Advanced Research and Technology Symposium 2018

The Future of Advanced (Secure) Computing

DataSToRM: Data Science and Technology Research Environment

Mr. Vitaliy Gleyzer

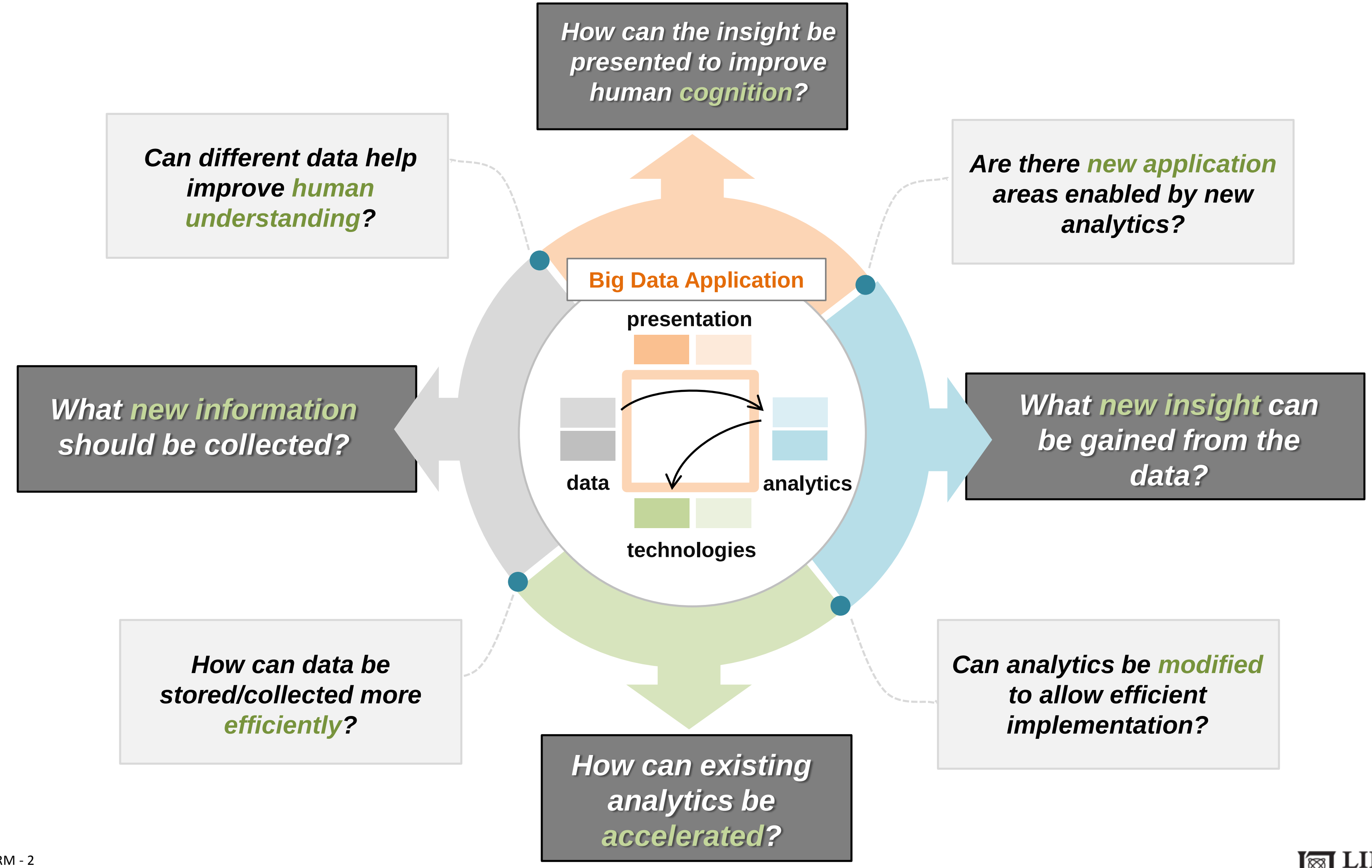
MIT Lincoln Laboratory

5 March 2018

This material is based upon work supported by the Assistant Secretary of Defense for Research and Engineering under Air Force Contract No. FA8721-05-C-0002 and/or FA8702-15-D-0001. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Assistant Secretary of Defense for Research and Engineering.

LINCOLN LABORATORY
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

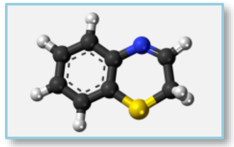
Advancing the State of Big Data Analytics: *Raw Data to Insight*



Large-Scale Graph Applications Today

Applications

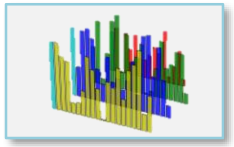
Drug Discovery



Personalized Healthcare

Fraud Detection

Recommender Systems



Functional Brain Mapping

Cyber Data Analysis

01110010111001111101011
10001100101010010101011
10101101101010110110111
11101011HACKED11110110
00010101001000010111111
10010101010101010101000
11111001111111011001000

Development Environment

Hardware Platform

Lincoln Laboratory Supercomputing Center

Graph Analysis Frameworks and Databases

neo4j, Project Pegasus, D4M, MATLAB, Pregel, Apache Giraph, GraphX, JanusGraph, GraphLab

...

Algorithms

- PageRank
- Centrality
- Walktrap
- InfoMap
- K-Truss
- BFS
- MST
- ...

Visualization Toolkits

Gephi, Tulip, KeyLines, graphistry

...

Diverse, quickly evolving ecosystem

Advancing the State of Big Data Analytics: *Challenges*

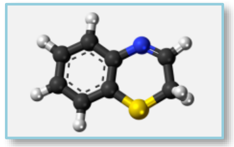
- Technology moves quickly
  - New algorithms and analytic techniques
  - New storage solutions
  - New processing technologies
  - New database technologies and frameworks
  - New applications
- New framework adoption is a serious investment
- How to leverage new technologies?
- How to enable co-design opportunities?
- How to integrate disparate communities to enable co-design?

Keeping up with big data technology is challenging

Example Standardization Efforts

Applications

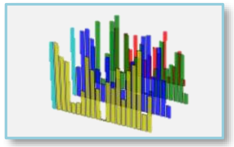
Drug Discovery



Personalized Healthcare

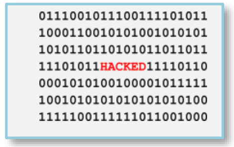
Fraud Detection

Recommender Systems



Functional Brain Mapping

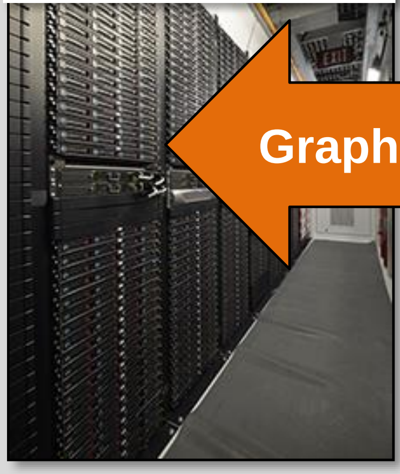
Cyber Data Analysis



Development Environment

Hardware Platform

Lincoln Laboratory Supercomputing Center



Graph Analysis Frameworks and Databases

neo4j

Project Pegasus

D4M

MATLAB

Pregel

JanusGraph

GraphLab

...

Algorithms

PageRank

Centrality

W trap

uss

BFS

MST

...

Visualization Toolkits

Gephi

Tulip

KeyLines

graphistry

...

GraphBLAS

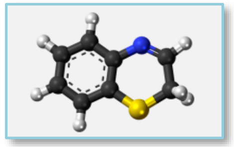
Apache Gremlin

Different communities are attempting to unify and standardize interface and languages in the ecosystem

Example Standardization Efforts

Applications

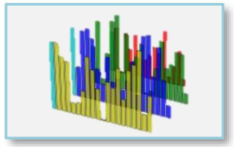
Drug Discovery



Personalized Healthcare

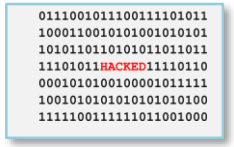
Fraud Detection

Recommender Systems



Functional Brain Mapping

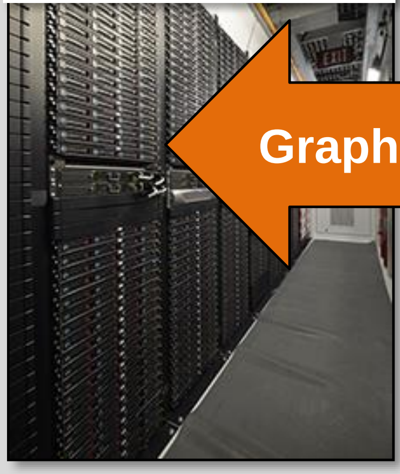
Cyber Data Analysis



Devel

Hardware Platform

Lincoln Laboratory Supercomputing Center



Graph Analysis Dat

JanusGrap

GraphBLAS

Backend interface standardization enables hardware innovation!

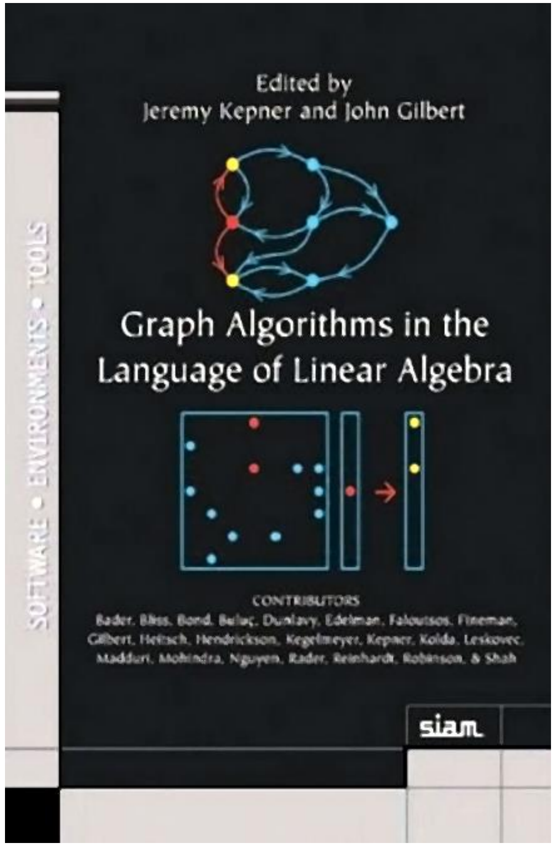
Graph Processor

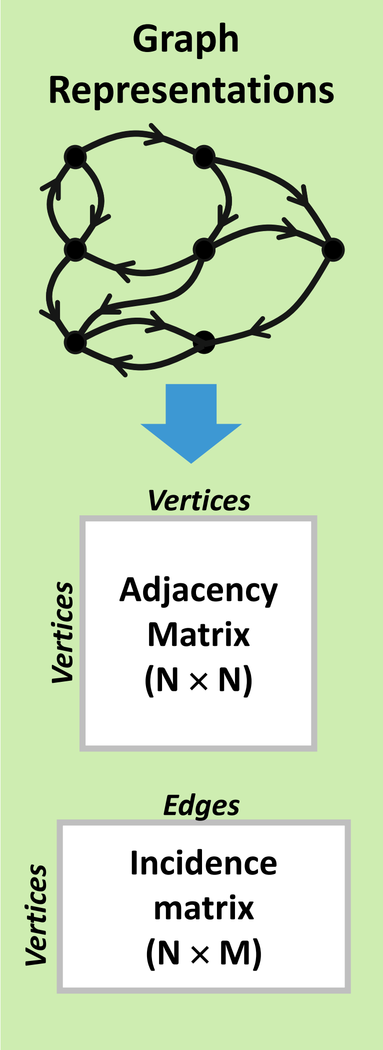
- Parallel in-memory database
- 100×–1000× graph edge traversal speed
- Simple linear algebra API

Different communities are attempting to unify and standardize interface and languages in the ecosystem

Unifying Principles for Big Data Graphs

- Graphs capture relationship information between entities
  - Molecular forces
  - Social interactions
  - Semantic concepts
  - Vehicle tracks
- Graphs can be fully expressed in the language of linear algebra
  - Represented as sparse matrices
  - Enable mathematic foundation for data analysis
  - Leverage existing linear algebra techniques and methods
  - Define a small set of well-defined mathematical operations





LINCOLN LABORATORY
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

DataSToRM: Data Science and Technology Research Environment

Applications	Threat Detection	Sentiment Analysis	Recommender Engine	…
Graph Analysis Kernels	Community Detection	Classification	Centrality Analysis	…
API	GraphBLAS (Semi-ring Linear Algebra API)			
Hardware	Graph Processor		AMD, NVIDIA, intel, XILINX	

Composed of analytics

Implemented on top of a standard API

Enables hardware diversity

Hardware acceleration of a small number of well-defined mathematical operations enable an extensive analytic ecosystem

GraphBLAS Overview

- Five key operations

  $A = \mathbb{S}^{N\times M}(i,j,v)$ $(i,j,v) = A$ $C = A \oplus B$ $C = A \otimes C$ $C = A\,B = A \oplus.\otimes B$
- Can be used to build 12 GraphBLAS standard functions

  buildMatrix, extractTuples, Transpose, mXm, mXv, vXm, extract, assign, eWiseAdd, eWiseMult, apply, reduce
- Can be used to build a variety of graph utility functions

  Tril(), Triu(), Degreed Filtered BFS, …
- Can be used to build a variety of graph algorithms

  K-Truss, Jaccard Coefficient, Non-Negative Matrix Factorization, …
- That work on a wide range of graphs

  Hyper, multi-directed, multi-weighted, multi-partite, multi-edge

Unifying interface for backend graph processing

Lincoln Laboratory Technologies Targeting Large-Scale Graph Analytics

Dynamic Distributed Dimensional Data Model (D4M)

Data analysis framework based on associative array algebra

- **Concise language for complex graph analytics**
- **Mathematical closure**
- **Linear Algebra underpinning**

Graph Algorithms

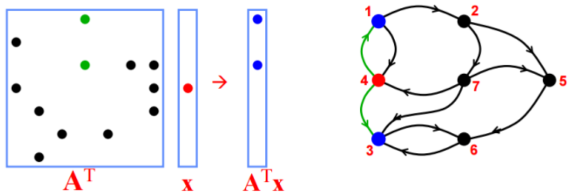
D4M

GraphBLAS

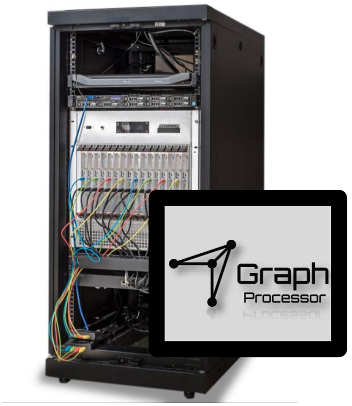
Graph Processor | **LLSC**

GraphBLAS

Standard API for graph analytics using Sparse Linear Algebra primitives



- **Simple hardware agnostic API**



Graph Processor

Novel graph processing architecture

- **Scalable graph processing hardware architecture**
- **Unprecedented performance**
- **Native linear algebra instruction set**

Lincoln Laboratory Super Computing Center (LLSC)

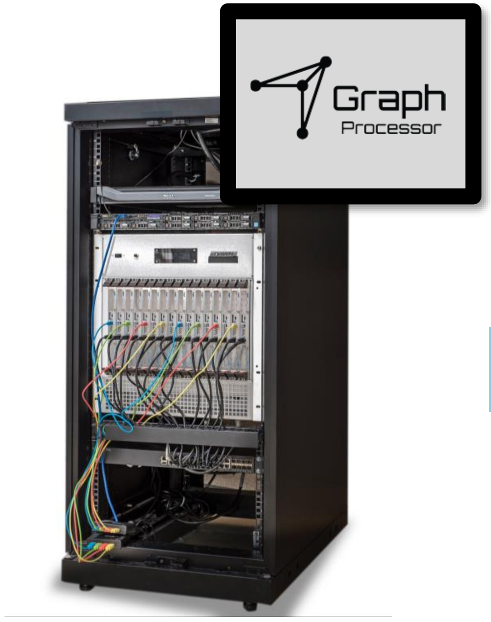
State-of-the-art super computing environment

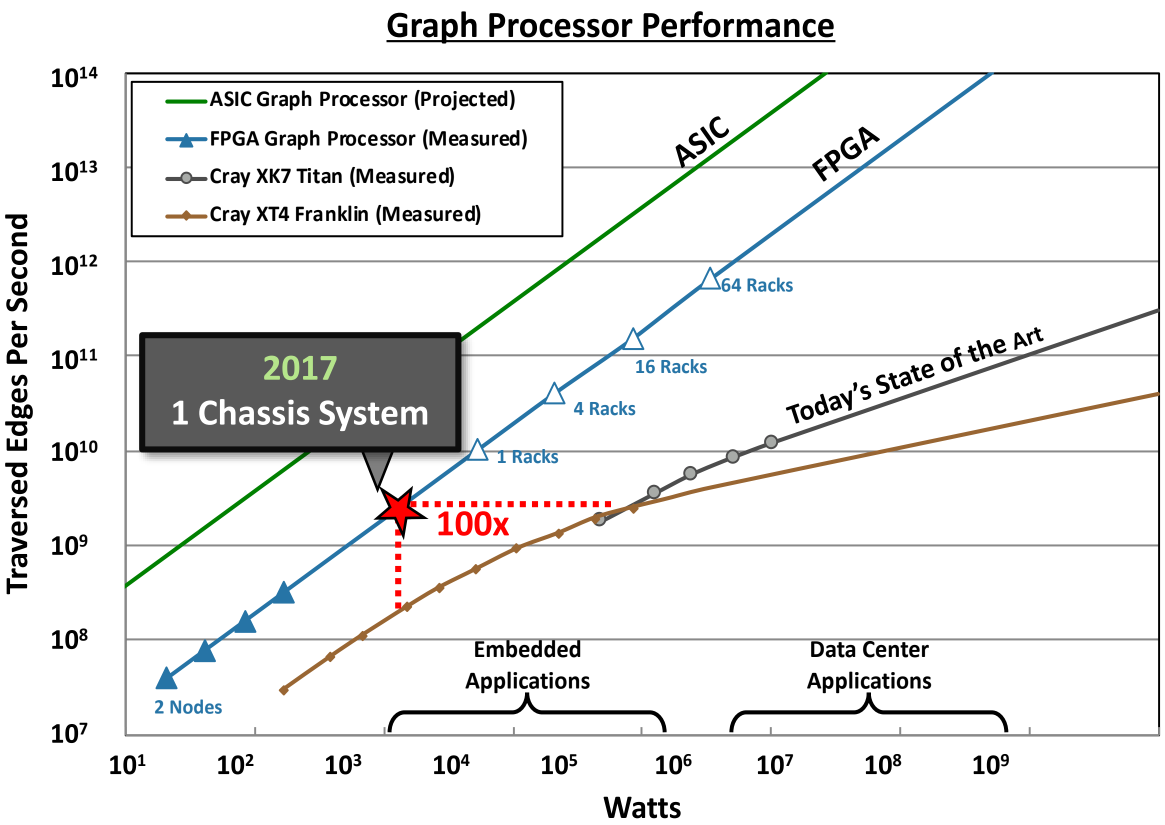
- **Heterogeneous processing capabilities**
- **Ideal technology integration environment**

slurm workload manager

lustre File System

Graph Processor Matrix Multiply Performance





Highly efficient graph processing technology that is 100s to 1000s of times more efficient compared to traditional architectures

Collaboration Opportunities

- Developing graph algorithms in the language of linear algebra
  - Community detections, subgraph isomorphism, subgraph matching, etc.
- Developing graph algorithms that can scale to datasets with billions to trillions of vertices
  - Sparsity-aware, distributed memory algorithms
- Identifying or developing new technologies to leverage the linear algebra abstraction
  - Compilers, optimizer, hardware accelerators, etc.
- Integrating GraphBLAS backend as part of popular frameworks
  - e.g., Apache TinkerPop, Neo4j, ElasticSearch, etc.