

Syntactic Annotation Service Contract Description - v1.1.1

- Brief description
 - Version
 - Overview and Definitions
 - Assumptions
 - References
 - Known Issues
- ROA Layer API
 - Base URLs
 - Client Responsibilities in Requests to Secured BSP Web Services
 - (1) A client must be configured as a Trusted Application if requests are to be treated other than as Anonymous
 - (2) A Trusted client is expected to pass HTTP Request Headers to identify itself and an authenticated user
 - Schema
 - Annotation Format
 - List Export Formats
 - List Import Formats
 - Format Options
 - Annotation Repository
 - List Annotation Repository instances
 - List an Annotation Repository
 - Text Repository
 - List Text Repository instances
 - List a specific Text Repository
 - Annotation
 - Request a specific annotation or set of annotations
 - Request an annotation or set of annotations matching certain criteria
 - Annotation Template (Document)
 - Create an annotation template for a specific document.
 - Annotation Template (Text)
 - Create an annotation template for supplied text
- SOA Layer API
- Notifications
- Tests

Brief description

The Syntactic Annotation service provides retrieval, conversion and abstract query services for annotated corpora.

Version

This page describes the Syntactic Annotation Service API, v. 1.1.1.

To discover the version and other metadata about deployed service code that fulfills this API, please utilize the [Service Catalog Service](#).

Overview and Definitions

The Syntactic Annotation Service 1.1.1 Release provides a standard API via RESTful HTTP and OSGI Service interfaces for requesting (a) syntactic annotations from supported annotation repositories and (b) templates for creation of syntactic annotations of passages and texts. It is currently configured with access to the Alpheios repository of the Perseus Greek and Latin treebank annotations. It can also retrieve text for annotation template creation from any CTS-API compliant repository, and can be easily configured to access other annotation and text repositories which provide HTTP based RESTful interfaces. The Syntactic Annotation service also supports plugins of [PepperModules](#) for converting between annotation formats. The Release 1.1.0 interface supports the Perseus import format and Perseus and PAULA export formats.

For a full description, visit the [Syntactic Annotation Service Description and Assumptions - v1.1.1 page of this wiki](#) (a child to this page).

Assumptions

None.

References

`${REPOSITORY_ROOT}` is <http://svn.code.sf.net/p/projectbamboo/code/>

- [Entity Diagrams](#)
- Codebase `${REPOSITORY_ROOT}/platform-services/annotation-service/trunk/annotation-service/`

Known Issues

- [BSPSVC-87](#) Cannot configure which TEI tags to ignore when tokenizing text to create annotation templates.
- There is an error registering the Pepper Log Reader service, which results in extremely verbose messages being sent to the Service Mix log. This can be worked around by adding the following to the log config file (normally `org.ops4j.pax.logging.cfg`)

```
log4j.logger.de.hu_berlin.german.korpling.saltnpepper.logger=FATAL,sift
log4j.logger.de.hu_berlin.german.korpling.saltnpepper.pepper-
logReader=FATAL,sift
```

ROA Layer API

RESTful service methods performing Create and List functions over Annotations, Annotation Repositories, Formats and Text Repositories.

Base URLs

It is assumed in this documentation that no centrally-run instances of the Bamboo Services Platform will be running after the project ends on 31 March 2013. Therefore, base URLs are assumed to be on a developer's machine, localhost. The base URL with a port number assumes that the BSP is running unsecured; the URL without a port number assumes that security is enforced and Apache Web Server is intercepting and redirecting service calls. Please see the page [Identity and Access Management - Authentication and Authorization](#) for context, as well as links to installation and configuration instructions for secured instances of BSP.

Note that *ONLY services at v0.9 or greater* will run properly in a secured instance of the BSP.

Currently available base URLs:

- <http://localhost:8181>
- <http://localhost/services>

Client Responsibilities in Requests to Secured BSP Web Services

As of March 2013, policy for the Syntactic Annotation Service permits anonymous access.

This section of the Service API documentation describes a client application's responsibilities when making requests to secured Web Services hosted on the Bamboo Services Platform (including this service).

A secured instance of the Bamboo Services Platform (BSP) implies a significant set of installation and configuration tasks for which the operator of the BSP is responsible. These are described in overview on the wiki page [Identity and Access Management - Authentication and Authorization](#), and in detail on pages linked from that one.

(1) A client must be configured as a Trusted Application if requests are to be treated other than as *Anonymous*

A client application – whether a web app or a simple testing client such as Firefox Poster or curl – may make requests anonymously or as a "Trusted Application." Only a Trusted Application may assert the identity of a user on behalf of whom the request is made, and scoped roles to be assigned to that user; Bamboo Services trust such clients to assert the identity and assigned-roles *only* of users who have authenticated in the current session of application activity. (A special-case type of client application, termed *Innovation Licensed* applications, are trusted to assert the identity of and roles assigned to a fixed set of special-case users without those users having to authenticate in the current session.)

Configuration of client applications are described in detail in this wiki page: [Configure Apache Web Server for Client Auth](#). It is assumed in #2, below, that this configuration has been performed.

(2) A Trusted client is expected to pass HTTP Request Headers to identify itself and an authenticated user

A client application that is *Trusted* in the security context of the Bamboo Trust Federation (cf. [Identity and Access Management - Authentication and Authorization](#)) must augment each request to a service hosted by a secured instance of the Bamboo Services Platform (BSP) with a set of HTTP headers, as follows:

- **X-Bamboo-AppID:** A UUID that identifies the client research environment, application, tool, or service; this UUID is issued as part of the process of *registering* a trusted client in the Bamboo Trust Federation as described in overview on the page [Identity and Access Management - Authentication and Authorization](#), and in detail with respect to physical establishment of trust on the page [Configure Apache Web Server for Client Auth](#). The value of this header is linked to the X.509 certificate by which the application establishes an SSL connection to the BSP host in the registration process, and a match between this *Application ID* and the linked X.509 certificate is checked by the BSP on receipt of *every* request.
- **X-Bamboo-BPID:** A UUID that identifies the logged-in user on whose behalf the request is being sent; this value, a Bamboo Person Identifier, or BPID, is obtained via a call to the *Person Service* that occurs in time between user login and any other service request. See the *Read a BambooPersonId* method of the [Person Service API](#) for details. [†]
- **X-Bamboo-Roles:** A pipe-delimited (|) set of scoped roles asserted by the trusted client to belong to the logged-in user, of the form *role@domain*, which are to be evaluated as factors in the determination of whether the request satisfies policies (access restrictions) that apply to the requested resource. If a user is *authenticated*, the client is expected to include the role *undefined@domain* where *domain* identifies the organization that authenticated the user (example: *undefined@google.com* is a client app's assertion that the user authenticated to Google). This header is otherwise optional (depending on policies governing the requested resource that may require one or more scoped roles for access to be permitted). Example of multiple roles asserted in this header: *roleA@domainOne.xxx|roleB@domainTwo* [‡]

[†] The value of X-Bamboo-BPID is set to the identifier for the application itself (X-Bamboo-AppID) when a client application calls the Person Service to *create* a new Bamboo Person Identifier; or to *retrieve* the BPID for a user based on the identifier of the IdP with which she has logged in and an SHA-256 hash of that IdP's user identifier for the logged-in person.

[‡] Policies and policy evaluation are described on the page [Authorization and Policy](#). Also see [Conventions for representing Identity Providers in the Bamboo Trust Federation](#).

Schema

`#{REPOSITORY_ROOT}` is <http://svn.code.sf.net/p/projectbamboo/code/>

- `#{REPOSITORY_ROOT}/platform-services/annotation-service/trunk/annotation-service/resource/src/main/resources/AnnotationRepositoryListRepresentation.xsd`
- `#{REPOSITORY_ROOT}/platform-services/annotation-service/trunk/annotation-service/resource/src/main/resources/TextRepositoryListRepresentation.xsd`
- `#{REPOSITORY_ROOT}/platform-services/annotation-service/trunk/annotation-service/resource/src/main/resources/AnnotationFormatListRepresentation.xsd`

Annotation Format

A format available for ingestion and serialization of annotation bodies. There are two sub-categories of formats: import and export. Import formats are those formats in which the service is able ingest annotations, and export formats are those formats to which the service is able to serialize annotations. Annotations may be converted from one format to another by supplying a format for export which differs from the annotations' import format.

List Export Formats

Lists the set of available export formats for annotations.

Calling Method and Arguments

Invoked as an HTTP GET method. Send an HTTP request of the form:

```
GET /annotationservice/format/export HTTP/1.1
```

Parameter	Supported Values
Accept Header	application/xml,text/xml,application/json

Response

On success, a response with a "202 Accepted" HTTP status code will be returned:

Parameter	Meaning
HTTP Body	An instance of an AnnotationFormatListRepresentation schema XML document

Exceptions

If an error occurred, some non-2xx code will be returned. Check the [HTTP Status Code](#) that is returned in the response's HTTP headers for the specific error. The following errors may be returned in response to the Get request:

Error (Status Code)	Meaning	Returned When
400	Bad Request	Request is missing required parameters
500	Internal Server Error	A service error occurred

List Import Formats

Lists the set of supported import formats for annotations.

Calling Method and Arguments

Invoked as an HTTP GET method. Send an HTTP request of the form:

```
GET /annotationservice/format/import HTTP/1.1
```

Parameter	Supported Values
Accept Header	application/xml,text/xml,application/json

Response

On success, a response with a "202 Accepted" HTTP status code will be returned:

Parameter	Meaning
HTTP Body	An instance of an AnnotationFormatListRepresentation schema XML document

Exceptions

If an error occurred, some non-2xx code will be returned. Check the [HTTP Status Code](#) that is returned in the response's HTTP headers for the specific error. The following errors may be returned in response to the Get request:

Error (Status Code)	Meaning	Returned When
400	Bad Request	Request is missing required parameters
500	Internal Server Error	A service error occurred

Format Options

Most requests to retrieve annotations, as documented below, support a `format_option` parameter in which you can supply instructions specific to the requested export format. In general, the valid values for this parameter depends on the 'specialParameters' parameters supported by the particular Pepper Importer/Exporter module for the requested template format. The following lists the the general parameters specific to the Syntactic Annotation Service which apply to all conversions between template formats:

- `pepperImporter.corpusName=<name of the corpus>` (this option specifies the name to be used to represent the corpus being transformed)

Annotation Repository

A pre-configured annotation repository. The 1.1.0 release of the service supports remote url based repositories that follow a template pattern for retrieval of annotations according to an identifier or a specific type of annotation repository. These are the repositories from which the service can retrieve annotations.

List Annotation Repository instances

List the annotation repositories configured for this instance of the service.

Calling Method and Arguments

Invoked as an HTTP GET method. Send an HTTP request of the form:

```
GET /annotationService/repository
```

Parameter	Supported Values
Accept Header	application/xml,text/xml,application/json

Response

On success, a response with a "202 Accepted" HTTP status code will be returned:

Parameter	Meaning
HTTP Body	An instance of an AnnotationRepositoryListRepresentation schema XML document

Exceptions

If an error occurred, some non-2xx code will be returned. Check the [HTTP Status Code](#) that is returned in the response's HTTP headers for the specific error. The following errors may be returned in response to the Get request:

Error (Status Code)	Meaning	Returned When
400	Bad Request	Request is missing required parameters

404	Not Found	The requested document does not exist and /or the requested analysis returned no results
500	Internal Server Error	A service error occurred

List an Annotation Repository

List a specific annotation repository configured for this instance of the service.

Calling Method and Arguments

Invoked as an HTTP GET method. Send an HTTP request of the form:

```
GET /annotationservice/repository/{RepoId}
```

Parameter	Supported Values
Accept Header	application/xml,text/xml,application/json

Example:

```
GET /annotationservice/repository/alpheios HTTP/1.1
```

Response

On success, a response with a "202 Accepted" HTTP status code will be returned:

Parameter	Meaning
HTTP Body	An instance of an AnnotationRepositoryListRepresentation schema XML document

Exceptions

If an error occurred, some non-2xx code will be returned. Check the [HTTP Status Code](#) that is returned in the response's HTTP headers for the specific error. The following errors may be returned in response to the Get request:

Error (Status Code)	Meaning	Returned When
400	Bad Request	Request is missing required parameters
404	Not Found	The requested document does not exist and /or the requested analysis returned no results
500	Internal Server Error	A service error occurred

Text Repository

A repository from which texts/documents can be retrieved and transformed to create templates for annotation. The 1.1.0 version of the services supports both CTS repositories and repositories which can respond to URI based requests.

List Text Repository instances

List the text repositories configured for this instance of the service.

Calling Method and Arguments

Invoked as an HTTP GET method. Send an HTTP request of the form:

```
GET /annotationService/textrepository HTTP/1.1
```

Parameter	Supported Values
Accept Header	application/xml,text/xml,application/json

Response

On success, a response with a "202 Accepted" HTTP status code will be returned:

Parameter	Meaning
HTTP Body	An instance of an TextRepositoryListRepresentation schema XML document

Exceptions

If an error occurred, some non-2xx code will be returned. Check the [HTTP Status Code](#) that is returned in the response's HTTP headers for the specific error. The following errors may be returned in response to the Get request:

Error (Status Code)	Meaning	Returned When
400	Bad Request	Request is missing required parameters
404	Not Found	The requested document does not exist and /or the requested analysis returned no results
500	Internal Server Error	A service error occurred

List a specific Text Repository

List a specific text repository configured for this instance of the service.

Calling Method and Arguments

Invoked as an HTTP GET method. Send an HTTP request of the form:

```
GET /annotationService/textrepository/{RepoId}
```

Parameter	Supported Values
Accept Header	application/xml,text/xml,application/json

Example:

```
GET /annotationService/textrepository/xmlRepository HTTP/1.1
```

Response

On success, a response with a "202 Accepted" HTTP status code will be returned:

Parameter	Meaning
HTTP Body	An instance of an TextRepositoryListRepresentation XML document

Exceptions

If an error occurred, some non-2xx code will be returned. Check the [HTTP Status Code](#) that is returned in the response's HTTP headers for the specific error. The following errors may be returned in response to the Get request:

Error (Status Code)	Meaning	Returned When
400	Bad Request	Request is missing required parameters
404	Not Found	The requested document does not exist and /or the requested analysis returned no results
500	Internal Server Error	A service error occurred

Annotation

An annotation on a document, text or fragment.

Request a specific annotation or set of annotations

Calling Method and Arguments

Invoked as an HTTP GET or HTTP POST. Send an HTTP request of the form:

```
GET /annotationservice/annotation HTTP/1.1
POST /annotationservice/annotation HTTP/1.1
```

parameter	accepted values	cardinality	notes
document_id	cts urn identifying the document, document fragment or token	1	
template_format	code for the requested template format	1	
format_option	format-specific option	0..1	
repos_order	repository search order	0..1	
filter	TBD filter for limiting results to a specific annotation if multiple exist	0..*	
notify	email address or bamboo person id for notification upon completion	0..1	
wait	true or false	0..1	if not supplied, default is true (request will be submitted synchronously)
Accept Header	application/xml,text/xml,application/json		

Example:

The following request asks for all annotations on the document fragment identified as urn:cts:greekLit:tlg0085.tlg003.alpheios-text-grc1:1-100 (lines 1-100 of 'Prometheus Bound', from the 'alpheios' annotation repository):

```

POST /bsp/annotationService/annotation HTTP/1.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Content-Length: 131
template_format=Perseus&lang=grc&document_id=urn%3AActs%3AgreekLit%
3Atlg0085.tlg003.alpheios-text-grc1%3A1-100&repos_order=alpheios

```

Response

Synchronous requests (param wait=true)

A response with "200 (OK)" HTTP status code will be returned.

Parameter	Meaning
HTTP Body	OAC/RDF annotations as annotation body, adhering to the schema identified in the template_format request parameter.

Asynchronous requests (param wait=false)

A response with "202 (Accepted)" HTTP Status code will be returned

Parameter	Meaning
HTTP Body	A single annotation which represents a summary annotation for the request. The rdf:about value for the Annotation itself will be set to a urn representing a unique identifier for the request. The rdf:about value for the Annotation Body will be the url identifying the location the results will be stored at when complete.

Exceptions

If an error occurred, some non-2xx code will be returned. Check the [HTTP Status Code](#) that is returned in the response's HTTP headers for the specific error. The following errors may be returned in response to the request:

Error (Status Code)	Meaning	Returned When
400	Bad Request	Request is missing required parameters
404	Not Found	The requested annotations were not found or could not be retrieved.
500	Internal Server Error	A service error occurred

Parameter	Meaning
HTTP Body	XML document with AnnotationServiceError as root element containing a descriptive message.

Request an annotation or set of annotations matching certain criteria

Calling Method and Arguments

Invoked as an HTTP GET or HTTP POST. Send an HTTP request of the form:

```

GET /annotationService/annotation/match HTTP/1.1
POST /annotationService/annotation/match HTTP/1.1

```

parameter	accepted values	cardinality	notes
document_id	cts urn identifying the document, document fragment or token	0..1	
lang	language of interest	0..1	
template_format	code for the requested template format	1	
format_option	format-specific option	0..1	
repos_order	repository search order	0..1	
filterBy	filter field for match	1	(valid query options for each repository are available in the repository list output)
filterOp	filter operation for match	1	(currently only supported value is 'equals')
filterValue	filter field value match	1	
notify	email address or bamboo person id for notification upon completion	0..1	
wait	true or false	0..1	if not supplied, default is true (request will be submitted synchronously)
Accept Header	application/xml,text/xml,application/json		

Example:

The following request asks for annotations on the document fragment identified as urn:cts:greekLit:tlg0085.tlg003.alpheios-text-grc1:1-100 (lines 1-100 of 'Prometheus Bound', in the 'alpheios' annotation repository, where the word's part of speech matches 'noun':

```
POST /bsp/annotationservice/annotation/match HTTP/1.1
Accept: application/json
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Content-Length: 177
template_format=Perseus&lang=grc&document_id=urn%3Acts%3AgreekLit%3Atlg0085.tlg003.alpheios-text-grc1%3A1-100&repos_order=alpheios&filterOp>equals&filterValue=noun&filterBy=pos
```

Response

Synchronous requests (param wait=true)

A response with "200 (OK)" HTTP status code will be returned.

Parameter	Meaning
HTTP Body	OAC/RDF annotations as annotation body, adhering to the schema identified in the template_format request parameter.

Asynchronous requests (param wait=false)

A response with "202 (Accepted)" HTTP Status code will be returned

Parameter	Meaning
HTTP Body	A single annotation which represents a summary annotation for the request. The rdf:about value for the Annotation itself will be set to a urn representing a unique identifier for the request. The rdf:about value for the Annotation Body will be the url identifying the location the results will be stored at when complete.

Exceptions

If an error occurred, some non-2xx code will be returned. Check the [HTTP Status Code](#) that is returned in the response's HTTP headers for the specific error. The following errors may be returned in response to the request:

Error (Status Code)	Meaning	Returned When
400	Bad Request	Request is missing required parameters
404	Not Found	The requested document was not found or could not be retrieved.
500	Internal Server Error	A service error occurred

Parameter	Meaning
HTTP Body	XML document with AnnotationServiceError as root element containing a descriptive message.

Annotation Template (Document)

An annotation template for a specified document.

Create an annotation template for a specific document.

Calling Method and Arguments

Invoked as an HTTP GET or HTTP POST. Send an HTTP request of the form:

```
GET /annotationservice/annotation/template/document HTTP/1.1
POST /annotationservice/annotation/template/document HTTP/1.1
```

parameter	accepted values	cardinality	notes
document_id	document (or document fragment) id (cts urn, cmis object id)	1	
repos_type	repository type (e.g. cts, xml)	0..1	
repos_uri	base uri for the repository	0..1	
lang	languages of interest in the document	1..*	
template_format	code for the requested template format	1	
format_option	format-specific option	0..1	
notify	email address or bamboo person id for notification upon completion	0..1	
wait	true or false	0..1	if not supplied, default is true (request will be submitted synchronously)
Accept Header	application/xml,text/xml,application/json		

Example:

```
POST /bsp/annotationService/annotation/template/document HTTP/1.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Content-Length: 164
template_format=Perseus&lang=lat&document_id=urn%3AActs%3AlatinLit%
3Asto0179.stoa001.stoa001%3A1.1.1&repos_type=cts&repos_uri=http%3A%2F%
2Fhclatin.appspot.com%2FCTS
```

Response

Synchronous requests (param wait=true)

A response with "200 (OK)" HTTP status code will be returned.

Parameter	Meaning
HTTP Body	One or more OAC/RDF Annotation templates for requested text

Asynchronous requests (param wait=false)

A response with "202 (Accepted)" HTTP Status code will be returned

Parameter	Meaning
HTTP Body	A single annotation which represents a summary annotation for the request. The rdf:about value for the Annotation itself will be set to a urn representing a unique identifier for the request. The rdf:about value for the Annotation Body will be the url identifying the location the results will be stored at when complete.

Exceptions

If an error occurred, some non-2xx code will be returned. Check the [HTTP Status Code](#) that is returned in the response's HTTP headers for the specific error. The following errors may be returned in response to the request:

Error (Status Code)	Meaning	Returned When
400	Bad Request	Request is missing required parameters
404	Not Found	The requested document was not found or could not be retrieved.
500	Internal Server Error	A service error occurred

Parameter	Meaning
HTTP Body	XML document with AnnotationServiceError as root element containing a descriptive message.

Annotation Template (Text)

An annotation template for supplied text.

Create an annotation template for supplied text

Calling Method and Arguments

Invoked as an HTTP GET or HTTP POST. Send an HTTP request of the form:

```
GET /annotation-service/annotation/template/text HTTP/1.1
POST /annotation-service/annotation/template/text HTTP/1.1
```

parameter	accepted values	cardinality	notes
text	the supplied text	0..1	
text_uri	a uri for the supplied text	0..1	If both text and text_uri are supplied, the uri will be used as the identifier for the text only
template_format	code for the requested template format	1	
mime_type	mime-type of the supplied text (text/html, text/xml, or text/plain)	1	
format_option	format-specific option	0..1	
lang	languages of interest in the supplied text	1..*	
notify	email address or bamboo person id for notification upon completion	0..1	
wait	true or false	0..1	if not supplied, default is true (request will be submitted synchronously)
Accept Header	application/xml,text/xml,application/json		

Example:

```
POST /bsp/annotation-service/annotation/template/text HTTP/1.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Content-Length: 79
text=veni%20vidi%20vici&template_format=Perseus&lang=lat&mime_type=text%2Fplain
```

Response

Synchronous requests (param wait=true)

A response with "200 (OK)" HTTP status code will be returned.

Parameter	Meaning
HTTP Body	One or more OAC/RDF Annotation templates for requested text

Asynchronous requests (param wait=false)

A response with "202 (Accepted)" HTTP Status code will be returned

Parameter	Meaning
HTTP Body	A single annotation which represents a summary annotation for the request. The rdf:about value for the Annotation itself will be set to a urn representing a unique identifier for the request. The rdf:about value for the Annotation Body will be the url identifying the location the results will be stored at when complete.

Exceptions

If an error occurred, some non-2xx code will be returned. Check the [HTTP Status Code](#) that is returned in the response's HTTP headers for the specific error. The following errors may be returned in response to the request:

Error (Status Code)	Meaning	Returned When
400	Bad Request	Request is missing required parameters
404	Not Found	The requested document was not found or could not be retrieved.
500	Internal Server Error	A service error occurred

Parameter	Meaning
HTTP Body	XML document with AnnotationServiceError as root element containing a descriptive message.

SOA Layer API

See Javadoc.

Notifications

If the notify parameter has been supplied and set to an email address an email will be sent to that address upon completion of the request. If the notify parameter is set to a bamboo person id, notifications will be sent according to user preferences. The body of the notification messages will contain the unique identifier for the request (as returned in the rdf:about value for the annotation returned by the requested), the current status, and the location at which the results can be retrieved.

Tests

`${REPOSITORY_ROOT}` is <http://svn.code.sf.net/p/projectbamboo/code/>

See tests at `${REPOSITORY_ROOT}/platform-services/annotation-service/trunk/annotation-service/resource/src/test/tests.html`