Advanced Graphical Intelligence Logical Computing Environment (AGILE)

ISC 2022 - Workshop

Dr. William Harrod | June 2, 2022

Intelligence Advanced Research Projects Activity

IARPA

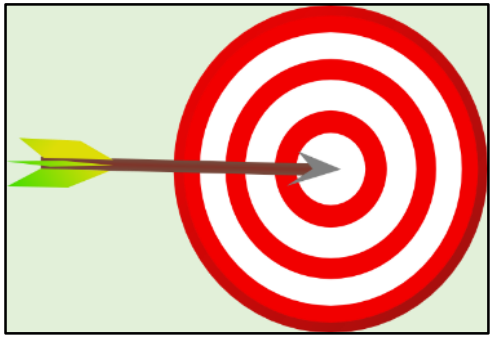
Creating Advantage through Research and Technology

AGILE Program

Open Designs

Co-design Process

Modeling & Simulation



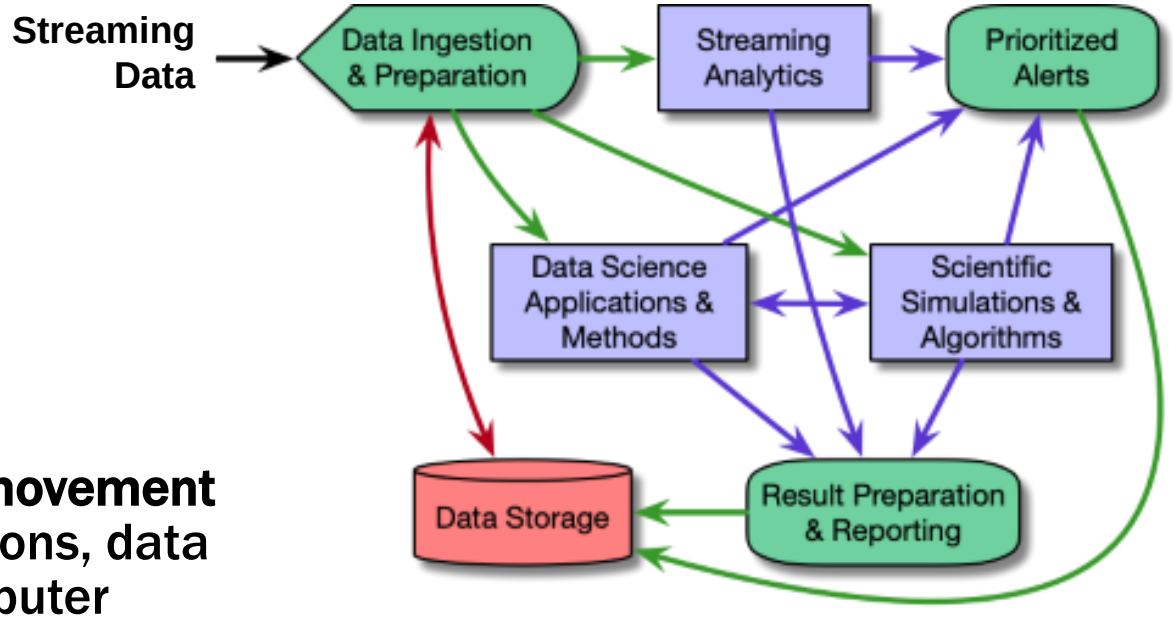
Target Metrics

Create novel computer architectures and designs that overcome the current and future data-analytics technical challenges.

The program will result in the delivery of system-level RTL designs where the performance has been evaluated by using an application modeling and simulation environment.

Develop validated designs that achieve or exceed the AGILE Program Target Metrics. These results will be validated by an independent test and evaluation team.

AGILE Technical Challenges



- Results required in **near-real-time up to hours**
- Streaming data causes **unpredictable changes** to stored data
- **Extremely fine grain data movement and parallelism**: computations, data are distributed across computer
- Data computation tasks to be performed are typically **determined by the data and streaming queries**
- Tasks have **extremely poor data locality and data reuse**
- **Many graph analytics algorithms can be recast as sparse linear algebra operations**

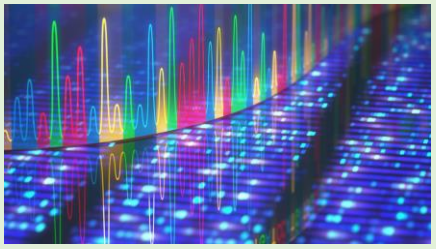
AGILE Driving Applications Domains

KNOWLEDGE GRAPHS

Groups, Relationships & Interests

PATTERN DETECTION

System and Event Patterns



SEQUENCE DATA

Identification and Clustering

NETWORK

Cyber-Physical Systems

Big Data (today)

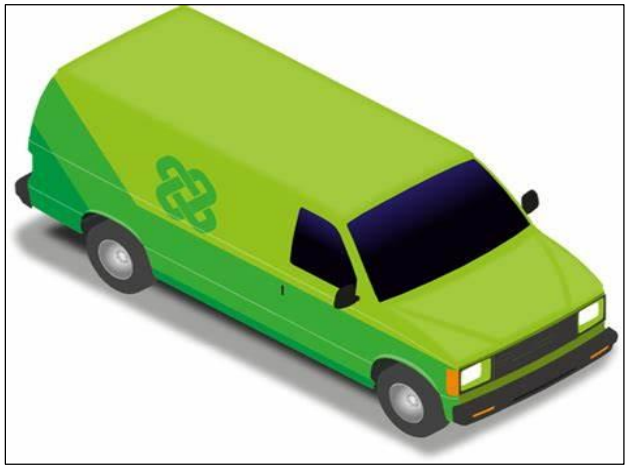
"Forensic Analysis"

Streaming Data Analytics

"Predictive Analysis"

Develop revolutionary new computers to solve critical data-intensive problems

AGILE Program Details



Program Objectives:

- Enable data analytic problems that involve 10X more data.
- Time to solution 10-100 times faster.

Research Effort:

- Develop validated designs that achieve or exceed the AGILE Program Target Metrics.
- These results will be validated by an independent test and evaluation team.

Deliverables:

- Phase 1: System-level functional model of architecture.
- Phase 2: Detailed (RTL) design for proposed AGILE system architecture.

https://www.iarpa.gov/research-programs/agile

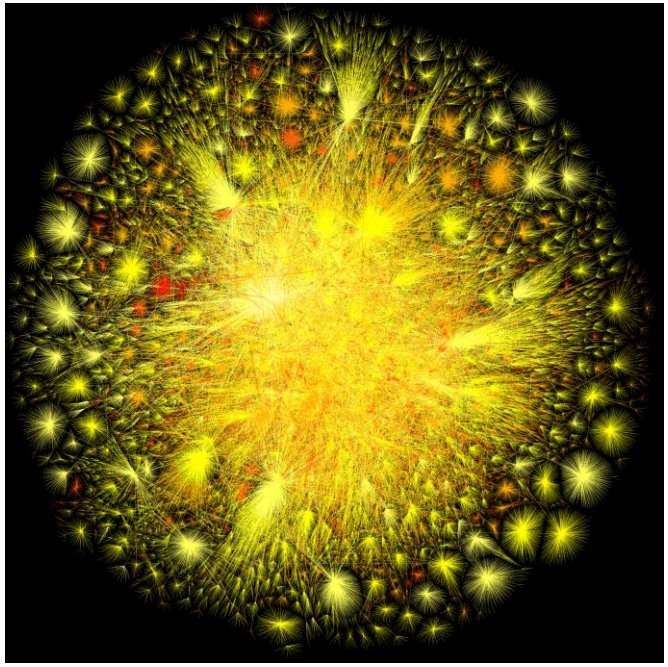
IARPA IS NO LONGER ACCEPTING PROPOSALS

Data Analytics Problem

- Entities are represented by vertices (V) with types and properties, and relationships are represented by edges (E) with types and properties.
- The graphs are typically sparse --- that is $|E| <<< |V|^2$

Graphs	Vertices	Edges
Social network	1 Billion	100 Billion
Internet	50 Billion	1 Trillion
Brain	100 Billion	100 Trillion

Technical Report NSA-RD-2013-056002v1, U.S. National Security Agency



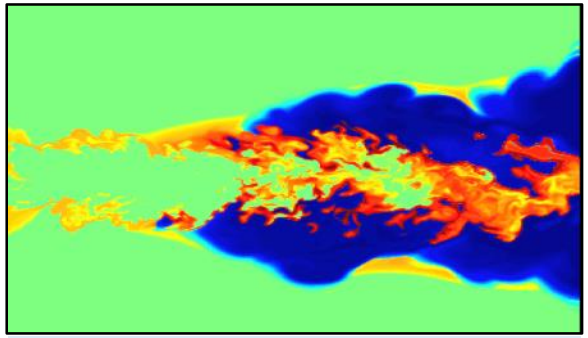
Internet Graph 2010 The Opte Project

Extracting Actionable Knowledge Methods

Graph Analytics	Machine Learning	Statistics Methods	Linear Algebra	Data Filtering

"The variety and volume of data collected (today) … far outpace the abilities of current systems to execute complex analytics … and extract meaningful insights." (Buono, D., *Computer*, August 2015)

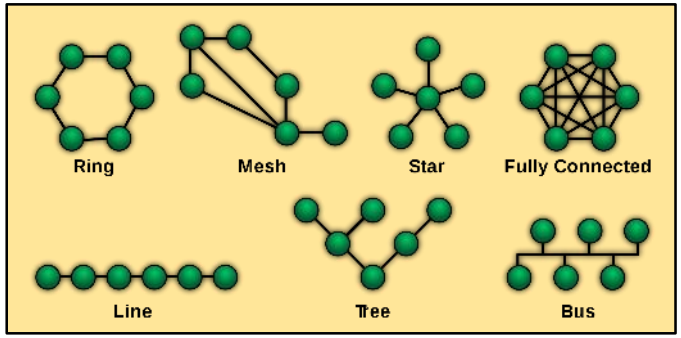
Today's Computers



Multi-Physics Simulation

Jackie Chen, SNL

Accelerators



Conventional Networks

Designed for yesterday's applications

- Multi-physics simulations

Vendors are focused on incremental improvements

- Accelerators & supporting memory components
- Focused on processing not data challenges

Computers are not computational efficient or scalable for large scale graph analytics problems

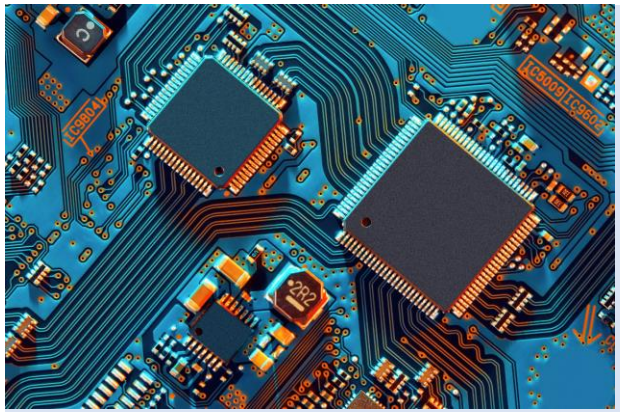
Over Provisioned Features

1. Deep hierarchical memories,
2. Large-message interconnection networks
3. Bulk, synchronous execution models

T&E Evaluation Efforts

Independent Test and Evaluation Process Based on The Following Areas

Design V&V

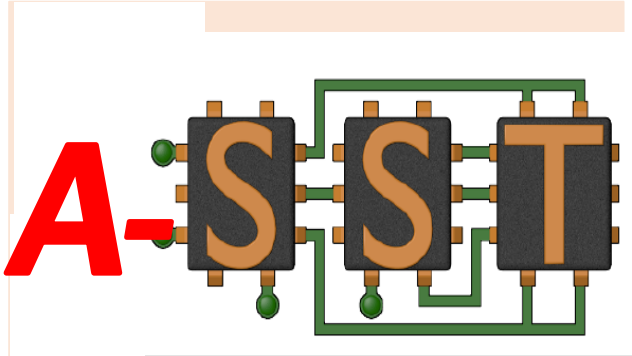


Validate Performers' Hardware & Application Test Plans

Evaluate Performers' models/designs for correctness & completeness

Validate the results generated using A-SST

ModSim



Validate Performers' models in the A-SST (Toolkit) & Firesim

Using A-SST, provides performance estimates of the Performers' models/designs

*AGILE-enhanced Structural Simulation Toolkit (Modeling and Simulation Environment)

Application Codes

Develop AGILE Workflows and kernels

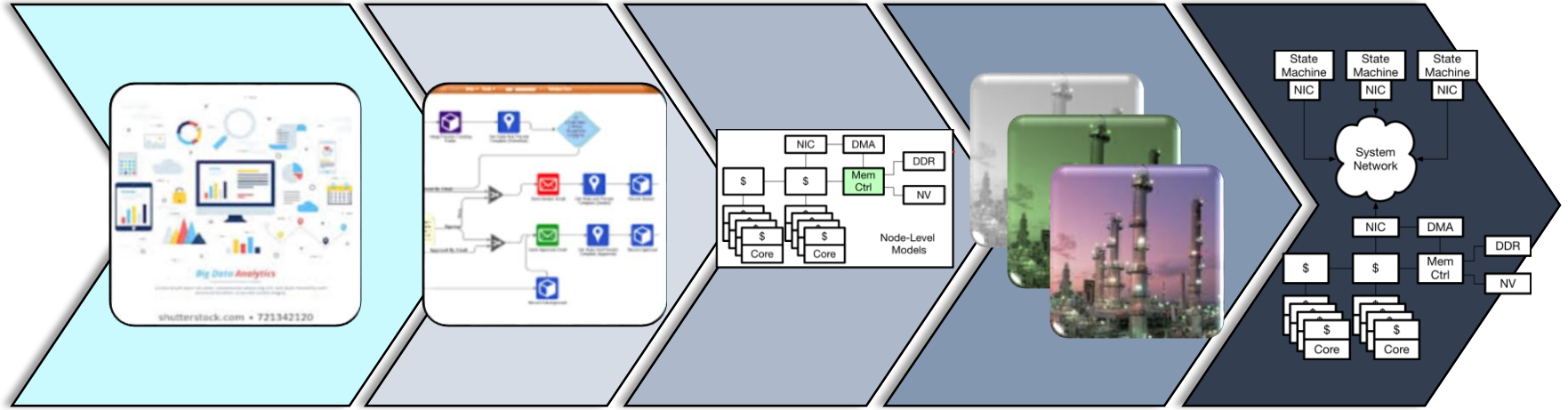
Baseline performance

Validate changes to the Performers' versions of the AGILE Workflows, kernels and benchmarks (optimized for their systems)

Multi-model Modeling & Simulation Methodology

Based on Sandia National Laboratories - Structural Simulation Toolkit (SST)

Application Representation	Workload-Defined Design Requirements	Execute High-Level Models	Refine / Optimize Designs	Execute Low-Level Designs



- Proxy of critical performance bottlenecks
- Capture workload

- Develop High-level Model
- Behavioral Model
- System – multi-node model

- Validate high-level bottlenecks
- Initial performance estimates and uncertainties

- Design low-level design experiments using initial high-level results
- Define ensembles for data-dependent workload sampling
- Develop multi-node design

- Refine performance models and model uncertainties
- Validate models against testbeds
- Improve design characteristics and performance

Workflows and Benchmarks

Objectives

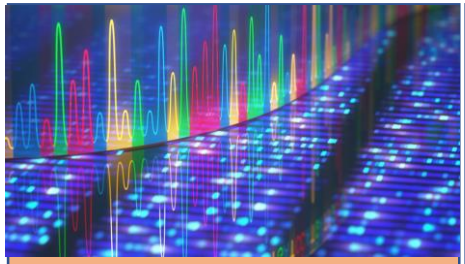
- Enable data analytic problems that involve 10X more data
- Time to solution 10 times faster
- Designs must be able to achieve the Target Metrics

- These objectives motivate the Workflow-driven R&D plans specified in the proposals.
- Performance estimates will be validated by T&E Team using A-SST (based on Sandia's SST)

Workflows

Knowledge Graphs

Pattern Detection



Sequence Data

Network

- **AGILE will provide reference implementations - Performers can modify to optimize for their design -** Performers will use in the Co-Design process.
- **Develop and investigate design that can achieve the target metrics for the Workflows**

Benchmarks

Breadth First Search	Counting Triangles	Jaccard Similarity

- Performers will utilize the Benchmark Codes in the Co-Design process.
- Develop and investigate design that can achieve the target metrics for the Benchmarks

Workflows and Kernels

- Given the heterogeneity and complexity of data analytic workloads, kernels that measure individual metrics - FLOPS, TEPS, cache misses, network bandwidth - for a single data type cannot reflect the performance and scalability of full applications
- Only end-to-end workflows can reflect the performance and scalability of real-world analytic jobs
  - Ingestion, transformation, and storage of input data can take significant time, energy, and machine resources
  - Prioritization and display of output results can be costly
- Kernels are still valuable when measuring the speeds-and-feeds of individual system components and when systems/tools are too immature to run complete workflows

AGILE APPLICATIONS

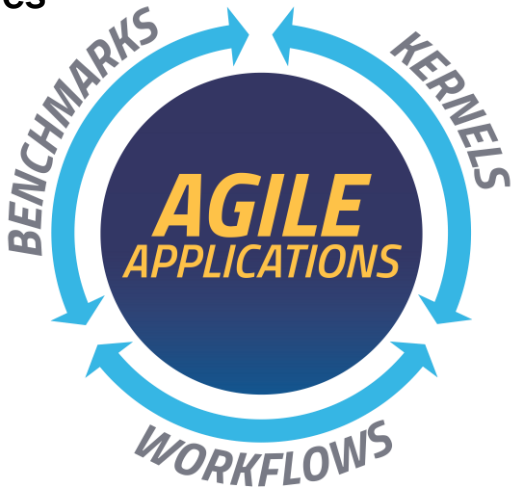
AGILE Applications

- Includes workflows, kernels and industry benchmarks
- Test programs / scripts
- Data sets or generators

Reference codes will be written using SHAD

SHAD
SCALABLE HIGH-PERFORMANCE ALGORITHMS & DATA-STRUCTURES

- Presents a shared–memory view of global memory
- STL-complaint, thread-safe, distributed data structures
- Concurrent insert/delete/modify and AMOs on all data structures
- Asynchronous data and task parallel programming constructs
- Multithreaded runtime that hides latencies (no data partitioning necessary)
- Runs on servers and clusters
- https://github.com/pnnl/SHAD



Algorithms can be substituted if they provide the same functionality

Target Metric Tables

Knowledge

ROW	Metric	Today	AGILE Target
1	Data ingestion rate	0.1 G data-elements per second	10 G data-elements per second from 3 or more sources
2	Time to learn embedding (Graph Size > 1 PB)	1,440 minutes	30 minutes
3|4	Time to classify vertices and edges	> 1,440 minutes	30 minutes
3|4	Time to predict and infer new relationship	> 1,440 minutes	30 minutes
5	Time to reason about higher-order relationships using multi-hop reasoning	1 – 2 hops (exact matches) in 30 minutes	3 – 5 hops (approximate/fuzzy matches) in 1 minute

Sequence Data

Metric	Today	AGILE Target
Size of graph	0.01 PB[4]	10 PB
Data ingestion rate	0.1 G data-elements per second from a single source, single data type	10 G data-elements per second from a three or more sources and data types
Insert/Delete/Modify rate for vertices and edges	0.01 G edits / second (batched)	10 G edits / second (continuous)
Pattern Detection per minute	Single event, linear paths, exact match	Multiple events, branches, prioritized approximate/fuzzy matching
Incremental analysis	NOT DONE	Commensurate with data rate
Time to complete multiple day / multiple location queries	NOT DONE	Completed in minutes

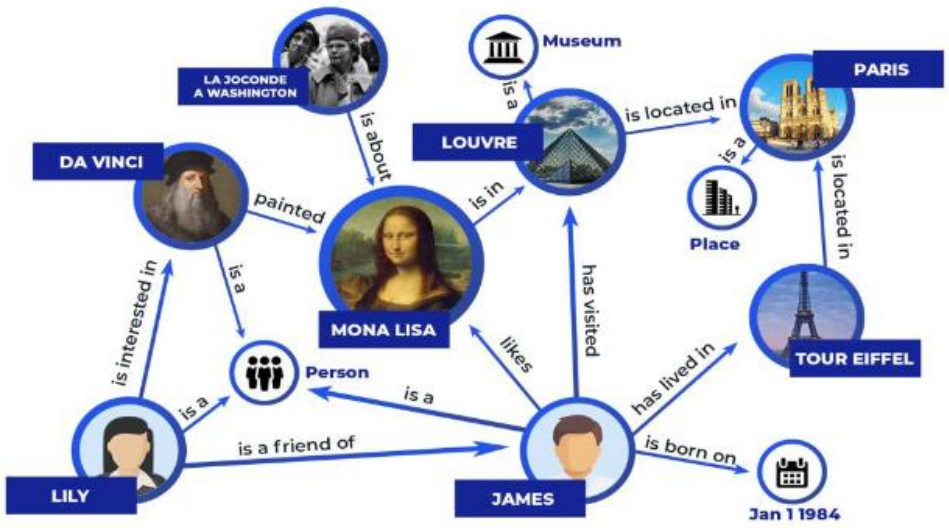
Detection

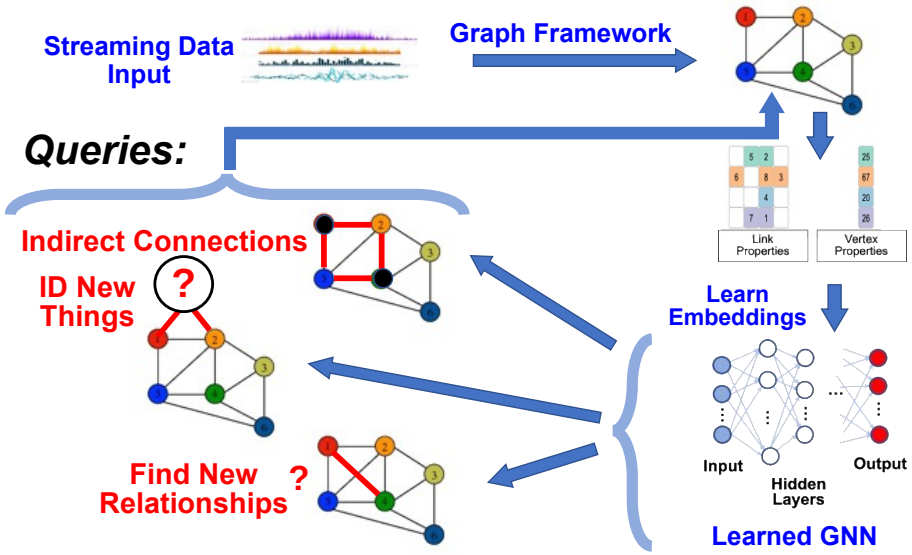
ROW	Metric	Today	AGILE Target
	Size of graph	0.01 PB[4]	10 PB
1	Data ingestion rate	0.1 G data-elements per second from a single source, single data type	10 G data-elements per second from a three or more sources and data types
2	Insert/Delete/Modify rate for vertices and edges	0.01 G edits / second (batched)	10 G edits / second (continuous)
3	Pattern Detection per minute	Single event, linear paths, exact match	Multiple events, branches, prioritized approximate/fuzzy matching
4	Incremental analysis	NOT DONE	Commensurate with data rate
5	Time to complete multiple day / multiple location queries	NOT DONE	Completed in minutes

Network

Metric	Today	AGILE Target
Construct 1 PB graph through game theoretic modeling	120 minutes	2 minutes (60x faster)
Identification of top k influential nodes (simple model)	60 minutes	1 minute (60x faster)
Identification of top k influential nodes (enhanced model)	600 minutes	30 minutes (20x faster)
Propagate labels/confidence score	120 minutes	2 minutes (60x faster)
Incremental analysis	NOT DONE	Never recomputed from scratch

Knowledge Graph





What is it:

- A semantic network of persons, places, objects, events, situations, or concepts, and the relationships among them
- Integrates multiple data sources with disparate types of entities (vertices) and relationships (edges)
- Ontologies are used to establish a logical, hierarchy of types creating a formal representation of the entities in the graph

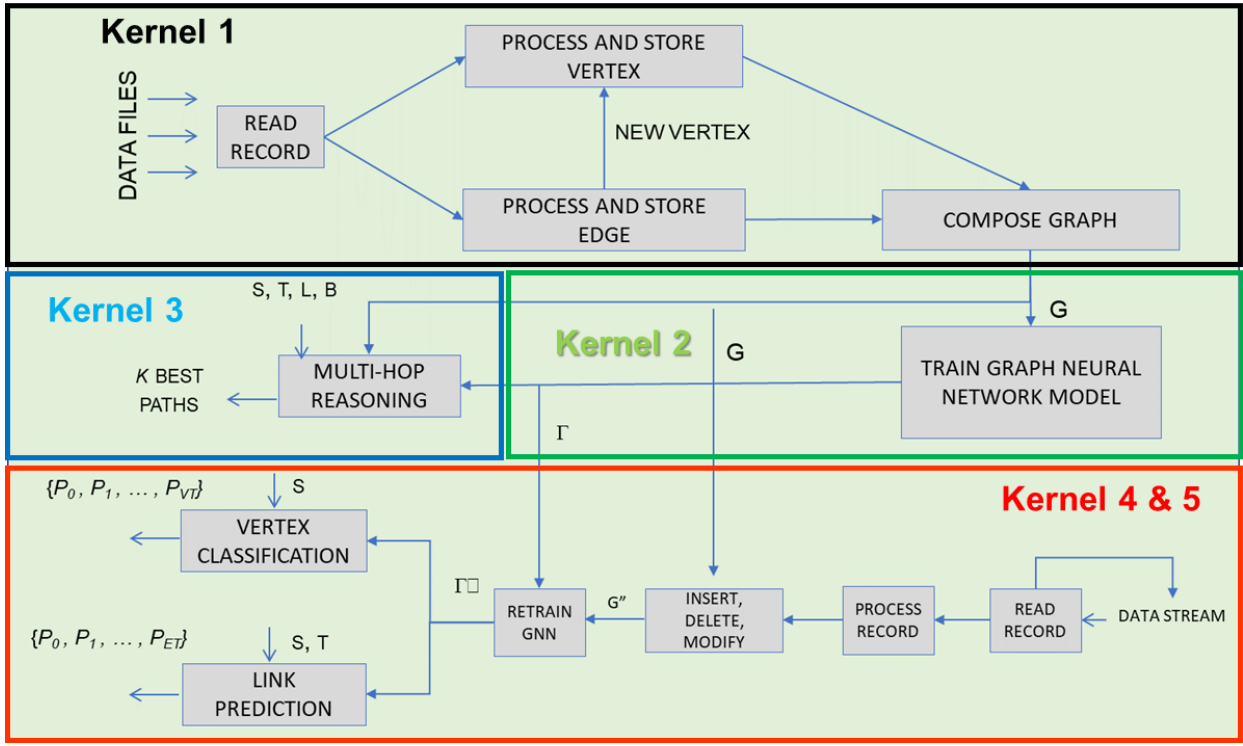
Knowledge graph use cases:

- Discover new entities, relationships & facts
- Explain the contextual reasons for a particular event
- Explain why a human expert should look at emerging event
- Answer complex questions that are beyond database queries

Workflow 1 – Knowledge Graph

- **Kernel 1:** measures streaming data ingestion rate, the time to read data records, transform the raw data, resolve vertex and edge ambiguities and build-out the common internal data structures used by downstream tasks.

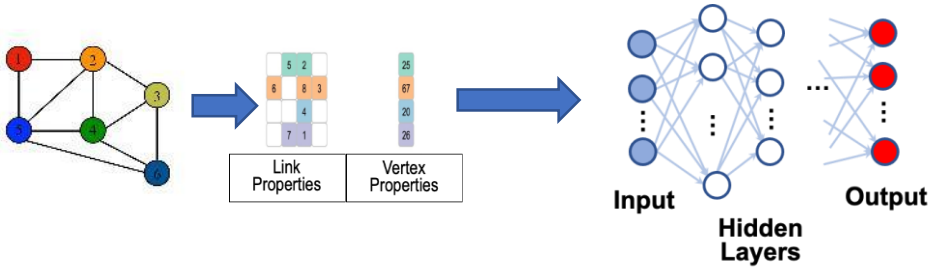
- **Multi-hop Reasoning** – *Indirect connections* given vertices *s* and *t* in G, return the "best" *k* paths from *s* to *t*
  Kernel 3
- **Vertex Classification** – *ID new things* given unlabeled *v* in G with properties (p_1, … p_n) and incident edges {e_1, … e_k}, return the type of *v*
  Kernel 4
- **Link Prediction** – *Find new relationships* given *s*, *t* in G such that edge {*s*, *t*} does not exist in G, predict the existence and edge type of {*s*, *t*}
  Kernel 5



Knowledge Graph Target Metrics

Kernel 1 – Row 1 Kernel 4 – Row 3
Kernel 2 – Row 2 Kernel 5 – Row 4
Kernel 3 – Row 5
Kernels 4, 5 involve updating model from Kernel 2

Kernel 2: Learn Representation



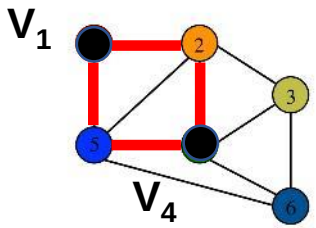
Graph

Learn Vertex, Link, Graph Embeddings

Quantitative Representation

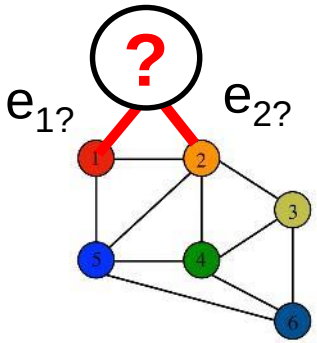
Graph Neural Network (GNN)

Kernel 3: Multi-hop Reasoning



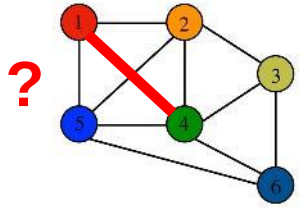
Find the k-best multi-hop paths from v_4 to v_1 with $d<L_{max}$

Kernel 4: Vertex Classification



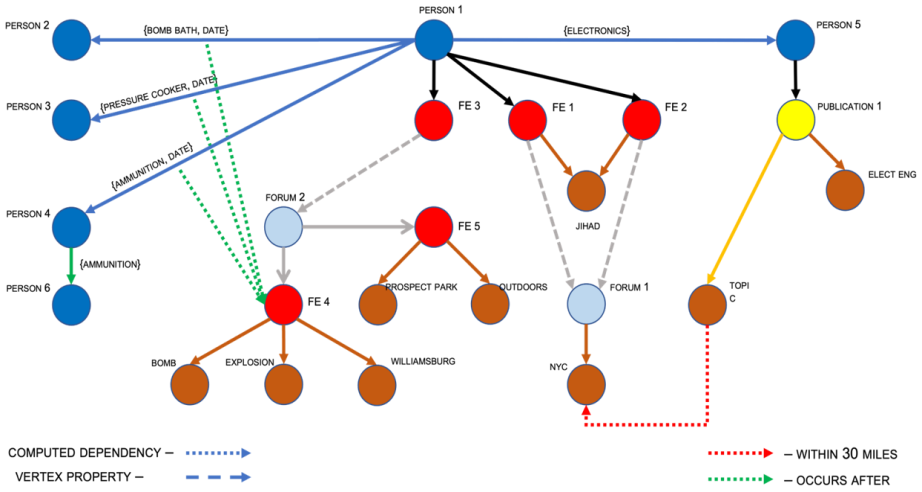
For new unlabeled vertex, $V_?$ in the labeled graph, G, with properties (π_1 , π_2, π_n) and edges ($e_{1?}$, $e_{2?}$), find the label for $V_?$

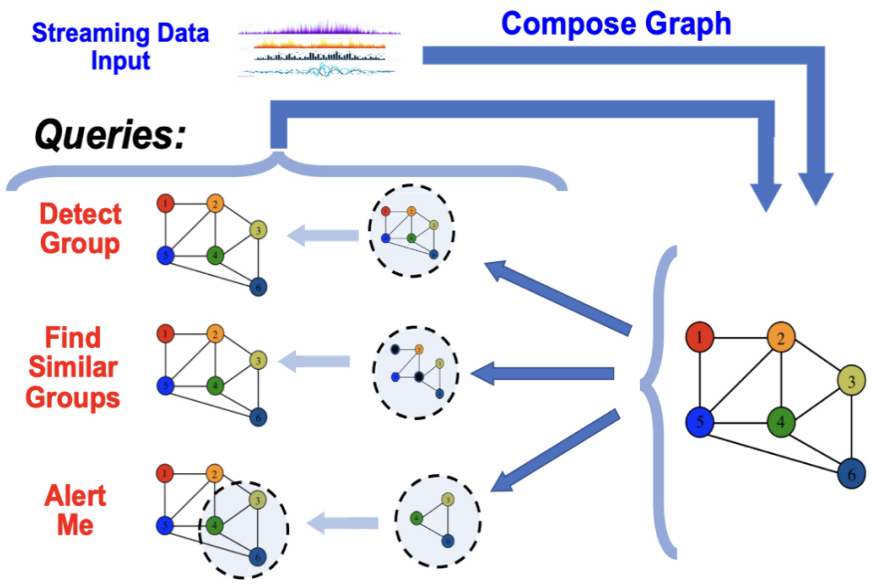
Kernel 5: Link Prediction



Predict the existence of new unlabeled edge, e_{14}

Pattern Detection





What it is:

Perform exact, approximate, and partial matching of a pattern graph against a world graph.

Let G = (V, E, C_V, C_E) be a property graph where V is a set of vertices, E is a set of edges, C_V is the set of vertex property labels, and C_E is the set of edge property labels. Let P be a pattern graph and let $\{T_1, T_2, \ldots T_K\}$ be K subgraphs of P such that their union is P.

Pattern detection use cases:

- **find similar images**
- **find organizations**
- **monitor for specific pattern**

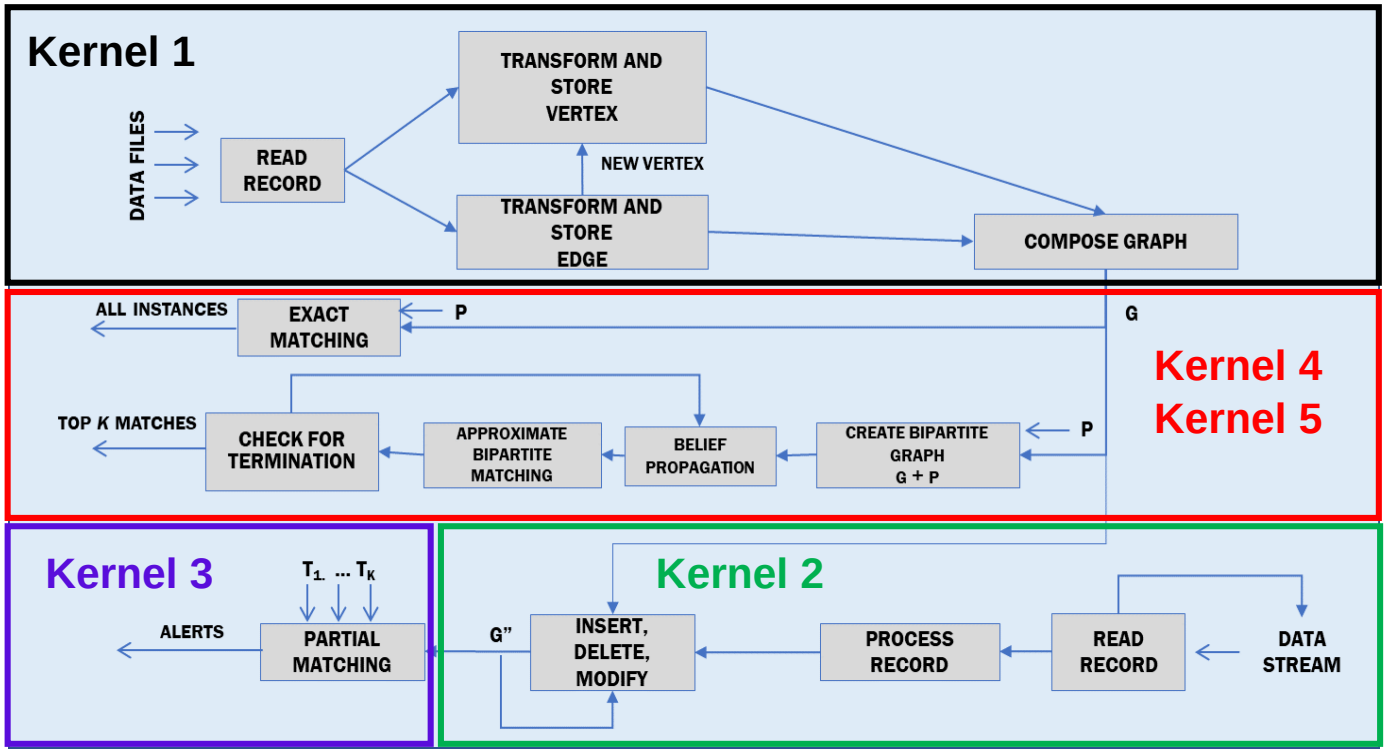
Workflow 2 – Pattern Detection

Kernel 1: measures streaming data ingestion rate, the time to read data records, transform the raw data, resolve vertex and edge ambiguities and build-out the common internal data structures used by downstream tasks.

Exact Match –
Detect Groups
Find all instances of P in G
Kernel 4

Approximate Matching –
Find Similar Groups
Return the *N* closest matches of P in G as measured by some graph edit function.
Kernel 5

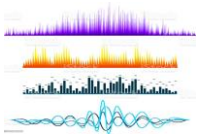
Partial Matching -
Alert Me
As new data is added to G, alert when a subgraph T_i appears in G.
Kernel 2 & 3



Pattern Detection Target Metrics

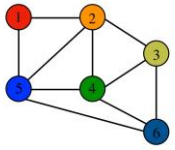
Kernel 1 – Row 1	Kernel 4 – Row 3
Kernel 2 – Row 4	Kernel 5 – Row 3
Kernel 3 – Row 4	

Kernel 1: Compose Graph



Streaming Data

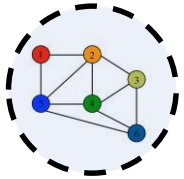
Ingest Data
Compose Graph



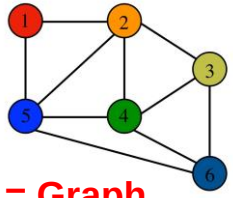
G = Graph

Compose Graph in Real Time

Kernel 4: Exact Match



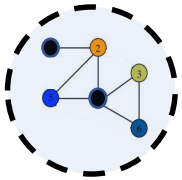
P = Pattern



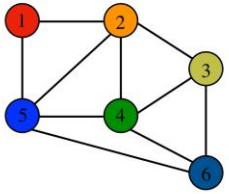
G = Graph

Find all exact instances of P in G

Kernel 5: Approximate Match



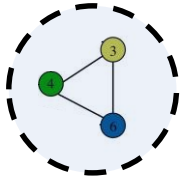
P = Pattern



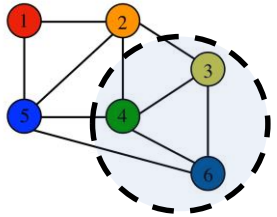
G = Graph

Find the *N* best matches of P in G,
via Belief Propagation

Kernel 2 & 3: Partial Match



T_i = i^{th} Subgraph



G = Graph

As new data is added to G, alert when
Subgraph T_i appears in G

Summary

AGILE Program Has Three tier evaluation process

AGILE will develop and release

1) End-to-end applications (Workflows) that measure full system performance
   - Data sets at different scales
   - Data ingestion and preparation
   - Multiple computational components

2) Kernels derived from data-intensive applications

3) Industry standard benchmarks
   - Breadth-first search
   - Triangle counting
   - Jaccard coefficient

AGILE utilizes ModSim to estimate and evaluate the performance of the designs, when executing the AGILE Applications