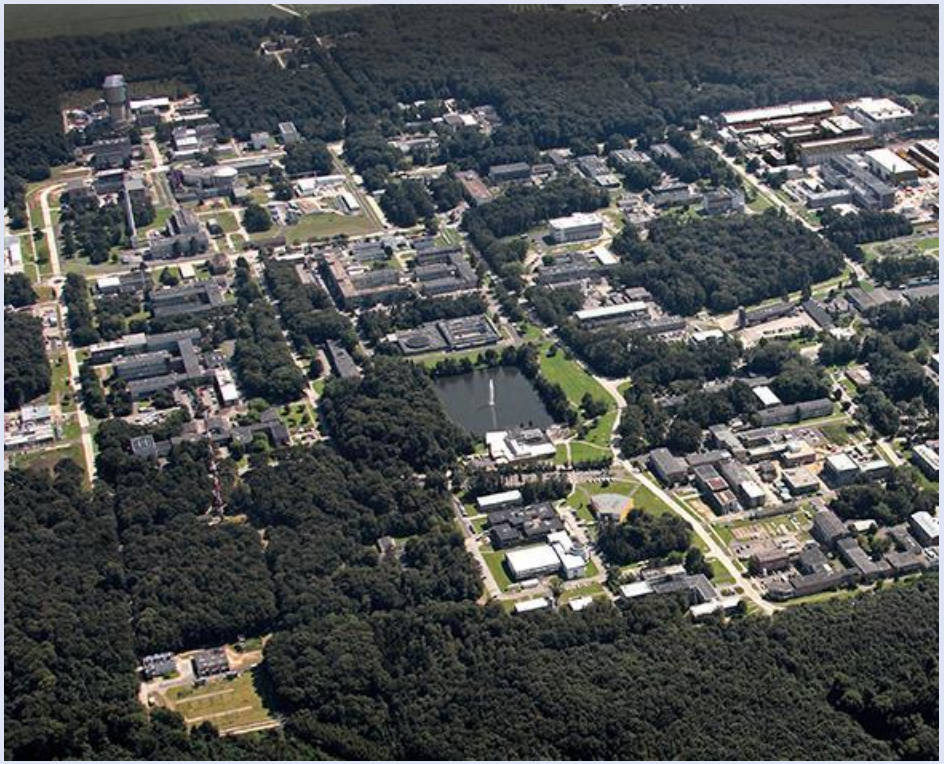
NEW HPC USAGE MODEL @ JÜLICH

MULTI PB USER DATA MIGRATION

MARCH 2019 I MARTIN LISCHEWSKI (JSC)

RESEARCH AND DEVELOPMENT

on 2.2 Square Kilometres



AT A GLANCE

Facts and Figures

1956

FOUNDATION

on 12 December

Shareholders

90 % Federal Republic of Germany

10 % North Rhine-Westphalia

11

INSTITUTES

2 project management organizations

609.3

million euros **REVENUE total**

(40 % external funding)

5,914

EMPLOYEES

2,165 scientists

536 doctoral researchers

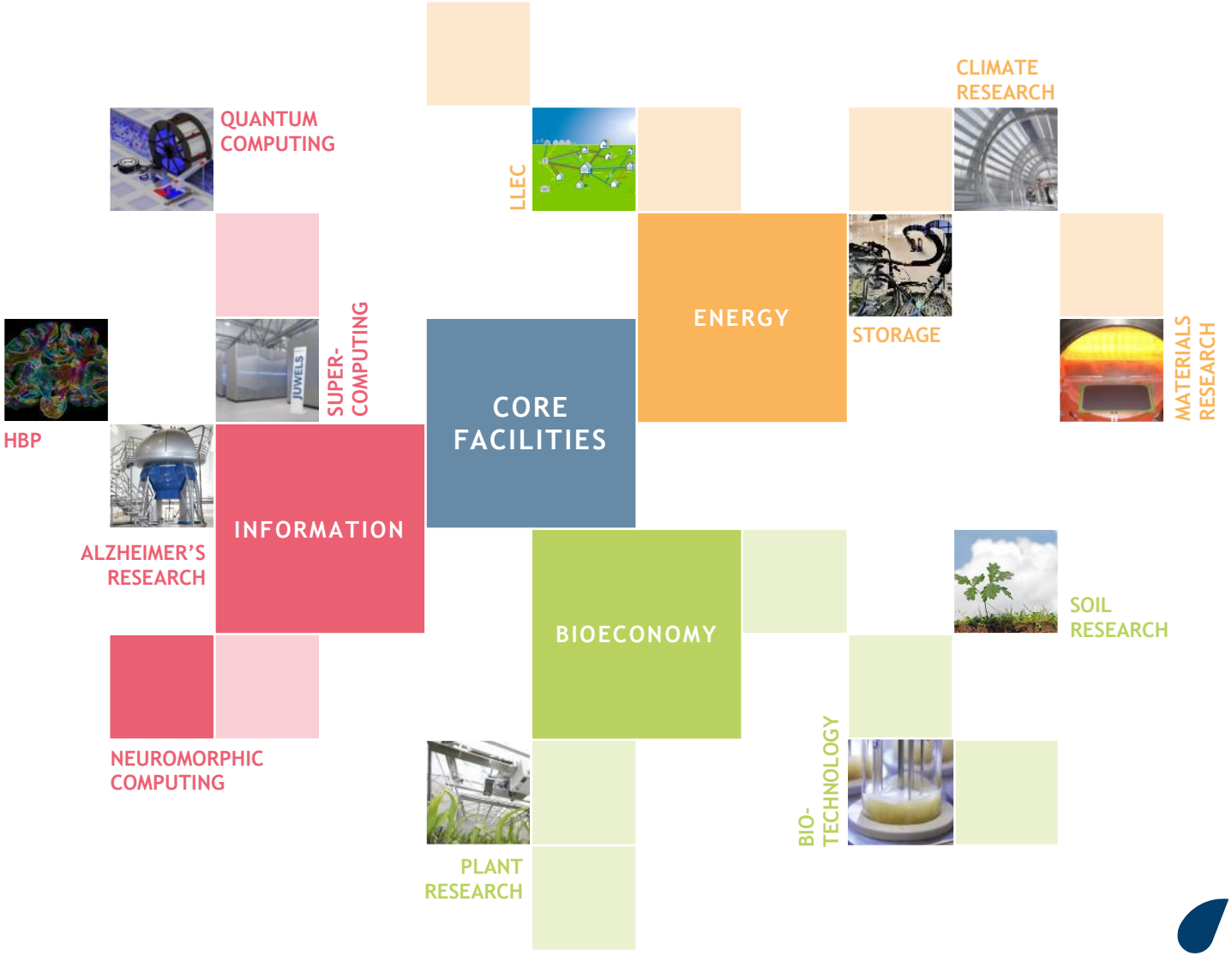
323 trainees and students on placement

867

VISITING SCIENTISTS

from **65** countries

STRATEGIC PRIORITIES

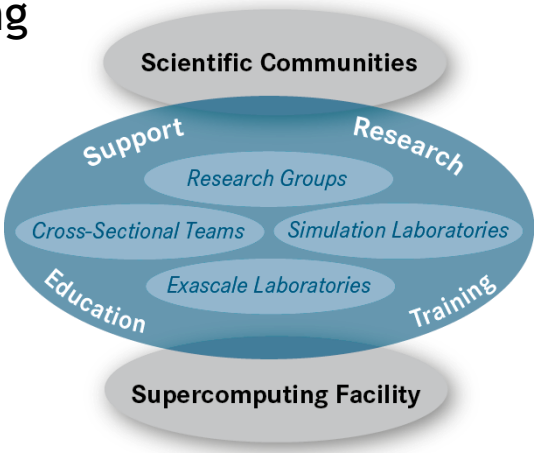


JÜLICH Forschungszentrum

JÜLICH SUPERCOMPUTING CENTRE

JÜLICH SUPERCOMPUTING CENTRE

- **Supercomputer operation for:**
  - Center – FZJ
  - Region – RWTH Aachen University
  - Germany – Gauss Centre for Supercomputing
    John von Neumann Institute for Computing
  - Europe – PRACE, EU projects
- **Application support**
  - Unique support & research environment at JSC
  - Peer review support and coordination
- **R-&-D work**
  - Methods and algorithms, computational science, performance analysis and tools
  - Scientific Big Data Analytics
  - Computer architectures, Co-Design
    Exascale Laboratories: EIC, ECL, NVIDIA
- **Education and Training**



DEEP

JUELICH STORAGE

JUWELS
2600+ Nodes

EUDAT

DEEP

JUST

JUROPA3

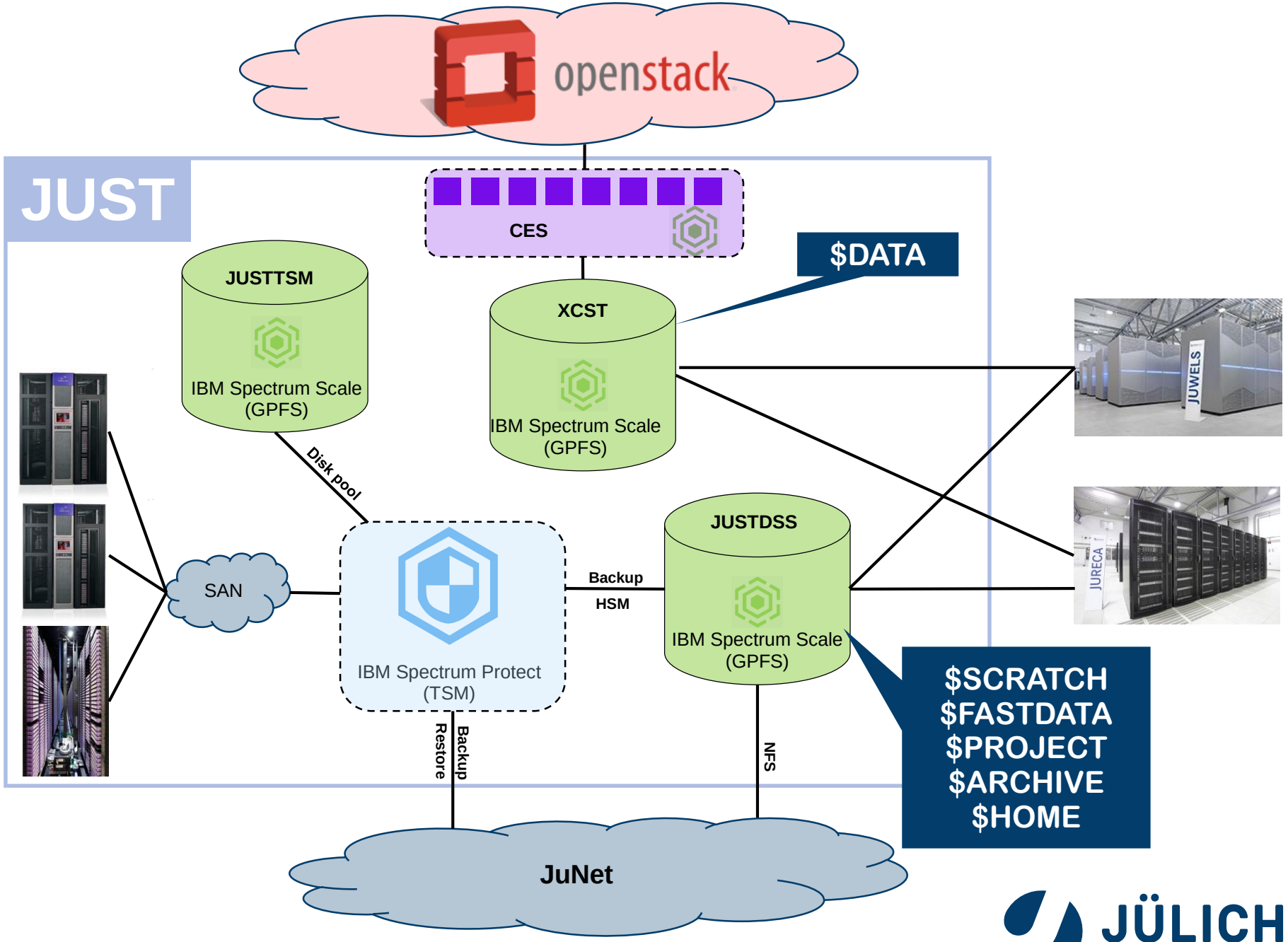
JURECA + JURECA Booster
3600+ Nodes

JUDAC

JuAMS

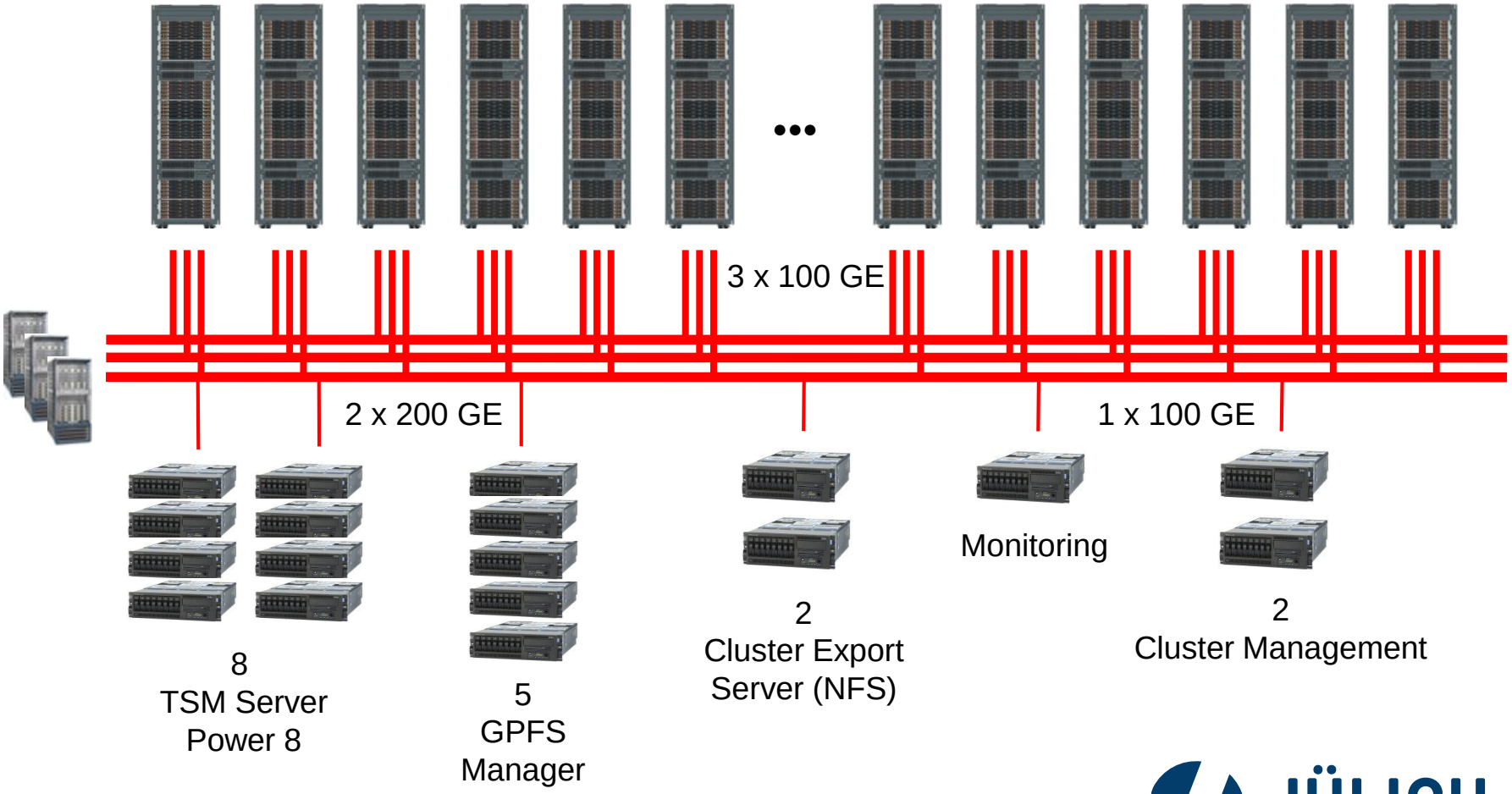
JUROPA3-ZEA

JÜLICH Forschungszentrum


openstack
JUST
CES
$DATA
JUSTTSM
IBM Spectrum Scale
(GPFS)
XCST
IBM Spectrum Scale
(GPFS)
JUWELS
Disk pool
SAN
IBM Spectrum Protect
(TSM)
Backup
HSM
JUSTDSS
IBM Spectrum Scale
(GPFS)
JURECA
$SCRATCH
$FASTDATA
$PROJECT
$ARCHIVE
$HOME
Backup
Restore
NFS
JuNet

JUST – 5TH GENERATION

21 x DSS240 + 1 x DSS260 → 44 x NSD Server, 90 x Enclosure → +7.500 10TB disks



JUST – 5TH GENERATION

21 x DSS240 + 1 x DSS260 → 44 x NSD Server, 90 x Enclosure → +7.500 10TB disks

Characteristics:

- Spectrum Scale (GPFS 5.0.1) + GNR (GPFS Native RAID)
  - Declustered RAID technology
  - End-to-End data integrity
- Spectrum Protect (TSM) for Backup & HSM
- Hardware:
  - x86 based server + RHEL 7
  - IBM Power 8 + AIX 7.2
  - 100GE network fabric
- 75 PB gross capacity
- Bandwidth: **400 GB/s**

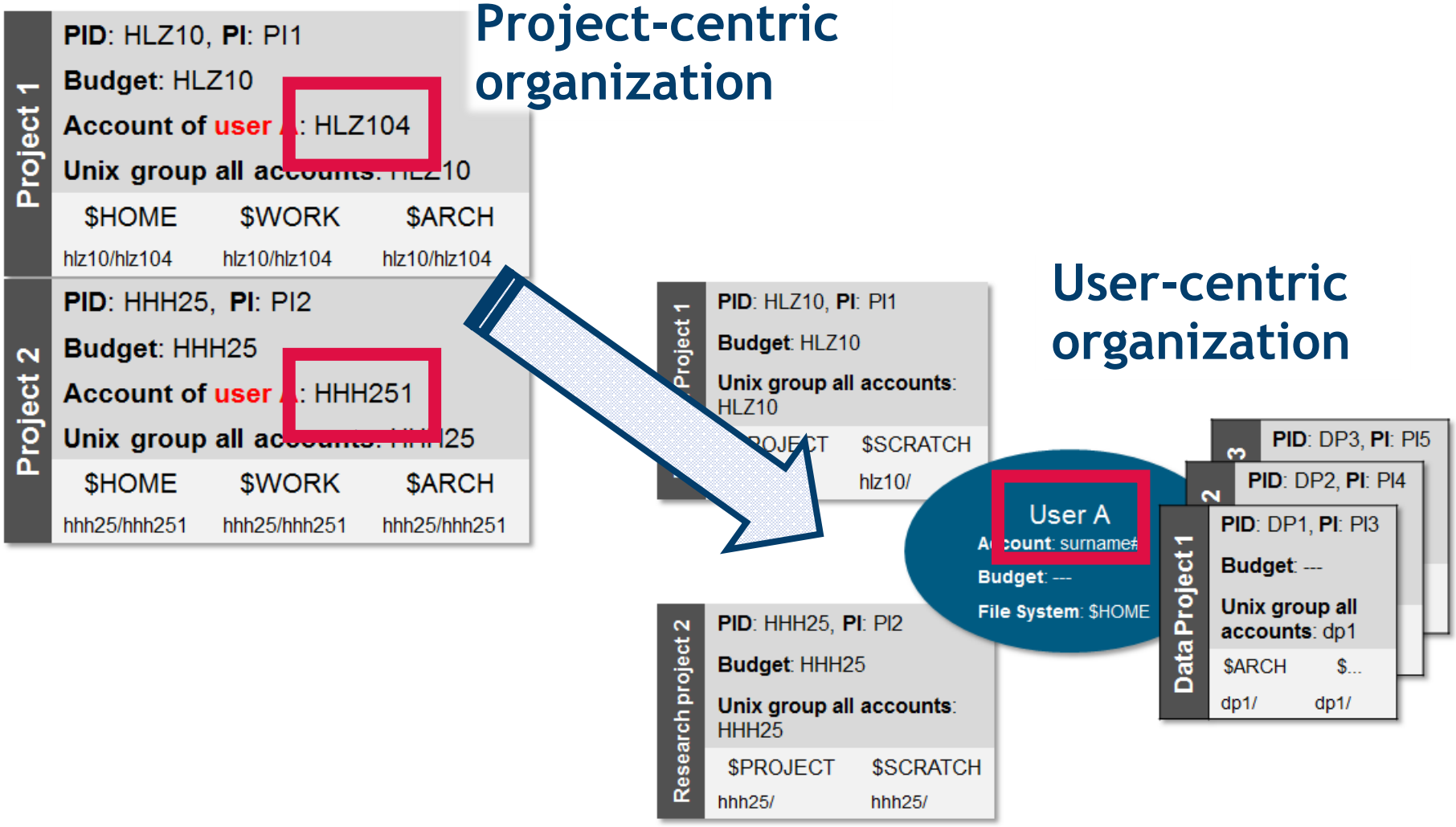
8
TSM Server
Power 8

5
GPFS
Manager

Cluster Export
Server (NFS)

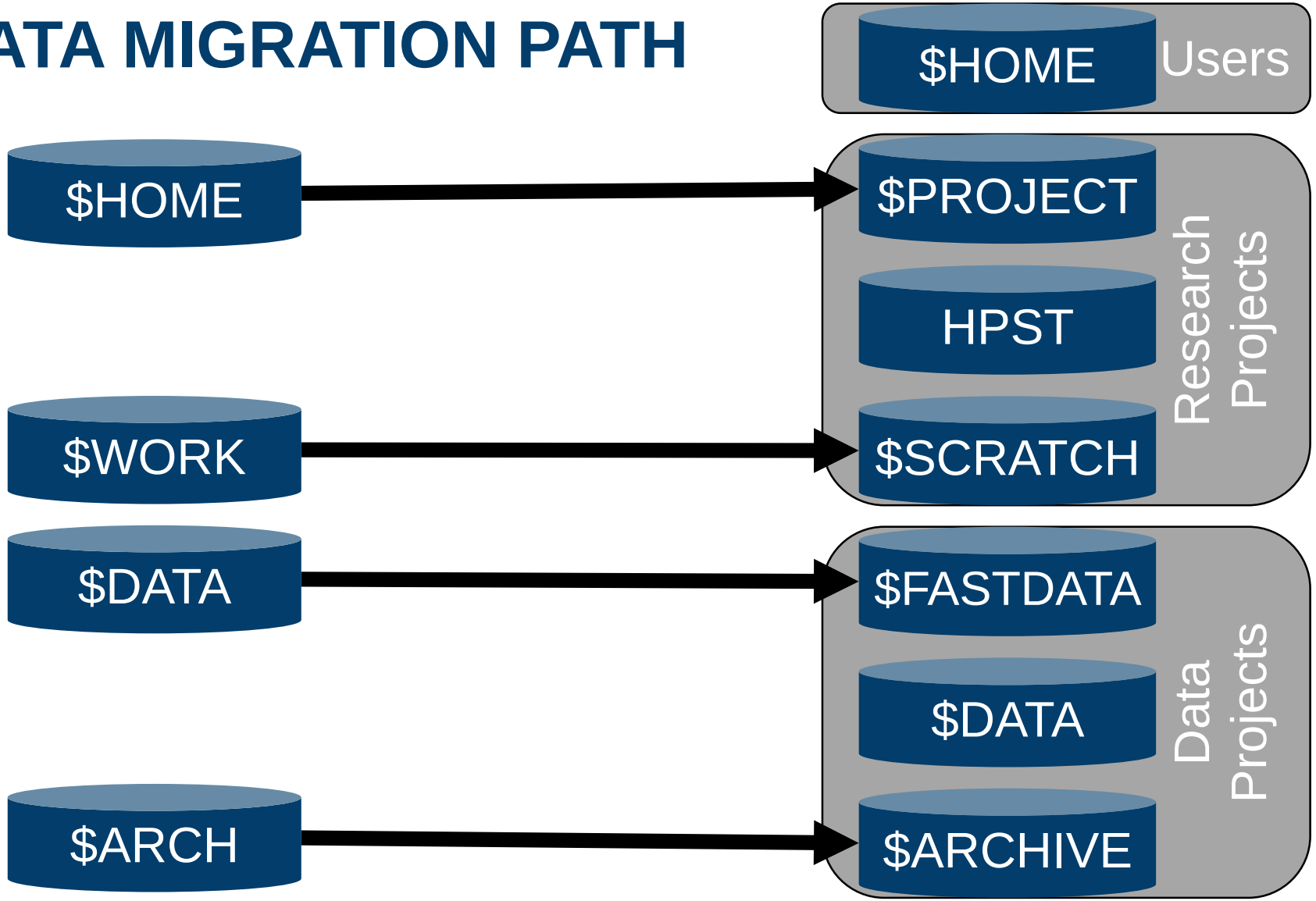
2
Cluster Management

“USAGE MODEL @ JSC” SEIT NOV 2018



JÜLICH Forschungszentrum

DATA MIGRATION PATH



DATA MIGRATION – CONDITIONS

- User mapping **n:1**
- **/arch[2]** stay as it is, only userid change required
  - 31 PB migrated data
- New file systems (new features)
- Project quota based on GPFS independent filesets
- To migrate:

File system	Capacity Usage	Inode Usage
/work	~ 3.9 PB	~ 180.000.000
/home[abc]	~ 1.6 PB	~ 380.000.000
/data	~ 4.8 PB	~ 43.000.000
∑	**> 10 PB**	**> 600.000.000**

- Double of capacity needed: JUST^{5th} comes into play

DATA MIGRATION – CONDITIONS

- User mapping **n:1**
- **/arch[2]** stay as it is, only userid change required
  - 31 PB migrated data

Filesystem creation:

```
mmcrfs project -F project_disks.stanza -A No -B 16M
-D nfs4 -E no -i 4K -m 2 -M 3 -n 16384 -Q yes -r 1 -R 3
-S relatime -T /p/project --filesetdf --inode-limit 1000M
--perfileset-quota
```

∑	> 10 PB	> 600.000.000

- Double of capacity needed: JUST[5th] comes into play

DATA MIGRATION – TOOL EVALUATION

1. approach: GPFS policy engine + rsync

   Pro: rsync is designed to do this job + UID/GID mapping possible

   Con: does not scale up

   → always stats files from file list

2. approach: GPFS policy engine + delete + copy + change ownership

   Pro: scales up much better than rsync

   Con: self implemented→ more effort

DATA MIGRATION – A HARD ROAD

- Projects: Directory quota, realized with GPFS independent filesets
  - Fileset creation time to long (0.5 - 24 hours)
    ~900 projects → Severity 1 case + complain @ IBM
    partial fix available in November
- Fancy file names
  - Control characters, UTF8, Other coding?
    → hard to handle in scripts
- Tests must run on real data → long test cyclus

DATA MIGRATION – A HARD ROAD

Ez_z_subgrid__overlay_000000.h5_$x_{lim} = 8, dx = 15.6e\,-\,3$_$x_{lim} = 8, dx = 31.2e\,-\,3 $_comp.pdf

- P ctory quota, realized with GPFS independent filesets
  - tion time to long (0.5 - 24 hours)
  - ts → Severity 1 case + complain @
  - vailable in November
- F mes
  - racters, UTF8, Other coding?
  - andle in scrip
- T n on real data → long test cyclus

0|\316
0|^_^B
0,]
0,\355
0,^D^A
0\254,^B
0\374?^A
0\374\301
0\374\253^A
0\234\240^A
0\254\370^A
0\354\214^A
0\354
^B
0\234^O^B
0\354^]^B
0^L,^A
0^L;^B
0^L\366
0^L\324^A
0^\@^B
0^\\
0l\375
0^\w
0\234X

Â°\Ã¯!Ã¼^?

bqcd-$\(jobid\).out

./ââ â«/â esâ .txt

H=-t\sum_{i,j\sig.pdf

\

黑河流域土壤水分降尺度产品算法流程.docx

Ð½ÐµÐ¿ÑÐ¸Ð»Ð¸ÑÐ½Ð¾Ðµ ÑÐ»Ð¾Ð²Ð¾

extract_bjÃ¶rn.awk

JÜLICH Forschungszentrum

DATA MIGRATION – A HARD ROAD

- Projects: Directory quota, realized with GPFS independent filesets
  - Fileset creation time to long (0.5 - 24 hours)
    ~900 projects → Severity 1 case + complain @ IBM
    partial fix available in November
- Fancy file names
  - Control characters, UTF8, Other coding?
    → hard to handle in scripts
- Tests must run on real data → long test cyclus

DATA MIGRATION – FINAL SYNC

Time line in offline maintenance 30th November – 4th December

Phase 1: Delete (project)

- 5 nodes in JUST
- 1 h Policy run per file system (project + home[abc])
- 1 h compare list + 20 minute delete files

Phase 2: Copy

- 128 nodes on JURECA (each 5 **cp** at same time)
  - 25 h for group **zam** (**homeb**) → **cjsc**
- **/data** finished Saturday morning, **/work** @ midday, **/home[abc]** @ evening

Phase 3: Change-owner

- 5 nodes in JUST
- Policy run + **chown** command: 2 h for $PROJECT

Create new $HOME in parallel: 12 h

DATA MIGRATION – FINAL SYNC

Time line in offline maintenance 30th November – 4th December

Phase 1: Delete (project)

- 5 nodes in JUST
- 1 h Policy run per file system (project + home[abc])
- 1 h compare list + 20 minute delete files

Phase 2: Copy

- 128 nodes on JURECA (each 5 **cp** at same time)
  - 25 h for group **zam** (**homeb**) → **cjsc**
- **/data** finished Saturday morning, **/work** @ midday, **/home[abc]** @ evening

Phase 3: Change-owner

- 5 nodes in JUST
- Policy run + **chown** command: 2 h for $PROJECT

Create new $HOME in parallel: 12 h

DATA MIGRATION – FINAL SYNC

Time line in offline maintenance 30th November – 4th December

Phase 1: Delete (project)

- 5 nodes in JUST
- 1 h Policy run per file system (project + home[abc])
- 1 h compare list + 20 minute delete files

Phase 2: Copy

- 128 nodes on JURECA (each 5 **cp** at same time)
  - 25 h for group **zam** (**homeb**) → **cjsc**
- **/data** finished Saturday morning, **/work** @ midday, **/home[abc]** @ evening

Phase 3: Change-owner

- 5 nodes in JUST
- Policy run + **chown** command: 2 h for $PROJECT

Create new $HOME in parallel: 12 h

DATA MIGRATION – FINAL SYNC

Time line in offline maintenance 30th November – 4th December

Phase 1: Delete (project)

- 5 nodes in JUST
- 1 h Policy run per file system (project + home[abc])
- 1 h compare list + 20 minute delete files

Phase 2: Copy

- 128 nodes on JURECA (each 5 **cp** at same time)
  - 25 h for group **zam** (**homeb**) → **cjsc**
- **/data** finished Saturday morning, **/work** @ midday, **/home[abc]** @ evening

Phase 3: Change-owner

- 5 nodes in JUST
- Policy run + **chown** command: 2 h for $PROJECT

Create new $HOME in parallel: 12 h

OPEN PMRS

- “mmchmgr” takes 16+ hours
  - “mmcheckquota” takes 16+ hours
  - Most probably “mmfsck” takes also a very long time
- “ls /p/project” sometimes takes more then 20 seconds
- Parallel directory creation from 800 compute nodes into one directory stucks for 12+ minutes
- “dd” into a newly created file gets stuck

THANK YOU