

**Matchen**  
**op**  
**Competenties**

**een ontologisch fundament**

**Jan L.G. Dietz**



## **INHOUD**

1	Inleiding	5
2	Begrippen aan de vraagzijde	9
3	Begrippen aan de aanbodzijde	15
4	Matchen op competenties	21
	Bijlage A: Het ICT-werkerterrein	25
	Bijlage B: Het ontologisch schema voor MoC	37



## 1. Inleiding

De afstemming van vraag en aanbod op de arbeidsmarkt verloopt niet optimaal, zo is de indruk van velen, terwijl het belang van een optimale afstemming alleen maar toeneemt. Nieuwe factoren die dit belang vergroten, zijn onder andere de behoefte aan het vergaand automatiseren van het matchingsproces en de overgang van een Nederlandse naar een Europese markt. Het groeiende bewustzijn dat functies en beroepen sterk samenhangen met opleiding en met certificering maakt het geheel nog complexer. Ziehier in een notendop het probleem waarvoor dit boekje beoogt een oplossing te bieden, niet een complete oplossing maar wel een aanzet tot, een fundament voor bruikbare oplossingen. Een centraal woord daarin is "*competentie*". Dat woord gonst al enige tijd in de kringen van degenen die zich beroepshalve druk maken over het geschetste probleem. Maar nieuwe woorden lossen natuurlijk niets op, dat kunnen alleen nieuwe begrippen en nieuwe denkwijzen. En dan nog alleen maar als ze haarscherp zijn gedefinieerd, als misverstanden vrijwel uitgesloten worden. Dat is wat dit boekje biedt, een inventarisatie van de kernbegrippen en een haarscherpe definitie ervan. Vandaar de subtitel "een ontologisch fundament". Wat is dat dan, ontologisch? Ontologie is een oorspronkelijk Grieks woord dat zoveel betekent als kennis van de essentiële aard der dingen. De moderne, praktische betekenis van het woord is: de precieze, samenhangende, definitie van de begrippen die essentieel zijn binnen een (professionele) gemeenschap. Een ontologie vormt de begripsmatige 'bijbel' voor een gemeenschap, die bijvoorbeeld het verplichte uitgangspunt zou moeten zijn voor elk automatiseringsproject. Want iedereen heeft inmiddels wel ervaren dat zulke projecten de ultieme en meedogenloze test zijn voor de kwaliteit van professionele 'bodies of knowledge'. Voorbeelden van professionele gemeenschappen zijn de medische wereld, de financiële wereld en de wereld van werk en opleiding, waarover dit boekje gaat. Die wereld is niet alleen groot maar ook groeiende en multiprofessioneel. Niet alleen arbeidsmakelaars, zoals de CWI's in Nederland, behoren er toe maar ook uitkeringsinstanties en personeelsmanagers en opleidingsinstututen. En wat te denken van de Europese ministeries van Sociale Zaken en van Onderwijs? Enzovoorts, enzovoorts.

Kortom, er bemoeien zich zeer veel personen van uiteenlopende professies met het probleem van de afstemming van vraag en aanbod op de arbeidsmarkt. Des te belangrijker is het dus dat er een gezamenlijk gedragen ontologie komt. Maar er is toch al van alles, zult u wellicht zeggen, we beschikken toch al over diverse heel bruikbare taxonomieën en classificaties? Zeker, en die zijn ook nodig maar het zijn niet de zaken waar de aandacht allereerst naar uit zou moeten gaan. Want wat betekent het te beschikken over een classificatie van gedragsfactoren als

het niet glashelder is wat onder een gedragsfactor moet worden verstaan? Zonder zo'n definitie is het bijvoorbeeld onduidelijk of twee classificaties van verschillende instituten wel 'hetzelfde' zijn, of je ze dus zou kunnen uitwisselen. Mensen zijn geneigd te denken dat ze elkaar verstaan als ze dezelfde woorden gebruiken, maar dat kan helaas alleen maar met tamelijke zekerheid worden gezegd van zeer homogene gemeenschappen met een lange traditie, zoals de oude ambachten. Daarbuiten, dat wil dus zeggen bijna overal, voeren misverstanden al gauw de boventoon. Om een heel ander voorbeeld te geven: je moet eerst weten wat temperatuur is, pas daarna heeft het zin te kiezen voor een thermometer in graden Celsius of Fahrenheit. En dat is precies wat ontologie doet: vastleggen wat de fysische grootte die we temperatuur noemen in essentie is, of wat beroep is of functieniveau of intelligentiesoort. Pas daarna heeft het zin te praten over standaarden, dat wil zeggen over het invullen van de gedefinieerde 'meetschalen'. Die hoeven ook niet voor de eeuwigheid vast te liggen. Er zullen altijd beroepen verdwijnen en bijkomen, en in het ene geval heb je behoefte aan meer functieniveau's dan in het andere, zoals bijvoorbeeld ook het aantal onderscheiden soorten van intelligentie is veranderd in de loop der tijd.

Zoals gezegd wordt het begrip competentie momenteel door veel partijen gezien als een nieuw ankerpunt, een nieuwe basis voor het aanpakken van de problematiek van de afstemming van vraag en aanbod, die daarom nu "matchen op competenties" wordt genoemd. De tijd zal leren hoe effectief dat begrip is. In elk geval heeft het een veel meer elementair karakter dan de traditionele begrippen beroep en functie, en dat is goed. Voor elke ontologie is het belangrijk een geschikt atomair niveau van begrippen te vinden, waarop zich zogezegd de 'legoblokjes' van de ontologie bevinden. Je kunt namelijk wel altijd grotere componenten maken maar nooit kleinere. Wat ik probeer te doen, is de benodigde 'legoblokjes' te bepalen, die zo helder mogelijk te definiëren en te kijken hoe je er mee zou kunnen 'bouwen'. Niet meer dus dan een aanzet voor de ontologie die moet worden ontwikkeld. Die aanzet staat wat mij betreft ook helemaal open voor elke discussie. De kern van het boekje is het ontologische schema dat is gepresenteerd in bijlage B. De eerdere hoofdstukken zijn eigenlijk vooral een uitleg van dat schema. Die uitleg zal voor de meeste lezers echter nodig zijn omdat weinigen gewend zullen zijn zulke schema's te lezen. Een ontologisch schema wordt uitgedrukt in een (meestal grafische) logische taal. Natuurlijke taal is nu eenmaal ongeschikt om dingen heel precies vast te leggen, zij is wel geschikt om uitleg en toelichting te geven. Het is daarom het beste om bij het lezen van de hoofdstukken 2, 3 en 4 het ontologische schema ernaast te hebben. De legenda van de gebruikte grafische taal wordt gaandeweg uitgelegd en is ook nog eens samengevat in bijlage B.

We zullen vier soorten begrippen onderscheiden, te weten basisbegrippen, schaalbegrippen, elementbegrippen en compositiebegrippen.

Onder *basisbegrippen* worden de klassen van dingen verstaan die we willen onderscheiden. Een voorbeeld van een basisbegrip is gedragskenmerk. Dat is een verzamelnaam voor een klasse van gedragskenmerken. Voorbeelden van (afzonderlijke) gedragskenmerken zijn sociale intelligentie en motivatie. Een ander voorbeeld van een basisbegrip is beroep. Voorbeelden van beroepen zijn arts, loodgieter en bakker. Onder *schaalbegrippen* worden de meetschalen verstaan waarop kan worden vastgelegd in welke mate een bepaald basisbegrip aanwezig is, bijvoorbeeld in welke mate een persoon over een sociale intelligentie beschikt. Dit schaalbegrip heet kenmerkniveau. Een ander voorbeeld van een schaalbegrip is beroepsniveau. De gebruikte meetschalen zijn meestal slechts ordinaal, dat wil zeggen een ordening van 'laag' naar 'hoog'. Ordinale schalen worden in het schema aangeduid door een "O". Onder *elementbegrippen* worden de begrippen verstaan die men definieert als samenstellingen van basisbegrippen en schaalbegrippen. Een voorbeeld van een elementbegrip is gedragselement. Dat is gedefinieerd als de combinatie van gedragskenmerk en kenmerkniveau. Een voorbeeld van een gedragselement is "het beschikken over sociale intelligentie op niveau 4". De naamgeving "element" duidt er op dat het de kleinst mogelijke te onderscheiden en te meten 'hoeveelheden' gedrag zijn. De elementbegrippen vormen de 'legoblokjes' waarmee *compositiebegrippen* kunnen worden gedefinieerd. Een voorbeeld van een compositiebegrip is "persoonlijk competentieprofiel". Dat is gedefinieerd als een verzameling van gedragselementen en kenniselementen.

Het probleem van het matchen van vraag en aanbod op basis van competenties bestaat uit drie deelproblemen:

1. Het vaststellen van de competenties waaraan de werkbiedende behoefte heeft.
2. Het vaststellen van de competenties waarover de werkzoekende beschikt.
3. Het vaststellen van de mate waarin deze twee sets van competenties overeenkomen.

Het eerste deelprobleem wordt aangepakt in hoofdstuk 2, het tweede in hoofdstuk 3 en het derde in hoofdstuk 4. Als voorbeeld van een werkterrein, en ter illustratie van het Matchen op Competenties (MoC), is in bijlage A het ICT-werkterrein opgenomen. Via de ontologische definitie van het begrip activiteit in dat werkterrein wordt het ICT-werkterrein verbonden met de begrippen die in de hoofdstukken 2, 3 en 4 zijn gedefinieerd.



## 2. Begrippen aan de vraagzijde

In dit hoofdstuk worden de begrippen behandeld die te maken hebben met de vraagzijde van de arbeidsmarkt. Dat zijn dus de begrippen die een werkbiedende nodig heeft om zijn behoefte aan competenties uit te drukken. Het uitgangspunt daarbij is dat men een bepaald werkterrein voor ogen heeft (bijvoorbeeld de ICT, de verpleging of de tuinbouw) en dat voor dat werkterrein de zogeheten eenheid van *activiteit* is vastgesteld. Een voorbeeld van zo'n eenheid is te vinden in bijlage A.

### **Rolsoort**

*basisbegrip*

Iemand kan zelf de activiteiten in een werkterrein uitvoeren, maar hij/zij kan ook leiding geven aan de uitvoerenden, of hen adviseren, of bijvoorbeeld cursussen geven in de uitvoering van die activiteiten. Anders gezegd, men kan in verschillende soorten rollen bij een activiteit betrokken zijn, vandaar de term *rolsoort*. Voorbeelden van te onderscheiden rolsoorten zijn:

*uitvoerend*  
*leidinggevend*  
*adviserend*  
*controlerend*  
*onderwijzend*

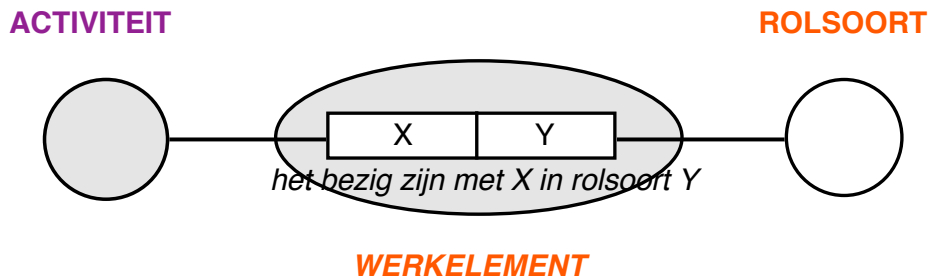
De gegeven lijst kan naar believen worden gewijzigd door er rolsoorten uit te verwijderen of er aan toe te voegen. Dat is een kwestie van standaardisatie. Er is een belangrijk verschil tussen de rolsoort uitvoerend en alle andere. Die is dat uitvoerend altijd betrekking heeft op een *object* (je verft bijvoorbeeld een deur of je dient medicijnen toe aan een patiënt), terwijl de andere rolsoorten betrekking hebben op een *persoon* in het werkterrein (je adviseert personen, je geeft leiding aan personen, je onderwijst personen enz.). Omdat deze rolsoorten zelf ook door personen worden uitgevoerd, kunnen ze gemakkelijk (en in principe eindeloos) in elkaar 'ingebod' worden: je kunt bijvoorbeeld leiding geven aan docenten die adviseurs opleiden om controleurs te adviseren.

### **Werkelement**

*elementbegrip*

De combinatie van een activiteit en een rolsoort definieert een eenheid die men een elementair stukje werk ofwel een *werkelement* zou kunnen noemen. Het werk, wat dat ook moge zijn, en door wie dat ook wordt gedaan, en of het nu uitvoerend of besturend of adviserend is, bestaat

altijd uit een of meer van deze werkelementen. De precieze definitie van werkelement staat in het volgende stukje ontologisch schema:



Basisbegrippen worden voorgesteld door gewone ellipsen of cirkels. De naam van het basisbegrip wordt ernaast geschreven in hoofdletters. In bovenstaand schema is ROLSOORT dus een basisbegrip. Om de begrippen aan de vraagzijde en aan de aanbodzijde snel te kunnen herkennen, gebruiken we de kleur oranje voor begripsnamen aan de vraagzijde en de kleur turquoise voor begripsnamen aan de aanbodzijde. Namen van begrippen uit het werkterrein worden paars gekleurd. Omdat een cirkel een bijzondere vorm van een ellips is, gebruiken we voortaan alleen nog de naam "ellips". Elementbegrippen worden voorgesteld door een ellips die grijs is ingekleurd. Het begrip ACTIVITEIT is een elementbegrip uit het werkterrein. De twee aaneengeschakelde rechthoeken (activiteit en rolsoort) stellen een conceptuele relatie voor tussen ACTIVITEIT en ROLSOORT. De betekenis ervan wordt uitgedrukt in de zin die eronder staat: "het bezig zijn met X in rolsoort Y". Door voor de relatierollen X (de activiteit) en Y (de rolsoort) een concrete activiteit en een concrete rolsoort in te vullen, ontstaat er een concrete relatie. De grote, grijs ingekleurde, ellips om de twee rechthoeken heen symboliseert de 'entificatie' van de relatie in een entiteit, namelijk WERKELEMENT. De cursivering van het woord duidt aan dat op deze plek in het ontologische schema het begrip wordt gedefinieerd. Als hetzelfde begrip wordt 'geleend' van een ander schema, wordt de cursivering weggelaten (zoals hierboven voor ACTIVITEIT is gedaan). Als we in de zin bijvoorbeeld voor X invullen "metselen van muren" en voor Y "leidinggeven" krijgen we het werkelement (in normaal Nederlands geformuleerd) "het leidinggeven aan het metselen van muren". Andere voorbeelden van werkelementen zijn:

*Het metselen van muren (uitvoerend dus).*

*Het leiding geven aan het metselen van bogen.*

*Het verlenen van thuiszorg aan cliënten (uitvoerend dus).*

*Het opleiden van verleners van thuiszorg.*

Door een aantal werkelementen samen te voegen, ontstaat er een pakket werk dat iemand als zijn of haar werk zou kunnen hebben.

Iemands functie zou bijvoorbeeld het verlenen van thuiszorg kunnen bevatten én het opleiden van verleners van thuiszorg.

### Moeilijkheidsgraad

*schaalbegrip (ordinaal)*

Het werk dat mensen doen (gedefinieerd dus als een aantal werkelementen) kan nog verder worden gedifferentieerd door verschillen in moeilijkheid in aanmerking te nemen. Die verschillen kunnen te maken hebben met de aard van het onderhanden object (vergelijk bijvoorbeeld een rechte muur met een kanteelmuur), met de omvang van de klus (een tuinmuur of een stadion), met het vereiste prestatieniveau (een voegmarge van 4 mm) enz. Laten we ons voor het gemak beperken tot drie niveaus:

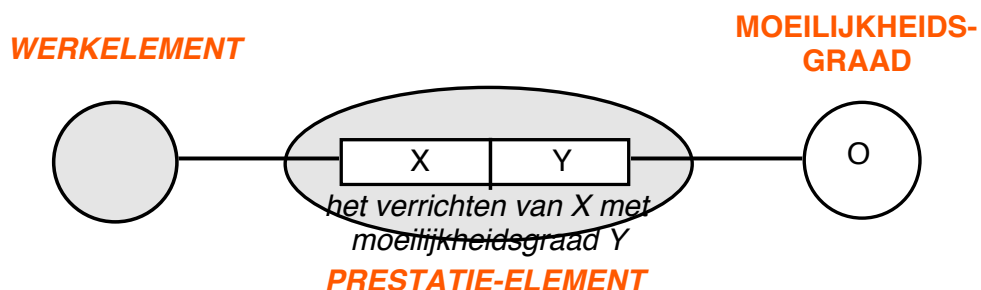
*hoog*  
*gemiddeld*  
*laag*

Dat zijn waarden op een zogeheten *ordinale schaal*, d.w.z er is altijd iets hoger of lager maar het is niet (goed) aan te geven hoeveel hoger of lager. Een ordinale schaal is gemakkelijk uit te breiden door steeds een waarde tussen twee bestaande waarden op te nemen of door een nieuwe laagste of hoogste waarde toe te voegen.

### Prestatie-element

*elementbegrip*

De combinatie van een werkelement en een moeilijkheidsgraad definieert een elementaire te leveren prestatie ofwel een *prestatie-element*. Onderstaand schema toont de definitie van het begrip prestatie-element. Het eerder gedefinieerde begrip WERKELEMENT wordt hierin dus 'geleend'.



De uitleg van het schema is soortgelijk aan de uitleg die hierboven is gegeven. Er is een relatie tussen WERKELEMENT en MOEILIKHEIDS-GRAAD, vastgelegd in de zin "het verrichten van X met moeilijkheidsgraad

Y". De entificatie ervan heet PRESTATIE-ELEMENT. Een voorbeeld van een prestatie-element (weer in gewoon Nederlands geformuleerd) is "*het verplegen van terminale patiënten met een gemiddelde moeilijkheidsgraad*". Andere voorbeelden van prestatie-elementen zijn:

*Het verplegen van dagverblijfpatiënten met een lage moeilijkheidsgraad.  
Het verplegen van terminale patiënten met een hoge moeilijkheidsgraad.*

*Het leiding geven aan metselwerk met een lage moeilijkheidsgraad (zoals een tuinmuur).*

*Het leiding geven aan metselwerk met een hoge moeilijkheidsgraad (zoals een stadion).*

## **Functie**

*basisbegrip*

Functies zijn aanduidingen van werk aan de vraagzijde van de arbeidsmarkt. Ze hebben vaak een algemeen karakter, zoals die van projectmanager, maar soms zijn ze specialistisch (bijvoorbeeld statistisch programmeur) of productspecifiek (zoals SAP-beheerder). Hoewel functies tamelijk bedrijfsgebonden kunnen zijn, hebben ze meestal een algemeen karakter, zodat het voor de werkzoekende mogelijk is ze onderling te vergelijken. Men zij er echter op bedacht dat er altijd interpretatieverschillen zullen zijn tussen organisaties (Java-programmeur bij de een hoeft niet precies dezelfde functie te zijn als Java-programmeur bij de ander).

Zoals we in hoofdstuk 4 zullen concluderen, zouden functies eigenlijk aanduidingen moeten zijn van te leveren prestaties en niet van gevraagde competenties. Bijvoorbeeld in plaats van een verpleegkundige 1<sup>e</sup> klas zou men moeten zeggen dat te te leveren prestatie is het in ploegendienst verplegen van terminale patiënten met een hoge moeilijkheidsgraad enz. Daarom is dat de manier waarop vacaturefunctieprofielen in dit boekje worden opgebouwd.

## **Functieniveau**

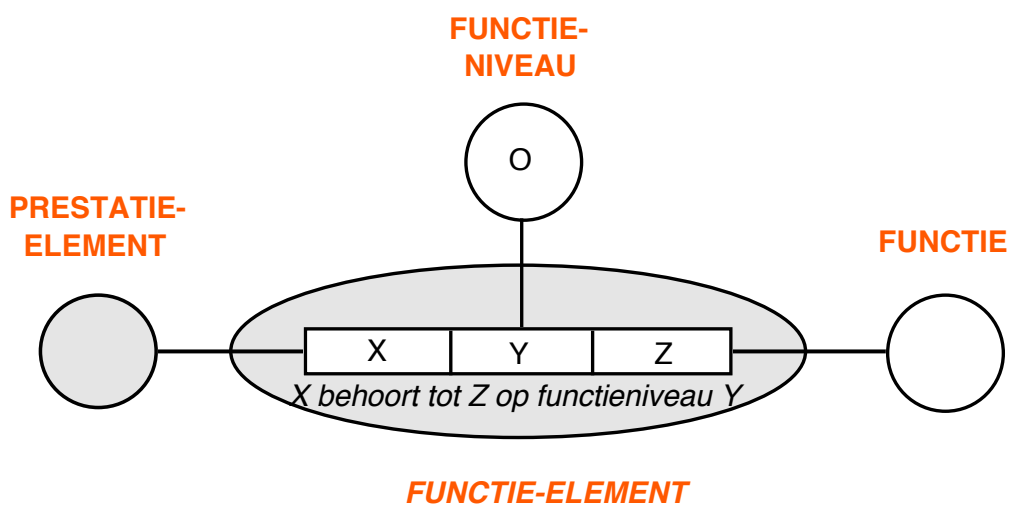
*schaalbegrip (ordinaal)*

Om de verschillen in niveau tussen functies met dezelfde naam uit te drukken gebruiken we de volgende ordinale schaal (die dus naar believen kan worden veranderd):

*expert  
senior  
professional  
junior*

**Functie-element***elementbegrip*

Een *functie-element* is gedefinieerd als de combinatie van een prestatie-element, een functie en een functieniveau. Een functie-element kan men zich het beste als volgt voorstellen: een bepaald prestatie-element (bijvoorbeeld het verplegen van dagverblijfpatiënten) is onderdeel van de functie verpleger op een bepaald functieniveau (bijvoorbeeld junior). Een prestatie-element kan onderdeel zijn van verschillende functies en op verschillende functieniveaus. In onderstaand schema is het begrip functie-element gedefinieerd (zonder verdere toelichting).





### 3. Begrippen aan de aanbodzijde

In dit hoofdstuk komen de begrippen aan de orde die van belang zijn voor de aanbodzijde van de arbeidsmarkt. Dat zijn dus de begrippen die een werkzoekende nodig heeft om zijn/haar competenties vast te stellen. Met *competenties* worden bedoeld de mogelijkheden die iemand in zijn/haar mars heeft. Daartoe behoren niet alleen kennisgebonden competenties, maar ook gedragsgebonden competenties, zoals sociale en emotionele vaardigheden. Die soorten competenties zijn vaak zelfs doorslaggevend voor het succesvol uitvoeren van taken.

Bij de wijze waarop werkzoekenden hun 'aanbod' beschrijven, is een soortgelijke opmerking te plaatsen als die we maakten bij de formulering van de vraag door de werkbiedenden. Vaak gebeurt dat in termen van prestaties, waar een formulering in termen van competenties zuiverder zou zijn. Wat je als werkzoekende namelijk eigenlijk, dat wil zeggen oorspronkelijk, biedt, zijn oplossingspotenties: je bent in staat bepaalde prestaties te leveren omdat je beschikt over de daarvoor benodigde competenties. Laten we eens kijken hoe competenties helder en precies kunnen worden gedefinieerd.

#### **Gedragskenmerk**

*basisbegrip*

De tijd dat met intelligentie alleen maar 'slimheid' werd aangeduid, is voorbij. Het is heel normaal om bijvoorbeeld van sociale en van emotionele intelligentie te spreken. Kenmerkend voor intelligentie is, dat men met een hoeveelheid ervan wordt geboren. Het is maar in beperkte mate mogelijk intelligentie op andere wijzen te verwerven. Daarmee is intelligentie duidelijk onderscheiden van kennis. Als een voorbeeld van een opsomming van intelligenties gebruiken we de zeven intelligenties die Howard Gardner<sup>1</sup> onderscheidt:

*taalkundige intelligentie*  
*logische intelligentie*  
*muzikale intelligentie*  
*bewegingsintelligentie*  
*ruimtelijke intelligentie*  
*sociale intelligentie*  
*intrapersoonlijke intelligentie*

---

<sup>1</sup> Howard Gardner: *Frames of Mind* (1983) en *Intelligence Reframed* (1999)

Wat voor intelligentie geldt, geldt voor alle gedragskenmerken: ze zijn in grote mate persoonsgebonden en slechts ten dele te verwerven door training en ervaring.

### **Kenmerkniveau**

*schaalbegrip (ordinaal)*

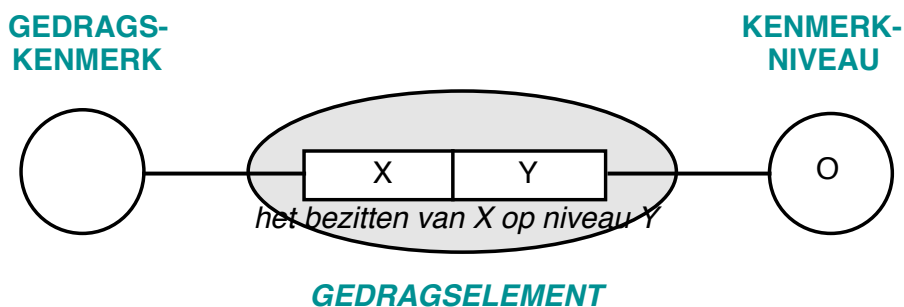
De verschillen tussen personen in het bezitten van een gedragskenmerk kunnen worden aangegeven op een ordinale schaal (logische intelligentie kan zelfs op een ratio-schaal worden aangegeven, maar voor bijvoorbeeld motivatie kan dat niet). Zoals ook al voor het begrip moeilijkheidsgraad is opgemerkt, is het aantal niveaus vrij te kiezen en zijn die gemakkelijk uit te breiden door er tussenniveaus in op te nemen. We onderscheiden voorlopig de volgende schaalwaarden:

*hoog*  
*gemiddeld*  
*laag*

### **Gedragselement**

*elementbegrip*

De combinatie van een gedragskenmerk en een kenmerkniveau vormt een elementaire hoeveelheid (potentieel) gedrag ofwel een *gedragselement*. De formele definitie in de ontologische taal luidt:



Ter illustratie volgen hier enkele voorbeelden van gedragselementen. Het bezitten van sociale intelligentie op een hoog niveau betekent dat men geschikt is voor het leveren van prestaties waarin het omgaan met mensen erg belangrijk is. Voorbeelden van zulke prestaties zijn het verkopen van systemen of het fungeren als projectmanager. Het bezitten van logische intelligentie op een gemiddeld niveau betekent dat men waarschijnlijk beter niet functies als analist en ontwerper kan ambiëren. Dat niveau kan echter wel voldoende zijn om als manager te functioneren.

**Kennissoort***basisbegrip*

Kennis is iets dat men zich eigen kan maken door te leren. Een voorwaarde daarvoor is dat men in voldoende mate over een aantal gedragskenmerken, zoals intelligentie(s), beschikt. Welke kennis men nodig heeft als professional hangt uiteraard sterk af van het vakgebied. In algemene zin is het wel mogelijk (en lijkt het ook nuttig) een onderscheid te maken tussen basiskennis en actualiteitskennis. *Basiskennis* is in de tamelijk permanente kennis van een vakgebied die men gewoonlijk 'op school', in de beroepsopleiding, leert. Daarnaast is het in elk vakgebied nodig bij te blijven, dat wil zeggen kennis op te doen van de actuele ontwikkelingen, van de 'state-of-the-art'. Laten we dat *actualiteitskennis* noemen. Deze kennis verkrijgt men door de vakliteratuur bij te houden en af en toe een bijscholingscursus te volgen.

Kennis verwerft men niet alleen in opleidingen of door zelfstudie, zij kan ook door ervaring worden opgebouwd. Ervaring is niets meer of minder dan het geleverd hebben van prestaties (vergelijk hoofdstuk 2). Iemands ervaring definiëren we als de optelsom van de afzonderlijke 'dingen' die hij/zij heeft gedaan, steeds in een bepaald werkelement en met een bepaalde moeilijkheidsgraad. Ervaring kan men niet meten, kennis gelukkig wel. Van ervaring kan wel worden gezegd dat zij indirect bijdraagt aan kennis.

**Kennisniveau***schaalbegrip (ordinaal)*

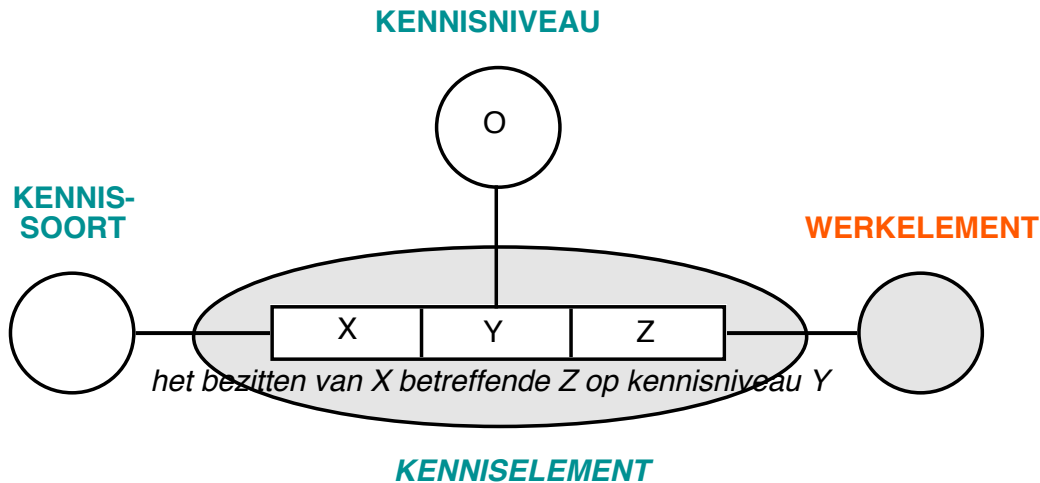
Net als bij kenmerkenniveau onderscheiden we niveaus van kennis. Als een voorzet onderscheiden we volgende schaalwaarden:

*hoog*  
*gemiddeld*  
*laag*

**Kenniselement***elementbegrip*

De combinatie van een kennissoort, een werkelement en een competentieniveau vormt een elementaire hoeveelheid kennis ofwel een *kenniselement*. Dat begrip wordt hieronder formeel gedefinieerd. Ter illustratie volgen hier enkele voorbeelden van kenniselementen. Het bezitten van basiskennis over de 'constructie' van bedrijfsprocessen op een laag niveau betekent dat men slechts elementaire kennis daarover bezit. Daarmee is men dus bijvoorbeeld niet in staat de prestatie te leveren van het met een gemiddelde of hoge graad van moeilijkheid ontwerpen van bedrijfsprocessen. Maar dat niveau zou voldoende kunnen zijn om leiding te geven aan personen die wél voor dat werk zijn

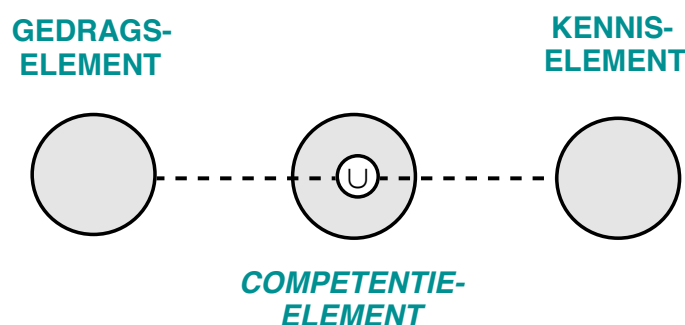
gekwalificeerd. Het bezitten van basiskennis én van actualiteitskennis over de constructie van netwerken kwalificeert iemand bij uitstek als ontwikkelaar ervan.



### Competentie-element

*elementbegrip*

Gedragselementen en kenniselementen vormen samen de competentie-elementen. Het begrip *competentie-element* is hieronder formeel gedefinieerd. Het schema drukt uit dat de verzameling competentie-elementen de (verzamelings-theoretische) vereniging is van de verzameling gedragselementen en de verzameling kenniselementen. De operatie van de (verzamelings-theoretische) vereniging wordt, zoals gebruikelijk, voorgesteld door het symbool "U".



### Beroep

*basisbegrip*

Een beroep is een specialisatie binnen een bepaald werkteerrein aan de aanbodzijde van de arbeidsmarkt. Voorbeelden van beroepen binnen het ICT-werkteerrein zijn: programmeur, IT-auditor, software engineer. Het is

aan de professionele gemeenschap te bepalen welke beroepen er moeten worden onderscheiden. In de geneeskunde bijvoorbeeld, wordt elk specialisme tot een apart beroep verheven.

### Beroepsniveau

*schaalbegrip (ordinaal)*

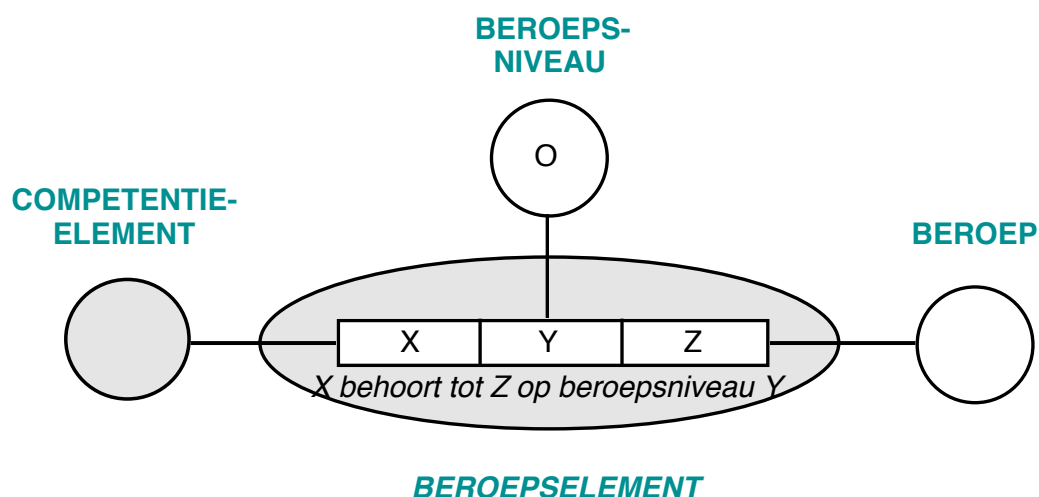
Tussen de beoefenaren van een beroep bestaan in het algemeen verschillen in bekwaamheid. Die verschillen berusten meestal deels op verschillen in kennis en deels op verschillen in gedragskenmerken. We zullen vooralsnog de volgende vier niveaus van bekwaamheid onderscheiden (maar ook deze schaal is te veranderen en uit te breiden):

*expert*  
*senior*  
*professional*  
*junior*

### Beroepselement

*elementbegrip*

De combinatie van een competentie-element met een beroep en een beroepsniveau heet een beroepselement. Een bepaald competentie-element kan dus tot verschillende beroepselementen behoren. Het begrip beroepselement is als volgt formeel gedefinieerd:



Ter illustratie gebruiken we de eerder vermelde competentie-elementen:

*Het beschikken over basiskennis van het functioneel ontwikkelen van informatiesystemen op een hoog niveau.*

Dit zou onderdeel kunnen zijn van het beroep applicatie-ontwerper op niveau expert, senior en professional, en van het beroep informatie-architect op niveau expert en senior.

*Het beschikken over sociale intelligentie als manager van het technisch beheren van netwerken op een gemiddeld niveau.*

Dit zou onderdeel kunnen zijn van het beroep beheermanager op niveau professional, en van het beroep netwerkbeheerder op niveau expert (omdat men van zo iemand mag verwachten dat hij/zij projecten kan leiden).

*Het bezitten van advieskennis op hoog niveau betreffende het functioneel implementeren van ERP-systemen.*

Dit zou onderdeel kunnen zijn van het beroep implementatie-adviseur op niveau expert en senior.

## 4. Matchen op Competenties

In dit hoofdstuk bespreken we het derde deelprobleem, zoals geformuleerd in hoofdstuk 1, te weten het vaststellen van de mate waarin de competenties waaraan een werkbiedende behoefte heeft overeenkomen met de competenties waarover een werzoekende beschikt. De werkwijze is dezelfde als die we in de vorige hoofdstukken hebben gevolgd: het bepalen van de begrippen die relevant (lijken te) zijn en die precies definiëren. We zullen eerst kijken naar hoe tot nu toe de afstemming van vraag en aanbod op de arbeidsmarkt verliep. Deze traditionele werkwijze bestaat uit het matchen van het *functieprofiel* van een vacature met iemands persoonlijke *beroepsprofiel*.

### **Functieprofiel**

*compositiebegrif*

Een *functieprofiel* is simpelweg gedefinieerd als een verzameling van functie-elementen. In het ontologische schema wordt dat uitgebeeld door een vetgerande ellips om de ellips van functie-element heen (zie bijlage B). Het enige gemeenschappelijke kenmerk van deze functie-elementen is dat ze dezelfde functie betreffen. Ze kunnen echter vrijelijk verschillen in de combinatie met prestatie-element en functieniveau.

### **Beroepsprofiel**

*compositiebegrif*

Zoals de functie-elementen de bouwstenen zijn voor functieprofielen, zo zijn de beroepselementen de bouwstenen van beroepsprofielen. En net zoals een functieprofiel een verzameling functie-elementen is, zo is een *beroepsprofiel* simpelweg gedefinieerd als een verzameling beroepselementen. Het enige gemeenschappelijke kenmerk van deze beroepselementen is dat ze hetzelfde beroep betreffen. In het ontologische schema wordt het begrip beroepsprofiel uitgebeeld door de vetgerande ellips om de ellips van beroepselement heen (zie bijlage B).

De traditionele wijze van matchen, dus van het met elkaar vergelijken van een functieprofiel en een beroepsprofiel is inherent moeilijk (en in principe eigenlijk onmogelijk) omdat je appels en peren vergelijkt. In functieprofielen zijn namelijk te leveren prestaties geformuleerd en in beroepsprofielen beschikbare competenties. Dat kan dus nooit automatisch, hetgeen ook niet gebeurde. Om de match te maken is namelijk iemand nodig die dat doet. Dat is in de praktijk de personeelsfunctionaris en het is zijn of haar vakmanschap dat bepaalt hoe goed de match wordt gedaan.

De nieuwe, moderne manier is dus het matchen op competenties (MoC). Dat bestaat uit het op elkaar afstemmen van iemands persoonlijke competentieprofiel en het prestatieprofiel van een vacature. Dat is duidelijk een 'fijnkorreliger' manier dan de traditionele manier. Immers, bij de traditonele manier worden 'hele' functies vergeleken met 'hele' beroepen, kleinere eenheden bestaan niet. Bij het matchen op competenties echter, is de kleinste eenheid het competentie-element en dat scheelt wel even! Voordat we het probleem dat ook daar nog bestaat bespreken, zullen we eerst de twee benodigde nieuwe begrippen definiëren.

### **Prestatieprofiel**

*compositiebegrif*

Een *prestatieprofiel* is simpelweg gedefinieerd als een verzameling van prestatie-elementen. In het ontologische schema wordt dat uitgebeeld door de vetgerande ellips om de ellips van prestatie-element heen (zie bijlage B). Een prestatie-profiel kan dus uit willekeurige combinaties van een werkelement en een moeilijkheidsgraad bestaan, dwars door de tradionele functies heen. Men kan zich zelfs voorstellen dat de daarin voorkomende werkelementen activiteiten betreffen die uit verschillende werkterreinen komen!

### **Competentieprofiel**

*compositiebegrif*

Een *competentieprofiel* is simpelweg gedefinieerd als een verzameling van competentie-elementen. In het ontologische schema wordt dat uitgebeeld door de vetgerande ellips om de ellips van competentie-element heen (zie bijlage B). Een competentieprofiel is dus een verzameling kenniselementen en gedragselementen. Daarbij is elke 'greep' toegestaan, dwars door de tradionele beroepen heen. En ook hier geldt dat de kenniselementen niet tot één werkterrein beperkt hoeven te zijn.

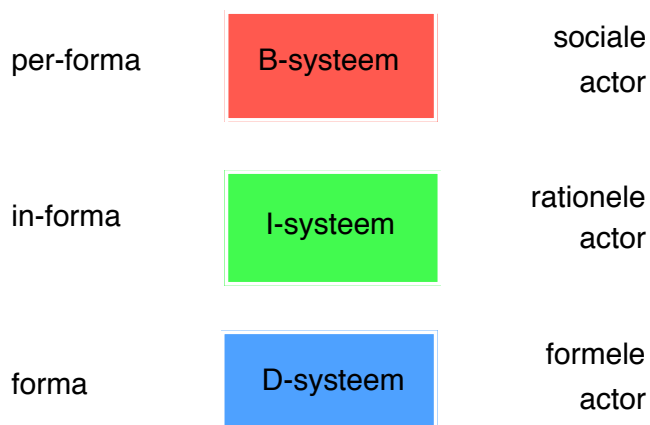
Is het probleem van het matchen op competenties daarmee opgelost? Nee, toch nog niet. Er zit namelijk net zo'n adder onder het gras als die we vonden bij het traditionele matchen. Die adder is nu dat competenties en prestaties inherent verschillende zaken zijn, weer appels en peren dus. De praktische oplossing is, dat het prestatieprofiel van een vacature wordt omgezet in een *vacature-competentieprofiel*. Maar dat vraagt om hetzelfde vakmanschap van de personeelsfunctionaris als nodig was voor het matchen van functies en beroepen. Pas als dat is gedaan, zijn er twee vergelijkbare soorten dingen, namelijk persoonlijke competentieprofielen en vacature-competentieprofielen. Het matchen van deze twee profielen kan gemakkelijk worden geautomatiseerd, maar de tussenstap van de personeelsfunctionaris niet!





## Bijlage A: Het ICT-werkterrein

ICT-ers houden zich bezig met informatieverwerking. Ze ontwerpen, bouwen, beheren en onderhouden informatieverwerkende systemen. Deze systemen ondersteunen menselijke actoren in organisaties of artificiële actoren in technische systemen (apparaten, machines enz.). We zullen ons vooral richten op de ondersteuning van menselijke actoren. Alles wat erover wordt gezegd is echter d.m.v. analogie eenvoudig te transponeren naar artificiële actoren in technische systemen.



*Figuur 1. De drie aspecten*

Aan de communicatie tussen menselijke actoren, en daardoor ook aan uitgewisselde informatie, zijn drie belangrijke aspecten te onderkennen, die in figuur 1 zijn uitgebeeld: het forma-, het informa- en het performa-aspect. Met het *forma*-aspect wordt alles wat met de verschijningsvorm van informatie te maken heeft bedoeld. Het omvat de in de semiotiek onderscheiden niveaus empirie en syntaxis. Als men door de 'forma-bril' naar een organisatie kijkt ziet men menselijke (en artificiële) actoren bezig zijn met het opbergen, verzamelen, transporteren en kopiëren van documenten (mono- en multimediaal). Deze actoren zijn de elementen van het D-systeem van de organisatie (de D van document en van data). De rol die mensen in dit perspectief spelen is die van formele actoren, vandaar dat ze zonder problemen kunnen worden vervangen door artefacten (ICT-systemen).

De inhoud van informatie is in het forma-aspect niet aan de orde, dat is het echter juist wel in het *informa*-aspect. Nu gaat het om wat er in de vorm is uitgedrukt. Dat aspect omvat de niveaus semantiek en pragmatiek uit de semiotiek, alles dus wat met betekenis heeft te maken. Als men

door de 'informa-bril' naar een organisatie kijkt ziet men menselijke actoren bezig zijn met het onthouden, zich herinneren en uitwisselen van kennis en ook met het berekenen van nieuwe kennis uit bestaande. Deze actoren zijn de elementen van het I-systeem van de organisatie (de I van informatie en intelligentie). In tegenstelling tot het D-systeem, is het I-systeem, vanwege de volledige abstractie van de vorm, puur conceptueel. De rol die de mens in het informa-aspect speelt is die van rationele actor, van iemand die weet en redeneert. Indien het informa-aspect en het forma-aspect van informatie-eenheden eenduidig aan elkaar worden gekoppeld (hetgeen in elke formele taal gebeurt), kunnen artefacten 'eruit zien als' rationele actoren. Hierin schuilt de kracht en het belang van ICT.

Resteert het *performa*-aspect. Daarmee wordt bedoeld wat de essentie van communicatie en informatie voor mensen is, wat ze ermee beogen en wat ze ermee tot stand brengen. Die essentie is het onderling aangaan en het nakomen van commitments. Kijkend naar een organisatie door de 'performa-bril' zien we sociale actoren, d.w.z. mensen die vanuit een bepaalde, aan hen toegekende, bevoegdheid en bijbehorende verantwoordelijkheid handelen. Dat handelen kan materieel handelen zijn, maar ook besluiten of oordelen. Kenmerkend voor het performatieve handelen (en fundamenteel verschillend van rationeel handelen) is, dat het de enige vorm van handelen is die originele, nieuwe feiten tot stand kan brengen. De mensen in hun rol van performa-actor zijn de elementen van het B-systeem van een organisatie (de B van bedrijf en van business). ICT-ers hebben in hun werk dus te maken met drie categorieën van systemen, te weten sociale systemen, rationele (conceptuele) systemen en formele systemen. In plaats van de benamingen B-systeem, I-systeem en D-systeem zullen we de meer gebruikelijke termen bedrijfsproces, informatiesysteem en infrastructureel systeem hanteren. Als eerste basisconcept introduceren we daarom hieronder de systeemsoort:

## **Systeemsoort**

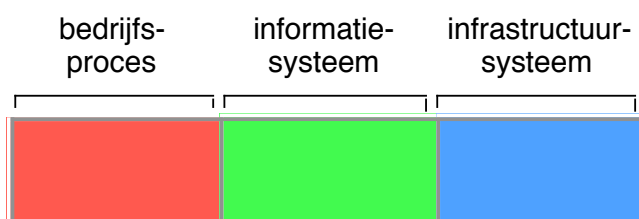
*basisbegrip*

Er worden drie systeemsoorten onderscheiden:

*bedrijfsprocessen*  
*informatiesystemen*  
*infrastructuur-systemen*

Elk van deze soorten heeft een eigen systeemcategorie: een bedrijfsproces is een sociaal systeem, een informatiesysteem is een rationeel systeem, en een infrastructureel systeem is een formeel systeem. De verdeling van de dimensie systeemsoort is totaal, d.w.z. er bestaan niet méér soorten systemen in de ICT. In figuur 2 zijn de drie soorten naast elkaar gezet. Het verband ertussen is af te leiden van de hiervoor gegeven uitleg. Een informatiesysteem (I-systeem) is

ondersteunend aan een bedrijfsproces (B-systeem), d.w.z. actoren in het bedrijfsproces gebruiken het informatiesysteem ter ondersteuning van hun onderlinge communicatieve acties (het aangaan en nakomen van commitments) en van hun besluitvorming. Een informatiesysteem is een conceptueel systeem, dat wordt gerealiseerd d.m.v. infrastructuursystemen (zoals interpreters, operating systems en database management systemen).



*Figuur 2 De systeemsoorten*

## **Systemoriëntatie**

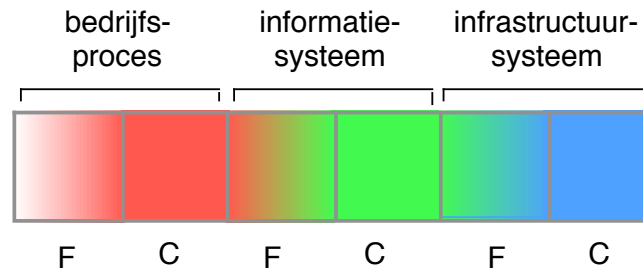
*basisbegrip*

We onderscheiden twee oriëntaties op systemen:

*functie-oriëntatie*

*constructie-oriëntatie*

Het innemen van de *functie*-oriëntatie betekent dat men is geïnteresseerd in de functionaliteit en/of het externe gedrag van een systeem. Het innemen van de *constructie*-oriëntatie betekent dat de aandacht is gericht op de constructie en/of de interne werking van een systeem. In plaats van de begrippen functie en constructie komt men ook wel eens teleologie en ontologie tegen. Een *teleologische* definitie van een systeem (het Griekse woord 'teleos' betekent doel) is een definitie van het doel c.q. de functie van het systeem. Een *ontologische* definitie is een definitie van wat het systeem eigenlijk is (het Griekse woord 'ontos' betekent werkelijk, dat wat bestaat), ofwel hoe het werkt. De verdeling van de dimensie systeemoriëntatie is totaal, d.w.z. er bestaan niet méér soorten oriëntaties.



*Figuur 3* *Systeemsoorten en systeemoriëntaties*

Figuur 3 laat de toevoeging van de dimensie systeemoriëntatie aan het model zien. Daarin representeert de letter "F" de functie-oriëntatie en de letter "C" de constructie-oriëntatie. In de 'functievlakken' lopen de kleuren van 'links' en van 'rechts' in elkaar over. Dat is gedaan om te laten zien dat het in functie-kolommen altijd gaat om het met elkaar verenigen van systemen die tot verschillende categorieën behoren. Helemaal 'links' van de bedrijfsprocessen ligt de missie van de organisatie, en helemaal 'rechts' van de infrastructuur-systemen bevinden zich de hardware-systemen.

### **Activiteitsoort**

*basisbegrip*

De activiteiten die door ICT-ers worden verricht met betrekking tot de hiervoor genoemde soorten systemen, kunnen worden ingedeeld in de volgende vier groepen:

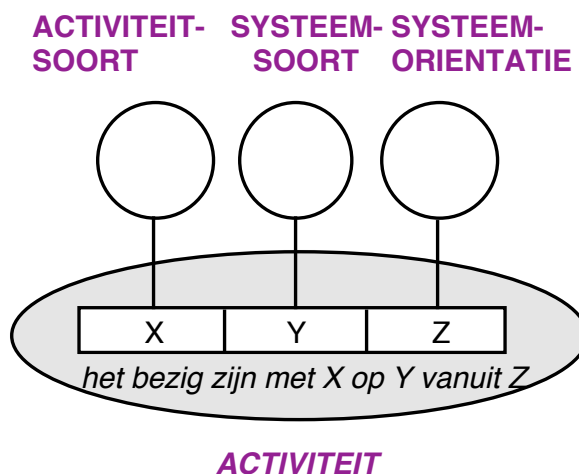
*architectureren*  
*ontwikkelen*  
*implementeren*  
*beheren*

Deze activiteitsoorten komt men ook in andere technische disciplines tegen, ze vormen tezamen de 'life cycle' van een systeem. De verdeling van de dimensie activiteitsoort is totaal, d.w.z. er bestaan niet méér soorten activiteit in het ICT-werkterrein. Anders gezegd, elke andere activiteitsoort is een specialisatie van een van deze groepen.

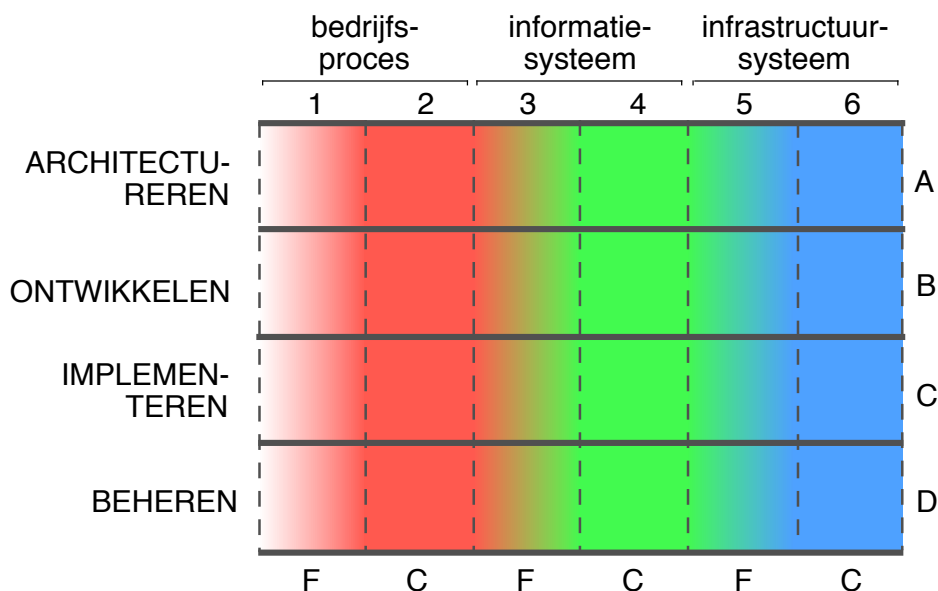
### **Activiteit**

*elementbegrip*

De combinatie van een activiteitsoort, een systeemsoort, en een systeemoriëntatie vormt een elementair stukje van het ICT-werkterrein, *activiteit* geheten. De formele definitie in een ontologische schema luidt:



In figuur 4 zijn de activiteitsoorten toegevoegd aan het model van figuur 3. Voorbeelden van activiteiten zijn het ontwikkelen van een informatiesysteem vanuit de functie-oriëntatie en het beheren van een infrastructureel systeem vanuit de constructie-oriëntatie. Omdat er maar twee systeemoriëntaties worden onderscheiden, kan de driedimensionale structuur van het ICT-werkterrein gemakkelijk in een plat vlak worden afgebeeld. Dat is in figuur 4 dan ook gedaan. De rijen en de kolommen zijn daarin tevens van een aanduiding voorzien. De rijen zijn genummerd van A tot D, en de kolommen van 1 tot 6.



*Figuur 4 Het ICT-werkterrein*

Een activiteit kan nu eenvoudig worden aangeduid door een letter en een cijfer, bijvoorbeeld A2, D3 en B6. Die duiden respectievelijk de volgende activiteiten aan:

*Het architectureren van bedrijfsprocessen vanuit de constructie-oriëntatie. Daaronder moet men verstaan het concipiëren van een architectuur voor bedrijfsprocessen waarin de actieve elementen daarin (de actoren) en de onderlinge coördinatie van hun activiteiten centraal staan.*

*Het beheren van informatiesystemen vanuit de functie-oriëntatie. Daarmee wordt bedoeld het in bedrijf houden van de functionaliteit van applicaties (ongeacht hun constructie). Tot de functionaliteit behoren niet alleen de 'knoppen waaraan men kan draaien' maar ook alle informatie waarnaar men kan vragen.*

*Het ontwikkelen van infrastructurele componenten vanuit de constructie-oriëntatie (bijvoorbeeld van netwerkserver of van database management systemen). Het gaat hier om het technisch ontwerp van die componenten, dus om het concipiëren van hun constructie. De elementen van zo'n constructie zijn hardware-componenten waarvan de functionele specificaties bekend zijn.*

Hieronder worden de basisbegrippen in het ICT-werkterrein nogmaals, maar nu uitgebreider besproken.

### *Architectureren*

Architectureren is het concipiëren van architecturen. Over de betekenis van de term architectuur heerst nog volop discussie. Het is niet de bedoeling in die discussie nu een standpunt te kiezen, maar we hebben wel een minimale, generieke definitie nodig. We zullen daarom de architectuur van een systeem definiëren als de essentie, die zich door middel van de realisatie manifesteert. Architectuur bestaat zowel in de functionele als in de constructieve oriëntatie. In de functionele oriëntatie is architectuur de *functionele essentie* die zich presenteert aan de gebruiker. Goede architectuur wordt gekenmerkt door 'vanzelfsprekendheid', door de aanwezigheid van mogelijkheden in de functionaliteit die je als gebruiker verwacht, en door de afwezigheid van zaken die je niet nodig hebt. Men denke bijvoorbeeld aan de functionele architectuur van de auto of aan die van een goed ontworpen applicatie (waarvan je intuïtief voelt wat je er mee kunt). Ook in de constructie-oriëntatie bestaat architectuur, en het belang van goede architectuur is daar minstens zo groot. In dat geval gaat het om 'vanzelfsprekendheid' voor de constructeur. Dat geldt voor constructeurs van auto's, maar ook voor die van hardware- en softwaresystemen. In de constructie van informatiesystemen is de vier-lagen-architectuur sterk in opkomst, en in die van infrastructuur-systemen is bijvoorbeeld de client-server-architectuur populair. In de constructie-oriëntatie staat architectuur dus voor *constructieve essentie*, voor de basis van de concrete constructie.

### *Ontwikkelen*

Onder ontwikkelen wordt zowel het ontwerpen van systemen verstaan (het bedenken en kiezen van een 'oplossing' voor een 'probleem') als het realiseren of bouwen van die oplossingen. Bij nadere beschouwing blijken tussen ontwerpen en bouwen, althans in het ICT-werkterrein, geen wezenlijke verschillen te bestaan. Dat komt door de immateriële aard van de dingen waar ICT-ers zich mee bezig houden. Die aard heeft tot gevolg dat realiseren of bouwen eigenlijk ook weer ontwerpen is, maar dan 'op een lager niveau'. Het doet er daarbij niet toe om welk soort systeem het gaat (bedrijfsproces, informatiesysteem of ICT-infrastructuur-component). Bij het ontwikkelen van informatiesystemen bijvoorbeeld, volgt traditioneel na functioneel en technisch ontwerp, het programmeren van de onderscheiden modules, en het opstellen van het database-schema. Bij nader inzien echter zijn beide soorten activiteiten het kiezen en samenstellen van geschikte ICT-infrastructurele componenten (denk aan objectklassen, routines en DBMS-en), en het bepalen van de juiste parameterwaarden van elke component. Het geheel wordt 'gerealiseerd' zodra de met die software geladen hardware in werking wordt gesteld. Het testen van een systeem wordt ook tot de activiteit ontwikkelen gerekend. Onder testen wordt daarbij niet alleen verstaan het verifiëren van de werking en de capaciteit conform het constructie-ontwerp, maar ook het valideren van de functionaliteit van het systeem.

### *Implementeren*

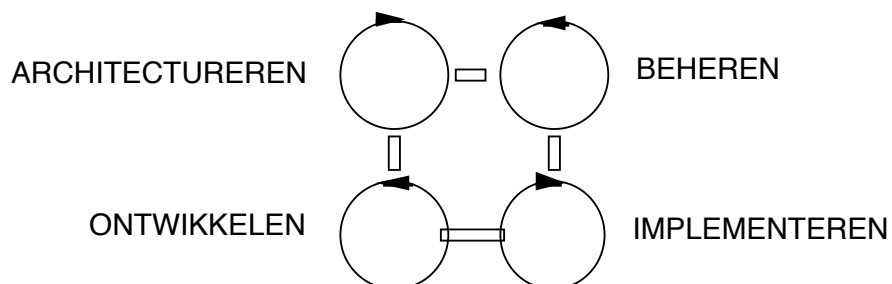
Implementeren is het in gebruik stellen van een ontwikkeld (en getest) systeem. Hoewel de tijd die daarmee is gemoeid, vergeleken met ontwikkelen of beheren, vaak relatief gering is, is het toch als een aparte, en aan de andere evenwaardige, activiteitsoort opgenomen in het model. De reden daarvoor is, dat implementeren om specifieke competenties vraagt. Wie ooit een systeem heeft ingevoerd, weet dat dat altijd gepaard gaat met onvoorziene problemen. Dankzij een goede voorbereiding kunnen die problemen meestal snel worden opgelost, maar soms niet. Een ander argument voor een gelijkwaardige plaats van implementeren, is dat het invoeren van grote standaard-applicaties (zoals ERP-pakketten) een omvangrijke klus is, die meestal de gehele rij C (met uitzondering van het vakje C1) omvat. Implementeren betekent ook iets nieuws, en dus iets 'vreemds', in een bestaande omgeving doen functioneren, en dat is in principe een complexe aangelegenheid. Het is in dat opzicht goed te vergelijken met het vervangen van een orgaan in het menselijk lichaam door een donor-orgaan of een kunstorgaan. Dat behoort tot de meest complexe chirurgische ingrepen. Tot implementeren behoort ook het voorbereiden van de omgeving waarin het systeem operationeel gaat worden (daar valt bijvoorbeeld het opleiden van gebruikers en het herinrichten van de organisatie onder), en de overgang of conversie van de oude naar de nieuwe situatie.

### *Beheren*

Onder beheren wordt het in bedrijf houden van een ingevoerd systeem verstaan. Het systeem in kwestie kan een bedrijfsproces, een informatiesysteem of een stuk ICT-infrastructuur zijn. Grofweg zijn er twee vormen van beheer te onderscheiden. De ene vorm is het op elk moment werkend en beschikbaar houden van de actuele functie en constructie van het beheerde systeem. De andere vorm van beheer is, ervoor te zorgen dat die actuele functie en constructie de 'optimale' zijn. Wat de functie betreft, betekent dat, dat men veranderende wensen en eisen signaleert en beoordeelt, en dat men zorgt voor het aanbrengen van die veranderingen. Kleine veranderingen brengt de beheerder meestal zelf aan, grote veranderingen leiden tot (her)ontwikkeling. In de constructie-oriëntatie houdt beheren in, dat een systeem bij veranderingen in de operationele omgeving in bedrijf blijft, maar ook dat men zulke veranderingen initieert indien de prestaties van het geheel daardoor verbeteren.

### *Samenhang tussen de activiteitsoorten*

Tussen de vier activiteitsoorten bestaat een samenhang die in figuur 5 is uitgebeeld. Ontwikkelen en implementeren hebben altijd betrekking op één systeem. Ze lopen ook 'synchroon', dat wil zeggen, aan het ontwikkelen van een systeem is het implementeren ervan gekoppeld. In de traditionele system life cycle valt implementeren helemaal ná ontwikkelen, in modernere life cycle modellen, zoals prototyping en RAD, overlappen ze elkaar. En bij het invoeren van ERP-pakketten is ontwikkelen eigenlijk ingebed in implementeren (als men tenminste het instellen van parameters als een vorm van ontwikkelen wil zien). Het rechtoekje dat de 'wielen' van ontwikkelen en implementeren met elkaar verbindt, beeldt deze sterke koppeling uit.



*Figuur 5 Samenhang tussen de activiteitsoorten*

Architectureren en beheren gaan in principe altijd over meerdere systemen tegelijk. Ze verlopen asynchroon, niet alleen ten opzichte van elkaar, maar ook ten opzichte van ontwikkelen en implementeren. Vandaar dat in figuur 5 drie 'losse' koppelingen zijn gelegd in de vorm van kleine rechthoekjes. Om de beeldspraak van de koppeling tussen ontwikkelen en

implementeren door te trekken, zou men kunnen zeggen dat architectureren en beheren door middel van magnetische sleepkoppelingen met elkaar en met de andere twee 'wielen' zijn verbonden: ze worden wel beïnvloed, maar indirect, en ze draaien ook autonoom.

### *Infrastructuur-systemen*

Het eerste en oudste deelgebied binnen het ICT-werkterrein betreft de opslag, het transport en de verwerking van data. Door de 'uitvinding' van de binaire code, en door de ontwikkeling van daarop gebaseerde coderingsstelsels (zoals ASCII en EBCDIC), werd het mogelijk elektrische, magnetische en optische media aan te wenden voor de opslag van ongekend grote hoeveelheden data, en voor het transport ervan met ongekend grote snelheden. Dezelfde 'uitvinding' maakte het ook mogelijk dat data met behulp van digitale elektronische apparaten zeer efficiënt kon worden verwerkt (algoritmische berekeningen en logische afleidingen). De inspanningen van degenen die zich op dit aandachtsgebied hebben toegelegd, hebben geresulteerd in zeer geavanceerde en breed toepasbare software-systemen. Voorbeelden van zulke systemen zijn tekstverwerkers, spreadsheetprogramma's, operating systems, compilers/interpreters, DBMS-en en programmatheken (inclusief de moderne variant van object libraries). Maar ook, om de 'C' in ICT recht te doen: netwerksystemen, internetdiensten, kortom systemen op alle zeven OSI-lagen. Deze systemen vormen tezamen de bouwstenen van wat tegenwoordig de (ICT-) infrastructuur wordt genoemd. In figuur 4 beslaan ze het blauwe vak. Het belangrijke gezamenlijke kenmerk van de 'blauwe' systemen, is dat ze *toepassings-onafhankelijk* zijn. Anders gezegd, de betekenis van de opgeslagen, getransporteerde of geproduceerde informatie voor een daarmee ondersteund applicatiedomein speelt geen rol. Alleen de vorm, de representatie in tekens, telt. Deze infrastructuur-systemen worden daarom gerekend tot de categorie van formele systemen, waartoe bijvoorbeeld ook wiskundige en de logische stelsels behoren.

### *Informatiesystemen*

De betekenis van informatie is juist wel van belang in het tweede aandachtsgebied. Gebruik makend van de infrastructuur-systemen als componenten, legt men zich in dit aandachtsgebied toe op de ontwikkeling van informatiesystemen ofwel applicaties. Zij worden in figuur 4 voorgesteld door het groene vak. Informatiesystemen zijn steeds specifiek voor een bepaald applicatiedomein. Centraal in dit deelgebied van het ICT-werkterrein staat de gedachte dat een informatiesysteem een afbeelding bevat van dat applicatiedomein. Aan de waarheidsgetrouwheid van deze afbeelding (waaronder zijn begrepen zaken als relevantie, precisie en actualiteit) worden informatiesysteemspecifieke eisen gesteld. Die eisen worden afgeleid van de kenmerken van de (bedrijfs)activiteiten waarbij de

informatie wordt gebruikt. Ter illustratie: de eisen die aan een informatiesysteem voor de luchtverkeersbegeleiding worden gesteld, zijn veel strenger dan de eisen die men stelt aan een systeem dat de ledenadministratie van de tennisclub ondersteunt. Informatiesystemen behoren tot de categorie van de rationele of conceptuele systemen. Daartoe behoren alle 'bedenksels', zoals bijvoorbeeld elke theorie over de fysische of de geestelijke wereld. In rationele systemen gaat het om het vergaren, bewaren, ordenen en afleiden van kennis, onafhankelijk van de wijze waarop die is vormgegeven. Hoewel men bij informatiesystemen meestal eerst denkt aan de ondersteuning van typische bedrijfsprocessen, behoort ook de meer individuele ondersteuning van werkers ertoe, zoals beslissingsondersteunende systemen en CAD-systemen, en ook alle enter- en infotainmentproducten.

### *Bedrijfsprocessen*

Het derde en jongste aandachtsgebied van het ICT-werkterrein wordt gevormd door de uitvoerende en bestuurlijke activiteiten in een organisatie. De gedachte die nu centraal staat, is dat een informatiesysteem onderdeel is van de inrichting van een bedrijfsproces; het ondersteunt namelijk de eigenlijke uitvoerder(s) daarvan. De ontwerper van een informatiesysteem dient daarom een gedegen inzicht te hebben in het ondersteunde bedrijfsproces. Daarnaast is het zo, dat de effectiviteit van de bijdrage van een informatiesysteem aan het functioneren van een organisatie, mede afhangt van de effectiviteit van het ondersteunde bedrijfsproces zelf. In figuur 4 beslaan bedrijfsprocessen het rode vak. Lange tijd zijn ze alleen bestudeerd binnen de organisatiekunde, de bedrijfskunde en de bedrijfseconomie, ze kunnen echter ook vanuit een meer informatiekundig perspectief worden bestudeerd en begrepen. Sterker nog, de daarmee geboden constructionele kijk op bedrijfsprocessen vormt de lange tijd ontbeerde brug tussen de organisatiekundige en de informatiekundige invalshoek (kolom 2 tussen 1 en 3 in figuur 4). Een bedrijfsproces is in de constructie-oriëntatie een aaneenschakeling van afspraken ofwel commitments tussen actoren, die zelf weer het resultaat zijn van communicatieve acties zoals verzoeken (opdrachten, orders, bestellingen etc.), beloftes (offertes, orderacceptaties, toezeggingen etc.), en verklaringen en aanvaardingen van resultaten van acties. Bedrijfsprocessen behoren tot de categorie van de sociale systemen, dat wil zeggen, van systemen waarvan de elementen mensen zijn, in hun rol van sociaal individu, die zich vanuit een toegekende bevoegdheid verantwoordelijk voelen voor hun daden.

Bedrijfsprocessen worden altijd strikt onderscheiden en gescheiden van technische processen, dat zijn de processen die zich in apparaten voltrekken. De scheiding tussen beide heeft geleid tot verschillende opleidingen en verschillende beroepsgroepen. Er zijn echter goede argumenten om wel het onderscheid te maken maar niet zo'n strikte scheiding. Om technische processen te begrijpen en te modelleren blijken

de modellen die voor bedrijfsprocessen worden gebruikt vaak heel effectief te zijn. Dat is ook niet zo verwonderlijk, want veel 'technische' processen zijn eigenlijk vergaand geautomatiseerde bedrijfsprocessen. Denk bijvoorbeeld aan geldautomaten en aan andere 'automatische' diensten. Met bedrijfsprocessen worden daarom alle processen bedoeld die door informatiesystemen worden onder-steund, hoewel de nadruk op bedrijfsprocessen in organisaties ligt.

#### *Functie-oriëntatie*

Een chauffeur is iemand die een auto bestuurt, die er in rijdt. Dat kan hij op grond van de functionele kennis die hij van de auto bezit. Die kennis is geworteld in een *black-box-model* van de auto. In zo'n model wordt de invloed van de omgeving op het systeem voorgesteld door (waarden van) invoervariabelen, en de response daarop door (waarden van) uitvoervariabelen. Het verloop van de waarden van deze variabelen in de tijd heet het gedrag van het model. Vandaar dat het verband tussen de invoer- en de uitvoervariabelen de gedragsfunctie wordt genoemd. Hoe dat gedrag tot stand komt, is vanuit de functie-oriëntatie gezien irrelevant. De chauffeur kan volstaan met te weten welke effecten veranderingen in de waarden van de invoervariabelen hebben op de waarden van de uitvoervariabelen. Voor-beelden van de invoervariabelen zijn de stand van het gaspedaal en de stand van het stuurwiel. Voorbeelden van de uitvoervariabelen zijn de snelheid van de auto en de richting waarin hij beweegt. Het hanteren van een *black-box-model* komt overeen met het innemen van de *functie-oriëntatie*.

#### *Constructie-oriëntatie*

Een monteur repareert een auto en onderhoudt hem. Dat doet hij op grond van de constructionele kennis die hij van de auto heeft. Die kennis is geworteld in een *white-box-model* van de auto. Het komt er op neer dat hij precies weet uit welke componenten de auto is opgebouwd, wat de *werking* is van elke component, en hoe de werking van de componenten gezamenlijk ervoor zorgt dat de functie van de auto wordt 'gerealiseerd'. Die componenten zelf kent de monteur in principe slechts vanuit de functie-oriëntatie. Zo is het voldoende dat hij de functie/het gedrag kent van accu's, O-ringen en V-snaren. Die hoeft hij niet zelf te kunnen maken. Anders gezegd, op eenzelfde wijze als de coureur gebruiker is van de auto, is de monteur gebruiker van deze componenten. Het hanteren van een *white-box-model* komt overeen met het innemen van de *constructie-oriëntatie*.

Wat voor auto's geldt, geldt ook voor de eerder besproken systemen: bedrijfsprocessen, informatiesystemen en infrastructuur-systemen. Om welk systeem dan ook te bestuderen, moet men er altijd eerst een model van hebben of maken, en het soort model bepaalt welk soort kennis (functionele of constructionele) men gebruikt of verwerft. Figuur 4 laat zien hoe functie en constructie afwisselend verband met elkaar houden.

De functie van een infrastructureel systeem, bijvoorbeeld een database managementsysteem, (kolom 5) is zodanig dat het systeem als element kan optreden in de constructie van een informatiesysteem (kolom 4). De functie van het database managementsysteem wordt 'gerealiseerd' in de constructie ervan (kolom 6). Vanuit kolom 4 'naar links kijkend' zien we dat de constructie van het informatiesysteem een 'realisatie' is van de functie van het systeem (kolom 3), en dat die functie zodanig is dat zij past in de constructie van het ondersteunde bedrijfsproces (kolom 2). Tenslotte, de constructie van dat bedrijfsproces is de 'realisatie' van zijn functie (kolom 1).



## Bijlage B: Het ontologisch schema voor MoC

Het ontologische schema is als een losbladige bijlage opgenomen. Hieronder staat de verklaring van symbolen in het schema.

Een ontologisch schema is een formele beschrijving van de begrippen die voor het begrijpen van een bepaalde wereld worden gebruikt. Voor de wereld van het matchen van competenties maken we onderscheid tussen basisbegrippen, schaalbegrippen, elementbegrippen en compositiebegrippen. De basisbegrippen en de schaalbegrippen vormen de basis waarop de elementbegrippen zijn gedefinieerd. Compositiebegrippen zijn op basis van elementbegrippen gedefinieerd.

Ellipsen (soms zuivere cirkels) stellen *basisbegrippen* voor. De naam van het basisbegrip staat in hoofdletters naast de ellips vermeld. Een voorbeeld van een basisbegrip is KENNISSOORT. De ellipsen met een letter er in representeren *schaalbegrippen*. Een schaalbegrip is een geordende verzameling waarden, die tezamen een meetschaal vormen, bijvoorbeeld KENNISNIVEAU. De letter stelt de schaalsoort voor: "O" voor ordinaal (dat is een ordening van laag naar hoog).

De aaneengeschakelde rechthoeken stellen *conceptuele relaties* voor. Een rechthoek representeert de rol in de relatie van het begrip waarmee het is verbonden. De betekenis van een conceptuele relatie is door middel van een nederlandse zin, die er onder staat, uitgedrukt. De hoofdletters X, Y en Z verwijzen naar de rollen. Bijvoorbeeld, "het bezitten van X op kennisniveau Y" drukt een relatie uit tussen KENNISSOORT en KENNISNIVEAU.

Een grijs ingekleurde ellips om een conceptuele relatie representeert een *elementbegrip*. De naam ervan staat in cursieve hoofdletters naast de ellips genoteerd. Bijvoorbeeld, KENNISELEMENT is gedefinieerd als het bezitten van een kennissoort op een kennisniveau. Een concreet voorbeeld van een kenniselement is < Engels, gemiddeld >, hetgeen staat voor het hebben van kennis van de Engelse taal op een gemiddeld niveau.

Een grijsgekleurde ellips om een kleine cirkel met het symbool "U" erin die door stippellijnen is verbonden met andere ellipsen representeert een *generalisatie*. De ellips stelt het *compositiebegrip* voor dat de generalisatie is van de begrippen die ermee zijn verbonden. Bijvoorbeeld, COMPETENTIE-ELEMENT is de generalisatie van KENNISELEMENT en GEDRAGSELEMENT; een competentie-element is dus ofwel een kenniselement ofwel een gedragelement.

Een vetgerande ellips om een ellips is een *verzameling* van de begrippen die door de omvatte ellips worden voorgesteld. De naam van het op deze wijze gedefinieerde *compositiebegrip* staat in cursieve hoofdletters naast

de omvattende, vetgerande ellips genoteerd. Bijvoorbeeld, een competentie-profiel is een verzameling competentie-elementen, en een functieprofiel is een verzameling functie-elementen.

Het ontologische schema is in drie primaire gebieden verdeeld, gescheiden door grijze stippellijnen: werkterrein, vraag en aanbod. Omdat *werkterrein* per definitie specifiek is, volstaan we met het opnemen van het 'geleende' elementbegrip ACTIVITEIT uit het werkterrein. De gebieden *aanbod* en *vraag* bevatten respectievelijk de begrippen die met de aanbodzijde en die met de vraagzijde van de arbeidsmarkt hebben te maken. Traditioneel vindt de afstemming van vraag en aanbod plaats door het vergelijken van beroeps-profielen en functieprofielen.

In theorie is *matchen op competenties* (MoC) een kwestie van het bij elkaar laten passen van een competentieprofiel en een prestatieprofiel. Daar ligt echter een probleem (dat er traditioneel ook al was!): het vergelijken van competentieprofielen en prestatieprofielen is als het vergelijken van appels en peren. De praktische oplossing van dit probleem is het vertalen van een prestatieprofiel (dus dat waar een werkbiedende eigenlijk om vraagt) naar een bijpassend *vacature-competentie-profiel* (VCP). Vervolgens wordt die 'gematcht' met een aangeboden *persoonlijk competentie-profiel* (PCP). Het probleem (aangeduid door het uitroep-teken in het schema) is daarmee echter slechts verschoven: hoe vind je een *peer* (een competentieprofiel) die bij een *appel* (een prestatieprofiel) past?

# Matchen op Competenties (MoC)

ontologisch schema

