LaBRI

Satanas: Supports and Algorithms for High Performance Numerical Applications

Raymond Namyst

université de BORDEAUX

Team

→ Satanas gathers 4 Inria joint research teams

- Bacchus
  - Parallel tools for Numerical Algorithms and Resolution of essentially Hyperbolic problems
- HiePACS
  - High-end Parallel Algorithms for Challenging numerical Simulations
- Runtime
  - High Performance Runtime Systems for Parallel Architectures
- Phoenix (joined 2 years ago)
  - A Multi-Disciplinary Approach to Orchestrating Networked Entities

→ Composition

- 18 permanent members
- 29 PhD candidates

LaBRI

Context and Scientific Activities

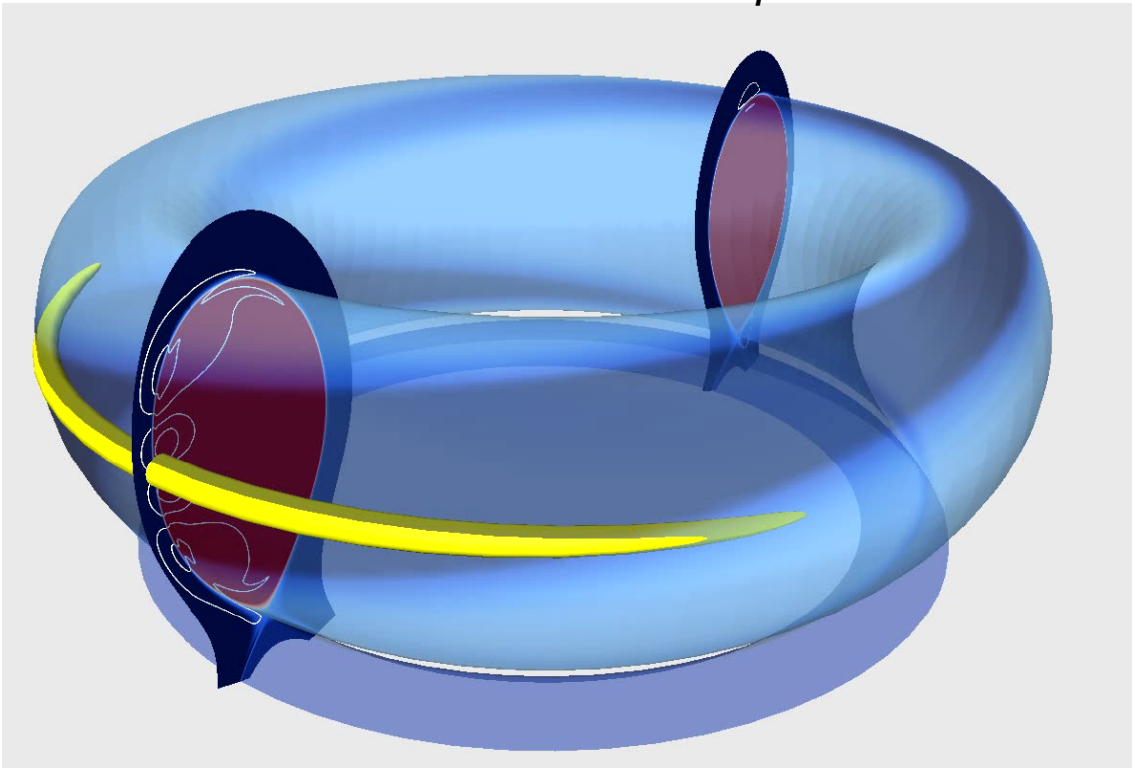
université de BORDEAUX

High Performance Computing

→ Highly demanding numerical simulations
  › Energy, Materials, Aeronautics, Seismology, Weather Forecast, etc.

→ Multi-scale, multi-physics problems
  › Code coupling

ITER: International Thermonuclear Experimental Reactor



Understanding the evolution of parallel machines

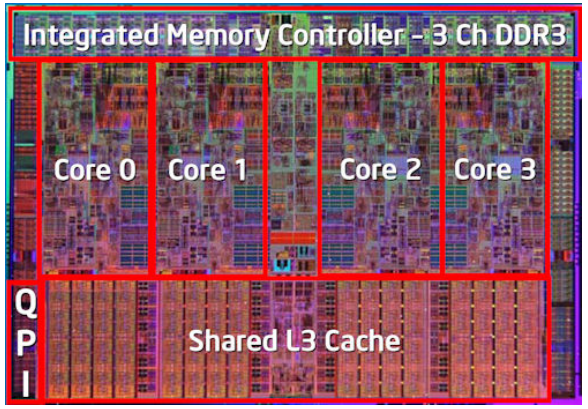
→ The end of single thread performance increase
- Clock rate is no longer increasing
- Thermal dissipation

→ Processor architecture is already very sophisticated
- Prediction and prefetching techniques achieve a very high percentage of success
- Actually, processor complexity is decreasing…

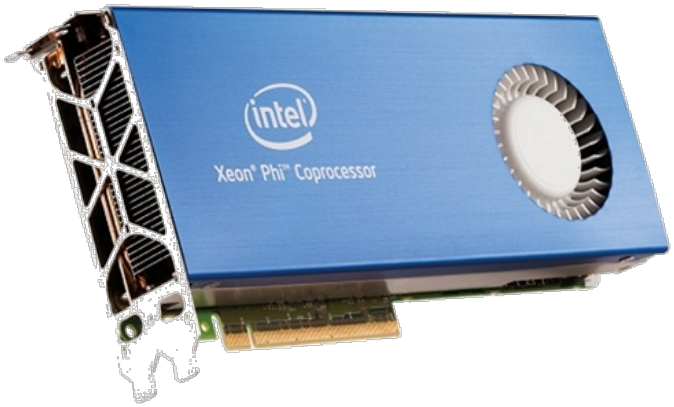
→ Question: What kind of circuits should we add on a chip?

The evolution of computer architecture

- Multicore chips
  - Deep memory hierarchies
    - Non Uniform Memory Access
    - Non Uniform I/O Access (NUIOA)
  - Non-coherent cache architectures
    - Intel SCC, IBM Cell/BE
  - Clusters can no longer be considered as "flat sets of processors"

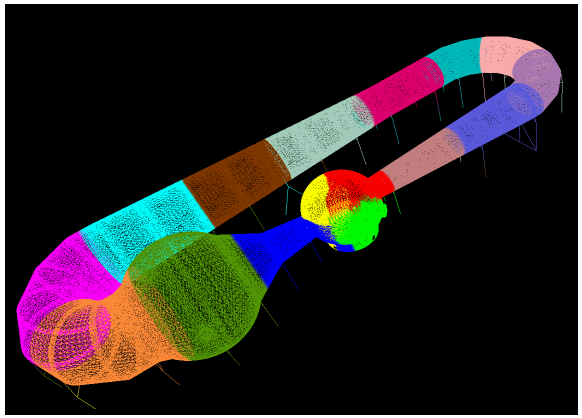


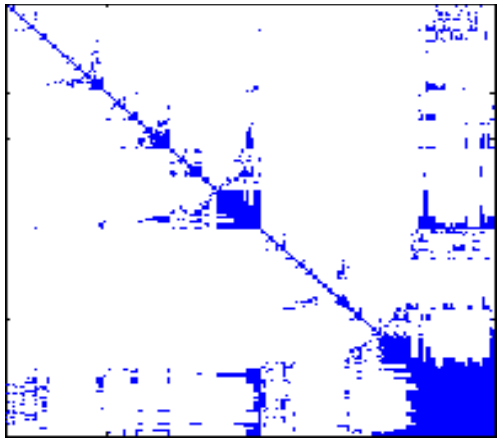
- Accelerators
  - Nvidia & AMD GPUs
  - Intel MIC
  - Intel Ivy Bridge, AMD APUs
  - Different execution model



Performing simulations over parallel machines

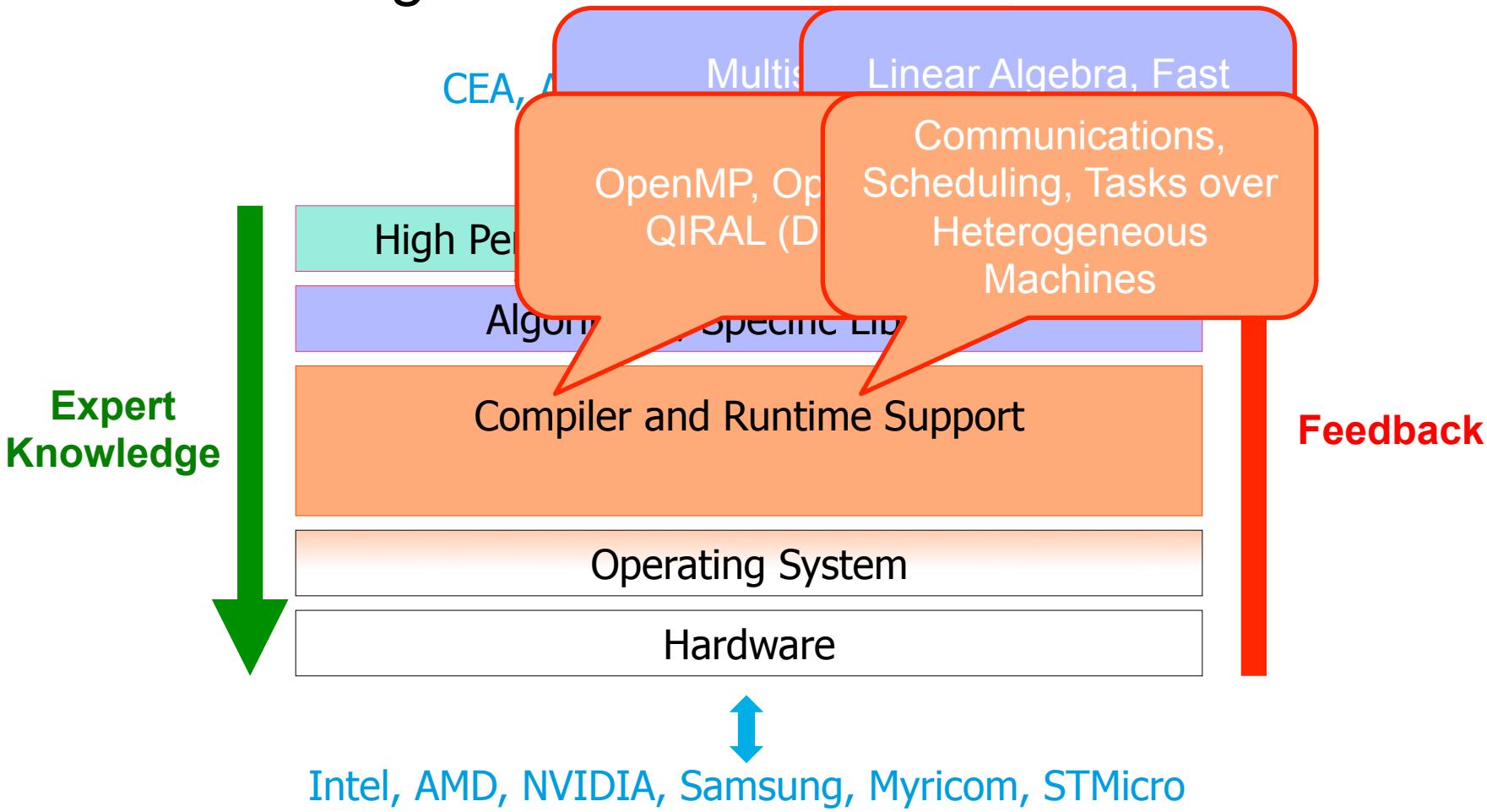
- Discretization
  - Meshes, finite elements, etc.
- Data (re)distribution
  - Graph/mesh partitioning techniques
- Parallelization
  - Linear algebra solvers
  - N-body computations
  - Stencils
- Code optimization
  - Domain-specific languages
  - Code analysis, profiling
- Runtime Systems
  - Architecture abstraction
  - Communication protocols
  - Scheduling, load balancing





A Holistic and Multidisciplinary Approach

→ Addressing the whole stack



LaBRI

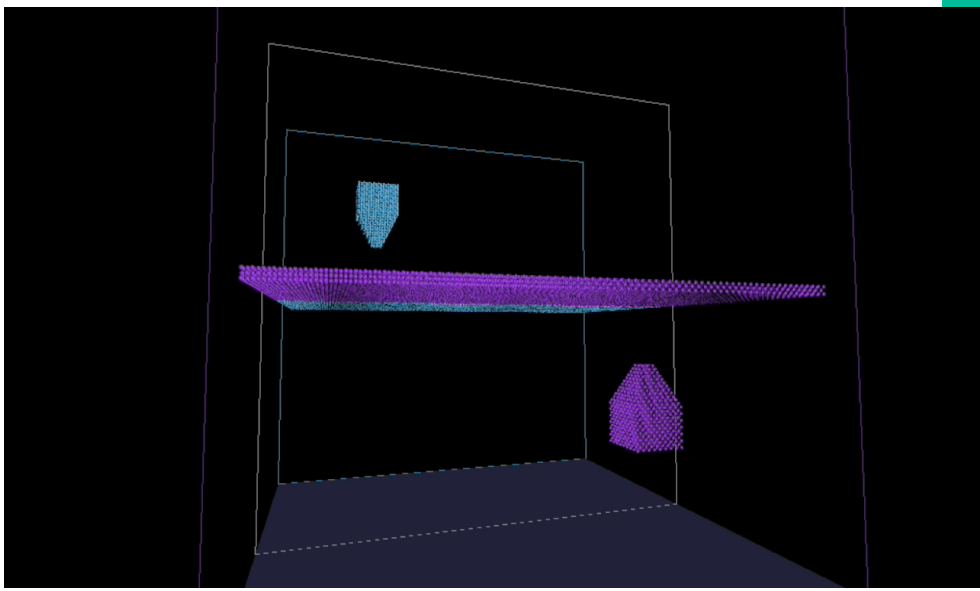
Major Achievements

cnrs

université de BORDEAUX

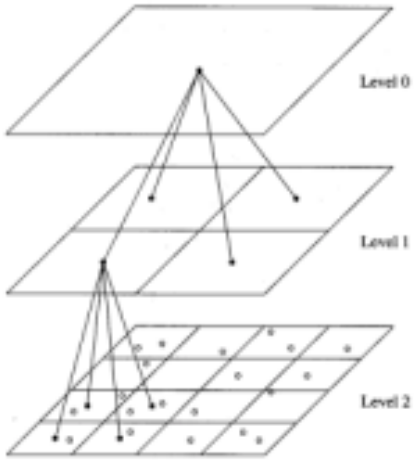
Bordeaux INP AQUITAINE

Accurate simulation of Materials



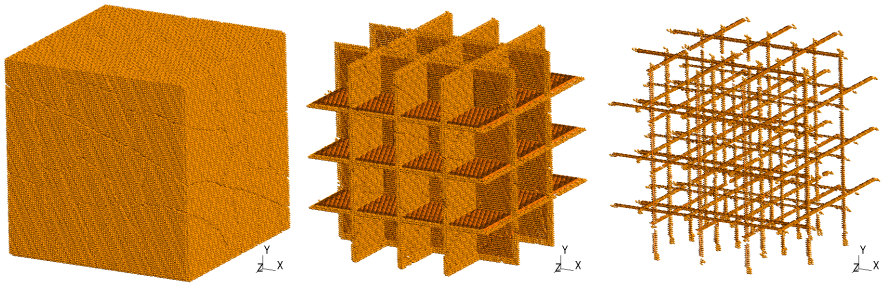
→ N-Body computations
- astrophysics, material physics, biology, electromagnetism
  - ExaStamp: hundreds of billions of atoms over heterogeneous clusters [CEA/DPTA]

→ Fast Multipole Method
- Reduce computational complexity from $O(N^2)$ to $O(N)$
- ScalFMM/StarPU
- FAST-LA with LBNL Stanford

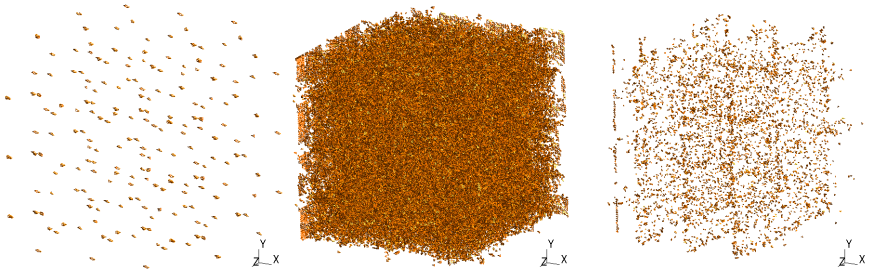


Parallel graph partitioning and remeshing

→ Parallel algorithms for graph mapping on distributed memory architectures
  - › Dynamic remapping
  - › Distributed algorithms
  - › Scotch
    - Widely used in academy and industry
    - Graphs with 2+ Billions vertices on 80k cores

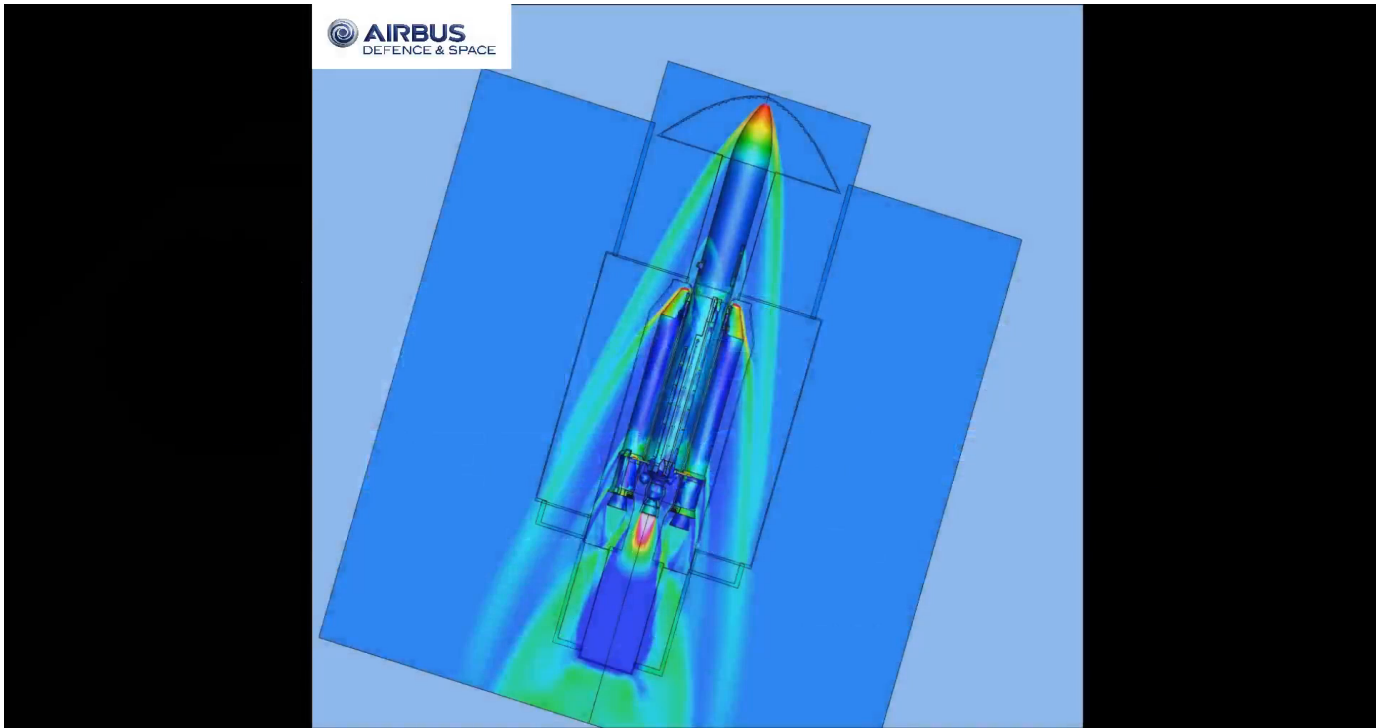


→ Remeshing unstructured meshes on distributed memory architectures
  - › PaMPA
    - Parallel mesh partitioning, data exchange, remeshing and redistribution
    - TurboMECA, ITER



Bringing Industrial Applications to the Exascale Era

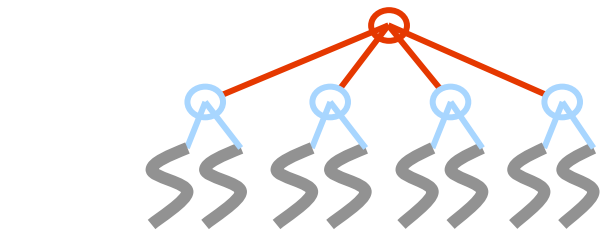
- → Aerodynamics of Ariane 6
  - › Propelling flow study (unsteady CFD)
  - › FLUSEPA: Airbus+HiePACS (+Runtime)
- → ITER
  - › Gysela: HiePACS+CEA
  - › 91% efficiency over 458,000 cores (BlueGene Q)

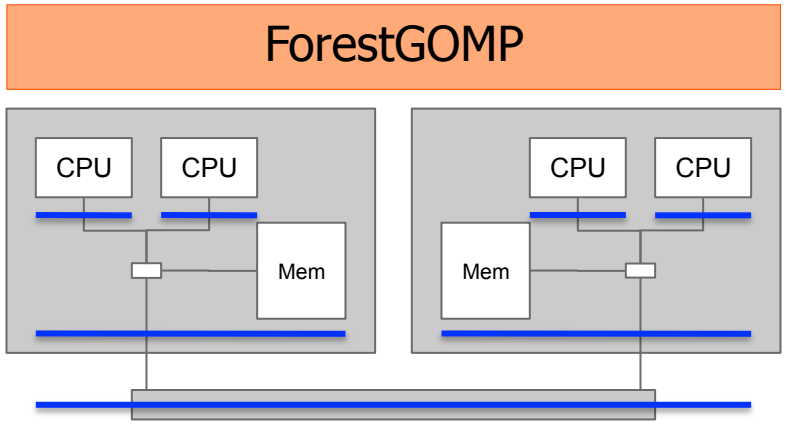


Code optimization and scheduling

- Performance Analysis and Tuning
  - Combine static binary performance model with dynamic behavior analysis
    - Identify performance bottleneck
    - Provide high-level feedback
  - MAQAO
    - With Exascale Computing Lab. [Intel]
    - Integration in TAU [Oregon, LANL], Score-P
    - Samsung
- Multicore-aware OpenMP
  - Capture application's structure with nested trees
    - Scheduling = mapping trees of threads onto a tree of cores
  - Improved Seismic simulations [BRGM]
  - Hwloc Library (Hardware Locality)
    - Open MPI, MPICH, Intel OpenMP, etc.

```
#pragma omp parallel for
  for (int i=0; i<MAX; i++)
      ...
#pragma omp parallel for
      for (int k=0; k<MAX; k++)
        ...
```

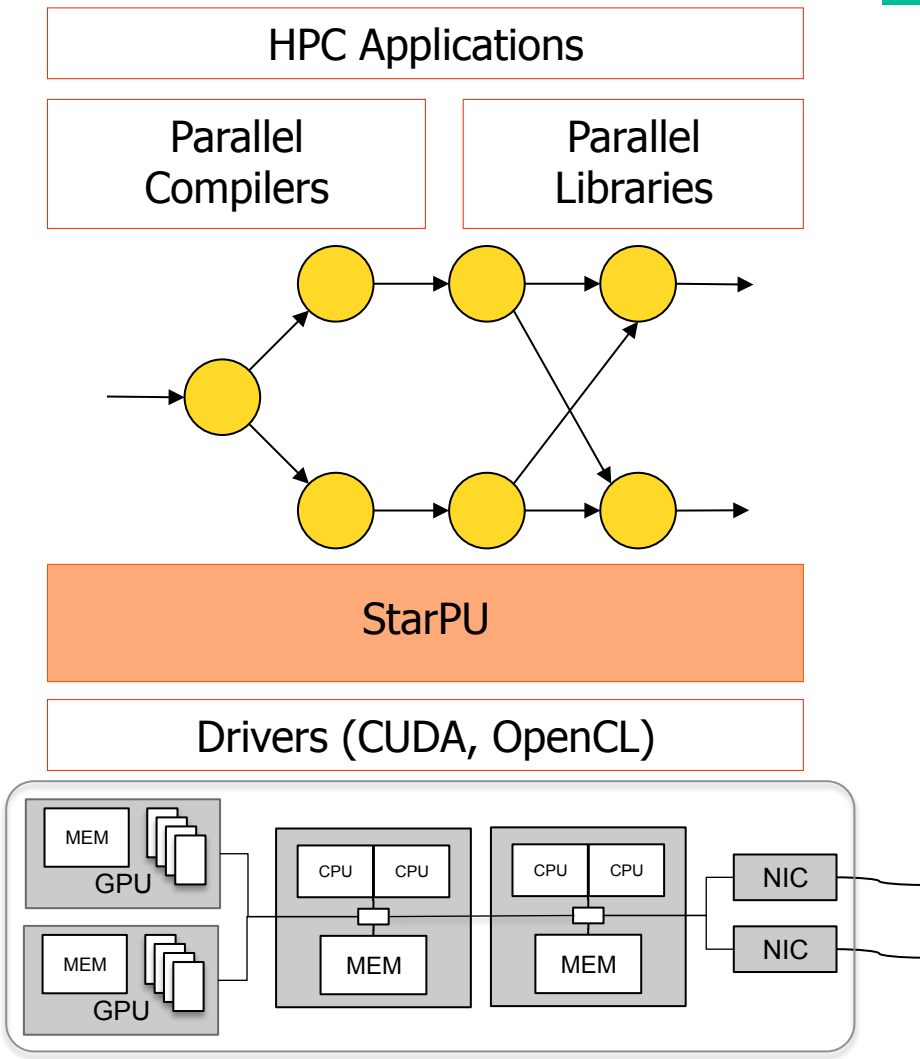




Exploiting Heterogeneous Architectures

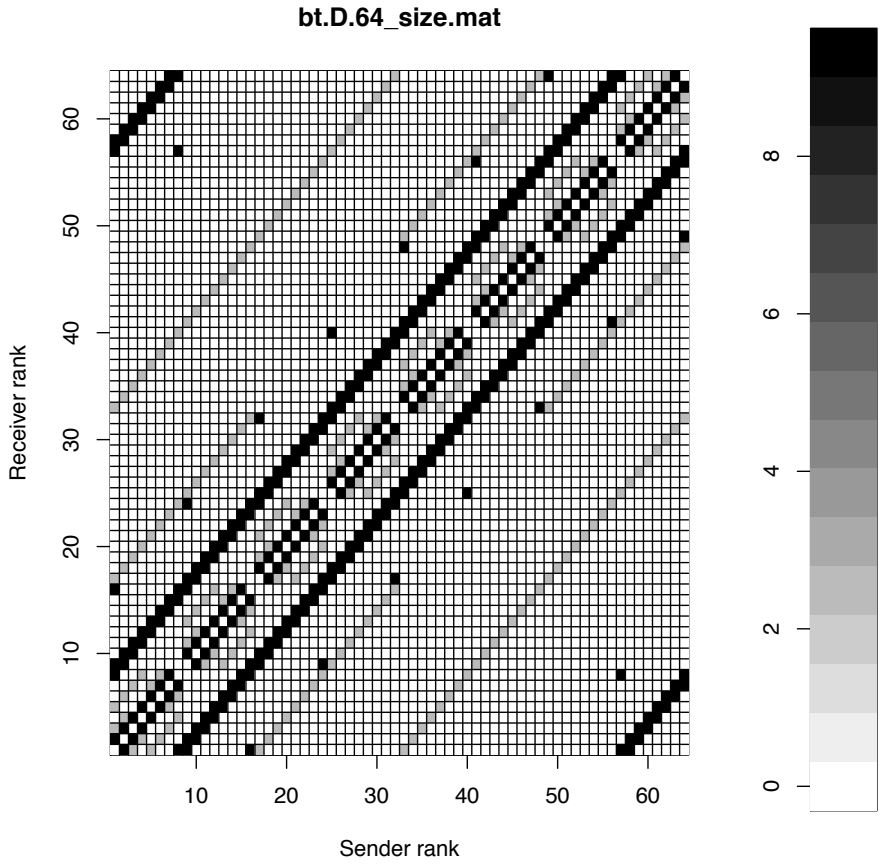
→ Scheduling tasks over heterogeneous machines
- › Impact:
  - Compilers
    - HMPP [CAPS], XMP
    - GCC
  - Libraries
    - MAGMA-MORSE [UTK, USA]
    - SkePU [Univ. Linköping]
- › StarPU Pioneered research about CPU+GPU runtime systems
  - [CCPE2011] cited 520+ times

→ Composability of parallel codes
- › Resource negotiation
- › Towards “code reusability” in HPC



Optimizing Communication over Clusters

- → Adapting inter-process communication to process locations
  - › KNEM: Direct-copy between processes
    - Integrated into MPICH2, Open MPI and MVAPICH2 (ANL, UTK)
- → Mapping communication patterns over hierarchical machines
  - › TreeMatch
    - Recursive algorithm to map (group of) processes onto topology
    - Used as process renumbering strategy for MPI_Dist_graph_create()
      - Open MPI, MPICH2

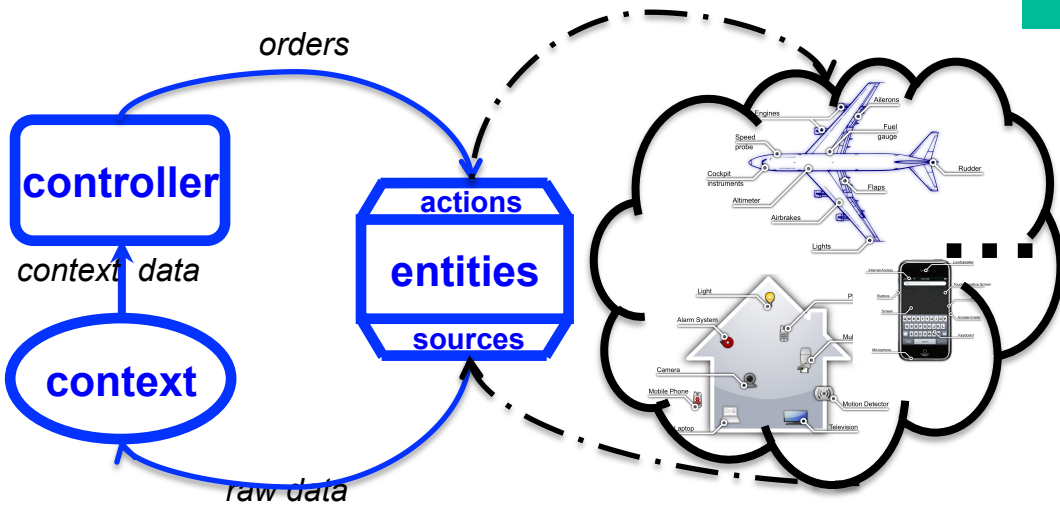


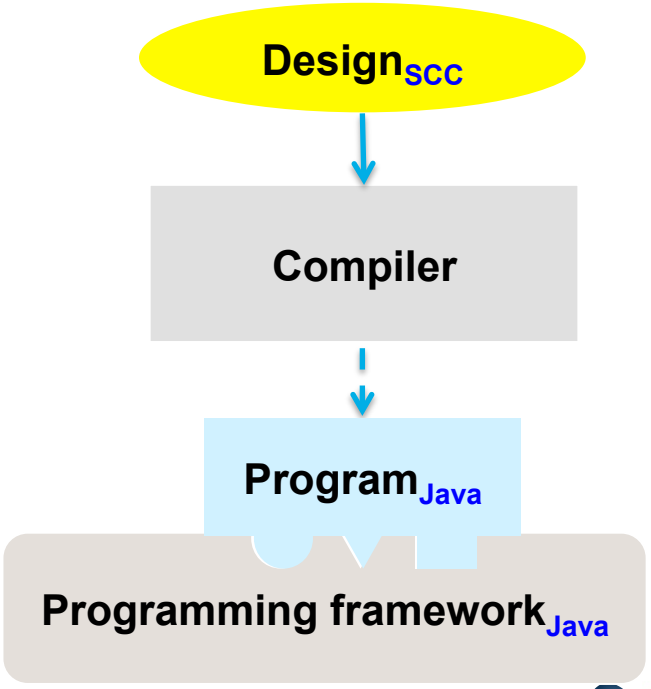
Orchestrating Networked Entities

→ Combining Domain-specific and General Purpose Languages
  - Sense-Compute-Control paradigm

→ DiaSuite
  - Designing – Declarations
  - Language – DiaSpec
  - Compilation – Java programming framework
  - Verification – Design time, compile time, run time
  - Programming – Java type
  - Support – Eclipse, APIs

  - Monitoring platform for older adults (installed in 24 houses)
    - Startup IQSpot (smart buildings)





LaBRI

Special Focus on

Linear Algebra Solvers over Heterogeneous Architectures

Emmanuel Agullo

université de BORDEAUX

LaBRI

Project

What is anticipated

→ SuperComputers expected to reach Exascale
- 10^{18} flop/s by 2020
- ETP4HPC, EESI, IESP, PRACE

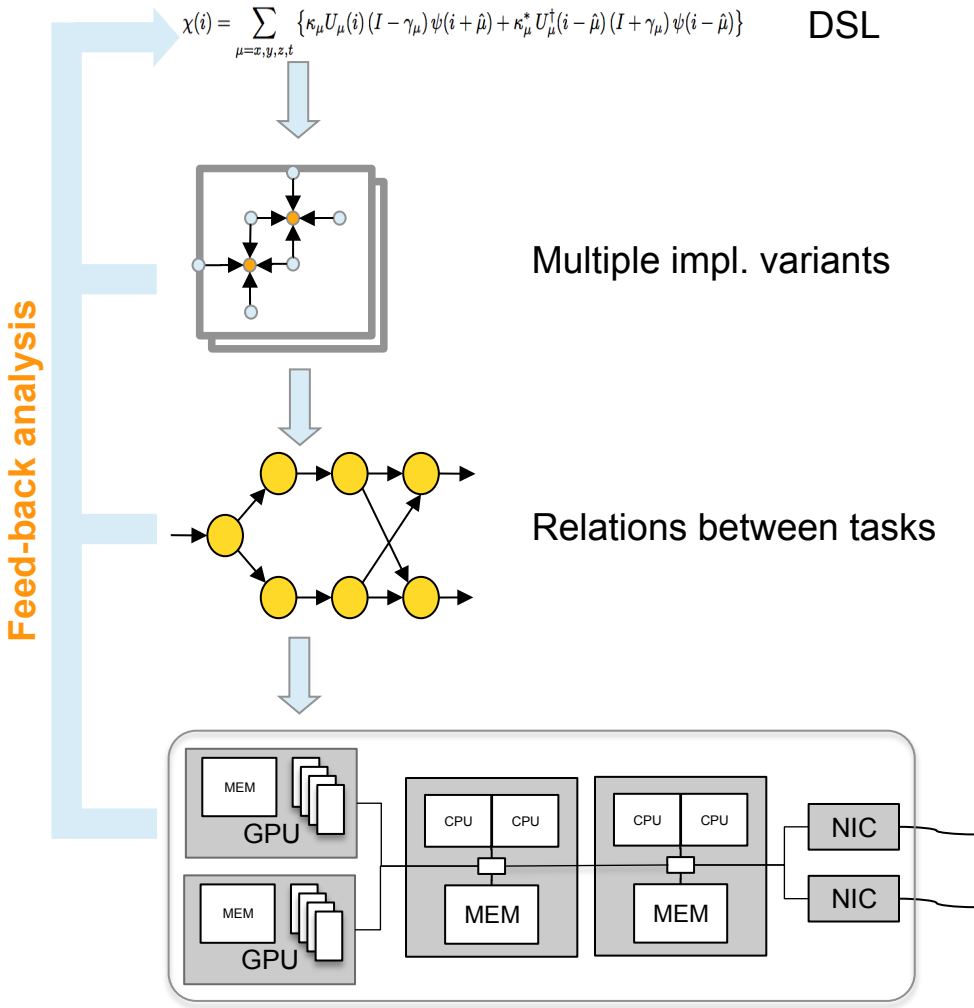
→ The biggest change will come from node architecture
- High number of cores
- Powerful SIMD units
- Hybrid systems

→ Extreme parallelism
- Total system concurrency is estimated to reach $O(10^9)$
- Sounds like embarrassingly parallel hardware!
  - Code coupling applications are welcome

→ Expensive data movements
- Both in terms of time and power

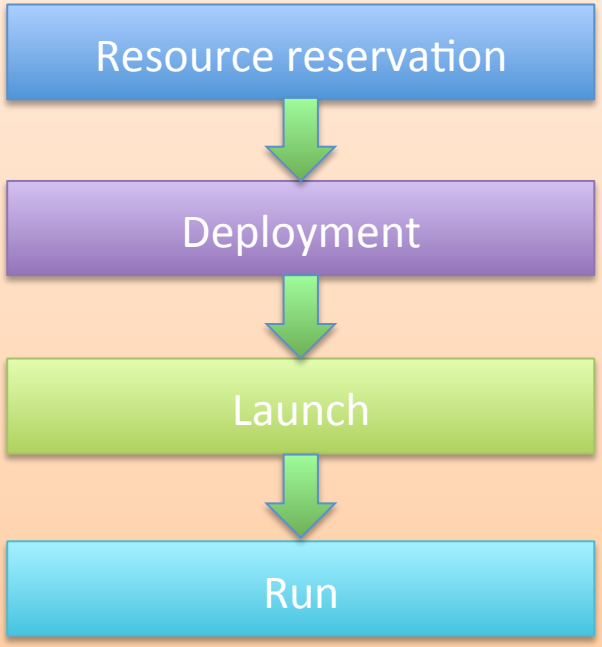
Languages and runtime systems

- → Pushing new programming models
  - › Exhibit maximum parallelism in architecture-agnostic manner
  - › Capture more semantics from high-level code
    - Math-style DSL
    - OpenCL extensions
- → Getting help from code generators
  - › Multiple variants for heterogeneous computing units (code, memory layout)
  - › Code analysis and hints for runtime systems
- → Designing better schedulers
  - › Adaptive Granularity (divisible tasks)
  - › Enable Parallel Code Composition (reusability)
    - Auto-dimensioning of parallel codes
- → Better Feedback to humans
  - › Diagnose problems like ”Your app is properly scheduled, but your kernels perform really bad”



Topology-aware data management

Application execution phases



Cross-topic research

Interaction with the ecosystem

Process placement

Data and graph partitioning
Process reordering

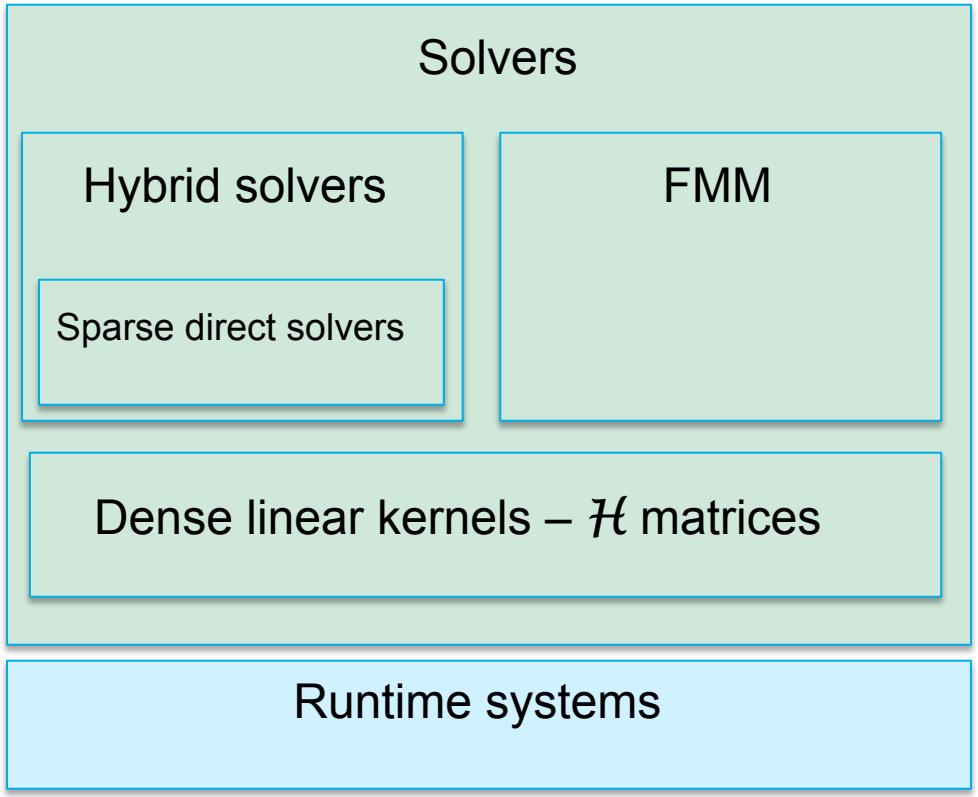
Affinity management
I/O and datapath optimization
Migration, Remeshing

Perf. and platfom models

Future Directions on Solvers

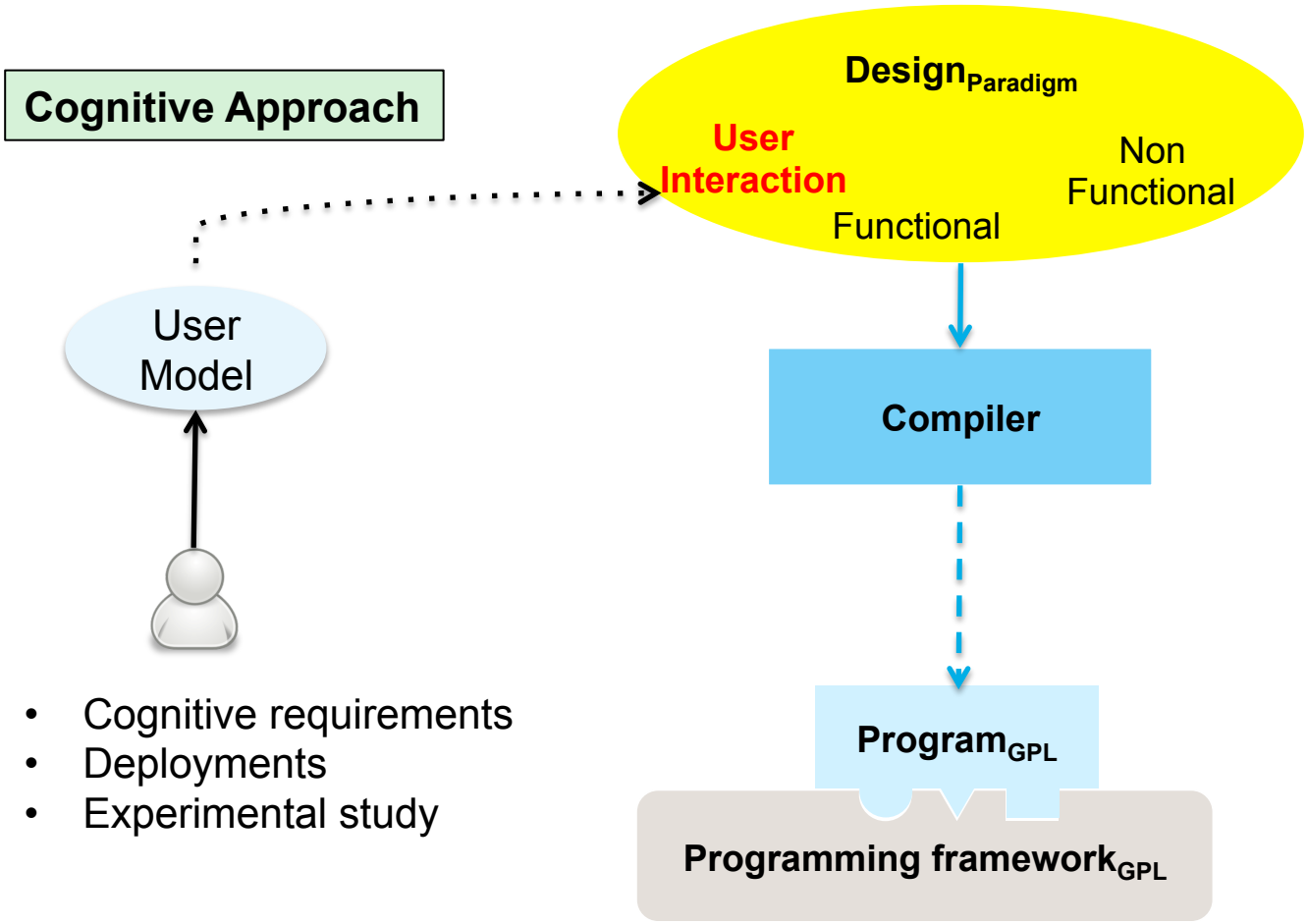
→ Bring whole solver stack to the Exascale

- Resilience
  - Naturally-resilient numerical approaches
- Scalable algorithms
- Complete software suite
  - Industrial consortium under construction



Expanding the Design-Driven Approach

→ Human-centric software development



- Cognitive requirements
- Deployments
- Experimental study

New SATANAS

→ 3 topics

- Algorithms and applications
  - HiePACS
- Runtime systems
  - TADAAM
  - STORM
- Orchestration of Networked Entities
  - Phoenix