

Mirroring the ADS Bibliographic Databases

Alberto Accomazzi, Guenther Eichhorn, Michael J. Kurtz, Carolyn S. Grant and Stephen S. Murray

Smithsonian Astrophysical Observatory, 60 Garden Street, Cambridge, MA 02138, USA

Abstract.

During the past year the Astrophysics Data System has set up two mirror sites for the benefit of users around the world with a slow network connection to the main ADS server in Cambridge, MA. In order to clone the ADS abstract and article services on the mirror sites, the structure of the bibliographic databases, query forms and search scripts has been made both site- and platform-independent by creating a set of configuration parameters that define the characteristics of each mirror site and by modifying the database management software to use such parameters. Regular updates to the databases are performed on the main ADS server and then mirrored on the remote sites using a modular set of scripts capable of performing both incremental and full updates. The use of software packages capable of authentication, as well as data compression and encryption, permits secure and fast data transfers over the network, making it possible to run the mirroring procedures in an unsupervised fashion.

1. Introduction

Due to the widespread use of its abstract and article services by astronomers worldwide, the NASA Astrophysics Data System (ADS) has set up two mirror sites in Europe and Asia. The European mirror site is hosted by the Centre De Données Stellaires (CDS) in Strasbourg, while the Asian mirror is hosted by the National Astronomical Observatory of Japan (NAO) in Tokyo.

The creation of the ADS mirrors allows users in different parts of the world to select the most convenient site when using ADS services, making best use of bandwidth available to them. For many users outside the USA this has meant an increase in throughput of orders of magnitude. For instance, Japanese users have seen typical data transfer rates going from 10 bytes/sec to 10K bytes/sec. In addition, the existence of replicas of the ADS services has taken some load off of the main ADS site at the Smithsonian Astrophysical Observatory, allowing the server to respond better to incoming queries.

The cloning of databases on remote sites does however present new challenges to the data providers. First of all, in order to make it possible to replicate a complex database system elsewhere, the database management system and the underlying data sets have to be independent of the local file structure, operating system, hardware architecture, etc. Additionally, networked services which rely

on links with both internal and external Web resources (possibly available on different mirror sites) need to have procedures capable of deciding how the links should be created, possibly giving users the option to review and modify the system's linking strategy. Finally, a reliable and efficient mechanism should be in place to allow unsupervised database updates, especially for those applications involving the publication of time-critical data.

2. System Independence

The database management software and the search engine used for the ADS bibliographic services have been written to be system-independent.

Hardware independence is made possible by writing portable software that can be either compiled under a standard compiler and environment framework (e.g., GNU gcc) or interpreted by a standard language (e.g., Perl5). All the software used by the ADS mirrors is first compiled and tested for the different hardware platforms on the main ADS server, and then the appropriate binary distributions are mirrored to the remote sites.

Operating System independence is achieved by using a standard set of Unix tools which abiding to a well-defined standard (e.g., POSIX.2). Any additional enhancements to the standard Unix system tools are achieved by cloning more advanced software utilities (e.g., GNU shell-utils) and using them when necessary.

File-system independence is made possible by organizing the data files for a specific database under a single directory tree, and creating configuration files with parameters pointing to the location of these top-level directories. Similarly, host name independence is achieved by storing the host names of ADS servers in configuration files.

3. Resolution of Hyperlinks

The strategy used to generate links to networked services external to the ADS which are available on more than one site follows a two-tiered approach. First, a "default" mirror can be specified in a configuration file by the ADS administrator. This configuration file is site-specific, so that appropriate defaults can be chosen for each of the ADS mirror sites depending on their location. Then, ADS users are allowed to override these defaults by using a "Preference Settings" page to have the final say as to which site should be used for each link category (see Figure 1). The use of preferences is implemented using HTTP "cookies" (Kristol & Montulli, 1997). The URLs relative to external links associated with a particular bibliographic references are looked up in a hash table and variable substitution is done if necessary to resolve those URLs containing mirror site variables, as shown in the examples below.

```
1997Icar..126..241S ⇒ $IDEAL$/cgi-bin/links/citation/0019-1035/126/241
⇒ http://www.idealibrary.com/cgi-bin/links/citation/0019-1035/126/241
1997astro.ph..8232H ⇒ $PREPRINTS$/abs/astro-ph/9708232
⇒ http://xxx.lanl.gov/abs/astro-ph/9708232
```

```

# $Id: ads.sites.config,v 1.1 1997/09/08 19:52:16 ads Exp ads $
# Sets possible values for mirror sites used by the ADS
# bibliographic services, with default values for the SAO site.

# ADS article sites
SET_1=ADS Article Mirrors
TAGS_1=ARTICLE
DESC_1=ADS Article Mirrors
ARTICLE_DESC_1=SAO, Cambridge, MA, USA
ARTICLE_DESC_2=NAO, Tokyo, Japan
ARTICLE_L_1=http://adsbit.harvard.edu
ARTICLE_L_2=http://ads.nao.ac.jp

# SIMBAD DB mirrors
SET_2=SIMBAD
DESC_2=SIMBAD Mirrors
TAGS_2=simbad
DESC_2=simbad
SIMBAD_DESC_1=CDS, Strasbourg, France
SIMBAD_DESC_2=SAO, Cambridge, MA, USA
SIMBAD_L_1=http://simbad.u-strasbg.fr/simbo.pl
SIMBAD_L_2=http://simbad.harvard.edu/simbo.pl

# AAS publications (University Chicago Press)
SET_3=AAS Mirrors
TAGS_3=AAS,AAP,AAS_AJ
DESC_3=AAS
AAS_DESC_1=University of Chicago Press, USA
AAS_DESC_2=CDS, Strasbourg, France
AAS_APJ_1=http://www.journals.uchicago.edu/ApJ/cgi-bin/resolve
AAS_APJ_2=http://cdsas.u-strasbg.fr/2001/ApJ/cgi-bin/resolve
AAS_APJ_3=http://www.journals.uchicago.edu/AJ/cgi-bin/resolve
AAS_AJ_2=http://cdsas.u-strasbg.fr/2001/AJ/cgi-bin/resolve

# Preprint Servers
SET_4=PREPRINTS
DESC_4=PREPRINTS Mirrors
TAGS_4=preprints
DESC_4=preprints
PREPRINTS_DESC_1=LAM, Los Alamos, NM, USA
PREPRINTS_DESC_2=CDEx, Trier, Germany
PREPRINTS_DESC_3=CDEx, Trier, Germany
PREPRINTS_DESC_4=Yukawa, Kyoto, Japan
PREPRINTS_DESC_5=http://xxx.yukawa.kyoto-u.ac.jp

# IDEAL library (Academic Press publications)
SET_6=IDEAL
DESC_6=ideal Library Mirrors
TAGS_6=IDEAL
DESC_6=ideal
IDEAL_DESC_1=ideal
IDEAL_DESC_2=ideal
IDEAL_L_1=http://www.idealibrary.com
IDEAL_L_2=http://www.europe.idealibrary.com

```

Figure 1. Left: the Preference Setting form allows users to select which mirror sites should be used when following links on ADS pages. Right: mirror sites configuration file for the ADS server at SAO.

1997ApJ...486L..75F ⇒ \$AAS_APJ\$?1997ApJ...486L..75FCHK
 ⇒ <http://www.journals.uchicago.edu/ApJ/cgi-bin/resolve?1997ApJ...486L..75FCHK>

While more sophisticated ways to create dynamic links are being used by other institutions (Fernique et al. 1998), there is currently no reliable way to automatically choose the “best” mirror site for a particular user. By saving these settings in a user preference database indexed on the cookie ID, users only need to define their preferences once and our interface will retrieve and use the appropriate settings as necessary.

4. Mirroring Software

The software used to perform the actual mirroring of the databases consists of a main program running on the ADS master site initiating the mirroring procedure, and a number of scripts, run on the mirror sites, which perform the transfer of files and software necessary to update the database. The main program, which can be run either from the command line or as a CGI script, is an Expect/Tcl script that performs a login on the mirror site to be updated, sets up the environment by evaluating the mirror site and master site’s configuration files, and then initiates the updating process.

The updating procedures are specialized scripts which check and update different parts of the database and database management software (including the procedures themselves). The actual updating of the database files is done by using a public domain implementation of the rsync algorithm (Tridgell &

Mackerras, 1996), with local modifications. The advantages of using rsync to update data files rather than performing complete transfers are:

Incremental updates: rsync updates individual files by scanning their contents and copying across the network only those parts of the files that have changed. Since only a small fraction of the data files actually changes during our updates (usually less than 5% of them), this has proved to be a great advantage.

Data integrity: should the updating procedure be interrupted by a network error or human intervention, the update can be resumed at a later time and rsync will pick up transferring data from where it had left off. File integrity is checked by comparing file attributes and via a 128-bit MD4 checksum.

Data compression: rsync supports internal compression of the data stream by use of the zlib library (also used by GNU gzip).

Encryption: rsync can be used in conjunction with the Secure Shell package (Ylonen 1997) to transfer the data for added security. Unfortunately, transfer of encrypted data could not be performed at this point due to foreign government restrictions and regulations on the use of encryption technology.

5. Conclusions

The approach we followed in the implementation of automated mirroring procedures for the ADS bibliographic services has proved to be very effective and flexible. The use of the rsync algorithm makes it practical to update portions of the database and have only such portions automatically transferred to the mirror sites, without requiring us to keep track of what individual files have been modified. Because of the reduced amount of data that needs to be transferred over the network, we typically achieve speed gains from 1 to 2 orders of magnitude, which makes the updating process feasible despite poor network connections. We plan to improve the reliability of the individual transfers (which occasionally are interrupted by temporary network dropouts) by using sensible time-outs and adding appropriate error handlers in the main transfer procedure.

As a result of the proliferation of mirror sites, we have provided a user-friendly interface which allows our users to conveniently select the best possible mirror site given their local network topology. This model, currently based on HTTP cookies, can be easily adapted by other data providers for the benefit of the user. An issue which still needs to be resolved concerns providing a fallback mechanism allowing users to retrieve a particular document from a backup mirror site should the default site not be available. It is possible that new developments in the area of URN definition and management will help us to find a solution to this problem.

Acknowledgments. This work is funded by the NASA Astrophysics Program under grant NCCW-0024.

References

Fernique, P., Ochsenbein, & F., Wenger, M. 1998, this volume

- Kristol, D., & Montulli, L. 1997, HTTP State Management Mechanism, RFC2109, Internet Official Protocol Standards, Network Working Group.
- Tridgell, A., & Mackerras, P. 1996, The rsync algorithm, Joint Computer Science Technical Report Series TR-CS-96-05, Australian National University.
- Ylonen, T. 1997, SSH (Secure Shell) Remote Login Program, Helsinki University of Technology, Finland.