

## The 10 Million Tag System aka the Enterprise Control Foundation

by Timo Klingenmeier, General Manager / Software Architect, inmation



**Timo Klingenmeier**  
General Manager  
Software Architect,  
inmation

**inmation**

Today, we want to talk about scale. Scale matters in the real-time information world. Scale matters with process data. Scale matters for large enterprises.

For the ones of you, who have not spent the last years close to automated production and control systems: The term "Tag" refers to a named item or object in this domain – usually associated to some moving data properties and classified by a certain designation system. Designation systems in industry can be compared to stock market tickers. Everybody understands that MSFT.NAS refers to the share price of Microsoft at the NASDAQ stock exchange. Similarly, a control engineer understands that TC40101.PV is the current process value of the controlled temperature of tank 10 in area 40 of a distinct production plant. Or so. Unfortunately, industrial designation systems are not finally regulated on a global scale, as stock ticker symbols are. Money always wins.

So basically, if you would centrally collect all your worldwide operations measurements and controlled objects data *in one system*, you end up having to organize millions of tags and their associated data. This was merely impossible in the past. For three main reasons.

First, full data historization for readings out of control systems started in the early years to be a discipline executed on the production floor, close to the control systems. Plant engineers added a plant historian to their plant control system. Fine, this enabled the local workforce to better understand complex processes, debottleneck and improve. Data has always been the key to efficiency. Second, the network bandwidth which is required to transport loads of real-time data from production sites to the corporate headquarter and the analysis applications back in form of SaaS, which is also centrally managed for dozens and hundreds of remote sites has either not been available or was too costly to consider. This has changed.

Third, software systems which are able to scale to the true enterprise level were not available. Multiple servers had to be installed even on medium size production sites.

They had to be individually managed. Today, they are actually individually managed, usually by different system integrators, depending on the geographic region. Data protection and QoS are by nature not so well handled as it could be in a central Data Center, managed by the central IT group of the corporation. This situation also led to a certain gap between the local engineers and central IT – the two 'domains' which could generate so much additional value on the basis of an integrated, global, corporate data store... are never getting there.

Costly individual maintenance of on-site dinosaur software systems simply goes on, and the precious gains from a potential corp-wide integration are left on the plate. Which CEO or CIO would not dream of a unified, fully-integrated real-time spinal chord of the entire enterprise? Including better management and 100%-secure access control based on corporate standards. Creating a single source for all Business Intelligence processes? New fields for advanced process automation, steered directly from the ERP/SupplyChain backbone? Does anyone not smell untapped profit here?

The new trends of Big Data, Industry 4.0 and the (Industrial) Internet of Things are giving industries a wake-up call. But what really can add to the bottom line is the unleashed potential of 360° Real-time Enterprise Control.

It is time for Enterprise Control comprising all operational assets and secondary processes, even integrating different control domains. The process and the buildings, the smoke detectors, the energy management systems. External price information and control loops. Seismic and meteorological warnings from thousand miles away. Whatever data, if it is tag-based, or to be turned into a tagged information we can

use it in our infinite world of Enterprise Control. The only ingredient missing is the software system which holds it all together in a uniform manner.

What does it take to create an Enterprise Control system of any size?

The enterprise control system operates on a uniform, but highly distributed real-time/near-time data acquisition system plus a 100%-waterproof communication infrastructure plus an unlimited size data store plus the intelligence it takes to serve this huge address space of information to the corporate workforce. Plus a flexible layer which routes information access to the right source, given a sustained security context from top to bottom, from Rio to Shanghai.

First, we must be able to connect to any real-time, near-time and sporadic data source. For control systems, this all translates to OPC (either COM-based "classic" servers of Unified Architecture servers), but includes all interfaces (such as real-time data, alarms and events plus time-series history). For relational databases, it mostly means OLEDB/ODBC, plus JSON for the more modern ones. We need to connect Web-Services in order to collect data from external sources. And anything else can be taken from structured files (structured text, XML or JSON). The system must support to connect to such sources in an unlimited fashion. And, most important to be a true Enterprise Control system, no local configuration may be ever invoked at data source level. We need to integrate existing systems as they are. Eventually adding an off-the-shelf interface to it is the most we want to accept. Any interface configuration must be central. Dropping an executable to a remote hardware is all what we want to do. No compromise here! Not even a license-key we want to apply.

Next. We need to be able to transport the real-time data in bidirectional fashion. We want our system to be able to issue

remote control sequences. Obviously, it is not intended to start an exothermic chemical reaction from thousands of miles away by a junior fellow playing with the system in the headquarter cantina. Sure, the StuxNet demon appears on stage, once someone starts to mention remote control. But, it must be possible to initiate certain processes automatically from remote. Without downlink writing, there is no Enterprise Control. The answer is uncompromised security whenever a wire leaves the controlled space on its way to the remote controller. This includes a lot of checklists, but – it must only be standardized and maintained once. Not to forget No-Spy agreements with all involved parties and periodic penetration team missions on hire.

And inside the system (including all interfaces it supplies for the integration on Enterprise IT level), per-object security control must be implemented. Each user accessing the system in whatsoever fashion must be authenticated against the Enterprise AD. No compromise here!

The system must be built not

making any assumptions that remote links are always available. It must provide easy-to-maintain, self-monitored and multi-strategy redundancy. It must buffer collected data which can not be routed upwards. It must reject control commands which can not be forwarded downwards.

All events of our Enterprise Control system must be collected centrally and also serve as a single source for any emergency notifications. No, dear vendor, a text file on the harddisk is not considered a log.

In a similar fashion all other data has to be stored centrally. Time-series, alarms, events, aggregates, forecasted values, structural data, metadata... It must all be stored using the same database system. Guess, this is not SQL-based. We

want to cluster to the infinite, quickly and easy to maintain. Horizontally scaled, using multiple instances of moderately muscled machines. We want to add to it as required. An Enterprise Control system is not built at one shot and then handed over. An Enterprise Control system evolves every day. The database system

can only be a document-oriented, modern NoSQL database, able to flexibly store all kinds of data types and formats. Built for scalability, clustering and sharding. MongoDB is such a product.

Using MongoDB, the backbone of the Enterprise Control system is a sharded database, eventually consisting of dozens or even hundreds of instances, but forming one logical database. It can be distributed over multiple data centers in different world regions to be disaster-proof. Backups? We do not want to do backups. We use replicated storage for the on-premise data and hybrid storage for long-term storage. Old data is encrypted, anonymized and stored in the cloud. Or, in case even this is not wanted, in a private cloud instead.

Halt! Our Enterprise Control system evolves. How can it do so, if only highly specialized staff can ever tame the beast? It is true, the internal corporate IT and their partners must be the owner and skilled personnel from operations, engineering, finance and administration must be the stakeholders. And yes, a good, super-regional

system-integrator may still be a good option. But finally, the system must – as central and large it internally is – also allow for partial administration by division, site, branch or department. Definable at object level, again. It must be able to control security to the finest grain, but open and extensible to the infinite.

The developers at inmation have been in industrial system integration business for more than two decades. We have learned from global real-time data infrastructure projects on behalf of industry leaders. Fueled by this spirit, system:inmation, our own new product is designed to enable Enterprise Control in a much simpler way than this was ever possible before.



system:inmation

www.inmation.com