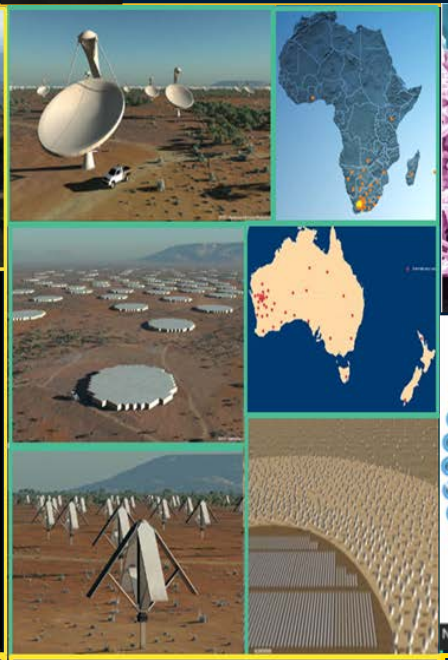
Machine Learning for the Next Generation Ecosystems with SDN: for LHC Run2 and Beyond

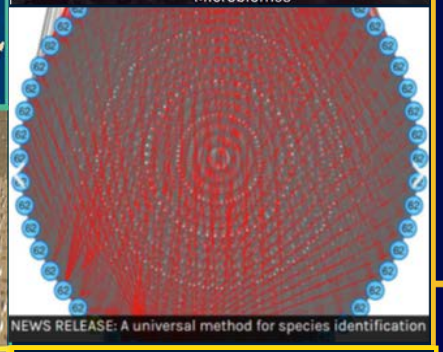
+ INDUSTRY

LHC

LSST



SKA



Joint Genome Institute

- *LHC Run2+: Beyond the Standard Model*
- *Data Intensive Exascale LCFs and SDN EcoSystems*

Gateways to a New Era

Harvey Newman, Caltech

S2I2 Workshop, May 2, 2017

https://www.dropbox.com/s/icuq1nkk3sxszmh/NGenIAES_S2I2PrincetonWorkshop_hbn050217.pptx?dl=0

Caltech Machine Learning Projects for HEP

J.R. Vlimant, M. Spiropulu et al.

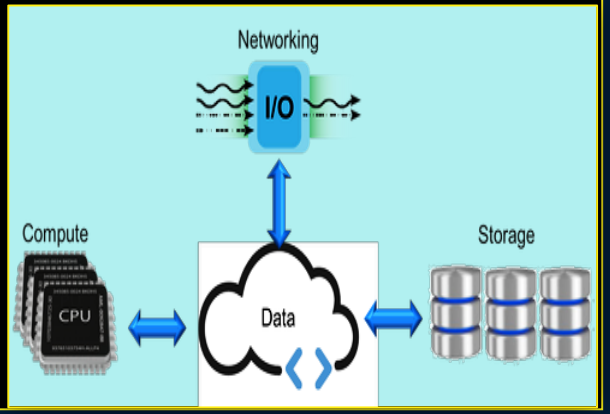
- **3D Imaging with LHC datasets :** energy regression and particle identification with 2D/3D convolutional neural nets for future generation calorimeters such as the CMS HGCal
- **Event classification using particle-level information :** use recurrent neural nets and long short term memory to learn the long range correlations in LHC collision events.
- **Charged particle tracking acceleration :** explore deep neural net methods for new ways of connecting the dots in the HL-LHC trackers and beyond.
- **Distributed learning :** accelerate training of deep neural net models over large datasets using MPI or spark frameworks.
- **Neuromorphic Hardware :** exploit existing neuromorphic systems for online data processing and event selection. Develop new hardware tailored to the characteristics of LHC data

We are also beginning to apply this knowledge to Network and Global System intent-based Operation/Management/Optimization

Consistent Operations Paradigm
Technical Implementation Directions

- **METHOD:** Construct autonomous network-resident services that dynamically interact with site-resident services, and with the experiments' principal data distribution and management tools
- To coordinate use of network, storage + compute resources, using:

1. Smart middleware to interface to SDN-orchestrated data flows over network paths with allocated bandwidth levels all the way to a set of high performance end-host data transfer nodes (DTNs),
2. Protocol agnostic traffic shaping services at the site edges and in the network core, coupled to high throughput data transfer applications that provide stable, predictable transfer rates
3. Machine learning + system modeling and Pervasive end-to-end monitoring
   - To track, diagnose and optimize system operations on the fly

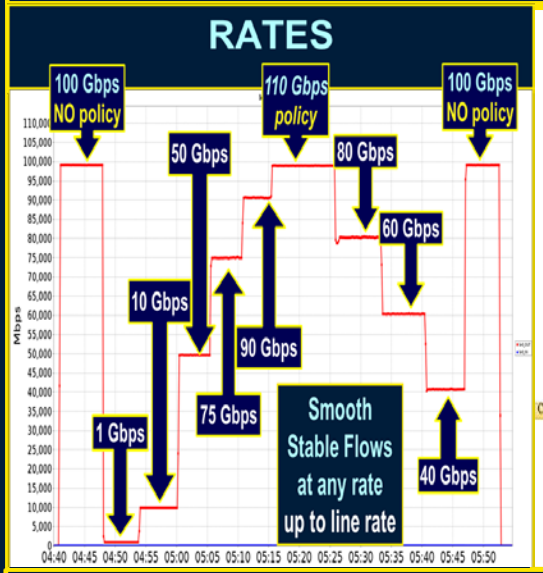


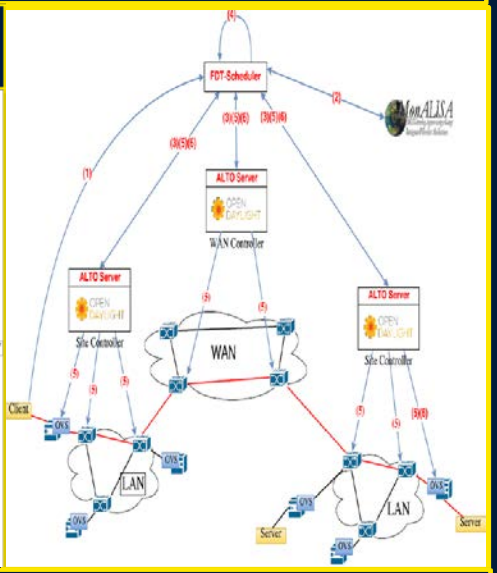
Next Generation "Consistent Operations": Site-Core Interactions for Efficient, Predictable Workflow

- Key Components: (1) Open vSwitch (OVS) at edges to stably limit flows, (2) Application Level Traffic Optimization (ALTO) in Open Daylight for end-to-end optimal path creation, flow metering and high watermarks set in the core
- Real-time flow adjustments triggered as above
- Optimization using "Min-Max Fair Resource Allocation" (MFRA) algorithms on prioritized flows
- Flow metering in the network fed back to OVS edge instances; to ensure smooth progress of flows end-to-end

Demos: Internet2 Global Summit in May
SC16 in November

Consistent Ops with ALTO, OVS and MonALISA FDT Schedulers



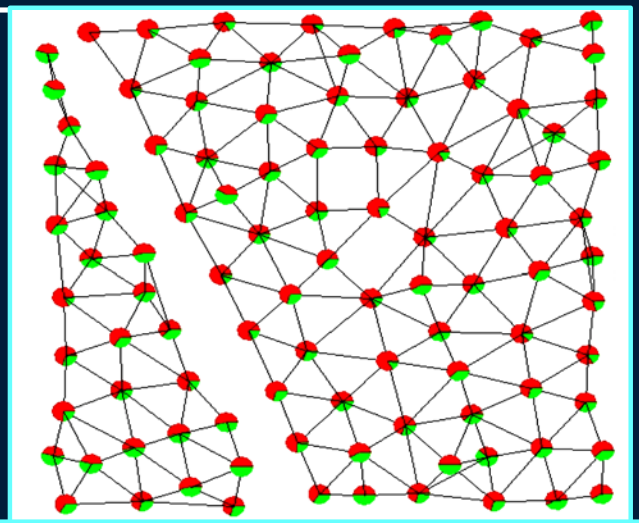


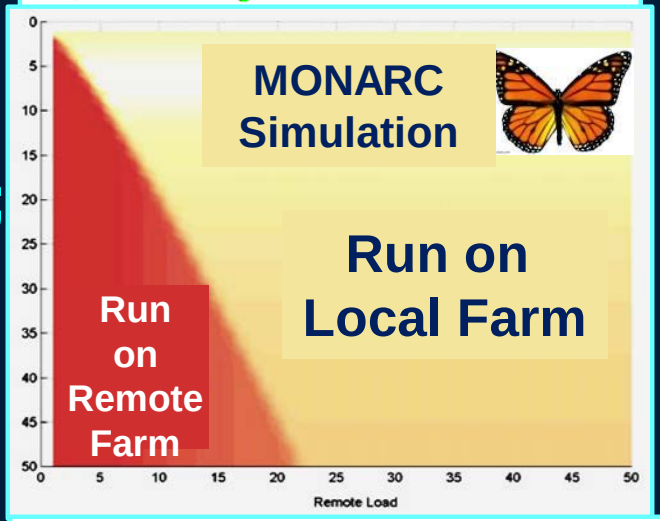
- Real-time adjustment of allocations triggered by: (1) new requests, (2) real-time feedback on progress of transfers, (3) network state changes or error conditions, (4) proactive load-balancing operations, or (5) rate-limiting operations imposed by controllers or emerging network operating systems (e.g. SENOS)

With Yale CS Team: Y. Yang, Q. Xiang et al.

Key Developments from the HEP Side: Machine Learning, Modeling, Game Theory

- Applying Deep Learning + Self-Organizing systems methods to optimize LHC workflow
  - Unsupervised: to extract the key variables and functions
  - Supervised: to derive optima
  - Iterative and model based: to find effective metrics and stable solutions [*]
  - Reinforced: according candidate metrics
- Complemented by modeling and simulation; game theory methods [*]
- Progressing to real-time agent-based pervasive monitoring
- Application to CMS Workflow





Self-organizing neural network for job scheduling in distributed systems

[*] T. Roughgarden (2005). *Selfish routing and the price of anarchy*

Game Theory and the Future of Networking

http://blog.eai.eu/game-theory-and-the-future-of-networking/

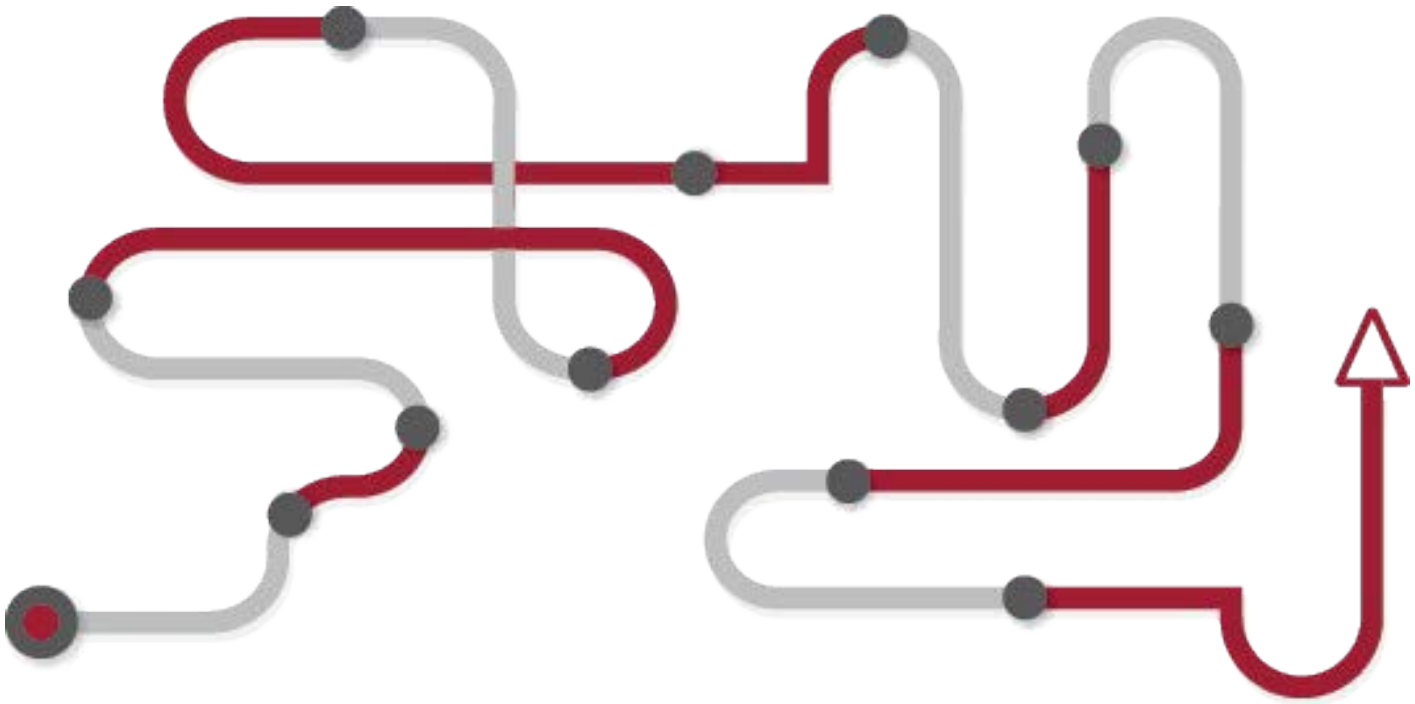
- **Game theory:** Mathematical models of conflict and cooperation among intelligent rational decision-makers
  - Studies participants' behavior in strategic situations.
- **Motive** and the need for **Increased Reach** induce selfish entities to cooperate in pursuit of a common goal
- **Application Pull:** the Internet calls for analysis and design of systems that span multiple entities with diverging information and interests
- **Technology Push:** math and science mindset of GT is similar to that of (many) scientists
- **Fields of Use:** economics, political science, psychology, logic, computer science, biology, poker… and now HEP exascale data

- **Emergence of the internet** has motivated development of **GT algorithms for finding equilibrium** in games, markets, auctions, peer-to-peer systems, security and information markets

- **GT is now applied** to a wide range of behaviors
- **It has become an umbrella term** for the science of logical decision making
- **In and among humans and computers**

- ***Coherent Interactions*** among the experiments' workflow management systems, the end sites, the network and the user groups ***as a System***

Optimizing CMS Production



Production Time of Delivery

- **Central production** of massive samples of simulated, reconstructed and digitized LHC collisions (events) **currently lacks a good estimate of the delivery time (DT) of the samples**
- **GOAL: Predict the DT of the samples**
  - Critical to impatient users
  - Critical to notice on-going issues (large EDT)
- **Technical Goals: monitor, track, predict the progress of workflows**
- **Production is a complex, time-dependent system**
  - Dozens of computer centers of various size, availability, fair-share,...
  - Tens of thousands of jobs with a range of priorities and requirements
  - **CPU, storage and network capacity and throughput vary from site to site, and over time**
- **Inputs Monitored and Recorded**
  - Global utilization of resources, by site, priority, … in time slices
  - Workflow: running and pending jobs conditions in time slices
  - Workflow completion times
    - **Method: Predicting DTs with Machine Learning,** based on Caltech HEP experience

      Train a model that predict the completion time given N time slices of (a) the global utilization, (b) workflow advancement in each slice

Motivations

- ✔ Learning to Predict the Delivery Times themselves: better estimates enable better planning
- ✓ Get experience with recording the relevant data for the computing + storage + network (CSN) optimization problem
- ✓ Get experience with building and training models on the CSN infrastructure data
- ✓ Get experience with solving this type of problem with a view towards **applying the knowledge to the more complex problem of overall production throughput optimization**

Network-Aware Strategy

Possible Improvements

- **Transfer requests are posted to the transfer system (PhEDEx)**
  - **System should allocate network resources dynamically in response to transfer requests and request parameters (priority, size, age, ...)**
    - **Reactively when there is back pressure, or feedback**
    - **Preallocating *end-to-end bandwidth* where needed**
- **Remote reads are done transparently to improve delivery time (AAA)**
  - **Remote read framework (xrootd) should dynamically allocate network resources**
  - **Reactively based on backpressure during file reads**
- **Destination sites are considered using the CPU pledge probability distribution function (pdf): mean and width**
- **Consider network link capacity, source-destination mapping to define a better pdf: larger mean and/or narrower width; while keeping resource utilization uniform**

Backup Slides Follow

Machine Learning and Understanding for Intelligent Extreme Scale Scientific Computing and Discovery

DOE Workshop Report

January 5-7, 2015
Rockville, MD

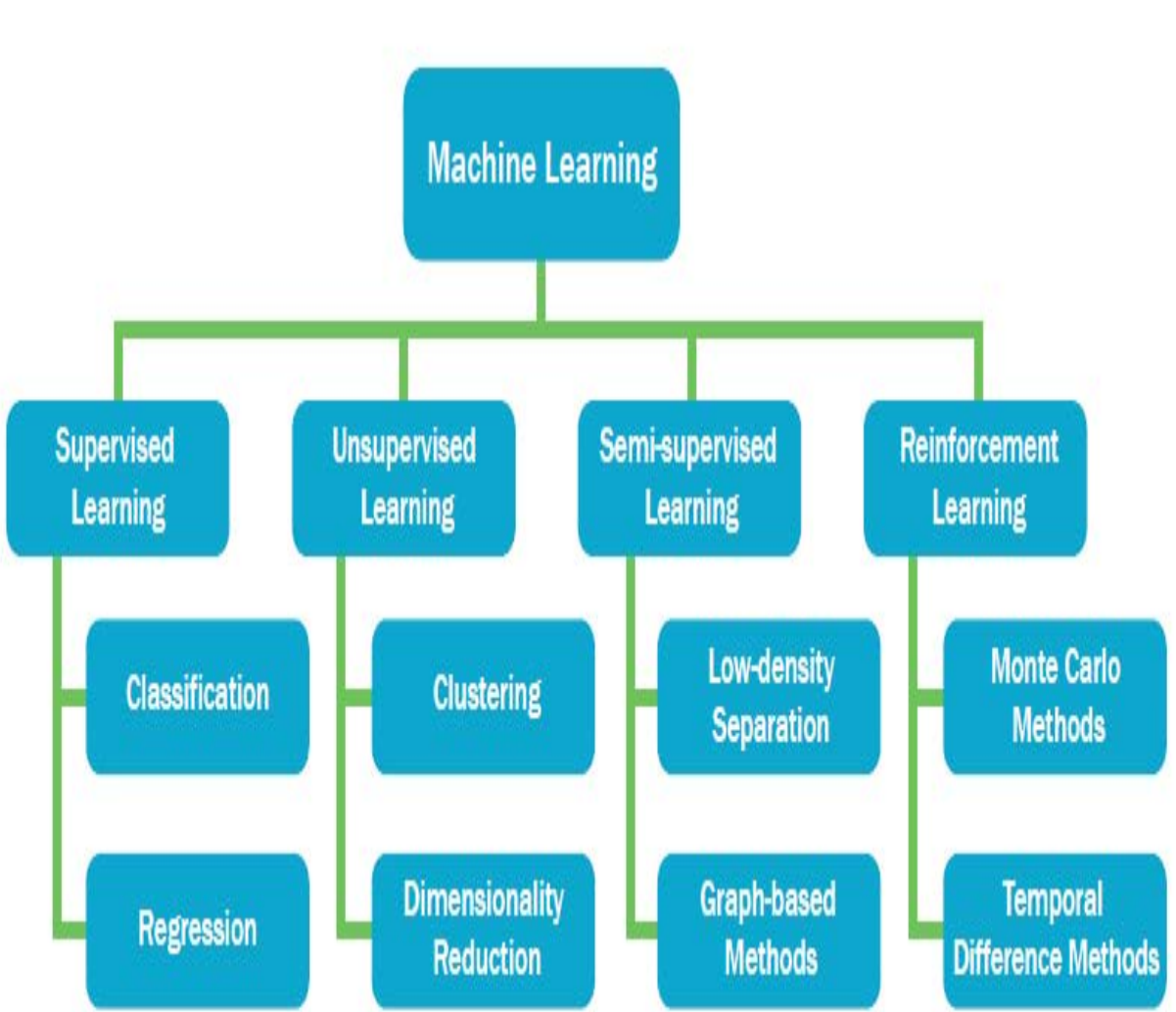
Machine Learning and Understanding for Intelligent Extreme Scale Scientific Computing and Discovery

DOE Workshop Report

January 5-7, 2015
Rockville, MD

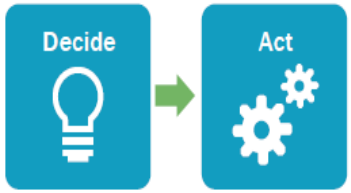
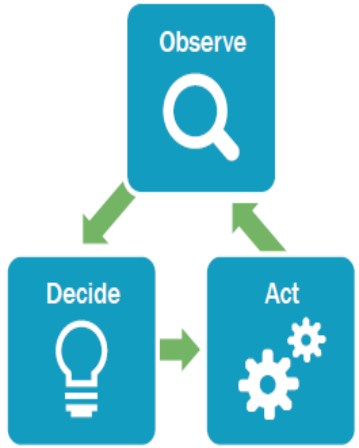
Machine learning represents a broad class of techniques that can help provide well-founded solutions to many of these exascale challenges.

We believe that machine learning is a critical technology that will further the fourth paradigm of modern scientific discovery, and complement and support other models of scientific inquiry.



Machine Learning and Understanding for Intelligent Extreme Scale Scientific Computing and Discovery

openlab.web.cern.ch

	Traditional OS/R	Self-aware OS/R
		
When Decisions Made	Design Time	Runtime
How Decisions Made	Ad-hoc, based on guesses about future	Evidence-based
Understanding User Goals	No	Yes
Optimizes For	System Metrics (Utilization)	Application Metrics (Science accomplished)
Performance	Static	Improves without user action

DOE Workshop Report

January 5-7, 2015
Rockville, MD

Data Placement Strategy

Already in place

- Heterogeneous size/event in primary input and time/event
  - Usually anti-correlated : constant size/time.
  - **Automated transfers** for 1-3 copies to production sites
- Data might be held by other groups
  - **Re-use existing copies** whenever possible
- Disk space is handled by DDM/Dynamo
  - **80% of the allowed quota** is used as operation quota for placing input, leaving enough room for output datasets
- Not all workflows can go run everywhere
  - **Pre-matching job/resource** to decide destination according to pledge CPU resource and within quota
- The more sites the better for load sharing
  - Input are **split in 4Tbyte chunks** and distributed to sites
- Simulation of LHC events requires event mixing: Overlaying a simulated signal of interest with the known backgrounds
  - **Secondary inputs are positioned automatically** according to adopted strategy

Work Assignment Strategy

Already in place

- Not all workflows can go run everywhere
  → **Pre-matching job/resource** to decide

- Simulation of LHC event overlay requires event mixing
  - → Standard mixing with **high-read rate restricts the list of sites**
  - → Pre-mixing with **lower-read read over the network (xrootd)**

- The more sites the better for load sharing
  - → Use **all sites that hold part of the input** are candidated

- Input dataset can be too large, and still need to be processed in many places
  - → Setup **reading the primary over xrootd** to sites holding full copies and their **WAN "neighbors"**

- Diversity of workflow and complexity of submission
  - → Some **job splitting tweaks** are performed upon rules

- Some large (e.g. cloud) resources might become available temporarily
  - ➢ Include the flexibility to add **specific assignment rules**