Workflow Designer

Special Actions

Call Webservice

Parameters from top to down:

This task could do a background call to a configured URL and transfer multiple configured parameters.

URL	Define the URL of the Webservice
Method	Transfer parameters as **GET** or **POSTGET** means the Parameters will be send in the url like this: “https:/ /?parameter1=value1&pareter2=value2&…”**POST** means the Parameter won’t be send in the URL, but are also added to the request. This allows bigger amount of data, but must be supported by the webservice.
Response Format	Is the Response Type recognized as plain text or converted as JSON Will automatically save JSON result into $env response
Response $env	Store response within this $env variable
Content-Type	Do you want to transfer Parameters as **application/json** or **x-www-form-urlencoded**
Parameter	Define the single parameters you want to send
nested Parameters	When you send POST parameters, you can use a conversion of Parameters like param1[a], param1[b], param1[c] into a JSON compatible array
Headers	add additional Headers, for example to provide additional authentication tokens

Changelog

Date	Update
2016-12-06	Fix issue when using “call webservice” together with redirection function
2022-01-06	Implement option to send JSON POST body as nested array

Decision table

When you know DMN from BPMN, then you also know how to use the Decision table. It is similar to a Switch case with multiple conditions and multiple actions per condition.

You can define conditions, called decision and Outputs, which will be set, when a decision chain is true.

The table have two main parts. The blue left part to define decisions and the green right part to define Output actions. Every part can have multiple columns to allow multiple decisions/output actions for 1 row.
A row is a set of multiple decisions and a set of multiple output actions.

Every input field with a blue template Icon, will provide ALL template functions of Workflow Designer. So you can use powerful decisions by an easy and efficient configuration.

Decisions

Decision columns contain a single header input field, where you can define the Input, which will be used to check against the single rows.
Every decision input within the data rows can check “is equal”, “contain”, “expression returns true”. **You should only use the last one, when you know, how to use Expressions, and have PHP development skills.**

Output

Output columns have two header fields, which define the type of output you write to, when decisions are true.
You can select to write into an **$env** Variable or directly into record field of current main record.

The rows are checked from top to down and stops after the first row, with decisions are completely true.

Example

In the example image, there are two decision columns and two output columns.
The first column is checked against a **$env** Value. The second one checks against the field value of **website** field.

So execution is done in the following way:

In first row, decisions are checked. If $env value is equal to “q”, then website field will checked if it contains “stefanwarnat”. If no, next row is checked.
In second row, decisions are checked. If $env value is equal to “q2”, then website field will checked, if it contains “mustermann”. If no, next row is checked.
In third row, decisions are checked. If $env value is equal to “q4”, then expression is checked. The result of the expression won’t be checked against website field. You directly set the result of

decision within the expression.
Values like true, “true”, “1”, “on”, “yes” will interpreted as true. Everything else is false.

Let’s define the $env Variable to “q2” and website field to “mustermann.com”.
Then first row is going within first decision false and second row will result into “true” for all decisions and is our “Output” for this execution. Every next rows are ignored.

Now the $env[“var”] variable is set to “Multimedia $accountname”, where $accountname is replaced by the fieldvalue of “Account name” field.
Also the “Versand Stadt” field is set to “Value B”. You don’t need to worry about anything, when this field is set. It behave like the “set values” task.

Execute & Store Mysql Query

Only use this task if you could work with MySQL Querys!

This task executes a query against the database and save the first row of the result into an environmental variable, which could be used by other tasks.

An equal result you could get with the function wf_dbquery inside custom expressions, which does exactly the same.

Example

You execute a query which get a row with the columns “colA”, “colB”, “colC”, “colD” and you save this to the environmental variable *mysqlresult*.
You could use these values by $env[“mysqlresult”][“colA”] or $env[“mysqlresult”][“colB”] …

Generate Recordlist

This block only works in combination with a “global Search” or a “exists related record” action, because it could transform the results of these blocks to a html list you could include in E-Mails or into a PDFMaker Template.

The configuration is easy.

Records are in Module	Here you must select the same modules, like you did in he “global Search” or “exists related record”
Environment ID	Here you must enter the same ID, like you did in he “global Search” or “exists related record”

At least add fields you want to have in the list.
Different to other field selects, you could add a value “Link to the Record”, which should be self-explaining.

Edit

Integration of PDFMaker

You get in the configuration window the code you could add into a PDFMaker Template.

There are one additional limitations for this.
This code only works during this workflow execution.
(The Template didn’t work externally or in other workflows, without a “generate recordlist” block)

Mail accessed

This block could check if an email you send to a record, was already opened.
It use the AccessTracker build in with vtigerCRM.

You could create a filter to select one/multiple mails, which will be checked. Only one of the mails needs to be accessed.

The field **mails related to this record** could be used to check mails, related do another record. If you leave this field blank, the current record will be used.
Also you could limit the number of mails, which will be checked. Mails which come later in the sort order, will be ignored.

This result is not very reliable, because lot’s of mail clients block images by default. In this case the

Opening will not be recognized.

Example

If you send an email to record 819, which subject “Are you interested?”, you could set the record id to 819 or a variable, which contain this and create a filter “**Subject** is equal ‘Are you interested?’ “. If you only want to check the first mail you send, set Sort order to “Created Time ASC” and set the amount of mails to “1”. Then you check, if customer had already opened this mail.

MySQL Check

Only use this task if you could work with MySQL Querys! You could destroy your database!

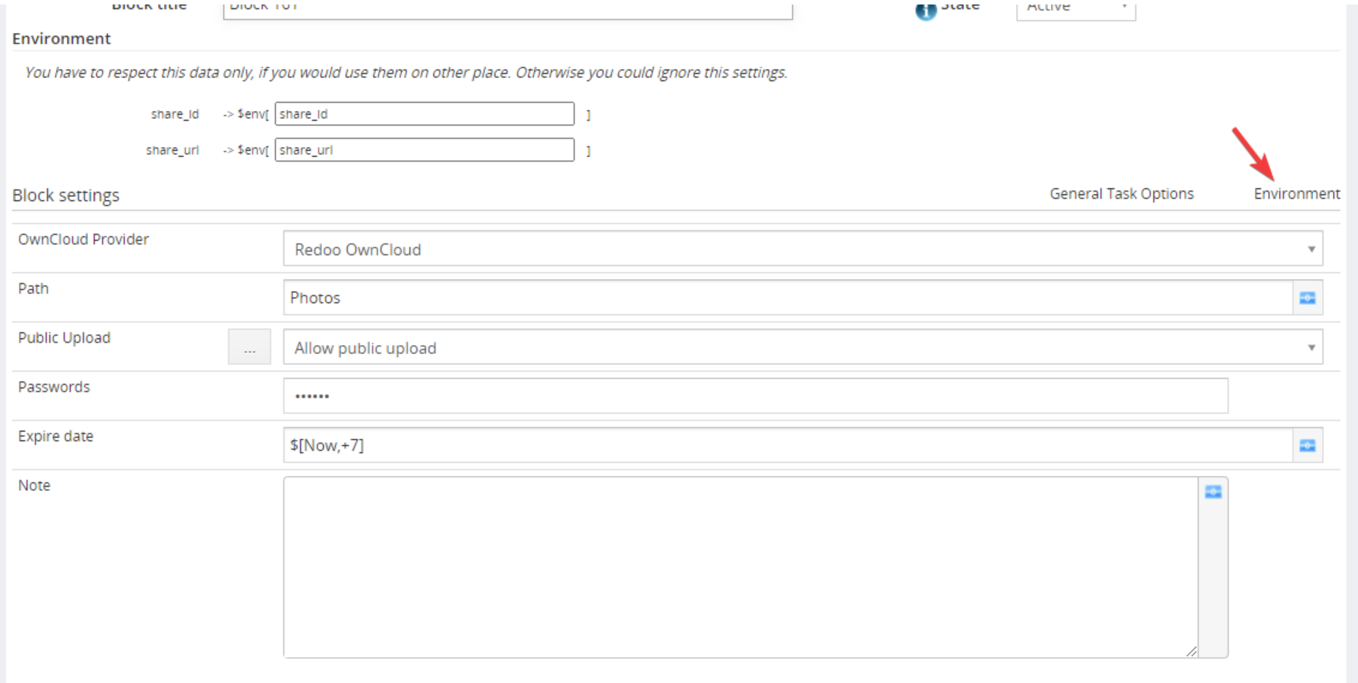
This block allows you to execute a Query directly to the database.
You could also use Update and INSERT Queries. But only to the vtiger database.

If you use a Select Query, the number of result rows could be used to select an output.
For this you could enter the number of results the query should return at least to return “true”.

Owncloud – Create Share

This task allows you to create a Share automatically into your OwnCloud.

The configuration contains the values a public share can have:

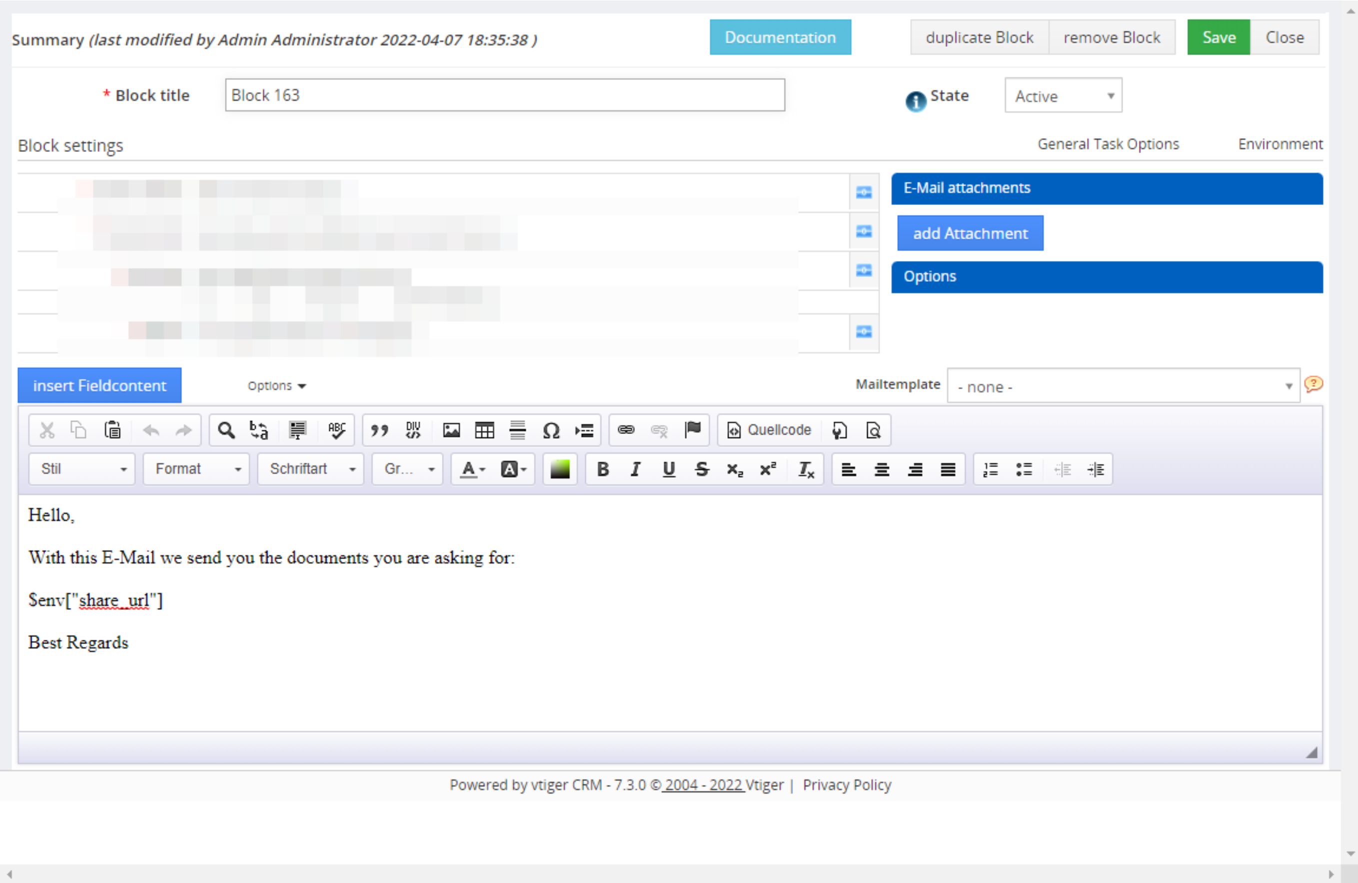


At first you need to create an OwnCloud Provider within the Provider Manager.
You can select the created Provider within this task.

Set / Generate Path you want to share and define several optional configurations from OwnCloud.
The Upload capability only works for folders.
You can use the features of Template input within all fields.

To get the share ID (When you also want to delete the share with the block “Owncloud – Delete Share”), or the Share URL, you need to configure an environment variable, where you want to write the values for later use.

In this example you can use the Share URL in E-Mails over variable **$env[“share_url”]**



OwnCloud – Delete Share

This task allows you to delete a previously created Share within your ownCloud.

Configuration needs the OwnCloud provide, which currently store the share and the share ID, you get from “create Share” action.

PDFMaker Integration

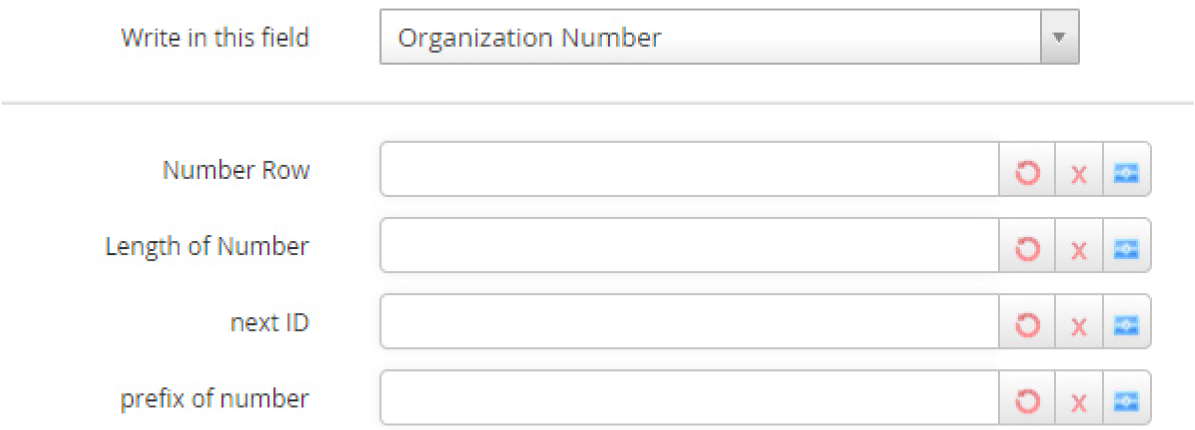
This block implement almost all feature the PDFMaker will provide.

1. At first choose if you want to create the PDF from the current record (default) or from all records, a configurable search will return
   If you will use the second option, the PDF you get will be merged together to one file, like the option to use PDFMaker in ListView
2. Choose the template you want to use

You could also choose multiple templates to merge multiple files to one file

3. Choose a filename
   (Optionally if you use the current record with only One Template. Important if you want to merge multiple PDF's to one. Otherwise you get only a generalfile)
4. Choose what should happen with this file (See File Actions)

Set custom Record Number



If you want to use different Record Numbers for Invoice/Accounts/Contacts based on some conditions, you could use this task.

This allows to set a increasing Number across differnet modules. **Every number will be used only one time!**

You could create multiple different "number sequences".

The field **Write in this field** will set the field, which should contain the sequence.
The field **Number Row** contains the name of the sequence. If you use two blocks with the same name, they will use the same sequence. You could choose the sequence also from a list of already existing sequences.

The next fields are only relevant if you want to create a new sequence. They are read only if you choose a existing one.
The field **Length of Number** contains the complete length of the string. If the next sequence Number will be shorter, it will filled with leading zeros.
The field **nextID** will contain the next used ID. (You could start for example on 1000)

The field **prefix of number** will set the prefix, which will set before the Number and written in the field, too.

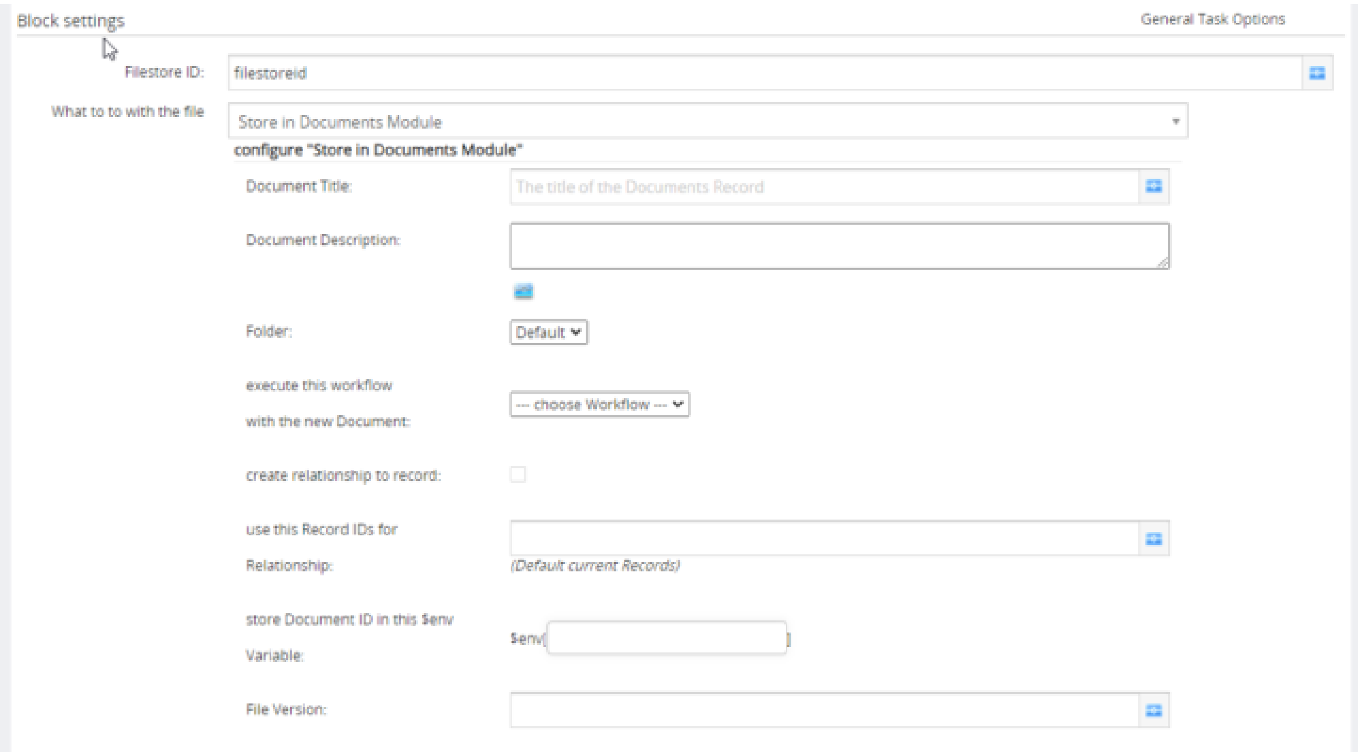
For example you could configure to create Record No like this:

Quote A	R0001
Quote B	R0002
Invoice A	R0003
SalesOrder A	R0004
Quote C	R0005
SalesOrder B	R0006

Filestore entry action

Read more about filestore on page about filestore.

This action allows to pick a file from temporarily filestore and apply an action on this file.
For example you can download them or store as Documents record.

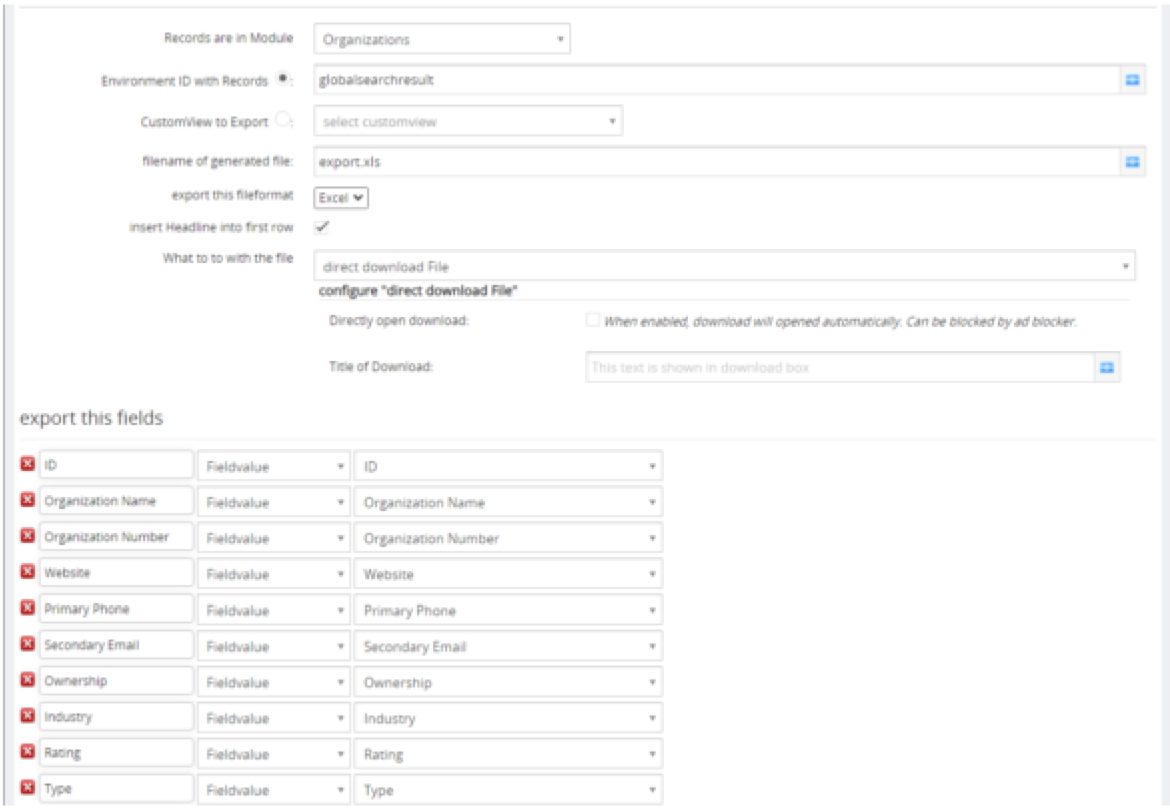


You must take care only about one topic. The Filestore ID, which defines a file, must be equal to the action, which creates a file within filestore.

export records

This action works best in combination with previous "global search" action.

Basically you can export records into CSV or XLS files with this block and configure the columns of export.
The Records, which should be exported, are taken from a default vTigerCRM List or a previous global search, where you set a unique ID in field "Store result Records".

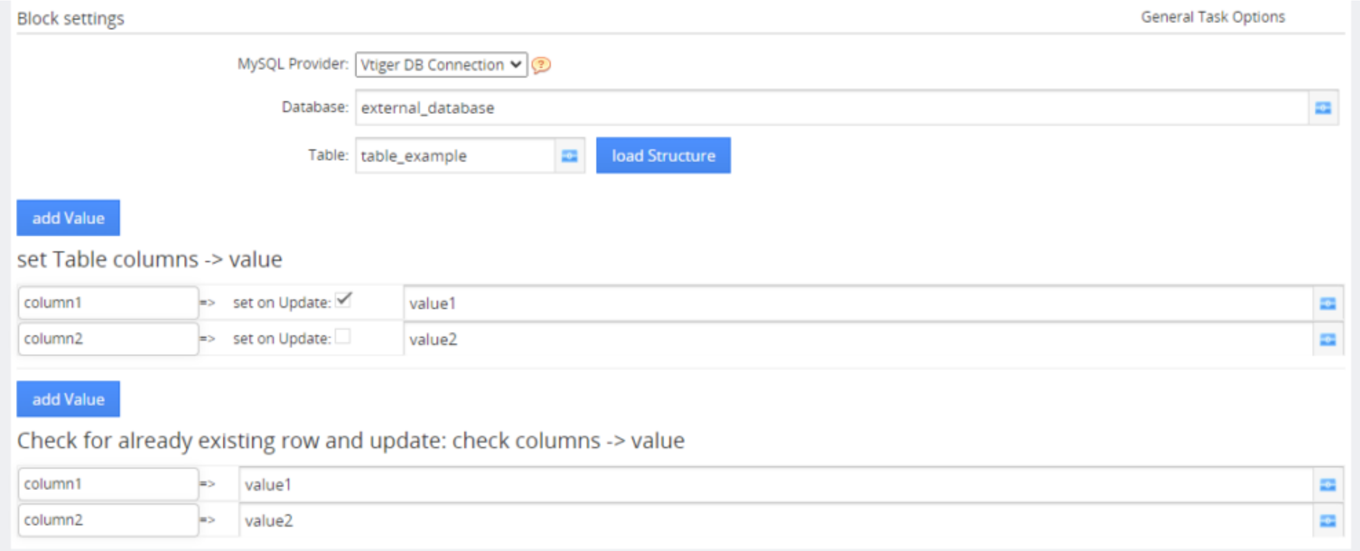


generate Campaign Status

This action generates a HTML table, which show the status of the given recordids over all campaigns.
You get a good overview, within which campaigns a lead, contact or account was involved.

Insert into MySQL table

This easy to use action allows you to insert values into mysql tables. You can access any MySQL provider, you configure within the Provider Manager.



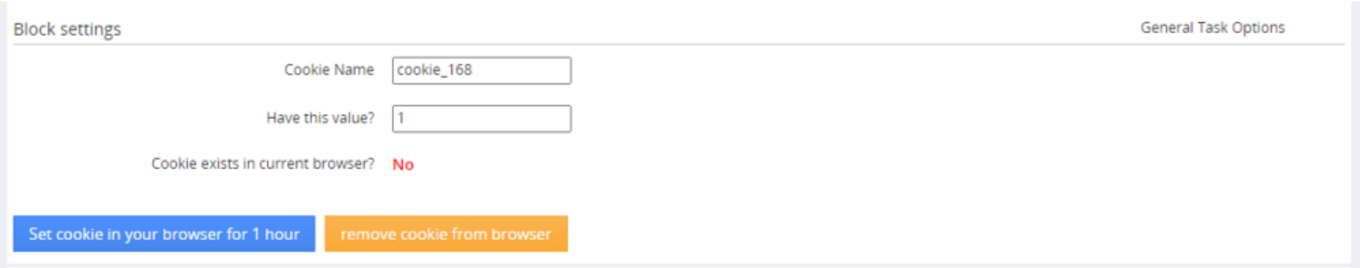
In first part you define the MySQL provider, database and table.

In second part, you set the columns, which should be inserted or, when checked, updated on duplicate.

In third part, you can optionally configure a duplication check. When there already is an row, which match this configuration in third part, then only the columns, you “set on update” will updated.

Check Cookie value

This actions allows you to use a path only, when a defined cookie is existing in current browser. Use this to test new paths only on your local machine, before the complete vTigerCRM is using it.



Print with Printnode

When you are using printnode to print documents over the internet, you can use this action to send documents to PrintNode.

Path with other record

This block allows to temporarily switch the context of a path. Use this with care, because it can confuse, if not every action is executed with same record.

But there are cases, when this is usefull.
The configuration within this block is a default record selection, which selects records, which are used to execute the path. This can one or more records. The path is executed once for every record.

prevent reload after finish

Sometimes it is required to not reload, when workflow is executed. Add this block into workflow and make use of that behaviour.

The action do not have any configuration.

create user

Because users are a special case in vTigerCRM, there is a special action to create them.
Configuration is equal to “set values” action.

set user values

Updating users is another special case. You can only update users with this block.

You cannot update the user password with this action!

Javascript response

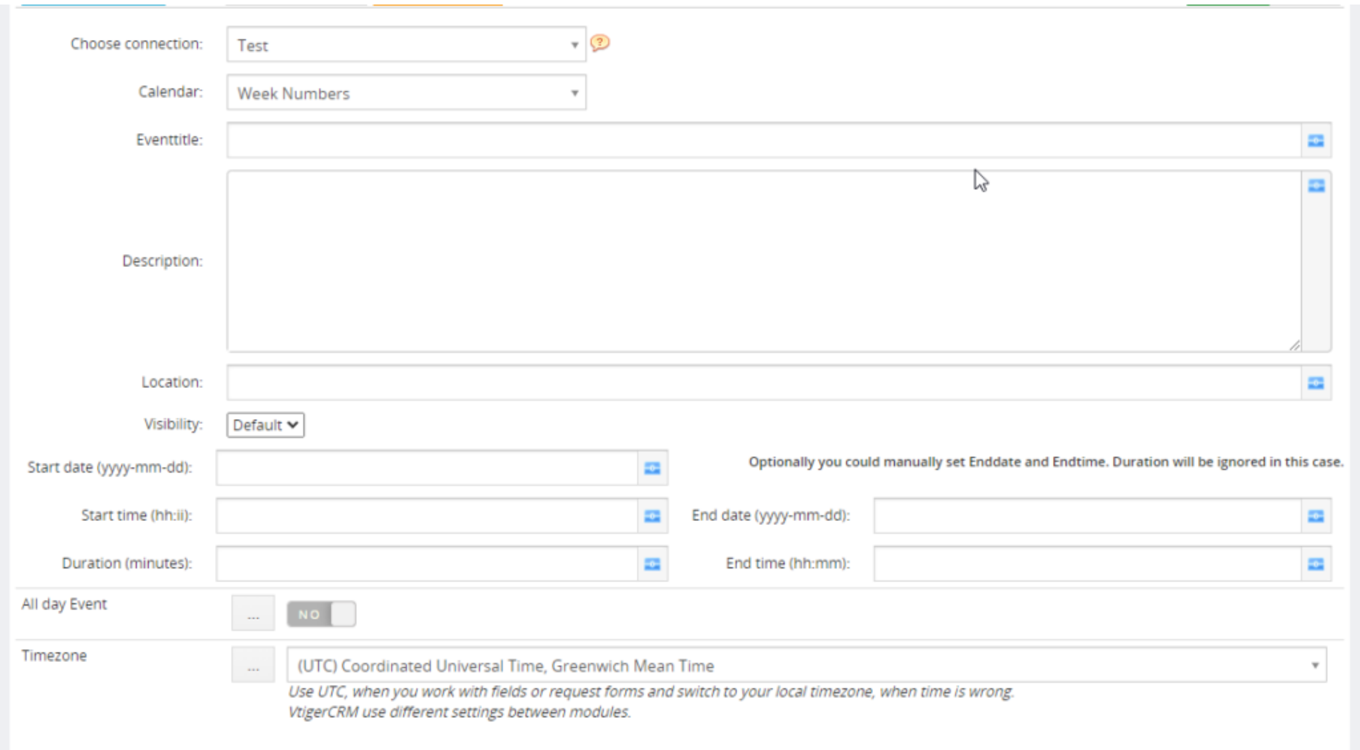
When you execute a workflow manually and want to have a special value from workflow, you can use this action to generate a custom JavaScript response of the execution request.

The configuration is a default key -> Value assignment with support for variables and expressions.

add to Google Calendar

This action allows to connect with a Google Calendar Provider to add events into Google Calendar.

You can select Calendar and set all information of the event.
You can set either duration of end time of the event, depending which values you can access.

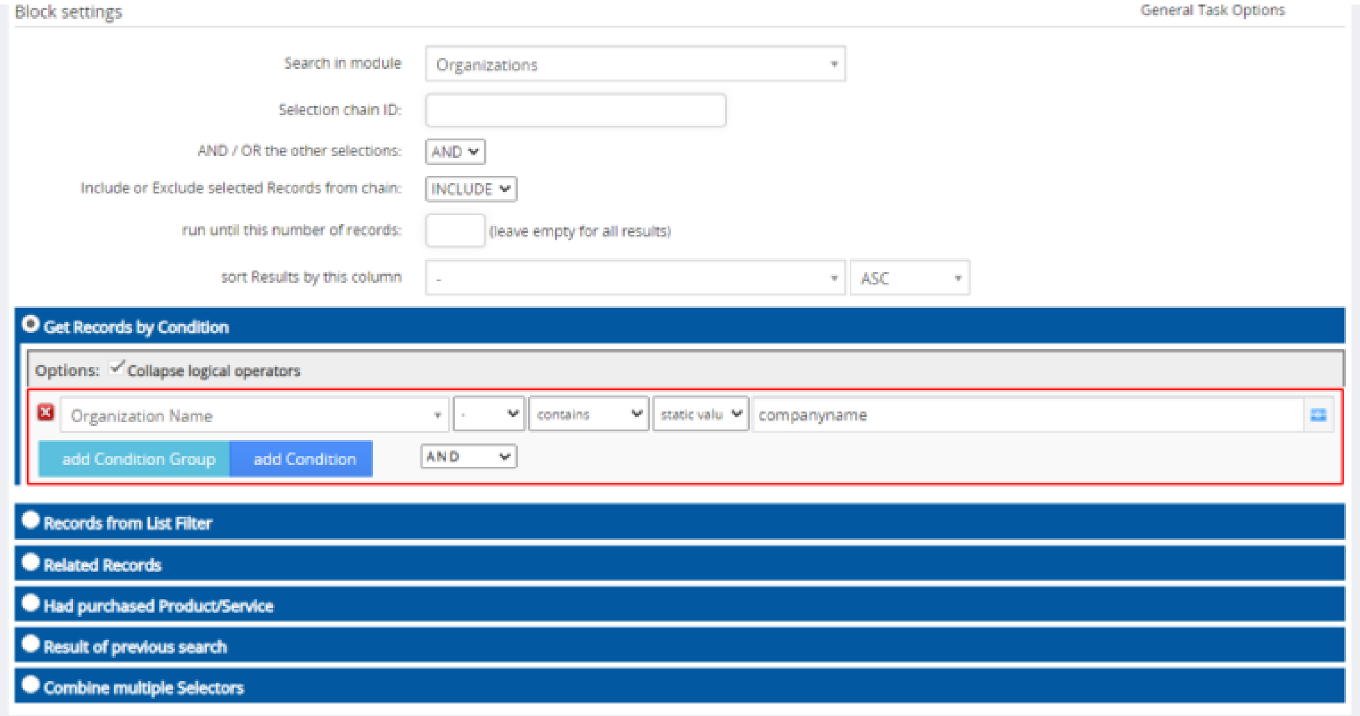


Complex Record selection

Whenever you need to select records, which cannot selected by a single selection, you can combine multiple record selections.
Or you select a big amount of records and want to exclude some record ids from the result, you can use this action.

All complex record selection actions with same “**Selection chain ID**” create a single resultset of record ids.
The records theoretically don’t even needs to be in the same module. But there are only very rare usecases, where you apply some action on records from different modules.

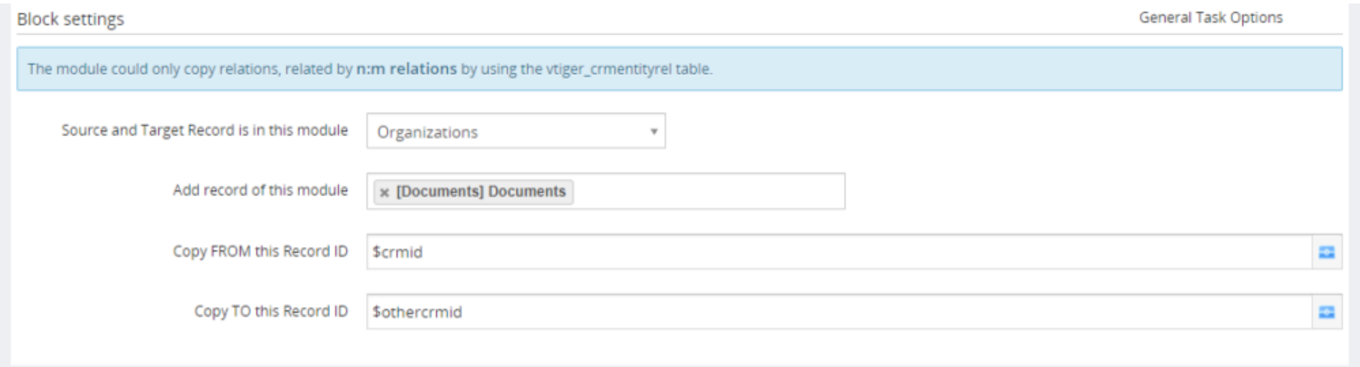
You can use the resultset in every action, which use the default record selection, by selecting the “**Combine multiple Selectors**” selection and insert the same “**Selection chain ID**“.



The single record selections can be combined by "AND" or "OR". When you select "AND" only these records are part of the resultset, which are covered by every complet record selection.

copy relation

This action copy all relations from one record to another one. Typical usecases are to clone all documents from a quote and attach them to a Salesorder.

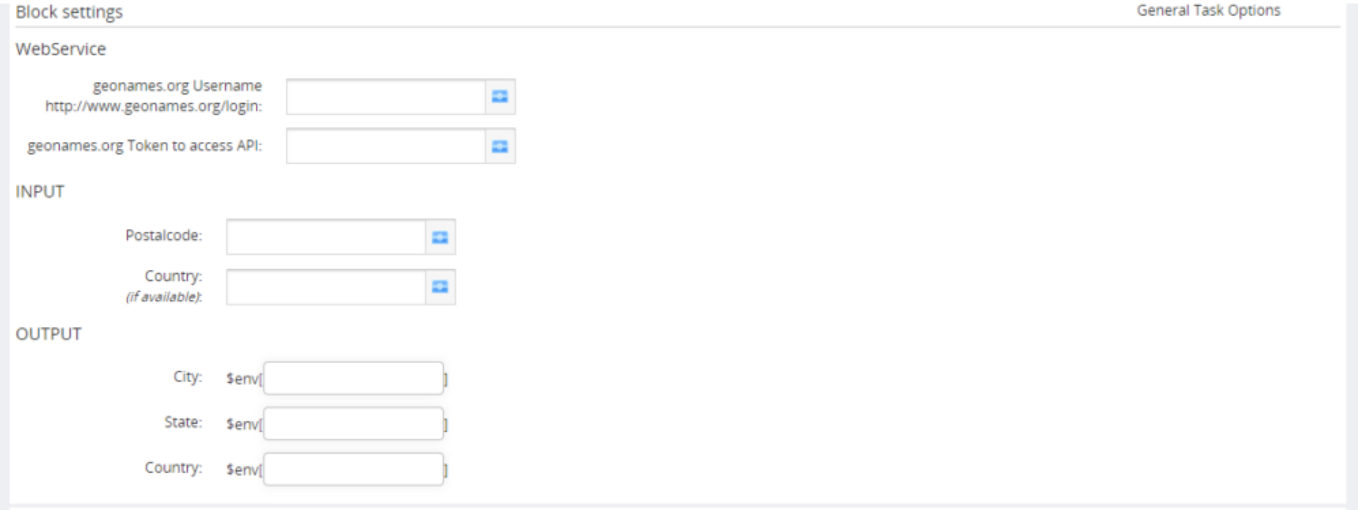


parse postalcode

Whenever you want to know state, country and city to a postalcode, you can use this action to connect with the API of http://www.geonames.org to determine the values.

Make sure you set a geonames.org account and access token, when defined.

The result is written into configured $env variables.

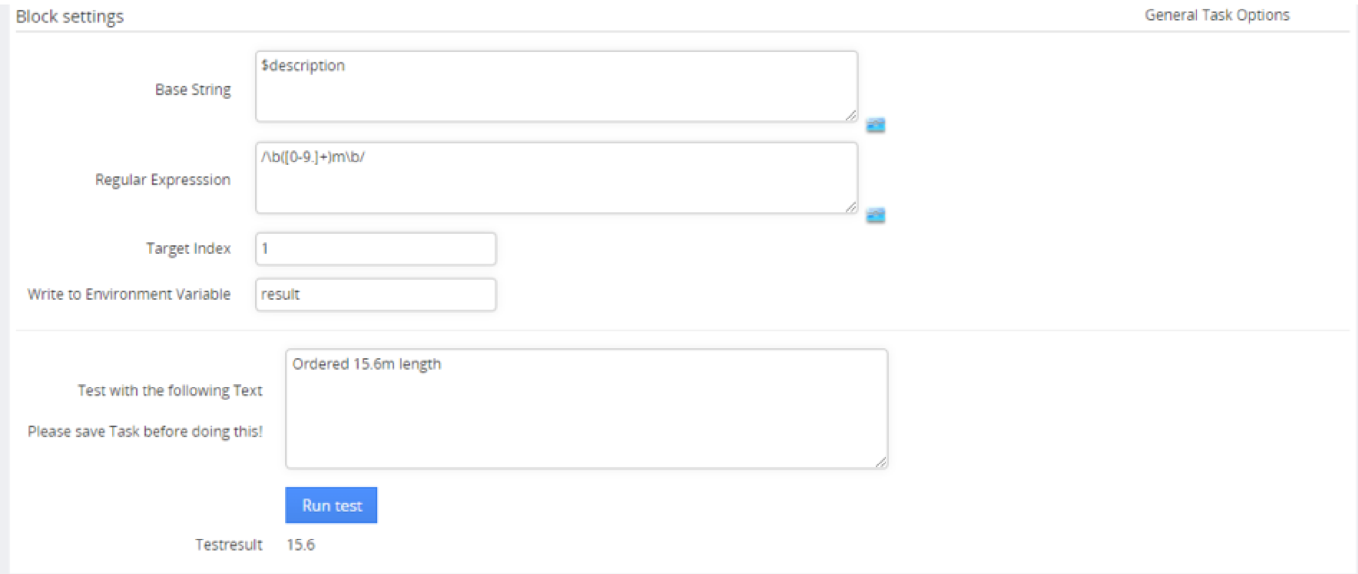


Regex String

This action allows to parse some data from a fulltext string, by using a regular expression.

You set the regular expression, which include result groups and define the index you want to have. You can only fetch 1 result per definition.

The testfield have no effect during execution, but can be used to check the configuration.



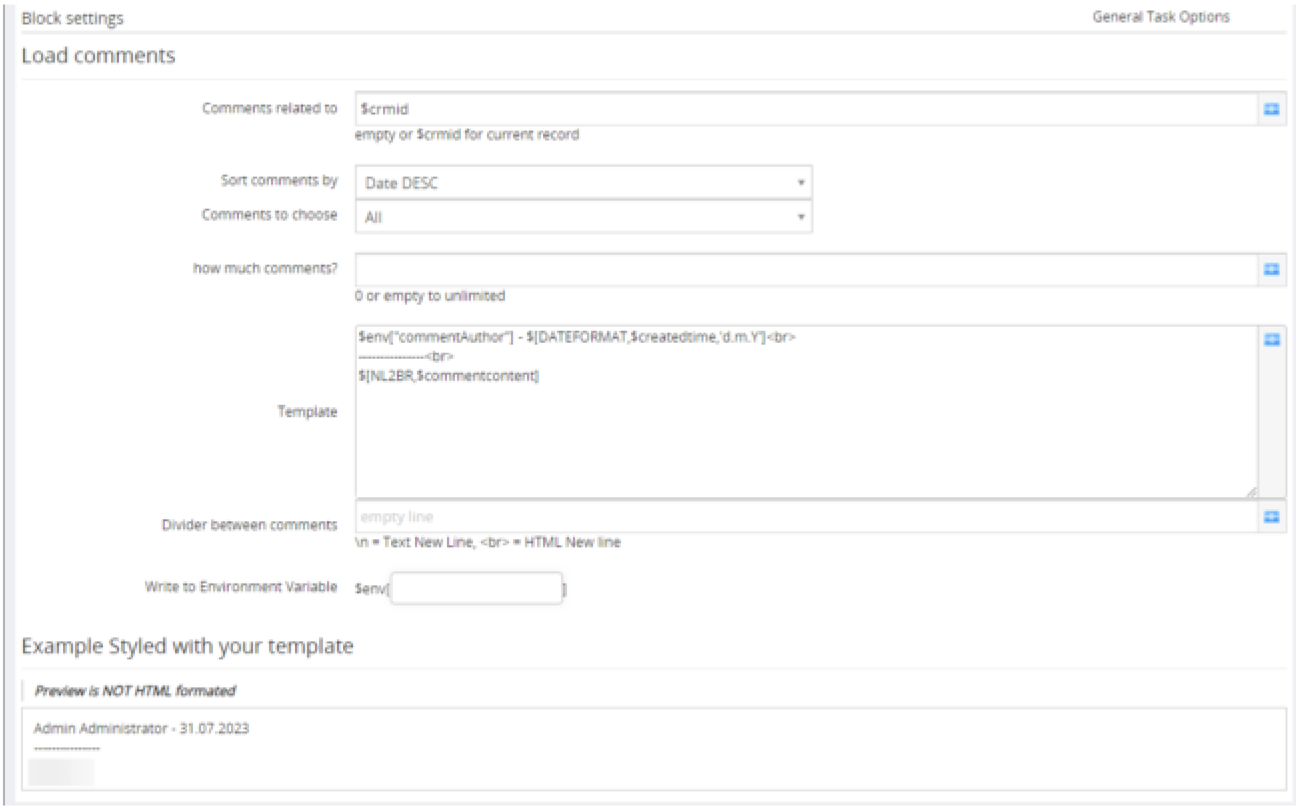
execute import process

This action can start an import workflow from another workflow. Use case can be, that you download a file over CloudFile module and use that file to trigger import.

The configuration defined everything you would also input during a manual import execution of this workflow.

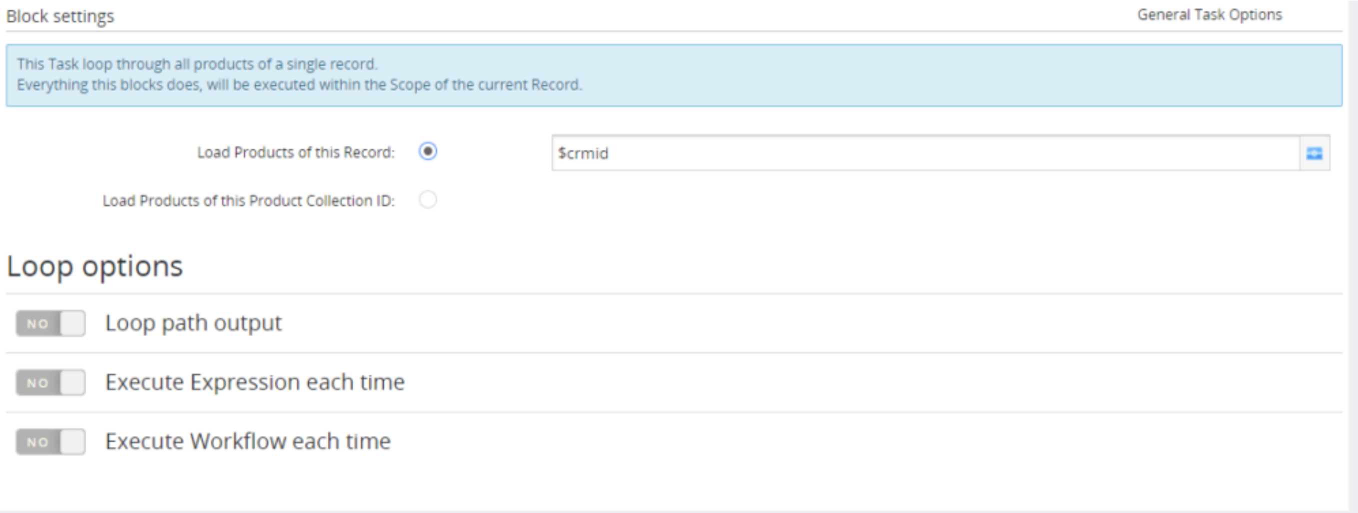
style record comments

This action generates a formatted output of all comments, which are assigned to a record.
A possible use case would be to use this generated content within an email body.



Loop through Products

When you need to loop over products of a record or product collection, you can do this by using this action.
You can executed a path output (The top one), a expression or a complete workflow per product. The $env Variable is shared over all ways.



move eMail to folder

In Workflows, which are executed within MailScanner, you can move the currently processed mail to folder in IMAP connection.

To do this, you need to select the MailScanner, which is assigned with this workflow and then the folder, where you want to move the mail.

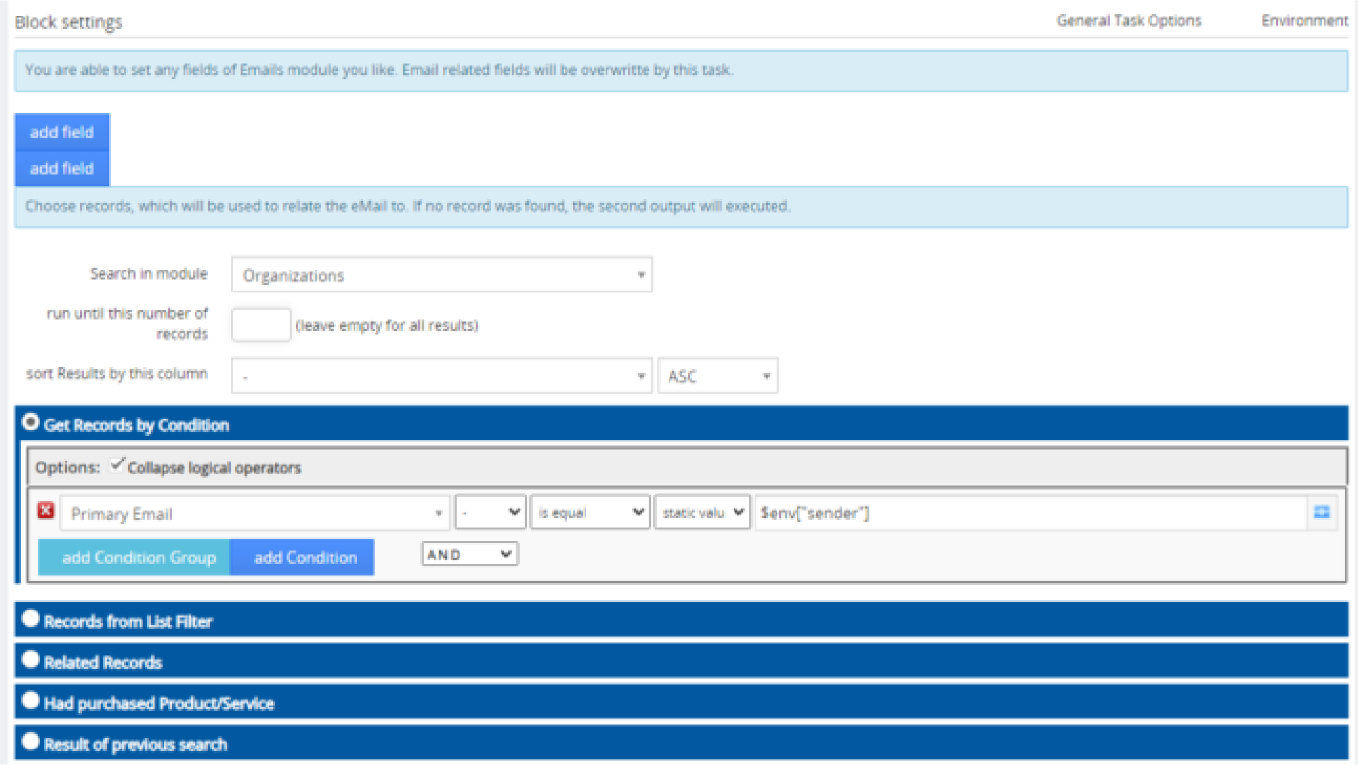
set eMail read/unread

In Workflows, which are executed within MailScanner, you can set/remove the SEEN flag of the currently processed eMail.

store eMail to record

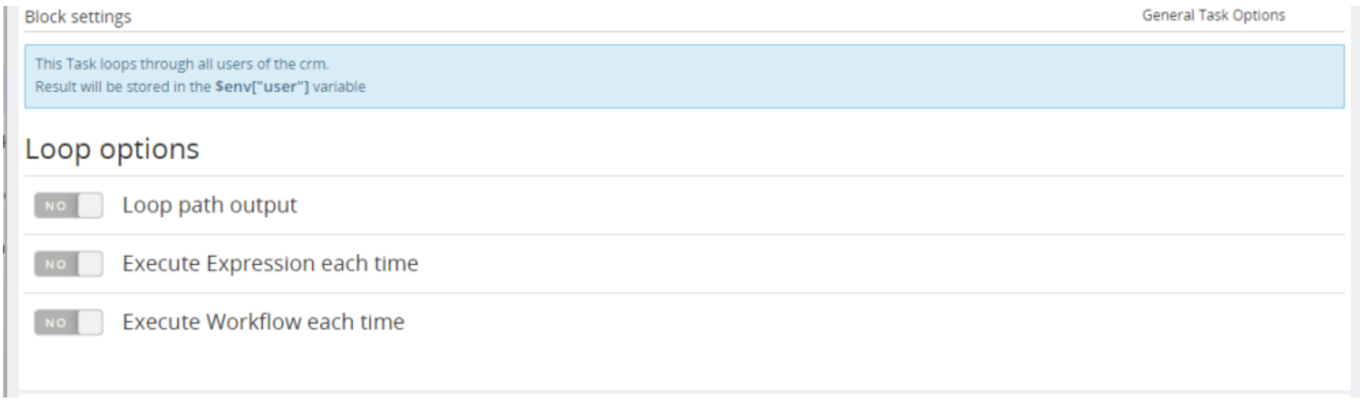
In Workflows, which are executed within MailScanner, you can add this mail to a one or more records in CRM.

Additionally, when your EMail activity type have some custom fields, you can set them.



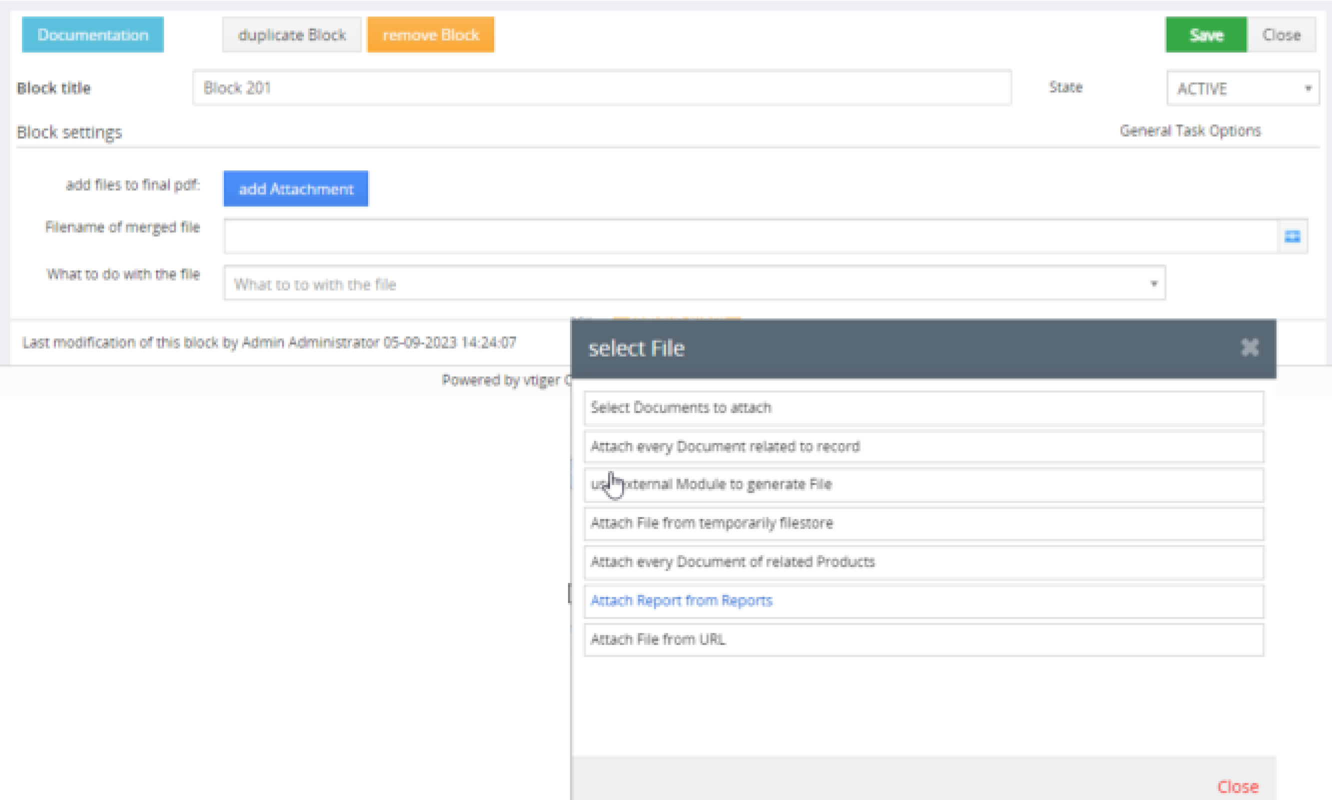
Loop over users

This action loop over all users in CRM and execute a path, a expression or a complete workflow with this record.
The Userdata is always available as **$env[“user”]** variable. You cannot change anything during execution.



merge PDF files

This action can combine multiple PDF files to a single one. The PDF files can be taken from Document records, filestore or by any other supported way of generating attachments.



PDF Generator Integration

Similar to PDFMaker integration, you can also generate PDF's by using the PDFGenerator module from redoo Networks.
It works equal to the PDFMaker action.