Confirm®

Confirm Task Processor - Referential Integrity
Version v18.20a.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 Pitney Bowes Inc., One Global View, Troy, New York 12180-8399.

© 2018 Pitney Bowes. All rights reserved. MapInfo, the MapInfo logo and Confirm are trademarks of Pitney Bowes and/or its affiliates.

Corporate Headquarters:
Phone: 518 285 6000
Fax: 518 285 6070
Sales: 800 327 8627
Government Sales: 800 619 2333
Technical Support: 518 285 7283
http://www.pitneybowes.com/us

UK and EMEA Headquarters:
Phone: 1 800 840 0001
Technical Support (UK): 1 800 840 0001
Technical Support (International): 44 1634 880141
http://www.pitneybowes.com/uk

Asia Pacific Headquarters:
Phone: 61 2 9437 6255
Fax: 61 2 9439 1773
Technical Support: 1 800 648 899
www.pitneybowes.com.au/software

Contact information for all Pitney Bowes offices is located at: http://www.pitneybowes.com/uk/contact-us.

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 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 https://github.com/mbostock/d3/blob/master/LICENSE. 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

Referential Integrity Agent  6
Operation  6
Configuration  6
Map Layer Specification  7
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

Referential Integrity Agent  
6
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))