A Taxonomy of Network Centric Warfare Architectures

Anthony Dekker
Defence Science & Technology Organisation
DSTO Fern Hill
Department of Defence, Canberra ACT 2600, Australia
Email: tony.dekker@dsto.defence.gov.au

ABSTRACT

The Australian Defence Force (ADF) is committed to transitioning, over time, to Network Centric Warfare (NCW). NCW sees the elements of the ADF as "nodes" in a network. The capability of such a force is determined not so much by the individual capabilities of each node, but by the systems properties of the network as a whole. In this paper, we provide a taxonomy of possible NCW architectures, in order to illuminate the possible options and to foster debate and experimentation regarding the appropriate NCW architectures for use by the ADF. The taxonomy is based on the fundamental concepts of value symmetry and homogeneity/heterogeneity, and distinguishes Centralised, Request-Based, and Swarming architectures, as well as combinations of these. We provide several examples of each architecture, and a list of key questions.

Keywords: Architecture, NCW, Network, Swarming, Topology

1. INTRODUCTION

The ADF is committed to transitioning, over time, to Network Centric Warfare (NCW). The ADF's Force 2020 document states (Australian Department of Defence 2002*a*):

> *"In the force of 2020, we will have transitioned from 'platform-centric' operations to 'Network-Enabled Operations'. As the name suggests, Network-Enabled Operations derive their power from effectively linking different elements of the organisation to conduct warfare more effectively. Network-Enabled Operations treat platforms as 'nodes' of a network. Since all elements of the network are securely connected, they can collect, share, and access information."*

This perspective sees the elements of the ADF as "nodes" in a network. The capability of such a force is determined not so much by the individual capabilities of each node, but by the capabilities of a group of connected nodes which can synchronise in order to tackle a particular problem.

Depending on the NCW architecture chosen, the nodes of the network may be platforms (such as ships or aircraft), weapons or C4ISR systems within platforms, groups of one or more people, or combinations of these.

The ADF's NCW Roadmap document sets out a direction of increasing investment in the network, but does not yet make a choice between different possible NCW architectures. As the draft Roadmap puts it (Australian Department of Defence 2003):

> "*the full implications of enhancing collaboration and shared situational awareness have yet to be identified*."

In this paper, we provide a taxonomy of possible NCW architectures, in order to illuminate the possible options and to foster debate and experimentation regarding the appropriate NCW architectures for use by the ADF. It is important to understand these options and how they might be used tactically, since there is no “one size fits all” NCW solution. The taxonomy should therefore be of use both to capability planners and to staff developing operational doctrine for NCW.

It should be noted that increasing investment in networking does not mean that every node should be directly linked to every other node. This would, in general, be both prohibitively expensive and unnecessary. What we need is two conditions to be satisfied:

1. The right information should be delivered to the right node at the right time. This will often require direct links for time-critical communication, such as real-time targeting or avoidance. On the other hand, indirect connections may suffice for situational awareness information not relevant to immediate targeting.

2. The network must have sufficient redundancy in communication paths to provide robustness in the face of nodes being destroyed.

We can measure progress towards condition (1) using NCW metrics based on message transfer time (Dekker 2002, 2005), and we can measure progress towards condition (2) by assessing the robustness of individual nodes and links, and by counting the number of independent paths available between any two nodes (Dekker & Colbert 2004).

However, we cannot decide what constitutes sufficient networking until we can answer the question: *how is C2 carried out using the network?* The network is of little value until this question is answered. The taxonomy in this paper represents a step towards answering it.

Our NCW taxonomy is based on two important concepts for describing what nodes are like. The first is **value symmetry**. We say that an NCW architecture is **value-symmetric** if all nodes have the same value, in the sense that the loss of any node is as serious as the loss of any other. An NCW architecture is **non-value-symmetric** if some nodes are more critical than others. For example, losing an AWACS aircraft (Clancy 1995) is far more serious than losing one of the individual fighter aircraft it controls. Therefore an NCW architecture built around an AWACS aircraft and its fighters will be non-value-symmetric. There is a spectrum of value symmetry between these two extremes, ranging from totally value-symmetric, through to totally non-value-symmetric. The vertical arrow in Figure 1 shows this spectrum.

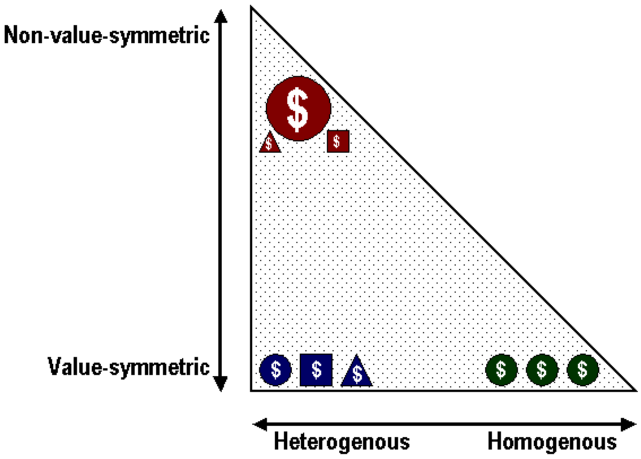


Figure 1: Two Basic Concepts for NCW Taxonomy

The second concept is **homogeneity/heterogeneity**. We say that an NCW architecture is **homogenous** if all the nodes are identical, and **heterogenous** if all the nodes are different. Again there is a spectrum ranging from nodes being completely identical, through to totally different. The horizontal arrow in Figure 1 illustrates this. Combining this spectrum with the previous one produces a triangle of possibilities, since a homogenous architecture must obviously also be value-symmetric.

Each of the options in this triangle of possibilities leads to a different option for NCW, as shown in Figure 2. We have labelled key points in Figure 2 with the letters A to G, representing seven major options for NCW. We will first describe the most extreme cases at the three corners of this triangle:

- Type A: the Centralised Architecture
- Type E: the Request-Based Architecture
- Type G: Swarming Architectures

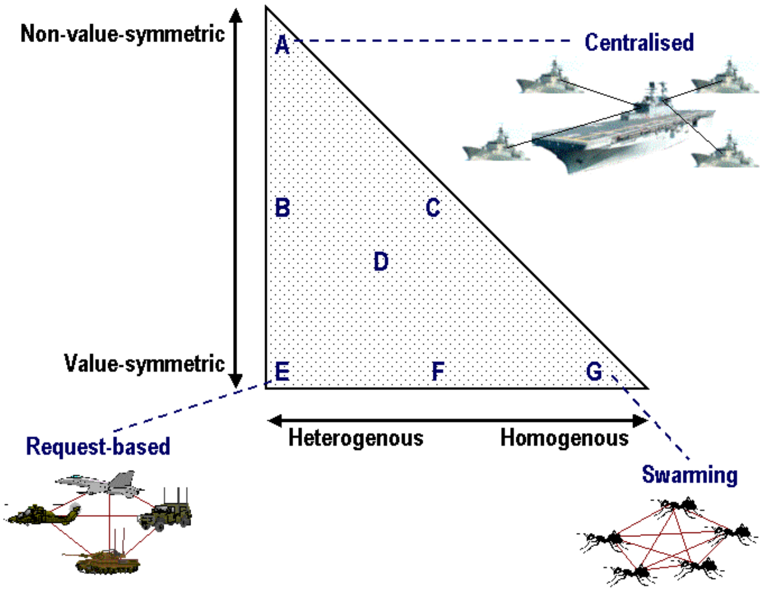


Figure 2: Seven NCW Architectures

The three "corner" architectures A, E, and G may not be totally realistic for the ADF network as a whole, although they may occur in sub-networks, such as in the air environment. More realistic architectures (such as Types B, C, D, and F) will be combinations of these extreme cases, and will be discussed later. However, in order to understand these "combined" architectures, it is important to understand the extreme cases first. Understanding the strengths and weaknesses of the different architectures leads to specific questions that need to be addressed. Table 1 lists some of these questions.

2. TYPE A: CENTRALISED NCW

The most non-value-symmetric architecture (Type A) has a single high-value central "hub" node, surrounded by a cluster of nodes of lower value. If there are multiple high-value nodes, a Type B (Hub-Request) or Type D (Joint) architecture results. The central "hub" in a Type A architecture provides services of such high value that the force cannot operate effectively without it. The "hub" is therefore what Clausewitz called the "*centre of gravity... on which everything depends*" (Clausewitz 1873).

For example, for Australian troops in Vietnam in August 1966, the base at Nui Dat provided recovery and replenishment to Australian infantry patrols, as well as artillery support (McAulay 1986). A US aircraft carrier (Clancy 1999) provides a runway and repair/refuelling/support facility to about 50 strike aircraft, without which they could not operate. A US Air Force AWACS aircraft (Clancy 1995) provides critical surveillance information to fighter aircraft, without which they would be much less effective.

Table 1: Some Questions of Interest for the Seven NCW Architectures

Architecture	Questions
A: Centralised	Which high-value "hub" nodes (e.g. AWD, AEW&C) will be purchased? How many will be needed? What tactics will be used to protect these assets? Will fully centralised control be used?
B: Hub-Request	Can the "hub" be fitted into the request framework?
C: Hub-Swarm	Will high-value "hub" nodes be used without centralised control? Can the nodes combine Swarming behaviour with effective use of the "hub"?
D: Joint	How can the different types A, B, C, E, F, and G be combined effectively?
E: Request-Based	Does the networking exist to pass requests across Corps, Service, and Coalition lines? Are there cultural and organisational barriers? Are there technical interoperability barriers? How are requests prioritised and balanced? What is the services matrix? What are the service-level agreements?
F: Mixed	Will sensor, C2, and engagement sub-nodes on platforms be networked separately? Will we adopt CEC?
G: Swarming	Will we have the network bandwidth to utilise fully distributed networking? Will we use autonomous platforms? How will issues of Unity of Command and ROE be handled? How will sensor data be fused? Which synchronisation mechanisms will be used?

The central "hub" has an up-time period during which it is available (limited by fuel, crew fatigue, etc.) and a downtime period for crew rest, refuelling, repair etc. The "hub" is also vulnerable to attack. To allow for downtime and possible loss, at least 3 (and often more) "hub" units will need to be acquired per area of operation, in order to give a capability for sustained operations. For example, the US Navy has 12 active aircraft carriers, in order to support two-ocean operation.

The vulnerability of the "hub" also means that a significant fraction of the total force capability needs to be devoted to protecting it. For example, a US aircraft carrier is supported by about 8 escort vessels (cruisers, destroyers, frigates, and submarines), which have a heavily defensive role (though with some offensive missile capability).

Why bother with such an expensive and vulnerable asset? Because the central "hub" acts as a **force multiplier**, increasing the effectiveness of the other nodes significantly. In successful Type A architectures, this force multiplier effect more than makes up for the cost. US experience with AWACS aircraft (Clancy 1995) is that they justify themselves because of the way that they increase the effectiveness of the fighter aircraft they control. The British discovered in the Falklands (Ward 1992) that refuelling can extend the reach of short-range strike aircraft, but that the use of an aircraft carrier provides much greater capability. Similarly, the Australians at Long Tan (McAulay 1986), outnumbered 25 to 1, would have lost the battle if not for the 3000 or so rounds of artillery fired from the base at Nui Dat.

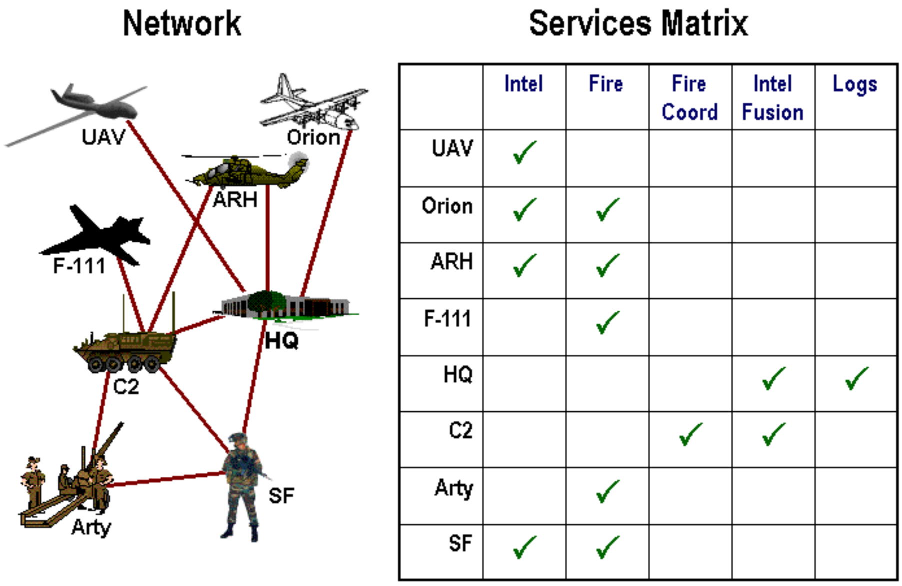
In Centralised Type A architectures like these, the central "hub" is very well defended, because the force as a whole cannot operate effectively without it (see the description of Type B and C architectures below for cases where the force **can** continue to operate without the "hub"). This means that we might as well make the "hub" the centre of the communications network, and perhaps also include a C2 element (for example, a US aircraft carrier houses about 70 flag staff). In many cases, it then makes sense for a moderate to high degree of centralised C2 to be applied, particularly if the network is of high quality, or if the “hub” is an ISR asset (such as an AWACS aircraft).

Fully centralised C2 makes sense when the operational and tactical problems are suitable, when the “hub” has access to all the required information and has the necessary facilities for decision-making, and when the network allows centralised instructions to be disseminated sufficiently quickly (Dekker 2003*a*). Such fully centralised C2 appears to be more suited to the air and maritime environments than the land environment. Australia’s commitment to Mission Command (Lind 1985, Australian Department of Defence 2002*b*), however, favours more decentralised C2.

3. TYPE E: REQUEST-BASED NCW

The combination of fully value-symmetric and heterogenous forces is a collection of pure specialists, all different, but all of equal value. Each node does only a few things, and does them extremely well. Since military operations require multiple coordinated tasks, each node must call on many others to perform tasks that it cannot do. The result is a request-based architecture similar to the design of service-oriented computing technologies like **Jini** (Oaks & Wong 2002). In this kind of architecture, requests for services are broadcast across the network, and the network identifies possible nodes which can satisfy the request (Hall *et al* 2004). These nodes in turn may require additional services, thus generating further requests.

For example, a C2 node in charge of a small group of land forces may need information on the terrain ahead. A request on the network may find a UAV node which provides video on the area. This in turn may reveal enemy concentrations that need to be destroyed. Policy may require that requests for fire support go to a fire-support-coordination node (which can balance priorities), and the fire-support-coordination node will then select a combat node to provide the required fire support. The fire support node will in turn request engagement-quality sensor data, which may be provided by a node other than the original UAV. As the mission progresses, a complex web of requests is formed, which requires an efficient and well-connected network. Planning a Request-Based architecture requires constructing a **services matrix**, indicating the services that can be requested or provided by nodes, as shown in Figure 3.



	Intel	Fire	Fire Coord	Intel Fusion	Logs
UAV	✓				
Orion	✓	✓			
ARH	✓	✓			
F-111		✓			
HQ				✓	✓
C2			✓	✓	
Arty		✓			
SF	✓	✓			

Figure 3: An Example Request-Based Architecture

Request-Based architectures also raise a number of other issues, such as:

- Whether requests can be successfully passed across organisational boundaries (Dekker 2003*b*), especially joint or coalition boundaries. The Organisational Interoperability Model (Clark & Moon 2001) provides a useful tool for addressing the organisational issues in this kind of architecture.

- How prioritisation and balancing of requests is handled (Hall *et al* 2004).

- Service-level agreements for the entries in the services matrix (Hall *et al* 2004), covering issues such as completion rate and timeliness.

- Whether nodes **trust** each other (Zolin *et al* 2004) sufficiently to rely on the result of their requests.

4. TYPE G: SWARMING NCW

The combination of fully value-symmetric and homogenous forces is a “swarm” of identical nodes, and is therefore appropriate only for Single-Service architectures, such as flights of aircraft or naval flotillas. None of these nodes is a specialist in any particular task. Each node has a sensor (perhaps with limited range). Each node has a weapon (perhaps of limited power). Each node also has a limited C2 capability. To operate effectively, these nodes must **share** their sensor information, and **self-synchronise** in order to mass the effect of their weapons. There are two main ways of doing this, which we call **Emergent Swarming** and **Situationally Aware Swarming**. Situationally Aware Swarming in turn has three subtypes. Figure 4 illustrates the possibilities.

Sometimes the swarm of identical nodes is supplemented by a centralised ISR asset: this is described in Type C (Hub-Swarm Architectures) below. It is also worth noting that each Type G node is part of the sensor grid as well as of the C2 and engagement grids. Sometimes it is useful to split each node into sensor, C2, and engagement sub-nodes which are then networked separately. This is particularly beneficial for physically large nodes like ships, where there is a good chance that the destruction of on-board sensors may leave weapons untouched, or vice versa. Such a splitting results in a Type F (Mixed) architecture, as described below.

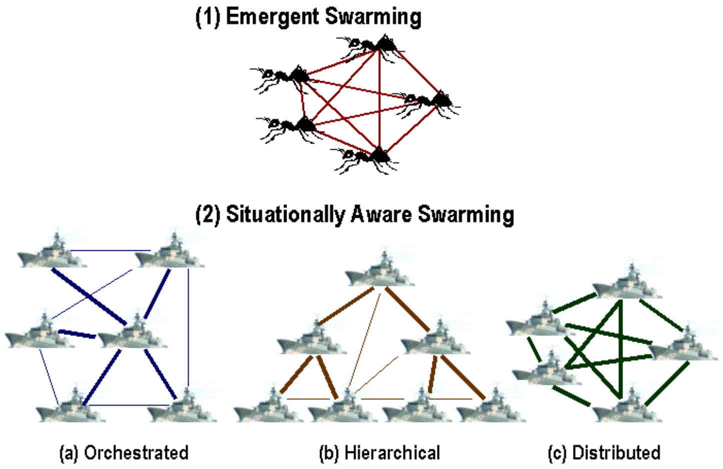


Figure 4: Swarming (Type G) Architectures

Appropriate networks for swarming architectures include symmetrical networks where each node “looks the same,” or networks where connections are made at random. Theoretical work has shown that both forms of network can be robust in the face of node destruction (Dekker & Colbert 2004).

4.1 Type G1: Emergent Swarming

Emergent Swarming occurs in nature among insects such as ants (Gordon 1999):

> "*The basic mystery about ant colonies is that there is no management. A functioning organization with no one in charge is so unlike the way humans operate as to be virtually inconceivable. There is no central control. ... No ant is able to assess the global needs of the colony, or to count how many workers are engaged in each task and decide how many should be allocated differently. The capacities of individuals are limited. Each worker need make only fairly simple decisions*."

For example, in far northern Australia, "magnetic termites" build large termite mounds which are oriented north-south and contain a complex ventilation system which controls temperature, humidity, and oxygen levels. But termite brains are too small to store a plan for such a complex system, and since they are blind, they have no situational awareness of how much progress they are making. Instead, the termite mound structure emerges as a result of the termites following very simple rules, and exchanging very simple pheromone signals (Solé & Goodwin 2000).

This style of operation has received considerable interest in the United States (US ASD C3I 2003). Although it may suit termites, it does not suit human beings (at least in Western armed forces). Following mindless rules without situational awareness would be tremendously corrosive of morale in a combat situation, and would have significant risks, such as that of reinforcing defeat. However, this style of operation is ideal for low-cost autonomous aerial (UAV), underwater (UUV), or terrestrial robotic devices. One possible example is the Area Dominance Munition (ADM), which the US Air Force is developing (Jane's 2003*a*). This is an expendable air-delivered UAV designed to loiter over enemy lines, and deploy multi-purpose shaped-charge warheads when targets are detected. Finding targets is done by sharing the limited information collected by onboard sensors (although this does raise ROE issues). Early experience with termite-like behaviour in robots is promising (Holland & Melhuish 1999), but much work is still required.

The rules which nodes would follow in emergent swarming would probably need to be fine-tuned beforehand using, for example, genetic algorithms (Goldberg 1989, Smith *et al* 2004). This is a process which mimics evolution in nature and has proven itself very successful in making difficult design decisions. The evolution process would need to be combined with an agent-based simulation environment, in order to evaluate performance.

4.2 Type G2: Situationally Aware Swarming

Situationally Aware Swarming uses networking to fuse sensor information from individual nodes to produce an integrated situational awareness picture, and also to synchronise actions. There are three basic ways of doing this, which have been developed in Distributed Database Theory, an area of Computer Science that has studied information flow in networks extensively (Ceri & Pelagatti 1984, Mullender 1993). Figure 4 summarises the three models.

Type G2(a): Orchestrated Swarming — In Orchestrated Swarming, one of the nodes is chosen as a temporary "leader." In the Centralised Architecture, the C2 node was the node best equipped for command and control activities, but in Swarming Architectures, all the nodes are identical. The choice of "leader" is therefore made on the basis of suitable position, current combat situation, or other transient factors. This approach is sometimes used in Special Forces teams, where members can, if necessary, take over command from the nominal commander.

Sensor data is sent to the "leader" node, where it is fused to produce an integrated situational awareness picture and an integrated plan of action. These are then broadcast to the other nodes. If the leader is unable to continue for any reason, the nodes agree on a replacement, which takes up where the previous leader left off. This approach limits network traffic, but it puts great stress on the C2 capability of the leader, since all the (identical) nodes in Type G architectures have limited C2 capability. This option is therefore not suitable for very difficult problems, or for a very large number of nodes. However, Orchestrated Swarming potentially produces better plans than other Swarming techniques, provided that the C2 capability of the "leader" is not overloaded.

Type G2(b): Hierarchical Swarming — Hierarchical Swarming is closest to the traditional military C2 architectures, and this is because it represents an extremely good solution for dealing with complex problems. Of course, the people in a traditional military hierarchy are not identical "nodes," but something resembling Hierarchical Swarming is used because human beings share many of the same limitations.

In Hierarchical Swarming, the nodes are organised into a hierarchy. In the event of nodes being lost, the hierarchy is maintained by promoting other nodes. Situational awareness information is fused going up the hierarchy, and at the same time, low-level tactical detail is dropped out. This means that the commanding node gets the "big picture" situation awareness that it needs. This simplifies the situational awareness fusion problem and avoids over-straining the information fusion capability of nodes. The commanding node then produces a "big picture" plan (often called an "intent"). This is passed down the hierarchy, and tactical detail is added by subordinate nodes. This avoids over-straining the planning capability of nodes.

In the absence of computer technology, such a hierarchy has been the most effective mechanism of command. However, it is not very fast, and some of the other Swarming approaches allow for more rapid response.

Type G2(c): Distributed Swarming — Distributed Swarming has no "leader" role, and all decisions are made through consensus. Situational awareness is handled by all nodes broadcasting their sensor information, so that every node builds up an individual situational awareness picture. This generates a large amount of network traffic, but if the network can handle the traffic, it is extremely fast.

There are two ways of handling planning with a Distributed Swarming architecture. The first has been called "collective" or "borg" decision-making (Wheeler & White 2004). In this style, each node goes through exactly the same decision process that the "leader" would have gone through if there was one, and then carries out the role that it assigns itself. This strategy only works for simple problems, where there is a single best decision, and each node therefore comes up with the same plan. This strategy also requires each node to have exactly the same situational awareness information, and to make decisions in a totally predictable way. For most military problems, these circumstances will not apply.

The second style of planning in Distributed Swarming has been called Mission Agreement (Lambert & Scholz 2005) or Negotiation (Dekker 2002). In this style, each node has its own individual plan, and these plans are **synchronised** with each other through a negotiation process. Figure 5 illustrates this process for a situation where six vehicles wish to cross some complex terrain together. Initially, each vehicle has a different suggested route. Each vehicle negotiates with its neighbour in a ring network, and after four negotiation steps, they have reached a consensus on the path which is best for the team as a whole. This approach is more widely applicable than the "collective" or "borg" strategy, but it can be very slow, and requires a large amount of network traffic. Human factors may also mean that Distributed Swarming works best with some degree of cultural uniformity (Fewell & Hazen 2005).

In practice, the three approaches to Situationally Aware Swarming can be combined. For example, the US Navy's **Cooperative Engagement Capability** air defence system (CEC) uses Distributed Swarming for situational awareness information (which in this case is engagement-quality raw radar data), but Orchestrated Swarming for target selection, with a "global scheduler" in the "leader" ship (Jane's 2000, 2003*b*). This can be viewed as applying Distributed Swarming within the "sensor grid" and Orchestrated Swarming within the "C2 grid" (although CEC does not actually have distinct sensor and C2 grids).

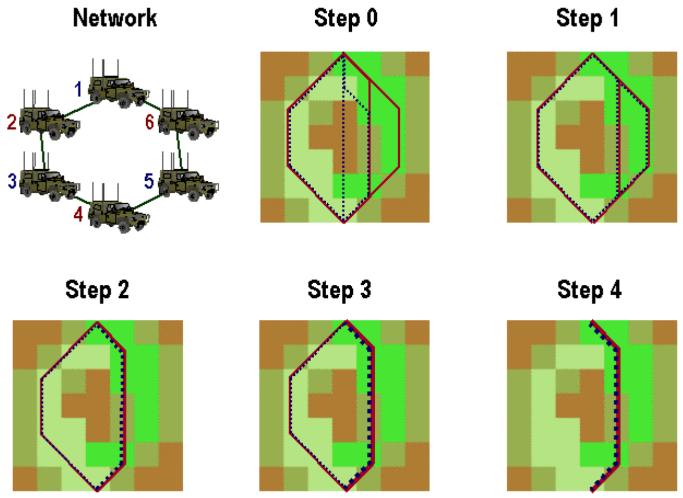


Figure 5: An Example of Mission Agreement for Distributed Swarming

This completes our survey of the three "extreme" types of NCW: Centralised (Type A), Request-Based (Type E), and Swarming (Type G). We now describe the intermediate types B, C, D, and F, which are combinations of these extremes.

5. TYPE F: MIXED SWARMING/REQUEST-BASED NCW

Type F architectures are value-symmetric but only partly homogenous, i.e. nodes are partly similar and partly different. NCW using such nodes is therefore a combination of Swarming Architectures (to the extent that nodes are similar) and Request-Based Architectures (when differences are important). There are two main subtypes: architectures with limited node types, and architectures with node commonality, as shown in Figure 6.

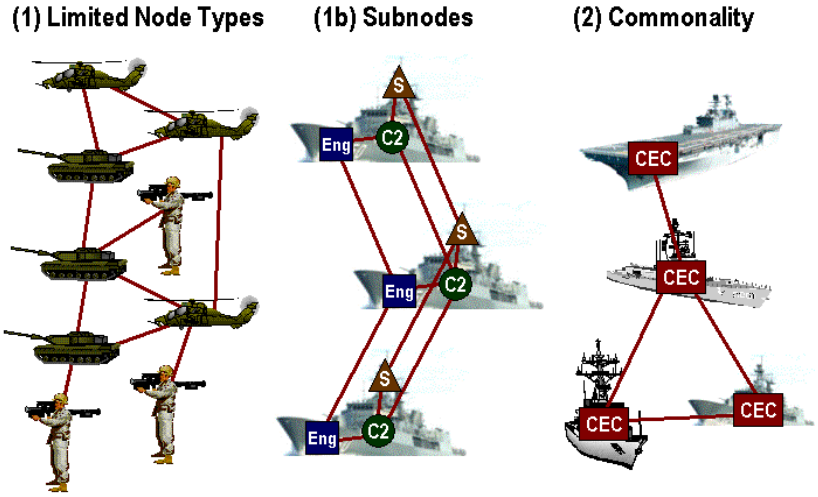


Figure 6: Mixed (Type F) Architectures

5.1 Type F1: Limited Node Types

In Type F1 architectures, there are only a limited number of kinds of node. For example, in the US Army's **Future Combat Systems** (FCS) project (Riggs 2003), there are 16 kinds of nodes, including C2 vehicles, NLOS cannons, medical vehicles, UAV's, unattended ground sensors, etc. Amongst nodes of the same kind, Swarming behaviour can occur. For example, Swarming occurs amongst sensor nodes as they cooperate to produce a fused situational awareness picture (using Hierarchical Swarming). Swarming also occurs amongst combat nodes as they cooperate to carry out their mission (using Distributed Swarming). Between different kinds of nodes, requests (e.g. for fire support) must be used.

A special case of type F1 occurs when a collection of identical platforms is split up into sensor, C2, and engagement sub-nodes, as previously discussed. The resulting network then has three kinds of node, with Swarming behaviour between nodes of the same type. The collection of all the sensor nodes is called the **sensor grid**, and similarly the **C2 grid** and the **engagement grid**. Between the different grids, Request-Based operation occurs.

5.2 Type F2: Node Commonality

In Type F2 architectures, all the nodes have some common characteristics. At the human level, this occurs, for example, in Special Forces teams, where members have the same core skills, but also fill specialised roles such as medic, communications, explosives, etc. This means that Swarming can occur for activities where the nodes are similar, while Request-Based operation occurs for the specialised roles.

The US Navy's CEC air defence system (Jane's 2000, 2003*b*) relies on ships having significant commonality in terms of sensors, computer software, and networking. Different radar systems are possible, as long as they satisfy consistent interfaces. This commonality enables a combined form of Swarming, as described above. However, there are also significant differences between ships, which means that Request-Based operation is needed for specialised roles. Ships using CEC can therefore be described as having, overall, a Type F2 architecture.

6. TYPE C: HUB-SWARM NCW

Hub-Swarm architectures result from taking a Type G Swarming architecture, and adding a high-value "hub" which acts as a force multiplier, while retaining the swarming behaviour. For example, a flotilla of frigates would be a Type G2 architecture, but the addition of an Air Warfare Destroyer (AWD) as a "hub" results in Type C. Similarly, if fighter aircraft are combined with an AWACS aircraft while retaining some self-synchronisation ability, the result would be Type C. Another example is the US Air Force Area Dominance Munition (Jane's 2003*a*), when combined with a centralised "eye in the sky" ISR asset.

The major issue in Type C architectures is to ensure that the "hub" is effectively utilised, and properly protected, while retaining the Swarming behaviour, i.e. not introducing fully centralised control.

7. TYPE B: HUB-REQUEST NCW

Adding a high-value "hub" to a Type E Request-Based architecture results in Type B. This integration is easy to do, since the "hub" provides a service and responds to requests in much the same way as the other nodes. The high-value of the "hub" means that its services will be in high demand, and some method is required to prioritise and balance requests, but this was already true for Type E Request-Based architectures. The potential vulnerability of the "hub" means that it must be protected, and this can be achieved by using high-priority requests to combat nodes to provide this protection when needed.

An example of a Type B Hub-Request architecture would be the Vietnam-era Fire Support Base (McAulay 1986). It would act as a "hub," responding to fire-support requests from units in the field. However, if it was itself threatened it would issue a priority request to all units to help defend it.

8. TYPE D: JOINT NCW

Type D architectures involve a mix of nodes of different kinds and values. Such a mix arises particularly in a Joint force, and involves mixing elements of all the other types of NCW. How this mixture should be achieved is dependent on understanding fully the other types. High-value nodes within a Joint architecture will behave more or less like "hubs." Groups of similar nodes will to some extent display Swarming behaviour. In addition, requests will be passed between different kinds of node. Achieving such a "seamless" Joint force will therefore require exploring the other types of NCW at least at the concept demonstrator level.

9. CONCLUSIONS

In this paper, we have explored seven possible NCW architectures, summarised in Figure 7 and Table 2. Since there is no "one size fits all" NCW solution, all seven architectures are candidates for implementation in different parts of the overall ADF network. The choice of architecture will be influenced by our two basic concepts: **value symmetry** (do the nodes differ significantly in importance?) and **homogeneity/heterogeneity** (are the nodes similar or different?).

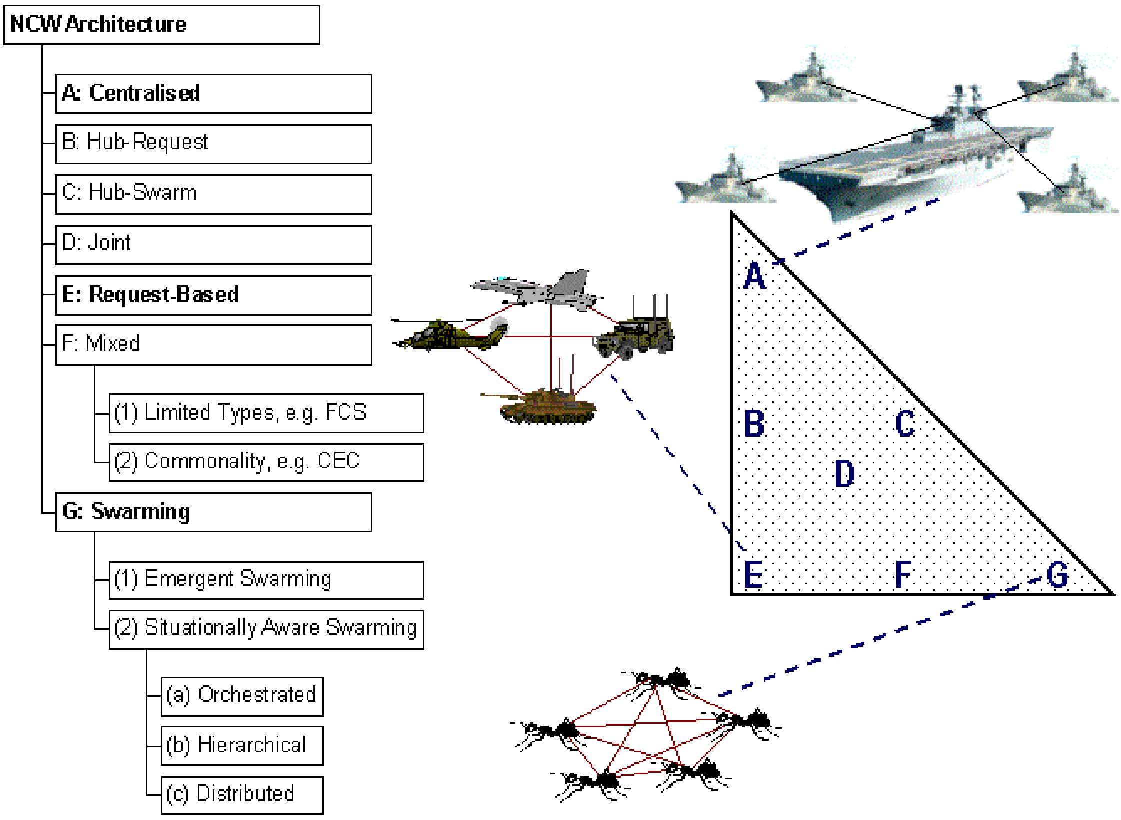


Figure 7: Summary of NCW Taxonomy

The utility of our taxonomy lies in the specific questions raised by each architecture, summarised in Table 1. For example, high-value "hubs" require tactics which gain the greatest benefit from the "hub," while at the same time protecting it. Architectures based on heterogenous nodes raise questions about interoperability and request prioritisation. Swarming architectures raise questions about the specific methods for reaching agreement — methods which have specific strengths and weaknesses. Addressing questions like these provides a way forward for the implementation of NCW within the ADF.

10. ACKNOWLEDGEMENTS

The author is grateful to Ed Kruzins, Bernard Colbert, Richard Taylor, Paul Main, Daniel Hall, Ruth Gani, Gina Kinston, Patrick Hew, Robert Mun, Terry Moon, and Robert Amos for discussions on these issues.

Table 2: Summary of the Seven NCW Architectures

Architecture			Characteristics
A: Centralised			One central high-value "hub," with other nodes of low value, networked and controlled via the "hub."
B: Hub-Request			Type E "Request-Based" plus one or more central high-value "hubs."
C: Hub-Swarm			Type G "Swarming" plus one or more central high-value "hubs."
D: Joint			Mixture of other 6 types.
E: Request-Based			Nodes of same value, but with different specialised capabilities. Requests for service between nodes of different kinds.
F: Mixed			Mixture of "Request-Based" and "Swarming."
	F1: Limited Types		Small number of node types (includes the case of separate sensor, engagement, and C2 grids).
	F2: Commonality		Nodes are different, but have significant commonality, e.g. CEC.
G: Swarming			Nodes identical, or nearly so.
	G1: Emergent Swarming		Nodes follow simple rules, like insects.
	G2: Situationally Aware Swarming		Nodes share information to build up Situational Awareness picture.
		G2(a): Orchestrated	One node is a temporary "leader"
		G2(b): Hierarchical	Nodes are arranged in a hierarchy.
		G2(c): Distributed	No leader or hierarchy.

11. REFERENCES

Australian Department of Defence, 2002*a*, *Force 2020*, June. Available online at http://www.defence.gov.au/publications/f2020.pdf

Australian Department of Defence, 2002*b*, *The Australian Approach to Warfare*, June. Available online at http://www.defence.gov.au/publications/taatw.pdf

Australian Department of Defence, 2003, *Australian Defence Force Network Centric Warfare Roadmap (First Draft)*, September.

Ceri, S. & Pelagatti, G., 1984, *Distributed Databases: Principles and Systems*, McGraw-Hill.

Clancy, T., 1995, *Fighter Wing: A Guided Tour of an Air Force Combat Wing*, HarperCollins.

Clancy, T., 1999, *Carrier: A Guided Tour of an Aircraft Carrier*, Sidgwick & Jackson.

Clark, T. & Moon, T., 2001, Interoperability for Joint and Coalition Operations," *Australian Defence Force Journal*, **151**, 23–36, Nov/Dec. Available online at http://www.defence.gov.au/publications/dfj/adfj151.pdf

Clausewitz, C., 1873, *On War*, Book 8, Chapter 4, translated by J.J. Graham.

Dekker, A.H., 2002, *Applying the FINC (Force, Intelligence, Networking and C2) Methodology to the Land Environment*, DSTO Report DSTO-GD-0341, October. Available online at http://www.dsto.defence.gov.au/corporate/reports/DSTO-GD-0341.pdf

Dekker, A.H., 2003*a*, Centralisation and Decentralisation in Network Centric Warfare, *Journal of Battlefield Technology*, **6** (2), 23–28, July.

Dekker, A.H., 2003*b*, Using Agent-Based Modelling to Study Organisational Performance and Cultural Differences, *Proc. MODSIM 2003 International Congress on Modelling and Simulation*, Townsville, July, Volume 4, 1793–1798.

Dekker, A.H., 2005, C4ISR, the FINC Methodology, and Operations in Urban Terrain, *Journal of Battlefield Technology*, **8** (1), March, 25–28.

Dekker, A.H. & Colbert, B., 2004, Network Robustness and Graph Topology, Proc. 27th Australasian Computer Science Conference, Dunedin, NZ, January. *Conferences in Research and Practice in Information Technology*, **26**, 359–368. V. Estivill-Castro, ed. Available online at http://crpit.com/confpapers/CRPITV26Dekker.pdf

Fewell, M.P. & Hazen, M.G., 2005, *Cognitive Issues in Modelling Network-Centric Command and Control*, DSTO Report DSTO-RR-0293, May. Available online at http://www.dsto.defence.gov.au/publications/3950/DSTO-RR-0293.pdf

Goldberg, D.E., 1989, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley.

Gordon, D., 1999, *Ants at Work: How an Insect Society is Organised*, Free Press.

Hall, D., Gani, R., Smith, N., Dong, L., Clark, T., Kingston, G. & and Bell, J., 2004, Networked Services, *Proceedings of the 9th International Command and Control Research and Technology Symposium*, Copenhagen, Denmark, September 14–16. Available online at http://www.dodccrp.org/events/2004/ICCRTS_Denmark/papers/018.pdf

Hecht-Nielsen, R., 1990, *Neurocomputing*, Addison-Wesley.

Holland, O. & Melhuish, C., 1999, Stigmergy, Self-Organization, and Sorting in Collective Robotics, *Artificial Life*, **5**, 173–202.

Jane's Information Group, 2000, Delivering CEC potential, *Jane's Navy International*, March 01.

Jane's Information Group, 2003*a*, US Air Force eyes loitering battlefield munition, *Jane's International Defence Review*, January 01.

Jane's Information Group, 2003*b*, CEC, *Jane's Naval Weapon Systems*, **38**.

Lambert, D.A. & Scholz, J., 2005, A Dialectic for Network Centric Warfare, *Proceedings of the 10th International Command and Control Research and Technology Symposium (ICCRTS)*, MacLean, VA, June 13–16. Available online at http://www.dodccrp.org/events/2005/10th/CD/papers/016.pdf

Lind, W.S., 1985, *Maneuver Warfare Handbook*, Westview Press.

McAulay, L., 1986, *The Battle of Long Tan*, Arrow Books.

Mullender, S., 1993, *Distributed Systems*, 2nd edition, Addison-Wesley.

Oaks, S. & Wong, H., 2002, *Jini in a Nutshell*, O'Reilly.

Riggs, J.M., 2003, The Objective Force, in *Proceedings of "Swarming: Network Enabled C4ISR" Conference*, Tysons Corner, VA, January.

Smith, T., Prekop, P. & Burnett, M., 2004, Deaf, Dumb and Stupid: Harnessing evolution to create successful and adaptive strategies for automated UAVs, *Presentation at 1st Complex Adaptive Systems in Defence Workshop*, Adelaide University, Australia, 22–23 July.

Solé, R. & Goodwin, B., 2000, *Signs of Life: How Complexity Pervades Biology*, Basic Books.

US ASD C3I, 2003, *Proceedings of "Swarming: Network Enabled C4ISR" Conference*, Tysons Corner, VA, January.

Ward, N., 1992, *Sea Harrier over the Falklands*, Cassell.

Wheeler, S. & White, G., 2004, Swarm or Borg? A comparison of Distributed and Collective Control, *Proc. 7th Asia-Pacific Conference on Complex Systems*, Cairns, Australia, 6–10 December.

Zolin, Z., Hinds, P.J., Fruchter, R. & Levitt, R.E., 2004, Interpersonal trust in cross-functional, geographically distributed work: A longitudinal study, *Information and Organization*, **14**, 1–26.