GoFFish: A Sub-Graph Centric Framework for Large-Scale Graph Analytics

Guide:
Prof. Dinkar Sitaram
Department of Computer Science and Engineering
PES Institute of Technology

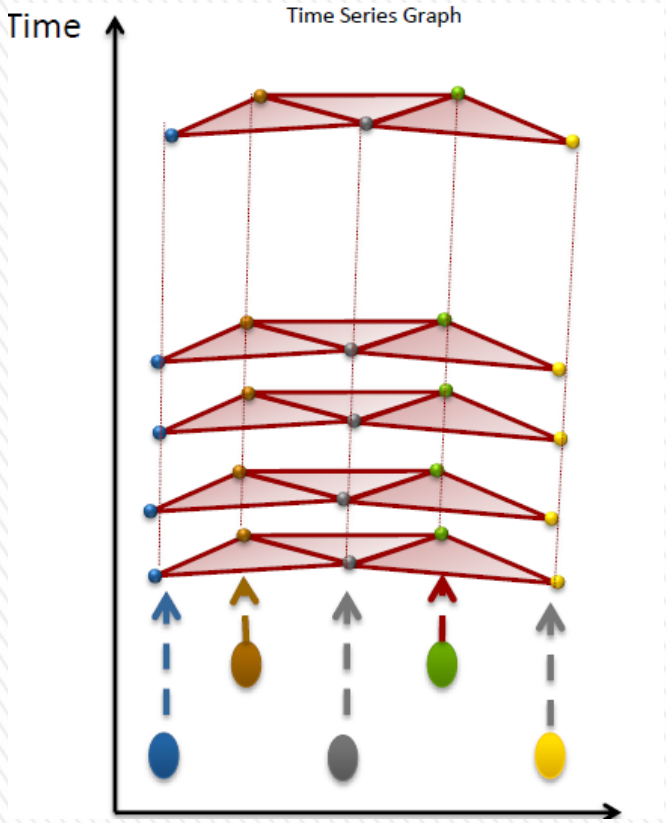
Ms. Prafullata
Department of Computer Science and Engineering
PES Institute of Technology

Mentor:
Prof. Yogesh Simmhan
Department of SERC
Indian Institute of Science

Team Members:

Bhavani B	1PI11IS027
Anushree PK	1PI11IS017
Mithilesh K G	1PI11IS059

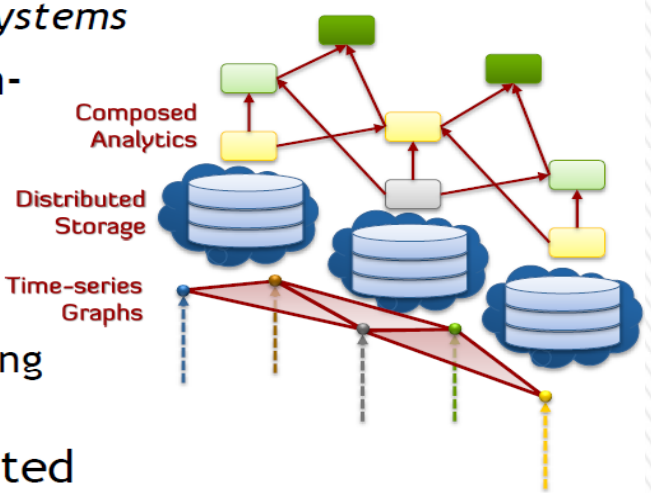
Analytics on Time series Graphs and the GoFFISH Software Platform



•Time series graph Ṗ is a series of graph instances Ǵ , over time.

•A Graph instance ǵ in Ǵ , is a ordered pair ǵ = (G , ṭ) where G=(V,E) is a ordered pair comprising set V of vertices and or nodes together with set of E edges ; ṭ denotes a time associated with a given Graph instance.

- Platform to *store*, *compose* & *execute* analytics on *time-series graph* datasets
  - *At scale, on distributed systems*
- **GoFS**: Distributed Graph-oriented File System
- **Gopher**
  - Compose sub-graph centric complex analytics
  - Executed on ***Floe*** streaming dataflow engine
- Data & Compute collocated



Problem Definition

Using the GoFFish framework:

- ▶To store the real time graphs as – time series and subgraph-centric formats
- ▶To compare the efficiencies of both based on the input query

Time Series Format

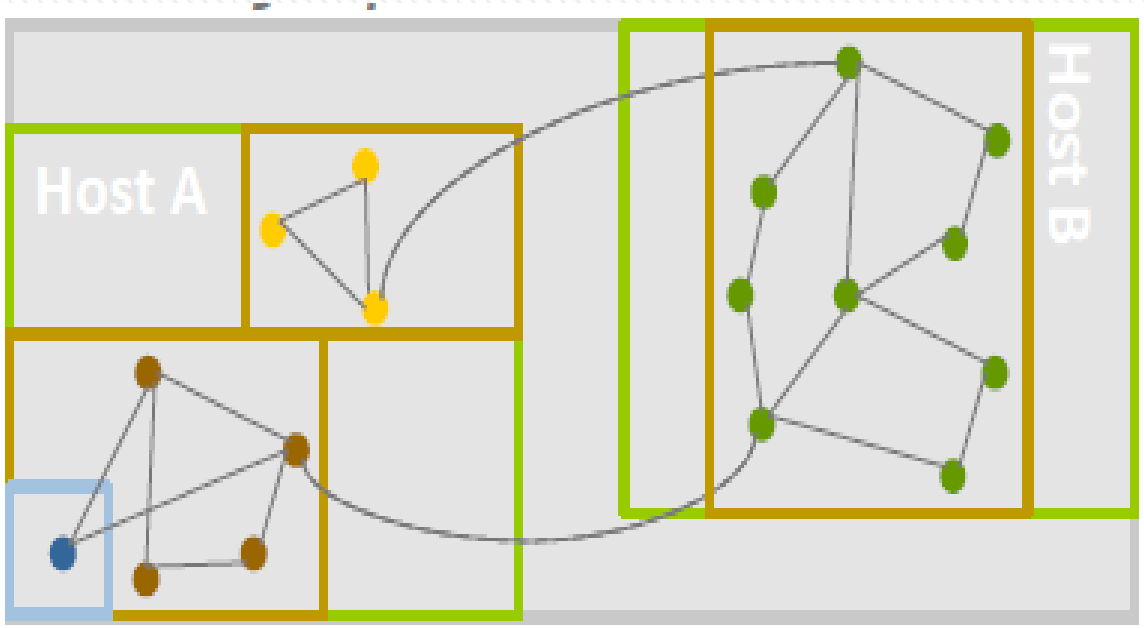
Timestamp	Graph
t0	graph_instance_0
.	
.	
.	
tn	graph_instance_n

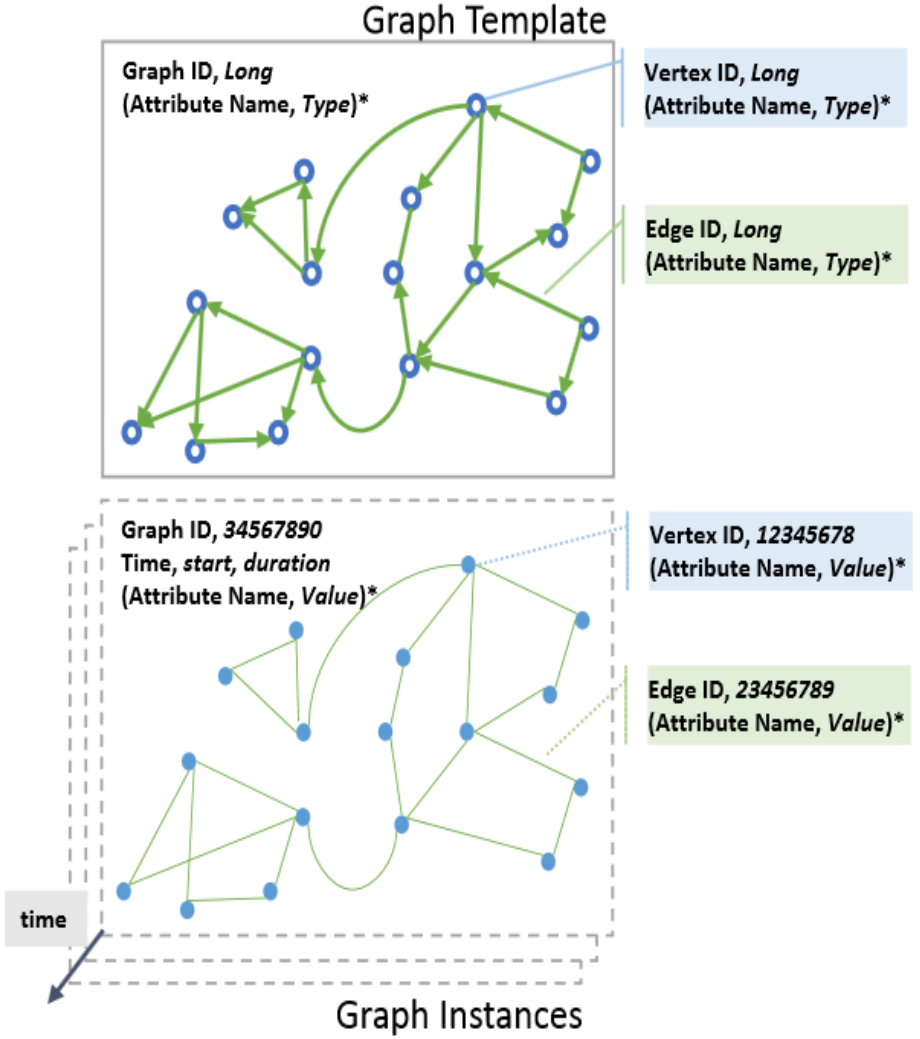
Subgraph Centric Format

Subgraph	t0	t1	t2	.	.	tn
Subgraph_1	val0	val1	val2			valn
.						
.						
.						
Subgraph_n	.	.	.			.

Our Approach

- Single Node Installation
- Load a graph and submit a Gopher job
- Compute time taken to run the Gopher job
- Print the result

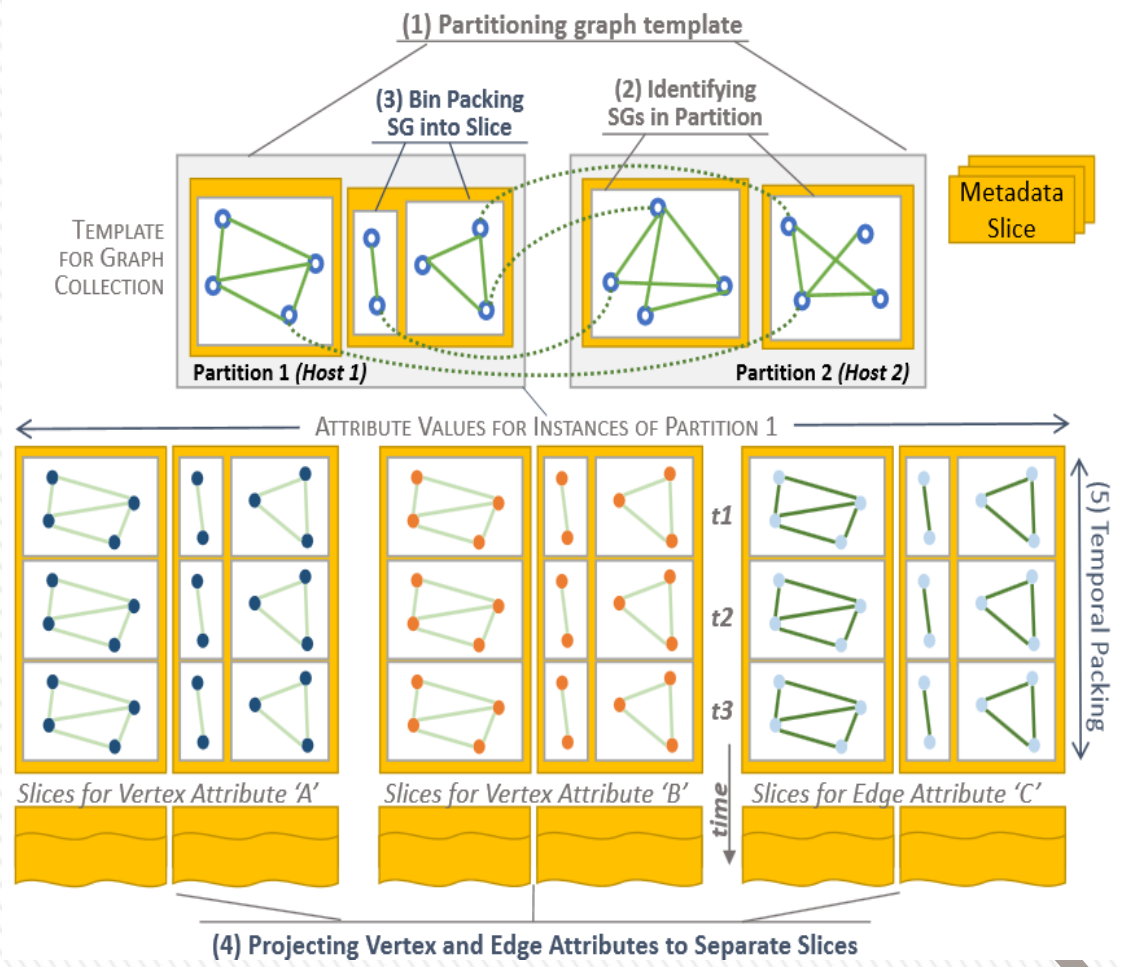




Temporal and Subgraph bin packing

Temporal instance packing - packing
nearby instances together within a single slice. Thus, an attribute slice storing a subgraph instance values will contain adjacent instances, and the slice will contain instances that span a time duration.

Subgraph bin packing-partitioning large graphs results in partitions with hundreds of
subgraphs with highly variable vertex and edge counts, causing imbalances in slice read times
across subgraphs and also imbalances in execution. y having a fixed number of slices (bins) and packing multiple subgraphs into a slice (bin) to balance the number of vertices/edges/vertices+edges in a bin, we limit the slice size and count



Set serializer:instancegroupingsize to ALL and set serializer:numsubgraphbins to a large value

Our Approach

- Set the parameters:
serializer:instancegroupingsize to ALL
serializer:numsubgraphbins to 10000
- Compute time taken

RESULTS:

Algorithm : Vertex Count

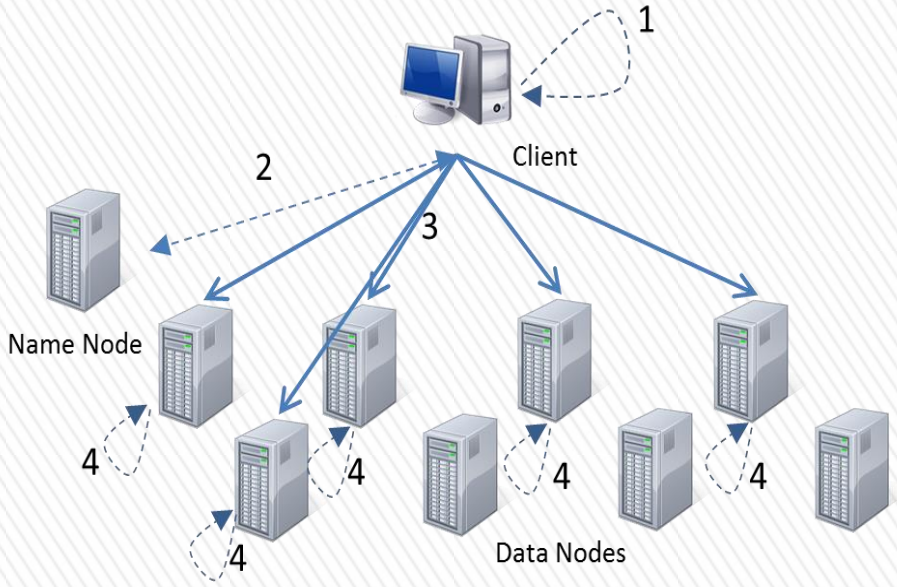
Graph Datasets Source: Stanford Network Analysis Platform (**SNAP**)

Graph	Graph Size(Nodes)	No. of Instances	No. of Partitions	Time Diff(without params set) in ns	Time Diff(with params set) in ns
P2P -1	6,301	4	1	171580	147348
P2P-1	6,301	4	2	117204	140786
P2P-2	62,586	4	1	250666	269528
P2P-2	62,586	4	2	269528	230438
P2P-2	62,586	200	1	188698	184639
P2P-2	62,586	200	2	165393	138883
Citation	34,546	4	4	236474	163621

GoFFISH Cluster

- One of the machine in GoFFish cluster acts as head node, other nodes are referred to as worker nodes
- All machines in GoFFish cluster should be able to do password less ssh among each other
- Deploy GoFFish
- Load sample graph in GoFS
- Run sample Gopher job

Our Cluster:-
10.10.1.58 – Head node + Client + Worker node
10.10.1.59 & 10.10.1.60 – Worker nodes

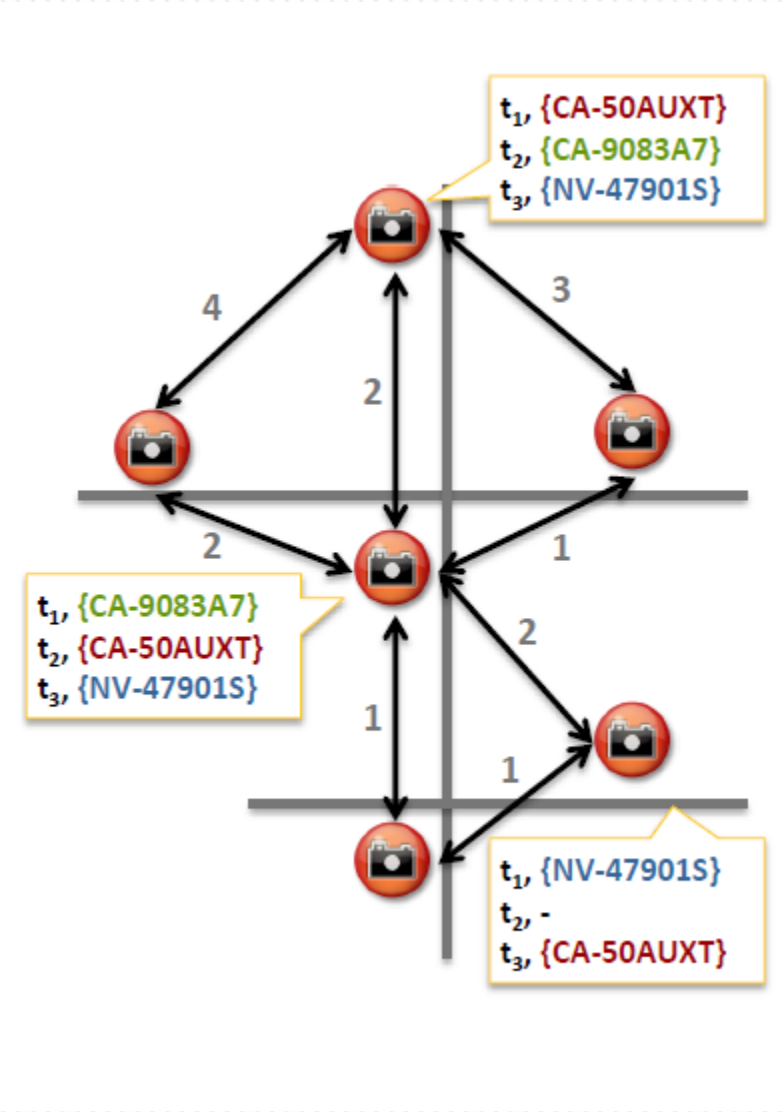


Approach

Dataset :- Road network graph

Vehicle route tracking using traffic cams
–Time-series graph of sync camera snapshots
–Sensors are vertices
–*Edges* are road connectivity w/ distance weight
Graph instance is image metadata every *N* sec
–License plate, vehicle color, direction, speed
Urban

Algorithms :- Vert-count,Connected components



GoFFISH Cluster : Loggers

There are three Log files .
The Subgraph log contains information about the lifecycle of the app.
The Partition log contains the time for each of the steps as well the total times for Compute , IO and both.
The Container log is a combination of the other two.

Dataset Information: CA Road Network
Number of instances : 4
Number of Partitions : 3
Algorithm : Connected Components

Dataset statistics	
Nodes	1965206
Edges	2766607

Results obtained from Logger

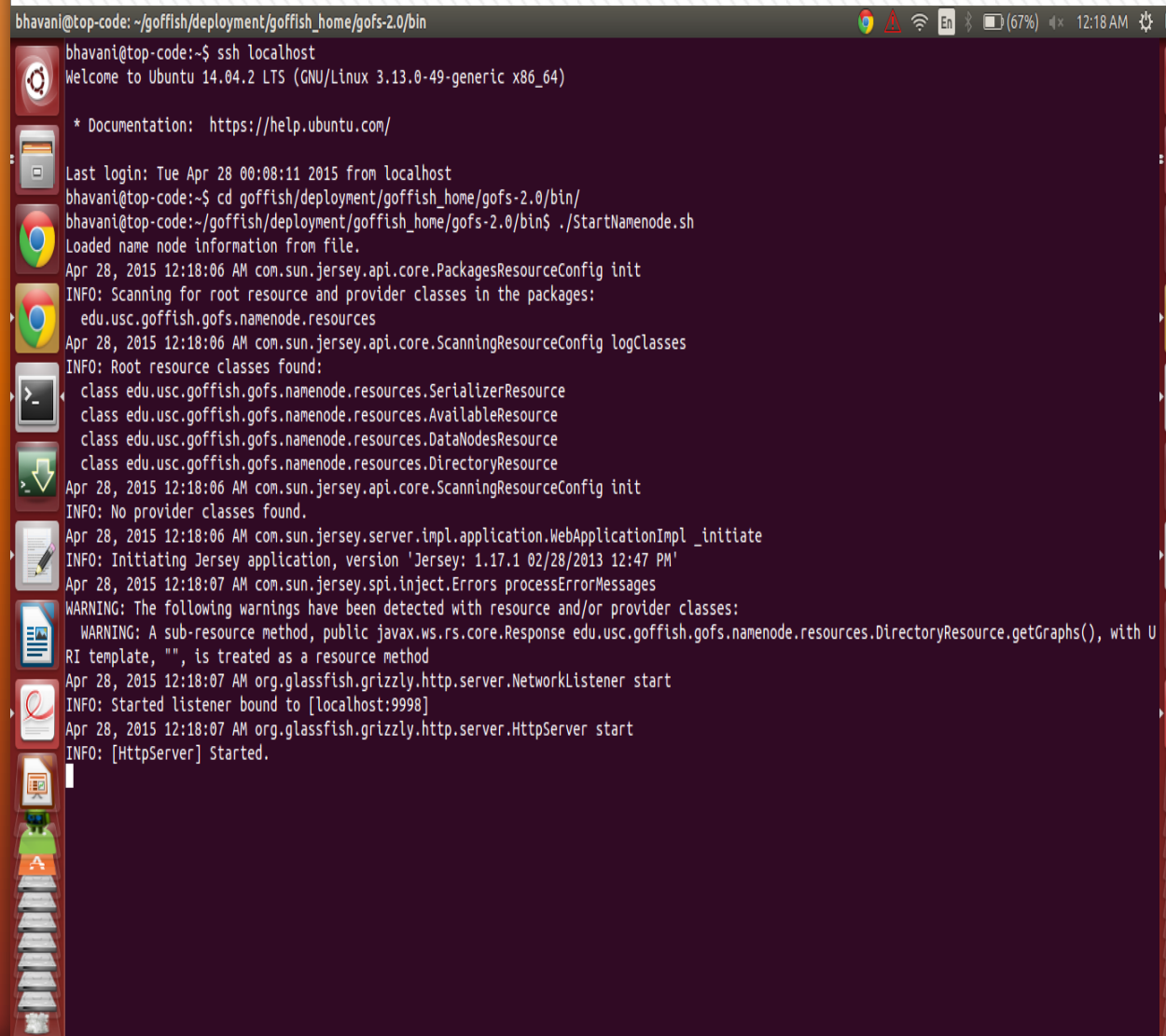
Before setting the slicing pointers

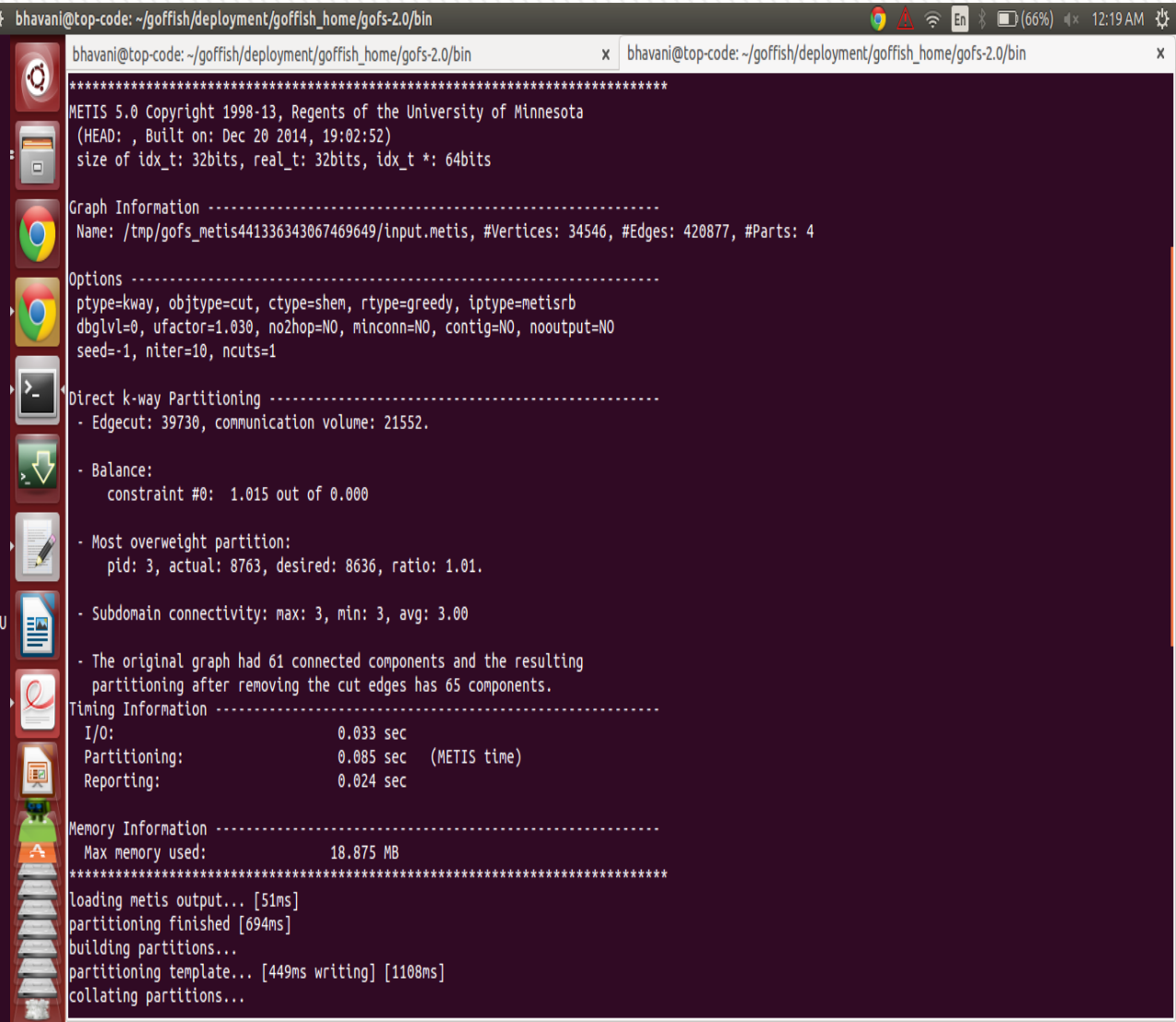
```
PERF,1,,0,3,1430287604262,2015-04-29 11:36:44.262+0530,ALL_SG_COMPUTE_TASKS,0
PERF,1,0,3,1430287604262,2015-04-29 11:36:44.262+0530,PERF.PART.SS_WALL_TIME,1430287604256,1430287604262,6
STATE,1,0,4,1430287604397,2015-04-29 11:36:44.397+0530,STATE.PART.START_SS
STATE,1,0,4,1430287604397,2015-04-29 11:36:44.397+0530,STATE.PART.PART_VOTE_HALT
STATE,1,0,4,1430287604402,2015-04-29 11:36:44.402+0530,STATE.PART.END_SS
PERF,1,,0,4,1430287604402,2015-04-29 11:36:44.402+0530,ALL_SG_COMPUTE_TASKS,0
PERF,1,0,4,1430287604402,2015-04-29 11:36:44.402+0530,PERF.PART.SS_WALL_TIME,1430287604397,1430287604402,5
STATE,1,0,4,1430287604402,2015-04-29 11:36:44.402+0530,STATE.PART.HALT
PERF,1,0,4,1430287604403,2015-04-29 11:36:44.403+0530,PERF.PART.TOTAL_WALL_TIME,1430287600063,1430287604403,4399
PERF,1,,0,4,1430287604403,2015-04-29 11:36:44.403+0530,PERF.PART.TOTAL_SG_COMPUTE_TASK_DURATION,601
PERF,1,,0,4,1430287604403,2015-04-29 11:36:44.403+0530,PERF.PART.TOTAL_SG_COMPUTE_SEQ_DURATION,0
PERF,1,,0,4,1430287604404,2015-04-29 11:36:44.404+0530,PERF.PART.TOTAL_SG_MSG_SEQ_DURATION,0
STATE,1,0,0,1430287604404,2015-04-29 11:36:44.404+0530,STATE.PART.STOP
```

After setting the slicing pointers

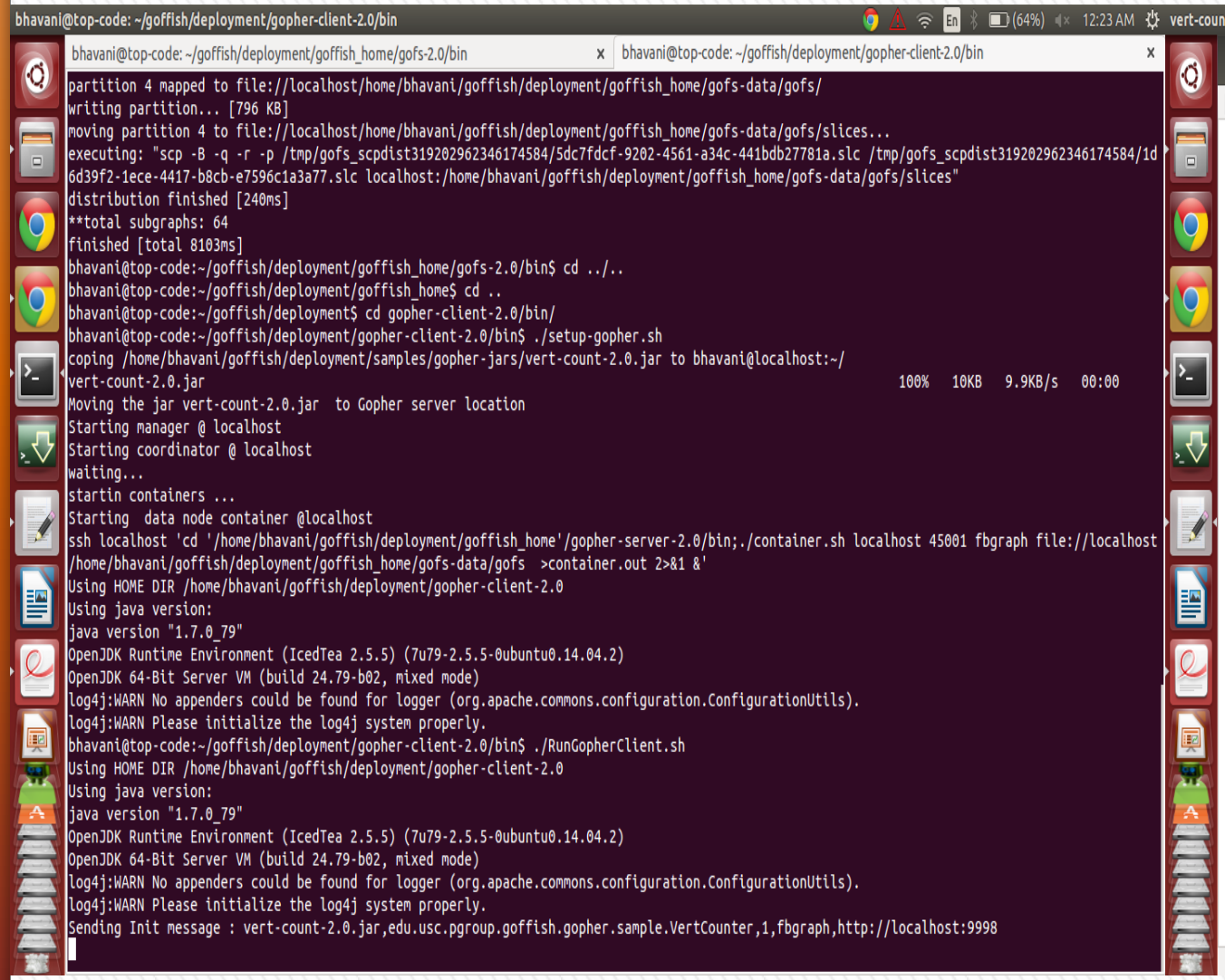
```
PERF,1,,0,3,1430288257812,2015-04-29 11:47:37.812+0530,PERF.PART.SG_MSG_SEQ_DURATION,0
STATE,1,0,4,1430288257929,2015-04-29 11:47:37.929+0530,STATE.PART.START_SS
STATE,1,0,4,1430288257929,2015-04-29 11:47:37.929+0530,STATE.PART.PART_VOTE_HALT
STATE,1,0,4,1430288257934,2015-04-29 11:47:37.934+0530,STATE.PART.END_SS
PERF,1,,0,4,1430288257934,2015-04-29 11:47:37.934+0530,ALL_SG_COMPUTE_TASKS,0
PERF,1,0,4,1430288257934,2015-04-29 11:47:37.934+0530,PERF.PART.SS_WALL_TIME,1430288257929,1430288257934,5
STATE,1,0,4,1430288257935,2015-04-29 11:47:37.935+0530,STATE.PART.HALT
PERF,1,0,4,1430288257936,2015-04-29 11:47:37.936+0530,PERF.PART.TOTAL_WALL_TIME,1430288253517,1430288257936,4419
PERF,1,,0,4,1430288257936,2015-04-29 11:47:37.936+0530,PERF.PART.TOTAL_SG_COMPUTE_TASK_DURATION,628
PERF,1,,0,4,1430288257936,2015-04-29 11:47:37.936+0530,PERF.PART.TOTAL_SG_COMPUTE_SEQ_DURATION,0
PERF,1,,0,4,1430288257936,2015-04-29 11:47:37.936+0530,PERF.PART.TOTAL_SG_MSG_SEQ_DURATION,0
STATE,1,0,0,1430288257936,2015-04-29 11:47:37.936+0530,STATE.PART.STOP
```

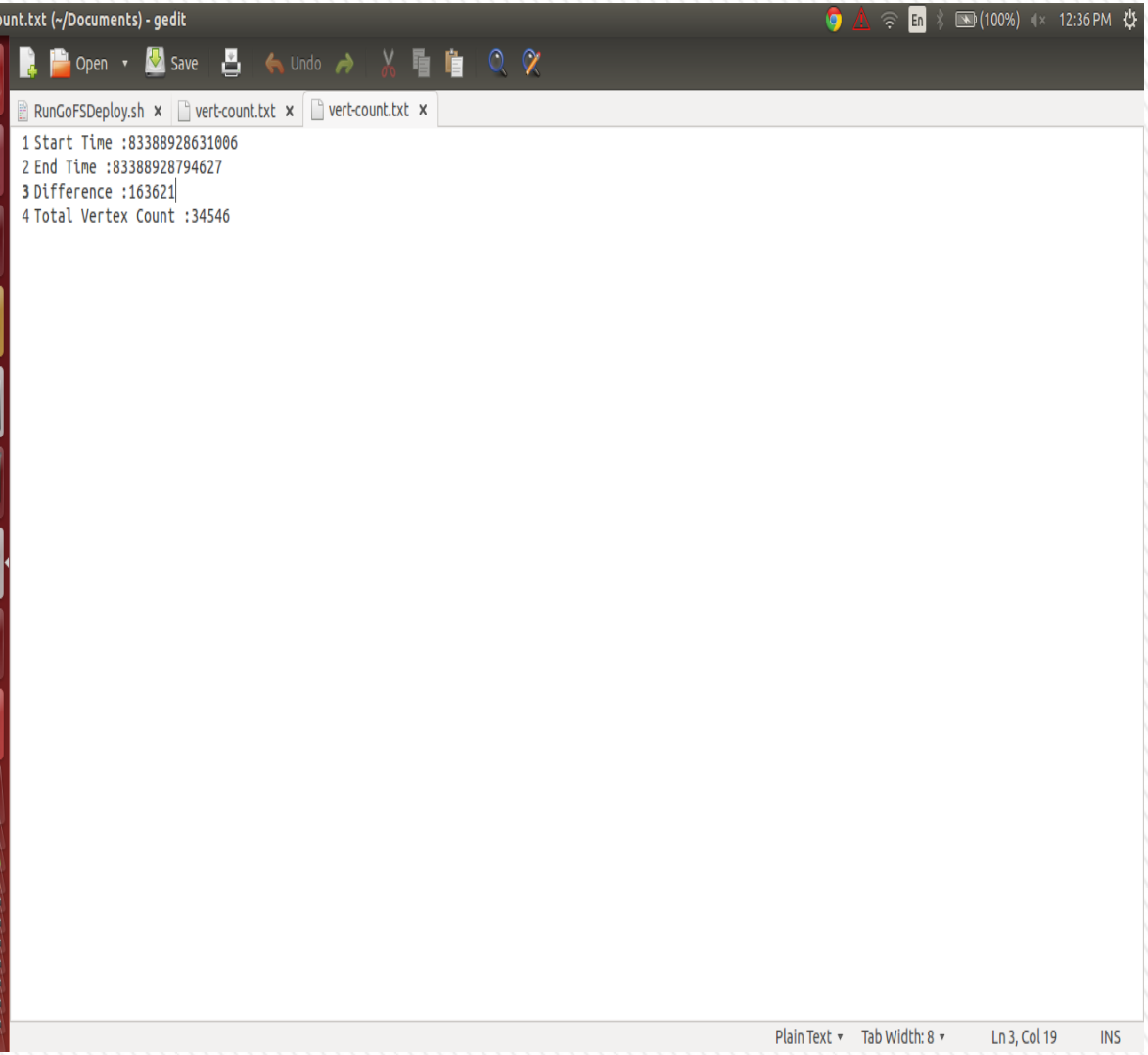
Screenshots





Screenshots





Challenges

- Single Node Installation

-- The pom links to some packages were missing

-- we manually ran every step in the script with new links to the packages.

- Understanding bin packing schemes and how it affect our storage format
- Identifying the slicing pointers from the GoFFish code base.

- Cluster Installation

-- Java shared libraries (libjli.so) were initially not found

Literature Survey

▶GoFFish: A Sub-Graph Centric Framework for Large-Scale Graph Analytics - Indian Institute of Science, Bangalore 560012 India, University of Southern California, Los Angeles CA 90089 USA, November 26, 2013

▶Scalable Analytics over Distributed Time-series Graphs using GoFFish - Indian Institute of Science, Bangalore 560012 India, University of Southern California, Los Angeles CA 90089 USA, June 23, 2014

▶Chronos: A Graph Engine for Temporal Graph Analysis - Tsinghua University, University of Science and Technology of China, Microsoft Research

Thank you