

Data Persistence Project Group (Inaugural) Meeting – 8/2/18

Attendees: Itamar Haber, Andre Srinivasan (Redis Labs); Akram, Jim, Brandon, Trevor (Dell), Rodney (Beechwoods), David (Forge Rock), Priya Rajagopol (Couchbase), Tony & Ian (Canonical), Markus Junginger (ObjectBox), Andy, Steve, & Ed (IoTech), Vinay Gajjala (Axzon – formerly RFMicron), Greg Zuro, Tom Poole, Michael Hall (LF). Attendees that may have joined after the start of the meeting may not have been captured and listed.

Discussion and action items as a result of meeting in **RED**

Initial questions:

- How big is Mongo?
- Does it have to be Apache 2? (versus AGPL of Mongo)
- Do we have a tiered approach to data persistence? Do we talk about these concepts – long term versus short term?
- Are we going to address data backup?
- Above two issues should be addressed by a data maintenance project group upcoming
 - Should we merge that effort to this one? TBD, but perhaps.
- Need interfaces to be easier to work in order to be able to use different databases!

New Business

- Our charter (any objections or changes?):
 - Document EdgeX persistence requirements
 - Outline potential persistence stores and architectures to meet the requirements
 - Research the options
 - Make a recommendation to the Core WG on persistence architecture and reference implementation for future releases of EdgeX (for Delhi release)
 - Complete agnostic approach in the services to what the data base is – want to push anything specific to a “database layer”. Example- how we handle the database id, or dealing with database type
 - Define a strategy around multiple database support
No objections to the charter and adopted for this project group.
- Timeline: to make a recommendation to Core WG and then TSC on EdgeX reference implementation of data persistence by Delhi release
 - Outline required changes and level of effort in code base to support for Edinburgh planning session (Oct 22).
- What are our data persistence requirements? **Jim to refine and to try to find some use cases, users, and competitive product requirements for comparison**
- **Everyone asked to review and provide feedback by email list, RocketChat channel or email to Jim directly (james_white2@dell.com)**
 - Must haves
 - License of product – compliant with Apache 2
 - **Store and forward needs**
 - Platform support:
 - Intel , ARM 64 bit; 32 bit is moved to nice to have

- OS??
- Small footprint
 - Need a holistic view. Small but slow is not acceptable.
 - Memory size, footprint, CPU, network? What is size? All these must be taken into consideration together.
- Performant
 - How quantified? # of transactions/sec?
 - Speed of queries?
 - Support real time?
 - Do we want to focus (prioritize) read/write or read or write
 - Probably writes over reads
 - We need real world use case data.
 - Need millisecond or sub millisecond (near real time)
 - RFID Use case: 100+ / sec sensor reads
 - Building automation use cases: 1000+/sec
 - **Jim to check with NOV, Samsung**
 - **Chandra will provide some data to Jim**
 - Concern impact of backup processes
- Latency
- Durable across EdgeX shutdown (no in-memory db)
 - Nice to have a database that also runs in memory
 - In-memory databases does not mean not-durable
- Run in a container (Docker/Kubernetes/Snap/etc.)
 - Has to manageable from one control plane
 - Should we consider embedded database in the same process?
 - How isolated/decoupled should the application be from the database? Think JDBC type layer or ORM to SQL databases, but slow.
 - Much of this depends on use case
 - Performance considerations
 - Should EdgeX be more performant out the box or more flexible? Is it a platform/framework or is it a reference implementation?
 - Many stressed importance of flexibility given EdgeX nature.
- Secure
 - Password protected
 - Supports data encryption (protect data at rest)
 - ???
- Has Java, Go, C, C++ drivers/connectors
 - What others are probably needed
- Community support and user-base size
 - How to quantify?
- Binary support (as long as we can identify what type of binary)
 - Max size (up 16MB)
- Nice to have
 - NoSQL (versus SQL)?
 - Synch capability?
 - Backup support?

- Transactional (ACID) or Eventual Consistent (which CAP axis)?
 - Support multi-tenancy?
 - 32 bit support (Intel or ARM)
 - ??
- Should we start collecting potential options and can someone lead that effort
 - Not a review or evaluation at this stage, just a list of potential databases
 - MongoDB
 - Mongo Mobile
 - Redis
 - Influx
 - Cassandra
 - Couchbase
 - SQLite
 - Etc.
- Next meeting time – Jim is out next week.
 - Does 9am CDT on Thursdays (before Core WG work for everyone)? Proposed alternate days/time
 - 30 minutes or 60 minutes?
 - Who else do we really need on the call?
 - **Jim to send out Doodle poll**