

Analytics ICPC 2012

From ICPC-Contest Control Standard

Jump to:[navigation](#), [search](#)

Contents

- [1 Address in Warsaw](#)
- [2 WF Schedule/Plan, May 13-17](#)
- [3 WF Full Scale Test, March 16-18](#)
- [4 Preparations before Warsaw](#)
- [5 Preparations in Warsaw, May 13-16](#)
- [6 ICPC WF 2012, May 17](#)

Address in Warsaw

University of Warsaw
Faculty of Management
1/3 Szturmowa Street
02-678 Warszawa, Poland

WF Schedule/Plan, May 13-17

Availability of analysts

Person	Sun 13/5	Mon 14/5	Tue 15/5	Wed 16/5	Thu 17/5	Fri 18/5
Fredrik H	8-22	8-17, (17-18), 18-20, (20-22.30)	8-22	8-22	7.30-22	8-17
Fredrik N	8-22	8-22	8-22	8-22	8-22	8-22
Stein N	x	x	x	8-22	8-22	x
Tommy F	8-22	8-22	8-22	8-22	8-22	8-17
Greg H	13-22	8-22	8-22	8-22	8-22	8-22
David S	15-22	8-22	8-22	8-11, (11-15), 15-22	8-22	8-22
Jaap E	12-22	8-18	8-22	8-22	8-22	8-16
Jan K	12-22	8-22	8-22	8-22	8-22	8-16
Tobias W	9-22	8-18	8-22	8-22	8-22	8-12
Thijs K	12-22	8-13, 16-18	8-22	8-22	8-22	8-16
Michal F	x	x	x	8-22	8-22	8-22

- [WF Schedule](#)
- Sunday May 13
 - 8.00 Transport to FoM
 - 8.30 Daily meeting (preliminary)
 - MS1: All analytics computers set up and configured according to [Analytics Computer Configuration](#)

- MS2: iCAT installed and configured according to [iCAT Configuration](#)
- MS3: Kattalyzer installed and configured [Kattalyzer Configuration](#)
- MS4: Codealyzer installed and configured [Codealyzer Configuration](#)
- Monday May 14
 - 8.00 Transport to FoM
 - 8.30 Daily meeting (preliminary)
 - 16.30-17.30 Walking tour of Old Town, optional
 - 20.00-22.00 CLI Workshop, optional
 - MS5: Interaction with studio set up and tested
- Tuesday May 15
 - 8.00 Transport to FoM
 - 8.30 Daily meeting (preliminary)
 - 18.30-21.30 Opening Ceremony & Dinner
 - MS6: Interaction with ICPC Social agreed
- Wednesday May 16
 - 8.00 Transport to FoM
 - 8.30 Daily meeting (preliminary)
 - 13.00-15.00 CLI Workshop, optional (preferably not)
 - MS7: Everything ready for dress rehearsal
 - 15.00-17.00 Dress rehearsal, everything as for the competition
 - 17.30 Dress rehearsal debriefing
 - 18.00-20.00 UPE Dinner
- Thursday May 17 WF day
 - 7.30 Transport to FoM
 - 8.00 Daily meeting (preliminary)
 - MS8: Everything ready for competition
 - 8.30 Briefing on the problems by the judges

WF Full Scale Test, March 16-18

- [Finals Test Plan](#)
- Document how to set up a new contest, **partly done**
 - Backup database
 - Reset database
 - Load team.tsv
 - Load groups.tsv (?)
 - Load problems.tsv (?)
 - Save flags.tar.gz
 - Save logos.tar.gz
 - Get Top Coder rankings
 - Configure iCAT
 - Scoreboard
 - Configure Kattalyzer
 - Kattis event feed
 - Configure Codalyzer
- Make sure that the Kattalyzer works on the current event feed format **done**
- Make sure that accessing the contestants files works **done**
- Make a rudimentary implementation of the Codalyzer **not done**
 - Create a Python script which scans the contestants directories for new files
 - for each new file check it against a set of rules

- if a rule is triggered it can either save the file to a special directory and/or generate an iCAT message
- Discuss with the Kattis people about annotating test cases and running all test cases **done**
- Discuss with Fredrik about improvements he would like to see to the interface **not done**
- Discuss iCAT setup, ok to use webserver in Sweden? **done, ok**
- Discuss access to scoreboard, need fast access to a local copy **done, use Contest Model provided by Mattias**
- Discuss interactions with spectators and the general public **not done**

Preparations before Warsaw

- General issues
 - Clean up and write down the iCAT installation and configuration process on the iCAT wiki (Fredrik)
 - Store the Contest Model in a database? Then many programs can update and analyze it using different languages.
 - Which program ought to generate data for graphs and statistics? Kattalyzer could do most of it, however it does not have access to the contestant files analysis. With a central Contest Model this could be done by any program with access to the DB.
 - Wiki for internal communication
 - Get flag logos from ICPC Live (country_code.png)
 - Create an IRC channel for communicating with the external ICPC/programming community. This could also provide feedback on solutions to problems.
 - Try to visualize tricky corner cases that can be shown in the stream - here it would be good to get some input from the judges, which problems can be easily visualized. We then can use DOMjudge to find the interesting cases.
- Things to estimate
 - How many problems will the best solve?
 - How many will solve a problem?
 - Correlate time of Xth submission with how many will solve it by the end of the contest
 - Incorrect submissions indicate harder problems
 - See the time between solved
 - See time from first code to solved
 - Predict which problems are easy/hard, start from initial guess then improve with data from contest
 - log files from 2008 should exist, definitely from 2010 forward
 - Estimate a difficulty measure of the problem set relative to this years teams
 - See statistic for each region over time to see improvement (since number of submissions and solved is linear the slope of the curve indicates the relation to previous years' competitions and between region)
 - Finns det en korrelation mellan lösta problem/poäng och hur mycket tid som spenderas vid tangentbordet? Låt oss definiera att tangentbordet är övergivet om inget skrivits på 5 sekunder, då borde ni kunna läsa av hur mycket dödtid som finns. Vilket lag är mest aktivt?
 - Analyze how many teams use the same header for each problem, how many use FOR (or similar) macros to shorten their code. Do these habits correlate to the number of solved problems? Difficult: how often do teams change the person that is currently coding?
 - Show graphs on icpclive.com
- Kattalyzer
 - Integrate Mattias' Contest Model with Stein's Kattalyzer (Stein)
 - Rename Katalyzer to Kattalyzer (Stein/Fredrik)

- Integrate hypothetical solutions (Stein/Mattias)
- The currently generated graphs are images (which are generated somewhat slowly at least on my machine). I would recommend to use a JS library, e.g. <http://code.google.com/p/flot/> for that task instead. Nice example; <http://people.iola.dk/olau/flot/examples/visitors.html>
- Codalyzer
 - Find a suitable keylogger and ask SysOp to install it on the team machines (Fredrik)
 - Write initial version of the Codalyzer for analyzing the contestant's files (David)
 - For each team, find out which problems they have begun on, when they last changed it, and the edit frequency
 - List with unknown file names to improve matching during contest
 - Template detection
 - Compute intensity of work, time since last save,
 - Keylogger - count typing speed (characters per minute), count number of runs of eclipse (shift+f5)
 - How much erasing are the teams doing (backspace or remove the whole line)? Is there a correlation to their rank?
 - How long are the programs? Do the best teams write shorter or longer programs than the others? Compute average length per problem, language and team. Normalize relative to problem and language.
 - How many times do the teams run their programs with test data before submissions? (shift+F5 i Eclipse)
 - Aggregate information over regions and the contest
 - Keep track of which language(s) a team is using (part of Kattalyzer?) Correlate to results?
 - Check if a team is using more than one language and correlate to results.
- iCAT
 - Generate static names of teams and problems from Kattis feed; store in database instead of PHP? (Fredrik/Stein)
 - Create a scoreboard table where Kattalyzer can store the current scoreboard, including the hypothetical submissions (Fredrik/Stein)
 - Include information like "if team X solves problem Y within Z minutes (seconds?) they will gain W positions in the ranking" (for all X, Y). Also, the data presentation is simplified by the fact that for a given X many Ys will give the same Z and W as the only relevant property of the problem for this purpose is how many incorrect submissions the team have already sent.
 - This could be extended to a list of rankings they would get depending on how fast they solve the problem (0-10: 23 (+10); 11-32 30 (+3); ...).
 - Create a scoreboard (for ICPC-live internal use) with additional color(s) to encode that a team is actively working on a problem. This provides a quick graphical overview.
 - Organize the iCAT code as widgets that can be implemented in different ways and then put together to create UIs (Fredrik/Greg)
 - Design a proper MVC architecture for iCAT, create a PHP Contest Model Wrapper for the Contest Model in the DB, restructure the iCAT code according to the new design (Fredrik)
 - Include links to VNC, VLC and Kattis if possible to iCAT (Fredrik)
- iCAT Contest Model
 - Import TopCoder stats for the teams (how useful is this?)
- Interfacing with ICPC Live Graphics
 - Coordinate regarding what statistics and facts to display (Fredrik)
 - Total number of submissions and total number of solved linear
 - Contest map, region map to display who solves problems
 - A graph with time on one axis and teams/problems on the other and display a point when each problem is updated or team updates something (in the color of the problem)
 - It would be useful to have some way to provide the audience and/or the ICPC-live team with

contestants' code, with lines highlighted and annotated with remarks. What would be the best way to fit this into the current data structures and communication with Fredrik Niemelä (assuming he is the commentator) and the ICPC-live team? The same infrastructure could be used to present fun facts about submissions to the public.

- Interfacing with the judges
 - Provide Fredrik N with a list of requests from the judges (annotated test cases, presentations of problems and solutions in a as TV friendly format as possible, access to the problems, solutions and a preliminary order of difficulty preferably at least two hours in advance
- Interfacing with ICPC HQ
 - Ask Mikael R for updated contest attendees (for 2012) and updated contest results (for 2011)
- Interfacing with ICPC Digital
 - Ask them to collect interesting facts about the teams
 - Import collected facts
 - Create a separate interface for them?

Preparations in Warsaw, May 13-16

ICPC WF 2012, May 17

Retrieved from "https://clics.ecs.baylor.edu/index.php?title=Analytics_ICPC_2012&oldid=1637"

Navigation menu

Views

- [Page](#)
- [Discussion](#)
- [View source](#)
- [History](#)
- [PDF Export](#)

Personal tools

- [Log in](#)

Navigation

- [Main page](#)
- [Recent changes](#)
- [Random page](#)
- [Help](#)

Search

Tools

- [What links here](#)
- [Related changes](#)
- [Special pages](#)
- [Permanent link](#)
- [Page information](#)



- This page was last edited on 12 May 2012, at 17:00.
- Content is available under [Creative Commons Attribution-ShareAlike](#) unless otherwise noted.
- [Privacy policy](#)
- [About ICPC-Contest Control Standard](#)
- [Disclaimers](#)