



# Storage Capacity Expansion Plan (initial)

Storage Budget: \$ \$ \$ (5PB)



## Rational:

\* the longer we wait, the more we can buy with the same \$ (hopefully)

## Usage management:

- \* allocations
- \* introduction of quotas & HSM (limited data offload capability)
- \* regular purging

## What have we learned in 1 ½ years of operation?

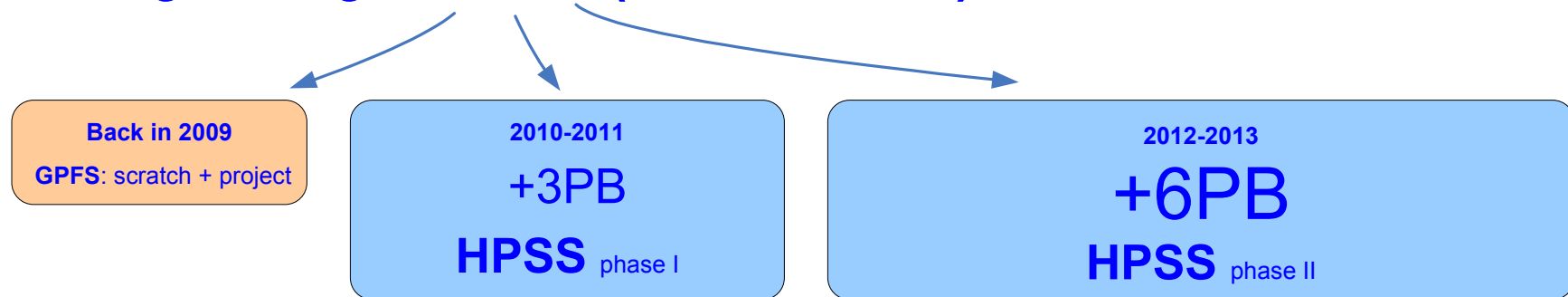
- \* GPFS problems and limitations at our scale (4000 nodes cluster)
- \* user data distribution patterns not GPFS or HPC friendly

## Conclusion:

- \* just adding spinning disks to active filesystems is not a good solution
- \* more users, more data, more files => more problems

# Storage Capacity Expansion Plan (revised)

Storage Budget: \$ \$ \$ (5PB or more)



## Solution:

- \* near online storage with HPSS (tape-backed hierarchical storage system)

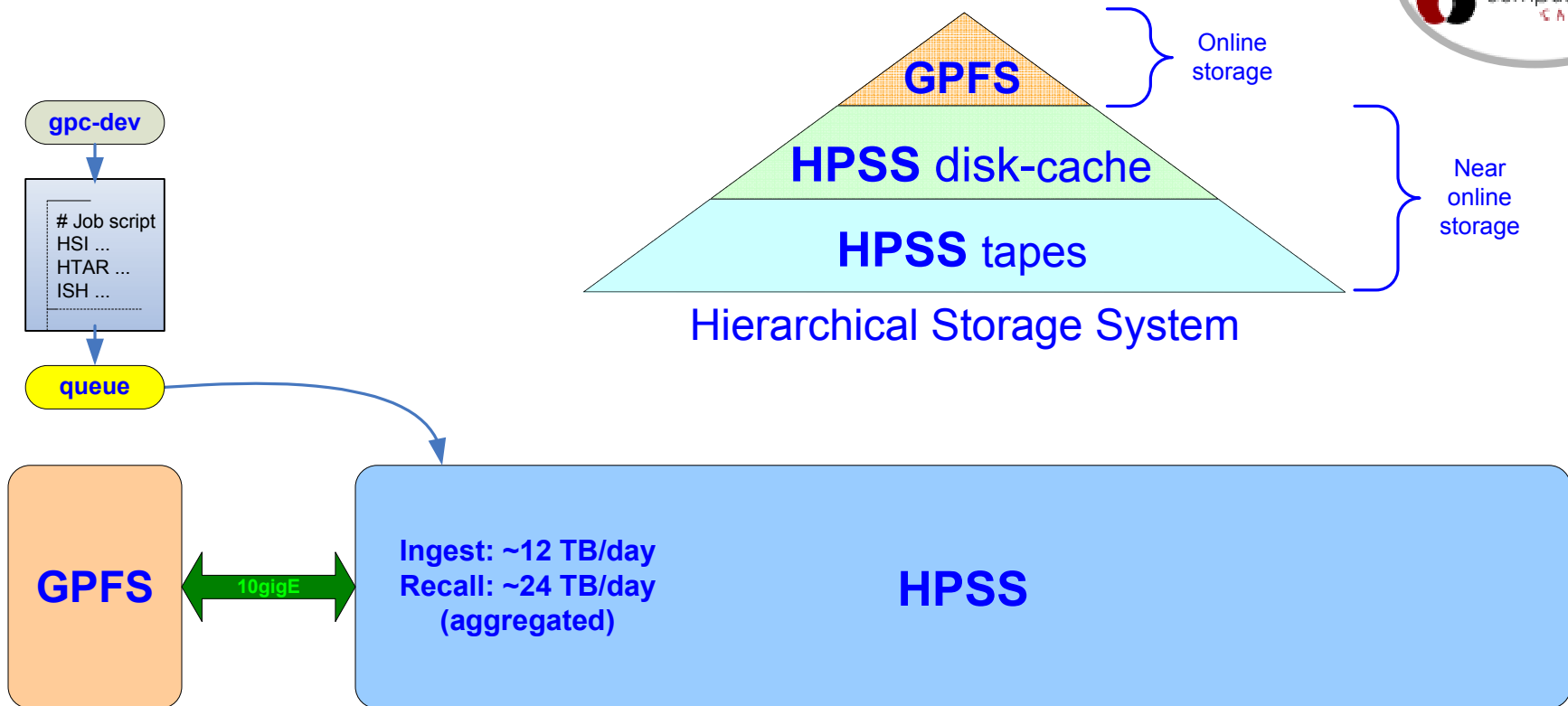
## Usage management:

- \* allocations: GPFS + HPSS (TBD)
- \* quotas & massive data offload to HPSS
- \* regular purging
- \* less utilization of small files
- \* more utilization of tarballs in the regular workflow by users (new campaign)

## About HPSS (High Performance Storage System):

- \* 10+ years history, used by 50+ facilities in the “Top 500” HPC list
- \* very reliable, data redundancy and data insurance built-in.
- \* highly scalable, reasonable performance at SciNet
- \* HSI/HTAR (and ISH) clients also very reliable and used on several HPSS sites.

# How it works ...



- \* access and transfer management through the GPC queue system
- \* end-user interaction via HSI/HTAR/ISH calls from the job scripts

**HSI:** ftp-like functionality & shell-like interface (put, get, ls, pwd, cd, du, rm, ...)

**HTAR:** tar formatted archives directly into HPSS & browsable index file (.idx)

**ISH:** shell-like utility to perform inventory of tarballs contents (by SciNet)



# Scripted File Transfers

File transfers in and out of the HPSS should be scripted into jobs and submitted to the archive queue. Scripts should use the HSI, HTAR and/or ISH clients as in the example below:

```
#!/bin/env bash
#PBS -q archive
#PBS -N hsi_put_file_in_hpss
#PBS -j oe
#PBS -me

/usr/local/bin/hsi -v <<EOF
cput -p /scratch/$USER/workarea/finished-job1.tar.gz : finished-job1.tar.gz
EOF
status=$?
if [ ! $status == 0 ];then
    echo '!!! TRANSFER FAILED !!!'
fi
exit $status
```

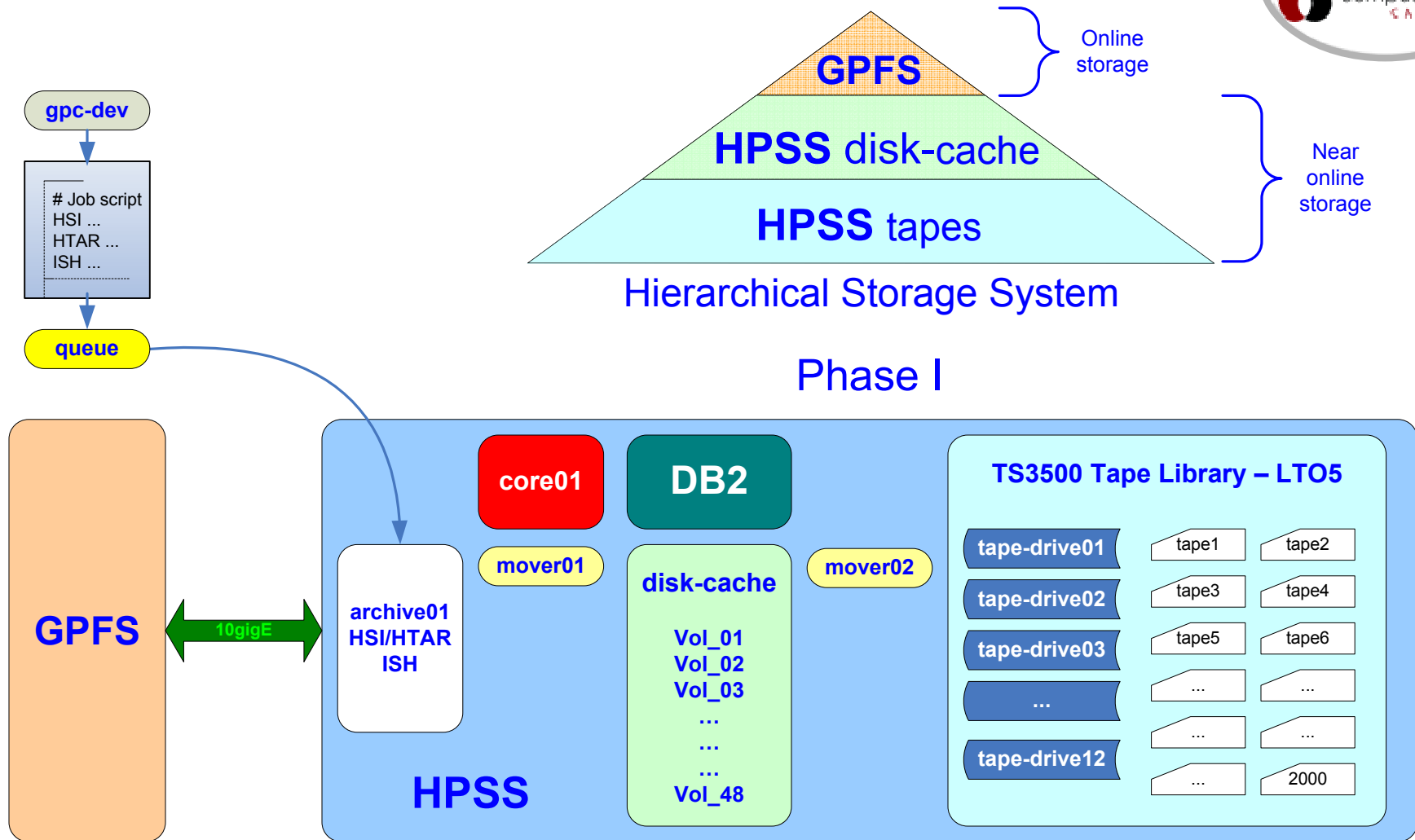
**Note:** Make sure to check the application's **exit code** and returned logs for errors after any data transfer or tarball creation process

The status of pending jobs can be monitored with showq specifying the archive queue:

```
showq -w class=archive
```

For more details and examples please consult the following wiki page:  
<https://support.scinet.utoronto.ca/wiki/index.php/HPSS>

# HPSS – main components



**HPSS** (broad use of the term) = nodes + disks + network + FC + HPSS + DB2 + HSI + HTAR + ISH + Library + tapes + services

# HPSS – scaling potential



Possible Phase II (TBD)

