The PUNCH4NFDI Consortium

Particles, Universe, NuClei and Hadrons for the NFDI

Baida Achkar

II. Institute of Physics - Georg-August-Universität Göttingen

TA2-WP2 Meeting, 13.05.2024

GEORG-AUGUST-UNIVERSITÄT GÖTTINGEN IN PUBLICA COMMODA SEIT 1737

CONTENT

- User Case and related TAx-WPx.
- Göttingen entries for the Q1-2024 report.
- (WIP) Integration into PUNCH4NFDI infrastructure.

PUNCH4NFDI-TA2-WP2 and TA6-WP5@GAU

- **Involved Deliverables:**
  - D-TA2-WP2-4: Integration of a variety of compute resources available in PUNCH4NFDI into Compute4PUNCH.
  - D-TA6-WP5-4:
    - Interfaces to the “HLRN – High Performance Computing in Northern Germany” in Göttingen will be developed to the PUNCH4NFDI communities.
    - A fraction of the GoeGrid grid computing cluster in Göttingen will be provided to PUNCH4NFDI and beyond for analysis of the CERN open data to *users without explicit CERN or experiment affiliation.*

1st 2024 Quarterly PUNCH report / Göttingen entries – TA2-WP2 and TA6-WP5 / 1 January- 31 March 2024

- **Involved Deliverable D-TA2-WP2-4 and D-TA6-WP5-4**
- **Recap on the local PUNCH4NFDI GoeGrid Pool (PGPool) infrastructure: HTCondor Overlay Batch System OBS using exclusively GoeGrid - Göttingen cluster resources is deployed and comprises:**
    1. HTCondor OBS (1 collector/negotiator, 1 submitter and 20x 8C-Working Nodes (20x 8C WNs allowed to be claimed /busy at the same time) equipped with Ganglia web-based monitoring,
    2. COBald-TARDIS resources manager, starts PGPool HTCondor working node drones that integrate themselves into the OBS, /cvmfs namepace, natively accessible on all GoeGrid computing nodes, is bind-mounted to the PGPool working nodes.
- **Work In Progress: Fulfilling requisites for integration of PGPool into Compute4PUNCH (C4P) infrastructure**
- **Note: Subsequent developments have been tested locally using the local PGPool infrastructure and need to be retested after joining the C4P HTCondor OBS.**
    1. The PGPool working node container updated to Rocky-Linux8 and htcondor-release-23.x-1.el8.noarch.rpm.
    2. Finalization of the PGPool WN configuration (a dynamical configuration from a Git repository using https://pypi.org/project/condor_git_config/ hook is enabled):
        I. Condor Connecting Brokering (CCB) for out bounding of PGPool WNs is set and need to be tested after joining the C4P Pool ,
        II. In order to reinforce isolation during execution of the users jobs, submitted jobs are restricted to run inside one of the pre-built Compute4PUNCH/container-stacks containers bind-mounted into the PGPool working nodes,
        III. Compute4PUNCH/container-stack job containers are read from /cvmfs/unpacked.cern.ch to enable efficient distribution and scalability using local caching. Enabling /cvmfs access in PGPool working nodes eliminates the need to enable user-/mount feature which enhances security and reduces the risk of unwanted /suspicious uses.
    3. The PGPool configuration folder is migrated to the C4P official Git Repository. compute4punch/c4p-htcondor-configs/GoeGrid-cloud
    4. Related presentations: Contribution to the 3rd-TA2-workshop@desy

Integration of the PGPool infrastructure into Compute4PUNCH

1. compute4punch/container-stacks/htcondor-wn is used to set-up the PGPool working node.
2. Configuration of the HTCondor Worker Node using ansible. HTCondor itself is configured using https://pypi.org/project/condor_git_config/ hook (dynamical configuration from a git repository).
3. Configuration folder for GoeGrid-cloud is merged to compute4punch/c4p-htcondor-configs GitLab repository .
4. /cvmfs is bind mounted from the GoeGrid WN host to a directory within the container —> No need to enable user-/mount namespaces.
5. Users jobs are restricted to run inside one of the pre-built /compute4punch/container-stacks images.
6. Pilot job that start the working node is launched by COBald-TARDIS manager using:

```
singularity run --userns -B $PWD/config:/srv/config -B $PWD:/scratch –B /cvmfs:/cvmfs
/cvmfs/unpacked.cern.ch/registry.hub.docker.com/matterminers/htcondor-wn\:latest
```

7. Next steps:
   - ✓ HTCondor IDTOKEN to join the Compute4PUNCH pool, and then update the secrets.yaml file.
   - ✓ Use of Condor Connection Brokering CCB on the working node for out bounding: CCB (CCB_ADDRESS = $(COLLECTOR_HOST)). To be tested after joining the pool.
   - ✓ The local COBald-TARDIS manager configuration file adjusted for communication with the Compute4PUNCH central manager. To be tested as well.
   - ✓ Use an access token, with a group scope. The same group will be defined for the resources we are offering. The access token defines which resources users are allowed to access/use. To follow with Harry, Manuel and Benoit.

Acknowledgements

This work was [in part] supported by DFG fund „NFDI 39/1“ for the PUNCH4NFDI consortium.

PUNCH 4NFDI

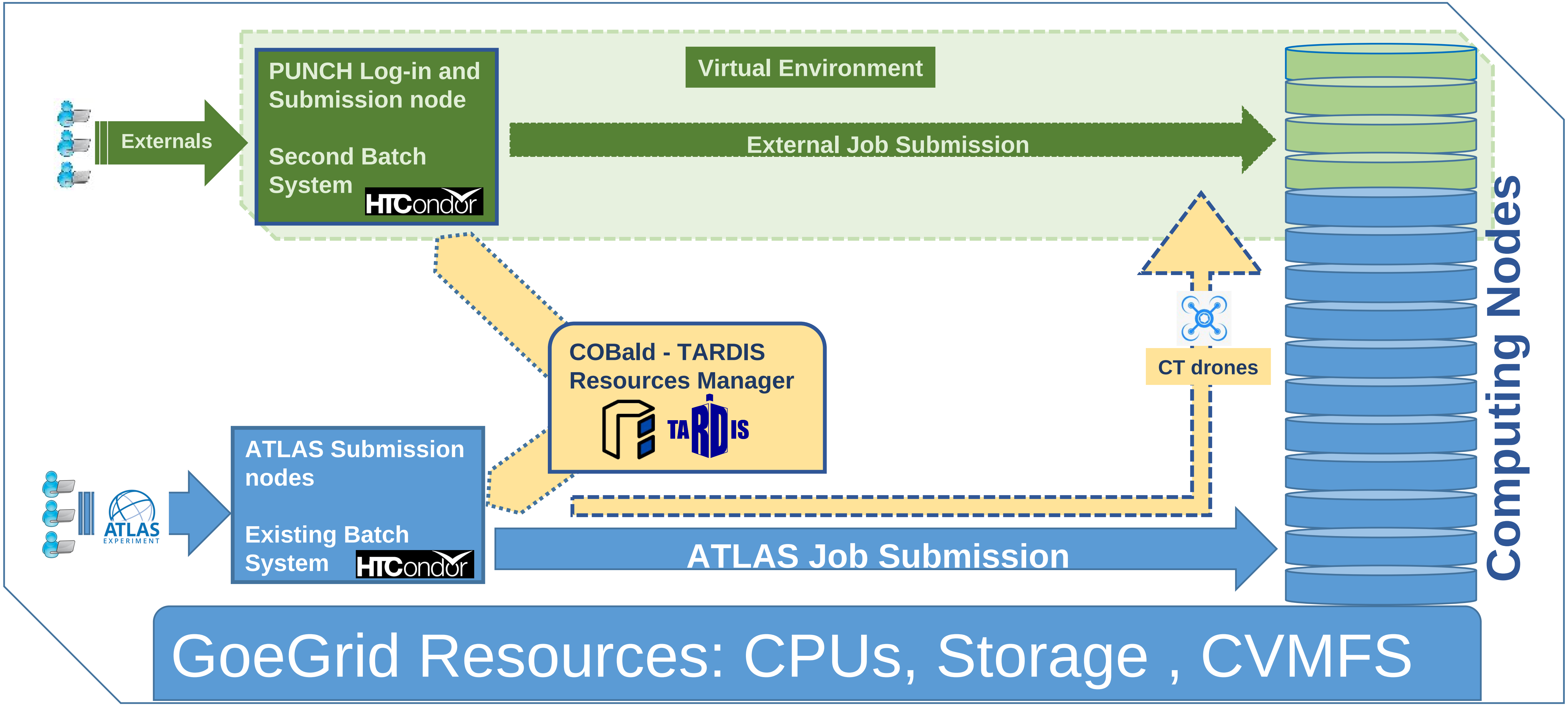
Funded by

DFG Deutsche Forschungsgemeinschaft

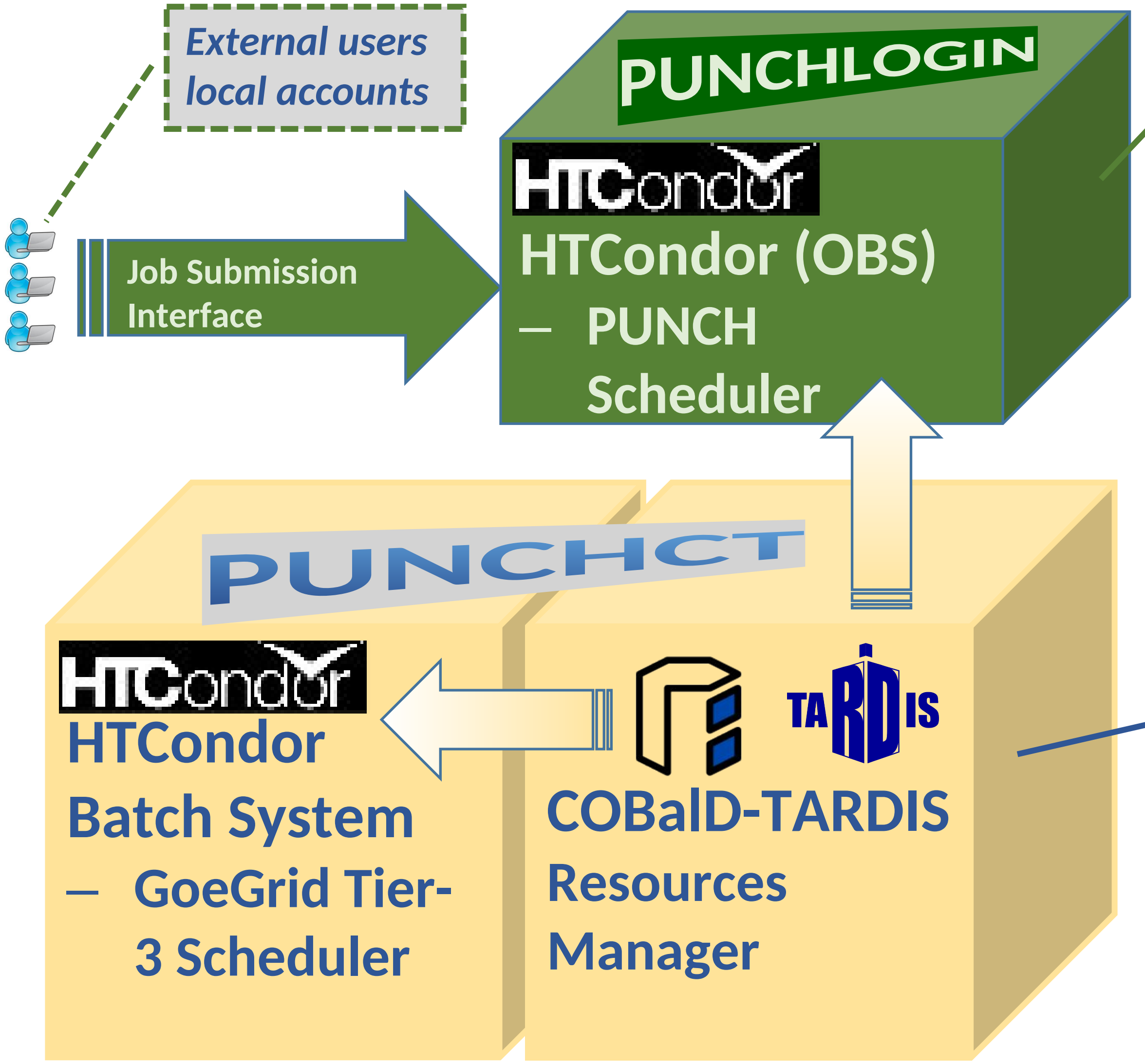
German Research Foundation

Backup

PUNCH4NFDI@GAU / PUNCH GoeGrid Pool PGPool Setup



PUNCH4NFDI@GAU / PGPool Infrastucture



HTCondor Overlay Batch System OBS:
MASTER, COLLECTOR, NEGOTIATOR,
SCHEDD & GANGLIAD

- ✓ A COBalD-TARDIS instance runs as a system service: systemctl start cobald@cobald_obs_site
- ✓ Configured with:
  - ✓ HTCondor Batch System Adapter,
  - ✓ Options: pool: punchlogin.local
  - ✓ HTCondor Site Adapter:

    quota: 160
    drone_heartbeat_interval: 60
    drone_minimum_lifetime: 3600
- ✓ TARDIS starts drones (via Vanilla pilot.jdl jobs), which then are integrated themselves into the OBS using the HTCondor Batch System Adapter.
- ✓ Pilot jobs run under the accounting group Group_ATLAS.punchct on GoeGrid WNs

20 drones x 8 CPU hou
~ 1% of GoeGrid
capacity (Tier-3)

TA2-WP2 Delivrables

- Deliverables:
  - D-TA2-WP2-1 (30 Jun 2022): Demonstrator for federated compute infrastructure Compute4PUNCH.
  - D-TA2-WP2-2 (30 Jun 2024): Adaption of Compute4PUNCH for domain specific large data collection (LOFAR, MeerKAT, CERN open data).
  - D-TA2-WP2-3 (30 Jun 2022): Prototype for container registry.
  - D-TA2-WP2-4 (31 Dec 2024): Integration of a variety of compute resources available in PUNCH4NFDI into Compute4PUNCH.
  - D-TA2-WP2-5 (30 Jun 2023): Realisation of entry points as JupyterHub and batch system.
  - D-TA2-WP2-6 (30 Jun 2024): Integration of opportunistic cache systems into Compute4PUNCH, and testing.
  - D-TA2-WP2-7 (30 Sep 2025): Data-locality aware scheduling available in the overlay batch system.
  - D-TA2-WP2-8 (30 Sep 2026): Fully capable container registry.