Build your first Internet of Things app today with open source
Trilateration and its limitations
Particle Filters - Calculating the optimum solution
The solution

1. Capture signal strength
2. Calculate distance from antenna
3. Trilaterate different sensors to predict location in real-time
4. Show on a map with live updates
Architecture Overview

Ingest

SpringXD

Groovy

Transform

Sink

Calculate Device Distance

+ Distance

Predict Location

Spring Boot

GUI

Application Platform

JSON

HTTP
Introduction to Geode

[Image of a box of Break Your Own Geodes and a collection of geodes]
A distributed, memory-based data management platform for data oriented apps that need:

• high performance, scalability, resiliency and continuous availability
• fast access to critical data set
• location aware distributed data processing
• event driven data architecture
• Cache
  • In-memory storage and management for your data
  • Configurable through XML, Spring, Java API or CLI
  • Collection of Region
Concepts

- Region
  - Distributed java.util.Map on steroids (Key/Value)
  - Consistent API regardless of where or how data is stored
  - Observable (reactive)
  - Highly available, redundant on cache Member(s).
Concepts

- Region
  - Local, Replicated or Partitioned
  - In-memory or persistent
- Redundant
- LRU
- Overflow
Concepts

- Persistent Regions
- Durability
- WAL for efficient writing
- Consistent recovery
- Compaction
Concepts

- Member
  - A process that has a connection to the system
  - A process that has created a cache
  - Embeddable within your application
• Client cache
  • A process connected to the Geode server(s)
  • Can have a local copy of the data
  • Can be notified about events on the servers
Concentric

- Functions
  - Used for distributed concurrent processing (Map/Reduce, \textit{stored procedure})
  - Highly available
  - Data oriented
  - Member oriented
Concepts

• Functions

```
filter = Keys X, Y

1. FunctionService.onRegion.withFilter.execute

2. execute

3. result

4. execute

5. result

6. ResultCollector.getResult
```
Concepts

- Listeners
  - CacheWriter / CacheListener
  - AsyncEventListener *(queue / batch)*
    - Parallel or Serial
    - Conflation
Introduction to SpringXD

Runs as a distributed application or as a single node

HTTP POST of Data Processing DSL

XD Admin

XD Container
Module

XD Container
Module

HTTP POST of Data Processing DSL

XD Admin

 XD Container
Module
A stream is composed from *modules*. Each module is deployed to a *container* and its channels are bound to the *transport*.
Apache Geode Roadmap

- HDFS Persistence
- Off-heap memory storage
- Lucene Search
- Spark Integration
- Cloud Foundry service
Spark Connector

"An RDD in Spark is simply an immutable distributed collection of objects. Each RDD is split into multiple partitions, which may be computed on different nodes of the cluster. RDDs can contain any type of Python, Java, or Scala objects, including user-defined classes."

- Geode Spark Connector allows:
  - Regions exposed as RDD
  - RDDs persisted in Geode
  - Execute queries on Geode using OQL
Spark Connector

- Web based REPL
- Multiple Interpreters
  - Apache Spark
  - Markdown
  - Flink
  - Python...
- Iterative & Exploratory

- Data Processing
- Columnar queries
- RDDs
- Machine Learning
- Analytics
- Streaming

- Data store
- In-memory & Persistent
- Highly Consistent
- Transaction processing
- Thousands of concurrent clients
Source code and detailed instructions available at:

https://github.com/Pivotal-Open-Source-Hub/WifiAnalyticsIoT