

Application	Applicatie
Life	Levens
Cycle	Cyclus
Management	Beheer

*ALCM in de praktijk versus legacy oftewel
hoe beheers je de kwaliteit van een systeem*

dr. Clemens H.J. Willemsen

ISBN: 978-94-0382-565-6

Gedrukt en gepubliceerd door: Bookmundo

Copyright: Clemens Willemsen 2025

Alle rechten voorbehouden. Niets uit deze uitgave mag worden verveelvoudigd, opgeslagen in een geautomatiseerd gegevensbestand of openbaar gemaakt, in enige vorm of op enige wijze, hetzij elektronisch, mechanisch, door fotokopieën, opname, of op enige andere manier, zonder voorafgaande schriftelijke toestemming van de auteur.

Inhoudsopgave

1	VOORWOORD	6
2	INLEIDING	7
3	STANDAARDEN EN KADERS	8
3.1	Levenscyclus - ISO	8
3.2	Kwaliteit - ISO	14
3.3	Levenscyclus en informatiebeveiliging - NIST	24
3.4	Application Lifecycle Management - ITIL	29
3.5	Handreiking verantwoord vernieuwen en TIME - BZK	30
3.6	Beoordelingskader IT Lifecycle Management- ARK	37
3.7	9 vlaks model - ArchiXL	40
3.8	Beleid, procedures en richtsnoer - EPA	42
3.9	Asset Lifecycle Management - Infraon	45
4	LEGACY	47
4.1	Hoe herken je legacy?	47
4.2	Hoe pak je legacy aan?	51
5	EEN NIEUW RAAMWERK	52
5.1	Uitgangspunten	52
5.2	Stap 1. Aspecten	54
5.3	Stap 2. Karakteristieken	59
5.4	Stap 3. Sub karakteristieken	59
5.5	Stap 4. Bereikte niveaus	60
5.6	Stap 5. Bedrijfs- en technische waarde	62
5.7	Stap 6. Score in het TIME kwadrant en levensloop	64
5.8	Stap 7. Tolereer, Investeer, Migreer of Elimineer	67
6	AFSLUITING	68
	BIJLAGE 1. LITERATUURLIJST	69
	BIJLAGE 2. HET SPREADSHEET	72

Lijst van figuren

Fig. 1. Processen ISO 15288 versus 12207	10
Fig. 2. ISO 24748 versus 15288 en 12207	12
Fig. 3. Stadia versus ontwikkelen/gebruiken	13
Fig. 4. ISO 25xxx serie	14
Fig. 5. ISO 25012 inherente/systeemafhankelijke gegevenskwaliteit ...	19
Fig. 6. ISO 25012 gegevens versus systeem	19
Fig. 7. ISO 25023 soorten kwaliteit en meting	22
Fig. 8. NIST levenscyclus en informatiebeveiliging	25
Fig. 9. NIST afstoting en informatiebeveiliging	26
Fig. 10. Bedrijfs versus technische waarde.....	35
Fig. 11. De vier opties van TIME.....	37
Fig. 12. Variant op 9 vlaks model	40
Fig. 13. Environmental Protection Agency	43
Fig. 14. Evolutie SAP software	48
Fig. 15. Omslagpunt gebruik/vervanging	50
Fig. 16. Het raamwerk	54
Fig. 17. Monitoring aspecten	55
Fig. 18. ISO 37301 Compliance Management Systeem	57
Fig. 19. Monitoring kosten in de loop der tijd	58
Fig. 20. Gelijkmatische kosten in de loop der tijd.....	58
Fig. 21. Monitoring aspecten in de loop der tijd	63
Fig. 22. Bedrijfs versus technische waarde met score	65
Fig. 23. Omslagpunt gebruik/vervanging en TIME	66
Fig. B2. Het spreadsheet	72

Lijst van tabellen

Tab. 1. ISO 25010 karakteristieken	18
Tab. 2. ISO 25010 meeteenheid	21
Tab. 3. ISO 25040 stappen evaluatie	23
Tab. 4. NIST-standaarden	24
Tab. 5. NIST informatiebeveiliging afstoting	27
Tab. 6. Fasen software ontwikkeling	30
Tab. 7. Bedrijfswaarde versus technische waarde	32
Tab. 8. Bedrijfswaarde	33
Tab. 9. Technische waarde	35
Tab. 10. Beoordelingskader stappen	38
Tab. 11. Beoordelingskader maatregelen	39
Tab. 12. EPA fasen	44
Tab. 13. EPA werkdocumenten	45
Tab. 14. Eisen aan hardware vanuit software	48
Tab. 15. Mogelijkheden voor investeren of migratie	51
Tab. 16. Herkomst raamwerk	53
Tab. 17. Subkarakteristieken	60
Tab. 18. Betekenis waarden aspecten	61
Tab. 19. Wijze en frequentie van analyse	61
Tab. 20. Methoden/bronnen van onderzoek	62
Tab. 21. Grenswaarden per aspect	66

1 Voorwoord

De boeken die ik de afgelopen jaren heb geschreven, zijn meestal geïnspireerd door hetgeen ik in mijn werk heb ervaren en waar ik uitwerking aan heb gegeven. Zo iets heeft dan vaak de vorm van een beleid dat centraal wordt vastgesteld en vervolgens in de organisatie wordt belegd. De aanleiding kan ook van buiten komen van een controlerende instantie zoals de Algemene Rekenkamer. Dat was het geval bij de Rijksoverheid inzake Applicatie Life Cycle Management (ALCM) en ook de volwassenheid van de informatiebeveiliging. Bij een ander onderwerp waar ik nauw mee van doen had, risicomanagement, zie je juist dat dit vanuit de organisatie zelf komt. In het eerste geval zie je dat de organisatie het oppakt maar het ook kan worden tot een papieren tijger omdat men niet altijd zelf de urgentie van het probleem en de aanpak onderkent. Na enige tijd gaat de aandacht verloren en gaat men over tot de orde van de dag. Ik mis dan nog wel eens de reeks van bevinding, conclusie en aanbeveling zoals dat wel in de wetenschap gebruikelijk is. Men blijft dan ergens in het midden hangen en komt niet tot actie. In het tweede geval wordt de aanpak omarmd en kan het succesvol zijn. In het geval van ALCM wil ik de organisatie helpen door te kijken naar de noodzaak van ALCM om zo legacy te herkennen en nog liever te voorkomen. Dit kan alleen door continu kritisch te kijken naar de status van de applicaties, hierop metingen te verrichten en zo het management te adviseren om te komen tot een goed onderbouwd besluit; Tolereer, Investeer, Migreer of Elimineer de applicatie (TIME). Degene die direct aan de slag wil gaan, kan het derde deel over het raamwerk lezen en toepassen. Het eerste en het tweede deel geeft aan hoe het raamwerk tot stand is gekomen. Ik wens de lezer veel succes in de toepassing van het raamwerk in het bijzonder door het beschikbare spreadsheet naar eigen behoefte aan te passen.

2 Inleiding

Als je online zoekt naar de term 'Application Life Cycle Management' (ALCM)¹ vind je veel Engelstalige boeken doch nagenoeg geen Nederlandstalige boeken. Het voorliggende boek vult deze leemte op ook als je zoekt op 'Applicatie Levens Cyclus Beheer' wat een minder gebruikte benaming is. Naast 'Application' wordt soms ook gesproken over 'System' dat ik ergens juist vind want het gaat niet alleen om het zorgen voor een goed werkende applicatie doch om de samenwerking met de benodigde hardware. Soms wordt 'Application' of 'System' als LCM geheel weggelaten maar dan hebben we met geheel andere domeinen te maken.

De benaming levenscyclus is op zich bijzonder want er is geen cyclus oftewel niets herhalends alleen een 'leven' omdat een systeem of applicatie begint en eindigt. Meestal refereert men dan aan de Plan-Do-Check-Act-cyclus. De cyclus ligt ook mogelijk in het feit dat een applicatie wordt opgevolgd door een andere applicatie. Levensloop is mijns inziens een betere naam. Kortom; er valt veel aan- en op te merken over de benaming ALCM maar het is wel de meest gebruikte term dus benoem ik dit ook zo in dit boek.

Wat houdt ALCM in? Het idee is dat je de applicatie of het systeem goed monitort op een aantal aspecten waardoor je tijdig weet wat een goede werking in de weg staat of kan gaan staan. Je wilt bijvoorbeeld voorkomen dat een systeem niet overweg kan met het jaar '2000' of de komst van de euro als valuta; in het deel over legacy ga ik daar op in. Regeren is vooruit zien en je wilt obstakels op een rijtje hebben staan voordat die zich (kunnen) gaan voordoen. Het begrip toekomstvast is zeker ook van toepassing en als dat niet het geval is, spreken we over legacy. Dit boekwerk legt ALCM en legacy in begrijpelijke termen uit en geeft door verwijzingen aan waar meer informatie is te vinden.

Ik verwerk de informatie uit de diverse bronnen aangevuld met mijn eigen bevindingen en ideeën in een nieuw raamwerk waarmee, mits goed gebruikt, het moment van vervanging van een applicatie beter valt te voorspellen. Dat raamwerk is ook uitgewerkt in een spreadsheet waarmee het kan worden toegespitst op de eigen organisatie.

¹ Niet te verwarren met '*air-launched cruise missile*'

3 Standaarden en kaders

Er zijn twee domeinen die enerzijds richting geven aan de levenscyclus van een systeem of applicatie en anderzijds invulling geven aan de kwaliteit van een systeem of applicatie; dat laatste vormt de basis van een voorspelbare levenscyclus. Deze beide manieren van aanpak komen hier aan bod waarbij ik de beperkingen aangeef. Dit vormt de inspiratie voor het raamwerk.

3.1 *Levenscyclus - ISO*

ISO 12207 en 15288

Er zijn twee naar elkaar verwijzende ISO standaarden op het gebied van levenscyclus van systemen. [ISO 15288] uit 2023 genaamd 'systeem levenscyclus processen' en [ISO 12207] uit 2017 genaamd 'software levenscyclus processen'. Op het eerste gezicht is dit een beetje verwarrend omdat er geen overkoepelend document bestaat.

De inleiding van 15288 geeft aan dat deze standaard een raamwerk geeft voor de beschrijving van de levenscyclus van een systeem op basis van een '*systems engineering*' benadering. Het betreft systemen met een of meer van de volgende elementen zoals hardware, software, gegevens, mensen, processen, diensten, procedures, bedrijven, materialen en natuurlijke entiteiten; een breed scala dus. De nadruk ligt op het bepalen van de behoeften van de stakeholder, de benodigde functionaliteit, de vereisten, het systeemontwerp en -validatie om een probleem op te lossen. Ook biedt het een raamwerk om de levenscyclus van een systeem in te richten en te verbeteren. Als het belangrijkste systeemelement uit software bestaat dan is het beter om ISO 12207 te gebruiken, stelt 15288. Beide standaarden gaan uit van hetzelfde procesmodel, delen veel activiteiten en taken en verschillen voornamelijk in de beschrijving. Er is dus de nodige overlap.

Als je de inleiding van 12207 leest, begint dit met het noemen van de complexiteit van software dat tot kansen maar ook tot uitdagingen leidt om systemen te ontwerpen en te gebruiken. Deze standaard richt zich evenals 15288 op de levenscyclus van systemen en op alle architectuur niveaus met een raamwerk voor de levenscyclus gebaseerd op een '*software engineering*' benadering. In de overview wordt aangegeven dat de keuze voor 12207 of 15288 afhangt van het '*system of interest*'.