19th ICCRTS

Title of Paper: Interoperable Architecture for Command and Control

Topics: Topic 1: Concepts, Theory and Policy

Topic 2: Approaches and Organisations

Name of Authors:

Cobus Venter (CSIR, South Africa; jpventer@csir.co.za);
Corné Smith (CSIR, South Africa; csmith@csir.co.za);
Rudolph Oosthuizen (CSIR, South Africa; roosthuizen@csir.co.za);
Arno Duvenhage (CSIR, South Africa; aduvenhage@csir.co.za);
Brian Naude (CSIR, South Africa; bnaude@csir.co.za);
Willem H le Roux (CSIR, South Africa; whleroux@csir.co.za);

Point of Contact: Cobus Venter (CSIR, South Africa; jpventer@csir.co.za);

Name of Organization: Council for Scientific and Industrial Research
PO Box 395
Pretoria
0001
info@csir.co.za

Abstract:

Current defence environments are characterised by unconventional, asymmetric threats and unpredictable enemies. As a result this has necessitated greater defence requirements for flexibility and interoperability of Command and Control systems to support ad hoc and reconfigurable mission requirements.

These requirements have created the need to re-evaluate defence system architectures in order to accommodate net-centricity, rapid re-configurability, greater information exchange and interoperability.

This paper proposes a gateway information link processer approach to establish a flexible Command and Control system architecture that will support the Process, Applications, Infrastructure, Data (PAID) model of the Levels of Information Systems Interoperability (LISI) framework defined by the C4ISR Architecture Working Group (AWG).

This paper provides experimental outcomes where this type of architecture has been used to increase Command and Control systems interoperability, flexibility, deployability and configurability in order to support changing defence environments while reducing systems integration costs.

Report Documentation Page	*Form Approved* *OMB No. 0704-0188*

1. REPORT DATE **JUN 2014**	2. REPORT TYPE	3. DATES COVERED **00-00-2014 to 00-00-2014**

4. TITLE AND SUBTITLE **Interoperable Architecture for Command and Control**	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Council for Scientific and Industrial Research,PO Box 395,Pretoria 0001 South Africa,**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

Presented at the 18th International Command & Control Research & Technology Symposium (ICCRTS) held 16-19 June, 2014 in Alexandria, VA. U.S. Government or Federal Rights License

14. ABSTRACT

Current defence environments are characterised by unconventional, asymmetric threats and unpredictable enemies. As a result this has necessitated greater defence requirements for flexibility and interoperability of Command and Control systems to support ad hoc and reconfigurable mission requirements. These requirements have created the need to re-evaluate defence system architectures in order to accommodate net-centricity, rapid re-configurability, greater information exchange and interoperability. This paper proposes a gateway information link processer approach to establish a flexible Command and Control system architecture that will support the Process, Applications, Infrastructure, Data (PAID) model of the Levels of Information Systems Interoperability (LISI) framework defined by the C4ISR Architecture Working Group (AWG). This paper provides experimental outcomes where this type of architecture has been used to increase Command and Control systems interoperability, flexibility, deployability and configurability in order to support changing defence environments while reducing systems integration costs.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**25**	

Introduction

Legacy defence systems have been characterised by requirements for operationally customised solutions and stringent logistics to ensure system availability, reliability and safety.

In contrast, current defence requirements focus on the use of multi-role platforms that should be adaptable for threats that are unpredictable and ever changing. The employment of these platforms is also expected to support joint, inter-departmental and multi-national operations.

Current systems engineering approaches have however been optimised for the customisation of systems for specialised operations. In many instances the incorporation of interoperability, flexibility, modularity and portability requirements come at the cost of customisation.

In addition, defence forces are setting goals to achieve reduced system procurement costs while significantly decreasing turnaround times between change requests and systems delivery.

These trends have forced defence forces to re-think the way defence systems are developed. A move towards establishing common building blocks between procurement projects are evolving. These building blocks have the potential to reduce project costs, reduce duplication, support interoperability and reduce development timescales.

Such building blocks are expected to conform to a common architecture. It has been envisaged that a Common Information Exchange Architecture (CIEA) could expedite the integration of systems into a common communications network and alleviate the need for different projects to define their own communications networks and information exchange mechanisms.

Current engineering approaches (Customisation)

Traditional systems engineering approaches have prescribed the decomposition of requirements when developing complex systems. As a result, the construction of such systems generally also relate to the same decomposition philosophy. This has been done in order to manage the complexity of integrated systems and to allow the configuration control of requirements for contracting purposes. The placement of defence acquisition contracts is thus also aligned with such approaches.

Due to these approaches a generalised systems hierarchy has evolved to support the contracting and development of defence solutions. Figure 1 illustrates such a generalised systems hierarchy.

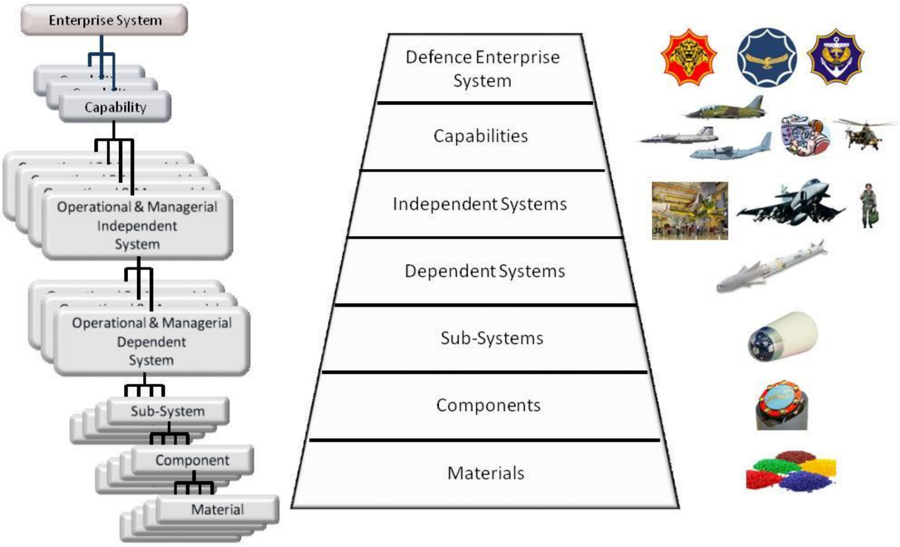


Figure 1: Generalised Defence Systems Hierarchy

For this systems hierarchy, the following definitions are applicable;

System: A system is an integrated and interacting set of elements that accomplish a defined objective. Elements can include other systems, people, processes, technology and other support elements (Adapted from [9]).

Enterprise System: An enterprise is an intentionally created entity of human endeavour with a certain purpose. An enterprise could be considered a type of system [7]. In this case the enterprise is a Defence Enterprise System required by government as a tool to maintain national sovereignty.

Capability: Capability is the ability to do something [7]. A capability is defined by operational attributes (what it can do). Capability is the ability of the enterprise to satisfy a distinctive competence or operational need through services being provided by one or more independent systems. A capability system is a type of system that utilises a combination of independent systems to provide an ability to operationally do something that is consistent with the desired attributes of the defined capability.

Independent System: An independent system is a type of system that is distinguished by two properties;

Operational Independence: Each independent system is able to fulfil customer-operator purpose on its own, but is not limited to independent operation. Each independent system should be able to operate in collaboration with other independent systems to facilitate the creation of capability systems.

Managerial Independence: Each independent system is separately acquired and developed and is maintained independently of other systems.

Dependent System: A dependent system is a type of system that is distinguished by two properties;

Operational dependence: Each dependent system could be dependent on other systems to fulfil customer-operator purpose. On its own, a dependent system cannot provide customer-operator purpose but contributes towards customer-operator purpose in conjunction with other systems.

Managerial dependence: Dependent systems are acquired and integrated as part of the creation of an independent system and cannot maintain continues operational existence without other systems.

Sub-system: A dependant system comprises a number of sub-systems. Sub-systems can be acquired from different vendors and need to be integrated to create a given dependant system.

Sub-systems comprise of components and components are made from materials.

The decomposition philosophy and contracting model illustrated in Figure 1 has not supported defence forces to allow the contracting or acquisition of defence capabilities, but has at best supported the acquisition of independent systems. This has lead to a silo approach to independent systems development, hampering efforts to establish interoperability, flexibility, resilience and adaptability towards defence capabilities.

The aim of establishing a CIEA is predominantly focused around solving information interoperability between independent systems and allowing flexibility for the creation of new capabilities within short timescales. This architecture will aim to provide an information exchange mechanism, based on standardised building blocks that could become compulsory for the acquisition of new independent systems. These building blocks shall be in the format of sub-systems (as per Figure 1).

An industry example of such an approach would be the Apple iPad. This device has been able to allow numerous applications to run on a platform that has the ability to connect numerous devices and applications via commercial networks and the internet. These communications networks are seamlessly abstracted from application developers, who simply use these standardised interfaces to create their applications.

Common architecture development approach

It should be noted at the start that the establishment of a Common Information Exchange Architecture (CIEA) is only applicable for Command and Control information exchanges (sensor, decision and affecter) in support of tactical defence operations.

Interoperable architecture development requirements have become more prevalent since the beginning of the new millennium. Approaches provided from the US DoD C4ISR architectures framework workgroup in the late 1990s are still valid and have been utilised as a starting point from which to define a Common Information Exchange Architecture (CIEA).

From the US DoD C4ISR architectures framework workgroup [2], the PAID (Procedures, Application, Infrastructure and Data) attributes have been adopted. These attributes have been defined within the Levels of Information Systems Interoperability (LISI) maturity model and aims to facilitate the establishment and measurement of interoperability.

The PAID attributes have provided valuable insight as to the aspects that need to be addressed when contemplating an interoperable CIEA. The PAID attributes are illustrated in Figure 2.

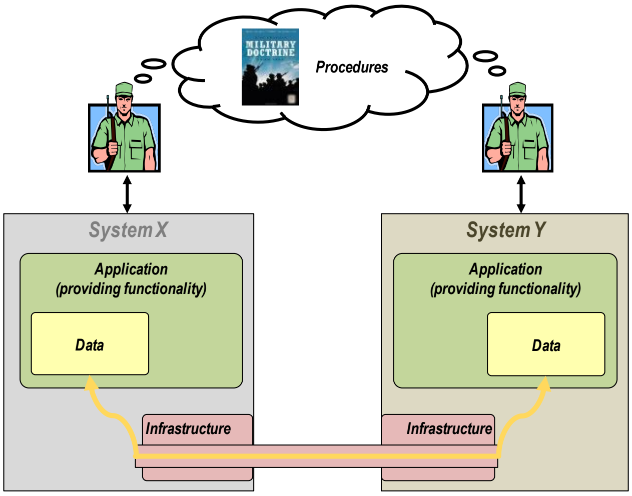


Figure 2: PAID Interoperability attributes

In order to establish interoperable systems it should firstly be considered how to provide connectivity through common infrastructure. This includes aspects like communications protocols and Radio Frequency (RF) technologies.

Once connectivity has been achieved, it should be considered if data can be exchanged in such a way that systems have a common way to understand and exchange data. This requires the establishment of common data formats.

Once data can be transferred between systems it should be considered if applications should interpret the data in the same way. Applications must express common meaning and interpretation of data and provide common awareness to the operator.

Once operators can exchange the awareness of information, their interactions, inputs, behaviour and decision making should be governed by processes, procedures or military doctrine that corresponds with their operational functions and level of Command and Control (C2).

The proposed CIEA aim to provide an architecture which can facilitate seamless and interoperable integration of applications (C2, sensors and tactical platforms) by providing managed communications infrastructure, common data meaning and providing syntactic meta data to allow applications to exchange awareness of operational context. It must be kept in mind that the intention is not that the CIEA shall dictate approaches to doctrine or procedures that are employed by war fighters.

Common Information Exchange Architecture

The aim of the CIEA is to get the most useful information to the right C2 application in time for war fighters to make decisions that will enable an appropriate action to be executed on the battlefield. In this respect the concepts of the Observe, Orientate, Decide and Act (OODA) loop and Net Centric Warfare are applicable [12].

It thus become apparent that the data required for such decision making shall have time dependencies. In other words, if data that is provided to a C2 decision maker is older than needed to support proactive influence on the battlefield, this data becomes useless.

We can thus propose the following battlefield C2 variables that will be applicable to the definition of the functional requirements of the CIEA;

Age of data: This is the difference in time between when a data artefact has been created and when this piece of data has been delivered to a war fighting decision maker in a format as to allow decision making.

Data delivery in time for impact; This is the difference in time between when a tactically useful data artefact has been created and when a final decision is required by a war fighting decision maker to allow the execution of an order that will exploit this data artefact to proactively influence a battlefield situation.

The following definitions of the states in which data can be managed also becomes applicable. Within computer system applications, data can be found in any one of three states;

Data-in-use: Refers to data movement stemming from actions taken by end users on their workstations, whether that would entail manipulating data, copying data, sending information, viewing information or even cutting and pasting between applications.

Data-in-rest: Refers to inactive data stored in physical data bases, data warehouses, spreadsheets, archives, tapes, off-site backup facilities etc.

Data-in-motion: Refers to data that is traversing a network or temporarily residing in computer memory to be read or updated for use.

When these states of data are contextualised within the requirements for war fighting decision making, the following observations can be made.

Time dependencies for data-in-use depends on the user of the data. C2 applications should thus be developed to optimise the interactions of war fighting decision makers and to ensure minimum effort is required to manipulate data for decision support.

When,

$$\frac{Age\ of\ data}{Data\ delivery\ in\ time\ for\ impact} \geq 1,$$

data is too old for useful C2 decision making.

When,

$$\frac{Age\ of\ data}{Data\ delivery\ in\ time\ for\ impact} < 1,$$

data will be in time for useful C2 decision making.

When data artefacts are not tactically useful they will not be have battlefield impact and should not clutter networks that are sending useful data. The age of irrelevant data artefacts are thus not important.

Data-in-rest is considered to be mostly historical data and is expected to have less tactical battlefield impact than near real-time data artefacts being produced by sensors. It is thus expected that for data-in-rest;

$$\frac{Age\ of\ data}{Data\ delivery\ in\ time\ for\ impact} \gg 1.$$

Data-in-motion is considered to be near real-time or timiously tactically useful data where;

$$\frac{Age\ of\ data}{Data\ delivery\ in\ time\ for\ impact} < 1$$

It can this be concluded that the CIEA should provide the seamless integration of Command and Control (C2), sensor and tactical effecting information between defence platforms on a data-in-motion basis. This means that C2, sensor and platform information maintains behavioural-state dependence in a near real-time fashion as information is exchanged between applications in a services oriented architecture over communications infrastructure. The CIEA is thus created for exclusive tactical battlefield information exchange.

Data Model:

In order to achieve data-in-motion a standardised data model with data exchange rules are required between applications. This data model is expected to comprise standardised information formats to support low bandwidth tactical networks and shall allow the inclusion of semantic (meaning) information to support applications in requesting information services and serving other applications as information clients.

To standardise the semantic information, it shall be required to establish a common “tag” library to support the data model. “Tags” shall be used as hidden labels to data that provides additional structure to data, but does not convey any meaning on their own.

In addition a Resource Description Framework (RDF) shall be required to create meaning to the data by classifying tags as subjects, verbs or objects. This RDF shall allow the development of information decision rules. An RDF structure shall aim to allow data to become information where “things” (platforms, entities, people) have “properties” (are part of, is associated with, etc) with a certain “value” (a combat group, an expected arrival, etc).[4]

In certain cases some of the “tag” associations might be assigned differently by different applications, but still have the same meaning. For this purpose, a common ontology could be required to list common “tags”. Such an ontology shall provide a shared vocabulary, which makes it easier to publish and share data [3]. Currently the need for an ontology is not fully explored.

Due to the data-in-motion nature of the CIEA, the data model should be supported by a distributed software architecture. Since limited connectivity bandwidth is expected to govern information flow, the "tag" libraries and RDF associations with common ontology shall have to be managed at each node or application.

In order to ensure communications integrity between applications in the tactical environment, two means of dedicated communications busses are envisaged.

Combat Communications Bus:

The Combat Communications Bus is envisaged to support connectivity and services between C2, sensor and platform applications in a localised environment (e.g. per combat grouping). This can be compared to a Local Area Network (LAN). This type of connectivity can be provided via tactical data links, tactical satellite connections, militarised wireless LAN technologies, mobile communications networks or commercial communications options as illustrated in Figure 3.

The Combat Communications Bus (CCB) shall support data-in-motion services for applications and shall require "clever" routing functions to deliver information via different connectivity technologies to different applications. The CCB shall also provide backhaul connectivity to the Static Communications Bus (SCB) through one or more of the connectivity mediums.

"Clever" routing can be described as the ability of an information routing function to determine the most optimal communications infrastructure to be used to exchange information between two applications. This function will have to perceive the different communication infrastructure mediums connected to it and will have to be aware of all other applications connected to each of these communication infrastructure mediums. It should be kept in mind that each communication infrastructure medium has its own network layer functions and that it is still to be established if the "clever" routing function shall require awareness of routing decisions taken by the network layer of each communications infrastructure medium.

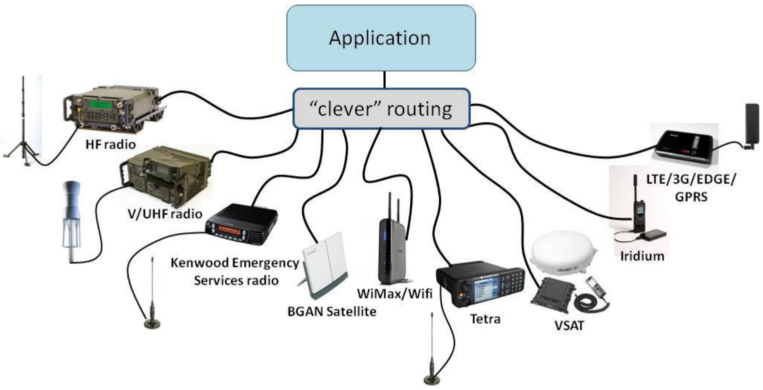


Figure 3: Combat Communications Bus

Static Communications Bus:

Normally the day-to-day logistics, human resources and financial systems and services of the organisation reside on a static communications network. Applications for operations at strategic levels are normally associated with static Head Quarters (HQs) that also utilises such networks.

The Static Communications Bus is envisaged to utilise the static communications networks that normally could consist of packet switching networks comprising static coaxial, twisted pair or fiber optic cable infrastructure supported by Ethernet link layer and Internet Protocol (IP) network layer technology. Most commonly Transport Communications Protocol (TCP) is used for transport layer support. These types of networks can also support Virtual Private Networks (VPNs) and/or Virtual Private LAN Services (VPLS) to create dedicated data-in-motion Static Communications Busses for tactical information.

The logistics, human resources and financial applications running on static communications networks do however adopt data-at-rest operational philosophies by accessing information from servers, data stores or other archives. A data-in-rest gateway shall allow specific data-in-rest information from servers/data bases etc. to be translated in the common data model format and passed to the CCB and vice versa allow specific data-in-motion artefacts to be translated and provided to logistics, human resources and financial applications and servers/data bases etc. on the SCB.

Applications at static operational HQs shall however also require data-in-motion information and shall require to have access to the Static Communications Bus to access this tactical information. The Static Communications Bus shall also manage the routing of data-in-motion between Combat Communications Busses of disconnected combat groupings as illustrated in Figure 4.

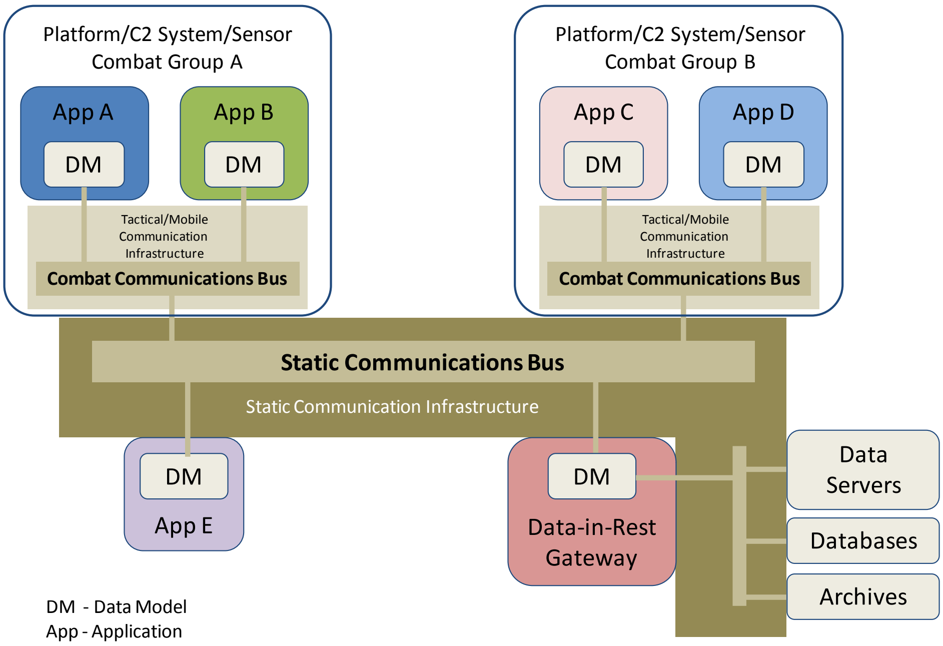


Figure 4: Common Information Exchange Architecture

CIEA and the Joint Consultation, Command and Control Information Exchange Data Model

The Multilateral Interoperability Programme (MIP) manages the Joint Consultation, Command and Control Information Exchange Data Model (JC3IEDM) on behalf of a number of member countries [11]. The data model formalises information exchange in a multinational or coalition environment, but can also be used as a national data model by a country. The model is applicable to tactical, operational and strategic information exchange and comprises of formal standards, specifications and procedures, but primarily supports the Land operational user in a Joint environment. The model is extended to cater for the maritime and air communities.

JC3IEDM could potentially be an option as the formal data model specification for a CIEA. JC3IEDM is not specific to a particular country's doctrine, and is promulgated by the North Atlantic Treaty Organisation (NATO), although MIP is not a NATO organ. Figure 5 depicts the MIP interfaces in a multinational environment. MIP does not explicitly distinguish between data-in-rest, data-in-motion or data-in-use, which makes it difficult to contextualise for this role.

An important principle, necessary in a multinational environment, is that one system can never delete data in another system. Only revisions or updates to data can be issued. Whereas the CIEA only specifies a tag system, RDF and rules for its data model, MIP relies on a complete, formal description with managed revisions and extensions. Routing of information between MIP compliant applications are based on static information exchange "contracts" and is not dynamically modifiable.

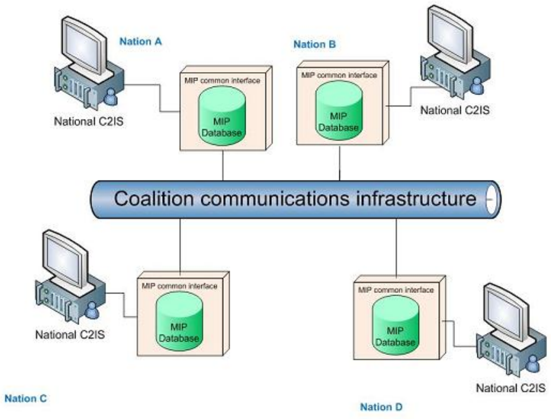


Figure 5: MIP Multinational Architecture [11]

CIEA concept experimentation

A gateway concept demonstrator has been developed to perform experimentation to validate numerous aspects of the proposed CIEA.

The concept demonstrator has been forged on behavioural simulation and modelling software technology and has allowed experimentation with aspects such as distributed information exchange, service oriented approaches and data model options.

The gateway approach was used in order to allow the use of real-world and experimental military applications to test the CIEA concept. The gateway could allow the definition of futuristic CIEA interoperability building blocks as well as support the DoD migration to a tactical CIEA.

The use of a Combat Communications Bus was simulated by fixed routing options through tactical data links, emergency communications radios and commercial communications equipment. The use of a Static Communications Bus was also simulated through the use of dedicated static communications infrastructure with fixed routing. The major aim of the experiment was however focussed on data model experimentation.

The Gateway concept

By employing the gateway concept, integration of real-world applications to an experimental data model could be achieved. Gateways could be run in a distributed computing fashion employed on a fixed Combat Communications Bus or Static Communications Bus.

Experimentation with the gateway allowed comparative CIEA experimentation as illustrated in Figure 6.

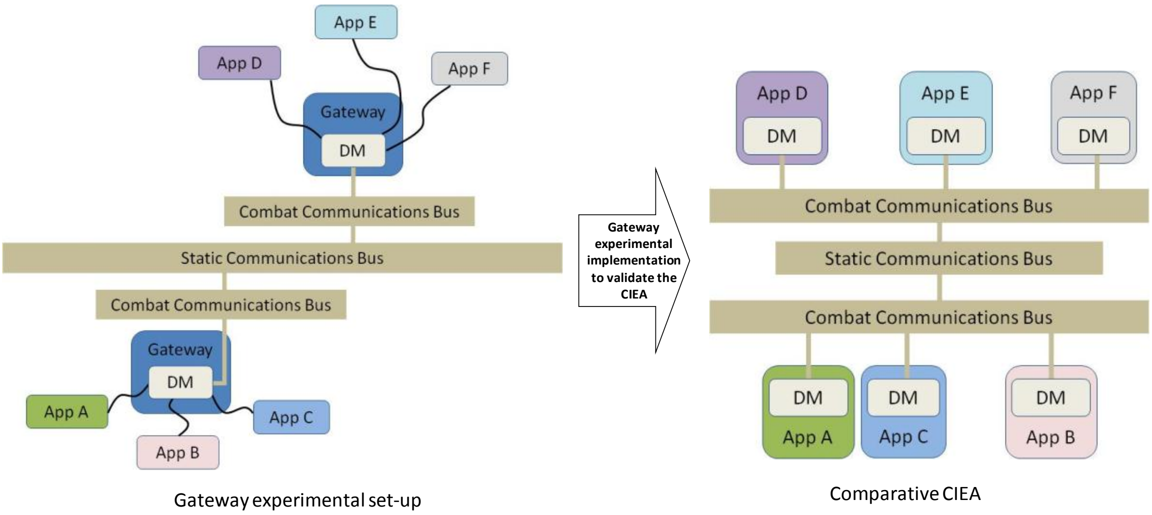

App E
App D
App F
Gateway
DM
Combat Communications Bus
Static Communications Bus
Combat Communications Bus
Gateway
DM
App A
App C
App B
Gateway experimental set-up
Gateway experimental implementation to validate the CIEA
App D
DM
App E
DM
App F
DM
Combat Communications Bus
Static Communications Bus
Combat Communications Bus
DM
App A
DM
App C
DM
App B
Comparative CIEA

Figure 6: Gateway CIEA experimentation

The gateway is built using multi-threaded, loosely coupled, extendable services which communicate using a generic object-oriented data model. The experimental gateway architecture utilised can be summarised as described below and illustrated in Figure 7;

The experimental gateway is built with a Publish-Subscribe type middleware and the gateway links, routing, filtering and user interface all run as loosely coupled services on the middleware. The middleware is time-stepped (i.e. all services running on the middleware run in discrete time steps) which provides a shared time reference between all the services. The shared time reference supported the creation of data-in-motion logs for playing back information and for integrating with simulation models and simulators.

The gateway services communicate over the publish-subscribe middleware using objects from an object-oriented data model. The data model is based on a national propriety tactical message model, but has been cleaned up to have consistent units and conventions for all object attributes. The data model is also extendable: objects can be added or extended using inheritance – the data model object attributes have already been extended to accommodate many non-national message model compliant platforms and legacy platforms. The goal is to create a generic internal data model with one addressing scheme that can be used for all supported externals platforms.

The gateway can support any number of links to externals platforms and systems. The links translate between the data model of the gateway and the message format of the external platform. The links also take care of the interfacing and any additional message framing required to communicate with the externals platforms. The external system address information differently and the links also have to translate between the unique addressing schemes of the externals systems and the addressing supported by the gateway data model. The links all run independent of each other on separate threads – slow interfaces and message coding does not affect the gateway performance.

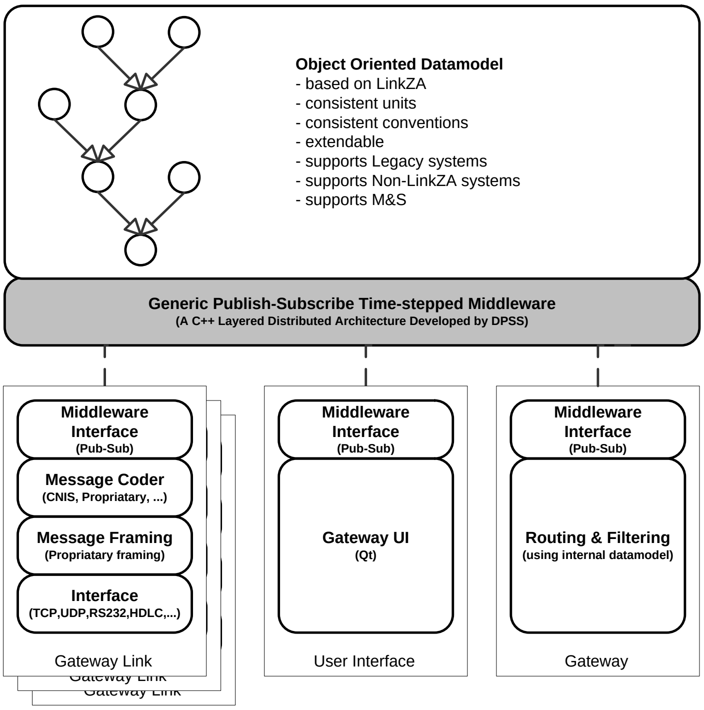

Object Oriented Datamodel
- based on LinkZA
- consistent units
- consistent conventions
- extendable
- supports Legacy systems
- supports Non-LinkZA systems
- supports M&S
Generic Publish-Subscribe Time-stepped Middleware
(A C++ Layered Distributed Architecture Developed by DPSS)
Middleware Interface (Pub-Sub)
Message Coder (CNIS, Propriatary, ...)
Message Framing (Propriatary framing)
Interface (TCP,UDP,RS232,HDLC,...)
Gateway Link
Middleware Interface (Pub-Sub)
Gateway UI (Qt)
User Interface
Middleware Interface (Pub-Sub)
Routing & Filtering (using internal datamodel)
Gateway

Figure 7: Experimental Gateway Architecture

The gateway routes specify how data model objects flow between the gateway links. The routes can filter based on object type or specific object attributes. The routes can also modify the type of an object or modify the attributes of an object. The gateway routes can for example filter sensor tracks based on speed, altitude, classification, etc. or modify the source and destination attributes of an object as it passes through. There is no limit on the number of routes.

All of the gateway functions can be controlled from a user interface built using the Qt framework. The user interface runs as a service on the middleware and interacts with the other services through the data model. The user interface is however optional and the gateway can run without the user interface if required. This feature is very useful when the gateway has to run with a small footprint, when resources are low, or when a high level of reliability is required.

The individual links as well as the middleware have demonstrated a high level of performance and robustness during past exercises and military operations.

CIEA Experiment

An experiment was conducted to validate the proposed architecture. This experiment was based on a defence force exercise and included information exchange between four physically dispersed sites [7]. The operational view of the experiment can be illustrated as shown in Figure 8.

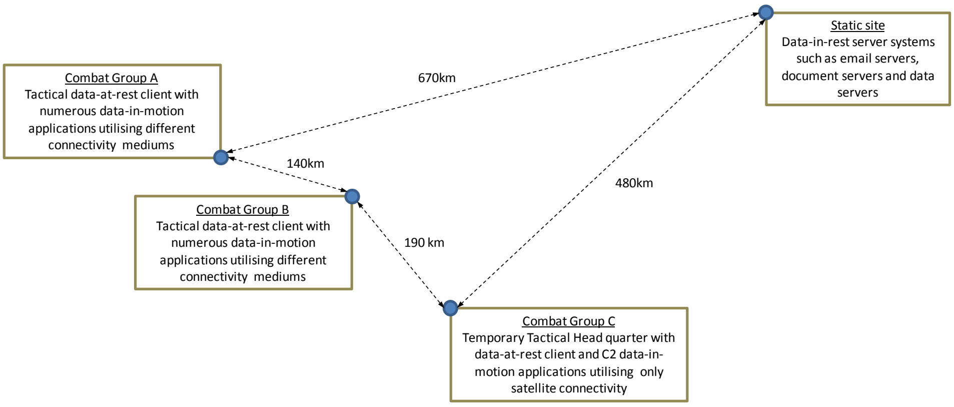

Static site
Data-in-rest server systems such as email servers, document servers and data servers
Combat Group A
Tactical data-at-rest client with numerous data-in-motion applications utilising different connectivity mediums
670km
140km
480km
Combat Group B
Tactical data-at-rest client with numerous data-in-motion applications utilising different connectivity mediums
190 km
Combat Group C
Temporary Tactical Head quarter with data-at-rest client and C2 data-in-motion applications utilising only satellite connectivity

Figure 8: Operational View of experiment

The applications, gateways and communications infrastructure utilised for this experiment is illustrated in Figure 9. It has been indicated which applications utilise data-in-motion and which applications utilising data-in-rest. It should be noted that for the experiment the data-in-rest gateway allowed translation of data-in-rest information (email server, time sync server and document server) to the common data model format when needed.

The data-in-rest gateway was also responsible to translate information from data-in-motion sources providing information via internet into the common data model format. An example would be GPS trackers sending information via GPRS/3G and the internet (ADSL) to populate this information in the data-in-rest gateway. Once the data-in-rest gateway has translated this information into the common data model, the information was distributed from the static site to combat groups.

Data-in-rest services (email server, time sync server and document server) and data-in-motion services (gateways) where connected to the static communication infrastructure and no dedicated service was created to separate data-in-motion and data-in-rest as indicated in the CIEA experimental system view (Figure 4).

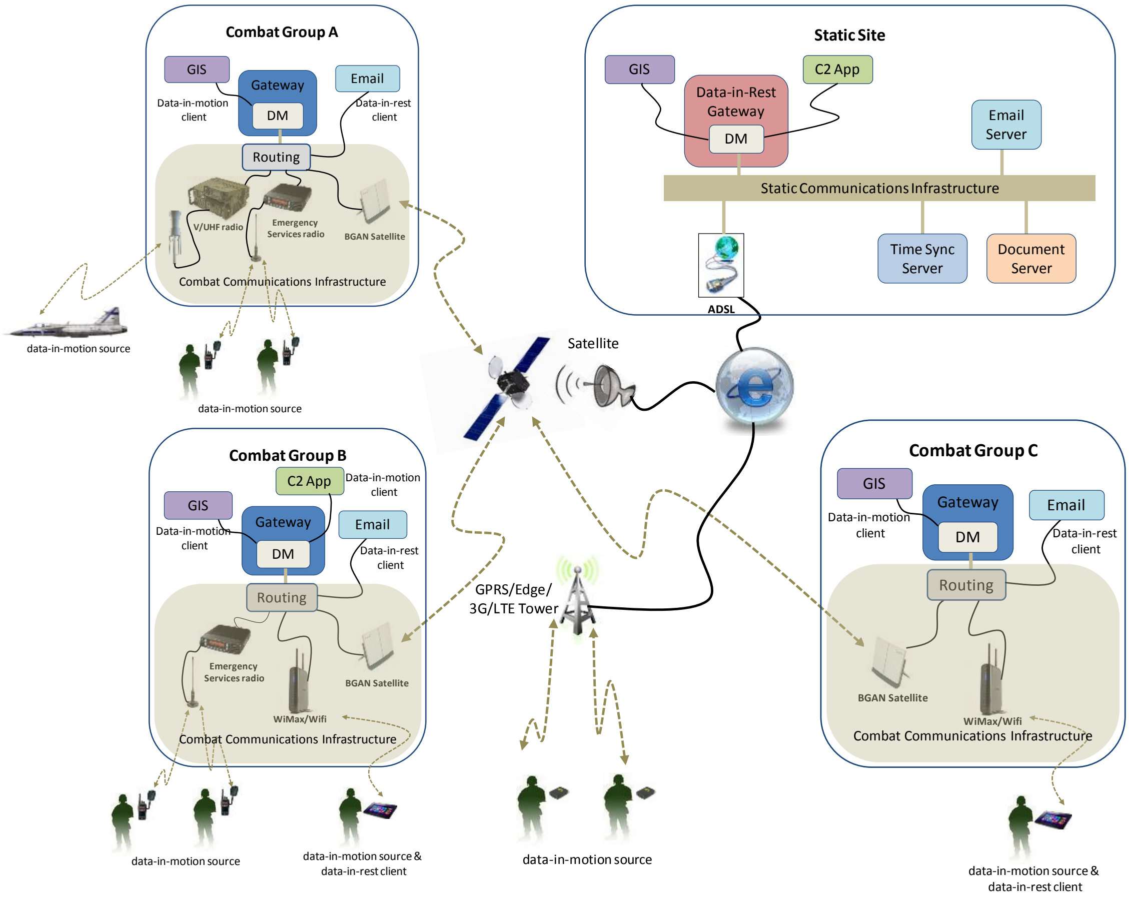


Figure 9: System view of experiment

CIEA experiment results

The experiment has shown the following promising results;

Through the use of a common data model, numerous data-in-motion sources can be integrated into the combat and static communications bus. Such integration is expedited by the use of the common data model, since it requires once-off integration at any node of the network. This approach shall allow significant improvement of interoperability management for next generation applications.

Through the use of a gateway concept, the integration of legacy systems can be accommodated.

The proposed architecture ensures flexibility to support the deployment of different organisational or operational structures.

Data filtering and decision rules can be applied to information.

When necessary the data-in-motion of critical events and operations can be recorded and saved in a data-in-rest database for evaluation and analysis of operations.

The experiment has highlighted the following future work that is required;

The management and establishment of combat communications bus and static communications bus Virtual Private Networks (VPN) should be explored and automated.

Security architecture should be developed to ensure the security for the flow of information on and between the CCBs and SCB. This should include information security via the internet.

"Clever" routing mechanisms and rules should be investigated to optimise data bandwidth over different connectivity mediums.

Data Link Processor concept

From the gateway experiments a futuristic Data Link Processor (DLP) interoperability building block has been envisaged. The DLP shall aim to combine the Data Model (DM) characteristics of an approved gateway CIEA with the proposed "clever" routing functions to manage information exchange on dedicated Combat Communications Buses and Static Communications Buses.

Each application on the CCB shall be equipped with a DLP to allow the establishment of the CCB between applications. The "clever" routing shall be configurable to manage the infrastructure options to be chosen for information exchanges.

It is envisaged that the DLP shall dynamically be able to create and manage the CCB on available tactical communications networks and similarly the DLP shall be able to dynamically create and manage the SCB on available static communications networks. It is envisaged that most backhaul connectivity between tactical and static networks shall be established via tactical and/or mobile communications networks.

The DLP shall aim to be the connectivity and data model CIEA interoperability building block that provides a standardised information exchange interface to C2, sensor and platform application developers. The use of the DLP concept is illustrated in Figure 10.

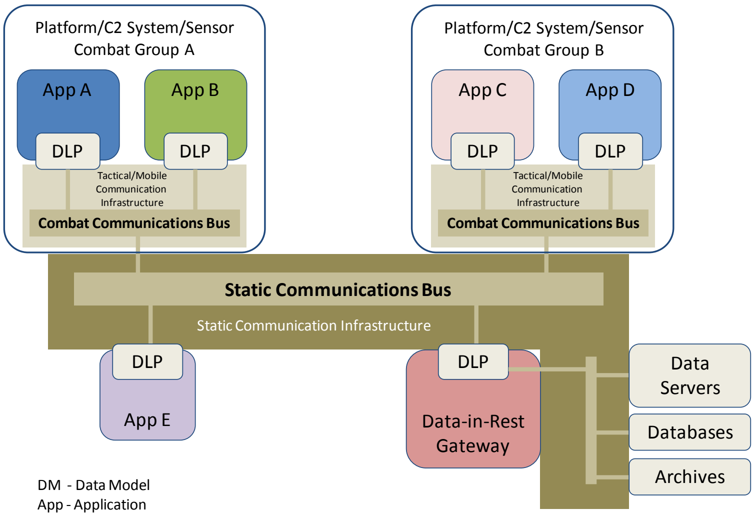

Platform/C2 System/Sensor
Combat Group A
App A
App B
DLP
DLP
Tactical/Mobile Communication Infrastructure
Combat Communications Bus
Platform/C2 System/Sensor
Combat Group B
App C
App D
DLP
DLP
Tactical/Mobile Communication Infrastructure
Combat Communications Bus
Static Communications Bus
Static Communication Infrastructure
DLP
App E
DLP
Data-in-Rest Gateway
Data Servers
Databases
Archives
DM - Data Model
App - Application

Figure 10: Data Link Processor Approach

Conclusion

In conclusion, this paper has identified potential causes that contribute to a lack of interoperable defence capabilities. It has been concluded that there are functional trade-offs that will exist between the customisation of systems versus the interoperability between them.

A Common Information Exchange Architecture was proposed that aims to provide a standardised information exchange mechanism between defence platforms for tactical information exchange (data-in-motion)

An experiment was performed to establish initial viability of the CIEA approach.

It has been shown that the utilisation of a common data model running in a distributed fashion shows favourable results for the establishment of interoperability between legacy and future systems.

It has also been identified that future work for the definition and implementation of combat and static communication busses as well as “clever” routing mechanisms are required.

References

[1] Real Time Innovations white paper – Interoperable Open Architecture (IOA). MOD and DoD Architecting for Interoperability

[2] Levels of Information Systems Interoperability (LISI), US DoD C4ISR Architectures Framework Workgroup, 1998.

[3] Gerrit Niezen. 2012. Doctorate Thesis: Eindhoven University, Netherlands. Ontologies for interaction: Enabling serendipitous interoperability in smart environments. Eindhoven University Press.

[4] Berners-Lee, T., Hendler, J., Lassila, O. 2001. The Semantic Web. Scientific America .

[5] Naude, B. 2012. DPSS-SM-EDERI-C2-014. IDE Experiment 10: Plan for border safeguarding support.

[6] http://www.gc-sat.com/downloads/Impi-Specifications.pdf

[7] Naude, B. 2011. DPSS-DPSS-IDE-1135. NDLOVU: Report on IDE Activities During the Exercise: Period 2011-11-2 to 2011-11-17

[8] Gonçalves, D., De Vries, M. 2012. Towards a force Planning/Design Methodology. INCOSE South Africa conference.

[9] INCOSE-TP-2003-002-03.2.2. 2011. INCOSE Systems Engineering Handbook. Version 3.2.2.

[10] Duvenhage, A., Kourie, D.G., Hancke, G.P. 2011. A layered distributed simulation architecture to support the C2 enterprise.

[11] MIP official site, https://mipsite.lsec.dnd.ca/Pages/WhatisMIP_3.aspx, Last accessed 29 February 2013.

[12] J. Boyd, “A discourse on winning and losing”, AL: Air University Library Document. No. M-U 43947 (Briefing Slides), 1987.

Interoperable Architecture for Command and Control

Cobus Venter (CSIR, South Africa; jpventer@csir.co.za)
Corné Smith (CSIR, South Africa; csmith@csir.co.za)
Rudolph Oosthuizen (CSIR, South Africa; roosthuizen@csir.co.za)
Arno Duvenhage (CSIR, South Africa; aduvenhage@csir.co.za)
Brian Naude (CSIR, South Africa; bnaude@csir.co.za)
Willem H le Roux (CSIR, South Africa; whleroux@csir.co.za)

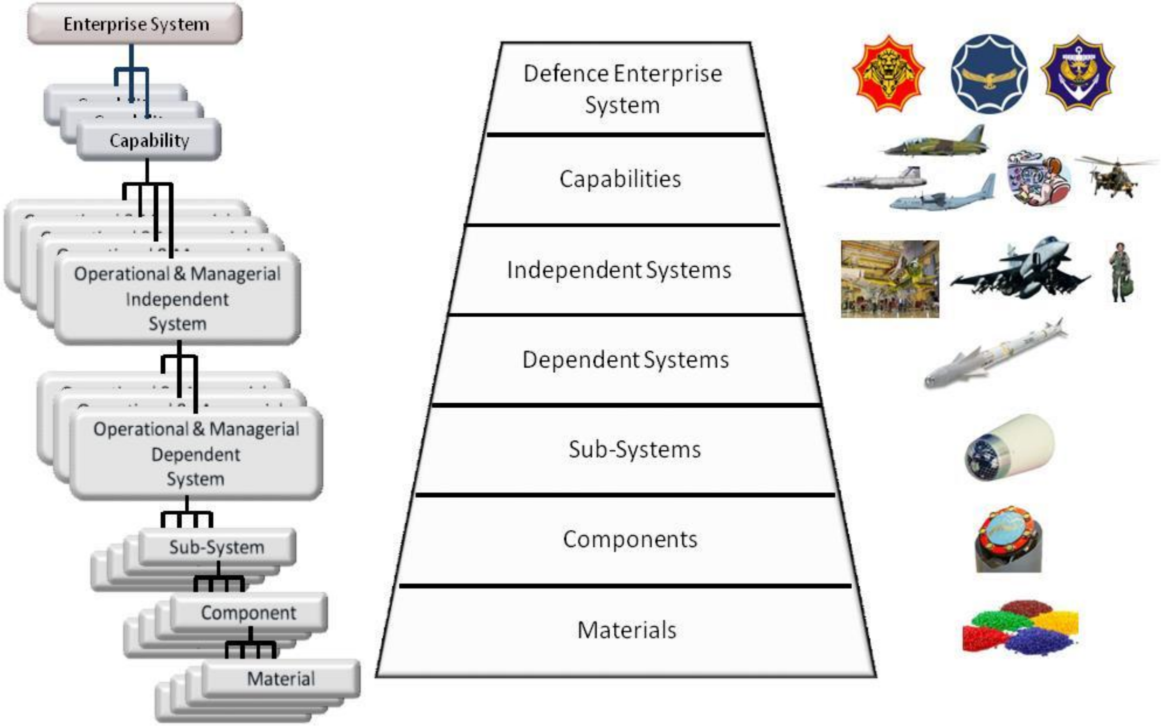
CSIR
our future through science

Contents

1. Introduction
2. Common architecture development approach
3. Common Information Exchange Architecture
4. CIEA concept experimentation
5. CIEA experiment results
6. Data Link Processor concept

Introduction

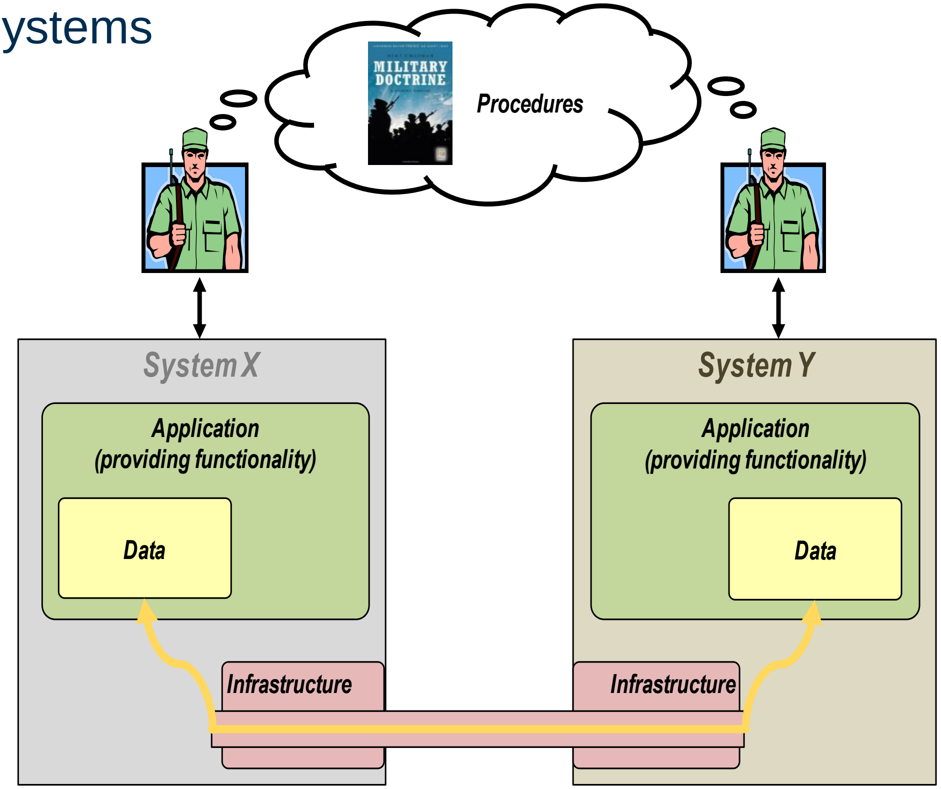
Traditional Engineering Approaches



our future through science

Common Architecture Development Approach

- Defence Forces forced to re-think systems development
- Move towards common building blocks emerging
- Building blocks are expected to conform to a common architecture
- Common Information Exchange Architecture (CIEA) could expedite the integration of systems



our future through science

Common Information Exchange Architecture

CIEA must address:

Interoperability : Flexibility : Shorten Acquisition Time

Common Information Model

Object-orientated (inheritance)
Resource Description Framework
Standard Information Formats
Tags for semantic meaning

Data States

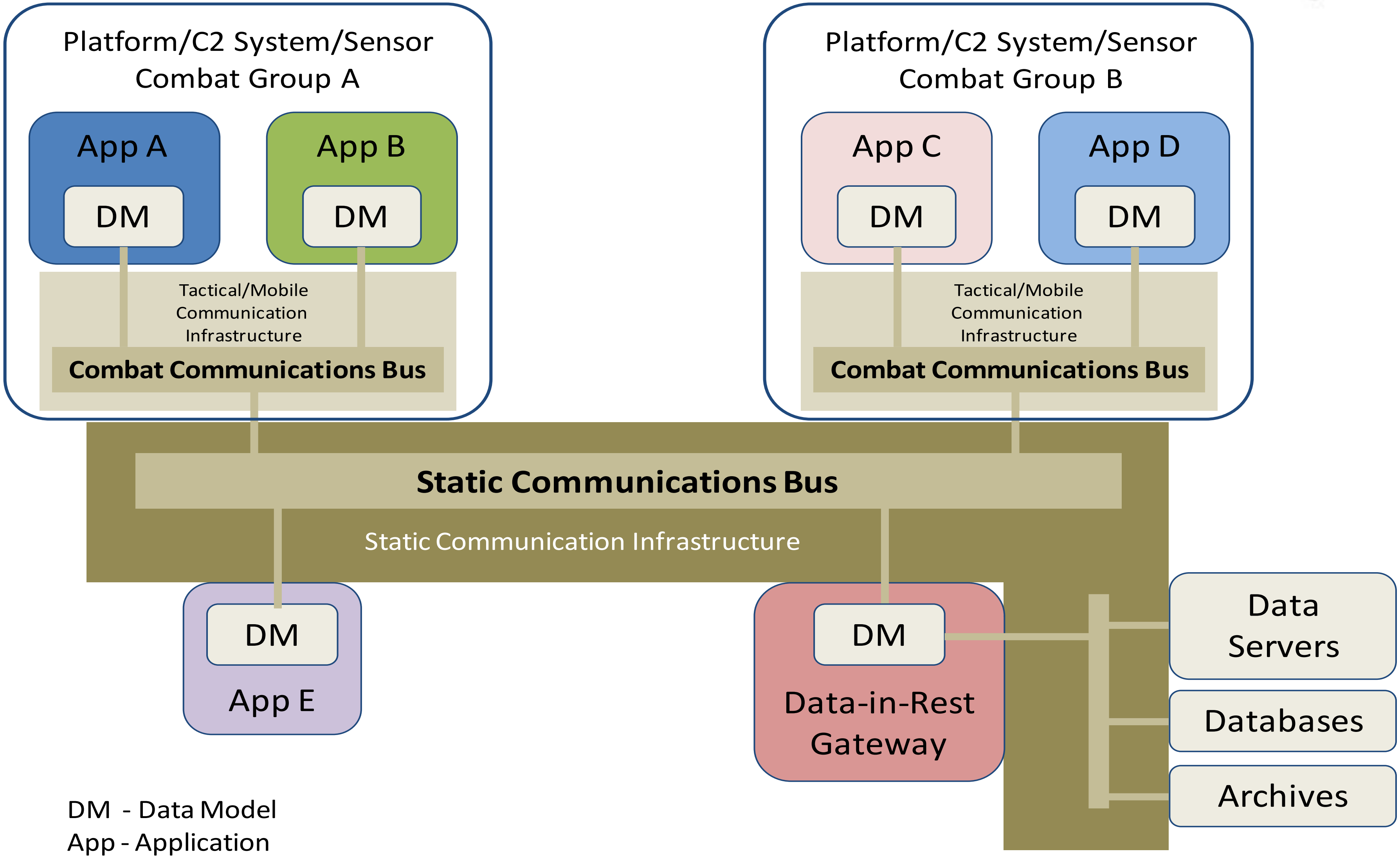
Data-in-Use
Data-in-Rest
Data-in-Motion

Communications

Combat communications bus
 - clever routing
Static communications bus

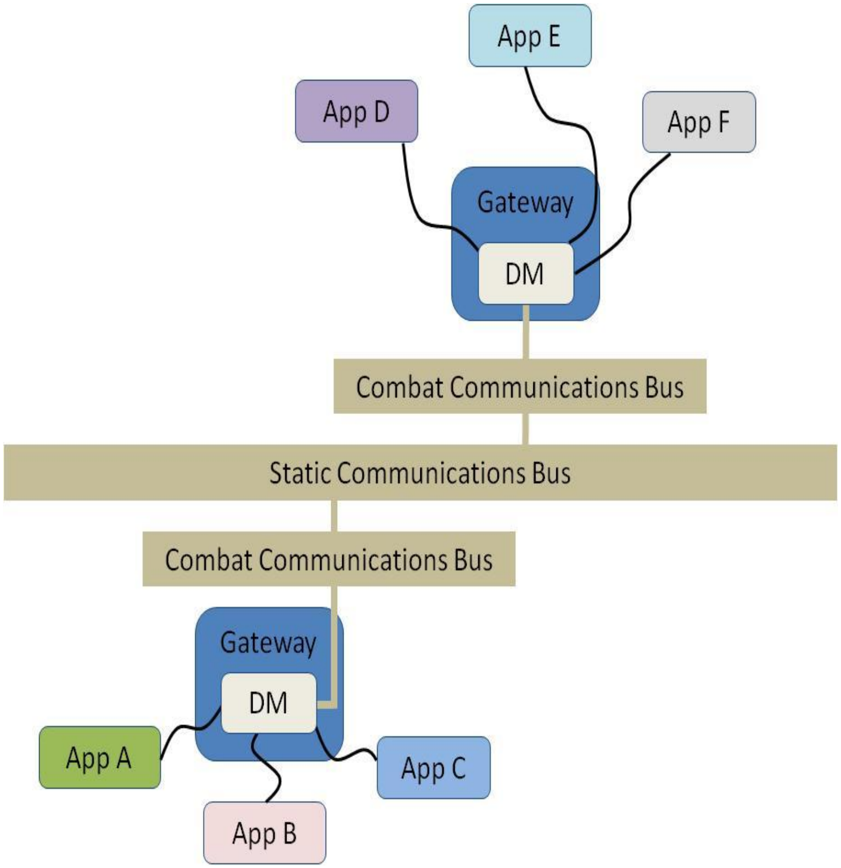
our future through science

Common Information Exchange Architecture


Platform/C2 System/Sensor
Combat Group A
App A
DM
App B
DM
Tactical/Mobile Communication Infrastructure
Combat Communications Bus
Platform/C2 System/Sensor
Combat Group B
App C
DM
App D
DM
Tactical/Mobile Communication Infrastructure
Combat Communications Bus
Static Communications Bus
Static Communication Infrastructure
DM
App E
DM
Data-in-Rest Gateway
Data Servers
Databases
Archives

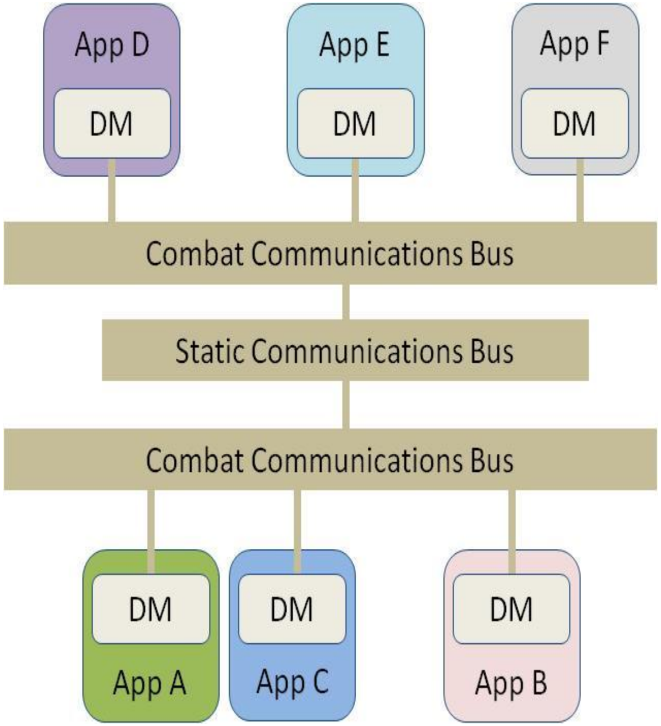
DM - Data Model
App - Application

Common Information Exchange Architecture: Experiment


App E
App D
App F
Gateway
DM
Combat Communications Bus
Static Communications Bus
Combat Communications Bus
Gateway
DM
App A
App C
App B

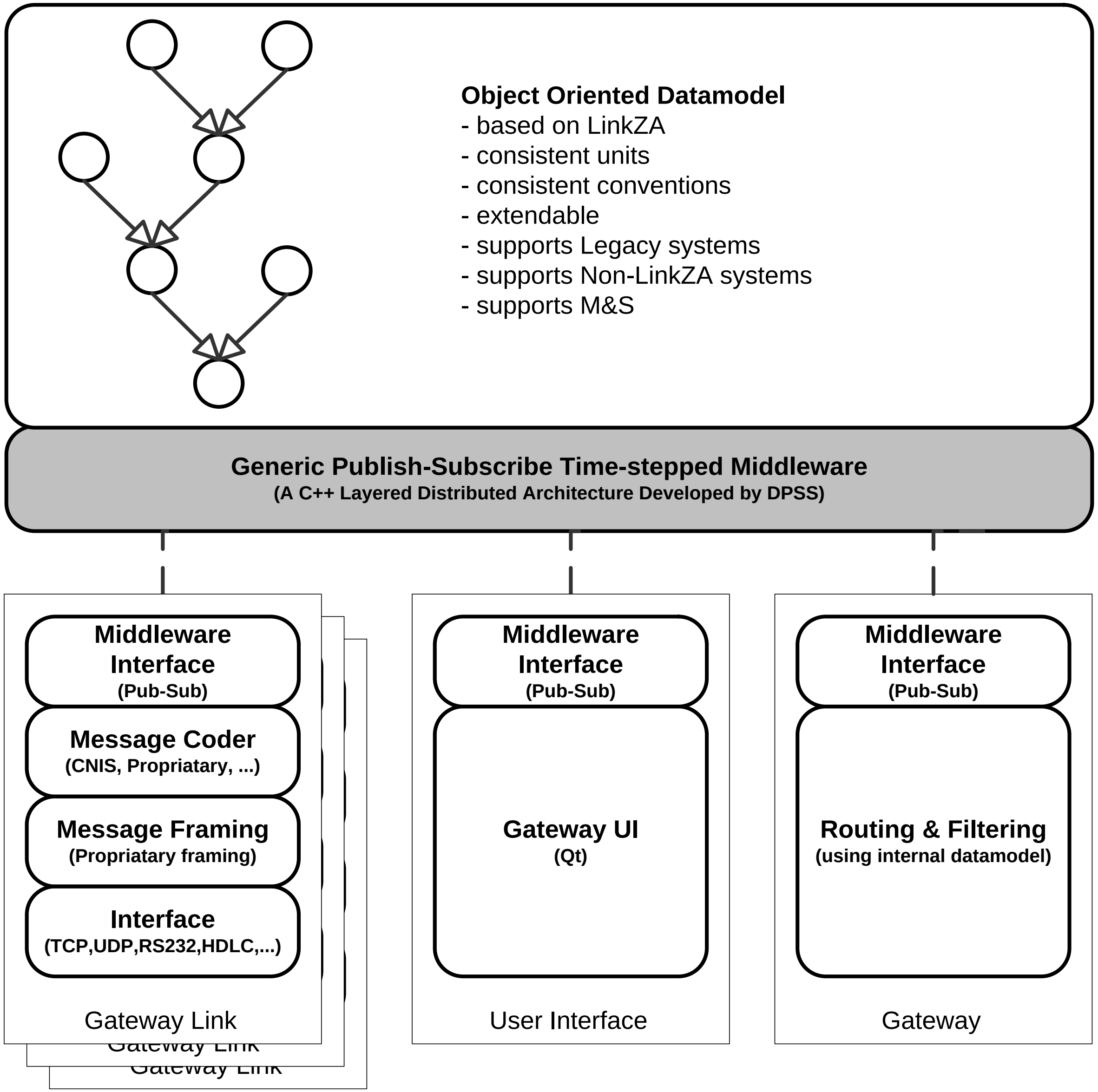
Gateway experimental set-up

Gateway experimental implementation to validate the CIEA


App D
DM
App E
DM
App F
DM
Combat Communications Bus
Static Communications Bus
Combat Communications Bus
DM
App A
DM
App C
DM
App B

Comparative CIEA

Common Information Exchange Architecture: Experiment



Object Oriented Datamodel
- based on LinkZA
- consistent units
- consistent conventions
- extendable
- supports Legacy systems
- supports Non-LinkZA systems
- supports M&S

Generic Publish-Subscribe Time-stepped Middleware
(A C++ Layered Distributed Architecture Developed by DPSS)

Middleware Interface
(Pub-Sub)

Message Coder
(CNIS, Propriatary, ...)

Message Framing
(Propriatary framing)

Interface
(TCP,UDP,RS232,HDLC,...)

Gateway Link

Gateway Link

Gateway Link

Middleware Interface
(Pub-Sub)

Gateway UI
(Qt)

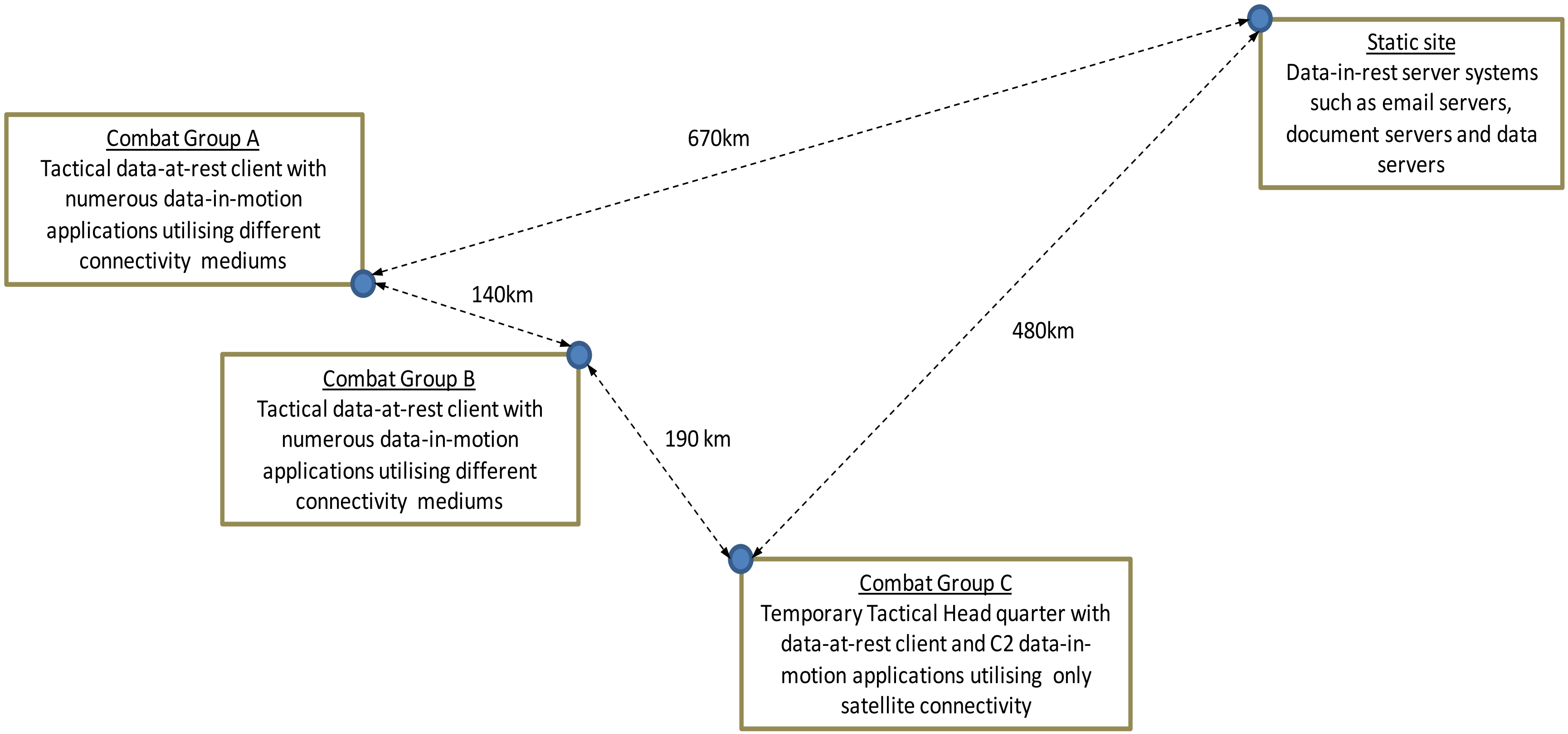
User Interface

Middleware Interface
(Pub-Sub)

Routing & Filtering
(using internal datamodel)

Gateway

Common Information Exchange Architecture: Experiment



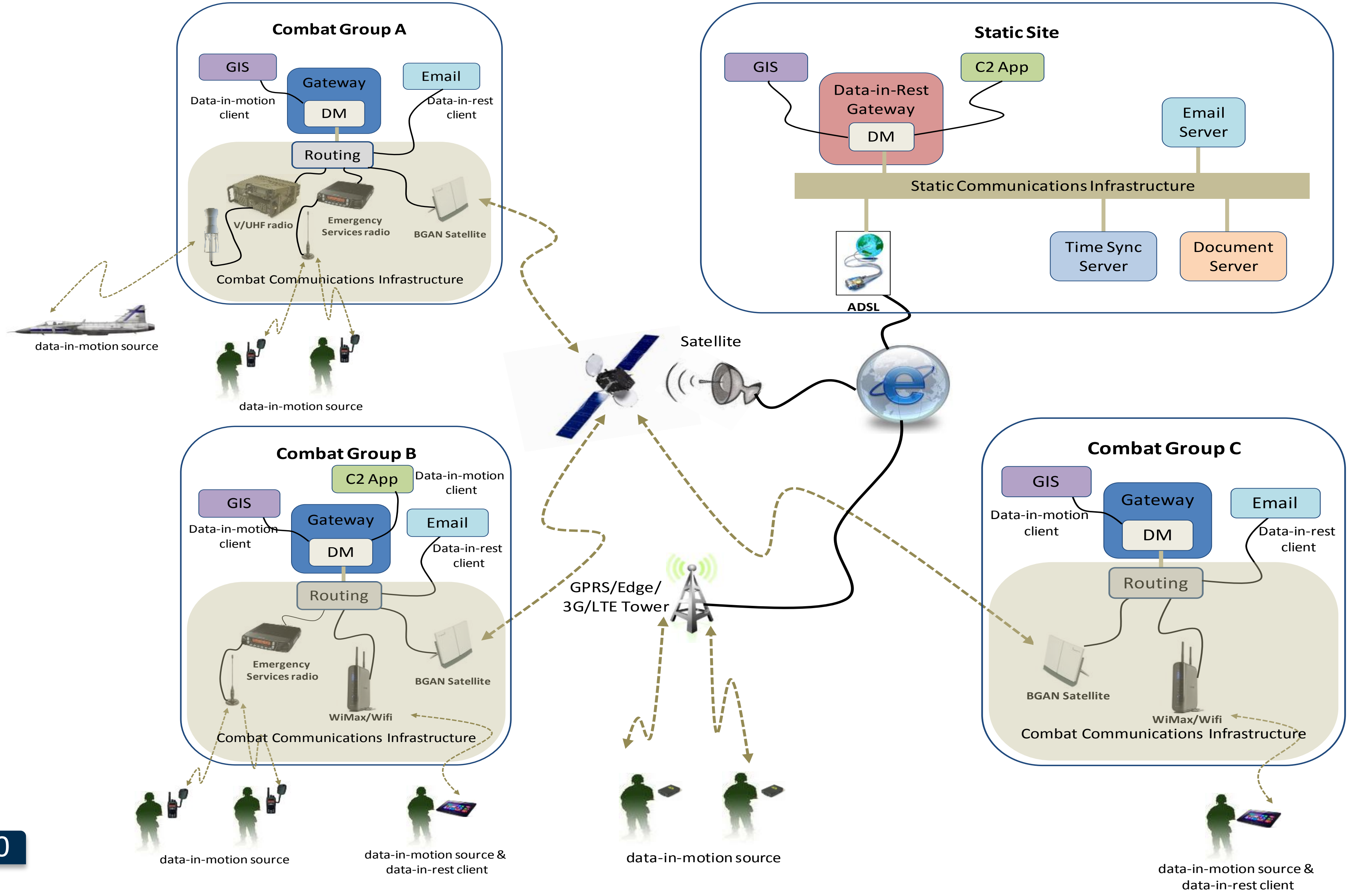
Combat Group A
Tactical data-at-rest client with numerous data-in-motion applications utilising different connectivity mediums

Combat Group B
Tactical data-at-rest client with numerous data-in-motion applications utilising different connectivity mediums

Combat Group C
Temporary Tactical Head quarter with data-at-rest client and C2 data-in-motion applications utilising only satellite connectivity

Static site
Data-in-rest server systems such as email servers, document servers and data servers

Common Information Exchange Architecture: Experiment

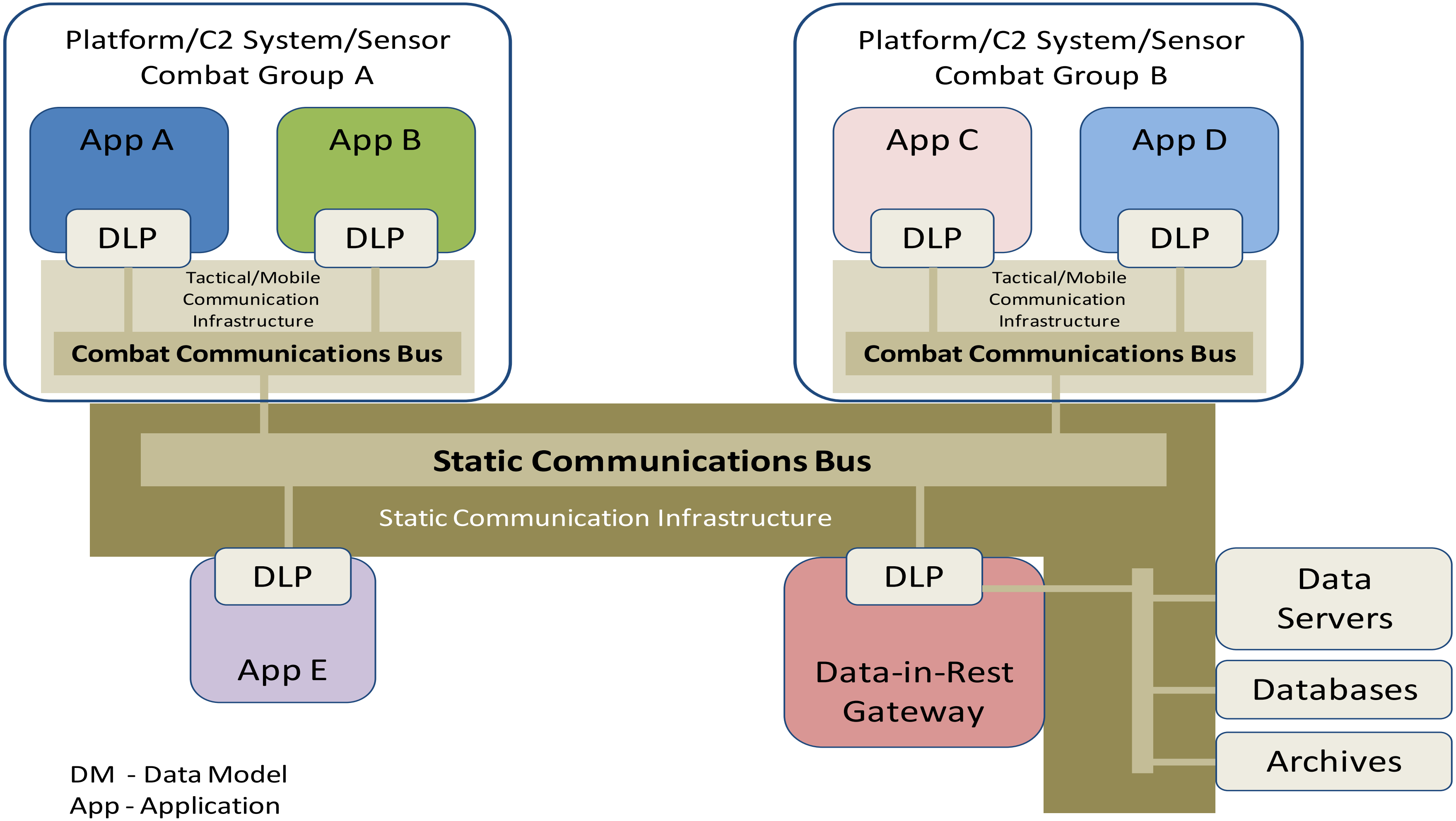


Common Information Exchange Architecture: Experiment Results

Common-Data Model Enables

- Integration of Data-in-Motion sources
- Integrations of legacy systems via gateway
- Flexibility
- Data Filtering Enabled
- Decision rules can be applied

Common Information Exchange Architecture



DM - Data Model
App - Application

Thank you

our future through science