

MANAGEMENT OF INFORMATION SYSTEMS

BME VIK TMIT
SOFTWARE ENGINEERING, BSc



BME VIK TMIT

MANAGEMENT OF INFORMATION SYSTEMS

6. BACK-UP AND RESTORATION



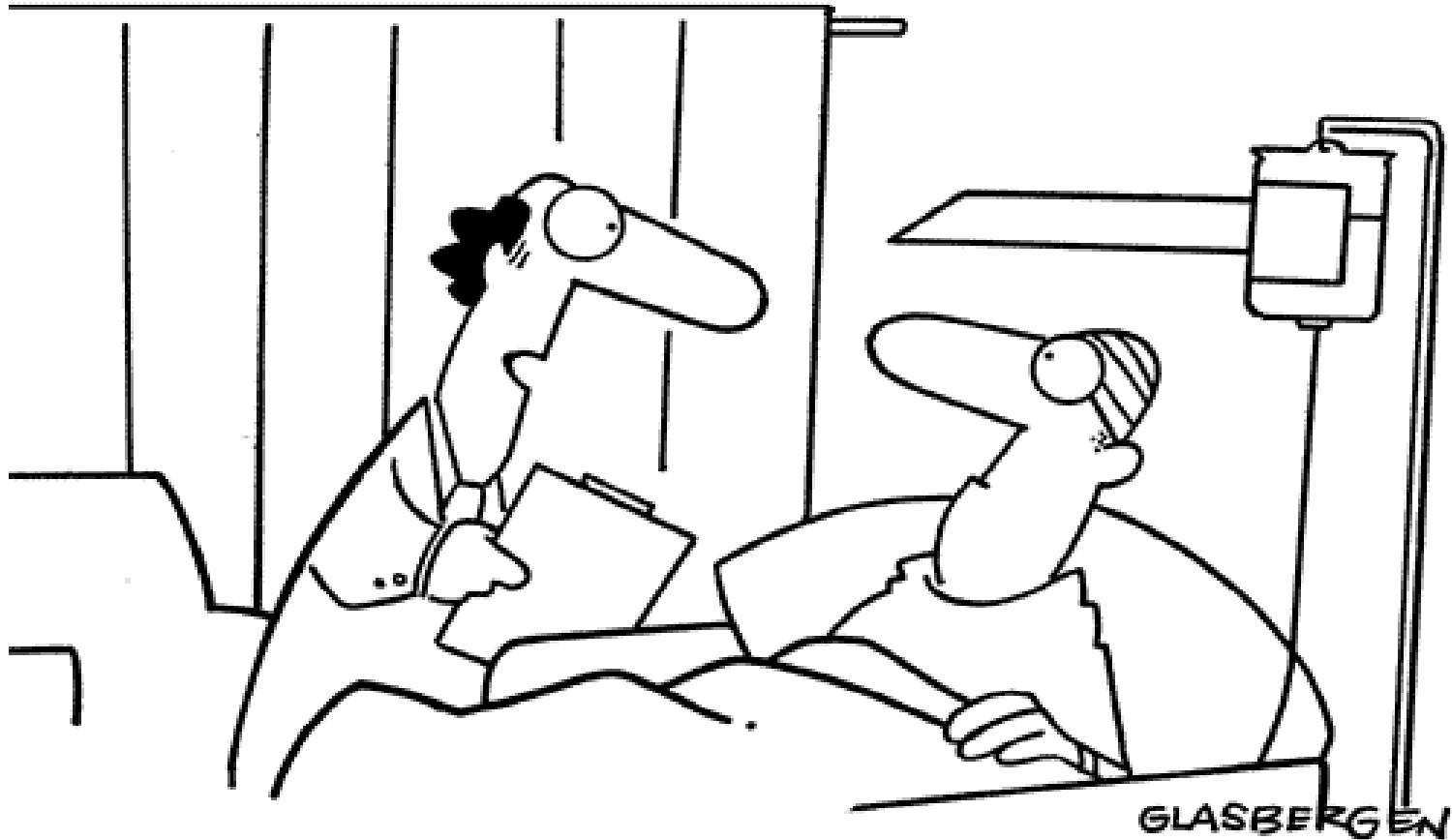
BACK-UP AND RESTORATION

- Back-up / archiving definition
- Back-up
 - Tape
 - Back-up systems
 - Back-up methods
 - Full, incremental, differential, progressive
 - Version control systems – Flash Copy
- Archives
 - Archivation requirements
- Design of back-up
 - Example
- Restoration



BACK-UP AND ARCHIVE

© 1999 Randy Glasbergen. www.glasbergen.com



“You caught a virus from your computer and we had to erase your brain. I hope you kept a back-up copy.”



BACK-UP AND ARCHIVATION

Common goal of back-up / archivation: recovery safeguard, make copies for preventing data loss.

Goal of back-up: copy of data that can be used to (quickly) restore the original in the event that your data is lost or damaged, to guarantee business continuity

- Unintentional file deleting-modification: the user accidentally / intentionally deletes/overwrites files
- HW Failure, disaster: storage device/system fails

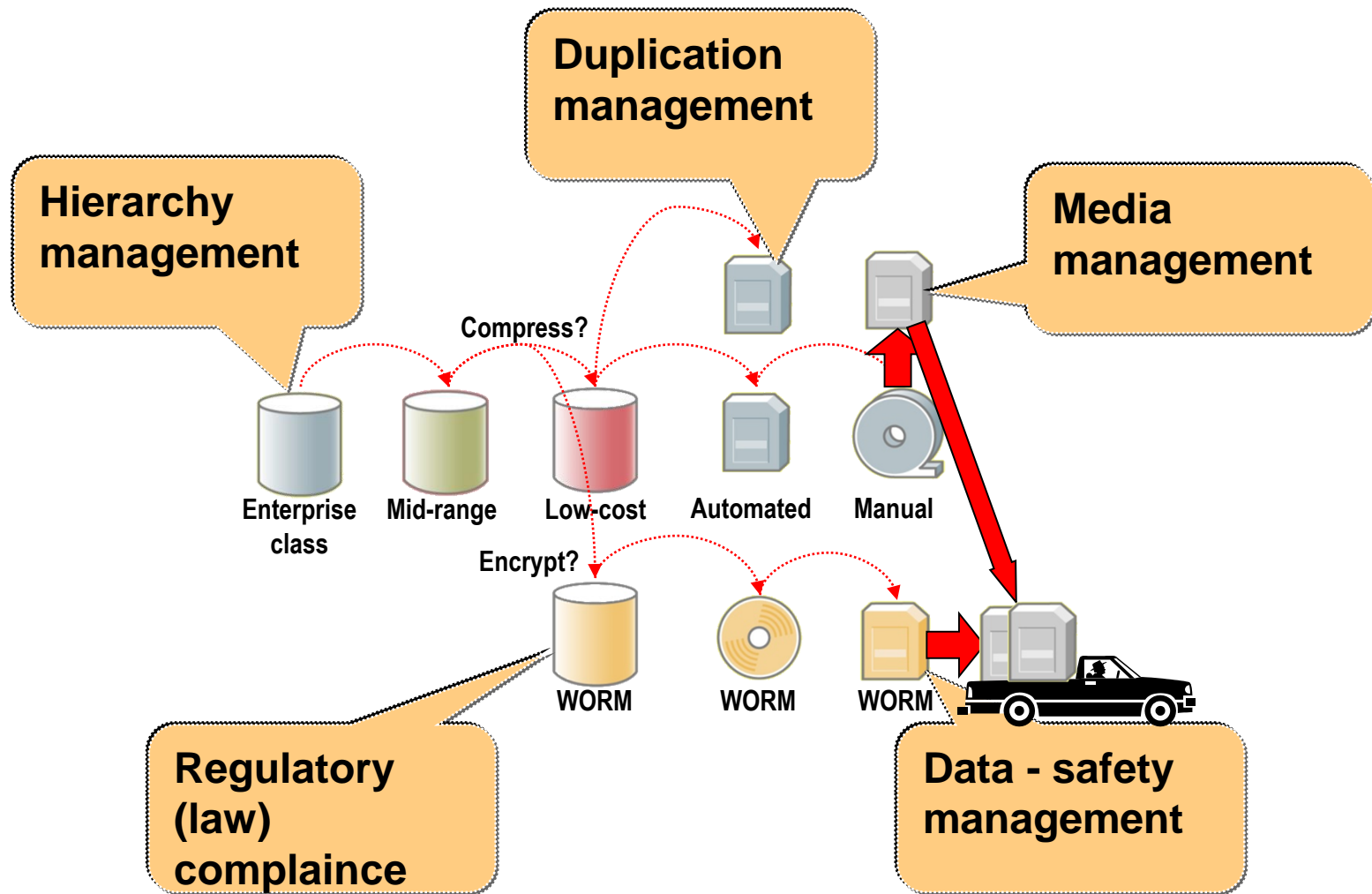
Goal of archivation: to reproduce the entire data of a certain point in time. An archive is a collection of historical records that are kept for long-term retention and used for future reference. Typically, archives contain data that is not actively used

- Business, legal etc. reasons: data serves as an evidence, basis for a comparison, reference
- Not used data to be deleted: maintenance / evidence(!) / legal(!) reasons

Typically common base technology for back-up/archivation

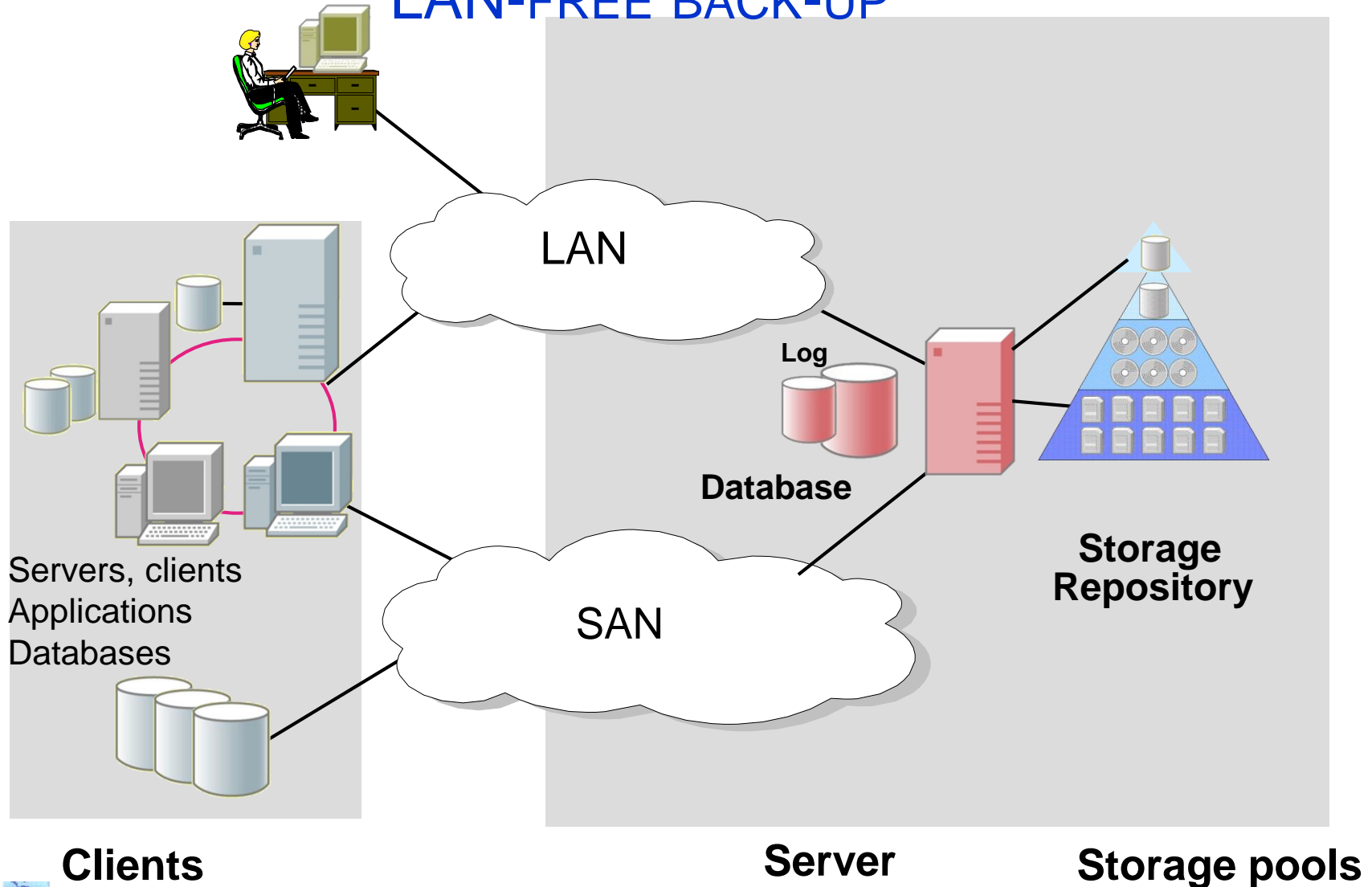


MODERN STORAGE SYSTEM REQUIREMENTS



STORAGE SYSTEM ARCHITECTURE

LAN-FREE BACK-UP



Clients

Server

Storage pools



LAN-FREE BACK-UP AND RECOVERY

- LAN-Free data transmission
 - (Back-up) Server manages the internal storage pool
 - The back-ups are typically scheduled (policies)
 - But the client can also initiate data movement / back-up from disk to tape, or to a SAN disk
 - Meta-data (from/to where, how much) transmitted on LAN
 - The back-up data is sent on SAN
 - LAN is not overloaded by heavy data traffic

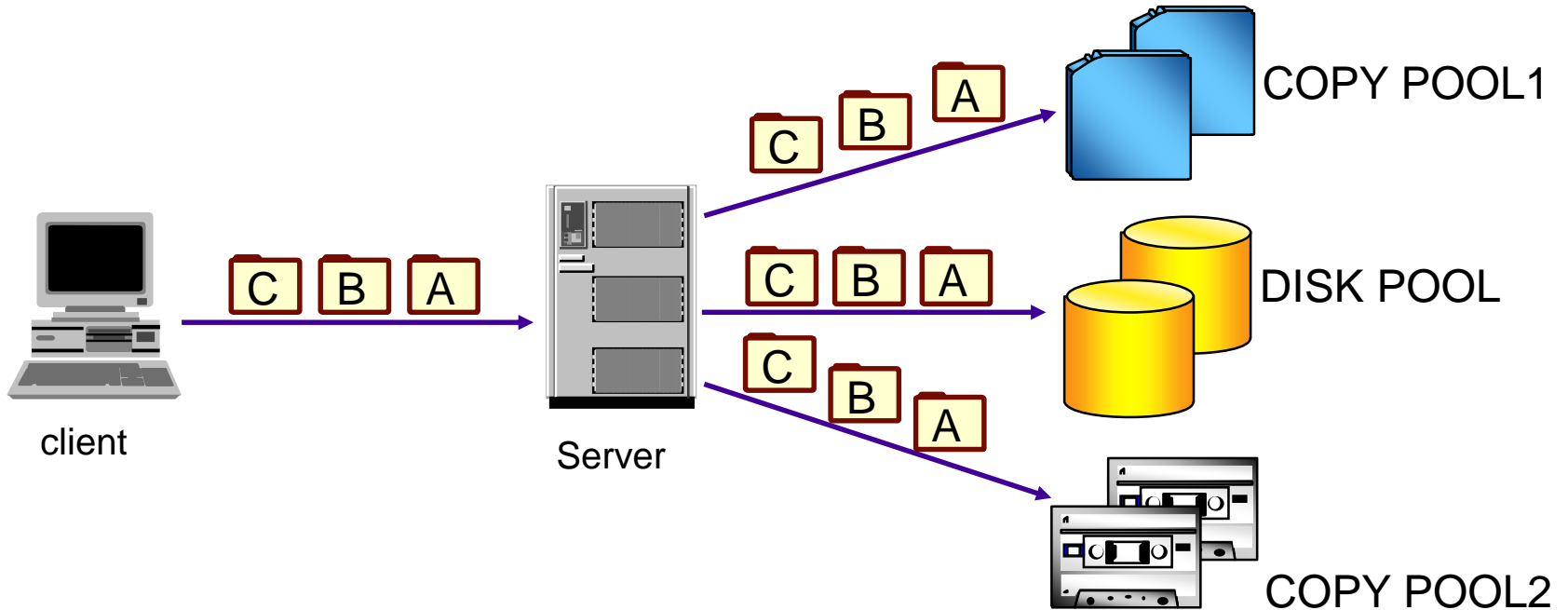


DATA COPYING IN A DISK SYSTEM

- Copy (very) large amount of data
- Block level copies
- Fast, firmware-supported back-up processes
- *Volume copies* – real copies of disks
 - Real duplicated file, cloning
 - Real copy for back-up, analytic, datamining etc.
 - Possibility of migration to other disk types



PARALLEL BACK-UP



More back-up storage pools can be defined, simultaneous writing
The target storage pools can be of different types (tape, disk)
Disaster tolerant systems

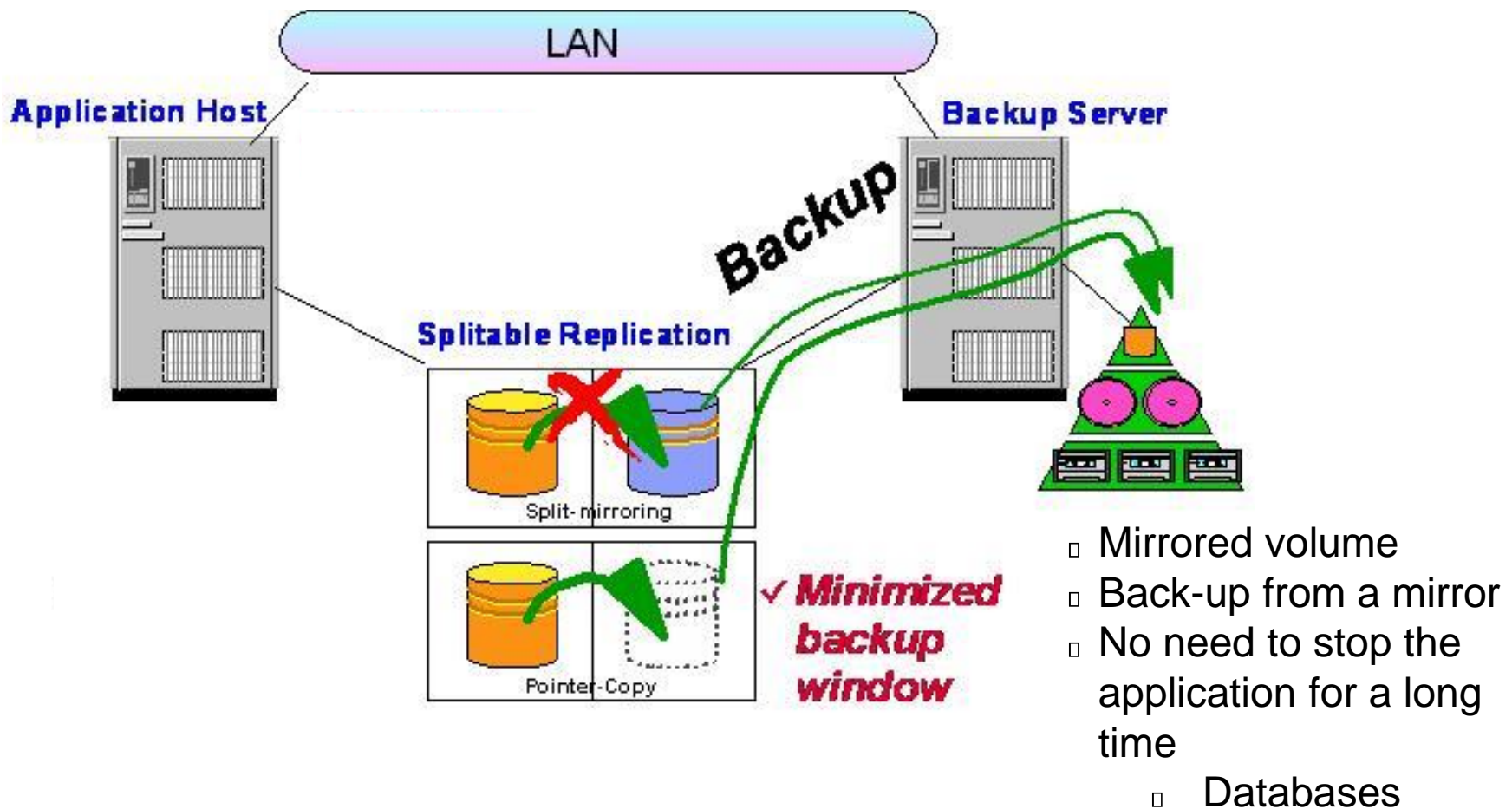


VOLUME COPY - CLONE

- Data-consistency have to be ensured – stop application during cloning
 - Long time
 - Problem especially in case of DataBase back-up
 - Alternative: split mirror



ZERO DOWN-TIME STORAGE –SPLIT MIRROR



BACK-UP METHODS

- Full Back-up
- Incremental Back-up
- Differential Back-up
- Progressive Back-up Methodology



FULL BACK-UP

- Back-up the whole disk content every day
 - Huge amount of data
 - Slow
 - Low utilisation of tapes
 - Backed-up many times, even if not changed
- BUT:
 - Fast restoration (one tape)

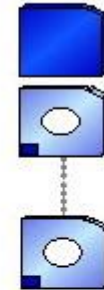


INCREMENTAL BACK-UP

- Full back-up only on the first day
 - Then only the changes **since the previous day**
 - Small amount of data
- BUT:
- Restoration is slow
 - Low utilisation of tapes
 - almost empty

Full + Incremental

Full Backup
Incrementals



Full Backup
Incrementals



DIFFERENTIAL BACK-UP

- Full back-up only on the first day
 - Then only the changes **since full back-up**
 - Greater, constantly increasing amount of data
- BUT:
- Shorter restoration time (max. 2 tapes)
 - More tapes

Full + Differential

Full Backup



Incremental



Differential



Differential



Full Backup



PROBLEM OF INCREMENTAL / DIFFERENTIAL BACK-UP

Day 1	Day 2	Day 3	Day 4	Day 5
File A	File A renamed to File F	File F	File F	File F deleted
File B	File B deleted			
File C	File C renamed to File G	File G	File G	File G
File D	File D moved to new location	File D (new location)	File D deleted	
File E	File E	File E	File E	File E

Files from Day 1 FULL backup		Files from Day 3 INCREMENTAL / DIFFERENTIAL backup		Hard Drive after a restore to Day 3
File A	+	File F	=	File A – wrong
File B				File F
File C		File G		File B – wrong
File D		File D (new location)		File C – wrong
File E				File G
				File D – wrong
				File D (new location)
				File E



PROGRESSIVE BACK-UP METHODOLOGY

- Full back-up only once
- Then only incremental
- But back-up the file system, too
 - A bit (?!) more to back-up
- At restoration, we can know what is the actual state of a file
 - Valid/deleted/renamed/moved
- Restoration with backward searching
- Much faster when restore files that
 - modified several times
 - deleted



ADVANTAGE OF PROGRESSIVE BACK-UP METHODOLOGY

Day 1	Day 2	Day 3	Day 4	Day 5
File A	File A renamed to File F	File F	File F	File F deleted
File B	File B deleted			
File C	File C renamed to File G	File G	File G	File G
File D	File D moved to new location	File D (new location)	File D deleted	
File E	File E	File E	File E	File E

Required files from Day 1 FULL backup
File E

+

Required files from Day 2 & Day 3 INCREMENTAL backups
File F
File G
File D (new location)

=

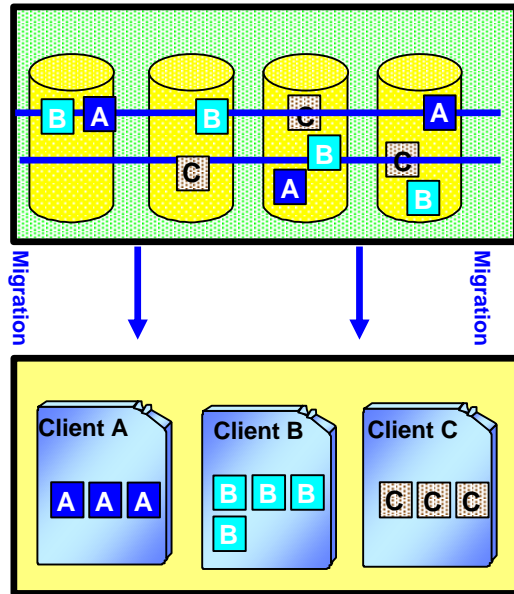
Hard Drive after a restore to Day 3
File F
File G
File D (new location)
File E



COLOCATION AND TAPE RECLAMATION

Colocation

Data of a client (group) to the same tape
Shorter restoration, fewer tape exchange



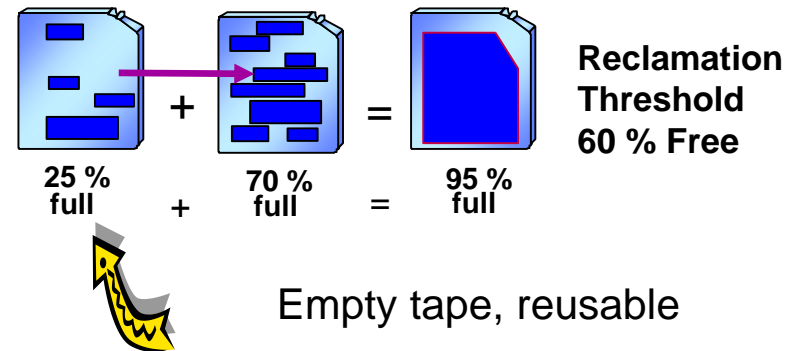
Hi Threshold
Lo Threshold

A Client A
B Client B
C Client C

Tape Reclamation

Copy valid data to a new tape after a user-specified threshold reached

This copying can be scheduled




VERSION HANDLING SYSTEMS

- Goal: to store/retrieve any version of a file
- Clone the file after each modification
 - Problem: Most part of the file was not changed
 - Storing the non-modified parts in several copies
 - Solution: Store only the modified parts + descriptor
 - Flash Copy



FLASH COPY (SNAPSHOT)

- If we modify a block – no overwrite – store at a different place
- Flash Copy table to point to the blocks of a file
 - „Snapshot”
 - Rollback possible to any time/version
- COW: Copy On Write 
- Not suitable for back-up, since no real copy generated!
 - For version handling systems



FLASH COPY - BLOCKS

	Blocks on the disk			Flash Copy table		
Time	T1	T2	T3	F1	F2	F3
	B0		B8	B0		B8
	B1	B1		B1	B1	
	B2		B9	B2		B9
	B3	B3	B3	B3	B3	B3
	B4	B4	B4	B4	B4	B4
	B5	B5	B5	B5	B5	B5
	B6	B6	B6	B6	B6	B6
	B7	B7	B7	B7	B7	B7
Total # of blocks	8			8	8	8
Delta (Flash Copy increment)	0			Virtual Volume A		



FLASH COPY - BLOCKS

	Blocks on the disk			Flash Copy table		
Time	T1	T2	T3	F1	F2	F3
Write t2	B0	B0>B8	B8	B0		B8
	B1	B1		B1	B1	
	B2		B9	B2		B9
	B3	B3	B3	B3	B3	B3
	B4	B4	B4	B4	B4	B4
	B5	B5	B5	B5	B5	B5
	B6	B6	B6	B6	B6	B6
	B7	B7	B7	B7	B7	B7
Total # of blocks	8			8	8	8
Delta (Flash Copy increment)	0			Virtual Volume A		



FLASH COPY - BLOCKS

	Blocks on the disk			Flash Copy table		
Time	T1	T2	T3	F1	F2	F3
Write t2	B0	B0>B8	B8	B0	B8	B8
	B1	B1		B1	B1	
	B2		B9	B2		B9
	B3	B3	B3	B3	B3	B3
	B4	B4	B4	B4	B4	B4
	B5	B5	B5	B5	B5	B5
	B6	B6	B6	B6	B6	B6
	B7	B7	B7	B7	B7	B7
Total # of blocks	8			8	8	8
Delta (Flash Copy increment)	0			Virtual Volume A		



FLASH COPY - BLOCKS

	Blocks on the disk			Flash Copy table		
Time	T1	T2	T3	F1	F2	F3
Write t2	B0	B0>B8	B8	B0	B8	B8
	B1	B1		B1	B1	
Write t2	B2	B2>B9	B9	B2		B9
	B3	B3	B3	B3	B3	B3
	B4	B4	B4	B4	B4	B4
	B5	B5	B5	B5	B5	B5
	B6	B6	B6	B6	B6	B6
	B7	B7	B7	B7	B7	B7
Total # of blocks	8			8	8	8
Delta (Flash Copy increment)	0			Virtual Volume A		



FLASH COPY - BLOCKS

	Blocks on the disk			Flash Copy table		
Time	T1	T2	T3	F1	F2	F3
Write t2	B0	B0>B8	B8	B0	B8	B8
	B1	B1		B1	B1	
Write t2	B2	B2>B9	B9	B2	B9	B9
	B3	B3	B3	B3	B3	B3
	B4	B4	B4	B4	B4	B4
	B5	B5	B5	B5	B5	B5
	B6	B6	B6	B6	B6	B6
	B7	B7	B7	B7	B7	B7
Total # of blocks	8			8	8	8
Delta (Flash Copy increment)	0			Virtual Volume A		



FLASH COPY - BLOCKS

	Blocks on the disk			Flash Copy table		
Time	T1	T2	T3	F1	F2	F3
Write t2	B0	B0>B8	B8	B0	B8	B8
	B1	B1		B1	B1	
Write t2	B2	B2>B9	B9	B2	B9	B9
	B3	B3	B3	B3	B3	B3
	B4	B4	B4	B4	B4	B4
	B5	B5	B5	B5	B5	B5
	B6	B6	B6	B6	B6	B6
	B7	B7	B7	B7	B7	B7
Total # of blocks	8	10		8	8	8
Delta (Flash Copy increment)	0			Virtual Volume A		



FLASH COPY - BLOCKS

	Blocks on the disk			Flash Copy table		
Time	T1	T2	T3	F1	F2	F3
Write t2	B0	B0>B8	B8	B0	B8	B8
	B1	B1		B1	B1	
Write t2	B2	B2>B9	B9	B2	B9	B9
	B3	B3	B3	B3	B3	B3
	B4	B4	B4	B4	B4	B4
	B5	B5	B5	B5	B5	B5
	B6	B6	B6	B6	B6	B6
	B7	B7	B7	B7	B7	B7
Total # of blocks	8	10		8	8	8
Delta (Flash Copy increment)	0	2		Virtual Volume A		



FLASH COPY - BLOCKS

	Blocks on the disk			Flash Copy table		
Time	T1	T2	T3	F1	F2	F3
Write t2	B0	B0>B8	B8	B0	B8	B8
	B1	B1		B1	B1	
Write t2	B2	B2>B9	B9	B2	B9	B9
	B3	B3	B3	B3	B3	B3
	B4	B4	B4	B4	B4	B4
	B5	B5	B5	B5	B5	B5
	B6	B6	B6	B6	B6	B6
	B7	B7	B7	B7	B7	B7
Total # of blocks	8	10		8	8	8
Delta (Flash Copy increment)	0	2		Virtual Volume A	B	



FLASH COPY - BLOCKS

	Blocks on the disk			Flash Copy table		
Time	T1	T2	T3	F1	F2	F3
Write t2	B0	B0>B8	B8	B0	B8	B8
Write t3	B1	B1	B1>B10	B1	B1	
Write t2	B2	B2>B9	B9	B2	B9	B9
	B3	B3	B3	B3	B3	B3
	B4	B4	B4	B4	B4	B4
	B5	B5	B5	B5	B5	B5
	B6	B6	B6	B6	B6	B6
	B7	B7	B7	B7	B7	B7
Total # of blocks	8	10		8	8	8
Delta (Flash Copy increment)	0	2		Virtual Volume A	B	



FLASH COPY - BLOCKS

	Blocks on the disk			Flash Copy table		
Time	T1	T2	T3	F1	F2	F3
Write t2	B0	B0>B8	B8	B0	B8	B8
Write t3	B1	B1	B1>B10	B1	B1	B10
Write t2	B2	B2>B9	B9	B2	B9	B9
	B3	B3	B3	B3	B3	B3
	B4	B4	B4	B4	B4	B4
	B5	B5	B5	B5	B5	B5
	B6	B6	B6	B6	B6	B6
	B7	B7	B7	B7	B7	B7
Total # of blocks	8	10		8	8	8
Delta (Flash Copy increment)	0	2		Virtual Volume A	B	



FLASH COPY - BLOCKS

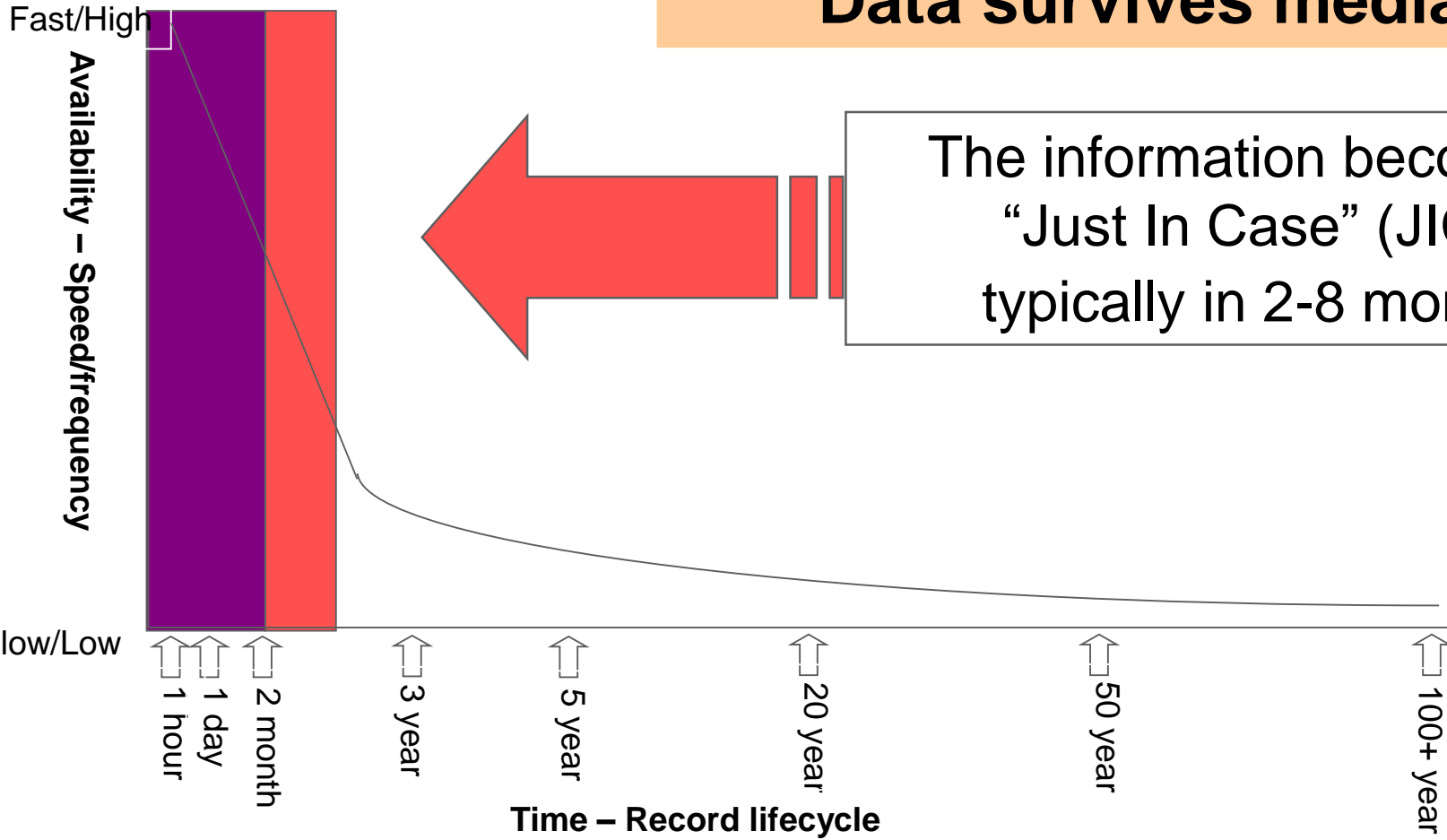
	Blocks on the disk			Flash Copy table		
Time	T1	T2	T3	F1	F2	F3
Write t2	B0	B0>B8	B8	B0	B8	B8
Write t3	B1	B1	B1>B10	B1	B1	B10
Write t2	B2	B2>B9	B9	B2	B9	B9
	B3	B3	B3	B3	B3	B3
	B4	B4	B4	B4	B4	B4
	B5	B5	B5	B5	B5	B5
	B6	B6	B6	B6	B6	B6
	B7	B7	B7	B7	B7	B7
Total # of blocks	8	10	11	8	8	8
Delta (Flash Copy increment)	0	2	3	Virtual Volume A	B	C



THE JIC INFORMATION

Data survives media!

The information becomes
“Just In Case” (JIC)
typically in 2-8 month

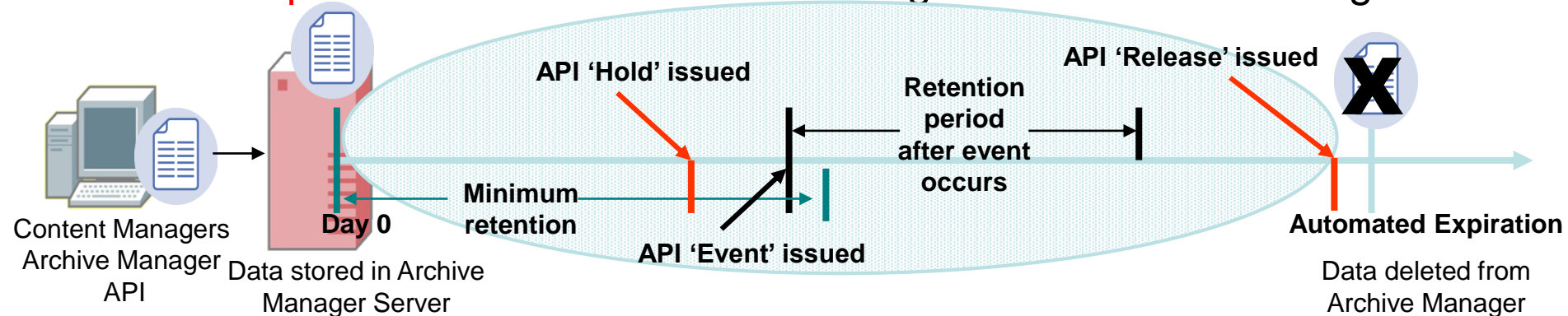


SPECIAL ARCHIVING REQUIREMENTS

Protect data from deletion till a predefined time / but deletes immediately after it

Archiving versions

- Chronological retention time policy:
 - Store the objects **for a predefined time** – e.g. 3 years –
- Event-based retention protection:
 - The storage time depends on **an event** – e.g. 70 years after client's death
- Deletion Suspend / Resume:
 - **Suspend deletion** of certain files – e.g. the end of an investigation



ARCHIVATION

- Goal is different
- On technical level
 - ALWAYS full back-up
 - Archive tapes must be treated separately from „normal” back-ups – duplicating, off-site storing
 - Store at different places
 - Long life (~10 years) so not only the tapes to keep, but
 - Devices creating/reading archives
 - Tools (programs)



WHY TO USE TAPES?

Advantages:

- Cheap
- Tapes can be taken out from archiving equipment (“Jukebox”) – no continuous mechanical stress, power requirement
- Long life – 30 years
- Stored data can easily be deleted

Disadvantages:

- Serial data access (slow)
- Tape insertation time
- More vulnerable



LINEAR TAPE-OPEN (LTO) STANDARD

- *Tape Format Standard*: IBM, HP and Certance (Seagate) cosortium
 - Open system standard technology
 - 8th generation
 - Backward compatibility
- Since its appearance (2000) wide range of industrial acceptance, leading tape technology
- Actual technology: Ultrium 8
 - 2000: 100 GB
 - 2017: 12 TB (v8)
 - Since v3:WORM
 - Since v4: Encryption
 - Data compression
 - New releases under standardisation



LINEAR TAPE-OPEN (LTO) FEATURES

Type	Year	Capacity	Entire-tape reads/writes	Approximate years of life assuming one tape filled...	
				per month	per week
LTO-1	2000	100 GB	200	17	4
LTO-2	2003	200 GB	250	21	5
LTO-3	2005	400 GB	364	30	7
LTO-4	2007	800 GB	200	17	4
LTO-5	2010	1.5 TB	200	17	4
LTO-6	2012	2.5 TB	—	—	—
LTO-7	2015	6 TB	—	—	—
LTO-8	2017	12 TB	—	—	—



BACK-UPS MUST BE PLANNED

- Not enough: „start at midnight”
 - Several back-up types!
 - *Back-up window* should not be the same
 - Back-up always reduces the performance of the system
 - Do at off-peak hours
 - But when are the peak hours?
 - Back-up outsourcing can lead to problems



PLAN OF BACK-UP

- Corporate Guidelines
- Service Level Agreement (SLA)
- Back-up and Restoration Policy
- Back-up Schedule



CORPORATE GUIDELINES

- Valid for the whole company
- Defines terminology and requirements for data-recovery systems
 - legal requirements
 - types of data to back-up
- Does not deal with low-level, implementation-close details



SLA

- An SLA is a written document that specifies what kind of services and performance are to be provided
- Created with involving the customers
- Defines:
 - types of back-up
 - requested restoration times
 - how often to back-up
 - how long to keep back-up tapes
 - back-up windows



SLA EXAMPLE

- Customers shall be able to get back any file
 - with a granularity of 1 business day for the past 6 months
 - with a granularity of 1 month for the last 3 years.
- Disk failures shall be restored in 4 hours, with no more than 2 days of lost data
- Archives shall be full back-ups on separate tapes generated quarterly and kept 'forever'
- Critical data will be stored on a system that
 - Creates user-accessible snapshots at every hour from 7 AM until 7 PM,
 - Keeps 'midnight snapshots' for 1 week
- Databases and financial systems shall have stricter requirements that shall be determined separately



BACK-UP POLICY

- When SLA approved, determine the policy how to achieve the requirements
- Typically obvious:
 - From SLA Example:
 - Daily back-ups
 - Tapes will be retained as specified in SLA
 - The policy determines how frequently to generate full back-ups
 - The rest can be incremental or differential



BACK-UP SCHEDULE

- The back-up schedule lists details down to which partitions of which hosts are to be backed-up and when
- SLA changes rarely, but back-up schedule often
- Typically not written – stored in the configuration of the back-up system



BACK-UP SCHEDULE EXAMPLE

- Size of a partition: 4GB
- Full back-up: every 4 weeks (28 days)
- Suppose, size of differential back-up grows by 5% every day
 - Day 1: Full back-up, 4 GB
 - Day 2: 200 MB
 - Day 3: 400 MB, etc.
 - Day 10: 2 GB
 - Day 11: 2.2 GB
 - Just these two days require more than a full back-up
 - Worth perform full back-ups at every 10th day!



BACK-UP POLICY – TELL TO USERS

- Back-ups are performed only on data stored on servers (your PC's Z: drive, or UNIX /home directory) every night between midnight and 8 AM.
- *We never create back-ups of your PC's local C: drive.*
- If you need a file recovered, go to [URL] for more information, or send email to [e-mail address] with the name of the server, the file's complete path, and which date you need the restoration from.
- Access problems, simple restores are done in 24 hours.



EXAMPLE

- We have **2 TB** data on a server
- Use incremental back-up
- Change 10% / day
 - a. In case of a weekly cycle how large amount of data is to be backed-up in 4 weeks?
- Full back-up: 2 TB
- Incrementals: $2 \text{ TB} * 10\% = 0.2 \text{ TB}$ (each day)
- One week: $2 \text{ TB} + 6 * 0.2 \text{ TB} = 3.2 \text{ TB}$
- Four weeks: $4 * 3.2 \text{ TB} = 12.8 \text{ TB}$



EXAMPLE CONT.

b. How large will be the back-up window if the writing speed of the back-up device is 100 GB/h?

- Sunday (full back-up)
 - $2 \text{ TB} / 100 \text{ GB/h} = 20 \text{ (!!)} \text{ hours}$
- Other days:
 - $0.2 \text{ TB} / 100 \text{ GB/h} = 2 \text{ hours}$



EXAMPLE CONT.

- c. How many devices are needed if the maximal allowed back-up window is 8 hours?
- Worst: Sunday: 20 hours
 - 3 devices needed



EXAMPLE CONT.

d. How many tapes are needed if we use new tape(s) every day and the capacity of a tape is 500 GB?

Sunday: $2 \text{ TB} / 500 \text{ GB} = 4 \text{ tapes}$

Other days: $0.2 \text{ TB} (= 200 \text{ GB}) = 1 \text{ tape}$

Total: $4 + 6 * 1 = 10 \text{ tapes} / \text{week}$

40 tapes / 4 weeks



EXAMPLE CONT.

e. Maximum how many tapes are needed to restore the content of a given day?

Worst: Saturday

Restore: 1 full + 6 incrementals

$4 + 6 * 1 = 10$ tapes needed



TAPE REQUIREMENT

- Back-up Policy affects the number of tapes needed



RESTORATION

- Slow...
- Reading and writing speed of a tape often very different
- + access time!!
 - Often longer than to restore a partition!
- Speed of restoration is typically determined by the writing speed of the file descriptors!!
- Tricks to perform the back-up faster (e.g. incremental back-up) makes the restoration slower
- Hardware limits
 - Tape fast, if the data arrives with exactly the same speed as the writing speed...
- Fastening: typically independent, dedicated network for back-up and restoration



RESTORATION: ACCESS RIGHTS-RELATED ISSUES

- Who has the right to claim the restoration of (and to use) a given file? – validation!
- File access rights change after restoration?
- File to be restored at the original place with the original access rights or at a different place with probably different access rights?
- Overwrites existing data?



CENTRALIZATION

- Centralization typically reduces costs of:
 - Equipment (expensive, because require high speed and high precision mechanics and high reliability).
 - Tape replacement (expensive, because needs human's work)
- Disadvantages of distributed back-up
 - Back-up device to every server – for high reliability: 2!
 - Tape replacement takes long time
- Network back-up systems
- Jukeboxes



TAPE INVENTORY

- A set of back-up tapes with no index or inventory is only slightly more useful than no back-ups at all...
- Automatic inventory generation
 - No – read every tape backward in time...
 - Partition level
 - File level – fast, but large
 - Compromise
- How to restore if the restoration system fails?
 - At least minimal info to store on the tape itself



FIRE DRILLS

- The only time to learn the quality of the back-up system is when doing a restoration
 - Restoring a randomly chosen file
 - Restoring a whole disk
 - rarely needed, maybe forget how to do
 - large amount of data – capacity/bandwidth is really enough?



CHANGES IN TECHNOLOGY

- Technology changes are different
 - Disk: almost linear (capacity doubles in ~1.5 years)
 - Tape: capacity remains the same for years and then a ‘big jump’
 - The tape units (jukeboxes) are expensive – not to replace *them* often
- When new tape type comes out – retain 1 (2) pieces from the old platform!
 - Equipment
 - Software
 - Or: migrate EVERYTHING to the new technology



BACK-UP AND RESTORATION - SUMMARY

- Back-up / archive
- Types of back-ups
 - Full, incremental, differential, progressive
- Version handling systems
- Design of back-up
 - Corporate Guidelines, SLA, Back-up policy, Back-up schedule, Time and Capacity Planning, Tape need calculation
- Restoration

