MTR070326

MITRE TECHNICAL REPORT

Tactical Edge Characterization Framework

Volume 2: Design Patterns for Tactical Environments

September 2007

Dr. Fatma Dandashi
Aaron Griggs
Jeffrey Higginson
James Hughes
Wilson Narvaez
Dr. Marwan Sabbouh
Salim Semy
Dr. Beth Yost

Sponsor:	OSD (NII)	**Contract No.:**	W15P7T-07-C-F600
Dept. No.:	E135	**Project No.:**	0707ECSE-CA

The views, opinions and/or findings contained in this report are those of The MITRE Corporation and should not be construed as an official Government position, policy, or decision, unless designated by other documentation.

MITRE

Executive Summary

Many Service Oriented Architecture (SOA) approaches in use today presume the availability of reliable, consistently available networks that provide limitless bandwidth and little or no latency. Since this is often not the case in the Department of Defense (DoD) or other Government tactical environments, the current methods of development may not provide reliable capability to users within this environment. In this paper, we propose a method for capturing design patterns for the tactical edge using the common vocabulary of a characterization framework described in Volume 1 of this set. We also provide a set of design patterns that minimize technical constraints associated with the disadvantaged tactical edge user, and derive the infrastructure requirements needed to implement a selected design pattern. This implementation serves as a reference, geared toward demonstrating the use of the characterization framework and validating the design patterns.

Based on the identification and extraction of design patterns for the tactical edge using the common vocabulary of the edge characterization framework, the following recommendations are proposed:

- The DoD should adopt the use of common, proven design patterns for the development of information services in disadvantaged environments, and use these patterns to help make investment decisions on infrastructure requirements and resource prioritizations.

In support of developing and fielding information systems and services to address the unique nature of tactical environments, the DoD should leverage the characterization framework in describing tactical environments and system design patterns as part of the acquisition process. This paper proposes an initial set of possible design patterns and infrastructure requirements needed to support tactical operations. (The list will continue to be augmented over time.) The DoD should work with appropriate stakeholders to adopt the use of these and similar design patterns, incorporating their use in to the acquisition process to ensure enhanced capability at the tactical edge.

- The DoD should engage industry to promote the development of tactical edge solutions and highlight existing implementations suited for edge users.

A subset of the design patterns identified in this paper, and the infrastructure required to implement them have been employed in a reference implementation. We recommend this successful use of proven design patterns be leveraged in the implementation of systems at the tactical edge as a feasible, fast, and economical way forward for DoD. To propagate this approach, existing guidance documents should be updated to recommend the approach and describe the reference implementation for others to adopt.

- The existing DoD net-centric guidance documentation, such as Net-Centric Implementation Documents (NCIDs) and Net-Centric Enterprise Solutions for Interoperability (NESI), should review the results of this work and consider adoption of the common vocabulary (as described in Volume 1 of this set) and design patters described here.

Highlighting the design solutions and approaches described here provides an effect manner to reuse proven solutions and enhance interoperability and delivery of information services to edge users. Vehicles such as NCIDs and NESI represent expanding guidance documents where the edge environment could be captured and design guidance recorded for development and test communities.

Table of Contents

List of Figures

1 Introduction

Many Service Oriented Architecture (SOA) approaches in use today presume the availability of reliable, consistently available networks that provide limitless bandwidth and little or no latency. Since this is often not the case in Department of Defense (DoD) or other Government tactical edge environments (defined as one in which the users operate within certain constrained environments such as limited communications connectivity and limited storage availability), current methods of development may not provide reliable capability to disadvantaged edge users. The challenge then lies in determining where traditional SOA approaches apply and support the edge user. Furthermore, in the case where current SOA approaches do not apply, how do we implement SOA-based solutions to operate within these environments?

In this paper, we propose a method for addressing SOA implementations within the constrained tactical environments described in Volume 1 of this paper. To do this, we propose a framework to describe the operational environment, capture design patterns using a common vocabulary, and derive infrastructure requirements from these patterns. We then employ this characterization framework for a particular use case, identifying a set of design patterns that minimize technical constraints associated with the use case. This implementation serves as a reference implementation for the characterization framework.

We begin with a brief overview of the characterization framework and the common vocabulary (similar to that provided in Volume 1). We then present an overview of the design patterns based around the following components: Name, Description, Problem, Context, Trade-offs, Illustration, and Solution Example. The Problem and Context components are defined using the common vocabulary of the characterization framework. The Illustration component is typically a graphic that demonstrates the use of the design pattern. We then employ the characterization framework and design patterns to illustrate a solution for a particular tactical use case. Finally, we highlight how design patterns can be used to derive infrastructure requirements, and conclude with a brief discussion of the implications of this work and further efforts.

This page intentionally left blank.

2 Characterization Framework and Common Vocabulary

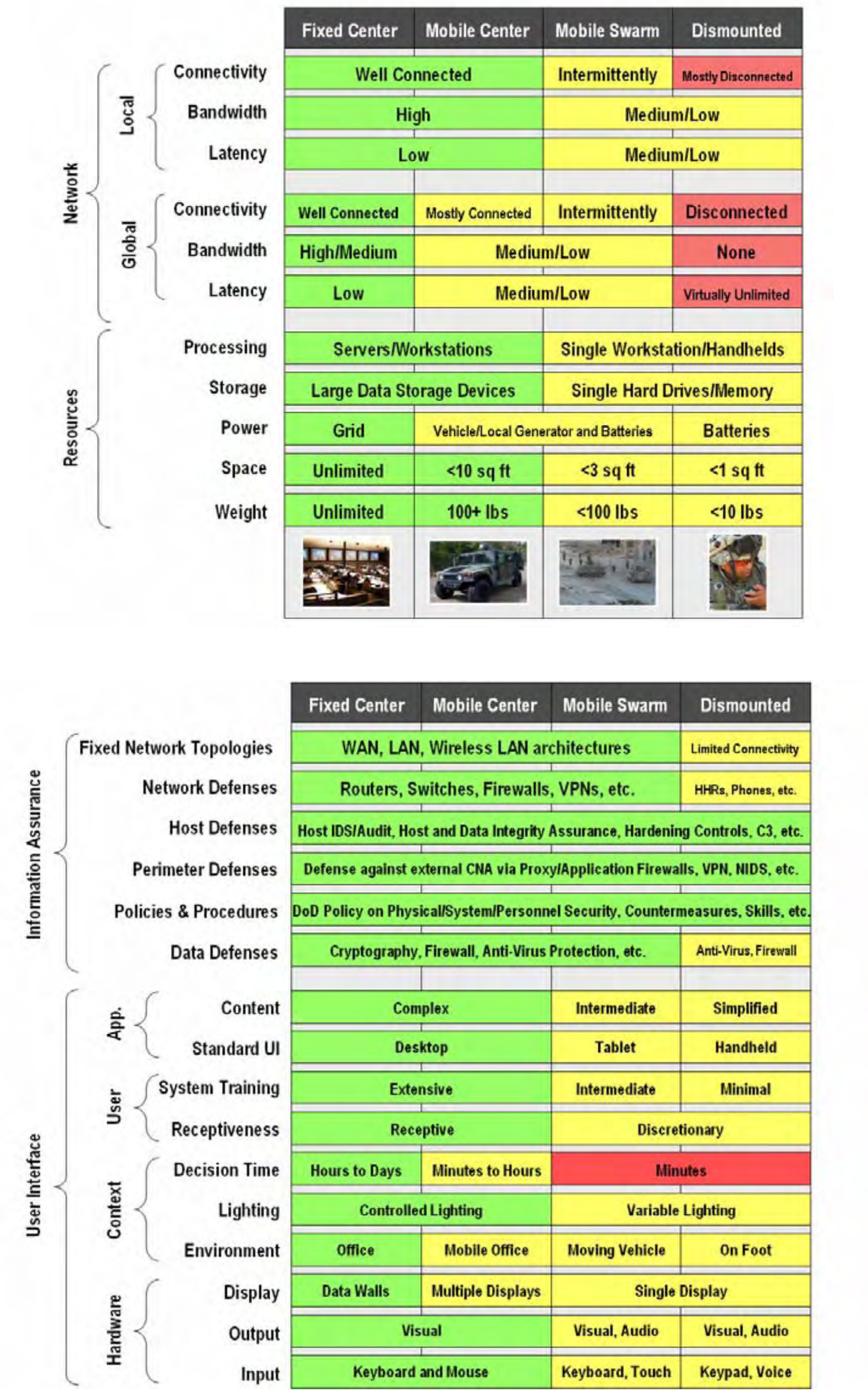
As described in Volume 1, the Tactical Edge Characterization Framework is comprised of a common vocabulary for tactical environments, a set of design patterns, infrastructure requirements derived from design patterns, and reference implementations geared toward demonstrating the value of applying common framework to delivery of service-based capabilities to disadvantaged users.

The Characterization Framework is aimed at defining the technical constraints in providing service-based capabilities the tactical edge. To define the framework and help identify design patterns, we began by gathering use cases, identifying common characteristics for the various edge support and tactical edge environments, and then focusing on defining a common terminology to describe those environments. Four environments were identified: fixed center, mobile center, mobile swarm, and dismounted. Each environment was then characterized by four dimensions:

1. The availability and robustness of a network
2. The availability of resources to execute a particular function
3. Information assurance
4. User interface.

These four dimensions were further quantified using a set of attributes and a range of possible values for each attribute. The network dimension was characterized by the attributes: connectivity, bandwidth, and latency, where both latency and bandwidth (i.e., speed and capacity) of the network define the throughput of the network. The resource dimension was characterized by the attributes: processing power, storage capacity, power, total system space, and total system weight. The information assurance dimension was characterized by the attributes: fixed network topologies, network defenses, host defenses, perimeter defenses, policies & procedures, and data defenses. The last dimension, user interface (UI), was characterized by the attributes: content, standard user interface, system training, receptiveness, decision time, lighting, environment, display, output, and input.

Figure 1 illustrates how the four environments of fixed center, mobile center, mobile swarm, and dismounted were characterized for each dimension. Values defined for each attribute appear in cells as detailed below. The classes shown in Figure 1 serve as the representational set of tactical environments for which design patterns can be specified.



<table>
<tr><th></th><th></th><th></th><th>Fixed Center</th><th>Mobile Center</th><th>Mobile Swarm</th><th>Dismounted</th></tr>
<tr><td rowspan="6">Network</td><td rowspan="3">Local</td><td>Connectivity</td><td colspan="2">Well Connected</td><td>Intermittently</td><td>Mostly Disconnected</td></tr>
<tr><td>Bandwidth</td><td colspan="2">High</td><td colspan="2">Medium/Low</td></tr>
<tr><td>Latency</td><td colspan="2">Low</td><td colspan="2">Medium/Low</td></tr>
<tr><td rowspan="3">Global</td><td>Connectivity</td><td>Well Connected</td><td>Mostly Connected</td><td>Intermittently</td><td>Disconnected</td></tr>
<tr><td>Bandwidth</td><td>High/Medium</td><td colspan="2">Medium/Low</td><td>None</td></tr>
<tr><td>Latency</td><td>Low</td><td colspan="2">Medium/Low</td><td>Virtually Unlimited</td></tr>
<tr><td colspan="2" rowspan="5">Resources</td><td>Processing</td><td colspan="2">Servers/Workstations</td><td colspan="2">Single Workstation/Handhelds</td></tr>
<tr><td>Storage</td><td colspan="2">Large Data Storage Devices</td><td colspan="2">Single Hard Drives/Memory</td></tr>
<tr><td>Power</td><td>Grid</td><td colspan="2">Vehicle/Local Generator and Batteries</td><td>Batteries</td></tr>
<tr><td>Space</td><td>Unlimited</td><td><10 sq ft</td><td><3 sq ft</td><td><1 sq ft</td></tr>
<tr><td>Weight</td><td>Unlimited</td><td>100+ lbs</td><td><100 lbs</td><td><10 lbs</td></tr>
</table>

<table>
<tr><th></th><th></th><th></th><th>Fixed Center</th><th>Mobile Center</th><th>Mobile Swarm</th><th>Dismounted</th></tr>
<tr><td colspan="2" rowspan="6">Information Assurance</td><td>Fixed Network Topologies</td><td colspan="3">WAN, LAN, Wireless LAN architectures</td><td>Limited Connectivity</td></tr>
<tr><td>Network Defenses</td><td colspan="3">Routers, Switches, Firewalls, VPNs, etc.</td><td>HHRs, Phones, etc.</td></tr>
<tr><td>Host Defenses</td><td colspan="4">Host IDS/Audit, Host and Data Integrity Assurance, Hardening Controls, C3, etc.</td></tr>
<tr><td>Perimeter Defenses</td><td colspan="4">Defense against external CNA via Proxy/Application Firewalls, VPN, NIDS, etc.</td></tr>
<tr><td>Policies &amp; Procedures</td><td colspan="4">DoD Policy on Physical/System/Personnel Security, Countermeasures, Skills, etc.</td></tr>
<tr><td>Data Defenses</td><td colspan="3">Cryptography, Firewall, Anti-Virus Protection, etc.</td><td>Anti-Virus, Firewall</td></tr>
<tr><td rowspan="10">User Interface</td><td rowspan="2">App.</td><td>Content</td><td colspan="2">Complex</td><td>Intermediate</td><td>Simplified</td></tr>
<tr><td>Standard UI</td><td colspan="2">Desktop</td><td>Tablet</td><td>Handheld</td></tr>
<tr><td rowspan="2">User</td><td>System Training</td><td colspan="2">Extensive</td><td>Intermediate</td><td>Minimal</td></tr>
<tr><td>Receptiveness</td><td colspan="2">Receptive</td><td colspan="2">Discretionary</td></tr>
<tr><td rowspan="3">Context</td><td>Decision Time</td><td>Hours to Days</td><td>Minutes to Hours</td><td colspan="2">Minutes</td></tr>
<tr><td>Lighting</td><td colspan="2">Controlled Lighting</td><td colspan="2">Variable Lighting</td></tr>
<tr><td>Environment</td><td>Office</td><td>Mobile Office</td><td>Moving Vehicle</td><td>On Foot</td></tr>
<tr><td rowspan="3">Hardware</td><td>Display</td><td>Data Walls</td><td>Multiple Displays</td><td colspan="2">Single Display</td></tr>
<tr><td>Output</td><td colspan="2">Visual</td><td>Visual, Audio</td><td>Visual, Audio</td></tr>
<tr><td>Input</td><td colspan="2">Keyboard and Mouse</td><td>Keyboard, Touch</td><td>Keypad, Voice</td></tr>
</table>

Figure 1. Summary of Tactical Edge Characterization

3 Design Patterns

Design patterns occur in many different disciplines. The concept of design patterns is summarized by the architect Christian Alexander as a manner to “Describe a problem which occurs over and over again in our environment, and then describe the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice.” [1] The computer science discipline later adopted Alexander’s idea and summarized design patterns as “a description of communicating objects and classes that are customized to solve a general design problem in a particular context.” [2]

For the tactical edge, we propose that the problem and the context can be described using the attributes from the Characterization Framework described in Volume 1 and summarized above. As defined here, each pattern starts with a unique name and description, followed by the context and problem statement, a description of the pattern’s Trade-offs, an illustration, and an application example. We note that no document could contain a complete set of possible design patterns, and any such attempt would certainly end in failure (after a exhausting search effort). Rather, this paper contains a representative set of patterns for each of the main dimensions of the edge environment and a template that can be used to capture, enable reuse, and refine evolving patterns. Over time, some of the patterns we recognize as useful today may become unnecessary or obsolete, replaced by more efficient approaches enabled by advancing technology, infrastructure improvements, or changes in operational constraints.

3.1 Design Pattern Definition Template

Each pattern identified for use at the edge was defined using a standard template. This template was intended to provide the minimally complete set of information necessary to support the use of the design pattern by information services developers (and the supporting test and evaluation communities, users, etc.). The template consists of the following elements:

- **Name:** A unique identifier for the design pattern.
- **Description:** A brief description of the design pattern, including the purpose of the pattern and overall approach to addressing the technical problems.
- **Context:** A description of where the solution can be used. For this paper, context consists of mappings to tactical environments that use the common vocabulary for the design pattern. Typically, context is specified as exchanges between the tactical environments of fixed center, mobile center, mobile swarm, and dismounted. At times, the context can be within a tactical environment.
- **Problem:** The technical problem(s) that this design pattern solves, using attributes defined by the common vocabulary. The attributes were summarized in the previous section. Examples of these attributes are: Internet Protocol (IP) availability, connectivity, bandwidth, and latency.
- **Trade-offs:** Benefits and limitations of the design pattern.

- **Illustration:** An illustration of the design pattern.
- **Solution Example:** Example of solutions employing this design pattern, either commercial or government.

Some patterns may be most useful when applied in composition with other patterns, in order to provide an adequate solution for a particular problem. In such cases, the problem definition is divided into a set of sub-problems and specific design patterns are defined, where applicable, for each sub-problem. In this manner, the composition of the identified patterns provides the solution to the overall problem.

3.2 Types of Design Patterns

1. The first set consists of patterns that alleviate resource and network constraints:
    - Messaging Bridge (MB) [3]: The data source service sends the message to the MB Source and the MB Source performs the actual transmission. Similarly, there is an MB Destination that receives a message and delivers it to the destination.
    - Notification [4]: Availability of new content is broadcast to all interested consumers that subscribed to it.
    - Personalized Delivery [4]: Intermediate service provides a customized data and interface to the end service requestor, based on a user profile.
2. The second set consists of patterns that alleviate resource constraints:
    - Reliable Asynchronous Messaging [4]: Messages produced by the service provider in response to the request are queued until the service requestor asks for the messages.
    - Store and Forward [5]: Network of nodes receive data, store data until connectivity is re-established, then forward data to other nodes.
    - Caching [6]: Replicate and synchronize data within local data stores to facilitate data retrieval.
    - Compression [7]: Compress the data for optimal use of bandwidth during transmission.
    - Publish and Subscribe [4]: Data consumers register subscriptions. When the data is available, it is automatically published by data providers to consumers.
3. The third set consists of patterns used for IA purposes. The following example design patterns have been excerpted from [8]
    - Simple Firewall Configuration: A firewall inspects and filters incoming and outgoing network traffic based on the protocol, port number, and the type of application service to be accessed or type of application service requesting access.
    - Demilitarized Zone: A Demilitarized Zone (DMZ) permits different protection roles to systems on the DMZ than internal systems. Typically, systems on the DMZ require

less protection than internal systems, as they can be accessed from the World Wide Web.

- Multilevel Security: In some environments, data and documents may have critical value and their disclosure could result in serious problems. This pattern describes how to categorize sensitive information and prevent its disclosure. It discusses how to assign classifications (clearances) to users and classifications (sensitivity levels) to data, and how to separate different organizational units into categories. Access of users to data is based on policies, while changes to the classifications are performed by trusted processes that are allowed to violate the policies.

4. The fourth set consists of patterns for the design of the UI. Examples are:
   - Canned Messages**:** Users can choose from a list of predetermined messages, rather than having to enter text.
   - Flattened Navigation: Users can select an option with a single click, rather than navigating through a series of cascading menus.

This page intentionally left blank.

4 Reference Implementation

There is a long list of design patterns available in the literature [2,3,9,10,11]. However, our challenge was to identify particular design patterns that alleviate the technical constraints associated with the disadvantaged tactical edge user. The purpose is to demonstrate the use of the common vocabulary to describe design patterns for a particular use case. We used the Common Vocabulary from our Characterization Framework to describe the design patterns identified for the tactical edge use case. In this section, we apply the identified design patterns to a tactical use case example and describe a composite design pattern in terms of its name and description, context, problem, proposed solution, Trade-offs, and some illustrations of the use case solution.

4.1 Data Dissemination at the Tactical Edge

Name: Data Dissemination (Pattern Composition)

Description: Move messages and data within constrained tactical edge environments (i.e., from the mobile center to the mobile swarm to the dismounted environments).

Use Case Description: A Tactical Operations Center (TOC), classified as a Mobile Center environment, sends various messages, e.g., Operations Orders (OPORDERs) to a multimedia high mobility multipurpose wheeled vehicle (HMMW-V), also known as a humvee or hummer, base station. The HMMW-V, classified as a mobile swarm environment, is employed to support dismounted squad operations using heterogeneous (multiple disparate) communications media. The HMMW-V acts as a Combat Vehicle Heterogeneous Cell site (CVHC). Dismounts are typically equipped with small commercial off-the-shelf (COTS) handheld radios and readily available devices that enable their operations to be conducted unconstrained by utilizing the CVHC parked in their vicinity. The important aspect of this is the assumption that dismounts typically operate within line-of-site (LOS) communications range of their supporting CVHC for most operations, which enables en-route coordination and planning. In these circumstances, dismounts may partition from the global network connecting the TOC to the CVHC, while remaining connected to the vehicle on another local network. Dismounts can still be provided with messages and order updates delivered directly to their handhelds. Such order updates may include a picture of a known terrorist or his last reported location. Similarly, a dismount may send a picture of a detained person up to the vehicle, which in turn may relay it to the TOC for identification against a watch list database.

Context: Mobile Center – Mobile Swarm –Dismounted

Problem:

- **Connectivity:** Well Connected – Intermittently Connected – Mostly Disconnected
- **Bandwidth:** High – Medium/Low
- **Processing Power:** Servers/Multiple Workstations – Single Workstation/ Handhelds

- **Storage:** Large Data Storage Devices – Single Hard Drives/Memory
- **Display:** Multiple Displays – Single Display

Solution: A Data Dissemination Service (DDS) implementing a Publish and Subscribe mechanism is employed with the Messaging Bridge design patterns to alleviate problems associated with the tactical edge:

Trade-offs:

> **Benefit**s: DDS uses standard Web protocols to transfer the content. As such, DDS requires high throughput networks and full connectivity for its operations. Since this is not the available for the use case described above, we identified and employed the Messaging Bridge design patterns to alleviate the technical constraints imposed by the tactical edge environments. The Messaging Bridge pattern shields the DDS from the intermittent connectivity, high latency, and the low bandwidth of the tactical edge. The Messaging Bridge pattern accomplishes this by implementing connection pooling to deal with the high latency of the tactical edge. The Messaging Bridge pattern also implements queuing and compression to address intermittent connectivity and low bandwidth. The Messaging Bridge can also cache data, provide Quality of Service (QoS), and more optimally continue with sending a message from the point of the network disruption in the case of a large message.
>
> **Limitations**: The addition of the Messaging Bridge to the DDS results in the significant increase of the DDS footprint, making it more difficult to deploy DDS in storage and processing challenged environments.

Illustration: Figure 2 illustrates the DDS with Messaging Bridge architecture. In a typical publish and subscribe paradigm, content providers publish their content to the data dissemination server, while consumers receive certain content by first registering their interest with the data dissemination server. With the insertion of the Messaging Bridge, the DDS server disseminates content through the DDS client proxy, which is typically co-located with it. Similarly, the DDS subscriber registers its interest in a particular content through the DDS server proxy, which is co-located with it. Both DDS server and client Messaging Bridge are implemented as Mule [12] servers. The Mule Enterprise Service Bus (ESB) provides support for connection reestablishment after a dropped connection, and facilities for compression and queuing.

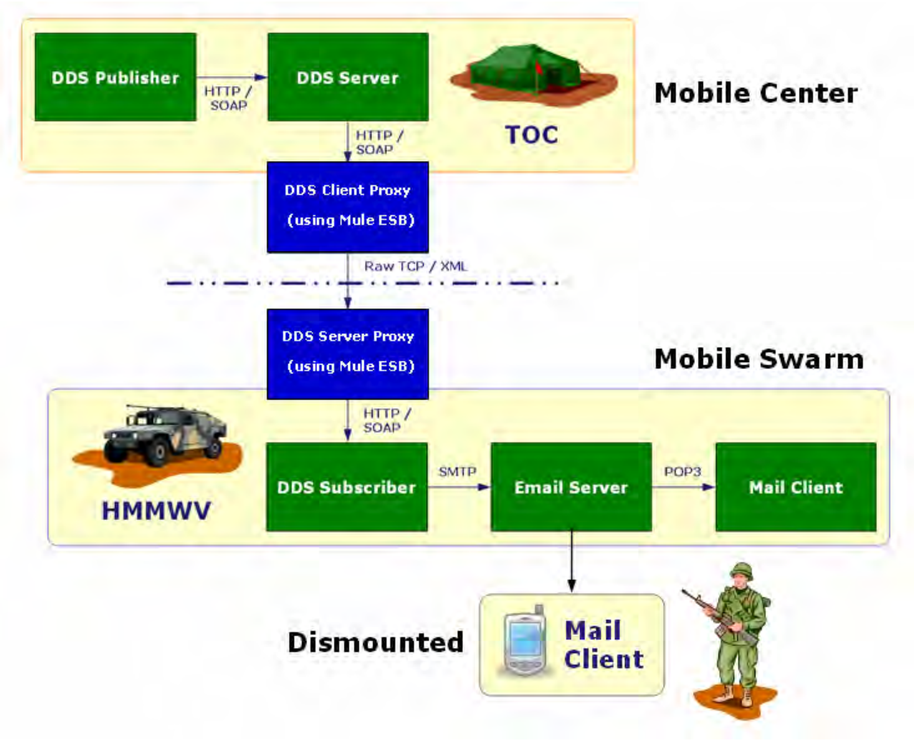


Figure 2. Illustration of DDS with Messaging Bridge

4.2 Infrastructure Requirements for Design Patterns

For the use case described above, there are a number of infrastructure components that are required to implement the solution. These components include information exchange infrastructure to work across constrained networks, as well as client-side applications supporting offline mode.

To handle network disruptions, a proxy service is an important infrastructure component. In our solution, the proxy's implementation is facilitated by the open-source Mule ESB [12]. This implementation is well suited for message-oriented information exchange. The Mule ESB also provides a platform to integrate additional patterns, such as compression. In some cases, dealing effectively with network disruptions on the tactical edge cannot be solved at the application layer alone, but requires architectures spanning the messaging layer, middleware layer, application server, and browser. An example of such an architecture is the Disruption Tolerant Network [13] being developed at the Defense Advanced Research Projects Agency (DARPA), which makes use of store and forward techniques, routing models, and persistence to overcome disruption in a network.

On the client side, supporting an offline mode for client applications is an important aspect in dealing with intermittent networks. Lately, we have witnessed new developments concerning the interactions between application servers and browsers. In particular, the Google Gears Toolkit and Dojo allow Web developers to program applications to support an offline mode of operations in addition to the online mode. These solutions work by implementing a local server on the browser side where resources are cached. Updates from the server are retrieved when resources are requested. Follow on work may include testing and incorporating these new solutions.

5 Recommendations

Based on the identification of design patterns for the tactical edge using the common vocabulary of this characterization framework, the following recommendations have been formed:

- Adopt the use of design patterns particular to disadvantaged environments and use these design to make investment decisions on infrastructure requirements as part of infrastructure upgrades and resource prioritizations.

In support of developing and fielding information systems and services to address the unique nature of tactical environments, the DoD should leverage the characterization framework in describing tactical environments and system design patterns as part of acquisition processes. This paper proposes an initial set of possible design patterns and infrastructure requirements needed to support tactical operations. The DoD should work with appropriate stakeholders to adopt the use of design patterns, incorporating into acquisition processes and subsequently alleviating restrictions typically found at the tactical edge.

- Engage industry to promote the development of tactical edge solutions and highlight existing implementations suited for the edge.

A subset of the design patterns identified in this paper and the infrastructure required to implement them have been employed in a reference implementation. We recommend this successful use of proven design patterns be leveraged in the implementation of systems at the tactical edge as a feasible, fast, and economical way forward for DoD. To propagate this approach, existing guidance documents should be updated to recommend the approach and describe the reference implementation for others to adopt.

- The existing DoD net-centric guidance documentation (such as NCIDs and NESI) should review the results of this work and consider adoption of the common vocabulary (as described in Volume 1 of this set) and design patterns described here.

Highlighting the design solutions and approaches described here provides an effect manner to reuse proven solutions and enhance interoperability and delivery of information services to edge users. Vehicles such as NCIDs and NESI represent expanding guidance documents where the edge environment could be captured and design guidance recorded for development and test communities.

This page intentionally left blank.

6 Conclusion

In this paper, we proposed a method for capturing design patterns for the tactical edge using the common vocabulary of the characterization framework. We provided a set of design patterns that minimize technical constraints associated with the disadvantaged tactical edge user and derived the infrastructure requirements needed to implement a design pattern. This implementation serves as a reference, geared toward demonstrating the use of the characterization framework and validating the design patterns.

The characterization framework proposed in this paper provides a number of benefits:

- It provides a common vocabulary to describe operational environments such that one can look across multiple use cases and identify commonality in the type of constraints introduced in each use case. Subsequently, this allows for sharing of design patterns and implementation solutions across these use cases.
- The framework provides the basis for a process to assess the readiness of a particular existing system for tactical environments, comparing the implementation against appropriate design patterns and infrastructure requirements for particular classes of environments.
- As the characterization framework is adopted, the framework will increasingly provide value as guidance for the development of new systems for the tactical edge, providing reusable design patterns, and reference implementations as the basis for future implementations.

Next steps for the characterization framework include exercising the framework within additional use cases to validate the classes of environments and associated attributes. Once sufficient validation is achieved, follow-on activities will focus on adoption of this framework by the DoD components.

This page intentionally left blank.

Appendix A Acronyms

AWS	Airborne Web Services
B2B	Business-to-Business
COTS	Common Of-The-Shelf
CSEL	Combat Survivor/Evader Location
CVHC	Combat Vehicle Heterogeneous Cell
DARPA	Defense Advanced Research Projects Agency
DDS	Data Dissemination Services
DKO	Defense Knowledge Online
DMZ	Demiliterized Zone
DoD	Department of Defense
DTNs	Delay Tolerant Networks
ESB	Enterprise Service Bus
FI	Fast Infoset
GID	Global Information Grid
IA	Information Assurance
Infoset	Information Set
IP	Internet Protocol
LOS	Line-Of-Site
MB	Messaging Bridge
NCID	Net-Centric Implementation Documents
NESI	Net-Centric Enterprise Solutions for Interoperability
OASIS	Organization for the Advancement of Structured Information Standards
OPORDERs	Operations Orders
QoS	Quality of Service
RAMP	Reliable Asynchronous Messaging Profile
SAR	Synthetic Aperture Radar
SOA	Service Oriented Architecture
TOC	Tactical Operations Center

UI	User Interface
WS	We Services
WSN	Web Services Notification

Appendix B References

1. Alexander, C., S. Ishikawa, M. Silverstein, 1977, “A Pattern Language: Towns/Buildings/Construction,” New York: Oxford University Press. ISBN 0-19-501919-9.
2. Gamma,E., R. Helm, R. Johnson, and J. Vlissides, 1995, “Design Patterns: Elements of Reusable Object-Oriented Software,” Boston, MA: Addison-Wesley.
3. Hohpe, G., B. Woolfe, 2004, “Enterprise Integration Patterns, Designing, Building, and Deploying Messaging Solutions,” Boston, MA: Addison-Wesley.
4. “ws-Notification, ws-pubsub, & personalized delivery standards,” Contributors: IBM, Akamai Technologies, Computer Associates International, SAP AG, Fujitsu Laboratories of Europe, Globus, Hewlett-Packard, Sonic Software, TIBCO Software, http://www.ibm.com/developerworks/library/specification/ws-notification/, http://www.ibm.com/developerworks/patterns/portal/access-personalized-runtime.html, http://www.ibm.com/developerworks/library/specification/ws-rm/, http://www.ibm.com/developerworks/library/specification/ws-pubsub/,
5. Chappell, D., 2004, “Enterprise Service Bus,” Cambridge, MA: O’Reilly.
6. Srinivasan, H., J. Conallen, E. Lane, “Building SOA Applications With Reusable Assets, Part 4: The Requester-Side Caching Pattern,” http://www-128.ibm.com/developerworks/library/ws-soa-reuse4/index.html
7. ISO/IEC 24824-1, “Information Technology -- Generic applications of ASN.1: Fast Infoset,” March 30, 2007, http://www.iso.org/ Schumacher, M. et al., 2005, “Security Patterns: Integrating Security and Systems Engineering,” Indianapolis, IN: John Wiley & Sons.
8. Schumacher, M. et al., 2005, “Security Patterns: Integrating Security and Systems Engineering,” Indianapolis, IN: John Wiley & Sons.
9. Adams, J., S. Koushik, G. Vasudeva, G. Galambos, 2004, “Patterns for e-business, A Strategy for Reuse,” IBM Corporation, http://www-128.ibm.com/developerworks/patterns/
10. Endrei, M., et al., 2004, “Patterns: Service-Oriented Architectures and Web Services, IBM RedBooks, IBM. http://publib-b.boulder.ibm.com/Redbooks.nsf/RedbookAbstracts/sg246303.html?Open
11. Fowler, M., August 2006, “Writing Software Patterns,” http://www.martinfowler.com/articles/writingPatterns.html
12. Open Source Mule ESB (Enterprise Service Bus) http://mule.codehaus.org/display/MULE/Home
13. DARPA, “Disruption Tolerant Network,” http://www.mitre.org/news/events/tech05/briefings/2184.pdf.
14. NSA, “Information Assurance Component of the GIG Integrated Architecture GIG Integrated Architecture v 1.1,” May 2006

Appendix C Design Patterns

C.1 Resource

Name: Reliable Asynchronous Messaging

Description: A requesting service (service requestor) invokes a query on a data providing service (service provider) and then is free to perform whatever tasks it wishes, independent of the service request. Messages produced by the service provider in response to the request are queued until the service requestor asks for the messages.

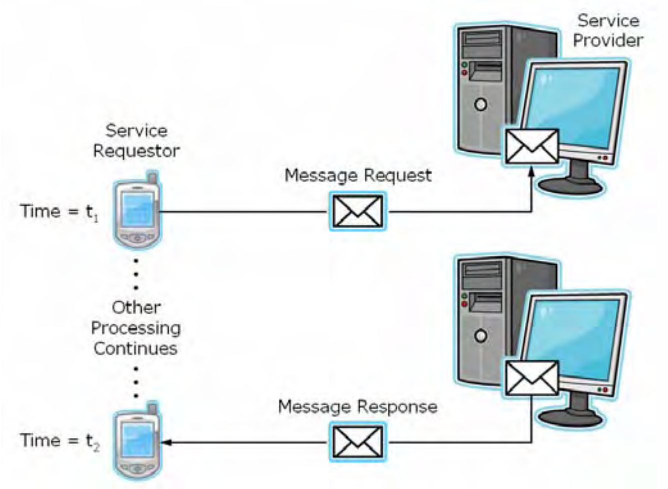
Context: Mobile Center–Mobile Swarm–Dismounted

Problem: Connectivity (Mostly Connected–Intermittently Connected–Mostly Disconnected)

Trade-offs:

> **Benefits:** Independence between data providers and consumers
>
> **Limitations:** Requires additional middleware components.

Illustration:



Solution Example: IBM's Reliable Asynchronous Messaging Profile (RAMP) 1.0 is a profile that enables basic business-to-business (B2B) integration scenarios using Web services technologies.

Name: Store and Forward

Description: A data dissemination technique where data transmission is sent from a data publishing service to a receiving service, but first passes through one or more intermediate services. The intermediate service stores the transmitted message until the receiving service or another intermediate service can be located. It then forwards the transmission to that service and deletes the message locally.

Context: Mobile Center–Mobile Swarm–Dismounted

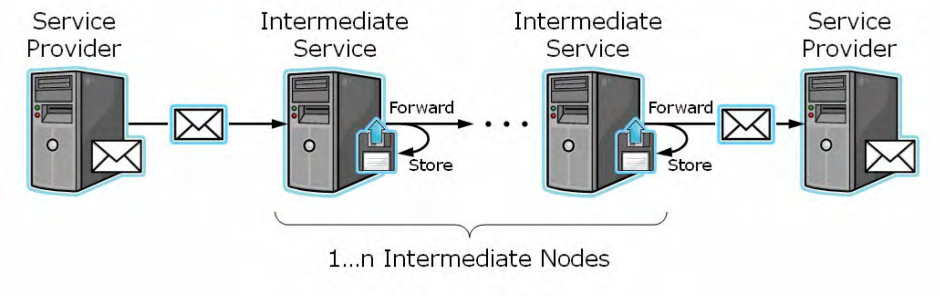
Problem: Connectivity (Mostly Connected–Intermittently Connected–Mostly Disconnected)

Storage: Data Center–Single Hard Drives and Memory

Trade-offs:

> **Benefits:** Provider resources are freed up to provide other services; services to deal with complexities of message exchange (across variable networks) are independent of application services.
>
> **Limitations:** Additional overhead required to exchange messages, which can result in performance degradation.

Illustration:



Solution Example: Disputant/Delay Tolerant Networks (DTNs)

Name: Caching

Description: Data is replicated and stored in multiple locations to support more optimal performance to retrieve the data. Usually data is cached closer to those that most often require the data and synchronization is periodically performed to ensure data is accurately replicated from the original data source and cached copies.

Context: Mobile Center–Mobile Swarm

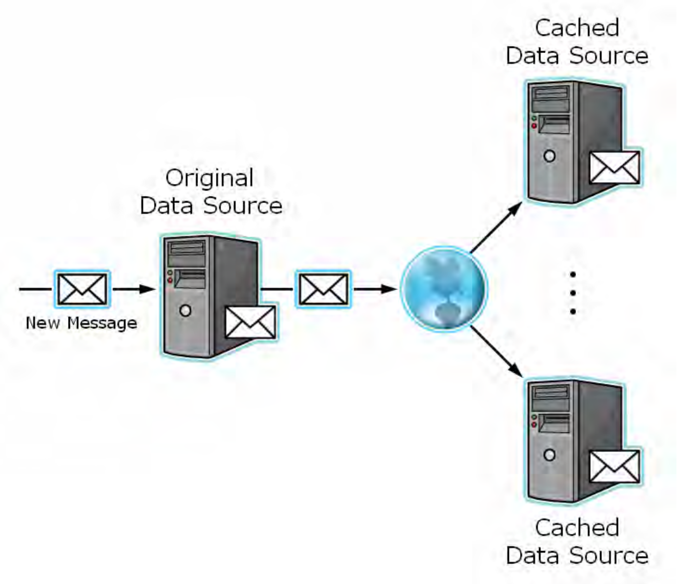
Problem: Connectivity: Intermittently Connected–Mostly Disconnected

Storage: Data Center–Single Hard Drives and Memory

Trade-offs:

Benefits: More optimal performance to access data
Limitations: Synchronization of replicated data sources can be difficult for tactical edge networks, possibly resulting in stale data.

Illustration:



Solution Example: Akamai (http://www.akamai.com/)

Name: Compression

Description: Data is compressed to reduce document size.

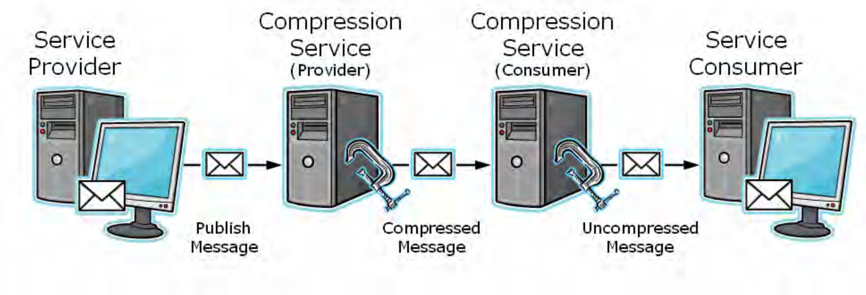
Context: Mobile Center–Mobile Swarm

Problem: Connectivity: Medium Bandwidth–Low Bandwidth

Trade-offs:

> **Benefits:** Reduced bandwidth requirements
> **Limitations:** Data needs to be “uncompressed’ at destination before it can be used, which increases processing overhead.

Illustration:



Solution Example: Sun’s Fast Infoset (FI) is an open, standards-based binary format for the efficient interchange of XML that is based on the XML Information Set (Infoset) to boost parsing speed and reduce document size.

Name: Publish and Subscribe

Description: Publisher service broadcasts data by different topics to those that subscribe to the particular topic area. Publisher service makes information categorized by different topics available to registered subscriber services. Subscribers can choose which topics they want to register for by interacting directly with the publisher or by communicating with a separate broker service. When a new piece of information on a given topic becomes available, a publisher broadcasts this information to all those services that have subscribed to that topic. Alternatively, a broker service can be used to perform the broadcast on the publisher's behalf.

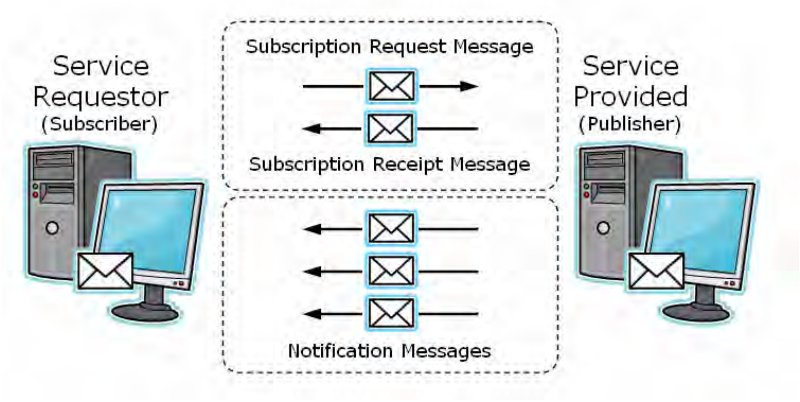
Context: Mobile Center–Mobile Swarm–Dismounted

Problem: Connectivity (Mostly Connected–Intermittently Connected–Mostly Disconnected)

Trade-offs:

Benefits: Decoupling of the publisher from the subscriber, allowing each to act independently and without knowledge of each other.
Limitations: Additional overhead required to exchange messages, which can result in performance degradation.

Illustration:



Solution Example: Data Dissemination Service (DDS)

C.2 Resource and Network

Name: Messaging Bridge (MB)

Description: When a data source publishes data via a service, the actual transmission of the message is performed by a Message Bridge. The data source service sends the message to the MB Source and the MB Source performs the actual transmission. Similarly, there is an MB Destination that receives the message from the MB Source and delivers the message to the message destination service.

Context: Mobile Center–Mobile Swarm

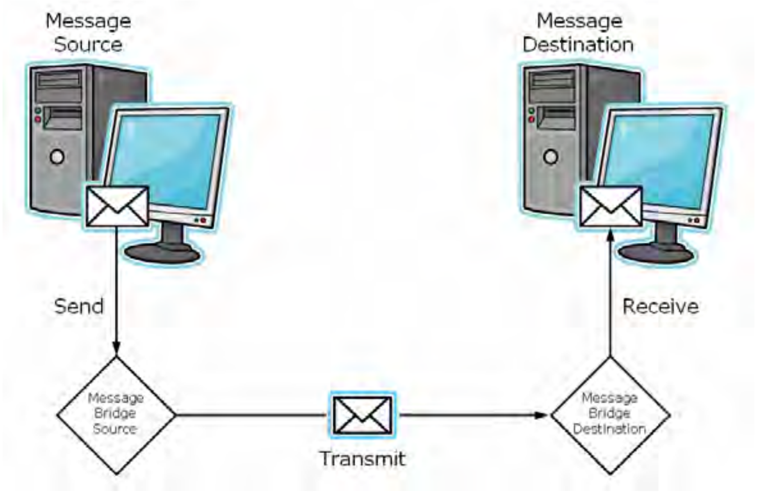
Problem: Connectivity (Intermittently Connected–Mostly Disconnected), Bandwidth (Medium–Low), Processing Power (Single Workstations–Handheld)

Trade-offs:

Benefits: A reliable interface for application services to invoke for message exchange, services to deal with complexities of message exchange (across variable networks) are independent of application services.

Limitations: Additional overhead required to exchange messages, which can result in performance degradation.

Illustration:



Solution Example:

We Services (WS)-ReliableMessaging protocol (a standard defined by IBM, Microsoft, BEA Systems and Tibco)

WS-Reliability specification (a standard defined by Oracle, Sun Microsystems, Hitachi, Fujitsu, NEC and Sonic Software). Submitted to the Organization for the Advancement of Structured Information Standards (OASIS).

Name: Notification

Description: Support for broadcast communication. Unlike an ordinary request, the notification that a provider sends need not specify its consumer. The notification is broadcast automatically to all interested consumers that subscribed to it. The provider does not care how many interested consumers exist. Its only responsibility is to notify its subscribers. This gives you the freedom to add and remove subscribers at any time. It is up to the subscriber to handle or ignore a notification.

Context: Mobile Swarm–Dismounted (store on server and email link)

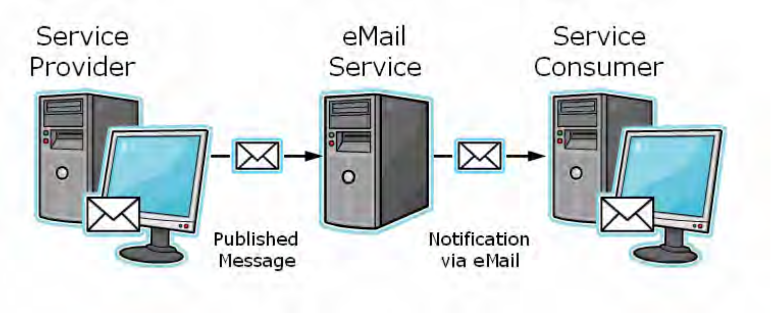
Problem: Connectivity (Intermittently Connected–Mostly Disconnected)

Storage: Single Hard Drives and Memory

Trade-offs:

> **Benefits:** Independence between data providers and consumers. There is also a storage benefit, since data is stored at email server and an email message with a URL link is sent as a notification.
> **Limitations:** Requires additional middleware components.

Illustration:



Solution Example: WS-Base Notification, WS-Topics, and WS-Brokered Notification are three Web Services Notification (WSN) specification documents that are part of the OASIS WSN family of specifications. They define a standard interoperable protocol through which Web services can disseminate events.

Name: Personalized Delivery

Description: Intermediate service provides a customized data and interface to the end service requestor, based on a user profile.

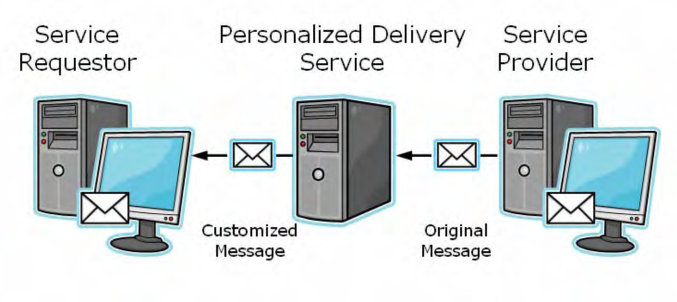
Context: Mobile Swarm–Dismounted

Problem: Connectivity (Intermittently Connected–Mostly Disconnected), Bandwidth (Medium–Low), Processing Power (Single Workstations–Handheld)

Trade-offs:

> **Benefits:** Users have the ability to receive data optimized to meet their operational needs and environment constraints.
> **Limitations:** Additional overhead required to exchange messages, which can result in performance degradation.

Illustration:



Solution Example: Airborne Web Services (AWS): Thumbnail Synthetic Aperture Radar (SAR) Imagery is delivered to disadvantaged users, while higher resolution imagery is made available to users with greater bandwidth and resources.

C.3 Information Assurance

Name: Simple Firewall Configuration

Description: A proxy-based firewall inspects and filters incoming and outgoing network traffic based on the type of application service to be accessed or type of application service requesting access. This pattern interposes a proxy between the request and the access, and applies controls through this proxy.

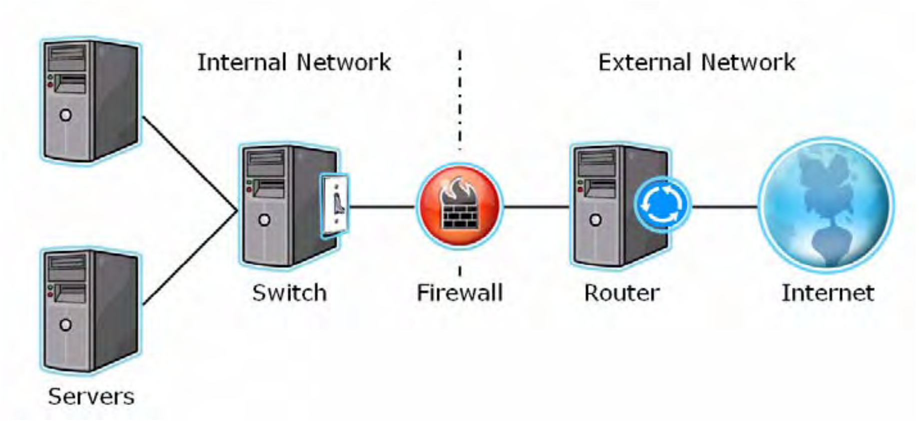
Context: Fixed Center–Mobile Center–Mobile Swarm–Dismounted

Problem: Perimeter Defenses

Trade-offs:

> **Benefits:** Increased security to control attacks on specific layers of the network.
> **Limitations:** Increased complexity of the network design, decreased network speed.

Illustration:



Solution Example: U.S. .mil Domain—NIPRNET

Name: Demilitarized Zone

Description: A Demilitarized Zone (DMZ) separates the business functionality and information from the Web servers that deliver it, and places the Web servers in a secure area.

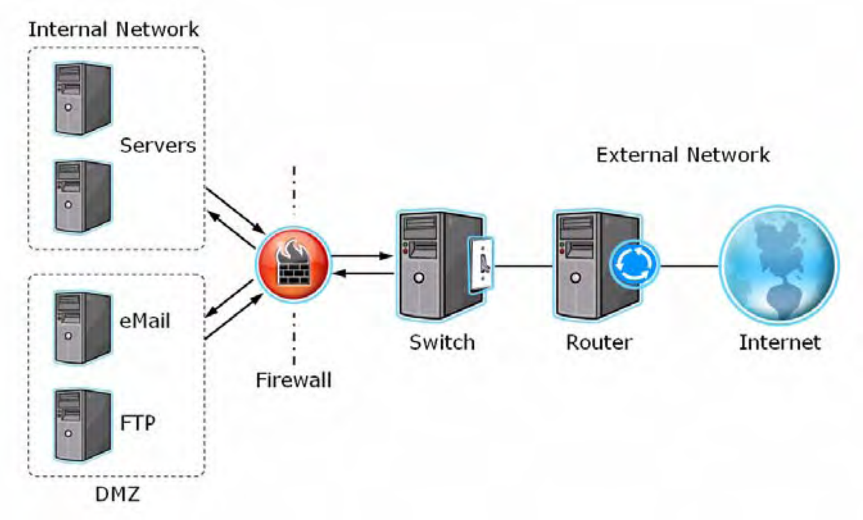
Context: Fixed Center–Mobile Center–Mobile Swarm.

Problem: Fixed Network Topologies, Network Defenses.

Trade-offs:

Benefits: Reduces the 'surface area' of the system that is open to attack.
Limitations: increased complexity of the network design.

Illustration:



Solution Example: Air Force Portal, Defense Knowledge Online (DKO).

Name: Multilevel Security

Description: In some environments, data and documents may have critical value and their disclosure could result in serious problems. This pattern describes how to categorize sensitive information and prevent its disclosure. It discusses how to assign classifications (clearances) to users and classifications (sensitivity levels) to data, and to separate different organizational units into categories. Access of users to data is based on policies, while changes to the classifications are performed by trusted processes that are allowed to violate the policies.

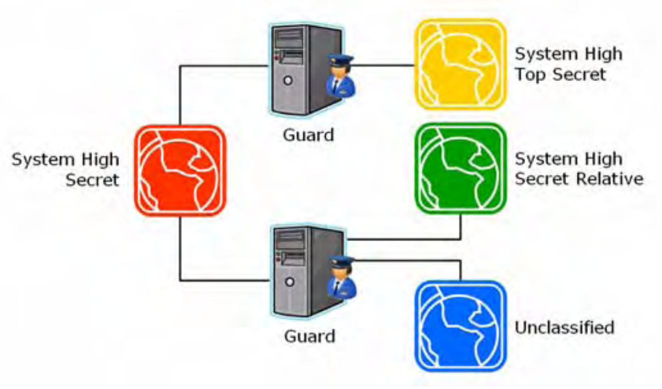
Context: Fixed Center–Mobile Center–Mobile Swarm–Dismounted

Problem: Network Defenses and Host Defenses

Trade-offs:

> **Benefits:** Enable protected information exchanges across classification levels.
> **Limitations:** Increased network design complexity.

Illustration:



Solution Example: Global Information Grid (GIG) Integrated Architecture v1.1, Transactional Information Protection [14]

C.4 User Interface

Name: Canned Messages

Description: Users choose from a list of predetermined messages, rather than entering text.

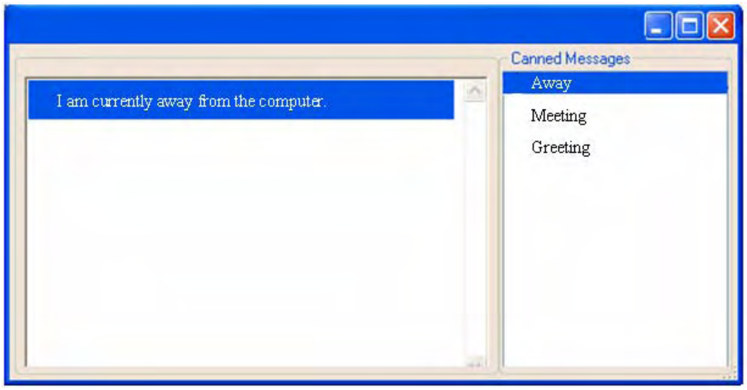
Context: Mobile Swarm, Dismounted

Problem: System Training (Intermediate), Minimal Decision Time (Minutes) Input (Keyboard/Touch Screens, Keypad), Power (Vehicle Generator, Batteries)

Trade-offs:

> **Benefits:** Minimal system training is necessary to begin using the application because the user need not know how to enter text. Choosing from a list of options is faster for a novice user than entering text with unfamiliar devices. Faster navigation reduces the drain on the battery.
> **Limitations:** The input provided by the user will be more general. If many specific choices are required, then very long option lists would be needed.

Illustration:



I am currently away from the computer.

Solution Example: Combat Survivor/Evader Location (CSEL) provides canned messages such as “Capture is imminent.”

Name: Flattened Navigation

Description: Users can select an option with a single click, rather than navigating through a series of cascading menus.

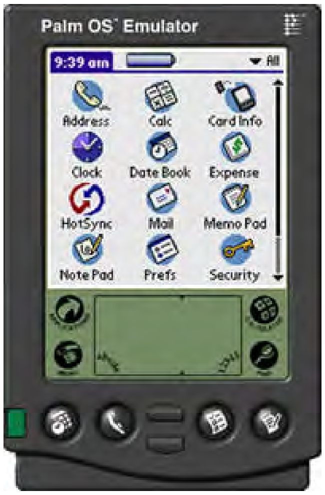
Context: Mobile Swarm, Dismounted

Problem: Decision Time (Minutes), Input (Keyboard/Touch Screens, Keypad), Power (Vehicle Generator, Batteries)

Trade-offs:

> **Benefits:** It only takes one click to start an application. Therefore, it works well when time is short. Cascading menus, which are hard to use with touch screens and on handheld devices, are not necessary. Battery consumption is reduced since user interaction and navigation occurs more quickly.
> **Limitations:** Only a limited number of options can be shown on the main screen.

Illustration:



Solution Example: Palm interface provides large icons on the start screen for each application.