Addressing the Data Distribution Challenge in SCADA and Utilities Systems

A. Corsaro, Ph.D.

angelo.corsaro@prismtech.com
Chief Technology Officer
PrismTech
4 rue Angiboust, 91460 Marcoussis. France

Abstract: SCADA and Utilities Automation systems, such as power grid control, industrial automation, and high throughput telemetry systems, must increasingly meet the need to (1) deal with more data in a timely and predictable manner, often over a widening geographical scale, (2) provide secure communication, (3) provide filtering capabilities that help optimise data distribution and sense-making, and (4) reduce time-to-market. Conventional infrastructure software technologies fall short in addressing most, if not all, of these requirements due to their lack of flexibility and configurability, limited support for dealing with large volumes of data, lack of security, data filtering, and limited expressiveness and productivity. This paper introduces the OMG Data Distribution Service for Real-Time Systems and highlights how its features ideally address the key requirements of existing and next generation SCADA and Utilities Applications.

Keywords: Real-Time Data Distribution, OMG DDS, SCADA, Utilities, Middleware

1. Introduction

SCADA and Utilities Automation systems, such as power grid control, water and wastewater management, industrial automation, railway control systems, and high throughput telemetry systems, must increasingly meet the need to (1) deal with more data in a timely and predictable manner, often over a widening geographical scale, (2) provide secure communication, (3) provide filtering capabilities that help optimise data distribution and sense-making, and (4) reduce time-to-market. Conventional infrastructure software technologies fall short in addressing most, if not all, of these requirements due to their lack of flexibility and configurability, limited support for dealing with large volumes of data, lack of security, data filtering, and limited expressiveness and productivity.

The steadily growth in adoption of mainstream COTS (Consumer-Off-The-Shelf) technologies for the communication layer, such as, Ethernet, and Industrial Ethernet, along with the established trend toward processor architectures such as PPC- or x86-microprocessors, is enabling the use of software technologies that provide extended functionalities along with improved productivity, thus opening a new set of possibilities for architecting and implementing distributed real-time sensing/monitoring and control systems such as those found in SCADA and Utilities.

The Data Distribution Service (DDS) for Real-Time Systems [1,2,3], a standard-based publish/subscribe middleware defined by the OMG, is a COTS technology that is experiencing a constantly growing adoption in SCADA and Utility applications. This success is mainly due to its ability to address the challenges outlined above by providing (1) a very powerful data distribution and event processing model, (2) extremely high performance, (3) hard-real-time determinism, and (4) a rich set of non-functional properties fully configurable by means of Quality of Service (QoS) policies, used to control properties, such as, deadline, data prioritisation, data availability, as well as resource (network, CPU, and memory) usage and reservation. In addition, leading implementations of this standard [4] also provide information assurance thus securing data communication, enforcing access control, and information flows.

2. OMG DDS: The Data Distribution Service for Real-Time Systems

2.1 Overview

The Data Distribution Service for Real-Time Systems (DDS) is an Object Management Group (OMG) standard for Publish/Subscribe (Pub/Sub) that addresses the needs of mission- and business critical applications, such as, financial trading, air traffic control and management, and complex supervisory and telemetry systems. That key challenges addressed by DDS are to provide a Pub/Sub technology in which data exchange between producers and consumers are:

- **Real-time**, meaning that the right information is delivered at the right place at the right time--all the time. Failing to deliver key information within the required deadlines can lead to life-, mission- or business-threatening situations. For instance in a financial trading 1ms could make the difference between loosing or gaining $1M. Likewise, in a supervisory applications for power-grids failing to meet deadlines under an overload situation could lead to severe blackout such as the one experienced by the northeastern US and Canada in 2003 [1].

- **Dependable**, thus ensuring availability, reliability, safety and integrity in spite of hardware and software failures. For instance, from the reliable functioning of an Air Traffic Control and Management System depends the life of thousands of people flying over the area it is managing. Thus these systems have to ensure 99.999% of availability and guarantee that regardless of experienced failures critical data is reliably delivered.

- **High-Performance**, hence able to distribute very high volumes of data with very low latencies. As an example, financial auto-trading applications have to deal with millions of messages per second each of which has to be delivered with minimal latency, in the order of tens of microseconds.

The OMG DDS standards family is today composed, as shown in Figure 1, by the DDS v1.2 API [2] and the DDS Interoperability Wire Protocol (DDSI v2.1) [3]. The DDS API standard guarantees source code portability across different vendor implementations, while the DDSI Standard ensures on the wire interoperability across DDS implementations from different vendors. The DDS API standard defines several different profiles (see Figure 1) that enhance real-time pub/sub with content filtering, persistence, automatic fail-over, and transparent integration into object oriented languages.

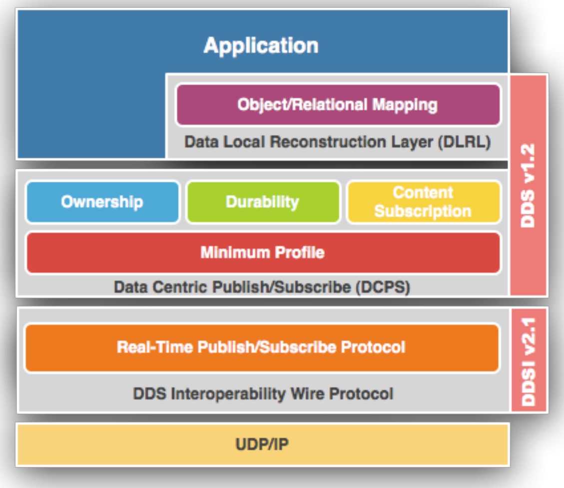

Figure 1: The DDS Standard

The DDS standard was formally adopted by the OMG in 2004. In less than 6 years, it has become the established Pub/Sub technology for distributing high volumes of data, dependably and with predictable low latencies in applications such as, Radar Processors, Flying and Land Drones, Combat Management Systems, Air Traffic Control and Management, High Performance Telemetry, Large Scale Supervisory Systems, and Automated Stocks and Options Trading. Along with wide commercial adoption, the DDS standard has been recommended and mandated as the technology for real-time data distribution by several key administrations worldwide, such as the US Navy, the DoD Information-Technology Standards Registry (DISR) the UK MoD, the MILVA Vehicles Association, and EUROCAE--the European organisation that recommends standards for air navigation systems.

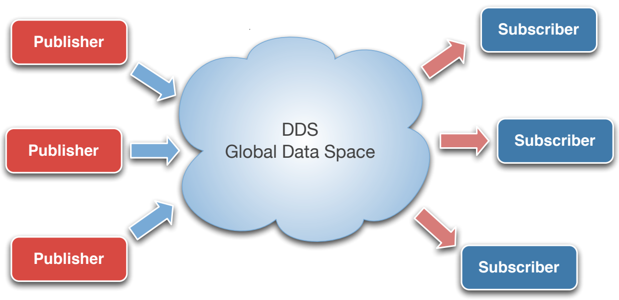

Figure 2: The Data Distribution Service Conceptual Model

As shown in Figure 2, the key abstraction at the foundation of DDS is a fully distributed Global Data Space (GDS). It is important to remark that the DDS specification requires the implementation of the Global Data Space to be fully distributed to avoid the introduction of single point of failure or single point of bottleneck. Publishers and Subscribers can join or leave the GDS at any point in time as they are dynamically discovered. The dynamic discovery of Publisher and Subscribers is performed by the GDS and does not rely on any kind of centralised registry such as those found in other pub/sub technologies such as JMS. Finally, it is worth to mention that the GDS also discovers application defined data types and propagates them as part of the discovery process.

In essence, the presence of a GDS equipped with dynamic discovery means that a deployment time, no configuration is required for the system. Everything will be automatically discovered and data will begin to flow. Moreover, since the GDS is fully distributed, nobody has to fear the crash of some server inducing unknown consequences on the system availability -- in DDS there is no single point of failure, although applications can crash and restart, or connect/disconnect, the system as a whole continues to run.

When compared with other publish/subscribe technologies, DDS crisply stands out because of (1) its powerful support for data modelling, event filtering and processing, and (2) its support for a very rich set of QoS which allow to configure the most important properties of data distribution, such as its temporal properties, its priority, its availability, and its presentation.

2.2 Data Modelling

DDS perfectly blends and extends the most useful features found in real-time messaging middleware and relational databases. From real-time messaging middleware, DDS draws the efficiency in distributing data, the low-latency, the real-time determinism, and the throughput. From relational databases, it draws the ability to define relational data models and operate on them via SQL92 expressions to specify content-based subscriptions, join, projection, filters, and queries (see Figure 3). DDS allows the application programmer to create object views from a relational information model. This ensures that the applications logic can completely focus on the business problem, while delegating to the middleware the mechanics of data distribution. Another typical use of the object-oriented view is to reconstruct and manage relationships out of the messaging data, perform complex filtering, and correlation. As shown in Figure 3, DDS can perform data filtering and correlation by relying on either SQL92 expressions, or on user provided filter-objects.

All these capabilities are provided via a fully distributed architecture that ensures performance, predictability, availability, and scalability, and are controlled and finely-tuneable by a rich set of QoS policies that allow traffic prioritisation, traffic shaping, hardware and software filtering, and persistence.

The native support for relational information modelling makes it very efficient to seamlessly integrate DDS implementations with databases. Moreover, the combination of data filtering—by means of content-based subscriptions—and event processing—by means of filter-objects—allows the execution of arbitrary complex event filtering and correlation in the middleware, thus providing the application with only the events it is really interested in.

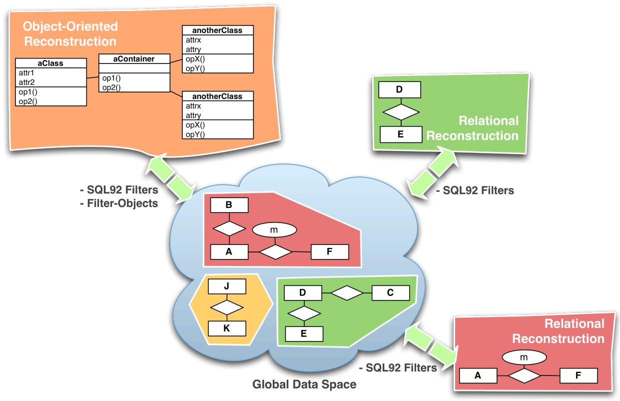

Figure 3: DDS Relational and Object-Oriented Data Modelling

The DDS Data Modelling features enable scalable modular application designs that effectively separate the data acquisition processing and communication modelling from the business or engineering processing that transforms acquired data into strategic control information. For example, large-scale SCADA systems are often combined with Energy Management Systems (EMS) that transform sensor data into control strategies for efficient and safe control of generation and transmission resources. Separation of data acquisition and transport concerns from complex data transformations are combined with the powerful relational information modelling capability of DDS to enable highly advanced application frameworks for evolving SCADA/EMS applications.

2.3 Quality of Service

Application domains like SCADA/EMS can rely upon DDS as a sure foundation for robust and reliable data acquisition and transport functions. DDS provides a rich set of QoS policies for finely tuning and controlling the data temporal properties, e.g., deadline, latency budget, etc., its priority, reliability, persistence, and resource usage. This set of QoS policies provide the control knobs that allow an architect to specify how the system should behave, thus making it possible to easily express the constraints derived from schedulability analysis and alike.

Two key sets of DDS QoS policies relevant to SCADA and Utilities applications are those controlling data timeliness and data availability.

DDS provides a rich set of QoS policies that control the timeliness properties of distributed data, the trade-off between latency and throughput, and the data prioritisation:

- The LATENCY_BUDGET QoS policy provides a means for applications to inform DDS of the urgency associated with transmitted data. The latency budget specifies the time period within which DDS must distribute the information. This time period starts from the moment the data is written by a publisher until it is available in the subscriber's data-cache ready for use by reader.

- The TRANSPORT_PRIORITY QoS policy allows applications to control the importance associated with a topic or with a topic instance, thus allowing a DDS implementation to prioritise more important data relative to less important data.

- The DEADLINE QoS policy allows applications to define the maximum inter-arrival time for data. DDS can be configured to automatically notify applications when deadlines are missed.

These QoS policies make it possible to ensure that (1) the system is tuned to operate in the right latency/throughput trade-off point, and (2) that important data is always able to preempt less important data, thus ensuring that even in overload condition the temporal properties of critical data are preserved. As an example, the combination of the LATENCY_BUDGET, DEADLINE and TRANPORT_PRIORITY can be used to set the properties needed to enforce the schedulability of data traffic as derived from a Rate Monotonic Analysis (RMA).

DDS provides the following QoS policies that control the availability of data to subscribers:

- The DURABILITY QoS policy controls the lifetime of the data written to the global data space. Supported durability levels include (1) volatile, which specifies that once data is published it is not maintained by DDS for delivery to late joining applications, (2) transient local, which specifies that publishers store data locally so that late joining subscribers get the last published item if a publisher is still alive, (3) transient, which ensures that the global data space maintains the information outside the local scope of any publishers for use by late joining subscribers, and (4) persistent, which ensures that the global data space stores the information persistently so it is available to late joiners even after the shutdown and restart of the system.

- The LIFESPAN QoS policy controls the interval of time during which a data sample is valid. The default value is infinite, with alternative values being the time-span for which the data can be considered valid.

- The HISTORY QoS policy controls the number of data samples (i.e., subsequent writes of the same topic) that must be stored for readers or writers. Possible values are the last sample, the last n samples, or all samples.

These QoS policies control the degree of time decoupling between publishing and subscribing applications, as well as the historical data that the middleware should keep on behalf of the application. Several other QoS policies exist that control (1) the replication of publishers, greatly simplifying the implementation of Hot-Swap and Hot-Hot replication styles, and (2) the use of computational resources, such as main memory, buffer sizes.

These DDS QoS policies allow highly decoupled and focused data transformation applications to be designed and deployed in a truly distributed architecture that leverages available computing and communication hardware technology with data transport agnosticism. An example of this is the SCADA/EMS application domain where it is essential that intensive compute and data hungry optimisation algorithms or electrical system contingency analysis solutions be decoupled architecturally from relatively simple but time critical resource allocation control operations.

2.4 Traffic Engineering

In addition to the DDS QoS policies that control the temporal properties of data, it is worth investigating how these QoS are leveraged by some of the most advanced DDS implementations [4] to enforce specific network traffic properties.

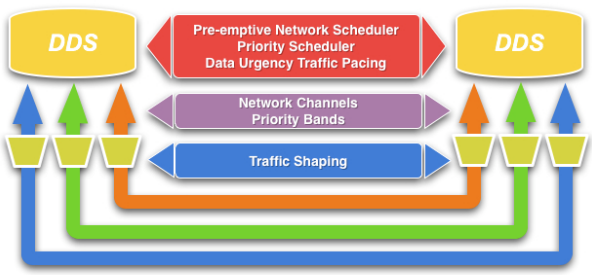

Figure 4: DDS Networking Technology and Traffic Engineering

- **Latency/Throughput Tradeoff**. Advanced DDS implementations, rely on the LATENCY_BUDGET QoS to bundle data across topics and applications to ensure optimal throughput and reduce CPU utilisation.

- **Data Prioritisation**. The TRANSPORT_PRIORITY QoS is used in combination with network channels to enforce messages priority even on non-priority-preserving transports, such as the TCP/IP or UDP/IP.

- **Traffic Shaping**. For every channel it is possible to define the traffic profile to ensure that the network utilisation never exceeds a user-specified value.

- **Traffic Partitioning.** DDS partitions are mapped to IP multicast addresses to segregate different traffic flows and rely on hardware filtering capabilities provided by modern routers, switches and network cards.

The combination of these architectural features and QoS ensure that DDS applications have full control over the characteristics of generated traffic. As a result, the system can be properly tuned and configured to operate at the right performance point, while retaining stability and determinism during overload conditions.

2.5 Information Assurance

Leading DDS implementations [4] provide security solution ensuring Information Assurance (IA) for all DDS-based cooperation and information exchange between the DDS nodes over untrusted communication infrastructures. The security solution allows the reliable separation of applications with different clearances deployed on different nodes in a way that ensures transparency to the applications, thus supporting full portability. This solution provides QoS-enabled IA offering end-to-end security between all applications (distributed or co-located), including mandatory access control for all data flowing between applications and detailed security audit of application interactions.

The security solution extend the real-time communication with configurable cryptographic protection implementing the following security properties:

- Information exchanged over unsecured networks cannot be eavesdropped or modified without detection while in transit (data integrity & confidentiality).

- Complete, reliable, and readily evaluable separation between the area in which the information is processed in unencrypted form (RED, on the node) and the area to which critical (classified) information is not permitted to flow in unencrypted form (BLACK, network). \

- Information of different classification is cryptographically separated while in transit between different nodes, resulting in stronger separation than only labelling and performing no infiltration or ex-filtration between different classifications while in transit.
- Information received via the network can only be retrieved in its unencrypted (RED) form on nodes that:
    - are accredited for the security level of this information, and
    - host applications that have a need-to-know for the information.
- Authenticity of all information exchanged between nodes.

Security and reliability of critical infrastructure is of paramount importance today. Recent developments of infrastructure security and protection standards require Utilities compliance. It is no longer just good business but critical to national security to protect infrastructure resources. DDS Information Assurance is a large step toward securing SCADA/EMS and advanced protection systems that provide the backbone of the Utility infrastructure. SCADA/EMS applications design can rely upon application authentication and transport encryption provided with DDS IA. The separation of concern between DDS and application data transformation allows SCADA/EMS application developers to focus on secure control strategies with the knowledge that underlying data transport is secure.

3. Concluding Remarks

The Data Distribution Service (DDS) for Real-Time Systems is a standard data-centric, topic-based publish/subscribe middleware that has been designed for addressing the data distribution challenges of real-time distributed systems. Due to its high performance, powerful support for data modelling, event filtering and processing, and its finely tuneable control over key properties of data distribution, such as deadline, priority, latency, availability, and security, DDS is the perfect technology for addressing the data distribution needs of the most challenging SCADA and Utility applications. Companies that should seriously consider deploying DDS are (1) companies developing utility automation such as power-grid control, looking for more scalable and high performance data distribution middleware, (2) companies struggling with keeping up with high volume telemetry systems, who are looking for a standard-based solution that can cope with massive volumes of data in real-time to either replace home-grown solutions, or under-performing proprietary solutions, (3) Industrial control companies that want to fully leverage their move to Ethernet or Industrial Ethernet, and are looking for more productive data distribution middleware that can be easily integrated with enterprise systems.

4. References

[1] OMG, "Data Distribution Service for Real-Time Systems Specification," http://www.omg.org/docs/formal/04-12-02.pdf.

[2] A. Corsaro. "The Data Distribution Service for Real-Time Systems", Dr. Dobb's Journal. http://bit.ly/dhQCDb

[3] OMG DDS Forum, www.dds-forum.org.

[4] OpenSplice DDS, www.prismtech.com/openplice-dds.

5. Glossary

COTS: Components Off The Shelf

DDS: Data Distribution Service for Real-Time Systems

EMS: Electronics manufacturing service

GDS: Global Data Space

Pub/Sub: Publish/Subscribe

OMG: Object Management Group

QoS: Quality of Service

SCADA: Supervisory Control and Data Acquisition