ATTENDEE INTERACTIVE

CONFERENCE MANAGEMENT SOFTWARE

REST Integration Guidelines

For AI-Evolution 4.0

Version 1.0

Date Created: September 1, 2009

Last Updated: Feb 10, 2015

FOR PROFESSIONAL USE ONLY

Notice of non-liability:

Attendee Interactive (AI) is providing the information in this document to you AS-IS with all faults. AI makes no warranties of any kind (whether express, implied or statutory) with respect to the information contained herein. AI assumes no liability for damages (whether direct or indirect), caused by errors or omissions, or resulting from the use of this document or the information contained in this documents or resulting from the application or use of the product or service described herein. AI reserves the right to make changes to any information herein without further notice.

AI does not guarantee that the features described in this document will be announced or made available to anyone in the future.

Content will vary by client. Some data responses may contain HTML or markdown elements.

The information contained in this documentation is proprietary and confidential information of Attendee Interactive LLC. Such information is not to be disclosed outside of your organization, nor is it to be duplicated, used, or disclosed, in whole or in part for any purpose other than to evaluate this proposal without Attender Interactive's written consent.

Summary

This document outlines the REST (Representational State Transfer) RPC (Remote Procedure Call) operations allowed to obtain data from an AI-Evolution powered web site.

Intended Audience

This document is written for developers familiar with REST style application programming and XML data responses. For more information about REST please check out the following:

The Beauty of REST by Jon Udell
Second Generation Web Services by Paul Prescod Representational
State Transfer in the Wikipedia

Security

All requests must contain an API key used to authenticate the validity of the request. The API key is an alphanumeric string, e.g. 65121C2FPFAB254D13DCE86EB82DF38D, which is issued to each vendor separately. Different keys are issued for each site.

REST Request Format

REST is the simplest request format to use - it's a simple HTTP GET or POST action.

The REST Endpoint URL is TBD

To request the sessions' service, invoke like this:

```
http://XXX/?do=cnt.getservice&service=XXXXX
```

By default, all requests will send a REST Response payload.

REST Response Format

REST is the simplest response format to use - it's a simple XML block.

A method call returns this:

```
<?xml version="1.0" encoding="utf-8" ?>
<rst stat="ok">
        [xml-payload-here] </rst>
```

If an error occurs, the following is returned:

```
<?xml version="1.0" encoding="utf-8" ?>
<rst stat="fail">
        <err code="[error-code]" msg="[error-message]" /> </rst>
```

Additional Info

* - Indicates a required parameter for a request.

Event Service Options

service=getEvents

Returns an XML payload containing a full listing of all event or session data. Event or session data consists primarily of the name, date, time description and location of events. Note: the tag, AI_ATTENDEEID, may be used within data elements and serves as a placeholder to be replaced by a valid attendee ID.

This method is only available in its entirety between 7PM and 7AM. Between 7AM and 7PM the method will at most 5 event entries.

Sample Request URL: http://.../?do=cnt.getservice&service=getEvents

Parameter Options:

***apiKey:** Valid API key
statusFlag: Valid values: 1-Active; 0-Inactive; -1-All; Used to indicate if the event is active or inactive. Generally inactive events are events that have been cancelled or removed by the client.
publicUseFlag: Valid values: 1=Public; 0=Private; -1=All; Used to determine the types of events to include. Public events are viewable to the general population. Private events are events clients are witholding from the general viewing public

approvedFlag: Valid values: 1=Approved; 0=Not Approved; -1=Both; Used to determine the types of events to include. Note that a site setting can be used to use approvedFlag=1 by default. However, passing the parameter via the url will override the site setting parameter.

categoryID: Numeric value; Use the 'getCategories' REST call to pull a list of valid categories.
plannerID: Numeric value containing a valid planner.

startsOnDate: Date value containing a valid date. Use the 'getSearchOptions' call to obtain a valid list of dates.

Sample XML Payload: See REST_getEvents.xml

service=getEvent

Returns an XML payload containing the details of a single event.

Sample Request URL: http://.../?do=cnt.getservice&service=getEvent

Parameter Options:

***apiKey:** Valid API key
***eventID**: Numeric value of an event.

Sample XML Payload: See REST_getEvent.xml

service=getEventSearchOptions

Returns an XML payload containing search options for the getSearchResults service. The results of this call are used to obtain valid search values which can be passed to the 'getSearchResults' request.

Sample Request URL: http://.../?do=cnt.getservice&service=getEventSearchOptions

Parameter Options:

***apiKey:** Valid API key

Sample XML Payload: See REST_getEventSearchOptions.xml

service=getEventSearchResults

Returns an XML payload containing an event listing. This data set returned is smaller than the getEvents request and intend for listing results. Use the result in combination with the getEvent call to pull addition information on a specific event.

Sample Request URL: http://.../?do=cnt.getservice&service=getEventSearchResults

Parameter Options:

***apiKey:** Valid API key
searchText: Text string used to perform a general search of the events. This parameter is used to find matches in the title, description, objectives or speaker names.
categoryID: Numeric value containing a valid category. Use the 'getSearchOptions' call to obtain a valid list of categories
locationID: Numeric value containing a valid location. Use the 'getSearchOptions' call to obtain a valid list of locations
speakerID: Numeric value containing a valid speaker. Use the 'getSearchOptions' call to obtain the list of possible speakers.
startsOnDate: Date value containing a valid date. Use the 'getSearchOptions' call to obtain a valid list of dates.
startsOnTime: Time value in 12 hour format. E.g. 6:30PM
statusFlag: Valid values: 1-Active; 0-Inactive; -1-All; Used to indicate if the event is active or inactive. Generally inactive events are events that have been cancelled or removed by the client.

Sample XML Payload: See REST_getEventSearchResults.xml

service=getEventsCreditsSummary

Returns an XML payload containing a full listing of event or session credit data. Event or session data consists primarily of the name, date, time description and location of events. Credit data primarily consists of the total credits claimed and people who have claimed those credits. Note: The tag, AI_ATTENDEEID, may be used within data elements and serves as a placeholder to be replaced by a valid attendee ID.

This method is only available in its entirety between 7PM and 7AM. Between 7AM and 7PM the method will at most 5 event entries.

Sample Request URL: http://.../?do=cnt.getservice&service=getEventsCreditsSummary

Parameter Options:

***apiKey:** Valid API key
statusFlag: Valid values: 1-Active; 0-Inactive; -1-All; Used to indicate if the event is active or inactive. Generally inactive events are events that have been cancelled or removed by the client.

publicUseFlag: Valid values: 1=Public; 0=Private; -1=All; Used to determine the types of events to include. Public events are viewable to the general population. Private events are events clients are withholding from the general viewing public

approvedFlag: Valid values: 1=Approved; 0=Not Approved; -1=Both; Used to determine the types of events to include. Note that a site setting can be used to use approvedFlag=1 by default. However, passing the parameter via the url will override the site setting parameter.

categoryID: Numeric value; Use the 'getCategories' REST call to pull a list of valid categories.
plannerID: Numeric value containing a valid planner.

startsOnDate: Date value containing a valid date. Use the 'getSearchOptions' call to obtain a valid list of dates.

Sample XML Payload: See REST_getEventsCreditsSummary.xml

service=getEventCreditsSummary

Returns an XML payload containing the credit details of a single event.

Sample Request URL: http://.../?do=cnt.getservice&service=getEventCreditsSummary

Parameter Options:

***apiKey:** Valid API key
***eventID**: Numeric value of an event.

Sample XML Payload: See REST_getEventCreditsSummary.xml

service=getEventsCreditsByRegTypeSummary

Returns an XML payload containing a full listing of event or session credit data grouped by registration type. Event or session data consists primarily of the name, date, time description and location of events. Credit data primarily consists of the total credits claimed and people who have claimed those credits. Note: The tag, AI_ATTENDEEID, may be used within data elements and serves as a placeholder to be replaced by a valid attendee ID. The data excludes people who do not have a registration type.

This method is only available in its entirety between 7PM and 7AM. Between 7AM and 7PM the method will at most 5 event entries.

Sample Request URL: http://.../?do=cnt.getservice&service=getEventsCreditsByRegTypeSummary

Parameter Options:

***apiKey:** Valid API key
statusFlag: Valid values: 1-Active; 0-Inactive; -1-All; Used to indicate if the event is active or inactive. Generally inactive events are events that have been cancelled or removed by the client.
publicUseFlag: Valid values: 1=Public; 0=Private; -1=All; Used to determine the types of events to include. Public events are viewable to the general population. Private events are events clients are withholding from the general viewing public

approvedFlag: Valid values: 1=Approved; 0=Not Approved; -1=Both; Used to determine the types of events to include. Note that a site setting can be used to use approvedFlag=1 by default. However, passing the parameter via the url will override the site setting parameter.
categoryID: Numeric value; Use the 'getCategories' REST call to pull a list of valid categories.
plannerID: Numeric value containing a valid planner. **startsOnDate:** Date value containing a valid date. Use the 'getSearchOptions' call to obtain a valid list of dates.

Sample XML Payload: See REST_getEventsCreditsByRegTypeSummary.xml

service=getEventCreditsByRegTypeSummary

Returns an XML payload containing the credit details grouped by registration type of a single event.

Sample Request URL: http://.../?do=cnt.getservice&service=getEventCreditsByRegTypeSummary

Parameter Options:

***apiKey:** Valid API key
***eventID**: Numeric value of an event.

Sample XML Payload: See REST_getEventCreditsByRegTypeSummary.xml

Exhibitor Service Options

service=getExhibitors

Returns an XML payload containing a full listing of all exhibitor data. Exhibitor data consists primarily of the company name, address and contact information.

Sample Request URL: http://.../?do=cnt.getservice&service=getExhibitors

Parameter Options:

***apiKey:** Valid API key

Sample XML Payload: See REST_getExhibitors.xml

service=getExhibitor

Returns an XML payload containing a full listing of one exhibitor. Exhibitor data consists primarily of the company name, address and contact information.

Sample Request URL: http://.../?do=cnt.getservice&service=getExhibitor

Parameter Options:

***apiKey:** Valid API key
***exhibitorID:** Numeric value of an exhibitor

Sample XML Payload: See REST_getExhibitor.xml

service=getExhibitorSearchOptions

Returns an XML payload containing search options for the getExhibitorSearchResults service. The results of this call are used to obtain valid search values which can be passed to the 'getExhibitorSearchResults' request.

Sample Request URL: http://.../?do=cnt.getservice&service=getExhibitorSearchOptions

Sample XML Payload: See REST_getExhibitorSearchOptions.xml

service=getExhibitorSearchResults

Returns an XML payload containing an exhibitor listing. This data set returned is smaller than the getExhibitors request and intend for listing results. Use the result in combination with the getExhibitor call to pull addition information on a specific event.

Sample Request URL: http://.../?do=cnt.getservice&service=getExhibitorSearchResults

Parameter Options:

***apiKey:** Valid API key
searchText: Text string used to perform a general search of the events. This parameter is used to find matches in the title, description, objectives or speaker names.
categoryID: Numeric value containing a valid category. Use the 'getSearchOptions' call to obtain a valid list of categories

Sample XML Payload: See REST_getExhibitorSearchResults.xml

Abstract Service Options

service=getAbstracts

Returns an XML payload containing a full listing of all abstract data. Abstract data consists primarily of the title, abstract number, purpose, materials & methods, results & conclusions.

Sample Request URL: http://.../?do=cnt.getservice&service=getAbstracts

Parameter Options:

***apiKey:** Valid API key **includeEventData:** Boolean value (0 or 1). Default is 0.
categoryID: Numeric value containing a valid category. Use the 'getSearchOptions' call to obtain a valid list of categories
speakerID: Numeric value containing a valid speaker. Use the 'getSearchOptions' call to obtain the list of possible speakers.
abstractTypeID: Numeric value containing a valid abstract type. Use the 'getSearchOptions' call to obtain the list of possible abstract types.
acceptedFlag: Boolean value (0 or 1). Default is 1.

Sample XML Payload: See REST_getAbstracts.xml

service=getAbstract

Returns an XML payload containing a full listing of one abstract. Abstract data consists primarily of the title, abstract number, purpose, materials & methods, results & conclusions. Note: The result set provides urls to the figures or documents which are actual files.

Sample Request URL: http://.../?do=cnt.getservice&service=getAbstract

Parameter Options:

***apiKey:** Valid API key
***abstractID:** Numeric value of an abstract.

Sample XML Payload: See REST_getAbstract.xml

service=getAbstractSearchOptions

Returns an XML payload containing search options for the getAbstractSearchResults service. The results of this call are used to obtain valid search values which can be passed to the 'getAbstractSearchResults' request.

Sample Request URL: http://.../?do=cnt.getservice&service=getAbstractSearchOptions

Sample XML Payload: See REST_getAbstractSearchOptions.xml

service=getAbstractSearchResults

Returns an XML payload containing an abstract listing. This data set returned is smaller than the getAbstracts request and intend for listing results. Use the result in combination with the getAbstract call to pull addition information on a specific abstract.

Sample Request URL: http://.../?do=cnt.getservice&service=getAbstractSearchResults

Parameter Options:

***apiKey:** Valid API key
searchText: Text string used to perform a general search of the events. This parameter is used to find matches in the title, description, objectives or speaker names.
categoryID: Numeric value containing a valid category. Use the 'getSearchOptions' call to obtain a valid list of categories
speakerID: Numeric value containing a valid speaker. Use the 'getSearchOptions' call to obtain the list of possible speakers.
abstractTypeID: Numeric value containing a valid abstract type. Use the 'getSearchOptions' call to obtain the list of possible abstract types. **acceptedFlag:** Boolean value (0 or 1). Default is 1.

Sample XML Payload: See REST_getAbstractSearchResults.xml

Attendees Service Options

service=getSpeakers

Returns an XML payload containing a full listing of all speaker data. Speakers are defined as attendees whom are participating in the meeting in a specific capacity. E.g. Presenting in a session or Moderating a session or Presenting a Poster. This service also returns the schedule for each speaker. This includes all itinerary items that are 'active' (can be changed via includeActiveStatus parameter), and 'public'.

Sample Request URL: http://.../?do=cnt.getservice&service=getSpeakers

Parameter Options:

***apiKey:** Valid API key
isScheduledFor: Numeric value of (1,2 or 3) – This filters the speakers returned by schedule type: Valid values; 1 = For Event, 2 = For Abstract, 3 = For Event OR Abstract. Note that this parameter only affects whether or not a speaker record is included in the results set. It does not change the itinerary items that are returned in each speaker's schedule block. For example, if isScheduledFor=2, Any speaker who is linked to at least 1 Abstract will be shown. Any 'active', and 'public' itinerary items that are linked to an Event will still be shown in the schedule for this speaker.
includeActiveStatus: Numeric value of (0,1, or 99) – Determines if active, inactive, or ALL itinerary items are shown in the <schedule> block.
formIDs: Comma delimited list of form ids
modifiedOnAfter: Valid date string. Using this filter will return speakers who have updated contact information, schedule, or form responses after/on this date.
modifiedOnBefore: Valid date string. Using this filter will return speakers who have updated contact information, schedule, or form responses before/on this date.

Sample XML Payload: See REST_getSpeakers.xml

service=getSpeaker

Returns an XML payload containing a full listing of one speaker. Speaker data consists of their contact information, bio and photo. Note Photo's are returned as a url to the actual photo. This service also returns the schedule for the speaker. This includes all itinerary items that are 'active' (can be changed via includeActiveStatus parameter), and 'public'.

Sample Request URL: http://.../?do=cnt.getservice&service=getSpeaker

Parameter Options:

***apiKey:** Valid API key
***speakerID:** Numeric value of a speaker.
includeActiveStatus: Numeric value of (0,1, or 99) – Determines if active, inactive, or ALL itinerary items are shown in the <schedule> block.
isScheduledFor: Numeric value of (1,2 or 3) – This filters the speakers returned by schedule type: Valid values; 1 = For Event, 2 = For Abstract, 3 = For Event OR Abstract. Note that this parameter only affects whether or not a speaker record is included in the results set. It does not change the

itinerary items that are returned in each speaker's schedule block. For example, if isScheduledFor=2, Any speaker who is linked to at least 1 Abstract will be shown. Any 'active', and 'public' itinerary items that are linked to an Event will still be shown in the schedule for this speaker.
formIDs: Comma delimited list of form ids

Sample XML Payload: See REST_getSpeaker.xml

service=getAttendeesRegData

Returns an XML payload containing a full listing of all attendee purchases. For each attendee their purchase or order history will be returned in the XML payload. This service is only allowed to be run between 11 PM EST and 5 AM EST

Sample Request URL: http://.../?do=cnt.getservice&service=getAttendeesRegData

Parameter Options:

***apiKey:** Valid API key
attendeeID: Numeric value, Identifies the attendee record to update.
startInitial: 1 character used to indicate the starting initial of the last name range to include. E.g. A
endInitial: 1 character used to indicate the starting initial of the last name range to include. If the 'startInitial' is provided but no 'endInitial' is provided the system uses the 'startInitial' as the 'endInitial'. E.g. B
formID: Numeric value of a demographic form

Sample XML Payload: See REST_ getAttendeesRegData.xml – Only available on request.

service=getAttendeesItineraryData

Returns an XML payload containing a full listing of all attendee itinerary data. For each attendee their itinerary will be returned in the XML payload. This service is only allowed to be run between 11 PM EST and 5 AM EST. Note: For clients using the Attendee Itinerary Update service, only itinerary items with the attributes='ReadWrite' can be modified.

Sample Request URL: http://.../?do=cnt.getservice&service=getAttendeesItineraryData

Parameter Options:

***apiKey:** Valid API key
attendeeID: Numeric value; Identifies the attendee record to update.
startInitial: 1 character used to indicate the starting initial of the last name range to include. E.g. A
endInitial: 1 character used to indicate the starting initial of the last name range to include. If the 'startInitial' is provided but no 'endInitial' is provided the system uses the 'startInitial' as the 'endInitial'. E.g. B
insertedInLastHoursSpan: Used to indicate the number of hour of newly inserted records to include. The default is 24. This must be a valid integer between 1 and 26,280.

Sample XML Payload: See REST_getAttendeesItineraryData.xml – Only available on request.

service=getAttendeesFormResponses

Returns an XML payload containing a full listing of attendee contact data and form responses for the specified form. For each attendee a 'formResponses' node returned in the XML payload. This service is only allowed to be run between 11 PM EST and 5 AM EST.
Sample Request URL: http://.../?do=cnt.getservice&service=getAttendeesItineraryData

Parameter Options:

***apiKey:** Valid API key
***formID:** Numeric value; Identifies the form for which to retrieve responses.
attendeeID: Numeric value; Identifies the attendee record to update.
startInitial: 1 character used to indicate the starting initial of the last name range to include. E.g. A
endInitial: 1 character used to indicate the starting initial of the last name range to include. If the 'startInitial' is provided but no 'endInitial' is provided the system uses the 'startInitial' as the 'endInitial'. E.g. B

Sample XML Payload: See REST_getAttendeesFormResponses.xml – Only available on request.

service=updateItinerary

Returns an XML payload containing the results of the attendee itinerary data update. The 'stat' attribute is used to indicate the success or failure of the request. Extended description of all options are listed below.

Sample Request URL: http://.../?do=cnt.getservice&service=updateItinerary&[Parameter List]

Parameter Options:

***apiKey:** Valid API key
***attendeeID:** Identifies the attendee record to update. Always required.
***itineraryID:** Identifies the itinerary record to update. Valid options: update, delete; The update option is assumed if no action is provided.
abstractID: Identifies the abstract record being added, updated or removed
eventID: Identifies the event record being added, updated or removed
exhibitorID: Identifies the exhibitor record being added, updated or removed title. Title of the activity. Limit 300 characters. *Required if record is not linked to an event, abstract or exhibitor. If linked to an event, abstract or exhibitor, title is ignored.*
Description: Description of the event. Limit 500 characters.
startsOn: Date & Time of activity. Format: mm/dd/yyyy hh:mm tt, e.g. 05/01/2011 12:00 PM. Date / Time will be ignored if not a valid format or if passed with an event or abstract which already have a time associated.
endsOn: Date & Time of activity. Format: mm/dd/yyyy hh:mm tt, e.g. 05/01/2011 12:00 PM. Date / Time will be ignored if not a valid format or if passed with an event or abstract which already have a time associated.

The results of this call are either success for failure. Both XML payloads are described next.

Sample Success XML Payload:

```
<?xml version='1.0' encoding='UTF-8'?>
<rst stat='ok' version='1.0'>
<result stat='ok' id='123456' />
</rst>
</xml>
```

The ‘id’ attribute will contain the updated itinerary record identifier. This value can be ignored on updates but must be stored when adding items in order to perform updates or deletions at a later date.

Sample Failure XML Payload:

```
<?xml version='1.0' encoding='UTF-8'?>
<rst stat='ok' version='1.0'>
<result stat='fail' msg='see reason nodes'>
      <reason>Scheduled items cannot be altered.</reason>
</result>
</rst>
</xml>
```

Sample Abstract Parameter Lists: Add:
`attendeeID=1093&abstractID=10000&startsOn=06/24/2011 12:00 PM&endsOn=06/24/2011 01:00 PM`

Add: `attendeeID=10930&abstractID=10000`

Update: `attendeeID=10930&itineraryID=11937&abstractID=10000&startsOn=06/24/2011 12:00 PM&endsOn=06/24/2011 01:00 PM`

Delete: `attendeeID=10930&itineraryID=11937&abstractID=10000&action=delete`

Sample Event Request:

Add: `attendeeID=10930&eventID=10000`

Delete: `attendeeID=10930&itineraryID=11937&eventID=10000&action=delete`

Sample Exhibitor Request:

Add: `attendeeID=10930&exhibitorID=10000&startsOn=06/24/2011 12:00 PM&endsOn=06/24/2011 01:00 PM`

Update: `attendeeID=10930&itineraryID=11937&exhibitorID=10000&startsOn=06/24/2011 12:00 PM&endsOn=06/24/2011 01:00 PM`

Delete: `attendeeID=10930&itineraryID=11937&exhibitorID=10000&action=delete`

Sample Personal Activity Request:

Add: `attendeeID=10930&title=My Personal Activity&startsOn=06/24/2011 12:00 PM&endsOn=06/24/2011 01:00 PM&description=This is a description`

Update: `attendeeID=10930&itineraryID=11937&title=My Personal Activity&startsOn=06/24/2011 12:00 PM&endsOn=06/24/2011 01:00 PM&description=This is a description`

Delete: `attendeeID=10930&itineraryID=11937&action=delete`

service=addAttendeeActivity

Returns an XML payload containing the results of the attendee add activity method. The 'stat' attribute is used to indicate the success or failure of the request. Extended description of all options are listed below.

Sample Request URL: http://.../?do=cnt.getservice&service=addAttendeeActivity&[Parameter List]

Parameter Options:

***apiKey:** Valid API key
***attendeeID:** Numeric value used to identify the attendee record to update. attendeeID or registrationNumber required.
registrationNumber: String value used to identify the attendee record to update. attendeeID or registrationNumber required.
eventIDs: Comma delimited list of event to be added to a person's activity list. eventIDs or eventID required
eventID: Numeric value identifying an event to be added to a person's activity list. eventIDs or eventID required.

The results of this call are either success for failure. Both XML payloads are described next.

Sample Success XML Payload: Includes list of the activity identifiers added.

```
<?xml version='1.0' encoding='UTF-8'?>
<rst stat='ok' version='1.0'>
<result stat='ok' />
      <activities>
            <activity id='1234' />
            <activity id='1235' />
      </activities>
</rst>
</xml>
```

The 'id' attribute will contain the added activity record identifier. This value can be ignored on updates but must be stored when adding items in order to perform updates or deletions at a later date.

Sample Failure XML Payload:

```
<?xml version='1.0' encoding='UTF-8'?>
<rst stat='ok' version='1.0'>
<result stat='fail' msg='see reason nodes'>
      <reason>Scheduled items cannot be altered.</reason>
</result>
</rst>
</xml>
```

Sample Parameter Lists:

Sample Multiple Events Request:
Add: `attendeeID=10930&eventIDs=10000,1001,1002`

service=getPersonActivities

Returns an XML payload containing the results of the attendee get activity method. The 'stat' attribute is used to indicate the success or failure of the request. Extended description of all options are listed below.

Sample Request URL: http://.../?do=cnt.getservice&service= getPersonActivities&[Parameter List]

Parameter Options:

***apiKey:** Valid API key
attendeeID: Numeric value; Identifies the attendee record to retrieve.
lastNameInitialStart: 1 character used to indicate the starting initial of the last name range to include. E.g. A
lastNameInitialEnd: 1 character used to indicate the starting initial of the last name range to include. If the 'lastNameInitialStart' is provided but no 'lastNameInitialEnd' is provided the system uses the 'lastNameInitialStart' as the 'lastNameInitialEnd'. E.g. B

Sample XML Payload: See REST_ getPersonActivities.xml – Only available on request

service=getAttendeesCreditActivity

Returns an XML payload containing and attendee(s) data and credit activity grouped by planner/activity. Credit activities consist of the claimed activities, the activities' displayed credit, the activities' credit types/accreditation bodies (based on the attendee's registration type), and the credit awarded for each credit type/accreditation body.

Sample Request URL: http://.../?do=cnt.getservice&service= getAttendeesCreditActivity&[Parameter List]

Parameter Options:

***apiKey:** Valid API key
attendeeID: Numeric value; Identifies the attendee record to retrieve.
attendeesID: List of numeric values; Identifies attendee records to retrieve.
memberNumber: String value. Identifies the attendee record to retrieve.
memberNumberPartial: String value. Partial member number used to search/identify attendee records to retrieve.
lastNameInitialStart: 1 character used to indicate the starting initial of the last name range to include. E.g. A. Must be used with lastNameInitialEnd
lastNameInitialEnd: 1 character used to indicate the starting initial of the last name range to include. E.g. B. Must be used with lastNameInitialStart
plannerID: Numeric value; Used to filter the activities. Identifies the planner/activity to include.
claimedOnAfter: Date; Used to filter the activities.

claimedOnBefore: Date; Used to filter the activities

Sample XML Payload: See REST_getAttendeesCreditActivity.xml

service=getAttendeesCreditActivitySummary

Returns an XML payload containing and attendee(s) data and credit activity summaries grouped by planner/activity. The credit activities summaries consist of the total number of activities, display credits sum, awarded credit sum for each credit type/accreditation body.

Sample Request URL: http://.../?do=cnt.getservice&service=

getAttendeesCreditActivitySummary&[Parameter List]

Parameter Options:

***apiKey:** Valid API key
attendeeID: Numeric value; Identifies the attendee record to retrieve.
attendeesID: List of numeric values; Identifies attendee records to retrieve.
memberNumber: String value. Identifies the attendee record to retrieve.
memberNumberPartial: String value. Partial member number used to search/identify attendee records to retrieve.
lastNameInitialStart: 1 character used to indicate the starting initial of the last name range to include. E.g. A. Must be used with lastNameInitialEnd
lastNameInitialEnd: 1 character used to indicate the starting initial of the last name range to include. E.g. B. Must be used with lastNameInitialStart
plannerID: Numeric value; Used to filter the activities. Identifies the planner/activity to include.
claimedOnAfter: Date; Used to filter the activities.
claimedOnBefore: Date; Used to filter the activities

Sample XML Payload: See REST_getAttendeesCreditActivitySummary.xml

Category Service Options

service=getCategories

Returns an XML payload containing a full listing of all category data. Category data consists primarily of the category name and sub-categories.

Sample Request URL: http://.../?do=cnt.getservice&service=getCategories

Parameter Options:

***apiKey:** Valid API key
categoryTypeID: Integer valid values: 1= Event, 2=Exhibitor & 3=Abstract If not provided the system defaults to sending Event categories.
includeAdminOnlyCategories: 0 or 1 (represents true or false).

Sample XML Payload: See REST_getCategories.xml

Authentication Service Options

service=authenticatePerson

Returns an XML payload used to indicate if the sign in credentials provided are valid. When valid credentials are provided the XML payload will return the person's first name, last name, company and email.

Sample Request URL: http://.../?do=cnt.getservice&service=authenticatePerson

Parameter Options:

***apiKey:** Valid API key
***username:** Text string of the person's username
***password:** Text string of the person't password

Sample Valid XML Payload: See REST_authenticatePerson.xml
Sample Invalid XML Payload:

```
<authentication result='invalid'><firstname /><lastname /><company
/><email /></authentication>
```

service=authenticateCompany

Returns an XML payload used to indicate if the sign in credentials provided are valid. When valid credentials are provided the XML payload will return the company contact's first name, last name, company and email.

Sample Request URL: http://.../?do=cnt.getservice&service=authenticateCompany

Parameter Options:

***apiKey:** Valid API key
***username:** Text string of the person's username
***password:** Text string of the person't password
Sample Valid XML Payload: See REST_authenticateCompany.xml
Sample Invalid XML Payload:

```
<authentication result='invalid'><firstname /><lastname /><company
/><email /></authentication>
```

service=authenticateVendor

Returns an XML payload used to indicate if the sign in credentials provided are valid. When valid credentials are provided the XML payload will return the vendor company and email.

Sample Request URL: http://.../?do=cnt.getservice&service=authenticateVendor

Parameter Options:

***apiKey:** Valid API key
***username:** Text string of the person's username

***password:** Text string of the person’s password

Sample Valid XML Payload: See REST_authenticateVendor.xml
Sample Invalid XML Payload:

```
<authentication result='invalid'><firstname /><lastname /><company
/><email /></authentication>
```

Roomset Service Options

service=getRoomSets

Returns an XML payload containing a full listing of all Room Set data.

Sample Request URL: http://.../?do=cnt.getservice&service=getRoomSets

Parameter Options:

***apiKey:** Valid API key **postFlag:** 0 or
1 (represents true or false).

Sample XML Payload: See REST_getRoomSets.xml

service=getRoomSet

Returns an XML payload containing a full listing of one Room Set. Room Set data consists primarily of Room, Location, Room Set Details (supplies), and Vendors. Note: The result set provides urls to the figures or documents which are actual files.

Sample Request URL: http://.../?do=cnt.getservice&service=getRoomSet

Parameter Options:

***apiKey:** Valid API key

***roomSetID:** Numeric value of a Room Set.

Sample XML Payload: See REST_getRoomSet.xml

service=getRoomSetSearchOptions

Returns an XML payload containing search options for the getRoomSetSearchResults service. The results of this call are used to obtain valid search values which can be passed to the 'getRoomSetSearchResults' request.

Sample Request URL: http://.../?do=cnt.getservice&service=getRoomSetSearchOptions

Sample XML Payload: See REST_ getRoomSetSearchOptions.xml

Parameter Options:

***apiKey:** Valid API key

service= getRoomSetSearchResults

Returns an XML payload containing a Room Set listing. This data set returned is smaller than the getRoomSets request and intend for listing results. Use the result in combination with the getRoomSet call to pull addition information on a specific RoomSet.

Sample Request URL: http://.../?do=cnt.getservice&service=getRoomSetSearchResults

Parameter Options:

***apiKey:** Valid API key
searchText: Text string used to perform a general search of the Room Sets. Searches for matches in function name, code, location, and room. **locationID:** Numeric value containing a valid locationID. Use the 'getRoomSetSearchOptions' call to obtain a valid list of locations
roomID: Numeric value containing a valid roomID. Use the 'getRoomSetSearchOptions' call to obtain a valid list of rooms. **startsOnDate:** Date value containing a valid date. Use the 'getSearchOptions' call to obtain a valid list of dates. **code:** text value containing a valid code. Use the 'getSearchOptions' call to obtain the list of possible Room Set Codes.
postFlag: Boolean value (0 or 1).

Sample XML Payload: See REST_getRoomSetSearchResults.xml

System Service Options

service=getEventTypes

Returns an XML payload containing a listing of event types. Event type data contains the type name and roles that may be assigned to an event of this type.

Sample Request URL: http://.../?do=cnt.getservice&service=getEventTypes

Parameter Options:

***apiKey:** Valid API key
type: Integer valid values: 1= Event, 2=Abstract
If not provided the system defaults to sending Event roles.

Sample XML Payload: See REST_getEventTypes.xml

service=getForm

Returns an XML payload containing form data. Form data includes form settings, questions and answers.

Sample Request URL: http://.../?do=cnt.getservice&service=getForm

Parameter Options:

***apiKey:** Valid API key
***formID:** Numeric value containing a valid form ID.

Sample XML Payload: See REST_getForm.xml

service=getLocations

Returns an XML payload containing a full listing of location and room data.

Sample Request URL: http://.../?do=cnt.getservice&service=getLocations

Parameter Options:

***apiKey:** Valid API key

Sample XML Payload: See REST_getLocations.xml

service=getPublicFields

Returns an XML payload containing public fields for one of the following: Event, Exhibitor, Abstract, Speaker. Field data will contain the field ID and public label.

Sample Request URL: http://.../?do=cnt.getservice&service=getPublicFields
Parameter Options:

***apiKey:** Valid API key

type: Integer valid values: 1= Event, 2=Exhibitor, 3=Abstract & 4=Speaker
If not provided the system defaults to sending Event fields.

Sample XML Payload: See REST_getPublicFields.xml

service=getScheduledRoles

Returns an XML payload containing roles used by Events or Abstracts. Role data will contain the role name, type, and public label.

Sample Request URL: http://.../?do=cnt.getservice&service=getScheduledRoles

Parameter Options:

***apiKey:** Valid API key
type: Integer valid values: 1= Event, 2=Abstract
If not provided the system defaults to sending Event roles.

Sample XML Payload: See REST_getScheduledRoles.xml

service=getRegistrationTypes

Returns an XML payload containing registration type data. Registration type data will contain the registration type name, code, description, and evaluation links. Note: The tag, AI_ATTENDEEID, may be used within data elements and serves as a placeholder to be replaced by a valid attendee ID. The tag, AI_EVENTID, may be used within data elements and serves as a placeholder to be replaced by a valid event ID.

Sample Request URL: http://.../?do=cnt.getservice&service=getRegistrationTypes

Parameter Options:

***apiKey:** Valid API key

Sample XML Payload: See REST_getRegistrationTypes.xml

service=getMenu

Returns an XML payload containing a menu item and its content pages.

Sample Request URL: http://.../?do=cnt.getservice&service=getMenu&[parameters list]

Parameter Options:

***apiKey:** Valid API key
***menuID:** Numeric value containing a valid menu id.

Sample XML Payload: See REST_getMenu.xml

service=getMeetingTimes

Returns an XML payload containing a listing of all active meeting times/time ranges.

Sample Request URL: http://.../?do=cnt.getservice&service=getMeetingTimes

Parameter Options:

***apiKey:** Valid API key

Sample XML Payload: See REST_getMeetingTimes.xml

service=getMeetingTime

Returns an XML payload containing a meeting time/time range. This payload includes the meeting time's type, code, name, description, and start and end date.

Sample Request URL: http://.../?do=cnt.getservice&service=getEventTypes

Parameter Options:

***apiKey:** Valid API key
meetingTimeID: Numeric value containing a valid meeting time id.

Sample XML Payload: See REST_getMeetingTime.xml

Activity/Planner Service Options

service=getActivities

Returns an XML payload containing activity details. This payload includes the activity's title, codes, start/end dates, description, location, and ACCME PARS information. This payload also contains a listing of the activity's public role assignments.

Sample Request URL: http://.../?do=cnt.getservice&service=getActivities

Parameter Options:

***apiKey:** Valid API key
plannerID: Numeric identifier for a valid planner. Use this parameter to request the details of a single planner.
activityID: Numeric identifier for a valid activity. Use this this parameter to request the details of a single activity.
startsOn: Date value containing a valid date. Use this parameter to request the details of any activity that begins on this date.
startsOnOrAfter: Date value containing a valid date. Use this parameter to request the details of any activity that begins on or after this date.
startsOnOrBefore: Date value containing a valid date. Use this parameter to request the details of any activity that begins on or before this date.
insertedOn: Date value containing a valid date. Use this parameter to request the details of any activity that was added to the system on this date.
insertedOnOrAfter: Date value containing a valid date. Use this parameter to request the details of any activity that was added to the system on or after this date.
insertedOnOrBefore: Date value containing a valid date. Use this parameter to request the details of any activity that was added to the system on or before this date.

Sample XML Payload: See REST_getActivities.xml

service=getActivity

Returns an XML payload containing an activity's details. This payload includes the activity's title, codes, start/end dates, description, location, and ACCME PARS information. This payload also contains a listing of the activity's public role assignments and event (if the activity is one-to-one)

Sample Request URL: http://.../?do=cnt.getservice&service=getActivity

Parameter Options:

***apiKey:** Valid API key

plannerID: Numeric identifier for a valid planner. Use this parameter to request the details of a single planner.
activityID: Numeric identifier for a valid activity. Use this this parameter to request the details of a single activity.

Sample XML Payload: See REST_getActivity.xml