CREST 戦略的創造研究推進事業
Core Research for Evolutional Science and Technology

JST Japan Science and Technology Agency

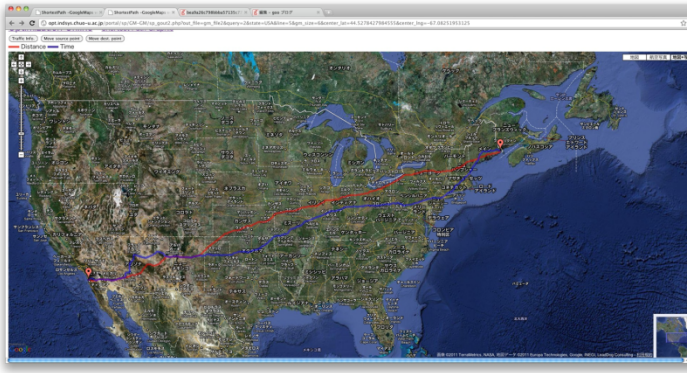
Large-Scale Graph Analysis for Cyber Security on Post Peta-Scale Supercomputers

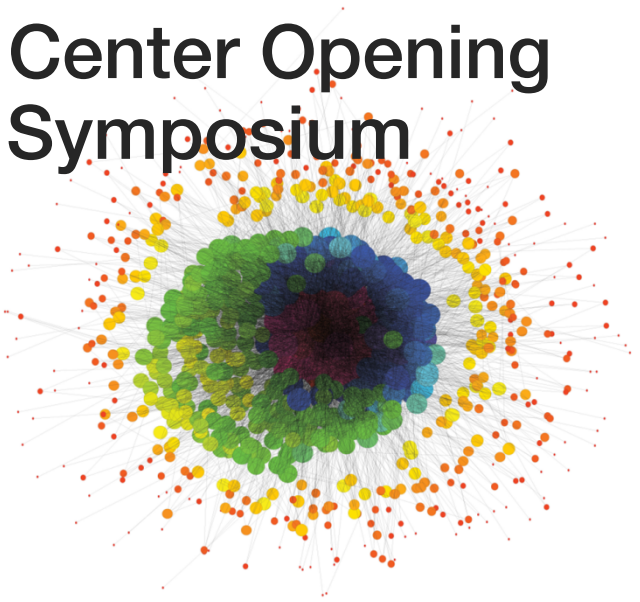
Katsuki Fujisawa

Kyushu University (IMI) & JST CREST

July 8, 2015

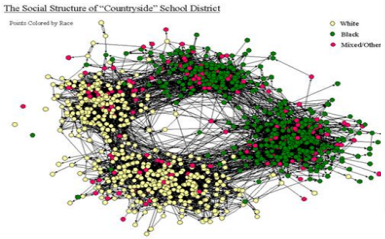
Kyushu Universuty Cybersecurity Center Opening Ceremony and Cybersecurity Symposium

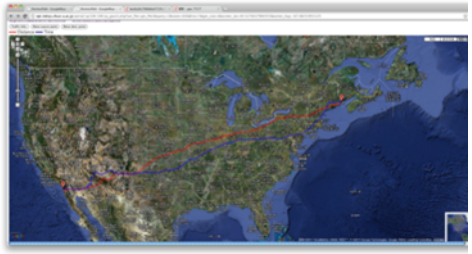




Advanced Computing and Optimization Infrastructure for Extremely Large-Scale Graphs on Post Peta-Scale Supercomputers

- **JST**(Japan Science and Technology Agency) **CREST**(Core Research for Evolutionaly Science and Technology) **Project** (Oct, 2011 ～ March, 2017)
- **3 groups, over 60 members**
  1. Fujisawa-G (Kyushu University) : Large-scale Mathematical Optimization
  2. Suzumura-G (University College Dublin, Ireland) : Large-scale Graph Processing
  3. Sato-G (Tokyo Institute of Technology) : Hierarchical Graph Store System
- **Innovative Algorithms and implementations**
  - Optimization, Searching, Clustering, Network flow, etc.
- Extreme Big Graph Data for emerging applications
  - **2^{30} ~ 2^{42} nodes** and **2^{40} ~ 2^{46} edges**
  - **Over 1M threads** are required for real-time analysis
- Many applications on post peta-scale supercomputers
  - Analyzing massive cyber security and social networks
  - Optimizing smart grid networks
  - Health care and medical science
  - Understanding complex life system

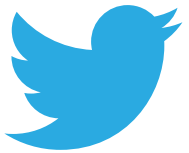




Background

- **The extremely large-scale graphs that have recently emerged in various application fields**
  - US Road network : 58 **million** edges
  - Twitter fellow-ship : 1.47 **billion** edges
  - Neuronal network : 100 **trillion** edges

Social network



Twitter

61.6 million nodes & 1.47 billion edges

- **Fast and scalable graph processing** by using **HPC**

Neuronal network @ Human Brain Project

89 billion nodes & 100 trillion edges

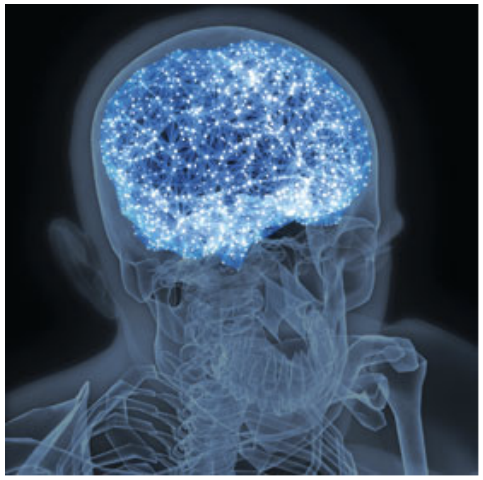
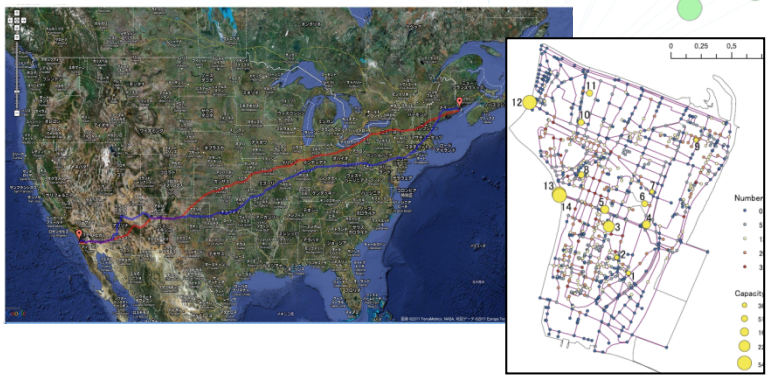


Image: Illustration by Mirko Ilic

US road network

24 million nodes & 58 million edges

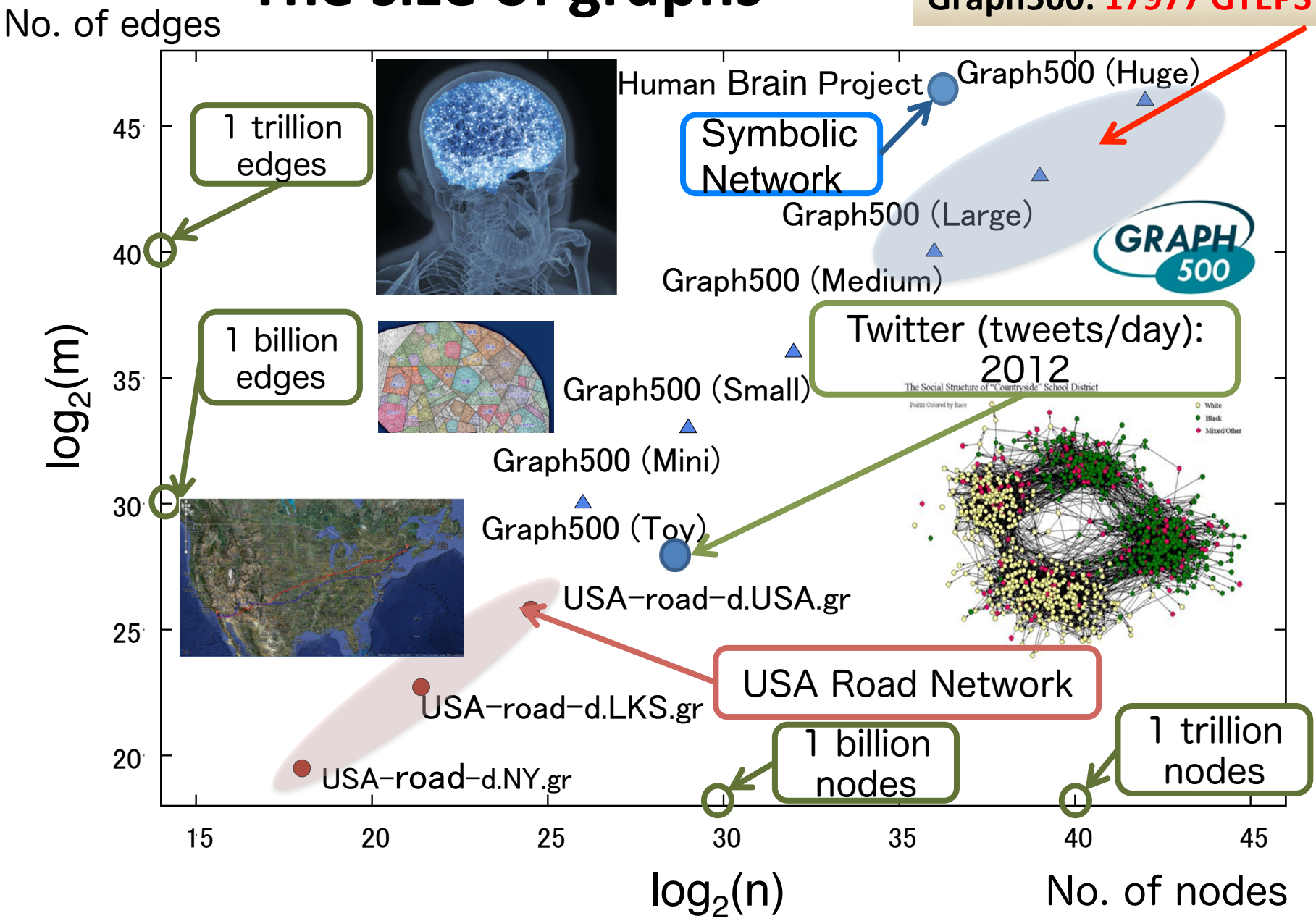


Cyber-security

15 billion log entries / day

The size of graphs

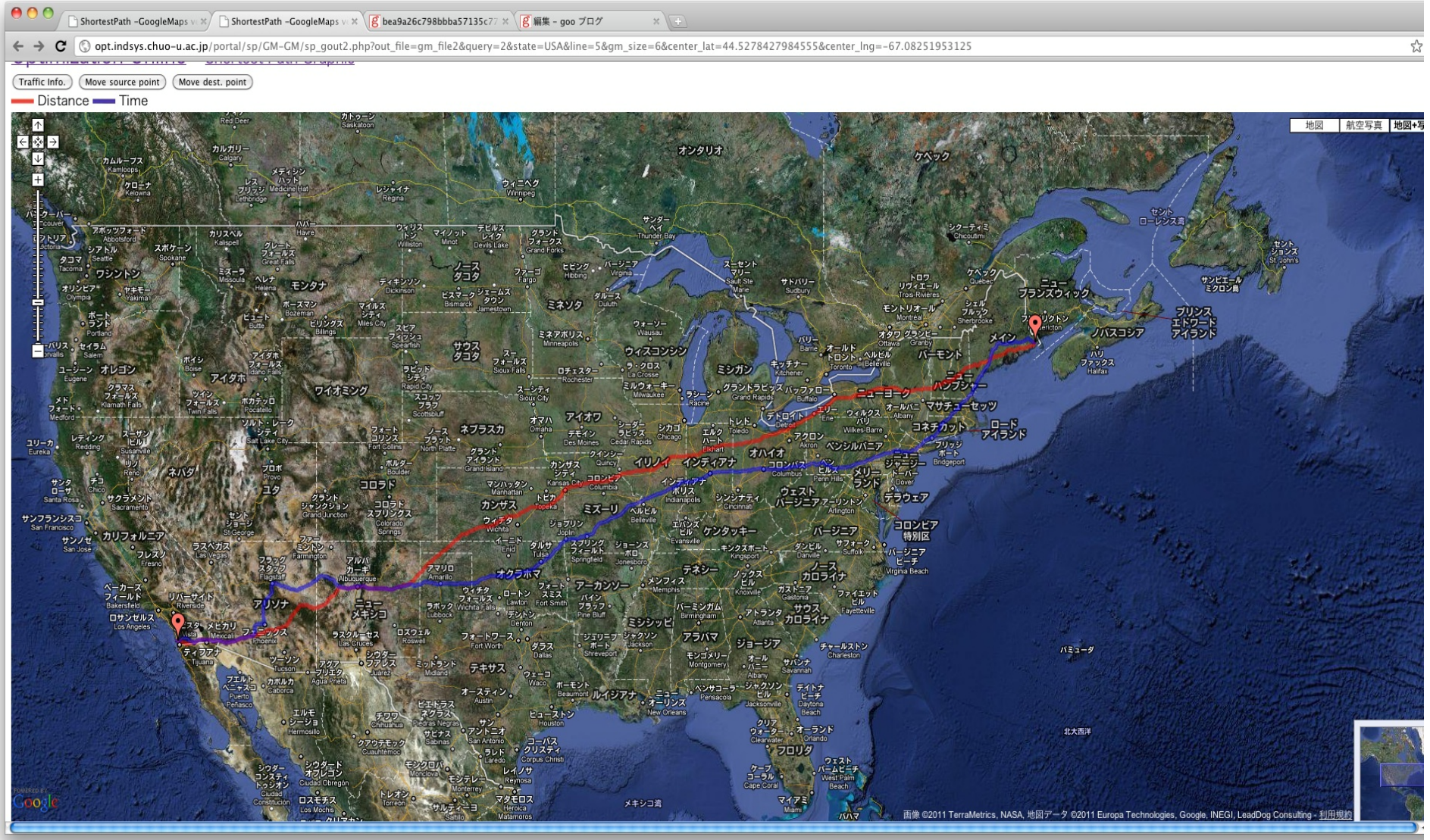
K computer: 65536nodes
Graph500: 17977 GTEPS



USA road network : **24 million vertices & 58 million edges**

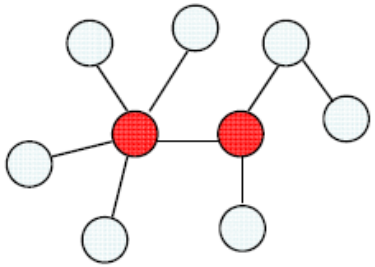
A shortest path from San Diego to Augusta

(**Red** : distance、**Blue** : time) : **http://opt.imi.kyushu-u.ac.jp/**



Centrality : vertex measures its relative importance within a graph

$$BC(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}$$



- σ_{st} : Number of shortest paths between vertices s and *t*
- $\sigma_{st}(v)$: Number of shortest paths between vertices s and *t* passing through *v*

$$C_C(v) = \frac{1}{\sum_{t \in V} d_G(v,t)}$$ *closeness centrality* (Sabidussi, 1966)

$$C_G(v) = \frac{1}{\max_{t \in V} d_G(v,t)}$$ *graph centrality* (Hage and Harary, 1995)

$$C_S(v) = \sum_{s \neq v \neq t \in V} \sigma_{st}(v)$$ *stress centrality* (Shimbel, 1953)

$$C_B(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}$$ *betweenness centrality* (Freeman, 1977; Anthonisse, 1971)

Betweenness centrality (BC)

- **Definition**

$$C_B(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

σ_{st} : # of (s,t)-shortest paths

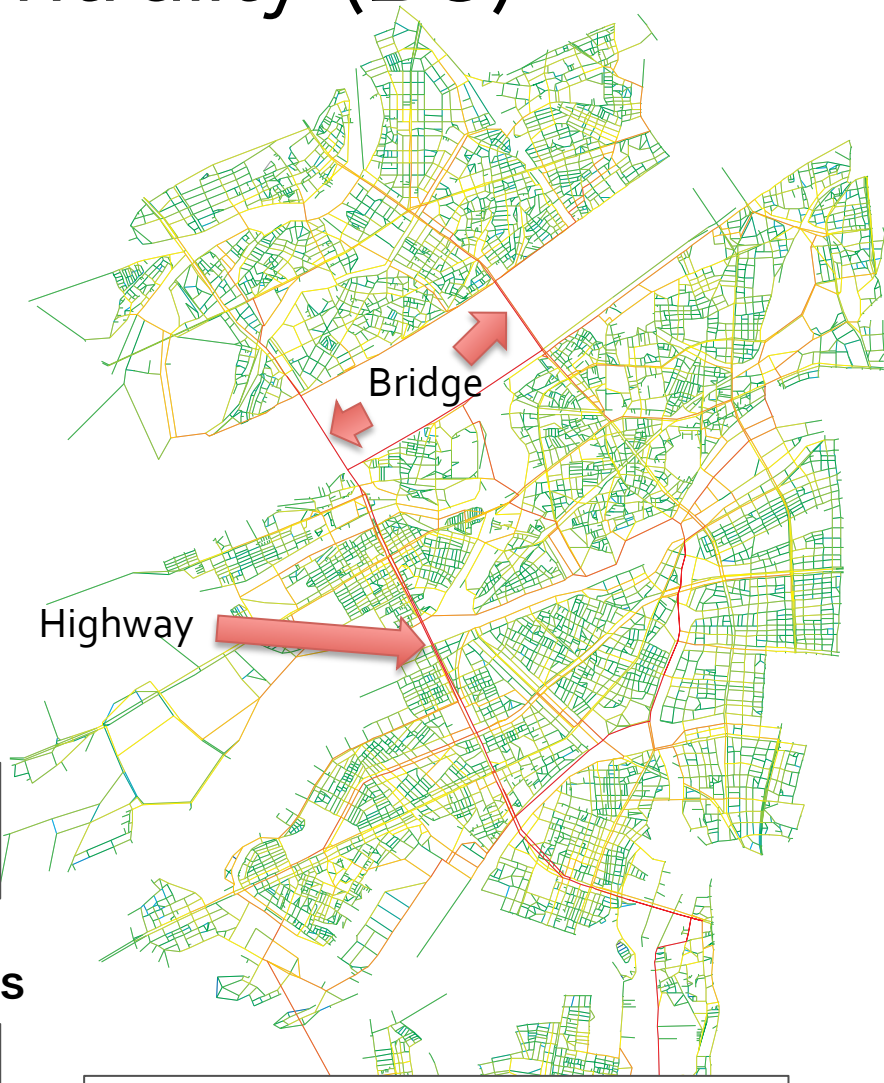
$\sigma_{st}(v)$: # of (s,t)-shortest paths passing throw v

- **BC measures important vertices and edges without coordinates**

High score vertex/edge = Important place
c.g.) Highway, Bridge

- **BC requires the all-to-all shortest paths**

- BFS => one-to-all
- <#vertices> times BFS => all-to-all



Osaka road network
13,076 vertices and 40,528 edges

=> **13,076** times BFS computations

Fukuoka road network

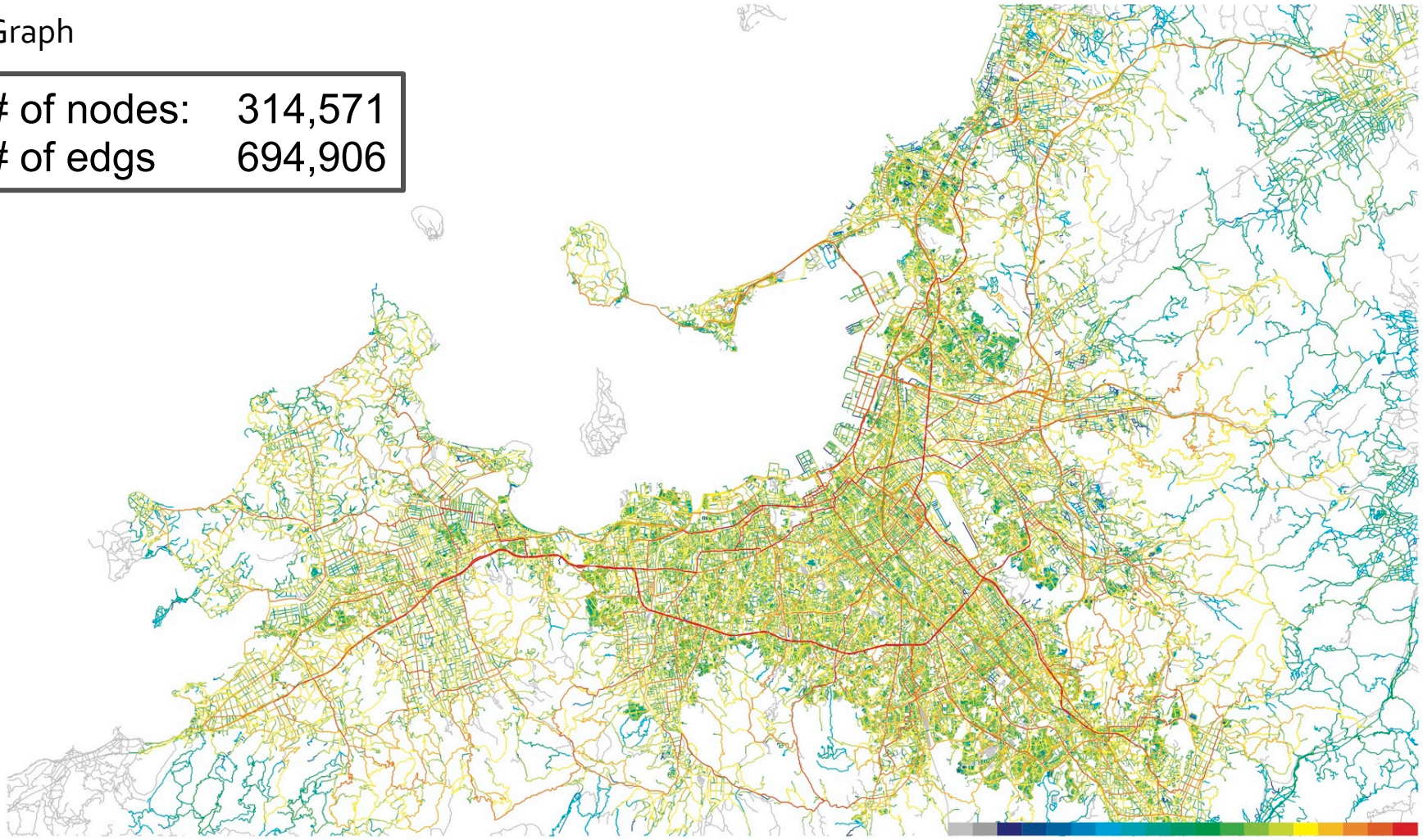
Betweenness centrality

Computation time
2m 30s (180 CPU cores)

HP ProLiant m710
Server cartridge

Graph

# of nodes:	314,571
# of edgs	694,906



All pairs shortest path problems for USA road network

No. of nodes = 23,947,347, No. of edges m = 58,333,344

4-way Opteron 6174 2.3 GHz (12 cores x 4), 256 GB, GCC-4.6.0

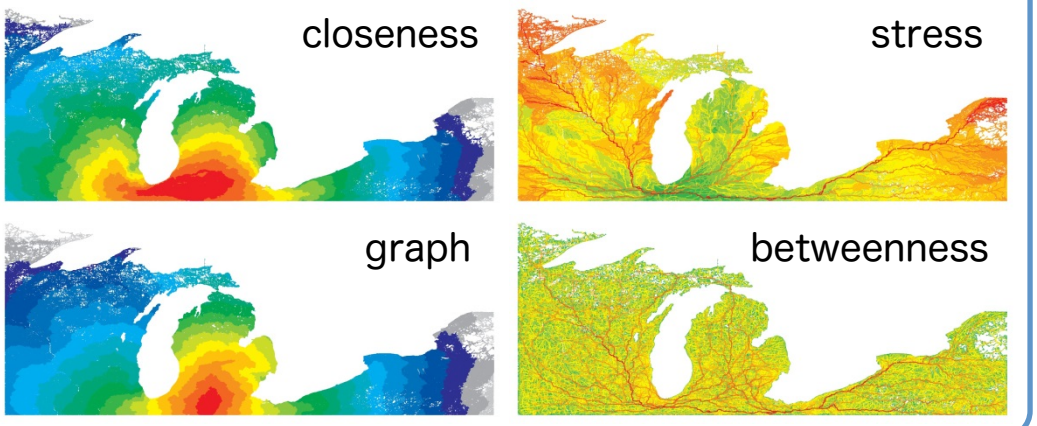
Algorithm	Comp. Time	ratio
Dijkstra's algorithm	512 Years	--
Δ-stepping	9.1 Years	60
Multi-Level Buckets	4.9 Years	110
MSLC + NUMA optimized	7.8 Days	24,000

Computation of 4 centralities for USA road network (Great Lakes)

No. of nodes = 2,758,119, No. of edges m = 6,885,658

4-way Opteron 6174 2.3 GHz (12 cores x 4), 256 GB, GCC-4.6.0

Computation time : 19.35 hours



GraphCT : Georgia Tech
20.6 Days (Only BC)

Graph 500 and Green Graph500 Benchmarks

- **New Graph Search Based Benchmarks for Ranking Supercomputers**
- BFS (Breadth First Search) from a single node on a static, undirected Kronecker graph with average vertex degree edgegactor (=16).
- No. of Nodes = 2^{SCALE}, Average degree = 16
- Performance Metrics：

  TEPS(Traversed Edges per Second) : Graph 500

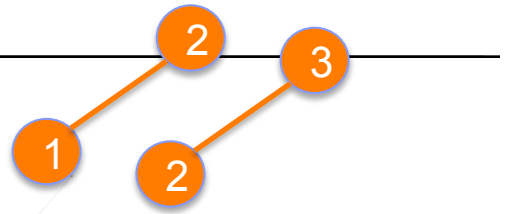
  TEPS/W (Traversed Edges per Second / Watt) : Green Graph500

Step.

1. Generate edgelist
2. Construct Graph（CSR format）
3. BFS
4. Validation

} Repeat 64 times for randomly selected source vertices

Benchmark Flow

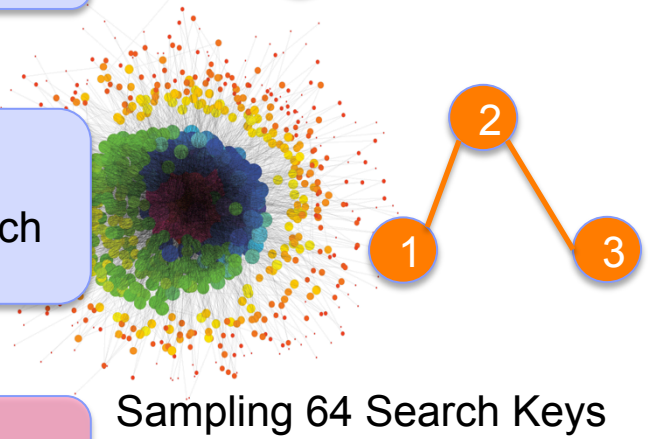


Graph (Edge List) Generation

Kernel 1

Graph Construction

(Conversion to any space-efficient format such as CSR or CSC)



Sampling 64 Search Keys

Kernel 2

Breadth First Search

(Reference or Customized Impl.)

Validation for BFS tree (5 rules)

64 times iterations

Five Business Areas --- Graph500

Graph CREST

Cybersecurity

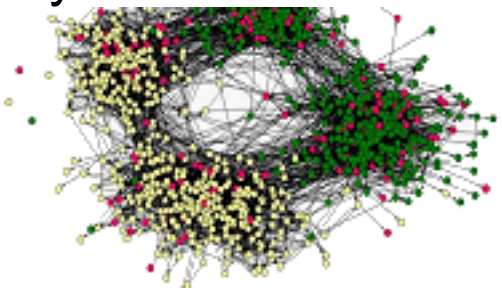
- 15 billion log entries/day
- Full data scan required

Medical Informatics

- 50 million patient records, with 20 to 200 records per patient, billions of individuals
- Entry resolution important

Social Networks

- 例） facebook twitter
- Nearly unbounded dataset size



Data Enrichment

- Easily PB of data
- 例）Maritime Domain Awareness
  - Hundreds of Millions of Transponders
  - Tens of Thousands of Cargo Ships
  - Tens of Millions of Pieces of Bulk Cargo
  - May involve additional data

Symbolic Network

- 例) the Human Brain
- 25 billion neurons
- 7,000+ connections per Neuron

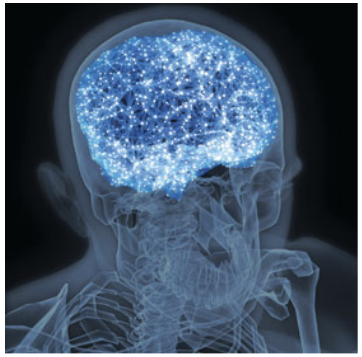
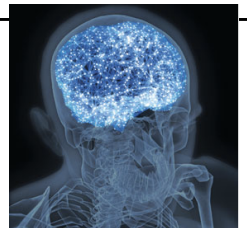


Image: Illustration by Mirko Ilic

Target networks



Human Project

Human Brain
89 B, 100 T

Graph500 (Huge)

Graph500 (Large)

Graph500 (Medium)

1trillion

Twitter2009

Graph500 (Small)

1blillion

Graph500 (Mini)

twitter-rv

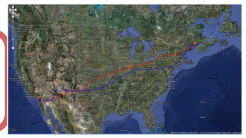
Graph500 (Toy)

soc-LiveJournal1

USA-road-d.USA.gr

cit-Patents

US road network



USA-road-d.NY.gr

1blillion

1trillion

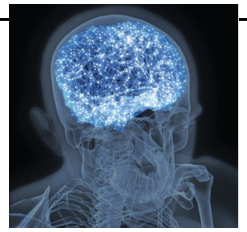
45
40
35
30
25
20

# of edges (in logscale)

15
20
25
30
35
40
45

# of vertices (in logscale)

Target networks on Smartphone



Human Project

Human Brain
89 B, 100 T

Graph500 (Huge)

Graph500 (Large)

Graph500 (Medium)

1trillion

Twitter2009

Graph500 (Small)

1blillion

Graph500 (Mini)

twitter-rv

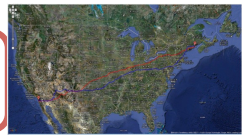
Graph500 (Toy)

soc-LiveJournal1

USA-road-d.USA.gr

cit-Patents

US road network



USA-road-d.NY.gr

1blillion

1trillion

Smartphone

Graph500 (SCALE20)
・Smartphone (4 cores)

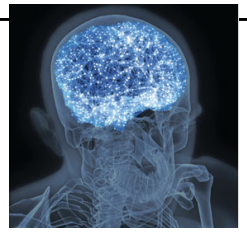
45
40
35
30
25
20

# of edges (in logscale)

20
25
30
35
40
45

# of vertices (in logscale)

Target networks on Single-server



Human Project

Human Brain
89 B, 100 T

Graph500 (Huge)

Graph500 (Large)

Graph500 (Medium)

1trillion

Twitter2009

Graph500 (Small)

1blillion

Graph500 (Mini)

twitter-rv

Graph500 (Toy)

Single server

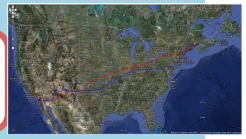
Graph500 (SCALE29)
・4-way Intel Xeon (64 cores)

soc-LiveJournal1

USA-road-d.USA.gr

cit-Patents

US road network



USA-road-d.NY.gr

1blillion

1trillion

Smartphone

Graph500 (SCALE20)
・Smartphone (4 cores)

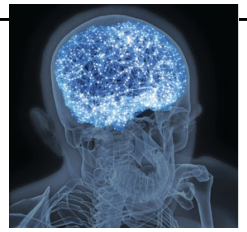
45
40
35
30
25
20

20
25
30
35
40
45

# of edges (in logscale)

# of vertices (in logscale)

Target networks on UV2000



Human Project

Human Brain
89 B, 100 T

Graph500 (Huge)

Graph500 (Large)

Graph500 (Medium)

1trillion

UV 2000

Graph500 (SCALE32)
・UV2000 (1rack, 640 cores)

Twitter2009

Graph500 (Small)

1blillion

Graph500 (Mini)

twitter-rv

Graph500 (Toy)

Single server

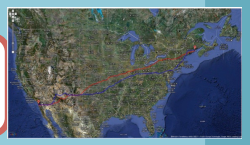
Graph500 (SCALE29)
・4-way Intel Xeon (64 cores)

soc-LiveJournal1

USA-road-d.USA.gr

cit-Patents

US road network



USA-road-d.NY.gr

1blillion

1trillion

Smartphone

Graph500 (SCALE20)
・Smartphone (4 cores)

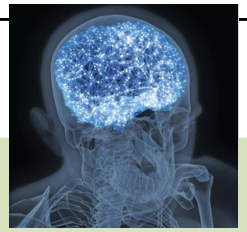
45
40
35
30
25
20

# of edges (in logscale)

20
25
30
35
40
45

# of vertices (in logscale)

Target networks on Supercomputer



Human Project

Human Brain
89 B, 100 T

Graph500 (Huge)

Graph500 (Large)

1trillion

UV 2000

Graph500 (SCALE32)
・UV2000 (1rack, 640 cores)

Graph500 (Medium)

K and Sequoia

Graph500 (SCALE40)
・BlueGene/Q (64K nodes)
・K computer (64K nodes)

Twitter2009

Graph500 (Small)

1blillion

Graph500 (Mini)

twitter-rv

Graph500 (Toy)

Single server

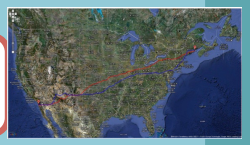
Graph500 (SCALE29)
・4-way Intel Xeon (64 cores)

soc-LiveJournal1

USA-road-d.USA.gr

cit-Patents

US road network



USA-road-d.NY.gr

1blillion

1trillion

Smartphone

Graph500 (SCALE20)
・Smartphone (4 cores)

45
40
35
30
25
20

# of edges (in logscale)

20 25 30 35 40 45

# of vertices (in logscale)

Our achievements : Graph500

GTEPS (in logscale)

16384
4096
1024
256
64
16
4
1

1st Nov. 2010
2nd June 2011
3rd Nov. 2011
4th June 2012
5th Nov. 2012
6th June 2013
7th Nov. 2013
8th June 2014

Top 1
K-computer
TSUBAME 2.5 (2.0)
FX-10
SGI UV2000
TSUBAME-KFC
4-way Xeon server

7
18
253
3541
15363
15363
15363

#4
#4
#4
5524
5524
5524
17977
×3.25
K computer
#1

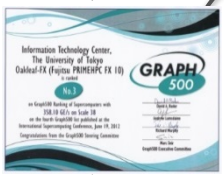
TSUBAME 2.5
1280
CPU only
993
1003
609
1003
462
462
462

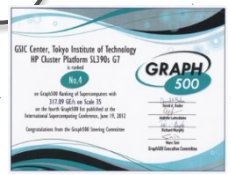
FX10
358
#3
317
#4
GPU

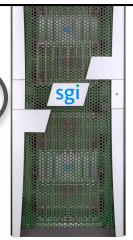
100
#3
CPU only

TSUBAME-KFC
131.4
SGI UV2000
44.0
104.3
45.7
31.6

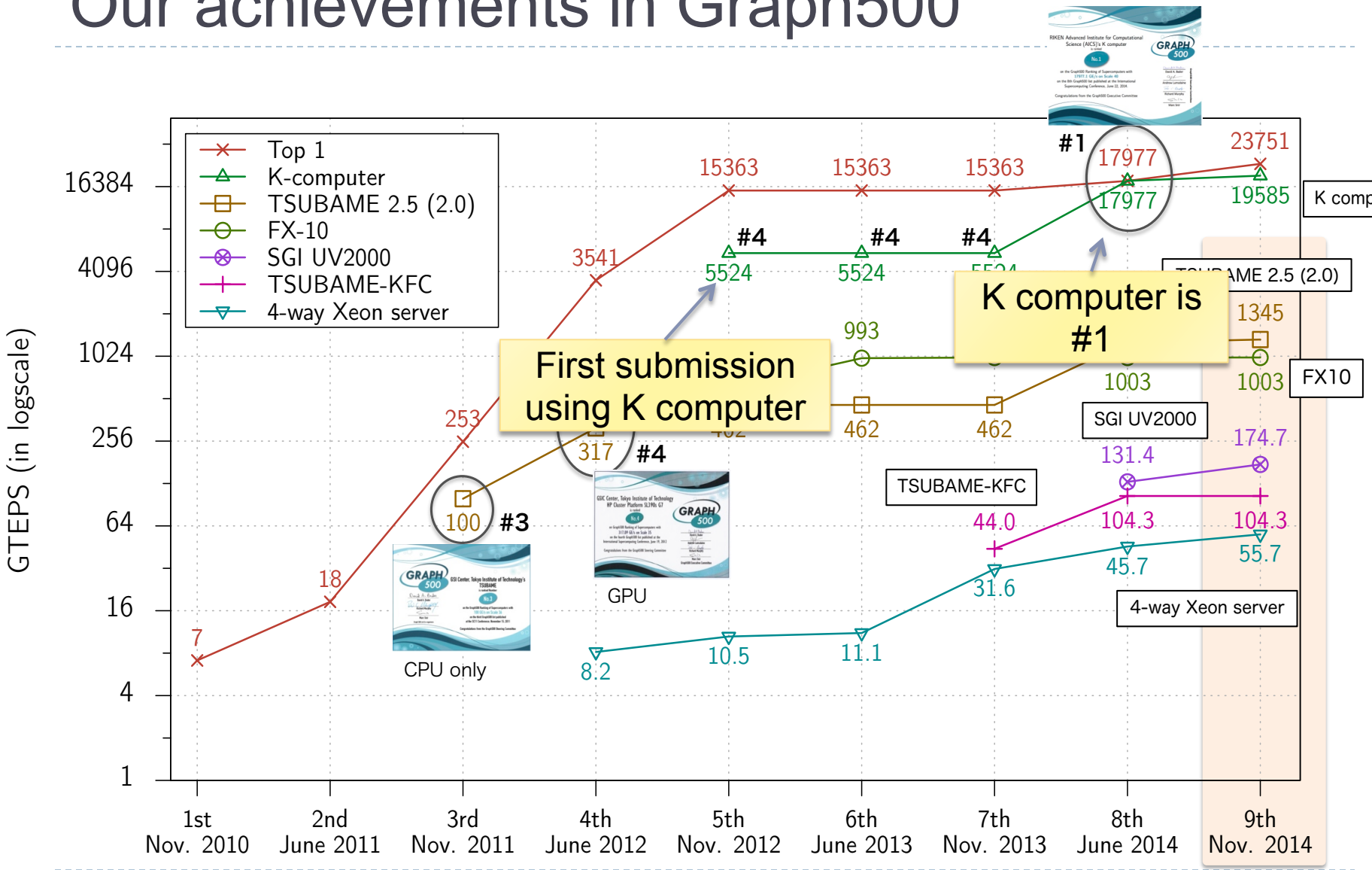
4-way Xeon server
8.2
10.5
11.1







Our achievements in Graph500



RIKEN Advanced Institute for Computational Science (AICS)'s K computer

is ranked

No.1

on the Graph500 Ranking of Supercomputers with

17977.1 GE/s on Scale 40

on the 8th Graph500 list published at the International Supercomputing Conference, June 22, 2014.

Congratulations from the Graph500 Executive Committee

GRAPH 500

David A. Bader

Andrew Lumsdaine

Richard Murphy

Marc Snir

Graph500 Executive Committee

Green Graph 500

Kyushu's University
GraphCREST-SandybridgeEP-2.4GHz

is ranked

No.1

in the **Big Data** category of the Green Graph 500
Ranking of Supercomputers with

59.12 MTEPS/W on Scale 30

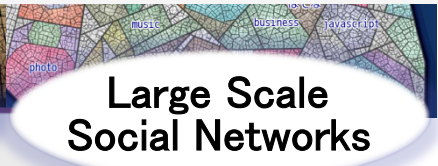
on the third Green Graph 500 list published at the
International Supercomputing Conference, June 23, 2014.

Congratulations from the Green Graph 500 Chair

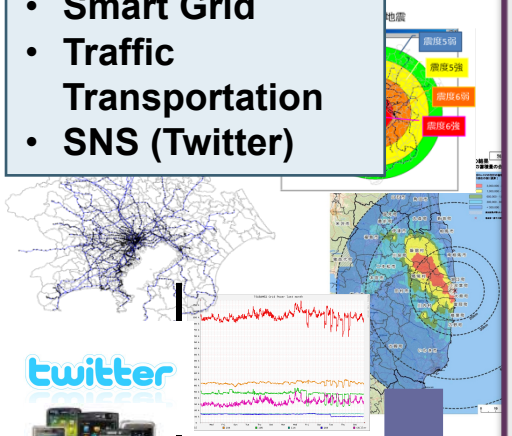
Torsten Hoefler

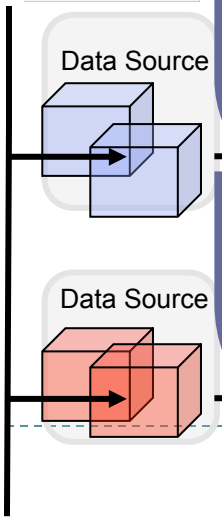
GreenGraph500 Chair

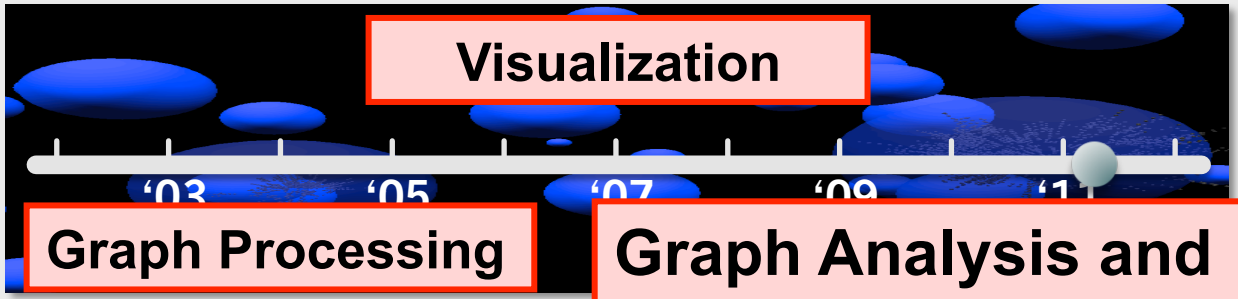
Extremely Large-scale Graph Analysis System



Large Sensor
- **Monitoring Data**
- **Smart Grid**
- **Traffic Transportation**
- **SNS (Twitter)**







Graph Processing

Graph Analysis and Optimization Library

Centrality | **Shortest Path** | **Connected Component**

Indexing | **Clustering** | **Page Rank** | **Mathematical Optimization**

Streaming Processing System

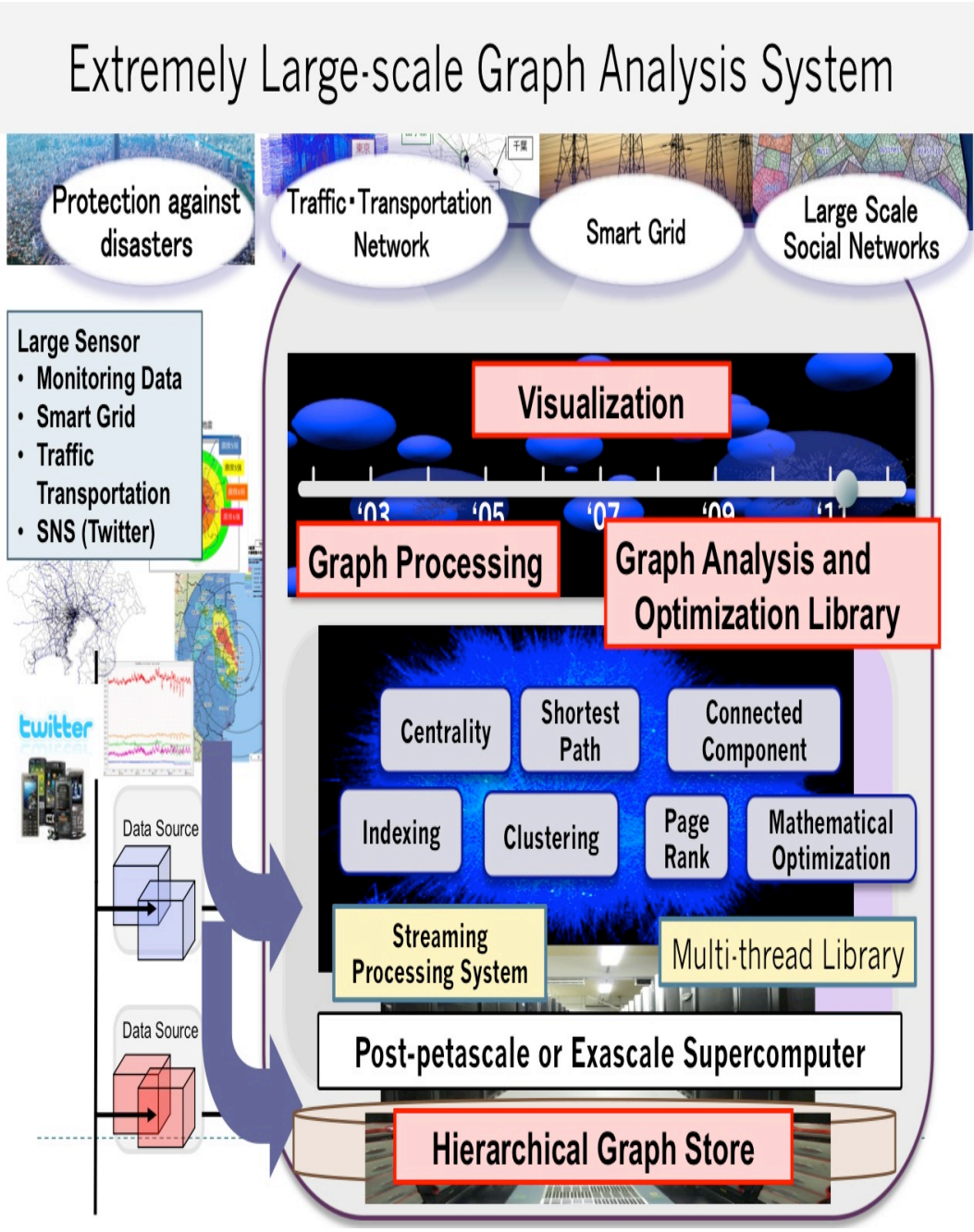
Multi-thread Library

Post-petascale or Exascale Supercomputer

Hierarchical Graph Store

Software stacks for an extremely large-scale graph analysis system

- **Hierarchal Graph Store:**
- Utilizing emerging NVM devices as extended semi-external memory volumes for processing extremely large-scale graphs that exceed the DRAM capacity of the compute nodes
- Design highly efficient and scalable data offloading techniques, PGAS-based I/O abstraction schemes, and optimized I/O interfaces to NVMs.
- **Graph Analysis and Optimization Library:**
- Perform graph analysis and search algorithms, such as the BFS kernel for Graph500, on multiple CPUs and GPUs. Implementations, including communication-avoiding algorithms and techniques for overlapping computation and communication, are needed for these libraries.
- Finally, we can make a BFS tree from an arbitrary node and find a shortest path between two arbitrary nods on extremely large-scale graphs with tens of trillions of nodes and hundreds of trillions of edges.
- **Graph Processing and Visualization:**
- We aim to perform an interactive operation for large-scale graphs with hundreds of million of nodes and tens of billion of edges.



Software Collections in GraphCREST

High-Performance General Solver for Large-scale Optimization Problems

SDPARA is a parallel implementation on multiple CPUs and GPUs for solving extremely large-scale **Semidefinite programming problems**. SDPARA can also perform parallel Cholesky factorization using thousands of GPUs and techniques to overlap computation and communication. SDPARA also achieved **1.713 PFlops** in double precision for large-scale Cholesky factorization using **4,080 GPUs on TSUBAME 2.5 supercomputer.**

`http://www.graphcrest.jp/eng/`

ScaleGraph Library

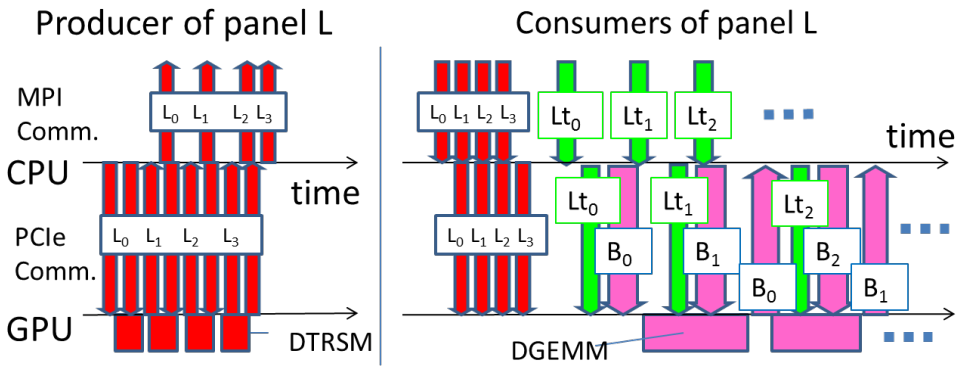
Highly Scalable Large Scale Graph Analytics Library beyond the scale of billions of vertices and edges on Distributed Systems

- Based on our extended X10
  - X10 is a new parallel distributed programming language.
- Fully utilizing MPI collective communication
- Native support for hybrid (MPI and multi-threading) parallelism

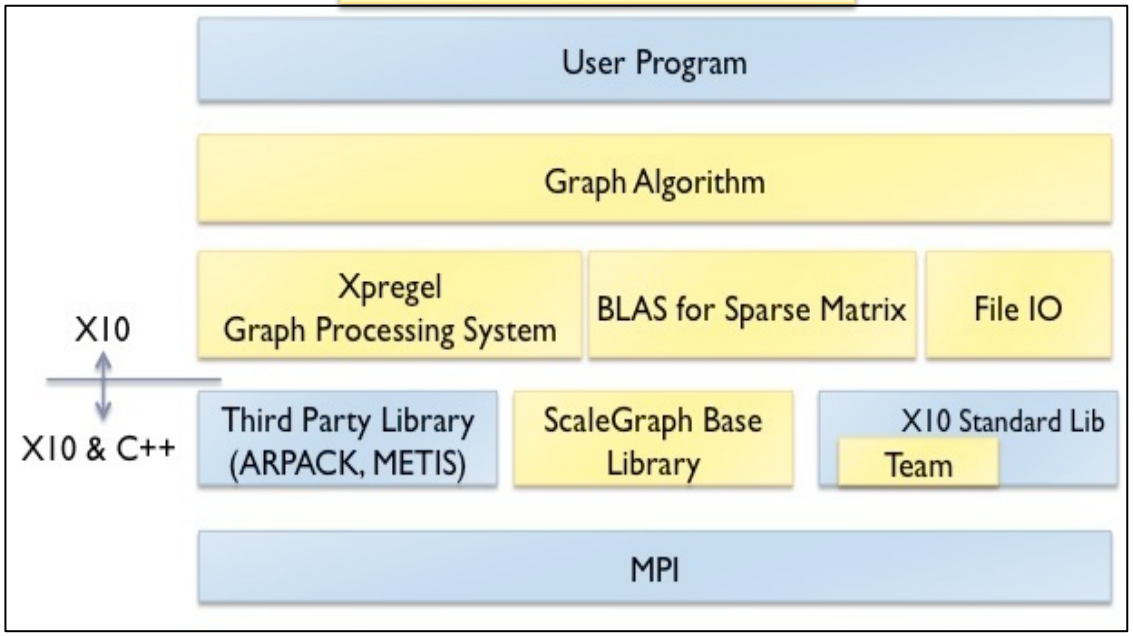
`http://www.scalegraph.org/`

Parallel Algorithm of Cholesky Factorization

GPU computation, PCI-e communication, and MPI communication are overlapped



Software stack

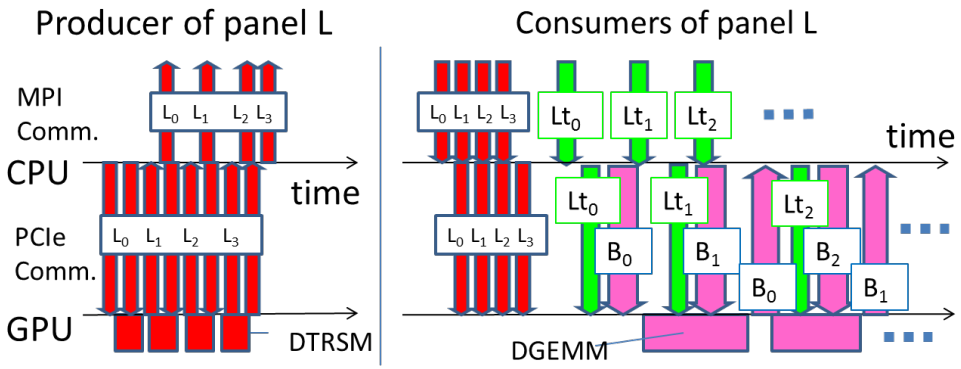


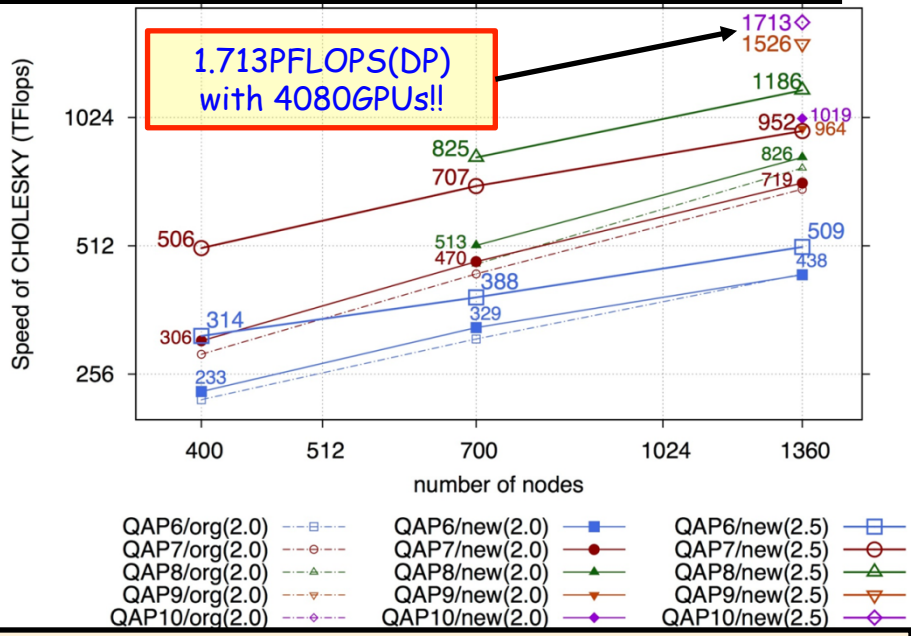
High-Performance General Solver for Extremely Large-scale Semidefinite Programming Problems

1. Mathematical Programming : one of the most important mathematical programming
2. Many Applications : combinatorial optimization, control theory, structural optimization, quantum chemistry, sensor network location, data mining, etc.

Parallel Algorithm of Cholesky Factorization

GPU computation, PCI-e communication, and
MPI communication are overlapped

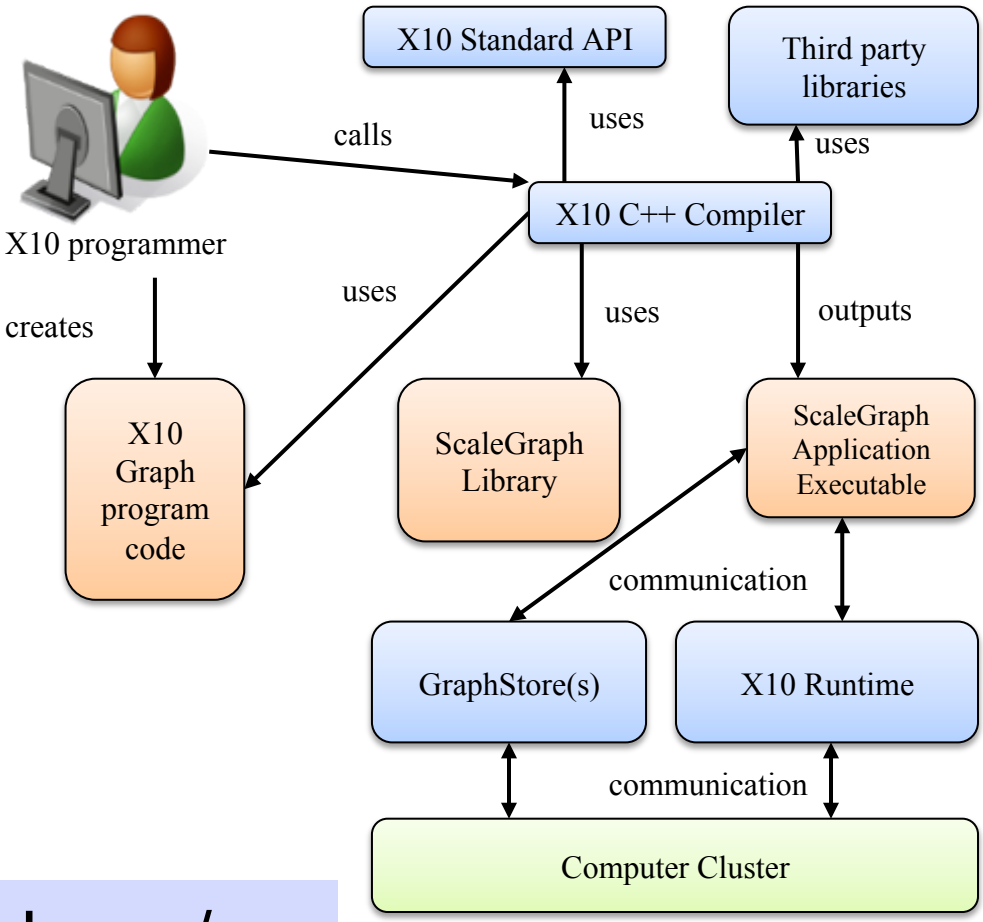




- SDPARA is a parallel implementation of the interior-point method for Semidefinite Programming Parallel computation for two major bottlenecks
  - ELEMENTS ⇒ Computation of Schur complement matrix (SCM)
  - CHOLESKY ⇒ Cholesky factorization of Schur complement matrix (SCM)
- SDPARA could attain high scalability using 16,320 CPU cores on the TSUBAME 2.5 supercomputer and some techniques of processor affinity and memory interleaving when the computation of SCM (ELEMENTS) constituted a bottleneck.
- With 4,080 NVIDIA GPUs on the TSUBAME 2.0 & 2.5 supercomputer, our implementation achieved 1.019 PFlops(TSUBAME 2.0) & 1.713PFlops(TSUBAME 2.5) in double precision for a large-scale problem (CHOLESKY) with over two million constraints.

ScaleGraph : Large-Scale Graph Analytics Library

- **Aim** - Create an open source **X10-based Large Scale Graph Analytics Library** beyond the scale of billions of vertices and edges.
- **Objectives**
  - To define concrete abstractions for Massive Graph Processing
  - To investigate use of X10 (I.e., PGAS languages) for massive graph processing
  - **To support significant amount of graph algorithms (E.g., structural properties, clustering, community detection, etc.)**
  - To create well defined interfaces to Graph Stores
  - To evaluate performance of each measurement algorithms and applicability of ScaleGraph using real/synthetic graphs in HPC environments.



URL: http://www.scalegraph.org/

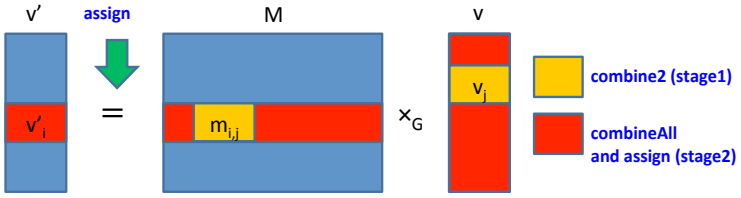
Large-scale graph processing for GPU-based supercomputers

- **HAMAR**
  - **Data parallel processing software framework (incl. MapReduce) for large-scale supercomputers w/ many-core accelerators (GPUs) and local NVM devices**
    - Weak-scaling over 1000 GPUs
  - Abstraction for deepening memory hierarchy w/ automatic out-of-core memory management
    - Device memory on GPUs, DRAM, Flash devices, etc.

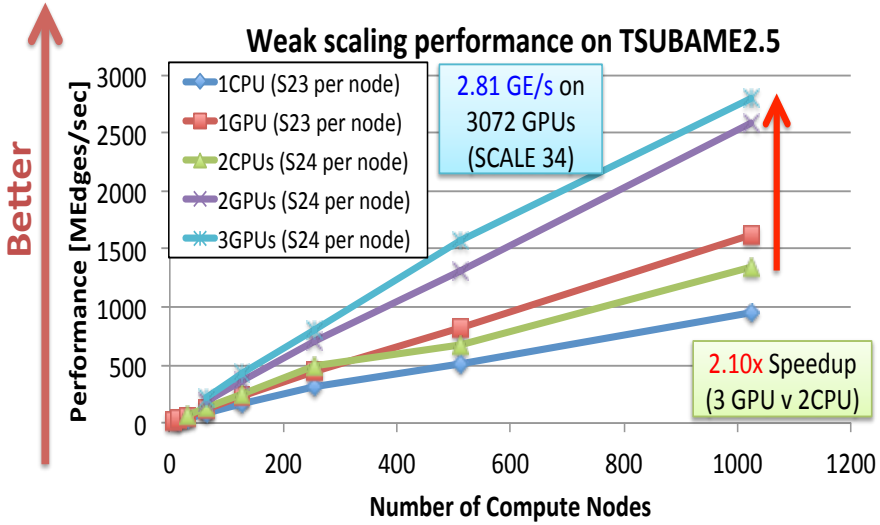
- MapReduce-based Graph Processing using HAMAR
  - **PageRank based on GIM-V (Generalized Iterated Matrix Vector Multiplication) using multi-node multi-GPUs with out-of-core memory management**
  - **Overlapping computation and CPU-GPU communication**

GIM-V: Generalized Iterative Matrix-Vector multiplication*1

- Easy description of various graph algorithms by implementing combine2, combineAll, assign functions
- PageRank, Random Walk Restart, Connected Component
  - $v' = M \times_G v$ where
    $v'_i = \text{assign}(v_j, \text{combineAll}_j(\{x_j \mid j = 1..n, x_j = \text{combine2}(m_{i,j}, v_j)\}))$ (i = 1..n)
  - Iterative 2 phases MapReduce operations



*1 : Kang, U. et al, "PEGASUS: A Peta-Scale Graph Mining System- Implementation and Observations", IEEE INTERNATIONAL CONFERENCE ON DATA MINING 2009



Advanced Computing and Optimization Infrastructure for Extremely Large-Scale Graphs on Post Peta-Scale Supercomputers

- **JST**(Japan Science and Technology Agency) **CREST**(Core Research for Evolutionaly Science and Technology) **Project** (Oct, 2011 ～ March, 2017)
- **3 groups, over 60 members**
  1. Fujisawa-G (Kyushu University) : Large-scale Mathematical Optimization
  2. Suzumura-G (University College Dublin, Ireland) : Large-scale Graph Processing
  3. Sato-G (Tokyo Institute of Technology) : Hierarchical Graph Store System
- **Innovative Algorithms and implementations**
  - Optimization, Searching, Clustering, Network flow, etc.
- Extreme Big Graph Data for emerging applications
  - **2^{30} ~ 2^{42} nodes** and **2^{40} ~ 2^{46} edges**
  - **Over 1M threads** are required for real-time analysis
- Many applications on post peta-scale supercomputers
  - Analyzing massive cyber security and social networks
  - Optimizing smart grid networks
  - Health care and medical science
  - Understanding complex life system