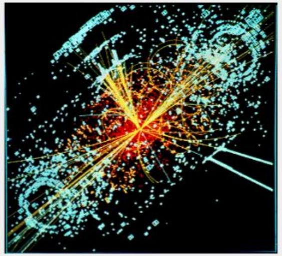
Towards Exascale Across Scales!

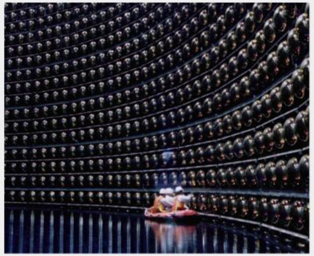
Shantenu Jha

Rutgers Advanced DIstributed Cyberinfrastructure & Applications Laboratory (RADICAL)
http://radical.rutgers.edu

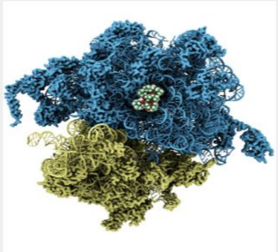
“Big Science” to the Long Tail of Science



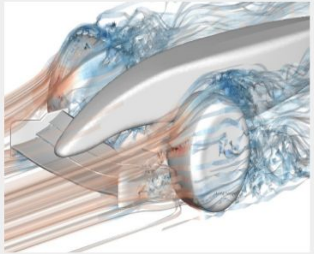
The ATLAS experiment at the Large Hadron Collider in Switzerland uses SAGA in conjunction with PanDA as a workload management system.



The Super-Kamiokande project searches for neutrinos to understand the creation of matter in the universe. It uses SAGA to simulate collisions on HPC clusters.

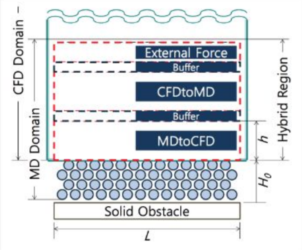


RADICAL-Pilot is being used by Chemistry researchers to support large-scale and multidimensional replica exchange simulations on supercomputers.



Nektar++ is a finite element package which uses SAGA in the backend to submit jobs to a variety of clusters. It tackles problems such as modeling air flow around automobiles.

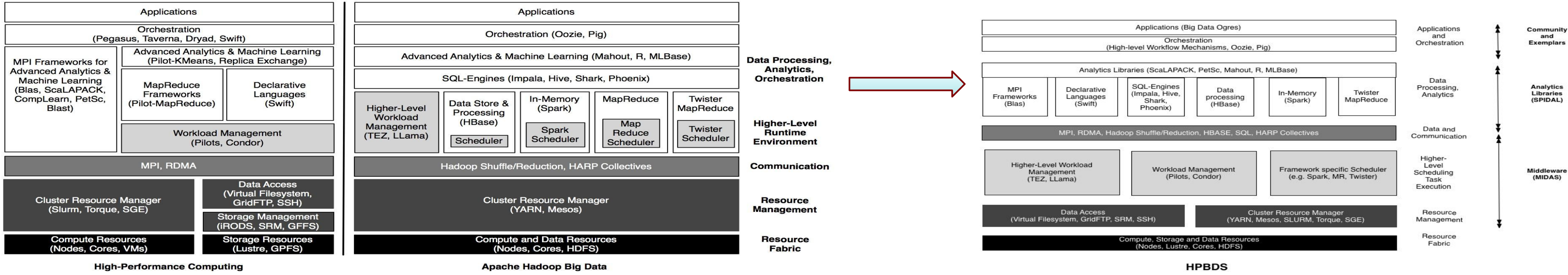
Researchers at UCL London are using RADICAL-Pilot to advance understanding of HIV drug resistance and make personalized treatment possible.



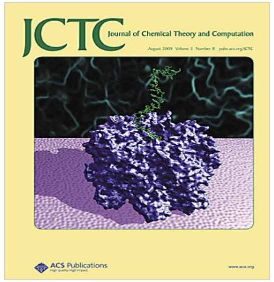
RADICAL-Pilot supports multi-physics and coupled simulations, such as hybrid CFD-MD simulations to understand Couette Flow, as well as PBM-DEM simulations for Cybermanufacturing.

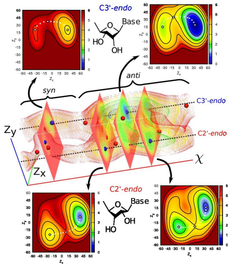
Convergence of HPC and “Data Intensive” Computing:

- Supercomputers were (historically) net producers of data, not consumers
- Convergence at multiple levels, including Software Environment
  - HP-ABDS: Integration of High Performance with Advanced Functionality
  - SPIDAL and MIDAS (http://spidal.org)

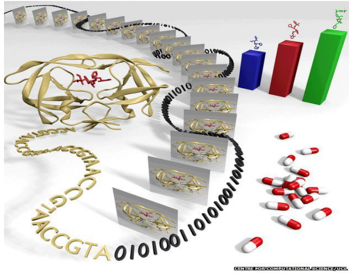


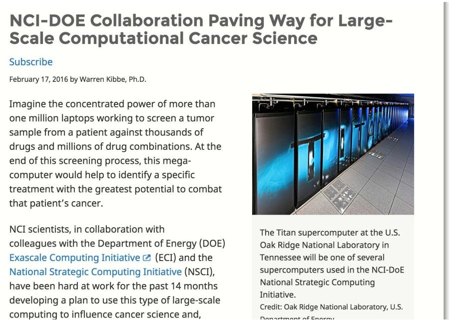
A Tale of Two Data-Intensive Paradigms:
Data Intensive Applications, Abstractions and Architectures
Jha, Qiu, Fox
http://arxiv.org/abs/1403.1528





Case Study: Biomolecular Sciences





NCI-DOE Collaboration Paving Way for Large-Scale Computational Cancer Science

Subscribe

February 17, 2016 by Warren Kibbe, Ph.D.

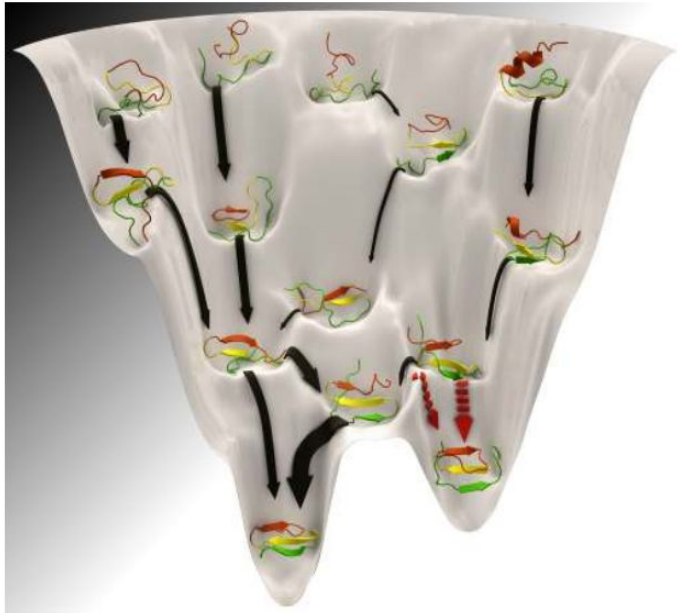
Imagine the concentrated power of more than one million laptops working to screen a tumor sample from a patient against thousands of drugs and millions of drug combinations. At the end of this screening process, this mega-computer would help to identify a specific treatment with the greatest potential to combat that patient's cancer.

NCI scientists, in collaboration with colleagues with the Department of Energy (DOE) Exascale Computing Initiative (ECI) and the National Strategic Computing Initiative (NSCI), have been hard at work for the past 14 months developing a plan to use this type of large-scale computing to influence cancer science and,

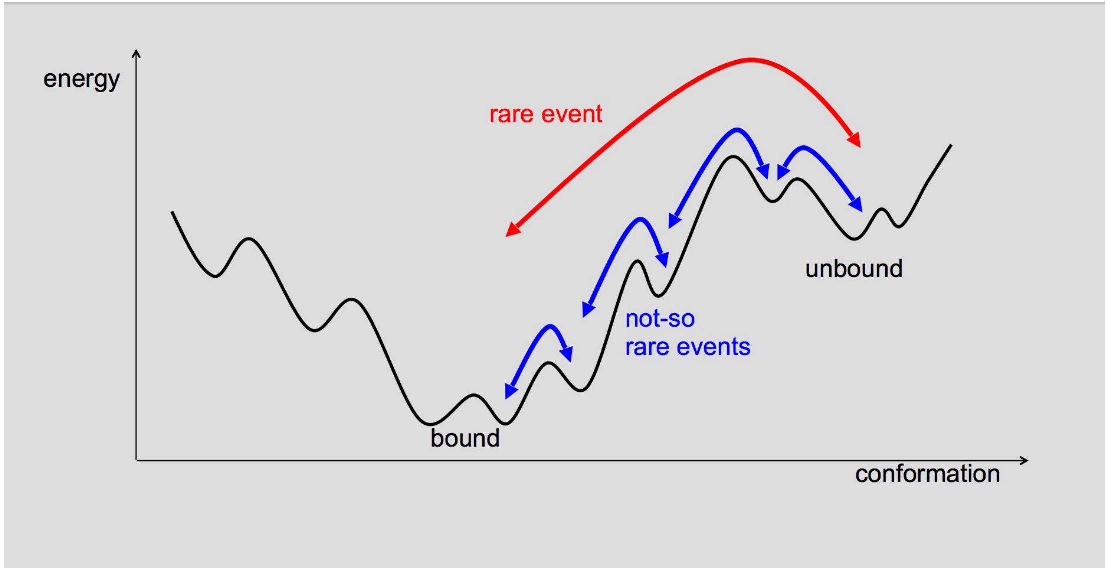
The Titan supercomputer at the U.S. Oak Ridge National Laboratory in Tennessee will be one of several supercomputers used in the NCI-DoE National Strategic Computing Initiative.

Credit: Oak Ridge National Laboratory, U.S.

Protein folding mechanisms



Noé et al, **PNAS** (2009)



A Schism in Biomolecular Simulations?

- *Given a finite amount of computing which is better:*
    - *Many simulations or Longer simulations?*

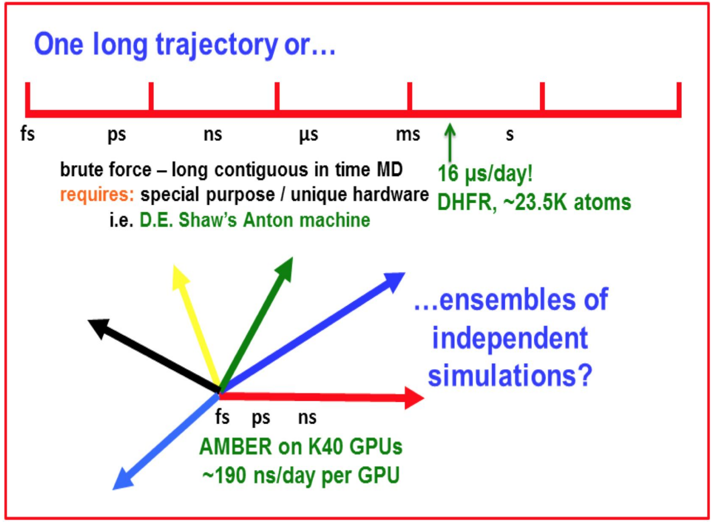
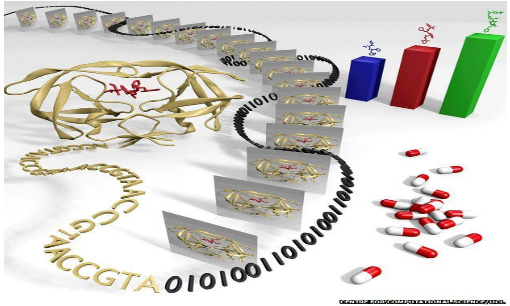
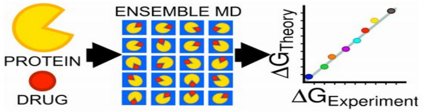
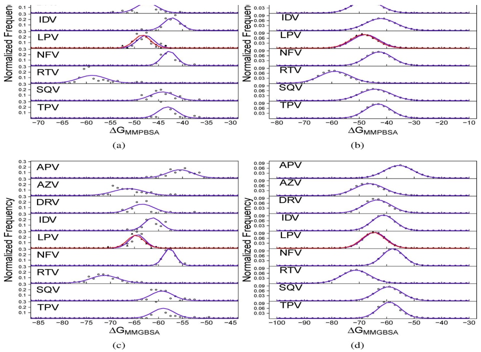


Figure 2: Schematic of the MD simulation time scale comparing long MD simulation on a special purpose machine like Anton to multiple independent MD runs on accelerators.

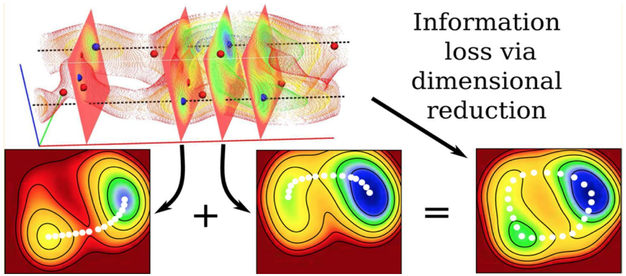






Landscape of Biomolecular Simulations

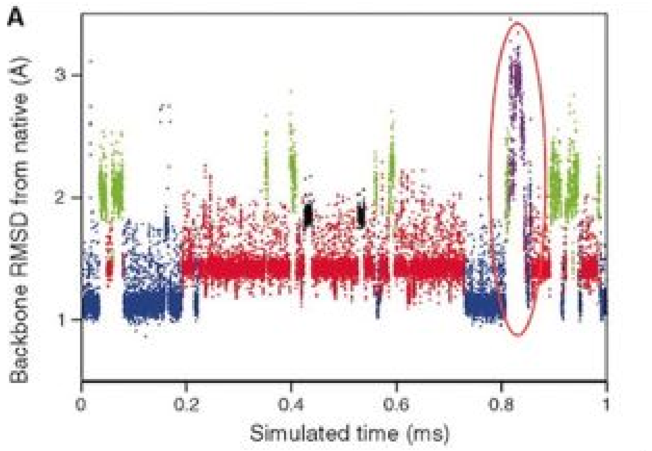
- Larger biological systems
  - Weak scaling
  - Status Quo: Size of systems: > 10M atoms
- Long time scale problem
  - Strong scaling
  - Status Quo: Duration of systems: > 10 ms
- Scaling challenges > than either single-partition strong and weak scaling.
  - Accurate estimation of complex physical processes, e.g., M-REMD
- **Gap between weak scaling and strong scaling** capabilities will grow.

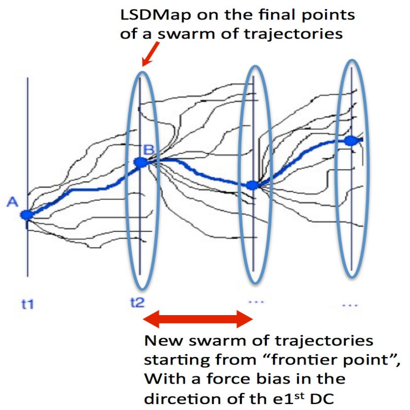


Multidimensional replica exchange umbrella sampling (REUS) simulations of a single uracil ribonucleoside.

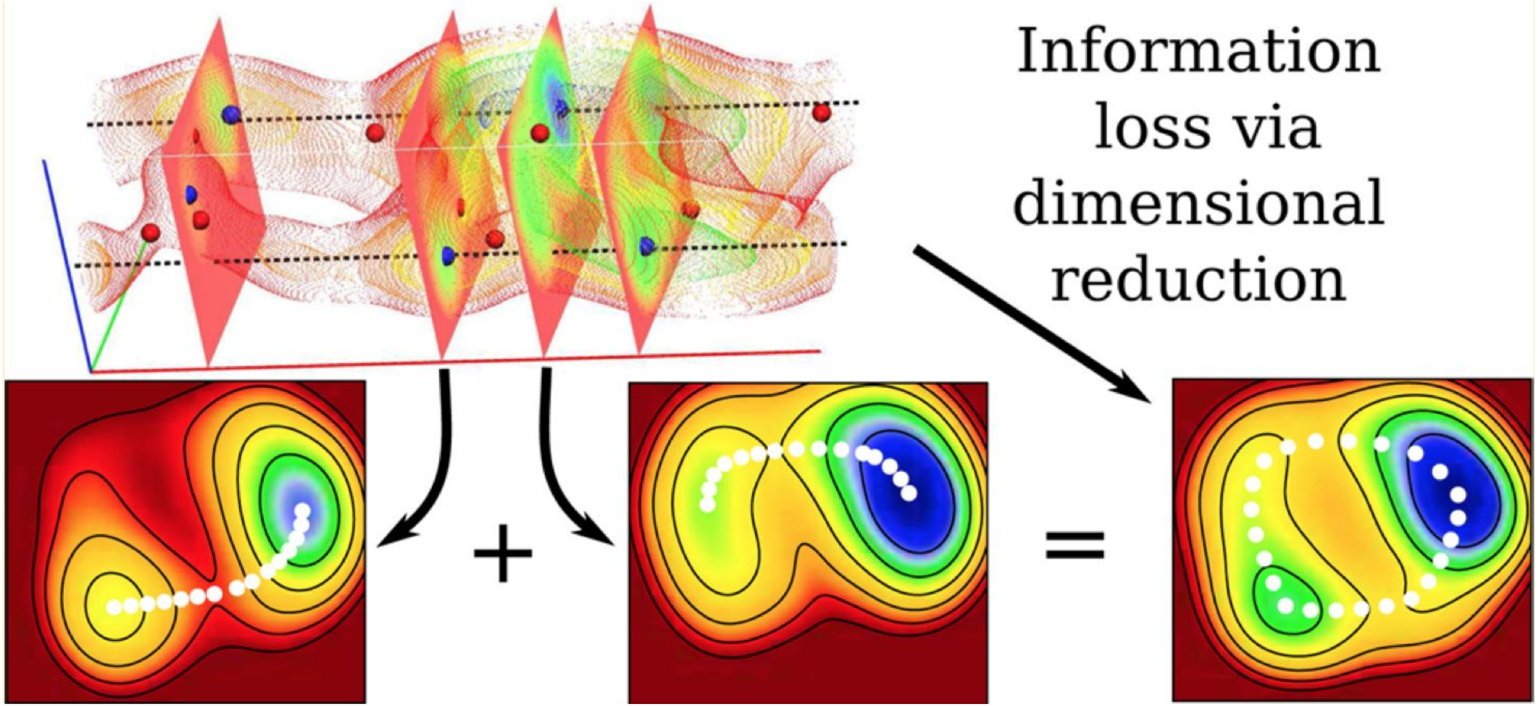
Brief Introduction to Sampling

- Sampling: BPTI, 1ms MD ~3 months on Anton (Shaw *et al*, Science 2010).
    - *More* sampling
    - *Better* sampling
    - *Faster* sampling
- ***More sampling:*** Hundreds or thousands of concurrent MD jobs
- ***Better Sampling:*** Drive systems towards unexplored regions, don't waste time sampling behaviour already observed
    - E.g. DM-d-MD, AMBER-COCO





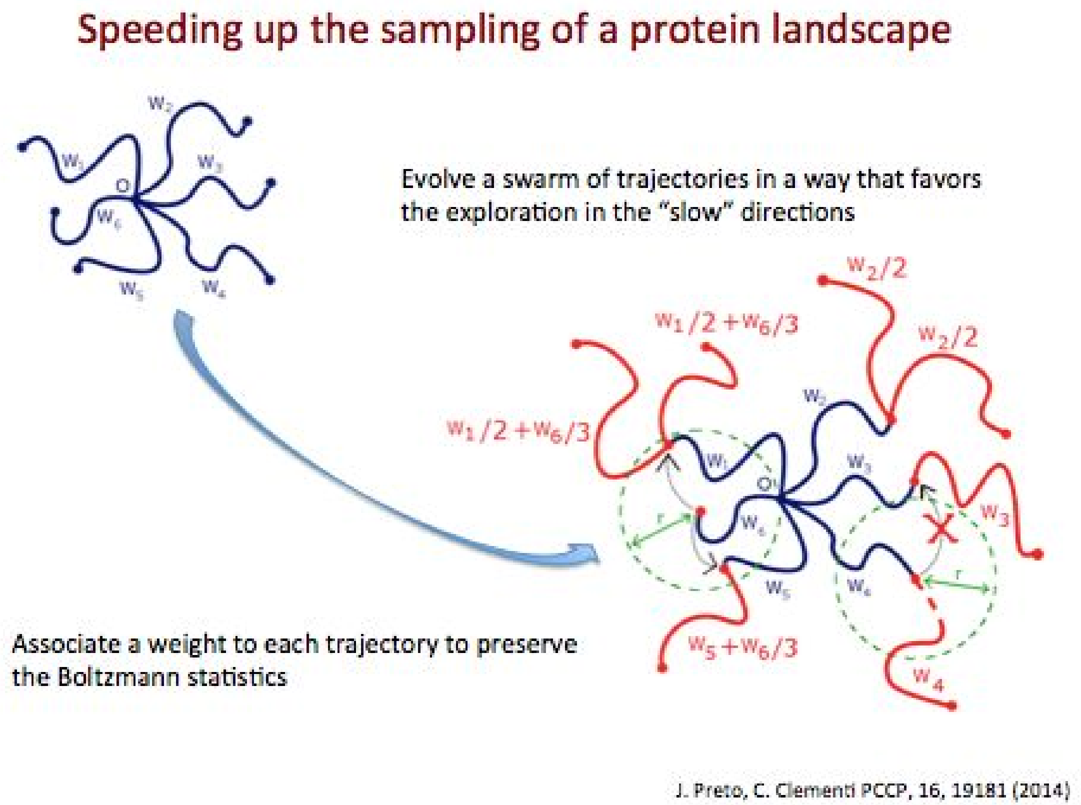
Multi-dimensional Replica-Exchange



When the number of replicas cannot > number of nodes/cores, 1D replica exchange is the “default” (only!) option

DM-D-MD: Diffusion Map Driven Molecular Dynamics

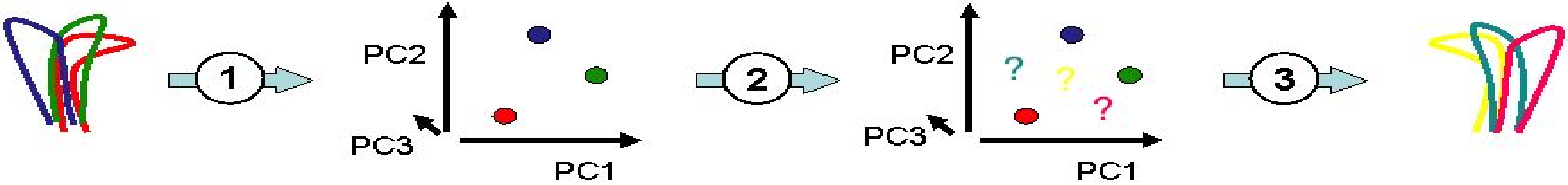
(Courtesy: Ceclia Clementi, Rice)



PROTEINS
STRUCTURE ■ FUNCTION ■ BIOINFORMATICS

COCO: A simple tool to enrich the representation of conformational variability in NMR structures

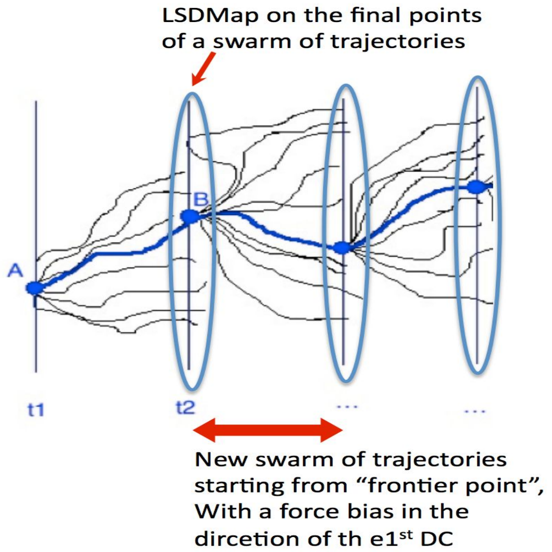
Charles A. Laughton,[1*] Modesto Orozco,[2,3,4] and Wim Vranken[5]



Proteins 2009; 75:206–216.

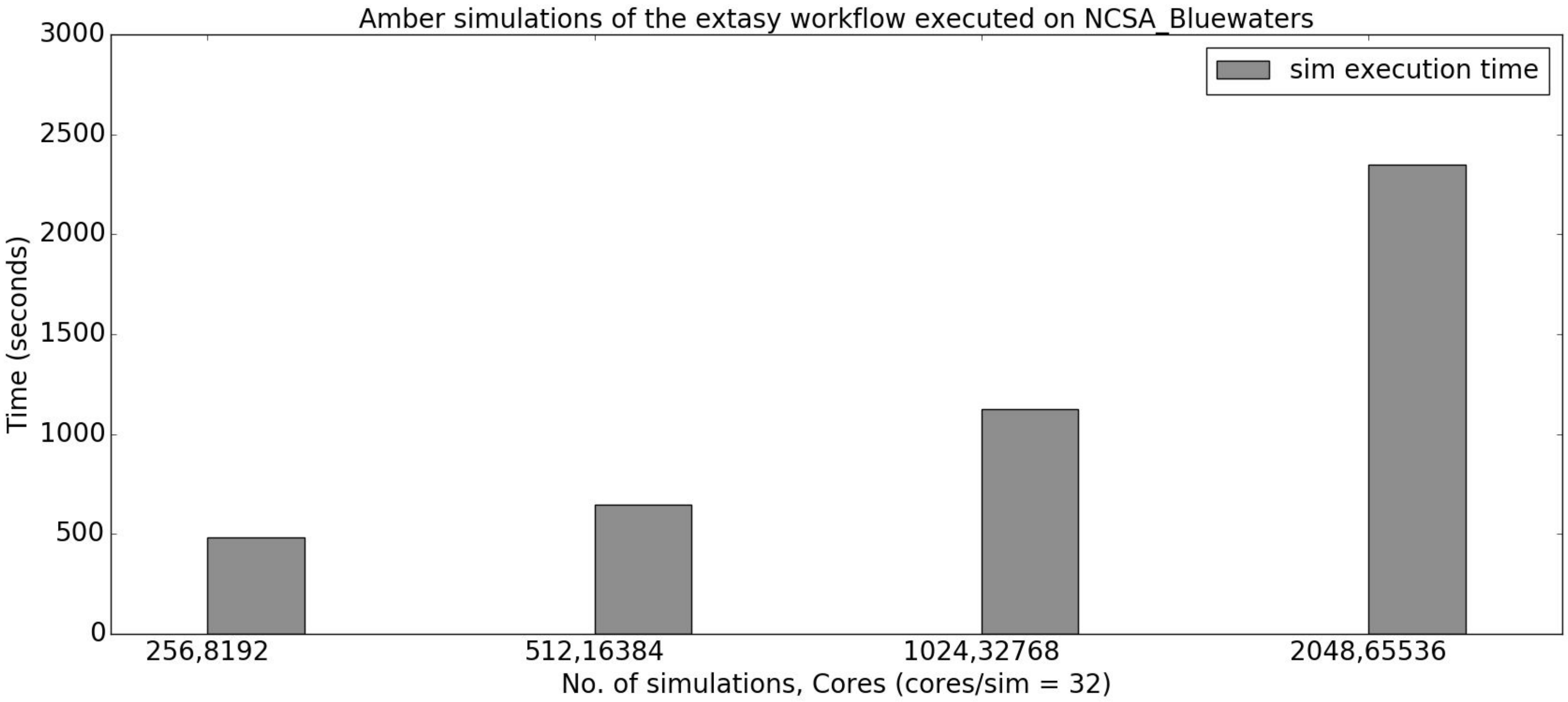
Advanced Sampling

- ***Better Sampling:*** Drive systems towards unexplored regions, don't waste time sampling behaviour already observed
- Iteratively run "analysis" and "sampling" phase
    - **Sampling phase:** multitude of trajectories are run in parallel
    - **Analysis phase:** Information gathered by the trajectories is analyzed and used to restart new trajectories to explore new regions of the configurational space.

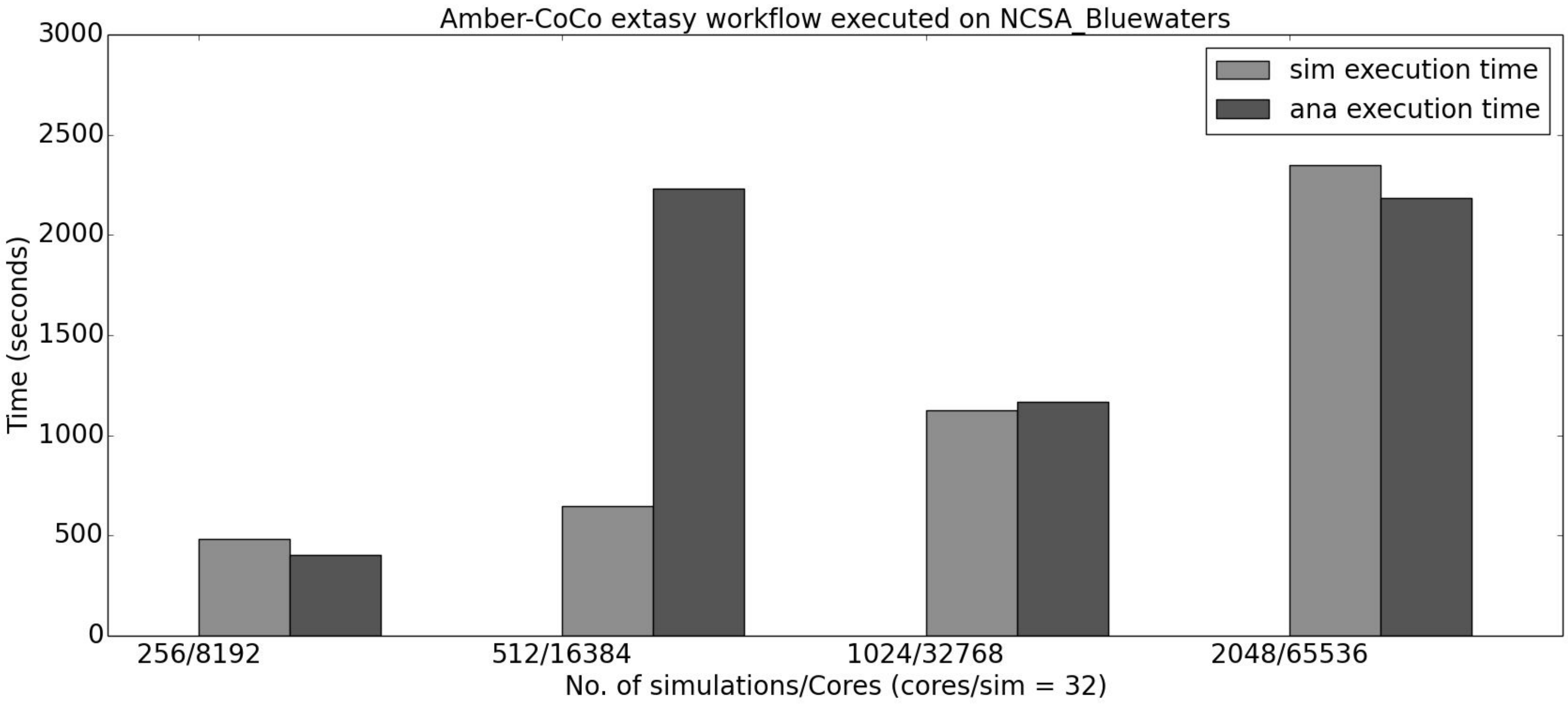


Diffusion Map driven Moleculad Dynamics (DM-d-MD), uses dimensionality reduction method of "Diffusion map" to extract a good reaction coordinate and use it to redistribute a large set of trajectories in the sampling of a complex configurational space.

Weak Scaling

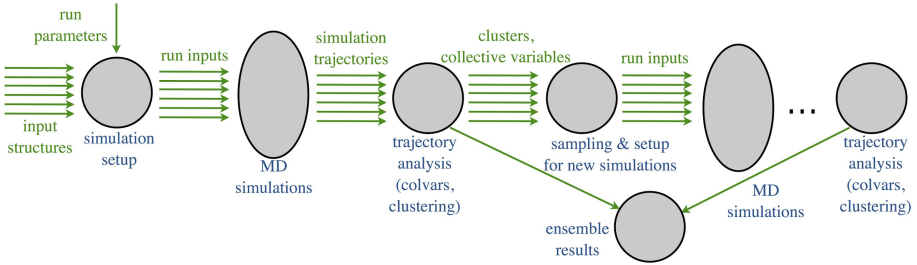


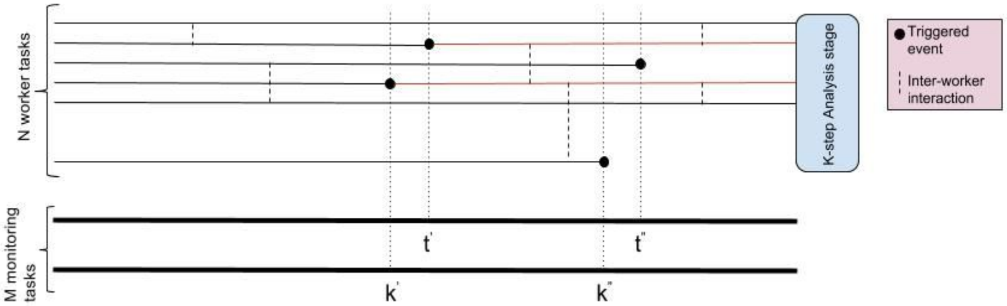
Weak Scaling: Simulation and Analysis



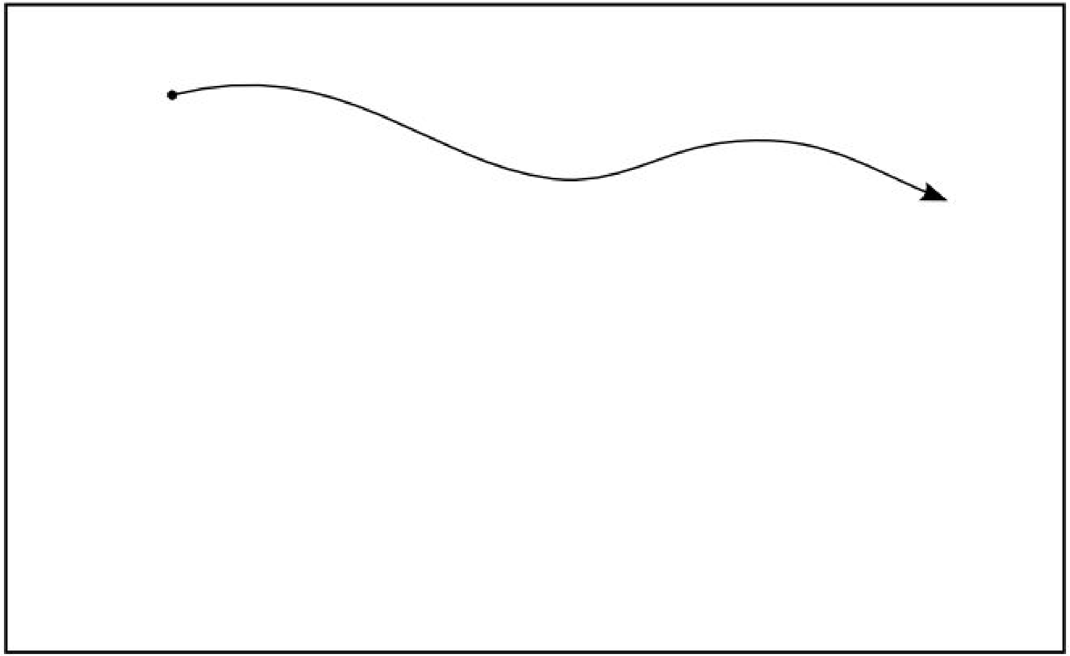
Adaptive and Steered Patterns

- However many applications involve adaptive execution and steering.
- Examples of **simulation algorithms**:
  - Commingle replica exchange simulation with a coarse-grained potential
  - Steer ensemble simulations based on intermediate analyses
  - Add more ensemble members...
- A framework that expresses different **simulation algorithms** as “adaptive execution patterns”. **How ?**
  - Generalise static patterns EnTK
  - Opens many research questions

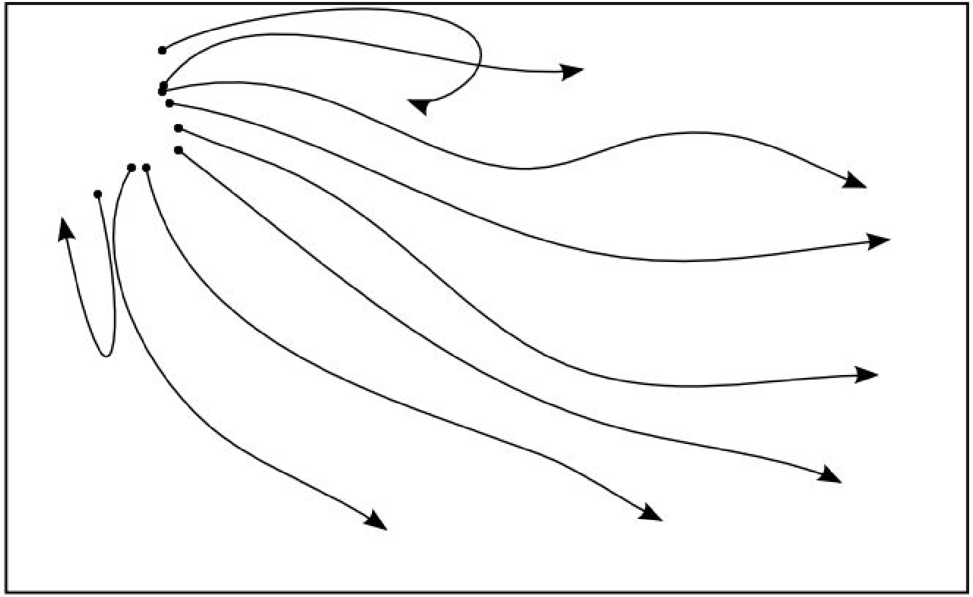




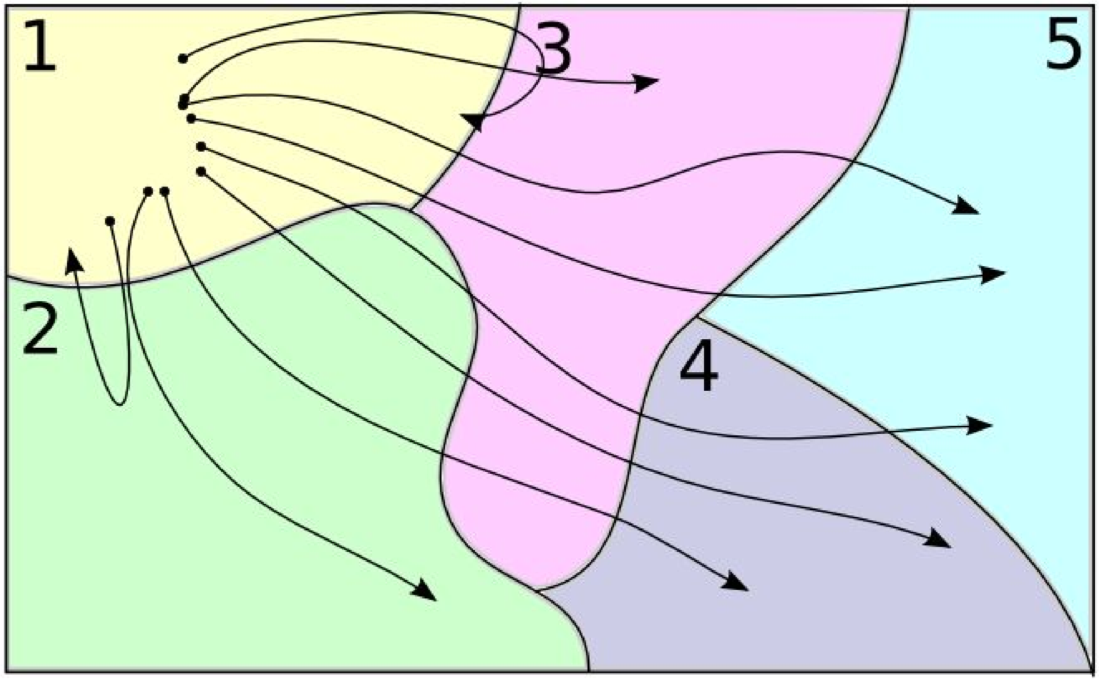
MSM: ML-driven Sampling



MSM: ML-driven Sampling

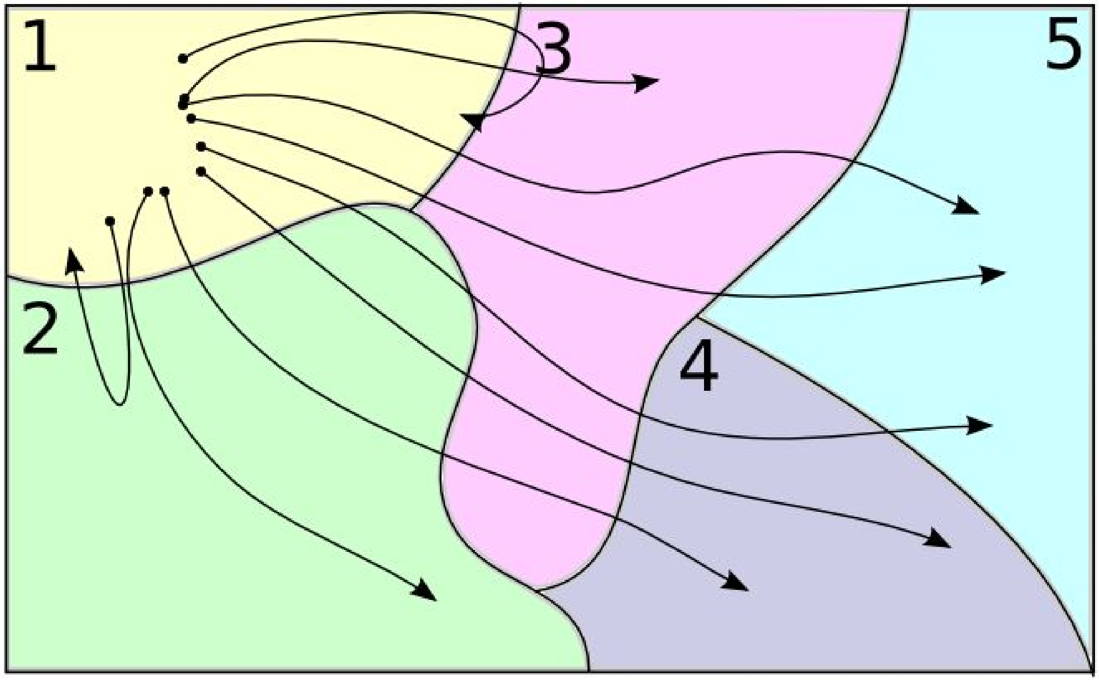


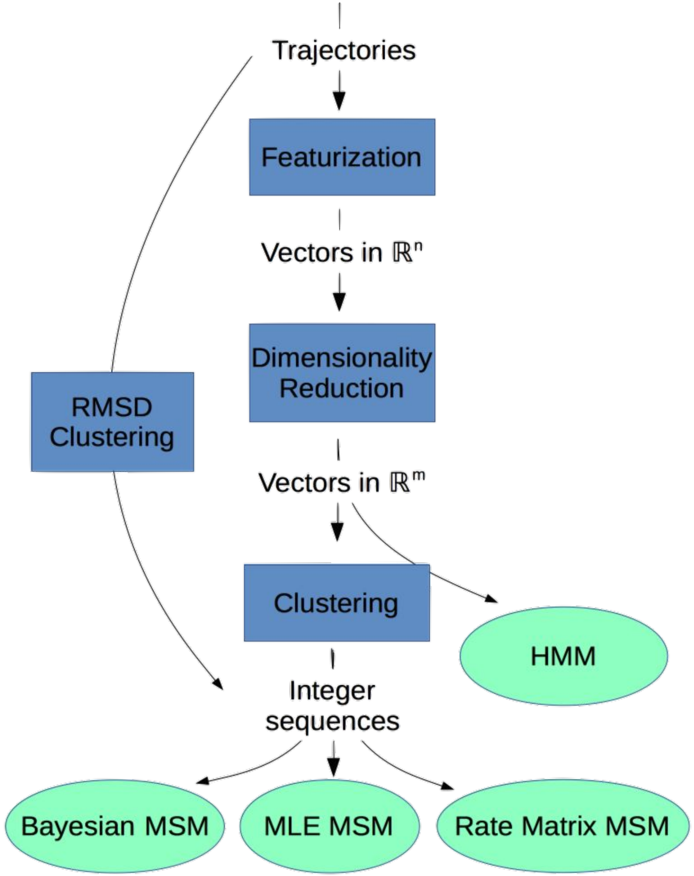
MSM: ML-driven Sampling



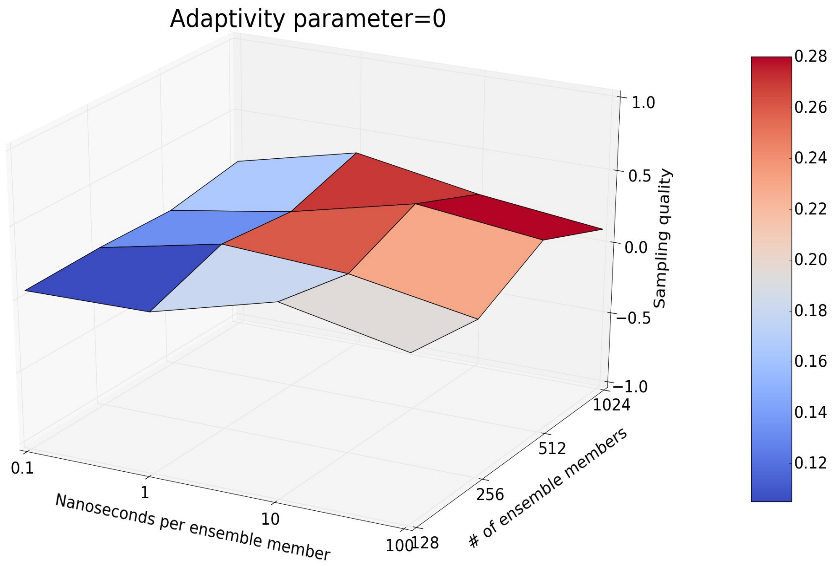
Credit: Kyle Beauchamp

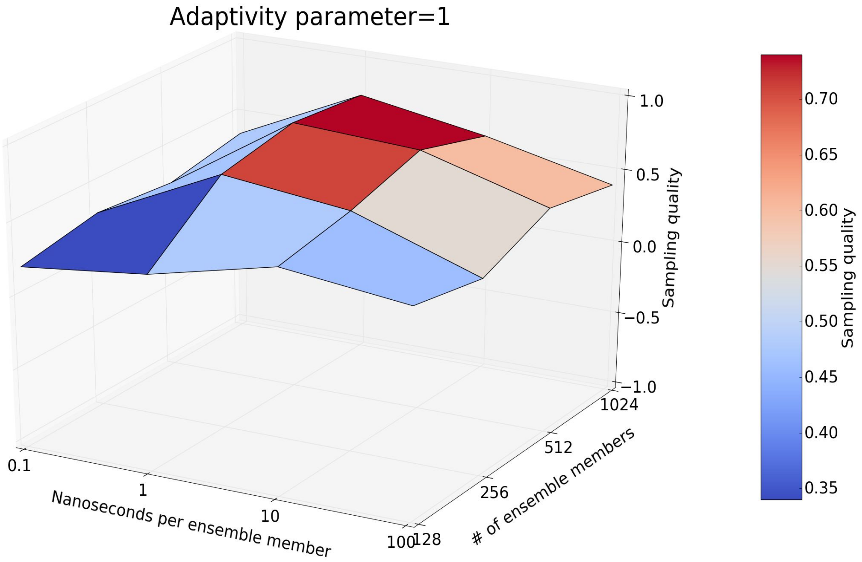
MSM: ML-driven Sampling





Better Sampling -- Requires Learning "on the fly"

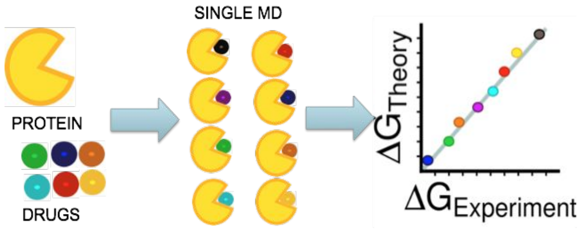


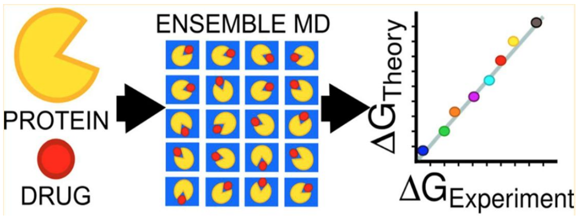


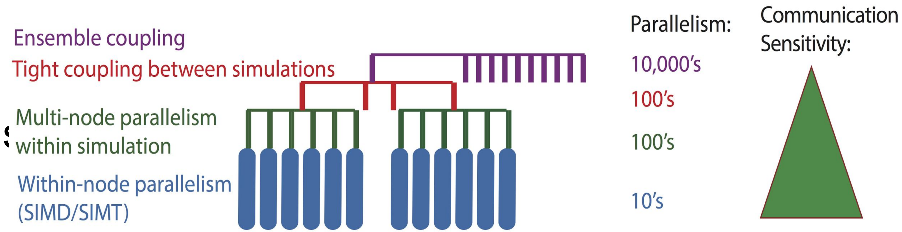
Finding the optimal resource configuration.

The Power of Many: RADICAL-Ensemble Toolkit

- Support for **heterogeneous** tasks
  - Multi-node and sub-node, application kernels, MPI/non-MPI
- Adaptive: Workload and resource: tasks and/or relations between tasks unknown *a priori*
- Range of concurrency and coupling of tasks
  - Multiple-levels and degree
- Multiple dimensions of scalability:
  - Concurrency: O(100K)-O(1,000K) tasks
  - Task size: O(1) - O(1,000) cores
  - Launch: O(100+) tasks per second
  - Task duration: O(1) - O(10,000) seconds
  - ….

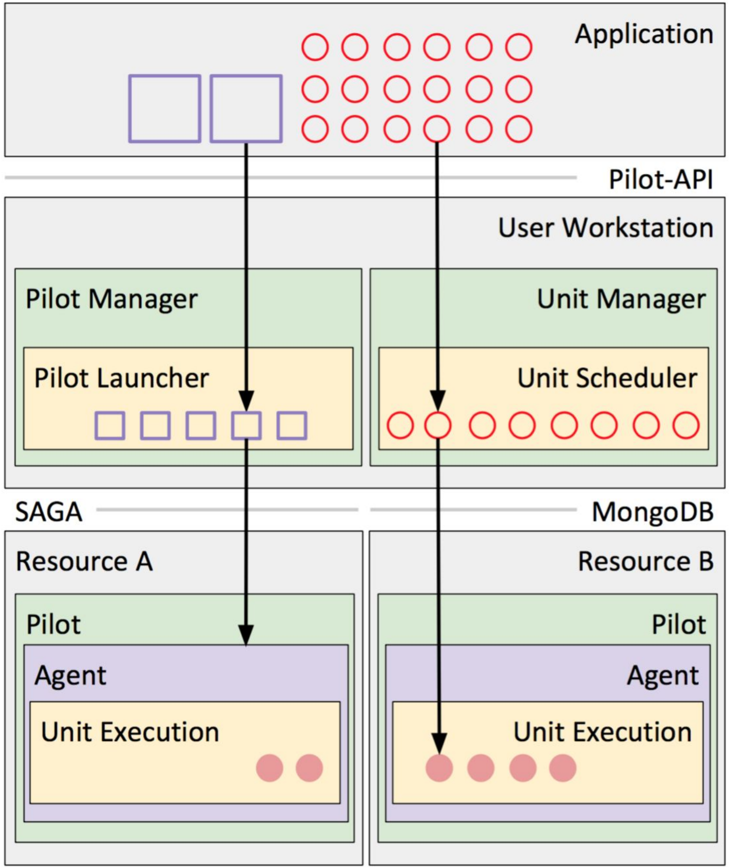




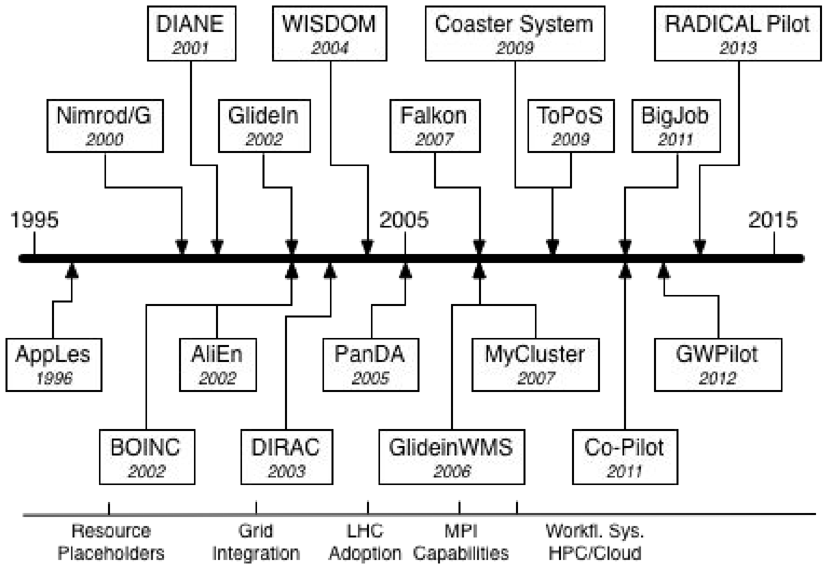


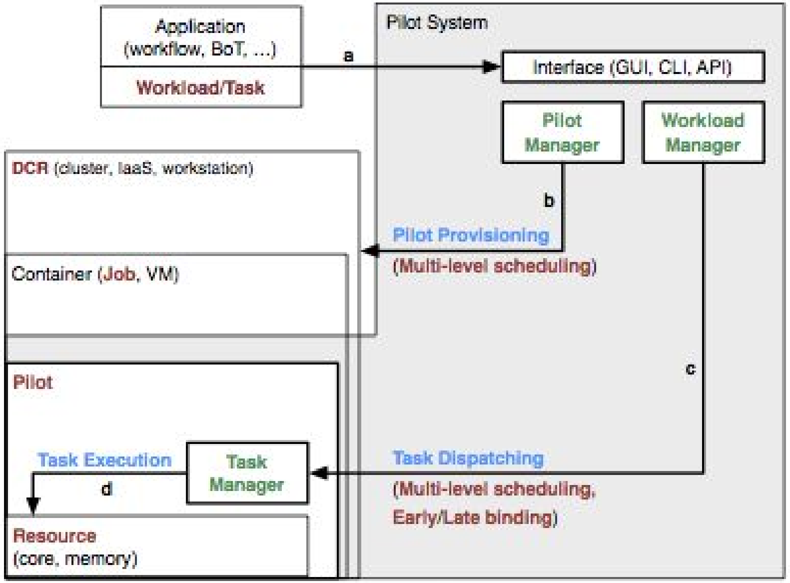
RADICAL-Pilot Overview

- **Programmable interface (arguably unique)**
  - Defined state models for pilots and units.
- **Supports research whilst supporting production scalable science:**
  - Agent, communication, throughput.
  - Pluggable components; introspection.
- **Portability and Interoperability:**
  - SAGA (batch-queue system interface)
  - Modular pilot agent for diff. architectures
  - Works on Crays, XSEDE resources, most clusters, OSG, Amazon EC2...



Pilot Jobs: Many Variations on a Theme



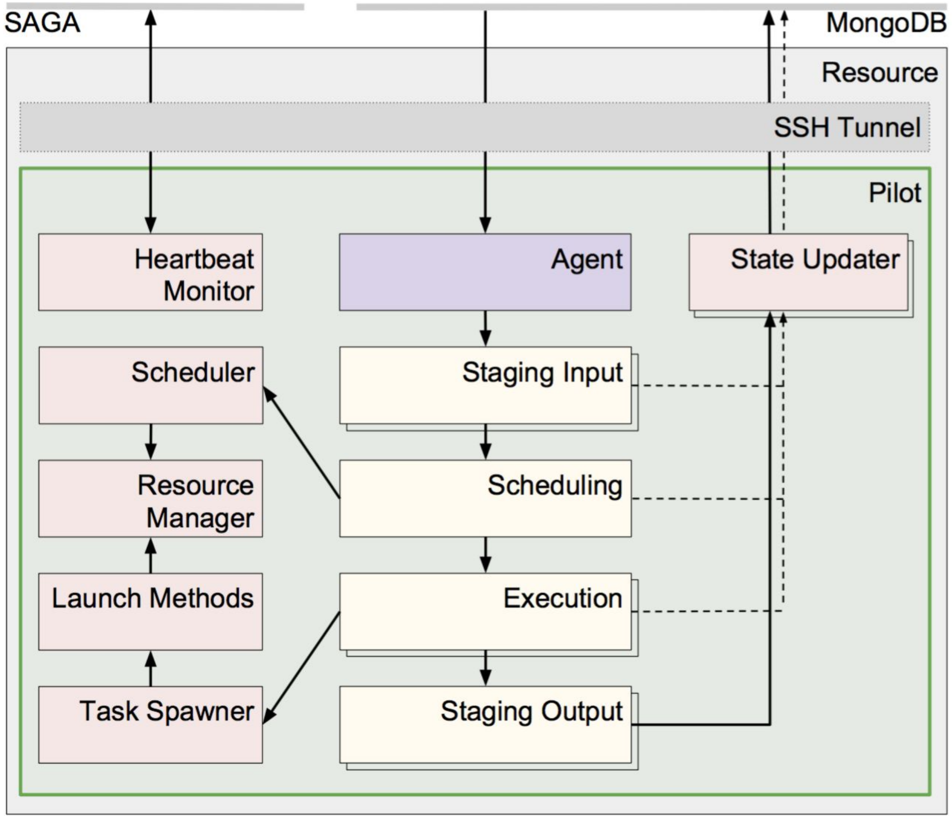


"Perfection is achieved, not when there is nothing more to add, but when there is nothing left to take away."
- Antoine Saint-Exupéry

- "P*: A Model of Pilot-Abstractions", *8th IEEE International Conference on e-Science (2012)*
- *A Comprehensive Perspective on Pilot-Jobs* http://arxiv.org/abs/1508.04180 (2015)

Agent Architecture

- **Components:** Enact state transitions for Units
- **State Updater:** Communicate with client library and DB
- **Scheduler:** Maps Units onto compute nodes
- **Resource Manager:** Interfaces with batch queuing system, e.g. PBS, SLURM, etc.
- **Launch Methods:** Constructs command line, e.g. APRUN, SSH, ORTE, MPIRUN
- **Task Spawner:** Executes tasks on compute nodes



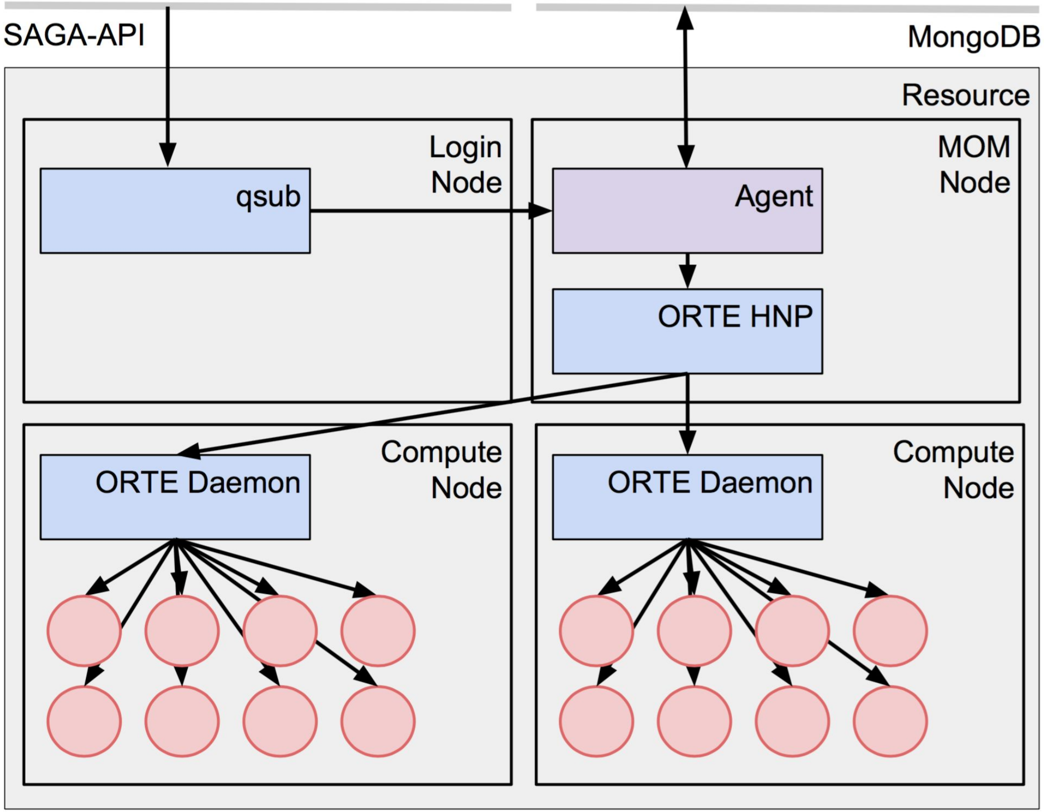
RADICAL-Pilot: ORTE

- ORTE: **O**pen **R**un**T**ime **E**nvironment
  - Isolated layer used by Open MPI to coordinate task layout
  - Runs a set of daemons over compute nodes
  - No ALPS concurrency limits
  - Supports multiple tasks per node

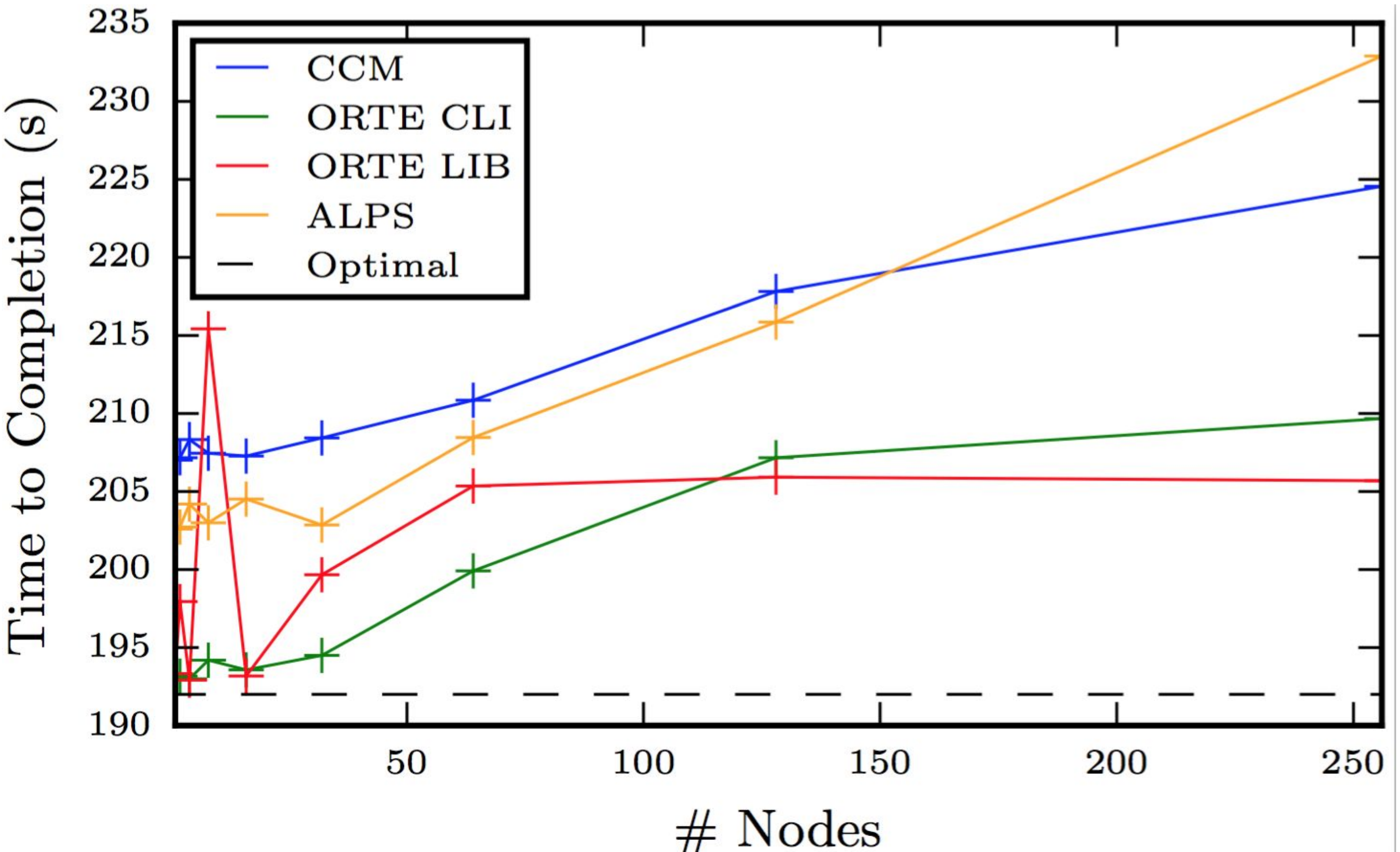
- orte-submit is CLI which submits tasks to those daemons
  - ‘sub-agent’ on compute node that executes these
  - Limited by fork/exec behavior
  - Limited by open sockets/file descriptors
  - Limited by file system interactions

RADICAL-Pilot + ORTE-LIB

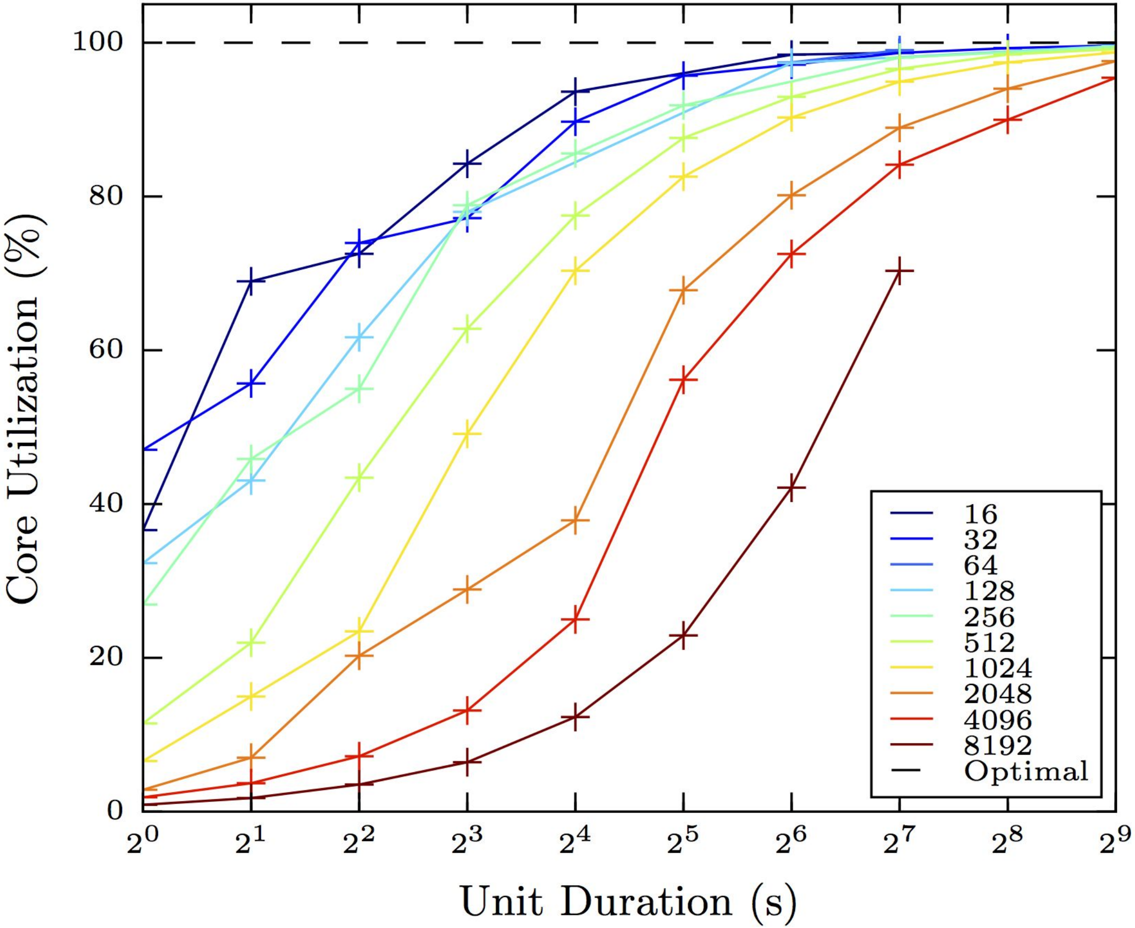
- All the same as ORTE-CLI, but
  - Uses library calls instead of `orterun` processes
  - No central fork/exec limits
  - Shared network socket
  - (Hardly) no central file system interactions



Agent Performance: Full Node Tasks (3xN, 64s)



Agent Performance: Resource Utilization

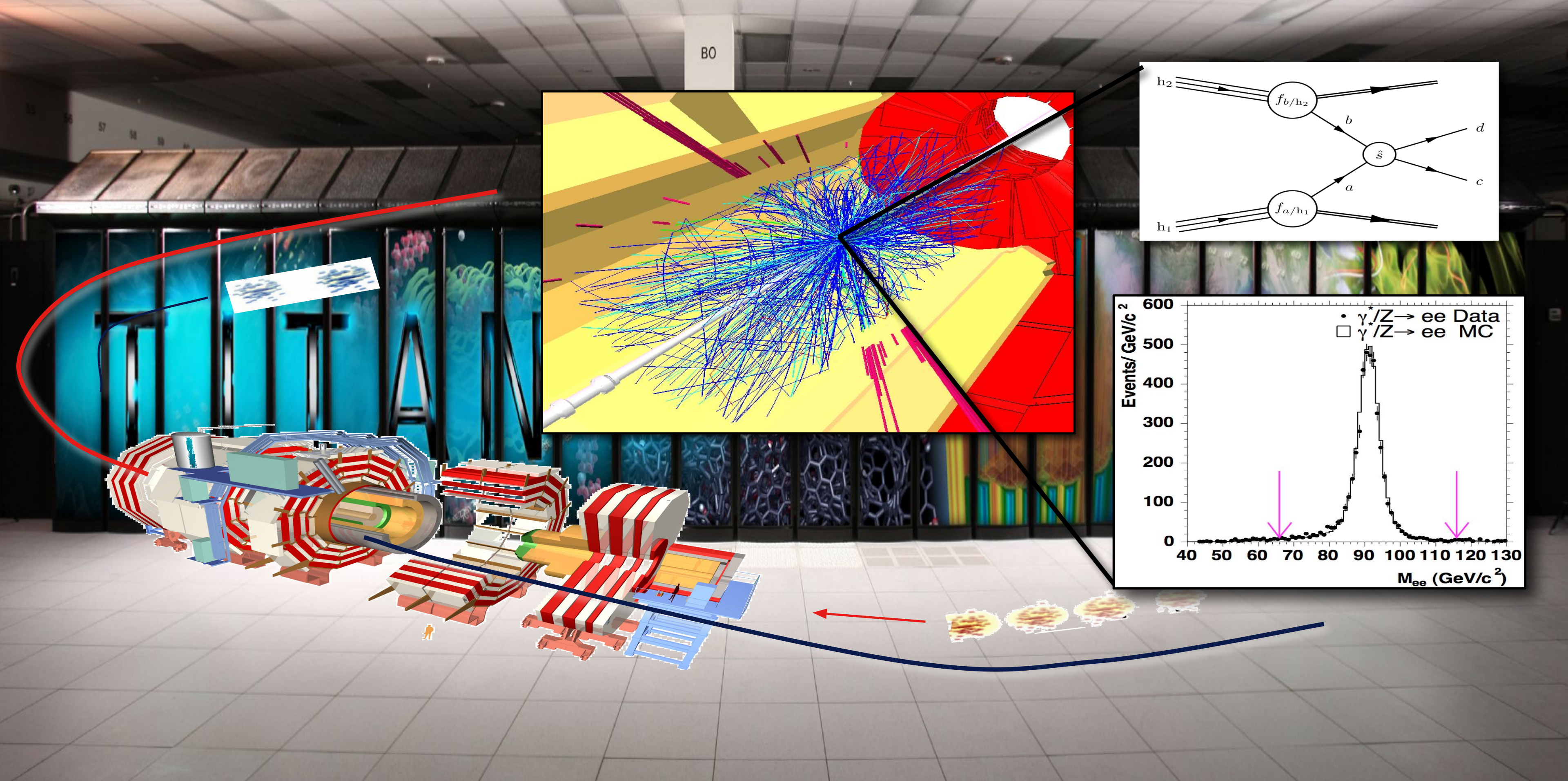


Challenges of O(100K) Concurrent Tasks

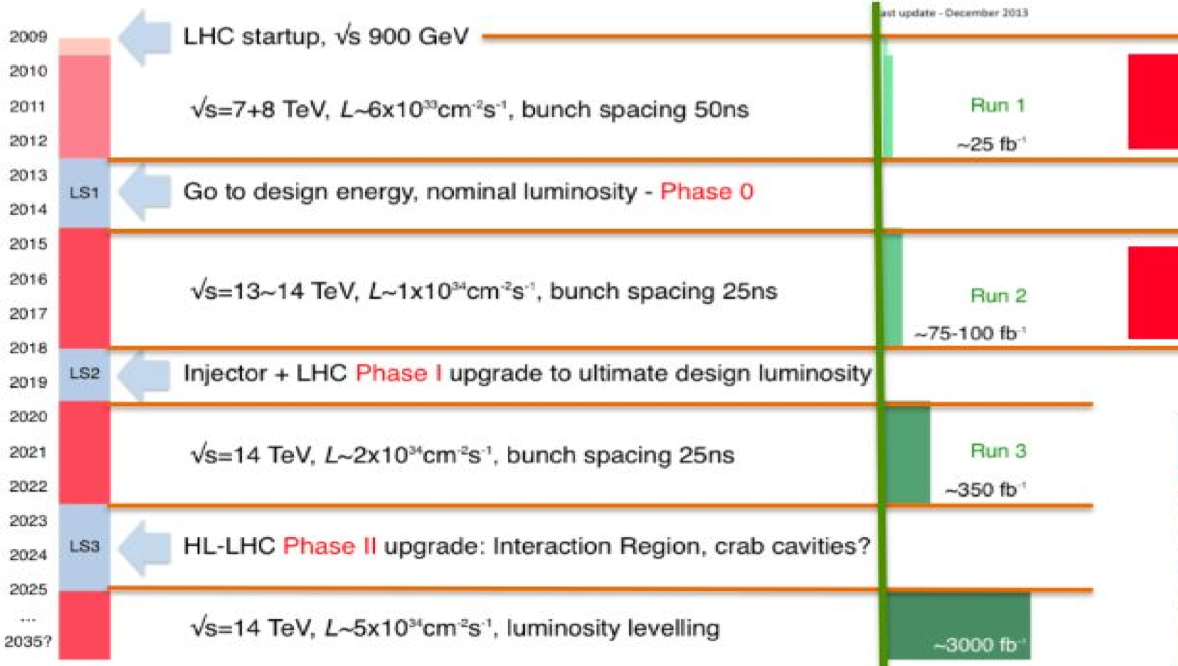
- Agent communication layer (ZMQ) has limited throughput
  - limit is not yet reached
  - bulk messages (is implemented now)
  - separate message channels
  - code optimization
- Agent scheduler (node placement) does not scale well with number of cores
  - bulk operations (schedule bag of tasks at once)
  - good scheduling algorithms and implementations exist
  - code optimization, C-module (instead of pure Python)
- Collecting complete jobs is just as hard as spawning new ones
  - decouple
- Interaction with DB and client side has limited scalability
  - replace with proper messaging protocol (also ZMQ?)

Distributed WLMS

Next Generation Workflow Management for High Energy Physics



LHC Upgrade Timeline



A new detector

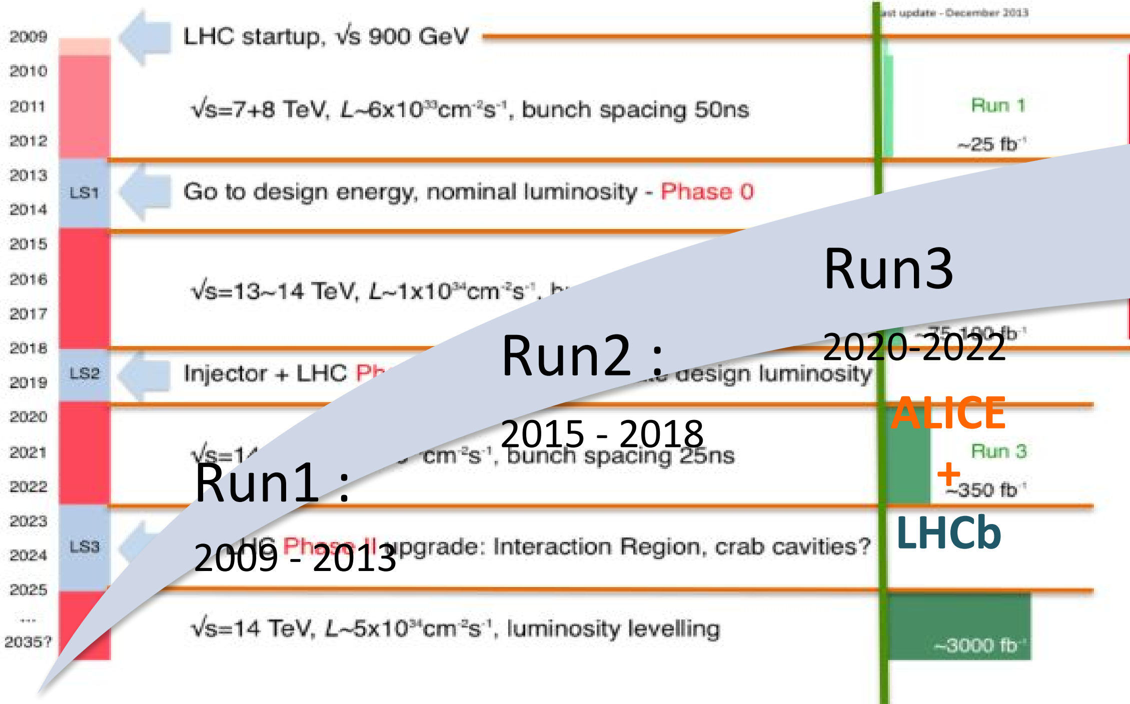
e.g. tracking, calorimeters

In 10 years, increase by factor 10 the LHC luminosity

→ More complex events

→ More Computing Capacity

LHC Upgrade Timeline



A new detector

e.g. tracking, calorimeters

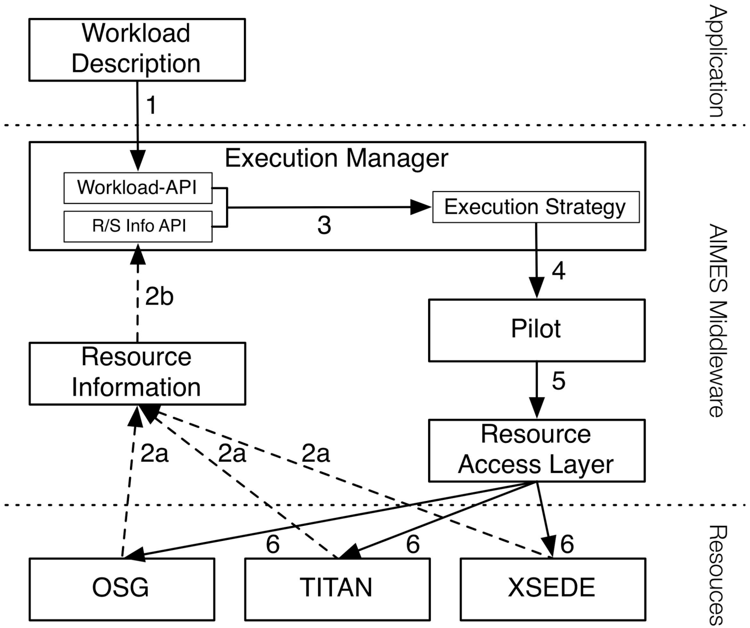
In 10 years, increase by factor 10 the LHC luminosity

→ More complex events

→ More Computing Capacity

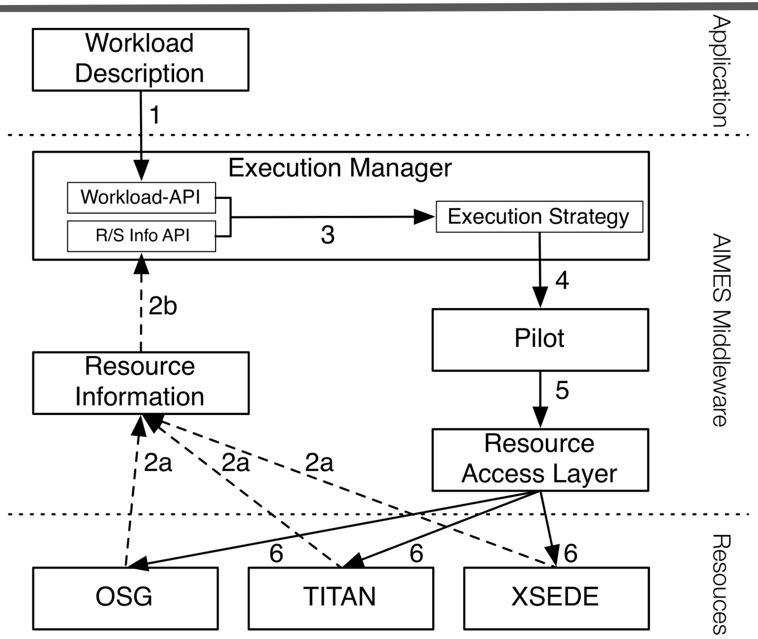
AIMES

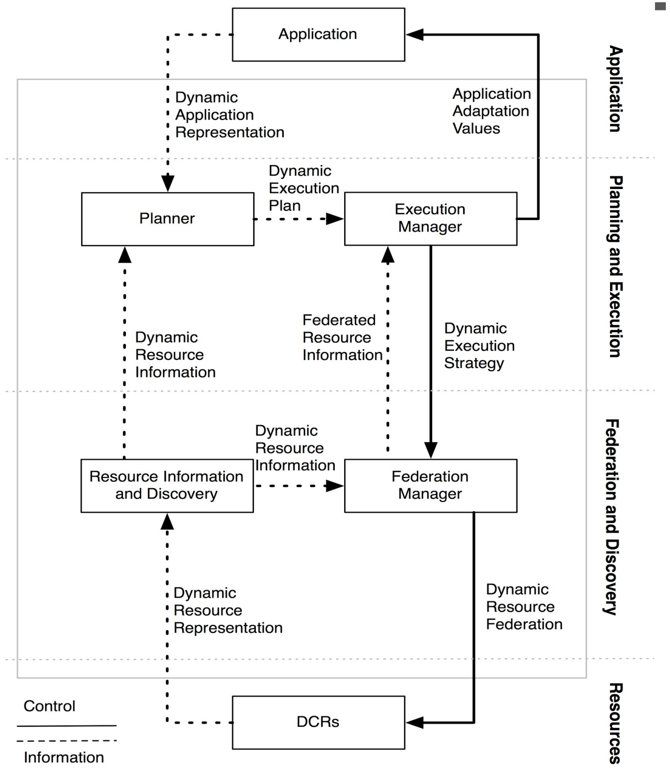
- AIMES: Investigate principles and identify abstractions for distributed execution.
  - *Uniformity in execution* across dynamically federated heterogeneous resources.
  - Conceptual → **implementation** improvements: *"Better"* mapping of workloads to infrastructure and thus also utilization
- AIMES Model of Workload Management:
  - Importance of **dynamic integration** of workload and resource information.
  - Pilot-based **Execution Strategy**: Temporally ordered set of decisions that need to be made when executing a given workload.



*Schematic of **RADICAL-WLMS** approach to workload-resource integration: Evaluate workload requirements & resource capabilities, derive an execution strategy, and enact it, executing the workload on the federated resources.*

Dynamic Resource Management





- PANDA-SAGA : BigPANDA Project (2012-2016)
- PANDA-Pilot : Ongoing redesign for TITAN
- PANDA-AIMES : Heterogeneous workloads and unified execution

Lessons for how we build workflow systems?

“Building Blocks” Approach to Workflow Systems ?

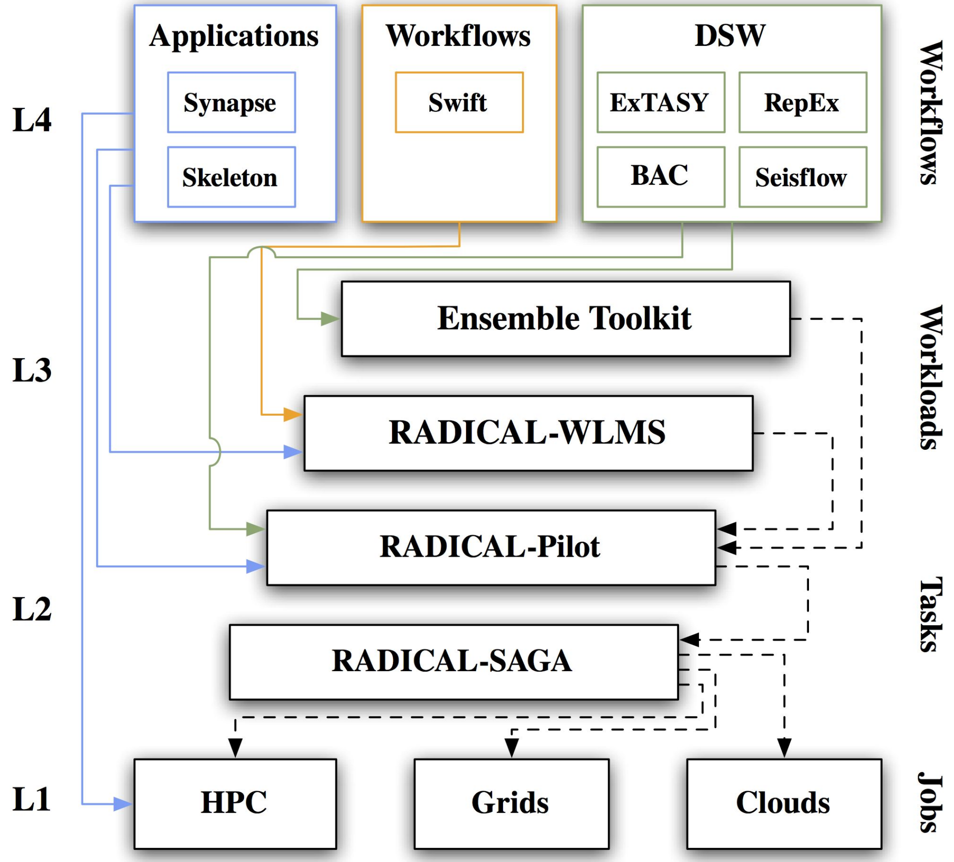
- Workflows aren’t what they used to be!
  - More pervasive, sophisticated but no longer confined to “big science”
  - Diverse requirements, “design points”; unlikely “one size fits all”

- Extend traditional focus from **end-users to workflow system/tool developers**!
  - Building Blocks (BB) permit workflow tools and applications can be built.

- **An illustrative example of a building block common across WFMS**
  - Pilot Job Systems to support scalable execution of multiple tasks

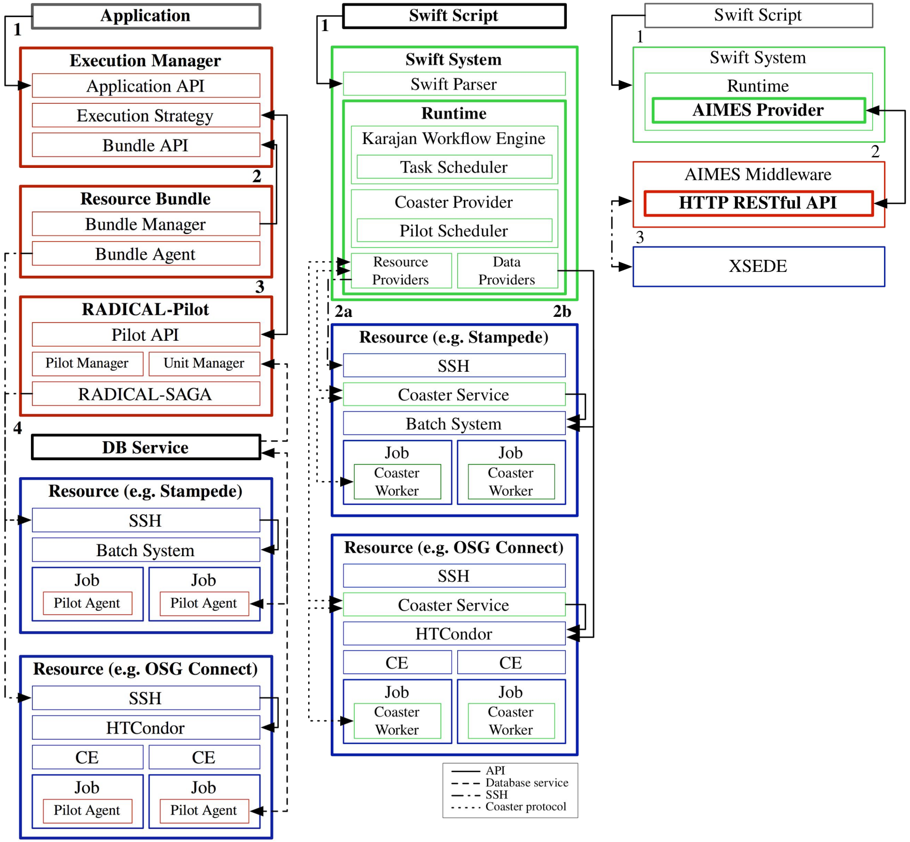
RADICAL-Cybertools:
Abstractions driven building block CI.

RADICAL Cybertools: Abstraction based BB



SWIFT - RADICAL Cybertools Integration

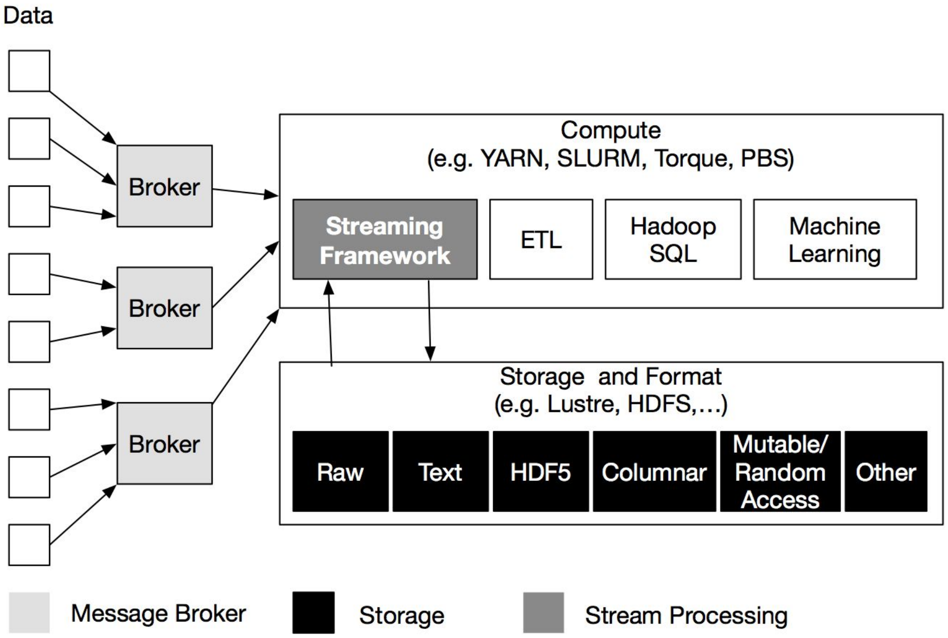
- Many WFMS use pilot systems; greater variance in use of WLMS:
  - Pegasus → Corral/glidein-WMS
  - Condor/glidein →glidein-WMS
  - Swift, Galaxy → No (XSEDE)
- Swift-RCT comparison and integration:
  - Workflow -> Workload -> Tasks abstractions
  - **Uniform execution Model: Binding of tasks and pilots to resources**
  - **Efficient scheduling across pilots and resources**

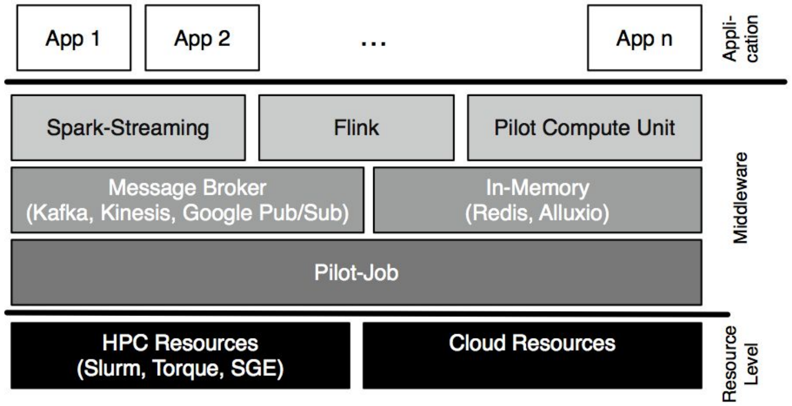


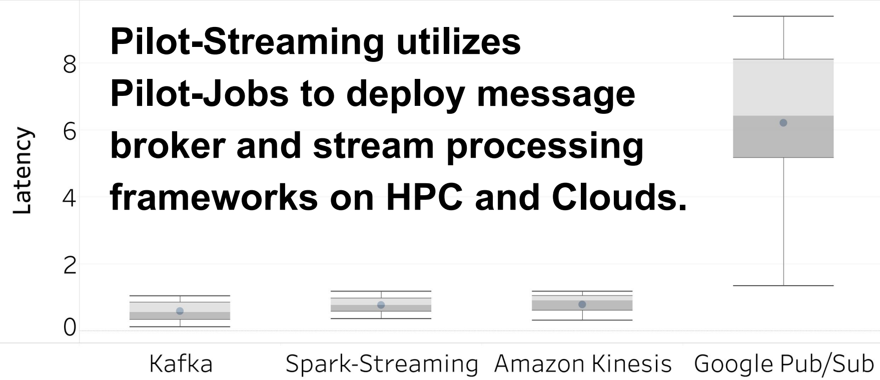
Reference: “Analysis of Distributed Execution of Workloads”, https://arxiv.org/abs/1605.09513

Pilot-Streaming

Pilot-Streaming enables the coupling of data production (simulations) and analysis within HPC environment.

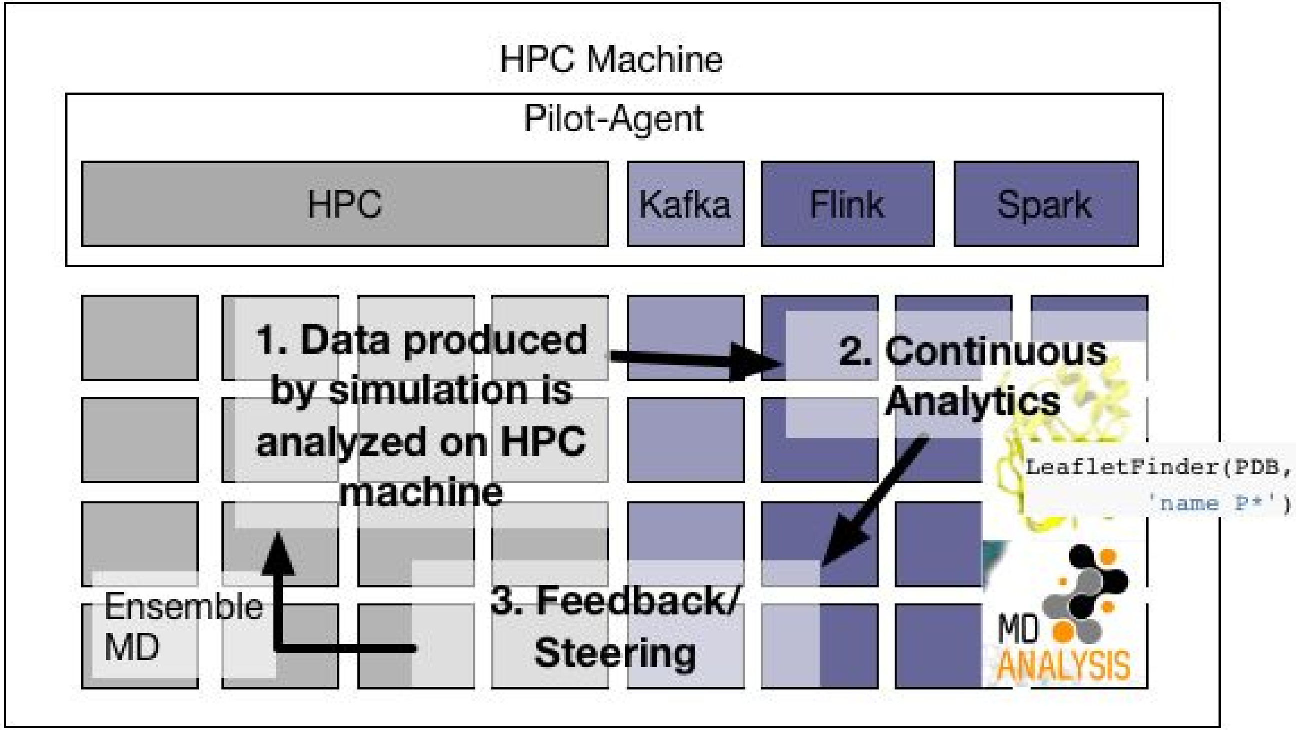






Pilot-Streaming utilizes Pilot-Jobs to deploy message broker and stream processing frameworks on HPC and Clouds.

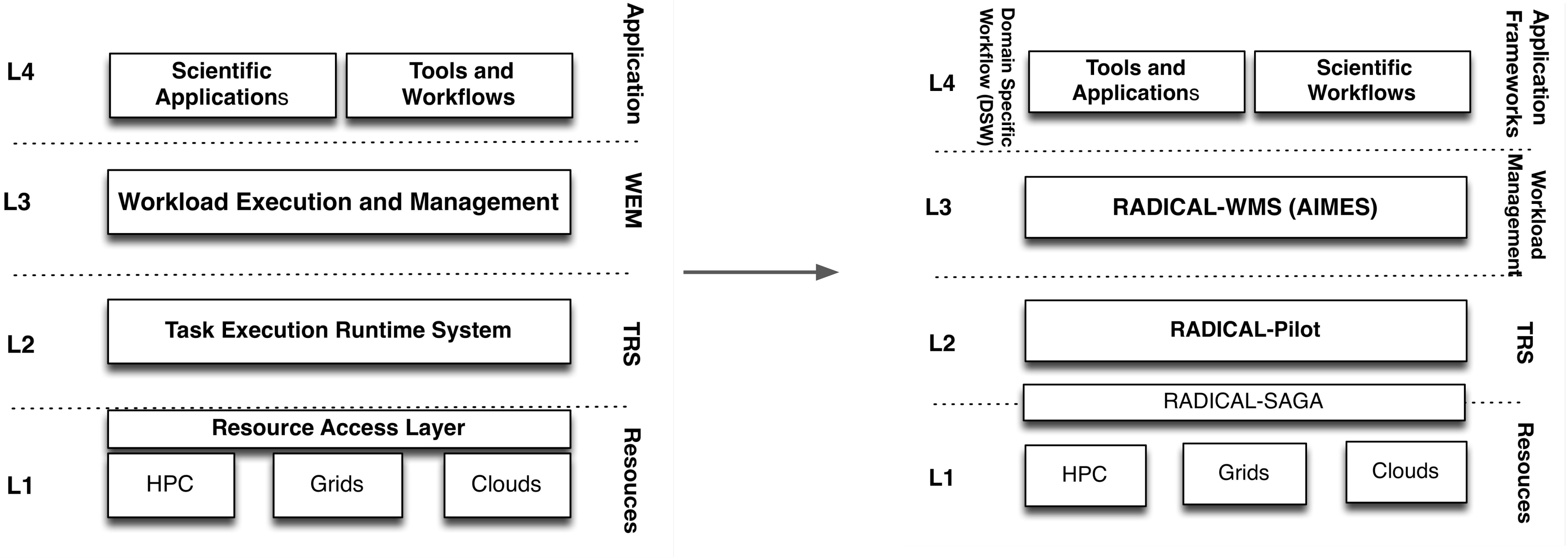
Pilot Streaming: EnsembleMD and MDAnalysis



Pilot-Streaming is utilized to couple MD simulations and continuous analytics (LeafletFinder). By continuously monitoring developed Leaflets.

Dynamic resource management is critical to balance data production rates and analytics needs.

PanDA: BIG and RADICAL!



- PANDA-SAGA : BigPANDA Project (2012-2016)
- PANDA-Pilot : Ongoing redesign for HPC Systems/TITAN
- PANDA-AIMES : Heterogeneous workloads and unified execution model.

Thank you!

Thanks to RADICAL Team

Geoffrey Fox, A Klimentov, K De, J Weissman, D Katz (CS/CI)

Cecilia Clementi, Peter Kasson, Frank Noe (BMS)

Thanks to NSF and DOE

http://radical.rutgers.edu