White Paper

What and why - semantic modelling

“extensive use of data pipelines will allow for an **end-to-end solution for semantic modelling**, collecting and processing data fast and in volume.**”**

Daniel Riddoch
Date of release 24/01/2025

What do we mean?

Semantic modelling is becoming a more and more important tool in understanding and interpreting the industry of the future. A semantic model gathers all the information relating to a particular device or group of devices, and groups it together. This is then made available with a framework which displays how these device(s) work together to form an overall system.

Semantic models represent a good way to understand the relationships between different parts of the plant, such as the machines, the sensors, the computational equipment, the workers, and the product flow, from raw material(s) to finished products. Furthermore, a semantic model of an operating system may display characteristics about each object in the system, and for machines this can be done from condition monitoring.

How does the problem arise?

A difficulty arising from semantic modelling, is the need to process large volumes of data in parallel. It is vital for the usefulness and validity of the model that all the data relating to each component is up to date and synchronous. This is difficult if, as it would be in most examples, the data is coming from different sources. Secondary to this, the need for the data to be synchronous means that any latency issues risk delaying the entire model, meaning that end users cannot react in real time to events, compromising potential safety or production risks.

Another issue which arises comes from the fact that semantic models can be used to infer meaning beyond that given by the data by relating several different simultaneous events. This means that the data must be of uniform quality, normalised and standardised in the same way. This leads to significant pre-processing demands, which in turn can lead to significant storage costs for data that is caught in the backlog.

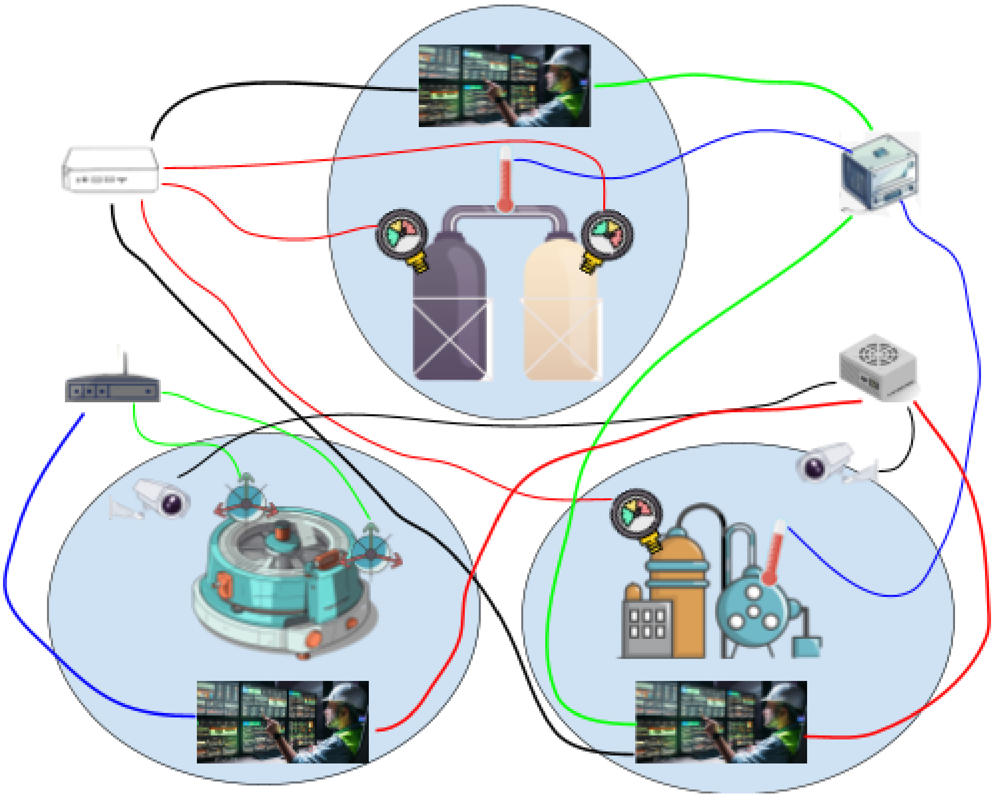

Figure 1: A sketch showing the complexity of a semantic modelling approach using traditional infrastructure

Finally, semantic modelling calls for a description of an entire process, plant, or facility. In many cases, this will involve lots of identical or very similar machines performing similar tasks in real time. This requires lots of repetition of code and logic to introduce all these machines into the model, and the model is therefore harder to scale.

How do we solve the problem?

Common to several of these issues, is the idea of data needing to be processed quickly, automatically, and separately from each asset. This is difficult in an environment where data is collected and sent to a central processing location centrally.

It is much easier to process the data separately, close to the collection location on the edge, and send the resulting cleaned data or calculated metrics. This can be achieved using intelligent data pipelines, which can intercept the data soon after collection.

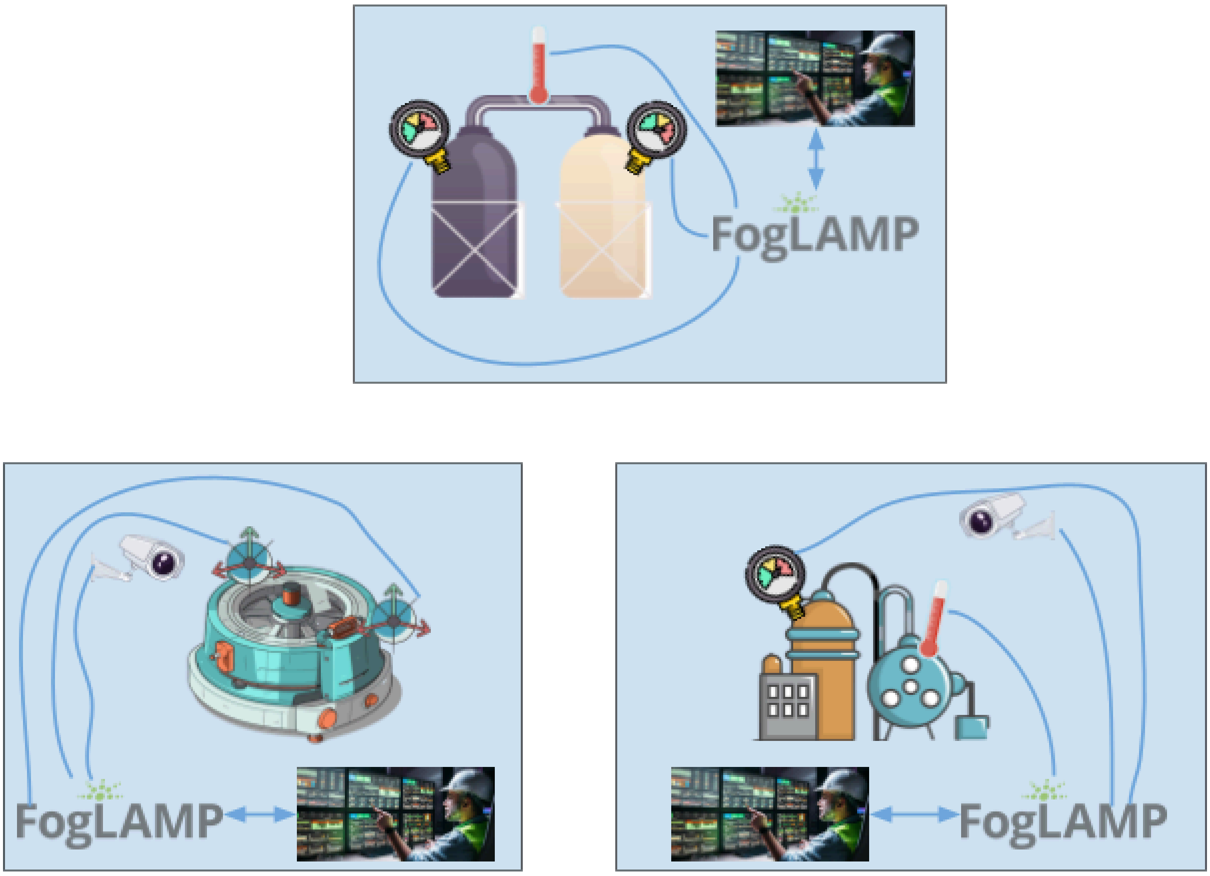

Figure 2: A sketch showing how a semantic model can be architecturally simplified using intelligent data pipelines

Data pipelines can be configured to perform almost any task, from basic conditional forwarding up to ML/AI inference. As such, an extensive use of data pipelines will allow for an end-to-end solution for semantic modelling, collecting and processing data fast and in volume.

What other benefits arise?

Using intelligent data pipelines will also allow for easier scalability and repeatability. Using management products, several identical pipelines can be deployed and managed 'as one,' meaning that a large and complex system can be built out trivially, eliminating risks of transcription error.

The ability to build such systems and scale will mean that a richer and fuller model may be built, generating more insights, understanding and control of processes and production. Data pipelines also eliminate the need for bespoke connectors or adapters, meaning that over a large model, significant cost savings may be achieved.

Example

The following example comes from a real use case, where Réseau de Transport d'Électricité(RTE) teamed up with Dianomic and the Linux Foundation to update their existing substation infrastructure to standardise in line with IEC 61850.
https://lfenergy.org/how-french-transmission-system-operator-rte-leverages-lf-energy-and-lf-edge-to-build-next-gen-substation-monitoring-and-controls/

IEC 61850 came from the need to unify the protocols that various different organisations used to provide communication and keep network infrastructures working well. It evolved into a set of communication protocols which effectively define a standard semantic model for electricity substations internationally.

The communications procedures provide interoperability and connectivity between devices within the substation, but it also describes the functionality of the equipment and provides the framework for a model. The protocol defines three levels, the process, bay and station levels, and it defines communications standards between these three levels.

Figure 3: A sketch showing the arrangement of devices within an IEC 61850 substation(Credit: SGRwin)

The process level is the most basic, and exists to extract data from sensors and other equipment in the substation. This level is not purely extractive and its other task is to receive control commands and execute them. As such, this can be thought of as the machine level. The process level is connected to the bay level by the process bus.

The bay level consists of intelligent electronic devices(IEDs). These can process data supplied by the process level and make local control decisions. Often these devices can be broken down into an electrical appliance(eg. transformer) and their associated control equipment and switching gear. This level is connected both to the process level by the process bus and to the station level by the station bus.

The station level can be split in two; the process functions, and the interface functions. Process functions combine the data from multiple bay level devices to make more educated decisions which can adjust multiple devices to ensure maximum efficiency, service quality and reliability. Interface functions are those which are responsible for the interaction of the system with human operators.

In other papers in this series we have discussed how FogLAMP by Dianomic was used by RTE within the Linux Foundation Energy's FLEDGEPower project. In that case the idea was to use FLEDGE and FogLAMP to create a unified namespace. However, it is equally true that the new IEC 61850 compliant system can act as a semantic model.

In this case, each IED was given its own FogLAMP instance, as the software can run on gateway infrastructure, as were each of the control devices which make up the station level. In some cases, the process level is best combined into the bay level(eg. The controller of an IED may give process level data but is best thought of in the bay level), and other groups of process level devices can be dealt with by a separate FogLAMP.

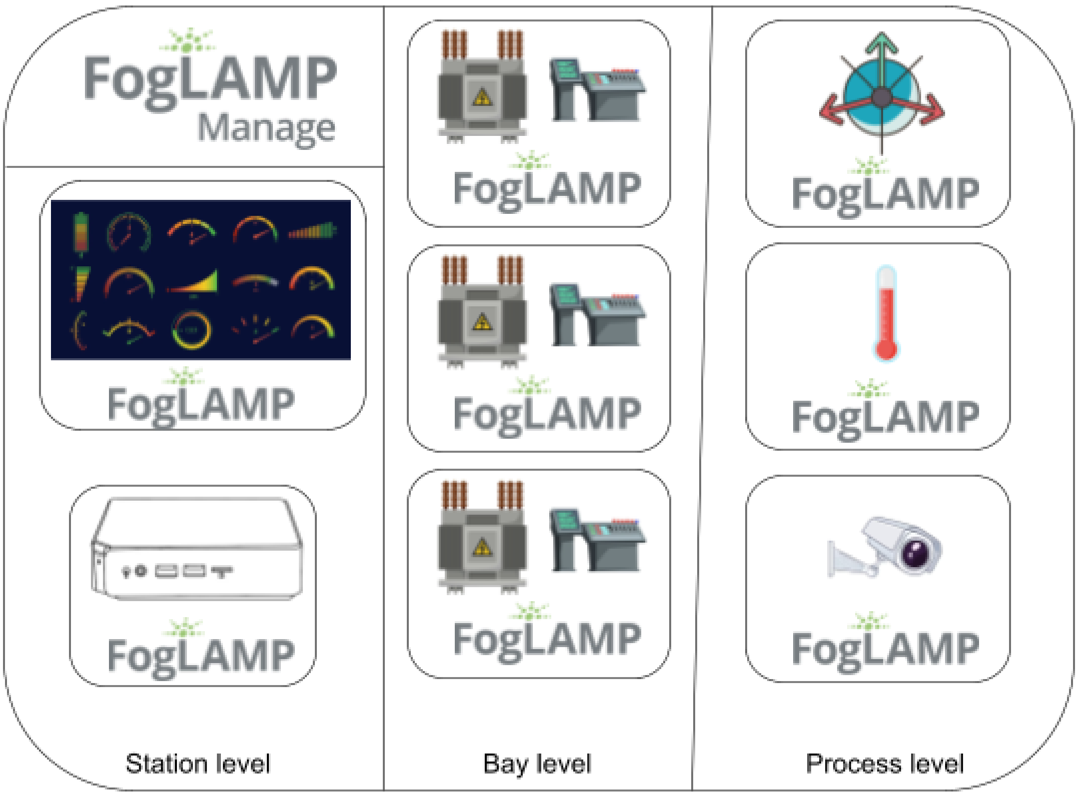

Figure 4: A sketch showing the architecture that can be created using the FogLAMP suite to facilitate semantic modelling in line with IEC 61850

These various FogLAMP instances can then be controlled centrally using FogLAMP Manage. This is an environment which allows for the control of several FogLAMP instances at once, meaning that the data from each device is dealt with internally, and can be viewed centrally, associated with each device. This gives the framework for a semantic model to be built, allowing RTE to leverage the benefits of semantic modelling to understand how devices within substations are performing and understand better the interactions between them.