BrainBuzz

Cramsession

Last updated October, 2000.
Click here for updates.

Click here to see additional documents related to this study guide.

Contents

Cramsession™ for Sun Solaris 7 Certified Systems Administration II

Abstract:

This Cramsession will help you to prepare for Sun Exam 310-010, Sun Solaris 7 Certified Systems Administration II. Exam topics include: The Solaris Network Environment, Installing a Server, Syslog and Auditing Utilities, Device Administration, Service Access Facility, Terminals and Modems, Disk Management, Networks, NFS Environment, CacheFS File Systems and WebNFS, Automount, Naming Services, NIS, Solstice AdminSuite, Adding Network Clients, JumpStart and Administration and Configuration of CDE.

Notice: While every precaution has been taken in the preparation of this material, neither the author nor BrainBuzz.com assumes any liability in the event of loss or damage directly or indirectly caused by any inaccuracies or incompleteness of the material contained in this document. The information in this document is provided and distributed “as-is”, without any expressed or implied warranty. Your use of the information in this document is solely at your own risk, and Brainbuzz.com cannot be held liable for any damages incurred through the use of this material. The use of product names in this work is for information purposes only, and does not constitute an endorsement by, or affiliation with BrainBuzz.com. Product names used in this work may be registered trademarks of their manufacturers. This document is protected under US and international copyright laws and is intended for individual, personal use only. For more details, visit our legal page.

The Solaris™ Network Environment

Describe the network capabilities of the Solaris™ 7 network environment

The Solaris™ networking environment is described as a client-server relationship. Computers configured as servers share services like email, applications, and databases to a number of connected client computers and users.

Solaris™ 7 supports NFS, an open, distributed file system, WebNFS and CacheFS, Sun proprietary file systems, and naming systems NIS, NIS+, DNS and FNS.

Define the following terms: server, client, file server, diskless client server, Autoclient server and standalone

Server – a host or process that shares services with other networked client machines

Client – a host or process that relies on shared services provided by server computers

File server – a server that shares files over the network, usually via NFS

Diskless client server – a server that provides startup and operating system files (`/usr`, `/opt`, `/export/swap`) to and stores users data (`/export`) from diskless workstations - those clients that have only CPU, memory and network hardware.

Autoclient server – Similar to a diskless client server, except it does not provide network swap space. AutoClient workstations have local hard drives for swap and cache.

Standalone – a workstation or server that contains all necessary system files and can operate independent of a network

Describe the basic hardware components and file system configurations for diskless clients, AutoClients and standalone workstations

Diskless Clients	Monitor, keyboard, mouse, ethernet and CPU Retrieve `/`, `/usr`, `/home`, `/opt` and `swap` from a server
AutoClients	Monitor, keyboard, mouse, Ethernet, CPU, internal hard disk Retrieve `/` and `/usr` from a server and cache other files to disk
Standalone	Monitor, keyboard, mouse, CPU, hard disk, CDROM, backup Contains `/`, `/usr`, `/opt`, `/export/home` and swap on local disk

Describe the file system configuration for diskless and AutoClient servers, and state the minimum required disk space (in megabytes)

The diskless client does not have a disk and must remotely access its root (/), /usr, /home and any other needed file systems from a server.

The AutoClient system requires a minimum of a 100 MB local disk space for swapping and for caching the root (/) and /usr file system downloaded from the AutoClient server. All other file systems must be remotely accessed from the server.

Installing a server

Identify the three phases of the installation process

System configuration: There are two methods to pre-configure system information. The first involves the use of the sysidcfg file. The second involves using a name service.

- The sysidcfg File Method: A file for each system is created that contains a set of lines in the form of KEYWORD=VALUE. The file can either be available over the network or on diskette.
- The Name Service Method: Entries for each system are added to the tables associated with the name service (NIS or NIS+).

System installation: There are four methods for installing Solaris™. These are interactive, Web Start, JumpStart™ and Custom JumpStart™.

- **Interactive:** Provides a step-by-step guide for installing Solaris™ software. Any co-packaged software must be installed later.
- **Web Start:** Provides a web browser user interface for installing both Solaris™ and co-packaged software.
- **JumpStart™:** Provides automatic installation of Solaris™ software on a new Sparc based system by inserting the Solaris™ 7 CD and powering on the system.
- **Custom JumpStart™:** Provides automatic installation of Solaris™ software on groups of systems, but requires preconfiguration.

Post Installation: Post installation consists of adding the appropriate patches or packages.

Identify the minimum required disk space, in megabytes, for a server installation

A one-gigabyte hard disk is required for Solaris™ interactive installation.

Verify that your system meets the hardware requirements for installing the Solaris™ 7 environment on a server

- SPARC system architecture (sun4c, sun4m, sun4d, or sun4u) or Intel™ system
- 64 Megabytes of RAM
- SCSI CDROM
- 1 gigabyte hard drive
- monitor and keyboard

Describe the terms software packages and software clusters

Clusters – collections of similar software, usually named SUNW<packageabbrev>

Packages – a group of files and directories that make up a particular application. A package is a standard way to distribute applications.

List and describe the software configuration cluster options

There are three types:

Core – The minimum base install, containing drivers for the GUI and network

End User – Core + OpenWindows and CDE

Developer – End User + compiler tools and man pages. Also has 64-bit software

Entire Distribution – All of Solaris™ 7, plus OEM packages

Use the Solaris™ interactive installation program to perform local custom installation of the Solaris™ 7 software

Local installation utilizes the `installtool`. It prompts the user for information – locale, hostname, IP address/subnet, name service, timezone, disk partitions and necessary software groups – then copies the files to the local hard drive.

There is an interactive GUI that provides dialogs for all the configuration information.

Solaris™ Syslog and Auditing utilities

Configure syslog messages by increasing the logging severity level for the login daemons

`syslog` is a daemon that sends messages from system processes to a log file, the console, or via email to specific users. It has several severity levels that configure the degree to which messages are logged.

The default configuration file is `/etc/syslog.conf`, and the default logfile is `/var/adm/messages`.

Two fields of the `syslog` are: selector and action. The selector field is divided into two fields delimited by a period: `facility.level`.

The following table contains the facility definitions:

user	kern	mail	daemon
auth	lpr	news	uucp
cron	local0 - 7	mark	*

The following table contains the severity levels (highest to lowest):

emerg
alert
crit
err
warning
notice
info
debug
none

Changing the severity level of the login daemons requires changing the level associated with the auth facility:

```
auth.info   /var/adm/messages
```

to

```
auth.crit   /var/adm/messages
```

Configure the syslog messages by increasing the logging severity level for the telnet daemons

```
daemon.info /var/adm/messages
```

to

```
daemon.crit /var/adm/messages
```

Given the log from syslog, list the correct actions to limit access to specific users

`syslog` records information like username and host machine IP address in the logs. This information can be examined by an administrator or parsed by a script to determine what type of system access is occurring. Host restriction files like /etc/hosts.deny can then be edited or user accounts disabled using regular system tools.

Use the who utility to determine who is logged onto a system

Who has several flags that determine how users are logged into the system:

-b – the time and date of the last system reboot

-H – print header information in the who display

-t – when the system clock was last changed

-u – who is presently logged in

-q – displays a list of the current users (with total)

Flags may be used together, for example, date and time of the last reboot:

```
$ who –bH
NAME       LINE         TIME
  .       system boot  Oct  1 11:28
```

Use the whodo utility to determine what tasks users are performing

whodo shows the processes in use at each console and terminal. This is useful for figuring out what each user is doing:

```
# whodo

console     deskuser   5:47
    pts/2         1690    0:00 sdt_shell
    pts/2         1692    0:00 tcsh
    pts/2         1706    0:06 dtsession
pts/4        jjaball   2:25
    pts/4         1730    0:00 tcsh
pts/3        josephh   2:25
    pts/3         1731    0:00 tcsh
pts/5        root      2:25
pts/8        root     19:46
    pts/8         8884    0:00 tcsh
```

Use the last utility to determine when the guest user last logged in

Last displays login and logout activity, as well as system reboot times.

```
# last guest

wtmp begins Sat Jul 15 10:02
guest     pts/5      localhost        Mon Sep 11 13:39 - 20:47  (07:07)
```

Use the logins utility to list user and login information

`logins` displays know user and system account activity. It may only be run by root. By default the output shows:

username, userID, primary group name, groupIDs, comment

```
# logins
daemon          1       other        1
bin             2       bin          2
sys             3       sys          3
adm             4       adm          4       Admin
lanman          100     DOS----      11      SunLink Server account
lmxadmin        101     DOS----      11      SunLink Server Admin
lmxguest        102     DOS----      11      SunLink Server GUEST Login
lmworld         103     DOS----      11      SunLink Server World Login
jordonl         104     other        1       Jordan
josephh         105     other        1       Joe
cathyt          106     other        1       Cathy
```

Use the ps utility to determine what processes are running on the system

`ps` displays what processes are running at the time it is executed. By using pipes and other utilities like `grep`, a system administrator can filter the output.

```
ps –ef | grep josephh

josephh 27548 27540  0 06:38:28 pts/3    0:00 /usr/local/bin/tcsh
josephh 27594 27548  0 06:41:35 pts/3    0:00 mailx
```

Device Administration

Define the following terms: serial device, port, serial port, modem, null modem cable, data carrier detect, and port monitor program

Serial device – a serial device is communications hardware that transfers data bit after bit

Port – a port is a means of connecting communications devices and hardware. It is represented physically by pins and wires, and in software via device drivers

Serial Port – a standard communications protocol (RS-232) that connects computers and peripheral hardware

Modem – stands for MOdulation/DEModulation, and is a DCE device that converts data bits into audio signals that may be transferred over phone lines

Null modem cable – allows two DTE devices to communicate without a modem. The direct wire crosses pins 2 and 3, so that DTE devices receive the correct signals

Data Carrier Detect – is a pin in a serial interface that relates the status of a line (if it is free or not). Usually it is pin # 8 on an RS-232 cable

Port monitor program – is software that regulates the communications across ports on a device. It listens for hardware signals that trigger software events

Name at least two serial devices

Termials, modems, printers are all examples of serial devices in the Solaris™ environment.

Describe different modes of modem access

Modems come in three varieties:

Auto-dial only – describes a modem that can dial out across a phone system and connect with a remote computer

Auto-answer only – describes a modem that can recognize a ring indicator and open a receive session with its DTE

Bi-directional – describes a single modem that can perform the functions of auto-dial and auto-answer modems

Give an example of a data communication equipment (DCE) device and a data terminal equipment (DTE) device

DTE devices are typically workstations or printers, and listen to pins on their RS-232 interface for transmits on 2 and receives on 3.

DCE devices are typically modems or switches that listen to pins on their RS-232 interface for transmits on 3 and receives on 2.

The service access facility

Name the top-level SAF process and describe its function

The top-level process for the SAF is `sac`, which monitors terminals and modems connected to the system.

State the run level at which sac is started

Run level *2*

Name the two port monitor types available under SAF and describe each

The two types of port monitors in Solaris™ are `ttymon`, which oversees physical port connections, and `listen`, which manages virtual connections (e.g. `telnet`)

Name the configuration file that tells sac which port monitors to initialize

The `/etc/sac/_sactab` file instructs sac on how to start the various port monitors

Define the term service

A service is process that performs a given function on a server. It is (usually) a daemon that listens for a connection on a certain port. When a port monitor listening on a port detects an incoming request, the connection is routed to the service.

Name the file used by the ttymon port monitor to initialize the speed and terminal settings for each port

The `/etc/ttydefs` is the configuation file for `ttymon`. It contains definitions of baud rates and terminal settings for all TTY ports.

Describe the function of the sacdadm, pmadm, and ttyadm commands

`sacadm` is an administrative command for adding, removing, disabling and monitoring port monitors on the system

```
sacadm –a –p pmtag –t type –c command –v version
```

`pmadm` associates each port monitor with an available service on the system

```
pmadm –a –p pmtag –s service_tag –i identity –fflag –v version –m \
“`ttyadm –l tty_label –d device –T term_type –I ‘message’ –s service \
-S y|n`”
```

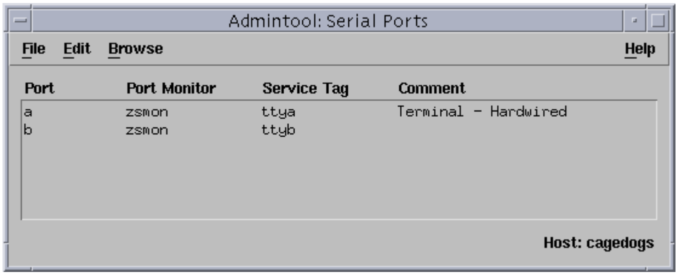
`ttyadm` provides a definition of a specific port, and is used in conjunction with the `pmadm` command (in bold below):

```
pmadm –a –p pmtag –s service_tag –i identity –fflag –v version –m \
“`ttyadm –l tty_label –d device –T term_type –I ‘message’ –s service \
-S y|n`”
```

Adding terminals and modems with Admintool™

Add a terminal to a system using admintool™

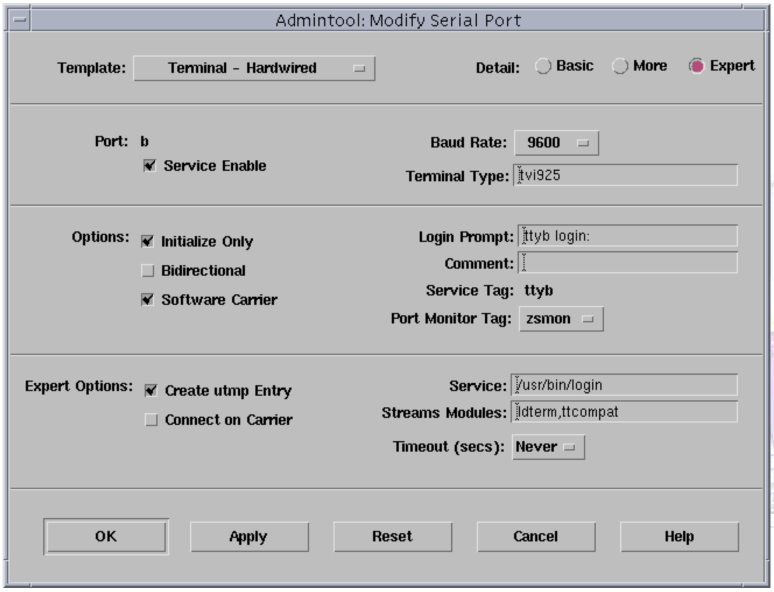
To display the Serial Ports window, select Browse then Serial Ports. Highlight the port to be used. To display the Modify Serial Port Window, select Edit, then Modify.



Port	Port Monitor	Service Tag	Comment
a	zsmon	ttya	Terminal - Hardwired
b	zsmon	ttyb	

Host: cagedogs

Choose Terminal-Hardwired from the User Template menu, change other settings as required, and then click on OK to configure the port.



Template: Terminal - Hardwired

Detail: ○ Basic ○ More ● Expert

Port: b

☑ Service Enable

Baud Rate: 9600

Terminal Type: tvi925

Options: ☑ Initialize Only

□ Bidirectional

☑ Software Carrier

Login Prompt: ttyb login:

Comment:

Service Tag: ttyb

Port Monitor Tag: zsmon

Expert Options: ☑ Create utmp Entry

□ Connect on Carrier

Service: /usr/bin/login

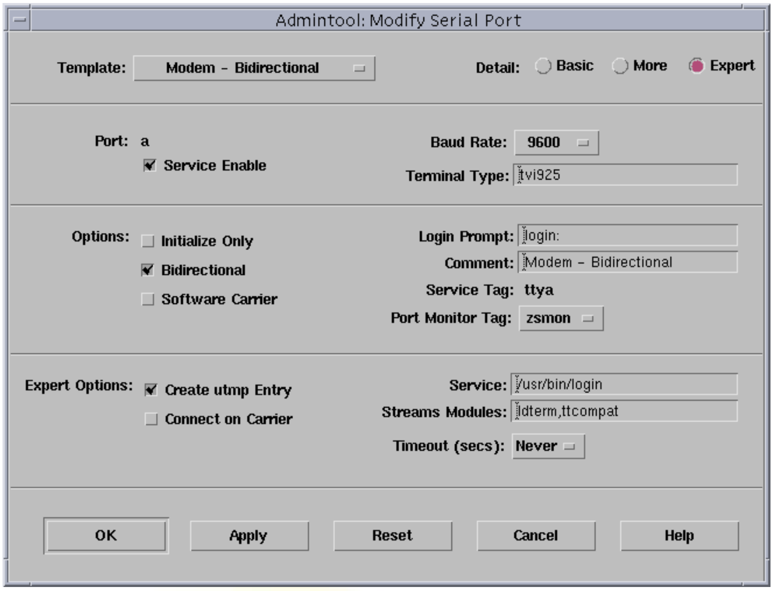
Streams Modules: ldterm,ttcompat

Timeout (secs): Never

Add a bidirectional modem using admintool™

To display the Serial Ports window, select Browse then Serial Ports. Highlight the port to be used. To display the Modify Serial Port Window, select Edit, then Modify.

Choose Modem-Bidirectional from the User Template menu, change other settings as required, and then click on OK to configure the port.



Admintool: Modify Serial Port

Template: Modem – Bidirectional

Detail: ○ Basic ○ More ● Expert

Port: a

☑ Service Enable

Baud Rate: 9600

Terminal Type: tvi925

Options: □ Initialize Only ☑ Bidirectional □ Software Carrier

Login Prompt: login:

Comment: Modem – Bidirectional

Service Tag: ttya

Port Monitor Tag: zsmon

Expert Options: ☑ Create utmp Entry □ Connect on Carrier

Service: /usr/bin/login

Streams Modules: ldterm,ttcompat

Timeout (secs): Never

Describe the syntax and purpose of the tip command

Use tip to connect to remote systems through a serial line (usually a modem)

```
tip [ -baud-rate ] hostname | phone#
```

Name the file that the tip command uses to identify remote systems and line speeds

`tip` reads configuration information from `/etc/remote`

Introduction to disk management

List the three utilities used to create, check, and mount file systems

`newfs` is a utility to create the ufs filesystem on a new partition (remember that it is a friendly front-end to the more configurable `makefs` command).

`fsck` is the utility used to check a new file system. It detects and repairs inconsistencies.

`mount` is the utility that is used to 'attach' a new file system to the existing hierarchy.

Identify the physical pathname differences between standard Solaris™ file systems and virtual file systems

Typically, file systems on Solaris™ are limited to just one partition or slice. Using tools like Solstice™ DiskSuite™ or Sun StorEdge™ Volume Manager, an administrator can span a file system over more than one partition.

Using DiskSuite™, a virtual file system path would look like:

```
/dev/md/rdsk/d10
/dev/md/dsk/d10
```

Using Volume Manager, a virtual file system path would look like:

```
/dev/vx/rdsk/tools/binaries
/dev/vx/dsk/tools/binaries
```

List the potential advantages of any virtual disk management application

Using a virtual disk management application, a systems administrator can overcome disk capacity limitations, and improve performance and reliability by supporting various RAID configurations.

Most virtual disk management utilities are in the form of GUIs that make system setup and breakdown easier.

List the basic difference between Solstice™ DiskSuite™ and Solstice™ enterprise volume manager

Disksuite™ combines disks that have been created using format. A collection of existing disks or partitions makes up a *metadevice*.

Volume manager manages disk space on *subdisks* by formatting into two initial partitions. Slice 3 is a private area that maintains information about the public file system. Slice 4 is used to provide disk space to create new volumes.

List the main advantages of using a concatenated virtual file system

A *concatenated* volume is created by ‘chaining’ disks together. When one volume or partition is full, the system can move on to the next. The advantage is that additional file space can be added ‘on-the-fly’.

List the main advantages of using a striped virtual file system

A striped file system breaks up the reads and writes to the file system across disks. A small section, called a *stripe*, of the actual data is contained on each disk. Much of the access can be done in parallel if the disks are attached to the system using multiple controllers. By adjusting the size of the stripe to the type of data being written, even more performance can be squeezed out.

Networks

Distinguish among different internetwork terms

Some basic internetworking concepts include:

Broadcast bus – stations connected by a wire, or bus that send messages via a broadcast

CSMA/CD – *Carrier Sense Multiple Access with Collision Detection*. Servers employing Ethernet technology use this to detect data collisions on a broadcast network.

Ethernet Interface – a physical device that allows the computer to communicate on a network. Would be identified in software by a device name like hme0.

Ethernet Address – a 48-bit, unique number that identifies the workstation or server on the network. It is located in the NVRAM of a Sun computer.

Internetwork – a group of computers or computer networks linked together to facilitate communication

Internet – the name for the global internetwork. Commercial enterprises, educational facilities and government offices all share the internet.

Describe IP addressing classes A, B and C

An IP address is a 32-bit number laid out in 8 bit fields, called octets. Each IP is unique to a particular host and network. The four default classes are:

Class A – Eight bits in the first octet define the network; 127 usable networks.
Class B – Sixteen bits in the first and second octet define the network; 16,384 usable networks.
Class C – Twenty-one bits in the first three octets define the network; 2,097,152 usable networks
Class D – The first 4 bits are 1110, which results in a first octet value of 224 – 239. The remaining bits define a multicast group.

IPV4 was first described in RFC 791

More information can be found on subnetting in these tutorials:

Learn to Subnet I

Learn to Subnet II

Quick and Dirt Subnetting

Describe the function of each of the following files: /etc/inet/hosts, /etc/nodename, and /etc/hostname.xxy

`/etc/inet/hosts` is a file that contains the mapping of alphabetical hostnames to numerical IP addresses. This file also contains alias names, which allow the host to be known by alternate names when performing special functions (like logging or mail serving). It is also symbolically linked to `/etc/hosts`.

`/etc/nodename` is a file that contains the system's name. It is referenced in the startup script `/etc/rcS.d/S30rootusr.sh`.

`/etc/hostname.xxy` identifies the device name of the Ethernet interface, for example, `/etc/hostname.hme0` references the interface `hme0`. The file contains the name or IP address of the system.

Identify the users logged into the local network

Use the `rusers` command to see all active users on all servers on the network. Running the command and specifying a particular server yields faster results.

Log in to one machine from another machine on the network

Use the `telnet` command to start a remote terminal session on another machine. After supplying a valid username and password, a user may perform tasks on the machine as if they were in front of the system console.

Describe the function of the /etc/hosts.equiv and .rhosts files in relation to network security

`/etc/hosts.equiv` is a list of hosts that the server or workstation trusts. When the file contains a hostname, and the hostname also appears in the `/etc/hosts` file, no password is required to log in. This is an advantage because it saves time when logging into systems that share a local network, and also because it does not transmit a password across a network. It is a security hole, however, because any user from the trusted system may gain access to the trusting machine.

`.rhosts` is a file that contains a list of trusted computers for a particular user. If the `.rhosts` file in a home directory contains a hostname, a log in session with the username/hostname combination does not prompt for a password. While not as wide open as `/etc/hosts.equiv`, using `.rhosts` still poses a security risk when a user may not have to supply a password to log into the system.

Send ping and spray requests to a remote host to test for response

`ping` is a standard command for testing connectivity between hosts. It uses the Internet Control Message Protocol (ICMP) to transmit a 64-byte message to another host, and then waits for an 'echo' reply. If the system cannot be contacted after 20 seconds, the ping times out.

```
# ping <hostname>
```

```
# ping –s <hostname>
```

will generate IP, sequence #, and packet trip time information

`spray` sends a stream of packets to a remote host using remote procedure call, or RCP. Using spray, an administrator can determine if higher-level protocols are running on a remote system.

```
# spray <hostname>
```

Use `spray` carefully, as the stream of packets it sends can be intense enough to adversely impact traffic for other hosts on the network!

Execute the netstat –i command and interpret the output

`netstat` displays status information about the interfaces on a machine. It can be a good starting point when troubleshooting network slowness or connectivity problems.

```
# netstat –i

Name  Mtu  Net/Dest   Address    Ipkts  Ierrs Opkts  Oerrs Collis Queue
lo0   8232 loopback   localhost  75036  0     75036  0     0      0
le0   1500 sparty     sparty     20584  145   55808  85    2134   0
```

Configuring the NFS environment

Describe the functions of an NFS server and an NFS client

NFS stands for network file system, and provides a means of distributing files to clients across a network.

An NFS server stores the files on a local disk, and runs the appropriate daemons to share them across a network.

A NFS client mounts the shared files from an NFS server across the network. The process is almost transparent, in that files on an NFS share appear to be local to the client machine.

Make resources available and unavailable for mounting

The `/usr/sbin/share` command makes files available for remote mounting

```
share [ -F fs-type ] [ -o options ] [ -d description ] path_name
```

```
# share –F nfs /export/home
```

will share out the /export/home directory. Data about the share is logged in the `/etc/dfs/sharetab` file.

Edit the /etc/dfs/dfstab file on an NFS server to enable sharing of resources

`/etc/dfs/dfstab` is similar to the `/etc/vfstab`, in the sense that it is a hard and fast recording of shares to be shared out during the boot process, or by using commands like `shareall`. The file simply contains a listing of share commands.

```
share –F nfs –o ro /export/updates
share –F nfs –o rw:sparty:beaumont /export/home
```

Display a server's available resources for mounting

Use the `dfshares` command to show the resources being shared on the server.

```
dfshares [ -F fs-type ] [ host ]
```

```
# dfshares
RESOURCE                      SERVER         ACCESS      TRANSPORT
sparty:/usr/share/man         sparty           -             -
```

```
# dfshares sparty
RESOURCE                      SERVER        ACCESS     TRANSPORT
sparty:/export/home           sparty          -            -
```

shows the shares available on machine `sparty`

Mount a resource from another system

Use the `mount` command to retrieve resources from a remote host.

```
mount [ -F fs-type ] [ -o options ] server:path_name local_mount_point
```

`mount` has options that include `rw` (read/write) and `ro` (read-only), which are separated by commas.

```
# mount –o ro sparty:/usr/share/man /usr/share/man
```

will mount the remote `/usr/share/man` directory from `sparty` as read only, and use the local `/usr/share/man` directory as the mount point. Local files in `/usr/share/man` will be 'covered up', and won't be accessible as long as the share is mounted.

Edit the /etc/vfstab file to mount resources on an NFS client

```
#device         device        mount        FS     fsck    mount   mount
#to mount       to fsck       point        type   pass    at boot options
#
#
sparty:/export/home -      /export/home   nfs     -       yes    bg
```

Note that the device to `fsck` is a -, because NFS shares are never `fsck`ed. Mount at boot is yes in this case, and the option `bg` will specify 'background' mounting (the system will retry the mount in the background if an initial attempt fails).

Describe the function of these commands: mountall, unmountall, shareall, unshareall

`mountall`, when executed on a client machine, will mount all the shares specified in the `/etc/vfstab` file as 'mount at boot'. When used with the `–r` option, only remote shares will be mounted.

```
# mountall –r
```

`unmountall` will unmount the current mounted file systems. When used with the `–r` option, only remote shares will be umounted.

```
# umountall –r
```

`shareall` is run on a server machine to automatically share all the filesystems listed in the `/etc/dfs/dfstab` file.

`unshareall` is run on a server machine to unshare all of the currently shared file systems. It reads from the `/etc/dfs/sharetab` file.

CacheFS file systems and WebNFS

Describe and configure the CacheFS file system

CacheFS is available to speed up access to slow remotely mounted file systems or devices like CDROMs. Basically, a local cache is created on the hard disk, and requests for the remote data are redirected to the cache.

CacheFS has three main terms to remember:

1. **Back File System** – the original source of the data, be it network or CDROM
2. **Front File System** – the mounted, cached local file system
3. **Consistency** – the synchronization status between the front and back systems

To create a CacheFS file system, use the `cfsadmin` utility:

```
# cfsadmin –c /export/c_home_dirs
# mount –F cachefs –o backfstype=nfs,cachedir=/slow_remote_system, \
cacheid=remote_system_0930 server1:/export/home /export/c_home_dirs
```

Where `server1` is the remote server sharing its `/export/home`, which is being cached on this server in `/export/c_home_dirs`.

Provide read/write Internet access to an NFS resource through a Web browser

Only the Sun Hotjava browser supports WebNFS. It allows a user to enter an address `nfs://<servername>/<pathname>` into the URL window, and the browser will display the contents of the share.

In order for a share to be available using WebNFS, the share command must be run with the –o public option:

```
# share –F nfs –o public /export/home
```

Using Automount

List three benefits of using the automount utility

File mounts on demand – automatic mounts when referenced share is needed.

A timeout feature can be configured to unmount shares that haven't been used.

A name service can be configured to manage NFS mounts in conjunction with automount.

Load balancing and redundancy between servers when multiple servers share the same file systems.

Describe the purpose of each of the types of automount maps

There are three types of maps used by the autofs:

Master Maps – automount reads these maps to determine what other maps are present for the autofs environment

Direct Maps – contain the absolute pathnames to mount points

Indirect Maps – contain the relative pathnames to mount points

Set up automount to read a direct map

Share out data that is stored in an `/export/opt` directory

The `/etc/auto_master` map file must be modified to specify a new direct map file

```
+auto_master
/-          auto_direct
/net        -hosts      -nosuid,nobrowse
/home       auto_home   -nobrowse
```

A new file called `/etc/auto_direct` must be created, and an entry containing the absolute pathnames to the shared data must be entered

```
/export/opt -ro   sparty:/export/opt
```

Re-run `automount` with the –v option to make sure the changes take effect. The autofs daemons may also be stopped and started. This must be done when making changes to either the master maps or the direct maps.

```
# automount –v
```

Describe when the automountd daemon should be restarted

As stated above in the direct map example, the `automountd` daemons should be restarted whenever the master maps or direct maps are modified.

Naming Services Overview

Describe the name service concept

Name services act as intermediaries between the numerical addressing scheme that computers use to identify themselves on a network, and alphabetical host names that are easy for humans to remember. Special software on clients and servers translate between the two on the fly. More complex name services also share account information and machine configurations.

List the name services available

Solaris™ 7 supports four name service options:

Domain Name Service (DNS)

Network Information Service (NIS)

Network Information Service Plus (NIS+)

Federated Naming Service (FNS)

Define DNS

DNS is a solution to the problems inherent to managing computer system hostnames. These hostnames must have an efficient way to resolve to their corresponding numerical addresses, and maintain uniqueness on the Internet with respect to certain organizations. (i.e. `host1.companya.com` and `host1.companyb.com` share similar hostnames but are unique machines on the internet).

DNS is described in RFC 1035

Describe the NIS service

NIS was designed by Sun to store and share more than just information about host names and addresses. NIS attempts to ease the headaches of maintaining separate configuration files on each client computer in a network by creating a *namespace* (also called a *domain*) that allows for centralized management. NIS servers replicate *maps* to client machines in their namespaces.

Describe the function of NIS+

NIS+ is not an update of NIS, it is a completely new NIS implementation. It stores much more information than NIS, including security, mail, Ethernet, printer and network services.

List the table objects in an NIS+ domain

Object	Description
`auto_home`	Location of all user home directories
`auto_master`	Automounter map information
`bootparams`	location of root, swap, and dump partitions of every diskless client in the domain
`cred`	credentials of the principals
`ethers`	Ethernet addresses of every workstation
`group`	Group name, ID, password and members of every UNIX group in the domain
`hosts`	Network address and every corresponding workstation
`mail_aliases`	Information about the mail aliases of users in the domain
`netgroup`	Network groups and their members
`netmasks`	Networks in the domain and their netmasks
`networks`	Networks in the domains and their canonical names

`passwd`	Password information about every user in the domain
`protocols`	List of IP protocols used in the domain
`RPC`	The RPC program numbers RPC services available in the domain
`services`	Names of IP services used in the domain and their port numbers
`timezone`	Timezone of every workstation in the domain

Explain the difference between authentication and authorization

Authentication is a way to restrict access to specific users when accessing a remote system, which can be setup at both the system and network level. For NIS+, a credentials check is performed before each access request is authenticated.

Authorization is a way to restrict operations that the user can perform on the remote system once the user gains access. For NIS+, every component in the namespace specifies the type of operation it will accept and from whom.

Describe the name service switch process and determine which configuration is appropriate for your network

The name service switch is a file (`/etc/nsswitch.conf`) that controls how network information is obtained. Each client on the network has a local copy of this file. Entries in the file determine how a particular type of information is obtained (e.g., from NIS, NIS+, DNS, etc.)

Tables or objects are listed in the file and can be can be configured individually for each name service and in which order those name services should be queried when a lookup is necessary.

A network that uses DNS as a primary host resolver would specify dns first in the list that trails the “hosts:” entry of in the `nsswitch.conf`. By default, the file is configured to use NIS.

Template files exist to make the configuration easier. For example, to enable NIS, simply copy the `/etc/nsswitch.nis` to `/etc/nsswitch.conf`.

NIS

Describe the NIS components, master server, slave server, client and NIS processes

An NIS system is comprised of servers that act as repositories for configuration information, which is shared with clients that are all members of the same domain.

The NIS master server is a single server that contains the master copies of configuration tables for the network. These map files are built from special ASCII files and stored in the `/etc/` directory. Having one server allows for a single configuration and control point for an entire domain. Remember that a NIS server is a client of itself in the domain.

The NIS slave server receives map files from the master server in the domain. They act as backups in the event that a master server becomes unavailable. In busy networks, their presence in the domain serves as load balancing for map requests.

The NIS client does not contain any local maps. Processes on the client bind to the master or a slave server for configuration information. In the event that the host it is bound to should become unavailable, it can dynamically rebind to another working server.

There are five main processes in an NIS environment:

1. `ypserv` – answers ypbind requests
2. `ypbind` – binds to domain server and stores binding information
3. `rpc.ypcpasswdd` – accounts for password changes and updates appropriate maps
4. `ypxfrd` – transfers NIS maps within the domain
5. `rpc.ypupdated` – also transfers NIS maps within the domain

`ypserv` is found on master and slave servers. `ypbind` is run on master, slave and client computers. `rpc.ypcpasswdd`, `ypxfrd` and `rpc.ypupdated` are all run on master servers only.

Configure an NIS master, slave and client

For a master server:

1. Make sure the computer is configured for NIS
2. Set the domain name using the `domainname` command and editing `/etc/defaultdomain` file
3. Make sure the text files with configuration information are up to date in the `/etc` directory (touch `ethers`, `bootparams`, `locale`, `timezone`, `netgroup` and `netmasks` files so they have a zero length)
4. Run `ypinit –m` and specify slave servers when asked.
5. Start the NIS daemons: `/usr/lib/netsvc/yp/ypstart`

For a slave server:

1. Make sure the computer is configured for NIS
2. Set the domain name using the `domainname` command and editing `/etc/defaultdomain` file
3. Initialize the machine as a client: `ypinit –c`, enter in the names of the other master and slave servers when prompted
4. Start the NIS daemons: `/usr/lib/netsvc/yp/ypstart`
5. Initialized the machine as a slave: `ypinit –s <master name>`
6. Stop and start the NIS daemons on the new slave server

For a client:

1. Make sure the computer is configured for NIS
2. Make sure the `/etc/hosts` file contains the master and slave servers
3. Set the local domain name by using the `domainname` command
4. Initialize as a client: `ypinit –c`, enter in the names of the master and slave servers when prompted

List the steps to add a new NIS map

NIS maps are built using the make utility. Make reads a file called `Makefile` that contains macros and other instructions for creating *targets*, which are the final maps. When adding a new map, the name of the map file must be entered into the `Makefile` at the end of the `all:` target list.

The command to build NIS maps is:

```
# /var/yp
# make all
```

Use commands to update and propagate an NIS map

Run the make command to update and propagate NIS maps to all slave servers:

```
# cd /var/yp; make
```

From a slave server, these scripts will pull new host maps:

```
# /usr/lib/netsvc/yp/ypxfr hosts.byaddr
# /usr/lib/netsvc/yp/ypxfr hosts.byname
```

(others can be specified, e.g. `ethers.byname`, `netgroup.byhost`…)

or

```
# ypinit –s <master name>
```

will pull down updated map files.

The command:

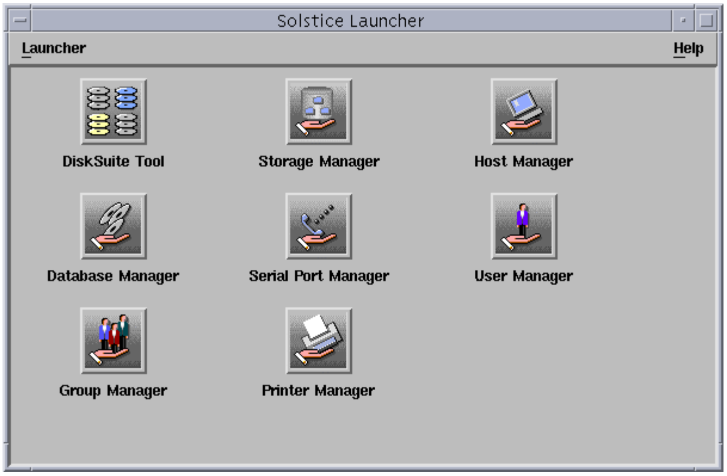
```
# /usr/lib/netsvc/yp/rpc.yppasswdd /etc/passwd –m passwd
```

will keep the `passwd` file in sync when a user in the domain changes her password, and push the updated map to slaves.

Solstice™ Adminsuite™

List the capabilities of Solstice™ Adminsuite™

Solstice™ AdminSuite™ is a collection of GUI tools and commands used to perform administrative tasks such as managing hosts, users, groups, system files, printers, disks, file systems, terminals and modems. These tools and commands provide a graphical interface to the Solaris™ command-line tasks. When using AdminSuite™, system files are automatically edited to eliminate errors. AdminSuite™ can also manage hosts remotely.



Use Solstice™ AdminSuite™ to add a host using Host Manager

To add support for a standalone system, OS server or other type of system using the Host Manager:

1. Start Host Manager from the Solstice™ Launcher by clicking on the Host Manager icon and then selecting the name service.
2. On the Host Manager window, select Add from the pull-down edit menu and the *Host Manager:Add* window is displayed.
3. Fill in the system information.
4. To use remote install capabilities click on Enable Remote Install and select the appropriate options.
5. Click on the OK button.

The command line equivalent for adding a host uses the **admhostadd** with the following arguments:

```
admhostadd -i client_ip_address -e client_ethernet_address \
specific_settings client
```

where *specific_settings* arguments such as `-x type=DATALESS`, `-x tz=US/Mountain`, `-x os=sparc.sun4c.Solaris_2.7`, etc. and *client* is the system name of the client.

Use Solstice™ AdminSuite™ to view mount point and disk slice information using the Storage Manager

The Storage Manager consists of the Load Context window, the File Manager and the Disk Manager Tools. The Load Context window allows the ability to select the host to manage with the File Manager and the disk set to manage with the Disk Manager.

To view mount point information using the Storage Manager:

1. Start Storage Manager from the Solstice™ Launcher by clicking on the Storage Manager icon and then selecting the appropriate host name in the *Storage Manager:Load Context* window.
2. On the *Storage Manager:Load Context* windows, select the *File Manager* entry in the Property Browser frame. The main window on the File Manager is

displayed. Information on the mount points for the current context is displayed.

To view disk slice information using the Storage Manager:

1. Start Storage Manager from the Solstice™ Launcher by clicking on the Storage Manager icon and then selecting the appropriate host name in the *Storage Manager:Load Context* window.
2. On the *Storage Manager:Load Context* windows, select the *Disk Manager* entry in the Proptery Browser frame. The main window of the Disk Manager is displayed. Information on the slices for the current context is displayed.

Use Solstice™ AdminSuite™ to view time zone information using the Database Manager

The Database Manager is a graphical user interface for managing the various network-related (`/etc`) system files such as `hosts`, `passwd`, services and `timezone`. To view `timezone` information (the contents of `/etc/timezone`) using the Database Manager:

1. Start Database Manager from the Solstice™ Launcher by clicking on the Database Manager icon. The Database Manager Main Window is displayed.
2. Select the name service used on the network.
3. Select the desired host.
4. Select the `timezone` file. The contents of the `timezone` file are displayed.

Use Solstice™ AdminSuite™ to view the characteristics of a serial port using the Serial Port Manager

To view the characteristics of a serial port using the Serial Port Manager:

1. Start Serial Port Manager from the Solstice™ Launcher by clicking on the Serial Port Manager icon. The Serial Port Manager Main Window is displayed.
2. If the serial port is on another system, select the View pull-down menu and select a host from the list of hosts displayed in the Host window.
3. Click on the port to be viewed to highlight it.
4. Select Modify from the Edit pull-down menu to display the *Serial Port Manager:Modify* window.
5. Click on the Cancel button when finished.

Use Solstice™ AdminSuite™ to add a user using the User Manager

To add a user account using the User Manager:

1. Start User Manager from the Solstice™ Launcher by clicking on the User Manager icon.

2. On the User Manager Main window, select Add from the pull-down edit menu and the *User Manager:Add* window is displayed.
3. Fill in the user account information. This includes user name, user ID, groups, password information, home directory path and login shell.
4. Click on the OK button

Use Solstice™ AdminSuite™ to add a user to a group using the Group Manager

To add a user to a group using the Group Manager:

1. Start Group Manager from the Solstice™ Launcher by clicking on the Group Manager icon. A list of groups is displayed.
2. Select the group to be modified.
3. On the Group Manager Main window, select Modify from the pull-down edit menu and the *Group Manager:Modify* window is displayed.
4. Add the user account to the Members List field.
5. Click on the OK button

Use Solstice™ AdminSuite™ to add a printer using the Printer Manager

The Printer Manager can be used to install both locally attached printers and network printers.

To install a local printer using the Print Manager:

1. Start Print Manager from the Solstice™ Launcher by clicking on the Print Manager icon and then selecting the name service.
2. On the Print Manager main window, select *Install Printer* from the pull-down Edit menu and the *Print Manager:Install Printer* window is displayed.
3. Fill in the printer information. This includes printer name, description, printer type, printer port, and various printer options.
4. Click on the OK button

To install a network printer using the Print Manager:

1. Start Print Manager from the Solstice™ Launcher by clicking on the Print Manager icon and then selecting the name service.
2. On the Print Manager main window, select *Install Network Printer* from the pull-down Edit menu and the *Print Manager:Install Network Printer* window is displayed.
3. Fill in the printer information. This includes: printer name, description, printer type, protocol and various printer options.
4. Click on the OK button.

Adding Network Clients

List the client requirements a server must meet to support diskless and AutoClient network clients

OS servers provide the operating system for diskless and AutoClient (sometimes called Field Replaceable Units) computers over the network. Support of network clients takes several forms:

- Providing network information to the clients about themselves and other clients (host names, IP addresses, etc.)
- Providing installation services to remote boot and install a system
- Providing operation system services to clients with little to no disk storage space

For Diskless clients and AutoClients, a server must provide ability to remotely access the operating system and application file systems via NFS. Diskless clients access this information remotely while AutoClients will locally cache root (`/`) and `/usr`.

Use Solstice™ to add support for a network client

The *Host Manager*, which is a tool provided with Solstice™ AdminSuite™, is used to add support for AutoClients, Diskless clients, JavaStation clients and Dataless clients. Adding support for all types of clients follows the same high-level procedure:

1. On the *OS Server*, add system information about the client using *Host Manager*.
2. Using *Host Manager*, select OS release and define appropriate root and swap paths for the client.
3. Boot the client and set the root password for the client.

List the steps necessary to create an OS Server

Using the *Host Manager*, a standalone system, dataless client or generic system can be converted to an *OS Server*.

1. Start *Host Manager* and select the appropriate host from the main window.
2. From the *Edit* menu, select *Convert to OS Server*. The *Host Manager Convert* window is displayed with the host information fields filled-in.
3. Specify the versions of operating systems (OS Services) that should be supported by the *OS Server* and click on the *Add* button.
4. Click on the *OK* button, then select *Save* from the *File* menu.

The command `admhostmod` can be used instead of *Host Manager* to convert a system to an *OS Server*.

Name at least three files in the /etc directory that the host manager edits when adding support for a network client

Host Manager has the ability to edit the following files:

`/etc/bootparams` - paths to client boot, root and swap areas

`/etc/dfs/dfstab` - share commands used to share file systems via NFS

`/etc/ethers` - ethernet addresses of clients

`/etc/hosts` - hosts names and IP addresses of clients

`/etc/timezone` - timezone settings of clients

`/etc/inetd.conf` - inetd configuration file

Jumpstart™-automatic installation

Briefly describe the Jumpstart™ feature

Jumpstart™ is built into Solaris™ 7 to allow administrators to quickly and consistently install or upgrade the operating system on new or existing Sun computers.

Custom Jumpstart™ is a method to automatically install groups of identical systems. To customize Jumpstart™, a text file called *rules* must be created that lists one or more *profiles*. A *profile* is a text file that defines how Solaris™ is to be installed on a group of systems. Once these files are completed, they are validated using the *check* script. In a non-networked environment, the validated files are placed on a diskette in the *jumpstart* directory and the system is booted. Then the appropriate profile is selected to direct the installation of Solaris™. In a networked environment, the *jumpstart* directory is located on a network server.

List the main components for setting up a network to use automatic installation

Remember, any of the four default install methods for Solaris™ (and not just JumpStart™) can be used when installing over the network.

Jumpstart™ requires the presence of certain servers for the process to work smoothly. The main components for setting up networks for automatic install are:

- An *install server* somewhere on the network with a CD-ROM drive. The OS CD is either copied to the server's hard disk or mounted and shared from the CD-ROM drive.
- If the *install server* is on a different subnet than the systems being installed, a *boot server* is required to boot the systems.
- If appropriate, a *name server* such as NIS or NIS+ is used to provide network information about the new systems.
- If appropriate, the Solstice™ Host Manager is used to add network information about the systems being installed to NIS or NIS+ or instead the `add_install_client` to add network information to the */etc* files of the install or boot server.

- If the systems being installed are diskless or Autoclients, an *OS* server that will be used to provide the Solaris™ operating system.

Set up the network to automatically provide the information necessary to configure a system

In order for a new system to begin the Jumpstart™ process, it must first be able to determine who it is uniquely on a network. This is handled by a NIS or NIS+ environment, or through the use of a sysidcfg file on the server (or accessible via floppy or CDROM) specific to the client.

A `sysidcfg` file looks similar to the following:

```
system_locale=en)us
timezone=US/Pacific
timeserver=10.10.15.20
name_service=NONE
root_password=MPAdmhcDb4q
timezone=US/Eastern
network_interface=hme0 (netmask=255.255.255.0)
```

Create an install server on the network

On the system that will be the *install server*:

1. Log in as root.
2. Insert the Solaris™ CD in the CD-ROM drive (and mount it if it does not automatically mount).
3. If the Solaris™ CD is to be used for installation, stop here. If the Solaris™ CD is to be copied to the hard disk of the *install server*, then change directory to the *Solaris_2.7/Tools* directory on the Solaris™ CD and execute the `setup_install_server` command to copy the contents of the Solaris™ CD to the hard disk.

   ```
   # ./setup_install_server /export/install
   ```

Create a boot server on the network

On the system that will be the *boot server*:

1. Log in as root.
2. Mount the Solaris™ CD either by inserting it in the CD-ROM drive or mounting it via *NFS* from another system.
3. Change directory to the *Solaris_2.7/Tools* directory on the Solaris™ CD and execute the `setup_install_server -b` command to boot the software from the Solaris™ CD to hard disk.

   ```
   # ./setup_install_server –b /export/install/<client-name>
   ```

4. To add install clients, cd to the client directory and run the `add_install_client` script

Remember, the functionality of the install and boot servers can be combined into one machine.

Create a configuration directory with a customized rules file and class files

The custom JumpStart™ files can either be located on a diskette or on a server (called a *profile server*) where they are shared via NFS. Preparing a custom JumpStart™ directory and files consists of:

1. Creating the JumpStart™ directory on diskette or on the *profile server*.
2. Create a rule for each group of systems in the *rules* file using the appropriate keywords and syntax. Example rules file entries:

   ```
   network  10.10.16.0  && ! model ‘SUNW, Sun 4_50’ – class_net16  -
   memsize  16-32       &&   arch  sparc   -  class_admin_support  -
   ```

3. Create a *profile* for each rule that specifies how a system will be installed using the appropriate keywords and syntax.
4. Test the *profile*(s).
5. Validate the *rules* file.

Be able to recognize a valid rule file and profile file.

Add install clients to install servers and boot servers

Boot and install servers need clients added to them. Adding an install client involves running the add_install_client script.

```
# ./add_install_client –e Ethernet –i ip_addr –s install_svr:/dist \
-c config_svr:/config_dir –p config_svr:/config_dir client_name \
client_arch
```

-e specifies the Ethernet address
-i specifies the IP address of the client
-s specifies the name of the install server and path to install distribution
-c specifies the configuration server and the path to the config directory
-p specifies the path to the sysidcfg file

Boot install clients

Turn on the new machine, and run the command from the ok prompt:

```
ok boot net – install
```

Administration and configuration of CDE

Configure the CDE login manager

The login manager is the users' initial contact with the CDE environment. If the username and password entered in at the prompt authenticate, the manager will start the desktop session.

The default appearance of the login screen is determined by the contents of the `/usr/dt/config/C/Xresources` file. To customize the login screen, copy the default `Xresources` file to `/etc/dt/config/C` and make the appropriate changes. Behavior (start, stop, etc.) of the Login manager is controlled by the `dtconfig` command.

Configure the CDE session manager

The Session Manager starts the desktop and restores the environment (applications, color, fonts, etc.) back to the state when the user last logged out. The Session Manager default is `/usr/dt/config/sys.dtprofile`, the system wide settings are under `/etc/dt/config/C/sys.resources` and the personal settings are under **home-directory**`/.dt/sessions/` or **home-directory**`/.dt/`**display**, where **home-directory** is the home directory of the login and **display** is the name of the display. The `dt.session` file contains names of the active windows and their state (size, placement, etc.).

Configure environment variables with CDE files

Default environment variables are set from the `.dtprofile` file. This file is read when the `DTSOURCEPROFILE` environment variable is set to true. The environment variables may come from several sources:

`$HOME/.dtprofile`

Login Manager Defaults

Personal environment variables

and are assigned based on precedence:

1. `$HOME/.dt/config`
2. `/etc/dt/config`
3. `/usr/dt/config`

Modify the Workspace Manager menus; including CDE

The workspace manager reads configuration from the following locations:

Personal:	`$HOME/.dt/$LANG/dtwmrc` `$HOME/.dt/$LANG/wsmenu.dtwmrc`
System:	`/etc/dt/config/$LANG/sys.dtwmrc` `/etc/dt/config/$LANG/wsmenu`
Built-in:	`/usr/dt/config/$LANG/sys.dtwmrc` `/usr/dt/config/$LANG/wsmenu`

To create a system wide workspace menu, copy `/usr/dt/config/C/sys.dtwmrc` to `/etc/dt/config/C/sys.dtwmrc` and edit.

Workspace and Front Panel menus

There are three menus to be aware of:

Workspace menu – the menu that appears when a right-click (far-right) button is pressed

Window Menu – The menu that appears when a left click is performed on the '-' (in the upper left corner of a window)

Front Panel Menu – the menu that appears when a user right-clicks on the front panel (bottom of screen)

The Front Panel pop up windows can be customized by editing a user's

```
$HOME/.dt/types/fp_dynamic
```

file.

Solaris™, SunInstall™, Jumpstart™, Admintool™, Solstice™ and DiskSuite™ are registered trademarks of Sun Microsystems, Inc.

Special Thanks to Matthew Kortas for contributing this
Cramsession. Make sure to visit his site at:
http://acm.cse.msu.edu/~kortasma