Community-Driven Graphs with JanusGraph

Jing Chen He • jinghe@us.ibm.com • Apache HBase PMC • JanusGraph TSC
Jason Plurad • pluradj@us.ibm.com • Apache TinkerPop PMC • JanusGraph TSC
HBaseCon West 2017 • June 12, 2017
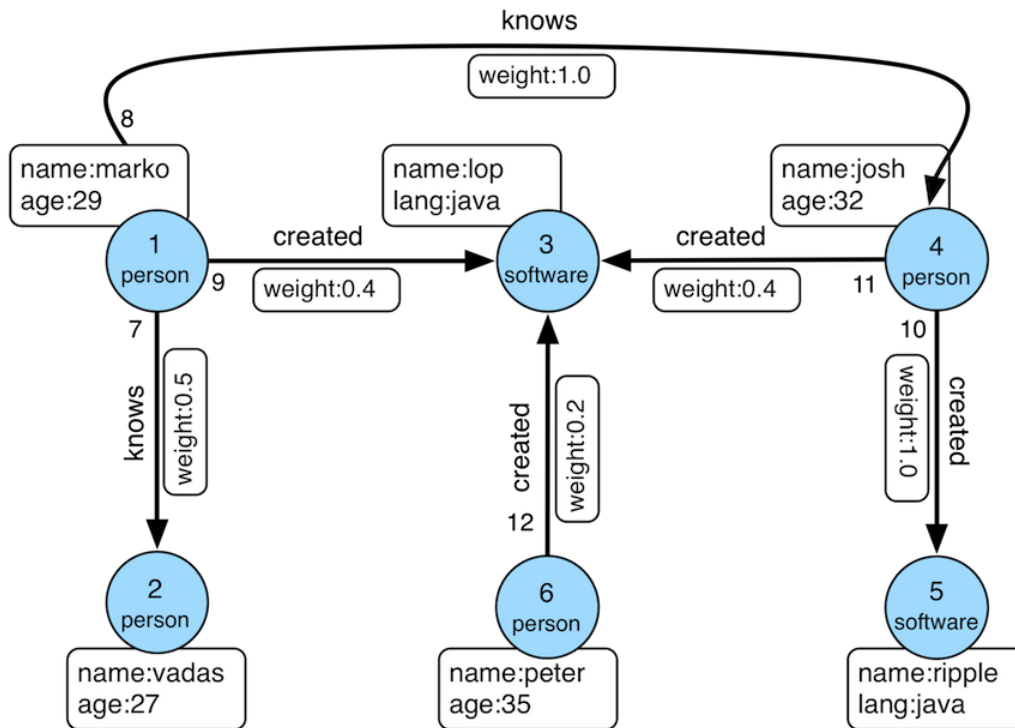
IBM

Agenda

➢Property Graphs

➢Graph Community

➢Introduction to JanusGraph

➢JanusGraph with HBase

#HBaseCon

Graph

- Born for relationship!

- Intuitive modeling

- Expressive querying

- Native analysis

  – *Vertex*
  – *Edge*
  – *Properties*

https://tinkerpop.apache.org/docs/3.2.4/reference/#intro
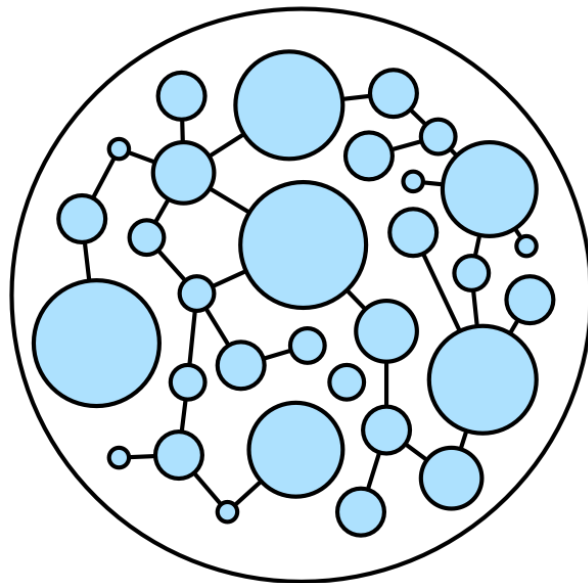
#HBaseCon

Graph Data Use Cases

- Social network analysis

- Configuration management database

- Master data management

- Recommendation engines

- Knowledge graphs

- Internet of things

- Cybersecurity attack analysis

#HBaseCon

Apache TinkerPop

- Open source, vendor-agnostic, graph computing framework

- Gremlin graph traversal language

Graph Computing

Structure + Process
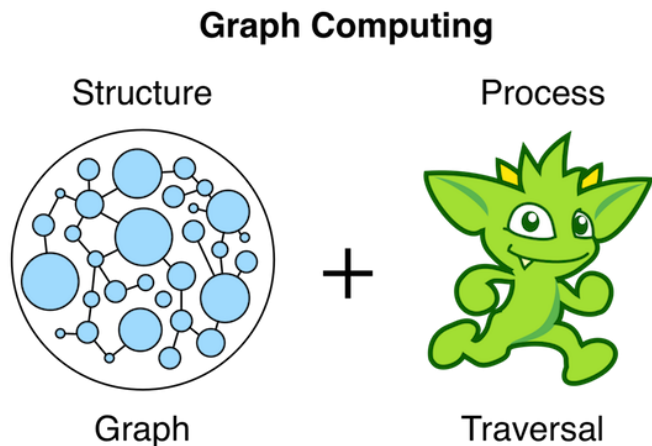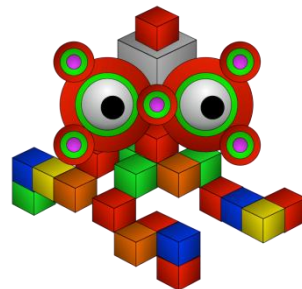
Graph + Traversal

Apache TinkerPop™	
Maintainer	Apache Software Foundation
License	Apache
Latest Release	3.2.4 February 2017
https://tinkerpop.apache.org	

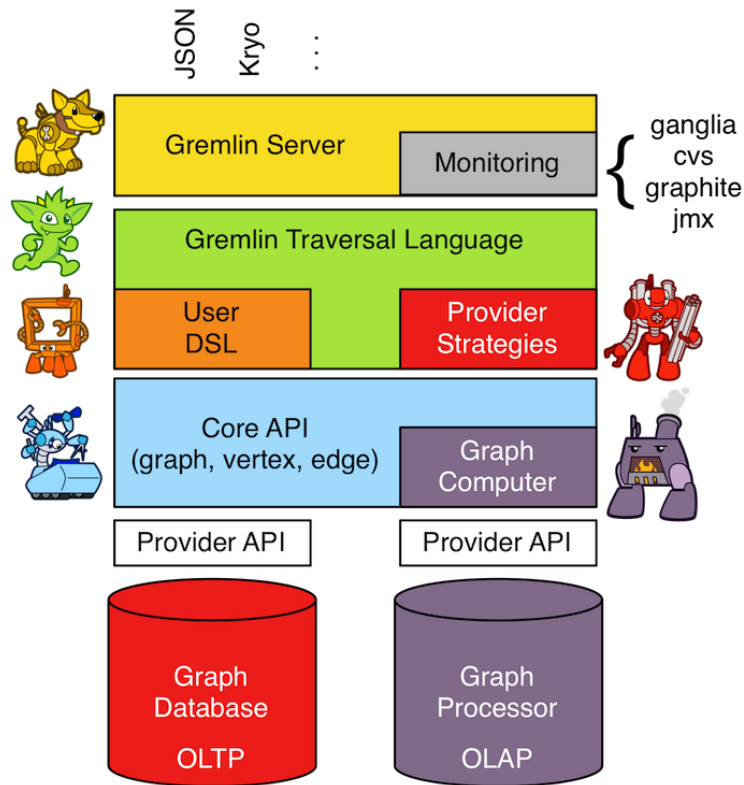#HBaseCon

Gremlin Graph Traversal Language


```
// What are the names of the managers in
//   the management chain going from Gremlin to the CEO?
g.V().has("name","gremlin").
  repeat(in("manages")).until(has("title","ceo")).
  path().by("name")
```
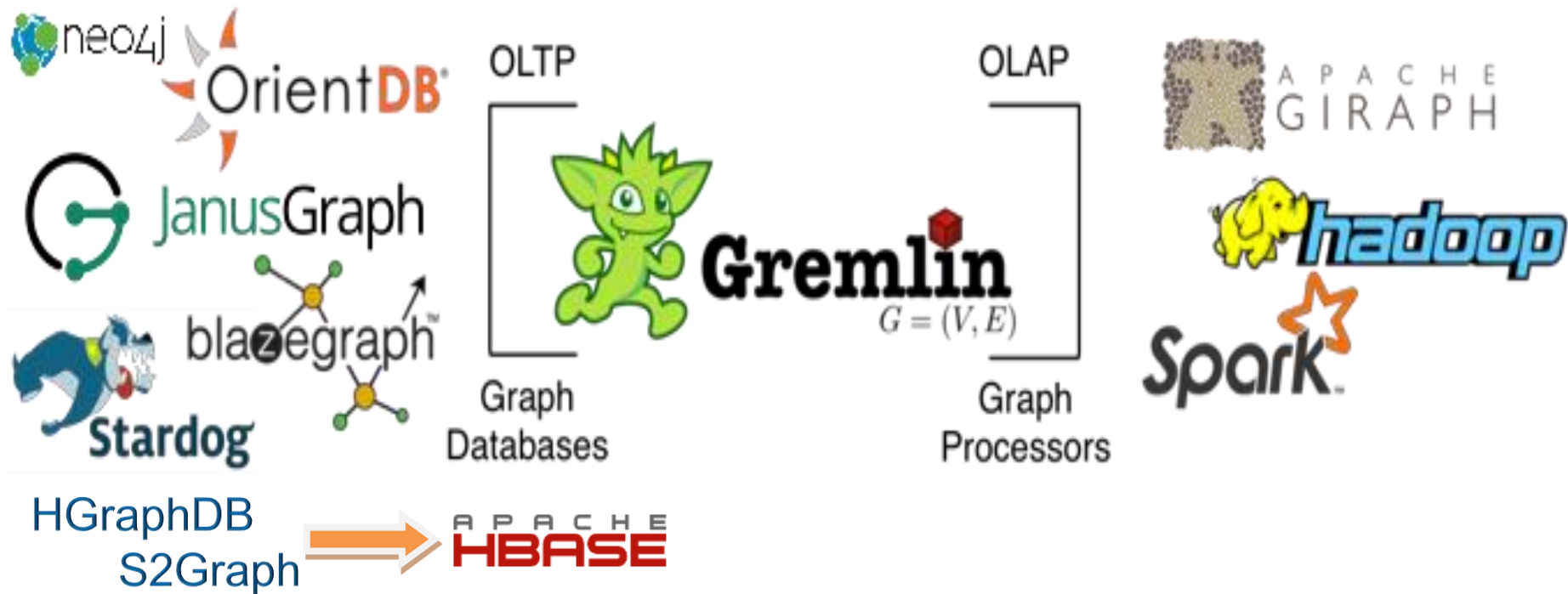
https://tinkerpop.apache.org/gremlin.html

#HBaseCon

TinkerPop Stack

https://tinkerpop.apache.org/docs/3.2.4/reference/#_graph_system_integration

#HBaseCon

Graph Landscape

OLTP — Gremlin $G = (V, E)$ — OLAP

Graph Databases — Graph Processors

HGraphDB
S2Graph → APACHE HBASE

https://tinkerpop.apache.org/gremlin.html#oltp-and-olap-traversals

#HBaseCon

JanusGraph

- Scalable graph database distributed on multi-machine clusters with pluggable storage and indexing

- Fully-compliant with Apache TinkerPop graph computing framework

- **Vendor-neutral, open community with open governance**
  - Founding members: Expero, Google, GRAKN.AI, Hortonworks, IBM
  - Latest members: Amazon, Netflix, Orchestral Developments, Uber
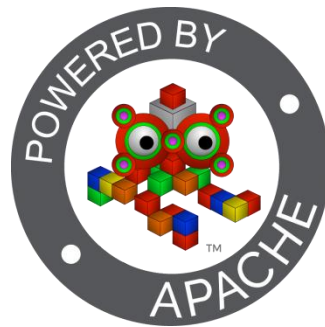
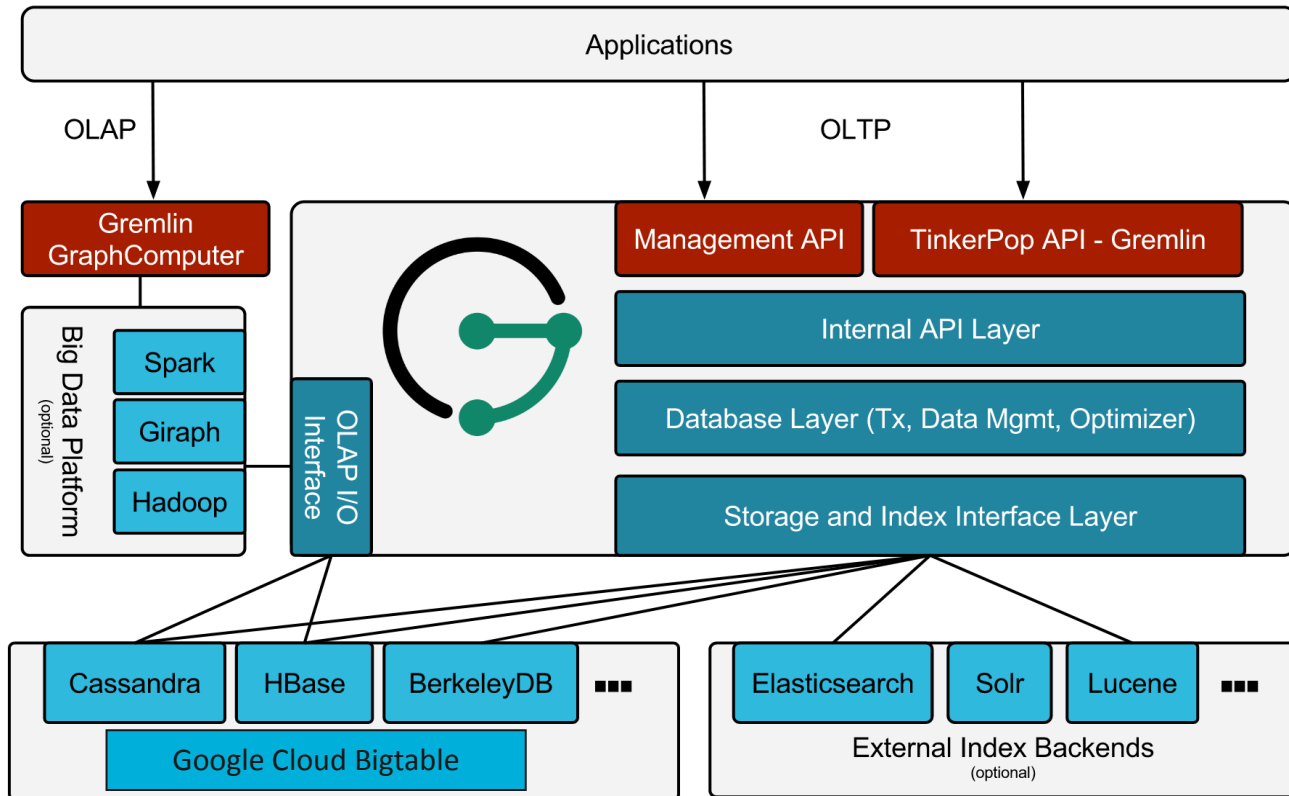| JanusGraph™	
Maintainer	Linux Foundation
License	Apache
Latest Release	0.1.0 April 2017
https://janusgraph.org	

#HBaseCon

JanusGraph Architecture

#HBaseCon

http://docs.janusgraph.org/latest/arch-overview.html

JanusGraph Storage Model

Adjacency list
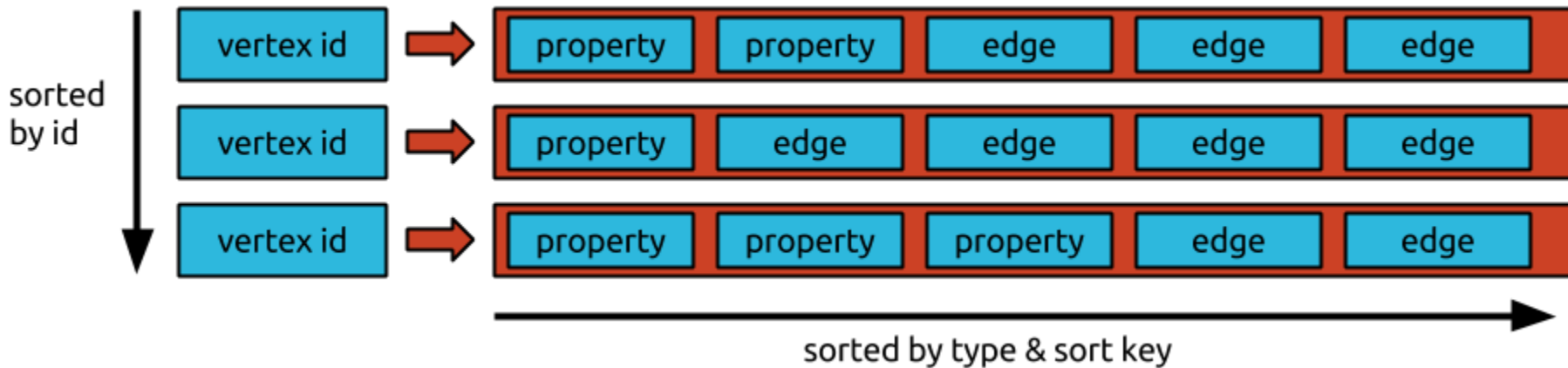
sorted by id

vertex id →	property	property	edge	edge	edge
vertex id →	property	edge	edge	edge	edge
vertex id →	property	property	property	edge	edge

sorted by type & sort key

#HBaseCon

JanusGraph Storage Model

Edge

| label id + direction | sort key | adjacent vertex id | edge id | signature key | other properties |

◄─────────────── column ───────────────■──── value ────■

Property

| key id | | property id | property value |

#HBaseCon
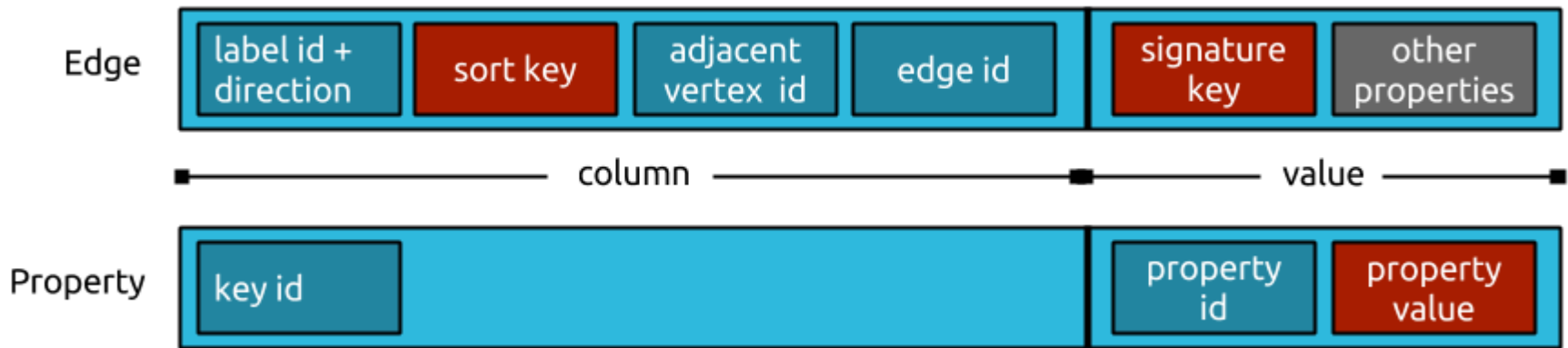
JanusGraph with HBase

- HBase – Perfect Storage Backend for JanusGraph

  - Big enough for your biggest graph!
  - The storage model
  - Read and write speed
  - Scalability and partitioning
  - Strong consistency
  - Tight integration with Hadoop Ecosystem
  - Great open community!

http://docs.janusgraph.org/latest/hbase.html
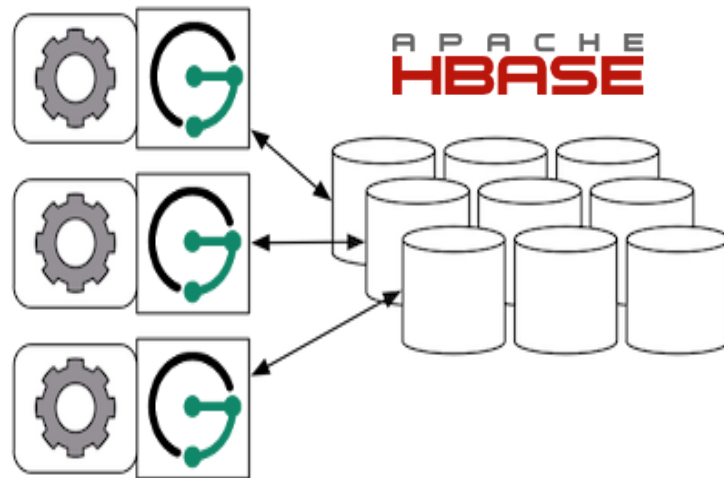
#HBaseCon

JanusGraph with HBase

- HBase – Perfect Storage Backend for JanusGraph

  - Simple configuration!

    - conf/janusgraph-hbase-solr.properties

    - ✓ storage.backend=hbase
    - ✓ storage.hostname=zookeeper-host1,zookeeper-host2,zookeeper-host3
    - ✓ storage.hbase.table=janusgraph
    - ✓ storage.hbase.ext.zookeeper.znode.parent=/hbase   *Optional*
    - ✓ storage.hbase.ext.hbase.zookeeper.property.clientPort=2181   *Optional*

  - Just open your graph!

    - graph=JanusGraphFactory.open('conf/janusgraph-hbase-solr.properties')

JanusGraph with HBase

- HBase – Perfect Storage Backend for JanusGraph

  - Throw in an Index Backend for better performance
    - conf/janusgraph-hbase-solr.properties
    - ✓ index.search.backend=solr
    - ✓ index.search.solr.mode=cloud
    - ✓ index.search.solr.zookeeper-url=zookeeper-host1:2181/solr,zookeeper-host2:2181/solr,zookeeper-host3:2181/solr
    - ✓ index.search.solr.configset=janusgraph

#HBaseCon

JanusGraph with HBase

- HBase – Perfect Storage Backend for JanusGraph

  ➢ Look into more details

    - Stores to Column Families

      ✓ Edge store → e

      ✓ Index store → g

      ✓ ID store → i

      ✓ Transaction log store → l

      ✓ System property store → s

    - CF attributes can be set. E.g. compression, TTL.

JanusGraph with HBase
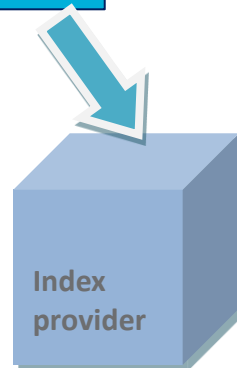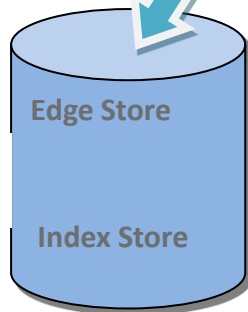
- HBase – Perfect Storage Backend for JanusGraph

  ➢ Look into more details

  g.V().has("name", "Alice").out("knows").out("knows").values("name")

#HBaseCon

JanusGraph with HBase

- HBase – Perfect Storage Backend for JanusGraph

  - Look into more details

    - ✓ A store (column family) is always specified.

    - ✓ Get or Multi Get

    - ✓ Batch to mutate

    - ✓ Key range scan

    - ✓ ColumnRangeFilter

    - ✓ ColumnPaginationFilter

  - HBase tuning

Edge Store

Index Store

APACHE
HBASE

#HBaseCon

JanusGraph Google Cloud Platform
with Google Cloud Bigtable

▪ Bigtable implements the HBase 1.0 client API

➢ Need the latest version of the bigtable-hbase-1.0 artifact.

✓ storage.backend=**hbase**

✓ storage.hbase.ext.hbase.client.connection.impl=

com.google.cloud.bigtable.hbase1_0.BigtableConnection

✓ storage.hbase.ext.google.bigtable.project.id=

<Google Cloud Platform project id>

✓ storage.hbase.ext.google.bigtable.instance.id=<Bigtable instance id>

#HBaseCon

Thank you!

Open *by design* ™