

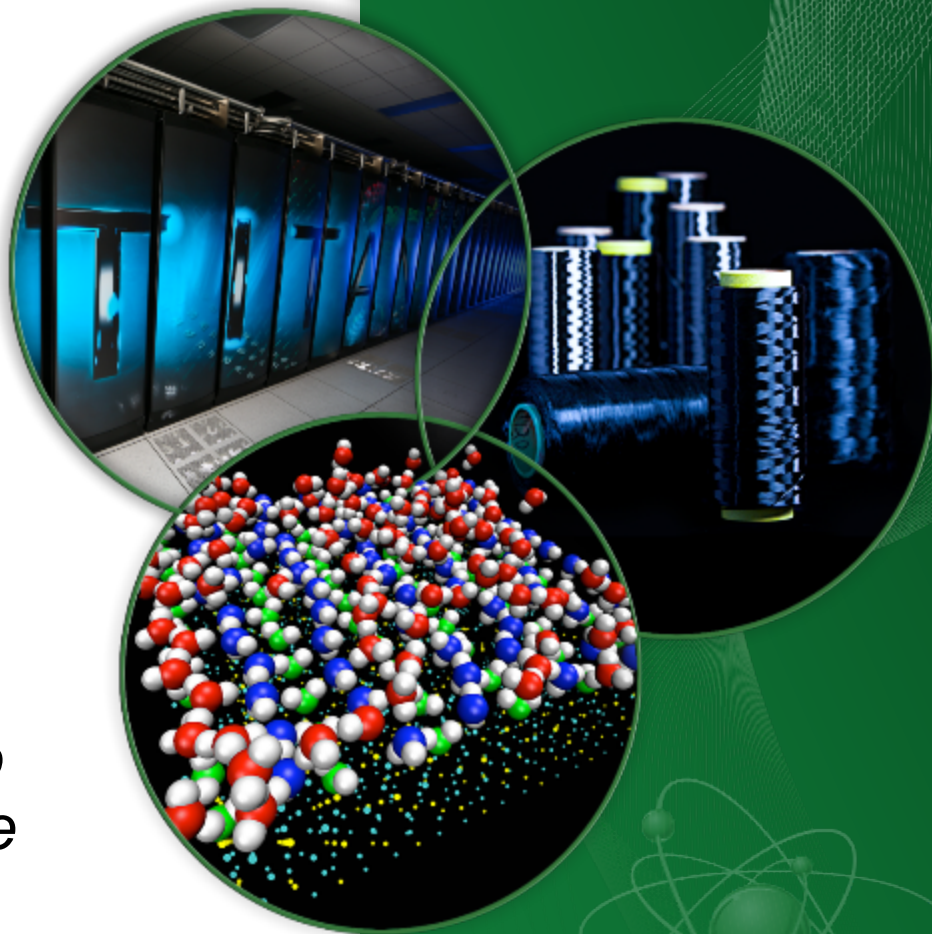
WHY CONVERGENCE?

A CONTRARIAN VIEW AND A PATH TO CONVERGENCE ENABLING SPECIALIZATION

Barney Maccabe
Director, Computer Science and
Mathematics Division



*Pathways to
Convergence*

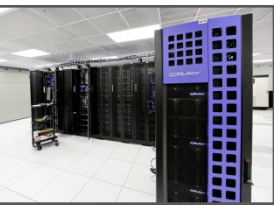


Merging of HPC and data analytics

Future architectures will need to combine HPC and big data analytics into a single box



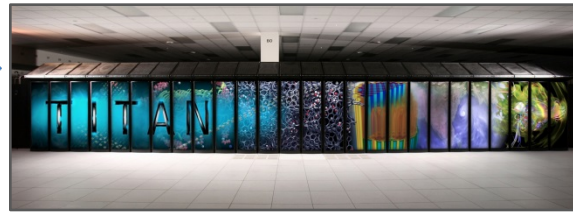
Apollo: Urika-GD
Graph Analytics



Helios: Urika-XA
BDAS
(Hadoop, Spark)



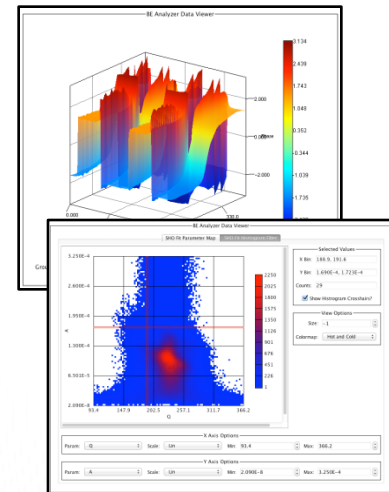
CADES Pods
Compute & Storage



OLCF's Titan
Cray XK7



Metis
Cray XK7



BEAM's "BE Analyzer" tool displaying interactive 2D and 3D views of analyzed multi-dimensional data generated at ORNL's Center for Nanophase Materials Sciences (CNMS)

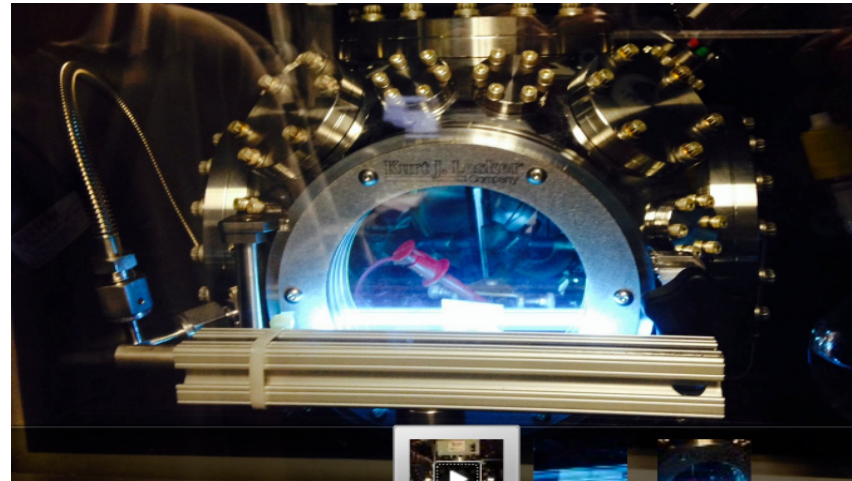
Understanding structure-function evolution in complex solutions of polymers

Scientific Achievement: Developed and utilized an unique environmental chamber for in-situ multimodal interrogation with direct feedback to data analytics and advanced simulations that enabled achieving a new level of control of polymer/small molecule assembly in solution and thin films.

Significance and Impact: A new capability for predictive understanding of structure, dynamics and function of soft materials on a continuous scale, from single molecule to mesoscale thin film assemblies.

Collaborators: Jim Browning, Ilia N. Ivanov, J. Zhu, N. Herath, K. Hong, Valeria Lauter, Rajeev. Kumar, Bobby Sumpter, Hassina Bilheux, Jim Browning, Changwoo Do, Benjamin Doughty, Yingzhong Ma

Citations: Scientific Reports 5: 13407 (2015), Nanoscale DOI: 10.1039/C5NR02037A (2015)



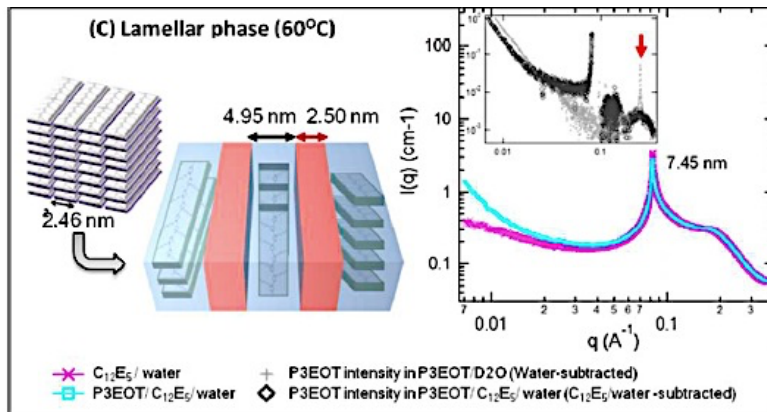
Environment: gas and gas mixtures, oxygen generator (0-100%), vapor of arbitrary liquids, pressure (atm- 10^{-6}), humidity (0-90%), temperature ($RT < T < 300^\circ\text{C}$), light (UV+laser)

Measurements: up to 8 modes simultaneously (PV, diode, transistor, etc.), broad frequency impedance spectroscopy, transmittance, reflectance, photoluminescence, Raman (1064 nm), neutron scattering and reflectometry

Sorption /desorption kinetics: 5 MHz Quartz crystal microbalance (frequency, impedance)

In situ analysis: Artificial neural networks (pattern recognition), statistical (PCA, MCR, etc.)

Structural measurements of thin films– beam line 4a,b Neutron reflectometry (SNS); MD and SCFT theory via OLCF



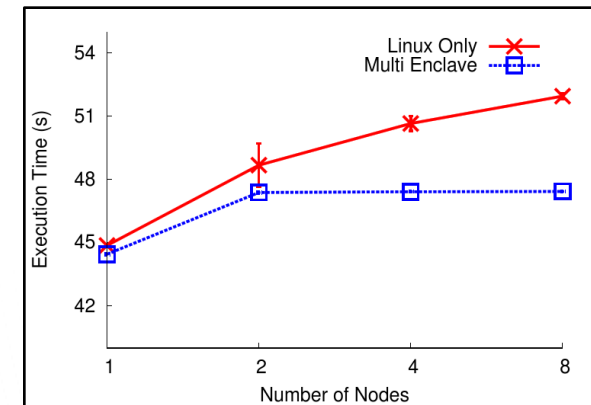
(nit) Picking words (and expectations)

- Converge – “tend to a common result”
 - Merge, become one
- Alternates
 - Integrate, Unify, Combine
 - These tend to preserve characteristics of the components
 - Integration at one level may appear as convergence at higher levels
- Perspective – expecting convergence is unrealistic
 - We still have multiple procedural (object influenced) languages
 - There are significant advantages to specialization
- Approach
 - Define a converged stack, but support combinations of existing stacks
 - Enable incremental migration to the converged environment
 - Migration may never be complete



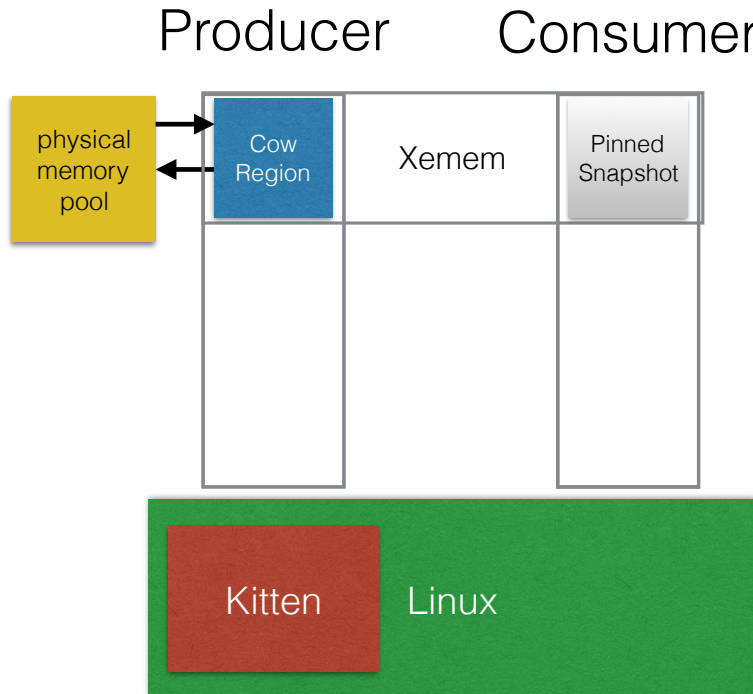
Enabling Multi-OS/R Stack Application Composition

- Problem
 - HPC applications evolving to more compositional approach, overall application is a composition of coupled simulation, analysis, and tool components
 - Each component may have different OS/R requirements, no “one-size-fits-all” OS/R stack
- Solution
 - Partition node-level resources into “enclaves”, run different OS/R instance in each enclave
Pisces Co-kernel Architecture: <http://www.prognosticlab.org/pisces/>
 - Provide tools for creating and managing enclaves, launching applications into enclaves
Leviathan Node Manager: <http://www.prognosticlab.org/leviathan/>
 - Provide mechanisms for cross-enclave application composition and synchronization
XEMEM Shared Memory: <http://www.prognosticlab.org/xemem/>
- Recent results
 - Demonstrated Multi-OS/R approach provides excellent performance isolation; better than native performance possible
 - Demonstrated drop in compatibility with both commodity and Cray Linux environments
- Impact
 - Application components with differing OS/R requirements can be composed together efficiently within a compute node, minimizing off-node data movement
 - Compatible with unmodified vendor provided OS/R environments, simplifies deployment



In-situ Simulation + Analytics composition in single Linux OS vs. Multiple Enclaves

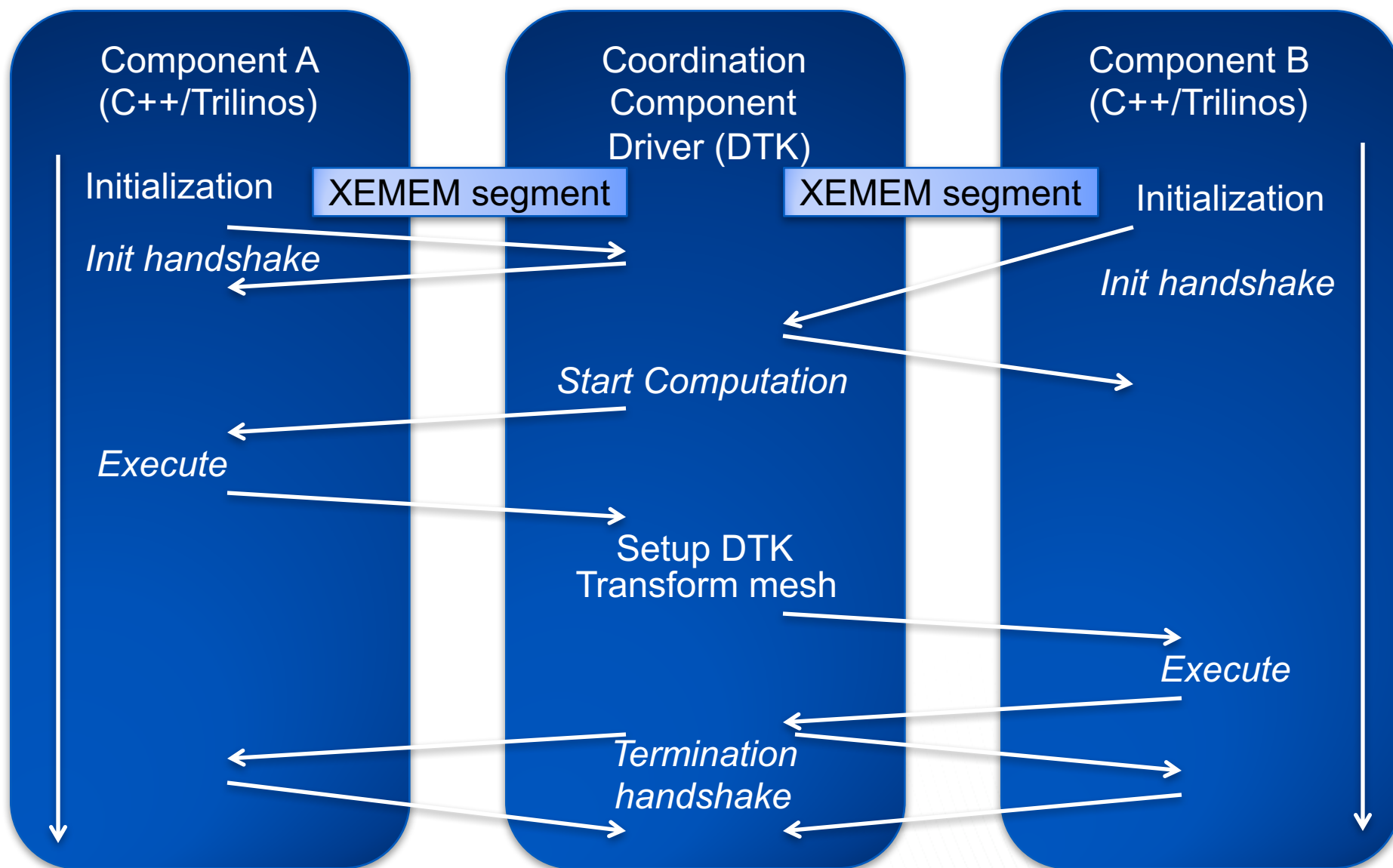
Hobbes XASM: Cross-Enclave Asynchronous Shared Memory



- Mechanism for composition
 - Producer exports a memory snapshot
 - Consumer attaches to the snapshot
 - Copy-on-Write used to allow both to continue asynchronously
- Works across enclave boundaries
 - Linux to Linux
 - Linux to Kitten
 - Kitten to Kitten
 - Native—Native, Native—VM, VM—VM
- Built on top of Hobbes infrastructure

ROSS'16: A Cross-Enclave Composition Mechanism for Exascale System Software

Demonstration Model



Current status and future prospects of optical communications technology and possible impact on future BDEC systems

Tomohiro Kudoh*, Kiyo Ishii**, Shu Namiki**

*The University of Tokyo

**National Institute of Advanced Industrial Science and Technology (AIST)

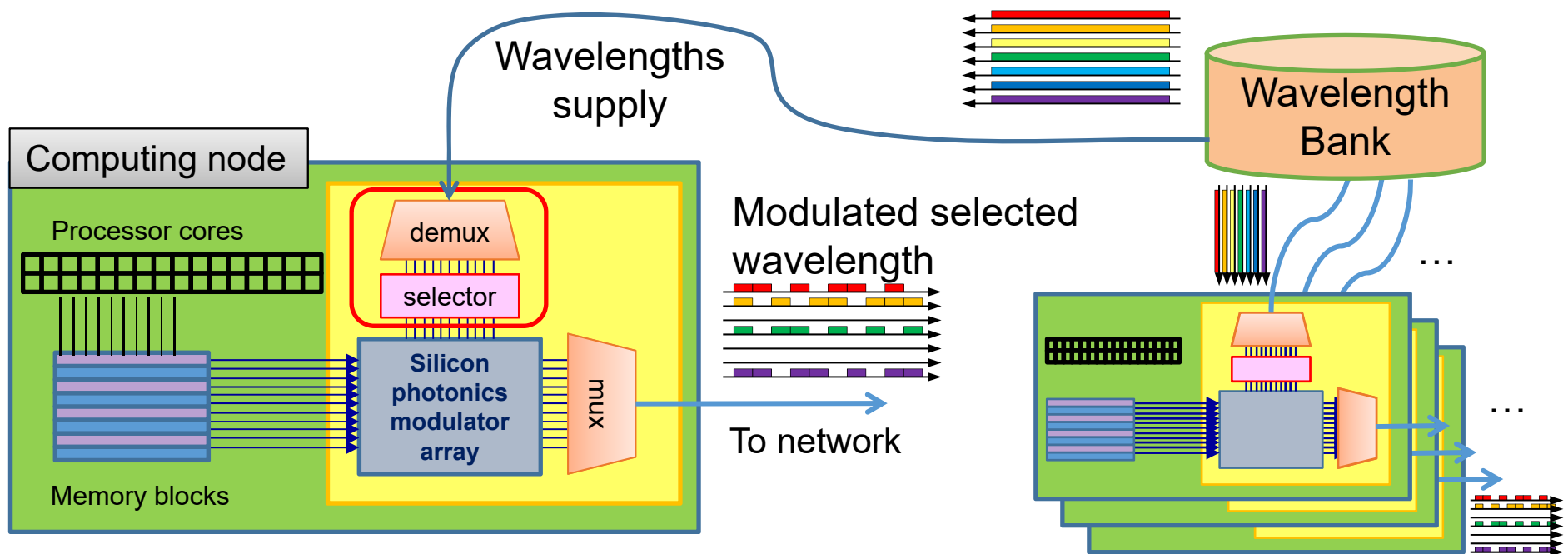
- Data movement
 - One of the keys in convergence of BDA and HPC systems
 - Data in BDA are large and sometimes require real time processing (streaming)
- Optical communication technology to support data movement in future BDEC systems
 - Current status and future prospects

Optical interconnection network

- Interconnection network = interconnections + switches
- Optical interconnections
 - HPC and data centers: direct modulation → around 100Gbps/fiber.
 - Wide area network: polarization/wavelength division multiplexing → tens of Tbps/fiber.
 - Heat and cost of DWDM light source: a wavelength bank (WB), a centralized generator of wavelengths, will solve the problem.
 - Silicon photonics optical circuits can be used for whole light wave processing, including modulation, at a computing node.
- Optical switches
 - Power consumption is not proportional to the bitrate.
 - Can switch more than 10Tbps DWDM signal in one bundle.
 - Disadvantage : slow switching speed and limited number of ports.
 - Expect only moderate progress in the future.

Optical Interconnects

- **Wavelength Bank (WB):**
 - ✓ Single DWDM light source in a system: Distributed to computing nodes via optical amplifiers
 - ✓ No light source is required at each computing node: low cost, low power
- **Silicon photonics optical circuit at each node**
 - ✓ De-multiplex, modulate, multiplex and transmit
 - ✓ Enables hybrid implementation with electronics



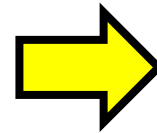
Optical Switches

	MEMS based	PLC based	Silicon photonics	WSS	AWG-R based	SOA based fast multicast switch
Technology	MEMS	PLC	Silicon waveguide	Mostly LCOS	PLC and tunable laser	SOA
Type	Fiber switch	Fiber switch	Fiber switch	Wavelength switch	--	--
Port Count	192x192	32 x 32 16 x 16	32x32	1x20 1x40	720x720	8x8
Port Bandwidth	Ultra wide (tens of THz)	Fairly wide (more than 5 THz)	Fairly wide (more than 5 THz)	Fairly wide (more than 5 THz)	25 - 100GHz	--
Physical Size	Can be large	110 x 115 mm (chip size)	11 x 25 mm (chip size)	--	--	--
Insertion Loss	About 3 dB	6.6 dB	About 20dB	3 - 6 dB	--	--
Crosstalk	very small	< -40dB	< -20dB	< -40dB	--	--
Switching Time	10s of ms	< 3ms	≅ 30 μs	10s of μs	100s of μs	< 10ns
Cost	Moderate to High	Moderate to High	Can be low	Depends on tech	Moderate to High	Moderate to High

Data Affinity to Function Affinity

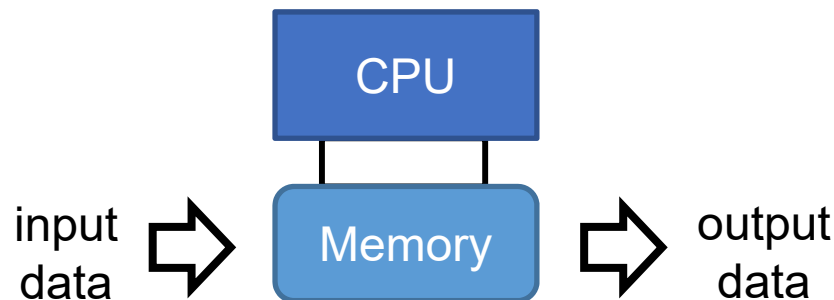
- 10s of Tbps is equivalent to memory bandwidth
- Combine task specific processors in a pipelined manner, instead of using general purpose CPUs with large memory

do **computation**
at where **data** exist



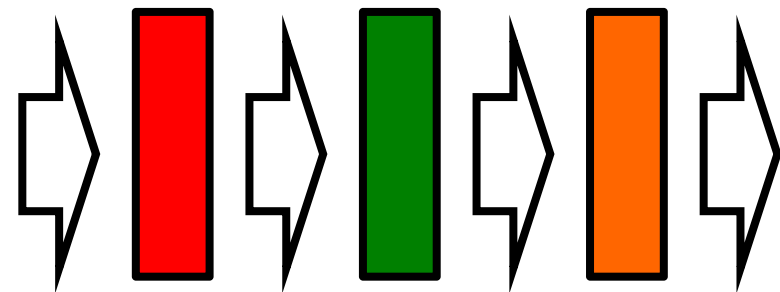
moving **data** to
computation

General purpose CPU



Data Affinity
Scheduling

Heterogeneous task specific
processors



Function Affinity
Scheduling

Numerical Algorithms, Libraries, and Software Frameworks for Future HPC Systems (Towards the Post Moore Era)

Takeshi Iwashita (Hokkaido University)



HOKKAIDO UNIVERSITY



Iwashita lab.



Current situation and future perspectives

(1) Massive parallelism

The growth in the performance of current computing systems relies on the parallelism.

- Increase in number of nodes and cores, instruction sets for parallel processing (SIMD)

At least, $O(10^3)$ threads and $O(10^5)$ computational nodes should be effectively utilized.

(2) New memory and networking system

Moore's law will end within 10 years.

- The flops on a single chip is no longer improved.
- The major architecture of the high-end computing system in the post Moore era is unclear (for me).
- Memory system and networking will be changed. Three dimensional stacking technology or the silicon photonics may contribute to the improvement of the data transfer rate. Moreover, non volatile memory system will be more used.

Complex and deep memory hierarchies and heterogeneity of memory latencies should be considered.



Current situation and future perspective

(3) Energy efficiency (performance per watt)

Flops/watt is more important than Flops in real applications.

- Even after Moore's law ends, the performance per watt can be improved.
- For specific applications or computational kernels, we can effectively use special instructions (e.g., SIMD), accelerators, and reconfigurable hardware (e.g., FPGA) to increase the (effective) flops per watt.

We should investigate implementation methods for these hardware systems and associated algorithms for the typical computational kernels required by real world applications.



Our research targets

(1) Iterative stencil computations

Temporal tiling for three dimensional FDTD method on Xeon Phi processors

[bandwidth reducing]

(2) Parallel in time technique for transient analyses

A parallel two-level multigrid in time solver for non-linear transient finite element analyses for electric motors

[more parallelism]

(3) Approximate matrix computations

Distributed parallel H-matrix library

An approximation technique for a dense matrix

[reducing flops and bandwidth demands, relatively high B/F method]

(4) Sparse matrix solver

Linear solvers using SIMD instructions, accelerators, or new devices

[increase in the performance per watt]

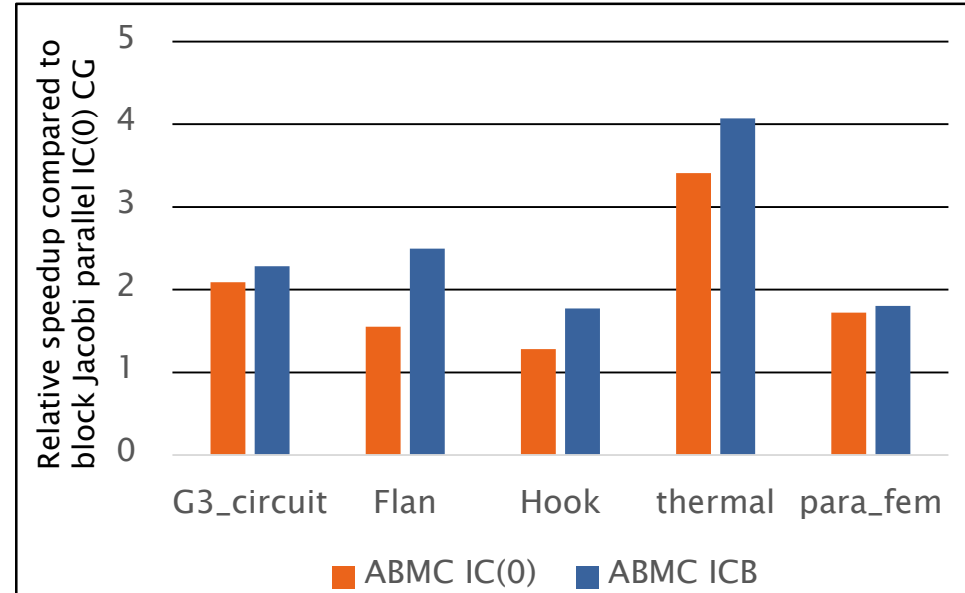
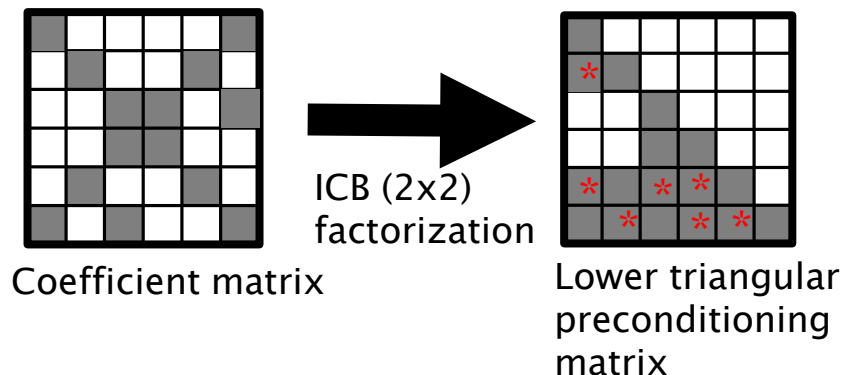


Preconditioning technique utilizing SIMD instructions

T. Iwashita et al., "A new fill-in strategy for IC factorization preconditioning considering SIMD instructions", ISPA 2015.

- **ICB preconditioning**: incomplete Cholesky factorization preconditioning with fill-in strategy based on nonzero blocks
- The preconditioning steps consist of small dense matrix computations which are efficiently processed by SIMD instructions.
- Numerical tests using UF matrix collection datasets showed the effectiveness of the proposed technique.

Experiments on Intel Xeon Phi (KNC) coprocessor using 240 threads



Big Data, Simulations and HPC Convergence

BDEC: Big Data and Extreme-scale Computing
June 15-17 2016 Frankfurt

<http://www.exascale.org/bdec/meeting/frankfurt>

**Geoffrey Fox, Judy Qiu, Shantenu Jha, Saliya Ekanayake,
Supun Kamburugamuve**

June 16, 2016

gcf@indiana.edu

<http://www.dsc.soic.indiana.edu/>, <http://spidal.org/> <http://hpc-abds.org/kaleidoscope/>

Department of Intelligent Systems Engineering

School of Informatics and Computing, Digital Science Center

Indiana University Bloomington



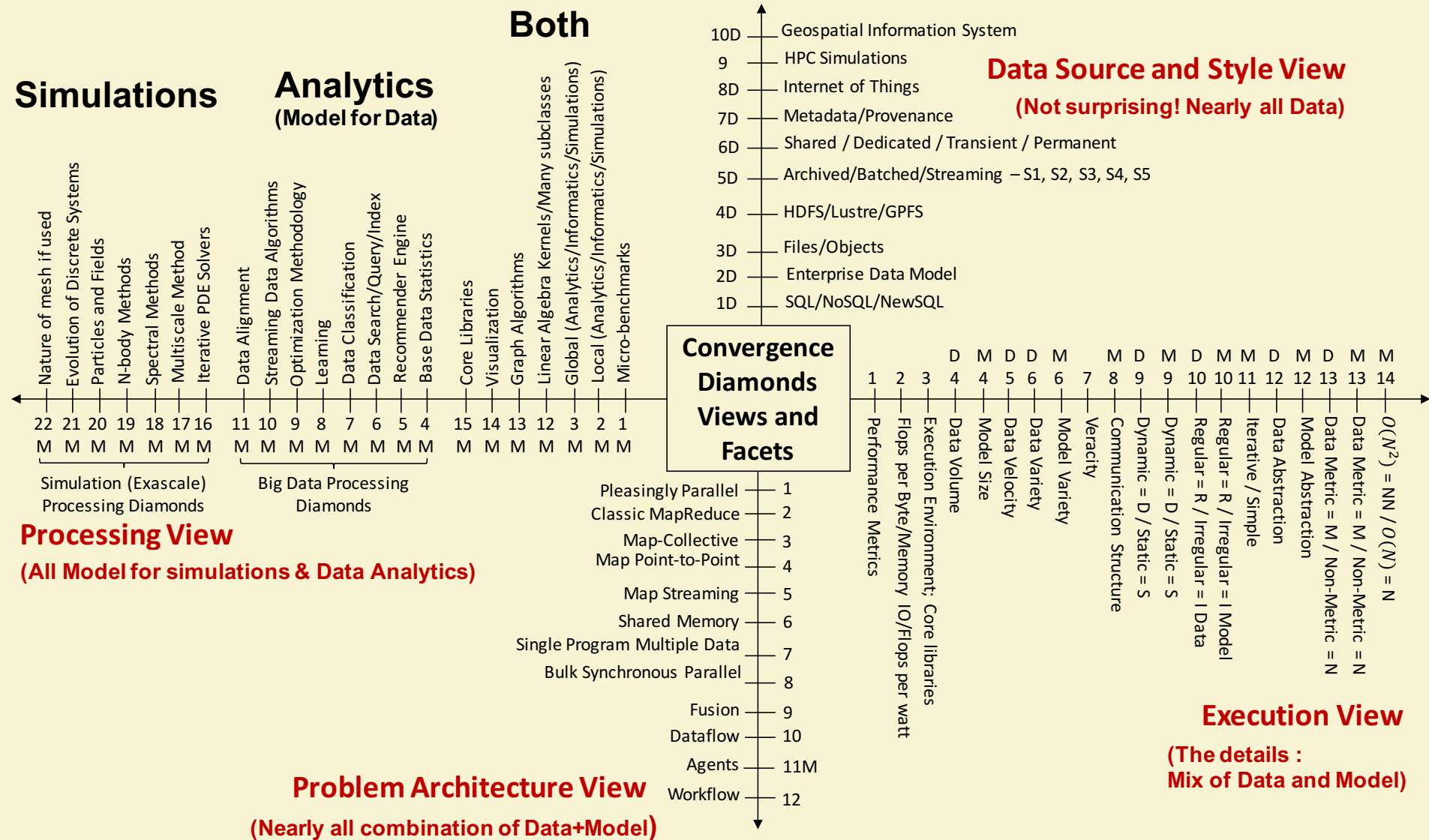
INDIANA UNIVERSITY BLOOMINGTON

SCHOOL OF INFORMATICS AND COMPUTING

Components in Big Data HPC Convergence

- **Applications, Benchmarks and Libraries**
 - 51 NIST Big Data Use Cases, 7 Computational Giants of the NRC Massive Data Analysis, 13 Berkeley dwarfs, 7 NAS parallel benchmarks
 - Unified discussion by separately discussing **data & model** for each application;
 - 64 facets– Convergence Diamonds -- characterize applications
 - *Pleasingly parallel or Streaming* used for data & model;
 - $O(N^2)$ *Algorithm* relevant to model for big data or big simulation
 - “*Lustre v. HDFS*” just describes data
 - “*Volume*” large or small separately for data and model
 - Characterization identifies hardware and software features for each application across big data, simulation; “complete” set of benchmarks (NIST)
- **Software Architecture and its implementation**
 - **HPC-ABDS**: Cloud-HPC interoperable software: performance of HPC (High Performance Computing) and the rich functionality of the Apache Big Data Stack.
 - **Added HPC** to Hadoop, Storm, Heron, Spark; will add to Beam and Flink
 - Work in Apache model contributing code
- **Run same HPC-ABDS across all** platforms but “data management” nodes have different balance in I/O, Network and Compute from “model” nodes
 - Optimize to data and model functions as specified by convergence diamonds
 - Do not optimize for simulation and big data

64 Features in 4 views for Unified Classification of Big Data and Simulation Applications



INDIANA UNIVERSITY BLOOMINGTON

SCHOOL OF INFORMATICS AND COMPUTING

HPC-ABDS

Kaleidoscope of (Apache) Big Data Stack (ABDS) and HPC Technologies

Cross-Cutting Functions	17) Workflow-Orchestration: ODE, ActiveBPEL, Airavata, Pegasus, Kepler, Swift, Taverna, Triana, Trident, BioKepler, Galaxy, IPython, Dryad, Naiad, Oozie, Tez, Google FlumeJava, Crunch, Cascading, Scalding, e-Science Central, Azure Data Factory, Google Cloud Dataflow, NiFi (NSA), Jitterbit, Talend, Pentaho, Apatar, Docker Compose, KeystoneML
1) Message and Data Protocols: Avro, Thrift, Protobuf	16) Application and Analytics: Mahout, MLlib, MLbase, DataFu, R, pbdR, Bioconductor, ImageJ, OpenCV, Scalapack, PetSc, PLASMA MAGMA, Azure Machine Learning, Google Prediction API & Translation API, mply, scikit-learn, PyBrain, CompLearn, DAAL(Intel), Caffe, Torch, Theano, DL4j, H2O, IBM Watson, Oracle PGX, GraphLab, GraphX, IBM System G, GraphBuilder(Intel), TinkerPop, Parasol, Dream:Lab, Google Fusion Tables, CINET, NWB, Elasticsearch, Kibana, Logstash, Graylog, Splunk, Tableau, D3.js, three.js, Potree, DC.js, TensorFlow, CNTK
2) Distributed Coordination : Google Chubby, Zookeeper, Giraffe, JGroups	15B) Application Hosting Frameworks: Google App Engine, AppScale, Red Hat OpenShift, Heroku, Aerobatic, AWS Elastic Beanstalk, Azure, Cloud Foundry, Pivotal, IBM BlueMix, Ninefold, Jelastic, Stackato, appfog, CloudBees, Engine Yard, CloudControl, dotCloud, Dokku, OSGi, HUBzero, OODT, Agave, Atmosphere 15A) High level Programming: Kite, Hive, HCatalog, Tajo, Shark, Phoenix, Impala, MRQL, SAP HANA, HadoopDB, PolyBase, Pivotal HD/Hawq, Presto, Google Dremel, Google BigQuery, Amazon Redshift, Drill, Kyoto Cabinet, Pig, Sawzall, Google Cloud DataFlow, Summingbird
3) Security & Privacy: InCommon, Eduroam, OpenStack, Keystone, LDAP, Sentry, Sqrrl, OpenID, SAML OAuth	14B) Streams: Storm, S4, Samza, Granules, Neptune, Google MillWheel, Amazon Kinesis, LinkedIn, Twitter Heron, Databus, Facebook Puma/Ptail/Scribe/ODS, Azure Stream Analytics, Floe, Spark Streaming, Flink Streaming, DataTurbine 14A) Basic Programming model and runtime, SPMD, MapReduce: Hadoop, Spark, Twister, MR-MPI, Stratosphere (Apache Flink), Reef, Disco, Hama, Giraph, Pregel, Pegasus, Ligra, GraphChi, Galois, Medusa-GPU, MapGraph, Totem
4) Monitoring: Ambari, Ganglia, Nagios, Inca	13) Inter process communication Collectives, point-to-point, publish-subscribe: MPI, HPX-5, Argo BEAST HPX-5 BEAST PULSAR, Harp, Netty, ZeroMQ, ActiveMQ, RabbitMQ, NaradaBrokering, QPid, Kafka, Kestrel, JMS, AMQP, Stomp, MQTT, Marionette Collective, Public Cloud: Amazon SNS, Lambda, Google Pub Sub, Azure Queues, Event Hubs
21 layers Over 350 Software Packages	12) In-memory databases/caches: Gora (general object from NoSQL), Memcached, Redis, LMDB (key value), Hazelcast, Ehcache, Infinispan, VoltDB, H-Store
January 29 2016	12) Object-relational mapping: Hibernate, OpenJPA, EclipseLink, DataNucleus, ODBC/JDBC
	12) Extraction Tools: UIMA, Tika
	11C) SQL(NewSQL): Oracle, DB2, SQL Server, SQLite, MySQL, PostgreSQL, CUBRID, Galera Cluster, SciDB, Rasdaman, Apache Derby, Pivotal Greenplum, Google Cloud SQL, Azure SQL, Amazon RDS, Google F1, IBM dashDB, N1QL, BlinkDB, Spark SQL
	11B) NoSQL: Lucene, Solr, Solandra, Voldemort, Riak, ZHT, Berkeley DB, Kyoto/Tokyo Cabinet, Tycoon, Tyrant, MongoDB, Espresso, CouchDB, Couchbase, IBM Cloudant, Pivotal Gemfire, HBase, Google Bigtable, LevelDB, Megastore and Spanner, Accumulo, Cassandra, RYA, Sqrrl, Neo4J, graphdb, Yarcdata, AllegroGraph, Blazegraph, Facebook Tao, Titan:db, Jena, Sesame Public Cloud: Azure Table, Amazon Dynamo, Google DataStore
	11A) File management: iRODS, NetCDF, CDF, HDF, OPeNDAP, FITS, RCFile, ORC, Parquet
	10) Data Transport: BitTorrent, HTTP, FTP, SSH, Globus Online (GridFTP), Flume, Sqoop, Pivotal GPLOAD/GPFDIST
	9) Cluster Resource Management: Mesos, Yarn, Helix, Llama, Google Omega, Facebook Corona, Celery, HTCondor, SGE, OpenPBS, Moab, Slurm, Torque, Globus Tools, Pilot Jobs
	8) File systems: HDFS, Swift, Haystack, f4, Cinder, Ceph, FUSE, Gluster, Lustre, GPFS, GFFS Public Cloud: Amazon S3, Azure Blob, Google Cloud Storage
	7) Interoperability: Libvirt, Libcloud, JClouds, TOSCA, OCCl, CDMI, Whirr, Saga, Genesis
	6) DevOps: Docker (Machine, Swarm), Puppet, Chef, Ansible, SaltStack, Boto, Cobbler, Xcat, Razor, CloudMesh, Juju, Foreman, OpenStack Heat, Sahara, Rocks, Cisco Intelligent Automation for Cloud, Ubuntu MaaS, Facebook Tupperware, AWS OpsWorks, OpenStack Ironic, Google Kubernetes, Buildstep, Gitreceive, OpenTOSCA, Winery, CloudML, Blueprints, Terraform, DevOpsSlang, Any2Api
	5) IaaS Management from HPC to hypervisors: Xen, KVM, QEMU, Hyper-V, VirtualBox, OpenVZ, LXC, Linux-Vserver, OpenStack, OpenNebula, Eucalyptus, Nimbus, CloudStack, CoreOS, rkt, VMware ESXi, vSphere and vCloud, Amazon, Azure, Google and other public Clouds Networking: Google Cloud DNS, Amazon Route 53

HPC-ABDS Activities of NSF14-43054

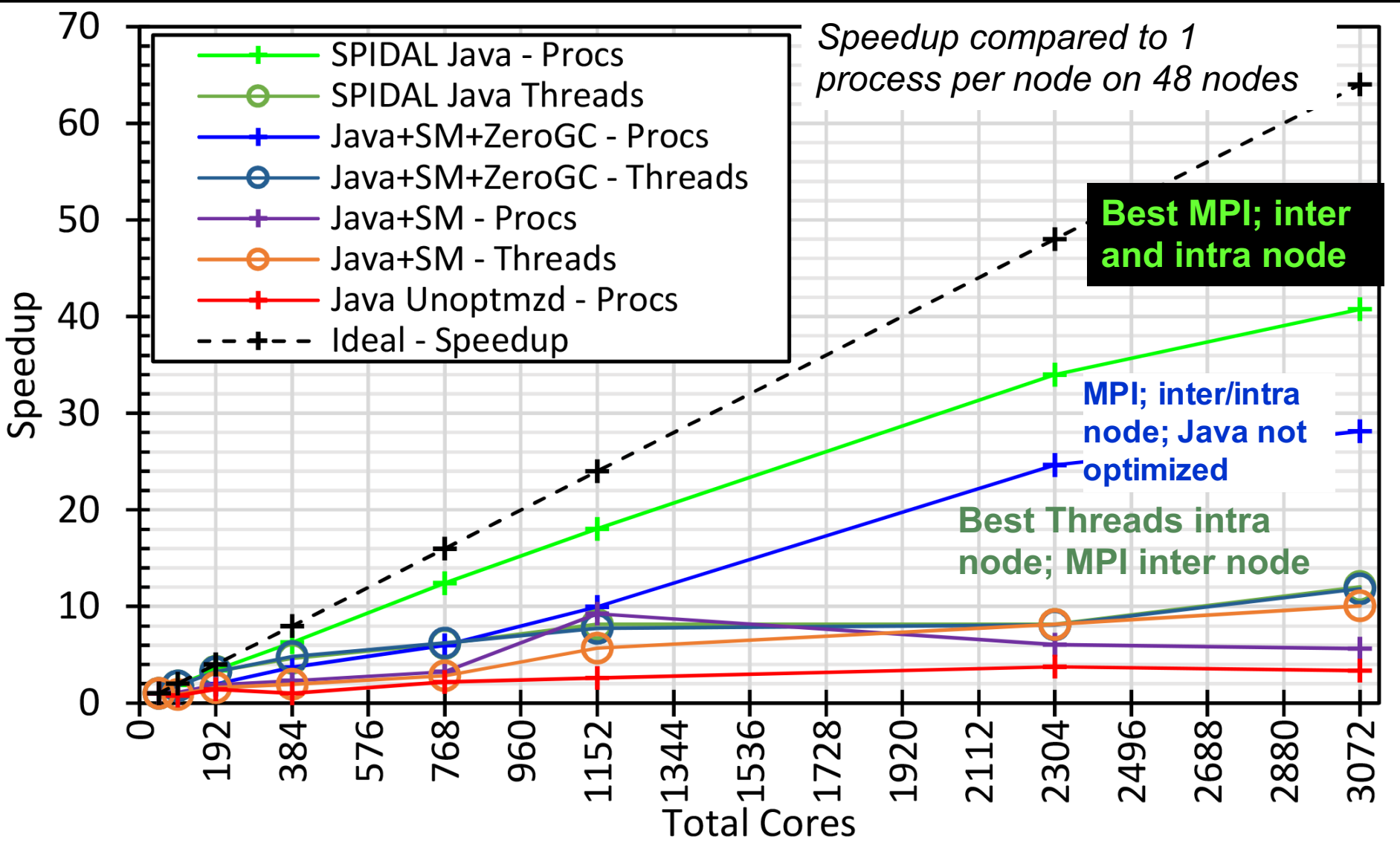
- **Level 17: Orchestration:** Apache Beam (Google Cloud Dataflow)
- **Level 16: Applications:** Datamining for molecular dynamics, Image processing for remote sensing and pathology, graphs, streaming, bioinformatics, social media, financial informatics, text mining
- **Level 16: Algorithms:** Generic and application specific; **SPIDAL Library**
- **Level 14: Programming:** Storm, Heron (Twitter replaces Storm), Hadoop, Spark, Flink. Improve Inter- and Intra-node performance; science data structures
- **Level 13: Runtime Communication:** Enhanced Storm and Hadoop (Spark, Flink, Giraph) using HPC runtime technologies, Harp
- **Level 11: Data management:** Hbase and MongoDB integrated via use of Beam and other Apache tools; enhance Hbase
- **Level 9: Cluster Management:** Integrate Pilot Jobs with Yarn, Mesos, Spark, Hadoop; integrate Storm and Heron with Slurm
- **Level 6: DevOps:** Python Cloudmesh virtual Cluster Interoperability



Convergence Language: Recreating Java Grande

128 24 core Haswell nodes on SPIDAL Data Analytics

Best Java factor of 10 faster than “out of the box”; comparable to C++





Big Data Analytics and High Performance Computing Convergence Through Workflows and Virtualization

Ewa Deelman, Ph.D.

Science Automation Technologies Group

USC Information Sciences Institute

BDEC Workshop, Frankfurt, June 15-17 2016

BDA and HPC convergence

- **Users don't want to worry about where to run**
 - need results in a timely manner
 - need ease of use and automation
- **Some applications naturally cross the system boundaries:**
 - simulation and data mining (ex-situ or in-situ)
- **Workflows naturally combine heterogeneous applications**
 - tightly coupled codes
 - machine learning loosely coupled applications
 - independent high-throughput tasks
 - a mix of all
- **Workflow Management Systems**
 - + can cross boundaries
 - + can select the appropriate resources, schedule the needed data movement, send tasks for execution on the target resources
 - keep the different infrastructures separate and makes it hard to co-locate extreme computation and analytics.



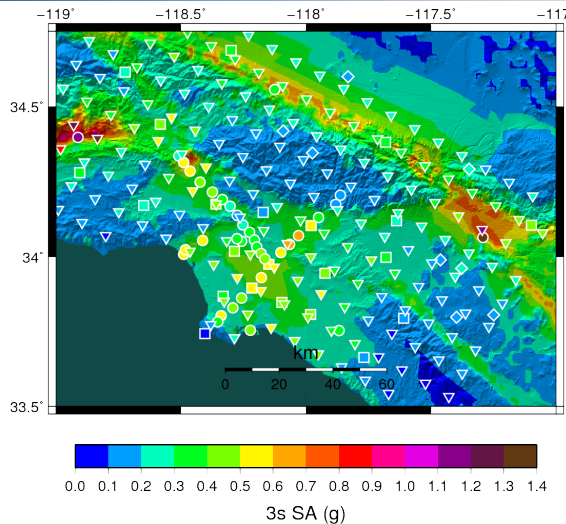
Southern California Earthquake Center

Mix of HPC and HTC codes

CyberShake PSHA Workflow

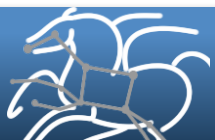
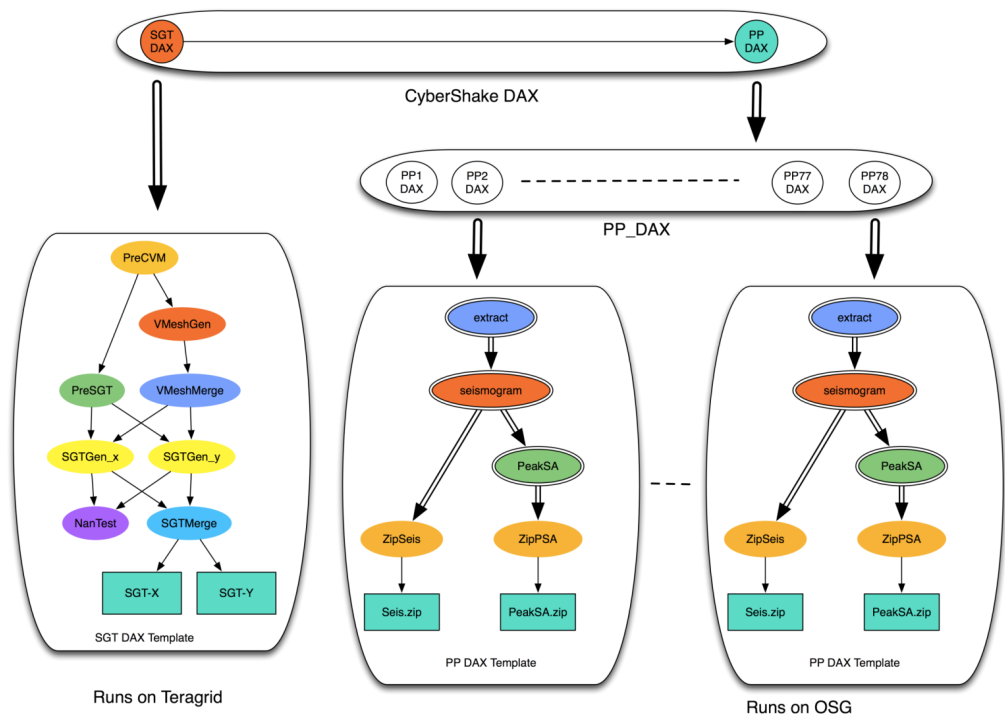
❖ Description

- ❖ Builders ask seismologists: “What will the peak ground motion be at my new building in the next 50 years?”
- ❖ Seismologists answer this question using Probabilistic Seismic Hazard Analysis (PSHA)



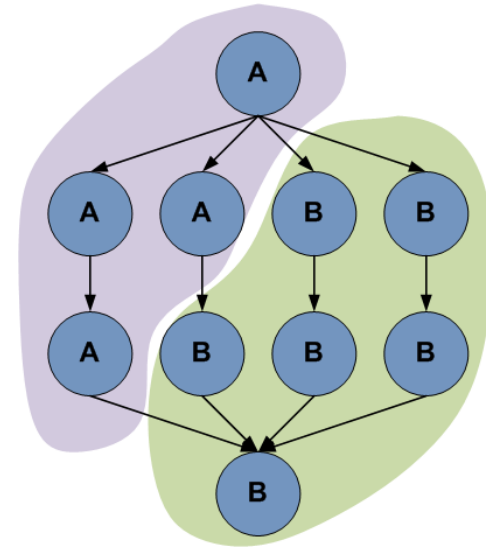
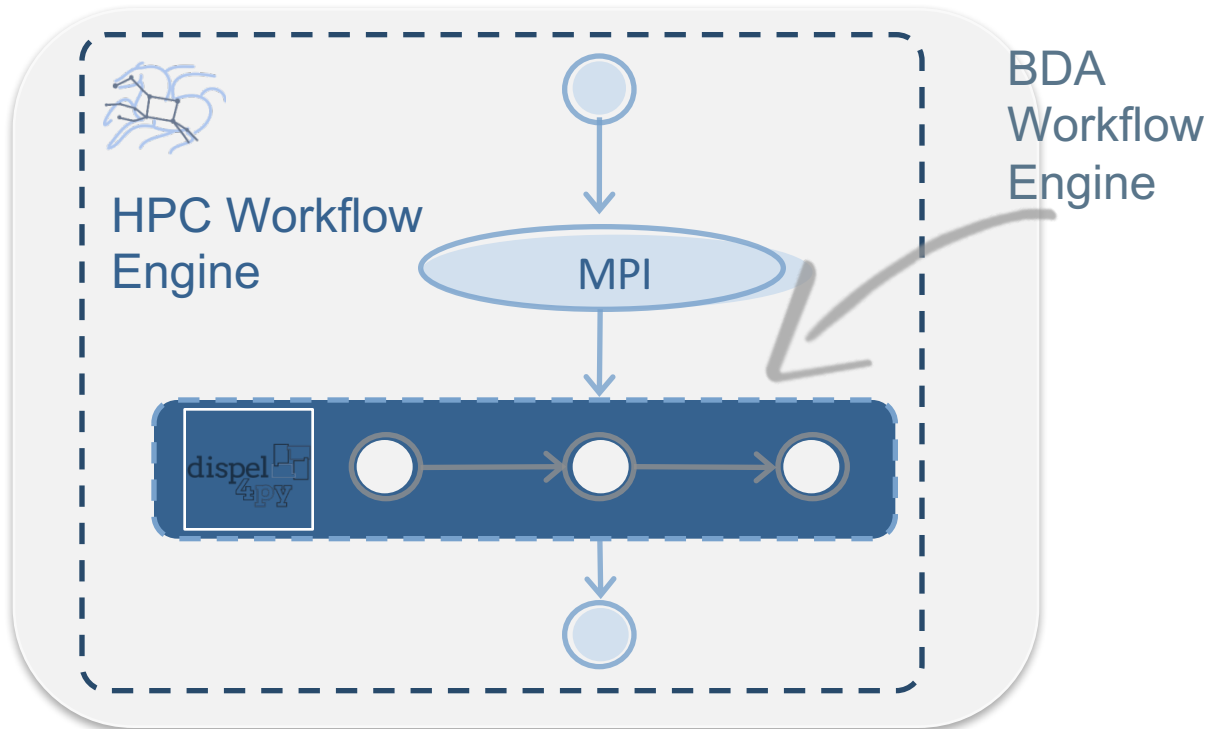
239 Workflows

- Each site in the input map corresponds to one workflow
- Each workflow has:
 - ❖ 820,000 tasks



Solutions

Partition the workflow into subworkflows and send them for execution to the target system, managed by an MPI-based workflow engine



Similar solution for a mix of HPC and BDA, embed a BDA workflow within overall workflow and use specific WE

Still BDA on BDA platforms



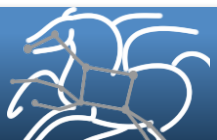
Where do we go from here?

- Need a more natural way of managing BDA tasks within HPC
- Could develop a workflow engine to manage BDA apps on HPC
- Potentially combine resource provisioning and task scheduling
 - Scheduler provides a portion of the machine to the WMS
 - WMS manages the software deployment, configuration, and task scheduling/BDA engine launch
- Problems:
 - Security concerns of HPC admins
 - Complexity of setting up the correct software environment
 - Complexity of the HPC system, in particular the deep memory hierarchy and its impact on the overall system performance and energy consumption
 - Potential performance degradation and suboptimal use of resources



Possible Solutions

- **Work closely with resource providers to understand concerns, develop “trusted” resource/work management systems, develop specialized monitoring tools, and auditing mechanisms**
- **Develop tools that automate the software environment set up, explore virtualization, need to manage the container deployment and environment testing automatically**
- **Develop data management capabilities that can seamlessly manage different types and amounts of data across workflow components**
 - Need an adequate level of abstraction and need to be easy to incorporate in legacy applications
 - Develop data-aware work scheduling
- **Realize that there may need to be some performance degradation in order to support scientific productivity and system manageability**
- **Help characterize resource usage and provide incentives for good resource usage**
- **Systems need to be made reproducibility aware:**
 - Insight into how reproducible the computation is
 - Transparency: how the computation was performed, how the environment and the applications were set up so that the results can be inspected
 - Support reuse and sharing



Extreme Scale Scientific Data Sets

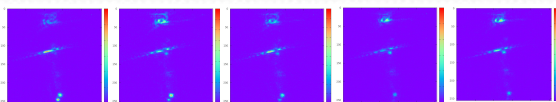
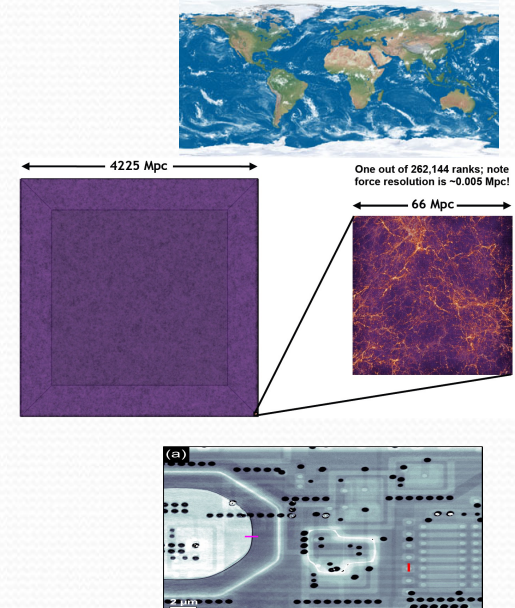
On demand Infrastructure and Compression (merge of 2 white papers)

Franck Cappello^{1,2}, Katrin Heitmann¹, Gabrielle Allen², Sheng Di¹, William Gropp², Salman Habib¹, Ed Seidel², Brandon George⁴, Brett Bode², Tim Boerner², Maxine D. Brown³, Michelle Butler², Randal L Butler², Kenton G. McHenry², Athol J Kemball², Rajkumar Kettimuthu¹, Ravi Madduri¹, Alex Parga², Roberto R. Sisneros², Corby B. Schmitz¹, Sean R Stevens², Matthew J Turk², Tom Uram¹, David Wheeler², Michael J. Wilde¹, Justin M. Wozniak¹.

¹Argonne National Laboratory, ²NCSA, ³UIC, ⁴DDN

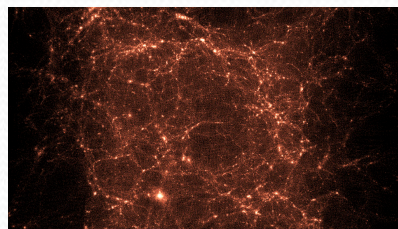
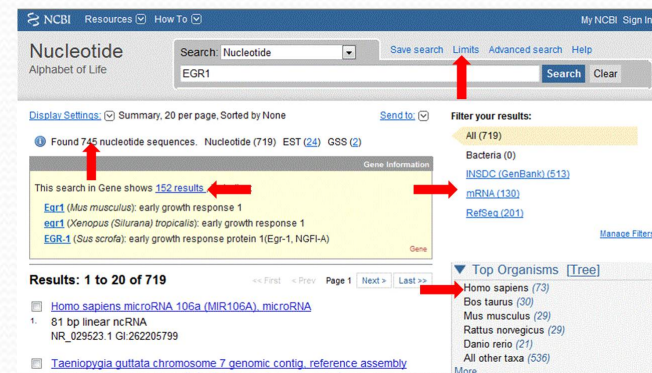
Sciences produce gigantic datasets that are hard to transfer, store & analyze

- Today's scientific research is using simulation or instruments and produces extremely large of data sets to process/analyze
- Examples:
 - Cosmology Simulation (HACC):
 - A total of **>20PB** of data when simulating trillion of particles
 - Petascale systems FS ~20PB
 - data reduction is needed
 - currently drop 9 snapshots over 10
 - APS-U (next-generation APS project at Argonne):
 - Brain Initiatives: in the order of **100PB** of storage: hundreds of specimens, each requiring 150TB of storage.



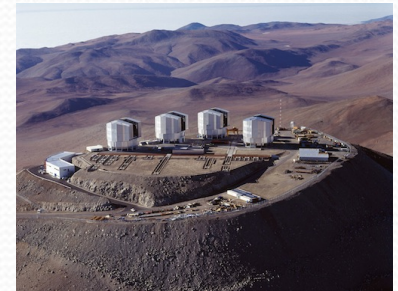
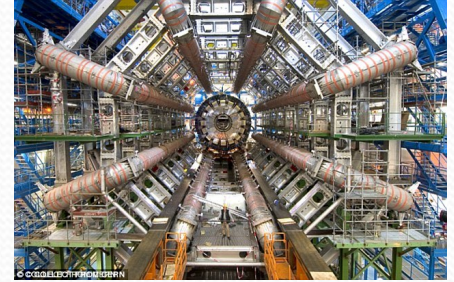
Cost of producing, moving and storing science data pushes toward sharing

- From 1 producer, 1 user to 1 producer, many users
- Examples:
 - LHC
 - The Cancer Imaging Archive
 - Cosmological surveys (e.g. Dark energy survey)
 - Nucleotide sequence, genome sequence, protein, etc. databases
 - Climate simulations (International Panel on Climate Change)
 - Cosmology simulations
 - Open Access Directory
 - Etc.

Systems and sites tend to specialize

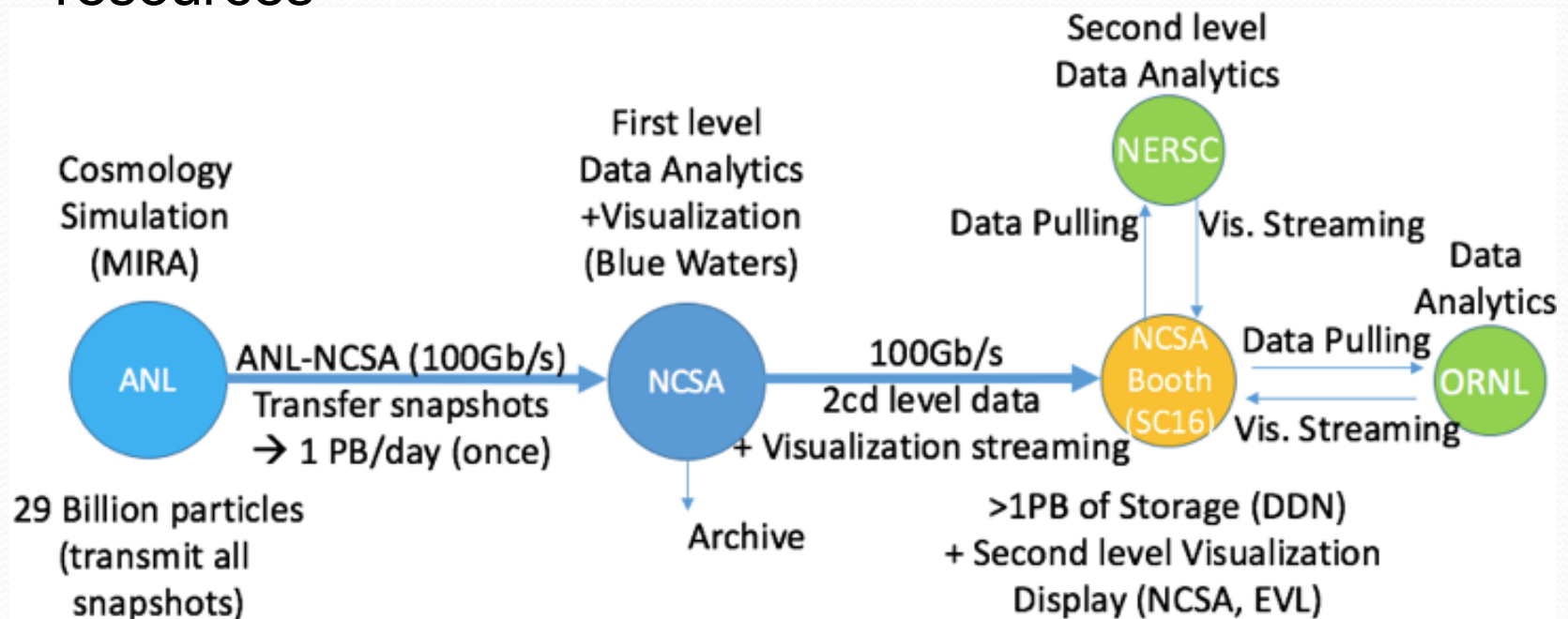
- Scientific instruments are specialized
- Some systems are better for simulation than data analytics (BlueWaters is a wonderful platform for data analytics). The opposite is also true.
- HPC Centers may not have both (ANL does not have a system like BlueWaters for data analytics)
- Data centers & Clouds designed for storage and access (not the priority of scientific instruments and HPC centers)
- The end of Moore's law may accelerate this specialization



ANL-NCSA SC16 experiment:

On demand infrastructure for data analytic and storage

- Objectives:
 - 1) Cosmology simulation **and analysis at full resolution**
 - 2) **Share the data** with other sites
 - Need to produce and analyze **all snapshots**
 - Need to create a virtual infrastructure of complementary resources



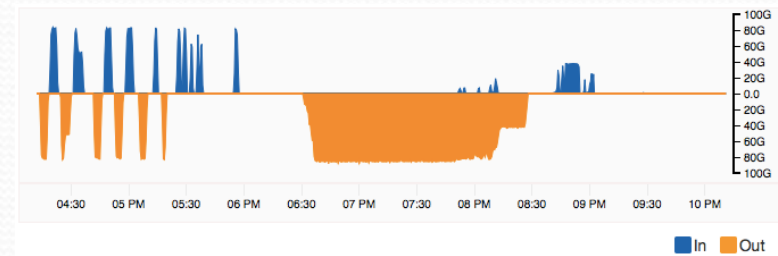
On demand infrastructure: Challenges

1) Simulation: Produce all snapshots

- could not be done before
- Snapshots transferred as soon as produced to BW (Orchestration)

2) Transmit data between remote sites at the rate of 1PB/day (~93Gbps sustained)

- Was done before with dedicated resources (requires Coordinated multi-node data movement: GridFTP)
- In our case: network path can be reserved but storage is shared by both compute nodes and data transfer nodes – e.g, NCSA, Argonne)



3) Storage: Build a self contained (Embedded), scalable Data Transfer Node (DTN)

- DDN will provide all the needed hardware



4) Visualization from all snapshots at full resolution

- Could not be done before
- Enable the analysis of all detailed history of all structures in the simulation

On demand infrastructure: Challenges

- 1) Simulation: Produce all snapshots

Density-based visualizaion,
full volume, low resolution

Particle-based visualization,
tiny sub-volume, high resolution

- 4) Visualization from all snapshots at full resolution
 - Could not be done before
 - Enable the analysis of all detailed history of all structures in the simulation

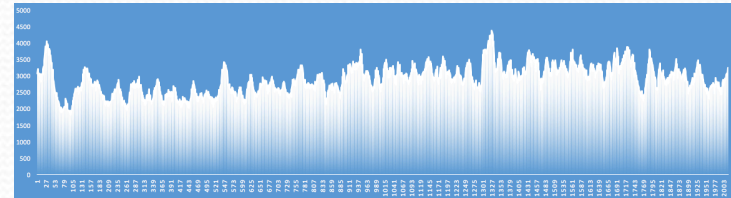
Lossy compression as a fundamental pattern (motif) of scientific computing

- Lossy compression: used in every domain where data cannot be communicated and stored entirely: Photos, videos, audio files, Medical imaging, etc.
- Compression is one aspect of data reduction (complementary)
- Compression is a fundamental motif of scientific computing
 - Simulations and experiments produce approximations
 - Lossy compression is another layer of approximation
 - It changes the initial data
 - It can be done in parallel
 - It has overhead (computational, communication, memory)
- Lossy compression for scientific data is still in its infancy
 - Only 12 papers on that topic in 26 years of IEEE DCC conference
 - Hard to compress data sets (compression factor of 3-5)
 - Few lossy compressors have parallel implementations

Lossy compression: Challenges

1) improve compression factor for hard to compress datasets (we do not understand them)

- Example: APS dataset



1) What can we do/don't with it?

- Compress data before analytics?
- before long term storage?
- for checkpoint/restart?
- Compress communications?

2) How do we use it?

- Can we perform data analytics directly on the compressed version of the dataset?
- Do we need to decompress? If yes, can we pipeline?



**Barcelona
Supercomputing
Center**

Centro Nacional de Supercomputación



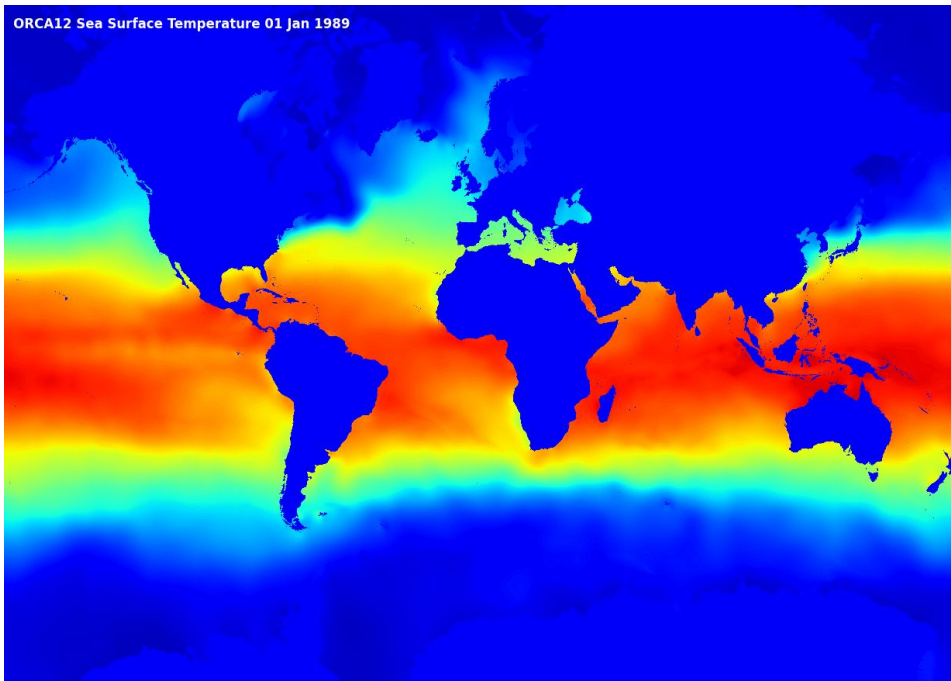
EXCELENCIA
SEVERO
OCHOA

Big Data for climate and air quality

BDEC 4th workshop, 15-17 June 2016, Frankfurt

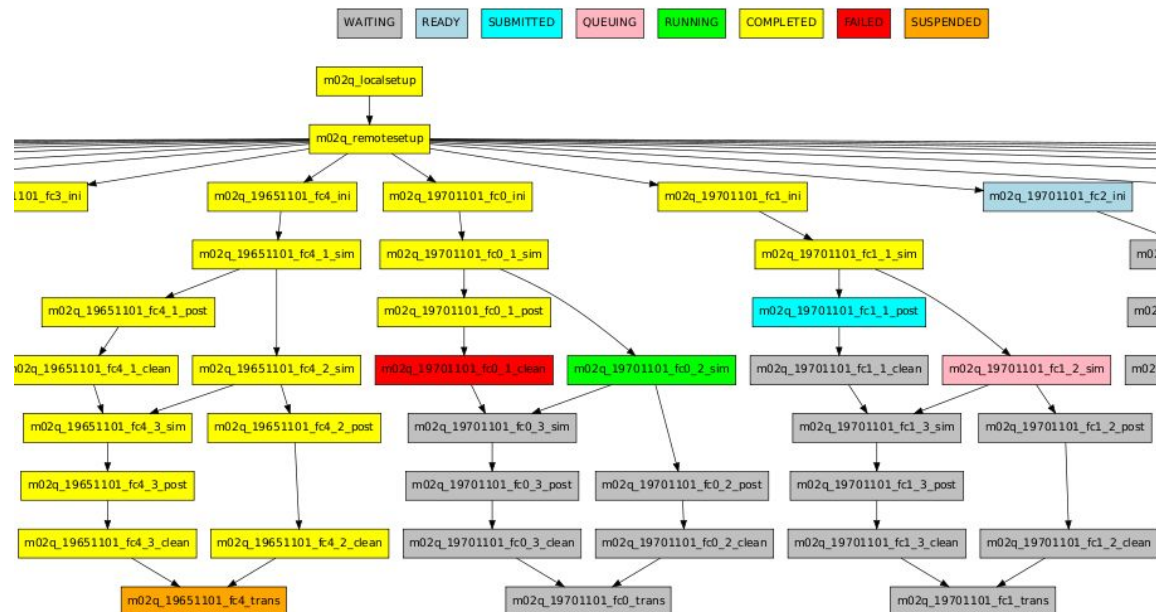
Francesco Benincasa
BSC Earth Sciences Department



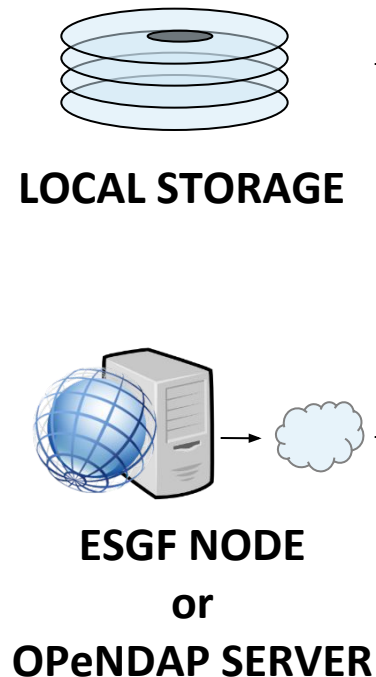


- **Automatisation:** Preparing and running, post-processing and output transfer, all managed by Autosubmit. No user intervention needed.
- **Provenance:** Assigns unique identifiers to each experiment and stores metadata about model version, configuration options, etc
- **Failure tolerance:** Automatic retrials and ability to repeat tasks in case of corrupted or missing data.
- **Versatility:** Currently run EC-Earth, NEMO and NMMB/BSC models on several platforms.

Workflow of an experiment monitored with Autosubmit (yellow = completed, green = running, red = failed, ...)



S2dverification is an R package to verify seasonal to **decadal** forecasts by comparing experimental data with observational data. It allows analysing data available either locally or remotely. It can also be used online as the model runs.



s2dverification package

- Supports datasets stored locally or in ESGF (OPeNDAP) servers.
- Exploits multi-core capabilities
- Collects observational and experimental datasets stored in multiple conventions:
 - NetCDF3, NetCDF4
 - File per member, file per starting date, single file, ...
 - Supports specific folder and file naming conventions.

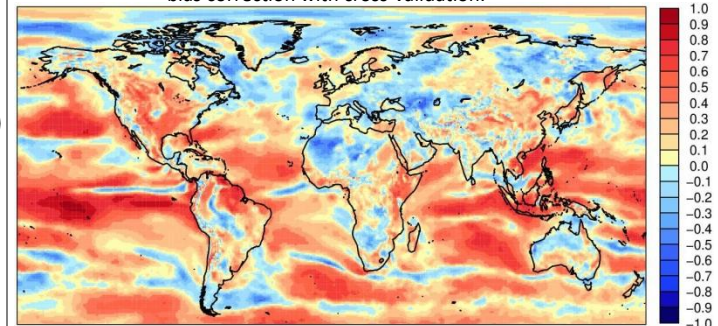
BASIC STATISTICS

SCORES

Correlation, ACC, RMSSS, CRPS, ...

PLOTS

Anomaly Correlation Coefficient. 10M Wind Speed ECMWF S4 1 month lead with start dates once a year on first of November and Era-Interim in DJF from 1981 to 2011. Simple bias correction with cross-validation.

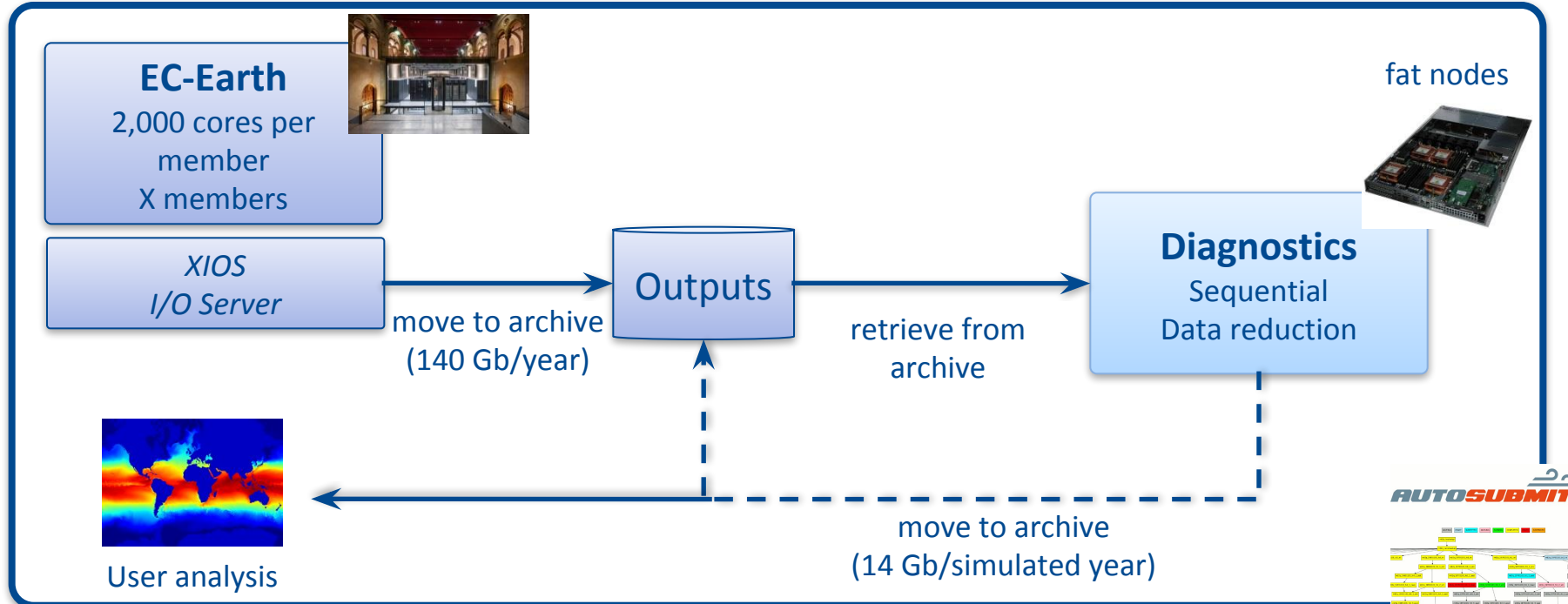


Current workflow for diagnostics



Barcelona
Supercomputing
Center
Centro Nacional de Supercomputación

EXCELENCIA
SEVERO
OCHOA



XIOS

- XIOS is an open source C++ I/O server widely used by the climate community
- XIOS is already integrated in NEMO and will be integrated in OpenIFS
- The diagnostics should be computed at the XIOS level
- Unfortunately, XIOS does not compute diagnostics yet

Drawbacks

- Diagnostics only computed offline (after model runs)
- High level of data traffic
- Fat nodes are required
- Delays on making significant data to the user

Proposed workflow for diagnostics



EC-Earth

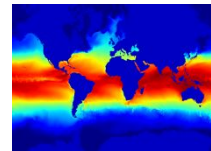
2,000 cores per
member
X members

XIOS
I/O Server

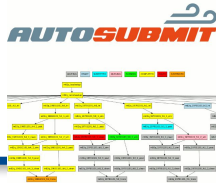
Diagnostics
computed as
AaaS

Outputs

move to
archive



User analysis



XIOS could be modified to add a layer of Analytics as a Service
(based in PyCOMPSs/COMPSs)

- Diagnostics online (during model run)
- Reduced data traffic
- Diagnostics possible on the computing nodes
- New diagnostics (data mining of extremes) possible
- The user gets the results faster



**Barcelona
Supercomputing
Center**

Centro Nacional de Supercomputación



Thank you!

francesco.benincasa@bsc.es



**Barcelona
Supercomputing
Center**

Centro Nacional de Supercomputación

Enablement of multi-scale simulation, analytics and visualization workflows

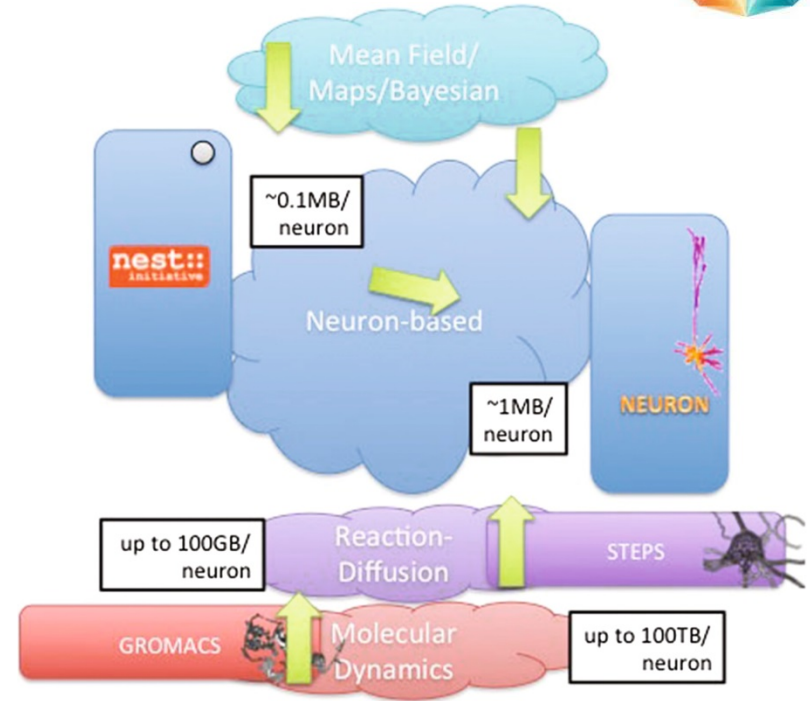
**Marc Casas, Miquel Moreto, Rosa M Badia, Javier Conejero,
Raul Sirvent, Eduard Ayguadé, Jesús Labarta, Mateo Valero**

16th June 2016

Multi-scale simulation



- Simulation of large and complex systems is still a challenge and one of the applications that will require exascale computing
- Multi-scale simulators compose simulators at different levels of granularity (detail), from coarser to finer grains, switching between them whenever necessary in order to attain the required accuracy
- At BSC, we propose the use of PyCOMPSs/COMPSs to orchestrate multi-scale simulations at HBP



* Lippert et al, "Supercomputing Infrastructure for Simulations of the Human Brain", chart courtesy of Felix Schürmann

PyCOMPSs/COMPSs

⌘ Programmatic workflows

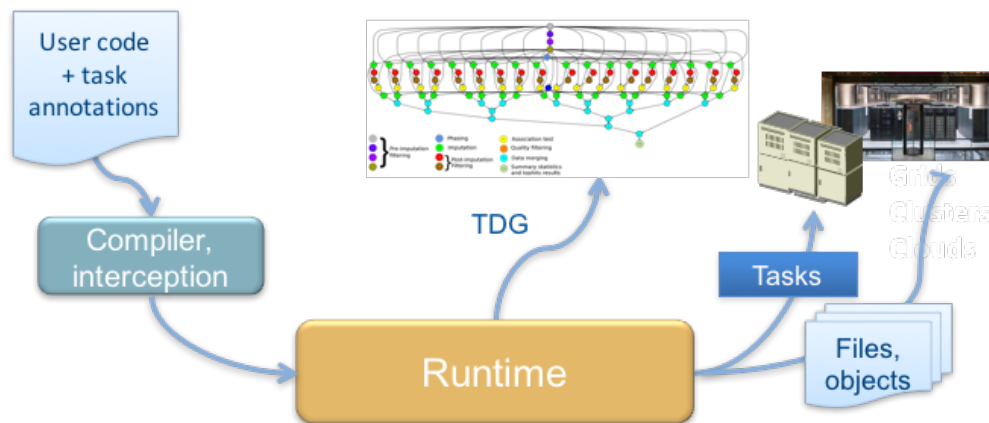
- Standard sequential coordination scripts and applications in Python or Java
- Incremental changes: Task annotations + directionality hints

⌘ Runtime

- DAG generation based on data dependences: files and objects
- Tasks and objects offload

⌘ Platform agnostic

- Clusters
- Clouds, distributed computing



Implementing multi-scale simulations with PyCOMPSs/COMPSs

- ❧ Each node of the task-graph becomes an instance of one of the required simulators
- ❧ PyCOMPSs enables the coupling of different simulators, each of them possibly parallelized with MPI or MPI+X
 - Possibly offloading computation to accelerators
- ❧ PyCOMPSs runtime will orchestrate the execution of the multiscale simulation
 - Deciding when each simulator should be invoked
 - Enabling the exchange of data between different simulators
- ❧ Each simulator will advance a number of time-steps during each invocation and then stop until it is invoked again
- ❧ Features required:
 - Support for hierarchy in the workflows
 - Support for parallel tasks: a task can be PyCOMPSs, MPI, OpenMP, ...
 - Support for persistency data in the tasks

Implementing multi-scale simulations with PyCOMPSs/COMPSs

@task

def doctor (conductivity):

Evaluate simulation

return status, medicine

Regular task

@service

def brainSimulator (conductivity, temperature):

perform a brain simulation

return brainActivity

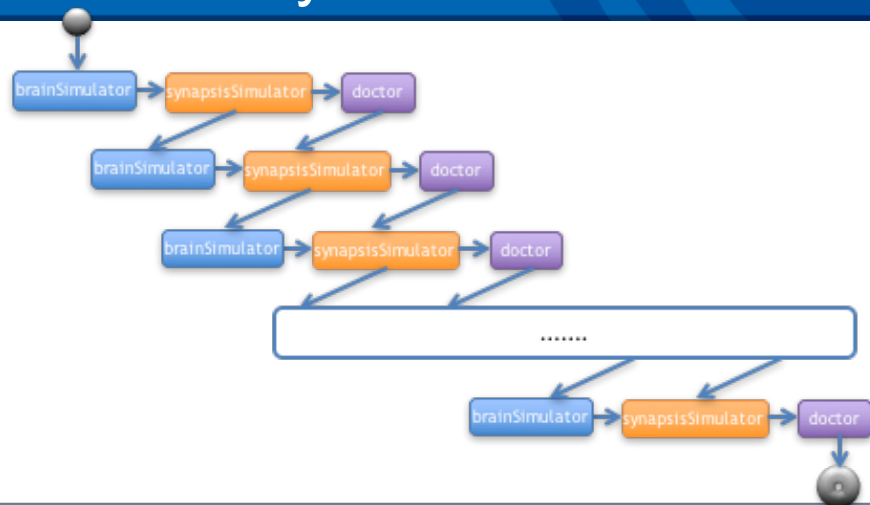
@service

def synopsisSimulator (brainActivity):

perform a synapse simulation

return conductivity

Stateful tasks: able to keep the state/initialized data between invocations



declare service brain

declare service synopsis

Loop:

temp = load (temperature) # Possible persistent storage access.

brainActivity = brainSimulator (conductivity, temp)

conductivity = synopsisSimulator (brainActivity, medicine)

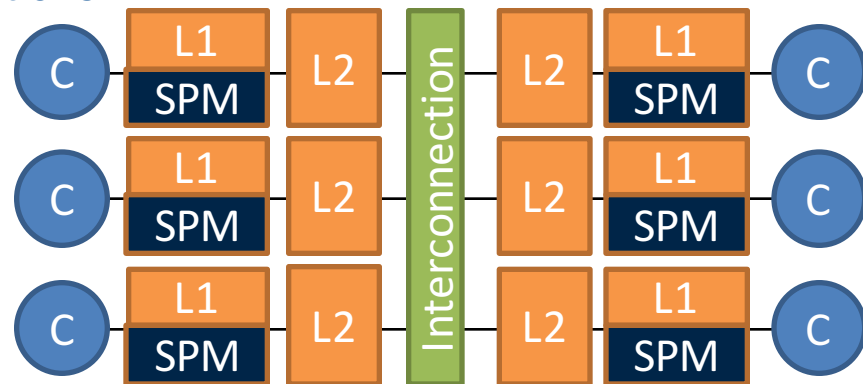
status, medicine = doctor (conductivity)

if status == 'healthy':

return medicine

New storage and memory

- ❧ Stateful tasks require new storage solutions
 - dataClay, Hecuba
- ❧ Requirements on memory of multi-scale simulations and others → 100 PB, sustained 100 PB/S
- ❧ Not achievable with regular RAM
 - Use of NVM memories, hybrid or global
- ❧ Hybrid memory hierarchies of scratchpad and cache storage
 - Partially or totally managed by the runtime system
 - Reduced power consumption
- ❧ Runtime system is in charge of mapping data specified by the programmer to the scratchpad devices
 - Use of task-based annotations
 - Rest of memory accesses served by the L1 cache.
- ❧ That same approach can be taken to the next level
 - Simulation workloads in machines with hybrid memory subsystems combining DRAM and NVM.





**Barcelona
Supercomputing
Center**

Centro Nacional de Supercomputación

Thank you!

Toward large scale distributed experiments for climate change data analytics in the Earth System Grid Federation (ESGF) eco-system

S. Fiore¹, D. Williams², V. Anantharaj³, S. Joussaume⁴, D. Salomoni⁵, S. Requena⁶, G. Aloisio^{1,7}

¹ Euro-Mediterranean Center on Climate Change Foundation, Italy and ENES

² Lawrence Livermore National Laboratory, Livermore, California, USA

³ Oak Ridge National Laboratory, Oak Ridge, Tennessee, USA

⁴ CNRS, France and ENES

⁵ INFN Division CNAF (Bologna), Italy

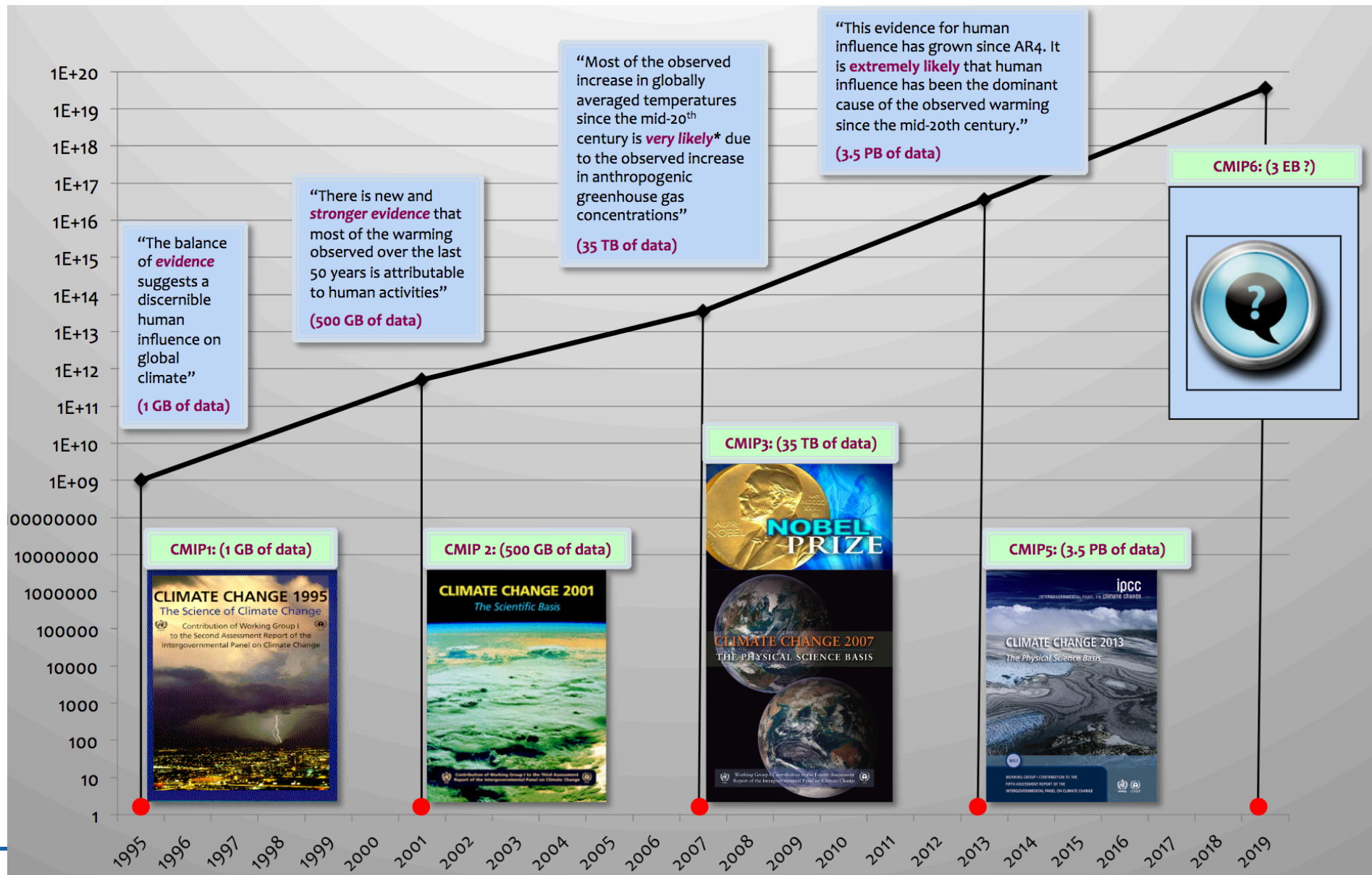
⁶ GENCI, France

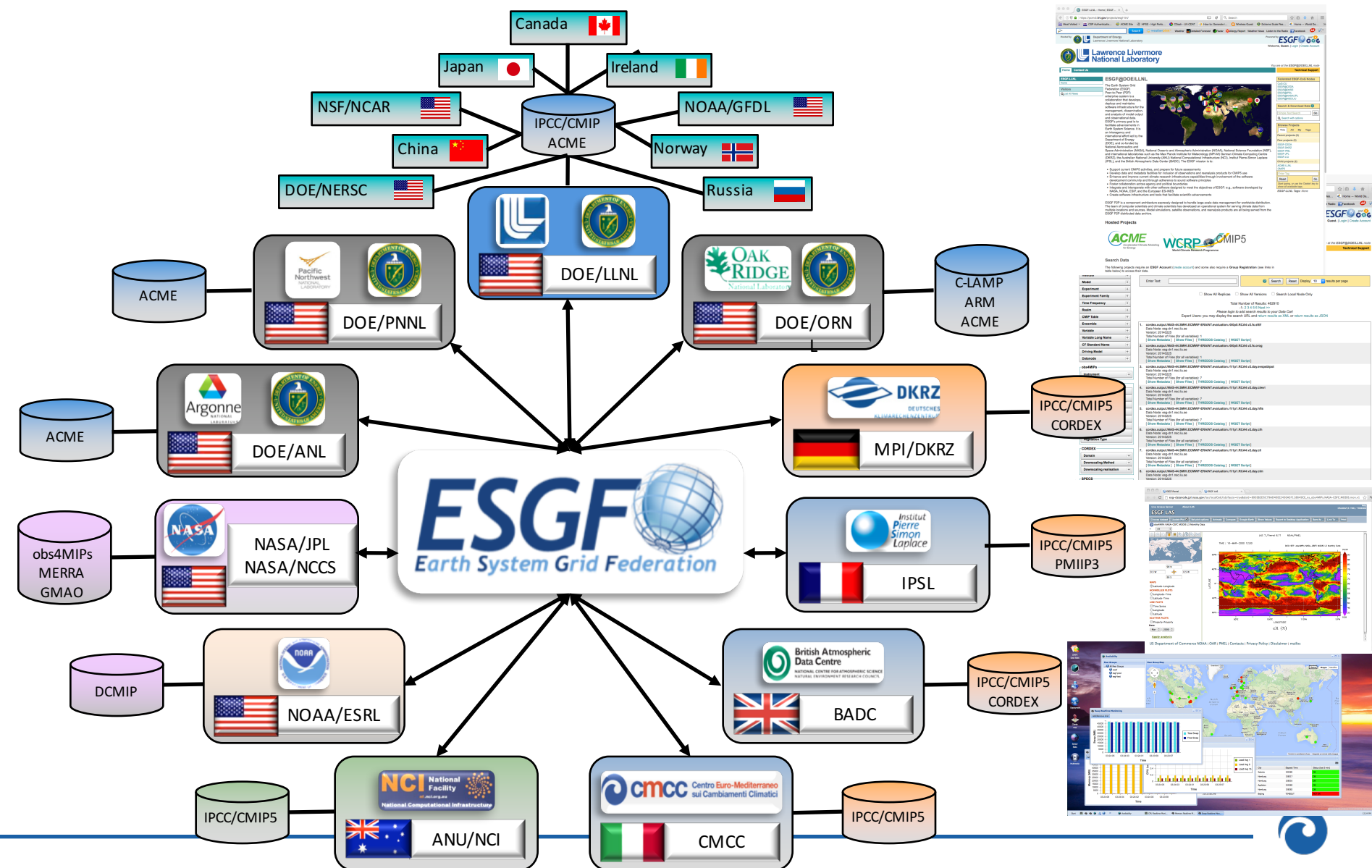
⁷ University of Salento, Italy

4th BDEC closed meeting - Frankfurt June 16-17, 2016



CMIP data history: a global community effort





Key issues and challenges regarding climate data analysis

- ESGF provides a large-scale, federated, data-sharing infrastructure
 - client-side and sequential nature of the current approach
 - The setup of a data analysis experiment requires that all the needed climate datasets must be downloaded from the related ESGF data nodes on the end-user's local machine.
 - for multi-model experiments data download can take a significant amount of time (weeks!)
- The complexity of the data analysis process itself leads to the need for end-to-end workflow support solution
 - analysing large datasets involves running tens/hundreds of analytics operators in a coordinated fashion.
 - Current approaches (mostly based on bash-like scripts) requires climate scientists to take care of, implement and replicate workflow-like control logic aspects in their scripts (which are error-prone too) along with the expected application-level part.
- The large volumes of data pose additional challenges related to performance, which requires substantial co-design efforts (e.g. at the storage level) to address current issues.



Centre of
Excellence in **S**imulation of **W**eather and **C**limate in **E**urope



A paradigm shift for data analysis to face the exabyte era

- A different approach based on (i) data-intensive facilities running high-performance analytics frameworks jointly with (ii) server-side analysis capabilities, should to be explored.
- Data intensive facilities close to the different storage hierarchies will be needed to address high-performance scientific data management.
 - parallel applications and frameworks for big data analysis should provide a new generation of “tools” for climate scientists.
- Server-side approaches will intrinsically and drastically reduce data movement; moreover...
 - download will only relate to the final results of an analysis
 - the geographic datasets distribution will require specific tools or frameworks to orchestrate multi-site experiments
 - they will foster re-usability (of data, final/intermediate products, workflows, sessions, etc.) as well as collaborative experiments
 - Need for interoperability efforts toward highly interoperable tools/environments for climate data analysis
 - Research Data Alliance (RDA) and ESGF are already working on these topics.
- In such a landscape, joining HPC and big data and cloud technologies could help on deploying in a flexible and dynamic manner analytics applications/tools enabling highly scalable and elastic scenarios in both private clouds and cluster environments.

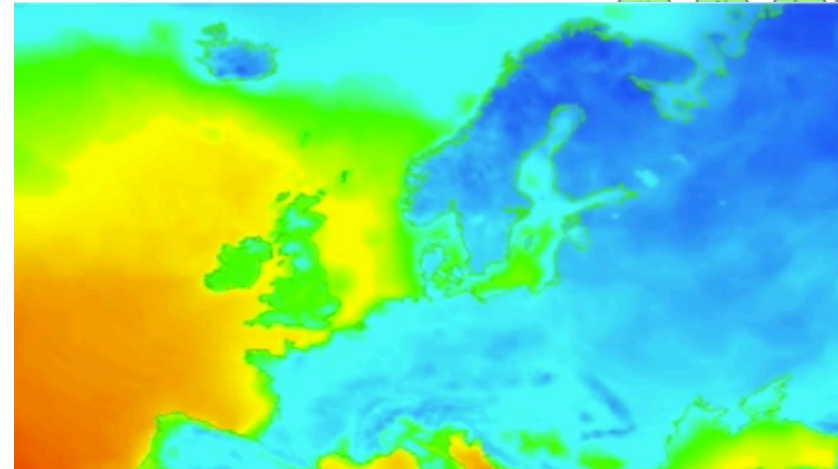
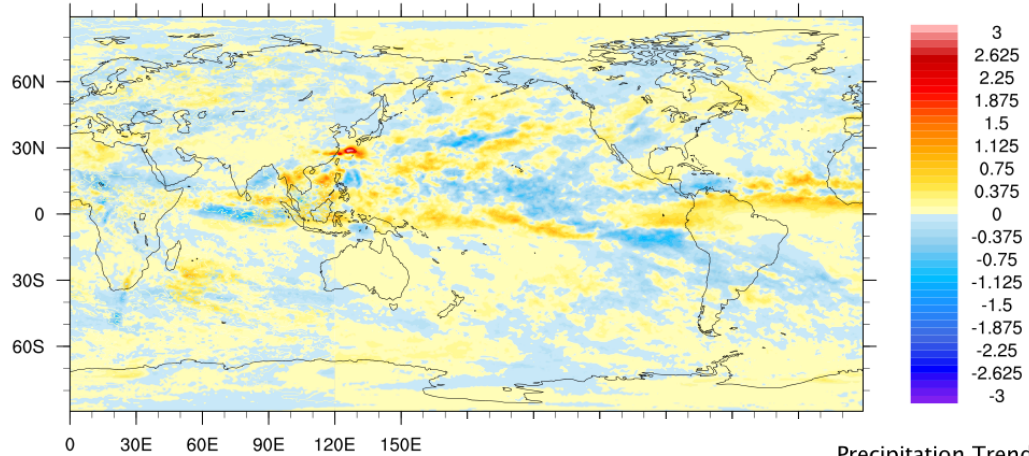


Related initiatives and projects

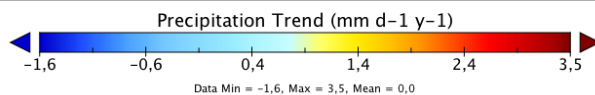
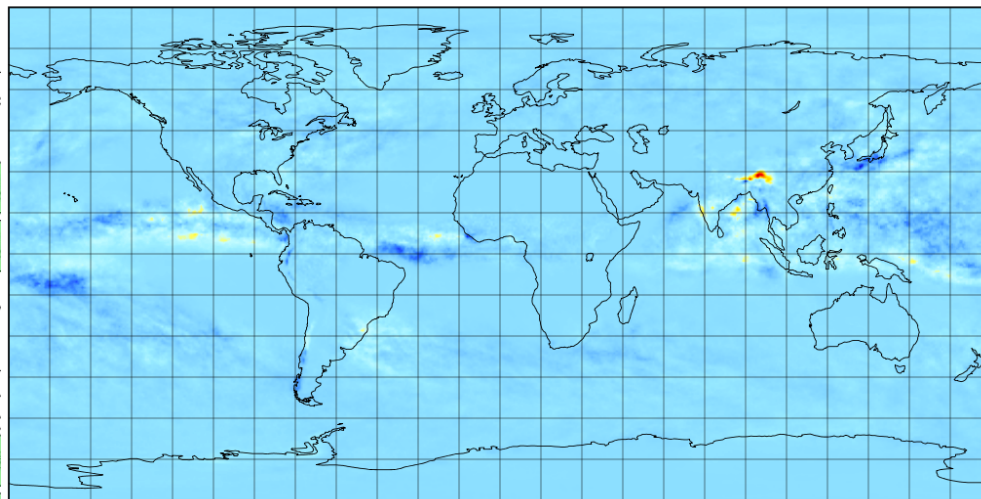
Precipitation Trend Analysis

Precipitation Trend

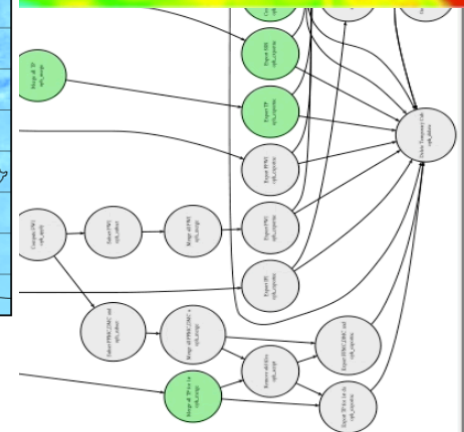
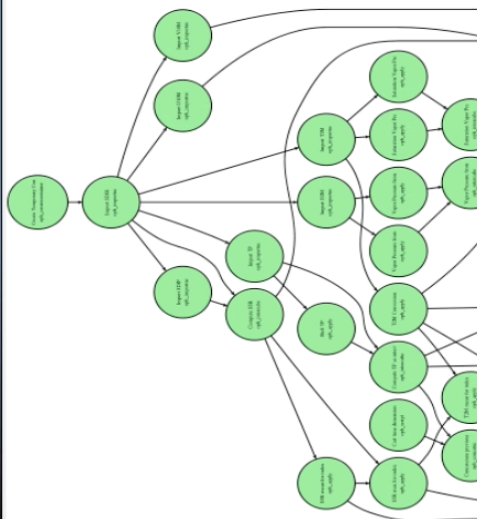
mm d-1 y-1



Precipitation Trend

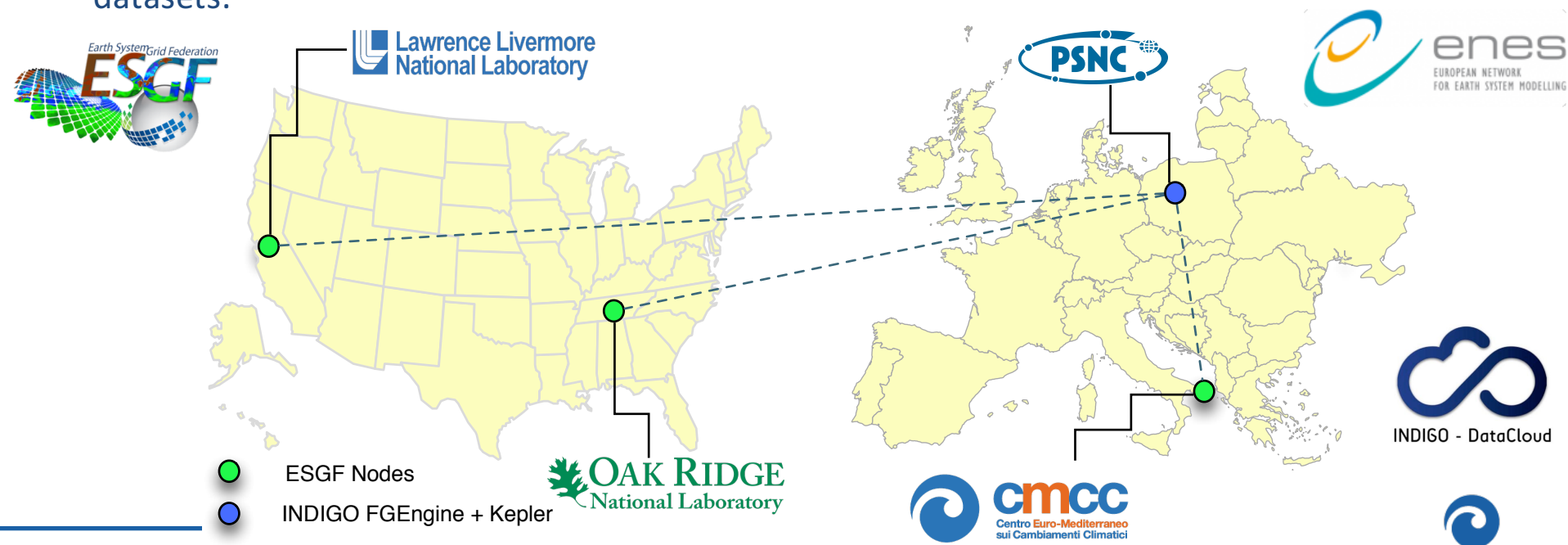


Data Min = -1,6, Max = 3,5, Mean = 0,0

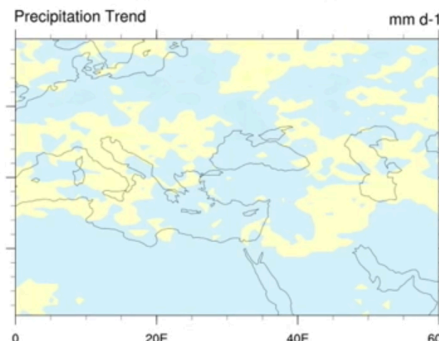


A real case study on multi-model climate data analysis

- In the context of the EU H2020 INDIGO-DataCloud project, a use case on climate models intercomparison data analysis is being implemented
- The use case relates to three classes of experiments for multi-model climate data analysis which require the access to one or more ESGF data repositories as well as running complex analytics workflows with multiple operators
- A geographically distributed testbed involving three ESGF sites (LLNL, ORNL and CMCC) represents the test environment for the proposed solution that is being applied on CMIP5 datasets.



-



Thanks





sage ¹ (sāj)

n.

One venerated for experience, judgment, and wisdom.

adj. **sag·er**, **sag·est**

1. Having or exhibiting wisdom and calm judgment.

2. Proceeding from or marked by wisdom and calm judgment: *sage advice*.

Percipient StorAGE for Exascale Data Centric Computing

Malcolm Muggeridge(Seagate)
BDEC Workshop, Frankfurt, June 2016

Per-cip-i-ent (pr-sp-nt)

Adj.

Having the power of perceiving, especially perceiving keenly and readily.

n.

One that perceives.

*The material presented reflects the
presenters view point and may not
represent the views of the European
Commission*

This project has received funding from the European Union's Horizon 2020
research and innovation programme under grant agreement No 671500





Atos



allinea



www.sagestorage.eu

ISC Booth #1340

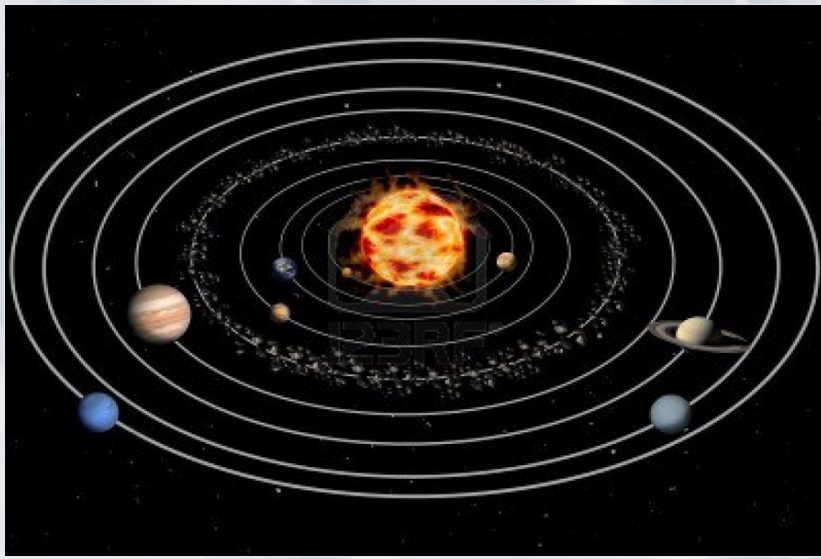
SAGE will validate a BDEC storage platform by 2018

Project Co-ordinated by Seagate

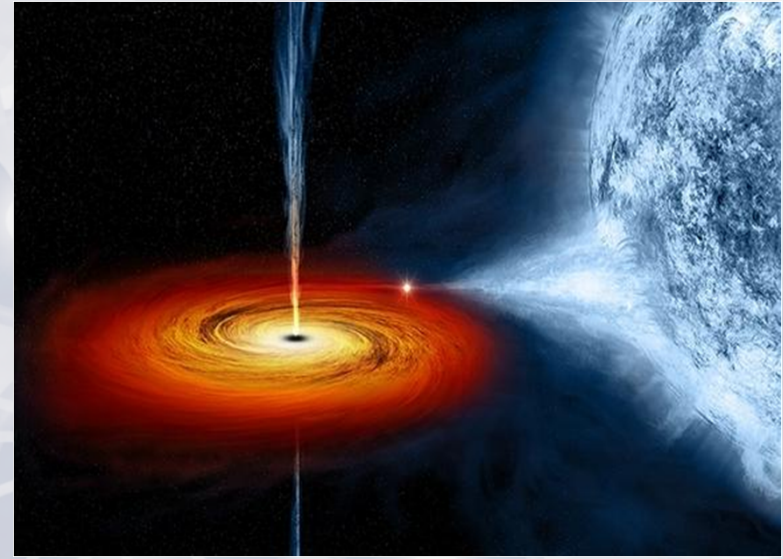
SAGE aims to lay the foundation for future Extreme Scale/BDEC Storage Platforms



SAGE



The Old Paradigm
of Storage &
Computing



The SAGE Paradigm

“PERCIPIENCE”

*Very Tightly Coupled Data &
Computation*

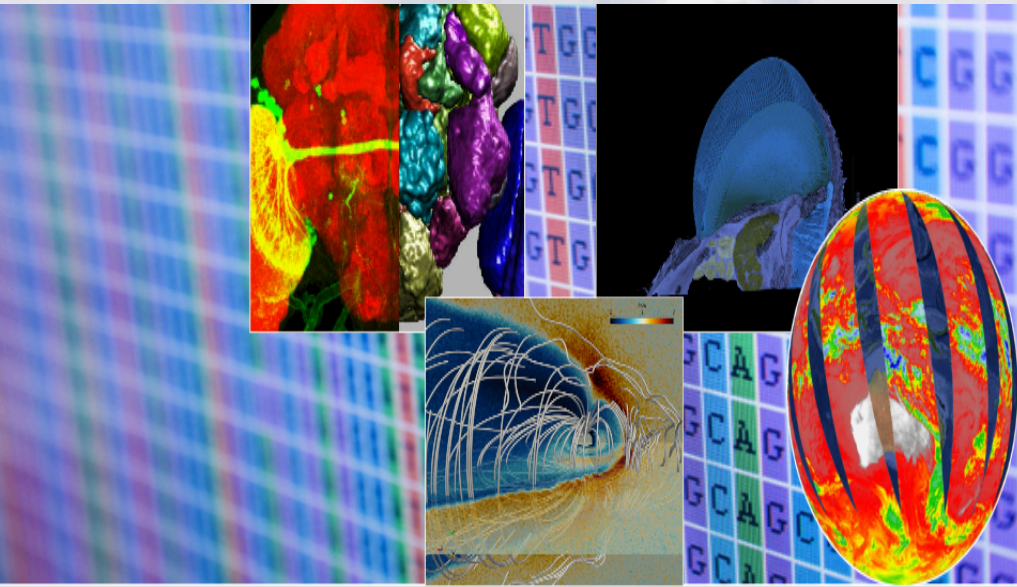


SAGE

Building A Storage System for BDEC



SAGE: Areas of Research



Co-Design with Use cases:

- Visualization
- Satellite Data Processing
- Bio-Informatics
- Space Weather
- Nuclear Fusion (ITER)
- Synchrotron Experiments

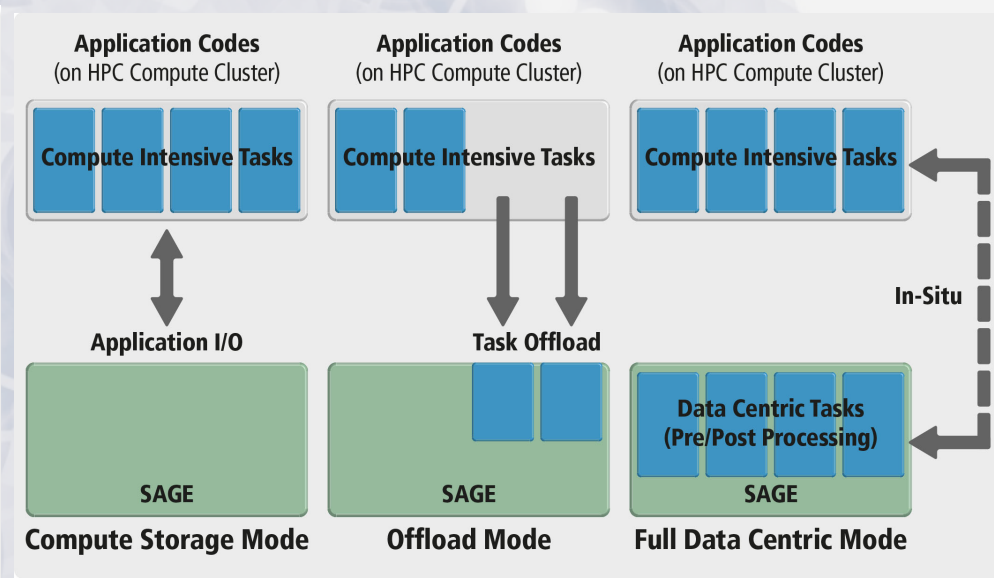
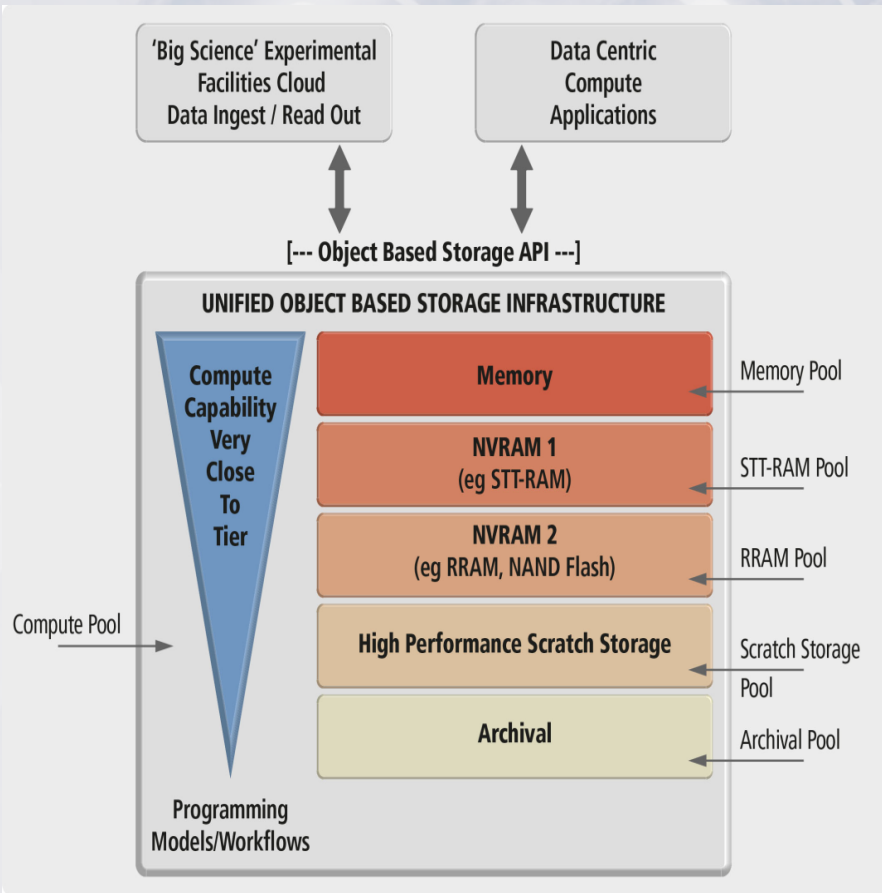
Validation at Juelich Supercomputing Center

*Growing HPDA/Big Science Requirement:
Simulation & Big Data Analysis as part of the same
workflow*



SAGE

SAGE: Co-Design/Validation with BDEC Use cases



Status

- ✓ Co -Design Activity
- ✓ Hardware Platform Definition
- ✓ Design of core software components
- ✓ Successful First EC Review

DATA INTENSIVE AND HIGH PERFORMANCE COMPUTING; AN HEP VIEW

ANSHU DUBEY, SALMAN HABIB

❑ Science in many communities needs HPC and large scale data flow and volume

❑ Need both performance and usability

❑ Examples

❑ High energy physics

❑ Light sources

❑ Biology

❑ Climate/Earth modeling

❑ Materials



BDEC 16
June 16, 2016

HEP COMPUTATIONAL REQUIREMENTS

- ❑ HEP focus on three frontiers
 - ❑ The energy frontier
 - ❑ Large experiments at colliders
 - ❑ 30PB/yr now, expected to reach 400PB/yr in a decade
 - ❑ The intensity frontier
 - ❑ Small to medium scale experiments
 - ❑ < 1PB/yr now, expected to grow to 10PB/yr in 5 yrs
 - ❑ The cosmic frontier
 - ❑ < 1PB/yr now, expected to become 10PB/yr in 10 yrs
- ❑ Experiments need support from theory => simulations with variable scale data



HEP COMPUTATIONAL CHALLENGES

- ❑ Complex data pipelines and “event” style analysis
 - ❑ Need to run many times
- ❑ Amount of I/O varies
 - ❑ In simulations data generation limited by I/O resources
 - ❑ In Energy Frontier experiments, triggers used to limit data B/W
- ❑ High throughput computing uses Grid resources in batch mode
 - ❑ Fast approaching a potential breaking point
- ❑ Edge services to handle security, resource flexibility, interaction with schedulers, external security, resource flexibility, interaction with schedulers, external databases and requirements of the user community



HEP WISH-LIST

- ❑ Software Stack
 - ❑ Ability to run arbitrarily complex software stack on demand
- ❑ Resilience
 - ❑ Ability to handle failures of job streams
- ❑ Resource flexibility
 - ❑ Ability to run complex workflows with changing computational 'width'
- ❑ Wide-area data awareness
 - ❑ Ability to seamlessly move computing to the data (and vice versa where possible); access to remote databases and data consistency
- ❑ Automated workloads
 - ❑ Ability to run automated production workflows
- ❑ End-to-end simulation-based analyses
 - ❑ Ability to run analysis workflows on simulations using a combination of in situ and offline/co-scheduling approaches



DE LA RECHERCHE À L'INDUSTRIE



BDEC Workshop June 16-17, 2016 Frankfurt

Edouard Audit

Christophe Calvin

Jean Gonnord

Jacques-Charles Lafoucrière

Jean-Philippe Nominé

www.cea.fr

.

Some observations and examples inspired by CEA experience in...

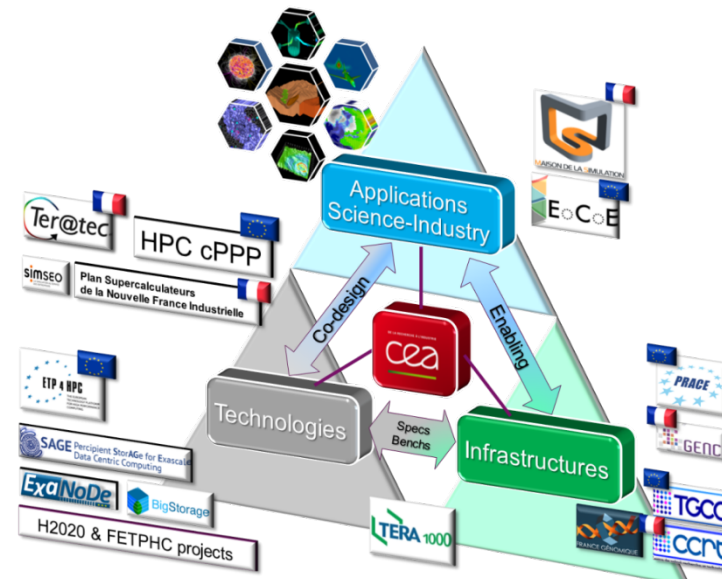
- Co-design of HPC systems with technology suppliers (first-of-a-kind TERA10/100/1000)
- Commissioning and operation of large computing infrastructures (currently 3 petascale systems – European Tier-0 CURIE 1.8 PF + CCRT cobalt 1.5 PF + TERA 2.7 PF)
- Development and usage of simulation applications in many different areas and with many different partners (research, industry) as well as for defense programmes
-
- ... with strong involvement in national and European HPC structures, programmes and initiatives

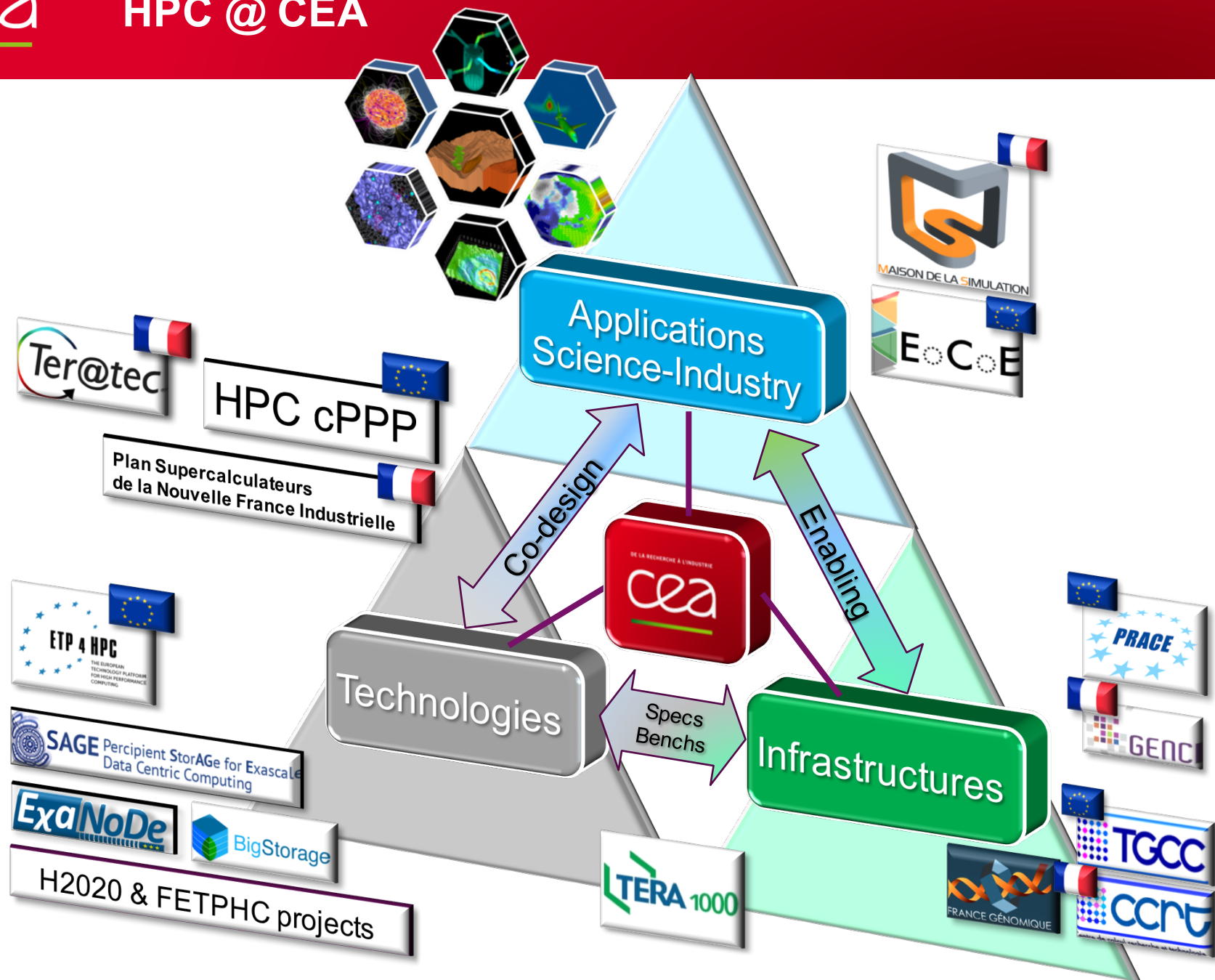
Plan d'Investissements d'Avenir / Nouvelle France Industrielle

Maison de la Simulation

Horizon 2020 (ETP4HPC and HPC PPP; FETHPC projects; Centres of Excellence; PRACE)

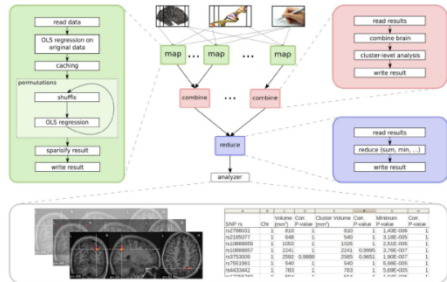
IPCEI



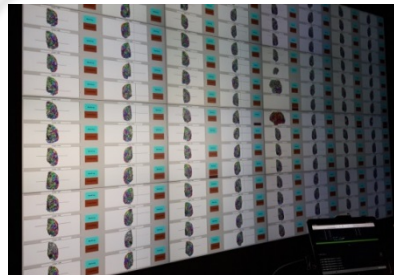


WHAT (IS CONVERGENCE)?

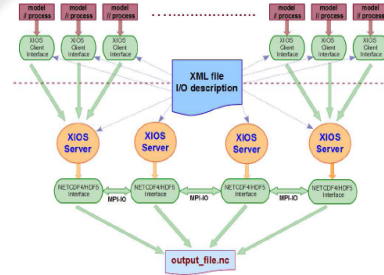
- De facto observation from the computing centre standpoint
 - ✓ More and more entangled compute/data-intensive activities
 - ✓ Sample applications: examples or forerunners of convergence
- Data flows becoming more complex / diverse / multi-directional
Actually more and more of a continuum HPC/HTC/data processing
 - ✓ Numerical simulations are data producers – but also consumers – data types becoming more diverse even in ‘conventional’ numerical applications
 - ✓ Observational and experimental sciences are rather data consumers
Data processing more and more compute-hungry... in addition to storage and network-hungry
 - ✓ Crossroads: e.g. climate (CMIP6); coupling of genomics with 3D imaging; comparative modelling
 - ✓ Computing centres operations also generate massive data (BigData analysis)



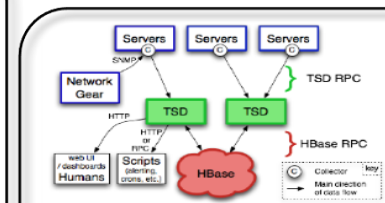
Genetic imaging – Neurospin - V. Frouin et al.
http://www.teratec.eu/library/pdf/forum/2012/presentations/A5_02_FTeratec_2012_VFrouin.pdf



Comparing numerical simulation and 3D modelling of pre-clinical brain models
Maison de la Simulation



XIOS
Y. Meurdesoif et al.
Re-engineering the whole climate I/O and data flow
<http://forge.ipsl.jussieu.fr/ioserver>

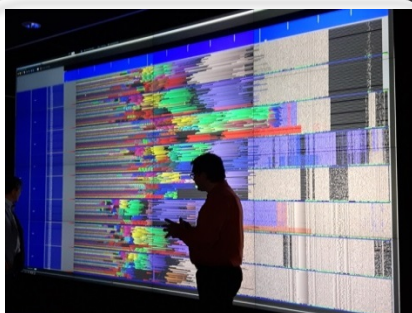


Statistics cluster
CEA/DIF/DSSI

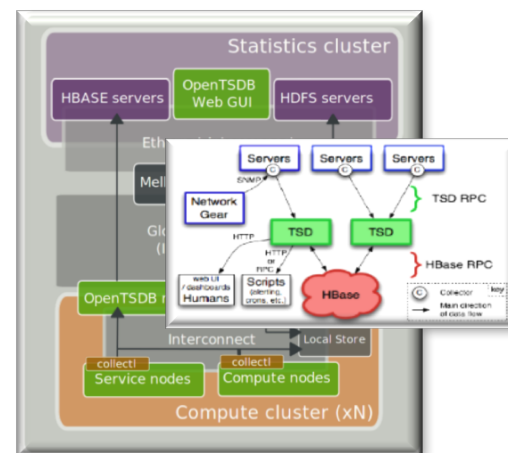
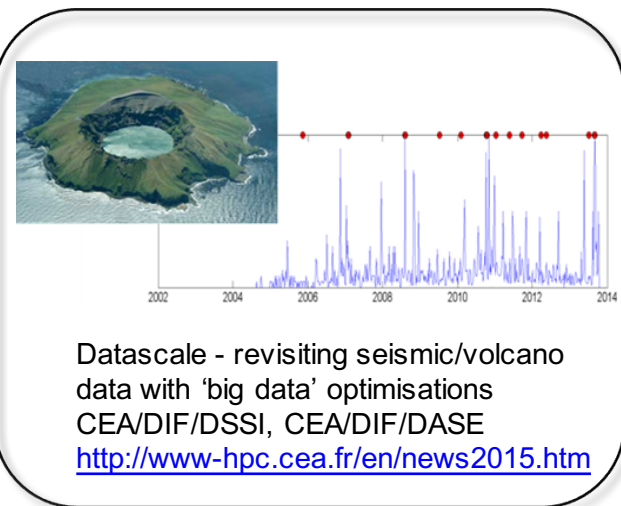
WHAT (IS CONVERGENCE)?

Some more examples....

- “Legacy” data: new science arising from data processing re-engineering / ‘big-data-style’ enhancement
- Supercomputer/datacentres and applications are themselves becoming objects of studies - producing huge amounts of introspection data! System & job logs, facility & energy monitoring...
- ✓ we now have dedicated ‘statistic clusters’ using hadoop and alike solutions + data analytics
- ✓ tricky visualisation of large data sets such as parallel traces



Large tiled display / parallel traces
Maison de la Simulation
(CEA/CNRS/INRIA et al.)



WHY (CONVERGENCE)?

- Commonalities that can be useful and beneficial, technology- infrastructure- and application-wide
- Technology (solutions = h/w + s/w)
 - ✓ HPC needs more data locality, I/O and storage efficiency
 - ✓ Current massive simulation data management may face limitations (post-posix FS needed?)
 - ✓ Data processing/analytics may need parallelism (hardware, productive programming)
- Infrastructures and services: optimise resource usage
 - ✓ Compute and storage equipment
 - ✓ (Wo)manpower and skills – developers and admins
- Applications
 - ✓ OK: big data useful for HPC & HPC useful for big data



SAGE Percipient StorAGE for Exascale
Data Centric Computing



BigStorage

PATHWAYS?

- Software easier to collaborate on than hardware
- Different possible paths / levels
 - ✓ Virtualisation
 - ✓ 'Standard' APIs or 'open interfaces', middleware
 - ✓ Potential game changers like NVRAM, 3D stacking (different compute/memory paradigms?)
 - ✓ Grasp opportunities...
- Should we distinguish Datacentre/HPC centre? Irrelevant question!
 - ✓ Difference is in resources and services offered, access and delivery modes, usage profiles (e.g. capability, HTC, data distribution&processing)
- New scientific paradigms and know-how convergence / cross-fertilisation
 - ✓ Data science + computer science

Technical convergence will happen – technology push, market pull, resource management pressure... of course not w/o efforts!

There is also a discrepancy/gap at the level of resource provisioning and usage/access models !

Equipment funding and commissioning - Capability allocations vs. elastic access to distributed data/processing...