

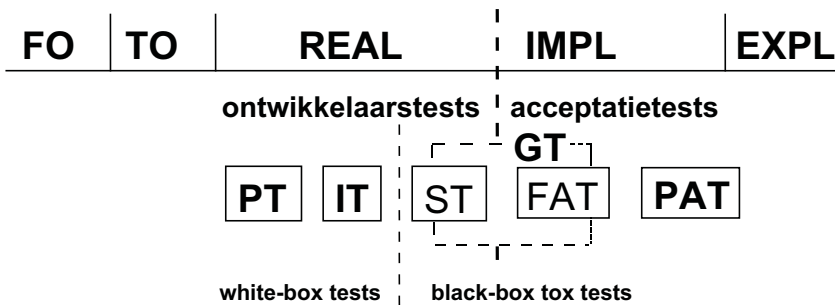


laat staan vergezeld van de voor de acceptatietest onmisbare documentatie, zoals de gebruikershandleiding.

Geleidelijk aan is daarom de behoefte ontstaan om bij de *black-box tests* (de systeem- en de acceptatietests) op een aantal punten deze afbakening te pragmatiseren door middel van afstemming en integratie. Dat is mogelijk zowel ten aanzien van de te realiseren dekkingsgraad als van de te gebruiken testomgeving.

Dit heeft echter tot gevolg dat ook de afbakening in verantwoordelijkheden minder duidelijk is geworden. Het is dan ook vereist dat er (vooraf) goede afspraken gemaakt worden over de verantwoordelijkheden van de betrokken disciplines. Slechts dan kan door samenwerking *een win-win situatie* bereikt worden. Er zal sprake zijn van tijdwinst, vroegere foutdetectie en betere kwaliteit van het testproces.

In veel organisaties wordt in dit kader de zogenaamde “geïntegreerde test” ingevoerd, waarbij de systeemtest en de functionele acceptatietest (geheel of gedeeltelijk) zijn samengevoegd (zie figuur 3.4). Deze test wordt nader beschreven in paragraaf 29.4.



**Figuur 3.4:** Geïntegreerde test

Zo'n samenwerkingsverband levert de gebruikers en beheerders vroege betrokkenheid en kennis van het systeem en de (test)omgeving alsmede test-know-how en planningsinvloed. Aan de ontwikkelaars wordt onder andere materiekennis en vroegere foutdetectie geboden. Ook zal een geïntegreerde test de niet onbelangrijke acceptatiegraad van het systeem verhogen.

Door ook af te stappen van het dogma dat een betrokkene zijn volledige test uitvoert in zijn “eigen” omgeving, kunnen per testsoort verschillende omgevingen worden gebruikt. De systeemtest zal zich gewoonlijk in een laboratorium-omgeving afspelen, maar voor bepaalde tests kan een omgeving worden gebruikt die bijvoorbeeld wat omvang betreft even groot is als de productie-omgeving. De functionele acceptatietest wordt traditioneel uitsluitend in een “als-ware-het-productie” omgeving uitgevoerd. Dit is echter niet altijd nodig. In een “als-ware-het-productie” omgeving is het testen door de (terecht!) strakke rekencentrumprocedures veel omslachi-

<b>C3</b> (=C3a & C3b & C3c & (C3d   C3e))	<b>1</b> contributie: = contributie + 50	<b>0</b> contributie: = contributie
datum inschrijving < 1-1-1988 (C3a)	<u>1.1.1.01</u>	<u>0.1.1.01</u>
soort lidmaatschap = 'algemeen' (C3b)	<u>1.1.1.01</u>	1.0.1.01
indicatie wanbetaler = 0 (C3c)	<u>1.1.1.01</u>	1.1.0.01
soort speler = 'comp' (C3d)	1.1.1. <u>10</u>	1.1.1. <u>00</u>
niveau speler = 'c' (C3e)	<u>1.1.1.01</u>	<u>1.1.1.00</u>

Uitgeschreven komen we zo tot de volgende logische testsituaties voor C3:

Testsituatie	C3.1	C3.2	C3.3	C3.4	C3.5	C3.6
C3	1 (11101)	1 (11110)	0 (01101)	0 (10101)	0(11001)	0(11100)
datum inschrijving	< 1-1-1988	< 1-1-1988	>= 1-1-1988	< 1-1-1988	< 1-1-1988	< 1-1-1988
soort lidmaatschap	algemeen	algemeen	algemeen	≠algemeen	algemeen	algemeen
indicatie wanbetaler	0	0	0	0	1	0
soort speler	≠comp	comp	≠comp	≠comp	≠comp	≠comp
niveau speler	c	≠c	c	c	c	≠c

## Bepalen logische testgevallen

Bij het bepalen van de logische testgevallen moet gezocht worden naar die functionele paden, waarbij elke situatie van iedere conditie minimaal één keer doorlopen wordt. Daarbij dient rekening te worden gehouden met de mogelijkheid dat eenzelfde vergelijking in meerdere condities voor zou kunnen komen. Een voorbeeld hiervan is de vergelijking 'indicatie wanbetaler = 0' in de condities 2 en 3. Ook moet er bij de keuze van padcombinaties op gelet worden dat het verwachte eindresultaat zo uniek mogelijk is voor de gekozen padcombinaties. In het voorbeeld geeft C2=waar én C3=waar hetzelfde resultaat als C2=onwaar én C3=onwaar, want de contributie blijft gelijk. Testgevallen met C2=waar en C3=onwaar of omgekeerd hebben wel een uniek eindresultaat en zijn als zodanig betere testgevallen.

Een hulpmiddel bij het vinden van de testgevallen en een controlemiddel of alle situaties in de testgevallen zijn opgenomen, is het gebruik van een matrix. Hierin worden in de eerste kolom alle gedefinieerde testsituaties opgenomen. In de tweede kolom wordt de waarde van de conditie bij die situatie vermeld en in de derde de daaropvolgende conditie. Vervolgens worden de testgevallen vermeld. Als de testgevallen goed zijn gedefinieerd, zullen alle te testen situaties minimaal een keer aangekruist zijn. De matrix van het contributievoorbeld ziet er als volgt uit:

## 15.10 Programma interfacetest

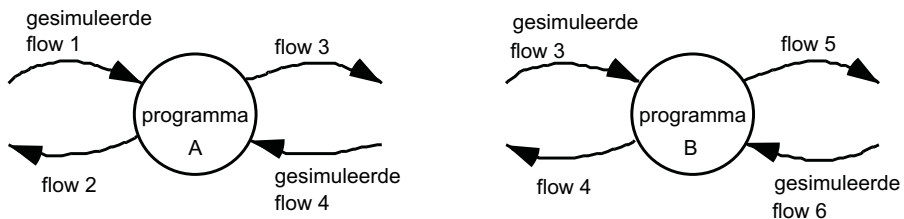
### 15.10.1 Algemeen

De programma interfacetest (PIT) is een testspecificatietechniek die wordt gebruikt om de interfaces tussen de verschillende programma's c.q. modules te testen. Voordat er met een programma interfacetest kan worden begonnen, moeten de te integreren programma's zelf reeds onafhankelijk van elkaar zijn getest met behulp van gesimuleerde gegevensstromen. De aanname is dat alle te integreren programma's los van elkaar goed functioneren mits aangestuurd met gesimuleerde gegevensstromen. Met behulp van deze testspecificatietechniek wordt geverifieerd of de programma's na integratie nog steeds correct functioneren. In dit geval worden de programma's aangestuurd met "echte" gegevensstromen in plaats van met gesimuleerde gegevensstromen.

De programma interfacetest is een testspecificatietechniek die met name wordt gebruikt bij de integratietest.

### 15.10.2 Globale werking

Globaal werkt de programma interfacetest als volgt:



Programma A en programma B zijn reeds getest met behulp van gesimuleerde gegevensstromen. De invoer van programma A is gesimuleerd door de stromen 1 en 4, de uitvoer (stroom 2 en 3) is bestudeerd en gecontroleerd op juistheid. Een soortgelijke redenering geldt voor programma B: de invoer is gesimuleerd door de stromen 3 en 6, de uitvoer (4 en 5) is bestudeerd en gecontroleerd op juistheid.

Tijdens de PIT moet worden geverifieerd of programma B nog steeds correct functioneert indien de voor invoer gebruikte gesimuleerde stroom 3 wordt vervangen door de "echte" stroom 3 en of programma A nog steeds correct functioneert indien de voor invoer gebruikte gesimuleerde stroom 4 wordt vervangen door de "echte" stroom 4.

- Het ontbreken van een detailplanning van het ontwikkelteam voor de diverse deelsystemen;
- De vooraf vastgestelde einddatum kan invloed hebben op de volledige uitvoering van de testactiviteiten zoals vermeld in het testplan;
- Het niet tijdig beschikbaar zijn van de testbasis (AO-procedures, gebruikershandleiding en ontwerpspecificaties);
- Onvoldoende kwaliteit van de testbasis;
- Het niet tijdig beschikbaar zijn van functiepunttellingen, alsmede de betrouwbaarheid ervan. Beide aspecten leiden indirect tot een onbetrouwbare testplanning bij het toepassen van testpuntanalyse;
- De beschikbaarheid van testpersoneel (capaciteit, geschiktheid qua testervaring en materiekennis);
- De groei van de omvang, uitgedrukt in functiepunten, van het te testen informatiesysteem;
- De beschikbaarstelling en de werking van de gewenste testomgeving;
- De beheersbaarheid van de testomgeving en van alle elementen (programmatuur en gegevens) daarbinnen;
- De introductie van een nieuwe testaanpak waarmee geen ervaring is binnen de organisatie.

## 17.6 Checklist Structurering

Indien gestructureerd testen wordt ingevoerd in een organisatie, dient allereerst te worden bepaald wat de huidige status ten aanzien van het testen is. Om dit te bepalen wordt een kort onderzoek uitgevoerd genaamd testinventarisatie. De hierna beschreven checklist kan worden gebruikt bij het uitvoeren van een testinventarisatie<sup>1</sup>.

### Testfasering

- Welke testsoorten worden onderscheiden (programmatest, systeemtest, acceptatietest, enzovoort) en wat wordt per testsoort getest?
- Wat is de samenhang tussen deze testsoorten?
- In hoeverre vullen de testsoorten elkaar aan c.q. bestaat er overlap tussen de testsoorten?
- Wordt het testen voor iedere te onderscheiden testsoort gefaseerd uitgevoerd? (Planning en beheer, Voorbereiding, Specificatie, Uitvoering en Afronding.)

<sup>1</sup> Noot van de auteurs: een dergelijke inventarisatie kan veel beter plaatsvinden met behulp van het in hoofdstuk 25 beschreven TPI-model dan met deze checklist. Omdat het TPI-model echter te omvangrijk is om volledig in dit boek gepubliceerd te worden, is deze checklist gehandhaafd.

Pijler	Aandachtsgebied	Omschrijving
I	Testautomatisering	<p>Automatisering binnen het testproces kan op heel veel verschillende manieren plaatsvinden en heeft in de regel een of meer van de volgende doelen:</p> <ul style="list-style-type: none"> <li>- minder benodigde uren;</li> <li>- kortere doorlooptijd;</li> <li>- meer testdiepgang;</li> <li>- grotere flexibiliteit bij het testen;</li> <li>- meer en/of sneller inzicht in de status van het testproces;</li> <li>- betere motivatie van het testpersoneel.</li> </ul>
I	Testomgeving	<p>De testuitvoering vindt plaats in een testomgeving. Deze omgeving is op hoofdlijnen samengesteld uit de volgende componenten:</p> <ul style="list-style-type: none"> <li>- hardware;</li> <li>- software;</li> <li>- communicatiemiddelen;</li> <li>- faciliteiten voor opbouw en gebruik van bestanden;</li> <li>- procedures.</li> </ul> <p>De omgeving moet dusdanig worden samengesteld en ingericht dat aan de hand van de testresultaten optimaal kan worden vastgesteld in hoeverre het testobject aan de gestelde eisen voldoet. De omgeving heeft grote invloed op de kwaliteit, doorlooptijd en kosten van het testproces. Belangrijke aspecten van de omgeving zijn verantwoordelijkheden, beheer, tijdige en voldoende beschikbaarheid, representativiteit en flexibiliteit.</p>
I	Testwerkplek	<p>Het testpersoneel heeft kamers, bureaus, stoelen, PC's, tekstverwerkingsfaciliteiten, printers, telefoons, enzovoort nodig. Een goede en tijdige inrichting van de kantoorinfrastructuur heeft positieve invloed op de motivatie van het testpersoneel, op de communicatie binnen én buiten het team en op de efficiënte uitvoering van de werkzaamheden.</p>
O	Commitment en motivatie	<p>Het commitment en de motivatie van de diverse betrokkenen rond het testen zijn belangrijke voorwaarden voor een goed verlopend testproces. Onder de betrokkenen vallen niet enkel de testers zélf, maar ook bijvoorbeeld het projectmanagement en het lijnmanagement. Deze laatstgenoemde betrokkenen zijn met name belangrijk in de voorwaarden-scheppende sfeer. Het testproces krijgt dan voldoende tijd, geld en middelen (kwantitatief en kwalitatief) om een goede test uit te voeren, waarbij de samenwerking en goede communicatie met de rest van het project een zo efficiënt mogelijk totaalproces geeft.</p>

veau omvatten ook de eisen behorende bij de lagere niveaus. Een testproces op het B-niveau voldoet zowel aan de eisen van het A-niveau als het B-niveau. Wanneer een testproces niet aan de eisen van het A-niveau voldoet, zit het proces op het startniveau. Aan dit laagste niveau worden geen eisen gesteld. In de volgende tabel zijn de niveaus voor de verschillende aandachtsgebieden weergegeven.

Aandachtsgebied	A	B	C	D
Teststrategie	Strategie voor betreffende testsoort	Strategie voor de black-box testsoorten	Strategie voor de black-box testsoorten plus white-box testsoorten óf toetssoorten	Strategie voor alle test- en toetssoorten
Toepassing fasering	Hoofd fasen: Planning, Specificatie, Uitvoering	Volledige fasering: Planning, Voorbereiding, Specificatie, Uitvoering en Afronding		
Moment van betrokkenheid	Afronding testbasis	Opstellen testbasis	Opstellen eisen	Initiatie project
Begroting en planning	Onderbouwde begroting en planning	Statistisch onderbouwde begroting en planning		
Testspecificatietechnieken	Informeel technieken	Formeel black-box technieken		
Statische testtechnieken	Detail intake	Checklists		
Metrics	Statistieken per project (over product)	Statistieken per project (over proces)	Statistieken per systeem	Statistieken per organisatie (>1 systeem)
Testautomatisering	Gebruik van tools	Beheerste test-automatisering	Optimale test-automatisering	
Testomgeving	Beheerste testomgeving	Testen in meest geschikte omgeving	“Omgeving op afroep”	
Testwerkplek	Testwerkplekken tijdig geregeld			

## Test Volwassenheid Matrix

	Schaal	0	1	2	3	4	5	6	7	8	9	10	11	12	13	
<b>Aandachtsgebied</b>																
Teststrategie		A						B				C		D		
Toepassing fasering		A				B										
Moment van betrokkenheid			A					B				C		D		
Begroting en planning					A							B				
Testspecificatietechnieken		A			B											
Statische testtechnieken						A		B								
Metrics							A			B			C		D	
Testautomatisering					A				B			C				
Testomgeving					A				B						C	
Testwerkplek					A											
Commitment en motivatie		A					B						C			
Testfuncties en opleidingen					A			B				C				
Toepassingsgraad van de methodiek						A						B			C	
Overleg			A			B							C			
Rapportage		A				B		C					D			
Bevindingenbeheer		A					B		C							
Testware beheer			A				B				C				D	
Testproces beheer		A			B								C			
Toetsen								A			B					
White-box testsoorten						A		B		C						
		Beheerst					Efficiënt					Optimaliserend				

De open vakjes tussen de verschillende niveaus geven aan dat het bereiken van een grotere volwassenheid voor het betreffende aandachtsgebied gerelateerd is aan de mate van volwassenheid van andere aandachtsgebieden. Er is dus geen gradatie tussen de niveaus aangebracht in het model: als een testproces voor een bepaald aandachtsgebied bijna (maar net niet helemaal) op niveau B zit, wordt het ingedeeld in niveau A.

Voldoet het testproces voor een bepaald aandachtsgebied niet aan het A-niveau dan wordt het proces voor dat aandachtsgebied op de nul-schaal gezet.

### Opbouw van de matrix

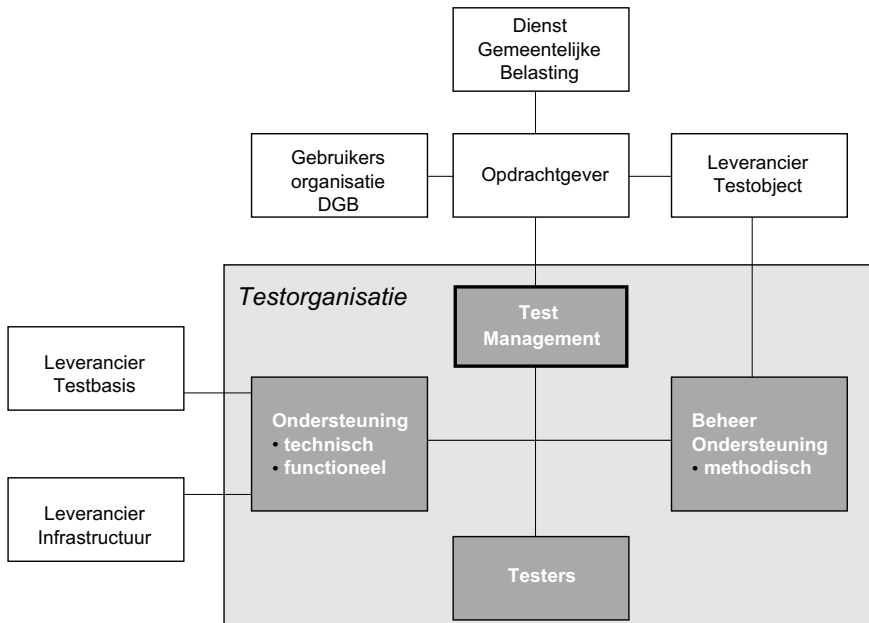
De diverse schalen van testvolwassenheid kunnen op hoofdlijnen in drie categorieën worden ingedeeld:

- beheerst;
- efficiënt;
- optimaliserend.

- Eerdere start  
Eerder in de systeemontwikkeling beginnen met het testen betekent niet alleen dat fouten eerder kunnen worden gevonden, maar ook dat afstemming tussen de verschillende tests en toetsen en tussen testen en ontwerp/bouw beter mogelijk is.  
*Aandachtsgebieden: Teststrategie, Moment van betrokkenheid, Toetsen, White-box testsoorten*
- Grotere betrokkenheid bij voorgaande fasen  
Het betrekken van testen bij eerdere fasen als ontwerp en bouw heeft een verbeterde communicatie als voordeel en helpt daarnaast om de testbaarheid van het systeem te vergroten. Hierdoor kan het testproces efficiënter werken en kunnen fouten eerder worden gevonden of zelfs worden voorkomen.  
*Aandachtsgebieden: Moment van betrokkenheid, Commitment en motivatie, Overleg, Rapportage*
- Steeds verdergaande automatisering van het testproces  
Met het automatiseren van het testproces kunnen voordelen als kortere doorlooptijd, minder kosten en hogere kwaliteit van het testproces worden behaald. Zoals voor alle automatisering geldt, dient het als een hulpmiddel te worden gezien en niet als een doel op zich.  
*Aandachtsgebieden: Testautomatisering*
- Professionalisering van het testen  
Het testen dient de aandacht te krijgen die het verdient. Er wordt verwacht dat het testproces wordt verbeterd, dat de testers deskundige adviezen geven, enzovoort. Dit betekent dat de mensen die dit moeten doen voldoende capabel zijn. Testopleidingen, functie-omschrijvingen en organisatorische ondersteuning kunnen hierbij helpen.  
*Aandachtsgebieden: Testfuncties en opleidingen, Commitment en motivatie*

## 4.2 Organisatiestructuur

In het volgende schema zijn de relaties en rapportagelijnen zowel binnen het testteam als daarbuiten weergegeven.



**Figuur A.1:** Organisatie acceptatietestteam

De volgende rapportagelijnen worden ingevuld:

- Wekelijks en op verzoek ad-hoc rapporteert het testmanagement aan de opdrachtgever over de voortgang van het testproces en de kwaliteit van het testobject;
- De inhoud van deze rapportage komt overeen met de ‘Concept inhoud voortgangsrapportage’, zoals beschreven in het hoofdstuk ‘Testbeheer’ van het boek “Testen volgens TMap” [5];
- De communicatie tussen het testteam en de Dienst Gemeentelijke Belastingen en de communicatie tussen het testteam en de diverse leveranciers (testbasis, testobject, testomgeving en testtools) over wijzigingen in de opleveringen, verlopen via de opdrachtgever;
- Beheer verzorgt de communicatie tussen het testteam en de leverancier van het testobject over de oplevering en overdracht van het testproject en de bevindingen;
- De technische en de functionele ondersteuning maken deel uit van het testteam. Zij verzorgen de communicatie tussen het testteam en de le-

# TMap-gerelateerde publicaties

- Pol, M., R. Teunissen, E. van Veenendaal (1996), *Gestructureerd testen: een introductie tot TMap*, Tutein Nolthenius, 's-Hertogenbosch, ISBN 90-72194-45-4
- Pol, M., E. van Veenendaal (1998), *Structured Testing of Information Systems: an Introduction to TMap*, Kluwer Bedrijfsinformatie, Deventer, ISBN 90-267-2910-3
- Koomen, T. en M. Pol (1998), *Test Process Improvement, Leidraad voor stapsgewijs beter testen*, ten Hagen & Stam, ISBN 90-440-0091-8
- Koomen, T. and M. Pol (1999), *Test Process Improvement, A practical step-by-step guide to structured testing*, Addison-Wesley, ISBN 0-201-59624-5.
- Broekman, B., C. Hoos en M. Paap (2001), *Automatisering van de Testuitvoering*, ten Hagen & Stam, ISBN 90-440-0103-5

- vrijgave productie 373
  - white-box test 240
  - CKL 200
  - client/server 538
  - comparator 518
  - compiler 518
  - complexiteit 206, 213
  - componenten 420
  - conditie 250, 252, 289
  - condition coverage 253, 274
  - condition/determination coverage 274
  - configuratiemanagement 512
  - configuratietest 555
  - connectiviteit 189, 344
  - continuïteit 186, 322
  - continuïteit 345
  - controle 394
  - controleerbaarheid 186, 351
  - controlepunten 492
  - converteren van productiegegevens 504
  - coördinatie en advies 395, 414
  - correctief planbaar onderhoud 534
  - correctieve maatregelen 30
  - coverage 60, 253, 511, 519
  - cross-reference tabel 302
  - CRUD 255
  - CRUD-matrix 301
- D**
- dagproductie 532
  - data-driven 521
  - dataflowtest 200, 232, 282
  - debugger 515
  - decharge 149, 156
  - decision coverage 253, 273
  - decision/condition coverage 253, 274
  - deelsysteem 104, 195
  - definitie van kwaliteit 29
  - definitie van testen 21 - 22, 35
  - degradatiemogelijkheid 186, 349
  - dekkingsgraad 60, 270
  - detail intake testbasis 125, 229
  - detailleringsmaat 265, 272
  - detaillplanning 116, 123
  - detectieve maatregelen 30
  - determinant 266
  - DFT 200, 282
  - diepgang 253, 258
  - discussie meeting 247
  - DIT 200
  - draagvlak 96, 118, 479
  - duivelsvierkant 432
  - dummy 210, 214
  - dynamisch expliciet testen 35
  - dynamisch impliciet testen 35, 337
  - dynamische kwaliteitsattributen 185
- E**
- EG 200
  - eisen aan testomgeving 496
  - electronic meeting systems 514
  - elementaire vergelijkingstest 200, 232, 288
  - entry-criteria 174, 245
  - equivalence partitioning 254
  - equivalentieklasse 254, 308, 334
  - ernstcategorie 429
  - error guessing 200, 233, 299
  - ervaringsgegevens 152
  - evaluatie testproject 361
  - evaluatierapport 154
  - evolutionaire systeemontwikkeling 556
  - EVT 200, 288
  - exit-criteria 174, 248
  - expertise 378, 403
  - extrapolatie 226
- F**
- faalkans 181, 183
  - faalkosten 153
  - facilitaire ondersteuning 398
  - fasering 54, 77
  - FAT 194
  - financiële planning 94, 115
  - fixed date 538
  - fixed price 538
  - flexibiliteit 136, 189, 352
  - FO-functie 389
  - follow-up 248
  - formele testspecificatietechnieken 252
  - foutkans 183, 201
  - foutregistratiemeeting 247
  - foutvandeffectiviteit 463
  - FPA 205
  - FPA-functie 210
  - framework architectuur 523
  - freeze/unfreeze 504
  - functie
    - Applicatie Integrator 398
    - beheer 392
    - controle 394
    - coördinatie en advies 395
    - facilitaire ondersteuning 398
    - functionele ondersteuning 389