



Addressing the Challenges of Tactical Information Management in Net-Centric Systems With DDS

Dr. Douglas C. Schmidt, Dr. Angelo Corsaro, and Hans van't Hag
PrismTech Corporation

Recent trends in net-centric systems motivate the development of tactical information management capabilities that ensure the right information is delivered to the right place at the right time to satisfy quality of service (QoS) requirements in heterogeneous environments. This article presents an architectural overview of the Object Management Group's (OMG) Data Distribution Service (DDS), which is a standards-based QoS-enabled data-centric middleware platform that enables applications to communicate by publishing information they have and subscribing to information they need in a timely manner. DDS is an important distributed software technology for mission-critical Department of Defense (DoD) net-centric systems because it supports the following: (1) location independence, via anonymous publish/subscribe (pub/sub) protocols that enable communication between collocated or remote publishers and subscribers, (2) scalability, by supporting large numbers of topics, data readers, and data writers and platform portability, and (3) interoperability, via standard interfaces and transport protocols.

Tactical information management systems increasingly run in net-centric environments characterized by thousands of platforms, sensors, decision nodes, and computers connected together to exchange information, support sense-making, enable collaborative decision making, and effect changes in the physical environment. For example, the Global Information Grid (GIG) is an ambitious net-centric environment being designed to ensure that different services and coalition partners, as well as individuals participating to specific missions, can collaborate effectively and deliver appropriate firepower, information, or other essential assets to warfighters in a timely, dependable, and secure manner [1]. Achieving this vision requires the following capabilities from the distributed middleware software:

- **Shared operational picture.** A key requirement for mission-critical net-centric systems is the ability to share an operational picture with planners, warfighters, and operators in real-time.
- **Ensure the right data gets to the right place at the right time** by satisfying end-to-end QoS requirements, such as latency, jitter, throughput, dependability, and scalability.
- **Interoperability and portability in heterogeneous environments.** Since net-centric systems are faced with unprecedented challenges in terms of platform and network heterogeneity, they are necessary.
- **Support for dynamic coalitions.** In many net-centric tactical information management systems, dynamically formed coalition of nodes will need to share a common operational picture and exchange data seamlessly.

Prior middleware technologies (such as

the Common Object Request Broker Architecture [CORBA] Event Service and Notification Service, the Java Message Service [JMS], and various other proprietary middleware products) have historically lacked key architectural and QoS capabilities, such as dependability, survivability, scalability, determinism, security, and confidentiality needed by net-centric systems for tactical information management. To address these limitations – and to better support tactical information management in net-centric systems like the GIG – the OMG has adopted the DDS specification, which is a standard for QoS-enabled data-centric pub/sub communication aimed at net-centric tactical information management systems [2]. DDS is used in a wide range of military and commercial systems including naval combat management systems, commercial air traffic control, transportation management, automated stock trading systems, and semiconductor fabrication devices.

The remainder of this article presents an overview of DDS that is geared to software architects. We also discuss the DDS QoS policies that are the most relevant for net-centric tactical information management systems. Finally, we explain how DDS has been applied in practice to address key challenges of developing and operating distributed software in current and planned net-centric tactical information management systems.

Overview of DDS

DDS provides the following capabilities for net-centric tactical information systems:

- **Universal access to information** from a wide variety of sources that run over potentially heterogeneous hardware/software platforms and net-

works.

- **An orchestrated information bus** that aggregates, filters, and prioritizes the delivery of this information to work effectively under the restrictions of transient and enduring resource constraints.
- **Continuous adaptation to changes** in the operating environment, such as dynamic network topologies, publisher and subscriber membership changes, and intermittent connectivity.
- **Standard QoS policies and mechanisms** that enable applications and administrators to customize the way information is delivered, received, and processed in the appropriate form and level of detail to users at multiple levels in net-centric systems.

This section describes the key capabilities and entities in DDS and shows how its QoS policies can be used to specify and enforce performance-related requirements of net-centric tactical information management systems. Figure 1 shows the various profiles and layers in the DDS standard. The lower layer defines a Data-Centric Publish Subscribe (DCPS) platform, whose goal is to provide efficient, scalable, predictable, and resource-aware data distribution. The higher layer is the Data Local Reconstruction Layer (DLRL), which is an optional interface that provides an object-oriented façade atop the DCPS. The DLRL can be used to map topics onto object fields and defines navigable associations between objects.

A separate specification, called the Real-Time Publish/Subscribe (RTPS) DDS interoperability wire protocol, defines the standard network protocol used to exchange data between publishers and subscribers that use different implementations of DDS [3]. The remainder of

this section describes the conceptual model of DDS and explains the QoS policies that are most relevant for net-centric tactical information management systems.

DDS Conceptual Model

Domains and Partitions

DDS applications send and receive data within a domain. Domains can be divided into partitions that allow the separation and protection of different data flows. Although DDS entities can belong to different domains, only participants within the same domain can communicate, which helps isolate and optimize communication within communities that share common interests. For example, each communication layer within the GIG could be associated with a DDS domain and further subdivided into partitions. This approach isolates domain participants across layers, which enables effective use of resources and helps enforce security and confidentiality policies.

Global Data Space

DDS provides a strongly typed global data space within each domain in which applications produce and consume the dynamically changing portions of a shared information model, as shown in Figure 2. DDS' information model capabilities are similar to those of relational databases, except that DDS' global data space is completely distributed, QoS-aware, and allows anonymous and asynchronous sharing of a common information model. The DDS information model is the only knowledge publishers and subscribers need to communicate, i.e., they need not be aware of each other nor be concerned with low-level network programming details, such as Internet protocol addresses, port numbers, remote object references, or service names. By allowing data to flow where and when needed, DDS's global data space enables the sharing of tactical information and situational awareness information needed to implement net-centric tactical information management systems.

Topic

A DDS topic is an association between a data type, a set of QoS, and a unique name, as shown in Figure 3 (see page 26). A topic is also the unit of information contained in DDS' global data space and is used by applications to define their information model and associate QoS policies with it. DDS applications in net-centric systems define their information model by identifying topics that are relevant for solving their requirements and organizing them into either relational or

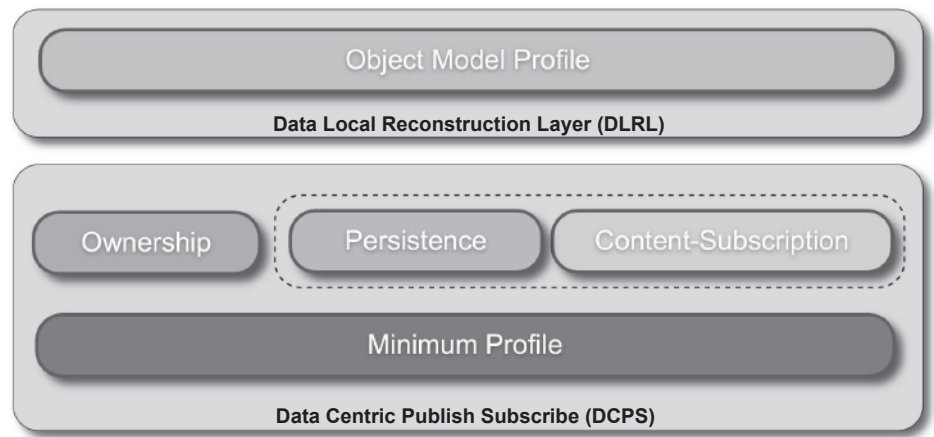


Figure 1: Profiles and Layers in the DDS Standard

object-oriented models. DDS thus allows the expression of the system information model as either a 1) topic relational model, which can be thought of as an extension of the familiar entity relationship diagrams used in data bases, decorated with QoS, or 2) an object-oriented model, which can also be synthesized as an object-oriented view of the relational model.

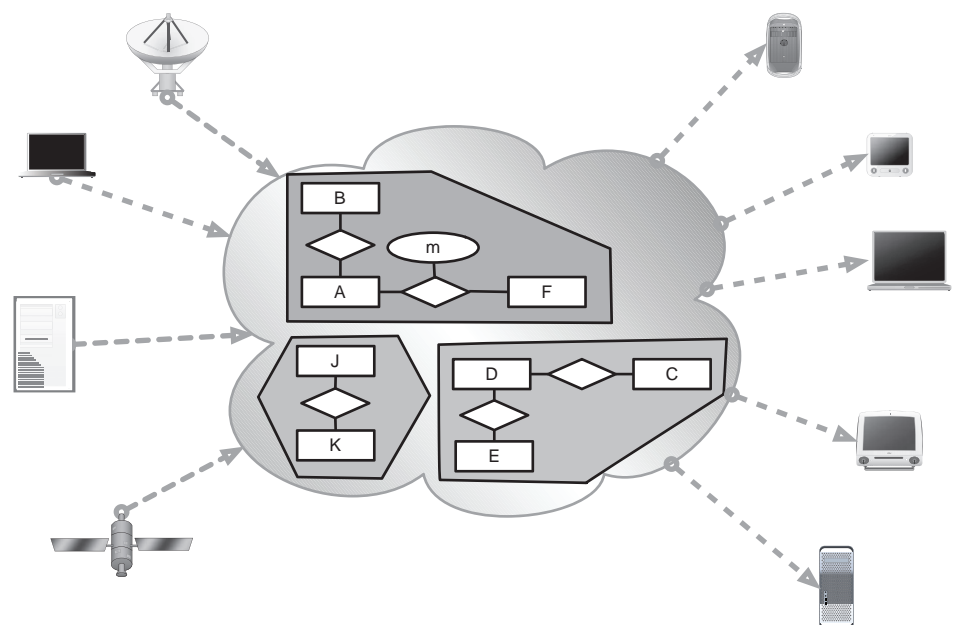
The DCPS layer provides support for relational modeling, while the DLRL extends the DCPS with an object-oriented facade, so that applications can either completely ignore the DCPS relational models or build an object model atop the DLRL. Data associated with DDS topics are expressed using types defined by the standard OMG Interface Definition Language (IDL), which simplifies the inter-working between DDS and CORBA. Relationships between topics can be captured via keys that can be used to distinguish between different instances of the same topic.

In net-centric tactical information systems, an information model will be associated with every layer in which DDS-based data exchange occurs. This information model, which can comply with DoD or North American Trade Organization standards, is the lingua franca used by the different applications in coalitions to exchange information and seamlessly interoperate. Likewise, the QoS policies decorating the information model determine how the data is disseminated, persisted, and received in the global data space.

Publishers and Subscribers

In net-centric tactical information management systems, publishers and subscribers correspond to a range of domain participants such as embedded devices, Unmanned Air Vehicles (UAVs), soldiers' equipment, as well as planning and simulation services in operations centers. DDS applications use data writers to publish

Figure 2: DDS Global Data Space in a Domain



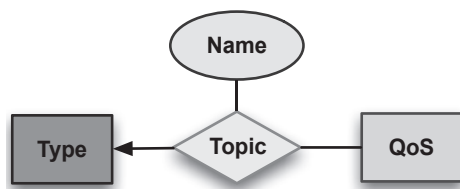


Figure 3: DDS Topic

data values to the global data space of a domain and data readers to receive data. A publisher is a factory that creates and manages a group of data writers with similar behavior or QoS policies, as shown in Figure 4. A subscriber is a factory that creates and manages data readers, as shown in Figure 4.

Publishers can declare their intent to produce data on particular topics with associated QoS, and they distribute the data on those topics to the global data space. Subscribers receive topic data in the global data space that match their subscriptions (the rules that define what represents a matching subscription are described below). QoS policies allow publishers and subscribers to define, first, their local behavior, such as the number of historical data samples they require and the maximum update-rate at which they want to receive data, and, second, how data should be treated once in transit with respect to reliability, urgency, importance, and durability. Topics can also be annotated with these QoS policies to drive the behavior of the data-distribution. The QoS policies of pre-defined topics serve as defaults for publishers and subscribers and can therefore ensure consistency between requested and offered QoS.

Subscriptions and Matching

A subscription is an operation that associates a subscriber to its matching publishers, as shown in the center of Figure 4. In addition to the topic-based subscriptions

described, DDS also supports *content-based subscription*, in which a subset of the standard Structured Query Language (SQL) is used to specify subscription filters. In DDS a *matching* subscription must match the following two types of a topic's properties: (1) its features, such as its type, name, key, and content; (2) its QoS policies, which are described in the *QoS Policies* section.

The matching process for QoS uses a requested/offered (RxO) model shown in Table 1, where the requested QoS must be less than or equal to the offered QoS. For example, subscribers requesting reliable data delivery cannot communicate with publishers that only distribute data using best effort delivery. Likewise, subscribers cannot request a topic update whose deadline is smaller than that declared by any publishers.

The subscription matching mechanism provided by DDS enforces a powerful form of *design by contract* [4], where QoS is used together with type information to decide whether publishers and subscribers can communicate. This extended form of design by contract helps ensure that net-centric systems will operate as intended, both from functional and QoS perspectives. These assurances are essential in the development, deployment, and operation of mission-critical net-centric tactical information management systems.

Discovery

Another key feature of DDS is that all information needed to establish communication can be discovered automatically, in a completely distributed manner. Applications dynamically declare their intent to become publishers and/or subscribers of one or more topics to the DDS middleware, which uses this information to establish the proper communication paths between discovered entities. This

capability supports dynamic scenarios common in net-centric tactical information management where cooperating domain participants join and leave throughout system operation.

QoS Policies

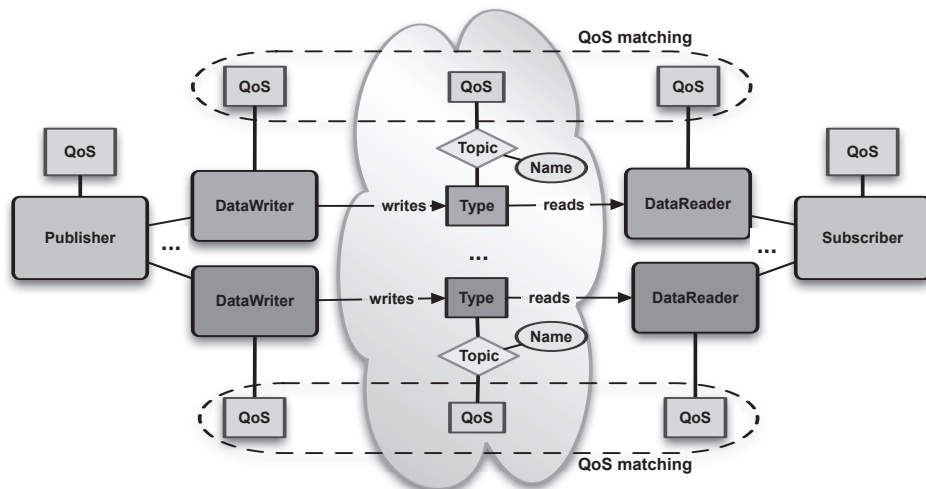
DDS is designed for mission-critical net-centric systems where the right answer delivered too late becomes the wrong answer. To meet timing requirements it is essential that the middleware controls and optimizes the use of resources, such as network bandwidth, memory, and CPU time. Table 1 shows the rich set of QoS policies that DDS provides to control and limit topic (T), data reader (DR), data writer (DW), publisher (P), and subscriber (S) resources and topic QoS properties, such as persistence, reliability, and timeliness [2]. Below we discuss the DDS QoS policies that are the most relevant for net-centric tactical information management systems.

Data Availability

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

- The Durability QoS policy controls the lifetime of the data written to the global data space in a domain. Supported durability levels include the following: (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. Durability is achieved by relying on a durability service whose properties are configured by means of the DURABILITY_SERVICE QoS.
- 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

Figure 4: DDS Publisher/Writer Subscriber/Reader and Subscription/QoS Matching



last n samples, or all samples.

These QoS policies provide the DDS global data space with the ability to cooperate in highly dynamic environments characterized by continuous joining and leaving of publisher/subscribers. This capability makes it possible for net-centric tactical information management systems to share a common operational picture regardless of the dynamism that characterizes portions of the systems, such as coalitions of soldiers collaborating in urban environments or coordinated UAVs in support of tactical operations.

Data Delivery

DDS provides the following QoS policies that control how data is delivered and which publishers are allowed to write a specific topic:

- The PRESENTATION QoS policy gives control on how changes to the information model are presented to subscribers. This QoS gives control on the ordering as well as the coherency of data updates. The scope at which it is applied is defined by the access scope, which can be one of INSTANCE, TOPIC, or GROUP level.
- The RELIABILITY QoS policy controls the level of reliability associated with data diffusion. Possible choices are RELIABLE and BEST_EFFORT distribution.
- The PARTITION QoS policy gives control over the association between DDS partitions (represented by a string name) and a specific instance of a publisher/subscriber.
- The DESTINATION_ORDER QoS policy controls the order of changes made by publishers to some instance of a given topic. DDS allows the ordering of different changes according to source or destination timestamps.
- The OWNERSHIP QoS policy controls which writer owns the write-access to a topic when there are multiple writers and ownership is EXCLUSIVE. Only the writer with the highest OWNERSHIP_STRENGTH can publish the data. If the OWNERSHIP QoS policy value is shared, multiple writers can concurrently update a topic. OWNERSHIP thus helps to manage replicated publishers of the same data.

These QoS policies control the reliability and availability of the data, thus allowing the delivery of the right data to the right place at the right time. More elaborate ways of selecting the right data are

QoS Policy	Applicability	RxO	Modifiable	
DURABILITY	T, DR, DW	Y	N	Data Availability
DURABILITY SERVICE	T, DW	N	N	
LIFESPAN	T, DW	-	Y	
HISTORY	T, DR, DW	N	N	
PRESENTATION	P, S	Y	N	Data Delivery
RELIABILITY	T, DR, DW	Y	N	
PARTITION	P, S	N	Y	
DESTINATION ORDER	T, DR, DW	Y	N	
OWNERSHIP	T, DR, DW	Y	N	
OWNERSHIP STRENGTH	DW	-	Y	Data Timeliness
DEADLINE	T, DR, DW	Y	Y	
LATENCY BUDGET	T, DR, DW	Y	Y	
TRANSPORT PRIORITY	T, DW	-	Y	
TIME BASED FILTER	DR	-	Y	Resources
RESOURCE LIMITS	T, DR, DW	N	N	
USER_DATA	DP, DR, DW	N	Y	Configuration
TOPIC_DATA	T	N	Y	
GROUP_DATA	P, S	N	Y	

Table 1: Key QoS Policies for Net-Centric Systems

offered by the DDS content-awareness profile that allows applications to select information of interest based upon their content.

Data Timeliness

DDS provides the following QoS policies that control the timeliness properties of distributed 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.
- 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(s).
- 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 prioritize more important data relative to less important data.

These QoS policies make it possible to ensure that tactical information needed to reconstruct the shared operational picture is delivered in a timely manner.

Resources

DDS defines the following QoS policies to control the network and computing resources that are essential to meet data dissemination requirements:

- The TIME_BASED_FILTER QoS policy allows applications to specify the minimum inter-arrival time between data samples, thereby expressing their capability to consume information at a maximum rate. Samples that are produced at a faster pace are not delivered. This policy helps a DDS implementation optimize network bandwidth, memory, and processing power for subscribers that are connected over limited bandwidth networks or which have limited computing capabilities.
- The RESOURCE_LIMITS QoS poli-

cy allows applications to control the amount of message buffering performed by a DDS implementation.

DDS's QoS policies support the various elements and operating scenarios that constitute net-centric tactical information management. By controlling these QoS policies it is possible to scale DDS from low-end embedded systems connected with narrow and noisy radio links, to high-end servers connected to high-speed fiber-optic networks.

Configuration

The QoS policies described above provide control over the most important aspects of data delivery, availability, timeliness, and resource usage. In addition, DDS also supports the definition and distribution of user specified bootstrapping information via the following QoS policies:

- The **USER_DATA** QoS policy allows applications to associate a sequence of octets to domain participant data readers and data writers. This data is then distributed by means of the DCPS participant built-in topic. This QoS policy is commonly used to distribute security credentials.
- The **TOPIC_DATA** QoS policy allows applications to associate a sequence of octets with a topic. This bootstrapping information is distributed by means of the DCPS Topic built-in topic. A common use of this QoS policy is to extend topics with additional information, or meta-information, such as eXtensible Markup Language schemas.
- The **GROUP_DATA** QoS policy allows applications to associate a sequence of octets with publishers and subscribers. This bootstrapping information is distributed by means of the DCPS subscription, and DCPS publication built-in topics, respectively. A typical use of this information is to allow additional application control over subscriptions matching.

DDS Success Stories

Although DDS is a relatively new standard (adopted by the OMG in 2004), it has been adopted quickly due to its ability to address key requirements of data distribution in net-centric systems, as well as the maturity and quality of available implementations, which are based on decades of experience developing data-centric middleware for mission-critical systems. Moreover, DDS has been mandated by the U.S. Navy's Open Architecture Computing Environment as the standard publish/subscribe technology to use in

next-generation combat management systems, and Defense Information Systems Agency as the standard technology for publish/subscribe to be used in all new or upgraded systems [5, 6]. Several major defense programs, such as the U.S. Navy's DDG-1000 land attack destroyer, U.S. Army's Future Combat Systems (FCS), and the Thales Tactical Information And Command System Operating System (TACTICOS), also adopted DDS even before it was mandated, underscoring DDS' ability to address the data distribution challenges of next generation net-centric defense systems.

For example, the TACTICOS combat management system developed by Thales Naval Netherlands is based on an implementation of DDS that allows them to achieve very good scalability, from small

***DDS provides the
fault-tolerant information
backbone onto which
all these applications are
deployed and is thus
responsible for providing
each application with
the right information
at the right time.***

ships to aircraft carrier grade, as well as high performance, availability, and determinism even under temporary overload conditions [7, 8]. TACTICOS is currently in use in 15 navies worldwide serving 20 ships-classes ranging from small patrol boats up to large frigates. The utilization of DDS is instrumental in its success since it provides both the scalability to support thousands of applications running on more than 150 distributed computers on a frigate size system. Another key feature of DDS is its battle-damage resistance, meaning that software can be dynamically re-allocated to the remaining computer pool in case of an error on a specific computer. The DDS Persistence Profile support is instrumental in this dynamic reallocation since it allows applications to store their internal state into the DDS middleware, which manages this state in a distributed and fault-tolerant way so that restarted applications can continue what they were doing before they crashed.

The DDS implementation used on TACTICOS supports a data-centric approach where at the start of the system design, the information model can be captured, annotated with proper QoS policies, and then shared between multiple parties. This federated architecture is common in existing and planned *coalition-based* developments where multiple parties jointly implement the overall combat system. DDS provides the fault-tolerant information backbone onto which all these applications are deployed and is thus responsible for providing each application with the right information at the right time.

Along with the rapid adoption of DDS in the defense domain, its use is also steadily growing in other domains, such as transportation, telecommunications, and finance. For example, in the context of Air Traffic Control and Management, DDS has been selected as the publish/subscribe middleware for distributing flight data plans in CoFlight [9], which is the next generation European Flight Data Processor. In general, DDS is an appropriate middleware technology for application domains that require rich support for QoS policies and high-performance and dependability standards-based, commercial-off-the-shelf implementations.

Concluding Remarks

DDS is a standards-based QoS-enabled data-centric publish/subscribe middleware that provides a feature rich data-centric real-time platform to support the needs of current and planned net-centric tactical information management systems. Its powerful set of QoS policies, together with its scalable architecture, makes it an effective and mature choice for solving the data distribution and information management problems net-centric systems [10]. Next, we summarize how DDS addresses the key challenges outlined in the introduction in a standard and interoperable manner:

- **Shared operational picture.** DDS provides effective support for these types of applications via its QoS policies for defining the scope, content, and QoS of the data model that underlies the operational picture.
- **The right data at the right time at the right place** via DDS QoS policies that enable a fine-grained control over information delivery, such as the ability to control many aspects of data dissemination to ensure timely delivery and optimal resource usage.
- **Heterogeneous environment.** By

providing standard QoS policies that control the bandwidth used for providing data to interested parties, DDS runs in heterogeneous platforms while providing different elements with a common operational picture.

- **Dynamic coalitions.** The highly dynamic nature of DDS, such as its support for dynamic discovery, provides an effective platform for supporting ad hoc interactions.

DDS continues to evolve to meet new operational and technical challenges of net-centric tactical information management systems. Three types of extensions are currently being pursued for DDS by the OMG. The first involves adding new platform-specific models that fully leverage programming language features, such as standard C++ containers. The second extension deals with extensible topics that enable incremental system updates by ensuring that changes in the data model do not break interoperability. The final set of extensions focus on network data representation and the syntax used to define topics. For example, upcoming versions of the DDS standard will likely allow the definition of topics using XML, as well as the use of XML or Java Script Object Notation as the network data representation. DDS security has not yet been stan-

dardized. The OMG will be addressing this area of standardization starting in the spring of 2008.

With multiple COTS and open-source implementations and a solid track record of success in mission-critical military and commercial projects, DDS has a bright future as the standards-based middleware of choice for net-centric tactical information systems. More information on DDS and its application in practice are available in online forums [11, 12] where experts discuss advanced features of the DDS standard and new directions for the technology, while DDS beginners can learn from past experiences and ask questions about patterns and best practices for applying DDS in their net-centric systems. ♦

References

1. "The United States Department of Defense Quadrennial Defense Review Report." Feb. 2006 <www.defense.gov/link.mil/qdr/report/Report20060203.pdf>.
2. OMG. "Data Distribution Service for Real-Time Systems Specification." <www.omg.org/docs/formal/04-12-02.pdf>.
3. OMG. "Real-Time Publish Subscribe Protocol – DDS Interoperability Wire Protocol Specification." <www.omg.org/cgi-bin/apps/doc?ptc/06-08-02.pdf>.
4. Bertrand, Meyer. Object Oriented Software Construction. 2nd ed., Prentice Hall, 2001.
5. "Open Architecture Computing Environment." <www.nswc.navy.mil/wwwDL/B/OACE>.
6. Defense Systems Information Agency. "DoD Information Technology Standards Registry." <<https://disron.line.disa.mil>>.
7. THALES. "TACTICOS Combat Management System, Exploiting the Full DDS Potential." <www.omg.org/docs/dds/06-12-06.pdf>.
8. "OpenSplice DDS." <www.prismtech.com/opensplice-dds>.
9. CoFlight eFDP <www.omg.org/docs/dds/07-07-04.pdf>.
10. Xiong, Ming, et al. "Evaluating Technologies for Tactical Information Management in Net-Centric Systems." Proceedings of the Defense Transformation and Net-Centric Systems Conference, Apr. 9-13, 2007, Orlando, FL.
11. OMG DDS SIG Portal <portals.omg.org/dds>.
12. OMG DDS Forum <www.dds-forum.org>.

About the Authors



Douglas C. Schmidt, Ph.D., is a professor of computer science at Vanderbilt University and is the chief technical officer of PrismTech.

His expertise focuses on distributed computing middleware, object-oriented patterns and frameworks, and distributed real-time and embedded systems. He has authored nine books and more than 350 papers in top technical journals, conferences, and books that cover high-performance communication software systems, real-time distributed computing, and object-oriented patterns for concurrent and distributed systems.

PrismTech Corporation
6 Lincoln Knoll LN
STE 100
Burlington, MA 01803
Phone: (781) 270-1177
Fax: (781) 238-1700
E-mail: doug.schmidt@prismtech.com



Angelo Corsaro, Ph.D., is the OpenSplice DDS product marketing manager at PrismTech and co-chairs the OMG DDS Special Interest Group and the Real-Time Embedded and Specialized Services task force. He is well-known in the distributed real-time and embedded systems middleware community and has a wealth of experience in hard real-time embedded systems, large-scale, and very large-scale distributed systems such as defense, aerospace, homeland security and transportation systems.

PrismTech Corporation
6 Lincoln Knoll LN
STE 100
Burlington, MA 01803
Phone: (781) 270-1177
Fax: (781) 238-1700
E-mail: angelo.corsaro@prismtech.com



Hans Van't Hag is the OpenSplice DDS product manager, at PrismTech. He has extensive experience in applying an information approach towards mission-critical and real-time net-centric systems. Hag is a contributor to the OMG DDS specification and has presented numerous papers on DDS and publish/subscribe middleware technologies. Prior to joining PrismTech, he worked at Thales Naval Netherlands (TNN) where he was responsible for the development of the data-centric real-time middleware as applied in TNN's combat system in service with 15 navies worldwide.

PrismTech Corporation
6 Lincoln Knoll LN STE 100
Burlington, MA 01803
Phone: (781) 270-1177
Fax: (781) 238-1700
E-mail: hans.vanthag@prismtech.com