Exploiting Operational Architectures for Application Coherence and Requirement Definition

Colonel (Retd) GJ Barrett
Hi-Q Systems
The Barn
Micheldever Station
Winchester, Hants, SO21 3AR
United Kingdom
Tel 01962 794200

Major K E Galvin
Maj T D Sumpter
Directorate of Information (Army)
(D Info(A))
Room 2-3
Ministry of Defence
Wellington Barracks
London, SW1E 6HQ
United Kingdom
Tel 0207 3409371

Dr S Smith
The Salamander
Organization Limited
York Science Park
York, YO10 5ZF
United Kingdom
Tel 0870 1611700

Abstract

In organisational change, it is common for piecemeal development of information systems to occur or for an incoherent information system inventory to be inherited. Such situations can harbour duplication, inconsistencies and an altogether inefficient use of information resource[1]. Defining coherent and efficient strategies, information system requirements and migration planning through an understanding the context of the overall 'picture' mitigates problems. To provide this understanding, work began on the UK Army Operational Architecture (AOA) in 1995. A number of projects have sought to exploit its development in support of information coherence and coalition warfare objectives. The paper describes how two AOA components - the activity model and its under-pinning information architecture - are used to deliver coherence in application delivery, requirement and interoperability definition. It touches the difficulties of examining the real world against a conceptual framework. It describes how the AMAT[2] has been developed further to support the aims of establishing a coalition baseline of command and control functions and information exchanges between U.S. and UK formations. Developments in a parallel suite of coherence analysis tools are also described. Lessons learnt and issues that should be addressed are highlighted before drawing conclusions.

Introduction

Decision makers constantly have to think creatively about whether what their organisation is doing, or is about to do, is consistent with its objectives, and whether the desired level of performance is optimal, is being achieved effectively and efficiently. Given that every organisation is undergoing change of one form or another (the extreme example being war fighting), a blue print is an absolute requisite to the facilitation of any initiative whether in the battle or business space. Many initiatives in the information arena are driven from the top down but implemented from the bottom up resulting in too little attention being given to processes (the answer to the question 'what?') and too much attention being given to the enablers (the answer to the question 'how?' for example IT support).

Where organizations are involved in change, it is common for piecemeal developments of information systems to occur or for an incoherent information system inventory to be inherited. Such situations can harbour duplication, inconsistencies and an altogether inefficient use of information resources. Defining coherent and efficient strategies,

[1] Information resources are defined as people, process and technology.
[2] AMAT – AOA Mapping and Analysis Tool

information system requirements and migration planning through an understanding the context of the overall 'picture' can mitigate these problems. To provide this understanding, work began on the development of the Army Operational Architecture (AOA) in the United Kingdom (UK) in December 1995. Its first version, released in May 1998, was based on three process-driven views of the Army, which included a High Intensity Conflict view, a Peace Support Operations view and a Business view based on delivering the Land component of military capability. The AOA was issued as a website with access to the models through a MooD®[3] viewer. For the British Army at that time, this was itself an innovative way of distributing the model and supporting analysis. Later development of the MooD Transformation Toolset has provided the ability to web-enable any model and view it locally or remotely using an Internet Explorer plug-in. From January 1999, work began to bring the three views together and to develop the supporting information architecture; this model was initially called the Single Army Activity Model (SAAM) and was briefed to the CCRTS in 1999 at Newport, RI. It has now been renamed to the Army Activity Model (AAM). The supporting information architecture forms the bedrock for the Army Information Architecture (AIA) and defines a prime set of information categories (IC) that are used to describe knowledge required, captured or made available in the processes. These have recently been restructured to reflect more closely an object-oriented approach since an IC resembles an object class.

Coalition Interoperability – AOA Mapping and Analysis Tool (AMAT)

The ability of military forces to operate effectively within the framework of a coalition, whether based on a formal alliance such as NATO, or an ad-hoc alliance, requires that the assigned military forces have a defined degree of interoperability to enable them to exchange critical information. In network-enabled operations, and to satisfy the essential need to avoid blue-on-blue collateral damage, the exchange of liaison staff and voice traffic over radio is no longer adequate. The precursor to any definition of a technical solution must be an analysis and statement of the information that needs to be exchanged between force elements.

The first area for exploitation of the AOA was in support of Coalition Operations planning. This aims to identify the Information Exchange Requirements (IERs) between a U.S. Corps, acting as a CJTF HQ or Land Component Commander, and an assigned UK Division. Work in this area started in March 1999 and continues with limited resources. The programme has delivered the (AMAT) to help with the association of real world Force Elements (organisations) and associated real-world activities (RWA) with a conceptual model. With ICs describing the information interfaces of conceptual process, the RWA introduces the construct of information products (IP) as the input to, or output from, a RWA. This separation is critical to preventing pollution of conceptual models with real world prejudice. IPs are mapped to IC. This provides the AMAT tool with a way to associate conceptual process and RWA through their information dependencies. AMAT delivers considerable benefit by allowing the conceptual model (the AAM) to provide a framework for organisational analysis. It can also show where support is provided by automated systems. This gives the analyst the ability to identify how an IER is provided to and from the UK Division and where mechanisms, whether human or automated, need to be in place to ensure that the IER could be satisfied.

[3] MooD® is a registered trademark of MooD International Ltd.

Facing up to Information Coherence

Gaps and overlaps in the provision of automated systems to support the British Army in both the business and battlespace needed to be identified. An early attempt had been made to conduct gap and overlap analysis in October 1997 by getting IS Project Managers to identify the processes their projects supported. Although this met with a degree of success, it was subjective and no information analysis was carried out. ICs had, at that stage, not been assigned as either inputs or outputs of a conceptual process. Inspecting the AOA alone proved an inadequate guide because this did not examine the information that applications provided nor did it address issues of technical coherence – the ability of applications actually to exchange and understand data. Additional work since October 2000 has focussed on ways of defining and measuring information coherence in and between both legacy and future applications. A study in 2001 concluded that an extension to IC analysis could become the basis for development of a method for measuring application coherence. However, a set of tools that provided a visual representation as well as textual reports would be required. These tools would exploit the MooD® case-tool in a similar way to the AMAT since all information relating to process, force element, application and information architecture were held within MooD® case-tool repository (hosted on MS-Access, Oracle or SQL Server). Figure 1 illustrates the concept.

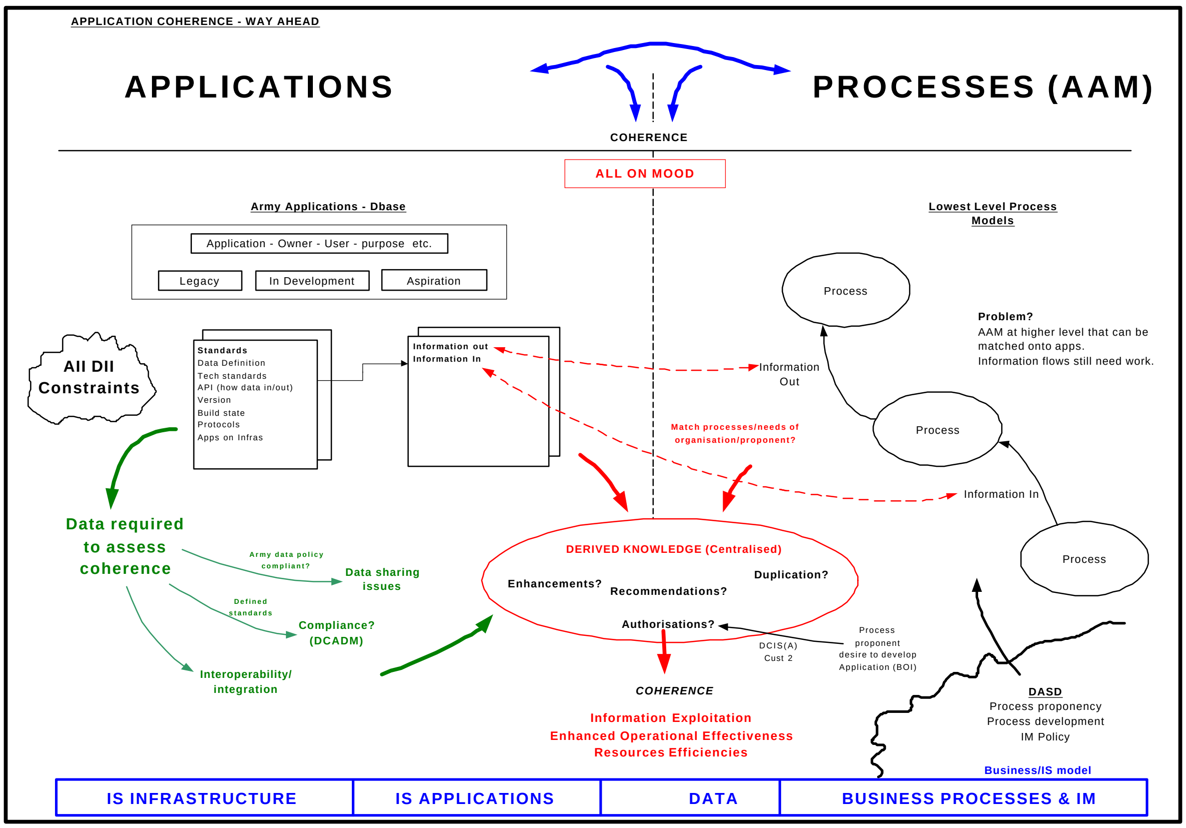

Figure 1: Application Coherence Way Ahead

Coherence Methodology

Director Information (Army) is championing application coherence as a means of delivering resource efficiency, enhanced operational effectiveness and improved information exploitation. His criteria for application coherence are shown in Figure 2.

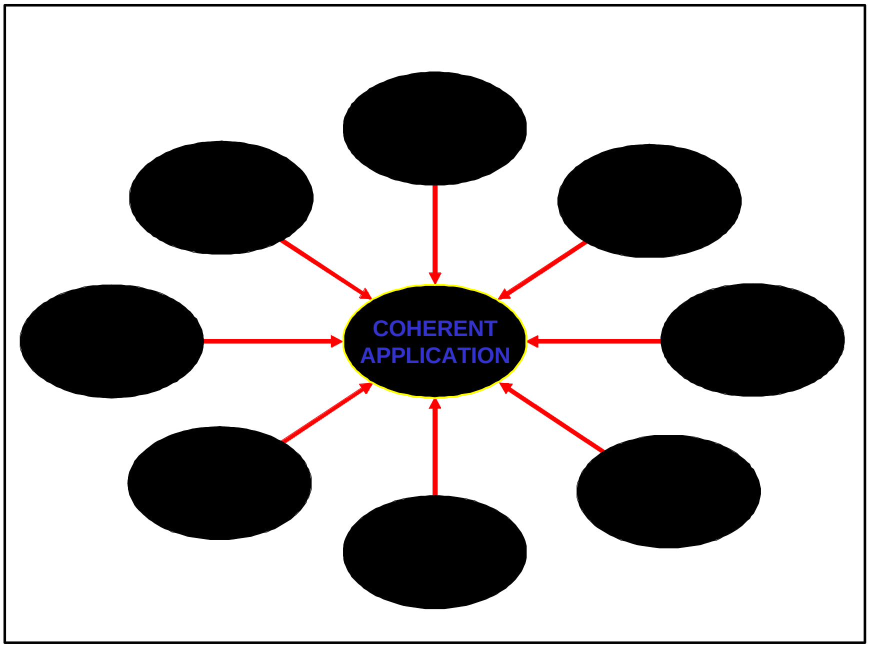

Figure 2 – Coherence Criteria

The study task was the derivation of a **defensible** method to support this aim. The method was to be supported with a simple proof of concept, a full process definition and guidance on an applications database with an underpinning taxonomy. The AAM and its supporting Army AMAT were provided as the essential GFI. The study was in five phases:

- Derivation of the value hypothesis.
- Derivation and documentation of the method.
- Method proof of concept.
- Application database and taxonomy.
- Assessment of the AMAT tool.

Derivation of the Value Hypothesis

The problem of improving application coherence is not one of software integration, but one of developing and maintaining an information view of processes, applications and their inter-relationship. The study was therefore strict in separating its treatment of conceptual processes (CP), real-world activities (RWA) and the types of information elements[4] that support them. The study also uses the term 'ORBAT element or force element' to document an organisation and a 'role' in an organisation. As introduced earlier in this report, the term 'information category' (IC) refers to elements of information that need to be 'known about' in order to conduct a CP. The term 'information product' (IP) refers to the real-world IPs that are consumed by, and produced by, a RWA or a supporting information system (IS).[5] The IS may be automated, that is hosted on a computer system, or maybe be hosted by a human.[6]

[4] Sometimes referred to as 'information objects'.
[5] Applying the context of "know about" (suggesting IC) and "have" (suggesting IP) to a typical "information element" of "identity" provides an illustration of the difference between IC and IP. Many activities require "knowing about identity" – identity is an IC whereas "having identity" is something quite different. A typical IP concerning "identity" might be an ID card, or a passport that one may indeed "have".
[6] Computer-based and human IS are collectively known as 'mechanisms' in AMAT. In most cases human IS are structured as ORBAT elements.

Judgement about coherence is about synergy and is manifest in the identification of gaps and overlaps in sets of target systems. Information-driven analysis will identify these, both in the information needed to support processes, and in the ability of applications to support information requirements. The software integration challenge affects the feasibility and cost of coherence delivery.

Coherence is also a function of the design of the IT applications and an assessment cannot be completed through the comparison of information-based strands alone. Capturing necessary technical details in a suitable data structure will be required. Thus addressing applications coherence is a complex, multi-faceted system of problems.

The starting point for coherence analysis depends on the task and is:

- Either a set of target applications. This has been associated with a relevant and coherent business area definition. It answers questions of the type "How 'well' do (will) applications A,B,C …. fit in supporting area X?"
- Or a coherent business area definition from which a list of potentially supporting applications can be associated. This starting point answers questions of the type "What applications have been (will be) introduced into service that support this area and how well that they fit together?"

A coherent and elaborated set of conceptual processes, relevant to a target problem of the type above, forms an essential reference. The conceptual reference provides the certain knowledge that such a set of processes is coherent.

The value hypothesis, illustrated in Figure 3, is that ICs derived from processes defined in a coherent process model when correlated with:

- ICs contained in IPs derived from applications will inform the coherence assessment of an application set.
- ICs derived from RWA will inform the coherent assessment for RWA.
- ICs contained in IPs consumed or produced by an organisation will inform the coherent assessment of an organisation.

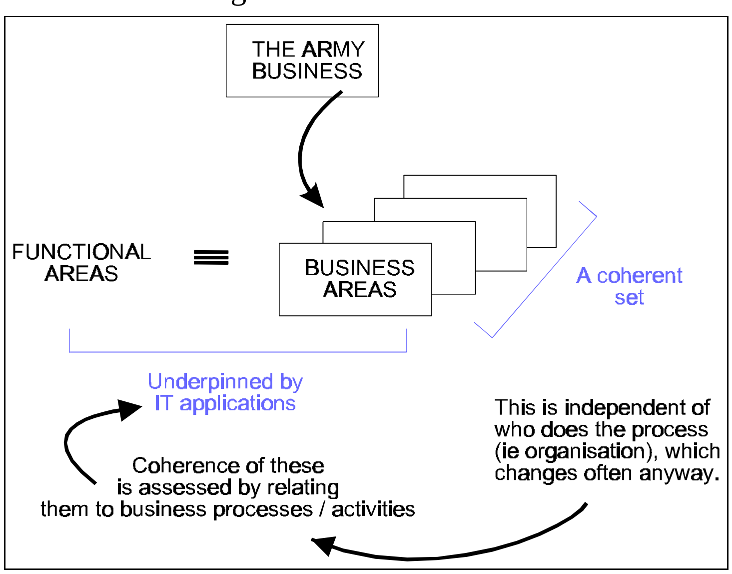

Figure 3: The Value Hypothesis

Derivation of the Method

Soft Systems Methodology (SSM)[7] was used to produce a process model, with a supporting Root Definition (RD), for coherence assessment. It elaborated twelve processes of which nine are at the top level. Of significance to this paper is the 'Intellectual Plan' that comprised the four processes essential to the intellectual integrity of the study. These are: application intelligence assembly, business model development, reporting and coherence management. These subjects are elaborated later. The remaining 7 processes deal with project management and quality control.

Since the concern of the project is coherence, the methodology and its reference components must be coherent and defensible. The production of an SSM-based model makes the assumption that, for its 'coherence' to be guaranteed, it must be distinct from the particular bit of the real world to which is relevant. It must have logical defensibility to the RD (since that is the source of the processes within it). This applies to all levels of resolution irrespective of the levels of detail to which the model is developed. The model is a concept not a description of reality.

The first AOA models from which the AAM was derived were conceptual but in merging the three models and models built to support C2, Targeting and CSS, which had been undertaken as separate work packages, but using the language of the real world, meant that the AAM was not a purely conceptual model. In particular, since all the IT applications were derived for bits of the real world, at the levels of detail appropriate to areas of the real world for which they are intended, there was potentially a difficulty in relating the levels of resolution in the AAM to the levels relevant to a target set of IT applications.

Despite these concerns, the AAM makes an explicit declaration of 'what the Army activities are taken to be'. To help with 'level matching', for each target business area for which application coherence is to be judged, a conceptual model is required either from a single RD or as a Conceptual Primary Task Model (CPTM). Mapping this model onto the AAM allowed a selection to be made of those activities, at whatever resolutions were appropriate, that were relevant to the particular business area. The construction of a simple model, at this stage, also served to reinforce the analyst's understanding of the business area and eased the task of finding suitable processes in the AAM. Any supplementary business area models, and any ICs derived directly from them, were then used to supplement the AAM repository. The creation of a single set of ICs required judgment and resource. They are still not yet in a final state.

The Method Defined

The method assumed that the 'business area' and the 'application set' were defined by a current real-world problem. It was also assumed that an application was a product of its history. An application represents a view of ORBATs, RWA and real-world problems current at time that its design was frozen.

The business area was mapped into the AAM, and ICs from the repository were collected. The applications were analysed for their component IPs, and the IPs analysed for ICs. These were compared with the ICs in the AAM supporting and bounded by the 'business area' and conclusions drawn based on the degree of correlation. This is illustrated in Figure 4.

---

[7] Soft Systems Methodology is a set of methodologies that have been used for over 30 years in problem solving. The key proponents of SSM were Peter Checkland and Brian Wilson see references or visit http://www.makinginformationeffective.com/ssmbook/.

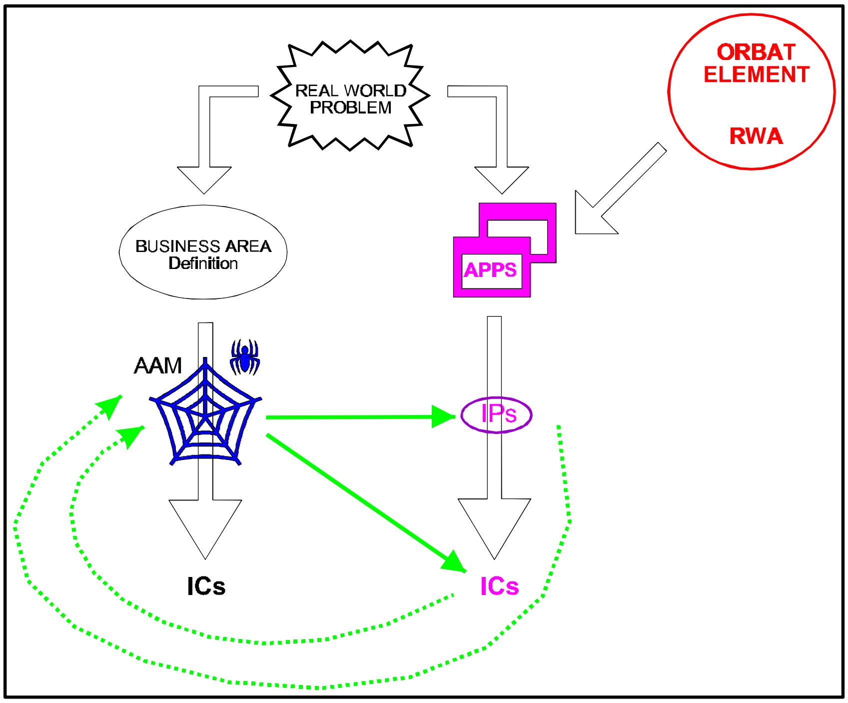

Figure 4: The Spiders Web

It is worth repeating that the assessment process does not require the construction of real-world models. It is the IT applications that are the source of IPs and the derived ICs. The applications are the real-world relation and encapsulate the necessary aspects of origin. However the method is at its strongest where it can be assumed that an application produces a finite set of IPs, for example a set of paper products. Typically, however, an application may produce both a defined set of formal products and an infinite set of products that are structured wholly by the user at the time[8]. To gain a comprehensive view of the application capability, it was both useful and necessary to consider those products that were derived from the ORBAT and RWA to which the application had an association. The AMAT helped this.

The intellectual plan comprised the four main processes essential to the intellectual integrity of the study:

- **Application Intelligence Assembly**. No single authoritative source was found for assembling intelligence about applications. It was therefore essential to establish and maintain relationships with the appropriate parts of other organisations, not only to ensure full collection, but also for subsequent maintenance of the intelligence. ‘Buy-in’ to the principle, supported by a perception of usefulness by the wider community, remains necessary if maintenance is to be achieved through controlled delegation. It was thought very likely that an e-business tool, with a workflow flavour (and probably based on intranet technology) would be useful in lessening the load and distributing management[9]. Collection of intelligence would need to be in sufficient detail for the underpinning information objects to be defined. It would also need to support a critical set of technical interoperability descriptions that may bear on the ability of the coherence assessment outcome to be delivered.
- **Business Model Development**. In addition to the need for intelligence about applications, there was a requirement to have, and maintain agreement, about the

[8] From an ad-hoc query on a database for example.
[9] For example SBM from Savvion

business area/functional area definitions and boundaries. Equally, guidance was required on how to deal with models from other sources that aspire to cover the same ground or exist at different levels of detail. Revisiting, and incorporating such models where necessary, checks the validity and completeness of representations in the AAM and embellishes it. Studying additional models also widens the analyst's perception.

- **Coherence Assessment**. The greater part of coherence assessment was specified in information, rather than technical, terms. It was also thought necessary to capture critical information measures of performance (MoP) that constrain the provision or delivery of an IC through an IS.
- **Reporting**. The reporting process was straightforward. However, it was noted that 'reporting relationships' should be established with those that the coherence assessment is designed to influence, and also with those who contributed to intelligence and model preparation earlier in the whole process. Visualisation of the necessary and potentially voluminous detail was flagged as an area of potential concern and led to the development of the CAT tool set described later.
- **Applications Database**. There is a mutual dependency between knowing about the information parameters of applications ('deriving and storing intelligence about applications') and knowing about constraints associated with application deployment. The study considered the feasibility of deriving a complete set of critical technical detail that, if captured, would allow early judgment on interoperability within the target application set. This proved very difficult to achieve since the interaction between factors are legion. Even so, the study commented on a number of constraints under the general headings of: software technology, communications, data, and architecture and provided a simple taxonomy. Judged most significant were factors derived from operational context. They ranged widely and included, for example, external policy, mandated standards, ownership issues and the perceived importance of an application by the stakeholder community. It was thought that a great deal effort could be wasted in trying to gather a complete set of information when much of it may prove useless, or indeed false, when examined against a real situation. Operational factors such as purpose, relevance and ease of use from which weighting factors may be derived must be known before the true impact of any specific interoperability constraints can be assessed

The Use of AMAT to support Application Coherence

The AAM provides a conceptual model of the activities of the Army in a hierarchy of CPs, each with input and output ICs. AMAT, by contrast, gives an operational view linking the model to RWAs. RWA information, as IPS, consumed and delivered, together with IS, are captured. In its current edition, the AMAT allows analysis to be conducted only in the context of a particular ORBAT element. This was a limitation for application coherence work.

The AMAT delivers structured analysis in a complex series of reports. On the positive side, reports already exist that associate ICs with IS' and ICs with CPs. Gaps and overlaps are also be reported thus providing a basis for convergence testing. On the negative side, the simplicity of its interface belies an arcane interior. Quite some time was expended in making sense of comments offered on its potential usefulness. It was thought better to understand how AMAT works in support of its current purpose before seeing whether its functionality could be readily extended for the coherence task.

The first, and simplest conclusion reached was that, apart from the IER Matrix Report, there must be some doubt over the usefulness of reports that are just lists of associations. Further work was required in the area of visualisation of ‘coherence data’.

Refinement of the thinking throughout 2001/2002 has lead to the development of a prototype set of tools based on MooD. They are web enabled and run in a .NET environment. They are known as I-CAT (Informatics Coherence Analysis Tool) and T-CAT (Technical Coherence Analysis Tool). Other tools in development are F-CAT (Framework Coherence Analysis Tool) and N-CAT (Network Coherence Analysis Tool). I-CAT was the first of the tools to be developed. It provides a visual view of the ICs that are required to support a process and, through analysis of target applications, the ICs that are provided through IPs of the supporting IS. A screen shot of the I-CAT is at Figure 5.

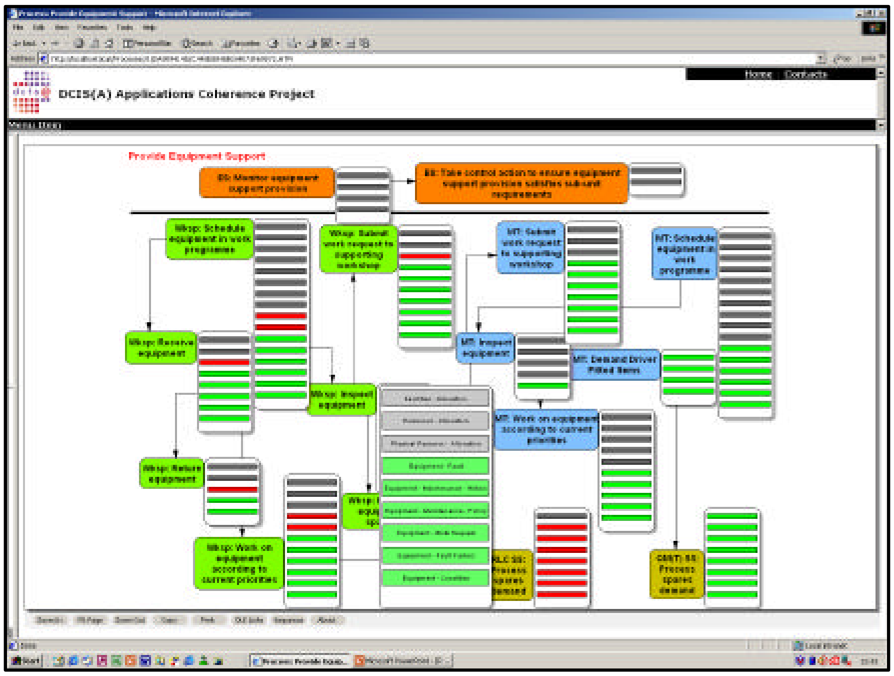

Figure 5: Screenshot of I-CAT Prototype

Each bar represents an IC required by a process. A green bar indicates an IC that is provided in the IP of one IS. A red bar indicates that the IC is available from more than one IS (a potential overlap). A black bar indicates an IC unsatisfied by any IS (a potential gap).

Collection of technical detail to help with judgment of application inter-working is, in many ways, a separate activity. There was also, potentially, a mass of detail. Incorporation into an extended descriptor set in the MooD repository was possible but was not recommended for two reasons: first, every change to the field would involve changes to the MooD data model: second, searches against the fields in MooD, if allowed, would yield ‘standard’ reports that may not usefully support a ‘coherence’ query.

Supporting Coherent Requirements Definition

The purpose this analysis was to look at a typical operational requirement set and make recommendations for presenting the set in a way that would allow analysis using the existing tool set of MooD, AMAT, iCAT and tCAT. The purpose of the analysis was not just to enable comment on a current requirement set but also to set a method for deriving improved, coherent information requirements from the AAM.

The method proposed in shown in Figure 6. The two primary players are a ViewPoint Model of the subject area derived from the AAM, and a user requirement document formatted as an Excel spreadsheet. Whilst a simple automation route to the scripting of requirements from the AAM was sought, the derivation of a preliminary ViewPoint model proved intellectual prerequisite essential.

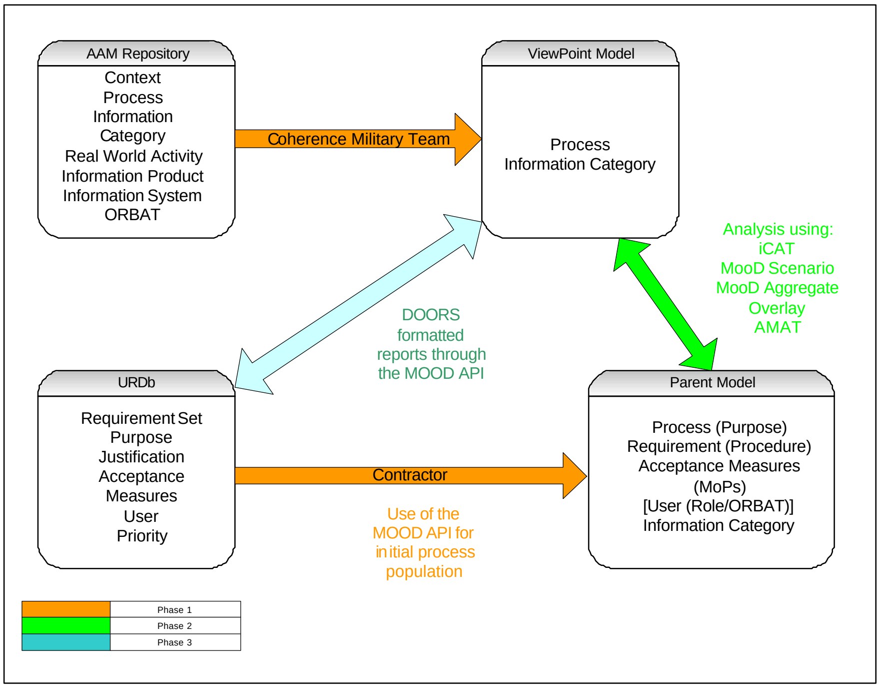

Figure 6: Requirement improvement method

The ViewPoint model encapsulated the key group of conceptual processes and their related ICs. It describes 'what' is to be achieved. A UR document usually answers the question 'how'. A Parent Model is derived by asking each requirement the question 'why?' to which the answer is 'what' and building a process model from the answers. ViewPoint and Parent Model are compared in terms of informatic content – the IC.

Each process (in MooD) is permitted to have 'memo' fields. These were populated to describe mandated elements in a requirement set - for example purpose, requirement, acceptance, user/role and so on. It is a straightforward, indeed almost trivial process, to script MooD API, to deliver a bespoke report for Excel. A direct export from MooD to the URD management system was also possible. Enabling this direct export is shown as a potential Phase 3 activity.[10] Phase 2, in green, is the conduct of analysis of a Parent process model derived in Phase 1 using the Coherence Military Team, MooD Developer, AMAT, iCAT. A validation exercise was conducted using Phase 2 tools. Their use enabled the analyst to reach conclusions about the 'value' that currently stated requirements held and where appropriate provided enhanced requirements. The use of Phase 2 tools also assisted the analyst in absorbing the 'real-world' opinion of current commanders and staff interviewed as part of a traditional requirement definition process. Phase 1 therefore was a short enabling phase. Its output was not completely specified as the effort required to achieve any part of the creation of a Parent process model was unknown.

---

[10] There is an apparent need to be able to 'cut' relevant portions of the AAM to help with ViewPoint model creation. This allows the ViewPoint model to be 'derived from the AAM' yet to be 'part of the AAM'. Language can be changed in the ViewPoint without changing the language in the AAM. Early indications show that this could be possible technically through the MOOD API at some unspecified level of effort. In particular rules concerning 'links' at the ViewPoint boundary required policy direction.

Process Extraction

Although the creation of the ViewPoint model and the Parent process model are essentially different processes, both require the drawing of relevant and necessary material from base repositories followed by the association of relevant information categories to each from a long list. The following section discusses some of the techniques tested.

Text Search

Natural language search engines of considerable sophistication are readily available at reasonable prices (< £1000)[11]. Most search engines offer basic logical searches (‘apples’ AND ‘pears’), and a means of scoring the closeness of a match. Additional features offered may include fuzzy search, phonic search, stemming and synonym. The MooD search engine is not at this level of sophistication.

For bottom-up analysis against a ViewPoint model, it was likely that a few hundred user requirements would be involved, each expressed as one or more sentences of natural language; AAM elements are labelled by a short phrase with amplification in memo fields. However, the search engine is designed to map a few key words to a longer piece of text – in other words, to map in the reverse direction (i.e. from URD to AAM).

These forms of search do not sit well with this analysis need. URs were typically only one sentence long, and used a vocabulary more specific than the high-level, generic phrases of the AAM. This can be offset by searching the whole of the requirement specification, not just the requirement definition. A second limitation is inherent. Since URs are traditionally geared to answering the question ‘how’ whilst the AAM is focussed on elaborating the answer to the question ‘what’. There was considerable doubt whether an automated tool can reliably cross-map the given requirements to AAM elements using words alone. It is more likely that a search tool can assist in identifying elements of the AAM that could form part of the ViewPoint model.

Requirements Categorisation

The field of Requirements Engineering is an active research field, and several academic approaches to automated requirements analysis have been investigated over the last 10-15 years. Some concepts from the General Modelling Approach for Requirement Capture, (GMARC [12] [13] [14]) were found to have relevance. GMARC analyses natural language requirements against an internal (functional) model of the domain to which those requirements apply. It suggests, as an intermediate step, a process to:

- Manually categorise requirements under four headings:
    - View point (this is a reference to the user not a ViewPoint model) – referring to the views that may be taken of the system.
    - Application - referring to the nature and behaviour of the system.
    - Support – referring to possible layers of system functionality.
    - Detail – referring to levels of detail to which the system may be applied.

---

[11] Documentation on ‘dtSearch’ text retrieval engine, available from www.componentsource.com.

[12] ‘GMARC Report R33, Analysis of GMARC Principles and Methodology: Development of Selected Options’, D Furber, S Green, D Bolton, S Jones, D Till, Aug 1992.

[13] ‘GMARC Report R41, Knowledge-Based Support for Requirements Capture’, D Furber, S Green, D Bolton, S Jones, D Till, Aug 1991.

[14] ‘GMARC Report R42, A Framework for Knowledge-Based Support of Requirements Elicitation’, D Furber, S Green, D Bolton, S Jones, D Till, June 1992.

- Requirements are further classified as objectives (‘ … shall …’) or constraints (‘ … shall not …’).
- Non-functional requirements are filtered out.

These classifications were used firstly to structure the requirements, and then help to provide a functional model of them. The procedure may be reasonably automated, but with manual oversight. Because the internal model GMARC uses is different in concept from the AAM, the analysis methods must necessarily also differ. However, the GMARC approach suggested a structure for the AAM technique.

Mapping

The starting point was to take the user requirement set to be an unstructured set of requirements expressed in natural language (even though they purport to have structure) Using the GMARC approach, each requirement was structured against one of three headings:

- Goals (eg ‘Be situationally aware’).
- User (eg commander, private, system administrator).
- Level (eg unit, brigade, division).

URs show many of these characteristics in their native state. As a first task, the names of different goals, users and levels were defined. The next task was to extract the subset of requirements that are capable of comparison with the AAM remembering that the AAM is composed entirely of information flows and processes. Not all requirements are expressed in suitable terms, (eg ‘ … the system shall be easy to use ..’). These are filtered out or, in this case, held in a separate category of ‘system specifications’ that need to ‘taken account of’. It makes no sense to attempt a comparison of these with the AAM. It is also difficult to make use of constraints and for the time being at least, these too were put to one side. The remaining requirements were classed in the categories given above– this effectively gives each requirement a three-dimensional (Goals, User, Level) coordinate. In fact, within this study, the use of the Level and User axes was not been taken further. This was merely a time constraint on the study since Level and User, and their relationship with Goal (process translated to RWA) would be handled within AMAT.

For each goal, a list of ‘information needs’ was prepared. These were in natural English (but using the vocabulary of the AAM). With the optional assistance of a search engine, the information needs were mapped, or translated to, ICs in the AAM. Once ICs are mapped, information needs may be discarded. This mapping was a far easier and more significant task, than trying the direct mapping of requirements (as they are not rich in information content and answer the question ‘how’) to ICs. With facilities already present in the MooD Developer/AMAT, and with the electronic version of the AAM, this mapping provided the last link in a chain. Associated in the model are:

- Requirements to Goals.
- Goals to Information needs.
- Information needs to AAM ICs.
- AAM ICs to AAM processes.

A text search-engine may then be used to match, to supplement and confirm conclusions drawn through the inference chain above. For example, each goal can be related to a set of processes as follows:

- Find the subset S1 of processes using ICs needed by a Goal (a deterministic lookup).
- Find subset S2 of processes that seem related to Goal objectives. (This is a natural language text search comparing all the requirements supporting the current goal with titles of AAM processes.)
- Find the most probable subset S3 = S1 $\cap$ S2. (With user override being permissible).

Critiquing

Software can help to critique requirements. The critique would highlight areas where the requirements do not appear to map well to the AAM. Initially, this may indicate poor associations, which require changes to the mapping of some elements; however, as the mapping was refined, the critique increasingly revealed genuine inconsistencies between the requirements and the AAM. An important concept underlying this analysis was that of clustering – identifying groups of items that are similar according to some criterion. In terms of requirements, the clusters can be identified through an analysis of the (G,U,L) categories to which the requirements are assigned. In the AAM, the clustering could be performed using the 'informational distance' (number of intervening information flows) between two processes. This is the subject of a current study. There is an underlying assumption that clusters in one domain should normally map to clusters in the other domain, and that singletons (either isolated selected items, or unselected items surrounded by selected items) merit more detailed attention. Reasonable tests would include:

- **Correctness.** Ideally, S1 should enclose S2. If not, further examination can be made, for example:
  - Are there processes in S2 not in S1 (i.e. processes which sound relevant but have ICs few in common)? If so, the information needs list, or the needs-category mapping, should be revisited.
  - Are there processes in S1 not in S2 (i.e. processes which use the same ICs as the goal, but appear not to be relevant)? If so, the requirements to make process relevant may be missing or information needs are redundant.
- **Completeness.** Ideally, S3 processes should precisely match the information categories that the goal requires (an unattainable ideal …). Checks can be made to:
  - Confirm that S3 processes use all the ICs that the goal needs.
  - Consider any other information categories used by S3 processes and confirm they are irrelevant.
- **Global Cohesion.** Cohesion is confirmation that requirements that are logically grouped relate to a coherent set of AAM processes. Checks can be made to ensure that:
  - S3 processes cluster well in the AAM (i.e. are reasonably close in AAM)
  - Isolated processes found within S3 are flagged, as well as the ICs they use. These singletons may indicate a misclassification of requirements or a badly directed requirement.
  - Singleton process exclusions are flagged – these may indicate missing requirements.
- **User Class Cohesion.** The basic premise is that requirements dealing with a single user class should be close together from an informational point of view, otherwise the user would perceive a fragmented procedure.
  - For each user class:
    - Find S3 processes relevant to the user class (using text search criteria).
    - Check that these cluster well, highlighting exceptions with high degree of scattering.
    - Further highlight singletons for more detailed consideration.
- **Coupling (interdependence).** Can an S3 set of processes be lifted from the AAM with few information dependencies (good) or are they tightly tied in (bad)?

Visualising

In an information-rich context, there are several attractive means of providing a visual summary of the requirements:

- Once categorised, requirements can be graphically visualised using 3D coordinates representing some or all of their (G,U,L) categories. A view of all requirements may provide insight into the evenness and extent of requirements coverage, and filtered views can also be provided. Colours can be used, for example to label all requirements that map to a given set of processes. Common technology with tCAT is available.
- The techniques developed in CAT tools, and in a parallel proposal to visualise the coherence of processes and applications by counting the number of information categories they have in common, can equally well be used here to compare AAM processes with goals.

Lessons Learnt

It became apparent very early on that addressing applications coherence was a complex, multifaceted set of problems. As fundamentals, a coherent reference information architecture, a defensible method, a knowledge of application informatics, data and technical genus, and a unified team with a 'cool head' would be required. Continuity of understanding within the team is essential. The potential scale of the task meant that, throughout subsequent work, pragmatic judgement was needed to maintain the balance between delivering scientific rigour and uninformed speculation.

A set of 'exam' questions was required to maintain both focus and scope. Fortunately the top Army management task sought 'promotion' of application coherence rather than 'delivery'. This placed emphasis in information coherence rather than technical interoperability. Dealing with business semantics is an essential prerequisite.

It was self-evident that the creation of a single set of ICs required judgement and resource.

The role of the real world, manifest as RWA and applications themselves, has been explored at some length. Although the study concludes that the assessment method does not require the construction of real-world models, the essential 'trigger' for the whole method is a real-world problem and hence real-world knowledge provides essential context. Care was be? required, however, to ensure that conceptual models were conceptual and not just descriptions of what the real-world ought to be. In particular, the study suggested that the derivation of IPs and then ICs directly from an application by inspection might lack necessary richness and would not adequately capture the 'products' of an ad-hoc enquiry.

Future Work

Lines of Operation

Although the AOA and the Application Coherence Methodology is able address all areas of application development through three lines of operation as shown in Figure 7, in the short term it is being utilised to support:

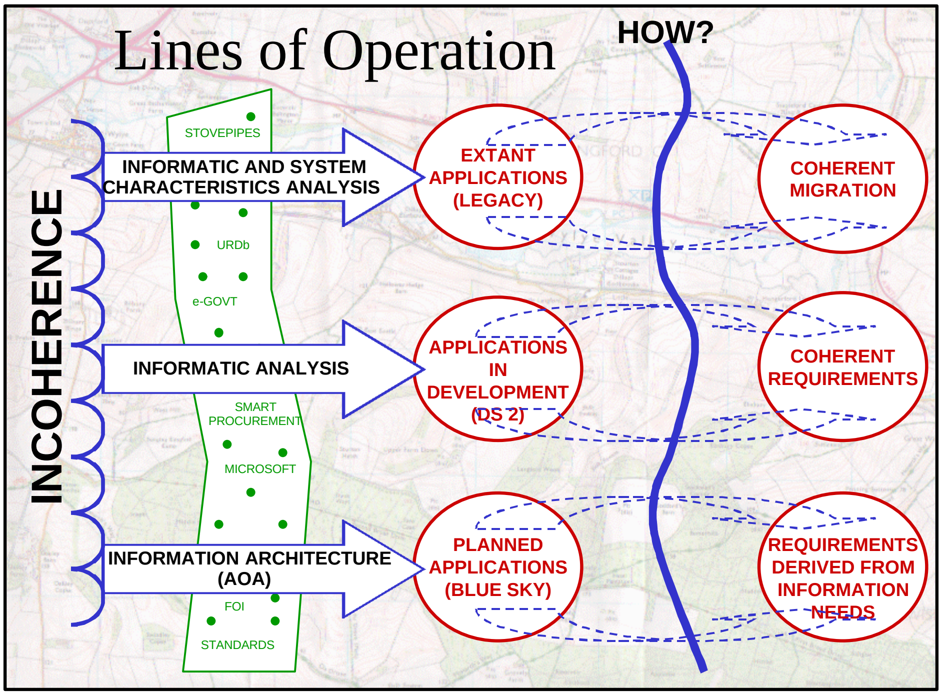

Figure 7: Lines of Operation

Line 1: Legacy applications (Extant Applications)

- Rationalisation of gaps and overlaps
- Pruning of competing applications
- Mapping to future requirements

Line 2: Current requirements (Applications in Development)

- Derived from User Requirements but modelled as an information model
- Grouped to support key processes
- Common requirements, interfaces and IERs identified

Line 3: Future requirements (Planned Applications)

- Based on information and process model of the business
- Fully aligned to our process needs

- A gap and overlap analysis of approximately 60 applications in use for the Unit within the G1-G9 functions for Director Information (Army).
- The potential of applications to support processes and mappings between roles/posts and organisations.
- An analysis of UNISOFT, which is a set of software applications hosted on an infrastructure that needs to be migrated to new target infrastructure. The applications are used by Units and there is a need for Army Digitization Steering Group (ADSG) to understand what the impacts of migrating to a new application are so they can make informed choices.
- The way the Unit will train its personnel in light of the BOWMAN radio coming into service.
- Work is also examining how a Unit on transition to a Battlegroup can more effectively organise to take advantage of BOWMAN and the BOWMAN SITUATIONAL AWARENESS MODULE (BSAM). A series of Limited Objective Experiments (LOEs) will take place in September 2002. To support this work, a Unit C2 model as been derived and mapped to the AAM. A Level 1 view of the model is shown at Figure 8.

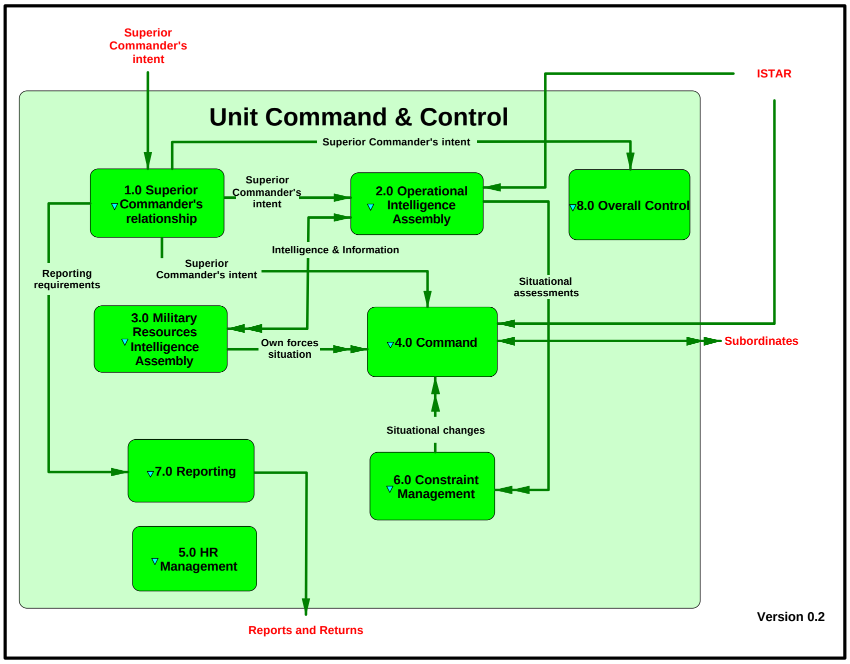

Figure 8: Unit C2 Level 1 Model

C4ISR Framework

Other work in progress is looking at the relationship between the concepts in use in the MooD, AMAT and the CAT tools and other real world classifications. In particular, a start has been made into looking at the cross-categorisation of process and IC, RWA and IP and information resource (application or system) to the elements of the C4ISR Framework.

Conclusion

The study derived a value hypothesis to test application coherence. Practical experimentation has tested it and concluded that it was defensible. Techniques such as these are essential planks in the transformation of the Army approach to information services in the network-enabled age. The Army Operational Architecture, and models dependant on it, are key components to understanding the transformation that is required. Coherent applications, developed through coherent requirements, are the media through which the transformation will be delivered. Success is not optional.

References

A. [Wilson, 1992] Dr. Brian Wilson *Systems: Concepts, Methodologies and Applications*. Second Edition, John Wiley & Sons (ISBN 0-471-92716-3).

B. A Method for Promoting Application Coherence – A Paper for DCIS(A) by HiQ Systems (Document No.C207-EP-01 Issue 1.0b April 2002.

C. [Wilson, 2001] Dr. Brian Wilson *Soft Systems Methodology: Conceptual Model Building and its Contribution*. First Edition, John Wiley & Sons (ISBN 0-471-89489-3).

D. GMARC Report R33, Analysis of GMARC Principles and Methodology: Development of Selected Options', D Furber, S Green, D Bolton, S Jones, D Till, Aug 1992.

E. 'GMARC Report R41, Knowledge-Based Support for Requirements Capture', D Furber, S Green, D Bolton, S Jones, D Till, Aug 1991

F. GMARC Report R42, A Framework for Knowledge-Based Support of Requirements Elicitation', D Furber, S Green, D Bolton, S Jones, D Till, June 1992.

Acknowledgements

- Dr Brian Wilson for his initial support in developing the Root Definitions and Level 1 CPTM for developing the Application Coherence Model.
- Dr Dick Whittington and staff at the Salamander Organisation Limited for their advice and support in the use of AMAT.
- Major Rob Gascoigne for his initial project support in enabling Hi-Q's Application Coherence Methodology before handing the project to Major Toby Sumpter who with the Salamander Organization and Hi-Q Systems are developing the visualisation tool set.
- Lieutenant Colonel Peter Nell for his promotion and encouragement to utilise the AOA in new ways to support Information Coherence.