A Common Service-oriented Infrastructure Approach for Defence Tactical Environments

Derek Dominish
Aerospace Division
Future Information Architectures

OMG – Berlin 17th June 2013
The NCW Integration Complexity Problem

**Problem Space**
- Network-centric, dynamic, very large-scale “systems of systems”
- Stringent simultaneous quality of service (QoS) demands
- Highly diverse & complex problem domains

**Solution Space**
- Enormous accidental & inherent complexities
- Continuous evolution & change
- Highly heterogeneous environments

Mapping & integrating *problem artifacts* to *solution artifacts* is extremely difficult

Adapted from: “Overview of the OMG Data Distribution Service”, Schmidt/Parsons (DDS.ppt; pp 7)
Open Architecture through Open Standards

Surface System

Aircraft System

Land System

Sub-Surface System

Service Oriented - Open Architecture

Common Computing Environment

Common Functions

Surface Combatant Unique Functions

Surface System Unique & Common Applications & Interfaces

Aircraft Unique Functions

Aircraft System Unique & Common Applications & Interfaces

Land Unique Functions

Soldier System Unique & Common Applications & Interfaces

Sub-Surface Unique Functions

Submarine System Unique & Common Applications & Interfaces

Standards Based Computing Environment

Common Computing Environment

Common Functions

Common Platform Functions

Platform Unique Functions

Interoperable System-of-Systems

Common Middleware Bus Architecture

Adapted from: “Open Architecture”, Strei; 2/2/2004 (NOAbef.ppt; slide 4)
Net Centric Mission Environment

Levels Of Information Systems Interoperability (LISI) Reference Model

Adapted from: “Overview of the OMG Data Distribution Service”, Schmidt/Parsons (DDS.ppt; slide 8)
Layered Reference Architecture
Tactical SOA (OA)

The Strategic Architecture Reference Model (SARM)

The SARM is:
- an open communications and information architecture framework supported by NCOIC,
- based upon commercial and government interfacing standards (IDL),
- organized to address system-wide network design issues,
- an enabling technology framework to allow platforms and tactical systems to interface to the Global Information Grid,
- allows for interoperable nodes on the network.

Adapted from:
Technical Reference Model for Network-Centric Operations
Bradley C. Logan. The Boeing Company
(Crosstalk Aug 2003; Vol16, No8)
Tactical SOA Reference Architecture

Business Services

- a unit of autonomous behaviour that meets a particular business need,
- constructed / assembled from one or more components that support a business ontology,
- conform to an infrastructure plug and play deployment policy,
- orchestrated through an application,
- collaborate with other peer services within a network,
- location and machine architecture independent,
- built upon standardised infrastructure services and mechanisms,
- often needed to adapt to legacy environments,
- expressed through formal and interoperable service interface definitions (IDL [PIM]) to the network.
- not necessarily reliant on any particular networking protocol or communications infrastructure.
Net-Centric Reference Architecture (SOA)

Platform Environment – Posix Compliant Operating System / Services (Solaris/Windows)

Communications Environment – Transports / Protocols (TCP/UDP/IIOP)

Information Infrastructure Core - Middleware Environment (CORBA/DDS/RMI)

Domain Environment (CCM/EJB/Web-Services)

Information Assurance Filter

Search / Query

Interface Control

Database Tables

Schema

Device/System

Security Boundary

Ontology / Schema

Application

Interface Control

Fusion

Correlation

Application

API Adaptor

Configuration

Domain Environment

COP Generator

COP Adaptor

Application Configuration

Correlation

Fusion

Ontology / Schema

Database Tables

Search / Query

Information Assurance Filter

Interface Control

Database Tables

Security Boundary

COP Generator

COP Adaptor

Device/System

Domain Environment (CCM/EJB/Web-Services)
System- of- Systems Integration Infrastructure Architecture
A Standardised Middleware Approach

System Integration Through Service Composition

Service Infrastructure – Middleware Fabric (L0)

Information Assurance
- Communication: QoS, encryption, bandwidth
- Security: boundaries, classification, communication routes
- Authentication: information ownership, need to know
Interoperating across vendors and systems
  - Ontological Conformity
Integrate “services” through layering
Standardisation
  - Interfaces
  - Protocols
Hooking into appropriate layer
Easier, cheaper to change and upgrade
Open business practices
Indeterminate liability
  - Certification
Application Server

‘Application Store’

Application Servers:
- manage component service complexity,
- dynamically and statically load and configure service components,
- provide infrastructure capabilities and services to hosted components;
  - logging,
  - aids to debugging,
  - instrumentation,
  - capabilities dynamically configurable and extensible.
- manage coupled middleware environment;
  - security and access policy enforcement,
  - Communications, protocols and bearers,
  - … many other environmental aspects.
- manage lifecycle of hosted components (initialisation, shutdown, suspend, resume),
- facilitate the navigability to individual component interface implementations,
- manage the optimisation of peer component interactions through collocation,
- location and machine architecture independent,
- built upon standardised infrastructure services mechanisms and patterns,
- supports a network accessible service control and status interface.
Technical Architecture

‘putting it all together’
Net Warrior D10 Process Domain

MSRC (ASCEL - AOD)
- Recorded Feed
- Data-Link Gateway
- Network Management System

TADIL Network (L16)

Middleware
- IDL/WSDL/SOAP
- Network / Physical
- OMG - DDS/CORBA
- OMG - DDS/CORBA IDL/WSDL/SOAP

External Feeds

ISRAIL (ISRD)
- ROC
- Correlation and Tracking
- Full Motion Video
- ISR Integration (ADIIB)

FOCAL (C3ID)
- Command Centre
- Visualisation
- GIS

Legend of Technology
- Tactical Data Links (Link-16)
- OMG - Data Distribution Service (DDS)
- OMG - Common Object Request Broker Architecture (CORBA)
- IBM - Web-Services (SOAP)
- Extensible Mark-Up Language (XML)
- Proprietary Technology / Interface

Legend of Adapter Transform
- DDS - CORBA
- DDS - XML
- DDS - Web-Services
- Proprietary - Web-Services
Net Warrior (D10 Event) Bus Architecture
A Standardised Middleware Approach - Demonstration Event

Real Time Publish/Subscribe & Synchronous Control Bus – (CORBA/DDS)

Enterprise Information Service Bus – (Web-Services)

Tactical Data Link Network – (Link16/Link11/VMF)
Demonstration Artefacts - This really works!

The Build Process.

- Create a service project MakeProjectCreator descriptor for MPC toolset

  ```
  project(CORBAService): taflib, taolib_with_idl
  {
  sharedname = *
  idlflags => -Wb.export_macer=\CORBAService_Export
  libout = $(DAF_ROOT)/lib
  libpaths => $(DAF_ROOT)/lib
  macros => CORBASERVICE_BUILD_DLL
  prebuild = perl $(ACE_ROOT)/bin/generate_export_file.pl CORBAService > CORBAService_export.h
  IDL_Files (CORBAService.idl)
  Header_Files (CORBAService.h)
  CORBAService.export.h
  }
  inline_files ()
  source_files (CORBAService.cpp)
  }
  ```

- Use MakeProjectCreator (MPC) to build solution files for platform toolsets (Windows/Linux ...).

  ```
  perl %ACE_ROOT%/bin/mwc.pl -type vc10 -name_modifier *_vc10 -apply_project TAF_DAF.mwc
  ```

- Load project into toolset environment (i.e. Windows - VC10)

  ```
  part %ACE_ROOT%/bin/mwc.pl -type vc10 -name_modifier*_vc10 -apply_project TAF_DAF.mwc
  ```

- Build binaries with toolset applicable to platform environment.

The Development Process.

- Define an interface (IDL)

  ```
  module DSTO
  {
  struct Structure
  {
  short i;
  long j;
  string k;
  }
  }
  ```

- Implement interface providing service descriptors for application server deployment control (CORBAService.cpp).

  ```
  DAF_EXPORT_FACTORY_DEFINE(DSTO,CORBAService);
  DAF_EXPORT_SVC_DEFINE(DSTO, CORBAService
    , DSTO::CORBAService::svc_name()
    , ACE_SVC_OBJ_T
    , &DSTO_MAKE_SVC_FACTORY_NAME(DSTO,CORBAService)
    , (ACE_Service_Type:DELETE_THIS | ACE_Service_Type:DELETE_OBJS)
    , false // service not initially active
    );
  ```

- Deploy with configuration (DSTO.conf)

  ```
  dynamic DSTO_DSEExample Service_Object * OpenDSEexample:make_DSTO_DSEExample()
  dynamic DSTO_CORBAExample Service_Object * CORBAExample:make_DSTO_CORBAExample()
  ```
Questions?

Just got an order from our sophisticated networking system. What is it?

Twenty push-ups

The Age – Australian Newspaper