

CIQ Technology Services Specifications

Document Details

Version: 1.0

Date Created: 6.16.2008

Last Updated: 11.20.2008

| | |
|--------------------------|----------------------------|
| Business Owner: | Jay Zachter, Michael Yusko |
| Technology Owner: | William Murphy (CIQ), |
| Business Analysts | Shawn West |
| Version | 1.0 |

CIQ Technology Services Specifications: Documents Details

Application Framework

The primary technology for this solution is XML Web Services (SOAP). Capital IQ hosts an API that responds to XML requests according to this API, and returns XML structured data in response. These XML requests are encrypted via the standard HTTPS protocol.

A secondary technology for this solution is the integration of CIQ DataFeeds on client database tier. This allows for reduced network traffic for common items that change infrequently.

Capital IQ hosts this data on Windows-based servers, powered by Microsoft SQL Server in an active-passive failover cluster configuration. Data is stored in multiple fully redundant EMC Storage Area Networks (SANs). The servers that run the platform are hosted at Quality Technology Services with a disaster recovery site at XO. At all levels, these environments are redundant, fault tolerant, and backed up to industry standards.

Web Services Description Language (WSDL) documents describe the detailed Services & Ports (Function Calls) available in this specification. See <http://www.w3.org/TR/wsdl> for more on WSDL.

Please note that all Web Service and WSDL URLs in this document are subject to change based on changing infrastructure requirements. CIQ will provide sufficient advanced notice to the client before changing any URL, hostname, IP address, etc. It is recommended that these URLs be configurable (via config files, etc.) on the client application so that changes can be handled with minimal user downtime. CIQ monitors activity on Production systems and may shut down improper-use processes or user accounts as required to preserve overall system health.

All Web Services requests and responses in this solution are encoded in the UTF-8 character set (<http://en.wikipedia.org/wiki/UTF-8>). Some string data in this solution is expected to only contain Windows-1252 characters (<http://en.wikipedia.org/wiki/Windows-1252>); these are labeled with "(W1252)" in this document. Other string data in this solution allows full UTF-8 characters; these are labeled with "(UTF-8)" in this document. Email addresses (labeled "(email)" in this document) and website URLs (labeled "(URL)" in this document) have more limited valid character sets. See http://en.wikipedia.org/wiki/Email_address and <http://en.wikipedia.org/wiki/URL> for more information.

All the web services have a WSDL definition that external developers will code against and pull in data that is served from the same Capital IQ data repository as our web platform. For a full menu of our Web Services and implementation documentation, please contact your account manager.

CIQ Technology Services Specifications: Documents Details

Web Service Versioning

Versioning Web Services: Over time, Capital IQ may need to extend the tags or datasets supported by our Web services. As a results we have created a URL based versioning solution provides a scalable framework for the future. Versioning provides a way for to accommodate these enhancements in a graceful manner.

Recommendation: Capital IQ recommends that all users upgrade to version 1.0 if they are using legacy services, to conform to the new URL formats.

How versioning works: Please note in the example below *<ServiceName.asmx>* is replaced with the name of the service and is used for illustration purposes only.

- i. Web Service changes are captured as a new version of the file in a new directory.
 - a. **Version 1** - <https://api.capitaliq.com/ciqdotnet/api/1.0/<ServiceName.asmx>> - Represents the first release of the service
 - b. **Version 2** - <https://api.capitaliq.com/ciqdotnet/api/2.0/<ServiceName.asmx>> - Represents the second release and breaking change or significant enhancement.
 - c. Clients have the ability to transition to the new version of the service or stay on the original version until they can transition older code.
- ii. Latest version of the Service will be located at the following URL.
<https://api.capitaliq.com/ciqdotnet/api/current/<ServiceName.asmx>>. Using the example in section i above <https://api.capitaliq.com/ciqdotnet/api/2.0/<ServiceName.asmx>> would be in its own directory and referenced in the current directory.

Service Changes

Get Document Details

| Service | Version | Comments |
|---------|---------|---|
| URL | Current | https://api.capitaliq.com/ciqdotnet/api/current/GetDocumentDetails.asmx?WSDL |
| URL | 1.0 | https://api.capitaliq.com/CIQDotNet/api/1.0/GetDocumentDetails.asmx?WSDL |

| Release | Version | Comments |
|---------|-------------|--------------------------------|
| 9/2008 | 1.0/Current | Launch of Document Web Service |

CIQ Technology Services Specifications: Documents Details

Scenario

This document represents the specifications for GetDocumentDetails. It allows the client application to retrieve specific document types stored within Capital IQ. For HTML documents clients can retrieve a list of documents and document anchor links within those documents as output.

Documents Details

This function returns information about Documents for a given Company ID, optionally limited by date, form types, keywords, format, and SupplierDocumentId. These documents include but are not limited to SEC, SEDAR, but will expand to support other document types in the future. Each type will require a separate entitlement please contact your Client Development Representative for details.

DocumentObject Ports (Functions):

| | |
|------------------|---|
| DocumentObject() | GetDocumentDetails(String SupplierDocumentId, Integer VersionId, Array Double DataValues()); |
|------------------|---|

Comments:

Client Note: For SEC Documents the SupplierDocumentId type is the Accession Number, a unique identifier issued by the SEC for each filing.

Parameters:

1. String **SupplierDocumentId** – A unique identifier for a Supplier document for an SEC filing, it would be the accession number, which is a unique number issued by the SEC to identify a document. **Input** [Required], [Single]
2. Integer **VersionId** – Some “documents” contain multiple “resources”, “formats” or “revisions”. CIQ combines these concepts into “Version”, which is uniquely identifiable by versionID. Multiple Versions can exist for the same Document. For example, in a typical SEC filing there may be two versionIDs: one for the HTML file delivered to EDGAR, and another Excel version containing the main financial tables from the filing, one table per worksheet. **Input** [Required], [Single]

Client Note: Version = content in a specific format, e.g. 10-K in PDF, Version = content in a specific format, e.g. 10-K in TXT

3. Array Double **DataValue** **O** – Series of numeric values that appear in the Filing document. **Input** [Optional], [Multiple].

Client Note: Applies to HTML documents only. **Additionally** at least one input is required for document retrieval either SupplierDocumentId or VersionId. An exception will be thrown if the input is not provided by the client application.

CIQ Technology Services Specifications: Documents Details

Returns:

1. **DocumentObject** – Each contains information about one Document Type.

DocumentObject

Attributes: (optional, multiple)–

- a. String **SupplierDocumentId** – A unique identifier for an SEC filing, issued by the SEC.
- b. Integer **VersionId** – Some “documents” contain multiple “resources”, “formats” or “revisions”. CIQ combines these concepts into “Version”, which is uniquely identifiable by versionID. Multiple Versions can exist for the same Document. For example, in a typical SEC filing there may be two versionIDs: one for the HTML file delivered to EDGAR, and another Excel version containing the main financial tables from the filing, one table per worksheet.

Client Note: Version = content in a specific format, e.g. 10-K in PDF, Version = content in a specific format, e.g. 10-K in TXT

- c. String **DocumentLocation** – A URL is provided to download all document formats including PDF, MSWord, MEXcel, RTF (Rich Text Format), and HTML.
- d. Array **DataValueLink** ()List of input data values supplied and their corresponding html anchor locations within document. If input is not provided for **DataValues** () this will not be present in the XML output.
 - i. Integer **DataValue** – The numeric values that the client application supplied.
 - ii. Array **AnchorList**() – An array (Integer/String) that link the values to specific point in the HTML presentation of the document. It allows the users to hyperlink within the document.

Client Note: The DataValueLink () applies to HTML output only, all other output types will not generate a return value.

Exceptions:

1. An exception will be thrown if the request cannot be authenticated via a session cookie.
2. An exception will be thrown if any parameter is out of range.
 - a. Must have SupplierId or VersionId as an input or and exception will be thrown

Web Services Description Language (WSDL)

Notwithstanding anything to the contrary in this Agreement, Capital IQ reserves the right to change, expand or modify Web Services Definitions and corresponding Web Services Description Language files (WSDL) at any time. Any such modifications are in accordance with industry standards that support backwards compatibility with previous WSDL files.

Client Note: Login using the supplied Username and Password provided by Client Support or your Client Development representative.

CIQ Technology Services Specifications: Documents Details

Appendices

1. **Windows-1252 A character encoding of the Latin alphabet**, used by default in the legacy components of Microsoft Windows in English and some other Western languages. The encoding is a superset of ISO 8859-1, but differs from the IANA's ISO-8859-1 by using displayable characters rather than control characters in the 0x80 to 0x9F range. It is known to Windows by the code page number 1252, and by the IANA-approved name "windows-1252". This code page also contains all the printable characters that are in ISO 8859-15 (though some are mapped to different code points).
2. **Extensible Markup Language (XML)** is a general-purpose markup language. Its primary purpose is to facilitate the sharing of data across different information systems, particularly via the Internet.
3. **dateTime [Definition:]** values may be viewed as objects with integer-valued year, month, day, hour and minute properties, a decimal-valued second property, and a Boolean timezoned property. Each such object also has one decimal-valued method or computed property, `timeOnTimeline`, whose value is always a decimal number; the values are dimensioned in seconds, the integer 0 is 0001-01-01T00:00:00 and the value of `timeOnTimeline` for other `dateTime` values is computed using the Gregorian algorithm as modified for leap-seconds. The `timeOnTimeline` values form two related "timelines", one for timezoned values and one for non-timezoned values. Each timeline is a copy of the `·value space·` of `decimal`, with integers given units of seconds.

The `·value space·` of `dateTime` is closely related to the dates and times described in ISO 8601. For clarity, the text above specifies a particular origin point for the timeline. It should be noted, however, that schema processors need not expose the `timeOnTimeline` value to schema users, and there is no requirement that a timeline-based implementation use the particular origin described here in its internal representation. Other interpretations of the `·value space·` which lead to the same results (i.e., are isomorphic) are of course acceptable.

All timezoned times are Coordinated Universal Time (UTC, sometimes called "Greenwich Mean Time"). Other timezones indicated in lexical representations are converted to UTC during conversion of literals to values. "Local" or untimezoned times are presumed to be the time in the timezone of some unspecified locality as prescribed by the appropriate legal authority; currently there are no legally prescribed timezones which are durations whose magnitude is greater than 14 hours. The value of each numeric-valued property (other than `timeOnTimeline`) is limited to the maximum value within the interval determined by the next-higher property. For example, the day value can never be 32, and cannot even be 29 for month 02 and year 2002 (February 2002). For more details <http://www.w3.org/TR/xmlschema-2/#dateTime>