

Inpassing van CASE-tools in LIA

ir. M.H. Wijnstok

Hoofd afdeling Methoden, technieken en tools
Stichting Technische en Fysische Dienst voor de Landbouw (TFDL-DLO)
Postbus 356, 6700 AJ Wageningen
Telefoon 08370-76600, telefax 08370-11312
e-mail m.h.wijnstok@tfdl.agro.nl

Inleiding

Sinds drie jaar wordt bij het Ministerie van Landbouw, Natuurbeheer en Visserij (LNV) gebruik gemaakt van Oracle*-CASE ter ondersteuning van de ontwikkeling van geautomatiseerde informatiesystemen. De inmiddels opgebouwde ervaring is vastgelegd in richtlijnen voor het gebruik van Oracle*CASE tijdens informatie-analyses (functioneel ontwerp) en systeemontwerpen (technisch ontwerp). Aanvankelijk werd alleen het systeemontwerp m.b.v. Oracle*CASE uitgevoerd, vervolgens ook de realisatie van informatiesystemen. Thans wordt het verder uitgebouwd naar de informatie-analyse.

Informatica organisatie

Het Ministerie van Landbouw, Natuurbeheer en Visserij (LNV) kent twee niveaus binnen de informatica-organisatie: het departementale niveau en de organisatie per directie of dienst.

Departementale organisatie

De Centrale Informatica Organisatie bestaat uit de afdeling Informatica van de directie Organisatie en Efficiency (O&E-I) en de afdeling Informatie Technologie van de Stichting Technische en Fysische Dienst voor de Landbouw (TFDL-IT). O&E-I verzorgt de beleidsvoorbereiding op het gebied van informatievoorziening en adviseert op het terrein van informatieplanning en -analyse en uitbesteding van systeemontwikkelingsactiviteiten. TFDL-IT is verantwoordelijk voor marktverkenning, testen en technische advisering omtrent apparatuur en programmatuur, alsmede ondersteuning van de lokale

verwerkingsorganisaties. De sectie Methoden, Technieken en Tools (MTT) van TFDL-IT adviseert over de binnen LNV gehanteerde methode van systeemontwikkeling, in het bijzonder op het terrein van technisch ontwerp, realisatie en acceptatietesten. Voor de daarmee gepaard gaande technieken en tools (waaronder Oracle*-CASE) worden richtlijnen (o.a. voor applicatie-ontwikkeling en gebruik van CASE) onderhouden op basis waarvan een toetsing van de kwaliteit van op te leveren producten plaatsvindt.

Organisatie

Directies en diensten dragen zelf de verantwoordelijkheid voor de inrichting van de eigen informatievoorziening. M.b.v. informatieplannen bepalen zij de behoefte aan geautomatiseerde systemen en plannen zij de ontwikkeling, het beheer en de exploitatie ervan. De decentrale opzet van de computercentrumfunctie heeft geresulteerd in ongeveer 60 rekencentra van uiteenlopende omvang, elk met een lokale verwerkingsorganisatie.

Landbouw Informatica Aanpak

De Landbouw Informatica Aanpak (LIA) omvat de standaarden voor (geautomatiseerde) informatievoorziening betreffende technische infrastructuur, methoden en technieken voor informatieplanning, ontwikkeling en exploitatie alsmede kwaliteitszorg. De standaarden zijn vastgelegd in een reeks handboeken, die gezien de ontwikkelingen in het vakgebied aan voortdurende bijstelling onderhevig zijn. De Landbouw Informatica Aanpak valt onder ver-

antwoordelijkheid van de Centrale Informatica Organisatie.

Technische infrastructuur

Standaardisatie van hardware en systeemsoftware is één van de instrumenten om de informatisering en automatisering binnen de diversiteit aan directies en diensten beheersbaar te houden. Al in een vroeg stadium is gekozen voor de DEC-VAX-omgeving met VAX/VMS als besturingssysteem. Daarnaast wordt gebruik gemaakt van PC's met MS-DOS. Als databasemanagementsysteem (DBMS) is sinds 1982 'Oracle' in gebruik. Momenteel zijn ca. 300 VAX'en operationeel, vrijwel allemaal opgenomen in het Agronet-netwerk van LNV. Oracle en applicaties worden op het merendeel daarvan geëxploiteerd.

LIA-ontwikkelmethodiek

De LIA-methode is een variant op de Information Engineering Methodology van James Martin Associates en omvat in hoofdlijnen de fasen Informatieplanning, Informatie-analyse (IA), Systeemontwerp (SO), Realisatie (RE), Testen en Invoering. De Informatieplanning schetst de gewenste informatievoorziening als basis voor de ontwikkeling van onderling samenhangende informatiesystemen. Tijdens de IA wordt een informatiegebied gedetailleerd op logisch niveau, d.w.z. onafhankelijk van technische en organisatorische omgeving. De IA resulteert in een volledig functioneel ontwerp, dat tijdens het SO wordt vertaald in een technisch ontwerp. De RE-fase levert vervolgens de programmatuur voor het ontworpen informatiesysteem. Nadat het systeem getest en geaccepteerd is door de gebruikers, wordt het in de organisatie ingevoerd.

Voor iedere LIA-fase wordt een afzonderlijke projectorganisatie ingericht, waarin de besturende, uitvoerende en kwaliteitsborgingsfunctie tot hun recht moeten komen. De uitvoerende taken zijn gewoonlijk gelegd bij een projectgroep. Essentieel voor de projectgroep is, dat de relevante disciplines met voldoende deskundigheid zijn vertegenwoordigd en dat doorstroom van kennis en ervaring naar de eerstvolgende fase gewaarborgd is.

De fasen SO en RE worden doorgaans uitbesteed aan een extern bureau. Randvoorwaarden voor uitbesteding zijn:

- voldoende kennis en ervaring bij de externe partij van/met de LIA-methode;
- eenduidige en toetsbare richtlijnen voor applicatie-ontwikkeling met een systematische kwaliteitstoetsing op de op te leveren producten en de projectaanpak;
- beschikbaarheid van een calculatiemethodiek voor het ramen van de omvang van de uit te besteden werkzaamheden (o.a. functiepuntberekening).

Het LIA-handboek beschrijft voor elke fase welke producten opgeleverd dienen te worden. Indien relevant, wordt in aparte handboeken voorgeschreven welke ontwikkeltools hierbij gehanteerd dienen te worden en op welke wijze er mee omgegaan dient te worden. Het Handboek Technische Richtlijnen voor Applicaties (TRA) bevat de richtlijnen voor de fase SO en RE in de applicatie-ontwikkeling. In het Handboek Technische Richtlijnen voor CASE (TRC) zijn richtlijnen vastgelegd voor het gebruik van Oracle*CASE tijdens IA, SO en RE. Dergelijke richtlijnen vormen de basis voor kwaliteitscontrole. Het houden van systematische kwaliteits-reviews moet zorgdragen voor een methodisch correcte werkwijze en producten van voldoende inhoudelijke en technische kwaliteit.

Introductie CASE-tools

Oracle*CASE was voor LNV niet de eerste kennismaking met CASE-tools. IEW was al enige jaren in gebruik tijdens de fasen Informatieplanning en Informatie-analyse. Teneinde ook de volgende LIA-fasen te kunnen ondersteunen met geautomati-

seerde hulpmiddelen is gezocht naar een meer complete CASE-omgeving. LNV heeft Oracle*CASE als ontwikkelhulpmiddel gekozen. Oracle*CASE bestaat uit CASE*Dictionary, de repository (=database) waarin de gegevens van de informatiemodellen worden opgeslagen, CASE*Designer voor de presentatie van die gegevens in diagrammen en matrices en CASE*Generator voor de generatie van programmatuur. CASE*Dictionary biedt een breed scala aan overzichten. Bovendien bestaat de mogelijkheid om maatwerk-rapportages te definiëren, aangezien de repository in een Oracle-database is opgeslagen.

Als motivatie van de keuze gelden de volgende overwegingen:

- alle fasen van systeemontwikkeling worden ondersteund, zodat conversie van een upper CASE-tool(gericht op de eerste fasen van het systeemontwikkelingstraject) naar een lower CASE-tool (gericht op de laatste fasen van het systeemontwikkelingstraject) met onvermijdelijk informatieverlies vermeden wordt;
- de LIA-methode en producten worden in voldoende mate ondersteund, zonder dat aanpassing van de methode vereist is;
- verwacht mag worden dat nieuwe functionaliteit in Oracle-ontwikkeltools snel door CASE*Generator wordt geïmplementeerd;
- de userinterface (SQL*Forms-schermen) kan als bekend beschouwd worden;
- het past in de beschikbare technische infrastructuur en vergt geen extra beheersinspanning van de verwerkingsorganisatie.

In 1990 is door LNV de bruikbaarheid van Oracle*CASE voor de LIA-methode geëvalueerd in een pilot project voor SO. Binnen het projectteam was vooraf geen ervaring met Oracle*CASE aanwezig. Twee medewerkers van de Centrale Informatica Organisatie zijn nauw bij de pilot betrokken geweest. Geconcludeerd werd dat de door LIA vereiste producten in voldoende mate worden ondersteund en dat de werkwijze past binnen het kader van de

LIA-fasering. Van de hulpmiddelen wordt verwacht dat ze een kwaliteitsverhogende werking hebben vanwege de consistentiebewaking, significante produktiviteitsverbetering in het ontwikkelproces wordt vooralsnog niet verwacht.

Uiteraard zijn ook minpunten geconstateerd. Zo zijn de rapportages niet alle even overzichtelijk (sommige zijn te uitgebreid) en er is veel overlap. In een aantal gevallen ontstaat een onleesbare vorm van Nederlands en Engels: de relatie "elk bedrijf heeft één of meer percelen" verschijnt op het rapport als "EACH bedrijf MUST BE heeft ONE OR MORE percelen". De toenmalige versie van CASE*Generator had als bezwaar dat na generatie aangebrachte handmatige aanvullingen in de programmatuur niet door het tool herkend en bewaard werden.

De voordelen wegen echter op tegen deze nadelen. De aanbeveling luidde dan ook om Oracle*CASE binnen LNV in te voeren, mede gezien de aangekondigde verbeteringen in de volgende versie van het tool. Voorts heeft de pilot geresulteerd in een overzicht waarin voor de fasen IA en SO per LIA-product wordt weergegeven welke CASE-rapporten de gewenste informatie geven. Inmiddels zijn verscheidene projecten m.b.v. Oracle*CASE uitgevoerd, in eerste instantie voornamelijk SO-projecten en momenteel ook IA-projecten.

CASE en LIA

De toepassing van Oracle*CASE binnen LNV is zoals eerder vermeld slechts zinvol indien het past in de LIA-methode. Oracle*CASE is gebaseerd op de door Oracle gedefinieerde CASE*Method. Het zal niemand verbazen dat de methoden van elkaar verschillen op het vlak van de fasering en de producten en dat beide een eigen terminologie hanteren. Met name de grenzen tussen fasen lopen niet helemaal synchroon. In het algemeen geldt bovendien dat fasegrenzen door de inzet van CASE-tools gemakkelijk kunnen vervagen. Het uitbestedingsbeleid van LNV vereist echter duidelijk afgebakende fasen, waarbij op grond van de opgeleverde producten besluitvorming dient plaats te vin-

den alvorens de volgende fase te starten. Door per fase te definiëren welke stappen in Oracle*CASE overeen komen met welke produkten (TRC), wordt aan die voorwaarde voldaan.

Niet alle LIA-produkten worden (volledig) ondersteund door Oracle*CASE. Enkele voorbeelden zijn: toestandovergangsdiagrammen van entiteitstypen, geclusterde Create-Read-Update-Delete-matrices, afhankelijkheidsdiagrammen van activiteiten inclusief beschrijvingen, scherm- en rapport-layouts en projectplannen. Overigens wordt er niet naar gestreefd uitsluitend CASE-produkten op te leveren. Een Informatieplanning- en IA-rapport dient een voor niet-informatici duidelijk leesbaar geheel te zijn, dat veel beter met een tekstverwerker kan worden opgesteld, zolang Oracle*CASE geen adequate tekstverwerkingsfaciliteiten biedt. De CASE-overzichten spelen daarin een ondersteunende rol in de vorm van bijlagen. In het SO is minder verklarende tekst nodig en kunnen de CASE-overzichten een meer overheersende rol vervullen. Een eindrapport waarin de samenhang wordt aangegeven blijft desondanks een vereiste.

De vraag is nu hoe discrepanties tussen LIA en Oracle*CASE behandeld moeten worden. Daarbij zijn de volgende overwegingen van belang geweest. Het aanpassen van een methode aan de (on-)mogelijkheden van tools verdient o.i. niet de voorkeur. Tools evolueren immers veel sneller dan methoden en voortdurende verandering van de methodiek ondermijnt het streven naar beheersbare ontwikkeling van informatiesystemen. Het aanpassen van tools is daarentegen veelal slechts in beperkte mate mogelijk via instelling van parameters of preferences. Waar ontwikkelaars zelf objecten kunnen definiëren en toevoegen aan de in het tool beschikbare objecten, ontstaan verschillende versies van het tool met negatieve effecten op het beheer en de uitwisselbaarheid van gegevens. Produkten buiten de CASE-omgeving om gemaakt bemoeilijken de consistentiebewaking, de onderhoudbaarheid en het (versie-)beheer. Creatief gebruik van de mogelijkheden van het tool, waarbij functies niet op de daarvoor bedoelde wij-

ze gebruikt worden, vereist een gedegen kennis van het tool om eventuele beïnvloeding van andere functies te kunnen voorstellen.

Aan alle genoemde alternatieven kleven zowel voor- als nadelen. In ieder geval heeft Oracle*CASE niet geresulteerd in veranderingen in de volgens de LIA-methode op te leveren produkten. Waar mogelijk zullen LIA-produkten met door Oracle*CASE gegenereerde overzichten en programmatuur worden ingevuld. Het is niet toegestaan daartoe eigen objecten te definiëren m.b.v. de "user extensibility"-optie. Dat impliceert tevens dat niet door Oracle*CASE ondersteunde produkten met een andere tool gecreëerd dienen te worden. In het Handboek Technische Richtlijnen voor CASE is dit in detail beschreven.

Kwaliteitszorg

Een belangrijke doelstelling van LIA is beheersbare ontwikkeling van kwalitatief goede informatiesystemen. Daarvoor is in de eerste plaats een werkwijze noodzakelijk die de kwaliteit vanaf het begin als onlosmakelijk aspect van het ontwikkelproces beschouwt en gericht is op het vroegtijdig ontdekken en herstellen van onjuistheden. De LIA-methode zelf is de belangrijkste pijler van het kwaliteitssysteem. Daarnaast zijn een adequate projectorganisatie en goed opgeleide projectmedewerkers een voorwaarde. Het per fase toetsen van de kwaliteit van opgeleverde produkten is de laatste pijler. De projectresultaten worden getoetst aan normen en richtlijnen die zijn vastgelegd in diverse handboeken.

Normen en richtlijnen

Het technisch ontwerp en de realisatie van informatiesystemen moeten voldoen aan de richtlijnen zoals gedefinieerd in het Handboek Technische Richtlijnen voor Applicaties (TRA). De TRA is ontstaan vóór de invoering van CASE-tools en gaat uit van handmatig ontwerpen en programmeren. Doelstellingen zijn verhogen van de onderhoudbaarheid, vereenvoudigen van het beheer door uniformiteit, verkleinen van de kans op ernstige fouten, vergroten van de integreerbaarheid

met andere systemen en verhoging van de efficiëntie door gebruik van een beperkt aantal ontwikkeltools. Het handboek gaat gepaard met geautomatiseerde procedures, waarvan een aantal bedoeld is om in de te ontwikkelen applicatie op te nemen, zoals templates. De overige procedures vergemakkelijken de realisatie, zoals procedures voor het genereren van documentatie. Omdat nog steeds een aanzienlijk deel van de werkzaamheden met name tijdens de Realisatie handmatig gebeurt, is besloten de TRA niet aan te passen en voor CASE-tools een afzonderlijk handboek (TRC) op te stellen.

In het Handboek Technische Richtlijnen voor CASE wordt voorgeschreven op welke wijze ontwikkelaars met Oracle*CASE dienen om te gaan om tot de gewenste produkten te komen. Voor alle te specificeren objecten is aangegeven welke gegevens verplicht of optioneel zijn of zelfs niet ingevuld dienen te worden. Uitgangspunt daarbij is dat het dagelijks beheer en gebruik van applicaties die aan deze richtlijnen voldoen niet verschilt van applicaties die zonder Oracle*CASE zijn vervaardigd. Verschil is vooral merkbaar voor degenen die bij de ontwikkeling van of onderhoud op applicaties betrokken zijn. Voor deze doelgroepen geldt dat waar mogelijk de default-waarden die Oracle*CASE biedt worden gevolgd, zodat extra inspanning kan worden vermeden. Ook bij de TRC worden enkele geautomatiseerde procedures meegeleverd. Daarbij kan gedacht worden aan procedures die de definitie van entiteitstypen, attributen en processen meenemen naar de bijbehorende tabel, kolom en procedure. Oracle*CASE verzorgt dit zelf niet.

Een conceptversie van de TRC is opgesteld nadat bij een aantal projecten ervaring is opgedaan met Oracle*CASE. Dit concept wordt momenteel ter toetsing, als praktijktest, bij enkele projecten toegepast. De hieruit voortvloeiende opmerkingen en aanbevelingen worden verwerkt alvorens een definitieve versie van het handboek wordt vrijgegeven. Ten behoeve van de educatieve werking van het handboek is de eerste versie van de TRC uitvoeriger dan wellicht noodzakelijk zou zijn.

Het zal duidelijk zijn dat een aantal TRA-richtlijnen niet verenigbaar is met de toepassing van CASE-tools. Indien tijdens het ontwikkelproces gebruik wordt gemaakt van Oracle*CASE, dan is de TRC leidend. Overigens is bij het opstellen van de TRC rekening gehouden met bestaande richtlijnen en zo mogelijk zijn deze overgenomen. Voor handmatig uitgevoerde activiteiten blijft de TRA onverkort van kracht. Op grond van de ontwikkeling van CASE-tools mag verwacht worden dat in de toekomst (vrijwel) alle IA-, SO- en RE-projecten binnen LNV m.b.v. Oracle*CASE zullen worden uitgevoerd. Het ligt voor de hand om dan de TRA en de TRC samen te voegen tot één consistent handboek, waarin de CASE-specifieke richtlijnen compacter beschreven zijn dan in de huidige TRC.

Kwaliteits-reviews

Tijdens het merendeel van de LIA-fasen worden op vaste momenten en op gedefinieerde producten reviews uitgevoerd door, niet bij de projecten belanghebbende, review teams. Een IA kent vier reviews: de eerste vindt plaats op het projectplan, de tweede op het informatiemodel, de derde op het geverifieerd informatiemodel en de vierde op het eindrapport. Het SO wordt getoetst in drie reviews: de eerste eveneens op het projectplan, de tweede op de fysieke gegevensstructuur en de procedure-ontwerpen, en de derde review op de eindproducten. Ook voor de RE gelden drie reviews: de eerste op het projectplan, de tweede op de eerste programma's en de laatste op het eindproduct. Afhankelijk van de complexiteit, omvang en verloop van het project kunnen extra reviews worden toegevoegd. Aandachtspunten voor kwaliteits-reviews zijn:

- de haalbaarheid van de projectdoelstellingen;
- de methodische en technische kwaliteit van opgeleverde producten;
- de manier waarop de projectdocumentatie is geproduceerd;
- de plausibiliteit van de inhoud;
- de projectplanning en -voortgang;
- de afstemming met departementale ontwikkelingen en eerder opgestelde plannen.

Elke review resulteert in een rapportage waarin de beoordeling wordt onderbouwd en zonodig adviezen of aanbevelingen worden gegeven voor verbetering van de getoetste producten. Een review wordt altijd uitgevoerd op basis van bestaande normen en richtlijnen. De handboeken waarin deze beschreven zijn, zijn vanaf het begin van het project ter beschikking aan de projectgroep. De bevindingen van de reviewer worden besproken met de projectleider en indien nodig projectleden, waarbij afspraken worden gemaakt over de afhandeling van geconstateerde knelpunten.

De inzet van Oracle*CASE heeft geen aanleiding gegeven tot wijziging van de review-procedures. Het aantal reviews per fase en de review-momenten zijn gelijk gebleven. De inhoud van de reviews is op een aantal aspecten wel aangepast. Dat geldt met name voor SO- en RE-reviews. Bij SO-reviews dient de CASE-repository aan de reviewer opgeleverd te worden. Hierop worden geautomatiseerde controles uitgevoerd, waarmee potentiële knelpunten worden gesignaleerd. Deze controleprocedures leveren overzichten van bijvoorbeeld tabellen zonder bijbehorend entiteitstype en entiteitstypen zonder bijbehorende tabellen als ook verschillen in datatype tussen kolommen en bijbehorende schermvelden. De bevindingen kunnen m.b.v. Oracle*-CASE nader geanalyseerd worden om te bepalen of er werkelijk sprake is van een onjuistheid. Op deze wijze kan een veel systematischer controle op kwaliteit plaatsvinden dan wanneer uitsluitend papieren documenten beschikbaar zijn. Hiervan mag een algemene verbetering van de kwaliteit verwacht worden. Voor RE-reviews geldt dat door CASE*Generator gegenereerde programmatuur als correct kan worden beschouwd, anders is de inzet van dit hulpmiddel niet gerechtvaardigd. Alleen de handmatig toegevoegde code wordt getoetst. Deze dient derhalve duidelijk herkenbaar te zijn. Ook voor RE-reviews beschikken de reviewers over geautomatiseerde procedures die systematische controle mogelijk maken. Oracle*-CASE heeft hierop vooralsnog geen invloed gehad.

Vooruitgang?

Terugkijkend op de afgelopen drie jaar kan gesteld worden dat de kwaliteit van individuele systeemontwerpen verbeterd is. De indruk bestaat dat dit voor een groot deel te danken is aan de toegenomen ervaring met de LIA-methode bij externe bureaus en dat hieraan door de inzet van Oracle*CASE slechts voor een gering deel is bijgedragen. Met name de grotere informatiesystemen, die uit meerdere deelsystemen zijn opgebouwd profiteren van Oracle*CASE. Voor kleine systemen is dat in veel mindere mate het geval.

Oracle*CASE leidt tot een verhoging van de uniformiteit van de eindproducten, wat vanuit de optiek van de Centrale Informatica Organisatie over heel LNV uiteraard een kwaliteitsverhogend effect heeft. Tevens kan een forse toename in het aantal pagina's van een SO-eindrapportage worden geconstateerd door toevoeging van CASE-overzichten. Of dat een verbetering is, is op zijn minst twijfelachtig.

Voor de realisatie van applicaties heeft CASE*Generator in het bijzonder aanzienlijke gevolgen gehad. De tijdens de SO-fase vastgelegde specificaties in de repository bieden de mogelijkheid om programmatuur te genereren. De aldus verkregen programma's kunnen verder bewerkt worden om ze te vervolmaken. De verhouding tussen gegenereerde code en handmatig toegevoegde code zal naar verwachting steeds meer in het voordeel van het genereren uitvallen. De waarde van Oracle*CASE tijdens de SO-fase wordt hierdoor benadrukt.

Er zijn nog te weinig IA-projecten afgerond om een oordeel te kunnen geven over het effect van Oracle*CASE daarop. Hetzelfde geldt voor onderhoud op m.b.v. Oracle*CASE ontwikkelde applicaties. Voor informatie-analyses wordt analoog aan systeemontwerpen een grotere uniformiteit verwacht van de eindproducten. Nemen we het gehele ontwikkelproces in ogenschouw, dan kan geconcludeerd worden dat bij gebruik van Oracle*CASE de fasen beter op elkaar aan zullen sluiten en dat de consistentie over de fasen heen veel beter gewaarborgd wordt.