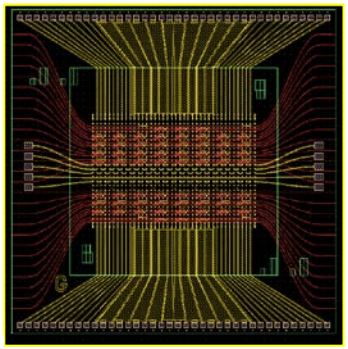
Exceptional service in the national interest

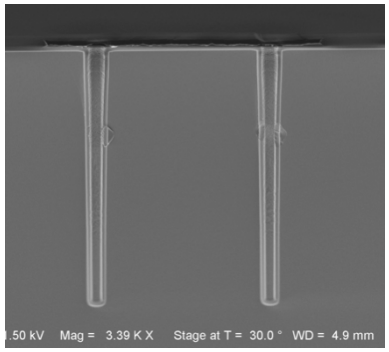
Sandia National Laboratories

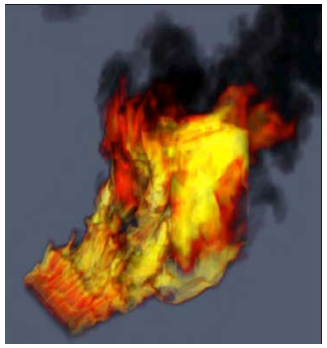
WSGA

CERI

DIMACS
Center for Discrete Mathematics and Theoretical Computer Science
Founded as a National Science Foundation Science and Technology Center







Workshop on Streaming Graph Algorithms (WSGA) Overview

Jon Berry, Cindy Phillips

Introduction by Bruce Hendrickson

With a special cameo from one of Sandia ops senior scientists

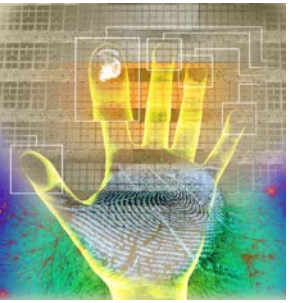
October 23, 2014

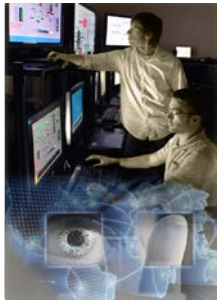
SAND2014-19712 PE

U.S. DEPARTMENT OF ENERGY

NNSA National Nuclear Security Administration

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. SAND NO. 2011-XXXXP





Cyber Engineering Research Institute

Sandia National Laboratories

Exceptional service in the national interest

U.S. DEPARTMENT OF ENERGY

NNSA National Nuclear Security Administration

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. SAND NO. 2011-XXXXP

Purpose

To help Sandia team with outside researchers in pursuit of the nation's most important cyber challenges

To help enable SNL to do ***leading edge open research*** that is threat-informed and mission-enabling

To help enable SNL to do ***mission work*** based on leading edge foundational knowledge

To help develop ***new cyber capabilities*** at Sandia

What is CERI?

- **A lab-wide community of talented cyber researchers**
  - Focused on exploratory research conducted in an open environment
- **Which serves as a resource for mission programs and line organizations**
  - A mechanism to develop a cyber-focused, interactive community at Sandia
  - A natural interface with industry, academia, and government
  - A magnet for talent and a means to develop such talent to serve the nation's cyber needs
  - Incubator for new ideas in cyber
- **That will be successful to the extent it contributes to national security by sustaining Sandia's excellence and leadership in cyber**
- **Through collaboration and open, innovative research.**

What is DIMACS?

- **The Center for Discrete Mathematics and Theoretical Computer Science (DIMACS)** catalyzes and conducts research and education in mathematical, computational, and statistical methods, algorithms, modeling, analysis, and applications.
- **Consortium of:**
  - **Direct participants:** Rutgers University, Princeton University, AT&T Labs - Research, Alcatel-Lucent Bell Labs, Applied Communication Sciences, and NEC Laboratories America
  - **Other partners:** Avaya Labs, Georgia Institute of Technology, HP Labs, IBM Research, Microsoft Research, Rensselaer Polytechnic Institute, Stevens Institute of Technology, and Yahoo Labs as well as from other federal sponsors
  - Also a Department of Homeland Security University Center of Excellence

Founded in 1989 as an NSF Science and Technology Center

Welcome!

- The **Cyber Engineering Research Institute** of Sandia National Labs welcomes you to its first official workshop!
  - Sister Computer Science Research Institute (CSRI) has hosted many
  - We're in the CSRI building now; CERI has two sites:
    - Cyber Engineering Research Laboratory (across the street)
    - Cybersecurity Technologies Research Laboratory (@Livermore, CA)

- **Goals of this talk:**
  - Give Sandia motivation for streaming graph applications
    - Theoretician alert: we have real applications you could impact!
  - Go over the workshop agenda, previewing all topic blocks

Thanks!

Rachel Leyba (Sandia): Logistics and planning

Phyllis Rutka (Sandia): Logistics and planning

Kathy Wilkerson (Sandia): Web support

Rebecca Wright (DIMACS director): DIMACS co-sponsorship

LeAnn Miller (Sandia math/CS research acting director) sponsorship

The Organizing Committee

Jon Berry (Sandia/NM)

Seshadhri Comandur (Sandia/CA, UC Santa Cruz)

S. Muthukrishnan (Rutgers, DIMACS)

Ali Pinar (Sandia/CA)

Cindy Phillips (Sandia/NM)

Steve Plimpton (Sandia/NM)

We worked hard to select interesting topics and to invite distinguished speakers!

Workshop Outline

- **Purpose:** gather top researchers to discuss topics related to streaming graphs
- **Structure:** topics broken into six blocks:
  - Block I: streaming graph application drivers, infrastructure
  - Block II: streaming models, frameworks, data generation
  - Block III: streaming graph algorithms without sampling
  - Block IV: streaming linear algebra
  - Block V: streaming data structures
  - Block VI: sampling and measurement
- **Session organization**
  - Let experts address each topic via short talks
  - Each block will have a panel discussion
  - We'll have open problem sessions to end each day

Workshop Goals

- **Influence each other's research directions**
- **Initiate and/or foster collaborations**
- **Introduce CERI and CSRI as desirable places to come visit**
  - We have real problems
  - We have a base of researchers and practitioners
  - We love academic visitors (Summer? Short visits? Just ask..)
  - Send us your students
- **Produce a deliverable: a DIMACS-style short report of open problems and workshop outcomes**

Social Event Tonight! ("no host")

- **Get dangled 1000 feet in the air!**
  - In a tram filled with WSGA friends
  - Probably talking about streaming algorithms
- **Enjoy a Canadian-style climate for a bit**
  - Bring a coat!
  - Drink water
- **Then descend and have dinner at the base**

- **Details in registration materials**

Note: please try to finalize reimbursement forms quickly

Types of Streaming in Practice

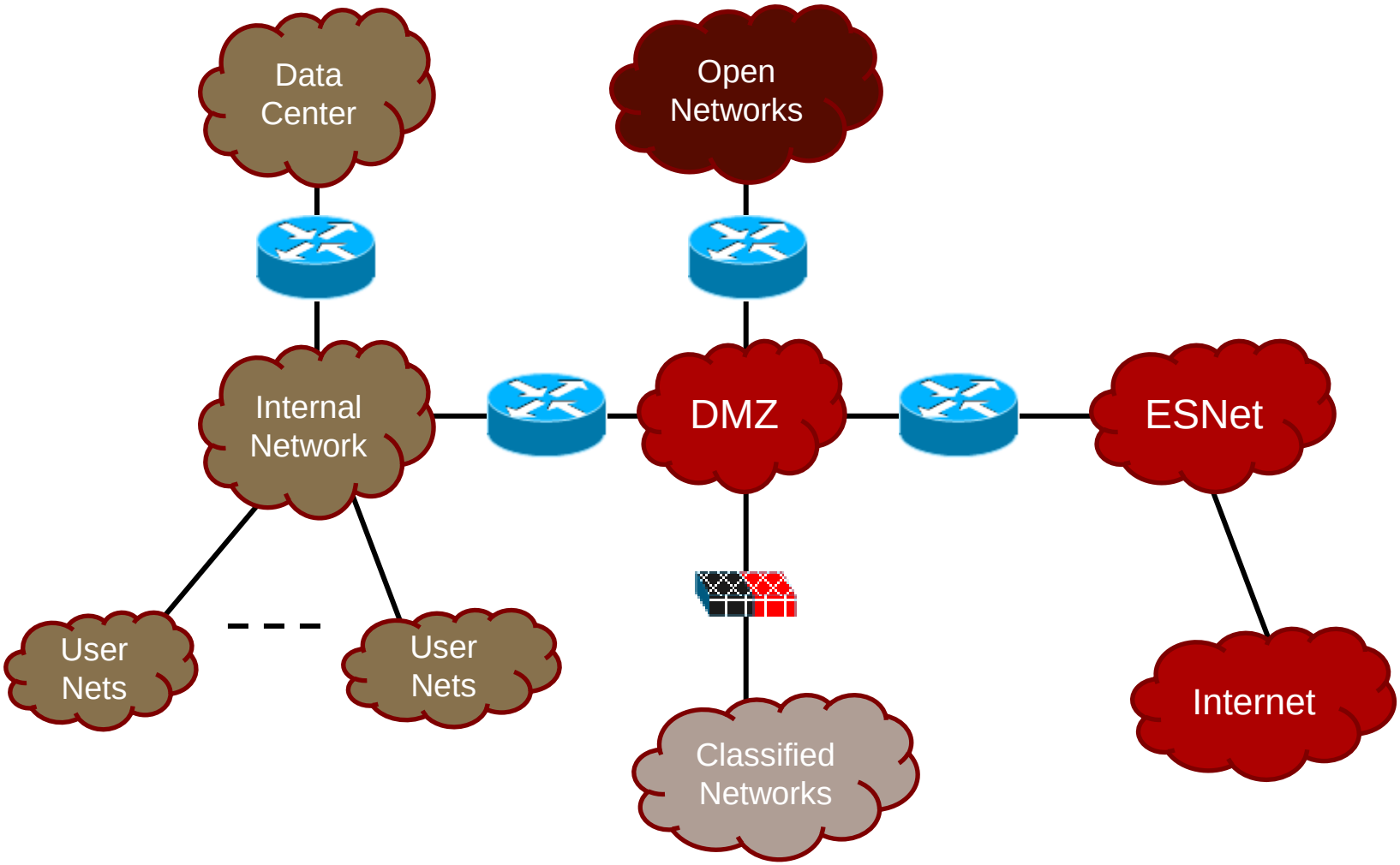
- **Type 1: a firehose of data is aimed at you**
  - You have only constant storage space for in-stream processing
  - The stream of relational data (e.g. edges) is intense and infinite
  - Forget about disk except for off-line storage
  - What can you do algorithmically?

- **Type 2: you have access to a large, finite pool of data**
  - You have limited RAM (no hope of bringing all data in together)
  - Stream the data through processors (e.g. MapReduce)
  - You are now probably disk-bound (and/or require HPC)
  - Now what can you do algorithmically?

Special Guest: John Naegle

- Senior Scientist
  - not many at Sandia (and Cindy Phillips is one too!)
- Knows Sandia cyber operations and networking
- He'll talk about infrastructure for Type 1 streaming

Sandia's Core Network

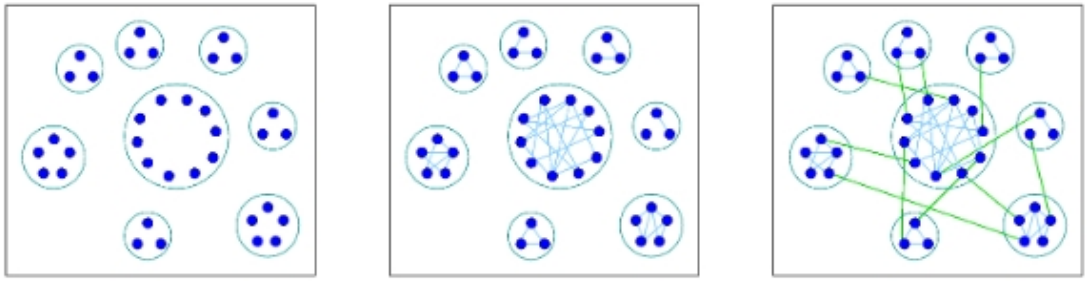


Sandia/DOE Streaming Data Sources

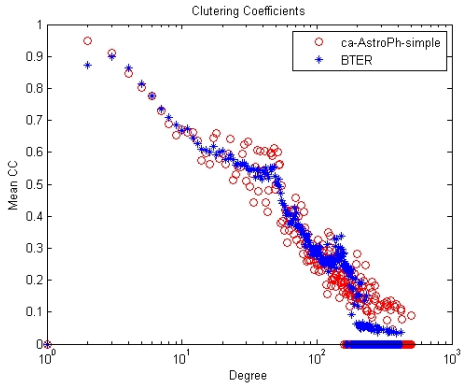
- **Cyber gateway traffic**
  - Sandia has to defend its networks (like any major entity does now)
  - Users have limited privacy in the name of defense
  - (Type 1, Type 2): bipartite and general graph and hypergraphs: web hosts, IP addresses, emails, internal host-to-host, process-to-resource
- **Sensor array data**
  - E.g. Distributed water sensors generating signals; are events happening?
- **Satellite image data**
  - (Type 2): stored images processed into graph features
- **Telescope data**
- **Power grid data**
- **HPC simulation** (Type 2)

Some of Sandia's Related Projects

- **"FEAST" (DARPA, current)**
  - Type 1 sampling algorithms
  - *Example: sample triangles and 4-cycles in an edge stream to est. counts*
  - *Example: "BTER" model*
  - POC's Tamara Kolda (tgkolda@sandia.gov), Ali Pinar (apinar@sandia.gov)

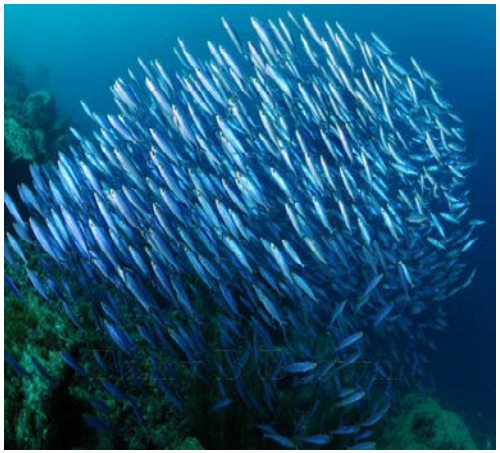


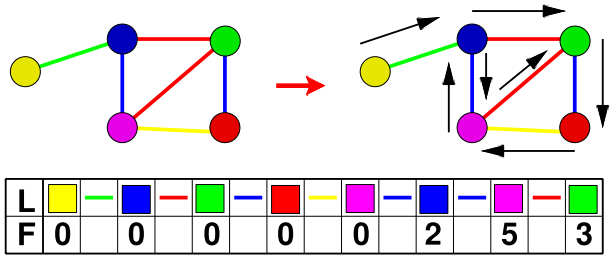
(a) Preprocessing: Distribution of nodes into communities (b) Phase 1: Local links within each community (c) Phase 2: Global links across communities



More of Sandia's Related Projects

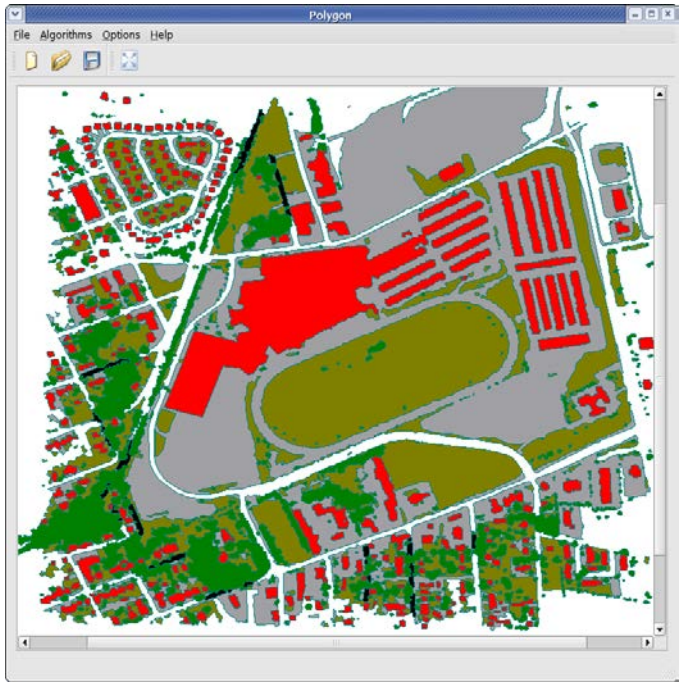
- **"Streaming LDRD" (Sandia-internal, finished)**
  - Type 1 and Type 2 graph and machine learning algorithms
  - *Example: created "phish" streaming framework (Plimpton, et al.)*
  - *Example: streaming subgraph isomorphism (typed) (Plimpton)*
  - *Example: maintain connected components in an infinite edge stream (B.,P.)*
  - *Example: build decision tree-based models in a stream (Kegelmeyer, et al.)*
  - POC Steve Plimpton (sjplimp@sandia.gov)

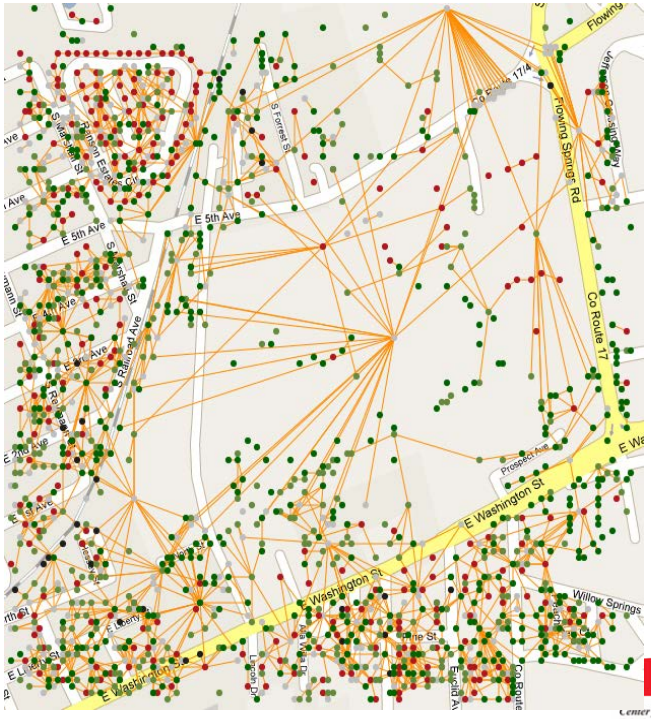




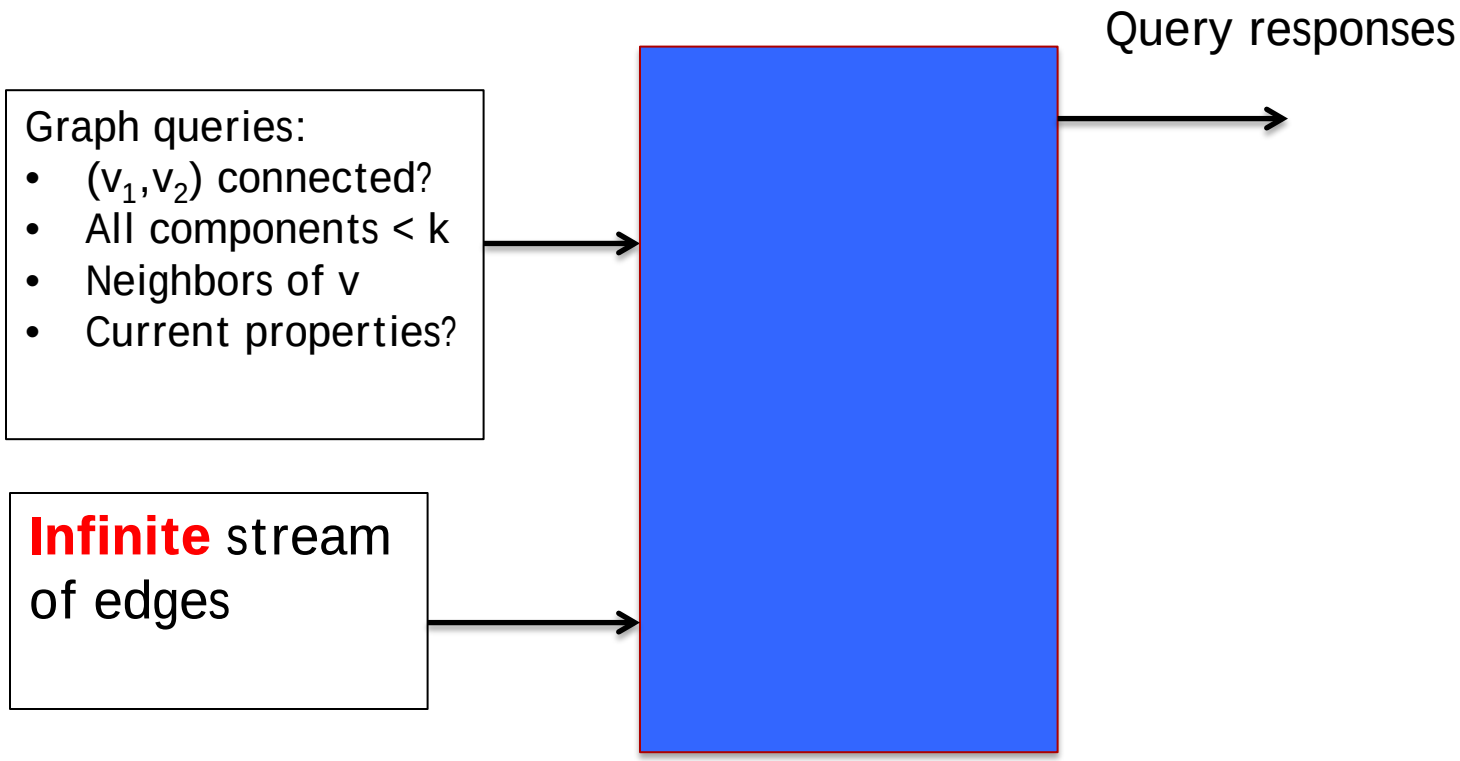
More of Sandia's Related Projects

- **"GeoGraphy" (Sandia-internal, current)**
  - P.O.C.'s R. Brost, W. McLendon, M.D. Rintoul
  - Type 2 algorithms
  - *Example: model overhead image features as graph edges, do pattern matching*



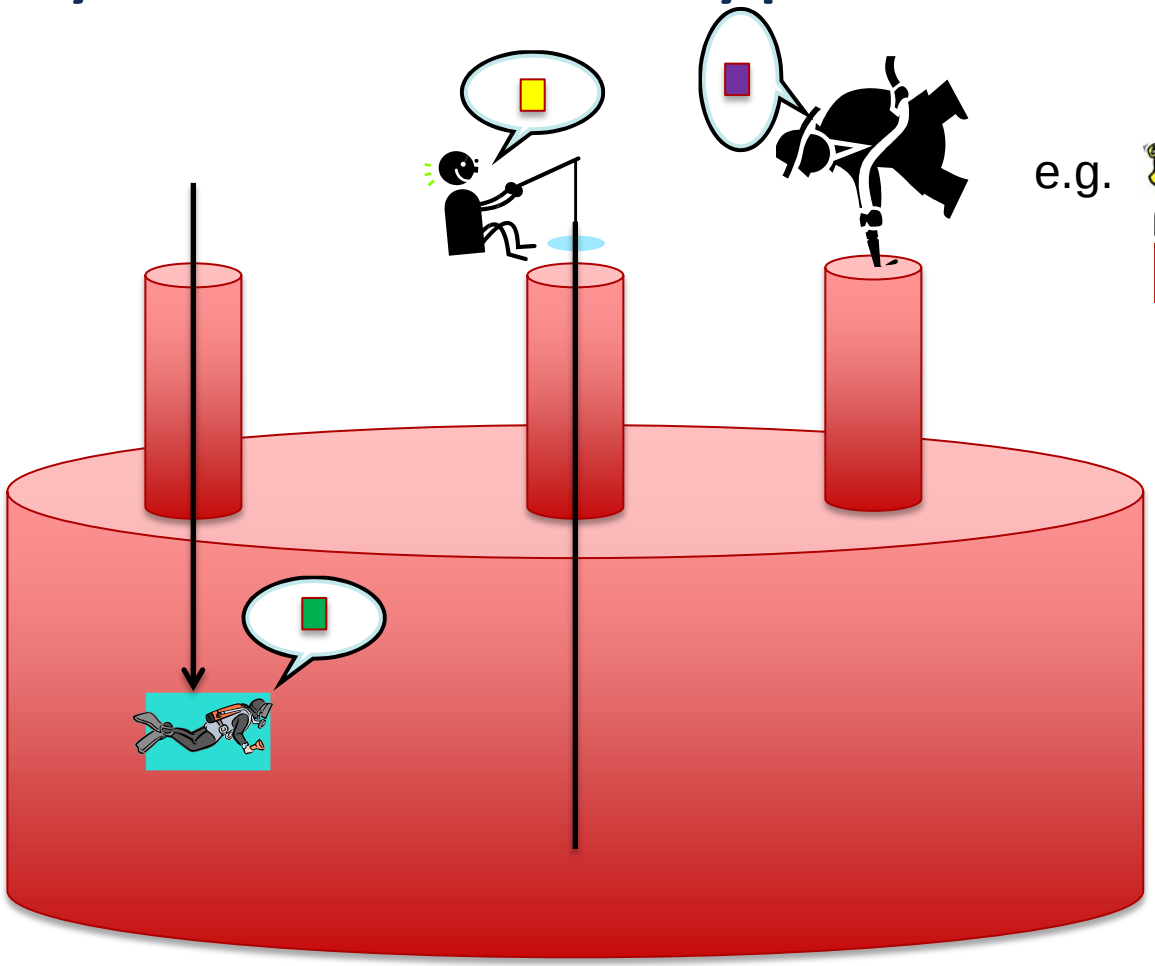


Analyst Needs for Type 1 Streaming



- Ediger et al: approximate queries, large batches

Analyst Needs for Type 2 Streaming



e.g. hadoop APACHE HBASE

- Point query
- Fast ingest
- Global slice & dice, Data mining

Fault tolerance, code transparency, flexibility, data residence

Streaming Models of Interest

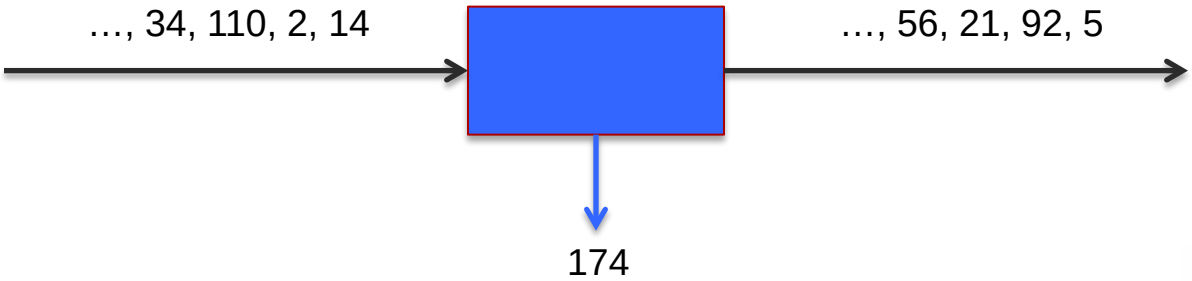
In the coming sequence of slides, we'll discuss theoretical models and their tie (or not) to practical Sandia problems

Classic Streaming

- Information arrives piece by piece as generated in a (huge) stream
- Answer a question as the data set streams by
  - Use much less local space than the stream size

- Example: Watch a permutation of 1,…,n (n known) with one number missing. You have space for one number. Determine the missing number.

- Answer: store the sum of the numbers you have seen.



Streaming Relevance

- Computer communication networks link entities
- Represent relationships with a graph

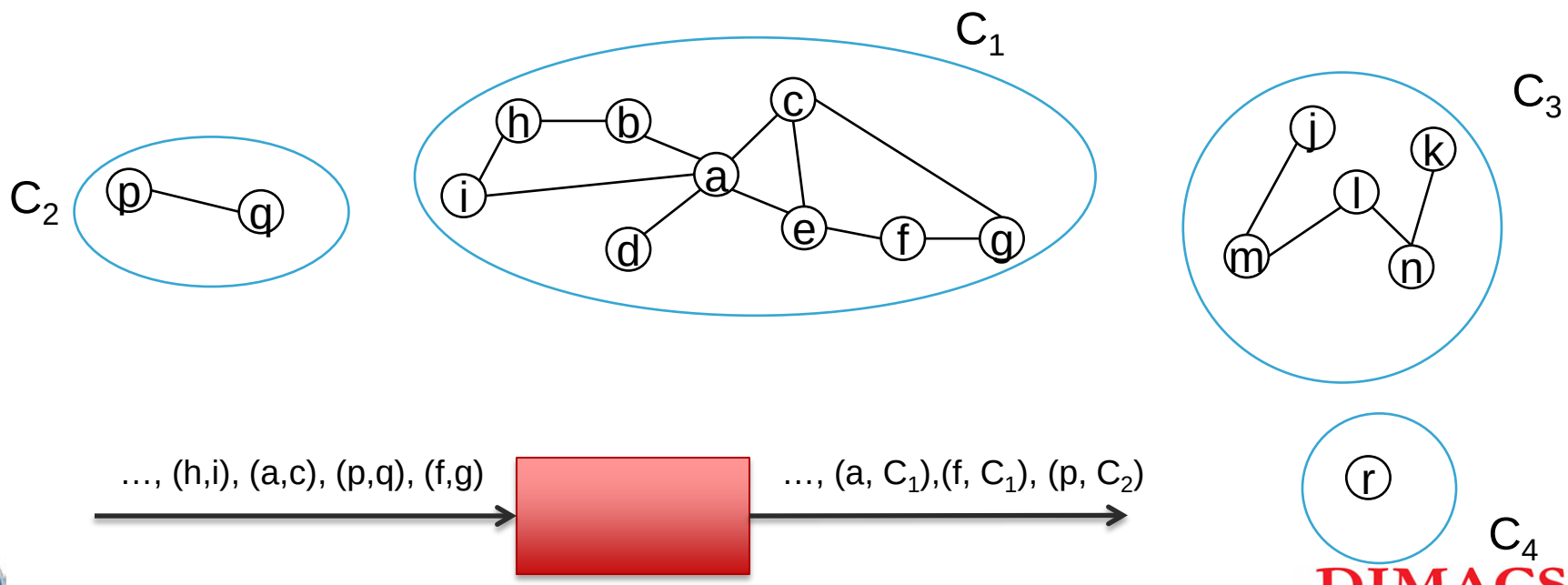
Cyber traffic/activity is a stream through time

- Stream are huge
  - Humans cannot keep up
  - Gap will only increase
- It's time to develop fundamental streaming graph algorithms to partially automate the analyst's tasks

- Study in careful control of data movement/aggregate storage use
  - Relevant for exascale hierarchy

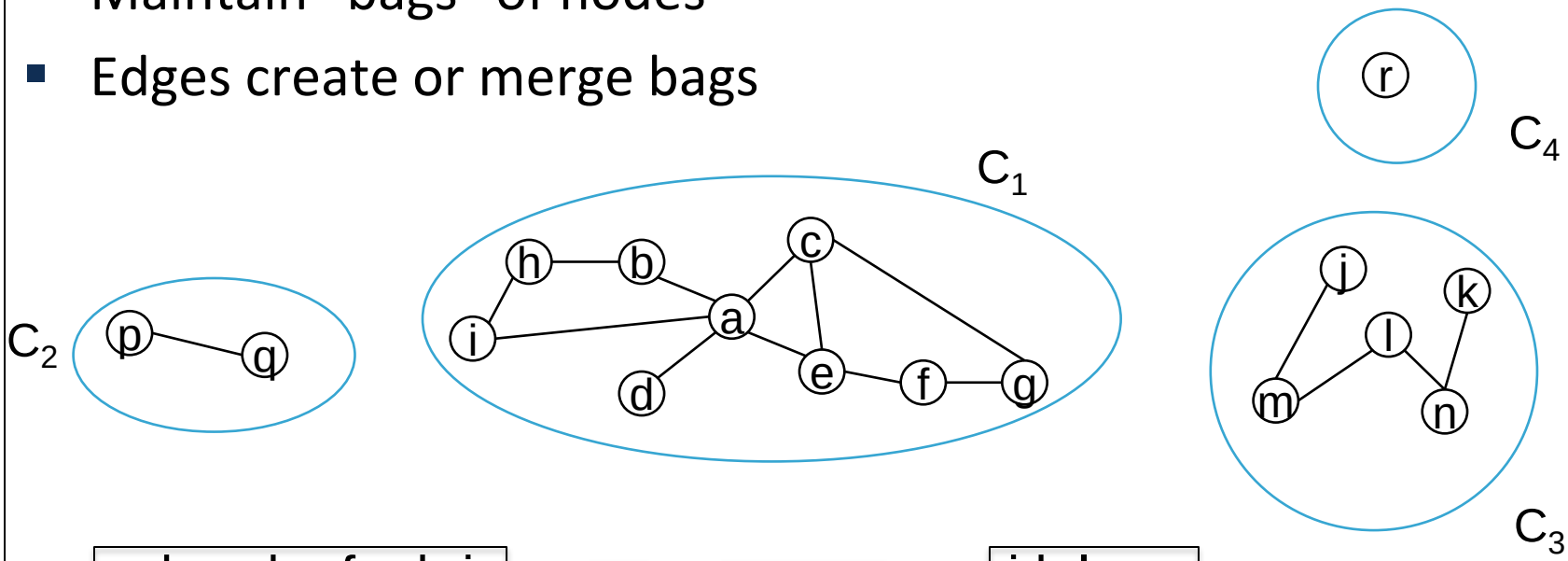
Connected Components

- Input: stream of edges (learn nodes from edge)
- Output: (node, label) pairs
- Two nodes have the same label if there is a path between them
- Can't output 2 pairs with different labels until seen all of (finite) graph
- Feigenbaum, Kannan, McGregor, Suri, Zhang, Ω(n)-bit space lower bounds for connectivity with one pass.



Semi-streaming Connected Components

- Graph with |V|=n. Allowed O(n polylog(n)) space
- Can store all the nodes
- Maintain "bags" of nodes
- Edges create or merge bags



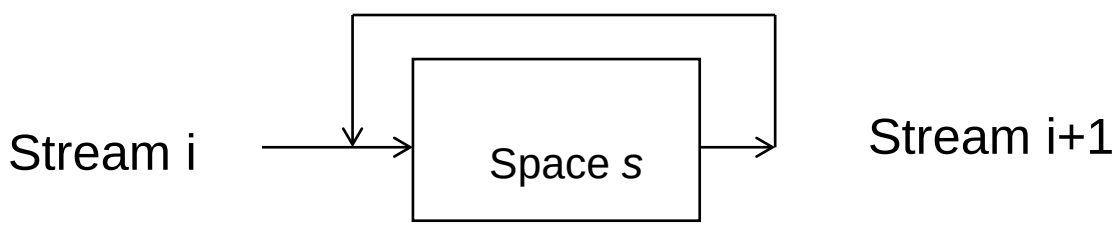
a,b,c,d,e,f,g,h,i | r | p, q | j,k,l,m,n

W-Stream Model

- Read a stream, write a stream for another pass
  - **Finite** Stream
  - The rewrite stream is "in the air."
  - Trade off space vs # passes
- Demestrescu, Finocchi, Ribichini:

$$s \text{ space}, O\left(\frac{n \lg n}{s}\right) \text{ passes}$$

n = # nodes



O.K. for Type 2 Streaming

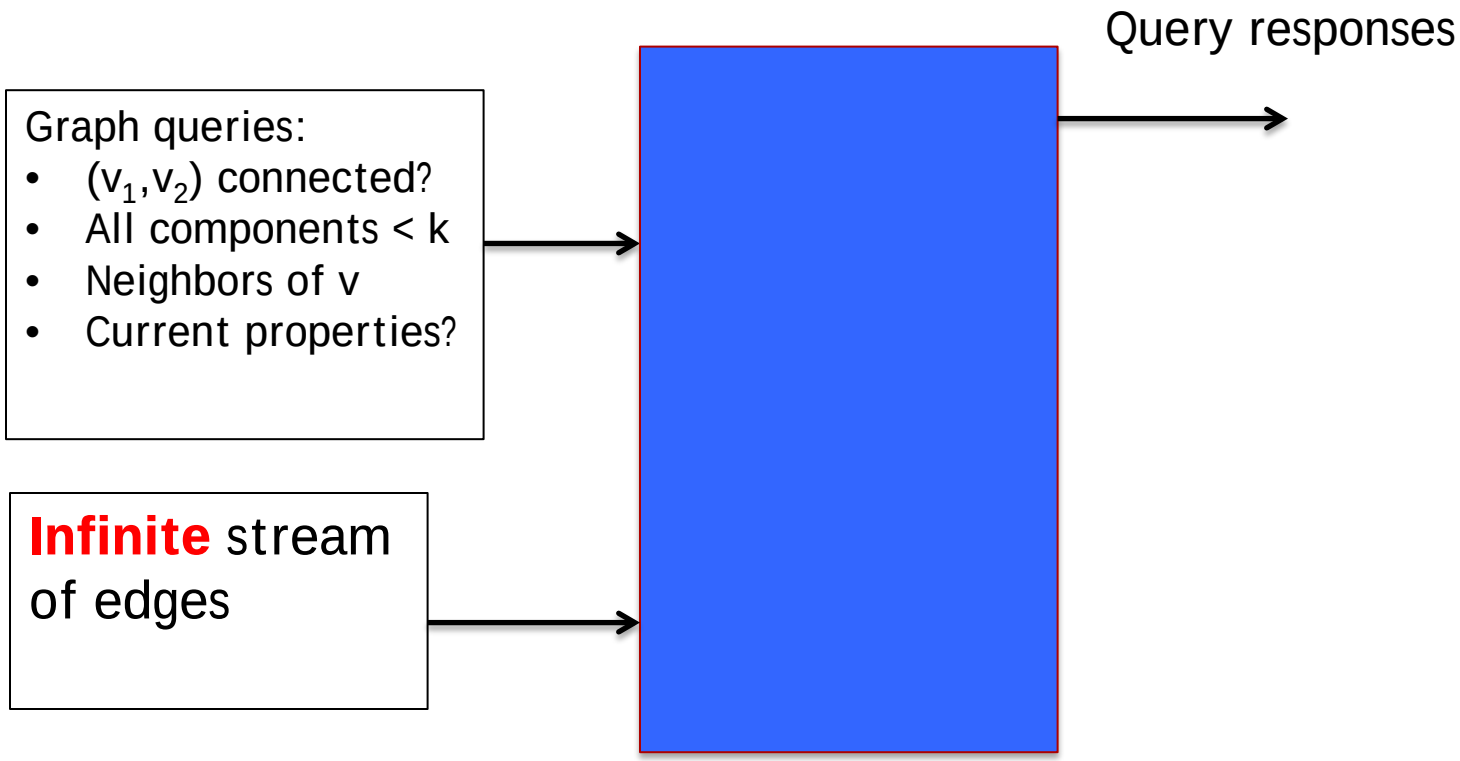
C. Demetrescu, I. Finocchi, and A. Ribichini. Trading off space for passes In graph streaming problems, ACM Transactions on Algorithms, Vol. 6, No 1, Dec 2009.

Other models somewhat like W-stream

- Streaming + sorting (Aggarwal, Datar, Rajagopalan, Ruhl, FOCS 2004)
  - Normal streaming passes and sorting pases
  - S-t connectivity in O(log n) passes using O(log n) space (randomized)
- Best-order checking
  - Stream provided in best possible order
  - One pass
  - Space complexity to test a property
  - Test graph connectivity (one component) with $O(\log^2 n)$ space

Analyst Needs for Type 1 Streaming

(Reprise)



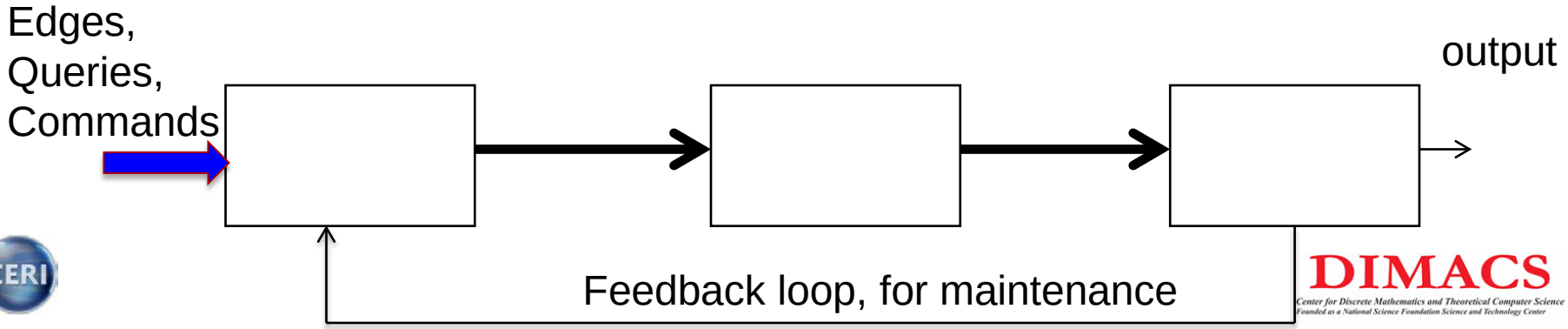
- Ediger et al: approximate queries, large batches

Dealing with an Infinite Stream

- Efficiently use aggregate local memory across processors
  - O(1) space per edge
  - Edges not dropped unless the system is storing Ω(ps) edges ("full")
- Aging
  - Command (in stream) to remove all edges older than t
    - Reduce space when the system is filling up
    - Newer edges likely more interesting
  - Must recompute components, so must store all edges
  - No queries till recomputed, output token when OK to resume
  - No edges dropped during recomputation
- Queries
  - Answer relative to graph at time of query
  - Constant-sized answer immediate, non-constant as able

Model: X-Stream

- Parallel ring architecture with systolic (synchronous) communication
  - Input is 1/k of bandwidth
  - Built system with asymmetric bandwidth distribution?
- For handling infinite streams (though graph finite)
  - W-stream (finite) can fill and spill, reduced latency
- Stream entry point can (must) move around the loop
- Queries stream through in one pass, possible latency
- Theoretical measures: k, worst-case/avg update/message, post-aging stabilization speed, constant in memory usage guarantee
- Practical measures: streaming rate, query response latency
- Explore effects of system balance



Type 2 Steaming Models

- MapReduce
- Are Pregel, etc. based on theoretical models (as opposed to styles like BSP)?
  - We haven't studied this topic much; discussions welcome