

# **Sentinel Batch Processing and Real Time API**

---

## **User's Reference Manual**



Truth Technologies, Inc.

[www.truthtechnologies.com](http://www.truthtechnologies.com)

© Copyright 2011, Truth Technologies. All rights reserved.

## **NOTIFICATION OF TRUTH TECHNOLOGIES' PROPRIETARY INFORMATION**

This document contains proprietary, competitively sensitive, and trade secret information. The information furnished in this document is not to be disclosed, in whole or in part, to persons outside your organization without the written permission of Truth Technologies, Inc.

### CONTACTING TRUTH TECHNOLOGIES, INC:



[support@truthtechnologies.com](mailto:support@truthtechnologies.com)

[www.truthtechnologies.com](http://www.truthtechnologies.com)

# Table of Contents

<b>INTRODUCTION .....</b>	<b>5</b>
<b>SENTINEL OVERVIEW .....</b>	<b>5</b>
<b>SYSTEM TO SYSTEM PROCESSING OVERVIEW .....</b>	<b>5</b>
BATCH PROCESSING .....	6
REAL TIME SERVICE .....	6
<b>SENTINEL BATCH PROCESSING .....</b>	<b>7</b>
FILE FORMATS .....	7
SUBMITTING DATA .....	9
Batch Upload .....	9
Advanced Batch Processing .....	10
XML File Formats .....	12
Recommended Applicant.xsd Data Items .....	12
Creating an XML Batch Request Document .....	13
BatchRequestPreprocessConfig Utility .....	13
Submitting a Batch Request .....	15
Retrieving a Batch Process .....	16
Batch Processing Using Web Services .....	16
<b>REAL TIME SERVICES .....</b>	<b>16</b>
TRUTH TECHNOLOGIES SUPPLIED API .....	16
ACCESSING WEB SERVICES DIRECTLY .....	16
<b>APPENDIX: .....</b>	<b>18</b>
UTILITY CLASSES .....	18
WSClientSubmitBatch .....	18
WSClientGetBatch .....	19
WSClientSingleCheck .....	19
BatchRequestPreprocessConfig .....	20
WSClientSubmitBatchSwift .....	22
WSClientSingleCheckSwift .....	24
WSClientApplicantStatusUpdate .....	26
WSClientRetrieveApplicantRiskStatus .....	27
XML and WSDL Documents .....	29
XML MAPPING .....	30
XML AND SOAP DOCUMENT SAMPLES .....	31
Batch Processing .....	32
Real Time Service - SOAP INPUT .....	32

## Introduction

Truth Technologies Incorporated is in the business of providing products and services that use information technology to increase the safety and freedoms of individuals and businesses involved in financial transactions.

This document outlines the batch processing function of the Sentinel product. Sentinel is based on the unique application of powerful technologies that have been extended for commercial use and is specifically tailored to assist the financial community in reducing fraud and meeting the requirements of the US Patriot Act, 3<sup>rd</sup> AML Directive, and other Anti-Money Laundering legislation.

To better understanding Truth Technologies' Sentinel and the purpose of the Batch Processing module, it is necessary to understand the basic points of Anti-Money Laundering (AML) legislation.

## Sentinel Overview

Truth Technologies Incorporated has developed a set of products that uses information technology to increase the safety and freedoms of individuals and businesses involved in financial transactions. The main product is Sentinel. Sentinel is based on the unique application of powerful technologies that have been extended for commercial use and is specifically tailored to assist the financial community in reducing fraud and meeting the requirements of the US Patriot Act, 3<sup>rd</sup> AML Directive, and other Anti-Money Laundering (AML) legislation.

Sentinel assesses the risk of an account holder participating in money laundering, fraud or criminal activity. This check can be done before or after an account is opened. The account holder (herein after referred to as 'the Applicant') is checked against a set of government and private data sources. Identifying information about the applicant is compared against these data sources to determine the accuracy of the identification data provided, and to see if the Applicant poses a high risk of being involved in fraud, crime or terrorism. A risk level of 'High', 'Medium' or 'Low' is then assigned for each data source check. Details about why an applicant received a particular risk level for a given data source are available for review. This aids the Compliance Officer in the final decision regarding the total risk of an applicant.

Sentinel assesses the risk associated with an applicant. However, the risk assessment process is not limited to the account holder; instead, the process is broadened to assess the risk associated with those which share a relationship with the account holder. For example, individuals may have a higher risk because of their employer, or because of the persons that control that employer are deemed to be high risk. This form of risk assessment is holistic. It is akin to an analyst building a 'network diagram', one that describes the relationships between entities and gives a truer picture of risk than checking just the applicant's data alone.

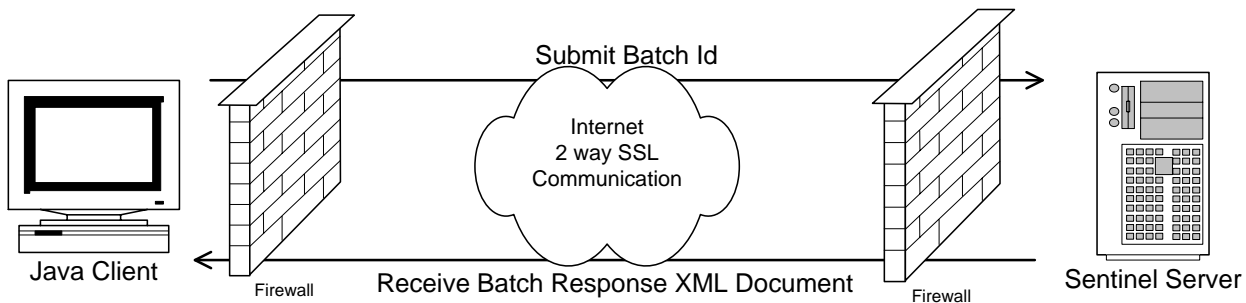
Sentinel Batch Processing provides integrated capability to embed the Sentinel process into existing systems or to provide periodic background checks on existing customers to ensure an existing customer doesn't appear on a government sanctions list.

## System to System Processing Overview

Sentinel allows system to system communication through asynchronous batch processing and a real time web service API. The real time API from an infrastructure standpoint works similarly to the batch process with the main difference being you get an immediate result but can check only one applicant at a time.

## Batch Processing

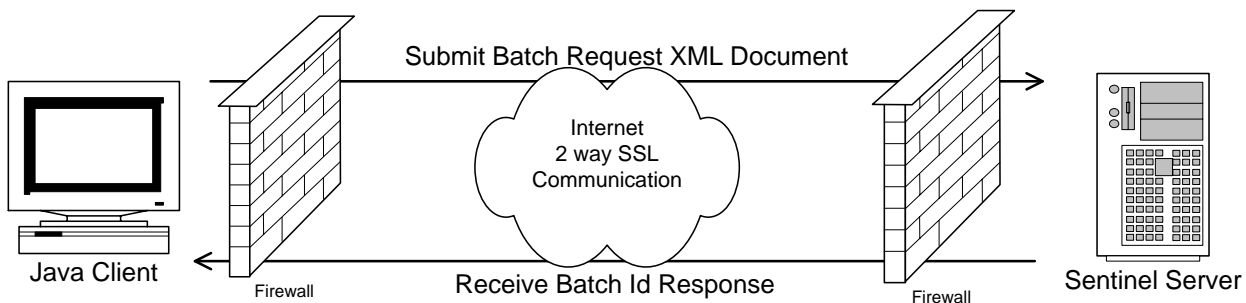
Batch processing is the submission of a set of customer data to the system to be processed at a later date. This is known as an asynchronous process as it takes two steps for a client to complete the batch process. First the client submits the batch request XML document, and then in a second later message the client requests the results of the batch process. A diagram of the submit batch request process and retrieve batch result process is provided below:



Figure

1: Batch Process Submission Process

The process is initiated by submitting a request to Truth Technologies' Sentinel Server. This request consists of identifying information, user name, password, certificates, etc, as well as the actual batch request XML document, see applicantBatch.xsd for definition of the batch request document. The server responds immediately with either an error message or the batch system id for the request. At a later time this batch system id can be used to retrieve the results of the batch process.



Figure

2: Batch Process Results Retrieval

The batch system id along with client identifying data is submitted from the client to Truth Technologies' servers for retrieval of the batch process results. The applicant check results are formatted into an XML document and immediately returned to the client.

## Real Time Service

The real time service allows systems to submit an xml document for one applicant check to Sentinel via a web service and get an immediate response, also in xml format. The service uses the same 2 way SSL communication used by the Batch Process and outlined in Figure 1.

## Sentinel Batch Processing

Batch processing allows clients to submit a file with a large number of customer names for screening as part of a single batch job. Once the batch job is complete clients can retrieve the results directly from Sentinel via web services or log in to the Sentinel application and review and remediate the results.

The customer data file submitted as part of the batch process can be either a "flat file" or an XML file format. A flat file is just a text delimited file that has one row of data for each of your customer names that needs to be screened. Most clients use a flat file because it is easier to produce, but you are free to use either format.

Data can be submitted and converted online, or Advanced Batch processing can be employed using TTI supplied utilities and web services. Typically clients with smaller batches and clean well defined customer data will use the online submission process as it requires little IT intervention and allows end users flexibility on when batch data is submit. Clients with larger batches, a need for an automated processes or customer data that has more advanced conversion and formatting requirements will use the Advanced Batch processing.

## File Formats

Regardless of the file format there are two typical sticking points. First you can only have one name per line in the file. For instance accounts that are joint accounts, have beneficiaries, board members or signatories each have to have one line for each name. Second the system works best if each record can be identified as either a person or a company. This identification must go in the first column and is typically either a C for companies or an I for individuals. Identifying each record is not a requirement and can be dealt with using the sample format `batch_sample_noIC.txt`.

If you use a delimited text file format you can use any delimiter you like, comma, pipe, tab, etc, just indicate what delimiter is used. If the Advanced Batch conversion utility is used then the actual file contents are flexible so the data points can be in any order except the Customer Type column which lists "I" for individuals and "C" for companies must be first. Last name and account number are the other key required fields, all other fields are optional.

Sample file formats in MS Excel and delimited text files are available in the help section of the web site; see the Help – User's Manual – Batch Processing page. If you are using a MS Excel customer data file the file must be converted to a text file. To convert the file open the Excel file and save it as a text tab delimited file.

There are two sample formats. In order to use the online batch upload screens you must use one of the sample formats. If your data does not conform to the samples you can use the Advanced Batch processing utilities provided by TTI. The Advanced Batch process allows you to use virtually any data file format.

The difference between the two samples is whether you know if your data is for a company or individual and whether the names for individuals are broken into parts, first and last name, or the whole name is in one column.

If you know if the records are people or companies and the name is broken into parts then use the batch\_sample.xls format.

If you do not know if the records are people or companies and the name is in one column then use the batch\_sample\_noIC.xls format.

The default file format requires that records be identified as either individuals or companies. The Record Type Unknown format has the entire customer name in one column and does not require that record type be identified.

## Input File Columns

Column Name	Required	Length	Description
Type	Yes / No	1	Customer type. Either a "C" for companies or an "I" for individuals (Not required for noIC file format)
Unique Identifier	No	50	Internal identifier assigned to a customer by the client. This may be an account number, customer number, master file record number, or a number you make up. The important point is that every time you submit the data it is the same for a customer name.
First Name	No	40	Customer first name
Middle Name	No	40	Customer middle name
Last Name / Company Name	Yes	60 / 300	This is a person's last name, or family names. It is also an entities name. entity names can be 300 characters
Citizenship	No	50	Citizenship or nationality country. TTI recommends using the ISO 3166 two or three character country codes, but country names can be used as well, although they will be less effective due to country name variation.
Date of Birth	No	10	Customer date of birth. The web upload requires a format of DD/MM/YYYY. If you use the conversion utility you can use whatever format you like.
Address 1	No	250	First line of customer address
Address 2	No	250	Second line of customer address
City	No	50	City, town or other address data



State / Province	No	50	State, province, district
Postal Code	No	25	Postal or zip code
Country	No	50	Address Country. TTI recommends using the ISO 3166 two or three character country codes, but country names can be used as well, although they will be less effective due to country name variation.
Occupation	No	150	Customer occupation
Notes	No	2000	Free form field for placing additional data. This is for informational purposes only and does not affect screening results

## Submitting Data

Sentinel allows users to submit a batch one of four ways:

- Batch Upload - upload a delimited text file in the Sentinel application
- Batch XML Upload - upload a XML file in the Sentinel application
- Command Line (Advanced batch Processing) - submit a batch file using secure web services
- Web Services - submit a batch file by calling the secure web services

### Batch Upload

**Note:** See the Help – User's Manual – Batch Processing page to download the sample file formats and additional documentation.

Users can upload delimited text files using the Sentinel web application, Resources - Batch Upload. This eliminates the need to have to run the command line file conversion and batch submission commands. This new feature only works if your file format meets one of two "predefined" file formats, for all other formats you will need to continue to use the Command Line process.

**Default File Format** - This is the default file format expected. It requires that the first column of your data include either a "C" for companies or an "I" for individuals. Individual names can be in three columns, first, middle, last, or completely contained in the last name column. If the name is completely contained in the last name column you need to mark "Customer name contained in a single column" as Yes.

**Record Type Unknown File Format** - This file format is used if you do not know if your data records are individuals or corporations

All file uploads use European date standard: DD/MM/YYYY

All file uploads use ISO 3166 2 or 3 character country codes for citizenship and address country. [ISO 3166](#)

## Upload Options

- Batch File (delimited) - The delimited text file containing your customer data. If your data is in a MS Excel file open the file and save the file as a Text Tab delimited file.
- Delimiter - The character used to delimit data columns. It is recommended to use a character that does not appear naturally in data such as a pipe |
- Header Row - Yes if the first row of the data file contains column header names
- Customer name contained in a single column - Yes if the entire customer name is in one field, usually the last name field. No if data is in first, middle and last name or just first and last name fields
- Record type identified in the first column - Yes if the first column of your data include either a C for companies or an I for individuals

## Batch XML Upload

See XML and WSDL Documents on page 22 of Batch Processing and Web Services Manual

## Command Line (Advanced batch Processing)

Advanced batch processing is a two step process. First you must convert your text file to XML, and then you must submit the XML file to Sentinel for processing.

You will need to download the Sentinel batch utilities and communication files from the Help – Download page.

## Web Services

See Accessing Web Services Directly on page 15 of Batch Processing and Web Services Manual

## Advanced Batch Processing

Advanced batch processing is a two step process. First you must convert your text file to XML, and then you must submit the XML file to Sentinel for processing.

Requirements: Java version 1.5 or 1.6

To set up the process follow these steps.

1. Create a working directory. This is the directory that your customer data file will reside in and that all conversion and submission commands will be run out of.
2. Download the needed files from the Help – Download Files page
3. Extract the sentinel\_batch\_files.zip into your working directory. Once extracted sample commands and files are located in the ./doc directory.
4. Delete the sentinel\_batch\_files.zip

5. Place your customer data file into the working directory. If you are using a MS excel customer data file, open the file and save it as a text tab delimited file
6. Open a command window and cd to the working directory. In windows go to start, run, type cmd
7. Verify that Java is installed by typing `java -version` You should get a message "java version 1.5.x or 1.6.x".  
NOTE: java 1.4 is no longer supported by Truth Technologies.
8. Edit the batchProcess.properties file to have the proper column settings for your data file format. See the doc dir for samples. Change the max rows to be more than the number of customer records in your data file.
9. Run the conversion command; be sure to replace the username, filename and options to match your setup. Samples of commands are listed below.
10. Run the Submit Data command. Samples of commands are listed below.

Name	Size	Type
doc		Folder
endorsed		Folder
commons-io-1.4.jar	107 KB	Executable Jar File
commons-lang-2.3.jar	240 KB	Executable Jar File
javassist.jar	611 KB	Executable Jar File
jboss-common-core.jar	567 KB	Executable Jar File
jboss-logging-spi.jar	16 KB	Executable Jar File
jboss-remoting.jar	1,21...	Executable Jar File
jbossws-common.jar	142 KB	Executable Jar File
jbossws-native-client.jar	8 KB	Executable Jar File
jbossws-native-core.jar	1,41...	Executable Jar File
jbossws-native-jaxrpc.jar	36 KB	Executable Jar File
jbossws-spi.jar	123 KB	Executable Jar File
jboss-xml-binding.jar	534 KB	Executable Jar File
jdom.jar	150 KB	Executable Jar File
mail.jar	445 KB	Executable Jar File
sentinelws-client.jar	104 KB	Executable Jar File
wife.jar	689 KB	Executable Jar File
wSDL4j.jar	195 KB	Executable Jar File
xercesImpl.jar	1,19...	Executable Jar File

### Sample Working Directory

### Batch Processing Command Samples

**Note:** < > indicates items that need to be replaced. Replace entire string including the < > with correct value. Some items have suggested values inside the < > which may be used others have ??? which needs to be replaced

# convert flat file for batch processing

# NOTE by default this command uses the batchProcess.properties file. If you need to use another property file add "-prop filename.proeprties"

```
java -classpath ".,./*" com.idevco.webservice.xmlUtil.BatchRequestPreprocessConfig -username <??> -infile <batch_sample.txt> -token tab -skip -identify C -outfile batchOut
```

# submit batch process

```
java -classpath ".,./*" com.idevco.webservice.client.WSClientSubmitBatch -username <??> -userpassword <??> -filename <batchOut1.xml>
```

# (Optional) retrieve results of batch process as XML

```
java -classpath ".,./*" com.idevco.webservice.client.WSClientGetBatch -username <??> -userpassword <??> -batch_id <??> -output <batchResult.xml>
```

## XML File Formats

### Recommended Applicant.xsd Data Items

For schema locations see XML and WSDL Documents applicantbatch.xsd and applicant.xsd.

XML Parent Tag	Tag	
applicant	<name first_name="Bob" middle_name="J" last_name="Smith"/>	First, middle and last name. If your data is not formatted this way populate last name and use first name for the rest of the name.
address	<address_line1>123 W. Mill St</address_line1> <address_line2/> <city>Annandale</city> <state>VA</state> <postal_code>22003</postal_code> <country country_code="US"/>	Country uses ISO 3166 two character country codes
date_of_birth	<date_of_birth>2001-05-28</date_of_birth>	YYYY-MM-DD format
Occupation	<occupation>Consultant</occupation>	
Citizenship	<country_code="US"/>	Country uses ISO 3166 two character country codes

company_data	<company_name>ACME Inc.</company_name>	
address	See above	The company address information

### **Creating an XML Batch Request Document**

Due to the international nature of the data used by Sentinel all XML documents use UTF-8 encoding. Sentinel java client classes are hard coded to use the UTF-8 code when writing and sending files. The input format for files can be controlled using the `-encode` command line switch. If you are having problems with invalid characters in your custom code run the conversion utility with the `-debug` switch to view file encodings.

Review the applicantBatch.xsd document to determine what applicant data is required, see above. Once you are familiar with what data is required determine the best way to extract that data from your back end system. Options include but are not limited to:

- Modifying the backend system to create the desired document.
- Extracting data from the back end database into a delimited file then parsing that file to create an XML file. (see [BatchRequestPreprocessConfig](#) utility)
- Writing a standalone Java, C or .Net application to create the XML document.
- Working with a third party, such as Truth Technologies, to create a utility to create the document.
- Creating the document by hand (this is not recommended).

Once the document has been created verify the validity of the document using an XML tool such as XML Spy, or other validating parser. Validity of the document may also occur on Truth Technologies side depending on the size of the document.

### **BatchRequestPreprocessConfig Utility**

This utility reads a flat file and converts it into a batch submit xml document. The utility differs from other batch utilities in that the location of data in delimited input text file is read from the batchProcess.properties file, located in the current working directory. This allows users greater control over file formats and reduces the number of command line arguments. See the sample batchProcess.properties file located in the doc directory of the sentinelws\_client.jar file for more details.

#### **Required Arguments**

- username      Sentinel account username
- infile        File name of input file with customer data to be checked
- token        Field separated used in input file.  
\$ is recommended but any char can be used

Use word 'tab' to indicate tab delimited files

-outfile File name of output file that will be used for batch processing

### **Optional Arguments**

-debug Print out debug information

-path (DEFAULT ./) Path where input file is located and output file will be placed, must end with /

-identify If the first column of the input file identifies the record type or parsing function, or if no identify character is supplied how to handle identifying whether a name is a person or company. If this argument is not supplied then both company and person data are read.

'C' - company or entity records are identified

'I' - individual records are identified

'BI' - records are not identified but the utility will try to guess whether the name is a person or company based on content. To account for errors, if the name is a person the name will be placed in both person and company

'B' - records are not identified but the utility will try to guess whether the name is a person or company based on content

'N' - records are not identified. This is the DEFAULT value

-identchar The character used to identify records. If not specified the same character used for identify is expected ('C' or 'I')

-namesplit tells the parser to attempt to split individual names into first and last name. DEFAULT value is false. If name split is used the util will read the last\_name column and attempt to split the name into first and last name.

-skip skip the first row

-encode file encoding type, usually UTF-8 or Cp1252

### **Sample batchProcess.properties File**

```
# Tell util which column the data resides in, first col = 1. If the data is not present leave col = 0.
```

```
# company column settings
```

```
company_name=0
```

```
caddress_line1=0
```

```
address_line2=0
ccity=0
cstate=0
cpostal_code=0
ccountry_code=0
cphone_type=0
cphone=0
```

```
# person column settings
```

```
first_name=3
middle_name=0
last_name=0
address_line1=0
address_line2=0
city=0
state=0
postal_code=0
country_code=0
phone_type=0
phone=0
date_of_birth=0
citizenship=0
```

```
# this can be used for customer occupation or as a placeholder for additional data
```

```
occupation-note=0
```

```
passport=0
```

```
ssn=0
```

```
# other properties
```

```
# refid is a unique id for a customer record
```

```
refid=0
```

```
# uses java standards - http://java.sun.com/j2se/1.4.2/docs/api/java/text/SimpleDateFormat.html
```

```
date_format=yyyy/MM/dd
```

```
# maximum row in a file. If number of input rows exceeds max more than one file will be created
```

```
max_rows=20000
```

```
note=0
```

### ***Submitting a Batch Request***

To submit a batch process, be sure that you have a valid batch document, and have set up your client machine correctly. See the section entitled "[Setting up the Client Machine](#)".

At the command line type the following:

```
java -classpath ".*/*" com.idevco.webservice.client.WSClientSubmitBatch -username <??> -  
userpassword <??> -filename <batchOut1.xml>
```

After a few seconds you will get a batch ID as a result. Save this ID as you will need it to retrieve the batch results.

**Note:** If your password contains a special character such as \$ or ^ you may need to enclose the password in single or double quotes. Single quotes should be used on Linux machines and double quotes on windows machines.

### **Retrieving a Batch Process**

To get the results of a batch process run the following command:

```
java -classpath ".*/*" com.idevco.webservice.client.WSClientGetBatch -username <??> -  
userpassword <??> -batch_id <??> -output <batchResult.xml>
```

The results will be written to the filename you supplied. If you prefer the results to be written to the screen omit the `-output` option. For large batch processes you may need to use command **java -Xmx512M ...**

### **Batch Processing Using Web Services**

The batch process can be accessed directly using web services. In order to use the batch process web service directly the client customer data must be formatted in an XML document using UTF-8 encoding. See Tables 1-3 in the [Accessing Web Services Directly](#) for document and format details.

## **Real Time Services**

### **Truth Technologies Supplied API**

You must have first completed the steps in [Setup the Client Machine](#).

NOTE: Single applicant checks conform to their own XSD see Appendix. All XML documents submitted as part of the real time API must be in UTF-8 format.

```
java -classpath ".*/*" com.idevco.webservice.client.WSClientSingleCheck -username <??> -  
userpassword <??> -filename <./RealTimeSamples/applicantCheckPEP.xml> -output  
<./realTimeResult.xml>
```

### **Accessing Web Services Directly**



The batch process and real time services can be accessed directly using web services. All XML documents submitted as part of the real time API must be in UTF-8 format. The US wsdl documents that describe the web services are located at: (Cayman wsdl documents should be at <http://cayman.truthtechnologies.com/wsdl...>)

See [XML and WSDL Documents](#) for a table listing the URLs for the web service WSDL files.

**Note:** All endpoints are preceded with the following

US Server: <https://sentinel.truthtechnologies.com/sentinelWS/webservices>

Cayman Server: <https://cayman.truthtechnologies.com/sentinelWS/webservices>

Message	End Point (Replace ... with text above)	Description
checkApplicants	.../batchService	Submit a batch process
getBatchStatus	.../batchService	Retrieve results of a batch process
checkApplicant	.../applicantService	Check a single applicant in real time
postApplicantUpdates	.../applicantService	Change the result status or add annotations for a search
retrieveRiskStatus	.../applicantService	Retrieves the risk status of the given applicant.

**Table 1**

Message	Argument	Description
checkApplicants	arg0 - string	User name
	arg1 - string	User password
	arg2 - string	Batch xml doc - see applicantBatch.xsd
getBatchStatus	arg0 - string	User name
	arg1 - string	User password
	arg2 - string	Batch Id
checkApplicant	arg0 - string	User name

	arg1 – string	User password
	arg2 – string	Applicant xml doc – applicantCheckSingle.xsd
postApplicantUpdates	arg0 - string	User name
	arg1 – string	User password
	arg2 – string	Audit user name
	arg3 - string	Applicant id
	arg4 – string	Data source question id (usually 392199)
	arg5 – string	Result ID
	arg6 – string	Annotation text
	Arg7 - string	Account Number
retrieveRiskStatus	arg0 - String	User name
	arg1 - String	User password
	arg2 - String	Data source question id (usually 392199)
	arg3 - String	Applicant id
	arg4 - String	Account Number

**Table 2**

## Appendix:

This section describes the Java Classes utilized in the Sentinel Batch Process. Note commands are listed in Windows format. For Unix use : as a path separator.

### Utility Classes

#### **WSClientSubmitBatch**

Full class name: com.idevco.webservice.client.WSClientSubmitBatch

Purpose: Submit batch request XML document to sentinel for processing.

Usage: **`java -classpath ".*" com.idevco.webservice.client.WSClientSubmitBatch -username <username> -userpassword <password> -filename <applicant_xml_file> [-debug] [-output]`**

Required Parameters	Description
<username>	Sentinel user login name
<userpassword>	Password for Sentinel user
<filename>	XML document you want processed as part of the batch. Must comply with the applicantBatch.xsd schema definition.

### **WSClientGetBatch**

Full class name: com.idevco.webservice.client.WSClientGetBatch

Purpose: Retrieve batch results.

Usage: **`java -classpath ".*" com.idevco.webservice.client.WSClientGetBatch -username <username> -userpassword <password> -batch_id <batch ID> [-debug] [-output]`**

Required Parameters	Description
<username>	Sentinel user login name
<password>	Password for Sentinel user
<batch ID>	Batch ID that was returned when the batch process was submitted.

### **WSClientSingleCheck**

Full class name: com.idevco.webservice.client.WSClientSingleCheck

Purpose: Access Real Time API to check a single applicant.

Usage: **`java -classpath ".*" com.idevco.webservice.client.WSClientSingleCheck -username <username> -userpassword <password> -filename <filename> [-debug] [-output]`**

Required Parameters	Description
<username>	Sentinel user login name
<password>	Password for Sentinel user
<filename>	XML document containing the applicant you would like checked.

### **BatchRequestPreprocessConfig**

Full class name: com.idevco.webservice.xmlUtil.BatchRequestPreprocessConfig

**Purpose:** This utility reads a flat file and converts it into a batch submit xml document. The utility differs from other batch utilities in that the location of data in delimited text file is read from the batchProcess.properties file. This allows users greater control over file formats and reduces the number of command line arguments. See the sample batchProcess.properties file located in the doc directory of the sentinelws\_client.jar file for more details.

**Usage:**

```
java -classpath ".:/"
com.idevco.webservice.xmlUtil.BatchRequestPreprocessConfig -username
<username> [-path <file path>] -infile <filename> -token <delim
token>[-outfile <file name>] [- identify <C I B I B or N>] [-namesplit] [-
identchar <?>] [-path <./>] [-prop <./batchProcess.properties>] [-encode
<UTF-8 or Cp1252>] [-skip] [-debug]
```

Required Parameters	Description
<username>	Sentinel user login name
<infile>	File name of input file with customer data to be checked
<outfile>	File name of output file that will be used for batch processing
<token>	Field separated used in input file. \$ is recommended but any char can be used. Use word 'tab' to indicate tab delimited

	files
--	-------

Optional Parameters	Description
<identify>	<p>If the first column of the input file identifies the record type or parsing function, or if no identify character is supplied how to handle identifying whether a name is a person or company. If this argument is not supplied then both company and person data are read.</p> <p>'C' - company or entity records are identified            'I' - individual records are identified            'BI' - records are not identified but the utility will try to guess whether the name is a person or company based on content. To account for errors, if the name is a person the name will be placed in both person and company            'B' - records are not identified but the utility will try to guess whether the name is a person or company based on content            'N' - records are not identified. This is the DEFAULT value</p>
<identchar>	The character used to identify records. If not specified the same character used for identify is expected ('C' or 'I')
<path>	Path where input file is located and output file will be placed (DEFAULT ./). Must end with /
<prop>	name of batch process mapping property file
<namesplit>	Tells the parser to attempt to split individual names into first and last name. DEFAULT value is false. If name split is used the util will read the last_name col and attempt to split the name into first and last name.
<encode>	File encoding type, usually UTF-8 or Cp1252. (DEFAULT) if this argument is not specified the system default encoding page will be used.
<skip>	Skip the first row of input
<debug>	Print debugging info

## WSClientSubmitBatchSwift

Full class name: com.idevco.webservice.client.WSClientSubmitBatchSwift

**Purpose:** Converts a swift message file into an xml document which conforms to <http://sentinel.truthtechnologies.com/schema/applicantBatch.xsd> and submits the batch.

<b>Usage:</b>	<pre>java -classpath ".;../*" com.idevco.webservice.client.WSClientSubmitBatchSwift - username &lt;username&gt; -password &lt;password&gt; [ -path &lt;file path&gt;] -infile &lt;filename&gt; [ -outfile &lt;file name&gt;] [ - custType &lt;C/BI/ B&gt;] [ -encode &lt;UTF-8 or Cp1252&gt;] [- parties &lt;S/B/A&gt;] [-debug]</pre>
---------------	--

Required Parameters	Description
<username>	Sentinel user login name
<infile>	File name of input file with customer data to be checked
<password>	Sentinel user's password.
<outfile>	File name of output file that will be used for batch processing

Optional Parameters	Description
<custType>	<p>How to determine if the customer is a person or a company. If this argument is not supplied all data is treated as a person.</p> <p>'BI' – (Default Value) the utility will try to guess whether the name is a person or company based on content</p> <p>'I' - all data is treated as a person</p> <p>'C' - all data is treated as company/business.</p>
<path>	Path where input file is located and output file will be placed (DEFAULT ./). Must end with /

<encode>	File encoding type, usually UTF-8 or Cp1252. (DEFAULT) if this argument is not specified the system default encoding page will be used.
<parties>	Include the sender and / or the beneficiary B = Beneficiary (Default) S = Sender A = All (both beneficiary and sender )
<debug>	Print debugging info
<submitFile>	By default the request gets submitted to the sentinel web service. If passed in with a parameter 'false' the submission will be skipped, just the output file gets generated.

## WSClientSingleCheckSwift

Full class name: com.idevco.webservice.client.WSClientSingleCheckSwift

Purpose: Converts a swift message file into an xml document and submits the generated document for real time check. See [Appendix - XML and WSDL Documents](#) for details on XML schema.

Usage:	<pre>java -classpath ".;../*" com.idevco.webservice.client.WSClientSingleCheckSwift - username &lt;username&gt; -password &lt;password&gt; [ -path &lt;file path&gt;] -infile &lt;filename&gt; [ -outfile &lt;file name&gt;] [ - custType &lt;C/BI/ B&gt;] [ -encode &lt;UTF-8 or Cp1252&gt;] [- parties &lt;S/B&gt;] [-debug]</pre>
--------	--

Required Parameters	Description
<username>	Sentinel user login name
<infile>	File name of input file with customer data to be checked
<password>	Sentinel user's password.
<outfile>	File name of output file that will be used for batch processing

Optional Parameters	Description
<custType>	<p>How to determine if the customer is a person or a company. If this argument is not supplied all data is treated as a person.</p> <p>'BI' – (Default Value) the utility will try to guess whether the name is a person or company based on content</p> <p>'I' - all data is treated as a person</p> <p>'C' - all data is treated as company/business.</p>
<path>	Path where input file is located and output file will be placed (DEFAULT ./). Must end with /
<encode>	File encoding type, usually UTF-8 or Cp1252. (DEFAULT) if this argument is not specified the system default encoding page will be used.



<parties>	Include the sender and / or the beneficiary B = Beneficiary (Default) S = Sender A = All (both beneficiary and sender ) Please note that since it is a real time check only one applicant can be submitted, either sender or the beneficiary but not both.
<debug>	Print debugging info
<submitFile>	By default the request gets submitted to the sentinel web service. If passed in with a parameter 'false' the submission will be skipped, just the output file gets generated.

## WSClientApplicantStatusUpdate

Full class name: com.idevco.webservice.client.WSClientApplicantStatusUpdate

Purpose: Post result status updates and annotation notes to applicant results.

Usage: **java -classpath ".:./\*" com.idevco.webservice.client.WSClientApplicantStatusUpdate -username <username> -userpassword <userpassword> -debug -auditUsername <auditusername> -applicantId <applicantId> -dataSourceQuestionId <dataSourceQuestionId> -resultId <resultId> -annotationText <annotationText>**

Required Parameters	Description
<username>	Sentinel user login name
<password>	Password for Sentinel user
<auditUsername>	The user name to be recorded in the audit log for the updates posted.
<applicantId>	The applicant's Id**
<dataSourceQuestionId>	The data source question Id. W-C questionId = 392199
<accountNumber>	Account Number **

\*\* Either applicant id or account number should be provided.

Note: At least one of the following parameters should be present in the request.

Optional Parameters	Description								
<resultId>	<p>The applicant's result Type will be updated to the passed in result Id. The following table shows valid result Ids and corresponding result Type descriptions.</p> <table border="1"> <thead> <tr> <th>Result Id</th> <th>Result Type</th> </tr> </thead> <tbody> <tr> <td>22</td> <td>Review – Pending</td> </tr> <tr> <td>23</td> <td>Reviewed – Match Verified</td> </tr> <tr> <td>27</td> <td>Reviewed - Match Verified (Accepted )</td> </tr> </tbody> </table>	Result Id	Result Type	22	Review – Pending	23	Reviewed – Match Verified	27	Reviewed - Match Verified (Accepted )
Result Id	Result Type								
22	Review – Pending								
23	Reviewed – Match Verified								
27	Reviewed - Match Verified (Accepted )								

	28	Reviewed - Match Verified (Declined)
	26	Reviewed – False Positive
	29	Inactive
<annotationText>	An annotation will be added to the applicant(s) 's record(s).	

### **WSClientRetrieveApplicantRiskStatus**

Full class name: com.idevco.webservice.client.WSClientRetrieveApplicantRiskStatus

Purpose: Retrieves risk status for the given applicant Id or account Number.

Results: The current risk status such as "High Risk" / "Medium Risk" etc., of the applicant / account will be returned. If the applicant or account number information is unavailable, risk status value returned will be "Unknown".

Usage:	<pre>java -classpath ".;../*" com.idevco.webservice.client.WSClientRetrieveApplicantRisk Status -username &lt;username&gt; -userpassword &lt;password&gt;- datasourceQuestionId &lt;questionId&gt; [-applicantId &lt;applicantId&gt;] [-accountNumber &lt;accountNumber&gt;]</pre>
--------	--

<b>Risk Status Values</b>
High Risk
High Risk - New Data
Medium Risk
Medium Risk - New Data
Review - Pending
Match Found
Error
Reviewed - Match Verified
Reviewed - Match Verified (Accepted)

Reviewed - Match Verified (Declined)
Reviewed - False Positive
Inactive
Unknown

## XML and WSDL Documents

The following describes the documents and schemas related to the Batch Process. Copies of all schema documents are available at:

[https://sentinel.truthtechnologies.com/schema/document\\_name.xsd](https://sentinel.truthtechnologies.com/schema/document_name.xsd) or  
[https://cayman.truthtechnologies.com/schema/document\\_name.xsd](https://cayman.truthtechnologies.com/schema/document_name.xsd)

Service Message	Document	Description
checkApplicants	applicant.xsd	Schema for applicant information. This schema is a child of applicantBatch.xsd.
	applicantBatch.xsd	Schema for XML document to submit a batch process.
	batchError.xsd	Schema for errors generated during batch processing. This schema is a child of batchResult.xsd.
getBatchStatus	batchResult.xsd	Schema for XML document that is returned from WSCClientGetBatch.
checkApplicant	applicantCheckSingle.xsd	Schema for XML document used by Real Time API
	applicantCheckResult.xsd	Schema for XML document response used by Real Time API

Document	Service Message	Description
<a href="https://sentinel.truthtechnologies.com/sentinelWS/webservices/batchService?wsdl">https://sentinel.truthtechnologies.com/sentinelWS/webservices/batchService?wsdl</a>	checkApplicants	Submit a batch process
Same as above	getBatchStatus	Retrieve results of a batch process
<a href="https://sentinel.truthtechnologies.com/sentinelWS/webservices/applicantService?wsdl">https://sentinel.truthtechnologies.com/sentinelWS/webservices/applicantService?wsdl</a>	checkApplicant	Check a single applicant in real time
Same as above	postApplicantUpdates	Change the result status or add annotations for a search
Same as above	retrieveRiskStatus	Retrieves risk status.

## XML Mapping

Whether there is or is not a match is identified by the “risk” attribute of the dsQuestion tag.

```
<dsQuestion data_source="World-Check" question_desc="Is this person in the World-Check data base?"
question_url="https://sentinel.truthtechnologies.com:443/mls?page=/datasources/pepWC_en.html"
risk="High Risk">
```

### Risk Result Summary

Customer Screening Risk Levels		
High Risk	Match against a brand new record in the data source. The first time customer data is screened all matches will be high or medium risk	Displayed in Red on the Result Detail Screen
High Risk - New Data	Matches existing record but data has been updated for an existing record	Displayed in Blue on the Result Detail Screen
High Risk - Duplicate	Matches existing record and the records is exactly the same as the last time this customer was screened	Displayed in Grey on the Result Detail Screen
Medium Risk	Match against a brand new record in the data source. The first time customer data is screened all matches will be high or medium risk	Displayed in Red on the Result Detail Screen
Medium Risk - New Data	Matches existing record but data has been updated for an existing record	Displayed in Blue on the Result Detail Screen
Medium Risk - Duplicate	Matches existing record and the records is exactly the same as the last time this customer was screened	Displayed in Grey on the Result Detail Screen
Low Risk	Reasonably confident that there is no issues with the data submitted (Watch Lists data sources only)	
No Match Found	Reasonably confident that there is no issues with the data submitted (Identity Verification data sources only)	
Not Applicable	The data needed to use this data source was not provided	
Error	Internal system error. Contact customer support	

If there is a match the details of the match are in the itemSet tag. Below is the mapping to the main tags in our XML document and the key name / value pairs

Tag	Sample	Notes
Result Type	<dsQuestion data_source="World-Check" question_desc="Is this person in the World-Check data base?" question_url="https://sentinel.truthtechnologies.com:443/mls">	

	<code>s?page=/datasources/pepWC_en.html" risk="High Risk"&gt;</code>	
Category	<code>itemData label="Category"</code>	
Sub-Category	<code>itemData label="Sub-Category" value="PEP"</code>	(PEP is the only valid value)
Source List(s)	<code>itemData label="Data Source"</code>	
Customer Name	<code>&lt;name first_name="Frank" middle_name="K" last_name="Kravoski" /&gt;</code>	All results have a name tag which will equal ? if it is a company
Customer Name	<code>&lt;company_name&gt;truth technologies inc&lt;/company_name&gt;</code>	May have a company tag
Customer Address	<code>&lt;address&gt; &lt;address_line1&gt;2145 wolftrap ct&lt;/address_line1&gt; &lt;city&gt;vienna&lt;/city&gt; &lt;state&gt;va&lt;/state&gt; &lt;postal_code&gt;22182&lt;/postal_code&gt; &lt;country country_code="US" /&gt; &lt;/address&gt;</code>	
Matched Name	<code>&lt;itemData label="Last Name" value="MILLER" /&gt; &lt;itemData label="First Name" value="Anthony Vincent" /&gt;</code>	
Alias	<code>&lt;itemData label="Alias" value="MILLER,Tony" /&gt;</code>	Repeating tag
Matched Address	<code>&lt;itemData label="Locations person is active in" value="~ Dallas, Texas ~ USA" /&gt;</code>	Repeating tag
UID	<code>&lt;itemData label="Source Reference ID" value="487273" /&gt;</code>	
Status	<code>&lt;dsQuestion data_source="World-Check" risk="High Risk"&gt;</code>	
Description	<code>&lt;itemData label="Additional Information" value="[BIOGRAPHY] Member of the Shura Council (Nov 2002 - Dec 2006) (Dec 2006 - ). Chairman of Vision 3, an alliance formed by Gulf Finance House, Ithmaar Bank and Abu Dhabi Investment House (Jan 2009 - ). Chairman of First Energy Bank (Jul 2008 - ). Vice Chairman of Khaleeji Commercial Bank ( - Feb 2010). Chairman of Gulf Finance House (reported 2010). Deputy Chairman of Masraf Al Rayan (bank) (reported 2008). Chairman of Bahrain Financial Harbour (reported 2008). [IDENTIFICATION] Married. [REPORTS] To be determined." /&gt;</code>	

## XML and SOAP Document Samples

Additional document examples are available in the RealTimeSamples directory of the Sentinel Libraries zip file. The file is available in the help – download files page.

## **Batch Processing**

### **Real Time Service - SOAP INPUT**

```
<env:Envelope xmlns:env=
  http://schemas.xmlsoap.org/soap/envelope/ xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<env:Body env:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <submitApplicant>
    <username xsi:type="xsd:string">???
```