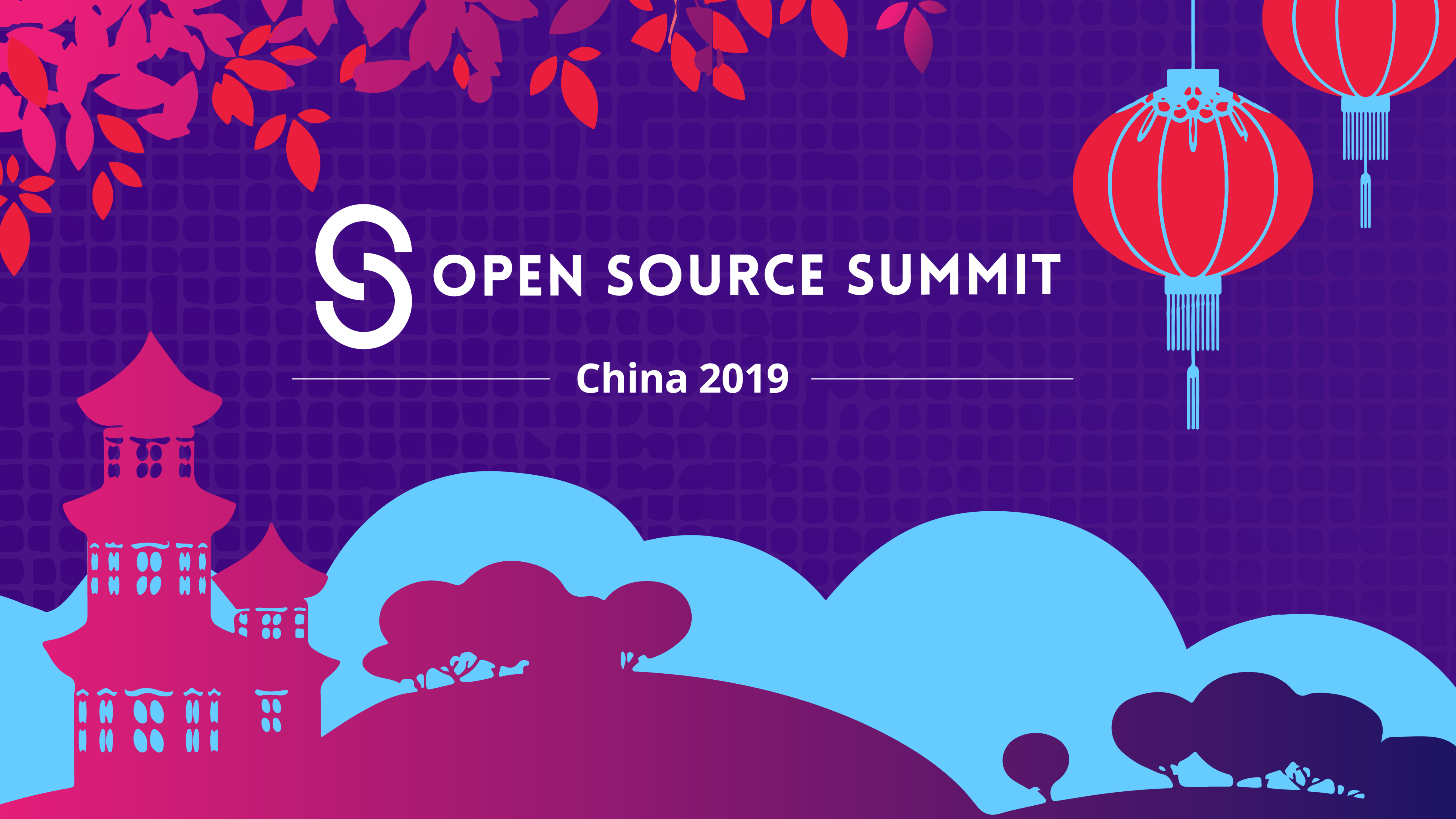


OPEN SOURCE SUMMIT

China 2019

OPEN SOURCE SUMMIT

China 2019

Introduction to JanusGraph

Jason Plurad
Software Developer
IBM Cognitive Applications

Agenda

- **Graph Use Cases ←**
- Open Source Graph Evolution
- Future Directions

Property Graph Model

OPEN SOURCE SUMMIT China 2019

- Vertices
- Edges
- Properties

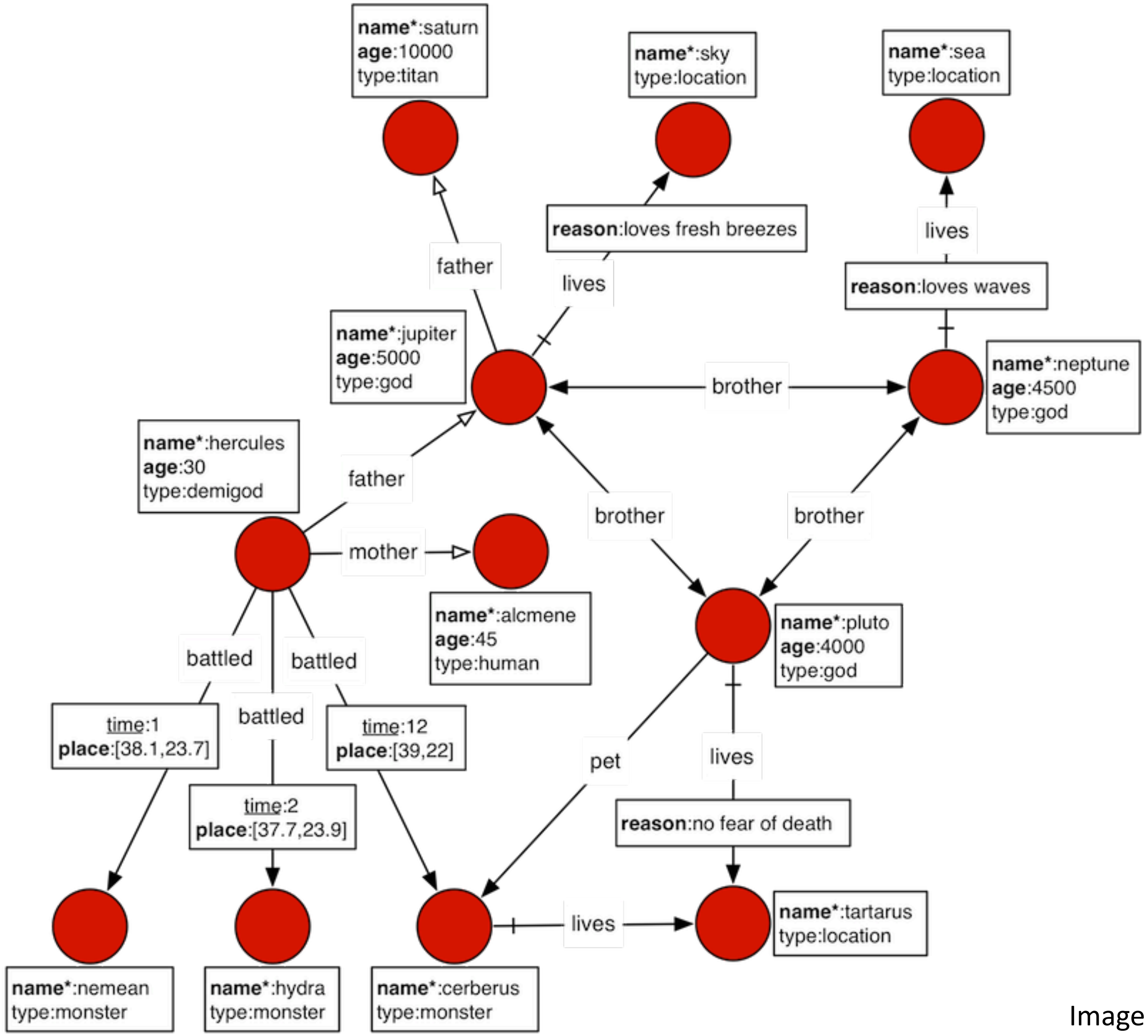
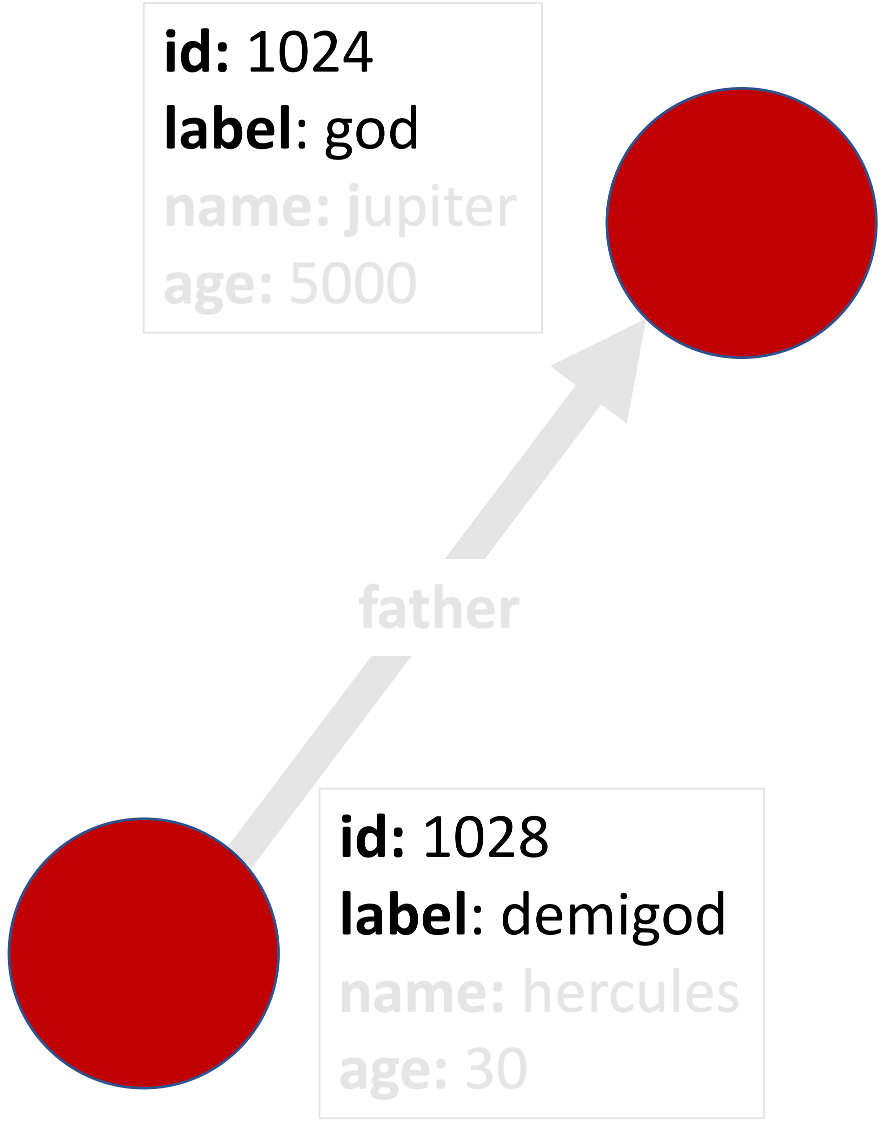


Image: Graph of the Gods (CC-BY-4.0)
https://janusgraph.org

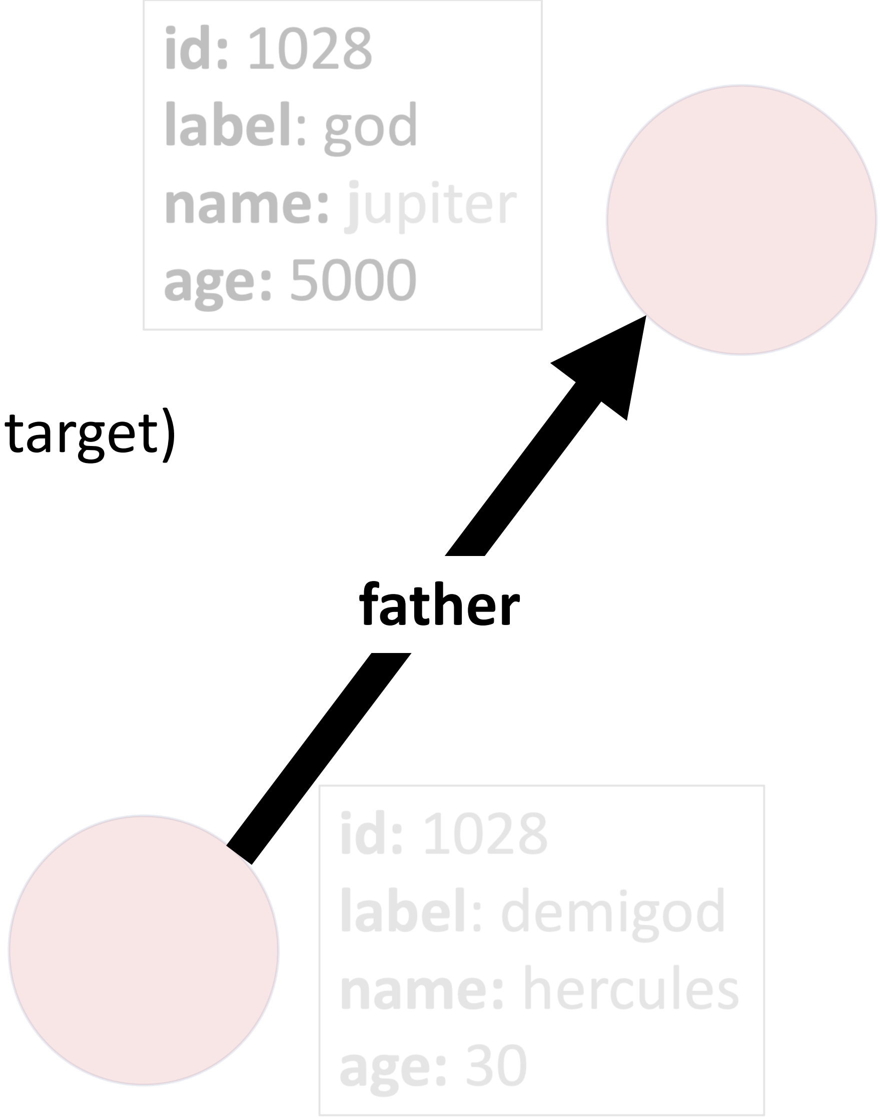
Vertex

- **Vertices ←**
  - An entity in the graph
  - Has a unique identifier and a label
  - Can connect to other vertices with an edge
  - Can contain additional properties
- Edges
- Properties



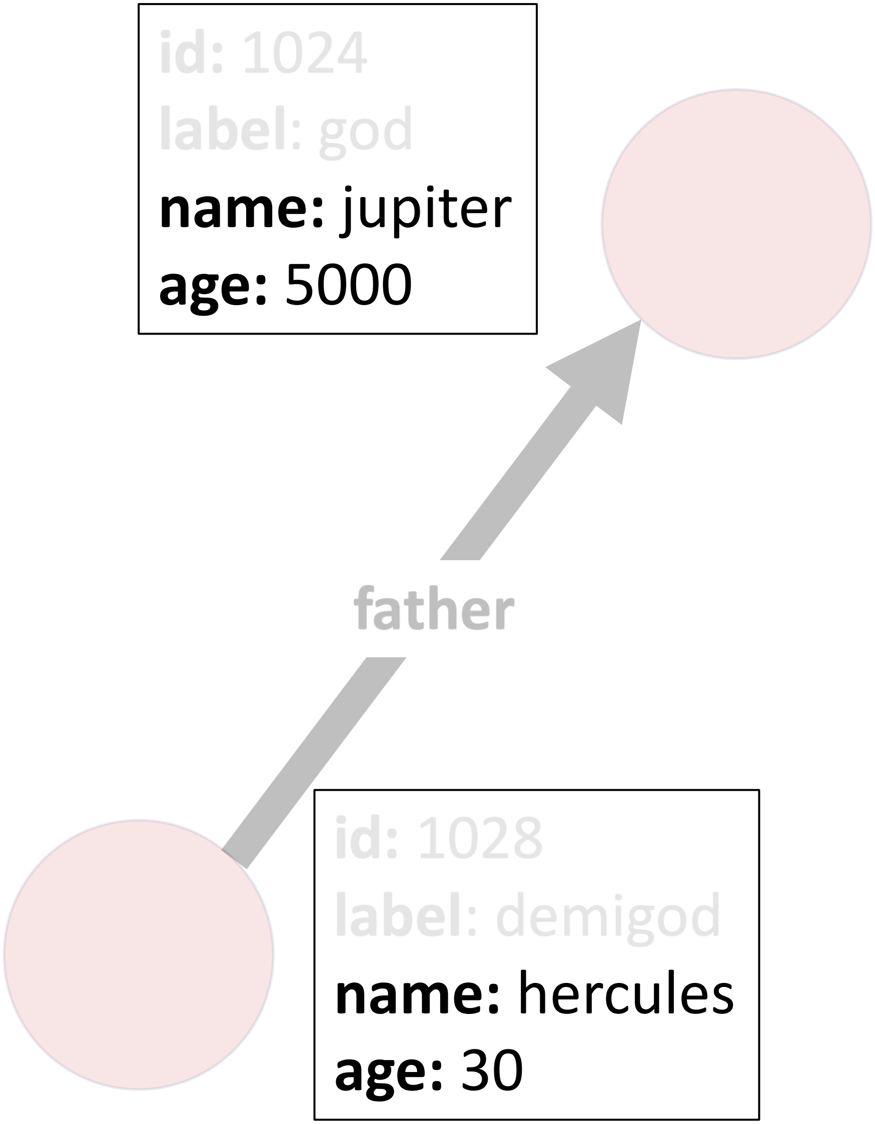
Edge

- Vertices
- **Edges** ←
  - A directional relationship in the graph
    - Out-vertex (source) connects to in-vertex (target)
  - Has a unique identifier and a label
  - Can contain additional properties
  - Multiple edges are possible between the same 2 vertices
  - Navigate through edges in either direction
- Properties

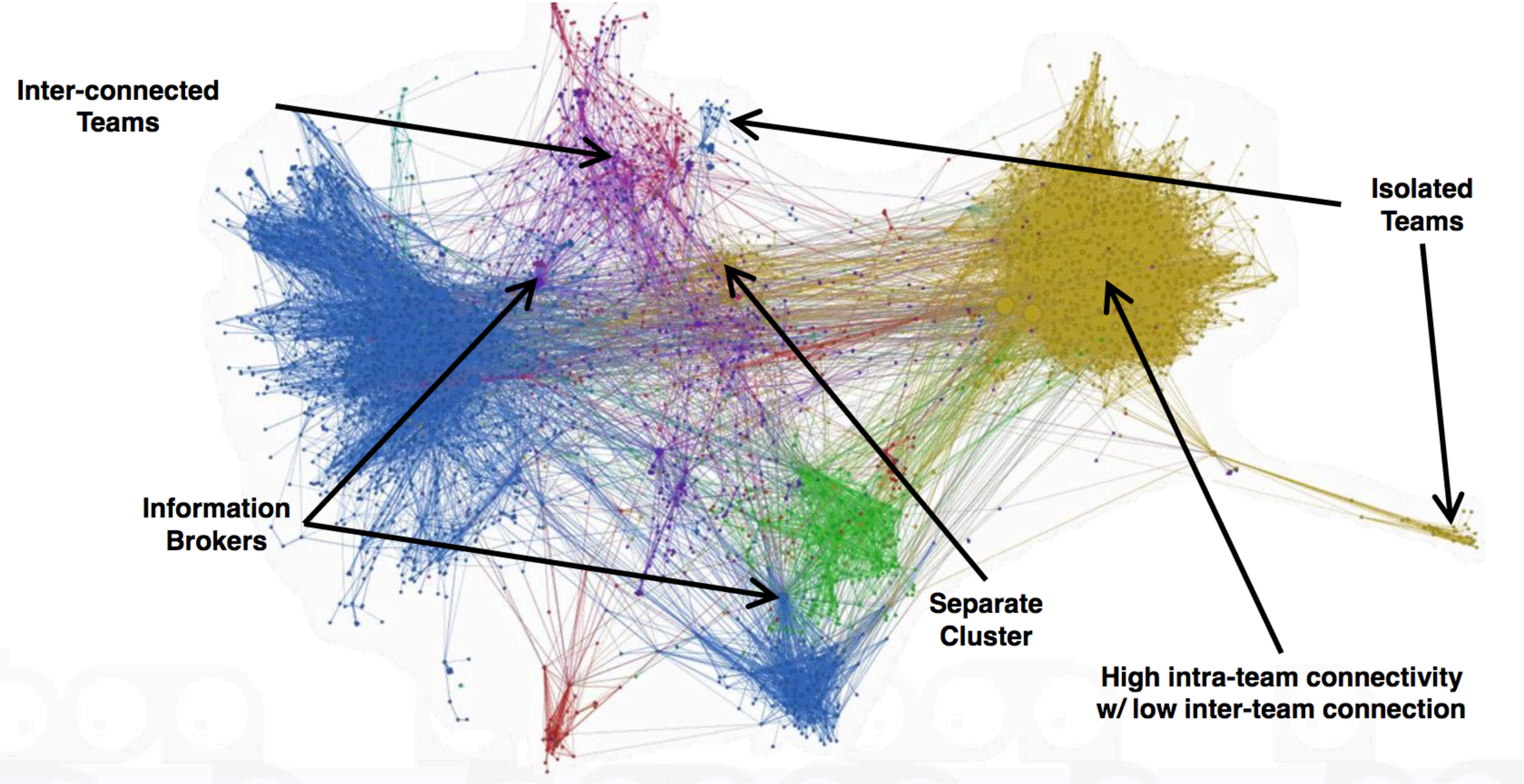


Properties

- Vertices
- Edges
- **Properties ←**
  - Additional metadata for vertex or edge
  - Key-value pairs
  - Values can be singular or multiple



Engagement Analytics



Inter-connected Teams

Isolated Teams

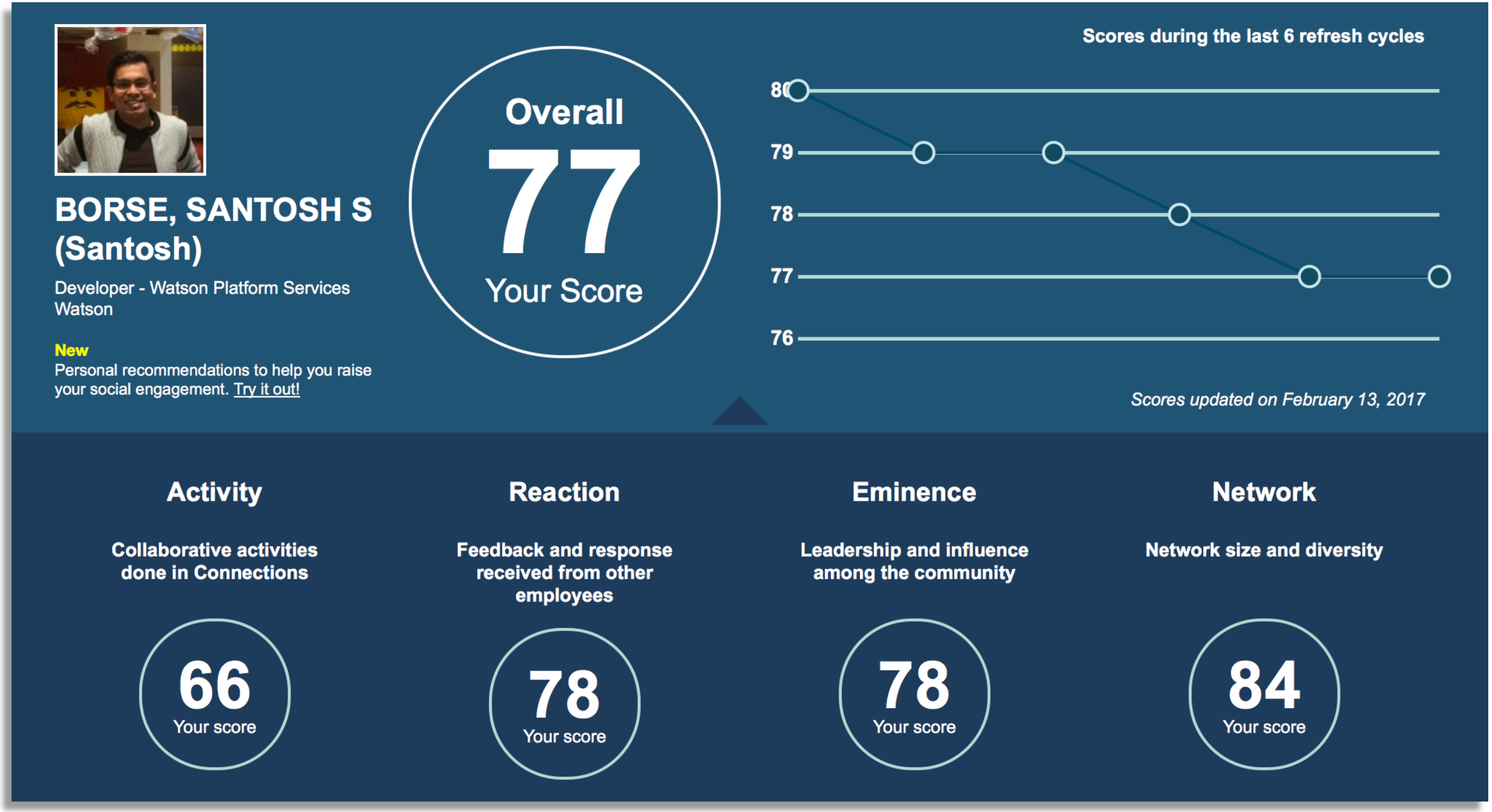
Information Brokers

Separate Cluster

High intra-team connectivity w/ low inter-team connection

Image: © IBM Corporation

Personal Dashboard



BORSE, SANTOSH S (Santosh)

Developer - Watson Platform Services
Watson

New
Personal recommendations to help you raise your social engagement. Try it out!

Overall
77
Your Score

Scores during the last 6 refresh cycles

Scores updated on February 13, 2017

Activity

Collaborative activities done in Connections

66
Your score

Reaction

Feedback and response received from other employees

78
Your score

Eminence

Leadership and influence among the community

78
Your score

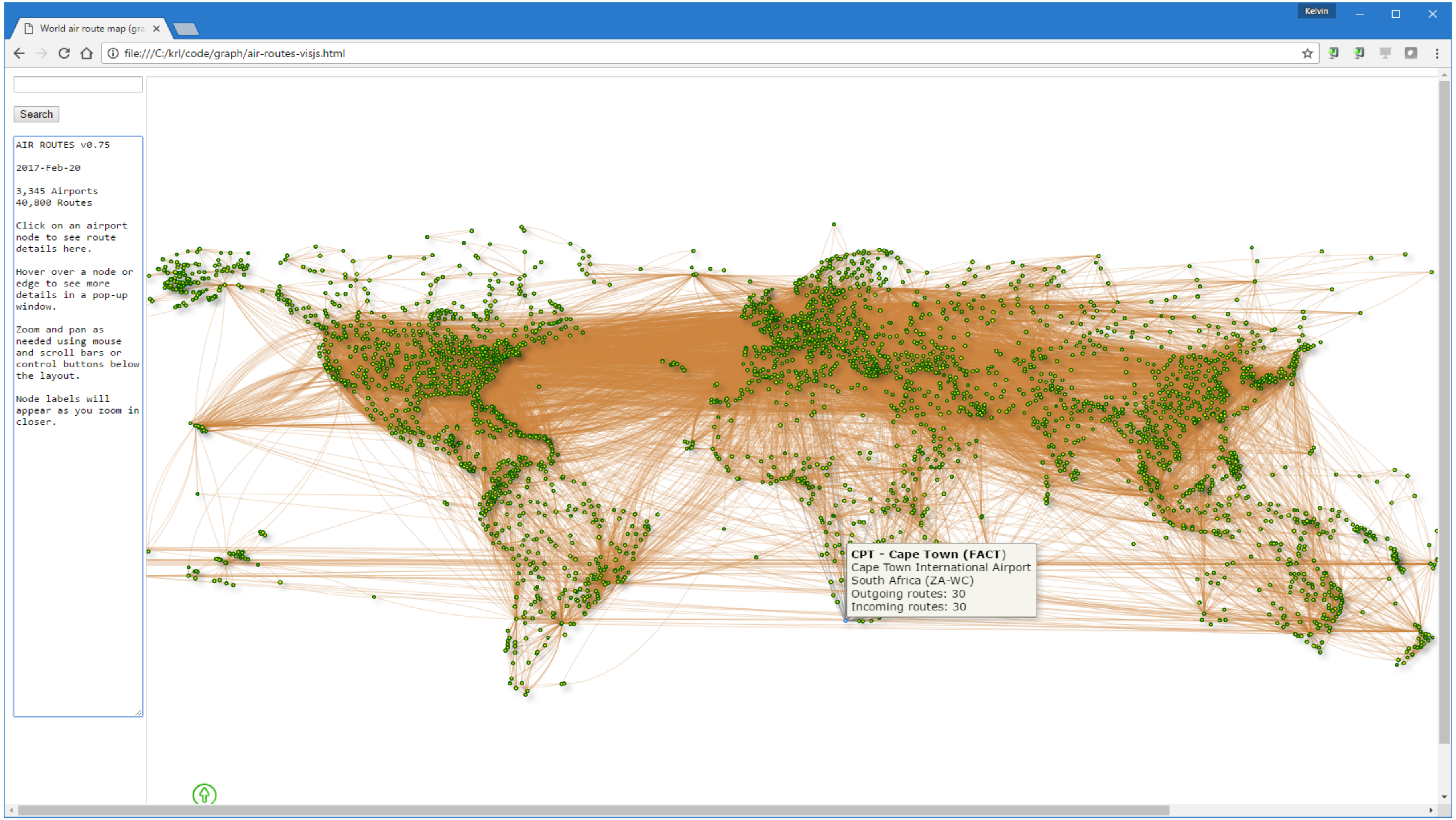
Network

Network size and diversity

84
Your score

Image: © IBM Corporation

Airline Routing



```
AIR ROUTES v0.75

2017-Feb-20

3,345 Airports
40,800 Routes

Click on an airport
node to see route
details here.

Hover over a node or
edge to see more
details in a pop-up
window.

Zoom and pan as
needed using mouse
and scroll bars or
control buttons below
the layout.

Node labels will
appear as you zoom in
closer.
```

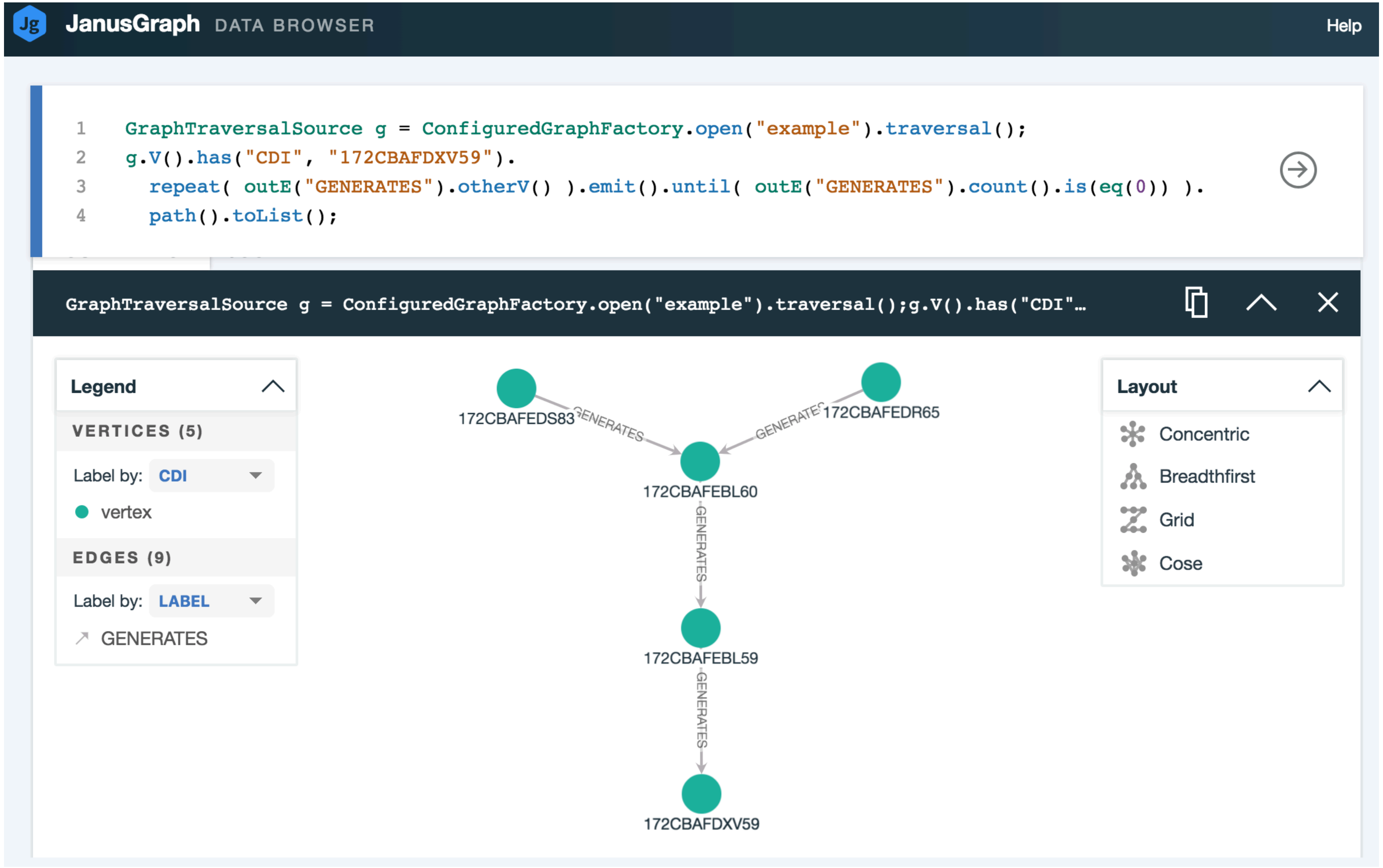
CPT - Cape Town (FACT)
Cape Town International Airport
South Africa (ZA-WC)
Outgoing routes: 30
Incoming routes: 30

Image: Map by Kelvin Lawrence (ALv2)
https://github.com/krlawrence/graph

Cloud Databases

OPEN SOURCE SUMMIT

China 2019



JanusGraph DATA BROWSER

Help

```
GraphTraversalSource g = ConfiguredGraphFactory.open("example").traversal();
g.V().has("CDI", "172CBAFDXV59").
  repeat( outE("GENERATES").otherV() ).emit().until( outE("GENERATES").count().is(eq(0)) ).
  path().toList();
```

```
GraphTraversalSource g = ConfiguredGraphFactory.open("example").traversal();g.V().has("CDI"…
```

Legend

VERTICES (5)

Label by: CDI

● vertex

EDGES (9)

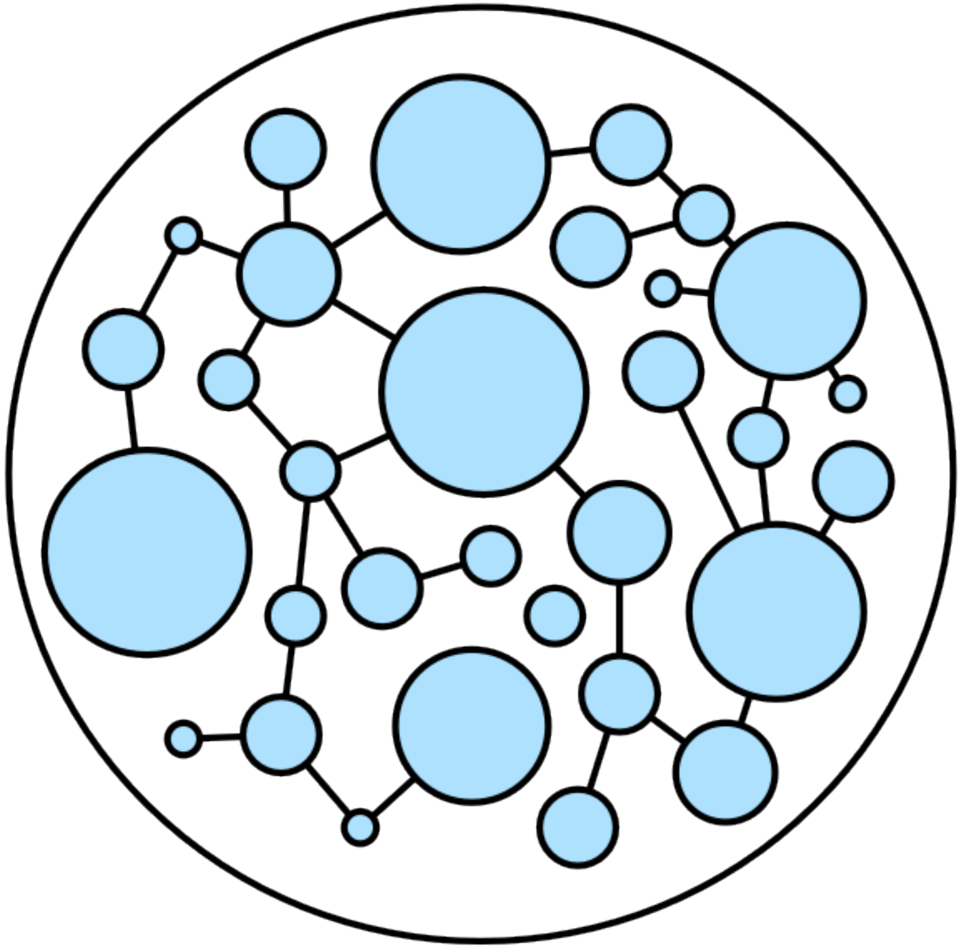
Label by: LABEL

↗ GENERATES

Layout

- Concentric
- Breadthfirst
- Grid
- Cose

Image: © IBM Corporation



Where Do You See Graphs?

Image: Graph Globe (ALv2)
https://tinkerpop.apache.org

Agenda

- Graph Use Cases
- **Open Source Graph Evolution ←**
- Future Directions

Graph Framework

- Apache TinkerPop
  - Vendor-neutral graph computing framework
  - Created in 2009 by Dr. Marko A. Rodriguez
  - https://tinkerpop.apache.org

- Vendor Implementations
  - Neo4j
  - Datastax Enterprise Graph
  - Microsoft Azure Cosmos DB
  - Amazon Neptune

Image: Gremlin Running (ALv2)
https://tinkerpop.apache.org

Gremlin Traversals

- Traversal
  - A walk through the graph from one vertex to another along a connected edge

- Gremlin
  - Graph domain-specific language for traversals in TinkerPop-compliant systems

Image: Keep on Traversin' (ALv2)
https://tinkerpop.apache.org

Traversal Example

- Which projects did Marko’s colleagues create with others?

```
g.V().
   has(‘name’, ‘marko’).
   out(‘knows’).
   outE(‘created’).
   has(‘weight’, lt(1.0))
   inV().
   values(‘name’)
```

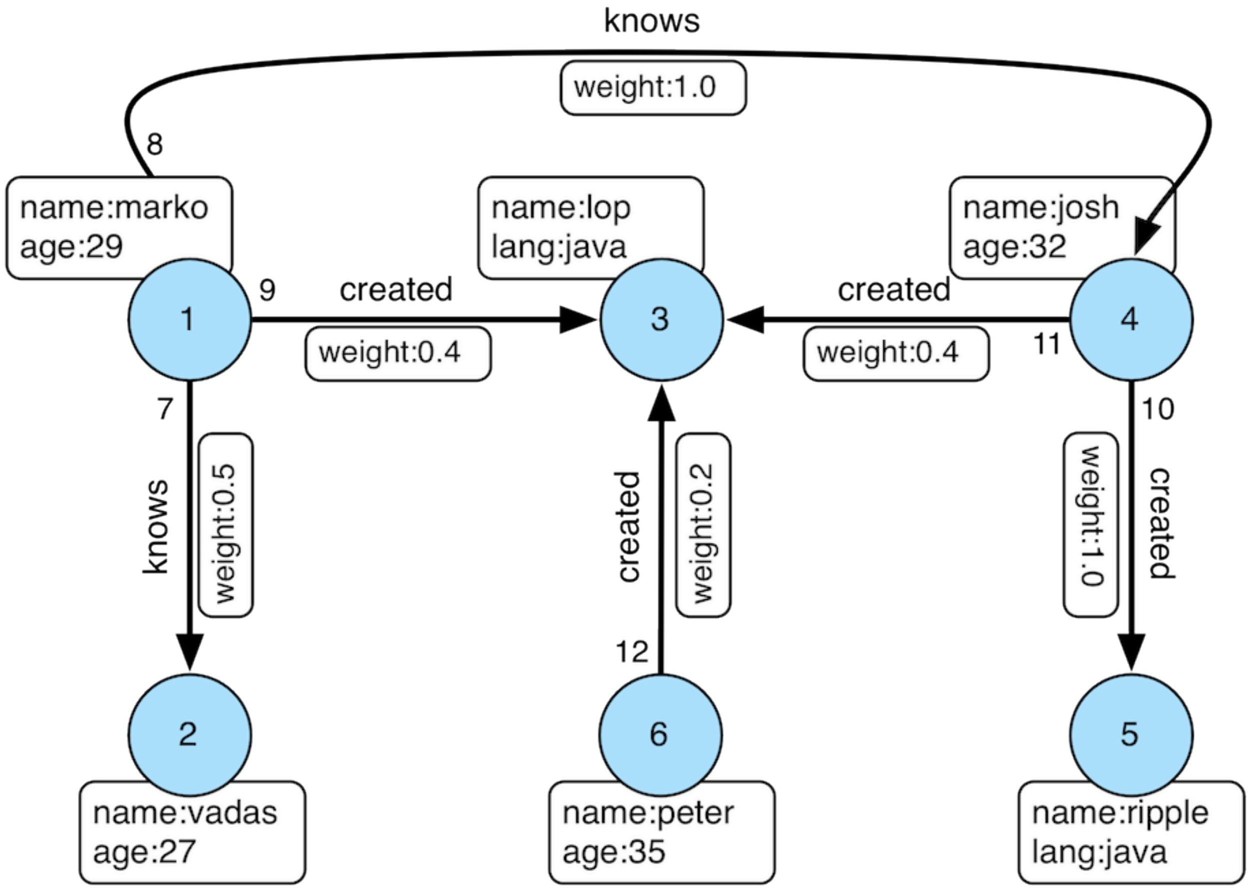


Image: TinkerPop Modern (ALv2)
https://tinkerpop.apache.org

Traversal Example

OPEN SOURCE SUMMIT

China 2019

- Which projects did Marko’s colleagues create with others?

```
g.V().
  has(‘name’, ‘marko’).
  out(‘knows’)
  outE(‘created’).
  has(‘weight’, lt(1.0))
  inV().
  values(‘name’)
```

- “Select all vertices”

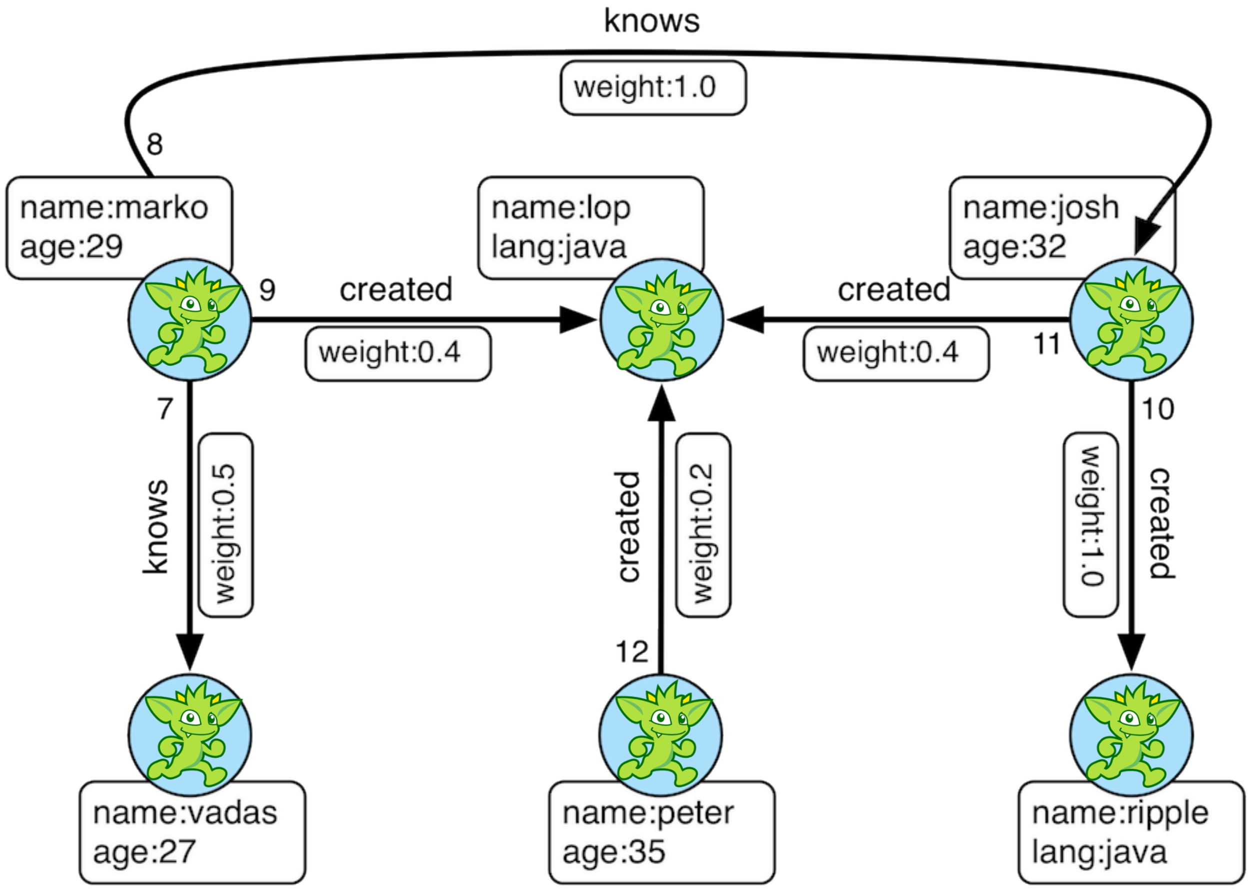


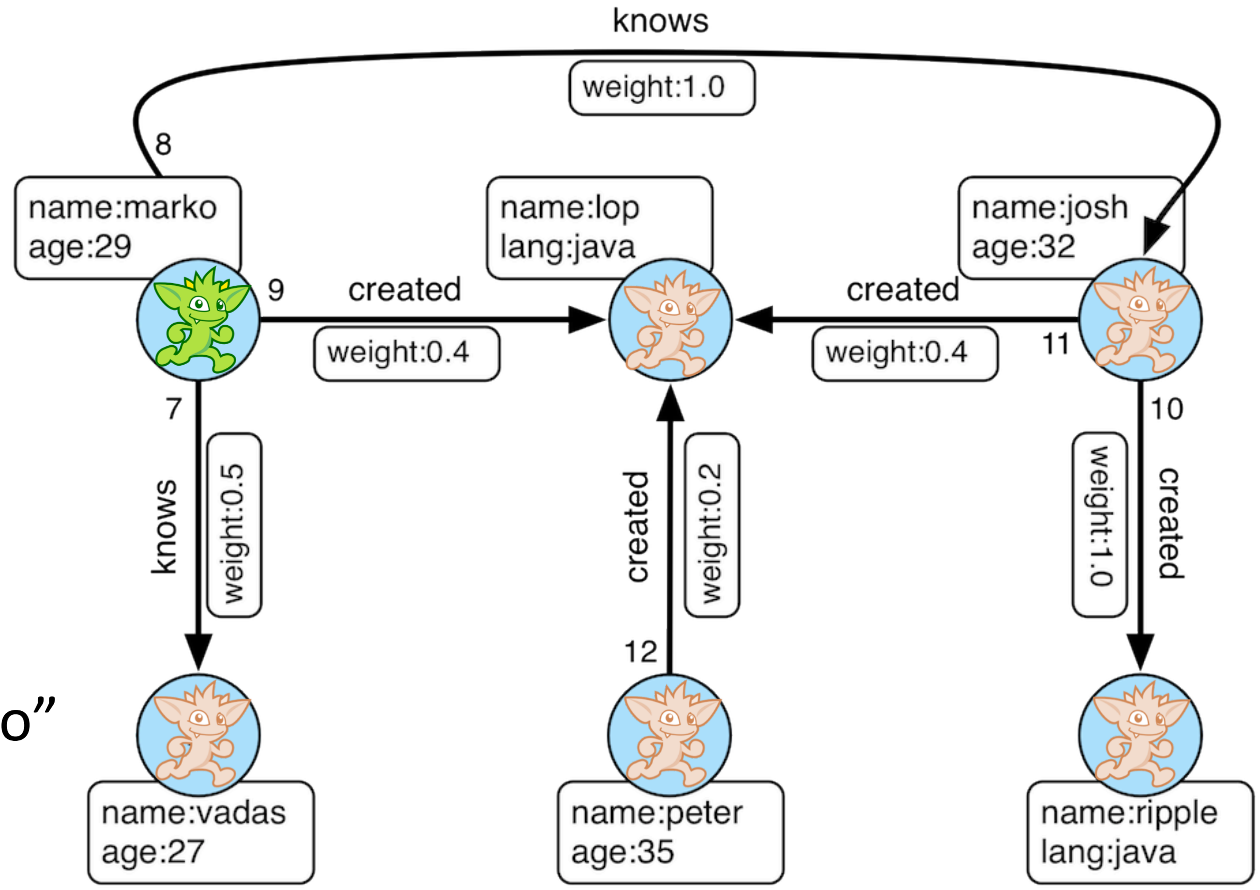
Image: TinkerPop Modern and Gremlin Running (ALv2)
https://tinkerpop.apache.org

Traversal Example

OPEN SOURCE SUMMIT
China 2019

- Which projects did Marko's colleagues create with others?

```
g.V().
   has('name', 'marko').
   out('knows')
   outE('created').
   has('weight', lt(1.0)).
   inV().
   values('name')
```



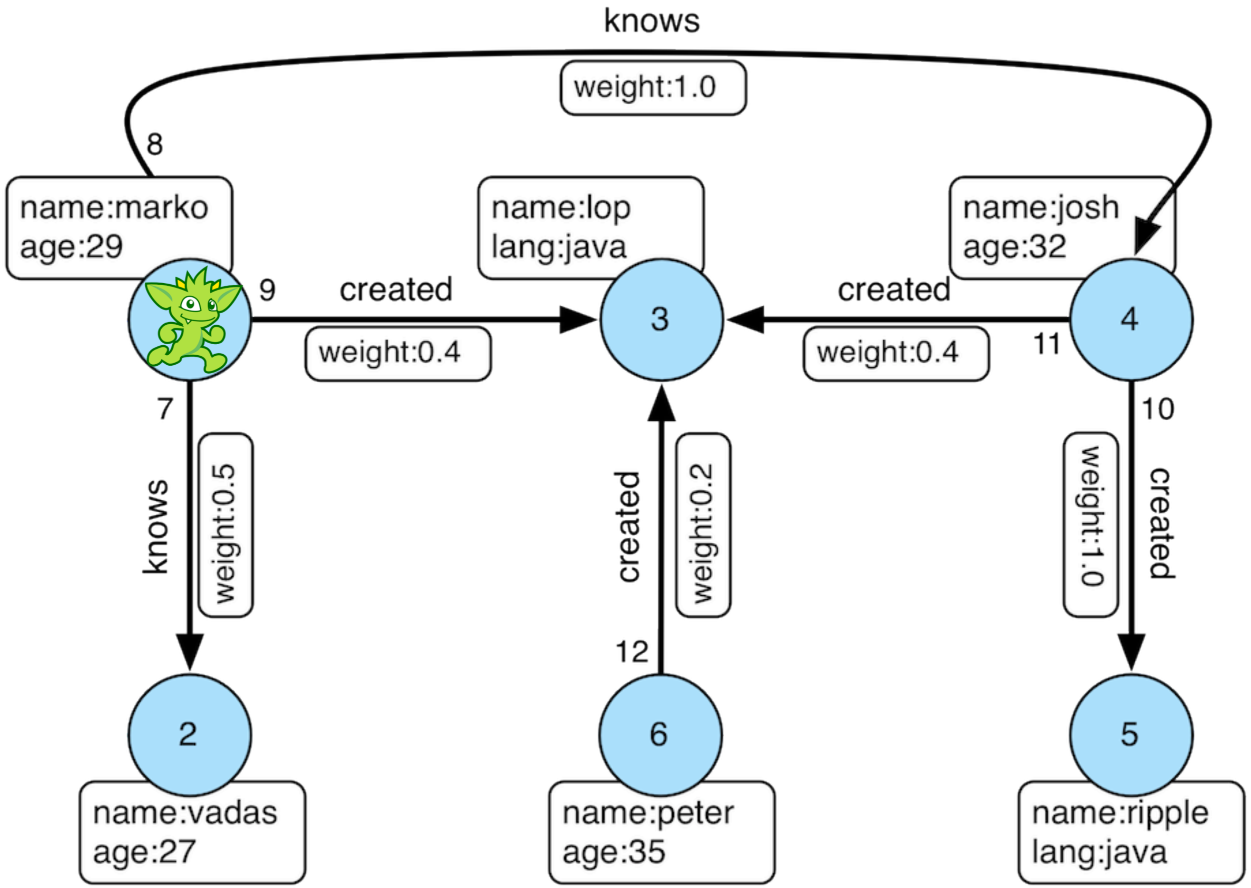
- "Filter vertices where 'name' is Marko"
- Global vertex scan

Image: TinkerPop Modern and Gremlin Running (ALv2)
https://tinkerpop.apache.org

Traversal Example

- Which projects did Marko’s colleagues create with others?

```
g.V().
   has(‘name’, ‘marko’).
   out(‘knows’)
   outE(‘created’).
   has(‘weight’, lt(1.0))
   inV().
   values(‘name’)
```



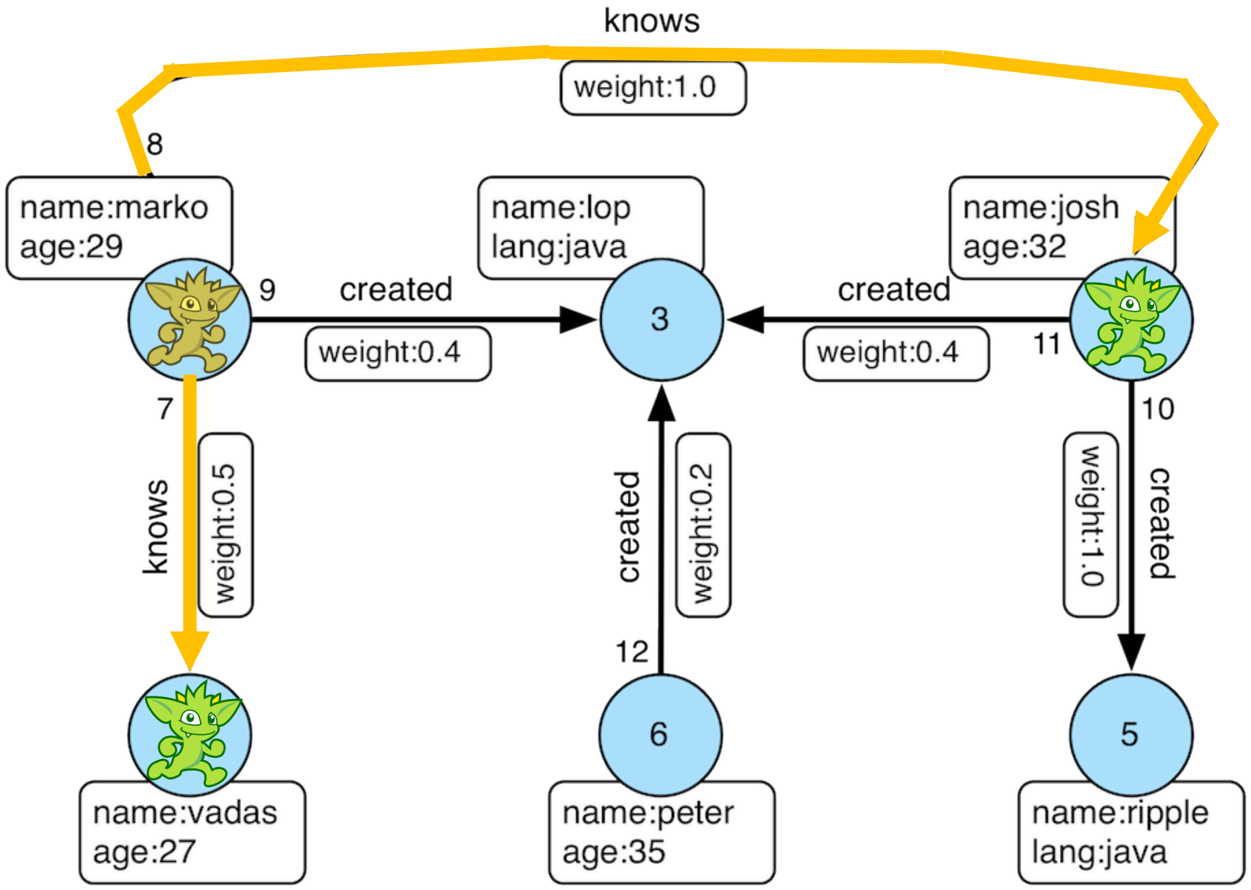
- “Utilize index on ‘name’ property”
- Leverage indexes for performance

Image: TinkerPop Modern and Gremlin Running (ALv2)
https://tinkerpop.apache.org

Traversal Example

- Which projects did Marko’s colleagues create with others?

```
g.V().
  has(‘name’, ‘marko’).
  out(‘knows’)
  outE(‘created’).
  has(‘weight’, lt(1.0))
  inV().
  values(‘name’)
```



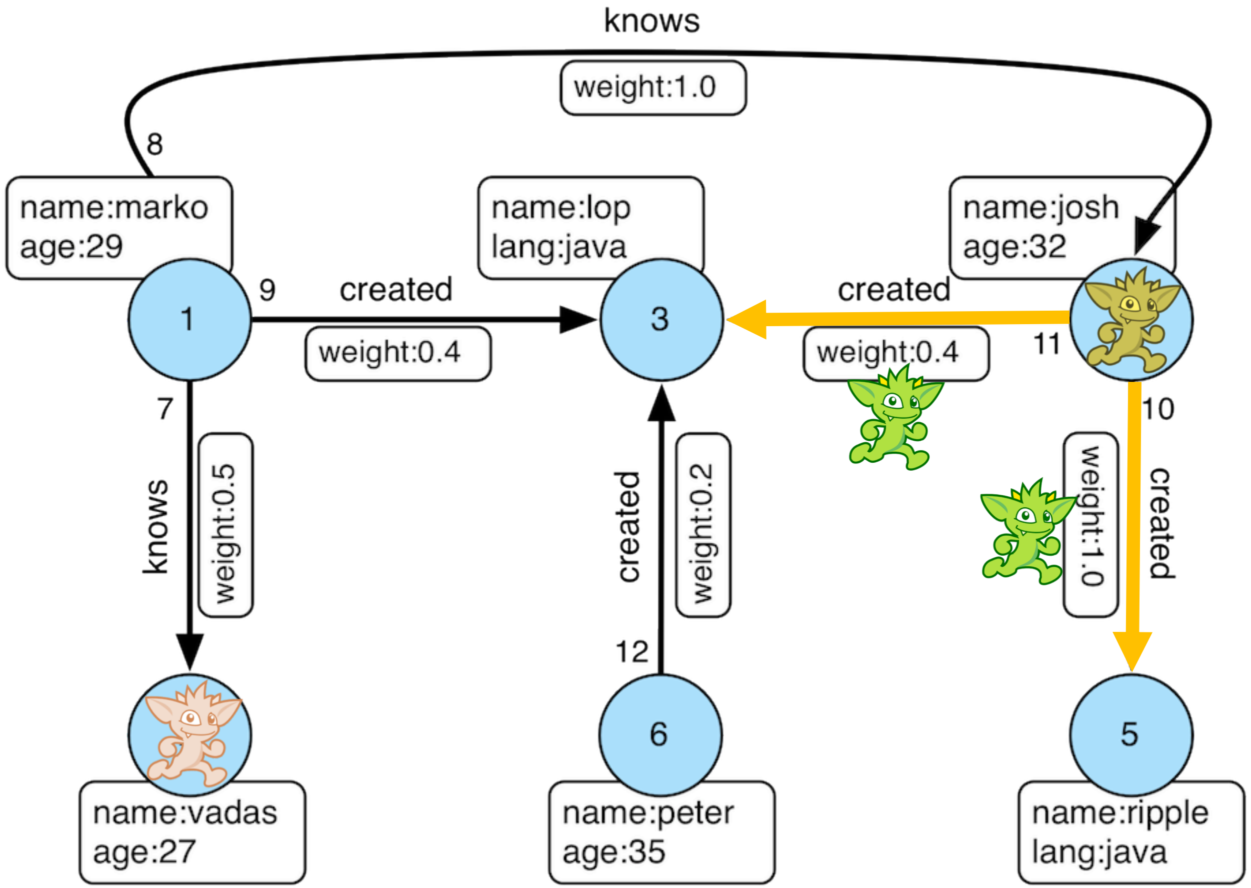
- “Follow outward on ‘knows’ edges to adjacent vertices”

Image: TinkerPop Modern and Gremlin Running (ALv2)
https://tinkerpop.apache.org

Traversal Example

- Which projects did Marko’s colleagues create with others?

```
g.V().
  has(‘name’, ‘marko’).
  out(‘knows’)
  outE(‘created’).
  has(‘weight’, lt(1.0))
  inV().
  values(‘name’)
```



- “Follow outward on ‘created’ edges”

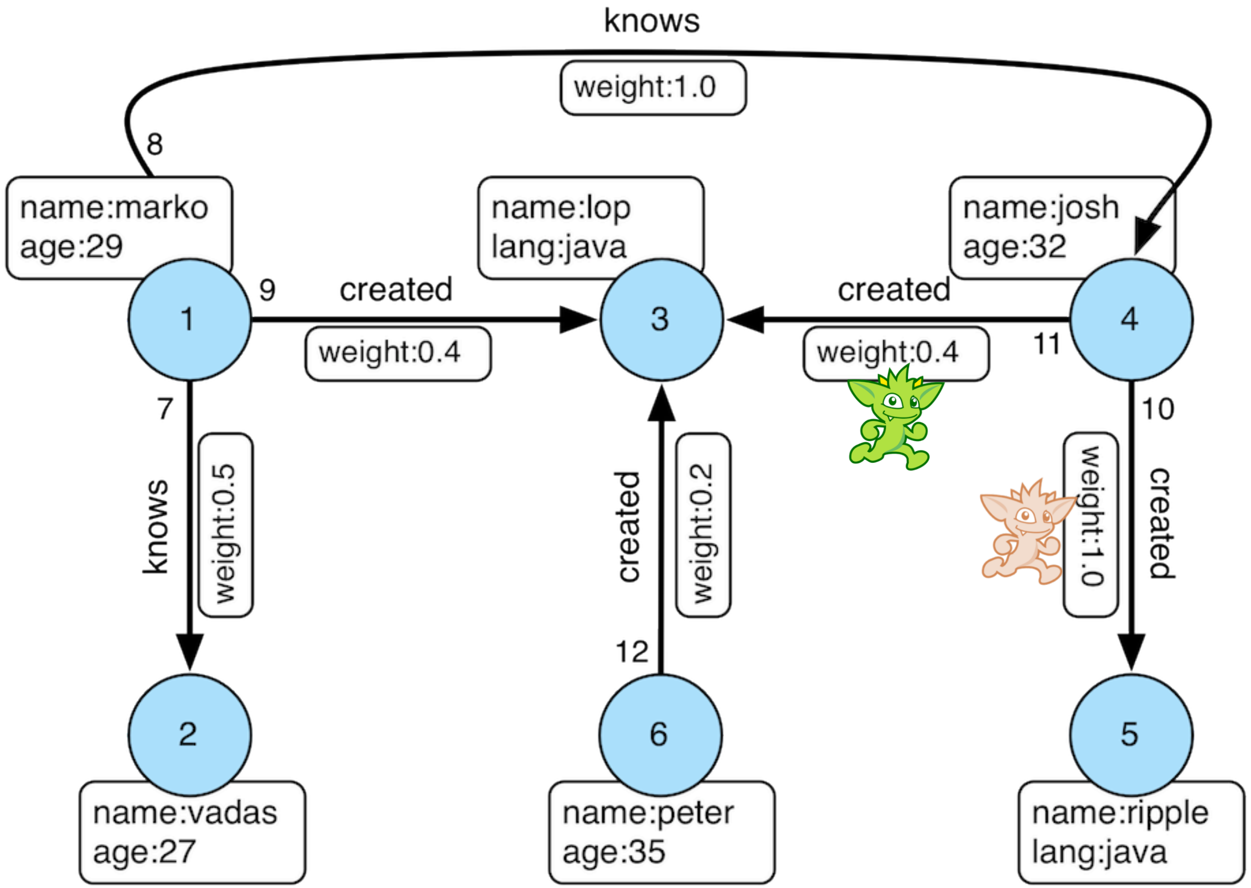
Image: TinkerPop Modern and Gremlin Running (ALv2)
https://tinkerpop.apache.org

Traversal Example

OPEN SOURCE SUMMIT
China 2019

- Which projects did Marko’s colleagues create with others?

```
g.V().
  has(‘name’, ‘marko’).
  out(‘knows’)
  outE(‘created’).
  has(‘weight’, lt(1.0))
  inV().
  values(‘name’)
```



- “Filter edges where ‘weight’ is less than 1.0”

Image: TinkerPop Modern and Gremlin Running (ALv2)
https://tinkerpop.apache.org

Traversal Example

- Which projects did Marko's colleagues create with others?

```
g.V().
   has('name', 'marko').
   out('knows')
   outE('created').
   has('weight', lt(1.0))
   inV().
   values('name')
```

- "Follow inward to the target vertex"

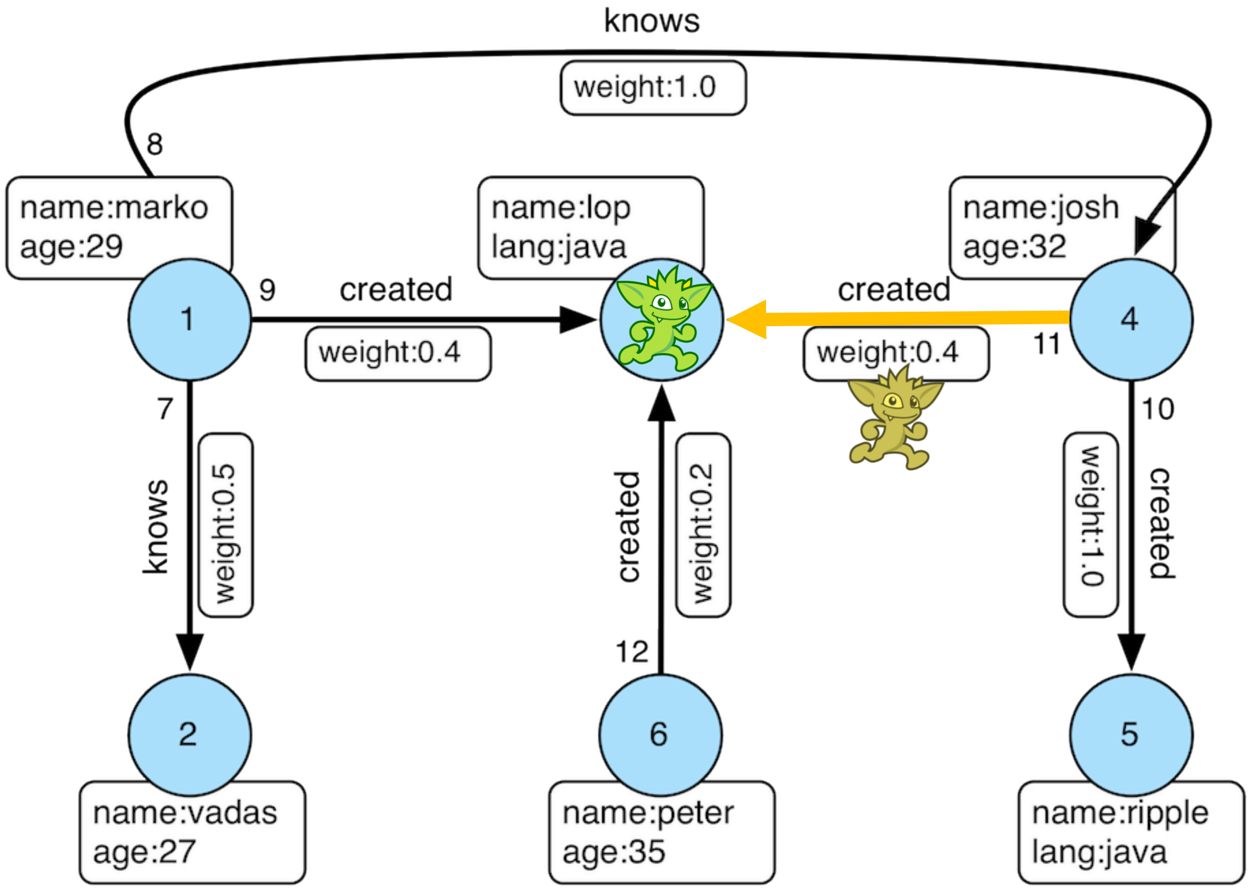


Image: TinkerPop Modern and Gremlin Running (ALv2)
https://tinkerpop.apache.org

Traversal Example

- Which projects did Marko’s colleagues create with others?

```
g.V().
  has(‘name’, ‘marko’).
  out(‘knows’)
  outE(‘created’).
  has(‘weight’, lt(1.0))
  inV().
  values(‘name’)
```

- “Emit the value of ‘name’ property”
- lop

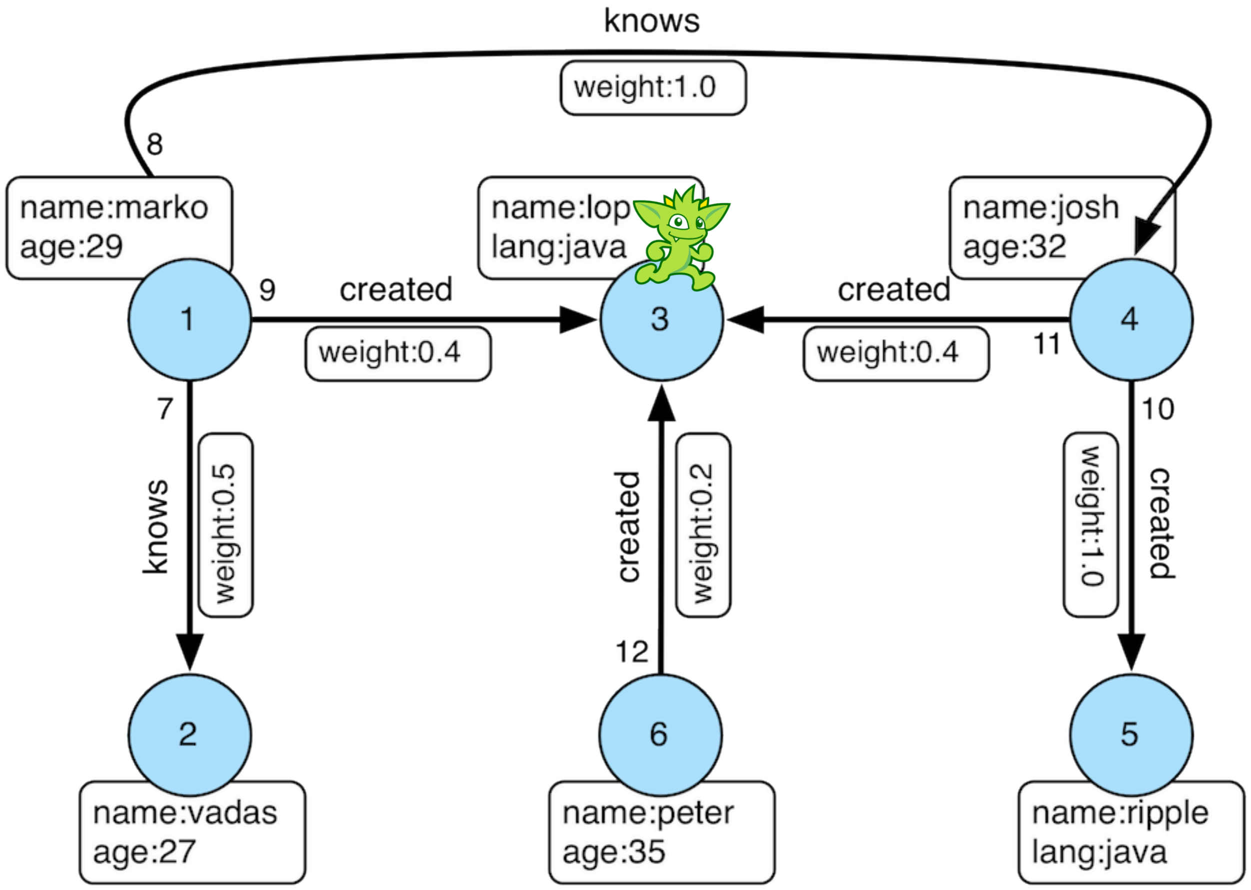


Image: TinkerPop Modern and Gremlin Running (ALv2)
https://tinkerpop.apache.org

Graph Database

- Titan DB
  - Open source scalable graph database
  - Created by Dr. Matthias Broecheler
  - Acquired by Datastax in 2015
  - Titan 1.0 released
  - Community left hanging

Image: Traversal Strategy (ALv2)
https://tinkerpop.apache.org

Graph Database

- JanusGraph
  - Fork of Titan DB hosted at The Linux Foundation
  - Reconnect open source community
  - Embrace open governance
  - https://janusgraph.org

- Diverse Community
  - Founders: Expero, Google, Grakn, Hortonworks, IBM
  - Comcast, Goldman Sachs, Netflix, Uber, VMWare, and many others

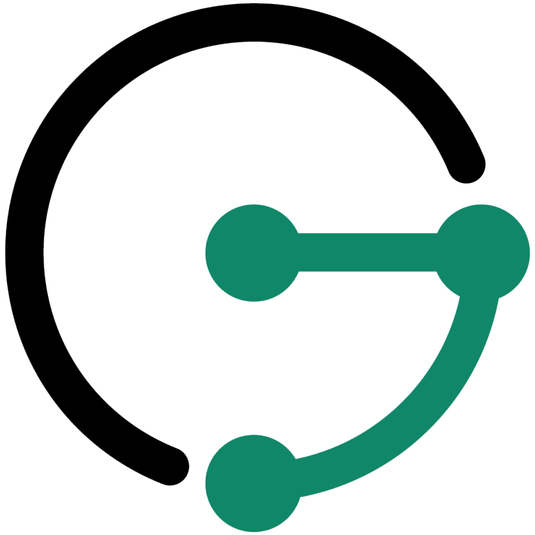


Image: JanusGraph (CC-BY-4.0)
https://janusgraph.org

Built with JanusGraph

- Apache Atlas
  - Metadata management for governance
- Egeria (ODPi, Linux Foundation)
  - Open metadata and governance
- Eclipse Keti
  - Access control service to protect RESTful APIs
- Exakat
  - PHP static analysis
- Open Network Automation Platform (Linux Foundation)
  - Automation and orchestration for software-defined networks
- Windup by Redhat
  - Application migration and assessment

JanusGraph Architecture

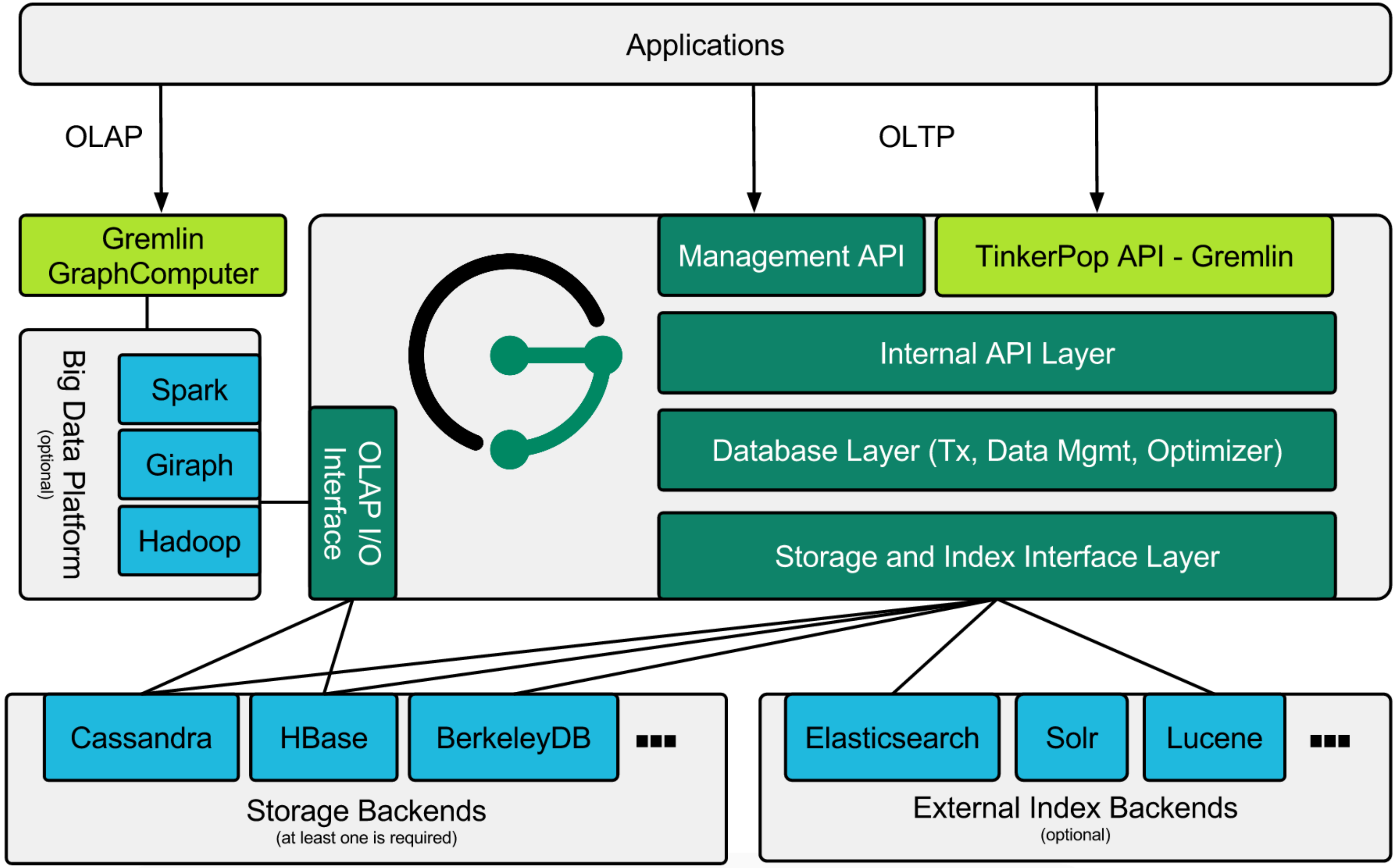


Image: Architecture Layer Diagram (CC-BY-4.0)
https://janusgraph.org

Key Benefits

- Apache-licensed open source
  - Permissive software licensing
  - Free to use anywhere
- Open governance community
  - No single vendor control or lock-in
  - Open collaboration and development
- Pluggable storage and indexing
  - Leverage existing technology and skills
  - Compare performance characteristics

Open Source at IBM

IBM Developer
Topics
Community
More open source at IBM
Learn about our approach to open source
Show me all IBM projects on GitHub

Open source at IBM

IBM's home for open source code, community, and culture

For the past 20 years, IBM has invested significantly in open source code, communities, and governance. Learn where we partner, how you can join us, and how you can create an open enterprise.

https://developer.ibm.com/open

- Code Patterns
  - https://developer.ibm.com/patterns/develop-graph-database-app-using-janusgraph/
- JanusGraph Utilities
  - https://github.com/IBM/janusgraph-utils

- Getting Started blog series
  - https://developer.ibm.com/dwblog/2018/whats-janus-graph-learning-deployment/
- Tips and Tricks blog series
  - https://developer.ibm.com/articles/janusgraph-tips-and-tricks-pt-1/

Agenda

- Graph Use Cases
- Open Source Graph Evolution
- **Future Directions ←**

Project Directions

- Diversify client driver support
  - .NET
  - Python
  - Javascript

- Platform support
  - Windows
  - Docker
  - Apache Ambari

- Administration Console
- Operations tooling, monitoring

- Diversify backend storage support
  - In-memory
  - FoundationDB
  - Couchbase

- Benchmarking
- ETL, bulkloading, serialization
- Query profiling, traversal optimization

- Apache TinkerPop 4
- Property Graph Schema Working Group

Thank You!

Jason Plurad | pluradj@us.ibm.com
@pluradj GitHub | LinkedIn | Twitter
Apache TinkerPop | JanusGraph