19th ICCRTS – C2 Agility: Lessons Learned from Research and Operations

"Agility through Automated Negotiation for C2 Services"

Topic 1: Concepts, Theory, and Policy

Topic 7: Autonomy

Topic 5: Modelling and Simulation

LTC Jose B. Neto, Dr. Michael R. Hieb, Dr. Paulo Costa
Center of Excellence for C4I
George Mason University
4400 University Drive
Fairfax, VA 22030
USA
001-703-993-3990
mhieb@c4i.gmu.edu
(Point of Contact)

Dr. Celso Massaki Hirata
Instituto Tecnológico de Aeronáutica
São José dos Campos, SP
Brazil
hirata@ita.br

Agility through Automated Negotiation for C2 Services

Abstract

Command and Control (C2) systems currently are developed for specific functions and limited application. Many systems deal with logistics and the management of resources during operations (e.g., Incident Response). Because different C2 systems often interoperate in very limited ways, they are difficult to get to work together without much manual intervention. This also limits the agility of operations due to the constraints of the automation used. However, Internet technologies have been developed to interoperate in a different way. Google and Amazon use web services that employ a "Negotiation" model to allow the development of very flexible responses to market conditions.

There are many advantages to using negotiation protocols with automated systems. The traditional resource allocation process requires numerous meetings between representatives from the organizations involved to develop agreements. There are few tools available to assist in this process. We propose an innovative dynamic and agile methodology for supporting C2 using automated negotiation of electronic contracts (e-contracts). These e-contracts can be implemented by commercial Web Services and provide an alternative to having to specify in advance all possible interactions between C2 systems. There is a main negotiation cycle where agreements are put into e-contracts prior to operations. During operations, e-contracts are invoked to perform rule-based transactions triggered by situational data.

Using e-contracts for automated negotiation could increase efficiency in decision-making. We present a case study on using this e-contract negotiation methodology in a C2 context in Brazil. We have modeled the operations of the Rio de Janeiro Command Center that will be in place for the World Cup (2014) and the Olympics (2016) in Brazil and show a detailed example of how automated negotiation could be used for Incident Response.

Keywords: Negotiation, Web Services, Simulation, Operations, Planning, Collaboration

1. Introduction

The importance of interoperability, reuse and sharing data can be seen by the success of Smartphones. These mobile devices keep people connected by allowing them access to multiple databases distributed throughout the Internet. Users can store information in remote locations, share applications with other people, control hardware and publish information in real time to social networks. The main support for the Smartphone technology is Internet technologies (i.e. Web 2.0).

The Internet was designed to be evolutionary in nature, enabling new features to be rapidly incorporated into its services. Current applications are designed to run on different kinds of devices in a variety of mobiles phones. Some of the popular types of are called mashups. They allow simple implementation of complex interfaces. These Application Programming Interfaces (APIs) specify how different software components should interact with each other and allow developers to instantiate and integrate data and functions easily, instead of building them separately. Google Maps, Twitter and Amazon provide content for this type of mashups (Pautasso et. al., 2008).

Information Technology (IT) protocols and applications created for business purposes can usually be adapted or directly used within the military domain. However, this transfer often does not occur within the Command and Control (C2) environment. Many factors contribute to reduce the interoperability between C2 systems: specific devices, different network technologies, many enterprises and new cryptographies. Despite being focused on coordinating actions in a general sense, C2 technologies must also be secure and robust for military uses. Outsiders cannot be allowed access to C2 information or systems. In fact, a C2 Center needs to receive information from many different systems, and process this information in a timely fashion. Armies, Air Forces and Navies have different C2 projects, technologies and patterns of IT. Many nations maintain autonomy of their service branches and C2 systems and this causes challenges in integration, sharing, and reuse of resources. Development of new C2 applications also needs to adhere to current IT standards. The result is increased use of Service-Oriented Architecture (SOA) and web services.

In this paper we develop and investigate an innovative framework for negotiating resources through e-contracts. The scenario chosen to demonstrate this methodology is a security incident in Rio de Janeiro, host city of the next World Cup (2014). What differentiates our approach from related work on Incident Response is the use of e-contracts as a basis for resource management within a C2 domain. The goal of the framework we have implemented is to negotiate resources for responding to an incident semi-automatically. In our approach, e-contracts are designed to be used by web services as a computational framework for integrating different systems.

Assume the resources used in a Combined Operation (soldiers, helicopters and aircraft) are described in an e-contract, to be coordinated between the military branches and the agency that controls the operation. If one assumes that all resources are controlled by these e-contracts, operation orders would use the terms and rules documented in an e-contract associated with the resources needed. Then the organizations involved will be able to coordinate according to the e-contract in a more autonomous and dynamic fashion.

This paper is organized as follows: **Section 2** – We present the background and concepts describing the lifecycle of an e-contract based in web service transactions. **Section 3** – We describe the C2 environment in Rio and how the methodology for electronic negotiation can be used for Incident Response there. **Section 4** – We present the e-contract implementation developed for the Rio Scenario. **Section 5** – Discusses an analysis of the approach implemented. **Section 6** – Gives concluding remarks and presents some final considerations for using this approach.

2. Background and Basic Concepts

2.1 SOA and Web Services

In C2, when we talk about technology integration, SOA is the most popular method currently used. Innumerable papers have addressed SOA as a way of integrating IT in the military area. For SOA an application becomes a service, and the set of services an inventory. One or more services can create a new service and the resulting service can be shared or added to another composition. Just as humans can interact with Internet services, web services can also interact with Internet services without human support. Web services exchange data and update representational states and often use Extensible Markup Language (XML) representations of Web resources. Two approaches commonly used are Simple Object Access Protocol (SOAP) and Representational State Transfer (REST). The first approach uses envelops with encrypted messages inside. It is a World Wide Web Consortium (W3C) standard and difficult to build a framework without SOAP components. The second approach is the REST protocol – it uses a simpler approach based on reuse and native HyperText Transfer Protocol (HTTP) methods to exchange data.

The REST protocol was first proposed in a doctoral thesis (Fielding, 2000). Currently, many enterprises exchange their services using SOAP, which is considered slower and complex than using the REST protocol (Pautasso et. al., 2008). The REST protocol follows the traditional model of the web services schema, but does not need Universal Description, Discovery and Integration (UDDI). Web service APIs that adhere to the REST constraints are called RESTful. RESTful Web service APIs can designate the location of resources using a Uniform Resource Identifier (URI) (Couloutis and Kindberg 2010). It is possible to adapt the code of RESTful Web services as necessary and messages are exchanged exactly when needed. It is a better style for when you want to exchange data under low bandwidth communication. By using data in the REST style everything can be turned into an action: general information, a map, software version, a relationship, a web directory, a photo, etc.

2.2 Lifecycle of E-contract for web service transactions

From a business viewpoint, Internet services can be represented in terms of contracts, with a provider and a consumer. Digital contracts are called “e-contracts”. However, many e-contracts are merely copies of physical contracts, failing to leverage the opportunity of automating data manipulation and processes. Neto and Hirata (2013b) propose that the lifecycle of the e-contract for managing the agreement between providers and client have six sequential phases (see Figure 1). Each phase has an input and output as requirements. In general, each phase produces a specific artifact (see Figure 2.3) or manages one produced during a previous phase.

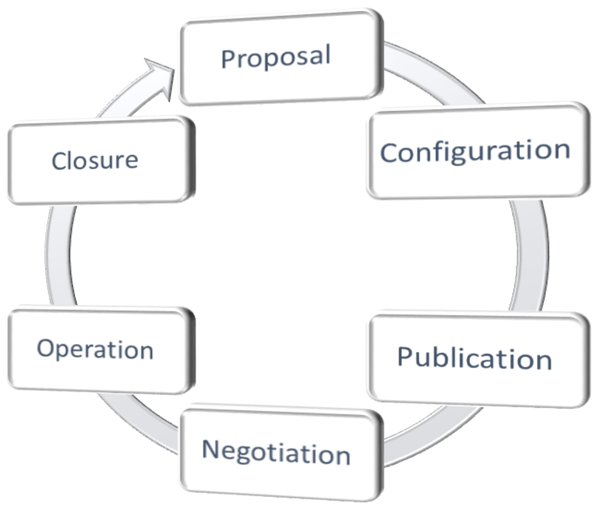

Figure 1 – The Lifecycle of an e-contract

Figures 2.1 and 2.2 below show how a simple transaction is conducted between a Provider (P), a Client (C) and a single Broker (B). The Broker is a filter and also responsible for formatting agreements, validating signatures, and saving the e-contracts in use. The provider has Services (S) and builds a Draft e-contract (D) by using a Template (T), stored in the Broker. This Template (of an e-contract) can contain information such as component type, methods and data. Other components are a Pre-contract (Pc), and an e-contract (E). The e-contract will contain both parties’ signatures – Client and Provider. Figures 2.1 and 2.2 show the sequence [(a) to (f)] of an e-contract’s lifecycle. In the first phase, (a) *Proposal*, the Provider P, who has services to offer, searches for templates that are relevant. The purpose of this phase is to get information from the Broker B, for preparing a draft contract, based on the template’s fields. In the second phase, (b) *Configuration*, the Provider’s services fill out the fields in the draft contract. The most important action in this phase is the virtual connection between the Provider’s services and fields in the draft contract. In the next phase, (c) *Publication*, the Provider “signs” the Draft e-contract and it becomes a Pre-contract.

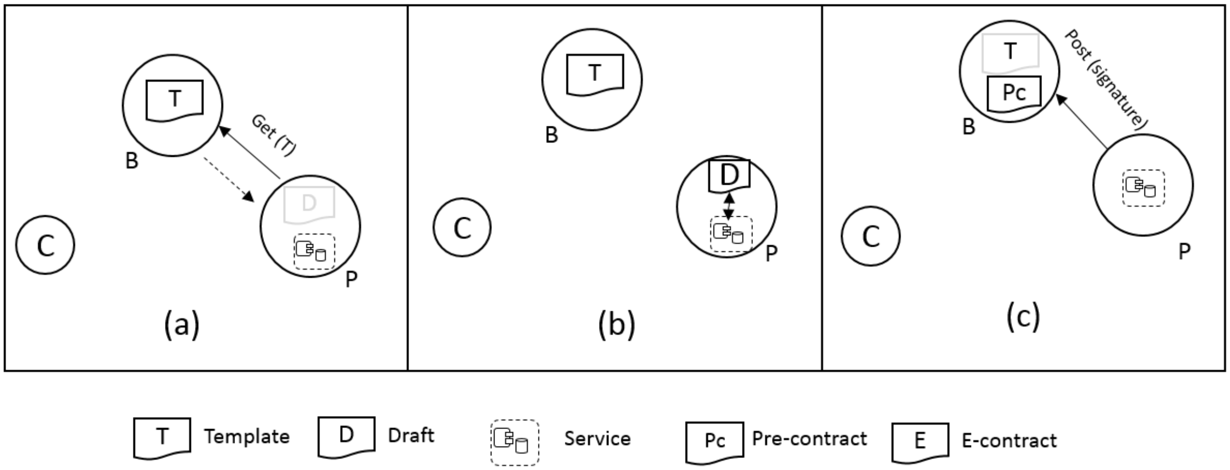

Figure 2.1 – *Proposal, Configuration,* and *Publication* phases in the Lifecycle of an e-contract

Pre-contracts are published and available to search in the *Negotiation* phase. In the *Negotiation* phase (d) the Client C, can request a change in a field in the Pre-contract. The Broker is responsible for the certification of the signatures and validation of the documents exchanged. Many messages could be exchanged depending upon the number of resources concerned. When the Client signs the Pre-contract it ends the *Negotiation* phase and the next phase (e) starts, where the Pre-contract becomes an e-contract. The *Operational* phase (f) starts when the Broker stores the e-contract, and the e-contract is ready to use.

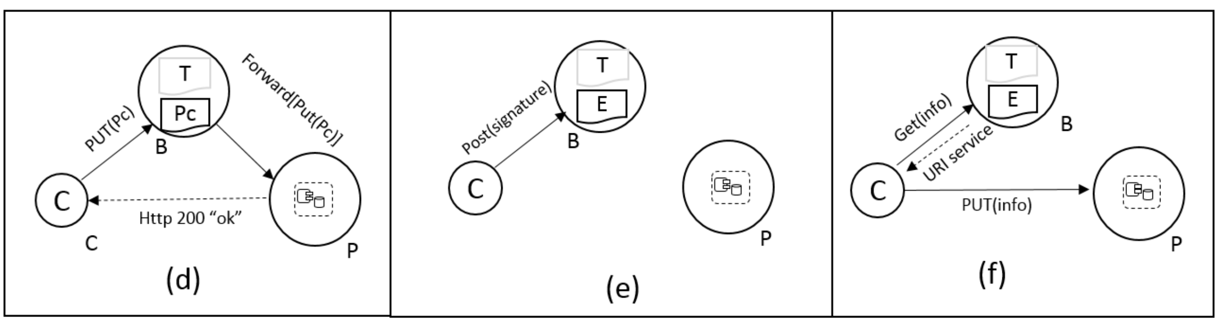

Figure 2.2 – *Negotiation* and *Operational* phases in the Lifecycle of an e-contract

During the *Operational* phase (f), the Client can retrieve information from the services in the e-contract. In this phase the Client obtains the quantities of resource that it needs by using the services in the e-contract. The *Closure* phase occurs when the transaction is concluded and is not shown.

The representational states (implemented by RESTful web services) discussed in this paper are divided in two categories: variable and constant. The first category, variable data, is created during the lifecycle of the e-contract, and may be updated and deleted if permission is given. The second category is constant data that does not change. This data is retrieved from military documents: doctrine manuals, standards, laws, and others. To clarify, we can "negotiate" the value of variables, while the constants are used to construct the e-contract.

Figure 2.3 shows the sequence of the contract types (artifacts) used to produce the e-contract.

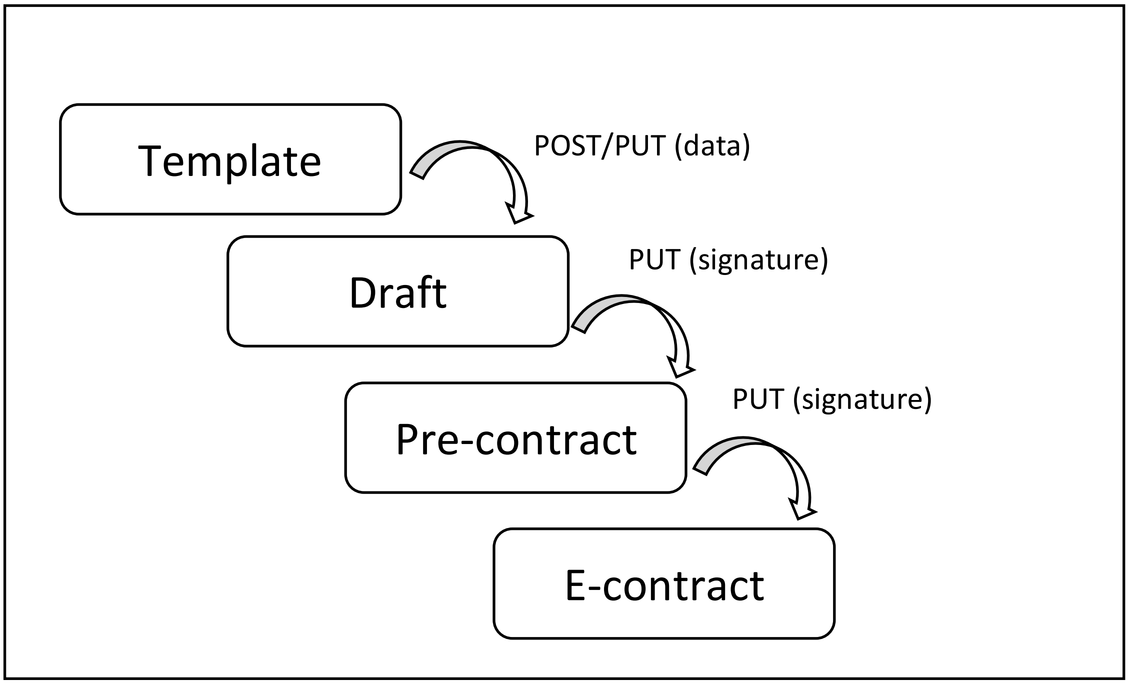

Figure 2.3 – Sequence of Contract Types in the Lifecycle of an e-contract

2.2.1 Formalizing the methodology as a Finite-State Machine

We can characterize the methodology as a finite automaton. A Finite State Machine is characterized by states, transitions and an alphabet (Lewis and Papadimitriou, 1997). The initial state is the Template and the final state is the e-contract (Figure 2.2).

The methodology is formalized as a quintuple $E = \{Q, q_0, \{q_f\}, \Sigma, \sigma\}$ where: $Q = \{S_1, S_2, S_3, S_4, S_5, S_6\}$, $q_0 = \{S_1\}$, $qf = \{S_6\}$, $T = \{t_1, t_2, t_3, \ldots, t_n\}$, $S = \{s_1, s_2, s_3, \ldots, s_n\}$, $\Sigma = \{T \cup S\}$, $\delta = \{\Sigma \times q_n \subset Q\}$.

E represents the lifecycle of the e-contract and has phases, transitions and tokens to move from one phase to another. Q represents the set of phases, q_0 is the initial phase, $\{q_f\}$ the final phase, δ the transitions, and Σ the tokens (alphabet – Terms and Signatures). The alphabet is the union of Terms and Signatures. The set of Terms T, are representational states and are the parts of the contract used to produce rules (Neto and Hirata, 2013b). The term "must" in the e-contract represents an *obligation*. Signatures are the mechanism used to allow the movement from a *Publication* phase to the *Operation* phase. In particular, during the *Negotiation* phase, we can change many representational states, but to finalize this phase a signature is necessary. In the S_1 phase we have a Draft e-contract and in the S_5 phase the e-contract is ready to use, as defined above. In fact, the artifacts follow the phases because, for example, during the Negotiation phase, the artifact handled is the Pre-contract. S_6 is the final phase, Closure, when the obligations of the e-contract are concluded.

3. Improving C2 Operations in the Center of Command and Control in Rio de Janeiro

This section presents an application of the methodology described above in a security scenario in Rio de Janeiro.

3.1 Integrated Center of Command and Control of Rio de Janeiro

The Integrated Center of Command and Control of Rio de Janeiro (CI-CCRJ) was inaugurated in January 2013 and is shown in Figure 3. Inspired by integrated security models adopted in London, New York, Mexico and Madrid, the CI-CCRJ houses various state, county and federal government agencies, as well as Military Police, Civil Police, Fire Department, Medical Service, Federal Police, Federal Highway Police, Civil Defense and Traffic Engineering Company of Rio (CI-CCRJ, 2014). It is a command center where live images from more than 500 cameras around the city will be monitored (CI-CCRJ, 2013). The Center supports 30 state agencies during routine city operations, as well as the Ministry of Defense during major events. In Rio de Janeiro, during the Soccer World Cup, there will be more than twenty different agencies working in public safety (Santos, 2006).

Figure 3 – Internal and External View of the CI-CCRJ

New and sophisticated equipment will be used in the CI-CCRJ, such as Unmanned Aerial Vehicles, (UAVs). Figure 4 shows a picture taken by a UAV of a Soccer Game, giving an example of the situational awareness that will be available. The major challenge of the Brazilian's Ministry of Defense is to achieve a higher level of interoperability between the C2 Systems of its service branches.

Figure 4 – Picture of Brazilian Soccer Game Taken by a UAV

The Ministry of Defense has conducted Combined Operations between the branches as a means to improve interoperability among them and has applied in the CI-CCRJ. More specifically, each branch has its own operational system, network communication, C2 systems and specific protocols.

During operations, representatives of each agency stay within the CI-CCRJ, and are responsible for managing the orders related to incidents, usually by using the agency's own network. The CI-CCRJ's C2 systems work disconnected from the different agencies C2 systems and it is not possible to update them in real time. One reason is because it is too hard integrating different technological projects from each branch, built in different decades, with different designs. Even after the Ministry of Defense's creation in Brazil in 1999, the branches still have considerable autonomy in decision-making regarding the technologies used. Within the branches there are still incompatibilities between systems that create technological barriers to integration (Santos, 2006).

Bates (2013) defines several maturity levels of capabilities in the C2 area. It is expected that there are different levels of integration between the CI-CCRJ and agencies, depending on the C2 maturity level of each agency. The ability to integrate is highly dependent upon the level of maturity for each agency. In general the Army, Air Force and Navy have a similar level of maturity, however it is not realistic to expect that civil defense, police or other agencies are at the same level. We present a way to integrate that is performed by a web services integrated with a C2 system that stores the data exchanged and negotiates e-contracts.

An extract of the planning process used in Combined Operations of the Brazilian Air Force is presented below (from the Brazilian Air Force Manual of Combined Operations). The process is similar for many operational sectors. It incorporates feedback from analyzing situational awareness, so that new orders can be generated to respond to changing conditions.

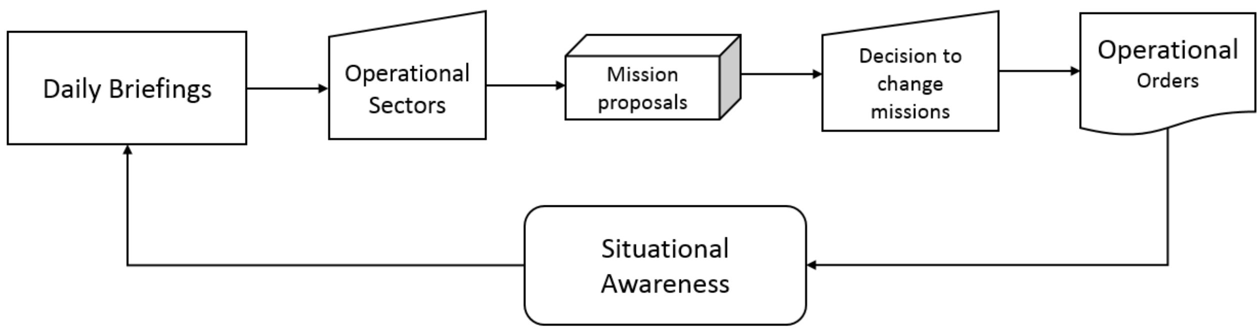

Figure 5 – Military Combined Operations Planning Process

The process used at the CI-CCRJ takes a total of approximately 12 hours for detailed planning. Every mission is updated from the analysis of current situational awareness. The next day, new missions are planned for the new situations detected. The challenge in this paper is to present a proposal to adapt the Combined Operation process so that sharing of resources is done semi-automatically. We use an automated negotiation process relying upon Templates that list the relevant data necessary for the lifecycle of an e-contract (as in Figure 1).

3.2 Using e-contracts to Support Combined Operations

The literature about the application of electronic contracts in C2 environments is scarce. (Aagedal and Milosevic, 1998) described the use of contracts as a tool to support complex distributed systems. They defined rules that regulate the use of e-contracts to support interrelationships between the general community, service providers, and government players in the context of disaster relief. However, their work was focused on Quality of Service (QoS) and did not address the use of e-contracts to query the status of resources in real time, and had no provision for updating the behavior of resources during the operation.

We base our work on the use of e-contracts for operations. It’s not necessary to pre-determine exact tasks as general types of tasks can be pre-approved and used to define the Pre-contract. An example is a military escort task, since the entity implementing the task will follow certain rules – as moving in a given trajectory for some time or space, or for forming a convoy with other agent. The states involved are: Duration, Location, Time, Altitude, etc.

Assuming implementation in the CI-CCRJ, the CI-CCRJ agencies' web service could make a Draft e-contract. The framework will support phases autonomously during the negotiation phase, allowing the use of the e-contract in the operating phase (Neto, et al., 2013a).

In this methodology reports and forms must be adapted to the e-contract context. By using web services the publication of the resources is available in the form of representational states that may be handled via REST methods. There must also be a representation of permissions in the various types of contracts during the lifecycle. Until the template becomes an e-contract many different permissions are needed. Figure 6 shows that a Unified Modeling Language (UML) representation of an e-contract based on different kind of military documents. The e-contract would be signed by one or more agencies and defines rules about the representational states allowed. Each agency has one or more operational systems that can publish the URI of its data in the e-contract representational state. The tasks in the e-contract are linked to operational data.

In our methodology C2 can be seen as the process of determining the relationship between desired and actual results to guide a more rapid response to incidents (Brehmer, 2007).

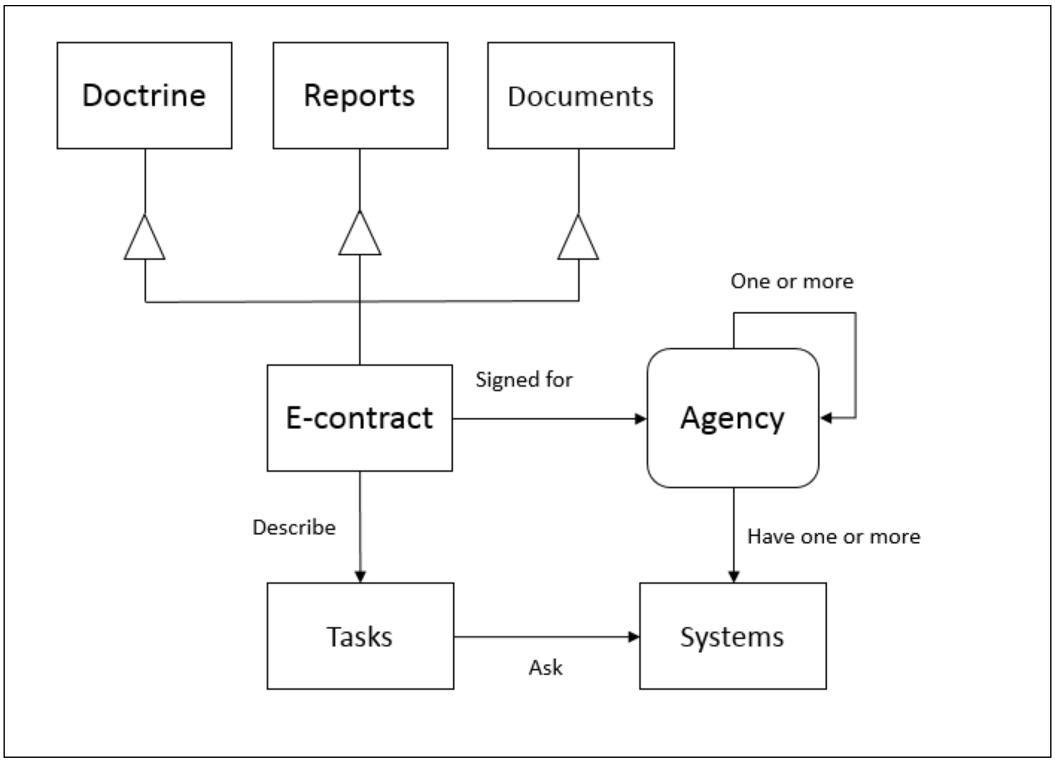

Figure 6 – Testbed UML e-contract

In a military scenario, for example, a Medical agency may request one or more helicopters from the Brazilian Air Force. The CI-CCRJ can further define the parameters of the operation. Trading is done on modifiable parameters (representational states in the e-contract) as in the phases in Figure 2.1. A new template from the Broker would be developed based upon doctrine and other documents. The *Negotiation* phase is closed when the Medical agency signs the contract. At this point we are ready to make use of the *Operational* phase as shown in Figure 2.2, where the specific operation will specify the helicopters needed, and resources are requested from the appropriate organization by the CI-CCRJ. The lifecycle finishes when the transactions in the e-contract are concluded. Later, the proposal stage can be performed again and revalidated iteratively, as in a lifecycle of software development, but both parties must agree.

To implement the framework proposed by Neto, et al. (2013), you first need to make a scalable architecture with two or more Brokers. Our proposal is to have Brokers running at each organization to integrate operating systems with CI-CCRJ. Within this hierarchy, the CI-CCRJ Broker works on the upper level and Brokers of the agencies at the second level, as in Figure 7. The Brokers (using a UDDI) store the Templates of the agreements between the agencies and the CI-CCRJ. When the Template is first filled out it is designated “draft” and signed when the agency approves it. The Draft e-contract then becomes a Pre-contract. As in Figure 2.3, negotiation takes place using the Pre-contract file and is limited to data that defines resources. No negotiation is possible on the template, just about data. The Pre-contract posted by an agency can be accessed by web services from other agencies, under the coordination of the CI- CCRJ.

Using Figure 7 we can illustrate the details of a typical e-contract negotiation. First e-contracts are negotiated internally within the operational Intranets. The e-contracts are shared, summarizing the interests of the agency. A final contract is created from two or more published by a Broker and signed by CI-CCRJ contracts. In fact the e-contract is a set of representational states stored in a Broker. The composition of e-contract after the negotiation is a virtual set of representational states from all e-contracts available at Level 2.

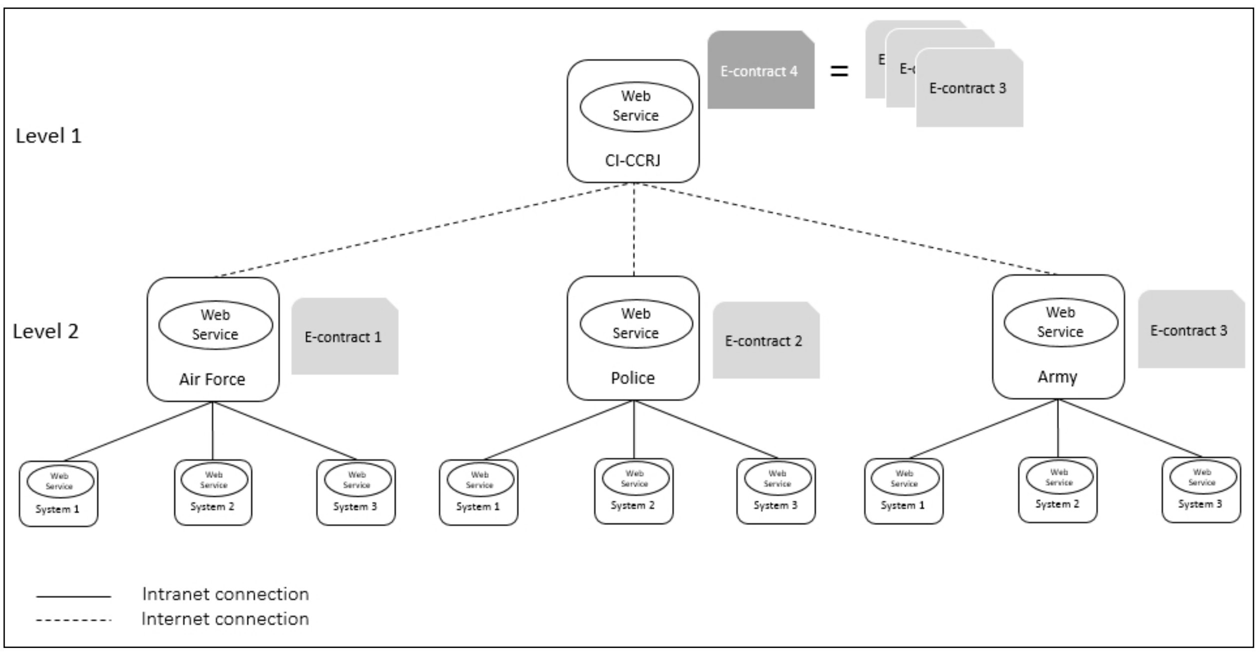

Figure 7 – CI-CCRJ Web Service Framework

After the *Negotiation* phase the e-contracts from the branches at Level 2 are ready to use in the negotiation phase with the CI-CCRJ. We can use the negotiation in two ways. The same e-contract can be used during the entire Combined Operation or there can be negotiation before each major phase of the operation. The contracts will be used in the operation phase, to support the allocation of resources for the daily missions. The process should prioritize which organizations can enter data for various operational areas in the e-contracts. For example, the operating systems of the Air Force may provide most of the information required in the e-contract, but another branch may propose modifying the data in the e-contract. Everything must be coordinated by the CI-CCRJ.

During the *Negotiation* phase, when resources are allocated for sharing with agencies by the Center, it is possible to simulate the results of actions in a virtual setting. For example, how many helicopters or soldiers would be needed to cover an area to adequately respond to multiple incidents?

4. Investigations

In 2010, during the XII Symposium of Operational Applications in Defense Areas in Brazil, a partnership was established between George Mason University (GMU) and Aeronautics Institute of Technology – Brazil (ITA) to support C2 research by providing a Modeling and Simulation environment for C2 Planning and Cyber Warfare. A simulation environment was developed, the C2 Collaborative Research Testbed that uses several COTS (Commercial Off-The-Shelf) tools and open standards, to provide a rapid prototyping and modeling environment for C2 missions.

The main COTS tool used is MÄK VR-Forces (MAK, 2012), which is a powerful and flexible simulation environment for scenario generation. It has all the necessary features for use for developing Computer Generated Forces (CGF) for simulating a complex operational environment.

Several scenarios involving the CI-CCRJ were simulated. Portions of the e-contract lifecycle were implemented and analyzed (Neto, et al., 2013). The scenario involved determining which assets (helicopters and fixed wing aircraft) that can be used to respond to a public safety incident (as in a riot, explosion or natural disaster) in the Rio area covered by the CI-CCRJ. E-contracts between agencies and the CI-CCRJ were developed before any operations were simulated. They specify the resources available for the operation and their requirements. Contracts are XML schema files handled by RESTful Web Services that can be accessed in real time. Using e-contracts it is possible to identify the current situation of resources and to send orders. In the Rio scenario, the plan for flight operations (the Air Tasking Order) is updated daily, and can be changed in response to some significant event. A resource relocation algorithm identifies the resources available. Only the Operational Phase was simulated, assuming the negotiation phase was already complete.

The Rio scenario works with active orders, and specifies the assets and resources necessary to complete a mission. Events in the scenario trigger automatic orders for dispatching aircraft and helicopters. Agencies can send requests for tasks (to the CI-CCRJ) or for resources. The C2 Testbed allows use of either simulated sensor data, or real sensor data from the actual sensors in the Rio de Janeiro city network. All data are representational states resources that can be manipulated by web services.

The scenario works in the *Operational* phase, using already signed e-contracts. The e-contracts are checked when a new order is sent. Each task has contractual clauses that specify conditions for permission, obligation or prohibition (Neto and Hirata, 2013b). Requests may be made either by the CI-CCRJ, represented by a server that manages the asset e-contracts, or by the agencies, represented by virtual machines.

The contracts are instantiated from doctrinal documents, reports of previous operations, military legislation or even state and federal laws involving the operation documents. These documents form the basis of templates that will be available in the CI-CCRJ server. Contracts will connect and share data from operational systems that manage the resources during a Combined Operation. The solution addresses a major problem for C2 systems, the time it takes to respond to an incident (Oosthuizen and Pretorius, 2013). The implementation of the e-contracts in the Rio Scenario uses a dynamic allocation of resource. The C2 Testbed also has the ability to use mobile devices for data access and to update the variable data in the e-contract.

5. Analysis and Discussion

Our analysis was based on actual documents used during Combined Operations by the Brazilian Air Force. The documents reviewed were examined to determine what information could support the e-contract lifecycle. Then this information was divided into two types: template and data. 93% of the total can be converted into the template and 7 % handle data (representational state). We were concerned about the impact of the lifecycle on the network throughput, both during the entire lifecycle, and in particular the *Negotiation* phase. We used *Ethereal* software to analyze the number of packets exchanged during a simulation of an e-contract lifecycle with a simple e-contract with 10 fields between two different agencies.

Figure 8 is an analysis of this negotiation between two agencies The Broker is represented by the CI-CCRJ. The different shapes are methods allowed during each phase. We use a logarithmic (log) scale because the first three phases, in general, are on the order of milliseconds (they are on the same Local Area Network – LAN). The *Negotiation* and *Operation* phases need to share different networks and this directly impacts the total time used. The resulting time of these transactions is higher than a simple hyperlink access and lower than an impact of one download in a LAN with restricted bandwidth.

Figure 8 shows the number of messages exchanged during the lifecycle of a nominal e-contract. It can be seen that the negotiation phase has the highest number, almost 32 messages. The methods used per phase can also be seen in the Figure 8. For example, in the Configuration phase the main method used is PUT. One recommendation for using our approach is to use low-bandwidth connections and mobile devices.

The direct benefit of this approach is to reduce the total time to respond to incidents, since coordination can be done before the actual response, and since both the pre-coordination and activities during the response can be semi-automated. Prior to any operations, documentation of coordination meetings can be used to identify the terms needed to develop the Template of the e-contract. Using this initial input and other documentation the automated negotiation can be initialized. For the Rio scenario we analyzed four documents, two from the Brazilian Air Force and two from the Brazilian Ministry of Defense. The static part of the document is used to populate the Template that will be stored in the processing module.

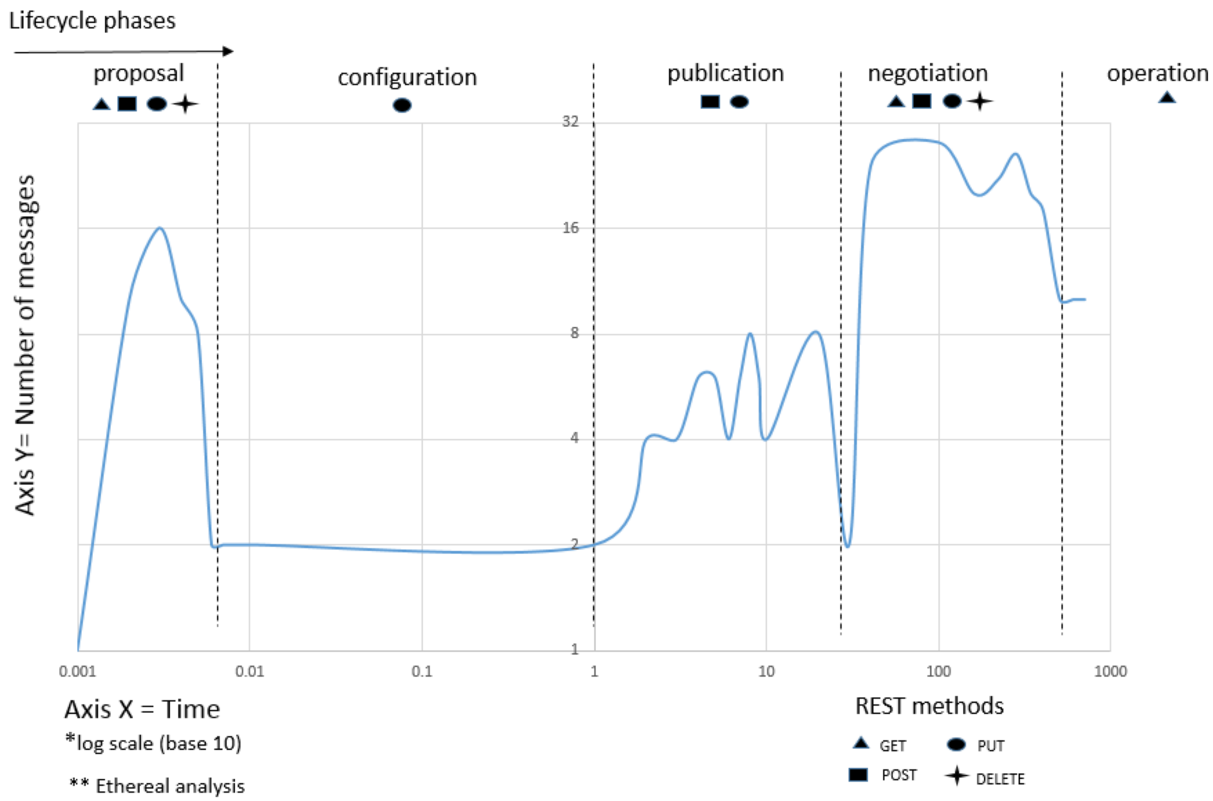

Figure 8 – Graphical Analysis of the Number of Messages during an e-contract Lifecycle

The automation of negotiation using web services could significantly reduce the manual effort of coordination. All parameters of the negotiation can be put in the web service rules. This approach could reduce the number of meetings in the planning process. The e-contract is then used in the *Operational* phase for delineating orders. During the Negotiation phase it is possible to simulate the scenario for various situations and triggers.

6. Conclusions

In this paper we introduce a new method for interoperability in the C2 area. E-contracts, constructed from military documents that are usually manually accessed during operations, are also used for C2 integration. We expose data normally found in paper documents in order to formalize agreements that can then be automatically processed via web services. The physical contract becomes both a template and a way to represent the state of an operation. Since each piece of data is given its own URI, agencies can easily manipulate this information in real time. We use rules within the e-contract to validate the orders and define tasks. Lessons learned in the C2 Testbed project facilitated the evolution of this architecture and showed how to implement a negotiation framework for Combined Operations. Our case study investigates negotiation at different levels in the CI-CCRJ and demonstrates how a virtual e-contract in the CI-CCRJ could be used. The next step of our research is to develop a formal model for contract language and validate its robustness.

7. References

Aagedal, J.O. & Milosevic, Z. (1998). Enterprise Modeling and QoS for Command and Control systems. Enterprise Distributed Object Computing Workshop, La Jolla, CA, USA.

Alberts, D.S, Hayes, R.E., 2006. *Understanding Command and Control*. CCRP Publication, ISBN 1-893723-17-8, USA.

Alberts, D.S, 2011. *The Agility Advantage: A Survival Guide for Complex Enterprises and Endeavors*. CCRP Publication, ISBN 978-1-893723-23-8, USA.

Allamaraju S., 2010. *RESTful Web Services Cookbook*. O'REILLY.

Bates, E., 2013. *Determinants of Achieving Effective Shared Situational Awareness within the Context of Global Maritime Partnerships*. 18th International Command and Control Research and Technology Symposium, Washington, DC.

Brehmer, B., 2007. *Understanding the Functions of C2 Is the Key to Progress*. The International C2 Journal. Vol 1, No 1: 211-232.

CI-CCRJ, January 2014. *Technology and Intersections of Information to Make Decisions in Times of Crisis and Organize Events.* Retrieved from <http://tecnoblog.net/108550/centro-de-operacoes-rio/>.

CI-CCRJ, February 2014. Rio de Janeiro Government. Retrieved from <http://www.rj.gov.br/web/imprensa/exibeconteudo?article-id=1608500>.

Couloutis, D. J. and Kindberg, T., 2010. *Distributed Systems, Concepts and Design*, Addison Wesley.

Fielding, R. Thomas, 2000. *Architectural Styles and the Design of Network-based Software Architectures*. Doctoral dissertation, University of California.

Harry R. Lewis and Christos H. Papadimitriou. 1997. *Elements of the Theory of Computation* (2nd ed.). Prentice Hall PTR, Upper Saddle River, NJ, USA.

Heffner, K. and Gupton, K., 2013. *Implementing a Standards Development Framework for the Coalition Battle Management Language*. 18th International Command and Control Research and Technology Symposium, Washington, DC.

MÄK, 2013. *VR-Forces*. Retrieved December 04, 2013 Retrieved from <http://www.mak.com/products/simulate/computer-generated-forces.html>.

Neto, J. B., Hieb, M., Hirata, C. M. and Paulo, C., 2013a. *Simulating the Allocation of Resources for Command and Control with E-Contracts*, The Interservice/Industry Training, Simulation & Education Conference, Orlando, USA.

Neto, J.B. and Hirata, C.M., 2013b. *Lifecycle of E-contract Based on Web Service*. World Congress on Engineering & Computer Science 2013, San Francisco, USA.

Oosthuizen, R., Pretorius, L. 2013. *Assessing Command and Control System Vulnerabilities in Underdeveloped, Degraded and Denied Operational Environments*, 18th International

Command and Control Research and Technology Symposium, Washington, DC, USA.

Pautasso, C., Zimmermann, O., Leymann, F., 2008. *RESTful Web Services vs. "Big" Web Services: Making the Right Architectural Decision,* International World Wide Web Conference Committee, Beijing, China.

Pullen, J.M., Corner, D., Gustavsson, P. M. and Grönkvist, M., 2013. *Underdeveloped, Degraded and Denied Operational Environments.* 18th International Command and Control Research and Technology Symposium, Washington, DC, USA.

Santos, J.C, 2006. *The Data Model of NATO for C2 Information Exchange*. Work towards Satisfaction of the Course on Specialization in Policy, Strategy and Military Administration at the Army Officers School in Rio de Janeiro, Brazil.

Ray, D., Gulla, U., & Dash, S. 2007. *Interoperability of e-Government Information Systems: A Survey.* In J. Bhattacharya (Ed.), Towards Next Generation e-Government (pp. 12-25). Secunderabad, India: Computer Society of India.

Trude, H., and Frank, T. J., 2011. *Employing Web Services Between Domains with Restricted Information Flows*. 16th International Command and Control Research and Technology Symposium, Québec, Canada.