The Constellation Project

Andrew W. Nash

14 November 2016

THE UNIVERSITY OF TENNESSEE KNOXVILLE

The Constellation Project: Representing a High Performance File System as a Graph for Analysis

- The Titan supercomputer utilizes high performance file systems that change significantly as scientists run simulation algorithms
- The metadata are a rich source for data analysis to extrapolate similarities between the various entities with modern graph algorithms
- An efficient graph library must be utilized in order to perform analysis
- This project in lieu of thesis (PILOT) examines the Constellation graph library and implements graph analytics algorithms, including PageRank and SimRank
- Results from the analysis are examined to determine if importance in the graph correlates to power users of the system in a given period of time

Introduction: Titan

- Titan Cray XK7 supercomputer
  - Managed by the Oak Ridge Leadership Computing Facility
  - Peak performance of 27 petaFLOPS
  - Third fastest in the world on the *Top500* June 2016 benchmark list

Introduction: Titan

- Atlas1 high performance file system
  - Contains over 1,000 object storage targets
  - Total usable capacity of 14 petabytes
- Some subject areas using Titan
  - Chemical sciences
  - Climate change science
  - Combustion science
  - Molecular sciences
  - Multiple topics in physics

Introduction: Data

- Two snapshots of Atlas1 were utilized, one from 20 July 2015 (J20) and the other from 21 July 2015 (J21)

Entity Type	J20 Quantity	J20 File Size	J21 Quantity	J21 File Size
App	0	303 KB	13	472 KB
File	187,325,446	39.09 GB	187,754,436	39.17 GB
Group	11,505	728 KB	11,510	729 KB
Job	600	48 KB	859	67 KB
User	12,987	658 KB	12,990	658 KB
TOTAL	187,350,538	39.09 GB	187,779,808	39.18 GB

Introduction: Data

- Example App line in the data set
  - Host: `titan;`
  - App ID: `100;`
  - User ID: `0;`
  - Start time: `2015-07-20 00:00:01;`
  - End time: `2015-07-20 00:01:00;`
  - Number of processing elements: `32;`
  - Exit code: `0;`
  - Command: `./aprun -n 32 -N 1 ./io -f script`

Introduction: Data

- Example File line in the data set
  - Access time: `1434850000|`
  - Modify time: `1434850000|`
  - Change time: `1434850000|`
  - Owner user ID: `0|`
  - Group user ID: `0|`
  - Access setting: `40700|`
  - Size in bytes: `4096|`
  - Inode: `148373000|`
  - Path: `/ROOT/sample_file`

Introduction: Data

- Example Group line in the data set
  - User ID: `52;`
  - Group ID: `10;`
  - User name: `nash;`
  - Group name: `users;`

Introduction: Data

- Example Job line in the data set
  - Host: `titan;`
  - User: `root;`
  - Job ID: `2111111;`
  - Job name: `job1;`
  - Project: `physics1;`
  - Start time: `1434850000;`
  - Stop time: `1434859225;`
  - Wall time: `02:00:00;`
  - Nodes: `1;`
  - Exit code: `0`

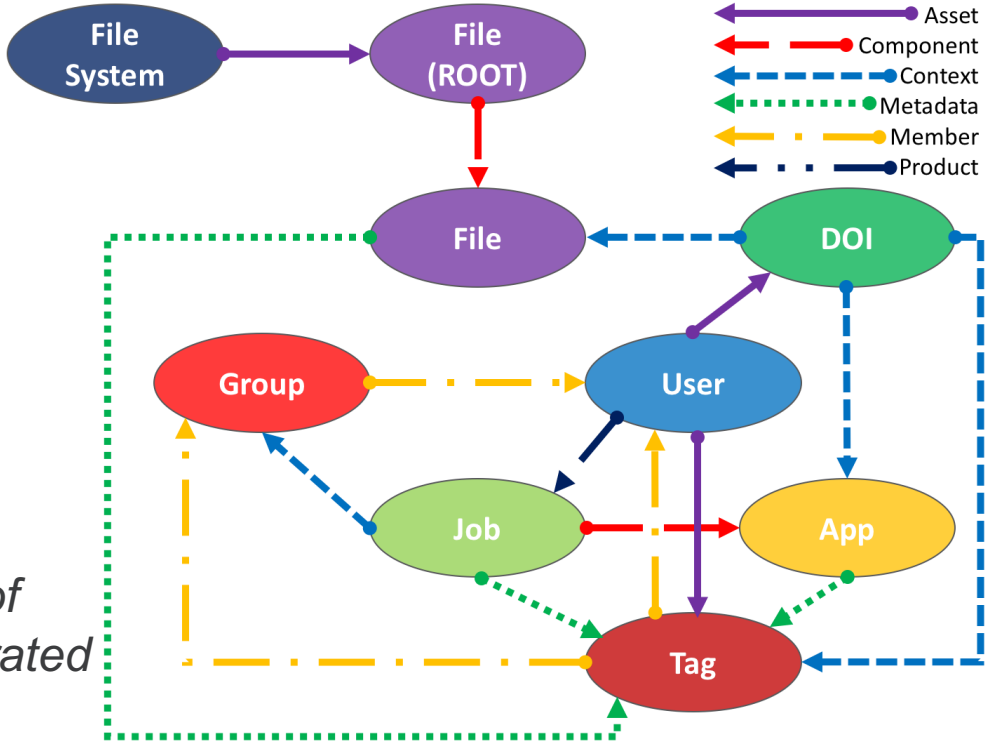
Introduction: Data

- Example User line in the data set
  - User ID: `52;`
  - User name: `nash;`
  - First: `Andrew;`
  - Middle: `W;`
  - Last: `Nash;`
  - Email address: `nashaw@ornl.gov`

Introduction: Constellation

- Constellation Data Service (CDS)
  - Created by the Technology Integration Group at ORNL
  - Extracts entities from the snapshot files and generates a graph that can be loaded into a high performance system's memory for analysis
  - Each entity in the snapshot is represented as a vertex, and the program automatically adds appropriate edges
  - There are different types of edges to add context to the relationship between two vertices in the CDS

Introduction: Constellation



Visual representation of the graph generated by the CDS

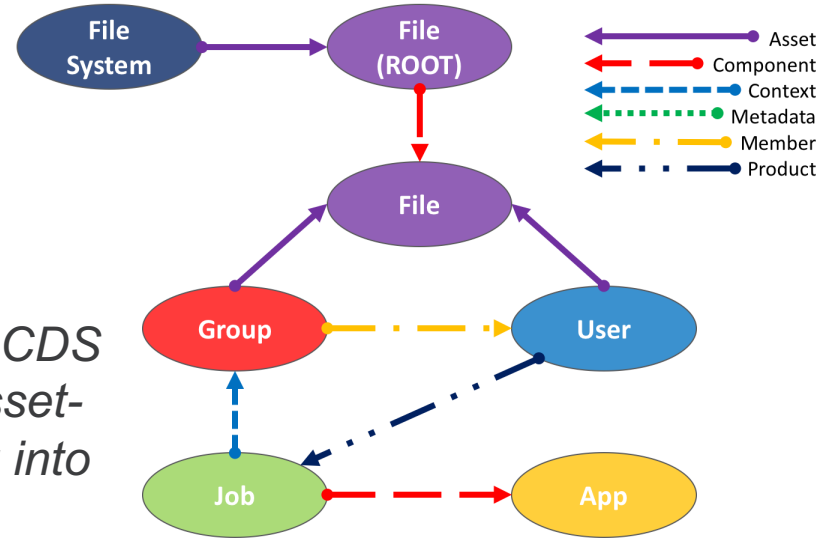
Implementation: CDSAnalytics

- Starts by loading a graph file generated by the CDS into memory
- In order to perform analysis, CDSAnalytics assigns a key to each vertex, known as a VKey, based on the attributes below

Vertex Type	First Letter	+	Example
App	a	System ID	a100
DOI	d	System Number	d9999
File	f	CDS Vertex Number	f10
File System	s	System Name	satlas1
Group	g	System ID	g100
Job	j	System ID	j100
Tag	t	System Name	tclimate
User	u	System ID	u100

Implementation: CDSAnalytics

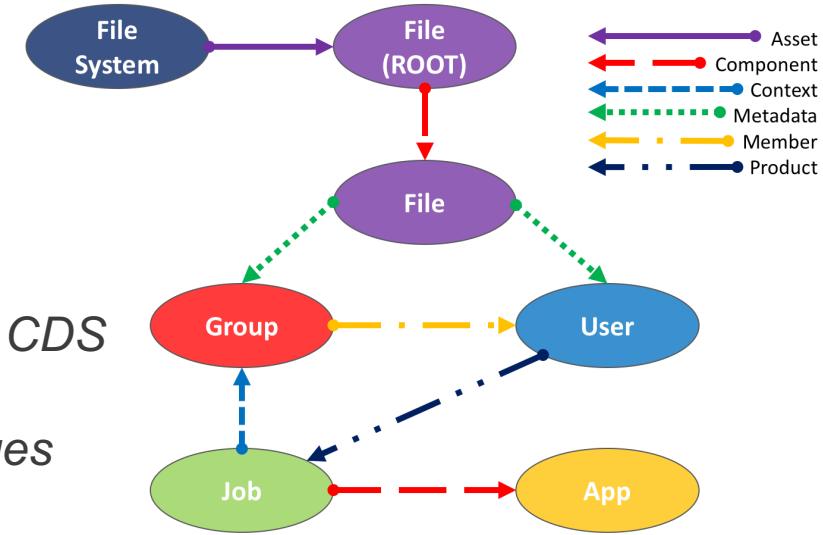
- *Edges* feature: Adds asset-type edges from file vertices to the respective user and group vertices that own the file



Modification of the CDS graph to include asset-type edges leading into files

Implementation: CDSAnalytics

- *Edges* feature: Also adds metadata-type edges from file vertices to the respective user and group vertices

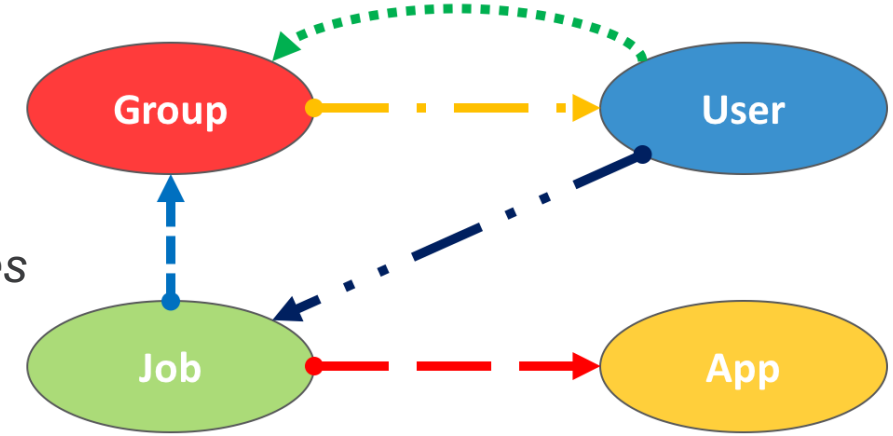


Modification of the CDS graph to include metadata-type edges leading from files

Implementation: CDSAnalytics

- *Modify* feature: For every file vertex in the graph, an edge is added from the user vertex that owns the file to the group vertex that owns the file, if such an edge does not already exist

Modification of the CDS graph that removes file vertices and replaces them with metadata-type edges



Implementation: CDSAnalytics

- *Export* feature: Exports the graph to a comma-separated value (CSV) file as a list of edges, where each line in the resulting file represents an edge
- *Index*: Generates an index that maps a vertex's VKey to a pointer that indicates the location of the vertex in memory to allow for an extremely fast query
- *PR*: Generates a file containing the matrix necessary to run the PageRank algorithm

Implementation: CDSAnalytics

- *Print*: Prints the first level of file vertices that are children of the root file
- *SR*: Generates a file containing the matrix necessary to run the SimRank algorithm
- *Store*: Stores the current state of the graph to a file so that it can be reloaded later
- *Total*: Shows total number of the vertices, broken down by each type

Implementation: CDSAnalytics

- *Query*: Queries a vertex by its VKey to retrieve all of its metadata stored in the graph, as shown below

```
Enter vertex key: u52
User Vertex:  u52
          ID: 52
    Username: nash
        Name: Nash, Andrew W.
       Email: nashaw@ornl.gov
    Edges In: 3
   Edges Out: 0
```

Implementation: DATAnalytics

- *PageRank*: Takes an adjacency matrix generated by CDSAnalytics to calculate the PageRank values for all vertices using the power method [Austin, 2006] with a user-specified damping factor and tolerance of 0.001
- Since the adjacency matrix is sparse, as shown below, DATAnalytics only stores the non-zero values in three vectors and performs parallel calculations using Open MP
- PageRank values are used to attempt to rank vertices by importance based on the number of edges entering a particular vertex

$$
\begin{array}{c} \\ System \\ Root \\ File \\ Group \\ User \\ Job \\ App \end{array}
\begin{array}{c}
\begin{array}{ccccccc} System & Root & File & Group & User & Job & App \end{array} \\
\left(\begin{array}{ccccccc}
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1/2 & 0 & 0 & 1/2 & 0 \\
0 & 0 & 1/2 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1/2 & 0
\end{array} \right)
\end{array}
$$

Implementation: DATAnalytics

- *SimRank*: Takes the smaller adjacency matrix, without the files, generated by CDSAnalytics to calculate the SimRank values for all vertices, using the method shown below [Antonellis et al., 2008]
- Since the adjacency matrix is sparse, DATAnalytics only stores the non-zero values into a mapped matrix
- Since matrix multiplication is computationally intensive, DATAnalytics ignores all multiplications by zero and performs calculations in parallel using OpenMP
- SimRank values are used to attempt to compare vertices by similarity based on the connections between particular vertices

Algorithm SimRank

Require: Adjacency matrix P, Decay factor $decay$, Number of iterations k

$S \leftarrow I$

for $i = 1 : k$ **do**

$T \leftarrow decay * P^T * S * P$ // parallel operations

$S \leftarrow T + I -$ Diag(diag(T)) // parallel operations

end for

Results

- The CDSAnalytics and DATAnalytics programs were run on multiple high-performance systems that each contained approximately 96 GB of memory
- Each system utilized 8 cores to allow for a peak of 800% CPU usage while performing parallel matrix operations
- All processes completed within approximately four hours of wall time, indicating that these algorithms could be reasonably implemented into the CDS and periodically run as batch jobs to monitor graph analytics or power a search engine

Results: PageRank

Data	α	ID	Subject	In	Out	Notes
J20 A	0.15	G94	Biophysics	122	7967	Biophysics 2nd, 3rd
J20 A	0.15	U67	Comp. modeling	14	1	
J20 A	0.50	G94	Biophysics	122	7967	Biophysics 2nd, 4rd, 7th
J20 A	0.50	U05	Scientific comp.	16	41,324	
J20 A	0.85	G94	Biophysics	122	7967	Biophysics 2nd, 3rd
J20 A	0.85	U67	Comp. modeling	14	1	
J21 A	0.15	G73	Biophysics	119	11	Biophysics 2nd, 3rd, 9th
J21 A	0.15	U67	Comp. modeling	14	1	
J21 A	0.50	G73	Biophysics	119	11	Biophysics 2nd, 3rd, 10th
J21 A	0.50	U67	Comp. modeling	14	1	
J21 A	0.85	G73	Biophysics	119	11	Biophysics 2nd, 3rd, 10th
J21 A	0.85	U67	Comp. modeling	14	1	
J20 M	0.15	G70	Computer science	6	9	
J20 M	0.15	U11	Computer science	12,417,090	2	G70 leader
J20 M	0.50	G70	Computer science	6	9	
J20 M	0.50	U11	Computer science	12,417,090	2	G70 leader
J20 M	0.85	G70	Computer science	6	9	
J20 M	0.85	U52	Computer science	5	0	G70 member
J21 M	0.15	G80	Climate	1,098,650	28	Climate 10th
J21 M	0.15	U32	Computer science	3043	0	
J21 M	0.50	G80	Climate	1,098,650	28	Climate 7th, 10th
J21 M	0.50	U32	Computer science	3043	0	
J21 M	0.85	G96	Nuclear physics	256,657	2	G80 3rd
J21 M	0.85	U70	Accelerator physics	61,347	2	

Results: PageRank

- The change in damping factor had minimal effect on the PageRank calculations
- Running PageRank on the graphs with asset-type edges did not yield meaningful results since most edges were distributed
- With the graphs that have metadata-type edges, the results are quite interesting since the edges are clustered around specific groups with high activity
- In the J20 M data, a user, U52, with few files received the highest PageRank score due to working closely with a power user in the same group, U11
- PageRank has significant potential to allow researchers to identify constellations among users of supercomputing systems by ranking the objects on the system

Results: SimRank

- Since SimRank, even with a parallel implementation, is computationally intensive, 5 iterations were chosen, along with the standard decay factor of 0.80

Data Set	ID	Subject	Best match	Match subject	SimRank Score
J20	U11	Computer science	G70	Computer science	0.043
J20	U85	Staff	Multiple	Staff	0.800
J20	G29	Staff	U91	Staff	0.109
J21	G96	Nuclear physics	G02	Physics	0.518
J21	U70	Accelerator physics	G89	Accelerator physics	0.152
J21	G96	Nuclear physics	G22	Computer science	0.518
J21	G96	Nuclear physics	G75	Physics	0.259

Results: SimRank

- U11 is the primary scientist in the G70 research group, so similarity is expected
- U85 is a generic user account that is utilized by students in a classroom setting, and SimRank accurately identified it as being very similar to all of the other generic student user accounts on the system
- G29 is a group of staff system engineers that maintain the supercomputer, and SimRank accurately identified U91 as most similar with a score of 0.106. It is interesting that the next most similar staff user has a score of 0.084

Results: SimRank

- G96, a physics research group, and G02 have a similarity score of 0.518. G02 is the default group of the U03 physics researcher. Interestingly, G96's most similar user is U03, with a lower similarity score of 0.207, so SimRank tends to favor relationships between vertices of the same type
- U70 is one of the leading accelerator physics researchers responsible for the G89 group, so the high similarity is expected
- G22 is the default group of a user that is a researcher in the G75 group, and since the research in G96 and G75 is similar, G96 and G22 are also related, leading to the identification of a small constellation
- Additional modification to the graph would lead to even more useful SimRank results

Observations

- Since the data sets are so large, the Constellation Data Service does an effective job of efficient memory management while allowing for fast traversal of the graph
- The introduction of the VKey in CDSAnalytics gives algorithms and users the ability to find a vertex in the J20 and J21 data sets quickly by utilizing a mapped index that can fit into the memory of a high performance system
- Even though the adjacency matrix used for SimRank did not contain the file vertices in the data set, the similarity scores between users and groups correlated with actual similarities in research areas
- Therefore, placing a greater emphasis on users and groups, while considering quantity of file connections, can lead to identifying constellations more quickly

Observations

- The algorithms are computationally demanding for large data sets, thus requiring efficient parallel performance and memory usage
- Since CDSAnalytics and DATAnalytics were written as modules that can be customized, these programs could be integrated into the CDS itself, or the CDS could automatically perform system calls to CDSAnalytics and DATAnalytics to generate new results
- Regular PageRank and SimRank comparisons of the graph snapshots could identify the hot spots of high activity, and these identified hot spots could then be compared with Titan usage logs to determine a correlation

Future Work

- Future improvements could include additional modifications to the graph algorithms to generate custom results
- For example, the various edge types could be weighted differently, or groups in the same area of research could be directly linked with tag vertices
- While these graph algorithms are a solid foundation for analysis of the file system, implementation into a production CDS environment could yield additional insight into user behavior and allow for real-time analysis that would be helpful to high performance file system engineers and administrators
- New experimental algorithms could be tested to attempt to extrapolate relationships among all of the file system entities more efficiently
- This experimental graph research of a high performance file system could ultimately lead to better collaboration among researchers of various scientific disciplines and even more efficient utilization of limited high performance computing resources

Acknowledgements

- MS PILOT Committee
  - Dr. Michael W. Berry, Major Professor
  - Dr. Arvind Ramanathan
  - Dr. Audris Mockus
- Technology Integration Group at Oak Ridge National Laboratory

Acknowledgements

- This research used resources of the Oak Ridge Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC05-00OR22725.

OAK RIDGE National Laboratory | OAK RIDGE LEADERSHIP COMPUTING FACILITY

References

- E. Strohmaier, J. Dongarra, H. Simon, and M. Meuer. TOP500 Supercomputer Sites. http://www.top500.org, 2016.
- Titan Cray XK7. https://www.olcf.ornl.gov/computing-resources/titan-cray-xk7/.
- Lustre Basics. https://www.olcf.ornl.gov/kb articles/lustre-basics/.
- S. S. Vazhkudai, J. Harney, et al. Constellation: A Science Graph Network for Scalable Data and Knowledge Discovery in Extreme-Scale Scientific Collaborations. In *IEEE Workshop on Big Data Metadata and Management*, 2016.
- S. Brin and L. Page. Reprint of: The anatomy of a large-scale hypertextual web search engine. In *Computer networks* 56(18):3825-3833, 2012.
- D. Austin. How Google finds your needle in the webs haystack. In *American Mathematical Society Feature Column* 10:12, 2006.
- G. Jeh and J. Widom. SimRank: a measure of structural-context similarity. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2002.
- A. Goel, A. Sharma, D. Wang, and Z. Yin. Discovering similar users on Twitter. In *11th Workshop on Mining and Learning with Graphs*, 2013.
- Sparse Matrix. http://www.boost.org/doc/libs/1_42_0/libs/numeric/ublas/doc/matrix_sparse.htm.
- I. Antonellis, H. Garcia-Molina, and C. Chang. Simrank++: query rewriting through link analysis of the click graph. In *Proceedings of the VLDB Endowment* 1(1):408-421, 2008.

The Constellation Project

Andrew W. Nash
Department of Electrical Engineering and Computer Science
Tickle College of Engineering
University of Tennessee
Knoxville, TN 37996
anash4@vols.utk.edu

THE UNIVERSITY OF
TENNESSEE
KNOXVILLE