# A Roadmap for Developing Architectures in a Net-Centric World

**John V. Tieso**
Silver Bullet Solutions Inc.
2121 Crystal Drive, Suite 708
Arlington VA 22202
703-892-6062

**David McDaniel**
Silver Bullet Solutions Inc.
2121 Crystal Drive, Suite 708
Arlington VA 22202
703-892-6062

## Abstract

Architectures are beginning to impact on the process that should accompany data collection and dissemination, more needs to be done to ensure that valid, useful data reaches the right people at the right time, enabling them to take the right action in the emerging Net-Centric Environment. This paper discusses architectural approaches that will facilitate required levels of information transfer and utilization.

Information transfer is the critical ingredient in Net-Centric Transformation. Interoperability, integration, and convergence all rely on the availability of valid, current, and confirmable data that can be intelligently 'pulled' as required to satisfy some aspect of a Command, Control, Communications, Computers, Intelligence, Security or Reconnaissance (C4ISR) requirement.

A *roadmap* defines how the architecture is used in creating new systems, systems-of-systems and applications that support net-centricity. The authors propose a method for creating a roadmap to facilitate and focus architecture creation efforts that will maximize their usefulness in development of the Net-Centric Environment. Underlying the paper is a discussion of how two established tools the authors had roles in creating, the *DoD Architecture Framework, version 1.0, (DoDAF 1.0)* and the *Core Architecture Data Model (CADM),* provide the structure needed for creating architectures under the roadmap.

## The Challenge of Change

**Architecture:** The structure of components, their interrelationships, and the principles and guidelines governing their design and evolution over time.

The evolution of technology as a driver for change has been both powerful and dramatic over the last twenty years. However, the power and drama are often associated with results that were not successful, or, if apparently successful at the time, did not achieve long-lasting change. Part of this result is due to the natural tendency of people to fall back into familiar patterns when there is little pressure to use the products of change. Far too often, however, we suggest that the results are really the product of failed planning, and the lack of understanding in how the proposed change 'fits' logically into its larger surroundings.

In past times, the large percentage of failure in producing needed change was not as important as it is in the present. After all, the change process itself took a long time to accomplish, particularly in a governmental environment. Within DoD, that process is even more extended because changes designed at the OSD level still had to be translated and incorporated into both Joint and Service-level Policy & Doctrine before actual change occurred in daily practice. The process of change was extensive and included elements of doctrine, organizational change, training requirements, materiel needs, leadership & education changes, personnel structure change and facilities requirements.[1]

In most cases, where system developments were identified as the means to provide technologic enhancement of changes processes, the execution of change was delayed even further, and often to a five-to-eight year cycle to enable development of the technology required.

In other eras where more traditional warfare circumstances existed, the delay in providing new capabilities and supporting systems was an acceptable risk since there were time-honored and proven methods for waging conventional war. However, the days of waging a conventional war are gradually moving past and giving way to situations where non-traditional adversaries (i.e. terrorists, religious fanatics, etc) are pursuing a different type of non-conventional conflict that does not rise to the level of war. Rather, as we are seeing in Iraq and Afghanistan, along with several nations on the continent of Africa, we see what is being described as Operations Other Than War (OOTW) that require a wholly different approach to both planning and operations.

In these circumstances, information in the form of intelligence and capacity, along with capabilities of an anticipated opponent must be known quickly, responses developed, and action taken—sometimes in days and hours rather than months and even years. The challenge to DoD, and to any other nation similarly faced is to develop significant changes in the doctrine of conducting such operations; assembling the necessary doctrinal, organizational, training, materiel, personnel and leadership, facilities (DOTMLPF) resources to effect those change; apply appropriate technology that can facilitate and enhance the response; and execute a robust, effective response that succeeds in eliminating the threat.

The past ten years have seen the rise of information-based architectures as one means of organizing a suitable response to these new types of threats. DoD created the Command, Control, Communications, Computers, Intelligence, Security and Reconnaissance (C4ISR) Architecture Framework in 1996[2] as its means to focus energies toward the resources (i.e. DOTMLPF Resources) most critical to success on the battlefield—whatever forms the battlefield took in a crisis. The DoD Architecture Framework (DODAF), (Version 1.0), replaced the C4ISR Architecture Framework in 2003. Both Frameworks documents describe three views of any process:

- Operational – i.e. What action(s) being performed and by whom
- Systems –i.e. What technology supporting actions being performed
- Technical—i.e. What standards are being employed; what sources of information are available for use

Creation of these views of a process (Defined as an action being executed to produce a desired result) encourages focus and scoping through a rigorous procedure of defining an activity (either actual 'baseline', or desired 'to-be') and all of its supporting resources, to include information that is passed through the activity. We describe these architectural views in more detail below.

A second critical element in defining the architecture space was the development of a common method for collecting, describing, and storing the architecture data. This is the data, such as events, activities, system functions, systems, nodes, facilities, etc, that is collected during creation of the architecture. DoD satisfied this requirement through the creation of the Core Architecture Data Model *(CADM), an entity-relationship (ER) model. CADM defines the metadata commonly used in architecture development, and can be instantiated in databases through a wide range of available commercial tools. Versions of the CADM have been developed in MS Access©, Oracle©, Sybase©, MySQL© and others.[3]

## Why Architectures?

For many years, the Command and Control (C2) community was satisfied with models and simulations of varying types that exercised scenarios for possible execution in a wide range of potential actions. Modeling and simulation allowed software-based testing that approximated future systems alongside present systems in a pseudo real-time environment. There were also several planning tools, workflow analysis tools, and other kinds of management toolsets that could determine requirements, and facilitate development of management and technology solutions. However, these 'solutions' pre-supposed traditional responses that involved separate, but sometimes coordinated, Army missions, Navy missions, Air missions, etc., and vertically developed (i.e. 'stovepipe') systems supported them.

The Iraqi invasion of Kuwait, followed by the coalition invasion of Iraq, along with a similar invasion in Afghanistan, changed, probably forever, the traditional methods of doing the military's business. These campaigns were the first truly 'joint' efforts where one military service was not simply augmented by support of the other services. Instead, a Joint commander determined his organizational requirements, and operated from a joint-service-staffed headquarters that viewed the entire battle space as one continuous operational area. Aside from long-term problems that had been reported for years involving communications and logistics, there now arose more pressing problems of making major modifications to the fundamental methods and doctrine that drives military operations. Instead of making incremental changes, major transformation of the military community was required, along with the acquisition and fielding of supporting resources and technology that could enhance transformation efforts. In short, the military needed to reinvent itself, and it needed to do it in a way that response to multiple conflicts and problems could be handled successfully without disabling the infrastructure.

Architectures provide principles and methods for understanding the present situation (often called 'the baseline') and defining changes necessary to achieve new and different goals and objectives that need to be translated into new and different processes for executing a response (often called 'the future', or 'to-be' view'). However, architecture is not a solution. Rather, it is the planning, defining and understanding of the needed change that creates a workable solution. These three steps—*planning, defining and understanding*—imply some formal process. Increasingly, these processes are called *roadmaps*. This paper proposes a roadmap for the development of information architectures that will, in turn, facilitate development of systems, systems-of-systems and families-of-systems that respond to C4ISR requirements.

In common terms, a roadmap is used to determine the most direct route from one place to another. An information architect uses a roadmap in much the same way. However, the architect is often concerned with a single problem or need and develops architectures to facilitate a solution of that problem. The identification of potential solutions for problems or requirements in the command and control (C2) community is often much more complex, requiring robust solutions that often encompass several partially related mission areas. That is certainly the case today, as capability-based requirements are becoming the standard for acquisition actions in DoD.

## Evolution of architectures as a discipline

Development of architectures to represent critical C2 requirements emerged in the mid-1990's with the development of the *Command, Control, Communications, Computers, Intelligence, Security and Reconnaissance (C4ISR) Framework, Version 1.0* issued by the Office of the Assistant Secretary of Defense (C3I) in 1997 after nearly two years of development.[4] Eventually, this document was revised as Version 2.0, and eventually evolved to the DOD Architecture Framework, Version 1.0, (The Framework) issued in February 2004.[5] These documents described the detail for both *operational architectures* and *systems architectures* in a number of views that could facilitate both decision-making and development.

Concurrently, DOD had been developing a Command and Control (C2) Data Model (CADM) that would collect and organize critical common data utilized in the C2 community. Within that model was contained a collection of data that was also identified for use in information architectures. This data was organized in an Entity-Relationship-based data model and published by the ASD (C3I) as the Core Architecture Data Model in 1997. The model is a meta-model in that it defines high-level data about data that can then be used to create databases supporting specific architecture developments.

These two documents, one the structure for creating architectures, and the second that data that supports architecture development, gave information architects a set of tools that, if utilized, created architectures that were not just pictures, but rather were capable of analysis and revision over time. Others whose requirements for architectures were similar or who had common requirements could reuse supporting data. Utilizing the CADM to develop databases, and populating the databases with architecture data, (such as operational activities, systems, system functions, operational nodes, physical nodes (i.e. locations), standards, performance measures, and authoritative sources of information) provides, over time, a wealth of common data and its definition or value in an organized, searchable format.

Through the efforts of the ASD (NII), a range of tools that can utilize both the Framework and CADM to build, store and reuse architectures are becoming available. To facilitate that effort, versions of CADM have been developed in XML to ease commercial vendor needs for transferability of data to central repositories; and also in Universal Modeling Language (UML) to enable developers in the Object-oriented development platforms to have access to architecture data that is useful to them.

In our view, the importance of a particular architecture lies in the data that it contains in the *views* that are presented graphically or in data form. Architectures, as they have been defined in DOD, consist of three views, Operational, Systems, and Technical. The *operational view* represents the work being performed on a specific process, tactical or business-related, and the locations, facilities, and organization performing the work. The systems view provides an overlay of the technology that supports operations, to include systems, families of systems, systems of systems, hardware, equipment and communications paths. The systems view also includes representations of the locations where technology is being applied in support of operations. The *technical view* provides the authoritative sources that support both operations and systems. These consist of standards, governmental and commercial, sources for terminology (i.e. taxonomies, categories of information, etc.) and other laws, rules, regulations and standard practices.

A Roadmap is a series of steps employed to ensure that data and information needed for complete understanding of a solution to a need or problem is collected, analyzed, and supportive of decisions for development and employment. Creation of such a roadmap for development and utilization of architectures generally follows Figure 1, below.

| Process | Actions |
|---|---|
| Identification | Define a problem or requirement to be improved |
| Planning | Determine methods for solution/Create a Business Case for consideration |
| Analysis | Create a baseline of existing processes and resources<br>Determine how resources can be associated to create solution<br>Create operational and systems views of existing resources<br>Develop views of desired future state |
| Decision-Making | Develop priority for funding and resources<br>Approve budget requirements |
| Funding | Apply approved funding |
| Development | Develop/Test/Evaluate |

**Figure 1 Architecture Roadmap Development and Utilization**

Roadmap development starts with identification of a specific problem, requirement or desired new capability to be developed. Problem identification generally comes from lessons learned that have identified deficiencies in resources or capabilities that could dramatically change the outcome of an activity. Communications bandwidth deficits, interoperability, and doctrinal or operational changes needed, but not currently defined all arose from lessons learned in the actions involving Kuwait, Iraq and Afghanistan.

Lessons learned provide a wealth of information that can provide a very clear picture of specific problems or issues that often are already the subject of a ‘work-around’ in the field, but require more long-term solutions.  Lessons learned often spawn major changes in the overall perspective of National Strategy. Therefore, a need became obvious to joint planners that there had to be an organized and efficient process for translating lessons learned into potential changes in National Strategy, and utilizing these changes to determine specific policy and program changes in the Executive Departments (such as DoD and Homeland Security.)  Within DoD, the identification of needs and requirements is the first step of the Joint Capability and Integration Development System (JCIDS). Figure 2, below, shows the JCIDS process in Detail.

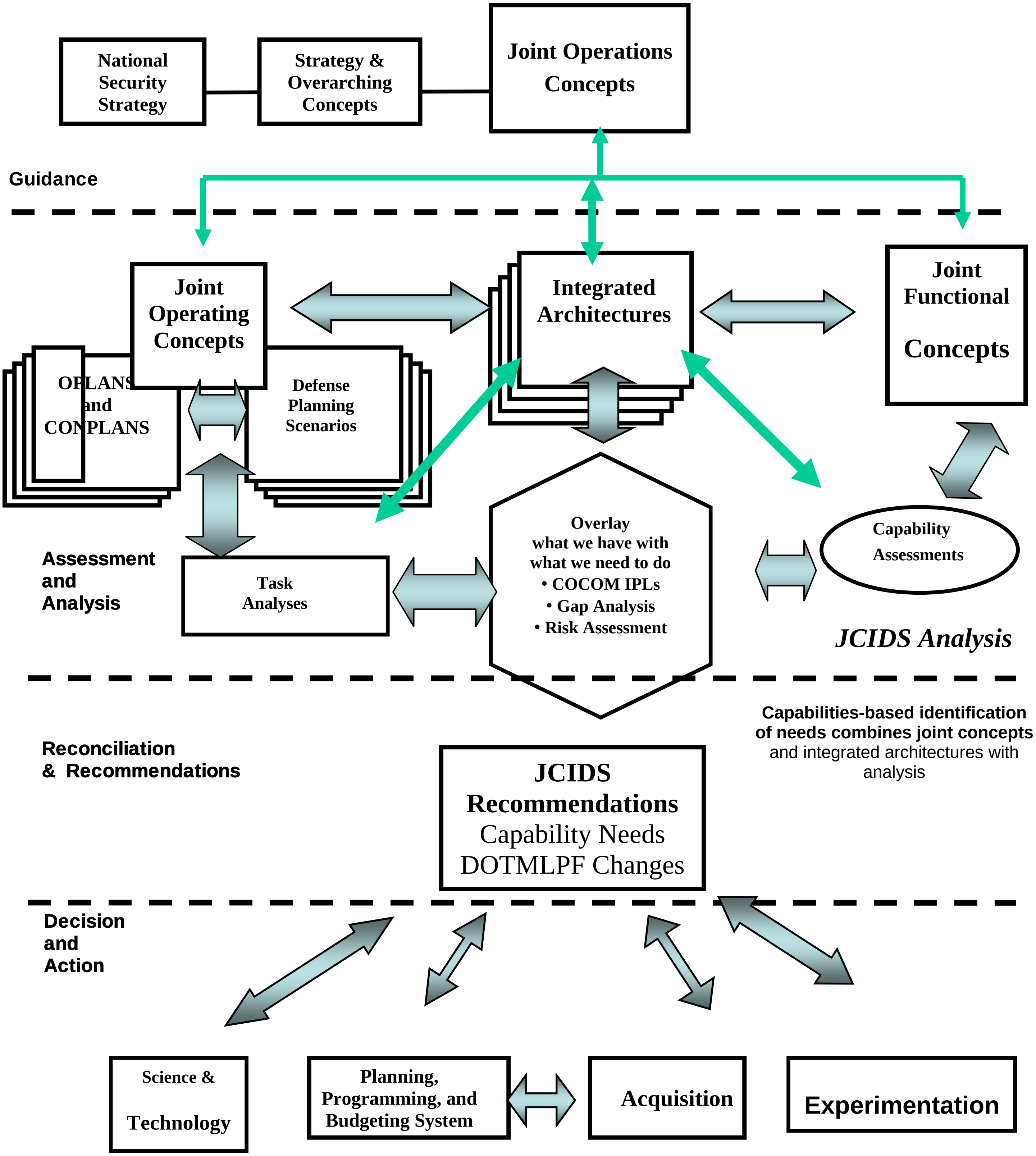
National Security Strategy
Strategy & Overarching Concepts
Joint Operations Concepts
Guidance
Joint Operating Concepts
Integrated Architectures
Joint Functional Concepts
OPLANS and CONPLANS
Defense Planning Scenarios
Overlay what we have with what we need to do
• COCOM IPLs
• Gap Analysis
• Risk Assessment
Capability Assessments
Assessment and Analysis
Task Analyses
JCIDS Analysis
Capabilities-based identification of needs combines joint concepts and integrated architectures with analysis
Reconciliation & Recommendations
JCIDS Recommendations
Capability Needs
DOTMLPF Changes
Decision and Action
Science & Technology
Planning, Programming, and Budgeting System
Acquisition
Experimentation

Figure 3 JCIDS Process

Joint Chiefs of Staff Instruction (CJCSI) 3170.01C, *Joint Capabilities and Integration Development System* defines the JCIDS process as the means to identify, analyze, and determine solutions to complex problems related to National Strategy.[6] That process includes the development of integrated architectures emerging from the operational and functional analyses that occur during the JCIDS review process.

Figure 2 represents the traditional view of the workflow associated with the JCIDS Capability-based needs assessment process. It reflects the politico-military reality that National Security Strategy is first developed in a global/political context and expanded through the use of overarching concepts of execution that meet those strategies. Once the overarching concepts are formalized, it is then possible to develop Joint Operations Concepts at the National level that may involve one, or many, Federal agencies and Departments. The Defense view represents the prospective requirements, configuration, organization, and capabilities of the Armed Forces that may be required to respond to various events and/or contingencies contained in the national Security Strategy, amplified in JOpsC documents. Joint Operations Concepts impact on, and are impacted by *Joint Operating Concepts*, *Integrated Architectures*, and *Joint Functional Concepts,* each of which is discussed briefly below.

Joint Operations Concepts provide a framework for development of Joint Operating Concepts and Joint Functional Concepts, each of which reflect a viewpoint of the overarching Joint Operations Concept in real-time. As these concepts are developed, they provide the necessary information for creation of Integrated Architectures that graphically represent capabilities, and expected execution actions in specific scenarios. Exercising these scenarios to ensure viability, in turn, provides a vehicle for updating the Joint Operations Concepts, and the development of new and improved doctrine supporting joint operations.

Task analyses determine the validity and usability of Joint Operating Concepts (JOCs). Analysis evaluates the potential requirements outlined in a specific scenario against OPLANS and CONPLANS to create a list of required capabilities, and also to evaluate the full range of DOTMLPF requirements that emanate from the scenario. These DOTMLPF requirements are initially categorized as ‘Materiel’ (i.e. systems, families of systems, or systems of systems configured for a specific purpose) or ‘Non-Materiel’ (i.e. doctrine, organizational, training, leadership, education, personnel or facility) requirements.

***Joint Functional Concepts*** are enablers for the joint force commander in composing a force with the appropriate set of capabilities, resources, and troop mix to achieve success in combat or Operations-Other-Than-War (OOTW). They represent the first opportunities to define and/or refine DOTMLPF principles that can be applied consistency to achieve the desired result.

***Capability assessments*** against Joint Functional Concepts determine whether or not the joint force commander can execute a desired course of action with available resources, or if other resources must be found or developed.  Joint Operations Concepts define critical capabilities that must be available in any contingency.

***Integrated Architectures*** describe the activities performed that contribute to joint capabilities.  Integrated Architectures generally reflect either the current state of Joint Operating Concepts and Joint Functional Concepts (i.e. the baseline or current view), or some future state that is desirable (i.e. the 'future' or 'objective' view).  The real power of these architectures lies in the ability to take architecture information on process, systems, and technical capabilities, and relate that information to the task analyses and capability assessments.

As shown in Figure 3, the results of task analyses and capability assessments, reflected against the baseline architecture provides a view of the gaps that presently exist in capabilities, either in whole or in part

In broader terms, The JCID Process provides the means to anticipate future requirements and potential adversaries; apply levels of expectation on how these potential adversaries might commit aggression against American or allied forces; and how a US-led or US-supported coalition force would be required to respond.  Having gained such an understanding of the potential requirements, JCIDs is able to ensure a carefully controlled, spiral evolution of increased or changed capability mixes by requiring that major impacts to the current capability base be carefully planned utilizing a DOTMLPF approach so that development produces systems, families of systems, and systems of systems that execute successfully with no unexpected side effects.

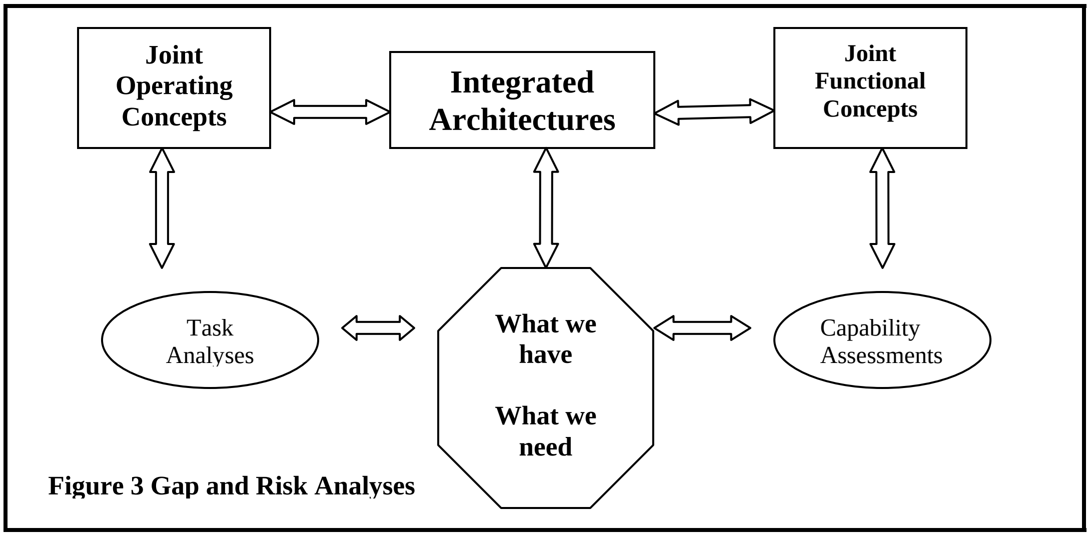

**Figure 3 Gap and Risk Analyses**

Architectures are developed to reflect the evaluation of current capability and present operational, technical, and systems views of the existing baseline. This baseline provides a view of the present levels of capability for selected critical resources.  These resources include the broad range of DOTMLPF resources applied to provide those capabilities.

## Utilizing Architectures

Architectures are created for a specific purpose. We believe that they are particularly useful in the development of views that support the Net-Centric Environment. During the past several years, as the Global Information Grid (GIG) Architecture has evolved[7], and other architectures in the Joint community have been developed, the utilization of CADM-based databases to collect and store this architecture data has greatly simplified the ability of architects to develop net-centric architectures, based on GIG. This has happened because a number of architectures, particularly operational architectures (i.e. those that show events and activities ongoing) have utilized GIG as their starting point for development and described their activities as instances where a high-level set of GIG Activities has been used.

The utilization of higher-level architecture views to develop lower-level views in selected areas provides a level of integration not previously possible.

Net-centricity crosses broad operational and system-based boundaries. It focuses the resources of both the C2 community, and the supporting business infrastructure toward solutions enabled by large-capacity networks with scalable bandwidth to accommodate the volume of transmissions and transactions. In Net-centric Operations, it is these two types of activities –*transmissions* and *transactions*—that almost exclusively utilize bandwidth. Transmissions provide information, graphics and non-directive message traffic. Transactions range the broad spectrum of DOTMLPF providing prescriptive information, such as orders, directions, organizational changes, logistical transactions, etc.

Transmissions and transactions are expected to be two-way communications and can involve both manual and repetitive operations. All of these activities, and the information that passes in these activities, can be modeled, simulated, and architected. Models define the expected order of the activities; simulations provide a graphical representation and analysis of the activities; and architectures provide a representation of the linkages between activities, data transmission, system function requirements, systems and the locations and paths utilized in those activities.

Architectures provide two important services within a net-centric environment. First, architectures map the development of systems, families of systems (FOS), systems of systems (SOS) and the passage of data in transmissions and transactions. Second, architectures provide a graphical representation of activities and organizational alignments. They clarify and simplify understanding and reduce ambiguity through adoption of a common set of descriptions and definitions of data that support the graphical presentations. Reduced ambiguity and clarity are critical aspects of network-based organizations. Architectures help define in a clear way, the preferred methods for collecting, storing, and utilizing data in an unambiguous and efficient manner. They ensure, on the one side that subscribed data is received by the correct subscriber; while published data in the reverse circumstance, is received by the appropriate user of that data.

Architectures lend additional clarity to activities when they are developed in a way that both the developer and user understand them using the same terms. The DoD Architecture Framework, version 1.0, used in conjunction with the GIG Architecture and its Net-centric Operations Reference Model provide a clear, direct route to creating architectures that respond to, and aid in providing solutions to pressing requirements. These architecture efforts can quickly and accurately present a set of views that will highlight changes needed to ensure that critical capabilities are being developed and maintained. Moreover, architectures support the notion that the most direct route to a solution—a roadmap—is often the best solution.

We believe that the key to development of critical capabilities lies in the creation of architectures that factually represent both existing and desired states of readiness and capability. Developing a roadmap of logical steps to ensure that these architectures are developed and maintained goes a long way, in our view to breakthrough improvement in capability in the Department of Defense.

[1] DOTMLPF, the acronym for these categories is now codified in multiple Joint, DoD and Service regulations and procedures as the minimum set of issues that must be addressed in change proposals.

[2] *Command, Control, Communications, Computers, Intelligence, Security & Reconnaissance (C4ISR) Architecture Framework, Version 1.0,* Office of the Assistant Secretary of Defense (Command, Control, Communications and Intelligence), 17 Jun 1996. A Version 2.0 was published in 1997.

[3] *Core Architecture Data Model (CADM), Baseline Version 1.1* is the current official version of the CADM as published by DoD. There have been several versions of this model since 1996 until it was placed under configuration control in 2003.

[4] *Command, Control, Communications, Computers, Intelligence, Security and Reconnaissance (C4ISR) Framework*, 1997, version 1.0, Office of the Assistant Secretary of Defense (C3I), 7 June 1996. Version 2.0 was approved and published on 18 December 1997

[5] *DOD Architecture Framework, Version 1.0,* 8 February, 2004, Office of the Assistant Secretary of Defense (NII)

[6] Chairman Joint Chiefs of Staff Instruction 3170.01C, *Joint Capabilities and Integration Development System (JCIDS)* Organization of the Joint Chiefs of Staff,

[7] *Global Information Grid Baseline Architecture, version 2.0*, 2003, OASD (NII)