INFORMATION PROCESS ARCHITECTURE
A SYSTEMS ARCHITECTING AND ENGINEERING APPROACH FOR THE DEVELOPMENT AND DOCUMENTATION OF COMPLEX INFORMATION AND COMMUNICATIONS SYSTEMS

Peter G. Cook (Hypres, Inc. Elmsford, NY, USA, pgcook@pgcook.);
James Rodenkirch (Diversified Technology, Inc., rodenkirch_llc@msn.com

ABSTRACT

In contemporary society, Information Systems play an increasingly prominent role. Because they operate at the speed of light, and exchange neither mass not energy, Information Systems are replacing physical systems in many application areas. This paper explores the consequences of exponential growth in information, and proposes Information Process Architecture (IPA) to describe information systems. It also describes Views to describe the complex systems that evolve when previously independent information systems merge. An important characteristic of complex systems is that they are never "finished", so the goal becomes to satisfice rather than optimize system operation, accepting the 80% solution.

1. INTRODUCTION

The Software Defined Radio (SDR) Forum was established to further the technology of radios whose operating characteristics were under software control, and thus more flexible than terminals implemented solely in hardware. As SDR technology matured, Cognitive Radio (CR) concepts emerged, involving autonomous reconfiguration of radio frequency (RF) and other system parameters for optimal performance. [1] CR has potential to improve operating efficiency and is a logical extension of SDR concepts – making communicating easier and more efficient. The SDR Forum is now exploring means by which current and future technologies can be utilized to improve spectrum efficiency, provide dynamic spectrum access (DSA), and operate so as to anticipate user needs rather than reacting to user commands.

Radio frequency (RF) wireless links have the significant advantage of untethered operation. Their generic capability is the same as links such as fiber and copper: the ability to make information available at a distant location. But system design must take into account that their mobility is constrained by spectrum congestion, interference, and ambient noise.

In this document we move beyond SDR and CR as technologies, and consider their role as functions to significantly enhance the capabilities of complex systems. We will look at the high-level characteristics of Information Processing Systems (IPA), and at Communication Systems as components of Information Systems. We describe how precise system definition and common representation can improve operating efficiency, and illuminate opportunities for system interaction and integration using communications links between independently developed systems.

2. IT – FROM INFORMATION TELCHNOLOGY (IT) TO INFORMATION TSUNAMI (IT_S)

With widespread acceptance of personal computers and the Internet, a prodigious amount of data is being generated. One estimate [2] is that monthly Internet traffic is 5 to 8 exabytes, 10^{18} bytes. (All the books ever written represent about 10^{13} bytes.) Information in digital representation is rapidly replacing physical formats. Checks are no longer returned in paper form; Wikipedia is a major information resource; payments are made by digital communication without ever being captured in physical form.

Communication systems are the means by which information is made available at distant locations. Unlike material objects, information is not moved; it is replicated in new locations and remains unaltered at the source. We can liken this vast amount of information to a tidal wave; inundation in information, without adequate provisions for coping with it, will almost certainly have unintended consequences. An Information Tsunami (IT_S) is upon us.

One consequence of IT_S is unexpected or unplanned system intersection. For example, mobile terminals, connected over RF links, keep users connected while in motion. Automobiles are sophisticated systems, and ubiquitous for local transportation. But mobile telephone systems and automobile systems have proven as immiscible as oil and water. On-board facilities for voice communication from cars are difficult, data applications are almost unknown, and the promise of highway automation seems far in the future.

IPA is an attempt to address some of these issues. If individuals working in unrelated application domains can independently express their architectures in a common form then the possibility of integrating them may be greatly increased, and the effort to resolve differences when they collide is greatly reduced.

3. IPA OBJECTIVE

The proposed objective statement for an IPA project in the SDRF Cognitive Radio Working Group is:

> From the concepts and tools of relevant frameworks commonly used for developing and documenting the architecture of complex systems, select appropriate functionality for representation of complicated systems, such as a Cognitive Radio, in ways most meaningful to users, administrators and developers of communication systems.

A great deal of work has been done to develop approaches for architecting complex systems at a level high enough to be tractable, yet detailed enough to locate, across domain boundaries, common attributes in independent sub-systems that make up the complex system. IPA intends to draw on experience with approaches such as the Department of Defense Architecture Framework and IEEE 1471.

4. INFORMATION PROCESS ARCHITECTURE

Wireless links are complex system components; such systems do not exchange energy or mass. They rely on wireless communications to move data needed to achieve their objectives.

To better understand these relationships, it is useful to consider independently the three components of the name "Information Process Architecture".

4.1 Information

Data is a symbolic representation of a particular state of affairs, often at a specific point in time. Information processing systems collect data, consolidate it, and put it into context and useful form for use in specific applications.

When data is represented in digital format, the restrictions of Newtonian mechanics are no longer applicable. With electronic systems information can be computed in fractions of a second, and made instantaneously available around the globe.

4.2 Process

Complex systems, as described by IPA, are processes, operations involving data, and are simultaneously whole and part – they are subordinate to higher-level processes and themselves contain sub-processes.[4]

Processes can be categorized as origination, storage, data integration, computation, replication in other locations, and presentation. Thus key considerations to understanding a complex system are an application-independent means of identifying and describing processes and activities accomplished, component process functions required for those activities to be accomplished, and process interconnections. Additionally, this identification and description effort aids in selecting and integrating independent systems through competent trade studies and Analysis of Alternatives efforts.

4.3 Architecture

A complex system, or System of Systems, is the result of integrating myriad, diverse component systems. An architecture is a set of products that define, specify, and explain the structure of a complex system in its "as-is" and "to-be" states. "Systems Architecting" is a verb indicating the actions taken to develop an architecture. Architecture is a *User-oriented* process to define what a system does; Systems Engineering is *Developer-oriented* guidance for system designers.

5. COMPLEX SYSTEM STRUCTURE

A complex system's make-up or domain has, among many attributes and components, an Information Part that is implemented in software and a Hardware Part on which the software executes.

5.1 Information Process Layers

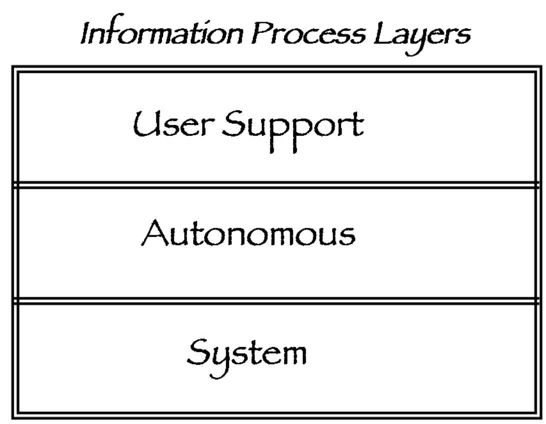

Figure 1.

Figure 1. shows three conceptual Information Process layers. These layers categorize software processes by their function in the system.

5.1.1 User Software Layer

This layer is composed of software operating in direct support of the system user. It provides a user interface with means to display information to the user and receive user directives. It enables command and control of the system, supports system test and maintenance, and provides a development environment for new software.

5.1.2 Autonomous Services and Cognition Software Layer

This layer operates autonomously, in that it does not require direct user intervention. It monitors the state of the system, the system environment, and user activities. When it encounters a system state or user activity that it recognizes, it activates processes to improve operation. For example, if it detects that the quality of the RF link has deteriorated, it may initiate a search for a better channel. If it senses that the user is attempting to accomplish something, it offers to help.

These activities, of course, are characteristics of CR functionality, whether or not that terminology has been used in describing the system. Most information systems have some capability that can be described as "cognitive", and that capability is implemented in this layer. A simple radio could have very little functionality here.
Note that system partitioning is not represented in this figure, so such functionality can be located anywhere in the system, and can operate through communications links. This is a very general concept: software in this layer can make the radio adaptive, smart, or cognitive.

5.1.3 System Software Layer
This layer provides the functionality needed to interface between the system and hardware that implements it. It is the operating system, device drivers, database management system, air interfaces, and memory management system. SDR functionality is in this layer, as is all hardware-specific software.

5.1.4 Layer Interfaces.
The boundaries between these layers are not well defined, a situation that reflects the ambiguity inherent in terminology such as SDR and CR. Some precise definitions are recognized, such as operating system interfaces and application program interfaces. But, while computer operating systems operate autonomously and contain very sophisticated processes, they are not normally considered either cognitive or examples of artificial intelligence.

A key concept of IPA is the ability to group a set of functionality, and encapsulate it to facilitate understanding of system process structure.

5.2 Hardware structure
The geographic and topological structure of the hardware part of a complex system is specific to the application and the user set that it serves. Very little generalization is possible, except to note that the system is never "finished": it will change over time to reflect changes in the user community, dispersion of that community, and to incorporate newly developed technology. In general, user interface equipment will be where users are, processing and storage will concentrate in points of convergence of data to be processed, and where support facilities, such as power lines and cooling are readily provided. Communication links will be dispersed between system equipment venues where minimal traffic volume flows, or in the case of wireless links, where mobility is required.

Performance of all hardware system elements has developed to the extent that increasingly dispersed system structure is feasible. As an example, with evolving RF capacity and increased computational effectiveness, processing capability available at individual user sites now exceeds that formerly available only at central sites, and enables new system structure and enhanced levels of application effectiveness.

6. SYSTEMS ENGINEERING AND ARCHITECTING

Complex systems have several attributes or characteristics that distinguish them from a complicated system (e.g., a Cognitive Radio). A primary characteristic is that the systems were developed and funded independently of the other, and component systems can be used in other complex systems. If dissembled, the component systems can operate independently,

Complex Systems are never fully formed; makeup of systems and activities performed by the users can change or be shifted in terms of sequencing or arrangement and use over a short period of time – hours and even minutes). There is a wide geographical extent: complex systems can exist over hundreds and thousands of miles, there is no exchange of energy or mass amongst the component systems, only information. Complex systems are often characterized as ambiguous and uncertain.

System engineers of today, accustomed to orchestrating and coordinating the design and development of a single, complicated system, face the prospect of designing (i.e., architecting) multiple, integrated complicated systems, a system of systems. Since complex systems are never fully formed, optimizing "the system," is replaced with satisficing – achieving an 80% solution. Since the complex system is never fully formed, striving for a 100% solution is useless. Instead of a well-defined problem, they face an emerging, ever changing problem space. Instead of fixed boundaries, the systems engineer now faces fluid, ever changing boundaries and constraints.

Because the "system" is ever changing, integrating systems that were never designed to interface with each other, and facing an ever-changing buffet of processes to utilize, achieving a satisficed state is a difficult prospect. These activities are the realm of the complex system's users and an engineering challenge that today's system engineers have not often faced. The scope of complex systems engineering is elevated from project or product time frames, with discrete beginning and end, to Enterprise and Capability System Life Cycles. They are now faced with multiple, interacting system life-cycles, amorphous beginnings, and a reliance on history and precursors. Organizations concerned with singular system development were unified and authoritative, while complex systems are steeped in fiefdoms, disparate funding sources, and varying requirements for development. Collaborative networks reflect the complex systems' organizational make up of today.

One can see a distinct separation of foci between the engineering of a complicated system and a complex system – the system engineer's intent is satisfying the builder of the system while the complex system requires decidedly more attention to the user who has the buffet of current systems to choose from to cobble together his/her complex system. In short, Information Process Architecture is a needed discipline.

Architects are concerned with the function of complexity – what it is, not what it does. They bring order out of chaos – architects are not "general engineers" but specialists in complexity, uncertainty and ambiguity reduction. They reduce those three evils to simplify the problem to an extent that engineering analysis can be performed on the remaining variables. They work closely with the client to develop a feasible plan

System(s) engineers focus on form, deriving subsystems and components by hierarchical decomposition; those subsystems then are designed and built by engineering specialists. System(s) engineers work for a builder, with a client. Developers typically don't do system architectures because they don't believe complex systems are to be designed -- they just happen..." [6] This is the argument on "magical self-organization" in systems development - sort of like Wikinomics principles. Complex systems just "don't happen – they need to be architected and designed to "do well."

People like to tinker and seize on point designs-going off in their own direction "just because they want to." They rarely ask whether or not a large, complex system concept will behave statically much less dynamically and assessing complex systems in a static and dynamic state is the reason for needing architectures. A static assessment leads to gap analysis, while dynamic assessing reduces risks and costs.
It is costly to acquire all of the systems and the platforms they reside on in order to assess them – thus, modeling and simulation (M&S) efforts are an important means of validating the product's value and efficacy.

7. IPA ARCHITECTURE

Characteristics of the IPA views that constitute the product set of the architecture include:

- The component systems required for the process to work can and do operate independently.
- They were acquired separately and maintain a continuing operational existence independent of the complex system they participate in.
- The Information Process and its architectural rendition is never fully formed – its development and existence is evolutionary with functions and purposes added, removed, and modified over time.
- There is a wide/large geographic extent. Note: wide or large implies that only information can be exchanged between the component systems, not energy or mass. (see Figure 2)

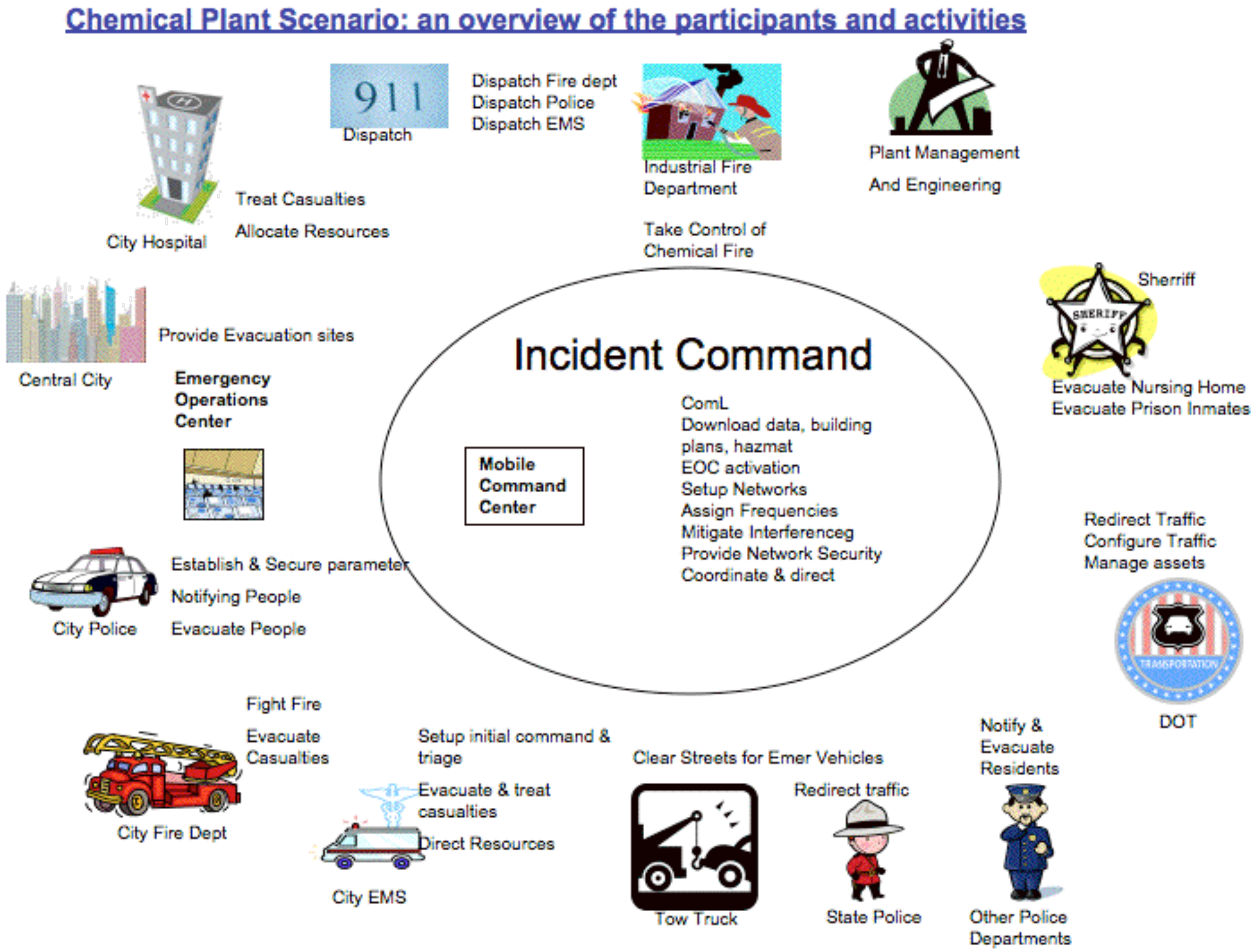

Figure 2.

These characteristics demonstrate the complexity of the Information Process, and an attempt to document the 100% solution is futile - it is simply too difficult to wrap one's arms around it all. Architectural products provide a venue for depicting the structure of component systems, their relationships, (e.g., the organization of information sharing across diverse and sometimes competitive user communities), the technical aspects of data collection and presentation, and guidelines governing their design and evolution over time. The result is an architectural framework with views that accommodate those approaches, models, and definitions that are needed to communicate and facilitate the presentation of key architectural information. The views describe architecture vision, principles, guidance, processes, and other characteristics, and establish a common foundation for understanding, comparing, and federating the information process. [5]

There are, at a minimum, three architecture views that need to be evaluated. Regardless of those employed, certain architectural views are necessary to portray the various levels of system and user interaction. The authors believe the framework for the Information Process Architecture should be drawn from all frameworks currently in use.

7.1 Above All Views (AVs)

Regardless of the intended use of the architecture, an AVs documents assumptions, constraints, and limitations that may affect high-level decision processes for a specific architecture. AVs document viewpoints from which the architecture is developed and the context – the vision and the tasks being accomplished by the user communities, doctrine and policy, concepts of operation and circumstances under which the complex system users and systems are assembled to do something. They identify start and stop dates, level of effort and costs necessary to develop the architectural products and the user communities that perform activities. Most importantly, AVs provide textual definitions – e.g., a glossary, the repository of architecture data, taxonomies and their metadata (data about the architecture data). AVs offer the common lexicon and dictionary from which all architectural product users base their interpretations as well as a central repository for a given architecture's data and metadata - thus, enabling architectural products to stand alone, allowing them to be read and understood with minimal reference to outside resources.

7.2 Operational Views (OVs)

Operational Views describe the tasks and activities required for information to be processed successfully, the participating users and the operational nodes where user activities take place and the associated information exchanges. The descriptions found in OVs are the vehicle for:

- Examining business processes for reengineering or the insertion of new technologies (e.g. Cognitive Radio).
- Identifying training needs for the user community.
- Exploring the implications of doctrine and policy.
- Coordinating the myriad user relationships
- Defining the high level requirements that need to be supported by resources and systems (e.g., communications throughput, specific node-to-node interoperability levels, information transaction time frames, security protection, etc.).

7.3 Systems and Services Views (SVs)

A "system" can be thought of as a compilation of resources and procedures united and synchronized by interaction or interdependence to facilitate the user's completion of an activity. Services, a unit of work done by a service providers – software agents - are offered by providers and used by the consumers while performing activities within the process; both provider and consumer are roles played by software agents on behalf of their owners.

Systems or Services Views describe the systems and services of concern and the connections among them as they relate to the OVs. The SVs are developed to establish the complex system's baseline, make investment decisions concerning cost-effective ways to meet the user's operational requirements as well as evaluate the state of the complex system's interoperability and/or make interoperability improvement recommendations.

7.4 Technical Views (TVs)

The Technical Views provide the basic set of rules governing the arrangement, interaction, and inter-dependence of system parts or elements and assist in ensuring that a system satisfies operational requirements. The TVs provide technical systems implementation guideline(s) upon which engineering specifications are based, common building blocks are initiated and product lines are developed. These views include a collection of the technical standards, implementation conventions, standards options, rules, and criteria. Technical views can be thought of as the technical standards criteria governing the implementation and integration of the selected systems. Technical views offer opportunities to articulate the technology and implementation roadmaps for the complex system's integration approach or individual system development.

7.5 View Summary

The various views fill the requirements of the users, new system designers and the support plans to:

- Describe what needs to be done, who does it, the information exchanges required to get it done and the operational activities performed by the system users. (OVs)
- Relate specific system functions required to satisfy the information exchanged and those specific systems providing same. (SVs)
- Identify/prescribe the standards for integrating existing systems as well as what new systems must meet. (TVs)
- Set the tone for the overarching aspects of an architecture that relate to all three views and set the scope and context of the architecture. (AVs)

8. CONCLUSIONS

We have described a paradigm shift: IT_S, the Information Tsunami, is poised to engulf society as we know it, where previously independent systems impinge on each other, and need to interoperate effectively. The results are complex systems, or systems of systems.

Systems with Cognitive capability arising from processes that function autonomously provide a means to reduce complexity, increase system capability, provide users with convenient wireless connectivity, and improve utilization of the RF spectrum. We have introduced the concept of Information Process Architecture (IPA) as a means of describing the structure of such systems.

IPA, with its multiple views (types include: All View, Operations View, Systems and Services View, and Technical View) provides a path to understanding the components of complex systems, and serves to facilitate the problems associated with their merger.

9. REFERENCES

[1] For definitions see: SDRF Cognitive Radio Definitions SDRF-06-R-0011-V1.0.0 8 November 2007
[2] Wikipedia article "Exabyte", June 2009
[3] Wikipedia article "Communication"
[4] Wikipedia article "Holon"
[5] Department of Defense Architectural Framework, Volume II: Product Descriptions, 9 February, 2004
[6] D. Tapscott and A.D. Williams, "How Mass Collaboration Changes Everything, Wikinomics