



FDSN Web Service Specification Commonalities

Version 1.2

2019-06-27

Purpose

To specify the common characteristics of web service interfaces for the exchange of time series data, related metadata, event parameter and other data within the context of the International Federation of Digital Seismograph Networks (FDSN). These common characteristics are referenced by multiple web service interface specifications, each defined in separate documents.

Versioning

Web service implementations are versioned according the following three-digit (x.y.z) pattern:

SpecMajor.SpecMinor.Implementation

Where the fields have the following meaning:

SpecMajor: The major specification version, all implementations sharing this *SpecMajor* value will be backwards compatible with all prior releases. Values start at 1.

SpecMinor: The minor specification version, incremented when optional parameters or behavior is added to the previous specification but backwards compatibility is maintained with the previous major versions, i.e. all 1.# service versions will be compatible with version 1.0. Values start at 0.

Implementation: The implementation version, an integer identifier specific to the data center implementation. Useful to track service updates for bug fixes, etc. but with no implication on conformance to the specification.

Together the *SpecMajor* and *SpecMinor* versions imply a minimum expected behavior of a given service. This versioning scheme allows clients to expect specific behavior based on the *SpecMajor* version, while allowing the FDSN to extend the services with optional parameters while maintaining backwards compatibility. Each version number is service specific, there is no implication that *SpecMajor* version numbers across services are related.

Calling pattern

The services will be invoked using a subset of REST and HTTP methods. In particular all services should be invoked using the HTTP GET method when possible and the HTTP POST method where the selection parameters are potentially large in size.

Service path and port

The following base URI pattern is to be used at each data center implementing FDSN services:

`<site>/fdsnws/<service>/<majorversion>/`

where **service** is the name of the service and **majorversion** is an integer value specifying the major specification (*SpecMajor*) version supported by the service. A **site** is the domain name of the data

center hosting FDSN web services. For example the base URI for version 1 of the FDSN station service at ORFEUS might be:

www.orfeus-eu.org/fdsnws/station/1/

All services should be available on TCP/IP port 80.

Common service methods

All services should support these common methods:

query – to submit a data or information request

version – to request the full service version (*SpecMajor.SpecMinor.Implementation*)

application.wadl – to request a WADL for the interface

Details of each method and further methods specific to a service are documented in the relevant service specifications.

For example, submitting queries to a service named station at ORFEUS would be performed using this URI:

<http://www.orfeus-eu.org/fdsnws/station/1/query?<key=value parameters>>

Similarly requesting the implementation version of the same service would be done using this URI:

<http://www.orfeus-eu.org/fdsnws/station/1/version>

which returns as simple text string indicating the software version, e.g. “1.2.8”.

Requesting the WADL for a the same interface:

<http://www.orfeus-eu.org/fdsnws/station/1/application.wadl>

Minimum functionality

Implementations of these service interfaces should support all methods specified as required. Additionally, interfaces should conform to the calling patterns and expected results identified in this document to be considered conformant with the specification. Authentication methods are an exception, and only need to be supported by data centers that wish to require authentication for access to specific data sets.

Each service definition includes required and optional parameters; an implementation must support the required parameters to be considered conformant. The optional parameters supported by any given implementation should be specified in the WADL returned by the service.

Request URI construction

Request URIs to FDSN web services must not exceed 2000 bytes in length including the URL encoding; most common requests will be much shorter. Requests should be URL Encoded as a standard practice before submission to the service, but services should handle non-encoded submissions as far as is possible.

Common service responses

General responses that should be used by all services are outlined below. Responses specific to each service may be present individual specification.

No data selected

If a properly formatted request is submitted but would result in no data being returned to the user the service should return **HTTP status 204** (No Content) by default. Alternatively, the optional *nodata* parameter can be used to request that **HTTP status 404** (Not Found) be returned instead. A 404 status helps clarify situations where a 204 may be confusing, such as a web browser user, by including a message in the response. Clients should be aware that the 404 status code may be returned for error conditions such as problems with the web application or application container. Due to the multiple meanings for a 404 status, the 204 status is preferred for programmatic (non-browser) access.

Errors messages

All errors reported to the client, either HTTP 4xx or 5xx status codes, should include an error message transmitted as MIME type **text/plain** using the following pattern:

```
Error <CODE>: <SIMPLE ERROR DESCRIPTION>
```

```
<MORE DETAILED ERROR DESCRIPTION>
```

```
Usage details are available from <SERVICE DOCUMENTATION URI>
```

```
Request:  
<SUBMITTED URL>
```

```
Request Submitted:  
<UTC DATE TIME>
```

```
Service version:  
<3-LEVEL VERSION>
```

Result set limitations

Limitations on the amount of information returned for any given request may be imposed independently for each service by each data center. If a client submits a request that would result in a data set beyond the service limit the service should return an **HTTP status 413** (Request Entity Too Large). A **413** should also be returned when the request itself is too large, the standard meaning of the status. In most cases, differentiating between these two cases will be obvious.

WADL conventions

The WADL documents returned by FDSN services should follow these guidelines:

- All public parameters supported by an interface shall be documented in the WADL.
- Parameters supported by the interface shall be documented using the long version of the parameter name and not the abbreviated version.
- Parameters not included in the specification (data center-specific extensions) shall be documented with a type and a short description.

Common HTTP status codes

Table 1 includes a list of common status codes returned by FDSN web services. In most cases these will be the only codes returned by the services. Under error, maintenance or otherwise unusual conditions a client may receive other HTTP codes generated by web service containers, layer 7 routers (e.g. load balancers), firewalls, etc.

Table 1. Common HTTP status codes returned by FDSN services

Code	Description
200	Successful request, results follow
204	Request was properly formatted and submitted but no data matches the selection
400	Bad request due to improper specification, unrecognized parameter, parameter value out of range, etc.
401	Unauthorized, authentication required
403	Authentication failed or access blocked to restricted data
404	Alternate to 204 (set via the 'nodata' parameter), normally used for results returned to a web browser.
413	Request would result in too much data being returned or the request itself is too large, returned error message should include the service limitations in the detailed description. Service limits should also be documented in the service WADL.
414	Request URI too large
500	Internal server error
503	Service temporarily unavailable, used in maintenance and error conditions

Common service parameter characteristics

Wildcards and lists in channel constraints parameters

The channel constraint parameters (network, station, location and channel) may include two specific wildcard characters with the following meaning:

- * – Matches zero to many characters
- ? – Matches any single character

The channel constraint parameters may also be submitted as comma-separated lists in order to select two or more values with a single request. For example, the channel parameter may be used to specify multiple channels:

channel=LHE,LHN,LHZ,BHZ (the individual values may also include wildcards)

Time parameter values

All **time** values are specified in UTC and should use one particular variation of the ISO 8601 standard with the addition of sub-seconds:

YYYY-MM-DDTHH:MM:SS.ssssss

Where the fields are as specified in ISO 8601 and a “T” separates the date and time. The “**ssssss**” are the sub-seconds expressed between 1/10 of a second to microsecond resolution; they do not all need to be present.

Two shorter variations for specifying time should be supported:

YYYY-MM-DDTHH:MM:SS (implies a time of **YYYY-MM-DDTHH:MM:SS.000000**)

where the sub-seconds are interpreted to be zeros and

YYYY-MM-DD (implies a time of **YYYY-MM-DDT00:00:00.000000**)

where the hour, minute, second and sub-second are interpreted to be zeros.

Time parameter interpretation

The *starttime* parameter should be interpreted as selecting any data or information (time series samples, earthquake origins, metadata epochs) at times on or later than the value specified. Similarly, the *endtime* selects any data or information at times on or prior to the value specified. The selection shall include any metadata epochs intersecting with the specified time range.

The *startbefore*, *startafter*, *endbefore* and *endafter* parameters, used only for the **fdsnws-station** service, specify the selection of information strictly starting or ending before or after a specified time value; they do not match any information occurring exactly at the time specified. These selection parameters are specifically for metadata and are useful for matching information that would otherwise be difficult with the other time parameters.

String, float, boolean and integer parameter types

String type parameters may include ASCII characters, including the asterisk (*) and question mark (?) that have special meaning as wildcards for specific parameters.

Float type parameters should be specified in decimal notation, for example “98.1023”. Scientific notation using an exponent or other representations will not be accepted and should generate errors.

Boolean type parameters should be specified as either “TRUE” or “FALSE”, case insensitive.

Integer type parameters should be specified in decimal notation without a decimal point, exponential or other notation.

Blank SEED location identifier selection

Blank or empty location identifiers are encoded in SEED as two space characters (ASCII decimal 32). These may be specified by setting the *location* parameter to a string of two spaces, but care must be taken to properly encode the URI. For clarity and convenience, blank SEED location identifiers may also be specified by setting the *location* parameter to “--” (two dashes) which the service should translate to two space characters as needed.

References

SEED data format - http://www.iris.edu/manuals/SEEDManual_V2.4.pdf

Web Application Description Language (WADL) - <http://www.w3.org/Submission/wadl/>

Credits

This document was designed and is maintained by the Incorporated Research Institutions for Seismology (IRIS), Data Services on behalf of the FDSN.

Changes

Note: Through 2019-06-27 the change entries reflect the original, monolithic specification document, which included **fdsnws-station**, **fdsnws-dataselect**, and **fdsnws-event** service definitions, and remain as a record.

2012-07-12 • Initial specification, release candidate 1

2012-07-18

1. Remove <Total> elements from **catalog** and **contributor** XML responses for **fdsnws-event**.
2. Designate the **fdsnws-station** specification as provisional until the FDSN StationXML schema is defined.

2012-11-20

1. Include example request URI's for each service.
2. Remove provisional status on fdsnws-station.

2013-04-24 • Version 1.0 final

1. Replace parameter name *preferredonly* with *includeallorigins* in the fdsnws-event section of Table 1 and make the default value FALSE. This now matches other references to the parameter names and was the original intention.
2. Change the example of exclusive parameter sub-groups to refer to **fdsnws-station** as a more appropriate illustration (page 4).
3. Add link to FDSN StationXML specification in references and remove provisional language.
4. Remove the DRAFT designation.

2013-09-18 • Integrate proposed version 1.1 changes

1. Add designation of required versus optional parameters for each service interface.
2. Allow **fdsnws-station** to accept arbitrary selections via the POST method. Support for this method is required for all 1.1 implementations of fdsnws-station.
3. Add optional *format* parameter to each service definition. The default formats of miniSEED, StationXML and QuakeML remain the same for each interface. For station and event, document an alternate, simple text format to return when *format=text* including examples.

4. Add optional *nodata* parameter to each service definition that accepts values of '204' (default) and '404'. This parameter controls the HTTP status code to return when the request was successful but matched no data.
5. Add optional *matchtimeseries* parameter to the **fdsnws-station** definition. This parameter is used to limit results to selections that match time series data availability.
6. Clarify definition of *starttime* and *endtime* parameters to explicitly include metadata epochs intersection with the specified time range.
7. Clarify definition of minimum and maximum latitude and longitude to be inclusive by adding "or equal to" to their description.

2013-10-04 · 1.1 release candidate

1. Add description and clarification for the columns of the text formats for **fdsnws-station** and **fdsnws-event**.
2. Avoid ambiguity of required parameters by re-naming the appropriate column "Support" and changing the values to "Required" and "Optional".
3. Add another example request to the **fdsnws-event** interface illustrating a query for text data and a portion of the response that might be expected.

2013-10-15 · 1.1 release candidate 2

1. Add section describing WADL conventions.
2. Clarify that when the *format* parameter is not specified the default format must be returned.
3. Clarify that an HTTP 413 status shall be returned when either the request entity itself is too large or the resulting data set would be too large, the latter being an extension of the standard meaning.

2013-10-25 · Version 1.1 final

1. Clarify limitations of text output format: vertical bars cannot be included in fields, multiple origins & magnitudes cannot be included, phase arrivals cannot be included.
2. Document intended behavior for the *matchtimeseries* parameter of **fdsnws-station**.
3. Change minimum value for the *limit* parameter to **fdsnws-event** to 1, previously 0.
4. Reformat Table 1 so complete table is visible.
5. Re-number tables, Table 1 was previously used for two different tables.

2015-06-19 · Version 1.1b (clarifying edit, internal version change only)

1. Add *SiteName* column to station-level text template (page 12) for **fdsnws-station** as was the original intention and to match the example and predominant implementations.

2019-04-02 · Version 1.2

1. Update **fdsnws-event** specification to version 1.2, with the following changes:
 - a. addition of the *eventtype* parameter
 - b. addition of the *EventType* column in the text output format (and example)
2. Split Table 2 into Table 2a and 2b to allow additions and increase legibility.

2019-06-27

1. Reformat document to only contain specification commonalities, removing service specific details that will now be in separate documents.

2. Change name to “FDSN-WS-Specification-Commonalities”.