Subsea and topsides controls integration: the new MDIS integration standard

David Walker
Chief Engineer
Yokogawa Australia

20 Oct 2016
1. Overview
2. OPC-UA
3. Subsea Architecture and Functionality
4. MDIS
5. Summary
Overview
What is MDIS

- **MCS DCS Interface Standardization**
  - MCS = subsea Master Control Station
  - DCS = topsides Distributed Control System

- OPC-UA based protocol

- Vendor and platform independent

- Object (device) based
The Need For MDIS

- Currently no standardized approach to DCS/Subsea communications

- This means that there is risk associated with complexity, data mapping, communications compatibility, and misunderstanding in requirements between Subsea vendor, DCS vendor and client

- Communication is a significant aspect of a DCS/Subsea integration project
Developed out of Microsoft OPC (classic):

- Three different types: OPC-DA, OPC-AE, OPC-HDA
- Platform dependent (Microsoft Windows)
- Could not be used in control devices (because they do not use Windows)
- Based on outdated Windows NT technology (OLE)
OPC-UA uses the good features of OPC:

- Vendor independent – no knowledge of the subsystem required
- Access to real time data, historical data and event messages but combined into a unified architecture
- Object grouping

  eg: SUBSEA.MANIFOLD01.WELL01.CV001
OPC-UA has added advantages:

- Platform independent – does not require Windows and so can run on embedded devices

- Object oriented – can create objects representing a device (such as a valve) and all data is encapsulated in the object (XML definitions). This is the basis for MDIS.

- Secure – OPC-UA has been developed from the ground up for secure communications
OPC-UA Specification
OPC-UA Data Model - Example

Create a model as an OPC-UA object.

Create instances of this object.
OPC-UA – External Information Models

 Already available

- IEC 61131-3 (PLCopen)
- ISA-95 (Enterprise Information)
- Many more……

 More planned

- ISA-95 (Phase 2)
- PRODML/WITSML
- …….

Very exciting – allows integration of controls and communications
Subsea Architecture
Subsea Controls Architecture

**MDIS Architecture Definition**

**Integrated**
- DCS HMI
- MCS (DCS Controller)
- SCI (subsea gateway)
- SEM

**Interfaced**
- DCS HMI
- DCS Controller (gateway)
- SCI (subsea gateway)
- SEM

- **DCS Vendor Scope**
- **Subsea Vendor Scope**

*Topside*  
*Subsea*
Architecture – Integrated System

- **MCS** (Master Control Station)
  - Logics (Interlocks, Sequences, etc)
  - Arbitration (redundant signals)

**Integrated System**

- **MCS** (Master Control Station)
- Logics (Interlocks, Sequences, etc)
- Arbitration (redundant signals)

**MDIS** (Medium Device Interface System)

**Integrated Control System**

- **SCI** (Subsea Communication Interface)
- **Subsea** Vendor Scope

- **DCS Controller**
- **Control Bus**

**Topsides**

- **Topsides PCS**
- **Control Bus**

**Subsea**

- **Umbilical** (Fibre optic/Copper/Power)
- **Communication Distribution Unit**
- **Subsea Control Module**

**CDU** (Communication Distribution Unit)

**SCM** (Subsea Control Module)
The MDIS Standard defines the following minimum functionality:

- Arbitration
- Interlocks
- Sequences
- Valve Status Validation
- Choke Position Validation
- Interfacing with the HPU and CIS
- Validation of Valve Profiles/Signatures
These are encapsulated in software modules designed for common devices in the SCMs

- Arbitration
- Valve Controls
- Indicators
- Start-up and Shutdown Sequences
- MEG Sequence

Modules expose data in MDIS format
MDIS = MCS DCS Interface Standardization

- An implementation of OPC-UA
- Specifies a set of OPC-UA objects that are common to all subsea communications
## MDIS Object Types

<table>
<thead>
<tr>
<th>MDIS Object</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discrete Instrument</td>
<td>Multi-state type data (such as valve position – open/moving/closed)</td>
</tr>
<tr>
<td>Digital Instrument</td>
<td>On/Off functions</td>
</tr>
<tr>
<td>Instrument</td>
<td>For analogue data (eg, pressure, flow)</td>
</tr>
<tr>
<td>Choke</td>
<td>Choke Valves</td>
</tr>
<tr>
<td>Valve</td>
<td>All other valves</td>
</tr>
</tbody>
</table>
Discrete Instrument (Multistate Device)
Digital Instrument (On/Off Device)
Instrument

Status Information
- Fault
- FaultCode (O)
- Warning (O)
- WarningCode (O)
- Enabled (O)

Commands
- Enable Disable (O)

Configuration
- TagId (O)

Process Information
- HHSetPoint (O)
- HSetPoint (O)
- LSetPoint (O)
- LLSetPoint (O)

MDISBaseObjectType

MDISInstrumentObjectType

MDISInstrumentOutObjectType

Subsea Controls Down Under
19 & 20 October 2016

24
Analogue Indicators

- Design based on MDIS Instrument Object
- Arbitration built in to the functional design
- Alarms and engineering range part of the MDIS Instrument object definition
MDIS Choke Valve

Choke Valve Object

Excerpt from the diagram:

- **Status Information**
  - Fault
  - FaultCode(O)
  - Warning(O)
  - WarningCode(O)
  - Enabled (O)

- **Commands**
  - EnableDisable (O)

- **Configuration**
  - TagId(O)

- **Interlocks**
  - HasInterlock
  - InterlockFor
  - InterlockFor
  - CommandRejected (O)

- **Info**
  - CalculatedPosition
  - PositionInSteps (O)
  - Moving
  - CommandRejected (O)
  - NonDefeatable OpenInterlock (O)
  - NonDefeatable CloseInterlock (O)
  - Defeatable OpenInterlock (O)
  - Defeatable CloseInterlock (O)

- **SetCalculated Position**
  - StepDuration Open(O)
  - StepDuration Close(O)
  - TotalSteps(O)
Production Choke Valves

- Can be a complex device depending on operator requirements
- Hydraulically stepped to open and close
- Different modes of operation
  - Step open/close
  - move to a percentage
  - Calculated within the MCS
- Product interlocks (Hydraulic and Electric)
- Data and command structure in MDIS format
MDIS Valve Object

Valve Object

Status Information
- Fault
- FaultCode(O)
- Warning(O)
- WarningCode(O)
- Enabled (O)

Commands
- EnableDisable (O)
- Move

Configuration
- TagId(O)

Status
- SignatureRequest (O)
- LastCommand (O)
- Position
- CommandRejected (O)
- CloseInterlock (O)
- Defeatable CloseInterlock (O)
- Defeatable OpenInterlock (O)
- NonDefeatable OpenInterlock (O)
- NonDefeatable CloseInterlock (O)
- InterlockFor

HasInterlock

InterlockVariableType::
- <InterlockPlaceholder>

InterlockFor

<Interlocks>
- InterlockVariableType::
- InterlockPlaceholder

MDISBaseObjectType

MDISValve ObjectType
Single Actuated Valves

- Design based on MDIS valve object
- Arbitration built in to the functional design
- Product interlocks and process interlocks as defined in the MDIS specification
- Customised faceplate includes diagnostic information and detailed feedback information via pop-ups
Creating new objects (Aggregation)

Example for creating an aggregated model of a simple CIMV object. A generic model of a CIMV has the following items:

- **Valve Device State** – Discrete Instrument
- **Valve Status** – Discrete Instrument
- **Valve Flow Target** - Instrument
- **Valve Position Target** - Instrument
Chemical/MEG Control Valves

- Complex device with many diagnostic values
- Different modes of operation
- Usually electrically controlled (no hydraulic restrictions)
- Data and command structure in MDIS format
MDIS – Controls Integration

MDIS enables the integration of communications and controls

- Linked to IEC61131-3 (PLCOpen – XML based)
- Function Blocks in IEC61131-3 can mirror MDIS blocks
- Control functionality and communications are therefore defined at the same time
Summary
Summary

- MDIS provides a standardized approach to DCS/Subsea communications
  - Reduces engineering effort
  - Reduces risk
  - Provides DCS and Subsea system independence
  - Integrates with control standards (IEC 61131-3)
References


- **OPC UA**: [https://opcfoundation.org/about/opc-technologies/opc-ua/](https://opcfoundation.org/about/opc-technologies/opc-ua/)
Any questions?