ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-TUPHA029

LIVE VISUALISATION OF EXPERIMENT DATA AT ISIS AND THE ESS

M. J. Clarke, F. A. Akeroyd, L. A. Moore, ISIS, STFC Rutherford Appleton Laboratory, Oxon, UK
O. Arnold, N. J. Draper, M. Gigg, M. D. Jones, Tessella, Abingdon, UK
T. S. Richter, European Spallation Source, Lund, Sweden

Abstract

As part of the UK's in-kind contribution to the European Spallation Source [1], ISIS [2] is working alongside the ESS and other partners to develop a new data streaming system for managing and distributing neutron experiment data. The new data streaming system is based on the open-source distributed streaming platform *Apache Kafka* [3].

A central requirement of the system is to be able to supply live experiment data for processing and visualisation in near real-time via the Mantid data analysis framework [4]. There already exists a basic TCP socket-based data streaming system at ISIS, but it has limitations in terms of scalability, reliability and functionality. The intention is for the new Kafka-based system to replace the existing system at ISIS. This migration will not only provide enhanced functionality for ISIS but also an opportunity for developing and testing the system prior to use at the ESS.

INTRODUCTION

Currently under construction, the ESS is an accelerator-based spallation neutron source located in Lund, Sweden. After completion, it will be the World's most powerful neutron spallation source and as such will generate significantly more neutrons per pulse than existing neutron sources [5]. Initially the ESS will have a suite of 15 class-leading beamline instruments which will cover a range of different applications across many scientific and engineering disciplines.

For each neutron experiment, a range of different data need to be recorded to allow the scientist to analyse the results. Naturally, the core of these are the neutron detection events, but they also include vital complementary data such as sample environment measurements, details of the proton pulse from the accelerator, and experiment metadata. For the beamline instruments at the ESS, the intention is to capture each individual neutron detection event and every change in complementary data. Neutron data will typically form the largest contribution to produced data in terms of both frequency and volume, however the other data are equally as important for the scientist. The complete experimental information will be stored on disk using the NeXus data format [6].

As well as having the final data available after the experiment, it is becoming increasingly critical to be able visualise and process them during the experiment in near real-time. Being able to visualise and process data during the experiment allows the scientist to make more efficient use of their beamtime; for example: stopping an experiment early once the counting statistics are acceptable or aborting an experiment early because the sample is not aligned optimally.

To handle the predicted data rates of the ESS, whilst simultaneously enabling live visualisation and processing of data, requires a robust and scalable high-throughput data streaming system which is capable of handling both numerous sources of data and multiple consumers of data. It was decided that Apache Kafka would provide the backbone of the ESS data streaming system as it meets these requirements.

Collaboration of a team at ISIS with the ESS Data Management Group through an in-kind project provides an excellent opportunity to improve on current systems in use at ISIS and to prototype the system ahead of time for the ESS. Like ISIS, the ESS will use the Experimental Physics and Industrial Control System (EPICS) [7] for instrument control and the Mantid data analysis framework for live visualisation and processing. This overlap of technologies permits testing and gaining operational experience with the data streaming system at ISIS prior to it being used at the ESS.

THE SOCKET-BASED SYSTEM

Currently at ISIS, neutron event information from a beamline's detectors is collated in the *Data Acquisition Electronics* (DAE) which provides a time-stamp to each neutron event received. Depending on the requirements of the science the data is either stored as raw events or histogrammed in the DAE. Increasingly for analysis the preference is for the data to be available as events [8].

The operation of the DAE is controlled by the *Instrument Control Program* (ICP). The ICP's primary responsibility is to extract the event data from the DAE and combine them with sample environment measurements and experiment metadata to produce a NeXus file for data analysis. The sample environment data and experiment metadata are stored in a database independently of the ICP and are imported by the ICP when writing the NeXus file.

The ICP was previously extended to stream event data over TCP to visualisation clients, such as Mantid's Live Listener interface. Unfortunately, this simple streaming solution has a number of practical and functional limitations:

- Each new client connection requires its own TCP socket and data buffer, placing additional strain on the resources of the ICP which could cause data to be lost if the ICP cannot process neutron events in a timely manner.
- Clients are only able to view neutron events produced after they first connect; there is no way to "playback" previous data.
- Clients are unaware of any data that has been lost on-the-wire.

ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-TUPHA029

- Only neutron event data is streamed; it does not support sending other data types such as sample environment data or images from the EPICS areaDetector module [9].

These limitations place a restriction on the long-term usefulness of the system at ISIS.

THE NEW SYSTEM

A data streaming system without the limitations highlighted in the previous section is critical for the success of the ESS. This is particularly the case as the limitations would be exacerbated by the high data rates resulting from the ESS's increased neutron flux and new detectors with large solid angle coverage and high pixel counts. In this section the new system will be described in the context of updating the existing system at ISIS prior to its use at the ESS. By both upgrading the existing system and testing the resultant new system at ISIS this approach is of mutual benefit to the two institutions.

Apache Kafka

The basis of the new data streaming system is Apache Kafka, which is a high throughput distributed streaming platform. Kafka is designed for building real-time data streaming pipelines for reliably transferring data between systems and applications. Kafka implements a publish-subscribe architecture similar to a conventional messaging system combined with functionality like that of a distributed filesystem. Figure 1 shows an overview of the Kafka architecture.

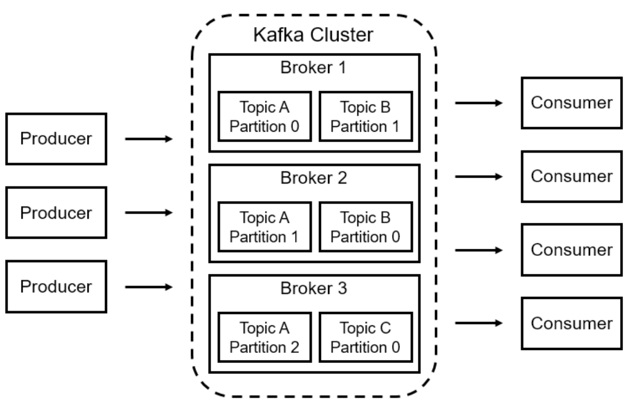

Figure 1: The Apache Kafka architecture.

The core of the Kafka system is the Kafka cluster which consists of one or more servers each running an instance of a Kafka broker. The brokers are responsible for receiving the data from producers and supplying it to the consumers. The brokers store the data streams in categories called topics which consist of one or more structured commit logs known as partitions. The partitions that make up the topic can be distributed across the cluster to enable parallelisation for increased throughput. The partitions can also be replicated across multiple machines for fault tolerant operation. The data retention policy for the topics is configurable; it can be based on the age of the data and/or the amount of data in the topic. Older data is removed first.

The Initial Design

The first implementation of a Kafka-based data streaming system only transmitted neutron event data; thus matching the functionality of the TCP socket-based system. Figure 2 shows the basic architecture of the new Kafka-based neutron event data pipeline.

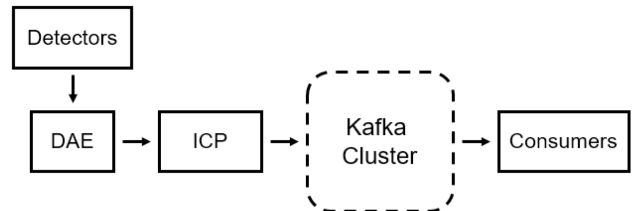

Figure 2: The architecture for the Kafka-based neutron event data pipeline.

By modifying the ICP to export event data as a Kafka producer removes some of the limitations of the simple socket-based system:

- Kafka provides a separation between the ICP and the clients - its operation is now completely unaffected by the number of clients connected.
- As the data is stored on the Kafka cluster, clients joining the stream late can replay the stream from the beginning.
- Kafka brokers and consumer libraries have built-in mechanisms to deal with any data loss in transit.

Kafka is data agnostic and as such does not have a specific serialisation mechanism for encoding the data being passed across it. Instead it simply treats data as a binary blob leaving the serialisation and deserialisation to the producers and consumers respectively. Google's FlatBuffers [10] serialisation library was chosen for the new system as it is flexible and efficient. By designing the appropriate schemata it is relatively straightforward to transport different data types across Kafka.

First Results

To be able to verify the data being sent from the ICP via Kafka was correct, Mantid's Live Listener interface was extended to accept neutron data from Kafka. The Live Listener subscribes to a Kafka topic containing the encoded neutron event data and at a fixed interval inserts the new data into an instance of Mantid's *workspace* data structure. The new data is then accumulated with the previously received data and Mantid's data visualisation display is updated accordingly. To test this Kafka-based prototype data pipeline the modified ICP was installed temporarily on the ISIS instrument LARMOR [11]. Both the modified ICP and the modified Mantid were operational while LARMOR performed a calibration run lasting roughly 30 minutes. During the run, Mantid successfully showed a near real-time, updating detector image

16th Int. Conf. on Accelerator and Large Experimental Control Systems ICALEPCS2017, Barcelona, Spain JACoW Publishing
ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-TUPHA029

of the streamed neutron data from LARMOR's ICP. The final Mantid detector image at the end of the run is shown in Figure 3.

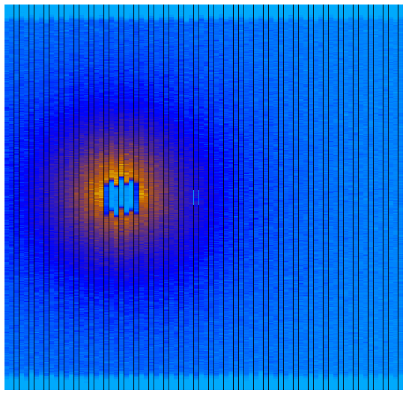

Figure 3: The detector image generated in Mantid from the streamed neutron data produced on LARMOR.

The Design for the Next Version

After successfully streaming real neutron event data, the next step is to incorporate streaming of sample environment data. One way to accomplish this would be to extend the ICP to supply this information; however, this would put extra load on the ICP and differs from how the ESS intends to handle sample environment data. The ESS will stream EPICS sample environment data into Kafka using the EPICS to Kakfa Forwarder [12]. The Forwarder was developed as part of the ESS's BrightnESS programme [13] by Dominik Werder of the Paul Scherrer Institute [14].

As IBEX, the control system developed and used at ISIS [15], is also EPICS-based it is relatively trivial to use the Forwarder at ISIS for putting sample environment data into Kafka. This provides an opportunity for testing the forwarder on operating instruments before it is used at the ESS and, also, gives an overall data streaming architecture similar to that envisioned for the ESS. The basic architecture is shown in Figure 4.

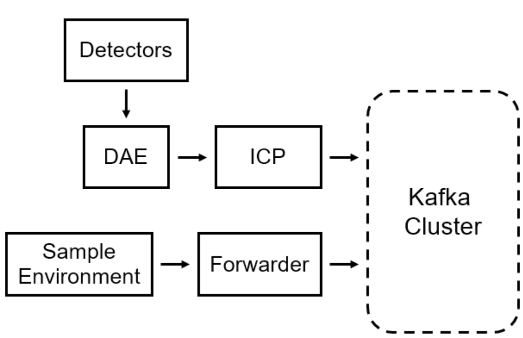

Figure 4: The proposed architecture for the neutron event and sample environment data pipeline.

One of the requirements for both the ESS and ISIS is to be able access the sample environment data stream in Mantid. To achieve this, prototype functionality to consume sample environment data from Kafka was added to the Mantid Live Listener. To fully test the sample environment data pipeline an EPICS device driver was operated in simulation mode to produce simulated data for the Forwarder to consume while Mantid's Live Listener was consuming the appropriate Kafka topic. The simulated data was successfully received by Mantid as shown in Figure 5.

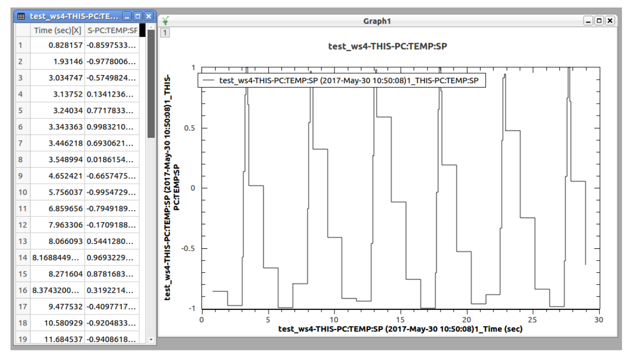

Figure 5: The simulated sample environment data received by Mantid from Kafka.

CONCLUSION

It has been shown that it is possible to stream neutron event data in real-time using a prototype system based on Apache Kafka at ISIS. This new system overcomes the limitations of the existing system and will provide scientists with the means to make best use of their limited beamtime. Incorporating Kafka did not require major changes to the existing software at ISIS, rather it was a matter of extending the functionality already present.

Longer term, the intention is for the prototype data streaming system to evolve to better match the architecture planned for the ESS. As a step towards this work has already been undertaken to incorporate the EPICS to Kafka Forwarder into the ISIS architecture. This allows sample environment measurements and other information from EPICS to enter the data pipeline. As shown in Figure 5, streaming simulated EPICS data through Kafka into the Mantid data analysis framework has been already been achieved.

The data streaming system is a critical component for the success of the ESS; one of the aims of the collaboration is to provide a battle-tested data streaming system for use at the ESS. The intention is for ISIS to be actively using the system prior to its use at the ESS. In the short-term, the new and the existing data streaming systems will be run in parallel on one beamline instrument at ISIS. This will provide the opportunity to modify and test the system without affecting operation of the instrument. Longer term, the new system will fully replace the existing one across all beamline instruments. By adopting the system at ISIS, the two institutions will be able to continue to collaborate beyond the term of the in-kind project.

ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-TUPHA029

ACKNOWLEDGEMENT

The authors wish to acknowledge the contributions of M. Brambilla, M. Könnecke and D. Werder at the Paul Scherrer Institute, and M. J. Christensen, A. H. C. Mukai, J. Nilsson and M. Shetty at the European Spallation Source.

REFERENCES

[1] The European Spallation Source, `https://europeanspallationsource.se`

[2] The ISIS pulsed neutron and muon source, `http://www.isis.stfc.ac.uk`

[3] Apache Kafka, `https://kafka.apache.org`

[4] Mantid, `https://www.mantidproject.org`

[5] E. Klinkby and T. Soldner, "Fundamental physics possibilities at the European Spallation Source", *Journal of Physics: Conference Series*, vol. 746, no. 1, p. 012051, Feb. 2016.

[6] NeXus Data Format, `http://www.nexusformat.org`

[7] EPICS, `http://www.aps.anl.gov/epics/`.

[8] P. F. Peterson *et al.*, "Event-based processing of neutron scattering data", *Nuclear Instruments and Methods in Physics Research, Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 803, pp. 25–28, Dec. 2015.

[9] areaDetector, `https://github.com/areaDetector`.

[10] FlatBuffers, `https://google.github.io/flatbuffers/`.

[11] The LARMOR instrument at ISIS, `https://www.isis.stfc.ac.uk/Pages/Larmor.aspx`

[12] The EPICS to Kafka Forwarder, `https://github.com/ess-dmsc/`.

[13] BrightnESS, `https://brightness.esss.se`

[14] The Paul Scherrer Institute, `https://www.psi.ch`

[15] M. J. Clarke *et al.*, "IBEX - the new EPICS based instrument control system at the ISIS pulsed neutron and muon source", in *Proc. ICALEPCS'15*, Melbourne, Australia, Oct. 2015, paper MOPGF048, pp. 205–208.