

De kloof en de brug

*Bouwers en beheerders zijn net als twee zijden van een muntstuk.
Ze vormen een eenheid maar kijken elkaar nooit aan.
(Oude Chinese wijsheid, 2002 NC)*

Inleiding

Binnen het IT wereldje zijn we dol op zogenaamde "buzzwords", begrippen die plots volop in de aandacht staan en waar iedereen even plotseling een mening over heeft.

Dit geldt ook voor het begrip "applicatiemanagement" (AM). Op zichzelf niets revolutionairs, de term en de bijbehorende problemen bestaan al jaren, maar sinds de Application Services Library (ASL) de kop heeft opgestoken stort iedere zichzelf respecterende ICT organisatie zich op dit vakgebied. Daarbij heeft iedereen in de eerste instantie dezelfde vragen: Wat is ASL, Hoe te gebruiken, Welk probleem lost ASL op, en wat zijn de voordelen? En zeker de ICT-organisaties die met ITIL werken stellen de vragen: Is ASL wel nodig? En hoe sluit dat aan op mijn ITIL processen?

De twee auteurs, een uit de ITIL wereld, de ander uit de AM wereld, hebben voor u de antwoorden.

Waarom doen we eigenlijk aan applicatiemanagement?

Organisaties zijn in hoge mate afhankelijk van de applicaties die ze gebruiken. Ze moeten de juiste functionaliteit bieden, zowel nu als in de toekomst. En omdat organisaties continu in beweging zijn veranderen de behoeften van de bedrijfsprocessen binnen die organisaties ook continu. Dit resulteert dan ook in de behoefte aan wijzigingen in de gebruikte applicatie. Dit komt zeer sterk tot uiting in bijvoorbeeld e-commerce omgevingen, waar een organisatie zo flexibel mogelijk wordt ingericht om maximaal te kunnen inspelen op continu veranderende marktveranderingen.

Naast deze functionele behoeften zijn er trouwens ook technische eisen die een rol kunnen spelen bij aanpassingen van een applicatie. Zo kan het bijvoorbeeld vanuit het oogpunt van performance noodzakelijk zijn veranderingen in een applicatie door te voeren. Maar ook deze veranderingen staan uiteindelijk in het teken van een optimaal beantwoorden aan de behoeften van de bedrijfsprocessen.

De mate waarin applicaties aansluiten op deze behoeften is mede bepalend voor de efficiëntie van de bedrijfsprocessen. Een verminderde efficiëntie kost altijd geld.

Vanuit hun ondersteunende rol ten opzichte van de bedrijfsprocessen proberen ICT organisaties in de regel in te spelen op hun behoefte aan verandering.

Feitelijk staat dit haaks op wat een beheerorganisatie het liefst wil, namelijk een stabiele omgeving. Stabiliteit is namelijk het equivalent van betrouwbaarheid, kwaliteit en optimaal beheersgemak. Hier komt dan ook de kreet vandaan dat de ideale beheersomgeving een omgeving zonder eindgebruikers is. Dat hiermee direct het bestaansrecht van de beheersomgeving komt te vervallen moge duidelijk zijn...

ITIL en change management

Om toch de gewenste flexibiliteit te kunnen koppelen aan de noodzakelijke stabiliteit hebben veel ICT beheerorganisaties gekozen voor het toepassen van ITIL processen. Met deze verzameling van best practices proberen ze hun infrastructuur, zowel de "harde" infrastructuur (de "dozen" en de "touwtjes") als hun applicaties, in dit spanningsveld van dynamiek en stabiliteit overeind te houden. Dat dit in de regel geen sinecure is, blijkt wel uit het feit dat 75% van alle verstoringen die in een ICT infrastructuur optreden, worden veroorzaakt door wijzigingen die zijn doorgevoerd.

Dat is dan ook de reden dat binnen ITIL het Change Management proces is opgenomen. Een proces dat tot doel heeft veranderingen door te voeren met minimale of aanvaardbare risico's. Veranderingen worden pas doorgevoerd nadat ze van alle kanten zijn belicht en de maatregelen zijn genomen.

Nu leer je in iedere ITIL cursus dat een van de grootste risico's (zo niet het grootste risico) bij het Change Management proces het passeren van dit proces is: het doorvoeren van wijzigingen op de infrastructuur zonder Change Management daarin te kennen.

Ontwikkeltraject

Met dit gegeven in het achterhoofd en met de ITIL bril op, gaan we eens naar een standaard ontwikkeltraject van een applicatie kijken.

Vanuit het bedrijf wordt een behoefte aan functionaliteit geuit. Nadat impactanalyses zijn uitgevoerd en alternatieven zijn afgewogen, wordt er een (ontwikkel) project gestart dat via een meestal gefaseerde aanpak van ontwerpen, bouwen en testen tot een eindresultaat komt. Uiteindelijk vindt, na acceptatie door de opdrachtgever, een overdracht aan beheerorganisatie plaats. Het is natuurlijk goed gebruik om een beheerorganisatie een audit te laten doen op het nieuw ontwikkelde applicatie om te onderzoeken of dit systeem daadwerkelijk in beheer kan worden genomen, maar in veel situaties kun je je afvragen of deze beheerorganisatie wel met goed fatsoen kan weigeren, gezien de veelal aanzienlijke belangen die op het spel staan. Je ziet dan ook vaak dat een projectorganisatie na acceptatie en oplevering nog enige tijd (in afgeslankte vorm) in stand gehouden wordt voor het leveren van support aan de beheerorganisatie.

In de hierboven geschetste werkwijze zit een cruciale tekortkoming.

De ICT beheerorganisatie, die primair verantwoordelijk is voor het beschikbaar houden van de applicatie, wordt pas aan het eind van het ontwikkeltraject voor een voldongen feit geplaatst. Het is het schoolvoorbeeld van het doorvoeren van een wijziging zonder Change Management hierin te betrekken.

Nu is er wel degelijk een tendens die ertoe neigt beheerders te betrekken bij het ontwikkelingstraject van applicatie. In de praktijk blijkt deze aanpak slechts marginaal resultaat op te leveren. Beheerders gunnen zich meestal geen tijd om optimaal in dergelijke projecten te participeren, daarnaast wordt hun bijdrage menigmaal ondergewaardeerd. De kloof tussen ontwikkeling, AM en beheer is tenslotte behoorlijk diep...

Wat zouden we kunnen doen om dit dilemma te beëindigen?

Een alternatief is in het vervolg alle aanvragen voor nieuwe of gewijzigde functionaliteit via een wijzigingsverzoek (RFC) in te dienen bij Change Management. Hierbij ontstaat dan de

opmerkelijke situatie dat een verzoek tot het ontwikkelen van een applicatie wordt gericht aan de partij die het zal gaan beheren. Toch is deze situatie minder vreemd dan hij lijkt, wanneer je je realiseert dat de kosten van ontwikkeling slechts een relatief bescheiden deel uitmaken van de totale kosten die met een applicatie gemoeid zijn (20% ontwikkelingskosten versus 80% beheerkosten).

Vanuit Change Management zal dit wijzigingsverzoek ter beoordeling worden voorgelegd aan alle belanghebbenden (de stakeholders). Zij beoordelen dit verzoek vanuit hun expertise en zorgen ervoor dat de uiteindelijk door te voeren wijziging, in dit geval dus de nieuwbouw of aanpassing van een applicatie, zal worden doorgevoerd waarbij ieders belang de revue is gepasseerd. Als een van de belanghebbenden een probleem heeft met het ingediende voorstel, dan zal een afweging worden gemaakt hoe hier mee om moet worden gegaan. Een consequentie van de invoering van een nieuw applicatie zou bijvoorbeeld kunnen zijn dat er een capaciteitsuitbreiding op de service desk zal moeten plaatsvinden om de verwachte toestroom van gebruikersvragen het hoofd te kunnen bieden. Hier kunnen dan maatregelen voor genomen worden.

Uiteindelijk zal, onder de verantwoordelijkheid van Change Management, een project worden opgestart waarin de ontwikkeling van de applicatie zal plaatsvinden.

Dit lijkt in eerste instantie een aardige verbetering. Vanaf het eerste moment is de beheerorganisatie actief betrokken bij een ontwikkelingstraject en kan daar vanuit alle beheerdisciplines op reageren.

Toch is dit geen volledige oplossing. Het probleem zit hem er namelijk in, dat de best practices van Change Management ophouden bij het opstarten van het ontwikkelingsproject. Het enige dat er verder binnen Change Management over geroepen wordt is dat je na afloop een post implementation review moet doen om te kijken wat je kunt leren van het afgelopen traject zodat je het in de toekomst beter kunt doen.

Dit is ook niet zo verwonderlijk als je je realiseert dat ITIL zich eigenlijk op hoofdlijnen bezig houdt met de hardware kant van de infrastructuur. Wanneer er gesproken wordt over software, gaat het in de meeste gevallen over kant en klare applicatiesoftware, die als een pakket wordt aangeschaft en gedistribueerd, zoals bijvoorbeeld een MS Office pakket. Zeker wanneer het gaat over moderne, op componenten gebaseerde applicatieontwikkeling, zie je dat ITIL tekort schiet. Daar doet zelfs het ITIL proces Release Management, dat zorgt voor het beheer van de software versies en de distributie daarvan, geen afbreuk aan.

Maar hoe dan wel verder?

Een antwoord hierop lijkt te liggen in het toepassen van ASL. Deze methodiek kan prima beschouwd worden als een verlengstuk van / uitbreiding op ITIL waarbij bovenstaande problematiek ondervangen wordt.

Hierbij moet aangetekend worden dat ASL momenteel gepositioneerd wordt als een op zichzelf staande oplossing voor applicatie management, alhoewel bij de samenstelling van ASL wel naar ITIL is gekeken. Beide aanpakken, zowel ASL als ITIL, hebben echter dezelfde doelstelling, en wel ervoor te zorgen dat de bedrijfsprocessen die van hun diensten gebruik maken zo goed mogelijk worden ondersteund.

Wat is ASL?

Hoe en waar te beginnen met professioneel AM (application management)? Hoe niet te verzanden in niet toe gepaste processen, ambtelijke procedures, ongelezen werkinstructies, onbegrepen samenhang, goede intenties, onduidelijkheid, demotivatie,

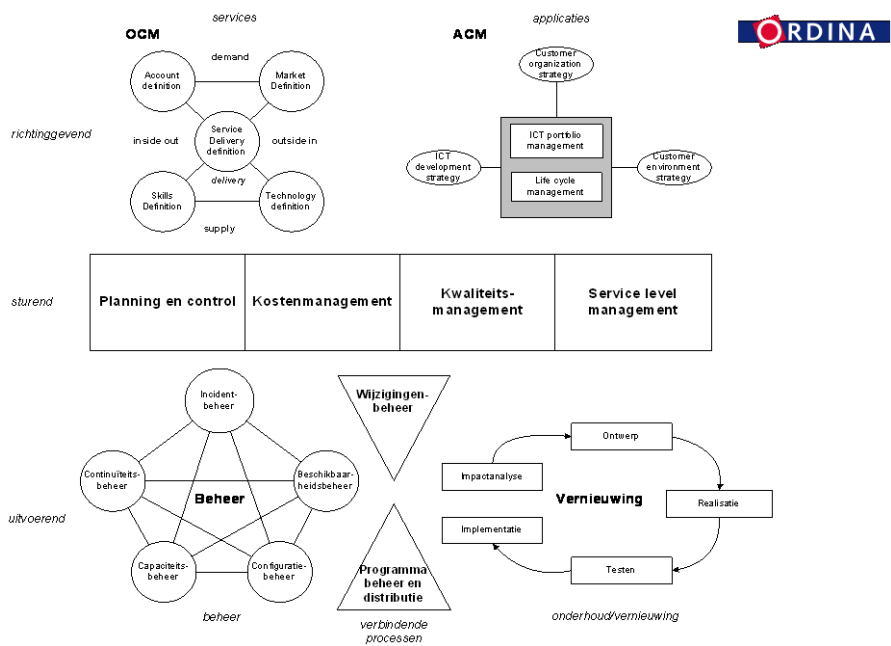
klantvriendelijkheid, het voortschrijdende inzicht, de chaos, deceptie en het financieel onbestuurbare?

Het professioneel - dus het nakomen van afspraken en betrouwbaar voorspellen - onderhoud en beheer van applicaties is onmisbaar om deze essentiële schakel tussen ICT en Business blijvend goed te laten functioneren. Nieuwe inzichten en het standaard framework, ASL, maken dit wel degelijk mogelijk.

Het Doel

ASL heeft ten doel om bedrijfsprocessen optimaal te ondersteunen met gedurende de gehele levenscyclus van deze bedrijfsprocessen.

Het Framework



In het ASL framework worden de volgende procesclusters onderscheiden:

de **beheerprocessen**, die er voor zorgen dat de applicaties dagelijks datgene doen wat ze moeten doen (Incident, Configuration, Availability, Capacity en Continuity Management);

de **onderhoud en vernieuwing processen**, waar de applicaties worden aangepast naar aanleiding van verstoringen en op gewenste nieuwe functionaliteit (Impact Analyses, Design, Realization, Testing, Implementation);

de **verbindende processen**, die onder andere de overdracht van dagelijks beheer naar onderhoud en vice versa regelen (Change Management, Software Control & Distribution);

de **(tactische) managementprocessen** (Planning & Control, Cost Management, Quality Management, Service Level Management);

de **richtinggevende (strategische) procesclusters**, waarin enerzijds de strategie ten aanzien van de ondersteuning van de bedrijfsprocessen door ICT wordt bepaald (Applications Cycle Management) en anderzijds de toekomstvisie van de ICT serviceorganisatie zelf (Organization Cycle Management).

De Foundation

Het initiatief om tot te komen tot een internationale standaard op het gebied van applicatiebeheer heeft geleid tot de oprichting van de stichting ASL Foundation. In deze stichting participeren de gerenommeerde ICT-leveranciers met AM expertise. De ASL processen binnen ieder van de procesclusters zijn beschreven op basis van een verzameling van bijbehorende Best Practices. Deze ICT-leveranciers verzamelen en ontwikkelen ondermeer de Best Practices.

ASL beoogt uit te groeien naar een standaard van ITIL niveau.

Zie verder: www.aslfoundation.org

Waarom zou ASL wel werken?

Om deze vraag te kunnen beantwoorden zullen we eerst de problemen moeten identificeren die in organisaties in meer of mindere mate op het gebied van AM spelen.

Welke narigheid kan met ASL worden opgelost en voorkomen?

Veranderingen doorvoeren zonder tijdig Change Management in te schakelen leidt tot de volgende verstoringen:

- falend versie- en releasebeheer, ondermeer herkenbaar aan het opduiken van eerder opgeloste fouten;
- klagende gebruikers;
- de kosten van AM zijn per definitie altijd te hoog;
- verantwoordelijkheden in het samenspel van functioneel-, technisch- en applicatiebeheer zijn onduidelijk;
- onvolledige kennis borging, wat ondermeer leidt naar afhankelijkheid van bepaalde ICT-medewerkers;
- de veelheid van de systemen en (nieuwe) technologieën maakt het probleem van de kennis borging nog groter;
- releases zijn niet goed planbaar, zowel in tijd als in kosten;
- gedemotiveerde ICT-medewerkers, de ambitie om vooral aan **projecten** bij te dragen is veelal groter dan je te moeten inspannen voor "suffige" beheer;
- wijzigingen doorvoeren gaat moeizaam, hiermee komt de time-to-market in het geding;
- ontbreken van SLA's (Service Level Agreements) of te algemeen opgestelde SLA's.

Stop: we hebben toch ook CMM. CMMI, ISO en TQM?

Zeker, en er zijn meer methodieken die het AM ondersteunen. Het uitgangspunt is dat ASL het beste aansluit op organisaties die bekend zijn met ITIL. ASL is tenslotte geboren in de ITIL wereld.

Als een AM-organisatie het versie- en releasebeheer niet onder controle heeft, betekent dat dus dat ze hun primaire bedrijfsproces niet beheersen. De roep om professionalisering is hiermee urgent geworden. En welk bedrijf wil nu geen reductie op de ICT-kosten, verbeterde

dienstverlening, betrouwbare voorspelbaarheid en flexibiliteit met de garantie op hoge kwaliteit?

Hoe draagt ASL bij aan professionalisering van AM?

Zoals bekend is een set van gedegen processen en procedures geen enkele garantie voor succes. Dit geldt natuurlijk ook voor de ASL processen. ASL onderkent wel de volgende elementen die ten grondslag liggen aan het kunnen leveren van de AM diensten volgens afspraak:

- de complexiteit van AM is onderkend. Dit is onder meer duidelijk terug te vinden in de samenhang van 15! processen die nodig zijn voor het herstellen van één bug in een applicatie;
- het koppelt Beheer aan Onderhoud, Beheer aan Management, Service aan Applicaties, Strategie aan Management;
- het plaatst AM binnen het veel grotere geheel van infrastructuur, bedrijfsprocessen en strategie, zowel binnen de eigen organisatie en de ICT-wereld.

Een belangrijke randvoorwaarde voor het succesvol implementeren van ASL is dat het bedrijfsprocesgericht denken in de afgelopen jaren is toegenomen. Dit is ondermeer terug vinden business alignment. De controle over de infrastructurele zaken, bijvoorbeeld middels ITIL, is de afgelopen jaren verbeterd. Kortom organisaties zijn er nu aan toe om AM te verbeteren.

Wat draagt ASL bij of waartoe leidt de weg?

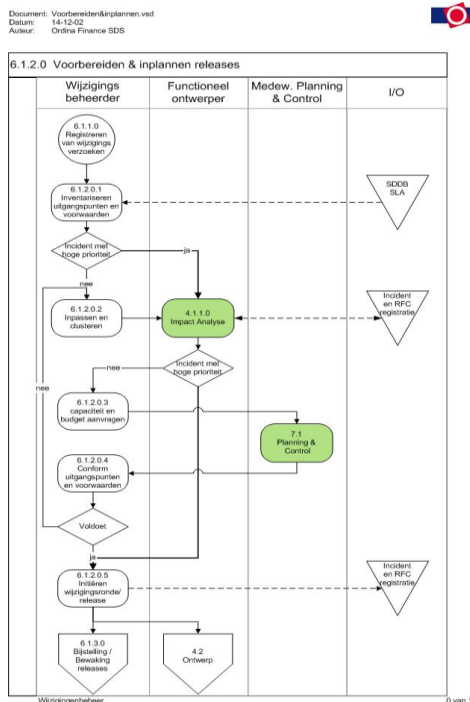
Met andere woorden: wat zijn de merkbare en meetbare voordelen van goed geïmplementeerde ASL processen?

- Het primaire bedrijfsproces van een AM-organisatie is geheel onder controle en dus bestuurbaar;
- heldere en meetbare afspraken door middel van een op maat gesneden SLA: duidelijke verwachtingen en adequate besturing worden mogelijk;
- standaard ICT-services, deze leiden tot reductie van de kosten en een hogere service graad en kwaliteit;
- een geborgde continuïteit, beschikbaarheid en kwaliteit;
- kennis borging, ook van oude of verouderde applicaties;
- het snel kunnen doorvoeren van functionele wijzigingen (time to market);
- er ontstaat ruimte binnen AM-organisatie om nieuwe technologieën eigen te maken en toe te passen;
- je kunt korte en langere termijn strategieën voor applicaties opstellen;
- Er ontstaat een geïntegreerde afstemming op de bedrijfsprocessen;
- door het spreken van dezelfde AM taal, de duidelijk rol verdeling en heldere verantwoordelijkheden is het mogelijk om integraal samenwerken met de verschillende (externe) organisatie en organisatieonderdelen;
- van kostenbeheersing naar kostenbezuiniging en/of verhoogde kwaliteit.

ASL praktijkervaringen

Enkele praktische ervaringen van het introduceren van ASL en het toepassen in de praktijk zijn:

- de discussie of ASL als nieuwe standaard naast een goed functioneerde ITIL organisatie ingezet moet gaan worden is zinloos: in dit soort gevallen moet je niet meer over ASL praten, maar over het uitbreiden van ITIL met specifieke AM processen (deze processen zijn natuurlijk wel op ASL gebaseerd maar dat is niet relevant voor de discussie);
- bij het uitwerken van de processen ontstaat de neiging om verschillende delen van samenhangende processen toch onder te brengen onder één proces;
- ASL is een framework, dus hands on, het voordeel is echter dat hergebruik van procedures goed mogelijk is;
- om niet ASL deskundigen toch met ASL te kunnen laten werken, zonder een ASL foundation te kunnen volgen, zijn ASL processen, procedures en activiteiten uitgewerkt door middel van swim lanes diagrams. Hierbij zijn activiteiten onderverdeeld naar rollen;



Swim lane

- zoals eerder aangegeven combineren technisch-, functioneel- en applicatiebeheer met elkaar. Deze vormen van beheer worden door verschillende organisaties en/of organisatieonderdelen uitgevoerd. Hoe moet je nu omgaan met die organisatieonderdelen die geen ASL procedures hanteren? Hiervoor bestaat een procesinterface, gebaseerd op de swim lanes, ontworpen. In deze procesinterface zijn de niet ASL processen en de ASL processen gecombineerd;

- door de swim lines en procesinterfaces als web pagina's te publiceren kunnen alle betrokkenen hun deel in het proces **zien** en begrijpen;
- de SLA is een zeer bepalend document. In feite wordt de AM-organisatie ten behoeve van het uitvoeren van de SLA ingericht. Het is van groot belang dat de SLA zorgvuldig wordt opgesteld zodat de gewenste service goed wordt afgestemd op de bedrijfsprocessen;
- de zelfstandige functionerende serviceteams (per SLA of applicatie) zijn de warm kloppende harten van AM. De verantwoordelijkheid per serviceteam kan oplopen over de proceslagen Operationeel, Management tot Strategisch. De medewerkers kunnen in deze situatie afhankelijk van hun **eigen** motivatie, ambities, vaardigheden en kennis doorgroeien;
- Stel een bonusstructuur per serviceteam in. Dit verstrekt de teamspirit en geeft een directe aansluiting met de SLA;
- Een ASL implementatie vergt visie en bezieling!

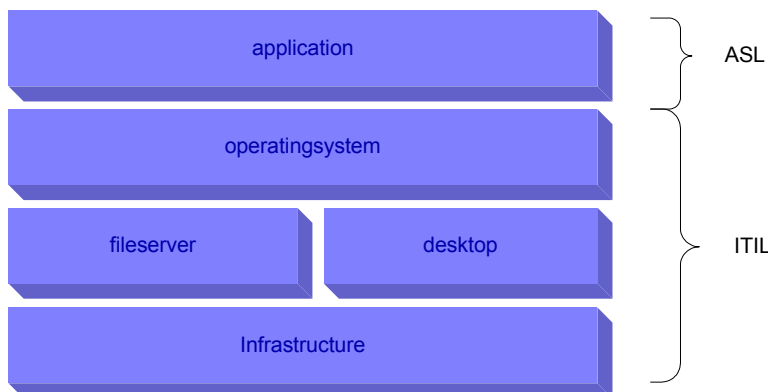
De toekomst en de ontwikkeling van ASL

De ambitie van de ASL Foundation om deze methodiek te laten uitgroeien tot een internationale standaard, vergelijkbaar met ITIL, lijkt haalbaar. Om dit te realiseren zullen de Best Practices in 2003 uitgewerkt en gepubliceerd moeten zijn. De processen die leiden tot verbetering van deze processen zullen dan ook geïmplementeerd zijn. Tooling ter ondersteuning van alle ASL processen zal ontwikkeld moeten worden.

In 2004 zal de samenhang van ASL met ITIL, CMM, CMMI, ISO en andere gangbare methodieken inzichtelijk moeten zijn waarbij het mogelijk is ASL als een onderdeel te integreren van bijvoorbeeld CMMI. De algemene samenhang tussen technisch en functioneel beheer en projecten zal verdiept en zover mogelijk gestandaardiseerd moeten worden. De volgende fase is om de strategische processen meer te laten aansluiten met de Business aligment. Het is niet ondenkbaar dat projecten eerst zullen starten met het inrichten van ASL om daarna vanuit het procescluster Management te worden aangestuurd.

ASL en ITIL

De plaats van ASL ten opzichte van ITIL is als volgt:



Vooraf de processen in de "Beheerbol" van ASL komen overeen met die van ITIL. Er zijn hierbij wel enkele inhoudelijk significante verschillen. Bij Incident management ligt bij ITIL de nadruk op het lokaliseren van het de oorzaak van het incident (is het de hardware, software of een combinatie?) bij ASL ligt de nadruk op het verzamelen van informatie zodat het Incident kan worden herhaald. Diepere problemen (bijvoorbeeld herhaling van een Incident) is bij ITIL ondergebracht in Problem management, bij ASL onder Kwaliteit Management. Configuration management binnen ITIL is registratie en onderling relateren van de infrastructuur. Bij ASL is het in opzet hetzelfde: registratie objecten, versie, status en heel belangrijk de samenhang (b.v. de gemeenschappelijke gebruikte DLL's). Door het ontbreken van deze CMDB kan bij sommige AM-organisaties de implementatie van een nieuwe applicatie, ook als het een standaard, maanden duren! Het ontbreken van een CMDB is een faalfactor.

De eerste versie van ITIL bestond uit 62 handboeken, nu zijn het er twee. ASL start met één boek. Change management speelt zowel bij ITIL als ASL een belangrijke rol voor het voorkomen van instabiliteit. Change management is de brug tussen de ITIL en ASL wereld.