Contest API

An ICPC standard for contest systems & tools

Fredrik Niemelä & Jaap Eldering

17 April 2018 – CLI Symposium

Contributors

The creation of this API was a collaborative effort over the last year of many groups/people involved with the World Finals systems:

- ICPC Tools (CDS, scoreboard resolver, ...) – Tim deBoer
- Kattis – Fredrik Niemelä
- PC^2 – John Clevenger
- DOMjudge – Tobias Werth, Jaap Eldering

We have also received valuable feedback from other people during the first implementation tests.

A brief history

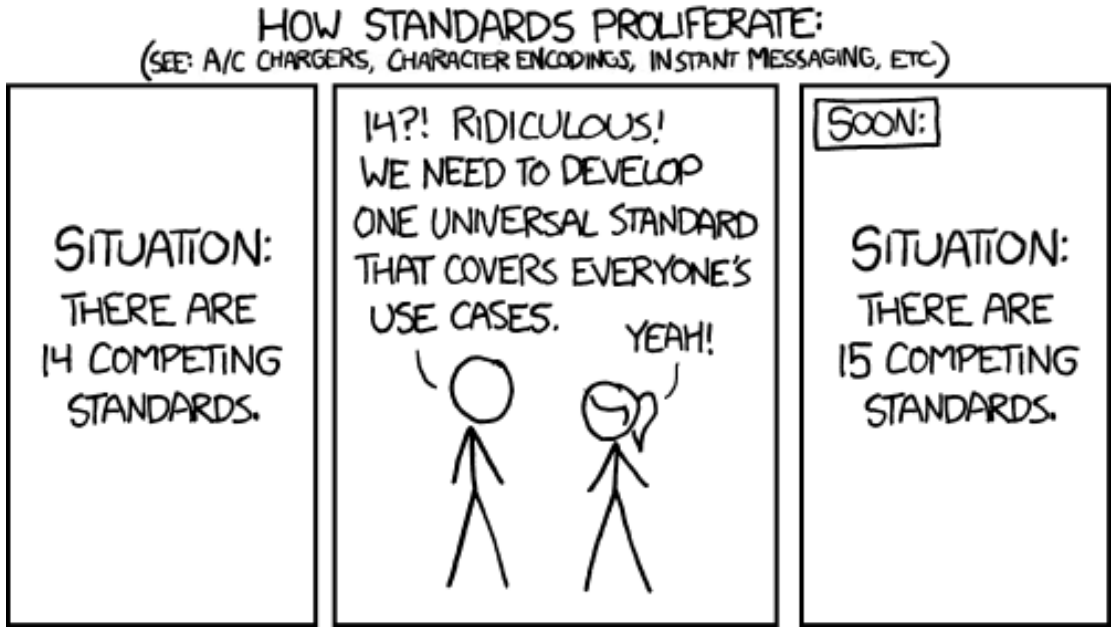
Over the years, various (ad-hoc) interfaces have been used to expose contest information:

- XML event feed, has grown organically since ∼2008
  initial use case: ICPC-live, scoreboard resolver
- Predictive scoreboard by ICPC analytics since 2012
- Submission forwarding (for shadowing) since 2014
- Various smaller single purpose APIs

Introducing one standard to rule them all

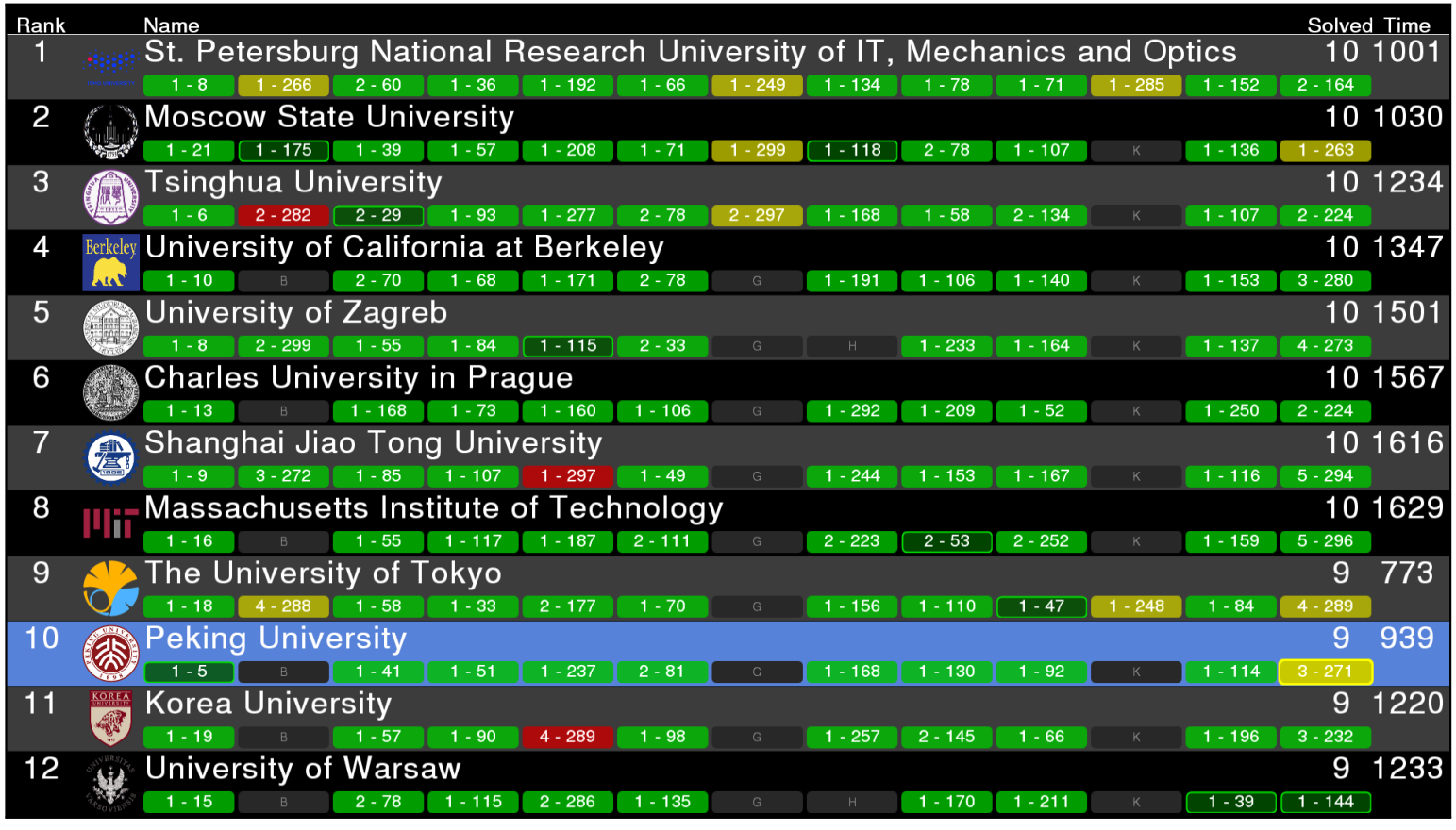
New **Contest API** standard to replace previous interfaces:

- Modern standard REST principles, JSON encoding
- Designed from scratch for general ICPC style contests
- In use at these World Finals!



Use cases

- External scoreboard, resolver (ICPC tools, analytics)



Rank	Name	Solved	Time
1	St. Petersburg National Research University of IT, Mechanics and Optics	10	1001
	1 - 8 | 1 - 266 | 2 - 60 | 1 - 36 | 1 - 192 | 1 - 66 | 1 - 249 | 1 - 134 | 1 - 78 | 1 - 71 | 1 - 285 | 1 - 152 | 2 - 164		
2	Moscow State University	10	1030
	1 - 21 | 1 - 175 | 1 - 39 | 1 - 57 | 1 - 208 | 1 - 71 | 1 - 299 | 1 - 118 | 2 - 78 | 1 - 107 | K | 1 - 136 | 1 - 263		
3	Tsinghua University	10	1234
	1 - 6 | 2 - 282 | 2 - 29 | 1 - 93 | 1 - 277 | 2 - 78 | 2 - 297 | 1 - 168 | 1 - 58 | 2 - 134 | K | 1 - 107 | 2 - 224		
4	University of California at Berkeley	10	1347
	1 - 10 | B | 2 - 70 | 1 - 68 | 1 - 171 | 2 - 78 | G | 1 - 191 | 1 - 106 | 1 - 140 | K | 1 - 153 | 3 - 280		
5	University of Zagreb	10	1501
	1 - 8 | 2 - 299 | 1 - 55 | 1 - 84 | 1 - 115 | 2 - 33 | G | H | 1 - 233 | 1 - 164 | K | 1 - 137 | 4 - 273		
6	Charles University in Prague	10	1567
	1 - 13 | B | 1 - 168 | 1 - 73 | 1 - 160 | 1 - 106 | G | 1 - 292 | 1 - 209 | 1 - 52 | K | 1 - 250 | 2 - 224		
7	Shanghai Jiao Tong University	10	1616
	1 - 9 | 3 - 272 | 1 - 85 | 1 - 107 | 1 - 297 | 1 - 49 | G | 1 - 244 | 1 - 153 | 1 - 167 | K | 1 - 116 | 5 - 294		
8	Massachusetts Institute of Technology	10	1629
	1 - 16 | B | 1 - 55 | 1 - 117 | 1 - 187 | 2 - 111 | G | 2 - 223 | 2 - 53 | 2 - 252 | K | 1 - 159 | 5 - 296		
9	The University of Tokyo	9	773
	1 - 18 | 4 - 288 | 1 - 58 | 1 - 33 | 2 - 177 | 1 - 70 | G | 1 - 156 | 1 - 110 | 1 - 47 | 1 - 248 | 1 - 84 | 4 - 289		
10	Peking University	9	939
	1 - 5 | B | 1 - 41 | 1 - 51 | 1 - 237 | 2 - 81 | G | 1 - 168 | 1 - 130 | 1 - 92 | K | 1 - 114 | 3 - 271		
11	Korea University	9	1220
	1 - 19 | B | 1 - 57 | 1 - 90 | 4 - 289 | 1 - 98 | G | 1 - 257 | 2 - 145 | 1 - 66 | K | 1 - 196 | 3 - 232		
12	University of Warsaw	9	1233
	1 - 15 | B | 2 - 78 | 1 - 115 | 2 - 286 | 1 - 135 | G | H | 1 - 170 | 1 - 211 | K | 1 - 39 | 1 - 144		

Use cases

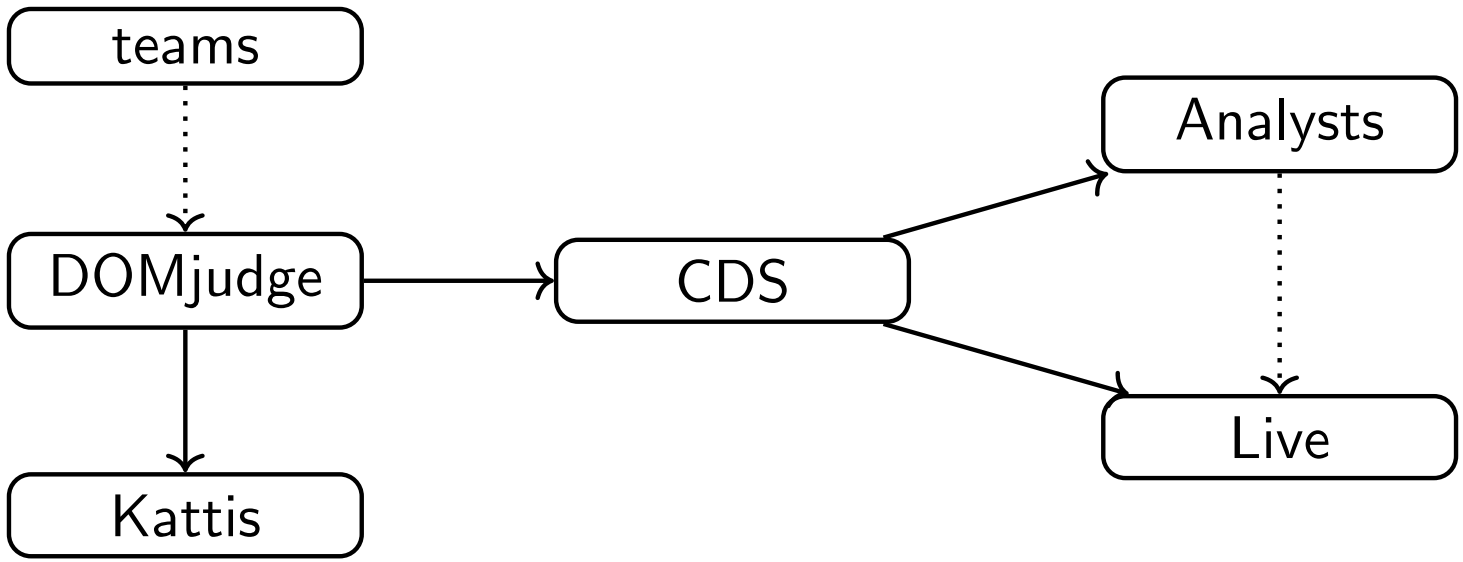
- External scoreboard, resolver (ICPC tools, analytics)
- Shadowing a CCS (Kattis/DOMjudge/PC^2)

Use cases

- External scoreboard, resolver (ICPC tools, analytics)
- Shadowing a CCS (Kattis/DOMjudge/PC^2)
- Proxy providing authentication, load balancer, . . . (CDS)

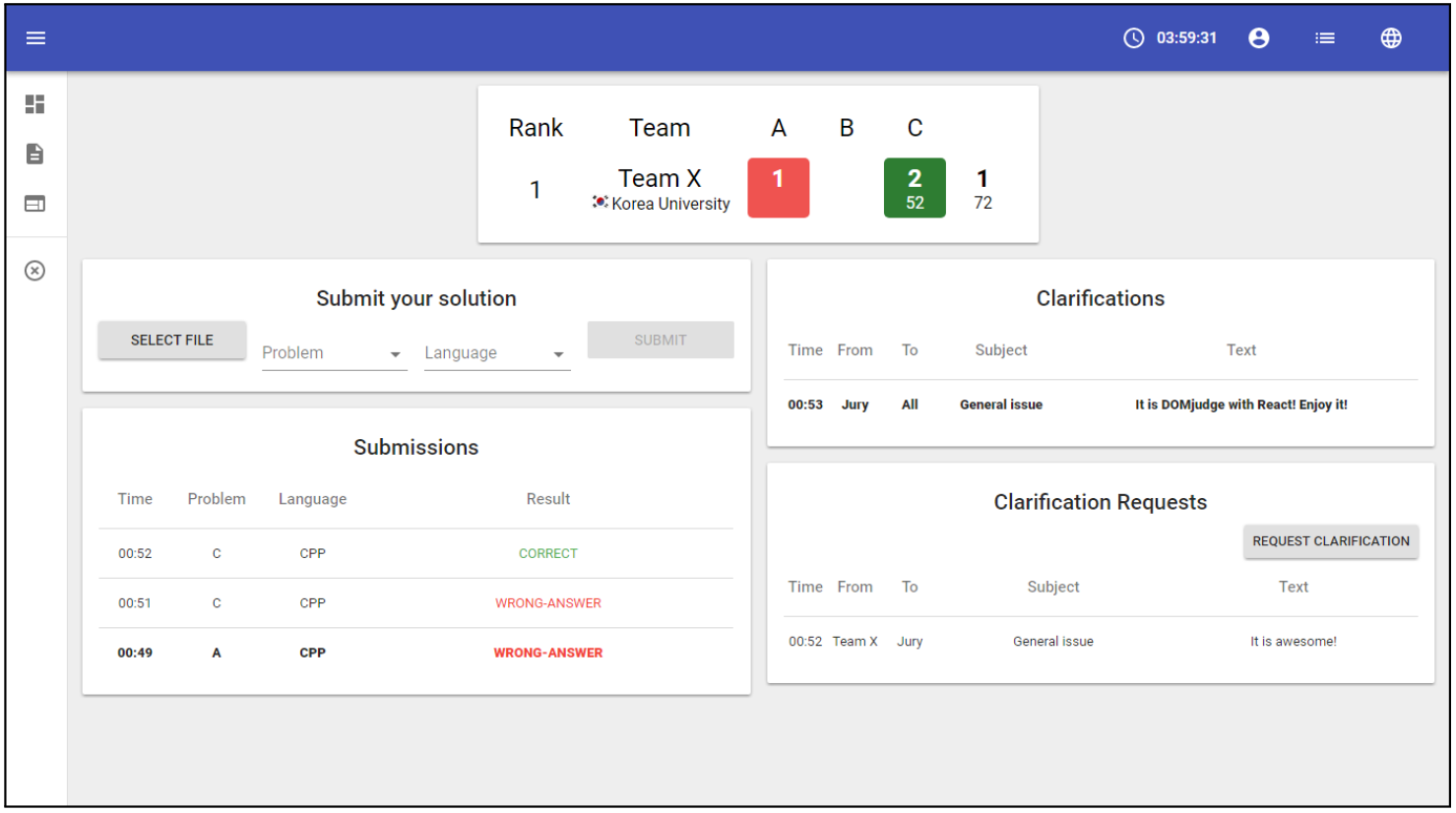
Use cases

- External scoreboard, resolver (ICPC tools, analytics)
- Shadowing a CCS (Kattis/DOMjudge/PC^2)
- Proxy providing authentication, load balancer, . . . (CDS)



Future use cases

- External CCS components
  - front-end web client react-domjudge



Future use cases

- External CCS components
  - front-end web client
  - submit client
  - judgedaemon (or kittens)

Future use cases

- External CCS components
  - front-end web client
  - submit client
  - judgedaemon (or kittens)
- Use for archiving format
  - Replay for testing purposes: CDS in a Box
  - Running a private *virtual* contest

A quick glance at the API endpoints

```
contests
  groups
  organizations
  teams
  team-members
```

A quick glance at the API endpoints

```
contests
  groups
  organizations
  teams
  team-members

  judgement-types
  languages
  problems
```

A quick glance at the API endpoints

```
contests
  groups
  organizations
  teams
  team-members

  judgement-types
  languages
  problems

  submissions
  judgements
  runs
  clarifications
```

A quick glance at the API endpoints

```
contests
  groups
  organizations
  teams
  team-members

  judgement-types
  languages
  problems

  submissions
  judgements
  runs
  clarifications

  state
  awards
  scoreboard
```

A quick glance at the API endpoints

```
contests
  groups
  organizations
  teams
  team-members

  judgement-types
  languages
  problems

  submissions
  judgements
  runs
  clarifications

  state
  awards
  scoreboard

  event-feed          (streaming NDJSON "changelog")
```

A new event feed

The new NDJSON event feed is a changelog of the REST endpoints, with `CREATE`, `UPDATE` and `DELETE` events.

A new event feed

The new NDJSON event feed is a changelog of the REST endpoints, with `CREATE`, `UPDATE` and `DELETE` events.

Examples:


```
{
   "id": "3241331",
   "type": "submissions",
   "op": "create",
   "data": {
      "id": "158412",
      "problem_id": "glyphrecognition",
      "language_id": "cpp",
      "team_id": "55",
      "contest_time": "4:00:11.048",
      "time": "2018-04-05T13:02:04.048+08:00",
      "files": [ { "href": "contests/finals/submissions/158412/files" } ]
   }
}
```

A new event feed

The new NDJSON event feed is a changelog of the REST endpoints, with `CREATE`, `UPDATE` and `DELETE` events.

Examples:


```
{
   "id": "3241309",
   "type": "state",
   "op": "update",
   "data": {
      "started": "2018-04-05T09:01:53.000+08:00",
      "frozen": "2018-04-05T13:01:53.000+08:00",
      "ended": null,
      "thawed": null,
      "finalized": null
   }
}
```

A new event feed

The new NDJSON event feed is a changelog of the REST endpoints, with `CREATE`, `UPDATE` and `DELETE` events.

Examples:

```
{
   "id": "3241553",
   "type": "team",
   "op": "delete",
   "data": {
      "id": "78"
   }
}
```

Improvements

- REST & JSON makes web page integration easy
- Standard HTTP(S) for transport simplifies auth, cache, ...
- Meant to be extensible; easier due to clearer layout

Improvements

- REST & JSON makes web page integration easy
- Standard HTTP(S) for transport simplifies auth, cache, ...
- Meant to be extensible; easier due to clearer layout

New NDJSON event feed replaces old XML feed:

- Dual/complimentary to REST endpoints
- Well-defined update and delete semantics
- Clear guarantees on referential integrity
- Line-based, easier to parse as streaming feed
- Specific types of events can be requested to reduce size

Still some work to be done...

The current release candidate is usable, but some more improvements to be made:

- More details on roles and access permissions
- Specify detailed requirements for `POST`, `PUT` and `PATCH`

Still some work to be done...

The current release candidate is usable, but some more improvements to be made:

- More details on roles and access permissions
- Specify detailed requirements for `POST`, `PUT` and `PATCH`

We expect to address these issues in a final release during/after the summer.

Feedback welcome!

Mailinglist: cliccs@ecs.csus.edu

The specification can be found at:

https://clics.ecs.baylor.edu/index.php/Contest_API

Questions?