

Een raamwerk voor het ondersteunen van beslissingen met betrekking tot Agile-initiatieven.

Sander Bollen^{1*}

¹Faculteit bedrijfseconomische wetenschappen, Universiteit Hasselt,
Martelarenlaan 42, 3500 Hasselt

*Correspondentie auteur: Sander Bollen, E-mail: sander.bollen@student.uhasselt.be

Ondanks de status van Agile Software Development (ASD) als industriestandaard, blijft de definitie ervan in zowel academische literatuur als de professionele praktijk gefragmenteerd en ambigu. Deze conceptuele onduidelijkheid belemmert organisaties in het effectief selecteren en implementeren van ASD-praktijken die hun specifieke doelstellingen ondersteunen. Dit onderzoek introduceert het *Collaboration-Flow-Learning* (CFL) raamwerk, een theoretisch en praktisch model dat ASD-concepten deconstrueert in drie orthogonale dimensies: *Collaboration*, *Flow* en *Learning*.

Door middel van een synthese van wetenschappelijke en praktische bronnen, aangevuld met expertvalidatie geïnspireerd door de Delphi-methode, toont dit werk aan dat ASD-praktijken en organisatieproblemen consistent in een gedeelde driedimensionale ruimte kunnen worden gesitueerd. De resultaten wijzen uit dat het raamwerk fungeert als een methode-onafhankelijke vertaallaag die *shared understanding* bevordert tussen diverse stakeholders, van software-ontwikkelaars tot strategisch management. Dit onderzoek legt hiermee de fundering voor een holistische softwaretheorie die de kloof tussen methodologische voorschriften en organisatorische behoeften kan overbruggen.

Kernwoorden: *Agile Software Development; Framework; Project Management; Shared Understanding*

1 Introductie

Agile softwareontwikkeling (ASD) is wijdverspreid in de software-industrie (Alami e.a., 2022, Kuhrmann e.a., 2022, M. Iqbal e.a., 2024) en wordt vaak als de industriestandaard beschouwd (Martinez-Sanchez en Vicente-Oliva, 2023, Kuhrmann e.a., 2022). Desondanks wijzen er onderzoekers op de relevantie van alternatieve standaarden, zoals het PMBOK (*Project Management Body Of Knowledge*) (Sathe en Panse, 2023) en *dynamic capabilities and knowledge management* (Kaur, 2023). Analooq aan *Total Quality Management* beogen deze uiteenlopende benaderingen de weerbaarheid (non-fragility) van organisaties te vergroten (Taleb en Douady, 2013, Mariana da Silva Barbosa Gama en Bonamigo, 2024).

De diversiteit aan standaarden en implementatiemethoden leidt echter tot een gefragmenteerd kennislandschap. Er ontbreekt momenteel een eenduidige theorie die de praktijken, rollen en fenomenen binnen ASD structureert (Stray e.a., 2022).

Dit gebrek maakt objectieve prestatievergelijkingen, zowel tussen organisaties met uiteenlopende paradigma's (zoals Agile versus PMBOK) als tussen specifieke ASD-methoden onderling, vooralsnog onmogelijk. Hierdoor ontbreekt cruciaal inzicht in de onderliggende succesfactoren en de bedrijfsbrede impact van ASD-beslissingen (Berntzen e.a., 2023). Precies dit kennistekort draagt direct bij aan het veelvuldig falen van grootschalige ASD-initiatieven (Stray e.a., 2022, Edison e.a., 2022, Meckenstock, 2024, Kanaparan en Strode, 2025, Rolland e.a., 2023). Aangezien een binaire opdeling tussen 'Agile' en 'niet-Agile' in de praktijk ontoereikend is gebleken, is de ontwikkeling van een overkoepelende softwaretheorie essentieel voor onderzoeksdomeinen zoals software engineering, information systems en management science (Stray e.a., 2022, Kuhrmann e.a., 2022, Keshavarz e.a., 2024, Meckenstock, 2024, Kanaparan en Strode, 2025).

Om dit gat in de literatuur te adresseren, stelt dit onderzoek een methodeonafhankelijk raamwerk voor dat de diverse perspectieven op ASD-initiatieven structureert. Dit raamwerk fungeert enerzijds als een noodzakelijke academische voorbereiding: pas wanneer de definitie van ASD eenduidig is, wordt accuraat vervolgonderzoek naar de daadwerkelijke prestatieverschillen tussen diverse implementaties ervan mogelijk. Anderzijds is het model expliciet ontworpen met het oog op directe praktische toepasbaarheid. Het stelt organisaties direct in staat om hun eigen initiatieven te evalueren en de verwachte impact van strategische beslissingen op de bedrijfsprestaties beter in te schatten. Door een *shared understanding* te faciliteren, slaat het raamwerk een brug tussen theorie en praktijk, en bevordert het weloverwogen besluitvorming in de industrie (Stray e.a., 2022, Sathe en Panse, 2024).

Hoewel het Agile Manifesto richtinggevende kernwaarden biedt, wordt het in de academische literatuur overwegend als een filosofisch artefact beschouwd dat onvoldoende sturing biedt voor concrete, complexe implementaties (Beck e.a., 2001). De constructie van dit raamwerk vereist daarom een expliciete vertaalslag naar de concrete operationele structuren die moderne software engineering dicteert (Kuhrmann e.a., 2022). Om deze brug tussen abstractie en praktijk methodisch te borgen, is een concreet toepasbaar theoretisch model vereist (Kuhrmann e.a., 2022, Palopak en Huang, 2024, Keshavarz e.a., 2024). Sathe en Panse, 2024 legden hiertoe reeds een belangrijke basis door kwantitatieve faciliterende relaties (enabling-relaties) tussen tien Agile-praktijken aan te tonen, waaronder de invloed van *Continuous integration and continuous deployment (CI/CD)* op *Agile Release Trains* en het positieve effect van Agile coaching op een *collaborative culture*. Als antwoord op hun oproep tot verdere verfijning presenteert deze studie een expansie die 46 wetenschappelijke en 16 praktijkgerichte bronnen synthetiseert. Verder werd deze gevalideerd door industrie-experts. Het resulterende raamwerk mapt praktijken en doelen in een meerdimensionale ruimte om de volgende centrale onderzoeksvraag te beantwoorden: "Aan de hand van welke dimensies kan de alignering tussen Agile-praktijken en organisatiedoelen binnen een organisatie structureel in kaart worden gebracht en geëvalueerd?"

De rest van deze paper is als volgt gestructureerd: Sectie 2 definieert de gehanteerde ASD-perspectieven om theoretische ambiguïteit uit te sluiten. Sectie 3 beschrijft de methodologie, inclusief literatuurselectie, syntheses technieken en expertvalidatie. In Sectie 4 worden de resultaten en de werking van het voorge-

stelde raamwerk gepresenteerd. Tot slot omvat Sectie 5 een discussie van de academische en praktische implicaties, gevolgd door de eindconclusie in Sectie 6.

2 Literatuurstudie

Dit deel synthetiseert de bestaande literatuur rondom ASD om de theoretische fundering te leggen voor het CFL-raamwerk. Allereerst problematiseert het de heersende ambiguïteit in ASD-definities en perspectieven. Vervolgens analyseert het de typische succesfactoren en gehanteerde praktijken van ASD-implementaties, om ten slotte aan te tonen waar de huidige methoden op focussen en waarom een nieuwe, methode-onafhankelijke abstractielaag noodzakelijk is.

2.1 Gefragmenteerde perspectieven

Hoewel het wijdverspreide gebruik van Agile Software Development (ASD) de illusie van een universele standaard wekt (Alami e.a., 2022; M. Iqbal e.a., 2024; Kuhrmann e.a., 2022), ontbreekt in de academische literatuur een eenduidige, algemeen geaccepteerde definitie. Een analyse van de literatuur onthult grofweg drie dominante, maar gefragmenteerde perspectieven op het concept ASD.

Een eerste stroming hanteert een strikte benadering en verankert de definitie van ASD uitsluitend in de originele waarden en principes van het *Agile Manifesto* (Alami e.a., 2022; Beck e.a., 2001; Palopak en Huang, 2024; Saeeda e.a., 2023). Een tweede perspectief benadert ASD daarentegen vanuit een pragmatisch, probleemgestuurd kader. Binnen deze stroming wordt ASD niet gedefinieerd als een rigide set principes, maar eerder gepositioneerd als een methodologische oplossing voor specifieke organisatorische noden, zoals de noodzaak tot intensievere samenwerking en communicatie (Keshavarz e.a., 2024; Sathe en Panse, 2023; Yigitbas e.a., 2024). Een derde, meer lokaal toegepast perspectief conceptualiseert ASD louter op basis van de te verwachten uitkomsten. Deze literatuur vermijdt een formele ontologische definitie en reduceert ASD tot een verzameling van potentiële voor- en nadelen—variërend van een versnelde *time-to-market* tot verhoogde technische schuld (M. Iqbal e.a., 2024; Lee en Chen, 2023; Meckenstock, 2024).

Deze versnippering, waarbij ASD afwisselend wordt geframed als een dogma, een probleemoplosser of een set van symptomen, leidt tot een fundamentele ambiguïteit in de literatuur (Alami e.a., 2022; Keshavarz e.a., 2024; Meckenstock, 2024). Deze conceptuele onduidelijkheid vloeit deels voort uit de inherent lage mate van structuur in, en het exploratief karakter van, softwareontwikkelingsprocessen (*less-structured business processes*), wat organisaties dwingt tot het hanteren van zelf veranderende, weinig onderbouwde methodes om met verandering om te gaan (Goni en Van Looy, 2024). Precies deze theoretische verdeeldheid onderstreept de noodzaak voor een overkoepelend, methode-onafhankelijk raamwerk om orde te scheppen in de ASD-literatuur.

2.2 Pragmatische uitdagingen en succesfactoren

Naast het ontologische debat rondom ASD-definities (Kuhrmann e.a., 2022), richt een aanzienlijk deel van de literatuur zich op de pragmatische en operationele impact van Agile-implementaties. Hoewel ASD aantoonbaar de productiviteit kan verhogen en anticipeert op technologische innovaties (Lee en

Chen, 2023; Meckenstock, 2024; Sathe en Panse, 2023; Yigitbas e.a., 2024), gaat de adoptie ervan gepaard met complexe, structurele valkuilen. Een analyse van de geïdentificeerde succes- en faalfactoren in de literatuur onthult een duidelijke tweedeling tussen contextgerelateerde (organisatorische en procesmatige) en mensgerelateerde factoren (M. Iqbal e.a., 2024; Junior e.a., 2024; Meckenstock, 2024; Rolland e.a., 2023). Tot de contextuele dimensie behoren de parameters projectgrootte, schaalbaarheidsproblemen en de stabiliteit van de organisatiestructuur. De mensgerelateerde dimensie omvat daarentegen de psychologische en sociale tol van ASD, waaronder verhoogde werkdruk door iteratieve opleveringscycli, moeizame teamcohesie en competentiekloven (Kula e.a., 2022; Meckenstock, 2024).

De literatuur toont bovendien een asymmetrie in de beïnvloedbaarheid van deze succesfactoren aan. Terwijl structurele en methodologische projectfactoren, zoals het instellen van kleine teams of iteratieve cycli, relatief eenvoudig te sturen zijn, blijken persoonlijke en culturele factoren weerbarstig (Junior e.a., 2024). Deze transitie wordt verder gecompliceerd door cognitieve patronen. Wanneer medewerkers bijvoorbeeld de overstap maken naar de multidisciplinaire, *cross-functional* rollen die ASD vereist, lopen zij het risico in de beginfase hun eigen competenties te overschatten, een fenomeen dat bekendstaat als het Dunning-Kruger effect (Kruger en Dunning, 1999; Meckenstock, 2024). Dergelijke psychologische valkuilen en menselijke barrières ondermijnen direct de theoretische effectiviteit van de gekozen Agile-methoden.

2.3 Concrete ASD-praktijken en hun methodologische oorsprong

Zoals eerder vastgesteld, lijdt het overkoepelende concept ASD aan fundamentele ambiguïteit. Desondanks bestaat er in de literatuur en de industrie meer eensgezindheid over de operationele invulling ervan, namelijk via de onderliggende, tastbare praktijken. Omdat individuele ASD-praktijken in de literatuur strikter gedefinieerd zijn dan het abstracte Agile-label, vormen zij robuustere ankerpunten voor de methodologie. Om deze praktijken echter niet in een conceptueel vacuüm te plaatsen, is het essentieel ze te verankeren in de specifieke methoden waarin ze opgebouwd en voorgesteld zijn. Dit onderzoek hanteert deze wijdverspreide, historisch verankerde praktijken dan ook bewust als inductieve bouwstenen voor het beantwoorden van de onderzoeksvraag.

De literatuur identificeert een brede waaier aan dergelijke geaccepteerde praktijken, waarvan de procesmatige (of structurele) ankers hun wortels hebben in de vroege iteratieve raamwerken (Sathe en Panse, 2023). Zo is het concept van strikte tijdsafbakening ter bevordering van voorspelbaarheid, algemeen bekend als *timeboxing*, fundamenteel gepioneerd door Scrum via de introductie van zogeheten Sprints (Lee en Chen, 2023; Sutherland en Schwaber, 2012). Kanban daarentegen formaliseerde de focus op procesvloeiendheid via continue opleveringskadansen, om zo korte feedbackcycli te creëren met de stakeholders (Anderson, 2010). Naarmate in de industrie de behoefte ontstond om deze teamgerichte concepten op organisatieniveau toe te passen, ontwikkelden schalingsraamwerken zoals SAFe nieuwe, overkoepelende synchronisatiemechanismen. Praktijken als *Program Increments (PI)* en *Agile Release Trains (ART)* bouwden conceptueel voort op deze initiële principes om operationele iteraties, over tientallen teams heen, af te stemmen op strategische bedrijfsdoelen (Ameta e.a., 2022; Studio, 2023).

Naast deze temporele en procesmatige ankers, pionierden specifieke ASD-methoden ook mensgerichte en collaboratieve praktijken, gericht op continue feedback en structurele kennisdeling (J. Iqbal

e.a., 2019). De *daily standup*, inmiddels een alomtegenwoordig synchroon afstemmingsmoment dat is overgenomen in vrijwel alle moderne raamwerken zoals SAFe en LeSS, werd oorspronkelijk door Scrum op de kaart gezet om dagelijkse blokkades proactief te identificeren (OConnor, z.d.; Sutherland en Schwaber, 2012). Aan de meer technische zijde van de samenwerkingsvormen was het eXtreme Programming (XP) dat radicale concepten zoals *pair programming* opperde, waarbij continue strategische code-reviews direct in het ontwikkelingsproces werden ingebakken (Andres en Beck, 2005).

2.4 De noodzaak voor een overkoepelende theorie

Uit de veelheid aan methodologische variaties, de contextafhankelijke succesfactoren en de fundamentele ambiguïteit in definities, volgt een onmiskenbare conclusie: empirische studies meten vaak wezenlijk andere fenomenen onder dezelfde noemer “Agile”. Dit sterk gefragmenteerde landschap vereist de ontwikkeling van een algemeen geaccepteerde, overkoepelende theorie (Stray e.a., 2022). Precies deze theoretische en praktische verdeeldheid onderschrijft de relevantie van dit onderzoek. Om de daadwerkelijke prestatieverschillen tussen diverse ASD-implementaties objectief te kunnen analyseren, is een methode-onafhankelijke abstractielaag noodzakelijk.

3 Methoden

Dit onderzoek hanteert een tweeledig onderzoeksontwerp om de besluitvorming binnen software-initiatieven te objectiveren: de constructie van een conceptuele raamwerk dat *shared understanding* faciliteert en een aansluitende empirische expertvalidatie.

Het methodologisch fundament bouwt voort op Donnelly en Hughes, 2023. Dit is specifiek afgestemd op Agile-implementaties via de integratie van de literatuurstudiebenadering van Psychogios, 2024 voor theorieopbouw, een iteratieve validatieopstelling volgens Goni en Van Looy, 2024, en een formele expertconsultatie naar het model van Mariana da Silva Barbosa Gama en Bonamigo, 2024.

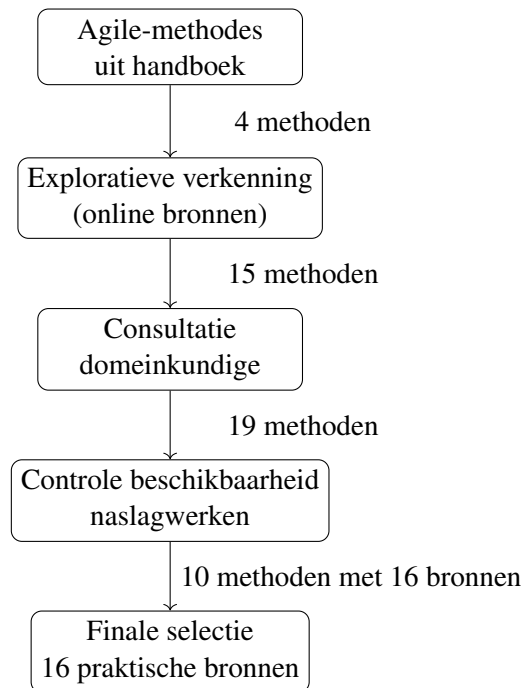
3.1 Opbouw raamwerk

De constructie van het raamwerk omvat drie fasen: een *bottom-up* extractie uit praktijkgerichte methodedocumentatie, een *top-down* systematische literatuurstudie uit academische bronnen, en de uiteindelijke synthese van beide perspectieven tot één geaggregeerd raamwerk.

3.1.1 Praktische opbouw

Om de praktische validiteit te borgen, baseert de eerste fase zich op documentatie van gangbare Agile-methoden. Industriestandaarden zijn vastgesteld aan de hand van grijze literatuur, industrieraapporten en platformdocumentatie (Atlassian, 2025; Digital.ai, 2025; Scrum.org, z.d.). Dit leverde een corpus van 16 bronnen op dat tien toonaangevende methoden representeert: Kanban (Anderson, 2010), Scrum (Scrum.org, z.d.), SAFe (Studio, 2023), eXtreme Programming (Andres en Beck, 2005), Theory of Constraints (Goldratt en Cox, 2014), Lean Startup (Ries, 2011), Lean (Modic en Åhlström, 2012), LeSS (Company, 2020), DevOps (Microsoft, 2022) en het Spotify Model (Cruth, 2012).

Selectie van de praktische bronnen De bronselectie verliep iteratief (zie figuur 1). Een initiële set, gebaseerd op het academische handboek Wysocki, 2019, is via expertconsultatie en platformexploratie uitgebreid. Enkel methoden met consistente, diepgaande brondocumentatie over concrete praktijken als Andres en Beck, 2005 en Modic en Åhlström, 2012 zijn geselecteerd. Methoden zonder robuust bronnenkader werden geëlimineerd.



Figuur 1: Proces van selectie van de praktische bronnen

Extractie patronen Aangezien het concept ASD aan fundamentele ambiguïteit lijdt, richt dit onderzoek zich op de onderliggende, tastbare praktijken. Deze zijn in de literatuur strikter gedefinieerd en vormen daardoor robuustere methodologische ankerpunten. Om deze reden zijn 16 geselecteerde bronnen systematisch gedeconstrueerd om abstracte methodieken te herleiden tot vergelijkbare elementen.

De selectie van de geanalyseerde methoden, gebaseerd op breed gedragen industrieraapporten waaronder het State of Agile rapport, resulteerde in de extractie van 112 unieke elementen (bijv. *Daily Stand-up*, *Squads*). Deze zijn elk geanalyseerd op hun beoogde organisatorische impact. Deze effecten zijn gecategoriseerd in negen impacttypes (waaronder voorspelbaarheid en symbiose), wat resulteerde in 55 unieke patronen (zie tabel 10 in appendix D). Elk element is vervolgens op basis van zijn impactprofiel gepositioneerd in een multidimensionale ruimte om zijn operationele voetafdruk visueel te duiden. Dit wordt verder besproken in de resultaten.

- **Identificatie van elementen:** Uit de 16 praktische bronnen werden alle specifieke rollen, praktijken, principes en metrieke (KPI's) geëxtraheerd. Dit resulteerde in 112 unieke elementen (bijv. *Daily Stand-up*, *Squads*, *Pull System*).
- **Impact-analyse:** Voor elk element werd op basis van de brondocumentatie bepaald welke specifieke verbetering of "impact" het beoogt te realiseren binnen een organisatie.

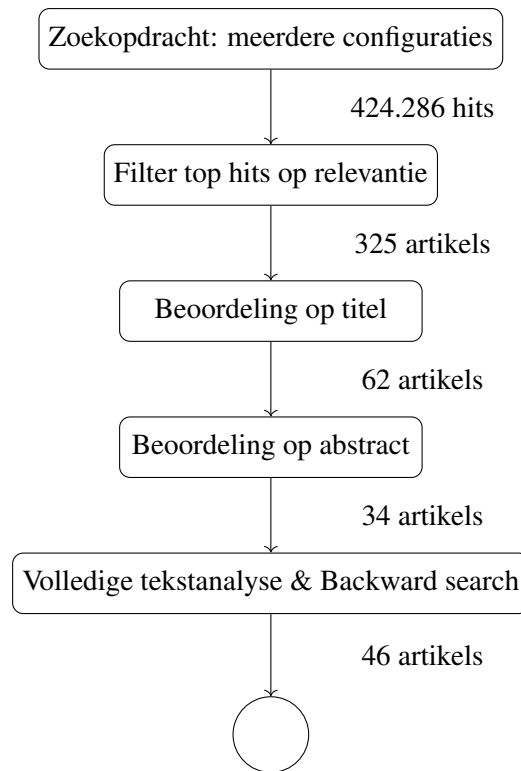
- **Cross-referencing en patroonherkenning:** De effecten werden gecategoriseerd in negen specifieke impacttypes: voorspelbaarheid, productkwaliteit, proceskwaliteit, transparantie WIP over teams, transparantie WIP in teams, symbiose, waardecreërend versus niet-waardecreërend werk, kennisdeling en kenniswinning. Door elementen met dezelfde impactprofielen te groeperen, konden 55 unieke patronen worden geïdentificeerd.
- **Positionering in de multidimensionale ruimte:** Op basis van de gemiddelde scores van de onderliggende impacttypes werd elk element en elke methode gepositioneerd binnen drie dimensies, elk bestaande uit drie impacttypes. Dit maakt het mogelijk om de “voetafdruk” van een praktijk (bijv. de focus van een *Squad* op de dimensie *Collaboration*) op een visueel overzichtelijke wijze te duiden.

Dit extractieproces vereist de reductie van complexe praktijken tot hun abstracte effecten (Jooker, 2025). Dit onderzoek veronderstelt dat wezenlijk verschillende praktijken met een identiek beoogd impactpatroon binnen een ASD-context functioneel uitwisselbaar zijn (Assumptie 1). Twee voorbeelden hiervan zijn de *team PI goals* van SAFe en de *sprint goal* bij Scrum of *pay-per-use* binnen eXtreme Programming en *validated learning* in Lean Startup.

Assumptie 1 (Impactpatronen in een ASD-context). *Elke praktijk binnen een ASD-context is methodologisch reduceerbaar tot een uniek patroon van abstracte impacttypes. Bijgevolg resulteert het toepassen van verschillende praktijken met een identiek impactpatroon in een functioneel vergelijkbare organisatorische uitkomst.*

3.1.2 Wetenschappelijke opbouw

Selectie van de wetenschappelijke bronnen De theoretische fundamenten zijn gedestilleerd via een systematische literatuurverkenning (zie figuur 2). De inclusiecriteria vereisten Engelstalige, *peer-reviewed*, *full-text* publicaties van de laatste drie jaar binnen *computer science* of *business*. Oudere bronnen werden enkel toegelaten tot de bestudeerde literatuur via *backwards search*. Een iteratief afbakeningsproces van 424.286 initiële hits (gebaseerd op zes zoektermconfiguraties en algoritme-relevantie) leidde na trapsgewijze filtering en *backward search* tot een finaal corpus van 46 relevante bronnen. De gehanteerde zoektermcombinaties zijn “*Agile software development*”, “*Agile software development & knowledge-based resources*”, “*Framework development*”, “*Framework design*”, “*Strategic framework*” en “*Strategic framework & Agile project management*”.



Figuur 2: Schematische weergave van het systematische selectieproces van wetenschappelijke bronnen voor het raamwerk

Extractie patronen Via een thematische inhoudsanalyse van deze 46 bronnen zijn de kernperspectieven binnen ASD geïdentificeerd. Na open codering op faal- en succesfactoren werden overlappende concepten gegroepeerd en gesynthetiseerd tot drie orthogonale dimensies: *Flow*, *Learning* en *Collaboration*.

Theoretisch steunt deze driedimensionale ruimte op de *Task-Technology Fit* theorie (Howard en Rose, 2019), die stelt dat methodologische effectiviteit afhangt van de aansluiting bij organisatorische noden. Dit impliceert dat praktijken (oplossingen) en doelen (problemen) operationeel compatibel zijn indien ze zich op dezelfde assen bevinden (Assumptie 2).

Het analyseproces bestond uit drie opeenvolgende fasen:

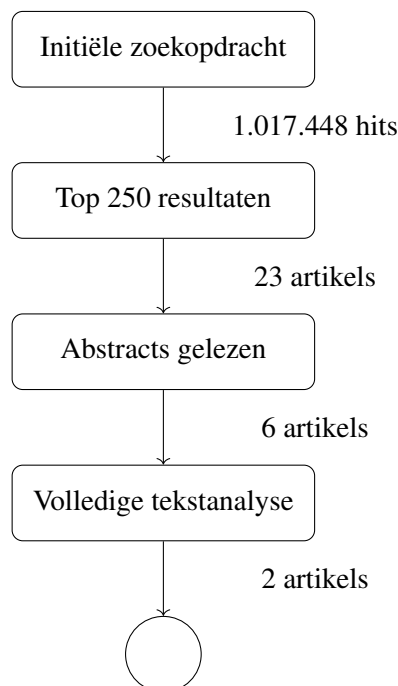
- **Open codering:** Elk artikel werd gescand op expliciete concepten die bijdragen aan het succes of de belemmering van ASD-initiatieven. Hierbij werden termen als “doorstroom”, “kennismanagement”, “teamvertrouwen” en “leiderschap” geëxtraheerd.
- **Categorisering en groepering:** De geëxtraheerde concepten werden vergeleken op basis van hun onderzoeksfocus. Hierbij werd gezocht naar patronen in de literatuur. Werken die focusten op procesoptimalisatie en snelheid werden samengebracht, evenals werken die de nadruk legden op de menselijke kant of de cognitieve aspecten van softwareontwikkeling.
- **Synthese tot dimensies:** Uit deze groepering werden drie overkoepelende wetenschappelijke perspectieven gedestilleerd. Deze fungeren als de fundamentele dimensies voor het verdere raamwerk.

Assumptie 2 (Matching principe). *Conceptuele nabijheid binnen de driedimensionale ruimte van het raamwerk is een indicator voor operationele compatibiliteit. Indien een praktijk en een organisatorisch probleem zich in dezelfde regio situeren, vormt de praktijk een geschikte praktische reactie op dat probleem.*

3.2 Validatiemethode

Voor de empirische validatie hanteert dit onderzoek een hybride, iteratieve methodologie, fundamenteel gebaseerd op de validatieprotocollen van Donnelly en Hughes, 2023 en Goni en Van Looy, 2024. Door middel van meerdere individuele feedbackrondes wordt academische consensus bereikt zonder dat dominante stemmen de data beïnvloeden (Ven en Delbecq, 1974, Dalkey en Helmer, 1963; Paré e.a., 2013). Deze benadering is gegrond in een interpretivistische epistemologie (Edwards, 2017; Symon e.a., 2018), waarbij de subjectieve implementatie-ervaringen van respondenten actief worden benut om het raamwerk te optimaliseren.

Selectie raamwerk-bronnen De methodologische basisbronnen voor deze validatie zijn geselecteerd via een systematische zoekopdracht (zoekterm “*Framework*”) naar *frameworks* in de universiteitsbibliotheek (zie figuur 3). Van de initiële 1.017.448 hits bleven na strikte filtering en kwalitatieve weging twee studies (Donnelly en Hughes, 2023; Goni en Van Looy, 2024) over, geselecteerd op basis van hun maximale methodologische relevantie voor deze validatieopzet.



Figuur 3: Systematisch selectieproces van wetenschappelijke bronnen voor validatie

Expertselectie Een hybride panel van 11 experts (8 praktijkexperts, 3 academici) borgt een balans tussen theorie en praktijk (Goni en Van Looy, 2024). Het zwaartepunt op de praktijk reflecteert de interpretivistische focus op reële implementatiedynamiek (Donnelly en Hughes, 2023). Academici vereisten een gerelateerd doctoraat en lidmaatschap van een relevante onderzoeksgroep. Praktijkexperts dienden

minimaal vijf jaar IT-ervaring te hebben in rollen zoals Agile-expert of projectmanager (Donnelly en Hughes, 2023).

3.2.1 Dataverzameling en analyse

De dataverzameling is methodologisch gestructureerd in drie fasen (Goni en Van Looy, 2024; Schmiedel e.a., 2013), zoals weergegeven in tabel 1. Na een eerste validatieronde ($n = 8$) ter verfijning van het model, volgde een formele validatiefase ($n = 6$). Voor extra betrouwbaarheid (*member checking*) vonden bij drie respondenten verdiepende vervolgesprekken plaats.

Tabel 1: Overzicht interviewproces

Ronde	Input	Output
Vorbereidingsfase	Start selectie experts en Check expertcriteria	Lijst van geselecteerde experts
Eerste ronde – Initiële feedback	Achtergrond respondent en Gerichte vragen naar expertkennis	Overzicht van opinies van experts
Tweede ronde – Finale validatie	Verifieer verwerkte feedback en Verzamel nieuwe expert opinies	Gevalideerd raamwerk

Tijdens de interviews definieerden respondenten specifieke concepten en positioneerden deze in de driedimensionale ruimte. Concepten zonder aanwezige praktijkervaring van de respondent werden genegeerd. Uitsluitend concepten met $n \geq 4$ waarnemingen zijn in de finale dataset opgenomen (zie appendix E).

Om *response bias* en interbeoordelaarsvariabiliteit te neutraliseren, werd de dataset per respondent genormaliseerd (zie Python-script in appendix H). De totale impactscore $T_{i,j}$ van concept j door respondent i is de som van de ruwe assenscores:

$$T_{i,j} = C_{i,j} + F_{i,j} + L_{i,j} \quad (1)$$

De maximale impactscore van respondent i over de sessie heen diende als normalisatiefactor M_i :

$$M_i = \max_j(T_{i,j}) \quad (2)$$

Hierdoor ontstaat voor elke dimensie een vergelijkbare, relatieve coördinaat tussen 0 en 1 (hier geïllustreerd voor *Flow*):

$$\tilde{F}_{i,j} = \frac{F_{i,j}}{M_i} \quad (3)$$

Waarbij:

- i : de index van de respondent.
- j : de index van het specifieke concept (het antwoord).
- $T_{i,j}$: de totale impactscore van concept j toegekend door respondent i .
- $C_{i,j}, F_{i,j}, L_{i,j}$: de ruwe scores voor respectievelijk de dimensies *Collaboration*, *Flow* en *Learning*.

- M_i : de maximale totale impactscore die door respondent i over alle concepten heen is toegekend.
- $\tilde{F}_{i,j}$: de genormaliseerde, relatieve score voor de dimensie *Flow* (deze logica is analoog toepasbaar op de andere dimensies).

Abstracte problemen en doelen werden geëvalueerd via een 7-punts Likert-schaal (van “extreem laag” tot “extreem hoog”) om cognitieve overbelasting te voorkomen. Deze ordinale data is getransformeerd naar een discrete waardeschaal (0-6) en onderworpen aan dezelfde normalisatielogica, waardoor praktijken en problemen vergelijkbaar worden geplot.

De kwantitatieve analyse bestaat uit de berekening van gemiddelden en standaardafwijkingen, aangevuld met een drietraps statistische toetsing op as-verschillen: de Friedman-test voor hoofd-effecten, gevolgd door een Wilcoxon signed-rank test voor individuele koppels, en een Bonferroni-correctie om vals-positieven te reduceren (Pereira e.a., 2015).

3.2.2 Validiteit en betrouwbaarheid

Om de interne consistentie systematisch te evalueren, zijn expliciete acceptatiecriteria (AC) vastgesteld (Goni en Van Looy, 2024). De 3D-ruimte is opgedeeld in acht gelijke **octanten**, gescheiden door de mediaan van elke as. Tijdens de interviews fungeerden de onderstaande criteria diagnostisch voor iteratieve verfijning, en finaal als drempelwaarden voor wetenschappelijke robuustheid en pragmatische bruikbaarheid.

Acceptatiecriteria voor positionering van problemen en doelen

- AC-P1** Minstens 75% van de experts plaatst zes van de zeven problemen/doelen in exact dezelfde octanten.
- AC-P2** Ten minste één dimensie (*Collaboration*, *Flow* of *Learning*) verschilt statistisch significant van de andere twee ($p < 0,25$; gecorrigeerde Friedman-test), wat een expliciete, dominante positionering garandeert.

Acceptatiecriteria voor praktijken en methoden

- AC-M1** Geen enkele expert positioneert dezelfde praktijk/methode met hoge zekerheid in tegenovergestelde extrema van de ruimte.
- AC-M2** Minstens 75% van de experts plaatst zes van de zeven praktijken/methoden in dezelfde octanten.
- AC-M3** Ten minste één dimensie verschilt statistisch significant van de andere twee ($p < 0,25$; gecorrigeerde Friedman-test).

Acceptatiecriteria voor overeenkomst probleem-oplossing (Matching)

- AC-A1** Passende problemen en oplossingen worden door minstens 75% van de respondenten in hetzelfde octant gepositioneerd.

AC-A2 Niet-passende problemen en oplossingen worden door minstens 75% van de respondenten in verschillende octanten gepositioneerd.

Acceptatiecriteria voor leerzaamheid

AC-L1 Respondenten geven het CFL-raamwerk gemiddeld een score van ≥ 5 (op 7) voor het faciliteren van een *shared understanding*.

AC-L2 Geen enkele individuele respondent scoort het raamwerk lager dan 3 op 7.

4 Resultaten

Deze sectie presenteert de onderzoeksresultaten in twee delen: de constructie van het conceptuele raamwerk en de aansluitende expertvalidatie. De validatie omvat zowel kwantitatieve analyses als kwalitatieve observaties uit de interviews.

4.1 Raamwerk

4.1.1 Praktische Resultaten

De tien geanalyseerde methoden bevatten in totaal 112 unieke prestatieverhogende elementen (rollen, praktijken, principes en KPI's), zoals de *squad* (Spotify), het *pull system* (Kanban) en de *daily standup* (Scrum). Een compleet overzicht inclusief beoogde doelen is opgenomen in Appendix D.

Uit de brondocumentatie zijn negen unieke impacttypes geëxtraheerd en logisch gegroepeerd in drie hoofddimensies (zie tabel 2): **Flow** (continue waarde-oplevering via voorspelbaarheid, product- en proceskwaliteit), **Collaboration** (symbiotische samenwerking en WIP-transparantie¹) en **Learning** (kenniscreatie, kennisdeling en de focus op waardecreërend werk).

Tabel 2: Overzicht van impacttypes per impactgroep

Impactgroep	Impacttype
<i>Flow</i>	Voorspelbaarheid
	Productkwaliteit
	Proceskwaliteit
<i>Collaboration</i>	Transparantie van lopend werk (WIP) over teams heen
	Transparantie van WIP binnen teams
	Symbiotisch samenwerken
<i>Learning</i>	Waarde- versus niet-waardecreërend werk
	Kennisdeling
	Kenniswinning

De 112 elementen zijn systematisch beoordeeld op hun bijdrage aan deze negen impacttypes, wat resulteerde in 55 unieke impactpatronen. Dit bracht functionele overlap aan het licht tussen schijnbaar verschillende praktijken, zoals *validated learning* (Lean Startup Ries, 2011) en *Inspect and Adapt* (SAFe Studio, 2023), of de *Daily Scrum* (LeSS Company, 2020, Scrum Scrum.org, z.d.) en de *Daily meeting* (Kanban).

¹WIP (*work-in-progress*) vindt zijn oorsprong in Little's Law Little, 1961.

Door methoden te deconstrueren tot hun fundamentele elementen, kunnen zij ruimtelijk worden gepositioneerd langs de drie orthogonale assen: **Collaboration**, **Flow** en **Learning**. Een praktijk als *Squads* (Cruth, 2012) correleert sterk met **Collaboration** en adresseert zodoende primair samenwerkingsproblemen. Een *Sprint Retrospective* (Scrum.org, z.d.) richt zich daarentegen op **Learning**. Voor de validatiestap zijn vier praktijken als volgt gekwantificeerd (scores op Collaboration, Flow, Learning): de *daily standup* focust op **Collaboration** en secundair **Learning** (0.67, 0, 0.33), de sprint integreert **Flow** en **Collaboration** (0.5, 0.5, 0), korte feedbackcycli domineren op **Learning** (0, 0, 1) en *pair programming* combineert **Collaboration** met **Learning** (0.5, 0, 0.5).

4.1.2 Wetenschappelijke Resultaten

Thematische analyse van 46 wetenschappelijke bronnen onthult drie dominante ASD-perspectieven die de praktische dimensies theoretisch funderen:

Flow In de literatuur is de voorspelbare waardeestroom (Anderson, 2010) het meest onderzochte perspectief. Kanaparan en Strobe, 2025 definieert succesvolle initiatieven via een vloeiende workflow met minimale verstoringen, terwijl Sathe en Panse, 2023 focussen op procesbottlenecks. Studies bevestigen de positieve impact van *early estimation* (Rivera Ibarra e.a., 2024) en continu testen (Singh e.a., 2025) op werkdoorstroom. Hoewel continue oplevering niet exclusief is voor ASD (Saeeda e.a., 2023), dicteert **Flow** de opleveringskadans van softwareontwikkeling.

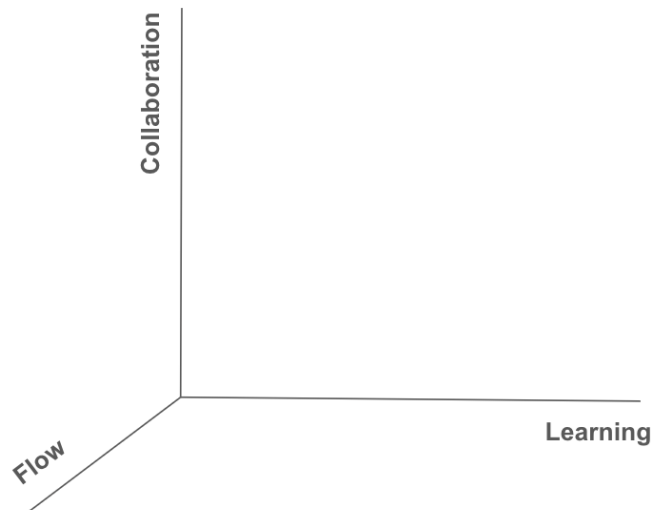
Learning Een tweede perspectief beschouwt organisaties vanuit hun kennisgebaseerde capaciteiten. Effectief kennismanagement en -integratie (Idrees e.a., 2024; Nasution e.a., 2023; Ouriques e.a., 2023) sturen ASD-innovatie. Dit sluit aan bij Agile-fundamenten waar werkende software fungeert als primaire kennisdrager ten nadele van zware documentatie (Beck e.a., 2001; Lee en Chen, 2023; Sathe en Panse, 2023). Literatuur toont aan dat organisatiebrede kennisdeling leidt tot *multi-skilled* teams, verhoogde kwaliteit en superieure innovatiecapaciteit (Karhapaa e.a., 2024; Martinez-Sanchez en Vicente-Oliva, 2023; Nasution e.a., 2023).

Collaboration Het derde perspectief adresseert de socio-technische complexiteit van ASD. Menselijke factoren en onderling vertrouwen bepalen in grote mate projectsucces (Junior e.a., 2024; Meckenstock, 2024), temeer daar samenwerkingscomplexiteit op schaal super-lineair toeneemt (Rolland e.a., 2023). Holistische studies beschouwen teamafstemming en coördinatie als kritische Agile-capaciteiten (Kanaparan en Strobe, 2025; Martinez-Sanchez en Vicente-Oliva, 2023). Verder analyseert dit domein groepsdynamiek (Berntzen e.a., 2023; Edison e.a., 2022), virtuele tools (Yigitbas e.a., 2024), en de adaptieve waarde van gedistribueerd leiderschap (Mariana da Silva Barbosa Gama en Bonamigo, 2024).

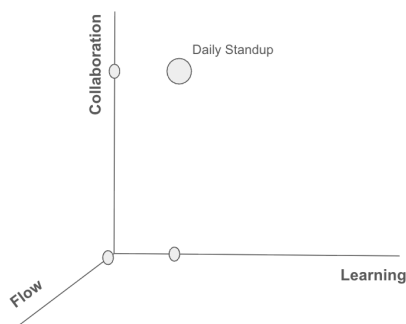
4.1.3 Aggregatie

De convergentie van de *bottom-up* praktijkanalyse en *top-down* literatuurstudie resulteert in het **Collaboration-Flow-Learning** (CFL) raamwerk. Deze driedimensionale ruimte ordent kwantitatief zowel

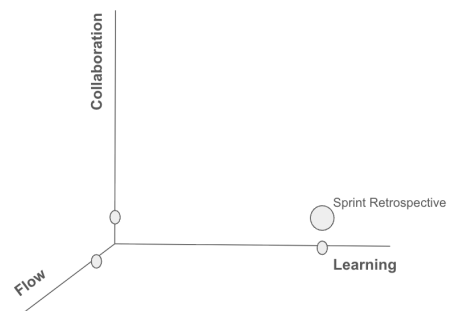
organisatorische problemen (noden) als praktijken (oplossingen). Zo plot een geografisch samenwerkingsprobleem (Yigitbas e.a., 2024) primair op de **Collaboration**-as, terwijl procesbottlenecks (Sathe en Panse, 2023) uitslaan op **Flow**. De diagnostische kracht van het model schuilt in de ruimtelijke interactie: operationele besluitvorming wordt geoptimaliseerd door problemen te matchen met praktijken die zich in een gelijkaardige dimensionale regio situeren (geïllustreerd in figuren 4a tot en met 4c).



(a) CFL raamwerk



(b) CFL Daily Standup



(c) CFL Retrospective

Figuur 4: Overzicht van de CFL-activiteiten

4.2 Validatie

4.2.1 Kwantitatieve analyse

De validatie baseert zich op de genormaliseerde scores van elf experts (zie tabellen 5 t.e.m. 8 en visualisaties in appendix G). Enkel concepten met minimaal vier onafhankelijke beoordelingen zijn geanalyseerd (zie appendix E).

Bij de praktijken wijzen lage standaardafwijkingen op consensus omtrent hun dimensionele impact.

De *daily stand-up* toont een uitgesproken **Collaboration**-focus (0.333 vs. 0.158 en 0.169), terwijl korte feedbackcycli domineren op **Learning** (0.386). *Time boxing* presteert sterker op **Flow** (0.319), en *pair programming* vertoont een duale piek op **Collaboration** (0.336) en **Learning** (0.340) met een minimale **Flow**-impact (0.231). Friedman-testen (tabel 3) valideren significante as-verschillen, voornamelijk bij de hoofddimensies van de *daily standup* en feedbackcycli. Cruciaal is dat de expertbeoordelingen voor de *daily stand-up*, korte feedbackcycli en *pair programming* exact overeenkomen met de theoretische opbouw; enkel bij *time boxing* divergeerde de empirische inschatting licht op de **Collaboration**-as.

Tabel 3: Friedman-test waarden praktijken (0.25 significantieniveau)

Praktijk	Friedman	Flow- Collaboration	Collaboration- Learning	Flow- Learning
Daily stand-up	0.061*	0.107*	0.083*	1.000
Time boxing	0.109*	0.484	1.000	0.369
Korte feedbackcycli	0.013*	0.414	0.204*	0.089*
Pair programming	0.174*	0.326	1.000	0.675

Ook bij de positionering van organisatorische problemen tekent zich consensus af. Een *needs-requirements mismatch* beïnvloedt primair de **Flow** (0.342). *Collaboration across time zones* (0.332) en *verschillende prioriteiten in multi-teams* (0.385) zijn uitgesproken **Collaboration**-vraagstukken. *Researching new knowledge* (0.379) en *key person dependencies* (0.373) plotten prominent op **Learning**. Hoewel de p-waarden (tabel 4) deels inboeten aan statistische kracht door de steekproefgrootte, wijzen de significante varianties systematisch op een gedeelde perceptie onder experts over het ruimtelijke zwaartepunt van ASD-problemen.

Tabel 4: Friedman-test waarden problemen en doelen (0.25 significantieniveau)

Probleem	Friedman	Flow- Collaboration	Collaboration- Learning	Flow- Learning
Needs-requirements mismatch	0.131*	1.000	0.747	0.527
Collaboration across time zones	0.163*	0.132*	1.000	0.612
Researching new knowledge	0.004*	0.326	0.094*	0.094*
Key person dependencies	0.424	1.000	0.432	0.592
Multi-teams hebben elk verschillende doelen en prioriteiten	0.092*	0.375	0.952	1.000

Gebruikersperceptie en Acceptatiecriteria Het CFL-raamwerk scoorde gemiddeld 5.55 op 7 inzake “leerzaamheid” ($n = 9$). Experts rapporteerden dat het model abstracte Agile-discussies objectificeert en direct toepasbaar is, waarmee de kwalitatieve acceptatiecriteria **AC-L1** en **AC-L2** duidelijk behaald zijn. Wegens het datareductie-criterium ($n \geq 4$) was de steekproef ontoereikend om rigide validatie-uitspraken te doen over probleem-oplossingsmatching (**AC-A1**, **AC-A2**) en extremen (**AC-P2**). Aan de kerneisen voor modelmatige positionering (**AC-M1**, **AC-M2**, **AC-M3** en **AC-P1**) wordt methodologisch voldaan, met inachtneming van de statistische gevoeligheid inherent aan het expertoverleg.

4.2.2 Kwalitatieve analyse

De interviews genereerden drie fundamentele kwalitatieve observaties die fungeren als contextuele randvoorwaarden voor de implementatie van het raamwerk:

1. **Autonomie en besluitvorming:** Zonder reële, gedistribueerde beslissingsbevoegdheid voor teams fragmenteert de impact van Agile-praktijken, wat de organisatorische wendbaarheid fnuikt.
2. **Dimensionale hiërarchie:** Meerdere experts duiden *Flow* aan als het uiteindelijke rendementsdoel, waarbij *Collaboration* en *Learning* fungeren als noodzakelijke *enablers*. Dit aligneert met Porter, 1979: voorspelbare oplevering (*Flow*) faalt zonder klantrichting (*Learning*) en optimale structuren (*Collaboration*). Andere respondenten prefereerden een puur egalitaire, contextgestuurde weging van de assen.
3. **Contextuele relativiteit:** De operationele context beïnvloedt de exacte coördinaten van een praktijk. De positionering van *time boxing* devalueert of escaleert afhankelijk van de tijdschaal (uren vs. kwartalen) en de teamcompositie. Dit is een logisch gevolg van het feit dat er verschillende doelen zijn om *time boxing* te hanteren. Twee voorbeelden zijn het opzetten van vaste leermomenten met een focus op *Learning* en het creëren van een opleveringskadans vanuit het *Flow*-perspectief. Evenzo verschuift de dimensionale voetafdruk van een *daily stand-up* significant wanneer de context wisselt van co-locatie naar gedistribueerde teams.

5 Discussie

Het driedimensionale CFL-raamwerk (*Collaboration*, *Flow*, *Learning*) expliciteert de onderliggende perspectieven van Agile Softwareontwikkeling (ASD) concepten. Door praktijken, methoden, doelen en problemen ruimtelijk te situeren, objectificeert het model uiteenlopende, vaak impliciete Agile-assumpties en faciliteert het gerichte, datagedreven besluitvorming. Deze sectie evalueert de bruikbaarheid van het raamwerk door de consensus en discrepanties tussen expertbeoordelingen te analyseren, en sluit af met een kritische reflectie op de methodologische validiteit en beperkingen.

5.1 Consensus

De negen meest bevraagde concepten vertonen een duidelijke consensus aangaande hun dimensionale positionering, gekenmerkt door significante afwijkingen in hun as-scores (bijvoorbeeld korte feedbackcycli met gemiddeld 0.386 op *Learning*, versus 0.150 voor *Flow* en 0.276 voor *Collaboration*). Echter, bij praktijken buiten hun dagelijkse expertise gaven respondenten doorgaans meer diffuse, structureel hogere scores (+50%) over alle dimensies heen. Deze theoretische overschatting wijst op een manifestatie van het Dunning-Kruger effect (Kruger en Dunning, 1999).

Een diffuus, uniform hoog profiel duidt zelden op een feilloze heilmiddel, maar wijst in de praktijk op een gebrek aan *shared understanding*. Zonder dit gedeelde begrippenkader stagneert collectieve besluitvorming. Het CFL-raamwerk objectificeert deze impliciete assumpties, waardoor concepten niet

langer benaderd worden als universele hulpmiddelen, maar als gerichte interventies met specifieke organisatorische perspectieven. Deze focus stelt teams in staat concrete, verifieerbare doelen te formuleren, een noodzaak die vier respondenten onafhankelijk van elkaar benadrukten.

Daarnaast fungeert het raamwerk als een methode-onafhankelijke vertaalaag. Besluitvorming rondom ASD-initiatieven verzandt vaak in technisch jargon of discussies over specifieke methodieken (zoals Scrum of SAFe). Het CFL-model dwingt alle stakeholders om te communiceren op een gedeeld abstractieniveau, gericht op de beoogde impact: *Collaboration*, *Flow* of *Learning*. Hierdoor verschuift de strategische dialoog van operationele vormvereisten naar fundamentele organisatiedoelen, wat een inclusievere en efficiëntere besluitvorming over disciplines heen mogelijk maakt.

5.2 Verschillen

Discrepancies in de expertbeoordelingen zijn grotendeels te herleiden tot verschillen in professionele achtergrond. Zo beschouwde een subset van respondenten *Flow* als het absolute einddoel van elk initiatief. Zij classificeerden secundaire effecten van praktijken (zoals indirecte doorstroomverbetering via samenwerking) systematisch als primaire *Flow*-impact.

Dit reflecteert de intuïtieve, toch problematische aanname dat het efficiënt produceren van werk per definitie waarde creëert. Zoals Porter, 1979 beargumenteert, draagt een verhoogde doorstroom zonder geverifieerde klantwaarde niet bij aan strategische bedrijfsdoelstellingen. Het ontdekken van deze waarde is een actief leerproces (*Learning*), structureel ingebed in praktijken als *Pay Per Use* of *validated learning*. Hoewel *Flow* een valide primair doel kan zijn, toont het raamwerk aan dat wendbaarheid evengoed vergroot wordt door te investeren in kennisverwerkingscapaciteit of samenwerkingsstructuren. Bij interpretatieverschillen fungeert het CFL-raamwerk als een katalysator: het dwingt partijen hun perspectieven te kristalliseren, neutraliseert impliciete misverstanden en resulteert in een robuustere *shared understanding*.

5.3 Validiteit en Beperkingen

De resultaten van deze studie dienen geïnterpreteerd te worden in de context van vijf methodologische beperkingen:

Ten eerste beperkt de steekproefgrootte ($n = 11$) de externe validiteit. Ondanks een diverse representatie van rollen is de steekproef niet representatief voor de brede, internationale ASD-gemeenschap, noch werd gecorrigeerd voor culturele of geografische bias. De bevindingen vormen een verkennende validatie, geen universele generalisatie.

Ten tweede bestaat er een risico op variatie in conceptvaliditeit. De semi-gestructureerde interviews lieten ruimte voor uiteenlopende semantische interpretaties van terminologie (bijvoorbeeld de definitie van *time boxing*). Aangezien het raamwerk momenteel uitsluitend getoetst is in geïsoleerde validatiescenario's, is longitudinaal vervolgonderzoek via in-depth *case studies* binnen actieve bedrijfstransformaties aangewezen.

Ten derde berust de praktische constructie van het raamwerk deels op grijze literatuur. Hoewel dit een weloverwogen keuze was om de directe koppeling met de industriële realiteit te waarborgen, introduceert dit onvermijdelijk een risico op selectiebias.

Ten vierde vormt de aanname van functionele inwisselbaarheid, de premisse dat Agile-praktijken met een gelijk impactprofiel equivalent zijn, een bewuste, maar ingrijpende theoretische simplificatie (zie assumptie 1). Dit dimensionele reductionisme loopt het risico de sociaal-psychologische complexiteit van organisatorische verandering te onderschatten. Binnen de complexe realiteit van softwareontwikkeling zijn de adoptie en effectiviteit van een praktijk niet louter afhankelijk van het structurele impactpatroon, maar worden deze sterk gefaciliteerd door menselijke factoren zoals psychologische veiligheid, heersende bedrijfscultuur en teamdynamiek. Zo kunnen een daily stand-up en een rigoureuze, asynchrone rapportagestructuur theoretisch identiek scoren op het raakvlak van *Collaboration* en *Flow*, maar appelleren zij in de praktijk fundamenteel anders aan de intrinsieke motivatie en impliciete kennisdeling van ontwikkelaars. Het CFL-raamwerk mag derhalve niet geïnterpreteerd worden als een deterministisch beslissingsalgoritme dat de bedrijfscontext neutraliseert. Het is veeleer een heuristische lens die de strategische dialoog binnen bedrijven objectiveert, waarna het sociotechnische debat bij de implementatie onontbeerlijk blijft.

Ten slotte wordt de interne validiteit gemedieerd door de persoonlijke expertise van de respondenten. Zoals eerder geobserveerd, induceert een gebrek aan praktische ervaring een sterk theoretische inschatting, wat de positionering van het concept beïnvloedt. Dit resulteert in een risico op beoordelaarsbias.

6 Implicaties

6.1 Onderzoek

De initiële validatie van het CFL-raamwerk vormt een fundament voor diverse academische vervolgotrajecten:

- **Kwantitatieve opschaling:** Grootschalig, geografisch divers vervolgonderzoek over verschillende hiërarchische lagen is noodzakelijk om de statistische robuustheid en positioneringsconsistentie van het raamwerk te verifiëren.
- **Longitudinale casestudies:** Operationele observaties over langere periodes moeten aantonen hoe de implementatie van het raamwerk de *shared understanding* en de kwaliteit van besluitvorming dynamisch beïnvloedt.
- **Causaliteit en orthogonaliteit:** Verder onderzoek dient de onderlinge causale relaties tussen de dimensies te bepalen, om te toetsen of *Flow* fundamenteel fungeert als einddoel of dat de assen zuiver orthogonaal zijn.
- **Domeinextrapolatie:** Aangezien het raamwerk filosofie-agnostisch is, kan de toepasbaarheid van de *Collaboration*-, *Flow*- en *Learning*-dimensies getoetst worden in andere contexten, zoals DevOps, watervalmethodieken of Lean manufacturing.
- **Praktische verankering en maturiteit:** De ontwikkeling van een gekwantificeerde maatstaf met referentiepraktijken als ijkingspunten kan Agile-maturiteit evalueren op basis van doelrealisatie. Aanvullend kan de impact van visuele weergaven (Appendix C) op interne bedrijfsdebatten onderzocht worden.

6.2 Praktijk

Het CFL-raamwerk bevordert structureel de *shared understanding* door als methode-onafhankelijke ver-
taallaag te fungeren voor alle stakeholders.

Agile-coaches kunnen het model inzetten om interventies te verantwoorden door praktijken en con-
textuele problemen te koppelen aan specifieke, ruimtelijk gedefinieerde organisatiedoelen. **Management**
gebruikt het raamwerk vervolgens om strategische beslissingen transparant te communiceren, wat gecen-
traliseerde weerstand vermindert door onderliggende perspectieven te expliciteren. Voor **softwareont-
wikkelaars** verlaagt het abstractieniveau de drempel van specifiek vakjargon, waardoor zij kritisch en
geïnformeerd kunnen deelnemen aan de besluitvorming. Dit faciliteert het benodigde *shared ownership*
en resulteert in een inclusieve, datagedreven dialoog over de meest effectieve stappen naar wendbaarheid.

7 Conclusie

Dit onderzoek werd gedreven door de vraag welke dimensies ASD-praktijken en organisationele doelen
beter gealigneerd kunnen worden.

De centrale bevinding is dat met behulp van het CFL-raamwerk de effectiviteit van problemen en
oplossingen in ASD-contexten kan worden ingeschat. Dit stelt organisaties in staat de impact op hun
doelen beter in kaart te brengen en geïnformeerde, breder gedragen beslissingen te nemen. Het raamwerk
stelt drie perspectieven op impact voor. Deze zijn samengevat in het CFL-raamwerk met de orthogonale
dimensies *Collaboration*, *Flow* en *Learning*.

Het ontwikkelde CFL-raamwerk biedt een antwoord op de groeiende nood aan theoretische structuur
binnen het ASD-domein. Het stelt organisaties in staat om:

- **Impliciete aannames expliciet te maken:** Door problemen en oplossingen ruimtelijk te mappen,
wordt duidelijk waarom bepaalde praktijken wel of niet bijdragen aan specifieke doelen.
- **Methode-onafhankelijkheid te waarborgen:** Het raamwerk overstijgt methodologische hokjes
en jargon, waardoor een inclusieve dialoog tussen technische en niet-technische stakeholders mo-
gelijk wordt.
- **Besluitvorming te structureren:** Het biedt een objectieve basis voor de allocatie van middelen
en het opstellen van meetbare KPI's die verder gaan dan oppervlakkige 'Agile volwassenheid'.

Hoewel de validatie met een selecte groep experts de robuustheid van het raamwerk onderstreept,
dient dit onderzoek als een startpunt voor verdere exploratie. Deze studie concludeert dat de alignering
tussen Agile-praktijken en organisatiedoelen structureel in kaart gebracht kan worden door deze concep-
ten te projecteren op de orthogonale assen *Collaboration*, *Flow* en *Learning*. Dit expliciteert impliciete
aannames en neutraliseert methodologisch jargon, wat resulteert in objectievere, datagedreven besluit-
vorming en biedt een antwoord op de hoofdonderzoeksvraag "Aan de hand van welke dimensies kan
de alignering tussen Agile-praktijken en organisatiedoelen binnen een organisatie structureel in kaart
worden gebracht en geëvalueerd?".

In het kader van verdere exploratie dient toekomstig onderzoek een breder toepassingsgebied, de lange-termijn-impact van de toepassing van het CFL-raamwerk en eventuele onderlinge correlaties tussen de dimensies bloot te leggen. Zo zal de universele kracht van het model robuuster bewezen kunnen worden. In essentie transformeert dit raamwerk Agile van een diverse verzameling aan opvattingen naar een continue, tastbare, strategische dialoog voor continue organisatorische verbetering. Zo ondersteunt het CFL-raamwerk organisaties bij het nemen van ASD-gerelateerde beslissingen.

Referenties

- Alami, A., Krancher, O., & Paasivaara, M. (2022). The journey to technical excellence in agile software development. *Information and Software Technology*, 150, 106959. <https://doi.org/10.1016/j.infsof.2022.106959>
- Ameta, U., Patel, M., & Sharma, A. K. (2022). Scaled Agile Framework Implementation in Organizations', its Shortcomings and an AI Based Solution to Track Team's Performance [Conference Proceedings]. *2022 IEEE 3rd Global Conference for Advancement in Technology (GCAT)*, 1–7. <https://doi.org/10.1109/GCAT55367.2022.9971968>
- Anderson, D. J. (2010). *Kanban, Succesccful Evolutionary Change for Your Technology Business* (1ste ed.). Blue Hole Press.
- Andres, C., & Beck, K. (2005). *Extreme Programming Explained: Embrace Change* (2de ed.). Addison-Wesley. <https://go.exlibris.link/pL3dTW6c>
- Atlassian. (2025). What is Agile?
- Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., & Thomas, D. (2001). *Manifesto for Agile Software Development*. Verkregen juli 20, 2025, van <https://agilemanifesto.org/>
- Berntzen, M., Hoda, R., Moe, N. B., & Stray, V. (2023). A Taxonomy of Inter-Team Coordination Mechanisms in Large-Scale Agile [Place: New York]. *IEEE transactions on software engineering*, 49(2), 699–718. <https://doi.org/10.1109/TSE.2022.3160873>
- Company, T. L. (2020, november). LeSS Framework. <https://less.works/less/framework>
- Cruth, M. (2012). Discover the spotify model. <https://www.atlassian.com/agile/agile-at-scale/spotify>
- Dalkey, N., & Helmer, O. (1963). An Experimental Application of the DELPHI Method to the Use of Experts. *Management Science*, 9(3), 458–467. <https://doi.org/10.1287/mnsc.9.3.458>
- Digital.ai. (2025). State of Agile report.
- Donnelly, R., & Hughes, E. (2023). The HR ecosystem framework: Examining strategic HRM tensions in knowledge-intensive organizations with boundary-crossing professionals [Place: Hoboken, USA]. *Human resource management*, 62(1), 79–95. <https://doi.org/10.1002/hrm.22115>
- Edison, H., Wang, X., & Conboy, K. (2022). Comparing Methods for Large-Scale Agile Software Development: A Systematic Literature Review [Place: New York]. *IEEE transactions on software engineering*, 48(8), 2709–2731. <https://doi.org/10.1109/TSE.2021.3069039>
- Edwards, P. K. (2017). Making 'Critical Performativity' Concrete: Sumantra Ghoshal and Linkages between the Mainstream and the Critical. *British Journal of Management*, 28(4), 731–741. <https://doi.org/10.1111/1467-8551.12227>
- Goldratt, E. M., & Cox, J. (2014). *The Goal: a process of ongoing improvement* (4de ed.). North River Press.
- Goni, J. I. C., & Van Looy, A. (2024). Developing a framework for innovating less-structured business processes: a Delphi study [Place: Berlin/Heidelberg]. *Information systems and e-business management*, 22(2), 385–413. <https://doi.org/10.1007/s10257-024-00680-4>
- Howard, M. C., & Rose, J. C. (2019). Refining and extending task–technology fit theory: Creation of two task–technology fit scales and empirical clarification of the construct. *Information & Management*, 56(6), 103134. <https://doi.org/10.1016/j.im.2018.12.002>

- Idrees, H., Xu, J., & Haider, S. A. (2024). Impact of knowledge management infrastructure and processes on automobile manufacturing firm innovative performance through the mediating role of agile project management practice [Place: Kempston]. *Journal of knowledge management*, 28(10), 3046–3074. <https://doi.org/10.1108/JKM-12-2023-1166>
- Iqbal, J., Omar, M., & Yasin, A. (2019). An Empirical Analysis of the Effect of Agile Teams on Software Productivity [Conference Proceedings]. *2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, 1–8. <https://doi.org/10.1109/ICOMET.2019.8673413>
- Iqbal, M., Ijaz, M., Mazhar, T., Shahzad, T., Abbas, Q., Ghadi, Y., Ahmad, W., & Hamam, H. (2024). Exploring issues of story-based effort estimation in Agile Software Development (ASD). *Science of computer programming*, 236(Journal Article), 103114. <https://doi.org/10.1016/j.scico.2024.103114>
- Jookan, L. (2025). *Mining Human Collaboration Behavior from Data* [Phd thesis]. Universiteit Hasselt. <https://documentserver.uhasselt.be/handle/1942/46282>
- Junior, A. C. P., da Silva, V. R., & Junior, P. T. A. (2024). Critical Success Factors for Agile Software Development. *IEEE transactions on engineering management*, 71(Journal Article), 14807–14823. <https://doi.org/10.1109/TEM.2024.3441829>
- Kanaparan, G., & Strode, D. E. (2025). Investigating the relationship between coordination strategy and coordination effectiveness in agile software development projects. *Information and software technology*, 182(Journal Article), 107708. <https://doi.org/10.1016/j.infsof.2025.107708>
- Karhapaa, P., Behutiye, W., Seppanen, P., Rodriguez, P., Oivo, M., Franch, X., Martinez-Fernandez, S., Lopez, L., Choras, M., Bagnato, A., Aaramaa, S., & Partanen, J. (2024). Evidence-Based Quality-Aware Agile Software Development Process: Design and Evaluation [Place: Piscataway]. *IEEE access*, 12(Journal Article), 86487–86512. <https://doi.org/10.1109/ACCESS.2024.3414614>
- Kaur, V. (2023). Knowledge-based dynamic capabilities: a scientometric analysis of marriage between knowledge management and dynamic capabilities [Place: Kempston]. *Journal of knowledge management*, 27(4), 919–952. <https://doi.org/10.1108/JKM-02-2022-0112>
- Keshavarz, H., Nematigavzan, M., Jalaliyoon, N., Nouri, M., & Keshavarz, H. (2024). Phenomenon of Fake Agile in Software Development (Hassan Keshavarz, Red.) [Place: Washington]. *IT professional*, 26(5), 85–91. <https://doi.org/10.1109/MITP.2024.3457348>
- Kruger, J., & Dunning, D. (1999). Unskilled and unaware of it: how difficulties in recognizing one's own incompetence lead to inflated self-assessments. *J Pers Soc Psychol.* <https://doi.org/10.1037/0022-3514.77.6.1121>
- Kuhrmann, M., Tell, P., Hebig, R., Klunder, J., Munch, J., Linssen, O., Pfahl, D., Felderer, M., Prause, C. R., MacDonell, S. G., Nakatumba-Nabende, J., Raffo, D., Beecham, S., Tuzun, E., Lopez, G., Paez, N., Fontdevila, D., Licorish, S. A., Kupper, S., ... Richardson, I. (2022). What Makes Agile Software Development Agile? [Place: New York]. *IEEE transactions on software engineering*, 48(9), 3523–3539. <https://doi.org/10.1109/TSE.2021.3099532>
- Kula, E., Greuter, E., van Deursen, A., & Gousios, G. (2022). Factors Affecting On-Time Delivery in Large-Scale Agile Software Development [Place: New York]. *IEEE transactions on software engineering*, 48(9), 3573–3592. <https://doi.org/10.1109/TSE.2021.3101192>
- Lee, W.-T., & Chen, C.-H. (2023). Agile Software Development and Reuse Approach with Scrum and Software Product Line Engineering [Place: Basel]. *Electronics (Basel)*, 12(15), 3291. <https://doi.org/10.3390/electronics12153291>
- Little, J. D. C. (1961). A Proof for the Queuing Formula: $L = W$. *Operations research*, 9(3), 383–387. <https://doi.org/10.1287/opre.9.3.383>
- Mariana da Silva Barbosa Gama & Bonamigo, A. (2024). Sustainable lean manufacturing as long-term strategy: performance framework development and prioritization [Place: Bingley]. *Journal of strategy and management*, 17(2), 205–221. <https://doi.org/10.1108/JSMA-05-2023-0104>

- Martinez-Sanchez, A., & Vicente-Oliva, S. (2023). Supporting agile innovation and knowledge by managing human resource flexibility [Place: Brentwood]. *International journal of innovation science*, 15(3), 558–578. <https://doi.org/10.1108/IJIS-11-2021-0200>
- Meckenstock, J.-N. (2024). Shedding light on the dark side – A systematic literature review of the issues in agile software development methodology use. *The Journal of systems and software*, 211(Journal Article), 111966. <https://doi.org/10.1016/j.jss.2024.111966>
- Microsoft. (2022). wat is DevOps? <https://azure.microsoft.com/nl-nl/resources/cloud-computing-dictionary/what-is-devops>
- Modic, N., & Åhlström, P. (2012). *This is Lean* (1ste ed.). Rheologica Publishing.
- Nasution, M. D. T. P., Sari, P. B., Aspan, H., & Rossanty, Y. (2023). How do social media-facilitated crowdsourcing and knowledge integration affect new product development? SME agile initiatives [Place: Abingdon]. *Cogent business & management*, 10(3), 1–26. <https://doi.org/10.1080/23311975.2023.2265093>
- OConnor, T. E. (z.d.). The Large-Scale Scrum (LeSS) framework, How LeSS applies the principles and ideals of scrum to the enterprise. <https://www.atlassian.com/agile/agile-at-scale/less>
- Ouriques, R., Wnuk, K., Gorschek, T., & Svensson, R. B. (2023). The role of knowledge-based resources in Agile Software Development contexts. *The Journal of systems and software*, 197(Journal Article), 111572. <https://doi.org/10.1016/j.jss.2022.111572>
- Palopak, Y., & Huang, S.-J. (2024). Perceived impact of agile principles: Insights from a survey-based study on agile software development project success. *Information and software technology*, 176(Journal Article), 107552. <https://doi.org/10.1016/j.infsof.2024.107552>
- Paré, G., Cameron, A.-F., Poba-Nzaou, P., & Templier, M. (2013). A systematic assessment of rigor in information systems ranking-type Delphi studies. *Information & Management*, 50(5), 207–217. <https://doi.org/10.1016/j.im.2013.03.003>
- Pereira, D. G., Afonso, A., & Medeiros, F. M. (2015). Overview of Friedman’s Test and Post-hoc Analysis. *Communications in statistics. Simulation and computation*, 44(10), 2636–2653. <https://doi.org/10.1080/03610918.2014.931971>
- Porter, M. E. (1979). How Competitive Forces Shape Strategy [Place: Boston]. *Harvard business review*, 57(2), 137. <https://go.exlibris.link/yJ0GCSrg>
- Psychogios, A. (2024). Re-conceptualising Total Quality Leadership: a framework development and future research agenda [Place: Bingley]. *TQM journal*, 36(3), 743–761. <https://doi.org/10.1108/TQM-01-2022-0030>
- Ries, E. (2011). *The Lean Startup* (1ste ed.). Crown Business New York.
- Rivera Ibarra, J. G., Borrego, G., & Palacio, R. R. (2024). Early Estimation in Agile Software Development Projects: A Systematic Mapping Study [Place: Basel]. *Informatics (Basel)*, 11(4), 81. <https://doi.org/10.3390/informatics11040081>
- Rolland, K. H., Fitzgerald, B., Dingsøyr, T., & Stol, K.-J. (2023). Acrobats and Safety Nets: Problematizing Large-Scale Agile Software Development [Place: New York, NY]. *ACM transactions on software engineering and methodology*, 33(2), 1–45. <https://doi.org/10.1145/3617169>
- Saeeda, H., Ahmad, M. O., & Gustavsson, T. (2023). Challenges in Large-Scale Agile Software Development Projects. *Applied computing review : a publication of the Special Interest Group on Applied Computing*, 23(2), 23. <https://doi.org/10.1145/3555776.3577662>
- Sathe, C. A., & Panse, C. (2023). An Empirical Study on Impact of Project Management Constraints in Agile Software Development: Multigroup Analysis between Scrum and Kanban. *Brazilian journal of operations & production management*, 20(3), 1796. <https://doi.org/10.14488/BJOPM.1796.2023>
- Sathe, C. A., & Panse, C. (2024). Modelling the enablers of productivity of enterprise-level Agile software development process using modified TISM approach [Place: Bradford]. *Journal of modelling in management*, 19(6), 1749–1772. <https://doi.org/10.1108/JM2-12-2023-0285>
- Schmiedel, T., vom Brocke, J., & Recker, J. (2013). Which cultural values matter to business process management?: Results from a global Delphi study. *Business Process Management Journal*, 19(2), 292–317. <https://doi.org/10.1108/14637151311308321>

- Scrum.org. (z.d.). What is Scrum? <https://www.scrum.org/learning-series/what-is-scrum/>
- Singh, M., Chauhan, N., & Popli, R. (2025). Test case reduction and SWOA optimization for distributed agile software development using regression testing [Place: New York]. *Multimedia tools and applications*, 84(10), 7065–7090. <https://doi.org/10.1007/s11042-024-19148-1>
- Stray, V., Hoda, R., Paasivaara, M., Lenarduzzi, V., & Mendez, D. (2022). Theories in Agile Software Development: Past, Present, and Future Introduction to the XP 2020 Special Section. *Information and software technology*, 152(Journal Article), 107058. <https://doi.org/10.1016/j.infsof.2022.107058>
- Studio, S. (2023, maart). SFAe Big Picture. <https://framework.scaledagile.com/>
- Sutherland, J., & Schwaber, K. (2012). Appendix 2: The Scrum Guide. In *Software in 30 Days*. Wiley. <https://go.exlibris.link/d5vqZlWw>
- Symon, G., Cassell, C., & Johnson, P. (2018). Evaluative Practices in Qualitative Management Research: A Critical Review. *International Journal of Management Reviews*, 20(1), 134–154. <https://doi.org/10.1111/ijmr.12120>
- Taleb, N. N., & Douady, R. (2013). Mathematical definition, mapping, and detection of (anti)fragility. *Quantitative Finance*, 13(11), 1677–1689. <https://doi.org/10.1080/14697688.2013.800219>
- Ven, A. H. V. D., & Delbecq, A. L. (1974). The Effectiveness of Nominal, Delphi, and Interacting Group Decision Making Processes. *Academy of Management Journal*, 17(4), 605–621. <https://doi.org/10.5465/255641>
- Wysocki, R. K. (2019). *Effective project management: traditional, agile, extreme, hybrid* (Eighth;8;8th;). John Wiley & Sons, Inc. <https://go.exlibris.link/Z3b3km9v>
- Yigitbas, E., Witalinski, I., Gottschalk, S., & Engels, G. (2024). Virtual Reality Collaboration Platform for Agile Software Development. In R. Kadgien, A. Jedlitschka, A. Janes, V. Lenarduzzi & X. Li (Red.), *Product-Focused Software Process Improvement* (pp. 3–19). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-49266-2_1

A Dankwoord

Ik wil mijn promotor en begeleider Frank Vanhoenshoven bedanken voor de vlotte samenwerking en de duidelijke feedback. Dit stelde me in staat om op een iteratieve manier te werk te gaan en tot een beter eindresultaat te komen. Daarnaast ben ik Patrick Steyaert en Wim Bollen zeer dankbaar voor hun ondersteuning bij het ontwikkelen van het kernidee voor deze masterproef. Met hun waardevolle praktijkervaring en leergierige ingesteldheid hebben ze me vanaf het prille begin geholpen bij het vertalen van de theorie naar de praktijk. Ook wil ik Arn Bollen en Vicky Croymans bedanken voor de ondersteuning bij respectievelijk grafische en taalkundige vragen die ik had. Tot slot wil ik alle respondenten bedanken voor hun tijd, de gedeelde inzichten en hun bereidheid om deel te nemen aan de interviews tijdens de validatiestap.

B Resultaten tabellen

Het overzicht van de tabellen met beschrijvende statistieken uit de resultatensectie.

Tabel 5: Gestandaardiseerde gemiddeldes per Agile-praktijk met vier of meer respondenten

Praktijk	<i>Flow</i>	<i>Collaboration</i>	<i>Learning</i>
Daily stand-up	0.158	0.333	0.169
Time boxing	0.319	0.251	0.264
Korte feedbackcycli	0.150	0.276	0.386
Pair programming	0.231	0.336	0.340

Tabel 6: Gestandaardiseerde standaardafwijkingen per Agile-praktijk met vier of meer respondenten

Praktijk	Flow	Collaboration	Learning
Daily stand-up	0.110	0.145	0.138
Time boxing	0.140	0.120	0.242
Korte feedbackcycli	0.123	0.154	0.049
Pair programming	0.149	0.056	0.084

Tabel 7: Gestandaardiseerde gemiddelden per perspectief van problemen en situaties met vier of meer respondenten

Probleem	Flow	Collaboration	Learning
Needs-requirements mismatch	0.342	0.300	0.241
Collaboration across time zones	0.190	0.332	0.261
Researching new knowledge	0.107	0.203	0.379
Key person dependencies	0.226	0.227	0.373
Multi teams hebben elk verschillende doelen en prioriteiten	0.264	0.385	0.318

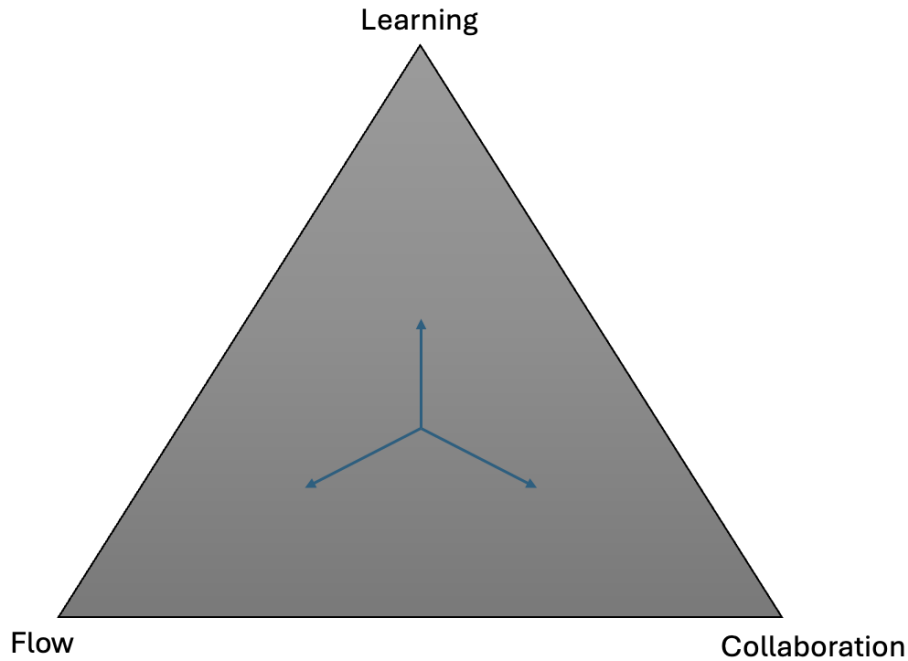
Tabel 8: Gestandaardiseerde standaardafwijkingen per perspectief van problemen en situaties met vier of meer respondenten

Probleem	Flow	Collaboration	Learning
Needs-requirements mismatch	0.121	0.048	0.126
Collaboration across time zones	0.123	0.051	0.120
Researching new knowledge	0.097	0.131	0.082
Key person dependencies	0.149	0.141	0.071
Multi-teams hebben elk verschillende doelen en prioriteiten	0.083	0.036	0.128

C Alternatieve weergave van het CFL-raamwerk

De voorstelling van het *Collaboration, Flow, Learning* raamwerk, gepresenteerd in de resultaten sectie met figuur 4, is de meest intuïtieve na het opstellen van de drie dimensies. Hier wordt zoals besproken elke dimensie op een orthogonale as geplaatst en op die manier een drie dimensionele ruimte gecreëerd. In deze ruimte kunnen praktijken en problemen geplaatst worden om zo beslissingen te faciliteren.

De vraag of de driedimensionale voorstelling de beste manier is om dit doel te bereiken is met dit onderzoek niet gevalideerd. Daarom worden hieronder nog twee alternatieve visualisaties voorgesteld en hun respectievelijke potentiële impact besproken. Eerst volgt de driehoeksvoorstelling, een visualisatie met de focus op het faciliteren van discussie. Deze wordt gevolgd door de oppervlaktevoorstelling, een versie die een kwantitatief potentieel sterker in de verf zet.

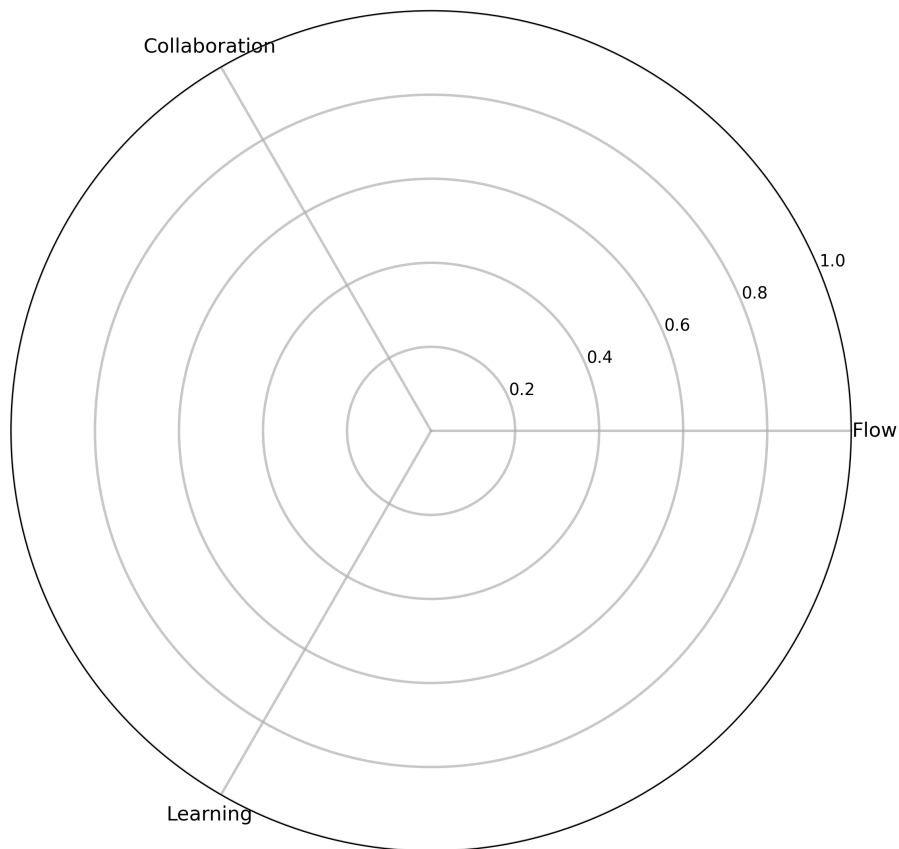


Figuur 5: Driehoek, Alternatieve visualisatie

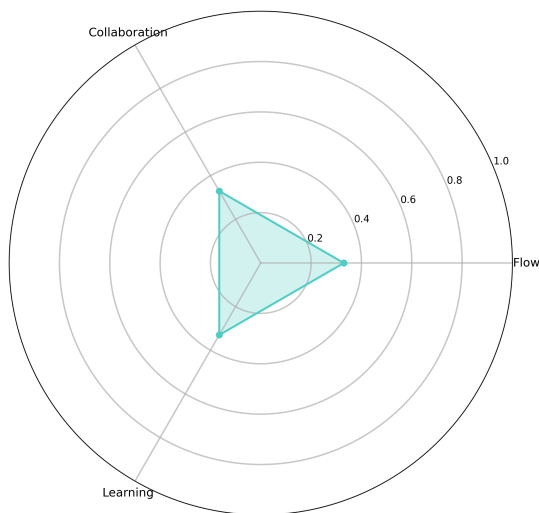
C.1 Driehoeksvoorstelling

De eerste alternatieve weergave is de driehoeksvoorstelling te zien in figuur 5. Dit alternatief is ten eerste gemakkelijker voor te stellen in een twee-dimensionale weergave dan de projectie van de drie-dimensionale ruimte. Verder dwingt het gebruikers tot het maken van keuzes bij het situeren van praktijken. Met andere woorden, het perspectief van een praktijk kan niet groter worden in een bepaalde richting zonder de situering ten opzichte van de andere hoekpunten te veranderen. Dit betekent dat er niet langer naar het raamwerk gekeken kan worden als een absolute rangorde van praktijken en problemen. Deze restrictie stelt gebruikers in staat te focussen op het creëren van de *shared understanding* van een praktijk in een context en niet op het bepalen van een absolute waarde van een concept in het algemeen.

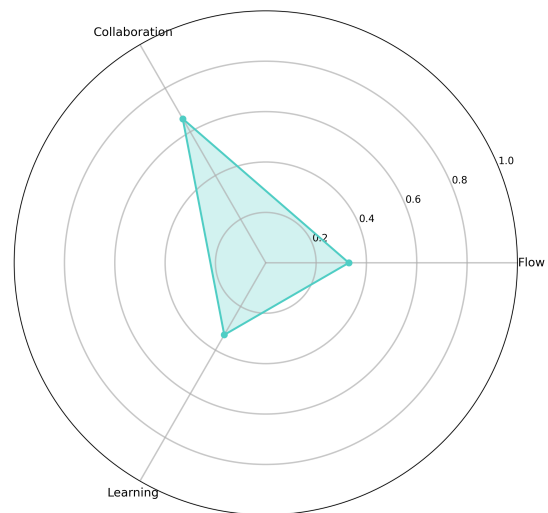
Deze voorstelling helpt gebruikers keuzes te maken en kan het diffuus antwoorden, beschreven in de subsectie “Consensus”, tegengaan. Organisaties die deze tool hanteren zullen gebruik kunnen maken van een wisselwerking tussen de tool en de *shared understanding* binnen de organisatie om zo doelen, initiatieven en beslissingen gericht en collectiever te nemen of faciliteren.



(a) Radar-vorm, Alternatieve visualisatie



(b) Radar-vorm voorbeeld methode 1, Alternatieve visualisatie



(c) Radar-vorm voorbeeld methode 2, Alternatieve visualisatie

Figuur 6: Overzicht radar-vorm, Alternatieve visualisatie

C.2 Oppervlaktevoorstelling

Een tweede alternatieve weergave is de oppervlaktevoorstelling in figuur 6a en lijkt op het eerste zicht zeer gelijkaardig aan de eerdere alternatieve weergave. Deze weergave is wederom gemakkelijker voor te stellen in twee dimensies en door het plaatsen van een waarde per dimensie wordt een driehoek gecreëerd. Het plaatsen van de waardes per dimensie in deze voorstelling creëert een driehoekige oppervlakte die

geïnterpreteerd kan worden om een ranking tussen praktijken te bekomen. Zo zijn in figuren 6b en 6c twee methodes afgebeeld, de eerste met een gelijke score op elke dimensie en de tweede met een sterkere focus op *Collaboration*.

Deze voorstelling kan de grondslag zijn voor een meer kwantitatieve kijk op het probleem. Het geeft een aanzet tot een gestructureerde methode om praktijken te rangschikken en zo te beoordelen. Op die manier verschuift de focus van het creëren van *shared understanding* naar het vaststellen van een soort score voor de efficiëntie van praktijken.

D Overzicht van geanalyseerde elementen

In deze appendix worden de elementen opgelijst die geëxtraheerd zijn uit de 16 praktische bronnen. Deze elementen vormen de basis voor de mapping naar de negen impacttypes en de uiteindelijke positionering binnen de dimensies *Flow*, *Collaboration* en *Learning*. Verder worden de 55 gevonden impacttypes opgelijst.

Tabel 9: Oplijsting van geïdentificeerde praktijken en KPI's uit Agile-methodieken

Praktijken & Rollen	Kritieke Prestatie-indicatoren (KPI's)
<i>Spotify Model</i>	
Squads Tribes Chapters Autonomy (Zelfsturing)	
<i>DevOps</i>	
CI/CD (Continuous Integration/Delivery) Test Driven Development (TDD) Agile Project Management Automatisering Pipeline monitoren Waarneembaarheid	Throughput Text coverage CPU-usage Schijfgebruik Netwerkconnectiviteit
<i>LeSS</i>	
Sprint Planning 1 (strategisch) Sprint Planning 2 (tactisch) Sprint Planning (operationeel)	Throughput Leadtime Story Points
Daily Scrum Sprint Review Sprint Retrospective Scrum roles Product Backlog Management	Completion Rate Stakeholder Prioritization Burn-through Rate
Time-boxing (Sprints) Overall Retrospective (LeSS)	Throughput Completion Rate Rework Rate

Praktijken & Rollen	Kritieke Prestatie-indicatoren (KPI's)
<i>SAFe</i>	
Agile Release Train (ART)	(Team-)throughput (Team-)lead time
Continuous Delivery Pipeline	
PI Planning	
Team PI Goals	
Inspect & Adapt (SAFe)	Split Metrics
Continuous Delivery	Klanttevredenheid
Value Stream Mapping	Flow efficiency
	Quality Indicators
SAFe Roles	
<i>Scrum</i>	
Time-boxing (2-4 week Sprints)	Completion Rate
Sprint Planning	Throughput
	Leadtime
	Storypoints
Sprint Goal	
Sprint Review	Completion Rate
Sprint Retrospective	
Scrum Roles	
Product Backlog	Throughput
	Burnthrough Rate
Definition of Done	
<i>Lean</i>	
Manage WIP	Flow Efficiency
	Resource Efficiency
Flow Efficiency Management	Internal SLA Compliance
Reduce Waste	Rework Rate
Continuous Improvement (Kaizen)	
Respect for People	
<i>Lean Startup</i>	
Validated Learning	Accepted or rejected hypotheses
Build-Measure-Learn	
Innovation Accounting	
Choose Actionable Metrics	Split Metrics
Use Kanban (pull) System	WIP
	Leadtime
<i>Theory of Constraints</i>	
Identify-Exploit-Subordinate-Elevate-Evaluate	Capacity
<i>Kanban</i>	
WIP Limiet	WIP
Pull Systems	Flow Efficiency
Internal SLA	Due Date Performance
	Flow Efficiency
Prioritization based on cadence	Throughput
Continuous Assessment	Blocked Work

Praktijken & Rollen	Kritieke Prestatie-indicatoren (KPI's)
Continuous Improvement Visualize Workflow Daily Meeting	Leadtime Due Date Performance Throughput
<i>eXtreme Programming</i>	
Sit Together Pair Programming Stories Cross-functional Teams Weekly Planning Quarterly Goals Slack Automated 10min Build Continuous Integration Test Driven development Incremental Design Customer Involvement Pay Per Use	# Of Handovers Between Teams Total Story Size in Scope Lifecycle of Branches Revenue Δ Revenue

Tabel 10: Overzicht Impactpatronen

Nr.	Voorspelbaarheid	Product Kwaliteit	Proces Kwaliteit	Transparantie WIP (teams)	Transparantie WIP (in teams)	Symbiose	Waarde versus niet-waarde creërend	Kennisdeling	Kenniswinning	ID
1	0	0	0	0	0	0	0	0	1	a
2	0	0	0	0	0	0	0	1	1	b
3	0	0	0	0	0	1	0	0	1	c
4	0	0	0	0	0	1	0	1	0	d
5	0	0	0	0	1	1	0	1	0	e
6	0	0	0	1	0	0	0	1	0	f
7	0	0	0	1	0	0	0	1	1	g
8	0	0	0	1	1	0	0	1	0	h
9	0	0	0	1	1	1	0	1	0	i
10	0	0	1	0	0	0	0	0	1	j
11	0	0	1	0	0	1	0	1	0	k
12	0	0	1	0	0	1	0	1	1	l
13	0	0	1	0	1	0	0	0	1	m
14	0	0	1	1	1	0	0	0	0	n

Nr.	V	PK	PR	TW	TI	S	W	KD	KW	ID
15	0	1	0	0	0	0	0	0	1	o
16	0	1	0	0	1	1	0	1	1	p
17	0	1	0	1	0	0	0	0	1	q
18	0	1	1	0	0	0	0	1	0	r
19	0	1	1	0	0	0	0	1	1	s
20	0	1	1	0	0	1	0	1	1	t
21	0	0	0	0	0	0	1	0	1	u
22	0	0	0	0	1	0	1	0	0	v
23	0	0	0	0	1	1	1	1	0	w
24	0	0	0	1	0	0	1	1	1	x
25	0	0	0	1	0	1	1	1	0	y
26	0	0	0	1	1	1	1	0	0	z
27	0	0	0	1	1	1	1	1	0	aa
28	0	0	1	0	1	0	1	1	0	ab
29	0	0	1	0	1	1	1	1	0	ac
30	0	1	0	0	0	0	1	0	1	ad
31	0	1	0	0	0	1	1	1	1	ae
32	0	1	0	0	0	1	1	1	1	af
33	0	1	0	1	0	0	1	1	0	ag
34	0	1	0	1	0	0	1	1	1	ah
35	1	0	0	0	0	0	0	0	1	ai
36	1	0	0	0	0	1	0	1	1	aj
37	1	0	0	0	1	0	0	0	0	ak
38	1	0	0	0	1	0	0	1	0	al
39	1	0	0	0	1	1	0	1	0	am
40	1	0	0	1	0	0	0	0	0	an
41	1	0	1	0	1	0	0	0	0	ao
42	1	0	1	1	0	0	0	0	0	ap
43	1	1	0	0	0	0	0	0	0	aq
44	1	1	1	0	0	1	0	1	0	ar
45	1	1	1	1	0	1	0	1	0	as
46	1	0	0	0	0	0	1	0	0	at
47	1	0	0	0	0	0	1	1	0	au
48	1	0	0	0	1	0	1	0	0	av
49	1	0	0	0	1	0	1	1	0	aw
50	1	0	1	1	0	0	1	0	0	ax
51	1	1	0	0	0	0	1	0	1	ay
52	1	1	0	0	1	1	1	1	1	az
53	1	1	0	1	1	1	1	0	0	ba
54	1	1	1	0	0	0	1	0	0	bb
55	1	1	1	1	0	1	1	1	0	bc

E Overzicht van beantwoorde concepten

Deze appendix geeft een overzicht van hoeveel respondenten vragen met betrekking tot specifieke concepten beantwoordden. De concepten worden opgesplitst in twee categorieën: (i) problemen/situaties/-doelen en (ii) praktijken. Voor elk concept wordt het totaal aantal beantwoorde vragen weergegeven.

Tabel 11: Frequentie van beantwoorde vragen per probleem/situatie/doel

Problemen/situaties/doelen	Aantal antwoorden
Needs–requirements mismatch	9
Collaboration across time zones	8
Researching new knowledge (e.g. nieuwe technologie)	6
Key person dependencies	5
Meerdere teams met verschillende doelen en prioriteiten	4
Process bottlenecks	2
Hiërarchische organisaties	1
Afhankelijkheid van externe platformen	1
Fixed price–fixed scope	1
Gebrek aan gedeelde verantwoordelijkheid	1
Old-school micromanagement	1

Tabel 12: Frequentie van beantwoorde vragen per praktijk

Praktijk	Aantal antwoorden
Timeboxing	11
Daily stand-up	10
Feedback cycles	8
Pair programming	6
Aligneren over afdelingen heen	1
Combineren van push- en pull-principes	1
Value slicing	1
OKR's (Objectives and Key Results)	1
Pull-system	1
Op voorhand plannen hoe om te gaan met updates	1
Negotiate fixed → variable	1
Hybrid teams	1
Community of practice	1
PaceMaker team	1

F Interview structuur

Deze bijlage beschrijft de structuur van het semi-gestructureerde interview dat werd afgenomen voor dit onderzoek.

F.1 Kennismaking

- Wat is uw hoofdbezigheid?
- In welke bedrijven vervult u deze rol en voor hoelang?
- Wat is uw gevoel met kenniswerk, Agile en Agile-initiatieven?

F.2 Bespreking en testen van raamwerk

Na een korte toelichting van het raamwerk werden de respondenten gevraagd de volgende oefeningen uit te voeren:

F.2.1 Oefening 1: Situeren van begrippen

De respondent werd gevraagd elk van de volgende begrippen te situeren in het raamwerk (zie afbeelding 4) met een score op iedere dimensie:

- Daily standup
- Process bottlenecks
- Timeboxing
- Feedbackcycles
- Pairprogramming
- Samenwerken in meerdere tijdzones
- Een voorbeeld uit hun context uit de kennismaking

F.2.2 Oefening 2: Evalueren van oplossingen

De respondent werd gevraagd op basis van het raamwerk te bepalen of onderstaande praktijken een efficiënte match kunnen zijn om de gestelde situatie te verbeteren of helpen aan de hand van twee figuren 4 naast elkaar, één voor de situatie en één voor de praktijk:

- Mismatch vraag-requirements → korte feedbackcycli
- Geografisch verspreid samenwerken → korte feedbackcycli
- Kennis ontginnen over nieuwe technologie (nieuwe tech toepassen) → timeboxing
- Process bottlenecks → pull-systeem
- Kennis bottlenecks → timeboxing
- Kennis bottlenecks → pairprogramming
- Een voorbeeld uit hun context uit de kennismaking

F.3 Feedback

Aan het einde van het interview werden de volgende vragen gesteld:

- Ziet u andere testbare toepassingen van het raamwerk?
- Hoe zou u het interview scoren in termen van leerzaamheid (1-7)?
- Andere feedback?

G Appendix: Validatie visualisaties

In deze appendix worden visualisaties gepresenteerd van de kwantitatieve data uit de validatiestap. Deze visualisaties dienen ter ondersteuning van de genormaliseerde gemiddeldes en standaardafwijkingen zoals gerapporteerd in de hoofdttekst. De data is onderverdeeld in de categorieën 'Problemen/Contexten' en 'Praktijken'.

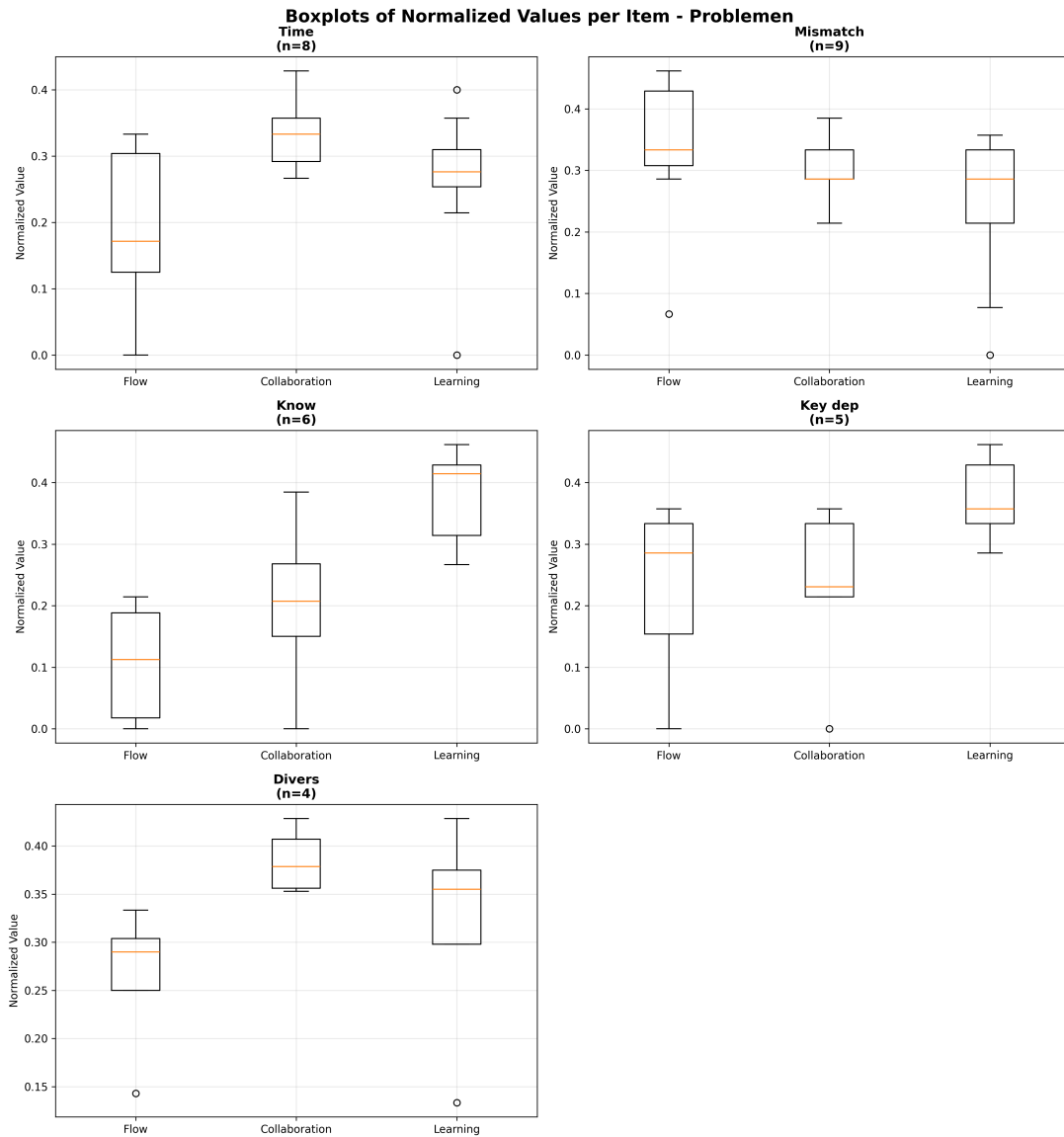
Om de leesbaarheid van de grafieken te waarborgen, zijn enkele conceptnamen afgekort. De onderstaande tabel geeft de volledige namen weer voor de gebruikte afkortingen:

Afkorting in visualisatie	Volledige naam
Divers	multi teams hebben elk verschillende doelen en prioriteiten
Know	researching new knowledge
Time	collab across time zones
Mismatch	needs-requirements mismatch
Key dep	key person dependencies

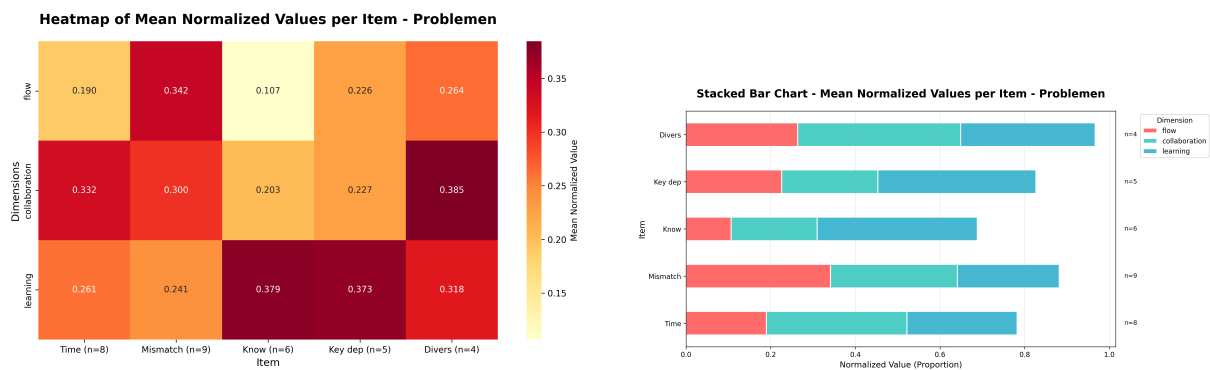
Tabel 13: Verklaring van afkortingen in de visualisaties.

G.1 Problemen en contexten in Agile Software Development

De figuren 7 tot en met 9 tonen de verdeling en gemiddeldes van de scores voor de meest besproken problemen en contexten onder de experts.



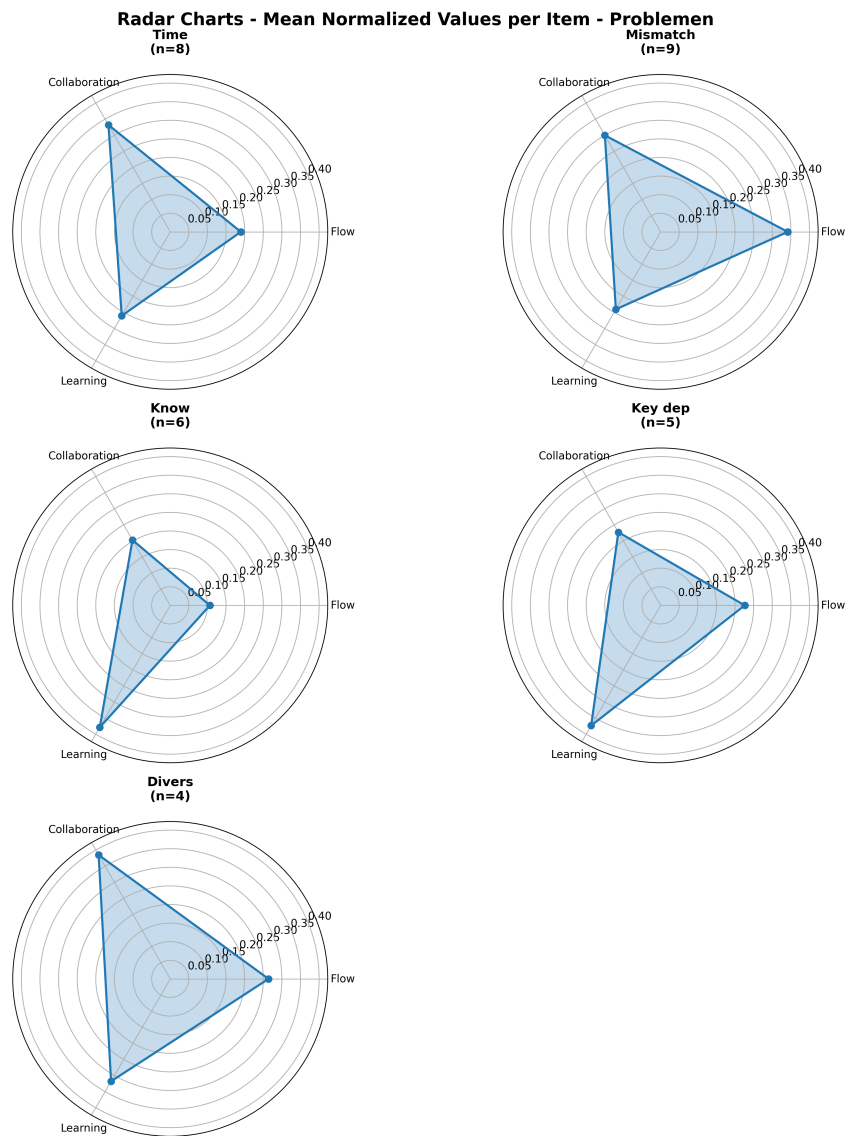
Figuur 7: Boxplots van genormaliseerde waarden per item (Problemen/Contexten)



(a) Heatmap van gemiddelde waarden

(b) Gestapeld staafdiagram (proporties)

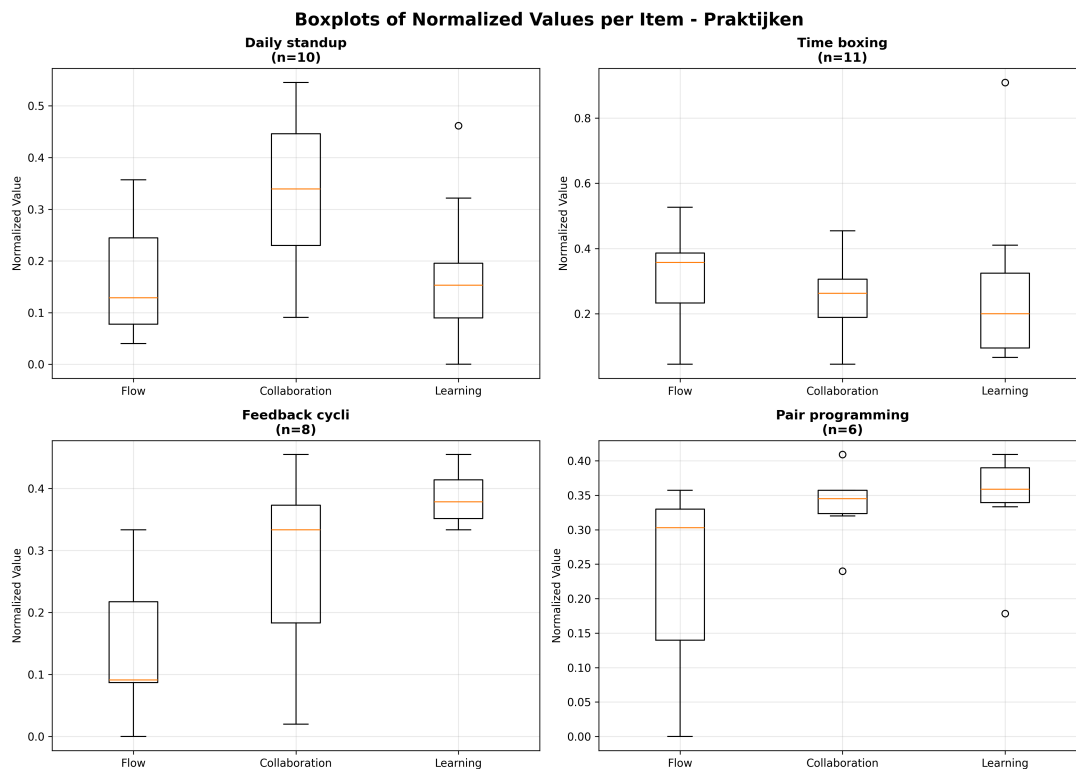
Figuur 8: Vergelijking van gemiddelde dimensie-scores voor ASD-problemen/contexten.



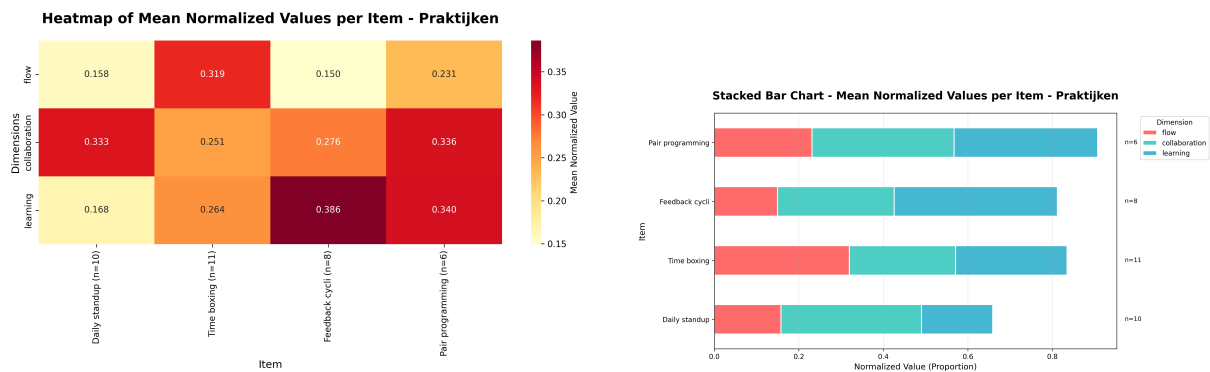
Figuur 9: Radar charts van de dimensie-profielen per probleem.

G.2 Praktijken in Agile Software Development

De figuren 10 tot en met 12 vatten de expertbeoordelingen samen voor de onderzochte ASD-praktijken.



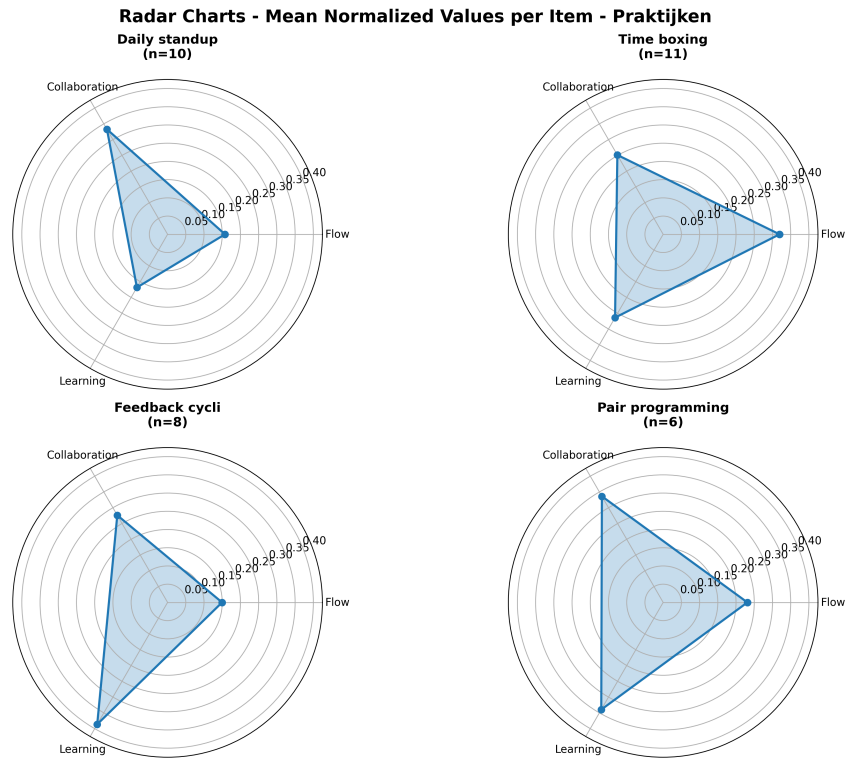
Figuur 10: Boxplots van genormaliseerde waarden per item (Praktijken)



(a) Heatmap van gemiddelde waarden

(b) Gestapeld staafdiagram (proporties)

Figuur 11: Vergelijking van gemiddelde dimensie-scores voor ASD-praktijken.



Figuur 12: Radar charts van de dimensie-profielen per praktijk.

Noot: In alle bovenstaande figuren staat n voor het aantal respondenten per specifiek item.

H Data analyse script

Deze bijlage bevat de kernlogica van het Python-script dat gebruikt werd voor de normalisatie van de interviewdata in dit onderzoek. Het script verwerkt data van zowel de *Praktijken* als de *Problemen* datasets en normaliseert de waarden per respondent op basis van hun maximale rijksom.

H.1 Python implementatie

Het volgende script toont de kernfunctionaliteit voor dataverwerking en normalisatie. Visualisatie- en outputfuncties zijn weggelaten voor de beknoptheid.

```
1 """
2 Beschrijving: Normalisatie, statistische verwerking en hypothesetoetsing
3               (Friedman/Wilcoxon) van interviewdata.
4 """
5
6 import pandas as pd
7 import matplotlib.pyplot as plt
8 import numpy as np
9 import seaborn as sns
10 import scipy.stats as stats
11
12 # --- CONFIGURATIE ---
13
14 # Coderingsschema voor kwalitatieve labels in de 'Problemen' dataset
15 CODING_SCHEME = {
16     'extreem laag': 0,
17     'heel laag': 1,
18     'laag': 2,
19     '-': 3,
20     'hoog': 4,
21     'heel hoog': 5,
22     'extreem hoog': 6
23 }
24
25 # --- DATA VERWERKING FUNCTIES ---
26
27 def process_data(df, data_type):
28     """
29     Codeert tekstuele waarden indien nodig en normaliseert scores op basis van
30     de maximale rij-som per respondent.
31     """
32     numeric_cols = ['flow', 'collaboration', 'learning']
33     df_processed = df.copy()
34
35     # Encodeer tekstwaarden naar numerieke scores voor de Problemen-tab
36     if data_type == 'Problemen':
37         for col in numeric_cols:
38             df_processed[col] = df_processed[col].map(CODING_SCHEME)
39
40     # Bereken de som per rij voor normalisatie
41     df_processed['row_sum'] = df_processed[numeric_cols].sum(axis=1)
42
43     # Identificeer de maximale rij-som per respondent voor relatieve schaling
44     max_row_sum_per_resp = df_processed.groupby('respondent')['row_sum'].
45     transform('max')
46
47     # Normalisatie-stap
48     df_normalized = df_processed.copy()
49     df_normalized[numeric_cols] = df_processed[numeric_cols].div(
50     max_row_sum_per_resp, axis=0)
51
52     return df_normalized
53
54 def calculate_statistics(df_normalized, data_type):
55     """
56     Genereert beschrijvende statistieken (gemiddelde, mediaan, standaarddeviatie
57     )
58     geaggregeerd per item.
59     """
```

```

57 numeric_cols = ['flow', 'collaboration', 'learning']
58 items = df_normalized['item'].unique()
59 stats_summary = []
60
61 print("-" * 30)
62 print(f"DESCRIPTIEVE STATISTIEK: {data_type}")
63 print("-" * 30)
64
65 for item in items:
66     item_data = df_normalized[df_normalized['item'] == item][numeric_cols]
67     n_samples = len(item_data)
68
69     for col in numeric_cols:
70         mean_val = item_data[col].mean()
71         median_val = item_data[col].median()
72         std_val = item_data[col].std()
73
74         stats_summary.append({
75             'data_type': data_type,
76             'item': item,
77             'sample_size': n_samples,
78             'metric': col,
79             'mean': mean_val,
80             'median': median_val,
81             'std': std_val
82         })
83
84     return pd.DataFrame(stats_summary), items
85
86 # --- STATISTISCHE ANALYSE FUNCTIES ---
87
88 def calculate_scenario1_tests(df_normalized, items, data_type):
89     """
90     Voert de Friedman-toets uit voor globale verschillen en Wilcoxon
91     signed-rank toetsen met Bonferroni-correctie voor post-hoc vergelijkingen.
92     """
93     results = []
94
95     for item in items:
96         data = df_normalized[df_normalized['item'] == item]
97         n = len(data)
98
99         # 1. Friedman-toets (non-parametrische ANOVA voor herhaalde metingen)
100         try:
101             _, p_friedman = stats.friedmanchisquare(data['flow'],
102                                                     data['collaboration'],
103                                                     data['learning'])
104         except ValueError:
105             p_friedman = np.nan
106
107         # 2. Wilcoxon signed-rank toetsen (Post-hoc met Bonferroni correctie: n
108         =3)
109         pairs = [('flow', 'collaboration'), ('flow', 'learning'), ('
110 collaboration', 'learning')]
111         p_corrected = {}
112
113         for col1, col2 in pairs:
114             try:
115                 _, p = stats.wilcoxon(data[col1], data[col2])
116                 p_corrected[f"{col1}_{col2}"] = min(1.0, p * 3)
117             except ValueError:
118                 p_corrected[f"{col1}_{col2}"] = np.nan
119
120         # 3. Interpretatie op diverse alfa-niveaus (0.05 tot 0.25)
121         alphas = [0.05, 0.10, 0.15, 0.25]

```

```

120     interpretations = {}
121
122     for alpha in alphas:
123         if pd.isna(p_friedman) or p_friedman > alpha:
124             msg = "Geen significant verschil"
125         else:
126             diffs = []
127             if p_corrected.get('flow_collaboration', 1) <= alpha: diffs.
append("F-C")
128             if p_corrected.get('flow_learning', 1) <= alpha: diffs.append("F
-L")
129             if p_corrected.get('collaboration_learning', 1) <= alpha: diffs.
append("C-L")
130
131             msg = "Verschil: " + " & ".join(diffs) if diffs else "Algemeen
diff, post-hoc niet sig."
132
133             interpretations[f'Alpha_{int(alpha*100)}'] = msg
134
135             results.append({
136                 'Data_Type': data_type,
137                 'Item': item,
138                 'N': n,
139                 'P_Friedman': p_friedman,
140                 'P_Corr_Flow_Collab': p_corrected.get('flow_collaboration'),
141                 'P_Corr_Flow_Learn': p_corrected.get('flow_learning'),
142                 'P_Corr_Collab_Learn': p_corrected.get('collaboration_learning'),
143                 **interpretations
144             })
145
146     return pd.DataFrame(results)
147
148 # --- VISUALISATIE PLACEHOLDERS ---
149
150 def create_boxplots(df_normalized, items, data_type):
151     """Create boxplots for each item"""
152
153     numeric_cols = ['flow', 'collaboration', 'learning']
154     n_items = len(items)
155
156     # Calculate grid dimensions
157     n_cols = 2
158     n_rows = int(np.ceil(n_items / n_cols))
159
160     fig, axes = plt.subplots(n_rows, n_cols, figsize=(14, 5 * n_rows))
161     fig.suptitle(f'Boxplots of Normalized Values per Item - {data_type}',
162                 fontsize=16, fontweight='bold')
163
164     # Flatten axes array for easier indexing
165     if n_items > 1:
166         axes = axes.flatten()
167     else:
168         axes = [axes]
169
170     for idx, item in enumerate(items):
171         item_data = df_normalized[df_normalized['item'] == item][numeric_cols]
172
173         axes[idx].boxplot([item_data['flow'], item_data['collaboration'],
item_data['learning']],
174                           labels=['Flow', 'Collaboration', 'Learning'])
175         axes[idx].set_title(f"{item}\n(n={len(item_data)})", fontsize=12,
fontweight='bold')
176         axes[idx].set_ylabel('Normalized Value')
177         axes[idx].grid(True, alpha=0.3)
178         # Note: removed fixed ylim since normalized values may not necessarily

```

```

be between 0-1 for individual items

179
180 # Hide extra subplots if any
181 for idx in range(n_items, len(axes)):
182     axes[idx].axis('off')
183
184 plt.tight_layout()
185 output_file = f'boxplots_{data_type.lower()}.png'
186 plt.savefig(output_file, dpi=300, bbox_inches='tight')
187 #plt.show()
188
189 print(f"\nBoxplots saved to '{output_file}'")
190 return output_file
191
192 def create_radar_charts(df_normalized, items, data_type):
193     """Create radar/spider charts for each item"""
194
195     numeric_cols = ['flow', 'collaboration', 'learning']
196     n_items = len(items)
197
198     # Calculate grid dimensions
199     n_cols = 2
200     n_rows = int(np.ceil(n_items / n_cols))
201
202     fig, axes = plt.subplots(n_rows, n_cols, figsize=(14, 5 * n_rows),
203                             subplot_kw=dict(projection='polar'))
204     fig.suptitle(f'Radar Charts - Mean Normalized Values per Item - {data_type}',
205                 ,
206                 fontsize=16, fontweight='bold')
207
208     # Flatten axes array for easier indexing
209     if n_items > 1:
210         axes = axes.flatten()
211     else:
212         axes = [axes]
213
214     # Set up angles for the three dimensions
215     angles = np.linspace(0, 2 * np.pi, len(numeric_cols), endpoint=False).tolist()
216     angles += angles[:1] # Complete the circle
217
218     # Calculate global max for consistent scaling across all items
219     max_mean_value = 0
220     for item in items:
221         item_data = df_normalized[df_normalized['item'] == item][numeric_cols]
222         max_mean_value = max(max_mean_value, item_data.mean().max())
223
224     for idx, item in enumerate(items):
225         item_data = df_normalized[df_normalized['item'] == item][numeric_cols]
226
227         # Calculate mean values
228         values = item_data.mean().tolist()
229         values += values[:1] # Complete the circle
230
231         # Plot
232         axes[idx].plot(angles, values, 'o-', linewidth=2, label='Mean')
233         axes[idx].fill(angles, values, alpha=0.25)
234         axes[idx].set_ylim(0, max_mean_value * 1.1) # Add 10% padding
235         axes[idx].set_xticks(angles[:-1])
236         axes[idx].set_xticklabels(['Flow', 'Collaboration', 'Learning'])
237         axes[idx].set_title(f'{item}\n(n={len(item_data)})', fontsize=12,
238                             fontweight='bold', pad=20)
239         axes[idx].grid(True)
240
241     # Hide extra subplots if any

```

```

240     for idx in range(n_items, len(axes)):
241         axes[idx].axis('off')
242
243     plt.tight_layout()
244     output_file = f'radar_charts_{data_type.lower()}.png'
245     plt.savefig(output_file, dpi=300, bbox_inches='tight')
246     #plt.show()
247
248     print(f"Radar charts saved to '{output_file}'")
249     return output_file
250
251 def create_heatmap(df_normalized, items, data_type):
252     """Create heatmap of average normalized values per item"""
253
254     numeric_cols = ['flow', 'collaboration', 'learning']
255
256     # Calculate mean values per item
257     mean_values = []
258     item_labels = []
259
260     for item in items:
261         item_data = df_normalized[df_normalized['item'] == item][numeric_cols]
262         mean_values.append(item_data.mean())
263         n = len(item_data)
264         item_labels.append(f"{item} (n={n})")
265
266     # Create DataFrame for heatmap
267     heatmap_data = pd.DataFrame(mean_values, index=item_labels).T
268
269     # Calculate min and max for dynamic scaling
270     vmin = heatmap_data.min().min()
271     vmax = heatmap_data.max().max()
272
273     # Create heatmap
274     fig, ax = plt.subplots(figsize=(max(10, len(items) * 1.5), 6))
275     sns.heatmap(heatmap_data, annot=True, fmt='.3f', cmap='YlOrRd',
276                 cbar_kws={'label': 'Mean Normalized Value'}, ax=ax, vmin=vmin,
277                 vmax=vmax)
278     ax.set_title(f'Heatmap of Mean Normalized Values per Item - {data_type}',
279                 fontsize=16, fontweight='bold', pad=20)
280     ax.set_xlabel('Item', fontsize=12)
281     ax.set_ylabel('Dimensions', fontsize=12)
282
283     plt.tight_layout()
284     output_file = f'heatmap_{data_type.lower()}.png'
285     plt.savefig(output_file, dpi=300, bbox_inches='tight')
286     #plt.show()
287
288     print(f"Heatmap saved to '{output_file}'")
289     return output_file
290
291 def create_stacked_bars(df_normalized, items, data_type):
292     """Create stacked bar charts showing normalized composition"""
293
294     numeric_cols = ['flow', 'collaboration', 'learning']
295
296     # Calculate mean values per item
297     mean_values = []
298     for item in items:
299         item_data = df_normalized[df_normalized['item'] == item][numeric_cols]
300         mean_values.append(item_data.mean())
301
302     mean_df = pd.DataFrame(mean_values, index=items)
303
304     # Create stacked bar chart

```

```

304 fig, ax = plt.subplots(figsize=(12, max(6, len(items) * 0.5)))
305 mean_df.plot(kind='barh', stacked=True, ax=ax,
306             color=['#FF6B6B', '#4ECDC4', '#45B7D1'],
307             edgecolor='white', linewidth=1.5)
308
309 ax.set_xlabel('Normalized Value (Proportion)', fontsize=12)
310 ax.set_ylabel('Item', fontsize=12)
311 ax.set_title(f'Stacked Bar Chart - Mean Normalized Values per Item - {
312 data_type}',
313             fontsize=16, fontweight='bold', pad=20)
314 ax.legend(title='Dimension', bbox_to_anchor=(1.05, 1), loc='upper left')
315 # Note: removed fixed xlim since values may not sum to 1 per item
316 ax.grid(axis='x', alpha=0.3)
317
318 # Add sample size annotations
319 for i, item in enumerate(items):
320     n = len(df_normalized[df_normalized['item'] == item])
321     ax.text(ax.get_xlim()[1] * 1.02, i, f'n={n}', va='center', fontsize=9)
322
323 plt.tight_layout()
324 output_file = f'stacked_bars_{data_type.lower()}.png'
325 plt.savefig(output_file, dpi=300, bbox_inches='tight')
326 #plt.show()
327
328 print(f"Stacked bar chart saved to '{output_file}'")
329 return output_file
330
331 # --- MAIN EXECUTION LOOP ---
332
333 if __name__ == "__main__":
334     # Inladen van brondata
335     FILE_PATH = 'interview_summary.xlsx'
336     df_praktijken = pd.read_excel(FILE_PATH, sheet_name='Praktijken')
337     df_problemen = pd.read_excel(FILE_PATH, sheet_name='Problemen')
338
339     # Verwerking: Praktijken
340     df_praktijken_norm = process_data(df_praktijken, 'Praktijken')
341     stats_praktijken, items_prak = calculate_statistics(df_praktijken_norm, '
342 Praktijken')
343     res_prak = calculate_scenario1_tests(df_praktijken_norm, items_prak, '
344 Praktijken')
345
346     # Verwerking: Problemen
347     df_problemen_norm = process_data(df_problemen, 'Problemen')
348     stats_problemen, items_prob = calculate_statistics(df_problemen_norm, '
349 Problemen')
350     res_prob = calculate_scenario1_tests(df_problemen_norm, items_prob, '
351 Problemen')
352
353     # Samenvoegen en exporteren
354     all_stats = pd.concat([stats_praktijken, stats_problemen], ignore_index=True)
355     all_tests = pd.concat([res_prak, res_prob], ignore_index=True)
356
357     with pd.ExcelWriter('summary_statistics.xlsx', engine='openpyxl') as writer:
358         all_stats.to_excel(writer, sheet_name='Descriptives', index=False)
359         all_tests.to_excel(writer, sheet_name='Statistische Testen', index=False)
360
361     print("\nAnalyse voltooid. Resultaten opgeslagen in 'summary_statistics.xlsx'")

```

Listing 1: Kernlogica voor data normalisatie per respondent

H.2 Software vereisten

Het script vereist de volgende Python-pakketten:

- pandas (versie 1.3.0 of hoger) – voor datamanipulatie en Excel I/O
- numpy (versie 1.21.0 of hoger) – voor numerieke berekeningen
- openpyxl (versie 3.0.0 of hoger) – voor het lezen en schrijven van Excel-bestanden

Visualisaties werden gegenereerd met matplotlib (versie 3.4.0 of hoger) en seaborn (versie 0.11.0 of hoger), maar deze code is weggelaten uit deze bijlage voor de beknoptheid.

H.3 Input en output

H.3.1 Input

Het script verwacht een Excel-bestand `interview_summary.xlsx` met twee tabbladen:

- Praktijken: Bevat kolommen `respondent`, `item`, *Flow*, *Collaboration*, *Learning* met numerieke waarden
- Problemen: Bevat dezelfde kolommen, maar met tekstuele waarderingen die gecodeerd worden volgens het codeerschema

H.3.2 Output

Het script genereert twee Excel-bestanden:

- `data_normalized.xlsx`: Bevat de genormaliseerde data voor beide datasets
- `summary_statistics.xlsx`: Bevat beschrijvende statistieken (gemiddelde, mediaan, standaarddeviatie) per item en dimensie en in een apart tabblad de gecorrigeerde Friedman-testresultaten.