KRAKEN MARE#

HPE'S EXASCALE MONITORING FRAMEWORK PROTOTYPE+

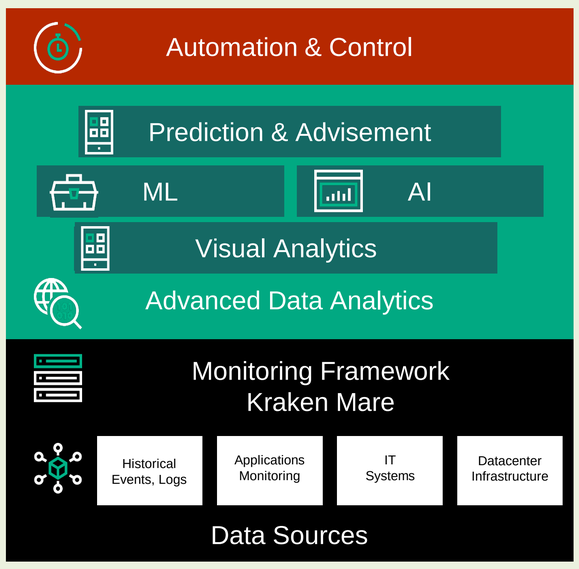
Sébastien Cabaniols, Aurélien Cogoluegne, Bruno Cornec, Nic Dube, Dennis Floyd, Jeff Hanson, Clément Poulain, and Torsten Wilde*

*wilde@hpe.com

Motivation

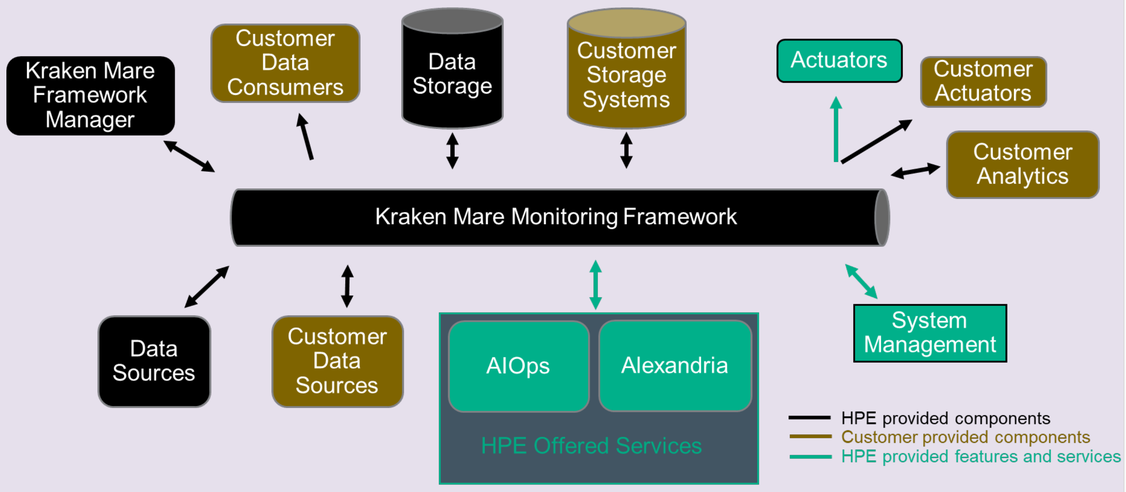
- Data Centers need to improve efficiency in light of ever growing power consumption and costs
- System and facility optimization at Exascale require machine learning approaches not used today
- These trends drive an exponential growth in system monitoring data both in terms of volume and frequency

- ***Kraken Mare (KM):*** Monitoring platform that collects, processes, and aggregates vast volumes of IT and facility telemetry from disparate sources and applies various algorithms to the data in real time



Solution

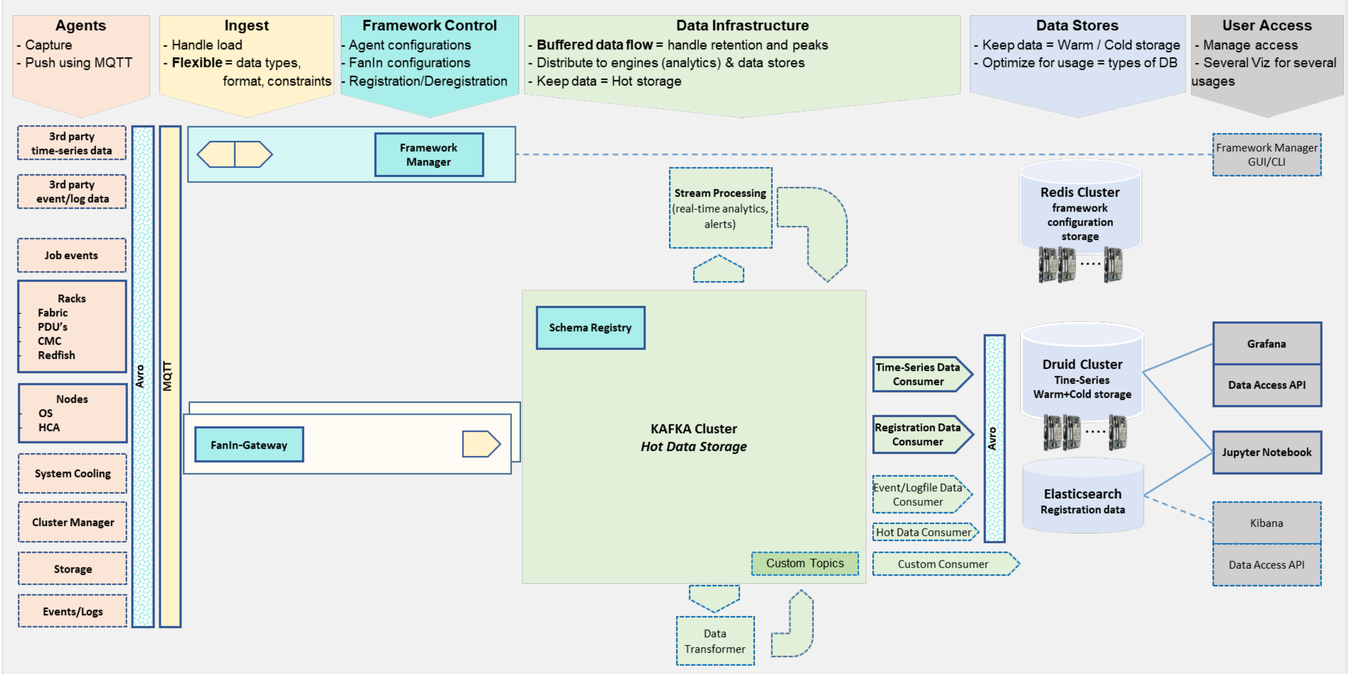
- Modular highly scalable architecture
  - Design point: 10 Million messages per second
  - Out-of-Band collection (over the management network)
- Dynamic (live reconfiguration / polling intervals / add-remove metrics/agents)
- Provides sensor meta-data including data quality indicators
- Based on open-source technologies and industry standards:
  - Component architecture built from the ground up as micro-services



Project Information

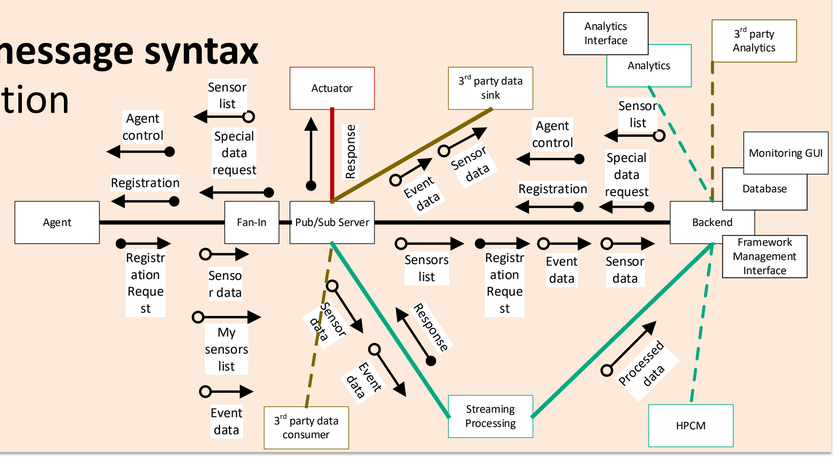
- HPE work package 1.3 as part of the ECP funded PathForward project
- Started 08-2017, ends 07-2020

Architecture



Uniqueness

- **Control channel:** registration and de-registration of Agents; enable /disable individual devices/sensors for collection; and change individual collection frequency
- **Standard and enforced message syntax**
- **Sensor meta-data:** collection frequency; measuring accuracy of the sensor measurement; accepted value range; and value change frequency in # of reported values

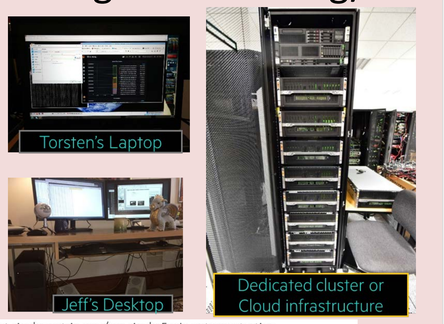


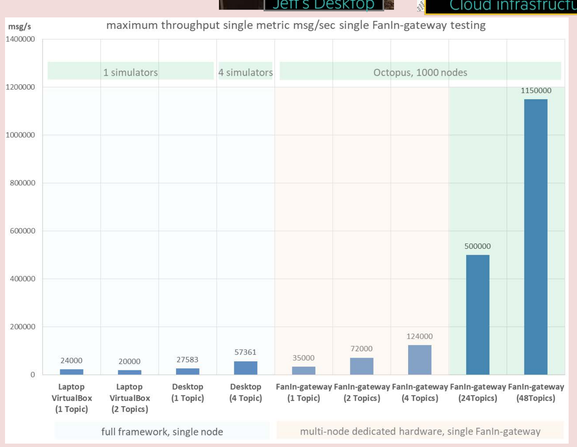
Current Status

- Current Proof of Concept prototype supports time series data, agent registration and de-registration, and streaming analytics
- Working on integrating Elasticsearch and Druid data using Jupyter notebook

Excerpt of preliminary scaling results

- Complete KM framework running inside VirtualBox on a single Laptop (6-threads of a Core i7-8850H) supports max. 24k single metric msg/sec
- A FanIn-Gateway (see Architecture picture) running on a HPE ProLiant DL560 Gen10 (quad socket server board with a 14-core Xeon on each board and 128 GB RAM) can support 1.15 million messages per second using 48 MQTT topics.
- With Kafka's soft limit of 100k publishers, this would translates to at least 1.15 billion individual messages per second when fully scaled out.
- We need 9 FanIn-Gateways with 48 MQTT topics each to reach our design point.





Hewlett Packard Enterprise

EXASCALE COMPUTING PROJECT

PathForward

This work has been supported in part by the U.S. Department of Energy under LLNL Subcontract B621301

+ Repository URL: https://github.com/HewlettPackard/KrakenMare