Computing with BOINC

Pan Zan

Seminar Presentation for Course 312023Z

November 8, 2012

Outline

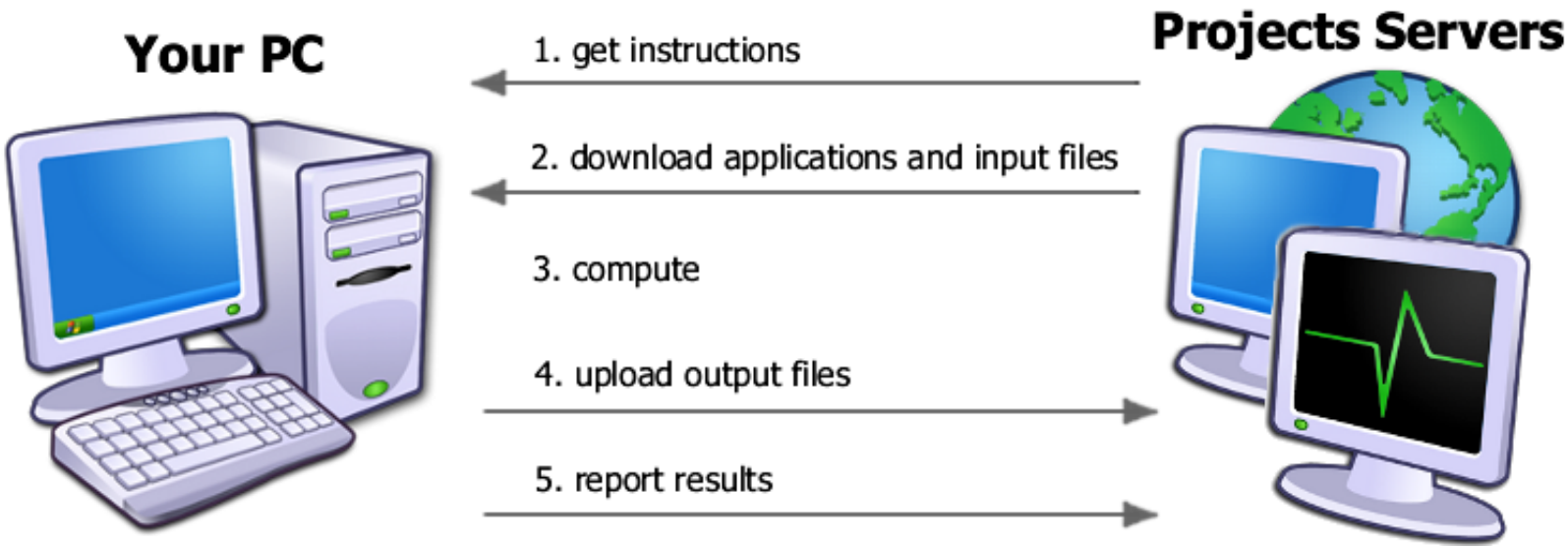
1. Introduction
2. Design and Structure
   - Client Structure
   - Server Structure
3. BOINC Applications
   - Customized Scheduling
   - Application Programming Interface
4. Project Management
   - Creating a Project
   - Handling Jobs
   - Maintaining the Project
5. Summary

Why is volunteer computing important?

- It can supply more computing power to science.
- It can't be bought; it must be earned.
- It encourages public interest in science.

Why is volunteer computing important?

- It can supply more computing power to science.
- It can't be bought; it must be earned.
- It encourages public interest in science.



What is BOINC?

Berkeley Open Infrastructure for Network Computing

Active users: 262,919 volunteers
Active hosts: 845,104 computers
24-hour average: 7.680 PetaFLOPS

What is BOINC?

Berkeley Open Infrastructure for Network Computing

Active users: 262,919 volunteers
Active hosts: 845,104 computers
24-hour average: 7.680 PetaFLOPS

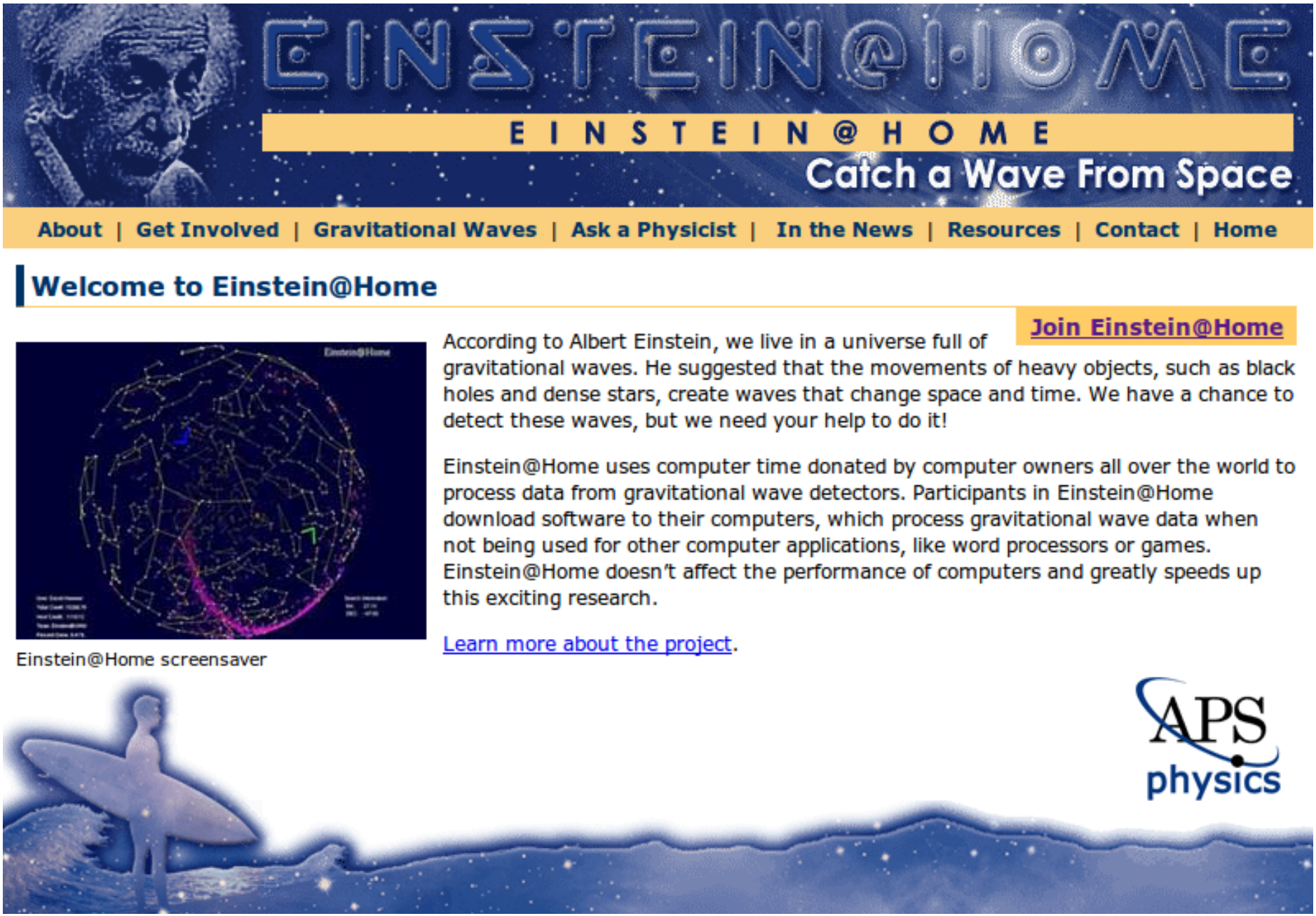
- originally developed to manage the SETI@home project
- the first project Predictor@home launched on June 2004
- over 50 projects on astronomy, biology, physics, games, ...
- licensed by LGPL, supported by Windows, Mac OS X, Unix

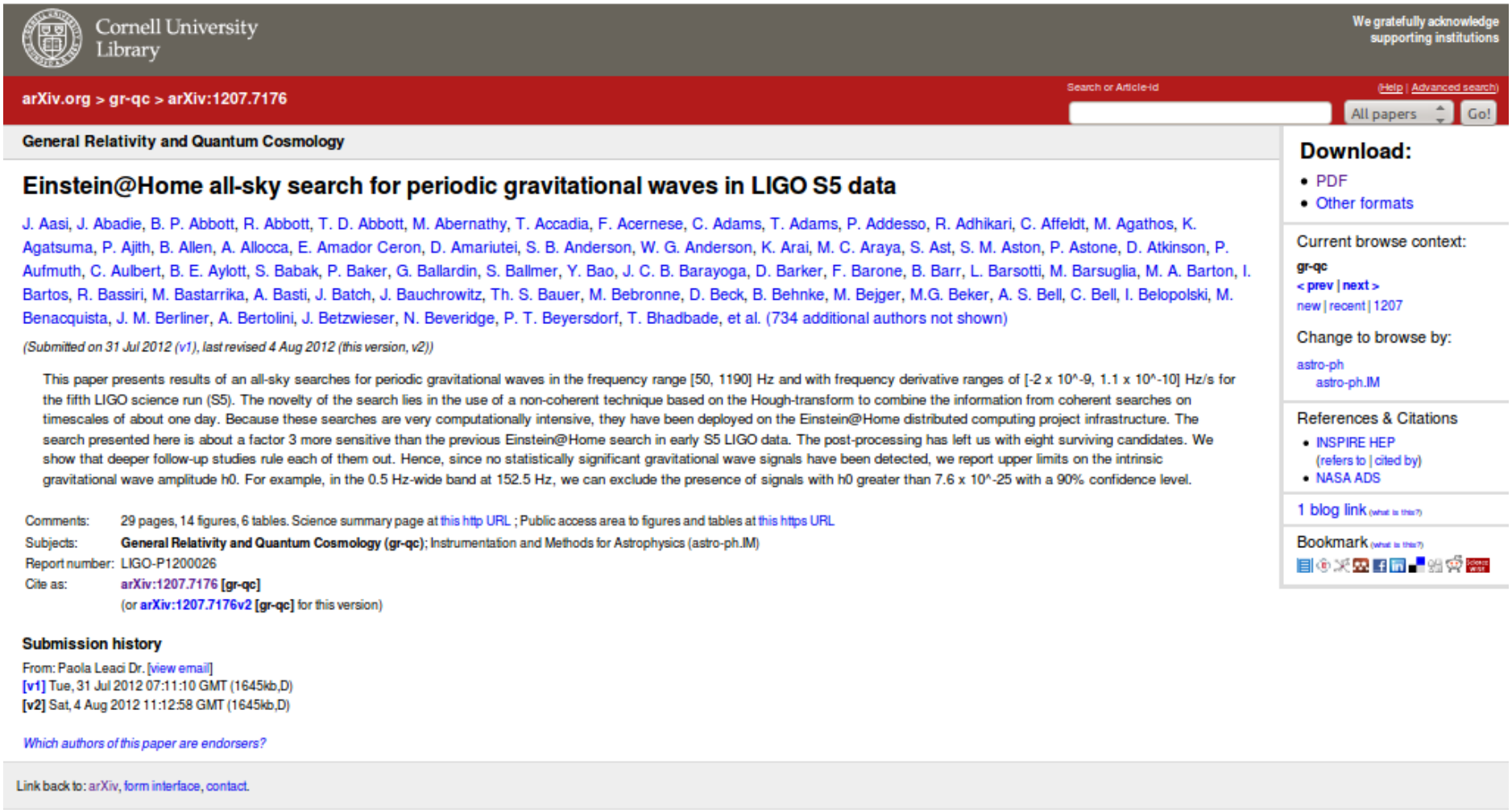
Project name	Users	last day	Hosts	last day	Teams	last day	Countries
BOINC combined	2,487,107	502	8,043,739	4,313	96,925	11	273
SETI@Home	1,336,610	314	3,280,733	749	60,929	3	233
World Community Grid	394,714	93	1,613,803	717	21,492	5	224
Rosetta@Home	349,530	71	1,087,724	212	9,965	2	225
Einstein@Home	330,143	64	3,072,916	4,106	10,470	0	221
Climate Prediction	263,902	0	539,909	0	7,635	0	221
MilkyWay@home	149,098	82	297,449	184	3,404	1	205
LHC@Home Classic	106,988	62	277,123	133	4,539	4	191
ABC@home	69,007	6	132,118	5	1,762	0	195
Malaria Control	65,140	28	153,332	74	2,154	0	208
Spinhenge@home	58,706	0	153,008	0	2,139	0	182
PrimeGrid	52,055	14	176,169	34	2,514	2	185
Cosmology@Home	50,200	27	93,989	61	1,708	2	188
QMC@Home	49,838	0	130,418	0	2,189	0	177
SIMAP	40,699	6	135,920	49	2,251	0	180
POEM@HOME	37,111	21	98,317	95	1,357	0	162
SZTAKI Desktop Grid	36,013	6	96,645	17	1,537	0	174
Docking@Home	30,477	12	79,990	35	1,056	0	140
Collatz Conjecture	30,010	7	72,425	47	1,334	0	158
uFluids	25,514	0	61,056	0	1,421	0	146
Enigma@Home	23,342	0	50,764	0	1,029	0	151
IBERCIVIS	19,625	0	54,098	0	814	0	124
Leiden Classical	19,152	0	60,087	8	1,219	0	142
GPUGRID	17,046	11	32,435	27	1,092	0	137

http://boincstats.com/en/stats/projectStatsInfo

Astronomy/Physics/Chemistry			
Einstein@home	Univ. of Wisconsin - Milwaukee, Max Planck Institute	Astrophysics	Mac OS X, Mac OS X (NVIDIA GPU), Mac OS X (AMD/ATI GPU), Linux/x86, Linux/x86 (NVIDIA GPU), Linux/x86 (AMD/ATI GPU), Mac OS X (PowerPC), SPARC Solaris 2.7, Windows, Windows (NVIDIA GPU), Windows (AMD/ATI GPU), Windows/x64 (AMD/ATI GPU), Linux/x64
LHC@home Test4Theory	CERN (European Organization for Nuclear Research)	Physics	Mac OS X, Linux/x86, Windows, Windows/x64, Mac OS X 64-bit, Linux/x64
Orbit@home	Planetary Science Institute	Astronomy	Mac OS X, Linux/x86, Windows
SETI@home	University of California, Berkeley	Astrophysics, astrobiology	Mac OS X, Linux/x86, Mac OS X (PowerPC), SPARC Solaris, SPARC Solaris 2.7, Windows, Windows (AMD/ATI GPU), Windows (NVIDIA GPU), Linux/x64
Cosmology@Home	University of Illinois at Urbana-Champaign	Astronomy	Linux/x86, Windows, Linux/x64
Leiden Classical	Leiden University, The Netherlands	Chemistry	Unknown
Spinhenge@home	Bielefeld University of Applied Sciences	Chemical engineering and nanotechnology	Unknown
eOn	University of Texas at Austin	Chemistry	Linux/x86, Windows, Mac OS X 64-bit, Linux/x64
Quantum Monte Carlo at Home	University of Muenster (Germany)	Chemistry	Linux/x86, Windows, Linux/x64
Milkyway@home	Rensselaer Polytechnic Institute	Astronomy	amd64-pc-freebsd, Mac OS X, Linux/x86, Linux/x86 (AMD/ATI GPU), Mac OS X (PowerPC), powerpc-linux-gnu, Windows, Windows (AMD/ATI GPU), Windows/x64, Windows/x64 (AMD/ATI GPU), Mac OS X 64-bit, FreeBSD/x86, Linux/x64, Linux/x64 (AMD/ATI GPU)
uFluids@home	Purdue University	Physics/Aeronautics	Windows
LHC@home	CERN (European Organization for Nuclear Research)	Physics	Linux/x86, Windows, Windows/x64, Mac OS X 64-bit, Linux/x64

http://boinc.berkeley.edu/projects.php


EINSTEIN@HOME
EINSTEIN@HOME
Catch a Wave From Space
About | Get Involved | Gravitational Waves | Ask a Physicist | In the News | Resources | Contact | Home
Welcome to Einstein@Home
Join Einstein@Home
According to Albert Einstein, we live in a universe full of gravitational waves. He suggested that the movements of heavy objects, such as black holes and dense stars, create waves that change space and time. We have a chance to detect these waves, but we need your help to do it!
Einstein@Home uses computer time donated by computer owners all over the world to process data from gravitational wave detectors. Participants in Einstein@Home download software to their computers, which process gravitational wave data when not being used for other computer applications, like word processors or games. Einstein@Home doesn't affect the performance of computers and greatly speeds up this exciting research.
Learn more about the project.
Einstein@Home screensaver
APS
physics



Cornell University Library

We gratefully acknowledge supporting institutions

arXiv.org > gr-qc > arXiv:1207.7176

Search or Article-id (Help | Advanced search) All papers Go!

General Relativity and Quantum Cosmology

Einstein@Home all-sky search for periodic gravitational waves in LIGO S5 data

J. Aasi, J. Abadie, B. P. Abbott, R. Abbott, T. D. Abbott, M. Abernathy, T. Accadia, F. Acernese, C. Adams, T. Adams, P. Addesso, R. Adhikari, C. Affeldt, M. Agathos, K. Agatsuma, P. Ajith, B. Allen, A. Allocca, E. Amador Ceron, D. Amariutei, S. B. Anderson, W. G. Anderson, K. Arai, M. C. Araya, S. Ast, S. M. Aston, P. Astone, D. Atkinson, P. Aufmuth, C. Aulbert, B. E. Aylott, S. Babak, P. Baker, G. Ballardin, S. Ballmer, Y. Bao, J. C. B. Barayoga, D. Barker, F. Barone, B. Barr, L. Barsotti, M. Barsuglia, M. A. Barton, I. Bartos, R. Bassiri, M. Bastarrika, A. Basti, J. Batch, J. Bauchrowitz, Th. S. Bauer, M. Bebronne, D. Beck, B. Behnke, M. Bejger, M.G. Beker, A. S. Bell, C. Bell, I. Belopolski, M. Benacquista, J. M. Berliner, A. Bertolini, J. Betzwieser, N. Beveridge, P. T. Beyersdorf, T. Bhadbade, et al. (734 additional authors not shown)

(Submitted on 31 Jul 2012 (v1), last revised 4 Aug 2012 (this version, v2))

This paper presents results of an all-sky searches for periodic gravitational waves in the frequency range [50, 1190] Hz and with frequency derivative ranges of [-2 x 10^-9, 1.1 x 10^-10] Hz/s for the fifth LIGO science run (S5). The novelty of the search lies in the use of a non-coherent technique based on the Hough-transform to combine the information from coherent searches on timescales of about one day. Because these searches are very computationally intensive, they have been deployed on the Einstein@Home distributed computing project infrastructure. The search presented here is about a factor 3 more sensitive than the previous Einstein@Home search in early S5 LIGO data. The post-processing has left us with eight surviving candidates. We show that deeper follow-up studies rule each of them out. Hence, since no statistically significant gravitational wave signals have been detected, we report upper limits on the intrinsic gravitational wave amplitude h0. For example, in the 0.5 Hz-wide band at 152.5 Hz, we can exclude the presence of signals with h0 greater than 7.6 x 10^-25 with a 90% confidence level.

Comments: 29 pages, 14 figures, 6 tables. Science summary page at this http URL ; Public access area to figures and tables at this https URL
Subjects: **General Relativity and Quantum Cosmology (gr-qc)**; Instrumentation and Methods for Astrophysics (astro-ph.IM)
Report number: LIGO-P1200026
Cite as: **arXiv:1207.7176 [gr-qc]**
(or **arXiv:1207.7176v2 [gr-qc]** for this version)

Submission history

From: Paola Leaci Dr. [view email]
[v1] Tue, 31 Jul 2012 07:11:10 GMT (1645kb,D)
[v2] Sat, 4 Aug 2012 11:12:58 GMT (1645kb,D)

Which authors of this paper are endorsers?

Link back to: arXiv, form interface, contact.

Download:
- PDF
- Other formats

Current browse context:
gr-qc
< prev | next >
new | recent | 1207

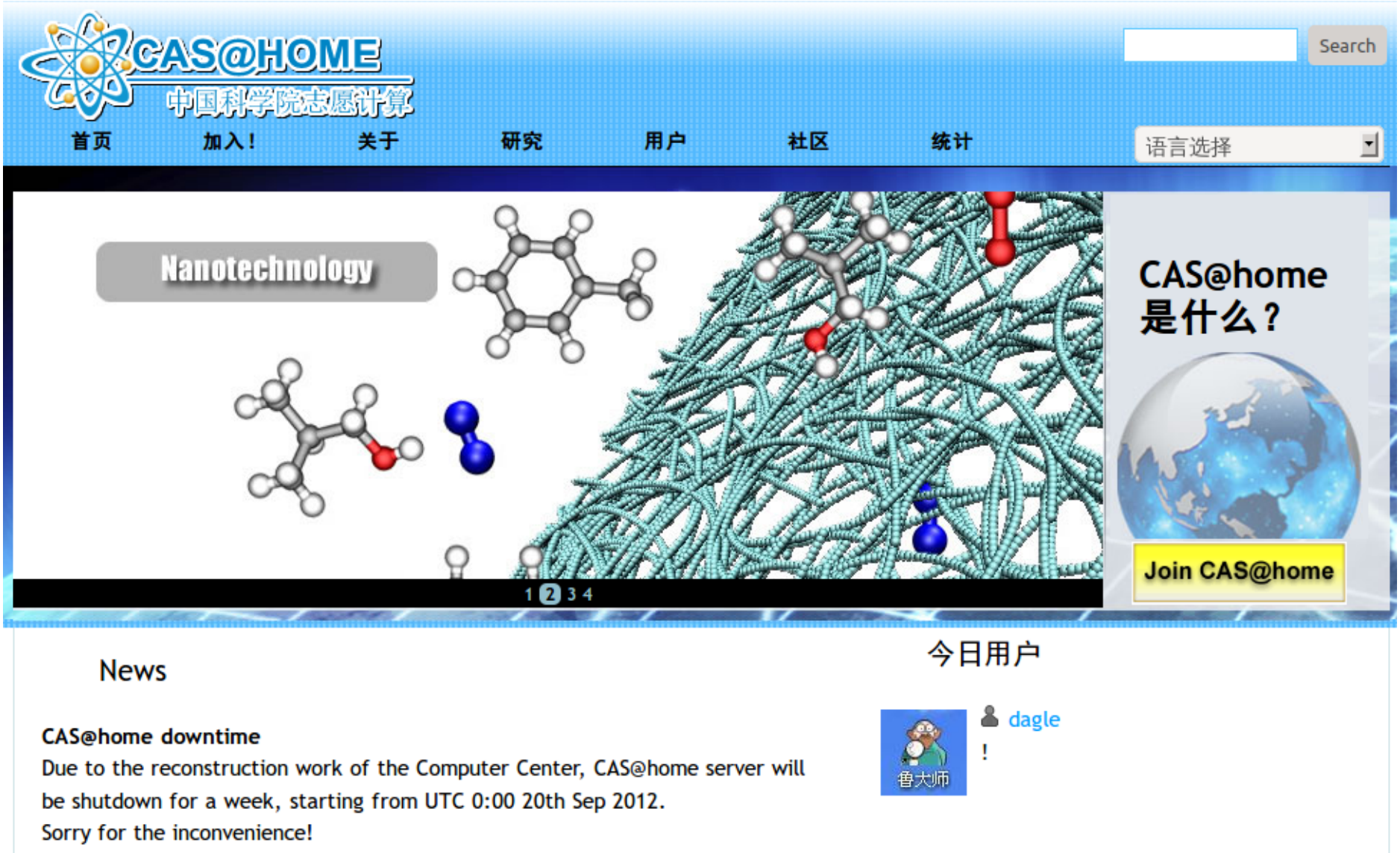
Change to browse by:
astro-ph
astro-ph.IM

References & Citations
- INSPIRE HEP (refers to | cited by)
- NASA ADS

1 blog link (what is this?)

Bookmark (what is this?)

http://arxiv.org/abs/1207.7176



http://casathome.ihep.ac.cn/index.php

More than volunteer computing

- create a Virtual Campus Supercomputing Center
- as a grid platform, e.g. the SZTAKI desktop grid project
- integrate BOINC with Condor-G, Globus GRAM, GRAM-WS

BOINC Manager

Easy to get started:

- Choose projects
- Download and run Boinc software
- Enter your account information

BOINC Manager

Easy to get started:

- Choose projects
- Download and run Boinc software
- Enter your account information

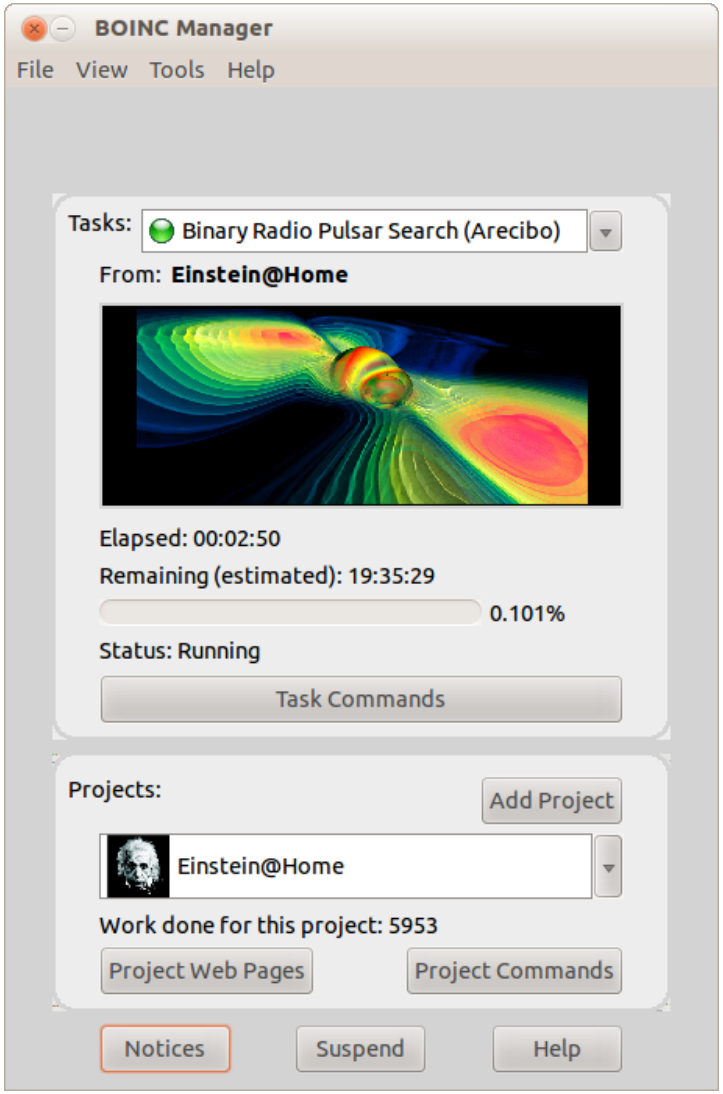
- Client-server architecture
- Remote procedure call mechanism
- Running as a daemon on Unix
- Command line and BOINC manager
- Advanced view and simplified GUI

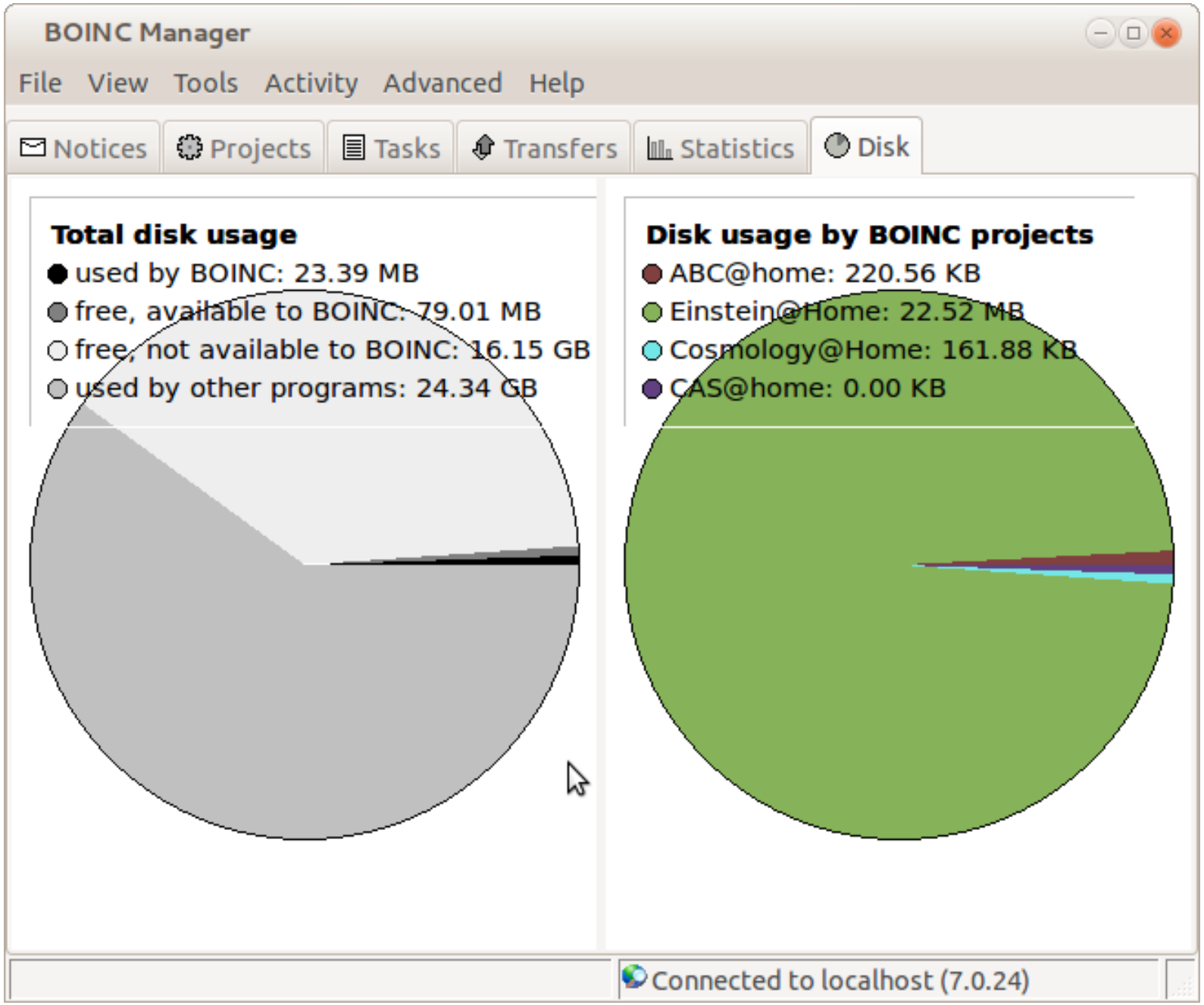
BOINC Manager

Easy to get started:

- Choose projects
- Download and run Boinc software
- Enter your account information

- Client-server architecture
- Remote procedure call mechanism
- Running as a daemon on Unix
- Command line and BOINC manager
- Advanced view and simplified GUI

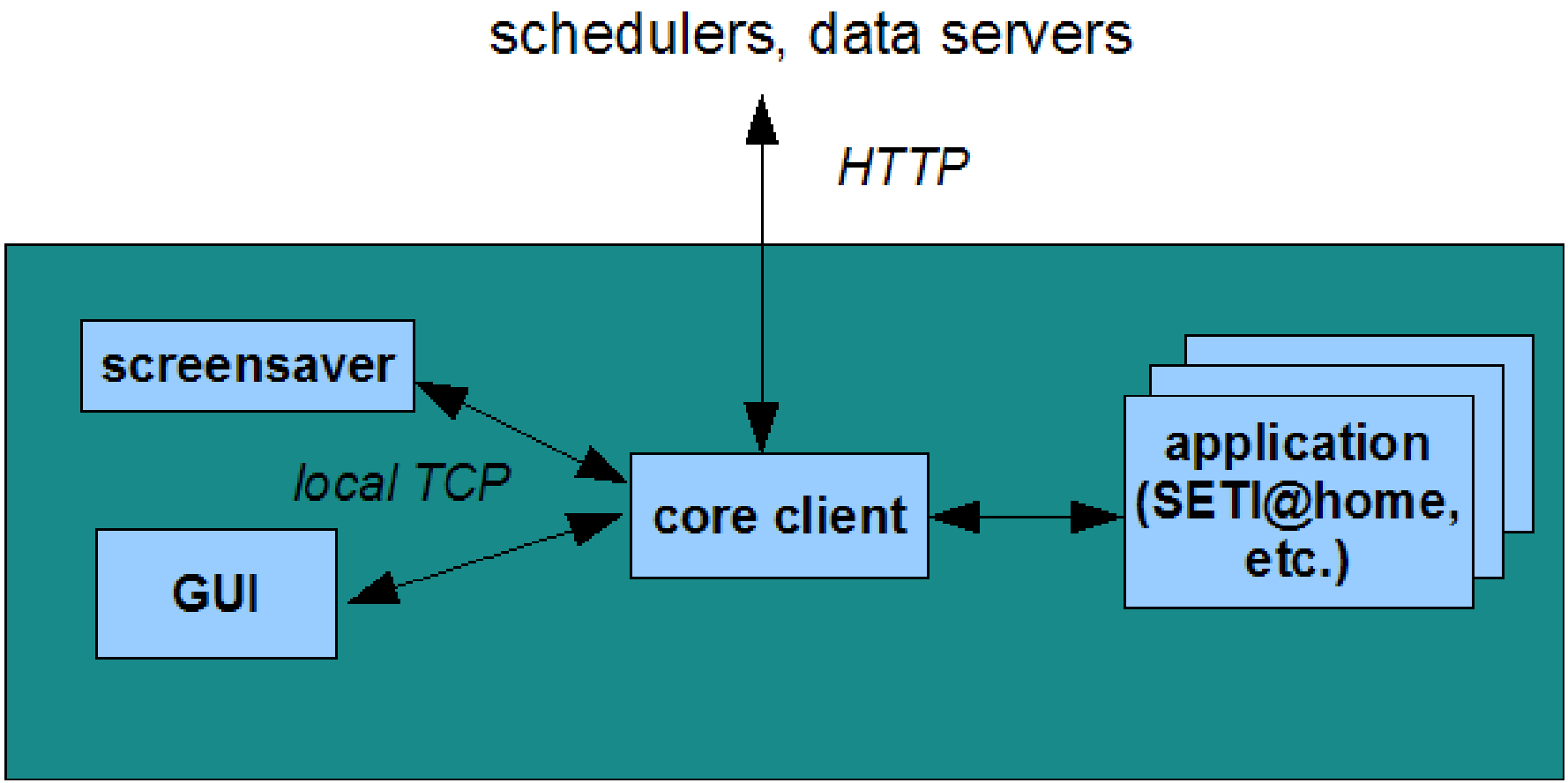



BOINC Manager
File View Tools Activity Advanced Help
Notices Projects Tasks Transfers Statistics Disk
Total disk usage
used by BOINC: 23.39 MB
free, available to BOINC: 79.01 MB
free, not available to BOINC: 16.15 GB
used by other programs: 24.34 GB
Disk usage by BOINC projects
ABC@home: 220.56 KB
Einstein@Home: 22.52 MB
Cosmology@Home: 161.88 KB
CAS@home: 0.00 KB
Connected to localhost (7.0.24)

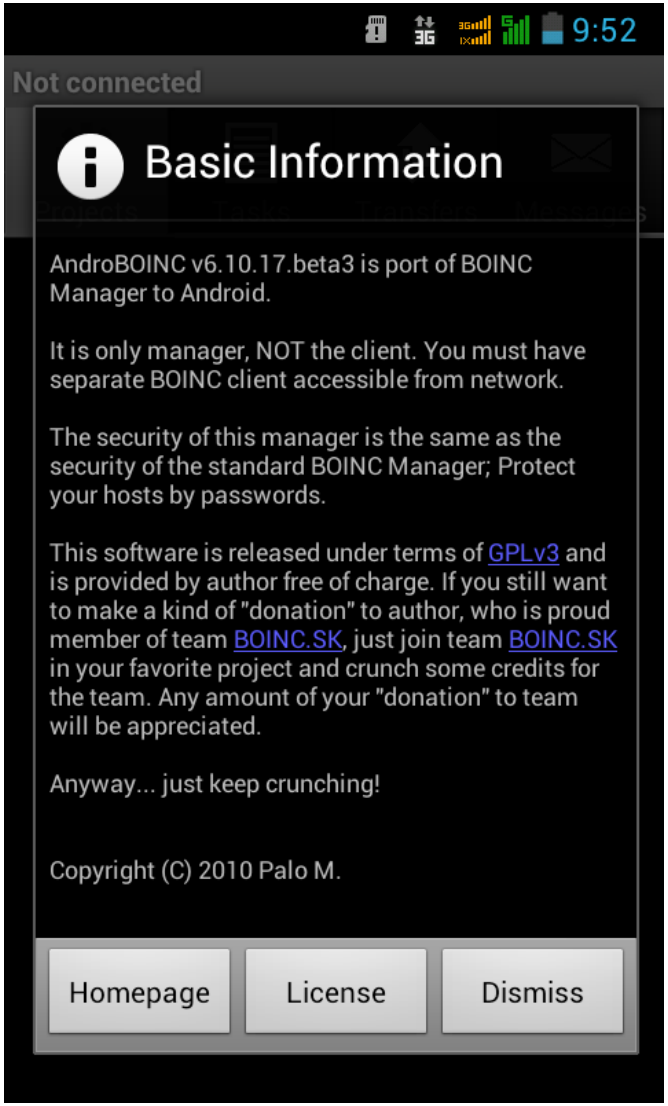
Boinccmd tool

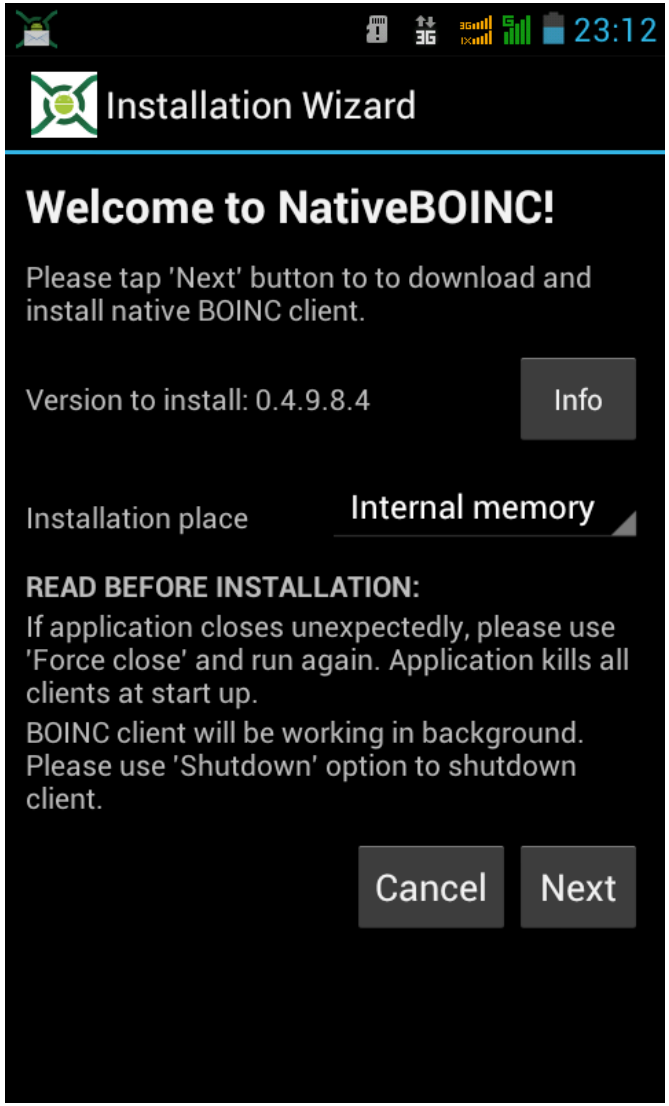
```
panzan@panzan-laptop:~$ boinccmd --version
boinccmd,  built from BOINC 7.0.24
panzan@panzan-laptop:~$ boinccmd --set_run_mode auto
panzan@panzan-laptop:~$ boinccmd --get_disk_usage
======== Disk usage ========
total: 43589697536.000000
free: 15335788544.000000
1) -----------
   master URL: http://abcathome.com/
   disk usage: 0.22MB
2) -----------
   master URL: http://einstein.phys.uwm.edu/
   disk usage: 22.52MB
3) -----------
   master URL: http://www.cosmologyathome.org/
   disk usage: 0.16MB
4) -----------
   master URL: http://casathome.ihep.ac.cn/
   disk usage: 0.00MB
panzan@panzan-laptop:~$
```

How the software works



BOINC for Android





Components and features

- Linux, Apache, MySQL, and PHP
- work generator, scheduler, feeder, validator, assimilator, file_deleter, transitioner

Components and features

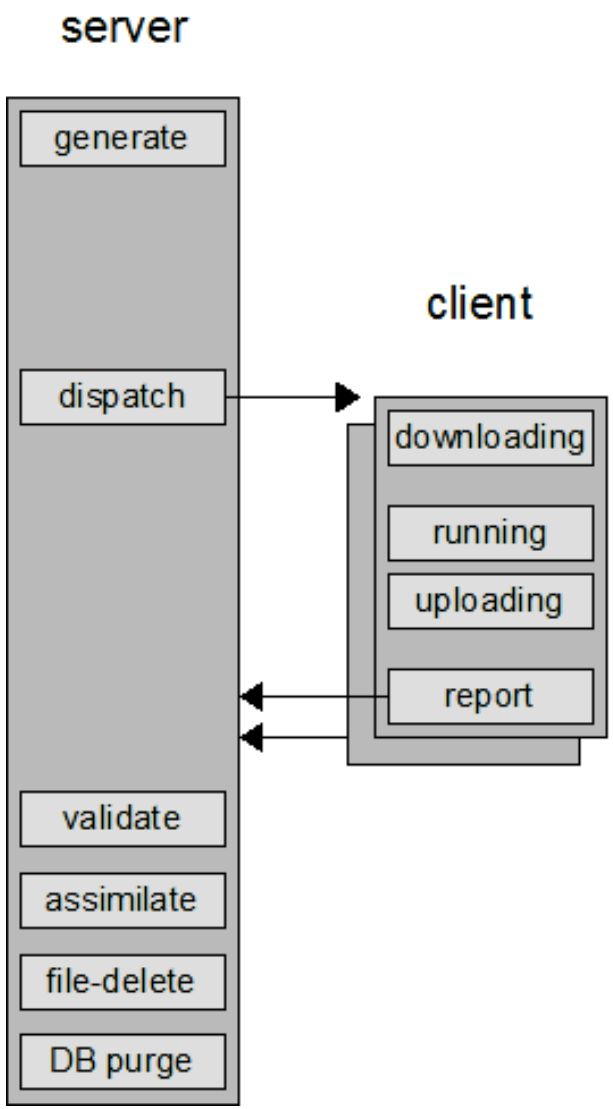
- Linux, Apache, MySQL, and PHP
- work generator, scheduler, feeder, validator, assimilator, file_deleter, transitioner

- homogeneous redundancy
- workunit trickling
- locality scheduling
- work distribution based on host parameters

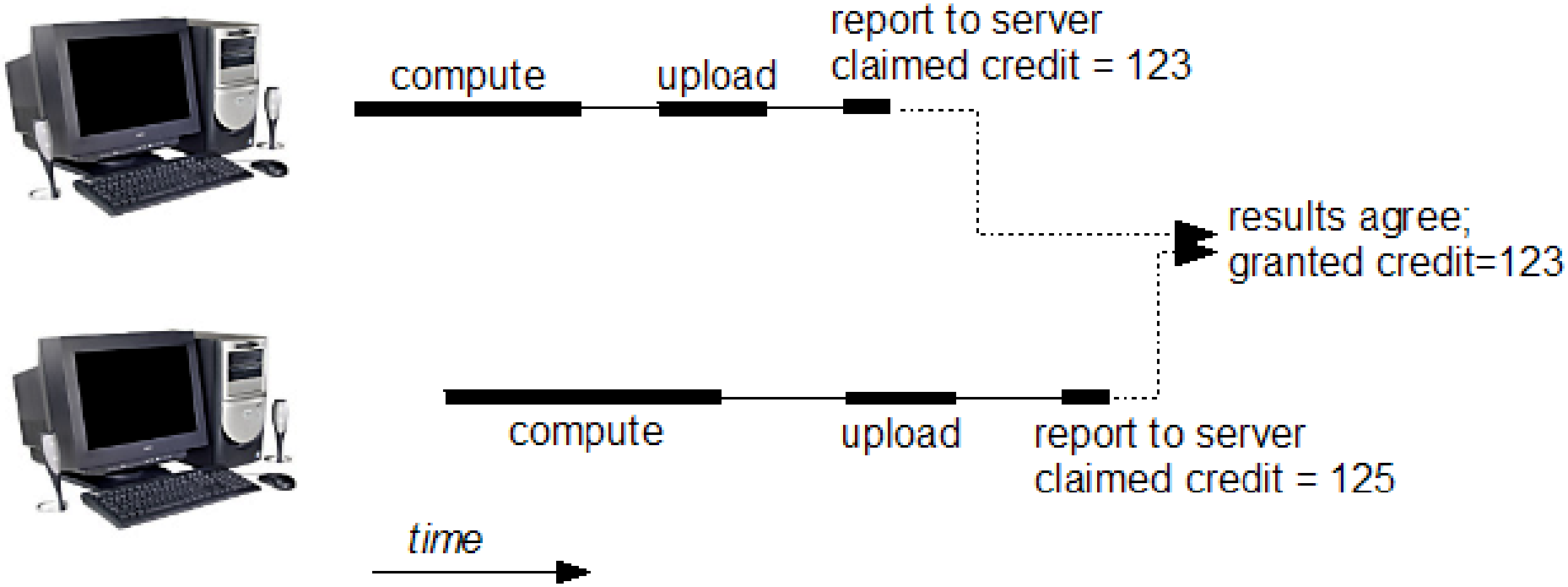
Components and features

- Linux, Apache, MySQL, and PHP
- work generator, scheduler, feeder, validator, assimilator, file_deleter, transitioner

- homogeneous redundancy
- workunit trickling
- locality scheduling
- work distribution based on host parameters



Credit system



https://boinc.berkeley.edu/wiki/How_BOINC_works

Certificate of Computation

This certifies that

panzan

has participated in Einstein@Home since 1 September 2010, and has contributed 6,453 Cobblestones of computation (5.58 quadrillion floating-point operations) to Einstein@Home.

2 November 2012

Security issues

Result or credit falsification	replication and validation
Malicious executable distribution	code signing
Overrun of data server	upload certificates
Theft of project files	encryption (little effect)
Theft of participant account info	firewall, encrypted protocols
Intentional abuse of participant hosts	account-based sandboxing
Accidental abuse of participant hosts	pre-released testing

http://boinc.berkeley.edu/trac/wiki/SecurityIssues

Job wrapper

- Any application can be run under BOINC using a wrapper.
- The wrapper handles all communication with the core client.
- The job file describes a sequence of tasks.

```
<job_desc>
    <task>
        <application>       worker    </application>
        <stdin_filename>    stdin     </stdin_filename>
        <stdout_filename>   stdout    </stdout_filename>
        <command_line>      10        </command_line>
    </task>
    <task>
        <application>       worker2   </application>
        <stdin_filename>    stdin2    </stdin_filename>
        <stdout_filename>   stdout2   </stdout_filename>
        <command_line>      10        </command_line>
    </task>
</job_desc>
```

Plan classes

- whether an application should run on a particular host;
- what resources it will use;
- how fast it is expected to run.

```
<plan_classes>
    <plan_class>
        <name>                  mt       </name>
        <min_ncpus>             2        </min_ncpus>
        <max_threads>           16       </max_threads>
        <projected_flops_scale> .99      </projected_flops_scale>
    </plan_class>
    <plan_class>
        <name>                  ati      </name>
        <gpu_type>              amd      </gpu_type>
        <min_driver_version>    1000000  </min_driver_version>
        <min_gpu_ram_mb>        250      </min_gpu_ram_mb>
        <gpu_ram_used_mb>       250      </gpu_ram_used_mb>
        <gpu_peak_flops_scale>  .2       </gpu_peak_flops_scale>
        <cpu_frac>              .1       </cpu_frac>
    </plan_class>
</plan_classes>
```

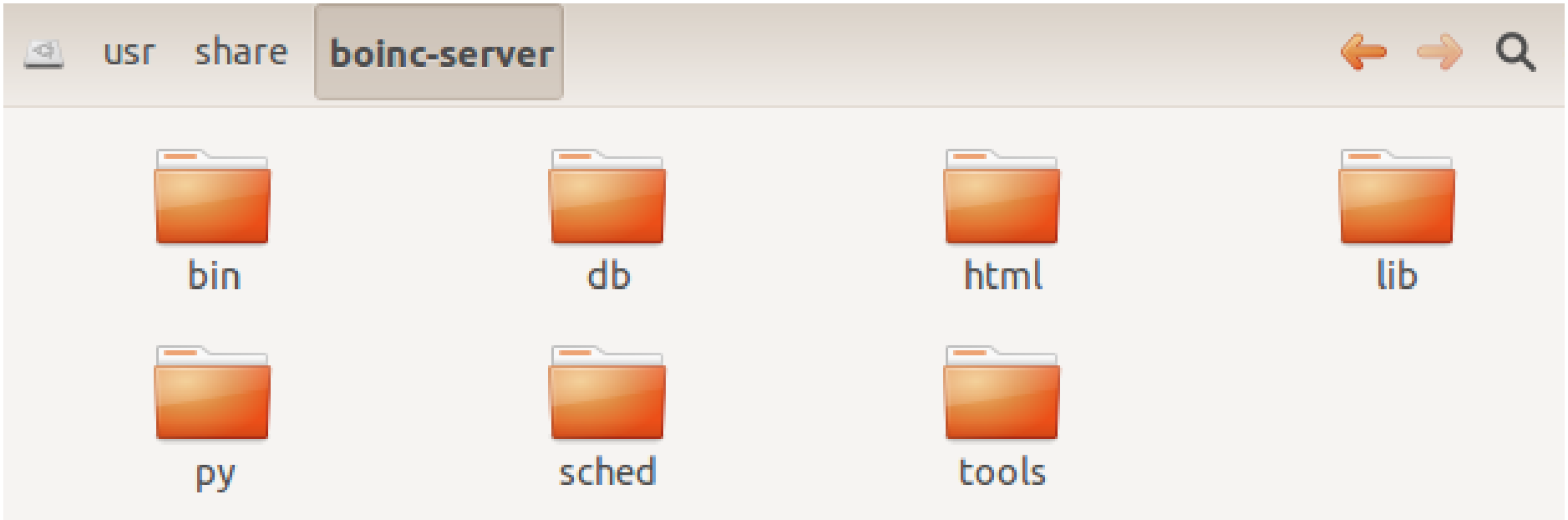
Basic APIs

- the header: `#include "boinc_api.h";`
- initialization: `boinc_init();`
- termination: `int boinc_finish(int status);`
- resolving file names: `int boinc_resolve_filename(char *logical_name, char *physical_name, int len);`
- I/O wrappers: `boinc_fopen(char* path, char* mode);`
- checkpointing: `int boinc_time_to_checkpoint();`
- critical sections: `void boinc_begin_critical_section();` `void boinc_end_critical_section();`
- reporting progress: `boinc_fraction_done(double fraction_done);`

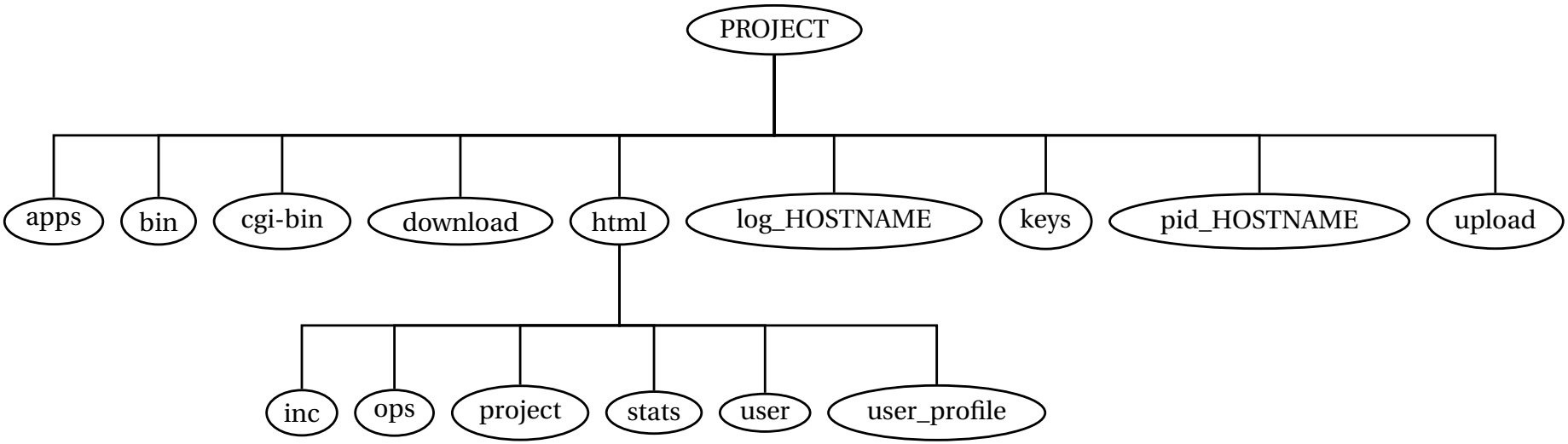
What is a project?

database + directory structure + configuration file

- It is easy to create a project with the `make_project` script
- Each project is publicly identified by a master URL.
- A project can be started or stopped by using control scripts.



Directory structure



The upload and download directories may contain millions of files. For efficiency they are normally organized as a hierarchy of subdirectories.

Single job submission

`boinc_submit [boinc-options] program [program-options]`

- `--infile`: specifie an input file
- `--stdin`: direct the given file to the program's stdin
- `--outfile`: specifie an output file
- `--stdout`: direct the program's stdout to a given file
- `--platform`: on which the program is to be run
- `--jobs`: show a list of jobs both in progress and completed
- `--abort`: abort a given job

Job processing

- A work generator creates jobs.
- A validator compares replicated results and selects one of them as 'canonical', or correct.
- An assimilator handles validated results, storing them in an archive or database.
- A feeder creates a shared-memory segment used to pass database records to CGI scheduler processes.
- A transitioner handles state transitions of workunits and results.
- BOINC has two mechanisms that let you control what hosts a job runs on: broadcast jobs and targeted jobs.

Other tools

- `xadd`: add platform and application records to the database
- `demo_submit`: submit a job for existing application
- `demo_query`: query a job created with demo_submit
- `status`: show the status of all daemons and tasks
- `cancel_jobs`: cancel the job(s) with the given name or IDs
- `update_versions`: create application versions
- `crypt_prog`: create a key pair for coding signing
- `sign_executable`: sign executable files
- `manage_privileges`: grant or revoke the manage privileges

Administrative Web Interface

- Browse the database
- Screen user profiles
- Create and edit applications and app versions
- Send mass email to users
- See a distribution of how many FLOPs results are using
- Cancel workunits
- View recent results, and analyze failures
- Browse stripcharts
- Browse log files

Project web site

- Customize the default web content
- Customize web appearance with CSS
- Make web pages translatable
- Create and manage message boards
- Protecte message boards from spam
- Add a wiki to the project
- Integrate web pages with WordPress
- Project skins, newsletters, notices
- Get more people to participate

Review of key points

- BOINC is an open-source middleware system for volunteer and grid computing.
- The BOINC framework consists of two layers which operate under the client-server architecture.
- The wrapper runs the applications as subprocesses, and handles all communication with the core client.
- To build a BOINC project, you should have these skills: C++, Shell, Python, PHP, MySQL, XML, HTML, CSS, JavaScript.
- Typically you need to develop these three application-specific programs: a work generator, a validator, and an assimilator.

- http://boinc.berkeley.edu/dev/
- http://www.equn.com/wiki/BOINC
- http://en.wikipedia.org/wiki/BOINC
- http://legion.pucp.edu.pe/wiki/index.php
- http://bioinfo.cs.technion.ac.il/superlink-online/
- http://desktopgrid.hu/
- http://boinc.gorlaeus.net/F2c.php
- http://www.primegrid.com/
- http://www.chess960athome.org/
- http://qah.uni-muenster.de/
- http://www.worldcommunitygrid.org/

Thank you!