

Leuchter, S. & Schönbein, R. (2006). Die Verwendung von Architectural Frameworks als Vorgehensmodell für die System-of-Systems-Entwicklung. In: C. Hochberger, R. Liskowsky (Hrsg.), *INFORMATIK 2006. Informatik für Menschen. Beiträge der 36. Jahrestagung der Gesellschaft für Informatik e.V. (GI), 2. bis 6. Oktober 2006 in Dresden* (Band 1, S. 669-675). Bonn: Gesellschaft für Informatik (LNI; P-93). <http://www.safety-critical.de/doc/informatik2006.pdf>

Die Verwendung von Architectural Frameworks als Vorgehensmodell für die System-of-Systems-Entwicklung

Sandro Leuchter & Rainer Schönbein

Abt. Interoperabilität und Assistenzsysteme
Fraunhofer-Institut für Informations- und Datenverarbeitung IITB
Fraunhoferstr. 1, D-76131 Karlsruhe
vorname.nachname@iitb.fraunhofer.de

Abstract: Ein *System of Systems* ist ein komplexer Verbundzusammenschluss von bis dahin einzelnen Systemen. Probleme bei der Integration von eigenständigen Systemen treten vor allem im Bereich der Interoperabilität, also bei der Gestaltung der Schnittstellen, den auszutauschenden Informationen und bei der Unterstützung der Benutzer auf. *Architectural Frameworks* sollen die Entwicklung solcher komplexen Systeme leiten. Dazu werden sowohl Vorgaben zur Software-Architektur gemacht, als auch ein Phasenmodell des Entwurfsprozesses vorgegeben. Im Bereich der Entwicklung von vernetzten Systemen für die Bundeswehr wird das *NATO Architectural Framework* (NAF) verwendet. Es enthält keine explizite Beschreibung des Entwicklungsprozesses. Implizit gibt es eine Reihenfolge von Aktivitäten vor. In diesem Beitrag wird die Reihenfolge explizit dargestellt. Eine Integration als Projektdurchführungsstrategie in das V-Modell® XT wird als Ausblick erörtert.

1 Entwicklung von Systems of Systems

Es gibt in der Luftfahrttechnik und bei der wehrtechnischen Industrie den Trend zur Herstellung von Verbänden von Einzelsystemen. Ein Beispiel für solch einen Verbund ist der Sensorverbund GSSW [SM+04]. Hier wurden bestehende Mittel der Fernerkundung und Aufklärung in einer agentenbasierten Softwarearchitektur miteinander verbunden. Die einzelnen Systeme funktionieren weiter in ihrer ursprünglichen Funktionalität. Zusätzlich können sie zusammen genutzt werden, um komplexere Aufgaben zu lösen. Dazu wird ein Plan generiert, welche Teilfähigkeiten der im Verbund integrierten Mittel in welcher Sequenz genutzt werden müssen. Die dabei entstehenden Teilergebnisse werden zwischen den Systemen ausgetauscht.

Die Entwicklung von solchen hochkomplexen Systems-of-Systems (SoS) geschieht durch die Kopplung existierender – bereits in sich komplexer – Systeme. Bei den Verbänden handelt es sich aus unterschiedlichen Gründen um lose gekoppelte Systeme. Folgende fünf Merkmale kennzeichnen solche Systeme: Operationelle und administrative Unabhängigkeit untereinander, räumliche Verteilung, emergentes Verhalten und evolutionäre Entwicklung [SC01].

Im Gegensatz dazu wird das Konzept Family-of-Systems unterschieden. Hier werden durch den Zusammenschluss keine über die einzelnen Teilfunktionalitäten der verbundenen Systeme hinausgehenden Fähigkeiten (emergentes Verhalten) erzeugt.

Die Entwicklung von SoS muss planerisch, organisatorisch und technisch die Integration von Teilfähigkeiten zu einem emergenten Verbundkomplex verwirklichen. Dazu müssen die Teilfähigkeiten identifiziert, analysiert und interoperabel gemacht werden. Das kann sich sowohl auf verteilte Datenbestände, die in einem einheitlichen Informationsraum zusammengefasst werden müssen, als auch auf algorithmische Funktionalitäten und durch Menschen erbrachte Dienstleistungen, die allen Teilsystemen zur Verfügung gestellt werden müssen, beziehen.

Um diesen Ansatz umzusetzen, ist technisch eine lose koppelnde verteilte Architektur erforderlich. Sie kann z.B. durch eine service-orientierte oder eine agentenbasierte Infrastruktur realisiert werden. Um emergentes Verhalten zu erzeugen, müssen die zu koppelnden Systeme miteinander interagieren. Eine Voraussetzung dazu ist, dass die Systeme Nachrichten austauschen und dadurch zusammenarbeiten können. Im Bereich der SoS-Integration wird diese Fähigkeit Interoperabilität genannt.

Interoperabilität kann auf technischer Ebene, syntaktischer, semantischer und pragmatischer Ebene hergestellt werden:

- Auf technischer Ebene geht es um die Voraussetzung, Daten austauschen zu können (Bsp.: Systeme sind mit serviceorientierter Architektur gekoppelt).
- Auf syntaktischer Ebene sind gemeinsam benutzte Datenformate (Bsp.: XML) vorzusehen.
- Auf semantischer Ebene werden Voraussetzungen für eine systemübergreifende Interpretation von Nachrichten und gemeinsam verwendeten Begriffen geschaffen (Bsp.: ontologiebasierte Integrationsfunktion).
- Auf pragmatischer Ebene wird dann eine einheitliche Beurteilung der Information geschaffen (Bsp.: Ableitung erforderlicher Handlungen).

Diese vier Ebenen bauen aufeinander auf, d.h. Voraussetzung für syntaktische Interoperabilität ist technische Interoperabilität etc.

2 Architekturgeleitete Entwicklung

Die grundlegende Organisation von Softwaresystemen wiederholt sich immer wieder [Has06]. Wurde eine konkrete Organisation mehrfach erfolgreich angewendet hat sie den Status einer Software-Architektur, die den Entwurf neuer Systeme leiten sollte. Die Architektur lässt sich durch die Beschreibung der verwendeten Komponenten sowie deren Beziehungen zueinander und zur Umgebung beschreiben. Ebenso sind die

Prinzipien, die den Entwurf und die Evolution des Systems bestimmen, Teil der Architektur.

In der NATO herrscht Übereinstimmung, dass ein Schlüsselfaktor für die Sicherstellung interoperabler und kosteneffektiver militärischer Systeme die Unterstützung durch Architekturvorgaben ist. Ein Weg dies zu erreichen sind Architectural Frameworks (AF). Ein AF besteht aus der Dokumentation einer Software-Architektur und aus Vorgaben, die die Entwicklung leiten. Viele AF werden durch Entwicklungswerkzeuge unterstützt. Für einige Anwendungsbereiche wurden bereits von unterschiedlichen Anbietern AFs entwickelt: Für den Bereich der Unternehmensanwendungen das „Zachman AF“, das „The Open Group Architecture Framework“ (TOGAF) sowie das „IT City Planning“ AF. Für die US Finanzbehörden ist das „Treasury Enterprise Architecture Framework“ (TEAF) entwickelt worden, für andere US Bundesbehörden das „Federal Enterprise Architecture Framework“ (FEAF).

3 NATO Architectural Framework

Im Bereich der Entwicklung von SoS im militärischen Bereich wurde im Rahmen der NATO das NATO Architectural Framework (NAF) entwickelt [Nat04]. Die Verwendung dieses AF ist eine verbindliche Vorgabe der technischen Architektur der Bundeswehr (TABw), die Standards für IT-Systeme der Bundeswehr definiert.

Die Vorgabe eines AF für die NATO hat zum Ziel, die Produktqualität von SoS zu verbessern, indem Anforderungen in strukturierter Weise dokumentiert werden, und die Komplexität der Entwicklung durch erprobte Sichten und Modellierungsschritte beherrschbar zu machen. Das Herstellen von Interoperabilität zwischen vernetzten Systemen von Bündnispartnern soll vereinfacht werden, indem eine einheitliche Entwicklungsdokumentationsvorgaben geschaffen wird, durch die ein gemeinsames Verständnis der jeweiligen Operationsprinzipien und Schnittstellen erreicht werden kann.

Für das NAF wurde eine grundlegende Trennung in die drei Sichten operationelle Architektur, Systemarchitektur und technische Architektur eingeführt. Alle Sichten beschreiben Produkte, die im Entwicklungsprozess zu erstellen sind. Die Produkte bauen untereinander auf, so dass Abhängigkeiten zwischen allen Sichten bestehen (s. Abbildung 1). Die operationelle Sicht beschreibt Produkte, die die Sicht der Nutzer wiedergeben. Es geht hier um operationelle Anforderungen und die Nachrichtenflüsse zwischen militärischen Organisationseinheiten. Die Systemsicht wird aus der operationellen Sicht abgeleitet und enthält einen konkreten Entwurf der technischen Komponenten, die die operationelle Sicht umsetzen. Die Fähigkeiten werden mit den operationellen Anforderungen in Beziehung gesetzt. Die technische Sicht schließlich beinhaltet Produkte des Entwicklungsprozesses, die die Komponenten der Implementierungsplattform festlegen und beschreiben. Es werden technische Standards und Konventionen ausgewählt und festgelegt.

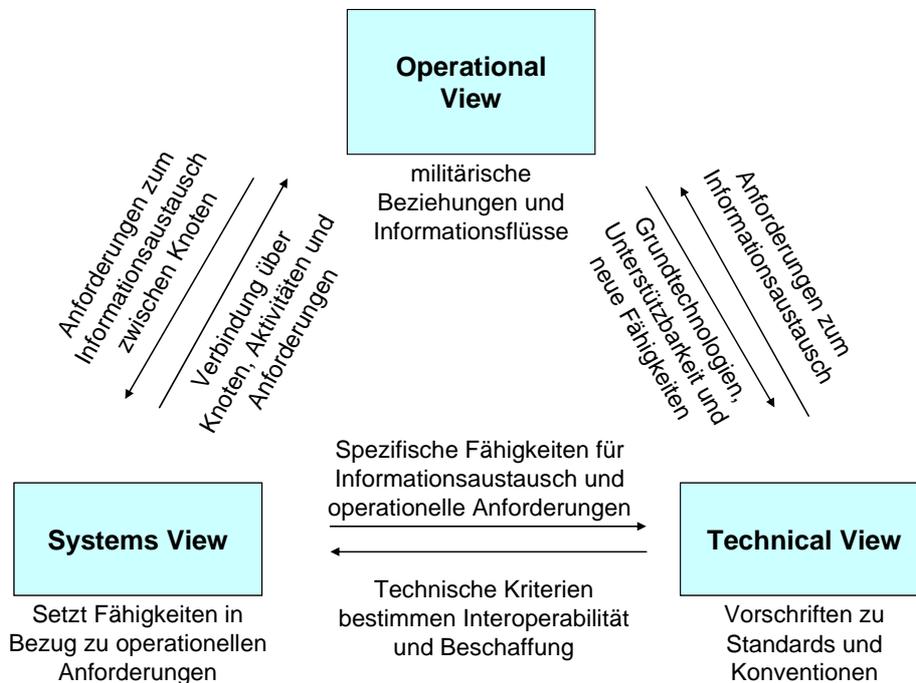


Abbildung 1: C4ISR AF, Trennung in drei Sichten nach [C4i97]

3.1 All View

- Overview and Summary Information (NAF All View: NAV-1): Eine kurze einführende Beschreibung der Mission des Entwicklungsprojektes.
- Integrated Dictionary (NAV-2): Das Glossar, das für alle Projektphasen verbindlich ist.

3.2 Operational View

- High-Level Operational Concept Graphic (NAF Operational View: NOV-1): Eine visuelle Beschreibung des Einsatzkonzeptes, das durch das zu entwickelnde System unterstützt werden soll.
- Operational Node Connectivity Description (NOV-2): Relevante organisatorische Einheiten und ihre Interaktion werden beschrieben.
- Operational Information Exchange Matrix (NOV-3): Der Informationsfluss zwischen den relevanten organisatorischen Einheiten wird beschrieben.
- Command Relationships Chart (NOV-4): Die Verantwortlichkeiten zwischen organisatorischen Einheiten werden beschrieben.
- Activity Model (NOV-5): Der Ablauf der Prozesse, die unterstützt werden, wird beschrieben.

- Concept Data Model (NOV-6): Art und Struktur der Daten, die im System verwendet werden.

3.3 System View

- System Interface Description (NAF System View: NSV-1): Die Komponenten, aus denen das System besteht, und die Verbindungen zwischen den Komponenten werden beschrieben.
- Systems Communications Description (NSV-2): Die Datenfluss zwischen den Systemkomponenten werden detailliert beschrieben.
- System-to-System Matrix (NSV-3): Die Art und Mittel der Kommunikation zwischen Systemknoten wird beschrieben.
- Systems Functionality Description (NSV-4): Prozesse und Datenflüsse im System werden beschrieben.
- Operational Activity to System Function Traceability Matrix (NSV-5): Systemfunktionen und operationelle Aktivitäten werden in einer Tabelle abgeglichen.
- System Information Exchange Matrix (NSV-6): Input-/Outputbeziehungen zwischen allen Komponenten werden spezifiziert.
- System Performance Parameters Matrix (NSV-7): Anforderungen und Abschätzungen für die Performanz von allen Kommunikationsbeziehungen werden angegeben.
- System Evolution Description (NSV-8): Eine Roadmap für die iterative Entwicklung des Systems wird angegeben.
- System Technology Forecast (NSV-9): Die Weiterentwicklung relevanter Technologiegebiete und Fähigkeiten wird in drei Zeitskalen (0-6 Monate, 6-18 Monate, mehr als 18 Monate) abgeschätzt.
- Physical Data Model (NSV-10): Eine Beschreibung des Datenmodells der Komponenten auf Datenbankebene.

3.4 Technical View

- System Standards Profile (NAF Technical View: NTV-1): Die technischen Standards der Implementierung der Architektur werden beschrieben.
- Standards Technology Forecast (NTV-2): Die Entwicklung der verwendeten Standards nach Dienstbereich wird abgeschätzt (bis zu zwei bzw. fünf Jahre).
- Technical Configurations (NTV-3): Eine detaillierte Beschreibung der Implementierung der genutzten Dienstbereiche der verwendeten Systeme.
- Software Configurations (NTV-4): Eine detaillierte Beschreibung der Konfiguration der genutzten Dienstbereiche der verwendeten Systeme.
- Product Selection Report (NTV-5): Die Eigenschaften der Produkte, mit denen Dienstbereiche in den verwendeten Systemen implementiert werden, werden beschrieben.

3.5 Entwicklungsprozess mit NAF

NAF beinhaltet keine expliziten Vorschriften zum Ablauf der Erstellung der Produkte. Nur die Aktivitäten als solche und die resultierenden Produkte sind spezifiziert. Andere Konzepte aus Vorgehensmodellen wie Rollen und Entscheidungspunkte fehlen in der NAF. Manche Produkte der NAF werden direkt durch Transformation aus anderen abgeleitet, manche entstehen aus der Fusion mehrerer Vorprodukte. Implizit ist dadurch eine Reihenfolge der Aktivitäten, die zu den Produkten führen, gegeben. Abbildung 2 gibt einen Überblick über die Abhängigkeiten der Produkte des NAF. Der Graph beschreibt damit gleichzeitig eine Bearbeitungsreihenfolge, wobei die Abhängigkeiten zwischen z.B. der NSV-4 und NSV-5 zeigen, dass es sich nicht um eine streng sequentielle Abfolge handelt. Diese Abfolge wurde in mehreren Entwicklungsprojekten erfolgreich erprobt. Es wurde Bedarf identifiziert, die Vorgaben zum Entwicklungsprozess aus dem NAF in ein Vorgehensmodell gemäß V-Modell zu integrieren.

4 Ausblick

Im Bereich der Bundeswehr werden SoS Entwicklungen im Rahmen der Beschaffung, gesteuert durch den CPM¹-Prozess, und im Rahmen von Forschungs- und Technologiestudien, gesteuert durch den CD&E²-Prozess, beauftragt. Die Verwendung des NAF als Rahmen für die Spezifikation und den Systementwurf ist für Vorhaben der Bundeswehr verpflichtend. Als übergeordnetes Vorgehensmodell für die Projektabwicklung ist weiterhin das V-Modell® XT [KBSt06] im Bereich CPM und in vielen Vorhaben im Bereich CD&E vorgeschrieben. Eine Integration von beiden Prozessen ist im Rahmen einer Projektdurchführungsstrategie oder eines Vorgehensbausteins für das NAF im V-Modell® XT möglich und wünschenswert, um den Vorgaben innerhalb eines durchgehenden Vorgehensmodells gerecht zu werden und die Entwicklungs- und Projektmanagementarbeit effektiv und effizient zu gestalten.

Ein erster Entwurf einer Projektdurchführungsstrategie für NAF-basierte Entwicklungen erfolgt gegenwärtig auf der Basis der inkrementellen und der komponentenbasierten Systementwicklung im V-Modell® XT. Dazu werden Aktivitäten, Entscheidungspunkte und Rollen der vorhandenen Projektdurchführungsstrategien mit denen, die aus dem NAF abgeleitet werden können, gegenübergestellt und aufeinander abgebildet. Die Aktivitäten ergeben sich direkt aus den einzelnen Sichten der NAF. Die Erstellung jedes Produktes wird in erster Näherung als eine einzelne Aktivität angesehen. Ob zur Spezifikation des Entwicklungsprozesses noch zusätzliche Annahmen zu weiteren Aktivitäten getroffen werden müssen, ist noch nicht geklärt. So fehlen in der NAF beispielsweise Informationen zu den Methoden der Anforderungserhebung und zu anderen Aspekten der Nutzerpartizipation.

¹ CPM: Customer Product Management. Vorgehen der Bundeswehr bei der Beschaffung. Eine Reihe von Maßnahmen soll den Bedarf ermitteln und die Qualität des Beschafften sicherstellen.

² CD&E: Concept Development and Experimentation. Iteratives Vorgehen der Bundeswehr bei der Entwicklung und Erprobung neuer Technologien und Vorgehensweisen bis zur operationellen Einführung.

In der Anwendung der NAF werden einige Produkte als optional angesehen (z.B. NOV-4, NOV-5), während andere zwingend erforderlich sind (z.B. NOV-2, NOV-3). Folglich kommen Aktivitäten und ein Entscheidungspunkt zur Auswahl der zu erstellenden NAF-Sichten hinzu. Den vier Bereichen (All, Operational, Systems, Technical) werden vier neue Analytiker-Rollen zugeordnet.

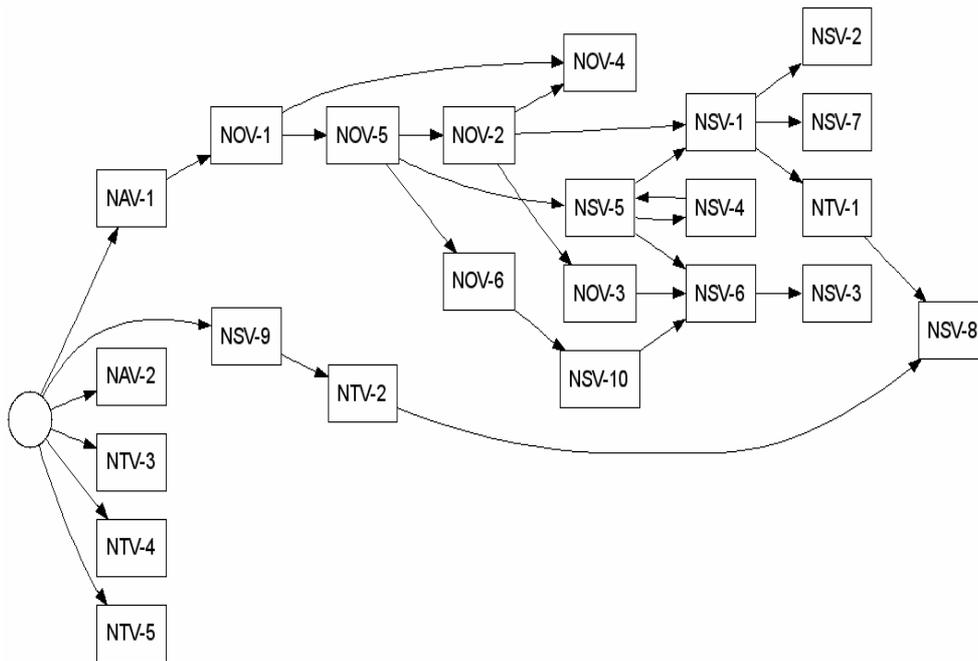


Abbildung 2: logische Abhängigkeiten zwischen den Sichten des NAF

Literaturverzeichnis

- [C4i97] C4ISR Architecture Working Group: C4ISR Architecture Framework Version 2.0, Dec. 18, 1997.
- [Has06] Hasselbring, W.: Software-Architektur. In: Informatik-Spektrum, 29 (1), 2006, S. 48-51.
- [Nat04] NATO C3 Board: NATO C3 System Architecture Framework (NAF) (Document AC/322-D(2004)0002 (INV)) NATO, 2004.
- [KBSt06] KBSt – Koordinierungs- und Beratungsstelle der Bundesregierung für Informationstechnik in der Bundesverwaltung: V-Modell@ XT. Online-Dokument <http://ftp.uni-kl.de/pub/v-modell-xt/Release-1.2/Dokumentation/pdf/V-Modell-XT-Komplett.pdf>.
- [SC01] Sage, A.P.; Cuppan, C.D.: On the Systems Engineering and Management of Systems of Systems and Federations of Systems. In: Information, Knowledge, and Systems Management, 2 (4), 2001, S. 325-345.
- [SM+04] Schönbein, R.; Mühlenberg, D.; Müller, W.; Pallmer, D.: Software Agents for semantic interoperability in German Smart Sensor Web. In: Military Applications of Agent Technology in ICT and Robotics. The Hague, NL: TNO, 2004.