



Education

Trends in Application Recovery

Andreas Schwegmann, HP

- ◆ The material contained in this tutorial is copyrighted by the SNIA unless otherwise noted.
- ◆ Member companies and individual members may use this material in presentations and literature under the following conditions:
 - ◆ Any slide or slides used must be reproduced in their entirety without modification
 - ◆ The SNIA must be acknowledged as the source of any material used in the body of any document containing material from these presentations.
- ◆ This presentation is a project of the SNIA Education Committee.
- ◆ Neither the author nor the presenter is an attorney and nothing in this presentation is intended to be, or should be construed as legal advice or an opinion of counsel. If you need legal advice or a legal opinion please contact your attorney.
- ◆ The information presented herein represents the author's personal opinion and current understanding of the relevant issues involved. The author, the presenter, and the SNIA do not assume any responsibility or liability for damages arising out of any reliance on or use of this information.

NO WARRANTIES, EXPRESS OR IMPLIED. USE AT YOUR OWN RISK.

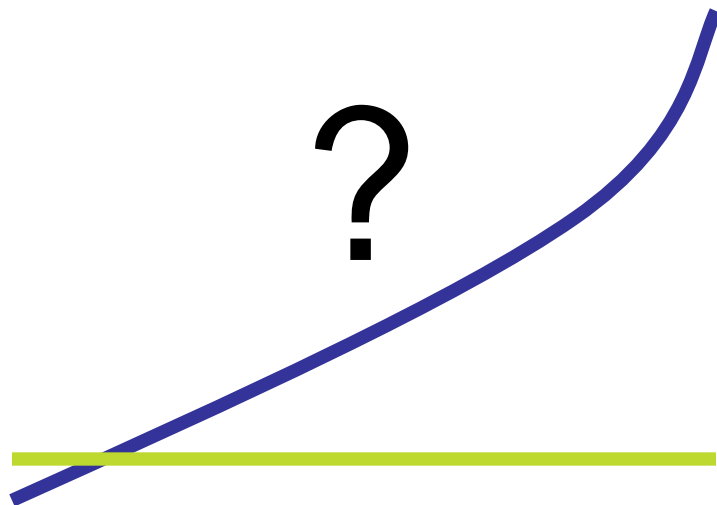
➤ Trends in Application Recovery

- ◆ This session will appeal to Data Center Managers, Backup Administrators, Application Administrators and those that are concerned how data lost in an application or database could be recovered.
 - › Challenges and trends in application backup and recovery.
 - › You will be guided through all layers from the service down to the physical hardware.
 - › You will learn how to recover individual lost pieces of information up to the recovery of the entire application distributed across complex and virtualized environments.
 - › You will get some ideas how to deal with the human factor in IT environments with distributed responsibilities.
 - › Finally the session discusses how to balance service levels against cost.

Application Recovery - Agenda



- Challenges and trends
- What to recover and by whom?
- Architectures
- Application backup
 - ◆ Consistency
 - ◆ Backup window
- Interfaces
- Recovery
 - ◆ Application entirely
 - ◆ Single items
- Virtualization
- Conclusion



- ◆ Blue line – exponential data volume growth & complexity growth = the bullets on this slide
- ◆ Green line – IT budget is flat

◆ Exponential data growth

- ◆ Cheaper storage
- ◆ More performance
- ◆ Long term retention
- ◆ Distributed data -> duplicate data

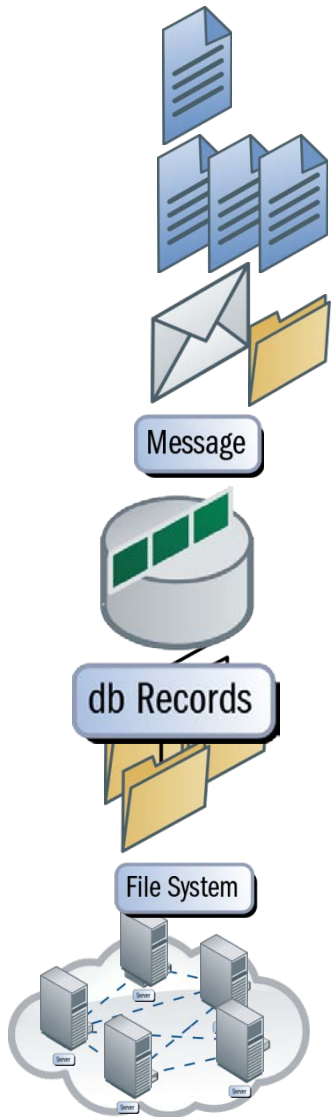
◆ Complexity growth

- ◆ Scale up & out
- ◆ Different data types in one application
- ◆ High availability
- ◆ Virtualization
- ◆ Cloud
- ◆ Everything as a Service
- ◆ Search
- ◆ Security & compliance
- ◆ More features

Application Recovery Trends

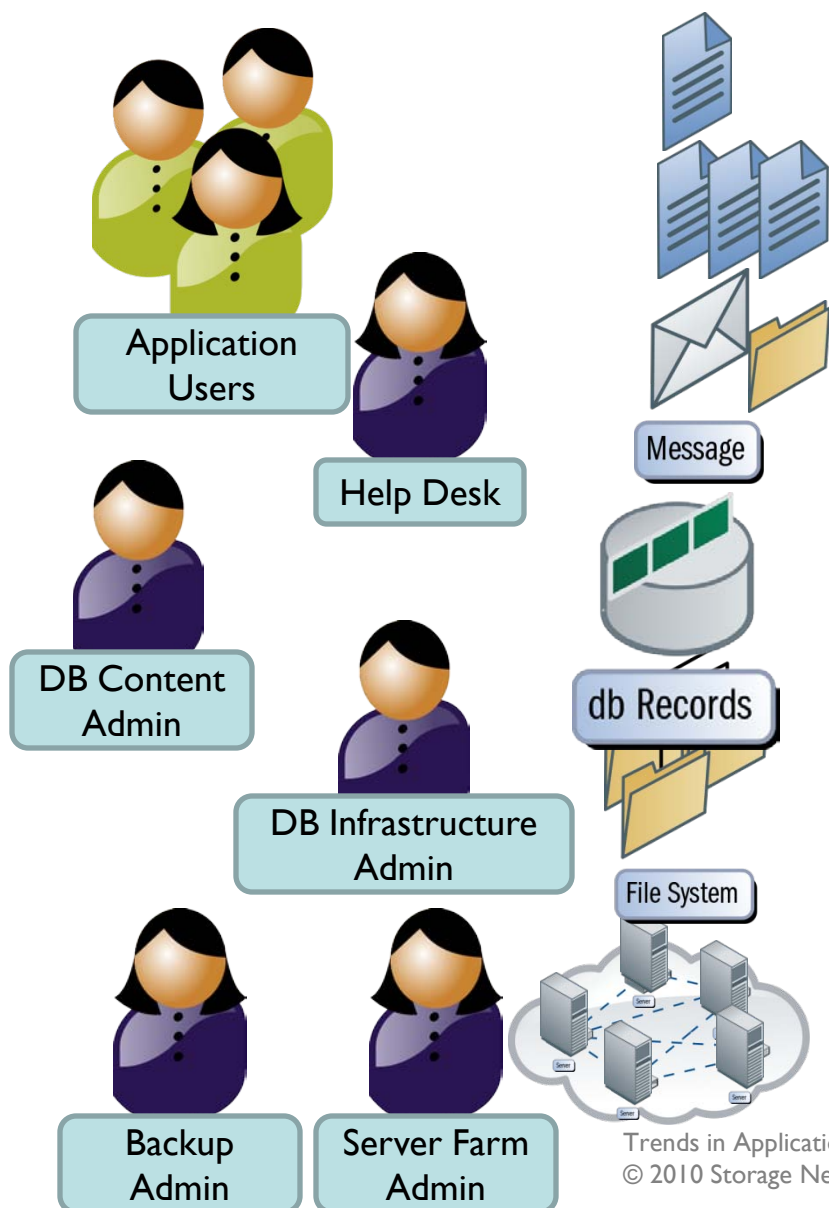
- Recovery to any point in time
- Snapshot based recovery
- Seamlessly link disk with tape
- Data reduction
 - ◆ Incremental and consolidation of incremental backups
 - ◆ Deduplication, primarily of repeated full backups
- Recovery automation
 - ◆ The know how to recover the application is build into the backup software
- Server farm down to single document, mail, ...
- The “cloud” – somebody does it for you

What to Recover?



- Single items / end user domain
 - ◆ Entry, record, transaction
 - ◆ Document, e-mail, blog
 - ◆ Calendar, tasks, contacts
 - ◆ Table, list, tree, folder, wiki
 - ◆ E-mail box, user site
- Database
 - ◆ Files system or raw disk
 - ◆ Cache
- Search Index & Services
- Application configuration
- Binaries, OS, configurations
- Server farm
- Virtual servers & physical servers
- Disk array

Who Does What?



- Different users groups use different user interfaces
 - ◆ Users and administrators don't want more tools, they want to manage from their tools.
- Security
 - ◆ User A should not be able to see data from user B.
 - ◆ The administrator should not be forced to break into the end user privacy.
 - ◆ DB administrator might not have the right to restore from backup.
- Processes
 - ◆ Application administrators might not be connected with backup administrators.

Application Architectures

- Single server, single database
- Server farm with multiple databases of different types
- Database across databases – search index
- Databases with links into the file system
 - ◆ Using standard files
 - ◆ BLOBs
- Cluster: 2 –N nodes
 - ◆ Load sharing, fail over, majority node, ...
- Replication
 - ◆ Active and passive databases
- DAS, SAN
- Physical servers, virtual machines
- On premise, cloud

How to reduce the backup volume?

- Full backup – file based or block based
 - ◆ Database
 - › Data files = “tables“ mapped to files
 - › Control files to find data files & log files
 - › Transaction log files, optional: move / delete
 - ◆ Trend features
 - › Files, BLOBs – typically used for storing large files outside of the DB
 - › Search Index, services, encryption keys, ...
- Incremental backup – changes since last backup
 - ◆ File based
 - › Transaction log backup and move / delete
 - › Files: File system incremental backup of single instance file store
 - ◆ Changed blocks
- Differential backup – changes since last full backup
- Compression
- Deduplication

➤ When is an application consistent?

- ◆ Data is valid at the same point in time
- ◆ Data is complete

➤ How to accomplish consistency for backup?

- ◆ Offline backup - application shutdown
- ◆ Crash consistent backup – snapshot without interaction
- ◆ Online backup – application interaction

Consistency - Offline Backup

- Shutdown the application / database
 - ◆ Guarantees application consistency
 - ◆ All cache data copied to disk
 - ◆ All transactions closed
 - ◆ Optional: database consistency check
- Backup to another disk / tape
 - ◆ OR create a snapshot
- Optional: move / delete the transaction logs
 - ◆ Frees disk space
 - ◆ Enables incremental backup based on transaction logs
- Start the application
- Optional: backup the snapshot to another disk or tape
- Recycle the snapshot
 - ◆ Keep the last N snapshots
 - ◆ Snapshot rotation

- ❖ Create a snapshot while the application runs
 - ◆ Application consistency has the same quality as after a system crash
 - ◆ Most applications / databases can survive system crashes
 - › But some don't and some not always.
 - ◆ Recovery can not be guaranteed
- ❖ Use cases
 - ◆ 7 x 24 operations -> no backup window
 - ◆ Virtual Machine backup without agent or service API
 - ◆ Application lacks online backup mode feature
 - ◆ No resources for transaction logging during backup
 - ◆ Snapshots enable more points in time
 - › Might reduce the risk

- Database(s) are in “backup mode“ during backup
 - ◆ Data files don't change while in backup mode
 - ◆ Changes during backup happen in the cache and go into logs
 - ◆ After backup all changes are applied to the data files
 - ◆ Optional: backup of the transaction logs & delete logs afterwards
 - ◆ Optional: ongoing log file backup after database backup -> “CDP“
- Consistent search index
 - ◆ All databases need to go into the backup mode
 - ◆ Across the server farm
- Use cases
 - ◆ 7 x 24 operations -> no backup window
 - ◆ Guaranteed & fully supported consistent recovery

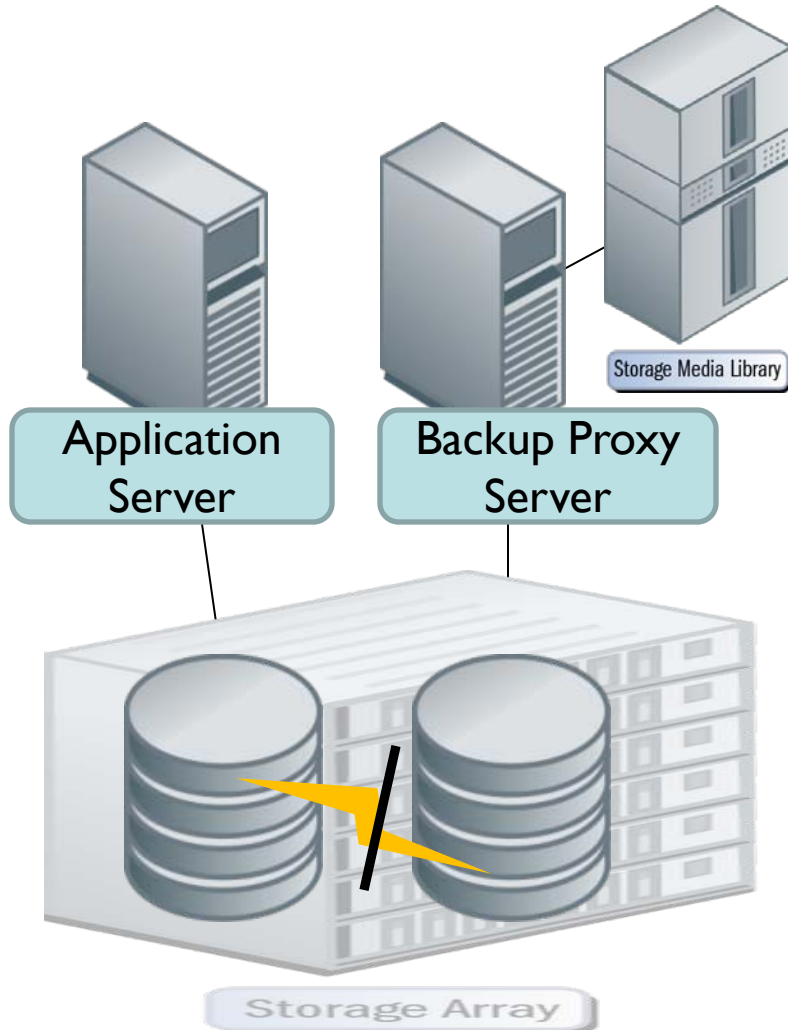
Application Backup Interfaces

- Application specific tools via GUI, CLI
- General purpose API
- Streaming backup API
 - ◆ Direct copy
 - › Access to in-memory copy of data, cached by the application
 - › Minimizes redundant memory copies
 - ◆ Incremental backup
 - › Access to changed blocks / pages or transactions
 - ◆ Optional features
 - › Granularity below database level
 - › Compression
 - › Encryption
 - ◆ Sequential access is optimal for streaming media

Volume based Backup API

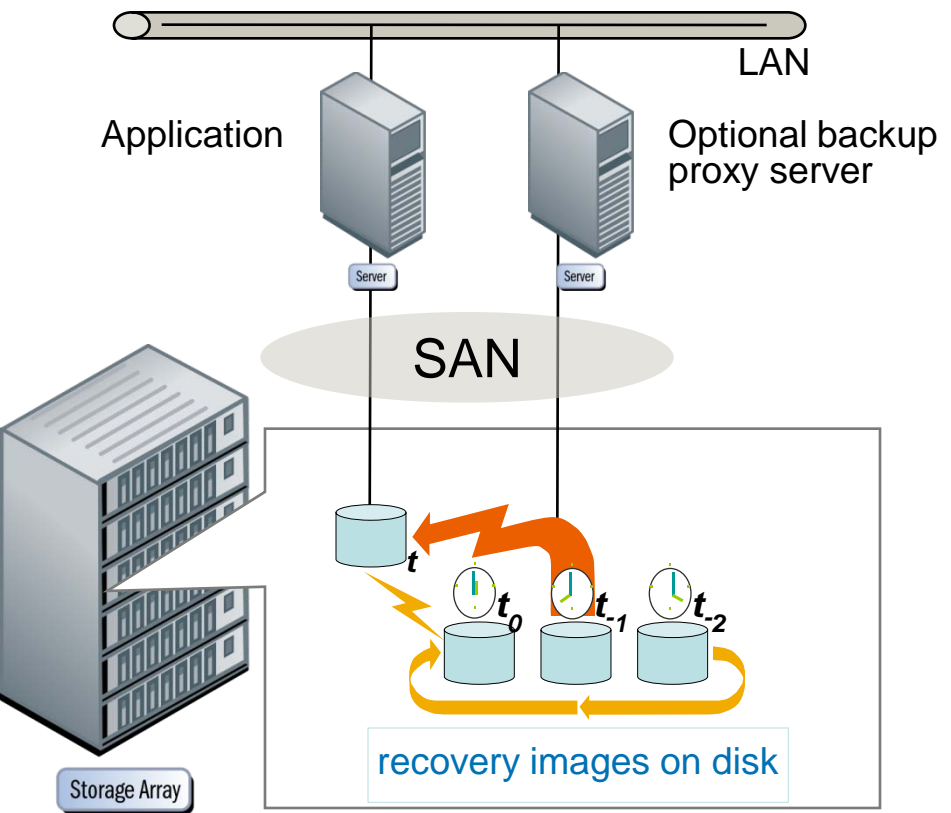
- Creates application consistent volumes ready for backup
- Use cases
 - ◆ Copy the entire volume via snapshot
 - ◆ Copy all files needed to recover the application
 - ◆ Incremental backup
 - › Changed blocks
 - › Changed files
 - ◆ Feature set might be different compared to streaming API
 - ◆ Backup to disk & restore from disk
- Trend: volume based backup
 - ◆ Better for backup to disk
 - ◆ Better for virtualization

Reduced Performance Impact



- Separate backup proxy server
- Backup from full copy snapshot (mirror)
 - ◆ Application switches into backup mode
 - ◆ Split the snapshot
 - ◆ Back to normal mode
 - ◆ Separate backup proxy server copies the data from split mirror
 - ◆ Resync the mirror after backup
 - › Copies changed blocks only

Application Recovery from Snapshot

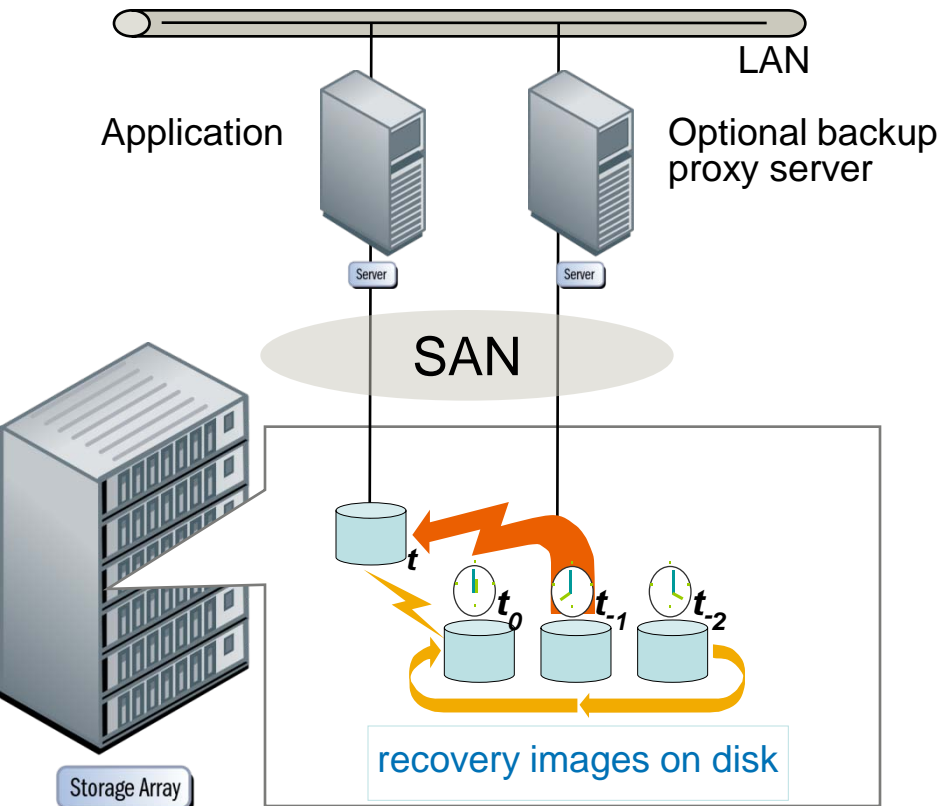


- Application shutdown
- Optional incremental transaction log backup
- Switch to selected snapshot
 - ◆ Instead of restore from tape
- Transaction log roll forward from backup or original disk up to the most recent point in time
- Application back online

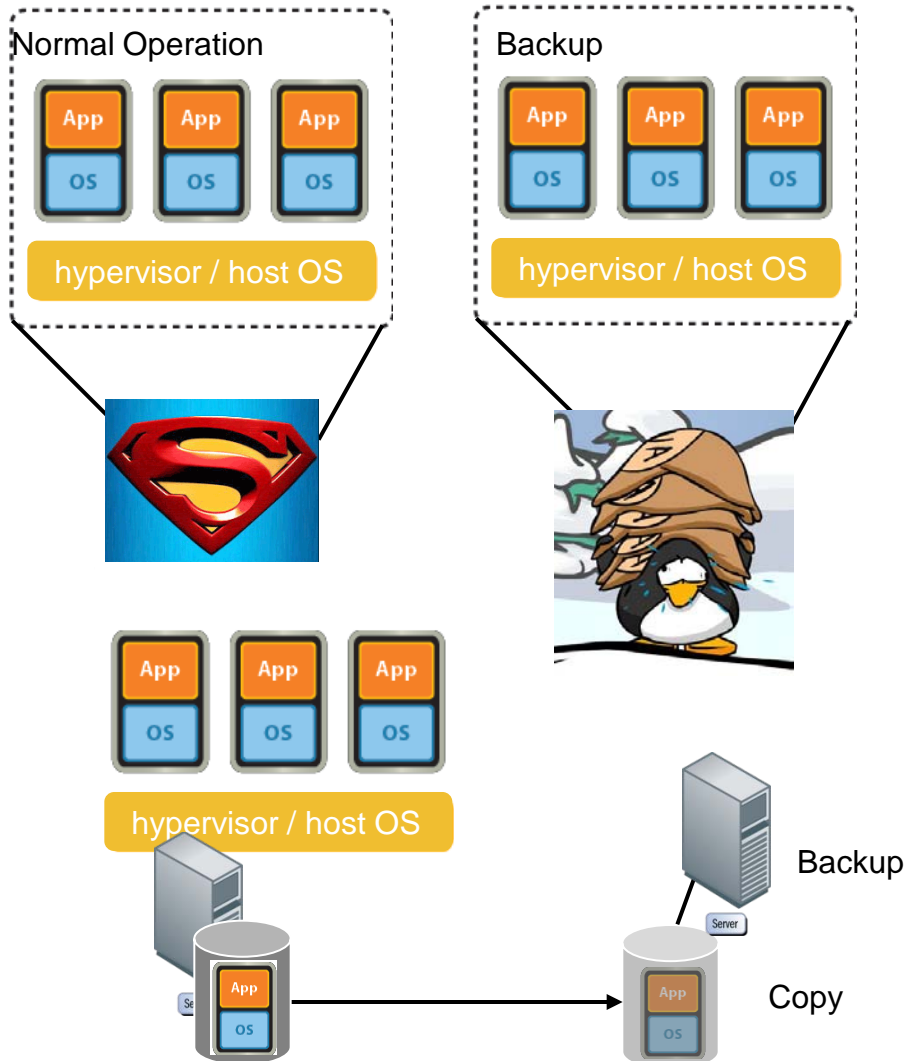
Single Item Recovery Options

- Dumpster, 2nd level dumpster, versioning, archive
- Full blown recovery environment & copy back
 - ◆ Spare systems
 - ◆ Virtual Machines
- The application can be used to extract single items from backup
 - ◆ Copy database from backup & mount as recovery database
 - › Needs extra space and time to copy the entire database
 - ◆ Mount the database from the backup directly into the application
- Open the backup database with a separate tool & extract
- Extract single items directly from the backup
 - ◆ Catalog of all single items during or shortly after backup
- Single item recovery from single item backup
 - ◆ Needs a separate “brick level” backup

Single Item Recovery from Snapshot

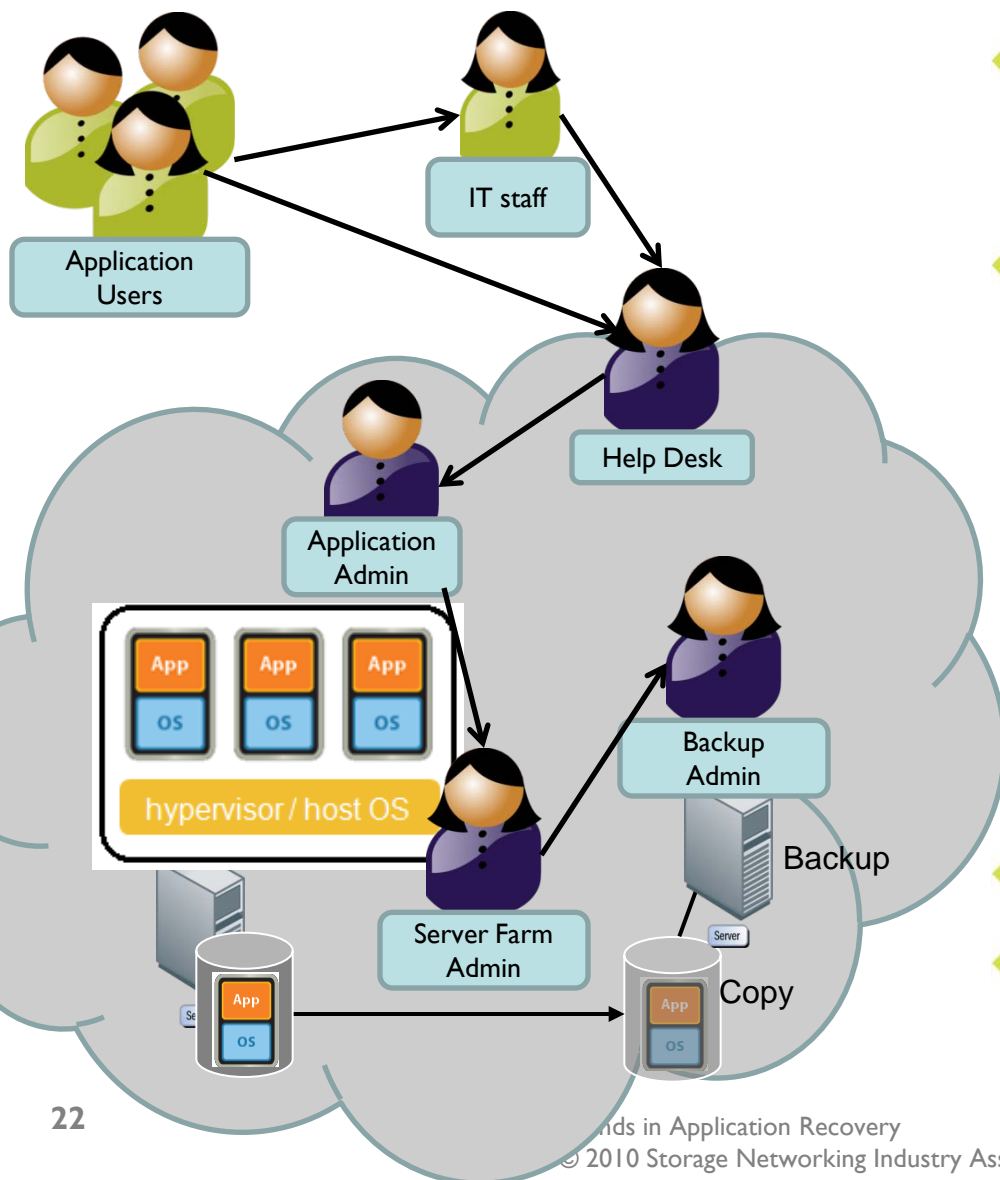


- Mount the database from the snapshot
- Browse & search through the database
 - Using 3rd party tool
 - Using the application
- Unmount the snapshot



- Resources shared among different applications
 - ◆ Normal load spreads evenly across day / week / month
 - ◆ Backup load is exception
- Resources on physical server often not enough for backup load
- Offload backup via dedicated physical machine
 - ◆ Utilize replication

Virtualization – Who Does What?



➤ Who does the backup?

- ◆ Same SLA for all VMs?
- ◆ Who can define the SLA?

➤ Who recovers what?

- ◆ Hypervisor / host
- ◆ Individual Virtual Machine
- ◆ Single file from the VM
- ◆ Application
- ◆ Application data object
 - › E-mail
 - › Document
 - › Tablesapce
 - › Record

➤ What's public / private?

➤ Security?



- Application specific solutions needed
 - ◆ There is no one size fits all
 - ◆ Hybrid solutions might be needed
- High complexity - many options
 - ◆ Who are your customers & users?
 - ◆ What are the requirements?
 - ◆ How complex is your environment?
 - ◆ What are your use cases?
 - ◆ What is your budget?
 - ◆ What are the Service Level Agreements?
 - › Backup window, RPO, RTO

- Please send any questions or comments on this presentation to SNIA:

trackvirtualizationapplications@snia.org

**Many thanks to the following individuals
for their contributions to this tutorial.**

- SNIA Education Committee

Andreas Schwegmann – this slide deck

Antal Nemes – application backup interfaces

Nancy Clay – tutorial program management