“Supporting Agile C2 with an Agile and Adaptive IT Ecosystem”

ICCRTS 2012, paper 044, Track 1

Harvey Reed,
Principal, Multi-Party Engineering, MITRE
hreed@mitre.org
Drivers
Need to address adaptability, timeliness, cost, alignment

DoD/CIO Campaign Plan, Oct 2011
- “…Therefore, the DoD CIO must address this challenge by providing the advocacy and guidance necessary to facilitate agile, rapid delivery of effective, secure information capabilities across all missions and functions.”

Joint C2 Capability AoA Conclusions and Recommendations, March 2011
- "Sustainment costs of current C2 capabilities dominate"
- "A joint C2 Modernization Strategy that comprehensively addresses both capability improvements and the transformation / migration of legacy capabilities is needed"
- No single program (e.g. NECC), rather a federated “Joint C2 Family-of-Programs”
- Critical function - C2 Enterprise-Wide Architecture Development & System Engineering

“Mission Command”, GEN Martin Dempsey, Army Magazine, Jan 2011
- “Confronting hybrid threats—combinations of regular, irregular, terrorist and criminal groups—in such an environment requires leaders who not only accept but seek and embrace adaptability as an imperative.”
Classic Standalone System Baselines

Current Methods
Deliver against a complete set of requirements, delivered over a long period of time as a singular tested, accredited capability

Challenges
Requirements are not well known in advance
Large capabilities have long delivery times
Capabilities difficult to modify in the field

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>M_i</td>
<td>Mission information</td>
</tr>
<tr>
<td>M_T</td>
<td>Mission thread</td>
</tr>
<tr>
<td>M_F</td>
<td>Mission functionality</td>
</tr>
<tr>
<td>C_M</td>
<td>Core middleware</td>
</tr>
<tr>
<td>C_H</td>
<td>Core hardware</td>
</tr>
<tr>
<td>C_N</td>
<td>Core network</td>
</tr>
</tbody>
</table>
Decouple into Components

\[ M_I \]
\[ M_T \]
\[ M_F \]
\[ C_M \]
\[ C_H \]
\[ C_N \]

Local

Need to Decouple

Mission Unique
- High need to address specific mission needs
- High need to tailor in the field

Shared Infrastructure
- High need to share across missions
- High need to consolidate

Based on public released documents 12-0212,12-1273

© 2012 The MITRE Corporation. All rights Reserved.
Multiple Component Providers

- Large number of mission specific component producers
- Smaller number of core component producers
- Core components can be reused
- Shared Agreements assure components can be assembled

Independent Component Development

CORE COMPONENT PROVIDERS

MISSION COMPONENT PROVIDERS

Shared Agreements
Assemble Capabilities

Independent Capability Assembly

Capabilities assembled using independently produced components
- Enabled by Shared Agreements
Components are a mix of core and mission specific
Feedback from end user to component producers
Capability adaptation in field is enabled

Subsequent component and shared agreement versions
Multi-Party Engineering – Tenets

Tenet #1 – Provide Small Components
- Short timeframes
- Version-able
- Suitable for iterations as requirements evolve

Tenet #2 – Certify Components to Shared Agreements
- Shared Agreements constrain usage to assure ability to assemble later
- Shared Agreements cover security, accreditation, testing, data semantics, etc.

Tenet #3 – Offer Components in Markets
- Markets enable component production and capability assembly to be decoupled in time, yet have integrity (via Shared Agreements) in the final assembly

Tenet #4 – Assemble Capabilities
- Capabilities are assembled from components which are certified to applicable shared agreements

Tenet #5 – Feedback Loops
- End users give direct feedback to the markets and component producers
- The feedback is made available to the component providers, captured in the markets, and drive future component development
Components, Shared Agreements, and Governance

Independent Capability Assembly

- Capabilities assembled using independently produced components
- Components are a mix of core and mission specific
- Feedback from end user to component producers
- Capability adaptation in field is enabled

Multi-Modal Shared Agreements

- Shared agreements bind mission with core components, as well bind mission components in a workflow
- Shared agreements used at development time for components, as well as for run time, as in SLAs.
Types of Shared Agreements

**VERTICAL SHARED AGREEMENTS**
- E.g. Mobile apps, Joint C2 UI, Agile Client, service hosting

**HORIZONTAL SHARED AGREEMENTS**
- E.g. Mission thread, process
- E.g. Federation between enclaves, tents

**CAPABILITY SHARED AGREEMENTS**
- E.g. Complete hosted capability

Many types of shared agreements:
- Security
- Data
- Data sources
- Other...

**M** Mission component

**C** Core infrastructure component

Assembled Capability
EXAMPLE
Generic “Pluggable UI” Marketplace Relationships
Mobile, Widget, Plug-in…

Component Provider

Provider

certifies / uploads mission-specific components to the

Provider
certifies / uploads core components to the

Provider
certifies / uploads infrastructure components to the

Market Provider

Mission Component Governance

Mission Components Marketplace

promotes common mission functionality to the

Core Component Governance

Core Components Marketplace

uses core plug-ins from the

Infrastructure Governance

Framework / Infrastructure Components Marketplace

installs the

Consumer / Assembler

Selects & downloads mission plug-ins from the

Core Component Governance

Core Components Marketplace

uses core plug-ins from the

Mission Component Governance

Mission Components Marketplace

promotes common mission functionality to the

Based on public released documents
12-0212,12-1273

© 2012 The MITRE Corporation. All rights Reserved.
Migration Concepts

Joint C2 Migration includes
- Modernization
- Deprecation
- New

Joint C2 Migration assumes
- Based on Joint C2 Objective Architecture
- Incremental approach, no big bang

Modernization
- Deconstruct baseline
- Harvest future components
- Create applicable shared agreements
- Add to agile and adaptive ecosystem

Agile and Adaptive Ecosystem
- Grown over time
- Evolves
- No single complete specification

New development, e.g.:
- Mobile Apps
- Agile Client Plugins
- Ozone Widgets
- Information Services
- Infrastructure

Modernization, e.g.:
- Deconstruct legacy baseline
- Deprecate
- Rehab

Agile and Adaptive Ecosystem
Questions?