Presented by: Aaron Bossert, Cray Inc.

CRAY

COMPUTE | STORE | ANALYZE

Network Security Analytics, HPC Platforms, Hadoop, and Graphs… Oh, My

The Proverbial Needle In A Haystack Problem

The Nuclear Option

Problem Statement and Proposed Solutions

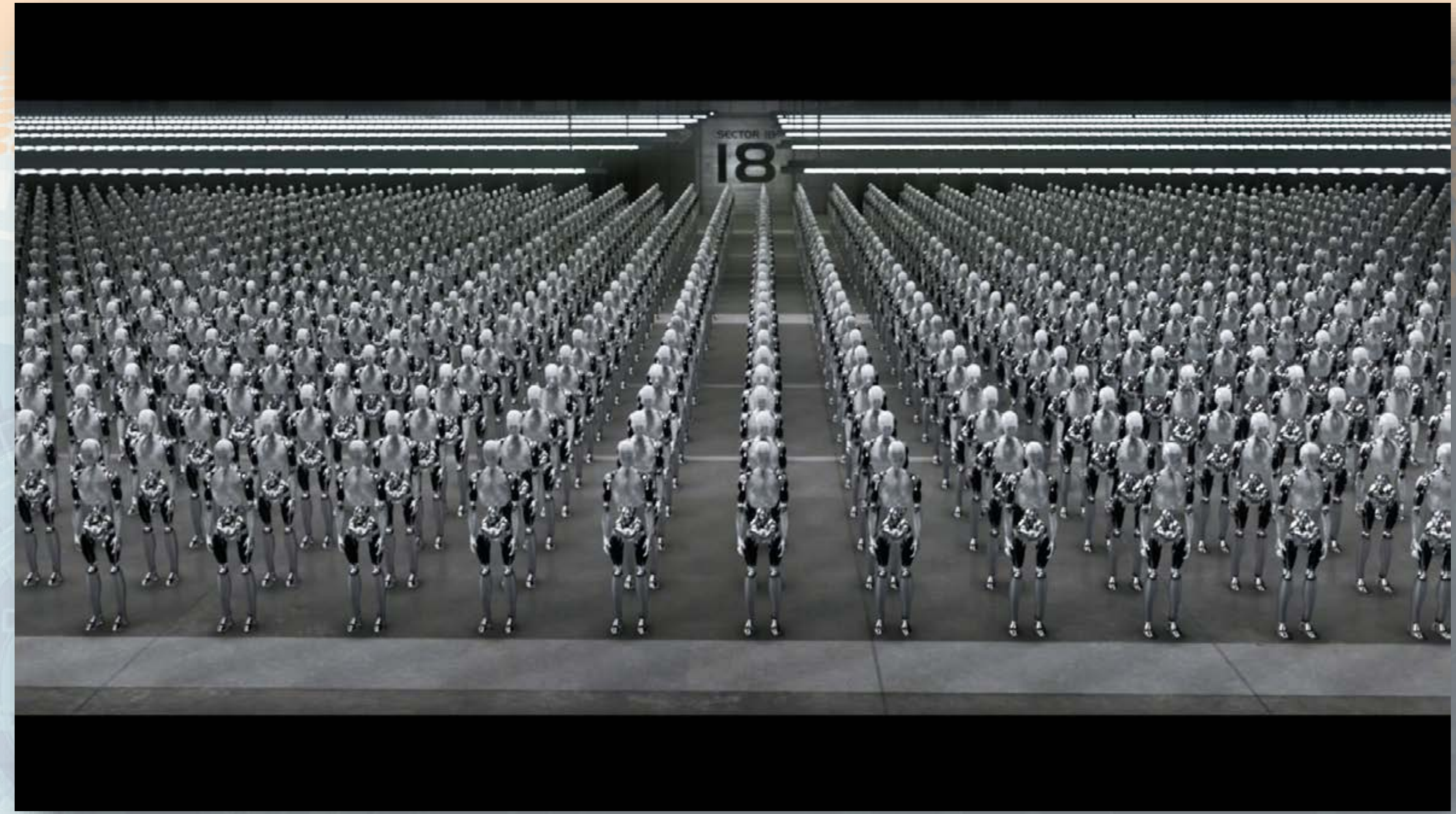
The “Spock” Option

Problem Statement and Proposed Solutions

The "how we've been doing it" Option

Problem Statement and Proposed Solutions

<u>We would like to humbly suggest bringing more workers to the party</u>



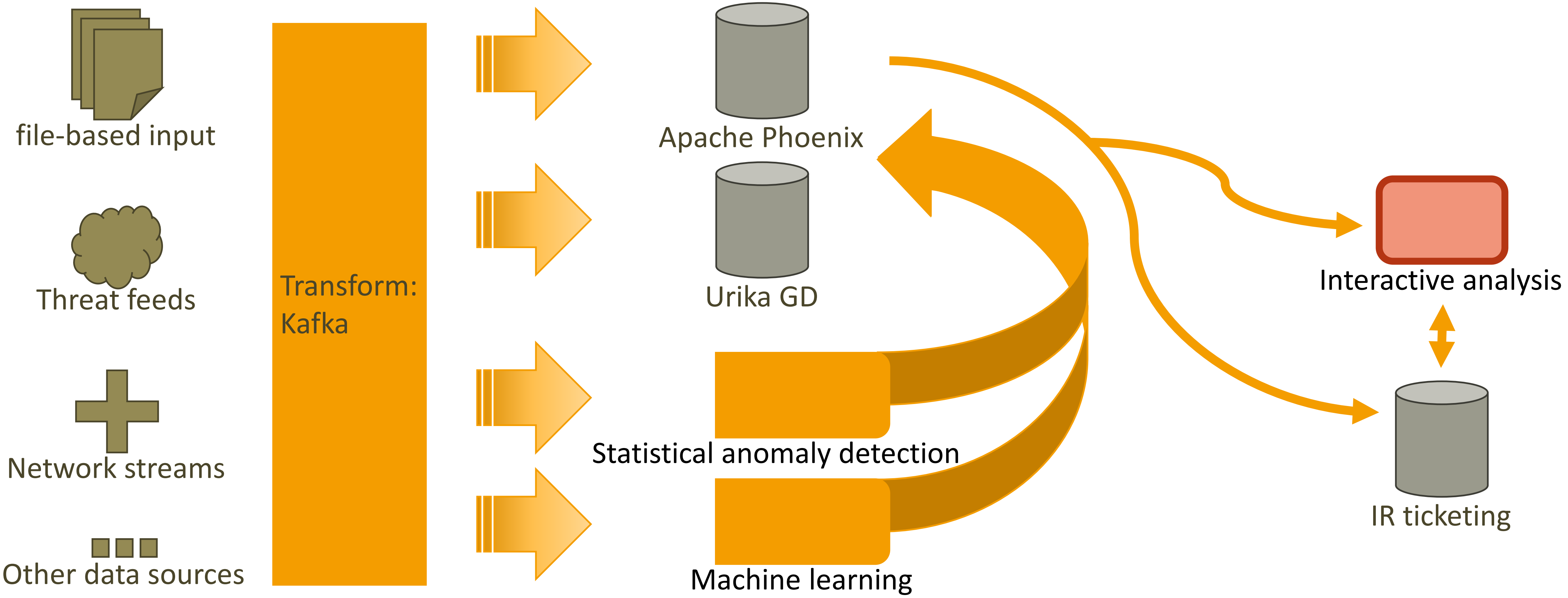
Problem Statement and Proposed Solutions

Prefer a less recent pop-culture reference?

Background

- **Technologies**
  - Urika GD – RDF triple store – proprietary architecture (XMT, XMT2)
  - Urika XA – Hadoop appliance – x86 based architecture
  - Next?
- **Customer needs**
  - Massive scale
  - Flexibility to develop different use cases on one platform
  - Prevent cluster sprawl (e.g. dense racks)
- **Example Use Case: Network security**
  - Near-real-time ingest
  - Machine learning applied to streaming and static data (e.g. IR and Forensic investigations)
  - Flexible framework – easy to extend and modify
  - “bag of tools, not a bag of hammers” (e.g. complementary technology stack to address different workloads)
  - Support novice to expert users (e.g. ”easy button”, if you want it; spin all the knobs if you don’t)

High-Level Architecture



Architecture Highlights

- **Credit where credit is due**
  - Architecture is heavily based off of and influenced by Cisco OpenSOC
  - Changes made to take advantage of newer technologies (e.g. Apache Phoenix)
- **Ingest**
  - Apache Kafka selected for high throughput
  - Kafka development is relatively language agnostic (i.e. lower learning curve)
  - Kafka handles streaming and file-based input well (assuming sufficient IO to/from disk)
- **Processing and machine learning**
  - Still evaluating Kafka and Apache Storm, bulk of processing is done with Kafka for now
  - Existing algorithms are leveraged, new ones implemented trivially
  - Queries can be directed to the most appropriate tool, taking advantage of both traditional row/column and graph store strengths to answer questions
- **The end result**
  - Nearly raw data stored in Phoenix for maximum flexibility
  - Automated and manual analytic results aggregated and used for confidence scoring
  - Automated alerts used to create tickets past a certain threshold
  - Near-real-time and forensic use cases can be supported on a single platform seamlessly
  - Most of the pipeline can be extended in any programming language and potentially re-use existing code bases, lowering the bar to entry in a new environment

Input Data

- **Off the wire, from files, or both**
  - Kafka Producers used to efficiently manage and add new data sources
  - Currently have parsers for the following:
    - Netflow
    - Cisco ASA
    - Passive DNS (collected from internal DNS servers)
    - Publicly available black/white lists (fetched at regular intervals based on the data source)
    - WHOIS
    - Active directory
    - GeoIP
    - DHCP
    - PCAP
    - Many more supported by Cisco OpenSOC

Scoring Suspicious behavior

- **Anomaly detection**
  - Track both internal and external entities on a per-entity basis
  - *Examples of dimensions tracked*
    - Temporal patterns (e.g. time of day, day of week, etc.)
    - Traffic volume
    - TCP/UDP port usage
    - Protocol usage
- **Existing threat data**
  - Black/white lists
  - Firewall/IDS/IPS/SIEM logs
- **Pulling it all together**
  - Scores are transient in the sense that they apply for a given window of time (e.g. arbitrarily by hour or by day)
  - Calculated across all alerting mechanisms; use weighting
  - Weighted entity or traffic (depending on context) score crossing a threshold is flagged for analysis/verification
  - Automated analytics can run side by side with ad-hoc queries
  - Ad-hoc analysis can be integrated into the automated workflow including replay of past traffic
- **Difference from standard IDS/IPS/SIEM**
  - More complex pattern and behavior-based risk scoring based on multiple dimensions
  - Risk score's temporal aspect can be used to potentially block traffic dynamically and in a more fine-grained fashion

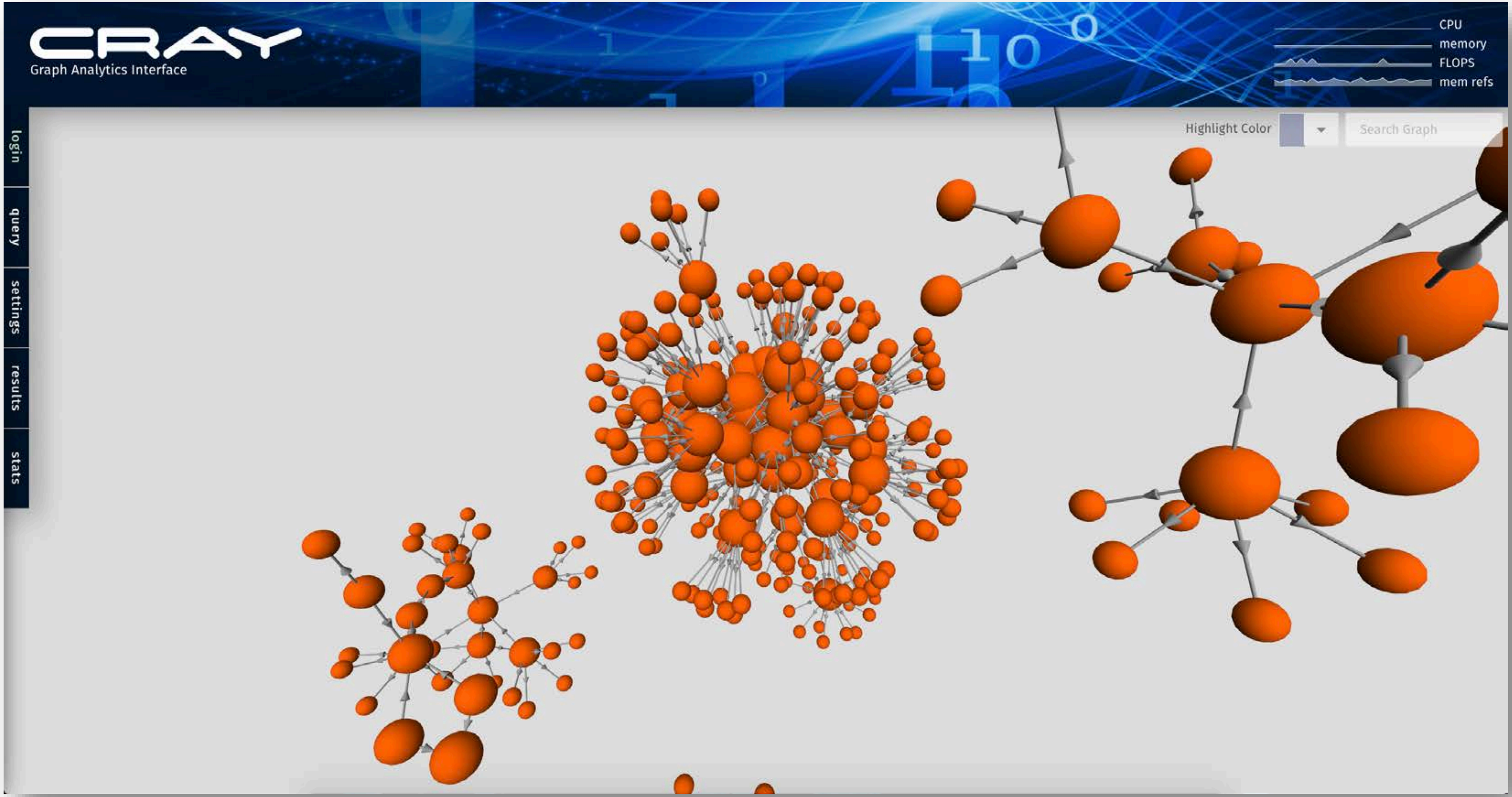
Scoring Example

Time	Anomaly	Weight	Score
2016-01-02 13:10:02.223657	Abnormal SSH activity	2	0.2
2016-01-02 13:14:33.114538	Abnormal UDP port usage	2	0.3
2016-01-02 13:36:21.685934	Blocked traffic to blacklisted IP/domain	4	0.7
Weighted score for **2016-01-02 13**:00:00.000000			**0.6**
2016-01-03 08:44:55.300978	Unusual temporal activity (compared to baseline)	1	0.3
Weighted score for **2016-01-03 08**:00:00.000000			**0.3**
2016-01-03 10:02:31.000494	IDS alert	5	0.8
2016-01-03 10:03:01.756002	Allowed transfer to domain closely associated with blacklisted IP (badRank)	4	0.6
Weighted score for **2016-01-03 10**:00:00.000000			**0.7**

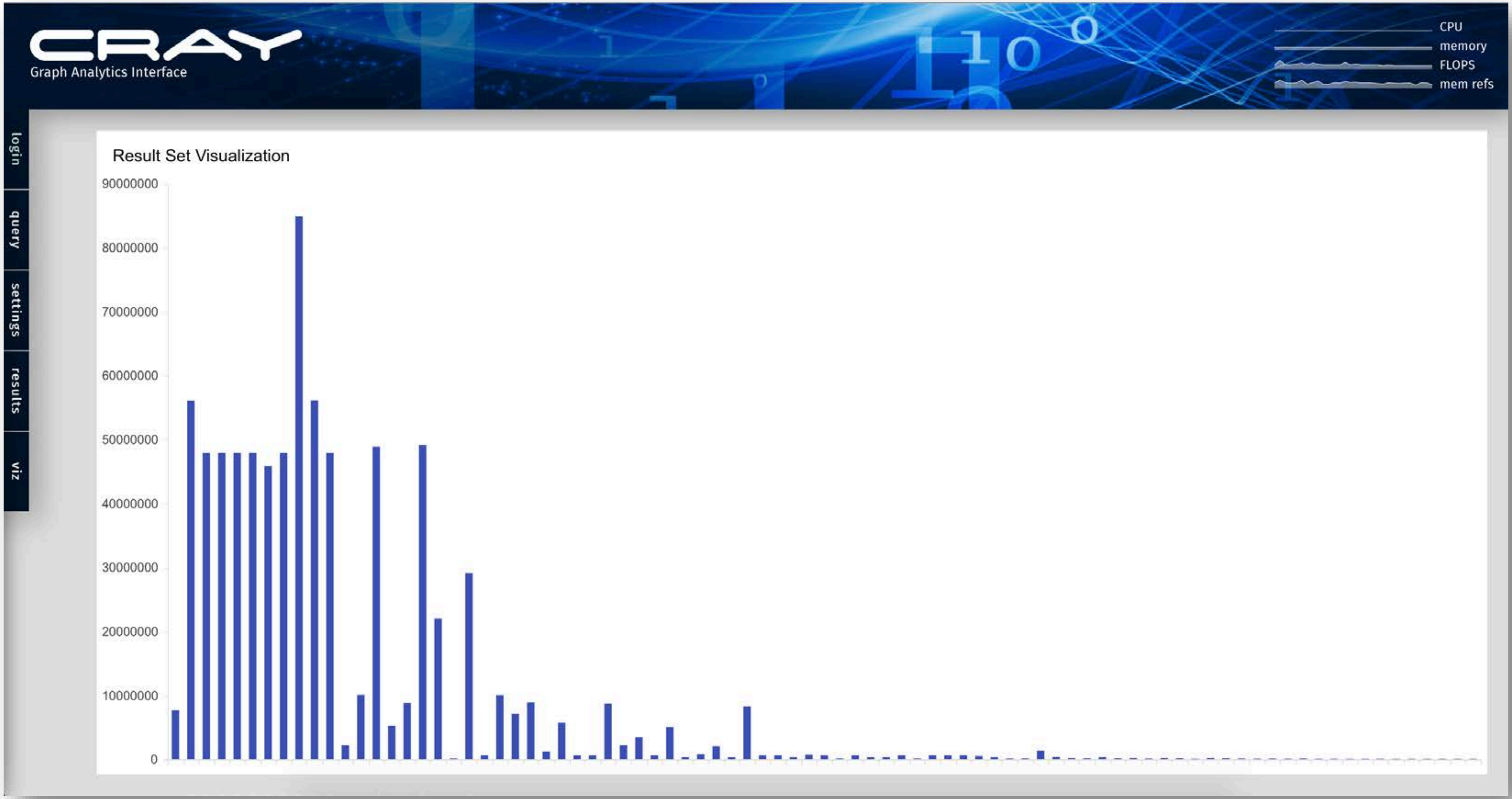
Graphs

- **BadRank**
  - Essentially a seeded PageRank score
  - Allows for determining guilt by association; Specifically, uses passive DNS and/or WHOIS
- **Centrality**
  - Identifies bridge nodes between clusters/groups
  - Enables Identification of chokepoints for blocking traffic
  - Can be used to analyze botnet C^2 structure
- **Community detection**
  - Flexible multi-dimensional similarity
  - Can be used to classify traffic patterns and/or hosts
  - Can be used to identify additional compromised/malicious entities
- **Summary**
  - Graph algorithms provide a distinct class of tools not able to be easily implemented with relational data
  - Compliments statistical anomaly detection by providing additional dimensions
  - Handles joining disparate and complex datasets for enrichment

User Interface – Graphs …



User Interface – Or tabular data in one UI



Questions, Contact, and Further Details

M. Aaron Bossert
bossert@cray.com
Cray Inc.