UNIT- 2



DISTRIBUTED SHARF MEMORY & DISTRIBUTED FILE SYSTEM

	
(1)	Basic concept of Distributed Shore Memory (DSM) -
	DSM system is a renounce management component of a
	distributed oferating system that implements the shared memory
	model in distributed system, which have no physical should
-	memory. The should memory model provides a virtual addres
	share that is showed among all nodes in a distributed systems.
	I DSM is also referred as Distributed Shared Visitual Memory
	(DSVM).
	Advantages -
-	(4) Shields programmes from send/receive primitives
	(2) hauge virtual memory space.
	(3) Single addres spore
	(4) Simple Hoftwore interfaces.
	(5) Programs portable
	Disadvantages -
	(1) May incur a ferformance penality
	(2) No probetion against shoul data.
2	DSM Architecture and its types -
-	<u>a</u>
	DISTRIBUTED SHARED MEMORY
	(exists only virtually)
	CPU1K MEMORY CPU1K MEMORY
	CPU By CPU n K
	MEMORY MAPPING MANAGER MEMORY MAPPING MANAGER
	Node 1 Node 2 Node n
	COMMUNICATION NETWORK
1	

ARCHITECTURE

Mycompanion

DSM

furcers on different nodes to enchange memages with each other A roftware memory-mapping manages noutine in each node maps the weal memory onto the shared virtual memory. To facilitate the mapping operation, the shared-memory space is partitioned into block Data caching is used in DSM systems to reduce niture lating.

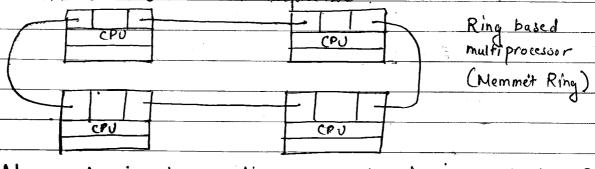
Type of DSM-

Type of DSM -	<u> </u>		
\(\frac{\alpha}{\cdot \cdot \cdot \} \)	vory - Several for	rocemors & a sha	ud memory one on the
rame chip.			chip howhome
•	CPU ···	MEMORY	, pro 1 g
	addus and do	ata lines	
	SINALE CHIP	COMPUTER	

(2) Bus-bared systems - CPU's and memories are connected to a BUS. (PU's either have a do not have local memories.

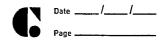
(PU) (PU) [MEMORY]

(3) Ring-band multiprocess. Single address space is divided into a fruite fant and a fruitie shared front. The private front is divided up into regions to that each markine has a piece for its starts and other unshared data and code. The shared front is common to all machines and is kept consistent by a hardware protocol roughly similar to those and on trus-based multiprocesses.



Memment ening is a modified Token-panning ring which has 26 farallel wires (16 data hits 44 control hits) to sent hit at every 100 nsec for a data rate of 160 Mbps. Shand memory is divided into 32-byte Wood any companion

(4) Switched Multiprocessor - Three clusters connected	hy an inter-duster
his to form one super-cluster. CLUSTER	Q
IC C C M	
CCCCCM	
ICI CI CI MI	INTER CLUSTER
THIORGINITES DA	
INTRACLUSTER BUS SUPER CLUSTER	
(5) NOMA (Non-Uniform Homory Acces) multipromore	2 -
PROCESSOR WITH	-
PRIVATE CACHE MEMORY	
INTER CONNECTION NETWORK	
PROCESSOR WITH PROCESSOR WITH	MEMORY
PRIVATE CACHE PRIVATE CACHE	,
A processors own internal computations can be	done in its own
local memory leading to reduced memory contentions According remote memory is possible but slower local memory	h -
Accuraing remote memory is horrible but slave	a the co
local menory	deriming of
Remote acces times are not hidden by caching	•
Dengin and Implementation Issues in DSM hystem -	
(1) Granulanty - When a nonlocal memory word	10 00 10 00 10
Chunk of memory containing the	vi regenera, a
chunk of memory containing the word is fetched fro	m us current
docution and put on the machine making the re	fireme. An
A word, block, page, or signest (multiple pages).	k would be 4
- (of the proper proges)	
Mycompanion	



	(2) Structured of shared memory space - depends on the type of application
_	and an Dam dynam armand to riphor.
	(3) Data water and access - to share data in a GSM hystem
_	(3) Data location and access - to share data in a DSM system (4) Replacement strategy - Data block of local memory must be replaced by a new data block.
	(5) Throshing - Data blocks migrate between rudes on demand
ì	Challenges in DSM -
	(1) How to keep track of the weation of remote data?
_	& How to overcome the communication delays & high overhead
	arrounted with the references to remote data?
	(3) How to allow "controlled" concurrent access to shared data?
	Shurtings of should memory shore -
	These commobile used about the to the train
	Three commobily used approaches for structuring -
	(2) Structuring by data type - Collection of objects or variables in a vource langue
	(3) Structuring on a database
	> should memory space is ordered as an amoriative memory, called a
	tuple space, which is a collection of tupes with data items in their
	fields-
-	
	Connolany Model -
	It determines when the date updates are propagated
	and what level of incornisting is acceptable. Models on as follows
	(1) since winning =
	A shared memory elystem is said to support the strict
	connisting model if the value returned by the read operation on a
١	memory address is always the rame as the value unitter by the

Not suitable in distributed system. Difficult to achive in good

most vicent write operation to that address.

real systems as network delays can be variable my companion

(5)

\dashv	
-	(2) Signantial Committing - (strongest memory model for DSM)
1	A DSM system is said to be signestially consistent if for any
-	encution there is some interleaving of the series of operations inmed
	g as the province that ranges the following two cutina -
+	(1) SC1 - The interleaved requence of operations is nuch that if occurs
	in the require then either the last with ofwation that occurs before it
	in the interleaved sequence is no write operation occurs before it.
-	(11) SCZ - The order of operations in the interleaving is consistent with
$\frac{\parallel}{\parallel}$	the program order in which each inclindual client encuted them.
∦	It provides single-copy semantics because all processes sharing
1	a memory location always see enactly the same contents stored in it
\parallel	
$\ \cdot \ $	(3) hinesingability-
\parallel	The result of any enecution is the same as if the operations
	of all processes on the data were executed in some total order.
\parallel	> Bring in server's view to define the ordering of concurrent events
╬	-> Real time actions performed on the servers.
#	Non-oberlapping requests.
F	Averlapping request - Enqueuing times of the requests one in different orders on different servers can have orbitary order, but sequentially committent.
	orders on different servers can have orbitary order, but sequentially commitent.
	0
(4) Causal consistency -
-	Operations that are causally related must be seen by all processes
	in the same conserponding order Concurrent writes from different procurous
(do not have any causal relationship and can be seen in different order by
-	different processors. There is no need to write enclusively cheaper wife operation.
-	Need to keep track of dependency relations in order to determine
L	whether two events are consulty related. This model is enfumice.
	Concurred writes (aurally related
	P1: $W(x)$ 1 $\int W(x)$ 3 P1: $W(x)$ 1
	P2: $R(x)1 W(x)2$ P2: $R(x)1 W(x)2$

does not obeyo casual consistincy

iny companion carnal connisting

(5) Pefielmid RAM Communy -
It can be implemented by imply sequencing the write operations performed at each nock to independently of the write operations performed on other nodes.
operations performed at each nock to independently of the write
operations performed on other nodes
It is simple and carry to implement and also has good performance
(6) Weak Connisting-
It has these properties -
(1) Access to synchronization variable are requestially consistent.
(1) Accesses to synchronization variable are requestially consistent. (2) No access to a synchronization variable is allowed to be performed
until all previous writes have completed everywhere
(3) No data access (write orward) is allowed to be performed until
all previous accesses to synchionization vonables have been performed.
Weak consistency requires the programmes to use locks to
ennue wads and writes are done in the proper order for data that needs it
Implementing sequential conniting model - There are different types of application & migration lecturgues -
There are different types of replication & migration lichniques -

(1) Non Replicated, Non Migrating Blocks (NRNMBs)

Client Node (sends agent and receive response)

Owner Mode of the Bock (receives request, performs data)

Response

- (1) high copy of each block in the contine system
- (2) hocation of a Hock never changes.

Prowtocks

- (1) Saidinging data occess creates a nottlement.
- (2) Parallelinn is not pumble

This method is simple and cary to implement

	(2) Non Replicated Migrating Blocks (NRMBS) -
	Client Node - Owner Node
	(becomes new owner node) () (owns the block before its)
	Client Node (becomes new owner hode of block after its migration) Plate missarbian Owner Node migration migration
	Block migration
	Characteristics -
	(1) No communication cost incurred when a proven access data
	currently held locally.
	(ii) Allow applications to take advantage of data access locally
	(ii) Allow applications to take advantage of data access locally. Drawbacks -
	(i) Prone to thrashing problem (poor performance) (ii) Parallelism is not possible.
	(11) Parallelism is not possible.
	(3) Replicated Migrating Blocks (RMBs) -
	Characteristics -
	(1) Parallelins is possible du to application of blocks.
	Drawbachs -
	(i) Increases the cost of write operation to be perform in all replicas. Turo protocols is used to for ensuing sequential consistency—
٦	Turo purious is used to ensuing sequential commence
	(1) Write I Malidate -
	3. Invalidate black (receives request, performe data)
	Client Node 1. Request Black (access and sends response)
	has the valid 2 Replicate Block copy of the data
	block after write operation) 3. Invalidate block
	3. Invalidate block
	(ii) Write Update - Nodes having valid whise of the data
	Client Node 1. Request Block
	Client Node (also has a valid 2. Replicate Block
_	copy of the data
	brock often write operation) 3 Update block
	3. Update block

(4) Replicated Non-migrating Blocks (RNMBs) -
(a) Replicated Non-migrating Blocks (RNMBs) - → Replication of Works.
- horation of Replica is fined.
-> Write-update protocol is used.
-> Sequential conniting is ensued by using a Global-sequences.
Thrashing -
······································
It occurs when retwork resources are enhausted, and more timo is spend invalidating data and sending applates that is used doing
actual work.
Two is more processes try to write the same shared block
The larger the a block, the more chances of false sharing (i.e. it
occurs when two different process access two unrelated variables the
reside in the same data block) that causes throshing.
reside in the same data block) that causes throshing.
(1) Allow a process to prevent a block from accessed from the other
uning a lock.
(2) Allow a process to hold a block for a certain amount of line.
(2) Allow a process to hold a block for a certain amount of line. (3) Apply a different wherene algorithm to each block.
DISTRIBUTED FILE SYSTEM-

Distributed file rystem is a aenousce management component of a distributed operating system. It provides a use with a unified view of the files on the network. A machine that holds the shared files is called a server. A elemachine that access the files is called a clint.

Goals of distributed file systems one -

(1) Network transparency

(2) High avoilability.

2	Derriable features of good distributed file rystem -
	(1) Transparency -
	(1) Structure Transponency - Client should not know the number or
	weations of the file sever and storage devices.
	(ii) Access Transparency - Both was and remote file should be
	accemble in the same way.
	(iii) Maring Transparency - Name of the file should give no hint
-	as to where the file is located.
	(in Replication Transparency - Clients do not need to know the
	existence or locations of multiple file copies.
	(2) Usu mobility -
	User should not force to work on a specific node but should have
	the flenibility to work on different nodes at different types times.
	(3) Performance
	Average amount of time needed to ratisfy chient requests
	(4) Scalability
	(5) High availability
	(6) High reliatifity
	a) security.
	0
(3)	File models -
	File model are board on following criteria -
	(1) Unstructured and shurtines files -
	Simple file model - File is an unstructioned sequences of data.
	There is no substructure known to the file sever leg-UNIX, MSDOS
	Structured file model - A file appears to the file server as an ordered
. 5.	sequence of neurals. They are two types-
	Indened records - Records have one or more key frields and can be
	addrined by specifying the values of the key fields . Eg- B-tree, RSS, Droile.
	Mon-Indeand nevords - A file record is accessed by springing its position
	within the file. Eg - IBM mainframe
	/a/companion

Most of the modern operating system was the unstructed file model. This is mainly because thaning of a file by different applications is easier with the unstructured file model as compared to the shutting file model.

(2) Mutable and Immutable file -

Mutable files - An update in performed on a file overwrites on its own old contents to produce the new contents. File is represented as a ringle stored sequence that is altered by each update operations Immutable file - A file cannot be modified once it has been created encept to be deleted. File vernoning approach is used. Eg-ledon file system Problems - Increased use of dish space and disk allocation activity

File Scurice Architecture -			Scher
	APPLICATION PROGRAM	APLICATION PROLEMA	DIRECTORY SERVICE
	CLIE	NT	 FUNT FILE
		- 	
#-			

There are three components -

(1) Flat file service -

Conserved with the implementation of operations on the contests of file. Unique File Identifies (VFIDs) are used to refer to file in all regjests for flat file service operations

(2) Directory Service, -

9t provides mapping between tent names for the files and their UFIDS. Chient may obtain the UFID of a file by quoting its tent name to directory service. to directory service

(3) Chint module

It provides integrated service (flat file & duestory) as a single API to application programs. It holds information about the returns ingcompanion

	,
	location of flat-file and directory serves processes; and achieves better
	location of flat-file and directory serves process; and achieves better furformance through implementation of a cache of recently used file
	blocks at the client
_	Elat file service operations -
	1) Read (File Id, i, n) - Data - throws Bad Pointin -
	If 1≤i≤ Length (File): Reads a requeree of up to n îtems from a
	file starting at item i and returns it in Data
	(2) Write (File Id, i, 1 Data) - throws Boad Pontion -
	If 1 ≤ i ≤ Length (File) + 1 1 White a signence of Data to a file, starting
	at item i, entending the file is necessary.
-	(3) Create () - Filed -
	heates a new file of length O and delivers a UFID for it.
	(4) Delete (File Id) - Removes the file from the file store
	(5) Get Attributes (File Id) - Attr - Returns the file attributes for the file
	(6) SetAttributes (File Id, Attr) - Set the file attributes
-	Directory service operations -
	Thookup (Dir, Name) - FileId - throws Not Found -
	hocates the tent name in the directory and returns the relevant UFID.
(2) Add Name (Dir, Name, File Id) - throws Name Duplicate -
	If Name is not in the directory, adds (Name, File) to the directory and
	updates the files attribute record.
	(3) UnName (Dir, Name) - throws Not Found -
	If Name is in the directory, removes the entry containing Name from directory
	(4) Get Names (Dir, Pattern) - Name Seg -
	Returns all the tent names in the directory that match the regular enforcement
	fattern Pattern
4	UNIX file system uses access control, hierarchy file system and
	UNIX file system uns access control, hierarchy file system and file group (9t is a collection of files that can be located on any server or

moved between sewers while maintaining the rame names)

(3)	File Accerning Models -
	It depends on ture factors -
	(1) The method used for accuming remote files (2) The Unit of data according
	Accurring Premote files - Two mircles -
	(1) Remote Service model -
	Clients against for the file occess is delivered to the server,
	the sever machine performs the access request, and finally the
	result is forwarded book to the client Request one transferred as merray
	Merit - A simple implementation Demarit- Communication Dischard
	(2) Data Caching Model -
	If the data needed to rating the client's access request is not fresent locally, it is copied from the server's node to the client node and is cached there.
	present weally, it is copied from the server's node to the client node
	Muit - Reducing network traffic Dement - Cache connoting problem
4	Unit of Data Transfer - Four models -
	1) File level transfer model - Complete file is moved & - AFS, Amod
	Merits - Simple, less communication overhead and immune to server
	Dements - A chient required to have large storage space
	2) Block level transfer model - Units of file blocks are moved &-NF.
	Muits - A client not required to have large storage space.
	Dements - More retwork traffic overhead
	3) Brote level transfer model - Units of bytes are moved by - Cambridge of
	Ments - Flenchlity manininged
	<u>Demints</u> - Difficult cache management to handle the variable-length data
	(4) Record level transfer model - Units of records are moved.
	Meinto - Handling shurtised and indired files
	Dements - More network traffic and more overhead to re-construct of a file
- 11	the second of th

6	File Sharing Semanties -
·	Define when modifications of the life data made by a use and
	Observable by other wers.
	(1)UNIX Simantico -
· · · · · · · · · · · · · · · · · · ·	
	as an enclusive resource Contention by this winds image that is associated
	as an enclusive resource. Contention for this single image causes delays in user processes. It is used in centralized as single processor systems. Unin file notion incluments—
	Unin file system implements -
	(1) Write to an open file vinite undertable to other uses of the same
	(1) Write to an open file visible unchediately to other uses of the same open file
	(2) Sharing file pointer to allow multiple usus to read and write concurrently
	(2) Sernion Semantico -
	A file can be arrainted with multiple news Almost no constraints
	are imposed on scheduling access. No user is delayed in reading or writing
	their personal copy of the file. Eg - Andrew File System (AFS)
	(3) Immutable Shared file semanties -
	(3) Immutable Shared file semanties - No upolatis one pomible; simplifies sharing and replication (4) Transactions like semanties -
	All changes occur atomically. Begin transactions, perform
-	operations and end transaction. The final file content is the rame as if all
	tramactions were run in some sequential order.
	File Caching Schemes
	It also address the Idlantic by allowing disk transfer.
	It also address the following key decisions -
	(1) tache location
	(i) Modification propagation
	(11) Cache validation.

Cache breation - Cache behaves girst like "returned membry" (Denvis main memory - Cost of cache mis = cost of acres dish + returns history Cost of cache mis = cost of acres dish + returns history Cost of cache het = Metunde transfe. Merit - One-time disk acres, sany implementation, Union like file thanks genometric. Denvis - Bury Network traffic. (2) Chimits disk - Cost of cache het = Time to access from local chiele. Merits - One-time Network Acres, No trize restrictions, buildful for sufficiently disconnected of sea time. Denvis - Coche consistency problem, file access semantics, Frequent disk access, No diskless winkington. (3) Chimits Main Memory - Cost of cache hit = maximum (high performance) Ments - Maumium Reformance, Institus workstation, scalability. Denvis - Maximum Reformance, Destruction y quotien, file access semantics. Mordification Propagation - The arm is keeping file data cached at multiple cleint node committent. It has a cultical effect on the tystims performance and reliability. The file semantics suffered defends greatly on the modification frequention scheme used. Coshe update feeling - (1) Write through - When a cache entry is modified, the runs value is invalidately sent to the sures for updating the original copy of the file. Pros - Union like semantics and high reliability Ono - Poor write performance. (2) Delayed Write - The aim is to addince networks traffin for write		Cost of course hit - Network transfer
Cache behaves just like "networked initial membry" (Wenter main memory— Cost of cache miss = cost of access din + retwork borness. Cost of cache het = Metwork borness. Ments - One-time dick access, Easy implementation, Union like file thaving semantics. Dements - Bury Network traffic. (2) Chinats dick - Cost of cache het = Time to access from break dick. Miss - One-time Network Access, No size sesticition, buildule for inflicting disconnected of the borness. No disconnected of the borness. Dements - Coche consistency problem, file access semantics, Frequent dick access, No dischession whether workstation. (3) Chinats Main Memory— Cost of cache het = manionum (high performence) Meits - Maunium Reformance, Institute workstation, feel access semantics. Modification Propagation— The aum is keeping file data cached at multiple cleint node comitish. The file semantics inflorted at periods questly on the modification propagation returne wood. Contentification propagation returns used. Coste update pastery— (1) Write through - When a cache entry is modified, the rew value is invadiately sent to the sever for updating the original copy of the file. Pasa - Union like semantics and high reliability. Cons - Poor write performance.		Cost of course hit - Neturelo transfer
Cache behaves just like "networked initial membry" (Wenter main memory— Cost of cache miss = cost of access din + retwork borness. Cost of cache het = Metwork borness. Ments - One-time dick access, Easy implementation, Union like file thaving semantics. Dements - Bury Network traffic. (2) Chinats dick - Cost of cache het = Time to access from break dick. Miss - One-time Network Access, No size sesticition, buildule for inflicting disconnected of the borness. No disconnected of the borness. Dements - Coche consistency problem, file access semantics, Frequent dick access, No dischession whether workstation. (3) Chinats Main Memory— Cost of cache het = manionum (high performence) Meits - Maunium Reformance, Institute workstation, feel access semantics. Modification Propagation— The aum is keeping file data cached at multiple cleint node comitish. The file semantics inflorted at periods questly on the modification propagation returne wood. Contentification propagation returns used. Coste update pastery— (1) Write through - When a cache entry is modified, the rew value is invadiately sent to the sever for updating the original copy of the file. Pasa - Union like semantics and high reliability. Cons - Poor write performance.		
Cache behaves just like "networked initial membry" (Wenter main memory— Cost of cache miss = cost of access din + retwork borness. Cost of cache het = Metwork borness. Ments - One-time dick access, Easy implementation, Union like file thaving semantics. Dements - Bury Network traffic. (2) Chinats dick - Cost of cache het = Time to access from break dick. Miss - One-time Network Access, No size sesticition, buildule for inflicting disconnected of the borness. No disconnected of the borness. Dements - Coche consistency problem, file access semantics, Frequent dick access, No dischession whether workstation. (3) Chinats Main Memory— Cost of cache het = manionum (high performence) Meits - Maunium Reformance, Institute workstation, feel access semantics. Modification Propagation— The aum is keeping file data cached at multiple cleint node comitish. The file semantics inflorted at periods questly on the modification propagation returne wood. Contentification propagation returns used. Coste update pastery— (1) Write through - When a cache entry is modified, the rew value is invadiately sent to the sever for updating the original copy of the file. Pasa - Union like semantics and high reliability. Cons - Poor write performance.		Cache Weation -
Cost of cache mis = cost of access dish + returns borning. Cost of course het = Network transfer. Menit - One-time disk access, Easy implementation, Union like five training semantic. Dement - Bruy Network traffic. (2) Chiefs disk - Cost of cache hit = Time to access from weal chiefs. Menits - One-time Metwork Access, No rize sestivitions, switable for infloring disconnected office little. Menits - One-time Metwork Access, No rize sestivitions, switable for infloring disconnected office little consistency. Demenits - One-time Memory - Cost of cache wit = maximum (high performance) Menits - Maximium Performance, Inseless weak station, fealability. Demenits - Frize restriction, Cache committency problem, file access semantics. Modification Perforantion - The arm is keeping file data cached at multiple cleint node consistent. The file semantics enflorted depends greatly on the node is insultipleation perforagation suffers and depends greatly on the node factor perforate perforage toin scheme wood. Cache upotate perforage time scheme wood. Cache upotate perforage time scheme wood. Cache upotate perfory - (1) Write through - When a cache entry is modified, the new value is insultably sent to the sewer for upotating the original copy of the file. Proc - Union like semantics and high reliability. Cons - Poor write performance.	*	Cache behaves just like. "networked initial membry"
Cost of cache hit = Network trongs Menit - One-time die access, Easy implimentation, Union like file Though smoothis. Dement - Bury Network traffic. (2) Chints disk. Cost of cache hit = Time to access from boal chike. Menits - One-time Network Access, No uze restriction, histolia for infloiting disconnected of water. No access, No diskless unditation. Borneits - Cache consistency problem, file access semantics, Frequent dish access, No diskless unditation. (3) Chints Main Menay - Cost of cache hit = maximum (high performence) Merits - Maximum Reformance, Diskless work station, fealability Dements - Frize restriction, Cache consistency problem, file access semantics Modification Propagation - The aim is keeping file data cached at multiple claint node (consistent 9t has a califical effect on the hystimis performance and reliability. The file semantics sufferted depends greatly on the modification perpagation returns enforted depends greatly on the modification perpagation returns and ache into the perpagation within a cache entry is modified, the new value is inmutiately sent to the seven for updating the original copy of the file. Pros - Union like semantics and high reliability Cons - Pron write performance (2) Debayed Write - The aim is to aduce networks thaffin for write		(1) Server's main memory -
Merit - One-time disk access, Easy implementation, Unix like file Menuty - Brusy Network traffic. 2) Chiests disk - Cost of cacke hit = Time to access from local chiete. Merits - One-time Metwork Access, No size sestivition, heitable for infforting disconnected of water. No size sestivition, heitable for inflorting access, No discless consistency problem, file access semantics, Frequent dish access, No discless consistency. (3) Chiests Main Menay - Cost of cache hit = manimum (high performance) Merits - Maunium Reformance, Institus work station, fealability Denneits - Frize restriction, Cache consistency problem, file access semantics Modification Propagation - The arm is keeping file data cached at multiple cleint north consistent. Its file semantics suffered on the systems preformance and reliability. The file semantics suffered defends greatly on the modification propagation without defends greatly on the modification propagation without acche entry is modified the new value is immediately sent to the seven for updating the original copy of the file Pass - Unim like semantics and high reliability Cons - Poor write preformance (2) Debayed Write - The aim is to reduce network thaffire for write		Cost of cache miss = cost of access dish + network hornings
Menits - One-time disk access, Easy implementation, Unin like file Whating senantics. Dements - Brown Network traffic. (2) About disks - Cost of cacke hit = Time to access from local clocks. Menits - One-time Network Access, No size restriction, huitable for hipporting disconnected operation. Parments - Cacke consistency problem, file access semantics, Trequest disk access, No diskless workstation. (3) Chiest Main Menay - Cost of cacke hit = maximum (high performance) Murits - Mauritim Reformance, Institute workstation, scalability. Demailts - Frze restriction, Cache consistency problem, file access semantics. Modification Propagation - The aim is keeping file data cached at multiple cluint node comistent. It has a cultical effect on the hystoms performance and reliability. The file semantics sufferted depends quartly on the modification propagation scheme used. Coache applate policy - (1) Write through - When a cache entry is modified, the new value is invadiately synt to the sever for applacting the original copy of the file. Pros - Union dike semantics and high valuability Cono - Poor write performance. (2) Debayed Write - The aim is to reduce retwork traffic for write	_	Cost of whe hit = Neturns transfer
Dements - Buny Notwork traffic. (2) Chint's disk - Cost of cache hit = Time to access from brad chish. Ments - One-time Network Access, No sign restriction, builtable for sufficiency disconnected operation. Dements - Coche consistency problem, file access semantics, Frequent disk access, No diskles workstation. (3) Chint's Main Menay - Cost of cache hit = maximum (high performance) Ments - Maximum Reformance, Institute growthem, file access semantics. Morlification Propagation - The aim is keeping file data cached at multiple chint node consistent. It has a cultival effect on the systems performance and reliability. The file semantics sufferted depends greatly on the modification propagation returns sufferted depends greatly on the modification propagation returns and. Cache update policy - (1) Write through - When a cache entry is modified, the new value is immediately sent to the saver for updating the original copy of the file. Pros - Unic discussmantics and high velocities. Cora - Poor write performance.		Merits - One-time disk access, Easy implementation, Unin like file
Cost of cache hit = Time to access from weal choise. Ments - One-time Network Access, No rize restriction, britable for sufficiency disconnected opens from. Betweents - Cache consistency problem, file access semantics, Frequent disk access, No diskless workstation. (3) Clients Main Meany - Cont of cache hit = manimum (high performance) Ments - Maumium Reformance, Brokens workstation, scalability. Demnits - Frize restriction, Cache consistency problem, file access semantics Movelification Propagation - The aim is Recepting file data cached at multiple client node committent. It has a cultical effect on the systems performance and reliability. The file semantics sufferted depends greatly on the modification perfaction scheme und. Cache update policy - (1) Write through when a cache entry is modified, the new value is immediately sent to the sever for updating the original copy of the file. Tros - Unia dike semantics and high reliability Cons - Poor write performance.		Sharing semantis.
Cost of cache hit = Time to accus from boal clock. Menis - One-time Network Access, No size restriction, buitable for sufficiting disconnected of walter. Between - Costa consistency problem, file access sumenties, Frequent disk access, No diskles workstation. (3) Claints Main Menay - Cost of cache hit = maximum (high performance) Menit - Maumium Reformance, Problem workstation, scalability Demuit - Frize restriction, Cache consistency problem, file access remembries Mordification Propagation - The aim is keeping file data cached at multiple claint node consistent. It has a cultival effect on the systems haformance and reliability. The file semantics sufferted depends greatly on the modification propagation rehene und. Cache update fullay - (1) Write through - When a cache entry is modified, the new value is immediately sent to the sever for updating the original copy of the file: Pros - Union like semantics and high reliability. Cons - Poor write performance.		· . · · · · · · · · · · · · · · · · · ·
Bernerits - Cache comintary problem, file accers semantics, Frequent dish access, No diskles workstation. (3) Christi Main Memory - Cost of cache hit = manimum (high performance) Ments - Maumium Performance, Diskles workstation, scalability. Dennuits - Size restriction, Cache consistency problem, file access semantics Mordification Propagation - The aim is keeping file data cached at multiple cleint node committent. It has a cultical effect on the systems performance and reliability. The file semantics sufferted depends greatly on the modification propagation rehene und. Coche update policy - (1) Write through - When a cache entry is modified, the new value is immediately sent to the sever for updating the original copy of the file. Page - Unin like semantics and high reliability Cons - Poor write performance. (2) Debayed Write - The aim is to reduce network thaffin for write		(2) Client's disk -
Bernerits - Cache comintary problem, file accers semantics, Frequent dish access, No diskles workstation. (3) Christi Main Memory - Cost of cache hit = manimum (high performance) Ments - Maumium Performance, Diskles workstation, scalability. Dennuits - Size restriction, Cache consistency problem, file access semantics Mordification Propagation - The aim is keeping file data cached at multiple cleint node committent. It has a cultical effect on the systems performance and reliability. The file semantics sufferted depends greatly on the modification propagation rehene und. Coche update policy - (1) Write through - When a cache entry is modified, the new value is immediately sent to the sever for updating the original copy of the file. Page - Unin like semantics and high reliability Cons - Poor write performance. (2) Debayed Write - The aim is to reduce network thaffin for write		Cost of cache het = Time to acces from weal dish.
Bernerits - Cache comintary problem, file accers semantics, Frequent dish access, No diskles workstation. (3) Christi Main Memory - Cost of cache hit = manimum (high performance) Ments - Maumium Performance, Diskles workstation, scalability. Dennuits - Size restriction, Cache consistency problem, file access semantics Mordification Propagation - The aim is keeping file data cached at multiple cleint node committent. It has a cultical effect on the systems performance and reliability. The file semantics sufferted depends greatly on the modification propagation rehene und. Coche update policy - (1) Write through - When a cache entry is modified, the new value is immediately sent to the sever for updating the original copy of the file. Page - Unin like semantics and high reliability Cons - Poor write performance. (2) Debayed Write - The aim is to reduce network thaffin for write	-	Ments - One-time Network Access, No rize restriction, Suitable for sufficiting
(3) Cluints Main Menoy - Cont of cache hit = manimum (high performance) Ments - Manimum Reformance, Inteless work tation, scalability Demnits - Size restriction, Cache commitmey problem, file access remantics Modification Propagation - The aim is keeping file data cached at multiple client node committent. It has a critical effect on the systems performance and reliability. The file remantics sufferted depends greatly on the modification propagation scheme und. Coche update policy - (1) Write through - When a cache entry is modified, the new value is immediately sent to the seven for updating the original copy of the file: Pros - Unin like remarities and high reliability Cons - Poor write performance. (2) Delayed Write - The aim is to reduce returns traffic for write		
(3) Cluints Main Menoy - Cont of cache hit = manimum (high performance) Ments - Manimum Reformance, Inteless work tation, scalability Demnits - Size restriction, Cache commitmey problem, file access remantics Modification Propagation - The aim is keeping file data cached at multiple client node committent. It has a critical effect on the systems performance and reliability. The file remantics sufferted depends greatly on the modification propagation scheme und. Coche update policy - (1) Write through - When a cache entry is modified, the new value is immediately sent to the seven for updating the original copy of the file: Pros - Unin like remarities and high reliability Cons - Poor write performance. (2) Delayed Write - The aim is to reduce returns traffic for write		Dements - Cache consistency problem, file acces smantics, Frequent dish
Muits - Maumium Performance, Inteles work Astron, fealability Demnit - Proje restriction, Coche consistency problem, file access semantics Modification Propagation - The aim is keeping file data eached at multiple client node committent. It has a critical effect on the systems performance and reliability. The file semantics sufferted depends greatly on the modification propagation scheme used. Coche update policy - (1) Write through - When a cache entry is modified, the new value is immediately sent to the seven for updating the original copy of the file. Pros - Unin like semantics and high reliability Cons - Poor write performance. (2) Delayed Write - The aim is to reduce returns traffic for write		access, No diskless uniteration.
Muits - Maumium Performance, Inteles work Astron, fealability Demnit - Proje restriction, Coche consistency problem, file access semantics Modification Propagation - The aim is keeping file data eached at multiple client node committent. It has a critical effect on the systems performance and reliability. The file semantics sufferted depends greatly on the modification propagation scheme used. Coche update policy - (1) Write through - When a cache entry is modified, the new value is immediately sent to the seven for updating the original copy of the file. Pros - Unin like semantics and high reliability Cons - Poor write performance. (2) Delayed Write - The aim is to reduce returns traffic for write	1	(3) Chints Main Menoy -
Meriti- Manmum Reformance, Inteless workstation, scalability Densuit - Sozie restriction, Cache consistency problem, file access semantics Modification Propagation — The aim is keeping file data cached at multiple client node consistent. It has a cultical effect on the systems performance and reliability. The file semantics sufferted depends greatly on the modification propagation scheme used. Cache update policy— (1) Write through — When a cache entry is modified, the new value is immediately sent to the sever for updating the original copy of the file. Pros - Union like semantics and high reliability. Cons — Poor write performance.		Cost of cache hit = manimum (high performance)
Modification Propagation— The aim is keeping file data cached at multiple client node committent. It has a cultical effect on the systems performance and reliability. The file semantics sufforted depends greatly on the modification people who rehere used. Cache update policy— (1) Write through— When a cache entry is modified, the new value is immediately sent to the sever for updating the original copy of the file. Paos—Union like semantics and high reliability Cons—Poor write performance. (2) Delayed Write—The aim is to reduce network traffic for write		Merits - Manumum Cenformance, Insteles workstation, Scalability
Modification Propagation— The aim is keeping file data cached at multiple client node committent. It has a cultical effect on the systems performance and reliability. The file semantics sufforted depends greatly on the modification people who rehere used. Cache update policy— (1) Write through— When a cache entry is modified, the new value is immediately sent to the sever for updating the original copy of the file. Paos—Union like semantics and high reliability Cons—Poor write performance. (2) Delayed Write—The aim is to reduce network traffic for write		Dennit - Sozie restruction, Cache consistency problem, file acces remanties
The aim is keeping file data cached at multiple client node committent. It has a cuitical effect on the systems performance and reliability. The file semantics sufforted depends greatly on the modification people on rehere used. Cache update policy— (1) Write throughs— When a cache entry is modified, the new value is immediately sent to the sever for updating the original copy of the file. Pros - Union like semantics and high reliability. Cons - Poor write performance. (2) Delayed Write - The aim is to reduce networks traffic for write.	1	
committent. It has a critical effect on the systems performance and reliability. The file semantics sufferted depends greatly on the modification people of whene used. Cache update policy— (1) Write through— When a cache entry is modified, the new value is immediately sent to the sever for updating the original copy of the file— Pros—Unin like semantics and high reliability Cons—Poor write performance. (2) Delayed Write— The aim is to reduce network traffic for write		1
nodification peopagation returns und. Coche update policy— (1) Write through— When a cache entry is modified, the new value is immediately sent to the sever for updating the original copy of the file. Pros—Unin like semanties and high reliability. Cons—Poor write performance. (2) Delayed Write— The aim is to reduce network traffic for write.		The aim is keeping file data cached at multiple client node
nodification peopagation returns und. Coche update policy— (1) Write through— When a cache entry is modified, the new value is immediately sent to the sever for updating the original copy of the file. Pros—Unin like semanties and high reliability. Cons—Poor write performance. (2) Delayed Write— The aim is to reduce network traffic for write.		committent. It has a cultival effect on the system's performance and
Coche update policy - (1) Write through - When a cache entry is modified, the new value is immediately sent to the seven for updating the original copy of the file. Pros - Unin like semantics and high reliability Cons - Poor write performance. (2) Delayed Write - The aim is to reduce network traffic for write		reliablety. The file semantics sufferted defends greatly on the
(1) Write through - When a cache entry is modified, the new value is immediately sent to the seven for updating the original copy of the file. Pros - Unin like semantics and high reliability. Cons - Poor write performance. (2) Delayed Write - The aim is to reduce network traffic for write.		modification propagation rehene und.
(1) Write through - When a cache entry is modified, the new value is immediately sent to the seven for updating the original copy of the file. Pros - Unin like semantics and high reliability. Cons - Poor write performance. (2) Delayed Write - The aim is to reduce network traffic for write.		Coche update policy -
is immediately sent to the sever for updating the original copy of the file Pros - Union like semantics and high reliability Cons - Poor write performance. (2) Delayed Write - The aim is to reduce returns traffic for write		(1) Write through - When a cache entry is modified, the new value
Cons - Poor write performance. (2) Delayed Write - The aim is to reduce network traffic for write	-	is immediately sent to the seven for updating the original copy of the file
(2) Delayed Write - The aim is to reduce network traffic for write		Pros - Unin like semanties and high reliability
(2) Delayed Write - The aim is to reduce network traffic for write when a cache entry is modified, the new value is written only to the cache		Cons - Poor unite performance.
When a cache entry is modified, the new value is written only to the cache		(2) Delayed Write - The aim is to reduce network traffic for write
(m/companion	-	when a cache entry is modified, the new value is written only to the cache

and elint just makes a note. <u>Pros</u> - Write accesses complete quickly, some writes may be smitted by the following writes, Gothering all writes mitigales network overhead. Cons - Delaying of write propagation results in fuzzier file-sharing semantics. -> Cache Validation -Cache conniting is another important ince that deturnines how caches are maintained when multiple clients may be accoming the same file Client - initiated approach -Client is responsible for checking with the server to verify that each file in its cache is connitent. A single cossept or malicious client could disrupt the complete Sower-initiated approach -Server acts as a central authority over which clients have up to date or invalid caches. Sewer is able to detect when reading and writing chints might conflict with each other, and will send menages to client to force them to invalidate their cache enthis and request them again. Concurrent unite thoung approach
A file is open at multiple clients and at least one client has its
open for writing. File serves keep track of the clients sharing a file. 3 File Replication A replicated file is a file that has multiple copies, with each file on a reparate file surer Advantages of replication -(5) Improved system throughput (6) Better realability (1) Increased Availability (2) Increased Reliability (3) Improved Response time

(4) Reduced Neturn traffic

Diffuente between Replication and Caching -

(1) A replica of a file is associated with sever, whereas a cashed

copy is normally associated with a client

(2) The enistince of a cached copy is primarily dependent on the locality in file access patterns, whereas the enistence of a replica normally depends on availability and performance requirements.

(3) As companied to cached copy, a replica is more personent, widely known, secure, available, complete and accurate.

(4) A cached copy is contingent upon a replica. Only by periodic revalidation with respect to a replica can a cached copy be wreful.

Replication Transparency -

Replication of files should be transponent to the uses so that multiple copies of a replicated file affect as a single ingical file to its uses. This calls for the assignment of a ringle identific / home to all replicas of a file.

In addition, replication control should be transforment i.e. the number and weathers of replicas of a replicated file should be hidden from the user. Thus replication control must be hondled automatically in a user-transforment manner.

Multicopy Update Problem -

Major denge issue of a distributed file system that supports file replication. To separe to avoid this problem, the can use -

- (1) Read only replication protocol
- (2) Read-cony-write-all protocol.
- (3) Available copies protocol Same as (2) protocol but writes to all available copies of the file
- (4) Primary Copy protocol Read operations can be performed using any copy, primary or secondary but write operations are performed only on the primary copy. Each server having a recordary copy updates its why.

9 Fault Tolerance -The approach of fault-tolerence expect faults to be meant during nystim operation, but employs design techniques which insure the continued correct enecution of the computing process The frimary file properties that directly influence the ability of a distributed file system to tobrate faults are as follows: (1) Availability (2) Robertness (3) Recoverability (hower to survive crashes) Stable Storage -Information never lost. Not actually possible, so approximated via replication or RAID to devices with independent failure modes. Iwo Vane Oferation -

(1) A read operation first attempts to read from dish 1. If it fails, the read is done from disk 2.

(2) A write operation writes to hold disks, but the write to disk 2 does not start until that for dish! has been succenfully completed

It is nutoble for applications that require high degree ofor fault tolerence & - Stomi transactions

Effects of Service Paradigm on fault tolerance. The file service that implement a distributed file service can be stateless or stateful

4) Stateful file server -

Server maintains information about a file opened by a client that means it store session state. File operations sufficiely by this server one -(1) Open (filename, mode)

- (2) Read (fid, n, buffer)
- (3) Write (fid, n, buffer)
- (1) Seek (fid, position)
- (5) Close (fid) cours the server to delete from its file table the file state information of the file identified by fid

3	Adventages of Stables severs - Advantages of Stateful severs -
	(1) Fault Tokrane. (1) Shorter request merages
	(2) NO OPEN/CLOSE calls needed (2) Better fur formance
	(3) NO sewer space wasted on lables (3) Resident points.
11	(4) No limits on number of open files a Gelempotency carrier
1	(5) No problem if a client caselier (5) File locking possible
	Diradientogis of stateful servers -
ł	(1) Problem of orphan delection of elimination
	(2) It loss all its volatile state in a crash
_	
I	1) her herbonnesse
,	2) There is no longer sequent memore and some horizon of anything
	2) There is no longer request menages and slower processing of request.
1	2) Statiling file years
4	2) Statiles file serves -
	serves maintains no information about client access to files that
	means it do not store any sernen state File operations are given as -
1	1) Read (filename, pasition, n, bufter)
_(21 Write (filename, position, n, butter)
-	
5	nample of state ful file server -
	Client Process Open (Flename, mode) Server Process
	Return (Fid) File Table Fid Mode R/W pointer
	Redd (Fid 100, but)
	Return (bytes 0 to 59)
	2
_	Enample of stables file server -
T	Client Process Server Process
+	File State Information Read (filename, 0, 100, but)
+	Fid Mode R/W pointer (Return (bytes 0 to 99)
4	
	<i>my</i> companion

	Naming -
	The maning facility (amon character-string names to objects) and
	locating facility (maps an objects name to the objects location) jointly
	form a naming rystem that provides the users with an abstraction
	of an object that hides the details of how and where an object is actually
	located in the network.
2	Torniable features of a good naming rystem -
	(1) hocation Transporancy (6) Group Naming
	@ horation Independency (7) Meaningful Names
	(3) Scalability (8) Performance
	(4) Uniform Naming Convention (9) Fautt Tolerance
	(5) Multiple user defined names for the rame object (10) Replication Transposerry
(3)	System Priented Names - (low level names)
	They are of fined rize bit pattern that can easily manipulate and
	stoud by machines. It banially meant for use by the system but may also
	be used by the user.
	Characteristics -
	(1) They are larger integers or hit strings.
	(2) Also reffered referred as unique identifier
	(3) height is variable
	(41) Automatically generated
	(5) They are hard to guen and provide good security.
	(6) They are suitable for efficient handling by machines.
-	Centralized approach for generating system-oriented names -
	It generates structured and unshurtured names. A Handard & uniform
	global identifier name is generated for each object in the nortem by a
	Centralized global unique identifier generator.
	UNSTRUCTURED NAMES STRUCTURED NAMES
	A single field of large integers on hit strings Node Industrifier horal Unique Edentific
	Mycompanion

	Advantages -
	(1) Simple and lang to implement
	(1) simple and lary to implement (2) Only method used for generating unstructured global unique identifies.
	Disadvantages
	(1) Poor efficiency and hour reliability
	(1) Pour efficiency and pour reliability (2) Single global unique identifier generator may become a bottleneck for
	large name space.
_	Distributed approach for generating history-oriented names -
	Hierarchical consatination, method is used to weater challed
	Distributed approach for generating rystem-oriented names - Hierarchical concatination method is used to create global unique identifiers by concatenating the unique identifier of a domain. Advantages -
	Advantages -
	(1) Better efficiency and reliability than centralized approach.
	(2) Node boundaries or servers are emplicitly vinible. (2) The form and length of identifier may be different for different comput resulting in non-uniform global images identifier.
	resulting in non-uniform global unique identifice
	Object Locating Mechanisms -
	It is the process of mapping an object's system-oriential unique
	edentifier to the replica locations of the object. Various Types are-
	(1) Broadcasting - Eg - Amoeta
_	A request is broadcast which is processed by all wides and then the nodes currently having the object reply back to the cleint wide.
	then the nodes currently having the object reply back to the cleant and
	Advantages - simple, high reliability.
	Induntages - Pour efficiency, pour scalability, number of node should
_	small, nieded high communication speed.
	(2) Enfanding Ring Broadcast -
	Modified form of twodcasting method. It wornists of LAN
	Modified form of twodcasting method. It wornsts of LAN connected by gateways. The distance metric used is a hop. A hop

conerfonds to a gativay between processors

	A name space is a collection of names which may or may not
	share an identical resolution muchanism.
	Ghalle and a market and a second
	It supplies reasest replica location but not recurainly all
	replica locations
	(3) Encoding water of object within its VID-
	(3) Encoding location of object within its VID- It was structured object identifies. It is straightforward
	and efficient scheme.
	himitation -
	(1) It is not clear how to support multiple replicas of an object
	(2) An object is not permitted to move once it is assigned to a node
	(3) An object is fined to one node Unoughout its lifetime.
14 × ×,	One robution is to use forward location pointers but it increases object
	localing cost, additional system overhead and difficult to brute if an intumediate pointer has been lost.
, ,,,,,,,	The way ton
(S)	Human-Oriented Names - (High level Names)
	The annual a shows to district the state of
	It is generally a character string and it is meaningful and identified bytits user. Not unique for an object and are normally variable in length.
	countified by its user. Not unique for an object and are normally
	4
	They cannot be easily manipulated, stoud and used by the
	machines for identification purpose.
	Characteristis -
	(1) They are defined and used by the user
	(2) Different users can define and me thui own mitable names for a
-	- tives of the
	(3) Due to the facility of alianing the same way have may be used
	(3) Due to the facility of alianing, the same time to refer to two different objects
	Mycompanion