

# Jiro™ Management Applications

Gary Mueller  
Storage Technology Corporation  
[gary\\_mueller@stortek.com](mailto:gary_mueller@stortek.com)  
303-673-6498

## Abstract

Jiro™ is a Sun product that provides the infrastructure and plumbing for building distributed applications. Distributed services such as event handling, logging, transaction management, security, as well as basic communication services for distributed objects are included. Jiro is written in Java and uses the Jini™ name and lookup services.

Jiro is a product implementation of the Federated Management Architecture. The Federated Management Architecture is a three tier architecture consisting of clients, management services and managed resources.

This paper discusses the basic architecture of Jiro as well as two Jiro prototypes based on the StorageTek ACSLS™ tape library software and the StorageTek 9840 tape drive.

## What is Jiro™?

Jiro is a distributed management infrastructure, which provides basic distributed services such as event handling, logging, and scheduling for distributed applications. In this sense, Jiro is similar to other distributed technologies such as DCOM or CORBA. Unlike CORBA, which is defined in a language independent manner using the Interface Definition Language (IDL), Jiro is written in Java and uses the Jini™ lookup service also written in Java.

## Why Jiro™ ?

Jiro provides the necessary infrastructure to build distributed management applications for Storage Area Networks (SAN). Since many companies plan to deliver products in this area, Jiro could provide the necessary infrastructure for SAN applications.

Sun is working very hard to make Jiro ubiquitous and would like Jiro to become the defacto standard for distributed applications. To that end, Sun is giving Jiro away free. There are no licensing or usage fees. The only condition

on the use of Jiro is the Sun Community Source License (SCSL) which requires that modifications to Jiro be made freely available. Sun is leveraging Jiro as a requirement to partner with Sun.

## Jiro Concepts

Jiro organizes managed resources into management domains. Management services such as backup, archiving or tape library storage are available within the management domains as well as base services such as event handling, scheduling, which provide the distributed application infrastructure.

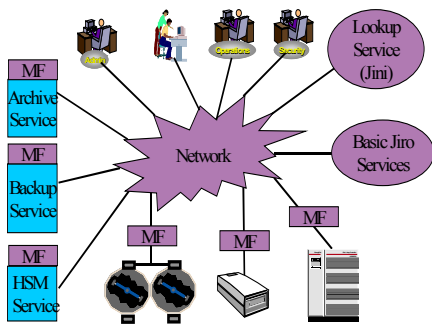
The management domains are logical groupings, which span some segment of a network. Beside the managed resources and the base services such as event handling, logging and scheduling, dynamic or management services reside in the management domain.

A management service can be a storage device, which provides storage at predetermined cost and performance or a hierarchical management service (HSM) which dynamically balances available storage according to a set of predetermined policies or rules.

In order to provide plug and play management services, Jiro provides an environment called a management station or station which hosts the management service. The management service is accessed through a management interface or Application Programmatic Interface (API) called a management façade.

Through the implementation of management facades, Jiro integrates storage devices as well as storage management services into a distributed architecture such as a Storage Area Network (SAN).

To request management services using Jiro, the management service must register its service and type with Jini. Jini is a directory or lookup service which tracks the location of management services within a management domain. Once the service is deployed and installed in a management station, the service is now registered and available to clients on the network.



**Figure 1. Basic Jiro Architecture**

A network client such as a management GUI contacts Jini with a request for a management service of a particular type and capability. As part of the service registration, a template is sent to Jini, which describes the type of service. Jini takes the client request and looks for a matching service. If one is found, the client is notified of the location of the service. The client can then connect to the service and begin sending requests to the service.

Jiro provides the infrastructure to communicate between the management service and the client. Java Virtual Machines (JVM) communicate using the Java Remote Method Invocation (RMI). RMI is conceptually the equivalent of remote procedure calls for objects. A client requests a method on what appears to the client to be a local object running on the local JVM. The local object is in reality a surrogate for the real object located on a remote JVM. Jiro provides the *plumbing* to facilitate the communication between the client and the service.

All Jiro components are host based and do not run directly in a storage device.

### StorageTek® 9840B Jiro™ Prototype

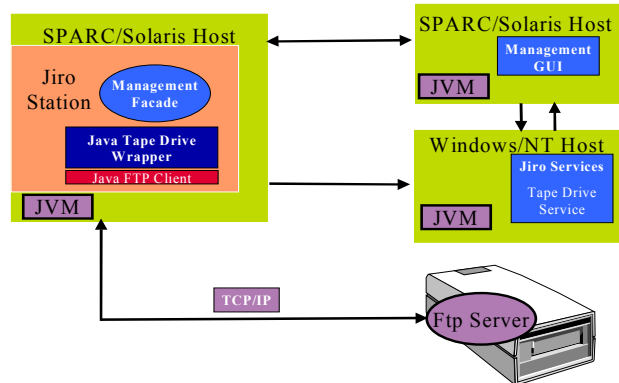
As shown in Figure 2, a prototype using Jiro has been built for the StorageTek 9840B tape drive. The 9840B tape drive has a TCP/IP connection. Included in the microcode of the 9840B is an FTP server. Many of the functions on the front panel of the 9840B are provided through the 9840B FTP interface.

The basic design of the FTP interface is file based. In order to perform a function such as unloading a tape drive, a file named *unload* is

sent over the TCP/IP interface to the FTP Server. The FTP Server then requests a tape unload from the tape microcode.

In order to integrate the 9840B into Jiro, a management façade was designed that uses the FTP interface. The management façade consists of two pieces of software, the Java FTP Client and the Java Tape Drive Wrapper. The Java FTP Client is a third party Java Bean. The Java Tape Drive Wrapper is internally developed software.

The Java FTP client sends and retrieves files over the TCP/IP channel. The Java Tape Drive Wrapper wraps the FTP client and implements the management façade for the 9840B tape drive. A management GUI was written which uses the 9840B management façade to remotely manage the 9840B tape drive.



**Figure 2. 9840B Jiro Prototype**

### StorageTek® ACSLS Jiro™ Proof of Concept

Another prototype shown in Figure 3 was written using Jiro and is based on the ACSLS™ tape library software. The ACSLS tape library software provides a C based programmatic interface to StorageTek tape libraries. This interface, the ACSAPI, is a network based interface which itself provides remote access to the tape libraries.

A management façade for ACSLS was written using the Java Native Interface (JNI). The JNI connects legacy applications written in C to Java objects. Wrapping the ACSAPI with the JNI and creating a set of objects to access ACSLS services is the basis of the management façade for the ACSLS. Again a management GUI was written to access ACSLS services through the management façade.

The ACSLS prototype was written as part of a proof of concept project sponsored by Sun Microsystems to demonstrate the viability of Jiro. This prototype was presented at the first Jiro developers forum and is currently running at the Sun Dot Com marketing and customer center.

The ACSLS prototype uses a 9738 StorageTek tape library with 9840 Tape Drive and runs ACSLS Release 5.3.2 on an Ultra 10 with Solaris 2.6.

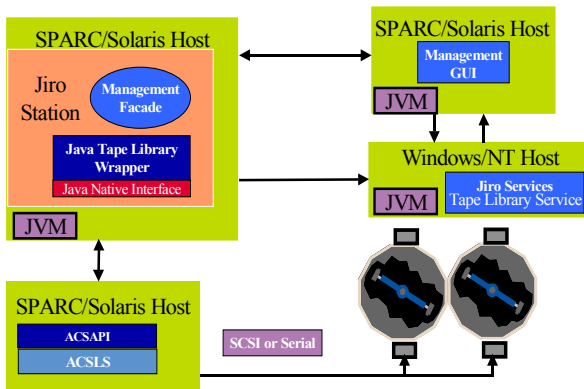


Figure 3. ACSLS Jiro Prototype

## Conclusions

Besides the 9840B tape drive and ACSLS Jiro prototype, StorageTek is evaluating Jiro for use in other projects.

Although the first release of Jiro was recently released, it provides much of the infrastructure for developing distributed management applications.

For more information on Jiro and to get the latest free releases go the Jiro web site,

<http://www.jiro.com>.

## References

1. <http://www.jiro.com>

