Repository of Availability Traces
(Beta version 0.1)

Brighten Godfrey <pbg@cs.berkeley.edu>

October 22, 2006

Contents

1 Introduction

This package includes, in a common file format, a distilled form of the data from a number of studies of machine availability and failure. The goal is to build a repository of availability data to make it easy for distributed systems researchers to obtain, use, and compare the data sets.

Before beginning, there are two things to keep in mind. First, the authors of the original studies [1–4, 6] deserve the majority of credit for the existance of this package, due to their time in producing the data and for allowing their data sets to be included. If you use this package in your papers, you are encouraged to directly cite the original studies whose data you use, rather than only citing this package. You can grab the citations from `availability.bib`, included with this distribution.

Second, no measurement is perfect. Additionally, in reducing the data to a common format, some information is necessarily lost, such as the exact fraction of pings that failed in one probe, or the latency of a ping. I have tried to mention some of the main caveats in this document, but for a full description of the data collection methodology you should refer to the original publications.

This package is maintained by Brighten Godfrey. Comments, contributions, etc. are welcome.

2 File format

The data sets have been reduced to the following simple `.avt` ("availability trace") file format. (This format was adopted from that used by Saroiu et al [5].) Each row is either a comment beginning with #, or contains the following:

- A node identifier X (any non-whitespace), such as an IP address, a hash of something, etc.
- Integer number n of X's sessions (contiguous intervals during which node X was available).

- List of real-valued start and end times of each session, in the form `<start> <end> <start> <end> ...`

The units of time are not specified. However, in all the traces currently included in this distribution, time is measured in seconds.

For example, in the following, there are three nodes and the third has a single session which lasts for 9500 time units.

```
# Example .avt file
# Lines beginning with # are comments.
128.2.1.2       2    0.0  1.0  5.6  17
136.152.132.74  0
127.0.0.1       1    500 10000
```

Note that a node can only "go up" when measurements for that node begin, even if it had already been up for some time, and a node "goes down" when the measurements stop even if it was up for some time afterwards.

3 The traces

PlanetLab All Pairs Ping (Stribling [6]): this data set consists of pings sent every 15 minutes between all pairs of 200-400 PlanetLab nodes from January, 2004, to June, 2005. (I plan to add the full data set in later versions of this package.) Of the traces in this package, All Pairs Ping is the only one which is not based on probes from a single source node. Several versions of this trace are included:

- `pl-app.avt`: We consider a node to be up at some time t when, in the batch of pings most immediately prior to t, at least half of the pings sent to it succeeded.
- `pl-app-cleaned.avt`: In a number of periods, all or nearly all PlanetLab nodes were down, most likely due to planned system upgrades or measurement errors. To exclude these cases, the `cleanse` tool was run on the above trace; see Section 4 for details.
- `pl-app-oneping.avt`: the same as `pl-app.avt`, except a node is considered to be up when at least one ping sent to it succeeds.
- `pl-app-oneping-cleaned.avt`: the same as `pl-app-oneping.avt`, after being run through the `cleanse` tool.
- `pl-app-cleaned-v3.oneping.avt`: the same as `pl-app-cleaned.avt`, except restricted to the period of time after the PlanetLab v3 rollout, which was completed in December 2004. During the rollout period, roughly late October until early December, 2004, PlanetLab nodes had an uncharacteristically high churn rate, about an order of magnitude higher than average. Thus, the rollout period may not be representative of PlanetLab as a whole (or it may provide a good stress test).

Web sites (Bakkaloglu et al [1]): This trace is based on probes sent from a single machine at Carnegie Mellon to 129 web sites every 10 minutes from September, 2001, to April, 2002. Each probe consisted of an HTTP request for a file on the web server, and was considered successful when the server responded and the response included `HTTP OK` in the header. As before, a node is up at some time t when the probe sent to it most immediately prior to t was successful. Since there is only a single source and the trace is long and not limited to one local network, network connectivity problems near the source result in periods when nearly all nodes are unreachable. (The `cleanse` tool is probably too simplistic to be a reliable filter in this case.) In the original paper, some web sites were excluded, in part due to protocol incompatibilities; further details are not given but the paper lists the 99 servers that were used. The full trace is included in the file

- `web_sites.avt`

Microsoft PCs (Bolosky et al [2]): 51,662 desktop PCs within Microsoft Corporation were pinged every hour for 35 days beginning July 6, 1999. This trace is included in the file

- `microsoft.avt`

DNS Servers (Pang et al [4]): This data set consists of probes initiated at exponentially-spaced intervals with mean 1 hour over a period of 2 weeks to 62,201 local DNS (LDNS) servers. Each probe consisted of up to three ICMP and DNS pings. To distill the data set, each probe was marked with the time that the first ping in the probe was sent (the exact time of each ping is not available in the data set). A node is up at time t when any of the ICMP or DNS pings in the probe marked as being most immediately prior to t was successful. A higher-frequency, shorter-duration dataset was produced by the same study but is not included here. The authors of [4] have already applied heuristics to weed out network errors, and almost half the nodes experience no failures at all. The trace is included in the file

- `ldns.avt`

Skype superpeers (Guha et al [3]): A set of 4000 nodes participating in the Skype superpeer network were sent an application-level ping every 30 minutes for one month beginning September 12, 2005. As before, a node is up at a given time if the most recent ping succeeded. In private communication Saikat Guha noted three artifacts in the trace: (1) an initial trial period during which only about 200 nodes were pinged; (2) a day-long outage at the measurement site near the end of the trace; (3) a number of instantaneous spikes throughout the trace due to network problems near the measurement site. The distilled trace is included in the file

- `skype.avt`

4 Visualization and other tools

The tools directory includes several programs and libraries written in OCaml (`http://caml.inria.fr`). To use them, install OCaml and run `make` in that directory. The tools assume a Unix-like environment with `gv` and `gnuplot` installed.

`vis` is the most useful tool. It will show you things like the number of nodes up vs. time, session time distributions, basic statistics about the trace, etc.

`split` splits a trace in some number of equal parts.

`cleanse` is the procedure that was run on the PlanetLab trace, and works as follows. For each period of downtime at a particular node, we remove that period (i.e. we consider the node up during that interval) when the average number of nodes up during that period is less than half the average number of nodes up over all time. This is just an heuristic which appeared to work well for the 18-month portion of the PlanetLab All Pairs Ping trace included in this package; use it with care.

`trace.ml` provides some useful routines for loading, manipulating, and storing `.avt` files, for those of you writing in OCaml.

5 Related papers

In addition to the original publications, the following papers analyze properties of the traces included in this distribution (as opposed to using the traces as realistic inputs to test some system):

- http://www.eecs.umich.edu/~jmickens/predictors.pdf

6 Other datasets

These are some availability datasets that are not (yet?) included in this distribution. Many I have not looked at in detail, and they may not be appropriate for this repository. In no particular order:

- Bianca Schroeder, Garth Gibson. A Large-scale Study of Failures in High-performance-computing Systems. Proceedings of the International Conference on Dependable Systems and Networks (DSN2006), Philadelphia, PA, USA, June 25-28, 2006. See also project website.

- D. Long, A. Muir, R. Golding. A longitudinal survey of Internet host reliability. 14th Symposium on Reliable Distributed Systems, 1995.
- J. Chu, K. Labonte, and B. N. Levine. Availability and locality measurements of peer-to-peer file systems. In Proc. of ITCom: Scalability and Traffic Control in IP Networks, July 2002. [Gnutella, Napster]
- S. Sen and J. Wang. Analyzing peer-to-peer trafic across large networks. In Proc. of ACM SIGCOMM Internet Measurement Workshop, Nov. 2002. [FastTrack]
- R. Bhagwan, S. Savage, and G. Voelker. Understanding availability. In Proc. IPTPS, Feb. 2003. [Overnet]
- K. P. Gummadi, R. J. Dunn, S. Saroiu, S. D. Gribble, H. M. Levy, and J. Zahorjan. Measurement, modeling, and analysis of a peer-to-peer file-sharing workload. In Proc. ACM SOSP, Oct. 2003. [Kazaa]
- [5] [Gnutella, Napster]
- RON Project Data: http://nms.csail.mit.edu/ron/data (This has only a few days of availability data.)
- D. Stutzbach and Reza Rejaie. Characterizing Unstructured Overlay Topologies in Modern P2P File-Sharing Systems. In IMC 2005. [Gnutella]
- C. Chambers and W. Feng. Measurement-based Characterization of a Collection of On-line Games. In IMC 2005.
- J. A. Pouwelse, P. Garbacki, D. H. J. Epema, H. J. Sips. The Bittorrent P2P File-sharing System: Measurements and Analysis. In 4th International Workshop on Peer-to-Peer Systems (IPTPS'05), Feb 2005.
- PlanetLab All Sites Ping: http://ping.ececs.uc.edu/ping/

7 Changelog

Version 0.1, April 16, 2006:

- Initial release.

References

[1] Mehmet Bakkaloglu, Jay J. Wylie, Chenxi Wang, and Gregory R. Ganger. On correlated failures in survivable storage systems. Technical Report CMU-CS-02-129, Carnegie Mellon University, May 2002.

[2] William J. Bolosky, John R. Douceur, David Ely, and Marvin Theimer. Feasibility of a serverless distributed file system deployed on an existing set of desktop PCs. In *Proc. SIGMETRICS*, 2000.

[3] Saikat Guha, Neil Daswani, and Ravi Jain. An Experimental Study of the Skype Peer-to-Peer VoIP System. In *Proceedings of The 5th International Workshop on Peer-to-Peer Systems (IPTPS '06)*, Santa Barbara, CA, February 2006.

[4] Jeffrey Pang, James Hendricks, Aditya Akella, Bruce Maggs, Roberto De Prisco, and Srinivasan Seshan. Availability, usage, and deployment characteristics of the domain name system. In *Proc. IMC*, 2004.

[5] Stefan Saroiu, P. Krishna Gummadi, and Steven D. Gribble. A Measurement Study of Peer-to-Peer File Sharing Systems. In *Proc. MMCN*, San Jose, CA, USA, January 2002.

[6] Jeremy Stribling. Planetlab all pairs ping. http://infospect.planet-lab.org/pings.