

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: June 25, 2016

L. Zhou
CNNIC
D. Ma
W. Wang
ZDNS Ltd.
N. Kong
X. Lee
CNNIC
J. Galvin
Affilias
December 23, 2015

**Verification Extension for the Extensible Provisioning Protocol (EPP)
Contact Mapping
draft-zhou-eppext-contact-verification-01**

Abstract

This mapping describes an verification extension to EPP contact mapping [RFC5733]. Specified in Extensible Markup Language (XML), this extended mapping is applied to provide additional features required for the provisioning of contact verification.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on June 25, 2016.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents

(<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

Table of Contents

1.	Introduction	3
2.	Conventions Used in This Document	3
3.	Object Attributes	4
3.1.	Distinction Type Values	4
3.2.	Verification Status Values	4
3.3.	Dates and Times	4
3.4.	Client Identifier	5
4.	Verification State Diagram	5
5.	EPP Command Mapping	6
5.1.	EPP Query Commands	6
5.1.1.	EPP <check> Command	6
5.1.2.	EPP <info> Command	8
5.1.3.	EPP <transfer> Command	11
5.2.	EPP Transform Commands	11
5.2.1.	EPP <create> Command	12
5.2.2.	EPP <delete> Command	12
5.2.3.	EPP <renew> Command	12
5.2.4.	EPP <transfer> Command	12
5.2.5.	EPP <update> Command	12
6.	Formal Syntax	12
7.	Internationalization Considerations	14
8.	IANA Considerations	15
8.1.	XML Namespace	15
8.2.	EPP Extension Registry	15
9.	Security Considerations	15
10.	Acknowledgement	16

11. Normative References	16
Authors' Addresses	17

[1. Introduction](#)

The verification of domain name and registrant identity are required in some registries according to local laws and regulations. The registry should ensure the domain registered does not contain any illegal words and the registrants should pass the real-name verification. There are efforts on verification mechanism by introducing a third party that providing verification service [I-D.draft-gould-eppext-verificationcode]. This method is intended to offer a verification framework but not detail the verification statuses which are employ in practice to indicate the verification process. To be in alignment with the verification status indication mechanism, EPP should be extended accordingly.

This document describes an extension mapping for version 1.0 of the Extensible Provisioning Protocol (EPP) [RFC5730]. This mapping, an extension to EPP object mappings like the EPP contact mapping [RFC5733], can be used to retrieve verification information in query commands.

This document is specified using the XML 1.0 as described in [W3C.REC-xml-20040204] and XML Schema notation as described in [W3C.REC-xmlschema-1-20041028] and [W3C.REC-xmlschema-2-20041028].

[2. Conventions Used in This Document](#)

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

In examples, "C:" represents lines sent by a protocol client and "S:" represents lines returned by a protocol server. Indentation and white space in examples are provided only to illustrate element relationships and are not a REQUIRED feature of this specification.

XML is case sensitive. Unless stated otherwise, XML specifications and examples provided in this document MUST be interpreted in the character case presented to develop a conforming implementation.

vericontact-1.0 in this document is used as an abbreviation for urn:ietf:params:xml:ns:vericontact-1.0.

3. Object Attributes

This extension adds additional elements to the EPP contact mapping [[RFC5733](#)]. Only the new elements are described here.

3.1. Distinction Type Values

A contact may be verified already and may have something like integrity records. So a distinction type values are defined to associate with a contact object. Distinction type value descriptions:

- o verified. A contact has been verified already.
- o blocked. A contact has blemished integrity records.
- o unverified. A contact has not pass the verification process.

3.2. Verification Status Values

The contact object MUST always have one associated verification status value. The verification status value can be set only by the server. The verification status of an object MAY change as a result of an action performed by a server operator. Verification status Value descriptions:

- o unverified. No verification materials are received.
- o pendingVerify. Verification action has not been completed by the server after receiving verification materials. Server operators can delay action completion for a variety of reasons, such as to allow for human review or third-party action.
- o pass. Successful verification.
- o failed. Failed verification. Further verification materials may be needed.

3.3. Dates and Times

Date and time attribute values MUST be represented in Universal Coordinated Time (UTC) using the Gregorian calendar. The extended date-time form using upper case "T" and "Z" characters defined in [[W3C.REC-xmlschema-2-20041028](#)] MUST be used to represent date-time values, as XML Schema does not support truncated date-time forms or lower case "T" and "Z" characters.

3.4. Client Identifier

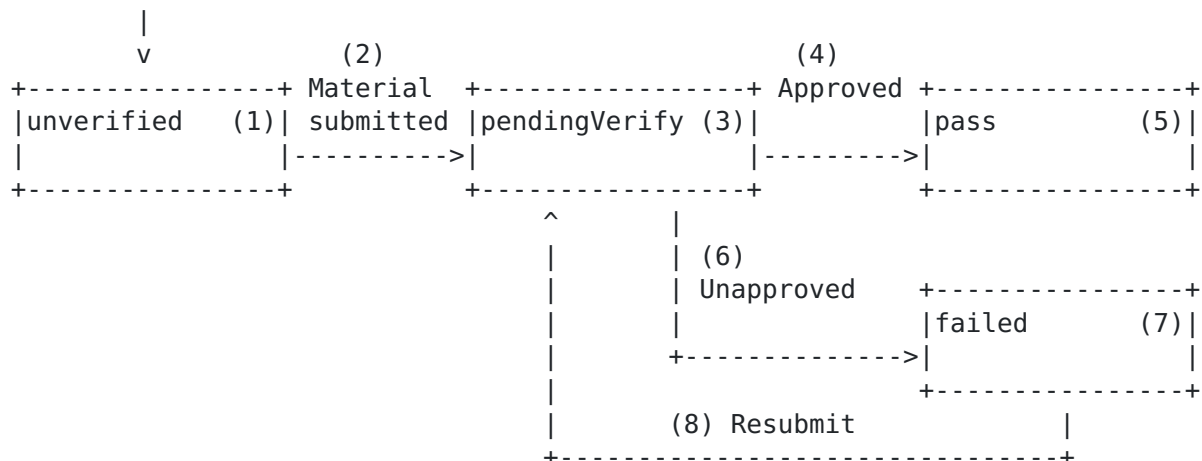
The client identifier represents the unique identifier assigned to the client by the server.

4. Verification State Diagram

Following is a general verification state transition process:

1. The initial verification status of a domain is "unverified".
2. The registrant submits the proof materials to the registry.
3. After receiving the proof materials, the verification status of the domain is changed to "pendingVerify".
4. The proof materials pass the human review or third-party verification.
5. The verification status is changed to "pass".
6. The proof materials are not approved.
7. The verification status is changed to "failed".
8. If the registrant resubmits the proof materials, the status will be set to "pendingVerify" again.

Figure 1: Verification State Diagram



5. EPP Command Mapping

A detailed description of the EPP syntax and semantics can be found in the EPP core protocol specification [[RFC5730](#)]. The command mappings described here are specifically for use in provisioning and managing verification information via EPP.

5.1. EPP Query Commands

EPP provides three commands to retrieve contact information: <check> to determine if a contact object can be provisioned within a repository, <info> to retrieve detailed information associated with a contact object, and <transfer> to retrieve contact-object transfer status information.

5.1.1. EPP <check> Command

This extension does not add any elements to the EPP <check> command described in the EPP contact mapping [[RFC5733](#)]. However, additional elements are defined for the <check> response.

Example <check> command:

```
C:<?xml version="1.0" encoding="UTF-8" standalone="no"?>
C:<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
C:  <command>
C:    <check>
C:      <contact:check
C:        xmlns:contact="urn:ietf:params:xml:ns:contact-1.0">
C:          <contact:id>sh8013</contact:id>
C:          <contact:id>sah8013</contact:id>
C:          <contact:id>8013sah</contact:id>
C:        </contact:check>
C:      </check>
C:    <clTRID>ABC-12345</clTRID>
C:  </command>
C:</epp>
```

When an <check> command has been processed successfully, the EPP <resData> element MUST contain child elements as described in the EPP contact mapping [[RFC5733](#)]. In addition, the EPP <extension> element SHOULD contain a child <vericontact:chkData> element that identifies the extension namespace if the contact object has data associated with this extension and based on its service policy. The <vericontact:chkData> element contains the following child elements:

- o An OPTIONAL <vericon:distinction> element is designed to indicate the verification status of a contact information with respect to the verification rules of a specific registry. The <vericon:distinction> element is only used for a <contact:id> element with the attribute "avail" that equals false. The element contains the following attributes:
 - * A "id" attribute associates with a specific contact identifier checked.
 - * A "type" attribute specifies whether a contact is verified or not as described in [section 3.1](#).

Example <check> response:

```

S:<?xml version="1.0" encoding="UTF-8" standalone="no"?>
S:<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
S:  <response>
S:    <result code="1000">
S:      <msg>Command completed successfully</msg>
S:    </result>
S:    <resData>
S:      <contact:chkData
S:        xmlns:contact="urn:ietf:params:xml:ns:contact-1.0">
S:        <contact:cd>
S:          <contact:id avail="0">sh8013</contact:id>
S:        </contact:cd>
S:        <contact:cd>
S:          <contact:id avail="0">sah8013</contact:id>
S:        </contact:cd>
S:        <contact:cd>
S:          <contact:id avail="0">8013sah</contact:id>
S:        </contact:cd>
S:      </contact:chkData>
S:    </resData>
S:    <extension>
S:      <vericontact:chkData
S:        xmlns:vericontact="urn:ietf:params:xml:ns:vericontact-1.0">
S:        <vericontact:distinction id="sh8013" type="verified"/>
S:        <vericontact:distinction id="sah8013" type="blocked"/>
S:        <vericontact:distinction id="8013sah" type="unverified"/>
S:      </vericontact:chkData>
S:    </extension>
S:    <trID>
S:      <clTRID>ABC-12345</clTRID>
S:      <svTRID>54322-XYZ</svTRID>
S:    </trID>
S:  </response>
S:</epp>

```

5.1.2. EPP <info> Command

This extension does not add any element to the EPP <info> command described in the EPP contact mapping [\[RFC5733\]](#). However, additional elements are defined for the <info> response.

Example <info> command:


```
C:<?xml version="1.0" encoding="UTF-8" standalone="no"?>
C:<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
C:  <command>
C:    <info>
C:      <contact:info
C:        xmlns:contact="urn:ietf:params:xml:ns:contact-1.0">
C:          <contact:id>sh8013</contact:id>
C:          <contact:authInfo>
C:            <contact:pw>2fooBAR</contact:pw>
C:          </contact:authInfo>
C:        </contact:info>
C:      </info>
C:    <clTRID>ABC-12345</clTRID>
C:  </command>
C:</epp>
```

When an <info> command has been processed successfully, the EPP <resData> element MUST contain child elements as described in the EPP contact mapping [[RFC5733](#)]. In addition, the EPP <extension> element SHOULD contain a child <vericontact:infData> element that identifies the extension namespace if the contact object has data associated with this extension and based on its service policy. The <vericontact:infData> element contains the following child elements:

- o A <vericontact:status> element that contains the current verification status defined in [section 3.2](#).
- o An OPTIONAL <vericontact:history> element that contains records with history verification process information. The <vericontact:history> element MUST contain following elements:
 - * <vericontact:record> element contains a single history record for the verification process. The <vericontact:record> element MUST contain following elements:
 - + A <vericontact:date> element contains the date and time when the operation has been executed.
 - + A <vericontact:op> element contains the name of an operation that has been executed.
 - + A <vericontact:clID> element contains the identifier of an sponsoring client.

Example <info> response for an authorized client:

Internet-Draft Verification Extension for the EPP Contact Module December 2015

```
S:<?xml version="1.0" encoding="UTF-8" standalone="no"?>
S:<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
S:  <response>
S:    <result code="1000">
S:      <msg>Command completed successfully</msg>
S:    </result>
S:    <resData>
S:      <contact:infData
S:        xmlns:contact="urn:ietf:params:xml:ns:contact-1.0">
S:        <contact:id>sh8013</contact:id>
S:        <contact:roid>SH8013-REP</contact:roid>
S:        <contact:status s="linked"/>
S:        <contact:status s="clientDeleteProhibited"/>
S:        <contact:postalInfo type="int">
S:          <contact:name>John Doe</contact:name>
S:          <contact:org>Example Inc.</contact:org>
S:          <contact:addr>
S:            <contact:street>123 Example Dr.</contact:street>
S:            <contact:street>Suite 100</contact:street>
S:            <contact:city>Dulles</contact:city>
S:            <contact:sp>VA</contact:sp>
S:            <contact:pc>20166-6503</contact:pc>
S:            <contact:cc>US</contact:cc>
S:          </contact:addr>
S:        </contact:postalInfo>
S:        <contact:voice x="1234">+1.7035555555</contact:voice>
S:        <contact:fax>+1.7035555556</contact:fax>
S:        <contact:email>jdoe@example.com</contact:email>
S:        <contact:clID>ClientY</contact:clID>
S:        <contact:crID>ClientX</contact:crID>
S:        <contact:crDate>2015-02-03T212:00:00.0Z</contact:crDate>
S:        <contact:upID>ClientX</contact:upID>
S:        <contact:upDate>2015-02-20T09:00:00.0Z</contact:upDate>
S:        <contact:trDate>2015-10-08T09:00:00.0Z</contact:trDate>
S:        <contact:authInfo>
S:          <contact:pw>2fooBAR</contact:pw>
S:        </contact:authInfo>
S:        <contact:disclose flag="0">
S:          <contact:voice/>
S:          <contact:email/>
S:        </contact:disclose>
S:      </contact:infData>
S:    </resData>
S:    <extension>
S:      <vericon:infData
xmlns:vericon="urn:ietf:params:xml:ns:vericon-1.0">
S:        <vericon:status>pass</vericon:status>
S:        <vericon:history>
S:          <vericon:record>
```



```
S:      <vericontact:date>2015-2-6T12:00:00.0Z</vericontact:date>
S:      <vericontact:op>PASS</vericontact:op>
S:      <vericontact:clID>ClientX</vericontact:clID>
S:      </vericontact:record>
S:      <vericontact:record>
S:      <vericontact:date>2001-2-3T15:00:00.0Z</vericontact:date>
S:      <vericontact:op>PENDINGVERIFY</vericontact:op>
S:      <vericontact:clID>ClientX</vericontact:clID>
S:      </vericontact:record>
S:      <vericontact:record>
S:      <vericontact:date>2015-2-3T12:00:00.0Z</vericontact:date>
S:      <vericontact:op>UNVERIFIED</vericontact:op>
S:      <vericontact:clID>ClientX</vericontact:clID>
S:      </vericontact:record>
S:      </vericontact:history>
S:      </vericontact:infData>
S:      </extension>
S:      <trID>
S:      <clTRID>ngcl-IvJjzMZc</clTRID>
S:      <svTRID>test142AWQONJZ</svTRID>
S:      </trID>
S:      </response>
S: </epp>
```

<info> response for the unauthorized client has not been changed, see [\[RFC5733\]](#) for detail.

An EPP error response MUST be returned if an <info> command cannot be processed for any reason.

5.1.3. EPP <transfer> Command

This extension does not add any elements to the EPP <transfer> command or <transfer> response described in the EPP contact mapping [\[RFC5733\]](#).

5.2. EPP Transform Commands

EPP provides five commands to transform domain objects: <create> to create an instance of a domain object, <delete> to delete an instance of a domain object, <renew> to extend the validity period of a contact object, <transfer> to manage domain object sponsorship changes, and <update> to change information associated with a contact object.

5.2.1. EPP <create> Command

This extension does not add any elements to the EPP <create> command or <create> response described in the EPP contact mapping [RFC5733].

5.2.2. EPP <delete> Command

This extension does not add any elements to the EPP <delete> command or <delete> response described in the EPP contact mapping [RFC5733].

5.2.3. EPP <renew> Command

This extension does not add any elements to the EPP <renew> command or <renew> response described in the EPP contact mapping [RFC5733].

5.2.4. EPP <transfer> Command

This extension does not add any elements to the EPP <transfer> command or <transfer> response described in the EPP contact mapping [RFC5733].

5.2.5. EPP <update> Command

This extension does not add any elements to the EPP <update> command or <update> response described in the EPP contact mapping [RFC5733].

6. Formal Syntax

An EPP object mapping is specified in XML Schema notation. The formal syntax presented here is a complete schema representation of the object mapping suitable for automated validation of EPP XML instances. The BEGIN and END tags are not part of the schema; they are used to note the beginning and ending of the schema for URI registration purposes.

BEGIN

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<schema targetNamespace="urn:ietf:params:xml:ns:vericontact-1.0"
  xmlns:vericontact="urn:ietf:params:xml:ns:vericontact-1.0"
  xmlns:epp="urn:ietf:params:xml:ns:epp-1.0"
  xmlns:eppcom="urn:ietf:params:xml:ns:eppcom-1.0"
  xmlns="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified">
```

```
<!-- Import common element types -->
```

```
<import namespace="urn:ietf:params:xml:ns:eppcom-1.0"
  schemaLocation="eppcom-1.0.xsd"/>
<import namespace="urn:ietf:params:xml:ns:epp-1.0"
  schemaLocation="epp-1.0.xsd"/>

<annotation>
  <documentation>
    Extensible Provisioning Protocol v1.0
    Contact Verification Extension Schema v1.0
  </documentation>
</annotation>

<!-- Child response element -->
<element name="chkData" type="vericontact:chkDataType"/>
<element name="infData" type="vericontact:infDataType"/>

<!-- <vericontact:chkData> response elements -->
<complexType name="chkDataType">
  <sequence>
    <element name="distinction" type="vericontact:distinctionType"
minOccurs="0"/>
  </sequence>
</complexType>

<complexType name="distinctionType">
  <simpleContent>
    <extension base="eppcom:labelType">
      <attribute name="id" type="eppcom:clIDType" use="required"/>
      <attribute name="type" type="vericontact:distinctionValueType"/>
    </extension>
  </simpleContent>
</complexType>

<simpleType name="distinctionValueType">
  <restriction base="token">
    <enumeration value="verified"/>
    <enumeration value="blocked"/>
    <enumeration value="unverified"/>
  </restriction>
</simpleType>

<!-- <vericontact:infData> response elements -->

<complexType name="infDataType">
  <sequence>
    <!-- current verification status -->
    <element name="status" type="vericontact:statusType"/>
    <!-- history records of verification process -->
    <element name="history" type="vericontact:historyType"/>
  </sequence>
</complexType>
```



```

    </sequence>
  </complexType>

  <simpleType name="statusType">
    <restriction base="token">
      <enumeration value="unverified"/>
      <enumeration value="pendingVerify"/>
      <enumeration value="pass"/>
      <enumeration value="failed"/>
    </restriction>
  </simpleType>

  <complexType name="historyType">
    <sequence>
      <element name="record" type="vericontact:recordType" minOccurs="0"
maxOccurs="unbounded"/>
    </sequence>
  </complexType>

  <complexType name="recordType">
    <sequence>
      <element name="date" type="dateTime"/>
      <element name="op" type="eppcom:minTokenType"/>
      <element name="clID" type="eppcom:clIDType"/>
    </sequence>
  </complexType>

<!-- End of schema. -->
</schema>
END

```

7. Internationalization Considerations

EPP is represented in XML, which provides native support for encoding information using the Unicode character set and its more compact representations including UTF-8. Conformant XML processors recognize both UTF-8 and UTF-16. Though XML includes provisions to identify and use other character encodings through use of an "encoding" attribute in an <?xml?> declaration, use of UTF-8 is RECOMMENDED.

As an extension of the EPP contact mapping, the elements, element content described in this document MUST inherit the internationalization conventions used to represent higher-layer domain and core protocol structures present in an XML instance that includes this extension.

8. IANA Considerations

8.1. XML Namespace

This document uses URNs to describe XML namespaces and XML schemas conforming to a registry mechanism described in [\[RFC3688\]](#). IANA is requested to assign the following URI.

Registration request for the contact verification namespace:

- o URI: urn:ietf:params:xml:ns:vericontact-1.0
- o Registrant Contact: See the "Author's Address" section of this document.
- o XML: See the "Formal Syntax" section of this document.

8.2. EPP Extension Registry

The EPP extension described in this document should be registered by the IANA in the EPP Extension Registry described in [\[RFC7451\]](#). The details of the registration are as follows:

Name of Extension: Contact Verification Extension

Document Status: Informational

Reference: (insert reference to RFC version of this document)

Registrant Name and Email Address: See the "Author's Address" section of this document.

TLDs: any

IPR Disclosure: none

Status: active

Notes: none

9. Security Considerations

The object mapping extension described in this document does not provide any other security services or introduce any additional considerations beyond those described by [\[RFC5730\]](#), [\[RFC5733\]](#) or those caused by the protocol layers used by EPP.

10. Acknowledgement

The authors would like to thank Galvin Brown from CentralNic for the idea behind use of verification state diagram, and Lin Dong from .top registry for his careful reviews.

11. Normative References

- [I-D.[draft-gould-eppext-verificationcode](#)]
Gould, J., "Verification Code Extension for the Extensible Provisioning Protocol (EPP)", November 2015,
<<https://tools.ietf.org/html/draft-gould-eppext-verificationcode-01>>.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#),
DOI 10.17487/RFC2119, March 1997,
<<http://www.rfc-editor.org/info/rfc2119>>.
- [RFC3688] Mealling, M., "The IETF XML Registry", [BCP 81](#), [RFC 3688](#),
DOI 10.17487/RFC3688, January 2004,
<<http://www.rfc-editor.org/info/rfc3688>>.
- [RFC5730] Hollenbeck, S., "Extensible Provisioning Protocol (EPP)",
STD 69, [RFC 5730](#), DOI 10.17487/RFC5730, August 2009,
<<http://www.rfc-editor.org/info/rfc5730>>.
- [RFC5733] Hollenbeck, S., "Extensible Provisioning Protocol (EPP)
Contact Mapping", STD 69, [RFC 5733](#), DOI 10.17487/RFC5733,
August 2009, <<http://www.rfc-editor.org/info/rfc5733>>.
- [RFC7451] Hollenbeck, S., "Extension Registry for the Extensible
Provisioning Protocol", [RFC 7451](#), DOI 10.17487/RFC7451,
February 2015, <<http://www.rfc-editor.org/info/rfc7451>>.
- [W3C.REC-xml-20040204]
Bray, T., Paoli, J., Sperberg-McQueen, C., Maler, E., and
F. Yergeau, "'Extensible Markup Language (XML) 1.0 (Third
Edition)", World Wide Web Consortium First Edition REC-xml-
20040204", February 2004,
<<http://www.w3.org/TR/2004/REC-xml-20040204>>.
- [W3C.REC-xmlschema-1-20041028]
Thompson, H., Beech, D., Maloney, M., and N. Mendelsohn,
"'XML Schema Part 1: Structures Second Edition", World
Wide Web Consortium Recommendation REC-xmlschema-
1-20041028", October 2004,
<<http://www.w3.org/TR/2004/REC-xmlschema-1-20041028>>.

[W3C.REC-xmlschema-2-20041028]

Biron, P. and A. Malhotra, "XML Schema Part 2: Datatypes
Second Edition", World Wide Web Consortium Recommendation
REC-xmlschema-2-20041028", October 2004,
<<http://www.w3.org/TR/2004/REC-xmlschema-2-20041028>>.

Authors' Addresses

Linlin Zhou
CNNIC
4 South 4th Street, Zhongguancun, Haidian District
Beijing, Beijing 100190
China

Phone: +86 10 5881 2677
Email: zhoulinlin@cnnic.cn

Di Ma
ZDNS Ltd.
4 South 4th Street, Zhongguancun, Haidian District
Beijing, Beijing 100190
China

Email: madi@zdns.cn

Wei Wang
ZDNS Ltd.
4 South 4th Street, Zhongguancun, Haidian District
Beijing, Beijing 100190
China

Email: wangwei@zdns.cn

Ning Kong
CNNIC
4 South 4th Street, Zhongguancun, Haidian District
Beijing, Beijing 100190
China

Phone: +86 10 5881 3147
Email: nkong@cnnic.cn

Xiaodong Lee

CNNIC

4 South 4th Street, Zhongguancun, Haidian District

Beijing, Beijing 100190

China

Phone: +86 10 5881 3020

Email: xl@cnnic.cn

James Galvin

Afilias

Afilias USA, Inc. Building 3, Suite 105 300 Welsh Road

Horsham, PA 19044

US

Email: jgalvin@afilias.info