naturalinsight®

Routing Web Service

The Natural Insight Routing Web Service exports Natural Insight XML Routing data useful for integration with proprietary or third-party fulfillment systems. The on-demand Routing API provides Routing data per submitted Route across all active projects according to the dates of submission of the Routes.

Definition of a Route in Natural Insight

A submitted Route in Natural Insight is the submitted itinerary of a staff member's trip on a specific date:

1. from the staff member's home address...
2. to any assignments scheduled for that day in chronological order and, then, after the last assignment of the day...
3. to the home address.

NOTE: Since **Routing** in Natural Insight is customizable per client based upon a company's particular **Routing** rules and reimbursement policy, the **Routing Web Service** provides data for each **Route** submitted in the given time frame based upon the configuration of a client's **Routing** in Natural Insight specified in **Configuration Settings**.

Each submitted Route is represented by a <routing> XML tag.

The Routing data in the Routing API allows clients to collect and calculate Routing expenses and, in doing so, track, analyze and manage these expenses. The Routing data from the

Routing API provides insight about how a client can change Routing policies to prevent financial drain from Routing expenses.

You can request Routing data according to the following search criteria:

- date range (*read* method)

with any submitted Routes falling within the date range returned in the XML pull.

Description

Retrieve and use Natural Insight Routing data across all active projects where each Route exported has been submitted within the specified date range.

EXAMPLE:

EXAMPLE OF TWO <ROUTING> ELEMENTS WITHIN THE <ROOT> ELEMENT REPRESENTING TWO ROUTES

```
<root>

 <routing routingId="1000" submittedDate="2016-02-03
 16:24:41" routingDate="2016-02-03" staffNumber="5003"
 staffName="Appleman, Ann" initialDistance="3" ini-
 tialTimeMinutes="15" firstLegDistance="0" firstLegMinutes-
 s="0" lastLegDistance="0" lastLegMinutes="0"
 challengeDistance="8" challengeMinutes="30" chal-
 lengeReason="Detour" challengeDate="2016-02-03 16:24:45"
 challengeStatus="approved" challengeStatusDate="2016-02-03
 16:39:38" challengeStatusUpdateBy="Smith, Jerry" over-
 rideDistance="" overrideMinutes="" overrideDate="" over-
 rideUpdateBy="" calculatedDistance="8"
```

```
calculatedMinutes="30" calculatedDistanceWithThreshold="8"
calculatedMinutesWithThreshold="30"/>

<routing routingId="1001" submittedDate="2016-04-20
12:26:19" routingDate="2016-03-14" staffNumber="5003"
staffName="Appleman, Ann" pro-
jectVisitIdList="277733,277734,277739" initialDistance="24"
initialTimeMinutes="50" firstLegDistance="2" firstLegMinutes-
s="7" lastLegDistance="2" lastLegMinutes="8" chal-
lengeDistance="" challengeMinutes="" challengeReason=""
challengeDate="" challengeStatus="" challengeStatusDate=""
challengeStatusUpdateBy="" overrideDistance="" over-
rideMinutes="" overrideDate="" overrideUpdateBy="" cal-
culatedDistance="24" calculatedMinutes="50"
calculatedDistanceWithThreshold="20" cal-
culatedMinutesWithThreshold="50"/>

</root>
```

NOTE: Each XML <routing> element within the <root> tag will be presented in chronological order according to the date each **Route** within the date range chosen has occurred.

Frequency

On-demand

WSDL

You can access the WSDL for the Natural Insight Routing API at:

https://my.naturalinsight.com/routingWebService.cfc?wsdl.

```
<wsdl:definitions xmlns:apachesoap="http://xml.apache.org/xml-soap"
xmlns:impl="http://niWeb" xmlns:intf="http://niWeb" xmlns:soa-
penc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:tns1-
1="http://rpc.xml.coldfusion"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:wsdlsoap-
p="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:xsd-
d="http://www.w3.org/2001/XMLSchema" targetNamespace="http://niWeb">

<!-- WSDL created by ColdFusion -->

<wsdl:types>

 <schema xmlns="http://www.w3.org/2001/XMLSchema" tar-
 getNamespace="http://rpc.xml.coldfusion">

   <import namespace="http://schemas.xmlsoap.org/soap/encoding/"/>

   <complexType name="CFCInvocationException">

      <sequence/>

   </complexType>

 </schema>

</wsdl:types>

<wsdl:message name="CFCInvocationException">

 <wsdl:part name="fault" type-
 e="tns1:CFCInvocationException"></wsdl:part>

</wsdl:message>
```

```
<wsdl:message name="readResponse">

  <wsdl:part name="readReturn" type="xsd:string"></wsdl:part>

</wsdl:message>

<wsdl:message name="readRequest">

  <wsdl:part name="authCd" type="xsd:string"></wsdl:part>

  <wsdl:part name="accessKey" type="xsd:string"></wsdl:part>

  <wsdl:part name="startDate" type="xsd:string"></wsdl:part>

  <wsdl:part name="endDate" type="xsd:string"></wsdl:part>

</wsdl:message>

<wsdl:portType name="routingWebService">

  <wsdl:operation name="read" parameterOrder="authCd accessKey
  startDate endDate">

    <wsdl:input message="impl:readRequest" name-
    e="readRequest"></wsdl:input>

    <wsdl:output message="impl:readResponse" name-
    e="readResponse"></wsdl:output>

    <wsdl:fault message="impl:CFCInvocationException" name-
    e="CFCInvocationException"></wsdl:fault>

  </wsdl:operation>

</wsdl:portType>
```

```
<wsdl:binding name="routingWebService.cfcSoapBinding" type-
e="impl:routingWebService">

  <wsdlsoap:binding style="rpc" trans-
  port="http://schemas.xmlsoap.org/soap/http"/>

  <wsdl:operation name="read">

    <wsdlsoap:operation soapAction=""/>

      <wsdl:input name="readRequest">

          <wsdlsoap:body encod-
          ingStyle="http://schemas.xmlsoap.org/soap/encoding/"
          namespace="http://niWeb" use="encoded"/>

      </wsdl:input>

      <wsdl:output name="readResponse">

          <wsdlsoap:body encod-
          ingStyle="http://schemas.xmlsoap.org/soap/encoding/"
          namespace="http://niWeb" use="encoded"/>

      </wsdl:output>

      <wsdl:fault name="CFCInvocationException">

          <wsdlsoap:fault encod-
          ingStyle="http://schemas.xmlsoap.org/soap/encoding/" name-
          e="CFCInvocationException" namespace="http://niWeb"
          use="encoded"/>

      </wsdl:fault>
```

```
  </wsdl:operation>

</wsdl:binding>

<wsdl:service name="routingWebServiceService">

  <wsdl:documentation>Routing Web service com-
  ponent</wsdl:documentation>

  <wsdl:port binding="impl:routingWebService.cfcSoapBinding" name-
  e="routingWebService.cfc">

    <wsdlsoap:address loc-
    ation="http://my.naturalinsight.com/routingWebService.cfc"/>

  </wsdl:port>

</wsdl:service>

</wsdl:definitions>
```

naturalinsight®

Routing Web Service Data Descriptions

The Natural Insight Routing Web Service exports Routing data in your client account per submitted Route according to the date range when each Route occurred. Since a Route is a collection of legs (home to assignment A to assignment B etc.) on a specific day for a specific staff member, the Routing Web Service will read all Routes submitted in Natural Insight according to the date when they occurred. The date range passed in refers to when the assignments included in the Route were completed.

EXAMPLE: If a date range of one day - September 24, 2016 - is sent in to the **Routing API**, then all of the **Routes** whose assignments had completed dates within that range (on September 24, 2016) will be returned.

An example of the XML returned when a call to the Routing API is made follows. The XML element <routing> represents a Route. As a result, a <routing> element will have more than one leg within the Route and the number of legs will depend on the assignments completed on the day of the Route.

EXAMPLE:

***EXAMPLE OF TWO <ROUTING> ELEMENTS WITHIN THE <ROOT> ELEMENT REPRESENTING TWO* ROUTES**

```
<root>
```

```
<routing routingId="1000" submittedDate="2016-02-03
16:24:41" routingDate="2016-02-03" staffNumber="5003"
staffName="Appleman, Ann" initialDistance="3" ini-
tialTimeMinutes="15" firstLegDistance="0" firstLegMinutes-
s="0" lastLegDistance="0" lastLegMinutes="0"
challengeDistance="8" challengeMinutes="30" chal-
lengeReason="Detour" challengeDate="2016-02-03 16:24:45"
challengeStatus="approved" challengeStatusDate="2016-02-03
16:39:38" challengeStatusUpdateBy="Smith, Jerry" over-
rideDistance="" overrideMinutes="" overrideDate="" over-
rideUpdateBy="" calculatedDistance="8"
calculatedMinutes="30" calculatedDistanceWithThreshold="8"
calculatedMinutesWithThreshold="30"/>

<routing routingId="1001" submittedDate="2016-04-20
12:26:19" routingDate="2016-03-14" staffNumber="5003"
staffName="Appleman, Ann" pro-
jectVisitIdList="277733,277734,277739" initialDistance="24"
initialTimeMinutes="50" firstLegDistance="2" firstLegMinutes-
s="7" lastLegDistance="2" lastLegMinutes="8" chal-
lengeDistance="" challengeMinutes="" challengeReason=""
challengeDate="" challengeStatus="" challengeStatusDate=""
challengeStatusUpdateBy="" overrideDistance="" over-
rideMinutes="" overrideDate="" overrideUpdateBy="" cal-
culatedDistance="24" calculatedMinutes="50"
calculatedDistanceWithThreshold="20" cal-
culatedMinutesWithThreshold="50"/>

</root>
```

Access the **Routing** XML-based WSDL (using the *read* method) at https://my.naturalinsight.com/routingWebService.cfc?wsdl.

A Route - i.e., a <routing> element - may have more than one assignment included in it. The data associated with a Route that is returned from the Routing Web Service for each <routing> tag includes the following information:

<routing> Element			
XML Attribute	**Data Type**	**Max Data Size**	**Description**
routingId	Integer	100	*Routing ID* that uniquely identifies a Route
submittedDate	Date & Time		The date and time the staff member submitted the Route in Natural Insight
routingDate	Date	50	The date the assignments included in the Route were completed.
staffNumber	VARCHAR	25	*Staff Member ID* that uniquely identifies the staff member who completed the Route
staffName	VARCHAR	200	Last name, first name of staff member (e.g. McLeod, Jennifer)
projectVisitIdList			If more than one assignment is included in the Route, a comma-delimited list of the *Assignment IDs* of the assignments that are part of the Route at the time of Routing submission. (For ex., *projectVisitIdList ="277733,277734,277739"*) If only one assignment is part of the Route, then the *projectVisitIdList* parameter is not included.
initialDistance			The distance of the Route calculated by our automated Routing provider.

<routing> Element			
XML Attribute	**Data Type**	**Max Data Size**	**Description**
initialTimeMinutes			The time to drive the Route calculated in minutes by our automated Routing provider.
firstLegDistance			When the distance (mileage) of the first leg of the Route (i.e., the distance from the staff member's home address to the location of the first assignment of the day) is included (i.e., if the *Mileage From Home* checkbox is checked in Configuration Settings), *firstLegDistance* will hold the value of the distance of the first leg. When the distance of the first leg of the Route is not included (i.e., the *Mileage From Home* checkbox is not checked in Configuration Settings), *firstLegDistance* will be null.

<routing> Element			
XML Attribute	Data Type	Max Data Size	Description
firstLegMinutes			When the drive time of the first leg of the Route (i.e., the time it takes to drive from the staff member's home address to the location of the first assignment of the day) is included (i.e., if the *Drive Time From Home* checkbox is checked in Configuration Settings), *firstLegMinutes* will hold the value of the number of minutes it takes to drive the first leg. When the drive time of the first leg of the Route is not included (i.e., the *Drive Time From Home* checkbox is not checked in Configuration Settings), *firstLegMinutes* will be null.
lastLegDistance			When the distance (mileage) of the last leg of the Route (i.e., the distance from the location of the last assignment of the day to the staff member's home address) is included (i.e., if the *Mileage To Home* checkbox is checked in Configuration Settings), *lastLegDistance* will hold the value of the distance of the last leg. When the distance of the last leg of the Route is not included (i.e., the *Mileage To Home* checkbox is not checked in Configuration Settings), *lastLegDistance* will be null.

<routing> Element			
XML Attribute	**Data Type**	**Max Data Size**	**Description**
lastLegMinutes			When the drive time of the last leg of the Route (i.e., the time it takes to drive from the location of the last assignment of the day to the staff member's home address) is included (i.e., if the *Drive Time To Home* checkbox is checked in Configuration Settings), *lastLegMinutes* will hold the value of the number of minutes it takes to drive the last leg. When the drive time of the last leg of the Route is not included (i.e., the *Drive Time To Home* checkbox is not checked in Configuration Settings), *lastLegMinutes* will be null.
challengeDistance[1]			When a staff member has challenged the distance (mileage) of a Route, the *challengeDistance* is the distance (mileage) value the staff member has submitted.
challengeMinutes			When a staff member has challenged the drive time of a Route, the *challengeMinutes* is the value in number of minutes the staff member has submitted for the time it took to drive the Route.

[1] *If a* Route *has not been challenged, the value of all* Challenge *parameters will be null.*

<routing> Element			
XML Attribute	**Data Type**	**Max Data Size**	**Description**
challengeReason	VARCHAR	255	When a staff member has challenged the distance and/or drive time of a **Route**, the *challengeReason* is the reason submitted by the staff member as to why he/she is challenging the **Route**.
challengeDate	Date/Time		When a staff member has challenged the distance and/or drive time of a **Route**, the *challengeDate* is the date and time when the **Challenge** was submitted by the staff member.
challengeStatus	*approved*, *declined*, or *pending*		When a staff member has challenged the distance and/or drive time of a **Route**, the *challengeStatus* is the status of the **Challenge**: *approved*, *declined*, or *pending* (neither approved nor declined).
challengeStatusDate	Date/Time		When a staff member has challenged the distance and/or drive time of a **Route**, the *challengeStatusDate* is the date and time when a manager has marked the status of the **Challenge** (*challengeStatus*) as either *approved*or *declined*.
challengeStatusUpdateBy			When a staff member has challenged the distance and/or drive time of a **Route**, the *challengeStatusUpdateBy* is the *Last name*, *First name* of the manager who has marked the status of the **Challenge** (*challengeStatus*) as either *approved*or *declined*.

<routing> Element			
XML Attribute	**Data Type**	**Max Data Size**	**Description**
overrideDistance[1]			When a manager overrides the distance (mileage) of a Route, the *overrideDistance* is the distance the manager submits for reimbursement of the Route.
overrideMinutes			When a manager overrides the drive time (minutes) of a Route, the *overrideMinutes* is the number of minutes the manager submits for reimbursement of the Route.
overrideDate			When a manager overrides the distance (mileage) and/or drive time (minutes) of a Route, the *overrideDate* is the date and time of the Override of the Route.
overrideUpdateBy			When a manager overrides the distance (mileage) and/or drive time (minutes) of a Route, the *overrideUpdateBy* is the *Last name*, *First name* of the manager who submitted the Override of the Route.
calculatedDistance			The calculated distance (mileage) for the Route based upon Natural Insight Routing rules. (See ***calculatedDistance and calculatedMinutes Rules*** in the next section for an explanation of the Natural Insight Routing rules.)

[1] *If a* Route *has not been overriden, the value of all* Override *parameters will be null.*

<routing> Element			
XML Attribute	**Data Type**	**Max Data Size**	**Description**
calculatedMinutes			The calculated drive time in minutes for the Route based upon Natural InsightRouting rules. (See ***calculatedDistance and calculatedMinutes Rules*** in the next section for an explanation of the Natural Insight Routing rules.)
calculatedDistanceWithThreshold			If a threshold is used, the value of *calculatedDistanceWithThreshold* reflects the application of the threshold value to *calculatedDistance*. If a threshold is not used, *calculatedDistanceWithThreshold* will be the same as the value of *calculatedDistance*.
calculatedMinutesWithThreshold			If a threshold is used, the value of *calculatedMinutesWithThreshold* reflects the application of the threshold value to *calculatedMinutes*. If a threshold is not used, *calculatedMinutesWithThreshold* will be the same as the value of *calculatedMinutes*.

calculatedDistance and *calculatedMinutes* Rules

Several rules exist in order to determine which values are used for the *calculatedDistance* and *calculatedMinutes* of a Route. If present for a Route, Override values always take precedence.

In the Case of Route Override

When a has a manager Override, the values the manager has entered for the Override will be used for *calculatedDistance* and/or *calculatedMinutes*.

> **NOTE:** Threshold values for distance or drive time are never applied to **Override** values.

In the Case of Route Challenge

If no Override but the Route has been challenged:

A. If the Challenge has been accepted or has not been reviewed, the values the staff member has entered for the Challenge will be used for *calculatedDistance* and/or *calculatedMinutes*.

B. If the Challenge has been declined, the values used for *calculatedDistance* and/or *calculatedMinutes* will be those provided by Natural Insight's automated Routing provider or, if the manager who declines the Challenge enters Override values for distance and/or drive time at the time of decline, those Override values will be used.

In the Case of Route Acceptance

If a staff member accepts a Route, the values used for *calculatedDistance* and/or *calculatedMinutes* will be those provided by Natural Insight's automated Routing provider.

In the Case of Thresholds

You may or may not use thresholds in your client configuration of Routing in Natural Insight.

If thresholds are used, the values for *calculatedDistanceWithThreshold* and/or *calculatedMinutesWithThreshold* reflect the application of distance and/or drive time threshold values to *calculatedDistance* and/or *calculatedMinutes* respectively.

If thresholds are not used, the values for *calculatedDistanceWithThreshold* and *calculatedMinutesWithThreshold* will be the same as the values of *calculatedDistance* and *calculatedMinutes* respectively.

NOTE: Threshold values for distance or drive time are never applied to **Override** values.

Consult with your Natural Insight Account Manager to determine if thresholds are being used in your client configuration of Routing in Natural Insight as well as any customization of Routing calculations for your account.

NOTE: If you would like more information about how **Routing** works in Natural Insight, please refer to ***ROUTING FOR THE STAFF MEMBER - A NATURAL INSIGHT FEATURE GUIDE***. Your company will most likely have customized **Routing** in Natural Insight according to your organization's travel rules and reimbursement policy. As a result, consult with your Natural Insight Account Manager for specifics regarding how **Routing** works for your client instance of Natural Insight.

Routing Web Service Method

The Natural Insight SOAP-based Routing Web Service allows clients to export Natural Insight Routing data as a data package according to a date range using the *read* method. In other words, any Routes that have been submitted within the provided date range will be returned by the *read* method.

The *read* public method in the Natural Insight Routing Web Service can be consumed using either the SOAP protocol over HTTPS or simple HTTP GET/POST requests. The WSDL definition for the Natural Insight Routing Web Service, located at the URL, https://my.naturalinsight.com/routingWebService.cfc?wsdl, describes its methods and arguments in detail.

To retrieve a Routing data package by date range (based upon Route submission date), use the following URL:

```
https://my.naturalinsight.com/routingWebService.cfc?wsdl
&method=read&authCd=XXX-XXX-XXXX&accessKey=XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX
&startDate=2016-10-01 HH:MM:SS&endDate=2016-10-11 HH:MM:SS
```

Replace the parameter values in the above URL with your personalized parameters described in the next section.

read Parameters

The *read* method takes four parameters:

1. required string *authCd* - Contact Natural Insight Client Services to obtain your client authorization code (*authCd*).

2. required access key (*accessKey*) as a string

   accessKey is a custom code created for each of your clients to whom you would like to grant access to the Natural Insight Web Services Routing API and, thus, give this client the ability to read Routes within your instance of Natural Insight. In this way, Natural Insight can revoke access to the client to whom you have granted access to the Routing API upon your request. If you have several clients to whom you would like to grant access to the Routing Web Service, you will request a unique access key (*accessKey*) for each one.

3. required *startDate* in ISO 8601 format

   NOTE: This date is compared to a **Route's** submit date.

4. required *endDate* in ISO 8061 format

   NOTE: This date is compared to a **Route's** submit date.

The *read* method returns the Routing data package in the default format of WDDX (Web Distributed Data eXchange) - a string of XML-encoded data.

Return Formats of the *read* Method

Though the default return format of the *read* method is WDDX, you can also request the package as a string only or in JSON (JavaScript Object Notation). To do so, append "&returnformat=JSON" or "&returnformat=plain" at the end of the *read* URL to indicate the preferred return format. Thus, the possible parameter values for *returnformat* are:

1. WDDX (the default)
2. Plain (for a string only)
3. JSON

Routing Web Service API Summary

URL

https://my.naturalinsight.com/routingWebService.cfc

WSDL

https://my.naturalinsight.com/routingWebService.cfc?wsdl

Method

read Method

```
<wsdl:operation name="read" parameterOrder="authCd accessKey
startDate endDate">

  <wsdl:input message="impl:readRequest" name-
  e="readRequest"></wsdl:input>

  <wsdl:output message="impl:readResponse" name-
  e="readResponse"></wsdl:output>

  <wsdl:fault message="impl:CFCInvocationException" name-
  e="CFCInvocationException"></wsdl:fault>

</wsdl:operation>
```

read Parameters

1. required string *authCd*
2. required access key (*accessKey*) as a string
3. required *startDate* in ISO 8601 format

> **NOTE:** This date is compared to a **Route's** submit date.

4. required *endDate* in ISO 8061 format

> **NOTE:** This date is compared to a **Route's** submit date.

read Description

The Routing *read* method allows you to request a Routing data package by date range and returns the Routing data (i.e., submitted Routes in Natural Insight within the specified date range) directly.

> **NOTE:** For the definition of a **Route** in Natural Insight, refer to the section, **Routing Web Service on page 1**.

read Potential Error Messages

- Routing not enabled in the Natural Insight account:

```
<error>Routing is not enabled.</error>
```

- Invalid start date:

```
<error>startDate must be a valid date and time.</error>
```

- Invalid end date:

```
<error>endDate must be a valid date and time.</error>
```

```
<error>endDate must be after startDate.</error>
```

- Database error:

```
<error>Unhandled exception!...Error message...</error>
```