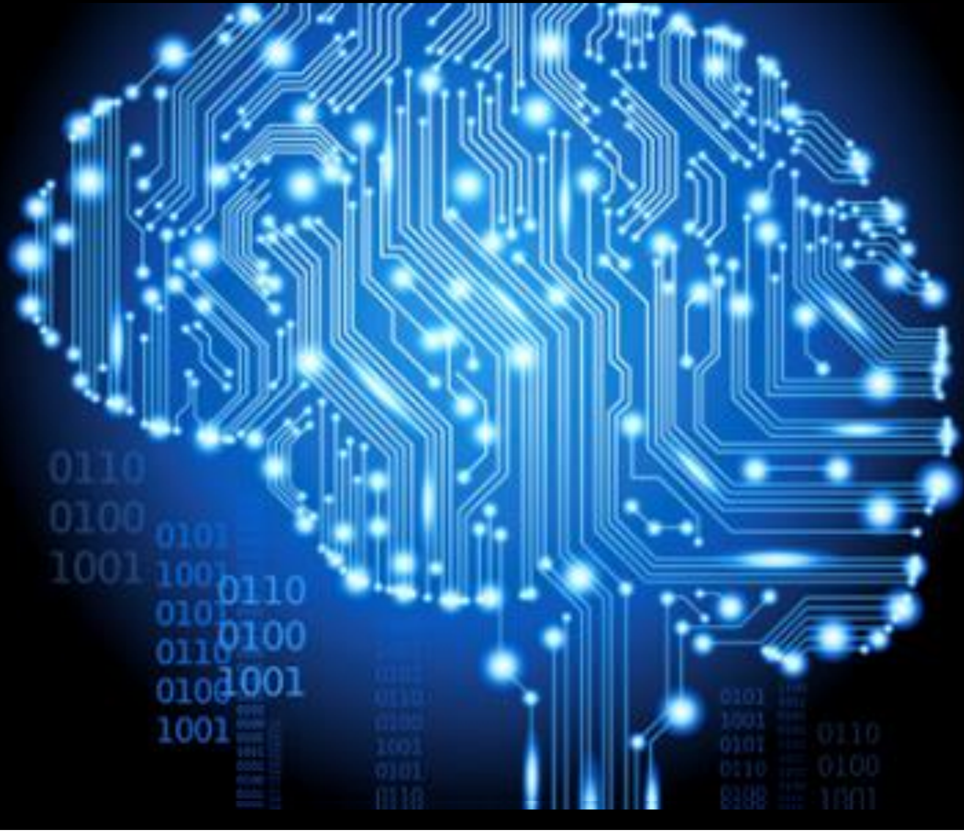
THE DOWNLOAD

TECH TALKS BY HPCC SYSTEMS



HPCC SYSTEMS®

The Download: Community Tech Talks Episode 6

August 1, 2017

LexisNexis® RISK SOLUTIONS

Welcome!

- Please share: Let others know you are here with #HPCCTechTalks
- Ask questions! We will answer as many questions as we can following each speaker.
- We welcome your feedback - please rate us before you leave today and visit our blog for information after the event.
- Want to be one of our featured speakers? Let us know! techtalks@hpccsystems.com

Community announcements

- HPCC Systems 6.4 now gold! Among the features include:
  - More performance improvements on Roxie
  - New ML Bundles for Logistic Regression & Linear Regression
  - Colorization & icon options in ECL IDE
  - Extended embedded language support for R, Python & SWS AWS plugins
  - Enhanced support for Dynamic ESDL
  - WsSQL 6.4.0 and wsclient 1.2 coming soon!
- Reminder: Call for Poster Abstracts still open for the 2017 HPCC Systems Community Day!
  - Poster Competition held on October 3
  - Submission instructions on the Wiki
  - Community Day will be held in Atlanta on October 4, 2017
  - NEW THIS YEAR!
    - Pre-Event Workshop on October 3
    - Registration is open to the public to attend
  - Details at https://hpccsystems.com/hpccsummit2017
  - Thank you to our Sponsors!

Cognizant Infosys Datum SOFTWARE

Dr. Flavio Villanustre
VP Technology
LexisNexis® Risk Solutions
Flavio.Villanustre@lexisnexisrisk.com

Community Day pre-event workshop

Mastering Your Big Data with ECL

This class is for attendees who want to understand the HPCC Systems platform and learn ECL to build powerful data queries. Anyone who needs a basic familiarity and learn best practices with ECL should attend. The one day class will take the student through the entire ETL cycle from Spray (Extract) to Transform (THOR) and finally to Load (ROXIE).

Topics include:

- **Part 1: Data Extraction and Transformation**
  - Quick overview of THOR cluster, and the parallel distributed data processing concept, setting up a cluster, ECL Watch overview, spraying data, ECL IDE, ECL language essentials, and more…
- **Part 2: Prepare the Data Search Engine**
  - Defining and building an INDEX, getting single and batch results, data indexing, filtering and normalization, searching, and more…
- **Part 3: Write and Publish ROXIE query**
  - Call Search, Implicit function, publish in ECL Watch, test in WS-ECL, and more…

What:
Mastering Your Big Data w/ ECL

When:
Tuesday October 3, 9am – 4pm

Where:
Ritz Carlton Buckhead, Atlanta, Ga

Register:
hpccsummit2017.eventbrite.com

Community Day agenda

Wednesday, October 4, 2017

The agenda will tentatively run from 8:30am – 5:00pm ET. We will have a fantastic line-up of speakers featuring industry experts, academia and thought leaders. We are currently finalizing the agenda but here is a sneak peek!

Time	Topic
7:00am – 8:30am	Registration and Breakfast
8:30am – 9:15am	Welcome and Sponsor Keynotes
9:15am – 10:30am	Track 1: HPCC Systems in Industry: Real World Use Cases Featuring DataSeers, Couchbase, CPL Online
10:30am – 10:45am	Break - Poster Presentations, Robotics Display & Exhibits
10:45am - 12:00pm	Track 2: HPCC Systems in Academia: Beyond the Classroom Featuring Humboldt University Berlin and North Carolina State University
12:00pm - 12:45pm	Lunch - Poster Presentations and Robotics Display
12:45 – 1:00pm	Community Awards Ceremony
1:00pm – 2:00pm	Panel Discussion: Integrated Scientific Discovery
2:00pm - 3:15pm	Track 3: HPCC Systems in the Limelight: Success Across RELX Group Featuring LexisNexis Risk Solutions, Reed Business Information and Reed Exhibitions
3:15pm - 3:30pm	Break - Poster Presentations, Robotics Display & Exhibits
3:30pm - 4:50pm	Track 4: HPCC Systems Roadmap Tech Talks Featuring topics on the Platform Roadmap, Visualization, Machine Learning and Architecture Improvements
4:50pm - 5:00pm	Closing Words & Adjourn

Register today at hpccsummit2017.eventbrite.com

Today's speakers

Lorraine Chapman

Consulting Business Analyst, LexisNexis® Risk Solutions

Lorraine.chapman@lexisnexisrisk.com

Lorraine has worked alongside software developers for over 20 years in a supportive role which has ranged from producing documentation including developing on-line help systems to software testing and release management.

Lorraine joined LexisNexis in 2004 and as well as continuing to work alongside the HPCC Systems platform development team, also administers the HPCC Systems Intern Program and manages our application to be an accepted organization for Google Summer of Code.

Lorraine is an active blogger on our website covering a wide range of subjects from new release information, features and improvements and the work students have completed during their internships.

Lily Xu

PhD Student, Computer Science, Clemson University

lilix@g.clemson.edu

Lily is a third year Ph.D. student studying in Computer Science at Clemson University in the USA. She is currently doing research in the DICE (Data Intensive Computing Eco-Systems) lab in the School of Computing. Her research mainly focusses on Machine Learning, Parallel and Distributed Computing, High Performance Computing.

Last year, she joined the team to implement the YinYang K-Mean machine learning algorithm in ECL. This year, she has returned to build on this work by optimizing this algorithm for large clusters.

Today's speakers

George Mathew

PhD Student, Computer Science
North Carolina State University
george2@ncsu.edu

George Mathew is a first year PhD student in CS at North Carolina State University working at RAISE lab(ai4se.net). He is a full stack software engineer. His prime areas of interests are machine learning and software development. In his free time he works on his maintains a repository of optimization algorithms, collects vintage vinyl records and goes biking. To know more about George, visit his website (bigfatnoob.us).

Vivek Nair

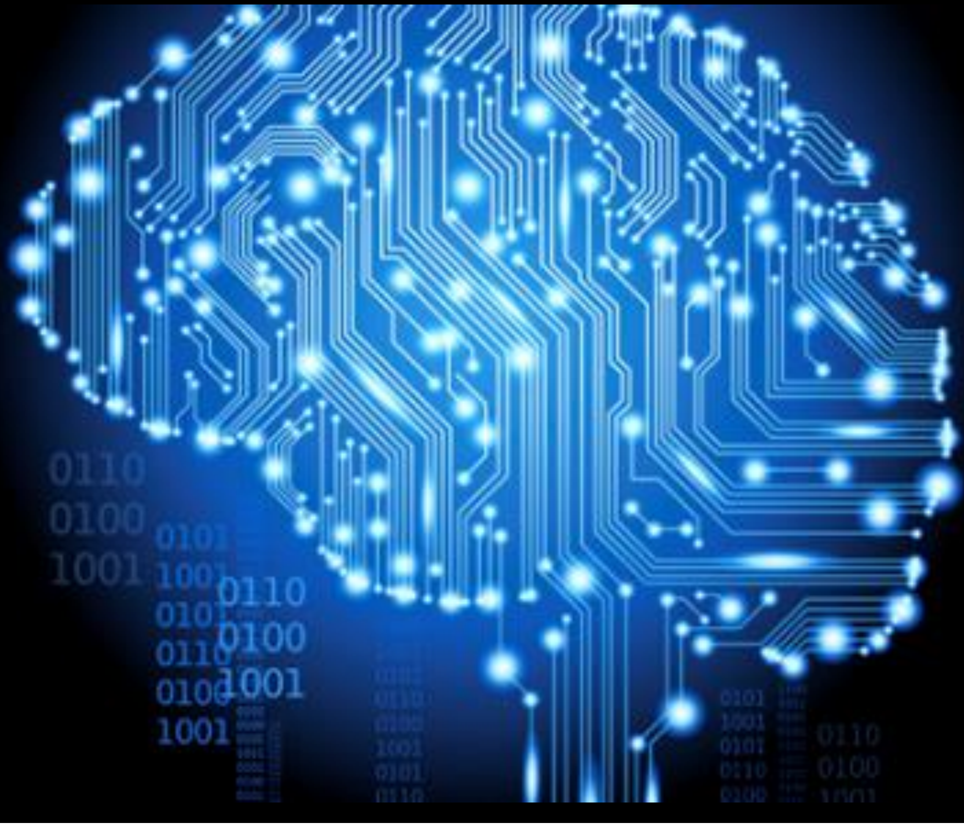
PhD Student, Computer Science,
North Carolina State University
vnair2@ncsu.edu

Vivek Nair is a fifth year Ph.D. student in the Department of Computer Science at North Carolina State University. His primary interest lie in using search-based techniques to solve software engineering problems. He is currently working on optimizing the performance of highly configurable systems. He received his master degree and worked in the mobile industry for a period of 2 years before returning to graduate school.

Vivek is currently (summer 2017) completing an HPCC Systems intern project which involves trying to connect HPCC Systems with Spark. For more information, visit his website and read his blog tracking his progress on his intern project.

THE DOWNLOAD

TECH TALKS BY HPCC SYSTEMS



HPCC SYSTEMS®

HPCC Systems Summer Internship Program

Lorraine Chapman
Consulting Business Analyst,
LexisNexis® Risk Solutions

LexisNexis®
RISK SOLUTIONS

More about the HPCC Systems Intern Program…

- Blogs about the program: https://hpccsystems.com/blog
- Available projects: https://wiki.hpccsystems.com/x/yIBc
- Previously complete projects: https://wiki.hpccsystems.com/x/g4BR
- Student wiki: https://wiki.hpccsystems.com/x/HwBm
- HPCC Systems Technical Presentation Competition 2016: https://wiki.hpccsystems.com/x/FQCv

Questions?

Lorraine Chapman

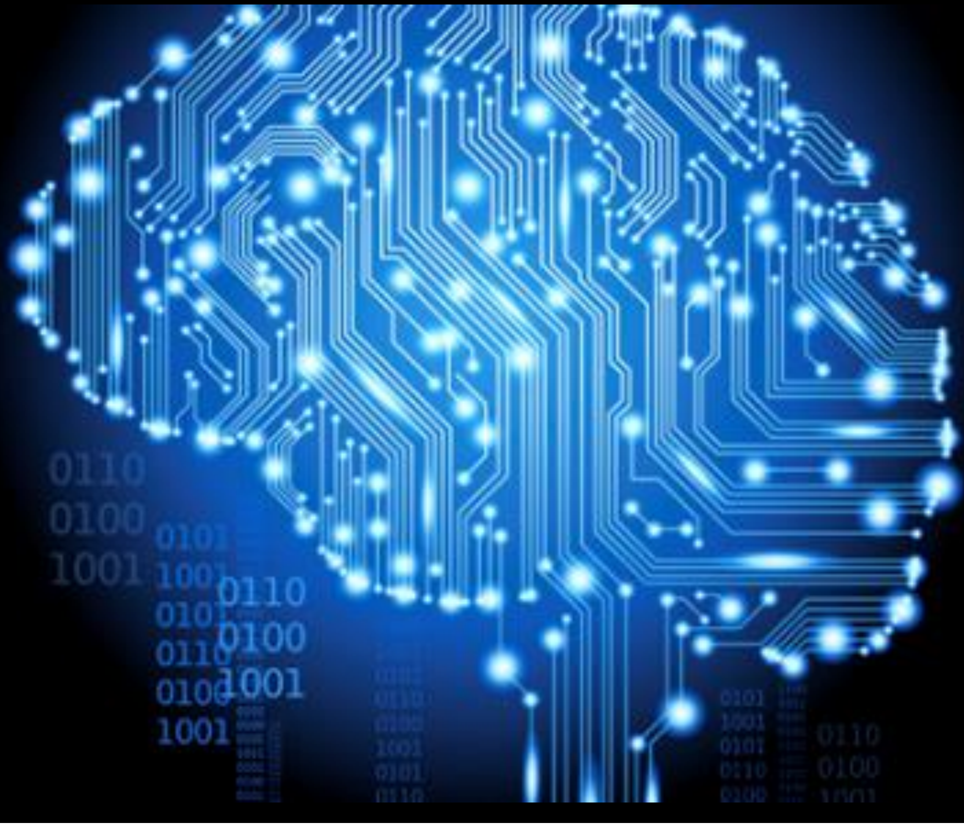
Consulting Business Analyst,

LexisNexis® Risk Solutions

Lorraine.chapman@lexisnexis.com

THE DOWNLOAD

TECH TALKS BY HPCC SYSTEMS



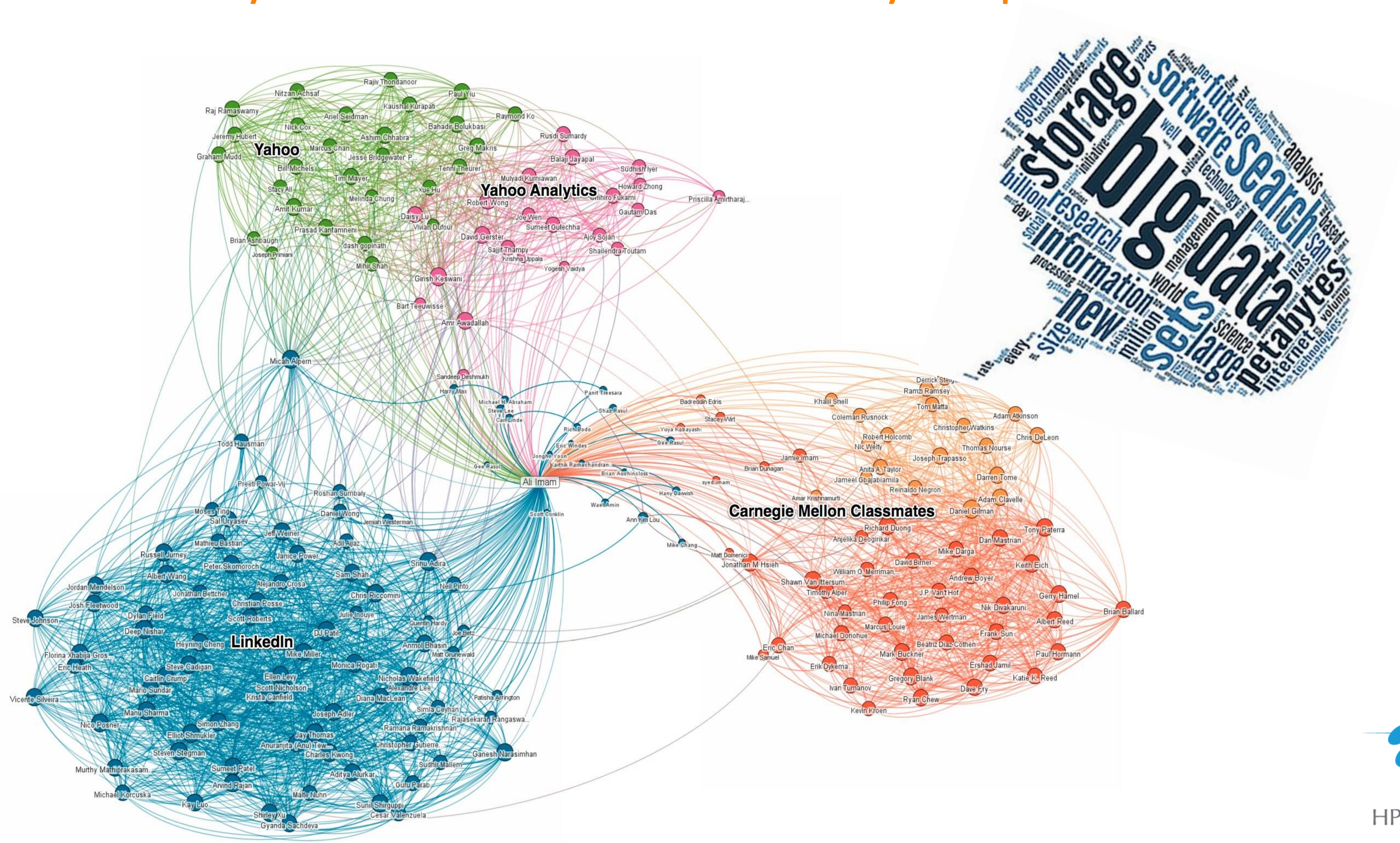
Yinyang K-Means Clustering Algorithm in HPCC Systems

HPCC SYSTEMS®

Lily Xu
PhD Student
Computer Science
Clemson University

CLEMSON UNIVERSITY

How to identify the elusive hubs between your professional worlds?



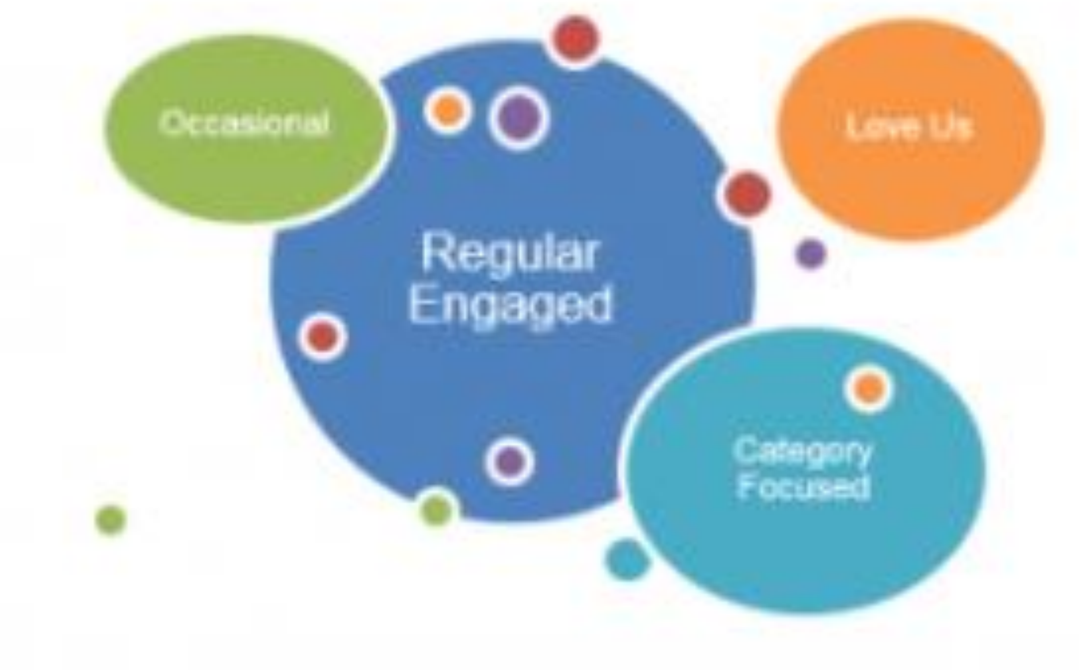
HPCC SYSTEMS®

Pricing segmentation

- Total spend
- Value of discounts
- % discounts across transactions
- Number of items bought on discounts

Cluster the discount orientations of the customers

Are you a loyal consumer?

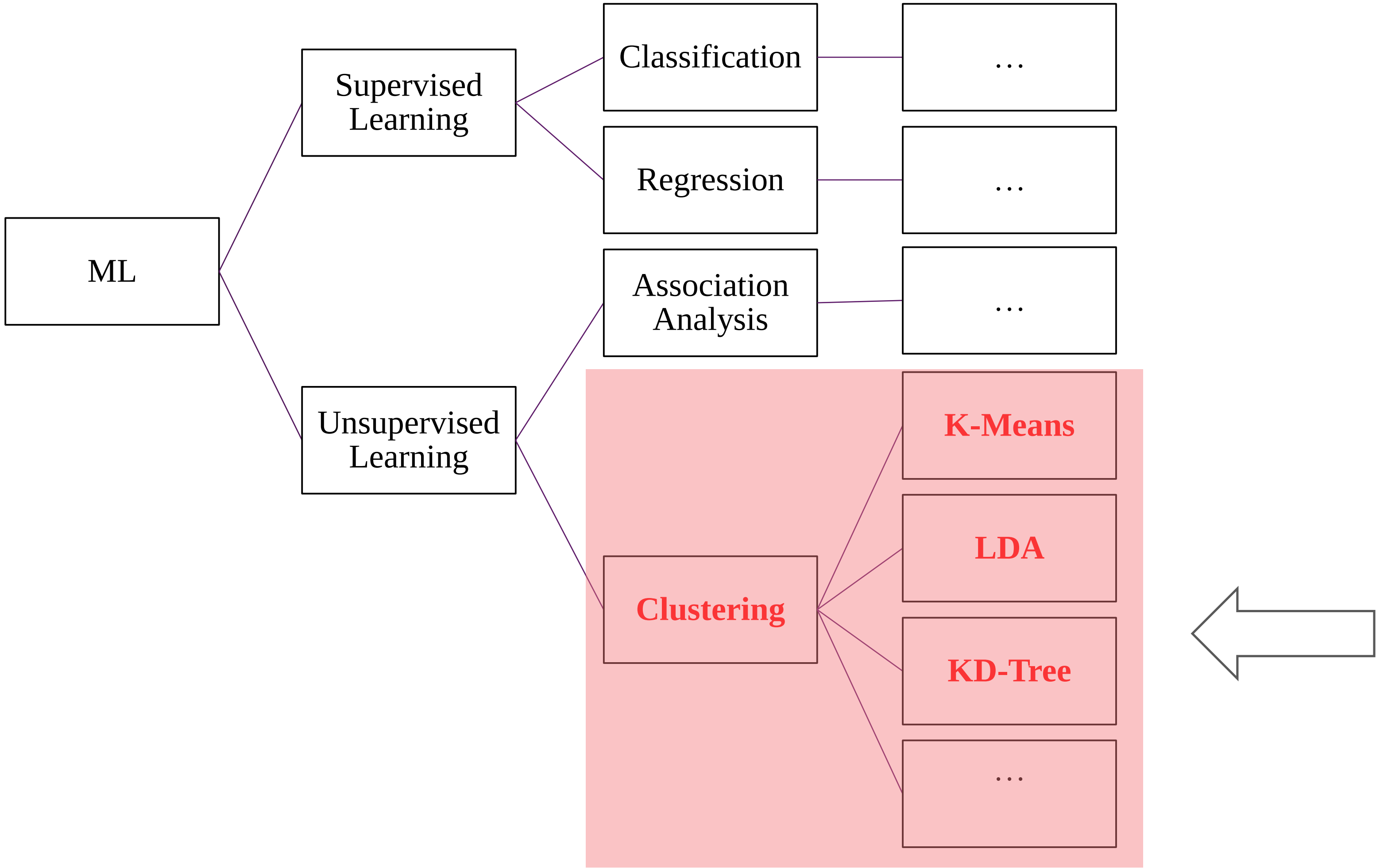


Cluster customers into 4 dimensions

↓

Focus engagement strategy

Machine learning library in HPCC Systems



Yinyang K-Means

Yinyang K-Means: WHAT?

A DROP-IN Replacement of the K-Means Clustering Algorithm

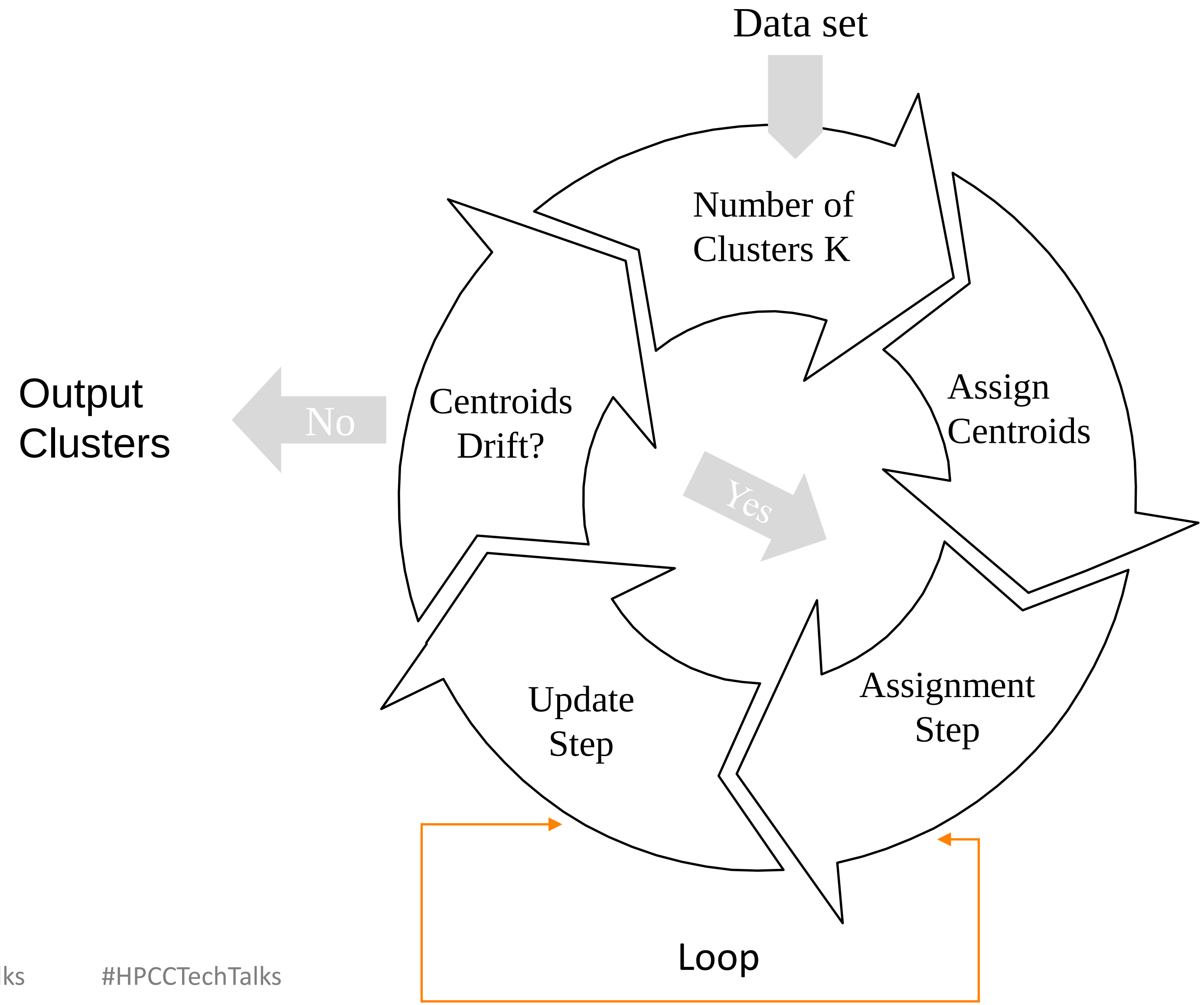
Yinyang K-Means: WHY?

A DROP-IN Replacement of the standard K-Means

Two times to an order of magnitude FASTER

GUARANTEE the same clustering results as the standard K-Means

K-Means clustering algorithm



Data set

Number of Clusters K

Assign Centroids

Assignment Step

Update Step

Centroids Drift?

Yes

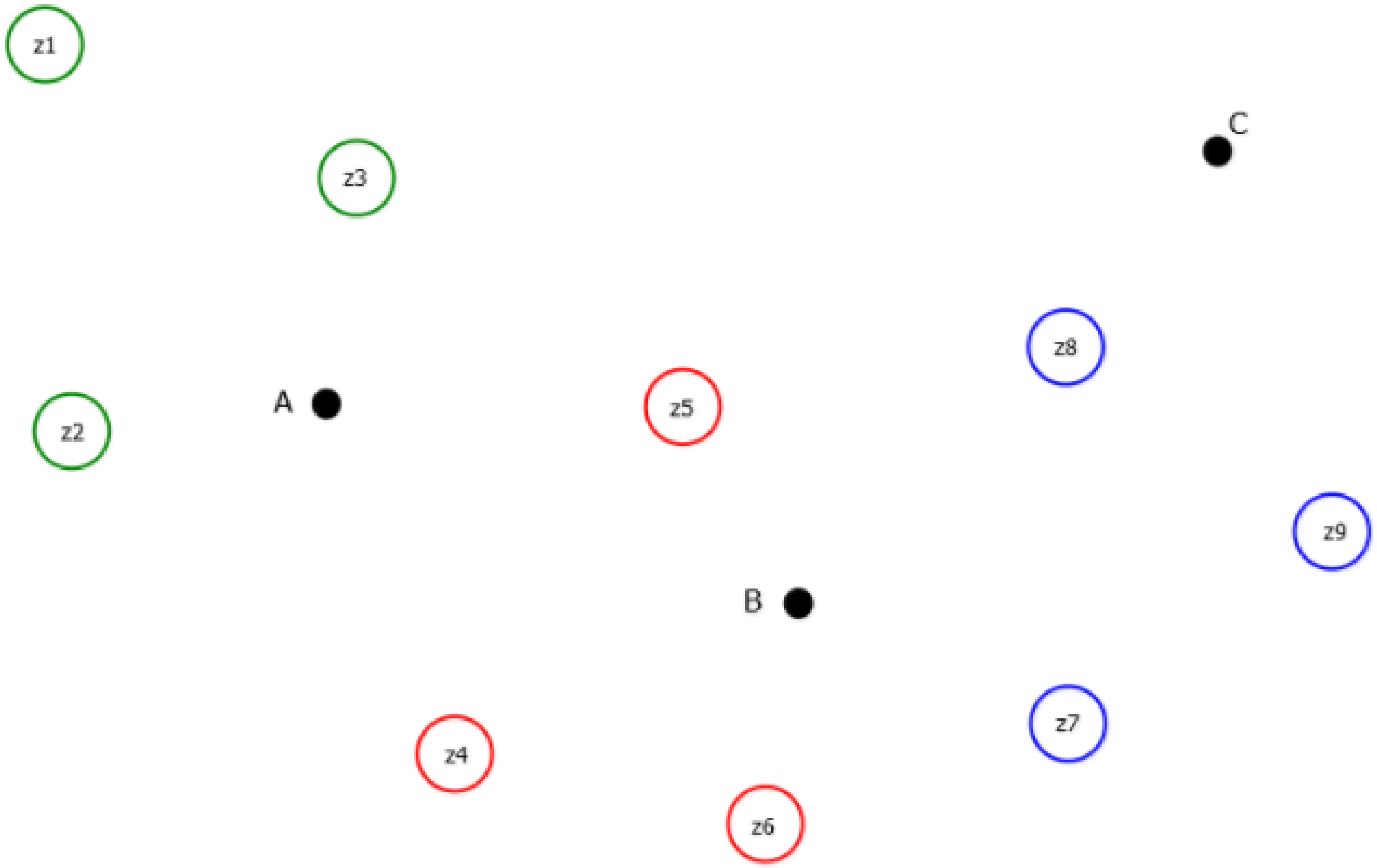
No

Output Clusters

Loop

HPCC SYSTEMS®

K-Means clustering algorithm



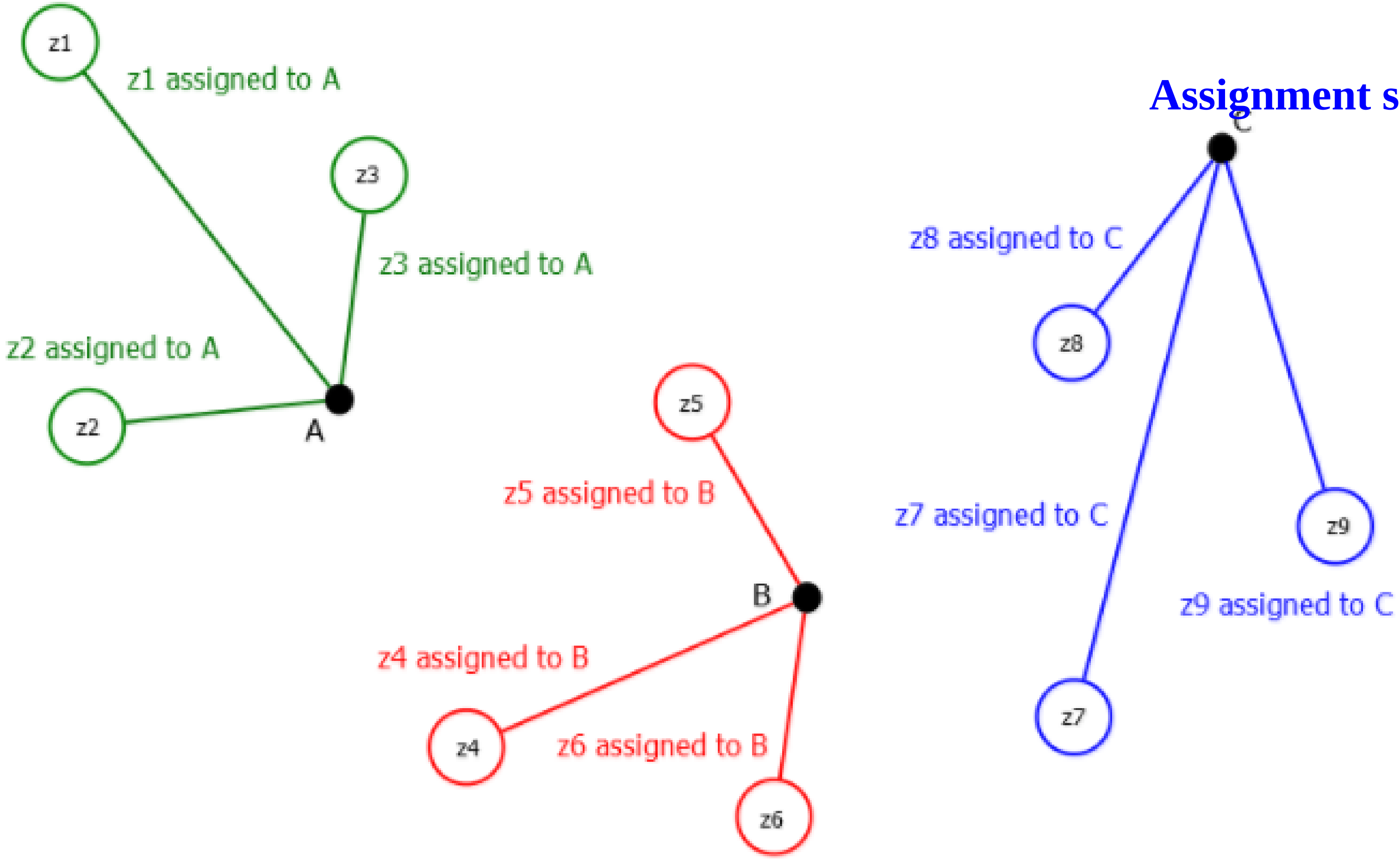
Initialization: Choose K and assign K cluster centroids (randomly)

K-Means clustering algorithm

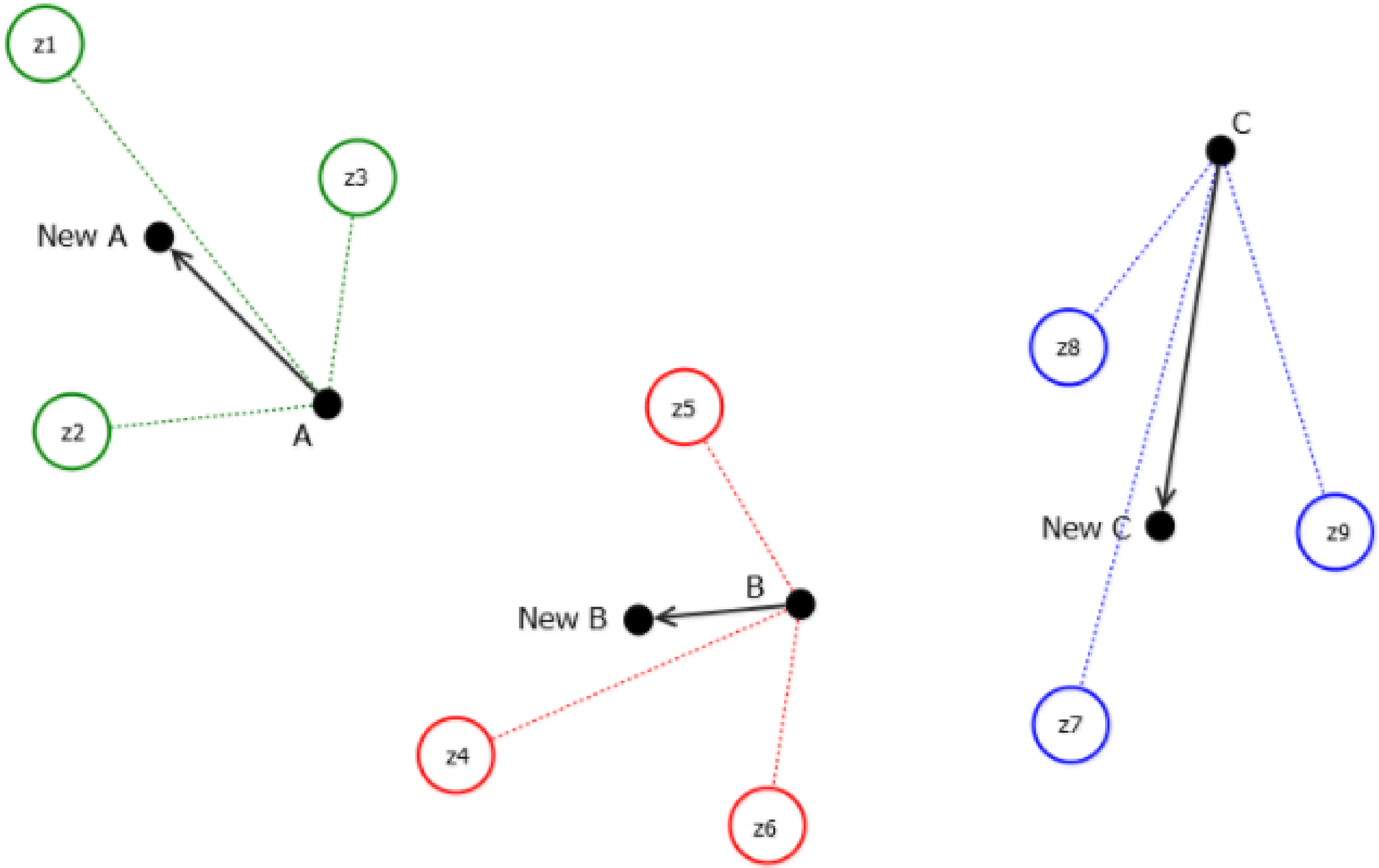
Initialization: Assign K cluster centroids (randomly)

↓

Assignment step : Assign each point to its closest centroid



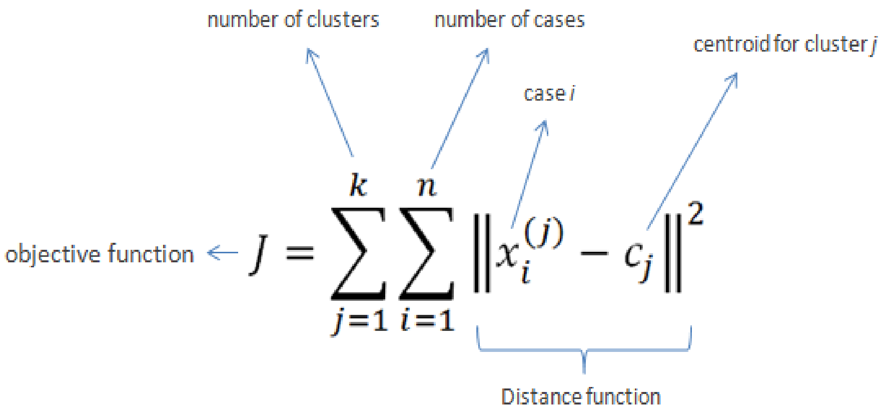
K-Means clustering algorithm



Initialization: Assign K cluster centroids (randomly)

Assignment step : Assign each point to its closest centroid

Update: Re-locate the K centroids



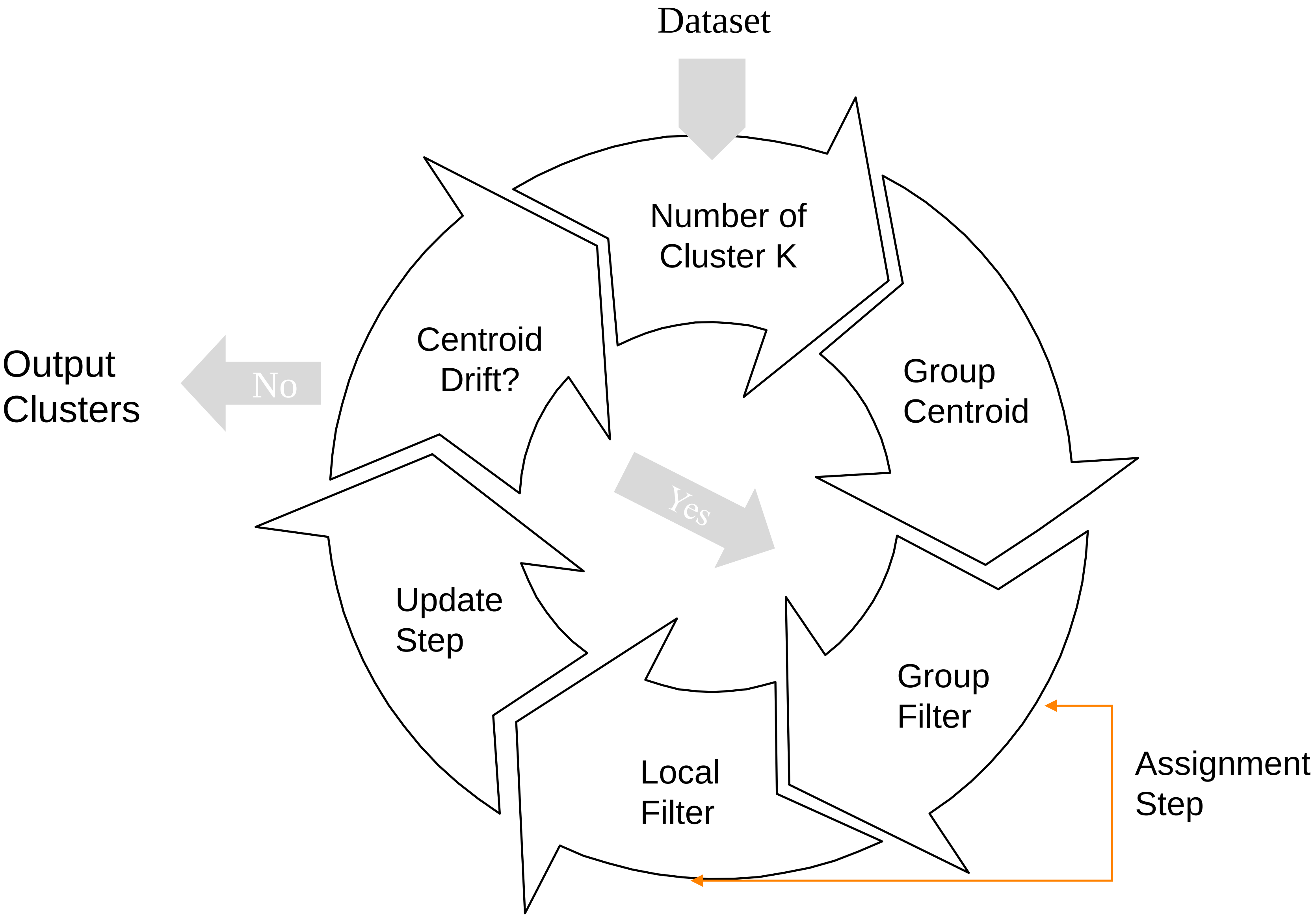
$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|^2$$

Loop?

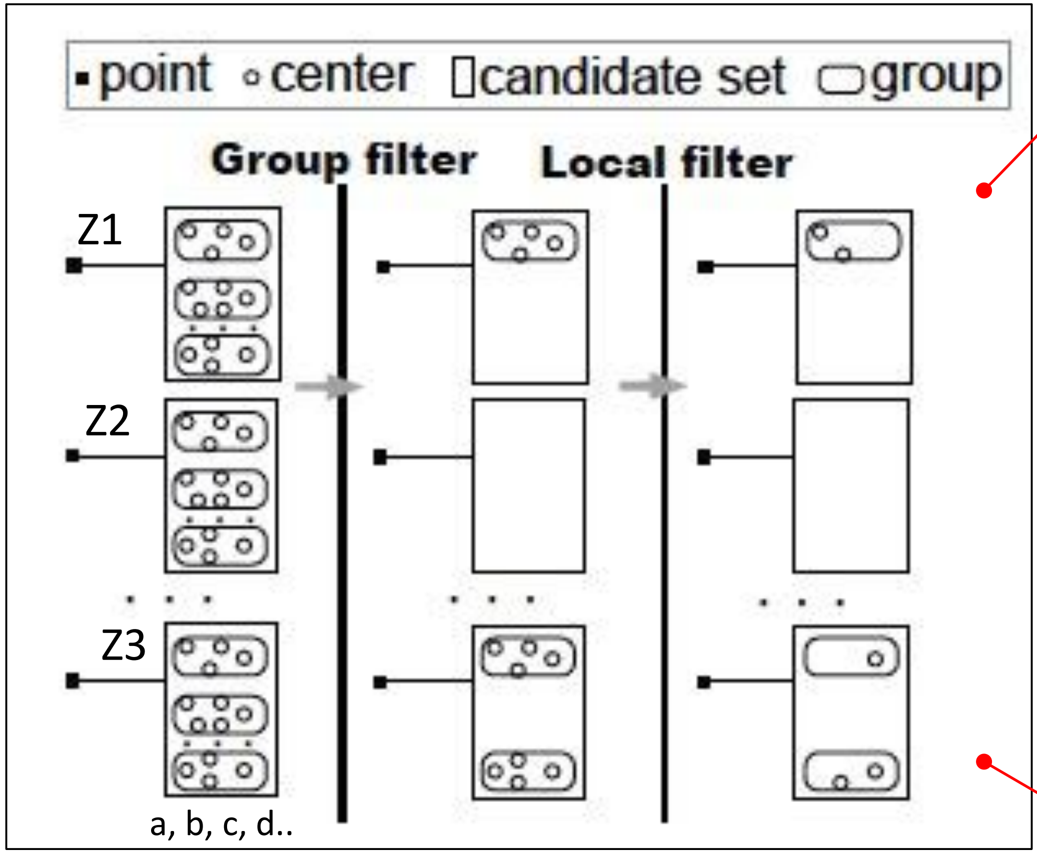
IF $\sum$ (C'-C) < Threshold: Output Clusters

ELSE: Go back to Assignment Step

Yinyang K-Means clustering algorithm



Yinyang K-Means - Assignment Step

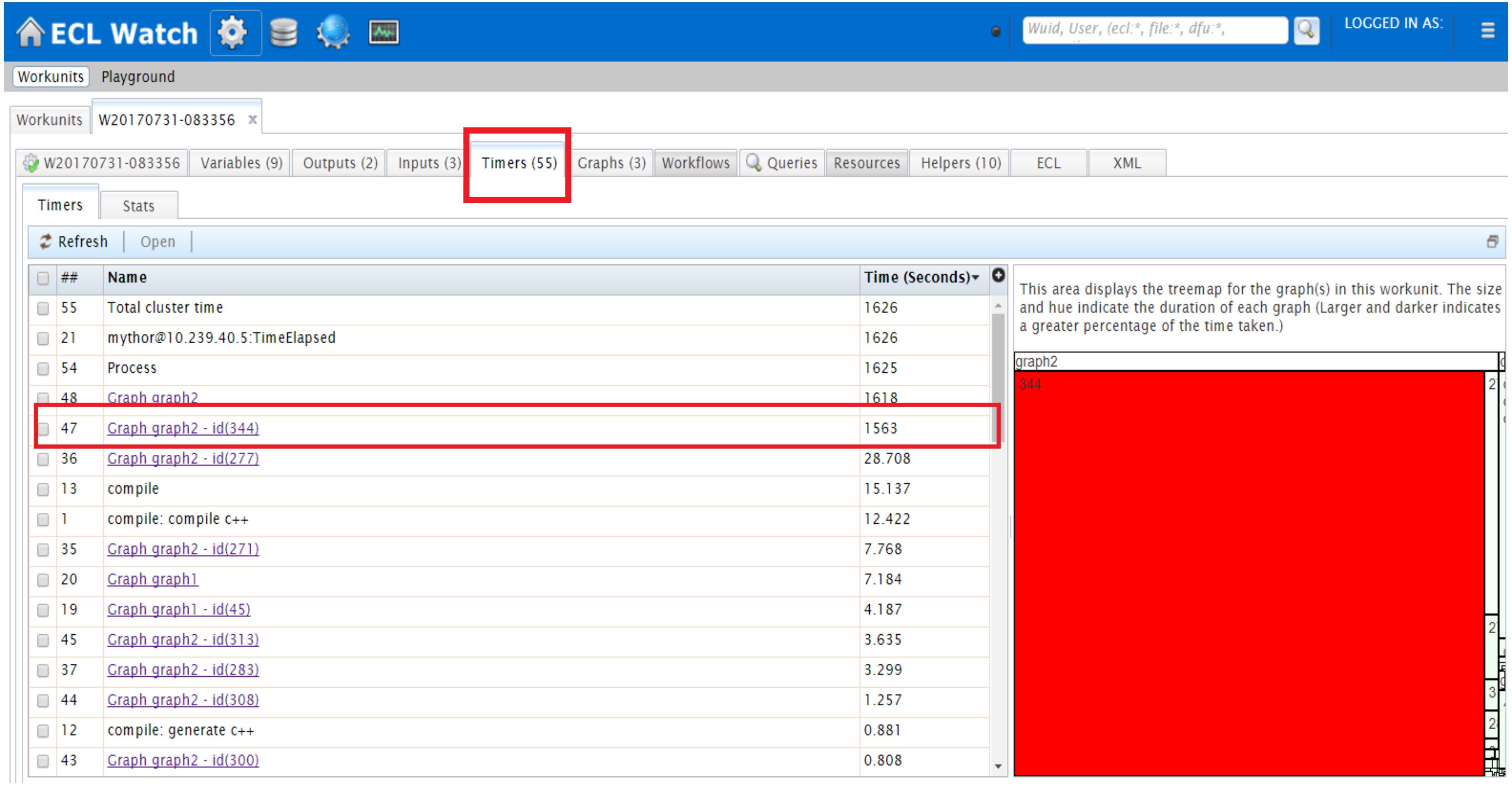


Yinyang K-Means -- Assignment Step

Group filter and local filter optimize the assignment step

Remove unnecessary distance calculations by filtering out unchanged centroids/point pair

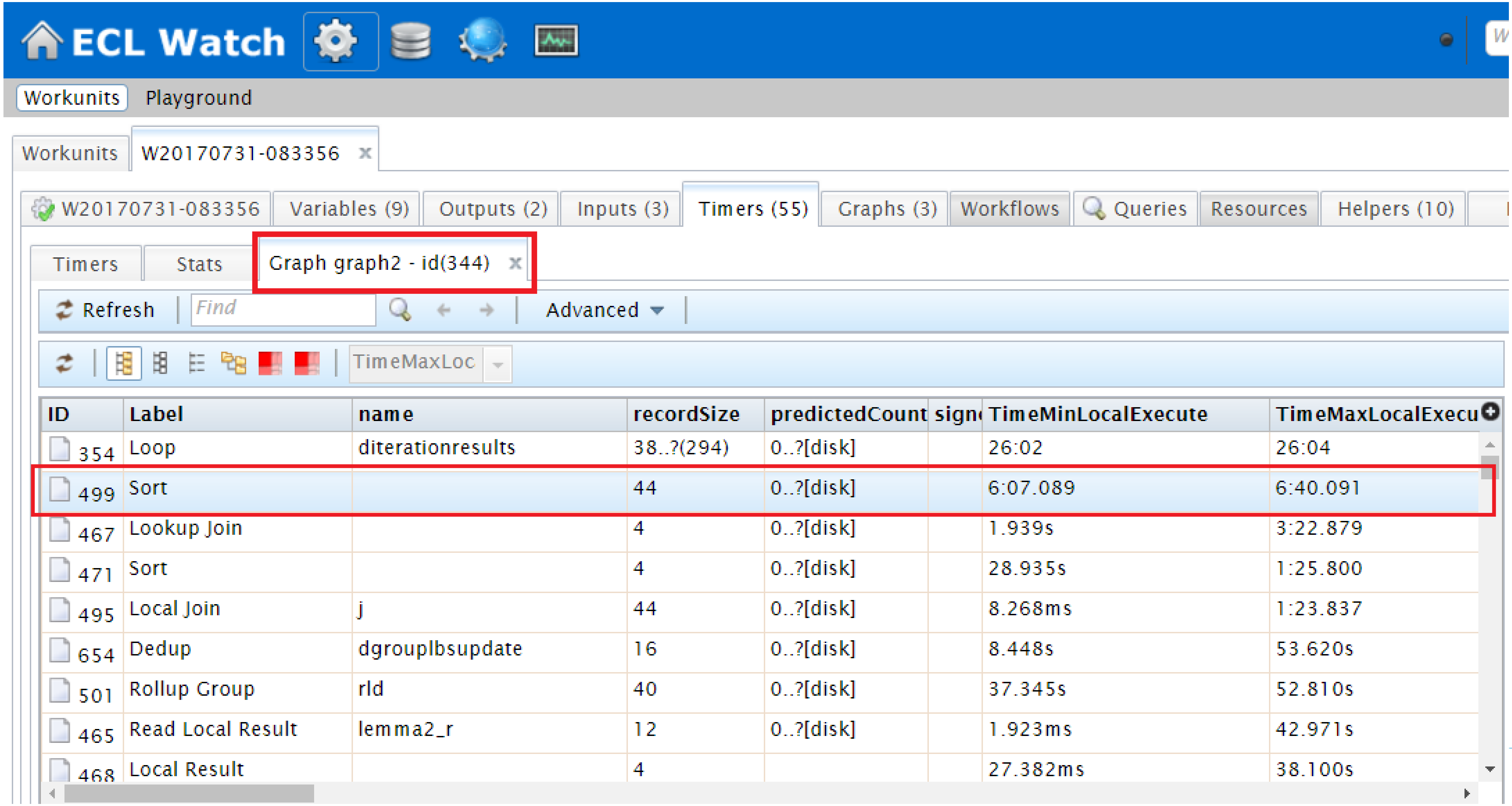
Yinyang K-Means – Performance Analysis in ECL Watch



##	Name	Time (Seconds)
55	Total cluster time	1626
21	mythor@10.239.40.5:TimeElapsed	1626
54	Process	1625
48	Graph graph2	1618
47	Graph graph2 - id(344)	1563
36	Graph graph2 - id(277)	28.708
13	compile	15.137
1	compile: compile c++	12.422
35	Graph graph2 - id(271)	7.768
20	Graph graph1	7.184
19	Graph graph1 - id(45)	4.187
45	Graph graph2 - id(313)	3.635
37	Graph graph2 - id(283)	3.299
44	Graph graph2 - id(308)	1.257
12	compile: generate c++	0.881
43	Graph graph2 - id(300)	0.808

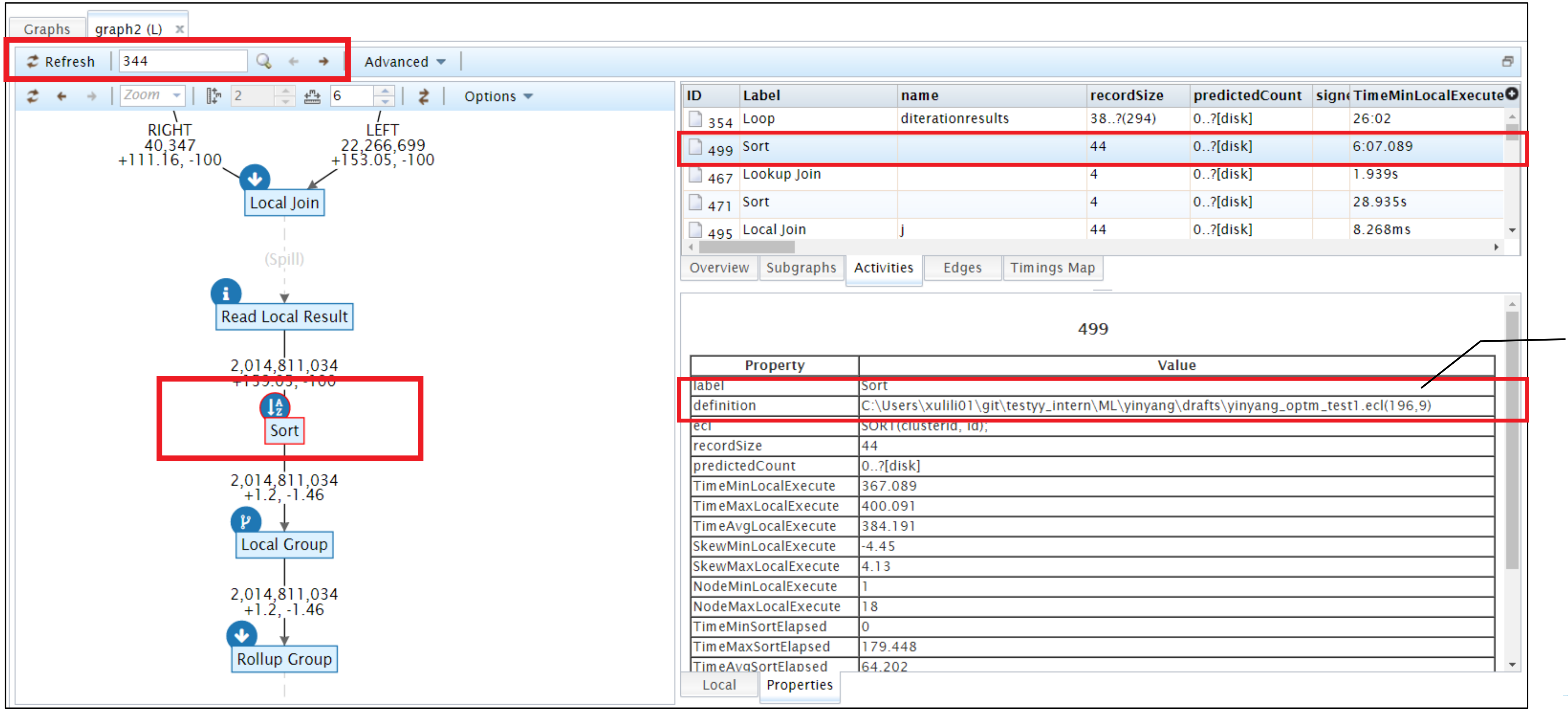
This area displays the treemap for the graph(s) in this workunit. The size and hue indicate the duration of each graph (Larger and darker indicates a greater percentage of the time taken.)

Yinyang K-Means – Graph Analysis in ECL Watch



ID	Label	name	recordSize	predictedCount	signi	TimeMinLocalExecute	TimeMaxLocalExecu
354	Loop	diterationresults	38..?(294)	0..?[disk]		26:02	26:04
499	Sort		44	0..?[disk]		6:07.089	6:40.091
467	Lookup Join		4	0..?[disk]		1.939s	3:22.879
471	Sort		4	0..?[disk]		28.935s	1:25.800
495	Local Join	j	44	0..?[disk]		8.268ms	1:23.837
654	Dedup	dgrouplbsupdate	16	0..?[disk]		8.448s	53.620s
501	Rollup Group	rld	40	0..?[disk]		37.345s	52.810s
465	Read Local Result	lemma2_r	12	0..?[disk]		1.923ms	42.971s
468	Local Result		4			27.382ms	38.100s

Yinyang K-Means – Graph Analysis in ECL Watch



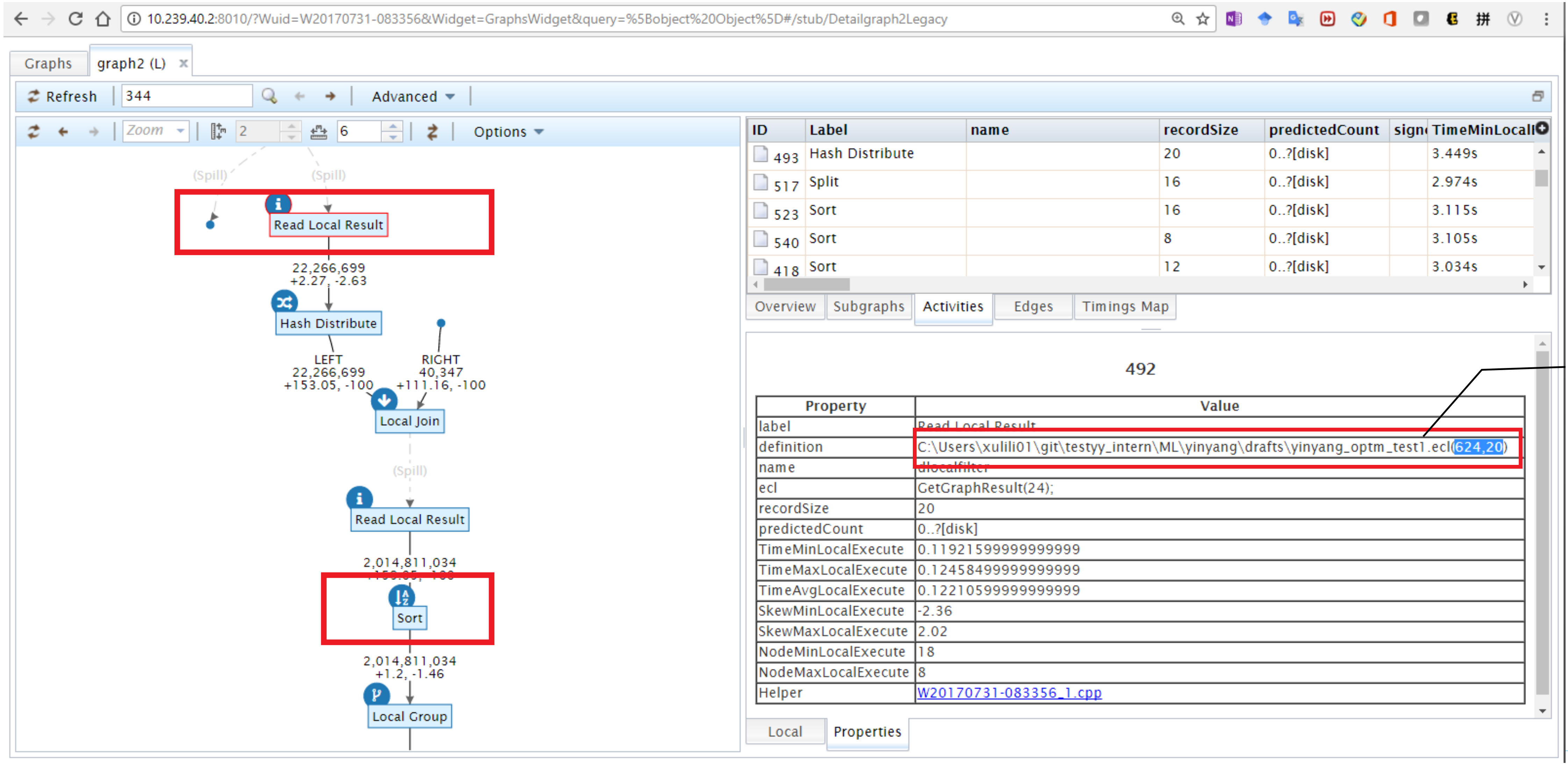
ID	Label	name	recordSize	predictedCount	signi	TimeMinLocalExecute
354	Loop	diterationresults	38..?(294)	0..?[disk]		26:02
499	Sort		44	0..?[disk]		6:07.089
467	Lookup Join		4	0..?[disk]		1.939s
471	Sort		4	0..?[disk]		28.935s
495	Local Join	j	44	0..?[disk]		8.268ms

499

Property	Value
label	Sort
definition	C:\Users\xulili01\git\testyy_intern\ML\yinyang\drafts\yinyang_optm_test1.ecl(196,9)
ecl	SORT(clusterid, id);
recordSize	44
predictedCount	0..?[disk]
TimeMinLocalExecute	367.089
TimeMaxLocalExecute	400.091
TimeAvgLocalExecute	384.191
SkewMinLocalExecute	-4.45
SkewMaxLocalExecute	4.13
NodeMinLocalExecute	1
NodeMaxLocalExecute	18
TimeMinSortElapsed	0
TimeMaxSortElapsed	179.448
TimeAvgSortElapsed	64.202

Line 196 : Helper Function

Yinyang K-Means – Graph Analysis in ECL Watch



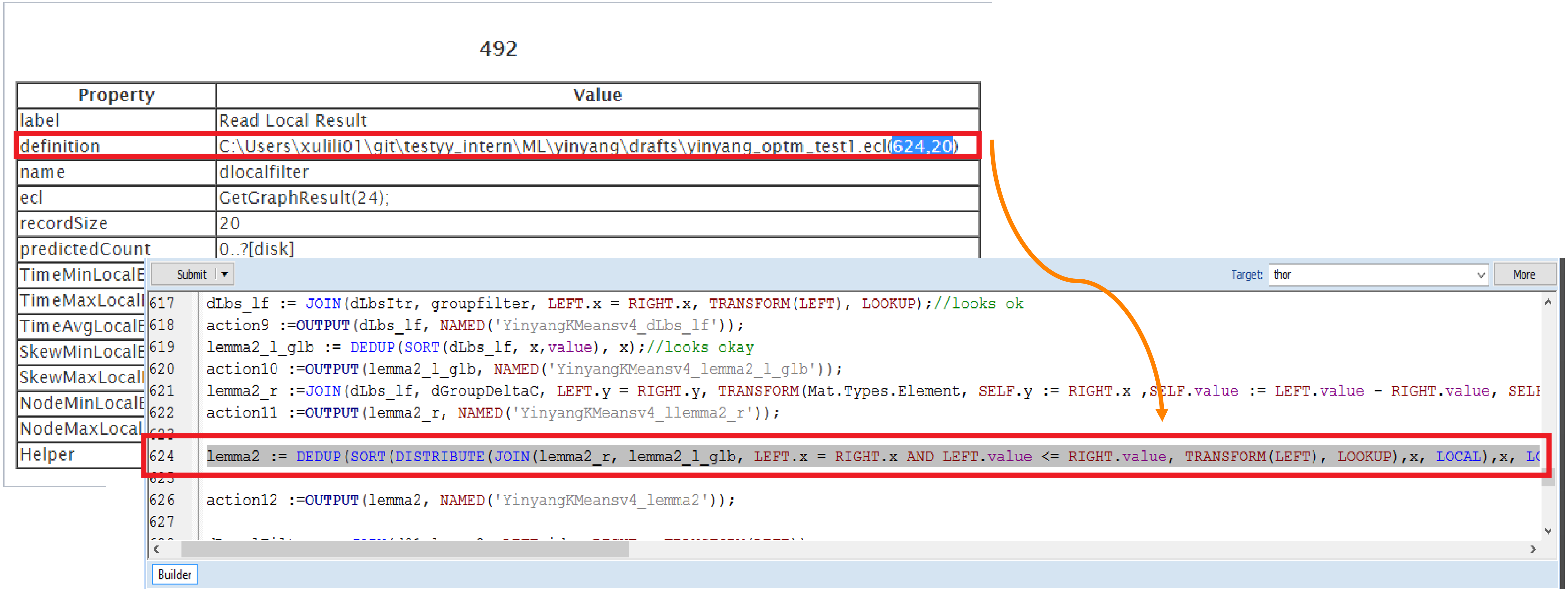
ID	Label	name	recordSize	predictedCount	sign	TimeMinLocal
493	Hash Distribute		20	0..?[disk]		3.449s
517	Split		16	0..?[disk]		2.974s
523	Sort		16	0..?[disk]		3.115s
540	Sort		8	0..?[disk]		3.105s
418	Sort		12	0..?[disk]		3.034s

492

Property	Value
label	Read Local Result
definition	C:\Users\xulili01\git\testyy_intern\ML\yinyang\drafts\yinyang_optm_test1.ecl(624,20)
name	dlocalfilter
ecl	GetGraphResult(24);
recordSize	20
predictedCount	0..?[disk]
TimeMinLocalExecute	0.11921599999999999
TimeMaxLocalExecute	0.12458499999999999
TimeAvgLocalExecute	0.12210599999999999
SkewMinLocalExecute	-2.36
SkewMaxLocalExecute	2.02
NodeMinLocalExecute	18
NodeMaxLocalExecute	8
Helper	W20170731-083356_1.cpp

Line 624: Code calls the helper function

Yinyang K-Means – Code Analysis in ECL



492

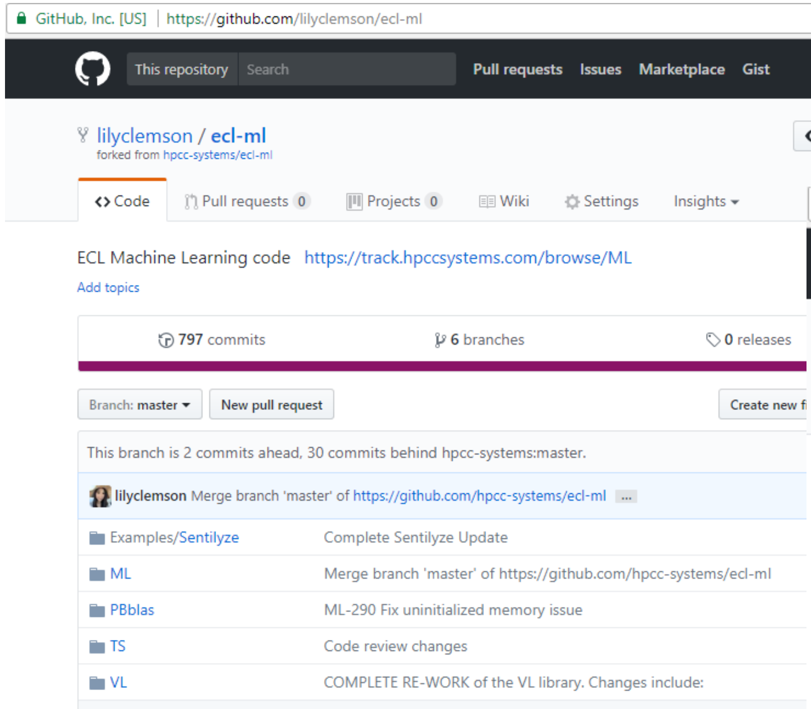
Property	Value
label	Read Local Result
definition	C:\Users\xulili01\git\testyy_intern\ML\yinyang\drafts\yinyang_optm_test1.ecl(624,20)
name	dlocalfilter
ecl	GetGraphResult(24);
recordSize	20
predictedCount	0..?[disk]
TimeMinLocalE	
TimeMaxLocalI	
TimeAvgLocalE	
SkewMinLocalE	
SkewMaxLocalI	
NodeMinLocalE	
NodeMaxLocal	
Helper	

```
617 dLbs_lf := JOIN(dLbsItr, groupfilter, LEFT.x = RIGHT.x, TRANSFORM(LEFT), LOOKUP);//looks ok
618 action9 :=OUTPUT(dLbs_lf, NAMED('YinyangKMeansv4_dLbs_lf'));
619 lemma2_l_glb := DEDUP(SORT(dLbs_lf, x,value), x);//looks okay
620 action10 :=OUTPUT(lemma2_l_glb, NAMED('YinyangKMeansv4_lemma2_l_glb'));
621 lemma2_r :=JOIN(dLbs_lf, dGroupDeltaC, LEFT.y = RIGHT.y, TRANSFORM(Mat.Types.Element, SELF.y := RIGHT.x ,SELF.value := LEFT.value - RIGHT.value, SELF
622 action11 :=OUTPUT(lemma2_r, NAMED('YinyangKMeansv4_llemma2_r'));
623
624 lemma2 := DEDUP(SORT(DISTRIBUTE(JOIN(lemma2_r, lemma2_l_glb, LEFT.x = RIGHT.x AND LEFT.value <= RIGHT.value, TRANSFORM(LEFT), LOOKUP),x, LOCAL),x, LO
625
626 action12 :=OUTPUT(lemma2, NAMED('YinyangKMeansv4_lemma2'));
627
```

Yinyang K-Means – Optimization

- Optimize the sequential algorithm by recognizing distributable or inefficient component in the distributed environment
- Add global filter and combine with group filter to avoid massive communication
- Distribute dataset/recordset smartly to avoid unnecessary communication

Yinyang K-Means – Code check-in & Code review



GitHub, Inc. [US] | https://github.com/lilyclemson/ecl-ml
This repository | Search | Pull requests | Issues | Marketplace | Gist
Unwatch 1 | Star 0 | Fork 56
Code | Pull requests 0 | Projects 0 | Wiki | Settings | Insights
Branch: master | New pull request | Create new f

lilyclemson / ecl-ml
forked from hpcc-systems/ecl-ml

ECL Machine Learning code https://track.hpccsystems.com/browse/ML

Add topics

797 commits | 6 branches | 0 releases

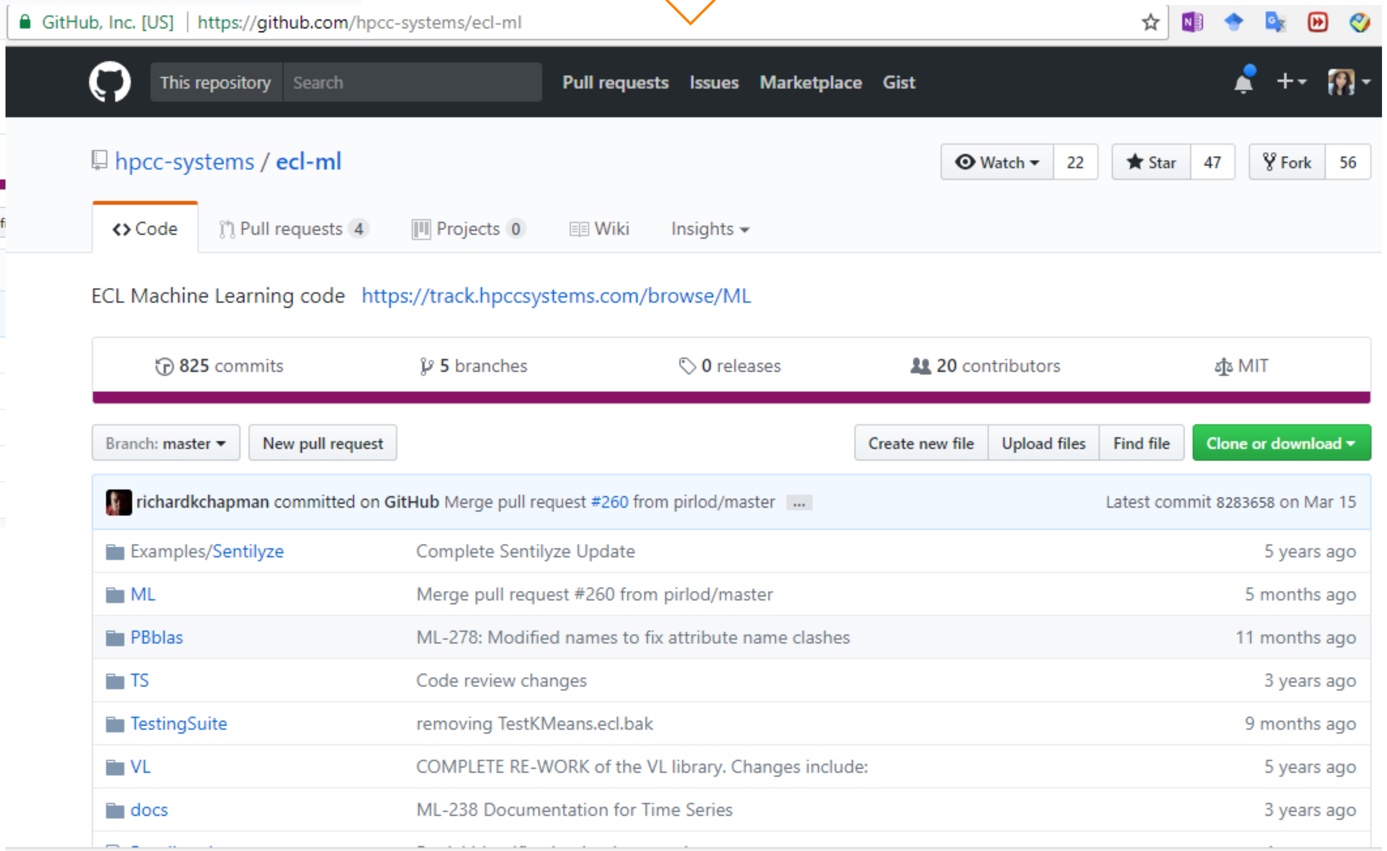
This branch is 2 commits ahead, 30 commits behind hpcc-systems:master.

lilyclemson Merge branch 'master' of https://github.com/hpcc-systems/ecl-ml ...

Examples/Sentilyze	Complete Sentilyze Update
ML	Merge branch 'master' of https://github.com/hpcc-systems/ecl-ml
PBblas	ML-290 Fix uninitialized memory issue
TS	Code review changes
VL	COMPLETE RE-WORK of the VL library. Changes include:

Lily's Github Account

Pull Request



GitHub, Inc. [US] | https://github.com/hpcc-systems/ecl-ml
This repository | Search | Pull requests | Issues | Marketplace | Gist
Watch 22 | Star 47 | Fork 56
Code | Pull requests 4 | Projects 0 | Wiki | Insights
Branch: master | New pull request | Create new file | Upload files | Find file | Clone or download

hpcc-systems / ecl-ml

ECL Machine Learning code https://track.hpccsystems.com/browse/ML

825 commits | 5 branches | 0 releases | 20 contributors | MIT

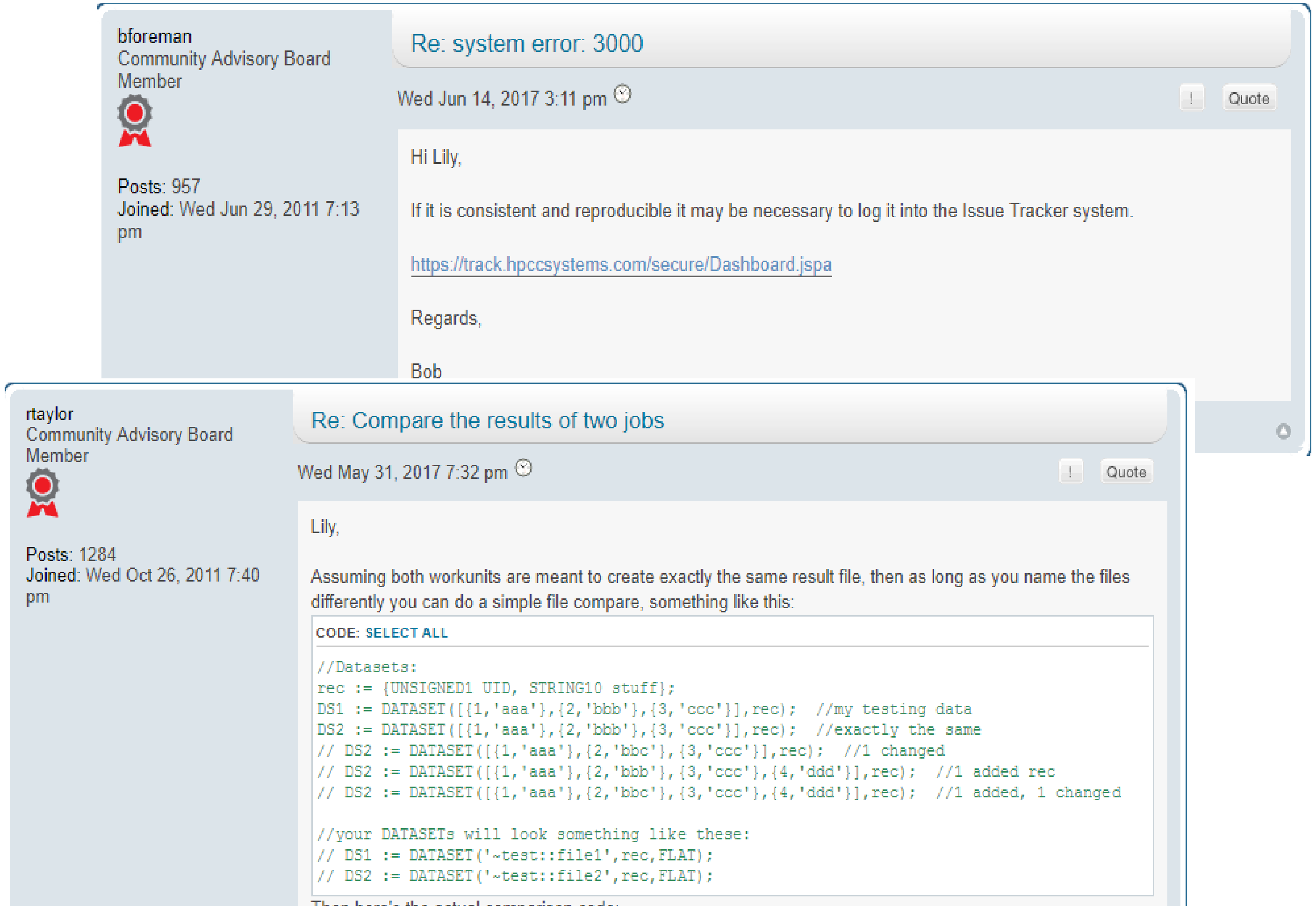
richardkchapman committed on GitHub Merge pull request #260 from pirlod/master ... Latest commit 8283658 on Mar 15

Examples/Sentilyze	Complete Sentilyze Update	5 years ago
ML	Merge pull request #260 from pirlod/master	5 months ago
PBblas	ML-278: Modified names to fix attribute name clashes	11 months ago
TS	Code review changes	3 years ago
TestingSuite	removing TestKMeans.ecl.bak	9 months ago
VL	COMPLETE RE-WORK of the VL library. Changes include:	5 years ago
docs	ML-238 Documentation for Time Series	3 years ago

HPCC Systems Official Github Account

Intern experience

1. Good communication
   - Mentor
   - Colleagues
2. Where to get help
   - HPCC Systems Forum
   - Online searching
   - Mentor
   - Colleagues
3. Work & Life Balance
   - On Campus Gym
   - Braves Game



bforeman
Community Advisory Board Member

Posts: 957
Joined: Wed Jun 29, 2011 7:13 pm

Re: system error: 3000

Wed Jun 14, 2017 3:11 pm

Hi Lily,

If it is consistent and reproducible it may be necessary to log it into the Issue Tracker system.

https://track.hpccsystems.com/secure/Dashboard.jspa

Regards,

Bob

rtaylor
Community Advisory Board Member

Posts: 1284
Joined: Wed Oct 26, 2011 7:40 pm

Re: Compare the results of two jobs

Wed May 31, 2017 7:32 pm

Lily,

Assuming both workunits are meant to create exactly the same result file, then as long as you name the files differently you can do a simple file compare, something like this:

CODE: SELECT ALL

```
//Datasets:
rec := {UNSIGNED1 UID, STRING10 stuff};
DS1 := DATASET([{1,'aaa'},{2,'bbb'},{3,'ccc'}],rec);  //my testing data
DS2 := DATASET([{1,'aaa'},{2,'bbb'},{3,'ccc'}],rec);  //exactly the same
// DS2 := DATASET([{1,'aaa'},{2,'bbc'},{3,'ccc'}],rec);  //1 changed
// DS2 := DATASET([{1,'aaa'},{2,'bbb'},{3,'ccc'},{4,'ddd'}],rec);  //1 added rec
// DS2 := DATASET([{1,'aaa'},{2,'bbc'},{3,'ccc'},{4,'ddd'}],rec);  //1 added, 1 changed

//your DATASETs will look something like these:
// DS1 := DATASET('~test::file1',rec,FLAT);
// DS2 := DATASET('~test::file2',rec,FLAT);
```

Then here's the actual comparison code:

HPCC Systems Community Forum

Acknowledgements

The internship opportunity I had with LexisNexis Risk Solutions was a great chance for learning and professional development. I am using this opportunity to express my deepest gratitude and special thanks to my mentor Edin Muharemagic who always there to guide and keep me on the correct path. Thank you John Holt who gives tremendous technical guidance and help. Thank you Gavin Halliday and Jake Smith and the operations team who is always there to give me systems support. I express thanks to Lorraine Chapman and Midha Renu who organized this amazing summer internship program.

HPCC SYSTEMS®

References

1. Bottesch, T., Bühler, T., & Kächele, M. (2016). Speeding up k-means by approximating Euclidean distances via block vectors. In Proceedings of The 33rd International Conference on Machine Learning (pp. 2578-2586)

2. Ding, Y., Zhao, Y., Shen, X., Musuvathi, M., & Mytkowicz, T. (2015). Yinyang k-means: A drop-in replacement of the classic k-means with consistent speedup. In Proceedings of the 32nd International Conference on Machine Learning (ICML-15) (pp. 579-587)

3. Bache, K. and Lichman, M. UCI machine learning repository, 2013. URL http://archive.ics.uci.edu/ml

4. Visualize your LinkedIn network with InMaps, https://blog.linkedin.com/2011/01/24/linkedin-inmaps

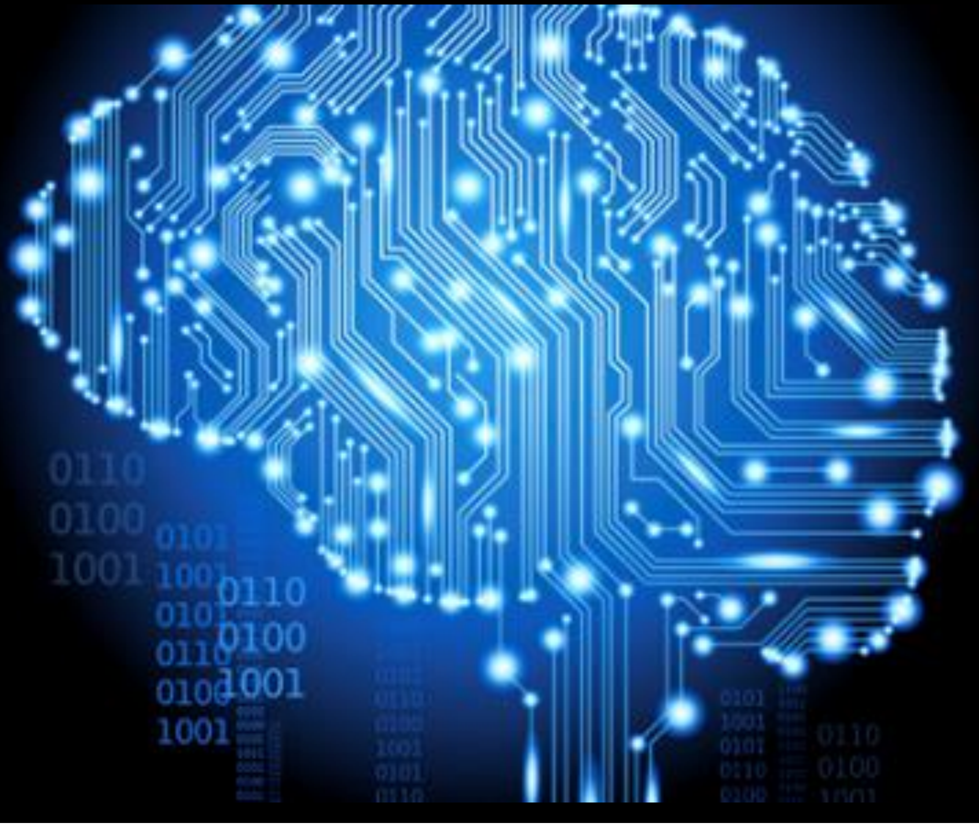
Questions?

Lily Xu

PhD Student, Computer Science,
Clemson University
lilix@g.clemson.edu

THE DOWNLOAD

TECH TALKS BY HPCC SYSTEMS



HPCC SYSTEMS®

Gradient Boosting Trees

George Mathew
PhD Student
Computer Science
North Carolina State University

NC STATE UNIVERSITY

Motivation

“Mistakes have the power to turn you into something better”

-- Anonymous

Gradient Boosting 101

- Empower the weak.
- Can work on different learner types:
  - Regression
  - Classification
- Gradient Boosting = Gradient Descent + Boosting
- Award Winning[1]

1. Chapelle, Olivier, and Yi Chang. "Yahoo! learning to rank challenge overview." Proceedings of the Learning to Rank Challenge. 2011.

Gradient Descent - The Math

- Incremental optimization
- Move towards direction of greatest change

$$y = F(X) + \gamma$$

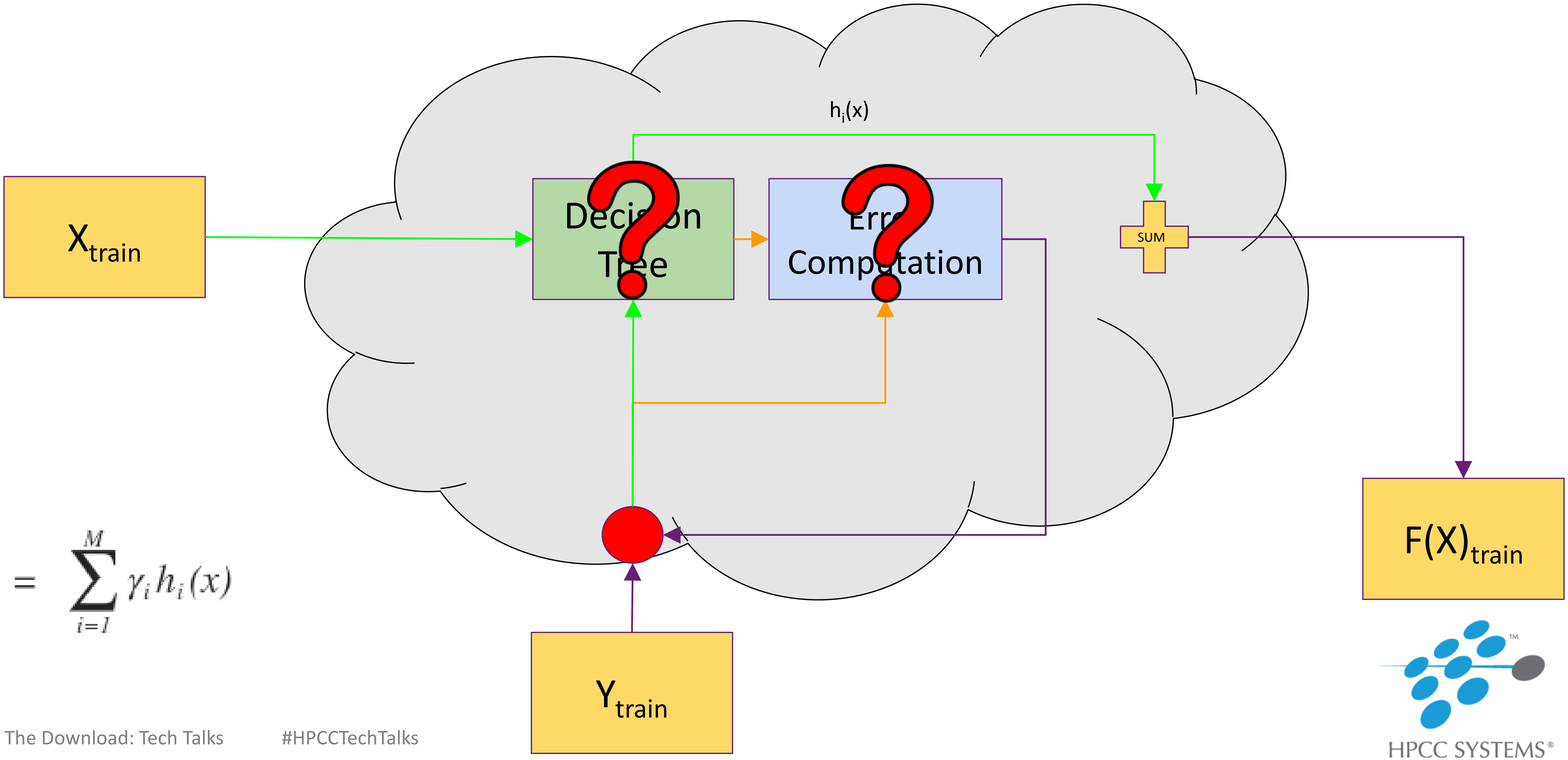
$$F_{i+1}(X) = F_i(X) + \gamma_i$$

$$F_{i+1}(X) \approx F_i(X) + (y - F_i(X))$$

$$F_{i+1}(X) \approx F_i(X) - \rho \frac{\partial L(y, F_i(X))}{\partial F_i(X)}$$

- X = Independent
- y = Dependent
- F(X) = Predicted
- $\boldsymbol{\gamma}$ = Error
- L = Loss function

Gradient Boosting Trees - High Level Block Diagram



$$F(x) = \sum_{i=1}^{M} \gamma_i h_i(x)$$

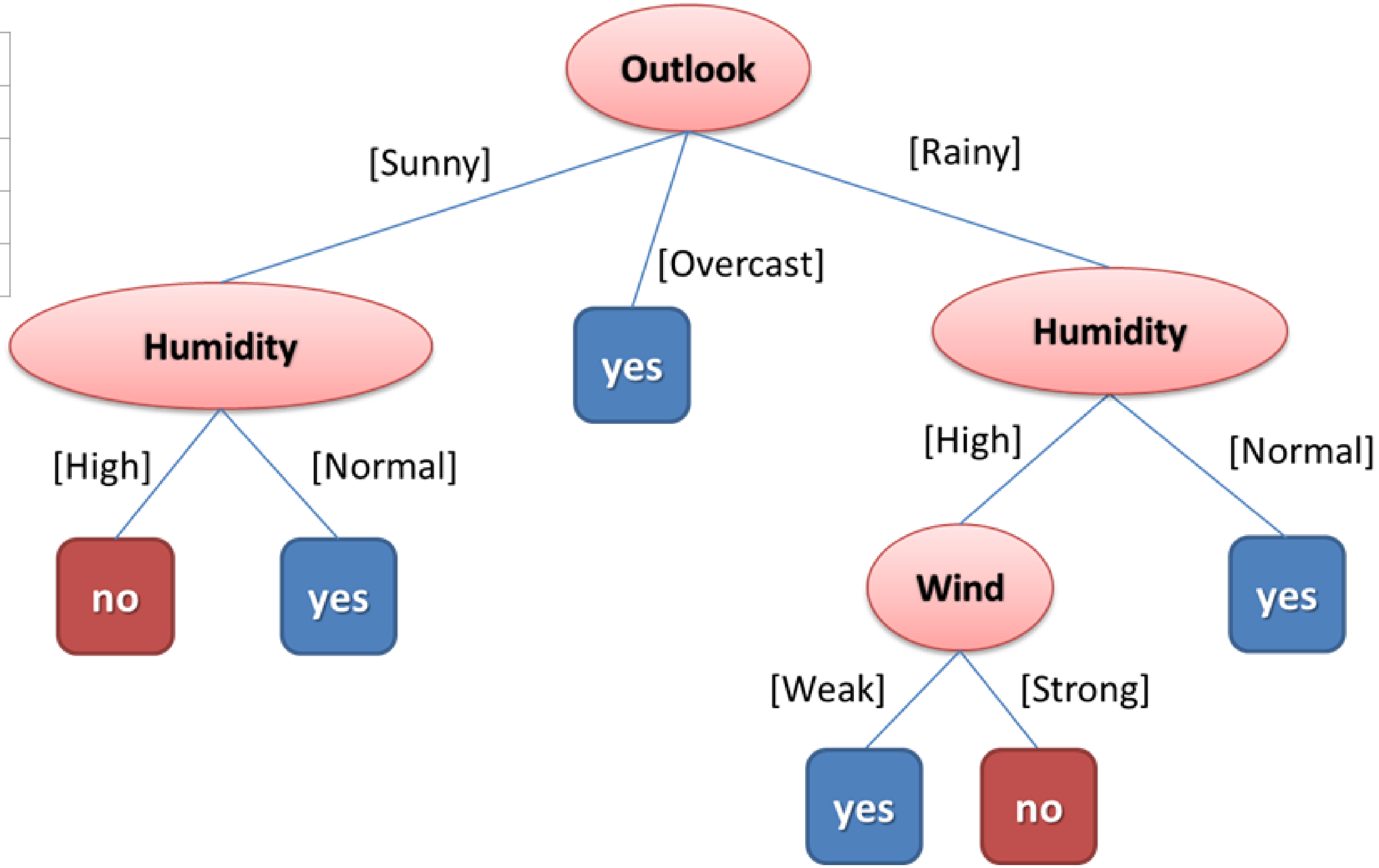
Decision Tree - Dataset

Outlook	S	S	O	R	R	R	O	S	S	R	S	O	O	R
Temp	H	H	H	M	C	C	C	M	C	M	M	M	H	M
Humidity	H	H	H	H	N	N	N	H	N	N	N	H	N	H
Wind	W	S	W	W	W	S	S	W	W	W	S	S	W	S
Play	N	N	Y	Y	Y	N	Y	N	Y	Y	Y	Y	Y	N

Outlook ∈ {**S**unny, **O**vercast, **R**ain}
Temp ∈ {**H**ot, **M**ild, **C**ool}
Humidity ∈ {**N**ormal, **H**igh}
Wind ∈ {**W**eak, **S**trong}
Play ∈ {**Y**es, **N**o}

Decision Tree - Process

Outlook	S	S	O	R	R	R	O	S	S	R	S	O	O	R
Temp	H	H	H	M	C	C	C	M	C	M	M	M	H	M
Humidity	H	H	H	H	N	N	N	H	N	N	N	H	N	H
Wind	W	S	W	W	W	S	S	W	W	W	S	S	W	S
Play	N	N	Y	Y	Y	N	Y	N	Y	Y	Y	Y	Y	N



- Select an attribute to split based on **Splitting Criteria**
- Split instances based on the attribute
- Repeat recursively for each attribute unless a node has purely one class.

HPCC SYSTEMS®

Decision Tree - Splitting Criteria

- **Gini**: Sum squared probability of majority class

$$Gini(E) = 1 - \sum_{j=1}^{c} p_j^2$$

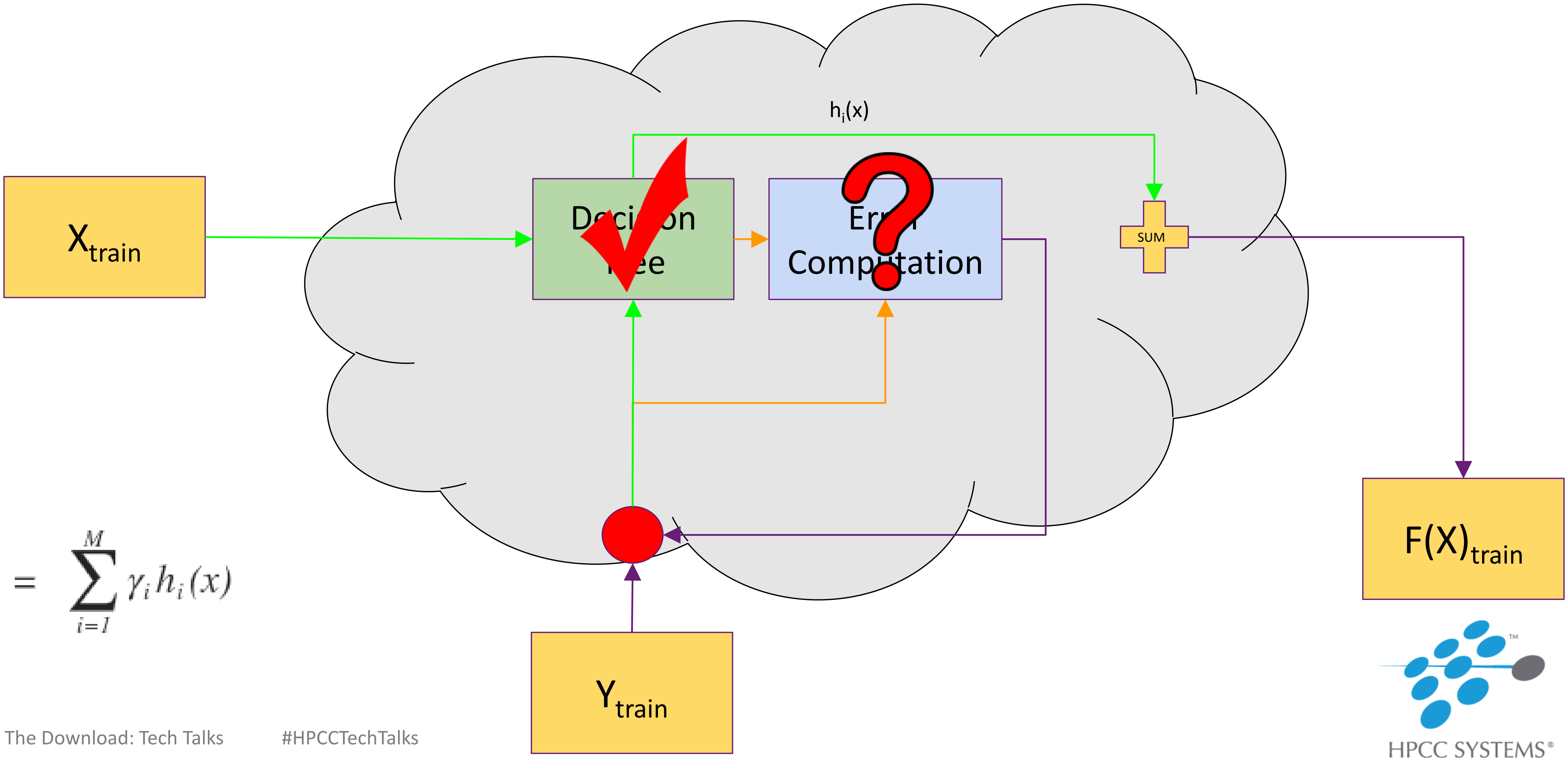
- **Entropy**: Sum of probability and log probability of majority class

$$Entropy(E) = - \sum_{j=1}^{c} p_j \log p_j$$

- **Variance:** Difference b/w Var. in class and Var. in class given attribute

$$Variance(E) = Var(C) - \sum_{j=1}^{c} Var(C|E_j)$$

Gradient Boosting Trees - High Level Block Diagram



$$F(x) = \sum_{i=1}^{M} \gamma_i h_i(x)$$

Error Computation

- **Absolute Loss**

$$L(y, F) = |y - F|$$

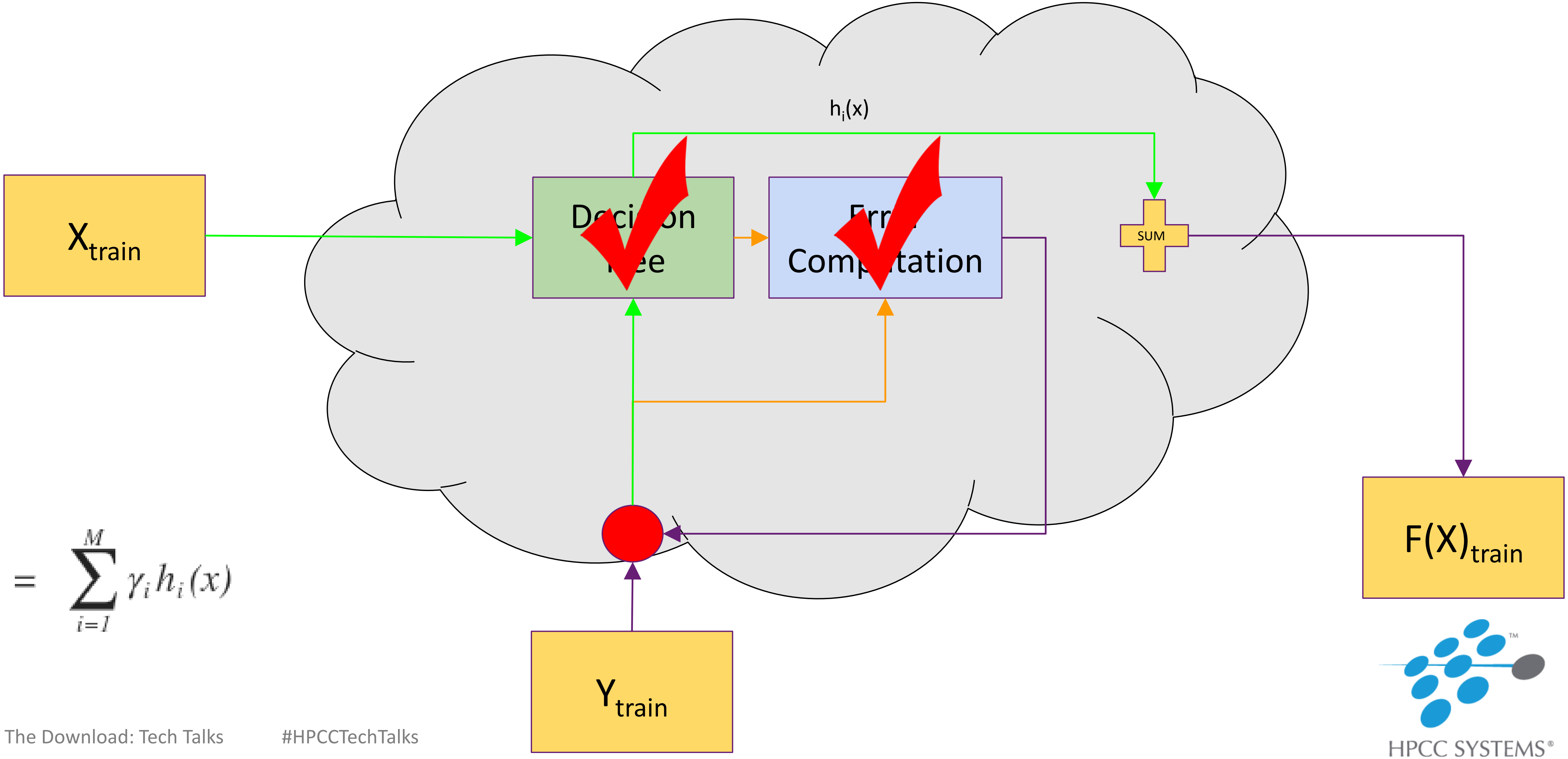
- **Square Loss**

$$L(y, F) = (y - F)^2$$

- **Huber Loss**

$$L(y, F) = \begin{cases} y - F & |y - F| \leq \delta \\ \delta\ sign(y - F) & |y - F| > \delta \end{cases}$$

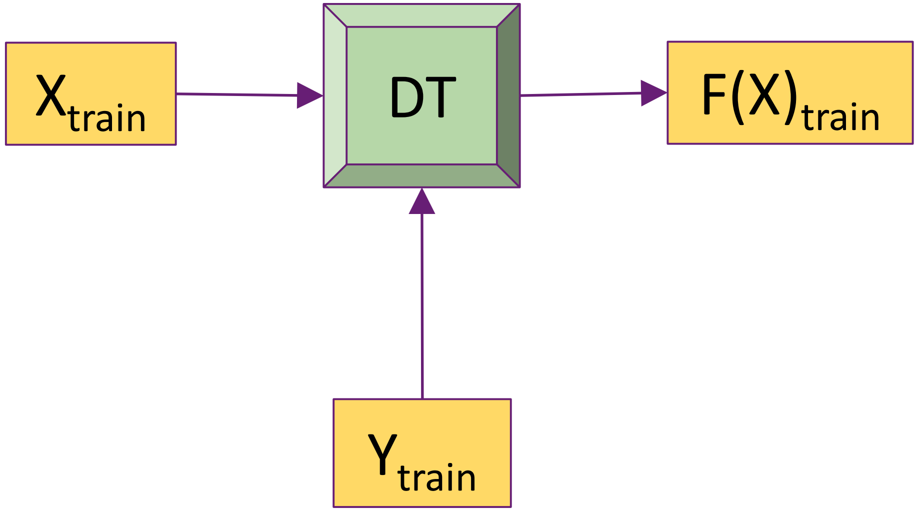
Gradient Boosting Trees - High Level Block Diagram



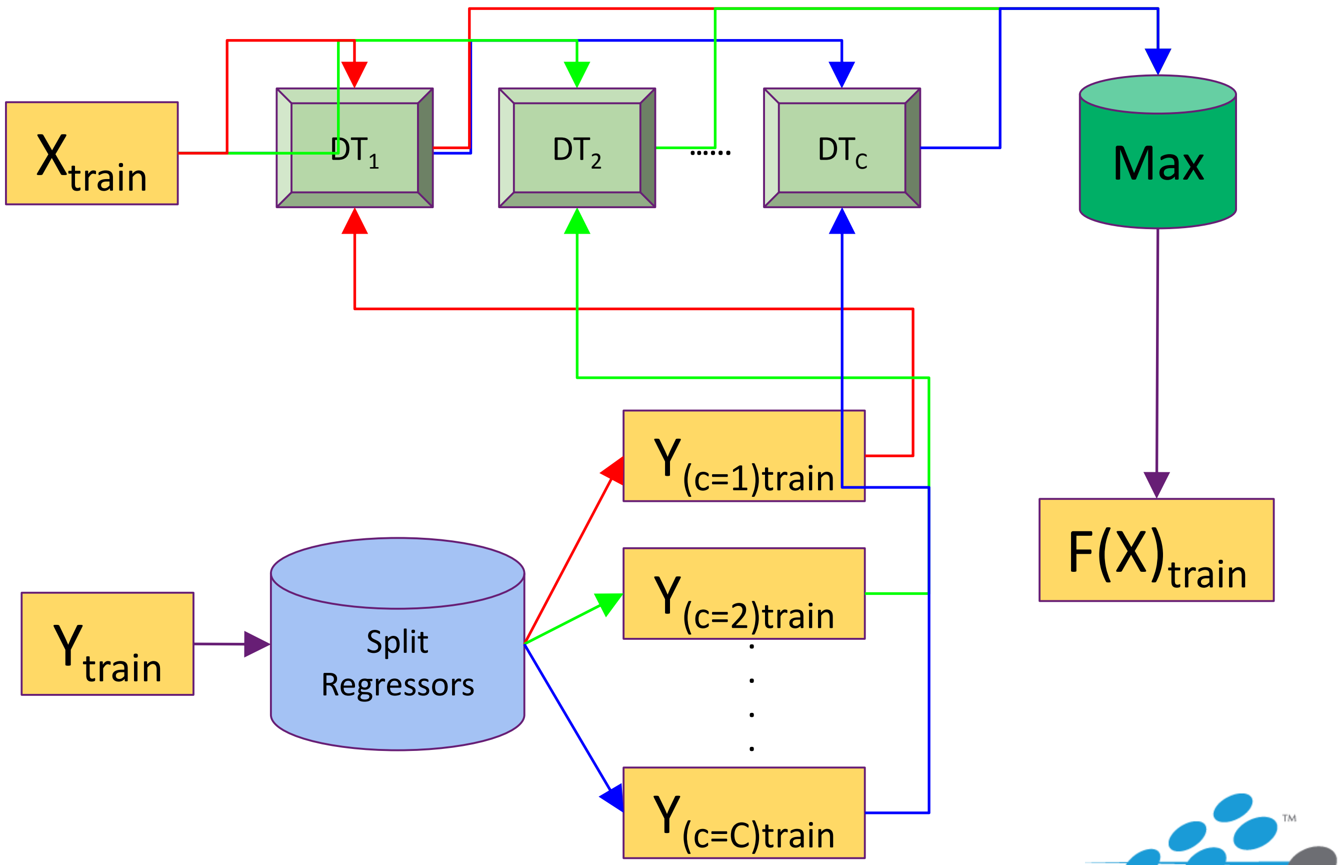
$$F(x) = \sum_{i=1}^{M} \gamma_i h_i(x)$$

Classification vs Regression

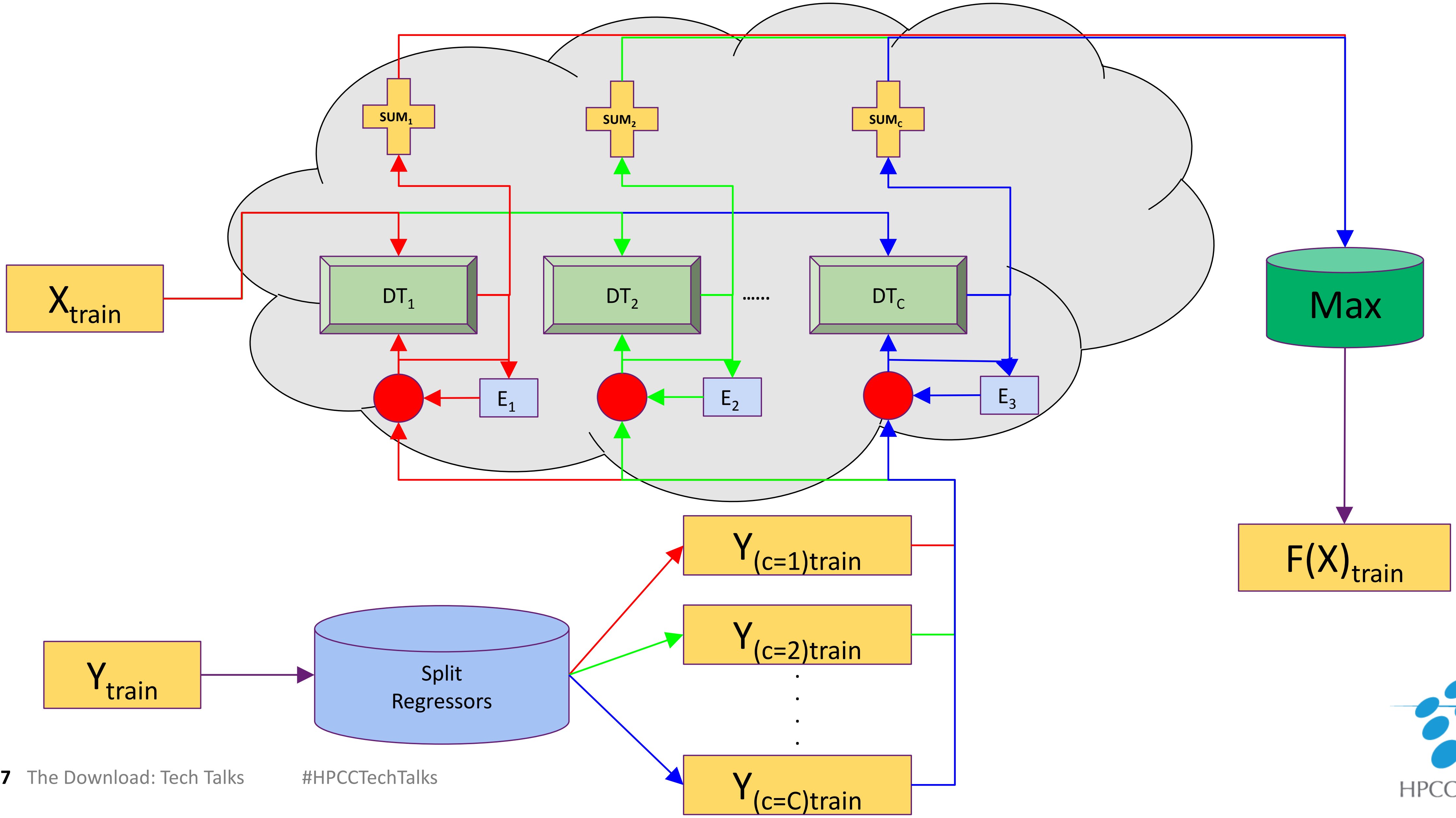
Regression



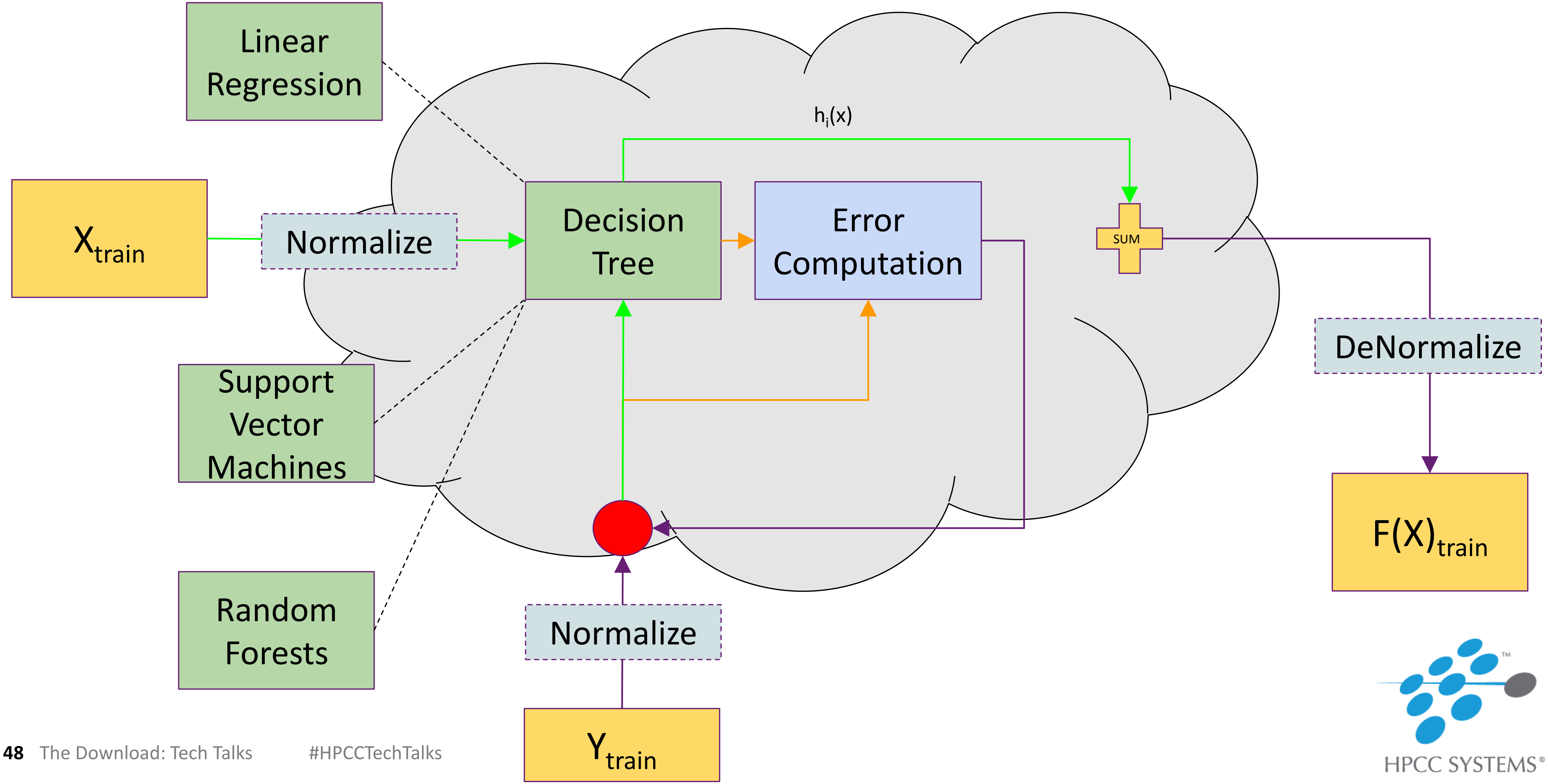
Classification



Classification - Extending To Gradient Boosting



Beyond an Algorithm



Regression: Compared to Native Approaches

Housing

Regressor	RMSE
Lin Reg	0.68
Dec Tree	0.74
GB(Lin Reg)	0.75
GB(Dec Tree)	**0.84**

Servo

Regressor	RMSE
Lin Reg	0.73
Dec Tree	0.52
GB(Lin Reg)	**0.77**
GB(Dec Tree)	0.73

Classification: Compared to Native Approaches

Yeast

Classifier	Prec	Rec	FA
Lin Reg	0.71	0.66	0.35
Dec Tree	0.62	0.67	0.36
GB(Lin Reg)	**0.73**	**0.71**	**0.30**
GB(Dec Tree)	0.65	0.67	0.33

Vehicle

Classifier	Prec	Rec	FA
Lin Reg	0.64	0.71	0.45
Dec Tree	0.81	0.78	0.21
GB(Lin Reg)	0.67	0.71	0.43
GB(Dec Tree)	**0.84**	**0.79**	**0.20**

Pros vs Cons:

Pros:

- Super-charge Weak Learner
- Works with less RAM
- Hardly any hyper-parameters(except for the Weak Learner)

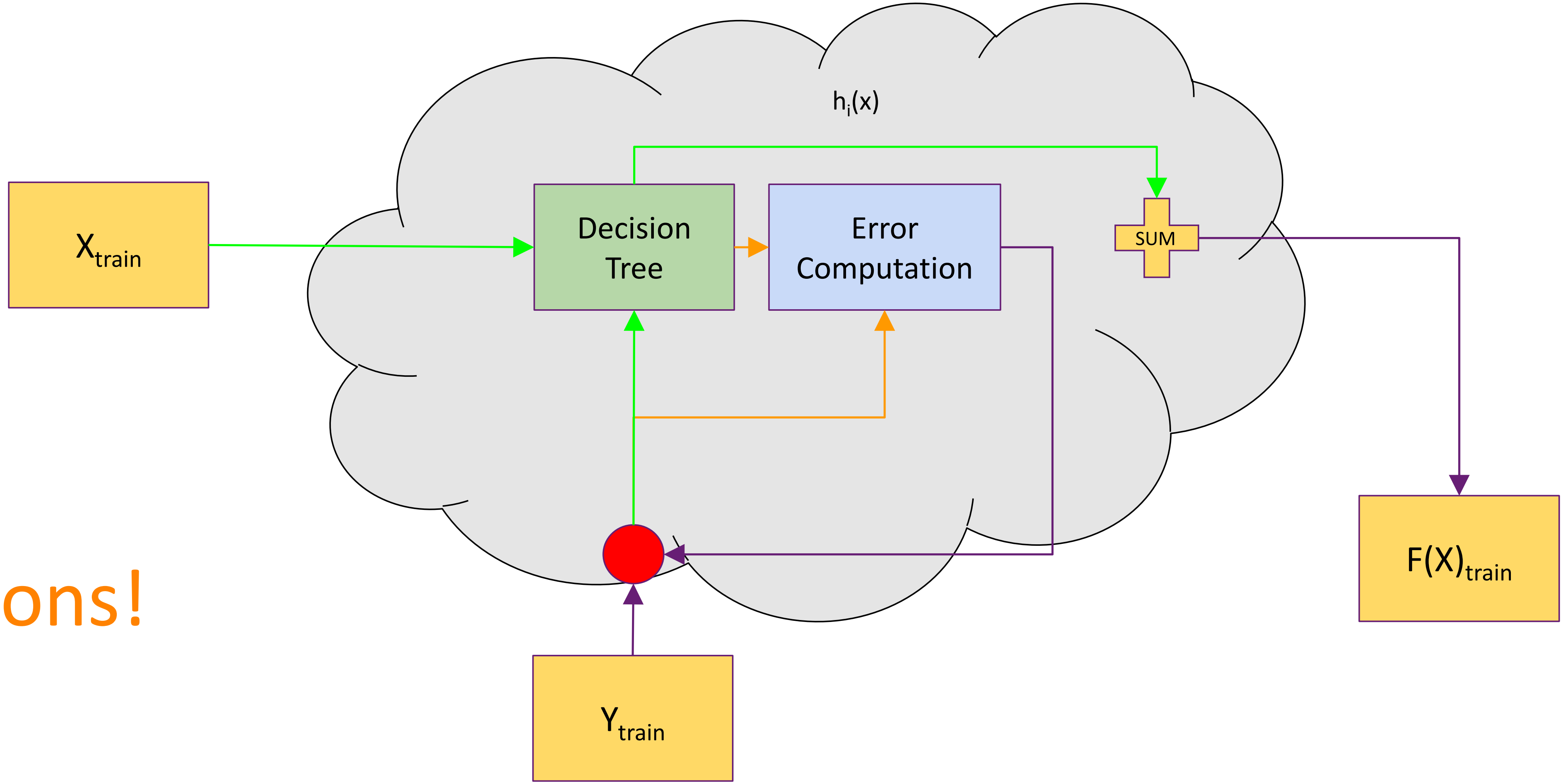
Cons:

- Cannot be parallelized efficiently.
- Runtime
  - Fixed to lesser extent by early termination

Work in Progress

- Classification can be parallelized.
- Incorporate Standardization.
- Make a bundle.
- Suggestions ….

Conclusion



Questions!

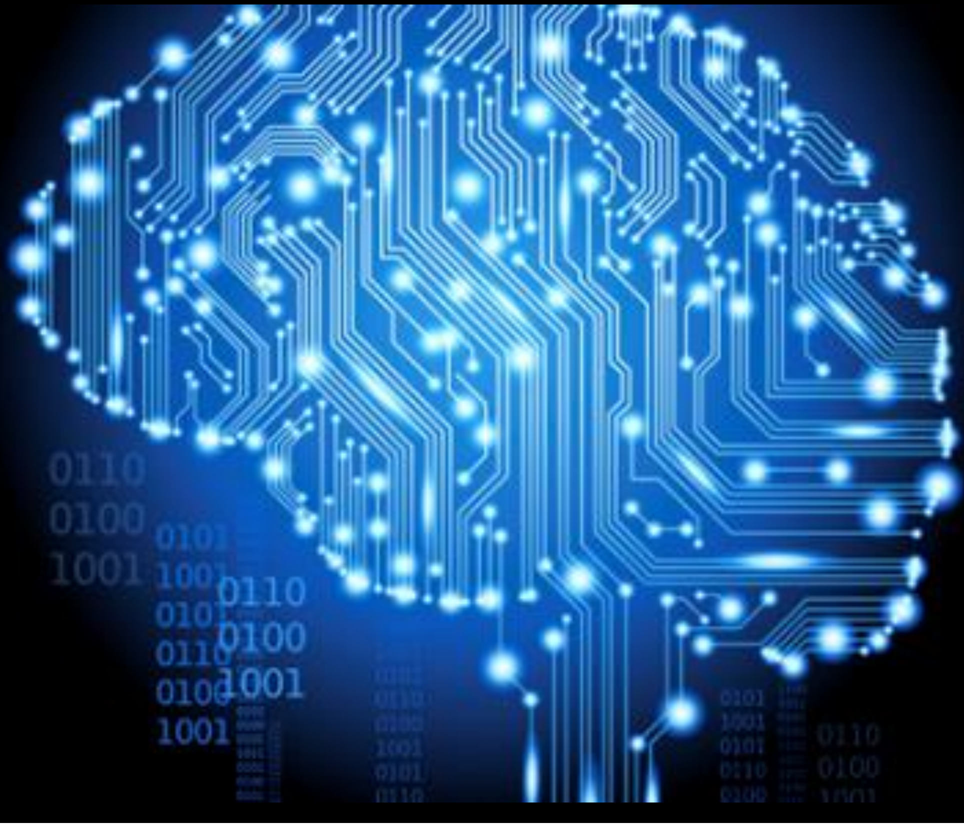
Questions?

George Mathew

PhD Student, Computer Science,
North Carolina State University
george2@ncsu.edu
bigfatnoob.us

THE DOWNLOAD

TECH TALKS BY HPCC SYSTEMS



HPCC SYSTEMS®

Spark-HPCC: HPCC Systems with Spark

Vivek Nair
PhD Student
Computer Science
North Carolina State University

NC STATE UNIVERSITY

Problem Statement

- Objective: Interoperability between HPCC Systems and Spark
  - ***Spark->HPCC***: Run Spark program (from Spark Shell) using data from HPCC Systems
  - ***HPCC->Spark***: Call Spark program (as sub-routine) from within ECL program (using ECL IDE)

- Side-Effects:
  - Can be used with **ANY** application by treating HPCC System's thor files as a local file
  - Can be used by analyst for quick exploration of data.

- Technologies used:
  - Python FUSE - Filesystem in User space HPCCFuseJ
  - Apache LIVY - enables interaction with a Spark cluster over a REST interface

HPCC SYSTEMS®

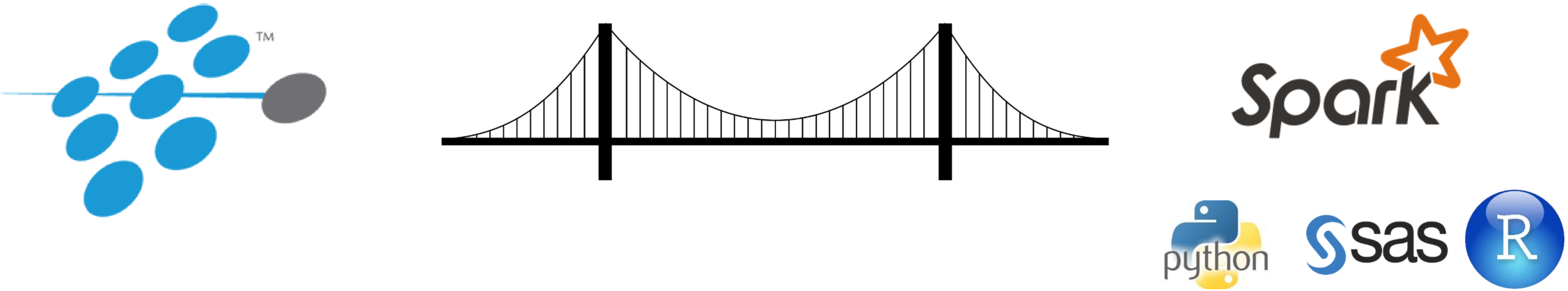
Agenda

- Motivation
- Introduction
- Possible Solutions
- Spark-HPCC: FUSE-based Solution
  - Spark->HPCC
  - HPCC->Spark
- Future Work

Agenda

- Motivation
- Introduction
- Possible Solutions
- Spark-HPCC: FUSE-based Solution
    - Spark->HPCC
    - HPCC->Spark
- Demonstration
- Future Work

Context



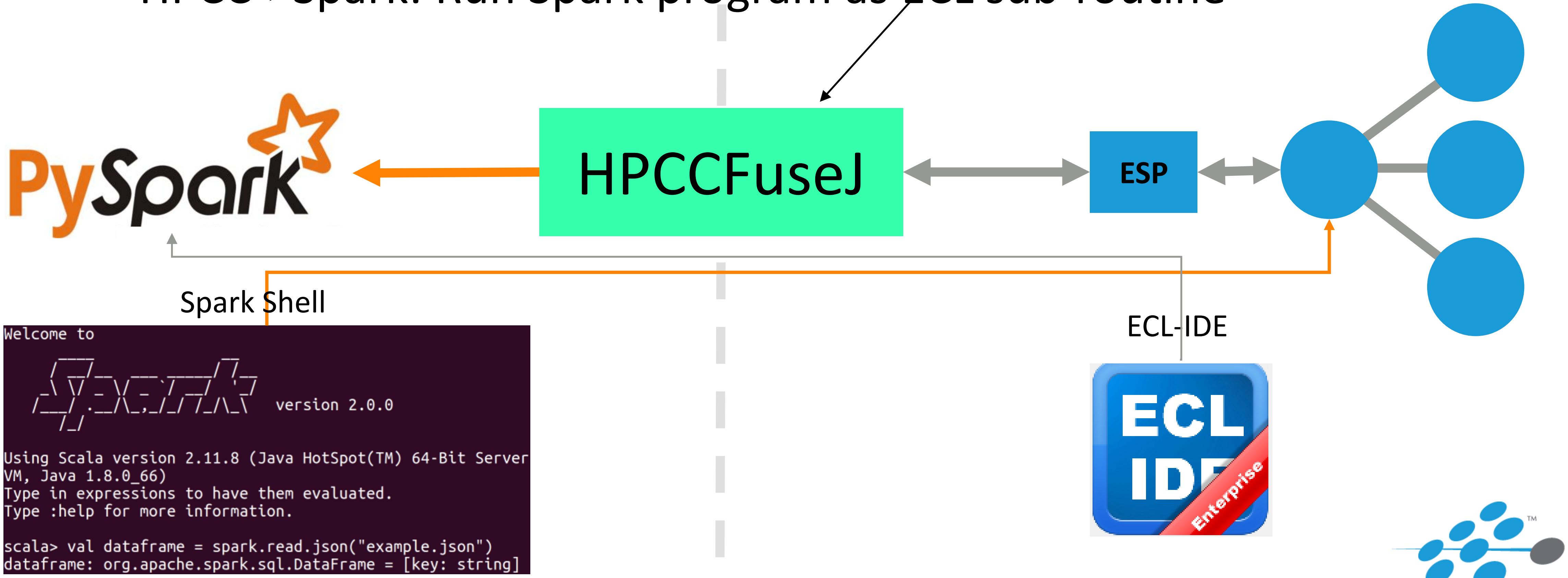
Agenda

- Motivation
- Spark-HPCC: Introduction
- Design
  - Spark->HPCC
  - HPCC->Spark
- Demonstration
- Future Work

Spark-HPCC

FUSE plugin which can mount HPCC Systems clusters as a local drive

- Spark->HPCC: Run Spark program using data stored in HPCC Systems
- HPCC->Spark: Run Spark program as ECL sub-routine

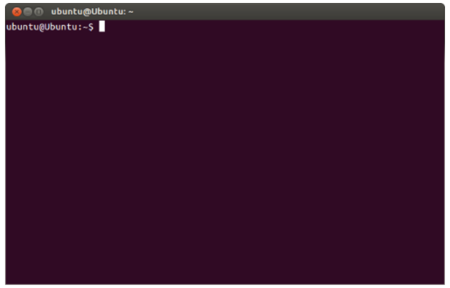


```
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /___/ .__/\_,_/_/ /_/\_\   version 2.0.0
      /_/

Using Scala version 2.11.8 (Java HotSpot(TM) 64-Bit Server
VM, Java 1.8.0_66)
Type in expressions to have them evaluated.
Type :help for more information.

scala> val dataframe = spark.read.json("example.json")
dataframe: org.apache.spark.sql.DataFrame = [key: string]
```

Spark-HPCC: Side Effect

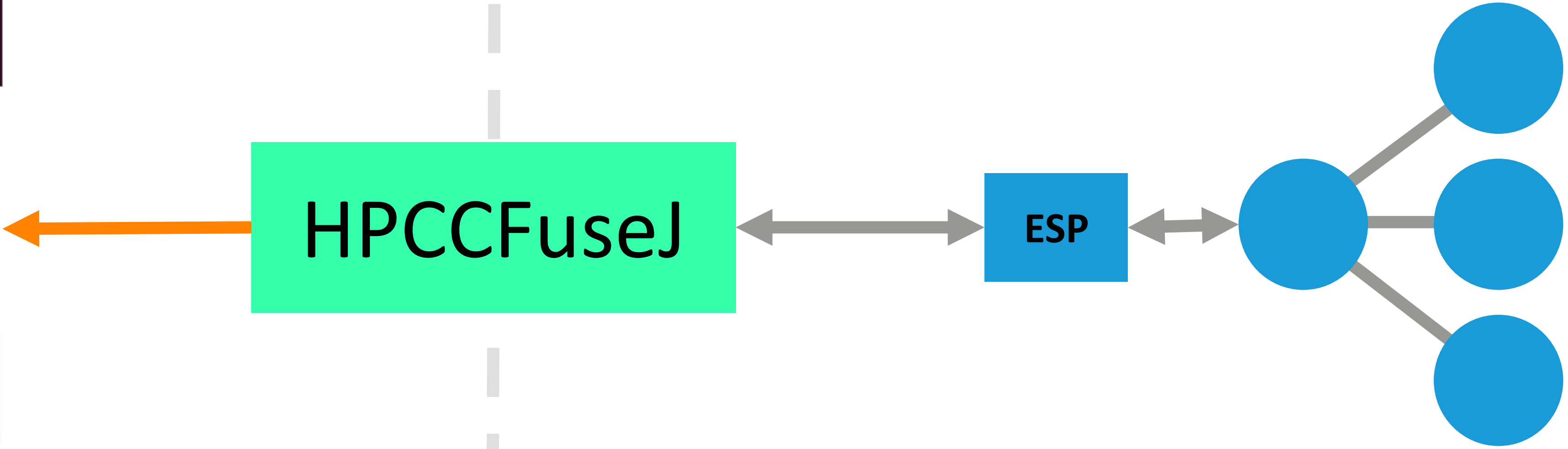


```
ubuntu@Ubuntu: ~
ubuntu@Ubuntu:~$ 
```

LibreOffice
The Document Foundation

MELD

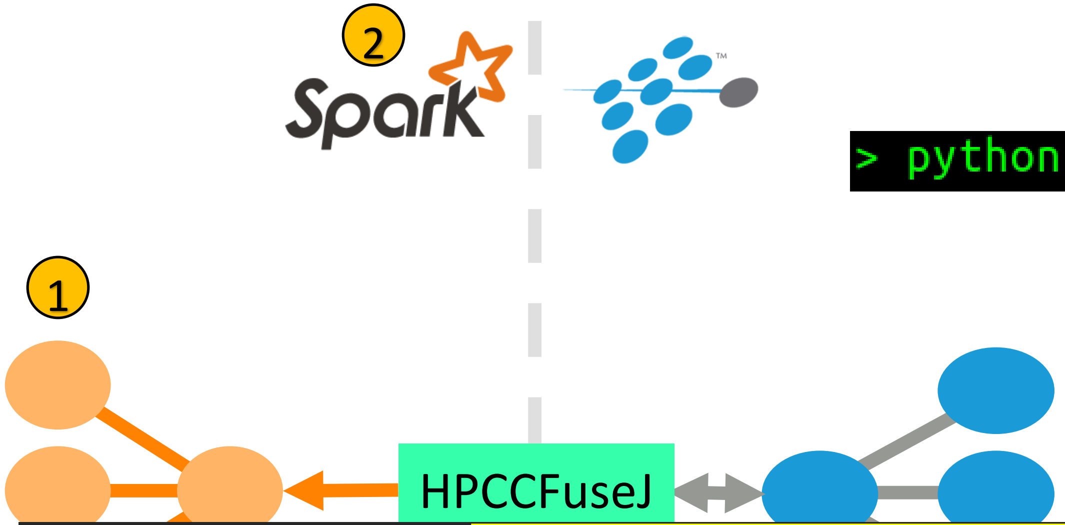
python



Agenda

- Motivation
- Spark-HPCC: Introduction
- Design
  - Spark->HPCC
  - HPCC->Spark
- Demonstration
- Future Work

Spark->HPCC: Run Spark using data from HPCC Systems



1. Mount HPCCFuseJ on the Spark master

```
> python passthrough_hpcc.py 10.239.227.6 8010 ~/GIT/mount_pnt2/
```

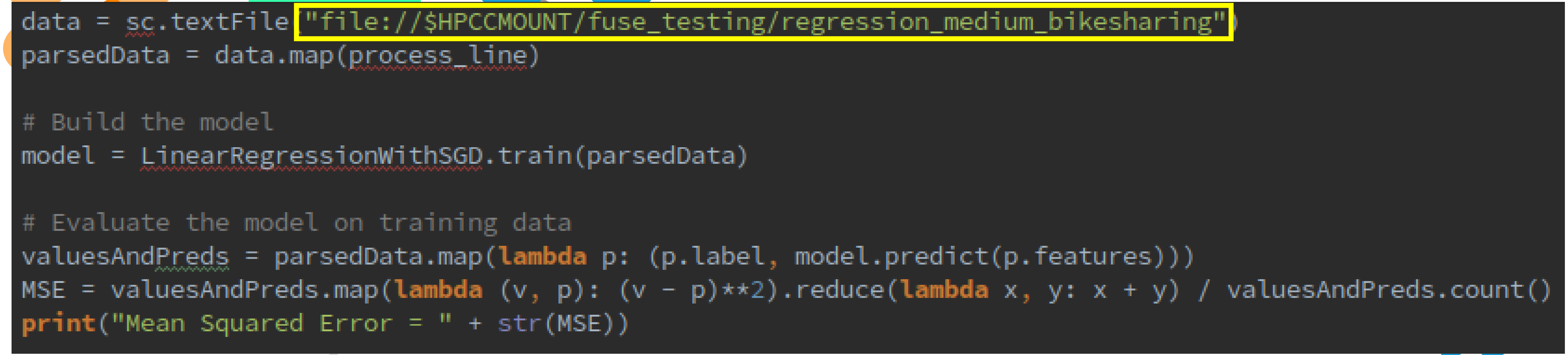
HPCC Cluster IP

Port

Local mount point

2. Run pySpark program

   - Treat HPCC Systems files as local files



```
data = sc.textFile("file://$HPCCMOUNT/fuse_testing/regression_medium_bikesharing")
parsedData = data.map(process_line)

# Build the model
model = LinearRegressionWithSGD.train(parsedData)

# Evaluate the model on training data
valuesAndPreds = parsedData.map(lambda p: (p.label, model.predict(p.features)))
MSE = valuesAndPreds.map(lambda (v, p): (v - p)**2).reduce(lambda x, y: x + y) / valuesAndPreds.count()
print("Mean Squared Error = " + str(MSE))
```

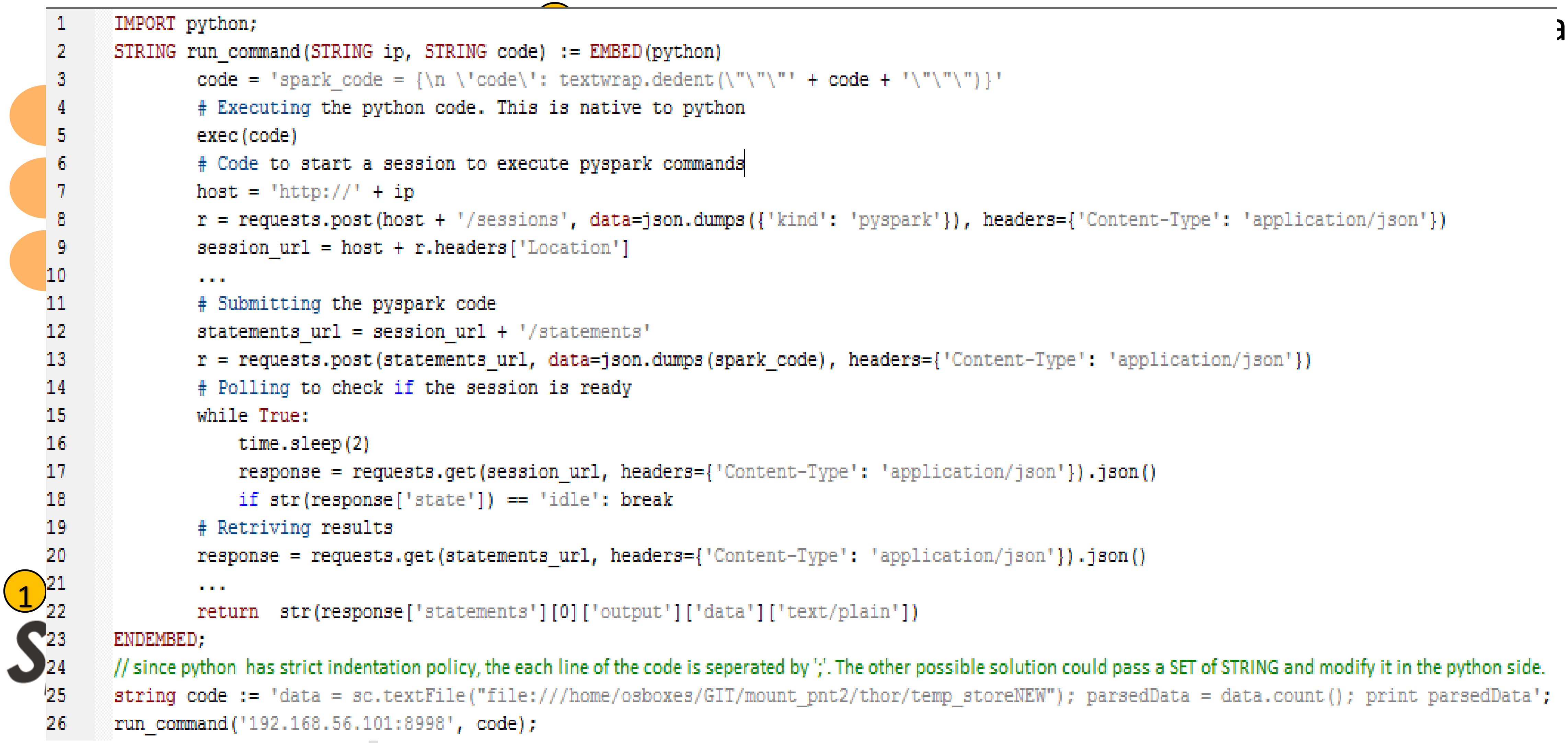
Agenda

- Motivation
- Spark-HPCC: Introduction
- Design
  - Spark->HPCC
  - HPCC->Spark
- Demonstration
- Future Work

HPCC->Spark: Run Spark from ECLIDE

aster

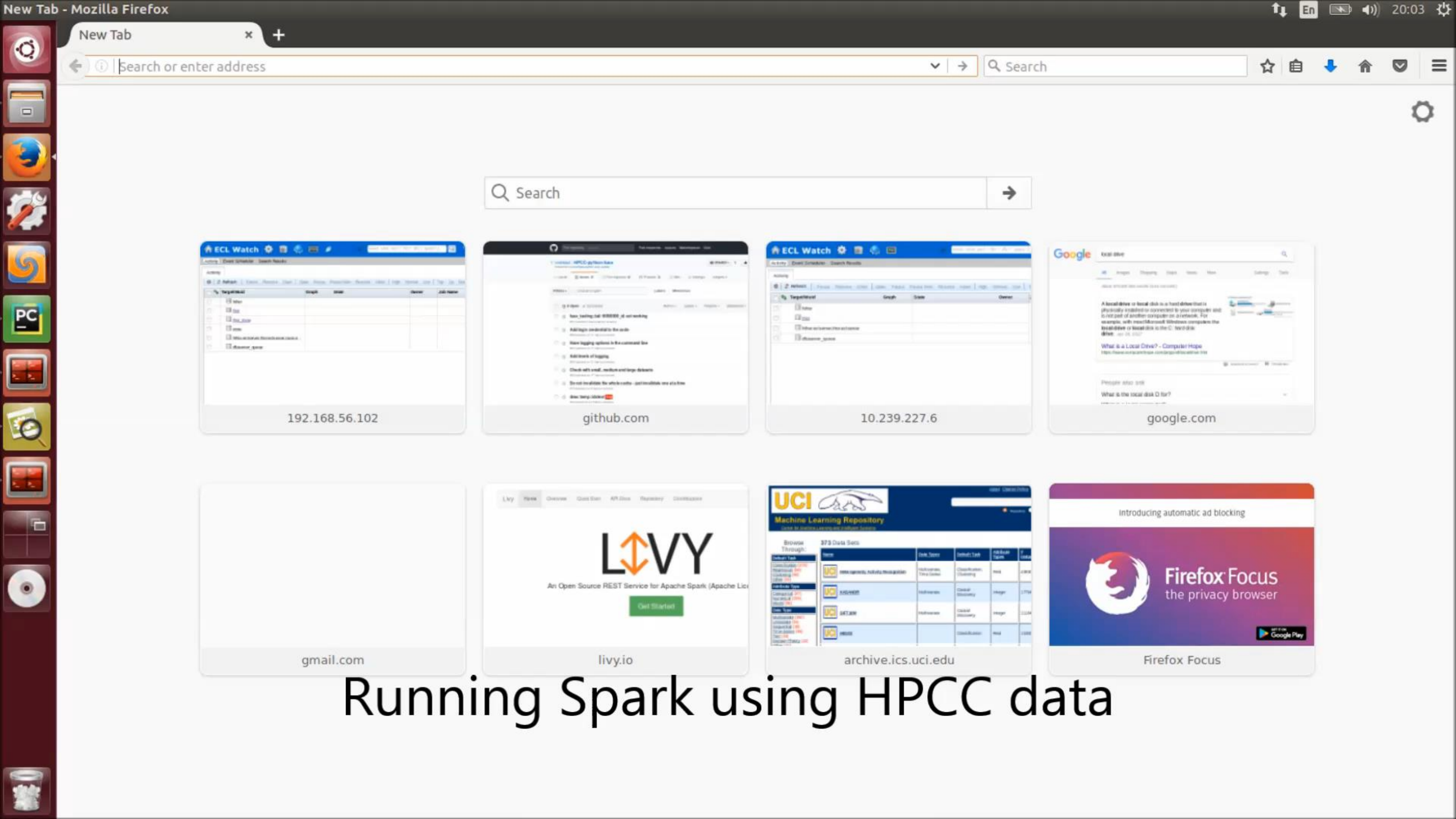
1



```
IMPORT python;
STRING run_command(STRING ip, STRING code) := EMBED(python)
        code = 'spark_code = {\n \'code\': textwrap.dedent(\"\"\"' + code + '\"\"\")}'
        # Executing the python code. This is native to python
        exec(code)
        # Code to start a session to execute pyspark commands
        host = 'http://' + ip
        r = requests.post(host + '/sessions', data=json.dumps({'kind': 'pyspark'}), headers={'Content-Type': 'application/json'})
        session_url = host + r.headers['Location']
        ...
        # Submitting the pyspark code
        statements_url = session_url + '/statements'
        r = requests.post(statements_url, data=json.dumps(spark_code), headers={'Content-Type': 'application/json'})
        # Polling to check if the session is ready
        while True:
            time.sleep(2)
            response = requests.get(session_url, headers={'Content-Type': 'application/json'}).json()
            if str(response['state']) == 'idle': break
        # Retriving results
        response = requests.get(statements_url, headers={'Content-Type': 'application/json'}).json()
        ...
        return  str(response['statements'][0]['output']['data']['text/plain'])
ENDEMBED;
// since python  has strict indentation policy, the each line of the code is seperated by ';'. The other possible solution could pass a SET of STRING and modify it in the python side.
string code := 'data = sc.textFile("file:///home/osboxes/GIT/mount_pnt2/thor/temp_storeNEW"); parsedData = data.count(); print parsedData';
run_command('192.168.56.101:8998', code);
```

Agenda

- Motivation
- Spark-HPCC: Introduction
- Design
  - Spark->HPCC
  - HPCC->Spark
- Demonstration
- Future Work



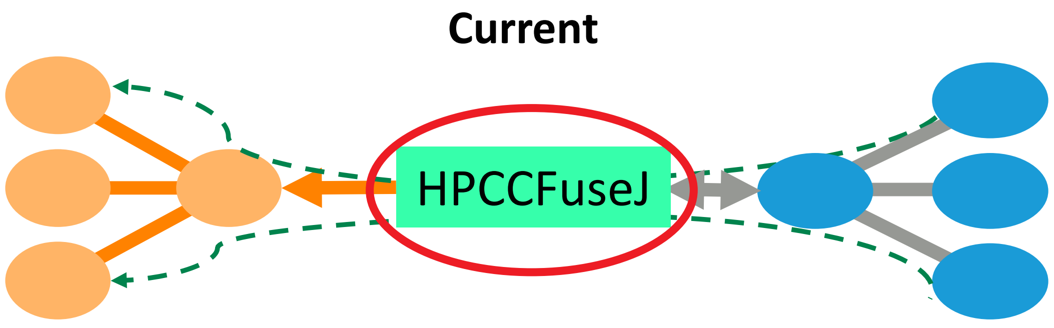
Running Spark using HPCC data

Agenda

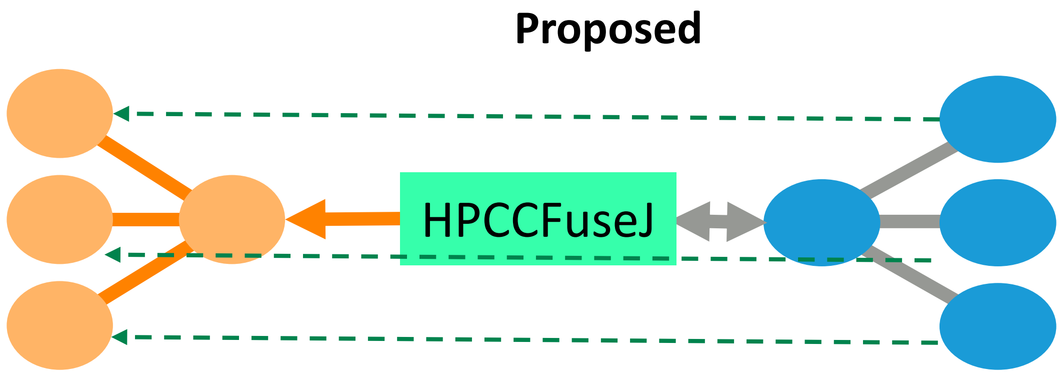
- Motivation
- Spark-HPCC: Introduction
- Design
  - Spark->HPCC
  - HPCC->Spark
- Demonstration
- Future Work

Future Work

- Remove bottleneck



Performance Report: tiny.cc/hpccfusej_perf



Expected: April 2018

- Streaming Data
  - Data needs to be persisted (saved) before executing Spark
  - Can data be streamed from HPCC Systems to Spark rather than persisting?

Expected: April 2018

Questions?

Vivek Nair

PhD Student, Computer Science,
North Carolina State University
vnair2@ncsu.edu
vivekaxl.com

More about the HPCC Systems Intern Program…

- Blogs about the program: https://hpccsystems.com/blog

- Available projects: https://wiki.hpccsystems.com/x/yIBc

- Previously complete projects: https://wiki.hpccsystems.com/x/g4BR

- Student wiki: https://wiki.hpccsystems.com/x/HwBm

- HPCC Systems Technical Presentation Competition 2016: https://wiki.hpccsystems.com/x/FQCv

Submit a talk for an upcoming episode!

- Have a new success story to share?
- Want to pitch a new use case?
- Have a new HPCC Systems application you want to demo?
- Want to share some helpful ECL tips and sample code?
- Have a new suggestion for the roadmap?
- Be a featured speaker for an upcoming episode! Email your idea to Techtalks@hpccsystems.com

Stay tuned for details on our next Tech Talk!

Visit The Download Tech Talks wiki for more information:
https://wiki.hpccsystems.com/display/hpcc/HPCC+Systems+Tech+Talks

Thank You!

HPCC SYSTEMS®

RELX Group

A copy of this presentation will be made available soon on our blog:
hpccsystems.com/blog