Destination Earth

Destination Earth: Challenges in Federation of Compute and Data Resources

Utz-Uwe Haus, Craig Prunty, Hans-Christian Hoppe

10/10/2023

Funded by the European Union

ECMWF

ETP 4 HPC

Context/Disclaimer

- This presentation is based on work identifying challenges with respect to federation of EuroHPC infrastructures provided for running DestinE twins.
- This is not a summary of a report commissioned by ECMWF in the DE-380 contract.
- We believe the points discussed here are more widely applicable than DestinE, but DestinE is the first project highlighting them.

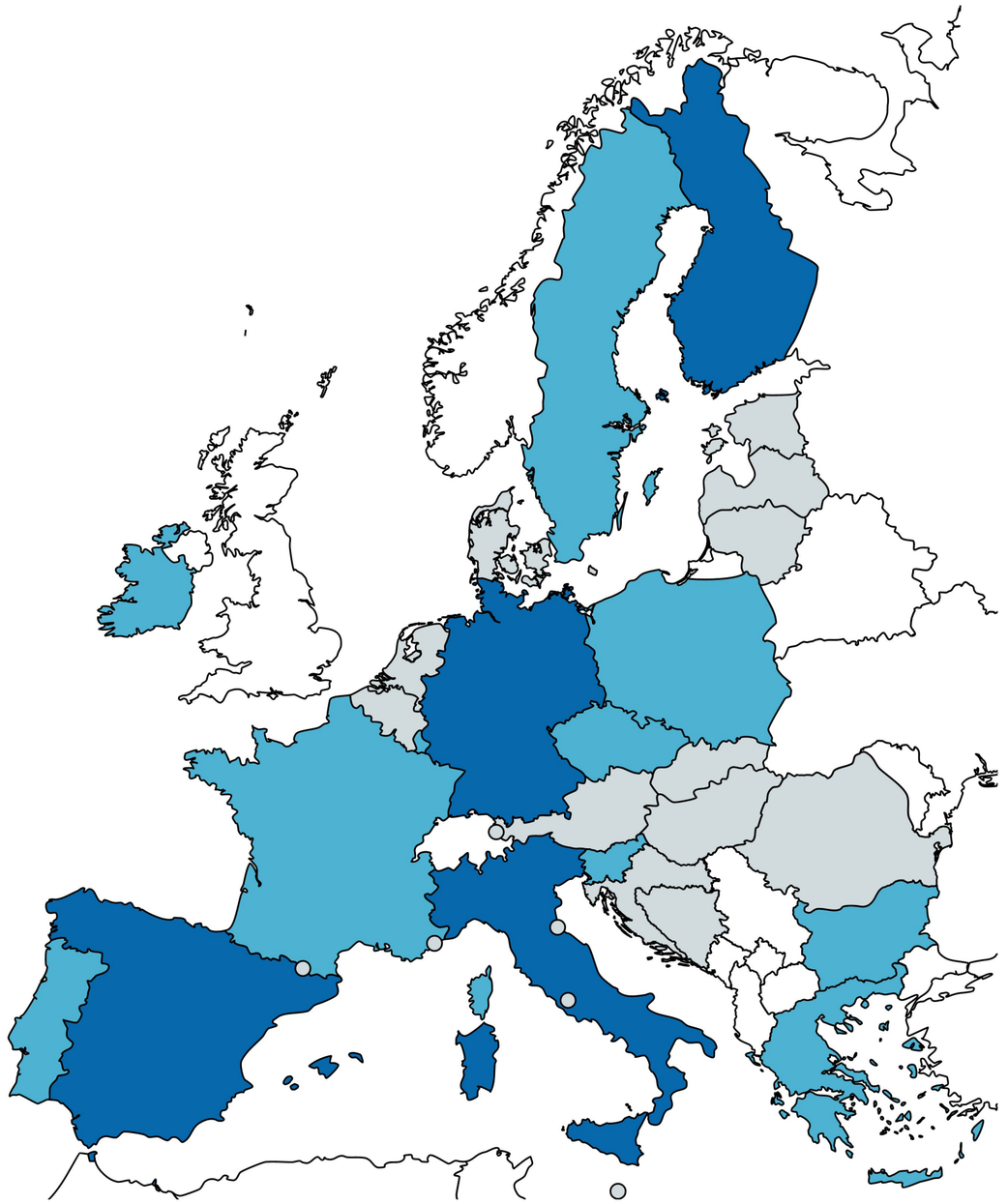
Funded by
the European Union

DESTINE Target Federation Locations:

- CSC, FI (Lumi)
- Cineca, IT (Leonardo)
- BSC, ES (Marenostrum5)
- Jülich, DE (Jupiter)

Potential Future Federation Locations

- Genci, FR (Jules Verne consortium)
- LuxConnect, LU (Meluxina)
- IT4Innovations, CZ (Karolina)
- MACC, PT (Deucalion)
- IZUM, SI (Vega)
- Sofia, BG (Discoverer)
- GRNet, GR (Daedalus)
- Cyfronet, PL (EHPCPL)
- ICHEC, IE (CASPIr)
- Linköping, SE (Arrhenius)



DestinE Target Federation locations
Potential future federation locations

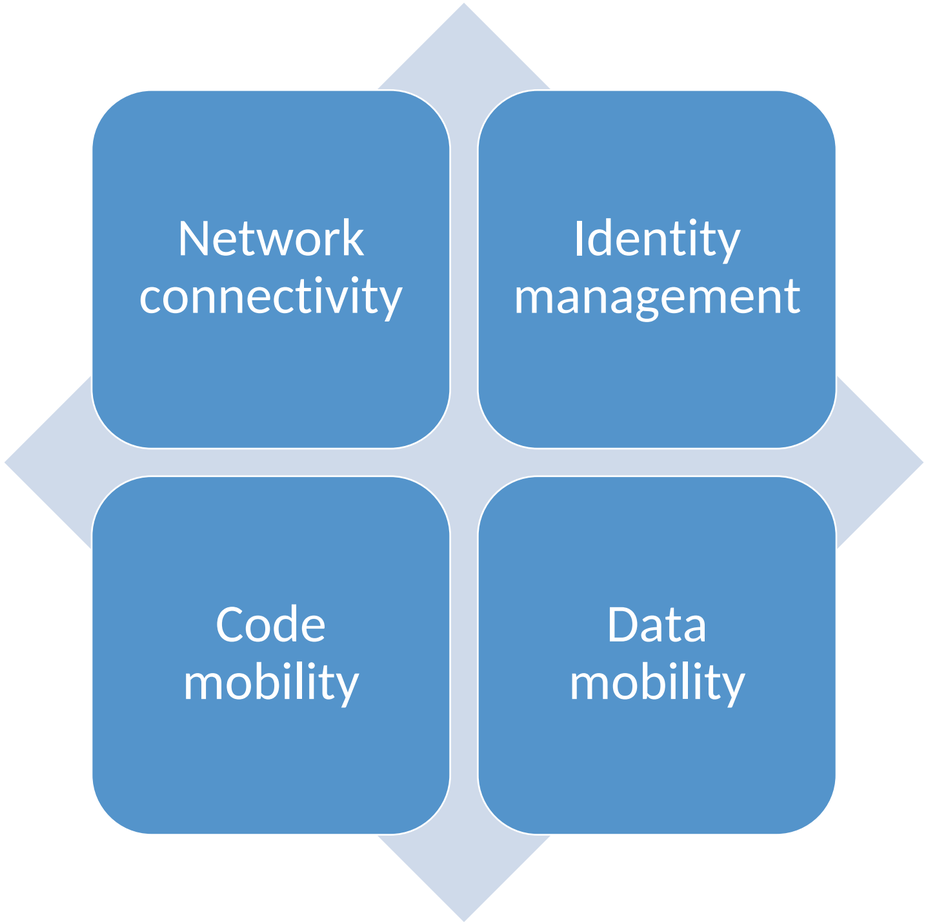
Funded by
the European Union

ECMWF

ETP 4 HPC

Fedederation

"The act of creating or becoming a union of organizations" (Merriam-Webster)

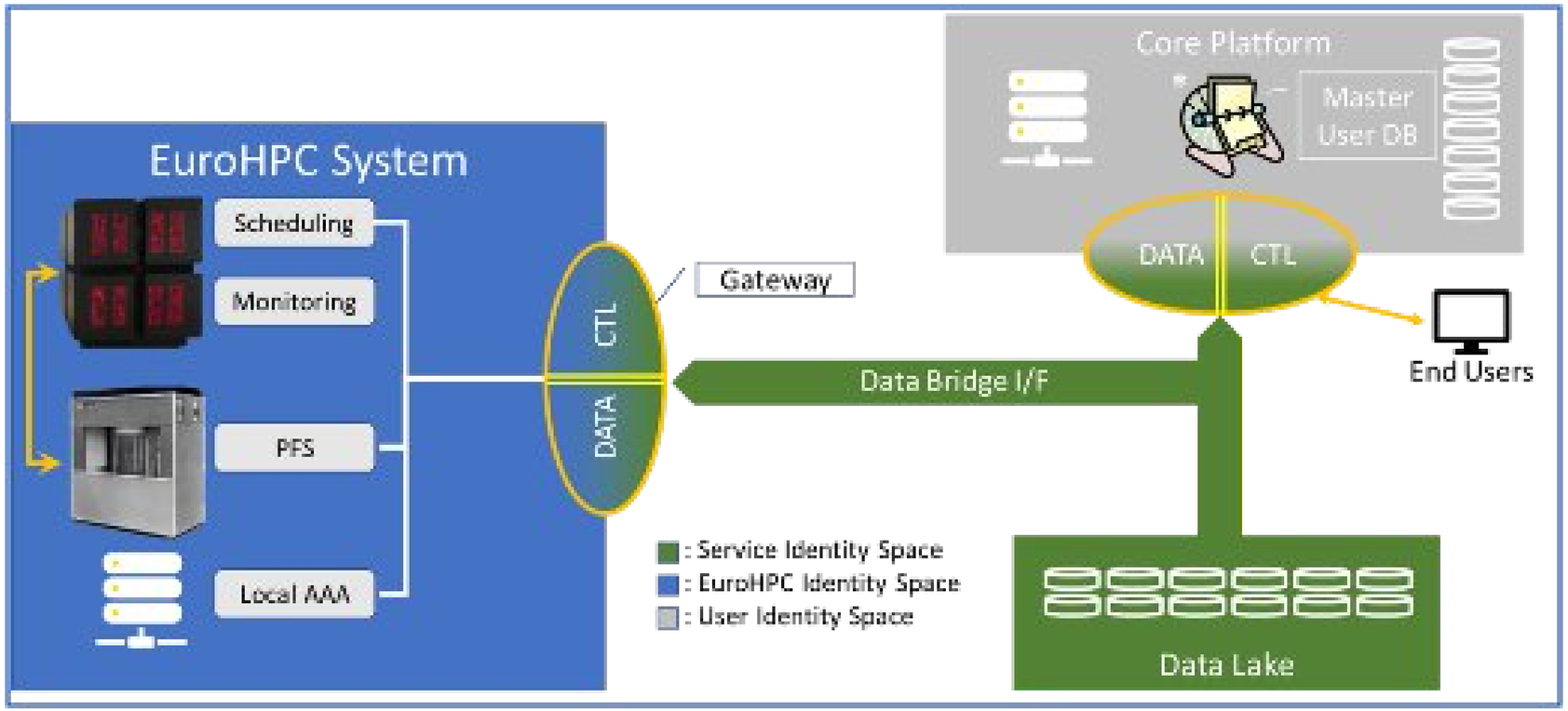


Funded by the European Union

Conceptual federation architecture

Assumptions

- Users only interface with Service platform
- Invocation of DT services automatic, using service identities
- HPC platforms = EuroHPC sites
- Data lake with common metadata schema

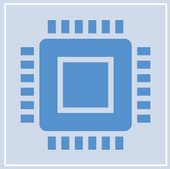


Funded by the European Union

Federation Challenges

User Management

User Policy
Authentication, Authorization & Accounting
Data Governance



Service Deployment /
Operation, Orchestration, Scheduling

Service interfaces and Instantiation
Local Service Scheduling and Orchestration
Distributed Service Scheduling
Service Deployment and Updates
Quality of Service (24h/7?)

System Integration

Independent systems, partially isolated,
Different architectures
Mixed-use with other tenants

Funded by
the European Union

User management

User Management

User Policy

Authentication, Authorization & Accounting

Data Governance

Authentication: validating identity
Authorization: requested operation or resource is granted
Accounting: amount of compute resources used or data accessed vs quotas or billing
QoS: description or measurement of performance of a service toward the user

User Policy:

Federation will introduce central authentication of users:

- nested user registration/authentication
- Definition and obsolescence of roles will increase in type and number.
- Proprietary (or at least restricted access) data will impose data governance requirements.

Complex due to heterogeneity of the various hosting sites compounded by introduction of the federated service infrastructure

Authentication, Authorization & Accounting (AAA):

AAA is performed locally, typically with different solutions for different centers. Ideally the centers align their local solutions and support the Federated users, likely as nested solutions. Either Federated solutions need to comply with local authentication processes or use authentication methods guaranteeing at least the same level of security.

- Authorization is performed by the HPC centres using their local data access control methods (based on user or access group rights or “project ID”
- Accounting also relies on user accounts or project IDs —federated systems must account across systems and will require fairness in allocating resources while managing user preferences and/or code limitations
- Accounting resolution may require higher precision to support nested allocation.
- Federation can introduce non-open data

Quality of Service (QoS)

Operational weather forecasts require solutions in prescribed times. Regular HPC centre scheduling and resource management typically maximizes total throughput, not completion times.

- Monitoring, prioritizing, and policing will need to be implemented , potentially across centers, routing urgent service requests to participating HPC centres with free capacity or desired compute. Existing jobs must either be migrated, downsized, or killed.
- Nested QoS required.

Funded by the European Union

Service Deployment / Operation

Migrating traditional, session-based HPC systems to target a service model

Service Deployment / Operation, Orchestration, Scheduling

- Service interfaces and Instantiation
- Local Service Scheduling and Orchestration
- Distributed Service Scheduling
- Service Deployment and Updates
- Quality of Service (24h/7?)

Service interfaces and Instantiation

- Will participating centres expose service interfaces directly to federated end-users? Or support remote execution of (shell) commands or jobs?
- Can the same user-side interface be imposed across centers? This could be achieved in different ways
  - HPC centres translate UI data internally to commands for the local systems
  - Federated SW translates commands to interface to the HPC centres

Local Service Scheduling and Orchestration

- Running Destination Earth DTs would require many nodes. Schedulers do not efficiently manage data movement across multiple nodes. To support SLAs for priority workloads, this imposes the need for either special mechanisms (data movement targeted networks, …) or restriction of resources per job.
- Prioritization implies pre-emption of running jobs, imposing the need for graceful degradation/migration and management of state information.
- Quotas (average use over time) can be scheduled but critical DT requests (like to aid disaster recovery) could exceed set quota limits, requiring a rebalancing.

Distributed Service Scheduling

- Longer term, DT services provisioned on multiple HPC centres will benefit from a meta scheduling level to route service requests to the most suitable system (per resource requirements and system statuses). A meta-scheduling mechanism needs local status knowledge for each participating centre, including predicted time of execution of any newly invoked service.

Funded by the European Union

Service Deployment / Operation (2)

Migrating traditional, session-based HPC systems to target a service model

Service Deployment / Operation, Orchestration, Scheduling

- Service interfaces and Instantiation
- Local Service Scheduling and Orchestration
- Distributed Service Scheduling
- Service Deployment and Updates
- Quality of Service (24h/7?)

Software Environment Compatibility and Stability

- Code and performance do not uniformly port across different EuroHPC systems
  - Some well used codes are portable but can see variations in execution across systems.
  - Many codes will require porting and optimization (including some DT couplers, data streaming middleware, databases, and deployment tools)
- Software updates will be managed locally but aligned with federated needs. Software containers for the DT components could help, but
  - would require a common view on containerization policy, tools, and security
  - Containers must be adjusted with system changes.
- services might stay within the boundaries of assigned quotas but still impact other applications through shared resources (e.g. bandwidth or latency impacts due to memory or network interface use).
  - At a minimum, the ability to monitor, diagnose and report these issues is required.
  - Longer term, QoS mechanisms could be developed to prioritize and police allocation.
- Monitoring and observability are not accounted for on local systems. This implies undetected observability of Destination Earth jobs by random users.

Service Deployment and Updates

- Local system updates must be coordinated to the federated system, and vice versa. Code changes should be automated where possible, and at a minimum reported to a central instance.
- SW upgrades may trigger a waterfall of service implementation updates, or at least recompilation, at the local and federated level.
- Longer term each centre should plan a continuous integration/continuous deployment (CI/CD) infrastructure to handle rebuilds and validation.
- In the event that only binaries are distributed, source code changes or rebuilds would have to managed centrally.

Funded by the European Union

System Integration

System Integration

Independent systems, partially isolated,
Different architectures
Mixed-use with other tenants

The **systems** executing the DestinE workload(s) **are independent** (with **different architectures**) **are partially isolated**

- will be required to support **mixed-use with local tenants**.
- creates **data pathing, scheduling**, and **workload portability/parametrization issues** in the federated solution
- DestinE Data Bridge design provides a starting point for data pathing between sites
  - but will not address all issues

Storage access and accounting are

- typically handled via the Unix permissions model
- on global shared storage

Storage access **with guaranteed quality of service** (minimum or maximum bandwidth) or extended data governance models would need to be implemented.

Funded by
the European Union

Outlook

- Executing DestinE workloads will be a good challenge for EuroHPC's vision of a federated European HPC environment
- More challenges than outlined here exist:
  - Data movement/Data streaming
  - Data acquisition/preprocessing/injection into twins
  - Security architecture/Data and code governance
  - Math and Algorithms

Funded by
the European Union

Thank you !

Funded by
the European Union