
# HPC Cloud Application Orchestration through Self-Organisation

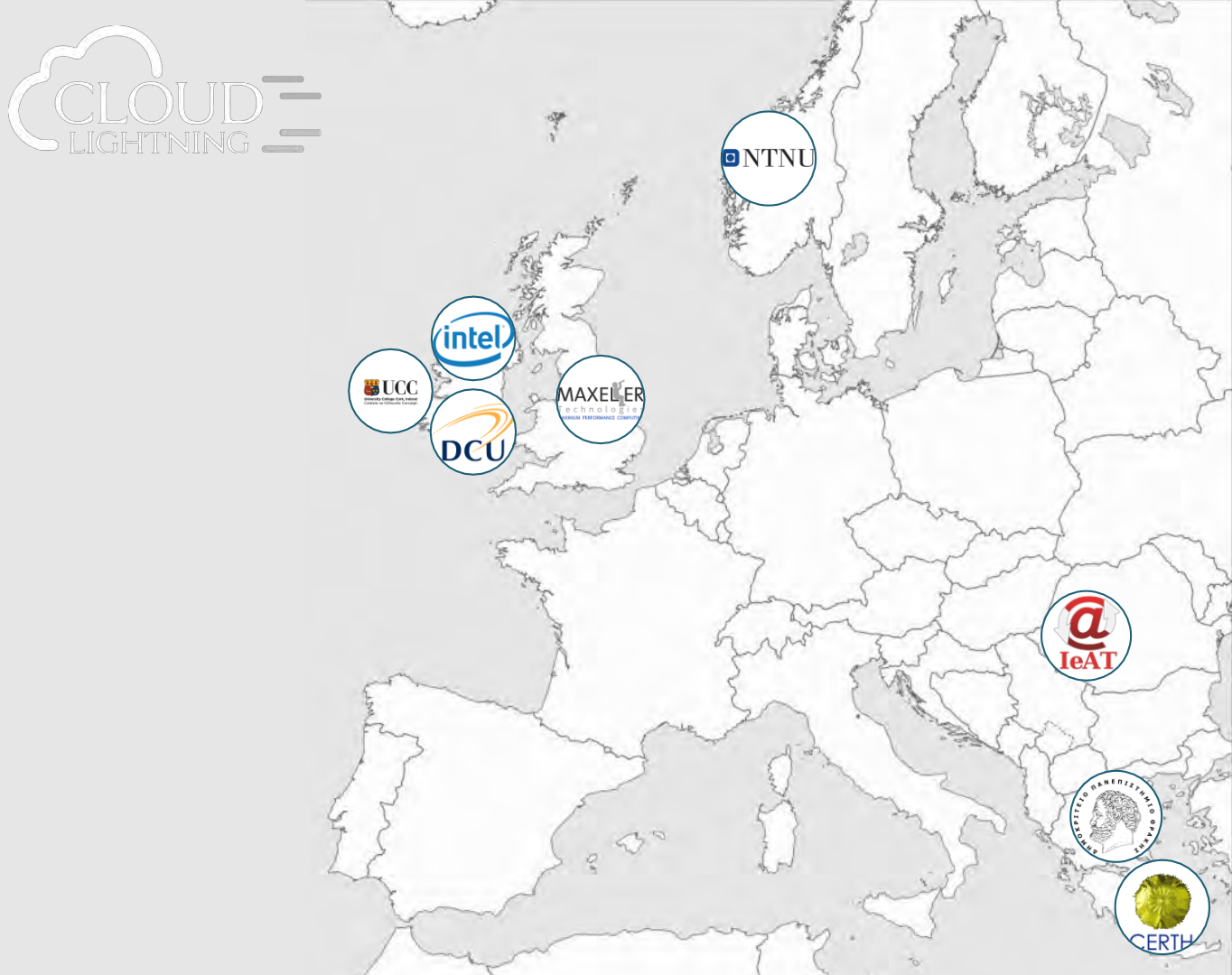
**Marian Neagul, Ioan Dragan, Dana Petcu**
**Institute e-Austria Timisoara, Romania**

# Partners

CloudLightning comprises of eight partners from academia and industry and is coordinated by University College Cork.

Industrial partners:
- Intel Ireland (IE)
- Maxeler (UK)

Academic partners:
- University College Cork (IE)
- Norwegian University of Science and Technology (NO)
- Institute e-Austria Timisoara (RO)
- Democritus University of Thrace (GR)
- The Centre for Research & Technology, Hellas (GR)
- Dublin City University (IE)



# Specific Challenge

**CloudLightning is funded under Call H2020-ICT-2014-1 Advanced Cloud Infrastructures and Services.**

**The aim is to develop infrastructures, methods and tools for high performance, adaptive cloud applications and Services that go beyond the current capabilities.**

- Cloud computing is being transformed by new requirements such as
  - **heterogeneity of resources and devices**
  - software-defined data centres
  - cloud networking, security, and
  - **the rising demands for better quality of user experience.**

- Cloud computing research will be oriented towards
  - **new computational and data management models (at both infrastructure and services levels) that respond to the advent of faster and more efficient machines,**
  - **rising heterogeneity of access modes and devices,**
  - **demand for low energy solutions,**
  - widespread use of big data,
  - federated clouds and
  - secure multi-actor environments including public administrations.

# HPC Challenges

"The challenge is less about educating users about cloud computing and more about the ability of clouds to handle more types of HPC jobs over time."

IDC, 2015

## Traditional High Performance Computing is…

1. Hard to use without deep IT knowledge
2. Expensive
3. Inaccessible to individuals and SMEs
4. Inflexible

**Most HPC workloads are not ready to run on today's cloud architectures.**

CLOUD LIGHTNING

## GPU related systems (NTNU)

**Testbed:**

- Numascale Shared Memory Cluster
  *Each node has 2 AMD Server CPUs + 1 Nvidia GPU*
- Heterogeneous 8-node cluster (CPUs + co-processors)
  *Each node has 1 Intel CPU + 1 Nvidia GPU*

- Numascale cluster system benchmarked with NPB and OSU benchmark suites
- Bandwidth tests executed to study the performance of Nvidia GPUs in each node
- Documented impact of when host process access GPUs located in same node

## DFEs (Maxeler)

**Testbed setup:**

- MAX3 system with 4 DFEs @ NTNU
- MAX4 system with 8 DFEs internally at Maxeler

- DFEs differs from conventional CPUs which are instruction and control-flow oriented:
  - split program into data-plane and control-plane
  - throughput oriented computing, massive parallelism in data-plane
- DFE architecture:
  - reconfigurable chip (FPGA) with configurable memories and DSPs
  - combine with large DDR memory and high-speed IO
  - create specialised computer architecture, optimised for application
- Programming:
  - Dataflow language based on Java syntax
  - inherently parallel

## MICs / Intel Xeon Phi (Intel)

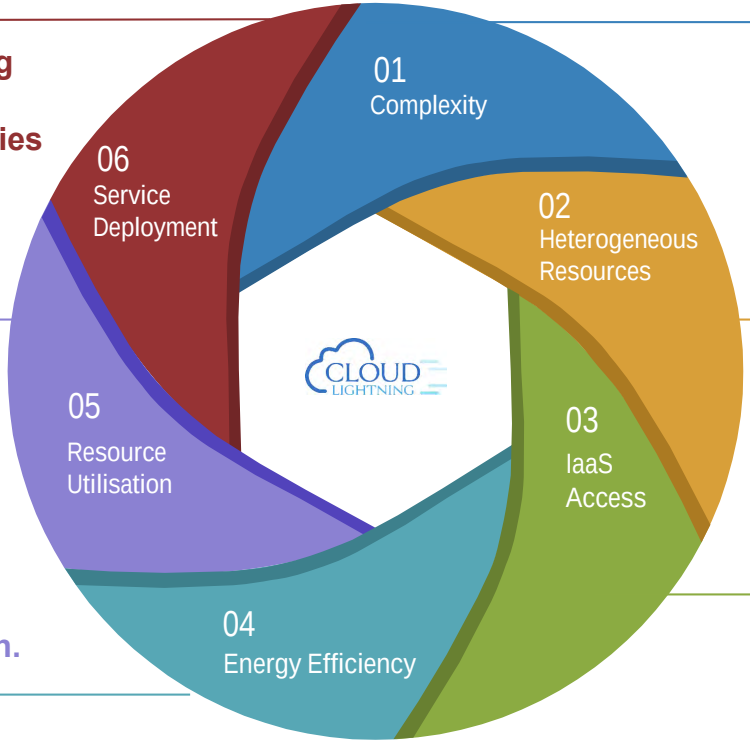
**Testbed setup with 6 node heterogeneous clusters (CPUs + MICs + GPU):**

- 3 Intel Xeon server nodes
- 2 Intel Xeon + XeonPhi server nodes
- 1 Intel Xeon + NVIDIA GPU server node

- Programming models used for Xeon-Phi documented (Offload, native and hybrid models)
- Captured the level of parallelism addressed by Xeon-Phi system (Instruction, data, thread and node level parallelism)
- Theoretical maximas and benchmark results for compute, memory, communication and power usage are reported
- Detailed flowchart on when to select a Xeon-Phi is captured along with different application domains supported

# CloudLightning Approach

CloudLightning proposes a novel architecture for provisioning heterogeneous cloud resources to deliver services, specified by the user, using a bespoke service description language.




The CloudLightning deployment mechanism simplifies the operational overhead for non-technical users

CloudLightning uses self-organisation and self-management to manage complexity effectively.

CloudLightning uses dynamic workload and resource management to increase the efficiency of resource utilisation.

CloudLightning is specifically for heterogeneous hardware

Achieved through heterogeneous resources, reducing overprovisioning, maximising VM/server density and turning off idle servers

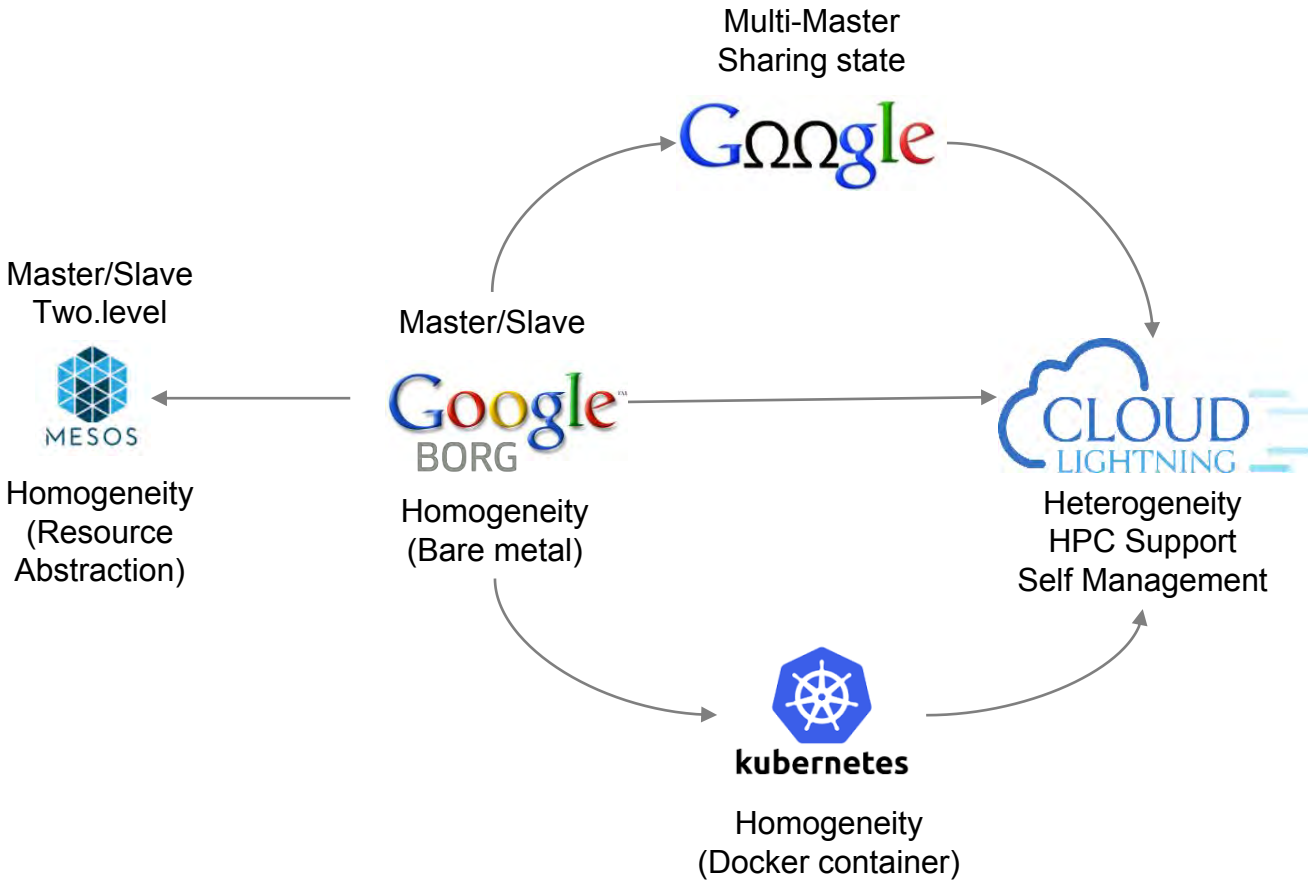
Clear service interface through separation of concerns between consumer and provider.

CLOUD LIGHTNING

# State of the Art

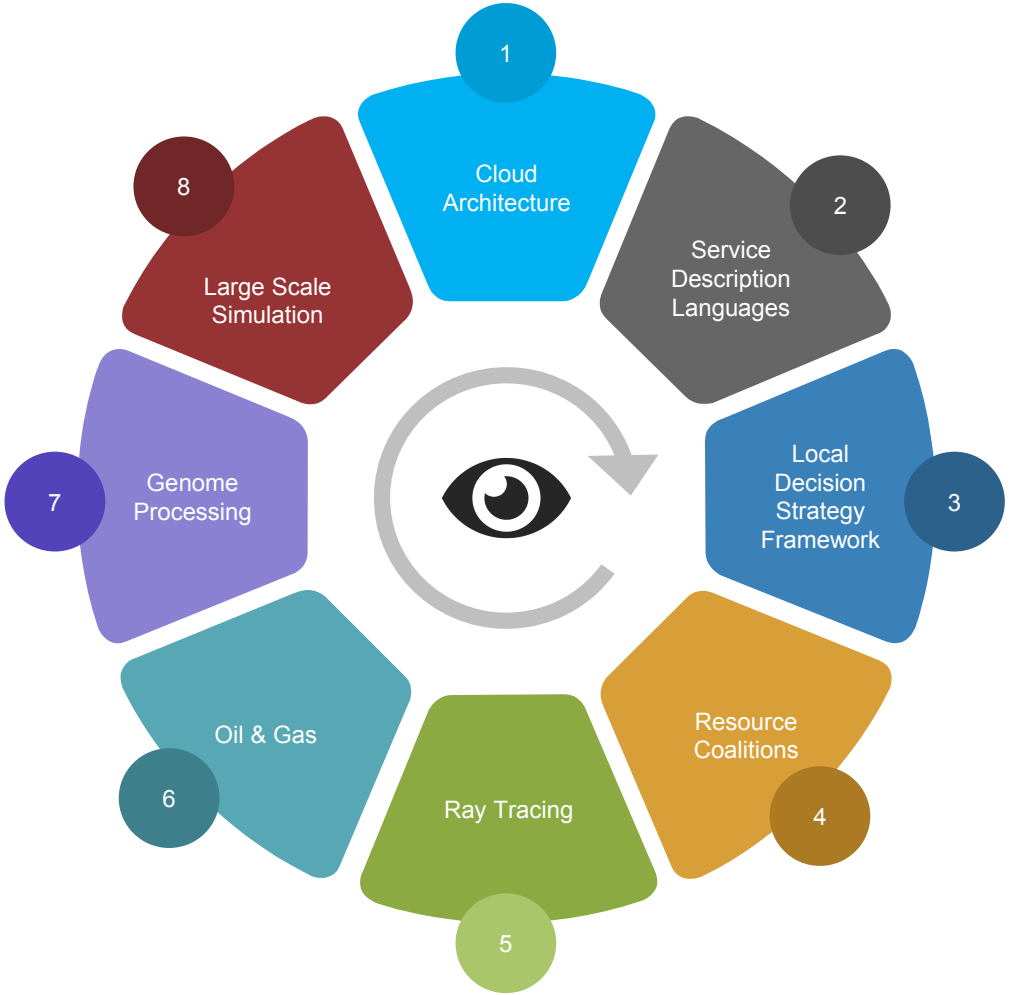
## Analysis of relevant technologies

Most frameworks focusing automatic deployment of applications are limited to manage heterogeneous resources.
The software design of these frameworks is based in a Master/Slave architecture



# Progress Beyond the State of the Art

**CloudLightning is, and will, contribute to progress beyond the state of the art across all technical work packages and primary use cases.**



# Benefits

CloudLightning anticipates a number of general impacts across all use cases including:

- Reduced complexity in deploying use case workloads in the Cloud
- Reduced CAPEX and IT Associated Costs
- Greater Energy Efficiency
- Improved Performance

INTEL and NTNU anticipate open-sourcing their IP under open sourcing licensing.

Maxeler anticipate bringing their DFE solution for genomics to market.

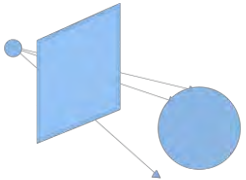
## Oil and Gas

- Improved physics simulations.
- Energy and cost efficient scalable solution for OPM/DUNE simulations.
- Reduced risk and costs of dry exploratory wells

## Genomics

- Improved performance/cost and performance/Watt.
- Faster speed of genome sequence computation.
- Reduced development times.
- Increased volume and quality of related research.

## Ray Tracing



- Reduced CAPEX and IT associated costs.
- Extra capacity for overflow (“surge”) workloads.
- Faster workload processing to meet project timelines.

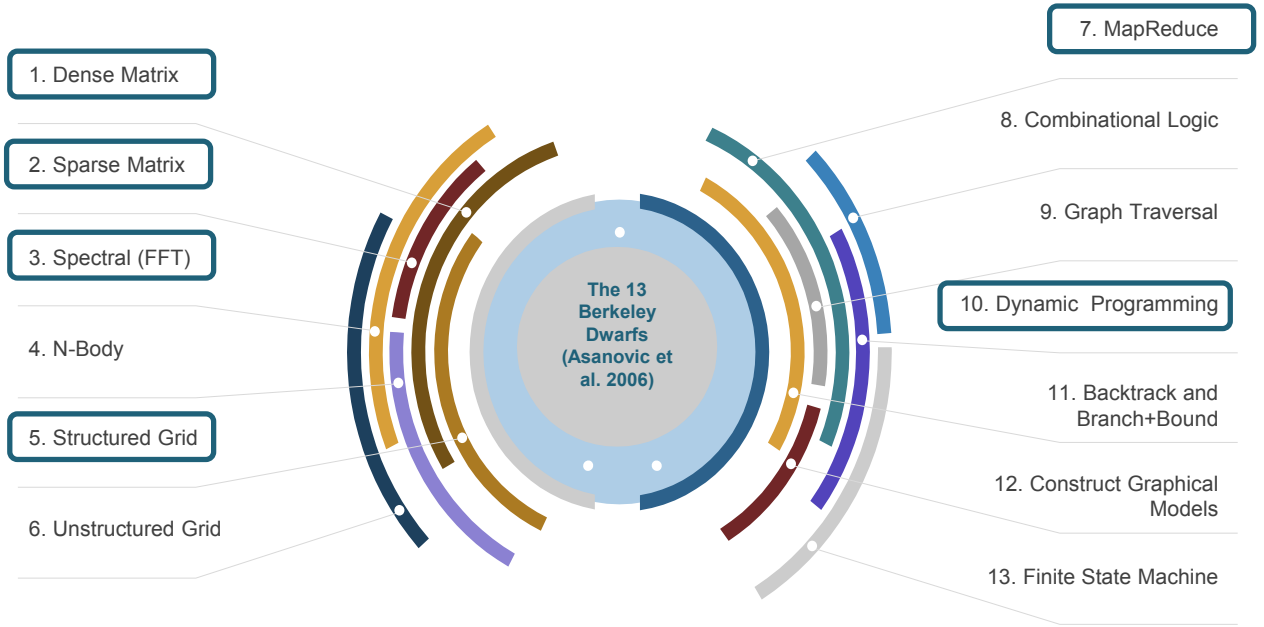
CLOUD LIGHTNING

# Use Case Technical Motivations

**The selection of use cases reflects a number of criteria including but not limited to:**

1. Partner technical expertise and interest
2. Exploit potential for heterogeneous computing
3. Compute-intensive and data-intensive
4. Real-time and batch processing
5. Non-trivial deployment challenges
   - Specialised software/hardware
   - Skilled personnel
6. CloudLightning evaluation potential

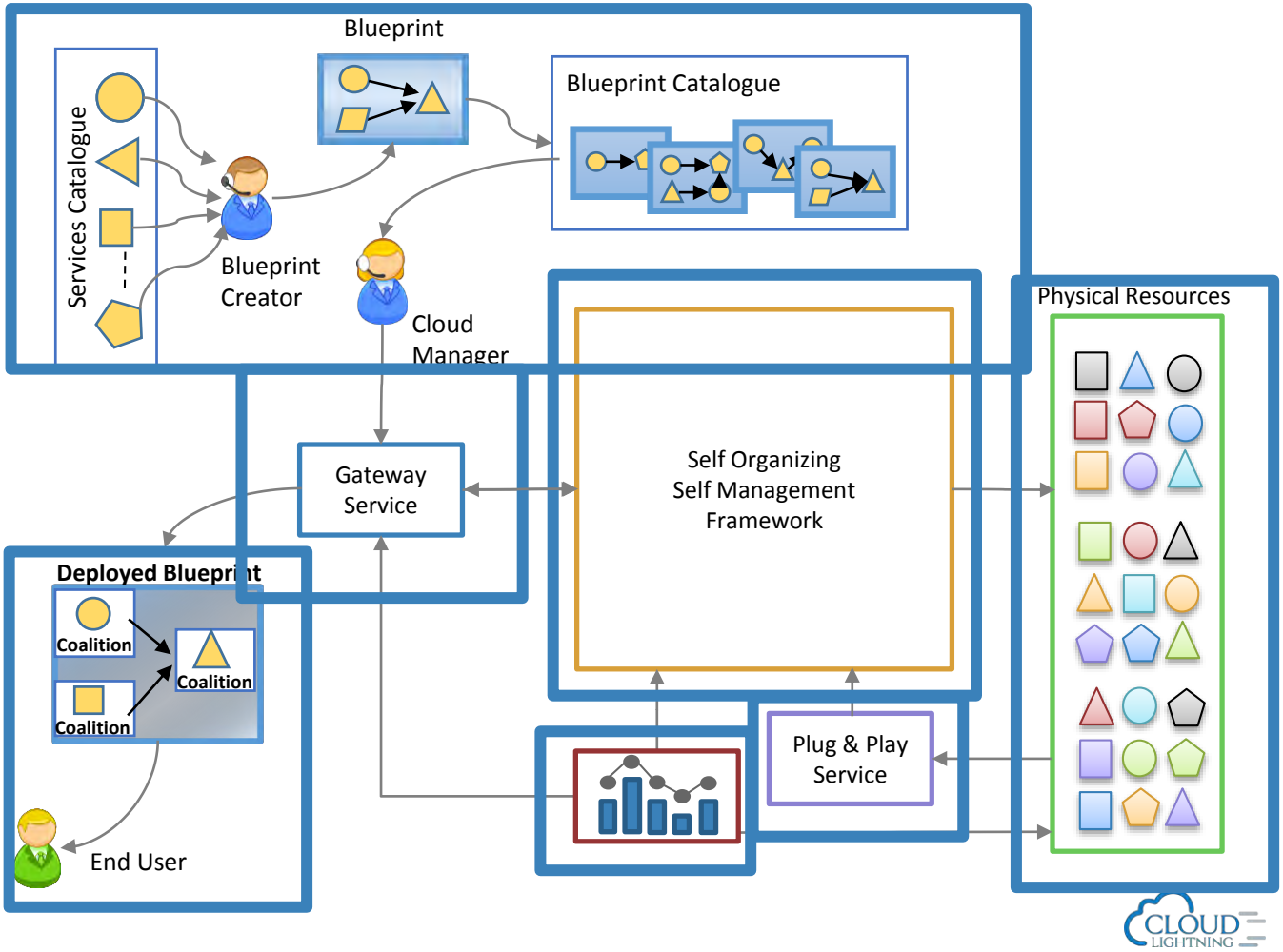
## CL examined a range HPC Grade applications covering 6 out of the 13 Berkeley HPC dwarfs (Asanovic et al. 2006)



# Architecture overview

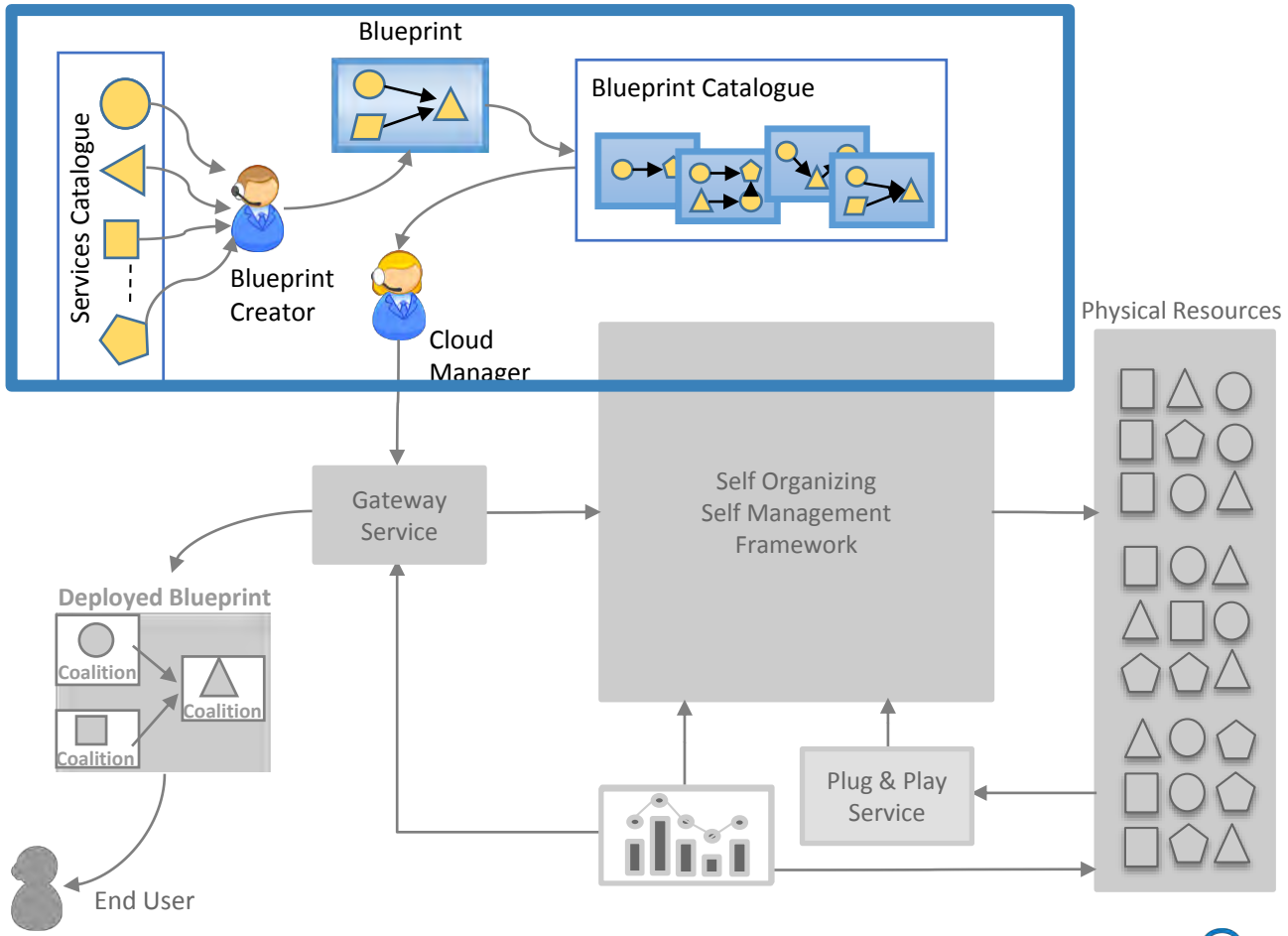
**Components of the CloudLightning System**

- Services and Blueprints
- Gateway service
- SOSM Framework
- Physical resources and coalition management
- Telemetry system
- Plug and Play system



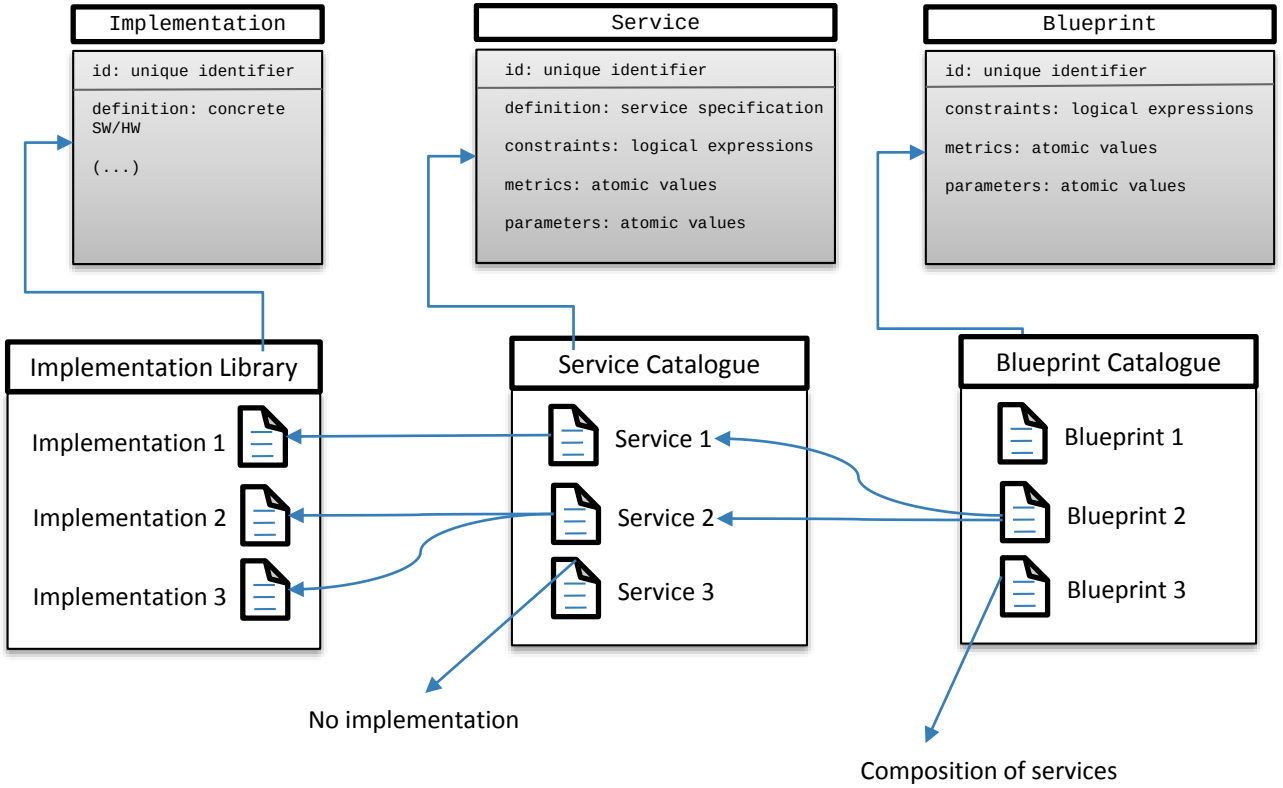
# Blueprints, Service Catalogue and Implementation Library

- A Blueprint is a composition of services.
- A service describes the features of many different hardware types and executable code for the same task.
- An implementation is an executable code on a hardware type of a task.



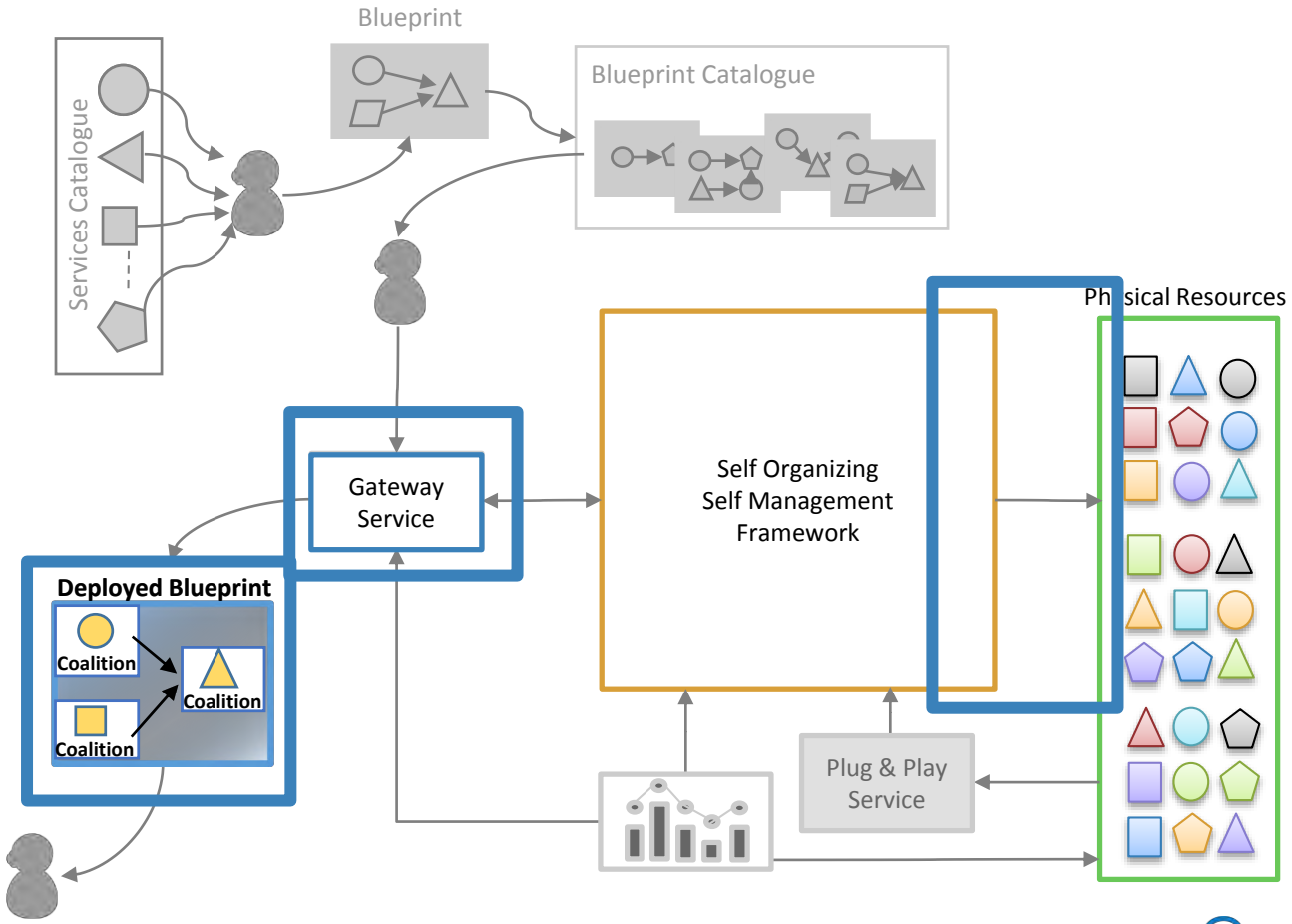
# Blueprints, Service Catalogue and Implementation Library

- A Blueprint is a composition of services.
- A service describes the features of many different hardware types and executable code for the same task.
- An implementation is an executable code on a hardware type of a task.



# Creating a resourced Blueprint

- Use SLA parameters to determine best implementation hardware type.
- Locate resources of the appropriate type.
- Return resource handlers to the Gateway via the Blueprint.
- Invoke the deployment mechanism.



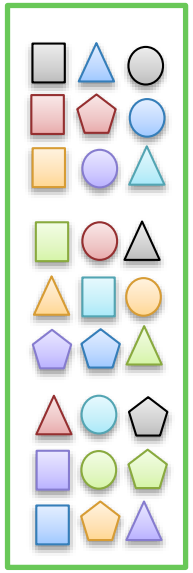
# Management of physical resources

We assume a Cloud with a Resource Fabric far greater than that currently available.

Adding structure to the Cloud Fabric by creating virtual partitions and grouping them together.

- The resource fabric is partitioned into vRacks.
- Each vRack is managed by a vRack Manager.
- A vRack Manager can form Coalitions of its resources to support services.
- vRack Managers self organize to optimize service delivery

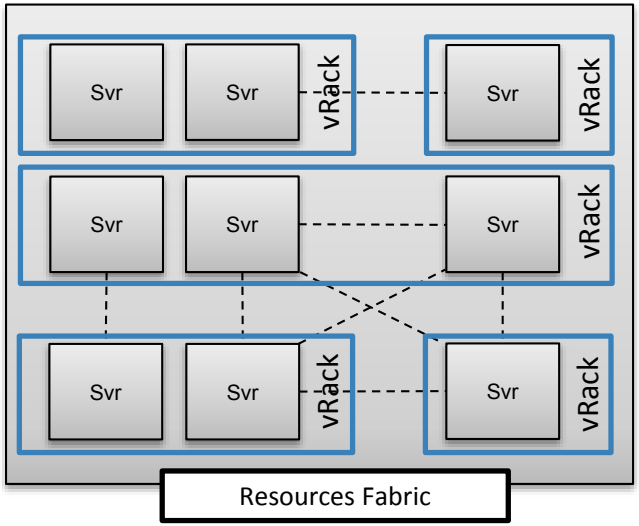
Heterogeneous Physical Resources

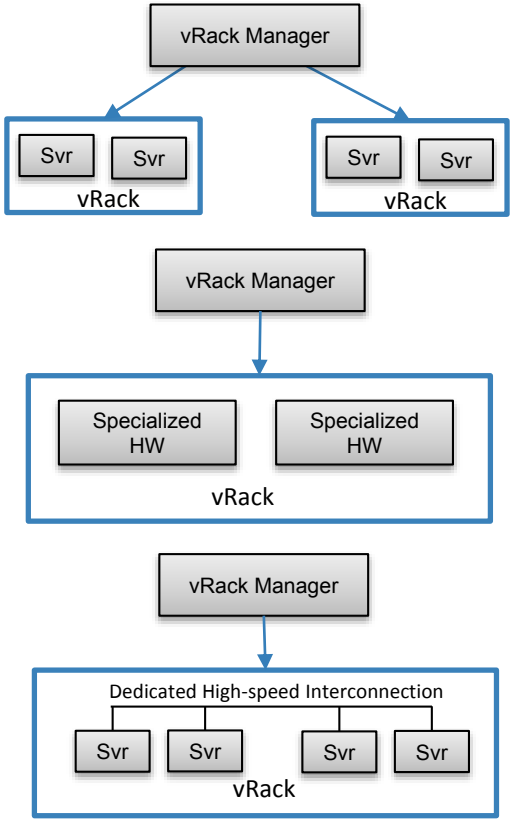


CLOUD LIGHTNING

# vRacks and vRack Managers

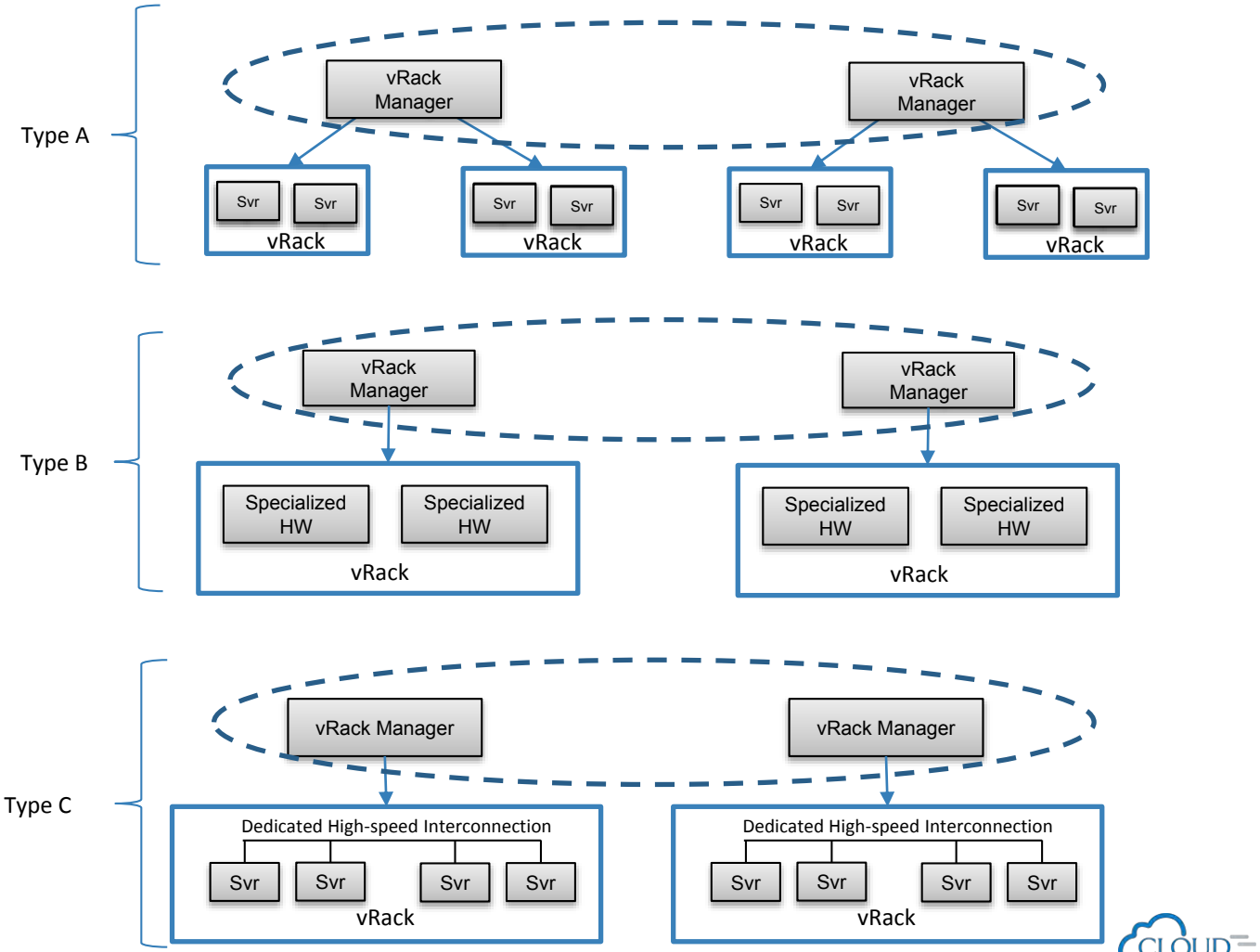
- A vRack is a homogeneous partition of the resource fabric.
- Each vRack is managed by a dedicated vRack Manager.
- vRack Managers of different types exist based on the resource types being managed.



# vRack Manager Groups

- Groups of vRack Managers can be formed to simplify access to resources and to enable self-organization
- There are three types of vRack Manager Groups.

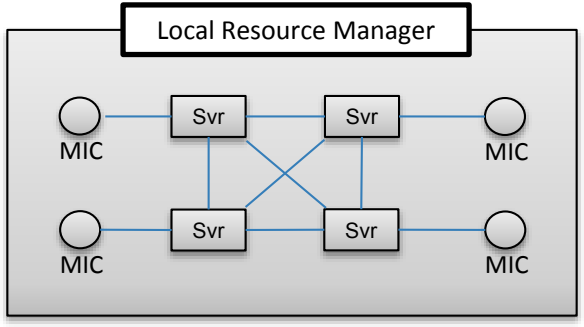


# CL-Resources

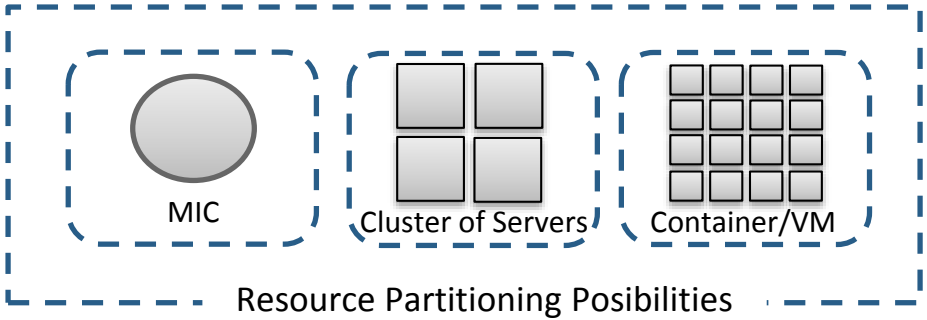
To generically manipulate resources of different types, the SOSM system introduces the concept of a CL-Resource.

CL-Resources refer to different hardware types and to different configurations of those type.

Thus heterogeneity can be introduced dynamically.

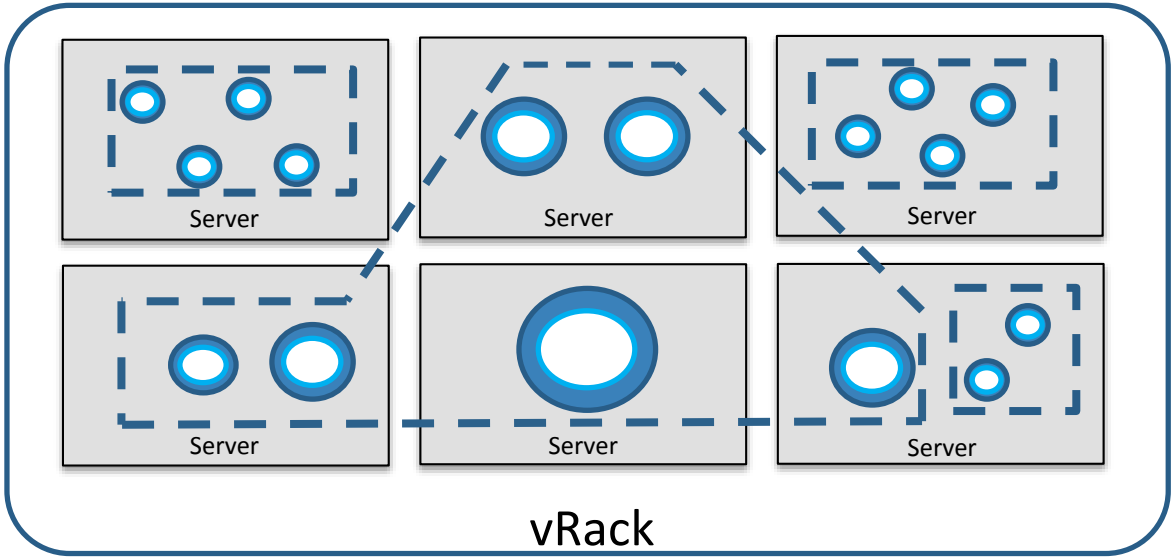



MIC-World



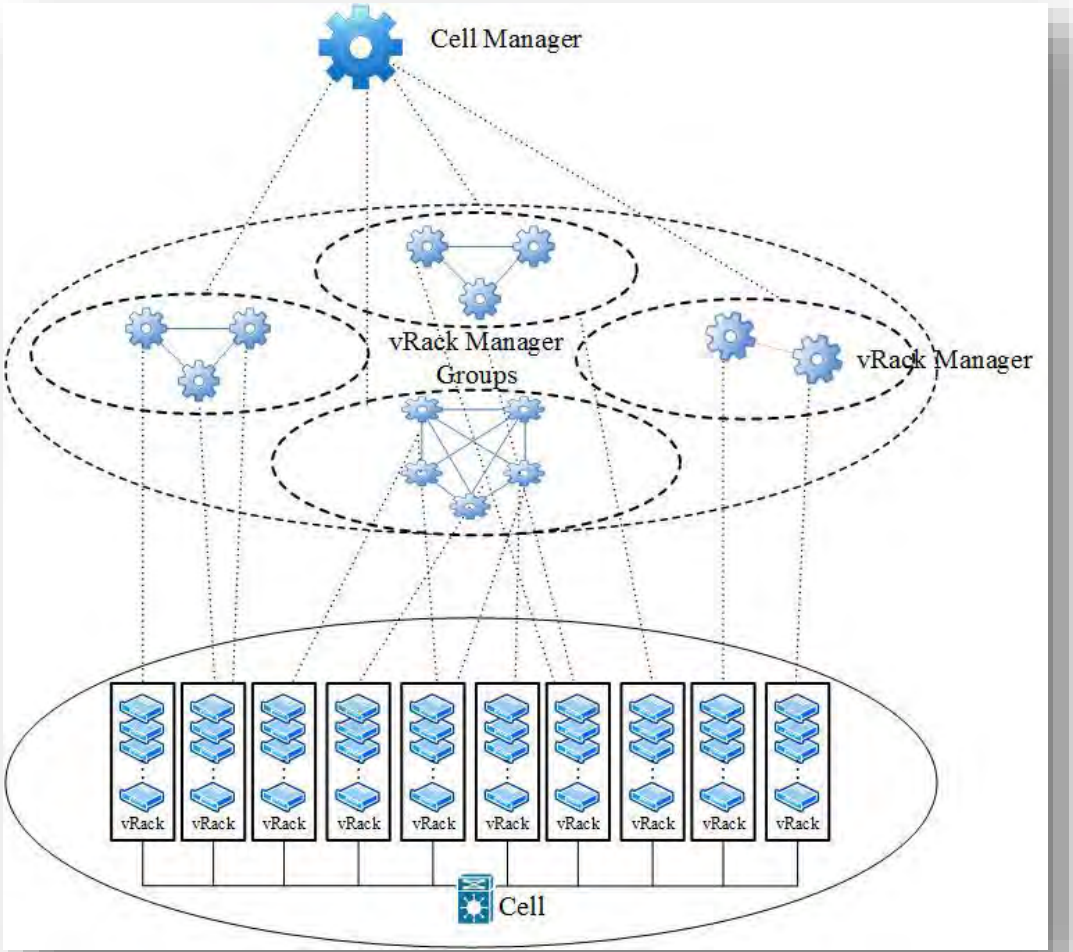
# Coalitions

- Coalitions are used to support the process parallelism within a service.
- Coalitions exist entirely inside a vRack.
- The CL-Resources of a Coalition may span multiple servers within the same vRack.



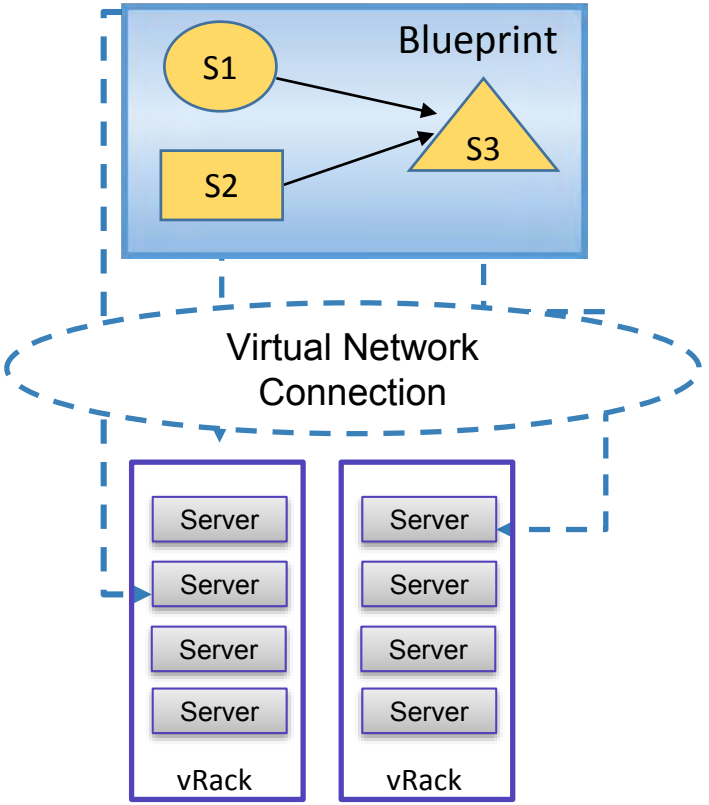
# The conceptual architecture

Architecture showing the components and their relationships.



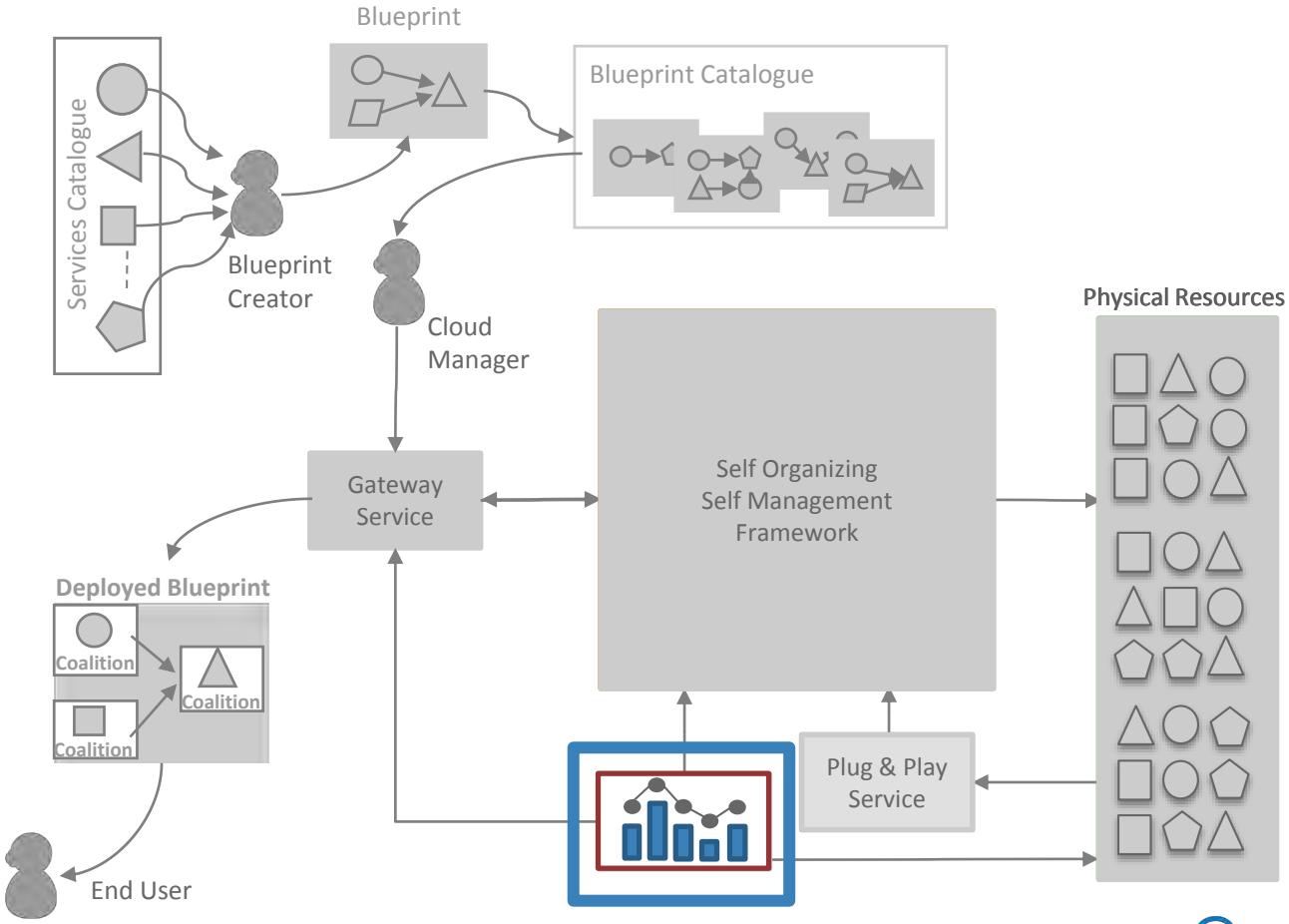
# Advanced architecture support

- VPN creation for Blueprint Service Execution
- Autoscaling
- High availability
- Data localityDynamic



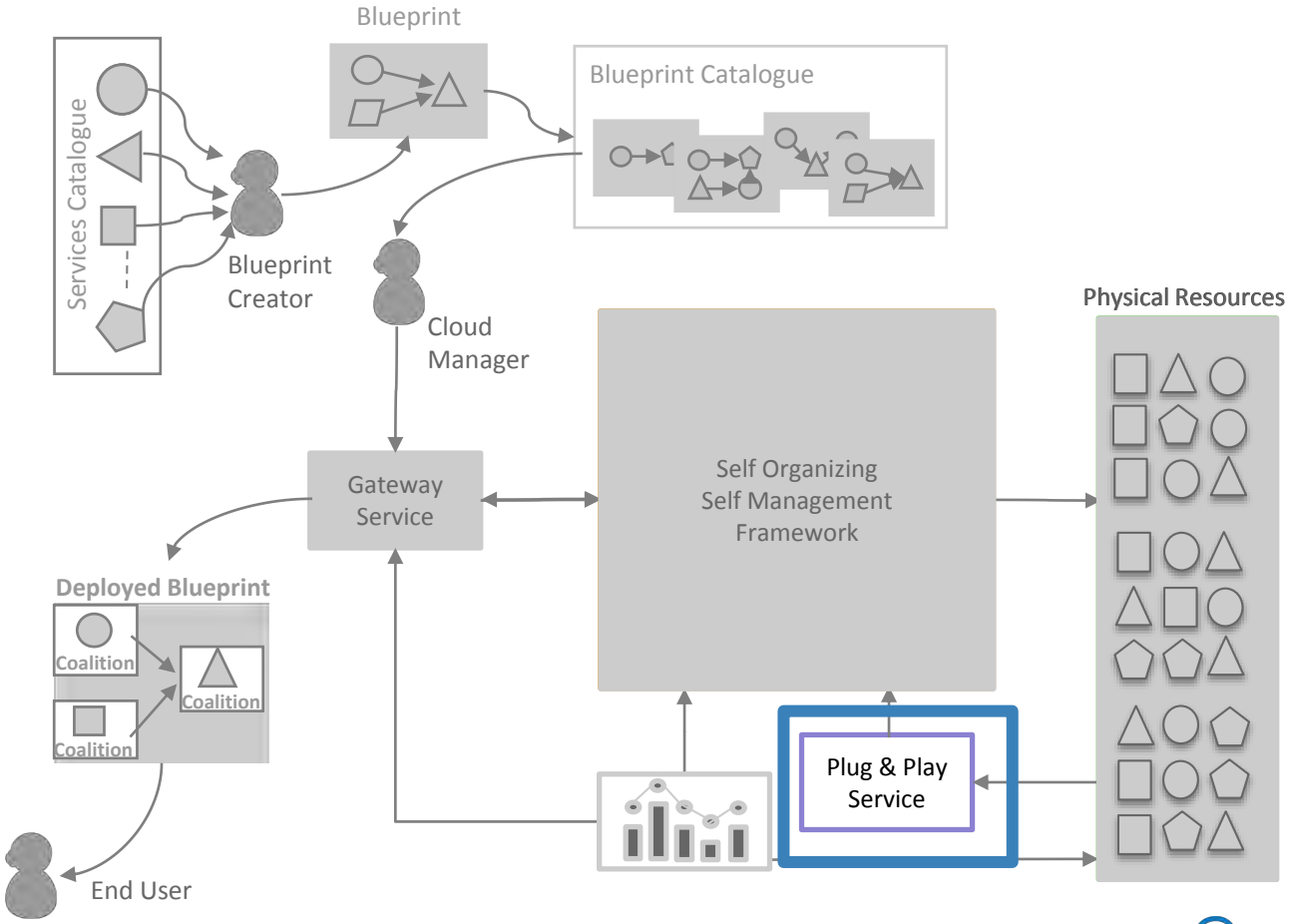
# Determining the local state

The Telemetry system provides to the SOSM system with updates on the status of resources fabric.



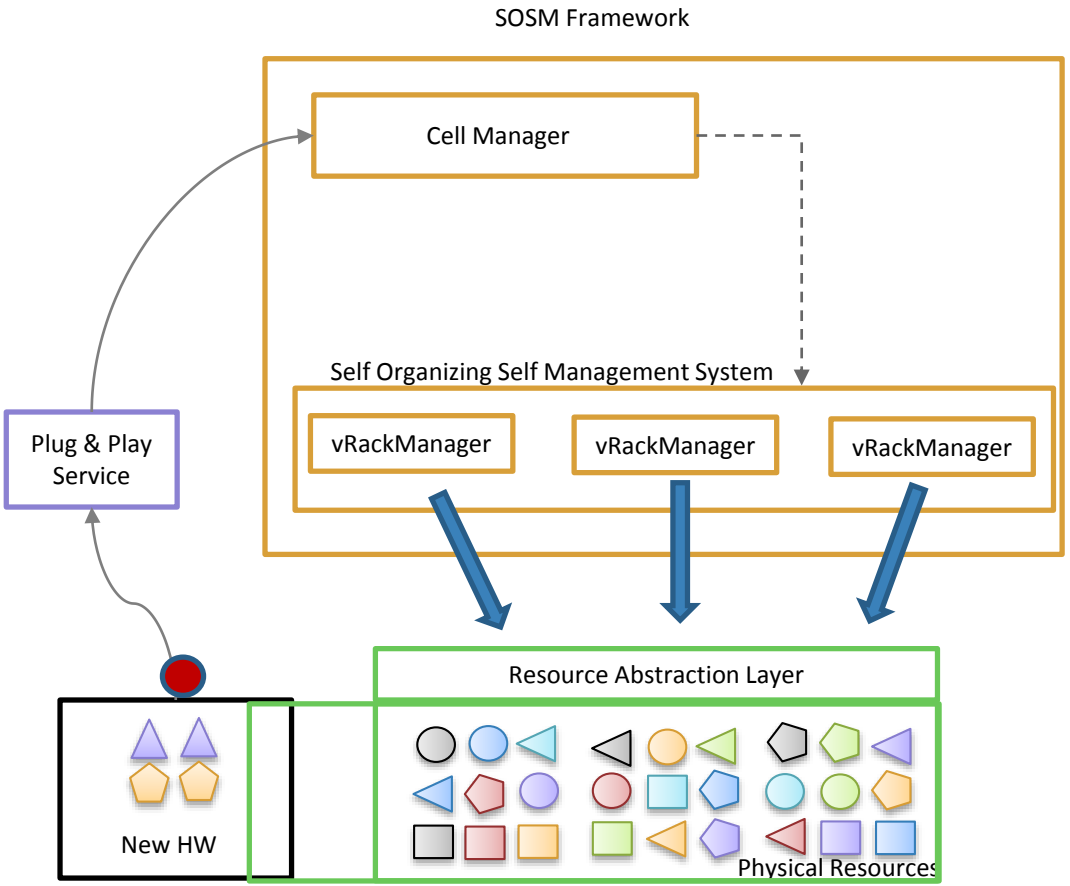
# Support for new hardware

- The SOSM system supports the addition of new hardware by using a plug and play mechanism.
- New hardware can register with SOSM and it is automatically added and managed.



# Support for new hardware

- The SOSM system supports the addition of new hardware by using a plug and play mechanism.
- New hardware can register with SOSM and it is automatically added and managed.



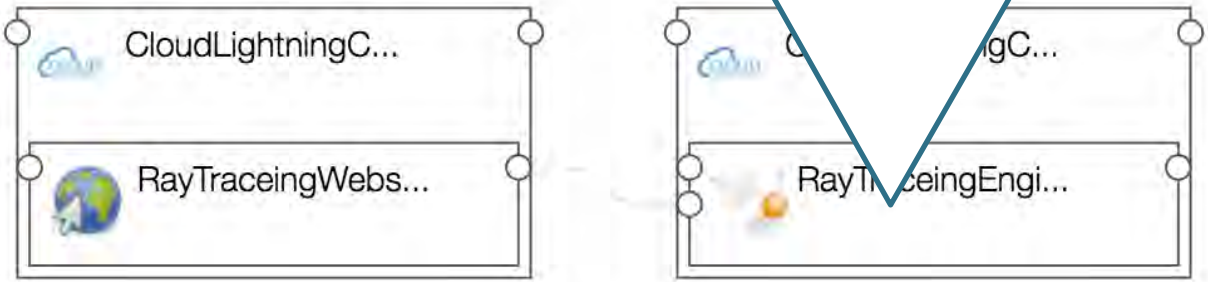
# Ray Tracing Blueprint Example

This example requires a ray tracing cluster selected automatically according to provided service-requirements

A specialized web app is deployed and connected to the previously deployed ray tracing cluster.

TOSCA

```
RayTraceingEngine:
  type: cloudlightning.nodes.meta.RayTraceingEngine
  requirements:
    - host:
        node: CloudLightningCore-2
        capability: tosca.capabilities.Container
        relationship: tosca.relationships.HostedOn
  capabilities:
    cloudlightning:
      properties:
        num_cpus: 1
        is_numascale: false
        has_phi: true
        has_gpu: false
```



# DEMO