

GrayBat - A Graph Approach for Highly Generic Communication Schemes Based on Adaptive Topologies

Erik Zenker

9th March 2015

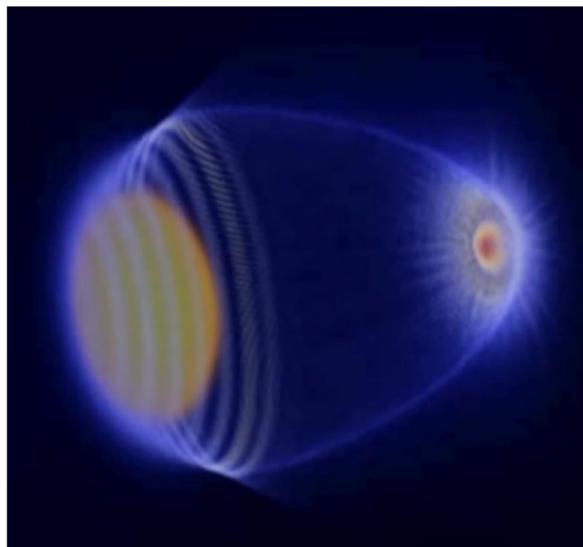


TECHNISCHE
UNIVERSITÄT
DRESDEN

HZDR

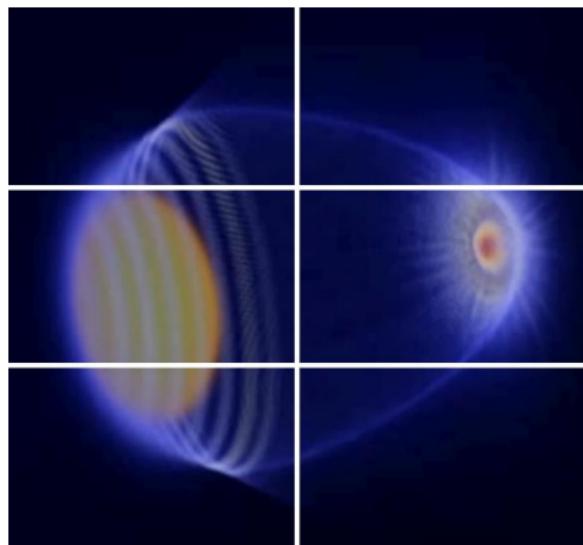
HELMHOLTZ
ZENTRUM DRESDEN
ROSSENDORF

Developing a Distributed Simulation is Simple



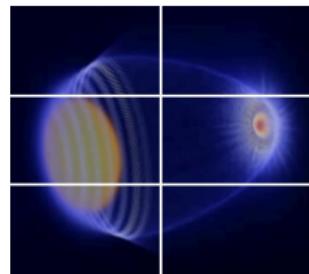
simulation
domain

Developing a Distributed Simulation is Simple

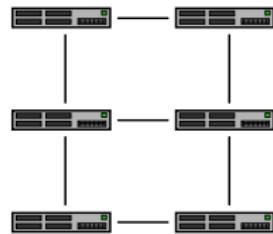


decomposed
domain

Developing a Distributed Simulation is Simple

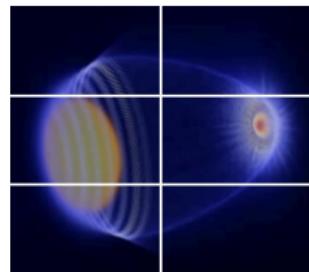


decomposed
domain

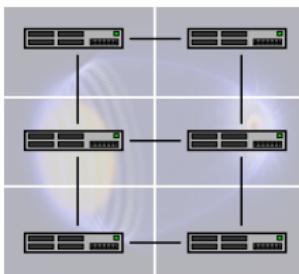
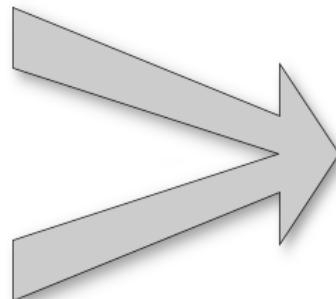


homogeneous
cluster

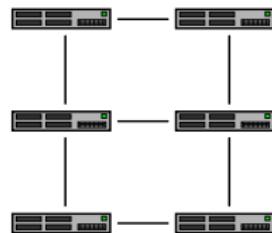
Developing a Distributed Simulation is Simple



decomposed
domain



one-to-one
mapping



homogeneous
cluster

Or Maybe not?



Or Maybe not?



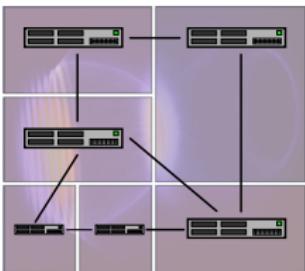
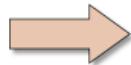
Δt
• • •



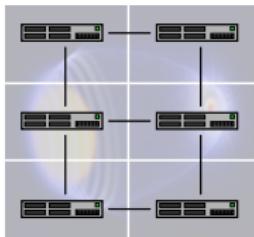
Or Maybe not?



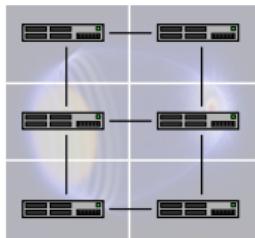
Δt
• • •



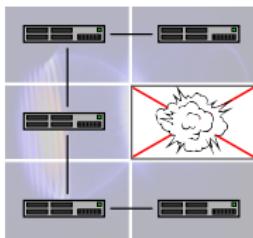
Or Maybe not?



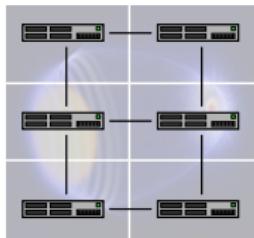
Or Maybe not?



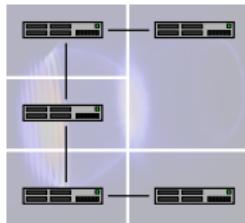
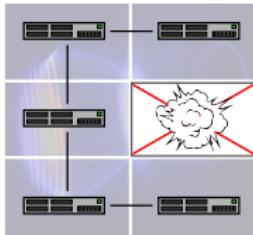
Δt
• • •



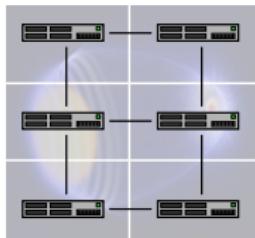
Or Maybe not?



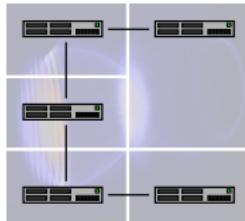
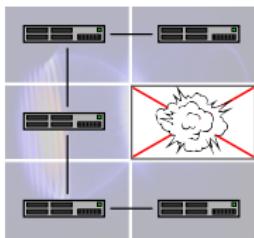
Δt
• • •



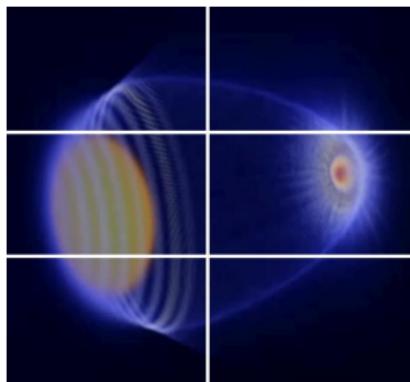
Or Maybe not?



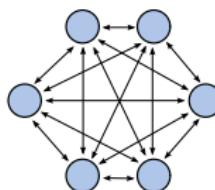
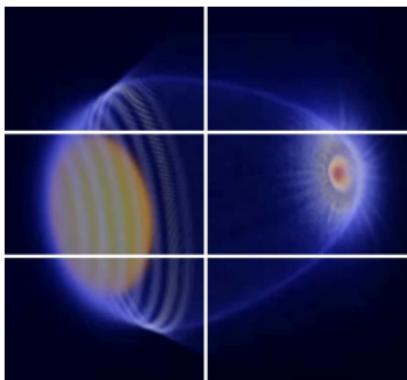
Δt
• • •



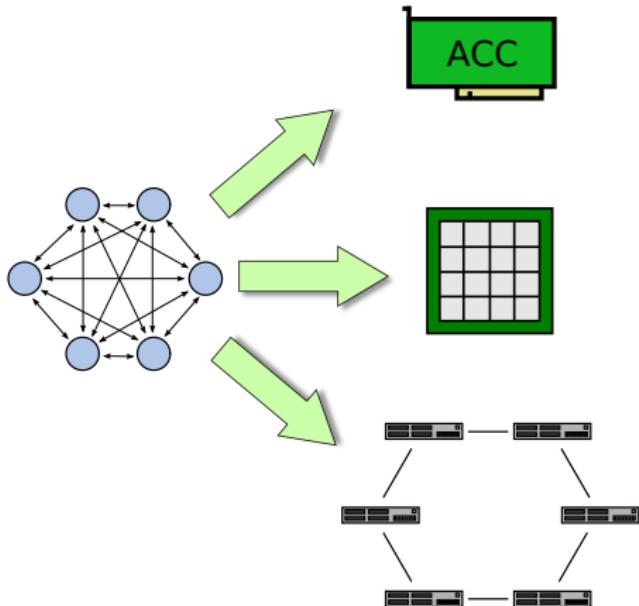
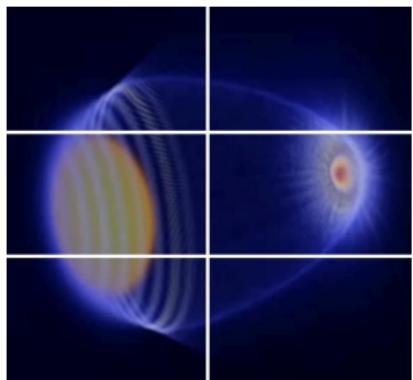
Or Maybe not?



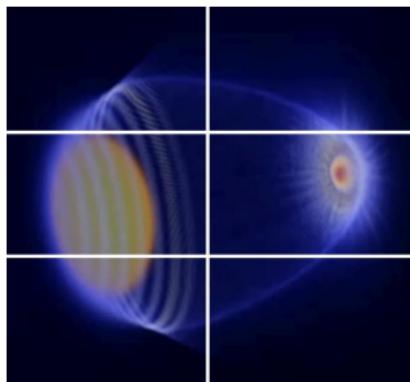
Or Maybe not?



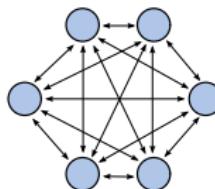
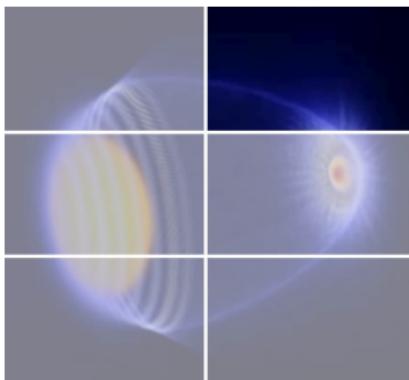
Or Maybe not?



Or Maybe not?

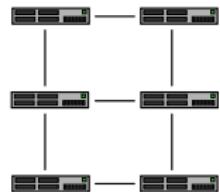
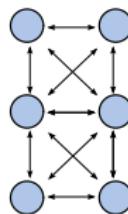
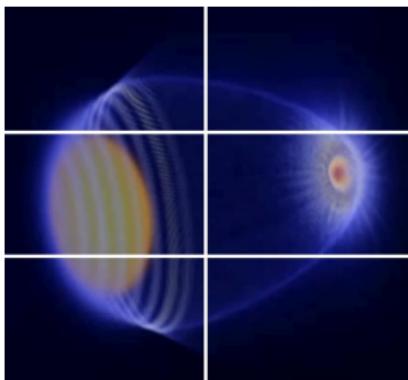


Or Maybe not?



ACC

Or Maybe not?



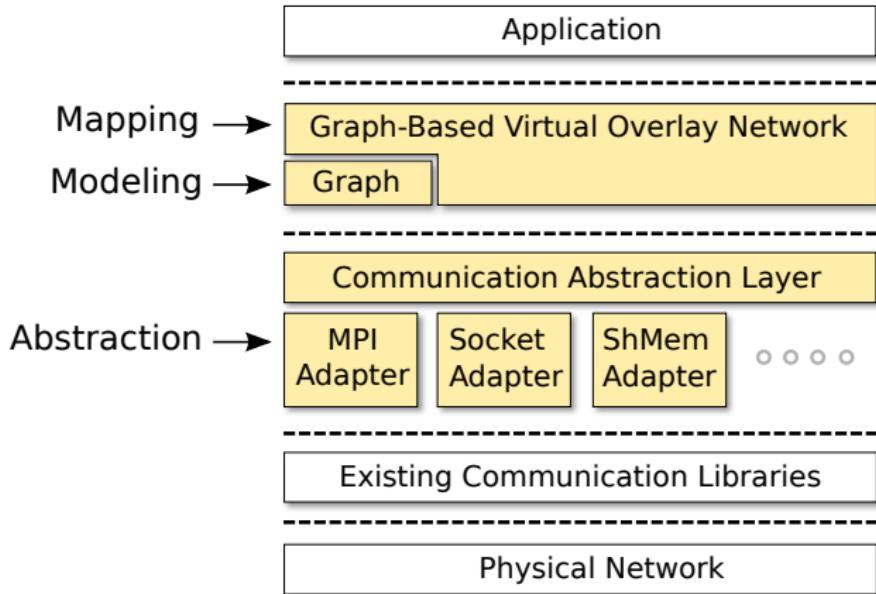
A More Flexible Approach is Required

- **Abstraction** from data transfer methods of hardware topologies
 - Abstraction from MPI send, memcpy, offload etc.
- **Modeling** of communication patterns
 - Next neighbor
 - 3D torus
 - All-to-all
- **Mapping** of communication patterns to hardware topologies

What is this Approach?

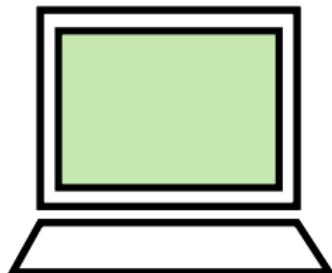
- It is **not** ...
 - a communication framework
 - a load balancing approach
 - a fault tolerance approach
- But, it **is** ...
 - a foundation to solve problems such as load balancing, fault tolerance, and computations in heterogeneous system
 - a mapping from decomposed domain to communication patterns
 - a mapping from communication patterns to hardware topologies

Design Overview



An Abstract Communication Layer is the Basis

peer



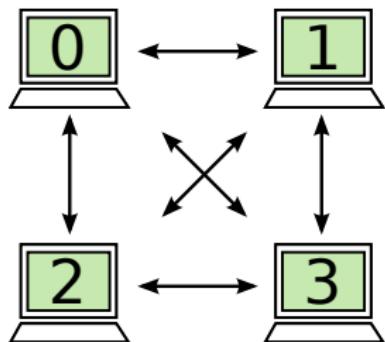
- A peer is an instance participating on communication

An Abstract Communication Layer is the Basis



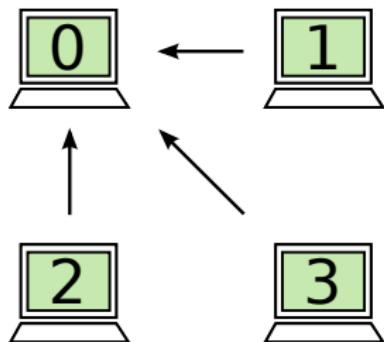
- A peer is an instance participating on communication
- Each peer is identified by a virtual address

An Abstract Communication Layer is the Basis



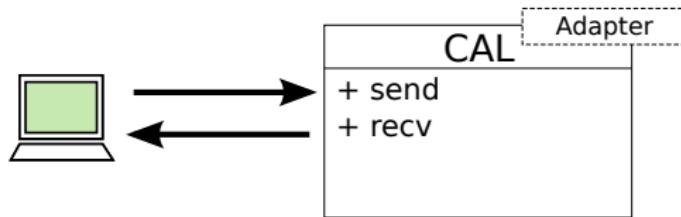
- A peer is an instance participating on communication
- Each peer is identified by a virtual address
- Peers exchange data
 - Point-to-point operations
 - Collective operations

An Abstract Communication Layer is the Basis

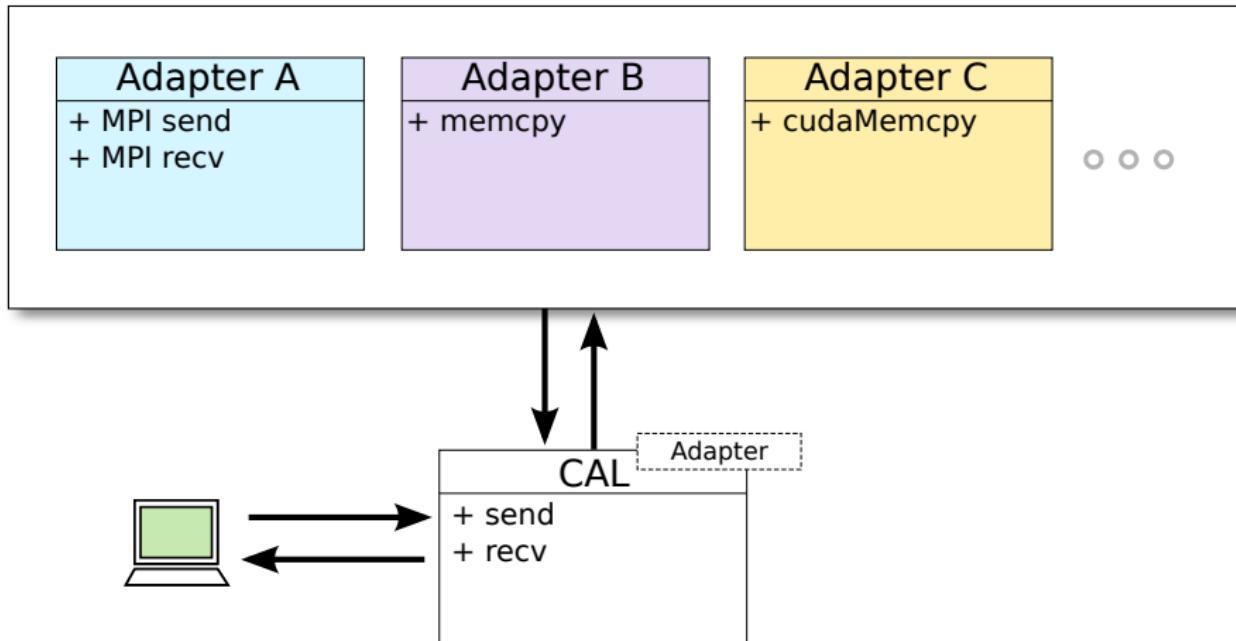


- A peer is an instance participating on communication
- Each peer is identified by a virtual address
- Peers exchange data
 - Point-to-point operations
 - Collective operations

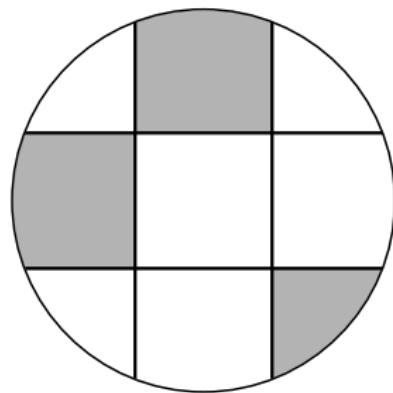
Communication is not Newly Invented



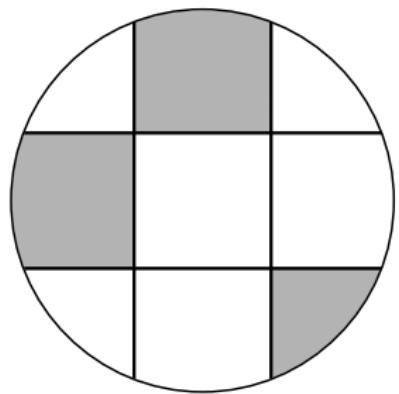
Communication is not Newly Invented



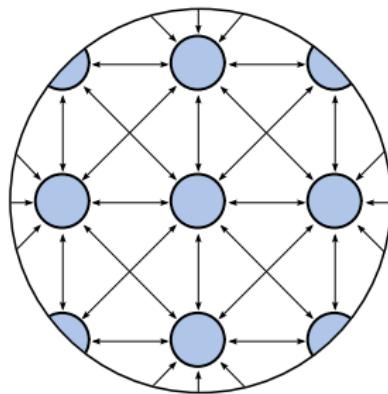
A Graph, the Most General Description



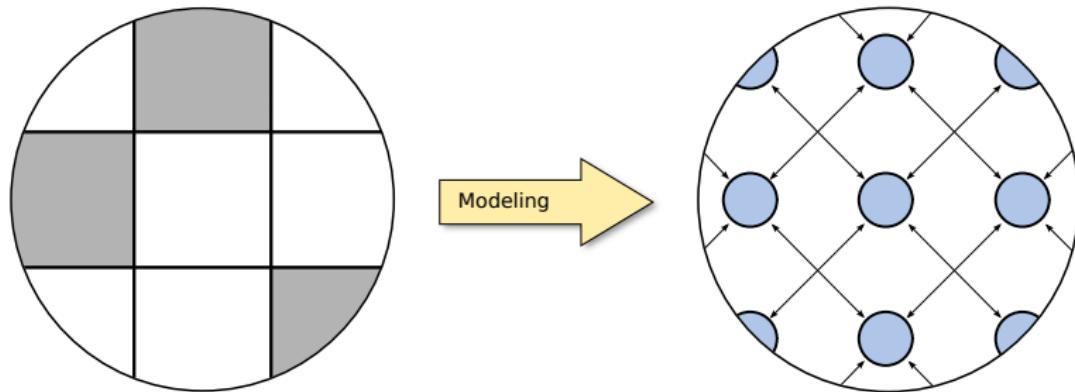
A Graph, the Most General Description



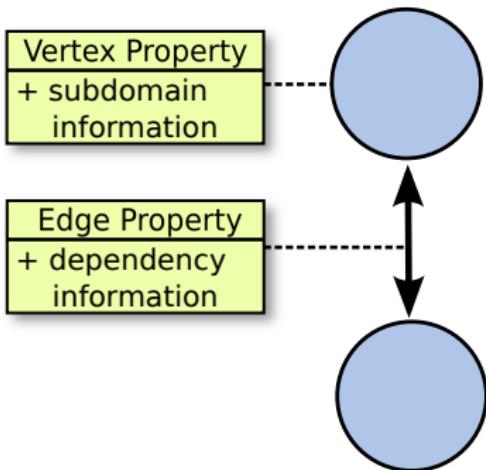
Modeling



Lets Change my World

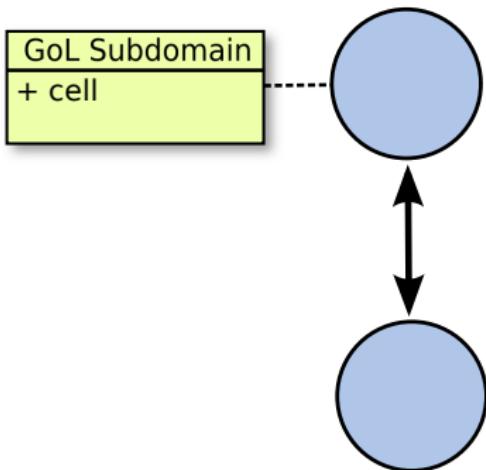


A Graph Can Provide More Information!



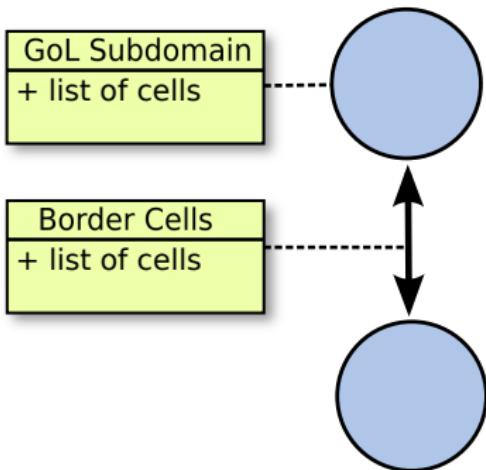
- Properties cover simulation domain specific information
- Properties are bounded to vertices and edges

A Graph Can Provide More Information!



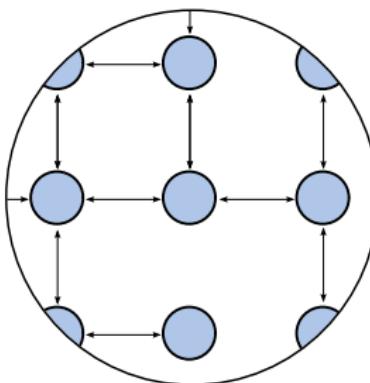
- Properties cover simulation domain specific information
- Properties are bounded to vertices and edges

A Graph Can Provide More Information!



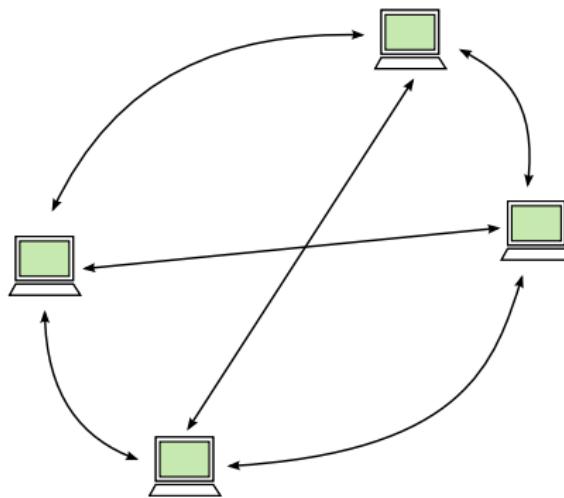
- Properties cover simulation domain specific information
- Properties are bounded to vertices and edges

Emerge the Flexibility by Combination



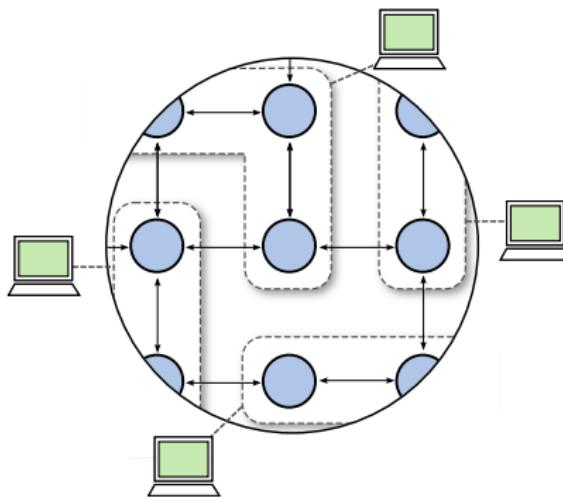
- The **Graph** provides the communication pattern

Emerge the Flexibility by Combination



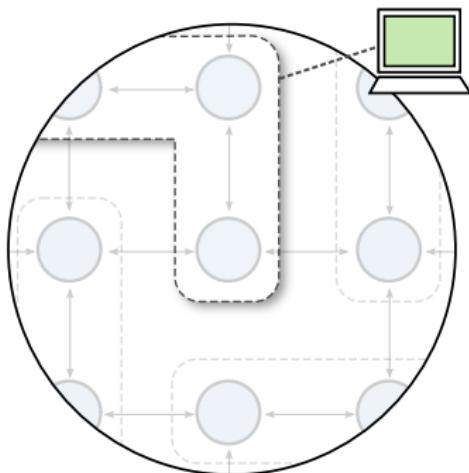
- The **Graph** provides the communication pattern
- The **CAL** provides communication abilities

Emerge the Flexibility by Combination



- The **Graph** provides the communication pattern
- The **CAL** provides communication abilities
- Together it creates a **Graph-based Virtual Overlay Network**

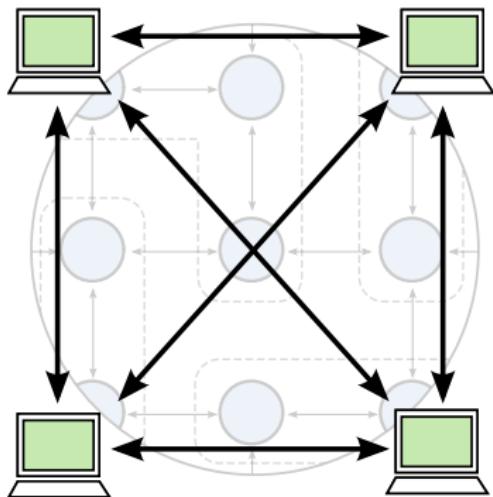
The GVON Maps Vertices to Peers



1 Distribution

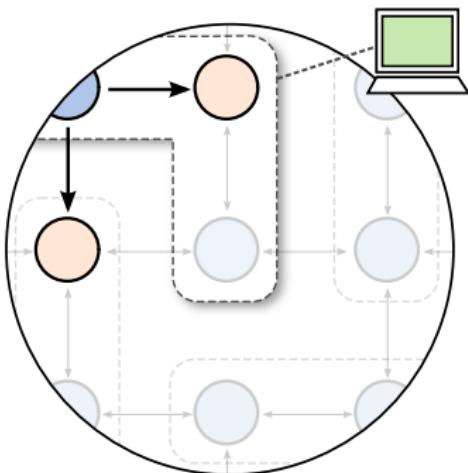
→ Distribute vertices to peers

The GVON Maps Vertices to Peers



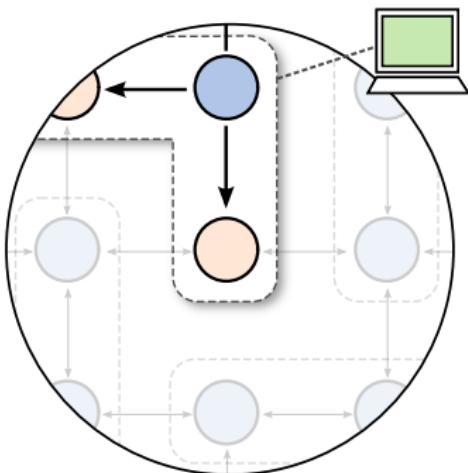
- 1 Distribution
 - Distribute vertices to peers
- 2 Announcement
 - Update mapping

The GVON Maps Vertices to Peers



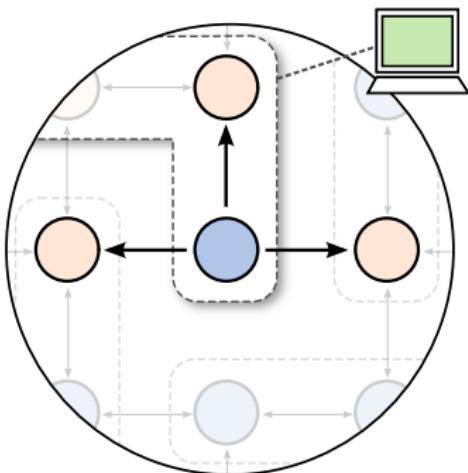
- 1 Distribution
 - Distribute vertices to peers
- 2 Announcement
 - Update mapping
- 3 Communication
 - Point-to-point operations
 - Collective operations

The GVON Maps Vertices to Peers



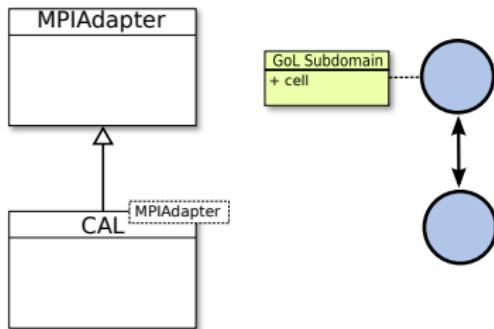
- 1 Distribution
 - Distribute vertices to peers
- 2 Announcement
 - Update mapping
- 3 Communication
 - Point-to-point operations
 - Collective operations

The GVON Maps Vertices to Peers



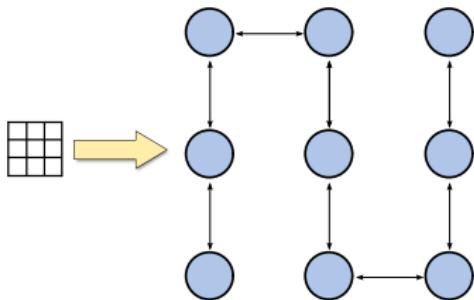
- 1 Distribution
→ Distribute vertices to peers
- 2 Announcement
→ Update mapping
- 3 Communication
→ Point-to-point operations
→ Collective operations

A Simulation in Three Steps



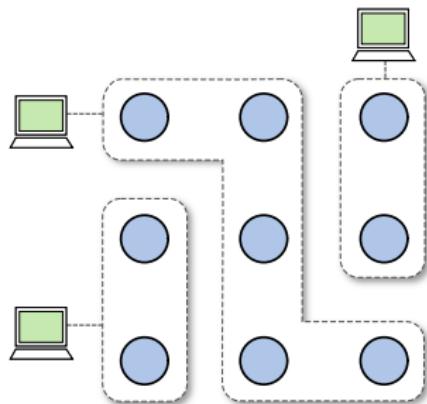
- 1 Configuration at compile-time
- CAL with MPI adapter
 - Graph with cell property
 - GVON with specific Graph and CAL

A Simulation in Three Steps



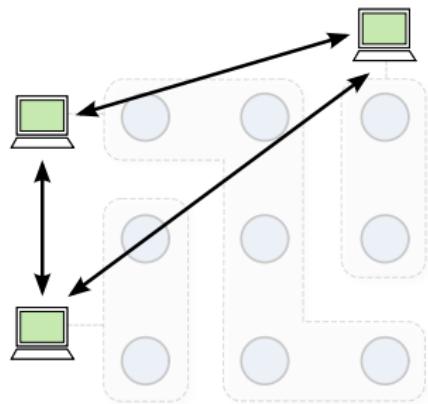
- 1 Configuration at compile-time
 - CAL with MPI adapter
 - Graph with cell property
 - GVON with specific Graph and CAL
- 2 Initialization
 - Create Gol graph
 - Distribute vertices to peers
 - Announce vertices

A Simulation in Three Steps



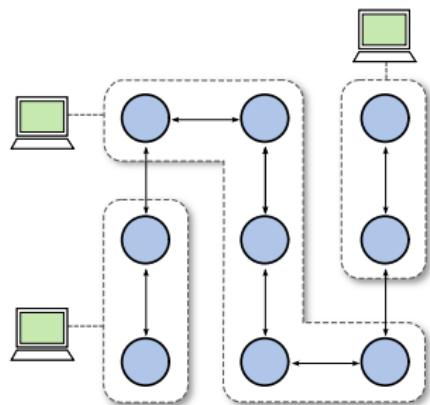
- 1 Configuration at compile-time
 - CAL with MPI adapter
 - Graph with cell property
 - GVON with specific Graph and CAL
- 2 Initialization
 - Create Gol graph
 - Distribute vertices to peers
 - Announce vertices

A Simulation in Three Steps



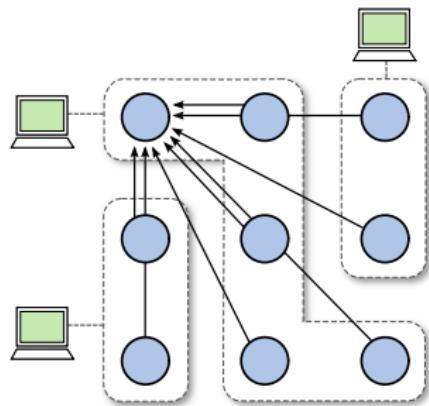
- 1 Configuration at compile-time
 - CAL with MPI adapter
 - Graph with cell property
 - GVON with specific Graph and CAL
- 2 Initialization
 - Create Gol graph
 - Distribute vertices to peers
 - Announce vertices

A Simulation in Three Steps



- 1 Configuration at compile-time**
 - CAL with MPI adapter
 - Graph with cell property
 - GVON with specific Graph and CAL
- 2 Initialization**
 - Create Gol graph
 - Distribute vertices to peers
 - Announce vertices
- 3 Communication**
 - Exchange data of neighboring cells
 - Collect state information of all cells

A Simulation in Three Steps



- 1 Configuration at compile-time**
 - CAL with MPI adapter
 - Graph with cell property
 - GVON with specific Graph and CAL
- 2 Initialization**
 - Create Gol graph
 - Distribute vertices to peers
 - Announce vertices
- 3 Communication**
 - Exchange data of neighboring cells
 - Collect state information of all cells