Event Feed version 2 draft

From ICPC-Contest Control Standard

The *Event Feed* is an externally-accessible XML feed of contest data supplied by a [Contest Control System](). The event feed is the primary way that 'external' clients can get the state of a running contest. Typical clients include:

- ICPC Live, video coverage of the event.
- Contest-floor scoreboards.
- The "resolver", a graphical program showing what happened during the contest freeze.
- A balloon printer.
- Supporting 'shadow' CCSs, e.g. running a second non-active CCS for verification purposes.
- Post-contest testing by replaying contests.

☐

Contents

Accessing the Event Feed

Clients make use of the event feed by connecting to the event feed port provided by the CCS (event-feed-port in [contest.yaml]()). All clients connect to this same port.

Clients must use SSL to connect; any client connection that doesn't use SSL is dropped.

Information Updates

CCSs will typically send all configuration elements, followed by a set of events. The time of each new event must be in increasing contest and wall-clock time so that clients know they have the complete state up to the time of the most recently received element.

The event feed must be consistent at any given time; elements that refer to other elements must be sent after the element they refer to. For instance `<run>` must be sent before the corresponding `<run-source>` or `<run-judgement>`.

Changes may cause any configuration or event element to be resent. These changes include but are not limited to:

- Changing the state of a problem
- Rejudgements of runs
- Fixing typos in names

Removal (or reinsertion) of a time interval will cause the connection to be dropped.

Different clients (or reconnecting clients) must receive a semantically identical event feed, but are not guaranteed to receive a syntactically identical event feed. For instance, if a typo was fixed in a problem name that causes the element to be reset, new clients might simply receive a single configuration element with the fixed name.

File Format

The event feed is an XML document within the root element `<contest>`. The root contest element is as follows:

Attribute	Description
version	The format version. Must be "1.0"

Example:

```
<contest version="1.0">
  <info ... />
  <language ... />
  ...
</contest>
```

Within the contest element are a number of configuration items and events, each of which are separate top-level elements:

Event types & tags:

Element	Contains	Type
info	contest information and updates	Configuration
language	submission language information	Configuration
group	group information	Configuration
judgement	judgement information	Configuration
problem	problem information	Configuration

team	team information	Configuration
clar-request	submitted clarification	Event
clar-response	clarification response	Event
run	submitted run and updates	Event
run-judgement	judgement of a run and updates	Event
run-source	source code submitted with a run and updates	Event
test-result	judgement of individual input file for a run	Event
team-penalty	adjusted penalty time for a team	Event
finalized	the contest standings/medals are assigned, contest is over	Event
reset	reset notification	Event

Notes

1. All child-elements and attributes are required unless otherwise documented.
2. All contest-time elements are in minutes.
3. All timestamp elements are in seconds since the Unix epoch, Jan 1st, 1970 00:00:00 UTC.

Configuration Elements

Info Element

Format:

Name	Description	Child Element/Attribute	Type
title	Contest title	element	xsd:string
length	Length of contest in HH:MM:SS format	element	xsd:string
penalty-time	Penalty time in minutes	element	xsd:int
started	True if the contest is started	element	xsd:boolean
start-time	Wall-clock start time of the contest	element	xsd:decimal

Example:

```
<info>
  <title>The 2010 World Finals of the ACM International Collegiate
Programming Contest</title>
  <length>05:00:00</length>
  <penalty-time>20</penalty-time>
  <started>false</started>
  <start-time>1265335138.26</start-time>
</info>
```

Language Element

Format:

Name	Description	Child Element/Attribute	Type
name	Language name	element	xsd:string

Example:

```
<language>
  <name>C++</name>
</language>
```

Group Element

Format:

Name	Description	Child Element/Attribute	Type
external-id	Identifier (integer) from [groups.tsv (Group ID)](groups.tsv)	attribute	xsd:int
name	Group name from groups.tsv	element	xsd:string

Example:

```
<group external-id="3012">
  <name>Europe</name>
</group>
```

Judgement Element

Format:

Name	Description	Child element/attribute	Type
id	Judgement ID	attribute	xsd:int
acronym	Judgement acronym	element	xsd:string
name	Judgement name	element	xsd:string
penalty	True if this judgement is a failure that causes penalty time	element	xsd:boolean

Example:

```
<judgement id="3">
  <acronym>WA</acronym>
  <name>Wrong Answer</name>
  <penalty>true</penalty>
</judgement>
```

Problem Element

Format:

Name	Description	Child Element/Attribute	Type
id	Problem ID	attribute	xsd:int
letter	Single letter problem label	element	xsd:string
name	Problem name	element	xsd:string
color	Color name	element	xsd:string
rgb	Color RGB value	attribute of color element	xsd:string

Example:

```
<problem id="1">
```

```
  <letter>A</letter>
  <name>APL Lives!</name>
  <color rgb="#ffff00">yellow</color>
</problem>
```

Team Element

Format:

Name	Description	Child Element/Attribute	Type
id	Team ID from teams.tsv (Team Number)	attribute	xsd:int
external-id	External team id from [[Contest_Control_System#teams.tsv|teams.tsv (External ID)	attribute	xsd:int
name	Team name from teams.tsv	element	xsd:string
country	Country code, ISO 3166-1 alpha-3, from teams.tsv	element	xsd:string
institution	Institution name from teams.tsv	element	xsd:string
group	Group name groups.tsv from teams.tsv	element	xsd:string

Example:

```
<team id="1" external-id="23412">
  <name>The Winners</name>
  <country>LBN</country>
  <institution>American University of Beirut</institution>
  <group>Europe</group>
</team>
```

Event Elements

Clar Request Event

The clar-request event is sent when a team submits a clarification request.

Format:

Name	Description	Child Element/Attribute	Type
id	Clarification Request ID	attribute	xsd:int
team-id	Team ID	attribute	xsd:int
question	A question, typically from a team	element	xsd:string
category	Clarification category	element	xsd:string
contest-time	Contest (elapsed) time when the clarification was submitted	element	xsd:decimal
timestamp	Wall-clock time when the clarification was submitted	element	xsd:decimal

Example:

```
<clar-request id="1" team-id="0">
  <question>What is the upper limit on the number of pieces of chocolate
```

```
requested by the friends?</question>
  <category>Problem C</category>
  <contest-time>118.48</contest-time>
  <timestamp>1265335256.74</timestamp>
</clar-request>
```

Clar Response Event

The clar response event is sent when a clar request has been answered.

Format:

Name	Description	Child Element/Attribute	Type
clar-request-id	Clarification Request ID	attribute	xsd:int
answer	Judge's answer to the question	element	xsd:string
to-all	True if the response was sent to all teams	element	xsd:boolean
contest-time	Contest (elapsed) time when the response was submitted	element	xsd:decimal
timestamp	Wall-clock time when the response was submitted	element	xsd:decimal

Example:

```
<clar-response clar-request-id="1">
  <answer>The number of pieces will fit in a signed 32-bit integer.</answer>
  <to-all>true</to-all>
  <contest-time>118.48</contest-time>
  <timestamp>1265335256.74</timestamp>
</clar-response>
```

Run Event

A run event is sent whenever a team submits a run.

Format:

Name	Description	Child Element/Attribute	Type
id	Run ID	attribute	xsd:int
team-id	Team ID	attribute	xsd:int
problem-id	Problem ID	attribute	xsd:int
language	Language name	element	xsd:string
contest-time	Contest (elapsed) time when the run was submitted	element	xsd:decimal
timestamp	Wall-clock time when the run was submitted	element	xsd:decimal

Example:

```
<run id="1410" team-id="74" problem-id="4">
  <language>C++</language>
  <contest-time>17960.74</contest-time>
  <timestamp>1265353100.29</timestamp>
```

```
</run>
```

Run Judgement Event

A run judgement event is sent whenever a judgement has been assigned to a run.

Format:

Name	Description	Child Element/Attribute	Type
run-id	Run ID	attribute	xsd:int
judgement-id	Judgement ID	attribute	xsd:int
judgement	Judgement acronym	element	xsd:string
contest-time	Contest (elapsed) time when the judgement was assigned	element	xsd:decimal
timestamp	Wall-clock time when the judgement was assigned	element	xsd:decimal

Example:

```
<run-judgement run-id="1410" judgement-id="3">
  <judgement>WA</judgement>
  <contest-time>179.74</contest-time>
  <timestamp>1265353100.29</timestamp>
</run-judgement>
```

Run Source Event

A run source event is sent along with every submitted run event.

Format:

Name	Description	Child Element/Attribute	Type
run-id	Run ID	element	xsd:int
file	Child element for each file	element	n/a

File element format:

Name	Description	Child Element/Attribute	Type
name	File name	attribute	xsd:string
main	True if the file is the primary file (optional, typically only marked when true)	attribute	xsd:boolean
(element content)	Base 64 encoded content of the file	element content	xsd:base64Binary

Example:

```
<run-source run-id="1410">
  <file name="hello.st" main="true">
    J0hlbGxvIFdvcmxkLicu
  </file>
```

```
  <file name="hello2.st">
    J0Fub3RoZXIgZmlsZSScu
  </file>
</run-source>
```

Test Result Event

A test result event is sent when a run is judged against an input file.

Format:

Name	Description	Child Element/Attribute	Type
run-id	Run ID	attribute	xsd:int
num-test	Input file index (1-based)	attribute	xsd:int
total-tests	Total number of input files	attribute	xsd:int
judgement-id	Judgement ID	attribute	xsd:int
judgement	Judgement acronym	element	xsd:string
contest-time	Contest (elapsed) time when the test was judged	element	xsd:decimal
timestamp	Wall-clock time when the test was judged	element	xsd:decimal

Example:

```
<test-result run-id="1" num-test="2" total-tests="5" judgement-id="3">
  <judgement>WA</judgement>
  <contest-time>939.75</contest-time>
  <timestamp>1265336078.01</timestamp>
</test-result>
```

Team Penalty Event

A team penalty event is sent when an administrator adds or removes penalty time for a team.

Format:

Name	Description	Child Element/Attribute	Type
team-id	Team ID	attribute	xsd:int
time	The number of added penalty minutes, may be negative	element	xsd:int
contest-time	Contest (elapsed) time when the penalty was added	element	xsd:decimal
timestamp	Wall-clock time when the penalty was added	element	xsd:decimal

Example:

```
<team-penalty team-id="74">
  15
  <contest-time>17960.74</contest-time>
  <timestamp>1265353100.29</timestamp>
</team-penalty>
```

Finalized Event

The finalized event is sent when the contest is finalized. It must be the last event and will be followed immediately by the `</contest>` tag.

Format:

Name	Description	Child Element/Attribute	Type
last-gold	The last place to receive a gold	element	xsd:int
last-silver	The last place to receive a silver	element	xsd:int
last-bronze	The last place to receive a bronze	element	xsd:int
comment	A comment provided during the finalization	element	xsd:string
timestamp	Wall-clock time when the contest was finalized	element	xsd:decimal

Example:

```
<finalized>
  <last-gold>4</last-gold>
  <last-silver>8</last-silver>
  <last-bronze>12</last-bronze>
  <comment>Finalized by John Doe and Jane Doe</comment>
  <timestamp>1265336078.01</timestamp>
</finalized>
```

Navigation menu

Page actions

- Page
- Discussion
- View source
- History
- PDF Export

Page actions

- Page
- Discussion
- More
- Tools

Personal tools

- Log in

Search

[   ] Go   Search