The Italian Army C2 Evolution

From the current SIACCON2 Land Command & Control system to the LC2EVO using "agile" software development methodology

Dr. Angelo Messina B.General IT Army
Italian Army General Staff IV Dep. Head CIS area
DSSEA Secretary
Rome ITALY
angelo.messina@esercito.difesa.it

Dr. Franco Fiore - Principal Scientist
NATO Communication and Information Agency
Directorate Application Services
The Hague, The Netherlands
franco.fiore@ncia.nato.int

Abstract—The continuous evolution of the military "land operations" generates instability of the Command & Control systems requirements obliging the developers to work with unstable and unconsolidated mission needs. In 2013, the Italian Army General Staff Logistic Department decided to overcome the problem of the "volatile requirement" transitioning to a completely different software development methodology derived from the commercial, area but almost completely new to the "mission critical" software applications: the so called "agile" methodology. The introduction of "agile" in the development of high reliability software was not easy and implied the generation of a brand new "agile" methodology called: Italian Army Agile or ITA2. Setting up the LC2Evo (the evolution software of the land C2) implied the solution of many problems and the construction of a solid structure based on four "pillars": User Community Governance, Specific "agile" Training, new "agile" CASE[1] Tools, custom "agile" development doctrine. This paper gives a short description of the Italian C2 software evolution from the current SIACCON2 to the LC2Evo whose first components are in use already in various national and out of area operations. The LC2Evo "dismounted Soldier segment software using augmented reality and goggle glass-like display is also described. The major problems tackled and solved for the construction of the "four pillars" are treated.

Keywords—C2, land command & control, agile development, software engineering, LC2Evo, volatile requirements, augmented reality, dismounted soldier.

I. ASYMMETRIC OPERATIONS & VOLATILE REQUIREMENTS

To run an analysis on the evolution of the C2 requirements, reference is taken to the operation "Enduring Freedom" (Afghanistan 2001) in which Italy took part almost since the beginning. As part of the operation, Italian forces have contributed to the ISAF, the NATO force in Afghanistan, and to the Provincial reconstruction team. Italy has initially sent troops, based on one infantry company from the 2nd Alpini Regiment tasked to protect the ISAF HQ, one engineer company, one NBC platoon, one logistic unit, as well as liaison and staff elements integrated into the operation chain of command. Italian forces also commanded a multinational engineer task force and deployed a platoon of Carabinieri military police.

First of all, we observe that initially the operation was conducted by a coalition of United States, United Kingdom, Australian, and Afghan (Northern Alliance) forces. The activity was broadly supported internationally and by the United Nations (UN), but it did not receive its first official sanction until the December 2001 (UN Security Council Resolution 1386). Even at then the international approval was authorizing a six-month deployment under the name of: "International Security Assistance Force" for Afghanistan or ISAF.

The initial mission was relatively limited and included providing security for Kabul and its surrounding areas. But as the conflict in Afghanistan continued more nations and non-governmental organizations (NGOs) began various assistance efforts in Afghanistan, ISAF's mission scope was expanded (UNSCR 1510). In 2003, when Italy joined the operation ISAF's mandate had been extended to the entirety of Afghanistan. Many of the assignments of the force on the ground were far different from the traditional "military operations" and so were the connected mission needs.

ISAF was charged with assisting the Afghan government in establishing a safe and stable environment in urban centers and areas beyond Kabul. The responsibility for supporting reconstruction and development projects designed to help the Afghan government cultivate institutions for good governance and in support of the rule of law, required substantial adaptation.

The first relevant transition in the C2 architecture was caused by the North Atlantic Treaty Organization (NATO) assumption of leadership. Ideally the "coalition network" was supposed to transition to a full NATO network, that at the time was almost not existing. The other relevant change was the geographic dispersion of the forces. At the end of 2006, ISAF expanded its

1Computer aided Software Engineering: the tools normally used to produce software applications in a controlled and traceable environment

[3] The Defense & Security Software Engineers Association is a no-profit organization aimed at the development of a new Software Engineering Paradigm. The association includes members from the Defense and Security areas as well as from Universities and Industry. (www.dssea.eu)

bases of operation throughout Afghanistan covering the whole territory. Increasing the geographical footprint, adding more members and more and more non military organizations, made operations radically more complex and unconventional. The number and type of potential customers of the C2 systems significantly changed as well. Initially, ISAF consisted of roughly 5,000 troops concentrated near Kabul. By 2010, ISAF consisted of well over 100,000 troops from 48 different countries, including NATO, NGOs and Afghan partner institutions.

The range of "military" operations was including, but not limited to, humanitarian assistance, stability operations, counterinsurgency operations, and combat operations.

Quoting from: "Lessons Learned from the Afghan Mission Network Developing a Coalition Contingency Network" [1]

"*Although operations in Afghanistan will be pared substantially in the near future, the lessons learned by ISAF and through the development of the AMN*[2] *will be enduring. And while future contingencies and the coalition partners participating in these contingencies cannot be effectively predicted, the types of networking challenges they will face most certainly can."*

It is now clear that the mission needs are largely unpredictable while the mission support system (hardware and networks) are for the most part substantially the same throughout the mission. The volatility of the mission requirements has to be mitigated by the flexibility of an innovative software development process capable of continuity of change. The original concept of *Network Centric Warfare* (NCW) ready to deploy in all situations has been integrated by the *mission oriented approach* based on *mission threads.*

II. THE LAND C2 SIACCON 2

The SIACCON System (**SI**stema **A**utomatizzato di **C**omando e **CON**trollo) [3] was born to digitize the functions performed by the Army Command Posts in operation. Based on the concept of *Network Centric Warfare* (NCW) the system initial acquisitions deliveries started as early as 2002 and went on until 2013 when the Italian Army units started setting up the so called Operational Control Rooms.

SIACCON2 software, supports many of the operational functions defined in the contest of the NCW such as situational awareness and command functions, based on collected data and filtered aggregation and dissemination of information and orders.

The software architecture is double layered with a Digitized Service layer and a Functional Area Service (FAS) layer. The first layer takes care of all operational info structure needed to the application layer, while the FAS layer is providing the specialized services indicated by the reference doctrine such as: maneuver, Fire, 3rd dimension control, Information and Counter Information warfare, mobility and counter mobility, C4, Logistic support, etc.

The system has been employed in many exercises and operations proving the effectiveness of layer one and providing a limited number of FAS (layer2). The system is in service as "SIACCON 2 AMN" (Afghanistan Mission Network). Obsolescence of some components, cost connected to the traditional software development cycle high cost and difficulty to accommodate quick requirement changes imposed by modern asymmetric warfare, has induced the Italian Army to change the development philosophy for the future LC2 system.

III. THE FUNCTIONAL AREA SERVICE APPROACH AND LC2EVO

The doctrinal reference point to start building an evolutionary C2 system was the "Mission Thread" based approach adopted by NATO.

A. *The LC2Evo overarching tools and the FAS*

In 2014 the Italian Army General Staff Logistic Department started the development of the: "*Land Command and Control - EVOlution*" System. The product major item is software, while most of the hardware systems are the same supporting SIACCON 2 with the addition of some COTS.

The main reason for this engagement was the need to support the evolution of the land C2 keeping high "customer" satisfaction in a "volatile" requirement situation. Another major issue to be solved, was the need to substantially reduce the budget necessary for this software development and the subsequent maintenance.

The Army software engineers experimented the principal "agile" software development methodology available at the time named "Scrum Agile". This methodology, better described in the next paragraph, was and is, very successful in commercial environment, where it is nowadays the method of choice for the majority of software applications producers especially for the Android and Linux based products.

LC2Evo was initially started using Scrum Agile with production cycles of three weeks and one experimental *Scrum team* with programmers, subject matter experts, security specialists, a Scrum Master and a Product Owner. The team was composed by developers from industry and military people and was based at the Army Staff major facility. [2]

The initial phase of the production was extremely successful and even the very first "Sprint" (production cycle of three weeks) which was supposed to be only a start-up trial one, actually delivered the planned product.

While the product became more and more complex and the stake-holders' expectations grew, it appeared very clear that commercial scrum methodology was not capable of handling the peculiarity of a high reliability software production with an

[2]Afghanistan Mission Network

articulated User Community as the one in charge of the land C2 operational requirement. [4]

The Army, with the help of experts from the universities, designed a particular "agile" software development process named "Italian Army Agile or ITA[2]" and tested it in the LC2Evo production. This methodology is currently shared with a broad community of interest including Defense Industries, Universities and the software engineers taking part in the DSSEA[3] (Defense & Security Software Engineers Association).

LC2Evo, is based on "Core Services" and "FAS" (Functional Area Services". Differently from the past, the core services are "web-based" (no more client-server architecture) and the FAS are derived by the "Mission Threads" defined in the ISAF CONOPS (Concept Of Operations) and currently adopted by NATO. Both the Core Services and the individual FAS software components, can be separately changed to accommodate the particular mission need defined by the user. At the same time all the FAS can share the data and the artifacts developed individually maximizing the "code" reuse.

The LC2Evo has been tested in a NATO exercise for the first time at CWIX 2015 (www.act.nato.int/cwix), with very favorable results.

Very synthetically, in the LC2Evo, the core services provide events integrated management on common cartography, interacting with all the relevant NATO software packages available. All the documents necessary for the command post activity are generated in collaboration, with large use of COTS product. Full integration is provided with garrison infrastructural data flow for common daily homeland base operations. Both the georeferenced military cartography and the web based commercial ones are available with all the available information (weather, traffic, etc.)

The FAS under continuous development process, but in use already are the following:

"Battle Space Management" originally designed to provide the C2 support for homeland security operations such as "Strade Sicure" is capable of tracking friendly secure mobile units on various type of cartography. It implements voice and messaging capability and has an integrated NATO JOCWHATCH feature. Most of the code and functions realized for this FAS have been reused for the LC2Evo-Infrastructure FAS [5] which provides and extensive and detailed set of functionalities needed to manage the Army real estate.

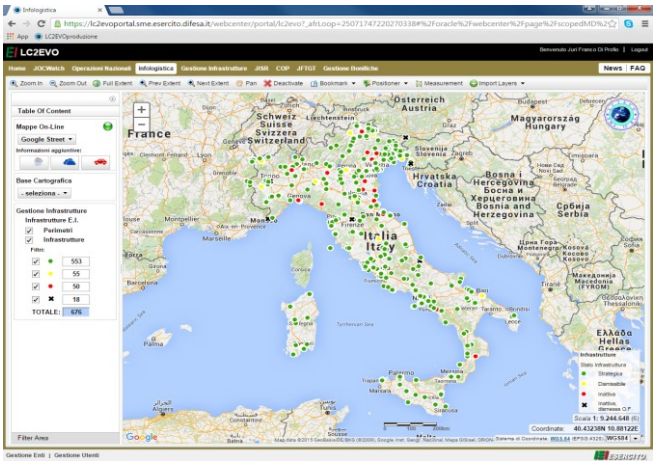

Fig1 the LC2Evo-Infrastructure FAS

This last quoted FAS, has more than one thousand registered user form more than 600 military facilities in the homeland and abroad. More "reuse" is going to happen as the realization of the " Infologistica" FAS, a strategic tool to be used to give to CINC the real time situation of all Army materials, proceeds.

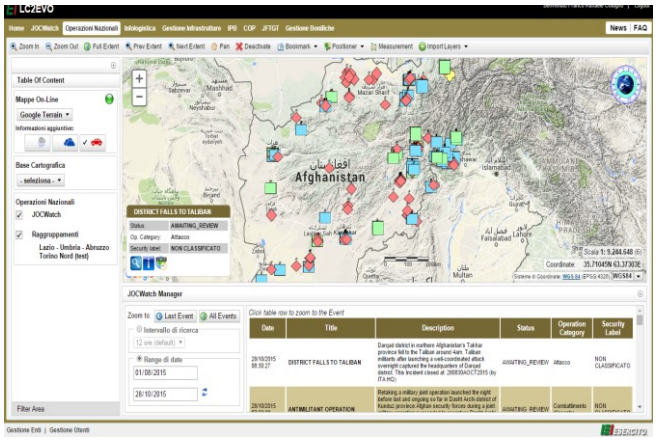

Fig2 The Battle Space Management the events screen-shot from Aghanistan

"Joint ISR" FAS, providing management and analysis functions for the Intelligence Preparation of the Battlefield.

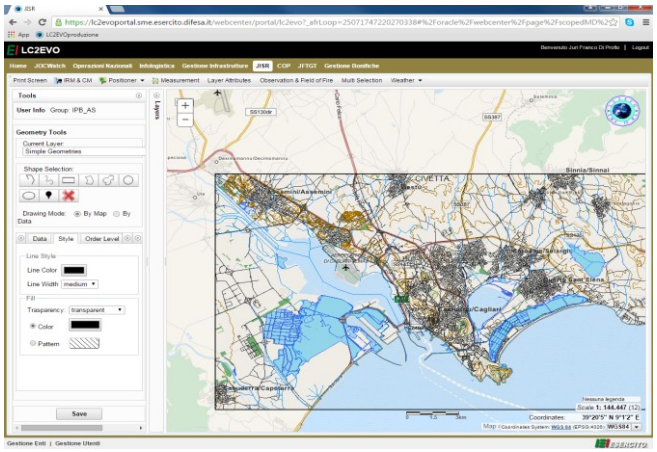

Fig 3 "Joint ISR" a screen-shot of the Intelligence Preparation of the Battlefield

"Joint Fires & Targeting" FAS, supporting all the coordination and planning activities related to the "Fire Power support" including all available effectors.

"Military Engineer & Counter IED" FAS, initially designed to support the collection and management of data about unexploded ordnance (UnExO) from WW2 found on the national territory but soon implemented to provide support to the Counter IED operations (*attack the net* and *force protection).*

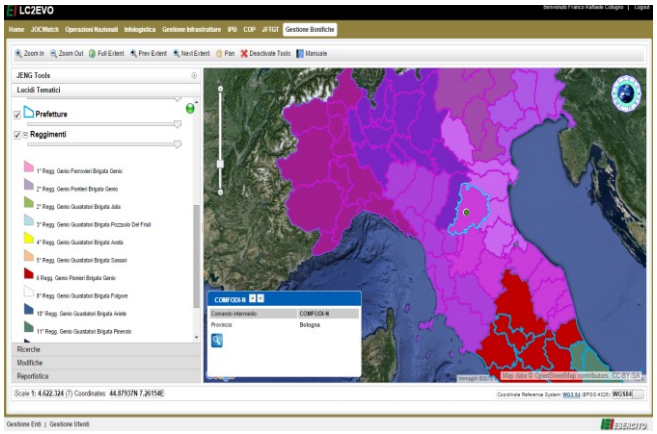

Fig 4 *"Military Engineer & Counter IED"* screen-shot of the UnExo area's of interests.

B. *The "Dismounted Soldier" segment and the augmented reality tools*

The NCI Agency and the Italian MoD [6] have been collaborating recently in developing mobile apps for the Italian "Soldato Futuro" (Future Soldier) handheld devices. The use-case for this project covers the situation when the soldier on patrol has his hands engaged while "busy" handling a weapon, and his ability to receive and send situational awareness data from his mobile device is limited. The goal is to keep a certain amount of situational awareness data available by receiving and sending this data through an augmented reality (AR) device such as Google Glasses or similar. The augmented data displayed on the device is retrieved from NATO FAS (Functional Area Services) while the data collected by the soldier is fed back to NATO FAS (and implementation of the "Soldier Sensor" concept used in the Intelligence Surveillance and Reconnaissance domain).

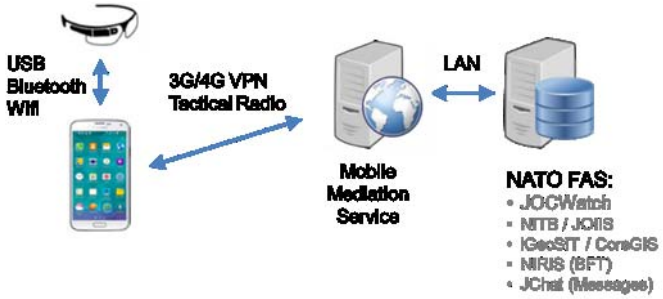

Fig. 5 HW and SW architecture of project "Eye – Catch"

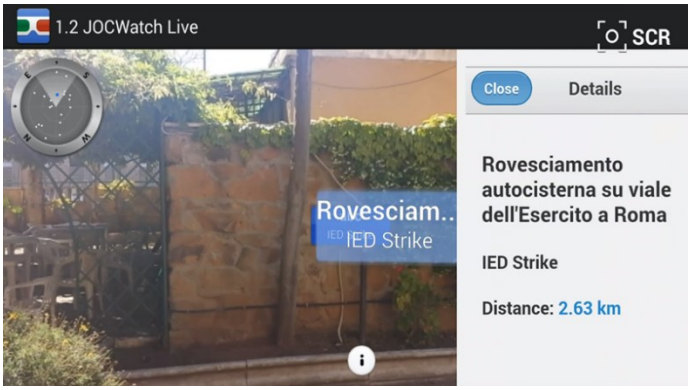

Fig. 6 NATO JOC-Watch events displayed on Google Glasses

To develop AR apps for the Italian MoD, the Directorate of Application Services (DAS) Joint ISR service line (JISR SL) started following a new software development methodology that the Italian MoD has been applying for many years: an agile methodology, but somewhat revised to encompass the mission-critical factor of the software to be developed (ITA[2]).

NATO approach to "agile": Software Intensive Project (SIP)

To emphasize the need to innovate in the area of software development, NATO nations have recently pointed out the systematic problems experienced by the Agency through the implementation of Software Intensive Projects (SIP).

In January 2014 a SIP Task Force (TF) was set up to identify means to improve the performance of software-intensive common-funded acquisition projects within NATO. A preliminary report identifying the problems was released to the Investment Committee in May 2014 and this identified different but interdependent root causes of the issues, broadly categorized as relating to requirements, regulations and risk.

- To allow focus on solutions, the TF identified six overarching principles that need to be followed in order to improve the performance of SIPs within NATO. Two of the six key principles point to the need to embrace modern and agile software development techniques.
- The first of these two key principles states the need to DELIVER EARLY AND OFTEN. This principle is aimed at changing the culture from one that is focused typically on a single delivery at the end of the development phase to a new model with multiple deliveries during development, leading to an ultimate version that supports the full set of requirements.
- The second states the NEED FOR INCREMENTAL AND ITERATIVE DEVELOPMENT AND TESTING. This principle embraces the concept that incremental and iterative development and testing, including the use of prototyping, yields better outcomes than those resulting from trying to deploy large complex IT network systems in one "Big Bang".
- These two principles tell us that the old-fashioned waterfall approach, in which the customer, after months of software development, is faced with a release he may not be happy with, needs to be replaced by more modern and innovative software engineering techniques and methodologies.

It is worth noting that in implementing these two principles additional costs should be considered on the "Purchaser" side to manage the sprints associated with frequent deliveries of partial capabilities

IV. THE BUDGET REDUCTION AND THE TRANSITION TO "AGILE"

The transition to "agile" was not only needed to accommodate quicker adaptation to dynamic mission needs change, but also mandated by a drastic reduction of the defense budgets experienced in many NATO Countries but particularly heavy in Italy.

A. *Waterfall vs Agile: cost reduction*

Very soon, as early as the sixties, it appeared impossible to run complex military operations without the use of a relevant amount of software. The first applications were born in the military and in aerospace almost at the same time.

Over the forty years from 1960 to 2000, the amount of functionality provided by software to pilots of military aircraft has grown from 8% to 80%, and software size has grown from 1000 lines of code in the F-4A to 1.7 M lines of code in the F-22. The newest fighter, the F-35 Joint Strike Fighter current version, according to one source, has 5.7 M lines of code of embarked software.

The need for a regulation in the software acquisition in the defense area generated from the necessity of risk mitigation especially for mission critical software created the "Software Quality Standards". One of the first was the well-known DOD (Department of Defense) Standard 2167A also called "the waterfall method" from the shape of its compact pictorial representation. 2167A, and its follower DoD 498, and all the others (V-Model, Spiral etc.) so called "linear" software development methods, had some limited success especially at the beginning of the software era, on the small-medium scale product where the complexity could be kept under control.

The growth of complexity and dimensions (equivalent line of source code number) of the software product has produced very articulated CASE tools and computer procedures for the production of the impressive quantity of documents needed for the contractual acceptance of the milestone baseline deliveries.

For great projects, often the effort needed to prepare the baseline documents was greater than the actual software coding and testing one. This huge effort and the CASE complication are anyhow useless to reduce the number of software errors. Many studies have demonstrated that software malfunctions are linked to intrinsic complexity only and cannot be eliminated with "quality" practices.

The "agile" manifesto in 2001 [7] changed the attitude of software engineering putting the programmers at the center of the production line and created a straight communication line between the Customer (in charge of the requirement) and the developers, minimizing the need for formal documentation.

There is, still today, a trend to consider "agile" a pretty unregulated environment but this absolutely wrong. The "agile" ritual is far more stringent and mandatory that any former standard regulated production procedure. In the Italian Army Agile and then in the DSSEA i*Agile* methodology every single step of the procedure is monitored, traced and evaluated being the programmers themselves, the first users of these monitoring tools.

B. *Building the 4 pillars*

Most of the effort performed to generate an adequate production structure for the LC2Evo, has been devolved to the setting of an innovative cultural and technical environment. Most of the difficulties found on the way of this innovation process were "human based", essentially due to cultural resistance based on consolidated practices. A whole brand-new environment had to be build. The four "pillars" of this "innovative software engineering paradigm" are:

- User Community Governance
- Innovative "agile" training
- Innovative CASE tools
- High Reliability "agile" doctrine

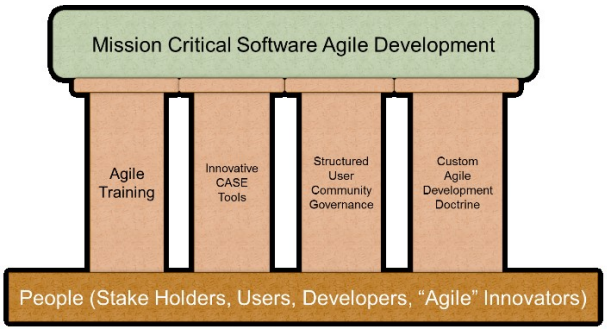

Fig. 7 The "Four Pillars" of innovation.

Not all the problems connected to these four areas have been solved. Some of the "pillars" are still under construction even if the LC2Evo production lines are fully operational and the working development teams have in time reached the number of 7.

The *User Community Governance Pillar* is of paramount importance and can be considered a pre-requisite for the entire development process. In the area of *land command &control*, the number and articulation of the reference stake-holders and users is huge. Functions such as the "Third Dimension Control" may have multiple stake-holders and users, at the same time: Artillery is using the 3D space to plan its fire power delivery, but the same space is used by the Army light Aviation and the Air Force (in joint operations). This situation makes it necessary to rationalize the "requirements" management. The Army General Staff has dedicated a huge effort to create the coordination lines and the permanent structure to allow an orderly fashion collection of user needs and provide the availability of subject matter experts to be placed in the development teams. Ad-hoc social networks have been designed for this scope.

As stated before, the "agile" training easily available from the market, could not provide the peculiar skills needed to work in the military and industry mixed multidisciplinary teams and the traditional "roles" described by the Scrum doctrine such as the "Product Owner" and "Scrum Master" had to be modified to be able to perform the Italian Army Agile methodology. Brand new training courses have been designed with the substantial help of DSSEA[3] and after an internal experimentation phase they are now available to the public as well, under the no-profit brand DSSEA i*Agile*.

Innovative CASE tools

Replacing the traditional and CASE tools poses a difficult challenge: keeping the momentum of the Agile innovation while implementing the new concepts for designing the high reliability related software development environments. The core of the Agile methods is the role of human element which is positioned at the center of the development process again, using its brain non linear capability to overcome the difficulties related to the user requirement incompleteness and volatility.

Agile methods, properly implemented, can take care of a significant part of this problem by capturing the user needs in lists of short user stories, written in natural language and confronting the user with the working segments of the product ready after few weeks or even days. This way, part of the non-linearity of the requirement conceptual design is overcome by the interaction between humans: the software developer is directly assisted by the user and they basically design together the application. In the process the two different complex 3D representations of the application (run time) imagined by the mind of the user and the one detailed by the mind of the software developer tend to converge. This method also reduces the number of translations needed to convert the requirements into coding tasks, significantly decreasing the loss of relevant information.

LC2Evo experience has fully proven the great effectiveness of the Italian Army Agile software production approach. However, the adaptation of the Agile methodology to the mission critical software production area is not finished and some problems have yet to be solved. The above described human to human non linear interaction coming from the true core of the Agile philosophy, can be very efficient but has two major critical features:

1) All requirements are captured in user stories written in natural language, but the communication in the team between users and developers is a full spectrum human communication which is only partially represented by the written stories: any user story tends to contain only part of the full background knowledge of the related requirement. This is not acceptable in a controlled environment where the requirement traceability is a must.

2) The use of natural language to write down the user requirements using the "agile" user stories product backlog fashion, mandates absolute freedom for the user even if a simple sentence template is used: " (as____I want to_____in order to____)" as the base for the user interview. It is almost unavoidable to introduce some linguistic uncertainty in the process [8].

As stated previously, many of the problems of adapting Agile to a high reliable software production process have been faced by the ITA Army Agile framework but something is still missing. The basic question is: what tools and procedures can be adopted in this new concept, to substitute the old fashioned software factories tools? What is the right amount of procedural elements that can be introduced in the agile teams management, without reducing the effectiveness of the new approach? Are there any available software technologies which could be effectively used to solve the problem of linguistic uncertainty?

Research is being planned to tackle these issues.

High Reliability "agile" doctrine

The last but not the least of the "Four Pillars" is the definition of a doctrine, a set of rules and procedures encompassing all the needed practices and artifacts to be used in the implementation of the new "agile" methodology. The Army General Staff Logistic Department has generated an internal document (SME L007) which acts as an interim guideline in

preparation of a Joint General Staff regulatory action. Similarly to what the U.S.A. D.O.D. has done with the DOD 5000.02 [9].

DoD Instruction 5000.02 (Dec 2013) heavily emphasizes tailoring program structures and acquisition processes to the program characteristics. Agile development can achieve these objectives through:

- Focusing on small, frequent capability releases
- Valuing working software over comprehensive documentation
- Responding rapidly to changes in operations, technology, and budgets
- Actively involving users throughout development to ensure high operational value

These indications are a clear encouragement to use "agile" practices to integrate planning, design, development, and testing into an iterative lifecycle to deliver software at frequent intervals. In "agile" developers can demonstrate interim capabilities to users and stakeholders monthly.

C. Results achieved

After two years in the process the Italian Army General Staff, the prime contractor Selex-Es and the subcontractors (Engineering, Oracle, Profesia, and others) have clearly demonstrated the effectiveness of the new software development methodology realizing the LC2Evo Command & Control software. The product is a continuous development effort which continues to produce new segments every five weeks. The first FAS of the product, the LC2Evo-infrasture is on line since June 2015 and is serving more than one thousand users daily and has registered customer satisfaction levels close to 100%. This first segment of LC2Evo "full release", was delivered only 8 months after the first line of code had actually been written. The FAS is still growing and changing in accordance with the user community requests.

All the other FAS are in continuous development as well and they are currently used in operations in the homeland and abroad by the Army units.

The assessed product cost per "line of code equivalent" has dropped by 50%.

V. Conclusions

The evolution of the Italian Army land command and control system into the LC2Evo has clearly demonstrated that the traditional software engineering paradigm is clearly outdated, as some authors have clearly pointed out. [10]

The Italian Army experience in developing the Command & Control software LC2Evo, confirms that the major problem in dealing with complex scenarios and rapidly evolving user needs is the management and evolution of the user requirements. The linear development cycles focus on the production process, once the requirement is consolidated but even in the defense and security software applications area, this is no longer possible. Similarly to the commercial world, the military applications resent of a quickly changing operational environment. The high reliability, security and quality of the product cannot be pursued with procedures and agents working outside the development cycle but have to be embedded in the production cycle.

The commercial "agile" is a good chance to introduce structural changes to the software engineering paradigm, but the need to keep some of the old fashion procedures concerning the formal configuration control and the traceability of the requirements has to be blended with the "agile" principles. The Army and DSSEA iCASE, is an effort to create a controllable but flexible environment to support synchronized scrum-like agile teams during the software development cycle in a non-invasive but effective way.

Some area of iCASE have to be investigated and some research has to be run on the most sensitive topics such as requirements management and innovative software production methods. These key challenges have been presented to the EU DG-Connect and have been recently reported in the H2020 ICT WP 2016-2017.

Relevant changes are necessary in the procurement administrative procedures as well, to accommodate the quick delivery of by-products provided by the "agile" production cycles that could be as shorts as few weeks. These changes are under elaboration at least in two countries: USA and Italy. [11]

a.

Acknowledgment

The Authors wish to thank the Italian Army General Staff Logistic Department for the availability of data, the courtesy of some pictures and in particular: Major General Francesco Figliuolo, Department Chief, his predecessor Major General Mario Ruggiero and the Operational Information System Office Chief, Col. Franco Cotugno. The support of many DSSEA members significantly helped in refining the content of this article. In particular the authors would like to mention: Professors Paolo Ciancarini, Giancarlo Succi and Alberto Sillitti. Special thanks to Dr. Cristian Coman and Mr. Pascal van Paridon - Directorate Application Services - Joint ISR for the technical support in the development of Augmented Reality app section.

References

[1] "Research Report Lessons Learned from the Afghan Mission Network - Developing a Coalition Contingency Network" Chad C. Serena, Isaac R. Porche III, Joel B. Predd, Jan Osburg, Bradley Lossing © Copyright 2014 RAND Corporation

[2] "Adapting SCRUM to the Italian Army: Methods and (Open) Tools (A. Messina, F. Cotugno)" The 10th International Conference on Open Source Systems San Jose, Costa Rica, May 6-9, 2014

[3] "Software characteristics for Program Forza NEC main systems Angelo Gervasio" - Proceedings of 4th International Conference in Software Engineering for Defence Applications, 978-3-319-27894-0, 339732_1_EN

[4] “Implementing SCRUM in the Army General Staff Environment(F. Cotugno, A. Messina)” The 3rd International Conference in Software Engineering for Defence Applications – SEDA Roma, Italy September 22-23 2014

[5] “ITA Army Agile Software implementation of the LC2EVO Army Infrastructure strategic management tool” Cristina Ventrelli, Dino Trenta, Davide Dettori, Vittorio Sanzari, Stefano Salomoni - Proceedings of 4th International Conference in Software Engineering for Defence Applications, 978-3-319-27894-0, 339732_1_EN

[6] “Consumer Augmented Reality in Defense Applications” Cristian Coman, Pascal van Paridon and Franco Fiore - Proceedings of 4th International Conference in Software Engineering for Defence Applications, 978-3-319-27894-0, 339732_1_EN

[7] . “Agile Project Management with SCRUM” K. Schwaber Microsoft Press, ISBN 9780735619937, 2004

[8] Foundations of language R. Jackendoff. Oxford University Press, 2003. ISBN: 9780199264377

[9] “Defense Agile Acquisition Guide” Pete Modigliani, Su Chang, MITRE Corporation 2014, 14-0391

[10] “New Software Engineering Paradigm Based on Complexity Jay Xiong. Science- Springer ISBN 978-1-4419-7325-2”]

[11] “How Agile Development Can Transform Defense IT Acquisition” Angelo Messina, Peter Modigliani and Su Chang Proceedings of 4th International Conference in Software Engineering for Defence Applications, 978-3-319-27894-0, 339732_1_EN