

LDBC Graphalytics

Tim Hegeman

LDBC TUC Meeting
June 2016

LDBC Graphalytics: Graph Analytics Benchmark

Graphalytics is a benchmark for graph analytics; complex and holistic graph computations which may not be easily modeled as database queries, but rather as (iterative) graph algorithms.

LDBC Graphalytics: The Motivation

- Graph analytics has a large number of applications, e.g., identifying key users in a social network, fraud detection in finance, analyzing biological networks.
- Many graph analytics systems exist, but a comprehensive benchmark does not. Alternatives like Graph500 are limited in scope.

LDBC Graphalytics: Progress

- Definition of benchmark elements, implementation of basic toolchain, first implementation of benchmark for 6 systems.
- Accepted VLDB 2016 article, with academic and industry partners.



Outline

1. Introduction
2. Benchmark Definition
3. Graphalytics Toolchain
4. Results
5. Future Plans
6. Conclusion

Benchmark Definition: Overview

Graphalytics consists of **Algorithms**,
Datasets, and **Experiments**.

Experiment: combination of datasets, algorithms, system configurations, and metrics designed to quantify specific properties of the system under test

Benchmark Definition:

Two-Stage Workload Selection Process

Workload (datasets + algorithms) were selected in two stages:

1. Identify common classes of datasets/algorithms (targets *representativeness*).
2. Select datasets/algorithms from common classes such that resulting set is diverse (targets *diversity/comprehensiveness*).

Benchmark Definition: Algorithms

Two-stage selection process for algorithms:

1. Surveys on classes of algorithms used on unweighted and weighted graphs.
2. Selection of algorithms based on computation and message patterns.

Graph	Class (selected candidates)	#	%
Unweighted	Statistics (PR, LCC)	24	17.0%
	Traversal (BFS)	69	48.9%
	Components (WCC, CDLP)	20	14.2%
	Graph Evolution	6	4.2%
	Other	22	15.6%
Weighted	Distances/Paths (SSSP)	17	34%
	Clustering	7	14%
	Partitioning	5	10%
	Routing	5	10%
	Other	16	32%

Benchmark Definition:

Datasets

- Graphalytics uses a typical graph data model;
 - A single collection of vertices and edges.
 - Vertices and edges may have properties.
 - Edges may be directed or undirected.
- Graphalytics does not impose semantics on datasets.
- Mix of 6 real-world graphs from 3 domains (knowledge, social, gaming) + 2 synthetic generators (Datagen, Graph500).

Benchmark Definition: Experiments

Experiments can be divided into 3 categories:

1. **Baseline experiments:** measure how well the system under test performs on a variety of workloads (**algorithm variety**, **dataset variety**).
2. **Scalability experiments:** measure how well the system under test scales. Includes experiments for **horizontal** vs **vertical** scalability and **strong** vs **weak** scalability.
3. **Robustness experiments:** measure the limit and the performance variability of the system under test.

Benchmark Definition: SLA

Execution of an algorithm is considered successful iff it upholds the Graphalytics SLA:

1. The output of the algorithm must pass the validation process.
2. The makespan of the algorithm execution must not exceed one hour.

Benchmark Definition: Validation Process

- Output for every execution of an algorithm is compared to reference output for equivalence.
 - Rules for equivalence are defined per algorithm.
 - Any implementation resulting in correct output is acceptable.

Benchmark Definition: Renewal Process

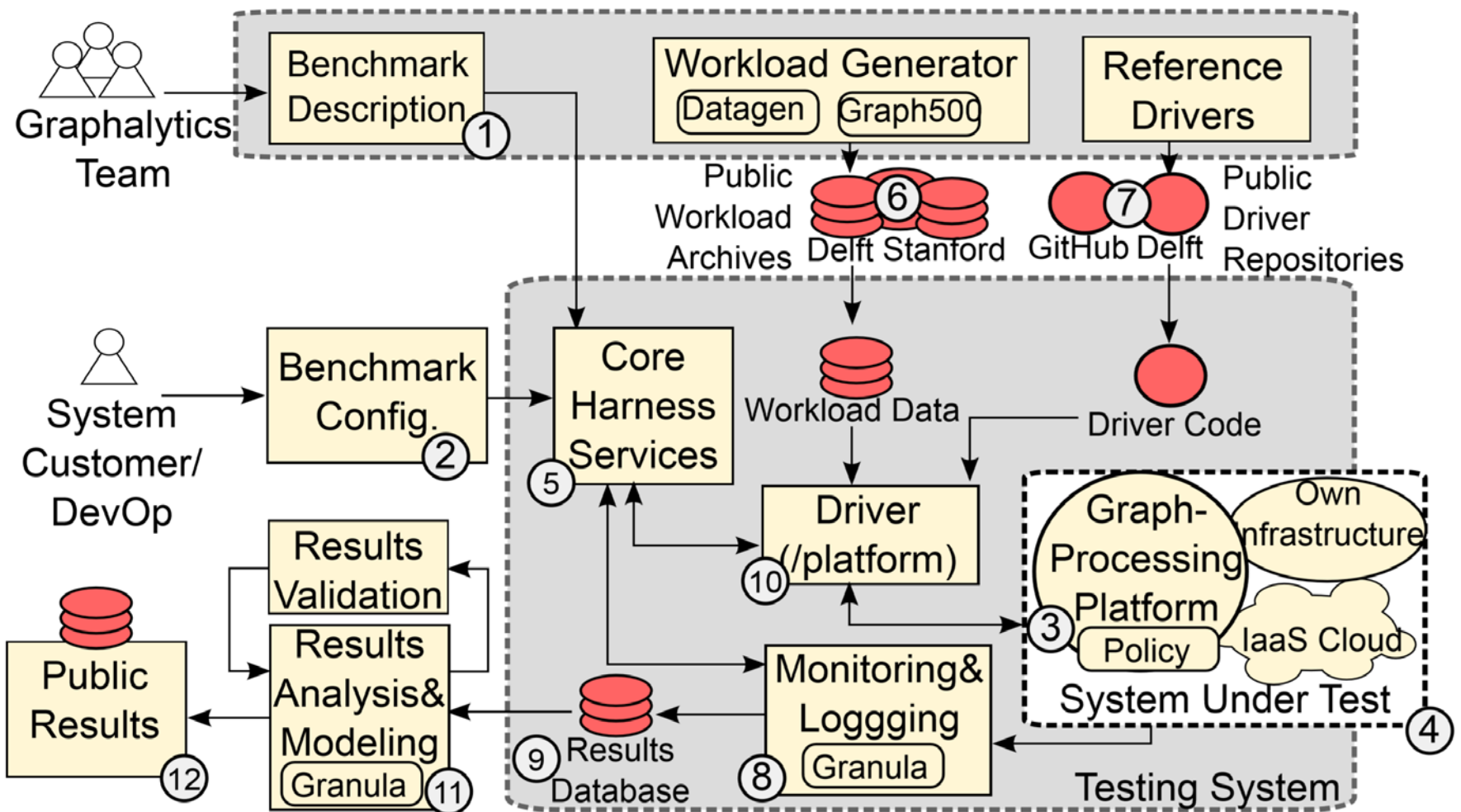
- Field of graph analytics is still rapidly evolving, so need for frequent but structured renewal of the benchmark.
- Every X years, Graphalytics Task Force repeats two-stage selection process to identify representative, diverse workload.

Outline

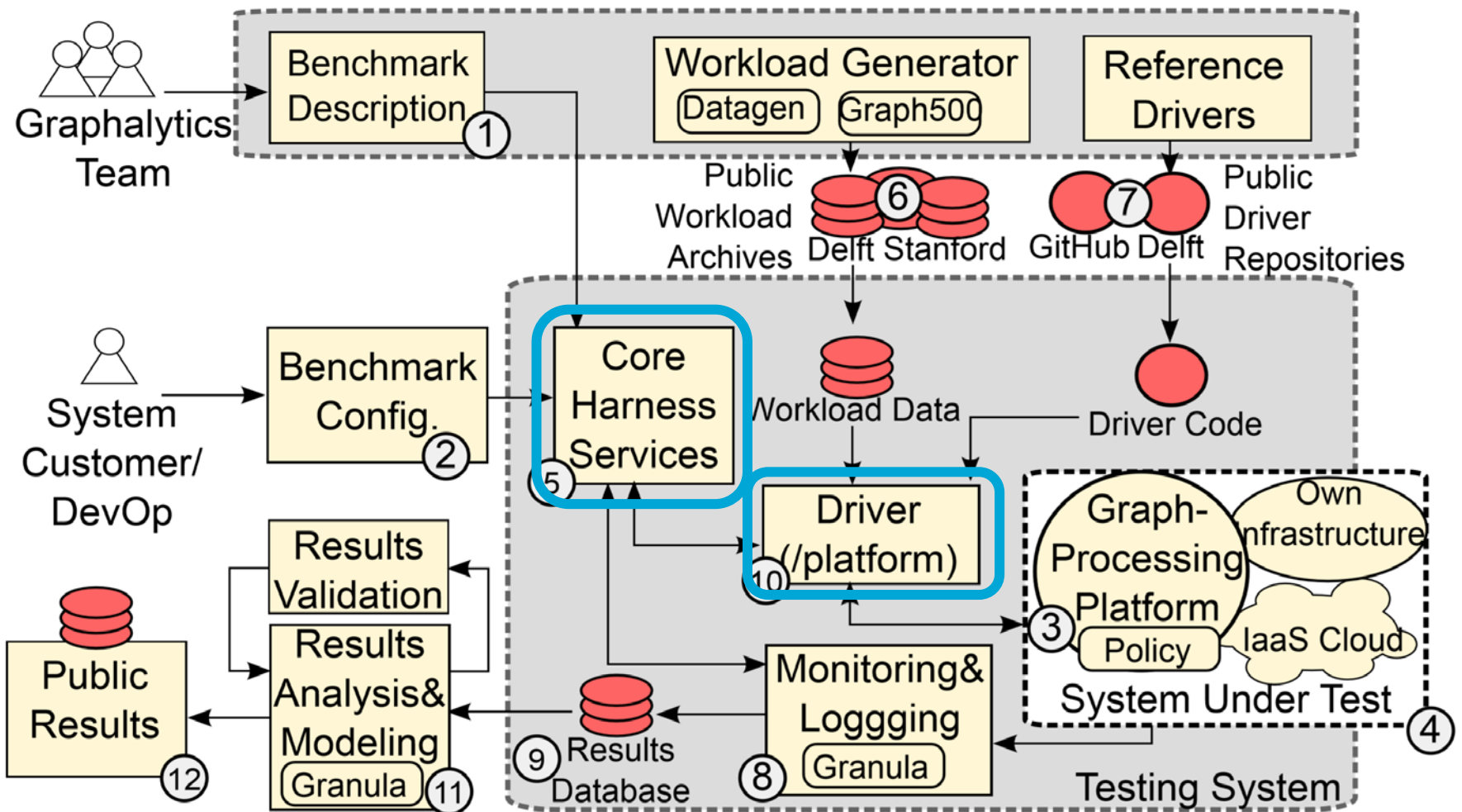
1. Introduction
2. Benchmark Definition
3. Graphalytics Toolchain
4. Results
5. Future Plans
6. Conclusion

<https://www.github.com/tudelft-atlarge/graphalytics>

Graphalytics Toolchain: Architecture



Graphalytics Toolchain: Architecture



Graphalytics Toolchain: Platform Driver

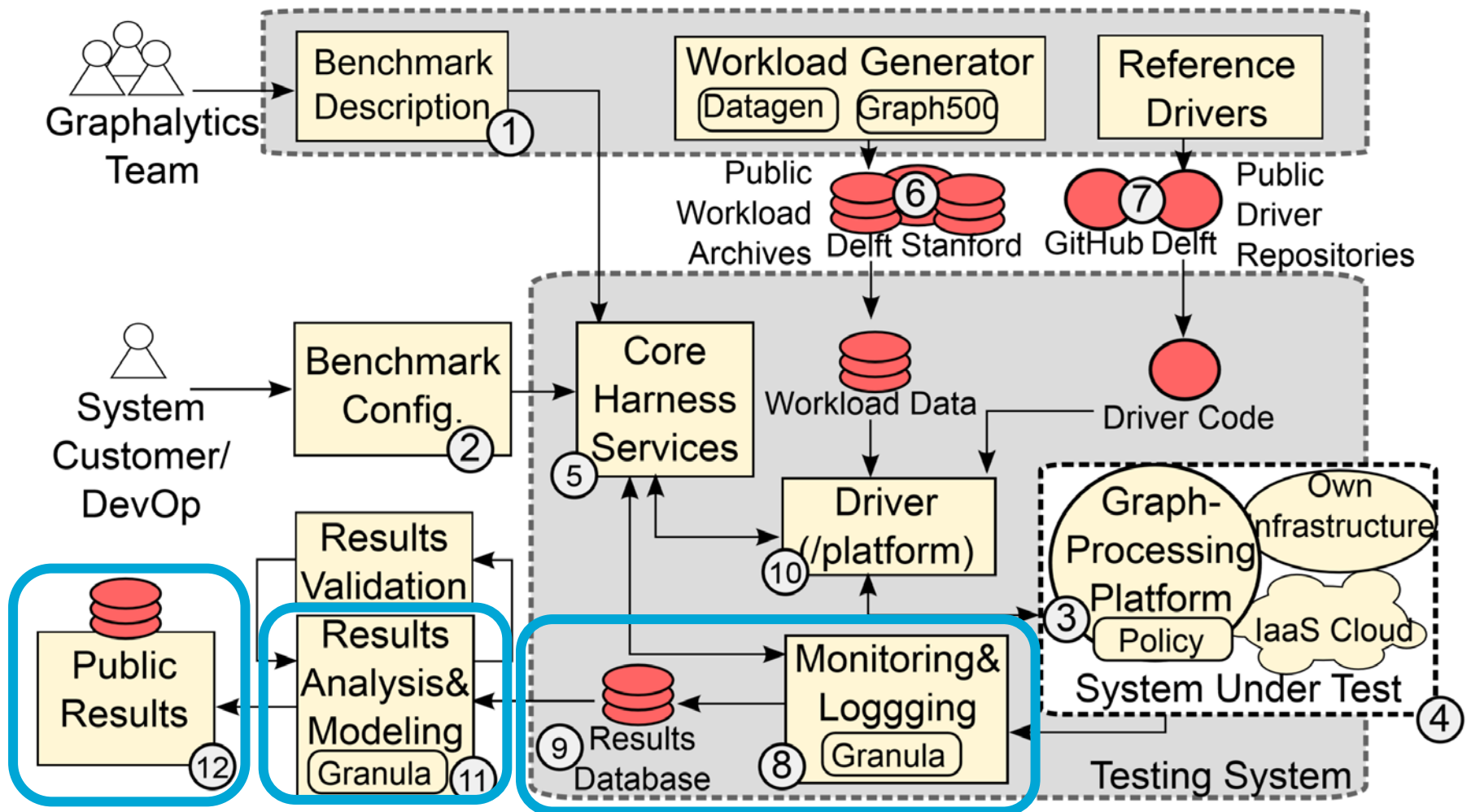
A Platform Driver must provide 3 basic functions:

1. **Upload a graph:** allows for pre-processing of a graph to convert it to a platform-specific format, copy it to a distributed filesystem, insert it into a database, etc.
2. **Execute an algorithm:** execute a single algorithm on an already uploaded graph and store the output on the machine running Graphalytics.
3. **Delete a graph (if needed)**

Graphalytics Toolchain: Benchmark Execution Process

- The Graphalytics harness calls the upload, execute, and delete functions required to complete a given experiment.
- Upload time for each graph and makespan of each algorithm execution are measured by Graphalytics. Processing time must be reported by the system under test through execution logs.

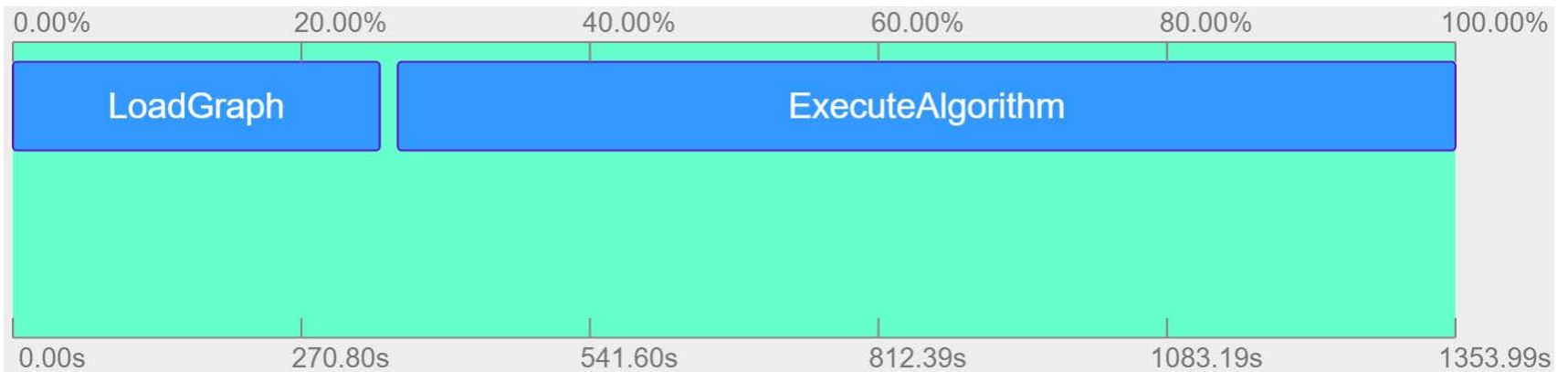
Graphalytics Toolchain: Architecture



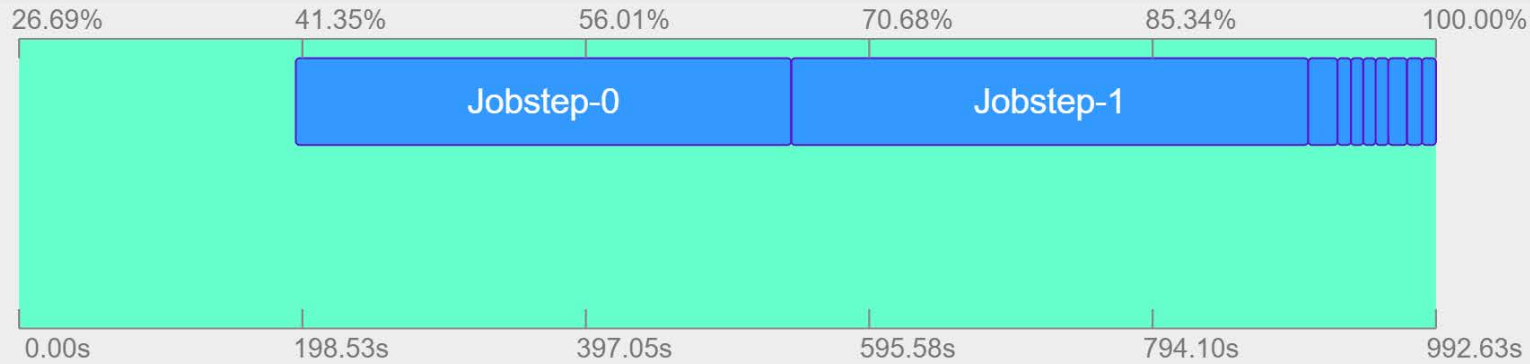
Graphalytics Toolchain: Granula

- Granula is a tool for Monitoring, Modeling, Archiving, and Visualizing the performance of graph analytics systems.
- Basic model (processing time vs overhead) required for benchmark compliance.
- Extended model + system monitoring provide additional insight in performance

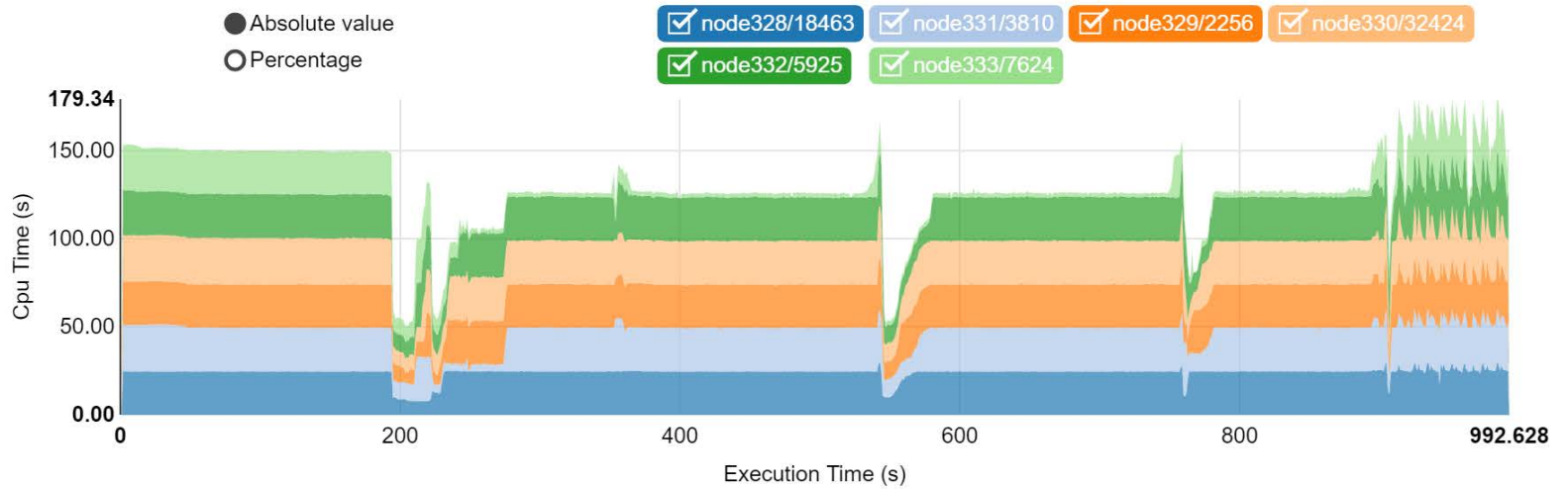
Granula in Action



Granula in Action







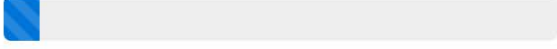
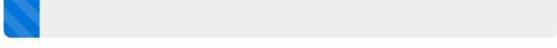
All CPU Memory Network-Snd Network-Rec



Granula in Action

All CPU Memory Network-Snd Network-Rec

Summary of the Job Environment

	Name	Measured value	Theoretical Peak	Percentage
CPU	CpuTime (all nodes, all cores)	128140.03 s	190584.58 s	
Memory	MemAvg (all nodes)	342.06 GB	384.00 GB	
	MemMax (all nodes)	364.64 GB	384.00 GB	
	MemMax (any nodes)	63.25 GB	64.00 GB	
Network	NetVol (SEND, all nodes)	486.44 GB	7444.71 GB	
	NetVol (REC, all nodes)	486.46 GB	7444.71 GB	

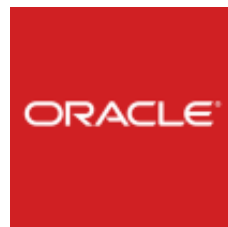
Outline

1. Introduction
2. Benchmark Definition
3. Graphalytics Toolchain
4. Results
5. Future Plans
6. Conclusion

Results:

Experimental Setup (1)

- Graphalytics has been implemented for 3 community-driven platforms (Giraph, GraphX, PowerGraph) and 3 industry-driven platforms (PGX, GraphMat, OpenG).



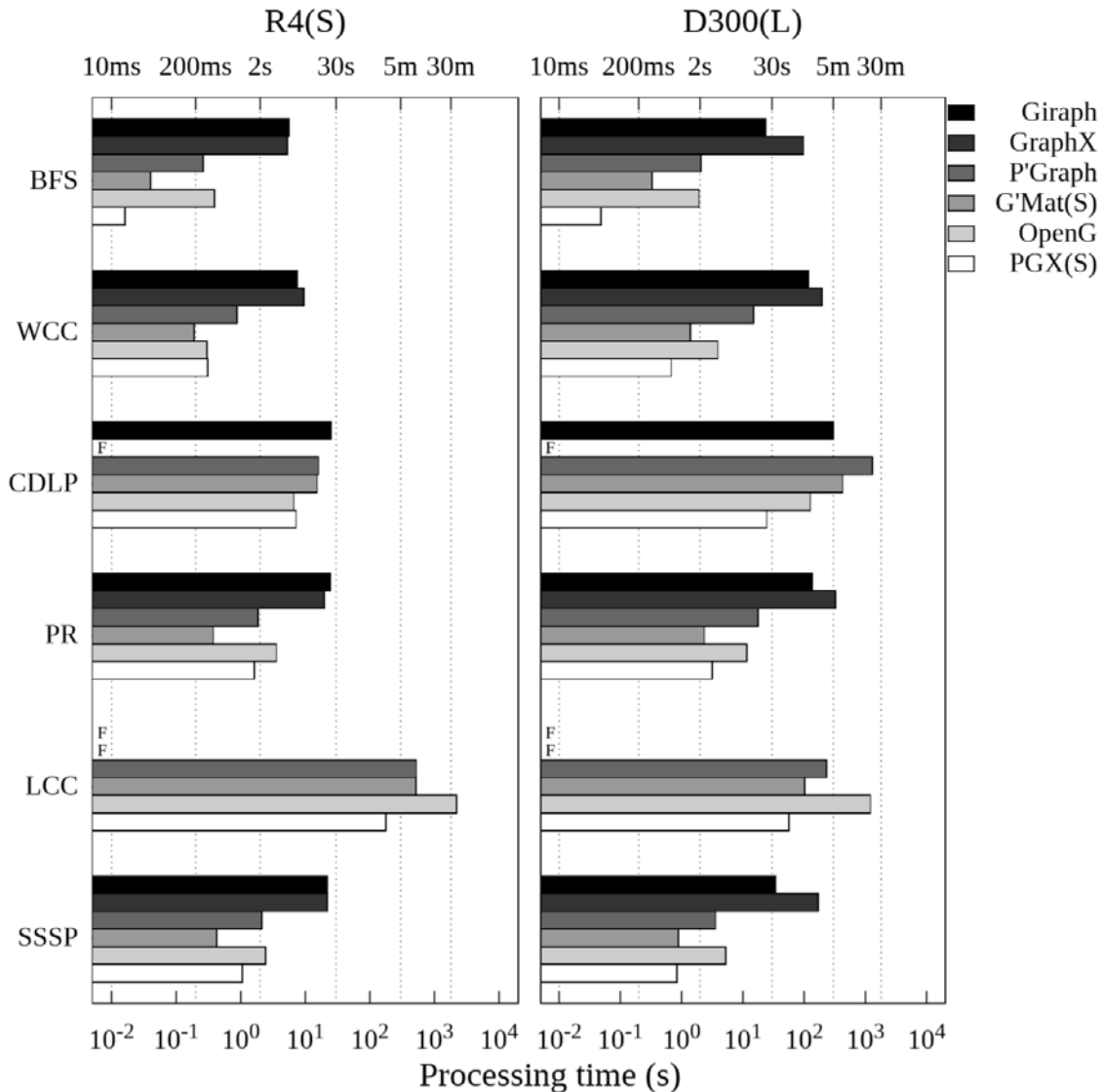
Results:

Experimental Setup (2)

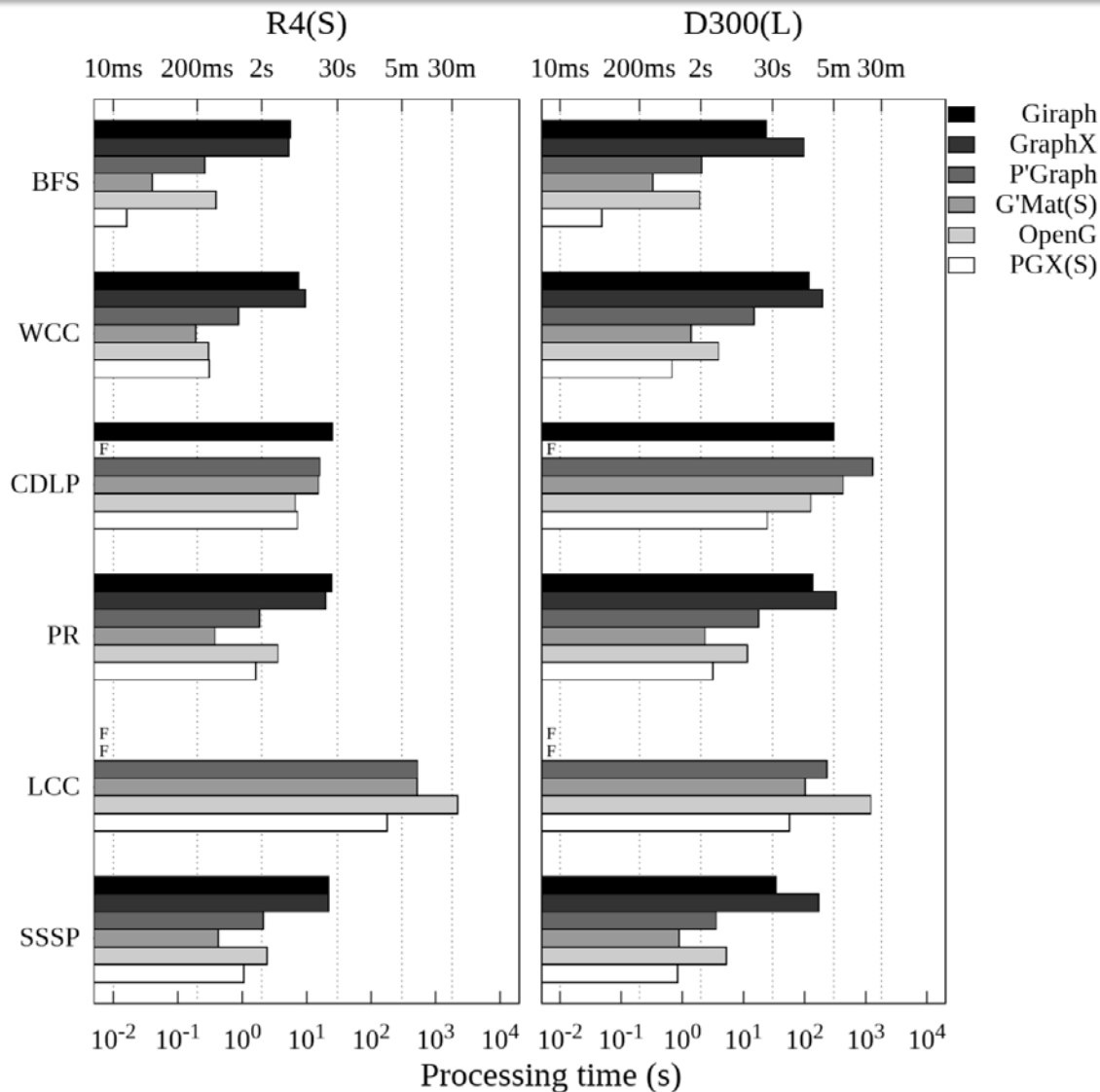
- All experiments run by TU Delft on DAS-5 (Distributed ASCI Supercomputer, the Dutch national supercomputer for Computer Science research).
- Details and additional results can be found in the VLDB article.

Results:

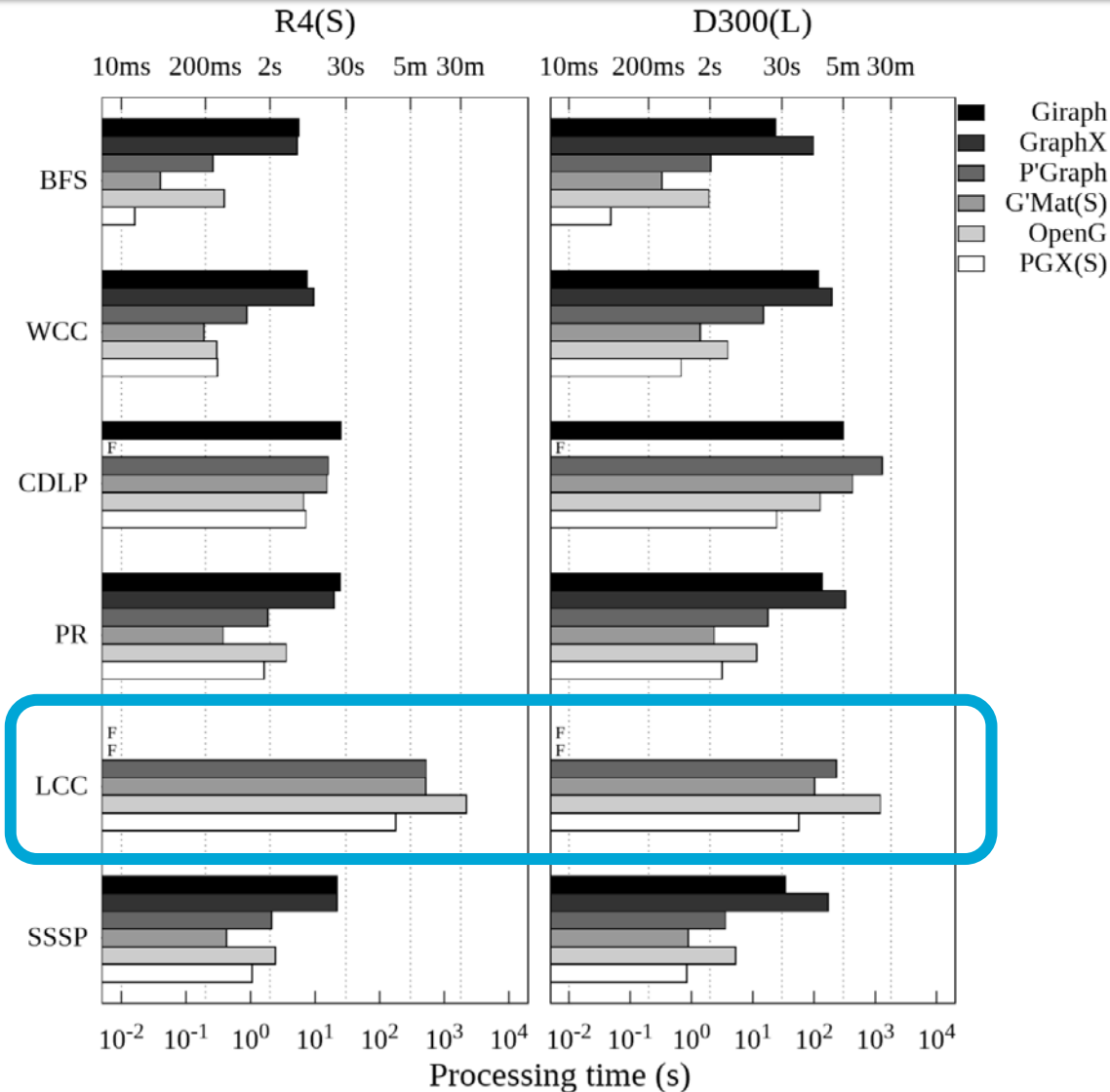
Baseline – Algorithm Variety



Significant variation in relative performance when comparing platforms



LCC slower on small graph due to much larger vertex degrees



Outline

1. Introduction
2. Benchmark Definition
3. Graphalytics Toolchain
4. Results
5. Future Plans
6. Conclusion

Future Plans:

First Public Specification Draft

- Required for first public draft of benchmark specification:
 - Complete definition of execution rules.
 - Auditing process.

Future Plans: Results Archive

- Periodically updated repository of audited results, including competition (similar to Top500, Graph500)
- Key question: How to present results across experiments?

Future Plans: Extending the Toolchain

- Optionally include Granula performance breakdown in public results.
- Addition of low-level performance counters to Granula.
- Automated bottleneck detection using Granula.

Outline

1. Introduction
2. Benchmark Definition
3. Graphalytics Toolchain
4. Results
5. Future Plans
6. Conclusion

Conclusion

- We defined Graphalytics: a benchmark for graph analytics.
- We published our experiences with 6 platforms.
- First public draft for the specification is coming soon.

LDBC Graphalytics

Tim Hegeman

LDBC TUC Meeting
June 2016