• 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/.

September 06, 2017
Table of Contents

Confirm Task Processor

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alerts and Notifications Agent</td>
<td>6</td>
</tr>
<tr>
<td>Operation</td>
<td>6</td>
</tr>
<tr>
<td>Configuration</td>
<td>7</td>
</tr>
</tbody>
</table>
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

Alerts and Notifications Agent 6
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.

<table>
<thead>
<tr>
<th>Alert Entity</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job</td>
<td>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.</td>
</tr>
<tr>
<td>Enquiry and Commitment</td>
<td></td>
</tr>
<tr>
<td>Offence</td>
<td>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.</td>
</tr>
<tr>
<td>Street Works Inspection</td>
<td></td>
</tr>
<tr>
<td>Street Works Occupancy</td>
<td></td>
</tr>
<tr>
<td>Street Works Reinstatement</td>
<td></td>
</tr>
<tr>
<td>Street Works Comment</td>
<td></td>
</tr>
<tr>
<td>Street Works Emergency Notice</td>
<td></td>
</tr>
<tr>
<td>Street Works Notice Sequencing</td>
<td></td>
</tr>
<tr>
<td>Street Works Notice</td>
<td></td>
</tr>
</tbody>
</table>

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 Pitney Bowes 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.