

CSC Financial Web Services Developer Guide

Last revised: April 20, 2015

Published: April 12, 2006



CORPORATION SERVICE COMPANY®

Copyright ©2015 Corporation Service Company. All Rights Reserved.

This document contains proprietary information and is provided to recipient subject to the mutual non-disclosure agreement signed by recipient and Corporation Service Company. Recipient shall not copy or distribute this document without written permission of Corporation Service Company.



Contents

CSC Financial Web Services	5
Welcome	5
1. System Overview	5
2. Architecture	6
2.1. General Considerations	6
3. Programming Guide	6
3.1. CSC Financial Web Services (FWS) Essentials	6
3.2. Retrieving Reference Information	7
3.3. Retrieving Order Information and Attachments	7
3.4. Submitting Online Searches	8
3.5. Submitting Offline Searches	9
3.6. Submitting Filings	9
4. API Reference	10
4.1. CSC Financial Namespace	10
4.2. General structure of the “params” parameter	10
4.3. Arrays	11
4.4. Key-Value Pairs	12
4.5. Sending File Attachments	12
4.6. Common Parameters	12
4.7. Search CSC Financial Web Services	13
4.7.1. AvailableSearches	13
4.7.2. GetJurisdictions	13
4.7.3. SubmitOnlineSearch	14
4.7.4. GetSummaryResults	16
4.7.5. GetDetailResults	16
4.7.6. GetReport	17
4.7.7. GetDocuments	18
4.7.8. SetSelectedSummaryResults	18
4.7.9. SetSelectedDetailResults	19
4.7.10. SubmitOfflineSearch	19



4.7.11 Examples of Offline Searches.....	27
Fixture Search ("Fixture")	27
Four Part Search ("FourPart")	28
Articles of Incorporation ("ArticlesofIncorporation").....	29
Certificate of Existence ("CertificateOfExistence").....	30
Corporate Annual Report ("CorporateAnnualReport").....	31
Fictitious Business Name ("FictitiousBusinessName")	32
Good Standing Certificate ("GoodStandingCertificate")	33
Verify Business Good Standing by Company Name ("Corp")	34
Federal Tax Liens ("FederalTaxLiens")	35
Judgement Liens ("JudgementLiens")	36
State Tax Liens ("StateTaxLiens")	37
Corporate Tracking ("CorporateTracking").....	38
Debtor Tracking ("DebtorTracking")	39
Civil Litigation Search ("CivilLitigation")	40
Federal Civil Litigation Search ("FederalCivilLitigation")	41
Federal Criminal Litigation Search ("FederalCriminalLitigation").....	42
Criminal Litigation Search ("CriminalLitigation")	43
Bankruptcy Search ("Bankruptcy")	44
Business Credit Report ("CreditReport").....	45
Flood Certification Search ("Flood")	46
Suits, Liens and Judgements by Company Name ("SLJ").....	47
4.8. Filing CSC Financial Web Services	48
4.8.1. CreateFiling	48
4.8.2. ValidateFiling	51
4.8.3. UpdateFiling	51
4.8.4. ApproveOrder	52
4.8.5. ContinueFiling	52
4.8.6. TerminateFiling	52



4.9. Document Center CSC Financial Web Services	53
4.9.1. Order Status Codes	53
4.9.2. GetOrderInformation	53
4.9.3. GetChangedOrders	55
4.9.4. UploadAttachment	56
4.9.5. GetAttachment	56



CSC Financial Web Services

Welcome

Welcome to the CSC Financial Web Services (CSC-FWS) Software Development Kit (SDK). This SDK provides the information necessary to create websites or applications that can integrate with the services that CSC offers.

This SDK is divided into the following sections:

1. System Overview

The overview section provides information on what you need to begin using FWS in your application.

2. Architecture

The architecture section describes the services that are currently provided through FWS and gives information about potential development considerations.

3. Programming Guide

The programming guide contains a detailed description of how to submit requests to DWS and the data that the service returns, as well as how to handle errors.

4. API Reference

The API reference contains a detailed reference to the syntax of all possible calls to the FWS.

1. System Overview

CSC Financial Web Services provides a programmatic way of performing a majority of the tasks that can be accomplished directly through the cscfinancialonline.com website.

Through FWS, you can:

- Run an online UCC, Corporate, "Suits, Liens and Judgements" (SLJ), or Corporate Name Verification (CNV) search
- Order an offline search
- Get detailed results from a search
- Get documents from a search, if available
- Generate reports from search results
- Retrieve search results from previous searches
- Create a UCC filing
- Update a UCC filing before it is approved
- Validate a filing
- Approve a filing
- Get acknowledgment information returned by the jurisdiction
- Get a PDF copy of the acknowledgment



2. Architecture

2.1. General Considerations

The main advantage to using Web Services is that it supports standardized protocols. By supporting these standards (XML and SOAP over HTTP and HTTPS), CSC reduces the amount of work that a developer must do to integrate with the CSC Financial Web application and opens the door to using a wide variety of existing tools and toolkits.

The other advantage to Web Services is that it is both language and platform independent. There are no restrictions on what type of system is used to submit requests and process results. Almost every operating system and programming language has the capability of issuing SOAP requests and parsing XML results, so it is only a matter of learning the syntax to perform these operations under any given system.

The following subsections describe common use cases for CSC Financial Web Services.

3. Programming Guide

3.1. CSC Financial Web Services (FWS) Essentials

There are several concepts to understand before you can begin writing applications that use FWS.

Instead of using Web Service Description Language (WSDL) to create a strongly-typed, message-passing paradigm, CSC opted to create a more loosely-typed system that offers greater flexibility than WSDL could provide. All methods in the FWS take an XML structure as a parameter and return an XML structure. The XML structure contains the names and values for the parameters to be passed into the method. This type of paradigm comes with two major benefits: updates to the interface for an FWS method do not break clients' existing code, and there is greater flexibility for CSC to customize interfaces for clients with special needs. WSDL, however, is not completely ignored. The FWS WSDL file still lists all the calls that can be made, since the calls themselves are not expected to be changed.

Once you are set up with CSC and ready to write code, point your favorite Integrated Development Environment (IDE) to our testing site (<http://eservices-test.diligenz.com>) and you are ready to begin developing and testing. To move to our production environment, remove the “-test” part of the URL.

To give you access to Web Services, CSC provides a globally unique identifier (GUID) that acts as a substitute for a login and password. This GUID should be easily changeable from the client application since different GUIDs may be given for testing and production. CSC also requires the contact number to be passed with every call so we can track the activities back to the appropriate user.

3.2. Retrieving Reference Information

The search databases at CSC are constantly being updated, so it is important for your application to be able to determine whether a given state and search type is available. Likewise, jurisdictional information for states changes often. This type of information should be cached on the client side on a daily basis. Figure 1 is an example process for retrieving and caching this data.

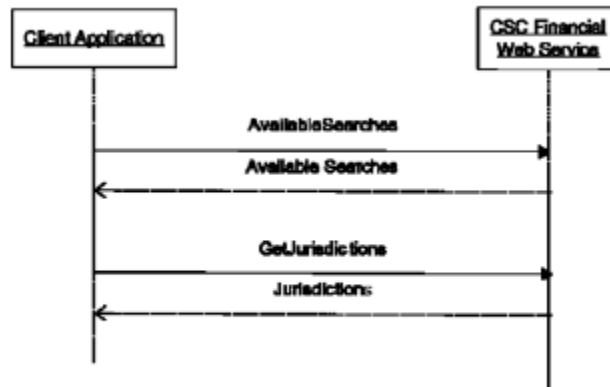


Figure 1. Example process for retrieving and caching reference data.

3.3. Retrieving Order Information and Attachments

It is often desirable to retrieve status information or documents pertaining to an existing pending order. Figure 2 demonstrates how this can be accomplished. First, you call `GetChangedOrders` to retrieve all OrderIDs that have changed since a specified date. Using that list, you can then get information about those orders or download any attachments from them.

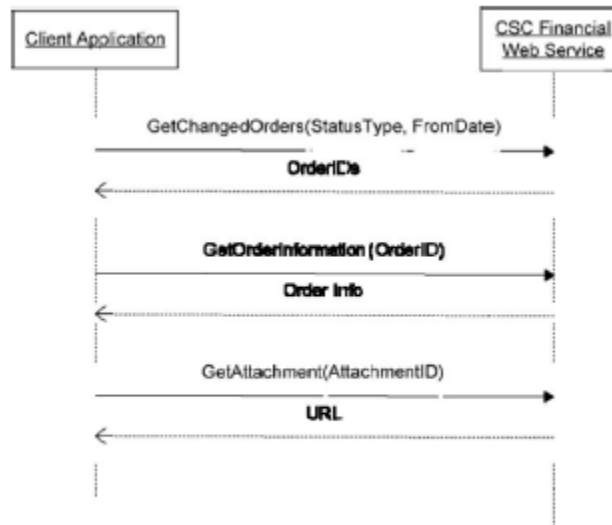


Figure 2. Example process for retrieving information and attachments for existing orders.

3.4. Submitting Online Searches

An online search is any search on the cscfinancialonline.com site that instantly returns results. Figure 3 shows the life cycle of a typical online search. First, the client application submits an online search request. CSC returns a summary report of the results found. The client has the option of either calling `SetSelectedSummaryResults` and then `GetDetailResults` or just calling `GetDetailResults` and passing the selected results along with the request. Once `GetDetailResults` is called, the search is locked and the detail results cannot be changed. After the detail results are retrieved, the client has the ability to either request documents or generate a summary report for selected rows.

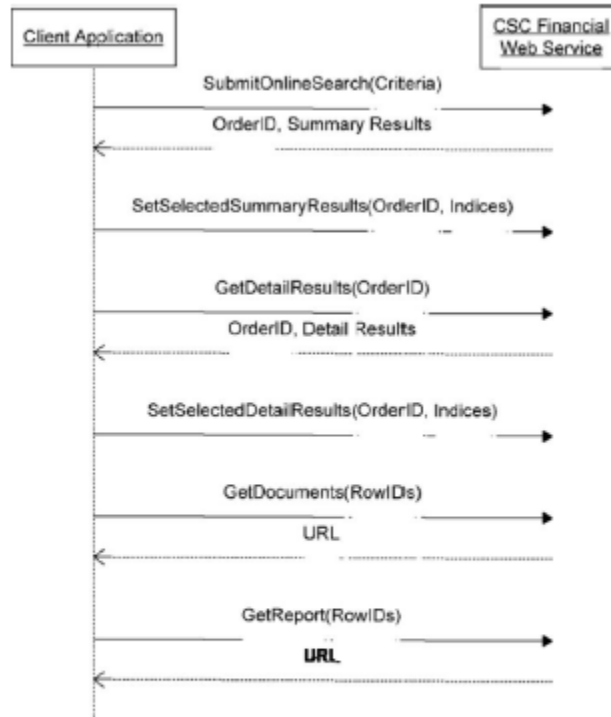


Figure 3. General procedure when running online searches.

3.5. Submitting Offline Searches

An offline search order should be requested when you select a search type or jurisdiction for which data is not available online. Once an offline search request has been submitted (Figure 4), it is processed by CSC. Once the order is completed, you can use the process described in section 3.3 to retrieve the completed results from the order.

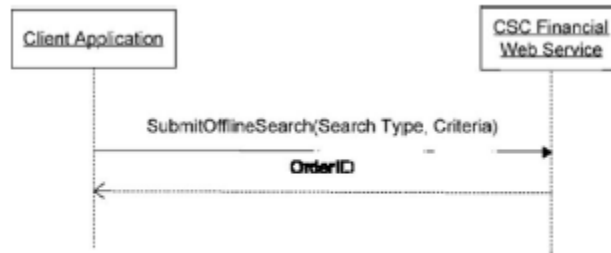


Figure 4. Example process for submitting offline searches.

3.6. Submitting Filings

Figure 5 shows the life cycle of a typical filing. First, the client calls CreateFiling, passing in an XML filing document. This creates an unapproved order in the cscfinancialonline.com website. The client can then call ValidateFiling to see if there are any validation messages for the filing. If the filing has no serious validation errors, the client can call ApproveOrder to send the filing out for processing. At any time, the client can call GetOrderInformation to check on the status of the order, or use the process as described in section 3.3 to retrieve the attachments for the order.

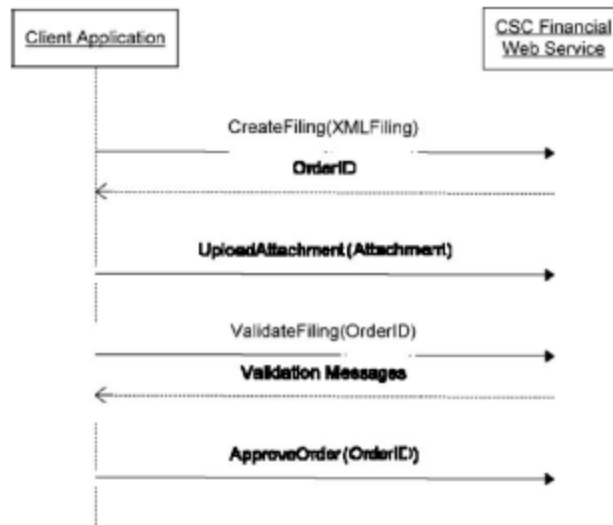


Figure 5. Example process for submitting filings.



4. API Reference

4.1. CSC Financial Namespace

All CSC Financial related elements passed in or returned will be under the “http://eservices.diligenz.com/” XML namespace. It can be assumed that any XML snippets in the examples are in that namespace.

4.2. General structure of the “params” parameter

The XML document that clients pass into the Web methods will have a common structure that adheres to the following schema:

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:element name="params">
    <xs:annotation>
      <xs:documentation>Collection of parameters to pass in</xs:documentation> </xs:annotation>
    <xs:complexType>
      <xs:all maxOccurs="unbounded">
        <xs:element ref="param"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <xs:element name="param">
    <xs:annotation>
      <xs:documentation>A parameter for the method</xs:documentation> </xs:annotation>
    <xs:complexType mixed="true">
      <xs:choice minOccurs="0">
        <xs:any namespace="##any" processContents="skip"/>
        <xs:element ref="array"/>
      </xs:choice>
      <xs:attribute name="name" type="xs:string" use="required"/>
    </xs:complexType>
  </xs:element>
  <xs:element name="array">
    <xs:annotation>
      <xs:documentation>A collection of items</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:all maxOccurs="unbounded">
        <xs:element ref="item"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <xs:element name="item">
    <xs:annotation>
      <xs:documentation>An item in an array</xs:documentation>
    </xs:annotation>
  </xs:element>
</xs:schema>
```



The following is an example of an XML document that uses the schema:

```
<params>
  <param name="loginGuid">12345678-90AB-CDEF-0123-4567890ABCDE</param>
  <param name="contactNo">1</param>
  <param name="someXmlDoc">
    <Document>
      <Test>This is a test xml doc</Test>
      <Test>Hello world</Test>
    </Document>
  </param>
  <param name="someDate">2004-04-28T15:48:33</param>
  <param name="anArrayOfValues">
    <array>
      <item>value1</item>
      <item>value2</item>
      <item>value3</item>
    </array>
  </param>
</params>
```

Here is what the message looks like for a SOAP request for a UCC search for "TCBY" in CA:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/">
  <soapenv:Body>
    <SubmitOnlineSearch xmlns="http://eservices.diligenz.com/">
      <params>
        <param name="loginGuid">Insert GUID</param>
        <param name="contactNo">Insert contact number</param>
        <param name="stateCD">CA</param>
        <param name="type">UCC</param>
        <param name="lapsedTerm">true</param>
        <param name="companyName">TCBY</param>
        <param name="showAllResults">true</param>
        <param name="references">
          <array>
            <item>
              <key>app_id1</key>
              <value>1705882</value>
            </item>
          </array>
        </param>
      </params>
    </SubmitOnlineSearch>
  </soapenv:Body>
</soapenv:Envelope>
```

The structure allows you to enter as many or as few parameters into the method as you wish.

4.3. Arrays

Parameters that accept arrays are required to be in a specific format that looks like this:



```
<array>
<item>
  <key>app_id1</key>
  <value>1705882</value>
</item>
</array>
```

4.4. Key-Value Pairs

Parameters that accept arrays of key/value pairs are required to be in a specific format that looks like this:

```
<param name="references">
  <array>
    <item>
      <key>BillingRef1</key>
      <value>test billing ref</value>
    </item>
    <item>
      <key>My Special billing ref</key>
      <value>testing 1234</value>
    </item>
  </array>
</param>
```

4.5. Sending File Attachments

Web methods that accept file attachments require the attachment to be in a specific format. Any file attachment needs to be Base64-encoded. A MIME type (or content type) also needs to be specified in a separate parameter in order to determine the type of the file. CSC accepts the following content types for upload:

- txt: text/plain
- pdf: application/pdf
- xml: text/xml
- tif: image/tiff
- doc: application/msword

4.6 Common Parameters

All methods require the following two parameters which are used for authentication and identification:

- loginGuid – a GUID assigned to each account wishing to use CSC Financial Web Services
- contactNo – the contact to use when performing operations



4.7. Search CSC Financial Web Services

4.7.1. AvailableSearches

This method returns a list of available searches with “thru dates” and online status. When this method is used with no parameters, it returns a listing of all searches possible in the CSC system that can return results.

Parameter	Description	Required?	Value
stateCD	Filters the list to only return the searches available for the state abbreviation provided	Optional	A two-letter state abbreviation
searchType	Filters the list to only return the searches available for the search type provided	Optional	UCC, Corp, SLJ or CNV
isOnline	Filters the list by online status	Optional	True or False

Output is an XML file consisting of:

- stateCD – two-character state code
- searchType – the type of search available (UCC, Corp, SLJ, CNV)
- thruDate – the through date of the state data
- isOnline – indicates if the state is online or offline
- thirdParty – indicates if the data is provided by a third party or directly from the state

4.7.2. GetJurisdictions

This method returns a list of jurisdictions to use within the CSC system. These jurisdictions are very important when placing orders. Using the wrong jurisdiction could send an offline request to the wrong county, which would result in additional costs incurred. If called with no parameters, all jurisdictions are returned.

Parameter	Description	Required?	Value
stateCD	Filters the list to only return the jurisdictions available for the state abbreviation provided	Optional	A two-letter state abbreviation

Output is an XML file consisting of:

- jurisdictionName – name of the jurisdiction as it appears in the CSC database
- jurisdictionID – the unique ID assigned to the jurisdiction
- stateCD – the two-character state code

The XML document that clients pass into the Web methods will have a common structure that adheres to the following schema:

```
<?xml version="1.0" encoding="UTF-8" ?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:eser="http://eservices.diligenz.com/">
  <soapenv:Header />
  <soapenv:Body>
    <eser:GetJurisdictions xmlns="http://eservices.diligenz.com/">
      <params>
        <param name="loginGuid" xmlns="http://eservices.diligenz.com/">Insert GUID</param>
        <param name="contactNo" xmlns="http://eservices.diligenz.com/">Insert contact number</param>
        <param name="stateCD" xmlns="http://eservices.diligenz.com/">NC</param>
      </params>
    </eser:GetJurisdictions>
  </soapenv:Body>
</soapenv:Envelope>
```



4.7.3. SubmitOnlineSearch

This method submits a search request and finds online search results. With this call, the user expects to receive search results immediately. If the database is offline, the method will throw an exception. The type of search run is determined by a combination of the “type” parameter and the criteria parameters chosen.

Parameter	Description	Required?	Value
type	The type of search to run	Required	UCC, Corp, SLJ, or CNV
references	A list of billing references to use for this search	It is required to have at least one billing reference	An xml element of the form: <array> <item> <key>BillingRef1</key> <value>testing</value> </item> </array>
stateCD	The state in which to run the search. The default (SOS) jurisdiction is assumed.	Required	A two-letter state abbreviation
lapsedTerm	True to run a lapsed/term search	Optional	True or False
useThirdParty	When true, the search is required to run on a third-party database. When false, the search is required not to run on a third-party database. If not specified, the search will run on a third-party database only if no others are available. Throws an exception if it is not valid in this context.	Optional	True or False
docNumber	The document number to use for the search	This is a criteria parameter. At least one of these is required.	Any string
firstName	The first name of the debtor to use for the search	This is a criteria parameter. At least one of these is required.	Any string
lastName	The last name of the debtor to use for the search	This is a criteria parameter. At least one of these is required.	Any string
companyName	The company name to use for the search	This is a criteria parameter. At least one of these is required.	Any string
securedPartyName	The secured party name to use	This is a criteria parameter. At least one of these is required.	Any string
businessID	The unique state business ID (usually the Org ID) to use for the search	This is a criteria parameter. At least one of these is required.	Any string



The following are accepted search types:

Type of Search	Value for "type" parameter	Required parameters
UCC Search by Company Name	UCC	companyName
UCC Search by First Name, Last Name	UCC	firstName lastName
UCC Search by Document Number	UCC	docNumber
UCC Secured Party Search	UCC	securedPartyName
Verify Business Good Standing by BusinessID	Corp	businessID
Verify Business Good Standing by Company Name	Corp	companyName
Verify Business Good Standing by First Name, Last Name	Corp	firstName lastName
Suits, Liens and Judgements by Company Name	SLJ	companyName
Corporate Name Verification	CNV	companyName

Output:

- orderID – A unique CSC order number which can be used to reference the search later

The return document conforms to the following schema:

```
<?xml version='1.0' encoding='UTF-8'>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <SubmitOnlineSearchResponse xmlns="http://eservices.diligenz.com/">
      <SubmitOnlineSearchResult>
        <return>
          <orderInfo>
            <OrderID>98364684</OrderID>
          </orderInfo>
          <summaryResults>
            <ID>2800391110</ID>
            <Selected>false</Selected>
            <Count>1</Count>
            <DebtorName>T.C.B.Y.</DebtorName>
          </summaryResults>
          <summaryResults>
            <ID>2800391107</ID>
            <Selected>false</Selected>
            <Count>6</Count>
            <DebtorName>TCBY</DebtorName>
          </summaryResults>
        </return>
      </SubmitOnlineSearchResult>
    </SubmitOnlineSearchResponse>
  </soap:Body>
</soap:Envelope>
```



4.7.4. GetSummaryResults

This method gets the summary results from a search that has been submitted. This is not available for offline search requests.

Parameter	Description	Required?	Value
orderId	The orderId of the search	Required	A positive integer

Output:

- An XML file containing all of the Summary results (includes the ID column that clients will use to select results)

4.7.5. GetDetailResults

This method gets the detail results from a search that has been submitted. If you call this method without calling GetSummaryResults you will get the same result as if all summary records were selected. Calling this method after calling GetSummaryResults without passing any selected results will return no records.

Parameter	Description	Required?	Value
orderId	The orderId of the search	Required	A positive integer
selectedResults	An integer array of summary row IDs to select before getting detail results	Optional	An xml element of the form: <array> <item>val1</item> <item>val2</item> <item>val3</item> </array>

Output:

- An XML file containing all of the detail results (includes the ID column that clients will use to select additional actions)



4.7.6. GetReport

Once you get detail results and optionally select a subset of those, you can call this method to generate a report of the results.

Parameter	Description	Required?	Value
orderId	The orderId of the search	Required	A positive integer
type	The report type to generate	Required	One of the following values: <ul style="list-style-type: none">• Summary• Excel
selectedResults	An integer array of detail row IDs to select before generating the results	Optional	An xml element of the form: <array> <item>val1</item> <item>val2</item> <item>val3</item> </array>
selectAll	When true, selects all detail results for the report	Optional	True or False Default: False

Output:

- A URL to the generated report file.

The return document conforms to the following schema:

```
<?xml version='1.0' encoding='UTF-8'>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetReportResponse xmlns="http://eservices.diligenz.com/">
      <GetReportResult>
        <return>
          <AttachmentURL>http://eservices.diligenz.com/docs/36981271-084549041015-29775-SLJSummary.pdf</AttachmentURL>
        </return>
      </GetReportResult>
    </GetReportResponse>
  </soap:Body>
</soap:Envelope>
```



4.7.7. GetDocuments

This method gets documents from a detail report.

Parameter	Description	Required?	Value
orderId	The orderId of the search	Required	A positive integer
selectedResults	An integer array of detail row IDs to select before requesting documents	Optional	An xml element of the form: <array> <item>val1</item> <item>val2</item> <item>val3</item> </array>

Output:

- A URL to the generated documents.

The return document conforms to the following schema:

```
<?xml version='1.0' ?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetDocumentsResponse xmlns="http://eservices.diligenz.com/">
      <GetDocumentsResult>
        <return>
          <ReportURL>http://eservices.diligenz.com/docs/31417314-084812041015-26425-Summary.pdf</ReportURL>
        </return>
      </GetDocumentsResult>
    </GetDocumentsResponse>
  </soap:Body>
</soap:Envelope>
```

The resulting PDF file at the URL returned can be used to download the PDF report.

4.7.8. SetSelectedSummaryResults

This method selects the summary results for an online search.

Parameter	Description	Required?	Value
orderId	The orderId of the search	Required	A positive integer
selectedResults	An integer array of summary row IDs to select	Required	An xml element of the form: <array> <item>val1</item> <item>val2</item> <item>val3</item> </array>

Output:

- None



4.7.9. SetSelectedDetailResults

This method selects the detail results for an online search.

Parameter	Description	Required?	Value
orderId	The orderId of the search	Required	A positive integer
selectedResults	An integer array of detail row IDs to select	Required	An xml element of the form: <array> <item>val1</item> <item>val2</item> <item>val3</item> </array>

Output:

- None

4.7.10. SubmitOfflineSearch

This method submits an offline search request. When submitting an offline search, results are not retrieved instantly. Instead, an order is created and submitted for CSC to fulfill.

Parameter	Description	Required?	Value
type	The type of search to run	Required	See table below for valid search types.
references	A list of billing references to use for this search	It is required to have at least one billing reference.	An xml element of the form: <array> <item> <key>BillingRef1</key> <value>testing</value> </item> </array>
stateCD	The state in which to run the search. If this parameter is supplied without a jurisdiction parameter, state-level (SOS) is assumed.	Required	A two-letter state abbreviation
jurisdictionName	The name of the jurisdiction in which to run the search. JurisdictionID is preferred over JurisdictionName because the parameter can easily be misspelled.	Optional	The exact name of the jurisdiction



jurisdictionID	The ID of the jurisdiction to search. IDs can be retrieved using GetJurisdictions	Optional. If not provided, SOS is assumed.	Positive integers
lapsedTerm	True to run a lapsed/term search.	Optional	True or False
lapsedTermYear	Specifies how many years for lapsed/term search	Optional. If lapsedTerm parameter is supplied and is True then it needs to be supplied	One of the following values: <ul style="list-style-type: none">• One• All
startDate	Specifies a specific start date for the search results	Optional	Any string in date format
deliveryMethodType	Specifies search results delivery method type	Optional	One of the following values: <ul style="list-style-type: none">• Email• Fax
deliveryMethodValue	Specifies delivery method value for search results	Optional. If deliveryMethodType is supplied, then this parameter needs to be supplied.	Any string in either email format or fax number format
secondaryDeliveryMethod	Specifies secondary delivery method value for search results	Optional	One of the following values: <ul style="list-style-type: none">• None• UsMail• FedexNextdayAir• FedexNextdaySaver• FedexSecondaryAir
useThirdParty	When true, the search is required to run on a third-party database. When false, the search is required not to run on a third-party database. If not specified, the search will run on a third-party database only if no others are available. Throws an exception if it is not valid in this context.	Optional	True or False
docNumber	The document number to use for the search	This is a criteria parameter. At least one of these is required.	Any string



firstName	The first name of the debtor to use for the search	This is a criteria parameter. At least one of these is required.	Any string
lastName	The last name of the debtor to use for the search	This is a criteria parameter. At least one of these is required.	Any string
companyName	The company name to use for the search	This is a criteria parameter. At least one of these is required.	Any string
securedPartyName	The secured party name to use	This is a criteria parameter. At least one of these is required.	Any string
businessID	The businessID or organizationalID to use for the search	This is a criteria parameter. At least one of these is required.	Any string
debtorName	The name of the debtor	This is a criteria parameter. At least one of these is required.	Any string
fictitiousBusinessName	The fictitious business name	This is a criteria parameter. At least one of these is required.	Any string
ownerName	The owner name	This is a criteria parameter. At least one of these is required.	Any string
document	The document	This is a criteria parameter. At least one of these is required.	Any string
certified	True to order a certified search	This is a criteria parameter. At least one of these is required.	True or False



subject	The subject of the search	This is a criteria parameter. At least one of these is required.	Any string
criteria	The criteria to use for the search	This is a criteria parameter. At least one of these is required.	Any string
openType	Used for litigation searches	This is a criteria parameter. At least one of these is required.	Open Closed Open/Closed
litigantType	Used for litigation searches	This is a criteria parameter. At least one of these is required.	Plaintiff Defendant Plaintiff/Defendant
docket	Used for litigation searches	This is a criteria parameter. At least one of these is required.	True or False
complaint	Used for litigation searches	This is a criteria parameter. At least one of these is required.	True or False
lastActionFiled	Used for litigation searches	This is a criteria parameter. At least one of these is required.	True or False
dateOfBirth	The date of birth	This is a criteria parameter. At least one of these is required.	Any string
SSN	The social security number	This is a criteria parameter. At least one of these is required.	Any string
address	The address	This is a criteria parameter. At least one of these is required.	Any string



city	The city	This is a criteria parameter. At least one of these is required.	Any string
state	The state	This is a criteria parameter. At least one of these is required.	A two-letter state code
country	The country	This is a criteria parameter. At least one of these is required.	Any string
county	The county	This is a criteria parameter. At least one of these is required.	Any string
borrower	The borrower	This is a criteria parameter. At least one of these is required.	Any string
secondBorrower	The second borrower	This is a criteria parameter. At least one of these is required.	Any string
loanNumber	The loan number	This is a criteria parameter. At least one of these is required.	Any string
taxNumber	The tax number	This is a criteria parameter. At least one of these is required.	Any string
legalDesc	The legal description	This is a criteria parameter. At least one of these is required.	Any string



The following are accepted offline search types and their required parameters:

Type of Search	Value for “type” parameter	Required parameters
UCC Search by Company Name	UCC	companyName optional: lapsedTerm
UCC Search by First Name, Last Name	UCC	firstName lastName
UCC Search by Document Number	UCC	docNumber
UCC Secured Party Search	UCC	securedPartyName
Fixture Search	Fixture	debtorName
Four Part Search	FourPart	companyName
Articles of Incorporation	ArticlesOfIncorporation	companyName
Certificate of Existence	CertificateOfExistence	companyName
Corporate Annual Report	CorporateAnnualReport	companyName
Fictitious Business Name Search	FictitiousBusinessName	companyName fictitiousBusinessName ownerName
Good Standing Certificate	GoodStandingCertificate	companyName
Verify Business Good Standing by Organizational ID	Corp	businessID
Verify Business Good Standing by Company Name	Corp	companyName
Verify Business Good Standing by First Name, Last Name	Corp	firstName lastName
Federal Tax Liens	FederalTaxLiens	debtorName
Judgement Liens	JudgementLiens	debtorName
State Tax Liens	StateTaxLiens	debtorName
Debtor Tracking	DebtorTracking	companyName criteria stateCD
Corporate Tracking	CorporateTracking	companyName state (optional) businessID



Federal Civil Litigation Search	FederalCivilLitigation	subject criteria openType litigantType docket complaint lastActionFiled
Civil Litigation Search	CivilLitigation	subject criteria openType litigantType docket complaint lastActionFiled
Federal Criminal Litigation Search	FederalCriminalLitigation	subject criteria openType litigantType docket complaint lastActionFiled
Criminal Litigation Search	CriminalLitigation	subject criteria openType litigantType docket complaint lastActionFiled dateOfBirth SSN address city state country
Bankruptcy Search	Bankruptcy	debtorName
Business Credit Report	CreditReport	companyName address city stateCD country



Flood Certification Search	Flood	companyName borrower (optional) secondBorrower (optional) address city stateCD county loanNumber (optional) taxNumber (optional) legalDesc (optional)
Suits, Liens and Judgements by Company Name	SLJ	companyName
UCC, Federal Tax Lien	2Part-001	companyName debtorName
UCC, Judgement Lien	2Part-002	companyName debtorName
UCC, State Tax Lien	2Part-003	companyName debtorName
Federal Tax Lien, Judgement Lien	2Part-004	debtorName
Federal Tax Lien, State Tax Lien	2Part-005	debtorName
Judgement Lien, State Tax Lien	2Part-006	debtorName
UCC, Federal Tax Lien, Judgement Lien	3Part-007	companyName debtorName
UCC, Federal Tax Lien, State Tax Lien	3Part-008	companyName debtorName
UCC, Judgement Lien, State Tax Lien	3Part-009	companyName debtorName
Federal Tax Lien, Judgement Lien, State Tax Lien	3Part-010	debtorName



4.7.11 Examples of Offline Searches

Fixture Search (“Fixture”)

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<SubmitOfflineSearch xmlns="http://eservices.diligenz.com/">
<params>
<param name="contactNo" xmlns="http://eservices.diligenz.com/">Insert Contact No</param>
<param name="loginGuid" xmlns="http://eservices.diligenz.com/">Insert GUID</param>
<param name="type" xmlns="http://eservices.diligenz.com/">Fixture</param>
<param name="stateCD" xmlns="http://eservices.diligenz.com/">WA</param>
<param name="debtorName" xmlns="http://eservices.diligenz.com/">Starbucks</param>
<param name="expedite" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field--> <param name="deliveryMethodType" xmlns="http://eservices.diligenz.com/">Email</param>
<!--optional field--> <param name="deliveryMethodValue"
xmlns="http://eservices.diligenz.com/">sravani.tummepalli@cscglobal.com</param>
<!--optional field-->
<!--optional field--> <param name="rush" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field-->
<param name="specialInstructions" xmlns="http://eservices.diligenz.com/">Special Instructions added by the
client</param>
<!--optional field-->
<param name="references" xmlns="http://eservices.diligenz.com/">
<array>
<item> <key>MyRef1</key> <value>E-Service</value> </item>
</array>
</param>
</params>
</SubmitOfflineSearch>
</soap:Body> </soap:Envelope>
```



Four Part Search ("FourPart")

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<SubmitOfflineSearch xmlns="http://eservices.diligenz.com/">
<params>
<param name="contactNo" xmlns="http://eservices.diligenz.com/">Insert Contact No</param>
<param name="loginGuid" xmlns="http://eservices.diligenz.com/">Insert GUID</param>
<param name="type" xmlns="http://eservices.diligenz.com/">FourPart</param>
<param name="stateCD" xmlns="http://eservices.diligenz.com/">CA</param>
<param name="companyName" xmlns="http://eservices.diligenz.com/">Starbucks</param>
<param name="expedite" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field--> <param name="deliveryMethodType" xmlns="http://eservices.diligenz.com/">Email</param>
<!--optional field--> <param name="deliveryMethodValue"
xmlns="http://eservices.diligenz.com/">sravani.tummepalli@cscglobal.com</param>
<!--optional field-->
<!--optional field--> <param name="rush" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field-->
<param name="specialInstructions" xmlns="http://eservices.diligenz.com/">Special Instructions added by the
client</param>
<!--optional field-->
<param name="references" xmlns="http://eservices.diligenz.com/">
<array>
<item> <key>MyRef1</key> <value>E-Service</value> </item>
</array>
</param>
</params>
</SubmitOfflineSearch>
</soap:Body> </soap:Envelope>
```



Articles of Incorporation (“ArticlesofIncorporation”)

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<SubmitOfflineSearch xmlns="http://eservices.diligenz.com/">
<params>
<param name="contactNo" xmlns="http://eservices.diligenz.com/">Insert Contact No</param>
<param name="loginGuid" xmlns="http://eservices.diligenz.com/">Insert GUID</param>
<param name="type" xmlns="http://eservices.diligenz.com/">ArticlesOfIncorporation</param>
<param name="stateCD" xmlns="http://eservices.diligenz.com/">CA</param>
<param name="companyName" xmlns="http://eservices.diligenz.com/">Starbucks</param>
<param name="expedite" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field--> <param name="deliveryMethodType" xmlns="http://eservices.diligenz.com/">Email</param>
<!--optional field--> <param name="deliveryMethodValue"
xmlns="http://eservices.diligenz.com/">sravani.tummepalli@cscglobal.com</param>
<!--optional field-->
<!--optional field--> <param name="rush" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field-->
<param name="specialInstructions" xmlns="http://eservices.diligenz.com/">Special Instructions added by the
client</param>
<!--optional field-->
<param name="references" xmlns="http://eservices.diligenz.com/">
<array>
<item> <key>MyRef1</key> <value>E-Service</value> </item>
</array>
</param>
</params>
</SubmitOfflineSearch>
</soap:Body> </soap:Envelope>
```

**Certificate of Existence (“CertificateOfExistence”)**

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<SubmitOfflineSearch xmlns="http://eservices.diligenz.com/">
<params>
<param name="contactNo" xmlns="http://eservices.diligenz.com/">Insert Contact No</param>
<param name="loginGuid" xmlns="http://eservices.diligenz.com/">Insert GUID</param>
<param name="type" xmlns="http://eservices.diligenz.com/">CertificateOfExistence</param>
<param name="stateCD" xmlns="http://eservices.diligenz.com/">CA</param>
<param name="companyName" xmlns="http://eservices.diligenz.com/">Starbucks</param>
<param name="expedite" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field--> <param name="deliveryMethodType" xmlns="http://eservices.diligenz.com/">Email</param>
<!--optional field--> <param name="deliveryMethodValue"
xmlns="http://eservices.diligenz.com/">sravani.tummepalli@cscglobal.com</param>
<!--optional field-->
<!--optional field--> <param name="rush" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field-->
<param name="specialInstructions" xmlns="http://eservices.diligenz.com/">Special Instructions added by the
client</param>
<!--optional field-->
<param name="references" xmlns="http://eservices.diligenz.com/">
<array>
<item> <key>MyRef1</key> <value>E-Service</value> </item>
</array>
</param>
</params>
</SubmitOfflineSearch>
</soap:Body> </soap:Envelope>
```



Corporate Annual Report ("CorporateAnnualReport")

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<SubmitOfflineSearch xmlns="http://eservices.diligenz.com/">
<params>
<param name="contactNo" xmlns="http://eservices.diligenz.com/">Insert Contact No</param>
<param name="loginGuid" xmlns="http://eservices.diligenz.com/">Insert GUID</param>
<param name="type" xmlns="http://eservices.diligenz.com/">CorporateAnnualReport</param>
<param name="stateCD" xmlns="http://eservices.diligenz.com/">CA</param>
<param name="companyName" xmlns="http://eservices.diligenz.com/">Starbucks</param>
<param name="expedite" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field--> <param name="deliveryMethodType" xmlns="http://eservices.diligenz.com/">Email</param>
<!--optional field--> <param name="deliveryMethodValue"
xmlns="http://eservices.diligenz.com/">sravani.tummepalli@cscglobal.com</param>
<!--optional field-->
<!--optional field--> <param name="rush" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field-->
<param name="specialInstructions" xmlns="http://eservices.diligenz.com/">Special Instructions added by the
client</param>
<!--optional field-->
<param name="references" xmlns="http://eservices.diligenz.com/">
<array>
<item> <key>MyRef1</key> <value>E-Service</value> </item>
</array>
</param>
</params>
</SubmitOfflineSearch>
</soap:Body> </soap:Envelope>
```



Fictitious Business Name (“FictitiousBusinessName”)

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<SubmitOfflineSearch xmlns="http://eservices.diligenz.com/">
<params>
<param name="contactNo" xmlns="http://eservices.diligenz.com/">Insert Contact No</param>
<param name="loginGuid" xmlns="http://eservices.diligenz.com/">Insert GUID</param>
<param name="type" xmlns="http://eservices.diligenz.com/">FictitiousBusinessName</param>
<param name="stateCD" xmlns="http://eservices.diligenz.com/">CA</param>
<param name="companyName" xmlns="http://eservices.diligenz.com/">BURGER KING OF CALIFORNIA, INC.</param>
<param name="fictitiousBusinessName" xmlns="http://eservices.diligenz.com/">burger</param>
<param name="ownerName" xmlns="http://eservices.diligenz.com/">Sravani</param>
<param name="expedite" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field--> <param name="deliveryMethodType" xmlns="http://eservices.diligenz.com/">Email</param>
<!--optional field--> <param name="deliveryMethodValue"
xmlns="http://eservices.diligenz.com/">sravani.tummepalli@cscglobal.com</param>
<!--optional field-->
<!--optional field--> <param name="rush" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field-->
<param name="specialInstructions" xmlns="http://eservices.diligenz.com/">Special Instructions added by the client</param>
<!--optional field-->
<param name="references" xmlns="http://eservices.diligenz.com/">
<array>
<item> <key>MyRef1</key> <value>E-Service</value> </item>
</array>
</param>
</params>
</SubmitOfflineSearch>
</soap:Body> </soap:Envelope>
```




Good Standing Certificate (“GoodStandingCertificate”)

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<SubmitOfflineSearch xmlns="http://eservices.diligenz.com/">
<params>
<param name="contactNo" xmlns="http://eservices.diligenz.com/">Insert Contact No</param>
<param name="loginGuid" xmlns="http://eservices.diligenz.com/">Insert GUID</param>
<param name="type" xmlns="http://eservices.diligenz.com/">GoodStandingCertificate</param>
<param name="stateCD" xmlns="http://eservices.diligenz.com/">CA</param>
<param name="companyName" xmlns="http://eservices.diligenz.com/">BURGER KING OF CALIFORNIA,
INC.</param>
<param name="expedite" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field--> <param name="deliveryMethodType" xmlns="http://eservices.diligenz.com/">Email</param>
<!--optional field--> <param name="deliveryMethodValue"
xmlns="http://eservices.diligenz.com/">sravani.tummepalli@cscglobal.com</param>
<!--optional field-->
<!--optional field--> <param name="rush" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field-->
<param name="specialInstructions" xmlns="http://eservices.diligenz.com/">Special Instructions added by the
client</param>
<!--optional field-->
<param name="references" xmlns="http://eservices.diligenz.com/">
<array>
<item> <key>MyRef1</key> <value>E-Service</value> </item>
</array>
</param>
</params>
</SubmitOfflineSearch>
</soap:Body> </soap:Envelope>
```



Verify Business Good Standing by Company Name ("Corp")

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<SubmitOfflineSearch xmlns="http://eservices.diligenz.com/">
<params>
<param name="contactNo" xmlns="http://eservices.diligenz.com/">Insert Contact No</param>
<param name="loginGuid" xmlns="http://eservices.diligenz.com/">Insert GUID</param>
<param name="type" xmlns="http://eservices.diligenz.com/">CORP</param>
<param name="stateCD" xmlns="http://eservices.diligenz.com/">WA</param>
<param name="companyName" xmlns="http://eservices.diligenz.com/">burger</param>
<param name="expedite" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field--> <param name="deliveryMethodType" xmlns="http://eservices.diligenz.com/">Email</param>
<!--optional field--> <param name="deliveryMethodValue"
xmlns="http://eservices.diligenz.com/">sravani.tummepalli@cscglobal.com</param>
<!--optional field--> <param name="secondaryDeliveryMethod"
xmlns="http://eservices.diligenz.com/">None</param>
<!--optional field--> <param name="rush" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field-->
<param name="specialInstructions" xmlns="http://eservices.diligenz.com/">Special Instructions added by the
client</param>
<!--optional field-->
<param name="references" xmlns="http://eservices.diligenz.com/">
<array>
<item> <key>MyRef1</key> <value>E-Service</value> </item>
</array>
</param>
</params>
</SubmitOfflineSearch>
</soap:Body> </soap:Envelope>
```



Federal Tax Liens ("FederalTaxLiens")

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <SubmitOfflineSearch xmlns="http://eservices.diligenz.com/">
      <params>
        <param name="contactNo" xmlns="http://eservices.diligenz.com/">Insert Contact No</param>
        <param name="loginGuid" xmlns="http://eservices.diligenz.com/">Insert GUID</param>
        <param name="type" xmlns="http://eservices.diligenz.com/">FederalTaxLiens</param>
        <param name="stateCD" xmlns="http://eservices.diligenz.com/">CA</param>
        <param name="debtorName" xmlns="http://eservices.diligenz.com/">BURGER KING OF CALIFORNIA,
        INC.</param>
        <param name="expedite" xmlns="http://eservices.diligenz.com/">True</param>
        <!--optional field--> <param name="deliveryMethodType" xmlns="http://eservices.diligenz.com/">Email</param>
        <!--optional field--> <param name="deliveryMethodValue"
        xmlns="http://eservices.diligenz.com/">sravani.tummepalli@cscglobal.com</param>
        <!--optional field-->
        <!--optional field--> <param name="rush" xmlns="http://eservices.diligenz.com/">True</param>
        <!--optional field-->
        <param name="specialInstructions" xmlns="http://eservices.diligenz.com/">Special Instructions added by the
        client</param>
        <!--optional field-->
        <param name="references" xmlns="http://eservices.diligenz.com/">
          <array>
            <item> <key>MyRef1</key> <value>E-Service</value> </item>
          </array>
        </param>
      </params>
    </SubmitOfflineSearch>
  </soap:Body> </soap:Envelope>
```



Judgement Liens (“JudgementLiens”)

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<SubmitOfflineSearch xmlns="http://eservices.diligenz.com/">
<params>
<param name="contactNo" xmlns="http://eservices.diligenz.com/">Insert Contact No</param>
<param name="loginGuid" xmlns="http://eservices.diligenz.com/">Insert GUID</param>
<param name="type" xmlns="http://eservices.diligenz.com/">JudgementLiens</param>
<param name="stateCD" xmlns="http://eservices.diligenz.com/">CA</param>
<param name="debtorName" xmlns="http://eservices.diligenz.com/">BURGER KING OF CALIFORNIA,
INC.</param>
<param name="expedite" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field--> <param name="deliveryMethodType" xmlns="http://eservices.diligenz.com/">Email</param>
<!--optional field--> <param name="deliveryMethodValue"
xmlns="http://eservices.diligenz.com/">sravani.tummepalli@cscglobal.com</param>
<!--optional field-->
<!--optional field--> <param name="rush" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field-->
<param name="specialInstructions" xmlns="http://eservices.diligenz.com/">Special Instructions added by the
client</param>
<!--optional field-->
<param name="references" xmlns="http://eservices.diligenz.com/">
<array>
<item> <key>MyRef1</key> <value>E-Service</value> </item>
</array>
</param>
</params>
</SubmitOfflineSearch>
</soap:Body> </soap:Envelope>
```



State Tax Liens (“StateTaxLiens”)

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<SubmitOfflineSearch xmlns="http://eservices.diligenz.com/">
<params>
<param name="contactNo" xmlns="http://eservices.diligenz.com/">Insert Contact No</param>
<param name="loginGuid" xmlns="http://eservices.diligenz.com/">Insert GUID</param>
<param name="type" xmlns="http://eservices.diligenz.com/">StateTaxLiens</param>
<param name="stateCD" xmlns="http://eservices.diligenz.com/">CA</param>
<param name="debtorName" xmlns="http://eservices.diligenz.com/">BURGER KING OF CALIFORNIA,
INC.</param>
<param name="expedite" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field--> <param name="deliveryMethodType" xmlns="http://eservices.diligenz.com/">Email</param>
<!--optional field--> <param name="deliveryMethodValue"
xmlns="http://eservices.diligenz.com/">sravani.tummepalli@cscglobal.com</param>
<!--optional field-->
<!--optional field--> <param name="rush" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field-->
<param name="specialInstructions" xmlns="http://eservices.diligenz.com/">Special Instructions added by the
client</param>
<!--optional field-->
<param name="references" xmlns="http://eservices.diligenz.com/">
<array>
<item> <key>MyRef1</key> <value>E-Service</value> </item>
</array>
</param>
</params>
</SubmitOfflineSearch>
</soap:Body> </soap:Envelope>
```



Corporate Tracking ("CorporateTracking")

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<SubmitOfflineSearch xmlns="http://eservices.diligenz.com/">
<params>
<param name="contactNo" xmlns="http://eservices.diligenz.com/">Insert Contact No</param>
<param name="loginGuid" xmlns="http://eservices.diligenz.com/">Insert GUID</param>
<param name="type" xmlns="http://eservices.diligenz.com/">CorporateTracking</param>
<param name="stateCD" xmlns="http://eservices.diligenz.com/">CA</param>
<param name="companyName" xmlns="http://eservices.diligenz.com/">BURGER KING OF CALIFORNIA,
INC.</param>
<param name="businessID" xmlns="http://eservices.diligenz.com/">0482830</param>
<param name="expedite" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field--> <param name="deliveryMethodType" xmlns="http://eservices.diligenz.com/">Email</param>
<!--optional field--> <param name="deliveryMethodValue"
xmlns="http://eservices.diligenz.com/">sravani.tummepalli@cscglobal.com</param>
<!--optional field-->
<!--optional field--> <param name="rush" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field-->
<param name="specialInstructions" xmlns="http://eservices.diligenz.com/">Special Instructions added by the
client</param>
<!--optional field-->
<param name="references" xmlns="http://eservices.diligenz.com/">
<array>
<item> <key>MyRef1</key> <value>E-Service</value> </item>
</array>
</param>
</params>
</SubmitOfflineSearch>
</soap:Body> </soap:Envelope>
```



Debtor Tracking (“DebtorTracking”)

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<SubmitOfflineSearch xmlns="http://eservices.diligenz.com/">
<params>
<param name="contactNo" xmlns="http://eservices.diligenz.com/">Insert Contact No</param>
<param name="loginGuid" xmlns="http://eservices.diligenz.com/">Insert GUID</param>
<param name="type" xmlns="http://eservices.diligenz.com/">DebtorTracking</param>
<param name="stateCD" xmlns="http://eservices.diligenz.com/">CA</param>
<param name="companyName" xmlns="http://eservices.diligenz.com/">BURGER KING OF CALIFORNIA,
INC.</param>
<param name="criteria" xmlns="http://eservices.diligenz.com/">burger</param>
<param name="expedite" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field--> <param name="deliveryMethodType" xmlns="http://eservices.diligenz.com/">Email</param>
<!--optional field--> <param name="deliveryMethodValue"
xmlns="http://eservices.diligenz.com/">sravani.tummepalli@cscglobal.com</param>
<!--optional field-->
<!--optional field--> <param name="rush" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field-->
<param name="specialInstructions" xmlns="http://eservices.diligenz.com/">Special Instructions added by the
client</param>
<!--optional field-->
<param name="references" xmlns="http://eservices.diligenz.com/">
<array>
<item> <key>MyRef1</key> <value>E-Service</value> </item>
</array>
</param>
</params>
</SubmitOfflineSearch>
</soap:Body> </soap:Envelope>
```



Civil Litigation Search (“CivilLitigation”)

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<SubmitOfflineSearch xmlns="http://eservices.diligenz.com/">
<params>
<param name="contactNo" xmlns="http://eservices.diligenz.com/">Insert Contact No</param>
<param name="loginGuid" xmlns="http://eservices.diligenz.com/">Insert GUID</param>
<param name="type" xmlns="http://eservices.diligenz.com/">CivilLitigation</param>
<param name="stateCD" xmlns="http://eservices.diligenz.com/">CA</param>
<param name="companyName" xmlns="http://eservices.diligenz.com/">BURGER KING OF CALIFORNIA,
INC.</param>
<param name="criteria" xmlns="http://eservices.diligenz.com/">test criteria</param>
<param name="openType" xmlns="http://eservices.diligenz.com/">Open</param>
<param name="litigantType" xmlns="http://eservices.diligenz.com/">Plaintiff</param>
<param name="docket" xmlns="http://eservices.diligenz.com/">1</param>
<param name="complaint" xmlns="http://eservices.diligenz.com/">1</param>
<param name="lastActionFiled" xmlns="http://eservices.diligenz.com/">1</param>
<param name="expedite" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field--> <param name="deliveryMethodType" xmlns="http://eservices.diligenz.com/">Email</param>
<!--optional field--> <param name="deliveryMethodValue"
xmlns="http://eservices.diligenz.com/">sravani.tummepalli@cscglobal.com</param>
<!--optional field-->
<!--optional field--> <param name="rush" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field-->
<param name="specialInstructions" xmlns="http://eservices.diligenz.com/">Special Instructions added by the
client</param>
<!--optional field-->
<param name="references" xmlns="http://eservices.diligenz.com/">
<array>
<item> <key>MyRef1</key> <value>E-Service</value> </item>
</array>
</param>
</params>
</SubmitOfflineSearch>
</soap:Body> </soap:Envelope>
```




Federal Civil Litigation Search (“FederalCivilLitigation”)

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<SubmitOfflineSearch xmlns="http://eservices.diligenz.com/">
<params>
<param name="contactNo" xmlns="http://eservices.diligenz.com/">Insert Contact No</param>
<param name="loginGuid" xmlns="http://eservices.diligenz.com/">Insert GUID</param>
<param name="type" xmlns="http://eservices.diligenz.com/">FederalCivilLitigation</param>
<param name="stateCD" xmlns="http://eservices.diligenz.com/">CA</param>
<param name="companyName" xmlns="http://eservices.diligenz.com/">BURGER KING OF CALIFORNIA,
INC.</param>
<param name="criteria" xmlns="http://eservices.diligenz.com/">test criteria</param>
<param name="openType" xmlns="http://eservices.diligenz.com/">Open</param>
<param name="litigantType" xmlns="http://eservices.diligenz.com/">Plaintiff</param>
<param name="docket" xmlns="http://eservices.diligenz.com/">1</param>
<param name="complaint" xmlns="http://eservices.diligenz.com/">1</param>
<param name="lastActionFiled" xmlns="http://eservices.diligenz.com/">1</param>
<param name="expedite" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field--> <param name="deliveryMethodType" xmlns="http://eservices.diligenz.com/">Email</param>
<!--optional field--> <param name="deliveryMethodValue"
xmlns="http://eservices.diligenz.com/">sravani.tummepalli@cscglobal.com</param>
<!--optional field-->
<!--optional field--> <param name="rush" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field-->
<param name="specialInstructions" xmlns="http://eservices.diligenz.com/">Special Instructions added by the
client</param>
<!--optional field-->
<param name="references" xmlns="http://eservices.diligenz.com/">
<array>
<item> <key>MyRef1</key> <value>E-Service</value> </item>
</array>
</param>
</params>
</SubmitOfflineSearch>
</soap:Body> </soap:Envelope>
```



Federal Criminal Litigation Search ("FederalCriminalLitigation")

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<SubmitOfflineSearch xmlns="http://eservices.diligenz.com/">
<params>
<param name="contactNo" xmlns="http://eservices.diligenz.com/">Insert Contact No</param>
<param name="loginGuid" xmlns="http://eservices.diligenz.com/">Insert GUID</param>
<param name="type" xmlns="http://eservices.diligenz.com/">FederalCriminalLitigation</param>
<param name="stateCD" xmlns="http://eservices.diligenz.com/">CA</param>
<param name="companyName" xmlns="http://eservices.diligenz.com/">BURGER KING OF CALIFORNIA,
INC.</param>
<param name="criteria" xmlns="http://eservices.diligenz.com/">test criteria</param>
<param name="openType" xmlns="http://eservices.diligenz.com/">Open</param>
<param name="litigantType" xmlns="http://eservices.diligenz.com/">Plaintiff</param>
<param name="docket" xmlns="http://eservices.diligenz.com/">1</param>
<param name="complaint" xmlns="http://eservices.diligenz.com/">1</param>
<param name="lastActionFiled" xmlns="http://eservices.diligenz.com/">1</param>
<param name="expedite" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field--> <param name="deliveryMethodType" xmlns="http://eservices.diligenz.com/">Email</param>
<!--optional field--> <param name="deliveryMethodValue"
xmlns="http://eservices.diligenz.com/">sravani.tummepalli@cscglobal.com</param>
<!--optional field-->
<!--optional field--> <param name="rush" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field-->
<param name="specialInstructions" xmlns="http://eservices.diligenz.com/">Special Instructions added by the
client</param>
<!--optional field-->
<param name="references" xmlns="http://eservices.diligenz.com/">
<array>
<item> <key>MyRef1</key> <value>E-Service</value> </item>
</array>
</param>
</params>
</SubmitOfflineSearch>
</soap:Body> </soap:Envelope>
```



Criminal Litigation Search (“CriminalLitigation”)

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<SubmitOfflineSearch xmlns="http://eservices.diligenz.com/">
<params>
<param name="contactNo" xmlns="http://eservices.diligenz.com/">Insert Contact No</param>
<param name="loginGuid" xmlns="http://eservices.diligenz.com/">Insert GUID</param>
<param name="type" xmlns="http://eservices.diligenz.com/">CriminalLitigation</param>
<param name="stateCD" xmlns="http://eservices.diligenz.com/">CA</param>
<param name="companyName" xmlns="http://eservices.diligenz.com/">BURGER KING OF CALIFORNIA,
INC.</param>
<param name="criteria" xmlns="http://eservices.diligenz.com/">test criteria</param>
<param name="openType" xmlns="http://eservices.diligenz.com/">Open</param>
<param name="litigantType" xmlns="http://eservices.diligenz.com/">Plaintiff</param>
<param name="docket" xmlns="http://eservices.diligenz.com/">1</param>
<param name="complaint" xmlns="http://eservices.diligenz.com/">1</param>
<param name="lastActionFiled" xmlns="http://eservices.diligenz.com/">1</param>
<param name="dateOfBirth" xmlns="http://eservices.diligenz.com/">25/01/1989</param>
<param name="SSN" xmlns="http://eservices.diligenz.com/">66</param>
<param name="address" xmlns="http://eservices.diligenz.com/">HYD</param>
<param name="city" xmlns="http://eservices.diligenz.com/">HYD</param>
<param name="state" xmlns="http://eservices.diligenz.com/">AP</param>
<param name="country" xmlns="http://eservices.diligenz.com/">INDIA</param>
<param name="expedite" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field--> <param name="deliveryMethodType" xmlns="http://eservices.diligenz.com/">Email</param>
<!--optional field--> <param name="deliveryMethodValue"
xmlns="http://eservices.diligenz.com/">sravani.tummepalli@cscglobal.com</param>
<!--optional field-->
<!--optional field--> <param name="rush" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field-->
<param name="specialInstructions" xmlns="http://eservices.diligenz.com/">Special Instructions added by the
client</param>
<!--optional field-->
<param name="references" xmlns="http://eservices.diligenz.com/">
<array>
<item> <key>MyRef1</key> <value>E-Service</value> </item>
</array>
</param>
</params>
</SubmitOfflineSearch>
</soap:Body> </soap:Envelope>
```



Bankruptcy Search (“Bankruptcy”)

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<SubmitOfflineSearch xmlns="http://eservices.diligenz.com/">
<params>
<param name="contactNo" xmlns="http://eservices.diligenz.com/">Insert Contact No</param>
<param name="loginGuid" xmlns="http://eservices.diligenz.com/">Insert GUID</param>
<param name="type" xmlns="http://eservices.diligenz.com/">Bankruptcy</param>
<param name="stateCD" xmlns="http://eservices.diligenz.com/">CA</param>
<param name="debtorName" xmlns="http://eservices.diligenz.com/">BURGER KING OF CALIFORNIA,
INC.</param>
<param name="expedite" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field--> <param name="deliveryMethodType" xmlns="http://eservices.diligenz.com/">Email</param>
<!--optional field--> <param name="deliveryMethodValue"
xmlns="http://eservices.diligenz.com/">sravani.tummepalli@cscglobal.com</param>
<!--optional field-->
<!--optional field--> <param name="rush" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field-->
<param name="specialInstructions" xmlns="http://eservices.diligenz.com/">Special Instructions added by the
client</param>
<!--optional field-->
<param name="references" xmlns="http://eservices.diligenz.com/">
<array>
<item> <key>MyRef1</key> <value>E-Service</value> </item>
</array>
</param>
</params>
</SubmitOfflineSearch>
</soap:Body> </soap:Envelope>
```



Business Credit Report ("CreditReport")

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<SubmitOfflineSearch xmlns="http://eservices.diligenz.com/">
<params>
<param name="contactNo" xmlns="http://eservices.diligenz.com/">Insert Contact No</param>
<param name="loginGuid" xmlns="http://eservices.diligenz.com/">Insert GUID</param>
<param name="type" xmlns="http://eservices.diligenz.com/">CreditReport</param>
<param name="stateCD" xmlns="http://eservices.diligenz.com/">CA</param>
<param name="companyName" xmlns="http://eservices.diligenz.com/">BURGER KING OF CALIFORNIA,
INC.</param>
<param name="address" xmlns="http://eservices.diligenz.com/">HYD</param>
<param name="city" xmlns="http://eservices.diligenz.com/">HYD</param>
<param name="state" xmlns="http://eservices.diligenz.com/">AP</param>
<param name="country" xmlns="http://eservices.diligenz.com/">INDIA</param>
<param name="expedite" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field--> <param name="deliveryMethodType" xmlns="http://eservices.diligenz.com/">Email</param>
<!--optional field--> <param name="deliveryMethodValue"
xmlns="http://eservices.diligenz.com/">sravani.tummepalli@cscglobal.com</param>
<!--optional field-->
<!--optional field--> <param name="rush" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field-->
<param name="specialInstructions" xmlns="http://eservices.diligenz.com/">Special Instructions added by the
client</param>
<!--optional field-->
<param name="references" xmlns="http://eservices.diligenz.com/">
<array>
<item> <key>MyRef1</key> <value>E-Service</value> </item>
</array>
</param>
</params>
</SubmitOfflineSearch>
</soap:Body> </soap:Envelope>
```



Flood Certification Search (“Flood”)

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<SubmitOfflineSearch xmlns="http://eservices.diligenz.com/">
<params>
<param name="contactNo" xmlns="http://eservices.diligenz.com/">Insert Contact No</param>
<param name="loginGuid" xmlns="http://eservices.diligenz.com/">Insert GUID</param>
<param name="type" xmlns="http://eservices.diligenz.com/">Flood</param>
<param name="stateCD" xmlns="http://eservices.diligenz.com/">CA</param>
<param name="companyName" xmlns="http://eservices.diligenz.com/">BURGER KING OF CALIFORNIA,
INC.</param>
<param name="address" xmlns="http://eservices.diligenz.com/">HYD</param>
<param name="city" xmlns="http://eservices.diligenz.com/">HYD</param>
<param name="state" xmlns="http://eservices.diligenz.com/">AP</param>
<param name="county" xmlns="http://eservices.diligenz.com/">S.O.S.</param>
<param name="expedite" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field--> <param name="deliveryMethodType" xmlns="http://eservices.diligenz.com/">Email</param>
<!--optional field--> <param name="deliveryMethodValue"
xmlns="http://eservices.diligenz.com/">sravani.tummepalli@cscglobal.com</param>
<!--optional field-->
<!--optional field--> <param name="rush" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field-->
<param name="specialInstructions" xmlns="http://eservices.diligenz.com/">Special Instructions added by the
client</param>
<!--optional field-->
<param name="references" xmlns="http://eservices.diligenz.com/">
<array>
<item> <key>MyRef1</key> <value>E-Service</value> </item>
</array>
</param>
</params>
</SubmitOfflineSearch>
</soap:Body> </soap:Envelope>
```



Suits, Liens and Judgements by Company Name ("SLJ")

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<SubmitOfflineSearch xmlns="http://eservices.diligenz.com/">
<params>
<param name="contactNo" xmlns="http://eservices.diligenz.com/">Insert Contact No</param>
<param name="loginGuid" xmlns="http://eservices.diligenz.com/">Insert GUID</param>
<param name="type" xmlns="http://eservices.diligenz.com/">SLJ</param>
<param name="stateCD" xmlns="http://eservices.diligenz.com/">CA</param>
<param name="companyName" xmlns="http://eservices.diligenz.com/">BURGER KING OF CALIFORNIA,
INC.</param>
<param name="expedite" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field--> <param name="deliveryMethodType" xmlns="http://eservices.diligenz.com/">Email</param>
<!--optional field--> <param name="deliveryMethodValue"
xmlns="http://eservices.diligenz.com/">sravani.tummepalli@cscglobal.com</param>
<!--optional field-->
<!--optional field--> <param name="rush" xmlns="http://eservices.diligenz.com/">True</param>
<!--optional field-->
<param name="specialInstructions" xmlns="http://eservices.diligenz.com/">Special Instructions added by the
client</param>
<!--optional field-->
<param name="references" xmlns="http://eservices.diligenz.com/">
<array>
<item> <key>MyRef1</key> <value>E-Service</value> </item>
</array>
</param>
</params>
</SubmitOfflineSearch>
</soap:Body> </soap:Envelope>
```



4.8. Filing CSC Financial Web Services

4.8.1. CreateFiling

This method creates a filing in the CSC system based on the supplied information and returns the OrderID of the created filing.

Inputs:

Parameter	Description	Required?	Value
xmlFiling	The XML filing to create	Required	An xml element
attachment	A file to attach to the order along with the filing	Optional	A base64-encoded file
contentType	The MIME type of the attachment OR Used to signify you will be sending the attachment through other means	Optional (required if "attachment" is used)	One of the following: <ul style="list-style-type: none">• text/plain• text/xml• application/pdf• application/msword• image/tiff• offline
description	A description of the attachment OR The offline method by which you will be providing the attachment	Optional	Any string (e.g., "Collateral") Offline methods: <ul style="list-style-type: none">• Fax• E-Mail• US Mail• Overnight Carrier

For the XML filing, some values need to be in a specific format:

- /Document/Record/Diligenz/FilingJurisdiction – The full name of the state and jurisdiction are expected. For state-level filings, use "(S.O.S.)".



Here is a sample of a UCC1 filing call:

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <CreateFiling xmlns="http://eservices.diligenz.com/">
      <params>
        <param name="contactNo">Insert contact number</param>
        <param name="loginGuid">Insert GUID</param>
        <param name="xmlFiling"></param>
        <Document xmlns=""></Document>
      </params>
    </CreateFiling>
  </soap:Body>
</soap:Envelope>
```

The XML snippet above is a sample of a UCC1 filing call. It is an SOAP envelope with a single body element, `<CreateFiling>`, which contains a `<params>` element. The `<params>` element contains three `<param>` elements: `<param name="contactNo">Insert contact number</param>`, `<param name="loginGuid">Insert GUID</param>`, and `<param name="xmlFiling"></param>`. The `<Document>` element is also present but empty. The XML is well-formed and follows the SOAP 1.1 standard.

After the filing is successfully created, the method returns the OrderID of the unapproved filing.

**Outputs:**

- An orderID
- A CSC-generated copy of the XML filing
- Validation Messages

The return document conforms to the following schema:

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <CreateFilingResponse xmlns="http://eservices.diligenz.com/">
      <CreateFilingResult>
        <OrderInfo>
          <OrderID>98365293</OrderID>
          <Filing>
            <Document xmlns="">
              <Header>
                <Filer>
                  <Names>
                    <OrganizationName>Corporation Service Company</OrganizationName>
                    <MailAddress>801 Adlai Stevenson Drive</MailAddress>
                    <City>Springfield</City>
                    <State>IL</State>
                    <PostalCode>62703</PostalCode>
                    <Country>USA</Country>
                  </Names>
                  <ClientAccountNum/>
                  <ContactName>Corporation Service Company</ContactName>
                  <ContactPhone>1-800-858-5294</ContactPhone>
                  <ContactEmail/>
                  <ContactFax>1-800-345-6059</ContactFax>
                </Filer>
                <PacketNum/>
                <Test>NO</Test>
              </Header>
              <Record>
                <SeqNumber>001</SeqNumber>
                <TransType Type="Initial"/>
                <SubmitterRef/>
                <AltNameDesignation AltName="NOAltName"/>
                <AltFilingType AltType="NOAltType"/>
                <AltFilingType1 AltType="NOAltType"/>
                <Debtors>
                  <DebtorName>
                    <Names>
                      <OrganizationName>My Third Debtor, Inc.</OrganizationName>
                      <MailAddress>6500 Harbour Heights Pkwy, Suite 400</MailAddress>
                      <City>Mukilteo</City>
                      <State>WA</State>
                      <PostalCode>98275</PostalCode>
                      <County>Benton</County>
                      <Country>USA</Country>
                      <OrganizationalType Type="NOType">My own type</OrganizationalType>
                      <OrganizationalJuris>WA</OrganizationalJuris>
                      <OrganizationalID>1234567890</OrganizationalID>
                    </Names>
                  </DebtorName>
                </Debtors>
                <Secured>
                  <SecuredName>
                    <Names>
                      <OrganizationName>Demo Bank</OrganizationName>
                      <MailAddress>100 main St</MailAddress>
                      <City>Maple Oaks</City>
                      <State>OH</State>
                      <PostalCode>12345</PostalCode>
                      <Country>USA</Country>
                    </Names>
                  </SecuredName>
                </Secured>
                <Collateral>
                  <CollText>Debtor grants Secured Party a security interest in all of Debtor's now existing and hereafter acquired
                </Collateral>
                <CollateralDesignation Type="NODesignation"/>
              </Record>
            </Document>
          </Filing>
          <Messages>
            <Message level="0">Filing passed with no validation errors.</Message>
          </Messages>
        </OrderInfo>
      </CreateFilingResult>
    </CreateFilingResponse>
  </soap:Body>
</soap:Envelope>
```



4.8.2. ValidateFiling

This method checks the filing against the Validation database and returns an XML string containing the validation errors and warnings that occurred. If no validation errors are present, the return value will be an empty tag.

Input:

Parameter	Description	Required?	Value
orderId	The orderId of the filing to validate	Required	A positive integer

The level corresponds to the different warning levels seen on the website; green (1), yellow (2) and red (3). Green messages are informational messages while yellow may require some action to be taken. Red messages indicate a failure will occur if you try to approve the order.

4.8.3. UpdateFiling

This method will allow you to update the data associated with a filing. This method is used the same way CreateFiling is used. You just need to provide the OrderID that needs to be updated.

Inputs:

Parameter	Description	Required?	Value
orderId	The orderId of the filing to validate	Required	A positive integer
xmlFiling	The XML filing to create	Required	An xml element
attachment	A file to attach to the order along with the filing	Optional	A base64-encoded file
contentType	The MIME type of the attachment OR Used to signify you will be sending the attachment through other means	Optional (required if "attachment" is used)	One of the following: <ul style="list-style-type: none">• text/plain• text/xml• application/pdf• application/msword• image/tiff• offline
description	A description of the attachment OR The offline method you will be providing the attachment.	Optional	Any string (e.g., "Collateral"). Offline methods: <ul style="list-style-type: none">• Fax• E-Mail• US Mail• Overnight Carrier

After the filing is successfully created, the method returns the OrderID of the unapproved filing.

Outputs:

- An orderId
- A CSC-generated copy of the XML filing



4.8.4. ApproveOrder

This method approves the filing. Filings can only be approved if they do not have any validation messages above level two. Any level two (yellow) messages should be displayed to users so they can decide how to best handle the message.

Input:

Parameter	Description	Required?	Value
orderId	The orderId of the filing to approve	Required	A positive integer

Output:

- None

4.8.5. ContinueFiling

This method allows you to file a continuation to an existing filing that is in the system. The following requirements must be met in order for the filing to be continued:

- The order must be a filing order
- The filing must be approved
- The filing must be filed within the US

Input:

Parameter	Description	Required?	Value
orderId	The orderId of the filing to continue	Required	A positive integer

Output:

- The OrderID of the created continuation

4.8.6. TerminateFiling

This method allows you to file a termination to an existing filing that is in the system.

Input:

Parameter	Description	Required?	Value
orderId	The orderId of the filing to terminate	Required	A positive integer

Output:

- The OrderID of the created termination



4.9. Document Center CSC Financial Web Services

4.9.1. Order Status Codes

The following is a list of order status codes and their meanings:

- **ARNL** - The Acknowledgment has been received from the jurisdiction but we are still waiting on the Debtor Notification letter.
- **ARWIP** - The Attachment has been received from the client and is currently being worked on.
- **B** - The order has been sent over to the Billing department, so that an invoice can be created.
- **C** - The order has been completed and billed.
- **DTJ** - The filing has been delivered (electronically or mailed) to the jurisdiction.
- **IMP** - Data for this filing has been imported from an external source (e.g., Excel document from the client)
- **NEW** - This order has been approved by the client and ready to be worked on.
- **NSTR** - An Acknowledgment has been received and needs to have a "Search To Reflect" run against it.
- **PCA** - The order has been created by the client but is still waiting for approval before it can be worked on.
- **PCQC** - The order is pending QC from the client before it can be worked on.
- **WCA** - Waiting for the client to send attachments related to the order.
- **WCR** - Waiting for response to a query that was sent out to the client.
- **WIP** - The order is currently being worked on.

4.9.2. GetOrderInformation

This method allows you to get various pieces of information about an order in the CSC database. This includes the types of services the order has, as well as information on the status of an order.

Input:

Parameter	Description	Required?	Value
orderID	The orderID for retrieval	Required	A positive integer

Output:

- Information about the order

The outgoing document conforms to the following schema:

```
<?xml version='1.0' encoding='UTF-8'>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:eser="http://eservices.diligenz.com/">
  <soapenv:Header />
  <soapenv:Body>
    <eser:GetOrderInformation xmlns="http://eservices.diligenz.com/">
      <params>
        <param name="loginGuid">Insert GUID</param>
        <param name="contactNo">Insert contact number</param>
        <param name="orderID">88361605</param>
      </params>
    </eser:GetOrderInformation>
  </soapenv:Body>
</soapenv:Envelope>
```



The return document conforms to the following schema:

```
<?xml version='1.0' encoding='UTF-8' ?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetOrderInformationResponse xmlns="http://eservices.diligenz.com/">
      <GetOrderInformationResult>
        <OrderInfo>
          <OrderID>88361605</OrderID>
          <CreateDate>2014-06-23T09:31:05.0930000-04:00</CreateDate>
          <ParentOrderID/>
          <Subject>tcby</Subject>
          <AccountID>330180</AccountID>
          <OrderStatus>C</OrderStatus>
          <FilingNumber/>
          <FilingDate>0001-01-01T00:00:00.0000000-05:00</FilingDate>
          <ExpirationDate>0001-01-01T00:00:00.0000000-05:00</ExpirationDate>
          <OriginalFilingNumber/>
          <OriginalFilingDate/>
          <ApproveDate>0001-01-01T00:00:00.0000000-05:00</ApproveDate>
          <Contact>John Smith</Contact>
          <Email>John.Smith@cscglobal.com</Email>
          <CompleteDate>2014-06-23T09:31:06.6770000-04:00</CompleteDate>
          <BillingRef1 Name="BillingRef1">1705882</BillingRef1>
          <BillingRef2 Name="BillingRef2"/>
          <Reference1 Name="reference1"/>
          <Rush>false</Rush>
          <Notes>
            <Note>
              <Description>Client run search -- Do not QC</Description>
              <CreateDate>2014-06-23T09:31:06.1930000-04:00</CreateDate>
            </Note>
          </Notes>
          <Attachments>
            <Attachment>
              <AttachmentID>50491778</AttachmentID>
              <Description>Summary Report</Description>
              <CreateDate>2014-06-23T09:34:00.0000000-04:00</CreateDate>
            </Attachment>
            <Attachment>
              <AttachmentID>50491778</AttachmentID>
              <Description>Summary Report</Description>
              <CreateDate>2014-06-23T09:34:00.0000000-04:00</CreateDate>
            </Attachment>
          </Attachments>
          <Services>
            <Service>
              <ServiceID>80</ServiceID>
              <Description>Suits, Liens and Judgments</Description>
              <CreateDate>2014-06-23T09:31:05.6700000-04:00</CreateDate>
              <State>TX</State>
              <StateID>46</StateID>
              <Jurisdiction>(S.O.S.)</Jurisdiction>
              <JurisdictionID>3184</JurisdictionID>
              <Certified>false</Certified>
              <Documents>false</Documents>
              <Expedite>false</Expedite>
            </Service>
          </Services>
        </OrderInfo>
      </GetOrderInformationResult>
    </GetOrderInformationResponse>
  </soap:Body>
</soap:Envelope>
```



4.9.3. GetChangedOrders

This method retrieves orders whose statuses have changed in the specified timeframe. This would be used to check on the status of orders in the system.

Inputs:

Parameter	Description	Required?	Value
status	The status code to check for	Required	One of the following: (Section 4.9.1 includes explanations of statuses) ARNL ARWIP B C DTJ IMP NEW NSTR PCA PCQC WCA WCR WIP
fromDate	The date from which to check for changed orders	Required	A dateTime value

Output:

- A list of orderIDs and their last modified date



4.9.4. UploadAttachment

This method is used to upload an attachment to an existing order.

Inputs:

Parameter	Description	Required?	Value
orderId	The target orderId	Required	A positive integer
attachment	The attachment to upload	Required	A base64-encoded file
contentType	The MIME type of the file	Required	One of the following: <ul style="list-style-type: none">• text/plain• text/xml• application/pdf• application/msword• image/tiff
description	A description of the attachment	Optional	Any string (e.g., "Collateral")

Output:

- The attachmentID for the uploaded attachment.

4.9.5. GetAttachment

This method is used to retrieve an attachment from an existing order.

Input:

Parameter	Description	Required?	Value
attachmentID	The attachmentID of the attachment (taken from GetOrderInformation)	Required	A positive integer

Output:

- A URL to the attachment