Confirm®
Central Enquiries API
v19.10c.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.

© 2019 Pitney Bowes. All rights reserved. MapInfo, the MapInfo logo and Confirm are trademarks of Pitney Bowes and/or its affiliates.

Corporate contact details for all offices can be found here:

Technical Support contact details can be found here:
https://www.pitneybowes.com/us/contact-dcs.html

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 https://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://github.com/mbostock/d3/.

- OpenLayers, version 2.12, which is licensed under the Modified BSD License. The license can be downloaded from https://github.com/openlayers/ol3/blob/master/LICENSE. The source code for this software is available from https://github.com/openlayers/ol3.

- 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 8.1.2, 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/.

April 09, 2019
# Table of Contents

## Specifications

- Confirm CentralEnquiries API 6
- Generating an OAuth token 6
- CentralEnquiries POST Request 6
- Confirm API Models 7
- API Response 8
Specifications

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

In this section

Confirm CentralEnquiries API 6
Confirm CentralEnquiries API

Confirm CentralEnquiries API allows uploading documents for an enquiry in Confirm. It can accept multiple documents in a Base64 encoded format. You have to first generate an OAuth token and then make your first API call. Refer the sections below to make your first API call.

Generating an OAuth token

Follow the steps to generate an OAuth token:

1. Obtain your API Key (Username) and Secret (password) from Confirm system administrator
2. To generate the OAuth Token, encode your credentials (API Key and Secret) using base64 computation mechanism. To do this, provide API KEY and Secret to the base64 encoder (online encoder can be used), and generate the encoded 'base64value'.
3. The following format should be used while computing the {BASE64VALUE}:
   
   {API KEY}:{SECRET}

4. Enter the generated 'base64value' in the header of the request and call the token URI as shown below:

   Authorization: Basic {base64Value}
   Content-Type: application/x-www-form-urlencoded
   POST {Confirm web url}/api/{tenant}/oauth/token
   grant_type=client_credentials

   Here, {tenant} is the tenant name and {Confirm web url} is the URL where Confirm web is deployed, like https://ConfirmWebServer/ConfirmWeb/.

5. The access token is returned as follows:

   ```
   {
       "access_token": "{your access token as a Base64 encoded value}"
       , "token_type": "bearer"
       , "expires_in": "{The expiry time in seconds}"
   }
   ```

Response:

The response of the token API follows a standard response pattern as described under API Response on page 8

CentralEnquiries POST Request

The POST request allows you to upload multiple documents for an enquiry in Confirm. It accepts and enquiry number and an array of documents.

To make an API call, you need to pass the OAuth access token in the request header (Making an API call).
Post Request Parameters:

<table>
<thead>
<tr>
<th>Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>enquiryNumber</td>
<td>integer</td>
<td>Required. The enquiry number.</td>
</tr>
</tbody>
</table>

Sample Request:

Authorization: Bearer {YOUR OAUTH ACCESS TOKEN}
Content-Type: application/json
POST {Confirm web url}/api/{tenant}/centralEnquiries

```json
{
   "enquiryNumber": "{Enquiry number}",
   "centralDocLinks": []
   {
       "documentName": "test.png",
       "blobData": "{Base64 encoded content}",
       "documentNotes": "Notes for test.png",
       "documentDate": "2018-09-03T16:20:22"
   },
   {
       "documentName": "test2.jpeg",
       "blobData": "{Base64 encoded content}",
       "documentNotes": "some notes"
   }
}
```

Response:

The response of the API follows a standard response pattern as described under API Response on page 8.

Confirm API Models

error

<table>
<thead>
<tr>
<th>Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>error_description</td>
<td>String</td>
<td>The error message explaining the reason why a document could not be uploaded.</td>
</tr>
</tbody>
</table>

centralDocLink

<table>
<thead>
<tr>
<th>Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>documentName</td>
<td>String</td>
<td>Required. The name of the document.</td>
</tr>
<tr>
<td>blobData</td>
<td>String</td>
<td>Required. The Base64 encoded string representing the document content.</td>
</tr>
</tbody>
</table>
Specifications

<table>
<thead>
<tr>
<th>Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>documentDate</td>
<td>String</td>
<td>Optional. The date when the document is uploaded. When omitted, current date and time is used.</td>
</tr>
</tbody>
</table>

**API Response**

An API returns a status 200 (OK) when successful. In some cases, specific error(s) can be returned in the response even though the overall API status code is 200 (for example, for each document that fails to upload for an enquiry, an error object is returned with the reason of failure, but the API as a whole returns a status code 200).

**Errors:**

The following table describes the possible errors that can be returned by an API

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>400</td>
<td>Invalid request parameters.</td>
</tr>
<tr>
<td>401</td>
<td>Authentication failures, invalid licenses or expired access tokens.</td>
</tr>
<tr>
<td>403</td>
<td>Insufficient permissions.</td>
</tr>
<tr>
<td>404</td>
<td>No entity found within Confirm with the supplied key.</td>
</tr>
<tr>
<td>500</td>
<td>Any other error.</td>
</tr>
</tbody>
</table>