

- [Administration](#)
- [BEN COTTON](#)
- [Dashboards](#)[Access more options \(Alt+d\)](#)
- [Projects](#)[Access more options \(Alt+p\)](#)
- [Issues](#)[Access more options \(Alt+i\)](#)
- [Agile](#)[Access more options](#)
- [CSV Importer](#)[Access more options](#)

- Quick Search
- [Create Issue](#)

• 1 of 2 [Return to search](#)



- [LRI Core Services](#)
- [LRISS-2554](#)

Design Plan: OpenHFT as Off-Heap (/dev/shm) Provider of Agg Engine QuantitativeReferenceData Operands (POC)

- Edit
- Assign
- Comment
- More Actions
- Assign for work
- Need more information
- Workflow
- Share
- Views
- Details

- Type: Task
- Status: In Analysis and Review
- Priority: High
- Resolution: Unresolved
- Affects Version/s: None
- Fix Version/s: [Future Release - TBD](#)
- Component/s: [LINUX-Agg Engine](#)
- Labels: None

- Work Calendar:
Standard - 40 hr/week

- Proj. Task Type:
Effort driven

- Release Date:
DECEMBER 31 2014

Description

This JIRA will document the ambition to deliver a POC deployment that soundly/completely exercises the potential veracity of OpenHFT (<http://www.openHFT.net>) being an Off-Heap (/dev/shm/) provider of Agg Engine ReferenceData operands.

J.P.Morgan

This JIRA continues the work started by

JIRA <https://issuetracking.jpmmchase.net/jira8/browse/LRISS-1142>

The central purpose of this JIRA is to document OpenHFT's demonstrated capability to affect

1. a ZERO-COPY transport whenever any Agg Eng operators (AE user requests/tasks) interact with any centrally shared Agg Eng operands (/dev/shm/ pinned RDR_DIMENSIONS)
2. a sub microsecond (< 1 us) average latency that soundly/completely satisfies #1 via native Linux IPC (but coded by AggEng application programmers via the OpenHFT ChronicleMap Java API). I.e. we get the **full** native Linux IPC capability via the OpenHFT provided Java API.
3. a solution that resolves the need to redundantly copy RDR_DIMENSION specific data amongst multiple Java VM processes' address spaces (i.e demonstrate the efficacy of a 1 x instance @ /dev/shm soundly/completely resolves the need of any redundancy of any kind)
4. a solution that brings to the native Linux process view an accommodating symmetry for MP (multi-process) synchronization/volatility/CAS/lock-free operator/operand interaction that exactly matches the symmetry for MT (multi-thread) capabilities (i.e. synchronization/volatility/CAS/lock-free operator/operand interaction) provided by the default (familiar) Java VM platform.

5. Document this POC effort using the classic 4+1 Architectural View Model of Software Development Methodology

(http://en.wikipedia.org/wiki/4%2B1_architectural_view_model and <https://www.cs.ubc.ca/~gregor/teaching/papers/4+1view-architecture.pdf>) – yielding to LRI CS AggEng Team and other LRI teams the following views of this

POC: (1) Logical View (2) Physical View (3) Process View (4)Development View and (4+1) Use Case/Scenarios

Activity

- All
- Comments
- Work Log
- History
- Activity
- Transitions Summary
- Subversion
- Source

There are no comments yet on this issue.

- [Comment](#)

People

- Assignee:

BEN COTTON

Reporter:

BEN COTTON

Vote (0)

Watching (1)

Dates

J.P.Morgan

- Due: 31/Dec/14
Created: Today 11:54 AM
Updated: Today 11:54 AM
Planned Date : 31/Dec/14
- Time Tracking
Estimated: 2m
Remaining: 2m
Logged: Not Specified
Agile
• View on Board

- [Bug tracking and project tracking for software development powered by Atlassian JIRA \(v5.1.8#787-sha1:823790c\)](#)
- [Report a problem](#)

FYI = <https://issuetracking.ipmchase.net/jira8/browse/LRISS-2554>

I will soon be seeking detailed counsel from Dmitry, Lev, Biao et. al. re: enumerating the full space of AggEng RDR_DIM RefData potential data structure anatomy(s) ... and accommodating that space with a ZERO-COPY capable OpenHFT representation.

The JIRA URL above will document this effort.

Thanks,

Ben

Ben D. Cotton III
J.P.Morgan
Liquidity Risk Technology
277 Park Ave Floor 8
New York, NY 10172-0003
212.622.5010
Ben.Cotton@jpmorgan.com