WHITEPAPER

How to Achieve Production-Grade Deployment with ROS 2 and RTI Connext™ DDS

USING OPEN STANDARD-BASED TOOLS TO BUILD RELIABLE, SCALABLE SYSTEMS

LEAD AUTHOR

Neil Puthuff
Software Integration Engineer, RTI

INTRODUCTION

This technical insights paper is for developers of autonomous and semi-autonomous systems based on ROS (Robot Operating System) who need to bring these systems to a deployable production-grade state, including those with safety certification requirements. While the original ROS enables robotics systems to be rapidly created, it was not designed for the rigors of production deployment or safety certification.

Enter ROS 2: an upgrade to expand ROS into new use cases while retaining a fundamentally ROS-like environment. ROS 2 is based on the open-standard Data Distribution Service™ (DDS) framework, already in use in thousands of critical applications worldwide. While ROS 2 is based on DDS, its implementation uses only a subset of the full range of capabilities of DDS for demanding real-world applications.

However, *ROS 2 is a DDS application*. This means that ROS 2 components can be freely intermixed with native DDS components, with full interoperability. This hybrid system design approach brings capabilities far beyond what ROS or ROS 2 alone can offer:

- Critical system components (requiring peak performance, safety certification or granular security) can be implemented directly in DDS.
- Components needing the full range of DDS capabilities that are unavailable in ROS 2 can also be implemented directly in DDS.
- Pre-existing production-grade components written in ROS 2 can be used as-is.
- All other components can be implemented in ROS 2 or DDS.
- The tools ecosystems of ROS 2 and DDS are available for all components.

This guide will introduce the capabilities of DDS and explore the potential of a hybrid system for achieving production-grade deployment.

BACKGROUND

The ROS framework simplifies the creation of robotics research applications; it includes a rich ecosystem of visualization tools and functional packages, and has support for many types of robotics components. Within a matter of hours, a supported mobile platform can be made to sense, map and navigate a controlled area.

However, ROS was designed for robotics research: It was meant to be run on workstation-class PCs with ample memory, always-available networking and human intervention as needed. ROS was not designed for real-time performance and determinism requirements, nor the ironclad safety and security needed for production release. Further, ROS itself was straining to meet the latest challenges at the forefront of robotics research: multi-robot systems, inconsistent networks, operation on constrained platforms, etc.

An improved ROS was needed to meet these new use cases, and the original ROS would require significant architectural changes. Therefore, the creators of ROS took a clean-sheet approach to a next-gen ROS design, resulting in a new framework called "ROS 2."

ROS 2 was designed from the start to use the best available technology for the system interconnectivity, while maintaining alignment with the spirit and ethos of the ROS community. In doing so, they could spend less time on the 'plumbing' and more time on the applications. After a thorough evaluation of the most-used networking solutions, the designers of ROS 2 chose the DDS (Data Distribution Service) framework as the foundation for creating ROS 2. The ROS 2 framework would be built as a layer on top of the DDS framework.

ROS 2 AND DDS: INTEROPERABLE BY DESIGN

DDS is one of the few standards to use a published set of open specifications that range from the software API and language bindings, down to the security implementation and wire-level protocol. These specifications can be freely downloaded from the omg.org website, and have resulted in more than a dozen implementations. Plug-fests are held regularly to ensure interoperability amongst the different implementations.

Because ROS 2 is implemented on DDS, all ROS 2 applications are technically also DDS applications and benefit from the standards-based interoperability of DDS. This means that developers can freely intermix native DDS applications with

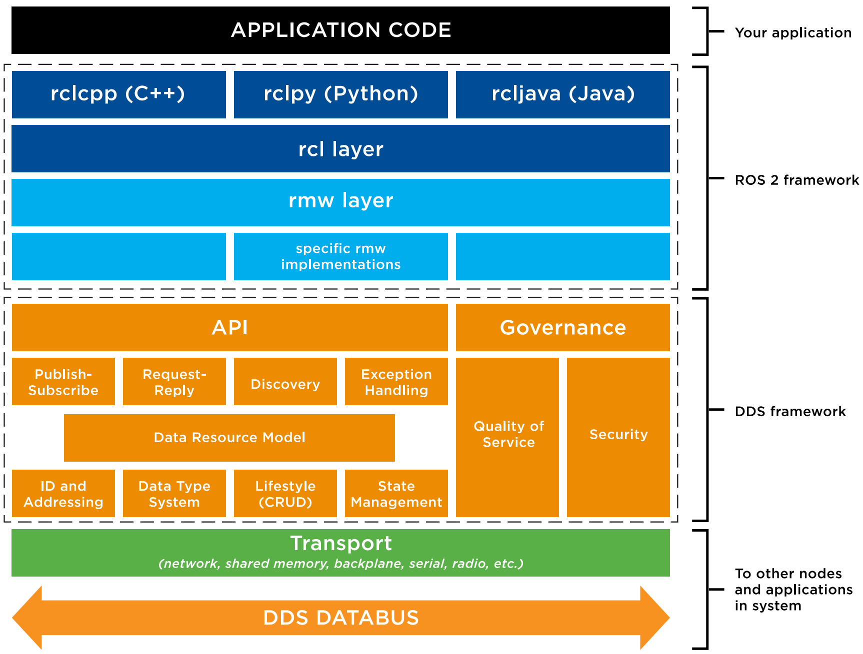

Figure 1: Illustration of DDS/ROS 2 technology stack

ROS 2 applications and use the ecosystems (tools, modules, services) of both ROS 2 and DDS to create more powerful autonomous systems (see Figure 2 below).

Interoperability also depends on using common message types in the system. A hybrid system using both ROS 2 and native DDS applications will seamlessly interoperate when using ROS 2-compatible messages in the native DDS applications. In fact, a native DDS application can be indistinguishable from a ROS 2 application when using an identical set of message types. This message compatibility is also the enabler for interoperation with the ROS 2 tools ecosystem.

Full access to the ROS 2 ecosystem of tools and packages combined with full access to the DDS ecosystem of tools and services provides system creators with many immediate benefits:

- A faster migration and integration path from research to deployment.
- Ability to quickly stand-up a functional robotic or autonomous system.
- Options to use applications available in the open-source community.
- Familiar API and design paradigm for legacy ROS users.
- Continued connection to ROS research groups and academia.
- Ability to deploy safety certifiable versions of DDS.
- Access to professional support, training and architectural guidance.

The DDS ecosystem includes tools to diagnose, analyze and tune your system; services to route and record/replay data and connect your system to the web or a database; and integrations with MATLAB, LabVIEW, AUTOSAR, UML modeling tools and other programming and testing tools and platforms.

The kinship and standards-based interoperability of the combined ROS 2 and DDS ecosystems and applications enables important new options for creating mission-critical systems. Functions may be executed within the ROS 2

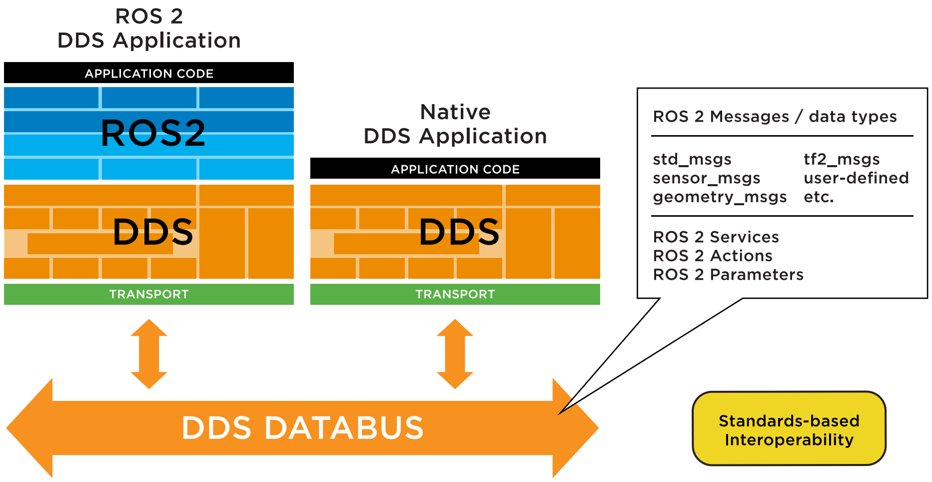

Figure 2: Native DDS applications can be freely shared with ROS 2 applications

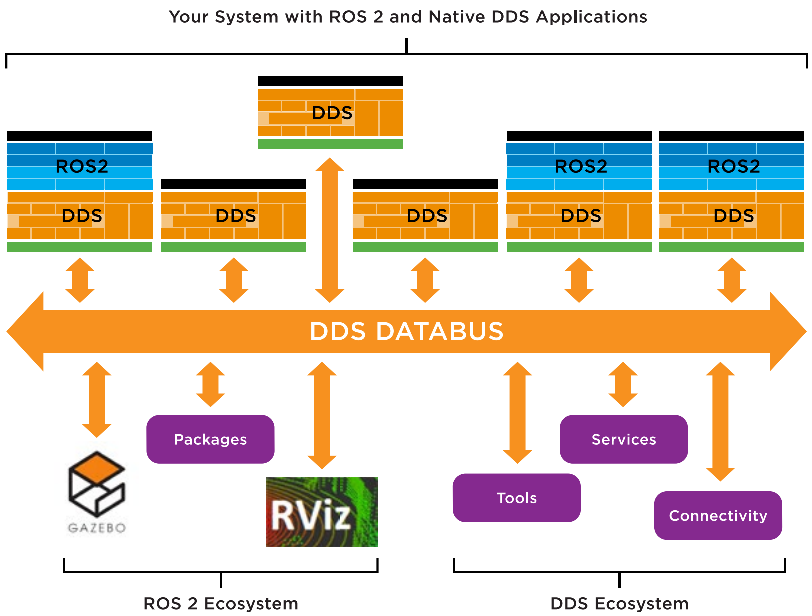

Figure 3: DDS and ROS 2 hybrid system and ecosystem

framework, or directly within the underlying DDS framework (without ROS 2) to meet the functional, performance and safety requirements of the system. Let's explore these options. But first, a brief introduction to DDS.

DDS: STANDARD-BASED DATA CONNECTIVITY FOR PRODUCTION-GRADE ENVIRONMENTS

DDS is an open-standard connectivity framework that offers a familiar environment to ROS developers: Pub/sub, data topics, automated discovery and independent modules, plus an ecosystem of layered services and tools.

Better still, DDS is field-proven: it has been deployed in thousands of the world's most demanding applications, including over 200 autonomous vehicle projects for commercial, research and military use. DDS connects the world's largest SCADA system at the Kennedy Space Center Launch Complex, systems for the US Navy Surface Fleet and the largest power plant in North America (Grand Coulee Dam).

What is unique about DDS? DDS is:

- Interoperable by Design: The DDS standard defines the software API, the wire-level protocol and the security architecture to ensure interoperability and portability across different DDS implementations.
- Secure by Design: DDS secures the data itself on a per-item/per-connection level of granularity. This brings security to unsecure transports (such as UDP, serial, radio link, etc.) and layered security when used with secure transports.
- Reliable by Design: DDS is completely decentralized, implements its own robust reliability mechanisms and includes support for redundant components and transports.
- Safety-Certifiable: DDS has been certified to DO-178C DAL A, the highest level of assurance for software-based aerospace systems. Certification evidence is available.

Key characteristics of DDS include:

- Wide-Ranging QoS: DDS is capable and proven to handle the endless and unforgiving variety of real-world device networking scenarios. This ensures peak performance, reliability, liveliness, filtering, persistence, etc. under the most extreme of conditions.
- Multi-Platform: DDS is supported on over 100 distinct platforms, from clustered servers down to small embedded processors.
- Multi-Language: DDS is available for C, C++, Java, Python, JavaScript, Ada and other popular programming languages.
- Open Standard, Multi-Source: DDS is an Object Management Group® (OMG®)-managed open standard with more than a dozen open source and commercially supported implementations. The complete specs are freely downloadable.
- Peak Performance: DDS uses a compact binary format, data compression, multicast and many other methods to improve system performance and reduce latency.
- Transport-Independent: The features and benefits of DDS are available across all supported transports: IP Ethernet, Backplane, Shared Memory, Radio and Serial. The transport is isolated from the application code.
- Tiered Services: DDS has a rich ecosystem of components for web connectivity, database access, large-scale architectural tiering, record/playback and connectivity to other protocols and transports.
- Familiar Environment: The DDS framework shares many similarities with ROS, making it easier for ROS developers to transition to DDS.
- Works Within Your Build System: DDS is available in source or library form and has code generation utilities that are easily integrated into different build systems.

- Commercially Supported: Paid professional services and architectural guidance are readily available for complex, mission-critical projects.
- Adoption by Industry Standards Organizations: DDS has received multi-industry adoption in standards for automotive (AUTOSAR Adaptive), avionics (FACE™), grid modernization (OpenFMB), medical (OpenICE, MD PnP) and has been named a "Core Connectivity Standard" by the Industrial Internet Consortium (IIC).

With this impressive list of features and credentials, it's easy to see why DDS was selected as the foundational framework for ROS 2, the next-gen version of ROS.

As ROS 2 is a next-gen upgrade to ROS, it seeks to replicate a ROS-like environment on top of DDS. Some important features of DDS may be unavailable from within ROS 2, but are now possible when implemented directly in DDS or in a hybrid DDS/ROS 2 system:

Performance
- Precise tuning to maximize throughput and minimize latency.
- Adaptive flow control to avoid network saturation.
- Transparently batching groups of small samples or splitting very large samples.
- Zero-copy transfers, data compression, multicast, etc.

Scalability
- Keyed topics to uniquely identify countless numbers of endpoints.
- Tiered hierarchy to create a system-of-systems.
- Logically- and security-partitioned networks.

Safety
- Safety-certified versions of DDS.
- Minimal code size to reduce testing burden.
- Built-in support for redundant nodes and networks, with auto-failover.

Security
- Securing insecure networks - without rip-and-replace.
- Granular encryption, authentication, access control.
- Tamper and event detection, distributed logging.

Reliability
- Fine-grained control over data reliability, persistence, history, lifespan.
- Assured delivery over unreliable transports.
- Persistent and durable data across reboots and power cycles.

Platform Support
- Production-grade OS/RTOS: from enterprise-grade down to bare-metal embedded.
- Production-grade hardware: ruggedized, constrained, deeply embedded, etc.
- Difficult networks: intermittent, low-bandwidth, unsecured, etc.
- Wide-ranging transports: IP Ethernet, Serial, Radio, Backplane, Shared Memory, etc.

A BRIEF TOUR OF DDS AND RTI CONNEXT

DDS shares many similarities to ROS, providing a familiar environment and easy start-up for experienced ROS developers. But while ROS was building an ecosystem for robotics research, DDS built an ecosystem around mission-critical systems of all types, with a rich set of diagnostic tools and layered services, field-tested in thousands of applications. The following section highlights some of the key features of RTI Connext DDS, and how it can support developing and deploying production grade systems:

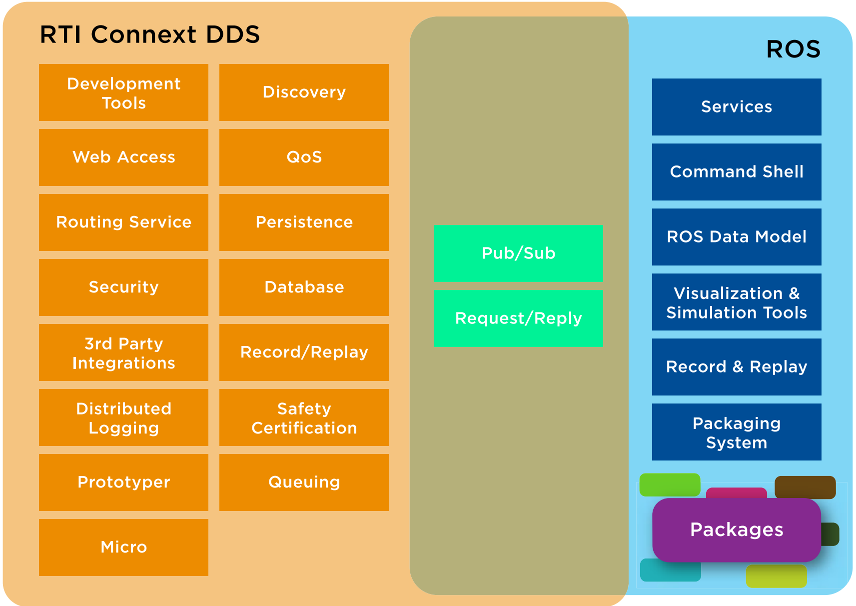

Figure 4: Intersection of RTI Connext DDS and ROS functionality

Development Tools

System developers need high-quality tools to help diagnose, analyze and visualize the system under development. RTI Connext offers a complete suite of tools for diagnostics, tuning and visualization, featuring:

- RTI Admin Console, which is the go-to resource for instantly diagnosing connectivity issues. Admin Console uses DDS discovery information to highlight any mismatches in data type or QoS, and can even subscribe to and visualize data from the topic of your choice.
- RTI Monitor, which gives developers a detailed look inside of DDS to help tune and optimize for maximum performance.
- Command-line Utilities for performance benchmarking, topic data examination, type conversion and code generation. These are standalone utilities, separate from the deployable DDS libraries.
- Recording & Replay Service, which will continuously record your DDS data traffic in real-time, enabling anytime replay of collected data.
- 3rd-party Integrations with popular simulation, modeling, diagnostic and control toolsets.

RTI Connext DDS tools provide development and troubleshooting capabilities throughout the entire development lifecycle, from: design to production; system monitoring and administration; and functional, systems and performance testing for development and quality assurance.

In addition, DDS has the flexibility to work directly with the tools in the ROS 2 ecosystem, so developers using DDS can work with the best of both ROS 2 and DDS tools.

System Scalability

Large-scale systems tend to adopt a layered architecture instead of a single large dataspace. This helps to compartmentalize local data where it is needed, while selectively routing other data hierarchically elsewhere in the system. In addition, the components of these systems use efficient identifiers on common data types, which allow systems to scale up to immense proportions.

DDS encourages the use of Keyed data topics. Similar to a database, Keys enable data items to be efficiently and uniquely identified, even in very large systems. RTI Connext also has an available gateway element (“Routing Service”) to help create a layered architectural pattern. This gateway can route, bridge, isolate and optionally translate selected data items, and even bridge between different levels of security.

Performance

DDS is designed for high performance. It sends only the data values (instead of the key:value pairs), uses CDR (a compact binary data representation) for encoding and has a broad range of QoS settings to improve throughput and reduce latency.

RTI Connext offers further enhancements, such as ZeroCopy (wherein large data is sent by reference instead of copying the entire data object), FlatData (this eliminates some processing steps to improve speed) and data compression to reduce network bandwidth usage.

Security

A layered security architecture offers increased resistance to attack. It’s analogous to having a room of locked boxes, inside a bank vault, located inside a locked and alarmed building. Layered security is much more difficult to compromise than a single-layer solution, such as Transport Layer Security (TLS). Security should also be flexible to quickly adapt to any new exploits that may be discovered and used in the field.

DDS takes a fine-grained approach to security by encrypting, authenticating and access-controlling individual data items with designated readers and writers. This enables a free mixture of secured and plaintext data according to the system’s needs, and is implemented in DDS using a plug-in architecture that allows rapid updates to the security algorithms.

By securing the data only, DDS security works equally well with secured transports (such as TLS) for effective security layering and unsecured transports (such as UDP, serial, backplane, etc.) to protect critical systems and sensitive data. DDS security is based on the OMG DDS Security Specification standard.

Safety Certification

Many production-track systems will require some form of software safety certification. Whether it’s DO-178C or an ISO certification standard, every line of code will add to the cost of certification and any non-deterministic software behavior will be problematic for safety.

RTI Connext DDS is available with a DO-178C level A certification package with supporting evidence, which meets the requirements of other (ISO) certification standards. This is a reduced-to-essentials DDS implementation with a small code footprint, easing the test burden.

Reliability

How can a sender of data know that the data was received when it is sent across a network, a backplane, a serial connection or even across shared memory? The TCP/IP network transport includes its own reliability mechanism, but UDP and most other transports do not. When creating a system that might not use TCP/IP, how can data be sent reliably?

DDS has its own integrated reliability mechanism. Rather than relying on the underlying transport (such as TCP/IP) to implement reliability, DDS enables unreliable transports (such as UDP, serial, backplane, etc.) to deliver data reliably by using a tunable range of controls over timing, buffer sizes and event notifications. Connext DDS includes advanced options for reliability, including ACK or NACK notification, operation over UDP multicast, application-level reliability and the option of using pre-made QoS ‘recipes’ for common scenarios, such as streaming data, alarms, etc.

ROS 2 simplifies the DDS reliability settings by offering 2 choices: “Best Effort” or “Reliable.”

Advanced Capabilities

DDS includes a spectrum of features to enable critical systems to run smoothly:

- Data Type Extensibility is a means by which a data type can be extended by adding additional data elements, while still maintaining forward and backward compatibility. This means that systems can be upgraded without breaking compatibility with earlier and non-upgraded portions of the system.

- Persistence and Durability provides the means for late-joining components to be immediately updated with the most-recent data, even if the system was rebooted or had a temporary power outage.
- Liveliness will let the system know that a node is still connected and active, even if it's not actively sending data. This is important for low-occurrence events, such as alarms.
- Lifespan of data samples can be specified in DDS, wherein data will be automatically discarded if it persists for more than a preset period of time.
- Filtering of data (by time or value) is available in DDS, enabling sample-rate reductions and suppressing the sending of data when that data's value falls outside of a specified range.

Platform Support

A key question for a production-grade software solution is, "Does it support my platform?" Will the software support the specific CPU, peripherals, software toolchain and operating system planned for production release of your system, or will you have to port and maintain this software yourself?

Connext DDS is available on more than 100 different platforms including commercial, industrial and military-grade hardware and conventional and real-time (RTOS) operating systems, with support for a wide variety of software toolchains and programming languages. Professional support and maintenance for new platforms is also available.

Technical and Architectural Support

Distributed and Autonomous systems development can be very complex. Where can a production-bound system developer go for technical support and architectural guidance?

Similar to ROS, DDS has community/open-source support for resolving issues. RTI Connext is available with paid technical support, hands-on training and personalized architectural studies and guidance from the experts involved in thousands of successfully-deployed projects across industries and applications.

APPLICATION CHALLENGES WHICH REQUIRE COMMERCIALLY-SUPPORTED DDS

Production-grade systems are often tasked to operate safely, securely and reliably in environments that are hostile to lab-grade software frameworks. Consider the following scenarios that must be accommodated in real-world deployments:

Performance and Scalability
- I need maximum performance/minimum latency in my system.
- I need to manage the dataflow to avoid saturating the network.
- My system is to be deployed at scale - from single units to millions.

Safety and Security
- My system must be safety certified.
- I need to send data securely over unsecured networks.
- Only some of my data needs to be encrypted, authenticated or access-controlled.
- I need to harden my system against unauthorized access and reverse-engineering.

Reliability and Redundancy
- I need to set up redundant nodes and redundant transports with auto-failover.
- Parameters must be immediately available, regardless of reboots or power-cycles.
- I need to set a limit on data-transfer latency, and know when it has been exceeded.

Platform Support
- My system must run on a production-grade OS/RTOS and constrained hardware.
- I have a non-ideal network: intermittent, low-bandwidth, unsecured, etc., on a transport using IP Ethernet, Serial, Radio, Backplane, Shared Memory, etc.
- I need to cleanly interface my system with different protocols, such as CAN, ModBus, AUTOSAR, RS-232, etc.
- I need guidance. Help!

These scenarios require a robust and well-designed DDS implementation such as RTI Connext DDS, working independently, cooperatively or as the foundation of ROS 2.

CREATING A SCALABLE, PRODUCTION-GRADE SYSTEM

Determining which parts of your system should be implemented directly in DDS vs. implementation in ROS 2 will depend on the goals for your system:

- Use ROS 2 for any pre-written ROS 2 packages and nodes that meet your functional, performance and safety needs.
- Implement everything else directly in DDS to enjoy the benefits of peak performance, scalability, safety certification and the vast range of capabilities inherent to DDS that are not implemented in ROS 2.

Note that all implementations above can benefit from the merged ROS 2 and DDS ecosystems: the visualization tools and packages of ROS 2 can work side-by-side with the diagnostic tools and layered services of DDS.

EXAMPLE SCENARIOS

A. I need maximum data performance from my sensor suite.

An autonomous system has a set of sensors that require high data throughput at low latency. Performance expectations are not being met; is there a way to fix this?

Yes - changing the system to implement the sensor data-handling nodes directly in DDS results in an optimized data flow with lower latency and higher throughput:

The performance is improved by several means:

- Shorter data path: the processing time spent in the ROS 2 layer has been completely eliminated (this is substantial).

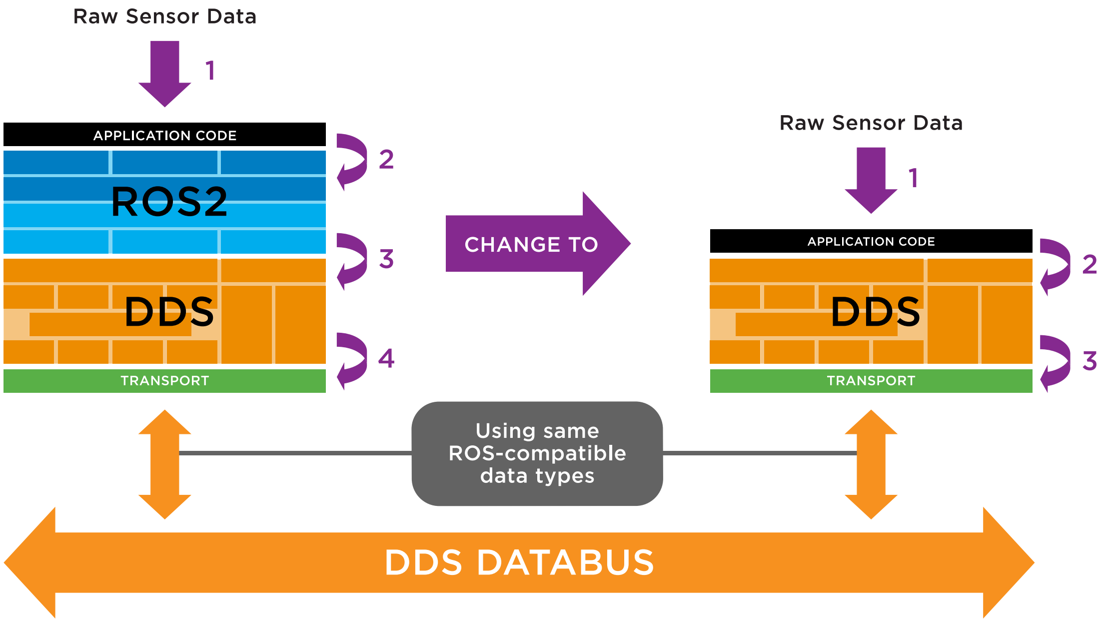

Figure 5: Developers can optimize data flow by changing the system to implement the sensor data-handling nodes directly in DDS

- Direct access to advanced DDS capabilities, such as these in RTI Connext:
    - *FlowController* - this shapes DDS traffic to avoid network saturation.
    - *FlatData* - this avoids some processing to reduce latency.
    - *ZeroCopy* - this transfers data without making any copies, dramatically improving performance of large data items.
    - *Batching* - this accumulates groups of high-rate data samples into 'batches' for more efficient sending, reducing network overhead and improving throughput.
    - *Filtering* - allows data to be sent only when it meets a value or timing criteria, thereby reducing the traffic of low-importance data on the network.

The new DDS nodes use the same ROS-compatible data types as the rest of the system; thereby ensuring compatibility with the ROS 2 nodes.

B. My system needs to be safety certified.

Safety certification can be a very expensive undertaking, so system developers will seek to minimize and isolate the portions of their system that fall under the certification and test requirements. Another strategy is to build these portions using safety-certified software components where possible, such as RTI Connext DDS CERT - a minimal-footprint, safety-certified DDS with available supporting certification evidence. The certification testing burden can be further minimized by implementing the to-be-certified portions of the system directly in DDS, without the extra layer of ROS code. This removes a large amount of code that would otherwise have to be tested and certified and helps to succinctly encapsulate the certifiable portions of your system, keeping them clear of the non-certifiable parts.

As in other examples, ROS-compatible data types can be used to maintain interoperability with ROS 2 nodes, packages and the ROS ecosystem of tools.

C. My system must run on a production-grade board and operating system.

As an open-source project, ROS 2 is officially supported on a few desktop-class PCs using general-purpose operating systems. But production-grade systems come in an endless variety of configurations, with different CPU architectures, operating systems and on-board resources.

To use ROS 2 on these platforms often requires developing the in-house expertise to not only port and verify ROS 2 on this platform, but also to take on its long-term support and maintenance.

A direct implementation in DDS offers a vast improvement in:

1. Support: RTI Connext DDS is professionally supported on more than 100 different platforms, from microcontrollers to enterprise-grade running bare-metal, RTOS and conventional operating systems. There's no porting, and you have someone to call if you run into challenges.
2. Control: A direct implementation in DDS means there's no waiting for the next ROS update to fix or break things. Your system is under your control.
3. Compatibility: Using ROS-compatible data types in your direct-DDS implementation means you can still enjoy the ROS ecosystem of tools and packages, along with the DDS ecosystem of tools and services.

D. My system needs to do more than ROS can offer.

Incorporate the rich functionality of DDS. DDS has been field-proven in thousands of the most extreme and difficult applications: Connectivity using intermittent radios? Check. Need full-time data recording and replay? Not a problem. Multiple levels of security? DDS is ready to go.

Help is available. Creators of mission-critical systems must have great discipline to face the "agony of deployment." Is my system reliable? Will it work as expected, every time, under every condition? Is it secure? What are the consequences of system failure? RTI Services has vast experience in autonomous system development and is available to help you deploy a scalable, production-grade framework that will address a full range of project requirements. This experience can help you to leverage the learnings from over 1,350 projects to design your system right from day one.

CONCLUSION

ROS 2 effectively bridges the ROS ecosystem and community (and its million-plus users) with the DDS ecosystem (and its field-proven robustness and multiple providers), thereby offering the best of both worlds for creators of complex systems. By utilizing DDS with ROS, your system really can have it all:

- The visualization/simulation tools, packages and university-trained talent pool of ROS.
- The 'works-anywhere' robustness, layered tools and services and professional support of DDS.

With this combination, your system is equipped with the key elements to design, develop and deploy a production-grade system that meets mission-critical and safety requirements.

RTI AND THE RTI CONNEXT DATABUS

RTI is the commercial leader in DDS technology. RTI Connext DDS is a complete framework for creating scalable high-performance systems. With its own ecosystem of support and services, Connext DDS is used in more than 1,300 critical systems, including:

- Over 200 autonomous vehicle platforms
- Grand Coulee Dam control system
- Kennedy Space Center launch complex (largest SCADA system in the world)
- Nearly every US Navy ship combat management system
- Robotic surgical systems (3kHz update rate on actuators)
- ISS telepresence (robotic hand control between ground and orbit)
- Wind power turbines
- Undersea robotic systems

For more information on Connext DDS, please visit www.rti.com.

ABOUT RTI

Real-Time Innovations (RTI) is the Industrial Internet of Things (IIoT) connectivity company. The RTI Connext® Databus is a software framework that shares information in real time, making applications work together as one, integrated system. It connects across field, fog and cloud. Its reliability, security, performance and scalability are proven in the most demanding industrial systems. Deployed systems include medical devices and imaging; wind, hydro and solar power; autonomous planes, trains and cars; traffic control; Oil and Gas; robotics, ships, and defense.

RTI is the largest vendor of products based on the Object Management Group (OMG) Data Distribution Service™ (DDS) standard. RTI is privately held and headquartered in Sunnyvale, Calif.

Download a free 30-day trial of the latest, fully-functional Connext DDS software today: https://www.rti.com/downloads.

Your systems. Working as one.

CORPORATE HEADQUARTERS
232 E. Java Drive, Sunnyvale, CA 94089
Telephone: +1 (408) 990-7400
Fax: +1 (408) 990-7402
info@rti.com

rti.com
rti_software
rtisoftware
company/rti
connextpodcast
rti_software