



KReporter v3.0

Custom Functions, Formulas, KReporter Fields

<http://www.youtube.com/user/theKReporter>



<http://www.facebook.com/kreporter.org>



Basics about Custom Functions and Formulas..... 3
KReporter Fields 4
Functions & Custom Functions..... 7
Formulas..... 12

Basics about Custom Functions and Formulas

The main concept behind KReporter is to gather data from the database out of the various modules Sugar can offer. There is a couple of standard transformations for values and of course a wealth of functionalities for data manipulation like grouping, averages, etc.

However there is typically in many case a need to apply custom transformation on data – either during selection or after selection from the database.

KReporter supports that with 3 different methods that help achieve this.

- (1) KReporter Fields: those are fields that are defined on vardef level in SugarCRM. They are available for the user and represent non database fields that gather data with certain SQL logic during Selection from the database. On the one hand they are easy for the user but on the other hand are more or less static in their definition. They are typically a good option for regular used data transformation
- (2) Custom function offer the possibility to add SQL logic within the reporter. The SQL code injected here can be defined through the KReporter design interface. Thus they are quite flexible but require knowledge of SQL statement. Those is a typical option that can be used for simpler or not frequent used transformations.
- (3) Formulas are applied after data has been selected. While (1) and (2) are processed on the database this logic is processed within PHP. This is a good option for data that you cannot get from the database or where additional data is required.

KReporter Fields

To define KReporttype fields they need to be added to the vardef of a module. Take an example that you want to add a field for the weighted amount of an opportunity where you simply multiply the amount by the probability of the opportunity and return that amount.

TO create the KReporter field add a file (I called) Opportunity.kreporter.weightedamount.php to directory custom/Extension/modules/Opportunities/Ext/Vardefs with the following content:

```
<?php
$dictionary['Opportunity']['fields']['weighted_amount'] = array(
    'name' => 'weighted_amount',
    'vname' => 'LBL_WEIGHTED_AMOUNT',
    'type' => 'kreporter',
    'source' => 'non-db',
    'kreporttype' => 'currency',
    'currency_id' => 'currency_id',
    'eval' => '{t}.amount * {t}.probability / 100'
);
?>
```

The label LBL_WEIGHTED_AMOUNT is defined in another file (I called) en_us.kreporter.opportunities.lang.php in directory custom/Extension/modules/Opportunities/Ext/Language. The content of the file is as follows:

```
<?php
$mod_strings['LBL_WEIGHTED_AMOUNT'] = 'weighted Amount';
?>
```

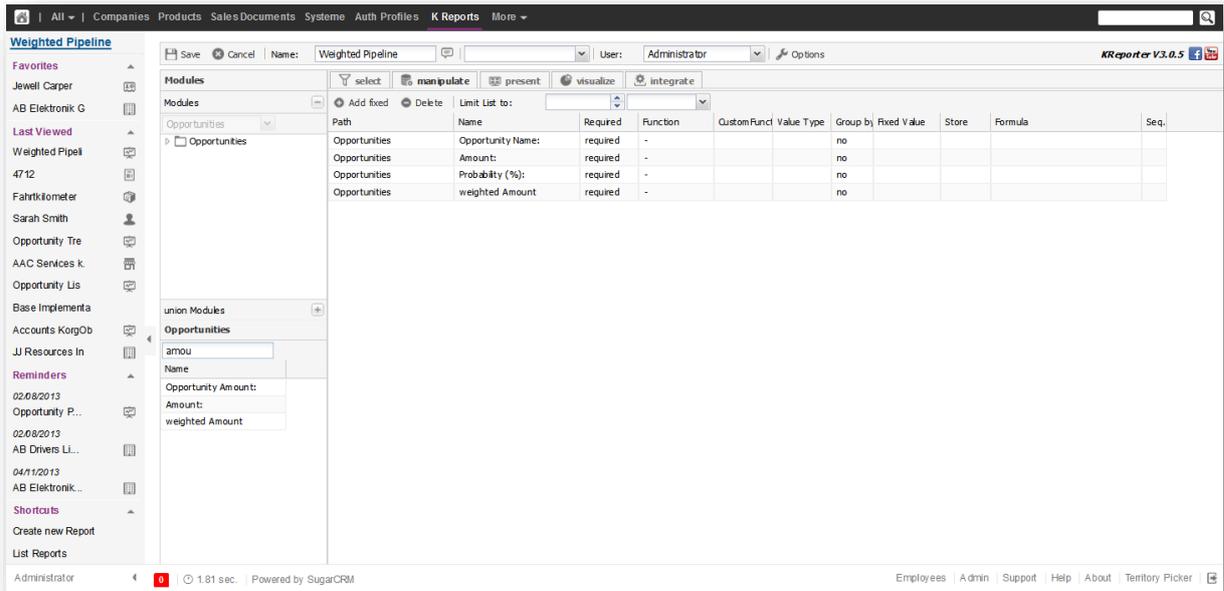
Do a quick repair in the admin panel to add this to the vardefs of Opportunities.

What this does is to add a non-db field (no data is stored in the database) that is only visible in KReporter when you create a report. 'name' and 'vname' are the typical settings. The type need to be set to 'kreporter' and the source to 'non-db'. With the kreporttype property you can define how the field is treated by the reporter in regards to select options and rendering. In our case we want to see this as a currency. The additional option 'currency_id' tells KReporter which currency to display. If this is not set it is displayed in the systems currency.

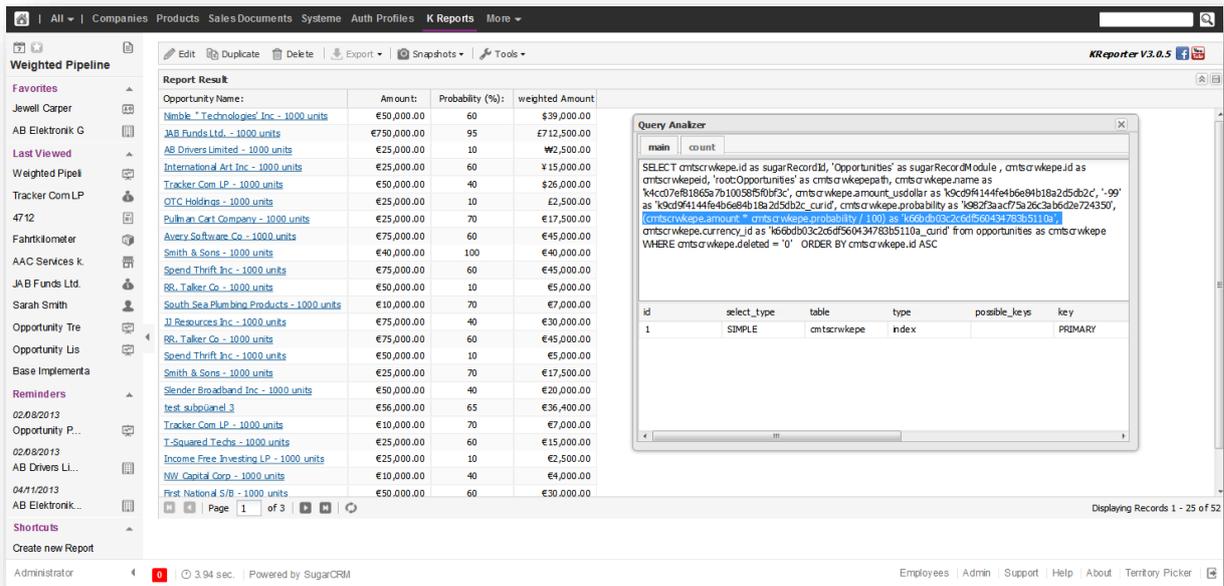
Last but not least the eval option is defined. This is an SQL statement that is dynamically replaced instead of simply selecting the fields. The options you can use are {t} for the table of the module and {tc}for the _cstm table of the module if you are using custom fields defined with Studio.

In our case we multiply the amount {t}.amount from the opportunity with the Probability {t}.probability and divide by 100.

If we create a new report for an opportunity the available fields look as follows and we can easily drag the kreporter field over to the listed fields.



If we save the report and run it the result looks as follows.



The field as stated in the eval statement is evaluated. If we look at the SQL statement generated in the Query Analyzer you can see that instead of simply selecting the field the statement I evaluated and the {t} we used in the eval statement if replaced by the alias for the opportunities table in this select statement. The field is rendered as a regular currency fields and the currency symbol is rendered from the currency_id field on the opportunity.

The KReporter field will also be evaluated if you use it in a where clause but has its limitations there due to the SQL language. This is still an option but in some cases is not sufficient when you e.g. need

to use an exist clause in the where statement and a subselect in the select part. To use the extended functionality you can define the eval variable now as an array with the following sections:

'presentation' another array that only contains one element named 'eval' that holds a snippet of SQL code. 'selection' an array of sql snippets where the index of the array equals the operator as listed in the section described above. With that also the list of available operators can be limited and each operator can have a custom SQL code injected.

The available variables in the SQL strings are:

{t} for the table we are on.

{p1} and {p2} for the 2 selection parameters. If they are filled are determined by the operator and you need to react to that.

An example for a kreporter field in vardefs can look as follows:

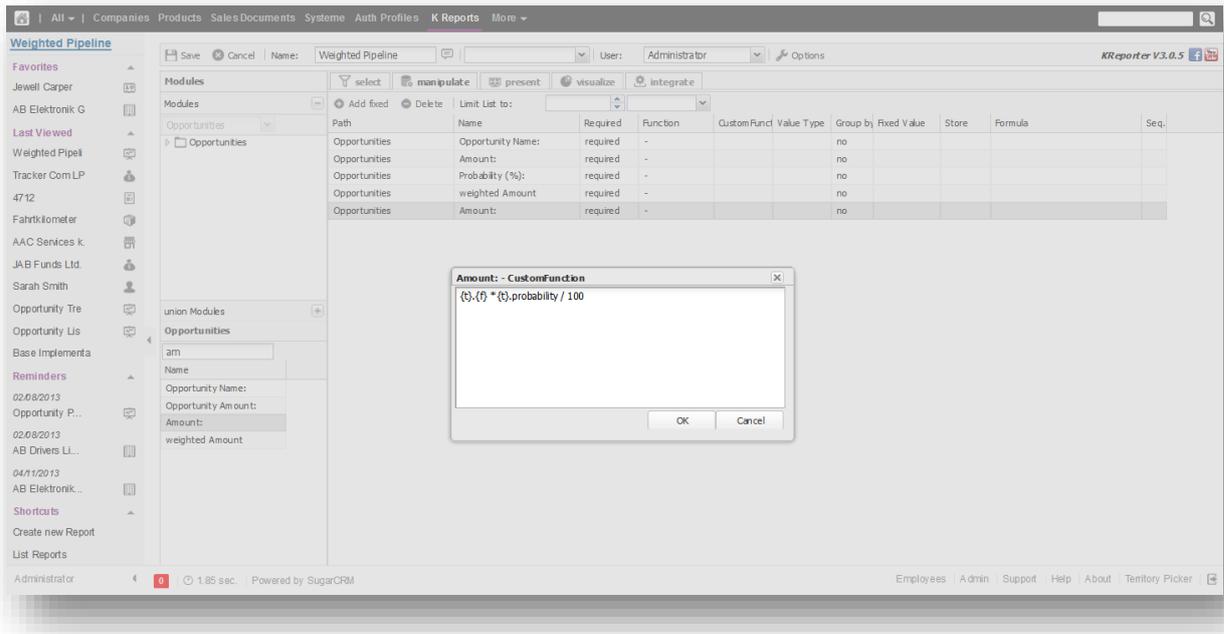
```
$dictionary['Account']['fields']['kpostalcode'] = array(
    'name' => 'kpostalcode',
    'vname' => 'LBL_KFZKZ',
    'type' => 'kreporter',
    'source' => 'nondb',
    'kreporttype' => 'varchar',
    'eval' => array(
        'presentation' => array(
            'eval' => 'select group_concat(kpostalcode.kfzkz) from kpostalcodees where
                kpostalcodees.postalcode = {t}.billing_address_postalcode',
        ),
        'selection' => array(
            'equals' => 'exists(select * from kpostalcodees where kfzkz = \''{p1}\'' and
                postalcode = {t}.billing_address_postalcode)',
            'between' => 'exists(select * from kpostalcodees where kfzkz >= \''{p1}\'' and kfzkz
                <= \''{p2}\'' and postalcode = {t}.billing_address_postalcode)',
            'starts' => 'exists(select * from kpostalcodees where kfzkz like \''{p1}\'' and
                postalcode = {t}.billing_address_postalcode)',
            'contains' => 'exists(select * from kpostalcodees where kfzkz like \''{p1}%'\'' and
                postalcode = {t}.billing_address_postalcode)'
        )
    )
);
```

The above defines an additional field that in the selection reads records from a separate table in the database and concatenates values. In the selection screen this field will offer the operators 'ignore' (this is always offered) as well as '=', 'between', 'starts' and 'contains'

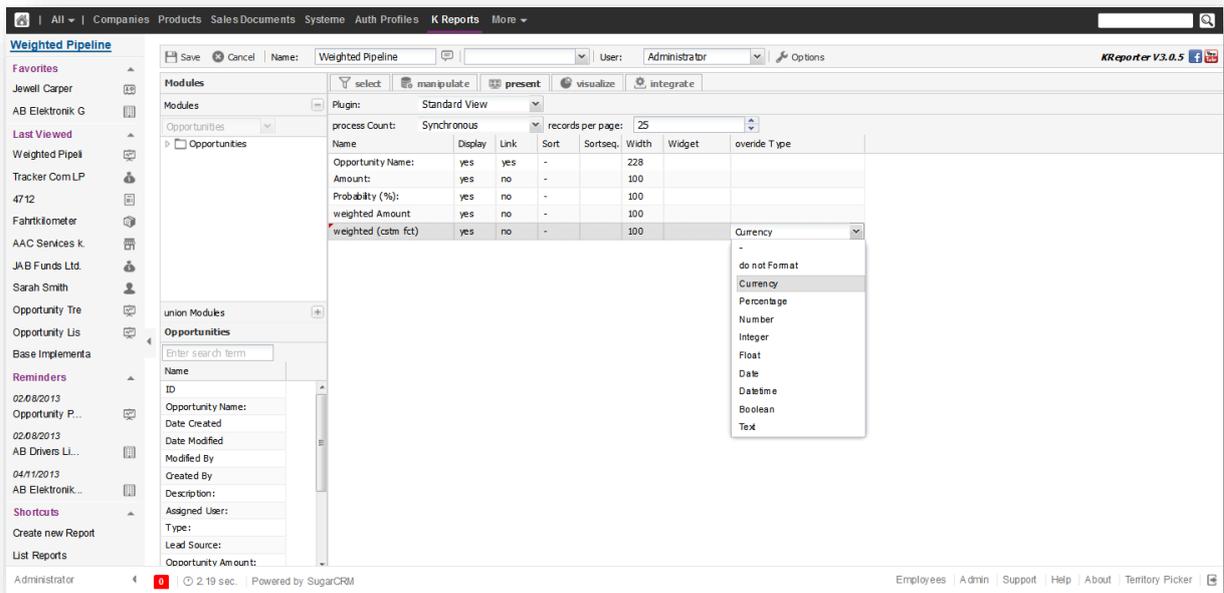
Functions & Custom Functions

In the Manipulation View of the Reporter data can (as the title indicates) be manipulated. There are standard function (like sum, count, average, ...) that can be selected from a dropdown and mainly apply to groupings being done. To add a custom function there is a separate field that when you click into it a popup opens and the custom function can be edited.

Following the example from above we could achieve the same with a custom function here. Drag the field amount over to the view and edit the custom function.



We also enter the custom function as follows: $\{t\}.\{ff\} * \{t\}.\text{probability} / 100$. Also rename the field to *weighted (cstm fct)* and on the presentation tab set the override Type to Currency.



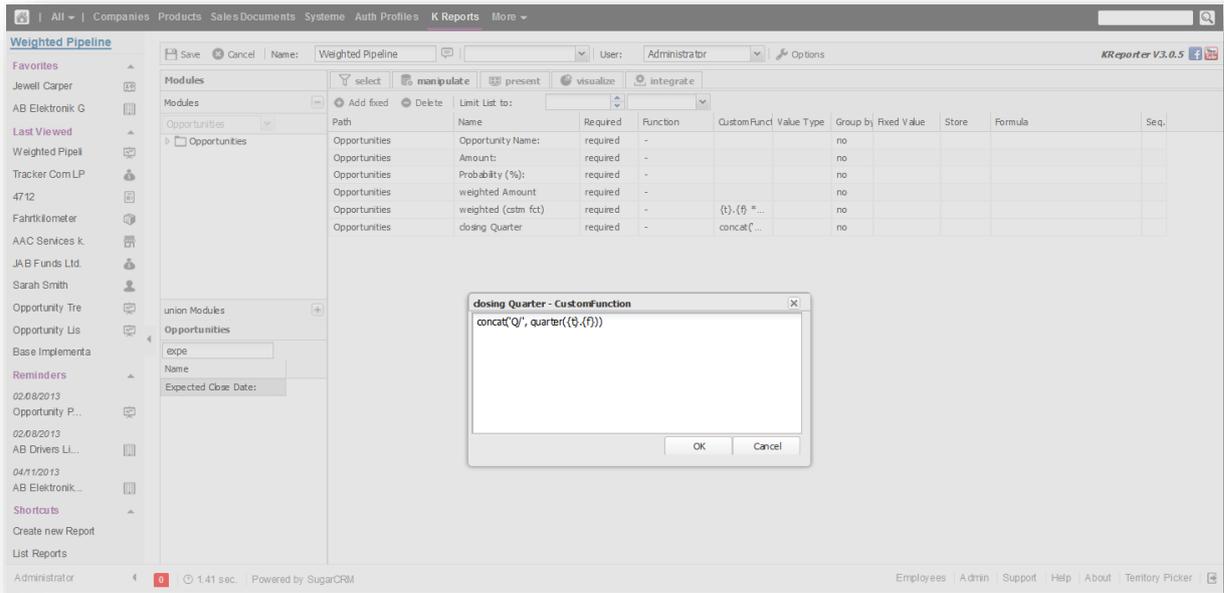
The latter one ensures that the field is rendered as currency again. Whenever you set a custom function KReporter will no longer consider the type of the original field for the rendering since the transformation most likely will change the format.

The variables you can use are similar to the KReporter field with {t} and {tc}. In addition you can use {f} to reference the field you did drag over. You could of course specify this by the name but this might cause issues in some union cases where the {f} comes in handy.

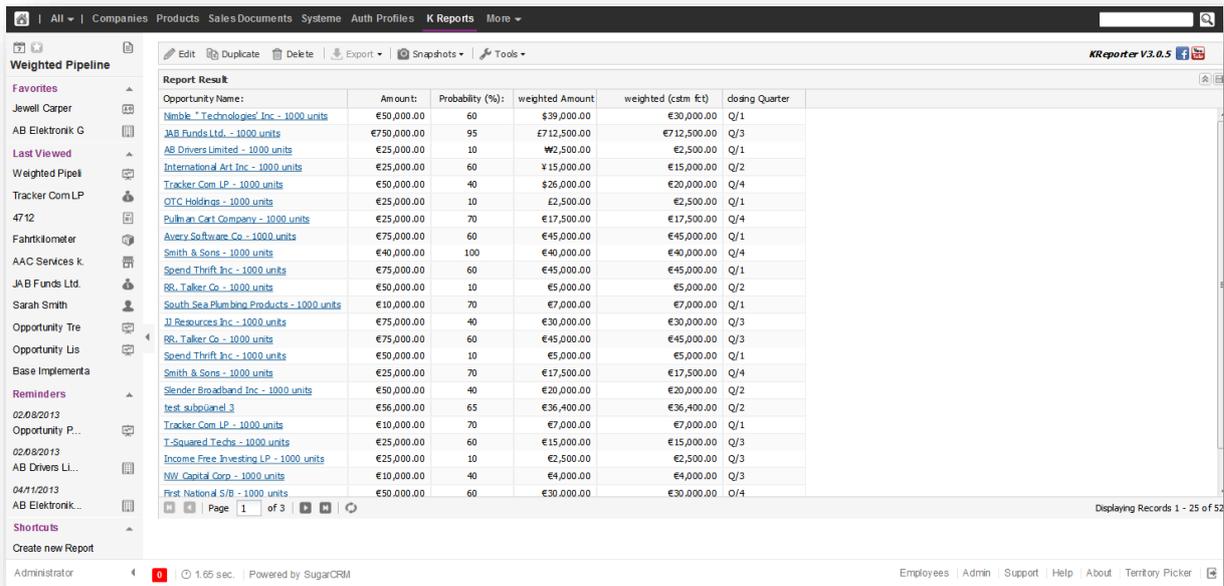
If we run the report this looks as follows. The difference in the currency symbol is attributed to the fact that the amount field is not linked to a currency_id.

Opportunity Name:	Amount:	Probability (%)	weighted Amount	weighted (cstm fc)
Nimble Technologies Inc - 1000 units	€50,000.00	60	\$39,000.00	€30,000.00
JAB Funds Ltd. - 1000 units	€750,000.00	95	£712,500.00	€712,500.00
AB Drivers Limited - 1000 units	€25,000.00	10	¥2,500.00	€2,500.00
International Art Inc - 1000 units	€25,000.00	60	¥15,000.00	€15,000.00
Tracker Com LP - 1000 units	€50,000.00	40	\$26,000.00	€20,000.00
OTC Holdings - 1000 units	€25,000.00	10	£2,500.00	€2,500.00
Pullman Cart Company - 1000 units	€25,000.00	70	£17,500.00	€17,500.00
Avery Software Co - 1000 units	€75,000.00	60	€45,000.00	€45,000.00
Smith & Sons - 1000 units	€40,000.00	100	€40,000.00	€40,000.00
Spend Thrift Inc - 1000 units	€75,000.00	60	€45,000.00	€45,000.00
RR Talker Co - 1000 units	€50,000.00	10	€5,000.00	€5,000.00
South Sea Plumbing Products - 1000 units	€10,000.00	70	€7,000.00	€7,000.00
JJ Resources Inc - 1000 units	€75,000.00	40	€30,000.00	€30,000.00
RR Talker Co - 1000 units	€75,000.00	60	€45,000.00	€45,000.00
Spend Thrift Inc - 1000 units	€50,000.00	10	€5,000.00	€5,000.00
Smith & Sons - 1000 units	€25,000.00	70	€17,500.00	€17,500.00
Slender Broadband Inc - 1000 units	€50,000.00	40	€20,000.00	€20,000.00
test subpanel 3	€56,000.00	65	€36,400.00	€36,400.00
Tracker Com LP - 1000 units	€10,000.00	70	€7,000.00	€7,000.00
T-Squared Techs - 1000 units	€25,000.00	60	€15,000.00	€15,000.00
Income Free Investing LP - 1000 units	€25,000.00	10	€2,500.00	€2,500.00
NW Capital Corp - 1000 units	€10,000.00	40	€4,000.00	€4,000.00
First National S/B - 1000 units	€50,000.00	60	€30,000.00	€30,000.00

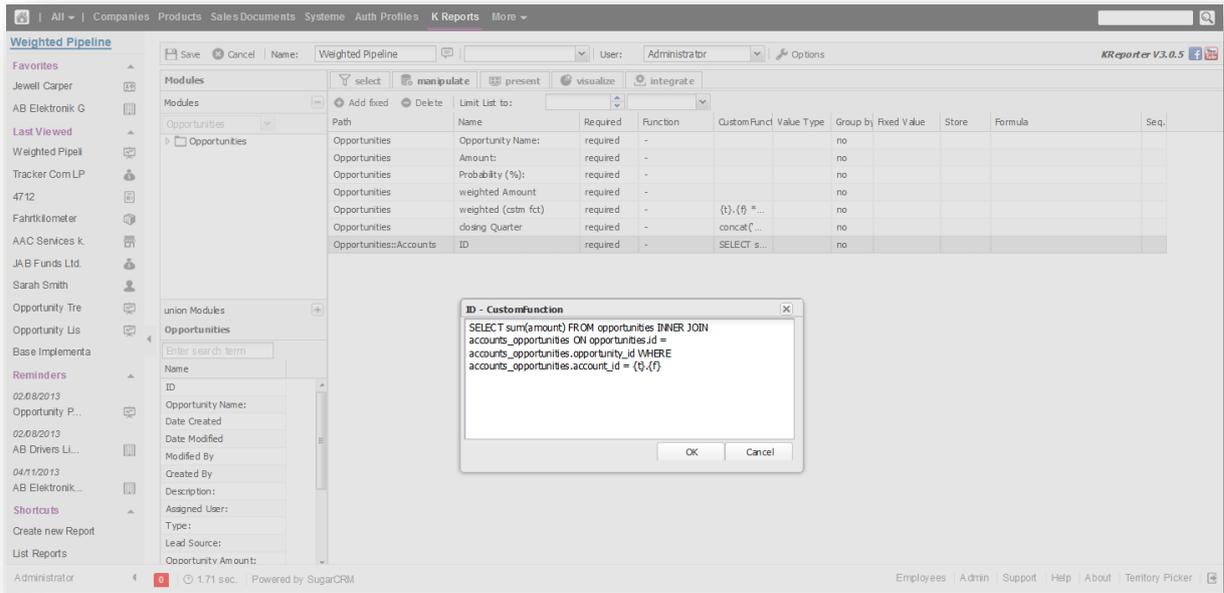
Another typical example to use the custom function is for date manipulation. E.g. when you want to convert the expected close date of the opportunity to a closing Quarter. To do that edit the report drag the expected close date field to the fields in the manipulation tab and set a custom function as follows: `'concat('Q', quarter({t}.{f}))'`, rename the field to Closing quarter.



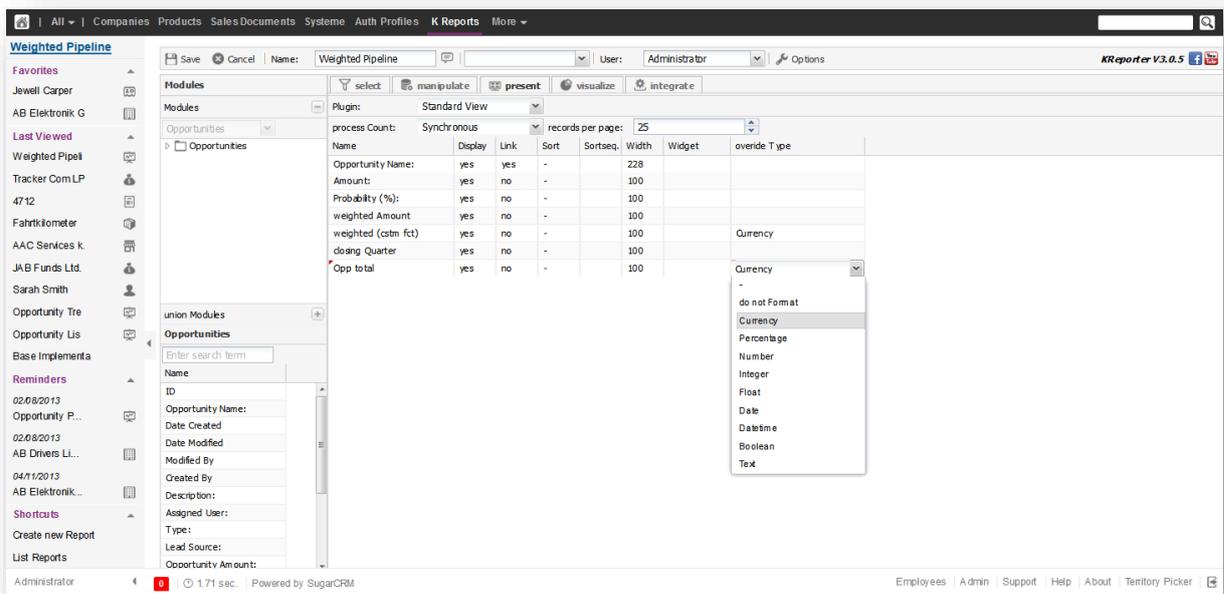
The Result looks as follows:



Another useful thing you can do is to do subselects. Assume we want to additionally have the total amount of all opportunities for that customer as a sum in the list. To achieve that we add the ID field from the accounts link in the opportunity and edit the custom function as follows: `SELECT sum(amount) FROM opportunities INNER JOIN accounts_opportunities ON accounts_opportunities.opportunity_id = opportunities.id WHERE accounts_opportunities.account_id = {t}.f}`.



Rename the field to *Opp total* and set the renderer again to Currency in the Presentation Tab.



Save the report and the Result looks as follows. The total of all opportunities for the account are summarized and then presented in the list.

Weighted Pipeline

Report Result

Opportunity Name:	Amount:	Probability (%)	weighted Amount	weighted (catm fc)	osing Quarter	Opp total
Nimble "Technologies" Inc - 1000 units	€50,000.00	60	\$39,000.00	€30,000.00	Q/1	€65,000.00
JAB Funds Ltd. - 1000 units	€750,000.00	95	£712,500.00	€712,500.00	Q/3	€750,000.00
AB Drivers Limited - 1000 units	€25,000.00	10	¥2,500.00	€2,500.00	Q/1	€25,000.00
International Art Inc - 1000 units	€25,000.00	60	¥15,000.00	€15,000.00	Q/2	€25,000.00
Tracker Com LP - 1000 units	€50,000.00	40	\$26,000.00	€20,000.00	Q/4	€65,000.00
GTC Holdings - 1000 units	€25,000.00	10	€2,500.00	€2,500.00	Q/1	€25,000.00
Pullman Cart Company - 1000 units	€25,000.00	70	€17,500.00	€17,500.00	Q/4	€25,000.00
Avery Software Co - 1000 units	€75,000.00	60	€45,000.00	€45,000.00	Q/1	€75,000.00
Smith & Sons - 1000 units	€40,000.00	100	€40,000.00	€40,000.00	Q/4	€40,000.00
Spend Thrift Inc - 1000 units	€75,000.00	60	€45,000.00	€45,000.00	Q/1	€75,000.00
RR Talker Co - 1000 units	€50,000.00	10	€5,000.00	€5,000.00	Q/2	€50,000.00
South Sea Plumbing Products - 1000 units	€10,000.00	70	€7,000.00	€7,000.00	Q/1	€10,000.00
JJ Resources Inc - 1000 units	€75,000.00	40	€30,000.00	€30,000.00	Q/3	€75,000.00
RR Talker Co - 1000 units	€75,000.00	60	€45,000.00	€45,000.00	Q/3	€75,000.00
Spend Thrift Inc - 1000 units	€50,000.00	10	€5,000.00	€5,000.00	Q/1	€50,000.00
Smith & Sons - 1000 units	€25,000.00	70	€17,500.00	€17,500.00	Q/4	€25,000.00
Slender Broadband Inc - 1000 units	€50,000.00	40	€20,000.00	€20,000.00	Q/2	€50,000.00
best subdomain 3	€56,000.00	65	€36,400.00	€36,400.00	Q/2	€66,000.00
Tracker Com LP - 1000 units	€10,000.00	70	€7,000.00	€7,000.00	Q/1	€10,000.00
T-Squared Techs - 1000 units	€25,000.00	60	€15,000.00	€15,000.00	Q/3	€25,000.00
Income Free Investing LP - 1000 units	€25,000.00	10	€2,500.00	€2,500.00	Q/3	€25,000.00
NW Capital Corp - 1000 units	€10,000.00	40	€4,000.00	€4,000.00	Q/3	€10,000.00
First National S/B - 1000 units	€50,000.00	60	€30,000.00	€30,000.00	Q/4	€50,000.00

Page 1 of 3

Displaying Records 1 - 25 of 52

Administrator | 1.96 sec. | Powered by SugarCRM | Employees | Admin | Support | Help | About | Territory Picker

You can literally use all kinds of statement the SQL database can accept. But be aware that this might impact report performance and also that if you mistype there might be SQL errors and no results at all.

Formulas

While KReporter Fields and Custom functions are processed during the select statement the Formulas are processed on the result record per each row in the PHP layer. This offers further capabilities but also needs to be done carefully since this has a further impact on the performance.

If we want to solve the issue of getting a weighted Pipeline via Formulas we can do this as well as follows. In the first step drag over the amount field in the manipulation view for another time. This time we want to make use of the fields Store in the Grid. For the probability we type *prob* in that field, for the amount *amt*.

Name	Required	Function	CustomFunc	Value Type	Group by	Fixed Value	Store	Formula	Seq.
Opportunities: Opportunity Name:	required	-			no				
Opportunities: Amount:	required	-			no				
Opportunities: Probability (%):	required	-			no		prob		
Opportunities: weighted Amount	required	-			no				
Opportunities: weighted (cstm fct)	required	-	{}, {} *		no				
Opportunities: closing Quarter	required	-	concat (no				
Opportunities:Accounts: Opp total	required	-	SELECT s...		no				
Opportunities: Amount:	required	-			no		amt		

You can use any name for a variable but generally do not use special chars and no spaces to make it easier for yourself. This options puts the values into a store where you can afterwards use it in the formula. For the Formula we type the following $\{amt\} * \{prob\} / 100$. Any value in {} refers to a variable that is defined as a store. Keeping that consistent is your responsibility. The language used is interpreted as PHP so you can use all functions PHP offers.

We again rename the field to *wght (formula)* and set the renderer to Currency. The result looks as follows:

we wrap a `` tag around the value with the color style set to red. If not we just return the percentage. In any case we calculate the percentage and round it to 2 digits using the round function of PHP and concatenate it with the `'` With a % sign.

The other important thing to mention here is the third field which is the Sequence in which the formulas are evaluated. In our case we first calculate the weighted amount and then the percentage. If the weighted sequence is not set the formulas will be evaluated in whatever order but cannot be predicted. Since we need the weighted amount to be calculated first this is important that we ensure the sequence is followed.

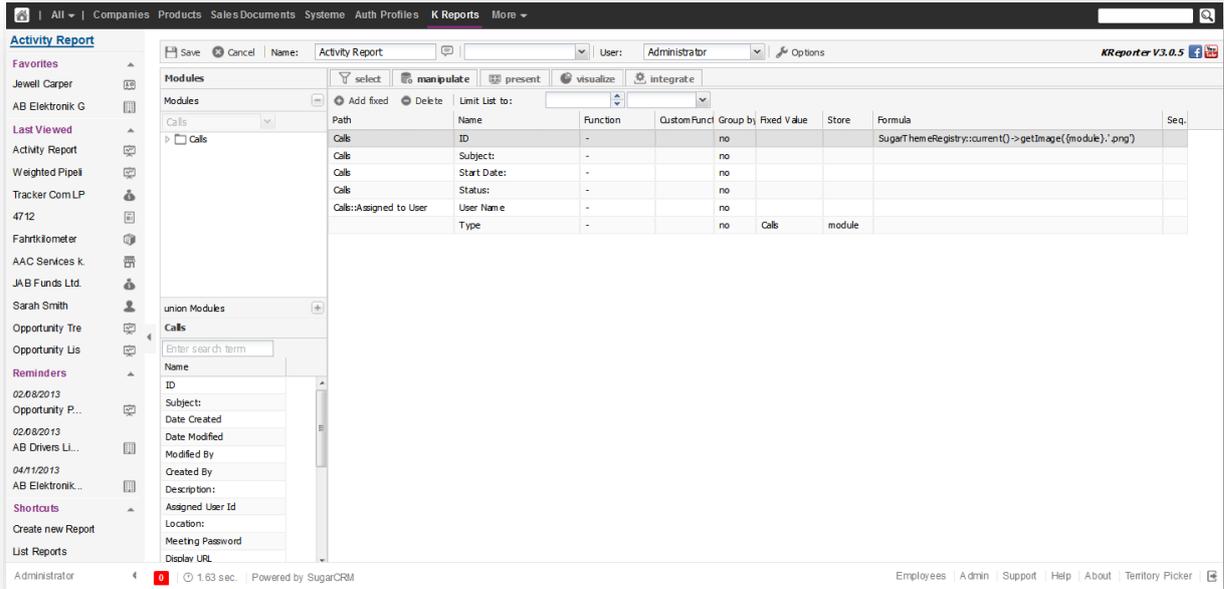
Save the report and the result looks as follows:

Opportunity Name:	Amount:	Probability (%)	weighted Amount	weighted (custom formula)	closing Quarter	Opp total	wght (formula)	% of total
Nimble "Technologies" Inc - 1000 units	€50,000.00	60	\$39,000.00	€30,000.00	Q/1	€65,000.00	€30,000.00	46.15%
JAB Funds Ltd. - 1000 units	€750,000.00	95	€712,500.00	€712,500.00	Q/3	€750,000.00	€712,500.00	95%
AB Divers Limited - 1000 units	€25,000.00	10	€2,500.00	€2,500.00	Q/1	€25,000.00	€2,500.00	10%
International Art Inc - 1000 units	€25,000.00	60	€15,000.00	€15,000.00	Q/2	€25,000.00	€15,000.00	60%
Tracker Com LP - 1000 units	€50,000.00	40	€20,000.00	€20,000.00	Q/4	€65,000.00	€20,000.00	30.77%
OTC Holdings - 1000 units	€25,000.00	10	€2,500.00	€2,500.00	Q/1	€25,000.00	€2,500.00	10%
Pullman Cart Company - 1000 units	€25,000.00	70	€17,500.00	€17,500.00	Q/4	€25,000.00	€17,500.00	70%
Avery Software Co - 1000 units	€75,000.00	60	€45,000.00	€45,000.00	Q/1	€75,000.00	€45,000.00	60%
Smith & Sons - 1000 units	€40,000.00	100	€40,000.00	€40,000.00	Q/4	€40,000.00	€40,000.00	100%
Spend Thrift Inc - 1000 units	€75,000.00	60	€45,000.00	€45,000.00	Q/1	€75,000.00	€45,000.00	60%
RR Talker Co - 1000 units	€50,000.00	10	€5,000.00	€5,000.00	Q/2	€50,000.00	€5,000.00	10%
South Sea Plumbing Products - 1000 units	€10,000.00	70	€7,000.00	€7,000.00	Q/1	€10,000.00	€7,000.00	70%
JJ Resources Inc - 1000 units	€75,000.00	40	€30,000.00	€30,000.00	Q/3	€75,000.00	€30,000.00	40%
RR Talker Co - 1000 units	€75,000.00	60	€45,000.00	€45,000.00	Q/3	€75,000.00	€45,000.00	60%
Spend Thrift Inc - 1000 units	€50,000.00	10	€5,000.00	€5,000.00	Q/1	€50,000.00	€5,000.00	10%
Smith & Sons - 1000 units	€25,000.00	70	€17,500.00	€17,500.00	Q/4	€25,000.00	€17,500.00	70%
Slender Broadband Inc - 1000 units	€50,000.00	40	€20,000.00	€20,000.00	Q/2	€50,000.00	€20,000.00	40%
test_sub@netel_3	€56,000.00	65	€36,400.00	€36,400.00	Q/2	€66,000.00	€36,400.00	55.15%
Tracker Com LP - 1000 units	€10,000.00	70	€7,000.00	€7,000.00	Q/1	€10,000.00	€7,000.00	70%
T-Squared Techno - 1000 units	€25,000.00	60	€15,000.00	€15,000.00	Q/3	€25,000.00	€15,000.00	60%
Income Free Investing LP - 1000 units	€25,000.00	10	€2,500.00	€2,500.00	Q/3	€25,000.00	€2,500.00	10%
NW Capital Corp - 1000 units	€10,000.00	40	€4,000.00	€4,000.00	Q/3	€10,000.00	€4,000.00	40%
First National S/B - 1000 units	€50,000.00	60	€30,000.00	€30,000.00	Q/4	€50,000.00	€30,000.00	60%

As wanted the percentage is shown and values higher than 50% are shown in red.

As mentioned this calls PHP functions and you can also call Sugar internal functions from here that are globally and statically available. A good example is to embed an image. In the following example we do a union report on activities and want to show a module specific icon. For that we add a fixed field for the module to the Union Join, put that in the Store *module* like in the example shown for the Calls and then use the Formula `SugarThemeRegistry::current()->getImage({module}. 'png')`.

The manipulate Tab looks as follows:



The Result looks as follows. For each record an image is rendered depending on the type of record. In our case Calls, Meetings and Tasks. And since the field we used the formula on also was set as ID with a link the images are also linked.

