

WebNFS SDK for Java

Brent Callaghan
Sun Microsystems, Inc
brent@eng.sun.com

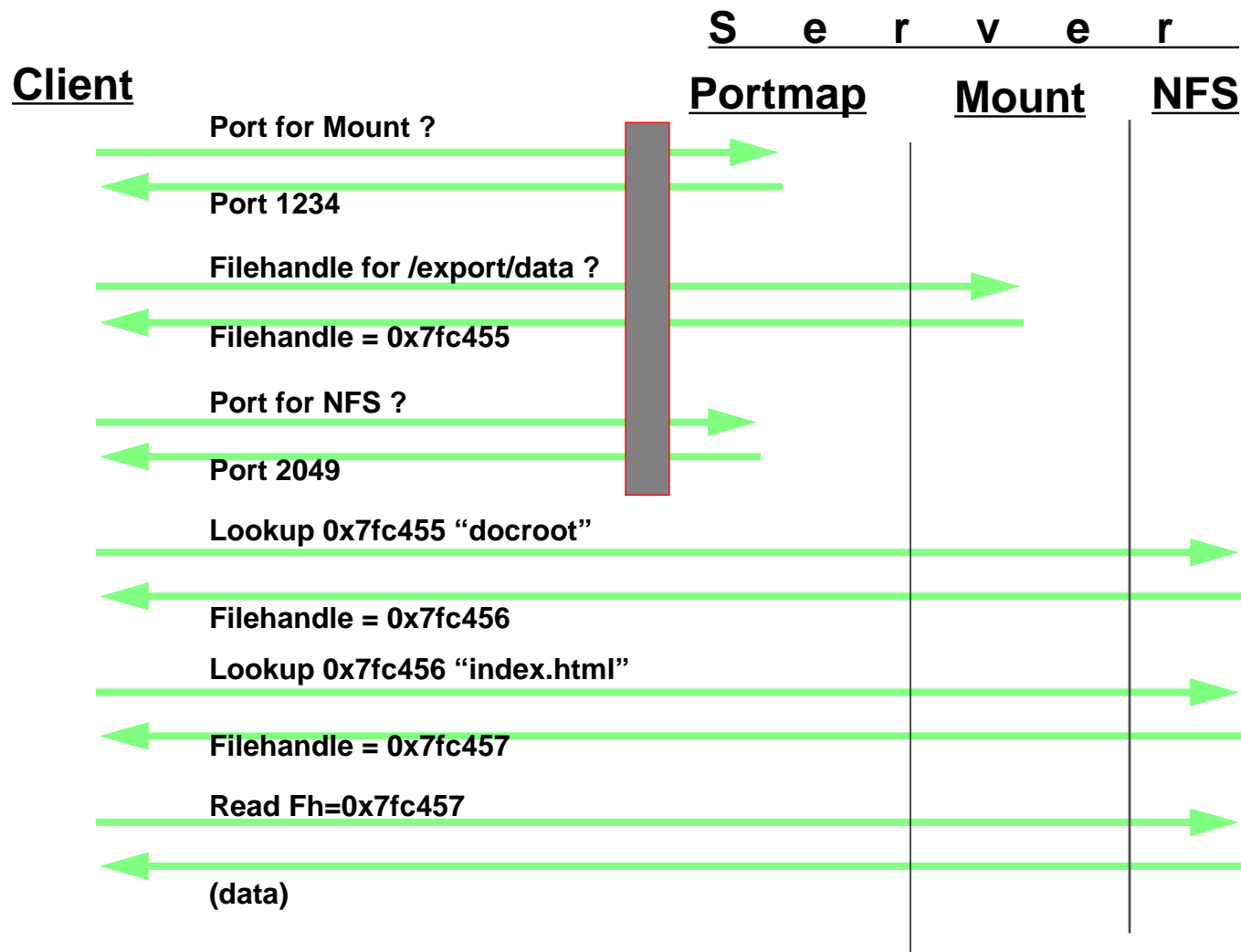
Internet NFS

- **Internet NFS works today!**
 - `gatekeeper.dec.com/archive`
 - `wuarchive.wustl.edu/archive`
 - `playground.sun.com/export/home/ftp`
- **NFS over TCP provides good Internet performance**
- **Convenient access**
 - **Use normal file browser, “ls” or “dir”**
 - **Access in-place. Don’t need to transfer file first (acoread)**

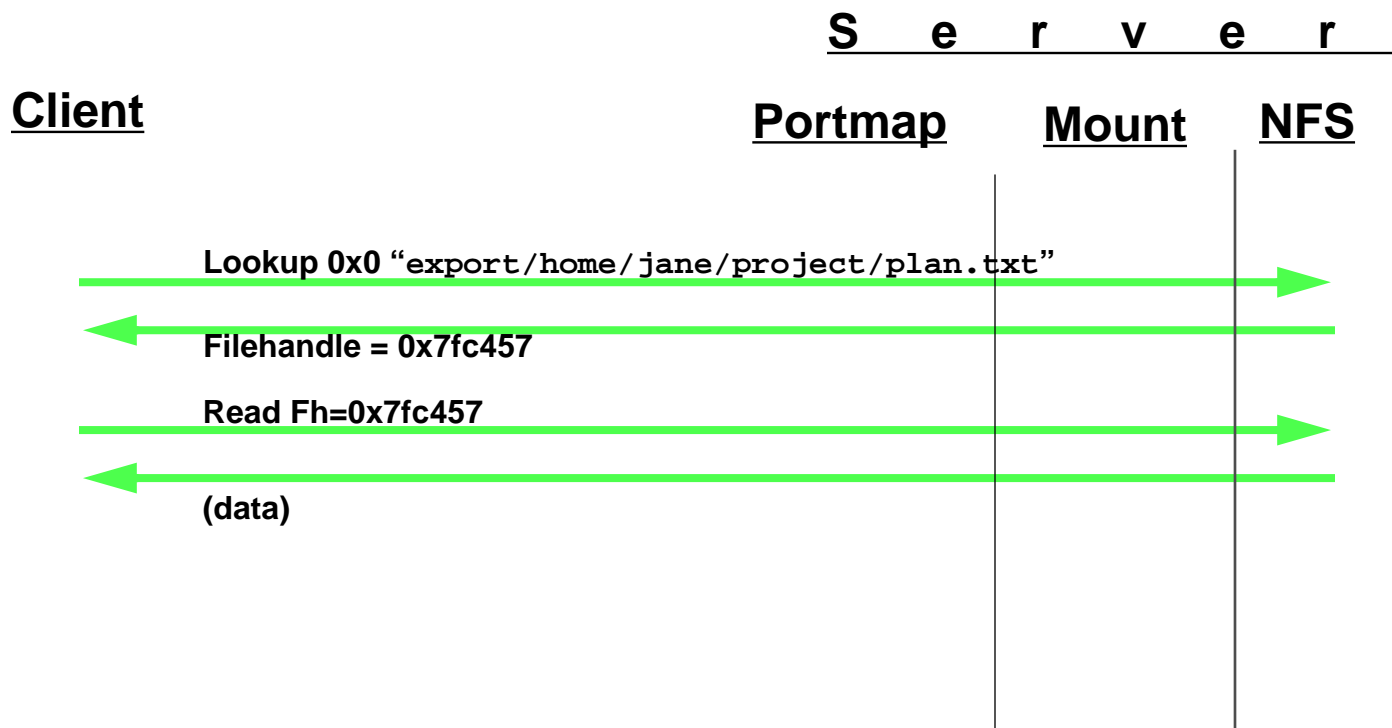
HOWEVER!

- **Firewalls are a barrier**

Access without WebNFS



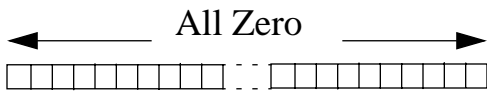
Access with WebNFS



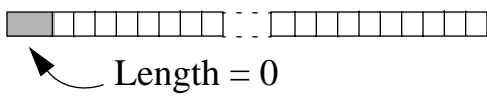
Public Filehandle

- **It's nothing really**

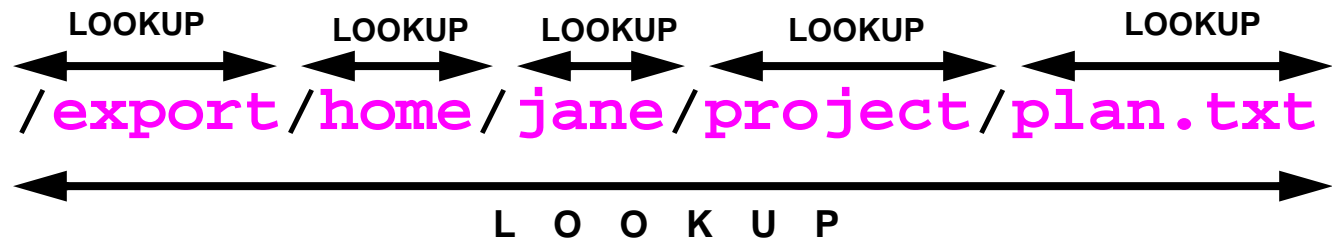
Version 2



Version 3

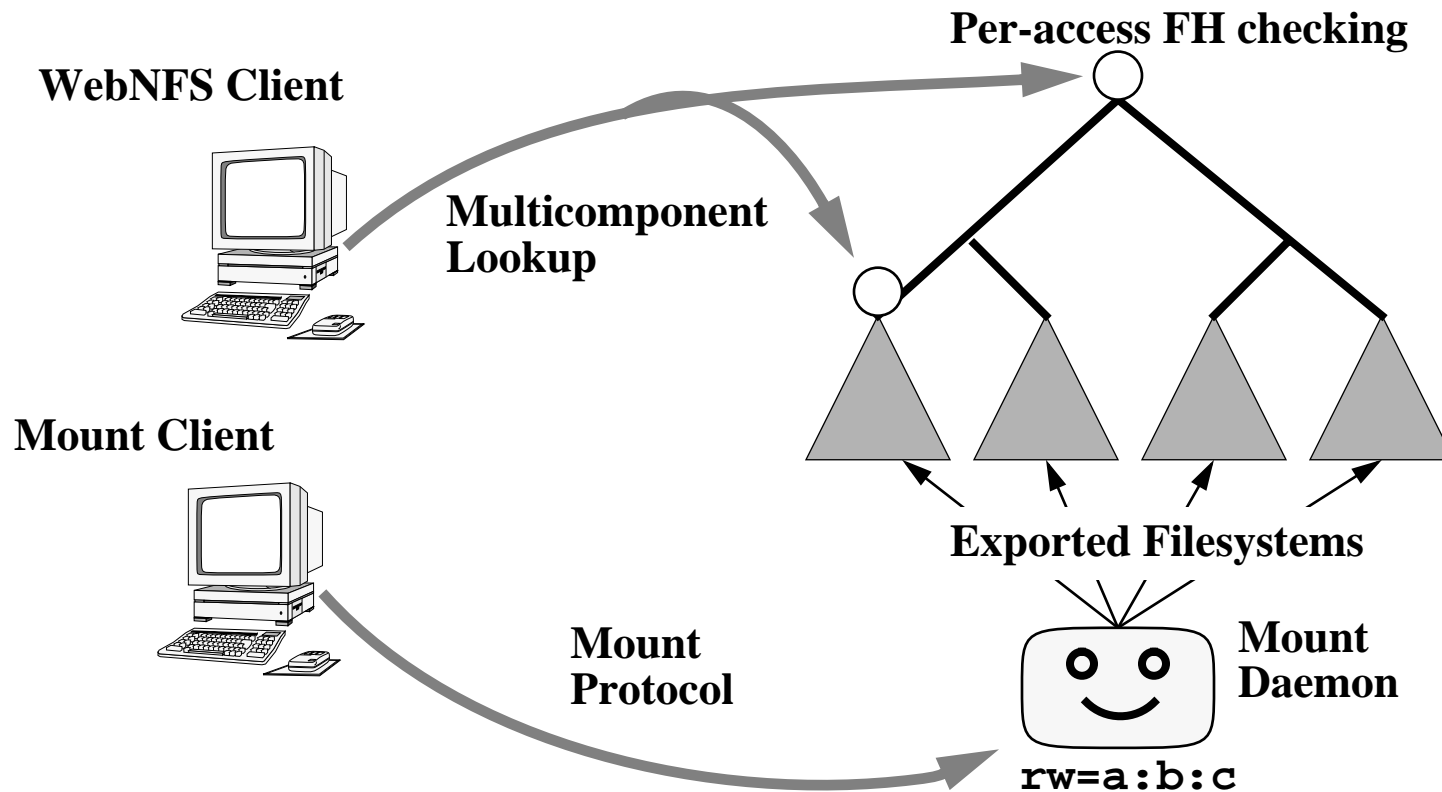


- **Multi-component Lookup**



Location of Public Filehandle

- WebNFS clients skirt the mount daemon
- Mount daemon access checking ineffective



NFS URL

- **NFS URL Scheme in RFC 2224**
- **Syntax: `nfs://server:port/pathname`**
- **Names any exported file on any NFS server anywhere**
- **Platform independent, slashes in the same direction even on Microsoft OS's**
- **Java applications: “*write once, run anywhere*”**
- **URL's are location dependent, however IETF has in development:**
 - **Uniform Resource Names (RFC 1737)**
 - **Service Location Protocol**

Problem: Java File Access

- **No 100% Pure Java applications have remote file access**
- **Network computers have no disks**
- **Can use remote file access of underlying OS**
 - `/net`
 - `\\server\path`
- **Need a consistent name across all platforms: URLs!**
 - `nfs://server/path`
 - `file:///A|/Programs`

SDK API

- **Mirrors the java.io package classes**
 - **XFileInputStream**
 - **XFileOutputStream**
 - **XRandomAccessFile**
 - :
- **Supports URL naming**
 - new **XFile("nfs://server/path")**
- **Plug-in filesystem protocols**
- **Available for download from**
<http://www.sun.com/webnfs>

SDK API Overview

Java Application

SDK API

nfs

cifs

file

native

NFS Client in Java

- **All naming with NFS URL: `nfs://server/path`**
- **Connect to NFS v2 or v3 servers**
- **Uses TCP or UDP connection to server**
- **Uses WebNFS for fast, firewall-friendly connections**
- **Supports MOUNT protocol for non-WebNFS servers**
- **Caches file data and metadata**
- **Automatically follows symbolic links**
- **Efficient read-ahead and write-behind with Java threads**
- **Can read/write create/delete files and directories on any NFS server**

Java NFS Security

- **Default: Unix creds with uid/gid “nobody”**
- **Can use of PCNFSD to map login/passwd to uid/gid**
- **Strong Security**
 - **Will integrate RPCSEC_GSS security framework that will support Kerberos v5 and public-key schemes**
- **WebNFS access => No security negotiation**

Java NFS Performance

- **“Java 20 to 40 times slower than C” - not necessarily!**
- **Reading files with read-ahead (10 MB file on 10Mb/s Ethernet)**
 - **Solaris FTP client:** 1,100 KB/sec
 - **Java NFS client:** **1,005 KB/sec**
 - **Solaris cp command:** 970 KB/sec
- **Writing files with write-behind**
 - **Solaris FTP client:** 1,100 KB/sec
 - **Solaris cp command:** 1,070 KB/sec
 - **Java NFS client:** **1,050 KB/sec**
- **How fast can it go ? Need to do loopback test**

Questions ?