



# Liberty ID-SIS Contact Book Service Implementation Guidelines

Version: 1.0

## **Editors:**

Guillaume Lambert, France Télécom

## **Contributors:**

Shin Adachi, NTT

Rajeev Angal, Sun

Salima Fazal Karim, France Télécom

Sean Franklin, American Express

Ariel Gordon, France Télécom

Jukka Kainulainen, Nokia

Sampo Kellomäki, France Télécom

Kurt Kolok, IEEE-ISTO

Rob Lockhart, IEEE-ISTO

## **Abstract:**

This document provides implementation guidelines supplemental to the Liberty ID-SIS Contact Book (ID-SIS-CB) service specification. It is also the general guideline for the Liberty ID-SIS Contact Book. The reader is expected to be familiar with the Liberty ID-WSF Web Services Framework Overview, XML, SAML, SOAP, vCard, and the Liberty ID-SIS Personal and Employee Profiles. The Liberty ID-SIS Contact Book is a web service hosted by an application provider and usually discovered via a discovery service. It offers the ability to manage a contact directory. The contact format is based upon the vCard Specifications and may contain several types of information such as telephone numbers and postal and email addresses. An extension mechanism allows other arbitrary data to be included. An ID-SIS-CB service also stores information regarding the Principal him or herself but is not intended to replace the Liberty ID-SIS Personal or Employee Profiles. An ID-SIS-CB service is an instance of a data-oriented identity web service. An ID-SIS-CB service, like all data services, is characterized by an ability to query and update attribute data. It incorporates mechanisms from other specifications for access control and for conveying data validation information and usage directives.

**Filename:** liberty-id-sis-contactbook-guidelines-v1.0.pdf

1 **Notice**

2 This document has been prepared by Sponsors of the Liberty Alliance. Permission is hereby granted to use the  
3 document solely for the purpose of implementing the Specification. No rights are granted to prepare derivative works  
4 of this Specification. Entities seeking permission to reproduce portions of this document for other uses must contact  
5 the Liberty Alliance to determine whether an appropriate license for such use is available.

6 Implementation of certain elements of this document may require licenses under third party intellectual property  
7 rights, including without limitation, patent rights. The Sponsors of and any other contributors to the Specification are  
8 not and shall not be held responsible in any manner for identifying or failing to identify any or all such third party  
9 intellectual property rights. **This Specification is provided "AS IS", and no participant in the Liberty Alliance  
10 makes any warranty of any kind, express or implied, including any implied warranties of merchantability,  
11 non-infringement of third party intellectual property rights, and fitness for a particular purpose.** Implementers  
12 of this Specification are advised to review the Liberty Alliance Project's website (<http://www.projectliberty.org>) for  
13 information concerning any Necessary Claims Disclosure Notices that have been received by the Liberty Alliance  
14 Management Board.

15 Copyright © 2005-2006 America Online, Inc.; American Express Travel Related Services; Ericsson; France  
16 Télécom; Nippon Telegraph and Telephone Corporation; Nokia Corporation; Sun Microsystems, Inc.; Symlabs, Inc.;  
17 and Vodafone Group Plc. All rights reserved.

18  
19 Liberty Alliance Project  
20 Licensing Administrator  
21 c/o IEEE-ISTO  
22 445 Hoes Lane  
23 Piscataway, NJ 08855-1331, USA  
24 info@projectliberty.org  
25

---

## Contents

26		
27	1. Introduction	4
28	1.1. Document Audience	4
29	1.2. Architectural Context of the ID-SIS-CB	4
30	1.2.1. ID-SIS-CB as an Interface	4
31	1.2.2. Participants and Compliance Testing	4
32	1.3. XML Document Instantiations	5
33	1.4. Extension Mechanisms	5
34	2. Overview of the Conceptual Data Model (a.k.a. "Generic vCard")	6
35	2.1. Relationship between ID-SIS-CB and vCard	6
36	2.2. Structure of the Conceptual Data Model	6
37	3. Discovery and Queries	7
38	3.1. Rationale	7
39	3.2. Supported XPATH Expressions	7
40	4. Managing the CB	8
41	4.1. ID-SIS-CB Overview	8
42	4.2. ID-SIS-CB Interrogation	8
43	4.2.1. ID-SIS-CB Interrogation Examples	8
44	5. Mapping the CDM	10
45	5.1. Mapping with vCard 2.1, vCard 3.0, vCard Jabber, and vCard RDF	10
46	5.2. Mapping with Liberty ID-SIS Personal Profile	10
47	6. Cultural Portability	11
48	References	12

## 1. Introduction

The Liberty ID-SIS Contact Book Service Specification [[LibertyCB](#)] defines a Liberty identity service that supports information regarding the Principal and his or her contacts. The information regarding the Principal is stored in a unique card called "My Card" or "Self Card." This card is not intended to substitute for the Liberty ID-SIS Personal Profile Service [[LibertyIDPP](#)]. The Liberty ID-SIS Contact Book (ID-SIS-CB) service also allows the Principal to manage contacts for private and business acquaintances, friends, family members, and even for him or her self. The Principal can also create a Distribution List in order to simplify the management of his or her contacts.

This document provides a rationale and guidance for implementers of the ID-SIS-CB. A companion document, Liberty ID-SIS Contact Book Service Specification, [[LibertyCB](#)], normatively describes the ID-SIS-CB. If there is a disagreement between the present document and [[LibertyCB](#)], the [[LibertyCB](#)] specification is prescriptive.

### 1.1. Document Audience

This document is intended for application developers and implementers. The reader is presumed to be familiar with XML, SAML, SOAP, vCard, and the Liberty ID-SIS Personal Profile, [[LibertyIDPP](#)] and [[LibertyIDPPGuide](#)]. The reader should be familiar with the Liberty ID-FF Architecture Overview ([LibertyIDFFOverview](#)) and the Liberty ID-WSF Web Services Framework Overview ([LibertyIDWSFOverview11](#)).

### 1.2. Architectural Context of the ID-SIS-CB

The ID-SIS-CB service uses the general framework and the methods described in the Liberty ID-WSF Data Services Template Specification [[LibertyDST20](#)]. The services that consult the ID-SIS-CB service use the Liberty architectural framework to prove that they are acting on behalf of the Principal or that the Principal has somehow consented to sharing the data, e.g., by means of a standing order or subscription. The identity services are further described in [[LibertyIDWSFOverview11](#)].

The ID-SIS-CB service is based on the vCard standards [[RFC2426](#)] and [[vCard21](#)]. These standards are not [[XML](#)]-based contact formats. An abstract conceptual data model in XML has been created to specify the selection structure in query and modification operations. Mappings with multiple vCard formats and [[LibertyIDPP](#)] have been provided in the ID-SIS-CB specification.

#### 1.2.1. ID-SIS-CB as an Interface

The data accessible through the ID-SIS-CB often comes from back-end systems that may serve other purposes as well. For example, an enterprise hosting an ID-SIS-CB service for its employees may choose to use their customer address database. Such sharing of back-end systems is considered normal practice and may cause one service to update data in another "out-of-band." Out-of-band updates are expressly allowed, but are considered out of scope for the purposes of the ID-SIS-CB specification.

The ID-SIS-CB service specification, at the formal and conceptual levels, specifies an [[XML](#)] document. However, this does not mean that data is necessarily stored as an XML document. The data could just as well be computed on the fly or fetched from a directory [[RFC2251](#)] or relational database (SQL) server and formatted into XML only for the purpose of speaking Liberty protocols. When this document specifies behavior against a conceptual XML document, the implementation has to behave as if the document existed, but does not necessarily have to implement it in concrete terms.

The ID-SIS-CB also allows the data to be returned in other formats than the conceptual data model. The ID-SIS-CB service specification explains how to handle the vCard 2.1 [[vCard21](#)], vCard 3.0 [[RFC2426](#)], vCard Jabber [[JEP0054](#)], and vCard RDF [[vCardRDF](#)] formats. Other formats may be supported for particular purposes but the implementer has to provide the discovery options keywords and the mappings for those formats with the XML conceptual data model. Since other such supported formats imply a proprietary implementation of the ID-SIS-CB service, there is no reason that other web service clients (WSC) must understand the discovery options keywords associated to those formats.

## 93 1.2.2. Participants and Compliance Testing

94 The ID-SIS-CB is provided by an attribute provider (AP) [[LibertyIDWSFGuide10](#)], sometimes referred to as an ID-  
95 SIS-CB provider. The AP is a Liberty ID-WSF web service that hosts the ID-SIS-CB. The ID-SIS-CB is queried  
96 or updated by a client which is usually a service provider (SP) [[LibertyIDWSFOverview11](#)] acting on behalf of the  
97 Principal [[LibertyIDWSFGuide10](#)]. The client is sometimes referred to as a WSC. The [[LibertyIDWSFGuide10](#)]  
98 describes the means by which the Principal may delegate to the SP a right to invoke his or her ID-SIS-CB service, i.e.,  
99 a service assertion. Before the SP can access the ID-SIS-CB, it usually, but not necessarily, has to discover which AP  
100 hosts the ID-SIS-CB for the Principal. This is accomplished using a discovery service [[LibertyDisco12](#)] that issues  
101 the service assertions.

102 ID-SIS-CB compliance testing addresses both implementations and instances. ID-SIS-CB specifies an interface to  
103 which an implementation and an instance (deployment) of an ID-SIS-CB service conform. The implementation may  
104 be a software product offered by a vendor. Typically, such a product, if configured and operated correctly, will  
105 provide an ID-SIS-CB service instance. For an AP instance to be ID-SIS-CB-compliant, it must correctly use an  
106 ID-SIS-CB-compliant implementation.

## 107 1.3. XML Document Instantiations

108 An ID-SIS-CB service may respond to a query with an XML instantiation of the conceptual data model schema. The  
109 XML documents that are specified by the [[LibertyCB](#)] XML schema are the most general serial representations of the  
110 information. The expression "most general" means that a document could fully instantiate that schema if all data  
111 has been provisioned and no permissions filtering occurred. After filtering, the transmitted content may no longer  
112 conform to this schema. Thus, implementers may need to adjust this schema before using it to implement services.

113 When queries that point to interior elements of the conceptual XML document are applied, the returned data contains  
114 the queried element, its contents, and the higher level containers to avoid possible confusions in the case where several  
115 elements are queried. If the query asks for one or more whole contacts to be returned, the format in which the contacts  
116 are returned inside the SOAP answer may be specified as long as the ID-SIS-CB supports the requested format. The  
117 supported formats are specified in the discovery option keywords.

118 A potential confusion is that, as requests to an ID-SIS-CB service are actually SOAP documents, there is one schema  
119 for the SOAP layer and another for the document that is returned inside the SOAP body. The [[LibertyCB](#)] Service  
120 Specification does not define the SOAP schemas.

## 121 1.4. Extension Mechanisms

122 The ID-SIS-CB schema has an extension container element which permits arbitrary schema extension. If an  
123 implementation supports schema extension, it must register the appropriate discovery option keyword.

## 124 **2. Overview of the Conceptual Data Model (a.k.a. "Generic** 125 **vCard")**

### 126 **2.1. Relationship between ID-SIS-CB and vCard**

127 The Liberty ID-SIS-CB is based on vCard as specified in [vCard21] and [RFC2426]. All vCard types and their  
128 semantics, but not necessarily format, are hereby incorporated. The conceptual data model or "generic vCard" stands  
129 for an XML representation of vCard attributes and types. This means in particular that vCard categories are supported.  
130 (However, their use is not encouraged because of the several possible interpretations of this attribute. The ID-SIS-CB,  
131 instead, allows distribution lists to group together individual contact cards.)

132 The XML representation is used to specify [XPath] query language and modification operations. Mappings to other  
133 vCard formats are provided in this specification. The purpose is to enable multiple vCard formats to coexist without  
134 endorsing any specific format. It is expected that new future vCard formats may be accommodated this way.

### 135 **2.2. Structure of the Conceptual Data Model**

136 The conceptual data model supports all attributes and types specified by [vCard21] and [RFC2426] except "BEGIN,"  
137 "END," "PROFILE," "NAME," and "SOURCE" which are meaningless in an XML representation.

138 The jabber extension "JABBERID" and "DESC" are also supported. Additionally, the types "FAVORITES,"  
139 "LISTMEMBER," "SELF," and "CARDID" have been added for ID-SIS-CB-specific purposes.

140 The type "SELF" is used to indicate the "Self Card" of the ID-SIS-CB owner, "FAVORITE" indicates the membership  
141 of a favorite list, and "LISTMEMBER" gives the membership of a distribution list.

142 "CARDID" is the identifier of the vCard.

## 143 3. Discovery and Queries

### 144 3.1. Rationale

145 The ID-SIS-CB is intended to satisfy the most possible needs for a contact directory in a Liberty environment.  
146 However, an implementer may choose to deploy only a few functionalities of the ID-SIS-CB. The ID-SIS-CB that a  
147 service provider sees is apt to be incomplete because:

- 148 • not all information about the contacts needs to be provisioned,
- 149 • not all of the vCard format needs to be implemented,
- 150 • an AP's policy forbids the SP from having some information, and/or
- 151 • the permissions that the Principal sets forbid the SP from accessing parts of the information.

152 The capacity of an ID-SIS-CB service implementation is expressed by the discovery option keywords. Option  
153 keywords are used to discover the existence or support for particular containers or groups of containers in a way that  
154 is meaningful to applications. See [[LibertyDisco12](#)] for the generic definition of the Discovery Service and the  
155 processing rules for discovering by keyword.

### 156 3.2. Supported XPATH Expressions

157 Effectively, the minimal set of supported "Modify" [[XPATH](#)] expressions (i.e., "slashed paths") defines the minimal  
158 granularity of updates that needs to be supported. If a client needs to update an attribute at a finer granularity than  
159 defined here, then it should first query the element and then execute a "Modify" with queried values and the value it  
160 wants to change. It is recognized that this approach has inherent problems:

- 161 1. Other update(s) may occur between the "Query" and "Modify." The client should deal with this race condition  
162 in an implementation-dependent way, e.g., make a second query to verify that the update succeeded, ignore the  
163 possibility of a race condition.
- 164 2. "Query" may return incomplete data due to permissions. Presumably, under these circumstances, the  
165 corresponding "Modify" will fail for similar reasons. Updates to the containers listed above should be atomic  
166 whenever feasible. For example, if the underlying database technology is [[RFC2251](#)], it is advisable to model  
167 each of the above-listed containers as an entry so that the directory server provides atomicity of the update.

## 4. Managing the CB

### 4.1. ID-SIS-CB Overview

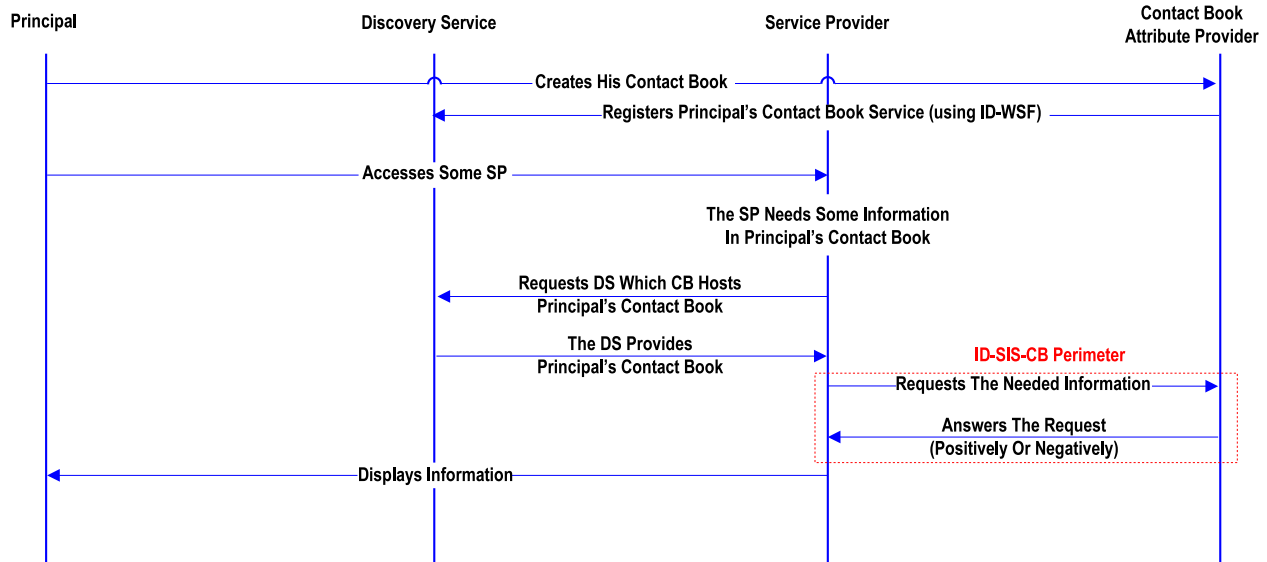


Figure 1. ID-SIS-CB Overview

### 4.2. ID-SIS-CB Interrogation

ID-SIS-CB is an AP as defined in [LibertyIDWSFGuide10]. It can be invoked using the classical [LibertyDST20] mechanisms.

The [XPATH] expression has to use the conceptual data model. As defined in the technical specification, when a whole card is requested, the returned format may be specified. When only a few vCard attributes are requested, the answer is mandatorily in the "generic vCard" format.

#### 4.2.1. ID-SIS-CB Interrogation Examples

Examples of the SOAP bodies of an ID-SIS-CB "query" and "response":

In the following request, we want to obtain all vCards (/cdm:vCard) where the contact's nickname is "Jane" ([/cdm:vCard/cdm:NICKNAME="Jane"]) in the vCard 3.0 format (<Select format="urn:liberty:cb:format:RFC2426">).

##### 4.2.1.1. ID-SIS-CB Interrogation Query Example

```

<Query
  xmlns="urn:liberty:id-sis-cb:2005-05"
  id="my ID">
  <ResourceID>My Resource ID</ResourceID>
  <QueryItem changedSince="2004-03-03T16:49:09Z"
    id="my ID"
    includeCommonAttributes="false"
    itemID="my Item ID">
    <Select format="urn:liberty:cb:format:RFC2426">
      /cdm:vCard[/cdm:vCard/cdm:NICKNAME="Jane"]
    </Select>
  </QueryItem>
</Query>
    
```

```
196     </QueryItem>
197 </Query>
```

#### 198 **4.2.1.2. ID-SIS-CB Interrogation Query Response Example**

199 The vCard is returned in raw vCard 3.0 format in the field <cb:CharData>

```
200 <QueryResponse
201     xmlns="urn:liberty:id-sis-cb:2005-05"
202     id="my ID">
203     <Status xmlns: ... >
204     </Status>
205     <Data itemIDRef="" >
206         <cb:Card cb:format="urn:liberty:cb:format:RFC2426">
207             <cb:CharData>
208 BEGIN:vCard
209 FN:Jane Doe
210 N:Doe;Jane
211 NICKNAME:Jane
212 TEL;WORK:123456789
213 EMAIL;TYPE=internet:jane@example.com
214 END:vCard
215         </cb:CharData>
216         </cb:Card>
217     </Data>
218 </QueryResponse>
```

## 219 **5. Mapping the CDM**

### 220 **5.1. Mapping with vCard 2.1, vCard 3.0, vCard Jabber, and vCard RDF**

221 The mappings provided by the [\[LibertyCB\]](#) specification are used to make the link between the conceptual data model  
222 and the different vCard formats used in the industry. Those mappings are almost intuitive but play an important role  
223 in the implementation. Even if the response format requested is not the "generic vCard," the ID-SIS-CB is queried  
224 by an [\[XPATH\]](#) expression upon the conceptual data model. It is important to have a standard and precise mapping  
225 to request of the ID-SIS-CB in order to ensure that different ID-SIS-CB implementations do not interpret the same  
226 [\[XPATH\]](#) query differently.

### 227 **5.2. Mapping with Liberty ID-SIS Personal Profile**

228 This mapping is provided to update the Principal's "Self Card" from his Liberty ID-SIS Personal Profile [\[LibertyIDPP\]](#).  
229 Although the [\[LibertyIDPP\]](#) Personal Profile and the [\[LibertyCB\]](#) conceptual data model are very different, it can be  
230 very useful for a Principal to update his "Self Card" from his Personal Profile. The reverse update is, apparently, not  
231 true. This mapping does not aim to be a perfect mapping. It identifies what containers in the Personal Profile provide  
232 the same information that vCard attributes provide.

## 6. Cultural Portability

234 An Internet environment is the underlying assumption for these system designs. End users may venture to web sites  
235 outside their own culture and interact with other users and businesses in foreign countries. This calls for a common  
236 language. A large part of the world, but not the entire world, has standardized on the use of the Latin alphabet  
237 (character set) with some variations.

238 As noted, the Liberty ID-SIS-CB conceptual data model relies on the vCard semantics.

239 The default character set of vCard is ASCII. It can be overridden for an individual property value by using the  
240 "CHARSET" property parameter. This property parameter may be used on any property. However, the use of this  
241 parameter on some properties may not make sense. Any character set registered with the Internet Assigned Numbers  
242 Authority (IANA - <http://www.iana.org/>) may be specified by this property parameter.

243 The default language is "en-US" (US English). It may be overridden for an individual property value by using the  
244 "LANGUAGE" property parameter. The values for this property are strings consistent with [RFC1766]'s "Tags for  
245 the Identification of Languages." This property parameter may be used on any property. However, the use of this  
246 parameter on some properties, e.g., "PHOTO," "LOGO," "SOUND," "TEL," may not make sense.

247 The Liberty Framework provides a different character set property (UTF-8) at the transport wrapper level. During  
248 transport, the vCard data stream is encoded in UTF-8. Since UTF-8 can represent all the characters of the other sets  
249 available in vCard, all languages supported by vCard may be supported.

# References

## Informative

- 252 [JEP0054] Saint-Andre, Peter, eds. (26 March, 2003). "Jabber Enhancement Proposal 0054: vcard-temp," v1.1, Jabber  
253 Software Foundation <http://www.jabber.org/jeps/jep-0054.html>
- 254 [LibertyCB] Kellomäki, Sampo, eds. "Liberty ID-SIS Contact Book Service Specification," Version 1.0, Liberty  
255 Alliance Project (11 November, 2006). <http://www.projectliberty.org/specs>
- 256 [LibertyDisco12] Sergent, Jonathan, eds. "Liberty ID-WSF Discovery Service Specification," Version 1.2, Liberty  
257 Alliance Project (12 December 2004). <http://www.projectliberty.org/specs/>
- 258 [LibertyDST20] Kainulainen, Jukka, Ranganathan, Aravindan, eds. "Liberty ID-WSF Data Services Template  
259 Specification," Version 2.0, Liberty Alliance Project (23 March, 2005). <http://www.projectliberty.org/specs>
- 260 [LibertyIDFFOverview] Wason, Thomas, eds. "Liberty ID-FF Architecture Overview," Version 1.2-errata-v1.0,  
261 Liberty Alliance Project (12 September 2004). <http://www.projectliberty.org/specs>
- 262 [LibertyIDPP] Kellomäki, Sampo, Lockhart, Rob, eds. "Liberty ID-SIS Personal Profile Service Specification,"  
263 Version 1.1, Liberty Alliance Project (29 September, 2005). <http://www.projectliberty.org/specs>
- 264 [LibertyIDPPGuide] Kellomäki, Sampo, Lockhart, Rob, eds. "Liberty ID-SIS Personal Profile Service Implementation  
265 Guidelines," Version 1.1, Liberty Alliance Project (29 September, 2005). <http://www.projectliberty.org/specs>
- 266 [LibertyIDWSFGuide10] Weitzel, David, eds. (22 May 2005). "Liberty ID-WSF Implementation Guideline," Draft  
267 v1.0-12, Liberty Alliance Project <http://www.projectliberty.org/specs/>
- 268 [LibertyIDWSFOverview11] Tourzan, Jonathan, Koga, Yuzo, eds. "Liberty ID-WSF Web Services Framework  
269 Overview," Version 1.1, Liberty Alliance Project (14 December 2004). <http://www.projectliberty.org/specs>
- 270 [RFC1766] Alvestrand, H., eds. (March 1995). "Tags for the Identification of Languages," RFC 1766, Internet  
271 Engineering Task Force <http://www.ietf.org/rfc/rfc1766.txt>
- 272 [RFC2251] "Lightweight Directory Access Protocol (v3)," M. Wahl T. Howes S. Kille (December 1997). RFC 2251,  
273 Internet Engineering Task Force <http://www.ietf.org/rfc/rfc2251.txt>
- 274 [RFC2426] Dawson, F., Howes, T., eds. (September 1998). "vCard MIME Directory Profile," RFC 2426, The Internet  
275 Engineering Task Force <http://www.ietf.org/rfc/rfc2426.txt>
- 276 [vCard21] Consortium, Versit, eds. (18 September, 1996). "vCard, The Electronic Business Card," Version 2.1,  
277 Internet Mail Consortium <http://www.imc.org/pdi/vcard-21.txt>
- 278 [vCardRDF] Iannella, Renato, eds. (22 February, 2001). "Representing vCard Objects in RDF/XML," W3C NOTE-  
279 vcard-rdf-20010222, World Wide Web Consortium <http://www.w3.org/TR/vcard-rdf>
- 280 [XML] Bray, Tim, Paoli, Jean, Sperberg-McQueen, C. M., Maler, Eve, Yergeau, Francois, eds. (04 February 2004).  
281 "Extensible Markup Language (XML) 1.0 (Third Edition)," Recommendation, World Wide Web Consortium  
282 <http://www.w3.org/TR/2004/REC-xml-20040204>
- 283 [XPath] Clark, J., DeRose, S., eds. (16 November 1999). "XML Path Language (XPath) Version 1.0,"  
284 Recommendation, W3C <http://www.w3.org/TR/xpath> [August 2003].