



# **Technical Description how to use IOF XML 3.0 in communication with IOF Eventor**

**Issued by the IOF IT commission**

# Contents

- 1 Introduction ..... 3
  - 1.1 IOF Interface Standard 3.0 ..... 3
  - 1.2 IOF Eventor ..... 3
- 2 The Person element ..... 4
- 3 The Organisation element ..... 6
- 4 Other elements ..... 6
  - 4.1 Racenumber ..... 6
  - 4.2 Snapshot and Delta ..... 6
- 5 Formatting dates and times in different time zones ..... 7
- 6 Appendix ..... 8
  - 6.1 Example of start list ..... 8
  - 6.2 Example of result list ..... 9

# 1 Introduction

This document describes for developers of sport management softwares how IOF Eventor interprets files in the IOF XML 3.0 format. The purpose of this document is to specify import of startlists and result lists in to IOF Eventor.

## 1.1 IOF Interface Standard 3.0

The International Orienteering Federation (IOF) Interface Standard is a specification of an XML data format for the interchange of orienteering-related information (events, entries, start lists, result lists, etc.) between applications, web services and databases. When adopted by orienteering software packages, it will allow the packages to exchange information between each other. This will benefit event organisers and contribute to simplifying the organisation of orienteering events.

The formal specification of the new interface standard can be found here:  
<http://www.orienteering.org/datastandard/IOF.xsd>

## 1.2 IOF Eventor

IOF Eventor is Intenational Orienteering Federation's Event Management Service managing events, entries and results.

## 2 The Person element

Represents a person, who could either be a competitor or contact person in an organisation.

Matching a person in the file which is being uploaded with the right person in the database is a crucial part of importing start list and result list.

For finding the matching person the following elements are considered, both are mandatory in the data exchange:

- Primary: Id element
- Secondary: Name element

It is important that the Id element is the same in the file and in the IOF Eventor database. A person element can have multiple Id elements, which makes it possible to include both national and international IDs. To distinguish the different IDs the type attribute is used, which is "IOF" in the IOF Eventor:

```
<Person>
<Id type="IOF">12345</Id>
<Id type="SOFT">ABC123</Id>
<Name>
  <Family>Wood</Family>
  <Given>George</Given>
</Name>
</Person>
```

The different Id elements may follow each other in any order, it doesn't affect the data transfer.

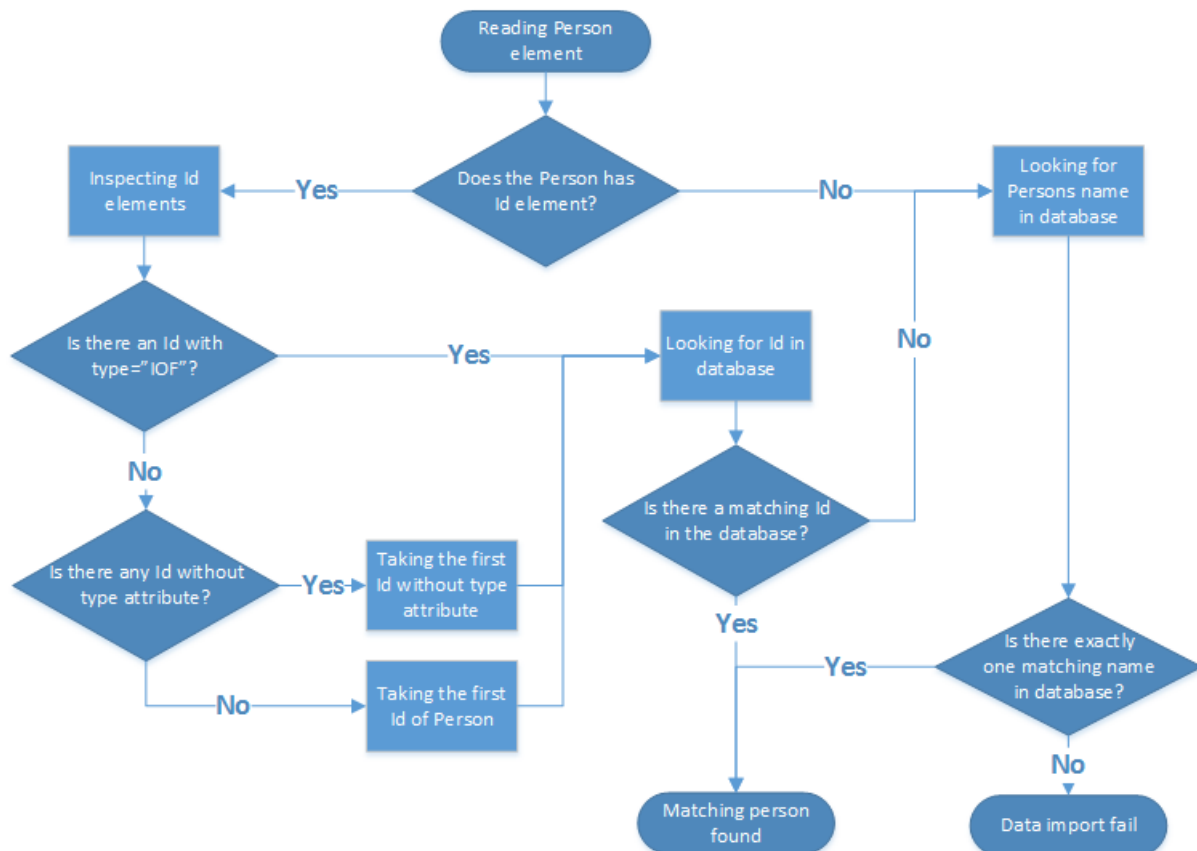


Figure 1. Flow diagram of finding a matching Person

If the matching person is found by the Id element and the name of the person differs between the file and the database, the name given in the file is given precedence. However, the name of the person in the database will not be updated, the name given in the file will only be used in the start-/result list.

If the Name element given in the file is empty, the name in the database will be used in the start-/result list.

IOF recommends that in the file only the IOF ID is given for identifying a person and the Name element should be empty. It will make sure that the same name will appear consistently in start-/result lists for one person.

```
<Person>  
<Id type="IOF">18340</Id>  
<Name>  
    <Family/>  
    <Given/>  
</Name>  
</Person>
```

## 3 The Organisation element

Represents an organisation, which is a general term including federations, clubs, etc. The element contains information i.e. address, contact person, etc.

Similar to persons, the organisations in Eventor have their unique IDs. However, only one ID per organisation is allowed.

The following algorithm is used for finding a matching organization in the database when importing data from file:

1. If there is an Id element present, Eventor uses the organisation in the database with this ID, if found.
2. If there is no Id element present, the organisation's Name element is used to find a matching organisation in the database. If there is exactly one matching organisation found, the ID of this organisation is used.
3. If no any or more than one matching organization found in the database, the name from the file will be presented in the start-/result list.

If a matching organisation is found, the name in the uploaded start-/result list will be taken from the database. This behavior is different from how person names are handled.

## 4 Other elements

### 4.1 Racenumber

IOF organises single-race events (WRE) even if the current race is one day of a multi-race event. Using the RaceNumber element and attribute can cause a problem in data exchange with IOF Eventor.

IOF strongly recommends **not using the RaceNumber element and attribute** in any case while importing start-/result lists.

### 4.2 Snapshot and Delta

IOF XML data standard makes it possible to exchange result lists in three different ways:

- Complete
- Snapshot
- Delta

The first one, Complete, includes all competitors and should be used after the event for publishing final results. This is the default behavior.

The latter two types are intended to be used for publishing “real time” results during the event, which is very convenient for live result applications.

However, IOF Eventor is meant to publish final results only and it can not handle the type Snapshot and Delta.

For this reason IOF strongly recommends **using only the status=”complete” attribute in the ResultList element** while importing files to IOF Eventor.

## 5 Formatting dates and times in different time zones

Dates and times are expressed in the [ISO 8601 format](#) yyyy-mm-ddThh:mm:ssZ. This format allows inclusion of a time zone identifier. Eventor handles times both with and without time zone information. However, for clarity, the recommendation to software vendors is to always include time zone information when generating IOF XML 3.0 files.

Most programming languages have a means to determine the time zone used on the computer, and the time offset to UTC (Coordinated Universal Time).

For example, the start time of a competitor starting at 12:00 in an event in Sweden organized the 1st of June 2015 can be set to either of these values, which all refer to the same instant of time:

**<StartTime>2015-06-01T12:00:00</StartTime>**

No time zone given: IOF Eventor assumes the local time of the event. \*)

**<StartTime>2015-06-01T12:00:00+02:00</StartTime>**

Time zone given: Central European Time (CET) with daylight saving.

**<StartTime>2015-06-01T10:00:00Z</StartTime>**

Time zone given: Coordinated Universal Time (UTC). 'Z' denotes UTC.

Observe, UTC never changes for Daylight Saving Time, so this time is not correct:

**<StartTime>2015-06-01T12:00:00+01:00</StartTime>**

Time zone given: CET without daylight saving, which is invalid for June.

IOF Eventor uses strictly the time zone information given in the uploaded files, which means that 2015-06-01T12:00:00+01:00 is be interpreted as 2015-06-01T11:00:00Z (UTC) and 2015-06-01T13:00:00+02:00 (CET with daylight saving).

*\*) After January 1<sup>st</sup> 2016. Until December 31<sup>st</sup> 2015 IOF Eventor assumes times without any time zone information to be in CET.*

## 6 Appendix

### 6.1 Example of start list

```
<?xml version="1.0"?>
<StartList xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
iofVersion="3.0"
xmlns="http://www.orienteering.org/datastandard/3.0"
xsi:schemaLocation="http://www.orienteering.org/datastandard/
3.0 http://www.orienteering.org/datastandard/iof.xsd">
  <Event>
    <Name>IOC Day 1 Sprint</Name>
    <StartTime>
      <Date>2015-05-01</Date>
    </StartTime>
  </Event>
  <ClassStart>
    <Class>
      <Name>M21E</Name>
    </Class>
    <PersonStart>
      <Person sex="M">
        <Id type="IOF">19219</Id>
        <Name>
          <Family>Pastrana</Family>
          <Given>Mario Nogales</Given>
        </Name>
        <BirthDate>1994-01-01</BirthDate>
      </Person>
      <Organisation>
        <Name>QUBOC</Name>
      </Organisation>
      <Start>
        <BibNumber>134</BibNumber>
        <StartTime>2015-05-01T18:12:00</StartTime>
      </Start>
    </PersonStart>
    <PersonStart>
      <Person sex="M">
        <Id type="IOF">13654</Id>
        <Name>
          <Family>Millar</Family>
          <Given>Ivan</Given>
        </Name>
        <BirthDate>1975-01-01</BirthDate>
      </Person>
      <Organisation>
        <Name>LVO</Name>
      </Organisation>
      <Start>
        <BibNumber>132</BibNumber>
        <StartTime>2015-05-01T18:14:00</StartTime>
      </Start>
    </PersonStart>
    .
    .
  </ClassStart>
</StartList>
```



## 6.2 Example of result list

```
<?xml version="1.0" encoding="utf-8"?>
<ResultList xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
iofVersion="3.0" createTime="2015-05-04T11:21:33.5287906"
creator="Eventor"
xmlns=http://www.orienteering.org/datastandard/3.0
status="Complete">
  <Event>
    <Id>4906</Id>
    <Name>Belgium 3 days & Coupe des Pays Latins</Name>
    <StartTime>
      <Date>2015-05-02</Date>
      <Time>00:00:00.0000000+02:00</Time>
    </StartTime>
  </Event>
  <ClassResult>
    <Class>
      <Name>M21E</Name>
      <ShortName>M21E</ShortName>
    </Class>
    <Course>
      <Length>6480</Length>
    </Course>
    <PersonResult>
      <Person>
        <Id type="IOF">10586</Id>
        <Name>
          <Family />
          <Given />
        </Name>
      </Person>
      <Organisation>
        <Id>166</Id>
        <Name>BEL</Name>
      </Organisation>
      <Result>
        <StartTime>2015-05-02T00:00:00</StartTime>
        <FinishTime>2015-05-02T00:33:04</FinishTime>
        <Time>1984</Time>
        <TimeBehind>0</TimeBehind>
        <Position>1</Position>
        <Status>OK</Status>
      </Result>
    </PersonResult>
    .
    .
  </ClassResult>
</ResultList>
```