Open GIS Consortium Inc.

Date: 2004-05-04

Reference number of this OpenGIS® project document: **OGC 04-011r1**

Version: **0.9.1**

Category: OpenGIS® Discussion Paper

Editor: Peter Schut

Geolinking Service (GLS)

Copyright notice

Warning

This document is not an OGC Standard. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an OGC Standard.

Recipients of this document are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

:

Document type: OpenGIS® Discussion Paper
Document subtype: Discussion Paper
Document stage: Publicly Available Draft
Document language: English

Contents

i. Preface

This document is the result of work undertaken to support the Canadian Geospatial Data Infrastructure (CGDI), and in particular the National Land and Water Information Service (NLWIS), and the National Forest Information Service (NFIS). The standard was first implemented as a prototype in 2002 by Agriculture and Agri-Food Canada (AAFC) with the help of DM Solutions Group.

ii. Submitting organizations

The following organizations submitted this Implementation Specification to the Open GIS Consortium Inc.:

a) GeoConnections / Natural Resources Canada

iii. Submission contact points

All questions regarding this submission should be directed to the editor or the submitters:

CONTACT	COMPANY	ADDRESS	PHONE/FAX	EMAIL
Peter Schut	Agriculture Canada	960 Carling Ave. Ottawa, ON K0A 1A0 Canada	+01-613-759-1874	schutp@agr.gc.ca
Xiaoyuan Geng	Agriculture Canada	960 Carling Ave. Ottawa, ON K0A 1A0 Canada	+01-613-759-1895	gengx@agr.gc.ca
Tom Kralidis	Environment Canada	867 Lakeshore Road, Burlington, ON L7R 4A6	+01-905-336-4409	tom.kralidis@ec.gc.ca
Rick Morrison	Natural Resources Canada	506 West Burnside Road, Victoria, BC Canada V8Z 1M5	+01-250-363-0772	rmorriso@pfc.cfs.nrcan.gc.ca
Brian Low	Natural Resources Canada	506 West Burnside Road, Victoria, BC Canada V8Z 1M5	+01-250-363-0600	Brian.Low@nrcan-rncan.gc.ca

Daniel Morissette	DM Solutions Group	30 Rosemount Ave, Suite 200 Ottawa, ON Canada K1Y 1P4	+01-613-565-5056	dmorissette@dmsolutions.ca

iv. Revision history

Date	Release	Author	Paragraph modified	Description
28 Feb 2004	0.9.0	P. Schut	All	Initial document, formatted for OGC template
16 Mar 2004	0.9.1	P. Schut	All	Complete edits. Added intro and schemas
04 May 2004	0.9.1	P. Schut	Various	Diagrams added to intro section, use case added.

v. Changes to the OpenGIS® Abstract Specification

The OpenGIS© Abstract Specification does not require changes to accommodate this OpenGIS© standard.

Foreword

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. Open GIS Consortium Inc. shall not be held responsible for identifying any or all such patent rights. However, to date, no such rights have been claimed or identified.

This is the first version of this standard.

This standard is related to the Geolinked Data Access Service (GDAS), which is currently a discussion document at OGC.

Introduction

This document is the specification for a Geolinking Service (GLS).

A Geolinking Service operates on geolinked data. Geolinked data is geographically-related data in which the geometry is not directly stored with the attribute data, but instead a geographic identifier is used. The geographic identifier refers to a geometric feature in a separate geospatial data set. An example of geolinked data is a table of populations of Canadian cities. The table does not contain the geographic location of each city, but instead uses the city name. The city name can be used to join the population data to a separate geospatial data set that contains the geographic location of each city, in order to map the information or perform some sort of geospatial analysis.

This specification is intended to provide a mechanism to take geolinked data delivered over a network, and either incorporate it into a database, or process it in some fashion that supports visualization or modelling. A geolinking service is an Internet-enabled "relate" (a.k.a. link/join/merge) function on two tables that share a common key field.

The development of services based on this specification presupposes the existence of corporate databases that contain geographically referenced information, but where the corporate environment does not contain a direct link to a GIS. The Geolinkage Field (the names of each of the cities in the above example) must exist in a compatible format in both the geolinked data set and the geospatial data set (e.g., the city name in both data sets must be identical), to provide the linkage between the two data sets. The corporate systems deliver their database contents to the Geolinking Service using a Geolinked Data Access Service (GDAS).

Figure 1 shows how a Geolinking Service can be used in conjunction with a WMS to create a Geolinking WMS

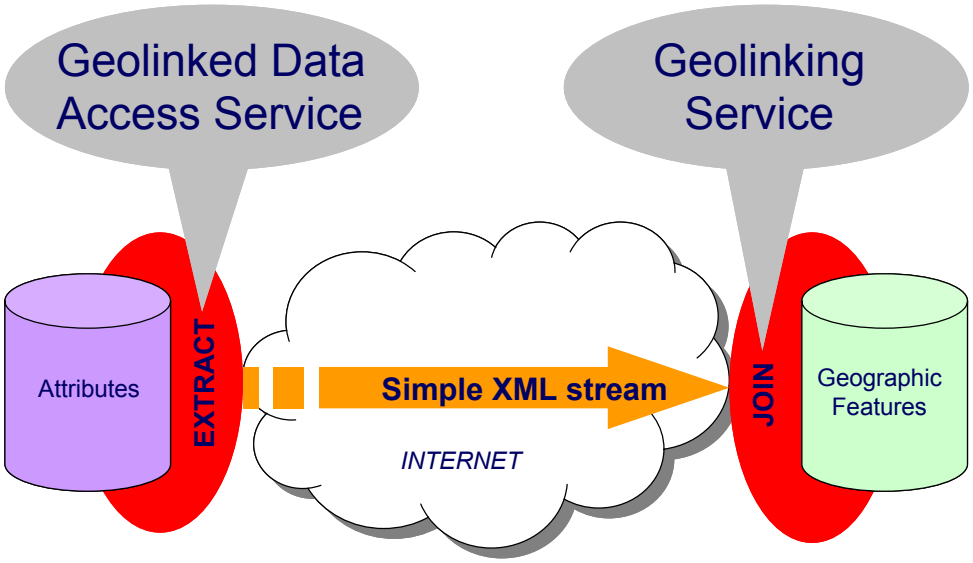

Figure 1. A Geolinking service interacting with a Geolinked Data Access Service

0.1 The requirement for Geolinking

Almost all corporate databases contain some kind of geographic identifier, regardless of whether or not the database is housed in a computing environment that supports GIS. Geographic identifiers can include postal codes, municipality names, telephone area codes, or more special purpose identifiers such as school districts. Geolinking technology allows this corporate data to be found and used for mapping or spatial analysis.

The geographic identifiers used in corporate databases usually reference a spatial framework. A spatial framework in this context is a partitioning of the surface of the earth into a set of management units. Municipalities, postal codes, telephone area codes, ecoregions, and watersheds are all examples of spatial frameworks. These frameworks all have one thing in common – they contain a descriptor that can be used to uniquely identify any individual management unit –and that same identifier is found in the corporate database.

Here are some typical database contents along with their spatial frameworks:

- Sales by retail outlet or by municipality
- Insurance payments by postal code
- Telephone numbers by area code
- Farms by Census Agricultural Region
- Students by school district.

When this corporate data is exposed to other systems as a GDAS XML stream, a Geolinking service can merge that data with the spatial data that describes the framework, in order to enable mapping or geospatial analysis.

0.2 Advantages of Geolinking

The advantages of Geolinking technology are that it allows organizations to house their corporate data on systems that are optimized for the management of that data, and yet to take advantage of GIS technology to examine and analyze that data.

Geolinking allows corporate data to be maintained closest to source, and yet allow the latest data to be obtained when analysis is being performed, regardless of whether or not the geospatial system can make a direct connection to the corporate data management system. In effect, Geolinking technology supports both distributed data management, as well as the distributed processing of geospatial data.

Geolinking is based on two related technologies. Attribute data is provided to other computers on the network by implementing a Geolinked Data Access Service (GDAS). At some other node on the network, a Geolinking Service (GLS) allows a computer to use data from a GDAS stream by incorporating it into a local dataset. This local dataset may support Web Mapping, desktop GIS, or even some analytical capability that may in turn produce another GDAS stream.

0.3 Applications of Geolinking

0.3.1 Geolinking and Web Mapping

The most widely applicable use of Geolinking may be to support of Web mapping. In this case, the Geolinking Service is placed on top of a Web Mapping Service (WMS), to create a Geolinking WMS. Attributes provided through a GDAS stream are merged into a local dataset by the Geolinking Service, which then allows a WMS layer to be produced. The resulting dataset can be stored for future queries (and exposed as a standard layer by the WMS), or discarded after processing if data volatility is of concern.

Figure 2 shows how a Geolinking Service can be used in conjunction with a WMS to create a Geolinking WMS

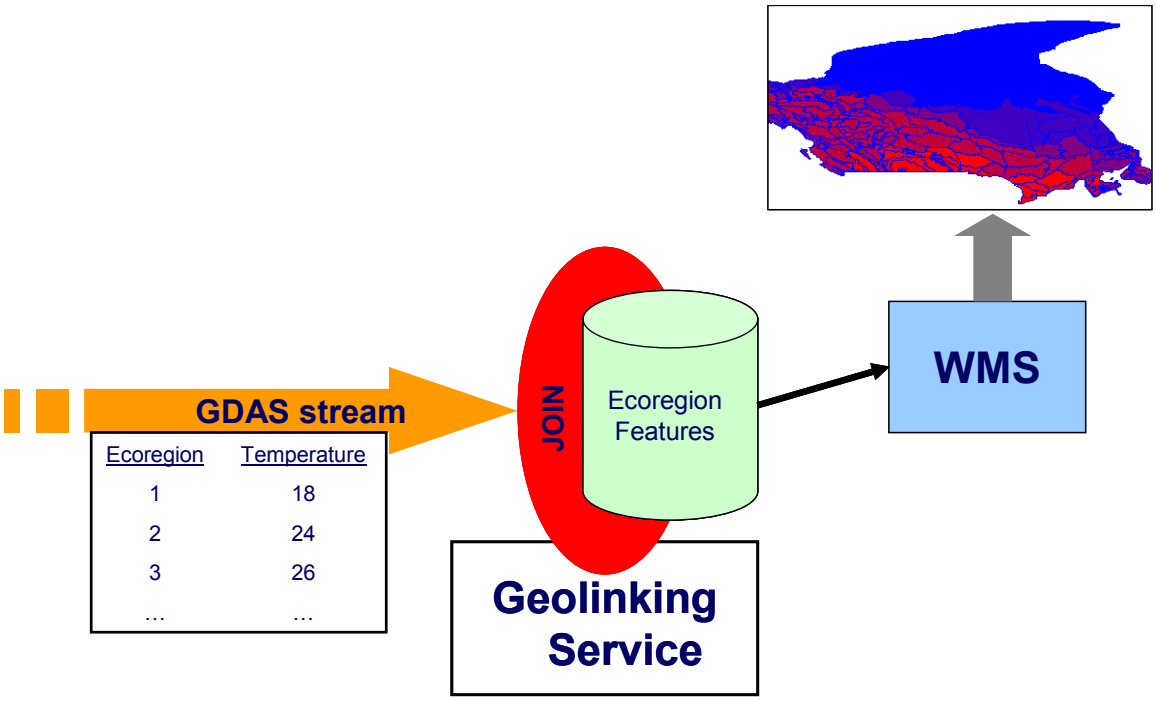

Figure 2. A Geolinking WMS

Figure 3 shows the components necessary to process transaction requests from client applications.

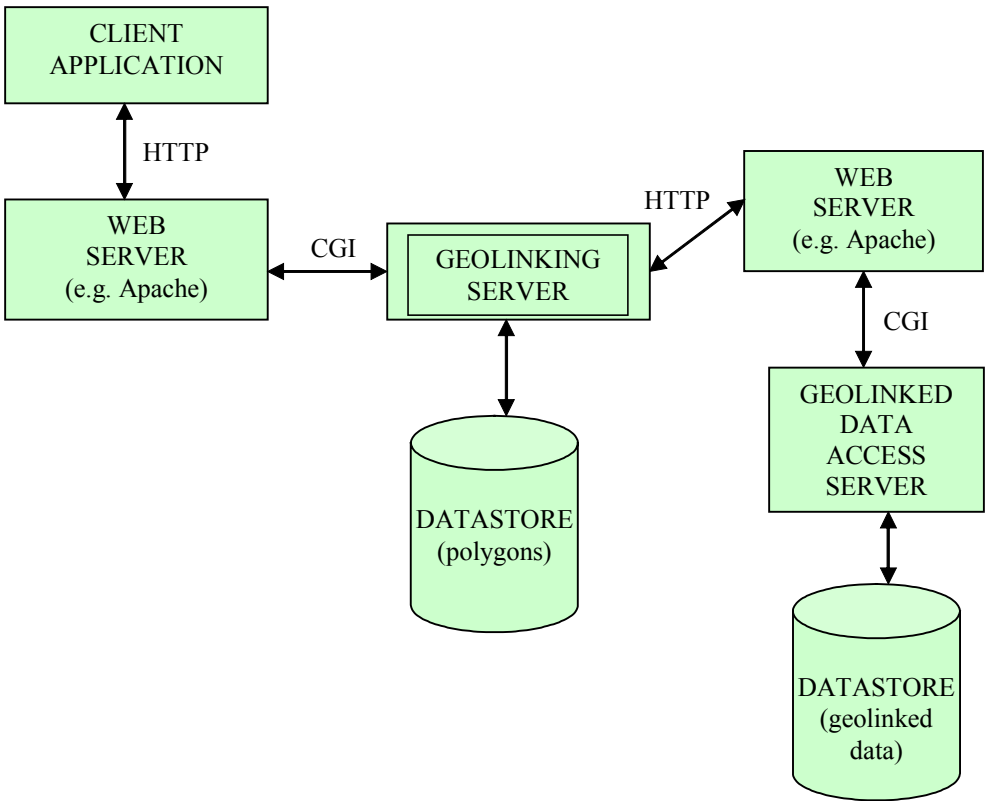

Figure 3. Geolinking Service Architecture

A description of each component in Figure 3 is:

CLIENT APPLICATION
Any program or process that communicates with a web server using HTTP. In most cases the client application would chain the output from this service to a Web Map Server.

WEB SERVER
Any program that services HTTP requests. For example, the Apache program is a Web server.

GEOLINKING SERVER
A program or module that implements interfaces that support merging of attribute data and associated polygonal geometry.

GEOLINKED DATA ACCESS SERVER
A program or module that implements interfaces to support delivery of geolinked data.

DATASTORE
A software component for persistently storing and managing the spatial and/or non-spatial properties of geographic features. The Datastore can be a SQL relational database, flat files, a GIS database, a static XML file, etc...

A use case for a Geolinking WMS is shown in Appendix E.

0.3.2 Geolinking and Desktop GIS

Similar to the case of Web Mapping, a spatial data warehouse that supports desktop GIS may be have its attribute data supplemented or amended by data provided via a GDAS stream. In this case, the Geolinking Service adds a field to a spatial dataset in the warehouse, or amends the contents of an existing attribute field.

0.3.3 Geolinking and Modelling

When geospatial data is exposed to Internet through a GDAS, it is easy to feed that data into models (e.g. climate change models). If a model is enabled with a Geolinking service on the front end to accept data, and a GDAS on the output end to provide results, it is easy to run different input scenarios through the model. This ultimately simplifies data management, and reduces the potential for errors in input data.

0.3.4 Geolinking and Spatial Data Warehousing

Geolinking also allows for the ability to do replicate databases across the Internet. A Geolinking Service can use GDAS streams to regularly update the contents of a data warehouse and its associated metadata tables, based on the latest information available from the primary data warehouse.

OpenGIS© Interface — Geolinking Service

1 Scope

This specification applies to the creation and use of a Geolinking Service (GLS) which processes geolinked datasets by merging the contents with related data tables, so that they might be used by particular clients for the purpose of mapping or additional data processing.

This specification is relevant to data provided in a format specified by the OGC Geolinked Data Access Service (GDAS) [currently an OGC discussion document].

This specification does not address the archival, cataloguing, discovery or retrieval of information that has been processed by a GLS.

2 Conformance

Conformance with this specification shall be checked using all the relevant tests specified in Annexes A and C (normative).

3 Normative references

[1] CGDI architecture pages at http://www.geoconnections.org/architecture/
.

[2] Geolinking Service Version 0.8.2:
http://www.geoconnections.org/architecture/archdocs.cfm?file=technical/specifications/geolinking/

[4] URL to Geolinker Discussion Paper

[5] XML 1.0 (October 2000), *eXtensible Markup Language (XML) 1.0* (*2nd edition)*, World Wide Web Consortium Recommendation, Bray, T., Paoli, J., Sperberg-McQueen, C.M., and Maler, E., eds., <http://www.w3.org/TR/2000/REC-xml>

IETF RFC 2045 (November 1996), *Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies*, Freed, N. and Borenstein N., eds., <http://www.ietf.org/rfc/rfc2045.txt>

IETF RFC 2119 (March 1997), *Key words for use in RFCs to Indicate Requirement Levels*, Bradner, S., ed., <http://www.ietf.org/rfc/rfc2119.txt>.

4 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

4.1
framework dataset
a geospatial dataset that partitions the surface of the earth into a set of management units. Municipalities, postal codes, telephone area codes, ecoregions, and watersheds are all examples of spatial frameworks

4.2
attribute dataset
a set of values that describe some aspect of a framework dataset. Population, temperature, and income are all examples of attribute data that could reference a framework dataset.

4.3
geolinking
the merging (joining) of an attribute dataset to a framework dataset

4.4
geolinked data
an attribute dataset that contains a field that references a spatial identifier field in a framework dataset.

4.5
operation
specification of a transformation or query that an object may be called to execute [OGC AS 12]

4.6
interface
named set of **operations** that characterize the behavior of an entity [OGC AS 12]

4.7
service
distinct part of the functionality that is provided by an entity through **interfaces** [OGC AS 12]

4.8
server
actual implementation of a **service**

4.9
client
software component that can invoke an **operation** from a **server**

4.10
request
invocation of an **operation** by a **client**

4.11
response
result of an **operation** returned from a **server** to a **client**

4.12
map
pictorial representation of geographic data.

4.13
Capabilities
service-level metadata describing the **operations** and content available at a **service instance**.

5 Conventions

5.1 Symbols (and abbreviated terms)

CGDI Canadian Geospatial Data Infrastructure
CGI Common Gateway Interface
EPSG European Petroleum Survey Group
GIF Graphics Interchange Format
GIS Geographic Information System
HTTP Hypertext Transfer Protocol
IETF Internet Engineering Task Force
JPEG Joint Photographic Experts Group
MIME Multipurpose Internet Mail Extensions
OGC Open GIS Consortium
OWS OGC Web Service
PNG Portable Network Graphics
RFC Request for Comments
SLD Styled Layer Descriptor
URL Uniform Resource Locator
WCS Web Coverage Service
WFS Web Feature Service
WMS Web Map Service
XML Extensible Markup Language

6 Basic service elements

6.1 Introduction

This clause specifies aspects of Geolinking Server behavior (more generally, of OGC Web Service behavior) that are independent of particular operations or are common to several operations or interfaces.

6.2 Version numbering and negotiation

6.2.1 Version number form

The published specification version number contains three positive integers, separated by decimal points, in the form "x.y.z". The numbers "y" and "z" will never exceed 99. Each OWS specification is numbered independently.

6.2.2 Version changes

A particular specification's version number **shall** be changed with each revision. The number **shall** increase monotonically and **shall** comprise no more than three integers separated by decimal points, with the first integer being the most significant. There may be gaps in the numerical sequence. Some numbers may denote experimental or interim versions. Service instances and their clients need not support all defined versions, but **must** obey the negotiation rules below.

6.2.3 Appearance in Requests and in Service Metadata

The version number appears in at least two places: in the Capabilities XML describing a service, and in the parameter list of client requests to that service. The version number used in a client's request of a particular service instance **must** be equal to a version number which that instance has declared it supports (except during negotiation as described below). A service instance may support several versions, whose values clients may discover according to the negotiation rules.

6.2.4 Version number negotiation

A Client may negotiate with a Service Instance to determine a mutually agreeable specification version. Negotiation is performed using the **GetCapabilities** operation [see Clause 7] according to the following rules.

All Capabilities XML must include a protocol version number. In response to a **GetCapabilities** request containing a version number, an OGC Web Service **must** either

respond with output that conforms to that version of the specification, **or** negotiate a mutually agreeable version if the requested version is not implemented on the server. If no version number is specified in the request, the server **must** respond with the highest version it understands and label the response accordingly.

Version number negotiation occurs as follows:

a) If the server implements the requested version number, the server **must** send that version.

b) If a version unknown to the server is requested, the server **must** send the highest version it knows that is less than the requested version.

c) If the client request is for a version lower than any of those known to the server, then the server **must** send the lowest version it knows.

d) If the client does not understand the new version number sent by the server, it **may** either cease communicating with the server **or** send a new request with a new version number that the client does understand but which is less than that sent by the server (if the server had responded with a lower version).

e) If the server had responded with a higher version (because the request was for a version lower than any known to the server), and the client does not understand the proposed higher version, then the client **may** send a new request with a version number higher than that sent by the server.

The process is repeated until a mutually understood version is reached, or until the client determines that it will not or cannot communicate with that particular server.

EXAMPLE 1 Server understands versions 1, 2, 4, 5 and 8. Client understands versions 1, 3, 4, 6, and 7. Client requests version 7. Server responds with version 5. Client requests version 4. Server responds with version 4, which the client understands, and the negotiation ends successfully.

EXAMPLE 2 Server understands versions 4, 5 and 8. Client understands version 3. Client requests version 3. Server responds with version 4. Client does not understand that version or any higher version, so negotiation fails and client ceases communication with that server.

The version parameter is mandatory in requests other than **GetCapabilities**.

6.3 General HTTP request rules

6.3.1 Overview

At present, the only distributed computing platform (DCP) explicitly supported by OGC Web Services is the World Wide Web itself, or more specifically Internet hosts implementing the Hypertext Transfer Protocol (HTTP). Thus the Online Resource of

each operation supported by a service instance is an HTTP Uniform Resource Locator (URL). The URL may be different for each operation, or the same, at the discretion of the service provider. Each URL **must** conform to the description in [HTTP] but is otherwise implementation-dependent; only the parameters comprising the service request itself are mandated by the OGC Web Services specifications.

HTTP supports two request methods: GET and POST. One or both of these methods may be defined for a particular OGC Web Service type and offered by a service instance, and the use of the Online Resource URL differs in each case.

An Online Resource URL intended for HTTP GET requests is in fact only a URL prefix to which additional parameters must be appended in order to construct a valid Operation request. A URL prefix is defined as an opaque string including the protocol, hostname, optional port number, path, a question mark '?', and, **optionally**, one or more server specific parameters ending in an ampersand '&'. The prefix uniquely identifies the particular service instance. For HTTP GET, the URL prefix **must** end in either a '?' (in the absence of additional server-specific parameters) or a '&'. In practice, however, Clients **should** be prepared to add a necessary trailing '?' or '&' before appending the Operation parameters defined in this specification in order to construct a valid request URL.

An Online Resource URL intended for HTTP POST requests is a complete and valid URL to which Clients transmit encoded requests in the body of the POST document. A GLS server **must not** require additional parameters to be appended to the URL in order to construct a valid target for the Operation request.

6.3.2 Key-value pair encoding (GET or POST)

6.3.2.1 Overview

Using Key-Value Pair encoding, a client composes the necessary request parameters as keyword/value pairs in the form "keyword=value", separated by ampersands (‘&’), with appropriate encoding [IETF RFC 2396] to protect special characters. The resulting query string may be transmitted to the server via HTTP GET or HTTP POST, as prescribed in the HTTP Common Gateway Interface (CGI) standard [IETF RFC 2616].

Table 1 summarizes the request parameters for HTTP GET and POST.

Table 1 –Parts of a Key-Value Pair OGC Web Service Request

URL Component	Description
http://host[:port]/path	URL of service operation. The URL is entirely at the discretion of the service provider.
{name[=value]&}	The query string, consisting of one or more standard request parameter name/value pairs defined by an OGC Web Service. The actual list of required and optional parameters is mandated for each operation by the appropriate OWS specification.

Notes: [ ] denotes 0 or 1 occurrence of an optional part; {} denotes 0 or more occurrences.

A request encoded using the HTTP GET method interposes a '?' character between the service operation URL and the query string, to form a valid URI which may be saved as a bookmark, embedded as a hyperlink, or referenced via Xlink in an XML document.

6.3.2.2 Parameter ordering and case

Parameter names **shall not** be case sensitive, but parameter values shall be case sensitive. In this document, parameter names are typically shown in uppercase for typographical clarity, not as a requirement.

Parameters in a request **may** be specified in any order.

An OGC Web Service **must** be prepared to encounter parameters that are not part of this specification. In terms of producing results per this specification, an OGC Web Service **shall** ignore such parameters.

6.3.2.3 Parameter lists

Parameters consisting of lists shall use the comma (",") as the delimiter between items in the list: e.g., `parameter=item1,item2,item3.` Multiple lists can be specified as the value of a parameter by enclosing each list in parentheses ("(", ")"): e.g., `parameter=(item1a,item1b,item1c),(item2a,item2b,item2c)`. If a parameter name or value includes a space or comma, it shall be escaped using the URL encoding rules [IETF RFC 2396].

6.3.3 XML encoding

Clients may also encode requests in XML for transmission to the server using HTTP GET or (more often) HTTP POST. The XML request must conform to the schema corresponding to the chosen operation, and the client must send it to the URL listed for that operation in the server's Capabilities XML file, in accordance with HTTP POST [IETF RFC 2616]).

NOTE To support SOAP messaging, clients need only enclose this XML document in a SOAP envelope as follows:

```
<env:Envelope
 xmlns:env="http://www.w3.org/2001/09/soap-envelope">
   <env:Body>
     request document here
   </env:Body>
</env:Envelope>
```

6.4 General HTTP response rules

Upon receiving a valid request, the service **must** send a response corresponding exactly to the request as detailed in the appropriate specification. Only in the case of Version Negotiation (described above) may the server offer a differing result.

Upon receiving an invalid request, the service **must** issue a Service Exception as described in Subclause 6.5 below.

NOTE As a practical matter, in the WWW environment a client should be prepared to receive either a valid result, or nothing, or any other result. This is because the client may itself have formed a nonconforming request that inadvertently triggered a reply by something other than an OGC Web Service, because the Service itself may be non-conforming, etc.

6.5 Service exceptions

Upon receiving an invalid request, the service **must** issue a Service Exception XML message to describe to the client application or its human user the reason(s) that the request is invalid.

Service Exception XML **must** be valid according to the Service Exception XML Schema in Subclause A.7. In an HTTP environment, the MIME type of the returned XML **must** be "`text/xml`". Specific error messages can be included either as chunks of plain text or as XML-like text containing angle brackets ("<" and ">") if included in a character data (CDATA) section as shown in the example of Service Exception XML in Subclause A.7.

Service Exceptions **may** include exception codes as indicated in Subclause A.7. Servers **shall not** use these codes for meanings other than those specified. Clients **may** use these codes to automate responses to Service Exceptions.

7 Geolinking Service Operations

A Geolinking Service (GLS) must have the knowledge about the data tables it services (in other words, the service is a data dependent service).

The GLS specification includes two mandatory operations:
GetCapabilities
GeoLink

For each operation three request methods are defined:
HTTP GET using URL-encoded Key-Value Pairs (KVP). All servers must support this method (the regular URL format we're used to)

HTTP POST using URL-encoded Key-Value Pairs (KVP). This method is optional.

HTTP POST with XML-encoded requests. This method is optional.

7.1 GetCapabilities (required)

7.1.1 General

The purpose of the GetCapabilities operation is described in the Basic Service Elements section, above. In the particular case of a GLS, the response of a GetCapabilities request is general information about the service itself and specific information about the tables upon which it can operate.

7.1.2 GetCapabilities Overview

The general form of a GetCapabilities request is defined in the Basic Service Elements section. When making this request of a GLS, which may offer other OGC Web Services as well, it is necessary to indicate that the client seeks information about the GLS in particular. Thus, the SERVICE parameter of the request **shall** have the value "GLS" as shown in Table 1 below.

TABLE 2: The parameters of a GetCapabilities request URL

Request Parameter	Required/Optional	Description
Version=0.9.1	Optional	Request Version. Defaults to the latest available version, currently 0.9.1
Service=GLS	Required	Service Type, must be GLS
Request=GetCapabilities	Required	Request Name

7.1.3 GetCapabilities Request Parameters

7.1.3.1 Version

The **optional** VERSION parameter, and its use in version negotiation, is specified in the Basic Service Elements section.

7.1.3.2 Service

The **required** SERVICE parameter indicates which of the available service types at a particular service instance is being invoked. This parameter allows the same URL prefix to offer Capabilities XML for multiple OGC Web Services. When invoking GetCapabilities on a GLS that implements this version of the specification or a later one, the service_name value "GLS" **shall** be used.

7.1.3.3 Request

This nature of the **required** REQUEST parameter is specified in the Basic Service Elements section. To invoke the GetCapabilities operation, the value "GetCapabilities" **shall** be used.

Here is an example of a **KeyValuePair GetCapabilities** request using the HTTP GET method:

```
http://foo.bar/foo?SERVICE=GLS&REQUEST=GetCapabilities&VERSION=0.9.1
```

7.1.3.4 XML Encoding

XML encoded **GetCapabilities** requests use the HTTP POST method with the body of the request (POST data) of type text/xml. For example, a client may encode a **GetCapabilities** request in XML as follows:

```
<GetCapabilities version="0.9.1" service="GLS"/>
```

7.1.4 GetCapabilities Response

The Basic Service Elements section specifies general rules about the GetCapabilities response.

In the particular case of a GLS complying with this version of the standard, the Extensible Markup Language (XML)[XML 1.0] response **shall** be valid according to the XML schema in Annex A.1 of this document. The schema specifies the required and optional content of the response and how the content is formatted.

A server's Capabilities XML **may** reference an exact copy of the schema in Annex A.1 instead of the master copy at the URL stated in the Annex. The schema copy **shall** be located at a fully-qualified and accessible URL to permit XML validating software to retrieve it.

A server **may** comply with other published or experimental versions, in which case it **shall** support Version Negotiation as described in the Basic Service Elements section. A schema for version 0.9.1 was published as an annex to that version of the GLS specification. Other schemas are archived at <http://www.geoconnections.org/>.

The response to a **GetCapabilities** operation is an XML document that contains the capabilities of an implementation of this service, the operations it supports, and the data it serves. An example of the results from a **GetCapabilities** request is found in Annex B

The **GetCapabilities** document includes the generic Service and Capability sections common to all OGC Web Services. These sections are not described in detail here, apart from two fields specific to a GLS service: GeolinkidsLimit and AttributeLimit.

The optional <GeolinkidsLimit> element in the service metadata is a positive integer indicating the maximum number of **GeolinkIds** a client is permitted to include in a single GeoLink Request. If the value of this element is "0", this feature is not supported. If this element is absent, the server imposes no limit.

The optional <AttributeLimit> element in the service metadata is a positive integer indicating the maximum number of **Attribute**s a client is permitted to include in a single GeoLink Request. If this element is absent, the server imposes no limit.

The Capability section contains a **Geolinkables** element describing the framework datasets available for Geolinking on this server.

The **Geolinkables** element and its sub-elements are described in more details below.

7.1.4.1 Geolinkables

The Geolinkables element contains the set of Datastores that can be serviced by the GLS. A GLS can provide geolinking services for more than one **Datastore**.

7.1.4.1.1 Datastore

A Datastore is a database or service that can be accessed by the GLS, and which contains the Framework data.

7.1.4.1.1.1 Host

The URL of the server that hosts the framework data

7.1.4.1.1.2 Supports

Indicates what the **Datastore** supports. Valid Supports keywords are shown in Table 2 below:

TABLE 2: Valid parameters for the Supports element

Supports Keyword	Description
WMS	Web Map Service, used when geolinking is used to modify a datastore that supports WMS
WFS	Web Feature Service, used when when geolinking is used to modify GML delivered from a WFS
WAREHOUSE	Data Warehouse (normally used when the geolinking target is a non-spatial data table instead of a framework dataset)
GIS	Used to denote a that the service modifies a datastore destined for access by conventional desktop GIS
OTHER	Other

7.1.4.1.1.3 Framework

The Framework element contains metadata describing the spatial Framework dataset to which a set of attribute data can be merged. A GLS provides the ability to merge data to one or more Frameworks.

7.1.4.1.1.3.1 Organization

The name of the organization that is responsible for maintaining the framework dataset.

7.1.4.1.1.3.2 Domain

The Internet domain name of the organization responsible for maintaining the framework dataset.

7.1.4.1.1.3.3 Name

The name of the framework dataset. Generally the name by which the dataset might be identified within a GIS environment.

7.1.4.1.1.3.4 Title

A human readable sentence fragment that might form a title if the framework dataset were displayed in map form.

7.1.4.1.1.3.5 Description

A complete description or abstract that describes the framework dataset

7.1.4.1.1.3.6 Version

The version identifier of the framework dataset. Version "1" can be used where no other versioning number is available.

7.1.4.1.1.3.7 Release date

The date that the framework dataset was first released for use. Alternatively, instead of a date, the content may be a version number of the framework dataset. In either case, this uniquely identifies the version of the dataset.

7.1.4.1.1.3.8 Documentation

Location (generally a URI) of complete documentation describing the framework dataset.

7.1.4.1.1.3.9 Geolinkage field

The key field within the **Framework** dataset (i.e. in the GIS) that is referenced by all **Dataset**s that apply to this **Framework**.

Note: The **Dataset**'s **GeolinkageField** value is a field name in the corporate database, for the **GeolinkageField** used by this **Dataset** and the **Framework** it belongs to. The

Framework's and the **Dataset**'s **GeolinkageField** values don't have to be identical; they can be different names for the same field.

7.2 Geolink operation (required)

This operation is used to direct the GLS to access a specified geolinked dataset, and join that information to the framework dataset. The mechanism used to perform that join is not specified, and the join may be a virtual or temporary join.

7.2.1 Key-value Pair Encoding

The following keywords are defined for **KeyValuePair GeoLink** requests:

TABLE 2: Required and Optional parameters for the GeoLink request

Request Parameter	Required/Optional	Description
Service=GLS	Required	Service Type
Request=**GeoLink**	Required	Request Name
Version=0.9.1	Required	Request Version
FrameworkHost	Optional	See below
GDAS	Required	See below
GDASVersion	Required	See below
FrameworkDomain	Required	See below
FrameworkName	Required	See below
FrameworkVersion	Required	See below
DatasetDomain	Required	See below
DatasetName	Required	See below
Attribute	Required	See below
Geolinkids	Optional	See below
SLD	Optional	See below
Cache	Optional	See below

A sample **GeoLink** request is shown in Annex B.4.

7.2.2 XML Encoding

The body of an XML-encoded **GeoLink** request is shown in Annex B.5.

7.2.3 GeoLink Request Parameters

Note that the parameters **Framework**, **Framework_Version**, **Dataset**, **Attribute**, and **GeolinkIds** will be used by the GLS to generate a GDAS request, thus the specification for these parameters mirrors that of the GDAS specification.

7.2.3.1 Service

Specifies the type of service being requested. For GLS services, the only valid value is "GLS". This is to allow other OGC services to be delivered from the same URI;

7.2.3.2 Request

Request specifies the type of request being made. For a **GetData** request, the only valid value is "**GetData**"

7.2.3.3 Version

The version of the GLS service requested. For this version of the specification the value is "0.9.1".

7.2.3.4 FrameworkHost

The URL of the server that hosts the **Framework** data to which the **Attribute** should be joined. This is to support Framework data being delivered via GML.

7.2.3.5 GDAS

The URL of the GDAS server that serves the **Attribute** data.

7.2.3.6 GDASVersion

The version of GDAS to be requested from the GDAS server.

7.2.3.7 FrameworkDomain

The domain that uniquely identifies the agency responsible for the framework - e.g. ec.gc.ca

7.2.3.8 FrameworkName

The Framework (geospatial) dataset to which the **Attribute** should be joined

7.2.3.9 FrameworkVersion

The version identifier of the **Framework** dataset.

7.2.3.10 DatasetDomain

The domain that uniquely identifies the agency responsible for the dataset - e.g. agr.gc.ca

7.2.3.11 DatasetName

The dataset on the GDAS server from which the **Attribute** should be drawn;

7.2.3.12 Attribute

One or more specific variable(s) being requested from the GDAS server (e.g. mean monthly temperature for July).

Note that all attributes in a **GetData** request must be from a single geolinked dataset.
In Key-Value Pair mode, the value of **Attribute** is a comma-delimited list of 1 or more attribute names, e.g.

ATTRIBUTE=cows,heifers,totcattle

In an XML-encoded request, there is one <Attribute>...</Attribute> element for each attribute in the request.

7.2.3.13 GeolinkIds

The optional GEOLINKIDS parameter can be used to limit the number of records that will be returned by the server. If this parameter is omitted then all records from the dataset are returned.

In **KeyValuePair** mode, the value is a comma-delimited list of values of the **GeolinkageField** for the selected dataset, e.g.

GEOLINKIDS=BC,AL,SA,MA

Note that when id values are part of a Key-value Pair Encoding request, then the encoding will have to form a valid URL.
In an XML-encoded request, the <**GeolinkIds**> element contains one <I>...</I> element for each value.

7.2.3.14 SLD

The URL of the map styling information to be applied if the GLS services a **Datastore** of type "WMS".

7.2.3.15 Cache

Indicates the number of days that a cache should be maintained. Intended primarily for use where the GLS services a **Datastore** of type "WMS".

7.2.4 GeoLink Response

The response to a **GeoLink** query is defined by the XML schema found in Annex A.2. A sample response to a **GeoLink** query is found in Annex B.6.

7.2.4.1 Response Structure

The body of the response is a GLS_Data element which includes

1) most of the metadata elements found in the GetCapabilities response, specifically: **Name**, **Title**, **Abstract**, **OnlineResource**, and **ContactInformation**.
2) all of the metadata found in the GetCapabilities which describe the Framework dataset to which the geolinked data was joined;
3) the **GDAS** element provided in the GeoLink request;
4) all of the metadata describing the **Dataset** that was delivered as a result of the GDAS request initiated by the GLS, including the metadata described in the **Attribute** element
5) the **SLD** element provided in the GeoLink request
6) the **Cache** element provided in the GeoLink request
7) a **GeoLinkage** element that is described below.

7.2.4.2 GeoLinkage

Geolinkage contains status information about the success or failure of the linkage process. It includes one or more **DataStore** elements.

7.2.4.3 DataStore

Contains metadata about the geolink operation and its status.

7.2.4.4 Host

The URL of the server that hosts the **Framework** data to which the **Attribute** was joined.

7.2.4.5 Supports

Indicates the type of **Datastore** which to which the **Attribute** was joined. Valid keywords are indicated under this element for the **GetCapabilities** operation.

7.2.4.6 Attribute

The specific variables which were requested from the GDAS server and joined to the Framework data - as indicated in the GeoLink request parameters.

7.2.4.7 Status

The status of the GeoLink operation. Valid keywords are "Succeeded" and "Failed".

7.2.4.8 Url

The URL at which the resultant product can be accessed. For example, if the GLS **Supports** WMS, the URL required to access the resultant layer is displayed.

Annex A
(normative)

GDAS XML Schemas

A.1 GetCapabilities Response Schema

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
    <!-- **  GeoLinking Service:  GetCapabilities Schema version 0.9.1 - Top-Level Element.  ** -->
    <xs:element name="GLS_Capabilities">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Service"/>
                <xs:element ref="Capability"/>
            </xs:sequence>
            <xs:attribute name="version" type="xs:string" use="required"/>
        </xs:complexType>
    </xs:element>
    <!-- ** Elements Used In Multiple Places. ** -->
    <xs:element name="Name" type="xs:string"/>
    <xs:element name="Title" type="xs:string"/>
    <xs:element name="Format" type="xs:string"/>
    <xs:element name="DCPType">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="HTTP"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="HTTP">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Get"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="Get">
        <xs:complexType>
            <xs:attribute name="onlineResource" type="xs:anyURI" use="required"/>
        </xs:complexType>
    </xs:element>
    <!-- ** GLS_Capabilities : Service -  General Service Metadata. ** -->
    <xs:element name="Service">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Name"/>
                <xs:element ref="Title"/>
                <xs:element ref="Abstract"/>
                <xs:element ref="OnlineResource"/>
                <xs:element ref="KeywordList"/>
                <xs:element ref="ContactInformation"/>
                <xs:element ref="GeolinkidsLimit"/>
                <xs:element ref="AttributeLimit"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="Abstract" type="xs:string"/>
    <xs:element name="OnlineResource" type="xs:anyURI"/>
```

```
<xs:element name="KeywordList">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="Keyword" maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="Keyword" type="xs:string"/>
<xs:element name="ContactInformation">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="ContactPersonPrimary"/>
            <xs:element ref="ContactAddress"/>
            <xs:element ref="ContactFacsimileTelephone"/>
            <xs:element ref="ContactElectronicMailAddress"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="ContactPersonPrimary">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="ContactPerson"/>
            <xs:element ref="ContactOrganization"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="ContactPerson" type="xs:string"/>
<xs:element name="ContactOrganization" type="xs:string"/>
<xs:element name="ContactAddress">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="AddressType"/>
            <xs:element ref="Address"/>
            <xs:element ref="City"/>
            <xs:element ref="StateOrProvince"/>
            <xs:element ref="PostCode"/>
            <xs:element ref="Country"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="AddressType" type="xs:string"/>
<xs:element name="Address" type="xs:string"/>
<xs:element name="City" type="xs:string"/>
<xs:element name="StateOrProvince" type="xs:string"/>
<xs:element name="PostCode" type="xs:string"/>
<xs:element name="Country" type="xs:string"/>
<xs:element name="ContactFacsimileTelephone" type="xs:string"/>
<xs:element name="ContactElectronicMailAddress" type="xs:string"/>
<xs:element name="GeolinkidsLimit" type="xs:positiveInteger"/>
<xs:element name="AttributeLimit" type="xs:positiveInteger"/>
<!-- ** GLS_Capabilities : Capability  ** -->
<xs:element name="Capability">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="Request"/>
            <xs:element ref="Exception"/>
            <xs:element ref="GeoLinkables"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="Request">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="GetCapabilities"/>
            <xs:element ref="GeoLink"/>
```

```
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="GetCapabilities">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Format"/>
                <xs:element ref="DCPType"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="GeoLink">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Format"/>
                <xs:element ref="DCPType"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="Exception">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Format"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="GeoLinkables">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="DataStore" maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="DataStore">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Host"/>
                <xs:element ref="Supports"/>
                <xs:element ref="Framework" maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="Host" type="xs:string"/>
    <xs:element name="Supports">
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:enumeration value="WMS"/>
                <xs:enumeration value="WFS"/>
                <xs:enumeration value="WAREHOUSE"/>
                <xs:enumeration value="GIS"/>
                <xs:enumeration value="OTHER"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:element>
    <xs:element name="Framework">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Organization"/>
                <xs:element ref="Domain"/>
                <xs:element ref="Name"/>
                <xs:element ref="Title"/>
                <xs:element ref="Description"/>
                <xs:element ref="Version"/>
                <xs:element ref="ReleaseDate"/>
                <xs:element ref="Documentation"/>
```

```
                    <xs:element ref="GeolinkageField"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="Organization" type="xs:string"/>
        <xs:element name="Domain" type="xs:string"/>
        <xs:element name="Description" type="xs:string"/>
        <xs:element name="Version" type="xs:string"/>
        <xs:element name="ReleaseDate" type="xs:date"/>
        <xs:element name="Documentation" type="xs:anyURI"/>
        <xs:element name="GeolinkageField" type="xs:string"/>
</xs:schema>
```

A.2 GetData Response Schema

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
    <!-- **  GeoLinking Service:  GeoLink Schema version 0.9.1 - Top-Level Element.  ** -->
    <xs:element name="GLS_GeoLink">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="Service" type="ServiceType"/>
                <xs:element name="Framework" type="FrameworkType"/>
                <xs:element ref="GDAS"/>
                <xs:element name="Dataset" type="DatasetType"/>
                <xs:element ref="SLD"/>
                <xs:element ref="Cache"/>
                <xs:element name="GeoLinkage" type="GeoLinkageType"/>
            </xs:sequence>
            <xs:attribute name="version" type="xs:string" use="required"/>
        </xs:complexType>
    </xs:element>
    <!-- ** Elements Used In Multiple Places. ** -->
    <xs:element name="Organization" type="xs:string"/>
    <xs:element name="Domain" type="xs:string"/>
    <xs:element name="Name" type="xs:string"/>
    <xs:element name="Title" type="xs:string"/>
    <xs:element name="Description" type="xs:string"/>
    <xs:element name="Version" type="xs:string"/>
    <xs:element name="ReleaseDate" type="xs:string"/>
    <xs:element name="Documentation" type="xs:string"/>
    <xs:element name="GeolinkageField" type="xs:string"/>
    <!-- ** GLS_Geolink : Service  ** -->
    <xs:complexType name="ServiceType">
        <xs:sequence>
            <xs:element ref="Name"/>
            <xs:element ref="Title"/>
            <xs:element ref="Abstract"/>
            <xs:element ref="OnlineResource"/>
            <xs:element name="ContactInformation" type="ContactInformationType"/>
        </xs:sequence>
    </xs:complexType>
    <xs:element name="Abstract" type="xs:string"/>
    <xs:element name="OnlineResource" type="xs:string"/>
    <xs:complexType name="ContactInformationType">
        <xs:sequence>
            <xs:element name="ContactPersonPrimary" type="ContactPersonPrimaryType"/>
            <xs:element name="ContactAddress" type="ContactAddressType"/>
            <xs:element ref="ContactFacsimileTelephone"/>
            <xs:element ref="ContactElectronicMailAddress"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="ContactPersonPrimaryType">
        <xs:sequence>
            <xs:element ref="ContactPerson"/>
            <xs:element ref="ContactOrganization"/>
        </xs:sequence>
    </xs:complexType>
    <xs:element name="ContactPerson" type="xs:string"/>
    <xs:element name="ContactOrganization" type="xs:string"/>
    <xs:complexType name="ContactAddressType">
        <xs:sequence>
            <xs:element ref="AddressType"/>
            <xs:element ref="Address"/>
            <xs:element ref="City"/>
            <xs:element ref="StateOrProvince"/>
            <xs:element ref="PostCode"/>
            <xs:element ref="Country"/>
```

```
    </xs:sequence>
</xs:complexType>
<xs:element name="AddressType" type="xs:string"/>
<xs:element name="Address" type="xs:string"/>
<xs:element name="City" type="xs:string"/>
<xs:element name="StateOrProvince" type="xs:string"/>
<xs:element name="PostCode" type="xs:string"/>
<xs:element name="Country" type="xs:string"/>
<xs:element name="ContactFacsimileTelephone" type="xs:string"/>
<xs:element name="ContactElectronicMailAddress" type="xs:string"/>
<!-- ** GLS_Geolink : Framework  ** -->
<xs:complexType name="FrameworkType">
    <xs:sequence>
        <xs:element ref="Organization"/>
        <xs:element ref="Domain"/>
        <xs:element ref="Name"/>
        <xs:element ref="Title"/>
        <xs:element ref="Description"/>
        <xs:element ref="Version"/>
        <xs:element ref="ReleaseDate"/>
        <xs:element ref="Documentation"/>
        <xs:element ref="GeolinkageField"/>
    </xs:sequence>
</xs:complexType>
<!-- ** GLS_Geolink : GDAS  ** -->
<xs:element name="GDAS" type="xs:string"/>
<!-- ** GLS_Geolink : Dataset  ** -->
<xs:complexType name="DatasetType">
    <xs:sequence>
        <xs:element ref="Organization"/>
        <xs:element ref="Domain"/>
        <xs:element ref="Name"/>
        <xs:element ref="Title"/>
        <xs:element ref="Description"/>
        <xs:element ref="Version"/>
        <xs:element ref="ReleaseDate"/>
        <xs:element ref="Documentation"/>
        <xs:element ref="GeolinkageField"/>
        <xs:element name="Attribute" type="AttributeType"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="AttributeType">
    <xs:sequence>
        <xs:element ref="Name"/>
        <xs:element ref="Units"/>
        <xs:element ref="Type"/>
        <xs:element ref="Title"/>
        <xs:element ref="Description"/>
        <xs:element ref="Version"/>
        <xs:element ref="ReleaseDate"/>
        <xs:element ref="Documentation"/>
    </xs:sequence>
</xs:complexType>
<xs:element name="Units" type="xs:string"/>
<xs:element name="Type" type="xs:string"/>
<!-- ** GLS_Geolink :SLD  ** -->
<xs:element name="SLD" type="xs:string"/>
<!-- ** GLS_Geolink : Cache  ** -->
<xs:element name="Cache" type="xs:string"/>
<!-- ** GLS_Geolink : Geolinkage  ** -->
<xs:complexType name="GeoLinkageType">
    <xs:sequence>
        <xs:element name="DataStore" type="DataStoreType" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
```

```
    <xs:complexType name="DataStoreType">
        <xs:sequence>
            <xs:element ref="Host"/>
            <xs:element ref="Supports"/>
            <xs:element ref="Attribute"/>
            <xs:element ref="Status"/>
            <xs:element ref="Url"/>
        </xs:sequence>
    </xs:complexType>
    <xs:element name="Host" type="xs:string"/>
    <xs:element name="Supports">
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:enumeration value="WMS"/>
                <xs:enumeration value="WFS"/>
                <xs:enumeration value="WAREHOUSE"/>
                <xs:enumeration value="GIS"/>
                <xs:enumeration value="OTHER"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:element>
    <xs:element name="Attribute" type="xs:string"/>
    <xs:element name="Status" type="xs:string"/>
    <xs:element name="Url" type="xs:string"/>
</xs:schema>
```

A.3 Service Exception Schema

This subclause contains the Service Exception Schema corresponding to this version of the GLS specification. This subclause also summarizes the defined exception codes and their meanings.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="http://www.opengis.net/ogc" xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:ogc="http://www.opengis.net/ogc" elementFormDefault="qualified" attributeFormDefault="unqualified">
    <xs:element name="ServiceExceptionReport">
        <xs:annotation>
            <xs:documentation> The ServiceExceptionReport element contains one or more
ServiceException elements that describe a service exception. </xs:documentation>
        </xs:annotation>
        <xs:complexType>
            <xs:sequence>
                <xs:element name="ServiceException" type="ogc:ServiceExceptionType" minOccurs="0"
maxOccurs="unbounded">
                    <xs:annotation>
                        <xs:documentation> The Service exception element is used to describe a
service exception. </xs:documentation>
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
            <xs:attribute name="version" type="xs:string" fixed="1.2.0"/>
        </xs:complexType>
    </xs:element>
    <xs:complexType name="ServiceExceptionType">
        <xs:annotation>
            <xs:documentation> The ServiceExceptionType type defines the ServiceException element. The
content of the element is an exception message that the service wished to convey to the client application.
</xs:documentation>
```

```
        </xs:annotation>
        <xs:simpleContent>
            <xs:extension base="xs:string">
                <xs:attribute name="code" type="xs:string">
                    <xs:annotation>
                        <xs:documentation> A service may associate a code with an exception by using the code
attribute. </xs:documentation>
                    </xs:annotation>
                </xs:attribute>
                <xs:attribute name="locator" type="xs:string" use="optional">
                    <xs:annotation>
                        <xs:documentation> The locator attribute may be used by a service to indicate to a client
where in the client's request an exception was encountered. If the request included a 'handle' attribute, this may be
used to identify the offending component of the request.
Otherwise the service may try to use other means to locate the exception such as line numbers or byte offset from
the beginning of the request, etc ... </xs:documentation>
                    </xs:annotation>
                </xs:attribute>
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
</xs:schema>
```

Table A.1 — Exception codes defined by this specification

Exception Code	Meaning
InvalidFormat	Request contains a Format not offered by the service instance.
FrameworkNotDefined	Request is for a Framework not offered by the service instance.
FrameworkNotDefinedOnGDAS	Request is for a Framework not offered by the GDAS service instance referenced.
FrameworkVersionNotDefined	Request is for a Framework Version not offered by the service instance.
FrameworkVersionNotDefinedOnGDAS	Request is for a Framework Version not offered by the GDAS service instance referenced.
DatasetNotDefinedOnGDAS	Request is for a Dataset not offered by the GDAS service instance referenced.
AttributeNotDefinedOnGDAS	Request is for a Attribute not offered by the GDAS service instance referenced.
AttributeLimitExceeded	Request exceeds allowable number of Attributes on this service.
AttributeLimitExceededOnGDAS	Request exceeds allowable number of Attributes on the GDAS service instance referenced.
GeoLinkIdNotDefined	Request is for a GeoLinkId that is not offered by this service instance.
GeoLinkIdNotDefinedOnGDAS	Request is for a GeoLinkId that is not offered by the GDAS service instance referenced.
GeoLinkIdLimitExceeded	Request exceeds allowable number of GeoLinkIds on this service.
GeoLinkIdLimitExceededOnGDAS	Request exceeds allowable number of GeoLinkIds on the GDAS service instance referenced.

CurrentUpdateSequence	Value of (optional) UpdateSequence parameter in GetCapabilities request is equal to current value of Capabilities XML update sequence number.
InvalidUpdateSequence	Value of (optional) UpdateSequence parameter in GetCapabilities request is greater than current value of Capabilities XML update sequence number.
MissingParameterValue	Request does not include a parameter value, and the service instance did not declare a default value for that parameter.
InvalidParameterValue	Request contains an invalid parameter value.

Annex B
(informative)

XML Examples

B.1 Introduction

As an aid to understanding and a guide for implementation, this annex contains **example** XML which is valid according to the XML schemas in Annex A. Implementers should consult the main body of the specification document and the schemas to ensure compliance rather than editing this XML without verification.

B.2 Example GetCapabilities request using the HTTP GET method

http://foo.bar/foo?SERVICE=GLS&REQUEST=GetCapabilities&VERSION=0.9.1

B.3 Example GetCapabilities XML response

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<GLS_Capabilities version="0.9.1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://www.geoconnections.org/architecture/technical/services/geolinking/gls_
0_9_1_cap.xsd">
    <Service>
        <Name>CGDI:GS</Name>
        <Title>2001 Census Frameworks Geolinking Service</Title>
        <Abstract>This service allows geolinked data to be merged with standard census geographical areas from
the 2001 Census of Agriculture for Canada.  The frameworks supported are Provinces, Census Districts, Census
SubDivisions, and Census Agricultural Areas.  This service makes the geolinked data available as a WMS layer
and a GIS (SDE) layer.</Abstract>
        <OnlineResource>http://aafc.gc.ca/</OnlineResource>
        <KeywordList>
            <Keyword>Census of Agriculture</Keyword>
            <Keyword>Canada</Keyword>
            <Keyword>Statistics Canada</Keyword>
            <Keyword>Statistics</Keyword>
        </KeywordList>
        <ContactInformation>
            <ContactPersonPrimary>
                <ContactPerson>Peter Brimacombe</ContactPerson>
                <ContactOrganization>Agriculture and Agri-Food Canada</ContactOrganization>
            </ContactPersonPrimary>
            <ContactAddress>
                <AddressType>postal</AddressType>
                <Address>960 Carling Ave.</Address>
                <City>Ottawa</City>
                <StateOrProvince>Ontario</StateOrProvince>
                <PostCode>K1A 0C6</PostCode>
                <Country>Canada</Country>
            </ContactAddress>
            <ContactFacsimileTelephone>613-759-1937</ContactFacsimileTelephone>
            <ContactElectronicMailAddress>bp@agr.gc.ca</ContactElectronicMailAddress>
        </ContactInformation>
```

```
        <GeolinkidsLimit>1000</GeolinkidsLimit>
        <AttributeLimit>1</AttributeLimit>
    </Service>
    <Capability>
        <Request>
            <GetCapabilities>
                <Format>text/xml</Format>
                <DCPType>
                    <HTTP>
                        <Get onlineResource="http://agr.gc.ca/gls?"/>
                    </HTTP>
                </DCPType>
            </GetCapabilities>
            <GeoLink>
                <Format>text/xml</Format>
                <DCPType>
                    <HTTP>
                        <Get onlineResource="http://agr.gc.ca/gls?"/>
                    </HTTP>
                </DCPType>
            </GeoLink>
        </Request>
        <Exception>
            <Format>text/xml</Format>
        </Exception>
        <GeoLinkables>
            <DataStore>
                <Host>http://wms1.agr.gc.ca/stc</Host>
                <Supports>WMS</Supports>
                <Framework>
                    <Organization>Statistics Canada</Organization>
                    <Domain>statcan.gc.ca</Domain>
                    <Name>provinces</Name>
                    <Title>Provinces and Territories of Canada</Title>
                    <Description>Provincial and Territorial boundaries used for the 2001 Census of Canada.
</Description>
                    <Version>2001</Version>
                    <ReleaseDate>2001-06-03</ReleaseDate>

    <Documentation>http://geodepot.statcan.ca/Diss/Data/Geography/n_cd_e.cfm</Documentation>
                    <GeolinkageField>pr</GeolinkageField>
                </Framework>
                <Framework>
                    <Organization>Statistics Canada</Organization>
                    <Domain>statcan.gc.ca</Domain>
                    <Name>cd</Name>
                    <Title>Census Divisions</Title>
                    <Description>Census divisions defined for the 2001 Census of Canada. </Description>
                    <Version>2001</Version>
                    <ReleaseDate>2001-06-03</ReleaseDate>

    <Documentation>http://geodepot.statcan.ca/Diss/Data/Geography/n_cd_e.cfm</Documentation>
                    <GeolinkageField>cd</GeolinkageField>
                </Framework>
            </DataStore>
            <DataStore>
                <Host>http://hp001.gis.agr.gc.ca/stc</Host>
                <Supports>GIS</Supports>
                <Framework>
                    <Organization>Statistics Canada</Organization>
                    <Domain>statcan.gc.ca</Domain>
                    <Name>csd</Name>
                    <Title>Census Subdivisions</Title>
                    <Description>Census subdivisions defined for the 2001 Census of Canada. </Description>
                    <Version>2001</Version>
```

```
                    <ReleaseDate>2001-06-03</ReleaseDate>

    <Documentation>http://geodepot.statcan.ca/Diss/Data/Geography/n_cd_e.cfm</Documentation>
                    <GeolinkageField>csd</GeolinkageField>
                </Framework>
            </DataStore>
        </GeoLinkables>
    </Capability>
</GLS_Capabilities>
```

B.4 Example GeoLink request using the HTTP GET method

```
http://foo.bar/gls?
    SERVICE=GLS&
    REQUEST=GetData&
    VERSION=0.9.1&
    FRAMEWORKHOST=http://ec.gc.ca/eco&
    GDAS=http://map1.agr.gc.ca/gdas&
    GDASVersion=0.9.1&
    DOMAIN=agr.gc.ca&
    FRAMEWORK=ecodistricts&
    FRAMEWORKVERSION=1&
    DATASET=RG_TEMP&
    ATTRIBUTE=tempmeanjan&
    GEOLINKIDS=1,5,6,7,8,9&
    SLD=http://map1.agr.gc.ca/sld/101.sld&
    CACHE=5
```

B.5 Example GeoLink XML request using the HTTP POST method

```
<?xml version="1.0" encoding="UTF-8"?>
    <GeoLink version="0.9.1" service="GLS">
    <FrameworkHost>http://ec.gc.ca/eco</FrameworkHost>
    <GDAS>http://map1.agr.gc.ca/gdas</GDAS>
    <GDASVersion>http://map1.agr.gc.ca/gdas</GDASVersion>
    <Framework>
        <Name>ecodistricts</Name>
        <Name>ecodistricts</Name>
        <Version>1</Version>
    </Framework>
    <Dataset>RG_TEMP</Dataset>
    <Attribute>tempmeanjan</Attribute>
    <GeolinkIds>
        <I>1</I>
        <I>5</I>
        <I>6</I>
        <I>7</I>
        <I>8</I>
        <I>9</I>
    </GeolinkIds>
</GetData>
```

B.6 Example GeoLink XML response

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<GLS_GeoLink version="0.9.1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://www.geoconnections.org/architecture/technical/services/geolinking/gls_
0_9_1_geo.xsd">
    <Service>
        <Name>CGDI:SRS</Name>
        <Title>Agriculture Canada GeoLinking Service</Title>
        <Abstract>This service registry contains information about all CGDI compliant services in AAFC, as well as
those of partner agencies that provide services used by NLWIS.</Abstract>
        <OnlineResource>http://agr.gc.ca/</OnlineResource>
        <ContactInformation>
            <ContactPersonPrimary>
                <ContactPerson>Peter Brimacombe</ContactPerson>
                <ContactOrganization>Agriculture and Agri-Food Canada</ContactOrganization>
            </ContactPersonPrimary>
            <ContactAddress>
                <AddressType>postal</AddressType>
                <Address>960 Carling Ave.</Address>
                <City>Ottawa</City>
                <StateOrProvince>Ontario</StateOrProvince>
                <PostCode>K1A 0C6</PostCode>
                <Country>Canada</Country>
            </ContactAddress>
            <ContactFacsimileTelephone>613-759-1937</ContactFacsimileTelephone>
            <ContactElectronicMailAddress>brimacombep@agr.gc.ca</ContactElectronicMailAddress>
        </ContactInformation>
    </Service>
    <Framework>
        <Organization>Statistics Canada</Organization>
        <Domain>ec.gc.ca</Domain>
        <Name>provinces</Name>
        <Title>Provinces and Territories of Canada</Title>
        <Description>Provincial and Territorial boundaries used for the 2001 Census of Canada.</Description>
        <Version>2001</Version>
        <ReleaseDate>2001-06-03</ReleaseDate>
        <Documentation>http://geodepot.statcan.ca/Diss/Data/Geography/n_cd_e.cfm</Documentation>
        <GeolinkageField>pr</GeolinkageField>
    </Framework>
    <GDAS>http://statcan.gc.ca/gdas</GDAS>
    <Dataset>
        <Organization>Statistics Canada</Organization>
        <Domain>statcan.ca</Domain>
        <Name>2001AgCensus</Name>
        <Title>2001 Census of Agriculture</Title>
        <Description>2001 Census of Agriculture</Description>
        <Version>2001</Version>
        <ReleaseDate>2003-06-12</ReleaseDate>
        <Documentation>http://www.statcan.ca/english/agcensus2001/about.htm</Documentation>
        <GeolinkageField>province</GeolinkageField>
        <Attribute>
            <Name>cattlecalves</Name>
            <Units>count</Units>
            <Type>interval</Type>
            <Title>Total cattle and calves</Title>
            <Description>Total cattle and calves assessed by the 2001 Census of Agriculture. </Description>
            <Version>2001</Version>
            <ReleaseDate>2003-06-12</ReleaseDate>
            <Documentation>http://www.statcan.ca/english/agcensus2001/about.htm</Documentation>
```

```
        </Attribute>
    </Dataset>
    <SLD>http://wms1.agr.gc.ca/stc/animals_SLD.xml</SLD>
    <Cache>30</Cache>
    <GeoLinkage>
        <DataStore>
            <Host>http://wms1.agr.gc.ca/stc</Host>
            <Supports>WMS</Supports>
            <Attribute>cattlecalves</Attribute>
            <Status>Succeeded</Status>
            <Url>http://map1.agr.gc.ca/tmp/109821712</Url>
        </DataStore>
        <DataStore>
            <Host>http://hp001.gis.agr.gc.ca/stc</Host>
            <Supports>GIS</Supports>
            <Attribute>cattlecalves</Attribute>
            <Status>Succeeded</Status>
            <Url>n/a</Url>
        </DataStore>
    </GeoLinkage>
</GLS_GeoLink>
```

Annex C
(informative)

Changes from previous versions of the Geolinking specification

C.1 Introduction

Previous versions of this specification were prepared under the auspices of the Geoconnections (Natural Resouces Canada). This annex details changes made to earlier versions of the specification produced by that organization.

C.2 Changes between version 0.8.2 and 0.9.0

- Added the ability to request multiple attributes in a single request:
- New ATTRIBUTES request parameter
- Former ATTRIBUTE parameter deprecated
- Set maxOccurs=unbounded for <Attribute> and <V> in **GetData** response schema
- Added the ability to pass a filter in the **GetData** request using the optional **GeolinkIds** parameter
- Made some parameters of **GetData** mandatory (they were all optional before) and added a note explaining that simple/static services can still operate in that context.
- Added definition of XML-encoded POST requests (**GetCapabilities** and **GetData**).
- Modified the **GetCapabilities** example to use text/xml instead of custom MIME types
- (application/cgdi.gdas_xml and application/cgdi.error_xml) in the Service definitions.
- Allow one or more Datasets inside a Framework in Capabilities (we used to be limited to one Dataset per Framework). Updated the text of the spec and the Capabilities schema accordingly.

C.2 Changes between version 0.8.1 and 0.9.1

- Reverted to ATTRIBUTE parameter in all cases. ATTRIBUTES parameter from 0.9.0 no longer valid.
- Added <GeolinkidsLimit> to indicate if the service supports selection of specific GeoLink Ids.
- Added <AttributeLimit> to indicate if the service supports simultaneous retrieval of multiple attributes.
- Dropped MIME type "application/vnd.ogc.*" in favour of "text/xml"
- Added <Url> to <Datastore>
- Changed Host to FrameworkHost in GeoLink Request
- Added GDASVersion to GeoLink Request

- Added FrameworkDomain to GeoLink Request
- Changed Framework to FrameworkName in GeoLink request
- Added DatasetDomain to GeoLink request
- Changed Dataset to DatasetName in GeoLink request
- Changed Interval to Measure in Attribute.Type

Annex D
(informative)

Plans for future developments

It may be desireable to merge the GDAS and Geolinking specifications, much as WMS specifies a client and a server side. In this case, there would be four operations: GetCapabilities, GetData, and GeoLink. By passing the URL of a GDAS server along with a GetCapabilities request, the GLS could put together a listing of what it can link, based on what is available locally, and from the other server.

Annex E
(informative)

Use Cases

Table A.2 — Create a WMS layer on the fly use case

Use case description	
Name	Create a WMS layer on the fly
Priority	
Description	Client identifies data for Geolinking, and uses a GeoLinking service to dynamically turn the contents of a GDAS stream into a WMS layer, which is then displayed by the client.
Precondition	Attribute data is available from a GDAS server. A Geolinking Service has been tied to a WMS server. The needed catalog server is available and already known by the client. The catalog server supports data schemas for all needed types of metadata. The archive and catalog servers store the needed metadata.
Flow of events – basic path	
1)	Client queries a catalog server to find what attributes are available from a GDAS for geolinking to some geospatial framework. (Note 1)
2)	Client queries a catalog server to find what Geolinking servers accommodate that geospatial framework. (Note 1)
3)	Client reviews query results and identifies the GDAS attribute to be mapped, and the Geolinking server to be used.
4)	Client requests the Geolinking server to use the GDAS attribute to create a WMS layer. (Note 2)
5)	Geolinking server requests the appropriate attribute data from the GDAS server. (Note 3)
6)	Geolinking server joins the attribute data to the framework data.
7)	Geolinking server uses that joined attribute to create a WMS layer.
8)	If needed, Geolinking server publishes the new WMS layer to a catalog. (Note 4)
9)	Geolinking server returns to the client the URL of the online resource for the WMS layer.
10)	The client places a request for the WMS layer and displays it. (Note 5)
Flow of events – alternative paths	
	If the client knows of the existence of the GDAS and/or Geolinking servers, steps 1 and/or 2 may be skipped.
Postcondition	If the WMS layer is not transient, the existence of the WMS layer is stored in a catalog.
NOTE 1	Client directly interacts with a catalog server through its interface.
NOTE 2	Client directly interacts with a Geolinking server through its interface.
NOTE 3	Geolinking server acts as a client and directly interacts with the GDAS server through its interface.
NOTE 4	Geolinking server directly interacts with the catalog server through its interface.
NOTE 5	Client directly interacts with WMS server server through its interface.

Bibliography