



ACTIVE Results REST API Reference

V1.9

April 10th, 2019

ACTIVE Network, LLC

© 2019 Active Network, LLC, and/or its affiliates and licensors. All rights reserved. No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any computer language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual, or otherwise, without prior written permission. You have limited permission to make hard copy or other reproductions of any machine-readable documentation for your own use, provided that each such reproduction shall carry this copyright notice. No other rights under copyright are granted without prior written permission. The document is not intended for production and is furnished "as is" without warranty of any kind. All warranties on this document are hereby disclaimed including the warranties of merchantability and fitness for a particular purpose.

ACTIVE Network, LLC
717 North Harwood Street
Suite 2500
Dallas, TX 75201

About ACTIVE Network, LLC

ACTIVE Network, LLC is a technology and media company that helps millions of people find and register for things to do and helps organizations increase participation in their activities and events.

For more information about ACTIVE Network, LLC products and services, please visit ACTIVEnetwork.com.

Table of Contents

Revision History	4
1. Getting Started	6
2. Integration Checklist	6
3. Terminology	6
4. Authentication and Access	7
4.1. Enabling the Public API License.....	7
4.2. Proxy Authentication Token.....	7
5. HTTP Status Codes	8
6. Methods	9
6.1. Creating an event.....	9
6.2. Updating an event	11
6.3. Creating a route.....	13
6.4. Deleting a route.....	16
6.5. Querying event routes.....	17
6.6. Importing a participant's results.....	21
6.7. Querying a result's import status.....	25

Revision History

Version	Date	Description
V1.0	May 18 th , 2018	Initial draft
V1.1	November 5 th , 2018	Creating an event: added Required/optional column. Creating a route: added Required/optional column.
V1.2	November 13 th , 2018	Importing a participant's results: added gender field.
V1.3	December 6 th , 2018	Creating a route API: corrected the API path . Importing a participant's results API: corrected the splitTime description to: "Cumulative split time".
V1.4	December 11 th , 2018	Creating an event API: changed address {- country, - stateProvince, - city} fields from Optional to Required .
V1.5	December 19 th , 2018	Creating an event API: If the event is an existing AW Endurance event, then it is NOT necessary to call this API to create the event. Importing a participant's results API: <ul style="list-style-type: none"> • Changed chipTimeResult, gunTimeResult and gunTime fields from Required to Optional. • Added optional city and stateProvince fields. • Added required finalResult field.
V1.6	January 3 rd , 2019	Added Terminology chapter. Added descriptions to each API. Creating an event API: <ul style="list-style-type: none"> • Clarified startDate datatype description. • Removed optional timeZone and logo fields. Updating an event API: <ul style="list-style-type: none"> • Added Required/Optional column. • Clarified startDate datatype description. • Removed optional timeZone field. Creating a route API: <ul style="list-style-type: none"> • Added additional routeLocations in the request example to make it correspond with the example in the Importing a participant's results API. • Added the default false state for the optional useChipTime field. • Enumerated the distanceUnit data type (KM, MI or MILE). • Changed routeLocations list to optional and clarified that at least one location corresponding to the start is required in the list. Importing a participant's results API: <ul style="list-style-type: none"> • Changed finishTime to optional. • Clarified finishTime and gunTime data type description. • Clarified averageSpeed units as KM/H. • Changed divisionRanks list to required. • Clarified description of – rank field (must be non-zero, positive number). • Clarified splitResults list ordering must be the same as the Creating a route > routeLocations list ordering, and the first location being the start does not have a splitTime or averageSpeed. Querying a result's import status API: <ul style="list-style-type: none"> • Enumerated the status field.
V1.7	February 1 st , 2019	For all APIs: <ul style="list-style-type: none"> • Changed definition of the externalApplicationId parameter to the event's source application.

		<p>Creating an event API:</p> <ul style="list-style-type: none"> • Added timingSystem parameter <p>Updating an event API:</p> <ul style="list-style-type: none"> • Added timingSystem parameter <p>Creating a route API:</p> <ul style="list-style-type: none"> • Added stages list <p>Importing a participant's results API:</p> <ul style="list-style-type: none"> • Added stageSequence to divisionRanks list • Added stageResults list • Changed finalResult to optional
V1.7a	February 1 st , 2019	<p>Creating an event API:</p> <ul style="list-style-type: none"> • externalApplicationId: added note regarding autobinding of participants' results and their AW Endurance registrations if this parameter is set to activeworksendurance • timingSystem: changed to optional <p>Updating an event API:</p> <ul style="list-style-type: none"> • timingSystem: changed to optional <p>For all APIs except the Creating an event API:</p> <ul style="list-style-type: none"> • externalApplicationId: added note: "Must be the same as when set using the Creating an event API."
V1.8	February 13 th , 2019	<p>Creating an event API:</p> <ul style="list-style-type: none"> • Removed note on conditions for creating an event. <p>Importing a participant's results API:</p> <ul style="list-style-type: none"> • Added penaltyTime
V1.9	April 10 th , 2019	Added Querying event routes API .

1. Getting Started

Welcome to the ACTIVE Results REST API Reference. The ACTIVE Results Service allows customers to create events and import participants' results.

The API uses HTTP methods and a RESTful endpoint structure. You can use the API to build add-ons for ACTIVE Results, integrate between ACTIVE Results and other applications, or script interactions with ACTIVE Results. This document depicts the REST resources available in ACTIVE Results.

2. Integration Checklist

Use the following checklist to identify the tasks required for successful integration with the ACTIVE Results API. If you require any assistance, please contact your Account Manager.

Item	Integration task	✓
1	Contact your Account Manager to: <ul style="list-style-type: none"> • Sign an API subscription agreement • Enable your Public API license • Request an API token for your organization 	
2	Develop your client application for calling the API.	
3	Test your client application.	

3. Terminology

This API uses the following terminology to describe results data:

Term	Description	Example
Event	A running or triathlon event hosted by the organizer.	Brighton Marathon Weekend 2018
Route	A single event can contain one or many routes. ACTIVE Results uses routes to understand how to group the results information for each race.	If the timer wishes to publish results to a single event page on ACTIVE Results for an event weekend consisting of a 5k, 10k and Marathon, then the data provided needs to consist of 3 routes: Route 1: Marathon Route 2: 10k Route 3: 5k
Route Group	A grouping of people within a single route, such as by age groups or divisions. There can be an unlimited number of route	Overall Overall Male Female 30-34 Male Masters

	groups per route, but there must be at least one.	Military
Route Location	Timing/split point locations for each route. May include common split point names such as 5k, 10k, Half, etc. There is no restriction to the naming convention for route locations.	5k 10k Half Swim finish Bike start
Split Result	The cumulative total time at each route location for each participant	5k split - 00:25:31
Stage	A sport-specific (or transition) section of a route.	Swimming Biking Running Transition
Stage results	A participant's final result for a specific stage.	PT2H22M59S
Place	Rank of an event participant within a Route and Route Group .	
Result	A gun or chip(net) finish time	

4. Authentication and Access

4.1. Enabling the Public API License

To access ACTIVE Results REST APIs, you must have a valid public API license. When your license expires, your access to the API will cease, even if you possess the API token.

To enable your API license, please contact your Account Manager.

4.2. Proxy Authentication Token

Before you make a RESTful API call, you must include the Proxy Authorization header with a valid access token with the following structure:

```
Proxy-Authorization: Token xxxxxxxxxxxxxxxxxxx
```

An API token is a string generated by ACTIVE Results, for example:

```
K/QYSpmum3unUUbKD9q7e4eZpTL2iRg7jQ3FF6F/wuONZi3dPGe0F+O+AiNp2C3h
```

The API token authenticates your organization's access to the API.

Note:

The API token is **CONFIDENTIAL** and should be distributed to authorized parties only.

- The API token is issued per account.
- To apply for an API token for your organization, please contact your Account Manager.

5. HTTP Status Codes

ACTIVE Results REST APIs use the following standard HTTP status codes:

Code	Name	Description
200	OK	The request was successful.
202	Accepted	The request has been accepted for processing, but the processing has not been completed. The request might or might not be eventually acted upon, and may be disallowed when processing occurs.
204	No data	The request was successfully processed, but no data was returned. E.g. invoking /agencies/{agencyID}/events with an agencyId that has no associated events
400	Bad Request	The server could not process the request due to a client error (e.g. malformed request syntax, invalid request message framing or deceptive request routing).
403	Forbidden	The server understood the request but refused to authorize it.
404	Not Found	The server did not find a current representation for the target resource or is unwilling to disclose that one exists.
405	Method Not Allowed	The method received in the request is known by the server, but not supported by the target resource.
407	Proxy Authentication Required.	This code is similar to 401(Unauthorized), but indicates that the client must first authenticate itself with the proxy.
500	Internal Server Error	The server encountered an unexpected condition that prevented it from fulfilling the request.
503	Service Unavailable	The server is currently unable to handle the request due to a temporary overload or scheduled maintenance.

6. Methods

ACTIVE Results APIs provide access to resources via URLs. To use a REST API, your application makes an HTTP request. The ACTIVE Results API uses JSON as its communication format and standard HTTP methods.

To import participants' results to ACTIVE Results, you must:

1. Create an event (NOT necessary for an existing AW Endurance event).
2. Create a route under the event which lists route locations (split or timing points).
3. Import participant results (final result time, division rankings and optionally split results) under a specific event and route.
4. Query the status of the import.

Additional methods are provided for updating an event's details or deleting a route within an event.

It is not possible to edit an individual participant's results after they have been uploaded for a specific route under a specific event (as this could change the rankings of other participants). The "Deleting a route" API can be used to delete a route and all participants' results under the route, then the route can be created again, and corrected results can be re-imported for all participants under the route.

6.1. Creating an event

Use this method to create an event into which to import participant results.

API path	https://awapi.active.com/rest/result-external-rest-api/events/		
HTTP method	POST		
Request body	<pre>{ "externalApplicationId" : "activeworksendurance", "externalId" : "20180515A", "timingSystem" : "ActiveTiming", "name" : "By gerry 20180514 A", "address" : { "country" : "US", "stateProvince" : "CA", "city" : "San diego" }, "startDate" : "2018-05-20T01:30:00Z", "visibility" : true }</pre>		
Request body parameter - sub-parameter	Format	Description	Required /optional
externalApplicationId	nvarchar(255) (activeworksendurance, activeworksswimming)	Name of the source application for the event (e.g. ACTIVEWorks)	Required

		Endurance or ACTIVEWorks Swimming; this parameter is case insensitive. If activeworksendurance is set for this parameter, then ACTIVEResults will automatically search and bind participants' results to their ACTIVEWorks Endurance registration).	
timingSystem	nvarchar(255)	The event's timing system (e.g. ActiveTiming)	Optional
externalId	nvarchar(255)	Event ID	Required
name	nvarchar(255)	Event name	Required
address	structure	Event address	Required
- country	nvarchar(255)	Event country	Required
- stateProvince	nvarchar(255)	Event state or province	Required
- city	nvarchar(255)	Event city	Required
startDate	nvarchar(255)	Event start date in ISO8601 time format (YYYY-MM-DDTHH:MM:SS) e.g. 2018-05-20T01:30:00Z	Required
visibility	boolean (true, false)	Event visibility on ACTIVE Results website	Optional (defaults to false)

<p>Response</p> <pre>{ "id": 10075, "name": "By gerry 20180514 A", "startDate": "2018-05-20T01:30:00Z", "address": { "country": "US", "stateProvince": "CA", "city": "San diego" }, "category": "", "applicationName": "Bygerry20180514A3", "externalId": "20180515A", "externalApplicationId": "ActiveworksEndurance", "timingSystem": "ActiveTiming", "visibility": true, "deleted": false, "enabledAD": false, "timeZone": "Asia/Shanghai", "logo": "NULL", "url": "https://results.active.com/events/Bygerry20180514A3", "routes": [] }</pre>
--

Parameter - sub-parameter	Format	Description
id	int	ID of the event in the results service
category	nvarchar(255)	Event sport category
applicationName	nvarchar(255)	Unique name for the event in the results service
deleted	boolean (true, false)	Event was deleted by a Timer
enabledAD	boolean (true, false)	Not in use
url	nvarchar(255)	URL of event home page on ACTIVE Results
routes	list	List of routes in this event (corresponds to data uploaded using the Creating a route API).

6.2. Updating an event

Use this method to edit the details of an event.

Notes:

- You may call this method to update event details even after you have created routes under the event and imported participants' results.
- Any empty parameters will be deleted. To preserve the value for a parameter, populate the parameter with the same value that was previously uploaded using the *Creating an event API*.

API path	https://awapi.active.com/rest/result-external-rest-api/events/{externalId}?externalApplicationId={externalApplicationId}		
HTTP method	PUT		
Request body	<pre>{ "name" : "gerry creates event 20180515", "address" : { "country" : "US", "stateProvince" : "CA", "city" : "San diego", }, "startDate" : "2018-05-20T01:30:00Z", "visibility" : true, "timingSystem": "ActiveTiming" }</pre>		
API Path parameter	Format	Description	Required /optional
externalId	nvarchar(255)	Event ID	Required
externalApplicationId	nvarchar(255) (activeworksendurance, activeworksswimming)	Name of the source application for the event (e.g. ACTIVEWorks Endurance or ACTIVEWorks Swimming; this parameter is	Required

		case insensitive). Must be the same as that set when calling the <i>Creating an event</i> API.	
Request body parameter - sub-parameter	Format	Description	Required /optional
name	nvarchar(255)	Event name	Required
address	structure	Event address	Optional
- country	nvarchar(255)	Event country	Optional
- stateProvince	nvarchar(255)	Event state or province	Optional
- city	nvarchar(255)	Event city	Optional
startDate	nvarchar(255)	Event start date in ISO8601 time format (YYYY-MM-DDTHH:MM:SS) e.g. 2018-05-20T01:30:00Z	Required
visibility	Boolean (true, false)	Event visibility on ACTIVE Results website	Optional
timingSystem	nvarchar(255)	The event's timing system (e.g. ActiveTiming)	Optional

Response		
<pre>{ "id": 10076, "name": "gerry creates event 20180515", "startDate": "2018-05-20T01:30:00Z", "address": { "country": "US", "stateProvince": "CA", "city": "San diego" }, "applicationName": "Bygerry20180514A4", "externalId": "20180515AB", "externalApplicationId": "ActiveworksEndurance", "visibility": true, "deleted": false, "enabledAD": false, "timeZone": "Asia/Shanghai", "logo": "abc", "url": "https://results.active.com/events/Bygerry20180514A3", "routes": [] }</pre>		
Parameter - sub-parameter	Format	Description
id	int	ID of the event in the results service
applicationName	nvarchar(255)	Unique name for the event in the results service
deleted	boolean (true, false)	Event was deleted by a Timer
enabledAD	boolean (true, false)	Not in use
url	nvarchar(255)	URL of event home page on ACTIVE Results

routes	list	List of routes in this event (corresponds to data uploaded using the Creating a route API).
--------	------	---

6.3. Creating a route

Use this method to create a route under an event.

Notes:

- A route consists of multiple route locations which correspond to split points or timing points.

API path	https://awapi.active.com/rest/result-external-rest-api/events/{eventExternalId}/routes?externalApplicationId={externalApplicationId}
HTTP method	POST
Request body	
<pre>{ "name" : "test route", "externalId" : "123456", "sportsType" : "RUNNING", "distanceName" : "10K", "useChipTime" : false, "distanceUnit" : "MI", "stages": [{ "stageName": "Swimming" "stageSequence": 1 "stageType" : "SWIMMING" "distance": 0.6 }, { "stageName" : "T1" "stageSequence" : 2 "stageType": "TRANSITION" "distance": }, { "stageName" : "Biking" "stageSequence" : 3 "stageType": "CYCLING" "distance": 14 }, { "stageName" : "T2" "stageSequence" : 4 }] }</pre>	

```

    "stageType": "TRANSITION"
    "distance":

  },

  {
    "stageName" : "Running"
    "stageSequence" : 5
    "stageType": "RUNNING"
    "distance": 8
  }

],

"routeLocations" : [
  {"name" : "location 1", "stageSequence": "1", "sequence" : 1}
  {"name" : "location 2", "stageSequence": "1", "sequence" : 2}
  {"name" : "location 3", "stageSequence": "2", "sequence" : 3}
  {"name" : "location 4", "stageSequence": "3", "sequence" : 4}
  {"name" : "location 5", "stageSequence": "3", "sequence" : 5}
  {"name" : "location 6", "stageSequence": "4", "sequence" : 6}
  {"name" : "location 7", "stageSequence": "5", "sequence" : 7}
  {"name" : "location 8", "stageSequence": "5", "sequence" : 8}
  {"name" : "location 9", "stageSequence": "5", "sequence" : 9}
  {"name" : "location 10", "stageSequence": "5", "sequence" : 10}
]
}

```

API Path parameter	Format	Description	Required /optional
eventExternalId	nvarchar(255)	Event ID	Required
externalApplicationId	nvarchar(255) (activeworksendurance, activeworksswimming)	Name of the source application for the event (e.g. ACTIVEWorks Endurance or ACTIVEWorks Swimming; this parameter is case insensitive). Must be the same as that set when calling the <i>Creating an event</i> API.	Required

Request body parameter - sub-parameter	Format	Description	Required /optional
name	nvarchar(255)	Route name	Required
externalId	nvarchar(255)	Route ID	Required
sportsType	nvarchar(255)	Sport type e.g. RUNNING, TRIATHLON...	Required
distanceName	nvarchar(255)	Distance name e.g. 10K, Sprint...	Required
useChipTime	boolean (true, false)	Route is using chip times	Optional (defaults to false)

distanceUnit	nvarchar(255) (KM, MI or MILE)	Distance unit	Required
stages	list	List of route stages (for multistage races such as triathlons)	Optional
-stageName	nvarchar(255)	Stage name	Required if stages list is uploaded
-stageSequence	1-65536	Stage sequence number	Required if stages list is uploaded
-stageType	nvarchar(255) (SWIMMING, CYCLING, RUNNING, TRANSITION)	Stage sport type	Required if stages list is uploaded
-distance	double	Stage distance, in units as specified by the distanceUnit field	optional
routeLocations	list	List of route locations (this is optional, but if used, then at least one location {- name, - sequence:1} corresponding to the start point must be submitted.	Optional
- name	nvarchar(255)	Location name	Required
- sequence	int	Location sequence number Note: must be a positive, non-zero integer.	Required

Response
None

6.4. Deleting a route

Use this method to delete a route under an event.

Notes:

- If participants' results have been uploaded under a routeExternalId, then calling this method for the routeExternalId can be used to delete all participants results under the routeExternalId.
- If you need to edit a participant's results, then it is necessary to delete all participants in a routeExternalId by calling this method, then re-creating the route, and finally re-uploading all participant's results again under the new route. It is **NOT** possible to edit an individual participant's results as this may change the ranking of other participants.

API path	https://awapi.active.com/rest/result-external-rest-api/events/{eventExternalId}/routes/{routeExternalId}?externalApplicationId={externalApplicationId}	
HTTP method	DELETE	
Request body	None	
API Path parameter	Format	Description
eventExternalId	nvarchar(255)	Event ID
routeExternalId	nvarchar(255)	Route ID
externalApplicationId	nvarchar(255) (activeworksendurance, activeworksswimming)	Name of the source application for the event (e.g. ACTIVEWorks Endurance or ACTIVEWorks Swimming; this parameter is case insensitive). Must be the same as that set when calling the <i>Creating an event</i> API.

Response
None

6.5. Querying event routes

Use this method to retrieve all routes under an event, including route ID numbers.

Notes:

- A route consists of multiple route locations which correspond to split points or timing points.
- The response structure is **NOT** the same as the **Creating a route** API request body structure.

API path	https://awapi.active.com/rest/result-external-rest-api/events/{eventExternalId}/routes?externalApplicationId={externalApplicationId}	
HTTP method	GET	
Request body		
None		
API Path parameter	Format	Description
eventExternalId	nvarchar(255)	Event ID
externalApplicationId	nvarchar(255) (activeworksendurance, activeworksswimming)	Name of the source application for the event (e.g. ACTIVEWorks Endurance or ACTIVEWorks Swimming; this parameter is case insensitive). Must be the same as that set when calling the <i>Creating an event</i> API.

Response
<pre>[{ "id": 100556, "name": "test route", "distanceName": "Olipc", "sportsType": "TriATHLON", "useChipTime": false, "externalId": "testRoute554433", "stages": [{ "id": 964, "sequence": 1, "name": "running", "type": "RUNNING", "distanceUnit": "MI", "distance": 0.6, "routeLocations": [{ "id": 103360, "name": "location 1", "description": "location 1", "sequence": 1 }, { </pre>

```

        "id": 103361,
        "name": "location 2",
        "description": "location 2",
        "sequence": 2
      }
    ]
  },
  {
    "id": 962,
    "sequence": 2,
    "name": "T1",
    "type": "TRANSITION",
    "distanceUnit": "MI",
    "distance": 0,
    "routeLocations": [
      {
        "id": 103362,
        "name": "location 3",
        "description": "location 3",
        "sequence": 3
      }
    ]
  },
  {
    "id": 966,
    "sequence": 3,
    "name": "Biking",
    "type": "CYCLING",
    "distanceUnit": "MI",
    "distance": 14,
    "routeLocations": [
      {
        "id": 103363,
        "name": "location 4",
        "description": "location 4",
        "sequence": 4
      },
      {
        "id": 103364,
        "name": "location 5",
        "description": "location 5",
        "sequence": 5
      }
    ]
  },
  {
    "id": 963,
    "sequence": 4,
    "name": "T2",
    "type": "TRANSITION",
    "distanceUnit": "MI",
    "distance": 0,
    "routeLocations": [
      {
        "id": 103365,
        "name": "location 6",
        "description": "location 6",

```

```

        "sequence": 6
      }
    ]
  },
  {
    "id": 965,
    "sequence": 5,
    "name": "Running",
    "type": "RUNNING",
    "distanceUnit": "MI",
    "distance": 8,
    "routeLocations": [
      {
        "id": 103366,
        "name": "location 7",
        "description": "location 7",
        "sequence": 7
      },
      {
        "id": 103367,
        "name": "location 8",
        "description": "location 8",
        "sequence": 8
      },
      {
        "id": 103368,
        "name": "location 9",
        "description": "location 9",
        "sequence": 9
      },
      {
        "id": 103369,
        "name": "location 10",
        "description": "location 10",
        "sequence": 10
      }
    ]
  }
]

```

Parameter - sub-parameter	Format	Description
id	int	Route ID number
name	nvarchar(255)	Route name
distanceName	nvarchar(255)	Distance name e.g. 10K, Sprint...
sportsType	nvarchar(255)	Sport type e.g. RUNNING, TRIATHLON...
useChipTime	boolean (true, false)	Route is using chip times
externalId	nvarchar(255)	External route ID
stages	list	List of route stages (for multistage races such as triathlons)
-id	int	Stage ID number

-sequence	1-65536	Stage sequence number
-name	nvarchar(255)	Stage name
-type	nvarchar(255) (SWIMMING, CYCLING, RUNNING, TRANSITION)	Stage sport type
-distanceUnit	nvarchar(255) (KM, MI or MILE)	Distance unit
-distance	double	Stage distance, in units as specified by the distanceUnit field
-routeLocations	list	List of route locations (this is optional, but if used, then at least one location {– name, - sequence:1} corresponding to the start point must be submitted.
--id	int	Location ID number
--name	nvarchar(255)	Location name
--description		Location description
--sequence	int	Location sequence number Note: must be a positive, non-zero integer.

6.6. Importing a participant's results

Use this method to upload a single participant's results for a specified route in a specified event.

Notes:

- A single participant's results must include their final result time and division rankings (at least one division ranking is required).
- The participant's split results are optional, but if they are included, then they must be in the same sequence as previously uploaded locations in the Creating a route > `routeLocations` list.

API path	<code>https://awapi.active.com/rest/result-external-rest-api/events/{eventExternalId}/routes/{routeExternalId}/participants?externalApplicationId={externalApplicationId}</code>
HTTP method	POST
Request body	<pre>[{ "registrationId" : "gerry2", "bib" : "gerry2", "firstName" : "Patric20180509A", "lastName" : "Patric20180509A", "penaltyTime": "PT23M1S", "age" : 26, "gender" : "MALE", "address" : { "country" : "China", "stateProvince" : "sc", "city" : "cd" } "chipTimeResult" : "PT2H22M59S", "gunTimeResult" : "PT2H23M1S", "finalResult" : "PT2H23M1S", "finishTime" : "2018-05-21T00:00:00Z", "gunTime" : "2018-05-21T00:00:00Z", "team": "abc", "averageSpeed" : 17.73, "disqualified": false, "divisionRanks" : [{ "name" : "overall", "rank" : 2 }, { "name" : "Male", "rank" : 2 }, { "name" : "M25-35", "rank" : 2 }, { "name" : "overall", "rank" : 2, "stageSequence": 1}, { "name" : "Male", "rank" : 2, "stageSequence": 3}, { "name" : "M25-35", "rank" : 2, "stageSequence": 5}], "stageResults": [{ "stageSequence" : 1, "finalResult" : "PT2H22M59S", "displayPace" : "01:46 /HM" }, { "stageSequence" : 2, "finalResult" : "PT2H22M59S",</pre>

```

    "displayPace" : ""
  },
  {
    "stageSequence" : 3,
    "finalResult" : "PT2H22M59S",
    "displayPace" : "22:00 MPH"
  },
  {
    "stageSequence" : 4,
    "finalResult" : "PT2H22M59S",
    "displayPace" :
  },
  {
    "stageSequence" : 5,
    "finalResult" : "PT2H22M59S",
    "displayPace" : "06:22/mi"
  },
]

"splitResults" : [
  { "stageSequence": "1", "sequence":1},
  { "stageSequence": "1", "sequence":2, "splitTime":"PT32M48S",
"averageSpeed":17.88 },
  { "stageSequence": "2", "sequence":3, "splitTime":"PT39M48S",
"averageSpeed": },
  { "stageSequence": "3", "sequence":4, "splitTime":"PT39M48S",
"averageSpeed":27.99 },
  { "stageSequence": "3", "sequence":5, "splitTime":"PT39M48S",
"averageSpeed":37.99 },
  { "stageSequence": "4", "sequence":6, "splitTime":"PT39M48S",
"averageSpeed": },
  { "stageSequence": "5", "sequence":7, "splitTime":"PT39M48S",
"averageSpeed":57.99 },
  { "stageSequence": "5", "sequence":8, "splitTime":"PT39M48S",
"averageSpeed":67.99 },
  { "stageSequence": "5", "sequence":9, "splitTime":"PT39M48S",
"averageSpeed":77.99 },
  { "stageSequence": "5", "sequence":10, "splitTime":"PT39M48S",
"averageSpeed":87.99 }
]
}]

```

API Path parameter	Format	Description	Required /optional
eventExternalId	nvarchar(255)	Event ID	Required
routeExternalId	nvarchar(255)	Route ID	Required
externalApplicationId	nvarchar(255) (activeworksendurance, activeworksswimming)	Name of the source application for the event (e.g. ACTIVEWorks Endurance or ACTIVEWorks Swimming; this parameter is case	Required

		insensitive). Must be the same as that set when calling the <i>Creating an event API</i> .	
Request body parameter - sub-parameter	Format	Description	Required /optional
registrationId	nvarchar(255)	Participant's registration ID	Optional
bib	nvarchar(255)	Participant's bib number	Required
firstName	nvarchar(255)	Participant's first name	Required
lastName	nvarchar(255)	Participant's last name	Required
penaltyTime	PT23M1S	Participant's penalty time	Optional
age	int (1-65536)	Participant's age	Optional
gender	nvarchar(255) ("MALE", "FEMALE")	Participant's gender	Optional
address	structure	Participant's address	Optional
- country	nvarchar(255)	Participant's country	Optional
- stateProvince	nvarchar(255)	Participant's state/province	Optional
- city	nvarchar(255)	Participant's city	Optional
chipTimeResult	PT2H61M1S	Participant's chip time result	Optional
gunTimeResult	PT2H61M1S	Participant's gun time result	Optional
finalResult	PT2H61M1S	Participant's final result	Required
finishTime	2018-05-21T23:59:59Z	Participant's finish time in ISO8601 time format (YYYY-MM-DDTHH:MM:SS)	Optional
gunTime	2018-05-21T23:59:59Z	Gun time in ISO8601 time format (YYYY-MM-DDTHH:MM:SS)	Optional
team	varchar(255)	Participant's team	Optional
averageSpeed	decimal	Participant's average speed in KM/H	Optional
disqualified	Boolean (true, false)	Participant's disqualification status	Optional, defaults to false
divisionRanks	list	List of participant's division rankings. Note: at least one division rank must be listed.	Required
- name	nvarchar(255)	Ranking name	Required
- rank	int	Rank Note: must be a non-zero, positive number.	Required

- stageSequence	1-65536	Stage sequence number. Including this parameter and its value indicates a stage ranking. Omitting this parameter and its value indicates an overall ranking.	Optional
stageResults	list	List of the participant's stage results	Optional
- stageSequence	1-65536	Stage sequence number, corresponds to the stageSequence list in the Creating a route API	Required if stageResults list is uploaded
-finalResults	PT2H61M1S	Stage final result	Optional
-displayPace	nvarchar(255)	Stage pace value and unit	Optional
splitResults	list	List of participant's split results. Note: split results must be in the same sequence as previously uploaded locations in the Creating a route > routeLocations list. The first location ({"sequence":1}) corresponds to the start and does NOT require a splitTime or averageSpeed.	Optional
-stageSequence	1-65536	Stage sequence number	Required for multistage sports. (e.g. triathlons or duathlons)
- sequence	int	Split sequence number	Required
- splitTime	PT2H61M1S	Cumulative split time	Optional
- averageSpeed	decimal	Average speed in KM/H	Optional

Response		
<pre>{ "uploadHistoryId": 170, "status": "QUEUED" }</pre>		
Parameter - sub-parameter	Format	Description
uploadHistoryId	int	ID of this import
status	nvarchar(255)	Status of this import

6.7. Querying a result's import status

Use this method to query the status of the import of a single participant's result for a route in an event.

Notes:

- Query the status of a participant's results upload using the corresponding uploadHistoryId value returned by the import.

API path	https://awapi.active.com/rest/result-external-rest-api/events/{eventExternalId}/histories/{uploadHistoryId}/status?externalApplicationId={externalApplicationId}	
HTTP method	GET	
Request body		
None		
API Path parameter	Format	Description
eventExternalId	nvarchar(255)	Event ID
uploadHistoryId	nvarchar(255)	The uploadHistoryId in the response from importing a participant's results
externalApplicationId	nvarchar(255) (activeworksendurance, activeworksswimming)	Name of the source application for the event (e.g. ACTIVEWorks Endurance or ACTIVEWorks Swimming; this parameter is case insensitive). Must be the same as that set when calling the <i>Creating an event</i> API.

Response		
<pre>{ "status": "PROCESSED" }</pre>		
Parameter - sub-parameter	Format	Description
status	nvarchar(255)	Status of this import e.g. PROCESSED, PROCESSING, IMPORT_FAILED, QUEUED, DELETING, DELETED, TIME_OUT