Location Intelligence
Infrastructure Asset Management

Confirm®

CRM Connector Schema

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

Specifications

CRM Connector Schema 6

Methods 7

Specifications

The following sections outline all the Specifications that exist within the Confirm functionality.

In this section

CRM Connector Schema 6

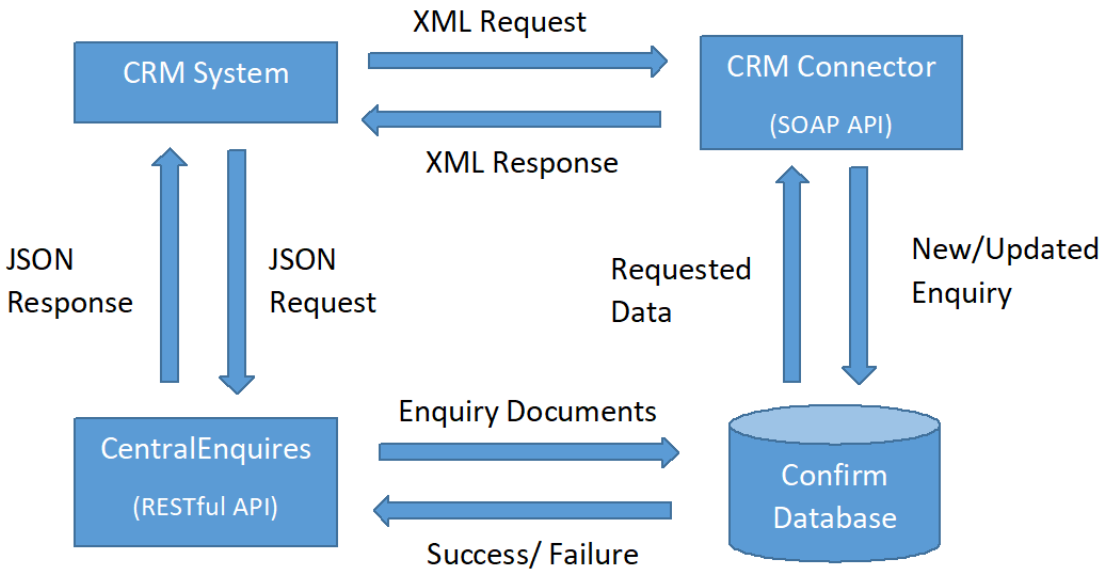
CRM Connector Schema

Introduction

This document describes the XML schema and API used to allow Confirm to be interfaced with a Customer Relationship Management (CRM) system using the Confirm Connector and the CentralEnquiries API.

The data transfer will be initiated by the CRM system. The Connector will always send data as a reply to the request.

Context Diagram



CRM Connector Operation Overview

The Request sent to the Connector contains an “Authentication” section detailing the database connection and Confirm User information, and an “Operation” section detailing the operations to be carried out on that database.

This document describes briefly each of the operations that might be included in a Request, and the corresponding Response for each.

Where data is provided in a lookup field it must correspond to a valid record in the Confirm database, otherwise a Fault will be returned.

Where subsequent sections refer to a “User” this is the Connector User, i.e. the Login Name the Connector uses in connecting to the Confirm database. This determines which data will be returned where Data Key Security is in place.

XML Schema Definitions

The Request XML is defined in the following schema documents:

- CRMRequest.xsd
- DataDictionary.xsd

The Response XML is defined in the following schema documents:

- CRMResponse.xsd
- CRMConnectorResponseData.xsd
- DataDictionary.xsd

Methods

This section describes the Requests to be sent from the CRM system to Confirm, and what the Responses will contain.

GetEnquiryLookups

This operation returns Active and Updatable Types of Service and live Subjects to which the User has Update Access, as well as other lookup records listed here:

- Enquiry Status records (only those not flagged as Dead)
- Enquiry Attribute Types and values
- Enquiry Subject Attributes
- Enquiry Classification records

GetCustomerLookups

This operation returns the following Customer related lookup records:

- Points of Contact
- Enquiry Methods
- Customer Types

GetActionOfficers

This operation returns all active Action Officers.

GetAllActionOfficers

This operation returns all active and inactive Action Officers.

GetEnquiry

The purpose of this operation is to retrieve an Enquiry using the EnquiryNumber field supplied. This will only return an Enquiry where the User has at least View Access to the Enquiry Subject and to the Action Officer’s Enquiries Data Key.

This Element returns a specific Enquiry identified by the EnquiryNumber. Returns no data if the Enquiry specified does not exist in Confirm or the User does not have access to it.

GetEnquiryByExternalRef

The purpose of this operation is to retrieve the Enquiry based on the external system identifier. This will only return an Enquiry where the User has at least View Access to the Enquiry Subject and to the Action Officer's Enquiries Data Key.

GetEnquiryBySite

This operation would generally be used to search for potential duplicate Enquiries prior to logging a new Enquiry in Confirm. This will only return Enquiries where the User has at least View Access to the Enquiry Subject and to the Action Officer's Enquiries Data Key.

As well as a Site, the Request can include a Type of Service, an Enquiry Subject, and a logged date range. The Enquiries returned will be restricted accordingly.

Note: If neither from nor to date is present in the Request, only Outstanding Enquiries will be returned.

NewEnquiry

This operation is used to create a new Enquiry in Confirm. Only live Enquiry Subjects to which the User has Update Access can be specified.

A new Enquiry Number will be generated by Confirm when the Enquiry is imported – the EnquiryNumber element in the Request is mandatory, but is used only for cross-referencing with other Mobile elements (not shown here). If such cross-referencing is not taking place then the supplied Enquiry Number is ignored.

If SiteCode is not supplied for a new Enquiry, it will attempt to assign one automatically by comparing the supplied coordinates for the Enquiry (EnquiryX and EnquiryY) with the Centroid and Extent coordinates stored on Site records in the Confirm database, if Customer Services Agent System Setting having "Derive Site from Enquiry Coordinates if not supplied" setting checked. Where no suitable Site can be assigned or if no coordinates are supplied for the Enquiry, it will be assigned to a default Site specified as a System Setting.

Enquiry Attributes, Customers and Document Links can be supplied in the Request for the new Enquiry.

The logged time supplied in the Request sets the LogEffectiveTime in Confirm. The actual LoggedTime in Confirm is set to the time when the data was processed. If not supplied then the current system date is used.

If NoticeFromOrgCode and WorksReference are supplied, the values will be appended to the Enquiry Description, in square brackets, as follows: [NoticeFromOrgCode / WorksReference].

A Job Number may be supplied, allowing the Enquiry to be linked to an existing Job.

EnquiryUpdate

This operation updates the status of an Enquiry already logged in Confirm. The User can only specify an Enquiry when the User has Update Access to the Enquiry Subject and to the Officer's Enquiries Data Key.

The Enquiry can be identified by either its Enquiry Number or its External System Reference. If both an Enquiry Number and External System Reference are supplied then the Enquiry will be found using the Enquiry Number and the External System Reference will be updated to the one supplied.

Updates to Enquiry Attributes can be supplied in the Request. Only the specific Attributes that have changed need to be supplied.

The Type of Service and Enquiry Subject of the Enquiry can be changed. Both codes must be supplied together in order to make a change.

New Customers can be added via this request. Details of existing customers cannot be updated.

New Document Links can be supplied and the notes of existing Document Links (where a Link already exists with the supplied Document Location) can be updated.

The logged time supplied in the Request sets the LogEffectiveTime in Confirm. The actual LoggedTime in Confirm is set to the time when the data was processed. If not supplied then the current system date is used.

The Response includes details of the Enquiry that was updated. No data is returned if the Enquiry specified cannot be found in Confirm.

GetEnquiryStatusChanges

This operation will retrieve details of all Enquiries (whether sent by the External System or logged directly in Confirm) where an update has created a new Status Log between the dates specified. It would normally be invoked as part of a background synchronisation process.

The Response includes all Enquires whose status has been updated between the dates specified in the Request.

The External System Number identifies the source system (Confirm code for External Issue System – database field ext_system_number).

The returned Logged Time is the date the status change was recorded in Confirm.

The Log Effective Time is the date specified by the user as to when the actual status change took place (may be different from the above).

GetNewEnquiryStatusChanges

This operation will retrieve details of all Enquiries entered or updated in Confirm since the last time this operation was invoked. This includes the initial Status Log entry for Enquiries that have been newly added in Confirm as well as updates to existing Enquiries where an update has created a new Status Log.

In addition, the Response will also include details of any new Customers that have been attached to the existing Enquiries.

Note: The operation is intended to inform a CRM system of changes made directly in Confirm. Hence Enquiries entered or updated via the CRM Connector will not be returned by this Request. Similarly, Enquiry Updates that have already been exported via the CRM Agent will not be returned by this Request (and vice versa).

GetActivityBySite

This operation will return all the Activities on a specified Site, which includes a list of Jobs, Street Works, and Enquiries for which the User has at least View access to the Enquiry Subject and to the Officer’s Enquiries Data Key.

The supplied ActivityFromDate and ActivityToDate define a date range of active Jobs, Street Works and/or Enquiries to be returned.

If the GetEnquiries element is present in the Request, Enquiries will be returned.

- Enquiries will only be returned if the Asset Type, determined by the “Type of Service” in EnquirySubject, matches the Asset Type of the User Id specified in the .INI for the current connection.
- Enquiries will be retrieved if a Status Log Entry has an “Effective Date” within the specified dates. Even though there may be more than one Status Log “Effective Date” within the specified dates, the Enquiry will only be returned once.

If the GetJobs element is present in the Request, Jobs will be returned.

- Jobs will be retrieved based on “Start Date” (if this is null then “Entered Date” will be used) and “Actual Completion” date (if null then “Est. Completion” date will be used). If the “Est. Completion” date is also null, then the Job will be retrieved if the “Start Date” (or “Entered Date”) lies between the supplied ActivityFromDate and ActivityToDate range.
- The returned StartDate will be Job Start Date, unless this is null and then it will be the Job Entry Date (which is mandatory).
- The returned EndDate will be Actual Completion date, unless this is null and then it will be the Estimated Completion Date.

If the GetWorks element is present in the Request, Street Works will be returned.

- There will be two database queries used to retrieve the Works Activity: Street Works with a Work Status of “Proposed” will be retrieved where the “Start Date” and “Estimated End Date” from the current Notice overlap with the ActivityFromDate and ActivityToDate of the Request. All other Works will be retrieved based on the “Actual Start Date” and “Actual End Date” of the Street Works Occupation record(s).

- This implies that the same Works could be retrieved more than once since multiple Occupation records may exist for the same Works.
- The ActivityDescription and LocationText in the Response will be retrieved from the most recent Notice.
- If the Works Status is “Proposed” or “In Progress” then the StartDate value comes from the Start Date of the most recent Notice. For all other Works Status values, the StartDate value comes from the “Actual Start Date” of the Occupation record(s).
- Similarly, if the Wks Status is “Proposed” or “In Progress”, then the EndDate value comes from the Estimated End date of the most recent Notice. For all other Works Status values, the EndDate value comes from the “Actual End Date” of the Occupation record(s).
- The CurrentStatusvalue is the Occupation status NOT the Wks Status, so P = Proposed, S = In Progress, R = Clear, D = Closed and A = Abandoned.

AssetSearch

This operation returns a list of Assets based on the two search criteria.

- If Site code is provided then this returns a list of Assets that occur on the specified Site.
- If Bounding Box extents are supplied then this returns a list of Assets lying partially or fully within the Bounding Box. If more than hundred such Assets are identified then hundred closest Assets to Bounding Box center are returned.

The search can also be restricted to specific FeatureGroupCodes in both the cases.

OpenRecord

This operation is used to launch a screen in Confirm and find a requested record or records.

Please note there is a limitation to the use of this request in that it requires a locally installed instance of the Connector in order to launch the user's Confirm application. The same effect can be achieved by launching Confirm directly with command line parameters. This request is therefore deprecated and it will be removed in a subsequent version.

If Confirm is already running the screen will be activated in the existing session.

If Confirm is not running the Connector will start Confirm and the user will need to provide Confirm login information before the requested screen is displayed. Alternatively, the Connector can be configured to automatically log into Confirm using the same user credentials that were supplied to the Connector.

OpenRecord details in the request parameters would include “RecordType” which should correspond to a table name in the Confirm schema. The table name specified is the internal table name and is not case sensitive.

Confirm will attempt to open an appropriate window to show records in the requested table. If there is no suitable window on the Confirm explorer menu, or if Module or Program Security prevents the User from opening the window, an appropriate message will be displayed.

The OpenRecord request would also include at least one “RecordId” tag representing a key value for the table. For example, for Jobs, Defects or Enquiries the key value “RecordId” would be Job, Defect or Enquiry number respectively.

For Features, the key value “RecordId” can be supplied as Site|Plot or as CentralAssetId as desired.

Note: In the case of Enquires, the window to be opened is determined at run time. If the Enquiry has a Customer (or in the case of multiples, if all the Enquiries have a Customer), the system will attempt to open Fast Entry Enquiry. If the Enquiry has no Customer, or the user has no access to Fast Entry Enquiry, the system will attempt to open Fast Entry Issue.