भारतीय प्रौद्योगिकी संस्थान गुवाहाटी
INDIAN INSTITUTE OF TECHNOLOGY GUWAHATI

CS 343 OPERATING SYSTEMS

File Management in Solaris

AB Satyaprakash
180123062
Mathematics and Computing

Presentation Outline

What will we learn today?

— Introduction to Solaris
— Overview of File Management
— Types of File Systems
  - Disk-based
  - Network-based
  - Virtual
— File Handling
— Mounting and Unmounting
— Security and Sharing
— References

Introduction to Solaris

- **Solaris** is a **proprietary** **Unix** operating system originally developed by Sun Microsystems. It superseded the company's earlier SunOS in 1993. In 2010, after the Sun acquisition by Oracle, it was renamed **Oracle Solaris**.

- Solaris is known for its scalability, especially on **SPARC** systems, x86-64 workstations and servers from Oracle and other vendors. It also supports Oracle Database and Java applications.

Overview of File Management

- **File management** describes the fundamental methods for storing, naming, accessing, mounting, sharing and handling files.
- Why is file management necessary?
    - — Efficient and organised data storage
    - — Data protection and recovery

- A **file system** is used for file maintenance (or management) operations. It is is a type of software that manages data files in a computer system.
- **ZFS**, **NTFS**, **FAT32**, **ext4**, **exFAT** are some commonly known file management systems, each with unique features.

File Sytems in Solaris

- The Oracle Solaris OS uses the **virtual file system (VFS)** architecture, which provides a standard interface for different file system types.
- The **VFS** architecture enables the kernel to handle basic operations, such as reading, writing, and listing files. The **VFS** architecture also makes it easier to add new file systems.

- Solaris supports a wide variety of system types to handle most storage media (CDs, DVDs, Hard Drives, floppy disks based storage) and network based file system protocols.
- Solaris also uses the system to implement various system interface features, and to export some **kernel** information as files visible to the user.

Types of File Systems

- The Solaris OS supports three types of file systems:
  - **Disk-based file systems**
  - **Network-based file systems**
  - **Virtual file systems**

- The classification is done on the basis of the use case, that is based on whether the data is stored on a disk, a server, or is concerned with specific kernel related information. We will see each in more detail in the upcoming slides.

Disk-based file systems

- Disk-based file systems are stored on **physical media** such as hard disks and DVDs.
- Disk-based file systems can be written in different formats. The available formats are
  — **ZFS (Zetabyte File System)**
  — **UFS (UNIX File System)**
  — **HSFS (High Sierra File System)**
  — **PCFS (PC File System)**
  — **UDFS (Universal Disk File System)**
- Each type of disk-based file system is customarily associated with a particular media device, as follows:
  - **ZFS** or **UFS** with **hard disk**
  - **HSFS** with **CD-ROM**
  - **PCFS** with **USB** diskette
  - **UDFS** with **DVD**

Disk-based file system	Format Description
ZFS	ZFS is the default disk-based and root file system in the Oracle Solaris 11 release. For more information, see the Oracle Solaris Administration: ZFS File Systems.
UFS	Legacy UNIX file system (based on the BSD Fat Fast File system that was provided in the 4.3 Tahoe release).
HSFS	High Sierra, Rock Ridge, and ISO 9660 file system. High Sierra is the first CD-ROM file system. ISO 9660 is the official standard version of the High Sierra file system. The HSFS file system is used on CD-ROMs, and is a read-only file system. Oracle Solaris HSFS supports Rock Ridge extensions to ISO 9660. When present on a CD-ROM, these extensions provide all file system features and file types, except for writability and hard links.
PCFS	PC file system, which allows read- and write- access to data and programs on DOS-formatted disks that are written for DOS-based personal computers.
UDFS	The Universal Disk Format (UDFS) file system, the industry-standard format for storing information on the optical media technology called DVD (Digital Versatile Disc or Digital Video Disc).

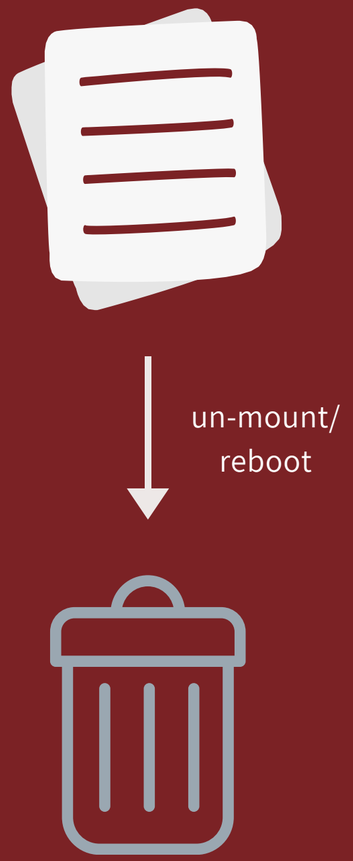
Network-based file systems

- Network-based file systems can be **accessed from the network**. Typically, network-based file systems reside on one system, typically a **server**, and are accessed by other systems across the network.

- With the **NFS service**, you can provide distributed resources (files or directories) by sharing them from a server and mounting them on individual clients.
- With the **Oracle SMB service**, you can provide distributed resources (files or directories) to Windows and Mac OS systems by sharing them from a server and mounting them on individual clients.

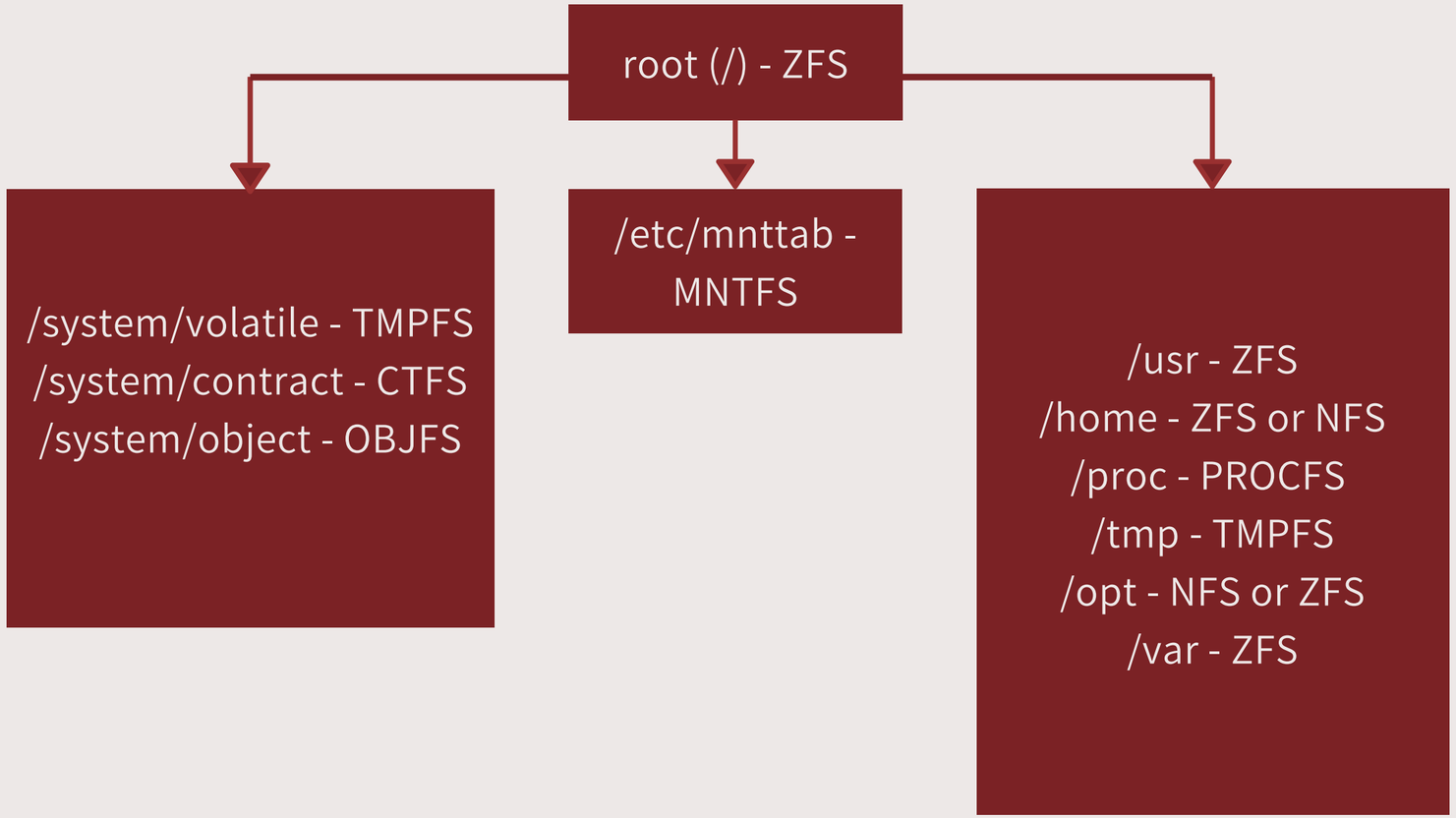
Virtual file systems

- Virtual file systems are memory-based file systems that provide access to special **kernel information** and facilities.
- Most virtual file systems do not use file system disk space. Also, some virtual file systems, such as the **temporary file system** (**TMPFS**), use the **swap space** on a disk.
- Other examples include the **loopback file system**, **process file system, swap file system,** etc.



ZFS: The default file system

- **Solaris ZFS**, a new file system provides simple administration transactional semantics, end-to-end data integrity and immense scalability.
- **Features in OS Lab 4** - Data **Deuplication**, Data **Compression**
- The ZFS file system is **hierarchical**, starting with the **root directory** (/) and continuing downwards through a number of directories..



File Handling

- The **fstyp** command is used to determine the file system type of a disk

```
$ fstyp /dev/rdsk/c0t0d0s0
  zfs
```

- The **cp** command is used to copy a file or a directory. There are 2 options:
  - i : for showing confirmation message before overwriting
  - r : for copying directories recursively

```
$ cp -i  <file-name> <new-location>
$ cp -r <directory-name> <new-location>
```

- The **mv** command is used to rename a file or move it.

```
$ mv <source> <target>
```

- The **touch** command is used to create an empty file.

```
$ touch <file-name>
```


- The **mkdir** command is used to create directories. We can create the parent and child directories simulataneously using the **-p** option.

```
$ mkdir directory(ies)
```

- The **rm** command is used to remove files, while **rmdir** is used to remove **empty directories**. The option **-r**, recursively, removes directories and their contents. One important thing to note here is that the files/directories are permanently removed, and not sent to the trash.

```
$ rm <file-name>
$ rmdir <empty directory-name>
$ rmdir -r <directory-name>
```

- The **diff** command is used to find difference in contents of similar files.

```
$ diff <left file-name> <right file-name>
```

- There are several other important file handling commands like **cat**, **find**, **ls**, **cd, chmod** (we will see this later), etc. For more information visit docs.

Mounting and Unmounting

- **Mounting** is a process by which the OS makes files and directories on a storage device available for users to access via the file system.
- In Oralce Solaris, we have several different file systems such as **ZFS**, **LOFS**, **TMPFS**, etc. Since **TMPFS** and **LOFS** are virtual file systems, you actually **access** them by mounting them.
- Let's look at 2 different cases and understand the way mouting occurs for these file systems.
  - **Mouting ZFS :**
    We first create a mirror storage pool with the name **tank**. We assume that the disk **/dev/sdb** is available for use. The pool is automatically mounted at **/tank**.

```
$ zpool create tank mirror /dev/sdb
$ zfs list -r tank
NAME    USED  AVAIL  REFER  MOUNTPOINT
tank    117K   268G   21K    /tank
```


- **Mouting TMPFS :**
  We create, mount, and limit the size of the **TMPFS** file system, **/export/reports**, to **50 MB**. You can set up the system to automatically mount a TMPFS file system at boot time by adding an **/etc/vfstab** entry.

```
$ mkdir /export/reports
$ chmod 777 /export/reports
$ mount -F tmpfs -o size=50m swap /export/reports
$ mount -v
```

- Similarly one can mount other file systems like **LOFS** and **Legacy UFS File Systems**.
- For unmounting file systems, we simply specify the mount-point and use **umount**. We can use **-f** option for forcibly unmounting.

```
$ umount <mount-point>
```

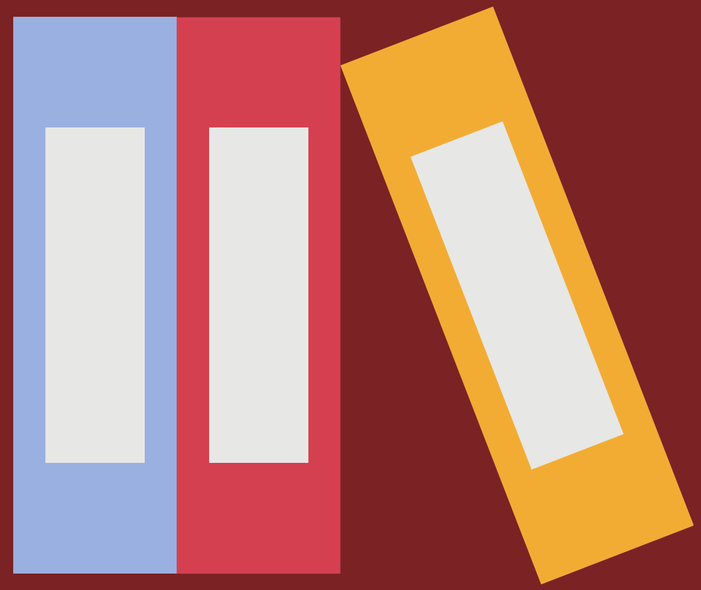
Security and Sharing

- **File Security** in Solaris OS is an important concern, since it's an **enterprise** operating system. Files are kept secure by restricting and monitoring access to them by different users.
- **Access Control Lists (ACLs)** can provide greater control over file permissions. You add ACLs when traditional UNIX file protections are not sufficient.
- Traditional UNIX file protections provide **read**, **write**, and **execute** permissions for the three user classes: **owner**, **group**, and **other**. An ACL provides finer-grained file security.

- A network file server can control which files are available for sharing. Access control is specified when resources are made available with the **share** command.
- The **Basic Audit Reporting Tool (BART)** enables you to comprehensively **validate** systems by performing file-level checks.

References

- Oracle Solaris Administration: Devices and File Systems
  https://docs.oracle.com/cd/E23824_01/html/821-1459/toc.html
- Solaris (operating system)
  https://en.wikipedia.org/wiki/Solaris_(operating_system)
- Operating Systems Documentation
  https://docs.oracle.com/en/operating-systems/index.html
- Operating Systems (CS 343) Module 6 - File System Management, IIT Guwahati



Thank You!

COURSE INSTRUCTORS

Dr. John Jose

Assistant Professor
Department of Computer
Science & Engineering

Dr. T. Venkatesh

Associate Professor
Department of Computer
Science & Engineering