

Inzicht in Architectuurkennis – Essentieel voor evolutie en samenwerking

Minstens zo belangrijk als de uiteindelijke IT-architectuur is de weg ernaartoe. In het proces om te komen tot een IT-architectuur nemen belanghebbenden (vaak IT- of softwarearchitecten) belangrijke ontwerpbeslissingen. Deze ontwerpbeslissingen inclusief de afwegingen die ertoe geleid hebben, zijn essentieel om vast te leggen. Deze architectuurkennis dient als basis voor het te ontwikkelen informatiesysteem en is nodig wanneer toekomstige aanpassingen aan de architectuur benodigd zijn tengevolge van diverse veranderingen.

GRIFFIN is een nationaal onderzoeksproject, medegefinancierd door NWO binnen het JACQUARD programma onder contractnummer 638.001.406. Het consortium bestaat uit twee universiteiten en vijf industriële partners. Vanuit de Vrije Universiteit en de Rijksuniversiteit Groningen werken in totaal zeven onderzoekers aan het centrale thema van dit project: het effectief en efficiënt managen van architectuurkennis. Voor vier van de betrokken onderzoekers zal dit project leiden tot een proefschrift. De vijf industriële partners – Stichting Astron, Philips, Getronics PinkRocade, LogicaCMG en DNV-CIBIT – hebben elk hun eigen concrete problemen en vraagstelling op het gebied van architectuurkennismanagement. Zij bewaken door hun lidmaatschap van de gebruikerscommissie, waar ook externe experts en toehoorders zitting in hebben, de praktische relevantie van de projectresultaten.

Recente publicaties vanuit het GRIFFIN-project concentreren zich rond de onderzoeksvragen

- Welke toolondersteuning is benodigd voor het delen van architectuurkennis?
- Hoe gebruiken architecten architectuurkennis in de praktijk?
- Hoe kan architectuurkennis gebruikt worden in multi-site ontwikkelsituaties?
- Wat zijn veel gebruikte kennismanagementstrategieën voor architectuurkennis?
- Welke elementen zijn essentieel om architectuurkennis op te slaan?

In de hierna volgende paragrafen wordt aan elk van deze vragen aandacht besteed. Een Engelse samenvatting van de artikelen waaraan wordt gerefereerd is te vinden aan het einde van dit overzicht. Hier vindt u ook informatie over de auteurs en aanwijzingen hoe u de volledige tekst van de artikelen kosteloos kunt verkrijgen.

Welke toolondersteuning is benodigd voor het delen van architectuurkennis?

Kennismanagement speelt een belangrijke rol tijdens het architectuurproces. Om de juiste afwegingen en beslissingen te nemen, hebben architecten veel relevante kennis van het te ontwikkelen software systeem en de context van het systeem nodig. Daarnaast is het belangrijk om op de hoogte te zijn van best practices in de organisatie, expertise van collega's op verschillende gebieden, en van recente standaarden en nieuwtjes binnen een bepaald domein. Al dit soort informatie definiëren wij als architectuurkennis. Een case study bij een grote IT-dienstverlener bracht de behoefte voor 'just-in-time architectuurkennis' aan het licht, wat betekent dat de juiste architectuurkennis te allen tijde bij de juiste personen

terechtkomt. In *EAGLE: Effective Tool Support for Sharing Architectural Knowledge* gaan Rik Farenhorst, Patricia Lago en Hans van Vliet dieper op dit onderwerp in en beschrijven zij het ontwerp voor een architectuurkennisportal. Dit portal biedt verschillende mechanismen om just-in-time architectuurkennis te ondersteunen, en is hierdoor een perfecte kennismanagement tool voor in het architectuurdomein.

Hoe gebruiken architecten architectuurkennis in de praktijk?

Enige tijd geleden is een uitgebreide enquête onder de Nederlandse architectengemeenschap gehouden. In deze enquête werden de deelnemers gevraagd om hun specifieke ervaringen omtrent het gebruik van architectuurkennis aan te geven. Het gebruik van architectuurkennis is in zogenaamde use cases weergegeven: korte, kernachtige zinnen die mogelijk gebruik van architectuurkennis aanduiden. De deelnemers gaven op deze manier het relatieve belang van architectuurkennis voor hun dagelijkse werkzaamheden aan. Op basis van een analyse van 107 volledige vragenlijsten is een beeld geconstrueerd van het gedrag van architecten ten aanzien van architectuurkennis. Over het algemeen blijkt dat belanghebbende een centrale rol innemen in het gebruik van architectuurkennis. Architectuurdocumentatie op maat (bijvoorbeeld volgens IEEE1471) wordt derhalve steeds meer gemeengoed. Ook vinden architecten het steeds meer van belang om de rationale van genomen beslissingen vast te leggen. Echter, de meerwaarde van het kunnen *minen* van deze verzameling architectuurkennis door het te beschouwen als een verzameling van ontwerpbeslissingen, is nog geen breed geaccepteerde gedachte. Verder valt op dat architectuurkennis met name gebruikt wordt om te zien of de architectuur en de implementatie nog *op schema* ligt: voldoet de architectuur aan de requirements, en geschiedt de implementatie binnen de kaders van de architectuur? Veel minder wordt architectuurkennis gebruikt om trade-off analyses of risicoanalyses uit te voeren. Blijkbaar zijn architecten van nature positief ingesteld; de architectuur is 'automatisch goed'.

Alle details over het uitgevoerde onderzoek, de vragenlijst en de resultaten kunt u lezen in het artikel *The Architect's Mindset* van Viktor Clerc, Patricia Lago en Hans van Vliet.

Hoe kan architectuurkennis gebruikt worden in multi-site ontwikkelsituaties?

Architectuurkennis dient als basis voor het verdere ontwerp en de implementatie van informatiesystemen. Veel grote IT-organisaties besluiten deze ontwerp- en implementatiefase te *off-shoren*. Door werk naar (nu nog) lage-lonen landen uit te besteden hopen zij onder andere kostenvoordelen te behalen. Multi-site softwareontwikkeling werpt ook echter nieuwe uitdagingen op die te maken hebben met culturele aspecten, samenwerking over lands- en tijdgrenzen heen. Architectuurkennis en, specifiek, architectuurregels kunnen helpen deze uitdagingen te overwinnen. Viktor Clerc, Patricia Lago en Hans van Vliet hebben onderzocht hoe twee organisaties de uitdagingen in multi-site softwareontwikkeling tegemoet treden. De twee organisaties zijn nogal verschillend: een grote organisatie die op vijf ontwikkelsites productlijn-gebaseerde software ontwikkeld op basis van volledig beschreven requirements, en een wat kleinere organisaties die op twee ontwikkelsites met behulp van een *agile* ontwikkelingsmethodiek in diverse iteraties werkende software realiseert. Het onderzoek toont aan dat naast regels voor architectuur ook diverse procesmaatregelen noodzakelijk zijn om alle uitdagingen het hoofd te bieden.

Een overzicht van alle uitdagingen in multi-site softwareontwikkeling en de maatregelen die de organisaties hebben genomen, kunt u lezen in het artikel *Global Software Development: Are Architectural Rules the Answer?* van Viktor Clerc, Patricia Lago en Hans van Vliet.

Wat zijn veel gebruikte kennismanagementstrategieën voor architectuurkennis?

De visie van architectuur als kennismanagementvraagstuk krijgt wereldwijd steeds meer aandacht. Diverse organisaties proberen tools en technieken te ontwikkelen en/of toe te passen om het architectuurproces in goede banen te leiden (en te houden). Een inventarisatie van deze tools en technieken laat zien dat zowel de praktische toepassing van deze technieken als het wetenschappelijk onderzoek daarnaar zich voornamelijk richt op 'codificatie': het expliciet maken van relevante architectuurkennis. Naast deze bewust gekozen strategie van codificatie blijkt onder de oppervlakte echter een tweede strategie een belangrijke rol te spelen; een strategie die in de literatuur bekend staat als 'personalisatie'. Deze kennismanagementstrategie, waar in het architectuurproces vaak niet bewust voor wordt gekozen, legt minder nadruk op de architectuurkennis zelf en meer op de vraag 'wie weet wat?'. De rol van deze onbewust gevolgde strategie suggereert dat tools en technieken voor architectuurkennismanagement meer ruimte zouden moeten bieden om ook personalisatie te faciliteren. Ook zouden organisaties zich beter bewust moeten zijn van de rol van personalisatie in het architectuurproces. Een bewuste keuze voor een hybride combinatie van codificatie en personalisatie lijkt meer recht te doen aan de complexiteit van het architectuurproces dan een eenzijdige focus op codificatie alleen.

Het artikel *Architectural Knowledge Management Strategies: Approaches in Research and Industry* van Muhammad Ali Babar, Remco C. de Boer, Torgeir Dingsøyr en Rik Farenhorst bevat alle details.

Welke elementen zijn essentieel om architectuurkennis op te slaan?

Eerdere inzichten uit het onderzoek naar architectuurkennis heeft geleid tot een model waarin elementen, benodigd voor het vastleggen en beheren van architectuurkennis, zijn genoemd en aan elkaar zijn gerelateerd. Dit model beschrijft elementen die gebruikt kunnen worden om architectuurbeslissingen te nemen, de architectuur te beschrijven, en het proces om deze beschrijvingen te maken en te gebruiken. Op basis van een viertal praktijkgerichte onderzoeken is dit model getoetst en compacter gemaakt. Vervolgens is gekeken naar diverse theorieën op het gebied van architectuurbeschrijving en architectuurkennis in de literatuur, en is getracht om dit compactere model te falsificeren: bevat het model geen ambigue ('dubbele') elementen? Bevat het model alle essentiële elementen die ook in de literatuur worden genoemd (wellicht in andere terminologie). Dit onderzoek toont aan dat het compacte, *core*, model voor architectuurkennis de essentiële elementen bevat om architectuurkennis op te slaan.

Meer over dit *core* model en het uitgevoerde onderzoek is te lezen in het artikel *Architectural Knowledge: Getting to the Core* van Remco C. de Boer, Rik Farenhorst, Patricia Lago, Hans van Vliet, Viktor Clerc en Anton Jansen.

EAGLE: Effective Tool Support for Sharing Architectural Knowledge. Rik Farenhorst, Patricia Lago, Hans van Vliet Knowledge management plays an important role in the software architecting process. Recently, this role has become more apparent by a paradigm shift that views a software architecture as the set of architectural design decisions it embodies. This shift has sparked the discussion in both research and practice on how to best facilitate sharing of so-called architectural knowledge, and how tools can best be employed. In order to design successful tool support for architectural knowledge sharing it is important to take into account what software architecting really entails. In this paper we define the main characteristics of architecting, based on observations in a large software development organization, and state-of-the-art literature in software architecture. Based on the defined characteristics, we determine how best practices known from knowledge management could be used to improve architectural knowledge sharing. This results in the definition of a set of desired properties of architectural knowledge sharing tools. Finally, we highlight the design and implementation of EAGLE, an architectural knowledge sharing portal that implements those properties.

Global Software Development: Are Architectural Rules the Answer?, Viktor Clerc, Patricia Lago, and Hans van Vliet

Global software development (GSD) faces additional challenges as compared to single-site software development. Some of the better known challenges include temporal, geographical, and socio-cultural differences. To overcome these challenges, organizations need to revert to measures in order to deliver software in time in a distributed setting. Some of these measures may exist in the form of architectural rules: principles and statements on the software architecture that must be complied with throughout the organization. From the GSD literature we distilled four main GSD challenges and seven sub-challenges, or issues. For each issue, we list possible solutions and observe that solutions to GSD challenges may be obtained by adhering to architectural rules. We present a study on how two organizations involved in GSD solve the GSD challenges and issues. One of the organizations mainly uses rules regulating the architecture of the product. The other organization does not emphasize these architectural rules but rather focuses on the joint team effort in establishing and committing to measures that mainly pertain to the architecture process. We conclude that rules regulating a combination of both proves valuable in handling GSD challenges.

The Architect's Mindset, Viktor Clerc, Patricia Lago, and Hans van Vliet

Software architecture and software architecture practices become increasingly important for information systems since they enable reasoning on the design of the system. The concept of architectural knowledge, i.e. architectural design decisions and the resulting design, plays a pivotal role in architecture. In order to get the most out of architectural knowledge, we need insight into the ways in which architectural knowledge is used. Currently, we lack this insight. We performed survey-based research in the Netherlands to collect feedback on the importance of architectural knowledge for the daily work of practitioners in architecture. We present our findings using two perspectives: the architectural roles practitioners fulfill and the architecture level practitioners are engaged in. We use these perspectives to construct and reflect on the architect's mindset on architectural knowledge. This mindset of architects reveals an approach which is focused on 'to create and communicate' rather than 'to review and maintain' an architecture.

Architectural Knowledge Management Strategies: Approaches in Research and Industry.

Muhammad Ali Babar, Remco C. de Boer, Torgeir Dingsøyr, Rik Farenhorst

The software architecture community has recently gained an increasing interest in managing architectural knowledge. However, up until now there have been no attempts to obtain an overview of the work in the field. In this paper we present a preliminary review on current approaches to architectural knowledge management. To this end, we compare approaches known from literature and encountered in industry with knowledge management theory. We found that in reports from research and practice there appears to be a preference to use the codification strategy. However, our observations of the software architecture industry show that organizations in general tend to use a personalization strategy unintentionally. This paper serves as a call for awareness of this gap between intention and reality, and questions the biased focus on intentional codification alone. We suggest to close this gap through focusing on hybrid approaches.

Architectural Knowledge: Getting to the Core

Remco C. de Boer, Rik Farenhorst, Patricia Lago, Hans van Vliet, Viktor Clerc, Anton Jansen

Different organizations or organizational units are likely to store and maintain different types of information about their software architectures. This inhibits effective management of architectural knowledge. We experimented with a model of architectural knowledge to characterize the use of architectural knowledge in four different organizations. Based on this experimentation we identified four perspectives on architectural knowledge management, and additionally adjusted the model to better align theory with practice. The refined model defines a minimal set of concepts with supposedly complete coverage of the architectural knowledge domain. Because of the minimalistic aspect of the model, we refer to it as a 'core model' of architectural knowledge. Supporting evidence for the validity of our model, i.e. the supposed complete coverage, has been obtained by an attempt to falsify this claim through a comparison with selected literature. Application of the core model to characterize the use of architectural knowledge indicates possible areas of improvement for architectural knowledge management in the four organizations

Over de auteurs



prof. dr. Hans van Vliet is hoogleraar Software Engineering aan de Vrije Universiteit in Amsterdam, en hoofd van de vakgroep Information Management en Software Engineering van dezelfde universiteit. Zijn onderzoek richt zich onder andere op softwarearchitectuur en software measurement. Eerder was hij onderzoeker bij het Centrum voor Wiskunde en Informatica te Amsterdam en bij het IBM Almaden Research Center in San Jose, Californie. Hij is de auteur van het boek “Software Engineering: Principles and Practice” uitgegeven door Wiley. Hans van Vliet is de wetenschappelijk verantwoordelijke voor de uitvoering van het GRIFFIN-project.

email: hans@cs.vu.nl



dr. Patricia Lago is universitair docent binnen de vakgroep Information Management and Software Engineering van de Vrije Universiteit Amsterdam. Haar onderzoek richt zich op softwarearchitectuur, het modelleren en delen van architectuurkennis, en service-orientatie. Zij was betrokken bij verschillende internationale en nationale onderzoeksprojecten. Ze is lid van de programmacommissie van verscheidene conferenties en reviewer voor diverse wetenschappelijke tijdschriften.

email: patricia@cs.vu.nl



dr. Paris Avgeriou is universitair docent aan het Instituut voor Wiskunde en Informatica van de Rijksuniversiteit Groningen. Hij was eerder als senior onderzoeker verbonden aan het Fraunhofer IPSI in Darmstadt (Duitsland) en het Laboratory for Advanced Software Systems van de Universiteit van Luxemburg. Paris is een van de oprichters van het World-Wide Institute of Software Architects. Zijn onderzoek richt zich op software engineering, met name op softwarearchitectuur, patterns, evolutie, business modeling en web engineering.

email: paris@cs.rug.nl



drs. Rik Farenhorst is promovendus aan de Vrije Universiteit te Amsterdam op het gebied van Software Engineering. Zijn onderzoek richt zich in het bijzonder op het succesvol toegankelijk maken en delen van architectuurkennis. Deze architectuurkennis, zoals architecturale ontwerpbeslissingen, speelt een belangrijke rol voor verschillende stakeholders (belanghebbenden) in het ontwikkelproces. Daarnaast is Rik parttime werkzaam bij Getronics PinkRocade, op de afdeling ServiceLine Architectuur van de Sector Public. Zijn werkzaamheden daar bestaan voornamelijk uit het begeleiden van jonge academici bij hun afstudeeronderzoek op het gebied van softwarearchitectuur.

email: rik@cs.vu.nl



drs. Viktor Clerc is consultant bij DNV-CIBIT en in deeltijd verbonden als promovendus aan de Vrije Universiteit Amsterdam.

email: viktor@cs.vu.nl ; viktor.clerc@dnv.com



drs. Remco C. de Boer was betrokken bij verschillende onderzoeksprojecten op het gebied van kennismanagement en *knowledge discovery*. Deze nationale en internationale projecten kenden diverse toepassingsgebieden, waaronder de verticale coatingindustrie, wetenschappelijke literatuur en medische triage. Als onderdeel van zijn R&D-activiteiten was Remco ook verantwoordelijk voor projecttaken op het gebied van Software Engineering. Momenteel is hij verbonden als promovendus aan de Vrije Universiteit te Amsterdam. In zijn huidige onderzoek ziet hij de Software Engineering zélf als toepassingsgebied voor kennismanagement en knowledge discovery. Hij richt zich met name op het ontdekken van relevante softwarearchitectuurkennis in productdocumentatie.

email: remco@cs.vu.nl



drs. Anton Jansen is postdoc onderzoeker aan de Rijksuniversiteit Groningen. Eerder was hij promovendus aan deze zelfde universiteit. Het onderwerp van zijn proefschrift, waarop hij komend jaar op hoopt te promoveren, is het modelleren van architecturale ontwerpbeslissingen. Hij richt zich met name op het ontdekken van relevante relaties tussen deze beslissingen en de interactie met het architectuurontwerp.

email: gradius@fmf.nl

Meer informatie

Geïnteresseerd in het GRIFFIN-project? Voor meer informatie kunt u contact opnemen met de wetenschappelijk verantwoordelijke van het project, prof. dr. Hans van Vliet (hans@cs.vu.nl). Bent u geïnteresseerd in de volledige tekst van een of meer van de bovengenoemde artikelen, neemt u dan contact op met de eerste auteur van het desbetreffende artikel. De emailadressen van de auteurs staan vermeld in de sectie 'over de auteurs'.
