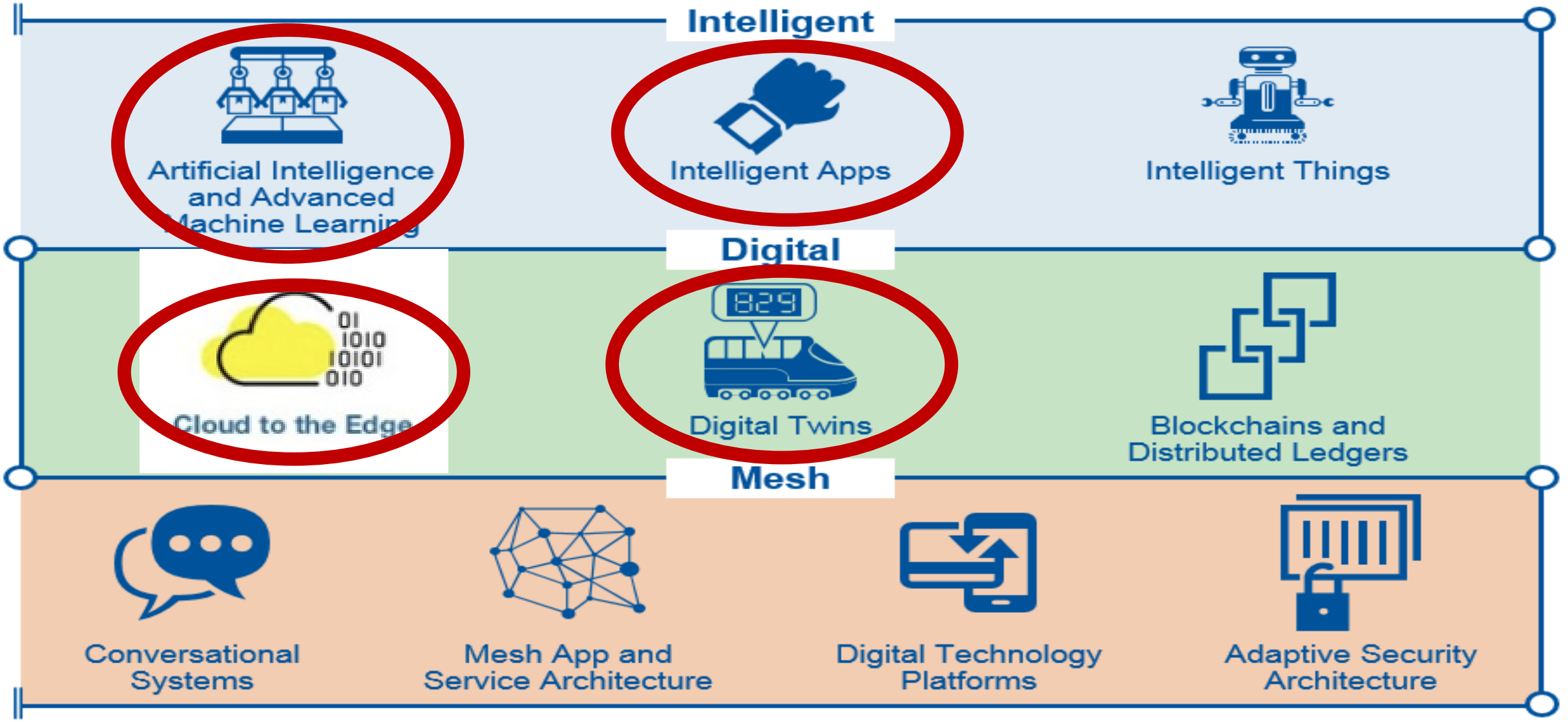
Real-Time Anomaly Detection from Edge to HPC-Cloud

September 6, 2018

Prof. Judy Qiu

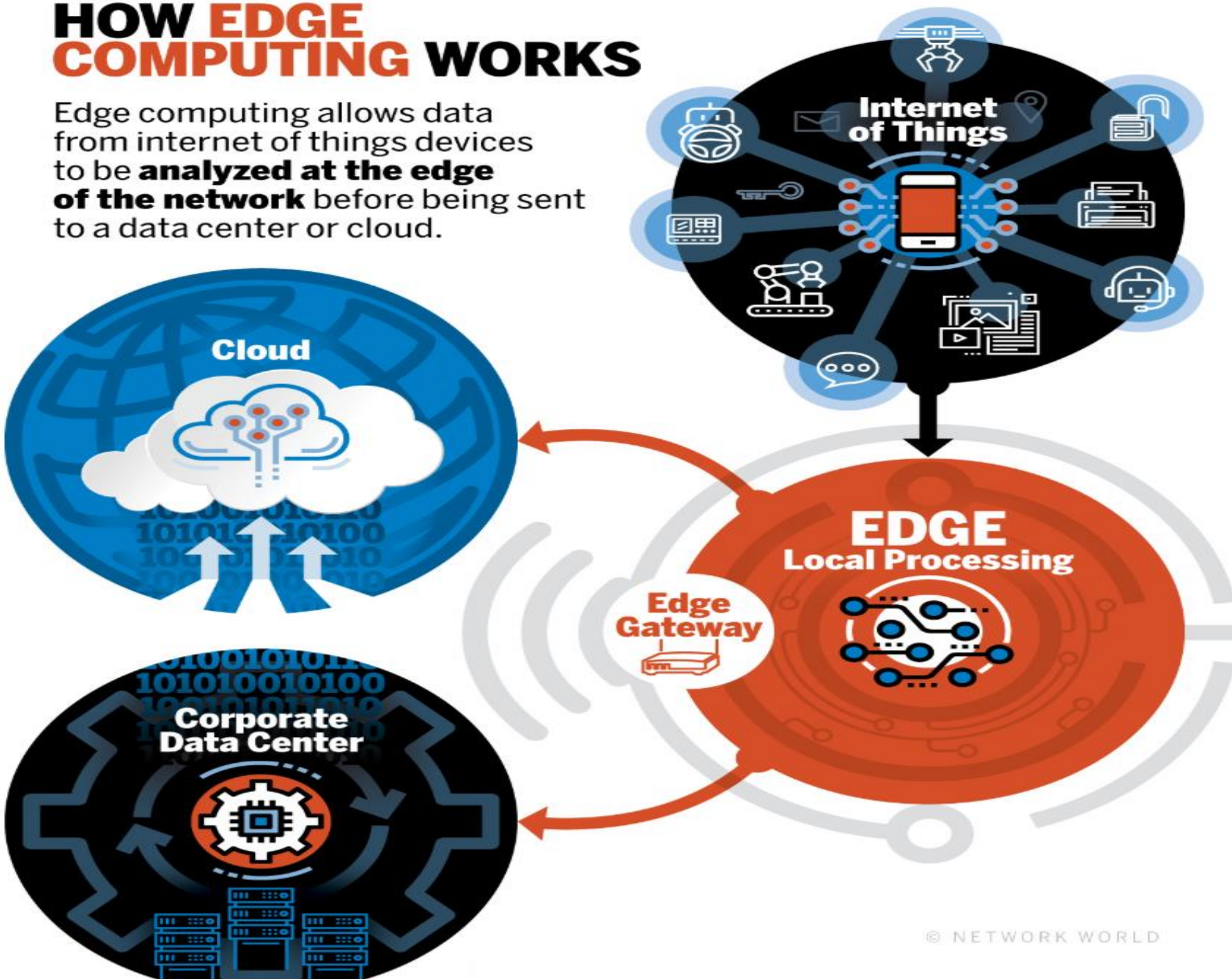
Intelligent Systems Engineering Department
Indiana University
Email: xqiu@indiana.edu

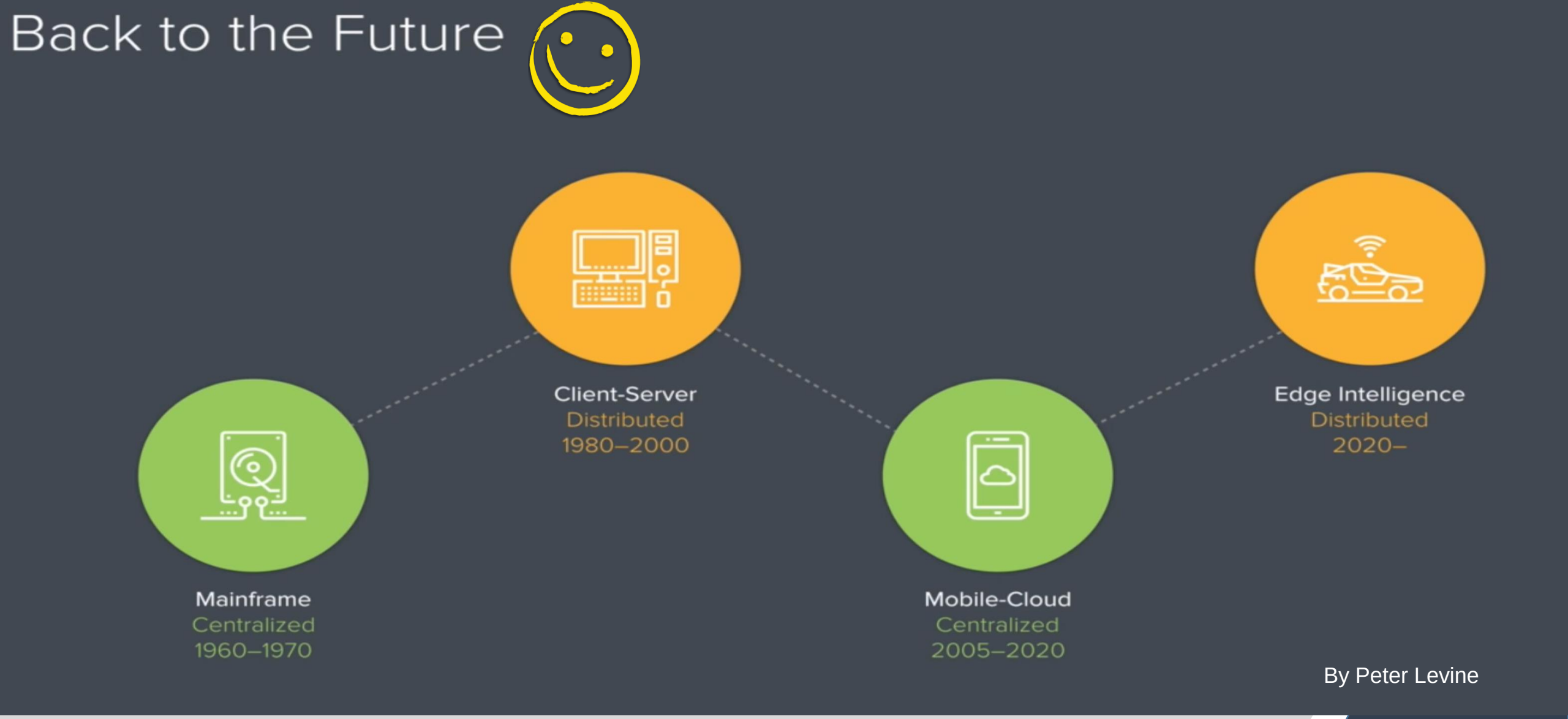


Gartner's report on Strategic Technology Trend for 2017-2018

HOW EDGE COMPUTING WORKS

Edge computing allows data from internet of things devices to be **analyzed at the edge of the network** before being sent to a data center or cloud.



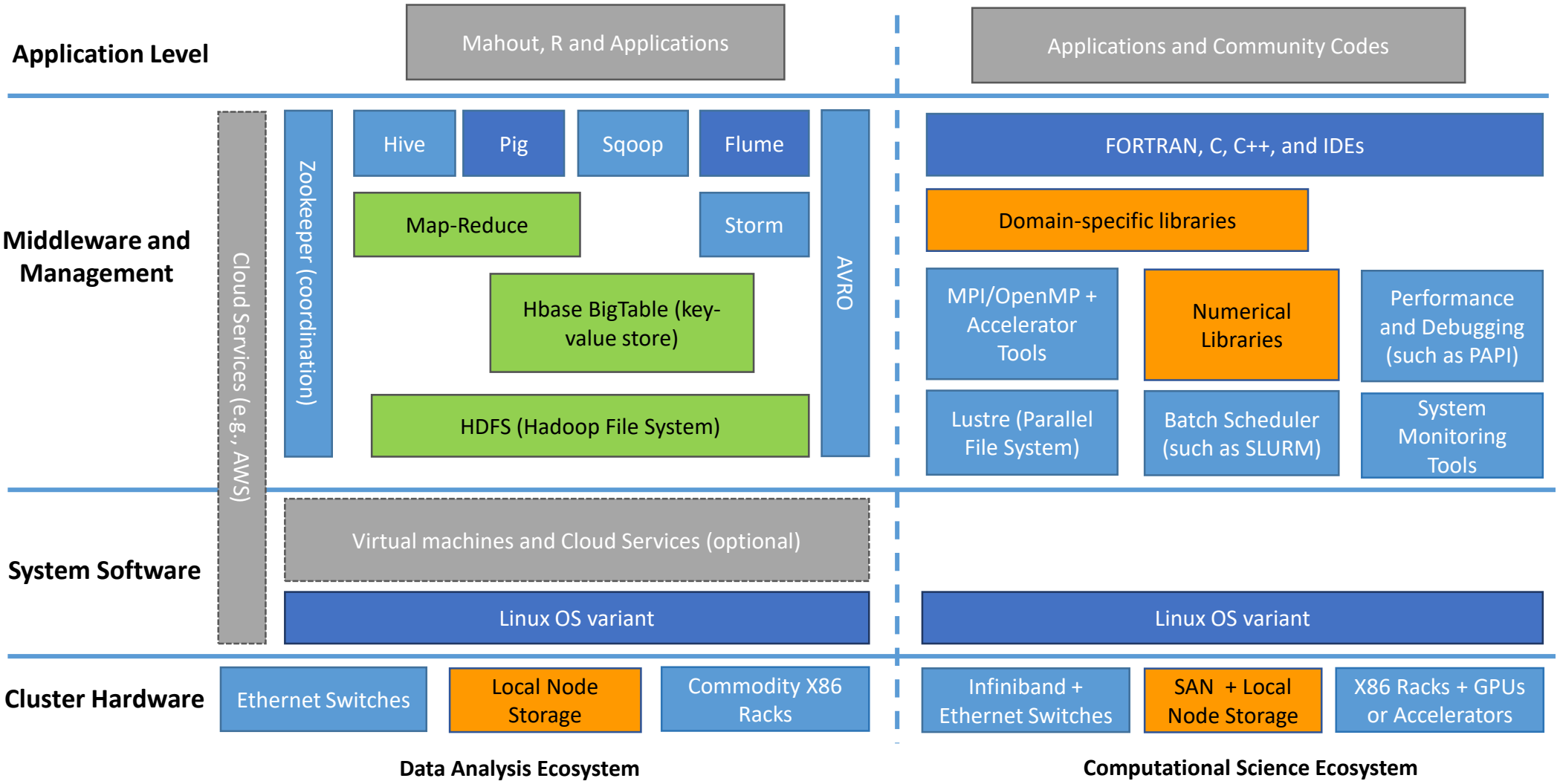


Future of Cloud Computing

HPC-Cloud Software Tools: Harp-DAAL

For High Performance Data Analytics and Machine Learning

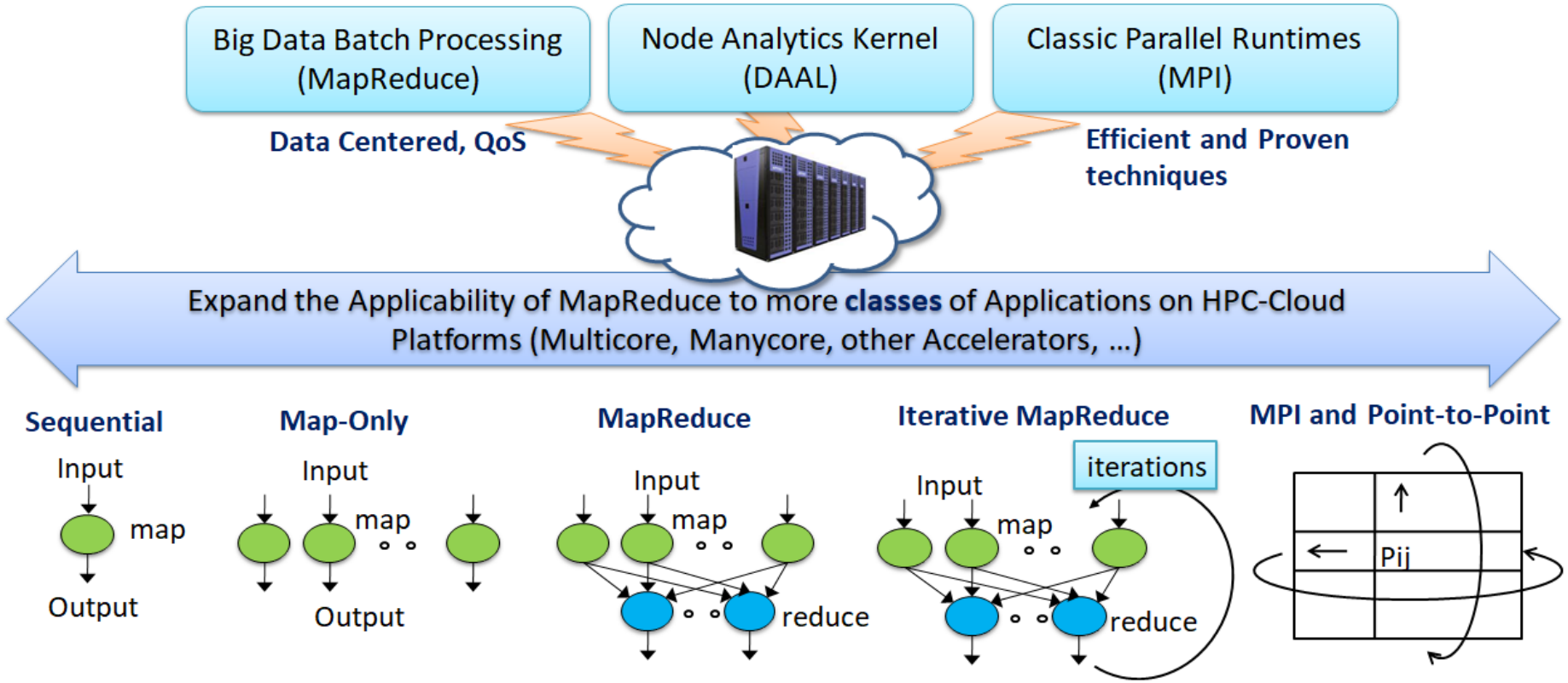
ENGINEERED nanoBIO
AN INDIANA UNIVERSITY RESEARCH NODE



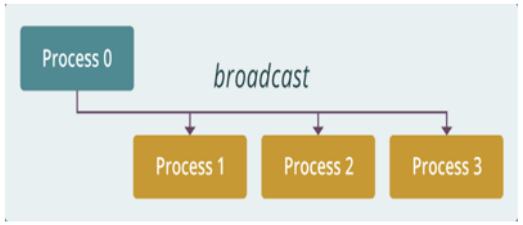
Daniel Reed and Jack Dongarra, Communications of the ACM, vol. 58, no. 7, p.58, July 2015

HPC-Cloud

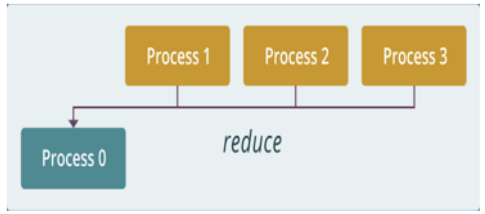
Data Analytics and Computing Ecosystem Compared



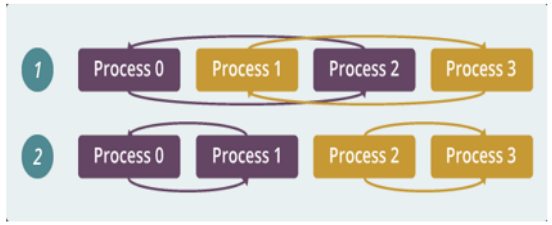
Harp and Harp-DAAL allow our data analytics to be scalable and interoperable across a range of computing systems, including clouds (Azure, Amazon), clusters (Haswell, Knights Landing) and supercomputers (IU Big Red II).



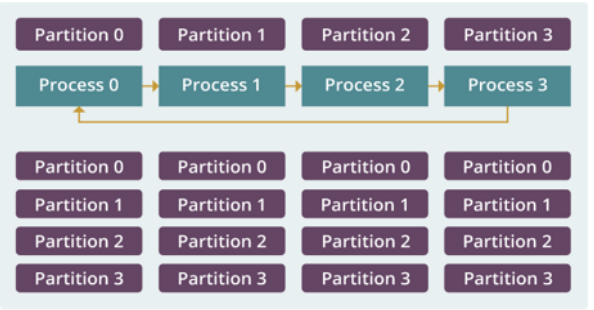
broadcast



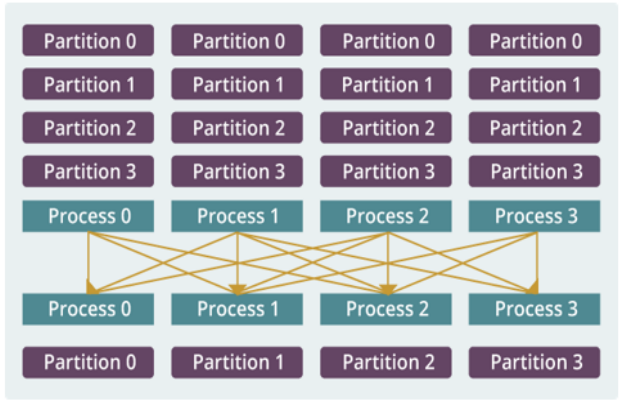
reduce



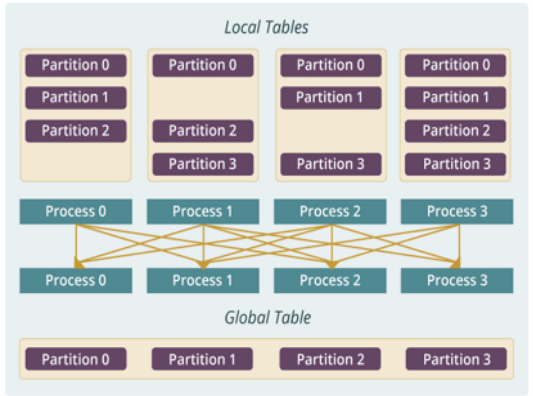
allreduce

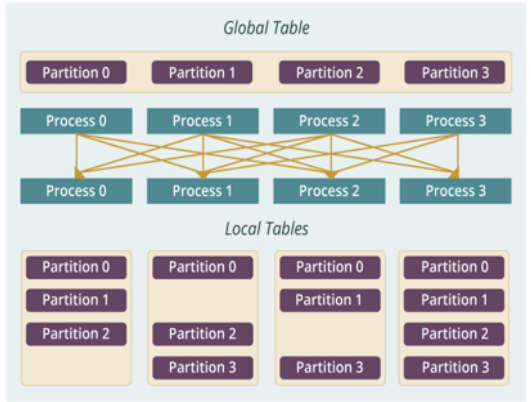


allgather

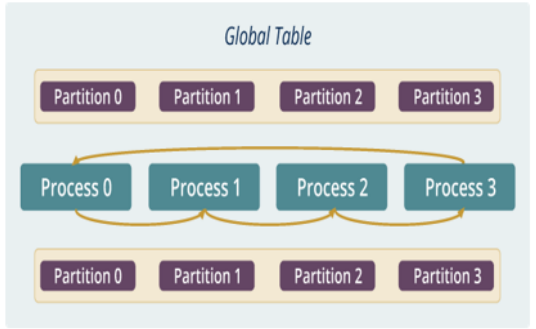


regroup





push & pull

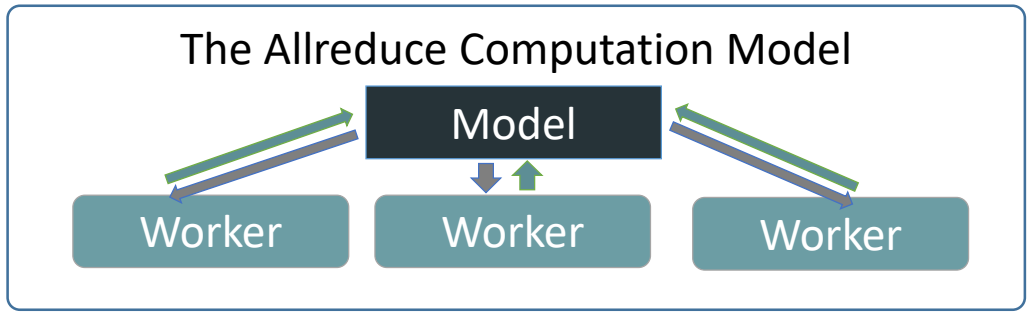


rotate

HPC-Cloud:

Tailored Communication Operations for High Performance

Example: K-means Clustering



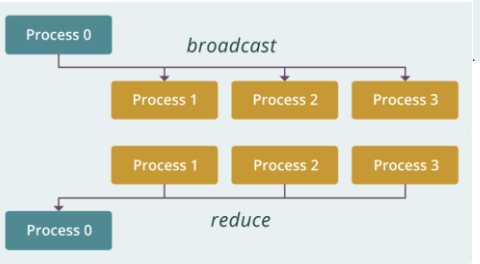
When the model size is small

When the model size is large but can still be held in each machine's memory

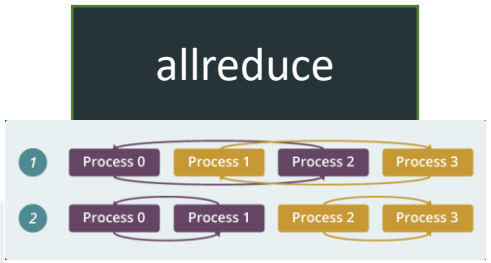
When the model size cannot be held in each machine's memory

broadcast

reduce

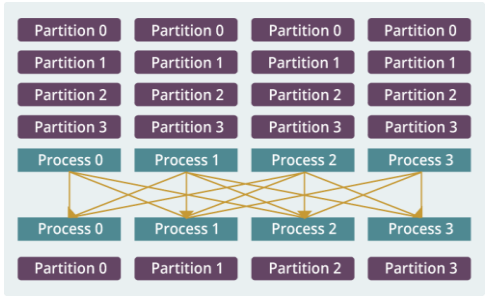


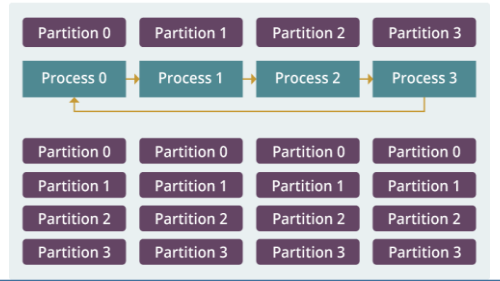
allreduce



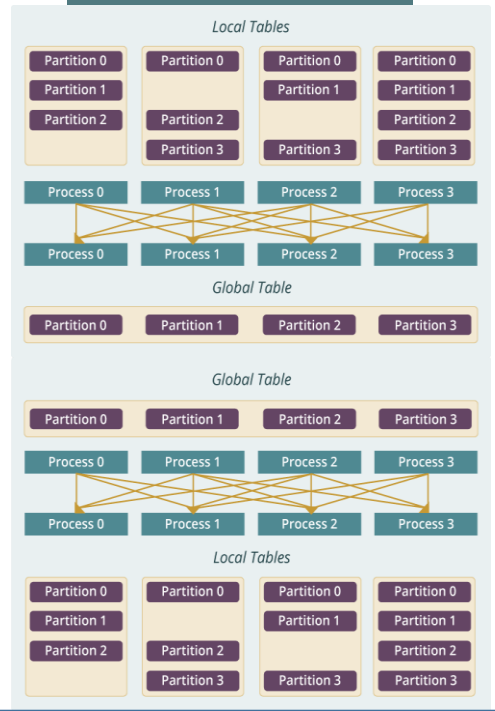
regroup

allgather

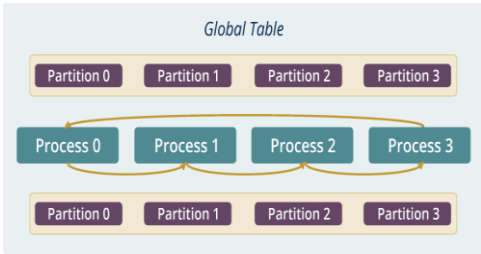


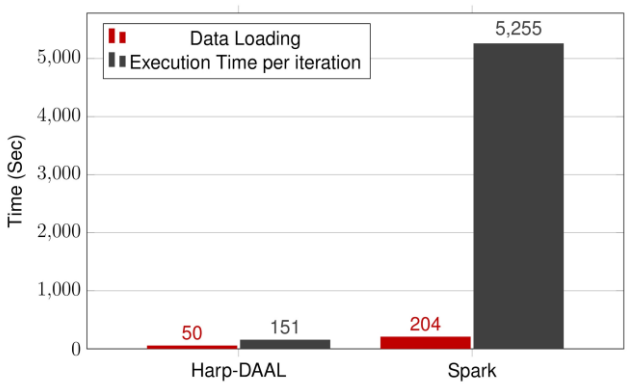


push & pull



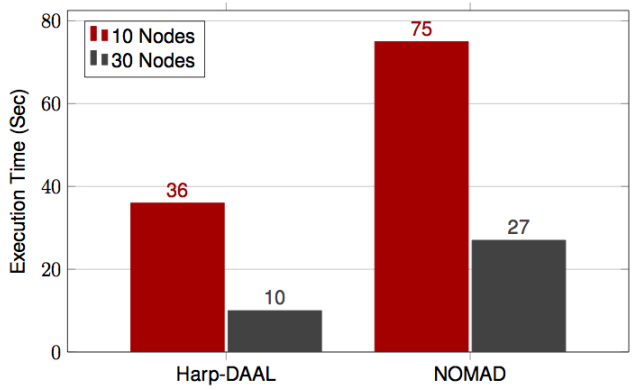
rotate





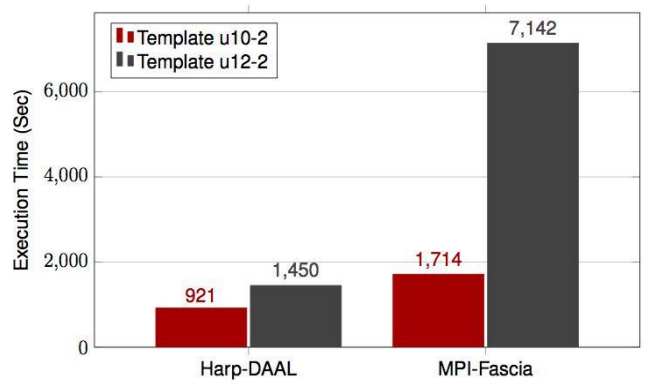
K-means

Yahoo! Flickr, including 100 million images, each with 4096 dimensional deep features (30x speedup)



MF-SGD

Twitter with 44 million vertices, 2 billion edges, subgraph template size of 10 to 12 vertices (3x speedup)

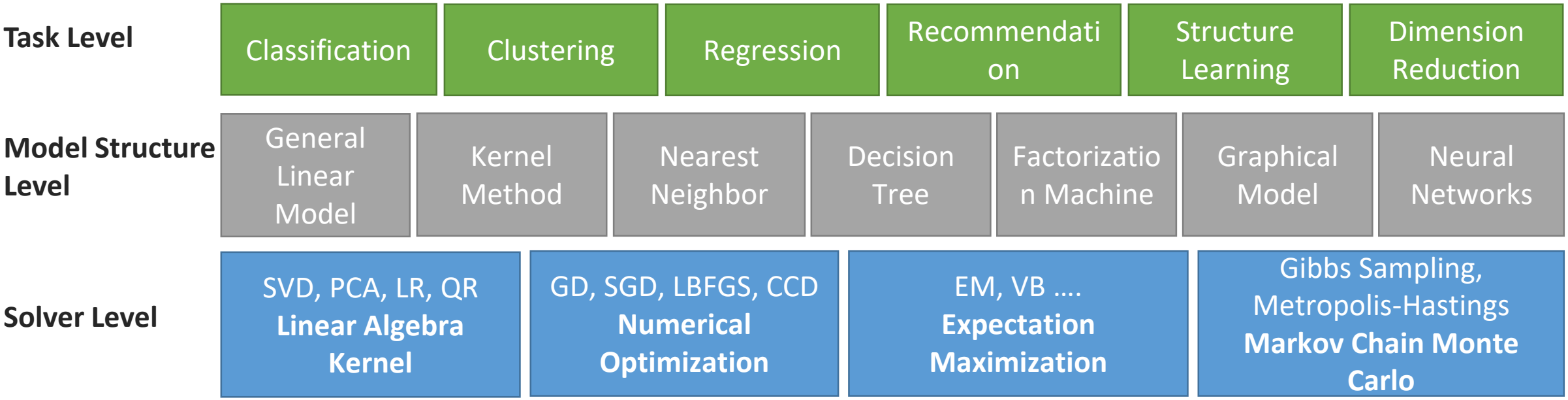


Subgraph Counting

Twitter with 44 million vertices, 2 billion edges, subgraph templates of 10 to 12 vertices (1.5x to 4x speedup)

Harp-DAAL provides fast machine learning solutions for Big Data applications. By leveraging Harp's inter-node communication innovation and Intel's highly optimized computation kernel, it delivers 2x to 15x speedups over other frameworks on Intel high-end servers.

High Performance Data Analytics

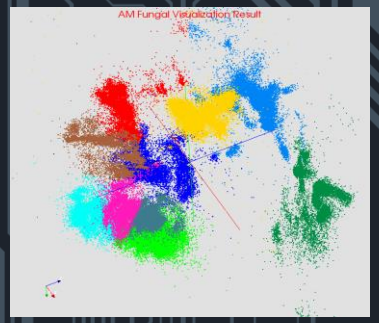


Optimization and related issues

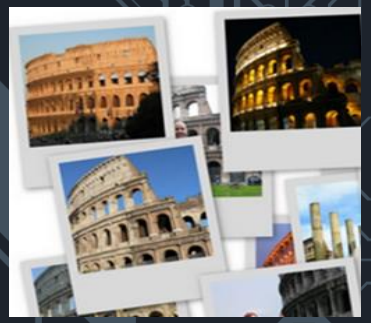
- Task level only can't capture the traits of computation
- Model is the key for iterative algorithms. The structure (e.g. vectors, matrix, tree, matrices) and size are critical for performance
- Solver has specific computation and communication pattern

Taxonomy for Machine Learning Algorithms

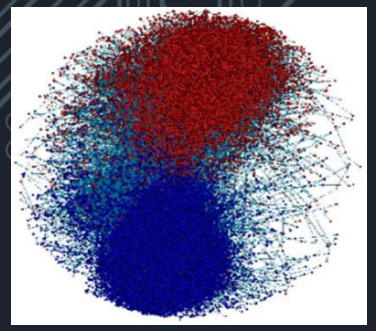
Explore Algorithms



Bioinformatics



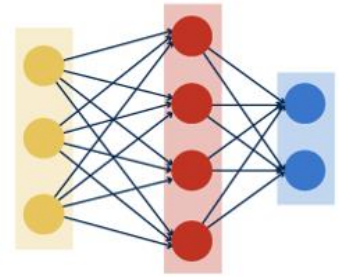
Computer Vision



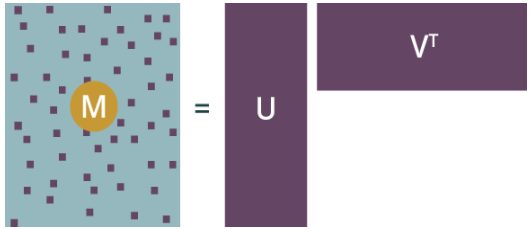
Complex Networks

Text Mining

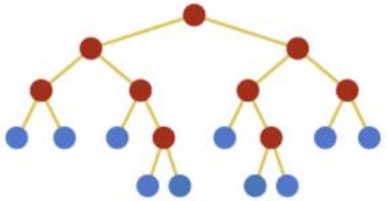
K-Means



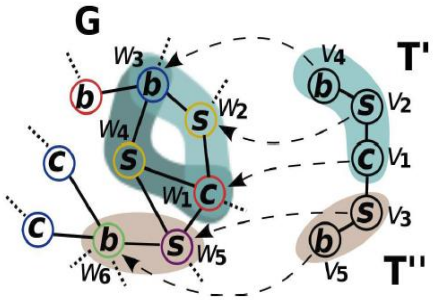
Neural Networks



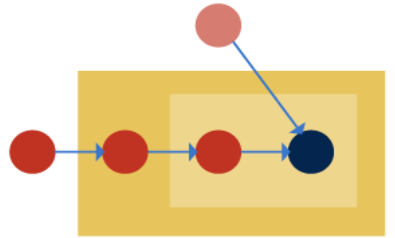
MatrixFactorization-SGD



RandomForest



SubGraph Counting



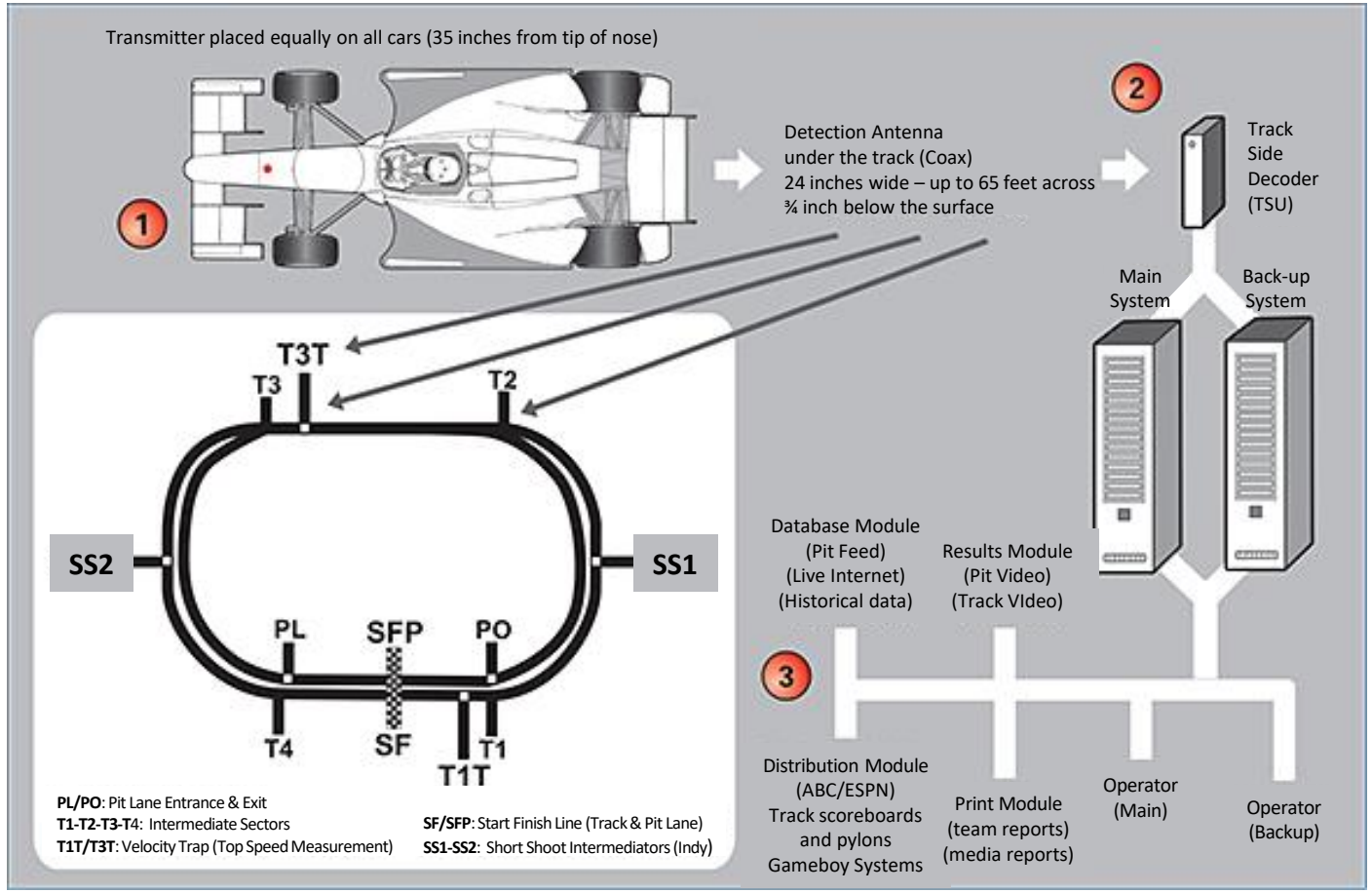
Latent Dirichlet Allocation

Real-Time Analytics Anomaly Detection

IndyCar Project & Preliminary Results

- The IndyCar Series is the premier level of open-wheel racing in North America.The series' premier event is the Indianapolis 500.
- Computing System and Data analytics is critical to the game, both in improving the performance of the team to make it faster and in helping the race control to make it safer.

IndyCar



- Sensors in the cars and under the track.
- Antenna and communication system.
- Telemetry data(including the many performance information of the cars like speed, gear, brake, throttle,etc) stream into the on-site computer system in a real-time fasion.

Timing and Score Data

Command	Count	Protocol	Description	Frequency
A	2052	MLP	Announcement	Every 60 seconds
C	19432	MLP	Completed Lap Results	Upon Event (new and repeated)
D	2652	RP	Invalidated Lap Information	Every 30 seconds
E	7737	MLP	Entry Information	Every 60 seconds
F	725	MLP	Flag Information	Upon Event (new and repeated)
G	7892	RP	Car Display Pit Stop Timer Information	Every 120 seconds
H	17260	MLP	Heart beat	Every Second
I	53	MLP	Invalidated Lap Information	Upon Event (new and repeated)
L	79884	MLP	Line Crossing Information	Upon Event
M	1738	eRP	Messages	Upon Event
N	3861	MLP	New Leader Information	Upon Event
O	33263	RP	Overall Results	Upon Event
P	3693653	eRP	Telemetry Data	
R	701	MLP	Run Information	Every 20 seconds
S	102272	MLP	Completed Section Results	Upon Event (new and repeated)
T	233	MLP	Track Information	Every 60 seconds
U	235	RP	Track Information	Every 30 seconds
V	117	MLP	Version Stream Information	Every 120 seconds
W	287	RP	Weather Data	Every 60 seconds
X	12124	RP	Heart beat	Every Second

- The INDYCAR Timing system supports retrieving timing data from the primary timing system – serial or sequential data feed for live data and report querying for historical or archived data.
- The Results Protocol is designed to deliver more detailed results information through the use of a single record command.
- Example: one Indianapolis 500 car race on the 28th of May 2017 which contained 750 MB of data and total of 3986170 records.

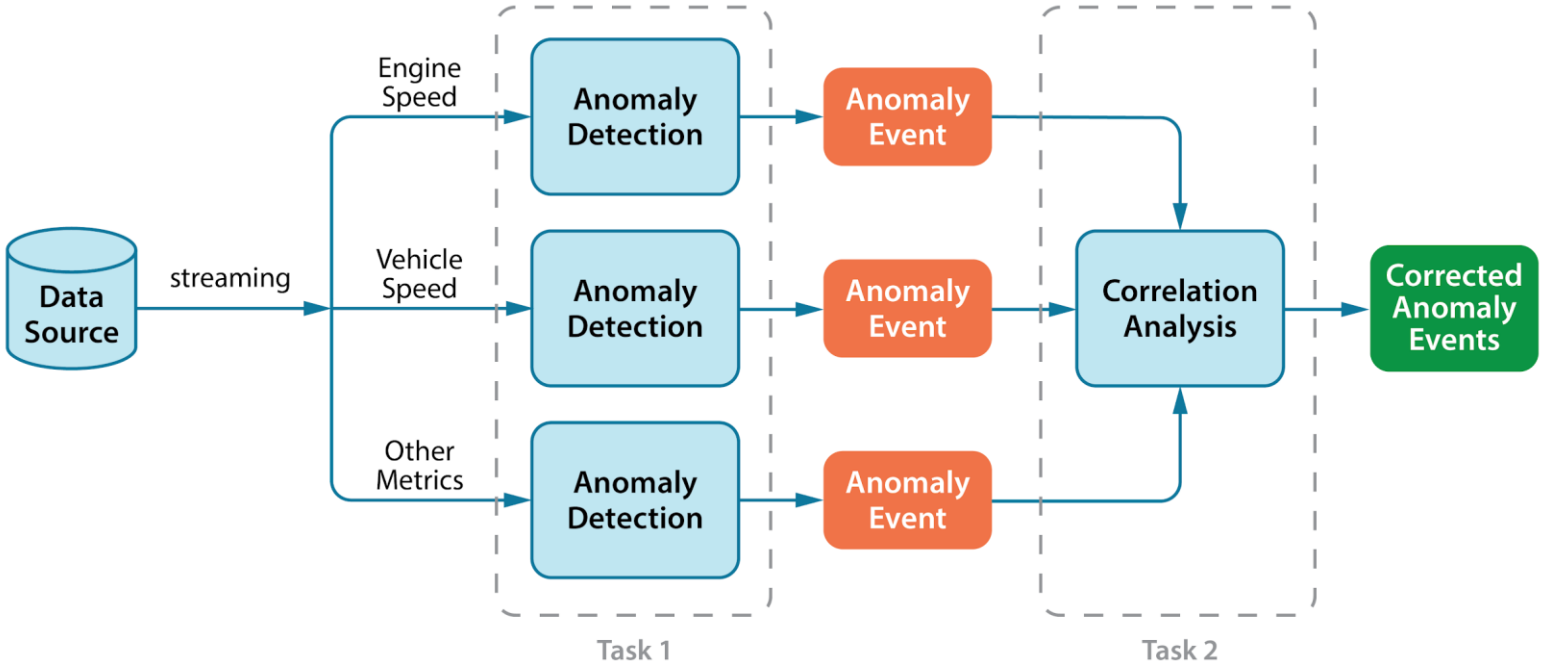
Dataset

Real-time data analysis is needed

"We want to know if it’s an expected event or a minor deviation that we need to be worried about. Helps race control people. "

" And we want to know the data corresponding to the anomaly; when car got into problems what kind of event it is and what is causing it."

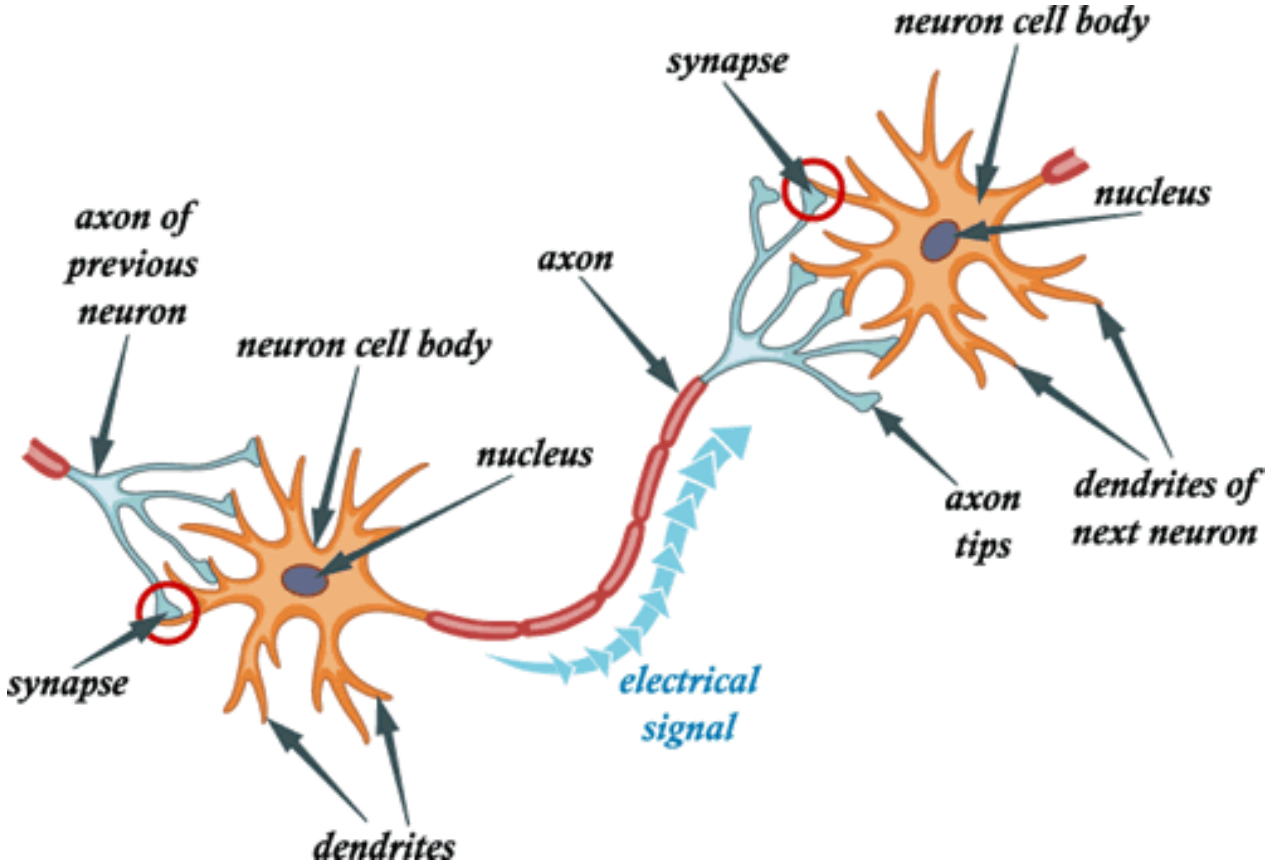
Problem



- Anomaly Detection: Learning algorithms themselves can only find the abnormal pattern in the data with best efforts under predefined assumption of what is “normal”.
- Correlation Analysis: Learning algorithms can find out the “events” from data and mine correlation relationship among the events.

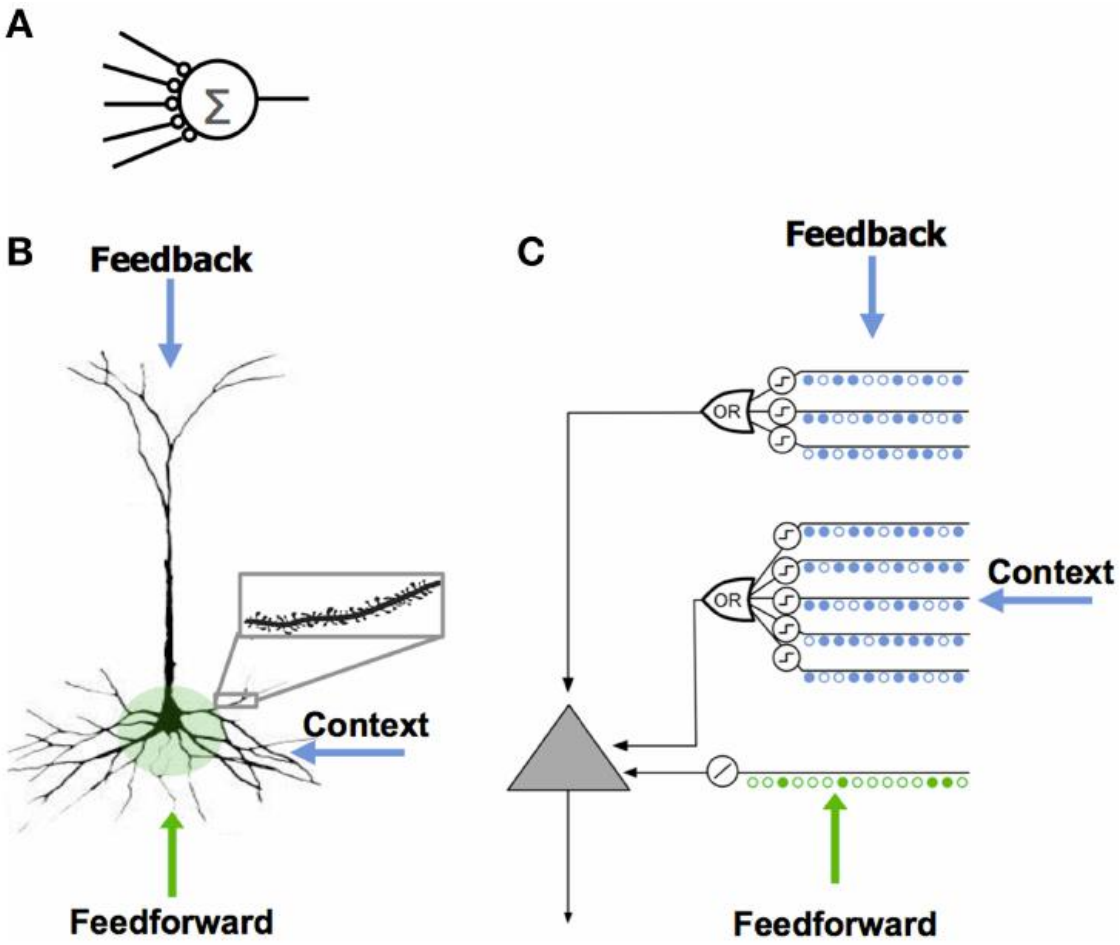
Tasks

HTM for Anomaly Detection



- Neurons have specialized projections called dendrites and axons.
- Dendrites bring information to the cell body and axons take information away from the cell body.
- Information from one neuron flows to another neuron across a synapse. The synapse contains a small gap separating neurons.

Axon, Dendrite and Synapse

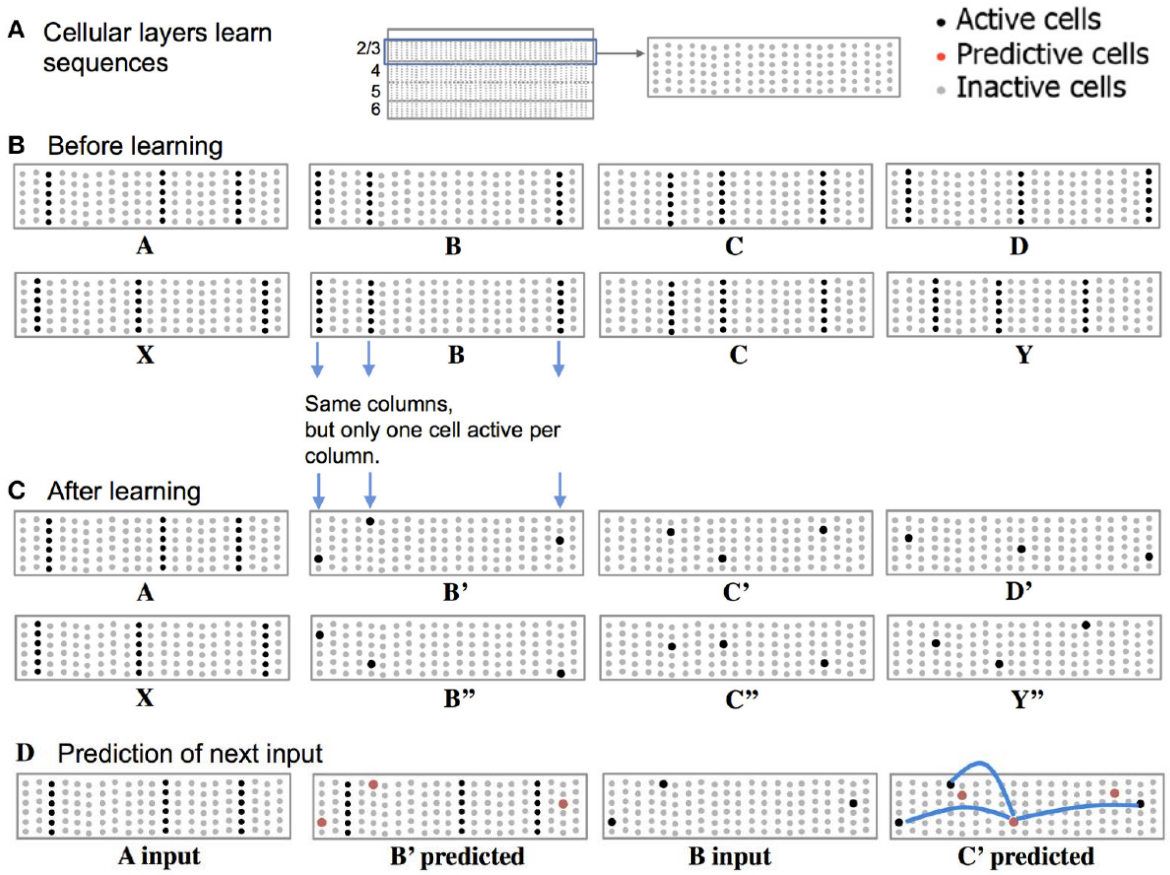


A. The neuron model used in most artificial neural networks has few synapses and no dendrites

B. Different source of input: feedforward, feedback and context

C. HTM sequence memory models dendrites with an array of coincident detectors each with a set of synapses

Hierarchical Temporal Memory(HTM)



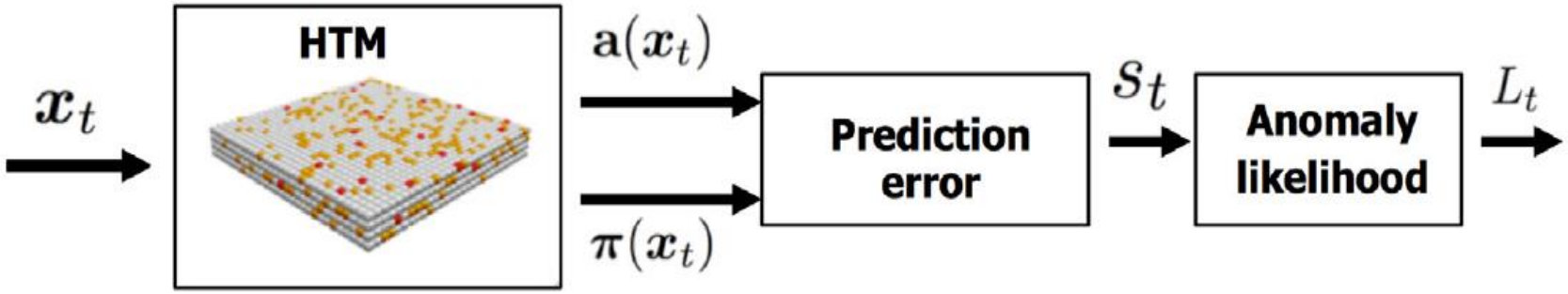
A. The layer consists of a set of mini-columns, with each mini-column containing multiple neurons.

B. Each sequence element invokes a sparse set of mini-columns, only three in this illustration.

C. Learning with context input, the inputs invoke the same mini-columns but only one cell is active in each column. Because C'and C''are unique, they can invoke the correct high-order prediction of either Y or D depending on the input from two time steps ago.

D. Learn connections to nearby neurons to predict the next input, which works as the context input.

Representing High-order Context

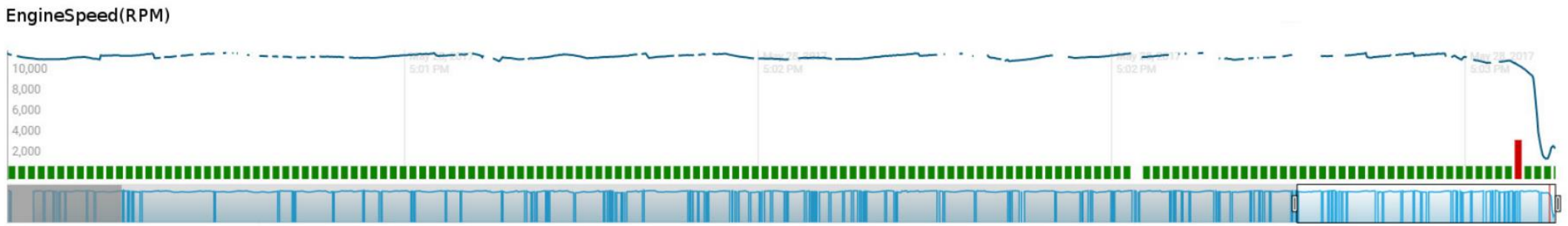


- The input time series x_t are fed to the HTM component. It models temporal patterns in $a(x_t)$ and output a prediction in $\pi(x_t)$.

- Then by building a statistical model on the prediction error, $\pi(x_t) - a(x_{t-1})$, anomaly likelihood score can be calculated on x_t.

Anomaly Detection Based on HTM

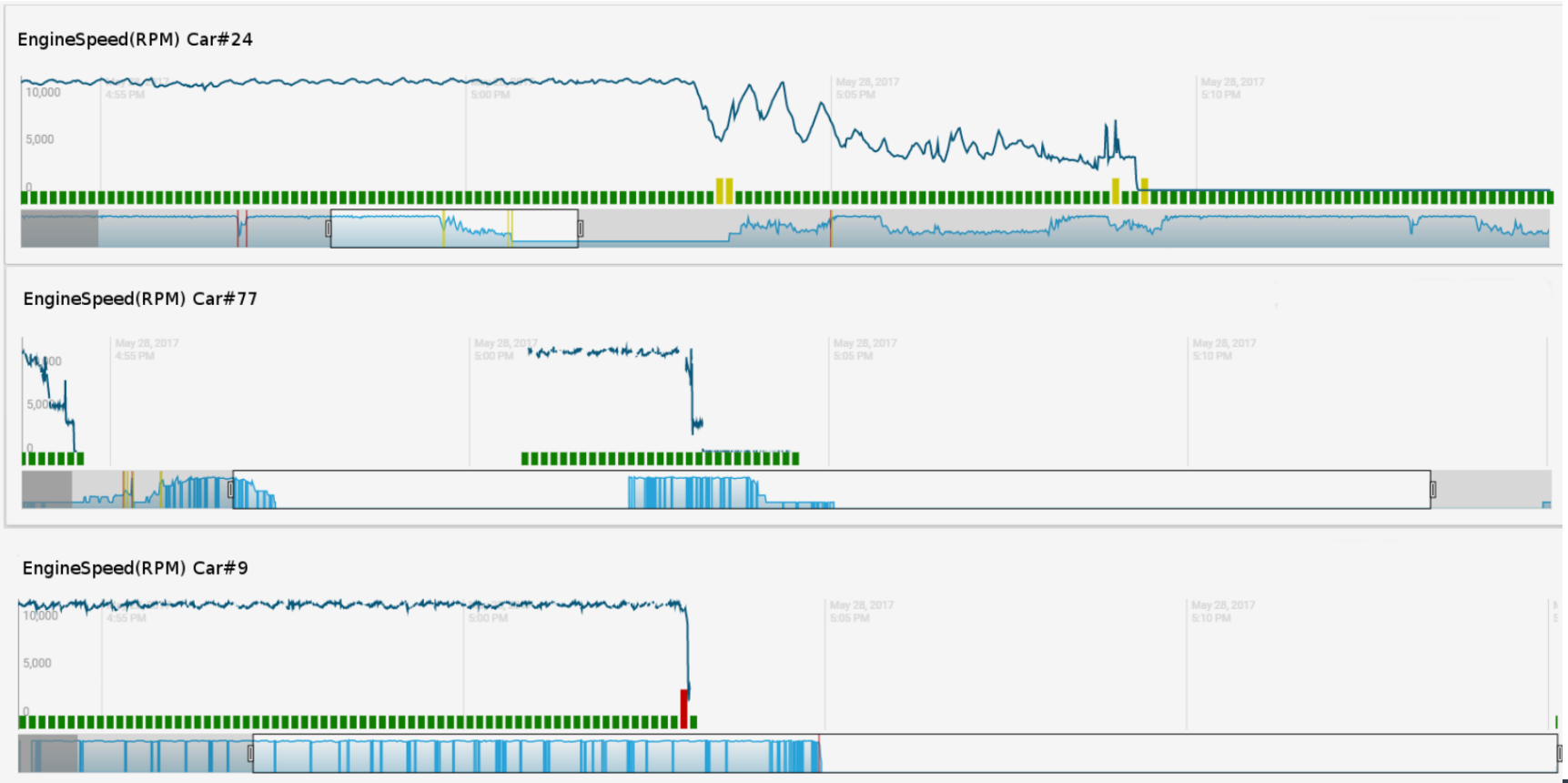
(a). Car#9 in middle of the race



(b). Car#9 at end of the race

- Detection algorithm used has the capability to detect certain type of anomaly in few seconds ahead of the time.

Anomaly Detection



Correlation of “events” among different cars: #9 #77 and #24

- The anomaly occurs around 15:03 pm, where the RPM of car #9 totally disappeared. In fact, car #9 got totaled due to a collision with car #77. The others cars, including car #24, all slowed down after the crash.

Correlation Analysis

INDIANAPOLIS 500 RACE

2017-05-28 08:20:00

Ambient Temp 82	Relative Humidity 54%
Barometer 2892	Time of the Day 16:45:38.3370

8 Scott Dixon *from Auckland New Zealand*

Chip Ganassi Racing Teams (#25)

Engine: Honda
License: Veteran
Competitor:37

Last Lap Section Speeds

1	2	3	4	5	6	7	8	9
221.045	221.045	221.045	221.045	221.045	221.045	221.045	221.045	221.045

12 Adam Miller *from Auckland New Zealand*

Chip Ganassi Racing Teams (#25)

HONDA

License	Competitor
Veteran	37

Last Lap Section Speeds

1	2	3	4	5	6	7	8	9
221.045	221.045	221.045	221.045	221.045	221.045	221.045	221.045	221.045

31 Seth Garcias *from Auckland New Zealand*

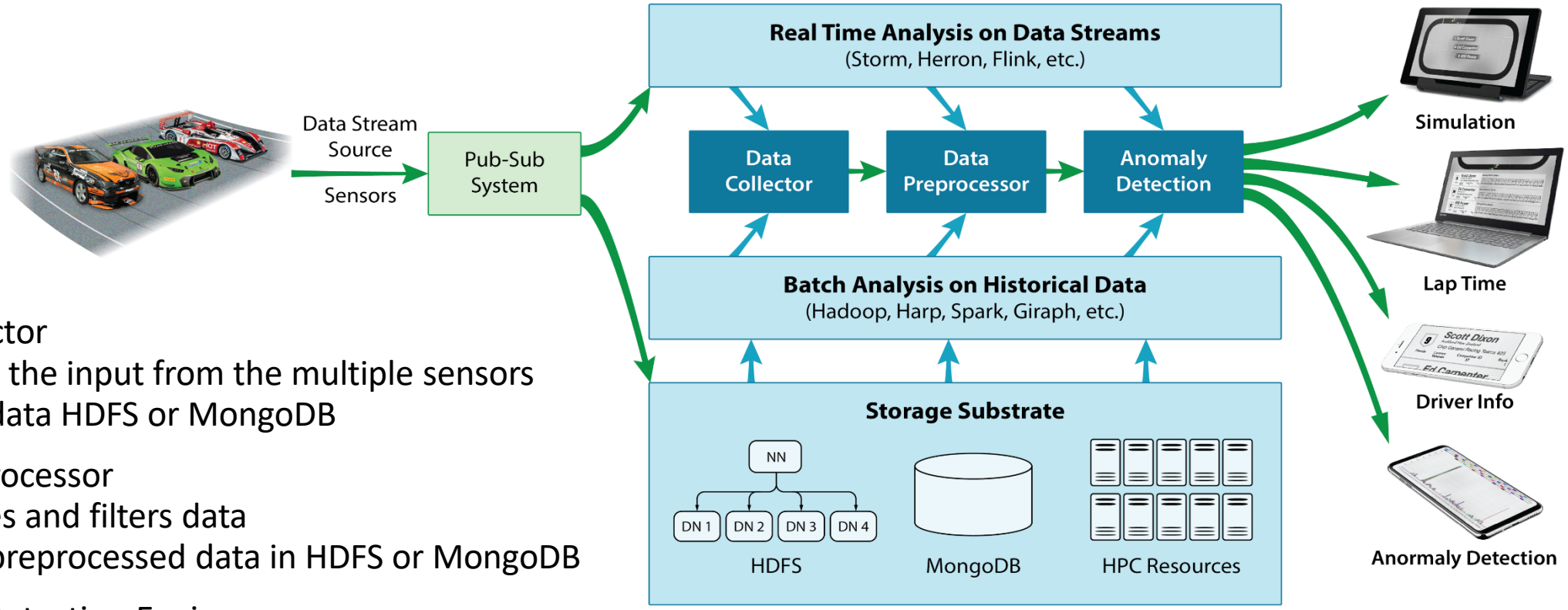
Chip Ganassi Racing Teams (#25)

3

Engine	License	Competitor
Honda	Veteran	37

Last Lap Section Speeds

1	2	3	4	5	6	7	8	9
221.045	221.045	221.045	221.045	221.045	221.045	221.045	221.045	221.045

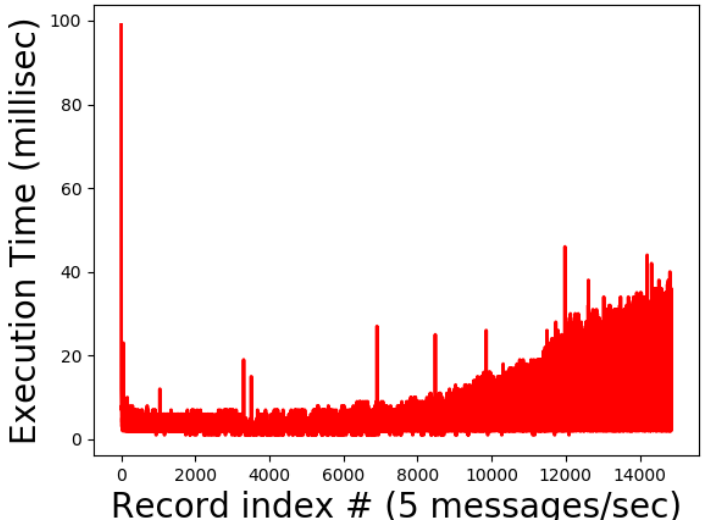


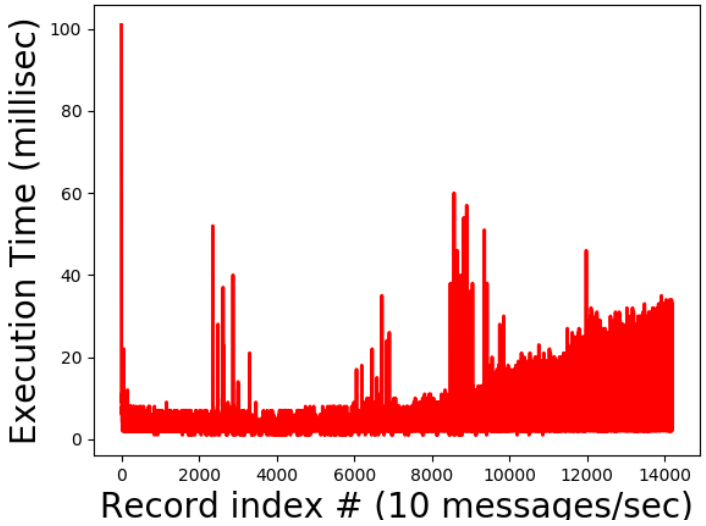
Data collector
- collects the input from the multiple sensors
- stores data HDFS or MongoDB

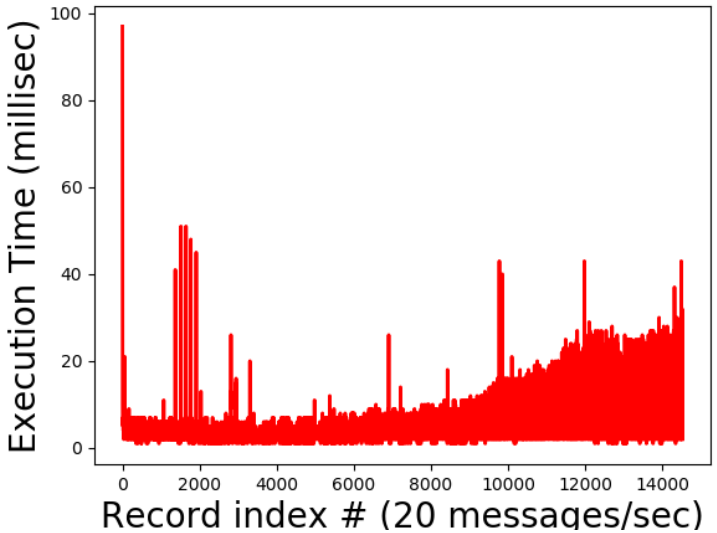
Data Preprocessor
- accesses and filters data
- stores preprocessed data in HDFS or MongoDB

Anomaly Detection Engine
- involves high performance batch and streaming data processing frameworks
- detects anomalies from preprocessed data pipeline

System Architecture

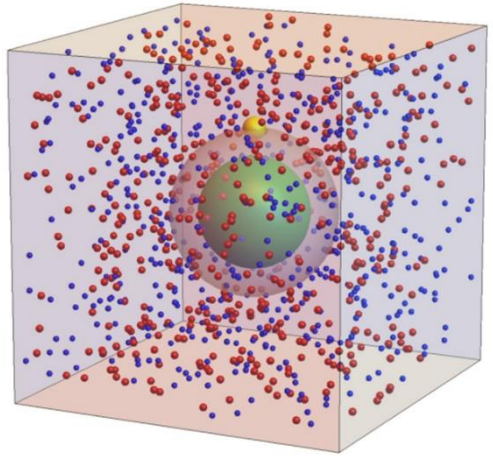


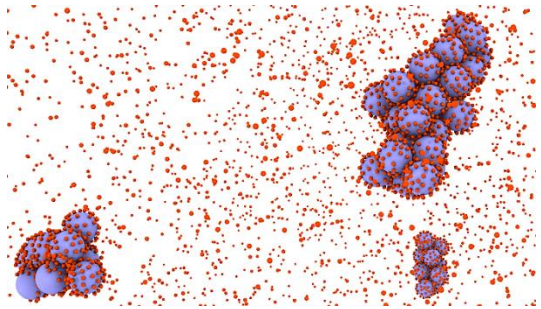


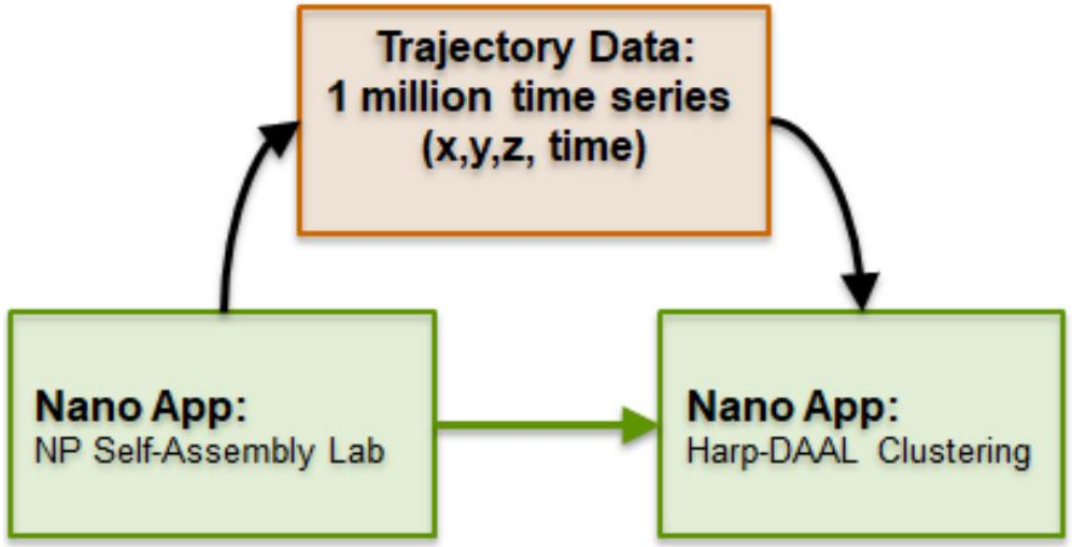


- Performance metrics for single node execution with parallelism =1 for two input rates: 10 and 20 message/sec
- Initial execution time of roughly 210 millisecond incurred in both cases to initiate the HTM network API and feed input from telemetry spout

Streaming Results using Apache Storm







- These Harp-DAAL HPC kernels could well serve in accelerating the process of clustering the nanoBIO simulation results to obtain different nanoparticle trajectories.
- NP Self-Assembly Lab simulates assembly of nanoparticles and generates 1 million high dimensional trajectory data. Harp-DAAL takes the input data and runs clustering over different nanoparticle trajectories that can be visualized for NP distribution.

Real-Time Simulation Data Analysis

Harp-DAAL: Prototype and Production Code

Available at https://dsc-spidal.github.io/harp

DSC-SPIDAL / harp

Unwatch 13 | Star 1 | Fork 6

Code | Issues 1 | Pull requests 2 | Projects 0 | Wiki | Pulse | Graphs | Settings

Branch: master ▾ harp / harp-daal-app / src / edu / iu / | Create new file | Upload files | Find file | History

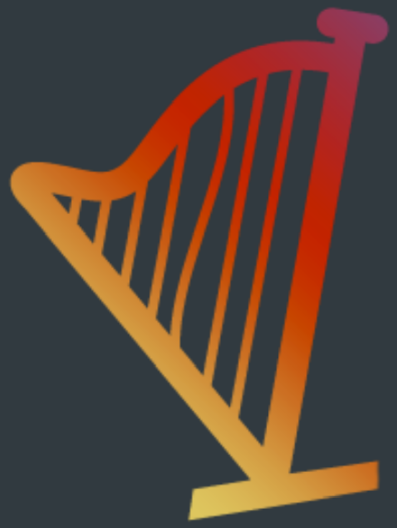
Chen add codes for harp-daal-als — Latest commit 158f8e9 5 days ago

..		
benchmark	re-structure the codes	2 months ago
daal	add daal_kmeans codes	2 months ago
daal_als	add codes for harp-daal-als	5 days ago
daal_kmeans/regroupallgather	add daal_kmeans codes	2 months ago
daal_sgd	re-structure the codes	2 months ago
dymoro	re-structure the codes	2 months ago
fileformat	re-structure the codes	2 months ago
kmeans	re-structure the codes	2 months ago
train	re-structure the codes	2 months ago
wdamds	re-structure the codes	2 months ago

Source codes became available on Github in February, 2017.

- Harp-DAAL follows the same standard of DAAL's original codes
- Twelve Applications
  - Harp-DAAL Kmeans
  - Harp-DAAL MF-SGD
  - Harp-DAAL MF-ALS
  - Harp-DAAL SVD
  - Harp-DAAL PCA
  - Harp-DAAL Neural Networks
  - Harp-DAAL Naïve Bayes
  - Harp-DAAL Linear Regression
  - Harp-DAAL Ridge Regression
  - Harp-DAAL QR Decomposition
  - Harp-DAAL Low Order Moments
  - Harp-DAAL Covariance

Open Source Github Website (https://dsc-spidal.github.io/harp)



Harp-DAAL

in collaboration with intel IU NSF

is a high performance framework with the fastest machine learning algorithms on Intel's Xeon and Xeon Phi architectures.

See how it works

Performance

Explore algorithms

Hands on

Slide deck

Acknowledgements

We gratefully acknowledge support from NSF, IU and Intel Parallel Computing Center (IPCC) Grant.

Langshi Cheng , Bo Peng , Supun Kamburugamuve, Sahil Tyagi , Sabra Ossen, Lynne Wang, Tiana Deckard,

Robert Henschel, Craig Stewart, Shaojuan Zhu, Lisa Smith

Intelligent Systems Engineering
School of Informatics and Computing
Indiana University