Alexa Site Thumbnail

Developer Guide

Alexa Site Thumbnail: Developer Guide

Published 2006-August-02

Table of Contents

Introduction

Welcome! Welcome to the official guide for the Alexa Site Thumbnail (AST) service. The Alexa Site Thumbnail service provides users with thumbnail pictures of millions of websites. This guide explains how to get your software to work with the web service API.

Usage. There are two ways to get thumbnails onto your webpage.

- **By sample code.** If you are using Java/JSP, C#/ASP, Perl, or PHP, you can simply use the sample code provided in the Resource Center. See the Getting Started Guide for more information about using the sample code.
- **By web service API.** If you are a developer and you need more flexibility, you can write your own code to interface with the web service API. The AST web service API accepts SOAP or QUERY requests.The rest of this guide explains how to use the AST web service API.

Tip

Most users do not need to learn the web service API. The Getting Started Guide will most likely explain everything you need to know to use the AST service.

ReferenceLinks.

- Resource Center

  Code samples, release notes, RSS feeds and more
- AlexaSiteThumbnail WSDL

  WSDL file for the AlexaSiteThumbnail SOAP API
- Developer Forums

  Questions and suggestions go here
- AlexaSiteThumbnail Homepage

AST Details

About Requests

Making Requests. Requests for thumbnails are made by making service calls to the Alexa Site Thumbnail web service. See the API Reference for more information on making requests. In order to use the Alexa Site Thumbnail web service, you must first sign up for an Amazon Web Services account, and then subscribe to the Alexa Site Thumbnail service. See the following links to sign up:

- Sign up for an Amazon Web Services account
- Subscribe to the Alexa Site Thumbnail service

About Thumbnails

Sample thumbnails. Thumbnail images are available in small or large sizes. Shown below are two thumbnails for the site `www.alexa.com` .

Small thumbnail	Large thumbnail
 www.alexa.com	 www.alexa.com

Comparison . The following table compares some properties of small and large thumbnails

Small thumbnail	Large thumbnail
111x82 pixels	201x147 pixels
jpeg format only	jpeg format only
about 3kb in size	about 8kb in size

No thumbnail available . The default behavior of the web service API when the requested thumbnail does not exist is to return the following set of images. When this image is returned, the request website will be put on queue for thumbnail generation. The correct thumbnail will appear once the thumbnail has been generated internally by Alexa.

Small thumbnail	Large thumbnail
 no image (gif format)	

Small thumbnail	Large thumbnail
	no image (gif format)

Note

AST automatically generates both small and large thumbnails when you request a thumbnail that does not exist.

Alexa Site Thumbnail webservice API Reference

Making requests. There are two types of requests that can be used with the web service; QUERY and SOAP.

- QUERY requests are HTTP GET requests where parameters are passed as HTTP parameters
- SOAP requests pass parameters in a SOAP envelope

The Making QUERY Requests and Making SOAP Requests sections describe how to create QUERY and SOAP requests.

> **Note**
> It is worth noting that QUERY requests are generally easier to test and implement from scratch. SOAP requests are generally handled by libraries that can consume WSDL files; however depending on the SOAP library used your mileage may vary.

Actions . All supported actions are documented in the Actions section.

Authentication . All requests must include a *`Signature`* parameter that is computed using your AWS secret key. This authentication mechanism guarantees that the request was created by you, because only you know the secret key. The Request Authentication section describes how to create a signature.

Response structure. The response data from either requests have the same structure (they share the same xml schema). For QUERY requests the response data is in the response HTTP body, and for SOAP requests the response data is in the response SOAP-ENV:Body element. Go to the Response Schema section for the response schema

Making QUERY Requests to Alexa Site Thumbnail

Some facts about QUERY requests:

- QUERY requests are simply HTTP GET requests (RFC2616).
- Service parameters are passed as simple HTTP parameters.
- QUERY requests can be tested on any browser.
- The response is in the HTTP response body and is in human readible XML format.
- QUERY requests support batching (sending more than one request at once).

QUERY Request Format

The Base URL. Every QUERY request to Alexa Site Thumbnail web service begins with the following:

Base URL

```
http://ast.amazonaws.com/Xino
```

Request Parameters. The base URL is followed by a series of parameters that define the request. Parameters are separated from the base URL by a question mark (**?**) and each other by an ampersand (**&**) character. Each parameter consists of a key and a value, separated from each other by an equals sign (**=**). Note that parameters and their values are case-sensitive; for example, **`Action=Thumbnail`** works correctly, but **`Action=thumbnail`** produces an error. Also note that all HTTP parameters must be UrlEncoded. The following example shows a simple QUERY request that returns a thumbnail for a given URL.

Sample Request

```
http://ast.amazonaws.com/Xino?

                    AWSAccessKeyId=
                    [your AWSAccessKeyId]
                    &Timestamp=
                    [timestamp now, format as described in ht-
tp://www.w3.org/TR/xmlschema-2/#dateTime]
                    &Signature=
                    [signature calculated from request]
                    &Action=
                    [desired action, such as Thumbnail]
                    &Url=
                    [URL of website to get thumbnail for]
```

Parameter Details. The parameters in the example are described in the following table:

QUERY Batch Request Format

Batch Requests. Batch requests are very similar to regular requests, except that it allows multiple service calls to be sent in one request. Batch requests save on HTTP turnaround time compared to making multiple individual service invocations.

Tip

For web pages with multiple thumbnail images on the same page, reduce server-side latency by aggregating multiple thumbnail requests into one batch request.

All parameters except the core AWS parameters (`AWSAccessKeyId`, `Timestamp`, `Signature`, `Action`) are batched according to the following rules.

- If a query request contains at least one parameter of the format `[ActionName].[integer].[ParameterName]`, it is a batched request. (example: `Thumbnail.1.Url`).
- The numbers for the request params must be contiguous (i.e. if `Thumbnail.3.Url` is present, `Thumbnail.1.Url` and `Thumbnail.2.Url` are required as well).
- To save on typing, batched requests are allowed to have shared parameters. These are the parameters that have the same value for all the batch sub-requests. Shared parameters have the form `[ActionName].Shared.[ParameterName]` (example: `Thumbnail.Shared.Size`). Short-hand form of `Shared.[ParameterName]` is supported as well.
- Up to 30 service calls can be batched at a time in a single request. If more service calls are needed, they must be divided into multiple batch requests.
- All service calls within a single batch request must be of the same `Action` type. You cannot mix different actions in a single batch request.

Sample Batch Request #1. The following example shows a batch QUERY request that returns three thumbnails for the given three URLs and sizes.

Sample Request

```
http://ast.amazonaws.com/Xino?
                    AWSAccessKeyId=
                    [your AWSAccessKeyId]
                    &Timestamp=
                    [timestamp now, format as described in ht-
tp://www.w3.org/TR/xmlschema-2/#dateTime]
                    &Signature=
                    [signature calculated from request]
                    &Action=
                    Thumbnail
                    &Thumbnail.1.Url=
                    [website #1]
                    &Thumbnail.2.Url=
                    [website #2]
                    &Thumbnail.3.Url=
                    [website #3]
                    &Thumbnail.1.Size=
                    Large
                    &Thumbnail.2.Size=
                    Small
                    &Thumbnail.3.Size=
                    Large
```

Sample Batch Request #2. The following example shows a batch QUERY request that returns three *large* thumbnails for the given three URLs.

Sample Request

```
http://ast.amazonaws.com/Xino?

                    AWSAccessKeyId=
                    [your AWSAccessKeyId]
                    &Timestamp=
                    [timestamp now, format as described in ht-
tp://www.w3.org/TR/xmlschema-2/#dateTime]
                    &Signature=
                    [signature calculated from request]
                    &Action=
                    Thumbnail
                    &Thumbnail.1.Url=
                    [website #1]
                    &Thumbnail.2.Url=
                    [website #2]
                    &Thumbnail.3.Url=
                    [website #3]
                    &Thumbnail.Shared.Size=
                    Large
```

QUERY Response Format

This section shows sample response data for both nonbatch and batch requests.

Sample Responses.

Sample response for a request with no batching

The request

```
http://ast.amazonaws.com/Xino?AWSAccessKeyId=1234567890ABCDEFGHIJ
                             &Signature=1234567890abcdefghijABCDEFGH
                             &Timestamp=2006-01-01T00:00:00.000Z
                             &Action=Thumbnail
                             &Url=alexa.com
```

corresponding response:

```
<?xml version="1.0"?>
<aws:ThumbnailResponse
xmlns:aws="http://ast.amazonaws.com/doc/2005-10-05/">
        <aws:Response>
                <aws:OperationRequest>
                        <aws:RequestId>
                                xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
                        </aws:RequestId>
                </aws:OperationRequest>
                <aws:ThumbnailResult>
                        <aws:Thumbnail Exists="true">
                                http://location_to_thumbnail_for_alexa.com
                        </aws:Thumbnail>
                        <aws:RequestUrl>
                                alexa.com
                        </aws:RequestUrl>
                </aws:ThumbnailResult>
                <aws:ResponseStatus>
                        <aws:StatusCode>
                                Success
                        </aws:StatusCode>
```

```
        </aws:ResponseStatus>
    </aws:Response>
</aws:ThumbnailResponse>
```

Sample response for a request with batching

The request

```
http://ast.amazonaws.com/Xino?AWSAccessKeyId=1234567890ABCDEFGHIJ
                             &Signature=1234567890abcdefghijABCDEFGH
                             &Timestamp=2006-01-01T00:00:00.000Z
                             &Action=Thumbnail
                             &Thumbnail.1.Url=alexa.com
                             &Thumbnail.2.Url=amazon.com
                             &Thumbnail.3.Url=a9.com
                             &Thumbnail.Shared.Size=Large
```

corresponding response:

```
<?xml version="1.0"?>
<aws:ThumbnailResponse
xmlns:aws="http://ast.amazonaws.com/doc/2005-10-05/">
        <aws:Response>
                <aws:OperationRequest>
                        <aws:RequestId>
                                xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
                        </aws:RequestId>
                </aws:OperationRequest>
                <aws:ThumbnailResult>
                        <aws:Thumbnail Exists="true">
                                http://location_to_thumbnail_for_alexa.com
                        </aws:Thumbnail>
                        <aws:RequestUrl>
                                alexa.com
                        </aws:RequestUrl>
                </aws:ThumbnailResult>
                <aws:ResponseStatus>
                        <aws:StatusCode>
                                Success
                        </aws:StatusCode>
                </aws:ResponseStatus>
        </aws:Response>
        <aws:Response>
                <aws:OperationRequest>
                        <aws:RequestId>
                                xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
                        </aws:RequestId>
                </aws:OperationRequest>
                <aws:ThumbnailResult>
                        <aws:Thumbnail Exists="true">
                                http://location_to_thumbnail_for_amazon.com
                        </aws:Thumbnail>
                        <aws:RequestUrl>
                                amazon.com
                        </aws:RequestUrl>
                </aws:ThumbnailResult>
                <aws:ResponseStatus>
                        <aws:StatusCode>
                                Success
                        </aws:StatusCode>
                </aws:ResponseStatus>
```

```
        </aws:Response>
        <aws:Response>
                <aws:OperationRequest>
                        <aws:RequestId>
                                xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
                        </aws:RequestId>
                </aws:OperationRequest>
                <aws:ThumbnailResult>
                        <aws:Thumbnail Exists="true">
                                http://location_to_thumbnail_for_a9.com
                        </aws:Thumbnail>
                        <aws:RequestUrl>
                                a9.com
                        </aws:RequestUrl>
                </aws:ThumbnailResult>
                <aws:ResponseStatus>
                        <aws:StatusCode>
                                Success
                        </aws:StatusCode>
                </aws:ResponseStatus>
        </aws:Response>
</aws:ThumbnailResponse>
```

Making SOAP Requests to Alexa Site Thumbnail

You can interact with the web service using SOAP 1.1 over HTTP. The WSDL file, which describes the API in a machine-readable way, is available at: http://ast.amazonaws.com/doc/2006-05-15/AlexaSiteThumbnail.wsdl . The AST response schema is available at: http://ast.amazonaws.com/doc/2006-05-15/AlexaSiteThumbnail.xsd.

Most SOAP users will interact with AST using a SOAP toolkit tailored for their language and development environment. Different toolkits will expose the API in different ways. Please refer to your specific toolkit documentation to understand how to use it. This section illustrates SOAP operations in a toolkit-independent way by exhibiting the XML requests and responses as they appear "on the wire."

WSDL Location and SOAP End Point

Your SOAP toolkit/library should be capable of creating SOAP requests based on the WSDL file. Here are the URLs for the WSDL file, schema file, and SOAP end point that your toolkit/library will need to know in order to consume the web service.

WSDL Location.

```
http://ast.amazonaws.com/doc/2006-05-15/AlexaSiteThumbnail.wsdl
```

Schema Location.

```
http://ast.amazonaws.com/doc/2006-05-15/AlexaSiteThumbnail.xsd
```

The SOAP End Point.

```
http://ast.amazonaws.com/Xino
```

Soap Request Format

Your soap toolkit is responsible for formatting and sending requests based on the provided WSDL. This section shows what SOAP requests look like "on the wire".

Sample Non-batch Request. This is a sample non-batch SOAP request. Notice that the parameters `Signature`, `Timestamp`, and `AWSAccessKeyId` are identical to those in QUERY requests. The `Action` parameter however is specified in the element name in the form of `<[Action]Request>`.

```
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
```

```
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
 <SOAP-ENV:Body>
  <m:ThumbnailRequest
xmlns:m="http://ast.amazonaws.com/doc/2006-05-15/">
   <m:Request>
    <m:Signature>CekuNgi1Tg4AoXNgnQ6Ic4rAkPI=</m:Signature>
    <m:Timestamp>2006-01-01T00:00:00.000Z</m:Timestamp>
    <m:AWSAccessKeyId>1234567890ABCDFGHIJ</m:AWSAccessKeyId>
    <m:Url>www.alexa.com</m:Url>
   </m:Request>
  </m:ThumbnailRequest>
 </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Sample Batch Request. This is a sample batch SOAP request with three service calls. Also, note that the *Signature*, *Timestamp*, and *AWSAccessKeyId* parameters need only be in the first `<Request` element.

```
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
 <SOAP-ENV:Body>
  <m:ThumbnailRequest
xmlns:m="http://ast.amazonaws.com/doc/2006-05-15/">
   <m:Request>
    <m:Signature>CekuNgi1Tg4AoXNgnQ6Ic4rAkPI=</m:Signature>
    <m:Timestamp>2006-01-01T00:00:00.000Z</m:Timestamp>
    <m:AWSAccessKeyId>1234567890ABCDFGHIJ</m:AWSAccessKeyId>
    <m:Url>www.alexa.com</m:Url>
   </m:Request>
   <m:Request>
    <m:Url>www.amazon.com</m:Url>
   </m:Request>
   <m:Request>
    <m:Url>www.a9.com</m:Url>
   </m:Request>
  </m:ThumbnailRequest>
 </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Parameter Details. The parameters in the example are described in the following table:

Soap Response Format

The SOAP response from the AST service has a format specified by the schema file embedded in the WSDL file. As in SOAP requests, your SOAP toolkit should be able to consume the response data and present it programmatically to you.

> **Note**
> The text in red is meant to indicate portions of the response that can change depending on the request.

Sample Non-batch Response. This is a sample non-batch SOAP response.

```
<?xml version="1.0"?>
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
 <SOAP-ENV:Body>
  <m:ThumbnailResponse
xmlns:m="http://ast.amazonaws.com/doc/2006-05-15/">
   <aws:Response
xmlns:aws="http://ast.amazonaws.com/doc/2006-05-15/">
    <aws:OperationRequest>
     <aws:RequestId>3f8ceabd-2d15-47f0-b35e-d52ee868a4a6</aws:RequestId>
    </aws:OperationRequest>
    <aws:ThumbnailResult>
     <aws:Thumbnail
Exists="true">http://location_to_thumbnail_for_alexa.com</aws:Thumbnail>
     <aws:RequestUrl>www.alexa.com</aws:RequestUrl>
    </aws:ThumbnailResult>
    <m:ResponseStatus>
     <m:StatusCode>Success</m:StatusCode>
    </m:ResponseStatus>
   </aws:Response>
  </m:ThumbnailResponse>
 </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Sample Batch Response. This is a sample batch SOAP response.

```
<?xml version="1.0"?>
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
 <SOAP-ENV:Body>
  <m:ThumbnailResponse
xmlns:m="http://ast.amazonaws.com/doc/2006-05-15/">
   <aws:Response
xmlns:aws="http://ast.amazonaws.com/doc/2006-05-15/">
    <aws:OperationRequest>
     <aws:RequestId>3f8ceabd-2d15-47f0-b35e-d52ee868a4a6</aws:RequestId>
    </aws:OperationRequest>
    <aws:ThumbnailResult>
     <aws:Thumbnail
Exists="true">http://location_to_thumbnail_for_www.alexa.com</aws:Thumbnail>
     <aws:RequestUrl>www.alexa.com</aws:RequestUrl>
    </aws:ThumbnailResult>
    <m:ResponseStatus>
     <m:StatusCode>Success</m:StatusCode>
    </m:ResponseStatus>
   </aws:Response>
   <aws:Response
xmlns:aws="http://ast.amazonaws.com/doc/2006-05-15/">
    <aws:OperationRequest>
     <aws:RequestId>3f8ceabd-2d15-47f0-b35e-d52ee868a4a6</aws:RequestId>
    </aws:OperationRequest>
    <aws:ThumbnailResult>
     <aws:Thumbnail
Exists="true">http://location_to_thumbnail_for_www.amazon.com</aws:Thumbnail>
     <aws:RequestUrl>www.amazon.com</aws:RequestUrl>
    </aws:ThumbnailResult>
    <m:ResponseStatus>
     <m:StatusCode>Success</m:StatusCode>
    </m:ResponseStatus>
   </aws:Response>
   <aws:Response
xmlns:aws="http://ast.amazonaws.com/doc/2006-05-15/">
    <aws:OperationRequest>
```

```
     <aws:RequestId>3f8ceabd-2d15-47f0-b35e-d52ee868a4a6</aws:RequestId>
    </aws:OperationRequest>
    <aws:ThumbnailResult>
     <aws:Thumbnail
Exists="true">http://location_to_thumbnail_for_www.a9.com</aws:Thumbnail>
     <aws:RequestUrl>www.a9.com</aws:RequestUrl>
    </aws:ThumbnailResult>
    <m:ResponseStatus>
     <m:StatusCode>Success</m:StatusCode>
    </m:ResponseStatus>
   </aws:Response>
  </m:ThumbnailResponse>
 </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SOAP Error Response

In SOAP, an error result is returned to the client as a SOAP fault, with the HTTP response code 500. If you do not receive a SOAP fault, then your request was successful. The SOAP fault code is comprised of a standard SOAP 1.1 fault code (either "Server" or "Client") concatenated with an AST specific error code. For example: "Server.InternalError" or "Client.RequestExpired". The SOAP fault string element contains a generic, human readable error message in English. Finally, the SOAP fault detail element contains miscellaneous information relevant to the error.

For example, if your SOAP request includes a Timestamp parameter that is more than 15 minutes away from the clock on AST servers, the body of the SOAP response contains a "RequestExpired" SOAP fault, which looks like:

```
<?xml version="1.0"?>
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:aws="http://webservices.amazon.com/AWSFault/2005-15-09">
 <SOAP-ENV:Body>
  <SOAP-ENV:Fault>
   <faultcode>aws:Client.RequestExpired</faultcode>
   <faultstring>Request has expired. Timestamp date is
2000-01-01T00:00:00.000Z</faultstring>
   <detail>
    <aws:RequestId
xm-
lns:aws="http://webservices.amazon.com/AWSFault/2005-15-09">21476789-748d-412
5-8a4a-f5addbdb0866</aws:RequestId>
   </detail>
  </SOAP-ENV:Fault>
 </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Request Authentication

Every request must contain authentication information to establish the identify of the principal making the request. In SOAP, the authentication information is put into elements in the <Request> element, and in QUERY, the authentication information is put into HTTP parameters. The parameters needed for authentication are in the following table.

Warning
Don't forget to UrlEncode the timestamp parameter for QUERY requests.

Actions

Definition. The *Action* parameter tells the AST service what operation to perform. In QUERY requests this parameter is specified like any other HTTP parameter. In SOAP requests the name of the first child element of the SOAP body determines the action. Currently the only supported action is Thumbnail. This action returns a URL to the desired thumbnail.

Thumbnail Action

Description. The `Thumbnail` Action tells the Alexa Site Thumbnail web service to return a URL to a thumbnail image for a specified web page. The response body includes a <Thumbnail> tag whose value is a signed and expirable URL to the desired thumbnail. After the request is processed the final consumer (e.g. client browser) has at least 10 seconds to fetch the thumbnail image. After this window the URL is expired and will return a non-200 response code with an error message.

If the image does not exist in the Alexa thumbnail store, a default image with the words "image coming soon" and the Alexa logo of the specified size will be returned. In this case the user does not get charged. Users can tell whether the image URL in the response document is the desired thumbnail or a default "no image" image by checking the value of the "Exists" attribute of the <Thumbnail> tag. "Exists" is set to "true" only if the returned image URL is a real thumbnail. Otherwise "Exists" is set to "false". Please refer to the Response Schema section for a more detailed description of the response object.

Sample Requests.

Using the Thumbnail Action - single request

The following `Thumbnail` action example demonstrates how to make a QUERY request.

```
http://awis.amazonaws.com/xino/xml?Action=Thumbnail
                    &AWSAccessKeyId=
                    [Your AWS Access Key ID]
                    &Signature=
                    [signature]
                    &Timestamp=
                    [timestamp used in signature]
                    &Url=
                    [Valid URL]
                    &Size=
                    [Large or Small]
```

Using the Thumbnail Action - batch request

The following `Thumbnail` action example demonstrates how to make a QUERY request in batch mode. Batch mode can support up to 20 requests at a time.

```
http://awis.amazonaws.com/xino/xml?Action=Thumbnail
                    &AWSAccessKeyId=
                    [Your AWS Access Key ID]
                    &Signature=
                    [signature]
                    &Timestamp=
                    [timestamp used in signature]
                    &Shared.Size=
                    [Large or Small for all thumbnails]
```

```
&Thumbnail.1.Url=
[Valid URL #1]
&Thumbnail.2.Url=
[Valid URL #2]
&Thumbnail.3.Url=
[Valid URL #3]
...
```

For more information on making signed requests, see Signing Requests .

Request Parameters. The Thumbnail Action takes the following parameters. Required parameters must be provided for the request to succeed.

Note
Parameter names and values are case sensitive!

Name	Required	Description	
Action	yes	Use the *Action* parameter to specify the name of the action you would like to perform.	
Url	yes	*Url* is required in all *Thumbnail* action requests to the Alexa Site Thumbnail web service. The *Url* parameter tells the service what web site to retrieve a thumbnail for. *Url* may include the "http://" protocol prefix, subdomain, port, path, and URL parameters. The path and parameter parts of *Url* only matter for certain web sites such as blog sites or portals. Do not forget to UrlEncode the *Url* parameter for QUERY requests.	
Size	no	Size of the returned thumbnail (Small = 111x82 pixels, Large = 201x147 pixels). If this parameter is not specified it defaults to "Large". See AST Details for more information on AST thumbnails.	

Response Schema

The response data for both SOAP and QUERY requests conform the the following response schema file for all actions:

Schema Location.

```
http://ast.amazonaws.com/doc/2006-05-15/AlexaSiteThumbnail.xsd
```

For sample response data, see QUERY Response Format and SOAP Response Format.