How are organizations (like yours) approaching Big Data?

Please visit:

http://lexisnexis.com/risk/SolutionSurvey

LexisNexis®

HPCC Systems

An Open-Source, Enterprise-Grade, Research-Ready, Data Intensive Computing Platform

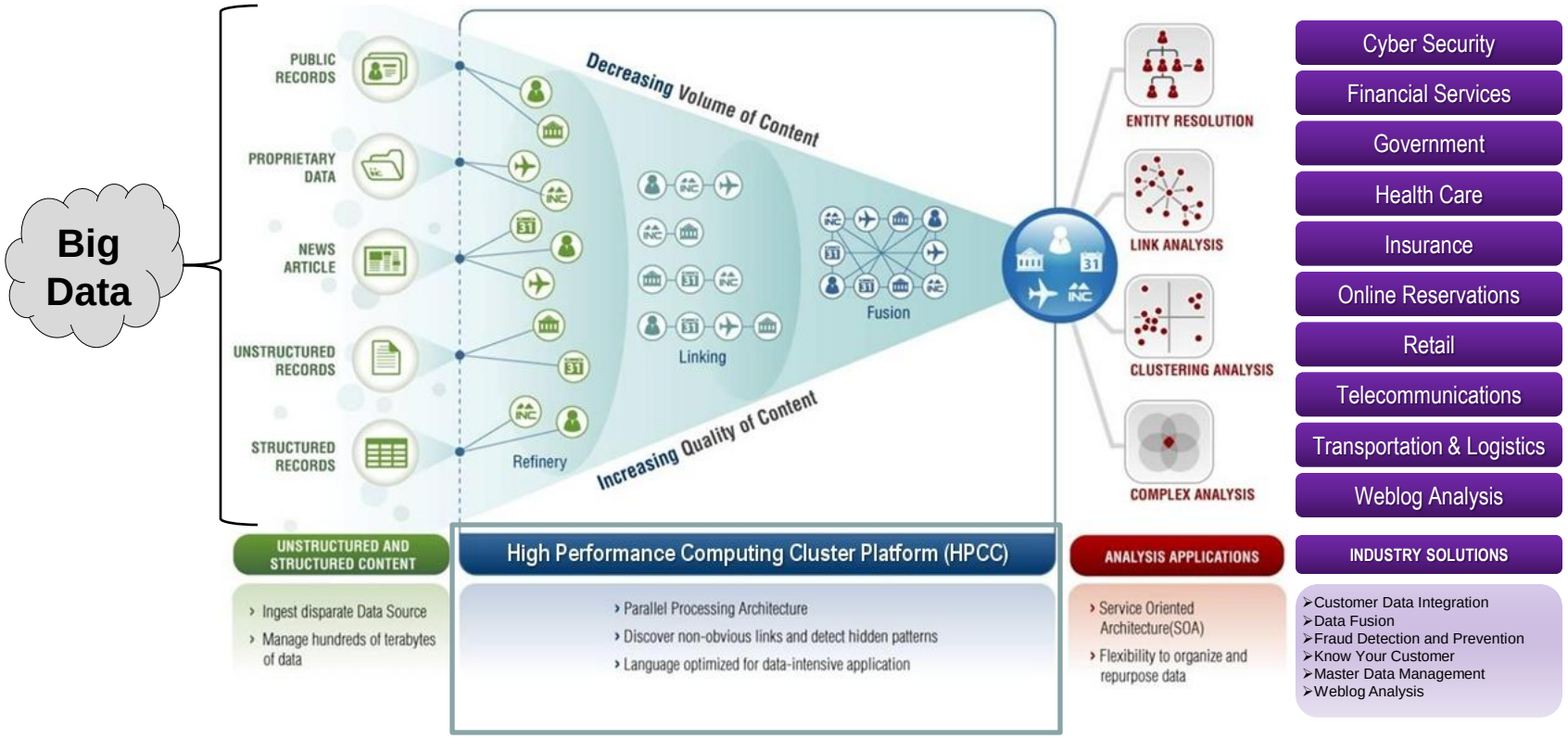
Charles Kaminski, Sr. Architect
Charles.Kaminski@LexisNexis.com
http://HPCCSystems.com

LexisNexis®

Who does our technology serve?

- Approximately 70 percent of local governments
- Approximately 80 percent of federal agencies
- Approximately 90 percent of Fortune 500 corporations
- Thousands of smaller businesses
- Academia
- Research Centers

Each of you uses this platform



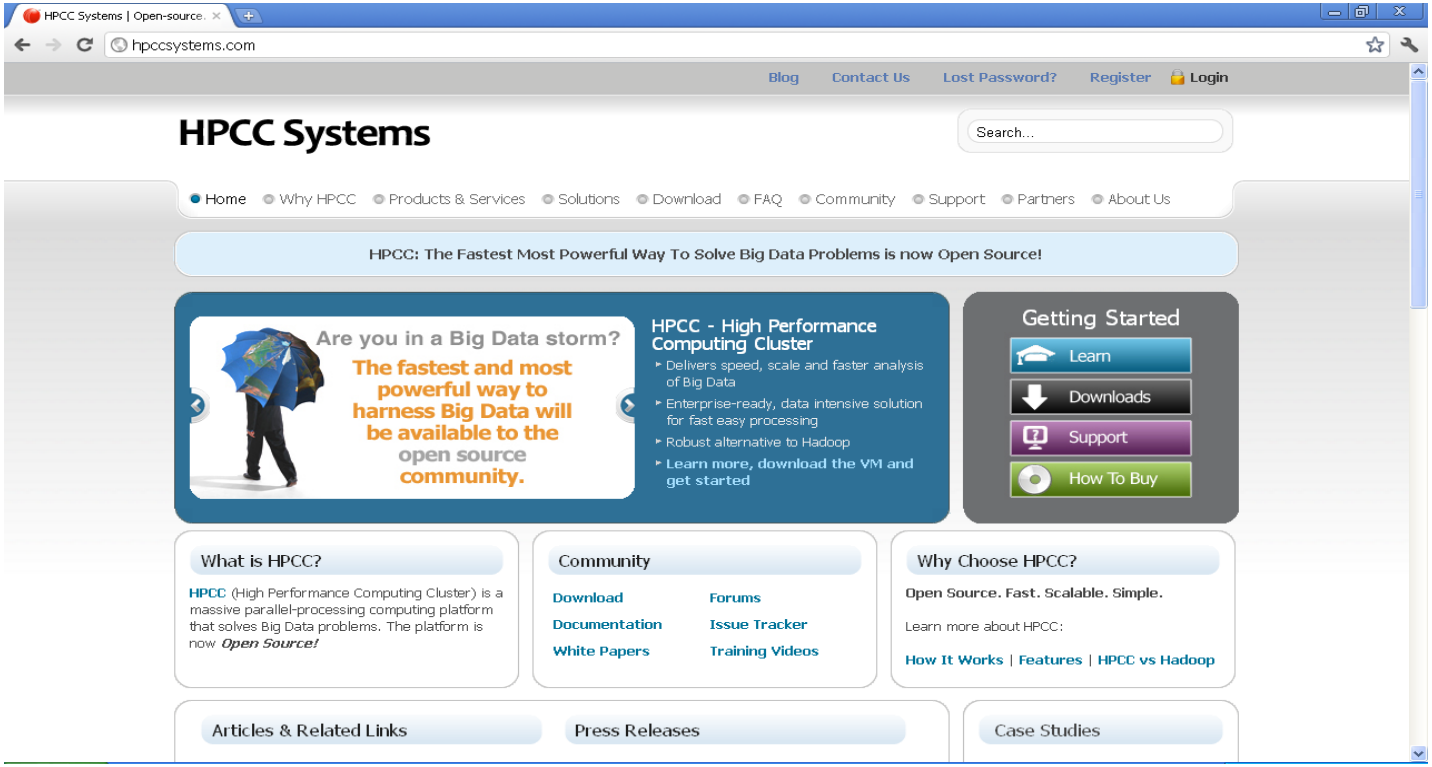
Types of data

- Structured
- Unstructured
- Semi Structured
- Free Text
- Big Endian
- Little Endian
- CSV
- XML
- CSV
- Fixed Width
- …
- and more

Now this platform is available to you for free

http://hpccsystems.com
http://aws.hpccsystems.com

Write these down



Free Training Available to Academia

Available to faculty and researchers based on availability.

Students included if faculty participate.

Contact:
Charles.Kaminski@LexisNexis.com

Our Technical People Continue to Improve the Platform

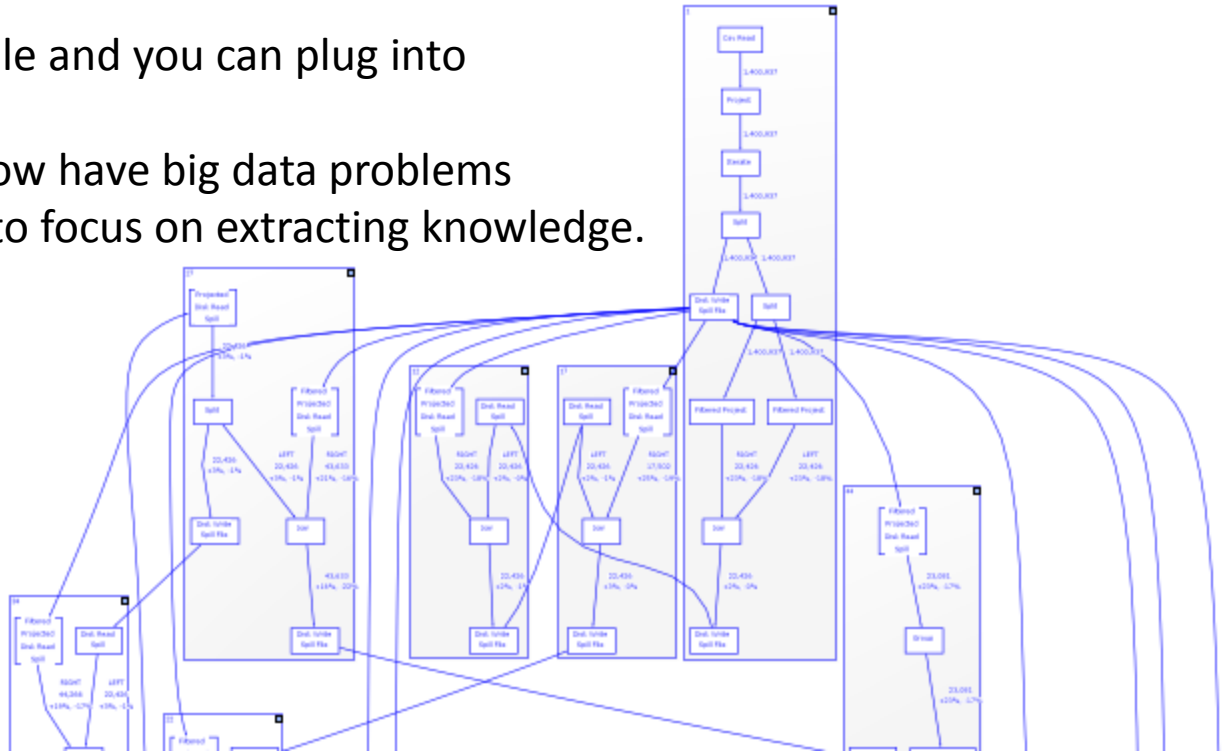
Data Intensive Computing
- HPC
- Machine Learning
- Programming Language Development
- NLP

Big Data Analytics
- Professors
- Researchers
- Students
- Seasoned Engineers
- Predictive Experts
- Modeling Experts



How are we different?

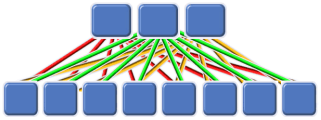
- We can solve problems difficult or impossible to solve with map reduce
- Engineering teams are smaller and faster
- Write fewer lines of code
- Technology stack is smaller and has fewer moving parts
- The language is extendable and you can plug into existing packages.
  - Many researchers now have big data problems
- The platform allows you to focus on extracting knowledge.



The Major Parts

Thor

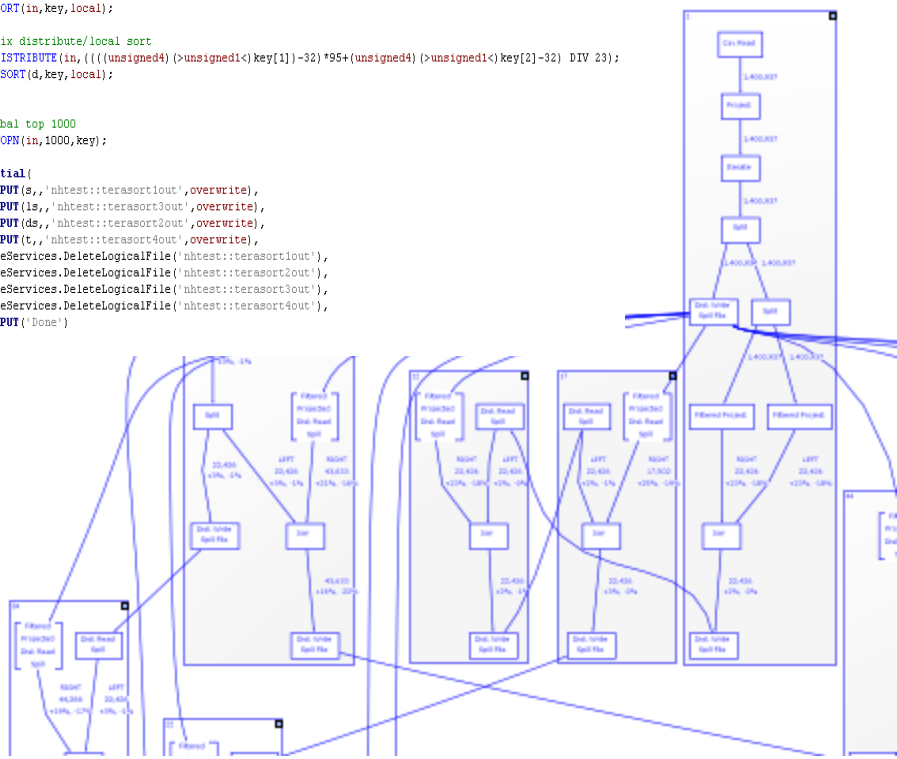
Roxie



ECL

1

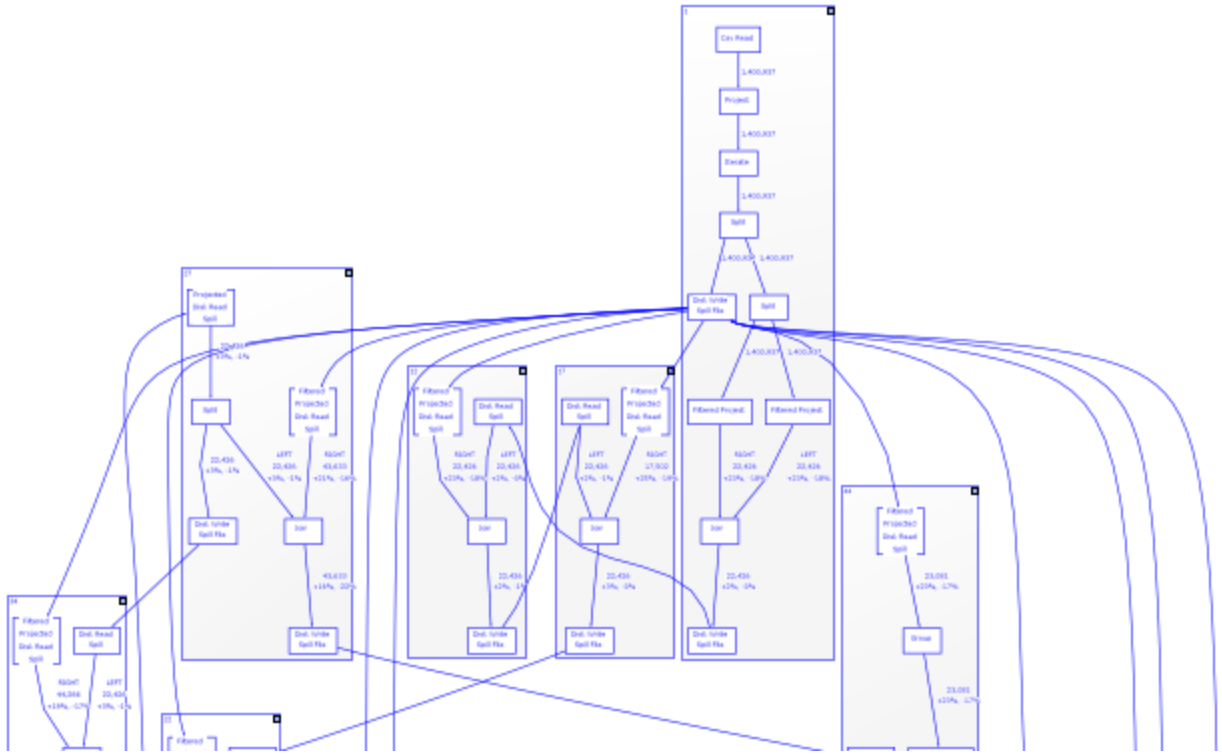
```
// TeraByte Sort Benchmark
import lib_fileservices;

rec := record
     string10  key;
     string10  seq;
     string80  fill;
       end;

in := DATASET('nhtest::terasort1',rec,FLAT);

// global sort
s:= SORT(in,key);


// local sort
ls:= SORT(in,key,local);

// radix distribute/local sort
d := DISTRIBUTE(in,((((unsigned4)(>unsigned1<)key[1])-32)*95+(unsigned4)(>unsigned1<)key[2]-32) DIV 23);
ds := SORT(d,key,local);


// global top 1000
t := TOPN(in,1000,key);

sequential(
   OUTPUT(s,,'nhtest::terasort1out',overwrite),
   OUTPUT(ls,,'nhtest::terasort3out',overwrite),
   OUTPUT(ds,,'nhtest::terasort2out',overwrite),
   OUTPUT(t,,'nhtest::terasort4out',overwrite),
   FileServices.DeleteLogicalFile('nhtest::terasort1out'),
   FileServices.DeleteLogicalFile('nhtest::terasort2out'),
   FileServices.DeleteLogicalFile('nhtest::terasort3out'),
   FileServices.DeleteLogicalFile('nhtest::terasort4out'),
   OUTPUT('Done')
);
```



How the platform makes it happen

- The right architectural decisions early on.
- Few moving parts.
- The Data Scientist have the power and control over the whole data lifecycle.
- Big data doesn't mean big teams.

Bringing up a cluster with one click.



Questions?

Check it out at

http://hpccsystems.com

Or chat with me directly

Charles Kaminski

Senior Architect

Academic Development Lead, HPCC Systems

Charles.Kaminski@lexisnexis.com

402-619-9413