Location Intelligence
Infrastructure Asset Management

Confirm®

Confirm Task Processor

v26.30a.AM

Information in this document is subject to change without notice and does not represent a commitment on the part of the vendor or its representatives. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, without the written permission of Confirm.

© 2026 Confirm. All rights reserved.

Products named herein may be trademarks of their respective manufacturers and are hereby recognized. Trademarked names are used editorially, to the benefit of the trademark owner, with no intent to infringe on the trademark.

Open Source Attribution Notice

The Confirm suite of products contain the following open source software:

- Feature Data Objects v 3.5.0, which is licensed under GNU Lesser General Public License, Version 2.1, February 1999 with the unRAR restriction. The license can be downloaded from: **http://fdo.osgeo.org/licenceAndGovernance.html**. The source code for this software is available from **http://fdo.osgeo.org/content/fdo-350-downloads**
- MrSID software (specifically the mrsid32.dll) is used under license and is Copyright © 1995-2002, LizardTech, Inc., 1008 Western Ave., Suite 200, Seattle, WA 98104. All rights reserved. MrSID is protected by U.S. Patent No. 5,710,835. Foreign patents are pending. Unauthorized use or duplication prohibited.

  Patented technology in the Software was developed in part through a project at the Los Alamos National Laboratory, funded by the U.S. Government and managed by the University of California. The U.S. Government has reserved rights in the technology, including a non-exclusive, nontransferable, irrevocable, paid-up license to practice or have practiced throughout the world, for or on behalf of the United States, inventions covered by the patent, and has other rights under 35 U.S.C. § 200-212 and applicable implementing regulations.

  For further information, contact Lizardtech.

- NodaTime, version number 1.3.10, which is licensed under the Apache license, version number 2.0. The license can be downloaded from http://www.apache.org/licenses/LICENSE-2.0 . The source code for this software is available from http://nodatime.org/.
- Chromium Embedded Framework, version 3, which is licensed under the New BSD License. The license can be downloaded from http://opensource.org/licenses/BSD-3-Clause. The source code for this software is available from http://code.google.com/p/chromiumembedded/downloads/list.
- Xilium.CefGlue, version 3, which is licensed under the MIT License (with portions licensed under the New BSD License). The licenses can be downloaded from http://opensource.org/licenses/MIT and http://opensource.org/licenses/BSD-3-Clause. The source code for this software is available from http://xilium.bitbucket.org/cefglue/.
- D3 Data Driven Documentation, version 3.4.1, which is licensed under the New BSD License. The license can be downloaded from from https://github.com/mbostock/d3/blob/master/LICENSE. The source code for this software is available from http://d3js.org/.
- OpenLayers, version 8.1, which is licensed under the BSD 2-Clause Licence. The license which can be downloaded from https://github.com/openlayers/openlayers/blob/master/LICENSE.md The source code for this software is available from https://github.com/openlayers/openlayers.
- Proj4js, version 1+, which is licensed under the Apache License, Version 2, January 2004. The license can be downloaded from http://www.apache.org/licenses/LICENSE-2.0.html. The source code for this software is available from http://trac.osgeo.org/proj4js/.
- requireJS, version 2.1.2, which is licensed under the MIT License or the New BSD License. The license can be downloaded from https://github.com/jrburke/requirejs/blob/master/LICENSE. The source code for this software is available from http://requirejs.org/.
- Apache Cordova, version 11.1.0, which is licensed under the Apache License, Version 2, January 2004. The license can be downloaded from http://www.apache.org/licenses/LICENSE-2.0.html. The source code for this software is available from http://phonegap.com/download/.
- Xilium.CefGlue, version 75.1, which is unlicensed. The source code for this software is available from https://gitlab.com/xiliumhq/chromiumembedded/cefglue.

- Chromium Embedded Framework, version 75.0, which is licensed according to the following criteria:

  Copyright (c) 2008-2014 Marshall A. Greenblatt. Portions Copyright (c) 2006-2009 Google Inc. All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  * Neither the name of Google Inc. nor the name Chromium Embedded Framework nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

  The source code for this software is available from http://opensource.spotify.com/cefbuilds/index.html#

October 02, 2026

Table of Contents

Confirm Task Processor

Setup 6
- Pre-requisites 6
- Installation 6
- Configuration 10

Agent Status Life cycle 15

FAQ 16

Agents 19
- Alerts and Notifications Agent 19
- Contractor Access XML Agent 20
- CRM Agent 23
- Enquiry FPN Delivery 24
- Financial Interface XML Agent 24
- Full Text Index Agent 26
- Import Address Agent 26
- Import Anything Agent 27
- Photo Upload Agent 29
- Referential Integrity Agent 29
- Repeating Jobs 33
- Scheduled Reporting Agent / Dashboard Agent 35
- Street Works Transfer 36
- Street Manager Notification Agent 43
- Connector Agent 44

Confirm Task Processor

The Confirm Task Processor is an individual component within the Confirm release package and is installed as a Windows service through its installer. It supports processing of the Agents that are available within the 'Scheduled Tasks' screen.

In this section

Setup 6
Agent Status Life cycle 15
FAQ 16
Agents 19

Setup

Pre-requisites

The following steps should be performed before commencing installation of the Confirm Task Processor. The pre-requisites are supplied on the Confirm Media unless otherwise stated:-

- Microsoft .NET Framework 3.5 SP1
- Microsoft .NET Framework 4.8
- Oracle or SQL Server Database client drivers (not supplied on Confirm Media)
- The MapXtreme 9.1 Redistributable must also be installed on the machine

Note: For 64 bit environments, the MapXtreme install requires both the 32 bit and 64 bit versions of Visual C++ Redistributables to be installed.

These prerequisites are available within the Prerequisites directory of the Confirm media.

The desktop heap of the Task Processor machine should be increased. This is optional but recommended. The Task Processor uses a number of background processes named 'TaskProcessorService.exe'. These are launched by the Task Processor Windows Service to process scheduled tasks. Windows limits the amount of Desktop Heap these processes have access to, which can cause problems in high load situations where multiple copies of the 'TaskProcessorService.exe' processes are run in parallel. To ensure stable running of the Task Processor under load please make a change to the following registry setting on your server:

1. Run RegEdit as an Administrator and browse to the following registry location: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\SubSystem
2. Select the 'Windows' key within this location and edit its value.
3. Find within the value the string 'SharedSection=' which is followed by 3 numbers such as 'SharedSection=1024,20480,768'.
4. Edit the third value (in the above example this will be 768) to be 4096, so our example will look like 'SharedSection=1024,20480,4096'. Do not change any other values in this string, even if they differ from those in this example.
5. Save your changes by clicking 'OK'. The server will now need to be restarted for the changes to take effect.

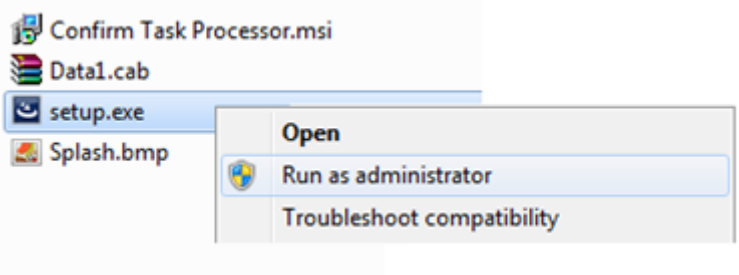
Once the Task Processor pre-requisites are installed Task Processor **Installation** needs to be done.

Installation

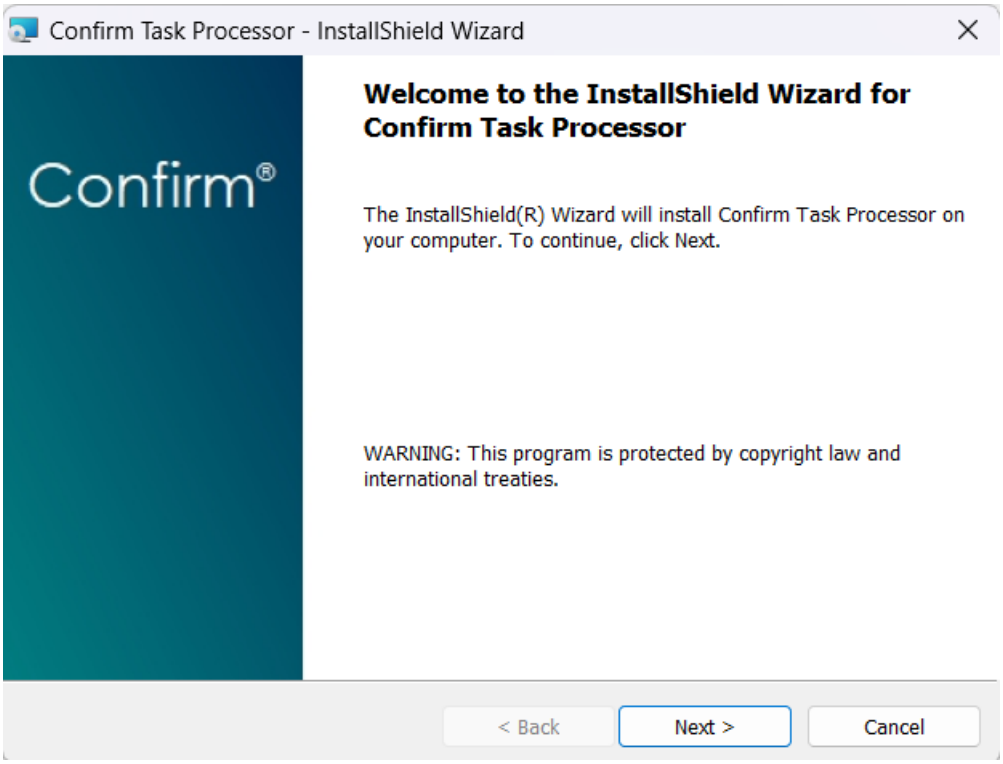
Note: We recommend running the Confirm Task Processor on multiple core machines as it helps improving the processing of tasks executing in parallel.

The Confirm Task Processor is available as a separate installer. It should be installed with administrator privileges on the machine. Follow the steps below to install the Task Processor:

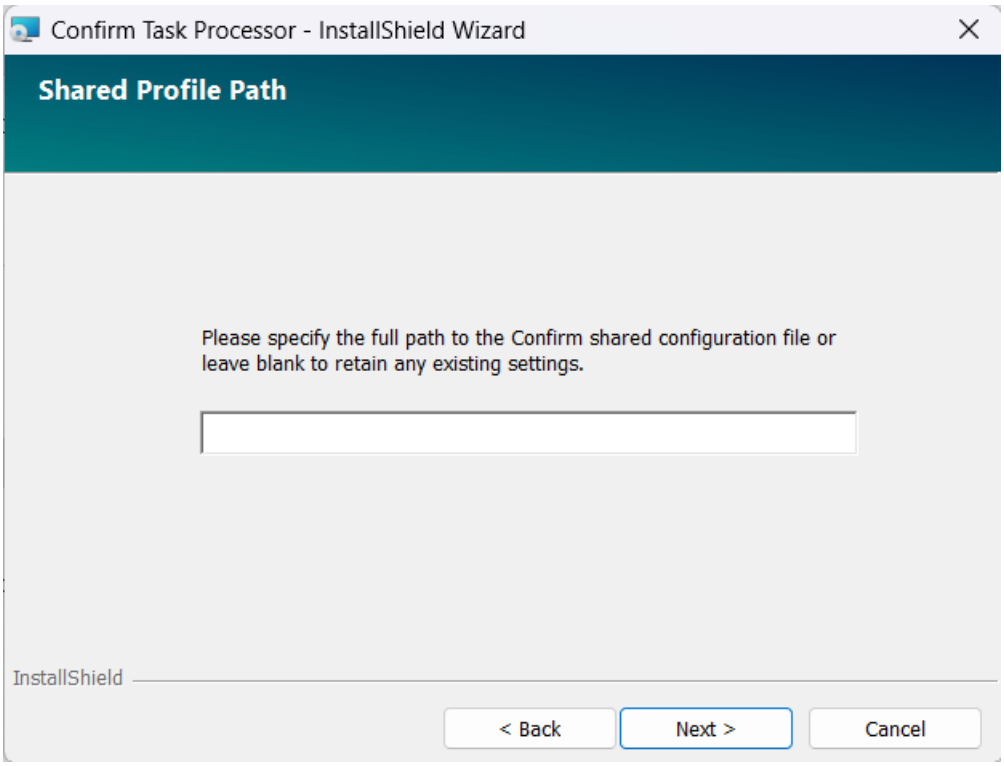
1. Launch the installation using 'Run as administrator' option from 'Setup.exe' from the Task Processor folder.



2. You will see the Confirm Task Processor Installer Welcome Screen. Click 'Next'.

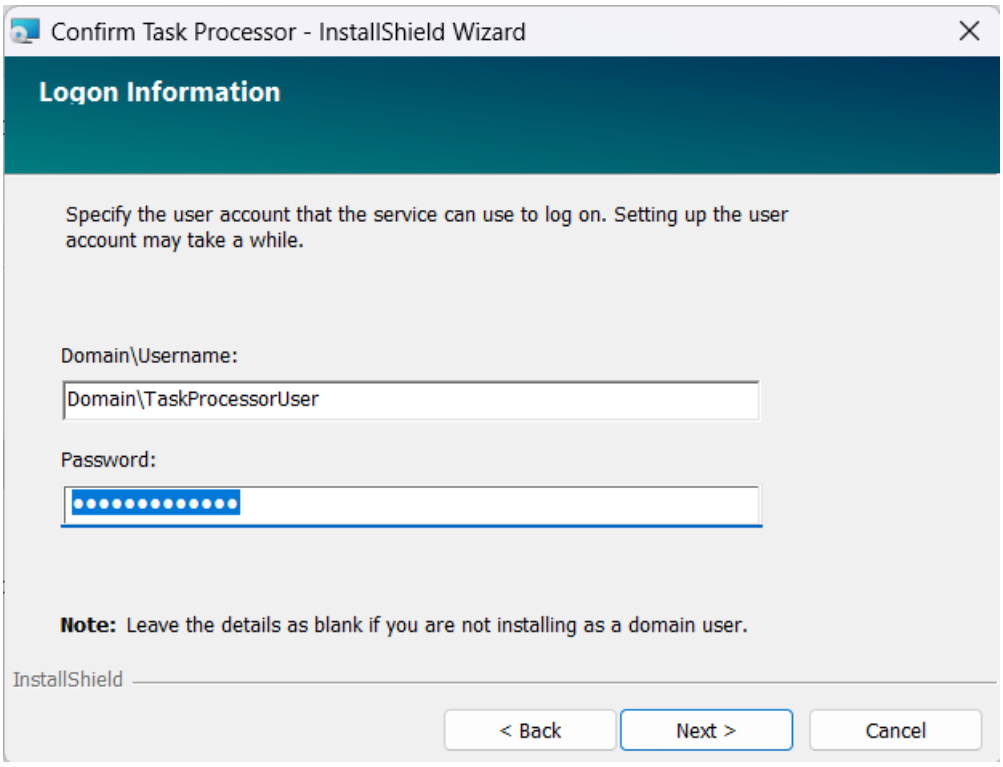


3. The Task Processor uses the configuration file in the same way as the Confirm Client. Enter the path to the configuration file (as shown in the screenshot below) ‘path\filename.cfg'. Click 'Next'.



Note: It is recommended to use the same (or copy of) configuration file as being used by Confirm client. Any typographical errors in manually edited configuration files may prevent the Task Processor from running properly.

4. Specify a domain user account. This account is required for the Task Processor service to access shared files and folders on the network. If you are not logged in as domain user, skip this step and you may set the domain user account as part of **Configuration**.



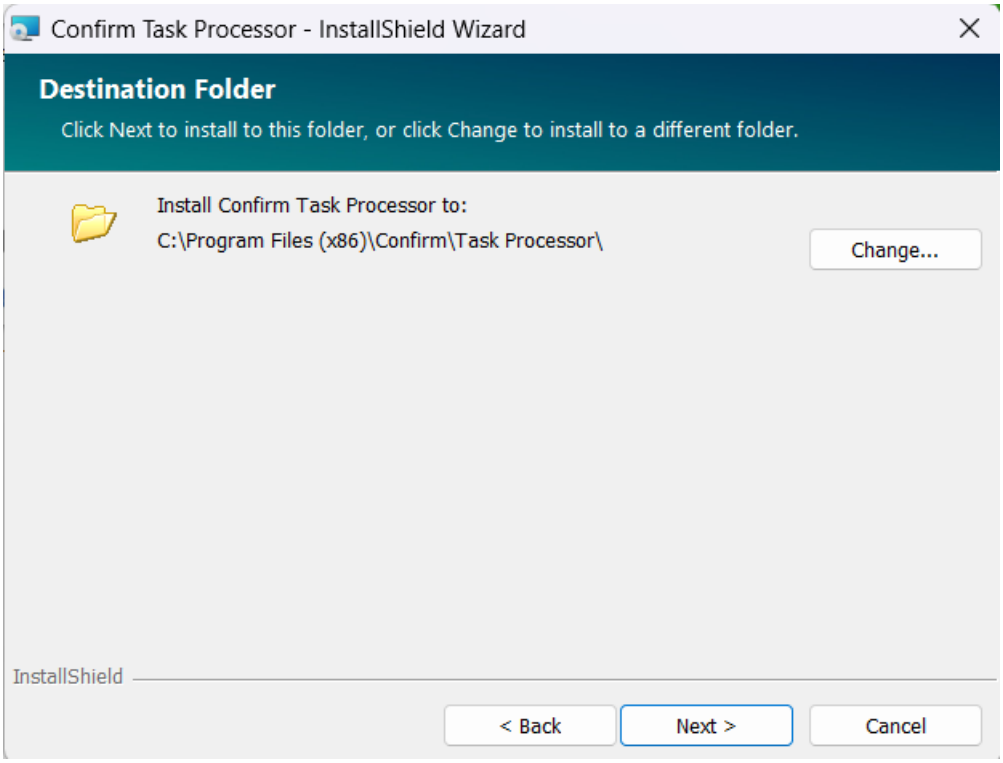
It is recommended to use a unique domain user whose password is set to never get locked or expire. This user should only be used to run the Confirm Task Processor service.

Note: This User should have same regional settings like Date Format, Timezone as that of Task Processor machine.

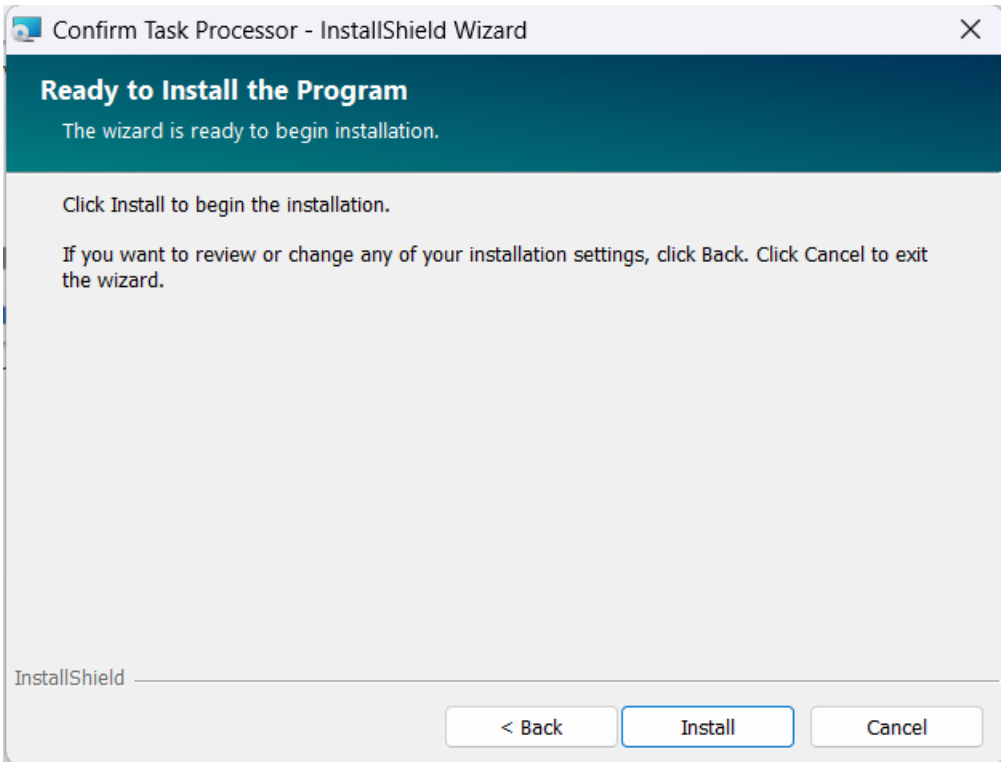
5. Select your preferred installation location.

   If you have not previously enabled the Confirm User Experience Improvement Program (UEIP) on the target machine, you will see a checkbox allowing you to enable it now. Click the ‘UEIP Details’ button for full details of this program.

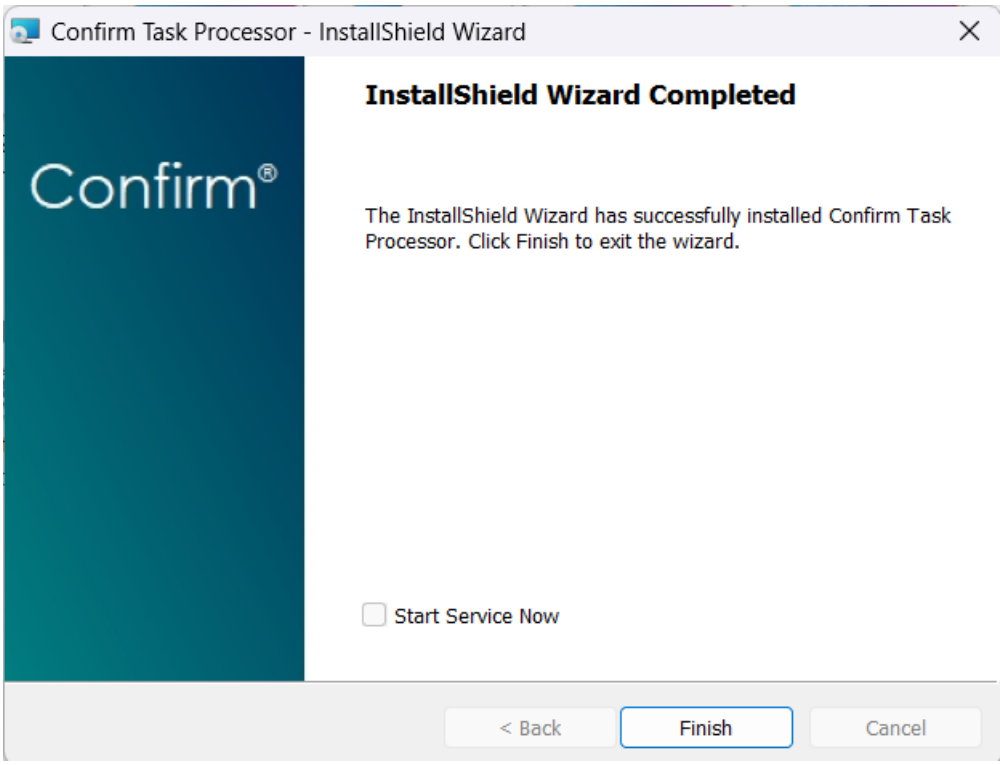
   Click 'Next' to continue.



6. Click 'Install' to begin the installation process.



7. Once the installer has completed, click ‘Finish’ to complete the installation.The Task Processor service is created, and is stopped by default. It requires manual configuration, before it is started. In case of upgrade, ‘Start Service Now’ is available can be checked to start the service immediately.



Once the Task Processor installation is complete Task Processor **Configuration** needs to be done. In case of upgrade, no configuration is required.

Configuration

Configuring Provider Logging

The Task Processor can be configured to produce provider logging if required. To enable, follow these steps:

- Open the registry.
- In the registry go to this location:

  32-bit Machine:

  'HKEY_LOCAL_MACHINE\SOFTWARE\Confirm\Task Processor'

  64-bit Machine:

  'HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Confirm\Task Processor'

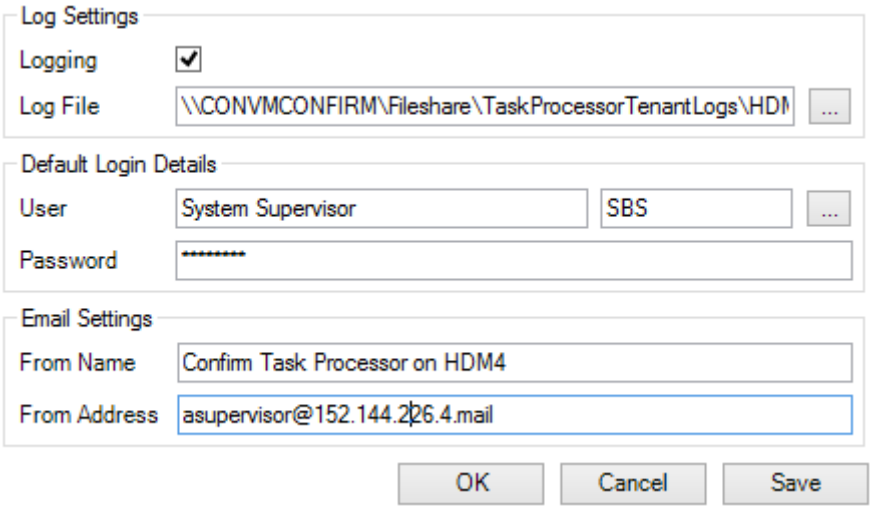
- Add values to the following registry keys.

Registry Key	Description
LoggingEnabled	Defaults as 'No'. When logging is set to 'Yes', all provider level logging will be written to the log file. When disabled, logging is cached and only written when an error occurs.
LogFile	Sets the path and filename for the Confirm Task Processor logging.

Note: These logging settings are common for all databases that are being serviced by the Task Processor. When upgrading the Task Processor existing entries are retained.

Configuring Database Logging

To enable logging for each database go to the 'Task Processor Settings' screen in the Confirm Client for the tenant you wish to configure. Tenant logging can be turned on or off in this screen and the log file name and location can be set.



Configuring Login Details

The Task Processor requires the User ID and Password of a Confirm User in order to run successfully. These need to be populated on the Task Processor Settings screen in the Confirm Client.

If the Task Processor is being configured to multiple database connections, this needs to be done for each database.

Configuring Email Alerts

The Confirm Task Processor requires the use of an email account to send alert emails. Email alerts regarding specific agent tasks can be sent to an email address set up in the 'Scheduled Tasks' screen for that agent.

If no agent tasks have been configured to send email alerts the default email behaviour will be used to alert the System Supervisor (the individual responsible for maintaining the Confirm Task Processor) if there is a problem with the Confirm Task Processor itself.

To set up email alerts follow these steps:

- Open the registry.
- Back the existing registry settings.
- In the registry go to this location:

  32-bit Machine:

  'HKEY_LOCAL_MACHINE\SOFTWARE\Confirm\Task Processor'

  64-bit Machine:

  'HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Confirm\Task Processor'

- Add values to the following registry keys.

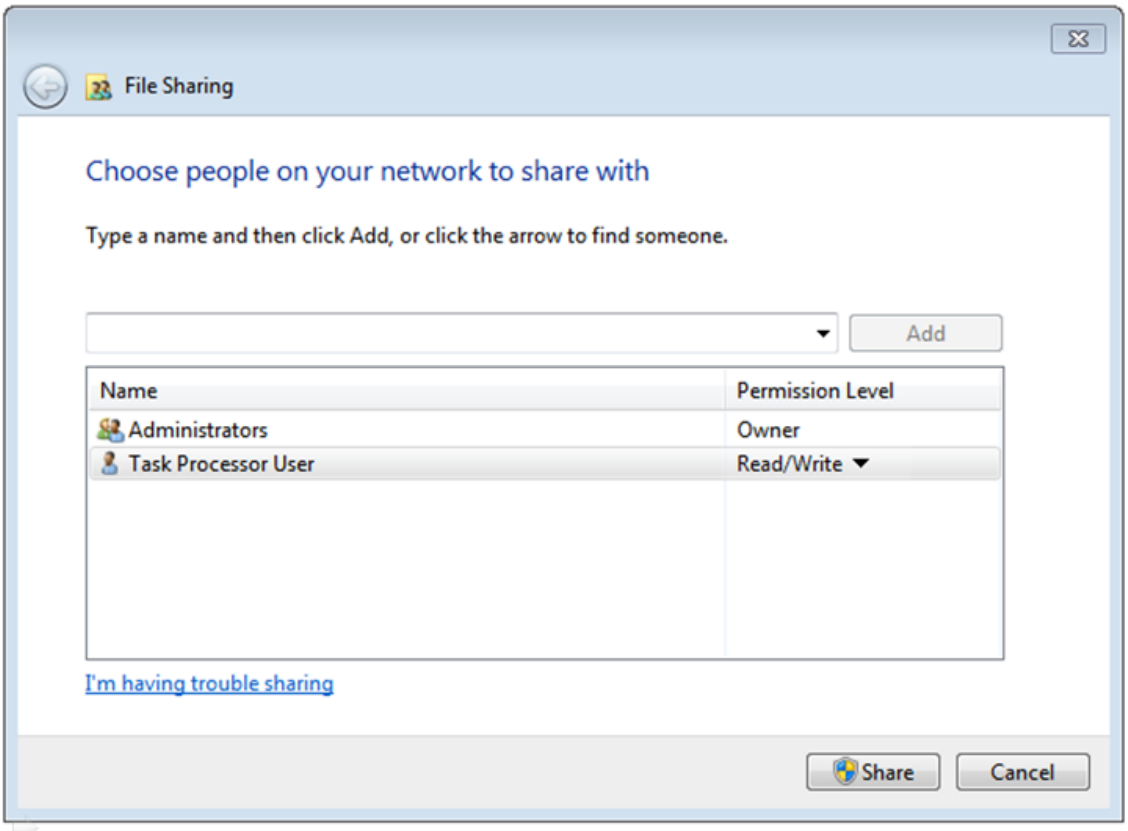
Registry Key	Description
MailRequired	This option toggles email on or off. To be set as 'Yes' if mailing is required. **Note:** All the following settings in this table are mandatory if MailRequired is set to 'Yes'
SmtpServer	Set to either the mail server name or IP address. To connect the Confirm Task Processor to an SMTP server which is running on a non standard port number (25) then append ':portnumber' to the server name / IP address
SmtpDomain	Optional. Required only if authentication is needed to access the SMTP Server.
SmtpSecured	Defaults as 'No'. When set to Yes, this setting allows all emails to be sent over SSL\TLS protolcol. This requires SMTPServer registry key to be set to a secured Port (like 587). The destination email server should be configured for SSL\TLS protolcol.
SmtpUserName	Optional. Required only if authentication is needed to access the SMTP Server.
SmtpUserPassword	Optional. Required only if authentication is needed to access the SMTP Server.
SupervisorEmail	Set to an email address which will receive emails should there be a problem with the Confirm Task Processor. This email address is normally an IT representative or principal user whom is responsible for maintaining the Confirm Task Processor.
UnhandledEmail	Set to an email address that will receive emails when an error that is not handled by Task Processor. If this setting is left empty, Task Processor will not send unhandled errors in an email.
ForwardEmail	Set to an email address where you would like to forward any read (by Confirm Task Processor) email to.

- An email address needs to be set for the Task Processor which will be the ‘from’ address whenever the Task Processor sends an email alert. This email address is configured in the ‘Task Processor Settings’ screen in the Confirm Client.

Note: The Task Processor only supports SMTP for sending mail.

Folder Access Permissions

The Task Processor works in a non-interactive mode. This means that the user configured for the Task Processor service needs to have read/write permissions to all folders where the Task Processor will need to read data from or write data to. If such a folder resides on the machine where Task Processor is installed, access should be provided on that folder to a Domain User. If the folder resides on a network location, access should be provided on that folder to a Domain User.



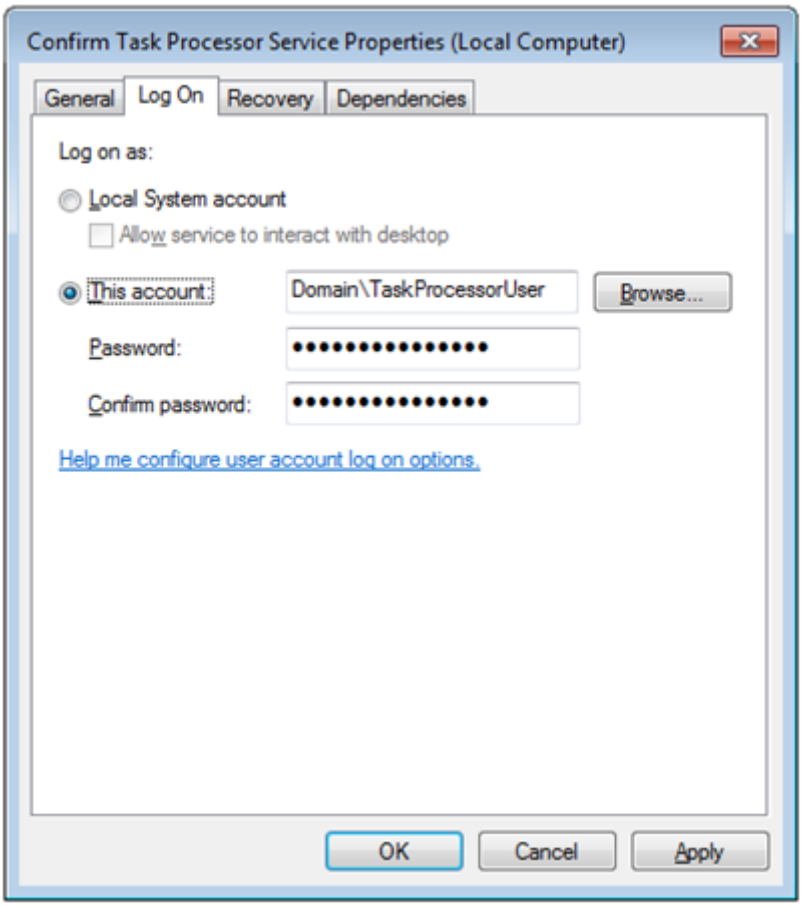
The following are some examples folders which need to be provided permissions to the Task Processor user:

- Where the Task Processors configuration file is accessed from.
- Log file where Task Processor will write logs (see logging section)
- All folders where individual agents need to read corresponding files or data.
- All folders where individual agents need to write corresponding files or data.

Configure Task Processor LogOn details

This needs to be done only if domain user account has not been set at the time of installation. To assign a user account to the Task Processor service:

- Open a prompt and type services.msc followed by <return> to open the services screen.
- Double click on the Confirm Task Processor service.
- Click on the ‘Log On’ tab and set the user account to be used together with its password.
- Click on ‘Apply’ and then close the screen.



It is recommended to use a unique domain user whose password is set to never get locked or expire. This user should only be used to run the Confirm Task Processor service.

Starting the Task Processor

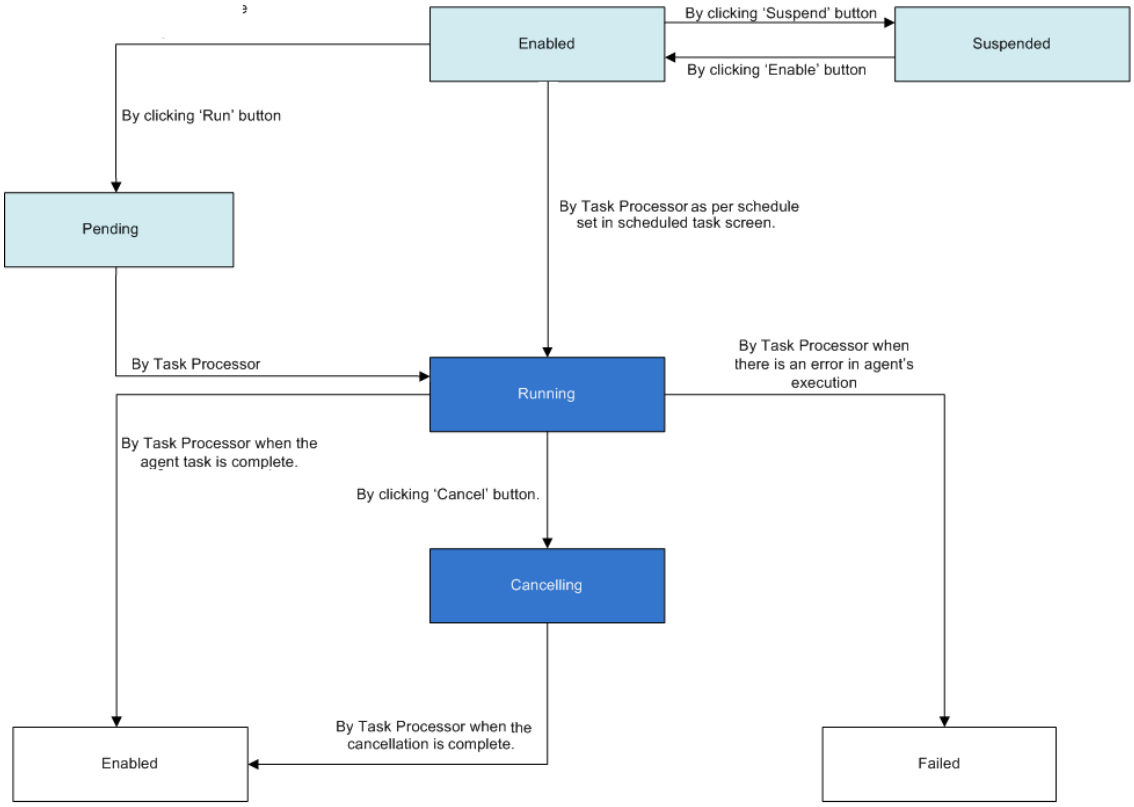
Once Task Processor has been configured follow these steps to start the Task Processor service:

- Open a command prompt and type services.msc followed by <return> to open the services screen.
- Find the 'Confirm Task Processor' service and start it.

Note: The Task Processor will only process agent tasks scheduled in the 'Scheduled Tasks' screen within the Confirm Client.

Agent Status Life cycle

This page illustrates when the agent status changes from one status to another.



FAQ

Q: How do I upgrade my Task Processor?

A: Please follow the steps as provided in the order below:

- Wait for all the running agents to complete their run. There should be no running tasks.
- Stop the Task Processor Windows Service.
- Run the Task Processor installer.
- Start the Task Processor Web Service manually if not started using the installer.

Q: How do I refresh my development DB with Production DB?

A: Please follow the steps as provided in the order below:

- Wait for all the running agents to complete their run and then suspend all Agent Tasks in 'Scheduled Tasks' screen.
- Stop the Task Processor Windows Service.
- Take the backup of the database and then restrore it on the new server.

Q: Can I upgrade Task Processor in silent mode?

A: It is recommended to upgrade using setup.exe . However, in case silent upgrade is required, follow the steps provided below:

- Prepare the silent upgrade command:
  - Copy the following command in any text editor (like Notepad):

    setup.exe /s /v"/qb! IS_NET_API_LOGON_USERNAME=<Domain\Task Processor User> IS_NET_API_LOGON_PASSWORD=<Password>"

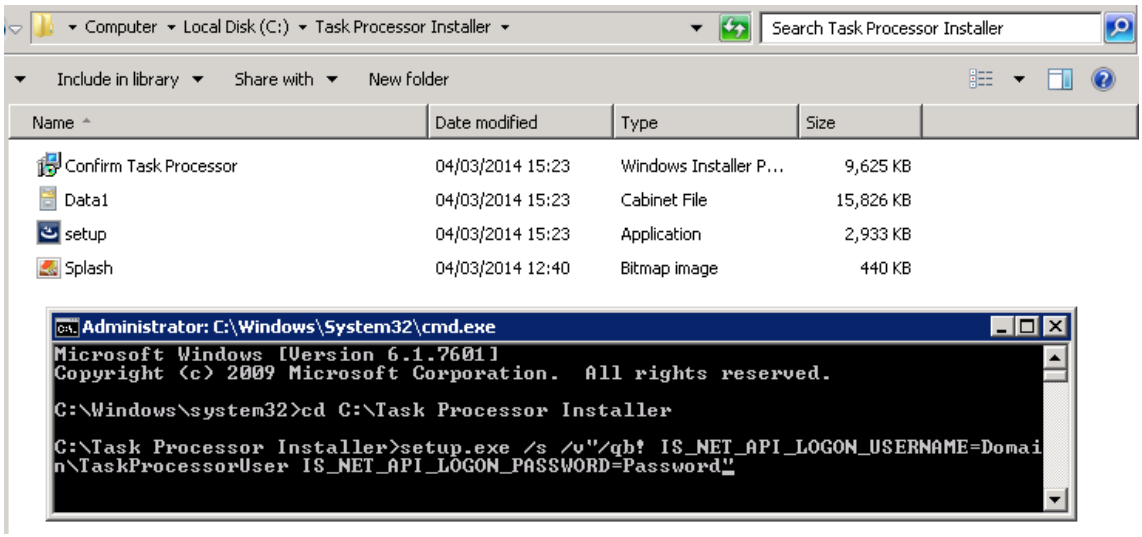
  - Replace <Domain\Task Processor User> with domain and user account used by Task Processor service.
  - Replace <Password> with actual password
- Copy the Task Processor setup installation folder at the local drive on installation machine.
- Open command prompt as administrator.
  - Change the directory to point at local installation folder
  - Run the command prepared in step 1 to perform silent upgrade.

Q: What do I do if Agent is stuck in Cancelling Status?

A: Please follow the steps as provided in the order below:

- Wait for all running agents to complete their run..
- Restart the Task Processor Service. This should change the status of the Agent stuck in 'Cancelling' status to 'Failed'.
- Task Processor will pick the 'Failed' Agent, as per the schedule and start processing.

Screen-shot below is an example of the steps described above.



Q: I use mapped drives to read and write to network locations. These mapped drives are used by multiple agent tasks to process data. Will they work with the Task Processor?

A: It is recommended to use UNC (fully qualified) paths when referring to network locations. The Task Processor can read/write data from a mapped drive only if a setting has been added in the **Mapped Drive Settings screen**.

Q: Some of my agents are stuck at 1%, while others are running fine. What should I do?

A: It is recommended to use the same (or copy of) configuration file as being used by Confirm Client. Any typographical errors in manually edited configuration file may prevent the Task Processor from running properly.

It is recommended to create a new configuration file from Database Settings screen and use that for running the Task Processor.

Q: I have multiple databases (like live, dev and test) to be serviced by Task Processor(s). How can I allocate different processing power to each database?

A: Following entry in Task Processor configuration file will limit the number of tasks that can run in parallel for a database:

MaxTasks=< number>

If this entry is not specified for any database, Task Processor(s) will use the default limit of 10 parallel tasks for that database.

Q: How can I ensure that Task Processor(s) always have capacity to process critical tasks?

A: Following entry in Task Processor configuration file will reserve the number of tasks that can run only if they are critical:

ReservedCriticalTasks=< number>

This limit is a subset of total tasks that can run in parallel for a database and is limited by it. If this entry is not specified for any database, Task Processor(s) will use the default number of 0 reserve slot for critical tasks for that database.

Q: I am using the Confirm System Agent. What steps should I follow to install the Confirm Task Processor?

A: Please follow the steps as provided in the order below:

- Upgrade the Confirm Client to V14 or above.
- Wait for agent tasks that are running to complete and then suspend all agent tasks in ‘Scheduled Tasks’ screen (applicable only if upgrading from V12.20).
- Stop the System Agent.

- Run the Task Processor installer.
- Configure the Task Processor.
- Enable the agent tasks in the ‘Scheduled Tasks’ screen.

Q: I am writing a Data Source that looks at the scheduled_task table but the times aren't consistent with what I see on the screen?

A: Date / times in the scheduled_task table are stored in Coordinated Universal Time (UTC). This is to ensure tasks run at the appropriate time even if your database server is setup with a different time zone from your Task Processor server.

Agents

The following sections outline the Agents can be run from the Confirm Task Processor.

Alerts and Notifications Agent

Overview

This Agent organises emails and notifications that are used to alert a user to a particular event which has happened or is about to happen in Confirm. Alerts can be delivered to an email address, a Confirm Inbox or both.

Alerts can be setup for the following entities:

- Enquiries
- Jobs
- Offences and Fixed Penalty Notices
- Street Works

Operation

On each scheduled run, the Agent processes each of the Alert entities in the table below for which any Alert Types have been configured, in the order shown in the table.

There are two stages to the processing, and each stage is carried out in full for an entity before moving on to the next one.

Alert Entity	Note
Job	Estimated Completion Date Alert Types are checked and Pending Alert records generated whenever a Job is created or updated. However the first stage below applies to all other Job Alert Types.
Enquiry and Commitment	
Offence	Offence Alert Types are checked and Pending Alert records generated whenever an Offence is created or updated. Hence the first stage below is skipped for Offence Alert Types.
Street Works Inspection	
Street Works Occupancy	
Street Works Reinstatement	
Street Works Comment	
Street Works Emergency Notice	
Street Works Notice Sequencing	
Street Works Notice	

First Stage - Generating Pending Alerts

During this stage, any record that has been changed since the last run is checked against the criteria of relevant Alert Types (in the case of Job Alerts, this would mean any new Job Status Log records created since the last run).

Wherever the criteria are met, a Pending Alert record is saved in the database.

Following this all the records are marked as having been checked so that they can be ignored on subsequent runs.

Note that if there are no Alert Types for a particular entity, then this stage will be skipped and so records will not be flagged in the database as having been checked. This means that when a new type of Alert is configured in an existing system, there will potentially be a large backlog of historical records that the Agent will attempt to process. For example in a system where only Enquiry Alert Types are configured, none of the Job Status Logs or Street Works Notices in the database will have been marked as checked by the Agent.

This is a common cause for the Alerts and Notifications Agent to fail to process.

If this problem occurs and prevents the Agent from running successfully, or if you wish to start using a new type of Alert and want the historical backlog to be ignored, please contact Confirm support to obtain SQL to flag the historical backlog as having been checked.

Second Stage - Sending Alerts

During this stage all Pending Alerts are checked to see if there are any that are due to be sent.

If there are, each is processed in turn. The Alert Message is constructed and sent to the configured destination, and then the Pending Alert record is deleted.

Configuration

This section details how to configure the Alerts and Notifications Agent.

1. Check that the Alert Modules you require are enabled.
2. Ensure that the agent is scheduled as a Task in the **Scheduled Tasks** screen within the Confirm client.
3. Setup the Alert Types that are required:
   - For Enquiries (module 1420) go to the **Enquiry Alert Type** lookup.
   - For Jobs (module 0785) go to the **Job Alert Type** lookup.
   - For Offences and Fixed Penalties (module 2710) go to the **Offence and Fixed Penalty Notice Alert Type** lookup.
   - For Street Works (module 3520) go to the **Street Works Alert Type** lookup.
4. Ensure that all users to be used with alerts have their 'Message Preference' field set in the **User Security** lookup.
5. Check that all specified email addresses are valid.
6. Ensure the Task Processor is configured and running.

Contractor Access XML Agent

This Agent provides an automated method of interfacing between Confirm and multiple Contractor systems. The Agent can be used for the following:

- Send requests for quotes for work from the Contractor.
- Receive quotes from the Contractor (items and quantities).
- Send Committed Works Orders to the Contractor for work to be carried out.
- Receive Job Status Updates on completion of work or at interim stages.
- Receive Payment Requests for interim or final payment.
- Send Payment confirmation.

Information is transferred from Confirm to the Contractor systems by means of text files or requests to a Web Service, and can be received from Contractor systems by means of text files in a variety of formats.

Operation

On each scheduled run, the Agent processes all configured Contractors in turn.

It first of all checks whether there are any Jobs, Works Orders, Variation Orders or Payments for the Contractor awaiting export, and if so exports them to each configured Export Transformation for that Contractor.

It then checks whether there are files in any of the import folders configured for the Contractor, and if so imports each one in turn.

Export

The Agent extracts information relevant to the Contractor from the Confirm database and creates an XML structure containing that information, to be sent to the Contractor system. The following types of record are included in the XML data:

- Jobs at an ‘Export to Contractor’ Status, and with a preferred Contract which is assigned to this Contractor.
- Committed Works Orders.
- Committed Variation Orders.
- Committed Payments.

The XML data is then processed by all the Export Transformations configured for the Contractor. Each of these may optionally transform the data using an XSL stylesheet (e.g. to select which parts of the data to output, or to define the format of that output), and determines whether the transformed data is exported to a file or to a Web Service.

Each export is recorded in the database with a Status of ‘Successful’ or ‘Failed’ and other information such as details of response from the destination Web Service. Previous exports can be seen on the Agent Export Log screen (see below).

Note that an export to a Web Service which fails due to a receiver error (for example the Web Service is currently down) will be set to a ‘Pending Resend’ Status, providing it has not already been sent 10 times. If it has reached this limit, it will be set to a ‘Failed’ Status.

Once all new data has been exported, the Agent will then attempt to resend any previous exports that are at a ‘Pending Resend’ Status. These could be failed exports to a Web Service as mentioned above, or exports manually selected to be resent by a User.

Import

The Agent can also import data from a Contractor system in a variety of file formats. The following data can be imported:

- Job Status Updates.
- Job Items.
- Job Document Links.

- Variation Orders.
- Payment Requests.

The Agent first checks for any files in the Import Directory specified on the Additional tab of the Contractor screen. Files imported from the Import Directory must be in the tab delimited format defined in the Contractor Access XML Agent Specification.

The Agent then checks each configured Import Transformation for the Contractor to see whether files are present in its Input Directory. If the Import Transformation has a Stylesheet specified, then the transformed file must meet the specification.

Agent Export Log

The Agent Export Log screen can be used from within Confirm client to view the status of recent or historic exports, and for example to see response detail to determine reasons for failure. It also allows previous exports to be flagged to be resent next time the Agent runs, and anything currently flagged to be resent to be cancelled.

See Agent Export Log for more details.

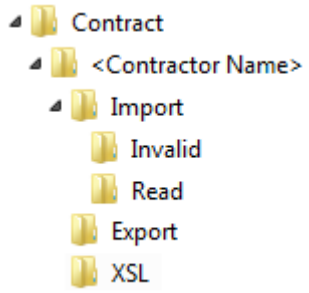
Specification

The **Specification** section for this agent provides more details on file format and structure.

Configuration

This section details how to configure the Contractor Access XML Agent.

1. Check that the Contract Access XML Agent Module (1620) is enabled.
2. Ensure that the agent is scheduled as a Task in the **Scheduled Tasks** screen within the Confirm client.
3. Setup any Import Transformations required. To do this go to the **Contractor Lookup** and retrieve the Contractor you wish to configure. Then click on the **Import** button to access the Import Transformation screen for that Contractor.
4. Setup any Export Transformations required. To do this go to the **Contractor Lookup** and retrieve the Contractor you wish to configure. Then click on the **Export** button to access the Export Transformation screen for that Contractor.
5. Authorise Contractors to be able to change a Jobs Status via an import file. This is achieved via the **Job Status Lookup**. To authorise a Job Status for a Contractor retrieve the Job Status in the lookup and tick the ‘Available to Contractor’ check box.
6. Create the directory structure on the Confirm Task Processor server for imports and exports. The following shows the preferred directory structure for the Contractor Access XML Agent (where '<Contractor Name>' denotes a separate sub-directory for each Contractor):



7. Ensure the Task Processor is configured and running

CRM Agent

Overview

This Agent is used to export newly created Enquiries and Enquiry updates to a Web Service.

If new customer information is added to an Enquiry then this will also be exported with the Enquiry.

Operation

On each scheduled run, the Agent checks whether a new Enquiry has been created or updated and if customer details have been applied or amended against an Enquiry. These details will then be exported by the CRM Agent to a configured Web Service.

The Web Service may be a client specific plug-in that transforms the data into a format supported by the Clients Web Service.

The following changes to an Enquiry will trigger an export of the information after the next agent run

- A new Enquiry has been created.
- An existing Customer has been added to an Enquiry.
- An Enquiry has been updated.
- A new Customer has been added to an Enquiry.

If the export fails for any reason (e.g. Web Service is down) then the export will be retried on the next agent run. Newer exports will be given priority.

It is possible to limit the number of exports the Agent will attempt to send on each run in the **CRM Agent System Settings** screen. When this limit is reached any Enquiries not yet exported will be sent on the next run of the Agent. This prevents the Agent from attempting to send a large number of Enquires that may have built up if the Agent has not been able to run for some time.

Export Restrictions

The following can restrict or prevent Enquiry information being exported:

- The Export flag on Enquiry Subject should be checked.
- The configured Confirm users data security access to the Enquiry Subject.
- Enquiry information will not be exported to the same external system that the Enquiry or update originated from.

Configuration

This section details how to configure the CRM Agent.

1. Check that the CRM Agent Module (1437) is enabled.
2. Ensure that the agent is scheduled as a Task in the **Scheduled Tasks** screen within the Confirm client.
3. Configure the details of the Web Service that will receive the exported Enquiry information. This is achieved via the **CRM Agent System Settings** screen.
4. Ensure that the configured Web Service is configured and running.
5. Ensure the Task Processor is configured and running.

Enquiry FPN Delivery

Overview

Enquiry FPN Delivery is a Scheduled Task that is responsible for sending Fixed Penalty Notices for Enquiries via email to their recipients.

Operation

The Enquiry FPN Delivery schedule will attempt to send FPN's via email to the specified email address. Penalties will only be sent once, if this fails then they will be reset to a Valid status. This allows the correction of email addresses should they have been entered incorrectly.

Configuration

This section details how to configure the Enquiry Fixed Penalty Delivery.

1. Ensure the Enquiry Fixed Penalty Delivery is scheduled as a Task in the **Scheduled Tasks** screen within the Confirm client.
2. Ensure the **FPN Email** settings have been specified in the **Penalty System Settings** screen.
3. Ensure the Task Processor has been installed and configured with email settings specified in the task processor configuration file (see below).

FPN Email Settings

The following settings define the structure and contents of FPN emails, and any error emails that are generated:

- Update TaskProcessorService.exe.config file, located in the Confirm Task Processor installed directory (e.g. C:\Program Files\Confirm\Task Processor)

  Note: For all of the below steps, ensure that you change the entry within quotes in the value="".

  - *<add key="SMTPHostIPAddress" value="127.0.0.1" />*

    Set to the IP Address of your SMTP server.
  - *<add key="SMTPHostPort" value="25" />*

    Set to the port number your SMTP server is running on.

Financial Interface XML Agent

Overview

This Agent is used to import authorisation on pending Orders from an external financial system.

It can also be used to export the following to external financial systems in XML format:

- Pending Order details.
- Committed Order details.
- Committed Variation details.
- Committed Payment details.

Operation

On each scheduled run, the Agent processes all configured Financial XML Import and Export transformations in turn.

It first checks for any export transformations configured. If there are any new Committed Works Orders, Variation Orders or Payment Orders due to be exported these are exported to their appropriate export transformation.

Then a check is made to see if there are any files in any of the import folders configured in the import transformations, and if so imports each one in turn.

Export

When a Works Order, Variation Order or Payment Order is committed in Confirm this acts as a trigger that the details need to be exported. The Agent extracts the information and exports it to the relevant directory and file name specified in the export transformation.

Export transformations can have a stylesheet applied to regulate the information that is exported to that transformation.

Import

Only one type of import file is received via the agent and this is an authorisation file. Contract Orders requiring this authorisation will need to have their lookup configured accordingly (see the **Configuration** section of this agent for more details).

Once the Contract is setup to require authorisation any Orders that are committed against this Contract will be set to a 'pending' status and will require an authorisation file to be imported. When a valid import file is received for the Order it will then be set to a 'committed' status.

When the import file is received the Contract and the Order Number are checked. If the Order is in a ‘pending’ status and a ‘PO Number’ has been specified on the import file, the Order will be set to a ‘committed’ status.

If a ‘PO Number’ is not specified on the import file and the Order is at a ‘Committed’ status, the Order will be set back to a ‘pending’ status.

Specification

The **Specification** section for this agent provides more details on file format and structure.

Configuration

This section details how to configure the Financial Interface XML Agent.

1. Check that the Financial System Agent Module(1710) is enabled.
2. Ensure that the agent is scheduled as a Task in the **Scheduled Tasks** screen within the Confirm client.
3. If financial authorisation is to be imported via the agent ensure that the Contract(s) have been set to require authorisation. This is achieved via the **Contract Interface** tab on the Contract Lookup. Set the 'Financial Interface' field to 'Requires Authorisation'.
4. If financial authorisation is to be imported via the agent ensure that the Import Transformation is setup. To do this go to the **Import Transformations Lookup** and create a transformation that has a 'Process Type' of 'Financial XML Agent'.

5. Setup any Export Transformations required. To do this go to the **Financial Interface Export Transformations** lookup and set the transformation with any stylesheets that need applying.
6. Create the directory structure on the Confirm Task Processor server for imports and exports.
7. Ensure the Task Processor is configured and running

Full Text Index Agent

Overview

Full Text Index is a Scheduled Task that is responsible for synchronizing the records available in the Confirm database with the index table.

The Database needs to be configured for Full Text Agent. Please refer to **Full Text Search Settings** for Database configuration steps. When full text search is enabled for the first time through Confirm web interface, the Full Text Index agent is added automatically in the scheduled tasks, to run at an interval of one minute

Operation

On each scheduled run, the agent updates the revision number of new or updated Enquiry and Address records in the database. If full text search is enabled, it synchronizes the Enquiry and Address records with the index table.

Import Address Agent

Overview

Import Address Agent is a Scheduled Task that allows importing address records from .csv or .xml files specified in the Input Directory of the Import Transformation record using a stylesheet.

The Agent can be used to set up a folder where users can drop files to be imported, or to configure an interface for external systems to load data into Confirm.

This is achieved by configuring Import Transformations to be run by the Agent. The agent polls the specified folder for any file waiting to be imported, transforms the file to the Agent's preferred format, validates the content and successfully imports all records which pass the validation.

The records that do not pass the validation are not imported and an _err file is created within the 'Read' sub folder containing such records.

Operation

When the Agent is triggered, it processes all defined Import Transformations with the 'Import Address' Process Type.

Each Import Transformation has a specified Input Directory, and a File Name Mask (*.* by default) used as a filter so the Agent does not attempt to import inappropriate files. The Agent will identify all files in the Input Directory matching the specified mask, and will attempt to import each one in turn. The files are processed in date created order, from the oldest to the most recent.

Before being imported, the file may be transformed to the format Import Address Agent expects. The transformation to be carried out is determined by the ‘File Pre-processing Format’ specified on the Import Transformation record, as follows:

- (None) – the file is already in Import Address format as defined in the specification.
- XML – the file is an XML file and a Style sheet will be used to transform it to Import Address format.
- CSV – the file is in a .csv delimited text format which will be pre-processed to XML and then a Style sheet applied.

Once transformed, data will be loaded into memory from the file and then validation will be applied depending on the type of data being loaded. This includes ensuring all mandatory fields have been supplied, referenced lookup values exist, etc.

If the validation is successful, data will be saved to the database and the import file moved to a ‘Read’ sub folder within the Input Directory. In the event of any error messages being generated (for example, invalid/wrong data passed to a field) a corresponding <filename>_err.xml file will be created in the same 'Read' sub folder location. This file will include all the records which were not imported successfully and will contain an <error> tag with actual error for that record.

If their is any error file generated in the 'Read' sub folder then user needs to correct the error for each node and also remove the <error> element for each node. The corrected file needs to be placed in the input folder again to be processed by the agent. A separate transformation is required to process these files. There should not be any Style sheet for the transformation and it's File pre-processing format should be set to (None).

Specification

The **Specification** section for this agent provides more details on file format and structure.

Configuration

This section details how to configure the Import Address Agent.

1. Ensure that the agent is scheduled as a Task in the **Scheduled Tasks** screen within the Confirm client.
2. Setup an Import Transformation via the **Import Transformation** screen. Set the 'Process Type' field to 'Import Address'.
3. Create the directory structure on the Confirm Task Processor server for imported files. The import folder should match the folder specified in the import transformation and will need a 'Read' directory created under it.
4. Ensure the Task Processor is configured and running

Import Anything Agent

This Agent provides an automated method of importing several different types of record into the Confirm database, in a variety of file formats. The records that can be imported include Features, Inventory, Jobs, Defects, and Traffic Count Surveys. The full list of supported record types and the associated Module requirements can be found in the specification.

The Agent can be used to set up a folder where users can drop files to be imported, or to configure an interface for external systems to load data into Confirm.

This is achieved by configuring a number of Import Transformations to be run by the Agent. Each of these polls a particular folder for any file waiting to be imported, transforms the file to the Agent’s preferred format, validates the content and, if there are no errors, saves the results to the database.

The same functionality is available interactively using the ‘Import Data’ option on the Confirm menu.

Operation

When the Agent is triggered, it processes all defined Import Transformations with the ‘Generic Import’ Process Type.

Each Import Transformation has a specified Input Directory, and a File Name Mask (*.* by default) used as a filter so the Agent does not attempt to import inappropriate files. The Agent will identify all files in the Input Directory matching the specified mask, and will attempt to import each one in turn. The files are processed in date created order, from the oldest to the most recent.

Before being imported, the file may be transformed to the format Import Anything expects. The transformation to be carried out is determined by the ‘File Pre-processing Format’ specified on the Import Transformation record, as follows:

- (None) – the file is already in Import Anything format as defined in the specification.
- XML – the file is an XML file and a Stylesheet will be used to transform it to Import Anything format.
- Others – the file is in a delimited text format which will be pre-processed to XML and then a Stylesheet applied.

Once transformed, data will be loaded into memory from the file and then validation will be applied depending on the type of data being loaded. This includes ensuring required Modules are present, all mandatory fields have been supplied, referenced lookup values exist, duplicates are not being created, etc.

If the validation is successful, data will be saved to the database and the import file moved to a ‘Read’ subfolder within the Input Directory. In the event of any warning messages being generated (for example, descriptive data being truncated to fit a field) a corresponding .err file will be created in the same location.

If the validation is not successful, the import file will be moved to an ‘Invalid’ subfolder within the Input Directory. A corresponding .err file will be created in the same location detailing the reasons for failure. Note that the Agent generally handles errors at file level, so any error detected within a file will result in the entire file being rejected.

Import Data Utility

The **Import Data Utility** within the Confirm Client can also be used to import files and provides the same functionality as the Agent

Specification

The **Specification** section for this agent provides more details on file format and structure.

Configuration

This section details how to configure the Import Anything Agent.

1. Ensure that the agent is scheduled as a Task in the **Scheduled Tasks** screen within the Confirm client.
2. Setup an Import Transformation via the **Import Transformation** screen. Set the 'Process Type' field to 'Generic Import'.

3. Create the directory structure on the Confirm Task Processor server for imported files. The import folder should match the folder specified in the import transformation and will need an 'Invalid' and 'Read' directory created under it.
4. Ensure the Task Processor is configured and running

Photo Upload Agent

Overview

This Agent moves entity task photos that have been uploaded from a mobile device from the Confirm database to a pre-configured data store location.

The photos can then be viewed or managed from the data store.

Operation

Photos are uploaded from mobile devices to the Confirm database.

On each scheduled run, the Agent checks for any new photos that have been added to the database.

The Agent will then check the **Document Links System Settings** screen and retrieve the document store details.

The Agent will then copy the photos to the document store and remove the photos from the database.

Configuration

This section details how to configure the Photo Upload Agent.

1. Ensure that the agent is scheduled as a Task in the **Scheduled Tasks** screen within the Confirm client.
2. Check that the 'Document Store' is configured in the **Document Links System Settings** screen.
3. Ensure the Task Processor is configured and running

Referential Integrity Agent

This Agent synchronizes changes made to entity records in the Confirm database with their corresponding map layer record. This ensures that entities can be accurately tracked on the Map.

The following entities are supported:

- Defects
- Enquiries
- Features
- Jobs
- Street Works (only EToN 4 and above records are updated on the map)

In addition the Agent also drives the **GPS Tracking** functionality.

Operation

When a Confirm entity of a type configured for Referential Integrity is added, amended or deleted within Confirm, it is flagged for processing by the Referential Integrity Agent. This is done by means of a record being inserted in the entity_modify table in the database.

When the Referential Integrity Agent is run, it examines the contents of the entity_modify table, and processes each as follows:

- If no map layer record exists, and the Confirm entity has coordinates, a map layer record is created using the coordinates of the Confirm entity. The attributes of the new map layer record are set to those of the Confirm entity.
- If a map layer record already exists, its coordinates and attributes are updated to match those of the Confirm entity.
- When a Feature is deleted or a Street Works is cancelled then the record on the map layer is deleted.
- When a Feature changes Feature Group, if the corresponding map layer is now different the Feature is deleted from the previous map layer and inserted onto the new map layer.

Once a record in the entity_modify table has been processed successfully it is deleted.

Once 1000 records have been successfully processed for a particular Confirm entity type then the Referential Integrity Agent will move onto the next entity type so it does not delay processing of other entities if there is a large backlog for one particular type.

Note: When processing Features, the geometry stored against the Feature is used in preference to coordinate data stored in Confirm, if the "Use Confirm Database Map Layer" option is selected on the **Feature Group**. If no geometry is present, and the corresponding feature on the map layer already exists, then the centroid coordinate will be used to update the location of the current map feature, if it is a point. Line and polygon map features would be left as they are in the case of no geometry being present on the Feature in Confirm.

Configuration

How to configure Referential Integrity

1. Ensure that both Map (0425) and Referential Integrity (0440) modules are enabled.
2. Create a layer with the correct fields according to the specification
3. A file-based layer should be in a location that is visible both to the Task Processor server and the Confirm Client.
4. Use the Map Workspace Manager to register the layer.
5. For Defects, Jobs, Enquiries and Street Works, open the **Map Settings screen** and specify the registered map layer.

   Note: With the exception of GPS Logging, layers are not created automatically. Blank entity layers can be supplied upon request.

6. For Features, the user can set different map layers for each of their Feature groups. Open the **Feature Group screen** and find the appropriate Feature Group. Specify the registered map layer in the Map Layer field on the Additional tab.
7. On the Scheduled Tasks screen, add the Referential Integrity task and set the correct schedule and a Supervisor email address.
8. Ensure the Task Processor is running.
9. Once the Task Processor is running, only Confirm Entities that are subsequently updated will be reflected on the corresponding Map layer. As such existing data will not be automatically shown on the map. In order to populate this data onto the Map Layers, SQL will be required and can be supplied upon request.

Street Works and Layer Colours

The Referential Integrity process will colour items on the layer according to the Promoters **Organisation via the Additional Tab**.

Whilst it is possible to change these colours, only Notices which have updates will be re-coloured by the Referential Integrity Agent.

Map Layer Specification

The Referential Integrity Agent will update the following columns depending on the Confirm entity being processed.

Referential Integrity will not automatically create the map layer unless stated in the section below.

Street Works Referential Integrity Map Layer

The layer must contain these primary columns:

- Work_Number (Char(12))

The layer may also contain the following optional columns. These attributes of the map layer record will be updated if present.

- Organisation (Char(100))
- Description (Char(254))
- Start_Date (Date/Time)
- End_Date (Date/Time)
- Organisation_Code (Char(8))
- Works_Reference (Char(20))

Defect Referential Integrity Map Layer

The layer must contain this primary column:

- Defect_Number, defect_no (Decimal(8))

The layer may also contain the following optional columns. These attributes of the map layer record will be updated if present.

- Defect_Type, def_type (Char(4))
- Description, descript (Char(254))
- Location (Char(254))
- Defect_Date, def_date (Date/Time)
- Defect_Status, def_status (Char(1))

Feature Referential Integrity Map Layer

The layer must contain one of these primary columns:

- Central_Asset_Id (Char(20))
- Asset_Id * (Char(15))

The layer may also contain the following optional columns. These attributes of the map layer record will be updated if present. Confirm columns can relate to different map column names depending on the flavour of Confirm being used and the layer's mapping technology. Below are the map column names that Referential Integrity can update:

- Site_Code (Char(10))
- Site_Name, Street_Name, Road_Name, Building_Name (Char(40))
- Plot_Number, Plot_No, Asset_Number, Sequence_Number, Feature_Number, Section_Number (Decimal(10,2))
- Feature_Id, Asset_Id *, Tag_Number, Road_Number, Label (Char(15))
- Feature_Type (Char(4)) (populated with the Feature Type Code)
- Feature_Type_Name, Type, Species, Carriageway_Type (Char(30))
- Feature_Group_Code, Feature_Gp (Char(4))
- Feature_Location, Location, Tree_Location, Feature_Location, Section_Location (Char(254))
- Primary_Measure, Prim_Meas (Decimal(10,2))
- Unit_Of_Measure, unit (Char(15))
- Layer (Char(50))
- Start_Date (Date/Time)
- End_Date (Date/Time)
- Centroid_Easting (Decimal(10,2))
- Centroid_Northing (Decimal(10,2))
- Extent_Easting_1 (Decimal(10,2))
- Extent_Northing_1 (Decimal(10,2))
- Extent_Easting_2 (Decimal(10,2))
- Extent_Northing_2 (Decimal(10,2))

* Asset_Id is only populated with the Feature Id column from Confirm if Central_Asset_Id is being used as the primary column. Otherwise Asset_Id will contain the Central_Asset_Id.

Enquiry Referential Integrity Map Layer

The layer must contain this primary column:

- Enquiry_Number, enq_no (Decimal(8,0))

For full functionality the layer must also contain the following columns:

- Service (Char(4))
- Subject (Char(4))
- Description, descript (Char(254))
- Location (Char(254))
- Logged_Date, log_date (Date/Time)
- Enquiry_Status, enq_status (Char(4))
- Site_Code (Char(10))

Job Referential Integrity Map Layer

The layer must contain this primary column:

- Job_Number (Decimal(8,0))

The layer may also contain the following optional columns. These attributes of the map layer record will be updated if present.

- Location (Character(250))
- Notes (Character(250))
- Job_Status (Char(4))
- Site_Code (Char(10))
- Plot_Number, plot_no (Decimal(10,2))

GPS Logging Referential Integrity Map Layer

Referential Integrity will automatically create this layer.

The layer will be populated with the Action Officer's current GPS location information when updates are received via the Confirm Connector.

The layer must contain this primary column:

- Officer_Code (Char(4))

The layer will also contain the following columns:

- Officer_Name (Char(30))
- Work_Group (Char(4))
- Officer_Telephone (Char(20))
- Location_Key (Decimal(9,0))
- Location_Date_Time (Char(30))

Repeating Jobs

Overview

Repeating Jobs is a Scheduled Task that is responsible for creating Jobs from Allocated Regimes and committing them to Works Orders as well as automatically assigning to Gangs.

Operation

The Repeating Jobs scheduled task will create Jobs against Features where Regimes have been allocated. This will apply when the Maintenance Regime is a "Repeating Jobs" type and there is no current (without an Actual Completion date and not cancelled) Job for the Allocated Regime.

The created Job will be placed on a Works Order and Committed according to the configured Priority for the chosen Regime. The Job will have the new Regime Code field set to the Maintenance Regime it was created from, to provide a history of why it was created. If the Allocated Regime has a Gang set against it then the Job will also be automatically allocated to the Gang and scheduled for the current Next Due Date.

When the Agent runs, it will process (create) up to a maximum of 1000 Jobs in any single run, should more Jobs be required, they will be created on subsequent runs.

When a Job created by the agent is completed a new Time-based repeating Job will be generated for the corresponding Allocated Regime. When calculating the due date for the new Job the due date rule for the Regime will determine whether the actual start/completion date or due date from the previous Job should be used or the original due date on creation of the previous job; The follow up period of the Regime (Repeating Period / Seasonal Follow Up) will be added to this date to determine the new Job due date. If the new Job due date is in the past, then the follow up period of the Regime will be repeatably applied until the new Job due date is a future date.

When an agent creates a Meter-based repeating Job, the meter next due is set while linking the regime to the Feature. A new Job is created when the task processor runs and the current condition value is greater than or equal to the meter next due.

If Allocate Regime Meter Next Due is selected as the due meter rule during regime configuration, the new meter next due is updated by adding the meter interval to the existing meter next due value.

If Lower of the Meter Next Due or Last Service Meter is selected during regime configuration, the new meter next due is calculated by adding the meter interval to the lower of the meter next due or the last meter reading.

If Higher of the Meter Next Due or Last Service Meter is selected during regime configuration, the new meter next due is calculated by adding the meter interval to the higher of the meter next due or the last meter reading.

For repeating types TIME or METER, a Job is not created if both the next due date and meter next due are missing. If either one is present, the Job is created according to the defined rules.

Note: If Seasonal Follow Ups have been configured then it will use the follow up period appropriate for the current time of year instead of the specified Repeating Period, if set. In the situation where the Seasonal Follow Ups are not commencing on 1st January, then the last Seasonal Follow Up for the year will be "wrapped around" and used for the time period from 1st January to the first Seasonal Follow Up that has been configured.

Similarly, when a Job created by the Agent is cancelled, set to Status that is not Outstanding and not Complete, a new Job will also be generated. For the Due Date calculation, instead of using the Actual Start/Completion date, the Date the Status was logged is used.

If the 'Repeating Routine Jobs Future Period' Job System Setting has been set, then only when the next Job due date falls within the period, will a Job be created by the Agent.

Configuration

This section details how to configure Repeating Jobs.

1. Ensure Repeating Jobs Agent is scheduled to run in the **Scheduled Tasks** screen within the Confirm client.
2. Ensure a **Maintenance Regime** has been configured in the **Maintenance Regime** section of Lookup Manager, in Confirm Web. A Job Type, Priority, Due Date Rule and Repeating Period / Seasonal Follow Ups will need to be set.

   Note: If the chosen Job Type contains a Default Hazard Type which has an Attribute (Operational) that has the Must change default flag set, the Job will not be created.

3. Ensure the **Maintenance Regime** is allocated to **Features** on the **Allocated Regimes** screen. An Officer will need to be set as well as a Contract if the **Priority** on the **Maintenance Regime** creates Orders. A Gang will also need to be set in order for Jobs to be automatically scheduled.

   Note: If the chosen Priority, set on the Regime and using the Device Creation of Orders, is also configured to Commit. Each raised Job will create its own Order. The Group options will not group Jobs onto Orders.

4. At least one SOR Item is applied to the Regime and has a Measurement Type specified.
5. The chosen Contract's BoQ must contain each SOR Item on the chosen Regime.

   Note: If the SOR Item is not on the Contract's BoQ, it can prevent the Job from being Committed (where the Regimes Priority is configured to commit).

6. Each Feature, the Regime is applied to, needs to contain an Attribute that matches the Measurement Type for an SOR Item on the Regime. That Attribute must have a value greater than 0, as this is used to set the Item Quantity values on the Item/s for the raised Job.
7. Set the 'Repeating Routine Jobs Future Period' **Job System Setting**, if desired, to prevent Repeating Jobs from being created too far into the future. Allocated Regimes that are not currently within this Future Period will get picked up by the Agent as soon as they do fall within this setting.

Scheduled Reporting Agent / Dashboard Agent

The Scheduled Reporting Agent and the Dashboard Agent are used to extract information from the Confirm database at regular intervals and publish it. They can be used to drive email alerts, to publish management reports, for interfaces exporting data to other systems, and to provide the data behind Confirm Dashboards.

The processing of Scheduled Reports is shared between the two Agents, in order to prevent time critical Dashboards being delayed by long running Reports or newly created Reports.

The behaviour of the two Agents is essentially identical, except for which Reports they process. The Dashboard Agent will process Scheduled Reports satisfying the following criteria:

- The Scheduled Report is set to 'output to database' i.e. it is used to power a Dashboard.
- The last successful run of the Scheduled Report took 60 seconds or less.

The Scheduled Reporting Agent will process all other Scheduled Reports. This includes any that are being run for the first time or where the underlying Data Source has been edited since the last run.

Note: If setting up a new system it is important to configure both Agents, otherwise some Scheduled Reports will not be processed.

Operation

The Agent will process all Scheduled Reports with a ‘Next Run Time’ less than the time at which it runs. It does this in order of Next Run Time, with the Scheduled Report with the earliest Next Run Time being processed first. If a Scheduled Report has ‘Process Backlog’ ticked then its entire backlog will be processed before moving on to the next Scheduled Report.

A Scheduled Report with a blank ‘Next Run Time’ will be ignored by the Agent so clearing the ‘Next Run Time’ in the screen in Confirm can be used as a way to temporarily or permanently disable a particular Scheduled Report.

A Scheduled Report will also be ignored if it has no Recipients and there are no Dashboards dependent on its results. In this case, the Agent will log a warning and then move on to process the next Scheduled Report.

The Agent has some Tolerance checks in place to detect Scheduled Reports that take too many resources, as detailed in the following table.

Tolerance	Default Value	Effect
Time Warning Tolerance	15 minutes	If the time taken for a Scheduled Report to retrieve its data exceeds this Tolerance, a warning is logged and emailed to the Agent Supervisor.
Time Error Tolerance	1 hour	If the time taken for a Scheduled Report to retrieve its data exceeds this Tolerance, the Scheduled Report is aborted and the Agent proceeds to the next one.
Memory Tolerance	1.8 GB	If the total memory allocated to the Task Processor while processing a Scheduled Report exceeds this value, then the current Scheduled Report is aborted. As the memory is not necessarily released immediately, the Agent will also end its run without processing subsequent Scheduled Reports.

Tolerance	Default Value	Effect
		The memory required in processing a Scheduled Report can be reduced by altering it to return a smaller result set (fewer rows or fewer columns). **Note:** There is a hard limit of 2.0 GB so the Tolerance should not be increased above this.

Note: The Tolerances may be adjusted from their default values using settings in the database. If the values above are found to be not appropriate, SQL can be obtained from Confirm to do this.

Configuration

The detailed configuration is done in the Scheduled Reports screen, where the behaviour of each individual Scheduled Report is specified.

Each Scheduled Report has a Data Source determining what results it will retrieve, and a Follow Up determining the frequency at which it will be run.

Each one is also configured as to how its results are published. Any combination of the following can be chosen:

- Results saved to one or more files.
- Results sent via email to one or more Contacts.
- Results saved to the database for use in Confirm Dashboards.

Street Works Transfer

Overview

Street Works Transfer is a Scheduled Task that is responsible for sending EToN 4 or above notices. This covers Notices, Comments, EToN 6 or above Inspections, Fixed Penalty Notices, Restrictions, Operational District Data and also progresses Events and Diversions through their lifecycle.

Note: When API Notification is subscribed from Confirm Web Settings, Street Works Transfer will be responsible for exporting notices. It will also handle importing designations and non-work related notices.

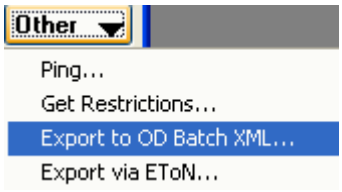
Operation

Testing OD Batch XML

The steps below configure data in order to test the sending of your OD Batch information:

- Launch Confirm and open the Organisation Lookup screen ‘Street Works > Street Works Lookups > Organisation’.
- Retrieve your Organisation details.
- Refer to the Online help file (Organisation / Operational District (OD) transactions section) to setup all the required fields.

- Click the ‘Other’ button and select ‘Export to OD Batch XML…’.



- Select an OD ‘Filename’ location and ensure the ‘Preview’ option is enabled, click ‘save’ then click ‘ok’.
- The Operational District Export preview screen will appear displaying whether the Organisation details pass validation. Assuming it passes validation, click ‘Close’ and you will be returned to the Organisation screen.

  Note: By default the Validation process will only check the Organisation you currently have selected; when the EToN export is performed it will attempt to send the information for all Organisations belonging to the Organisation Group that this Organisation belongs to. If the Organisation field is cleared then it will check all the Organisations belonging to the specified Group.

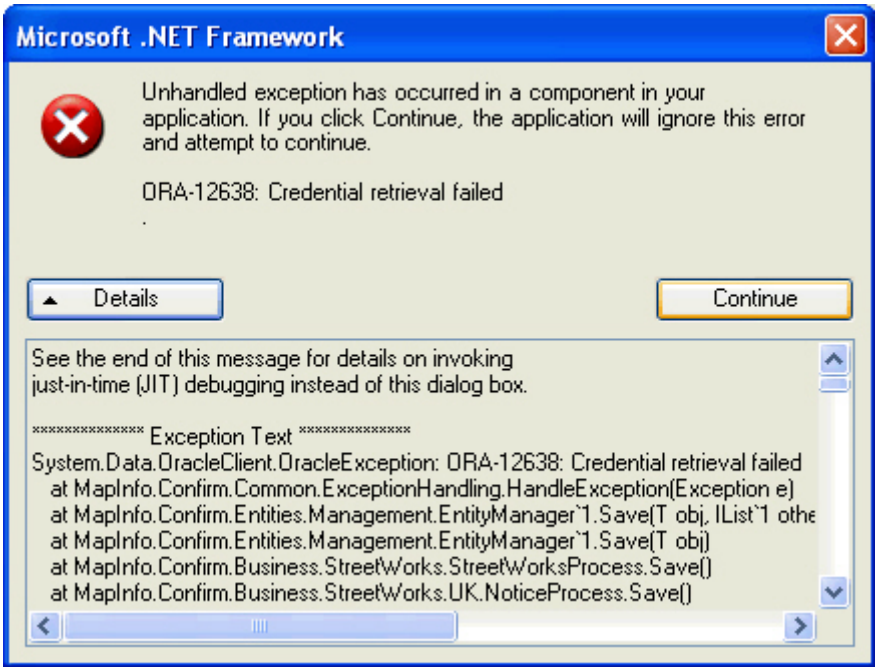
- Click the ‘Other’ button and select ‘Export via EToN…’.
- The batch Recipients screen will appear (assuming this is the first time you have run the Export, if it is not the first time then you will be asked if you wish to modify the existing batch.) In the To field enter the Organisation you wish to send the currently selected Organisation details to.
- Click ‘Save’, this will save the batch, making it available for export using Street Works Transfer.
- Finally, run the Street Works Transfer Agent to export the OD Batch information to the selected Organisation.

Should this fail for any reason, check the Street Work Transfer trace log file and information log file.

Troubleshooting

ORA-12638 ‘Credential Retrieval Failed’ Error

If the following error is received:



Locate the SQLNET.ora file. This will be found where the Oracle client has been installed, Edit the file with a text editor and change the line:

SQLNET.AUTHENTICATION_SERVICES = (NTS)

to

SQLNET.AUTHENTICATION_SERVICES = (NONE)

(NTS) is Windows Native Authentication and uses the client login to Access Windows NT Server and the database running on that server. This generally doesn't work for access to the Confirm database. Setting it to (None) means no special Authentication method is used and the database Username and Password are obtained from Confirm correctly in the normal way.

Logging

The Confirm Task Processor creates a log file as specified on the Task Processor System Settings. This will contain information of all transactions done by the Street Works Transfer Agent along side all other Agents. This file will have following messages:

- Notice Sent, Updated Database successfully

  This indicates that the transaction has been sent to a receiving web service and has been accepted.

- Sending <Temp File Path>\<Filename.xml> to: http://<URL>/eton4webservice.asmx

  This indicates that an EToN transaction is being sent to a web service.

- Receiver Fault received from Web Service: Error: Error processing request

  This indicates that an error has been received from the destination web service. Contact the Organisation which you are sending to. In the case of another Confirm system, review the Trace.log file for their web service.

- BatchFailed Record created. BatchID: 1234567

  This indicates that the EToN transaction failed to be sent to the web service. The next time the Street Works Transfer runs, it will attempt to re-send the same transaction.

- Due to Previous Notice failure. Previous Works: WZ00180000001

- Notice Validation Error: The element 'xxxxxxxxx' in namespace 'http://www.wrcplc.co.uk/Schemas/ETON' has invalid child element 'xxxxxxx' in namespace 'http://www.wrcplc.co.uk/Schemas/ETON'

  This indicates that the Operational District (OD) data which the Street Works Transfer is trying to send does not pass the EToN Schema validation checks. Refer to the Street Works user manual – Organisation section for details on how the Organisation data should be completed.

- Error: Client found response content type of 'text/html', but expected 'application/soap+xml'.

  The following error can occur when the Web Service receiving the notice has incorrectly configured on an IIS 7 server.

  26/04/2010 10:42:52 : Error : Error: Client found response content type of 'text/html', but expected 'application/soap+xml'.

  500 - Internal server error.

  Note: There is more content in the error.

- Error: Client found response content type of 'text/html; charset=utf-8', but expected 'application/soap+xml'.

  The following error can occur when the Web Service receiving the notice has incorrectly configured on an IIS 7 server.

  26/04/2010 13:48:48 : Error : Error: Client found response content type of 'text/html; charset=utf-8', but expected 'application/soap+xml'.

  <title>Runtime Error</title>

  Note: There is more content in the error.

Error Log

An Error Log file is created in parallel with the Log File specified above. The file name of the Error Log file will be the same as the Log File with 'error' prefixed to it. The Error Log file contains only error trace messages, and all other messages are omitted.

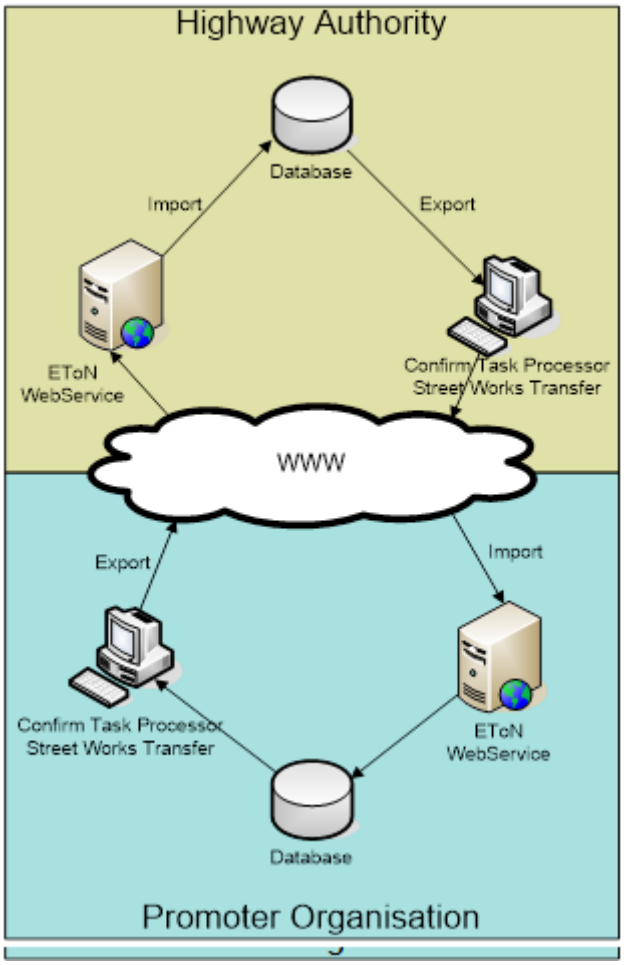
Configuration

Related Documentation

The following is a list of documentation related to Street Works Transfer

- CHA16002 - Confirm Street Works TMA Deployment Implications.

Note: Street Works Transfer only exports data. The Street Works EToN Web Service (CHA17408) does the Importing.



The above example displays two Confirm Environments for EToN 4 and above Transfers

Street Works Transfer Agent Directory Structures

The below is an example directory structure for Street Works Transfer Agent and may vary depending on your configuration options. Street Works Transfer uses XML (via HTTP) in order to send notices and the Web Service receives EToN notices.

Transfer Files	Description
Confirm\Transfer\EToNXML	Parent Transfer directory for EToN XML Notices.
Export	Parent directory for the Export of EToN XML notices.
Sent	Sent
Failed	Location of the unsuccessfully sent xml files.

Pre-requisities

Street Works Transfer requires the following to be installed and configured in order to successfully transfer notices via EToN:

- Ability to send XML data via HTTP
- (Optional) - SMTP mail access for sending of Alert emails.

Transferring Notices

EToN version 4 and above transfer data in XML format. The Street Works Transfer Agent will export the XML transactions for Notices, Inspections (EToN 6 and above), Comments, Restrictions and Operational District (OD) data.

Make the following changes to process all XML notices:

- Update TaskProcessorService.exe.config file, located in the Confirm Task Processor installed directory (e.g. C:\Program Files\Confirm\Task Processor)

  <add key="WebServiceCallTimeout" value="120000"/>

  This is the length of time that the Street Works Transfer will wait for the Web Service to respond in milliseconds.
- For General Settings of Street Works refer to **Street Works Transfer System Settings**

Email Options

The email functionality for Street Works Transfer will alert the selected email address when the following scenarios occur:

- Connection to database is lost or unavailable.
- When 3 failed attempts of the same notice have occurred.

To enable emails make the following changes:

Setting	Description	Location
SmtpServer	Set to either the mail server name or IP address. To connect the Confirm Task Processor to an SMTP server which is running on a non standard port number (25) then append ':port-number' to the server name / IP address	In the registry: • **32-bit Machine:** 'HKEY_LOCAL_MACHINE\SOFTWARE\Confirm\Task Processor' • **64-bit Machine:** 'HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Confirm\Task Processor'
From Name	Set to the name you wish the email to come from.	Refer to **Task Processor Settings.**
From Address	Set to the email address to be used to send emails.	Refer to **Task Processor Settings.**
Supervisor Email	Set to the email address to be used to receive emails.	Refer to **Supervisor Email within Scheduled Task Screen.**

FPN Email Settings

The following settings define the structure and contents of FPN emails, and any error emails that are generated:

- Update TaskProcessorService.exe.config file, located in the Confirm Task Processor installed directory (e.g. C:\Program Files\Confirm\Task Processor)

  Note: For all of the below steps, ensure that you change the entry within quotes in the value="".

  - *<add key="SMTPHostIPAddress" value="127.0.0.1" />*

Set to the IP Address of your SMTP server.

- *<add key="SMTPHostPort" value="25" />*

  Set to the port number your SMTP server is running on.

For General Settings of FPN Email refer to **Penalty System Settings.**

Proxy Server Settings

It is possible to configure the Street Works Transfer to use a Proxy Server when transferring Notices via the HTTP protocol. In practice this means Notices for EToN 4 or above.

The Street Works Transfer will not use a Proxy server by default. In order to enable the use of a proxy server you must edit the TaskProcessorService.exe.config file found in the Confirm Task Processor Installation directory and set the following key to 'Y':

- <add key="ProxyEnabled" value="Y"/>

You must then put proxy server connection settings in one of two places:

- In the Confirm database. These settings The General Systems Settings screen of the Confirm System Administration module.
- The TaskProcessorService.exe.config text file.

  Settings found in the TaskProcessorService.exe.config file will be used in preference to those in the General System Settings screen, but it is not necessary to have settings in the TaskProcessorService.exe.config file if they are set inside Confirm.

  The TaskProcessorService.exe.config proxy connection settings are as follows:

  - <add key="ProxyAddressAndPort" value="127.0.0.1:8050"/> Enter the IP address and port number of your Proxy Server. Make sure you use a colon to separate the IP address and port number.
  - <add key="ProxyUserName" value=""/> Enter the username if required for authenticating against your Proxy Server.
  - <add key="ProxyPassword" value=""/> Enter the password for the username being used in step 3.

Folder Permissions

When Street Works Transfer attempts to send a Notice it will create an XML file in the 'Sent' or 'Failed' directories (see section 3.2 step 5 and 6). In order for this to work you will need to grant the following permissions:

- Grant 'Full' access to the account being used to run the Confirm Task Processor to the Confirm Task Processor directory and all sub directories. This will allow the creation of the log file and the generation of the XML Notice files.

Note: If you change the default location of the FailedFiles and SentFiles, you will need to make sure you grant full permissions to the directories you have used.

Oracle Client

When using Oracle as the DBMS of choice, the Task Processor will need to be able to access the Oracle client drivers in order to connect to the Confirm database, by default, permission to these drivers is denied. Follow the steps below to grant permissions to the Oracle client drivers for the Task Processor:

- Launch Windows Explorer and navigate to the ORACLE_HOME directory; this is the path where the Oracle client has been installed.
- Right click on the ORACLE_HOME folder and choose the “Properties” option from the drop down list.
- Click on the “Security” tab of the “Properties” window.
- Click “Authenticated Users” item in the “Name” list (on Windows XP the “Name” list is called “Group” or “User Names”).
- Uncheck the “Read and Execute” box in the “Permissions” list called “Permissions for Authenticated Users”.
- Re-check the “Read and Execute” box under the “Allow” column (this is the box you have just unchecked).
- Click the advanced button and in the “Permissions Entries” list make sure you see the “Authenticated Users” listed there with:
  - Permission = Read & Execute.
  - Apply To = This folder, subfolders and files.
- If this is not the case, edit that line and make sure the “Apply onto” drop down is set to “This folder, subfolders and files”. This should already be set properly but it is important you verify this.
- Click the “Ok” button until you close out all of the security properties windows.

It has been known for the above process not to fix the problem, if you encounter this situation then follow the steps below:

- Use Windows explorer and navigate to the Oracle home directory right click then select properties, select the security tab.
- Tick “allow inheritable permissions from parent to propagate to this object”.
- Click copy, when the security window appears click apply.
- Click the advanced button and tick the “allow inheritable permissions from parent to propagate to this object”, click apply then click ok and finally click ok.

Street Manager Notification Agent

Overview

Street Manager Notification Agent is a Scheduled Task that is responsible for Importing Work related Events coming from Street Manager. This covers Permits, Comments, Inspections, Fixed Penalty Notices and other work related events.

Note: Street Manager Notification Agent will only be responsible for Importing Permits and it's related events, when API Notification is subscribed from Confirm Web Settings.

Operation

On each scheduled run, the Agent will process the import of Notices coming from Street Manager. The Agent will fetch the Data from the Database Table.

Import

The Agent will import data related to Notices. The following data can be imported:

- Comments.
- Notices.

- Inspections.
- Fixed Penalty Notices.
- Other Work Related Notices.

Logging

The Street Manager Notification Agent creates a log file as specified on the Task Processor System Settings. This will contain information of all transactions done by the Street Manager Notification Agent along side all other Agents

Error Log

An Error Log file is created in parallel with the Log File specified above. The file name of the Error Log file will be the same as the Log File with 'error' prefixed to it. The Error Log file contains only error trace messages, and all other messages are omitted.

Configuration

This section details how to configure the Street Manager Notification Agent.

1. Ensure that the agent is scheduled as a Task in the **Scheduled Tasks** screen within the Confirm client.
2. Ensure the Subscription is done using the option provided on Web Setting Screen.

Connector Agent

Overview

This Agent polls configured directories for files which are then sent to the Confirm Connector. This provides a way of accessing the functionality without the need for sending requests to a Web Service.

Configure the Import Transformations screen within the Confirm Client application with a 'Process Type' of 'Connector Agent'. When files are found to be processed they will be transformed (if required) and sent to the Connector.

Note:
- The resulting XML cannot contain an 'Authentication' element as the Task Processor user details will be used.
- Each file can only contain one XML transaction.
- This Agent requires the Confirm Connector to be installed on the same machine as the Task Processor in order to process.