

Middleware “EdgeWare”

The EdgeWare serves three main purposes. Firstly, it gathers raw-data from the driver. Secondly, the raw-data is filtered by a generic filtering mechanism. Finally the events that pass through the filter are dispatched to different destinations.

The filtering and dispatching tasks are generic and configurable. This ensures an EdgeWare that is easily adaptable to different fields of application. A range of filters and configurable destinations shall be part of future extensions. The EdgeWare has a layered architecture that includes the following components (from bottom to top):

The Java controller driver

is adapted to comply with the Java Connector Architecture (JCA). JCA has many advantages of which connection pooling and transaction handling make up some of the mechanisms that can be made use of.

A filter queue

is used to persistently save raw tag data that is placed in the queue before filtering takes place. Java Messaging Service (JMS) is used for the queues used in the EdgeWare. A Message Driven Bean is triggered when the raw data is deposited into the filtering queue. This bean is responsible for instantiating the appropriate filter that has been defined for the raw data.

The filtering framework

is a generic filter container in which newly defined filters can be deployed. The deployed filters are applied to the raw data received from the filter queue. Filtering is done making use of the Drools rules engine also known as JBoss Rules. This rules engine offers a unique language with which rule sets can be defined. These rule sets are then used to filter the raw tag data. The standard “Tag-In” and “Tag-Out” filters are part of V1.

Parameters for “Tag-In” :

- Number of transmissions for which the filter notices “Tag-In”
- Duration in seconds

Parameter for “Tag-Out” :

- Time (in seconds) without transmissions

Other parameters for event driven messages like the definition of a temperature range, or the detection of a motion is also part of the system.

The destination queue

is used to persistently save filtered data that is placed in the queue by the filtering layer. As with the filter queue, JMS is used for the destination queue. A Message Driven Bean is triggered when filter data is deposited into the destination queue. This bean is responsible for choosing the appropriate destination that has been configured for the filtered data.

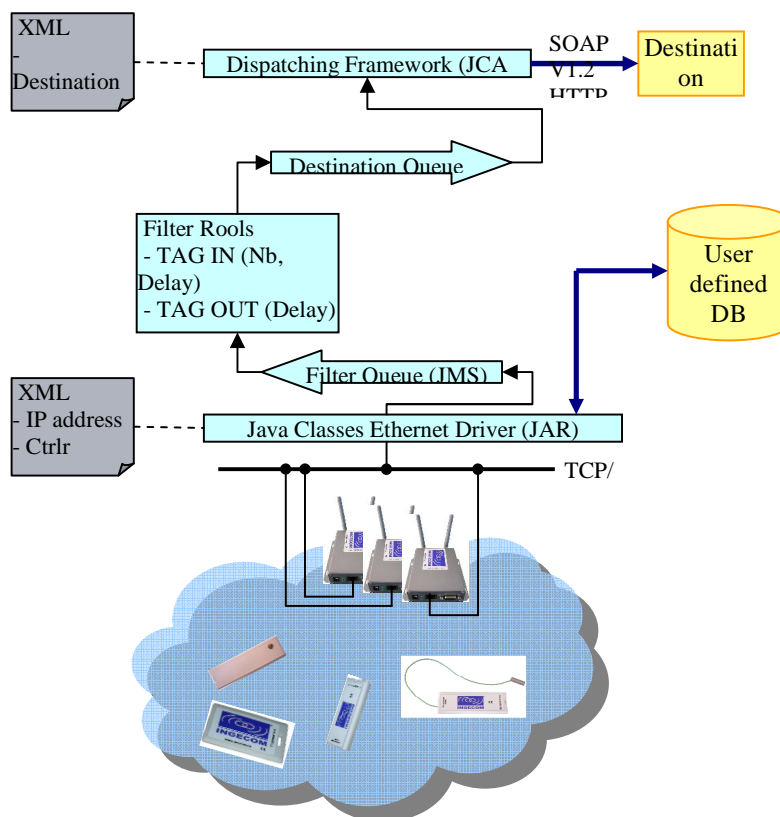
The dispatching framework

makes use of JCA connectors that support different destinations. These connectors are configurable and deliver data in a uniform format. The destinations are required to conform to this pre-defined format.

EdgeWare supports techniques as described above and the API uses SOAP V1.2 with standard internetworking protocols such as HTTP1.1 and TCP/IP. There is API message exchanges between a client application and the EdgeWare service. The EdgeWare service supports Query requests.

EdgeWare supports queries. The structure of the message includes

- o query name
- o filters → meaning the service shall only return those tags which are matching the filter criteria
- o field type (for future enhancements)
- o sort type



Order number:

R-AN-2000