Magento / ContactLab Integration

Content

[Preface 4](#_Toc410833748)

[Modules description 4](#_Toc410833749)

[Installation Guide 5](#_Toc410833750)

[System Requirements 5](#_Toc410833751)

[Install the plugin 5](#_Toc410833752)

[Configuration 6](#_Toc410833753)

[About [GLOBAL] and [STORE VIEW] configuration 6](#_Toc410833754)

[Setup 7](#_Toc410833755)

[General 7](#_Toc410833756)

[File Transfer Configuration 8](#_Toc410833757)

[SOAP configuration 8](#_Toc410833758)

[SOAP Options 9](#_Toc410833759)

[Behaviour 10](#_Toc410833760)

[Error notify email 11](#_Toc410833761)

[Cron configuration example 12](#_Toc410833762)

[Magento → ContactLab Configuration 13](#_Toc410833763)

[Setup 13](#_Toc410833764)

[Custom fields 15](#_Toc410833765)

[Aggregated fields (Statistics) 16](#_Toc410833766)

[Queue parameters 17](#_Toc410833767)

[Cron Configuration example 18](#_Toc410833768)

[ContactLab → Magento Configuration 19](#_Toc410833769)

[Setup 19](#_Toc410833770)

[Manage Queue 19](#_Toc410833771)

[Transactional emails 20](#_Toc410833772)

[Transactional email configuration 20](#_Toc410833773)

[Queue parameters 20](#_Toc410833774)

[Recurring Email 21](#_Toc410833775)

[Templates configuration 21](#_Toc410833776)

[Generic email campaign 21](#_Toc410833777)

[XML Delivery parameters 22](#_Toc410833778)

[XML Delivery Queue 22](#_Toc410833779)

[Recurring Email – Abandoned Cart 23](#_Toc410833780)

[Recurring Email – Wishlist Remind 24](#_Toc410833781)

[About Cron jobs Configuration 25](#_Toc410833782)

[Queue options 26](#_Toc410833783)

[Manage task queue 27](#_Toc410833784)

[Task operations description 27](#_Toc410833785)

[Manual queuing of tasks 28](#_Toc410833786)

[Import/export of subscribers 28](#_Toc410833787)

[Manage Uk Table 28](#_Toc410833788)

[Recurring email 28](#_Toc410833789)

[Newsletter subscribers export format 29](#_Toc410833790)

[View operation logs 31](#_Toc410833791)

[Export operative details 32](#_Toc410833792)

[Full export 32](#_Toc410833793)

[Incremental export 32](#_Toc410833794)

[UK Table 32](#_Toc410833795)

[A guide for the first export 33](#_Toc410833796)

[Checklist 33](#_Toc410833797)

[Queue the export 33](#_Toc410833798)

[Checklist for a retry 34](#_Toc410833799)

[DataExporter 35](#_Toc410833800)

[DataExporter on demand 35](#_Toc410833801)

[DataExporter on Cron 35](#_Toc410833802)

[Synchronize subscription status on real time 36](#_Toc410833803)

[ContactLab -> Magento 36](#_Toc410833804)

[Magento -> ContactLab 36](#_Toc410833805)

[Recurring emails 37](#_Toc410833806)

[Templates management 37](#_Toc410833807)

[Newsletter template attributes 37](#_Toc410833808)

[Template information 38](#_Toc410833809)

[Product templates 39](#_Toc410833810)

[Customer filter options 40](#_Toc410833811)

[Cron Information 40](#_Toc410833812)

[Template text 41](#_Toc410833813)

[General attributes/object 41](#_Toc410833814)

[Product snippets 42](#_Toc410833815)

[Preview 42](#_Toc410833816)

[Debug extraction queries 43](#_Toc410833817)

[Show plugin version 44](#_Toc410833818)

[Show plugin Release Notes 45](#_Toc410833819)

# Preface

This user manual is designed to facilitate the use of ContactLab modules for importing and exporting newsletter subscribers, send transactional email through Smart Relay ContactLab service and send generic Magento Newsletter templates, abandoned cart and wishlist remind through the ContactLab XML Delivery.

# Modules description

The main purpose of this module is to connect Magento to the ContactLab Platform to allow the sending of newsletters to Magento subscribers through the professional ContactLab service. The module is composed by the following parts:

* a configuration section where the module can be configured
* a function to export newsletter subscribers
* a function to import newsletter unsubscriptions
* a function to send transactional email through ContactLab Smart Relay
* a function to send Magento campaigns, abandoned cart and wishlist remind through the ContactLab XML Delivery
* a queue to manage tasks

The export is needed to send newsletter subscribers to the ContactLab platform, on the other side the import is needed to receive newsletter unsubscriptions made through the unsubscribe link in the mail.

**contactlab\_commons** module provides configuration for data exchange, and the task management, while **contactlab\_subscribers** module takes care about import/export newsletter subscribers from local or remote server to Magento or synchronize status in real time mode. **contactlab\_transactional** module takes care about sending email via Smart Relay Service. **contactlab\_templates** modify the standard Magento template / newsletter sending operation in order to send them with the ContactLab XML Delivery service. It offers also the function to send special custom email templates to abandoned cart customers and wishlist remind emails.

User has the possibility to setup Cron jobs or to use the user interface into Magento administration section for importing or exporting data.

Once an action starts with exporting or importing operations, tasks are added to a queue for completion of the queued operations.

In the meanwhile if some import or export task fails for some reason some retry attempts are performed and the task is reinserted into the queue for the completion. The user also has the possibility to suspend or to cancel tasks which are already in the queue and have not reached the completion nor fails.

When the operation succeed the completed tasks are marker as closed.

If the number of retry exceed the maximum number of retries for one type of operation, the task is marked with failed status.

# Installation Guide

## System Requirements

Below are annotated basic System Requirements for the ContactLab Commons and ContactLab Subscribers modules:

* PHP Compatibility:
  + 5.2.13 - 5.3.24
* Required extensions:
  + PDO\_MySQL
  + simplexml
  + mcrypt
  + hash
  + GD
  + DOM
  + iconv
  + curl
  + SOAP (if Webservices API is to be used)
  + Safe\_mode off
  + Memory\_limit no less than 256Mb (preferably 512)
* MySQL:
  + EE 1.13.0.0 or later: MySQL 5.0.2 or newer
  + EE 1.12.0.2 or earlier: MySQL 4.1.20 or newer
  + CE (all versions): MySQL 4.1.20 or newer
* Server - hosting - setup:
  + Ability to run scheduled jobs (crontab) with PHP 5
  + Ability to override options in .htaccess files

## Install the plugin

The integration plugin is deployed as two different compressed folder like this:

1. *contactlab\_x.y.z\_step1.tgz*
2. *contactlab\_x.y.z\_step2.tgz*

Decompress each file, first step1, then step2, in the root of your Magento installation. Remember to clear the Magento Cache and proceed to setup.

# Configuration

Once installed, the plugin create a new tab “CONTACTLAB” on Magento configuration.

Each page of ContactLab configuration provides group of fields useful for setup of each plugin’s functionality.

There is a common group of configuration (Setup):

* two configuration sections relative to subscribers synch between Magento/Contactlab and Contactlab/Magento
* one section relative to the transactional email sending
* three sections about the recurring emails (generic / abandoned cart / wishlist remind)

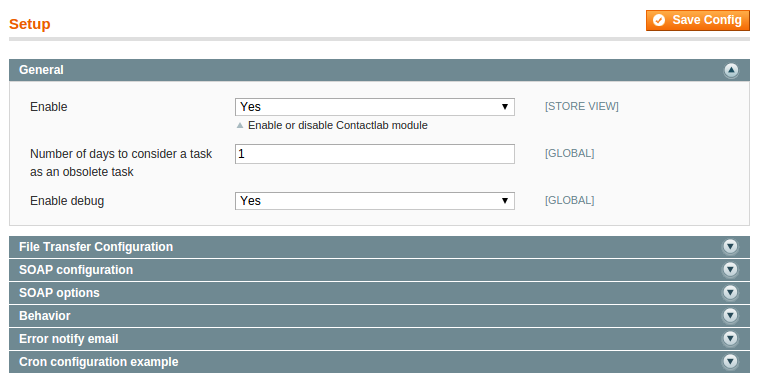
## About [GLOBAL] and [STORE VIEW] configuration

Next to every option of the Magento plugin configuration, is indicated the “scope” of the setting. Options can have a [GLOBAL] scope or a [STORE VIEW] scope:

* **[GLOBAL]** is a configuration option common to every website and store view. For example, the enable flag of every different module, or the period of aggregated fields statistic calculation.
* **[STORE VIEW]** is a configuration that has a default option that the administrator can change, but has a value that can be specified optionally for every store view of the Magento instance.

## Setup

### General

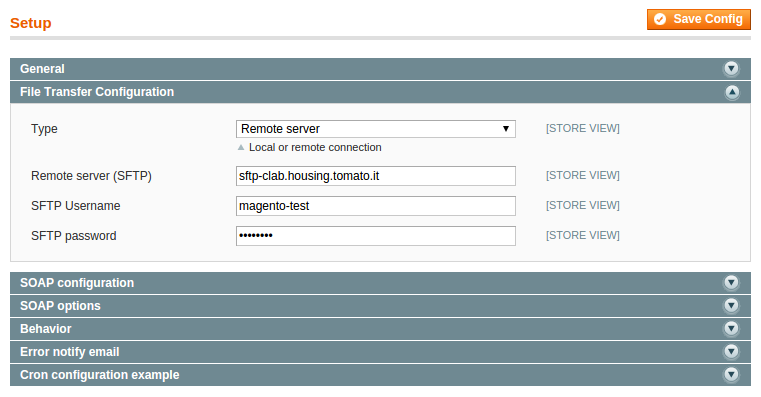


Into the Global configuration section, the installed ContactLab Commons module can be enabled or disabled.

User can define the number of days to consider a succeeded task old and let the Cron job task clear the queue.

The “Enable debug” field enable a set of tools used during the startup of the application, like more log verbosity, buttons in task queue page that force scan execution and running of tasks.

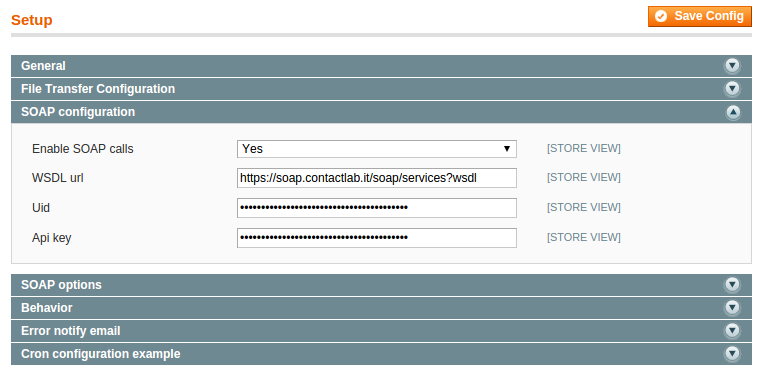
### File Transfer Configuration



In this section user can define the connection type, which can be local or remote. If the connection type is Local server, then the local folder path for import and export of the newsletter subscribers must be defined, default value is set to /tmp folder but should be replaced with your own folders.

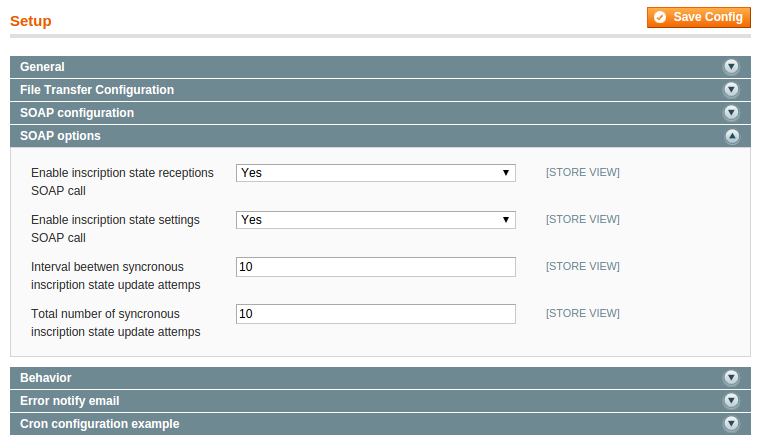
If the connection type is set to *Remote server,* then the remote server address, export and import path of the newsletter subscribers, SFTP username and password must be defined.

### SOAP configuration



Events’ synchronizations and real time operations between Magento and ContactLab are managed with SOAP calls. In this tab you can configure basic configuration for these SOAP calls. The SOAP wsdl url is pre-configured but can be customized as well. Uid and Api key are code provided during ContactLab setup.

### SOAP Options



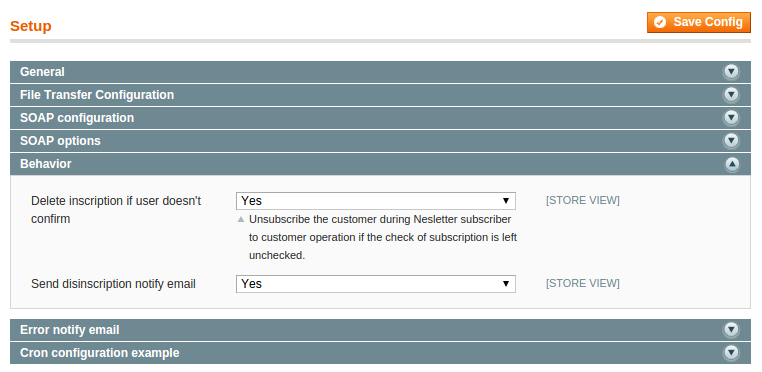
SOAP option section allow to enable/disable the synchronous call to ContactLab to get/set the status of subscription of a customer into remote ContactLab database.

When a customer request the account page, its subscription status is fetched from the remote ContactLab database with a SOAP. The same thing happen when the customer subscribe or unsubscribe from the newsletter.

The last two parameters define the behavior to keep in case of failure of the inscription. The SOAP inscription operation runs synchronously within the http request. In case of failure, the request will be queued and will be processed for a (configurable) number of attempts after a (configurable) number of minutes.

Read the “Queue options” section of this document for more information.

### Behaviour

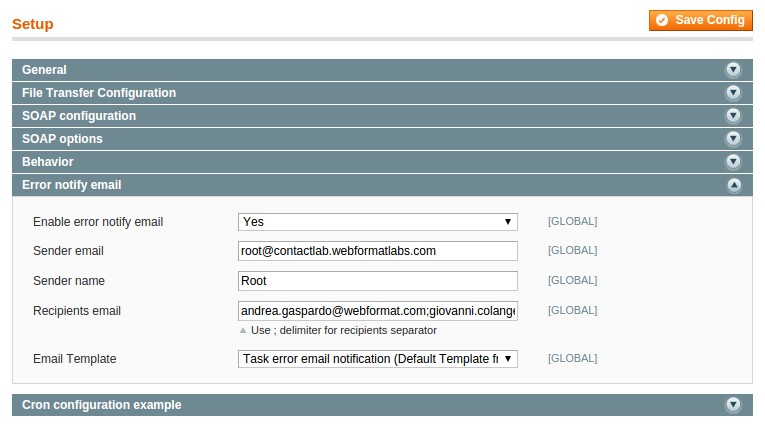


This section enable/disable two specific behaviors of Magento, related to ContactLab policies.

**Delete inscription if user doesn’t confirm**: if selected, modifies the standard policy of Magento unsubscribing the customer that, during the register as customer operations, does not confirm the subscription. So that, when a subscribed mail becomes a customer and leave the subscription unchecked, the subscription will be deleted.

**Send disinscription notify mail**: if the first option is selected, you can choose here if, in case of automatic disinscription, a notification mail to the customer should be sent.

### Error notify email



It is possible to receive email notification on tasks which are marked as failed enabling this option and defining sender and receiver parameters. The notification contains the list of accumulated failed tasks which are collected in the time range between Cron jobs.

### Cron configuration example

Magento allows you to schedule custom tasks in an XML configuration, in a similar manner to the UNIX Cron tab style. There are few Cron jobs already configured that:

* proceed to consume the queue every minute
* clear the queue from old completed tasks
* send an alert email if an error occurred during some data exchange
* scan Recurring Email templates, looking for pending abandoned cart or wishlist remind email



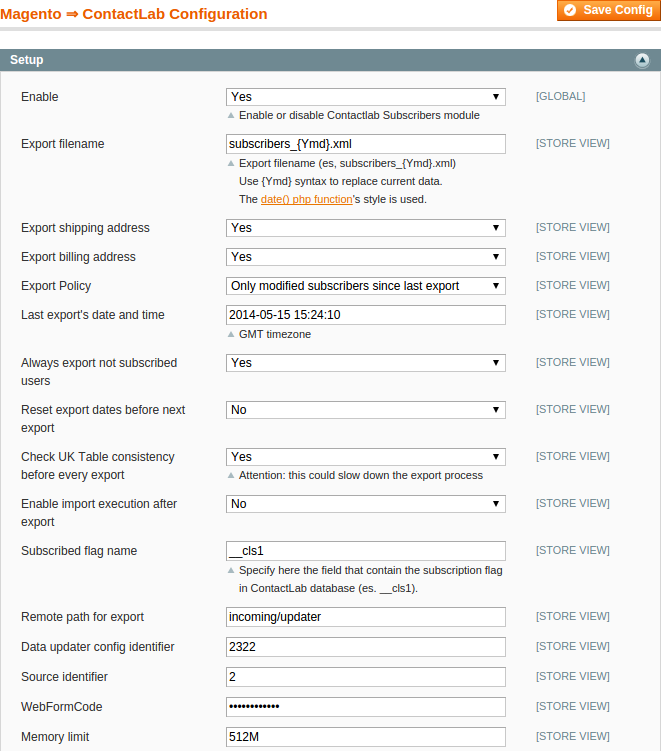
There should be two other cron jobs that should be manually configured:

* one that starts daily export of newsletter subscription
* one that import daily unsubscriptions from ContactLab

This configuration will be shown later in the “About Cron jobs Configuration” section.

## Magento → ContactLab Configuration

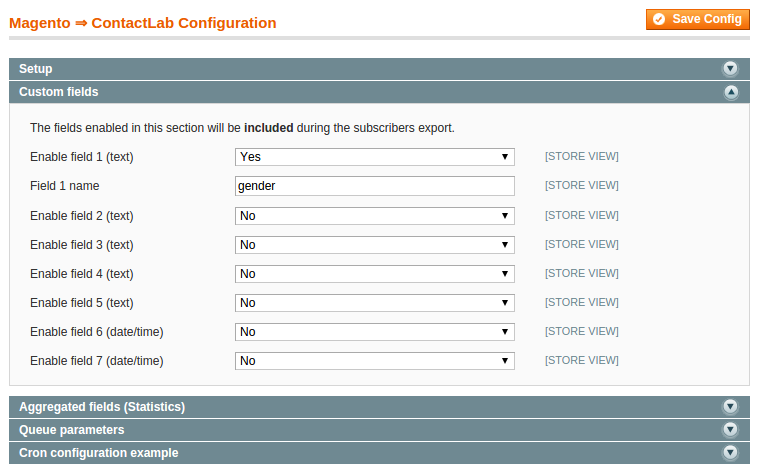
### Setup



Below the explanation of every configuration field:

|  |  |
| --- | --- |
| **Enable** | ContactLab Subscribers module can be enabled or disabled. |
| **Export filename** | The file name has to be defined for newsletter subscribers exporting operations.  The filename can contain user formatted date. The format specifications are the same of php date function (so refer to this page <http://php.net/manual/it/function.date.php> for more details) |
| **Export shipping address,**  **Export billing address** | The Export billing address and Export shipping address options allows user to determine which addresses can be exported for the single customer which is subscribed to the newsletter list.  In case the customer have not any of these addresses, an empty XML tag will be printed. |
| **Export Policy** | The Export policy has three defined options for exporting newsletter subscribers.  The export option can be applied on All subscribers, All subscribers just for the next time and Only modified subscribers since last export date.  If the third option, Only modified subscribers since last export date, is selected the last export date can be customized.  See “**Export operative details**” for other details. |
| **Last export's date and time** | Last export date time.  This option is updated automatically after every export, but user can modify it in order to re-export customers modified before the export. |
| **Always export not subscribed users** | This option is selected by default, so the plugin will export customers and newsletter subscribers even if they haven’t subscribed newsletter.  Disabling this option only subscribed customers will be exported.  *The reason of exporting a customer that’s not a newsletter subscriber is that he could be enabled to receive service communications via email.* |
| **Reset export dates before next export** | “Reset export dates before next export” is useful during startup operations and is available only in debug mode.  Enabling this option every exported record during the next export job, would be consider as “new”, so the \_\_cls field will be always exported. |
| **Enable import execution after export** | After the export process and after the file is copied into the destination server, a SOAP call can be configured to trigger ContactLab Platform to start the import process. |
| **Check UK Table consistency before every export** | Check the consistency of UK table after every export (true by default).  If the UK table isn’t consistent, the export won’t begin. |
| **Subscribed flag name** | This is the \_\_cls*N* flag name of the contact table in ContactLab database. By default is set to “\_\_cls1” but can be customized if necessary. |
| **Remote path for export** | This is the remote path for the export, setted with “incoming/updater” default value |
| **Data updater config identifier** | The Data updater config identifier provided during ContactLab setup (*ID DataUpdater*) |
| **Source identifier** | The source identifier provided during ContactLab setup (*Id DB*) |
| **WebFormCode** | The WebFormCode provided during ContactLab setup |
| **Memory Limit** | Memory limit is defined by php environment (php.ini or .htaccess file) but it’s often too low to fulfill some of the most expensive Magento/Contactlab elaborations.  The admin user can here define the memory\_limit value to override the system default value.  **Take care! Specify here the number of bytes (es “1048576” = 1 mega byte) or a shorthand value (like “1M”) as described in this page:**  [**http://php.net/manual/en/faq.using.php#faq.using.shorthandbytes**](http://php.net/manual/en/faq.using.php#faq.using.shorthandbytes) |

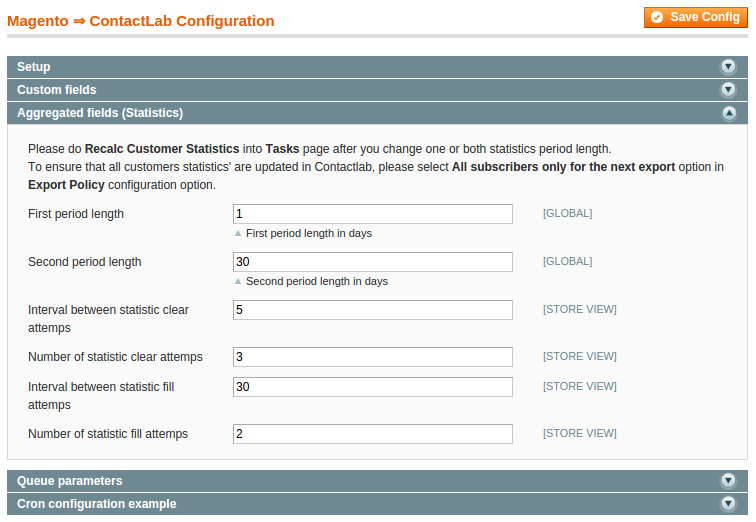
### Custom fields



User can define custom fields that will be exported into the XML document root. The available type of fields are text (5 fields) and date/time (2 fields) which are related to the customer specific data.

the fields configured in Magento must be the same configured in ContactLab. So, before configuring the custom field list, please check the ContactLab database setup with our customer support.

### Aggregated fields (Statistics)

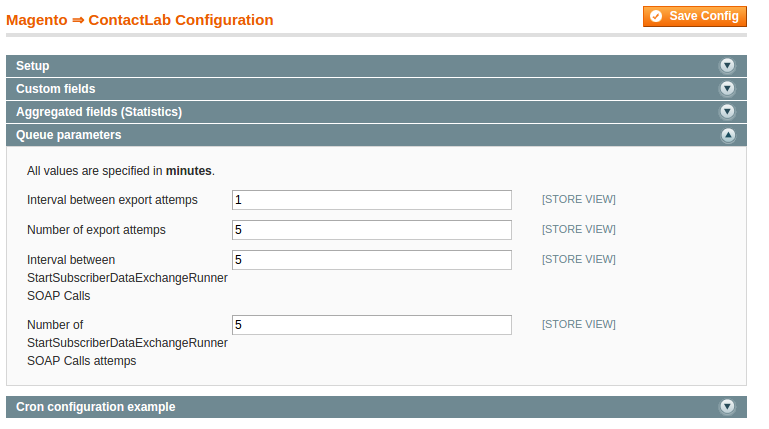


To export statistics for newsletter subscribers, user can use two different periods defined in days. In the example above, statistics would be calculated in this way:

|  |  |
| --- | --- |
| period1\_amount | Total amount of closed orders in the last day |
| period1\_products | Number of products in the last day |
| period1\_orders | Number of orders in the last day |
| period2\_amount | Total amount of closed orders in the last 30 days |
| period2\_products | Number of products in the last 30 days |
| period2\_orders | Number of orders in the last 30 days |

The last four parameters specify the number of attempts and the interval in minutes that will be used in case of failure during filling and cleaning of statistic. Please, read the section Queue options of this document.

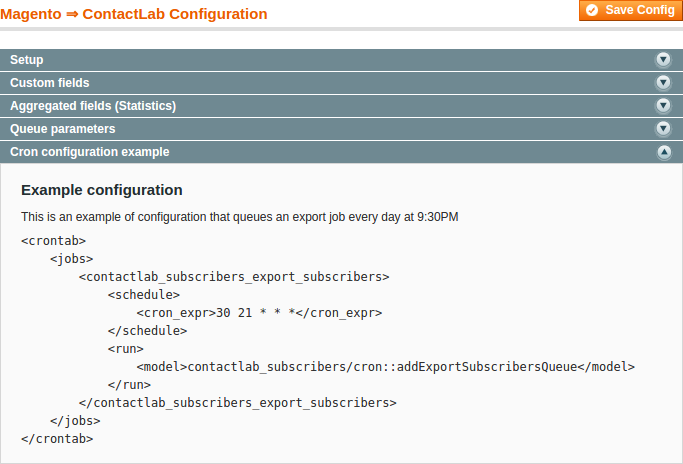
### Queue parameters



This section defines the queue parameters for Magento => ContactLab Sync operation.

In case of failure of the export process, there will be a (configurable) number of attempts within a (configurable) interval of minutes. Read the section Queue options of this document for more information.

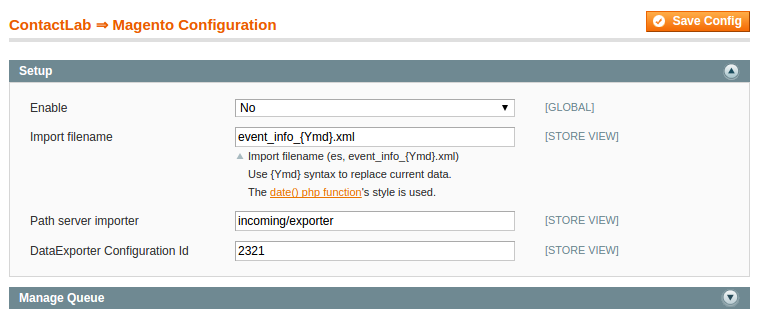
### Cron Configuration example



This tab show a Magento/ContactLab Sync operation cron configuration. In this example the export subscriber job is configured to run every day at 21:30.

## ContactLab → Magento Configuration

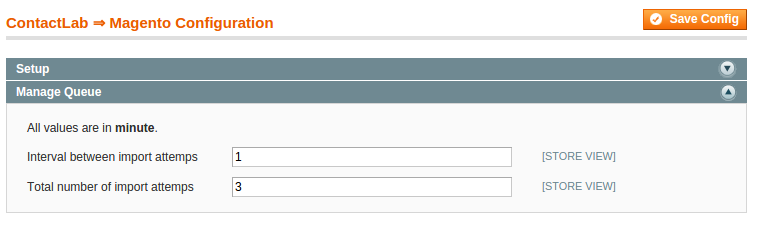
### Setup



This section defines the main configuration parameters useful for the import subscribers’ from ContactLab Db to Magento functions.

|  |  |
| --- | --- |
| **Enable** | Enable the import subscription functionalities |
| **Import filename** | Import file name for new subscribers, the file name can contain formatted date like the case of the export filename. |
| **Path server importer** | The relative path to be used during import operations |
| **DataExporter Configuration Id** | It’s the configuration id provided by ContactLab during service startup |

### Manage Queue

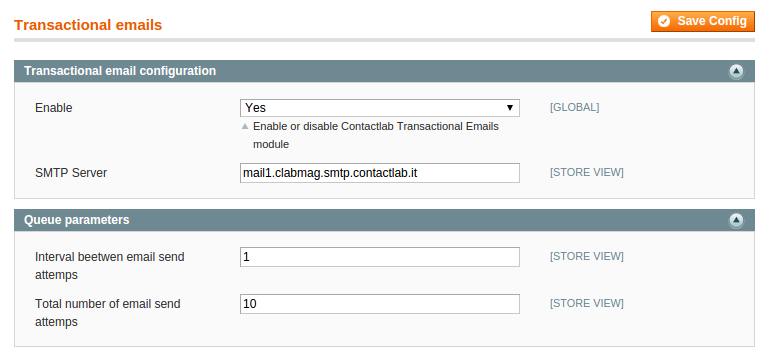


This section defines the queue parameters for ContactLab/Magento Sync operation. Read the section Queue options of this document.

## Transactional emails

### Transactional email configuration

### Queue parameters



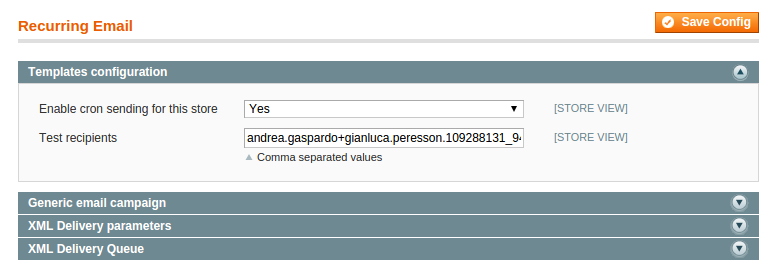
In this section, enable or disable the sending of transactional email through Smart Relay, and specify the SMTP server (with server host name or IP address).

In the queue parameters’ section, as usual, user can configure the delay between retries, and the number of retries of email sending. Read the section Queue options of this document for more details.

## Recurring Email

Recurring email configuration section contains configuration options common to every type of Recurring mail: generic, abandoned cart and wishlist remind.

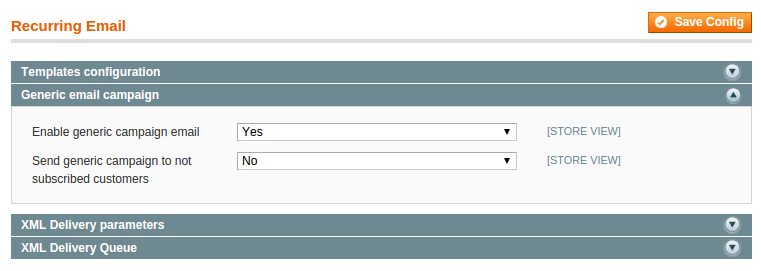
### Templates configuration



Into the “Template configuration” section the administrator specify if use the XML Delivery service to send campaigns in *cron* with the selected store. If disabled, the Magento default behavior will be used.

Test recipient is a comma separated list of email used to send test purpose email from the template setup page.

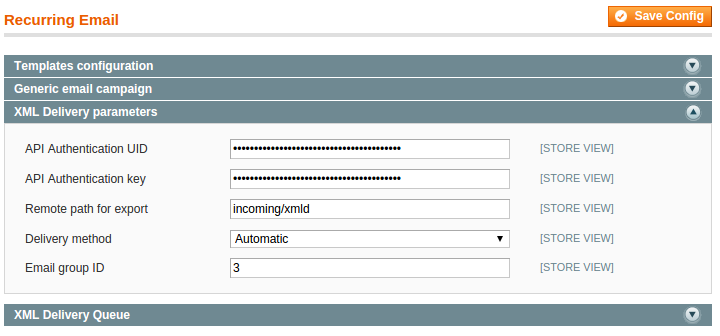
### Generic email campaign



The generic campaign is a campaign that will be sent manually by the administrator, and not in cron like abandoned cart or wishlist remind. This section tab allow the administrator to enable or disable the generic campaign XML Delivery.

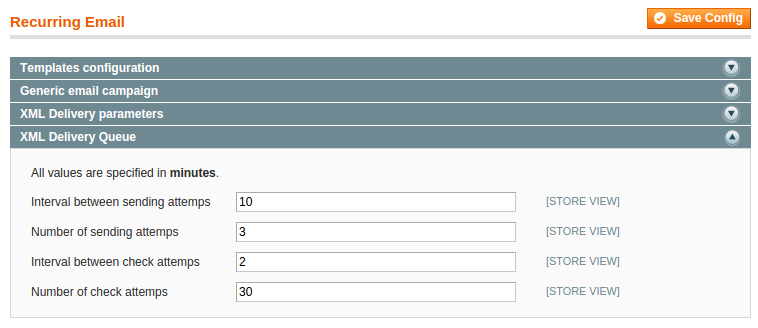
The “**Send campaign to not subscribed customers**”, that can be found into the other two recurring email types, allow to consider not subscribed to newsletter customers, into the list of possible email recipients.

### XML Delivery parameters



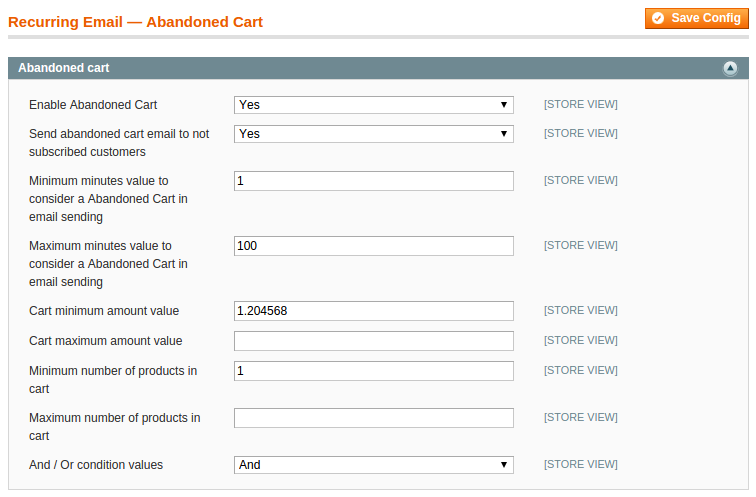
This tab contains authentication XML Delivery parameters, all provided during ContactLab services setup.

### XML Delivery Queue



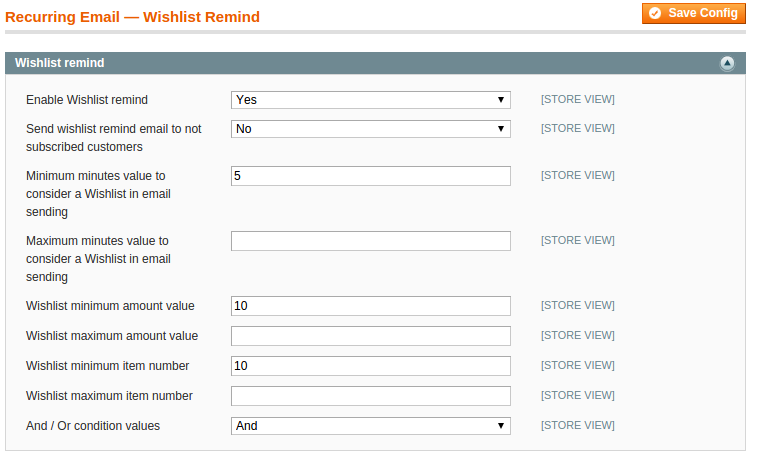
This section defines the queue parameters for XML Delivery. Read the section Queue options of this document for more details.

## Recurring Email – Abandoned Cart



The configuration of “**Abandoned cart**” is very similar to the Wishlist Remind one. See the next section “**Recurring Email – Wishlist Remind**” for details about this configuration.

## Recurring Email – Wishlist Remind



Both “Wishlist remind” and “Abandoned cart” sections have similar configuration, listed in the table below. These options will be default values for all templates that will be created.

|  |  |  |
| --- | --- | --- |
| Configuration | Description | Example |
| Enable Wishlist remind / Abandoned Cart | Once configured one or more templates for wishlist remind or abandoned cart, the admin can here enable or disable every abandoned cart or wishlist remind XML Delivery. | Yes |
| Send wishlist remind / Abandoned Cart email to not subscribed customers | Allow to consider not subscribed to newsletter customers, into the list of possible email recipients for each recurring email type. | Yes |
| Minimum / Maximum minutes value to consider a Wishlist / Abandoned Cart in email sending | Range of values (number of items) to consider or not a customer wishlist or cart into the possible list of recurring email recipients. | 120 / 1200 |
| Wishlist / Abandoned Cart minimum / maximum amount value | Range of values (prices) to consider or not a customer wishlist or cart into the possible list of recurring email recipients. | 0.00 € / 4000.00 € |
| Wishlist / Abandoned Cart minimum / maximum item number | Range of minutes to consider or not a customer wishlist or cart into the possible list of recurring email recipients, based upon the last date time of cart/wishlist modification. | 4 / 50 |
| And / Or condition values | Consider value and number of items in and / or into the data extraction query. | And |

### About Cron jobs Configuration

To configure Cron job operations, edit local.xml file in app/etc folder of your Magento installation and insert these lines:

**<contactlab\_subscribers\_export\_subscribers>**

**<schedule>**

**<cron\_expr>0,30 \* \* \* \*</cron\_expr>**

**</schedule>**

**<run>**

**<model>contactlab\_subscribers/cron::addExportSubscribersQueue</model>**

**</run>**

**<store>1</store>  
</contactlab\_subscribers\_export\_subscribers>**

**<contactlab\_subscribers\_import\_subscribers>**

**<schedule>**

**<cron\_expr>30 02 \* \* \*</cron\_expr>**

**</schedule>**

**<run>**

**<model>contactlab\_subscribers/cron::addImportSubscribersQueue</model>**

**</run>**

**</contactlab\_subscribers\_import\_subscribers>**

In “cron\_expr” tag specify (with the same syntax of GNU/Linux crontab) how often and in which moment the job would start. In this example the export of subscribers is scheduled every 30 minutes of every hour of the day, while the import operation would be performed every day at 2:30 a.m.

**Notice that in the above example the optional tag “store” with value 1 is meant to generate the subscribers’ export only for store with id “1”.**

## Queue options

For every type of operation managed by the task queue, the delay between attempt and the number of attempts can be defined by the administrator.

All **intervals** are defined in **minutes** and are triggered when the task operation **fails for some reason**.

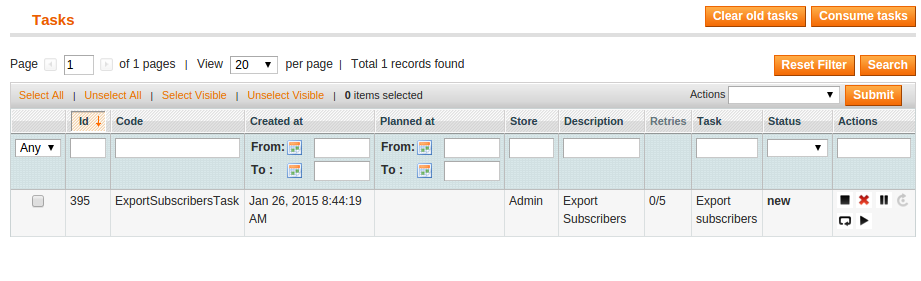
If the task does not succeed in defined **numbers of attempts**, the task will be marked as failed and the notice will be sent via email (if this option has been configured).

# Manage task queue

## Task operations description

Task queue operations can be monitored and managed with the Newsletter » ContactLab » Tasks page.

This page shows a list of all queued tasks, each row shows the task status, the date and time of creation, the panning date time, the store associated with it, the current number of retry and a set of actions that can be performed with each row or with every selected row (mass actions).

Each action within a single row can be enabled or disabled depending on the status of the row.

The “Description” column, in the case of long running tasks, shows the percentage of completion.

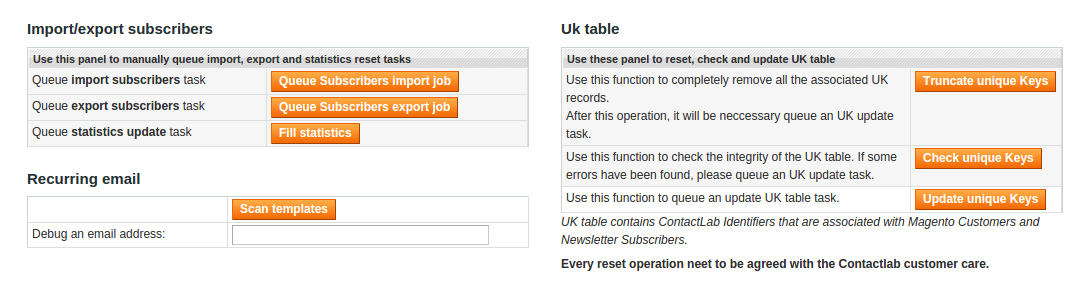
The operations and descriptions for tasks are listed in the table below.

|  |  |
| --- | --- |
| Task actions | Task action description |
| Run task | By selecting Run task the newsletter subscribers starts to be exported. |
| Retry task | If the task fail user can use Retry task for the export operation. |
| Suspend task | User can Suspend task when running. |
| Unsuspend task | With Unsuspend task, user unsuspend the previously suspended tasks. |
| Cancel task | Any tasks can be deleted from the operation queue with Cancel task. |
| Delete task | Remove the task from the queue |

# Manual queuing of tasks

Into the task management page there are 3 useful panels that allow to perform several manual operation. These manual operations should be used only during startup phase, because each of them should be queued by a cron operation.

**Note that every operation queued into the task queue is meant to be run by the Magento cron. If there’s no configured cron, the task have to be run manually with the “Run task” action by the administrator.**



## Import/export of subscribers

Into this panel you can:

* **Queue import subscribers task** – immediately import subscribers.
* **Queue export subscribers task** – immediately export subscribers.
* **Queue statistics update task** – truncate the statistics table and recalculate the whole table. This operation should be done before the first export and if a stats configuration has been changed (see “Aggregated fields (Statistics)” paragraph).

## Manage Uk Table

Use this section to perform task to manage Uk table. See more details into the “UK Table” paragraph.

* **Truncate unique Keys** – Use this function to completely remove all the associated UK records. After this operation, it will be neccessary queue an UK update task.
* **Check unique Keys** – Use this function to check the integrity of the UK table. If some errors have been found, please queue an UK update task.
* **Update unique Keys** – Use this function to queue an update UK table task.

## Recurring email

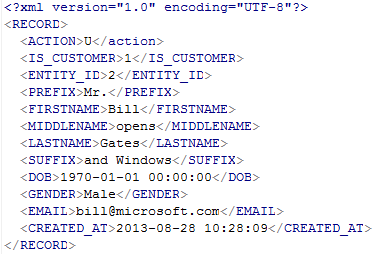
Use this section to manually perform a scan of every configured Abandoned Cart / Wishlist Remind newsletter template (see “Debug extraction queries” paragraph for more details).

## Newsletter subscribers export format

The example below show the XML document root for newsletter subscribers’ export.

The XML document root tree consist of main data for specific customer.

If there is newsletter subscribers which are not customers then the XML document root will contain only the subscriber email, while the other tags will be empty.

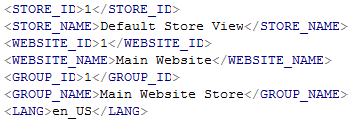


Below is the example of billing and shipping address data relative to specific customer and added to the XML document root.

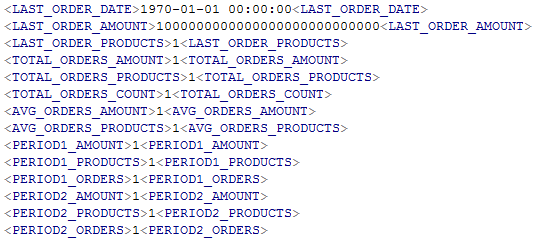
If the customer has billing and/or shipping address, both or existing will be added to the XML document root, if the address data does not exist for specific customer then XML tags will be empty.



Website specific data, like store, website, group and language, is added to the XML document root if the newsletter subscriber have customer account, in case the subscriber does not have the customer account empty tags will be printed.

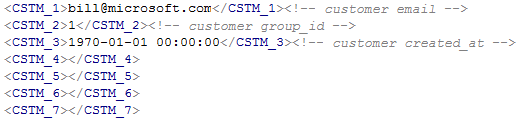


If customer have transactional data related to his account, the data will be exported within XML document root as follows.



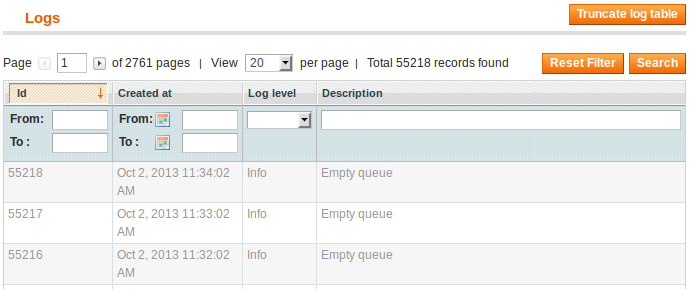
Optional custom fields, that can be defined **(picture 11)** in the system configuration section for installed module, if activated, will be shown only if the newsletter subscriber have the customer account.

In this example below we have set only 3 custom fields.



# View operation logs

Operation logs section allows user to view the state of the tasks queue. There are report export options for log table data and the user can choose to export log report in XLS or CSV format. The table can be truncated and all data within will be deleted.



# Export operative details

There are two types of export, full export and incremental. Full export will be use for the first execution, during the initial setup or in production if the Magento site has a small number of customers. Full export could be used also if some global policy (for example, statistics details) has changed. In any other case, incremental export should be used.

## Full export

In full export mode, every record will be exporter, regardless of its last modify date and time.

It’s important to notice that also in full export mode the column that identify the subscription flag, it’s synchronized only the first time a customer has been exported. That is to avoid the overwriting of ContactLab subscription status. To synchronize again the status of all the customer or guest subscribers, set the configuration flag “**Reset export dates before next export”**. This option will cause the overwriting of the ContactLab status with the Magento current status.

## Incremental export

In incremental export mode will be synchronized only customers or guest subscribers that has been modified, inserted or deleted since last export. Even the change of subscription status of a customer include it in the next incremental export.

### UK Table

When the module it’s installed a table it’s created that will contain references to:

1. Customers not subscribed to newsletter
2. Newsletter subscribers that are not customers (guest)
3. Customers subscribed to newsletter

This table will be populated during the install process, and will be kept updated by the plugin. The key of this table will be used to identify records between Magento and ContactLab.

**Attention:** In case of customers’ massive import or misconfiguration of the plugin, this table could lose its consistency. This would always introduce errors during the export process.

To check the consistency and repair this table, use the action “Check unique Keys” and “Update unique Keys” that can be found in the Tasks management page.

## A guide for the first export

This paragraph is a short description of the first Magento/ContactLab export.

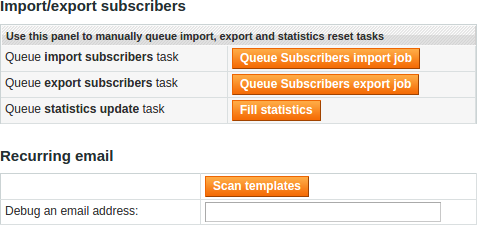
### Checklist

This is a checklist for a pre-export operation.

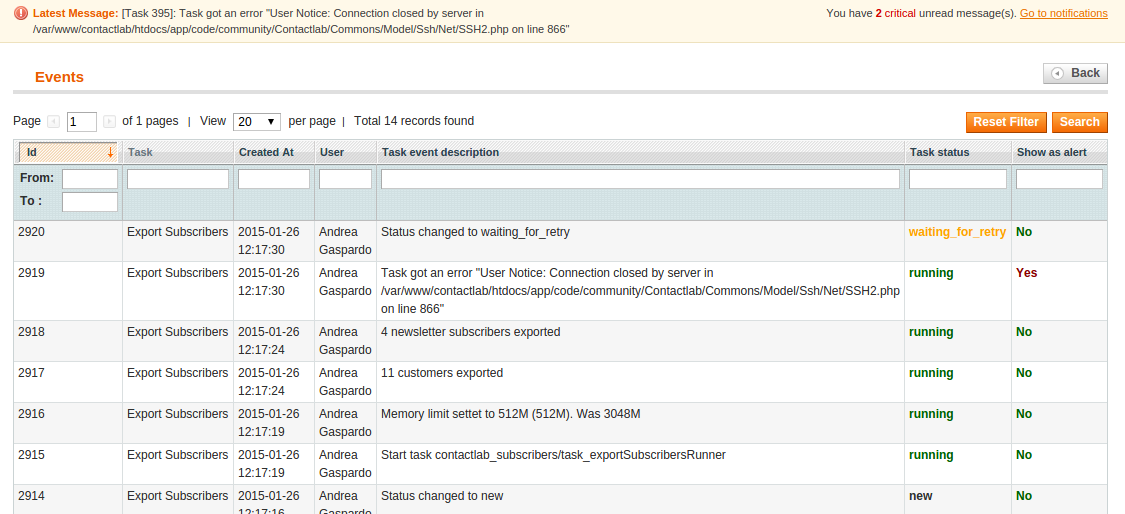
|  |  |
| --- | --- |
| Operation | Details |
| Check common configuration | Check the “ContactLab » Setup” configuration tab.  Check **FTP Configuration** and **SOAP configuration**. |
| Check UK table status | Open Task page and click “Check unique Keys”. If some errors have been found, queue an “UK update” task |
| Check statistics | Open Task page and click “Fill statistics”. |
| Check export configuration | Check “Magento ⇒ ContactLab Configuration”.  Is the export enabled?  Check “Export [billing/shipping] address” values.  Set Export Policy to “All subscribers only for the next export”  Check “Always export not subscribed users” value.  Set “Reset export dates before next export” to Yes. |

### Queue the export

After the checklist, you can queue the execution of the export manually by clicking “Queue Subscription export job” into the task page.



One queued, if there’s no cron configured for Magento, run manually the task. When the task is finished, check the exit status of the task clicking in the corresponding table row.



In this example, there has been an error during FTP transfer of the file.

### Checklist for a retry

If an export has failed, this is a checklist to perform another attempt, after the causes of the error has been removed.

|  |  |
| --- | --- |
| Operation | Details |
| Check export configuration | Set Export Policy to “All subscribers only for the next export”  Check “Always export not subscribed users” value.  Set “Reset export dates before next export” to Yes. |

Then, remove from the queue any other export task and queue again another export.

# DataExporter

DataExporter is used to export un-subscriptions from ContactLab to Magento. In Magento this task is called “Subscribers Import”. This operation can be made in Cron or on demand. Only unsubscription made from Contactlab platform are exported.

**Notice that the un-subscription event will be ignored during Magento import job if the subscription status has been changed after the creation of the export file**.

## DataExporter on demand

Use DataExporter on demand when you need to import un-subscription flag from ContactLab to Magento not programmatically. Magento will then check if the Subscriber Data Exchange status and then copy the xml data file, parse it and import data into Newsletter table. If the Subscriber Data Exchange is busy, the execution will be queued and repeated later.

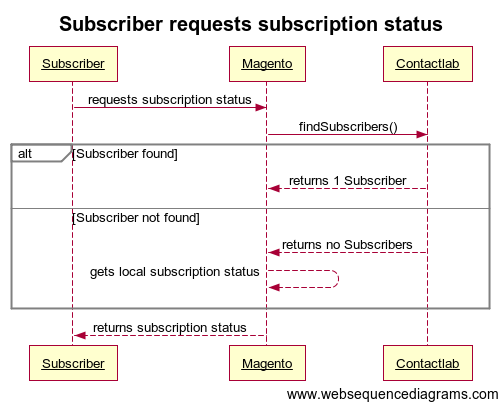
## DataExporter on Cron

On ContactLab it’s possible to schedule the execution of DataExporter. Since this operation is always incremental, in Magento the Subscriber Import operation has to be configured with the right time in order to be executed at the end of the DataExporter job.

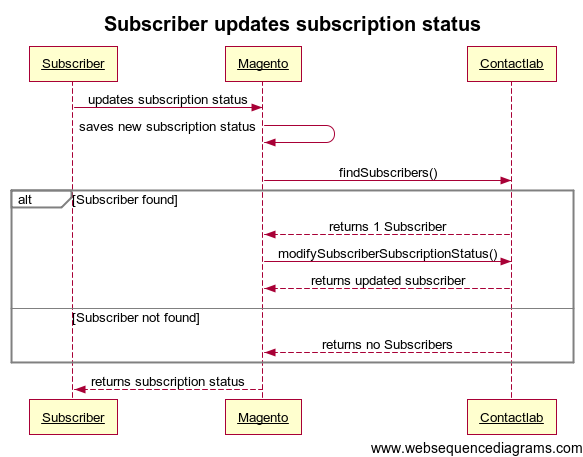
# Synchronize subscription status on real time

### ContactLab -> Magento

If the option “**Enable SOAP call to get subscription status**” has been enabled, every time a logged in user check its account information, Magento will check subscription status in ContactLab database with a SOAP call. If the user it’s not yet into ContactLab database or if a network error occur between servers communication, the Magento local value of subscription will be shown.



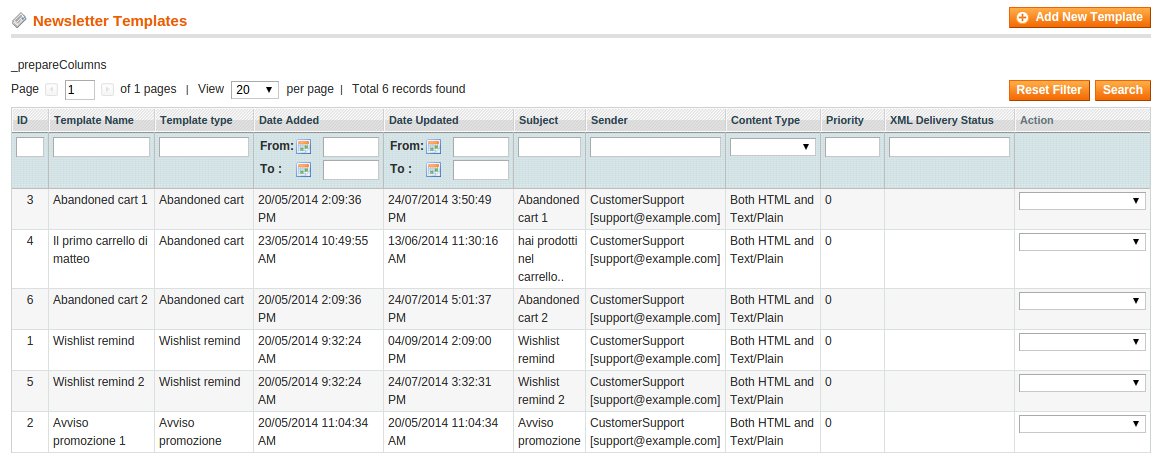
### Magento -> ContactLab



In the Magento account page the user can update his subscription status to subscribe or unsubscribe. If the option “**Enable SOAP call to set subscription status**” has been enabled, this operation will generate a SOAP call to ContactLab database in order to update the status of the subscriber.

# Recurring emails

## Templates management



After the configuration of recurring email it’s necessary to create one or more email template.

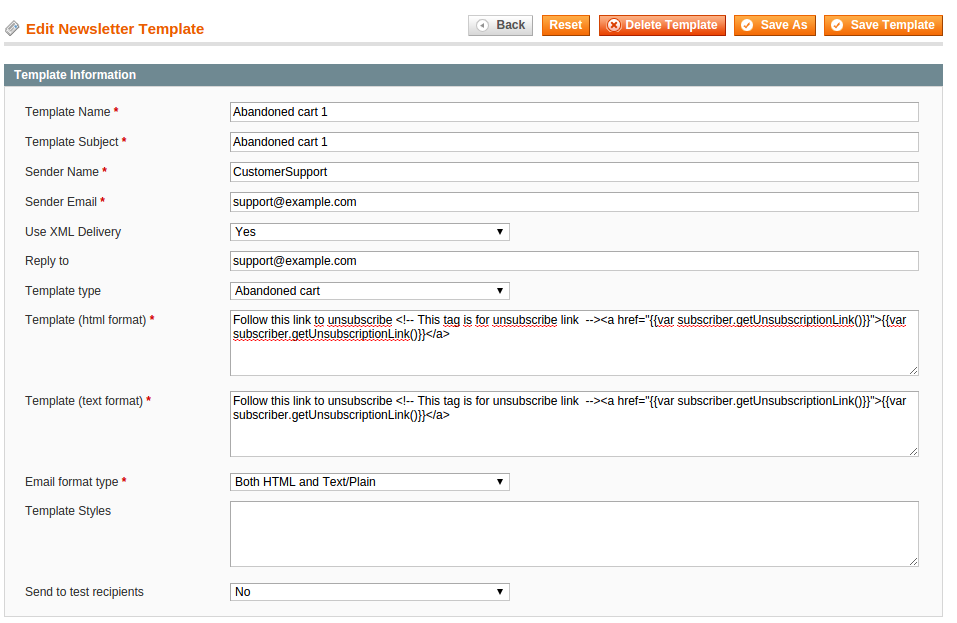
With the installation of ContactLab Plugin, the template list page has been modified with some more information relative to the XML Delivery sending process. Into the **XML Delivery Status** column, is shown a list of all XML Delivery tasks connected with this email template.

## Newsletter template attributes

To configure a new template for XML Delivery, into Newsletter template these are the attributes to specify values of four different groups:

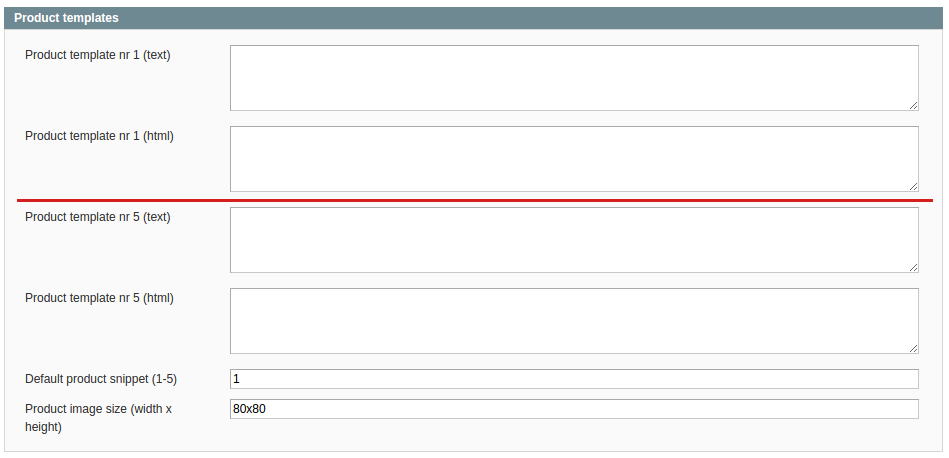
1. **Template information** – general information about the template
2. **Product templates** – template html/txt for product snippets
3. **Customer filter options** – options about how the processor will filter customer / subscribers
4. **Cron information** – options about automated execution of sending process

### Template information



|  |  |  |
| --- | --- | --- |
| Attribute | Description | Notes / Examples |
| Template Name | Name of the template | Use this field to identify this template into template list |
| Template Subject | Email subject |  |
| Sender Name/Email | Name and email of the sender |  |
| Use XML Delivery | Flag that enable the sending through XML Delivery engine | If disabled, the default Magento behavior will be executed |
| Reply to | Address for *reply* mail header field |  |
| Template type | Modifying this field, the page will show/hide custom attributes needed to manage products snippets into template and selection of customer collection for the abandoned cart / wishlist remind recurring email types. | Wishlist Remind  Abandoned Cart  Generic |
| Template (html/text) | Template text in html and/or txt | Read farther for more information |
| Email format type | Html and/or Text for template composition? |  |
| Template Styles | Css included for html formatting |  |
| Send to test recipients | Use this template to send test email? | If enabled, the cron sending is disabled, and a manual sending processing is required (read farther). |

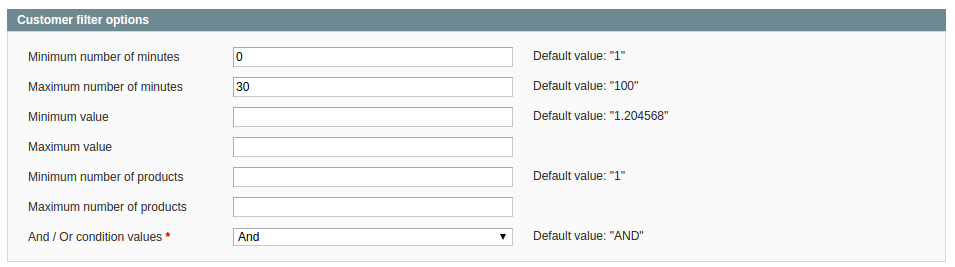
### Product templates



This section contains product template text areas from 1 to 5, with text and html, used for compile the product snippets included by the main email template. “Default product snippet” specify the default box if the number of snippets exceed the maximum of 5 snippets.

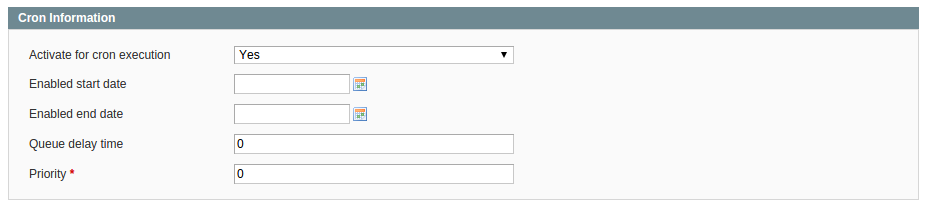
Product image size (width x height) is used for preparing the required resized images *before* that every email sending process.

### Customer filter options



|  |  |  |
| --- | --- | --- |
| Configuration | Description | Example |
| Minimum / Maximum minutes | Range of values (number of items) to consider or not a customer wishlist or cart into the possible list of recurring email recipients. | 120 / 1200 |
| Minimum / maximum amount value | Range of values (prices) to consider or not a customer wishlist or cart into the possible list of recurring email recipients. | 0.00 € / 4000.00 € |
| Minimum / maximum item number | Range of minutes to consider or not a customer wishlist or cart into the possible list of recurring email recipients, based upon the last datetime of cart/wishlist modification. | 4 / 50 |
| And / Or condition values | Consider value and number of items in and / or into the data extraction query. | And |

### Cron Information



|  |  |  |
| --- | --- | --- |
| Configuration | Description | Example |
| Activate for cron execution | Do manage this template in cron execution automatically? |  |
| Enabled start/end date | Date interval for enable/disable the template automatic sending | (optional) |
| Queue delay time | Execution offset in minutes after the queue of the template. | (optional) |
| Priority | In case of more than one template valid for Wishlist or Abandoned cart, the one with higher priority would be processed by first | 10 |

### Template text

The template text syntax is the same used by standard Magento template, so referrer to Magento documentation for more details. This is an example of dynamic content:

{{var obj.getAttribute()}}

### General attributes/object

Into ContactLab XML Delivery templates there are two specific objects (subscriber and products) that contains each customer/subscriber attributes and the related products data. Here are some usage example of these two objects:

**{{var subscriber.getUnsubscriptionLink()}}**

**{{var subscriber.getCustomerFirstname()}}**

**[...]**

**{{var subscriber.getCustomerDob()}}**

Another attribute has been added to the subscriber object:

**{{var subscriber.getProductsNumber()}}**

It contains the number of products into the products collection related to the processing customer.

This is the syntax used to print out the product template related to the nth product snippet:

**{{var products.getProduct1()}}**

**{{var products.getProduct2()}}**

**...**

**{{var products.getProductX()}}**

### Product snippets

Here are some example of how print out product attributes into the product snippets. Attributes names depends on Magento attribute set because there could be user defined attributes:

**{{var product.getSku()}}**

**{{var product.getName()}}**

**[...]**

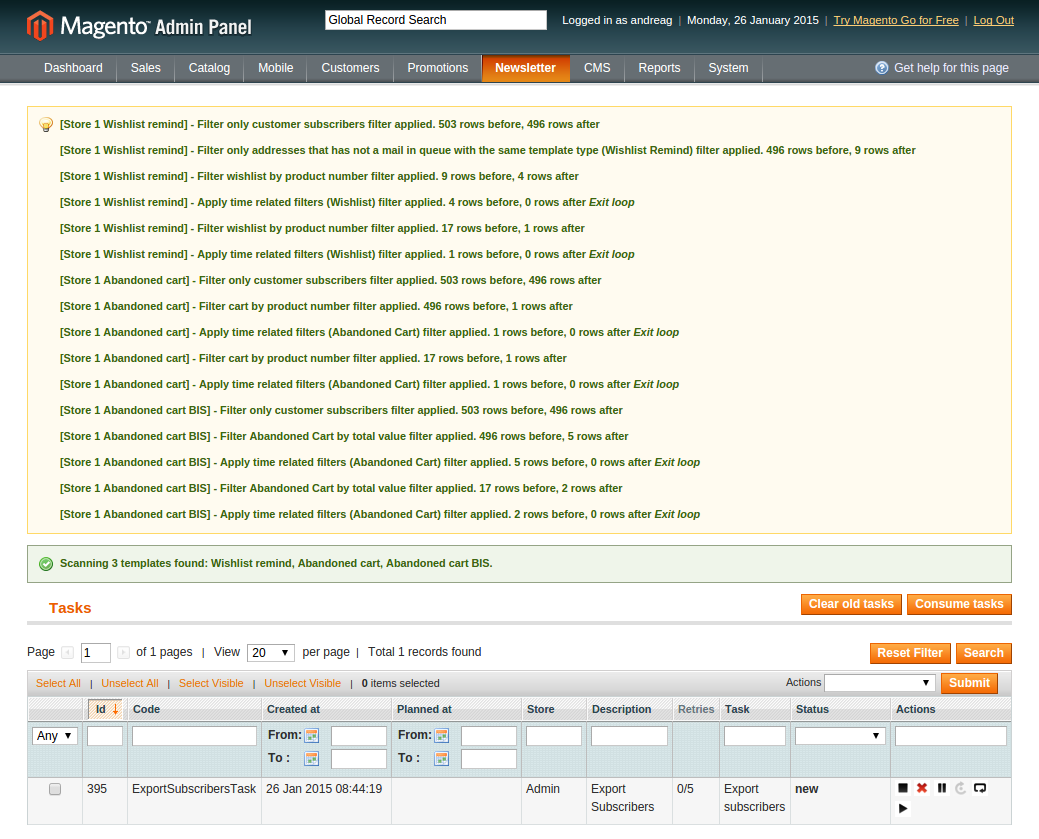
**{{var product.getQty()}}**

### Preview

Into the template list page, the last column of the table contains a set of actions. For templates that uses XML Delivery option, the customers specified into configuration for test mail addresses will be used. The product rendered into the mail will be taken from the same customers themselves.

### Debug extraction queries

Cron executed recurring email, in a production phase, would be executed automatically. During process setup, there a way to debug how the data extraction filters are working.

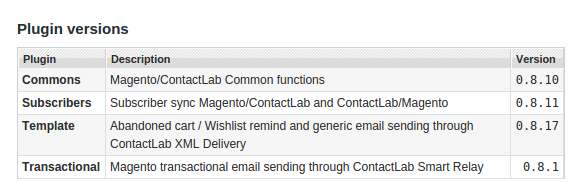


If the “Debug” option has been enabled, in the ContactLab Tasks admin page, pressing the button “Scan templates”, the system will show the list of filter, for each XML Delivery template and enabled store with the number of customer before and after applying the filter.

Writing in “Debug an email address” box an existing email, the system will run the extraction query limited to the chosen address. This will help to determine which filter has excluded the address from the collection of possible recipients.

## Show plugin version

If the enable mode has been enabled in the Plugin configuration of Magento, in the lower of ContactLab Tasks admin page there’s a box that show the current deployed version of each component of the Magento/ContactLab plugin.



## Show plugin Release Notes

Release Notes of this plugin can be viewed into the Magento backend: “Newsletter » Contactlab » Release notes” page.

