

CPR Broker test guide

Updated : 29 February 2012

Contents

Contents..... 2

1 Introduction..... 3

2 Unit tests..... 3

3 Client test application..... 3

4 Demo application..... 4

1 Introduction

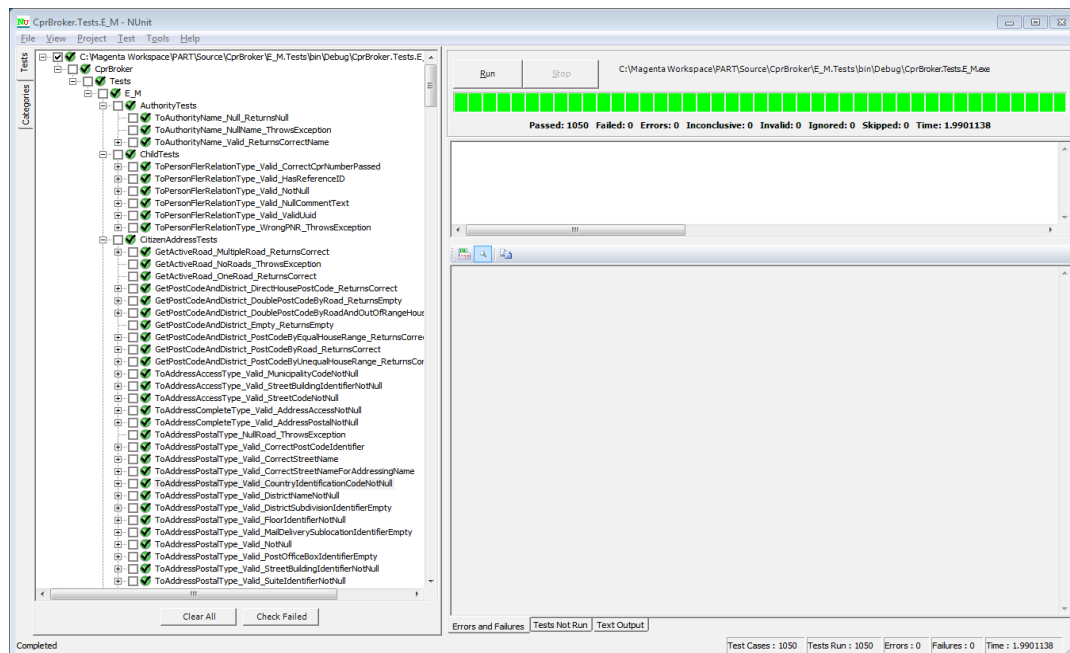
This document describes the possible ways to test CPR Broker implementation for E&M database.

2 Unit tests

The E&M implementation of CPR broker has been thoroughly tested using unit tests. The tests were based on the NUnit framework, and they cover almost every aspect of the E&M code.

The unit tests have been put into the E_M.Tests project (CprBroker.Tests.E_M.exe). It is available in the project source code under Source\CprBroker\E_M.Tests.

Please refer to the online user manual for NUnit at ([http://www.nunit.org](#)) for more details.



3 Client test application

Another method to test the application is to use the CprBroker.Tests.E_M.exe as a console application. The application will try to convert all the database records in the E&M database to PART standard, storing the xml to <working dir>\Runs\<current date and minute>\Registrations\.

The program takes the connection string as a parameter like this

E_M.exe "Data Source=<server>;Initial Catalog=<e&m databasename>;Integrated Security=SSPI"

By default, the app will try to use the Part.asmx service available at :

<http://CprBroker/Services/Part.asmx>

And will use the application token fb6e6e79-7505-4667-9736-d500092d844b (belonging to an application registered as a test app)

4 Demo application

Another way to test the implementation is to use the demo application. The demo application was created with the original version of the PART standard implementation in CPR Broker.

The demo application is located under Source\Demo.

Some screenshots and instructions are shown below

4.1 Initialization

This step is done only in the first time the demo application is run.

The screenshot shows a Windows-style dialog box titled "OptionsForm". It has three tabs: "Connection", "Notifications", and "User Token". The "Connection" tab is selected. Inside the dialog, there is a note: "Note: This is the base URL in which all CPR web services reside. Use 'Test connection' to enable Application Registration". Below the note, there are two text input fields. The first is labeled "CPR broker services base URL:" and contains the text "http://CprBroker/Services/". The second is labeled "Event broker services base URL:" and contains the text "http://EventBroker/Services/". To the right of these fields is a button labeled "Test connection". Below these fields is a section titled "Application Registration". Inside this section, there is a text input field labeled "Admin-application-token" containing the value "07059250-E448-4040-B695-9C03F9E59E38". Below this field is a button labeled "Register and approve Application with Web Service". At the bottom of the "Application Registration" section, there is a label "Registration status:" followed by the text "Application not yet registered.". At the bottom of the dialog box, there are three buttons: "OK", "Cancel", and "Apply".

Verify the service URLs and press 'Test Connection'.

Note

Please only specify the web folder path (as shown), not the actual asmx service URL

The screenshot shows a Windows-style dialog box titled "OptionsForm" with three tabs: "Connection", "Notifications", and "User Token". The "Connection" tab is active. It contains a note: "Note: This is the base URL in which all CPR web services reside. Use 'Test connection' to enable Application Registration". Below the note are two text input fields. The first is labeled "CPR broker services base URL:" and contains the text "http://localhost:1551/Services/". The second is labeled "Event broker services base URL:" and contains the text "http://EventBroker/Services/". To the right of these fields is a button labeled "Test connection". Below these fields is a section titled "Application Registration" which contains a text input field labeled "Admin-application-token" with the value "07059250-E448-4040-B695-9C03F9E59E38". Below this field is a button labeled "Register and approve Application with Web Service". At the bottom of this section is the text "Registration status: Application not yet registered." At the bottom of the dialog box are three buttons: "OK", "Cancel", and "Apply".

If you see the button 'Register and approve ...', press it to create a new application for the demo.

Then Press OK

4.1.1 Notification

To be able to test subscriptions, please enter a valid channel details on the Notifications tab. The channel will be passed whenever you try to create a new subscription.

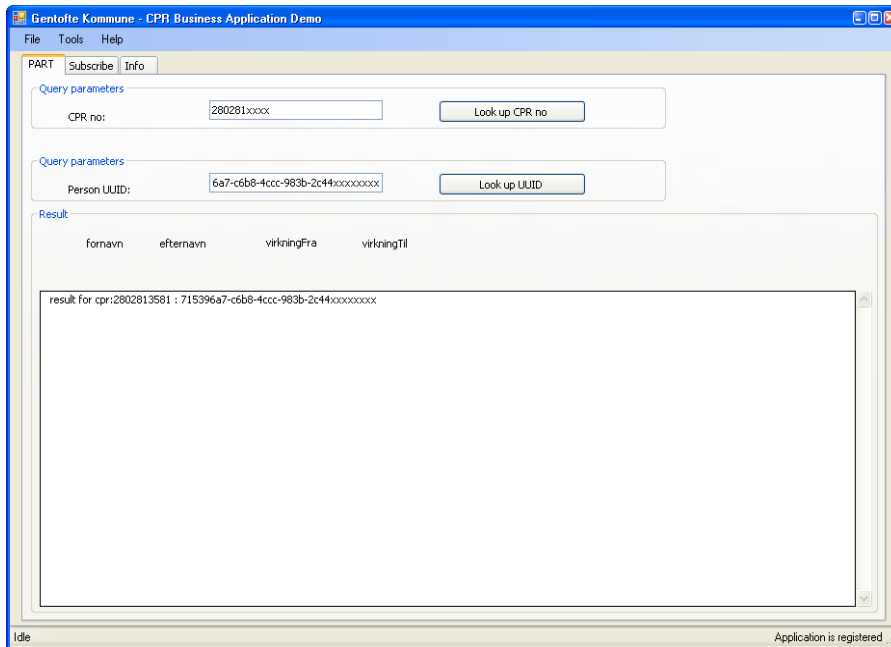
Easiest thing is to use a preexisting folder. This means that new notifications will be created as XML file in that folder.

The screenshot shows a Windows-style dialog box titled "OptionsForm". It has three tabs: "Connection", "Notifications" (which is selected), and "User Token". The "Notifications" tab contains two radio button options. The first option is "Disabled". The second option is "Callback Web Service", which is currently selected. Below this option is a text box labeled "Web Service URL:". The third option is "File Share", which is not selected. Below this option is a text box labeled "File Share:" containing the text "C:\Notofications", and a "Browse..." button to its right. At the bottom of the dialog box are three buttons: "OK", "Cancel", and "Apply".

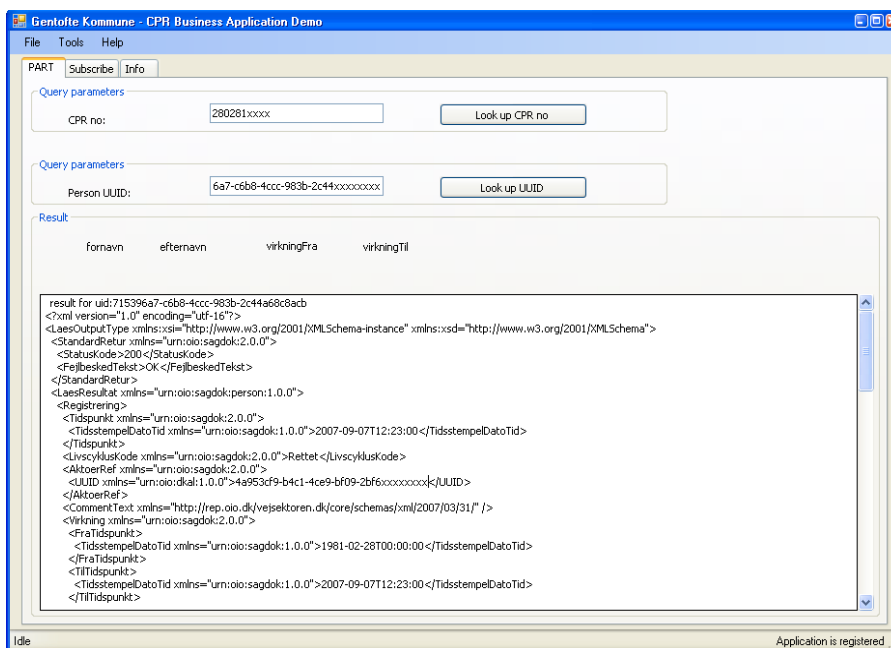
4.2 Using the demo app

4.2.1 Person data

Enter a CPR number and press 'Look up CPR no'. This will call the `GetUuid()` method and the result will be shown.



Now press 'Look up UUID'. You should see the result XML response below



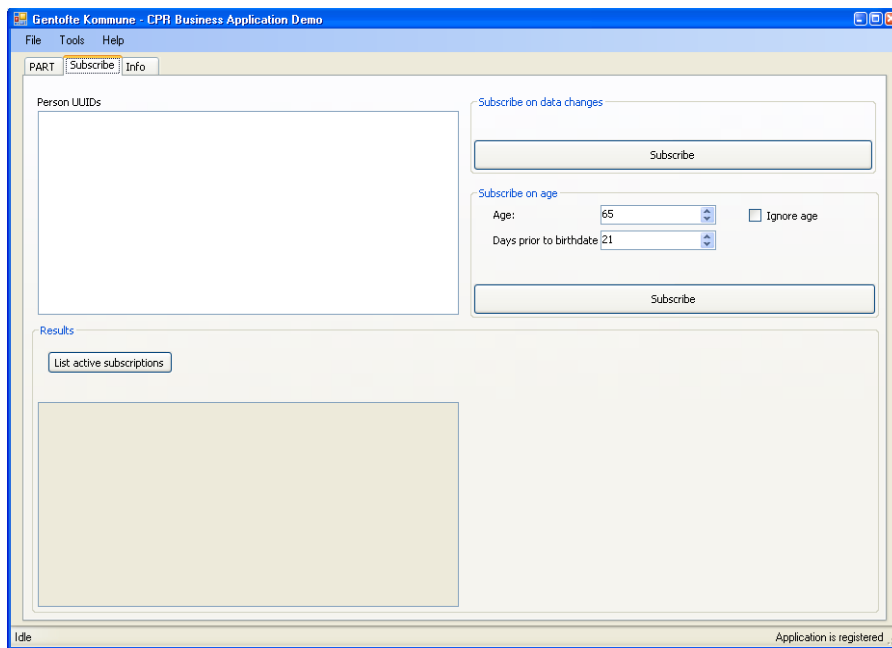
Verify the outcome XML. If you know that person, check the data (name, address,...etc)

4.2.2 Subscriptions

Now open the Subscribe tab.

Use the UUID you obtained previously as input, and then press one of the two 'Subscribe' buttons.

The first button will call the `Subscribe()` method. The second button will call `SubscribeOnBirthdate()` method.



You should see the result in the box at the bottom.