IBM Storage Tank™

A Distributed Storage System

D. A. Pease, R. M. Rees, W. C. Hineman, D. L. Plantenberg, R. A. Becker-Szendy,
R. Ananthanarayanan, M. Sivan-Zimet, C. J. Sullivan:
IBM Almaden Research Center

R. C. Burns: Johns Hopkins University

D. D. E. Long: University of California, Santa Cruz

January 23, 2002

Introduction

IBM Storage Tank™ is a SAN-based distributed object storage system for use in heterogeneous environments. It provides performance comparable to that of file systems built on bus-attached, high-performance storage, as well as advanced storage and data management functions. It is designed to be highly available and scalable. The Storage Tank project has been underway at IBM's Almaden Research Center for several years.

Storage Tank is designed to work with any Storage Area Network architecture, as well as with any SAN storage hardware. (It currently runs on both Fibre Channel and iSCSI SANs.) It is also designed to be portable to essentially any host system architecture.

This paper provides a high-level overview of Storage Tank's design and features.

IBM Storage Tank Architecture

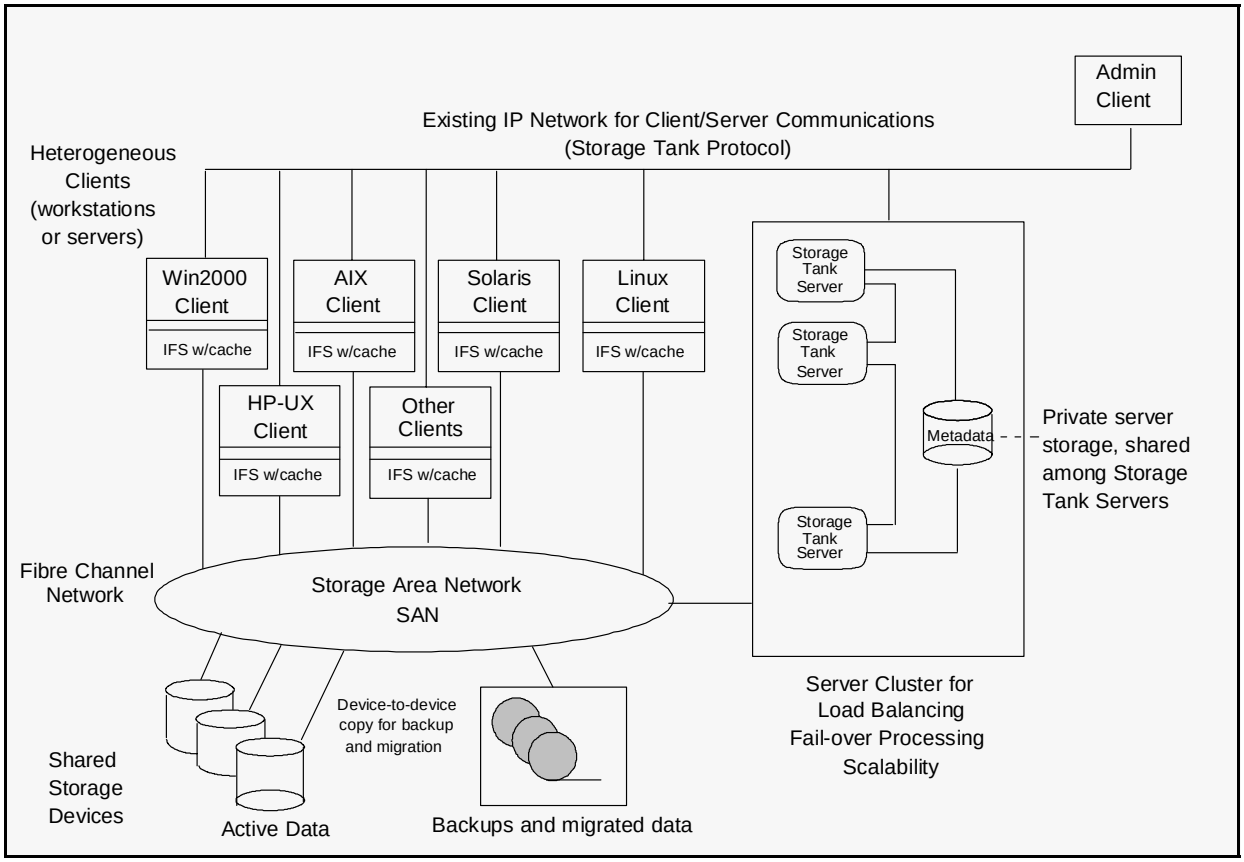

Figure 1 illustrates the basic IBM Storage Tank architecture.

Figure 1. IBM Storage Tank Architecture

Figure 1 shows that Storage Tank clients and the administrative client communicate with Storage Tank servers over an enterprise's existing IP network using the Storage Tank protocol. It

also shows that Storage Tank clients, servers, and storage devices are all connected to a high-speed Storage Area Network (SAN).

The Storage Tank administrative client serves as the administrative control point. An administrator can perform almost all administrative tasks online with no service interruption to clients.

An installable file system (IFS) is installed on each Storage Tank client. An IFS directs requests for metadata and locks to a Storage Tank server and sends requests for data to storage devices on the SAN. Storage Tank clients can access data directly from any storage device attached to the SAN.

Storage Tank clients aggressively cache file data, as well as metadata and locks that they obtain from a Storage Tank server, in memory.

An enterprise can use a single Storage Tank server, a cluster of servers, or multiple clusters of servers. Clustered servers provide load balancing, fail-over processing, and increased scalability. The Storage Tank servers in a cluster can be interconnected on their own high-speed network or on the same IP network that they use to communicate with Storage Tank clients. The private server storage (which contains the metadata managed by a cluster of Storage Tank servers) can be attached to a private network connected only to the cluster of servers, or it can be attached to the general-purpose Storage Tank SAN.

IBM Storage Tank Servers

IBM Storage Tank server runs on many different software platforms, such as AIX, Windows, and Linux. The server is a portable, user-level, C++ application, and can be ported to additional operating systems as needed.

Support for multiple operating systems allows an administrator to choose from a wide range of server machines on which to install the server programs. This allows the administrator to provide the appropriate level of performance for an enterprise. For example, an administrator can choose to install the server programs on computers built on Intel processors running Linux for cost-effective scalability or on IBM SP2 supercomputers running AIX for high-end scalability.

An enterprise can use a single server, a cluster of servers, or multiple clusters of servers. Using IBM Storage Tank servers in a cluster configuration has the following benefits:

- Load balancing — The workload and data structures for the IBM Storage Tank distributed storage system are partitioned and allotted to the servers in the cluster. This is a continuous process that keeps the cluster workload balanced at all times.

- Fail-over processing — IBM Storage Tank servers are designed to redistribute the load evenly among servers in the cluster if one of the servers should fail without interruption of server to the clients.

- Scalability — An administrator can add more servers to a cluster or add more server clusters to the SAN to serve more data and more clients. Note that multiple server clusters cooperate to maintain the IBM Storage Tank uniform global namespace.

The IBM Storage Tank servers in a specific cluster must all be of the same type. However, an installation can have multiple clusters of different types. For example, an enterprise might have one server cluster in which all the servers run AIX, and another server cluster in which all the servers run Linux.

IBM Storage Tank servers provide the following types of services:

- Metadata services
- Administrative services
- Storage and data management services

Metadata Services

An IBM Storage Tank server is designed to perform these metadata services:

- Manage allocation and placement of data in storage pools on storage devices
- Perform metadata writes
- Serve file system metadata to clients
- Grant file and data locks to clients
- Detect client failures and perform client recovery

Administrative Services

As part of a server cluster, there can be one or more administrative clients that an administrator uses to control the IBM Storage Tank servers. An administrative client is connected to the Storage Tank servers via an IP network. To perform administrative tasks, an administrator can choose to use either a graphical user interface or a command line interface.

Storage and Data Management Services

Based on storage and data management policies set up by an administrator, IBM Storage Tank is designed to automatically perform a variety of storage and data management services. These services provide the enterprise with automated management of all aspects of the life-cycle of their data. The system performs these services across the SAN with no client involvement.

IBM Storage Tank Clients

One of the goals of IBM Storage Tank is to enable full, transparent data sharing of files among heterogeneous clients, such as those running Windows 2000, AIX, Solaris, Linux, and HP-UX, and other operating systems.

All IBM Storage Tank clients can access the same data using Storage Tank's uniform global namespace. A uniform global namespace provides the capabilities for all clients to have a consistent view of the Storage Tank name tree.

File Server Support

A file or application server, such as an NFS, CIFS, or HTTP server, can also be an IBM Storage Tank client. For these clients, Storage Tank provides the following:

- Scalability — A file server can access all of the files in the IBM Storage Tank distributed storage system. Storage Tank is highly scalable and, therefore, can provide a file server access to a vast amount of data.

- Reliability and fail-over processing — Because many servers can be IBM Storage Tank clients and can serve the same data, requests from clients of a failed server can be transferred to another server using any technique supported by the file server, such as high availability cluster multi-processing (HACMP) or IP address stealing.

Installable File Systems

IBM Storage Tank requires an installable file system (IFS) on each IBM Storage Tank client. The IFS software, which is light-weight and easy to install, can be made available through a Web interface to an IBM Storage Tank server.

An IFS is a subsystem of the client's file system. The IFS is designed to direct all metadata operations to an IBM Storage Tank server, and direct all data operations to storage devices attached to a high-speed network. It makes the metadata that is visible to the client's operating system and applications look identical to metadata read from a native, locally-attached file system, so that no changes to existing applications are required.

Note that special purpose applications, such as digital libraries and databases, can also access data from the IBM Storage Tank distributed storage system by using an application programming interface (API) to the IBM Storage Tank protocol. Because these applications do not use the file system to access their data, the clients on which they run do not need to have the Storage Tank IFS installed.

IBM Storage Tank Protocol

The IBM Storage Tank protocol is a locking and data consistency model that allows the IBM Storage Tank distributed storage system to look and behave like a local file system. The objective of the Storage Tank protocol is to provide strong data consistency between clients and servers in a distributed environment.

The IBM Storage Tank protocol provides locks that enable file sharing among IBM Storage Tank clients, or, when necessary, provides locks that allow clients to have exclusive access to files. These locks are granted to clients by the Storage Tank server. The Storage Tank protocol guarantees that when a client reads data from a particular file, it always reads the last data written to that file from anywhere in the distributed storage system.

To open a file in the IBM Storage Tank distributed storage system, a client does the following:

1. Contacts a IBM Storage Tank server to obtain metadata and locks.

   Metadata supplies the client with information about a file, such as its attributes and location on storage device(s).

   Locks supply the client with the privileges it needs to open a file and read or write data. The IBM Storage Tank locking scheme is designed to ensure strong data consistency.

2. Accesses the data for the file directly from a shared storage device attached to a high-performance SAN.

More Information

See our web site at *http://www.almaden.ibm.com/cs/storagesystems/stortank*.

IBM Storage Tank Technical Papers

The following papers are available at *http://www.almaden.ibm.com/cs/storagesystems/stortank*.

R. C. Burns, R. M. Rees, and D. D. E. Long
Efficiently Distributing Data in a Web Server Farm
To appear in: *IEEE Internet Computing*, 2001.

R. C. Burns, R. M. Rees, and D. D. E. Long
An Analytical Study of Opportunistic Lease Renewal
In *Proceedings of the 16th International Conference on Distributed Computing Systems* (ICDCS), IEEE, 2001.

R. C. Burns and W. C. Hineman
A Bit-Parallel Search Algorithm for Allocating Free Space
To appear in: *Proceedings of the 9th International Symposium on Modeling, Analysis, and Simulation in Computer and Telecommunication Systems* (MASCOTS), IEEE, 2001.

R. C. Burns, R. M. Rees, L. J. Stockmeyer, and D. D. E. Long
Scalable Session Locking for a Distributed File System
In *Cluster Computing Journal*, Volume 4, Number 4, Dec. 2001.

R. C. Burns, R. M. Rees, and D. D. E. Long
Safe Caching in a Distributed File System for Network Attached Storage
In *Proceedings of the International Parallel and Distributed Processing Symposium* (IPDPS), IEEE, 2000.

R. C. Burns, R. M. Rees, and D. D. E. Long
Semi-Preemptible Locks for a Distributed File System
In *Proceedings of the 2000 International Performance Computing and Communication Conference* (IPCCC), IEEE, 2000.

R. C. Burns, R. M. Rees, and D. D. E. Long
Consistency and Locking for Distributing Updates to Web Servers Using a File System
In *Performance Evaluation Review*, 28(2), ACM, 2000.

R. C. Burns
Data Management in a Distributed File System for Storage Area Networks
A dissertation in completion of the Doctor of Philosophy degree,
Department of Computer Science, University of California, Santa Cruz, March 2000.

Special Notices

IBM Corporation
Storage Systems Group
5600 Cottle Road
San Jose, CA 95193

Produced in the United States of America

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

AIX
IBM ®
IBM Storage Tank
SP2 ®

The following terms are trademarks of other companies:

Microsoft, Windows, Windows 2000, and the Windows logo are trademarks of Microsoft Corporation in the United States and/or other countries.

Solaris is a trademark of Sun Microsystems, Inc. in the United States and/or other countries.

HP-UX is a trademark of the Hewlett Packard Corporation in the United States and/or other countries.

UNIX is a registered trademark in the United States and/or other countries licensed exclusively through X/Open Company Limited.

Other company, product, and service names may be trademarks or service marks of others.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785.

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PAPER “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO , THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes may be made periodically to the information herein; these changes may be incorporated in subsequent versions of the paper. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this paper at any time without notice.