Confirm

Confirm Task Processor - Financial Interface XML
Version v18.20a.AM
• 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 https://github.com/mbostock/d3/blob/master/LICENSE. The source code for this software is available from http://d3js.org/.

• OpenLayers, version 2.12, which is licensed under the Modified BSD License. The license can be downloaded from http://svn.openlayers.org/trunk/openlayers/license.txt. The source code for this software is available from http://trac.osgeo.org/openlayers/browser.

• OpenLayers, version 3, which is licensed under the BSD 2-Clause Licence. The license can be downloaded from https://github.com/openlayers/ol3/blob/master/LICENSE.md. The source code for this software is available from https://github.com/openlayers/ol3.

• 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 4.2.0, 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/.

July 26, 2018
# Table of Contents

**Confirm Task Processor**

- Financial Interface XML Agent  6
- Operation  6
- Configuration  7
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

Financial Interface XML Agent 6
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