OPEN MANAGEMENT GROUP DATA-DISTRIBUTION SERVICE (OMG-DDS) AS A DATA TRANSPORT FOR VEHICULAR INTEGRATION FOR C4ISR/EW INTEROPERABILITY (VICTORY) SERVICES

Leonard Elliott
Vehicle Electronics and Architecture
TARDEC
Warren, MI
**1. REPORT DATE**  
16 AUG 2012

**2. REPORT TYPE**  
Briefing

**3. DATES COVERED**  
01-07-2012 to 01-08-2012

**4. TITLE AND SUBTITLE**  
OPEN MANAGEMENT GROUP DATA-DISTRIBUTION SERVICE AS A DATA TRANSPORT FOR VEHICULAR INTEGRATION FOR C4ISR/EW INTEROPERABILITY SERVICES

**5. AUTHOR(S)**  
Leonard Elliott

**6. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)**  
U.S. Army TARDEC, 6501 East Eleven Mile Rd, Warren, Mi, 48397-5000

**7. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)**  
U.S. Army TARDEC, 6501 East Eleven Mile Rd, Warren, Mi, 48397-5000

**8. PERFORMING ORGANIZATION REPORT NUMBER**  
#23220

**9. SPONSOR/MONITOR’S ACRONYM(S)**  
TARDEC

**10. SPONSOR/MONITOR’S REPORT NUMBER(S)**  
#23220

**12. DISTRIBUTION/AVAILABILITY STATEMENT**  
Approved for public release; distribution unlimited

**13. SUPPLEMENTARY NOTES**  
Submitted to 2012 NDIA Ground Vehicle Systems Engineering and Technology Symposium August 14-16 Troy, Michigan

**14. ABSTRACT**  
VICTORY is a U.S. Army initiative to improve upon current military ground vehicle electronics architecture by providing C4ISR/EW system interoperability and portability. VDM and OMG-DDS P/S solutions appear to be suitable for addressing integration problems which fall into the current scope of VICTORY.

**15. SUBJECT TERMS**  
VICTORY data Messages, Publish subscribe, Open management Group

**16. SECURITY CLASSIFICATION OF:**  
<table>
<thead>
<tr>
<th>a. REPORT</th>
<th>b. ABSTRACT</th>
<th>c. THIS PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>unclassified</td>
<td>unclassified</td>
<td>unclassified</td>
</tr>
</tbody>
</table>

**17. LIMITATION OF ABSTRACT**  
Public Release

**18. NUMBER OF PAGES**  
18

**19a. NAME OF RESPONSIBLE PERSON**  
unclassified
Introduction

• VICTORY is a U.S. Army initiative to improve upon current military ground vehicle electronics architecture by providing C4ISR/EW system interoperability and portability.
• VICTORY Data-Messages (VDM) are VICTORY’s customized publish-subscribe (P/S) messaging solution.
• OMG-DDS is an open-standard COTS P/S messaging solution.
• Investigate replacing VDM with OMG-DDS interfaces
VICTORY Overview

• The VICTORY technical approach
  – Incorporate open-standards
  – Shared services (VICTORY services)
  – Data-bus centric (VICTORY Data-Bus [VDB])

• VICTORY Standards development
  – Working groups (government, industry)
  – Adopt-adapt-author methodology

• Current scope of VICTORY
  – No real-time applications
  – No safety critical applications
  – Intra-vehicle communication (i.e. high-availability)
• VICTORY core services (Position, Orientation, Direction-of-Travel, and Time)
• Service data interface
  – Publish data to VDB using available formats or VDMs
• Service management interface
  – Monitoring/control functions available via Simple Object Access Protocol (SOAP) Remote Procedure Calls (RPC)
Publish-Subscribe Communication Overview

- Provides loose-coupling and scalability
- P/S communication implementation
  - Centralized broker
  - Multi-broker
  - Peer-to-peer
- P/S technologies generally provide API and/or wire protocol standards
  - API standard: code portability
  - Wire protocol standard: “on-the-wire” interoperability
- Integration pattern supported by many COTS messaging technologies
• Several COTS technologies supporting P/S:
  – Java Message Service (JMS): standard Java API, centralized or brokered topology
  – Advanced Message Queuing Protocol (AMQP): standard wire protocol, multi-brokered
  – OMG-DDS: standard API and wire protocol, peer-to-peer implementations

• OMG-DDS is a good candidate for on-vehicle COTS P/S solution
  – Data-Centric Publish Subscribe (DCPS) API standard
  – Real-Time Publish Subscribe (RTPS) wire protocol
  – Peer-to-peer implementation (middleware spawn threads in app)
  – Mechanisms for supporting real-time P/S
VICTORY Data Message (VDM)

- Customized messaging solution supporting P/S on VDB
  - Used for services with no standard data type/protocol (e.g. Time service uses NTP/PTP not VDM)

- Provides scalability and loose-coupling
  - Internet Group Management Protocol (IGMP) to manage subscriptions
  - UDP multicast for data distribution

- Quality of Service (QoS)
  - Differentiated Service Code Point (DSCP): 6-bit field in the IP header
  - Indicates traffic priority
VICTORY Data Message (VDM)

- Standard message features including sequence numbers, format indicators, identifiers, and timing information
- Binary header with and XML payload:
**VICTORY Data Message (VDM)**

- **VDM P-S Behavior (control via SOAP RPC)**
  - Increase/decrease publishing frequency
  - Enable/disable publishing
  - Change multicast address and port for publishing

- **TARDEC reference implementation**
  - Standard POSIX libraries (e.g. sys/sockets.h)
  - C++
  - Red Hat Enterprise Linux (RHEL) on x86_64
  - RHEL on i386.
  - Ubuntu on ARM Cortex-A8

- **Good performance over high-availability network**
OMG-DDS Overview

- Open-Standard, COTS, real-time P/S, with multiple vendors.
- Extensive QoS for turning P-S behavior:
  - Deadline: The maximum time between data samples.
  - Durability: Previously published data can be stored and sent to late joining subscribers.
  - Lifespan: Specifies how long data sent by user application is considered valid.
  - Liveliness: Allows readers to detect when matching writers are no longer available.
  - Ownership: Specifies ownership of a Topic by a specific writer.
  - Reliability: Allows recovery of samples lost by the network.
OMG-DDS Insertion into VICTORY Services

- Convert VICTORY Position Service
  - Replace VDM interface with OMG-DDS
  - Test with multiple OMG-DDS implementations (RTI Connext DDS 4.5f and PrismTech OpenSplice 6.1.1)

- Evaluate
  - Development effort
  - API completeness and conformance (i.e. vendor-specific code)
  - Interoperability
OMG-DDS Insertion into VICTORY Services

- Convert VICTORY XML Schemas (XSD) to OMG-DDS Interface Definition Language (IDL).

```xml
<xsd:simpleType name="latitudeBounds_t">
  <xsd:restriction base="xsd:double">
    <xsd:minInclusive value="-90"/>
    <xsd:maxInclusive value="90"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="longitudeBounds_t">
  <xsd:restriction base="xsd:double">
    <xsd:maxInclusive value="180"/>
    <xsd:minExclusive value="-180"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="absolutePosition_t">
  <xsd:sequence>
    <xsd:element name="latitude" type="vmt:latitudeBounds_t"/>
    <xsd:element name="longitude" type="vmt:longitudeBounds_t"/>
    <xsd:element name="altitude" type="xsd:double"/>
    <xsd:element name="mgrs" type="xsd:string" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```cpp
module VICTORY
{
  module Types
  {
    struct doubleMeasurement_t
    {
      double value;
      double uncertainty;
      boolean estimated;
      boolean valid;
    }; //@top-level false
    // Military Grid Reference System
    struct MgrsPos_t
    {
      string<10> gridZoneDesignator;
      long easting;
      long northing;
    }; //@top-level false

    struct AbsPos_t
    {
      doubleMeasurement_t latitude;
      doubleMeasurement_t longitude;
      doubleMeasurement_t altitude;
      boolean hasMgrs;
      MgrsPos_t mgrs;
    }
    #pragma keylist AbsPos_t
  }; //Types
}; //VICTORY
```
OMG-DDS Insertion into VICTORY Services

- Modify Position Service code to use OMG-DDS interface (several days of training and ~100 lines of code)

On Publisher (i.e. Position Service)

1. Declare and initialize DDS Entities
2. Map received sensor data to DDS-data types
3. Call Data-Writer `write()` to Publish

On Subscriber (i.e. Position Service Client)

1. Declare and initialize DDS Entities
2. Call Data-Reader `take()` to read sample
3. Map received DDS data to user-types

UNCLASSIFIED
OMG-DDS Insertion into VICTORY Services

- Tested all combinations of RTI and PrismTech for modified Position Service and Position Service Client.
- Default OMG-DDS QoS used (VDM behavioral equivalent):

<table>
<thead>
<tr>
<th>QoS &amp; Attributes</th>
<th>Writer</th>
<th>Reader</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deadline</td>
<td>Infinite</td>
<td>Infinite</td>
</tr>
<tr>
<td>Domain ID</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Durability</td>
<td>Volatile</td>
<td>Volatile</td>
</tr>
<tr>
<td>Latency Budget</td>
<td>0 sec</td>
<td>0 sec</td>
</tr>
<tr>
<td>Liveliness duration</td>
<td>Infinite</td>
<td>Infinite</td>
</tr>
<tr>
<td>Liveliness kind</td>
<td>Automatic</td>
<td>Automatic</td>
</tr>
<tr>
<td>Ownership</td>
<td>Shared</td>
<td>Shared</td>
</tr>
<tr>
<td>Reliability</td>
<td>Reliable</td>
<td>Best Effort</td>
</tr>
<tr>
<td>Topic</td>
<td>Position</td>
<td>Position</td>
</tr>
<tr>
<td>Type</td>
<td>AbsPos_t</td>
<td>AbsPos_t</td>
</tr>
</tbody>
</table>
Code Modification Results

- Vendor specific code changes (3 statements)
  - Casting of generic data writer to type-specific data-writer
  - User defined type-support – no native *interface* support for C++
  - Disable default vendor-specific transport setting
- Example:

```cpp
#ifdef RTI
ddsSpecificWriter =
    VICTORY::Types::AbsPos_tDataWriter::narrow(
        ddsDataWriter);
#endif

#ifdef PRISMTECH
ddsSpecificWriter = VICTORY::Types::
    AbsPos_tDataWriter::_narrow( ddsDataWriter);
#endif
```
Interoperability Results

- Applications on separate hosts worked out-of-the-box
- Applications on same host required modification
  - RTI Connext DDS defaults to shared memory transport (fast)
  - Change transport to standard UDP transport:

    ```
    #ifdef RTI
    ddsDomPartQos.transport_builtin.mask = DDS_TRANSPORTBUILTIN_UDPv4;
    #endif
    ```
• Features of VDM and OMG-DDS compared.

<table>
<thead>
<tr>
<th>Aspect</th>
<th>VDM</th>
<th>OMG-DDS</th>
</tr>
</thead>
<tbody>
<tr>
<td>COTS Available</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Lines of Code</td>
<td>1500</td>
<td>100</td>
</tr>
<tr>
<td>Licensing Issues</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Standard API</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Standard Wire Protocol</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Unbrokered Architecture</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

• Notable Issues:
  – VDM Development complexity
  – Licensing OMG-DDS (based on # developers, computing platform, tools, etc).
Conclusion

• VDM and OMG-DDS P/S solutions appear to be suitable for addressing integration problems which fall into the current scope of VICTORY.

• OMG-DDS is a powerful tool
  – Built on open-standards
  – Multiple vendor implementations
  – Well-supported API and wire protocol
  – May enable more capabilities on VDB

• Continue to investigate
  – VDM versus OMG-DDS cost for vehicle program
  – Performance
  – OMG-DDS advanced features and use cases

• The VICTORY Work Group should consider adopting OMG-DDS as an incremental enhancement or real-time extension should the need for more complex P/S behavior arise.