



PRAKTIJKGERICHTE AANPAK VOOR

# End 2 End-Testen

---

“E2E” wil zeggen: End to End, van het ene uiterste einde naar het andere einde. E2E-processen lopen over de complete technische infrastructuur (de systeemketen) heen. Dit kunnen werkprocessen zijn van gebruikersafdelingen of beheerders, maar ook processen die door klanten worden uitgevoerd of processen waarbij andere organisaties informatie verkrijgen. Een E2E-test richt zich op de samenhang tussen deze E2E-processen en de systeemketen. De E2E-processen en de systeemketen samen worden hier de E2E-keten genoemd.

---

Gerard Numan

# Inhoudsopgave

<b>Deel 1</b>	Inleiding E2E-Testen .....	7
	1.1 E2E versus interface en systeemintegratie .....	8
	1.2 Hoe de aanpak te gebruiken .....	9
<b>Deel 2</b>	Samenvatting .....	10
<b>Deel 3</b>	Het belang van E2E-Testen .....	13
	3.1 Het belang van E2E-Testen.....	14
	3.2 E2E-processen en techniek sluiten niet aan.....	16
	3.3 Geboorte van E2E-test .....	18
	3.4 E2E-kennis .....	19
<b>Deel 4</b>	Begrippen- en literatuurlijst.....	20
	4.1 Begrippenlijst .....	21
	4.2 Literatuurlijst .....	23
<b>Deel 5</b>	E2E-inventarisatie .....	24
	5.1 Inleiding E2E-inventarisatie .....	25
	5.2 In kaart brengen E2E-processen .....	26
	5.3 In kaart brengen systeemlandschap.....	28
	5.4 Analyseren samenhang E2E-proces en systeemlandschap.....	28
	5.5 Detailleren van de werking van de E2E-keten per E2E-proces.....	31
<b>Deel 6</b>	E2E-teststrategie .....	33
	6.1 Inleiding E2E-teststrategie .....	34
	6.2 E2E-Productrisicoanalyse .....	34
	6.3 Teststrategie .....	40
<b>Deel 7</b>	E2E testen en risico's.....	46
	7.1 Productrisico's .....	47
	7.1.1 Functionaliteit.....	48
	7.1.2 Betrouwbaarheid .....	50
	7.1.3 Bruikbaarheid.....	52
	7.1.4 Efficiëntie.....	53
	7.1.5 Onderhoudbaarheid .....	55
	7.1.6 Portabiliteit .....	56
	7.2 Projectrisico's .....	57
	7.2.1 Testbasis .....	57
	7.2.2 Planning en beheer .....	58
	7.2.3 Testomgevingen .....	58
	7.2.4 Testdata .....	59
	7.2.5 Testuitvoering .....	59
<b>Deel 8</b>	E2E-testplanning .....	60
	8.1 Inleiding E2E-testplanning.....	61
	8.2 Uitgangspunten .....	61
	8.3 Vaststellen testclusters .....	63
	8.4 Inventarisatie van activiteiten en afhankelijkheden .....	64
	8.5 Planning, beheer en analyse.....	64
	8.6 Ontwerp en implementatie.....	65
	8.7 Doorlooptijd testuitvoering.....	65
	8.8 Inzet testuitvoering.....	68
	8.9 Invloeden op de planning .....	68
	8.10 Project- en organisatieoverschrijdende E2E-testen .....	69

<b>Deel 9</b>	E2E-testontwerp .....	70
	9.1 Testontwerp: logische testgevallen.....	72
	9.2 Testontwerp: fysieke testgevallen .....	76
	9.3 Opstellen testprocedure.....	79
	9.4 Speciale testtypen .....	80
	9.4.1 Betrouwbaarheidstest .....	80
	9.4.2 Efficiencytest (performancetest) .....	81
	9.4.3 Securitytest.....	82
<b>Deel 10</b>	De E2E-testorganisatie .....	83
	10.1 De onderdelen van de E2E-testorganisatie .....	84
	10.1.1 De E2E-testmanager .....	85
	10.1.2 De E2E-tester (E2E-kernteam).....	85
	10.1.3 De E2E-board .....	86
	10.1.4 Testers .....	86
	10.2 Plaats van de E2E-test in de organisatie .....	87
	10.2.1 Grensoverschrijding tussen IT en business: systeemtest of acceptatietest? .....	87
	10.2.2 Grensoverschrijding tussen projecten .....	89
	10.2.3 Grensoverschrijding tussen organisaties.....	90
	10.3 Eigenaarschap .....	91
	10.3.1 Technisch eigenaarschap .....	91
	10.3.2 Functioneel eigenaarschap.....	91
	10.3.3. Ketenmanagement.....	91
	10.3.4 Escalatiekanalen .....	92
	10.3.5 Acceptanten.....	92
	10.4 Schaling E2E-testen.....	93
	10.4.1 Integratie- en E2E-testen in testsoorten .....	93
	10.4.2 Afwegingen bij de invulling van een E2E-test.....	94
	10.4.3 Varianten uit de praktijk .....	95
<b>Deel 11</b>	De E2E-testinfrastructuur.....	98
	11.1 Testomgevingen .....	99
	11.1.1 Beheer .....	99
	11.1.2 Welke omgevingen?.....	100
	11.1.3 Varianten .....	101
	11.1.4 Continue integratie.....	103
	11.1.5 De speeltuin: continue acceptatie .....	103
	11.1.6 Testen in productie .....	103
	11.2 Testdata.....	104
	11.2.1 Datasynchroniciteit en databeheer.....	104
	11.2.2 Tijdreizen .....	105
	11.2.3 Productiedata, anonimiseren of laboratoriumdata .....	105
	11.3 Tools.....	106
	11.3.1 Bevindingenadministratie over de keten heen.....	106
	11.3.2 Simuleren .....	107
	11.3.3 Beoordelen en manipuleren .....	107
	11.3.4 Testautomatisering .....	107

<b>Deel 12</b>	Checklists en voorbeelden E2E-testen .....	108
12.1	Checklists .....	109
12.1.1	Voordelen van een E2E-test .....	109
12.1.2	Documenten voor de E2E-inventarisatie .....	110
12.1.3	Uitgangspunten E2E-productrisicoanalyse.....	110
12.1.4	Vaststellen acceptatiecriteria .....	110
12.1.5	Entry-criteria .....	111
12.1.6	Exit-criteria .....	111
12.1.7	Randvoorwaarden E2E-test .....	111
12.1.8	Vaststellen testomgeving.....	112
12.1.9	Testdatabeheer.....	112
12.1.10	Tijdreizen.....	112
12.1.11	Testdraaiboek .....	113
12.1.12	Profiel van een E2E-tester .....	113
12.1.13	Integratie- en E2E-testen in andere testsoorten .....	114
12.1.14	Afwegingen bij de invulling van een E2E-test.....	114
12.1.15	Intake testomgeving.....	115
12.1.16	Bevindingenbeheer.....	116
12.2	Voorbeeld logische testgevallen (data combinatie test) .....	116
12.3	Voorbeeld fysieke testgevallen.....	118
12.4	Voorbeeld draaiboek testomgeving.....	120

## Voorwoord

Toen ik in 1998 een loopbaan als tester begon, liep ik al snel tegen het fenomeen integratietesten aan. Testen bleek toen al vrijwel nooit alleen maar te bestaan uit het testen van "een" (1) systeem. Interfacetesten was al wel een ingeburgerd begrip en zelfs de systeemintegratietest of ketentest was bekend. Wat met deze begrippen exact werd bedoeld, was echter vaak niet duidelijk en nog steeds is het maar de vraag wat er is getest als men beweert een interfacetest of systeemintegratietest te hebben uitgevoerd.

Risico's op het gebied van integratie zijn echter toegenomen. Cloud, Agile, SOA zijn allesbehalve redenen om aan te nemen dat integratietesten minder relevant wordt.

Ik heb sinds 1998 al vermoed dat er in de definities van integratietesten iets miste, dat "wij IT-ers" een tunnelvisie hebben, waardoor wij bij voorbaat niet in staat zijn om bepaalde integratieproblemen te voorkomen.

In 2002 kreeg ik het voorrecht op een conferentie iets te mogen zeggen over wat ik toen met een ongelukkig woord "Chain Testing" noemde. De lezing ging vooral over ketenrisico's en de maatregelen die daarbij hoorden. Deze zocht ik voornamelijk in de organisatie van de testen en het in kaart brengen van alle koppelingen. Ook daar miste nog steeds een belangrijk punt: ik bekeek integratie als een schakeling van systemen. Op zich is dat niet onjuist. Het is echter onvolledig: integratie heeft namelijk als doel om een proces te ondersteunen.

Integratie is niet het integreren van systemen, maar het mogelijk maken van een proces. Pas in het proces blijkt de integratie, niet door in systemen te kijken. Daarom zijn interfacetesten en systeemintegratietesten niet genoeg (hoewel ze wel noodzakelijk zijn).

Dit is de ontdekking die ik gedaan heb. Ik zal er nooit een Nobelprijs mee winnen: ik ben niet de eerste die deze ontdekking heeft gedaan. In de geschiedenis, in de IT, in de testwereld; overal is dit inzicht al bekend. Het is mij echter niet bekend of er een aanpak is die dit expliciet (of expliciet genoeg) benoemt en die de consequenties (voor testen) voldoende uitwerkt. Deze aanpak is dan wellicht de eerste poging.

Ik richt mij in deze aanpak op de uiteindelijke integratietest en noem hem E2E-test. Ook dit is niet mijn ontdekking: deze term bestaat al in min of meer dezelfde betekenis en dekt daarmee de lading afdoende. Ik weet ook geen beter woord. "Doorlooptest", "Procestest", "Ketentest" lijken mij allen een of meer aspecten te missen.

Ik durf te stellen dat E2E-testen de mooiste en meest veelzijdige vorm van testen is. In de E2E-test komt alles samen: processen, techniek en bovenal mensen. Ik heb herhaaldelijk meegemaakt dat vanuit de E2E-test testers een rol kregen in functioneel en technisch beheer, ontwerp en projectmanagement. E2E-testen is in meerdere opzichten daarom grensoverschrijdend.

Ik hoop met deze aanpak wat licht te scheppen in de wereld van integratietesten. Ik denk dat ervaren testers en testmanagers er tools in kunnen aantreffen om hun dagelijks werk effectiever en efficiënter uit te voeren. Ik meen dat het aan managers uitlegt wat E2E is, wat het niet is, wat je eraan hebt en wat ervoor nodig is. Ik hoop dat het een verrijking mag zijn voor ons mooie testvak.

**Gerard Numan**

[gerard.numan@polteq.com](mailto:gerard.numan@polteq.com)

© Polteq

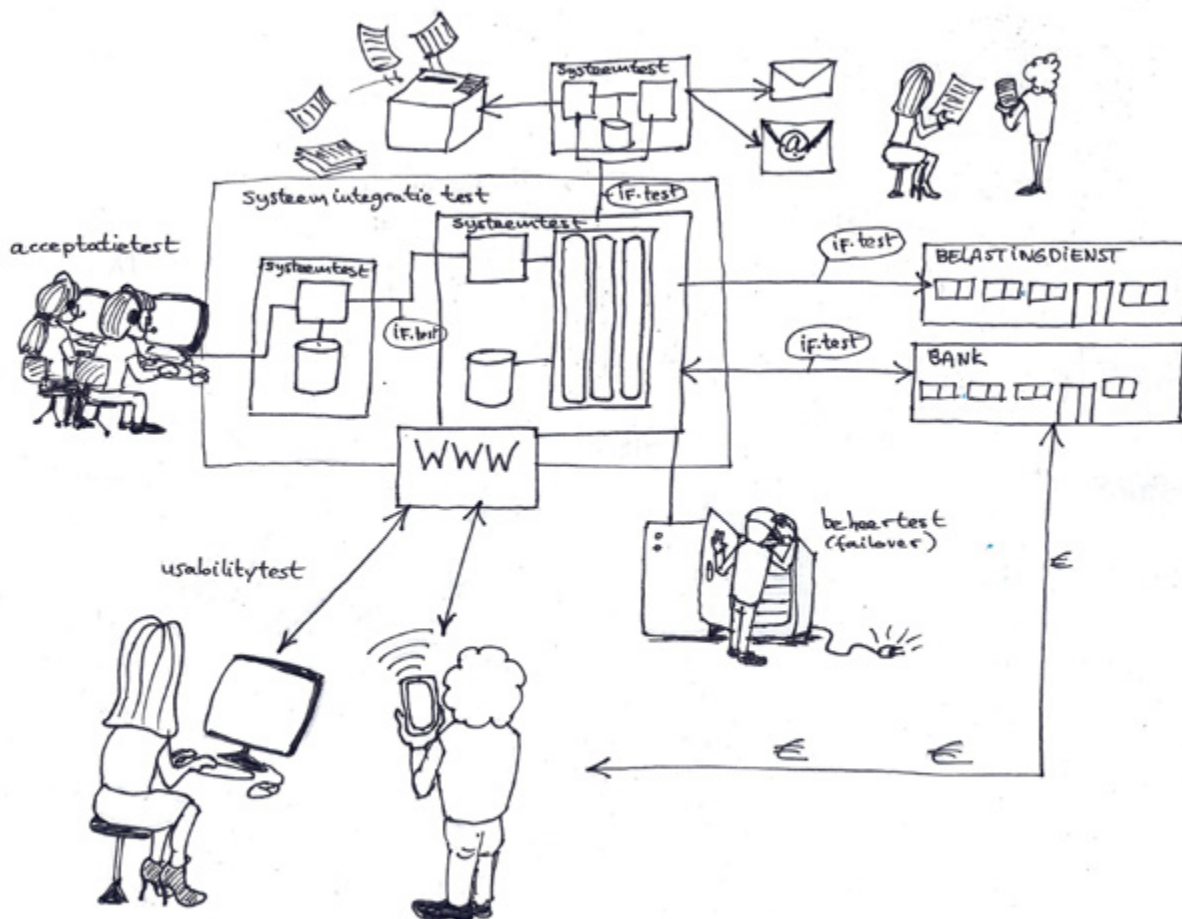
## Dankwoord

In de eerste plaats moet ik Kees Blokland en Martin Pol noemen. Kees Blokland heeft de meeste invloed gehad op het eindproduct, door zijn aanwijzingen, reviewcommentaar, suggesties, geschonken vertrouwen, organisatietalent en zijn rol als sparringpartner. Martin Pol vanwege zijn visionaire talent. Al vroeg zag hij het belang van het thema en gaf mij de ruimte en het vertrouwen om er de nodige uren in te steken.

Daarnaast wil ik Theo Wieringa bedanken. Theo heeft als testmanager in een aantal projecten met mij E2E-testen georganiseerd. Hij zal veel herkennen in de artikelen. Het meeste van de aanpak is in trajecten met Theo ontstaan en bijgeslepen.

Mijn laatste dankwoord gaat uit naar de reviewers: Carla de Graaff, François Meijerink, Erwin Lamberts, Marc Kramer, Michiel Pallandt, Johan van Ingen, Jelle Roovers, Maurice de Regt, Hans Rombouts, Jos van de Goede, Jan Wouter Lok, Linda Pol, Nico Reinders en Jako Datema. Zij zijn vitaal geweest voor de uiteindelijke consistentie en begrijpelijkheid.

### Gerard Numan



*Figuur: End-to-end test omvat alles*



# Deel 1

## Inleiding E2E-Testen

# Deel 1 Inleiding E2E-testen

Deze aanpak legt uit wat het belang van E2E-testen is, positioneert E2E-testen ten opzichte van andere testsoorten en testtypes, beschrijft technieken waarmee de belangrijkste activiteiten kunnen worden uitgevoerd en inventariseert de benodigde organisatie en infrastructuur. Ten slotte worden een verzameling sjablonen en checklists geleverd.

## 1.1 E2E versus interface en systeemintegratie

Op dit moment bestaat er nog geen eenduidigheid en consensus rondom het testen van interfaces, systeemintegratie en E2E-processen. In geval van complexe infrastructuren weet men vaak nog niet wat te testen, hoe te testen, wanneer te testen en hoe de testen te organiseren. Vaak is niet duidelijk wat het onderscheid is tussen interfaces, functies, systeemintegratie en E2E-processen en hoe deze elementen samenhangen. Zo zijn er veel namen in omloop voor het testen van integratie: Interfacetest, Integratietest, Systeemintegratietest, Ketentest, Keten Integratie Test, Functionele Integratie Test, End to End Test, Proces Keten Test en Business Process Test. Wat wordt bedoeld met deze namen kan in de praktijk verschillen.

Om de definities te ordenen worden hier drie logische niveaus van integratietesten onderscheiden. Deze corresponderen met de opeenvolgende stappen van integratie, elke definitie van integratietesten kan worden geplot in deze indeling:

1. Interfacetest: de unit of systeemtest van een koppeling.
2. Systeemintegratietest (SIT): de systeemtest van een aantal systemen samen.
3. E2E-test: waarin de samenhang van complete E2E-processen met het complete systeemlandschap wordt getest.

Elk van deze niveaus heeft zijn eigen scope, moment en invulling. Ze dienen begripsmatig en qua risicodekking van elkaar onderscheiden te worden.

Interfacetesten richten zich op de connecties tussen systemen. Systeemintegratietesten richten zich op het gemeenschappelijk functioneren van systemen. Hierbij worden systeemoverschrijdende functies als uitgangspunt genomen. Met E2E-testen wordt echter bedoeld: het testen van de samenhang tussen aan de ene kant het daadwerkelijk gebruik, de daadwerkelijke gebeurtenissen (de zogenaamde E2E-processen) en aan de andere kant de keten van geautomatiseerde systemen, de daarin bestaande gegevens en de middelen.

Systeemintegratietesten en interfacetesten zijn gebaseerd op vastgelegde specificaties (van functies en interfaces), E2E-testen richten zich op dat wat het uiteindelijke doel van geautomatiseerde systemen is: het ondersteunen van E2E-processen. Interface-, systeemintegratie- en E2E-testen zijn de 3 basisvarianten van integratietesten. Het kan wel efficiënt en praktisch zijn om systeemintegratie- en E2E-aspecten zo vroeg mogelijk, in samenhang te testen. SIT en E2E-test zijn dus niet per definitie van elkaar gescheiden testsoorten.

Het belang en de complexiteit van E2E-testen (maar ook van systeemintegratietesten) neemt alleen maar toe: automatisering dringt steeds dieper door in het dagelijks leven, processen en systemen worden steeds flexibeler aan elkaar gekoppeld. IT-projecten, en de testsoorten daarbinnen, hebben vaak moeite om een goed en volledig begrip te krijgen van de E2E-processen, het daadwerkelijke gebruik en wat er allemaal kan gebeuren in de systeemketens. Hiervoor kunnen geen "schuldigen" aangewezen worden: de wereld van de business en de wereld van de techniek blijken gewoon moeilijk op elkaar aan te sluiten. Gevolg is dat er, ondanks een vaak hoog niveau van formalisering van het ontwerp- en testproces, toch belangrijke risico's kunnen worden gemist op het gebied van integratie en samenhang in de E2E-keten.



## 1.2 Hoe de aanpak te gebruiken

De aanpak met zijn onderliggende deelartikelen is een handboek dat voornamelijk als naslagwerk en werkinstructie dienst zal doen. Het e-book bestaat uit een aantal delen die elk een aspect behandelen. Alle belanghebbenden bij een E2E-test kunnen in één of meer van deze delen de voor hen relevante informatie halen. De structuur van de aanpak en voor wie welk deel interessant is, wordt hieronder beschreven.

**2. Samenvatting.** De belangrijke boodschappen van de aanpak worden opgesomd en kort besproken. De punten worden elders in het boek onderbouwd en uitgewerkt.

**3. Het belang van E2E-testen.** In dit hoofdstuk wordt het belang van E2E-testen uitgelegd. Aan de hand van concrete praktijkvoorbeelden worden verschillende typen risico's geïllustreerd. Er wordt uitgelegd dat er belangrijke risico's kunnen zijn die alleen door een E2E-test kunnen worden gedekt. Voor iedereen die nog niet overtuigd is van het belang van E2E-testen of die anderen van het belang moet overtuigen.

**4. Begrippen- en literatuurlijst.** Een lijst met definities van centrale begrippen en gebruikte literatuur.

**5. E2E-inventarisatie.** De E2E-inventarisatie is een techniek om E2E-processen mee te verhelderen en te ontleden zodat op basis daarvan risico's kunnen worden geanalyseerd en testgevallen ontworpen. Voor E2E-testers is het één van de kerntaken. Zij moeten deze techniek kunnen hanteren.

**6. E2E-teststrategie.** De teststrategie biedt testmanagers en testcoördinatoren een handleiding om een productrisicoanalyse (PRA) gericht op E2E-risico's te organiseren, risico's vast te stellen en testmaatregelen te definiëren. Voor projectmanagers, beheerders en testers biedt dit artikel een nadere kennismaking met typische E2E-productrisico's.

**7. E2E Testen en risico's.** In dit hoofdstuk worden typische E2E-risico's beschreven aan de hand van de bekende kwaliteitsattributen (ISO9126). Per mogelijk risico worden criteria beschreven om de hoogte van het risico te bepalen en testmaatregelen te definiëren. Daarnaast worden ook meest voorkomende E2E-projectrisico's besproken. Dit hoofdstuk is te gebruiken als achtergrondinformatie om begrip van E2E te krijgen, bij een risico-analyse, om een teststrategie of testgevallen op te stellen, maar ook om een projectplan te schrijven.

**8. E2E-testplanning.** De testplanning voor een E2E-test heeft eigen afwijkende principes: ze moet bijvoorbeeld worden afgeleid van de doorlooptijd van testgevallen. Bovendien kent de E2E-test van alle testsoorten de meeste afhankelijkheden. De E2E-planning is voor testmanagers en testcoördinatoren geschreven. Voor projectmanagers en E2E-testers is de beschrijving te gebruiken als achtergrondinformatie.

**9. E2E-testontwerp.** E2E-testgevallen zijn niet gebaseerd op één of enkele volledige en duidelijke ontwerpen, maar op een combinatie van ontwerpen, requirements, handleidingen, beschrijvingen, werkinstructies, procesbeschrijvingen en informele informatie. De testontwerptechniek is erop gericht om op basis van al deze informatie een handzame en effectieve set testgevallen te ontwerpen die voor alle betrokkenen begrijpelijk zijn. In dit hoofdstuk kunnen E2E-testers de testtechniek en sjablonen vinden om E2E-testgevallen te ontwerpen.

**10. E2E-testorganisatie.** Dit deel is met name interessant voor projectmanagers, business managers en testmanagers. Het laat zien hoe de E2E-test moet worden ingevuld, wat daarbij de overwegingen zijn en welke impact dit heeft op andere delen van de organisatie.

**11. E2E-testinfrastructuur.** Dit hoofdstuk is bedoeld voor testmanagers en testcoördinatoren. Beheerders van testomgevingen en E2E-testers kunnen het lezen als achtergrondinformatie. De onderwerpen zijn: testomgevingen, testdata en tools.

**12. Checklists en voorbeelden E2E-testen.** De checklists kunnen worden gebruikt door testmanagers en testcoördinatoren. Het gaat o.a. om checklists voor kwaliteitsattributen, product- en projectrisico's en randvoorwaarden voor het testtraject. De checklists kunnen ook door ieder andere belanghebbende worden gebruikt. Daarnaast wordt een aantal praktijkvoorbeelden gegeven van een draaiboek, een testplan en een voortgangsrapport.



# Deel 2

## Samenvatting

## Deel 2 Samenvatting

**Noodzaak en belang.** Er is vaak veel onduidelijkheid en weinig consensus over wat er moet worden getest als het gaat om de integratie van E2E-processen en systemen. Begrippen en definities zijn nog onduidelijk, tegenstrijdig en vaag, waardoor organisaties vaak risico's onderschatten of niet weten wat, hoe en wanneer te testen. Daar komt bij dat systemen steeds complexer worden, steeds meer gekoppeld worden en dat automatisering steeds verder doordringt in het dagelijks leven en in E2E-processen binnen en buiten organisaties. Agile ontwikkeling, Service Oriented Architecture, Cloud en uitbesteding maken het testen van de samenhang tussen systemen bovendien alleen maar meer complex en kritisch.

E2E-testkennis kan ook door anderen worden gebruikt. Te vaak mist informatie over de precieze samenhang tussen E2E-processen en systemen. Systeemontwerpers, bouwers en procesontwerpers kunnen bijdragen aan en profiteren van de door E2E-testers verzamelde kennis. E2E-testen kan de basis worden van een E2E-testcentrum.

**Interface, Integratie, Acceptatie en E2E.** In geval van een reeks van geschakelde systemen moeten de 3 volgende testen worden overwogen: testen van interfaces (connecties tussen systemen); testen van integratie van systemen (gezamenlijk functioneren van systemen); testen van End to End (processen over de gehele keten van systemen heen). Dit betekent niet dat elk van deze testen altijd een aparte testsoort zou moeten zijn. De betreffende testfocus (connectie, functioneren, E2E-proces) moet echter goed worden belegd. Afhankelijk van de grootte van de keten en de risico's, kiest men al naar gelang voor een opzet in testsoorten of in testtypen. Interfaces en integratie zouden in sommige gevallen ook bij de systeemtestteams kunnen worden ondergebracht. E2E-testen is vaak een samenwerking met de gebruikersacceptatietesten en beheerders.

**Risico's.** Typische productrisico's die met name door een E2E-test gedekt worden, zijn onder andere: datasynchroniciteit, performance, gegevensverlies, beschikbaarheid, actualiteit van informatie, volledigheid van informatie, timing en failover. Deze risico's zijn vaak moeilijk in andere testsoorten te testen of worden niet of te laat onderkend.

**Processen in plaats van specificaties.** De testbasis van een E2E-test bestaat niet alleen maar uit de functionele specificaties, maar ook uit de samenhang tussen E2E-processen en de techniek. Met E2E-processen worden de E2E-processen bedoeld die gebruik van maken van de systemen, van begin tot eind, van startactiviteit tot eindresultaat. Dit zijn werkprocessen, zoals beheerders die een wijziging in een parameter uitvoeren of gebruikers die een mutatie op klantgegevens uitvoeren. Maar het kunnen ook processen zijn waarbij automatisch gegevens tussen organisaties worden uitgewisseld of een patiënt die een apparaat gebruikt voor een bloedmeting.

**Inventarisatie E2E-processen.** E2E-processen en hun samenhang met de systemen zijn vaak slecht beschreven of moeilijk te vinden. E2E-testen moet zich daarom baseren op een door testers zelf uit te voeren inventarisatie van E2E-processen en hun samenhang met systeemonderdelen. Daarnaast zal met belangrijke partijen moeten worden afgestemd welke entiteiten relevant zijn (producten, varianten, componenten, technische processen, E2E-processen en dergelijke) en hoe deze moeten worden gedekt door de testen.

**E2E-board.** De E2E-processen worden waarschijnlijk niet volledig door professionele testers gevat en doorgrond. Er moet daarom een vorm gevonden worden om relevante experts uit de organisatie op een, ook voor hen acceptabele, manier aan te sluiten bij de E2E-test. Dit kan door een E2E-board (commissie/ panel) in te stellen waarin deze experts dagelijks bijeenkomen en waar het E2E-test team verslag uitbrengt en om advies vraagt.

**Testontwerp.** Niet alle traditionele testtechnieken zijn toepasbaar in een E2E-test. Testgevallen moeten namelijk op basis van de E2E-inventarisatie en "echte" scenario's worden uitgewerkt. Hier binnen zijn, afhankelijk van risico's, op delen van de E2E-keten technieken als grenswaardenanalyse of een datacombinatietest toe te passen.

**Planning.** De planning van de E2E-test begint vanuit de minimale en maximale doorlooptijd van de testgevallen en de benodigde testronde. Vanuit deze informatie wordt gedurende het project het geraamte van de E2E-testplanning van steeds meer vlees voorzien. Planningstechnieken zoals Testpuntanalyse (TPA) zijn van geen waarde voor een E2E-test: er zijn te veel afhankelijkheden.

**Organisatie.** Een E2E-testteam bestaat uit een E2E-testmanager, tenminste één E2E-tester en een panel van E2E-proces- en systeemspecialisten: de E2E-board. De leden van de E2E-board zijn parttime betrokken bij de E2E-test. Ze kunnen zo nog steeds andere werkzaamheden uitvoeren en toch bijdragen aan de E2E-test, waar ze meedenken over testgevallen, risico's, bevindingen en acceptatie. Idealiter ontstaat een E2E-testcentrum waar E2E-kennis wordt bewaard en uitgedragen, testomgevingen worden beheerd en wordt aangesloten bij releasemanagement. Voor testuitvoering worden andere testers (bijvoorbeeld uit de gereed gekomen systeemtest) en gebruikers (elk op zijn eigen E2E-proces) ingezet.

**Flexibiliteit en beheersbaarheid.** E2E-testen worden in testclusters opgedeeld (per E2E-proces en per test type). Van elk testcluster is duidelijk wie de gebruikers en acceptanten zijn en deze zijn betrokken bij de testuitvoering. Op deze manier kan er voortgang worden geboekt, ook als nog niet alle systemen beschikbaar zijn en kunnen delen van de totale E2E-keten sneller worden geaccepteerd.

**Verantwoordelijkheden.** Management van IT en de beheerorganisatie, maar ook de systeemeigenaren en proceseigenaren dienen op de hoogte te zijn van het belang van E2E-testen en deze te ondersteunen met mensen en middelen. De E2E-testmanager houdt elk van hen op de hoogte van belangrijke ontwikkelingen en is partner in het uitdragen en verwezenlijken van organisatie- en afdelingsdoelen en behoeften. De E2E-test kan ontzorgen als het gaat om grote productrisico's en het ontwikkelen van een strategisch kwaliteitsbeleid, maar heeft daarbij het mandaat en de steun nodig van de belangrijkste stakeholders, zeker als E2E-testen projectoverstijgend of zelf organisatieoverstijgend zijn.

**Testomgevingen en testdata zijn de grond waar de E2E-test op staat.** Maar deze grond kan een moeras worden. Datasynchroniciteit, tijdreizen, beschikbaarheid en versiebeheer van systemen zijn niet vanzelfsprekend. Met name in een E2E-test is het van belang dat de gehele installatie, inrichting en uitvoering van batches en dergelijke zo gelijk als mogelijk aan de praktijk is. Beheer, opzet, tooling en controle van de omgeving en de data kunnen elk dagtaken zijn en worden moeilijk door testers ingevuld. Toch zal de regie bij het testteam moeten liggen en zijn de relaties met de betreffende experts vitaal.

**Regressietesten.** Binnen een E2E-testtraject vinden regressietesten plaats, bijvoorbeeld na oplossen van E2E-bevindingen. Maar E2E-testen worden in toenemende mate zelf als regressietest ingezet, bijvoorbeeld bij wijzigingen in het systeemlandschap vanuit meerdere projecten. Met "Continue integratie" wordt dan bedoeld dat een E2E-testset periodiek wordt uitgevoerd op een E2E-testomgeving.

De doorlooptijd van een E2E-test is relatief lang. Daarom moeten er elke keer testgevallen uit de complete set worden geselecteerd op basis van risico's. Traceerbaarheid van E2E-processen, componenten en functies naar E2E-testgevallen zijn hierbij een randvoorwaarde: elke keer moet, op basis van de wijzigingen in de E2E-proces- en systeemketen, kunnen worden bepaald welke testgevallen wel of niet dienen te worden uitgevoerd.

**Cloud computing en E2E-testen.** Met Cloud Computing en uitbesteding komen delen van het systeemlandschap steeds verder af te staan van de E2E-processen en de organisatie die gebruik maken van deze systemen. Om kwaliteit van dienstverlening te garanderen, moet er inzicht zijn in de samenhang tussen E2E-processen en systemen. Testen, en dan voornamelijk E2E-testen, zijn een uitstekend middel dit inzicht te verkrijgen. Verrassenderwijs is te voorzien dat de vaardigheden van E2E-testers en het E2E-testproces geschikt zijn om ingezet te worden in de aanvangsfasen van Cloud Computing projecten. Het gaat dan onder andere ook om de selectie van de te kiezen leverancier en een vroege validatie van de gekozen oplossing.

**Agile en E2E-testen.** Een risico van Agile werken is dat Agile teams het grote plaatje uit het oog verliezen. Agile sluit daarmee zeker niet de noodzaak van E2E-testen uit. Aan de andere kant biedt het multidisciplinaire karakter van Agile de mogelijkheid om vroeg vanuit E2E-testperspectief te werken. Het is daarom de kunst om als onderdeel van sprints en iteraties de juiste E2E-testactiviteiten op te zetten, bijvoorbeeld in de vorm van "Continue Integratie" (Zie sectie "Regressietesten" hierboven), waarbij een E2E testomgeving voor de Agile teams operationeel is waar tussentijdse versies aan E2E-testen kunnen worden onderworpen. Goede testomgevingen en synchrone testdata zijn dan wel een uitdaging.



## Deel 3

# Het belang van E2E-Testen

## Deel 3 Het belang van E2E-testen

### 3.1 Waarom worden sommige belangrijke fouten niet gevonden?

Waarom worden in projecten belangrijke risico's nog steeds gemist? Hoe komt het dat ondanks het toepassen van formele ontwikkel- en testmethoden nog grote problemen kunnen ontstaan na het in gebruik nemen van software?

Een deel van dit soort problemen ontstaat door de kloof tussen techniek en proces. Met proces wordt hier bedoeld: de zichtbare activiteit van een organisatie of organisatieonderdeel. Het gaat om activiteiten waarmee een dienst of product wordt gerealiseerd voor een andere organisatie, een ander organisatieonderdeel of een persoon.

Een verzekeringsmaatschappij heeft als processen in deze zin bijvoorbeeld: het afsluiten van een polis, het verwerken van schade, het opheffen van een polis. Deze processen worden vanaf nu "E2E-proces" genoemd, om ze te onderscheiden van deelprocessen en technische processen.

Vanuit de IT is er vaak te weinig begrip van het E2E-proces en vanuit gebruikers weinig kennis van de IT-techniek. Hierdoor sluiten E2E-processen en techniek vaak niet op elkaar aan. Dit begint in de requirements- en ontwerpfasen en zet zich voort tot en met de acceptatietesten en productie.

#### Een voorbeeld: de insuline meter

Een Amerikaanse producent van kleine glucose meters voor dagelijks gebruik door diabetici besloot zijn meters in Europa op de markt te brengen. De enige moeilijkheid was dat in de V.S. andere eenheden voor het meten van glucose werden gebruikt: 1 mg/dl in de V.S. staat voor 0.05556 mmol/l in de meeste Europese landen.

De software in het apparaat werd aangepast, waarbij voor de Europese apparaten de Europese schaal werd toegevoegd en deze standaard werd getoond. In geval het apparaat terug ging naar de fabrieksinstellingen (bijvoorbeeld na onderbreken van het stroomcircuit ten gevolge van het vervangen van de batterij), werden de oorspronkelijke Amerikaanse eenheden weer getoond. Als waarschuwing knipperden de karakters in het scherm, waarna de gebruiker eenvoudig kon instellen welke eenheden gewenst waren. Er vond, zoals dat gebruikelijk is bij medische toepassingen, een zwaar testtraject plaats waarna het apparaat op de Europese markt werd gebracht. Na verloop van tijd bleek dat er problemen ontstonden. Sommige patiënten bleken veel te veel insuline te hebben gespoten na het gebruik van de meter. Het gevolg was een zogenaamde hypoglycaemia, met soms dramatische gevolgen. Er vielen zelfs doden.

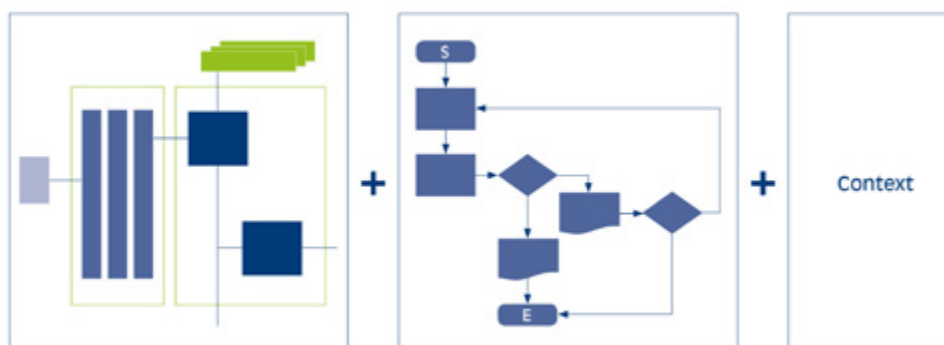
Wat bleek? Diabetes patiënten gebruiken de meter periodiek om vast te stellen of en hoeveel insuline moet worden gespoten. Als zij zich minder goed voelen, meten zij zich nog een keer. Als de glucosewaarde bij een patiënt hoog is, treedt vaak een vermindering van het gezichtsvermogen en de ooghandcoördinatie op. De apparaten vielen in deze gevallen dus vaak. Bij de val raakte het stroomcircuit dikwijls onderbroken en ging het apparaat terug naar de fabrieksinstellingen: de glucosewaarden werden daarna in Amerikaanse eenheden getoond. Dit betekende een 18 maal hoger getal dan in Europese eenheden. De eenheden indicator liet wel knipperend de Amerikaanse eenheden zien, maar de patiënten waren op dat moment niet in staat zich dat te realiseren. Patiënten raakten in paniek en namen veel te veel insuline in, met dramatische gevolgen. Het probleem is daarna simpel verholpen door de Amerikaanse eenheden te schrappen in de Europese versie.

**Lessen uit het voorbeeld.** Ontwerpers en bouwers hadden zich niet verdiept in de wereld van de patiënt: er was niet voorzien dat in sommige omstandigheden de patiënt het apparaat vaker laat vallen en dan met een andere blik naar het scherm kijkt. Het risico dat het tonen van Amerikaanse eenheden vaak zou optreden en dat dit tot een gevaarlijke beslissing door de patiënt zou kunnen leiden, was niet goed ingeschat.

Het voorbeeld laat een aantal zaken zien:

- Belangrijke risico's zijn afhankelijk van hoe een product daadwerkelijk wordt gebruikt, in de daadwerkelijke omstandigheden.

- Daadwerkelijk gebruik is eenvoudig verkeerd in te schatten indien men zelf geen gebruiker (of patiënt) is en men niet alle mogelijke situaties kent waarin het product zal worden gebruikt.
- Daadwerkelijk gebruik en omstandigheden kunnen wezenlijk verschillen van voorgeschreven of vastgesteld gebruik (zoals vastgelegd in handleidingen, werkinstructies, procesbeschrijvingen, specificaties).



*Figuur: systeem + proces + context bepaalt succes*

De problemen hadden kunnen worden voorkomen indien ontwerpers zich beter hadden verdiept in alle omstandigheden waarin patiënten het apparaat gebruiken en door echte patiënten de apparaten te laten testen onder begeleiding.

Hetzelfde geldt voor veel problemen die te maken hebben met software. Het gaat dan niet alleen maar om het niet begrijpen van de patiënt of de eindgebruiker maar ook om allerlei andere onvoorziene omstandigheden, situaties of combinaties van gebeurtenissen die het juist functioneren nadelig beïnvloeden.

- Als gegevens worden geconverteerd uit een ander systeem kan men vaak niet voorzien hoe het nieuwe systeem gaat functioneren met deze andere gegevens.
- Toegang geven aan klanten via het internet geeft een onvoorspelbaar gebruik van de software, met mogelijke gevolgen voor beschikbaarheid, veiligheid, functionaliteit en performance.
- Nieuwe koppelingen tussen systemen leveren nieuwe gegevensstromen op waarvan moeilijk is te voorspellen of deze overal dezelfde betekenis hebben en of alle E2E-processen goed zijn voorbereid op de nieuwe gegevens.
- Een nieuw groot pakket implementeren betekent in de praktijk dat werkprocessen moeten worden aangepast. Hoe het nieuwe werkproces er uit moet zien is vaak alleen vast te stellen door "het te doen", met echte gegevens, echte accounts en een complete testomgeving.

Dit soort risico's nemen alleen maar toe omdat systemen steeds meer worden gekoppeld en geïntegreerd in werkprocessen en het dagelijks leven; processen, producten en diensten steeds meer afhankelijk worden van software en de noodzaak van kleine, kortcyclische wijzigingen in systemen toeneemt.

Het is voor IT-professionals (en dus ook voor testers) steeds lastiger te begrijpen hoe het specifieke onderdeel waar zij aan werken samenhangt met andere systemen en de rol die het speelt in het E2E-proces. Voor de business (managers, klanten, gebruikers) wordt het steeds moeilijker om de samenhang in te zien tussen hun eigen processen en de technische infrastructuur.

## Conclusie

Risico's worden groter als het gaat om de samenhang tussen E2E-processen, techniek, techniek onderling en gegevens. Traditionele testmaatregelen, zoals unit testen, systeemtesten, systeemintegratietesten, acceptatietesten en formele reviews, schieten tekort in het vinden en voorkomen van dit type risico's.



### 3.2 E2E-processen en techniek sluiten niet aan

Een groot deel van de genoemde problemen ontstaat omdat er geen goede aansluiting is tussen diverse specialisten en belanghebbenden. De verschillende werelden hebben onvoldoende begrip van en voor elkaar. Dit speelt tussen (A) business en techniek (IT), maar ook tussen (B) business groepen onderling, tussen klanten en business of (C) techniek onderling (denk aan ontwerpers, programmeurs en testers onderling). Hierdoor sluiten E2E-processen vaak niet goed aan op andere E2E-processen, sluiten E2E-processen niet goed aan op techniek en sluit techniek niet goed aan op techniek.

#### Voorbeeld: E2E-proces sluit niet aan op techniek

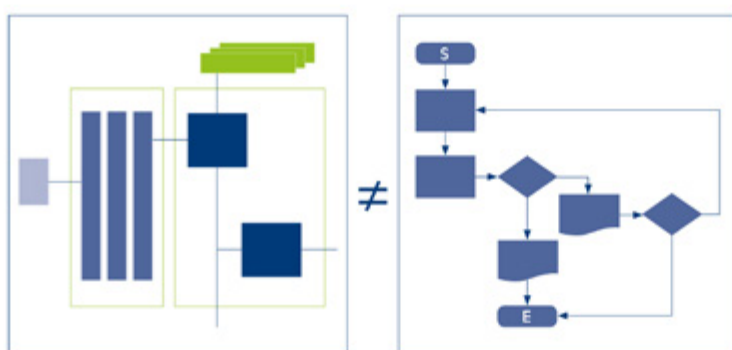
In een nieuwbouwtijdperk van een nieuwe klantenadministratie werd een aanzienlijk deel van de tijd en het geld besteed aan de ontwikkeling en test van de printfunctionaliteit. Een aantal weken na inproductienamen hoopten de prints zich op. Bij de gebruikersafdeling vond men het handiger om af en toe vanaf het scherm te printen en keek men niet meer om naar de grote hoeveelheden automatisch geprinte documenten.

#### Voorbeeld: business processen sluiten niet op elkaar aan

Binnen de Telecom branche werd in 1999 "nummerportabiliteit" ingevoerd. Dit betekende dat vanaf dat moment klanten konden overstappen naar een andere netwerkaanbieder waarbij ze hun mobiele nummer konden meenemen. Bij een van de aanbieders wist het winkelpersoneel niets van deze nieuwe E2E-processen. Klanten gingen daardoor wél over naar de concurrent, maar er werden geen klanten binnengehaald.

#### Voorbeeld: techniek sluit niet aan op techniek

In een administratief systeem worden te late betalingen automatisch doorgestuurd naar het incassobureau. Het technische proces dat opdrachten voor het incassobureau verstuurt, zoekt naar de waarde "9" in een veld "externe referentie 3". Deze waarde zou moeten worden geplaatst bij vorderingen die 2 maanden open staan. Tijdens de E2E-test bleek dat in deze gevallen geen opdrachten werden gestuurd. De waarde 9 werd door geen enkel ander technisch proces geplaatst. In de diverse ontwerpen ging men er steeds van uit dat dit in een ander technisch proces werd ondervangen.



Figuur: onvoldoende aansluiting

Oorzaken voor het niet goed aansluiten van processen en techniek zijn divers. We kunnen grofweg de volgende soorten oorzaak onderscheiden:

- E2E-processen en daadwerkelijk gebruik worden soms wel vastgelegd en beschreven, maar deze beschrijvingen zijn vaak onvolledig, onjuist, ongebruikt, niet actueel of onbegrijpelijk voor anderen.
- Wijzigingen in E2E-processen en techniek worden onvoldoende gecommuniceerd aan hen die ervan op de hoogte moeten zijn. Zo zijn functionele ontwerpen van software vaak onbegrijpelijk voor gebruikers en missen IT-professionals vaak het begrip van een domein om E2E-processen goed te begrijpen.
- De samenhang tussen technische onderdelen is zelden goed gedocumenteerd. Systeemplaten zijn wel te vinden, maar deze zijn vaak niet meer actueel of onvolledig. Documentatie over de samenhang tussen E2E-processen en techniek zijn vaak verspreid over meerdere documenten.
- Gebruikers en systeemontwerpers werken vanaf de start van een project samen om vast te stellen wat de



klantorganisatie wil en hoe IT hier in kan voorzien. In de communicatie over en weer gaat veel mis: men begrijpt vaak niet van elkaar wat de ander wel of niet begrijpt. Zo gaan gebruikers er vaak van uit dat ontwerpers hun E2E-processen wel snappen en gaan ontwerpers er vaak van uit dat datgene wat ze horen van de gebruikers alles is wat er te zeggen valt. Er wordt dan niet doorgevraagd en gecontroleerd of het beeld dat men heeft klopt of volledig is.

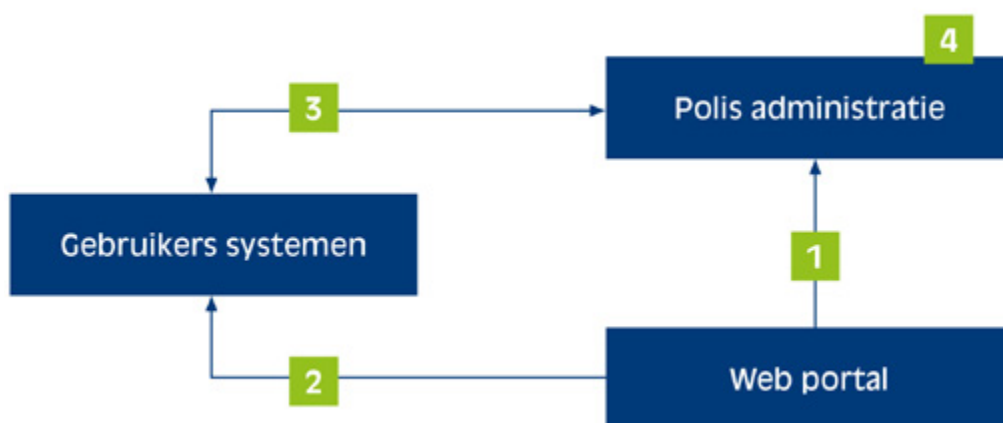
### Testen en dekking van risico's

In een unit test of een systeemtest worden de problemen uit bovenstaande voorbeelden niet gevonden. In deze testsoorten stelt men vast of een systeem is gebouwd zoals het is gespecificeerd in projectdocumenten. In een unittest of een systeemtest beoordeelt men niet of de specificaties zelf wel aansluiten op E2E-processen en andere systemen.

Een systeemintegratietest test het gezamenlijk functioneren van systemen en heeft als uitgangspunt systeem-overkoepelende ontwerpen. Hier vindt men met name problemen die voortkomen uit het niet aansluiten van techniek op techniek. Problemen in de aansluiting van E2E-processen op elkaar of E2E-processen op techniek worden minder snel gevonden, omdat dit in de regel niet de expliciete focus van een systeemintegratietest is. Van acceptatietesten wordt vaak verwacht dat deze de resterende risico's dekken. De werkelijkheid is echter anders: acceptatietesten worden meestal door verschillende groepen gebruikers uitgevoerd. Gebruikersgroepen kijken niet over de grenzen van het eigen werk heen, hebben weinig kennis van de samenhang tussen E2E-processen en techniek en hanteren (over het algemeen) een informeel testproces.

Naast deze dynamische testen moet hier nog statisch testen of reviewen worden genoemd. Dit is een goede manier om zo vroeg mogelijk fouten te vinden en dus te voorkomen. Dit geldt zeker voor de hierboven genoemde risico's. Voorwaarde is dan wel dat men de juiste kennis en tijd beschikbaar heeft om projectdocumentatie op samenhang te beoordelen en dat men de E2E-processen goed kent. Ook reviewen is niet perfect: reviewers zijn bijvoorbeeld vaak dezelfde mensen die ook aan de wieg hebben gestaan van de ontwerpen en zijn vatbaar voor dezelfde blinde vlekken die eerder tot onvolkomenheden in de samenhang hebben geleid.

Daarnaast zijn sommige problemen het eindresultaat van een lange keten van oorzaak en gevolg. In een lange keten van processtappen kunnen kleine problemen in het ene systeem een groot probleem verderop in de keten veroorzaken. Het oplossen van kleine problemen in het ene systeem kan zelfs voor grote problemen in een E2E-proces zorgen. Deze zaken worden ook in een review makkelijk gemist.



### Voorbeeld: onvoorzien probleem in een keten

Een financiële instelling beschikt over een polisadministratie, een separaat gebruikerssysteem en een webapplicatie voor verkoop. Nadat een polis is aangevraagd in de webapplicatie, wordt een bericht naar het gebruikerssysteem (1) en een bericht naar de polisadministratie (2) gestuurd. In het gebruikerssysteem wordt een offerte vastgelegd en verstuurd naar de klant. De polisadministratie legt voorlopige

polis vast (4), maar haalt daarbij eerst het offertenummer (3) in het gebruikerssysteem op. De keten was traag en men besloot tot een performanceverbetering van de polisadministratie. In de systeemintegratietest bleek toen dat polissen niet meer konden worden aangemaakt. De polisadministratie werkte nu te snel en haalde de offertegegevens in het gebruikerssysteem op, voordat deze klaar was met het aanmaken van dit document. Het technische proces in de polisadministratie werd afgesloten met een foutmelding, het gebruikerssysteem vervolgde zijn eigen technische proces. Er werd geen voorlopige polis vastgelegd maar wel een offerte verstuurd aan de klant! Dit probleem had men niet voorzien in de reviews van de ontwerpen.

---

### Conclusie:

Belangrijke risico's bevinden zich in de samenhang tussen E2E-processen en techniek en met name daar waar sprake is van grote, complexe systeemketens en lange, complexe procesketens. Deze risico's worden niet gevonden in een review, unittest, systeemtest, acceptatietest en vaak zelfs niet in een systeemintegratietest.

---

### Valideren versus verifiëren

In de vakliteratuur over testen wordt onderscheid gemaakt tussen verifiëren en valideren.

Verifiëren is vaststellen in hoeverre een product of tussenproduct overeenkomt met vastgestelde normen en eisen. Valideren is vaststellen in hoeverre een product zich staande kan houden en voldoet aan alle verwachtingen en omstandigheden.

Verifiëren geeft antwoord op de vraag: "hebben we het systeem goed gebouwd?" Valideren geeft antwoord op de vraag: "hebben we het juiste systeem gebouwd?"

Het onderscheid tussen verifiëren en valideren is belangrijk omdat veel testers en projectmanagers nog altijd menen dat verifiëren voldoende is om alle kwaliteitsrisico's te dekken. Valideren zal echter steeds belangrijker worden.

De unittest en de systeemtest zijn vormen van verifiëren. Een systeemtester beoordeelt of de software voldoet aan gedocumenteerde eisen: de specificaties. Dit geldt ook voor de systeemintegratietest: hierin zijn de interface specificaties en het globaal ontwerp de testbasis.

Een acceptatietest is zowel verificatie (op basis van vastgelegde requirements, procesbeschrijvingen en werk-instructies) als validatie (indien bijvoorbeeld gebruikers of beheerders hun dagelijks werk proberen uit te voeren op een productiegelijke testomgeving). Als validatie schiet het vaak tekort: het is te informeel en acceptatietesters missen kennis van techniek, andere E2E-processen en de samenhang daar tussen.

Een E2E-test is een systematische vorm van valideren doordat het de relaties tussen E2E-processen, techniek en de praktijk grondig onderzoekt en, gebaseerd op dit onderzoek, gestructureerd testgevallen ontwerpt.

### 3.3 Geboorte van de E2E-test

Er is een test nodig die zich richt op de samenhang tussen het daadwerkelijk gebruik en de technische infrastructuur. Deze test is de End to End (E2E) test. End to End wil zeggen: van begin tot eind.

De E2E-test heeft als basis de "echte" E2E-processen en gebeurtenissen die over de gehele technische infrastructuur lopen. Daadwerkelijke scenario's in daadwerkelijke tijdspaden worden in de E2E-test gesimuleerd. De E2E-test is daarmee de overtreffende trap van systeemintegratietesten. Een uitdaging is om de E2E-test geen nieuwe, extra aanslag te laten worden op doorlooptijden en budgetten van projecten en afdelingen.

In een aantal organisaties werken systeemintegratietesters en acceptatietesters samen. Deze nieuwe testsoort is meer dan een gecombineerde systeemintegratietest/acceptatietest. Door de proceskennis van acceptatie-

testers te combineren met de technische kennis en testproceskennis van professionele testers ontstaat een E2E-test die kritische en moeilijk te voorspellen problemen toch opspoot.

Door de nood en de praktijk gedwongen komt men dan al ver. Men moet echter steeds zelf het wiel opnieuw uitvinden en er bestaan nog steeds muren tussen business en IT en daarmee tussen systeemtesten en acceptatietesten. De voorbereiding en uitvoering van E2E-testen verschilt bovendien in de praktijk op belangrijke punten van zowel systeemintegratietesten als acceptatietesten. Daarnaast vergt een E2E-test andere vaardigheden dan een systeemintegratietest of een acceptatietest: E2E-testers zijn IT-professionals die de techniek doorzien, maar deze ook kunnen vertalen in gevolgen voor E2E-processen en andersom.

### 3.4 E2E-kennis

E2E-testen kan meer dan alleen testdoelen dienen. De informatie die E2E-testers verzamelen over E2E-processen en hun samenhang met de technische infrastructuur (de E2E-inventarisatie) kan ook voor anderen van nut zijn. Organisaties waar E2E-testteams dit soort informatie hebben verzameld, merken dat met name ontwerpers en bouwers vaak terugrijpen op de E2E-proces- en systeemplaten die door testers zijn opgesteld. Het blijkt dan dat dit soort informatie binnen andere expertisegroepen niet voorhanden is en niet wordt ontwikkeld of onderhouden.

E2E-testkennis wordt ingezet bij ontwerp vragen rond architectuur en systemen. Systeemarchitectuur wordt steeds flexibeler aangepast. Onderdelen van het systeemlandschap worden steeds gemakkelijker belegd bij andere organisaties en er wordt steeds vaker geëist dat de infrastructuur toegankelijk moet zijn voor andere organisaties. Service Oriented Architecture en Cloud Computing zijn hier voorbeelden van. Kennis van de E2E-keten speelt hierbij een belangrijke rol. De passendheid van een nieuw onderdeel in het systeemlandschap kan vanuit de door testers opgestelde E2E-inventarisatie beoordeeld worden. Daarnaast is een E2E-regressie-testset van belang om periodiek de hoogste risico's te testen na een wijziging in de systemen.

Een E2E-inventarisatie door testers kan ook een uitgangspunt zijn om E2E-processen te herstructureren. De opgebouwde kennis omtrent de E2E-processen en de techniek biedt de mogelijkheid om alternatieve processtappen door de techniek uit te tekenen en te testen.

E2E-testers vervullen vaak de rol van functioneel of zelfs technisch beheerder van systemen na de inproductie-name van nieuwe systemen in een groot systeemlandschap. De door hen tijdens de E2E-testen opgedane kennis wordt dan ingezet om de beheerorganisatie initieel of zelfs permanent te ondersteunen.

E2E-testen biedt daarmee tot voor kort onvermoede kansen voor testers. De kennis die testers zich eigen moeten maken om E2E-testen te kunnen uitvoeren betaalt zich ook op andere terreinen uit. Testen overschrijdt zo de professionele grenzen met de business, beheer en ontwerp.

# Deel 4

## Begrippen- en literatuurlijst

## Deel 4 Begrippen- en literatuurlijst

### 4.1 Begrippenlijst

<b>Actoren</b>	Deze staan aan het begin en het eind van de E2E-processen. Dit kunnen mensen (klanten, gebruikers, beheerders) zijn, maar ook andere systemen die een E2E-proces starten of een resultaat moeten krijgen. Actoren doen iets met of krijgen iets van de geautomatiseerde systemen.
<b>Beslissende factoren</b>	Ketengerelateerde, noodzakelijke onderdelen en stappen in een E2E-proces. De eerste stap komt vanuit een actor. De volgende stappen zijn acties in de systeemketen, door een andere actor, een programma of een interface. Uiteindelijk is een eindresultaat van het E2E-proces bereikt: bijvoorbeeld correspondentie, een betaling, een belastingaangifte.
<b>Business proces</b>	De processen die de diensten en producten van een organisatie mogelijk maken (bijvoorbeeld: verkopen van een polis, offreren van een polis). Een business proces is een E2E-proces indien het in zijn volledige context wordt beschouwd: dus vanuit het allereerste startpunt (bijvoorbeeld de klant) en al zijn resultaten en effecten (ook verderop in de keten of na verloop van tijd).
<b>Deelprocessen</b>	Delen van de E2E-processen die wel als apart proces (dus als samenhangende stappen met een start en een eind) zijn op te vatten. Vaak worden deelprocessen hergebruikt in meerdere E2E-processen. Bijvoorbeeld als het deelproces "uitbetalen" een onderdeel is van het E2E-proces "afsluiten polis" maar ook van het E2E-proces "schadeverwerking".
<b>E2E-board</b>	Een panel van E2E-proces- en systeemspecialisten. De leden van de E2E-board zijn parttime betrokken bij de E2E-test.
<b>E2E-inventarisatie</b>	Onderzoek op basis van systeemplaten, procesbeschrijvingen, werkinstructies, ontwerpen en interviews. Door dit onderzoek verkrijgt het E2E-testteam inzicht in de E2E-processen, de systeemketen, de deelprocessen en de samenhang tussen dezen (de E2E-keten).
<b>E2E-keten</b>	De samenhang tussen de E2E-processen en de systeemketen. Inzicht in de E2E-keten is het resultaat van de E2E-inventarisatie.
<b>E2E-proces</b>	Een proces beschouwd in zijn breedst mogelijke context: van het allereerste begin (Begin) naar (to) de uiteindelijke resultaten en effecten (Eind). Een E2E-proces start en eindigt altijd buiten de systeemketen. Een business proces is een E2E-proces als het wordt beschouwd vanuit zijn startpunt (bijvoorbeeld een klant, diens omstandigheden en verwachtingen) tot aan de mogelijke eindresultaten (voor die klant maar ook de effecten voor andere data en systemen). Business processen worden vaak niet als E2E-proces beschouwd, maar alleen voor zover ze relevant zijn voor de eigen organisatie of het actuele project. Zo

kan het klantperspectief of het belang voor andere organisaties en de eigen infrastructuur onderbelicht raken.

---

**E2E-testcentrum**

Organisatieonderdeel waar E2E-kennis wordt bewaard en uitgedragen, testomgevingen worden beheerd en wordt aangesloten bij releasemanagement.

---

**Gebruikersacceptatietest**

Test die wordt uitgevoerd door gebruikers of onder verantwoordelijkheid van de accepterende gebruikersafdeling. Dit kunnen zowel klanten, gebruikers of beheerders zijn. Iedereen die moet werken met het betreffende product moet in principe middels een gebruikersacceptatietest vaststellen in hoeverre aan zijn of haar acceptatiecriteria is voldoen. De testbasis bestaat uit requirements, werkinstructies, handleidingen of E2E-processen.

---

**Interfacetest**

De test van een koppeling tussen systemen. Typische aspecten die in een interfacetest worden gedekt zijn syntaxis (vulling van velden) en semantiek (betekenis van velden). Een interfacetest heeft als testbasis het functionele en technisch ontwerp van de betreffende interface. Vaak worden interfacetesten als onderdeel van een systeemtest uitgevoerd.

---

**Kritische factoren**

De omstandigheden, varianten en instellingen die ook geraakt worden of die van invloed kunnen zijn op het E2E-proces. Denk hierbij aan afwijkende actoren (andere groepen klanten), alternatieve paden, verschillende producten, tijdelijke toestanden van een systeem zoals een uitrol, een maandovergang, toestanden van het gegeven of de klant (incassotrajecten, product beëindigd). Historie moet hier ook worden genoemd: in welke stadia in de levenscyclus van een gegeven moeten er testen plaatsvinden en wat voor invloed kan hier van uit gaan? Denk hierbij aan polissen of klanten met veel mutaties, waaronder mutaties met terugwerkende kracht, of polissen die via migraties uit andere systemen zijn overgenomen en dus niet "volgens de regels" zijn ingevoerd. De kritische factoren moeten voorgelegd en bevestigd worden bij de procesexperts.

---

**Proces**

Geheel van opeenvolgende, samenhangende gebeurtenissen. Een proces wordt gestart (door iets of iemand), maakt gebruik van input (zoals gegevens) en van middelen (zoals functies, systemen, hardware) en heeft resultaten (output en effecten). Processen kunnen worden onderverdeeld in business processen, functionele processen, technische processen en delen van deze processen.

---

**Systeemintegratietest**

Testsoort waarin meerdere systemen voor wat betreft hun gezamenlijk functioneren worden getest. Focus van de systeemintegratietest zijn systeemoverstijgende functies. De testbasis zijn globale ontwerpen, functionele ontwerpen van de systemen en functionele beschrijvingen van de betreffende keten. In feite is dit de systeemtest van een verzameling systemen. De systeemintegratietest behoort dan te worden georganiseerd en uitgevoerd door de ontwikkelorganisatie(s) of het project.

---

**Systeemketen**

Geheel van samenhangende systemen en componenten, oftewel: de hardware en de software. De systeemketen ondersteunt E2E-processen.

<b>Systeemplaats</b>	Grafische weergave van de systeemketen.
<b>Systeemtest</b>	De eerste volledige test van een op zichzelf staand systeem. Dit is meestal een aparte testsoort. In de systeemtest wordt een volledige test uitgevoerd van de functionaliteit die door het systeem wordt ondersteund. De testbasis van de systeemtest is daarom de functionele ontwerpen die ten grondslag hebben gelegen aan de bouw van het systeem. De bouw zelf baseert zich meestal op een technisch ontwerp welke een vertaling is van het functionele ontwerp. Naast functionaliteit kunnen ook andere aspecten onderwerp van test zijn, zoals efficiency of bruikbaarheid. Technische processen: processen die zich afspelen in een systeem. Vaak zijn dit afzonderlijke systeemdelen die kunnen worden gestart en beëindigd, zoals achtergrondprogramma's of processen die op afroep een bestand of een rapport produceren.
<b>Testcluster</b>	Testgevallen behorende bij een E2E-proces. Eventueel kunnen er meerdere testclusters per E2E-proces zijn als verschillende testtypen nodig zijn die niet gezamenlijk kunnen worden voorbereid en uitgevoerd (bijvoorbeeld een functioneel testcluster én een performancetestcluster van het E2E-proces "aanvragen en afsluiten").
<b>Testpuntanalyse (TPA)</b>	Formele plantechiek die voor een groot deel is gebaseerd op Functiepuntanalyse (FPA).
<b>Testsoort</b>	Een verzameling testactiviteiten die qua organisatie en fasering afzonderlijk is van andere verzamelingen activiteiten. In het V-model worden bijvoorbeeld de systeemtest en de acceptatietest als afzonderlijke testsoorten voorgesteld.
<b>Testtype</b>	Testactiviteiten gericht op het testen van één bepaald aspect. Bijvoorbeeld: performancetest, securitytest. Een testtype wordt niet per definitie in één fase uitgevoerd maar kan eventueel over verschillende testsoorten worden uitgesmeerd.
<b>Validatie</b>	Controleren of iets waardevol is, dat wil zeggen: of het in de praktijk voldoet. Testen als validatie betekent: het product zo veel mogelijk in als ware het productieomstandigheden laten functioneren.
<b>Verificatie</b>	Controleren of iets klopt. Testen als verificatie betekent: vaststellen of software functioneert overeenkomstig hetgeen vastgelegd is (in een functioneel of technisch ontwerp).

## 4.2 Literatuurlijst

1. [Smit 09] Testen van ketens met TMap NEXT®; R. Smit, R. Baarda; 2009
2. [Tsai 01] Scenario based Functional Regression Testing (Tsai e.a.)
3. [Tsai 02] End-to-End Integration Testing Design (Tsai e.a.)
4. [Koomen 04] TMap Test Topics; Koomen, Baarda; 2004
5. [Koomen 06] TMap Next; Koomen, Van der Aalst, Broekman, Vroon; 2006



# Deel 5

## E2E-inventarisatie



## Deel 5 E2E-inventarisatie

### 5.1 Inleiding E2E-inventarisatie

Vanuit testperspectief zijn er drie opeenvolgende logische stappen van integratie. Deze corresponderen met betreffende testsoorten:

1. Interface: de koppeling tussen twee systemen.
2. Systeemintegratie: het samenwerken van meerdere systemen
3. End to End (E2E): volledige processen over een complete keten van systemen heen.

De interface test en systeemintegratietest hebben als testbasis de specificaties van systemen (van de interface zelf in het geval van een interfacetest en van de gezamenlijke specificaties in geval van een systeemintegratietest. Een E2E-test heeft als doel vast te stellen wat de kwaliteit van de ondersteuning van E2E-processen is door de gehele systeemketen. Hier zijn specificaties niet voldoende maar moeten ook de E2E-processen (werkprocessen, handleidingen, instructies, werking van systemen verderop in de keten en daadwerkelijk gebruik) worden mee gewogen.

Daarom bestaat voor een E2E-test de testbasis uit een, voor een groot deel door het E2E-testteam zelf uit te voeren, inventarisatie van de E2E-processen en hun samenhang met de systemen.

Het resultaat van de E2E-inventarisatie is de testbasis voor de E2E-test en dient al op hoofdlijnen tijdens de planningsfase te worden uitgewerkt. In volgende fasen wordt de inventarisatie van de nodige detaillering voorzien en wordt er veelvuldig gebruik van gemaakt, bijvoorbeeld tijdens de risicoanalyse, het testontwerp en het analyseren van testresultaten.

De volgende begrippen zijn van belang als uitgangspunt:

**E2E-processen:** het uiteindelijke doel van de gehele systeemketen. Dit kunnen werkprocessen zijn, diensten die worden geleverd aan klanten, informatie die moet worden geleverd aan andere organisaties.

**Deelprocessen:** onderdelen van E2E-processen. Het proces "Controleren van kredietwaardigheid" is een deelproces van het E2E-proces "Afsluiten polis".

**E2E-keten:** de keten van systemen die een E2E-proces ondersteunt en die gevolgen kan ondervinden van een E2E-proces. De E2E-keten bestaat daarom niet per definitie uit alle systemen in het systeemlandschap van een organisatie, maar betreft de systemen die van belang zijn voor het betreffende E2E-proces en de systemen die gevolgen kunnen ondervinden van het E2E-proces (zoals performance, security, kwaliteit van gegevens).

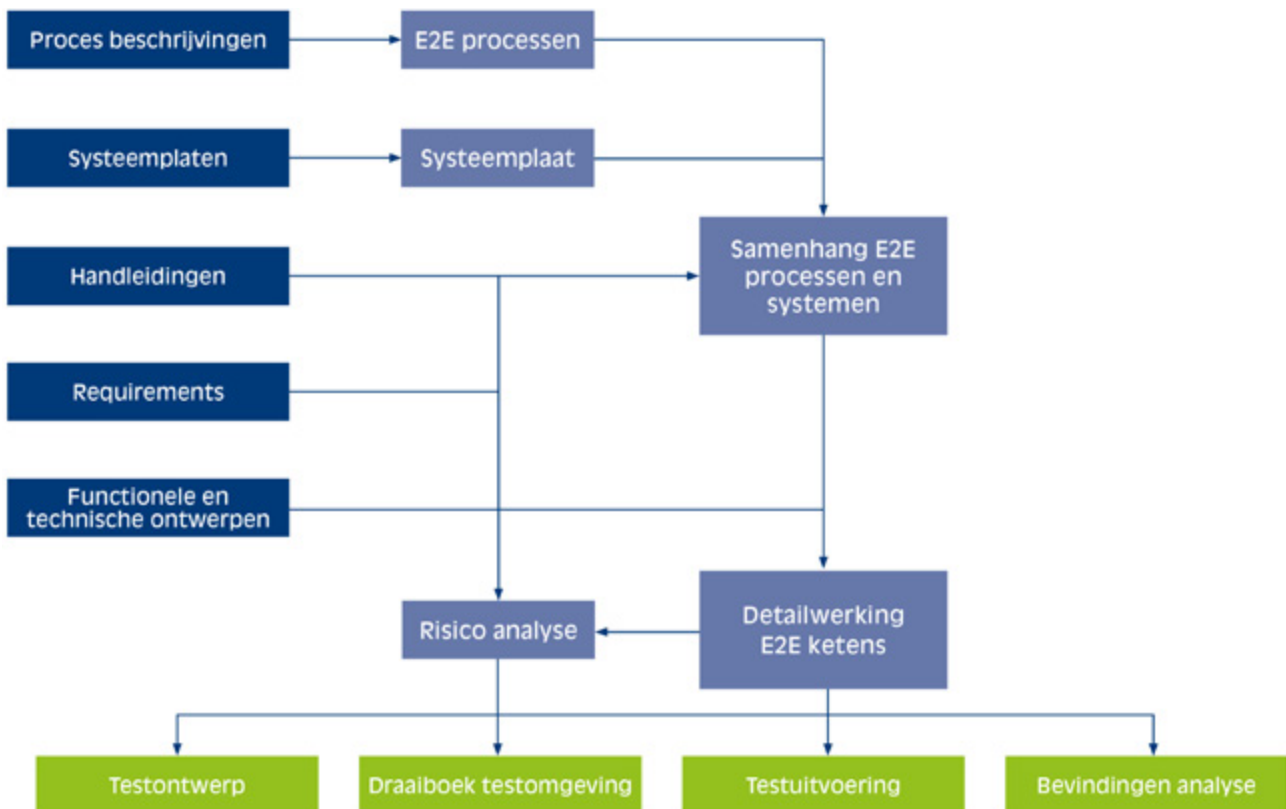
Er zijn goede redenen om E2E-processen gestructureerd te analyseren in hun samenhang met systemen. Dit dient niet alleen de E2E-testen. Binnen de IT denkt men namelijk vaak eenzijdig vanuit de technische infrastructuur (het systeemlandschap) of vastgelegde functionele eisen. In dit perspectief kan het oorspronkelijke doel van het systeemlandschap naar de achtergrond verdwijnen. Tijdens de gebruikersacceptatietesten krijgen gebruikers dan bijvoorbeeld te horen, wanneer ze een vanuit hun E2E-proces op zich terecht bevinding indienen, dat het systeem nu eenmaal niet zo werkt of dat het systeem conform specificaties functioneert en dat de bevinding dus niet valide zou zijn.

Daarnaast ontbreekt bij gebruikers vaak het begrip van de mogelijkheden en onmogelijkheden van de technische infrastructuur om processen te ondersteunen. Zo kunnen te hoge verwachtingen ontstaan of beseft men niet wat de gevolgen of risico's zijn van voorgestelde wijzigingen.

Een E2E-inventarisatie kan in deze gevallen voor alle partijen (gebruikers, ontwerpers, bouwers, testers) ondersteuning en duidelijkheid bieden.

Om tot een E2E-inventarisatie te komen moeten de volgende activiteiten worden uitgevoerd:

1. In kaart brengen E2E-processen.
2. In kaart brengen systeemlandschap.
3. Analyseren samenhang E2E-proces en systeemlandschap (de E2E-keten).
4. Detailleren van de werking van de E2E-keten per E2E-proces.

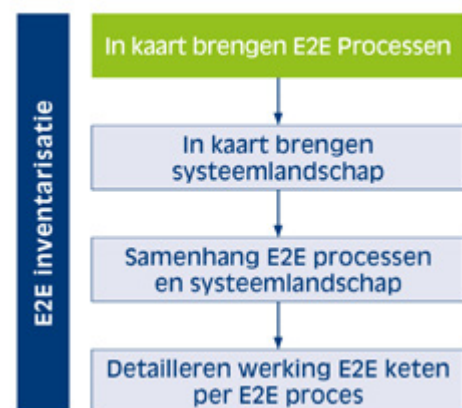


Figuur: E2E-inventarisatie in samenhang

In de volgende paragrafen worden de 4 stappen nader uitgelegd.

## 5.2 In kaart brengen E2E-processen

Met E2E-processen wordt hier bedoeld: logisch opeenvolgende en samenhangende activiteiten die samen een dienst of een taak volbrengen. Als voorbeeld wordt hier een verzekeringsmaatschappij genomen. Deze organisatie bestaat bij de gratie van het kunnen leveren van diensten aan klanten. Dit zijn polissen. Voor wat betreft het kunnen verkopen en onderhouden van polissen zijn diverse E2E-processen nodig. E2E-processen gericht op de primaire taken van een organisatie worden primaire E2E-processen genoemd. Daarnaast zijn er ook secundaire E2E-processen, deze hebben bijvoorbeeld te maken met personeel of huisvesting van de verzekeringsmaatschappij.



De polisadministratie is deel van het primaire proces van een verzekeringsmaatschappij. Om inzichtelijk te maken welke E2E-processen hiervan gebruik maken, worden alle mogelijke samenhangende handelingen gegroepeerd.

Vragen die men zich dan moet stellen zijn:

- Waar begint een polis?
- Wat kan een polis allemaal overkomen?
- Wanneer eindigt een polis?

Zo verschijnen de volgende (primaire) E2E-processen:

- Offreren (een offerte aanbieden).
- Aanvragen.
- Betalen.
- Muteren.
- Afhandelen schade.
- Verlengen (of: prolongeren).
- Beëindigen (of: expireren).

### Hoofdproces of subprocesses?

Stelregel bij het opdelen van hoofdprocessen en subprocessen is dat processen zich significant onderscheiden voor wat betreft eindresultaat, functie of verloop door een systeemketen.

Indien de E2E-keten bij het aanvragen van een polis via internet vergeleken met het aanvragen via een tussenpersoon voldoende verschilt, kan men ervoor kunnen kiezen het E2E-proces op te delen in twee subprocessen en testclusters. Het hoofdproces is dan "Aanvragen", de subprocessen daar binnen zijn "Aanvragen via internet" en "Aanvragen via een tussenpersoon".

Van elk E2E-proces moet het globale verloop worden opgetekend: wie wil wat bereiken en welke logische stappen leiden tot het gewenste resultaat? In het geval van een aanvraag via internet is het doel de klant in staat te stellen snel het product te vinden dat hij zoekt en vervolgens hem de polis te laten afsluiten. De stappen zijn dan: openen site, presenteren producten, uitzoeken van opties, keuze maken, afsluiten, controle aanvraag, polis opslaan, contract opsturen en premie betalen. Het maken van een simpele procesflow vergemakkelijkt de analyse.



Figuur: E2E-proces voorbeeld, aanvragen product

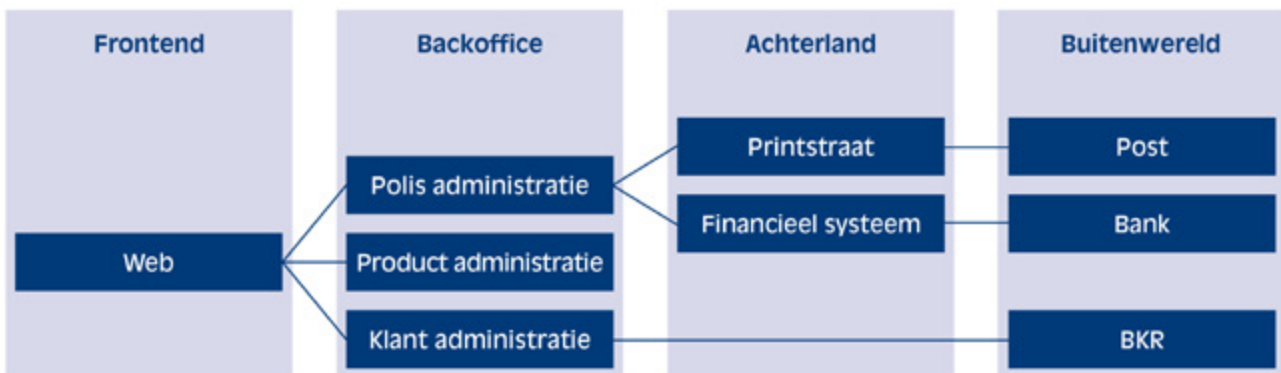
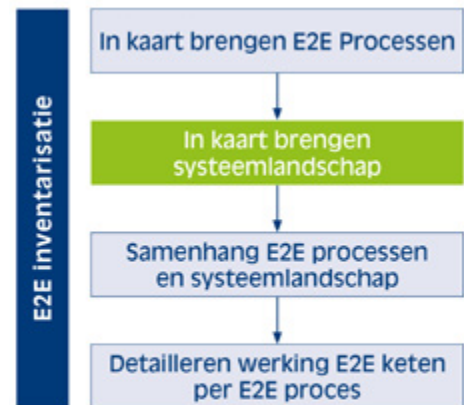
Stappen in een E2E-proces worden, ten minste deels, belegd bij systemen. Een klant bekijkt bijvoorbeeld het aanbod op een website. Een deel van het proces is dan belegd in een geautomatiseerd systeem waarin productinformatie wordt opgehaald uit het productsysteem en getoond aan de klant via internet. De klant kan dan een polis afsluiten middels één druk op de knop. Het afsluiten is dan belegd in een wisselwerking tussen de website en de polisadministratie. Als onderdeel daarvan wordt de aanvraag gecontroleerd: hier gaat informatie van de polisadministratie naar het klantsysteem en het Bureau Krediet Registratie. Uiteindelijk wordt er een polis vastgelegd in het polissysteem. Hier vanuit wordt een bericht gestuurd naar de website waardoor de klant weet dat de transactie wordt afgehandeld. Vervolgens wordt door het postsysteem het contract uitgeprint en opgestuurd aan de klant. De financiële afhandeling wordt belegd bij de financiële systemen en uiteindelijk bij de bank en de bankrekening van de klant.

Informatie over E2E-processen kan worden gehaald uit procesbeschrijvingen, globale ontwerpen, handleidingen, werkinstructies en gesprekken met procesdeskundigen zoals ontwerpers, functioneel beheerders of ervaren gebruikers. Het in kaart brengen van de processen is gereed zodra met de deskundigen is vastgesteld dat alle belangrijke processen zijn geïdentificeerd en deze van start tot einde zijn geanalyseerd.

### 5.3 In kaart brengen systeemlandschap

Een onmisbaar bestanddeel van de testbasis van de E2E-test is het overzicht van alle betrokken systemen en hun onderlinge verbindingen. Het resultaat van het in kaart brengen is een grafisch schema van het systeemlandschap (zie onderstaande figuur voor een simpel voorbeeld). Meestal is de systeemplaat al voorhanden in de organisatie of het project.

Het landschap moet compleet worden getekend, maar in de praktijk is dit vaak niet haalbaar. Ketens van systemen kunnen namelijk worden uitgebreid tot de gehele wereld, zeker waar connecties met het internet of andere organisaties bestaan. Daarnaast is een grote variatie mogelijk in diepgang van uitwerken van interfaces en systemen. Om niet te verdrinken moeten daarbij redelijkheid en aannemelijke risico's de doorslag geven. Deelprocessen en connecties waarvan men kan stellen dat deze weinig tot geen risico geven, kunnen globaal worden weergegeven of eventueel buiten beschouwing worden gelaten.

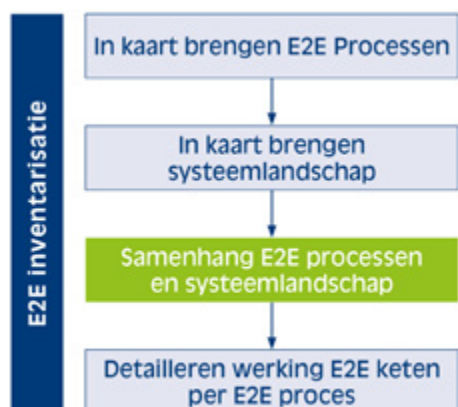


Figuur: Systeemlandschap voorbeeld

Noot: met "BKR" wordt het Bureau Krediet Registratie bedoeld.

### 5.4 Analyseren samenhang E2E-proces en systeemlandschap

Nu moet de samenhang tussen de E2E-processen en het systeemlandschap worden geanalyseerd. Het doel van deze activiteit is het inzichtelijk maken van de processtappen in samenhang met de onderdelen in het systeemlandschap. Dit is van belang voor het ontwerpen van testgevallen en vaststellen van controles maar ook om vast te stellen welke testomgeving moet worden opgezet. Voor de testuitvoering zal namelijk niet altijd een compleet systeemlandschap hoeven te worden opgetuigd. Indien niet alle onderdelen al beschikbaar zijn is het van belang te kunnen vaststellen welke E2E-testen kunnen worden uitgevoerd en welke niet. De voor een specifieke E2E-test benodigde systemen noemen we de E2E-keten van het betreffende E2E-proces.



De volgende stappen worden onderscheiden:

- Het E2E-proces wordt in logische onderdelen opgebroken (actoren, stappen, resultaten).
- De stappen worden geplot op de systeemonderdelen (beslissende factoren).
- Tot slot worden de omstandigheden geanalyseerd die ook nog van invloed kunnen zijn op de E2E-keten (kritische factoren).

## Actoren, stappen, resultaten

Processen zijn samenhangende acties (stappen) vanuit mensen of systemen (de actoren), die worden uitgevoerd in een bepaalde volgorde en met behulp van bepaalde instrumenten, teneinde bepaalde resultaten te bereiken.



Een actor kan een klant zijn of een gebruiker, maar ook een ander systeem vanuit een andere organisatie kan als actor worden aangemerkt. De actor is het beginpunt van een E2E-proces, maar ook de plek waar de zin en betekenis van het proces gevonden kan worden: de klant vraagt een polis aan, maar wat verwacht hij terug en wanneer? De actor start een E2E-proces op, hij voert zelf stappen uit (zoals het invoeren van persoonsgegevens) en er worden stappen in het vervolgtraject uitgevoerd door de geautomatiseerde systemen of andere actoren zoals gebruikers. Uiteindelijk leiden de stappen tot resultaten.

In het gegeven voorbeeld van een klant die via internet een polis afsluit, is de actor de klant.

De stappen in het voorbeeld zijn:

- Klant vraagt productinformatie op.
- Klant vraagt product aan.
- Aanvraag wordt gecontroleerd in klantsysteem.
- Klant wordt gecontroleerd bij Bureau Krediet Registratie.
- Contract wordt aangemaakt en verstuurd.
- De premiebetaling (incasso) wordt opgestart.
- Het resultaat van de premiebetaling wordt verwerkt.

De resultaten zijn:

- Een nieuwe klant is aangemaakt.
- Een contract wordt verstuurd.
- Premiebetalingen (incasso of acceptgiro's) worden gegenereerd.
- Resultaat vanuit de bank wordt verwerkt.

Een alternatief resultaat (op het scherm) is een bericht met de mededeling dat de aanvraag niet geslaagd is.

## Beslissende factoren

In het geval van een E2E-keten zijn de stappen de handelingen door actoren en de achtereenvolgende noodzakelijke onderdelen in het systeemlandschap, zoals interfaces en deelsystemen. Deze stappen, handelingen en systeemonderdelen worden hier "beslissende factoren" genoemd. Het zijn de schakelingen waaruit een keten is opgebouwd.

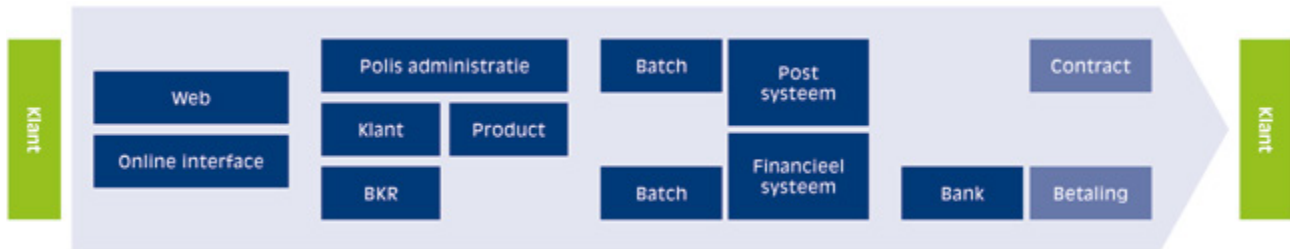
Om de beslissende factoren te vinden, moet van de stappen in het E2E-proces worden vastgesteld van welke systemen gebruik wordt gemaakt: waar in het systeemlandschap wordt de stap uitgevoerd en welke systemen zijn daarbij betrokken? Indien een klant productinformatie opvraagt dan opent hij een browser. Vanuit de site van het bedrijf wordt productinformatie opgehaald en weergegeven in de browser op het scherm van de klant. De beslissende factoren zijn dan: het scherm, de browser, de site, het productsysteem.

Per stap zijn de beslissende factoren in het voorbeeld:

- Klant vraagt productinformatie op: web > productadministratie > web.
- Klant vraagt product aan: web > polisadministratie.
- Aanvraag wordt gecontroleerd in klantsysteem: polisadministratie > klantsysteem.
- Klant wordt gecontroleerd bij Bureau Krediet Registratie: polisadministratie > BKR > polisadministratie > web.

- Contract wordt aangemaakt en verstuurd: polisadministratie > postsysteem > bezorgdienst.
- De premiebetaling (incasso) wordt opgestart: polisadministratie > financieel systeem > bank.
- Resultaat premieverwerking: bank > financieel systeem.

Het E2E-proces, met daarin de actoren, beslissende factoren en resultaten, ziet er als volgt uit:



*Figuur: Actoren, beslissende factoren, resultaten*

Er kunnen diverse detailniveaus worden gekozen: worden alleen de afzonderlijke systemen benoemd of ook alle interfaces, systeemonderdelen, tussenresultaten en bestanden? In het geval van een complexe en lange E2E-keten verdient het aanbeveling om met een globale insteek te beginnen: alleen systemen zoals in het getoonde voorbeeld en daar vanuit, stapsgewijs, meer detail toevoegen of de detailinformatie apart opslaan.

### Kritische factoren

Tot slot moet nog een plek worden ingeruimd voor alle zaken die het E2E-proces in zijn afhandeling kunnen beïnvloeden, zoals systeemtoestanden, momenten in de tijd, omstandigheden van de klant of de producten. Deze zaken worden hier "kritische factoren" genoemd. Kritische factoren zijn geen systeemonderdelen en zelf niet noodzakelijk om een basispad door een E2E-keten of E2E-proces te doorlopen, maar zij kunnen wel van invloed zijn op de wijze waarop dit pad afgelegd wordt en moeten daarom worden meegewogen in de bepaling van testgevallen.



*Figuur: Kritische factoren*

Het kiezen van kritische factoren gebeurt op basis van relevantie voor de E2E-keten. In dit voorbeeld zijn de kritische factoren de verschillende producten die kunnen worden gekozen door de klant, momenten die van invloed kunnen zijn op het E2E-proces (bijvoorbeeld een betaling die na een maand- of jaarovergang plaatsvindt) of klantomstandigheden (bijvoorbeeld indien de klant te laat betaalt, overlijdt of meerdere producten heeft). Het kan bijvoorbeeld relevant zijn om twee verschillende producten te kiezen indien deze onderling verschillen in de afhandeling in de keten. De één kent bijvoorbeeld een jaarlijkse betaling en de ander een maandelijkse, of er wordt verschillende correspondentie geproduceerd. Momenten kunnen ook relevant zijn: bij jaarovergangen vinden er aparte afrekeningen plaats, worden jaaroverzichten gegenereerd of wordt een fiscaal jaar afgesloten en een nieuwe gestart.

Klantomstandigheden ("Klant") kunnen van belang zijn: betalingsachterstanden kunnen leiden tot aparte informatiestromen in de keten en uiteindelijk tot het automatisch beëindigen van producten. Overlijden is een E2E-proces dat lopende deelprocessen, zoals betalingen en correspondentie, moet beëindigen.

Van de kritische factoren moet worden vastgesteld in welke onderlinge combinatie ze moeten worden getest. Combinaties van producten kunnen relevant zijn in combinatie met een jaarovergang of kunnen leiden tot kortingen, samengevoegde correspondentie, samengevoegde betalingen en dergelijke.



Kritische factoren kunnen ook van belang zijn voor bepaalde typen testen, door de mate waarin het systeem wordt belast: dit kan relevant zijn voor het testen van de efficiency of de stabiliteit.

## 5.5 Detailleren van de werking van de E2E-keten per E2E-proces

De inventarisatie die in de vorige stap is uitgevoerd, is voldoende om logische testgevallen mee te ontwerpen. In de inventarisatie zijn de actoren, beslissende factoren, kritische factoren en de resultaten vastgesteld per E2E-proces. De actoren, beslissende factoren, kritische factoren en resultaten hebben de rol die condities en beslispunten in een functionele test spelen: in het testontwerp worden testgevallen ontworpen om deze onderdelen in onderlinge combinatie te dekken.

Om fysieke testgevallen uit te schrijven en daarbij testomgevingen te beheren, draaiboeken te maken en analyses van bevindingen uit te kunnen voeren, is een diepere analyse nodig van de werking van de E2E-ketens.



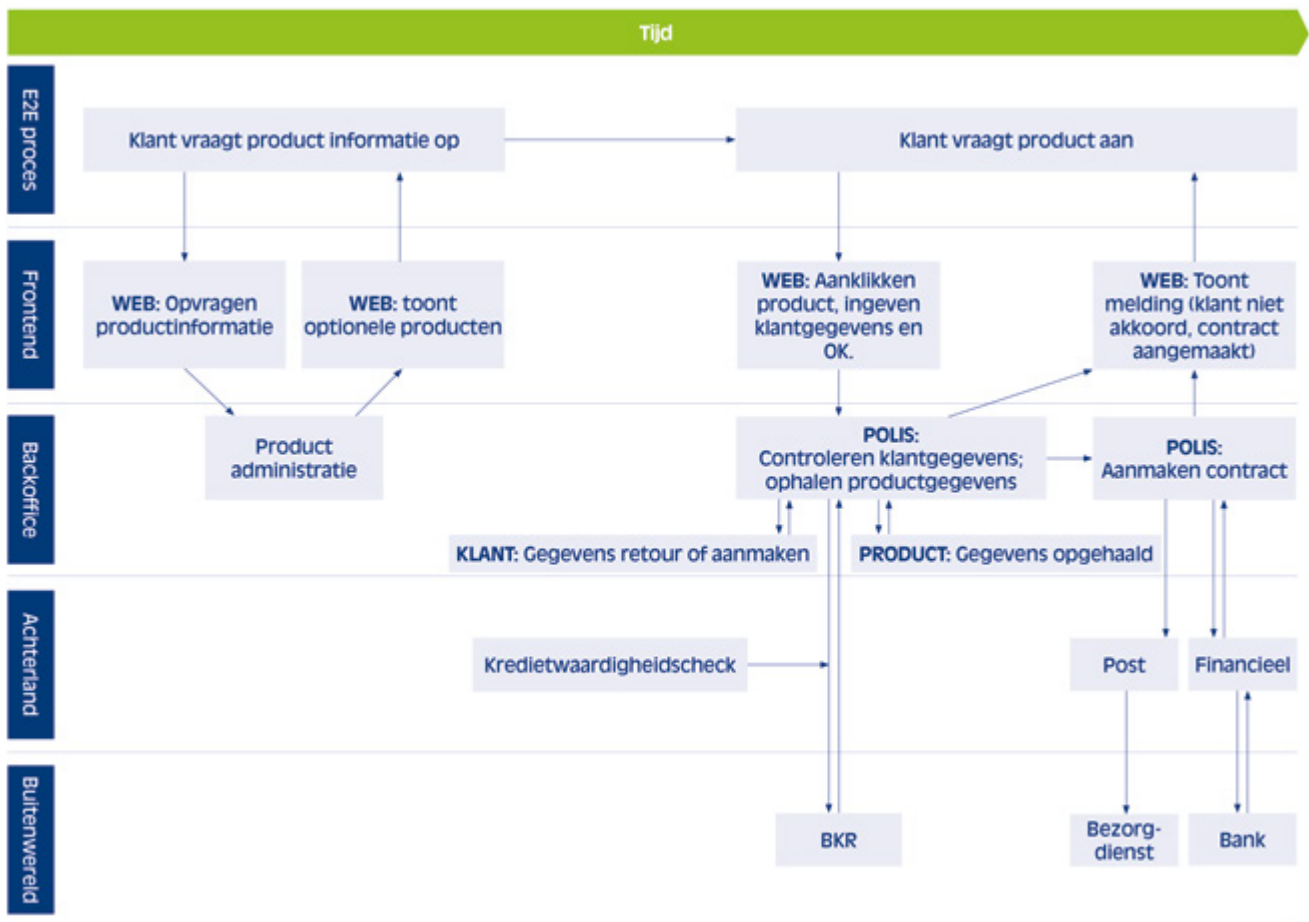
In deze analyse moeten de beslissende factoren in hun werking en onderlinge samenhang verder worden onderzocht.

Het gaat dan om:

- De exacte functionele en technische stappen die door de E2E-keten worden genomen.
- De bepaling van onderlinge afhankelijkheden van de technische en functionele stappen.
- De tussenresultaten per stap (zoals bestanden die tussen systemen worden verstuurd). De werking van interfaces.

De detaillering leidt tot een procesflow waarin de processtappen, functionaliteit en technische onderdelen tot uitdrukking komen. Een dergelijk overzicht is vaak, maar zeker niet altijd of volledig, al gemaakt door informatieanalisten of ontwerpers. Hier kan het E2E-testteam dan veel van de informatie betrekken. Toch zal het E2E-testteam dit E2E-proces zelf moeten nalopen en uittekenen om belangrijke ketenontwerpfouten in een vroeg stadium op het spoor te komen en om informatie in huis te hebben om testomgevingen in te richten, testgevallen voor te bereiden en uit te voeren en bevindingen te analyseren.

De informatie van de detaillering kan op verschillende manieren worden gepresenteerd. De figuur hieronder toont een veel gebruikte manier om een E2E-proces in een E2E-keten in detail uit te tekenen. Hierbij worden het E2E-proces, de processtappen, de functionele stappen en de verschillende functionele en technische lagen in onderlinge verband en in chronologische volgorde getoond. Wat hier nog ontbreekt, is informatie over de werking van de interfaces en de daarbij behorende tussenproducten. Hoe wordt bijvoorbeeld de aanvraag van een polis vanuit het web naar de polisadministratie gedaan: is dit via een achtergrondprogramma of online en wat voor bestandsuitwisseling vindt hier plaats? Deze vragen moeten in dit geval nog worden uitgewerkt, in de tekening zelf of in een bijlage.



Figuur: Voorbeeld van de samenhang tussen E2E-proces en systemen





# Deel 6

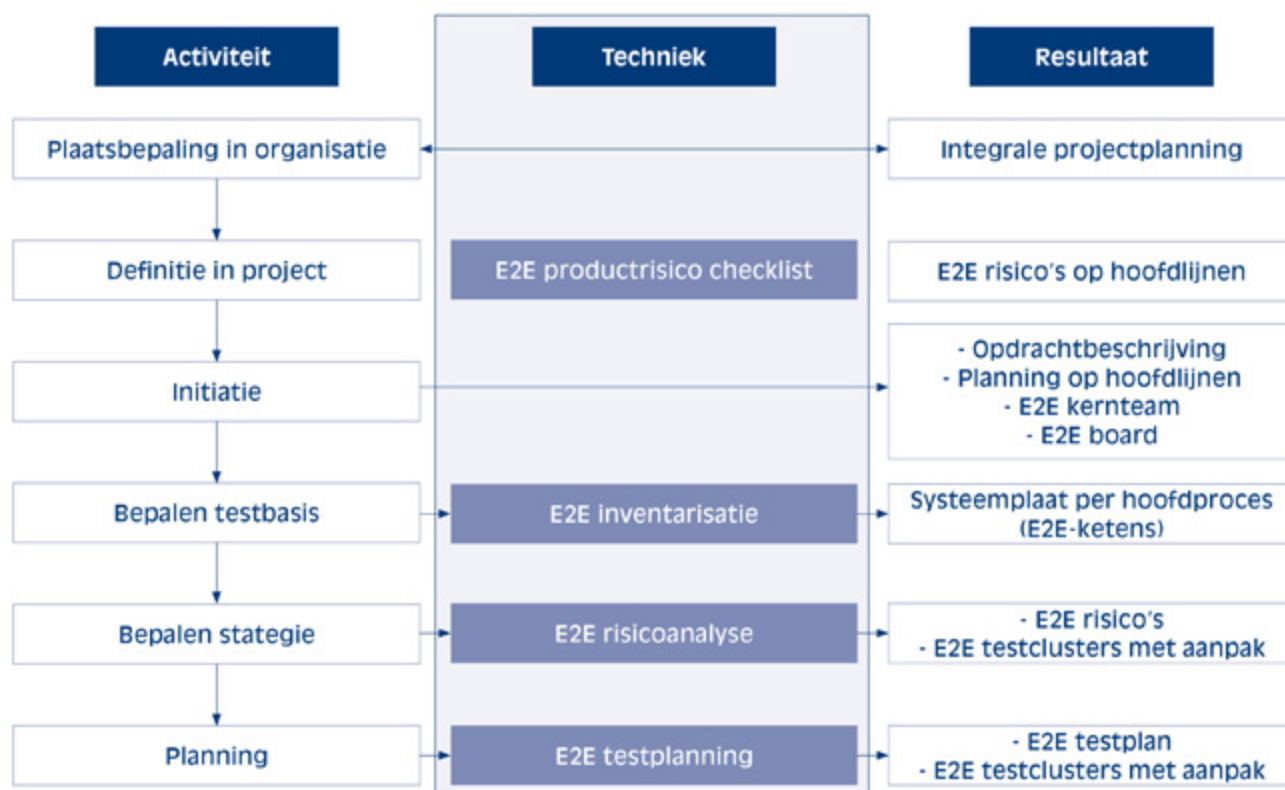
## E2E-teststrategie

## Deel 6 E2E-teststrategie

### 6.1 Inleiding E2E-teststrategie

In dit deel wordt het opstellen van de E2E-teststrategie uitgewerkt. Het analyseren van productrisico's wordt hier als onderdeel van de teststrategie gezien, hoewel dat officieel een testoverstijgende projectactiviteit is. In de praktijk zijn het vaak testmanagers die productrisicoanalyses initiëren en organiseren. In het vervolg van de tekst zal met PRA de productrisicoanalyse worden bedoeld.

Belangrijk voor de E2E-teststrategie is de E2E-inventarisatie. Hierin worden de E2E-processen vastgesteld en geanalyseerd. Resultaat van de E2E-inventarisatie is inzicht in de E2E-ketens: de handelingen, systemen, systeemonderdelen, deelprocessen en resultaten per E2E-proces. Deze inventarisatie wordt het E2E-team zelf uitgevoerd. De E2E-inventarisatie zal eerst globaal worden opgezet, waarna er steeds meer detaillering kan worden toegevoegd, bijvoorbeeld naar aanleiding van PRA-sessies. Hoewel de E2E-inventarisatie initieel en in globale vorm voorafgaat aan het opzetten van de E2E-teststrategie zullen E2E-inventarisatie en E2E-teststrategie parallel worden opgesteld.



Figuur: Overzicht activiteiten E2E-test Planning en beheer

De E2E-teststrategie bevat het opstellen van E2E-risico's op hoofdlijnen, het uitvoeren van de E2E-risicoanalyse en het opstellen van E2E-testclusters met testaanpak.

### 6.2 E2E-Productrisicoanalyse

Zoals vrijwel elke testmethode tegenwoordig predikt, is testen in de eerste plaats een risicobeperkende maatregel: "No risk, no test". Om te weten wat en hoe er getest moet worden is een risicoanalyse belangrijk. Diverse testmethodes geven handreikingen voor het uitvoeren van een risicoanalyse. Het uitvoeren van een productrisicoanalyse wordt desondanks meestal als lastig ervaren: er moeten diverse medewerkers worden bewogen om deel te nemen en de veelsoortige informatie moet worden vertaald in informatie waar een testaanpak op

kan worden gebaseerd. Als een risicoanalyse te formeel wordt beschreven ontstaat snel het risico dat men deze procedure gedachteloos gaat uitvoeren. Gevolg hiervan kan zijn dat het analyseren van de risico's (dat wil zeggen: goed nadenken over risico's) wegzakt.

De volgende uitgangspunten gelden voor een (E2E) productrisicoanalyse:

- Bereid een E2E-PRA goed voor: een E2E-inventarisatie is een goede input voor de risicoanalyse.
- De E2E-testmanager schat vooraf in wat de hoge risico's zijn en wat voor type risico's dit zijn (in termen van kwaliteitsattributen): is efficiency bijvoorbeeld een hoog risico? Gebruik voor deze inschatting de checklist E2E-risico's.
- Een E2E-PRA kan niet in één bijeenkomst worden afgedekt. Deel de E2E-keten op in verschillende gebieden (bijvoorbeeld per hoofdproces of samenhangende hoofdprocessen), maar waak ervoor dat dit niet tientallen gebieden worden. Per gebied moeten de risico's worden doorgesproken in korte bijeenkomsten (maximaal 2 uur) of individuele gesprekken.
- Bespreek de risico's in termen van echte scenario's. Dus niet alleen maar in termen van een abstract kwaliteitsattribuut ("Efficiency is een hoog risico") maar in termen van welke concrete gebeurtenis of omstandigheid tot welk concreet ongewenst effect kan leiden. Bijvoorbeeld: "als de indexering op de database niet goed werkt, kan een aanvraag van het product leiden tot een wachttijd van meerdere minuten voor de klant. De klant kan dan besluiten bij de concurrent een product te kopen".
- Maak er geen wetenschap van, met wiskundige berekeningen tot achter de komma. Het is voldoende om te onderkennen wat de hoogste risico's (veel testvaardigheid) en de lagere risico's (weinig testvaardigheid) zijn.
- Laat zien wat er met de risico's gebeurt: laat ze terugkomen in plannen, voortgangsrapporten en eindrapporten. Als er niets met risico's wordt gedaan, zal men in de toekomst minder geneigd zijn deel te nemen aan risicoanalyses.

Het kan voorkomen dat er elders (niet vanuit E2E-testperspectief) ook risicoanalyses worden uitgevoerd. Indien daar E2E-risico's worden onderkend, zal het testteam met relevante experts deze risico's verder moeten analyseren. Wordt er binnen het project nog geen PRA uitgevoerd of is de risicoanalyse te globaal of te beperkt en is er reden om aan te nemen dat er nog onontdekte E2E-risico's zijn, dan zal er vanuit het E2E-testteam een E2E-PRA moeten worden gestart. In een grote en complexe E2E-keten zal de E2E-PRA waarschijnlijk opgesplitst moeten worden op basis van E2E-processen en/of delen van het systeemlandschap. Dit om de PRA effectief en beheersbaar te houden.

Het is aan de E2E-testmanager om risico's te verzamelen en daar één consistent geheel van te maken. Hier leidt hij de teststrategie vervolgens uit af.

De volgende logische stappen worden uitgevoerd in een E2E-PRA:

- Globale risicoanalyse.
- Bepalen aanpak E2E-PRA. Identificeren van benodigde risicoanalyses. Kan worden volstaan met één E2E-PRA of moet dit worden opgebroken?
- Vaststellen van de testclusters voor wat betreft de E2E-test. Vanuit de E2E-processen worden testclusters benoemd en daarbinnen de testitems.
- Uitvoeren PRA: mogelijke fouten of ongewenste gevolgen identificeren en inschatten van kansen van optreden.
- Teststrategie op hoofdlijnen definiëren.

De stappen zullen niet altijd in deze volgorde hoeven te worden uitgevoerd. Het vaststellen van de testclusters zou bijvoorbeeld ook tijdens of vooraf aan de PRA kunnen worden uitgevoerd, of als onderdeel van de E2E-inventarisatie. De volgorde hangt af van de beschikbaarheid van informatie. Indien bijvoorbeeld al vroeg duidelijk is dat er een performancerisico is, zal daar een testcluster voor nodig zijn.

### Globale risicoanalyse

Zodra de E2E-testmanager betrokken is (bij het project, de release of een wijziging in de E2E-keten) zal hij op hoofdlijnen de E2E-risico's inschatten. Hiervoor kan hij gebruik maken van de checklist. Zodra, vanuit de E2E-inventarisatie, duidelijk is welke E2E-processen er zijn en wat hun samenhang met het systeemlandschap is, kan de E2E-testmanager de risico's verder detailleren.

Met behulp van deze eerste verkenning kan de E2E-testmanager een eerste planning maken en aangeven wat de noodzaak van de E2E-test is en wat er, globaal, van de organisatie wordt verwacht (in termen van mensen, tijd en materiaal).

### Bepalen aanpak E2E-PRA

Eerst wordt vastgesteld welke risicoanalyses er al gepland of uitgevoerd zijn (in het project of de release). Hierbij moet worden aangesloten en zo nodig moeten daar aparte E2E-PRA's aan worden toegevoegd. Op basis van de globale analyse kan worden besloten om op een bepaald E2E-proces, een aspect of systeem een gedetailleerde risicoanalyse uit te voeren.

Afhankelijk van de grootte van het project en de kwaliteit van de al uitgevoerde PRA's zijn er grofweg 3 varianten voor wat betreft een opzet van een E2E-PRA:

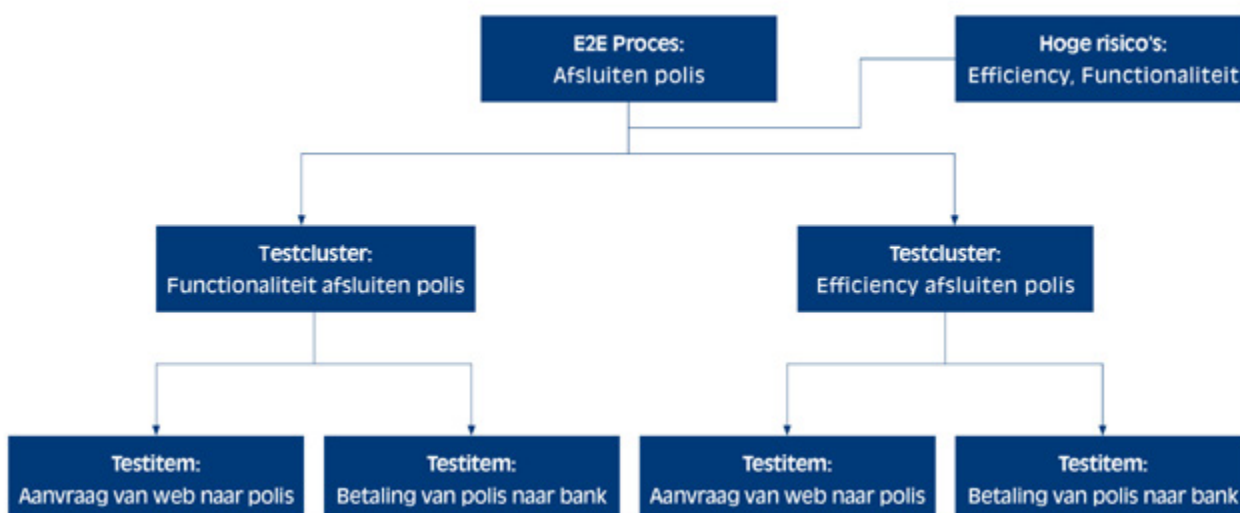
- Er wordt aangesloten bij een al bestaande overall PRA. Het E2E-testteam participeert hierin en zorgt er voor dat mogelijke E2E-risico's op de agenda staan. De E2E-testmanager gebruikt deze vervolgens voor de E2E-teststrategie.
- Er komt een E2E-PRA. Uitkomsten van eventuele andere PRA's worden hierin meegenomen. Er is één sessie of bijeenkomst, optioneel aangevuld met vervolginterviews met specialisten per E2E-proces of systeem.
- De E2E-keten wordt opgedeeld op basis van E2E-processen of delen van de E2E-keten. Per onderdeel volgen een PRA-bijeenkomst of interviews met specialisten. De E2E-testmanager bundelt alle uitkomsten in de E2E-teststrategie.

### Testclusters vaststellen

Een E2E-test is gericht op het testen van de samenhang tussen E2E-processen en systemen. Elk E2E-proces kent tenminste één testcluster: een testcluster is een type test per E2E-proces. Als er hoge risico's of andere goede redenen zijn om in organisatie, uitvoering, testomgeving of testdata verschillende testtrajecten op te zetten, dan worden dit aparte testclusters. Een performancetest is bijvoorbeeld een geheel ander type test dan een functionele test, met mogelijk geheel andere eisen qua bemensing, tools, testdata en omgevingen. Per testcluster zijn er meerdere testitems. Testitems zijn de te testen onderdelen. De testitems van een E2E-test zijn de identificeerbare delen van E2E-processen: handelingen door actoren, beslissende factoren, kritische factoren, deelprocessen en resultaten.

#### Voorbeeld

Het E2E-proces "afsluiten polis" loopt van de klant naar het betalingsverzoek richting de bank en een vastgelegde polis. De testclusters zijn: functionaliteit en efficiency. Binnen de testclusters zijn de testitems het versturen van een verzoek vanuit de webapplicatie naar de polisadministratie maar ook het versturen van een betaling vanuit de polisadministratie via het financiële systeem naar de bank.



*Figuur: Testclusters en testitems*

In de risicoanalyse moeten risico's per testcluster en testitem worden gelokaliseerd. In het geval van een risico op efficiency kan het voorkomen dat het risico is te lokaliseren in één van de testitems en dat de andere testitems minder testdekking behoeven. Dit is dan waardevolle informatie waarmee onnodige testinspanning kan worden voorkomen.

### Risicoanalyse

Kleine technische of functionele onvolkomenheden kunnen in een E2E-keten een groot effect hebben en functioneel perfect werkende systemen kunnen in onderling samenwerken tot niet werkende E2E-processen leiden. Dergelijke E2E-gerelateerde gevolgen openbaren zich soms pas na een lange keten van oorzaken en gevolgen. In de risicoanalyse moet daarom ook worden gedacht in ketens van oorzaak en gevolg. Het vergt verbeelding en domeinkennis om dit in een risicoanalyse te identificeren.

De E2E-inventarisatie kan als ondersteunend middel worden ingezet om ingewikkelde en onvoorspelbare risico's in kaart te brengen.

- In de eerste plaats als plattegrond en routeplanner van alle mogelijke processen, waarin gebeurtenissen en activiteiten kunnen worden doorgedacht en nagelopen. In dit geval worden dan mogelijke gebeurtenissen in hun effect doorlopen (van oorzaak naar gevolg).
- Daarnaast kan er van mogelijke gevolgen (worst case scenario's) worden teruggelopen in de keten om te kijken van waaruit deze worden veroorzaakt (van gevolg naar mogelijke oorzaak). "Is het mogelijk dat de klant een aanbieding te laat krijgt?" is dan het ongewenste gevolg. Terugkijkend in de E2E-keten kan men kijken of er momenten of omstandigheden zijn van waaruit dit wordt veroorzaakt.
- De E2E-inventarisatie levert daarnaast nog een verkenning op van de zogenaamde kritische factoren: dit zijn omstandigheden die van invloed kunnen zijn op de E2E-keten. Ook dit zijn interessante sporen die kunnen leiden naar E2E-risico's. Daarnaast kunnen er uit de risicoanalyse nieuwe inzichten komen betreffende de kritische factoren. Deze kunnen dan meegenomen worden in de E2E-inventarisatie en vervolgens in het testontwerp.
- Doorspreken van scenario's vanuit het perspectief van het uiteindelijke daadwerkelijke gebruik, door de daadwerkelijke gebruiker in de daadwerkelijke situatie.

De vorm waarin een risicoanalyse plaatsvindt, kan variëren. Keuzes hierbij zijn afhankelijk van de benodigde kennis en de hoeveelheid betrokkenen. In de literatuur wordt een aantal mogelijkheden benoemd:

- PRA-bijeenkomst. Deze wordt meestal georganiseerd door de E2E-testmanager. In de bijeenkomst worden de risico's geïdentificeerd door eerst een risicoanalyse te doen van relevante kwaliteitsattributen. Vervolgens wordt een risicoanalyse uitgevoerd op de systeemonderdelen (hier zijn dat dan de E2E-processen, testclusters of testitems).
- Interviews. Dit zijn gesprekken met specialisten of deskundigen. De gehele risicoanalyse is dan een samenvoeging van alle informatie uit interviews en bijeenkomsten. In geval van kleiner onderhoud kan hier vaak mee worden volstaan.
- Gebruik standaardscorelijsten en checklists. Op basis van eigenschappen van systemen en E2E-processen en de projectomstandigheden kunnen risico's worden ingeschat. Checklists zijn altijd handig, zeker in de aanvangsfase, om als tester een voorzet te geven voor de risicoanalyse, maar mogen nooit als enige worden ingezet.

In de praktijk, zeker in geval van een grotere E2E-keten, zal er sprake zijn van combinaties van bovenstaande werkwijzen.

De resultaten zijn een risicoanalyse op hoofdlijnen en een risicoanalyse per testcluster. In de risicoanalyse op hoofdlijnen staat benoemd, per E2E-proces, wat de typen risico zijn en wat de hoogte van de risico's is (dit wordt ook wel aangeduid met het begrip "risicoklasse").

**Tabel: Risicoanalyse op hoofdlijnen**

E2E-proces	Kwaliteitsattribuut	Risico	Kans	Impact	Risico
<b>Aanvragen offerte</b>	Functionaliteit	Niet beschikbare producten worden getoond. Niet uitvoerbare offerte	M	H	B
	Efficiency	Gegevens worden laat getoond, klant gaat weg	H	H	A
	Herstelbaarheid	Time-out tussen polis en product zorgt voor verlies webconnectie	L	H	C
<b>Sluiten polis</b>	Functionaliteit	Niet-beschikbare producten worden getoond. Niet-uitvoerbare aanvraag vanuit offerte. Klant krijgt niet genoeg info bij meldingen en gaat weg	H	H	A
	Herstelbaarheid	Time-out tussen polis en product zorgt voor verlies webconnectie	H	M	B
	Beveiligbaarheid	Betaalgegevens blijven beschikbaar in web-applicatie	L	M	D
	Efficiency	Gegevens en meldingen worden laat getoond, klant gaat weg	H	M	B

Op basis van bovenstaande risico's ligt het voor de hand om de volgende testclusters op te zetten:

- Aanvraag offerte – functionele test.
- Aanvraag offerte – performancetest.
- Aanmaken polis – functionele test.
- Aanmaken polis – performancetest.

Herstelbaarheid en beveiligbaarheid zijn qua risico niet hoog genoeg om aparte testclusters voor op te zetten en kunnen impliciet in een functionele en performancetest worden meegetest.

Vervolgens moeten, afhankelijk van de grootte van het project, de complexiteit en de risico's, in aparte sessies de risico's per testcluster worden gedetailleerd. We geven als voorbeeld hier het testcluster "Aanmaken polis – efficiency". In geval van efficiency en herstelbaarheid moet worden aangegeven wat het aspect is dat wordt geraakt en voor zover mogelijk, in welk item het risico is de lokaliseren. Dit is een onderverdeling van het betreffende risico en is van belang voor de uit te voeren testen.

**Tabel: Risicoanalyse per testcluster: Aanmaken polis - efficiency**

Testitem	Aspect	Risico	Kans	Impact	Risico
<b>Overall Efficiency</b>		Gegevens en meldingen worden laat getoond, klant gaat weg	H	M	B
<b>Overall herstelbaarheid</b>		Time-out tussen polis en product zorgt voor verlies webconnectie	H	M	B
<b>Aanvraag vanuit web naar polis</b>		Laag risico, veldvalidaties betrouwbaar, eenvoudige interface	L	M	D

Testitem	Aspect	Risico	Kans	Impact	Risico
<b>Polis haalt klantgegevens op</b>		In geval van geen issues, kans laag: interface en verwerking simpel	L	M	D
<b>Klant haalt BKR-gegevens op</b>	Respons	Bij BKR-check negatief: extra velden in interface	M	M	C
<b>Klant maakt nieuwe relatie aan</b>	Timing	Melding terug naar polis kan qua timing (batchprocessen) max. 1 dag wachten	H	M	B
<b>Klantgegevens kloppen niet &gt; melding</b>	Timing	Melding terug naar polis kan qua timing (batchprocessen) max. 1 dag wachten	H	M	B
<b>Melding vanuit polis naar web</b>	Stress, load	In geval van foutmeldingen kan dit max. 1 dag duren. Interface heeft wachtrij bij aantallen +1000	H	M	B
<b>Polis &gt; financieel</b>		Standaardbetalingen	L	M	D
<b>Financieel &gt; polis</b>		Bij retourbetaling standaardmelding, betrouwbaar	L	M	D
<b>Financieel &gt; bank</b>		Standaard	L	M	D
<b>Polis &gt; post</b>		Standaard interface, betrouwbaar en getest. Geen complexe combinaties	L	L	E
<b>Post &gt; Bezorgdienst</b>		Standaard interface, betrouwbaar en getest	L	L	E

Hier houdt de risicoanalyse op. In voorkomende gevallen willen specialisten (bijvoorbeeld als het gaat om efficiency- of beveiligingsrisico's) op nog dieper niveau weten waar de risico's moeten worden gelokaliseerd.

### Teststrategie op hoofdlijnen definiëren

Op de analyse van risico's volgt het vaststellen van de risicobeperkende maatregelen. Risico's worden nooit tot nul gereduceerd, maar het gaat er om de risico's tot een aanvaardbaar niveau te reduceren. De hoogte van een risico wordt bepaald door de kans op een ongewenst effect aan de ene kant, en de impact van het effect aan de andere kant. Beide kanten zijn kandidaten voor (gunstige) beïnvloeding.

Naast testen zijn er nog meer risicobeperkende maatregelen denkbaar, zoals het uitvoeren van reviews, simulaties, proofs of concept.

Niet alle risico's kunnen door testen voldoende worden afgedekt. Redenen hiervoor kunnen zijn:

- Testen is niet in staat om het risico te beperken, omdat het risico niet de software betreft maar de kwaliteit of kwantiteit van personeel en materieel.
- Het kan zijn dat testen niet de mogelijkheid heeft om het risico te dekken, bijvoorbeeld omdat bepaalde testgevallen niet kunnen worden uitgevoerd door beperkingen in de testomgeving.
- Andere maatregelen zijn effectiever of efficiënter (bijvoorbeeld gecontroleerde beta-testen, pilots in productie, simulaties, monitoring in productie).
- Het risico is zó groot dat naast E2E-testen ook andere maatregelen nodig zijn: het begrip van de processen of het gedrag van klanten is bijvoorbeeld te klein, waardoor men ook voorafgaand aan de definitieve uitrol kleine pilots, testen in usability labs of beta-testen moet uitvoeren.

Deze overwegingen zijn van belang omdat na een PRA niet te hooggespannen verwachtingen mogen worden gewekt voor wat betreft de potentie van testen om de hoogste risico's naar aanvaardbare niveaus te brengen. De



E2E-testmanager moet aangeven wat de rol, scope en diepgang van de testen zijn en in welke mate de testen de risico's reduceren. Risico's waarbij testen slechts een bescheiden rol kan spelen moeten worden overgedragen aan hoger management.

In checklists worden per elk van de ISO 9126-kwaliteitsattributen, typische E2E-risico's beschreven en worden risicobeperkende maatregelen gegeven.

### 6.3 Teststrategie

De teststrategie is de wijze waarop de risico's door testen worden beperkt. Van de productrisico's moet worden vastgesteld of ze kunnen worden gedekt door de E2E-test en vervolgens moet de testaanpak voor elk van de risico's en alle testclusters worden bepaald. Aansluitend moeten de uit te voeren activiteiten in samenhang worden uitgewerkt. Bij deze overwegingen moeten de beperkingen in tijd, materieel en bemensing in acht worden genomen.

Het opstellen van de teststrategie kent de volgende activiteiten:

- Vaststellen testtechnieken en dekkingvormen.
- Vaststellen testdiepgang per testcluster.
- Uitwerken benodigde activiteiten en testrondes.

#### Testtechnieken en dekkingvormen

Voor de diepgang wordt per testitem (actoren, beslissende en kritische factoren, resultaten) bepaald hoe en hoe vaak deze door de E2E-test geraakt moet worden. Is een E2E-proces al afdoende getest door andere teams? Is het resultaat van een testitem (bijvoorbeeld het type product) niet spannend voor het verloop door de keten heen, dan kan worden gekozen voor een lagere dekkinggraad en bijvoorbeeld maar één of twee representatieve producten.

Daarnaast moet worden gekeken naar combinaties van testitems: is de betalingsvorm bijvoorbeeld interessant in combinatie met bepaalde soorten producten? In dat geval kan men kijken naar datacombinaties van deze elementen. Ook kan men (als het gaat om overschrijdingen van maxima bijvoorbeeld) binnen een testitem kijken of men een dekkingvorm als grenswaardenanalyse moet toepassen.

Op basis hiervan kunnen logische testgevallen worden uitgewerkt totdat de benodigde dekking per testitem, van combinaties van testitems en van overstijgende zaken als historie en de levenscyclus van gegevens, is verkregen. Er moet dan waarschijnlijk nog een ontdebelling plaatsvinden waarbij men kijkt of verschillende testgevallen niet middels één en hetzelfde testgeval kunnen worden afgedekt.

Voor wat betreft testtypes als het testen van efficiency, failover en beveiliging zal men bij het bepalen van variatie in testgevallen moeten kijken naar voor deze testen relevante zaken. Bijvoorbeeld: welke deelprocessen en processtappen in de keten kunnen invloed hebben op efficiency, failover en beveiliging? Welke technische storingen zijn er te verwachten in het uitvoeren van een E2E-proces, in welk deel van de E2E-keten zitten deze en hoe kunnen we die tijdens een test opwekken? Er moet worden gewaakt voor het herhalen van systeemtesten en men moet er van uit gaan dat met name de functionaliteit van afzonderlijke systemen al eerder getest is.

Hieronder wordt een aantal testtechnieken beschreven die kandidaat zijn om te worden gebruikt in een E2E-test. Voor een detailbeschrijving en handleiding bij de techniek wordt hier verwezen naar de testliteratuur.

Voor het uitwerken van de testgevallen gelden de volgende hoofdregels:

- Ten eerste moeten de belangrijkste, meest voorkomende en meest risicovolle realistische scenario's als testgevallen worden uitgewerkt.
- De risicovolle delen (bijvoorbeeld een bepaalde interface of functie) moeten vervolgens dieper worden geraakt middels extra testgevallen. Hier kunnen testtechnieken en dekkingvormen behulpzaam zijn.
- Daarnaast wordt vastgesteld of er testitems of deelprocessen in de E2E-keten zijn die door deze testgevallen nog niet worden geraakt. Indien nodig moeten er extra testgevallen worden ontworpen totdat ook deze elementen allemaal tenminste één maal zijn geraakt door testgevallen. Het uitwerken van logische testgevallen moet resulteren in een overzicht van de dekking van testitems door de testgevallen.



**Tabel: Kandidaat-testtechnieken**

Testtechniek	Omschrijving/toepassing
<b>Classification tree (datacombinatie-test)</b>	De ideale E2E-testtechniek: grafisch en onderhoudbaar. Hiermee kunnen de verschillende elementen (actoren en factoren) gemakkelijk worden uitgetekend en testtechnieken en dekkingsvormen worden gecombineerd binnen één E2E-proces. Bovendien is in één oogopslag te zien welk testgeval welke elementen/klassen raakt. In het hier gegeven voorbeeld wordt gebruik gemaakt van deze techniek.
<b>Data flow</b>	Goede techniek om de flow door het systeem mee op te zetten en de paden door het systeem aan te geven.
<b>Procescyclus</b>	Idem als data flow
<b>Beslistabellen</b>	Goede techniek om combinaties van elementen (die dan elk als condities of beslissingen worden beschouwd) mee uit te werken. Meestal te zwaar voor E2E.
<b>State transition</b>	Techniek om gebeurtenissen in de infrastructuur (zoals achtergrondprogramma's of jobs) en hun onderlinge samenhang mee te testen. Is dan complementair aan een techniek als de Classification tree.
<b>Gegevenscyclustest</b>	Goede techniek om mogelijke mutaties en gebeurtenissen op gegevens systematisch te testen. Een groot deel van deze testgevallen moet al door systeemtesten gedekt zijn!
<b>Thin threads</b>	Goede overall techniek zoals de DCT. Gaat uit van een primair scenario (happyflow). Vanuit de happy flow kunnen varianten op basis van voor de E2E-keten relevante variabelen en factoren worden uitgewerkt.

Voor de volledigheid volgt een lijst met mogelijke dekkingsvormen. Een dekkingsvorm geeft de mogelijkheid te differentiëren in testdiepgang binnen een testtechniek. Voor een detailbeschrijving en handleiding bij de dekkingsvorm wordt verwezen naar de testliteratuur.

**Tabel: Kandidaat-dekkingsvormen**

Dekkingsvorm	Omschrijving
<b>Paden</b>	Doel: alle E2E-processen in de keten ten minste één keer raken.
<b>Goedpaden/foutpaden</b>	Naast reguliere paden van begin tot eind, ook mogelijke fouten initiëren.
<b>Equivalentieklassen</b>	Bepalen welke belangrijke groepen testdata er zijn en voor elke groep ten minste één representatief testgeval opstellen.
<b>Grenswaardenanalyse</b>	Per grenswaarde die beslissend is in de keten, zorgen dat er een testgeval is voor zowel de waarde net voor de grens, op de grens als na de grens.
<b>Pairwise testing</b>	Combinaties van beslispunten (of testitems in de E2E-test) uitwerken in testgevallen.
<b>Beslissing</b>	Voor alle beslispunten in de keten bepalen welke keuzes de systemen maken en voor elke uitkomst zorgen dat er een testgeval is dat dit dekt.
<b>Conditie</b>	Voor wat betreft alle beslispunten in de keten bepalen welke condities tot de beslispunten leiden. Er moeten testgevallen zijn die elke conditie beslissend laten zijn voor wat betreft de uitkomst. Dit leidt in de regel tot meer testgevallen dan de dekkingsvorm Beslissing.
<b>Statistisch gebruik</b>	Op basis van informatie met betrekking tot gebruik van processen en ketens, bepalen welke onderdelen of processen hoe of met welke zwaarte moeten worden getest.

Er wordt aangeraden om een grafische testtechniek als de datacombinatietest als uitgangspunt te nemen. Testgevallen zijn dan overzichtelijk, onderhoudbaar en goed uit te leggen aan anderen. De klassen binnen de DCT zijn dan de actoren, beslissende factoren, kritische factoren en resultaten zoals die in de E2E-inventarisatie zijn vastgesteld. In de meeste gevallen zal een E2E-testontwerp hiermee klaar zijn. In geval van hoge risico's in delen van de E2E-keten kunnen dan andere testtechnieken en dekkingsvormen worden ingezet.

### Testaanpak per testcluster

In de risicoanalyse zijn de testclusters vastgesteld. Dit zijn de E2E-processen, met daarbinnen een onderverdeling op basis van type risico's. Om dit te verduidelijken wordt verder ingegaan op het voorbeeld dat is gegeven in het hoofdstuk over risicoanalyse.

**Tabel: Risicoanalyse op hoofdlijnen**

E2E-proces	Kwaliteitsattribuut	Risico
Aanvragen offerte	Functionaliteit	B
Aanvragen offerte	Efficiency	A
Aanvragen offerte	Herstelbaarheid	C
Aanmaken polis	Functionaliteit	A
Aanmaken polis	Herstelbaarheid	B
Aanmaken polis	Beveiligbaarheid	D
Aanmaken polis	Efficiency	B

Op basis van bovenstaande risico's zijn de volgende testclusters vastgesteld:

- Aanvraag offerte – functionele test.
- Aanvraag offerte – performancetest.
- Aanmaken polis – functionele test.
- Aanmaken polis – performancetest.

Per testcluster worden de risico's nu gedetailleerd (voor wat betreft de hoge risico's). Per testcluster leidt dat tot het volgende overzicht:

**Tabel: Risicoanalyse per testcluster: Aanmaken polis - efficiency**

Testitem	Aspect	Risico	Risico
Overall efficiency		Gegevens en meldingen worden laat getoond, klant gaat weg	A
Overall herstelbaarheid		Time out tussen polis en product zorgt voor verlies webconnectie	B

#### Per onderdeel:

Aanvraag vanuit web naar polis		Laag risico, veldvalidaties betrouwbaar, eenvoudige interface	D
Polis haalt klantgegevens op		In geval van geen issues kans laag: interface en verwerking simpel	D
Klant haalt BKR-gegevens op	Respons	Bij BKR-check negatief: extra velden in interface	C
Klant maakt nieuwe relatie aan	Timing	Melding terug naar polis mag (batchprocessen) max. 1 dag wachten, dit kan worden overschreden	A

Testitem	Aspect	Risico	Risico
<b>Klantgegevens kloppen niet &gt; melding</b>	Timing	Melding terug naar polis mag (batchprocessen) max. 1 dag wachten, dit kan worden overschreden	A
<b>Melding vanuit polis naar web</b>	Stress, load	In geval van foutmeldingen mag dit max. 1 uur duren, dit kan worden overschreden Interface heeft wachtrij bij aantallen +1000	A
<b>Polis &gt; financieel</b>		Standaard betalingen	D
<b>Financieel &gt; polis</b>		Bij retourbetaling standaard melding, betrouwbaar	D
<b>Financieel &gt; bank</b>		Standaard	D
<b>Polis &gt; post</b>		Standaard interface, betrouwbaar en getest. Geen complexe combinaties	E
<b>Post &gt; bezorgdienst</b>		Standaard interface, betrouwbaar en getest	E

Nu moet een aanpak per testcluster worden opgesteld die de risico's naar rato dekt. Dit leidt tot een volgende testaanpak per testcluster:

**Tabel: Testaanpak per testcluster**

Testcluster	Kwaliteits-attribuuat/aspect	Risico	Aanpak
<b>Aanvraag offerte – functionele test</b>	Functionaliteit	B	Alle beslissende en kritische factoren ten minste één maal raken (DCT) met grenswaardenanalyse van opvragen productinfo (ingangdatum, geldigheidsperiode)
<b>Aanvraag offerte – performancetest</b>	Efficiency	A	Focus op interactie tussen polis en relatie en web. Aparte productiegelijke omgeving met productiegelijk draaien van achtergrondprogramma's en testdata. Specialisten inzetten.
	Herstelbaarheid	C	Triggeren enkele time-outs tussen polis en product.
<b>Aanmaken polis – functionele test</b>	Functionaliteit	A	Alle condities in het E2E-proces moeten ten minste één maal beslissend zijn met extra aandacht voor beschikbaarheid en aanvraagbaarheid producten. Gebruikmaken van daadwerkelijke accounts en autorisatiematrix.
	Beveiligbaarheid	D	Levensloop betaalgegevens als kritische factor
<b>Aanmaken polis – performancetest</b>	Efficiency	B	In performancetestomgeving responstijden testen van meldingen in de keten (testscenario's functionele test)
	Herstelbaarheid	B	Triggeren alle mogelijke time-outs tussen polis en product. Controle op verloop na einde time-out

#### Afhankelijkheden en beperkingen voor de testaanpak

Er is nu vastgesteld hoe er getest zou moeten worden per testcluster. Er moet nog rekening gehouden worden met de volgende zaken:

1. de beschikbaarheid van de juiste mensen met de juiste vaardigheden in de juiste perioden;
2. de beschikbaarheid van de juiste testomgevingen;
3. de beschikbaarheid van de juiste data;
4. afhankelijkheden met andere testsoorten.

**Uitleg:****1. Bemensing**

Welke expertise is nodig om alle geïdentificeerde testen uit te kunnen voeren? Hebben we bijvoorbeeld test-specialisten op het gebied van performancetesten tot onze beschikking en wat is er voor nodig om deze te krijgen? Hebben we E2E-testers die de capaciteiten hebben om testgevallen uit te schrijven en uit te voeren? Hoe groot is de beschikbare en benodigde capaciteit?

**2. Testomgevingen**

Wat is de beschikbaarheid van benodigde testomgevingen? Welke testomgevingen zijn nodig, beschikbaar en wat zijn de gevolgen van beperkingen in de beschikbaarheid en grootte van testomgevingen?

**3. Data**

Weten we al of we productiedata kunnen gebruiken, zo niet welke data dan wel en wat dit betekent voor de uitvoerbaarheid van onze testgevallen? Voor E2E-testen zijn vaak testgegevens nodig met historie die ook nog synchroon door de hele keten zijn. Dit is zonder productiedata moeilijker te realiseren.

**4. Afhankelijkheden**

Welke aspecten en risico's worden al door andere testsoorten gedekt en in hoeverre betekent dit dat de E2E-testen kunnen worden beperkt of juist moeten worden opgeschaald? Welke testen moeten klaar zijn voordat delen van de E2E-testen kunnen starten?

**Structurering van de testuitvoering in testrondes**

Een testronde is een verzameling activiteiten die in een bepaalde periode en op dezelfde testomgeving en verzameling testdata wordt uitgevoerd. De testronde duurt tot de doelen voor die ronde zijn gehaald.

Er kunnen meerdere testrondes zijn, met elk andere doelen en uitgangssituaties. Zo kan een testronde beginnen met als startdatum 1 januari, waarbij in de testronde het gehele jaar wordt doorlopen. In een andere testronde wil men juist een jaarovergang zien en begint men op 30 december.

Er kunnen verschillende redenen zijn om de testuitvoering op te splitsen in testronden:

- De beschikbare systeemtijd raakt op. De systeemdatum moet bijvoorbeeld weer terug naar een punt in het verleden.
- Hergebruik van testdata. De testdata raakt op en moet opnieuw geladen worden.
- Testgevallen moeten in verschillende systeemdatums worden uitgevoerd: de één in het verleden, de ander in de toekomst.
- Opnieuw inplannen van geblokkeerde testen. Er worden in de 1e ronde testblokkerende fouten gevonden, waardoor niet alle testgevallen uitvoerbaar zijn en deze opnieuw moeten worden ingepland.
- Testen van tot nu niet beschikbare systemen of systeemdelen. Niet alle systemen in de E2E- keten zijn nog beschikbaar waardoor slechts een beperkt aantal testgevallen of delen van testgevallen kunnen worden uitgevoerd.
- De 1e testronde heeft nog aanloopproblemen, maar men wil wel in korte termijn een bepaalde cyclus doorlopen. Hierdoor moeten testgevallen opnieuw worden uitgevoerd.
- Ronde 1 moet een "hit and run" zijn: met minimale inspanning in een zo kort mogelijke tijdsperiode zo veel mogelijk functionaliteit raken. Het hoofddoel is dan om blokkerende en zware fouten vroeg te vinden. In andere testronden richt men zich op het uitvoeren van alle testgevallen, hertesten van bevindingen of regressietesten.
- Er zijn belangrijke wijzigingen in systemen die ook moeten worden getest (bugs, changes). Testgevallen moeten dan opnieuw worden uitgevoerd, systemen worden opnieuw opgeleverd en testdata moet weer op een beginstand staan.
- Door een wijziging in een database moet data opnieuw worden ingeladen.
- Conversie- of productiedata moet apart of kan pas later worden getest.

Om deze redenen zijn er in de regel minimaal 4 testronden:

- Initiële ronde: hit and run. Doel is om de basis van de keten zo veel mogelijk te raken.
- Doortestronde: blokkerende fouten moeten zijn opgelost. Het gaat in deze ronde om het dekken van het merendeel van de testgevallen in een volledige keten. Doel is om na deze ronde alle zware en belangrijke fouten te hebben gevonden.
- Hertestronde/ rest: in deze testronde wordt de rest van de testgevallen uitgevoerd die in ronde 2 niet uitgevoerd kunnen worden, dienen de meeste en belangrijkste fouten te zijn opgelost en te worden hergetest.
- Regressietestronde: resterende bugfixes worden gehertest en een selectie van alle testgevallen wordt nogmaals uitgevoerd om regressie door wijzigingen en bugfixes te testen.

Per testcluster moet worden beoordeeld welke testronden nodig zijn en of testclusters in elkaars testomgevingen en testronden mee kunnen lopen. Van de hierboven aangegeven testronden kan natuurlijk worden afgeweken.

Hieronder een voorbeeld van een schema met testronden:

	Wk 1	Wk 2	Wk 3	Wk 4	Wk 5	Wk 6	Wk 7	Wk 8	Wk 9	Wk 10
<b>Testomgeving A</b>										
Testcluster A: offerte functioneel	Testronde 1		Testronde 2		Testronde 3		Testronde 4		Testronde 5	
Testcluster B: polis functioneel	Testronde 1		Testronde 2		Testronde 3		Testronde 4		Testronde 5	
Testcluster E: polis beveiliging					Testronde 1				Testronde 2	
<b>Testomgeving B</b>										
Testcluster C: offerte efficiency			Testronde 1				Testronde 2			
Testcluster D: polis efficiency			Testronde 1				Testronde 2			

In een complexe E2E-test gaat het om meerdere testclusters met elk vaak eigen planningen, doorlooptijden en testronden.



# Deel 7

E2E testen  
en risico's

## Deel 7 E2E testen en risico's

In dit hoofdstuk worden de typische E2E-risico's behandeld. Er wordt aangegeven hoe kan worden bepaald hoe hoog het risico is en wat de maatregelen zijn om het risico te reduceren. Er wordt onderscheid gemaakt tussen product-risico's (risico's met betrekking tot in productienaam) en projectrisico's (bedreigingen voor het E2E-testtraject).

### 7.1 Productrisico's

In dit hoofdstuk worden E2E-productrisico's nader uitgelegd. De gegeven productrisico's zijn gebaseerd op de lijst kwaliteitsattributen en aspecten ontleend aan ISO9126. Een aspect is een onderdeel van een kwaliteitsattribuut. Zo bestaat het kwaliteitsattribuut efficiency uit de aspecten load, stress en response.

Per kwaliteitsattribuut wordt een korte uitleg gegeven en vervolgens wordt per kwaliteitsattribuut en de aspecten binnen dat kwaliteitsattribuut een checklist gegeven waarin risicofactoren worden genoemd en testmaatregelen om het risico te beperken.

E2E-productrisico's worden opgedeeld in de volgende kwaliteitsattributen: functionaliteit, betrouwbaarheid, bruikbaarheid, efficiëntie, onderhoudbaarheid en portabiliteit.

Kwaliteitsattribuut	Risico/ aspect
<b>Functionaliteit</b>	Geschiktheid leveren van diensten.
	Actualiteit van gegevens.
	Juistheid van gegevens.
	Levensloop van gegevens.
	Toegankelijkheid.
	Koppelbaarheid (of interoperabiliteit).
	Beveiliging. Dit aspect omvat o.a. autorisatie, privacy en penetratie.
	Overbeveiliging.
<b>Betrouwbaarheid</b>	Foutbestendigheid.
	Storingsbestendigheid (wordt ook wel "failover" genoemd).
	Herstelbaarheid.
	Beschikbaarheid.
	Datavervuiling
	Timing.
<b>Bruikbaarheid</b>	Duidelijkheid en aantrekkelijkheid.
	Traceerbaarheid.
	Bedieningsgemak: installatie.
	Bedieningsgemak: achtergrondprogramma's.
<b>Efficiëntie</b>	Load
	Stress
	Response
<b>Onderhoudbaarheid</b>	Analyseerbaarheid.
	Wijzigbaarheid.
	Stabiliteit.
	Testbaarheid
<b>Portabiliteit</b>	Aanpasbaarheid.
	Installeerbaarheid.
	Samenwerkbaarheid.



### 7.1.1 Functionaliteit

Functionaliteit betreft de doelen van systemen, diensten of onderdelen daarvan.

Functionaliteit is niet in één systeem te testen als de functionele werking is verspreid over meerdere systemen.

Denk bijvoorbeeld aan een straaljager, waar de functie "vernietig een MIG in volle vlucht" niet is te valideren door één van de deelsystemen of combinaties van enkele systemen te testen.

Maar dit kan ook gelden voor administratieve processen. Het afhandelen van een betalingsachterstand bestaat bijvoorbeeld vaak uit samenwerking tussen meerdere systemen, waarbij signaleringen, verzenden van brieven, het aansturen van externe organisaties zoals incassobureaus en verwerkingen van betalingen op elkaar moeten aansluiten over een E2E-keten heen. Het moet dan bijvoorbeeld niet mogelijk zijn dat een klant wel heeft betaald maar toch een deurwaarder op de stoep krijgt!

#### Checklist E2E-risico's functionaliteit

<b>Risico/aspect</b>	Geschiktheid leveren van diensten
<b>Vragen/factoren die het risico bepalen</b>	Het gaat hier om de vraag of de uiteindelijke levering overeenkomt met datgene wat verwacht wordt of nodig is. In hoeverre is in de requirements rekening gehouden met de gebruiker, de daadwerkelijke situatie waarin de systemen worden gebruikt en de daadwerkelijke toestand van iemand tijdens het gebruik van de systemen? Elke functionaliteit kan dan conform specificaties, correct werken, maar er kunnen toch grote problemen optreden. Het tonen van gegevens in een bepaald formaat of in een bepaalde eenheid kan functioneel correct zijn, maar is dit wat de ontvanger of de klant of dat moment verwacht, begrijpt of nodig heeft?
<b>Testmaatregel/aandachtspunten</b>	De verwachtingen concreet maken: interviews met daadwerkelijke gebruikers of klanten. Alle situaties en factoren uittekenen die de levering kunnen beïnvloeden. Dit simuleren in de testomgeving en de resultaten laten beoordelen door de gebruiker of klant.
<b>Risico/aspect</b>	Actualiteit van gegevens
<b>Vragen/factoren die het risico bepalen</b>	Denk hier bijvoorbeeld aan informatie gepresenteerd en verwerkt door één systeem, maar verkregen uit meerdere systemen. Hoe wordt omgegaan met wijzigingen van gegevens in één systeem en in hoeverre moet dit onmiddellijk worden getoond in een ander systeem? Is er een volgordelijkheid in batches waardoor gegevens op de verkeerde dag kunnen worden verwerkt? In ketens waarin grootboekgegevens moeten worden bijgehouden kan dit laatste cruciaal zijn. In geval van jaar- of maandovergangen kan dan bijvoorbeeld het moment waarop een betaling wordt geboekt van belang zijn.
<b>Testmaatregel/aandachtspunten</b>	In de E2E-inventarisatie het tonen en beschikbaarstelling van gegevens door systemen aan elkaar detailleren: de functionele en technische stappen van deze data-flow uitwerken en de kritische factoren (invloeden op de flow) analyseren. Testgevallen die combinaties van deze stappen en factoren dekken.
<b>Risico/aspect</b>	Juistheid van gegevens
<b>Vragen/factoren die het risico bepalen</b>	Gegevens gaan via interfaces van het ene systeem naar het andere. In hoeverre wordt er van de gegevens gebruik gemaakt zoals ze zijn bedoeld? Is de betekenis van gegevens universeel door de keten heen? Persoonsgegevens kunnen bijvoorbeeld meerdere betekenissen hebben. Betreffen persoonsgegevens bijvoorbeeld de

premiebetaler of de verzekerde? Syntactische correctheid, zoals in een interface-test wordt vastgesteld, garandeert dit niet. Hoe rondt de verschillende systemen bedragen af en hoe kan dit in een reeks van overdrachten worden overgeërfd tot uiteindelijk onaanvaardbare afwijkingen?

**Testmaatregel/  
aandachtspunten**

Vanuit eindproducten van de keten (bijvoorbeeld brieven, facturen) centrale gegevens zoals bedragen en geadresseerden vaststellen en de bron traceren. Daarnaast de keten waar de gegevens doorheen lopen analyseren. Happyflows en varianten die de gegevens kunnen beïnvloeden uitwerken tot testgevallen.

**Risico/aspect**

Levensloop van gegevens

**Vragen/factoren  
die het risico bepalen**

Wat kan een klant, een product of een dienst allemaal overkomen? Hoe wordt dit verwerkt? Wat zijn de gevolgen daarvan in de keten? Stel dat een klant overlijdt. Hoe moeten de systemen omgaan met lopende deelprocessen, zoals rekeningen, betalingsachterstanden, opgetogen brieven met felicitaties over net afgesloten producten en dergelijke?

**Testmaatregel/  
aandachtspunten**

Voor het testen van de levensloop van gegevens moeten testers met ervaren gebruikers en beheerders analyseren welke mogelijke gebeurtenissen van belang kunnen zijn en hoe deze te simuleren.

**Risico/aspect**

Toegankelijkheid

**Vragen/factoren  
die het risico bepalen**

In hoeverre zijn gegevens vanuit een ander systeem beschikbaar? In bijvoorbeeld Shared Service Centers vindt vaak een samenballing plaats van gegevens uit diverse systemen. Waar is dit van afhankelijk en heeft elke type gebruiker de juiste mogelijkheden en autorisaties? Wat gebeurt er na een wijziging van autorisaties of een wijziging in een datamodel? Kan men dan nog dezelfde gegevens inzien of moeten er ook andere gegevens worden getoond?

**Testmaatregel/  
aandachtspunten**

Definiëren welke gegevens van belang zijn in welke situatie en deze als ware het productie aanmaken (of kopiëren uit productie). Gebruikers die dit in productie ook doen, de beschikbaarheid laten beoordelen.

**Risico/aspect**

Koppelbaarheid (of interoperabiliteit)

**Vragen/factoren  
die het risico bepalen**

Dit is het vermogen van een systeem om samen te werken met andere systemen. Hoe eenvoudig, inzichtelijk, onderhoudbaar en testbaar zijn de koppelingen en E2E-processen met andere systemen? In een systeemtest moeten interfaces worden getest en kunnen zelfs bestanden tussen systeemtesten onderling worden getest. Dit zijn belangrijke eerste stappen. Maar is dit voldoende?

**Testmaatregel/  
aandachtspunten**

De kwaliteit van interfaces en gedeelde processen blijkt pas door E2E-processen (ook over langere tijd) door te testen in een complete E2E-testomgeving.

**Risico/aspect**

Beveiliging. Dit aspect omvat o.a. autorisatie, privacy en penetratie

**Vragen/factoren  
die het risico bepalen**

Kunnen systemen door een directe koppeling met de database van een ander systeem een beveiligingsgat veroorzaken? Zijn er achtergrondprogramma's die alle rechten hebben, waardoor er een risico ontstaat?

<b>Testmaatregel/ aandachtspunten</b>	Het testen van beveiliging is vaak alleen in een E2E-test mogelijk omdat beveiligingsgaten op lokaal systeemniveau worden overgeërfd naar het geheel.
<b>Risico/aspect</b>	Overbeveiliging
<b>Vragen/factoren die het risico bepalen</b>	In hoeverre kunnen veiligheidsmaatregelen de handelingsvrijheid van gebruikers en klanten te veel beperken en wat gebeurt er met de werkbaarheid en aantrekkelijkheid? Zijn beveiligingsmaatregelen in het ene systeem een te grote rem op werkbaarheid in andere systemen? Gebruiken achtergrondprogramma's en andere systemen gebruikersaccounts waar grote restricties op zijn toegepast en kan dit hun functioneren beïnvloeden?
<b>Testmaatregel/ aandachtspunten</b>	Het testen van overbeveiliging is vaak alleen in een E2E-test mogelijk omdat beveiligingsmaatregelen op lokaal systeemniveau worden overgeërfd naar het geheel.

### 7.1.2 Betrouwbaarheid

Betrouwbaarheid is het vermogen om gedurende een vastgestelde periode onder vastgestelde omstandigheden het prestatieniveau te handhaven.

Een praktijkvoorbeeld van een betrouwbaarheidsprobleem (aspect foutbestendigheid) is een geval waarbij postcodes zowel met als zonder spaties tussen cijfers en letters kunnen worden ingevoerd. Dit was een geaccepteerde onvolkomenheid in een bronsysteem, waarbij men niet beseftte dat in een applicatie verderop in de keten het vrijwel onmogelijk werd om te zoeken op postcode.

Voorbeeld van een timingprobleem is een achtergrondprogramma dat elke avond draait om wijzigingen in gegevens te signaleren. Door een toename van de gegevens draaien de achtergrondprogramma's langer en eindigen daarom soms na middernacht. Een deel van de wijzigingen wordt niet meer gesignaleerd, omdat wordt gezocht op de huidige dag. Laatst gewijzigde gegevens werden na middernacht gezien als gewijzigde gegevens van de dag ervoor.

### Checklist E2E-risico's betrouwbaarheid

<b>Risico/aspect</b>	Foutbestendigheid
<b>Vragen/factoren die het risico bepalen</b>	De mate waarin E2E-processen kunnen worden uitgevoerd in geval van fouten of foutieve invoer. Het kan hier gaan om al dan niet bekende fouten in programmatuur waarvan men niet goed heeft ingeschat wat de gevolgen verder op in de keten zijn. Welke fouten heeft men in een systeem geaccepteerd? Hoe wordt de functionaliteit met de fouten door andere systemen gebruikt? Welke foutieve invoer of onverwachte acties kan een gebruiker, klant of een systeem uitvoeren? Wat kunnen de gevolgen hiervan zijn op E2E-processen?
<b>Testmaatregel/ aandachtspunten</b>	Dit aspect wordt getest door in een E2E-testronde alle technische processen en achtergrondprogramma's zo productiegelukkig te laten draaien en E2E-processen op te starten die functionaliteit met bekende fouten raakt.

<b>Risico/aspect</b>	Storingsbestendigheid (wordt ook wel "failover" genoemd)
<b>Vragen/factoren die het risico bepalen</b>	Hoe gaat de gehele keten om met het verwerken van technische processen of transacties als tijdens de verwerking ergens in een systeem een storing optreedt? Hoe traceerbaar is dit? Worden transacties dubbel uitgevoerd, verdwijnen ze of worden ze niet correct afgehandeld? Een E2E-test leent zich bij uitstek om hoge risico's m.b.t. failover in kaart te brengen en te testen.
<b>Testmaatregel/aandachtspunten</b>	Dit aspect wordt getest door in een E2E-testronde alle technische processen en achtergrondprogramma's zo productiegelijk te laten draaien en een aantal E2E-processen op te starten. Voorkomende storingen worden dan gesimuleerd en men start de E2E-ketens en E2E-processen weer op.
<b>Risico/aspect</b>	Herstelbaarheid
<b>Vragen/factoren die het risico bepalen</b>	Het gaat hier om de vraag hoe moeilijk het is om processen weer beschikbaar te krijgen na een storing en de impact van deze verstoring op het E2E-proces.
<b>Testmaatregel/aandachtspunten</b>	Zie storingsbestendigheid. Testers en beheerders moeten nauw samenwerken om tijdens E2E-processen storingen uit te voeren, de storingen te verhelpen en de verwerking van de E2E-processen te beoordelen.
<b>Risico/aspect</b>	Beschikbaarheid
<b>Vragen/factoren die het risico bepalen</b>	Wat kan de beschikbaarheid van een dienst of product beïnvloeden? Wat werkt er in de gehele keten niet meer als één applicatie of een deelproces niet meer werkt?
<b>Testmaatregel/aandachtspunten</b>	Vaak wordt dit aspect impliciet getest door in een E2E-testronde alle technische processen en achtergrondprogramma's zo productiegelijk mee te laten draaien. Kritische situaties en omstandigheden kunnen dan extra worden gesimuleerd.
<b>Risico/aspect</b>	Datavervuiling
<b>Vragen/factoren die het risico bepalen</b>	In hoeverre is het mogelijk dat er ondeugdelijke of ongebruikte gegevens in het systeem bestaan? In hoeverre is het mogelijk dat transacties of handelingen niet worden afgerond en dat er daardoor gegevens ongewenst blijven bestaan? Als er ergens in de keten een half afgeronde aanvraag van een klant zit kan deze klant dan wel opnieuw een account aanmaken? In hoeverre kan dit de functionele of non-functionele werking (zoals beschikbaarheid en efficiëntie) beïnvloeden? Kunnen deze gegevens (over alle systemen heen) volledig worden verwijderd of gewijzigd met beheerfunctionaliteit of moeten er diepgaande maatregelen worden verricht in diverse databases?
<b>Testmaatregel/aandachtspunten</b>	Ervaren beheerders weten vaak van voorbeelden van overbodige data: gebruik hen om testen en controles uit te voeren (Exploratory Testing). Met productieloads een productiegelijke inrichting van opzetten en dagelijkse achtergrondprogramma's draaien en de resultaten en het gedrag van de systemen beoordelen.

<b>Risico/aspect</b>	Timing
<b>Vragen/factoren die het risico bepalen</b>	Timing heeft te maken met afhankelijkheden in momenten van het opstarten en beëindigen van technische processen tussen systemen. Welk zijn dit? Zijn deze volledig productiegelijk gesimuleerd in de E2E-testomgeving? Welke omstandigheden of gebeurtenissen kunnen van invloed zijn (bijvoorbeeld een bepaalde load, een veel voorkomende storing of andere processen die de verwerkingstijden kunnen beïnvloeden)?
<b>Testmaatregel/aandachtspunten</b>	Vaak wordt dit aspect impliciet getest door in een E2E-testronde alle technische processen en achtergrondprogramma's zo productiegelijk mee te laten draaien. Kritische situaties en omstandigheden kunnen dan extra worden gesimuleerd.

### 7.1.3 Bruikbaarheid

Bruikbaarheid gaat over het gemak waarin gebruikers, klanten of beheerders handelingen kunnen verrichten. Bruikbaarheid betreft de inspanning benodigd voor het gebruik en de beoordeling daarvan door (mogelijke) gebruikers. De bruikbaarheid moet beoordeeld worden voor alle verschillende gebruikerstypen die bij de toepassing van het softwareproduct betrokken zijn.

Een voorbeeld van een bruikbaarheidsprobleem (aspect duidelijkheid) is wanneer systeemmeldingen in andere systemen als onbegrijpelijke meldingen verschijnen. Deze mogelijke meldingen moeten dan als kritische factoren worden opgenomen in de E2E-inventarisatie. In de testanalyse moet worden uitgezocht hoe deze meldingen zijn op te roepen, in welke situaties, en waar deze dan kunnen verschijnen. Dit kan aan gebruikers worden gedemonstreerd. Deze kunnen het uiteindelijke gevolg vaststellen en eventuele oplossingen kunnen worden geïnitieerd.

Een ander voorbeeld van een risico op het gebied van duidelijkheid: bij het plaatsen van een order via internet moet de klant geregistreerd zijn. De klant registreert zich maar maakt een fout in het registratieproces en hij klikt op de "back" button. Hij is nu niet volledig geregistreerd in de database van het klantsysteem. Als de klant door wil, kan hij niet inloggen (hij krijgt een melding dat hij niet bestaat), maar hij kan zich ook niet registreren omdat hij bij een nieuwe registratie de melding krijgt dat hij al wél bestaat. De klant kan geen order plaatsen. Dit risico heeft ook raakvlakken met datavervuiling (betrouwbaarheid).

### Checklist E2E-risico's bruikbaarheid

<b>Risico/aspect</b>	Duidelijkheid en aantrekkelijkheid
<b>Vragen/factoren die het risico bepalen</b>	Dit kan een E2E-risico betekenen indien het E2E-proces op een complexe manier gebruik maakt van de systeemketen en indien de gebruiker tijdens zijn E2E-proces met de diverse systemen wordt geconfronteerd. In dat geval kunnen zaken als onduidelijke foutmeldingen uit andere systemen, ontoegankelijkheid van informatie en efficiëntieproblemen een effect hebben op de bruikbaarheid. In hoeverre kunnen systeemmeldingen op schermen van gebruikers en klanten komen? Hoe loopt de verwachte flow van een gebruiker of klant en in welke stappen wordt daarbij uit verschillende systemen gegevens getoond?

<b>Testmaatregel/ aandachtspunten</b>	Gebruikers/ klantscenario in kaart brengen en deze relateren aan het systeem-landschap. Inventariseren van systeemmeldingen/ foutmeldingen en waar deze vandaan komen. Simuleren/ Exploratory testen van gebruikersscenario's in een E2E-testomgeving. (Bijvoorbeeld: probeer op alle verschillend mogelijke manieren om een polis af te sluiten).
<b>Risico/aspect</b>	Traceerbaarheid
<b>Vragen/factoren die het risico bepalen</b>	Wat moet traceerbaar zijn, voor wie, wanneer? Is het mogelijk dat een klant belt over een transactie die door een lange systeemketen heen loopt en waarbij gegevens uit eerste stappen in de keten nog niet actueel worden getoond? Welke stappen uit een E2E-proces moeten zichtbaar en duidelijk zijn in bijvoorbeeld gebruikerssystemen of een Web-applicatie?
<b>Testmaatregel/ aandachtspunten</b>	De E2E-inventarisatie is een middel om de traceerbaarheid duidelijk te maken (wat moet van waaruit, in welke situatie, waar zichtbaar zijn).
<b>Risico/aspect</b>	Bedieningsgemak: installatie
<b>Vragen/factoren die het risico bepalen</b>	Een installatie van een nieuw systeem of een nieuwe versie kan zo complex zijn dat systemen (te) lang niet beschikbaar zijn of koppelingen na een installatie niet meer werken.
<b>Testmaatregel/ aandachtspunten</b>	Op de E2E-testomgeving conform beheerprocedures uitvoeren van installaties, herstarten en dergelijke.
<b>Risico/aspect</b>	Bedieningsgemak: achtergrondprogramma's
<b>Vragen/factoren die het risico bepalen</b>	Is de onderlinge afhankelijkheid van achtergrondprogramma's en batches (ook over de systemen heen) goed in kaart gebracht? Wordt dit als software beheerd en getest? Gebruikt men bij deze testen representatieve data? Problemen die hieruit voort kunnen komen zijn: gegevens kunnen verloren gaan, databases kunnen vervuild raken, gegevens en diensten zijn niet meer actueel en volledig, grootboeken kloppen niet meer, gegevens zijn niet meer benaderbaar, et cetera.
<b>Testmaatregel/ aandachtspunten</b>	Achtergrondprogramma's laten draaien en beheren op de E2E-testomgeving conform werkwijze en planning van de productieomgeving.

#### 7.1.4 Efficiëntie

Efficiëntie (of: performance) omvat load, stress en respons. In de ISO lijst worden deze onder "transactiesnelheid" (respons) en "middelenbeslag" gevat. E2E-risico's met betrekking tot efficiëntie betreffen vooral de overerving van efficiency van deelsystemen door de keten heen. Het presteren van een keten is minimaal de optelsom van alle stappen in de keten. De zwakste schakel bepaalt de uiteindelijke efficiency van een E2E-proces maar ook alle schakels samen kunnen voor onverwachte efficiencyproblemen zorgen. Eén langzaam deel van de keten kan de efficiëntie van de keten omlaag brengen, maar een aantal elk op zich goed presterende systemen kan toch samen een lage efficiëntie hebben.

Efficiëntie is daarmee slechts deels op systeemniveau te testen. Men kan per systeem de efficiëntie van dat systeem beoordelen, maar het wordt dan moeilijk vast te stellen hoe de onderlinge afhankelijkheid tussen

systemen van invloed is op de snelheid van de integrale dienst. Efficiëntie kan daarom vaak alleen in een productiegelijke omgeving worden getest. Dit impliceert dan veelal een complete keten.

Het komt voor dat de snelheid van een transactie juist beperkt moet zijn en dat een te hoge snelheid het E2E-proces kan blokkeren. Een bijvoorbeeld is als processen in diverse systemen parallel starten maar wel elkaars resultaat nodig hebben. Als een proces dan sneller klaar is, is de info in het andere systeem nog niet beschikbaar. Een hogere snelheid kan dan juist een E2E-proces blokkeren! Zie hiervoor ook "[Timing](#)" onder Betrouwbaarheid.

### Checklist E2E-risico's Efficiency

<b>Risico/aspect</b>	Load
<b>Vragen/factoren die het risico bepalen</b>	Load betreft het gedrag van het systeem bij een gegeven belasting. Welke verwachte aantallen te verwerken gegevens en transacties zijn van belang? Zijn er pieken te verwachten? Kunnen systemen onderling voor pieken zorgen? Kan er na een storing een stuwmeer ontstaan die in één keer een ander systeem extra kan belasten?
<b>Testmaatregel/aandachtspunten</b>	E2E-testgevallen, E2E-testgegevens en E2E-testomgevingen kunnen hergebruikt worden in een performancetest. De E2E-inventarisatie is een goede basis om bottlenecks te signaleren of bevindingen te analyseren.

<b>Risico/aspect</b>	Stress
<b>Vragen/factoren die het risico bepalen</b>	Stress betreft de maximale belastbaarheid van het systeem. Is het van belang te weten hoe het systeem zich gedraagt bij een extreme piekbelasting of langdurig zware belasting? Kunnen systemen in die gevallen elkaar of het E2E-proces blokkeren? Zijn er loggingbestanden die dan kunnen vollopen en die transacties en E2E-processen kunnen blokkeren? Maken meerdere systemen of technische deelprocessen gebruik van dezelfde hardware, processoren of software? In hoeverre kan bij een zware belasting van meerdere van zulke systemen zulke gedeelde componenten de performance omlaag gaan?
<b>Testmaatregel/aandachtspunten</b>	E2E-testgevallen, E2E-testgegevens en E2E-testomgevingen kunnen hergebruikt worden in een performancetest. De E2E-inventarisatie is een goede basis om bottlenecks te signaleren of bevindingen te analyseren.

<b>Risico/aspect</b>	Response
<b>Vragen/factoren die het risico bepalen</b>	Response betreft de uitvoeringstijden en verwerkingsnelheden bij het vervullen van de functies. Zijn functies, transacties of delen van E2E-processen voor de verwerkingsnelheid afhankelijk van andere systemen? Welke resultaten (op scherm, bestand of output) zijn het meest kritisch? Zijn deze resultaten afhankelijk van een lange E2E-keten?
<b>Testmaatregel/aandachtspunten</b>	E2E-testgevallen, E2E-testgegevens en E2E-testomgevingen kunnen hergebruikt worden in een performancetest. De E2E-inventarisatie is een goede basis om bottlenecks te signaleren of bevindingen te analyseren.



### 7.1.5 Onderhoudbaarheid

De benodigde inspanning voor het aanbrengen van wijzigingen. Onder wijzigingen worden correcties, verbeteringen of aanpassingen van de software of veranderingen in de omgeving verstaan.

Risico's met betrekking tot onderhoudbaarheid zijn vaak moeilijk dynamisch te testen. In het uiterste geval moeten wijzigingen in de software worden aangebracht om te kijken hoe gemakkelijk of probleemloos deze kunnen worden uitgevoerd en wat de gevolgen zijn. Dit zal alleen in extreme omstandigheden mogelijk en zinnig zijn (bijvoorbeeld bij het ontwikkelen van commerciële pakketten).

Onderhoudbaarheid wordt daarom vaak alleen statisch beoordeeld: door een analyse uit te voeren op ervaringen, het invullen van een checklist of een audit door een specialist.

#### Checklist E2E-risico's Onderhoudbaarheid

<b>Risico/aspect</b>	Analyseerbaarheid
<b>Vragen/factoren die het risico bepalen</b>	Dit betekent inzicht in de wisselwerking tussen de concrete elementen in het E2E-proces en de systeemketen. Indien het moeilijk is om resultaten te herleiden tot momenten in de systeemketen, dan is de kans groot dat een fout niet of laat wordt opgelost. Problemen in productie zijn dan moeilijk op te lossen.
<b>Testmaatregel/aandachtspunten</b>	De E2E-inventarisatie kan worden gebruikt om de analyseerbaarheid als risico in te schatten maar ook om te analyseren.
<b>Risico/aspect</b>	Wijzigbaarheid
<b>Vragen/factoren die het risico bepalen</b>	De benodigde inspanning voor aanpassing, het verwijderen van fouten of voor omgevingsveranderingen. Wijzigbaarheid kan grote gevolgen hebben voor de beheersing van het project. Bevindingen uit een E2E-test zijn dan vaak moeilijk op te lossen. Wijzigbaarheid is bijna niet expliciet te testen, maar is wel een factor om rekening mee te houden.
<b>Testmaatregel/aandachtspunten</b>	Maatregelen om risico's te beperken zijn: een gedetailleerde analyse van de E2E-keten ten einde op wijzigingen te kunnen focussen en regressietesten te kunnen bepalen.
<b>Risico/aspect</b>	Stabiliteit
<b>Vragen/factoren die het risico bepalen</b>	Het risico van onverwachte gevolgen na wijzigingen. Dit is één van de grote risico's voor het project, met name in een complexe en grote E2E-keten. De laatste kleine bugfix die een hele E2E-keten in de war gooit is één van de nachtmerries voor een E2E-test. Stabiliteit is daarmee synoniem met regressie en dient als aspect nauwkeurig te worden behandeld en getraceerd in de PRA. Welk type wijziging of bug kunnen tot welke kwaliteitsproblemen elders in de keten leiden?
<b>Testmaatregel/aandachtspunten</b>	Een goede analyse van de risico's met betrekking tot stabiliteit kan leiden tot slagvaardige regressietesten.

<b>Risico/aspect</b>	Testbaarheid
<b>Vragen/factoren die het risico bepalen</b>	De benodigde inspanning om gemodificeerde software te testen. Laat opgeleverde, maar moeilijk te testen functionaliteit, kan leiden tot uitloop of zelfs niet-geteste en niet geaccepteerde software en E2E-processen. Testbaarheid kan afhangen van: benodigde testdata, die bijvoorbeeld opnieuw door de hele keten heen moet worden geïnstalleerd en synchroon gemaakt; doorlooptijd van de betreffende testgevallen; beschikbaarheid van testomgevingen; (onmogelijke) tijdreizen.
<b>Testmaatregel/aandachtspunten</b>	Maatregelen om risico's te beperken zijn: een gedetailleerde analyse van de E2E-keten ten einde op wijzigingen te kunnen focussen en regressietesten te kunnen bepalen.

---

### 7.1.6 Portabiliteit

Portabiliteit betreft het vermogen van software om van de ene omgeving naar de andere overgezet te kunnen worden. Dit kwaliteitsattribuut is relevant waar het gaat om het inrichten van testomgevingen, maar ook waar delen van het systeemlandschap worden vervangen door andere systemen. Cloud Computing is een actueel voorbeeld waar portabiliteit van belang is: software wordt dan als een onderdeel van een E2E-proces flexibel ingezet in de E2E-keten. In zulke gevallen is een E2E-test uitermate geschikt om de portabiliteit van mogelijk andere ketenonderdelen (uit de Cloud) te beoordelen.

#### Checklist E2E-risico's Portabiliteit

<b>Risico/aspect</b>	Aanpasbaarheid
<b>Vragen/factoren die het risico bepalen</b>	De mogelijkheden voor aanpassing aan verschillende gespecificeerde omgevingen, zonder andere acties of middelen toe te passen. Aanpasbaarheid verlaagt de kans op technische en lokale regressie, omdat er geen fundamentele wijzigingen in de software zelf hoeven te worden uitgevoerd. Het gevaar is dat men hier een te groot vertrouwen door krijgt en geen oog meer heeft voor functioneel en technische wijzigingen in de keten waar systeemoverstijgende E2E-processen toch regressie van ondervinden. Vanuit de E2E-test moet er dus waakzaamheid worden betracht (in de risicoanalyse en de regressietest) indien ketenonderdelen worden vervangen.
<b>Testmaatregel/aandachtspunten</b>	Ervaringen uit de testomgevingen kunnen worden gebruikt om een eindoordeel over dit aspect te vormen. Systemen moeten dan conform beheerprocedures in testomgevingen worden aangepast.

---

<b>Risico/aspect</b>	Installeerbaarheid
<b>Vragen/factoren die het risico bepalen</b>	De benodigde inspanning voor het installeren van de software in een gespecificeerde omgeving. Indien hier een hoog risico bestaat zal een testtraject problemen kunnen krijgen bij het inrichten van testomgevingen. Het vaststellen van de hoogte van het risico en de juiste maatregelen nemen is een aangelegenheid van technisch beheerders.
<b>Testmaatregel/aandachtspunten</b>	Ervaringen uit de testomgevingen kunnen worden gebruikt om een eindoordeel over dit aspect te vormen. Systemen moeten dan conform beheerprocedures in testomgevingen worden geïnstalleerd.

---

<b>Risico/aspect</b>	Samenwerkbaarheid
<b>Vragen/factoren die het risico bepalen</b>	Het vermogen van het software product om in samenhang met andere onafhankelijke software in een gemeenschappelijke omgeving gemeenschappelijke middelen te gebruiken. Dit aspect lijkt op koppelbaarheid (zie paragraaf 7.1.1 Functionaliteit), maar de nadruk ligt hier op het gebruik van gedeelde middelen, zoals processoren, data en interfaces. Mogelijke gevolgen van een lage samenwerkbaarheid zijn verstoringen en efficiency problemen.
<b>Testmaatregel/aandachtspunten</b>	Testen die uit dit aspect volgen zijn in de regel uitvoerig en vergen veel van de testers en de testomgevingen en worden vaak gecombineerd met performancetesten en failover testen. Een maatregel kan zijn om tijdens de E2E-testen uitvoerige monitoring van gebruik van middelen uit te voeren

<b>Risico/aspect</b>	Vervangbaarheid
<b>Vragen/factoren die het risico bepalen</b>	De mogelijkheden en benodigde inspanning om het product te gebruiken in de plaats van gespecificeerde andere software in de omgeving daarvan. Cloud Computing is een wijze van werken waarbij de vervangbaarheid hoog moet zijn. Software uit de Cloud moet eenvoudig in te voegen zijn in de systeemketen of door een andere software kunnen worden vervangen. Bij een lage vervangbaarheid zijn zware regressietesten nodig: de E2E-processen en de technische werking van de keten kunnen namelijk worden geschaad. Het testen van vervangbaarheid zal in geval van Cloud Computing een belangrijke taak worden voor E2E-testen.
<b>Testmaatregel/aandachtspunten</b>	Een E2E-testomgeving waarin periodiek een aantal happyflow E2E-testgevallen worden uitgevoerd. Aan de hand van de E2E-inventarisatie kunnen relevante testgevallen worden geselecteerd in geval een deel van de E2E-keten wordt vervangen. Dit kan men uitvoeren als een Proof of Concept met mogelijke kandidaat-systemen.

## 7.2 Projectrisico's

Projectrisico's worden als volgt onderverdeeld:

1. Testbasis.
2. Planning en beheer.
3. Testomgevingen.
4. Testdata.
5. Testuitvoering.

### 7.2.1 Testbasis

Specificaties zijn veelal op systeemniveau beschreven. Hierbinnen zijn interfaces en afhankelijkheden benoemd en deze kunnen op hun onderlinge samenhang worden gereviewd. Maar de testbasis van de E2E-keten als geheel is een ingewikkelde zoektocht. De testbasis voor de E2E-test is niet, zoals bij een systeemtest, een makkelijk lokaliseerbaar functioneel ontwerp maar vergt domein-, systeem-, E2E-proces- en organisatiekennis. E2E-testers moeten hier zelf een creatieve rol spelen en zelf hun testbasis op basis van onderzoek vaststellen.

### 7.2.2 Planning en beheer

1. **Complexiteit.** Een E2E-test kent verhoudingsgewijs veel randvoorwaarden en uitgangspunten<sup>1</sup>. Naast de moeilijkheid van het inschatten van de benodigde inspanning kan dit onbeheersbaarheid betekenen van het testtraject. Hier moet één partij leidend zijn die ook een door alle betrokkenen en leveranciers gedragen autoriteit heeft. Dit kan in ieder geval niet één van de leveranciers zijn. De oplossing is een E2E-testmanager die hiërarchisch gezien naast de overall projectleider staat en die een escalatiekanaal heeft bij de organisatie. Dit escalatiekanaal zou dan idealiter een E2E-ketenmanager of releasemanager zijn, die project- en systeemoverstijgende eindverantwoordelijkheid en gezag draagt.
2. **Acceptatie.** Elke testsoort (zoals bijvoorbeeld een systeemtest) heeft een acceptant nodig: iemand die beoordeelt of aan de volgende fase begonnen kan worden. Vaak zijn dit systeemeigenaren of proceseigenaren. Als dit niet goed belegd is v.w.b. alle E2E-ketens en E2E-processen dan kan dit leiden tot vertragingen of fouten in productie.
3. **Management.** Daarnaast heeft de E2E-keten een eigenaar of manager nodig die bij meningsverschillen knopen kan doorhakken. Een risico is dat de E2E-test voor het halen van zijn doelstellingen afhankelijk is van zaken waar het zelf geen invloed op heeft (denk aan voortgang in ontwerp, bouw en test van deelsystemen, oplossen van bevindingen).
4. **Budget.** In een landschap met meerdere partijen (o.a. externe leveranciers) heeft het overkoepelende project vaak al vroeg diverse potjes ingericht voor deze partijen. Als het project van start gaat lijkt het budget daarmee al verdeeld. Veelal wordt dan pas duidelijk, vanuit de E2E-testmanager, dat er ook nog een budget nodig is voor de E2E-test. Maar vanuit welk potje moet dit worden betaald? De E2E-test zal vanuit de klant vaak worden gezien als een veredelde systeemtest, maar elk van de systeemtesten (lees: leveranciers) zal zich niet geroepen voelen de E2E-test binnen zijn scope en budget te plaatsen. Daarnaast ziet men budgetoverschrijding tijdens de E2E-test ten gevolge van functionele fouten (die door een systeemtest gevonden hadden moeten worden). Hoe kan de E2E-test dan nog bewijzen aan zijn eigen succescriteria te hebben voldaan? Schuld en rekening komen vaak te liggen waar problemen zich tonen, niet waar ze veroorzaakt zijn of waar ze hadden moeten worden gevonden.
5. **Scope.** Een E2E-test kan een vergaarbak worden voor het project van alle voorgaande activiteiten die in de geplande fase niet volledig zijn uitgevoerd ("we hebben toch nog een E2E-test?"). Hoewel een E2E-test hier vaak niet aan kan ontkomen, betekent het een wijziging op de originele scope, planning en budget en er moet hier dan ook rekening mee worden gehouden.
6. **Configuratiemanagement.** Dit betreft documenten tot softwarecomponenten, testdata, testgevallen en testresultaten. In een complexe omgeving en organisatie bestaat het gevaar dat niet iedereen op de hoogte is van wijzigingen. Gevolg kan zijn dat de waarde van testen wordt beïnvloed. Uiteindelijk moet een test uitspraak kunnen doen over datgene (juiste versie) wat in productie gaat.

<sup>1</sup> Denk hierbij aan: kwaliteit per systeem na systeemtest, uitloop van systeemtest en impact op de E2E-test, inrichten en beheer testomgevingen en testdata, inzet van personeel voor ondersteuning en uitvoering bij de diverse partijen (met betrekking tot omgevingen, testen, fixen).

### 7.2.3 Testomgevingen

1. **Beheer.** Een E2E-testomgeving is door zijn omvang en eisen qua productiegelijkheid vaak lastiger te beheren. Hier is samenwerking nodig tussen testers, beheerders en project- en testmanagement. Wie gaat wat precies doen en wanneer? Heeft iedereen de benodigde kennis en rechten?
  - a. Het is aan te raden het beheer van de testomgevingen door de mensen te laten doen die dit ook in productie doen. Niet alleen wordt daarmee voorkomen dat het beheer niet goed wordt uitgevoerd, ook wordt het testtraject daarmee een test van de beheerprocessen.
  - b. De E2E-board kan hier zijn nut bewijzen: met een dagelijks panel van o.a. beheerders kan efficiënt samenwerking worden bereikt.
  - c. Het beheer van E2E-testomgevingen kan als een subproject worden ingericht: alle activiteiten, stappen, afhankelijkheden, mijlpalen en producten, voor- tijdens en na testuitvoering, worden nauwkeurig vastgesteld en beheerd door de E2E-testmanager.

2. **Beveiliging.** Wie verlenen we toegang tot welke omgevingen als testen bij andere partijen worden uitgevoerd? Als we gebruik maken van productiedata mogen dan alleen productiemedewerkers testen? Moet data worden geanonimiseerd?
3. **Autorisatiestructuur.** Is het wel efficiënt om de autorisatiestructuur productiegelijk in te richten, waarbij testers per handeling zich afvragen met welke rol of account iets moet worden uitgevoerd? Een manier om hier mee om te gaan is om de testuitvoering op te delen in een aantal testronden. De eerste testronden worden dan met een eenvoudige open autorisatiestructuur uitgevoerd en in ten minste één testronde wordt getest met een productiegelijke autorisatiestructuur.

#### 7.2.4 Testdata

De risico's rond testdata hebben allen betrekking op de waarde en de uitvoerbaarheid van de testen. Gevolgen kunnen onterecht ingediende bevindingen of uitloop zijn.

1. Het vaststellen van benodigde data vereist algemene ketenkennis en specifieke kennis per systeem. Indien dit niet goed wordt ingevuld kunnen er tijdens de testuitvoering allerlei problemen ontstaan, zoals het niet kunnen uitvoeren van testgevallen.
2. Het verkrijgen, onderhouden en beheren van testdata vergt organisatorische afstemming. Gevaar is dat de enige mensen die dit kunnen (dit zijn vaak de functionele en technische beheerders) (nog) niet verantwoordelijk zijn voor de testdata.
3. Hoe verkrijgen en houden we datasynchroniciteit door de keten heen als bijvoorbeeld voor de ene test in het verleden en voor de andere in de toekomst moet worden gewerkt? Wie heeft de kennis en de mogelijkheid om in de E2E-testomgeving deze zaken te waarborgen?
4. Veiligheid (privacy) van data. Hoe depersonaliseren we eventuele gegevens uit productie door de keten heen? Depersonaliseren kan een tijdrovend proces zijn en indien het niet goed wordt uitgevoerd kunnen testomgevingen onbruikbaar worden en testen uitlopen.
5. Testdata- en testomgevingenbeheer kunnen samen als een subproject worden ingericht: alle activiteiten, stappen, afhankelijkheden, mijlpalen en producten, voor- tijdens en na testuitvoering, worden nauwkeurig vastgesteld en als een detailplanning beheerd door de E2E-testmanager. Het gaat dan om zaken als ophalen dataloads, uitvoeren van depersonalisaties en back-ups.

#### 7.2.5 Testuitvoering

1. **Bemensing.** Wie gaat de E2E-test uitvoeren? Kan dit een partij zijn die commerciële belangen heeft zoals leveranciers? Zijn dit de systeemtesters die vervolgens hun eigen systeem in de E2E-ketentestomgeving gaan testen? Zijn het de gebruikers, beheerders of zelfs klanten? Als dit niet goed wordt gepland en afgesproken kunnen testgevallen niet of niet goed uitgevoerd worden.
2. **Bevindingen.** Hoe wordt er omgegaan met bevindingen? Er kunnen conflicten ontstaan tussen de diverse ontwikkelpartijen met als gevolg uitloop en extra kosten. Met name E2E-testgevallen zijn moeilijk te analyseren en verantwoordelijkheden ten aanzien van oplossen zijn moeilijk vast te stellen of te beleggen ("de fout zit in het andere systeem"). Hier is een link met integraal E2E-ketenmanagement: wie neemt uiteindelijk de beslissing, hoe iets moet worden opgelost en wie van de partijen moet dat gaan doen?
3. **Vaardigheden.** Het profiel van een E2E-tester is anders dan dat van een systeemtester. Een E2E-tester heeft kennis van de delen van de E2E-keten nodig of moet dit snel kunnen opbouwen. Daarnaast moet zij als persoon flexibel, doortastend, zakelijk en vooral neutraal met de diverse partijen en belangen om kunnen gaan. Goede E2E-testers zijn vitaal voor een E2E-testteam. Indien de invulling van testtaken niet bij de juiste persoon is belegd is uitloop van de testen onvermijdelijk. Testgevallen worden niet efficiënt uitgevoerd en belangrijke fouten worden niet of te laat gevonden.



# Deel 8

## E2E-testplanning

## Deel 8 E2E-testplanning

### 8.1 Inleiding E2E-testplanning

Elke testmanager en testcoördinator weet hoe lastig het opstellen van een testplanning is. Een testtraject is, meer dan welke andere projectactiviteit ook, afhankelijk van andere activiteiten. Elk testtraject heeft daarom een mate van onvoorspelbaarheid. Dit geldt in toenemende mate voor een Systeemintegratietest en nog meer voor een E2E-test. De doorlooptijd en hoeveelheid uren worden grotendeels beïnvloed door zaken waar testers maar weinig invloed op hebben, zoals de kwaliteit van de software, de ontwerpen of de plannings van bouw en het ontwerp per elk van de deelsystemen.

De testplanning van een E2E-test is daarom geen exacte wetenschap. Formele plantechieken zoals Testpuntanalyse hebben vrijwel geen praktische betekenis. Maar het niet afgeven van een planning is ook geen optie. Anderen, projectmanagement en afdelingen die mensen moeten leveren, hebben vroeg een indicatie nodig van wat hun te wachten staat. Bovendien moet de inzet van testers en verwachte diensten door anderen vroeg worden afgestemd en onderbouwd.

Het plannen van een test is een continue activiteit die er in bestaat alle activiteiten te groeperen in testclusters en de afhankelijkheden aan te geven. Een planning die in een vroeg stadium is gemaakt zal meer onnauwkeurigheden kennen. Wijzigingen in het halen van mijlpalen kunnen leiden tot een significante bijstelling van de planning: denk aan daadwerkelijke momenten van oplevering van systeemdelen, de geconstateerde daadwerkelijke kwaliteit tijdens een systeemtest of de daadwerkelijk bestede uren tijdens testontwerp in verhouding tot de geplande uren.

Een planning moet daarnaast instrumenten bieden om te kunnen sturen op tijd, geld en kwaliteit. Er moet een optie zijn om aan te geven wat de gewonnen doorlooptijd is indien met kiest voor een lagere kwaliteit. Een dergelijke menukaart van de testen veronderstelt een structuur waarbij project- en productrisico's kunnen worden gerelateerd aan testgevallen. Testgevallen die betrekking hebben op lagere risico's kunnen dan bijvoorbeeld worden uitgesloten.

Ten slotte moet de planning worden uitgewerkt voor wat betreft doorlooptijden, uren, kosten en bemensing. Deze zaken hebben weer onderlinge relaties. Indien bijvoorbeeld het testontwerp door een meer ervaren E2E-tester kan worden uitgevoerd kan dit voordelig zijn voor de doorlooptijd en de uren van de testvoorbereiding, maar ook voor de beheersbaarheid van de testuitvoering.

De E2E testplanning bestaat uit de volgende aspecten. Deze worden hierna uitgewerkt:

- Uitgangspunten: plantechieken
- Vaststellen testclusters
- Inventarisatie van de activiteiten en de afhankelijkheden
- Doorlooptijd testuitvoering
- Inzet testuitvoering
- Invloeden op de planning
- Project- en organisatieoverschrijdende E2E-planningen

### 8.2 Uitgangspunten

#### Formele plantechieken beperkt bruikbaar

Formele plantechieken, zoals functiepuntanalyse en het daarop gebaseerde testpuntanalyse, kunnen in een beheerste en met andere projecten vergelijkbare situatie worden ingezet. Dit geldt in de regel niet voor E2E-testen. Deze technieken zijn namelijk vooral gebaseerd op systeemontwikkeling van één systeem en de daarbij horende testen. Natuurlijk kunnen deze technieken worden ingezet, maar de waarde van gegevens uit zo'n analyse ontstaat pas na het vergelijken met ervaringscijfers uit meerdere projecten en het vinden van patronen.



Voor wat betreft een systeemintegratietest en een E2E-test zouden deze cijfers op een vergelijkbare manier gebruikt kunnen worden als één van de uitgangspunten voor de planning. In de loop der tijd kunnen patronen in deze getallen worden vastgesteld en gebruikt (bijvoorbeeld: de verhouding tussen een FPA van systeem X en de E2E-test is tussen 1:12 en 1:15). De ervaring leert echter dat SIT en E2E vanwege de vele variabelen moeilijk te vergelijken zijn met vorige trajecten.

### Industrie standaarden alleen initieel en beperkt bruikbaar

Voor wat betreft het afgeven van testplanningen zingen de zogenaamde "industrie standaarden" al jaren rond. Voor het gemiddelde aandeel van testen in projecten worden de volgende getallen gehanteerd:

Testen: 35% van het totale project.

Hiervan: 5-7 % voor unittesten, 10% voor systeemtesten en 18-20% voor acceptatietesten.

Onder acceptatietesten worden de SIT, E2E-test en de gebruikersacceptatietesten verstaan.

Hoewel deze getallen al decennia oud zijn, zijn ze, indien meer gegevens ontbreken, nog wel bruikbaar als een indicatie bij de start van een project. Het totale percentage en de onderlinge verhoudingen tussen de testsoorten zullen echter afwijken, zeker in het geval sprake is van een complexe keten. Het aandeel acceptatietesten moet dan voor een deel worden besteed aan Systeemintegratietesten en E2E-testen.

In sommige studies wordt een ander beeld geschetst van de huidige te verwachten verhoudingen tussen het project als geheel en testen, en binnen testen de verhouding tussen integratie of E2E-testen en andere testen. In complexe en samenhangende systeemlandschappen kan er zelfs sprake zijn van 50-70% van het totale project besteed aan testen en binnen het testen weer 50-70% besteed aan integratietesten!<sup>1</sup>

Daarnaast worden ook gemiddelden gehanteerd binnen de fasen van een testsoort:

**Tabel 1: Industrie standaarden: fasen als % van geheel**

Activiteit	Goede, complete testbasis	Onvoldoende testbasis
<b>Vorbereiding</b>	6 %	21 %
<b>Specificatie</b>	54 %	33 %
<b>Uitvoering</b>	21 %	24 %
<b>Afronding</b>	2 %	5 %
<b>Beheer/Infrastructuur</b>	17 %	17 %

Deze getallen zijn voor een groot deel gebaseerd op observaties tijdens systeemtesten. Voor een E2E-test zijn naar verwachting de uren voor testuitvoering en beheer groter. In ieder geval zal testuitvoering tijdens een SIT of E2E-test nadelige effecten hebben van te late deelopleveringen, onverwachte blokkerende bevindingen of beschikbaarheidsproblemen met betrekking tot de testomgeving. Testuitvoering van een E2E-test is bovendien intensiever, omdat er met zo productiegelijk mogelijke draaischema's zal worden gewerkt. Dit levert meer leeglooptijd (wachtijd, waste time) op, omdat niet alles op elke dag uitgevoerd of gecontroleerd kan worden. Vaak moet men bijvoorbeeld wachten op de verwerking door de volgende nachtruns. Hierdoor zal het beheer van de Infrastructuur arbeidsintensiever zijn dan in een systeemtest.

<sup>1</sup> Zie "E2E Integration Testing Design", Tsai, Bai, Paul, Shao, Agarwal.

### Marges op de nauwkeurigheid van de testplanning

Vanwege de grote mate van onnauwkeurigheid van vooral initiële planningen is het van belang om tijdens het traject een aantal vaste momenten af te spreken waarop een bijgewerkte, preciezere planning kan worden afgegeven. De volgende momenten en hun mate van nauwkeurigheid kunnen worden aangehouden:

**Tabel 2: Momenten planning en nauwkeurigheid**

Moment planning	Nauwkeurigheid	Relevante feiten	Document
<b>Kort na projectplan</b>	40%		Testplanning
<b>Na E2E-inventarisatie</b>	50%	Inzicht complexiteit E2E-processen	Testplanning
<b>Na teststrategie</b>	60%	Inzicht risico's en te gebruiken testaanpak	E2E-testplan/ Mastertestplan
<b>Na testontwerp</b>	65%	Daadwerkelijke testontwerp uren versus geplande uren	Voortgangsrapport
<b>Na intake testobject</b>	75%	Kwaliteit opgeleverde systemen, beschikbaarheid omgeving	Voortgangsrapport
<b>Tijdens testuitvoering</b>	75% -> 100%	Meer of minder bevindingen dan verwacht, beschikbaarheid omgevingen meer of minder dan verwacht,...	Voortgangsrapport

### Te gebruiken planmethoden: WBS, extrapolatie en verbeeldingskracht

De hierboven genoemde cijfers geven alleen een initiële indicatie en mogen nooit als vaststaande planning worden afgegeven. Om een E2E-test gefundeerd te plannen zijn twee bekende technieken de basis: "Work Breakdown Structure" (WBS) en extrapolatie. Dit zijn basale methoden die tot doel hebben om beschikbare kennis van het testproces en de uit te voeren activiteiten te ordenen (WBS) en verkregen ervaringscijfers uit te werken tot een completere planning (extrapolatie).

Deze methoden moeten worden aangevuld met de verbeelding en de intuïtie van de E2E-testmanager en zijn testers. Het belangrijkste onderdeel van de activiteit "plannen" is namelijk het voorstellen van alle mogelijke activiteiten en afhankelijkheden en hoe dit daadwerkelijk zal gaan verlopen en wat er zou kunnen gebeuren. De E2E-testmanager moet zich voorstellen hoe de testuitvoering op dag 1 zal starten: mensen loggen bijvoorbeeld in op de testomgeving en treffen een applicatie aan waar mogelijk bepaalde stuurgegevens kunnen missen. De gevolgen zijn bijvoorbeeld dat nieuwe polissen niet kunnen worden aangemaakt. De E2E-testmanager moet dan een controle inplannen, voorafgaand aan de testuitvoering, waarin stuurtabellen worden gevuld en gecontroleerd.

### 8.3 Vaststellen testclusters

Testclusters zijn voor wat betreft risico's, uitvoering, en planning samenhangende verzamelingen testactiviteiten. Een testcluster kan bijvoorbeeld de functionele test zijn van het E2E-proces "offerte". Een ander cluster kan de stresstest van hetzelfde E2E-proces zijn.

Testclusters kunnen verschillen in afhankelijkheden en daarom een afwijkende planning hebben. Voor het ene testcluster kan het opleveren van interface A randvoorwaardelijk zijn en voor het andere testcluster kan dit interface B of systeem C zijn. Als de interface later wordt opgeleverd kan het ene testcluster wel en het andere niet al worden getest. Bovendien kan de bemensing, de infrastructuur en de doorlooptijd van testclusters verschillen. Vandaar dat een E2E-testplanning de opdeling in afzonderlijke testclusters als uitgangspunt neemt. De opdeling in testclusters gebeurt tijdens het opstellen van de teststrategie.

## 8.4 Inventarisatie van activiteiten en afhankelijkheden

Het faseringsmodel van een E2E-test komt overeen met die van een systeemtest. In de opzet van de planning kan van de generieke lijst met activiteiten, zoals voorgesteld in methodieken als TMap of ISTQB, worden uitgegaan. Hieronder volgen, per fase in het testproces, opmerkingen over specifieke aandachtspunten voor een E2E en de gevolgen daarvan voor de planning. Het faseringsmodel van ISTQB is hierbij gekozen als uitgangspunt.

**Tabel 3: Aandachtspunten per fase**

Fase	Aandachtspunten voor E2E-test	Gevolgen voor planning
<b>Planning en beheer</b>	Meer overleg met diverse organisatie-onderdelen over risico's, bemensing en E2E-processen	Langere doorlooptijd, meer betrokkenheid diverse expertisegroepen
<b>Analyse en ontwerp</b>	Testbasis moet zelf verzameld worden. Testontwerp vergt veel overleg met diverse gebruikersgroepen	Langere doorlooptijd, meer betrokkenheid diverse expertisegroepen
<b>Implementatie</b>	Testomgeving is complexer: meer overleg en afstemming over oplevering, data en beheer	Langere doorlooptijd, meer betrokkenheid diverse beheergroepen
<b>Uitvoering</b>	Meer afhankelijkheden, zoals: kwaliteit van diverse onderdelen, oplevermomenten van systeemonderdelen, betrokkenheid meer en meer diverse groepen. Bevindingen hebben grotere gevolgen voor doorlooptijd, hertesten moeizamer	Langere doorlooptijd, grotere projectrisico's. Meer overleg over bugfixing en testresultaatanalyse. Risico van moeizaam op te lossen bugs
<b>Evaluëren exit criteria</b>	Complexer, meerdere acceptanten, resultaten minder eenduidig	Langere doorlooptijd, meer betrokkenheid diverse expertisegroepen

## 8.5 Planning, beheer en analyse

De E2E-testmanager en E2E-testers beginnen met de eerste inventarisatie van de E2E-processen en de samenhang met de systemen. Op basis van hun eerste ervaringen maakt de E2E-testmanager een planning op hoofdlijnen, waaruit moet blijken of er nog meer E2E-testers nodig zijn. Op deze manier vormt zich het E2E-kernteam. Dit team voert de risicoanalyse, de E2E-inventarisatie en het logisch testontwerp uit. Naast het E2E-kernteam moet voor de leden van de E2E-board een aantal uren per worden ingepland. Dit is ongeveer 2 tot 4 uur per week per persoon tot de testuitvoering.

De doorlooptijd van de fase Planning, beheer en analyse is afhankelijk van de status van het project, de beschikbaarheid van procesbeschrijvingen en de beschikbaarheid van experts uit de organisatie. De fase planning en analyse zal in de regel tussen de 4 en 6 weken in beslag nemen.

In te plannen activiteiten zijn:

### Planning:

- Plaatsbepaling binnen de organisatie.
- Definitie van de E2E-test binnen het project.
- Initiatie van de E2E-test.
- Bepalen testbasis.
- Bepalen teststrategie.
- Planning.

**Analyse:**

- Testbasis per E2E-proces detailleren.
- Detailrisicoanalyse per testcluster uitvoeren.
- Testtypeplan (detailplan) opstellen (optioneel).

## 8.6 Ontwerp en implementatie

Binnen ISTQB worden logische testgevallen opgesteld binnen de fase Analyse en Ontwerp en worden fysieke testgevallen opgesteld binnen de fase Implementatie en uitvoering. De activiteiten vloeien in elkaar over en hebben elk dezelfde bemensing nodig.

Het gaat om de volgende activiteiten:

**Ontwerp:**

- Logische testgevallen uitschrijven.
- Omgevingen detailleren.
- Testdata opzetten.

**Implementatie:**

- Fysieke testgevallen uitwerken.
- Testomgevingen inrichten.
- Bevindingenbeheer inrichten.
- Draaiboek per testomgeving creëren.

Het E2E-kernteam zal hier de meeste activiteiten uitvoeren, maar in het uitwerken van de testgevallen zal er ook betrokkenheid nodig zijn van experts uit de organisatie, zoals gebruikers, managers en functioneel beheerders. Daarnaast blijft de E2E-board operationeel (ongeveer 2 tot 4 uur per lid per week).

Bij het uitschrijven van de testgevallen moet per te testen E2E-proces rekening gehouden worden met de betrokkenheid van +/- 2 experts (bijvoorbeeld vanuit de BackOffice en vanuit de financiële afdeling) voor 2 tot 4 uur per week. Het E2E-kernteam zal de testgevallen uitschrijven maar experts beoordelen de testgevallen op zinnigheid, risicodekking en praktische uitvoering.

De bemensing en de doorlooptijd moeten worden afgestemd op de verwachte startdatum van de testuitvoering.

## 8.7 Doorlooptijd testuitvoering

Een E2E-test zit tijdens de uitvoeringsfase meestal op het kritieke pad van het project. De doorlooptijd van de E2E-test moet daarom beheersbaar zijn. De doorlooptijd zal echter niet alleen maar bepaald worden door het aantal uren werk en de beschikbaarheid van mensen. De doorlooptijd van een E2E-test wordt in eerste instantie bepaald door het aantal benodigde testronden en de minimale lengte van testgevallen. Daarnaast zijn er omstandigheden die moeilijk te beheersen zijn, maar die wel een grote invloed op de doorlooptijd kunnen hebben, zoals de beschikbaarheid van testomgevingen, de kwaliteit van opgeleverde systemen en de beschikbaarheid van beheerders om activiteiten ten bate van de E2E-test uit te voeren.

**Vaststellen benodigde testronden**

Een testronde is een cyclus waarin testdata en systeemtijden weer op een beginstand worden gezet, zodat testgevallen weer kunnen worden gestart.

Er zijn in de regel 4 testronden nodig:

- 1. Initiële ronde:** "Hit and run". Basisscenario's zo snel mogelijk testen en daarmee de zwaarste fouten zo vroeg mogelijk vinden.
- 2. Doortestronde:** blokkerende fouten moeten zijn opgelost. Het gaat in deze ronde om het dekken van het merendeel van de testgevallen in een volledige keten. Doel is om na deze ronde alle zware en belangrijke

fouten te hebben gevonden.

3. **Hertesten en exoten:** hierin wordt de rest van de testgevallen uitgevoerd die in ronde 2 niet uitgevoerd kunnen worden en dienen de meeste en belangrijkste fouten te zijn opgelost en te worden gehertest.
4. **Regressietest:** resterende bugfixes worden gehertest en een selectie van alle testgevallen wordt nogmaals uitgevoerd om regressie door wijzigingen en bugfixes te dekken.

Het daadwerkelijke aantal testronden hangt af van:

- Het aantal testgevallen dat in één dezelfde testronde uitgevoerd kan worden.
- Of systeemtijden of instellingen moeten worden terug gezet (bijvoorbeeld omdat door het draaien van batches de systeemtijd te ver in de toekomst is gekomen).
- Of de testdata verbruikt is (alle testgegevens zijn gewijzigd en voor hertesten moeten ze weer op een beginstand staan).
- Hoeveel opleveringen er nodig zijn.
- In hoeverre alle systemen in de infrastructuur zijn opgeleverd.
- De beschikbaarheid en kwaliteit van eventuele conversiedata.
- Het oplossen van belangrijke fouten.
- Wijzigingen in E2E-processen of systemen.

Zie [Deel 6 - E2E-teststrategie](#) voor een meer gedetailleerde beschrijving van het vaststellen van testronden.

### Doorlooptijd per testronde

Per testgeval moet worden vastgesteld wat de stappen en de handelingen zijn. Dit dient te gebeuren bij de fysieke uitwerking van de logische testgevallen. Dit is onderdeel van het testontwerp en zal dus voor de planning pas na deze activiteit definitief kunnen worden vastgesteld.

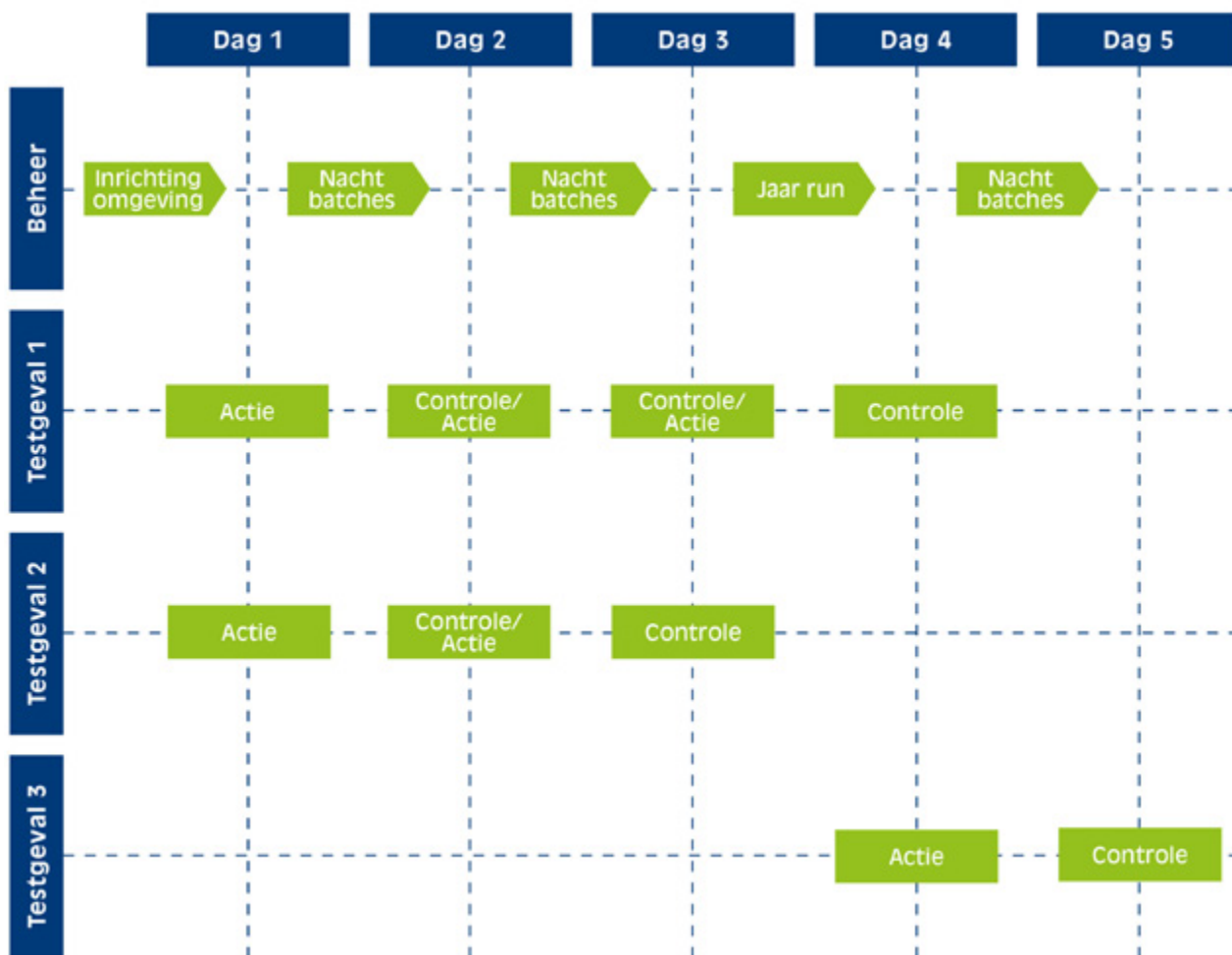
Het aantal stappen is afhankelijk van het aantal nachtruns, maandruns en dergelijke. De totale doorlooptijd van de testronde is echter niet altijd identiek aan de doorlooptijd van het langste testgeval. Dit komt doordat niet alle testgevallen tegelijk uitgevoerd kunnen worden of omdat acties in de testomgeving niet voor elk testgeval gelegen komen.

Een voorbeeld hiervan is een testgeval waarbij een verzekeringspolis moet worden verlengd (geprolongeerd). Eerst moet de polis worden aangemaakt. Dit neemt al snel twee doorlooptijden in beslag (aanmaken, nachtrun, polis krijgt definitieve status). Hierna moet een sprong naar het prolongatiemoment worden gemaakt. Hiervoor moet beheer een aantal tijdreisacties uitvoeren in de testomgeving, waarna een nachtrun moet draaien. De volgende dag kan de prolongatie worden gecontroleerd. Dit testgeval neemt dan totaal minimaal vier dagen in beslag, maar zal niet op elk willekeurig moment uitvoerbaar zijn omdat andere testgevallen wellicht deze tijdsprong niet moeten maken. Testgevallen en hun onderlinge afhankelijkheden en belemmeringen beïnvloeden op deze manier de totale doorlooptijd.

Per testronde en testomgeving moet daarom een puzzel worden opgelost waarin alle stappen en acties per testgeval en hun invloed op andere testgevallen moeten worden samengevoegd in één detailplanning.

Belangrijke componenten daarbij zijn:

1. Doorlooptijd van handelingen en controles per testgeval.
2. Maximale hoeveelheid handelingen die dagelijks kunnen worden uitgevoerd.
3. Acties op de testomgeving die de uitvoerbaarheid van testgevallen beïnvloeden.
4. Vervolgacties en de doorlooptijden na de acties uit 3.
5. Welke testgevallen in welke testronde moeten worden uitgevoerd (bijvoorbeeld per systeempriode: verleden, toekomst, bepaalde perioden en momenten).



Figuur: Voorbeeld doorlooptijd testronde

In dit voorbeeld zijn er **3 testgevallen**:

**Testgeval 1** moet worden ingevoerd. Vervolgens moet er worden gecontroleerd wat de resultaten zijn na nachtbatches en er zijn vervolgacties en controles nodig. Bovendien moet voor dit testgeval een jaarovergang worden getest.

**Testgeval 2** moet worden ingevoerd en het resultaat moet na een nachtbatch en vervolgacties worden getest.

**Testgeval 3** betreft een invoeracties en controle in het volgende jaar.

Deze drie testgevallen leveren samen een doorlooptijd op van vijf dagen.

De impact op de doorlooptijd kan een overweging zijn om bepaalde testgevallen (bijvoorbeeld die met complexe tijdreisscenario's) niet meteen in de eerste testronde uit te voeren of deze zelfs op een aparte testomgeving uit te voeren.

Het werken met gegevens uit productie kan een positieve invloed hebben op de doorlooptijd: het testgeval uit het voorbeeld, met een verlenging na een jaar, kan wellicht al gevonden worden in een populatie polissen die men op de testomgeving vanuit productie heeft geladen. Dit kan dan doorlooptijd besparen, omdat in de testgegevens geen historie hoeft te worden aangemaakt.

## 8.8 Inzet testuitvoering

De inzet kan beter in dagen dan in uren worden berekend. De doorlooptijd is een goede basis voor de inzetberekening. In het ergste geval zijn alle betrokkenen gehele dagen ingezet op de E2E-test. Dan geldt als de totale inzet: de totale doorlooptijd maal het aantal betrokkenen. Hiervan kunnen dan de dagen (of uren) worden afgetrokken die mensen op andere taken zijn ingezet.

Uit de fysieke testgevallen kan afgelezen worden welke acties en controles per dag moeten worden uitgevoerd. Indien deze acties door specifieke medewerkers moeten worden uitgevoerd (bijvoorbeeld invoeren en muteren door administratieve medewerkers van de BackOffice en controles door medewerkers van de financiële administratie) dan kan de benodigde inzet hier afgelezen worden. Zo zijn leden van de E2E-board slechts een aantal uren per week ingezet op de E2E-test. De doorlooptijd kan dan als uitgangspunt dienen.

Het is raadzaam om op voorhand werkzaamheden te verzamelen die geen dringende einddatum hebben en die eenvoudig in brokstukken kunnen worden opgedeeld. Testers kunnen dan bij uitloop en wachttijden hierop ingezet worden.

Een pragmatische oplossing voor inzet van gebruikers tijdens de E2E-testen is om Gebruikersacceptatietesten, Beheeracceptatietesten en E2E-testen te combineren. Dit kan door testontwerp en coördinatie bij het E2E-testteam te beleggen en de testuitvoering te beleggen bij de betreffende gebruikersgroep. Elke handeling in de testuitvoering wordt dan bijvoorbeeld uitgevoerd door de gebruiker die dat in productie ook doet. Dit geldt dan ook voor beheerders.

Door tijdens de E2E-testen gebruikers acceptatiecriteria voor hun eigen Acceptatietesten te laten beoordelen zal een groot gedeelte van de Acceptatietesten met de E2E-testen worden gedekt en heeft men een goede onderbouwing voor inzet van gebruikers tijdens de E2E-testen. Bovendien zal de kennis van gebruikers en beheerder de E2E-test ondersteunen.

## 8.9 Invloeden op de planning

Als de bovengenoemde doorlooptijd en inzet wordt berekend ontstaat een planning die als basis fungeert. In de praktijk zijn er diverse beïnvloedingsfactoren waarmee rekening gehouden moet worden.

Hierbij een opsomming van zaken die van invloed kunnen zijn op de planning van een E2E-test.

- **Bemensing:** het niet ter beschikking hebben van experts met proceskennis, ervaren beheerders of goede E2E-testers. Een planning begint daarom met een ideaalplaatje: de juiste mensen zijn op het juiste moment beschikbaar. Indien in het verloop van het traject er een kans is dat de praktijk tegenvalt, moet hiervoor een opslag in de doorlooptijd en de planning worden genomen.
- De zogenaamde **"idle time"** tussen testronden: de hoeveelheid tijd die in beslag wordt genomen door het installeren van nieuwe versies, maken en laden van back-ups, aanmaken van testdata, aanpassen en controleren van instellingen. Deze activiteiten moeten vooraf worden afgestemd met de betreffende beheerpartijen en de status en voortgang moet worden bewaakt door de E2E-testmanager.
- **Functionele bevindingen** kunnen een grote invloed hebben op de doorlooptijd en daarmee de inzet. Ten eerste neemt de analyse van onverwachte uitkomsten in verhouding veel tijd in beslag tijdens een SIT of E2E-test. Daarnaast betekent elke bevinding dat een deel van de testen mogelijk opnieuw moet worden uitgevoerd. Functionele bevindingen kunnen zelfs een deel van de E2E-testgevallen blokkeren. In de planning moet hier rekening mee worden gehouden (in de hoeveelheid testronden maar ook in de te bestede uren aan resultaatanalyse). Tegelijk moet ook een voorspelling worden gedaan, bijvoorbeeld op basis van een inschatting van de diepgang van voorgaande testen, van de kans op het aantreffen van functionele bevindingen. Ook moet men inschatten wat de mate is waarin E2E-aspecten al door anderen zijn gedekt. Het ontbreken van goede interfacetesten tijdens de systeemtest is bijvoorbeeld een groot risico voor de SIT of E2E-test.
- **Technische storingen op de testomgeving.** In geval van nieuwe hardware, nieuwe beheerpartijen of wijzigingen in het beheerproces nemen niet alleen de kans op verstoringen toe, ook de kans dat verstoringen snel verholpen kunnen worden, nemen af. De impact hiervan op de doorlooptijd kan groot



zijn: het kan betekenen dat de testomgeving tijdelijk niet beschikbaar is en indien door de verstoring de testdata wordt beschadigd (aanwezigheid, integriteit of synchroniciteit) kan de waarde van testronden verminderen.

- **Bekendheid met E2E-processen:** in geval nieuwe of ingrijpend gewijzigde E2E-processen onderdeel zijn van de test, is er een grote kans dat tijdens testontwerp en testuitvoering discussies worden gevoerd over de exacte werking en de vormgeving van de E2E-processen. Deze discussies hebben impact op de doorlooptijd van de testen. Daarnaast kunnen uit deze discussies nog grote bevindingen en wijzigingen in zowel het E2E-proces als de software voortkomen. De impact op de doorlooptijd kan dan groot zijn. Na de wijzigingen moeten mogelijk testronden opnieuw worden uitgevoerd. Nog beter is het om de E2E-processen zo concreet en vroeg mogelijk al door te spreken met de betrokkenen ten einde potentiële bevindingen en discussies van het kritieke pad tijdens de testuitvoering af te halen.
- **Bekendheid met beheerprocessen:** in geval er grote wijzigingen zijn in het beheer van systemen (en daarmee de testomgevingen) kan dit grote invloed hebben op de doorlooptijd en de inzet. Het kan dan gaan om wijzigingen in het beheerproces, de beheerorganisatie of de systemen die moeten worden beheerd. De impact hiervan op de doorlooptijd van testen kan bestaan uit: problemen in de beschikbaarheid van systemen of testgegevens, moeizamere analyse van testresultaten of het moeizamer uitvoeren van benodigde batches op de testomgeving. In het ergste geval wordt de waarde van een testronde onderuit gehaald en moet een testronde opnieuw worden gestart. Het is van belang dat, voorafgaand aan de start van testuitvoering, met de beheerpartijen een gedetailleerd draaiboek van het omgevingsbeheer wordt opgesteld. In dit draaiboek moet per dag worden benoemd welke beheeractiviteit door wie moeten worden uitgevoerd<sup>2</sup>.

Van deze factoren moet worden vastgesteld of de invloed en kans hoog, middel of laag is. Een hoge invloed resulteert in een hogere opslag voor de doorlooptijd en de inzet. Het is onmogelijk om hier een generieke, universele rekenformule van te maken, maar de praktijk leert dat bij veel factoren met een hoge invloed op de planning de doorlooptijd en inzet met meer dan 100% kunnen toenemen! Daarnaast moet bij dit soort risico's de te nemen maatregelen worden geformuleerd.

## 8.10 Project- en organisatieoverschrijdende E2E-testen

In toenemende mate voeren meerdere projecten tegelijk wijzigingen uit op dezelfde E2E keten en mogelijk zelfs op hetzelfde systeem. Dit vergt een organisatiebreed georganiseerde E2E-test, waarvan de kosten gezamenlijk door de projecten gedragen worden. Ieder project heeft belang bij een goed uitgevoerde E2E test voor het beperken van risico's en moet – wellicht naar rato van projectgrootte – bijdragen aan het budget voor de E2E-test.

Bij het plannen moet de E2E-testmanager rekening houden met de verschillende projecten. In volwassen organisaties bestaat er releasemanagement waar aansluiting bij gevonden kan worden. Bij het plannen moet extra rekening gehouden worden met:

- Planning en status van belangrijke opleveringen en mijlpalen.
- Planning en status van de diverse testen.
- Productrisico's per project per systeem.
- Invloeden van projecten op elkaar (doorlooptijd, regressie).
- Inzet van projectleden bij projectoverschrijdende testen.
- Inzet van mensen bij andere organisaties (systemen die binnen de E2E-keten vallen maar waarvan beheer en test buiten de eigen organisatie vallen).

<sup>2</sup> Zie voor een praktijkvoorbeeld van een draaiboek [Deel 12: Checklists en voorbeelden](#).



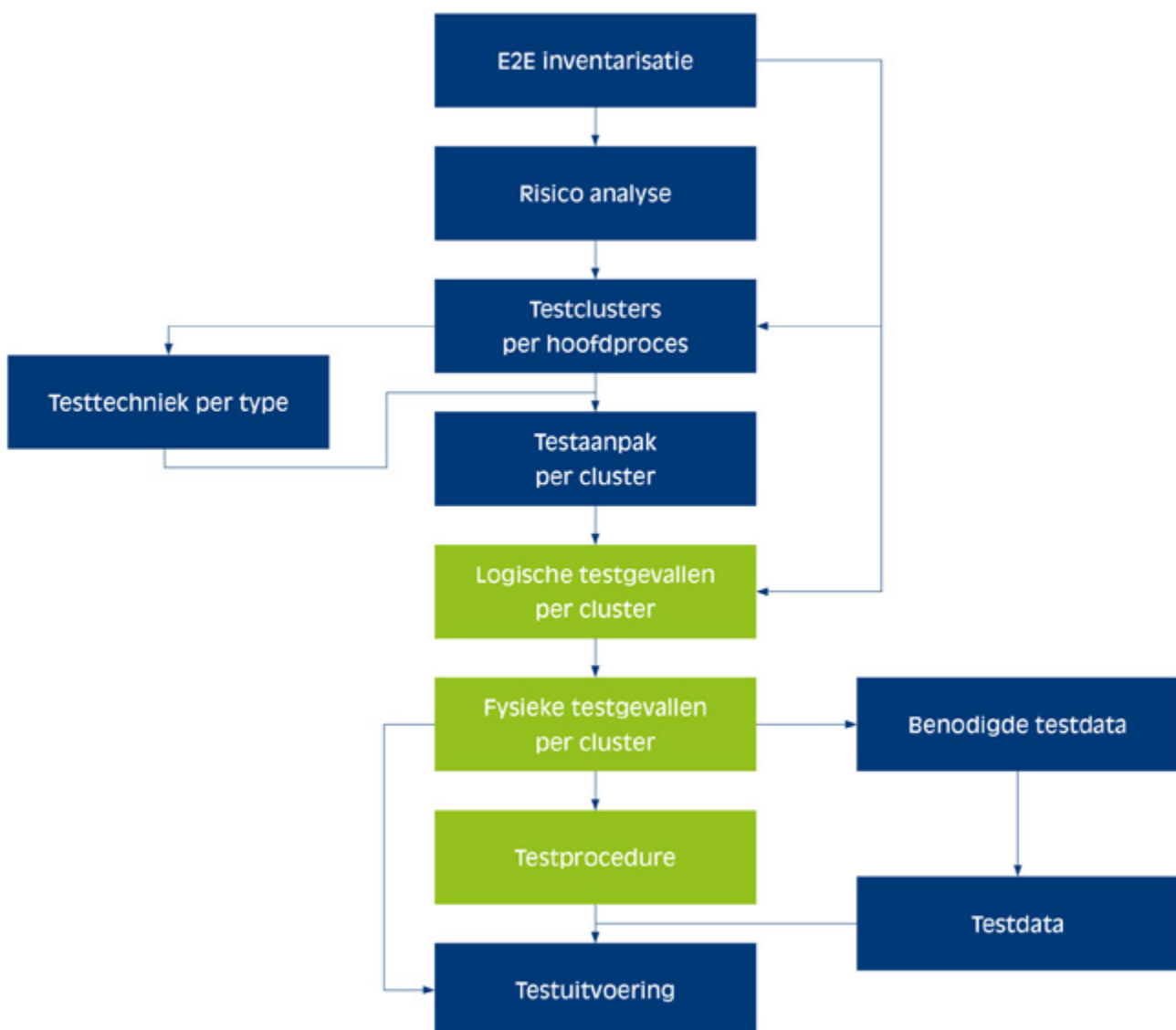
# Deel 9

## E2E-testontwerp

## Deel 9 E2E-testontwerp

In dit hoofdstuk zijn de onderwerpen: het afleiden van logische testgevallen, het uitwerken van fysieke testgevallen en het opstellen van een testprocedure. Daarna worden drie testtypes in meer detail beschreven. Deze testtypes zijn de meest voorkomende niet-functionele testen die tijdens een E2E-test worden uitgevoerd: Betrouwbaarheidstesten, Efficiencytesten (of: Performancetesten) en Securitytesten.

De resultaten uit de E2E-inventarisatie en de E2E-teststrategie vormen de basis voor het ontwerp van de testgevallen: achtereenvolgens de E2E-inventarisatie, de opdeling in testclusters en de testaanpak per testcluster.



*Figuur: Testontwerp in samenhang met andere activiteiten*

In de E2E-inventarisatie worden de E2E-processen en E2E-ketens uitgewerkt. In de E2E-teststrategie en de E2E-testplanning zijn de testclusters vastgesteld met bijbehorende aanpak. Het E2E-testontwerp pakt hier de draad op.

Per testcluster worden logische testgevallen en vervolgens fysieke testgevallen afgeleid. Logische testgevallen volgen uit de combinaties van actoren, beslissende factoren, kritische factoren en resultaten. In fysieke testgevallen worden de handelingen en randvoorwaarden (testdata, systeemtoestand) zo concreet mogelijk beschreven. Daarna wordt een testprocedure opgesteld. Hierin wordt vastgelegd in welke volgorde testgevallen moeten worden uitgevoerd, op basis van de onderlinge afhankelijkheid tussen testgevallen en hun prioriteit. Daarnaast staat in de testprocedure wat de randvoorwaarden zijn voor het uitvoeren van groepen testgevallen, zoals een bepaalde systeemdatum, het aanmaken van testdata of het draaien van achtergrondprogramma's.

## 9.1 Testontwerp: logische testgevallen

### **Testcondities: Actoren, beslissende factoren, kritische factoren, resultaten**

Als onderdeel van de E2E-inventarisatie is per E2E-proces een overzicht van de actoren, van de kritische en beslissende factoren en de resultaten gemaakt. De actoren voeren de (niet geautomatiseerde) handelingen uit in de E2E-keten. Dit zijn in de regel mensen (klanten, gebruikers, beheerders), maar kunnen ook andere systemen of organisaties zijn. In dat laatste geval wordt de interne geautomatiseerde werking van zo'n systeem of organisatie buiten beschouwing gelaten.

Een proces begint altijd met een actor. Resultaten zijn (meestal) ook bedoeld voor een actor. De beslissende factoren zijn de functionele en systeemstappen in de E2E-keten (de processtappen en hoe deze in de systemen worden uitgevoerd). De kritische factoren zijn zaken die van invloed kunnen zijn op een E2E-proces (zoals maandovertgangen, klantsituatie zoals overlijden of betaalachterstanden).

Op de volgende pagina treft u een praktijkvoorbeeld aan van zo'n overzicht. Het gaat hier om het proces 'aanmaken polis' voor een verzekeringsmaatschappij.

**Tabel: Lijst Actoren, factoren, resultaten**

Actoren		Functie
	Klant	Ontvangt mail
	Gebruiker polis/klantsysteem	Controle/mutaties/invoer
	Gebruiker financiën	Financiële verwerking
	Beheer	Draaien achtergrondprogramma's; versturen
	BKR	Terugkoppelen kredietstatus
	Bank	Verwerken betaling
	Postorganisatie	Verwerken communicatie
Beslissende factoren (proces)		
	Systeem	Functie
1	Web	Zoeken producten
3	Productstelsysteem	Leveren productgegevens
4	Web	Selecteren product
5	Web	Ingeven klantgegevens
6	Web	OK
7	Polisadministratie	Controleren klantgegevens
9	Klantsysteem	Controleren klantgegevens
10	Klantsysteem	Invoeren klantgegevens
11	Klantsysteem	Wijzigen klantgegevens
12	Klantsysteem	BKR check
13	BKR	Leveren BKR-status
14	Klantsysteem	Leveren klantgegevens
15	Polisadministratie	Verwerken klantgegevens
16	Polisadministratie	Verzenden melding NOK
17	Web	Tonen melding NOK
18	Polisadministratie	Ophalen productgegevens
19	Productstelsysteem	Leveren productgegevens
20	Polissysteem	Aanmaken polis
21	Polissysteem	Verzenden contract
22	Poststelsysteem	Verwerken contractberichten
23	Poststelsysteem	Aanmaken post
24	Polissysteem	Aanmaken betalingsgegevens
25	Financieel systeem	Aanmaken betalingen
26	Bank	Verwerken betalingen
27	Bank	Verwerken retourbetalingen
28	Financieel systeem	Verwerken berichten bank
29	Polissysteem	Aanmaken werklíjst: foutieve betalingen
Resultaten		Systeem
1	Contract verstuurd	Post
2	Betaling	Bank
3	Contract vastgelegd	Polissysteem
4	Klant aangemaakt	Klantsysteem
5	Klant gewijzigd	Klantsysteem
6	Melding	Gegevens niet correct
7	Melding	Kredietwaardigheid
8	Melding	Product niet geldig in opgegeven periode
Kritische factoren		Omschrijving
1	Maandovergang	
2	>Start productperiode	Ingangsdatum vlak voor/ na ingangsdatum product
3	< Einde productperiode	Ingangsdatum vlak voor/ na einddatum product

De hierboven genoemde factoren hebben de functie van testcondities. In het logisch testontwerp moet de dekking van deze factoren tot uitdrukking worden gebracht. Bovendien moeten hierin keuzes worden gemaakt voor wat betreft de combinaties van factoren die per testgeval worden gedekt.

### Testontwerp en testdekking

Per E2E-proces worden testgevallen ontworpen.

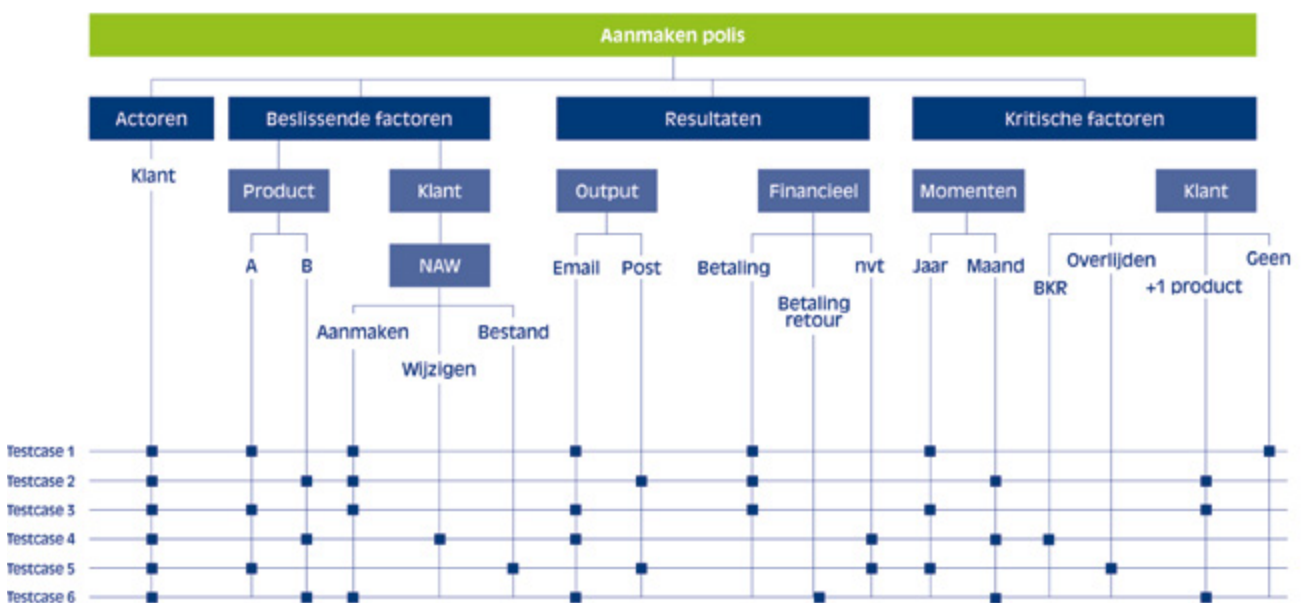
Dit begint met een standaard scenario, de happyflow. Dit is het meest voorkomende rechttoe rechtaan geval. Dit standaard testgeval is de basis voor testgevallen die er op volgen en moet goed worden afgestemd met ervaren gebruikers en beheerders die er in productie ook verantwoordelijk voor zijn.

Op basis van de happyflow worden vervolgens varianten ontworpen. Doel hiervan is om [a] alle actoren, beslissende factoren, kritische factoren en resultaten tenminste één keer te raken. [b] Daarnaast om de meest voorkomende varianten uit de praktijk te raken (dit verdient wederom afstemming met de ervaren gebruikers) en tenslotte [c] om specifieke combinaties te maken van testcondities waarvoor redenen zijn om deze te testen. Het ontwerpen van een E2E-test betekent realistische, representatieve scenario's uitwerken en niet: blind een techniek toepassen op een verzameling testcondities. Elk nieuw testgeval moet een toegevoegde waarde hebben omdat de testgevallen arbeidsintensief zijn qua uitwerking en uitvoering. Een E2E-test richt zich op de samenhang tussen systemen en processen. Hiervoor hoeft de interne werking van systemen, functies en interfaces niet expliciet getest te worden, dit is namelijk de taak van een systeemtest.

### Datacombinatietest

De meest geschikte ontwerpstechniek om een logisch testontwerp te maken voor een E2E-test is de Datacombinatietest (of: Classification Tree Method). Hiermee kan worden afgezekerd dat alle testcondities zijn geraakt en kan men laten zien hoe vaak en waar testcondities worden geraakt.

Hieronder volgt een voorbeeld van een uitgewerkte Datacombinatietest. Binnen deze techniek kan gevarieerd worden met dekkingsvormen, bijvoorbeeld indien men bepaalde combinaties van kritische factoren en beslissende factoren wil zien. In het voorbeeld zijn de klassen omwille van de leesbaarheid vereenvoudigd.



Figuur: Voorbeeld DCT: polis aanmaken

Een voordeel van een DCT-opzet is dat deze gemakkelijk aan andere partijen is uit te leggen. De testgevallen en de door hen gedekte elementen worden in één overzicht weer gegeven. Nieuwe elementen (bijvoorbeeld een extra kritische factor zoals een bepaalde mutatie) kunnen zonder problemen worden ingevoegd. Door een kopie te maken van het meest simpele basisscenario, met daarin de wijzigingen om het nieuwe element te testen, is dan ook het nieuwe element afgedekt.

Een DCT wordt in de boeken, en ook in bovenstaand voorbeeld, horizontaal uitgewerkt voor wat betreft de klassen. In geval van een ingewikkelde E2E-test, met veel klassen, kan het handiger zijn dit verticaal uit te werken, bijvoorbeeld in Excel. Er kan dan eventueel per punt aangegeven worden wat de vulling moet zijn, waardoor er tegelijk een fysieke uitwerking van het testgeval kan plaatsvinden. Bovendien kunnen testresultaten hier ook in worden bijgevoegd, waardoor er een goede basis is voor het monitoren en rapporteren van de testvoortgang.

Figuur: Uitwerking logische testgevallen DCT verticaal

			aantal malen geraakt	Testgeval 1: happyflow	Testgeval 2: klantgegevens wijzigen	Testgeval 3: klantgegevens NOK	Testgeval 4: retourbetaling	Testgeval 5: BKR NOK	Testgeval 6: product niet meer geldig	Testgeval 7: product nog niet geldig	Testgeval 8: klant overlijdt vlak na afsluiten product	Bevindingen/ status
<b>Actoren</b>	<b>Rol</b>											
1	Klant	vraagt aan, ontvangt mail, betaalt	8	x	x	x	x	x	x	x	x	
2	Gebruiker Polis/ Klantsysteem	Controle/mutaties/invoer	8	x	x	x	x	x	x	x	x	
3	Gebruiker Financiën	Financiële verwerking	3	x			x					x
4	Beheer	Draaien achtergrondprogramma's	5	x	x	x	x					x
5	BKR	Terugkoppelen kredietstatus	4	x			x	x				x
6	Bank	Verwerken betaling	3	x			x					x
7	Postorganisatie	Verwerken communicatie	5	x	x		x	x				x
<b>Beslissende factoren (proces)</b>			<b>0</b>									
	<b>Systeem</b>	<b>Functie</b>										
1	Web	zoeken producten	8	x	x	x	x	x	x	x	x	
2	Productstelsysteem	leveren productgegevens	8	x	x	x	x	x	x	x	x	
3	Web	selecteren product	8	x	x	x	x	x	x	x	x	
4	Polisadministratie	controleren klantgegevens	8	x	x	x	x	x	x	x	x	
5	Klantsysteem	invoeren klantgegevens	7	x		x	x	x	x	x	x	
6	Klantsysteem	BKR check	8	x	x	x	x	x	x	x	x	
7	BKR	leveren BKR-status	8	x	x	x	x	x	x	x	x	
8	Polisadministratie	ophalen productgegevens	6	x	x		x		x	x	x	
9	Polissysteem	aanmaken polis	4	x	x		x					x
10	Polissysteem	verzenden contract	4	x	x		x					x
11	Poststelsysteem	aanmaken post	4	x	x		x					x
12	Financieel systeem	aanmaken betalingen	4	x	x		x					x
13	Bank	verwerken betalingen	4	x	x		x					x
14	Polissysteem	Aanmaken werklijst: foutieve betalingen	1				x					
<b>Resultaten</b>			<b>0</b>									
	<b>Systemen</b>	<b>Systemen</b>										
1	Contract verstuurd	post	4	x	x		x					x
2	Betaling	bank	3	x	x							x
3	Contract vastgelegd	polissysteem	4	x	x		x					x
<b>Kritische factoren</b>			<b>0</b>									
	<b>Omschrijving</b>											
1	Maandovergang		1		X							
2	Overlijden		1									x bev 312
<b>Testresultaten</b>			<b>OK</b>	<b>nok</b>								
1	testronde 1		0	2	nok	nok						
2	testronde 2		2	1	ok	nok	ok					
3	testronde 3		1	0		ok						
4	testronde 4		2	2			ok	nok	ok	nok		
5	testronde 5		3	0	ok			ok			ok	



In het hierboven gegeven voorbeeld wordt een minimale dekking conform de Datacombinatietest behaald: elk element wordt ten minste één keer geraakt.

## 9.2 Testontwerp: fysieke testgevallen

De voor een fysiek testgeval benodigde informatie, bijvoorbeeld met betrekking tot de uit te voeren handelingen, is af te leiden uit werkinstructies, procesbeschrijvingen, beheerprocedures, ontwerpdocumenten, testgevallen van andere testteams of door gebruikers te interviewen.

De mate waarin een testgeval fysiek wordt uitgeschreven is afhankelijk van:

- De behoefte van de testuitvoerder aan concrete richtlijnen.
- De mate waarin kan worden voorspeld hoe het testgeval exact moet worden uitgevoerd. Vooral bij nieuwbouw, nieuwe E2E-processen of nog grote onduidelijkheden bij het testteam over de exacte werkprocedures, is het raadzaam te wachten met het uitschrijven van fysieke testgevallen.
- Het betreffende risico dat wordt gedekt (een hoog risico zal waarschijnlijk in meer detail moeten worden uitgeschreven).
- Externe redenen om een testgeval fysiek uit te schrijven, zoals audits, QA-standaarden die worden gehanteerd of wet- en regelgeving waarin wordt afgedwongen dat er een hoge mate van traceerbaarheid is van kwaliteitscontroles.

Afhankelijk van deze overwegingen moeten in het fysieke E2E-testgeval de volgende zaken worden beschreven:

- Startsituatie: toestanden van het systeem, te gebruiken gegevens, instellingen, systeemdatum, aanwezigheid van uitgangsbestanden, interfaces (voor zover dat niet vanzelfsprekend is in de testomgeving) en status van andere testgevallen die vooraf gaan aan dit testgeval.
- Acties per stap in het E2E-proces: dit kunnen handmatige acties of bijvoorbeeld te draaien programma's zijn of bestanden die moeten worden klaargezet.
- Controles per stap in het E2E-proces: hierin moet worden benoemd wat per stap het te verwachten tussenresultaat is.
- Verwacht resultaat (eindresultaat).
- Eindsituatie: toestand van het systeem na het uitvoeren van het testgeval, te starten vervolgstapen.
- Voor de acties en de controles moet worden bepaald wie dit moeten doen. In principe moeten acties en controles worden uitgevoerd door medewerkers die dit in productie ook doen. Op deze manier wordt de proceskennis het best vertegenwoordigd in de testen, kunnen bevindingen het best worden beoordeeld en kunnen acceptatieaspecten worden meegenomen in de E2E-test. Een en ander is natuurlijk afhankelijk van het belang van de actie of de controle, de inzichtelijkheid en voorspelbaarheid van de acties en controles en risico's. Het is in ieder geval van belang spoedig inzichtelijk te krijgen welke partijen worden geacht bij te dragen aan de testuitvoering. De organisatie van de testen kan dan worden opgezet. Met name in grote complexe ketens met diverse organisaties kan het organiseren van de testuitvoering een uitdaging zijn (voor de testmanager).
- Er wordt in het testgeval ook aangegeven of er bewijs van de controles moet worden opgeslagen en hoe dat moet gebeuren. Men kan daarin zo ver gaan dat alle tussenresultaten (zoals bestanden of databases) fysiek moeten worden bewaard of dat bewijzen van resultaten moeten worden gearhiveerd (zoals printscreens).

Een fysiek testgeval bestaat uit een startsituatie, controles, tussenresultaten en eindresultaten. Deze zijn verdeeld over diverse systemen en vaak over meerdere dagen. Helderheid, overzichtelijkheid, traceerbaarheid en onderhoudbaarheid zijn van belang. Het verdient daarom aanbeveling om logische testgevallen en de daaraan gekoppelde fysieke testgevallen in meerdere tabbladen, in één spreadsheet, op te slaan. Generieke teststappen (zoals het invoeren of annuleren van een polis of het draaien van een achtergrondprogramma) kunnen dan op één plaats of tabblad worden vastgelegd.

In onderstaande tabel wordt een voorbeeld gegeven van een fysieke uitwerking van logische testgevallen. Het detailniveau in dit voorbeeld is vrij laag, maar kan in de praktijk natuurlijk verschillen.

In dit praktijkvoorbeeld zijn de dagelijks geplande achtergrondprogramma's van belang. Daarom zijn er per dag acties en controles en is er sprake van het draaien van de achtergrondprogramma's in de diverse systemen tussen de dagen in. In dit geval is dit niet in detail uitgeschreven. Indien procedures rond het draaien van achtergrondprogramma's niet duidelijk of vanzelfsprekend zijn moet dit wel worden uitgewerkt.

Per dag worden regels per systeem aangegeven. Op deze manier kunnen de betreffende medewerkers uit de testgevallen aflezen wat ze per dag moeten doen en kunnen ze ook dienovereenkomstig worden aangestuurd.

Met kleuren, toegevoegde tekst of opmerkingen kunnen de testresultaten worden gelogd in de testgevallen.

**Tabel: Voorbeeld fysieke testgevallen**

	Testgeval 1: happyflow	Testgeval 4: retourbetaling
<b>Testdata:</b>		
<b>Naam</b>	F. Le Boulanger	G. Beenhouwer
<b>Sofinummer</b>	122344544	654654654
<b>Comm. Wijze verz nemer</b>	Post	mail
<b>Ingangsdatum:</b>	1-1-2014	1-1-2014
<b>Prolongatiedatum:</b>	1-1-2015	1-1-2015
<b>Dekking</b>	Volledig	Middel
<b>Betaalmethode</b>	Incasso	Incasso
<b>iban</b>	023ABN022448679708	023ABN022448644535434
<b>speciaal/randvoorwaarden</b>	Klant bestaat nog niet, geen BKR registratie	Klant bestaat nog niet, geen BKR registratie
<b>Testuitvoering</b>		
<b>Acties dag 1</b>	21-6-2014	
Klant/ web portaal	Klant voert zichzelf als nieuwe klant op; nieuwe polis	Klant voert zichzelf als nieuwe klant op; nieuwe polis
Polissysteem		
Klant systeem		
BKR		
Bank		
Post		
<b>Controles dag 1</b>	21-6-2014	
Klant/ web portaal		
Polissysteem	Polis met status "voorlopig"	Polis met status "voorlopig"
Klant systeem	Klant geregistreerd	Klant geregistreerd
Financieel systeem		
BKR		
Bank		
Post		
<b>Batchproces 1&gt;2</b>	<b>Beheer voert per systeem dagelijkse programma's uit</b>	<b>Beheer voert per systeem dagelijkse programma's uit</b>
<b>Controles dag 2</b>	22-6-2014	
Klant/ web portaal		
Polissysteem	Polis status actief	Polis status actief
Klant systeem		
Financieel systeem	Grootboek post x aangeveleverd	Grootboek post x aangeveleverd
BKR	BKR check uitgevoerd en retour gestuurd	BKR check uitgevoerd en retour gestuurd

Bank		
Post	Polisblad en factuur naar klant gestuurd	Polisblad en factuur naar klant gestuurd
<b>Acties dag 3</b>		
Klant/ web portaal		
Polissysteem		
Klant systeem		
Financieel systeem		
BKR		
Bank		
Post		
<b>Batchproces 2&gt;3</b>	<b>Beheer voert per systeem dagelijkse programma's uit</b>	<b>Beheer voert per systeem dagelijkse programma's uit</b>
<b>Controles dag 3</b>	23-6-2014	
Klant/ web portaal	actieve polis	actieve polis
Polissysteem	Status betaling: verstuurd	Status betaling: verstuurd
Klant systeem		
Financieel systeem	status betaling: verstuurd	status betaling: verstuurd
BKR		
Bank	Incassoopdracht verstuurd naar bank	Incassoopdracht verstuurd naar bank
Post		
<b>Acties dag 4</b>	23-6-2014	
Klant/ web portaal	Klant logt in: ziet actieve polis	Klant logt in: ziet actieve polis
Polissysteem		
Klant systeem		
Financieel systeem		
BKR		
Bank	verwerk incasso (retourbericht versturen): OK	verwerk incasso: retourbericht met status NOK, geen saldo
Post		
<b>Batchproces 3&gt;4</b>	<b>Beheer voert per systeem dagelijkse programma's uit</b>	<b>Beheer voert per systeem dagelijkse programma's uit</b>
<b>Controles dag 4</b>	24-6-2014	
Klant/ web portaal		
Polissysteem	Status betaling: voldaan	Status betaling: open
Klant systeem		
Financieel systeem	Status betaling: voldaan	Status betaling: open
BKR		
Bank		
Post		brief met betalingsverzoek en acceptgiro naar klant
<b>Acties dag 5</b>	24-6-2014	
Klant/ web portaal		
Polissysteem	Polis handmatig prolongeren	
Klant systeem		
Financieel systeem		
BKR		

Bank		retourbericht met betaald betalings- kenmerk acc giro (klant betaalt acceptgiro)
Post		
<b>Batchproces 4&gt;5</b>	<b>Beheer voert per systeem dagelijkse programma's uit</b>	<b>Beheer voert per systeem dagelijkse programma's uit</b>
<b>Controles dag 5</b>	25-6-2014	
Klant/ web portaal		
Polissysteem	Nieuwe dekkingsperiode aangemaakt	Status betaling: voldaan
Klant systeem		
Financieel systeem	Grootboekpost voor nieuwe periode aangeleverd	Status betaling: voldaan
BKR		
Bank	Incasso naar bank voor nieuwe premie	
Post	Nieuwe polis en factuur voor nieuwe periode	
<b>EINDE TESTGEVAL</b>		<b>EINDE TESTGEVAL</b>

### 9.3 Opstellen testprocedure

De testprocedure (ook wel testscript genoemd) bestaat uit samenhangende fysieke testgevallen die tot een gecombineerde verzameling handelingen en activiteiten worden gebundeld. Hierdoor kan de testuitvoering zo efficiënt en simpel mogelijk verlopen. Een testprocedure kan de samenvoeging zijn van de fysieke testgevallen tot één document, of een testprocedure is een apart document waarin een overzicht wordt gegeven van de onderlinge afhankelijkheden tussen de testgevallen.

In het testgeval moet worden aangegeven wat de eventuele afhankelijkheden zijn met andere testgevallen. Als de eindsituatie van het ene testgeval bijvoorbeeld de startsituatie van het andere testgeval is, of wanneer door combinaties van testgevallen andere testgevallen worden uitgevoerd. Logische testgevallen en fysieke testgevallen kunnen soms worden gebundeld en testgevallen uit het ene testcluster kunnen soms worden gebruikt binnen het andere testcluster.

**Tabel: overzicht afhankelijkheden testgevallen**

	Testgeval Offerte 1	Testgeval Offerte 2	Testgeval Offerte 3	Testgeval Offerte 4	Testgeval Aanmaken 1	Testgeval Aanmaken 2	Testgeval Aanmaken 3	Testgeval Aanmaken 4	Testgeval Mutenen 1	Testgeval muteren 2	Testgeval muteren 3	Testgeval muteren 4	Testgeval Beëindigen 1	Testgeval Beëindigen 2	Testgeval Beëindigen 3	Testgeval Beëindigen 4
Id	TGO1	TGO2	TGO3	TGO4	TGA1	TGA2	TGA3	TGA4	TGM1	TGM2	TGM3	TGM4	TGB1	TGB2	TGB3	TGB4
In					TGO1	TGO2	TGO3		TGA1	TGA2	TGA3	TGO4	TGA4	TGM1	DATA	DATA
Uit	TGA1	TGA2	TGA3	TGM4	TGM1	TGM2	TGM3	TGB1	TGB2							

Uit een tabel zoals hierboven, kan worden afgelezen of een testgeval start met de output van een ander testgeval en of een testgeval zelf de input voor weer een ander testgeval levert. Testgeval Aanmaken 3 (TGA3) kan starten met de eindsituatie van TGO3 en zijn eigen eindsituatie kan dienen als startsituatie voor TGM3.

Het hierboven gegeven voorbeeld is een vorm waarin een testprocedure kan worden opgesteld. Eventueel kunnen fysieke testgevallen zelf worden samengevoegd. De testprocedure wordt aangevuld met momenten waarop handelingen en controles worden uitgevoerd, namen van degenen die ze moeten uitvoeren en dergelijke. De testprocedure groeit dan uit tot een detailtestdraaiboek.

**Tabel: Testprocedure**

Tijd	Activiteit	Wie	Testgeval	Testgeval	Testgeval	Testgeval	Testgeval	Testgeval	Testgeval	Testgeval
			Offerte 1	Offerte 2	Offerte 3	Offerte 4	Aanmaken 1	Aanmaken 2	Aanmaken 3	Aanmaken 4
		Id	TGO1	TGO2	TGO3	TGO4	TGA1	TGA2	TGA3	TGA4
		Start					TGO1	TGO2	TGO3	
		Einde	TGA1	TGA2	TGA3	TGM4	TGM1	TGM2	TGM3	TGB1
9:00	Laden database A op systeem A	TB								
9:30	Draaien dagwerk A systeem A	FB								
10:00	Controle resultaat dagwerk	Test								
10:30	Testen	Test	Stap1	Stap1	Stap1	Stap1				Stap1
11:00	Draaien dagwerken	FB								
11:30	Controle resultaat dagwerk	Test	Contr1	Contr1	Contr1	Contr1				Contr1
12:00	Testen	Test					Stap1	Stap1	Stap1	Stap2
12:30	Draaien dagwerken	FB								
13:00	Controle resultaat dagwerk	Test					Contr1	Contr1	Contr1	Contr2
13:30	Testen	Test								Stap3
14:00	Draaien dagwerken	FB								
14:30	Controle resultaat dagwerk	Test					Contr1			Contr3
15:00	Testen	Test					Stap2			Stap3
15:30	Draaien dagwerken	FB								
16:00	Controle resultaat dagwerk	Test					Contr2			Contr3
16:30	Testen	Test					Stap3			Stap3

## 9.4 Speciale testtypen

In deze paragraaf worden de testtypen Betrouwbaarheid, Efficiency en Securitytest nader beschouwd. Deze testtypen representeren kwaliteitsattributen die steeds belangrijker worden en waarbij de E2E-test vaak de meest geschikte testsoort is om deze te testen. Met name de efficiencytest en de securitytest zijn inmiddels uitgegroeid tot specialismen waarbij de testvoorbereiding en uitvoering aan experts moet worden overgelaten in geval van hoge risico's.

Per testtype wordt vooral ingegaan op de samenhang met E2E-testen en E2E-processen.

### 9.4.1 Betrouwbaarheidstest

Indien in een productrisicoanalyse betrouwbaarheid als een hoog risico wordt ingeschat, moet een betrouwbaarheidstest worden opgezet. Afhankelijk van waar het risico wordt gelokaliseerd (en of dit systeemoverstijgend is) is de E2E-test een serieuze kandidaat om dit risico af te dekken. Indien betrouwbaarheid voornamelijk zit in de verwerking binnen één systeem of indien belangrijke constatering binnen één systeem kunnen worden gedaan, kunnen de testen ook bij de betreffende systeemtest worden belegd. Voor betrouwbaarheidstesten zijn nog geen standaarden en uitgewerkte technieken beschikbaar zoals die wel beschikbaar zijn voor efficiency- en securitytesten. Om deze reden worden betrouwbaarheidstesten hier nader uitgewerkt.

Betrouwbaarheid omvat een aantal onderling verschillende aspecten, die elk ook weer verschillend getest moeten worden:

- **Foutbestendigheid:** de mate waarin E2E-processen kunnen worden uitgevoerd in geval van fouten of foutieve invoer. Het kan hier gaan om al dan niet bekende fouten in programmatuur waarvan men de gevolgen verderop in de keten niet goed kan inschatten.
- **Storingsbestendigheid** (wordt ook wel "failover" genoemd): hoe gaat de gehele keten om met het verwerken van deelprocessen of transacties indien tijdens de verwerking "ergens" een storing optreedt? Wat is de impact op het E2E-proces?
- **Herstelbaarheid:** het gaat hierbij om de vraag hoe moeilijk het is om na een storing weer beschikbaar te krijgen.
- **Beschikbaarheid:** Wat kan de beschikbaarheid van een dienst of product beïnvloeden? Wat werkt er in de gehele keten niet meer als één applicatie of één deelproces niet meer werkt?

### Stap 1: vaststellen aspect en oorzaak-gevolganalyse

In geval van een hoog risico op het gebied van betrouwbaarheid wordt eerst vastgesteld om welke van de bovenstaande aspecten het gaat en in welke systeemdelen het risico zich voordoet voor wat betreft oorzaak en gevolg. Als het bijvoorbeeld gaat om failover of storingsbestendigheid moeten testers weten welke specifieke storingen, in welke situatie of op welk moment zouden kunnen leiden tot welke gevolgen verderop in de keten. Deze informatie kan worden verkregen door met betrokken systeemanalisten en beheerders het risico nader door te spreken. Voor deze analyse is de E2E-inventarisatie uitgangspunt.

### Stap 2: logische testgevallen

Het opstellen van de logische testgevallen geschiedt op basis van de oorzaak-gevolganalyse. Dit wordt in een E2E-plaat uitgetekend. Elk gevolg en elke oorzaak die moet worden ontketend of opgeroepen levert een logisch testgeval op. Er wordt vastgesteld welke variatie, load en overige omstandigheden (bijvoorbeeld gedeeltelijke uitval) moeten worden getest.

Eventueel wordt aangesloten bij andere testen. Voorbeelden hiervan zijn: een combinatie met securitytesten in geval van het breken van een website na een storing; een stresstest die een storing veroorzaakt; het testen van de functionele beschikbaarheid als controlemiddel na een storing; het gemeenschappelijk gebruiken van één testomgeving en de last die verschillende testclusters van elkaar kunnen hebben.

### Stap 3: fysieke testgevallen

Het uitwerken van fysieke testgevallen vereist kennis van de technische werking en de beheerprocedures van de systemen. Want hoe breng je een specifiek technisch proces op het juiste moment down om storingsbestendigheid te testen? In de praktijk betekent dit dat testers nauw moeten samenwerken met technisch beheerders.

Het fysieke testgeval zou er dan als volgt uit kunnen zien. We nemen hier als voorbeeld storingsbestendigheid en de E2E-inventarisatie uit de vorige paragrafen:

1. Voer polis in als klant in systeem A.
2. Verstuur aanvraag via interface X.
3. Tijdens verwerken van aanvraag: breng server Y down (actie: technische beheer).
4. Controleer of bericht niet al was ontvangen in systeem B.
5. Herstart server.
6. Controleer de status van de polis in systeem A en B.

Het controleren of de storing op het juiste moment is geslaagd geschiedt in controle 4: indien het bericht wel al was verwerkt zal het testgeval weer opnieuw moeten worden uitgevoerd omdat men de storing wil uitvoeren op het moment dat de aanvraag nog niet geheel afgehandeld was.

#### 9.4.2 Efficiencytest (performancetest)

Voor wat betreft de exacte uitvoering van efficiencytesten wordt hier verwezen naar de testliteratuur. Het testen van performance of efficiency is namelijk een vak apart. We beperken ons hier tot een samenvatting en de rol die het E2E-testteam speelt in het testen van efficiency.

### Vaststellen van de benodigde testen

Uit de risicoanalyse kan efficiency naar voren komen als een relevant aspect. Er moet dan worden vastgesteld welk type efficiency wordt bedoeld.

Grofweg zijn er 3 typen efficiency:

1. **Load** (gewenste verwerkingstijd bij een bepaalde verwachte belasting).
2. **Stress** (gedrag van het systeem bij een maximale belasting of na een maximale duur en volume).
3. **Respons** (de snelheid waarmee processen kunnen worden afgehandeld en leiden tot een resultaat).

Operational profiling is een methode om te analyseren met welke processen en handelingen welke belasting van het systeem kan worden verwacht. In deze methode wordt het gebruik van de systemen en de systeemonderdelen voorspeld, op basis van gebruikersprofielen en monitoring van activiteiten op productieomgevingen. Operational profiles kunnen zelf ook weer een goede input zijn voor een risicoanalyse ten behoeve van een functionele E2E-test. Immers: een veel gebruikt deel van de E2E-keten wijst op hoge risico's en de noodzaak van hogere testdekking, ook voor een functionele E2E-test.

Uit de risicoanalyse moet naar voren komen waar een efficiencyrisico moet worden getest: kan worden volstaan met de test van een lokaal proces in één of meer systemen of moet voor het testen van efficiency een volledige E2E-keten worden opgetuigd? Kunnen deelprocessen elk een goede efficiency hebben, maar kan dan toch het E2E-proces een te lage efficiency hebben? In deze gevallen zal kennis van het E2E-testteam omtrent de E2E-testomgeving en bijbehorende kennis van de infrastructuur en de E2E-procesen een rol kunnen spelen.

### Hergebruik

De kennis, vaardigheden, testomgeving, testdata en testgevallen van een E2E-test kunnen van nut zijn voor een efficiencytest. Dit geldt ook andersom. Voor efficiencytesten worden vaak tools ingezet die data of activiteiten genereren. Deze kunnen dikwijls ook ingezet worden om testdata te genereren voor de E2E-test of delen van de testuitvoering van de E2E-test te automatiseren.

Voor hergebruik van E2E-testgevallen in de efficiencytest is dan wel een gedetailleerd niveau vereist waarop testgevallen zijn uitgeschreven (fysieke testgevallen).

### 9.4.3 Securitytest

Securitytesten is, net als efficiencytesten, een vak apart.

Security kan samenhangen met de E2E-keten:

- Kwetsbaarheid voor penetratie moet geanalyseerd worden door de hele keten heen. Toegang tot de E2E-keten bestaat niet alleen uit het inlogscherf bij de web-frontend. Een systeem kan ongewenst toegang tot een ander systeem bieden.
- Gevolgen van securityproblemen moeten ook worden doordacht door de hele keten heen. Wat een klein integriteitprobleem lijkt aan de voorkant van een systeemketen, kan aan de achterkant een probleem worden, bijvoorbeeld wanneer privacygevoelig materiaal kan worden verstuurd of geprint.
- Securitymaatregelen kunnen de werking van de E2E-keten nadelig beïnvloeden. Overbeveiliging kan leiden tot gestremde E2E-processen. Een voorbeeld hiervan is een useraccount dat wordt gebruikt door een achtergrondproces. Zodra de rechten van groepen users worden beperkt, kunnen er in geautomatiseerde deelprocessen blokkades optreden (bijvoorbeeld nachtbatches die opeens geen velden meer kunnen bijwerken) en kan ook een E2E-proces vastlopen.
- Dit laatste kan ook de andere kant op werken: indien vanuit de E2E-test rechten worden verruimd om de nachtbatch uit het voorbeeld weer zijn werk te laten doen, kunnen hier weer andere securityproblemen uit voortkomen.

De E2E-inventarisatie en de E2E-testgevallen kunnen worden gebruikt in de testuitvoering en de analyse van mogelijke security problemen.





# Deel 10

De  
E2E-testorganisatie

## Deel 10 De E2E-testorganisatie

In vergelijking met andere testsoorten, zoals de systeemtest en de systeemintegratietest, zal een E2E-test een meer complexe en flexibele samenstelling hebben. Daarnaast moet rekening gehouden worden met het feit dat een E2E-test project- en organisatieoverschrijdend kan zijn. Dit betekent dat planning, budgettering en acceptatieverantwoordelijkheid ook een project- en organisatieoverschrijdende dimensie krijgt: wie betaalt er, wie doet wat en wie besluit dat het goed genoeg is? Ten slotte moeten de eisen die worden gesteld aan de E2E-tester niet worden onderschat: niet iedereen is geschikt om een centrale rol in een E2E-test in te nemen.

In dit hoofdstuk worden deze aspecten onderzocht en keuzemogelijkheden aangedragen. Eerst worden de globale vorm en de minimale onderdelen van een E2E-testorganisatie vastgesteld. Daarnaast wordt per onderdeel ingezoomd en ten slotte worden de keuzemogelijkheden in de invulling van de E2E-testorganisatie aangegeven.

Niet elk organisatieonderdeel hoeft altijd in elke omstandigheid aanwezig te zijn. Met sterretjes wordt per paragraaf aangegeven in welke omstandigheid een organisatieonderdeel gewenst is:

- \* = altijd, ook kleine organisaties
- \*\* = middelgrote organisaties, middelgrote projecten
- \*\*\* = hoge risico's, complex releasemanagement, grote organisaties
- \*\*\*\* = specifieke omstandigheden, alleen in bepaalde context.

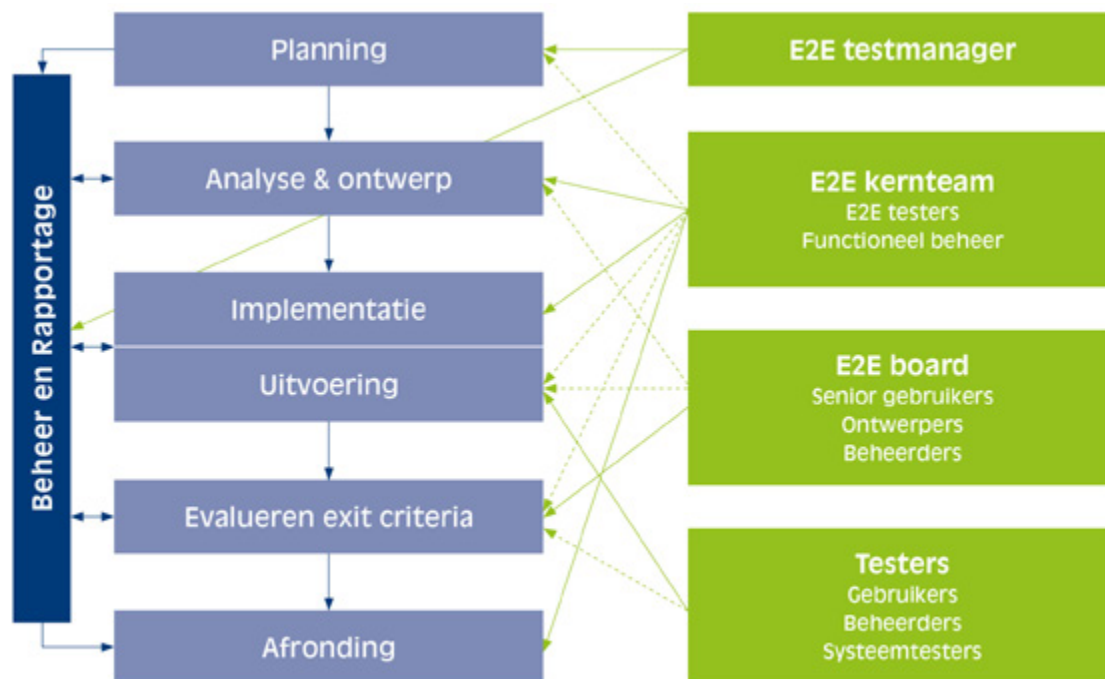
### 10.1 De onderdelen van de E2E-testorganisatie

Samenwerking tussen diverse expertises en het gebruik maken van diverse kennisbronnen staan centraal in de bepaling van de E2E-testorganisatie. Een E2E-test bestaat uit activiteiten (de bekende fasering van het testproces) die zich, meer dan een testsoort als de systeemtest, over een langere periode uitstrekken. De voorbereiding zal daarom vroeg in het project starten, met een kleine groep ervaren testers. De testuitvoering zal uiteindelijk plaatsvinden met een brede keur aan expertises vertegenwoordigd, zoals eindgebruikers, klanten, beheerders en systeemtesters.

De benodigde kennis voor een E2E-test is breder dan in andere testsoorten en varieert van E2E-proces en functionaliteit tot techniek. Testgevallen zijn niet volledig uit documenten af te leiden en behoeven samenwerking tussen procesexperts en experts op het gebied van de infrastructuur en de software. Testers hebben hier een bemiddelende en verzamelende rol.

De E2E-testorganisatie bestaat uit de volgende vier lagen:

1. **De E2E-testmanager.** Afhankelijk van de grootte en complexiteit van de E2E-test is dit de overall testmanager, een aparte E2E-testmanager of een coördinator. In grote organisaties en grote projecten kan zelfs gedacht worden aan meerdere testcoördinatoren (per organisatie of per E2E-proces).
2. **Het E2E-kernteam.** Dit wordt gevormd door één of meer E2E-testers en eventueel aangevuld met senior medewerkers of beheerders met een overzicht van E2E-processen en systemen. Het E2E-kernteam is vroeg in het traject betrokken en voert het grootste deel van de voorbereidingen uit (de E2E-inventarisatie en de logische testgevallen). Tijdens de testuitvoering vervullen de leden van het E2E-kernteam de rol van meewerkend voorman.
3. **De E2E-board.** Dit is een panel van deskundigen op het gebied van de E2E-processen en de infrastructuur. Dit panel komt regelmatig samen en dient als klankbord voor het E2E-kernteam als het gaat om risico's, testgevallen opstellen en foutanalyse. Leden van het board zijn senior procesdeskundigen, ontwerpers en functioneel beheerders.
4. **Testers.** Deze groep voert de testgevallen uit en beoordeelt de resultaten onder coördinatie van het E2E-kernteam. Dit kunnen zowel "vrijgekomen" systeemtesters, gebruikers als functioneel beheerders zijn. Er moet in ieder geval vertegenwoordiging zijn van de belangrijkste gebruikersgroepen omdat tijdens de testuitvoering volgens daadwerkelijke werkprocedures wordt gewerkt.



Figuur: E2E-testorganisatie: onderdelen en activiteiten

In deze afbeelding wordt de betrokkenheid van de vier lagen weergegeven. Een vaste lijn representeert eindverantwoordelijkheid en een onderbroken lijn een ondersteunende rol. Zo is het E2E-kernteam eindverantwoordelijk voor analyse en ontwerp en worden ze daarbij ondersteund door de E2E-board.

#### 10.1.1 De E2E-testmanager \*\*

Voor het leiden van een E2E-test zijn de volgende vaardigheden benodigd:

- Op hoofdlijnen E2E-processen en systeemketens in samenhang begrijpen.
- Communiceren met diverse typen medewerkers, met verschillende expertises en achtergronden. Deze medewerkers moeten samen betrokken worden in de E2E-test.
- Overzicht houden over een grote hoeveelheid en diversiteit aan informatie en activiteiten.
- In geval van wijzigingen in scope, status of voortgang, de juiste beslissingen nemen en deze uitdragen zodat de doelen gehaald worden.

De E2E-test wordt in de regel geleid door één testmanager. Dit is een manager en geen coördinator omdat het meer omvat dan het coördineren van één testteam. Indien de omstandigheden dit toelaten kan het management van de E2E-test ook door de overall testmanager worden ingevuld. Waarschijnlijk is er dan wel de noodzaak van een E2E-testcoördinator of meewerkend voorman van de E2E-test.

#### 10.1.2 De E2E-tester (E2E-kernteam) \*

Het E2E-kernteam bestaat uit één of meer E2E-testers, eventueel aangevuld met senior medewerkers met kennis van zowel de E2E-processen als ook de samenhang met de systeemonderdelen. Dit kunnen gebruikers, functioneel beheerders of ontwerpers zijn. De E2E-testers zijn ervaren testers die de vaardigheid bezitten om de E2E-processen en ketens te overzien en te analyseren. Ook vervullen zij een centrale rol als testcoördinatoren of meewerkend voormannen tijdens de testuitvoering. Een E2E-test verlangt een aantal extra vaardigheden waardoor niet iedere systeemtester automatisch een geschikte E2E-tester is.

Een typische E2E-tester is bij uitstek een allrounder en geen specialist. Dit komt in de volgende lijst met vaardigheden tot uitdrukking:

- Structuur van de E2E-keten: in staat zijn met verschillende systemen en systeemonderdelen te werken, zoals interfaces, schermen, achtergrondprogramma's.
- Doel van de E2E-keten: snel begrip krijgen van E2E-processen en producten.
- Met verschillende type mensen kunnen werken, communiceren, deze begrijpen en overtuigen: gebruikers, beheerders, ontwerpers, programmeurs, projectleiders, managers.
- Ervaring met front- en backend systemen.
- E2E-aspecten vinden in een veelheid aan specificaties, E2E-processen en organisatieonderdelen.
- Overzicht houden in een complex van werkzaamheden en statussen van diverse onderdelen.
- Effectief kunnen communiceren (de vraag: "hoe gaat het" niet beantwoorden met een waslijst aan bevindingen en testgevallen, maar in heldere termen per vitaal onderdeel indicaties "goed", "minder goed" kunnen aangeven en onderbouwen).
- Database ervaring (in verband met vinden van testdata en controles van tussenresultaten).
- Logbestanden en interfacebestanden kunnen lezen.

Gezien het toenemende belang van E2E-testen en de toenemende complexiteit rond het uitvoeren van E2E-testen wordt dit een veelgevraagde expertise.

### 10.1.3 De E2E-board \*

Een belangrijk onderdeel van de E2E-testorganisatie is de "E2E-board": er wordt een panel gevormd van belangrijke vertegenwoordigers met specialistische kennis. Dit zijn senioren beheerders per belangrijk systeem, ervaren gebruikers, business analisten, ontwerpers, architecten, programmeurs. In een groot project kunnen er meerdere E2E-boards zijn, bijvoorbeeld per E2E-proces.

In de E2E-board moet kennis van de E2E-processen, van de systeemketen en de wijzigingen uit het project worden verenigd. In de E2E-board worden E2E-zaken besproken: risico's, testgevallen, E2E-processen, bevindingen, voortgang, testdekking, voldoen aan acceptatiecriteria en dergelijke.

De frequentie, de vorm en lengte waarin de E2E-board samen komt hangt af van de fase waarin het testtraject zich bevindt en de specifieke project- en organisatieomstandigheden. Tijdens de fasen Analyse, Ontwerp en Implementatie vinden de zittingen 1 à 2 maal per week plaats en tijdens de fase Uitvoering dagelijks.

In geval van een complex en geografisch verspreide organisatie kan er gedacht worden aan telefonische of video conferenties. Van belang is dat er periodiek wordt gediscussieerd over de belangrijkste issues in het E2E-testtraject.

De E2E-board heeft positieve bijeffecten:

- De leden van de E2E-board zijn centrale medewerkers met autoriteit in de ogen van belangrijke acceptanten en gebruikers. De leden brengen hun, vaak positieve, ervaringen over naar andere delen van de organisatie. Via de E2E-board vindt daarom al een belangrijke deel van de acceptatie vroeg plaats. Dit gaat ook om zogenaamde "emotionele acceptatie", waarbij angst voor het grote onbekende wordt weggenomen.
- Deelnemers aan de E2E-board, die aanvankelijk vaak weerstand hebben tegen de "nieuwe" werkzaamheden zoals deelnemen aan een test, raken meer en meer betrokken bij het testen. Ze herkennen tijdens de testactiviteiten problemen en zien wat de toegevoegde waarde is van gestructureerd testen.
- Door de brede samenstelling, de kennis van het panel en het concreet doorspreken van praktijkscenario's en testscenario's vindt er een vroege validatie plaats van het ontwerp en de samenhang met de E2E-processen.

### 10.1.4 Testers \*

Testvoorbereidingen worden zo veel mogelijk gedaan door een klein E2E-kernteam. Voor de testuitvoering, en deels het uitschrijven van fysieke testgevallen, is het voordelig gebruik te maken van een groter team van

medewerkers. Kandidaten om het E2E-testteam aan te vullen tijdens de testuitvoering zijn:

1. Gebruikers uit relevante afdelingen of E2E-processen.
2. Systeemtesters van de betreffende systemen.
3. Functioneel beheerders.

Testers voeren de volgende activiteiten uit:

- Uitschrijven of reviewen van fysieke testgevallen: deels kan dit door gebruikers en systeemtesters worden gedaan. Systeemtesters hebben al "fysieke" ervaring met systeemdelen en functionaliteit en weten waar knoppen zitten en controles kunnen worden uitgevoerd. Gebruikers kunnen de fysieke testgevallen beoordelen en voorzien van inzicht in de daadwerkelijke E2E-processen en verwachtingen. Hiermee wordt dan de E2E-test verrijkt met gebruikersacceptatieaspecten en kan het eindproduct al worden vergeleken met verwachtingen vanuit de gebruikersorganisaties. De E2E-test kan op deze manier eventueel worden gecombineerd met de acceptatietest van gebruikers of beheerders.
- Uitvoeren testgevallen: de handelingen die in de rol van gebruiker of klant moeten worden uitgevoerd, zoals het invoeren van een nieuwe klant, een nieuwe polis of een mutatie. Deze handelingen moeten bij voorkeur worden uitgevoerd door medewerkers die dit in productie ook doen.
- Uitvoeren beheeractiviteiten, zoals het draaien van achtergrondprogramma's of het traceren van gegevens: hier worden de beheerders ingezet die dit in productie ook doen.
- Controleren resultaten: dit moet bij voorkeur worden uitgevoerd door de medewerkers die dit in productie ook doen.
- Uitvoeren acceptatietesten (in het geval men besluit de E2E-test en de acceptatietesten te combineren). De E2E-test is een uitstekende plek en het juiste moment om kritische gebruikersaspecten te valideren omdat de E2E-testomgeving representatief is voor de productieomgeving en kennis voorhanden is om bevindingen te analyseren.

## 10.2 Plaats van de E2E-test in de organisatie

Een E2E-test is systeemoverstijgend, vaak projectoverstijgend (in het geval projecten onderling invloed op elkaar hebben) en soms zelfs organisatieoverstijgend. Delen van de keten kunnen onder andere organisaties vallen, of soms zijn eigen systemen bij andere organisaties ondergebracht voor wat betreft bouw en systeemtest.

De E2E-test is daarmee in essentie een grensoverschrijdende test. Organisatorische grenzen die worden overschreden betreffen:

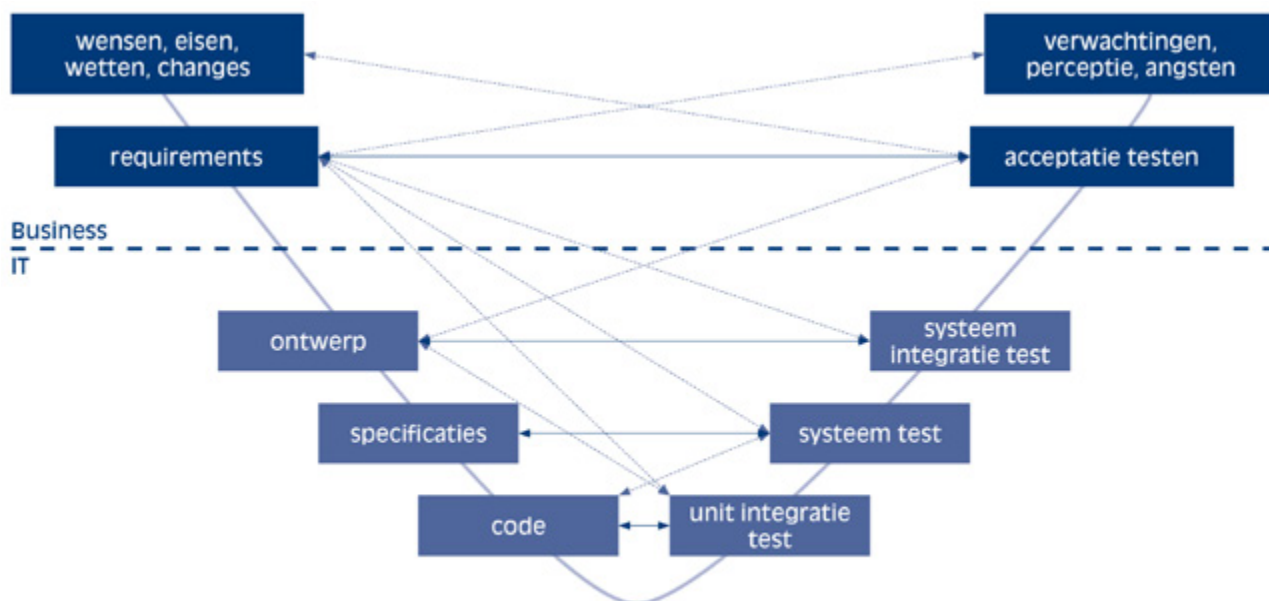
- De grens tussen IT en business.
- De grens tussen projecten.
- De grens tussen organisaties.

De organisatorische implicaties en keuzemogelijkheden van deze drie grensoverschrijdingen worden in de volgende paragrafen besproken.

### 10.2.1 Grensoverschrijding tussen IT en business: systeemtest of acceptatietest?

In hoeverre is de E2E-test een systeemtest of een acceptatietest? Met andere woorden: valt de E2E-test onder IT of onder de accepterende organisatieonderdelen? De systeemintegratietest is te beschouwen als de systeemtest van alle systemen samen en de organisatie van de systeemintegratietest valt daarmee onder IT. In het geval van een E2E-test liggen de zaken genuanceerder.

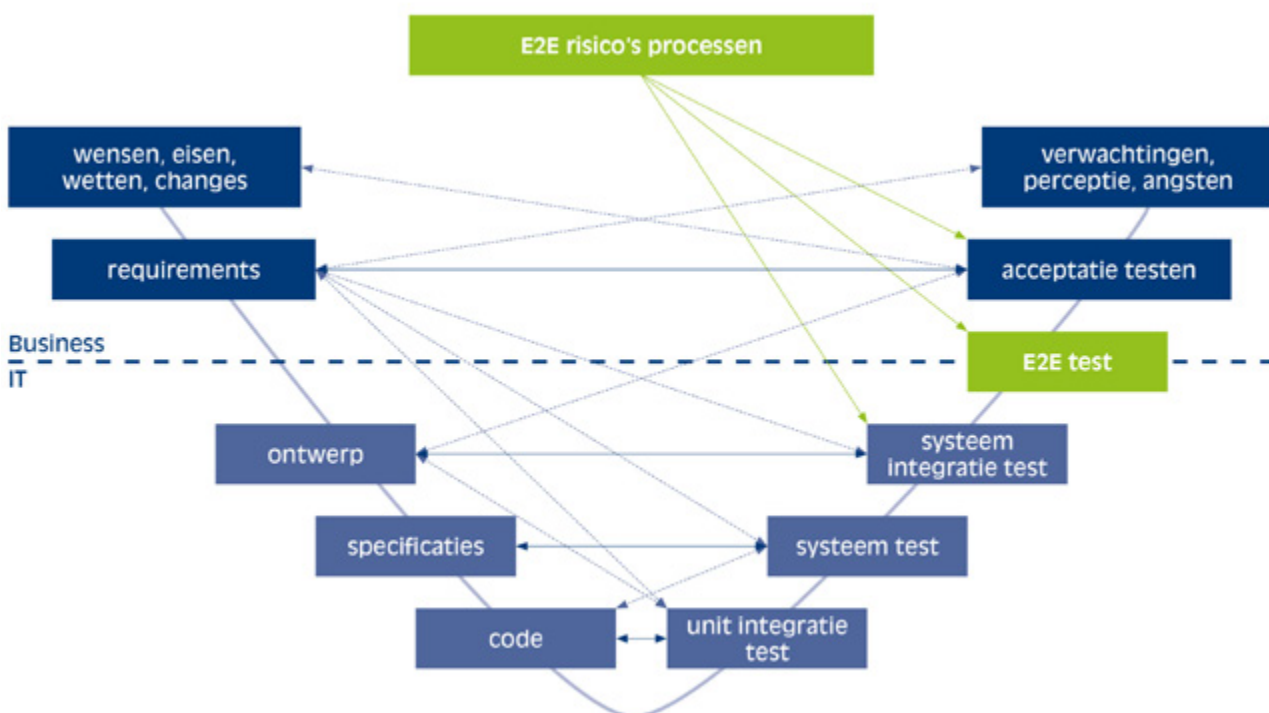
Hieronder wordt het welbekende V-model weergegeven. Hierin geldt de systeemintegratietest als de systeemtest van meerdere systemen samen en valt deze daarmee onder IT.



Figuur: V-model zonder E2E-test: SIT is IT

De aspecten die worden getest en de benodigde kennis, dwingen de E2E-test om met de gebruikers- en beheerorganisatie samen te werken. Betrokkenen bij de E2E-test zijn niet uitsluitend IT-medewerkers. De leden van de E2E-board en testers zijn o.a. ervaren gebruikers en beheerders. Daarmee worden ook acceptatieaspecten tijdens de E2E-test geraakt en gedekt.

Het V-model moet daarom worden verrijkt met een E2E-test die zowel onder de business als de IT-afdeling valt.



Figuur: V-model met E2E-test: grens IT en business vervaagt

Het ligt daarom voor de hand om de E2E-test, qua organisatie, als een hybride te zien van systeem- en acceptatietesten. Nu ziet men vaak dat de acceptatietesten E2E-aspecten dekken, of dat E2E-testteams en

acceptatieteams samen optrekken. Budgettering, kosten en bemensing moeten dan in overeenstemming hiermee worden ingevuld.

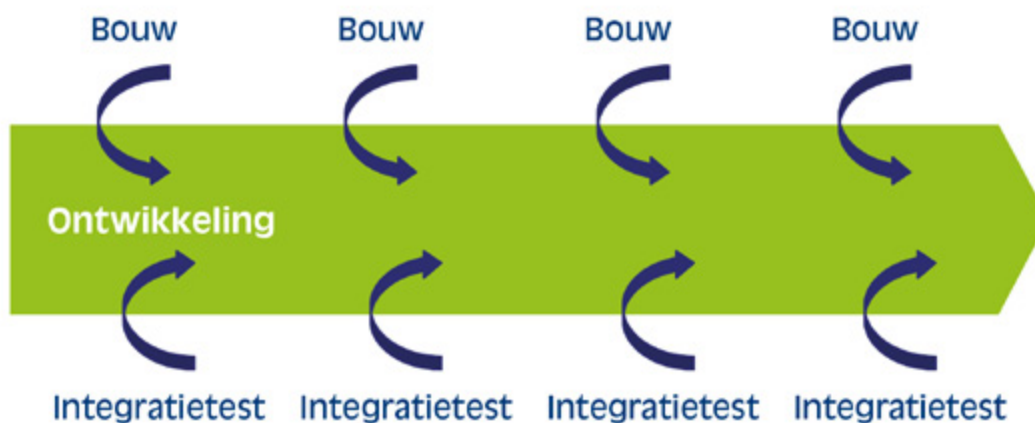
Tijdens de E2E-test moet nog wel onderscheid gemaakt worden tussen E2E-specifieke aspecten en acceptatieaspecten. Indien men namelijk de E2E-testgevallen als de acceptatiecriteria voor gebruikersafdelingen gaat zien, loopt men het risico gebruikersaspecten te missen. Beide groepen van aspecten (E2E en gebruikersacceptatie) kunnen wel goed in onderlinge samenwerking worden gedekt.

### 10.2.2 Grensoverschrijding tussen projecten

#### - Continue Integratie \*\*\*

In het geval meerdere projecten parallel wijzigingen uitvoeren op dezelfde systemen, bestaat de kans dat deze projecten de kwaliteit van elkaars wijzigingen beïnvloeden (regressie). Deze projectoverstijgende risico's worden vaak niet vanuit de projecten voorzien. In het geval plannings gaan schuiven of projecten onderling wisselen, wordt dit extra ingewikkeld. Er zal per moment van inproductienamen rekening gehouden moeten worden met extra uit te voeren regressietesten. Een E2E-testset is een uitstekende basis voor zo'n regressietest

Om deze redenen heeft een aantal grote organisaties een zogenaamde "Continue Integratie" opgezet. De term "Continue Integratie" komt uit de wereld van systeemontwikkeling en betekent oorspronkelijk het dagelijks uitvoeren van integratietesten in het geval van dagelijkse deelopleveringen ("daily builds") op een gemeenschappelijke omgeving.

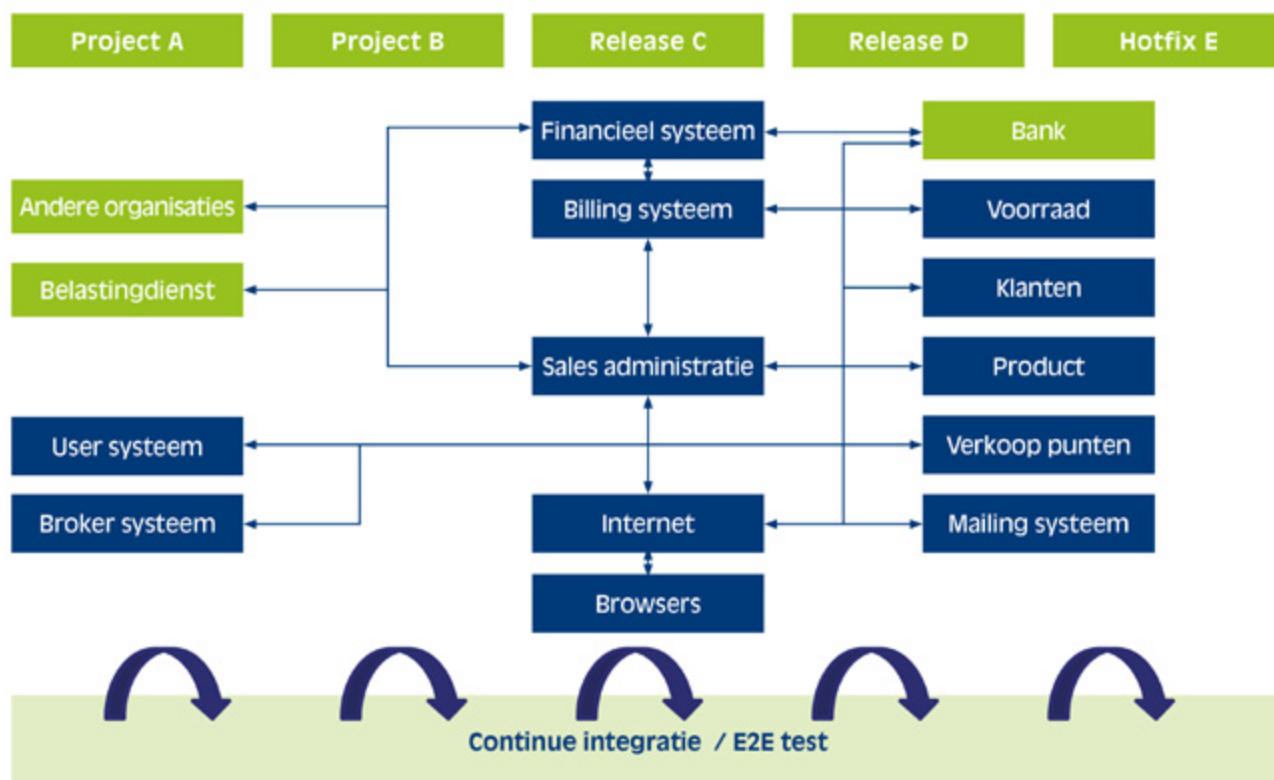


Figuur: Continue Integratie op unittest niveau

Het doel van Continue Integratie is om efficiënt en frequent regressie te kunnen traceren en op te lossen. Het concept leent zich met name voor iteratieve ontwikkeling.

In een systeemlandschap waar parallel wijzigingen door meerdere projecten, releases en bugfixes worden aangebracht, moet er op systeemintegratieniveau Continue Integratie plaatsvinden. De wijzigingen per project in systemen en systeemonderdelen kunnen dan als de "daily builds" worden gezien. Deze moeten dan door middel van een beperkte E2E-testset periodiek worden gehertest.





Normaal zal elk van deze projecten regressietesten als onderdeel van zijn projectactiviteiten hebben opgenomen. Maar hoe zit het met de regressie vanuit projecten op andere projecten? Daarnaast kan het duur en tijdsintensief zijn indien elk project een aparte integrale testomgeving en een aparte E2E-test opzet. Dit zijn aanvullende redenen om een centrale E2E-test en E2E-testomgeving in te richten.

#### - E2E-testcentrum \*\*

In enkele organisaties is het model van een continue integratie/E2E-test geïmplementeerd zoals in bovenstaande afbeelding is weergegeven. Een projectoverstijgend E2E-testcentrum, met een eigen integrale testomgeving, voert bij elke significante wijziging in de infrastructuur, een aangepaste E2E-test uit om grote regressierisico's af te dekken. De projecten dragen elk, als een soort abonnementsgeld, bij aan de bemensing, de kosten en de testomgeving. De noodzaak voor een dergelijke geformaliseerde E2E-organisatie hangt af van de risico's, de grootte van de infrastructuur en de complexiteit van het releasemanagement.

Een E2E-testcentrum kan ook de functie vervullen van kenniscentrum: E2E-inventarisaties worden hier gearchiveerd, actueel gehouden en beschikbaar gesteld aan testtrajecten. Kennis die aldus door E2E-testers wordt opgebouwd kan ook nuttig zijn voor anderen, bijvoorbeeld in het geval van ontwerp of het opstellen van werkinstructies.

Daarnaast is het E2E-testcentrum de lijnorganisatie van E2E-testers en E2E-testmanagers.

#### 10.2.3 Grensoverschrijding tussen organisaties

Indien systeemlandschappen tussen organisaties nauw samenwerken en deze samenwerking risicovol is, heeft men een organisatieoverschrijdende E2E-test nodig. Er zijn diverse gradaties waarin men dit kan opzetten.

##### 1. Organisatieoverschrijdende E2E-test per wijziging \*

Elke organisatie voert zijn eigen E2E-testen uit op het eigen systeemlandschap. Tijdens deze E2E-testen worden bij

gelegenheid andere organisaties betrokken, bijvoorbeeld om een gewijzigde interface te testen of een gewijzigd E2E-proces te hertesten. Deze testen worden opgezet vanuit de betreffende projecten. De organisaties hebben geen aparte testomgevingen die alleen voor dit doel worden gebruikt: vaak zijn het systeemtestomgevingen die tijdelijk over de organisatiegrens heen, met elkaar worden verbonden.

## 2. Permanente E2E-testomgeving voor organisatieoverschrijdende testen \*\*

Indien organisatieoverschrijdende testen periodiek terugkeren is het nuttig om een permanente E2E-omgeving operationeel te houden. Vanuit deze testomgeving kunnen E2E-testen over de organisatiegrens worden uitgevoerd. Niet alle betrokken organisaties hoeven zo'n omgeving te hebben. Het zijn vaak de centrale organisaties, die als een spin in het web van bepaalde processen zitten, die een dergelijke testomgeving hebben (bijvoorbeeld de Belastingdienst). Bij wijzigingen in interfaces of processen kunnen dan andere organisaties testen uitvoeren op deze testomgeving.

## 3. Organisatieoverschrijdend E2E-testcentrum \*\*\*\*

Een testorganisatie kan ook onafhankelijk van de deelnemende organisaties opgezet worden. Dit is een extreem geval. De reden kan zijn dat de overheid onafhankelijke testen afdwingt, of dat de organisaties onderling een testcentrum laten opzetten waarvanuit interfacetesten en E2E-testen worden gefaciliteerd. Een voorbeeld hiervan is de Telecom. Hier moeten concurrenten (Telecom operators) onderling testen uitvoeren. In sommige gevallen wordt dit afgedwongen door de overheid. Er kan dan sprake zijn van een overkoepelende instelling die de E2E-testen uitvoert of coördineert. Alle betreffende organisaties nemen deel aan de E2E-testen en dragen bij in de kosten en de bemensing.

## 10.3 Eigenaarschap

Testers moeten weten voor wie ze testen. Over het algemeen is dit de opdrachtgever. In het geval van de systeemtest is dit de projectmanager en in het geval van een gebruikersacceptatietest het afdelingshoofd. Voor een systeemintegratietest of E2E-test is het eigenaarschap meer complex. Het heeft dan zin om onderscheid te maken tussen het technisch eigenaarschap, het functioneel eigenaarschap, de escalatiekanalen en de acceptanten.

### 10.3.1 Technisch eigenaarschap

Het technisch eigenaarschap betreft de verantwoordelijkheid voor het beheer van de systemen. Dit is met name van belang als het gaat om de voortgang van de testen. Hier is de beslissingsbevoegdheid te vinden als het gaat om keuzes met betrekking tot het oplossen van bugs of het inrichten van omgevingen. Daarnaast moet er vanuit de techniek worden gekeken naar risico's voor de technische kwaliteit bij inproductienamen (onderhoudbaarheid, uitvoerbaarheid van batches en dergelijke). In de regel wordt dit ingevuld door het hoofd beheer.

### 10.3.2 Functioneel eigenaarschap

Het functionele eigenaarschap heeft betrekking op het uiteindelijke doel van de systemen: het vermogen de E2E-processen van de organisatie te ondersteunen. Hier treffen we de beslissingsbevoegdheid aan als het gaat om het accepteren van de keten van systemen voor wat betreft de functionele kwaliteit. Functioneel eigenaarschap kan verspreid zijn over een aantal systeemeigenaren of afdelingshoofden.

Beide type eigenaren, technisch en functioneel, hebben daarmee een eigen acceptatieverantwoordelijkheid en zijn belangrijke klanten van de E2E-test.

### 10.3.3 Ketenmanagement

Er kunnen in een grote E2E-keten meerdere functionele en technische eigenaars zijn. In dat geval is het van belang om deze verschillende bevoegdheden goed te benoemen en af te spreken hoe zal worden omgegaan met voorkomende situaties, wie welke beslisbevoegdheid heeft en dergelijke. Het kan dan raadzaam zijn om één eigenaar als de overkoepelende ketenmanager aan te stellen of hier een aparte functie van te maken.

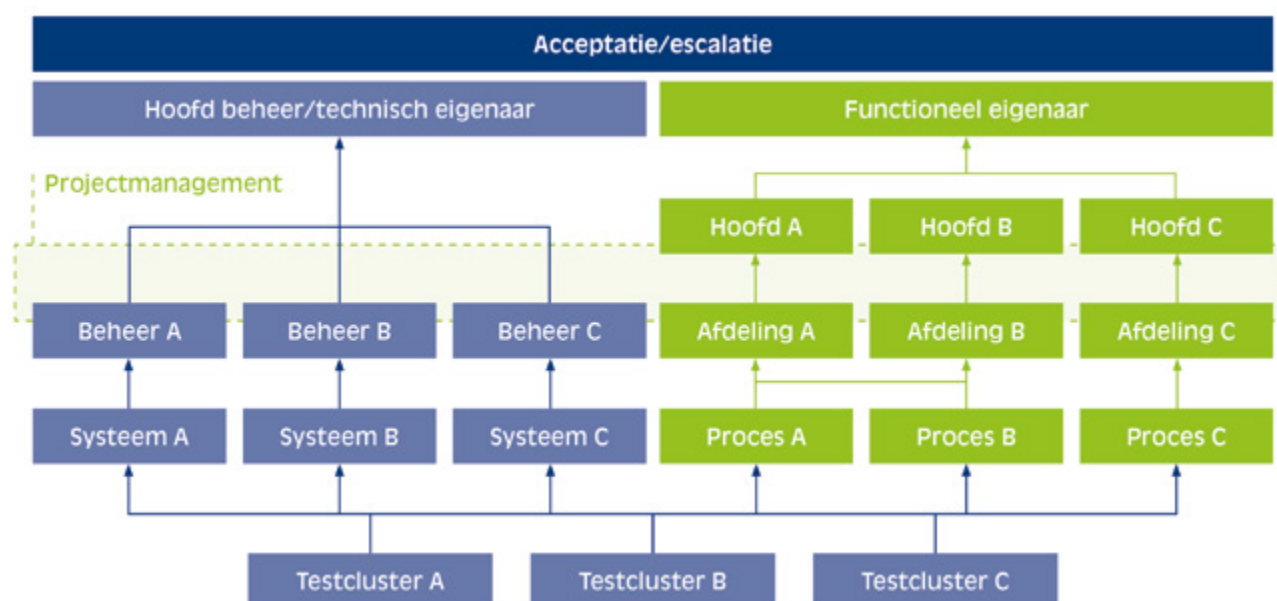
### 10.3.4 Escalatiekanalen

Het technische en het functionele eigenaarschap laten allebei een hiërarchische piramide zien, waarin op het laagste niveau functioneel en technisch beheerders, een stap hoger de systeemverantwoordelijken en aan de top de uiteindelijke functionele dan wel technische eigenaar staan. Deze piramiden kunnen als escalatiekanalen worden gezien, voor situaties waarin de scope en kwaliteit van het eindproduct onder druk komen te staan, om de vraag te beantwoorden welke bevindingen opgelost moeten worden of te beslissen welke partij een bevinding moet oplossen. In de praktijk zal de E2E-board de eerste instantie zijn waar dit soort zaken worden behandeld en van waaruit eventueel wordt geëscaleerd.

### 10.3.5 Acceptanten

Acceptanten zijn meestal de systeemeigenaren en de proceseigenaren. Dit kunnen hoofden van een gebruikersafdeling, hoofden infrastructuur en beheer, maar ook vertegenwoordigers van klanten of andere organisaties zijn. Dit kan zelfs de politiek zijn, denk hierbij bijvoorbeeld aan de belastingdienst. Het doel van een test, ook die van de E2E-test, is om acceptanten te voorzien van de juiste informatie zodat deze tot het juiste acceptatiebesluit kunnen komen. Voor een systeemtest is de projectleider de acceptant. Voor een gebruikersacceptatietest is dit hoogstwaarschijnlijk het hoofd van de betreffende gebruikersafdeling.

Een E2E-test zal met vrijwel alle acceptanten te maken krijgen en zal zich daarom in een vroeg stadium er van moeten vergewissen welke acceptant voor welk deel of aspect verantwoordelijkheid draagt. Afstemming van risico's, scope en acceptatiecriteria, maar ook testrapportages hebben belang voor deze acceptanten. Bovendien zijn het mogelijke medestanders in lastige discussies met andere teams. Hier kan de E2E-board een belangrijke rol spelen: de leden daarvan krijgen in de praktijk vaak de rol van de ogen en oren van de uiteindelijke acceptanten en hun nauwe betrokkenheid bij de testen zal de communicatie en het begrip van de acceptanten vergemakkelijken.



Figuur: Acceptatie- en escalatiestructuur

In bovenstaande figuur wordt de acceptatie- en escalatiestructuur weergegeven. De testclusters hebben verbindingen naar de E2E-processen die worden getest en de systemen die worden geraakt. Escalatie- en acceptatiekanalen volgen het spoor naar boven. De E2E-testmanager kan uit deze sporen het publiek voor de testrapportages afleiden.

Vanuit beheer en E2E-proces moet er uiteindelijk worden geaccepteerd. Bij discussie of onduidelijkheid beslissen de technisch en functionele eigenaars samen of wellicht nog hoger management.

Tussen deze partijen zweeft projectmanagement. Deze heeft niet dezelfde autoriteit als bij een systeemtest, waar projectmanagement de directe en belangrijkste opdrachtgever en acceptant is.

## 10.4 Schaling E2E-testen

In deze paragraaf worden aanwijzingen gegeven hoe E2E-testen moeten worden ingevuld. Dit kan verschillen per organisatie en E2E-keten.

Eerst wordt in deze paragraaf een overzicht gegeven van de “smaken” van integratie- en E2E-testen. Daarna worden de criteria geïnventariseerd om te bepalen welke invulling van toepassing is. Ten slotte wordt een aantal varianten uit de praktijk gegeven.

### 10.4.1 Integratie- en E2E-testen in testsoorten

Hieronder volgt een korte beschrijving van testsoorten en de mate waarin deze integratie- en E2E-aspecten kunnen dekken.

- **Systeemtest:** integratie van systeemonderdelen/componenten. Functionele test waarin deze onderdelen op hun onderlinge samenwerken worden beoordeeld.
- **Interfacetest:** test van de connecties tussen systemen. Is vaak een onderdeel van een systeemtest of als een aparte systeemtest opgezet (van de interface of het cluster van interfaces, bijvoorbeeld als dit als Middleware is opgezet). De testbasis bestaat uit functionele en technische ontwerpen van de interfaces.
- **Systeemintegratietest:** test van een functie die systeemoverschrijdend is. In feite wordt een cluster van systemen en interfaces als één systeem beschouwd en wordt deze functioneel getest. De testbasis bestaat uit globale ontwerpen, functionele ontwerpen en/of niet-functionele eisen.
- **Project E2E-test:** test waarbij de infrastructuur vanuit het blikveld van het project wordt onderworpen aan een procestest, waarbij men over de grenzen van de infrastructuur heen kijkt. Wie of wat wil iets en hoe wordt dit bereikt middels de infrastructuur? De testbasis bestaat uit een verzameling (de E2E-inventarisatie) van procesbeschrijvingen, interfacebeschrijvingen en architectuurplaten.
- **Projectoverschrijdende E2E-test:** E2E-test in het geval er meerdere projecten tegelijk wijzigingen op dezelfde infrastructuur uitvoeren en deze projecten elkaar kunnen beïnvloeden. Deze test is een gezamenlijke inspanning van deze projecten.
- **Organisatieoverschrijdende E2E-test:** E2E-test in het geval meerdere organisaties gevolgen kunnen ondervinden van wijzigingen in de gezamenlijk gebruikte infrastructuur. Deze test is een gezamenlijke inspanning van de betrokken organisaties of een test door een organisatie-overschrijdend orgaan.
- **Acceptatietesten:** testen door eindgebruikers (beheerder, medewerkers van helpdesks en dergelijke) om vast te stellen in hoeverre requirements zijn ingevuld en systemen de E2E-processen voldoende ondersteunen. De testbasis bestaat uit procesbeschrijvingen, werkinstructies en requirements. In een acceptatietest worden vaak interface-, integratie- en E2E-aspecten impliciet getest maar mist men de vaardigheden en kennis om dit diepgravend te doen.

In een extreem geval moeten elk van deze testen daadwerkelijk worden uitgevoerd en als aparte deelprojecten worden opgezet. Dit komt echter in de praktijk weinig voor. Vooral organisatieoverschrijdende testen worden door een project E2E-test meegenomen. E2E-testers nemen dan contact op met hun collega's bij de andere organisaties en organiseren gezamenlijke testomgevingen en testen.

Daarnaast komt het in de praktijk vaak voor dat een systeemintegratietest en een E2E-test of een E2E-test en een acceptatietest in nauwe samenwerking worden uitgevoerd en dan als één deelproject zijn te beschouwen. Elk van deze testen vertegenwoordigt echter wel afzonderlijke aspecten en dekt specifieke risico's.

Het nadenken over de mate waarin interfacetesten, SIT en E2E-testen moeten worden ingevuld, speelt zich af in meerdere dimensies. Voor elk van de dimensies moet worden afgewogen wat de noodzaak is (in termen van risico's) en wat de mogelijkheden zijn van de organisatie (in termen van budget, tijd, organisatie, kennis en cultuur).

### 10.4.2 Afwegingen bij de invulling van een E2E-test

Van de volgende zaken moeten de specifieke omstandigheden van een organisatie worden beschouwd:

- **Productrisico's.** Afhankelijk van de hoogte van de risico's en de mate waarin deze alleen zijn af te dekken door een E2E-test, varieert de noodzaak voor het opzetten van een E2E-test.
- **Grootte/cultuur van de organisatie.** Idealiter zijn een interfacetest, SIT en E2E-test ingericht als testtypen en geen afzonderlijke testsoorten. Deelactiviteiten en testclusters kunnen vaak grotendeels worden belegd bij systeemtesten en in nauwe samenwerking met acceptatietesten worden uitgevoerd.
- **Geografische verspreiding,** het zich bevinden in een complexe en grote organisatie, een hoge mate van juridische en culturele formaliteit en scherpe afbakening tussen organisatieonderdelen zijn redenen om een SIT en E2E-test formeler in te steken. Activiteiten moeten dan in meer detail worden gepland en er moeten meer gedetailleerde afspraken vooraf worden gemaakt over uit te voeren werkzaamheden en inzet van medewerkers. Dit zal druk uitoefenen op de doorlooptijd. Het vergroten van het E2E-kernteam kan dan een voordeel opleveren. De afhankelijkheid van de medewerkers neemt dan namelijk af en voorbereidingen kunnen gedetailleerder worden uitgevoerd.
- **Grootte/complexiteit van het systeemlandschap.** Complexe systeemlandschappen zijn een reden om een E2E-testset als regressietest op te zetten. De E2E-test wordt dan ook een periodieke regressietest van alle wijzigingen en projecten in het systeemlandschap. Dit kan in het uiterste geval organisatieoverstijgend zijn, zoals een regressietest van interoperabiliteit tussen mobiele netwerk aanbieders. Organisatorisch kan hieruit volgen dat er een aparte E2E-afdeling opgezet wordt, waarin een permanent operationele testomgeving wordt beheerd en waarvuit E2E-activiteiten worden gecoördineerd en uitgevoerd.
- **Projectmatige verspreiding.** Cloud Computing en uitbesteding zijn omstandigheden waarin projectactiviteiten zijn verspreid over organisaties. Beheer, ontwikkeling en testactiviteiten worden "elders" uitgevoerd. In de praktijk betekent dit vaak dat qua integratie en E2E-aspecten extra product- en projectrisico's worden gelopen. Men heeft vaak minder zicht op de te verwachten kwaliteit. Ontwikkelpartijen hebben moeite om E2E-processen en daarmee de E2E-risico's te begrijpen en door middel van testen af te dekken. Een SIT en E2E-test geregisseerd of zelfs uitgevoerd door de klantorganisatie wordt dan van groter belang. Dit kan een heikel punt zijn omdat één van de doelen van uitbesteding het afstoten van projectactiviteiten is. Uitbesteding kan echter betekenen dat juist een E2E-test zwaarder moet worden aangezet vanuit de eigen geledingen. De E2E-test komt dan vaak bij de acceptatietest, en dus de gebruikersorganisaties, te liggen, omdat zij de laatste "quality gate" naar productie zijn. Om een goede systeemintegratietest of E2E-test uit te voeren is de gebruikersacceptatietest echter vaak niet voldoende toegerust.

Afhankelijk van de bovenstaande analyse, moeten de volgende keuzes worden gemaakt voor wat betreft de invulling van interfacetesten, SIT en E2E-test:

#### Organisatie

- Lijnorganisatie: hoe moeten de testen worden ondersteund? Moet er een aparte E2E-organisatie worden opgezet of worden de testen bemenst door de projecten zelf, afwisselend met projectmedewerkers, gebruikers, beheerders en aangevuld met externe testers?
- Moeten aparte teams voor IT, SIT en/of E2E-testen worden opgezet?
- Moeten specialisten worden ingehuurd, zoals E2E-testers, efficiency- en securitytesters?
- Grootte van het E2E-kernteam: in welke mate kan of moet het E2E-kernteam alle activiteiten uitvoeren en kunnen of moeten andere medewerkers worden betrokken?
- Is een aparte, project- of organisatieoverstijgende E2E-test nodig en mogelijk?

#### Infrastructuur

- Zijn aparte testomgevingen per IT, SIT of E2E (of zelfs per testcluster) nodig en mogelijk?
- Is simulatie van systemen en systeemonderdelen noodzakelijk en mogelijk door middel van tooling en de ondersteuning daarvan? Hierdoor kunnen SIT en E2E-aspecten eventueel vroeg en goedkoop worden getest maar kan de waarde van een E2E-test worden gecompromitteerd.
- Kunnen testactiviteiten worden geautomatiseerd? Dit vergt een initiële investering die moet kunnen worden terugverdiend. Daarnaast is er een organisatie (kennis, beheer) nodig om dit mogelijk te maken en in stand te houden.

### Fasering

- Aparte IT, SIT en E2E-test: zijn dit aparte teams of deelprojecten? Wat is de onderlinge afhankelijkheid: start de E2E-test pas als de SIT klaar is?
- Testtype of testsoort: worden de activiteiten belegd in aanwezige projectfasen (zoals de systeemtest en acceptatietest) of moeten hier aparte fasen en projectteams voor worden ingericht (SIT en E2E-test als testsoort)?
- Welke testtypen binnen SIT en E2E (zoals efficiency, security, failover) moeten worden opgezet? Hebben deze een eigen team, fasering en testomgeving?

#### 10.4.3 Varianten uit de praktijk

In deze paragraaf wordt een aantal varianten van invulling van een SIT en E2E geschetst. In deze varianten komen de bovengenoemde dimensies tot uiting. De varianten zijn voorbeelden uit de praktijk en worden geordend in toenemende mate van grootte en formaliteit. De lezer kan zijn eigen context hierin projecteren en de contouren van de voor hem meest geschikte invulling vinden.

#### **- Kleine organisatie: ST en IT gecombineerd, SIT, E2E en GAT gecombineerd**

##### **Context.**

Een kleine organisatie (rond 500 medewerkers) met een kleine, eenvoudige systeemarchitectuur: de basis is een centrale polisadministratie en een financieel systeem. Vanuit het project vindt een ingrijpende wijziging plaats op de centrale ondersteunende administratie. Er zijn een handvol E2E-processen die een connectie hebben met externe organisaties. De diverse betrokken afdelingen werken op één locatie, er is een informele cultuur en een grote betrokkenheid bij het project. Er is een eigen beheerafdeling maar geen eigen ontwikkelafdeling en er wordt gebruik gemaakt van leveranciers van standaardpakketten.

##### **Invulling.**

Leveranciers voeren eigen testen uit, maar hier wordt geen strenge regie op gevoerd. Hierdoor worden systeemtesten door de organisatie ook zelf uitgevoerd (een zogenaamde Functionele Acceptatie Test). De systeemtesten worden vanuit het project voorbereid en uitgevoerd door ingehuurde testexperts, aangevuld door gebruikers van de diverse afdelingen. Testcoördinatie vindt plaats door eigen beheerders en ingehuurde adviseurs. Interfacetesten worden gezien als onderdeel van de systeemtesten. SIT, E2E-testen en GAT worden gezamenlijk en op één testomgeving uitgevoerd. Door testexperts worden de SIT en E2E-testgevallen gecoördineerd en hierbij wordt gebruik gemaakt van de expertise van ervaren gebruikers. Waar mogelijk wordt getracht gebruikersaspecten al tijdens de systeemtest te beoordelen. De testuitvoering is een gezamenlijke inspanning van de E2E-testers, de ervaren gebruikers en de systeemtesters.

Na de diverse systeemtesten vindt er eerst een SIT plaats. Hierbij worden stapsgewijs de functionele interacties tussen systemen getest met laboratoriumdata. Na 1 of 2 testronden wordt er steeds meer met een complete infrastructuur en productiedata gewerkt. De laatste testronde is een "echte" E2E-test waarbij de gehele infrastructuur als een black-box wordt gezien en E2E-processen worden gestart en gecontroleerd. Hierbij worden ook dagelijkse werkzaamheden door gebruikers van afdelingen uitgevoerd.

De betrokkenheid van alle delen van de organisatie was groot en werd steeds groter, vooral omdat de E2E-testomgeving een "speeltuin" werd waar een ieder in staat werd gesteld praktische kennis op te doen van het nieuwe systeem en de nieuwe processen. In de E2E-test werden belangrijke, grote, bevindingen gedaan waardoor de E2E-test steeds meer de status kreeg van de finale acceptatietest.

#### **- Middelgrote organisatie met uitbesteding: ST afzonderlijk, SIT/E2E/GAT gecombineerd**

##### **Context.**

Een middelgrote financiële organisatie met daarbinnen verschillende domeinen (zoals bank, schade, hypotheek, beleggingen) met elk een eigen systeemarchitectuur. De systemen van de diverse domeinen hebben periodiek



interactie met elkaar, bijvoorbeeld als het gaat om een centrale klantenadministratie of gecombineerde producten. De afgelopen jaren zijn het merendeel van de centrale systemen uitbesteed. De cultuur binnen de organisatie is informeel, wat zich weerspiegelt in de contracten met leveranciers waar partnerschap voorop staat. Hierdoor kunnen uitvoerende medewerkers van de klant direct communiceren en samenwerken met medewerkers van de leverancier. Een nadeel van de informele werkwijze is dat de opgeleverde kwaliteit onvoorspelbaar is. Er zijn organisatiebreed vier vaste releasedatums per jaar waarbij problemen met de doorlooptijd binnen projecten worden opgelost door extra inzet en overwerk of de acceptatiecriteria en de inhoud aan te passen.

### **Invulling.**

Systeemtesten worden door de leveranciers uitgevoerd op eigen omgevingen. De leveranciers blijken moeite te hebben met de invulling hiervan: het ontbreekt hen vaak aan kennis van testen en de E2E-processen en producten bij de klant. Gevolg hiervan is dat de oorspronkelijk geplande, onafhankelijke, GAT is uitgegroeid tot een gecombineerde SIT/E2E/GAT, waarin men naast integratie en gebruikersaspecten ook een uitgebreide regressietest op functionaliteit uitvoert. Deze gecombineerde test wordt bemand door interne en externe test-specialisten en gebruikers. Naast deze combinatie van testsoorten vinden per project korte E2E-testen plaats over de domeinen heen. Vanuit de SIT/GAT/E2E wordt steeds meer samengewerkt met de leverancier om de kwaliteit van de systeemtest te verhogen.

De flexibiliteit wordt uiteindelijk als voordeel ervaren. Door het samenwerken van diverse expertises is men in staat om in korte tijd veel te testen en te hertesten. Nadeel is de grote druk die testtrajecten op de gebruikersorganisaties uitoefenen. Er zijn relatief veel gebruikers langere tijd bezig met testen die aanvankelijk als de verantwoordelijkheid van IT werden gezien.

### **- Grote organisatie, uitbesteding: SIT/E2E gecombineerd, ST en GAT als afzonderlijke testsoorten**

#### **Context.**

Een grote Telecom aanbieder, met diverse producten op het gebied van mobiele en vaste telefonie, internet en televisie. De organisatie heeft een commerciële en formele cultuur en besteedt alle automatisering zoveel mogelijk uit.

#### **Invulling.**

De systeemtesten worden uitgevoerd door de leveranciers van de systemen. Daarnaast is een SIT/E2E-test uitbesteed aan een andere leverancier. Deze voert deze testen uit op eigen omgevingen en een eigen locatie. Binnen deze SIT/E2E wordt ook een korte Functionele Acceptatie Test uitgevoerd. De klantorganisatie voert zelf de GAT uit op haar eigen omgeving. Er zijn strakke afspraken over de diverse testen en de klantorganisatie voert regie over de verschillende testen.

Aanvankelijk werd geworsteld met de kwaliteit, de doorlooptijd en de afstemming tussen de diverse testen. Toch is vastgehouden aan deze verbrokkeling. Men wil zich richten op de kernactiviteiten. De organisatie heeft daarom de strategische keuze gemaakt flexibel te zijn in bemensing en kosten. Door een simpele en rigide procedure rond regie en afstemming heerst nu tevredenheid.

### **- Overheid (groot): aparte ST en SIT, afzonderlijke E2E-testomgeving**

#### **Context.**

Een grote overheidsorganisatie waarvan de automatiseringsafdeling diverse diensten van de overheid ondersteunt. Kwaliteit van de diensten heeft een hoog belang. De cultuur van de organisatie is procedureel.

#### **Invulling.**

Er is een apart, projectoverstijgend, organisatieonderdeel opgezet dat een integrale testomgeving onderhoudt. In dit organisatieonderdeel zijn aparte teams opgezet voor beveiliging en efficiency. Deze teams moeten verplicht



betrokken worden bij projecten en voeren testen uit op de integrale omgeving, voorafgaand aan inproductie-name. Daarnaast zijn er meerdere afdelingen, per domein, waarbij een domein een eigen systeemtestonderdeel heeft die ST en SIT uitvoert per project. Andere domeinen (zoals Inkoop) zetten per project systeemtesten en systeemintegratietesten op en laten die uitvoeren door beheerders, aangevuld met testspecialisten.

Per project worden gebruikersorganisaties betrokken om een GAT uit te voeren.

### **- Internationale organisatie: ST, IT, SIT, E2E en GAT als afzonderlijke testsoorten**

#### **Context.**

Een grote internationale financiële instelling. Er is geregeld sprake van systeem- en projectoverstijgende wijzigingen zoals fusies en afsplitsingen. De cultuur is formeel en er worden hoge eisen gesteld aan projecten, die zowel intern (productrisico's) als extern (SOX, aangescherpte financiële wet- en regelgeving) gemotiveerd zijn. Het systeemlandschap is een lappendeken waarin diverse leveranciers en beheerpartijen opereren. Daarnaast zijn er ook meerdere gebruikersorganisaties zoals helpdesks, backoffices en marketingafdelingen vanuit de verschillende domeinen (bank, verzekering, pensioen, beleggingen, hypotheek, schade, uitvaart).

#### **Invulling.**

Per project, systeem en leverancier vinden systeemtesten (door de leverancier) en Functionele Acceptatietesten (door de klantorganisatie) plaats. Per project zijn SIT en E2E-testen verplicht en elke gebruikersorganisatie voert per project een GAT uit.

Daarnaast is er sprake van complex releasemanagement waarin men regie houdt over alle wijzigingen in het systeemlandschap. Daarbinnen heeft men een organisatiebrede E2E-testorganisatie (het E2E-testcentrum) ingericht als onderdeel van releasemanagement dat projecten voorafgaand aan inproductie-name een periode op een integrale testomgeving E2E-testen laat uitvoeren. De E2E-testorganisatie beheert informatie over alle E2E-processen en testgevallen die als regressietest van de E2E-keten kunnen worden uitgevoerd.

De afzonderlijke, organisatiebrede, E2E-test en de daarbij horende organisatie wordt ervaren als een noodzakelijk middel om de kwaliteit en risico's te kunnen beheersen. De kennis die E2E-testorganisatie heeft ontwikkeld wordt inmiddels ook gebruikt door ontwerp- en bouwteams.



# Deel 11

## De E2E- testinfrastructuur

## Deel 11 De E2E-testinfrastructuur

In een E2E-test verschillen deze activiteiten op beslissende punten van de wijze waarop ze in andere testsoorten worden uitgevoerd. Een E2E-inventarisatie is in feite het verzamelen van de testbasis. In een E2E-test zullen de testers deze zelf grotendeels moeten opstellen. De teststrategie is in het geval van een E2E-test gericht op specifieke E2E-risico's. De testplanning van een E2E-test gaat uit van de testronden en de doorlooptijd van de testgevallen. Het testontwerp bestaat uit het slim combineren van de stappen uit de E2E-processen en de stromen door de systeemketen.

Dit deel van onze praktijkgerichte aanpak voor E2E testen behandelt de **E2E-testinfrastructuur** en omvat de volgende onderwerpen:

- Testomgevingen
- Testdata
- Tools

### 11.1 Testomgevingen

E2E-ketens stoppen niet bij de grens van een organisatie: als er systeemplaten worden getekend blijkt al snel dat vrijwel elke organisatie tegenwoordig connecties heeft met andere organisaties, het internet of algemene services zoals het Bureau Krediet Registratie of de Rijksdienst voor het Wegverkeer. Het voor het project relevante systeemlandschap moet daarom altijd worden afgebakend (dit gebeurt in feite al in de E2E-inventarisatie). Indien externe organisaties of algemene services een onderdeel worden van het project en de test, vereist dit aparte afstemming met de achterliggende organisaties over testomgevingen, bevoegdheden, testdata en uit te voeren technische processen.

Gebruik van gegevens en systemen moet worden afgestemd met alle betrokken partijen. Elke systeem in de testomgeving heeft dezelfde systeemdatum nodig. Het draaien van achtergrondprogramma's (zie hieronder) moet tussen systemen op elkaar afgestemd zijn en de testdata moet synchroon zijn door de systeemketen heen.

In deze paragraaf worden de aspecten van de E2E-testomgevingen kort besproken: de belangrijke keuzes, overwegingen en consequenties worden geïnventariseerd.

Met een achtergrondprogramma (of: batch) wordt een functie of proces bedoeld die door beheerders wordt gestart of ingepland en die onderdeel is van de geautomatiseerde afhandeling van processen. Zo draaien in polisystemen bij verzekeraars periodiek achtergrondprogramma's die uit de ingevoerde gegevens uitbetalingen of premie-inhoudingen destilleren en verwerken richting andere systemen in de keten. Bij het volgende systeem draaien dan weer periodiek andere achtergrondprogramma's die de resultaten oppakken en verder verwerken.

#### 11.1.1 Beheer

Het beheer van testomgevingen moet zo veel mogelijk door de beheerders worden uitgevoerd die daar ook in productie verantwoordelijk voor zijn. Beheer en inrichting van testomgevingen kunnen namelijk vrijwel nooit door het testteam zelf gedaan worden. Het testteam ontbeert de benodigde kennis en vaardigheden. De beheerders van de systemen moeten daarnaast zelf ook testen en de systemen op beheeraspecten accepteren. Het is daarom het meest efficiënt, en ook in overeenstemming met het doel van de E2E-test, om een ieder, ook beheerders, de taken die ze in productie hebben tijdens de E2E-test te laten uitvoeren.

Het E2E-testteam moet wel zo goed als mogelijk is inzicht krijgen in de beheerprocedures om het beheer zo goed mogelijk aan te sturen. Vanuit de testgevallen wordt door het E2E-testteam geïnventariseerd of er afhankelijkheden zijn die door beheer moeten worden ingevuld. Op basis hiervan wordt in nauwe samenwerking met

de beheerders een draaiboek van de testomgevingen opgesteld. In dit draaiboek moeten de volgende zaken zijn gedetailleerd:

- Wanneer welke achtergrondprogramma's moeten draaien.
- Welke instellingen moeten zijn geregeld.
- Welke draaidatums moeten worden gehanteerd en dergelijke. (Zie voor een voorbeeld van een draaiboek voor testomgevingen [Deel 12, pagina 120](#))

Voor het beheer van testomgevingen gelden de volgende regels:

- Systeembeheer wordt uitgevoerd door het beheerteam dat dit ook in productie doet.
- Beheerprocedures worden productiegelijk uitgevoerd.
- Er wordt een detailldraaiboek opgesteld per testomgeving (systeem) waarin alle activiteiten en de verantwoordelijkheden systematisch en chronologisch zijn uitgewerkt. De E2E-testmanager is hierbij leidend en hij onderhoudt en communiceert het draaiboek.
- Vermijd te grote formaliteit en onderhoud directe, persoonlijke contacten met beheerders. Dit betekent ook dat er vanuit het testteam moet worden geïnvesteerd in de relatie met beheerders. Dit kan bijvoorbeeld door beheerders te ondersteunen bij de analyse van productieproblemen en het opzetten van nieuwe beheerprocessen.
- In de E2E-board hoort een representatieve vertegenwoordiging te zijn van de beheerders. In geval van een groot aantal systemen moet hier een keuze worden gemaakt: bijvoorbeeld alleen seniore beheerders met overzicht over meerdere systemen en andere beheerders per systeem op afroep.

De E2E-board is een dagelijks bijeenkomend panel van specialisten waarin kennis over processen en techniek samenkomt en die functioneert als klankbord en vraagbaak voor het E2E-testteam.

### 11.1.2 Welke omgevingen?

Bij het kiezen van de juiste omgevingen spelen de volgende afwegingen:

- In welke mate moet de testomgeving productiegelijk zijn? Productiegelijkheid geeft de beste, want representatieve, resultaten en is beheerbaar omdat gelijke databestanden uit productie synchroon zijn over de keten. Productiegelijkheid kan echter ook duur zijn (vanwege de eisen die worden gesteld aan de hardware) of risicovol (vanwege de beveiligbaarheid van klantgegevens) of onmogelijk (vanwege wet- en regelgeving rond gebruik van productiedata).
- Het aantal testomgevingen: kunnen systeemtestomgevingen worden opgenomen in integratietestomgevingen en E2E-testomgevingen? Moet de E2E-omgeving dezelfde omgeving zijn waar ook de acceptatietesten op plaatsvinden? Welke testclusters hebben een eigen testomgeving nodig (zie onder)?
- Releasemanagement van de testomgevingen. Omdat E2E-testen vaak arbeidsintensief zijn en een lange doorlooptijd hebben, zal er behoefte aan stabiliteit zijn. Daarom moet men voorzichtig zijn met het doorzetten van wijzigingen in deelsystemen naar de E2E-testomgeving. Kleine haperingen kunnen lange en zware testen verstoren waardoor er uitloop ontstaat. Dit is dan een argument voor een scheiding tussen systeemtest- en E2E-testomgevingen.
- Indien de E2E-testomgeving kan worden ingezet als acceptatietestomgeving kunnen gebruikers (in de acceptatietest) worden blootgesteld aan ongeteste software. Hier tegen bestaat nog steeds huiver. Het is onze ervaring dat gebruikers, mits nauw betrokken, hier geen enkel probleem mee hebben. Het heeft daarnaast voordelen om een acceptatietest parallel of in samenwerking met een E2E-test uit te voeren. De kennis van gebruikers kan in een E2E-test van belang zijn en gebruikers kunnen in een E2E-test vroeger acceptatiecriteria valideren. Er kan dan doorlooptijd worden gewonnen.
- De scheiding tussen de testomgevingen van de E2E-test en de acceptatietest kan echter onvermijdelijk zijn. Als gebruik van productiedata een must is voor de acceptatietest en alleen productiegebruikers daarmee mogen werken dan kunnen externe medewerkers of niet-gebruikers niet op dezelfde omgeving werken.

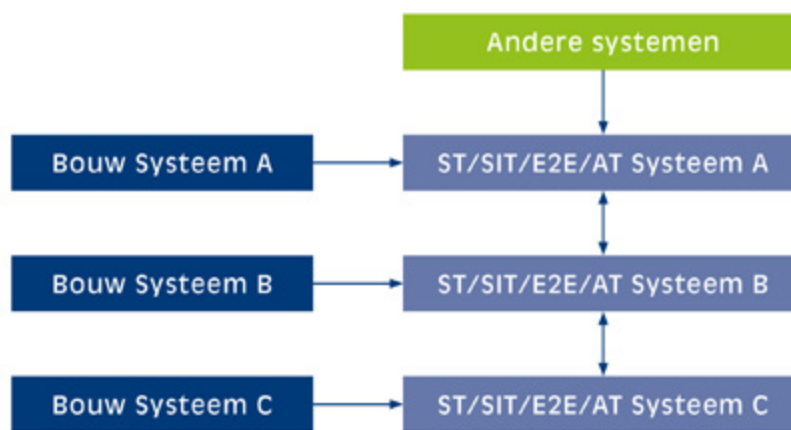
Of een testcluster een aparte testomgeving nodig heeft is afhankelijk van benodigde systeemtijd of tijdreizen, batches die testdata onbruikbaar voor andere testen kunnen maken (bijvoorbeeld conversies). In de praktijk zal het moeilijk zijn om een testomgeving voor een E2E-test te verdubbelen: andere systemen moeten namelijk ook verdubbelen en aan elkaar worden gekoppeld en synchroon worden gemaakt.

### 11.1.3 Varianten

De volgende basisvarianten in het invullen van testomgevingen worden onderscheiden:

#### Model 1: Systeemtestomgevingen aan elkaar gekoppeld

Dit is de meest eenvoudige optie. De systeemtestomgevingen worden steeds meer aan elkaar gekoppeld en zijn de omgevingen waar alle testen (na bouw) op plaatsvinden.



Voordelen:

goedkoop, minimale hoeveelheid op te zetten en te koppelen testomgevingen.

Nadelen:

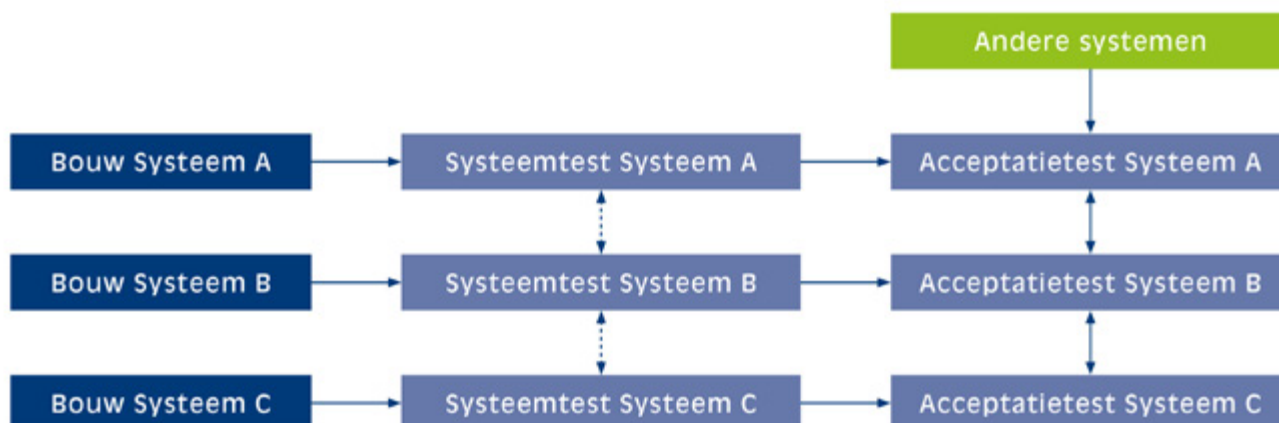
gezamenlijk datagebruik met invloed op elkaars voortgang en ongeteste versies op alle omgevingen met gevaar van afbraak en uitloop.

Randvoorwaarde:

nauwe samenwerking tussen systeemintegratie-, E2E- en acceptatietesten.

#### Model 2: OTAP

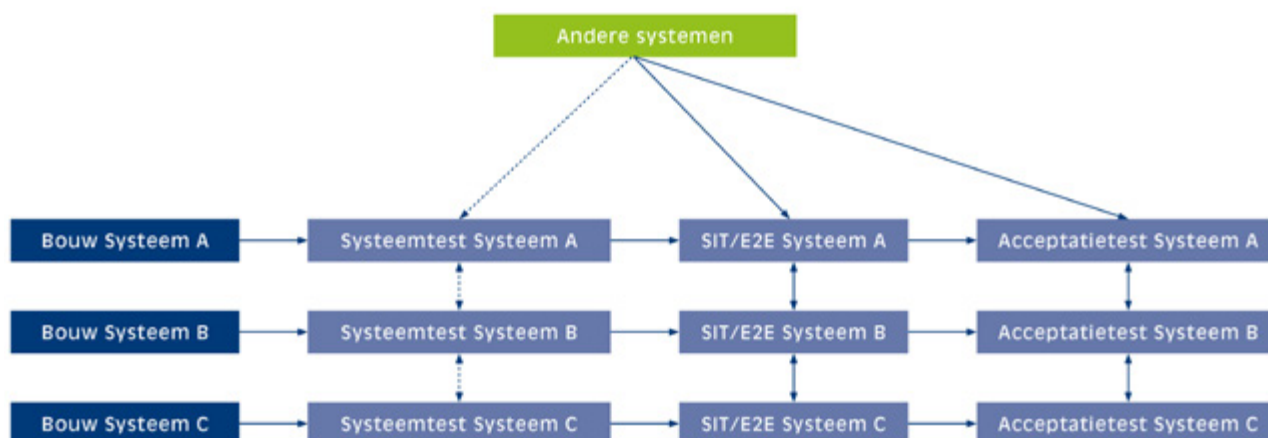
De acceptatietestomgevingen zijn permanent aan elkaar gekoppeld en vormen de enige E2E-testomgeving. Hierop vinden de systeemintegratie-, E2E en acceptatietesten plaats. Tijdens de systeemtesten worden interfaces aangelegd om interfacetesten en alvast delen van de systeemintegratietesten uit te voeren.



Voordelen:	relatief goedkoop.
Nadelen:	acceptatie en E2E moeten op elkaar worden afgestemd en er is samenwerking nodig tussen E2E-test en acceptatietest.
Randvoorwaarde:	nauwe samenwerking tussen systeemintegratie-, E2E- en acceptatietesten.

### Model 3: Project richt aparte E2E-omgeving in

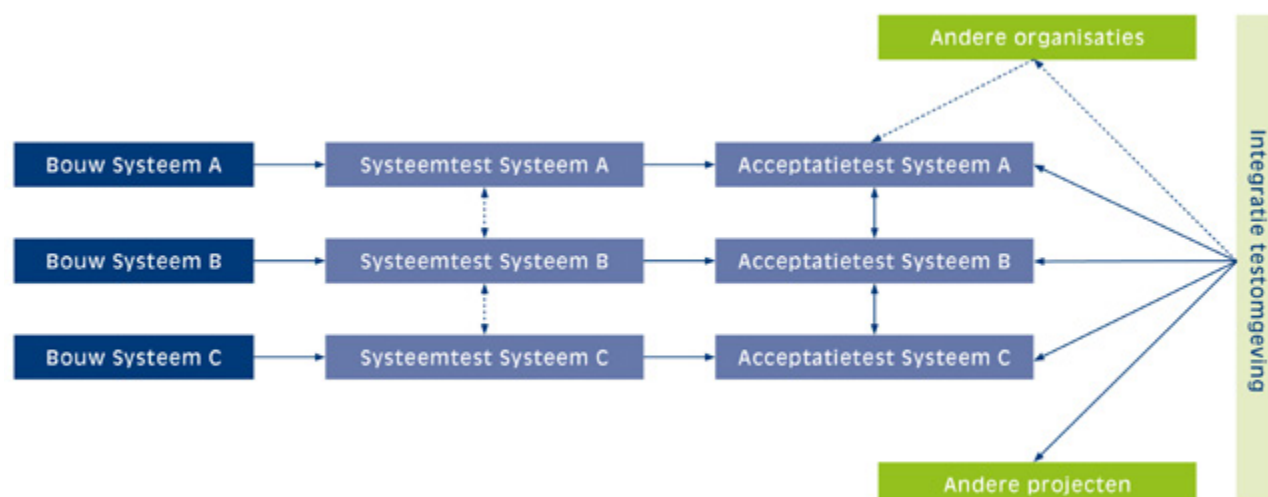
Het project richt, naast de acceptatietestomgevingen, een aparte omgeving in waar alle koppelingen en interfaces werkend zijn. De acceptatieomgevingen hoeven dan minder uitgebreid te zijn (dit is afhankelijk van de zwaarte en inhoud van de acceptatietesten).



Voordelen:	Elke test kan data en een omgeving inrichten naar eigen behoefte. In het geval er geen productiedata gebruikt mag worden door externe medewerkers kan dit een oplossing zijn.
Nadelen:	Duur. Vaak is het zo dat er maar één koppeling met externe systemen kan bestaan (deze stelt bijvoorbeeld maar één omgeving beschikbaar voor testen met anderen). In het laatste geval kan er maar één complete geïntegreerde testomgeving zijn.

### Model 4: Continue integratie

Een organisatiebrede integratieomgeving waar alle projecten langs moeten lopen en E2E-testen moeten laten uitvoeren. Projecten hebben hiernaast vaak nog een eigen, beperktere integratieomgeving zoals in model 3. Er ontstaat een projectoverstijgende, organisatiebrede E2E-test die regressietesten uitvoert voor alle projecten samen.



Voordelen:	Er is altijd een integrale, productiegelijke omgeving aanwezig. Hierdoor zijn wisselwerkingen tussen meerdere parallele projecten min of meer gedekt.
Nadelen:	Planning en beheersing van deze omgeving is complex. Vaak betekent dit dat een project een beperkte tijd krijgt op de omgeving. Het project is niet flexibel in het uitvoeren van testen. Er zal vaak ook nog een eigen E2E-omgeving nodig zijn. Een permanente integrale testomgeving is alleen interessant voor grote organisaties, vanwege de kosten en de organisatie die nodig is om de integrale omgeving te beheren.

#### 11.1.4 Continue integratie

In een groot systeemlandschap lopen vaak meerdere parallele wijzigingstrajecten. Denk hierbij aan verschillende projecten, releases of hotfixes. Elke wijziging moet getest worden tegen de omgeving zoals die op moment van inproductiename verwacht wordt. Daarnaast heeft elke wijziging een mogelijke regressie tot gevolg voor andere wijzigingen. Vanuit deze dynamiek is het concept van "continue integratie" ontstaan: hierbij wordt een complete E2E-omgeving permanent actief onderhouden. Elk project of wijziging moet in deze omgeving een aantal E2E-testen doorstaan. In de testomgeving van de continue integratie staan altijd de laatste versies van systemen, zoals deze in productie zijn tijdens het geplande moment van inproductiename.

Daarmee wordt de E2E-test een periodieke regressietest van het gehele systeemlandschap en de belangrijkste processen. Denk hierbij ook aan de invloed van steeds makkelijker en wendbaarder uit te voeren wijzigingen ten gevolge van Service Oriented Architecture (SOA) en Agile Development.

#### 11.1.5 De speeltuin: continue acceptatie

"Continue Acceptatie" en "Emotionele acceptatie" betekenen het aanbieden van faciliteiten om vroeg in het traject (wellicht continu) acceptanten en hun vertegenwoordigers in de gelegenheid te stellen te werken (of spelen) met nieuwe versies van het systeem. Een dergelijke omgeving zou dan een E2E-testomgeving kunnen zijn. Het mes snijdt dan aan twee kanten. Er wordt (sneller) naar een juiste mate van acceptatie en vertrouwen toe gewerkt en de kennis en de ervaringen van dergelijke acceptanten kunnen ook worden opgepakt door andere projectleden. Bovendien kunnen mensen met proceskennis op deze manier "geïnfecteerd worden met het testvirus" en wordt het gemakkelijker ze in te zetten in het project.

De achtergrond van deze concepten is dat acceptatie meer is dan een aantal gebruikers die aan het eind van het traject kijken of het eindproduct overeenkomt met gestelde eisen en wensen. Eisen en wensen zijn namelijk vaak niet eenduidig vastgesteld en zijn ook nog eens veranderlijk. Daarnaast is acceptatie ook afhankelijk van emoties, zoals angst voor het (nog) onbekende nieuwe systeem of weerstand tegen een afstandelijk project waarvan men de indruk heeft dat deze de (E2E-)processen niet begrijpt.

De uiteindelijke acceptatietest levert daarom nogal eens onaangename verrassingen op aan het eind van een project. Gebruikers komen met nieuwe of andere eisen en hebben weerstand tegen de veranderingen die door het project worden ingebracht. Continue en emotionele acceptatie kunnen deze onbeheersbaarheid en weerstand wegnemen.

#### 11.1.6 Testen in productie

Testen in productie wordt steeds meer een geaccepteerd verschijnsel. De praktijk is namelijk dat niet alles voor inproductiename getest kan worden. De beperkte doorlooptijd van cycli waarmee wijzigingen in productie moeten gaan, vergroten de noodzaak van testen in productie. Soms kunnen bepaalde (E2E-) testgevallen niet in een testomgeving worden uitgevoerd. Testen in productie kan daarmee een noodzakelijk kwaad worden.

Er zijn twee methoden om te testen in productie:

- Het analyseren van fouten die optreden tijdens echt productiewerk.
- Het uitvoeren van testscenario's in productie.



In beide gevallen is de expertise van E2E-testers onmisbaar.

Bij het uitvoeren van testscenario's in een productieomgeving is het zaak deze te isoleren van klantgegevens.

Bij het analyseren van de correcte afhandeling van productiewerk kunnen E2E-testers ingezet worden om kritische E2E-processen of zelfs deelprocessen te analyseren en snel eventuele bugfixes te begeleiden. Varianten van geconstateerde probleemgebieden kunnen daarnaast parallel aan de productieomgeving in de E2E-testomgeving nagespeeld worden. Bugfixes moeten hier worden gehertest.

Testen in productie betekent voor testers dat hun inzet zich uitbreidt van projecten naar beheer. De verantwoordelijk voor de testen zal in de regel onder verantwoordelijkheid van de beheerafdeling worden uitgevoerd.

## 11.2 Testdata

Testdata is de grond waar een E2E-tester op loopt. Als de synchroniciteit (bijvoorbeeld wijzigingen die niet zijn verwerkt in alle systemen), kwaliteit of volledigheid van de testdata niet voldoet dan kan dit grote gevolgen hebben voor de testvoortgang, de uitvoerbaarheid en uiteindelijk de waarde van de testen! Denk hierbij aan onterechte foutmeldingen vanwege niet synchrone data, ontbrekende data waardoor testgevallen niet uitgevoerd kunnen worden of onterechte bevindingen op slechte testdata.

Voor het beheer van testdata zal nauwe samenwerking gezocht moeten worden met databasebeheerders en technisch en functioneel beheer. Testers moeten in staat zijn het beheer van testdata aan te sturen, te begeleiden en te beoordelen.

In de volgende paragrafen wordt ingegaan op de aspecten waarmee rekening gehouden dient te worden als het gaat om testdata.

### 11.2.1 Datasynchroniciteit en databeheer

Om een testgeval uit te voeren over alle systemen heen, moet de data in de systemen op elkaar afgestemd zijn. Verwijzingen, identificerende velden, datumvelden, soms zelfs archiveringslogs op de achtergrond in databases, mogen elkaar niet tegenspreken en behoren naar de juiste gegevens te verwijzen.

Vaak wordt er initieel een dataset door de hele keten gemaakt of uit productie opgehaald op een zelfde moment en stand. Deze dataset is dan synchroon of wordt synchroon gemaakt. Maar wat gebeurt er met deze synchroniciteit indien er andere versies worden geïnstalleerd, er door de tijd wordt gereisd, databases opnieuw worden geïnstalleerd of op één systeem bepaalde testen worden uitgevoerd die weer van invloed zijn op testen die van een andere kant de E2E-keten raken? Het kan zelfs voorkomen dat applicaties niet meer draaien als er geen volledige datasynchroniciteit is. Het beheer van testdata is daarom van groot belang.

De volgende lijst kan gebruikt worden als een checklist testdatabeheer:

- Wie hebben er allemaal toegang nodig tot welke omgeving, wat voor type toegang is dit (welk autorisatieniveau) en welke daadwerkelijke toegang is er actief?
- Wie mag welke data wanneer gebruiken?
- Hoe worden relevante wijzigingen op de omgevingen uitgevoerd (voor wat betreft nieuwe versies, datasets, tijdreizen, database acties, achtergrondprogramma's, instellingen)?
- Welke datasets zijn nodig en onder welke voorwaarden worden deze gebruikt, opgeslagen en ververs (productiedata, een bepaalde draaidatum, laboratoriumdata en dergelijke)?
- Wie is verantwoordelijk voor de diverse beheertaken (zoals het draaien van batches, het instellen van parameters)?
- Hoe worden wijzigingen in de testdata beheerd?
- Kan worden geborgd dat testdata behorende bij één testgeval niet wordt gebruikt vanuit een ander testgeval?
- Kunnen wijzigingen in testdata getraceerd worden naar testgevallen en testers?

Er is daarom een belangrijke taak weggelegd voor de E2E-testmanager om te zorgen dat alle belanghebbenden de beschikking krijgen over juiste, volledige en bruikbare testdata en dat men elkaars testdata niet onbruikbaar maakt. Testdatabeheer en omgevingen beheer moeten middels het detaildraaiboek testomgevingen worden ondersteund. Voor een voorbeeld van een draaiboek testomgevingen, [zie pagina 121](#) van dit e-book.

### 11.2.2 Tijdreizen

Bij bepaalde testgevallen moet op een bepaald moment in het verleden, het heden of de toekomst gewerkt worden. Voorbeelden hiervan zijn een financiële jaarafsluiting of het creëren van specifieke historie, die alleen door het zelf aanmaken verkregen kan worden. Al tijdens het opstellen van testgevallen moet worden nagedacht of er tijdreizen nodig zijn.

Als een aantal testgevallen in de toekomst en andere weer in het verleden moeten worden uitgevoerd en de omgeving hiervoor moeilijk te configureren is, moet over alternatieven worden nagedacht, zoals: andere testomgevingen opzetten, het op andere meer flexibele omgevingen uitvoeren (zoals de systeemtestomgeving) of het uitstellen van testgevallen. Het kan voorkomen dat niet aan alle eisen kan worden voldaan en dan zal naar een gulden middenweg moeten worden gezocht.

Zoals zo vaak in het testvak geven risico's dan de doorslag. Wat is bijvoorbeeld het risico als we het ene geval wel en het andere geval niet of later uitvoeren? Wat zijn deze risico's voor het project (voortgang, stabiliteit van omgevingen, doorlooptijd en dergelijke) of het product (faalkans en impact van het betreffende testgeval)? Het mag niet zo zijn dat een minder risicovol testgeval een geheel testtraject ophoudt. In ieder geval is het van belang om dit soort dilemma's vroeg te onderkennen.

De volgende checklist moet worden doorlopen voor het bepalen en opzetten van tijdreizen:

- Welke testgevallen moeten tijdreizen en welke tijdsmomenten zijn er (is er een groepering van testgevallen in tijdsmomenten/ periodes)?
- Kunnen alle systemen in de E2E-keten wel op dezelfde manier tijdreizen?
- Wat gebeurt er als het ene systeem wel en de ander niet door de tijd reist?
- Hoe moet tijdreizen worden geïnitieerd? Wie gaat dat doen en wie moet daar van weten? Moet de serverdatum worden aangepast, de applicatiedatum, de databasedatum of een combinatie?
- Kan het resultaat van de testen zijn beïnvloed door onvolkomenheden in het tijdreizen, zoals het wijzigen van een draaidatum in een systeem terwijl op de server nog steeds de reële datum is ingesteld?
- Wat betekent het tijdreizen voor de doorlooptijd van de testen?
- Is het mogelijk en noodzakelijk om tijdreizen te verspreiden over testronden of testomgevingen om doorlooptijd te winnen?
- Zijn er andere negatieve effecten mogelijk van tijdreizen (zoals het niet meer werken van accounts en licenties)?

### 11.2.3 Productiedata, anonimiseren of laboratoriumdata

Bij gebruik van productiedata zijn beschikbaarheid, historie en synchroniciteit beter gewaarborgd. Denk aan de moeite, kennis en tijd die nodig is om synchrone data door de keten heen aan te maken! En toch is het vaak een probleem om deze data te gebruiken: privacyregels en beveiligingsregels, soms afgedwongen door wetgeving en zware sancties, verhinderen dan het inzetten van productiedata. In sommige gevallen kan dit worden ondergaan door alleen productiewerknemers te laten testen met productiedata of de data te anonimiseren.

Een ander alternatief zou kunnen zijn om testers die werken op de testomgeving dezelfde screening en voorwaarden als productiemedewerkers te laten ondergaan. Een en ander zal dan moeten worden afgestemd met de betreffende autoriteiten binnen de organisatie.

Er kunnen ook testtechnische redenen zijn om laboratoriumdata (= niet in productie aangemaakte data) te gebruiken:

- Er is minder afhankelijkheid van de aanwezigheid van specifieke gevallen in de productiedata, men hoeft deze niet te zoeken en kan met meer flexibiliteit uitgangssituaties aanmaken.
- Aanmaken van data kan een soort regressietest worden en kan testers informatie verschaffen over hoe de applicatie werkt.
- Het automatiseren van de aanmaak van testdata kan een goede eerste stap zijn op weg naar automatisering van de testuitvoering.
- Met zelf aangemaakte data kan worden uitgesloten dat productiedata tot een verstoring leidt. Argumenten hiertegen zijn dat hiermee dus ook geen fouten in de productiedata gevonden worden of dat het aanmaken van de data zelf fouten kan veroorzaken. Men kan daarom ook overwegen om in een eerste testronde met zelf aangemaakte testdata te werken en in latere testronden met productiedata.

Anonimiseren kan een uitkomst bieden. Hierbij wordt productiedata zo versleuteld dat er geen relatie meer te leggen is met daadwerkelijke personen en organisaties. Hiervoor zijn tools op de markt waarvan enkelen zelfs claimen de versleutelde data door een keten van systemen heen synchron te kunnen houden.

Het zelf aanmaken van testdata in een complexe keten vergt kennis en vaardigheid. Sommige zaken (zoals ingewikkelde historie) zijn moeilijk aan te maken, zeker als deze ook nog synchron moeten zijn met dezelfde historie in andere systemen in de keten. Ontwerpers of eindgebruikers kunnen vaak helpen bij het aanmaken van data. Bovendien is het aan te raden om deze gegevens op te slaan zodat ze opnieuw kunnen worden geladen.

Vanuit ervaringen met E2E-testen kan gesteld worden dat productiedata gebruikt moeten worden tenzij het niet anders kan.

## 11.3 Tools

E2E-testers maken, net als elke andere tester, bij vrijwel elke activiteit gebruik van tools. In deze paragraaf wordt ingegaan op een aantal aandachtspunten die specifiek voor E2E-testen gelden.

### 11.3.1 Bevindingenadministratie over de keten heen

In een systeemoverstijgend traject wisselen bevindingen vaak van verantwoordelijke. Wat initieel een bevinding in systeem A lijkt te zijn, moet later toch in systeem B worden opgelost. Iedereen dient dus toegang te hebben tot de complete verzameling bevindingen. In een complexe omgeving met meerdere systemen, teams en organisaties is het echter moeilijk om iedereen uniform en met dezelfde tools te laten werken. Dit geldt ook voor de bevindingenadministratie. Een vaak gehoorde kreet is: "wij werken nu eenmaal in bevindingen-tool X". Maar als elke leverancier en afdeling een afwijkende bevindingenadministratie gebruikt dan wordt de integrale bevindingenadministratie zelf een E2E-keten met project- en productrisico's. Testers worden dan vaak verantwoordelijk voor het actueel houden van gegevens en krijgen dan een dagtaak aan het bijhouden en overzetten van bevindingen in de diverse systemen, bij de diverse teams. In het ergste geval kunnen bevindingen verdwijnen uit administraties, blokkeert het E2E-proces en worden bevindingen niet, te laat of niet goed opgelost.

Elk team (bouw en test) moet daarom zelf verantwoordelijk gehouden worden voor de aansluiting op een centrale bevindingenadministratie. Bij uitbesteding zal in een vroeg stadium moeten worden afgedwongen dat iedereen in dezelfde (door de klant) voorgeschreven tools werkt.

Er zijn tegenwoordig toegankelijke, online, gratis tools beschikbaar, zodat kosten en toegankelijkheid geen excuus meer kunnen zijn. Voorkom dat testers een dagtaak krijgen in het overtuigen van bevindingen en hun mutaties in diverse administraties.

### 11.3.2 Simuleren

Het begrip “End to End” suggereert een complete keten van systemen. E2E-testen impliceert de intentie om in een zo volledig mogelijke omgeving te testen. Maar moet dit ten koste van alles worden afgedwongen? Systemen zijn vaak niet allemaal op het zelfde moment beschikbaar en er kunnen goede redenen zijn (kosten, inspanning, doorlooptijd, risico's) om toch delen van de keten te simuleren (met behulp van stubs en drivers).

Het advies is om snel, vanuit de inventarisatie van de E2E-processen, in kaart te brengen welke delen van de infrastructuur op welk moment beschikbaar zijn. Op basis daarvan moet de keuze worden onderbouwd om eventueel simulatie op te zetten.

Hier kan worden geprofiteerd van het werk dat reeds bij andere teams is gedaan: om interfaces te kunnen testen zullen ontwikkelteams en systeemtesters ook al simulatietools hebben ingezet.

- Inventariseer noodzaak tot simuleren. Bedenk het effect van simuleren op de waarde van de testen.
- Onderzoek het gebruik van tools door andere teams.
- Voorafgaand aan testuitvoering moeten de tools en bijbehorende inrichting en vaardigheden zijn verkregen.
- Hou het simpel: vaak is een teksteditor, een spreadsheet of een simpele html- of xml-editor al voldoende.

### 11.3.3 Beoordelen en manipuleren

Als het gaat om het beoordelen van tussenresultaten, het aanmaken of manipuleren van specifieke invoer zoals bestanden uit andere systemen, kan het nodig zijn om specifieke tools in te zetten. Een E2E-tester wordt geacht hier in thuis te zijn en zal, soms met hulp van bijvoorbeeld programmeurs of systeemtesters, in staat zijn om deze taken met beschikbare ontwikkeltools uit te voeren. In de meeste gevallen is dat afdoende.

De invloed van manipulatie op de waarde van de testen moet wel worden afgewogen. Indien data wordt gewijzigd, bijvoorbeeld een ingangsdatum wordt in het verleden gezet, moet worden onderzocht in hoeverre hiermee het gedrag van het systeem kan zijn beïnvloed.

### 11.3.4 Testautomatisering

Er zijn grote ontwikkelingen op het gebied van testautomatisering en deze vergroten de toepasbaarheid in integratietesten en E2E-testen. Testers zijn steeds beter opgeleid in het gebruik van deze tools. De groeiende noodzaak van integratie- en E2E-testen en vooral ook het toenemende gebruik van deze testen als regressietest van parallele wijzigingen (continue integratie) zorgen ervoor dat de business case voor testautomatisering in een E2E-test steeds positiever wordt.

Voor het succesvol inzetten van testautomatisering in een E2E-test moet het volgende worden overwogen:

- Ingewikkelde testscenario's zoals in een E2E-testen zijn moeilijker geautomatiseerd te testen dan meer lokale testgevallen. Met name controles en beoordelen van proceskenmerken en het uitvoeren van meerdere handelingen in meerdere systemen binnen één scenario zijn lastig te automatiseren.
- Testautomatisering is vaak meer gebaat bij stabiele, voorspelbare omgevingen en synchrone, samenhangende en identieke testdata dan handmatig testen.
- Voor het succesvol inzetten van testautomatisering is een gestructureerd opgezette testset randvoorwaardelijk.
- Testautomatisering begint met kleine en meer haalbare doelen. Complete testclusters in één keer volledig automatiseren zal niet lukken. Wat te denken van het in eerste instantie gebruiken van automatiseringstools voor het aanmaken van testdata of performancetesten? Van daaruit kan de scope van testautomatisering worden uitgebreid.
- Het opzetten van testautomatisering in een E2E-test kan betekenen dat een keten van testautomatiseringsstappen en tools ingezet gaan worden. Een hele suite van applicaties die nodig is om de testen uit te voeren, is in zichzelf weer een keten met benodigd beheer, kennis, ondersteuning, licenties, eigenaarschap en dergelijke.

# Deel 12

Checklists en  
voorbeelden  
E2E-testen

## Deel 12: Checklists en voorbeelden E2E testen

Dit laatste deel van de praktijkgerichte aanpak voor E2E-testen bestaat uit een verzameling checklists en voorbeelden van een logisch testgeval, een fysiek testgeval en een testdraaiboek. Een deel daarvan is al eerder getoond in vorige delen.

Checklist of voorbeeld	Deel/ locatie
<b>Voordelen van een E2E-test</b>	Niet in ander deel vermeld
<b>Documenten voor E2E-inventarisatie</b>	Niet in ander deel vermeld
<b>Uitgangspunten E2E-productrisicoanalyse</b>	6 E2E-teststrategie
<b>Vaststellen acceptatiecriteria</b>	Niet in ander deel vermeld
<b>Entry- en exitcriteria</b>	Niet in ander deel vermeld
<b>Randvoorwaarden E2E-test</b>	Niet in ander deel vermeld
<b>Vaststellen testomgeving</b>	11 E2E-testinfrastructuur/Testomgevingen/Welke testomgeving?
<b>Testdatabeheer</b>	11 E2E-testinfrastructuur/Testdata
<b>Tijdreizen</b>	11 E2E-testinfrastructuur/Tijdreizen
<b>Testdraaiboek</b>	Niet in ander deel vermeld
<b>Profiel van een E2E-tester</b>	10 E2E-testorganisatie/De onderdelen van E2E-testorganisatie
<b>Integratietesten en E2E-testen in testsoorten</b>	10 E2E-testorganisatie/Schaling E2E-testen
<b>Afwegingen bij invulling van de E2E-test</b>	10 E2E-testorganisatie/Schaling E2E-testen
<b>Intake Testomgeving</b>	Niet in ander deel vermeld
<b>Bevindingenbeheer</b>	Niet in ander deel vermeld
<b>Voorbeeld van een logisch testgeval</b>	9 Testontwerp
<b>Voorbeeld van een fysiek testgeval</b>	9 Testontwerp
<b>Voorbeeld van een testdraaiboek</b>	11 E2E-testinfrastructuur

### 12.1 Checklists

#### 12.1.1 Voordelen van een E2E-test

Indien de organisatie nog moet worden overtuigd van het belang van een E2E-test kunnen de volgende argumenten in overweging worden genomen:

- Beperken van E2E-risico's in productie: meer vertrouwen in de kwaliteit van de gehele systeem keten en de stabiliteit van bedrijfsprocessen.
- Beperken van implementatieproblemen door het naspelen van productiegelijke scenario's met beheerders en gebruikers.
- Vroege acceptatie van onderdelen door betrokkenheid van gebruikers en beheerders
- Gedeeltelijk inzetten van centrale medewerkers in de zogenaamde E2E-board, waardoor deze kennis van het nieuwe systeem en de E2E-processen opbouwen en tegelijk ook productiewerk kunnen blijven doen.
- Samenwerking met systeemtest, systeemintegratietest en acceptatietest levert besparingen in deze testsoorten op.
- In geval van complex releasemanagement kan de E2E-test gezien worden als een globale regressietest waarmee de allergrootste risico's van alle betrokken projecten kunnen worden gedekt.
- Een reguliere, projectoverschrijdende, E2E-test levert een testset en expertise op die door meerdere partijen is te gebruiken in toekomstige projecten.

### 12.1.2 Documenten voor de E2E-inventarisatie

De volgende documenten komen in aanmerking om te worden meegenomen in de inventarisatie:

- Procesbeschrijvingen.
- Werkinstructies.
- Systeembeschrijvingen op hoofdlijnen.
- Globaal ontwerp.
- Systeemplaten.
- Requirements.
- Use Cases.
- Functionele ontwerpen.
- Technische ontwerpen.
- Interfacebeschrijvingen.

### 12.1.3 Uitgangspunten E2E-productrisicoanalyse

De volgende uitgangspunten gelden voor een (E2E) productrisicoanalyse:

- Bereid een E2E-PRA goed voor: een E2E-inventarisatie is een goede input voor de risicoanalyse.
- De E2E-testmanager schat vooraf in wat de hoge risico's zijn en wat voor type risico's dit zijn (in termen van kwaliteitsattributen): is efficiency bijvoorbeeld een hoog risico? Gebruik voor deze inschatting de checklist E2E-risico's.
- Een E2E-PRA kan niet in één bijeenkomst worden afgedekt. Deel de E2E-keten op in verschillende gebieden (bijvoorbeeld per hoofdproces of samenhangende hoofdprocessen), maar waak ervoor dat dit niet tientallen gebieden worden. Per gebied moeten de risico's worden doorgesproken in korte bijeenkomsten (maximaal 2 uur) of individuele gesprekken.
- Bespreek de risico's in termen van echte scenario's. Dus niet alleen maar in termen van een abstract kwaliteitsattribuut ("Efficiency is een hoog risico") maar in termen van welke concrete gebeurtenis of omstandigheid tot welk concreet ongewenst effect kan leiden. Bijvoorbeeld: "als de indexering op de database niet goed werkt, kan een aanvraag van het product leiden tot een wachttijd van meerdere minuten voor de klant. De klant kan dan besluiten bij de concurrent een product te kopen".
- Maak er geen wetenschap van, met wiskundige berekeningen tot achter de komma. Het is voldoende om te onderkennen wat de hoogste risico's (veel testandacht) en de lagere risico's (weinig testandacht) zijn.
- Laat zien wat er met de risico's gebeurt: laat ze terugkomen in plannen, voortgangsrapporten en eindrapporten. Als er niets met risico's wordt gedaan, zal men in de toekomst minder geneigd zijn deel te nemen aan risicoanalyses.

### 12.1.4 Vaststellen acceptatiecriteria

De risicoanalyse en de daaruit volgende aanpak per testcluster geven een goede indruk van de uit te voeren activiteiten. De uit te voeren testgevallen zijn eerste acceptatiecriteria. De acceptatiecriteria moeten verder nog worden uitgewerkt. De volgende punten kunnen daarbij als checklist dienen:

- Acceptanten: per E2E-proces en deel van de architectuur moet worden vastgesteld wie hier uiteindelijk een handtekening onder moeten zetten. Dit zijn in de regel de acceptanten zoals die tijdens de fase Initiatie al zijn vastgesteld. Tijdens de planningsfase moet met deze acceptanten worden vastgesteld onder welke voorwaarden zij tot acceptatie overgaan.
- In hoeverre is de uitvoering van de testgevallen voldoende? Wil men bijvoorbeeld ook nog de werking van achtergrondprogramma's gedurende een bepaalde periode zien? Welke niet-functionele acceptatiecriteria zijn er verder nog (zoals gebruikersvriendelijkheid)? Hoe kunnen deze worden gecontroleerd?
- Welke testgevallen moeten wel en welke hoeven niet te worden uitgevoerd?
- Van wie verlangen de acceptanten een aanvullend oordeel (bijvoorbeeld een audit of een oordeel van een bepaalde medewerker)?
- Hoe moeten testresultaten worden vastgelegd en gepresenteerd?
- Hoe vaak moeten testgevallen zijn uitgevoerd?
- Zijn alle aspecten gedekt?



- Hoeveel fouttolerantie wordt bij in productiename geaccepteerd?
- Welk type fouten zijn wel en welke absoluut niet acceptabel?
- In hoeverre moet een complete hertest of regressietest worden uitgevoerd op de laatste versies van alle systemen in de E2E-keten?

### 12.1.5 Entry-criteria

Entry-criteria kunnen verschillen per testcluster, zeker in een grote E2E-keten.

De volgende zaken zijn van belang:

- Vereiste status per systeemtest, interfacetest of zelfs systeemintegratietest, en hoe hiervan bewijs wordt geleverd (testrapporten, testverslagen, output).
- Beschikbaarheid van de testomgeving.
- Status van een uit te voeren intake op de testomgeving.
- Oplevering van testdata door de E2E-keten heen.
- Beschikbaarheid van ondersteuning door beheerders van betreffende systemen.

Als wordt besloten tot de start van de test terwijl niet alle entrycriteria zijn gehaald, dan moeten de risico's en de impact op de planning worden aangegeven. Typische testprojectrisico's zijn dan: verminderde waarde van testresultaten, het later opnieuw moeten uitvoeren van testgevallen, het vastlopen van E2E-processen, uitloop, het niet meer beschikbaar hebben van testpersoneel, et cetera.

### 12.1.6 Exit-criteria

Exitcriteria vallen voor een groot deel samen met de acceptatiecriteria, maar moeten worden aangevuld met producten die door het E2E-testteam moeten worden opgeleverd:

- Regressietestset.
- Testgevallen en testresultaten.
- Testgegevenssets.
- Testrapport.
- Overdracht aan beheerorganisatie.

### 12.1.7 Randvoorwaarden E2E-test

De E2E-test is een samenhang van E2E-processen, systemen, diverse groepen mensen uit meerdere organisaties en het uitvoeren van ingewikkelde testgevallen over meerdere dagen. Er zijn veel afhankelijkheden en daarom zijn uitloop en onbeheersbaarheid reële risico's. Het vaststellen van de randvoorwaarden om een E2E-test beheersbaar te houden en deze randvoorwaarden afstemmen met de betreffende partijen is daarmee van groot belang.

Generieke randvoorwaarden voor E2E-testen zijn:

- Omgevingenbeheer: voldoende beschikbaarheid van beheerders, betrokkenheid bij E2E-traject. Afgestemde opleverdatums van de testomgevingen van alle betrokken systemen.
- Configuratiebeheer: centrale versiecontroles over alle omgevingen, afgestemde criteria en beslissingsbevoegdheden bij nieuwe installaties of data.
- Bemensing: voldoende beschikbaarheid van testers (bijvoorbeeld vanuit systeemtesten, gebruikersafdelingen en beheer), gebruikers en beheerders.
- Kwaliteit: exitcriteria van ontwerpfasen, systeemtesten en systeemintegratietesten bij oplevering aan E2E-test. Oplevering lijsten met openstaande bevindingen en uitgevoerde testgevallen.
- Bevindingenbeheer: communicatie, eenduidigheid van begrippen zoals prioriteit, impact en status, escalatiekanaal en beslissingsbevoegdheid bij onenigheden. Centrale bevindingenadministratie over de E2E-keten heen.
- Communicatie: directe communicatie met teamleden (dit kan nog wel eens een issue zijn in geval van uitbesteding) en beschikbaarheid van mensen voor vragen en uitvoering van activiteiten die randvoorwaarde zijn voor een succesvolle E2E test. E2E-testteam is aangesloten op berichtgeving en rapportage over status van systemen.

- Goodwill van alle projectleden. Dit kan in de praktijk een grote belemmering zijn. Zorg voor periodiek direct contact tussen de diverse teams.

### 12.1.8 Vaststellen testomgeving

Bij het kiezen van de juiste omgevingen spelen de volgende afwegingen:

- In welke mate moet de testomgeving productiegelijk zijn? Productiegelijkheid geeft de beste, want representatieve, resultaten. Productiegegevens gebruiken is ook een efficiënte manier om synchroniciteit van testgegevens door de gehele keten te verkrijgen. Productiegelijkheid kan echter ook duur zijn (vanwege de eisen die worden gesteld aan de hardware) of risicovol (vanwege de beveiligbaarheid van klantgegevens) of onmogelijk (vanwege wet- en regelgeving rond gebruik van productiedata).
- Het aantal testomgevingen: kunnen systeemtestomgevingen worden opgenomen in integratietestomgevingen en E2E-testomgevingen? Moet de E2E-omgeving dezelfde omgeving zijn waar ook de acceptatietesten op plaatsvinden? Welke testclusters hebben een eigen testomgeving nodig?
- Releasemanagement van de testomgevingen. Omdat E2E-testen vaak arbeidsintensief zijn en een lange doorlooptijd hebben, zal er behoefte aan stabiliteit zijn. Daarom moet men voorzichtig zijn met het doorzetten van wijzigingen in deelsystemen naar de E2E-testomgeving. Kleine haperingen kunnen lange en zware testen verstoren waardoor er uitloop ontstaat. Dit is dan een argument voor een scheiding tussen systeemtest- en E2E-testomgevingen.
- Indien de E2E-testomgeving ook wordt ingezet als acceptatietestomgeving kunnen gebruikers (in de acceptatietest) worden blootgesteld aan nog deels ongeteste software (voor wat betreft E2E-aspecten). Hiertegen bestaat nog steeds huiver. Het is onze ervaring dat gebruikers, mits nauw betrokken, hier geen enkel probleem mee hebben. Het heeft daarnaast voordelen om een acceptatietest parallel of in samenwerking met een E2E-test uit te voeren. De kennis van gebruikers kan in een E2E-test van belang zijn en gebruikers kunnen in een E2E-test vroeger acceptatiecriteria valideren. Er kan dan doorlooptijd worden gewonnen.
- De scheiding tussen de testomgevingen van de E2E-test en de acceptatietest kan echter onvermijdelijk zijn. Als gebruik van productiedata een must is voor de acceptatietest en alleen productiegebruikers daarmee mogen werken dan kunnen externe medewerkers of niet-gebruikers niet op dezelfde omgeving werken.

### 12.1.9 Testdatabeheer

De volgende lijst kan gebruikt worden als een checklist voor testdatabeheer:

- Wie hebben er allemaal toegang nodig tot welke omgeving, wat voor type toegang is dit (welk autorisatieniveau) en welke daadwerkelijke toegang is er actief?
- Wie mag welke data wanneer gebruiken?
- Hoe worden relevante wijzigingen op de omgevingen uitgevoerd (voor wat betreft nieuwe versies, datasets, tijdreizen, database acties, achtergrondprogramma's, instellingen)?
- Welke datasets zijn nodig en onder welke voorwaarden worden deze gebruikt, opgeslagen en ververst (productiedata, een bepaalde draaidatum, laboratoriumdata en dergelijke)?
- Wie is verantwoordelijk voor de diverse beheertaken (zoals het draaien van batches, het instellen van parameters)?
- Hoe worden wijzigingen in de testdata beheerd?
- Kan worden geborgd dat testdata behorende bij één testgeval niet wordt gebruikt vanuit een ander testgeval?
- Kunnen wijzigingen in testdata getraceerd worden naar testgevallen en testers?

### 12.1.10 Tijdreizen

De volgende checklist moet worden doorlopen voor het bepalen en opzetten van tijdreizen:

- Welke testgevallen moeten tijdreizen en welke tijdstipmomenten zijn er (is er een groepering van testgevallen in tijdstipmomenten/ periodes)?
- Kunnen alle systemen in de E2E-keten wel op dezelfde manier tijdreizen?
- Wat gebeurt er als het ene systeem wel en de ander niet door de tijd reist?
- Hoe moet tijdreizen worden geïnitieerd? Wie gaat dat doen en wie moet daar van weten? Moet de serverdatum worden aangepast, de applicatiedatum, de databasedatum of een combinatie?

- Kan het resultaat van de testen zijn beïnvloed door onvolkomenheden in het tijdreizen, zoals het wijzigen van een draaidatum in een systeem terwijl op de server nog steeds de reële datum is ingesteld?
- Wat betekent het tijdreizen voor de doorlooptijd van de testen?
- Is het mogelijk en noodzakelijk om tijdreizen te verspreiden over testronden of testomgevingen om doorlooptijd te winnen?
- Zijn er andere negatieve effecten mogelijk van tijdreizen (zoals het niet meer werken van accounts en licenties)?

### 12.1.11 Testdraaiboek

In het draaiboek moeten de volgende zaken worden overwogen:

- Projectmijlpalen die voor het testtraject van belang zijn, zoals "Systeemtest A gereed".
- Oplevering van testdata.
- Opleveringen van de diverse systemen.
- Beheerhandelingen die een systeem en de data gebruiksklaar maken. Laden van een database, eventueel draaien van bepaalde programma's, vullen van referentietabellen, inlezen van bestanden, draaien conversieprogrammatuur, draaien batches, et cetera.
- De dagelijks uit te voeren beheerhandelingen, zoals het draaien van programma's waarmee output en input wordt gegenereerd of verwerkt. Deze zaken moeten zo productiegelijk (ook in volgorde en moment!) worden opgezet. Indien er goede redenen zijn om hiervan af te wijken moet dit worden afgestemd met beheerders en acceptanten.
- De uit te voeren testactiviteiten, zoals invoer, mutaties, controles met verwijzing naar de betreffende testgevallen of groepen testgevallen. Het is niet de bedoeling dat in het draaiboek complete fysieke testgevallen komen te staan maar wel informatie als: "controleren van output richting Belastingdienst voor wat betreft testgeval 1,5,7".
- Opleveringen van bugfix-releases (voor zover die in te plannen zijn).
- Specifieke beheerhandelingen zoals het maken of het terugzetten van een bepaalde back-up.
- Per activiteit moeten de verantwoordelijken, de volgorde, de afhankelijkheden en de status (% gereed) inzichtelijk zijn.
- In het draaiboek moeten de verschillende testronden tot uitdrukking komen. Testronden hebben vaak een verschillende lengte (ronde 1 is vaak een soort intake ronde, ronde 2 een volledige ronde en ronde 3 en 4 zijn vaak hertest rondes).
- Afsluitende activiteiten en mijlpalen, zoals de GO/No GO beslissing.

### 12.1.12 Profiel van een E2E-tester

Een typische E2E-tester is bij uitstek een allrounder en geen specialist. Dit komt in de volgende lijst met vaardigheden tot uitdrukking:

- Structuur van de E2E-keten: in staat zijn met verschillende systemen en systeemonderdelen te werken, zoals interfaces, schermen, achtergrondprogramma's.
- Doel van de E2E-keten: snel begrip krijgen van E2E-processen en producten.
- Met verschillende type mensen kunnen werken, communiceren, deze begrijpen en overtuigen: gebruikers, beheerders, ontwerpers, programmeurs, projectleiders, managers.
- Ervaring met front- en backend systemen.
- Overzicht houden in een complex van werkzaamheden en statussen van diverse onderdelen.
- Effectief kunnen communiceren (de vraag: "hoe gaat het" niet beantwoorden met een waslijst aan bevindingen en testgevallen, maar met het oog op acceptatiecriteria en risico's in heldere termen per vitaal onderdeel de status kunnen aangeven en onderbouwen).
- Database ervaring (in verband met vinden van testdata en controles van tussenresultaten).
- Logbestanden en interfacebestanden kunnen lezen.

### 12.1.13 Integratie- en E2E-testen in andere testsoorten

Hieronder volgt een korte beschrijving van testsoorten en de mate waarin deze integratie- en E2E-aspecten kunnen dekken:

- **Systeemtest:** integratie van systeemonderdelen/componenten. Functionele test waarin deze onderdelen op hun onderlinge samenwerken worden beoordeeld.
- **Interfacetest:** test van de connecties tussen systemen. Is vaak een onderdeel van een systeemtest of als een aparte systeemtest opgezet (van de interface of het cluster van interfaces, bijvoorbeeld als dit als middle ware is opgezet). De testbasis bestaat uit functionele en technische ontwerpen van de interfaces.
- **Systeemintegratietest:** test van een functie die systeemoverschrijdend is. In feite wordt een cluster van systemen en interfaces als één systeem beschouwd en wordt deze functioneel getest. De testbasis bestaat uit globale ontwerpen, functionele ontwerpen en/of niet-functionele eisen.
- **Project E2E-test:** test waarbij de infrastructuur vanuit het blikveld van het project wordt onderworpen aan een procestest, waarbij men over de grenzen van de infrastructuur heen kijkt. Wie of wat wil iets en hoe wordt dit bereikt middels de infrastructuur? De testbasis bestaat uit een verzameling (de E2E-inventarisatie) van procesbeschrijvingen, interfacebeschrijvingen en architectuurplaten.
- **Projectoverschrijdende E2E-test:** E2E-test in het geval er meerdere projecten tegelijk wijzigingen op dezelfde infrastructuur uitvoeren en deze projecten elkaar kunnen beïnvloeden. Deze test is een gezamenlijke inspanning van deze projecten.
- **Organisatieoverschrijdende E2E-test:** E2E-test in het geval meerdere organisaties gevolgen kunnen ondervinden van wijzigingen in de gezamenlijk gebruikte infrastructuur. Deze test is een gezamenlijke inspanning van de betrokken organisaties of een test door een externe organisatie die de E2E-keten beoordeelt.
- **Acceptatietesten:** testen door eindgebruikers (beheerder, medewerkers van helpdesks en dergelijke) om vast te stellen in hoeverre requirements zijn ingevuld en systemen de E2E-processen voldoende ondersteunen. De testbasis bestaat uit procesbeschrijvingen, werkinstructies en requirements. In een acceptatietest worden vaak interface-, integratie- en E2E-aspecten impliciet getest maar mist men de vaardigheden en kennis om dit diepgravend te doen.

### 12.1.14 Afwegingen bij de invulling van een E2E-test

Van de volgende zaken moeten de specifieke omstandigheden van een organisatie worden beschouwd:

- **Productrisico's.** Afhankelijk van de hoogte van de risico's en de mate waarin deze alleen zijn af te dekken door een E2E-test, varieert de noodzaak voor het opzetten van een E2E-test.
- **Grootte/cultuur van de organisatie.** Idealiter zijn een interfacetest, SIT en E2E-test ingericht als testtypen en geen afzonderlijke testsoorten. Deelactiviteiten en testclusters kunnen vaak grotendeels worden belegd bij systeemtesten en in nauwe samenwerking met acceptatietesten worden uitgevoerd.
- **Geografische verspreiding.** Het zich bevinden in een complexe en grote organisatie, een hoge mate van juridische en culturele formaliteit en scherpe afbakening tussen organisatieonderdelen zijn redenen om een SIT en E2E-test formeler in te steken. Activiteiten moeten dan in meer detail worden gepland en er moeten meer gedetailleerde afspraken vooraf worden gemaakt over uit te voeren werkzaamheden en inzet van medewerkers. Dit zal druk uitoefenen op de doorlooptijd. Het vergroten van het E2E-kernteam kan dan een voordeel opleveren. De afhankelijkheid van de medewerkers neemt dan namelijk af en voorbereidingen kunnen gedetailleerder worden uitgevoerd.
- **Grootte/complexiteit van het systeemlandschap.** Complexe systeemlandschappen zijn een reden om een E2E-testset als regressietest op te zetten. De E2E-test wordt dan ook een periodieke regressietest van alle wijzigingen en projecten in het systeemlandschap. Dit kan in het uiterste geval organisatieoverstijgend zijn, zoals een regressietest van interoperabiliteit tussen mobiele netwerk aanbieders. Organisatorisch kan hieruit volgen dat er een aparte E2E-afdeling opgezet wordt, waarin een permanent operationele testomgeving wordt beheerd en waar vanuit E2E-activiteiten worden gecoördineerd en uitgevoerd.
- **Projectmatige verspreiding.** Cloud Computing en uitbesteding zijn omstandigheden waarin projectactiviteiten zijn verspreid over organisaties. Beheer, ontwikkeling en testactiviteiten worden "elders" uitgevoerd. In de praktijk betekent dit vaak dat qua integratie en E2E-aspecten extra product- en productrisico's worden gelopen. Men heeft vaak minder zicht op de te verwachten kwaliteit. Ontwikkelpartijen hebben moeite om E2E-processen en daarmee de E2E-risico's te begrijpen en door middel van testen af te dekken.

Een SIT en E2E-test geregisseerd of zelfs uitgevoerd door de klantorganisatie wordt dan van groter belang. Dit kan een heikel punt zijn omdat één van de doelen van uitbesteding het afstoten van projectactiviteiten is. Uitbesteding kan echter betekenen dat juist een E2E-test zwaarder moet worden aangezet vanuit de eigen geledingen. De E2E-test komt dan vaak bij de acceptatietest, en dus de gebruikersorganisaties, te liggen, omdat zij de laatste "quality gate" naar productie zijn. Om een goede systeemintegratietest of E2E-test uit te voeren is de gebruikersacceptatietest echter vaak niet voldoende toegerust.

Afhankelijk van de bovenstaande analyse, moeten de volgende keuzes worden gemaakt voor wat betreft de invulling van interfacetesten, SIT en E2E-test:

### Organisatie

- Lijnorganisatie: hoe moeten de testen worden ondersteund? Moet er een aparte E2E-organisatie worden opgezet of worden de testen bemenst door de projecten zelf, afwisselend met projectmedewerkers, gebruikers, beheerders en aangevuld met externe testers?
- Moeten aparte teams voor IT, SIT en/of E2E-testen worden opgezet?
- Moeten specialisten worden ingehuurd, zoals E2E-testers, efficiency- en securitytesters?
- Grootte van het E2E-kernteam: in welke mate kan of moet het E2E-kernteam alle activiteiten uitvoeren en kunnen of moeten andere medewerkers worden betrokken?
- Is een aparte, project- of organisatieoverstijgende E2E-test nodig en mogelijk?

### Infrastructuur

- Zijn aparte testomgevingen per IT, SIT of E2E (of zelfs per testcluster) nodig en mogelijk?
- Is simulatie van systemen en systeemonderdelen door middel van tooling noodzakelijk en mogelijk? Hierdoor kunnen SIT en E2E-aspecten eventueel vroeg en goedkoop worden getest maar men moet zich de vraag stellen of E2E-risico's afdoende worden gedekt.
- Kunnen testactiviteiten worden geautomatiseerd? Dit vergt een initiële investering die moet worden terug verdiend. Daarnaast is er een organisatie (kennis, beheer) nodig om dit mogelijk te maken en in stand te houden.

### Fasering

- Aparte IT, SIT en E2E-test: zijn dit aparte teams of deelprojecten? Wat is de onderlinge afhankelijkheid: start de E2E-test pas als de SIT klaar is?
- Testtype of testsoort: worden de activiteiten belegd in aanwezige projectfasen (zoals de systeemtest en acceptatietest) of moeten hier aparte fasen en projectteams voor worden ingericht (SIT en E2E-test als testsoort)?
- Welke testtypen binnen SIT en E2E (zoals efficiency, security, failover) moeten worden opgezet? Hebben deze een eigen team, fasering en testomgeving?

#### 12.1.15 Intake testomgeving

Zodra delen van de omgeving beschikbaar zijn, start de testuitvoering. Als eerste moet er een intake test worden uitgevoerd. Het volgende wordt daarin vastgesteld:

- Of de juiste versies zijn geïnstalleerd.
- Of de juiste data is geïnstalleerd.
- Of de juiste stuurgegevens zijn ingericht (denk aan systeemdatum, vulling van referentietabellen).
- Of de juiste configuratie van toepassing is (omgevingsfactoren waardoor het systeem kan worden gebruikt zoals gepland, denk aan juiste schaling, gebruik van applicatiesoftware door andere testomgevingen).
- Of alle accounts aanwezig zijn en of deze de juiste toegang geven tot de systemen.
- Een minimale controle van de kwaliteit van de opgeleverde systemen (dit wordt ook wel een smoketest genoemd).
- Door middel van het uitvoeren van een happyflow vaststellen of de E2E-processen kunnen worden doorlopen.

### 12.1.16 Bevindingenbeheer

De volgende zaken verdienen aandacht:

- De bevindingenadministratie is een administratie en niet primair een communicatiemiddel. Indien dit laatste het geval is, is de kans groot dat de voortgang stagneert: medewerkers communiceren dan namelijk niet meer direct met elkaar.
- Stel een E2E-bevindingenbeheerder aan. Dit kan de E2E-testmanager zelf zijn, of een ander teamlid wanneer de werkzaamheden een dagtaak worden. De bevindingenbeheerder houdt de uitvoering van het bevindingenproces in de gaten en waakt over de betrouwbaarheid van de gegevens.
- Bevindingenbeheer begint met het volledig en duidelijk opschrijven van bevindingen. Bevindingen horen de volgende zaken eenduidig en uniform te beschrijven:
  - Omschrijving: korte logische omschrijving. Bijvoorbeeld: "na overlijden tijdens lopende mutatie wordt de mutatie gewoon uitgevoerd".
  - Uitgangssituatie: gebruikte data, testomgeving, dag, testgeval, stap in het testgeval waar vanuit de bevinding zich voor deed (bijvoorbeeld: "polis heeft een mutatie op looptijd met terugwerkende kracht gekregen, dagwerk is gedraaid").
  - Logische omschrijving van de test: het doel van het testgeval (bijvoorbeeld: "overlijden klant met lopende mutaties").
  - Handeling: in welk systeem wordt welke actie uitgevoerd (bijvoorbeeld: "in het klantsysteem voer ik bij datum overlijden dag vandaag in").
  - Verwachting: het verwachte resultaat. (bijvoorbeeld: "ik verwacht na dagwerk dat vanuit het klantsysteem, de polis in het polissysteem is geannuleerd en de mutatie ook niet is verwerkt").
  - Resultaat: de afwijking van het verwachte resultaat (bijvoorbeeld: "de mutatie staat nog open en wordt alsnog verwerkt. Resultaat: er vinden correctiebetalingen plaats").
  - Omgeving en systemen, eventueel aangevuld met loggingsbestanden, tussenrapporten en dergelijke.
- Voortgang in het oplossen: in een E2E-test vallen bevindingen gemakkelijk tussen wal en schip. De verantwoordelijkheid voor de bevindingen wijzigt omdat vaak niet meteen duidelijk is wat de oorzaak is en wie de bevinding moet oplossen. De bevindingenbeheerder bewaakt bevindingen en signaleert stagnaties aan de E2E-testmanager.
- De E2E-testmanager organiseert periodiek een bevindingenoverleg.
- De E2E-board fungeert als klankbord en functioneert als stuurgroep. Niet elke bevinding wordt met de E2E-board besproken, maar de leden kunnen een belangrijke rol spelen in de analyse, het prioriteren en de besluitvorming van bevindingen.

### 12.2 Voorbeeld logische testgevallen (data combinatie test)

In onderstaand voorbeeld zijn de testcondities (actoren, beslissende factoren, resultaten en kritische factoren) in rijen opgelijst. Elke kolom bevat een testgeval, beginnend met een happyflow. Per testgeval is de dekking van de condities aangegeven. Elke conditie moet ten minste één maal worden geraakt door een testgeval.

		aantal malen geraakt	Testgeval 1: happyflow	Testgeval 2: klantgegevens wijzigen	Testgeval 3: klantgegevens NOK	Testgeval 4: retourbetaling	Testgeval 5: BKR NOK	Testgeval 6: product niet meer geldig	Testgeval 7: product nog niet geldig	Testgeval 8: klant overlijdt vlak na afsluiten product	Bevindingen/ status	
<b>Actoren</b>		<b>Rol</b>										
1	Klant	vraagt aan, ontvangt mail, betaalt	8	x	x	x	x	x	x	x		
2	Gebruiker Polis/ Klantsysteem	Controle/ mutaties/ invoer	8	x	x	x	x	x	x	x		
3	Gebruiker Financiën	Financiële verwerking	3	x			x				x	
4	Beheer	Draaien achtergrondprogramma's	5	x	x	x	x				x	
5	BKR	Terugkoppelen kredietstatus	4	x			x	x			x	
6	Bank	Verwerken betaling	3	x			x				x	
7	Postorganisatie	Verwerken communicatie	5	x	x		x	x			x	
<b>Beslissende factoren (proces)</b>		<b>0</b>										
	<b>Systeem</b>	<b>Functie</b>										
1	Web	zoeken producten	8	x	x	x	x	x	x	x		
2	Productsysteem	leveren productgegevens	8	x	x	x	x	x	x	x		
3	Web	selecteren product	8	x	x	x	x	x	x	x		
4	Polisadministratie	controleren klantgegevens	8	x	x	x	x	x	x	x		
5	Klantsysteem	invoeren klantgegevens	7	x		x	x	x	x	x		
6	Klantsysteem	BKR check	8	x	x	x	x	x	x	x		
7	BKR	leveren BKR-status	8	x	x	x	x	x	x	x		
8	Polisadministratie	ophalen productgegevens	6	x	x		x		x	x		
9	Polissysteem	aanmaken polis	4	x	x		x				x	
10	Polissysteem	verzenden contract	4	x	x		x				x	
11	Postsysteem	aanmaken post	4	x	x		x				x	
12	Financieel systeem	aanmaken betalingen	4	x	x		x				x	
13	Bank	verwerken betalingen	4	x	x		x				x	
14	Polissysteem	Aanmaken werkljst: foutieve betalingen	1				x					
<b>Resultaten</b>		<b>Systeem</b>	<b>0</b>									
1	Contract verstuurd	post	4	x	x		x				x	
2	Betaling	bank	3	x	x						x	
3	Contract vastgelegd	polissysteem	4	x	x		x				x	
<b>Kritische factoren</b>		<b>Omschrijving</b>	<b>0</b>									
1	Maandovergang		1		X							
2	Overlijden		1								x bev 312	
<b>Testresultaten</b>		<b>OK</b>	<b>nok</b>									
1	testronde 1		0	2	nok	nok						
2	testronde 2		2	1	ok	nok	ok					
3	testronde 3		1	0		ok						
4	testronde 4		2	2				ok	nok	ok	nok	
5	testronde 5		3	0	ok				ok		ok	



## 12.3 Voorbeeld fysieke testgevallen

	Testgeval 1: happyflow	Testgeval 4: retourbetaling
<b>Testdata:</b>		
<b>Naam</b>	F. Le Boulanger	G. Beenhouwer
<b>Sofinummer</b>	122344544	654654654
<b>Comm. Wijze verz nemer</b>	Post	mail
<b>Ingangsdatum:</b>	1-1-2014	1-1-2014
<b>Prolongatiedatum:</b>	1-1-2015	1-1-2015
<b>Dekking</b>	Volledig	Middel
<b>Betaalmethode</b>	Incasso	Incasso
<b>iban</b>	023ABN022448679708	023ABN022448644535434
<b>speciaal/randvoorwaarden</b>	Klant bestaat nog niet, geen BKR registratie	Klant bestaat nog niet, geen BKR registratie
<b>Testuitvoering</b>		
<b>Acties dag 1</b>	21-6-2014	
Klant/ web portaal	Klant voert zichzelf als nieuwe klant op; nieuwe polis	Klant voert zichzelf als nieuwe klant op; nieuwe polis
Polissysteem		
Klant systeem		
BKR		
Bank		
Post		
<b>Controles dag 1</b>	21-6-2014	
Klant/ web portaal		
Polissysteem	Polis met status "voorlopig"	Polis met status "voorlopig"
Klant systeem	Klant geregistreerd	Klant geregistreerd
Financieel systeem		
BKR		
Bank		
Post		
<b>Batchproces 1&gt;2</b>	<b>Beheer voert per systeem dagelijkse programma's uit</b>	<b>Beheer voert per systeem dagelijkse programma's uit</b>
<b>Controles dag 2</b>	22-6-2014	
Klant/ web portaal		
Polissysteem	Polis status actief	Polis status actief
Klant systeem		
Financieel systeem	Grootboek post x aangeveleverd	Grootboek post x aangeveleverd
BKR	BKR check uitgevoerd en retour gestuurd	BKR check uitgevoerd en retour gestuurd
Bank		
Post	Polisblad en factuur naar klant gestuurd	Polisblad en factuur naar klant gestuurd
<b>Acties dag 3</b>		
Klant/ web portaal		
Polissysteem		
Klant systeem		

Financieel systeem		
BKR		
Bank		
Post		
<b>Batchproces 2&gt;3</b>	<b>Beheer voert per systeem dagelijkse programma's uit</b>	<b>Beheer voert per systeem dagelijkse programma's uit</b>
<b>Controles dag 3</b>	23-6-2014	
Klant/ web portaal	actieve polis	actieve polis
Polissysteem	Status betaling: verstuurd	Status betaling: verstuurd
Klant systeem		
Financieel systeem	status betaling: verstuurd	status betaling: verstuurd
BKR		
Bank	Incassoopdracht verstuurd naar bank	Incassoopdracht verstuurd naar bank
Post		
<b>Acties dag 4</b>	23-6-2014	
Klant/ web portaal	Klant logt in: ziet actieve polis	Klant logt in: ziet actieve polis
Polissysteem		
Klant systeem		
Financieel systeem		
BKR		
Bank	verwerk incasso (retourbericht versturen): OK	verwerk incasso: retourbericht met status NOK, geen saldo
Post		
<b>Batchproces 3&gt;4</b>	<b>Beheer voert per systeem dagelijkse programma's uit</b>	<b>Beheer voert per systeem dagelijkse programma's uit</b>
<b>Controles dag 4</b>	24-6-2014	
Klant/ web portaal		
Polissysteem	Status betaling: voldaan	Status betaling: open
Klant systeem		
Financieel systeem	Status betaling: voldaan	Status betaling: open
BKR		
Bank		
Post		brief met betalingsverzoek en acceptgiro naar klant
<b>Acties dag 5</b>	24-6-2014	
Klant/ web portaal		
Polissysteem	Polis handmatig prolongeren	
Klant systeem		
Financieel systeem		
BKR		
Bank		retourbericht met betaald betalings- kenmerk acc giro (klant betaalt acceptgiro)
Post		

Batchproces 4>5	Beheer voert per systeem dagelijkse programma's uit	Beheer voert per systeem dagelijkse programma's uit
<b>Controles dag 5</b>	25-6-2014	
Klant/ web portaal		
Polissysteem	Nieuwe dekkingsperiode aangemaakt	Status betaling: voldaan
Klant systeem		
Financieel systeem	Grootboekpost voor nieuwe periode aangeleverd	Status betaling: voldaan
BKR		
Bank	Incasso naar bank voor nieuwe premie	
Post	Nieuwe polis en factuur voor nieuwe periode	
<b>EINDE TESTGEVAL</b>		<b>EINDE TESTGEVAL</b>

#### 12.4 Voorbeeld draaiboek testomgeving

Een draaiboek van de testomgeving bevat alle belangrijke handelingen die nodig zijn om de testomgeving en zijn inhoud te beheren en in de juiste toestand te krijgen voor de testen. De relevante testhandelingen en controles (alvorens volgende handelingen door bijvoorbeeld beheerders te kunnen laten uitvoeren) zijn ook benoemd (bijvoorbeeld 8 en 19).

Nr	Activiteit	wie	draaidat	%	start	eind	Opmerkingen
1	Laden save in overleg!	beheer		100	23-11	23-11	
2	laatste dagwerk van de maand	beheer		100	8-9	8-9	
3	prolongatie januari	beheer		100	9-9	9-9	
4	signaleringsperiode expiratie run op 3/4 mnd, LLH oud 3 mnd, X en SYSB 0 mnd. Brieven klant 3 mnd voor expdatum	beheer		100	23-11	23-11	
5	Toewijzingsdatum en kaseinde datum aanpassen	beheer		100	9-9	9-9	
6	Vaste waarde in PD: AANT_KAL-DGN-ADV-HERINN aanpassen in 266	beheer		100	9-9	9-9	
7	expiratie run draaien	beheer		100	10-9	10-9	
8	controle registraties Expiraties	test		100	10-9	10-9	
9	dagwerk draaien na checken registratie	beheer		100	10-9	10-9	
10	één expiratie polis X-G en voor andere gebeurtenis registreren	test		100	13-9	13-9	
11	checken of alle SYSB polissen klaar zijn voor kastoewijzing	test		100	13-9	13-9	
12	aanpassen kassen - VEW potje en Toewijzingsdatum/kaseinde datum aanpassen	beheer		100	14-9	14-9	
13	Invoer uitkeringsgegevens voor SYSB polissen. Status op T	test		100	9-9	9-9	
14	mutaties overlijden uitvoeren voor open kassen en sp.kassen in december ivm overlevingswinst run	test		100	9-9	9-9	
15	doordraaien naar 31-12-2009	beheer	31-12-2009	100	10-9	13-9	
16	inhoud omgeving bekijken (onterecte BFX boekingen)!!!!	beheer		100	13-9	13-9	
17	Kas gegevens vullen(potjes(vev), nrs codes etc) en noteren	test		100	13-9	13-9	
18	save maken	beheer		100	23-11	23-11	
19	akkoord Systeemtest	PM		0	23-11	23-11	
20	aanpassen pd release	beheer		100	23-11	23-11	
21	installeren programmatuur	beheer		100	23-11	23-11	
22	éénmalige programmatuur draaien	beheer		100	24-11	24-11	
23	save maken	beheer		100	24-11	24-11	
24	Incidentele Lijst X draaien	beheer		100	24-11	24-11	
25	save maken	beheer		100	24-11	24-11	
26	intaketest	test		100	24-11	24-11	
27	checken of alle SYSB polissen klaar zijn voor kastoewijzing	test		100	24-11	24-11	
28	draaien herstelacties op de H omgeving	test		100	25-11	25-11	
29	controle op omgeving inhoud(incl alle herstelacties)	test		100	26-11	26-11	
30	eerste dagwerk van de maand januari 2010 draaien	test	01-01-2010	100	26-11	26-11	
31	Kastoewijzing SYSB draaien	beheer	04-01-2010	100	29-11	29-11	
32	Staat420-incidenteel-v2:	beheer		100	3-12	3-12	
32	Controle kastoewijzing door Ingrid	beheer		100	6-12	6-12	
33	fiattering van de kastoewijzing	beheer		100	6-12	6-12	
34	controle op kassen, expiratie en vullen afvoerdatum polissen	test		100	6-12	6-12	
35	MAKEN SAVE	test		100	6-12	6-12	
36	selectie voor IA voor de 107 polissen + extra reserve lijst	beheer		100	6-12	6-12	
37	controle selectie IA en vergelijken met N omgeving ?	test		0	6-12	6-12	
38	IA voor 107+ reserve polissen	beheer		100	6-12	6-12	
39	Controle IA en vergelijken met de N omgeving?	test		0	6-12	6-12	
40	regulier dagwerk	beheer		100	6-12	6-12	
41	uitvoeren testronde 1: regressie/progressie Exp items gesloten kassen	test		10	6-12	16-12	
42	uitvoeren testronde 1: regressie SpB incl HWN	test		0	6-12	13-12	
43	JIW boekingen controleren Budget B	test		100	6-12	6-12	
44	controle van ckk en CFK boekingen en stand potjes controleren VOW incl ckk	test		100	6-12	6-12	
45	maken Save	beheer		nvt	7-12	7-12	
46	Controle online op afsluitende correspondentie	test		100	7-12	7-12	
47	regulier dagwerk	beheer		100	6-12	6-12	
48	Controle verwerking afhandelautomaat en correspondentie	test	05-01-2010	100	7-12	7-12	
49	save maken	beheer		100	30-11	30-11	HLLMAL1.S00.HLLM5004.D2010348.T132407 (na 50)
50	draaien standaard dagwerk (LDW + PROL feb 2010)	beheer	09-01-2010	100	30-11	30-11	
51	VOW potjes controleren	test		nvt	1-12	1-12	
52	mutatie overlijden SYSB/X-G polis in december !!	test		nvt	1-12	1-12	
53	draaien overlevingswinst run DD16 (dagwerk)	beheer	18-01-2010	100	1-12	1-12	
54	Staat420-incidenteel-v2:	beheer			1-12	1-12	
55	controle overlevingswinst verdeling VOW potje controle	test	18-01-2010		1-12	1-12	
56	dagw29-01 / dagw02-01 /	beheer	29-01-2010	75	1-12	1-12	
57	maken Save (na restore en dagw30-01)	beheer	30-01-2010	0	1-12	1-12	
58	invoeren uitkeringsgegevens + status X-G	test		0	2-12	2-12	
59	één polis voor andere gebeurtenis geregistreerd laten staan	test		0	2-12	2-12	
60	VOW stand noteren vóór run	test		0	2-12	2-12	VOW
61	controle van ckk en CFK boekingen en stand potjes controleren VOW incl ckk	test		0	2-12	2-12	
62	maken Save 3b	beheer		0	2-12	2-12	zie 31
63	Kastoewijzing proberen in dagwerk (melding één openstaande andere gebeurtenis)	beheer		0	2-12	2-12	