Veeam Scale-Out Backup Repository

Luca Dell'Oca
Veeam Evangelist
@dellock6

Solving your backup storage issues…

How do you use repositories today?

Name	Type	Host	Path	Capacity	Free
pxy01-FCrep1	Windows	pxy01	O:\	15,0 TB	7,2 TB
pxy01-rep1	Windows	pxy01	E:\	10,0 TB	3,2 TB
pxy01-rep10	Windows	pxy01	N:\	15,0 TB	5,6 TB
pxy01-rep2	Windows	pxy01	F:\	10,0 TB	1,1 TB
pxy01-rep3	Windows	pxy01	G:\	10,3 TB	8,7 TB
pxy01-rep4	Windows	pxy01	H:\	10,5 TB	2,9 TB
pxy01-rep5	Windows	pxy01	I:\	10,0 TB	2,5 TB
pxy01-rep6	Windows	pxy01	J:\	15,0 TB	4,0 TB
pxy01-rep7	Windows	pxy01	K:\	15,0 TB	10,4 TB
pxy01-rep8	Windows	pxy01	L:\	15,0 TB	6,8 TB
pxy01-rep9	Windows	pxy01	M:\	15,0 TB	9,7 TB
pxy02-rep1	Windows	pxy02	E:\	10,0 TB	891,3 GB
pxy02-rep10	Windows	pxy02	N:\	15,0 TB	10,2 TB
pxy02-rep2	Windows	pxy02	F:\	15,0 TB	6,0 TB
pxy02-rep3	Windows	pxy02	G:\	10,0 TB	3,1 TB
pxy02-rep4	Windows	pxy02	H:\	10,0 TB	4,7 TB
pxy02-rep5	Windows	pxy02	I:\	10,0 TB	3,2 TB
pxy02-rep6	Windows	pxy02	J:\	10,0 TB	1,3 TB
pxy02-rep7	Windows	pxy02	K:\	15,0 TB	6,7 TB
pxy02-rep8	Windows	pxy02	L:\	15,0 TB	7,5 TB
pxy02-rep9	Windows	pxy02	M:\	15,0 TB	4,5 TB
pxy03-rep1	Windows	pxy03	E:\	10,0 TB	2,6 TB
pxy03-rep10	Windows	pxy03	N:\	15,0 TB	7,6 TB
pxy03-rep2	Windows	pxy03	F:\	10,0 TB	2,0 TB
pxy03-rep3	Windows	pxy03	G:\	10,0 TB	1,3 TB
pxy03-rep4	Windows	pxy03	H:\	10,0 TB	1,0 TB
pxy03-rep5	Windows	pxy03	I:\	10,0 TB	2,5 TB
pxy03-rep6	Windows	pxy03	J:\	10,0 TB	1,4 TB
pxy03-rep7	Windows	pxy03	K:\	15,0 TB	6,0 TB
pxy03-rep8	Windows	pxy03	L:\	15,0 TB	5,8 TB
pxy03-rep9	Windows	pxy03	M:\	15,0 TB	6,7 TB
pxy04-rep1	Windows	pxy04	E:\	10,0 TB	6,1 TB

- Repositories have a fixed size
- Add multiple repositories to leverage multiple threads and create one job per repository
- A ton of wasted space
- Micro-management of backup jobs to adapt to changes (and relocating backups is painful)

Looking for something better


More and more
demanding RPO and RTO

Data skyrocketing

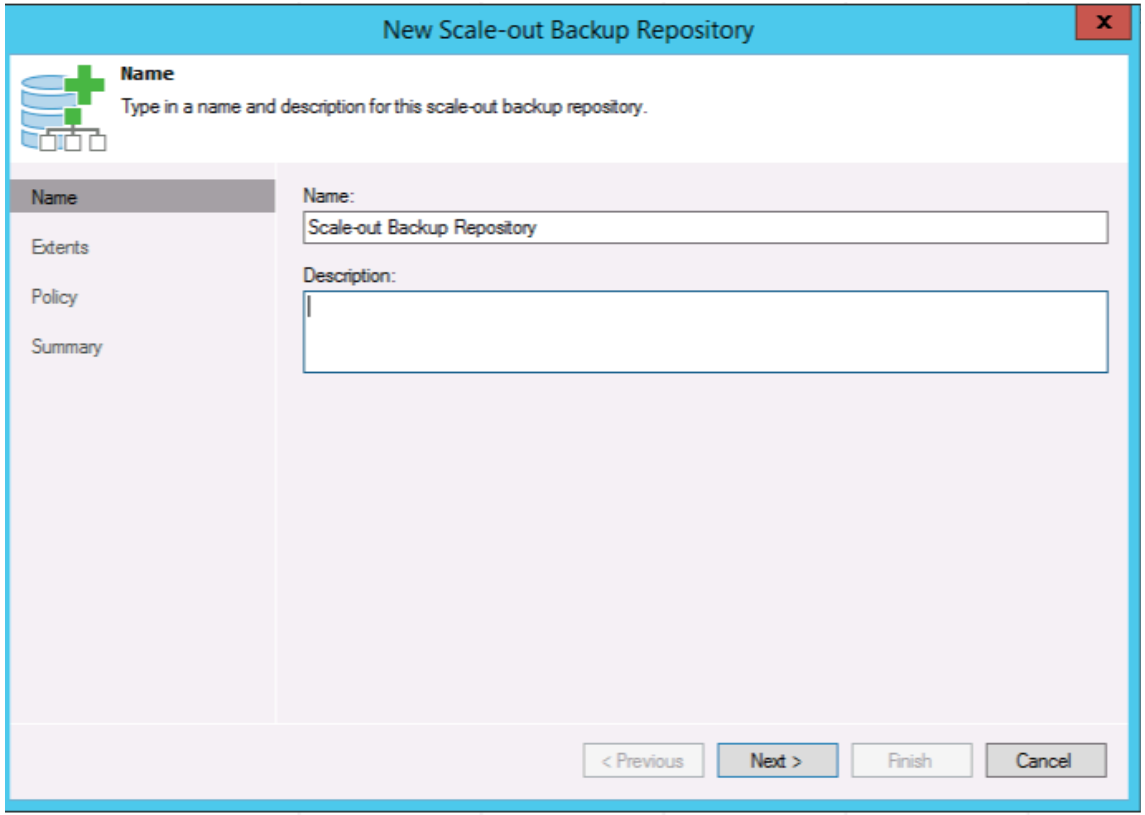

Ease of management

Save budgets!

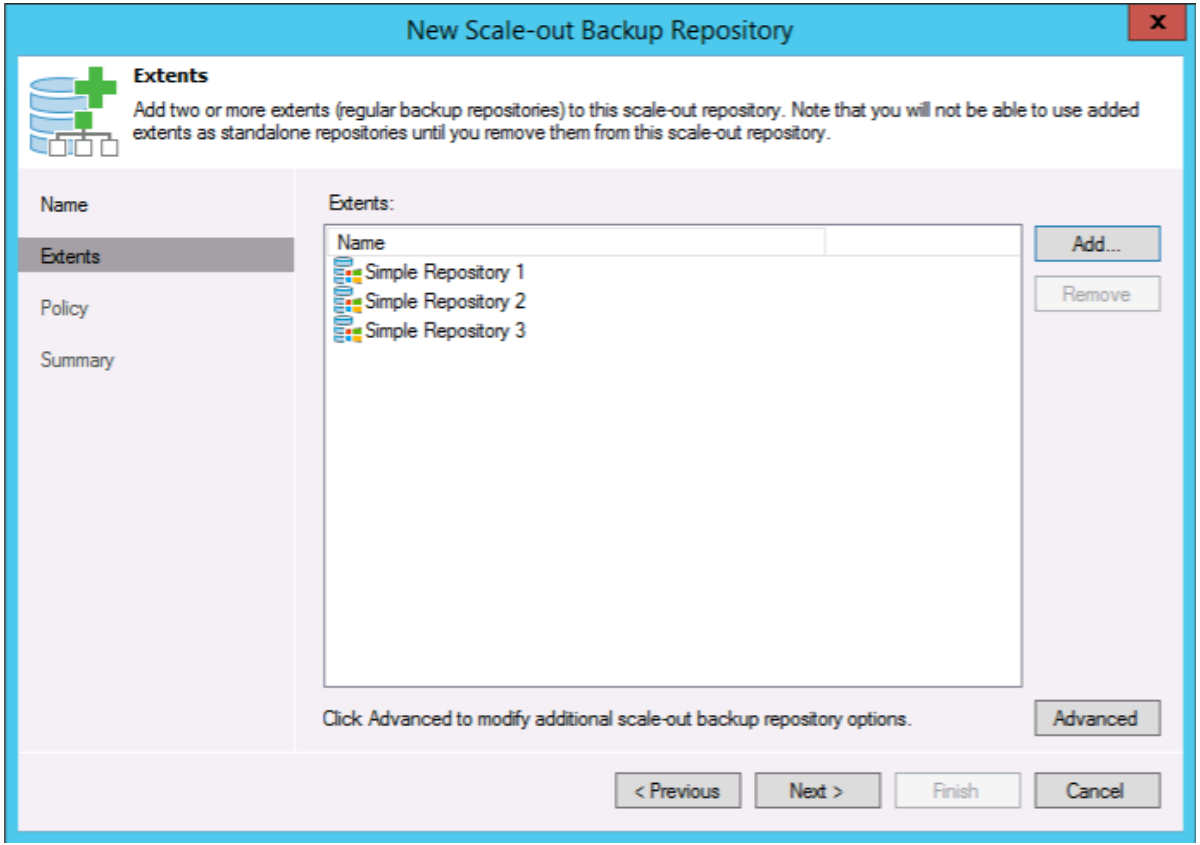
Veeam Scale-Out Backup Repository

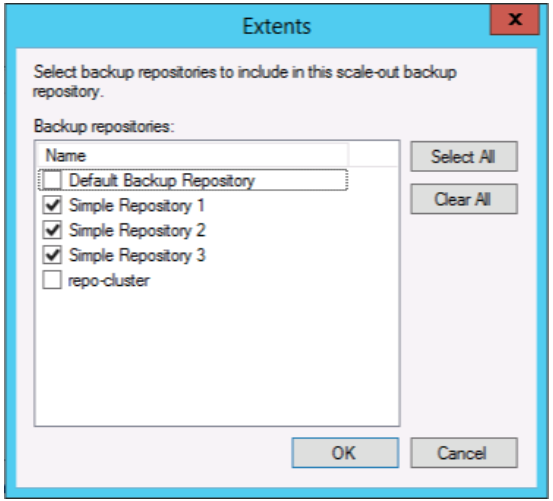
A scale-out repository groups multiple “simple” repositories into a single entity which will then be used as a target for any backup and backup copy job operation.

Super easy setup: Give it a name

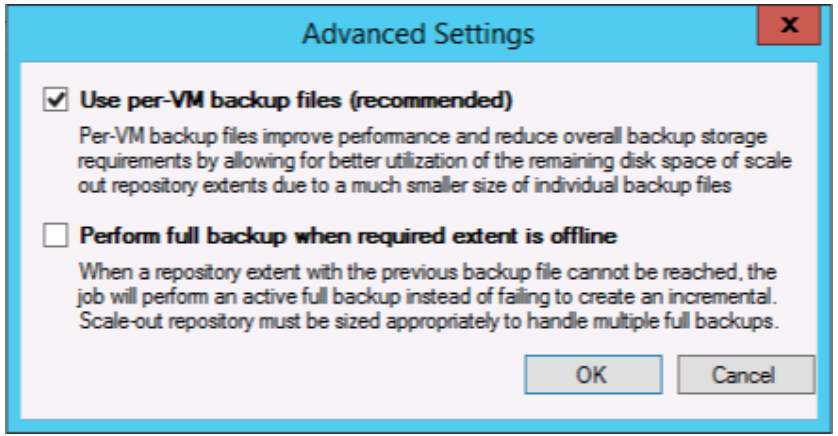


Add repositories as extents

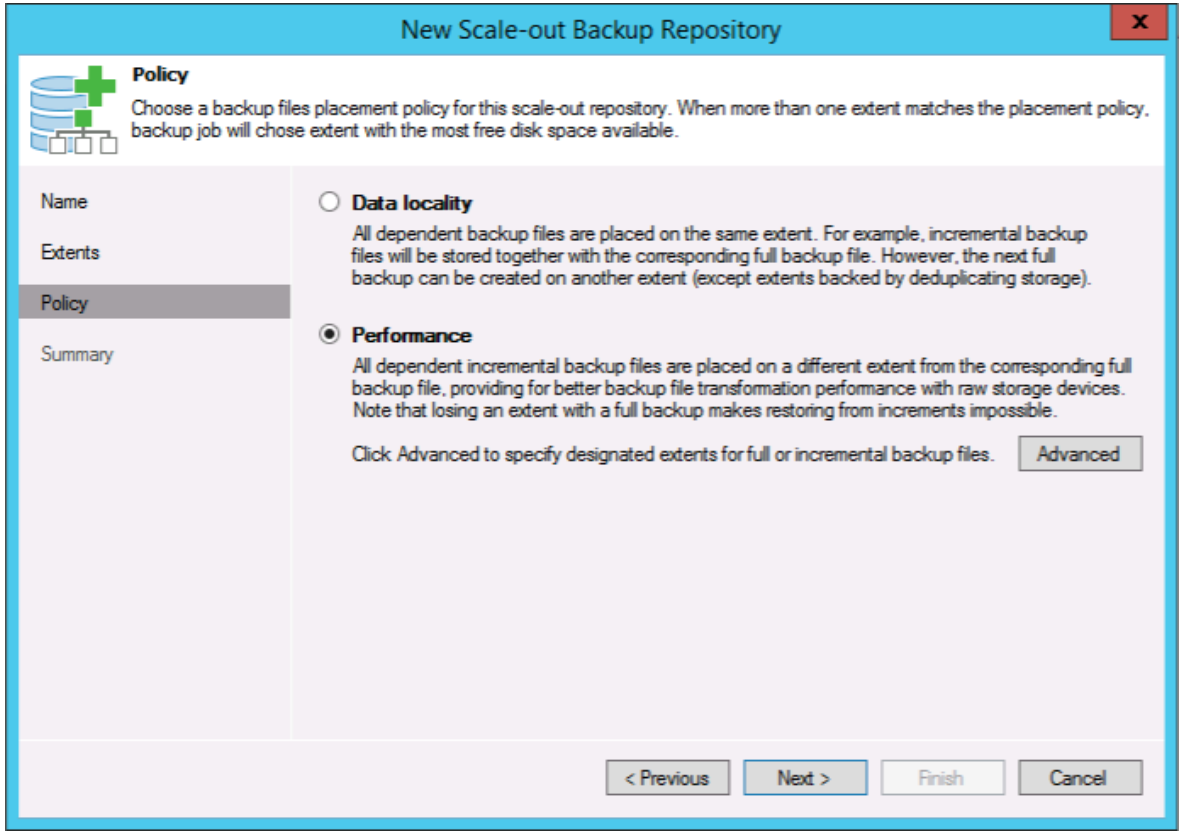




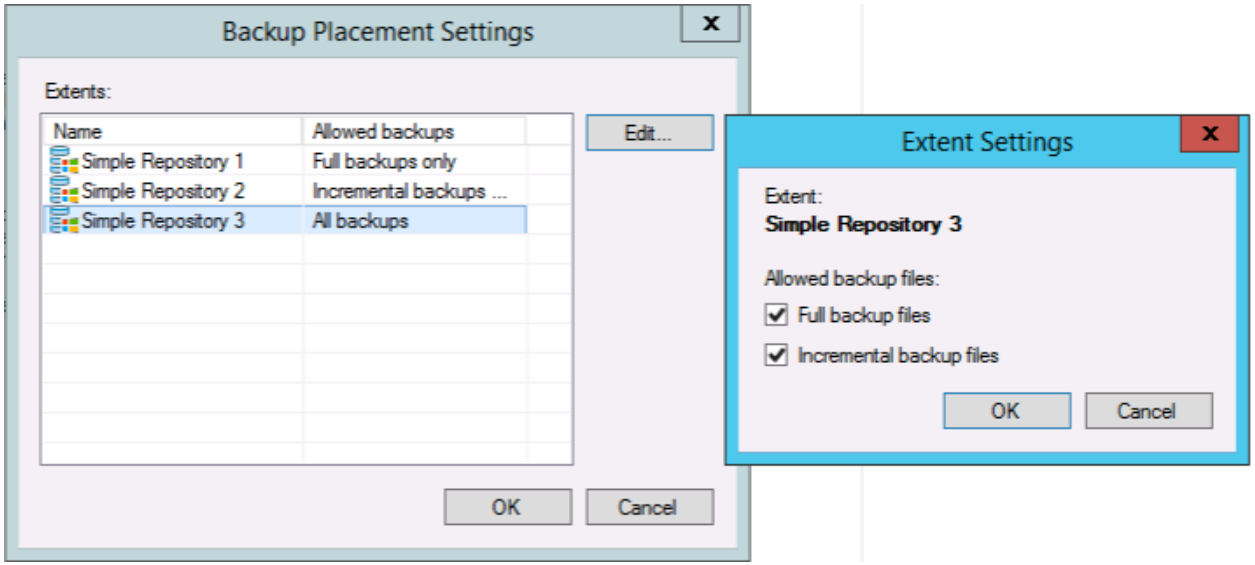
Configure advanced settings



Set placement policies



Performance policy



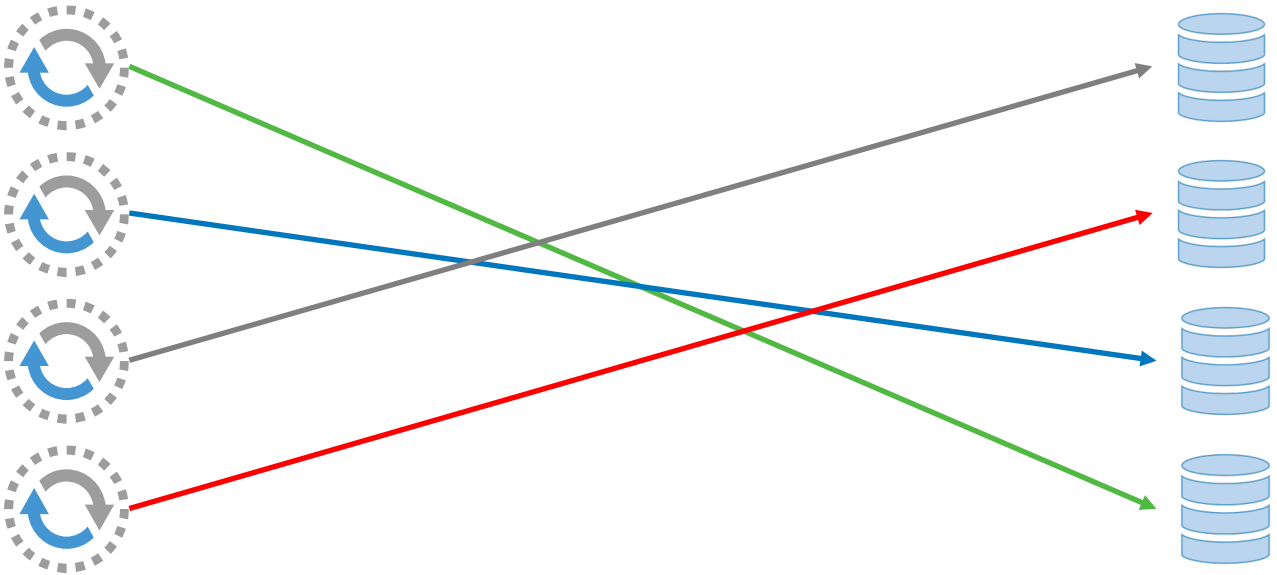
Data locality policy

Data locality

"All dependent backup files are placed on the same extent."

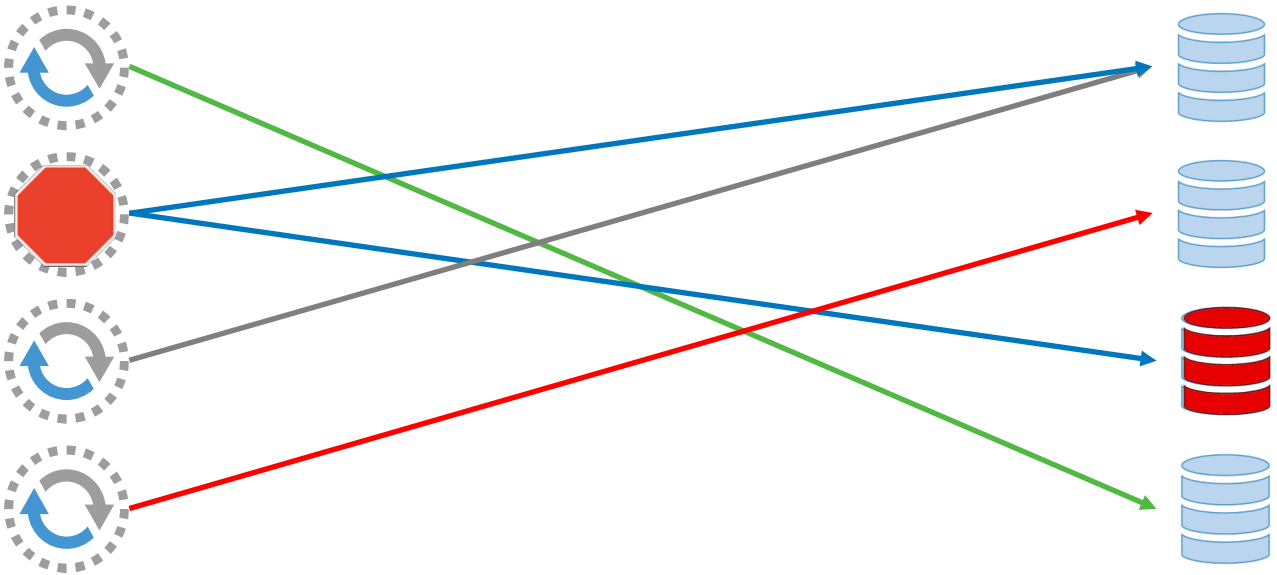
- All extents are treated equally
- Good with multiple units of similar storage
- Easy to extend with another node
- Can failover a full into a new extent if first is full
- A single extent is the failure domain

Job management, before SOBR



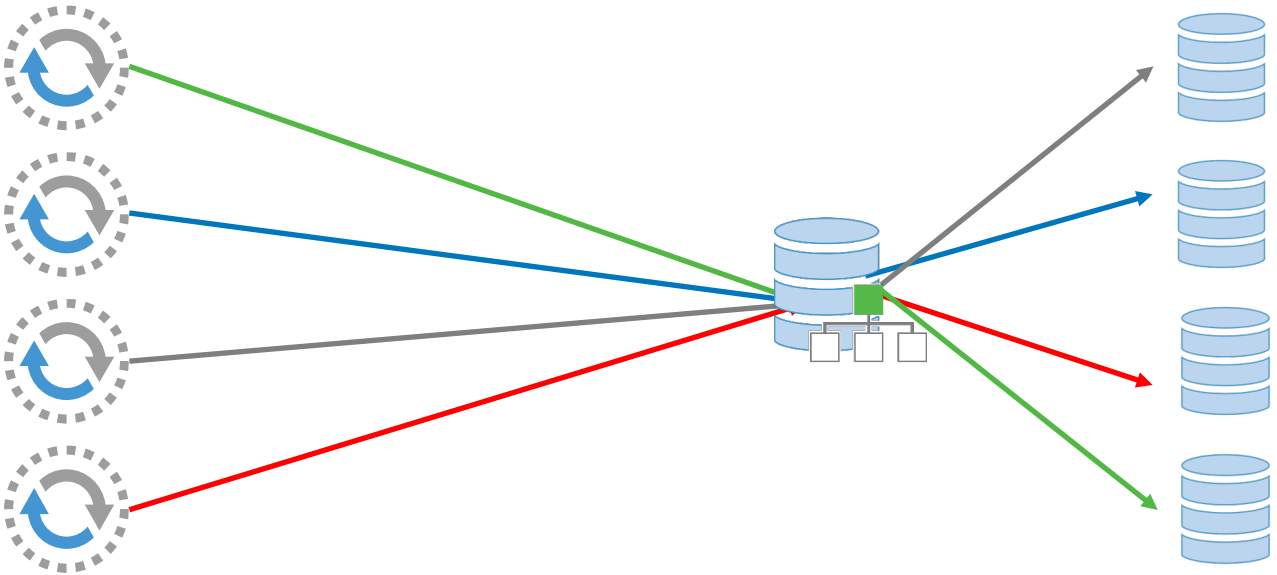
Each job needs to be manually assigned to one repository.

Job management hell



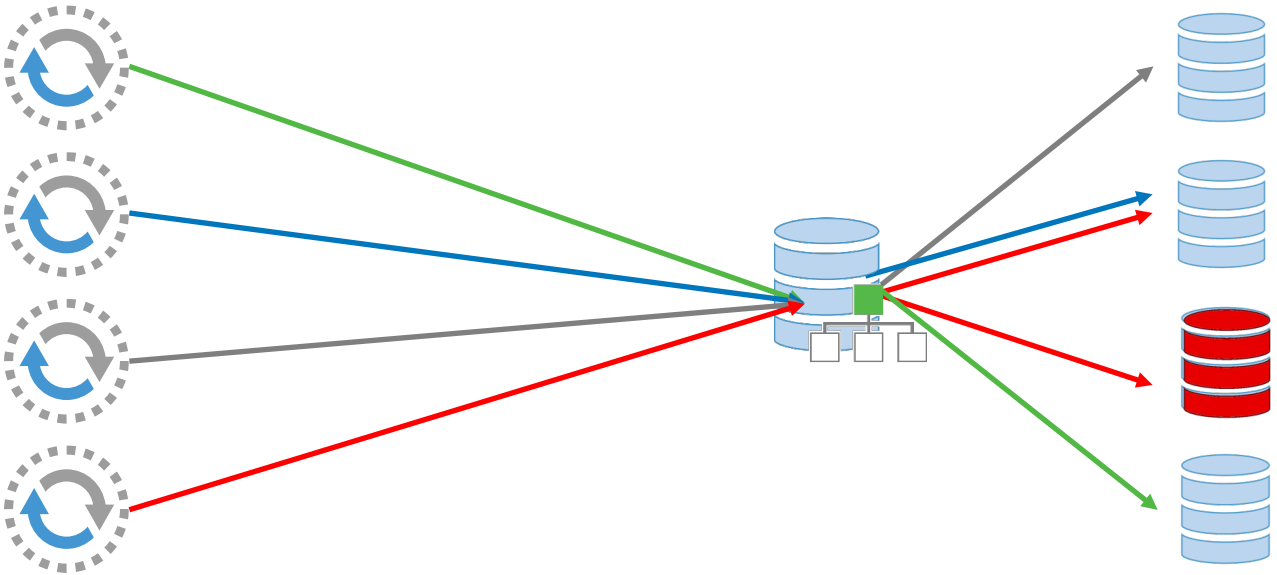
If a repository runs out of space, a job needs to be stopped and relocated.

Automated job management



Jobs are just pointed towards Scale-out Backup Repository, that sends data to the best target.

Automated job management



Any change in the backend is transparent to frontend

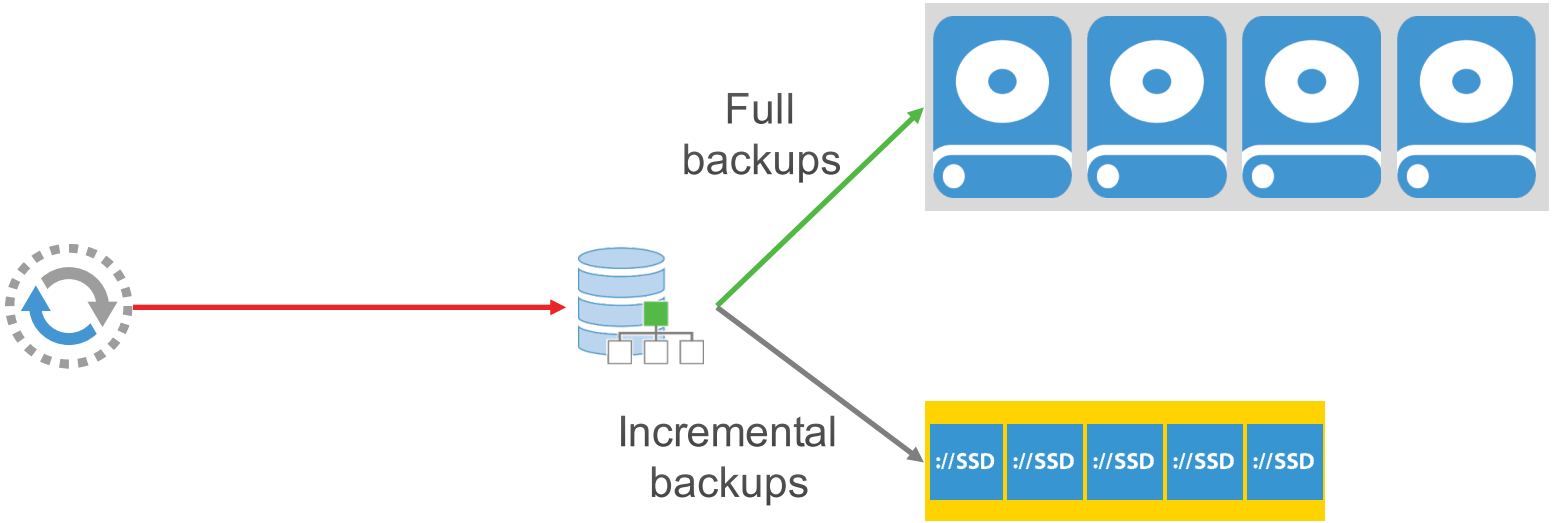
Performance policy

Performance

"All dependent incremental backup files are placed on a different extent from the corresponding full backup file."

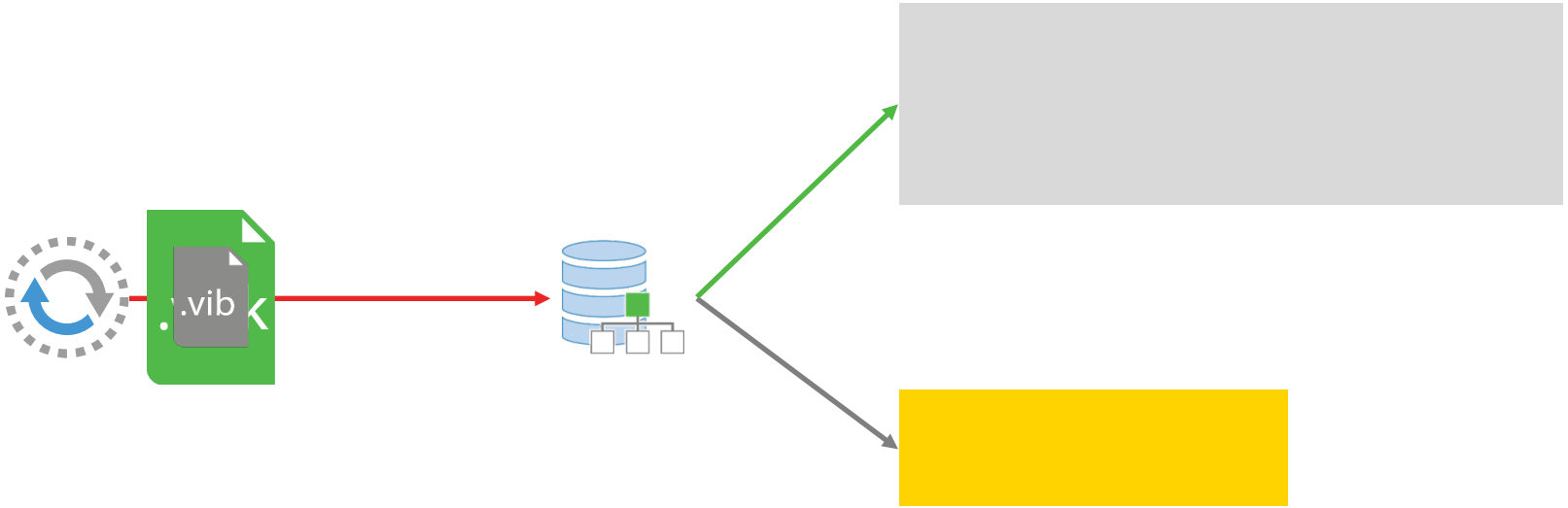
- Great for leveraging storage capabilities
- Mix different storage types
- Assign them full or incremental
- The sum of extents used for a backup chain is the new failure domain

Leverage storage capabilities



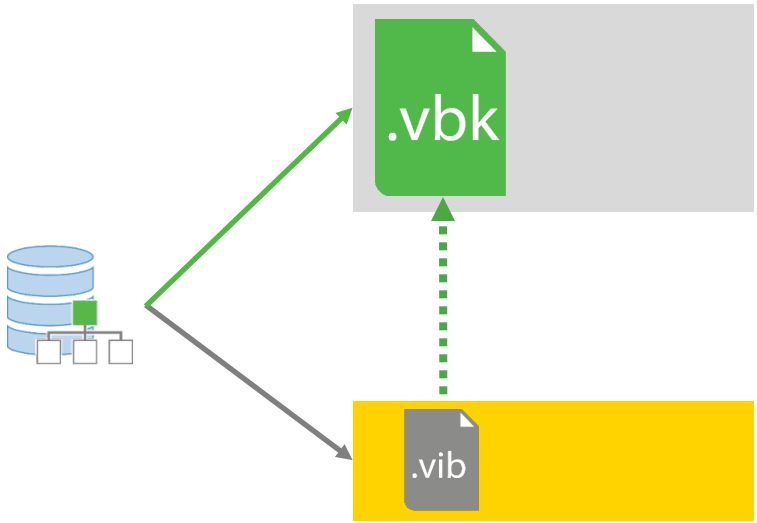
Use fast repositories to ingest incrementals, and large storage systems to host full backups

Leverage storage capabilities



Use fast repositories to ingest incrementals and large storage systems to host full backups

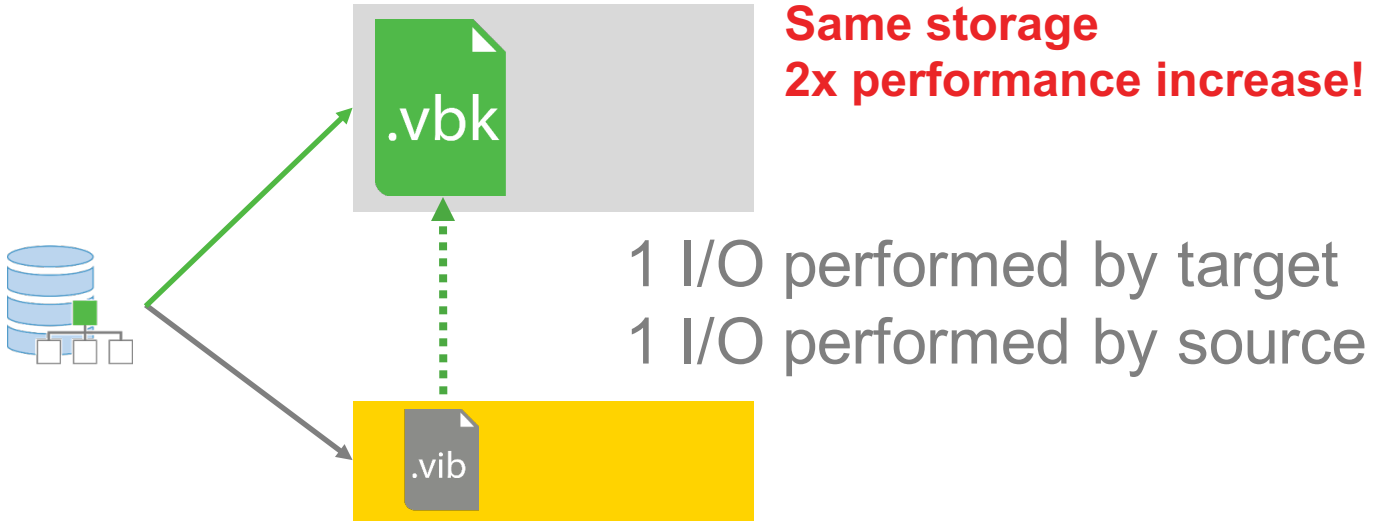
Leverage storage capabilities



Transform uses 2 I/O
For each updated block

Improve dramatically transform operations by spreading I/O to different repositories.

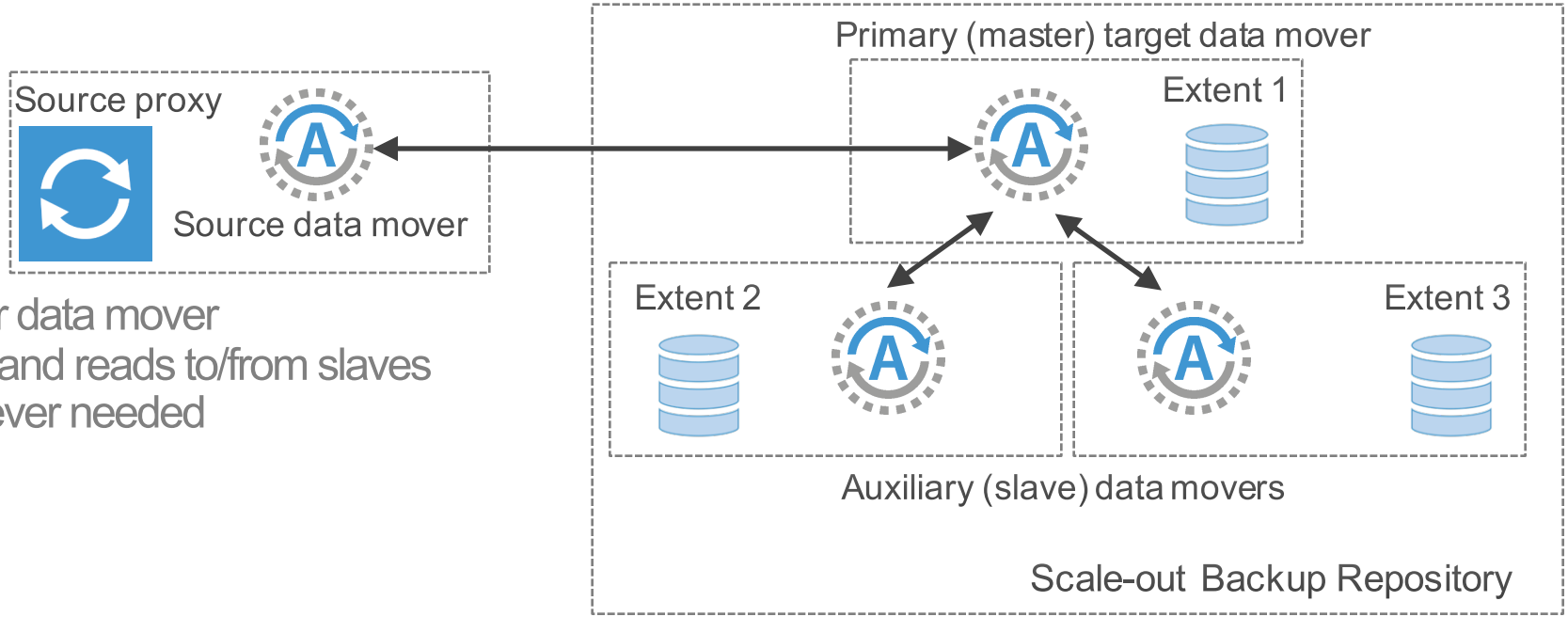
Leverage storage capabilities



Improve and dramatically transform operations by spreading I/O to different repositories

Architecture deep dive

Data movers

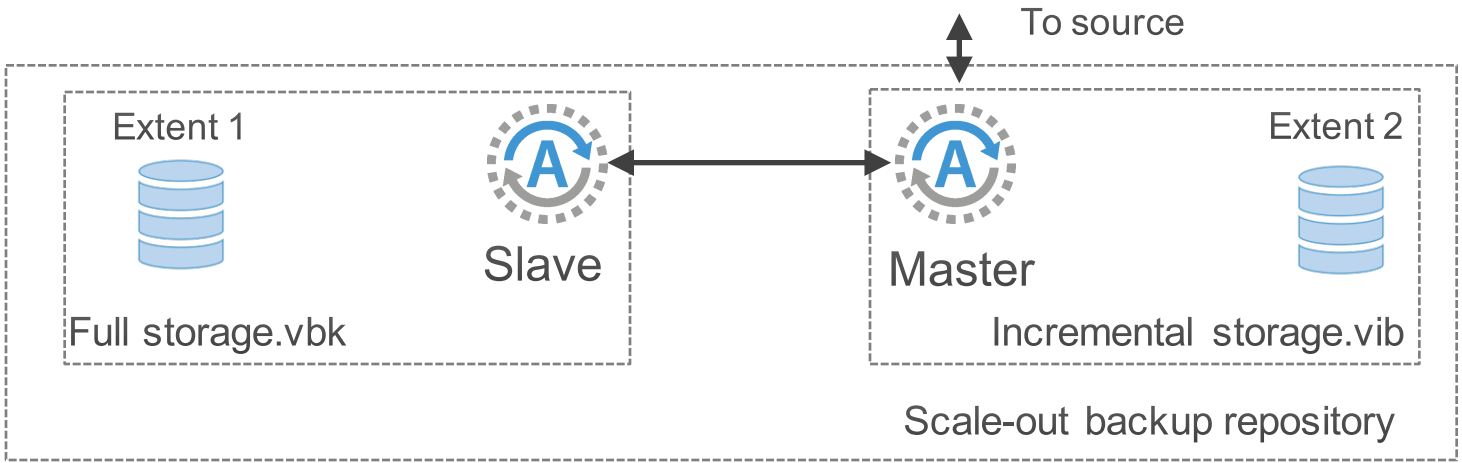


Master data mover
writes and reads to/from slaves
whenever needed

Data movers in backup

Master data mover starts on the host where the write process happens:

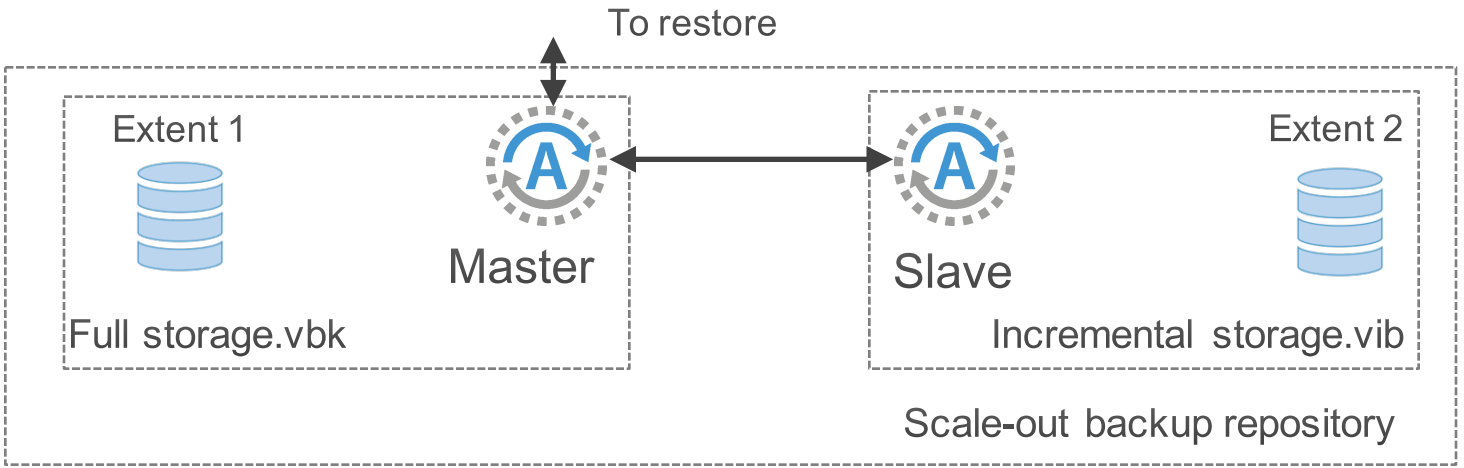
- Forward Incremental mode – incremental storage (vib)
- Reversed incremental mode – full storage (vbk)



- With SMB or DDBoost with automatic gateway server, master data mover is the source proxy.

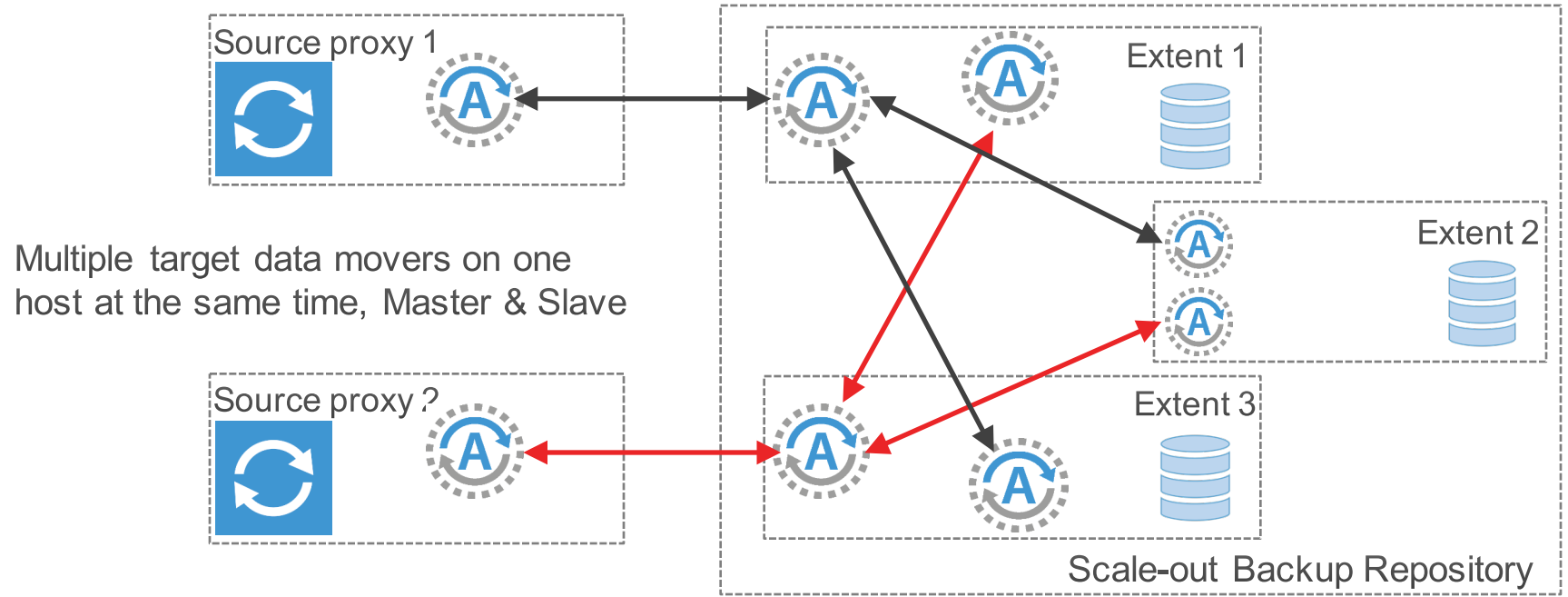
Data movers in restore

Master data mover starts on host with Full storage
(Because most of the data is in it, so there's less traffic on the network)

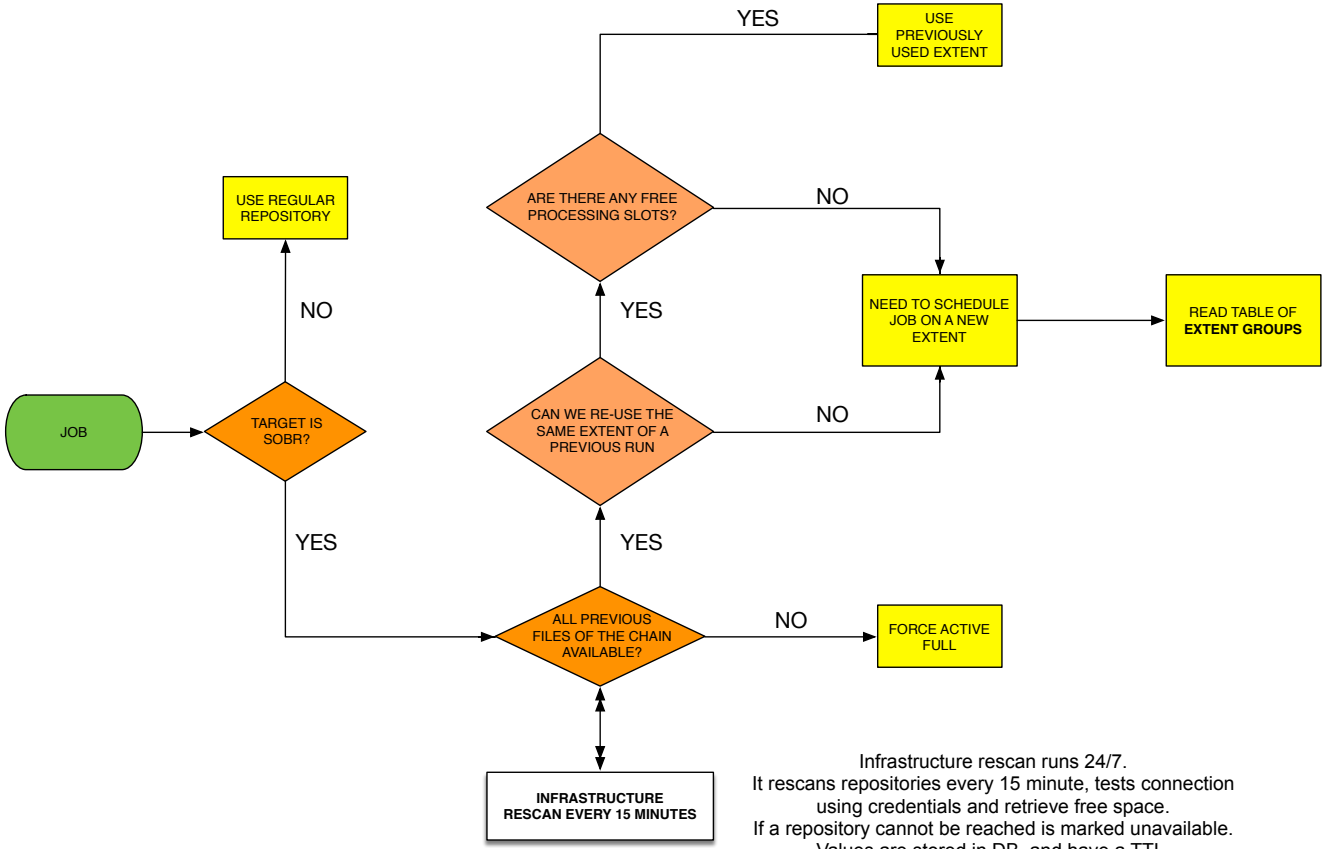


Data movers with per-VM backup chains

Each extent can be selected as Master, depending on backup files layout



A job is started …



Infrastructure rescan runs 24/7.
It rescans repositories every 15 minute, tests connection using credentials and retrieve free space.
If a repository cannot be reached is marked unavailable.
Values are stored in DB, and have a TTL.

A job is started …

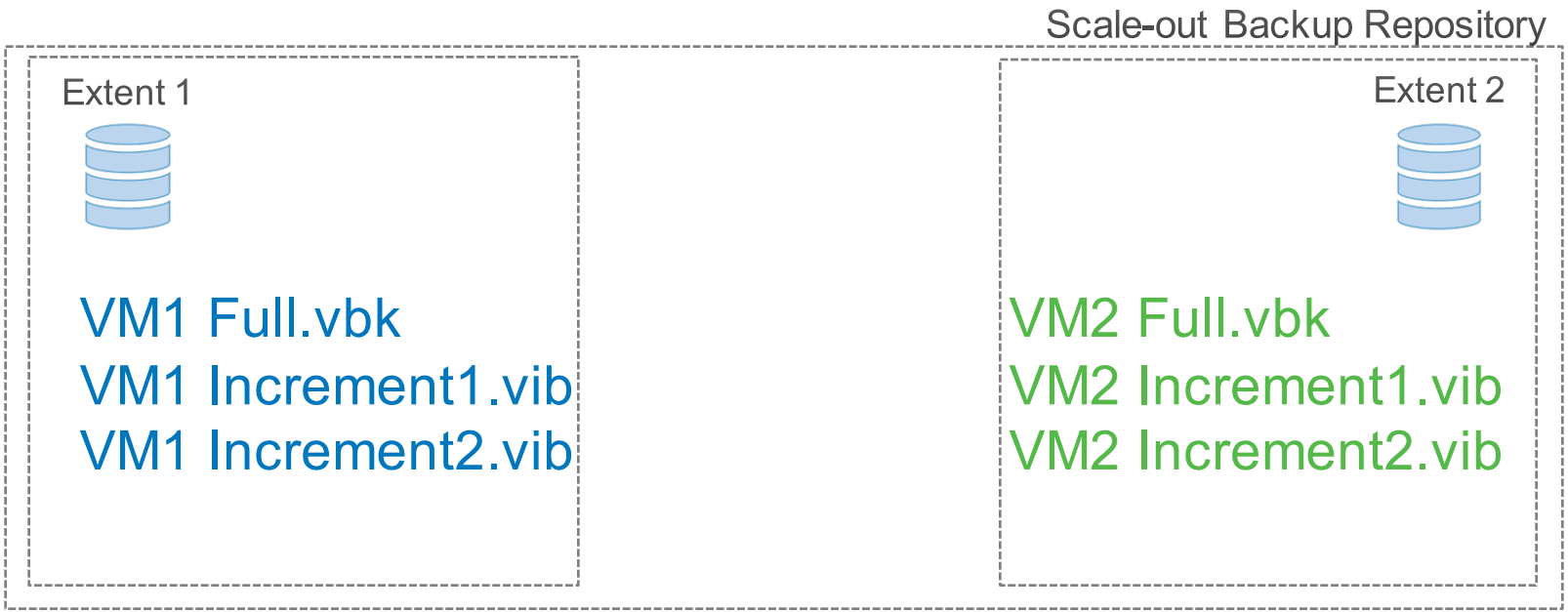
POLICY TYPE	FULL STORAGE		INCREMENTAL STORAGE	
DATA LOCALITY	PREFERRED	Extent that contains a previous full (try to put fulls together only on dedup storages)	PREFERRED	Extent where related full is (that's what this policy about)
	OPTIONAL	All other extents	OPTIONAL	Extent with other increments when we broke the policy (so all increments should be with full, but if we ever broke policy and put an increment somewhere else, we will try to put other increments with it)
	POLICY BREAKING	none	POLICY BREAKING	All other extents
PERFORMANCE	PREFERRED	Extent that contains a previous full (try to put fulls together only on dedup storages)	PREFERRED	Extents with existing increments (try to put all increments together), but not with full vbk ,and allowed for full only (if we ever broke policy)
	OPTIONAL	All extents that allow fulls (specific type option in advanced options)	OPTIONAL	All extents available for increments, except preferred
	POLICY BREAKING	All other extents	POLICY BREAKING	All others (would be ones with full + only full allowed)

A job is started …

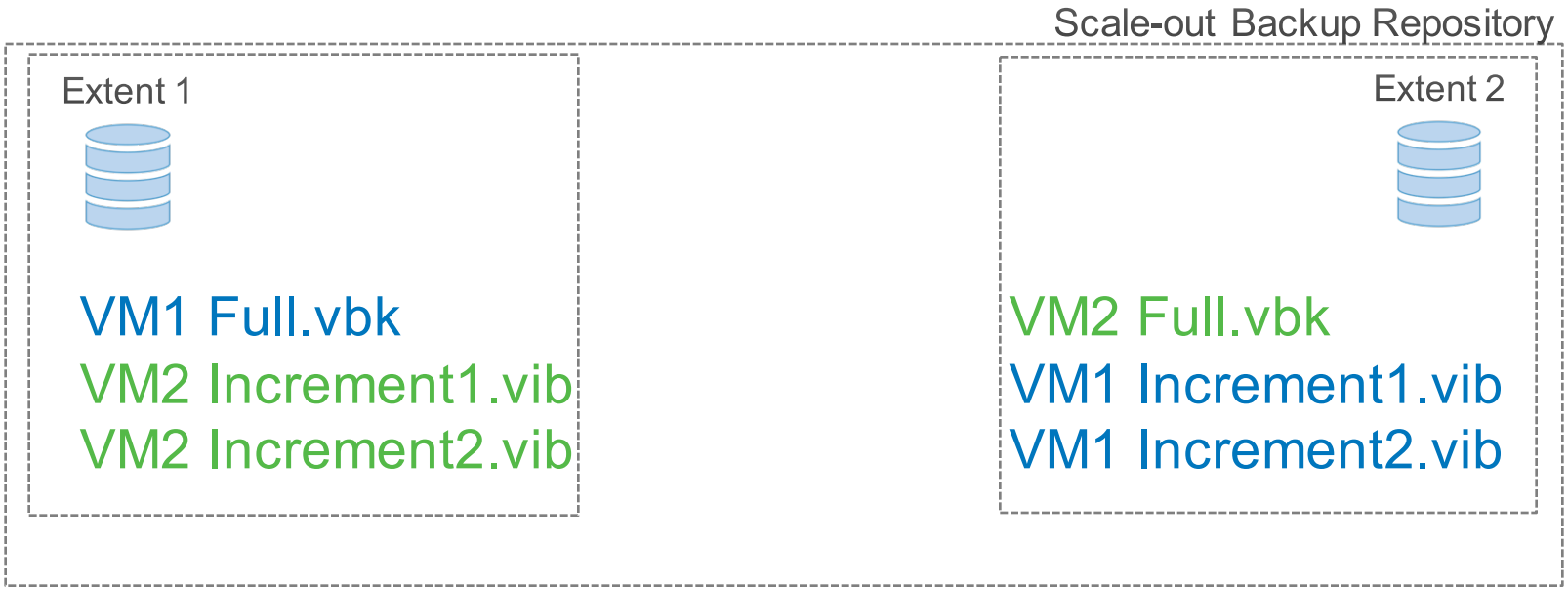
We select the best extent from each group by doing multiple screenings:

1. Skip all unavailable extents
2. Skip all extents with no slots left
   If no extents are left we say, “Resource not ready”
   (Don’t break policy just because no slot is left)
3. Select the extent with most estimated free space available (calculate this value if any other job is using this extent, or use value from the latest rescan)
4. Special for HP StoreOnce, check if it has free file session slots

File structure: Data locality

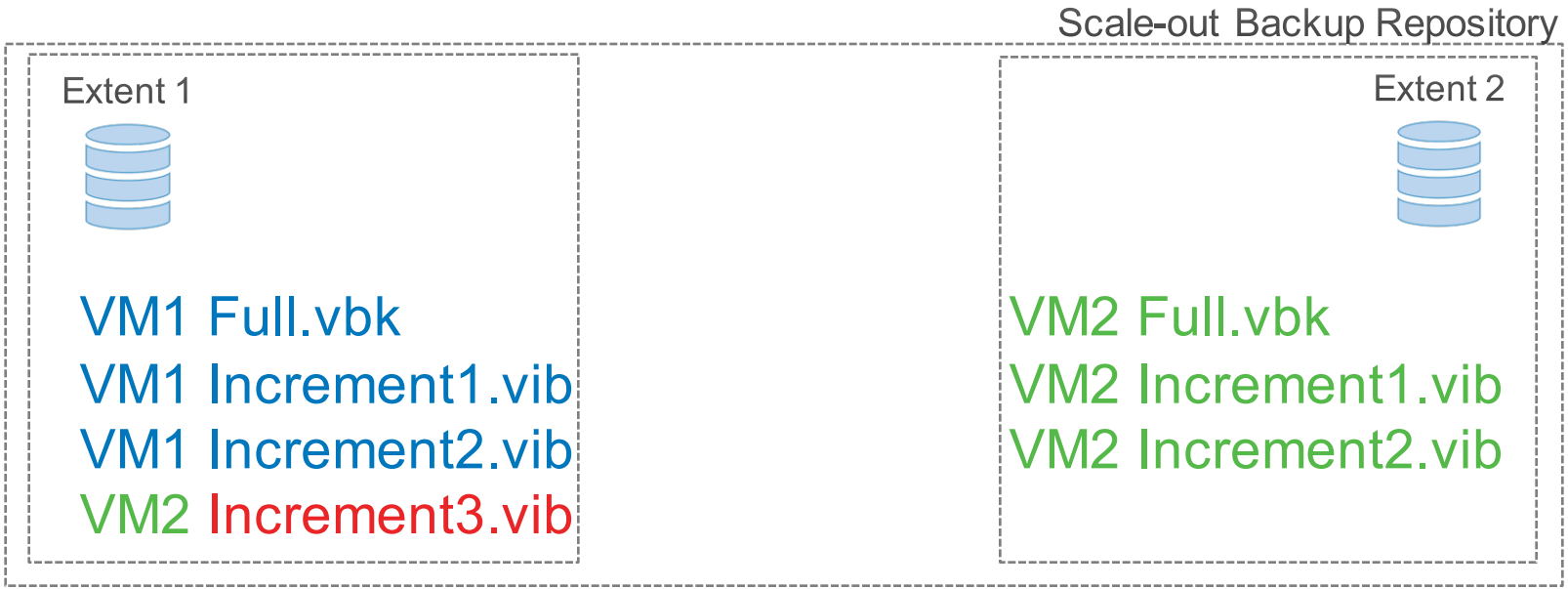


File structure: Performance



Descriptor file (VBM) is stored in multiple copies, each for every extent holding a backup chain file

File structure: Policy breaking



Import existing simple repositories

You can import any supported standard repository

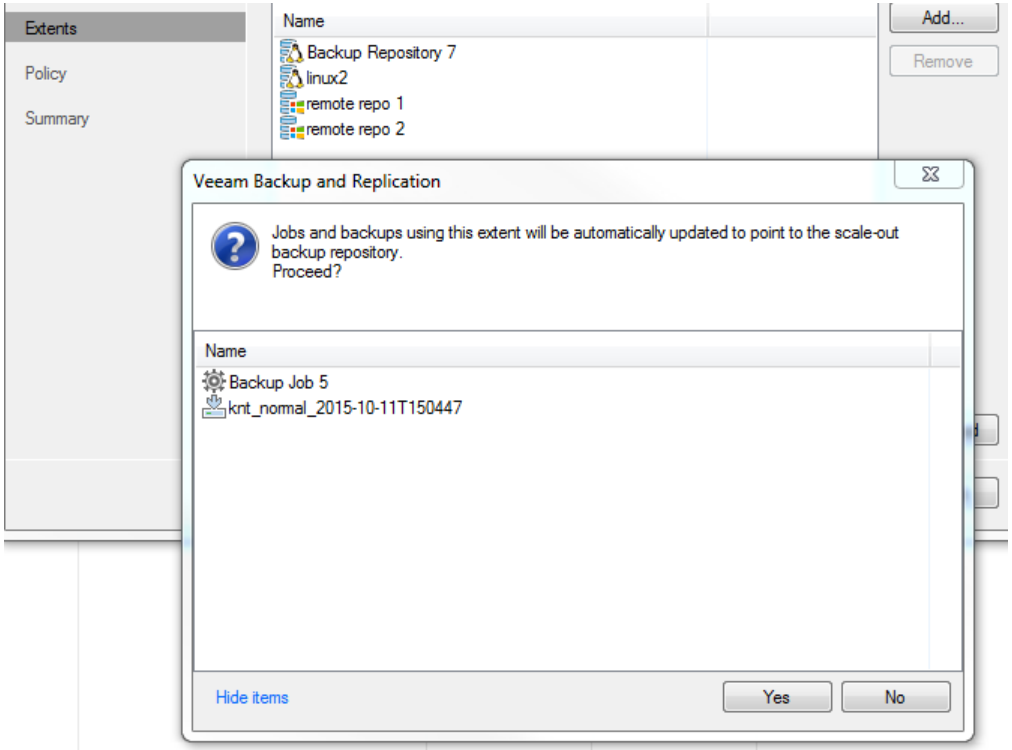
You cannot do nesting or use Cloud Repository inside SOBR

Replica metadata, configuration backup and VM copy jobs cannot use SOBR

Backup set will remain in place — no moving, even if policy would say so. Correct placing happens on the first run of the job after configuring SOBR.

Existing backup job can be remapped to SOBR

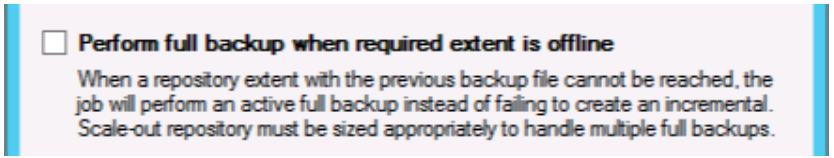
Update existing jobs

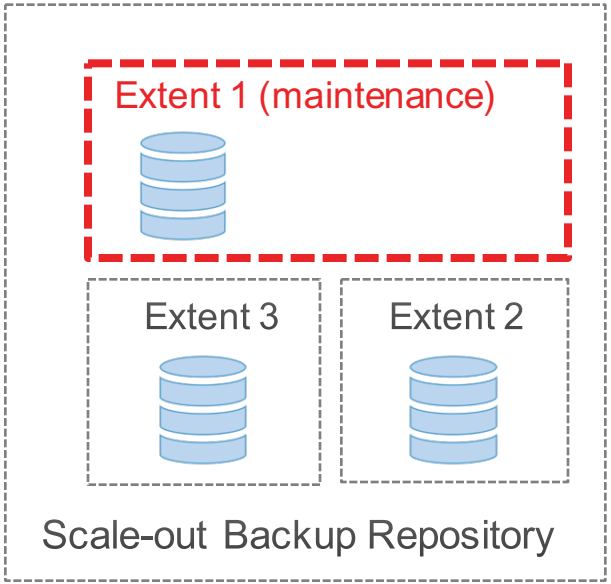


Maintenance

No new tasks of jobs (next VM in the queue) can use this extent. Those that already running can complete.

Be careful, it can trigger the “Force Active Full.”





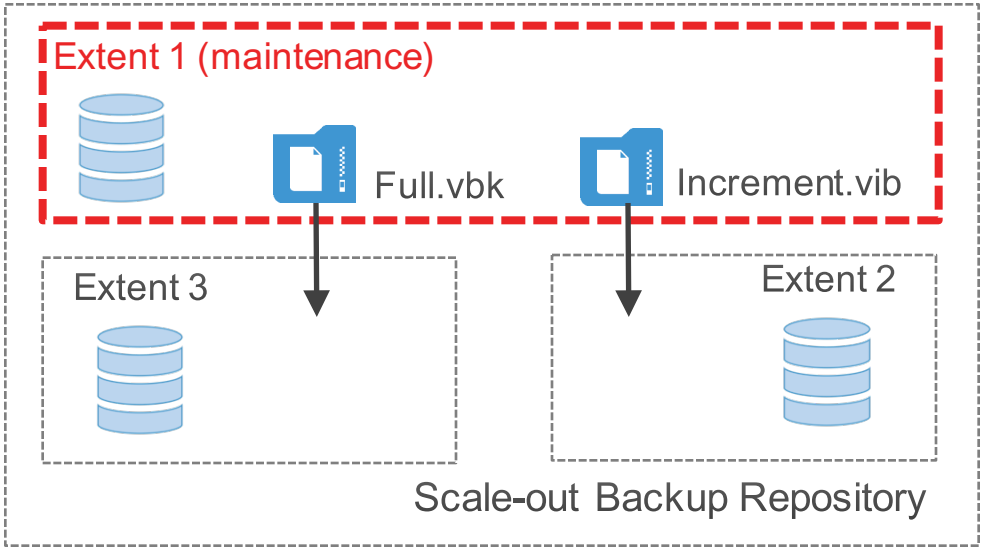
Evacuation

Available only after maintenance is ON

Moves existing backup files to other available extents (vbk first)

Tries as much as possible to adhere to policy

Files are copied: It's not a backup copy



Manual file movement

You can manually move files across extents.

After relocation, just run a rescan. SOBR will update the information about the new location of the files.

It's useful to move few selected files, compared to move all files as with the evacuation.

It's good to fix placement if policy has been broken.

This is prone to human errors; use evacuation if possible.

A few final warnings

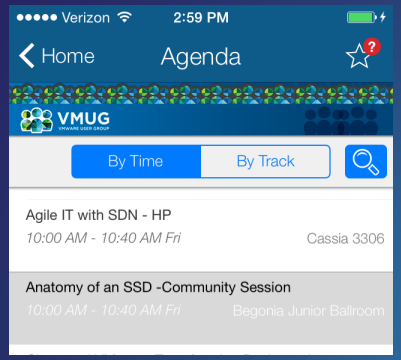
Be careful with deduplicating devices and place them into a Data Locality Policy (they do synthetic operations internally)

It will not support Cloud Connect Backup in v9 (delayed to the next update)

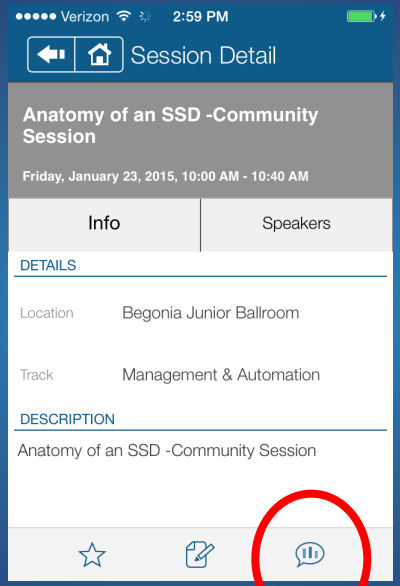
Questions?

VEEAM
IT JUST WORKS!™

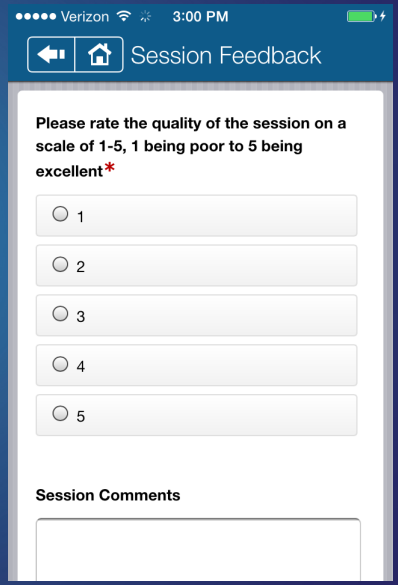
Session Feedback



1. Select your session from the agenda.



2. Click the survey button.



3. Rate your session & submit.

Thank you!

VEEAM
IT JUST WORKS!™