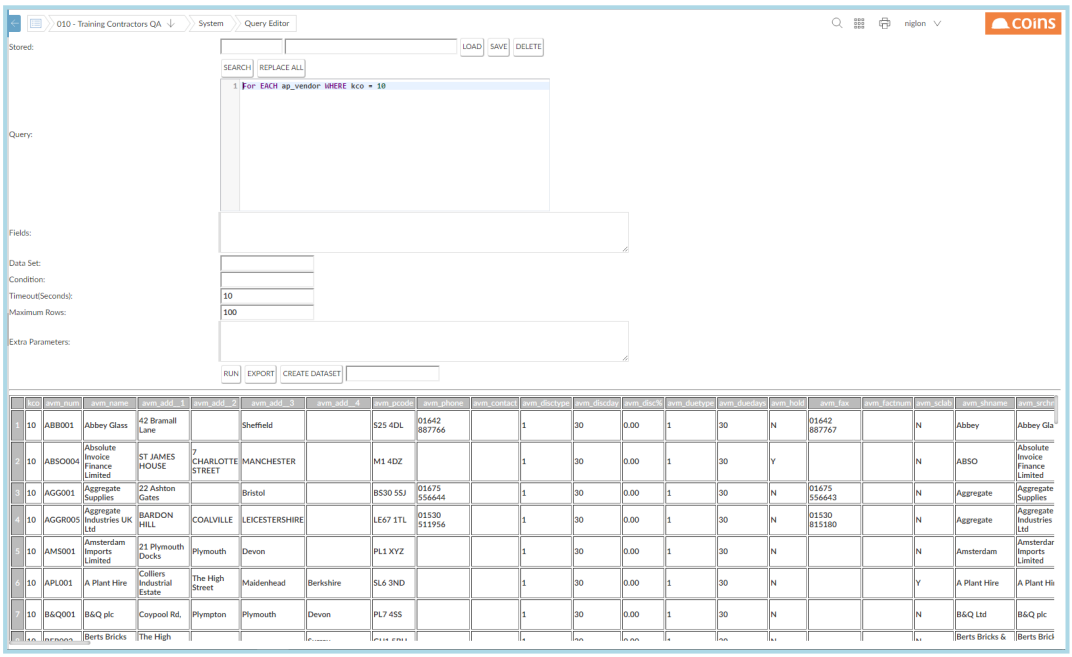

This screen will allow the creation, deletion and maintenance of the stored queries.

> **Note: To see the results of the queries, you will need to return to Query Editor and Load the appropriate query.**

### 3.2.4 Creating Datasets from the Query Editor

When building datasets, designers often use the Query Editor to test the query and select their fields before moving onto Dataset Maintenance and transferring the design to there. In 10.27 onwards, it is now possible to send the basic Dataset design directly into Dataset Maintenance without the need for manual re-keying.

A new button Create Dataset is now available at the bottom of the Query Editor fields, located next to the Export button:



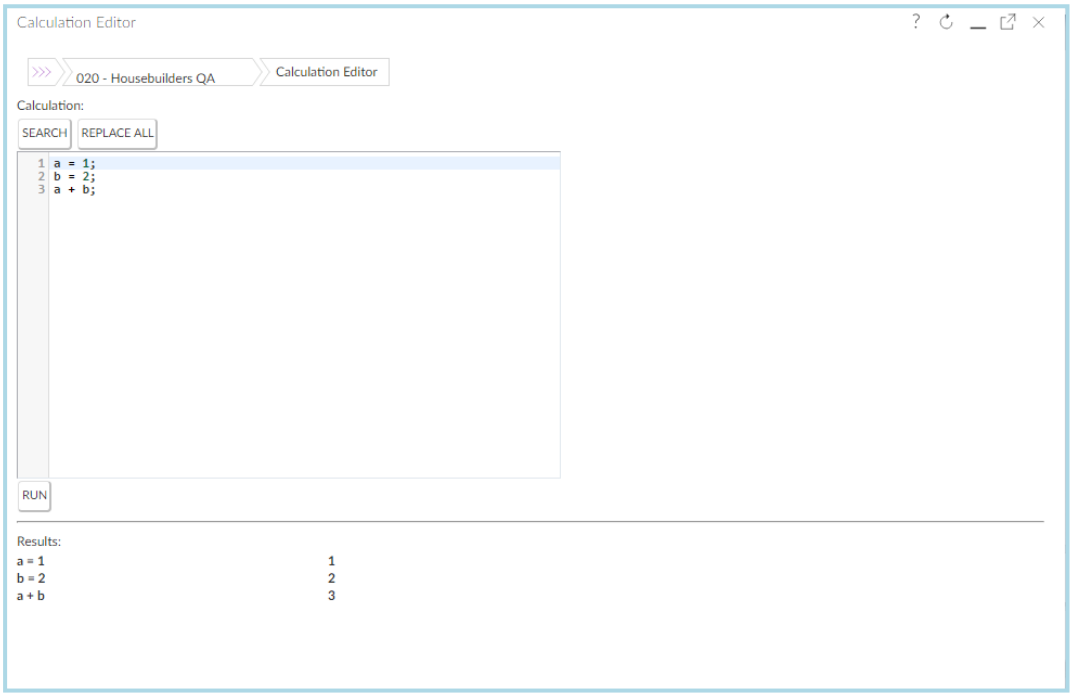

Once you have a working query and field selection, enter a name for the new dataset and Select the Create Dataset button.

> **Note: You need to specify the fields to be created in the dataset by adding them to the Fields section. Leaving this blank will result in the dataset fields section being empty.**

The dataset will be created and will be available in Dataset Definitions in the OA Designer Menu.

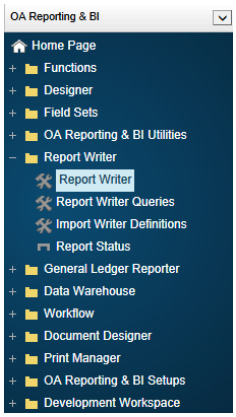
## 3.3 Calculation Editor

The Calculation Editor allows you to try out calculations using the coins business logic calculation methods (see accompanying documentation). This function can be useful to test calculations before being used in reports and enquiries.

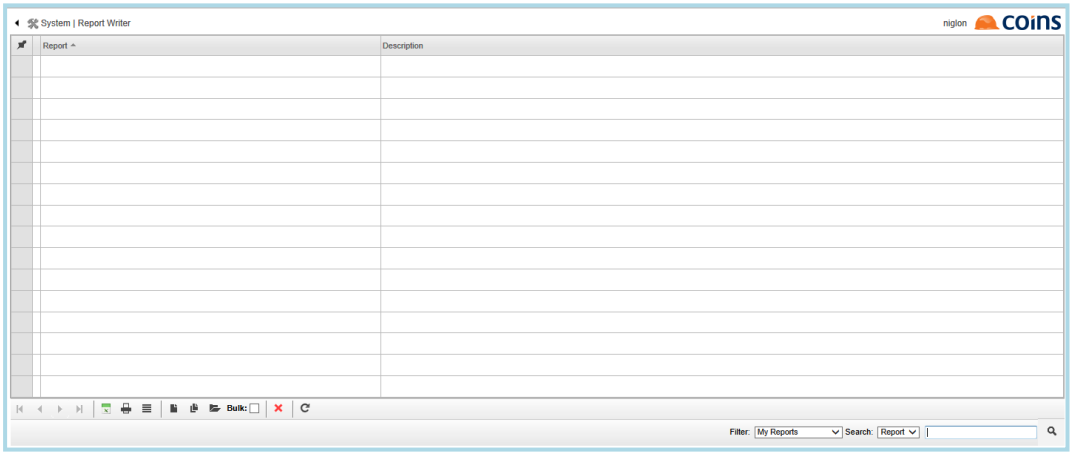



Simply enter in the calculations using the appropriate syntax and click run to test the results.

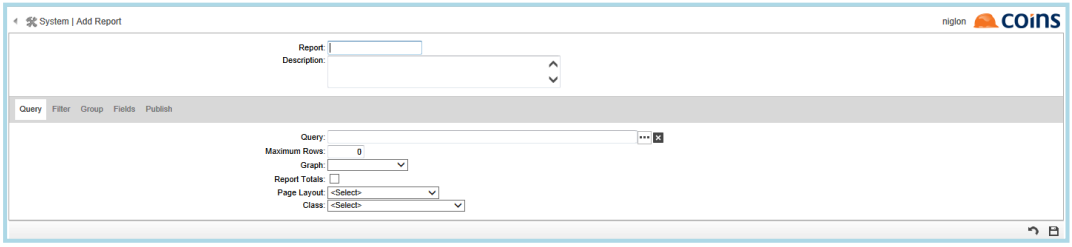
# 4 Report Writer - Creating a New Report



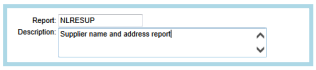

Report Writer is accessed from the OA Reporting and BI Menu.




To create a new report, select the New icon .

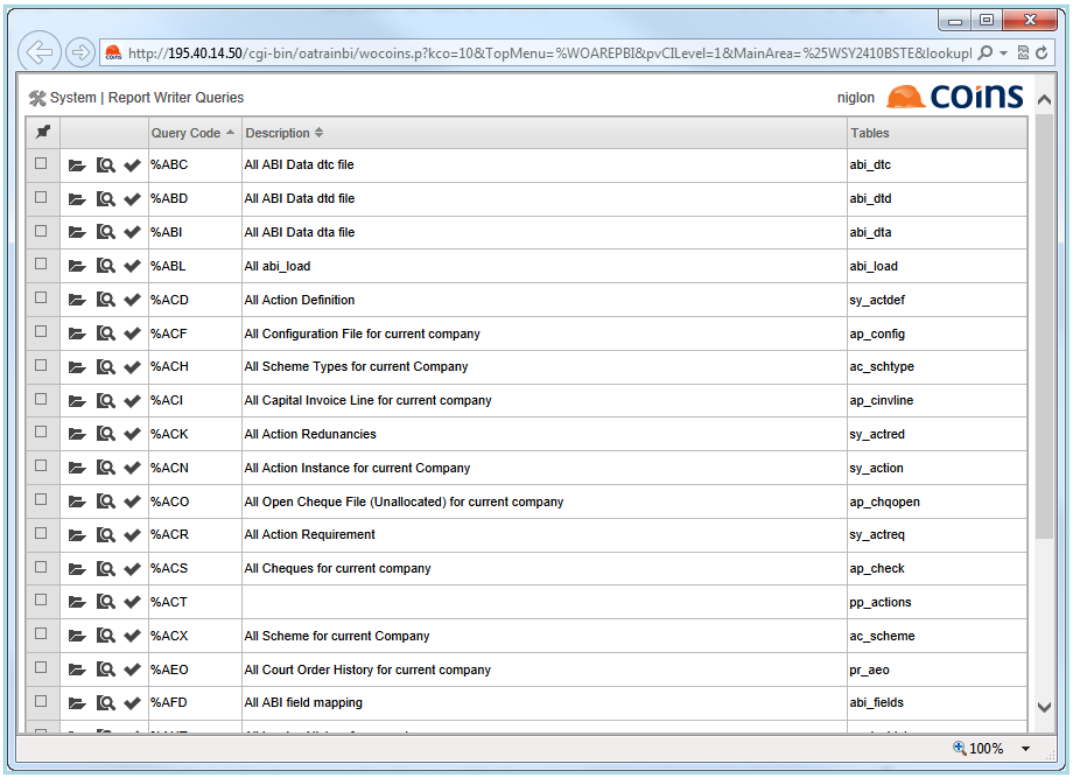



In this panel, assign a code and description to the report. The code will be used to identify the report later on and the description should contain a brief overview of the purpose of this report so that others Users will know what it does.




You will then need to select a Query from the pre-defined cache of queries.


Click on the Lookup button ▪▪▪ to view all the available queries.




A Query is a pre-defined section of PROGRESS 4GL code that selects the records that the report will include. COINS comes with a large number of pre-installed queries and your system administrator may also have added some additional queries for your use.

NOTE : Queries are filtered by table security; you will only see queries for which you have access to all the tables.

To help you find the query you need, a filter is provided in the lower right-hand side of the screen

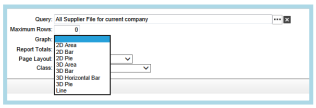
You may search by Query Code, Query description, or the data tables used by the query. Remember to press the filter button when you change the filter selection.

You can modify the basic queries later on by setting up filters, groups and sort orders.

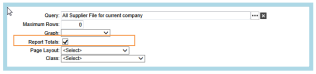
The queries will extract the basic information for the report from the database and in some instances provide default Report Selection criteria. Select the query required and assign to the report using the Choose button .

If you want to report on a selected number of records then you can enter a number in the Maximum Rows field to limit the number of rows on the report. (Eg:- useful if you want to report on the Top 5). Leave this blank to run a full report.

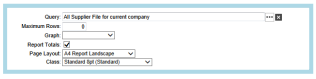
If a graph is required on the report then select the graph type to be included on report.



Using the Report Totals tick box, confirm whether you want the report to have run totals at the end of the report.



It is also necessary to confirm the Page Layout (this can be selected from a predefined list of standard options) and Class (again this can be selected from a predefined list of standard options).

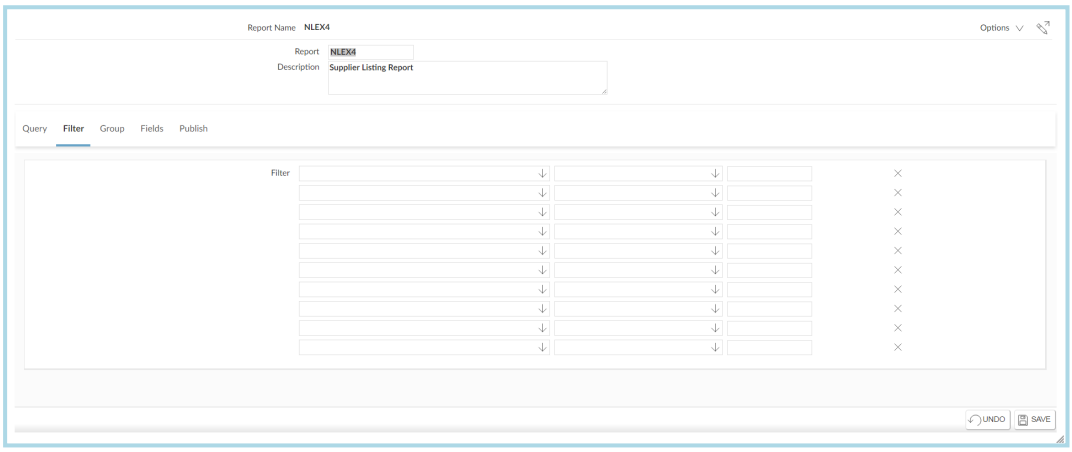


The Page Layout determines the paper size and orientation and the report class will determine font, size and formatting of the body text of the report.

Once you have assigned all of the above information Click  to save the report.

## 4.1 Adding Filters to the Report

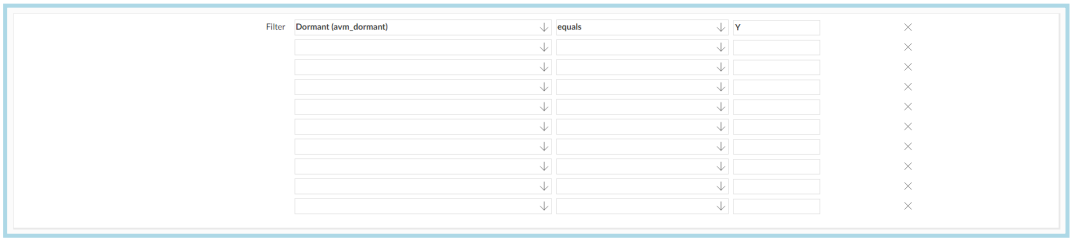
Filters can be applied to the main report query to further refine the information that is included in, or excluded from, the report.



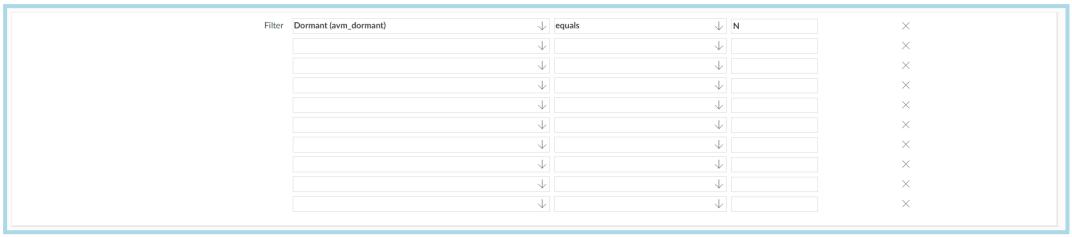

The First Column of the Filter Tab will provide a selection list of the fields available in the table being queried. Select the field required, the nature of the filter and the criteria against which the filter is to be applied.

The second Column is the comparison list and allows you to determine the type of filter to be applied against the field you have chosen.

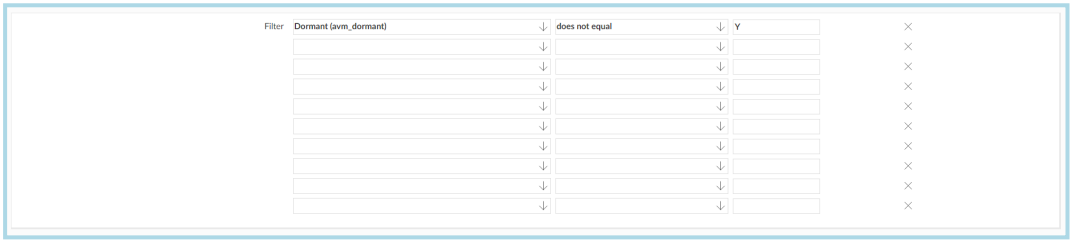
The third column allows you to enter the value to compare against the filter field. So, in our example, to create a filter to show only dormant suppliers the filter criteria would be:




By contrast, a report to show all suppliers excluding the dormant accounts could be either:




Or



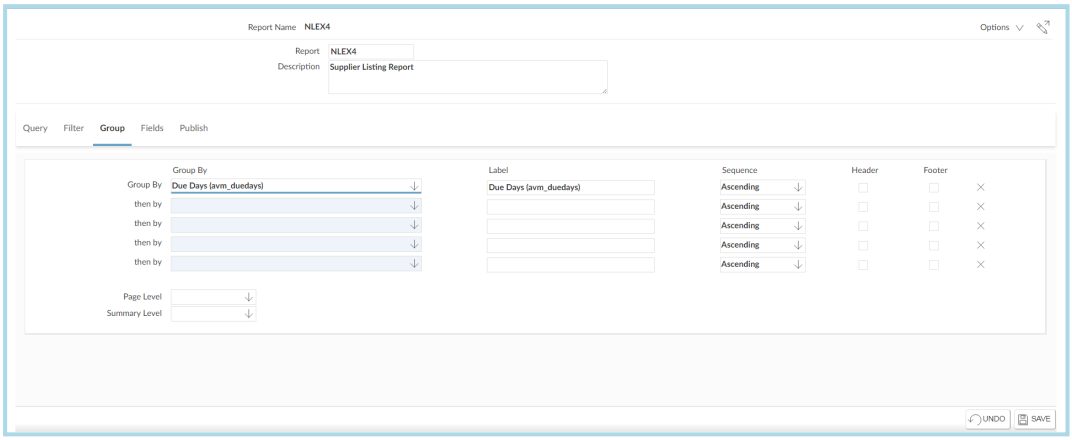

**Note: If you choose Matches from the comparison list, you can use a can-do (wildcard) list in the value field. So for example to include only cost codes that end with 99, select Cost Code from the Filter list, select Matches from the comparison list, and type *99 in the value field.**

Repeat the process for other fields you want to filter on. Note that the filters are 'AND' filters; so for example if you specify contracts where Contract Location equals London, and where Contract Manager equals John Smith, the report will only include contracts in London that are managed by John Smith.

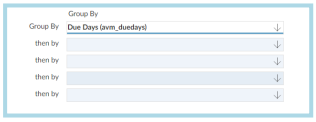
Once all filters have been assigned, Click Save

## 4.2 Sorting, Sub-totalling, Page Breaks and Summary Levels

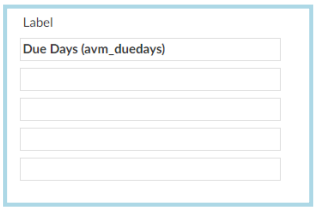
The options under the Group Tab define how the data will be sorted on your report.




In the Group By column, select the field you want to group the report by. Records on which the value of this field are the same will be grouped together and sorted in the order you specify, and you can show totals for each group.

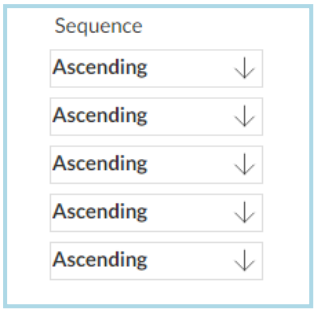



In the Label column, enter a label to show on the headers and footers (before and after each group).




NOTE : Your system administrator may set up more detailed group headers (using Default Report Labels), which show, for example, a code and a description for each group.

Choose whether to sort the group in ascending or descending order.




Choose whether or not to display group headers (a label and the value of the group by field) before each group, and group footers (a label and the value of the group field, plus the totals of any totalled columns).


If you want the report to start on a new page when the value of one of the fields changes (for example, to print information about different contracts on separate pages), use Page Level to select the field.


If you want to produce a summarised report based on details that you do not want to show, use Summary Level to select the field to summarise by.

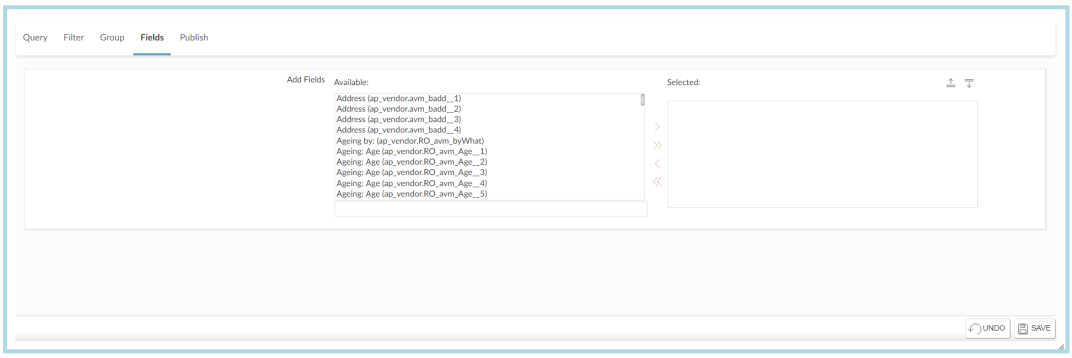
For example, if you want to produce a report on contract costs by section, you could produce a cost code report, but summarised by section. This would show the cost code totals for each section, but would not show the individual costs for each cost code.

> Note: **Experiment with the options to see the diverse effects.**

## 4.3 Selecting the fields to print on the report

### 4.3.1 New Reports

The fields Tab allows you to specify which fields from the data table will appear on the printed report.




When you are creating a new report you will be able to select the fields required from a list of all fields available list using the green arrows to select. Once a field has been moved to the selected column, it can be highlighted and removed again using the red arrows.

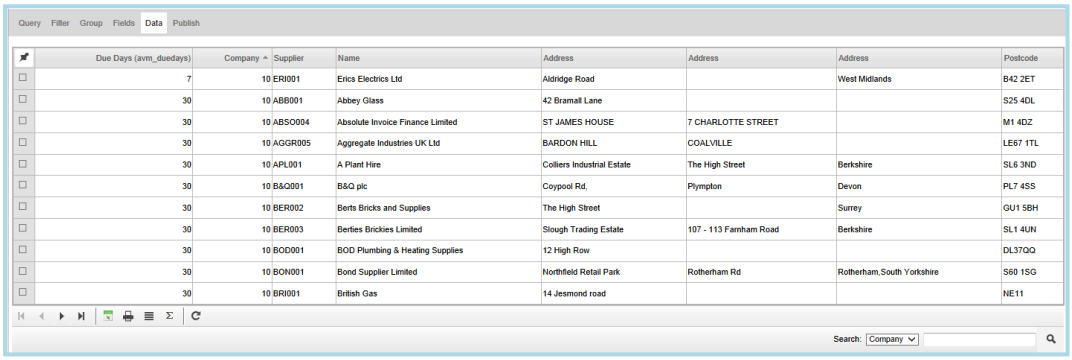

You can use the filter to find the fields you require. Simply type in a few letters of the field, or label required, in the box below the field names and the list will be dynamically filtered.

To change the order of the field in the selected column, highlight a field and use the up and down selectors  to reposition the field in the list.

## 4.4 Previewing or exporting the data

When you are writing the report, the Data tab shows the data that the report will contain.

If the report requires selection criteria, the Input tab allows you to select which data to include, in the same way as when you run the report.




| Due Days (avm_duedays) | Company | Supplier | Name | Address | Address | Address | Postcode |
|---|---|---|---|---|---|---|---|
| 7 | 10 | ERI001 | Erics Electrics Ltd | Aldridge Road | | West Midlands | B42 2ET |
| 30 | 10 | ABB001 | Abbey Glass | 42 Bramall Lane | | | S25 4DL |
| 30 | 10 | ABSO004 | Absolute Invoice Finance Limited | ST JAMES HOUSE | 7 CHARLOTTE STREET | | M1 4DZ |
| 30 | 10 | AGGR005 | Aggregate Industries UK Ltd | BARDON HILL | COALVILLE | | LE67 1TL |
| 30 | 10 | APL001 | A Plant Hire | Colliers Industrial Estate | The High Street | Berkshire | SL6 3ND |
| 30 | 10 | B&Q001 | B&Q plc | Coypool Rd, | Plympton | Devon | PL7 4SS |
| 30 | 10 | BER002 | Berts Bricks and Supplies | The High Street | | Surrey | GU1 5BH |
| 30 | 10 | BER003 | Berties Brickies Limited | Slough Trading Estate | 107 - 113 Farnham Road | Berkshire | SL1 4UN |
| 30 | 10 | BOD001 | BOD Plumbing & Heating Supplies | 12 High Row | | | DL37QQ |
| 30 | 10 | BON001 | Bond Supplier Limited | Northfield Retail Park | Rotherham Rd | Rotherham,South Yorkshire | S60 1SG |
| 30 | 10 | BRI001 | British Gas | 14 Jesmond road | | | NE11 |

When you click the  [Apply Filter] button, COINS refreshes the Data tab with the data you specified.

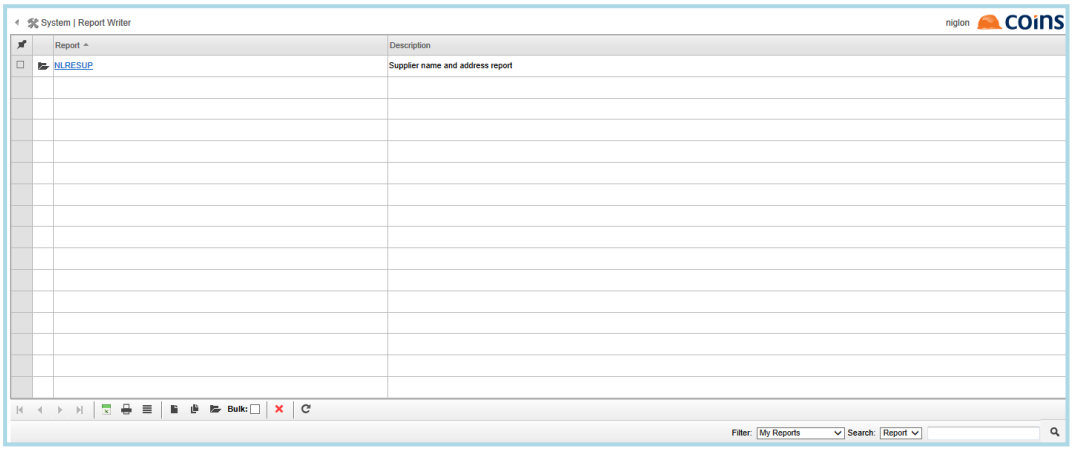
On the Data tab, you can also:

Show totals (for the fields marked to show totals): click the  button.

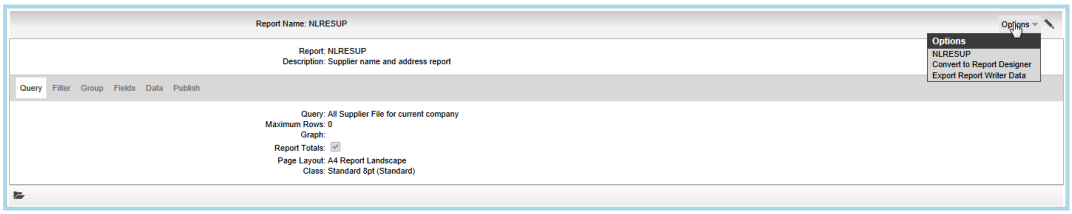
Export the data: click the  [Export] button.

## 4.5 Running a Report

1. Go to Report Writer.
2. Find the report you want to produce and click the link in the Report column.




3. From the Options menu, select the name of the report




4. Choose how you want COINS to produce the report:

   - If the report requires selection criteria, you can select which data to include on the report. For example, if there is a contract selection tab, you can choose which contracts to report on, according to the different fields that are available on the tab.
   - Choose the type of output you want. For example, you can choose to display the report on your screen as soon as it is ready (foreground). You can choose to email the report to someone, or to have it generated at a later date or time.

5. Click the forward button 

The report can then be accessed via the Report Status Workbench in the normal manner.

# 5 Publishing Report Writer Reports

Once a report has been created in Report Writer the author may wish to make this available to other COINS users who do not have access to Report Writer.

Three options are available:

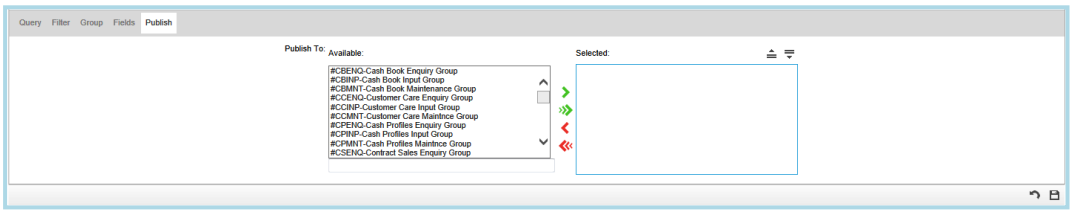
- Publish to Report Runner – Requires a Report Runner Licence per user. This allows the Report Writer user to publish this report to the appropriate COINS User Groups. Those Users who have access both to the Report Runner and the defined User Group will have access to the run the Report via the Report Status Workbench.
- Export to Function – Creates a standard COINS function that may be added to any COINS menu by an Administrator. This requires no additional licencing.
- Export to Designer – Requires an OA Designer Licence. This is the same as Export to Function, but will open the report in OA Designer for additional Report Designer functionality to be added to the report.

## 5.1 Publishing a report to Report Runner

To use this feature, you must have a Report Runner licence for each user who will need to access the report – they will only be able to run the reports, not make any changes to it.

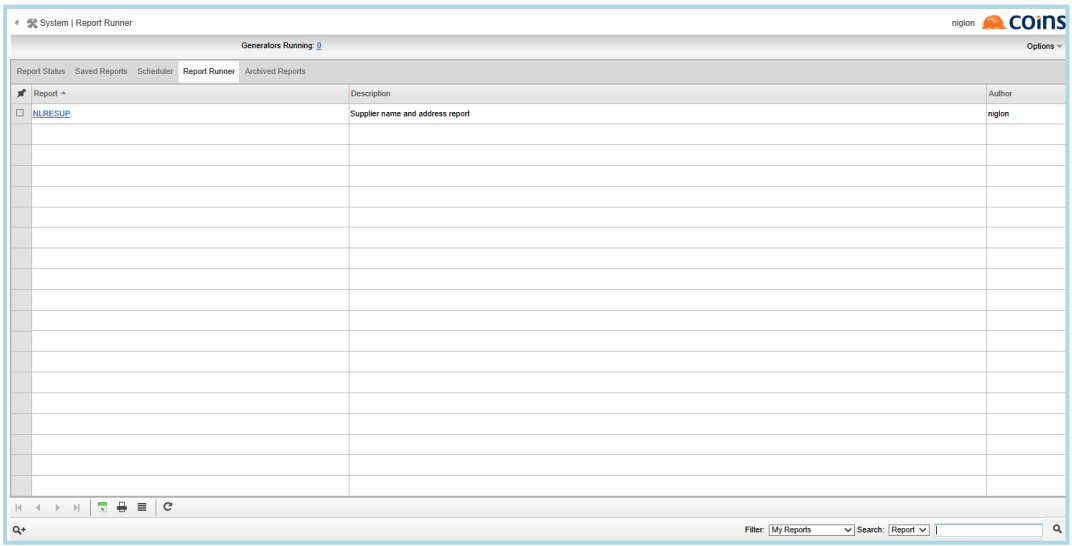
Open the report in Report Writer and select the Publish Tab.

Select the User Groups as required from the available list using the green and red arrows to select and deselect respectively.



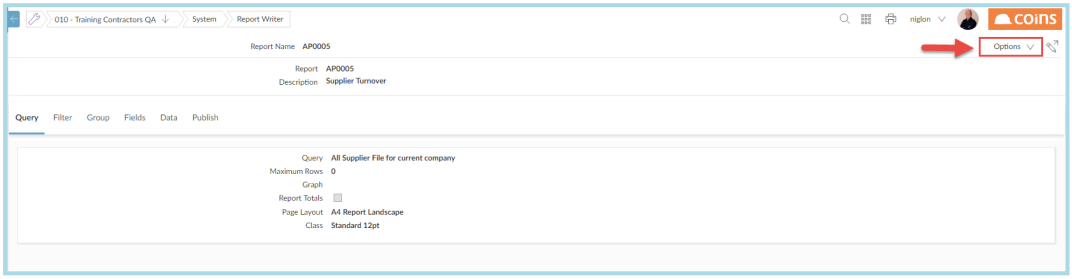

You can use the filter to find the fields you require. Simply type in a few letters of the field or label required and the list will be dynamically filtered.

Those Users with access can then simply run the report from the Report Status Workbench by clicking on the link to the Report Code.

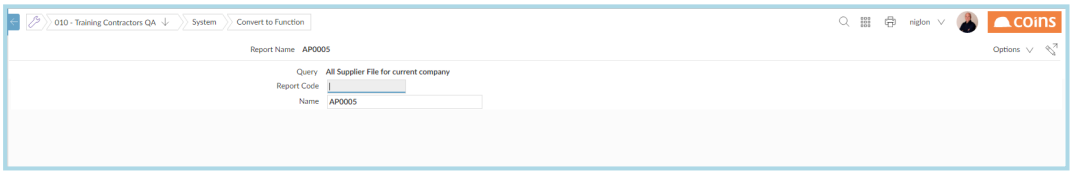


## 5.2 Convert to Function

Open the desired report in Report Writer and click the Options button in the top right of the screen




From the drop-down menu, click Convert to Function




| Field | Description |
|---|---|
| Report Code | Enter here the name of the function that is to be created. This may be the same name as the report in Report Writer (which may assist you in locating the report in the future if you need to make changes), or any other function name that meets your own internal standards. If in doubt, consult with your system administrator on a suitable naming convention. Do not include spaces or special characters in the function name. |
| Name | This is the description of the report that will appear on the COINS menus once your administrator has added it.<br><br>For example: Outstanding Invoices Report |

Once you have completed the fields above, click → NEXT

The function will be created and you will be returned to the Report Writer screen. You can now advise your system administrator of the details so that they can add this to a COINS menu and add the function to the required security groups (users will not be able to see the report unless they are granted security access to it).

If the function name has already been used, you will get a Function already exists error and you will either need to ask your Administrator to delete the old version (if appropriate), or choose a different Report Code).

### 5.2.1 Amending Reports that have been converted to Functions

Converted reports are separate to the Report Writer tool. If you need to make changes to a converted report, open the report in Report Writer and make any changes as required. You will then need to ask your administrator to remove the existing function and Report Design before you can follow the steps above again.

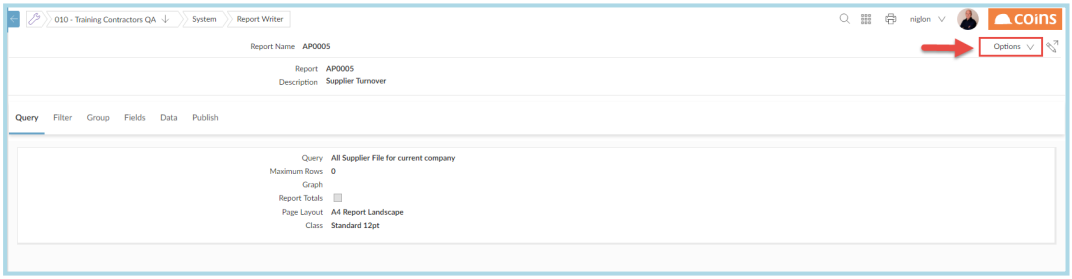
## 5.3 Convert to Report Designer

If you have an OA Designer licence on your system, you can take a Report Writer Report and convert it to an OA Report Designer report. This will perform the same outcome as Convert to Function, but will then open the Report in OA Designer so that more complex report functionality can be added based on the original report design

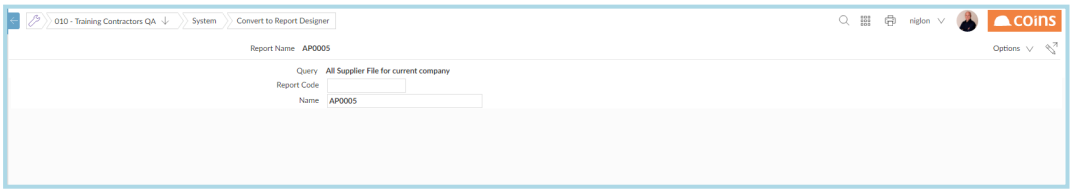
.

> Note: **Please note: Once you start to make changes in OA Designer, the report cannot then be maintained in Report Writer as OA Designer reports are not backwards compatible with Report Writer.**

Open the desired report in Report Writer and click the Options button in the top right of the screen




From the drop-down menu, click Convert to Report Designer.



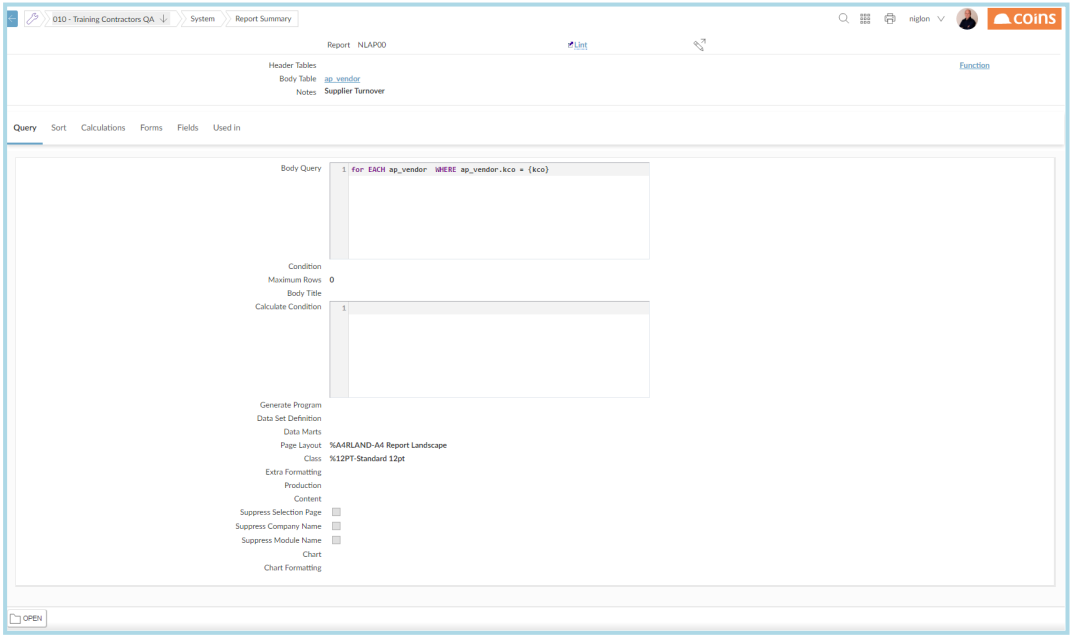

| Field | Description |
|---|---|

| | |
|---|---|
| Report Code | Enter here the name of the function that is to be created. This may be the same name as the report in Report Writer (which may assist you in locating the report in the future if you need to make changes), or any other function name that meets your own internal standards. If in doubt, consult with your system administrator on a suitable naming convention. Do not include spaces or special characters in the function name. |
| Name | This is the description of the report that will appear on the COINS menus once your administrator has added it.<br><br>For example: Outstanding Invoices Report |

Once you have completed the fields above, click → NEXT

If the function name has already been used, you will get a Function already exists error and you will either need to ask your Administrator to delete the old version (if appropriate), or choose a different Report Code).

The function will be created and the report design will open in the OA Report Designer.



Once you have completed your design in OA Report Designer, you can advise your system administrator of the details so that they can add this to a COINS menu and add the function to the required security groups (users will not be able to see the report unless they are granted security access to it).

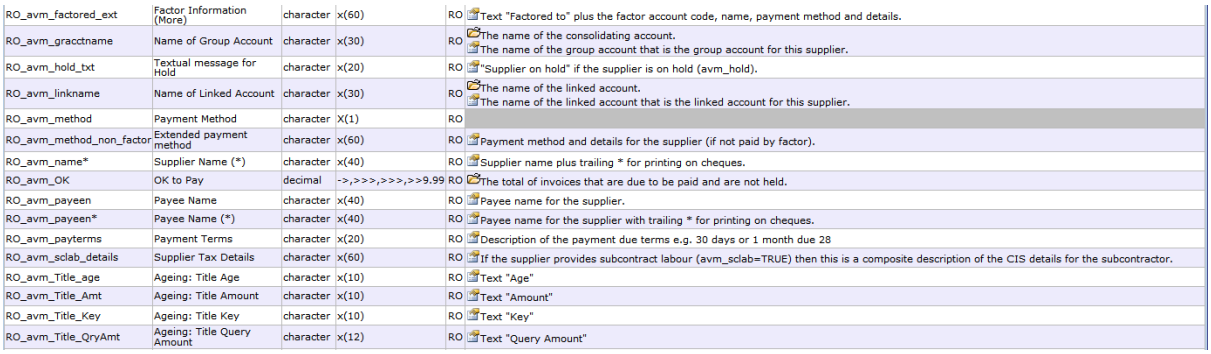
### 5.3.1 Amending Reports that have been converted to OA Designer

Converted reports are separate to the Report Writer tool. If you need to make changes to a converted report once you have made changes in OA Report Designer, you must continue to make any additional changes in Report Designer.

Reports in OA Designer are not backwards compatible with OA Report Writer.

# 6 Read Only (Calculated) Fields

In addition to the standard tables and fields in the COINS database, Open Architecture also provides access to certain calculated and non standard fields. These are known as “RO” or Read Only fields and are also fully documented in the Database Enquiry.[1]



| | | | | | |
|---|---|---|---|---|---|
| RO_avm_factored_ext | Factor Information (More) | character | x(60) | RO | Text "Factored to" plus the factor account code, name, payment method and details. |
| RO_avm_gracctname | Name of Group Account | character | x(30) | RO | The name of the consolidating account.<br>The name of the group account that is the group account for this supplier. |
| RO_avm_hold_txt | Textual message for Hold | character | x(20) | RO | "Supplier on hold" if the supplier is on hold (avm_hold). |
| RO_avm_linkname | Name of Linked Account | character | x(30) | RO | The name of the linked account.<br>The name of the linked account that is the linked account for this supplier. |
| RO_avm_method | Payment Method | character | X(1) | RO | |
| RO_avm_method_non_factor | Extended payment method | character | x(60) | RO | Payment method and details for the supplier (if not paid by factor). |
| RO_avm_name* | Supplier Name (*) | character | x(40) | RO | Supplier name plus trailing * for printing on cheques. |
| RO_avm_OK | OK to Pay | decimal | ->,>>>,>>>,>>9.99 | RO | The total of invoices that are due to be paid and are not held. |
| RO_avm_payeen | Payee Name | character | x(40) | RO | Payee name for the supplier. |
| RO_avm_payeen* | Payee Name (*) | character | x(40) | RO | Payee name for the supplier with trailing * for printing on cheques. |
| RO_avm_payterms | Payment Terms | character | x(20) | RO | Description of the payment due terms e.g. 30 days or 1 month due 28 |
| RO_avm_sclab_details | Supplier Tax Details | character | x(60) | RO | If the supplier provides subcontract labour (avm_sclab=TRUE) then this is a composite description of the CIS details for the subcontractor. |
| RO_avm_Title_age | Ageing: Title Age | character | x(10) | RO | Text "Age" |
| RO_avm_Title_Amt | Ageing: Title Amount | character | x(10) | RO | Text "Amount" |
| RO_avm_Title_Key | Ageing: Title Key | character | x(10) | RO | Text "Key" |
| RO_avm_Title_QryAmt | Ageing: Title Query Amount | character | x(12) | RO | Text "Query Amount" |

Although these fields have certain restrictions, they are incredibly powerful when used in enquiries and reports. For example many of the calculated fields reflect similar fields to the COINS + Configurable Reporter, such as Accruals, Costs and Revenue fields.

There are three main types of calculated fields :

Simplethat return descriptions from associated tables

Calculatedthat return calculated values from associated tables

Complex(also known as parameter drive) that return calculated values from associated tables based on parameter passed to the fields

1

# 7 Using Parameter driven RO_ fields in Report Writer

Certain RO_ fields can be passed parameters to enhance the information returned to a report. Typically these fields can be limited by dates, values and financial periods as well simply parameters such as "TD" for a To date value.

The parameters available for each RO_ field are documented in the Database Enquiry, below is an example :

In the Database Enquiry RO_ fields are shown in a format as the example below. Any parameters immediately after the caret are mandatory, each parameter is then separated by a pipe. Any parameters which are encapsulated in square brackets are optional.

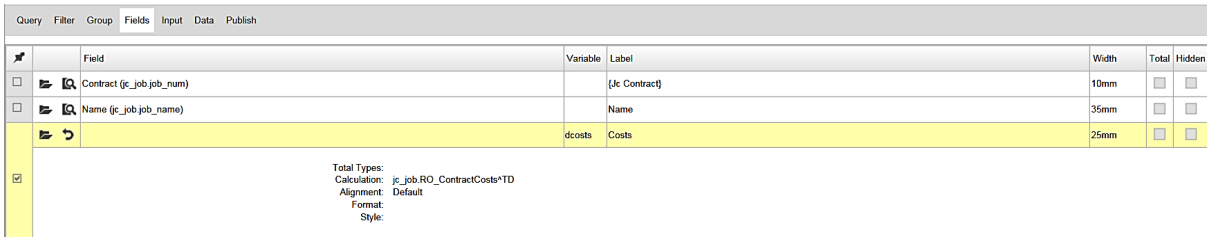
RO_ContractCosts^<PeriodType>[|<PeriodOffset>[|<FDate>[|PhaseMasks [|CostcodeMasks[|CategoryMasks[|AnalysisMask]]]]]

In order to use the Read Only (RO_) fields in the Report Writer, they can either be exposed, i.e. document the parameters which are to be passed to a field, or they may be used directly in a calculation within a column on the report.

## 7.1 Using RO Fields directly in a calculation

RO fields, unless exposed, cannot be selected as a column by the user. However, they may be used within a calculation in a report column which allows the report writer user to directly control how the RO field is used and can change this at any time without needing to refer to a system administrator.

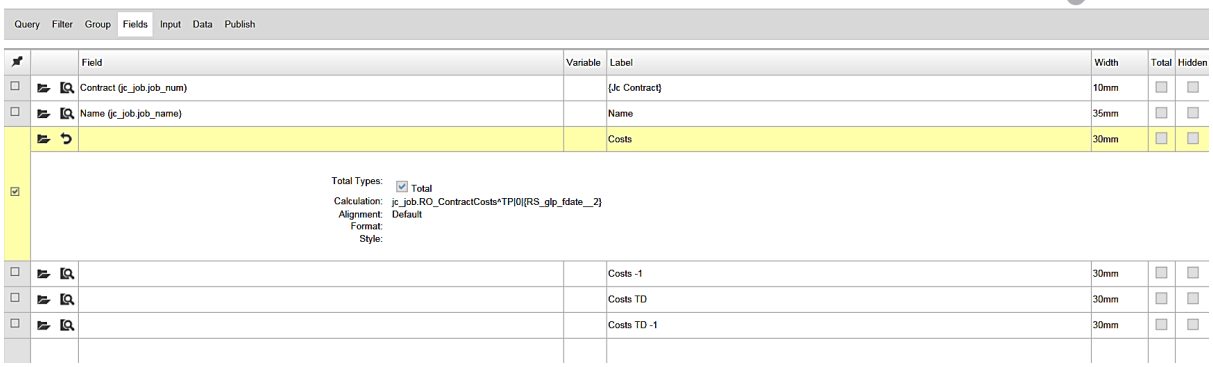
To create a calculated field, select ADD on the fields tab and leave the Field name blank. Enter a label for the field and within the calculation field enter the RO field and the required parameters. The field must be fully qualified as table.fieldname such as jc_job.RO_ContractCosts^TD|0| If the table name is not specified, the field will not work as Report Writer will assume it is a variable rather than a database field.




A variable name will need to be assigned in the Variable column if this column is to be included in Totals, as calculations require a variable to accumulate totals.

RO fields in calculations may also be passed values from sources such as input forms – if available in the query (for example period selections) using OA standard substitution {}. So if our input screen has a period field of RS_glp_fdate__2, our RO field might become:

jc_job.RO_ContractCosts|0|{RS-glp_fdate__2}. This RO field will then generate different figures depending on the period date selected by the user when they run the report, making the report much more dynamic.


Query
Filter
Group
Fields
Input
Data
Publish
Field
Variable
Label
Width
Total
Hidden
Contract (jc_job.job_num)
{Jc Contract}
10mm
Name (jc_job.job_name)
Name
35mm
Costs
30mm
Total Types:
Total
Calculation:
jc_job.RO_ContractCosts^TP|0|{RS_glp_fdate__2}
Alignment:
Default
Format:
Style:
Costs -1
30mm
Costs TD
30mm
Costs TD -1
30mm