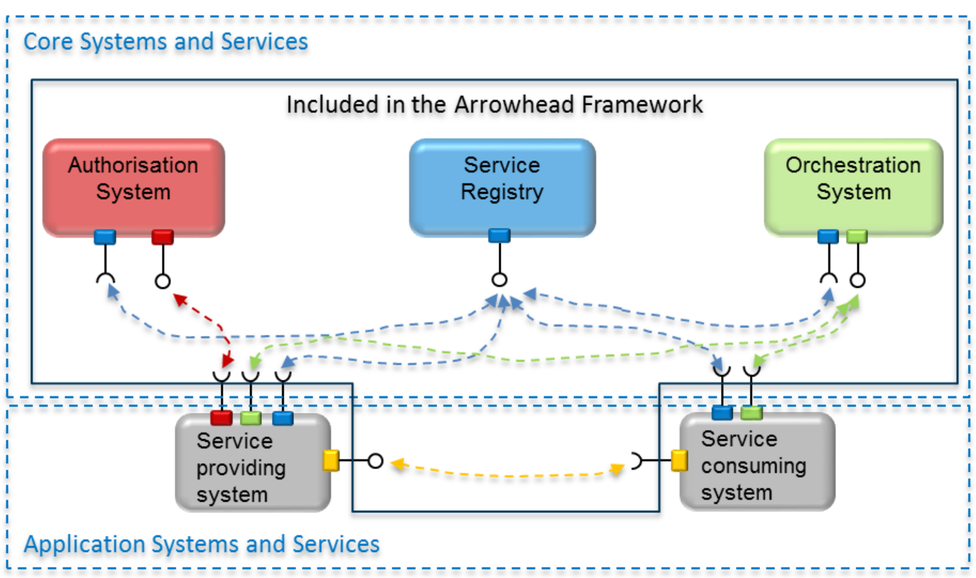
Improving the performance of a Publish-Subscribe message broker

CISTER – Research Centre in Real-Time & Embedded Computing Systems

Rafael Rocha, Cláudio Maia, Luis Lino Ferreira, Pedro Souto, Pal Varga
{rtdrh, crr, llf}@isep.ipp.pt, pfs@fe.up.pt, pvarga@tmit.bme.hu

Arrowhead Framework

- Defines SOA framework for IoT applications;
- Based on 3 core services:
  - service discovery
  - orchestration
  - authentication
- Several other services: QoS Manager, Gate keeper, Event Handler, etc



Improving the Event Handler

Problems:

1. none of the three components (producer, consumer and the Event Handler) reused connections
2. Event Handler creates a new thread for every incoming request

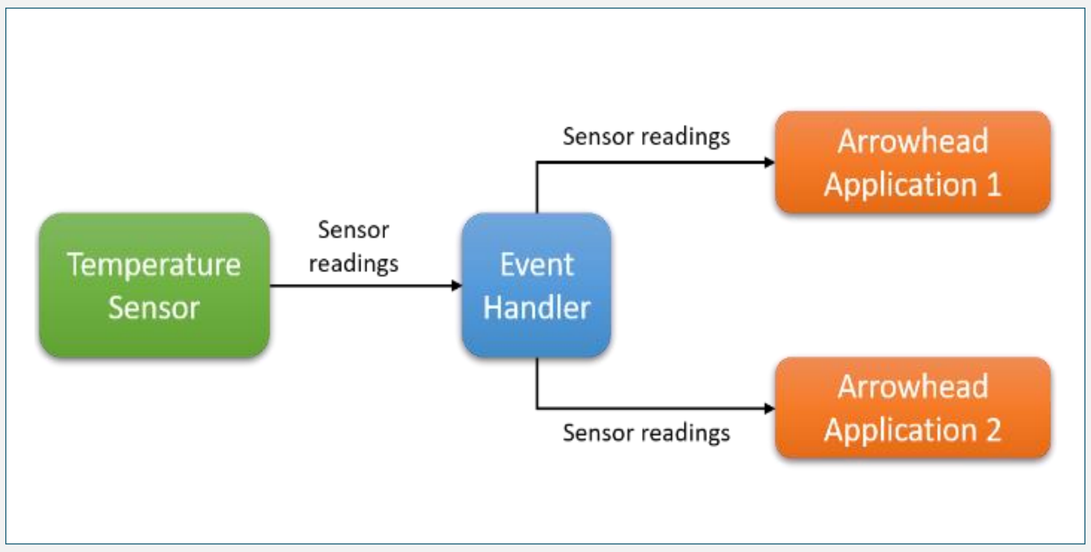
Solutions:

1. A connection pool at the publisher side
   - request for a route for which the client already has a persistent connection available in the pool will be handled by renting a connection from the pool rather than creating a brand-new connection
2. Server-Sent Events in the Event Handler and Subscriber
   - Just by changing the configuration of the Event Handler on Jersey
3. Thread Pool in the Event Handler
   - At the Grizzly HTTP server module

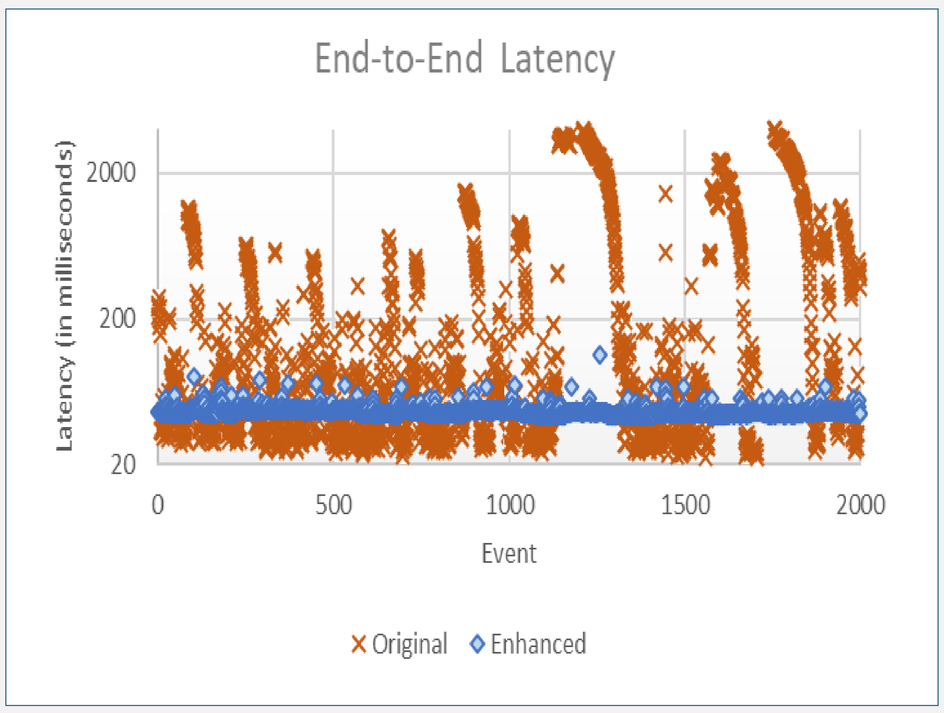
Event Handler

Used for sending periodic updates from a producer service to several consumer services. It is also based on a SOA approach.

- REST/HTTP(S) implementation of a publish-subscribe message broker
- High demanding applications require:
  - High throughput
  - Low end-to-end delay



Experimental results



References

[1] P. Varga, et al, “Making system of systems interoperable – The core components of the arrowhead framework”, Journal of Network and Computer Applications, Volume 81, 2017, Pages 85-95, ISSN 1084-8045.
[2] "Arrowhead Consortia", GitHub, 2019. [Online]. Available: https://github.com/arrowhead-f. [Accessed: 22- Mar- 2019].
[3] The Apache Software Foundation, "Chapter 2. Connection management", Hc.apache.org, 2019. [Online]. Available: https://bit.ly/2TQpBjK. [Accessed: 21- Mar- 2019].
[4] Project Jersey, "Chapter 14. Server-Sent Events (SSE) Support", Docs.huihoo.com, 2019. [Online]. Available: https://bit.ly/2TpoQsK. [Accessed: 21- Mar- 2019].
[5] N. Babcock, "Know Thy Threadpool: A Worked Example with Dropwizard", Nbsoftsolutions.com, 2016. [Online]. Available: https://bit.ly/2Js4Shm. [Accessed: 21- Mar- 2019].
[6] "Why using many threads in Java is bad", Iwillgetthatjobatgoogle.tumblr.com, 2012. [Online]. Available: https://bit.ly/2HCpHVE. [Accessed: 21- Mar- 2019]. .

Co-financed by Unidade de I&D CISTER - CEC/04234:
FCT Fundação para a Ciência e a Tecnologia

CISTER Research Centre
ISEP, Polytechnic Institute of Porto
Rua Dr. Antº Bernardino de Almeida, 431
4249-015 Porto, Portugal
+351 228 340 502
www.cister.isep.ipp.pt
cister-info@isep.ipp.pt
facebook.com/cisterrealtime

isep Instituto Superior de Engenharia do Porto
P.PORTO