



# **IPDR/XML File Encoding Format**

**Version 3.5.0.1**

**November, 2004**

**© 1999-2004 IPDR.org, Inc.**

## **Table of Contents**

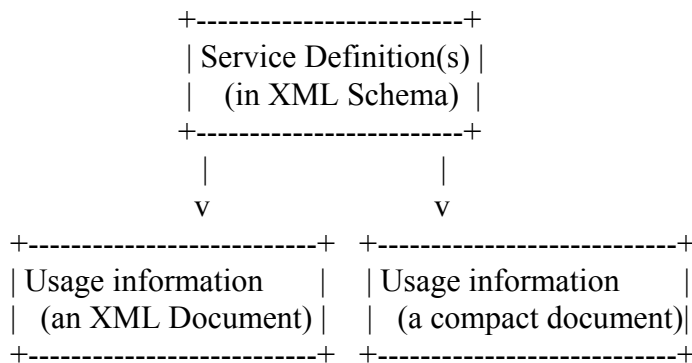
1	Introduction	3
2	XML Encoding Definition	4
2.1	Example IPDR/Service Definition	4
2.2	Example IPDR Document Instance	5
2.2.1	Discussion of Example Instance Document	6
3	Applying IPDR/XML in WSDL	7
4	Applying IPDR to other Legacy Protocols	8

## 1 Introduction

IPDR/XML is an encoding format that directly follows the W3C's XML and XML-Schema specification's rules for encoding well-formed and valid XML documents. IPDR also defines a more compact encoding format called IPDR/XDR which is binary in nature. The decision to use IPDR/XML vs. IPDR/XDR is dependent on the use context. For exchange with IPDR/XDR capable systems, the use of IPDR/XDR is recommended. For exchange with the much larger set of XML and XML-Schema compliant systems or when it is desirable to produce encodings which are directly human readable, or when used in conjunction with XML processing tools such as XSL for translation to other ASCII or XML formats, the use of IPDR/XML is recommended.

Both IPDR/XML and IPDR/XDR are driven by the same XML-Schema and can easily be converted without loss of information in either direction. Open source reference implementations further facilitate the use of IPDR/XDR as a compact machine-to-machine mechanism for exchange.

The relationship between IPDR/XDR and IPDR/XML is shown below.



## 2 XML Encoding Definition

IPDR/XML encoding is simply a basic XML encoding which is both well formed and valid according to the rules in XML-Schema.

IPDR/Service Definitions are written as XML-Schema documents which follow the guidelines set forth in the IPDR Service Specification Guidelines.

IPDR/Service Definitions make heavy use of the inherent ability of XML-Schema to support extensions through the use of the <import> mechanism and the ability to extend existing <complexType> definitions. IPDR/Service definitions also use the <annotation> mechanism defined in XML-Schema to add additional meta data which does not directly impact the encoding of documents, but provides semantic details which can be helpful for tools which manipulate IPDR documents.

In particular, IPDR defines a base XML-Schema from which all compliant IPDR/Service Definitions extend. The base IPDR Schema defines some additional primitive types which may be used by IPDR/Service Definitions, a base <complexType> from which all other <complexTypes> which use IPDR derive and an overall document model which can be used to group a collection of IPDR compliant records.

Encoding a document from one or more IPDR/Service Definitions is done by using the base <IPDRDoc> element and populating it with XML elements which are valid according to the XML-Schema rules implied by the various IPDR/Service Definitions.

### 2.1 Example IPDR/Service Definition

As a simple example, consider a single IPDR/Service Specifications Guideline Service Definition as shown below (some recommended annotations are left off to improve readability).

```
<schema xmlns="http://www.w3.org/2001/XMLSchema"
  xmlns:ipdr="http://www.ipdr.org/public/IPDR"
  targetNamespace="http://www.foo.com/public/aaservice"
  xmlns:foo="http://www.foo.com/ipdr/namespace">

  <import namespace="http://www.ipdr.org/public/IPDR"
    schemaLocation="http://www.ipdr.org/public/IPDRDoc3.1.xsd"/>

  <element name="subscriberId" type="string">
    <annotation>
      <documentation>
        The operator's defined identifier used for naming this subscriber.
      </documentation>
    </annotation>
  </element>

  <element name="ipAddress" type="ipdr:ipV4Address">
    <annotation>
      <documentation>
        The IP address used by this subscriber at the time of this event.
      </documentation>
    </annotation>
  </element>

  <element name="nasIdentifer" type="string">
    <annotation>
      <documentation>
        The network access point name associated with this subscribers access.
      </documentation>
    </annotation>
  </element>
```

```

    </documentation>
  </annotation>
</element>

<element name="acctInputOctets" type="integer">
  <annotation>
    <documentation>
      The number of bytes the subscriber received over the connection.
    </documentation>
  </annotation>
</element>

<element name="acctOutputOctets" type="integer">
  <annotation>
    <documentation>
      The number of bytes the subscriber sent over the connection.
    </documentation>
  </annotation>
</element>

<complexType name="AA-Type">
  <complexContent>
    <extension base="ipdr:IPDR"\>
      <element ref="subscriberId"\>
      <element ref="ipAddress"\>
      <element ref="nasIdentifier"\>
      <element ref="acctInputOctets"\>
      <element ref="acctOutputOctets"\>
    </extension>
  </complexContent>
</complexType>
</schema>

```

Note that the example schema imports the base IPDR Schema definition. This base schema defines both the IPDRDoc and IPDR elements which form the basic structure of an IPDR document. The specialized schema for the foo service defines the additional details which are extend the basic <IPDR> element.

## 2.2 Example IPDR Document Instance

A resulting well formed and valid document which is defined by the previous IPDR/Service Definition is shown below:

```

<ipdr:IPDRDoc
  xmlns:ipdr="http://www.ipdr.org/public/IPDR"
  xmlns="http://www.foo.com/ipdr/namespace"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.foo.com/public/aaservice
    http://www.foo.com/public/aaservice.xsd"
  docId="f81d4fae-7dec-11d0-a765-00a0c91e6bf6"
>
  <ipdr:IPDR xsi:type="AA-Type">
    <subscriberId>joe</subscriberId>
    <ipAddress>192.168.2.64</ipAddress>
    <nasIdentifier>nas1.foo.com</nasId>
    <acctInputOctets>13444</acctInputOctets>
    <acctOutputOctets>77777</acctOutputOctets>
  </ipdr:IPDR>
  <ipdr:IPDR xsi:type="AA-Type">
    <subscriberId>fred</subscriberId>
    <ipAddress>192.168.6.66</ipAddress>
    <nasIdentifier>nas2.foo.com</nasId>
    <acctInputOctets>81222</acctInputOctets>
    <acctOutputOctets>5712392</acctOutputOctets>
  </ipdr:IPDR>
</ipdr:IPDRDoc>

```

### 2.2.1 Discussion of Example Instance Document

The document follows the rules defined in XML and XML-Schema for creating a well formed and valid document according to an IPDR/Service Definition, which is a valid XML Schema.

The document itself consists of the root element <ipdrDoc> which is defined in the base IPDR Schema file. References to the various namespaces for the elements which will appear in this document instance are also contained in the <ipdrDoc> element via the xmlns:xxx attributes.

Finally a hint is given via the <schemaLocation> tag as to where the schema file itself may be located. This makes the document completely self describing, including the additional information contained in the <annotation> markup of the IPDR/Service Definition.

The document may then contain some optional attributes in the <IPDRDoc> element and an optional <IPDRDoc.end> element. These optional fields are omitted from this example instance.

### **3 Applying IPDR/XML in WSDL**

Because of the linkage with the standard W3C XML-Schema mechanism, subsets of IPDR/Service Definitions may be reused with web services defined by WSDL.

In particular, WSDL allows the importation of XML-Schema documents to define the structure of XML elements carried by WSDL defined web services.

The combination of WSDL and XML-Schemas which are compliant with the IPDR/Service Specification Guidelines offer a means of unlimited extension of web services and a means to record their interactions as either IPDR/XML or IPDR/XDR documents.

## 4 Applying IPDR to other Legacy Protocols

The mechanisms defined by the IPDR 3.5 series of documents also provide a means to bridge the information models for existing non-XML protocols into the world of web services (and vice versa). Specifically this is done by using XML-Schema, with IPDR restrictions to describe these information models along with necessary annotations to define unambiguous mappings to the legacy encoding model of the target protocol. RADIUS and IPFIX are examples of two IETF protocols for which this mapping has been done.