Softwarearcheologie als basis voor strategie

Bestuurlijke grip op software

Als het management grip wil krijgen op de software binnen de organisatie, kan onderzoek door onafhankelijke experts uitkomst bieden. De auteurs nemen de lezer mee in de beslommeringen van de softwarearcheoloog bij het combineren van sporen uit het verleden met de feitelijke kenmerken van het heden om zo greep te krijgen op de toekomst van de software.

Wim Goes en Chris Verhoef

Het in de greep krijgen en houden van software staat de laatste tijd meer en meer in de belangstelling. Alle branches, van administratieve organisaties tot productiebedrijven, allemaal ondervinden ze aan den lijve de problemen met IT. Alleen wordt het nog niet veel beter: na beloften dat het nu uit is met de problemen, haalt men toch weer de krant omdat het niet van een leien dakje gaat. Toch is er een lichtpuntje: over sommige organisaties hoor je nooit iets. Die doen kennelijk iets goed. Wat dat is, is niet zomaar te kenschetsen, maar een belangrijke conditie is dat men de IT-functie in de tang heeft. Elders in dit blad wordt inderdaad ook betoogd dat het bestuur 'in control' van de software moet zijn (zie het artikel van Klusener & Kuipers).

Weerwoord bieden aan de IT-afdeling blijkt in de praktijk steeds weer lastig. Zelfs vooraf gemaakte afspraken bieden geen soelaas om bij oplevering kanttekeningen te plaatsen bij de geleverde prestatie. Contracten opstellen voor maatwerk-IT en contractmanagement voor projecten en het beheren van bestaande IT is een vak apart en vergt diepgaande kennis van software-intensieve systemen voor zowel product als (voortbrengings) proces.

Een mogelijk gevolg is dat men te veel betaalt voor wat men krijgt. Toch zijn er velen die graag iets meer betalen als daarmee de prijs-kwaliteit-verhouding maar op orde zou zijn. Ben je gebonden aan Europees aanbestedingsrecht, dan hoort de overgang van de ene leverancier naar de andere sowieso bij het spel. Maar ook die overgang blijkt vaak lastig. Organisaties in zowel de collectieve als private sector die dit soort problemen veelal voorblijven, hebben met elkaar gemeen dat ze zich op bestuursniveau laten voorlichten door de feiten te kennen. Organisaties die uit het dal komen van de ene na de andere operationele crisis idem dito. Dat voorlichten is niet gemakkelijk. Om vanuit de bestuurskamer zicht te krijgen op de IT-functie op geaggregeerd, feitelijk niveau vergt een andere aanpak. Een mogelijkheid is het in opdracht van de topbestuurders laten uitvoeren van een onderzoek door een gespecialiseerd (onafhankelijk) team van experts om het ontbrekende of verminderde zicht op de IT ten volle te openbaren. In dit artikel geven we aan hoe een dergelijk onderzoek er in de praktijk kan uitzien.

Een aardige aanleiding voor zo'n onderzoek is een majeure operatie die de organisatie te wachten staat. Denk aan van zaak naar klant, van innen naar uitkeren, van batch naar online, van eiland naar keten, van probleem naar oplossing.

Het eerste dat in alle gevallen van belang is, is om de ist-situatie te kennen, zeg maar de boe-

Samenvatting

Om vanuit de bestuurskamer zicht te krijgen op de IT-functie op geaggregeerd, feitelijk niveau kan het management een onderzoek laten uitvoeren door gespecialiseerde (onafhankelijke) experts om het ontbrekende of verminderde zicht op de IT ten volle te openbaren. De zogenoemde softwarearcheoloog is dienstbaar aan de IT-strategie van de organisatie en is daarom van grote waarde om feitelijk te kunnen besturen.

delinventarisatie: wat hebben we zoal in huis, wat is de staat waarin het verkeert en hoe is het zo gekomen? De antwoorden daarop sturen of je, en zo ja, hoe je in de soll-situatie terecht kunt komen. Daartoe is een onderzoek nodig om de toekomstvastheid van de bestaande situatie te leren kennen. Dat is zo gemakkelijk nog niet en al snel dringt zich de vergelijking op met archeologen die met stukjes bot en steen een oude beschaving proberen te reconstrueren. We nemen de lezer mee in de beslommeringen van de softwarearcheoloog bij het combineren van sporen uit het verleden met de feitelijke kenmerken van het heden om zo greep te krijgen op de toekomst van de software. De focus ligt hierbij op het aspect onderhoudbaarheid. De softwarearcheoloog vertaalt feiten op 'bit'-niveau naar visie op 'board'-niveau en is hiermee dienstbaar aan en van grote waarde voor de IT-strategie van de organisatie.

»De softwarearcheoloog vertaalt feiten op 'bit'-niveau naar visie op 'board'-niveau«

Gebiedsverkenning

In de archeologie wordt bij de start een nauwkeurige kaart gemaakt van het gebied dat moet worden onderzocht voordat met de opgravingen wordt begonnen. Tijdens de opgravingen wordt op de kaart aangegeven wanneer waar welke artefacten gevonden zijn. Bij het graven in IT-intensieve systemen speelt dezelfde afbakeningsvraag: wat zijn de grenzen van het te onderzoeken systeemcomplex, al dan niet binnen andere systeemcomplexen?

IT-huishoudingen van organisaties en de applicaties daarin zijn veelal evolutionair gegroeid. De applicaties zijn in sterke mate verweven met de werkprocessen die ze ondersteunen. Het zijn vaak dermate complexe stelsels van systemen dat niet precies bekend is wat er in huis is of hoe systemen met elkaar samenwerken en wat de impact is van het verwijderen of wijzigen van een deel van het complex. In veel gevallen is de functionele documentatie niet volledig of ontbreekt die zelfs, is de software al jaren in gebruik en voortdurend in flux, terwijl ook de technische documentatie onvolledig is of zelfs ontbreekt.

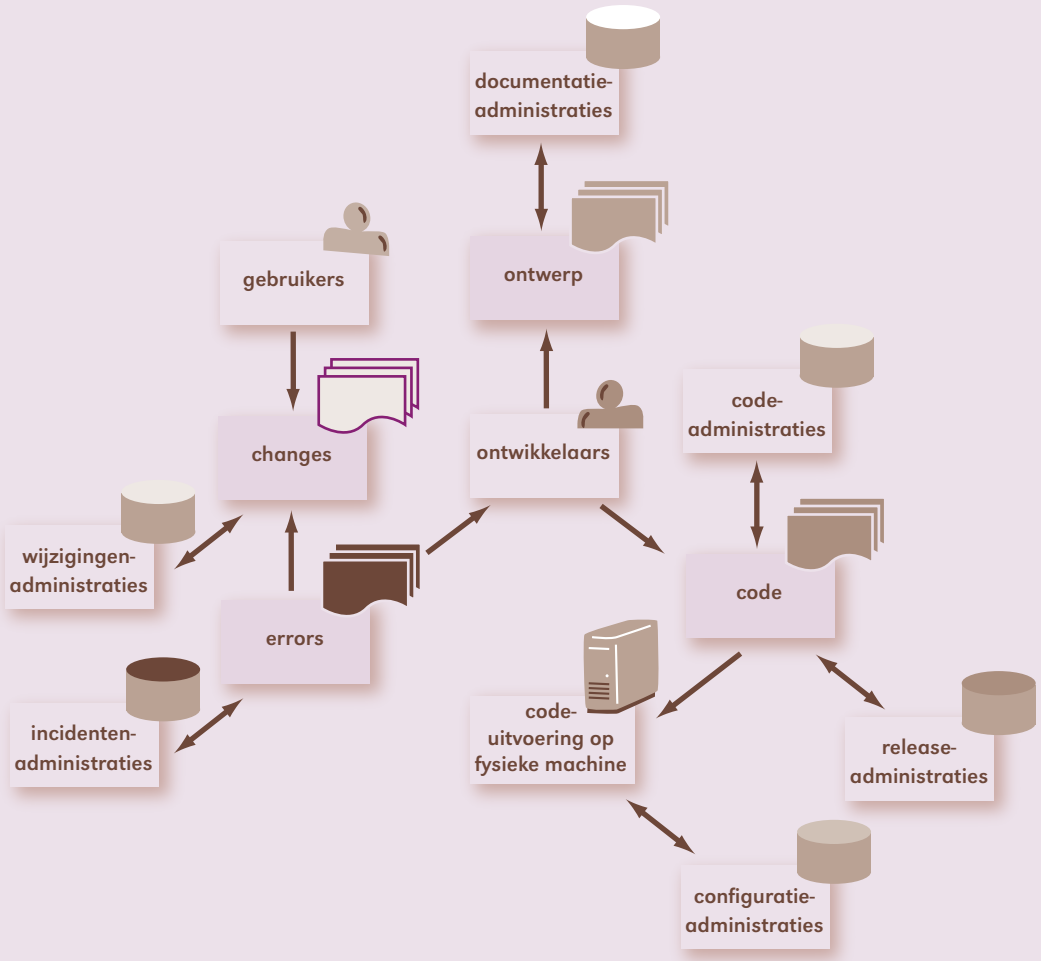
Een manier om invulling te geven aan de gebiedsverkenning is om een zogenoemde overzichtskaart te maken. Deze aanpak, die in de praktijk ontwikkeld is, heeft tot doel om de (werk)processen in samenhang met hun systemen te kunnen afbeelden. Op deze wijze worden de vaak ingewikkelde relaties tussen werkprocessen en hun machinematige ondersteuning transparant gemaakt. Voordat een grote verandering kan plaatsvinden, is het noodzakelijk de huidige situatie volledig te kennen. Wat impliciet is geworden in een lange reeks van jaren, moet expliciet gemaakt worden. Door de bedrijfsprocessen als leidend uitgangspunt te nemen komen alle gebruikte systemen in beeld. Overigens, systemen die niet in beeld komen maar wel op de boedellijst staan, zijn dan direct kandidaat voor uitfasering. Alle relevante systemen zijn via die methode daarna met naam en toenaam bekend. Het gebied is in kaart gebracht. De opgravingen kunnen beginnen.

Opgravingen

Bij het opgraven zoeken we naar zoveel mogelijk artefacten van de binnen het gebied aangetroffen systemen. De archeoloog weet pas of een artefact interessant is tijdens en na de verdere analyse: graaf dus alles op en neem het mee naar het laboratorium. Er is veel dat in aanmerking komt. Voor de hand ligt te starten met het opgraven van de broncode van de systemen binnen het geselecteerde gebied. Dit is vaak minder eenvoudig dan het op het eerste gezicht lijkt.

Wat valt er allemaal onder de broncode? Onder de volledige broncode vallen alle zaken die nodig zijn om de huidige productieomgeving volledig te kunnen reproduceren. Hieronder worden bijvoorbeeld ook verstaan build-, installatie- en databasescripts, inputcode voor generatoren, outputcode door generatoren en de code van de generatoren zelf. De taal (of talen) waarin het systeemcomplex is geschreven, is geen beperkende factor voor een dergelijk onderzoek.
Omdat in het laboratorium alles nog moet worden geanalyseerd, doet een softwarearcheoloog er goed aan de softwarecode in tekstuele (ASCII-)vorm te verzamelen. Er zijn veel hulpmiddelen waarmee broncode wordt gemaakt. Bij de hulpmiddelen waarmee de broncode standaard in tekstueel formaat wordt opgeslagen, is het graaf- en analysewerk relatief het eenvoudigst. Er zijn echter ook hulpmiddelen die een 'eigen' formaat hebben voor het opslaan van de broncode. In de praktijk blijkt dat er altijd wel een manier is om de broncode in tekstuele vorm op te graven, te exporteren. Hiervoor wordt gezocht naar exporteerfuncties in de menu's van het hulpmiddel of worden databaseselecties gebruikt als de code in een database wordt opgeslagen.
Soms is er sprake van code in vermomming. Zo zijn er bijvoorbeeld hulpmiddelen die de broncode ombouwen tot XML-formaat, deze in een zipfile stoppen en de zipfile een andere naam geven. Als er, na een zeer intensief traject, niet verder gegraven kan worden en er is nog steeds geen tekstuele, dus toetsbare, vorm van de broncode gevonden, dan is er sprake van een ernstige situatie. De code kan slechts beperkt worden getoetst en kan bijvoorbeeld niet gekend worden op kwaliteit. Voor veiligheidskritische systemen is dat een onacceptabele situatie.
Na identificatie van de bronbestanden is het karwei nog niet af. Er is een groot aantal artefacten dat ook nog moet worden meegenomen. Ten eerste zijn er de documenten die gemaakt zijn bij het tot stand komen van het systeem, zoals functionele specificaties en ontwerpen. Releasedocumentatie van alle voor productie opgeleverde versies van het systeem is een tweede mogelijke set aan documentatie. Deze releasedocumenten geven een beeld van de toegevoegde functionaliteit in opvolgende releases.
Hiernaast vinden er gedurende de hele levensduur van een systeem, startend vanaf het moment dat het eerste document wordt gemaakt en de eerste code wordt geschreven, verschillende gebeurtenissen plaats die voor een toekomstvastheidsonderzoek van belang zijn. Als eerste zijn er de wijzigingen. Er zal in de praktijk sprake zijn van een continue stroom van aanpassingen vanuit de gebruikersorganisatie. Daarom moet er gezocht worden naar de administratieve vastlegging van deze wijzigingsverzoeken. Alle gevonden administraties moeten mee naar het onderzoekslaboratorium.
Bij de ontwikkeling en exploitatie van software worden fouten gemaakt. Ook hier zijn er, als het goed is, (incidenten)administraties over de hele levensduur van het systeem die dit registreren. Ook deze administraties zijn van groot belang voor onderzoek.

Voor alle artefacten geldt: graaf zo diep mogelijk. Neem naast de recente versies van artefacten zoveel mogelijk versiegeschiedenis mee naar het laboratorium. De administraties voor incidenten en wijzigingen kunnen vele verschillende vormen hebben: van ongestructureerde tekstdocumenten en spreadsheets tot en met gestructureerde versiebeheersystemen en incidentenadministraties. Er wordt ook soms van registratiewijze en hulpmiddel veranderd. Voor een gedegen toekomstvastheidsonderzoek is dat allemaal van belang.
De verschillende artefacten zijn vaker dan eens in beheer bij verschillende afdelingen en/of organisaties. Het team van archeologen gaat voor elk systeem dat binnen het in kaart gebrachte gebied valt, op zoek naar deze beheerders. Wie welke artefacten in beheer heeft is afhankelijk van de wijze waarop het proces georganiseerd is. Een goed vertrekpunt is het hanteren van de driedeling van functioneel beheer, exploitatie en ontwikkeling. Functionele artefacten worden opgevraagd bij functioneel beheer, exploitatie kan zorgen voor overzichten van incidenten en ontwikkeling heeft de broncode in beheer. Het opleveren van al die artefacten is voor organisaties in de praktijk soms lastig. Beheerders die het versiebeheer in de vingers hebben, leveren de broncode snel en compleet op. Het komt ook voor dat er geen versiebeheersysteem in gebruik is of dat het foutief gebruikt wordt of dat de noodzakelijke discipline ontbreekt. In deze fase leert de softwarearcheo-



Figuur 1. **Overzicht van softwareartefacten**

loog van het opleveringsproces en observeert hoe dat verloopt. Het verloop van de aanlevering geeft een goede eerste indruk van de staat van de IT-huishouding.
Naast de hiervoor genoemde artefacten moet een softwarearcheoloog altijd de gemaakte kosten voor de systemen kunnen achterhalen. Bij deze kosten wordt onderscheid gemaakt tussen regulier onderhoud enerzijds en projecten en vernieuwingen anderzijds.

Eerste analyses in het laboratorium

Het uit te voeren onderzoek moet volledig transparant en reproduceerbaar zijn. Het labelen van de ontvangen artefacten met een ontvangstdatum en een identificatie is een eerste stap. Soms worden niet alle artefacten tegelijk en/of compleet aangeleverd en zijn opvolgende opleveringen nodig. Tevens wordt van elke aanlevering een zogenoemde digitale vingerafdruk gemaakt die de artefacten en uiteindelijk de totale aanlevering uniek identificeert. Achteraf is er dan geen enkele twijfel op basis van welke artefacten het onderzoek uitgevoerd is. Een archeoloog doet iets vergelijkbaars: een foto ter plaatse en het labelen van de artefacten.
De eerste analyses hebben tot doel de aanlevering te controleren op compleetheid en analyseerbaarheid. Een vraag die de softwarearcheoloog zich moet stellen is of van de aangeleverde artefacten eenduidig kan worden vastgesteld wat de broncode is. Dit klinkt als een open deur: er is immers een gegevensdrager aangeleverd waarvan aangegeven is dat de broncode erop staat. Maar vaststellen of het ook daadwerkelijk om broncode gaat, is vaak verre van triviaal. Meestal zijn meerdere analyses nodig. Een softwarearcheoloog beschikt over een breed palet aan mogelijke onderzoeken. Een veelgebruikte eerste analyse is de zogenoemde synoniemenanalyse op basis van de digitale vingerafdruk. Hierbij worden alle als broncode aangeleverde bestanden uniek geïdentificeerd. Alleen bestanden met exact dezelfde inhoud hebben dezelfde vingerafdruk. Daarmee worden eventuele duplicaten in

kaart gebracht, ook de duplicaten die verschillende namen hebben. Een aanlevering van broncode die op orde is, kan in beperkte mate duplicatie bevatten, maar deze moet dan volledig verklaarbaar zijn. Als het versiebeheer wordt uitgevoerd door handmatig bestanden naar verschillende locaties te verplaatsen of te hernoemen, zijn er meestal duplicaten zonder een plausibele verklaring. Voor een goede analyse van de werkelijke broncode moeten de duplicaten eruitgefilterd worden.
Een andere uit te voeren analyse is de zogenoemde extensieanalyse. Een extensie wordt ruwweg gedefinieerd als de tekst na de laatste punt van een bestandsnaam. Van alle aangetroffen extensies wordt een frequentieoverzicht gemaakt. Ook de bestanden zonder een extensie worden in kaart gebracht. Als een aanlevering vervuiling bevat, zijn er over het algemeen veel verschillende extensies en meerdere extensies die slechts zelden voorkomen. Voor alle aangetroffen extensies moet de softwarearcheoloog onderzoeken op welke wijze ze bijdragen aan het totaal. Test-, back-up- en debugbestanden zijn bijvoorbeeld vervuiling en horen niet thuis in de broncode van een productiesysteem. Het komt ook voor dat aan bestanden zeer ongebruikelijke extensies gegeven worden zoals '.dit is niet meer nodig'. Het is duidelijk dat dit niet in productiebroncode thuishoort.
Ook de andere artefacten onderzoekt de softwarearcheoloog op compleetheid en bruikbaarheid. Hiervoor worden ruwe tellingen gemaakt van het aantal keren dat datums, auteursnamen, wijzigingscodes en incidentregistratienummers voorkomen. De gevonden aantallen geven een indicatie welke soorten analyses kunnen worden uitgevoerd en waarvoor (vooralsnog) onvoldoende gegevens voorhanden zijn. De vroegste datums in de aangeleverde administraties geven de tijdsspanne weer waarover de analyses kunnen worden uitgevoerd. Ook geven deze datums een indicatie of de administraties voor de volledige levensduur van het systeem zijn aangeleverd. Het komt natuurlijk voor dat in het verleden gegevens verloren zijn gegaan. Zo wil een beheerder wel eens een versieadministratie opruimen en de oudste gegevens verwijderen om de performance van het versiebeheersysteem te verbeteren – net zoals men in oude tijden nu zeer waardevolle zaken gewoon weggooide. Dit opruimen lijkt op het eerste gezicht verstandig, maar een belangrijk deel van de ontwerpkennis van het systeem, de onderbouwing van in het verleden gemaakte keuzes, gaat daardoor verloren. Zo brengen we de verschillende 'tijdperken' van de software in kaart, alleen de tijdschaal is anders dan bij archeologie.
De eerste analyses resulteren meestal in nadere vragen aan de hoeders van de artefacten. Om het volledige onderzoek zo goed mogelijk te stroomlijnen moeten de artefacten in een vroeg stadium zo compleet mogelijk zijn. Als uit de eerste analyses blijkt dat dit niet het geval is, moet de softwarearcheoloog terug het veld in.

Terug het veld in

Er zijn tijdens de eerste analyse verschillende vragen gerezen die alleen door het veld kunnen worden beantwoord. Een aanlevering van alle artefacten in één keer is de norm, maar komt niet altijd voor. Intensieve interactie met de beheerders is dan noodzakelijk. Dit zijn enkele voorbeelden van vragen die een softwarearcheoloog stelt:

- Wat betekenen de extensies 'abc' en 'xyz' en kunnen deze ook tot de broncode gerekend worden?
- Bestanden met de extensie '123' zijn niet tekstueel leesbaar. Kan het ontwikkelhulpmiddel waarmee deze bestanden gemaakt zijn een tekstuele versie exporteren?
- Er lijkt een aantal gegenereerde bestanden in de aanlevering te zitten. Met welke generator zijn deze bestanden gemaakt en waar is de input?
- Als de bestanden met een zelfgemaakte generator gegenereerd zijn, welk deel van de aangeleverde broncode vormt dan die van de generator?
- In de aangeleverde versiehistorie zijn geen auteursnamen en wijzigingscommentaar opgenomen. Worden deze niet bijgehouden of is een nieuwe aanlevering nodig?
- De versiehistorie gaat niet verder terug dan de laatste paar jaar terwijl het systeem ouder is. Waarom is deze oudere versiehistorie niet aangeleverd?

Soms is dan een nieuwe aanlevering van een aantal artefacten nodig. Terug in het laboratorium volgt de softwarearcheoloog weer de eerdergenoemde stappen. De nageleverde artefacten worden ingeboekt en een aantal analyses wordt opnieuw uitgevoerd. Het doel is te bepalen of de tot nu toe verkregen artefacten geschikt zijn voor het uitvoeren van het onderzoek. Als er nog steeds

onduidelijkheden zijn in de aanleveringen, moet de softwarearcheoloog weer terug het veld in. Als in deze fase van het onderzoek meerdere iteraties nodig zijn om de artefacten geschikt te maken voor onderzoek, is dit tevens een belangrijke bevinding. De wijze waarop de spullen beheerd worden, is waarschijnlijk onder de maat. Dat heeft soms gevolgen die zich openbaren bij gebruik: fouten en falen gaan namelijk hand in hand. Het aanleveren van verkeerde versies aan productie is bijvoorbeeld een direct continuïteitsrisico.
De verkregen artefacten zijn de basis voor verder onderzoek: het maken van foto's en de films.

Foto's en films maken

Bij het uitvoeren van het verdere onderzoek maakt de softwarearcheoloog foto's en films. Fotoanalyses geven een statisch beeld van de huidige staat en kwaliteit van het IT-systeem in termen van bijvoorbeeld complexiteit en onderhoudbaarheid. Voorbeelden van onderdelen uit de fotoanalyses zijn:

- Modulecomplexiteit (McCabe): hierbij wordt onder andere een maat bepaald voor het aantal paden van de softwaremodules van een systeem. Hogere waarden van deze maat zijn een indicatie van te complexe software.
- Omvang en distributie van modules, in termen van regels: als er bijvoorbeeld veel grote bestanden zijn, kan dit een risico vormen voor de aanpasbaarheid.
- Aanwezigheid of afwezigheid van commentaren: de mate en kwaliteit van het commentaar in de code zijn indicaties voor de mate van toekomstvastheid.
- Inter-modulecomplexiteit: het analyseren van globale modulerelaties en datamodelcomplexiteit. In het datamodel worden de gegevens opgeslagen en de verschillende modules van het systeem maken gebruik van (delen van) het datamodel. Complexe relaties tussen modules en data zijn indicaties van een inflexibel systeem met een complexe impact bij veranderingen.
- Naamgevingsanalyse: consistentie in structuur en naamgeving van variabelen, constanten, procedures, modules et cetera; gebruik van en zich houden aan conventies en richtlijnen en dat benchmarken aan best practices.

Het doel van filmanalyses is de wordingsgeschiedenis van het IT-systeem te bepalen op basis van een reconstructie van het verleden. De filmanalyse reconstrueert het onderhoudsboekje van een IT-systeem en daarmee wordt onder meer inzichtelijk of op tijd onderhoudsbeurten uitgevoerd zijn. Er wordt gezocht naar sporen van goede of slechtere momenten in het verleden. Het verleden vertelt veel over de huidige staat, bijvoorbeeld over de onderhoudbaarheid en aanpasbaarheid, maar ook over performance en andere aspecten van de ISO 9126-norm voor softwarekwaliteit. Was het vroeger nooit een probleem om aanpassingen te maken, dan is de kans aanzienlijk dat dit in de toekomst ook het geval zal zijn. Waren er bijvoorbeeld altijd al performanceproblemen, dan is de verwachting dat die in de toekomst niet zomaar verdampen.
Voor het maken van de film kunnen bijvoorbeeld de volgende historische analyses worden gemaakt:

- Datumanalyse: wanneer zijn er grote veranderingen in de software doorgevoerd? Hadden deze veranderingen een acceptabele impact en doorlooptijd?
- Ontwikkelaaranalyse: welke ontwikkelaars hebben (wanneer) kennis van welk deel van het systeemcomplex? Zijn er 'gaten' in de systeemkennis of bedreigingen voor de borging hiervan?
- Incidentenanalyse: hoe heeft het systeem zich in het verleden gedragen en hoe verhoudt zich dit tot de doorgevoerde wijzigingen?

Interpretatie

Als de eerste foto- en filmbeelden er zijn, moet de softwarearcheoloog terug het veld in, in gesprek met de ontwikkelaars en beheerders. Er zijn veel manieren om software te ontwikkelen. De ontwikkelaar kan hiervoor een grote verscheidenheid aan methoden, technieken en hulpmiddelen gebruiken. Maatwerktoepassingen verschillen hierdoor onderling sterk. Deze zeer grote verscheidenheid maakt dat er bij het onderzoeken van software enerzijds sprake is van een exploratief proces. Anderzijds kan door deze grote verscheidenheid alleen een goed kwaliteitsoordeel gegeven worden na interactie met beheerders in het veld.
De normen voor bijvoorbeeld complexiteit van programmatuur vergen naast de kwantitatieve ook een kwalitatieve insteek. Er kunnen bijvoorbeeld goede ontwerpredenen (denk aan ingewikkelde wet- en regelgeving) zijn waardoor een module wel hoog op McCabe-complexiteit scoort maar er toch sprake is van een testbare en stabiele module. Dat soort zaken komt in overleg met ontwikkelaars tot stand. Een andere standaard is de zogenoemde J-STD-016 voor systems engineering. Ook voor het toetsen op een dergelijke standaard moeten de overwegingen

van de ontwerpers/ontwikkelaars in ogenschouw genomen worden.
Op basis van de feiten en de verschillende analyses legt het team van experts verbanden. Bijvoorbeeld: zijn er delen van het systeem waar veel wijzigingen in doorgevoerd moeten worden, die tevens heel complex zijn en waar feitelijk maar één ontwikkelaar kennis van zaken heeft? Dan is een hotspot in de programmatuur ontdekt. Dan komt de vraag 'Hoe nu verder?' aan bod en die wordt aan de bestuurders gesteld.

Vertalen naar bestuurlijke visie

Aan het eind van het onderzoek is er een grote hoeveelheid kwalitatieve en kwantitatieve gegevens verzameld. De softwarearcheologen vertalen deze nu vanuit verschillende gezichtspunten naar de impact, waaronder risico's. Voorbeelden van deze gezichtspunten zijn:

- Financieel. Vanuit het financiële gezichtspunt kan worden bepaald of de kosten die voor het systeem worden gemaakt voor onderhoud en vernieuwing, in verhouding staan tot de inhoudelijke aspecten, zoals omvang en complexiteit.
- Organisatorisch. Als de vertaling vanuit het organisatorische gezichtspunt wordt gemaakt, komen aandachtspunten zoals het niet optimaal functioneren van de versiebeheer- en ontwikkelprocessen of beperkte kennis bij ontwikkelaars naar voren.
- Technisch. Vanuit het technische gezichtspunt komt bijvoorbeeld naar voren welke delen van het systeem technisch risicovol zijn voor de toekomstvastheid en welke investeringen nodig zijn om dit te verbeteren.

Op basis van deze bestuurlijke vertaling kunnen op feiten gebaseerde beslissingen genomen worden over de strategische en tactische omgang met het IT-systeem. In plaats van tasten in het duister is dit een grote stap voorwaarts en heeft het management, dankzij de softwarearcheoloog, de software weer in de greep.

Drs. ir. Wim Goes
is bij PricewaterhouseCoopers Advisory adviseur Information Management, waaronder Software Assessment & Control.
E-mail: wim.goes@nl.pwc.com.

Prof. dr. Chris Verhoef
is hoogleraar informatie aan de Vrije Universiteit en wetenschappelijk adviseur voor overheid en bedrijfsleven.
E-mail: x@cs.vu.nl.