A Contract That Manages Itself

The Time Has Arrived

Russell Chesebro

This article is about change. It is about taking a brave new step in the way contracting is performed within the Department of Defense (DoD) and how contracts are handled.

In fact, this article goes further. The contract will cease being a static document file on a computer server. The contract will manage itself. The contract will have a voice and it will speak. The contract will become empowered, and it will take action apart from human direction. The technology exists to bring the contract to life—and, once the first step on this road is taken, the world of contract administration will leap into the next century. The cost potential savings are so great that they are incalculable.

For those who are intrigued by the opening paragraph, for the skeptics, for everyone with a vested interest in how contract management within the DoD is performed, read on and you won't be disappointed. A computer file has been brought to life and has spoken its first words, "Hello Contracting World." We will never be the same.

In the year 2010, The DoD doled out $368 billion in contract awards. Each contract award resulted in a physical contract that was turned into a PDF file and stored as a static document on a computer network. How many contractors are supplying goods to DoD? How many of those contractors are still in business? How many unpaid contracts are out there, and how do they get closed if the contractor is no longer in business? How many information-technology (IT) business systems and spreadsheets does the Army use to manage contracts? The Navy? The Air Force? How many static contract files are there for each Service?

Chesebro *has an MBA in Information Technology Management and was a member of the engineering team that created the PDF file.*

The Office of the Secretary of Defense (OSD), on its public website for Defense Procurement and Acquisition Policy, lists nine software tools/websites that contractors can use for eBusiness. That is just one portion of the conglomeration of IT business systems used for managing DoD acquisition. But what is the essence of acquisition? It is a contract. Currently, contracts reside as static files on one or more IT business systems. These contracts are accessed by scores of people, all of whom have their own interest in the contract information and perform varied functions of contract management.

With so many people handling so many contracts on so many various IT business systems, how can that be managed? This is what is referred to as a wicked problem. A wicked problem is one that cannot be solved but must be continually managed. Some examples of wicked problems are poverty, disease, hurricanes and war. Trying to manage $368 billion worth of contracts across the different branches of Service within the DoD is a wicked problem. Breaking that problem down into smaller problems reveals an interesting pattern.

Wicked Problem No. 1: Data Integrity. A contract that is a static file can be copied, amended and stored in many different locations and versions.

The first small problem to look at is that of one contract residing on multiple systems in multiple locations. Although the contract may be the same in its original form, not all modifications will be synchronized across the multiple systems. That leaves a contract in many different states and—depending on which system a person uses to access the contract—will result in getting a correct or incorrect version of that contract.

Wicked Problem No. 2: Factoids. A static-file contract is dependent on institutionalized knowledge.

The second problem to note is one of institutionalized knowledge. Perhaps there is a contract that cannot be closed because there is an unpaid amount of $3.50. The company to whom the money is to be paid no longer is in business and the only person with the knowledge to close out contracts such as this retired last year. The contract then becomes an unresolved problem that will require extra labor costs to resolve.

These two wicked problems revolve around one reality: The contract is a static file managed by humans. In an age in which airplanes fly themselves and cars drive themselves, it is time to create a contract that manages itself. Contracting challenges technology and, in turn, technology inspires contracting.

The Smart Contract

The paradigm of a contract as a static document is about to change. The days of a contract being read, interpreted, acted upon and managed by contracting personnel is over. We don't need people to manage contracts because contracts can manage themselves. This concept was first discovered in 2014 at the Defense Contract Management Agency (DCMA) Contract Management Office (CMO) in Philadelphia, Pennsylvania. Having one contract reviewed by many people is an inefficient use of time and resources. In these days of tightening federal budgets, efficiency is of paramount importance in order for an organization to perform its mission.

The Smart Contract as an Object

In discussions about programming, the word "*object*" means a component with properties and methods. Properties are what an object knows about itself and methods are what an object knows it can do. A contract, as an object, will know things about itself. It will know how much it is worth. It will know who signed the contract, who administers the contract and when the contract is supposed to be complete. With a little additional development in the environment in which the contract object (smart contract) exists, the contract will be able to interact with other objects. That will enable the contract to know how much money has been paid to the contractor and how much is left. The contract will know how to close itself out. And if a problem arises, such as funds still not spent with the contractor no longer in business, the contract will know how to handle the situation. Among its many advantages, the smart contract will eliminate the problems associated with institutionalized knowledge. This is not an implementation of push notification. It is bringing a contract to life within its environment.

The smart contract will understand itself, its environment, the objects with which it must interact and the personnel with whom it will interact. A smart contract won't be ignored. A smart contract will know what actions to take when timelines are not met. A smart contract will manage itself, and that in turn will eliminate many contract management functions currently performed by humans.

Beyond the Smart Contract

The first problem to note before beginning this section is one of catch-up-to-fall-behind. In its basic form, this problem arises when an organization begins planning an upgrade to its system. By the time the planning and execution of the upgrade are completed, the upgraded technology already is obsolete. The smart contract is a first step. But a bolder move, a leap into the future, is needed so that—when the development is finished—the system remains far advanced.

The Intelligent Contract

The intelligent contract can be described in one word: ontology—the study of being. In this context, ontology involves describing information and relationships in an informative way. That sounds like a database. But unlike a relational database that stores and retrieves data items, an ontological database system brings understanding into the realm of data queries.

What does that mean in simple terms? Look at the iPhone assistant Siri as an example. When a person asks Siri a question, such as "Do I need an umbrella," Siri has to understand that this is a weather-related question and then search the national weather database for the weather conditions at the user's location to provide an answer. Siri understands the question in context, and also knows how to find the answer and report that answer to the user.

The World Wide Web Consortium is involved in this endeavor of creating smart, understanding programs and ontological databases by developing OWL (Web Ontological Language). It also supports a new query language developed for OWL called SPARQL—a pattern-matching scheme in which a database is queried for matches and certain of the results are graphed to determine a pattern. This is an enormous development because giving meaning to data has many practical uses. The future is here. But how does that fit with the DoD and the contracting world?

In an age in which airplanes fly themselves and cars drive themselves, it is time to create a contract that manages itself. Contracting challenges technology, and, in turn, technology inspires contracting.

Knowledge Is Power

A contractor has made two variations of the same product for one of the branches of the Armed Services. A modification to the product is requested, a new contract is signed and work begins. During final testing in a field environment, a major failure occurs. The representative of the Armed Services tells the contractor that the product does not meet the requirements put forth in the contract. The contractor states that the equipment meets the requirements to perfection. When asked to explain the failure, the contractor states that the requirements are met perfectly when the equipment is tested in a laboratory and that the requirements don't state anything about passing a test in a field environment.

Even though the contractor had produced two similar products and met the requirements for field performance, this contract did not specify field performance in the requirements. The Armed Forces representative failed to specify that part in the requirements section of the contract. Now the contractor has to be awarded more money to meet the new specification. Does this happen often? Yes. And it can be stopped with the implementation of an intelligent contract.

The Intelligent Contract Knows Itself

A smart contract knows details about itself (properties) and how to interact with other objects (methods). An intelligent contract knows its own being. Every member of the military who has driven a tracked vehicle knows that it must be able to pivot 360 degrees in the mud. But the mere fact that this is known does not mean it is written in the contract. An ontological database will solve this problem.

In the intelligent contract paradigm, an ontological database will be developed to link data from the disparate departments of the DoD into understandable knowledge. The chief focus at first will be the linking of data that deal with requirement specifications found in contracts. The methods used in the intelligent contract ontology are semantic methods. Interestingly enough, this effort was begun by the Defense Advanced Research Projects Agency (DARPA) in 1999. DARPA developed the DARPA Agent Markup Language (DAML) and a variety of programs, tools and datasets for use by government and commercial clients. It was the foundation of semantic Web programming.

Linking data from various organizations within the DoD—with the links based on semantics—will form the ontological database that will be used for understanding requirements specifications. What documents exist on the Army website that describe 360-degree pivot steering on a tracked vehicle? How would those documents match other documents within the Air Force web, or the Navy web?

The basic concept of the semantic methods is to search domains looking for similar data tags. The tags are matched in a logical order. This results in a semantic match. Then the matter of intent has to be evaluated. Hence, when Siri is asked whether an umbrella is needed, a search of the Web for the word "umbrella" would be insufficient. The intent of the person posing the question is to see if the weather forecast calls for rain. To understand the intent, the ontological database is built on semantic relationships.

Conclusions

When the ontological database is incorporated and the smart contract has dominion over its environment, amazing potentials become ripe for the harvest. Imagine using your voice to ask a contract who its suppliers are on its supply chain. Imagine asking the contract how a particular supplier has performed in the past. Imagine asking a contract if the supplier is likely to complete the order on time and within budget.

Turning those exercises in imagination into reality now becomes a matter of action because the foundational blocks already exist. These steps—implementing a smart contract and then an intelligent contract—will take the contracting IT business systems for the DoD into the future. There will be no catch-up-to-fall-behind issues.

The author can be contacted at **russell.chesebro@dcma.mil**.