

Pro Bono Net Web Services

Version 1.120080624

OVERVIEW

The Pro Bono Net (PBN) Advocate Web Service is a Representational State Transfer (REST) based application programmable interface (API) enabling computer to computer information exchange and communication. It is designed to empower external applications to access and interact with content within the PBN data store.

The PBN Advocate API is designed with security, extensibility, stability, and scalability as key prerequisites. Security is applied through user credentials and encrypted transport. Extensibility is provided by creating a framework for PBN to continually add new Services and Resources. Stability is addressed through the correct handling of HTTP status codes, and parameterized Resources allow for the service to scale from simple Resource access to full system integrations.

The API supports a number of XML formats such as Atom 1.0, and Legal Services XML. By default the Service returns all responses in Atom 1.0 XML documents. Alternate formats can be setup on a per user basis.

DEFINITIONS

- **Service**
The name applied to the PBN API application.
- **Client**
Refers to the person or computer accessing the service.
- **Resource**
Represents an end point or URI of the service with or without out parameters. Each Resource URI will return data based on parameters passed in from the Client to the Resource.
- **Parameters**
Optional arguments passed to the resource that allow for the client to get a focused resource result.
- **Credentials**
The username and password issued to the Client from the Service Administrator
- **Method**
The operation to perform on the Resource specified in the HTTP request.
- **Request**
The action of the Client to ask the service to take action on a Resource
- **Response**
The message sent to the Client from the Service as a result of the Request
- **Client Errors**
Requests to the Service resulting in a 4XX status code.
- **Service Errors**
Requests to the Service resulting in a 5XX status code.

- **Service Administrator**
Contact at Pro Bono Net to contact regarding issues around the usage of the Service. Brett Suwyn can be contacted at bsuwyn@probono.net

SECURITY

Basic HTTP AUTH is required when accessing restricted API services and views. These specific services and views are outlined below.

Since Basic HTTP Auth sends the credentials in plain text the access to the feed can be considered vulnerable to network eavesdroppers. To prevent this, communication may be routed over a secure sockets layer (SSL), effectively encrypting the traffic between the parties and providing end to end encryption and security.

REQUESTS

The PBN API can be accessed over the Internet at <http://api.probono.net>

Communication Process

1. Client issues a resource request (URI) passing along any credentials in the HTTP header. The credentials may or not be required to access the resource but should always be sent to the Service.
2. Service responds to the resource requested, providing the client with status information and response data
3. Client examines the HTTP header and determines the status of the request
 - If the Client find a status code of 200 OK it assumes that the request was valid and that the response represents the requested information.
 - If the Client finds a status code of 4XX (client errors) the client must remedy the issue before requesting the resource again. Excessive client errors will lead the Service placing a ban on service access for the client.
 - If a status code of 5XX is encountered the Client should log the response data along with the date and time of the Request. The Client should then wait at least 10 minutes before making another request to the resource to allow for the API to recover from the error.
4. Resource request has completed

Abuse

The Service may deem that the Client is abusive and place a temporary or permanent ban on the Client. A temporary ban will restrict the Client from accessing any resource for a period of time. If a permanent ban is placed on the Client, the Service Administrator must be contacted to resolve the issue.

To avoid a Client Ban the following rules should be respected

- The Client must not make more than one request a second to the Service
- The Client must take action to correct any Client Errors

- The Client must respect Service Errors and provide the Service with enough time between Resource requests to recover.

Status Codes

HTTP status code are used to report the context of the Response from the Service.

Status Code	Status Text	Reason	Client Next Step
200	OK	The request was processed as expected without error	Process response data
401	Unauthorized	The Resource has restricted access. Credentials should be sent with the request to access the resource.	Submit credentials along with Resource request.
403	Forbidden	The client does not have access to the requested resource. The client either does not have the correct rights over the resource or has been banned from accessing the service.	No action. Service administrator should be contacted for resolution.
404	Not Found	The requested Resource does not exist.	No action
405	Method Not Allowed	The requested HTTP method is not allowed for the specific resource	Inspect response body to determine allowable methods for the Resource
400	Bad Request	The resource requested contained a parameter or data that did not conform the expected type. E.g. A date parameter was invalid. More information can be found by examining the response data	Inspect the Response data for the reason of failure. The Parameter will need to be corrected to access the Resource
500	Internal Server Error	An unexpected error occurred while processing the resource request. The response document should be check for more information.	Pause; wait at least 10 minutes for the Service to recover from the error. If the Service does not recover contact the Service Administrator

Service Entity Pattern

There is a distinct pattern to accessing the Resources of the Service:

```
http://{domain}/{service}/{resource}?[ parameters]
[parameters] define optional arguments as defined by the method called on the service.
```

For example to retrieve all calendar events that have been updated between Nov 8, 2005 @ 1:13 pm and Jan 1, 2006 a Request to the Event Service Updates Resource would be made with parameters:

```
http://api.probono.net/events.updated?dtspan=2005-11-08T13:15z.2006-01-01T00:00z
```

The parameter here, dtspan, defines a date and time span in which to return updated events. dtspan is the specified in the first position in ordered parameters

Parameters

Parameters maybe appended to Resource requests to allow for the client to modify the data returned in the Response. This may include pagination, date constraints, record limits, etc.

Name	Data Type	Constraints	Example	Comments
dtspan	Date/time	ISO 8601	2002-03-01T13:00:00Z/2003-05-11T15:30:00Z	If a single date/time is provided, the Service will assume the span includes all resources from the date/time until the current time.
limit	Integer	100 max	20	Specifies the number or items to return in Resources that return multiple items. Defaults to 25.
page	Integer		1	Specifies the page to return allowing the Client to use in coordination with limit to paginate a large number or item listings.
listing	text	One of the following: <ul style="list-style-type: none">• list• summary• detail	summary	Specifies the level of detail to return for Resources. List will return a minimal amount of information, Summary will include a bit more metadata, and detail will include the full metadata for a Resource

XML FEED WRAPPER

A common XML wrapper is applied to all XML documents returned by the Service. This XML wrapper contains metadata information that describes the feed itself, such as total records, pagination, feed last update date, and URIs to the feed requested and the next feed in the pagination. There will only ever be one feed element per request.

The OS namespace references Amazon's Open Search specification, details can be found here:
http://www.opensearch.org/Specifications/OpenSearch/1.1#OpenSearch_response_elements

The frequency column specifies how many elements may be specified in the field.

Element	Frequency	Comments
title	1	Title of the feed requested
updated	1	Date/Time the feed was last updated. ISO 8601 format
os:totalResults	0-1	The total number of results matching the request
os:startIndex	0-1	The index of the first item in the feed
os:itemsPerPage	0-1	Items per page. Number of items represented in the current feed
link@rel=self	1	URL to the feed being requested
link@rel=next	0-1	URL to the next page of the feed if the total number of results is larger than the index of the last item in the feed.
link@rel=prev	0-1	URL to the previous page of the feed if it exists.

ATTACHMENTS

Some content items returned by the API have enclosures or attachments associated with them. These are defined by the link@rel="enclosure" element or can be referenced by URI in the Service specific format used. The enclosures then may be retrieved from the Attachment service outlined below.

Summary of Resources Available at the Attachments Service

- <http://api.probono.net/events/{id}.{filename}>

<http://api.probono.net/attachments/{id}.{filename}>

Accesses the attachment defined the ID in the Resource Locator

Methods Supported

- GET

Content Types

The following content types may be returned by the attachment service

- text/html
- text/plain
- application/pdf
- application/rtf
- application/msword
- application/vnd.ms-excel
- application/mspowerpoint
- application/wordperfect5.1
- image/gif
- image/jpeg
- image/tiff
- image/x-png
- application/x-mspublisher

Parameters

Name	Default	Comments

Examples

Returns a document attachment with an id of 43128

```
http://api.probono.net/attachments/43128.Legal+Advice.pdf
```

References

- HTTP Basic Auth
<http://tools.ietf.org/html/rfc2617>
- HTTP Status Codes
http://en.wikipedia.org/wiki/List_of_HTTP_status_codes
- HTTP Methods
<http://www.w3.org/Protocols/rfc2616/rfc2616-sec9.html>
- ISO 8601 (date time format)
http://en.wikipedia.org/wiki/ISO_8601
- Uniform Resource Names
http://en.wikipedia.org/wiki/Uniform_Resource_Name
- Atom Syndication Format (rfc 7287)
<http://atompub.org/rfc4287.html>
- Atom Syndication Format
<http://www-128.ibm.com/developerworks/xml/library/x-atom10.html>
- Open Search Specification
<http://www.opensearch.org/Specifications/OpenSearch/1.1>
- REST wiki
<http://rest.blueoxen.net/>
- REST for the rest of us
http://doc.opengarden.org/Articles/REST_for_the_Rest_of_Us
- Logical references to target for rel/rev
http://www.topxml.com/xhtml/xhtml_tag_link.asp