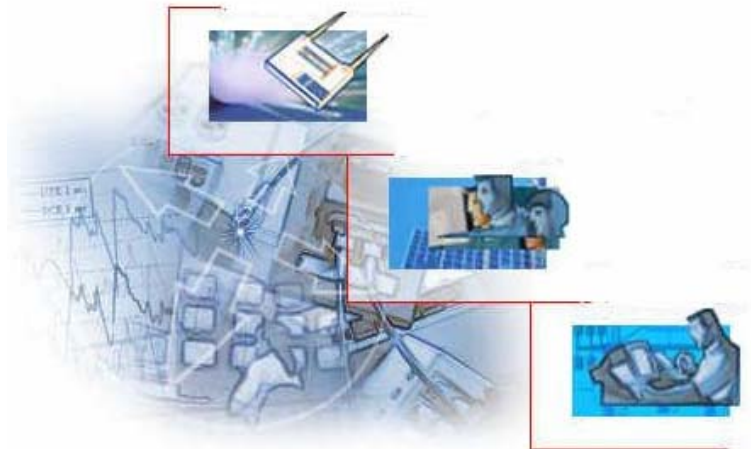




# *Technical Implementation Guide*



**Written by:**

**Proxicom  
11600 Sunrise Valley Drive  
Reston, VA  
20191**

**888.PROXICOM  
[www.proxicom.com](http://www.proxicom.com)**

**Published by:**

**COMCARE  
1701 K Street, N.W.  
Fourth Floor  
Washington, DC  
20006  
202.429.0574  
[www.comcare.org](http://www.comcare.org)**

## TABLE OF CONTENTS

<b>1. Introduction</b>	<b>3</b>
<b>2. Accessing EPAD Web Service</b>	<b>4</b>
<b>Getting Access to EPAD</b>	<b>4</b>
<b>Testing the Web Service</b>	<b>4</b>
<b>3. Using the EPAD Web Service</b>	<b>5</b>
<b>Data Synchronization</b>	<b>5</b>
<i>GetEventTypes</i>	<b>5</b>
<i>GetAgencyTypes</i>	<b>6</b>
<i>GetLevelCodes</i>	<b>6</b>
<i>GetNotificationFormats</i>	<b>7</b>
<b>Single Sign-On</b>	<b>7</b>
<b>Retrieving Agency Information from EPAD</b>	<b>8</b>
<i>GetAgencyIdentification</i>	<b>8</b>
<i>GetAgencyAddresses</i>	<b>9</b>
<i>GetEmailsForAgency</i>	<b>10</b>
<b>Incident Routing Queries</b>	<b>11</b>
<b>4. Using the HTTP POST Method</b>	<b>12</b>
<b>HTTP POST Flow Diagram</b>	<b>12</b>
<b>Receiving EDXL Messages via HTTP POST</b>	<b>13</b>
<b>Handling the EDXL Message Stream</b>	<b>14</b>
<b>Appendices</b>	
<b>A. .NET Code Examples</b>	<b>17</b>
<b>Building .NET Client Code</b>	<b>17</b>
<b>Data Synchronization</b>	<b>18</b>
<b>Single Sign-On and Retrieving Agency Information</b>	<b>19</b>
<b>Incident Routing Queries</b>	<b>20</b>
<b>B. Glossary</b>	<b>21</b>
<b>C. Agency Types</b>	<b>23</b>
<b>D. Event Types</b>	<b>25</b>

The EPAD Technical Implementation Guide was prepared by John Rowland and Jim Lawton of Proxicom for COMCARE on December 6, 2004 and May 6, 2005. Version 1.1 of this document refers to interfacing with the EPAD prototype Web Services.

## 1. Introduction

The purpose of this guide is to explain how to access the EPAD Web Services and integrate your application with the Emergency Provider Access Directory prototype (“EPAD”)<sup>1</sup>. This guide assumes the reader understands the basics of Web service programming and SOAP. It will focus on the use of specific Web Service methods used to implement the following features:

- Data Synchronization (Incident Types, Agency Type, Level Codes)
- Single Sign-on
- Retrieving Agency Information from EPAD (e.g. e-mail address)
- Incident Routing Queries

For each feature, input and output examples are provided and documented. Finally, this guide provides information on how you can create an EPAD Web Service client on the .NET platform.

This guide will also explain how the HTTP POST method for sending EDXL messages works and the steps necessary to receive EDXL messages using EPAD and EPAD Connect. This section assumes the reader understands the basics of the HTTP protocol and is proficient in Web scripting languages needed to write the data-handler for the HTTP POST method.

---

<sup>1</sup> This document refers to the EPAD prototype Web Services. More information is available at <http://www.epad.us>.

## 2. Accessing EPAD Web Services

EPAD, the Emergency Provider Access Directory, is a GIS-enabled database registry of local, state, and federal emergency authorities and public service providers (referred to in general as agencies) to enable instant, interoperable communication and accurate notification of emergency events and related situations. From a Web service client perspective, EPAD provides multiple methods which can be used to query the EPAD database. These methods are accessible via SOAP and HTTP.

### a. Getting Access to EPAD

Before you can access the EPAD database, you must first create a login and password for yourself in EPAD.

---

**Helpful Tip:** For information on obtaining a login and password, please send an e-mail to: [epad-help@comcare.org](mailto:epad-help@comcare.org).

---

### b. Testing the Web Service

Once you have a login and password, you should test your access to the EPAD Web Service by using the Web Service client Web site.

From any Web browser, access the site at the following URL:

<http://webservices.dicecorp.com/epadinterface/epad.asmx>

A list of Web Service methods is displayed. To test your access to EPAD:

1. Click on the **UserAuthentication** link.
2. In the “Login” and “Password” fields, enter your EPAD login and password.
3. A new window will appear. If your login and password are valid, you will see basic information provided in XML format, including:
  - Agency Name
  - Agency UUID (a unique identifier)

## 3. Using the EPAD Web Services

In this section, we will describe how you can use the EPAD Web Service in your application.

---

**Helpful Tip:** In this section you will also see references to <YourEPADLogin> and <YourEPADPassword> where you will need to enter the EPAD login and password you created when you registered with EPAD.

---

### a. Data Synchronization

Several EPAD Web Service methods use various lookup values for input or as output. To avoid getting out of synch with updates to these lookup values, it is recommended that your application use the following Web Services which are provided to allow you to map these lookup values.

#### GetEventTypes

The **GetEventTypes** method provides a mapping between the standard EDXL incident types and their respective lookup value within EPAD. The steps below demonstrate how to access the

**GetEventTypes** method:

1. From any Web browser, access the site at the following URL:  
<http://webservices.dicecorp.com/epadinterface/epad.asmx>
2. Click on the **GetEventTypes** link.
3. Enter the following input values:
  - login: <YourEPADLogin>
  - password: <YourEPADPassword>
4. A new window will appear. If your login and password are valid, you will see a lookup value map is provided in XML format.

(See Appendix A to view sample code that demonstrates how to call the **GetEventTypes** method from an application.)

### GetAgencyTypes

The **GetAgencyTypes** method provides a mapping between agency types and their respective lookup value within EPAD. The steps below demonstrate how to access the **GetAgencyTypes** method:

1. From any Web browser, access the site at the following URL:  
<http://webservices.dicecorp.com/epadinterface/epad.asmx>
2. Click on the **GetAgencyTypes** link.
3. Enter the following input values:
  - login: <YourEPADLogin>
  - password: <YourEPADPassword>
4. A new window will appear. If your login and password are valid, you will see a lookup value map is provided in XML format.

(See Appendix A to view sample code that demonstrates how to call the **GetAgencyTypes** method from an application.)

### GetLevelCodes

The **GetLevelCodes** method provides a mapping between levels of government (state, local, etc) and their respective lookup value within EPAD. The steps below demonstrate how to access the **GetLevelCodes** method:

1. From any Web browser, access the site at the following URL:  
<http://webservices.dicecorp.com/epadinterface/epad.asmx>
2. Click on the **GetLevelCodes** link.
3. Enter the following input values:
  - login: <YourEPADLogin>
  - password: <YourEPADPassword>
4. A new window will appear. If your login and password are valid, you will see a lookup value map is provided in XML format.

(See Appendix A to view sample code that demonstrates how to call the **GetLevelCodes** method from an application.)

### GetNotificationFormats

The **GetNotificationFormats** method provides a mapping between the different notification formats and their respective lookup value within EPAD. The steps below demonstrate how to access the **GetNotificationFormats** method:

1. From any Web browser, access the site at the following URL:  
<http://webservices.dicecorp.com/epadinterface/epad.asmx>
2. Click on the **GetNotificationFormats** link.
3. Enter the following input values:
  - login: <YourEPADLogin>
  - password: <YourEPADPassword>
4. A new window will appear. If your login and password are valid, you will see a lookup value map is provided in XML format.

(See Appendix A to view sample code that demonstrates how to call the **GetNotificationFormats** method from an application.)

### b. Single Sign-On

The EPAD database includes one or more login and password combinations for each agency registered in EPAD. If you are creating a separate application which will use EPAD as a facilitation service and you do not wish to maintain a separate user database, it is possible to leverage the EPAD user database. The EPAD Web Service interface includes a method which will allow you to pass along a login and password entered in your application for authentication. It should be noted that the EPAD prototype does not provide a true token-based “single sign-on” interface, but it does provide a way for a user to create and maintain a single login and password and leverage it across multiple applications.

The single sign-on is possible through the use of the **UserAuthentication** method. The steps below demonstrate how to use the **UserAuthentication** method to manually authenticate a login and password:

1. From any Web browser, access the site at the following URL:  
<http://webservices.dicecorp.com/epadinterface/epad.asmx>
2. Click on the **UserAuthentication** link.
3. Enter the following input values:

- login: <YourEPADLogin>
  - password: <YourEPADPassword>
4. A new window will appear. If your login and password are valid, you will see basic information provided in XML format, including:
- Agency Name
  - Agency UUID (a unique identifier)
  - Agency Type Code

The information returned by the **UserAuthentication** method can be used to enhance the user experience in your application (for example, by displaying the Agency Name returned). More importantly, the **UserAuthentication** method returns a unique identifier (Agency UUID) which is required by other methods in the EPAD Web Service interface to retrieve information. It is recommended therefore that you save the UUID returned in your application during the user session for future reference.

(See Appendix A to view sample code that demonstrates how to initiate the **UserAuthentication** method on a login page within your application.)

### **c. Retrieving Agency Information from EPAD**

Once you have leveraged the **UserAuthentication** method to authenticate a login and password against the EPAD database, there are additional methods you can leverage to access information about the Agency associated with the login ID. All of these methods require the Agency UUID (described above) in order to retrieve agency information.

You may find the following EPAD Web Service methods useful to your application. Additional agency information methods are also available on EPAD, however the ones below are more typically used in an application.

## GetAgencyIdentification

The **GetAgencyIdentification** method allows you to retrieve several bits of information regarding an agency saved within EPAD. The steps below demonstrate how to use the **GetAgencyIdentification** method:

1. From any Web browser, access the site at the following URL:  
<http://webservices.dicecorp.com/epadinterface/epad.asmx>
2. Click on the **GetAgencyIdentification** link.
3. Enter the following input values:
  - login: <YourEPADLogin>
  - password: <YourEPADPassword>
  - agencyUuid: a9f6eb23-2c09-4828-ae5e-4c0817ec000f

---

**Helpful Tip:** Please note that in the following examples, we will use a demo Agency UUID. In practice, you would use the Agency UUID you received via the **UserAuthentication** method.

---

4. A new window will appear. If your input values are valid, you will see Agency information provided in XML format, including Agency Type Description and Level of Government information.

(See Appendix A to view sample code that demonstrates how to call the **GetAgencyIdentification** method from an application.)

## GetAgencyAddresses

The **GetAgencyAddresses** method allows you to retrieve any physical location information regarding an agency saved within EPAD. The steps below demonstrate how to use the **GetAgencyAddresses** method:

1. From any Web browser, access the site at the following URL:  
<http://webservices.dicecorp.com/epadinterface/epad.asmx>
2. Click on the **GetAgencyAddresses** link.
3. Enter the following input values:
  - login: <YourEPADLogin>
  - password: <YourEPADPassword>
  - agencyUuid: a9f6eb23-2c09-4828-ae5e-4c0817ec000f

4. A new window will appear. If your input values are valid, you will see Agency information provided in XML format, including Agency Address information.

(See Appendix A to view sample code that demonstrates how to call the **GetAgencyAddresses** method from an application.)

### GetEmailsForAgency

The **GetEmailsForAgency** method allows you to retrieve any e-mail information regarding an agency saved within EPAD. The steps below demonstrate how to use the **GetEmailsForAgency** method:

1. From any Web browser, access the site at the following URL:  
<http://webservices.dicecorp.com/epadinterface/epad.asmx>
2. Click on the **GetEmailsForAgency** link.
3. Enter the following input values:
  - login: <YourEPADLogin>
  - password: <YourEPADPassword>
  - agencyUuid: a9f6eb23-2c09-4828-ae5e-4c0817ec000f
4. A new window will appear. If your input values are valid, you will see Agency information provided in XML format, including Agency E-Mail information.

(See Appendix A to view sample code that demonstrates how to call the **GetEmailsForAgency** method from an application.)

## d. Incident Routing Queries

One of the primary functions of EPAD is to provide incident routing query services. Incident routing query services always include a GIS component, but they can also include additional filters (such as incident type) as well.

The EPAD Web Service includes several different types of incident routing queries. All of these methods are variations on the same theme. For purposes of this guide, we will focus on the one method that includes all other possible filters. From this single method you can decide if all of these filters will meet your needs. If not, you can

choose to research other methods on the EPAD Web Service which provide fewer filter options.

---

**Helpful Tip:** For all incident routing queries in EPAD, you must provide GIS coordinates in Well Known Text (WKT) format. For more information on WKT format, check out: [http://mysqld.active-venture.com/GIS\\_WKT\\_format.html](http://mysqld.active-venture.com/GIS_WKT_format.html)

---

The steps below demonstrate how to use the **GetAgenciesForMultipleEverything** method to manually perform an incident routing query:

1. From any Web browser, access the site at the following URL:  
<http://webservices.dicecorp.com/epadinterface/epad.asmx>
2. Click on the **GetAgenciesForMultipleEverything** link.
3. Enter the following input values:
  - login: <YourEPADLogin>
  - password: <YourEPADPassword>
  - gisCoordinates: gisCoordinates POINT(-77.3313 38.9624)
  - levelCodes: 1,2,3,4,5,6
  - incidentCodes:  
1001,1002,1003,1010,1012,1013,1015,1019,1020,1021,1022,1023,1024,1025,1026,1027,1028,1030,1031,1032,1033,1034,1035,1036,1038,1039,1040,1041,1042,1043,1044,1045,1046
  - agencyCodes:  
1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29
4. A new window will appear. If your input values are valid, you will see a list of agencies to notify provided in XML format, including:
  - Agency Name
  - Agency UUID (a unique identifier)
  - Agency Type Code
  - Notification URL(s)

(See Appendix A to view sample code that demonstrates how to initiate the **GetAgenciesForMultipleEverything** method on a login page within your application.)

## 4. Using the HTTP POST Method

### a. HTTP POST Flow Diagram

Figure 4.1 illustrates the interaction between the user, source agency system (e.g. EPAD Connect) and EPAD.

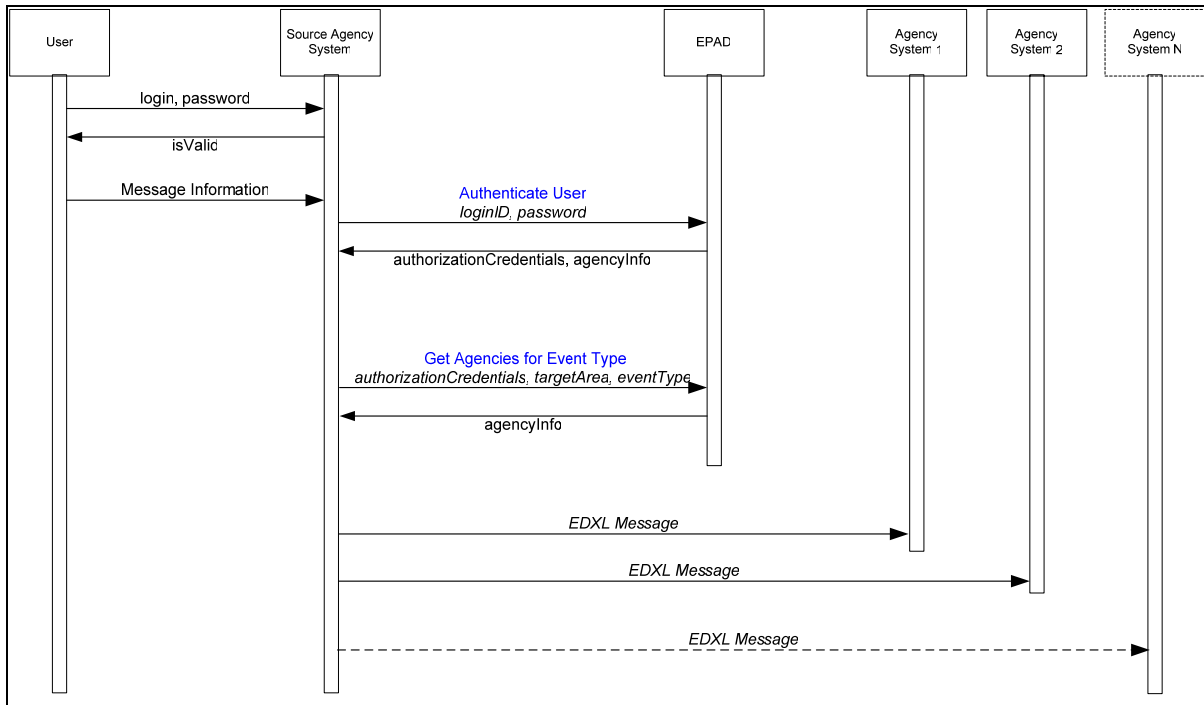


Figure 4.1: HTTP POST Flow Diagram.

In this scenario, a user logs in to the Source Agency System and is successfully authenticated. The user then sends the Source Agency System the Message Information. The Source Agency System then logs in to EPAD and passes EPAD the Event Type to get a list of agencies that need to be notified.

The Source Agency System goes through the list of agencies and, for agencies that are registered to receive EDXL messages using HTTP POST, sends out the EDXL message to each agency's data-handling process.

## b. Receiving EDXL Messages via HTTP POST

EPAD allows users to select the method they want used to deliver EDXL messages. One of these is the HTTP POST method. The HTTP POST method sends the EDXL message as a data stream to a data-handling process (URL).

In order to receive EDXL messages using the HTTP POST method, you need to update your EPAD account and register the URL for your data-handling process.

### Step 1: Log into EPAD.

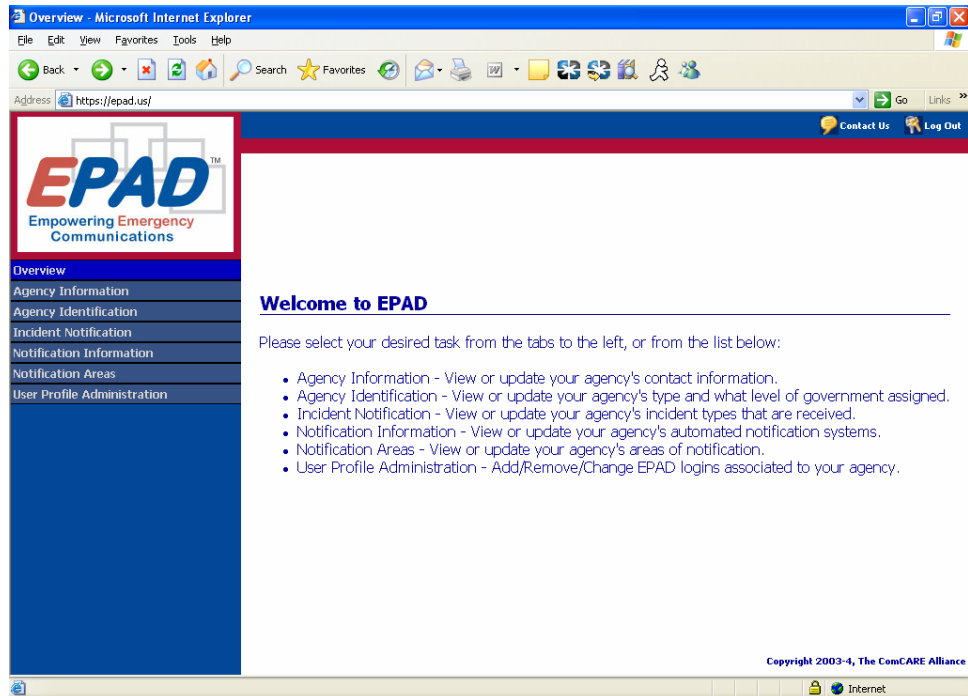
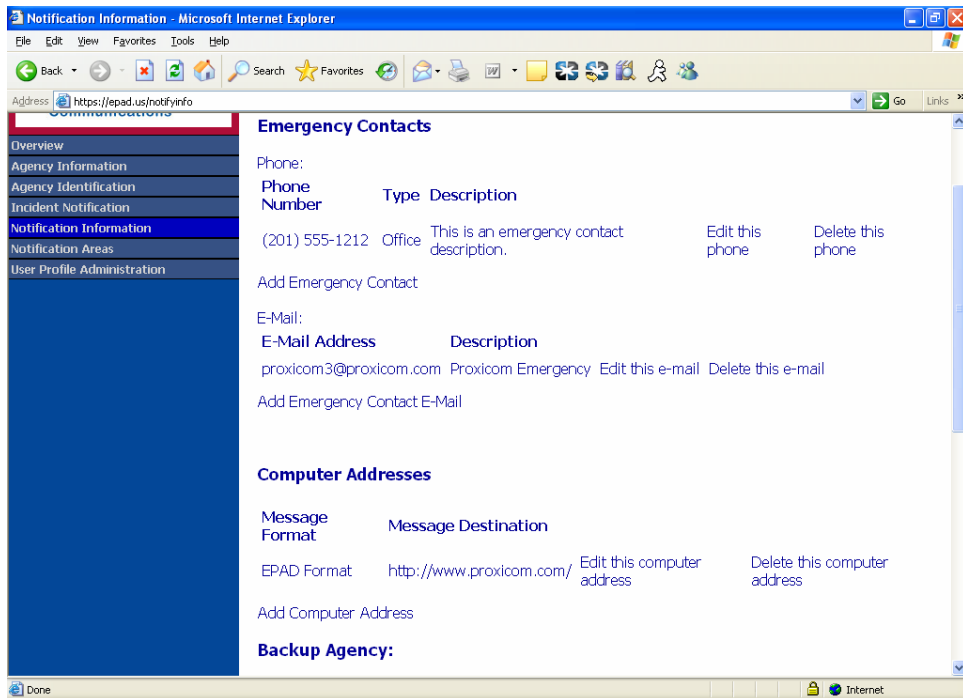


Figure 4.2: EPAD Overview Screen.

**Step 2: Select Notification Information in the left column.**



**Figure 4.3: EPAD Notification Information screen.**

**Step 3: Select Add Computer Address.**

**Computer Address:**

Message Format    Message Destination

EDXL HTTP

**Figure 4.4: Add Computer Address.**

**Step 4:** In the **Message Format** drop down, select EDXL HTTP. In the **Message Destination** field, enter the URL for the data-handling process used by your system to receive EDXL messages via HTTP POST.

## c. Handling the EDXL Message Stream

The data-handling process is a script that extracts EDXL messages from the POST and then parses and processes the data. The script can be written using a variety of languages including, but not limited to, ASP, JavaScript and PHP. It is up to the user to write the data-handling script. EPAD Connect will send the using HTTP POST to any valid URL that is publicly accessible on the Internet (i.e. you cannot use NAT addresses or Intranet addresses).

The following is an example of an EDXL message stream sent by EPAD Connect. For details on the EDXL specification read the *Emergency Data Exchange Language Standard Message Format* (draft 9/23/2004).

```
POST / HTTP/1.1
Content-Type: text/xml
Content-Length: 1984
Expect: 100-continue
Connection: Keep-Alive
Host: 24.150.239.251

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
  <soap:Envelope xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
    soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
    <soap:Header>
      <e:Distribution xmlns:e=http://www.dhs.gov/edxl/smf/0.9
        soap:relay="true">
        <e:messageID>EPADConnect:613</e:messageID>
        <e:senderID>proxicom1@proxicom.com</e:senderID>
        <e:senderType>Proxicom</e:senderType>
        <e:dateTimeSent>2005-02-01T17:04:30</e:dateTimeSent>
        <e:messageStatus>Exercise</e:messageStatus>
        <e:messageType>Report</e:messageType>
        <e:messageFormat>http://www.incident.com/cap/1.0</e:messageFormat>
        <e:eventType>SEC-AMBER</e:eventType>
        <e:targetArea>
          <e:polygon>
            38.927900220638094,-77.02167829520069 38.927900220638094,
            -76.97469373986735 38.897695863638084,-76.97469373986735
            38.897695863638084,-77.02167829520069 38.927900220638094,
            -77.02167829520069
          </e:polygon>
        </e:targetArea>
      </e:Distribution>
    </soap:Header>
    <soap:Body>
      <cap:alert xmlns:cap="http://www.incident.com/cap/1.0">
        <cap:identifier>613</cap:identifier>
        <cap:sender>proxicom1@proxicom.com</cap:sender>
        <cap:sent>2005-02-01T17:04:30</cap:sent>
        <cap:status>Exercise</cap:status>
        <cap:msgType>Alert</cap:msgType>
      </cap:alert>
    </soap:Body>
  </soap:Envelope>
```

```
<cap:scope>Public</cap:scope>
<cap:info>
  <cap:language>en-US</cap:language>
  <cap:category>Security</cap:category>
  <cap:event>SEC-AMBER</cap:event>
  <cap:urgency>Expected</cap:urgency>
  <cap:severity>Severe</cap:severity>
  <cap:certainty>Likely</cap:certainty>
  <cap:senderName>Proxicom</cap:senderName>
  <cap:headline>Message Headline</cap:headline>
  <cap:description>Message Description</cap:description>
  <cap:instruction>Requested Action</cap:instruction>
  <cap:area>
    <cap:areaDesc>Incident Location</cap:areaDesc>
    <cap:polygon>
      38.927900220638094,-77.02167829520069 38.927900220638094,
      -76.97469373986735 38.897695863638084,-76.97469373986735
      38.897695863638084,-77.02167829520069 38.927900220638094,
      -77.02167829520069
    </cap:polygon>
  </cap:area>
  <cap:contact>proxicom1 PHONE: FAX: </cap:contact>
</cap:info>
</cap:alert>
</soap:Body>
</soap:Envelope>
```

## Appendix A: .NET Code Examples

In this Appendix are code examples provided in the Microsoft .NET C# language. They are intended to help demonstrate how you would access the EPAD Web Service programmatically.

### Building .NET Client Code

Once you have confirmed your login and password to the EPAD database, you can proceed with creating the Web Service client-side code you will need to access the EPAD web service programmatically. The instructions in this section focus on using Microsoft Visual Studio .NET to generate the Web Service client-side code. Because the EPAD Web Service uses a standard WSDL interface, you should be able to follow a similar procedure in your development environment of choice to auto-generate the Web Service client-side code.

The steps below focus on generating static Web Service client-side stubs. Prior to generating these stubs, we recommend downloading the latest version of the Web Services Enhancements (WSE) for Microsoft .NET, which can be accessed at the following URL:

<http://msdn.microsoft.com/webservices/building/wse/default.aspx>

After installing WSE, you will now need to generate the Web Service client code. From your start menu, find the folder for Visual Studio .NET, and look for the .NET command prompt icon, which yields a command prompt. At the command prompt, you will now generate the Web Service client code. In this example, C# is the .NET language being used, but they are all very similar.

Type:

```
wsdl /language:cs http://webservices.dicecorp.com/epadinterface/epad.asmx?WSDL
```

Once completed, a new file labeled "EPADWebServices.cs" will be generated. To use this file in your project, you must cut and paste the file from the current directory into your project directory. From Microsoft Visual Studio you can then add this file to your project.

---

**Helpful Tip:** Please note, from this point forward, the documentation will assume you have generated the Web Service client code as described above and that you have included the output file into your project.

---

## Data Synchronization

Because there are lookup values in EPAD which do not change often but you will need when you access other queries, it is recommended that you load these values during application startup into your application memory (for example, in your Global.asax file). The following code examples show how to load the data synchronization methods during application startup:

```
EPADWebServicesClient WebServicesClient = new
EPADWebServicesClient();

protected void Application_Start(Object sender, EventArgs e)
{
    string username = "<YourEPADLogin>";
    string password = "<YourEPADPassword>";
    try
    {
        // --Load EPAD variables
        Application["EPAD.AgencyTypes"] =
        WebServicesClient.GetAgencyTypes_EPAD(username, password);

        Application["EPAD.IncidentTypes"] =
        WebServicesClient.GetEventTypes_EPAD(username, password);

        Application["EPAD.LevelCodes"] =
        WebServicesClient.GetLevelCodes_EPAD(username, password);
    }
    catch (Exception err)
    {
        string errMsg;
        errMsg = err.Message;
    }
}
```

## Single Sign-On and Retrieving Agency Information

Single sign-on can easily be tied to a login form within your own application. Once you have called the EPAD **UserAuthentication** method, you can save the data received into the application session for future reference. In the same location where you authenticate the user, it is also recommended that you use additional EPAD Web Service methods for retrieving Agency Information. The following code examples show how to call the **UserAuthentication** method to authenticate the user and then use the Agency UUID to retrieve other agency information from EPAD as well:

```
EPADWebServicesClient WebServicesClient = new EPADWebServicesClient();

private void UserAuthentication_EPAD()
{
    AgencyInfo myAgencyInfo;
    try
    {
        // --Authenticate the user
        myAgencyInfo =
myEPADWebServ.UserAuthentication(txtUsername.Text.Trim(),txtPassword.
Text.Trim());

        // -- Set user session variables
        Session["User.Username"] = txtUsername.Text.Trim().ToString();
        Session["User.password"] = txtPassword.Text.Trim().ToString();

        Session["User.AgencyName"] =
myAgencyInfo.AgencyName.ToString();
        Session["User.AgencyTypeCode"] =
myAgencyInfo.AgencyTypeCode.ToString();
        Session["User.AgencyUuid"] =
myAgencyInfo.AgencyUuid.ToString();

        Session["User.AgencyIncidents"] =
WebServicesClient.GetAgencyIncidents_EPAD((string)Session["User.Usern
ame"],(string)Session["User.password"],(string)Session["User.AgencyUu
id"]);

        Session["User.AgencyContactEmails"] =
WebServicesClient.GetContactEmails_EPAD((string)Session["User.U
sername"],(string)Session["User.password"],(string)Session["Use
r.AgencyUuid"]);
    }
}
```

## Incident Routing Queries

As discussed previously, the **GetAgenciesForMultipleEverything** method provides the most options for performing an incident routing query. The following code examples show how to call the **GetAgenciesForMultipleEverything** method:

```
EPADWebServicesClient WebServicesClient = new
EPADWebServicesClient();

private void GetAgencies_EPAD(string gisCoordinates, string
levelCodes, string incidentCodes, string agencyCodes)
{
    try
    {
        // ---- call the webservice and return recipients
        Session["User.MessageRecipients"] =
        WebServicesClient.GetAgenciesForMultipleEverything_EPAD(
            (string)Session["User.Username"],
            (string)Session["User.password"],
            gisCoordinates,levelCodes,incidentCodes,agencyCodes);
    }
    // -- Authentication failed
    catch(SoapHeaderException eSoap)
    {
        lblError.Visible = true;
        lblError.Text = eSoap.Message.ToString();
    }
}
```

## Appendix B: Glossary

There are several terms in this document which are particular to EPAD or the EPAD Web Services that need to be explained. A brief explanation is found below. Note that terms relating to web services or SOAP in general are not found here; for a glossary on Web-services-related terms, see <http://www.w3.org/TR/ws-gloss/>.

<b>Term</b>	<b>Definition</b>
<b><i>Agency Address</i></b>	The physical location information for an agency
<b><i>Agency Identification</i></b>	The Agency Type Description and Level of Government for an agency
<b><i>Agency Name</i></b>	Official name of the agency
<b><i>Agency Type</i></b>	Category that describes an agency's function (See Appendix C)
<b><i>Agency Type Code</i></b>	A unique identification for each Agency Type
<b><i>Event Type</i></b>	Same as Incident Type
<b><i>Geographic Area</i></b>	Area for which an agency has some responsibility, designated by a polygon or set of polygons
<b><i>Geographic Information Systems (GIS)</i></b>	A database system with software that can analyze and display data using digitized maps and tables for planning and decision-making. Any location with known latitude and longitude or other geographic grid system can be a part of GIS.  EPAD is a GIS-enabled database which means that you must provide geographic data (for example, a longitude and latitude) to return results.
<b><i>Incident Type</i></b>	The types of emergency events of which the agency wishes to receive notification, same as Event Type.
<b><i>Incident Type Code</i></b>	A unique identifier for each Incident Type
<b><i>Incident Type Description</i></b>	Same as Incident Type
<b><i>Level Codes</i></b>	A unique identifier for each Level of

## Term

## Definition

	Government (national, state, local, etc.)
<b>Level of Government</b>	Describes the extent of responsibility of an agency
<b>Notification Format Code</b>	A unique identifier for each Notification Type
<b>Notification Format</b>	Notification Format is an internal classification of the EPAD prototype for routing purposes, and is application specific. If your application is not listed, then you need to use the 'EPAD Format'
<b>Notification Information</b>	Notification type, notification computer addresses
<b>Notification Format Description</b>	Same as Notification Format
<b>UUID</b>	A unique user identifier used by EPAD for each agency registered. The UUID is important for accessing specific information about an agency stored in EPAD.
<b>Well-Known Text (WKT)</b>	<p>The Well-Known Text (WKT) representation of Geometry is designed to exchange geometry data in ASCII form. Examples of WKT representations of geometry objects are:</p> <ul style="list-style-type: none"><li>▪ POINT(15 20)</li><li>▪ POLYGON((0 0,10 0,10 10,0 10,0 0),(5 5,7 5,7 7,5 7, 5 5))</li></ul> <p>For more information, check out: <a href="http://mysqld.active-venture.com/GIS_WKT_format.html">http://mysqld.active-venture.com/GIS_WKT_format.html</a></p>

## Appendix C: Agency Types

The Agency Types in Appendix B are the Agency Type values used in EPAD, and are based on those in the EDXL draft standard.

<b>Agency Type</b>	<b>Example Agencies</b>
<b>911 / Public Safety Answering Point</b>	Primary PSAP, Secondary PSAP, Transfer PSAP
<b>Agriculture</b>	Department of Agriculture, Extension Offices
<b>Animal Welfare / Service / Veterinary</b>	Animal Control, Human Society
<b>Coast Guard</b>	Coast Guard
<b>Commercial</b>	Corporations, incl. Oil and Petroleum, Chemical Manufacturing, Software Companies, Computer Services Companies
<b>Educational</b>	Department of Education, Administration, School Boards, Higher Education, Preschool, Elementary and Secondary Education, Schools, public and private
<b>Electric Utility</b>	Electric Companies, Power plants and stations
<b>Emergency Management</b>	Emergency Management, Department of Public Safety
<b>Emergency Medical</b>	Emergency Medical Services, Department of Public Safety, Air Medical Services, Public and Private Ambulance
<b>Emergency Support</b>	American Red Cross, other Emergency Support Organizations
<b>Environmental</b>	Environmental Protection Agency, State Environmental Departments
<b>Financial</b>	Financial and Insurance Institutions
<b>Fire / Rescue</b>	Fire and EMS, Department of Public Safety, Transportation Authority or Airport FD, Private or Industrial Fire Brigade, Contract Fire Department
<b>Geophysical</b>	USGS, NASA, Earthquake Hazards Program Earthquake Centers, National Geophysical Data Center
<b>Government</b>	City, County, State, Federal or Tribal Supervisory
<b>Hazardous Materials</b>	DOT – Hazmat, EPA, Chemical Hazard and Safety Investigation Board
<b>Homeland Security</b>	Department of Homeland Security, DHS – FEMA, Department of Public Safety, Border Control
<b>Hospital / Healthcare</b>	Hospitals, Healthcare Facility, Clinics, Medical Offices, Nursing Homes, Urgent Care
<b>Individual / Household</b>	Individual / Household
<b>Information Sharing and Analysis Center</b>	Information Sharing and Analysis Center (ISAC)

## Agency Type

## Example Agencies

<b><i>Intelligence</i></b>	National Security Administration, DOJ – Federal Bureau of Investigation, DOD – Defense Intelligence Agency, DOD – National Geospatial Intelligence Agency
<b><i>Law Enforcement</i></b>	Law Enforcement, Department of Justice, DOJ – Bureau of Alcohol, Tobacco, Firearms, and Explosives, DOJ – Drug Enforcement Administration (DEA), DOJ – Federal Bureau of Investigation, Department of Public Safety
<b><i>Mass Media</i></b>	Television, Radio, Newspaper, Internet
<b><i>Meteorological</i></b>	National Weather Service, NOAA
<b><i>Military</i></b>	Department of Defense, DOD – Defense Intelligence Agency, DOD – National Geospatial Intelligence Agency
<b><i>National Guard</i></b>	National Guard
<b><i>Natural Gas Utility</i></b>	Natural Gas Utility
<b><i>Non-governmental / Community Organization</i></b>	Community Emergency Response Teams, Citizen Councils
<b><i>Park Service / Department</i></b>	United States National Park Service, State Parks Departments
<b><i>Poison Control Center</i></b>	Poison Control Center
<b><i>Recreation</i></b>	Stadiums, Malls, Event Venues, Amusement Parks, Auditoriums
<b><i>Sanitation and Sewer</i></b>	Sanitation and Sewer
<b><i>Search and Rescue</i></b>	Search and Rescue
<b><i>Social Services (inc. psychological)</i></b>	Child Protective Services
<b><i>Telecommunications</i></b>	Telephone Providers, Wireless Carriers, Internet Providers
<b><i>Transportation</i></b>	Department of Transportation, DOT – Hazmat, Train Stations, Airports, Roadside Assistance Provider, Commercial Vehicle Operator, Port Authority, Transportation Administration, Air, Rail Shipping, Traffic Management, Trucking
<b><i>Warning Agency / Service (non-commercial)</i></b>	NOAA, Emergency Alert System
<b><i>Water Utility</i></b>	Water Utilities, including Water Treatment Facilities

## Appendix D: Event Types

The Event Types in Appendix D are the Incident Event Type values used in EPAD, and are based on those in the EDXL draft standard. They are sorted by the Message Topic they are associated with in the application.

### **Geophysical Events**

- Earthquake
- Volcano
- Tsunami / Tidal Wave
- Landslide
- Sinkhole / Subsidence
- Glacier / Iceberg
- Extraterrestrial Event
- Geophysical - Not Otherwise Categorized

### **Meteorological Events**

- Tornado
- Hurricane / Tropical Storm
- Severe Weather
- Flooding
- Flash Flooding
- Avalanche
- Meteorological - Not Otherwise Categorized

### **Law / Security Events**

- Abduction/Kidnapping
- Missing Person
- AMBER Alert
- Violent Crime
- Non-violent Crime
- Vehicle Theft
- Theft of Sensitive Property
- Cyber Crime
- Access Control / Restriction
- Traffic Control /Restriction
- Border Control Incident / Restriction
- Civil Disturbance
- Terrorist or Criminal Threat
- Law / Security - Not Otherwise Categorized

### **Rescue Events**

- Confined Space Rescue / Urban Search & Rescue

Wildland Search and Rescue  
Vehicle Extrication / Rescue  
Water Rescue  
Marine Search and Rescue  
Recovery (Casualty)  
Salvage / Property Recovery  
Rescue - Not Otherwise Categorized

**Fire Events**

Structure Fire  
Wildfire  
Wildland Fire Use  
Vehicle Fire  
Prescribed Fire / Controlled Burn  
Debris / Product Fire  
Burned Area Emergency Rehabilitation  
Explosion  
Fire Alarm  
Fire Conditions Watch / Fire Weather Watch / "Red Flag" Alert  
Fire - Not Otherwise Categorized

**Medical and Public Health Events**

Public Health Alert / Watch  
Public Health Hazard / Warning  
Medical Emergency - Individual  
Medical Emergency - Multiple Casualty  
Medical Emergency - Mass Casualty  
Medical Facility Bypass  
Health / Medical - Not Otherwise Categorized

**Environmental Events**

Chemical Hazard  
Biological Hazard  
Radiological Hazard  
Oil Spill - Land  
Oil Spill - Waterway Involved  
Environmental Conditions Watch  
Environmental Hazard Warning  
Agricultural Hazard  
Environmental - Not Otherwise Categorized

**Transportation Events**

Air Crash / Accident  
Rail Crash / Accident

Waterway Collision / Accident  
Commercial Water Transport Collision / Accident  
Roadway Collision / Accident  
Commercial Roadway Collision / Accident  
Aircraft Hijacking  
Rail Hijacking  
Vessel Hijacking  
Roadway Hijacking  
Aircraft Emergency  
Rail Emergency  
Waterway Emergency  
Roadway Emergency  
Airway Disruption or Delay  
Railway Disruption or Delay  
Waterway Disruption or Delay  
Roadway Disruption or Delay  
Mass Transit Disruption or Delay  
Transportation - Not Otherwise Categorized

**Infrastructure Events**

Building / Structure Collapse  
Telecommunications Disruption  
Electrical Utility Disruption  
Electrical Line Down or Exposed  
Fuel Supply Disruption  
Natural Gas Leak  
Water Supply Disruption  
Food Supply Disruption  
Information Network Disruption  
Mass Media Disruption  
Financial Services Disruption  
Government Services Disruption  
Infrastructure - Not Otherwise Categorized

**Other Events**

9-1-1 Call  
Request for Assistance  
Planned Event  
Resource Status Report  
Preparedness / Mobilization  
Other - Not Otherwise Categorized