



Graph Databases

What makes them Different ?

Darren Wood

Chief Architect, InfiniteGraph

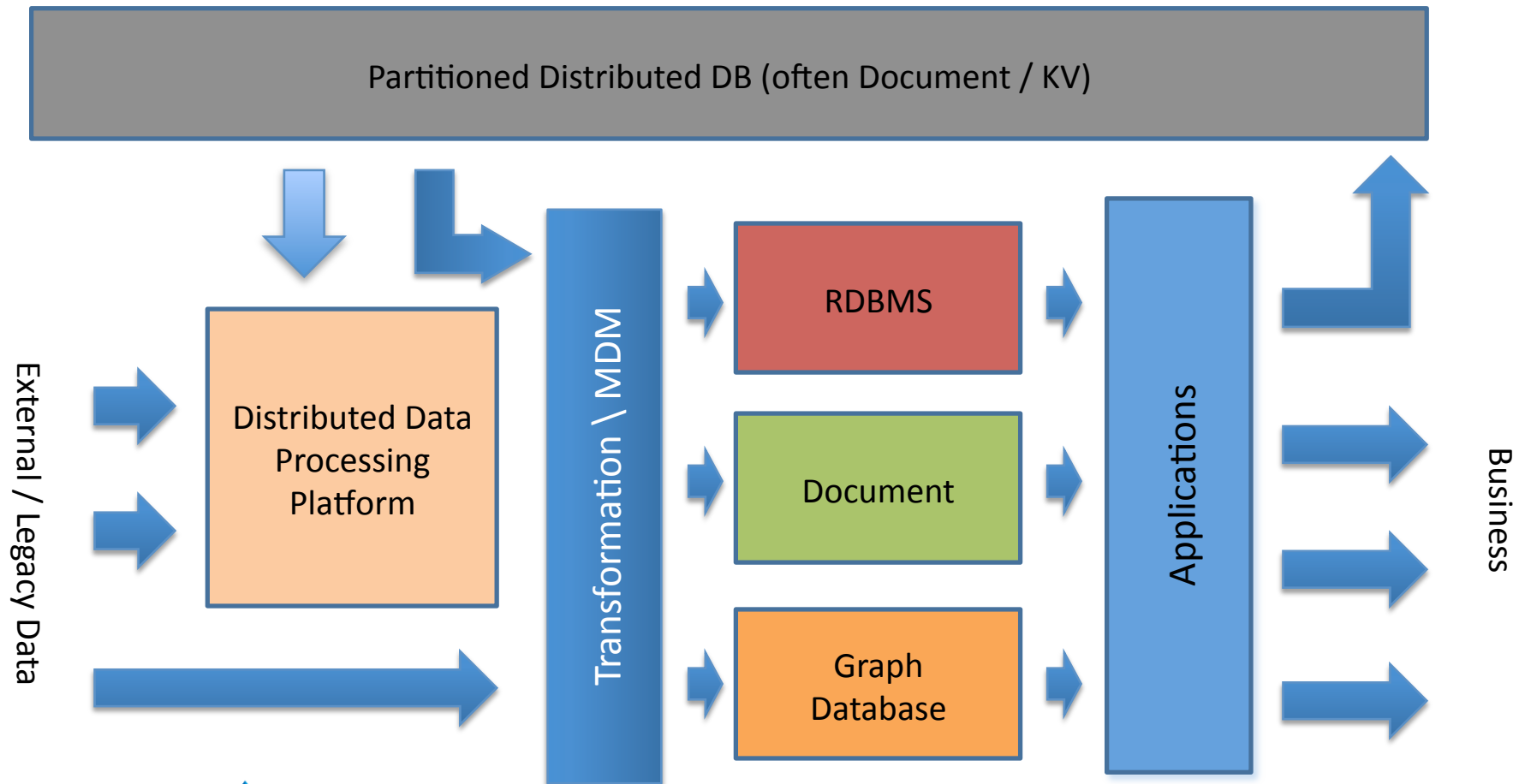


NoSQL – Data Specialists

- Everyone specializes
 - Doctors, Lawyers, Bankers, Developers ☺
- Why was data so normalized for so long !
- NoSQL is all about the data specialist
- Specializing in...
 - Distribution / deployment
 - Physical data storage
 - Logical data model
 - Query mechanism

Polyglot NoSQL Architectures

Users



The Physical Data Model

- Becoming a relationship specialist...

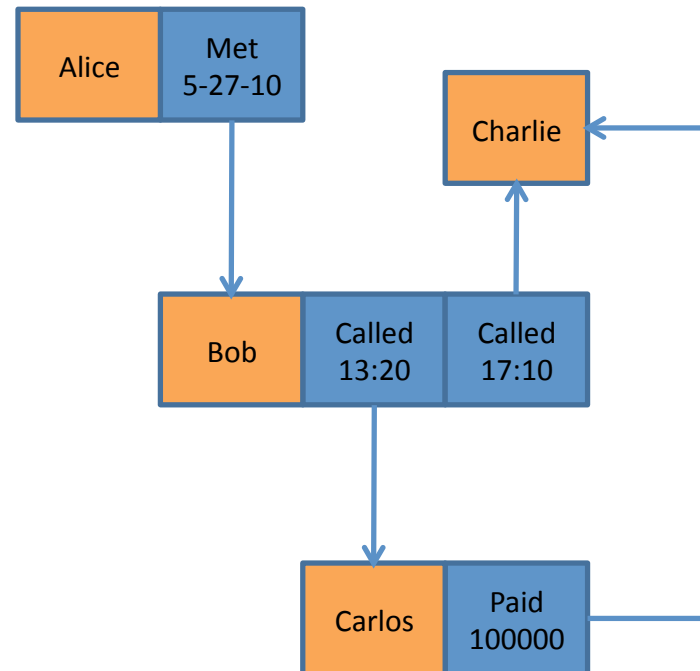
Rows/Columns/Tables

Meetings			
P1	P2	Place	Time
Alice	Bob	Denver	5-27-10

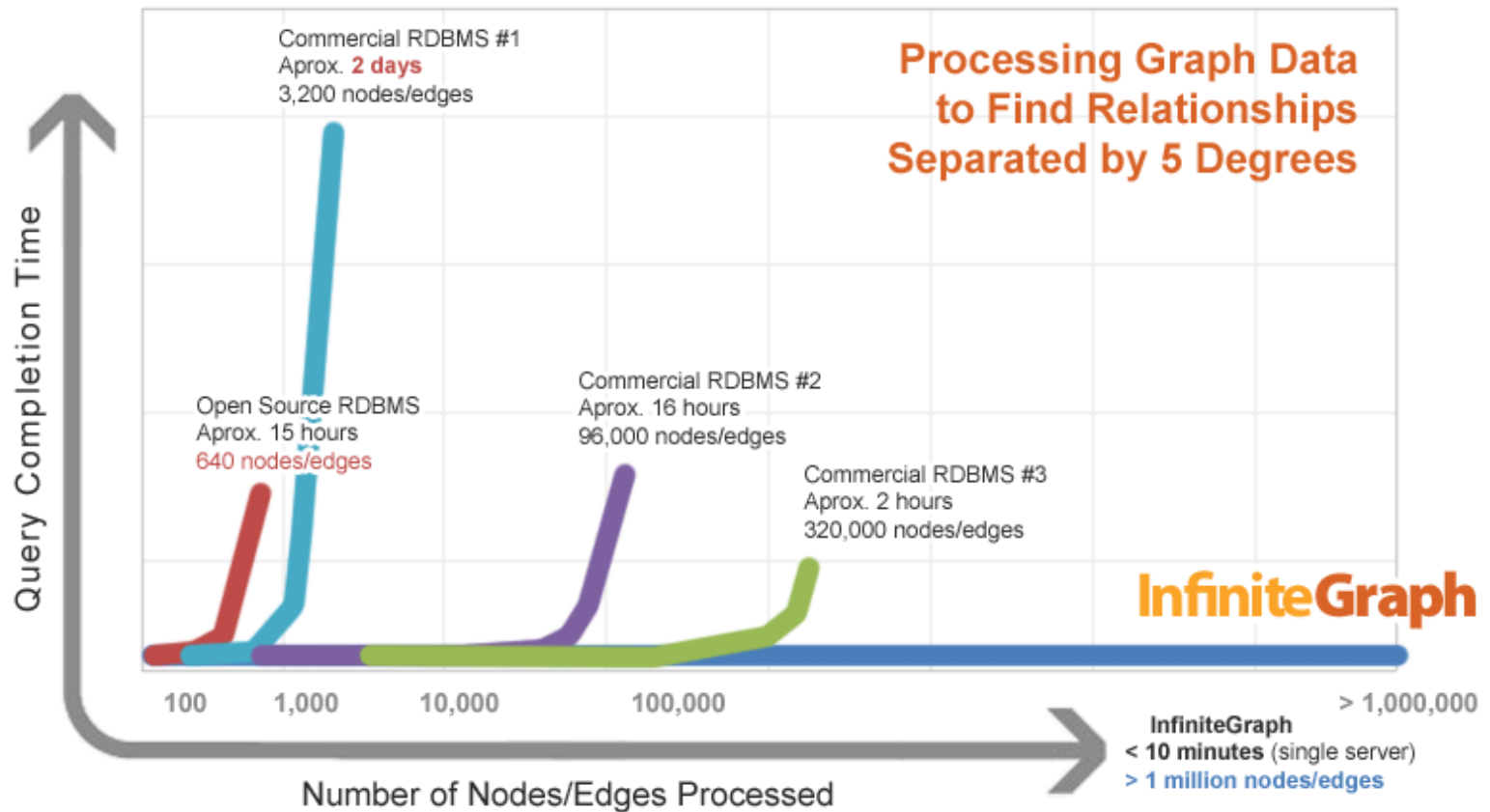
Calls			
From	To	Time	Duration
Bob	Carlos	13:20	25
Bob	Charlie	17:10	15

Payments			
From	To	Date	Amount
Carlos	Charlie	5-12-10	100000

Relationship/Graph Optimized



Navigational Query Performance

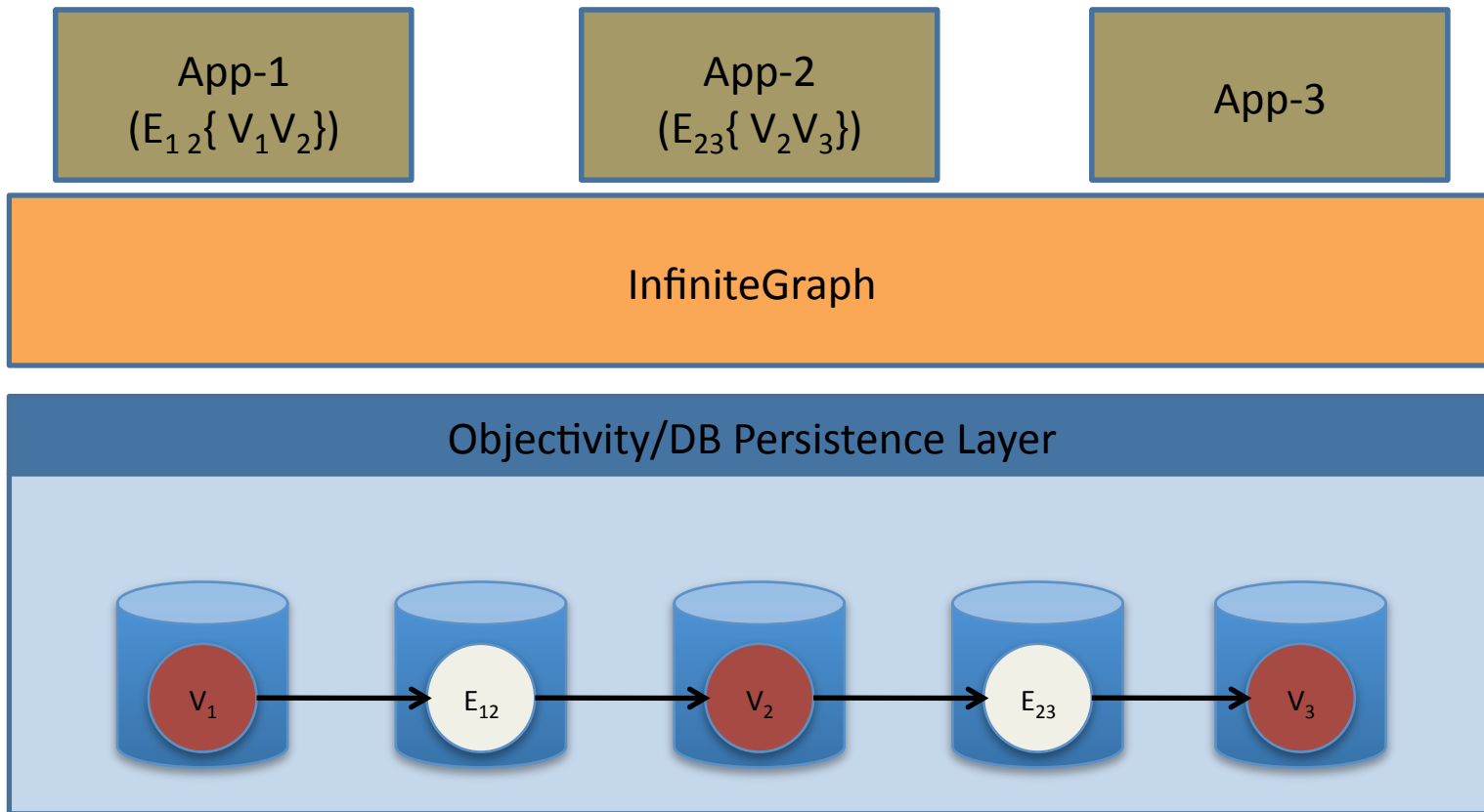


Scaling Writes

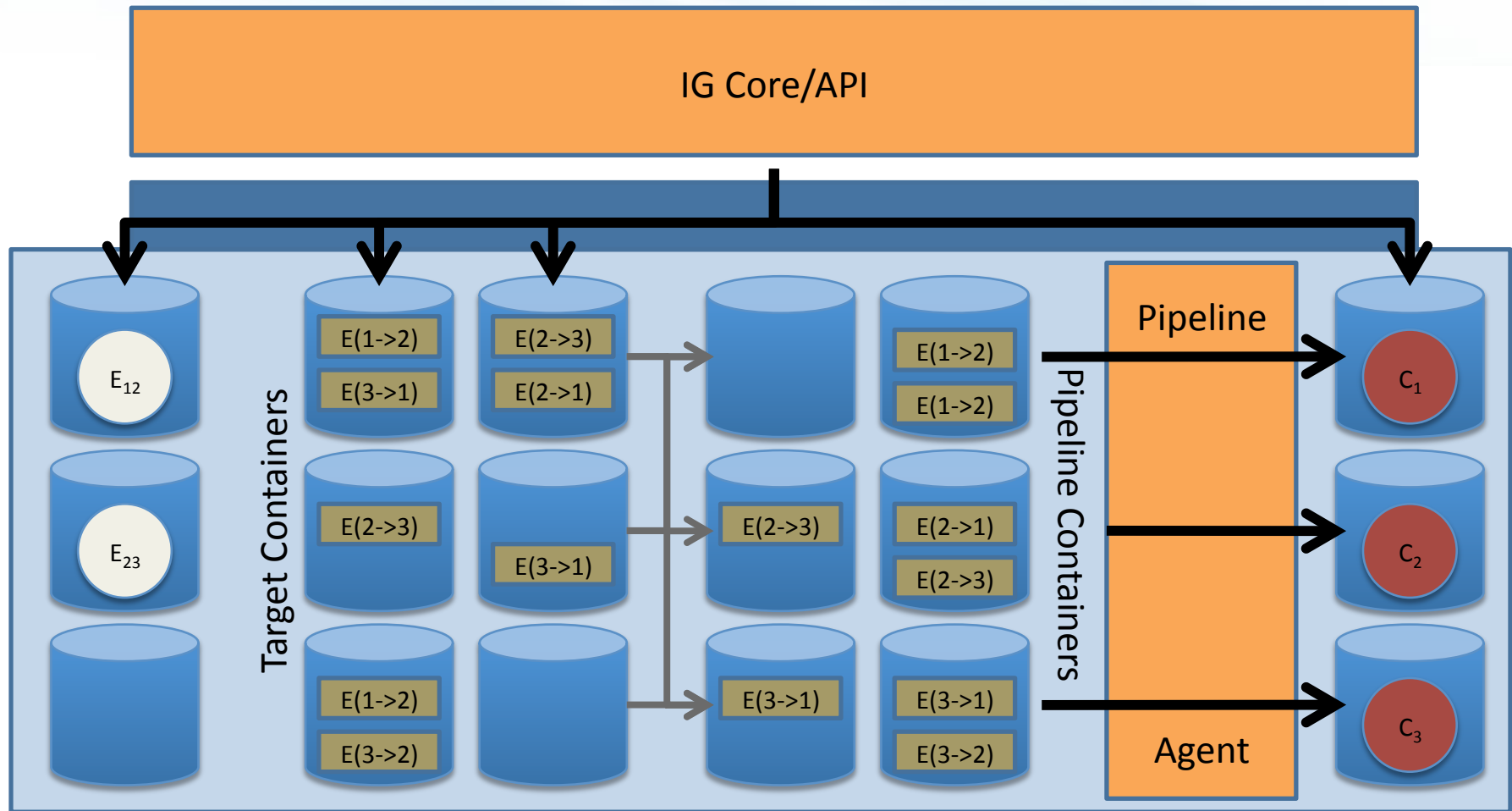
- Big/Fast data demands write performance
- Most NoSQL solutions allow you to scale writes by...
 - Partitioning the data
 - Understanding your consistency requirements
 - Allowing you to defer conflicts

Scaling Graph Writes

ACID Transactions



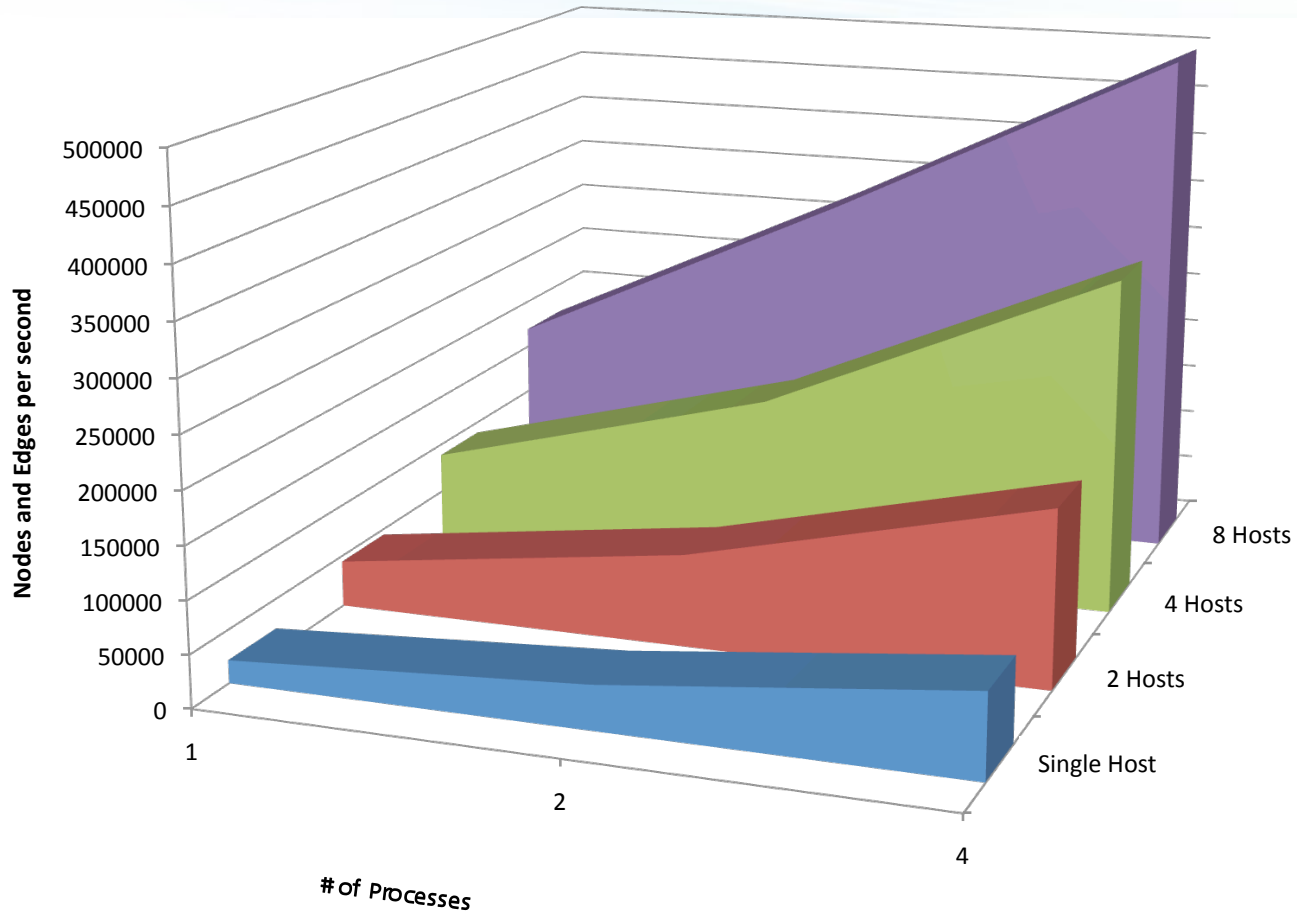
High Performance Edge Ingest



Trade offs

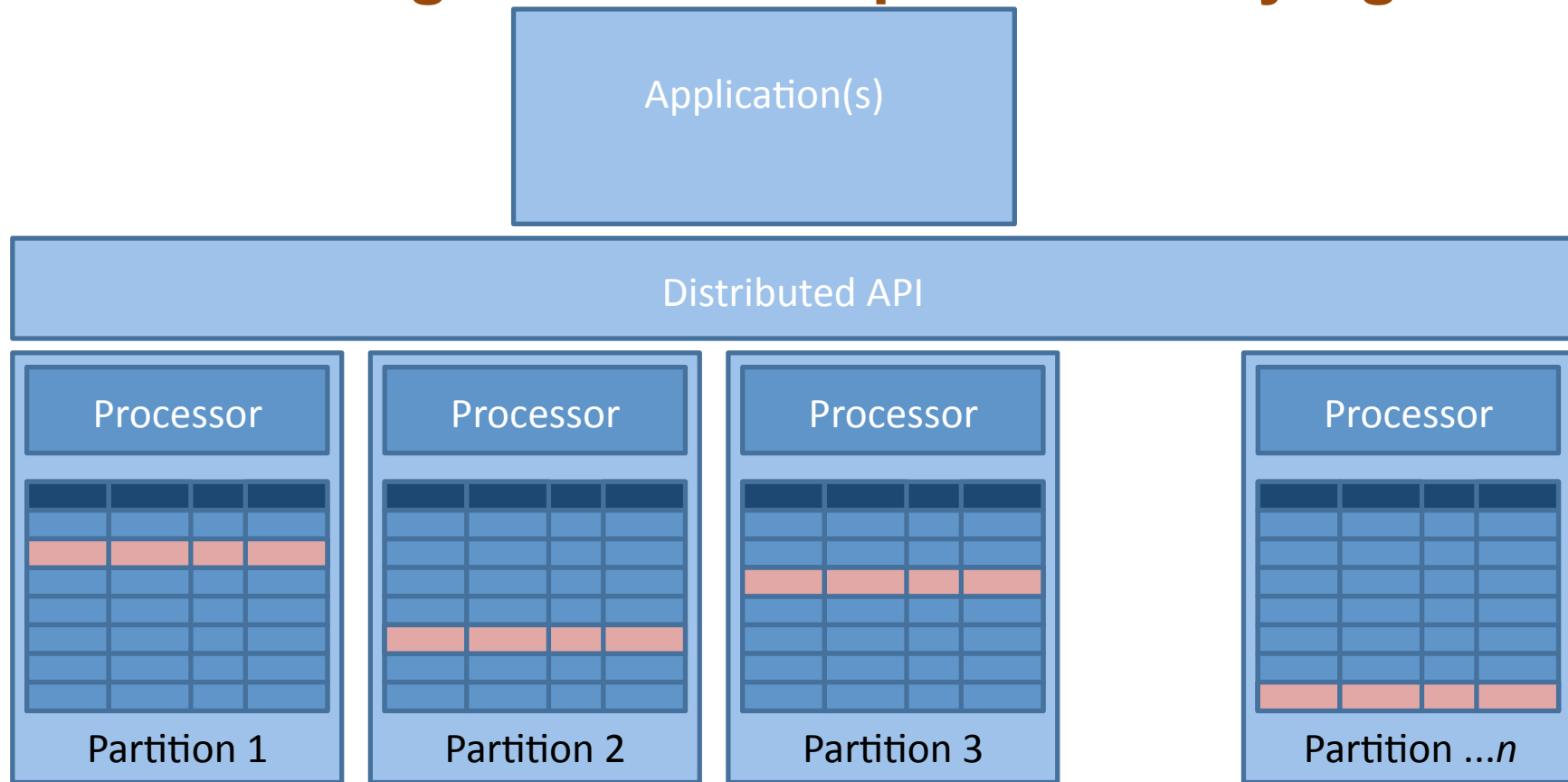
- Excellent for efficient use of page cache
- Able to maintain full base data consistency
- Achieves highest ingest rate in distributed environments
- Almost always has highest “perceived” rate
- Trading Off :
 - Eventual consistency in graph
 - Updates are still atomic, isolated and durable but phased
 - External agent performs graph building

Result...

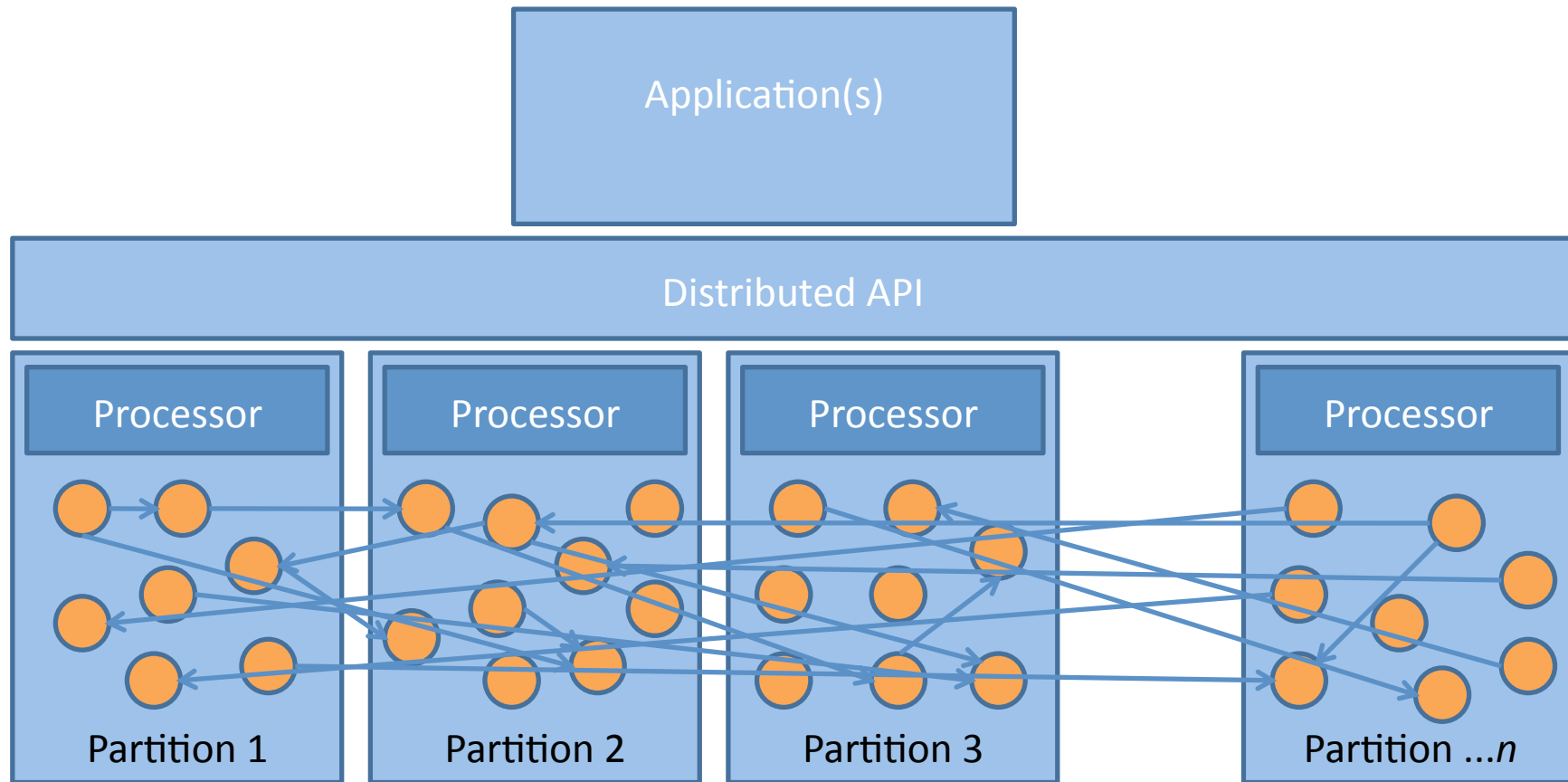


Scaling Reads and Query

Partitioning and Read Replicas... easy right !



Why are Graphs Different ?

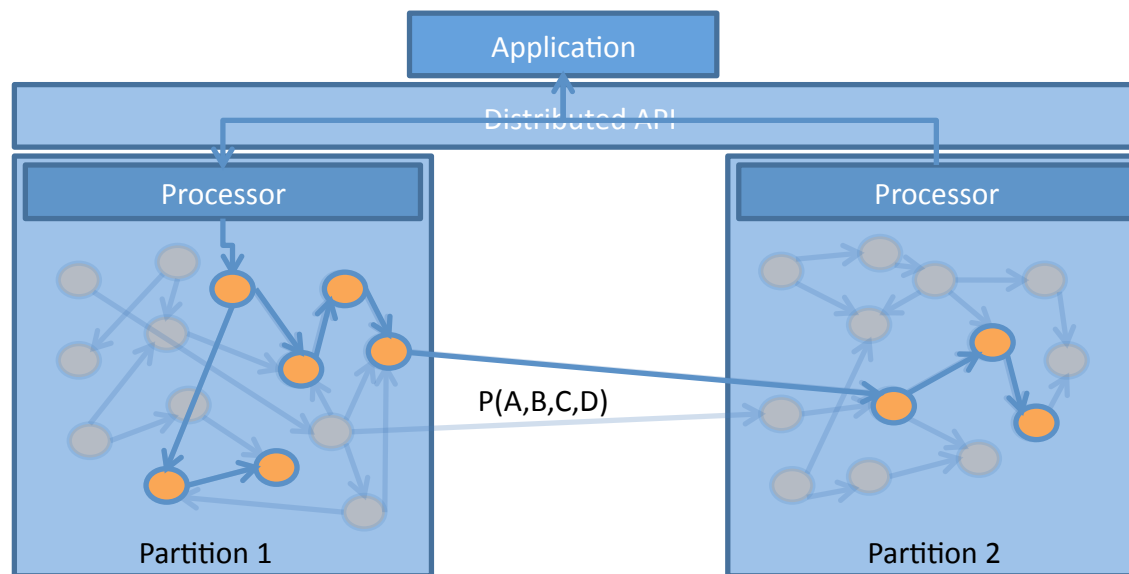


Optimizing Distributed Navigation

- Pregel Clones
 - Message passing for each hop, messages processed at the data host for the target vertex
 - High concurrency, no data leaves its physical host
 - Message marshalling and transport is expensive
- Distributed Caching Models
 - Essentially trying to cache graph in memory over multiple hosts
 - Requires too much memory for large graphs
 - Issues with cache consistency

Optimizing Distributed Navigation

- InfiniteGraph... both
 - Detect local hops and perform in memory traversal
 - Intelligently cache remote data when accessed frequently
 - Route tasks to other hosts when it is optimal



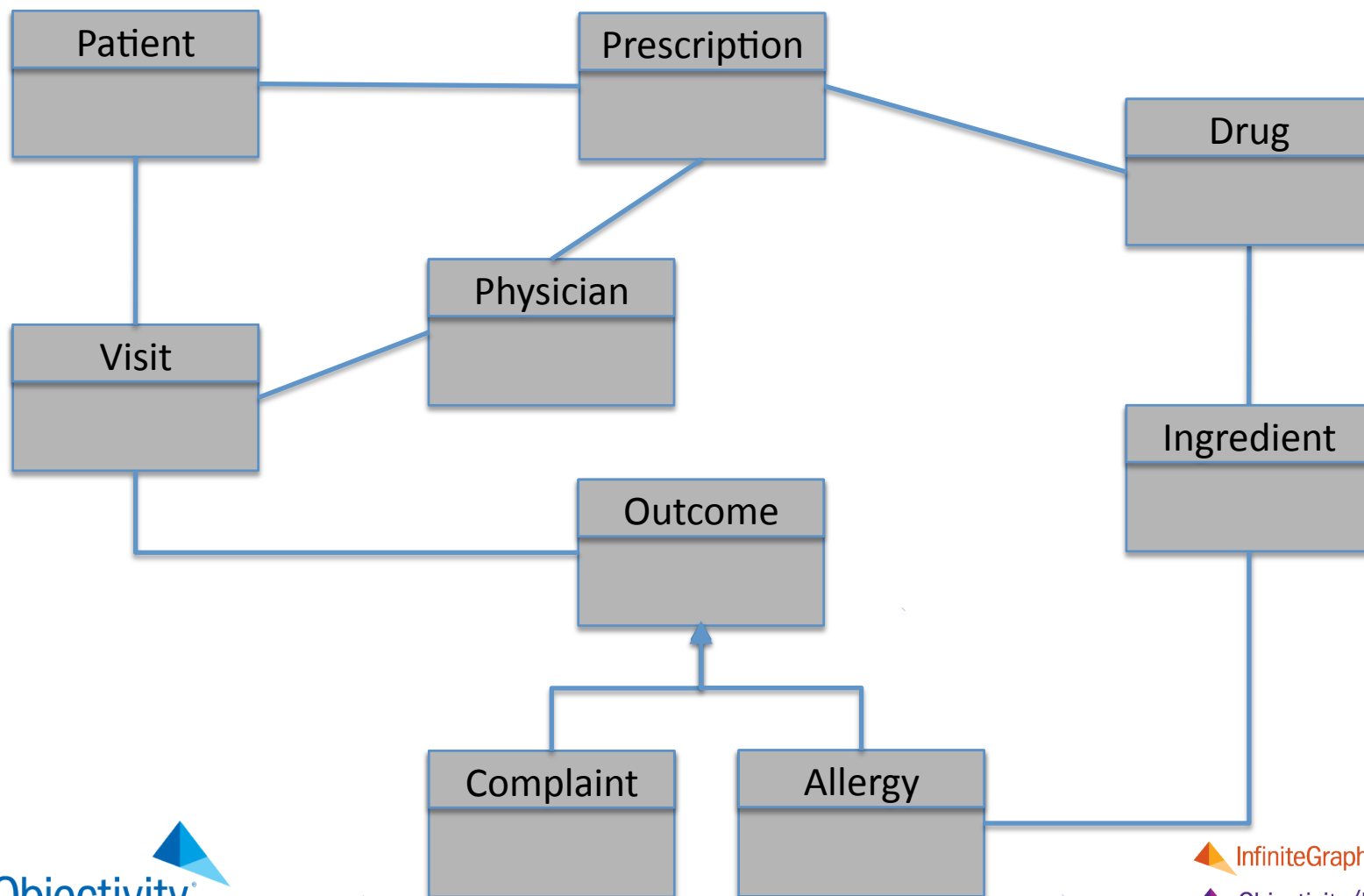
Schema – It's not your enemy !

(at least not all the time...)

- Schema vs Schema-less
 - Database religion
 - No time for a full debate here
 - InfiniteGraph supports schema, but does not restrict connection types between vertices
 - Planning to also support “Document Style” Nodes

GraphViews

Leveraging Schema in the Graph

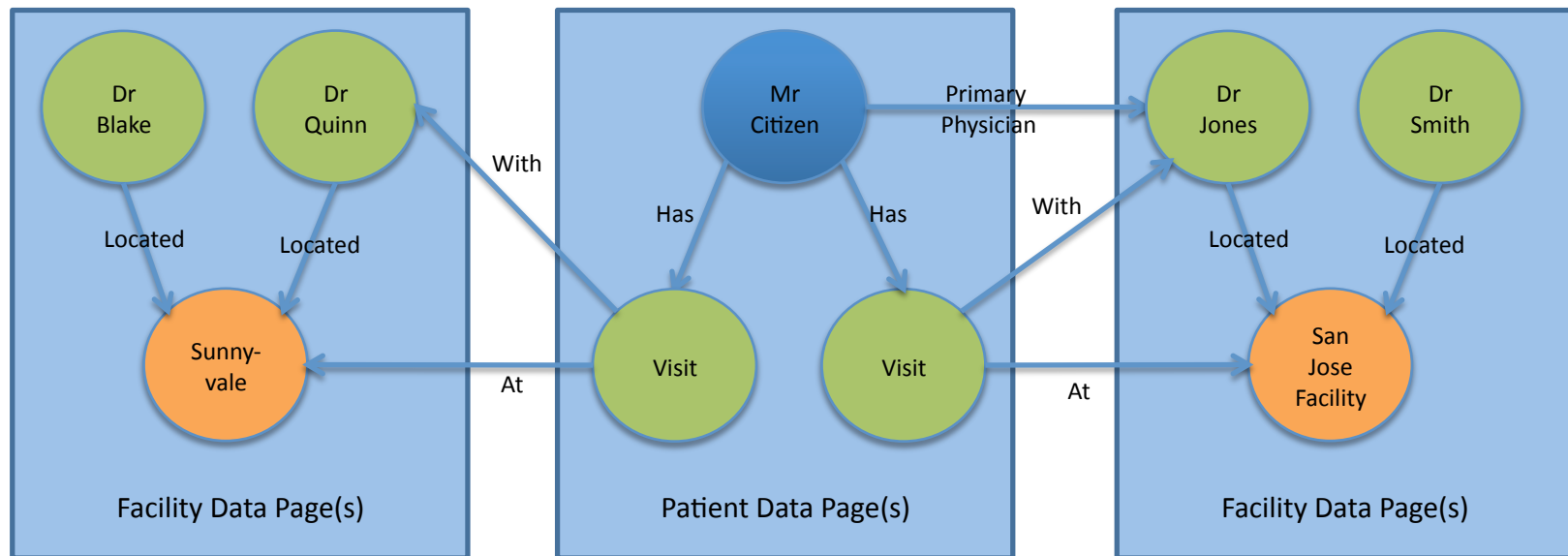


Schema Enables Views

- GraphViews are extremely powerful
- Allow Big Data to appear small !
- Connection inference can lead to exponential gains in query performance
- Views are reusable between queries
- Built into the native kernel

Advanced Configured Placement

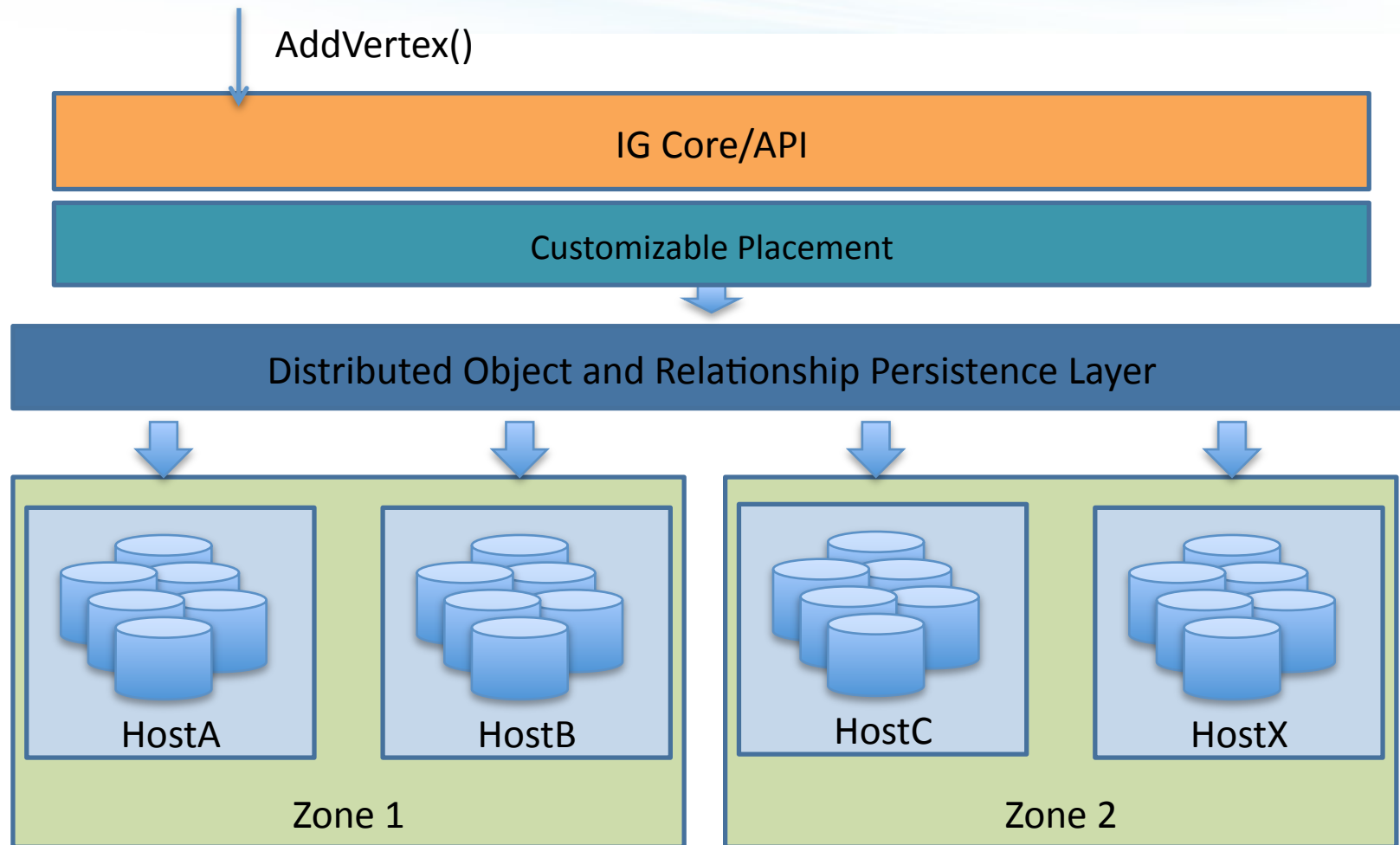
- Physically co-locate “closely related” data
- Driven through a declarative placement model
- Dramatically speeds “local” reads



Why InfiniteGraph™ ?

- Objectivity/DB is a proven foundation
 - Building distributed databases since 1993
 - A complete database management system
 - Concurrency, transactions, cache, schema, query, indexing
- It's a Graph Specialist !
 - Simple but powerful API tailored for navigation of data
 - Easy to configure distribution model

Fully Distributed Data Model



InfiniteGraph is a Complete Database

- InfiniteGraph helps manage the things you don't want to do, but want to have done:
 - **Concurrency**
 - Transactions (commit/rollback)
 - Controlled multi-user reading during updates
 - **Schema Control**
 - Build complex data structures, make changes easily and migrate existing data
 - **Distribution**
 - Sharing large amounts of distributed data between distributed processes
 - **Indexes**
 - Choose built-in key-value, b-tree or other indexes
 - **Cache**
 - Keep large sections of the graphs in configurable memory caches

Super Simple API

```
Person alice = new Person("Alice");  
helloGraphDB.addVertex( alice );
```

```
Person bob = new Person("Bob");  
helloGraphDB.addVertex( bob );
```

```
Person carlos = new Person("Carlos");  
helloGraphDB.addVertex( carlos );
```

```
Person charlie = new Person("Charlie");  
helloGraphDB.addVertex( charlie );
```

Adding Edges

```
MyEdgeType edge = new MyEdgeType();  
vertexA.addEdge ( edge, vertexB, EdgeKind.??? );
```

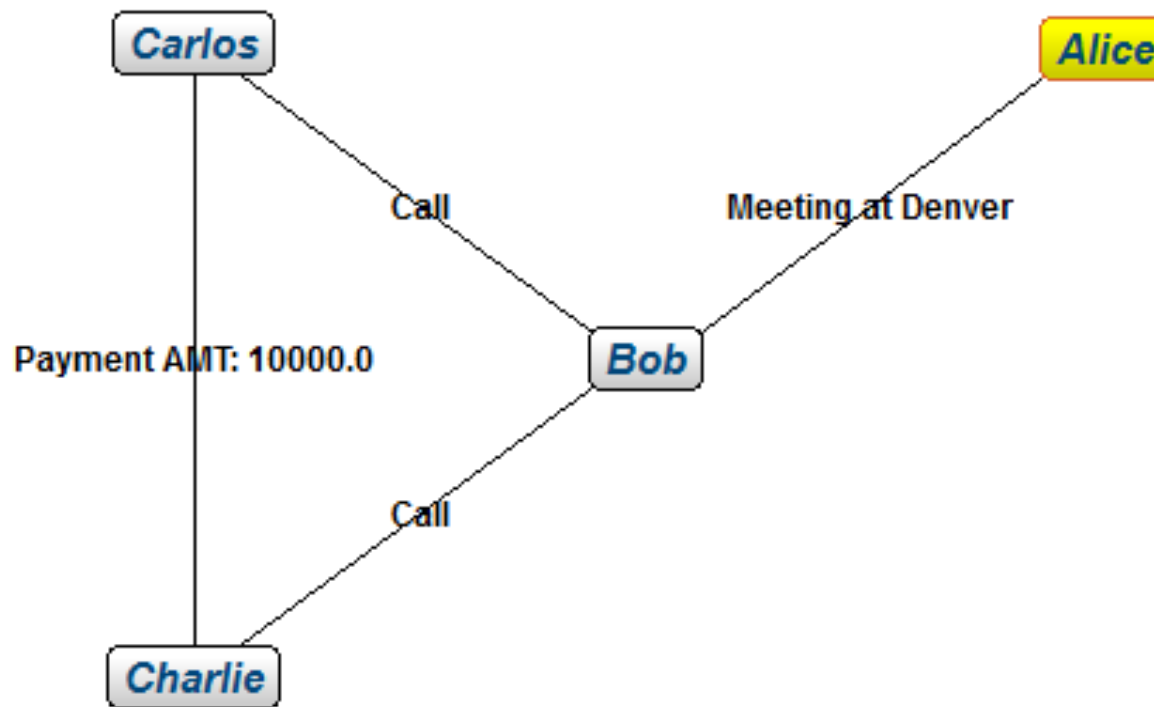
```
Meeting denverMeeting = new Meeting("Denver", "5-27-10");  
alice.addEdge(denverMeeting, bob, EdgeKind.BIDIRECTIONAL);
```

```
Call bobCallToCarlos = new Call(getRandomJulyTime());  
bob.addEdge(bobCallToCarlos, carlos, EdgeKind.BIDIRECTIONAL);
```

```
Payment payment = new Payment(10000.00);  
carlos.addEdge(payment, charlie, EdgeKind.BIDIRECTIONAL);
```

```
Call bobCallToCharlie = new Call(getRandomJulyTime());  
bob.addEdge(bobCallToCharlie, charlie, EdgeKind.BIDIRECTIONAL);
```

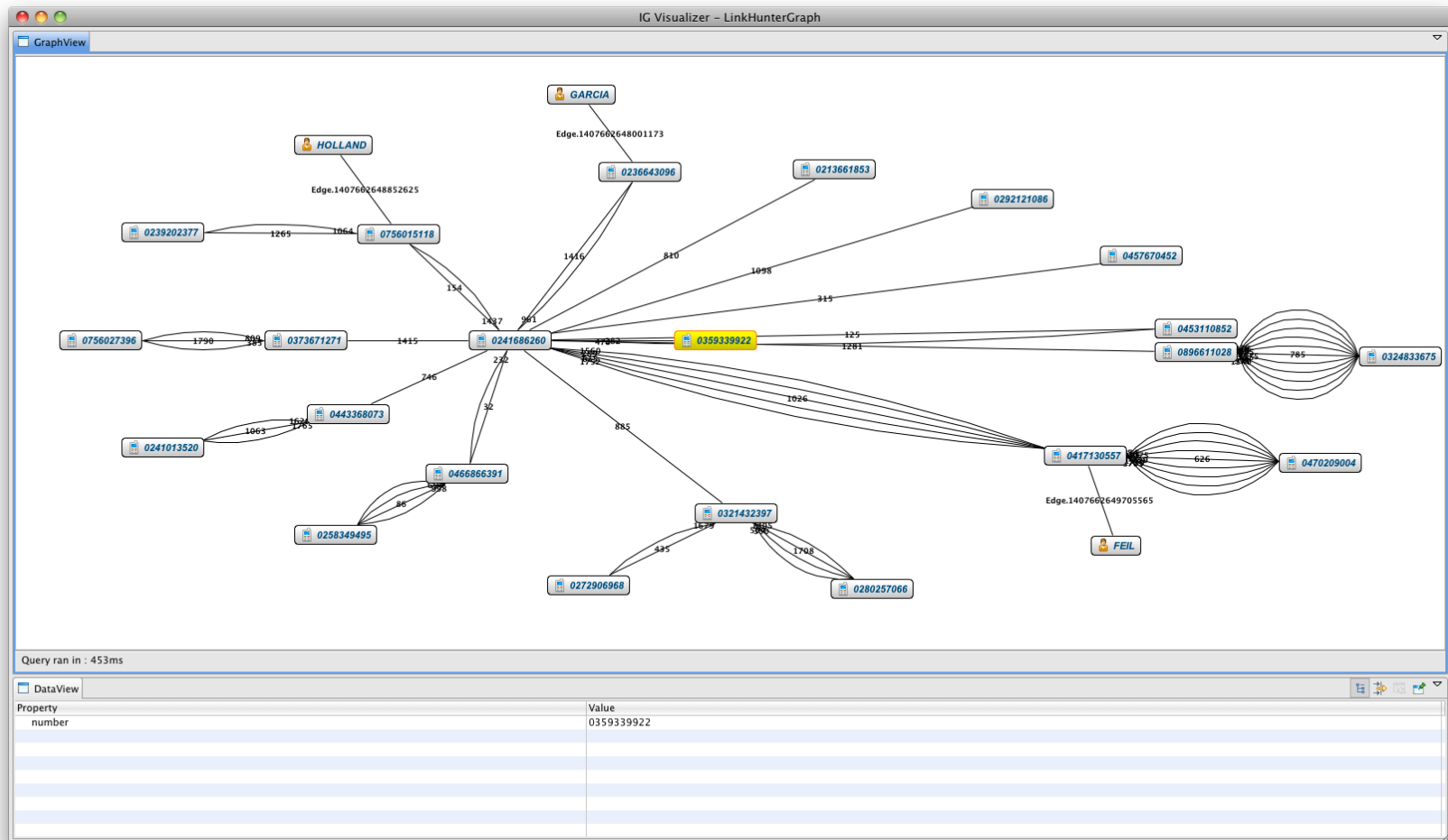
The Result...

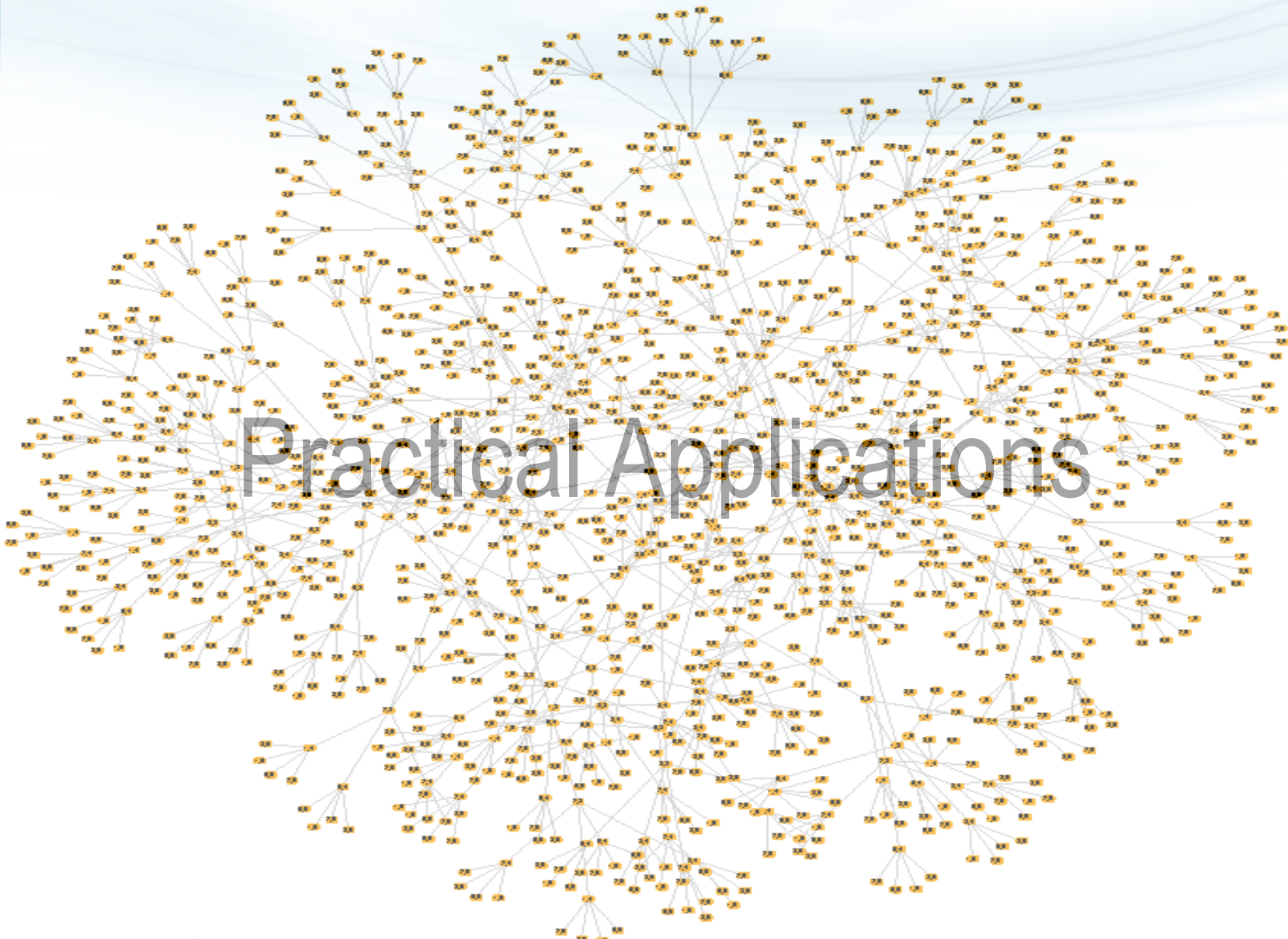


Graph Traversal (Navigation) Queries

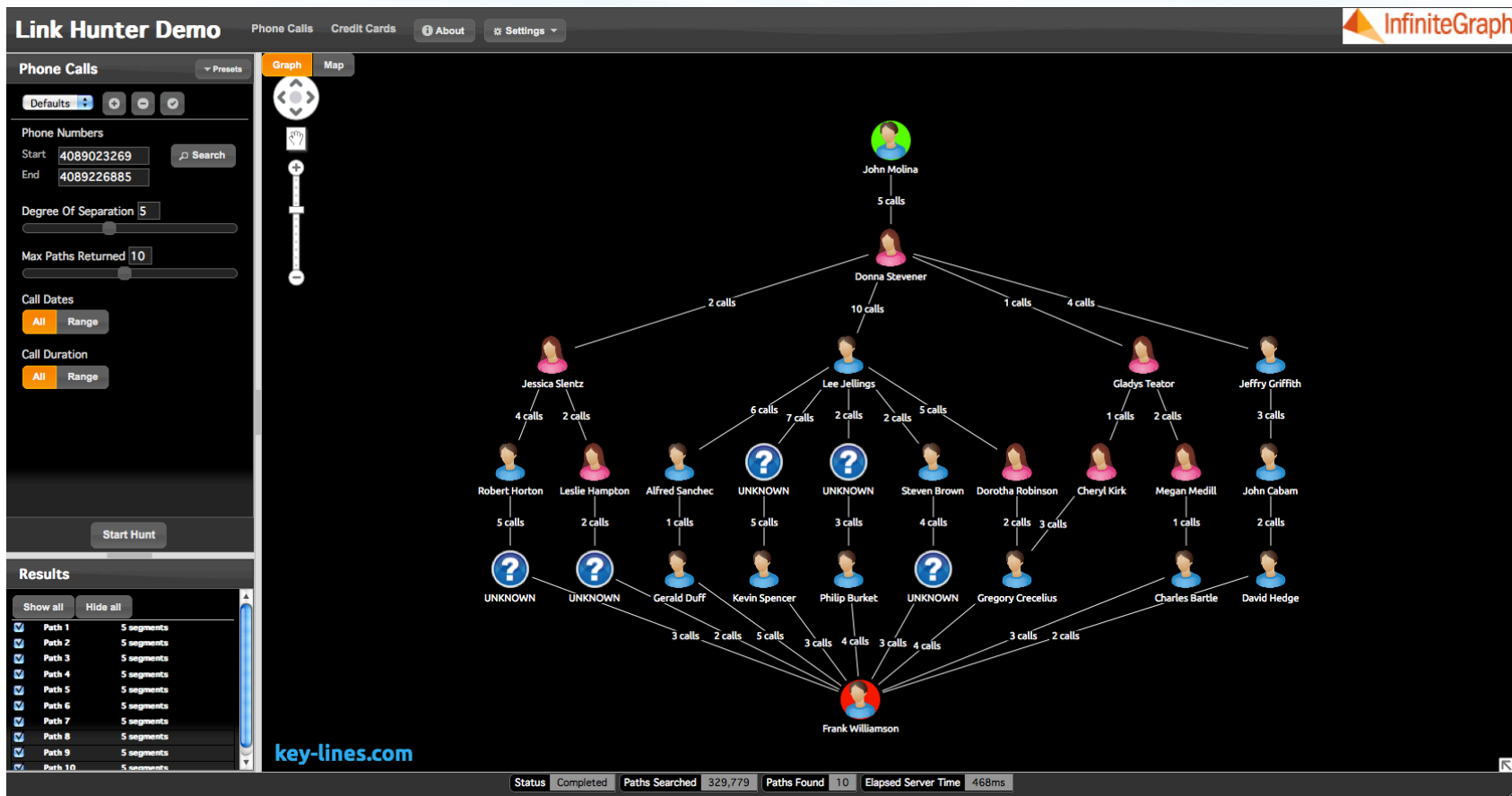
- Use an instance of the Navigator class to perform a navigation query.
- A navigation instance is highly customizable, but is comprised of the following basic parts:
 - Origin : The vertex from which to begin
 - Guide strategy
 - **Guide.Strategy.SIMPLE_BREADTH_FIRST**
 - **Guide.Strategy.SIMPLE_DEPTH_FIRST**
 - Graph Views : Powerful filtering and
 - Qualifiers
 - Qualifying valid intermediate paths and results
 - Handlers
 - A result handler

Tools To Suit the Solution

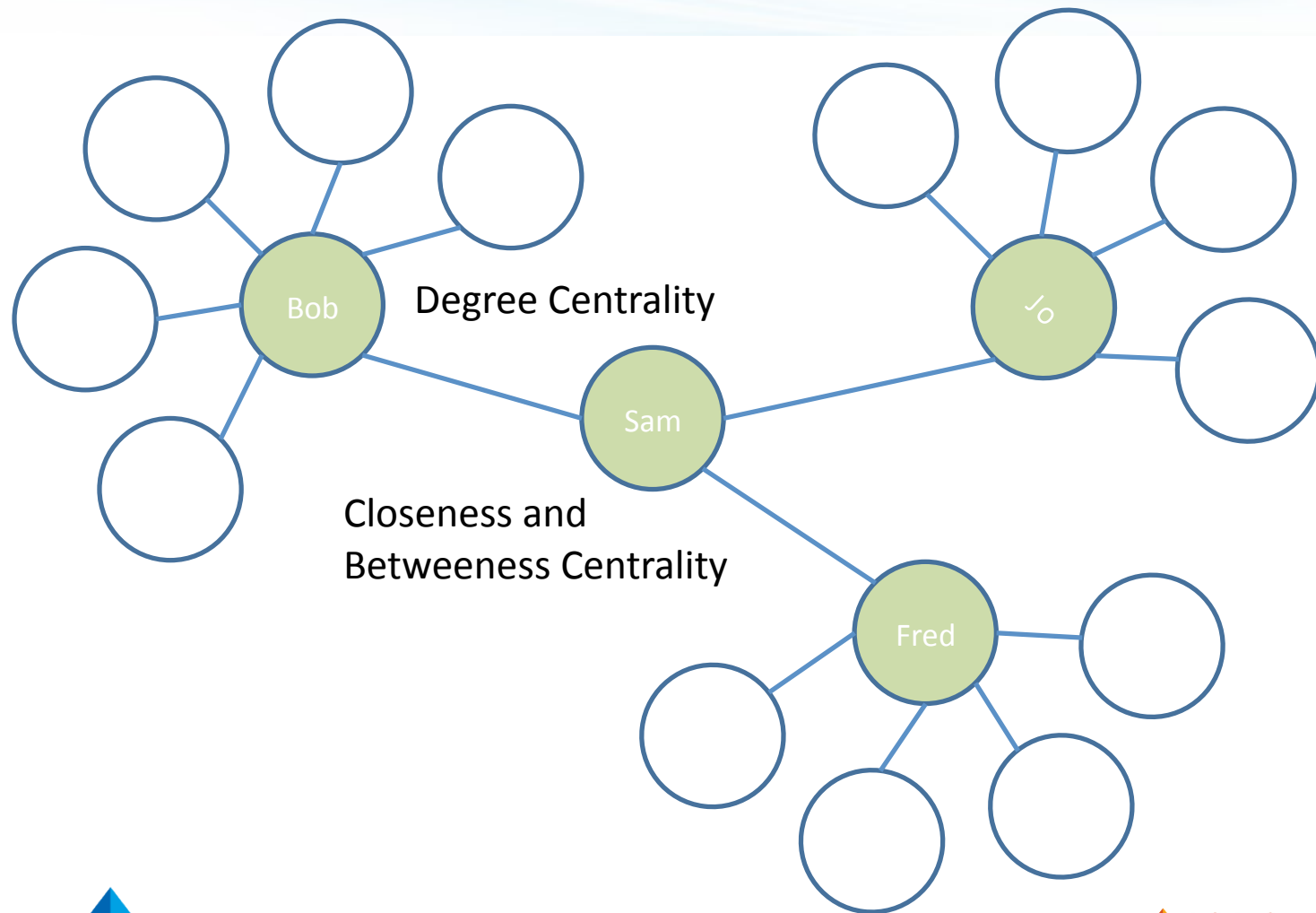




Pathfinding



Graph Analysis (Algorithms)



Graph Analysis (Algorithms)

- Social Networks
 - Most connected participants
 - Influencers
 - Important Syndicates or Sub-networks
- Central figures in crime organisations
- Business Intelligence
 - Discovering Knowledge Assets
 - Complex Big Data Analytics

Graph Analysis (Patterns)

- Crime (again)
 - Recognize common patterns of activity
 - Complex chains of interaction
- Security
 - Recognize attack/threat patterns
 - Auditing / log analytics
- Targeting Advertising
 - To specific browsing patterns

Many Many More

- Spatial data
- Financial Services
- Defense / Situational Awareness
- Sciences
- Health Care
- Genealogy
- Logistics
- Tracking
- PLM