



IBM Software Group

IBM Informix Dynamic Server Backup and Restore

Wan Guo Dong

2009-11

@.business on demand software

Right, Fast, or Cheap

- There are three ways to implement recovery:
right, fast, or cheap.
- Choose the two highest on your priority list. These choices should directly relate to your data and time-loss recovery goals.
- Most people will choose the *right* option as their first priority. The difficult step is to weigh the advantages and disadvantages of speed and cost.

Look at Available Tools

Now that recovery goals have been set, it is time to identify the IBM Informix backup and recovery tools. In addition, other useful features and utilities that provide added data redundancy and high availability will be discussed.

IBM Informix Backup Utilities

- ontape
- ON-Bar
- external backup/restore
- Archecker

IBM Informix Fault Tolerance Mechanisms

- IBM Informix Mirroring
- High Availability Data Replication
- Enterprise Data Replication

Load/Unload Utilities

- High Performance Loader
- SQL load/unload
- dbexport/dbimport
- onload/onunload
- dbschema
- dbload

Ontape

Backup and restore utility for IDS that provides:

- Backup at the server level
- Support for incremental backup
- Manual or continuous logical-log backup
- Restore at the system or dbspace level

Ontape - Advantages

- Reliability - Data can be restore up to point of failure because of logical-log backups.
- Ability to restore at the system or dbspace level
- Support for remote tape devices.
- Support for external backup and restore
- Simple configuration.
- Easy to use command syntax.
- Generally used for small database servers.

Ontape - Disadvantages

- Backup can only be done at the system level
- Each dbspace is backed up serially to a single device. For large systems, this can mean Very slow backups.
- Requires user input when device media must be changed.
- No media management.

ON-Bar

Backup and restore utility for IDS that provides:

- Parallel backup and restore.
- System level and dbspace level backup and restore.
- Support for incremental backups.
- Manual or automatic backup of logical logs
- Point-in-time recovery for an instance
- Provides an open interface for communication with storage manager (XBSA)
- Third party storage managers may provide support for sophisticated tape stacker and jukebox devices.

ON-Bar - Advantages

- Parallel backup and restore provides speed
- Reliability-Data can be restore up to point of failure because of logical-log backups
- Support for multiple devices
- Ability to restore at the server or dbspace level. This means the server can be made available to users once the critical dbspaces, i.e. root, logs, etc., have been restored.
 - ▶ Non-critical dbspaces can then be restored in order of importance.

On-bar enhancements over **ontape**:

- High speed parallel backup and restore
- Unattended operations
- Automatic backup of logical logs via the `ALARMPROGRAM` parameter
- Support for sophisticated backup storage devices through a storage manager
- Backup of selected dbspaces
- Restartable restore

ON-Bar - Disadvantage

- Requires configuration and use of the IBM Informix Storage Manager or an additional cost third-party storage manager.

External Backup Utilities

Methods of backing up data at the OS level:

- Unix commands such as
 - ▶ cpio
 - ▶ tar
 - ▶ dd
- File system snapshots
- 3rd mirror break off
- Storage manager “raw” backup
- External backup/restore feature of ON-Bar allows logical recovery

External Backup Utilities - Advantages

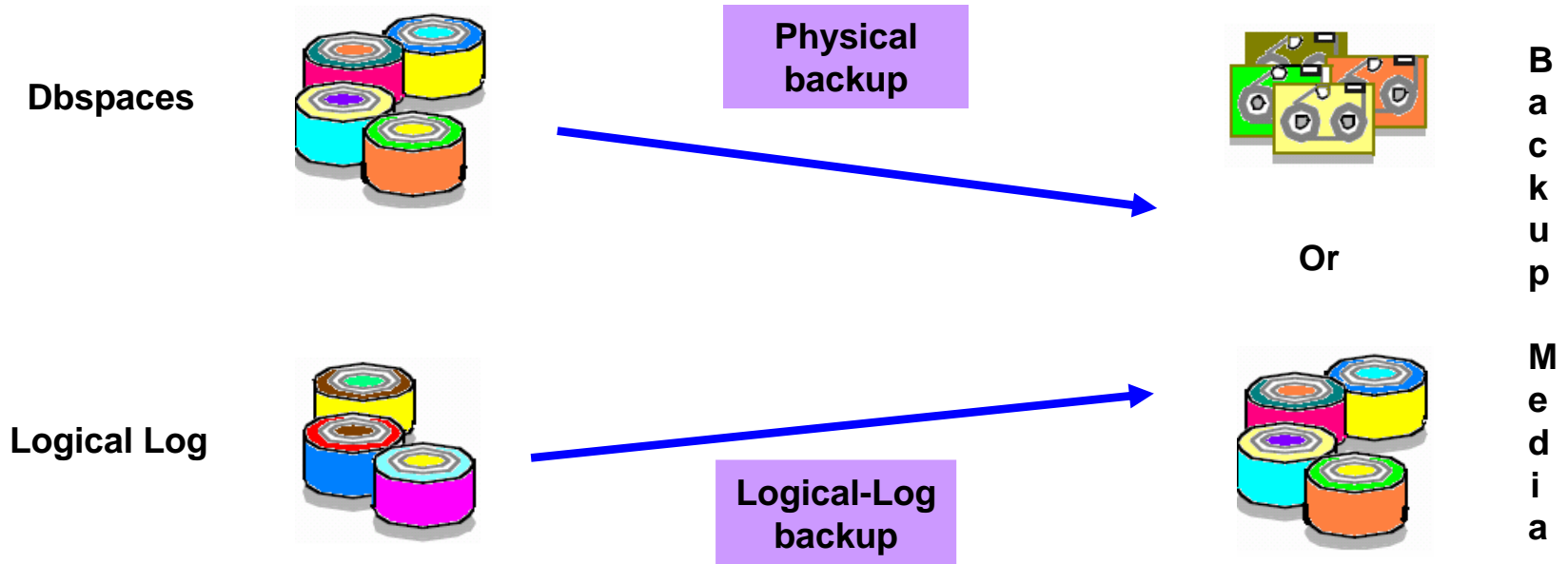
- Availability, speed-Data can be backed up and restored more quickly, making the data available sooner.
- Multiple copies of the same data can be kept.
- Ability to copy raw physical devices.
- Logical restore available if combined with ON-Bar External Restore.

External Backup Utilities - Disadvantages

- No logical recovery available when using external backup utilities unless you use the ON-Bar External Backup/Restore feature.
- Greater costs associated with third mirror break off solutions.
- No media tracking for external backups is done by ON-Bar.

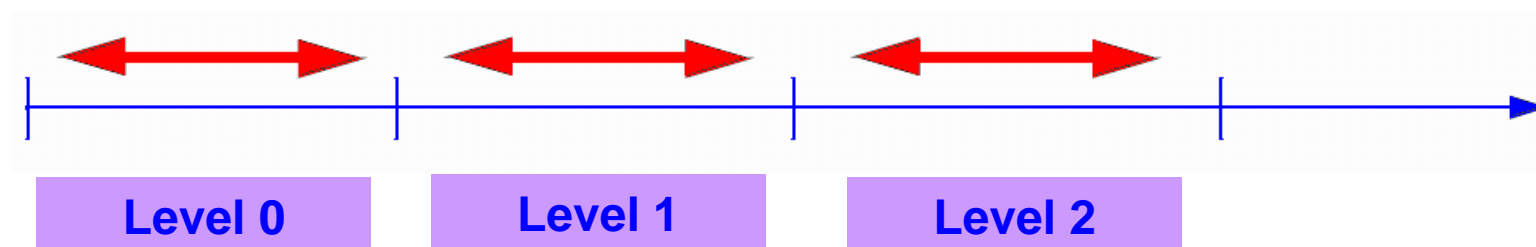
What is a Backup?

A backup occurs when all or some server database objects are copied to storage media.



External Data Blade data, temporary spaces and unused pages are not backed up during the physical backup.

Incremental Backups



- A **level 0** backup contains a copy of all data in the server system as of the time the backup started.
- A **level 1** backup contains a copy of all data in the server system that has changed since the last level 0 backup.
- A **level 2** backup contains a copy of all data in the server system that has changed since the last level 1 or level 0 backup.

Backup Schedule

Monday	Level 0 backup
Tuesday	Level 1 backup
Wednesday	Level 2 backup
Thursday	Level 1 backup
Friday	Level 0 backup
Saturday	Level 1 backup
Sunday	Level 2 backup

Ensuring Consistent Backups

How is this bit of magic performed?

1. Backup of the system is initiated at 9:00 a.m.
2. 300 users and processes continue to make updates and changes
3. System checkpoints continue to occur as configured
4. Backup completes at 1:05

Result:

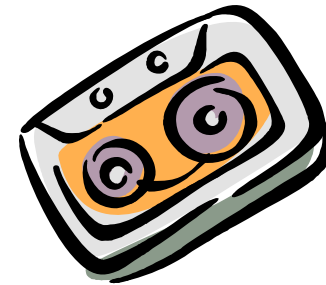
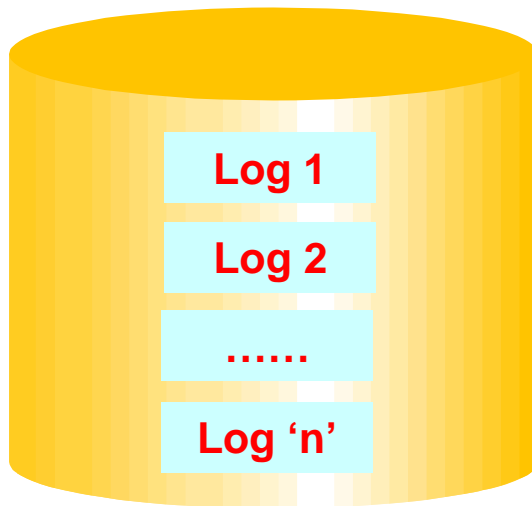
- ▶ The backup tape contains all pages required to restore the system to 9:00 a.m.
- ▶ The pages on the tape do not reflect any updates, inserts or deletes that occurred after 9:00 a.m.
- ▶ Changes made between 9:00 a.m. and 1:05 p.m. are recovered by applying logical logs.

Synchronizing Administrative Tasks

You need to make a level-0 backup of the root dbspace when you:

- ▶ Add mirroring
- ▶ Add a logical log
- ▶ Add a chunk or dbspace
- ▶ Drop a chunk or dbspace
- ▶ Drop a logical log
- ▶ Change the size or location of the physical log
- ▶ change your storage manager configuration

What is a Logical-Log Backup?



A logical-log backup is the process of copying the contents of a logical log file to secondary storage media. This allows the logical log to be reused.

In order to successfully complete a logical restore, the logical-log backup must have been created with the *same backup utility* used to create the physical backup.

When are Logs Backed Up?

- On-Demand
 - ▶ Administrator or operator issues a command
 - ▶ All used logical logs are backed up

- Automatic
 - ▶ Triggered by ALARMPROGRAM
 - ▶ Logical logs are backed up when a log full event is triggered

- Continuous
 - ▶ Running process backs up each logical log as it fills

Restoring logical log records is slow compared to restoring pages from a dbspace backup. To minimize the number of logical log records that must be applied during a recovery, create frequent dbspace backups.

Additional Backup Steps

- In addition to dbspace and log backups, regularly back up:
 - ONCONFIG file
 - sqlhosts file
 - TEXT and BYTE data stored on optical platters
 - Informix Storage Manager files and sm_versions file
 - External DataBlade data
 - Audit Configuration File (adtcfg)
 - Audit Mask Input File (adtmask)

- If ON-Bar is your backup and restore utility, also back up:
 - Emergency Boot File (ixbar)
 - oncfg
 - onbar script

Why is Additional Backup Needed?



Ontape or ON-Bar
doesn't backup
configuration and
supporting files
on file system

Operating System Files

Occasional backup of operating system files should be kept:

- ✦ Storage Manager directories
 - ✦ configuration files
 - ✦ installation directories

- ✦ Full \$INFORMIXDIR directory

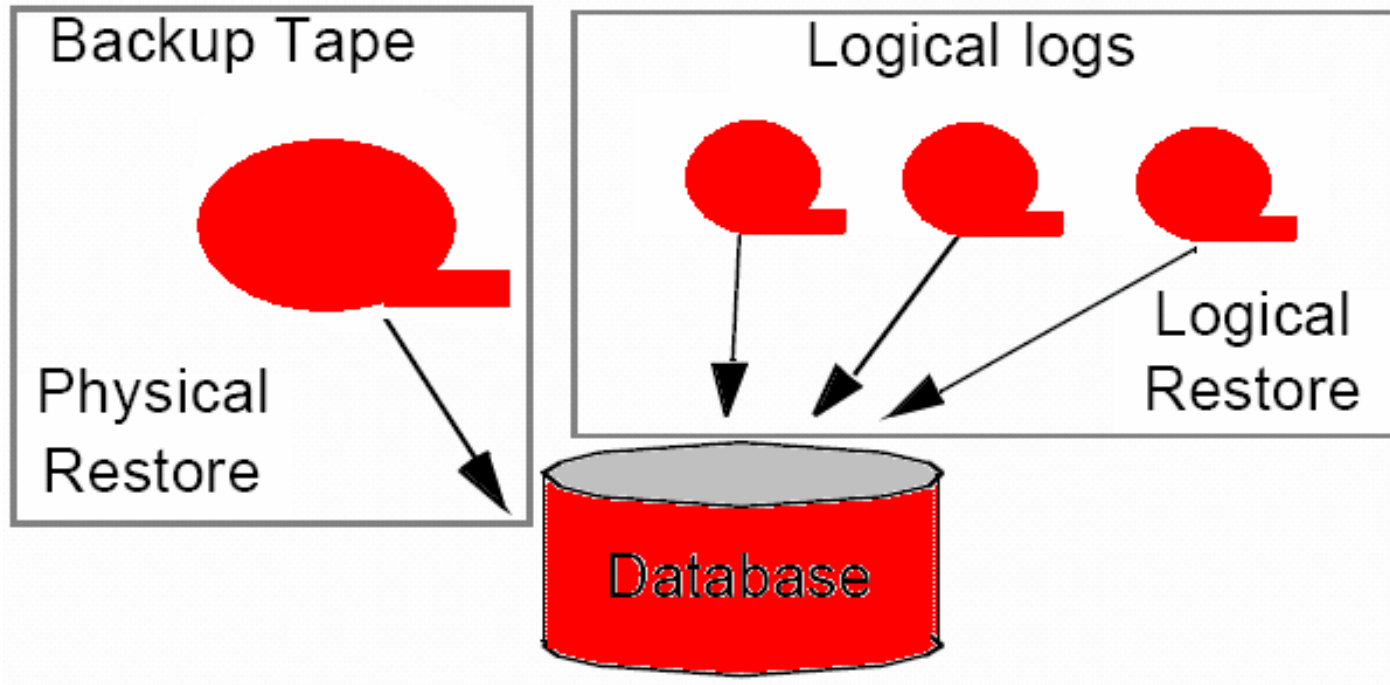
Other Information to Save

In addition to the administrative files, some other files that you may want to create and save for disaster recovery include:

- ⊕ output from **dbschema -d *database* -ss *file*** for each database
 - provides a complete data model for recreating your database

- ⊕ output from **oncheck -pr**
 - provides a list of all dbspaces, chunks, device names, offsets, sizes, and types
 - includes information about previous backups and logs required
 - Onconfig information

Physical and Logical Restore



Cold vs. Warm Restore

Cold restore:

- A restore that occurs when the root dbspace or a dbspace that holds the physical or logical logs is inaccessible

Warm restore:

- A restore that occurs when the root dbspace and the dbspace(s) containing the physical and logical logs are accessible
- The server must be in online, quiescent or single-user mode

Mixed restore:

- Cold restore followed by a warm restore

Renaming Chunks During a Restore (IDS)

- Do we have to have the chunks existing?
- Can I do Mixed restore with rename?

Renaming Chunks with Command Line Options

- To rename the chunks by supplying information on the command line, use this command:
- `onbar -r -rename -p /chunk1 -o 0 -n /chunk1N -o 20000 -rename -p /chunk2 -o 10000 -n /chunk2N -o 0`
- Perform a level-0 archive after the rename and restore operation is complete.

Renaming Chunks with a File

- To rename the chunks by supplying a file named **listfile**, use this command: `onbar -r -rename -f listfile`
- The contents of the **listfile** file are: `/chunk1 0 /chunk1N 20000 /chunk2 10000 /chunk2N 0`
- Perform a level-0 archive after the rename and restore operation is complete.

“Ontape”

- The ontape utility is the oldest utility available and one of the simplest to use. It does not require a storage manager. It can backup the data from the database server to tape or disk. The ontape utility can perform physical and logical log backups and restores. The ontape utility performs backups and restores serially.
- The main ontape features are:
 - Simplicity and ease of use
 - Backup capability to tape, file, or directory
 - Support for Table level restore
 - Support for External backup and restore

Ontape configuration and requirements

- **TAPEDEV**

This can be the absolute pathname of the tape device used for storage-space backups. This is the destination where ontape will write storage space data during an archive and the source from which ontape will read data during a restore. To configure ontape to use stdio, set TAPEDEV to STDIO. In IDS 11.10 and above, TAPEDEV can be set to a directory (see topic discussion later on in this material).

- **TAPEBLK**

This is the block size of the tapes used for storage-space backups, in kilobytes. Specify the block-size parameters TAPEBLK and LTAPEBLK as the largest block size, in kilobytes, that your tape device permits.

Ontape configuration and requirements

- **TAPESIZE**

This is the size of the tapes used for storage-space backups, in kilobytes. Can also be 0, meaning write to end of tape or until you run out of space.

- **LTAPEDEV**

This is the logical log tape device.

- **LTAPEBLK**

This is the block size of tapes used for logical log backups, in kilobytes.

- **LTAPESIZE**

This is the size of tapes used for logical-log backups, in kilobytes. Can also be 0, meaning write to end of tape or until you run out of space.

Ontape requirements

- You should always specify an **auto-rewind tape device** for the TAPEDEV and LTAPEDEV parameters
- Setting LTAPEDEV to /dev/null (NUL on Windows) turns off logical log backups.
- Verify that the tape device can read the block size that you specified. If not, you cannot restore the tape. TAPESIZE and LTAPESIZE specify the maximum amount of data that you can write to a tape.

Tips for ontape successful backup and restore

- If possible, specify separate devices for server backups and logical log backups.
- Consider using symbolic links to specify tape devices.
- Ensure that you specify valid block sizes and tape sizes for the tape devices specified. An invalid block size can fail to restore.

Tips for ontape successful backup and restore

- in general, the ontape utility is not designed for unattended backup and restore operations. Ontape is designed to be run interactively by an administrator or operator, as you must provide responses to various prompts.
- Starting in IDS 11.10, however, an ontape backup to directory feature makes it easier to setup unattended backups. (Note that the OnBar utility has always been designed for unattended backups, and may meet your needs better if unattended backups is a requirement).
- In order to use ontape, you must be user informix or root.

Ontape backup

- Backing up with ontape can be performed to achieve the following:
 - Changing database logging status.
 - Creating a backup (level 0 or incremental level 1 or 2).
 - Requesting a logical-log backup.
 - Starting continuous log backups.

Ontape backup commands and syntax

- `$ontape -s`

This command will perform a full-system backup of all dbspaces.

- `$ontape -a`

The ontape process will write the logical logs to the backup devices specified by the LTAPEDEV configuration parameter.

Ontape backup commands and syntax

- The `ontape` utility may be run in continuous backup mode.
 - ▶ `$ontape -c`
 - ▶ The **ontape -c** command should not be run as a background process.
 - ▶ You can use the `-a` option to periodically back up logical logs to tape; or you can run the `-c` option to back up each logical log as soon as it becomes full. The `-c` option will cause `ontape` to run continuously, and requires a dedicated terminal and tape drive in order to function. It cannot be run as a daemon.

Backup to directories with ontape

- You can use the ontape utility to backup and restore data from a file system without interactive prompts.

- To enable this feature,
 - ▶ set the TAPEDEV and LTAPEDEV configuration parameter to a valid directory of a local or remote mounted file system.
 - ▶ The ontape utility generates file names automatically and performs physical and log backups.
 - ▶ The person who runs the backup must have write permission to the directory. The directory must have enough disk space to hold the backed-up data. You can use operating system utilities to compress the data after it is backed up.

Backup to a directory - advantages:

- Multiple instances can simultaneously backup to the same directory file system.
- You can use operating system utilities to compress or otherwise process the data.
- You can easily configure your system to automatically backup a log file when it is full.

Backup to a directory - continued

- When TAPEDEV or LTAPEDEV is a directory path, the default permanent file name consists of
- hostname_servernum_Ln (L n for levels)
 - ▶ Example (blazer_110_L0)
- For log files backups
hostname_servernum_Lognnnnnnnnnnn ()

Overriding the default name of the archive files:

- `IFX_ONTAPE_FILE_PREFIX`.

if you set `IFX_ONTAPE_FILE_PREFIX` to "My_Instance", then during archive, the files are named `My_Instance_L0`, `My_Instance_L1`, `My_Instance_L2`, and, `My_Instance_Log0000000001`, `My_Instance_Log0000000002`, and so on..

- During restore, `ontape` searches for files in the `TAPEDEV` directory with file names like `My_Instance_L0`, and searches for files in the `LTAPEDEV` directory with file names like `My_Instance_Log0000000001`.

Backing up to a directory.

Example: (Continuous log backup to a directory)

- `$ ontape -c` Performing continuous backup of logical logs.

File created: /space/TAPE/v11backup/blazer_110_Log0000000006

File created: /space/TAPE/v11backup/blazer_110_Log0000000007

File created: /space/TAPE/v11backup/blazer_110_Log0000000008

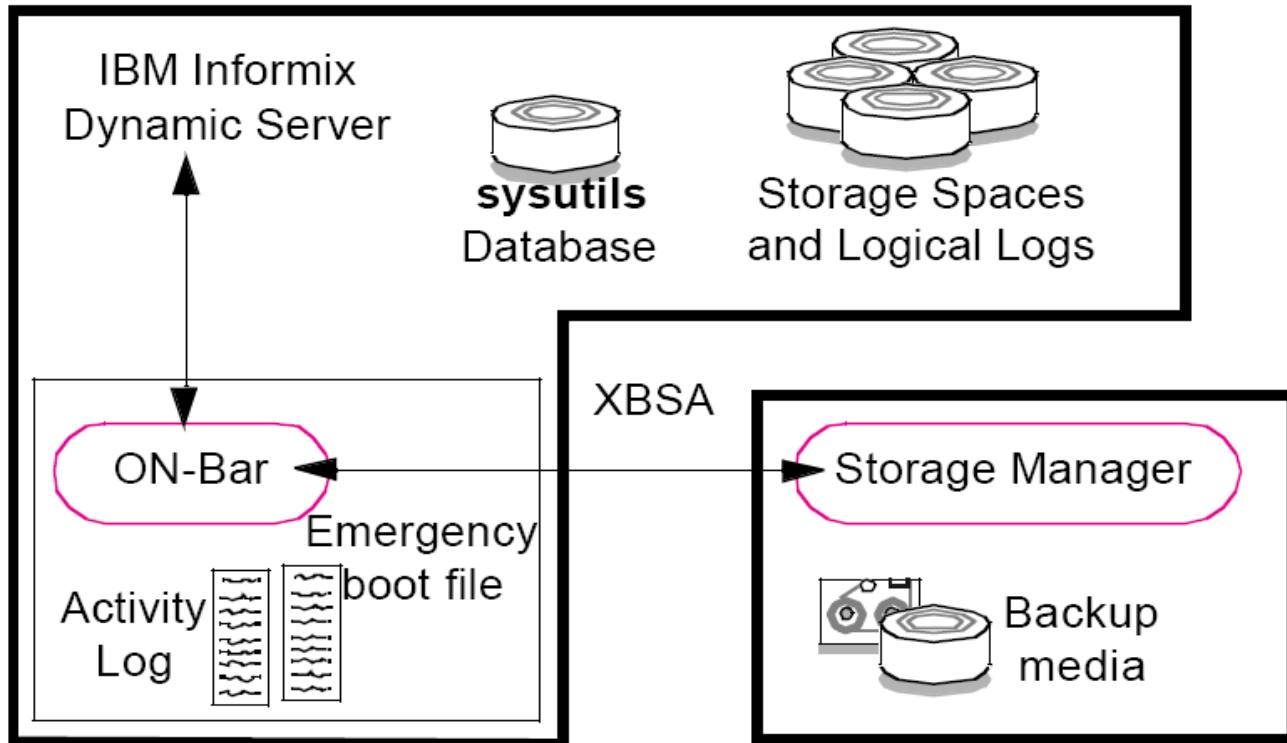
Restoring with ontape.

- `ontape -r` # Full system restore.
- `ontape -p` # Physical restore only and intended for a secondary setup in HDR pair.
- `ontape -p -e` # External ontape restore.

General points about ontape restores:

- The same restore concepts applies as in OnBar (physical, logical, cold, warm, and mixed) restores.
- Ontape backups are always serial,
- Ontape restores **do not** require a logical restore.
- Although you cannot backup specific dbspaces with ontape yet, ontape allows for specific dbspaces to be restored using a warm restore.

Backup and Restore with ON-Bar



Backup and Restore with ON-Bar

The ON-Bar backup and restore procedure includes the following subsystems and their components:

- IBM Informix Dynamic Server
 - ▶ Storage spaces (i.e. dbspaces and blobspaces)
 - ▶ Logical logs
 - ▶ ON-Bar catalog tables in the **sysutils** database

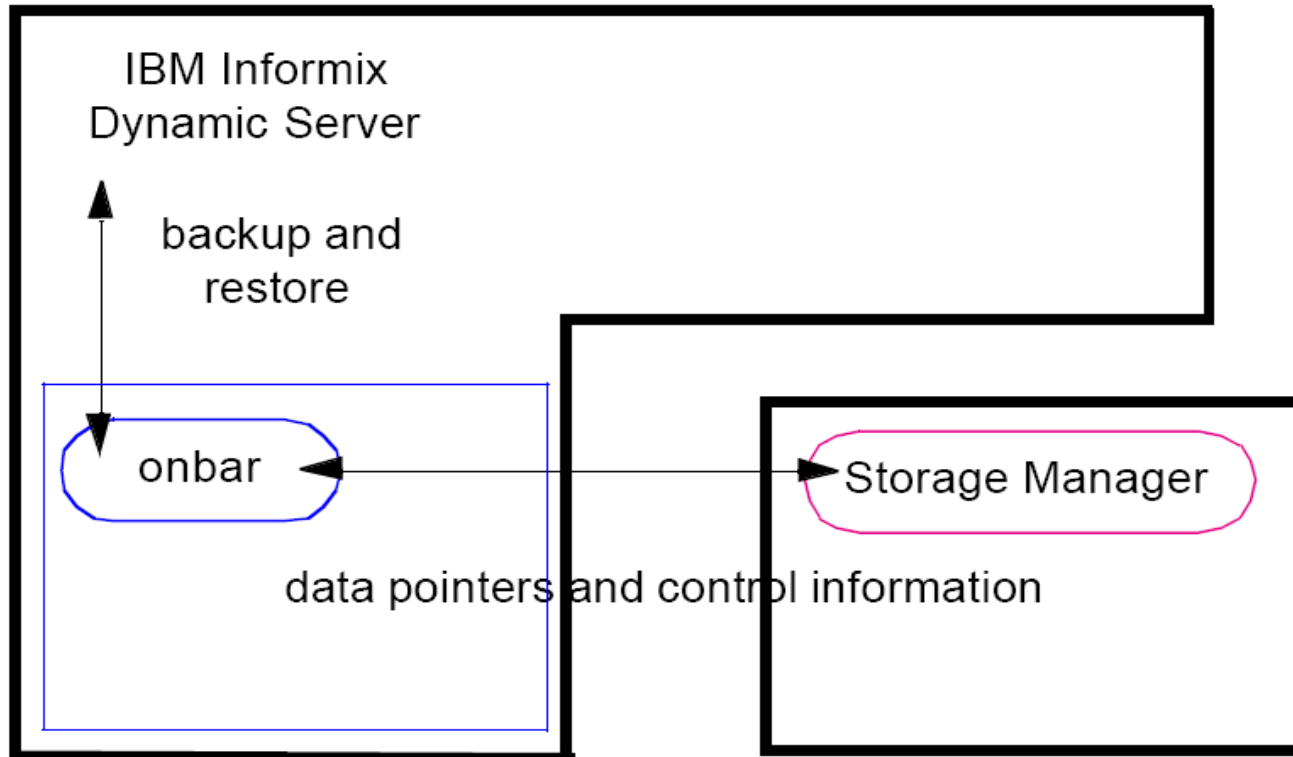
Backup and Restore with ON-Bar

- ON-Bar
 - ▶ ON-Bar activity log
 - ▶ ON-Bar emergency boot file

- Storage Manager
 - ▶ Backup media
 - ▶ XBSA interface shared library

These subsystems will be discussed in the following pages.

The Onbar Program

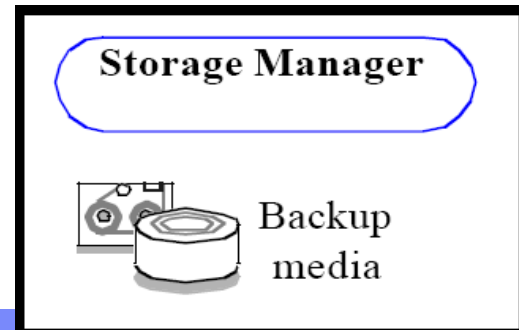


The Onbar Program

- The **onbar** program communicates with both the server and a storage manager to execute backup and restore requests.
- For each database object being backed up, **onbar** sends requests to the server.
- Then using the XBSA, **onbar** passes information to the storage manager which is responsible for transferring the backup data to storage media.
- ON-Bar performs no I/O to tape or disk devices.

The Storage Manager

- Required as part of the ON-Bar system
- Manages storage devices and media
- Uses XBSA interface to communicate with ON-Bar
- Available to support capabilities including:
 - ▶ Data compression and decompression
 - ▶ Data encryption and decryption
 - ▶ Use of a variety of storage media
 - ▶ Networked and distributed backup and restore
 - ▶ Scheduled backups

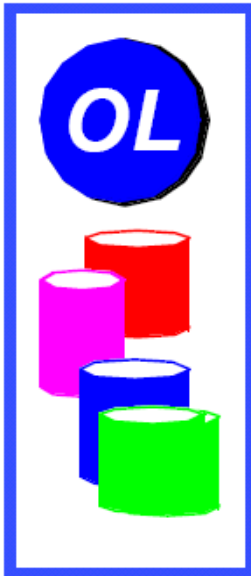


ISM: IBM Informix Storage Manager

Introduced with IDS 7.3, 9.2 and 8.21

- ISM is a *distinct entity* separate from the database server and ONBar
- ISM manages:
 - ▶ Backed-up database server data and logical logs
 - ▶ Backup media
 - ▶ Movement of data (I/O) from the server shared memory to the backup media
- Retention period for data and log backups
- Administered using the ISM Administrator

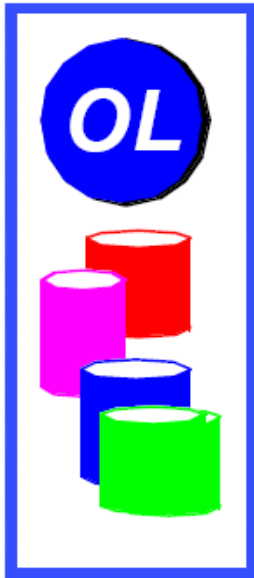
Types of Database Server Backups



IBM Informix provides four types of backups:

- Dbspace
- Whole-system
- External
- Fake

Dbospace and Whole System Backups



Dbospace backup

- + All or selected dbspaces
- + Dbspaces may be backed up in parallel
- + Requires logical recovery

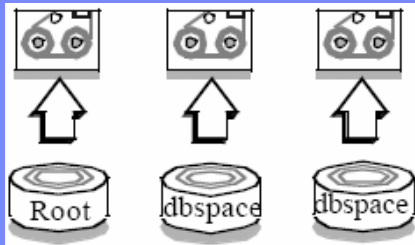
Whole-system backup

- + All dbspaces
- + Starting with IDS v.11 dbspaces backed up in parallel
- + Does **NOT** require logical recovery

Performing Backups

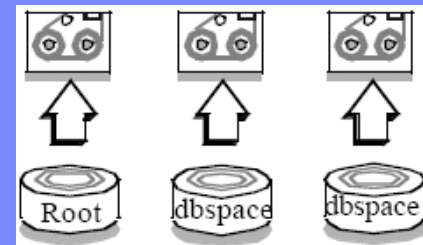
To perform a dbspace backup:

- `onbar -b`
- `onbar -b dbs1 dbs2`
- `onbar -b -f file_name`

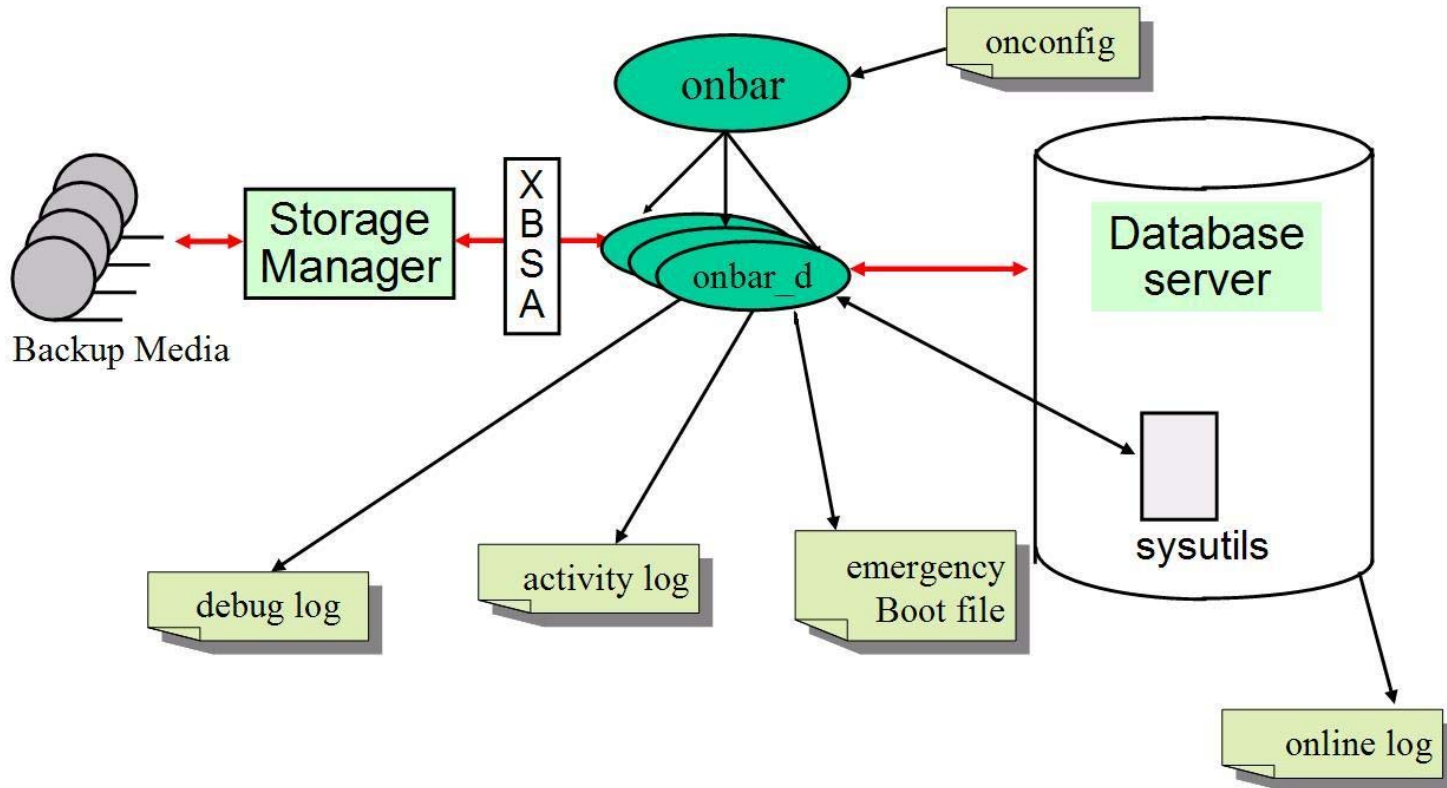


To perform a whole-system backup:

- `onbar -b -w`



How Data Backed Up?



Performing Incremental Backups

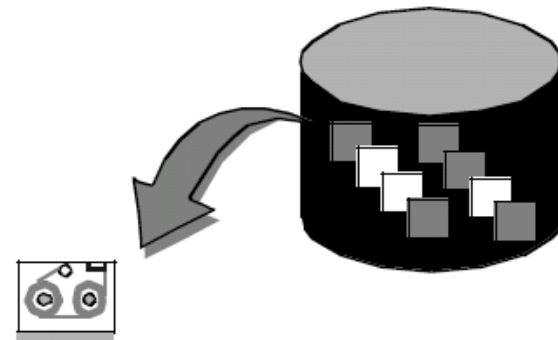
The 'L' flag enables you to specify the backup level from the command prompt.

■ Dbspace backup

- ▶ `onbar -b -L 1`
- ▶ `onbar -b -L 0 dbspace1`
- ▶ `onbar -b -L 2 -f file_name`

■ Whole-system backup

- ▶ `onbar -b -w -L 1`
- ▶ `onbar -b -w -L 2`

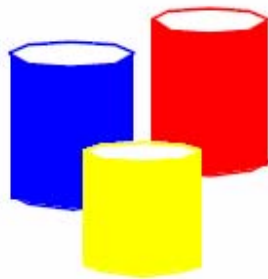


Whole-System Backups

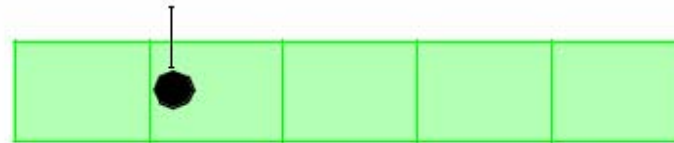
- ✦ rootdbs backed up first and separately
- ✦ The rest of dbspaces backed up in parallel, based on BAR_MAX_BACKUP
- ✦ Order of dbspaces backup depends on the size
- ✦ Largest dbspace will be backed up first and smallest last
- ✦ The dbspaces will be restored in the same order it backed up

Whole-System Backups & Log Records

Why can I restore from a whole-system backup without restoring the logical logs?

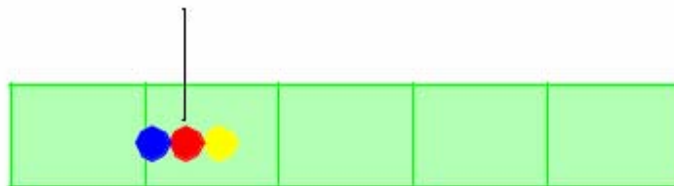


A whole-system backup generates one checkpoint that marks consistency for the entire instance.

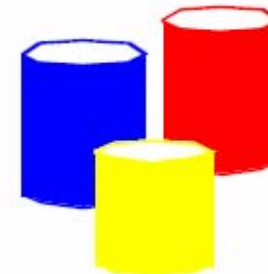


Logical logs during a whole-system backup

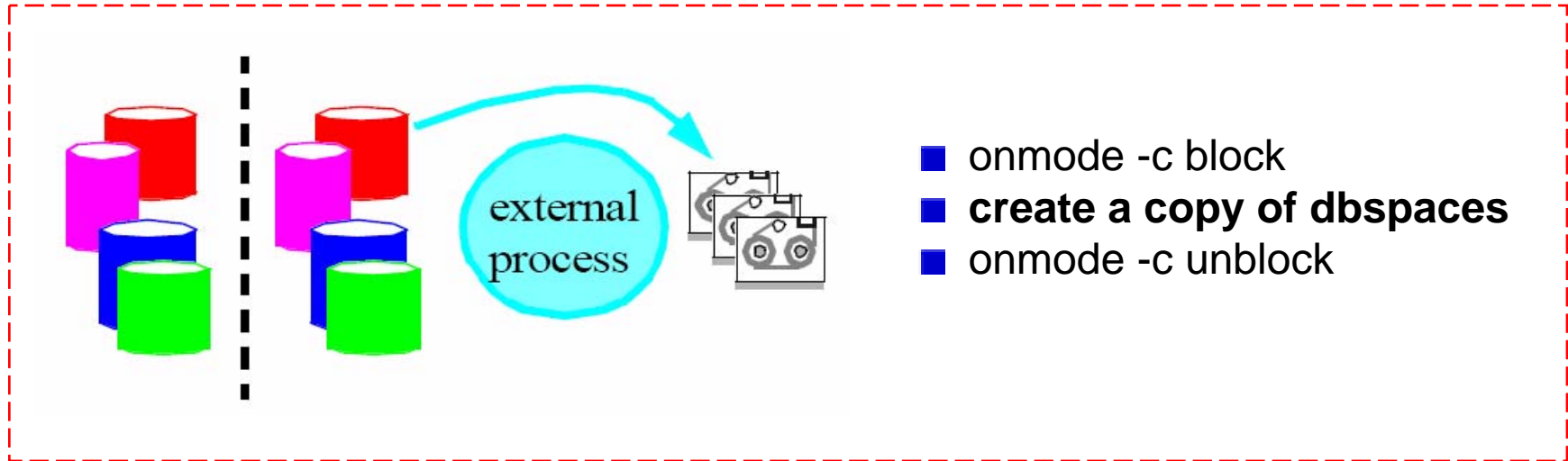
Dbspace backups generate checkpoints for each dbspace backup process.



Logical logs during a dbspace backup



Performing an External Backup



If you wish to perform a logical recovery in conjunction with the restore of an external backup, you must backup your logical logs using the IBM Informix onbar backup and restore utilities. Logical recovery, during an external restore, is required *unless* you are restoring all dbspaces, and all dbspaces were backed up in a single operation.

Performing a Fake Backup



```
onbar -b -F
```

Indicates a
fake
backup

- Facilitates changes in logging mode
- Allows users to use new logs, chunks, or mirrors
- Does not actually back up any data!
- **No restore possible from a fake backup**

The Message File

Contains a log of ON-Bar activity:

- ✦ The default file name is **`/usr/informix/bar_act.log`**
- ✦ Use the `BAR_ACT_LOG` configuration parameter to specify a file name and location
- ✦ Records status, warnings and error messages
- ✦ Displays a list of dbspaces, blobspaces, and logical logs included in a backup or restore
- ✦ Viewing Recent ON-Bar activity:
`onbar -m`

Message Log Format

<i>TIMESTAMP</i>	<i>PROCESS ID</i>	<i>PARENT PROCESS ID</i>	<i>MESSAGE</i>
2007-06-27 18:46:23	15122	8558	onbar -b
2007-06-27 18:46:25	15122	8558	Begin level 0 backup rootdbs.
2007-06-27 18:46:25	15122	8558	Successfully connected to Storage Manager.

Emergency Boot File

- ✦ Is located in `$INFORMIXDIR/etc/ixbar.<servernum>`
- ✦ Contains an entry for each backup object
- ✦ Replaces sysutils database when server is off-line during restore

Server Name	Object	Whole System Backup	Save ID	Timestamp	On-Bar Version	
ids11	rootdbs	R	0 12 0	3201 0	2007-06-08 09:53:28	1 12 ...
ids11	dbspace1	ND	0 13 0	3202 0	2007-06-08 09:53:37	1 13 ...
ids11	23	L	0 14 0	3203 0	2007-06-08 09:53:43	1 0 ...
ids11	24	L	0 15 0	3204 0	2007-06-08 10:33:37	1 0 ...
ids11	26	L	0 22 0	3206 0	2007-06-18 00:08:47	1 0 ...

Object Type Action ID Level First Log Needed for Restore

What is Not Backed Up by ON-Bar?

ON-Bar backups *do not* include:

- Dbspace pages that are allocated but *unused*.
- Pages from mirror chunks, if the corresponding primary chunks are available
- Large objects in blobspaces stored on optical platters
- Temporary dbspaces

Additionally, ON-Bar does not backup server configuration and administrative files, such as the:

- onconfig file and sqlhosts file
- emergency boot file
- oncfg

Why Backup Logical Logs?

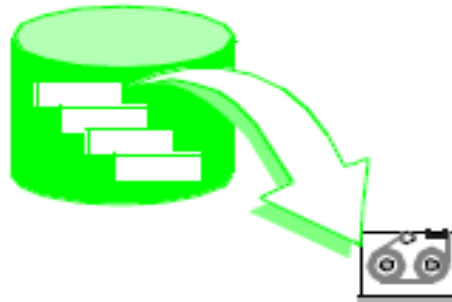
You must perform frequent logical log backups to:

- Provide full transaction recovery
- Prevent the logical logs from filling up and locking up the database server
- Prepare yourself for a failure of the disk that contains logical log information
- Enable you to restore from parallel dbspace backups!

On-Demand Backups of Logical Logs

Logical logs can be backed up **on demand** by using:

- `onbar -b -l` <<< Logical log backup
- `onbar -b -l -c` <<< Backup Current log



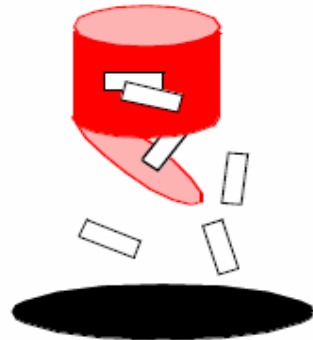
Special Log Considerations

Key points to remember when working with logical logs:

- LTAPEDEV warning
- Salvage logs
- Databases containing Simple Large Objects (BYTE and TEXT data types)

LTAPEDEV Warning

Do *not* set LTAPEDEV to **/dev/null**



Do not set the LTAPEDEV configuration parameter to **/dev/null**.

IDS recognizes **/dev/null** as a special device and, as the logical logs become full, they are freed by the server before ON-Bar is able to back them up. This will cause logical logs to be marked as backed up (**U-B----**) as soon as the server switches to the next log and before ON-Bar has a chance to send the data to a storage manager. If LTAPEDEV is **/dev/null**, only *whole-system restores* can be performed.

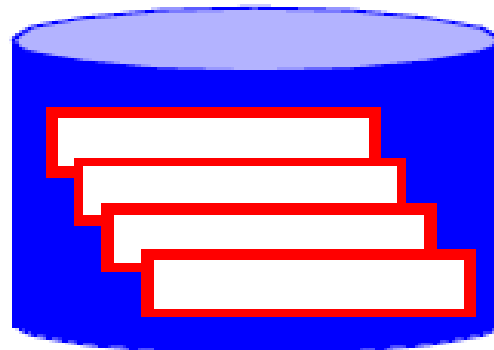
Salvaging Logical Logs

To manually salvage logical logs on disk prior to a restore:

- `onbar -b -l -s`

Useful when you

- perform a *physical only* restore
- replace disks containing logical logs
- restore to a different machine.



Salvaging Logical Logs

When you perform a restore, ON-Bar will automatically attempt to salvage log data that was not backed up at the time of the failure, unless:

- LTAPEDEV is set to /dev/null
- You are performing a *physical only* restore (**onbar -r -p**)

Why Salvage Logical Logs?

After a system failure some logical log data may not have been backed up to disk. This data should be *salvaged* if you wish to recover the system to the point in time of the failure. It should *not* be salvaged if you do not want to replay a log containing errors which resulted in a system failure.

A cold restore after a system failure will automatically attempt to salvage any logs except under the conditions described above. However, the user also has the option of manually salvaging the logs prior to the cold restore.

In IDS.2000, the onbar salvage log command is **onbar -b -l -s**.

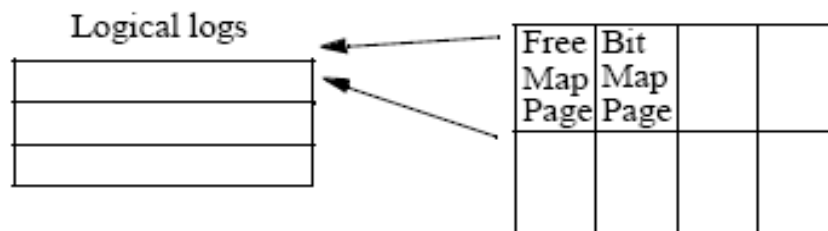
In prior releases, the salvage log command is **onbar -l -s**.

The salvage log command may be useful if a device containing logical log files must be replaced prior to the cold restore process or if you need to restore a system to a different physical machine.

Logical Logs and Simple Large Objects

Special considerations for BYTE and TEXT data stored in blobspaces:

- Back up logical logs to ensure timely reuse of *before image* blobpages.
- Switch logical logs to allow use of newly added chunks.



Restore with onbar

- As we have discussed in previous slides.
- Backup is only good if it can be restored.
- Do the required backup and restore exercises to test your ability in restoring with on-bar.
- Onbar `-r` or onbar `-r -p`
- Point in Time / Point in Log
- Onbar `-r -t datetime` or onbar `-r -n logid`
- Onbar `-r -e` or onbar `-r -e -p` (if no logical restore is required)

Question ?



THANK YOU!