

Ontwerp en prototyping van een multi-channel contentmanagement systeem

Contentstrategiebepaling en flexibel workflowmanagement
voor contentmanagement systemen

Fred van der Wouden
12 januari 2005



Ontwerp en prototyping van een multi-channel contentmanagement systeem

Contentstrategiebepaling en flexibel workflowmanagement
voor contentmanagement systemen

Auteur: Fred van der Wouden
Type verslag: Verslag afstudeerproject, definitieve versie
Datum: 12 januari 2005
Versie: 2.0

iCure ventures
Baan 10
3011 CB Rotterdam
Nederland

Technische Universiteit Delft
Faculteit Elektrotechniek,
Wiskunde en Informatica
Afdeling Software Technology



Voorwoord

Dit document bevat het eindverslag van het afstudeerproject zoals dat is uitgevoerd in het kader van de studie Technische Informatica aan de faculteit Elektrotechniek, Wiskunde en Informatica (EWI) van de TU Delft. Het afstudeerproject is het laatste onderdeel van de studie en is uitgevoerd onder de afdeling Software Technology (ST) in opdracht van iCure Ventures B.V. te Rotterdam.

iCure Ventures B.V. is aandacht gaan besteden aan productontwikkeling in de vorm van een multi-channel contentmanagement systeem (M-CMS) project. Het afstuderen heeft hieraan bijgedragen in de vorm van de analyse, ontwerp en prototyping van het M-CMS. Met het binnenkort afronden van het afstudeerproject in het bijzonder en de studie informatica in het algemeen hoop ik definitief de goede weg ingeslagen te hebben van langdurige stabiliteit in en controle over mijn chronische vermoeidheid.

Mijn dank gaat uit naar een groot aantal mensen: in de eerste plaats naar mijn begeleiders prof. dr. ir. J.L.G. Dietz en ir. H. Leliveld voor hun onbegrensd geduld, begripvolle begeleiding en ondersteuning tijdens het uitvoeren van het afstuderen. Bovenal naar mijn ouders en oma, wiens onvoorwaardelijke vertrouwen en steun op meer gebieden dan ik kan opnoemen mij meer dan eens door moeilijke perioden heen hebben geholpen. Naar Ralph Vanhauten en Lion Traas van Thinkworks B.V. die door de beschikbaarstelling van de tool Intelligent Objects en hun nimmer aflatende bereidbaarheid tot helpen een bijzonder waardevolle bijdrage hebben geleverd aan het project. En tenslotte naar alle vrienden die mij de afgelopen maanden mijn rust hebben gegund opdat ik mij volledig op het afronden van het afstuderen kon concentreren. Ik hoop hen spoedig weer vaker te zien, te beginnen bij het afstuderen op 27 januari 2005.

Fred van der Wouden
Rotterdam, 12 januari 2005

Samenvatting

Dit document bevat het verslag van het afstudeerproject, zoals dat in opdracht van iCure Ventures B.V. is uitgevoerd. iCure besteedt aandacht aan de ontwikkeling van een multi-channel contentmanagement systeem (M-CMS). Het M-CMS project vertaalt zich naar een afstudeerproject, dat geformuleerd is als de 'analyse, ontwerp en prototyping van een multi-channel contentmanagement systeem'.

Het M-CMS is een generiek en flexibel CMS-product, dat op maat gemaakt kan worden voor een specifieke klant. De belangrijkste functionaliteiten binnen het M-CMS zijn binnen het prototype gerealiseerd. Deze betreffen ondersteuning voor multi-channel contentlevering, voor workflowmanagement ten behoeve van het (content-) publicatieproces en ondersteuning voor het uitvoeren van beheeractiviteiten.

Binnen het M-CMS is een contentpijpleiding gerealiseerd die een contentstrategie hanteert die contentcreatie combineert met contentaanpassing. De XML content wordt op basis van een contentaanvraag dynamisch gegenereerd en getransformeerd met XSLT om de juiste layout en formaat te verkrijgen. Hierbij worden selectiemechanismen gehanteerd die de content aanpast op het doelapparaat in de zin van fysieke beperkingen en formaat- en standaardondersteuning. Bij de contentpijpleiding zijn de voornaamste prestatie-indicatoren de bruikbaarheid, de volledigheid en de leveringssnelheid van de content.

Het workflowmanagement biedt een adequate aansluiting op verschillende organisaties door een flexibele en generieke ondersteuning van het publicatieproces. Dit wordt bereikt door binnen het M-CMS middels een taakgeoriënteerde benadering ondersteuning te bieden voor alle fasen van de workflowlevenscyclus, te weten de procesinrichting, het procesmanagement en – bijsturing op basis van prestatie-indicatoren en signaleringsmogelijkheden en de procesevaluatie en –herinrichting. Dit levert ook enige ondersteuningsmiddelen voor de beheeractiviteiten, te weten het content-, site- en templatemanagement en systeembeheer, hoewel deze activiteiten een individualistischer karakter hebben en als separate functionaliteiten worden beschouwd.

Bij het M-CMS project is de CBD-gebaseerde ontwikkelmethode Rational Unified Process (RUP) gebruikt in combinatie met de objectgeoriënteerde (OO) modeleerwijze UML en de OO ontwikkeltool Intelligent Objects (IO). Deze onderdelen bleken goed op elkaar aan te sluiten en lenen zich goed voor de ontwikkeling van een component gebaseerd, generiek en flexibel CMS-product.

Met het M-CMS prototype zijn de kernfunctionaliteiten gerealiseerd en hun werking aangetoond. Voor de vervaardiging van een volwaardig basisproduct dient evenwel niet alleen verdere implementatie plaats te vinden en nadere aandacht voor contenttypes anders dan het uitvoerig behandelde teksttype. Tevens is het nodig dat er uitgebreide toetsing op basis van een uitgebreide testcasus in een live-omgeving plaatsvindt.

Inhoudsopgave

Voorwoord.....	iii
Samenvatting	v
1 Inleiding.....	1
1.1 Situatieschets	1
1.2 Onderzoeksdoelen.....	2
1.2.1 Contentstrategie	2
1.2.2 Systemeisen	3
1.2.3 Ontwikkelmethodiek en ontwikkeltool	3
1.3 Maatschappelijke relevantie	3
1.4 Wetenschappelijke relevantie	4
1.5 Structuur van het verslag	4
2 Afbakening en globale beschrijving M-CMS	6
2.1 Afbakening M-CMS.....	6
2.1.1 Karakteristieke doelorganisatie	6
2.1.2 Publicatieproces	7
2.1.3 2.1.3 Beheeractiviteiten	10
2.2 Globale beschrijving van M-CMS	11
2.2.1 Contentengine	11
2.2.2 Workflowmanagement.....	11
2.2.3 Beheeractiviteiten	14
3 Multi-channel contentstrategie	15
3.1 Uitgangspunten voor de contentstrategie	16
3.1.1 Standaard contentstrategieën	16
3.1.2 Raamwerk prestatie-indicatoren voor contentstrategie	18
3.1.3 Externe randvoorwaarde: HTTP request.....	19
3.1.4 Projectmatige randvoorwaarden en afbakening	20
3.2 Gehanteerde contentstrategie	21
3.2.1 XML en XSL(T) toepassing bij contentcreatie	21
3.2.2 Mechanismen binnen contentstrategie.....	24
3.3 Inrichting contentpijpleiding	27
3.3.1 Stap 1: Ontvangst HTTP request	28
3.3.2 Stap 2: Creeer contentaanvraag	29
3.3.3 Stap 3: Genereer contentpagina	29
3.3.4 Stap 4: Creeer XML-Node	29
3.3.5 Stap 5: Toepassen XSL-stylesheet en leveren content.....	30
3.4 Casus Nieuwssite.nl	30
3.4.1 Beschrijving casus Nieuwssite.nl.....	30
3.4.2 Inrichting van contentstructuur	31
3.4.3 Werking en resultaat van de contentpijpleiding	34
3.5 Evaluatie contentstrategie	37
3.5.1 Toetsing leveringssnelheid	38
3.5.2 Fijnstellen contentpijpleiding.....	40
3.5.3 Evaluatie op basis van raamwerk prestatie-indicatoren	40
4 Workflowmanagement	42
4.1 Plaatsbepaling workflowmanagement	42
4.1.1 Procesinrichting	44

4.1.2	Procesmanagement en procesbijsturing	44
4.1.3	Procesevaluatie en procesherinrichting	45
4.2	Workflowmanagement van het publicatieproces	46
4.2.1	Beargumentatie toepassing workflowmanagement	46
4.2.2	Procesinrichting publicatieproces.....	47
4.2.3	Procesmanagement en procesbijsturing publicatieproces.....	54
4.2.4	Procesevaluatie en procesherinrichting publicatieproces	60
4.3	Ondersteuning voor beheeractiviteiten.....	62
4.4	Evaluatie workflowmanagement.....	63
5	Het Rational Unified Process	65
5.1	Component based development: de keuze voor RUP	66
5.2	De denkwijze van het RUP	67
5.2.1	Use case driven	67
5.2.2	Architectuur-centrisch	68
5.2.3	Iteratief en incrementeel	68
5.3	De werkwijze van het RUP.....	69
5.4	De modellerwijze van het RUP: UML	73
5.5	De beheerswijze van het RUP	74
5.6	De ondersteuningswijze van het RUP	74
5.7	Toepassing van het RUP binnen het M-CMS project.....	74
5.7.1	Toepassing denkwijze.....	75
5.7.2	Toepassing werkwijze	75
5.7.3	Toepassing modellerwijze.....	75
5.7.4	Toepassing beheerswijze	76
5.7.5	Toepassing ondersteuningswijze	77
5.8	Evaluatie van het RUP	77
5.8.1	Denkwijze van het RUP	77
5.8.2	Werkwijze van het RUP	78
5.8.3	Modellerwijze van het RUP	78
5.8.4	Beheerswijze van het RUP	78
5.8.5	Ondersteuningswijze van het RUP	79
6	Intelligent Objects tool	80
6.1	Beschrijving Intelligent Objects	80
6.2	Prototyping van het M-CMS binnen IO.....	86
6.2.1	Aansluiting op UML en RUP	86
6.2.2	Geschiktheid voor internettechnieken.....	88
6.3	Evaluatie IO.....	89
7	Conclusies en aanbevelingen	92
7.1	Conclusies.....	92
7.1.1	Conclusies contentstrategie en templatemanagement	92
7.1.2	Conclusies workflowmanagement.....	92
7.1.3	Conclusies ontwikkelmethode RUP en ontwikkeltool IO.....	93
7.2	Aanbevelingen	93
	Literatuurverwijzingen.....	95
	Bijlage A: Acroniemen	97
	Bijlage B: Begrippenlijst.....	99
	Bijlage C: Ontwerp- en implementatiemodel UML.....	102

1 Inleiding

Dit afstudeerverslag is het resultaat van het afstudeerproject dat is uitgevoerd bij iCure e-business services B.V. te Rotterdam. In dit hoofdstuk wordt het kader geschetst waarbinnen het afstudeerproject heeft plaatsgevonden. In paragraaf 1.1 wordt de aanleiding van het afstudeerproject beschreven aan de hand van een situatieschets. Vervolgens worden in paragraaf 1.2 onderzoeksdoelen geformuleerd die aan het afstudeerproject ten grondslag liggen. In de paragrafen 1.3 en 1.4 worden de maatschappelijke respectievelijk wetenschappelijke relevantie van het afstuderen beschreven. Tenslotte geeft paragraaf 1.5 een overzicht van de structuur van het verslag.

1.1 Situatieschets

iCure e-business services B.V. te Rotterdam levert e-business diensten en applicaties aan diverse klanten met gebruikmaking van de nieuwste technologieën op het gebied van internet en mobiele toepassingen. iCure heeft opdrachten uitgevoerd die varieerden van de ontwikkeling van websites en softwaretoepassingen (zoals contentmanagement systemen en PDA-applicaties) tot het (her)ontwerpen van (e-business) processen.

iCure besteedt via iCure Ventures B.V. aandacht aan de ontwikkeling van een multi-channel contentmanagement systeem (M-CMS). Het afstudeerproject is binnen dit kader geformuleerd als:

Analyse, ontwerp en prototyping van een multi-channel contentmanagement systeem.

Een contentmanagement systeem (CMS) is een IT-gebaseerd systeem ter organisatie, beheer en doorvoering van contentmanagement (dat is gedefinieerd als de regelmatige, doelgerichte en systematische omgang met content). Binnen de onderzoekstaak [vdWouden] is onderzoek verricht naar CMS-en en de commerciële mogelijkheden die er voor iCure bestaan om een CMS-product neer te zetten. De conclusie luidde dat de ontwikkeling van een CMS met de volgende kenmerken de voorkeur verdient:

- Het product moet met name gericht zijn op content intensieve, middelgrote en kleine bedrijven (of afdelingen);
- Het speerpunt moet zijn de multi-channel beschikbaarstelling van content;
- Het product moet een aantal generieke functionaliteiten bevatten, waaronder ondersteuning voor workflowmanagement en beheeractiviteiten waaronder contentmanagement, sitemanagement en templatemanagement.

Teneinde het M-CMS geschikt te laten zijn voor verschillende organisatievormen is ervoor gekozen om een generiek basisproduct op te zetten, dat voldoende flexibiliteit biedt om op maat gemaakt te kunnen worden voor toepassing binnen diverse organisaties. Hierbij heeft iCure een content-intensieve doelorganisatie voor ogen, die een publicatie-proces kent waarbinnen een aantal actoren participeren, die beheeractiviteiten wenst uit te voeren en die de stap naar multi-channel publicatie wil maken of gemaakt heeft. De afbakening van het M-CMS, de karakteristieken van de

doelorganisatie en een overzicht van de generieke functionaliteiten van het M-CMS worden nader beschreven in hoofdstuk 2.

Centraal binnen het afstuderen staat de term content, die door [Lohr] wordt gedefinieerd als informatie [in de vorm van een uitwisselingsobject], die zich kenmerkt door de eigenschappen:

- Inhoud: de pure, ongeformatteerde basisinformatie;
- Structuur: het indelen van de content in componenten en het toekennen van eigenschappen aan de componenten voor een gestructureerde opzet;
- Layout: de textuele en grafische uitdrukkingmogelijkheden;
- Mediaformaat: het formaat van de content in softwarematige zin;
- Dragermedium: het medium waarop de content zich bevindt in hardwarematige zin.

In het algemeen wordt content beschouwd als onafhankelijk van dragermedium, zodat dit kenmerk verder niet in beschouwing zal worden genomen. Deze definitie leent zich vooral goed voor de beschouwing van tekstuele content: de inhoud als ongeformateerde tekst, de structuur als opdeling naar titel, alinea's en dergelijke, de layout als lettergrote en -type en het formaat als bijv. WORD of HTML vorm. Ook de andere contenttypes zoals image of video kunnen binnen deze definitie worden gevat, waarvoor in hoofdstuk 3 bij de afbakening van de contentstrategie een nadere beschouwing zal plaatsvinden.

1.2 Onderzoekdoelen

Ten behoeve van het afstudeerproject zijn drie categorieën onderzoekdoelen gedefinieerd die specifieke aandacht verdienen binnen de uitvoering van het M-CMS project. Allereerst wordt er gekeken naar de toe te passen contentstrategie om verschillende kanalen van content te voorzien. Vervolgens worden enkele doelen geformuleerd ten aanzien van systeemeisen op het gebied van workflowmanagement en templatemanagement. Tenslotte worden onderzoekdoelen geformuleerd met betrekking tot de gehanteerde ontwikkelmethodiek en de toe te passen ontwikkeltool.

1.2.1 Contentstrategie

In navolging van de voornamelijk theoretische beschouwing van de onderzoekstaak zal het zwaartepunt van het M-CMS liggen op:

Het bepalen, toetsen en evalueren van de meest geschikte contentstrategie voor het beschikbaar stellen van content aan verschillende kanalen.

In de onderzoekstaak [vdWouden] zijn de standaard contentstrategieën contentselectie, contentcreatie en contentaanpassing beschreven. Voor de keuze van de meest geschikte contentstrategie voor het M-CMS wordt antwoord gezocht op de volgende onderzoeksvragen:

- Kan er een praktisch werkbaar contentstrategie worden gevonden die content onafhankelijk houdt van formaat, structuur en vormgeving in zowel het publicatie- als het beheerproces?
- Welke combinatie van de strategieën contentselectie en contentcreatie verdient de voorkeur en hoe kan deze worden gecombineerd met de inrichting van de strategie gericht op contentaanpassing?

Voor de toetsing en evaluatie van de gekozen contentstrategie kan antwoord op de volgende onderzoeksvraag uitkomst bieden:

- Kan er een raamwerk voor prestatie-indicatoren worden opgesteld die toetsing, evaluatie en een eventuele vergelijking van contentstrategieën mogelijk maakt?

1.2.2 Systeemeisen

De systeemeisen leveren onderzoeksdoelen op ten aanzien van workflowmanagement en templatemanagement.

Onder workflowmanagement wordt verstaan het managen van processen waarbinnen een scheiding is aangebracht tussen de uitvoering en de besturing van activiteiten [vdBerg]. Het volgende onderzoeksdoel verdient daarbij invulling:

Het inrichten van het workflowmanagement van het M-CMS zodat een goede ondersteuning voor het publicatie- en beheerproces wordt verkregen, met behoud van een flexibele opzet zodat deze binnen verschillende organisatievormen toepasbaar is.

Onder templatemanagement wordt verstaan het creëren en beheren van templates die de structuur van content en vormgeving vastleggen. Hierbij kan gedacht worden aan de definitie van een eXtensible Markup Language (XML) structuur voor specifieke vormen van contentpagina's tot de definitie van een (organisatiebrede) Document Type Definition (DTD). Hiertoe wordt het volgende onderzoeksdoel verwezenlijkt:

Het inrichten van het templatemanagement om enerzijds structuur aan te brengen in content- en vormgevingsjablonen en anderzijds deze flexibel te houden door aanpasbare systeeminstellingen te implementeren.

1.2.3 Ontwikkelmethodiek en ontwikkeltool

Ten aanzien van de gehanteerde ontwikkelmethodiek Rational Unified Process (RUP) in combinatie met de modelleerwijze van de Unified Modelling Language (UML) is het volgende onderzoeksdoel geformuleerd:

Beoordeling en evaluatie van het RUP voor het M-CMS project zowel kijkend naar het incrementeel en iteratief karakter van de ontwikkelmethodiek als naar de bruikbaarheid en volledigheid van tussen- en eindresultaten.

De tool Intelligent Objects (IO) is binnen het M-CMS project de gehanteerde ontwikkeltool. Hiervoor is het volgende onderzoeksdoel geformuleerd:

De toetsing en evaluatie van de geschiktheid van de ontwikkeltool IO voor de toepassing bij systemen die zich richten op internettechnieken.

1.3 Maatschappelijke relevantie

Hoewel de Internet hype de laatste jaren minder sterk is toegenomen dan werd verwacht, is op het gebied van de introductie van nieuwe media en apparaten via welke content wordt verspreid veel ontwikkeling waar te nemen. Bovendien is voor de ondersteuning van contentmanagement een behoorlijke wildgroei ontstaan van systemen. Deze CMS-en sluiten enerzijds lang niet altijd goed aan op de (bedrijfs)processen van bestaande doelorganisaties. Anderzijds

bieden de CMS-en niet altijd de mogelijkheid om de content distributie via verschillende kanalen te verzorgen, zeker niet op het segment van kleine en middelgrote bedrijven, of lossen de multi-channel problematiek hiervan onvoldoende op. Deze multi-channel problematiek is ontstaan als gevolg van de explosieve toename van nieuwe apparaten die met het Internet communiceren, waardoor grote onzekerheid bestaat omtrent het correct afbeelden van de content vanwege de sterk uiteenlopende (fysieke) eigenschappen van de diverse apparaten en hun user agent.

Het M-CMS is ontwikkeld om deze twee discrepanties te reduceren. Enerzijds biedt het M-CMS middels het workflowmanagement een flexibele en eenvoudige ondersteuning voor het publicatieproces en beheeractiviteiten. Hiermee kan de internetpublicatie binnen doelorganisaties gestructureerd en professioneel verlopen, anders dan de hobbyistische aanpak die resulteerde uit het HTML gebruik van de beginjaren. Anderzijds maakt het M-CMS door middel van de contentpijpleiding de effectieve publicatie naar diverse mediakanalen mogelijk. Hiermee wordt voor de multi-channel problematiek een oplossing geboden, die zorgt voor een betere aansluiting van de internetstandaarden en -formaten op de snelle ontwikkelingen op het gebied van multi-channeltoepassingen en -apparaten.

1.4 Wetenschappelijke relevantie

Het Internet heeft zich in de beginjaren na haar introductie ontpopt als een medium dat zeer veel mogelijkheden biedt en waarop nauwelijks beperkingen zijn ten aanzien van toegankelijkheid en gebruik. De ontwikkeling van HTML als laagdrempelige markeertaal heeft er ook toe bijgedragen dat iedereen zonder al te veel inspanning in de Internet ontwikkeling kon participeren.

Deze ontwikkelingen hebben er evenwel aan bijgedragen dat wetenschappelijk gezien de internettechnieken als weinig gestructureerd en onvolwassen werden beschouwd. Dit uitte zich bijvoorbeeld door het probleem van exponentiële groei van webpagina's (onder andere zoektechnisch door gebrek aan structuur). Om de standaardisatie en het volwassen worden van internettechnieken te stimuleren is het World Wide Web Consortium [W3C] in het leven geroepen.

De ontwikkeling van het M-CMS draagt bij aan het verkrijgen van inzicht in de mate waarin nieuwe internettechnieken en -standaarden de ontstane problemen kunnen oplossen door het toevoegen van structuur aan content en het verhogen van compatibiliteit en toepasbaarheid door standaardisatie. Dit wordt specifiek toegepast op de multi-channel problematiek middels de facto standaarden XML en XSL(-T), waarmee zowel compatibiliteit met hedendaagse als oudere standaarden wordt verkregen alsmede standaardisatie wordt aangebracht om op toekomstige technieken en standaarden te kunnen aansluiten.

1.5 Structuur van het verslag

Het verslag is in belangrijke mate ingedeeld op basis van de in paragraaf 1.2 beschreven onderzoeksdoelen. Om het globale kader van het M-CMS project te schetsen zal eerst in hoofdstuk 2 de afbakening van het project worden gepresenteerd, op basis van de randvoorwaarden en de kenmerken en processen van de doelorganisatie. Bovendien zullen generieke functionaliteiten

van het M-CMS worden beschreven om een overzicht te geven van het CMS-product.

In de daaropvolgende hoofdstukken komen specifieke onderwerpen aan bod, die tegemoet komen aan de onderzoeksdoelen. In hoofdstuk 3 komt de contentstrategie inclusief de inrichting van het templatemanagement aan de orde middels een beschrijving van de geïmplementeerde contentpijpleiding die zorgt voor distributie van content naar verschillende kanalen. In hoofdstuk 4 wordt vervolgens de implementatie van het workflowmanagement voor het publicatieproces en de ondersteuning voor de beheeractiviteiten geschetst.

In hoofdstuk 5 wordt aandacht besteed aan de toegepaste ontwikkelmethode van het Rational Unified Process (RUP) en de modelleerwijze volgens de Unified Modelling Language (UML). De ontwikkeltool Intelligent Objects die gebruikt is bij de implementatie van het M-CMS prototype wordt in hoofdstuk 6 beschreven en geëvalueerd. Hoofdstuk 7 sluit het verslag af met enkele conclusies en aanbevelingen.

De voorkomende literatuurverwijzingen komen in twee verschijningsvormen voor, te weten als '[auteur]' in geval van verwijzingen naar boeken of artikelen en in geval van website verwijzingen als '[referentie]'. Vanwege het veelvuldig gebruik van acroniemen en specialistische terminologie in het verslag is ter verheldering in bijlage A een lijst met acroniemen opgenomen en in bijlage B een begrippenlijst. In het verslag zijn op verschillende punten onderdelen van het ontwerpmodel opgenomen voor toelichting van de betreffende onderwerpen. In bijlage C is het volledige RUP ontwerpmodel opgenomen inclusief een toelichting van de UML modelleringen.

2 Afbakening en globale beschrijving M-CMS

Het multi-channel contentmanagement systeem (M-CMS) is een flexibel CMS product dat voor verschillende organisaties geschikt moet zijn. Voor de afbakening binnen het analyse- en ontwerptraject van de productontwikkeling is het noodzakelijk een generieke organisatie voor ogen te houden met bepaalde karakteristieken. In paragraaf 2.1 worden ten behoeve van de afbakening van het project enkele randvoorwaarden beschreven, in het bijzonder de kenmerken van de doelorganisatie. In paragraaf 2.2 wordt globaal de functionaliteit van het M-CMS beschreven om een overzichtelijk globaal beeld te schetsen van het product tijdens de ontwerpfase.

2.1 Afbakening M-CMS

Om te komen tot een afbakening van het M-CMS is het noodzakelijk om enkele randvoorwaarden te definiëren waaraan het M-CMS moet voldoen. Dit is enerzijds nodig voor de afbakening van de activiteiten binnen de analyse- en ontwerpfase. Anderzijds is het nodig om in de implementatie fase een prioriteitsstelling toe te kennen aan functionaliteiten voor het maken van een prototype. Globaal gezien zijn de volgende randvoorwaarden te definiëren:

- Het M-CMS moet een basisfunctionaliteit bezitten die toepassing ervan mogelijk maakt binnen doelorganisatie, die in paragraaf 2.1.1 beschreven wordt;
- Het M-CMS moet voldoende flexibiliteit bezitten om binnen specifieke organisaties ingezet te worden (indien nodig op bepaalde punten op maat te maken), met als uitgangspunt de doelorganisatie;
- Het prototype van het M-CMS moet tenminste die functionaliteit implementeren, die binnen de onderzoeksdoelen toetsing behoeven. Deze minimale functionaliteit betreft de realisatie van de contentengine, het workflowmanagement van het publicatieproces en de ondersteuning voor de beheeractiviteiten. Deze worden beschreven in de paragraaf 2.2.

Deze randvoorwaarden zijn globaal en behoeven per aandachtsgebied nadere specificatie. In de hoofdstukken 3 en 4 zal dan ook per onderwerp nadere invulling plaatsvinden ten aanzien van functionaliteit en flexibiliteit.

2.1.1 Karakteristieken doelorganisatie

In deze paragraaf worden enkele karakteristieken van de doelorganisatie beschreven, alsmede enkele voorbeelden van organisatievormen binnen welke het M-CMS toegepast moet kunnen worden. In grote lijnen is er een beeld van een doelorganisatie met de volgende generieke kenmerken:

- Contentintensief: de organisatie heeft een hoge mate van contentverloop. Dat wil zeggen dat de content regelmatig wordt aangepast of dat er regelmatig (al dan niet periodiek) nieuwe content wordt gepubliceerd. Hierbij is niet het volume van de content het belangrijkste, maar de frequentie waarmee content eenheden worden gepubliceerd of aangepast;
- Publicatieproces: vanwege de contentintensiviteit is er sprake van een publicatieproces dat gebonden is aan een bepaalde tijdvoorwaarde. Het kan zijn dat een snelle 'time to publication' gewenst is of dat er een publicatiedeadline is op basis van periodieke publicatie. Dit wordt in de volgende paragraaf nader uitgewerkt;

- Beheerproces: de organisatie zal mogelijk (onderdelen van) het beheer van de content, site(s), templates en het systeem willen uitvoeren;
- Multi-channel gericht: de organisatie wil de stap maken naar het publiceren naar verschillende media of heeft deze al gemaakt. Hierbij gaat de aandacht met name uit naar kanalen als het internet, het wireless application protocol (WAP) en het bedienen van apparaten als personal digital assistants (PDA's) en niet zozeer naar contentpublicatie naar andere digitale informatiedragers als CD-roms of DVDs.

Er is binnen de doelorganisatie onderscheid te maken tussen het publicatieproces en het beheerproces:

- Het publicatieproces betreft alle activiteiten die leiden tot het publiceren van de content op een betreffende site. Dit proces wordt nader belicht in paragraaf 2.1.2;
- Het beheerproces betreft alle activiteiten die na de publicatie gehanteerd worden om de content te onderhouden, archiveren (site- en contentmanagement) en het beheer te verzorgen van het systeem en de templates (systeembeheer en template-management). Het beheerproces bestaat uit een aantal min of meer afzonderlijke activiteiten, die binnen iedere organisatie anders worden ingevuld of (deels) niet worden uitgevoerd. Als zodanig is er geen sprake van een bedrijfsproces, want dit wordt gedefinieerd als een causaal samenhangend geheel van activiteiten [Dietz]. In het vervolg zal derhalve gesproken worden van beheeractiviteiten, die in paragraaf 2.1.3 worden behandeld.

De volgende voorbeelden kunnen van de doelorganisatie worden gegeven, waarvan de eerste twee leidend zijn binnen dit document en regelmatig zullen worden aangehaald voor illustratieve doeleinden:

- Een online magazine, dat periodiek (wekelijks of maandelijks) via het web en mogelijk andere kanalen wordt gepubliceerd;
- Een organisatie die een nieuwssite beheert, zoals nu.nl, die verdeeld over verschillende rubrieken diverse artikelen aanbiedt met een hoog contentverloop;
- Een organisatie als een sportbond, die een grote verscheidenheid aan artikelen, uitslagen en mogelijk dynamische elementen wil presenteren;
- Een online store voor bijvoorbeeld boeken, multi-media of internet producten, waarbij producten via diverse kanalen te bekijken en bestellen zijn;
- Een lijnorganisatie, die met enige regelmaat nieuwe producten of diensten via het net presenteert, en daarbij gebruik makend van diverse kanalen.

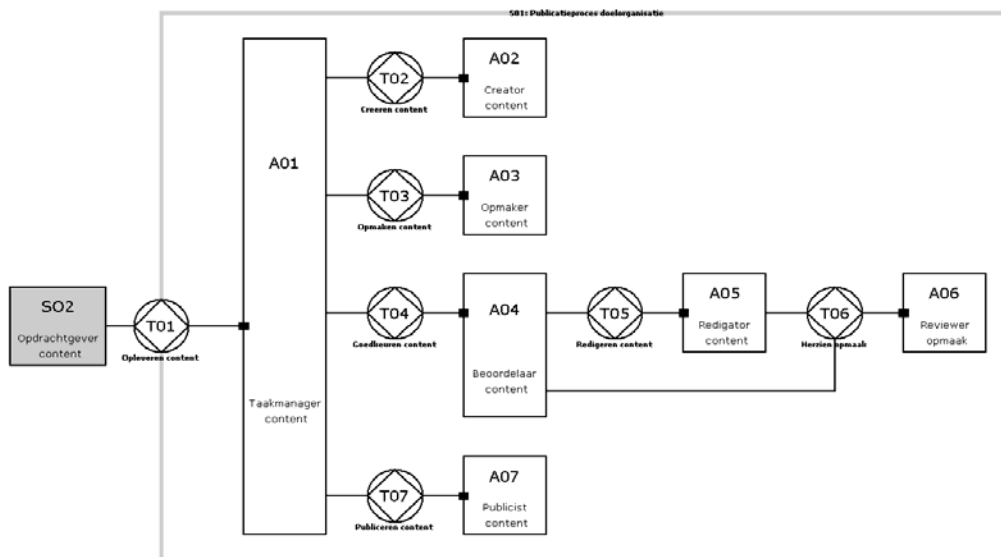
2.1.2 Publicatieproces

In deze paragraaf zal het publicatieproces zoals dat binnen de doelorganisatie bestaat worden beschreven en gemodelleerd. De gehanteerde ontwikkelmethode het Rational Unified Process (RUP) en de bijbehorende modelleerwijze Unified Modelling Language (UML) worden voor het gehele M-CMS project gehanteerd voor het modelleren en documenteren van de projectdelen. Er bestaat hierbinnen echter geen ondersteuning voor het op hoog abstractieniveau modelleren van bedrijfsprocessen anders dan middels tekstuele beschrijvingen. Om het publicatieproces helder in beeld te brengen

zal in deze paragraaf de DEMO (Design and Engineering Methodology for Organisations, [DEMO]) methodiek van [Dietz] gebruikt worden om het publicatieproces op het zogenaamde essentieel niveau te beschrijven. Zodra het RUP en UML zich lenen voor de beschrijving van aspecten van het M-CMS, dan zal van de DEMO modellering worden overgestapt naar de RUP documentatie en UML modellering.

De DEMO methodiek [Dietz] hanteert het transactieconcept als uitgangspunt voor het modelleren van (het) primaire bedrijfsproces(sen) binnen een organisatie. De volvoering van een transactie voltrekt zich volgens het OER-concept in drie fasen: de opdrachtfase (O-fase), de executiefase (E-fase) en de resultaatfase (R-fase). In figuur 2.1 is het communicatiediagram van DEMO opgenomen. De transactietypen (de T-elementen afgebeeld als een ruit binnen een cirkel) zijn hierbij de transacties zoals deze binnen het publicatieproces plaatsvinden tussen de actoren. De actoren zijn afgebeeld als rechthoeken (A-elementen) waartussen de transactie plaatsvindt. Hierbij is één actor de initiator en de andere actor is de executor van de transactie (eindigend met dikke punt). Bijvoorbeeld: de actor A01 'Taakmanager content' is de initiator van de transactie T03 'Opmaken content' die door de actor A03 'Opmaker content' wordt uitgevoerd.

Het communicatiediagram van figuur 2.1 dient als volgt gelezen te worden: het proces wordt geïnitieerd door een verzoek voor het indienen van een stuk content. Hiervoor wordt de content gecreëerd, van opmaak voorzien en ter beoordeling voorgelegd. Indien de content wordt goedgekeurd, kan deze worden gepubliceerd. Als de content wordt afgekeurd, dan kan deze voor het redigeren van de content en / of het herzien van de opmaak in aanmerking komen; het is mogelijk dat slechts het herzien van de opmaak benodigd is en niet het redigeren. Deze cyclus van afkeuren en herzien kan zich herhalen totdat de opgemaakte content wordt goedgekeurd.



Figuur 2.1: DEMO communicatiediagram van het publicatieproces

Transactietype	Transactieresultaat
T01 Opleveren content	De contentpagina <cp> is opgeleverd
T02 Creëren content	De contentpagina <cp> is gecreëerd
T03 Opmaken content	De contentpagina <cp> is opgemaakt
T04 Goedkeuren content	De contentpagina <cp> is goedgekeurd
T05 Redigeren content	De contentpagina <cp> is geredigeerd
T06 Herzien opmaak	De contentpagina <cp> is herzien van opmaak
T07 Publiceren content	De contentpagina <cp> is gepubliceerd

Tabel 2.1: DEMO transactieresultaat tabel

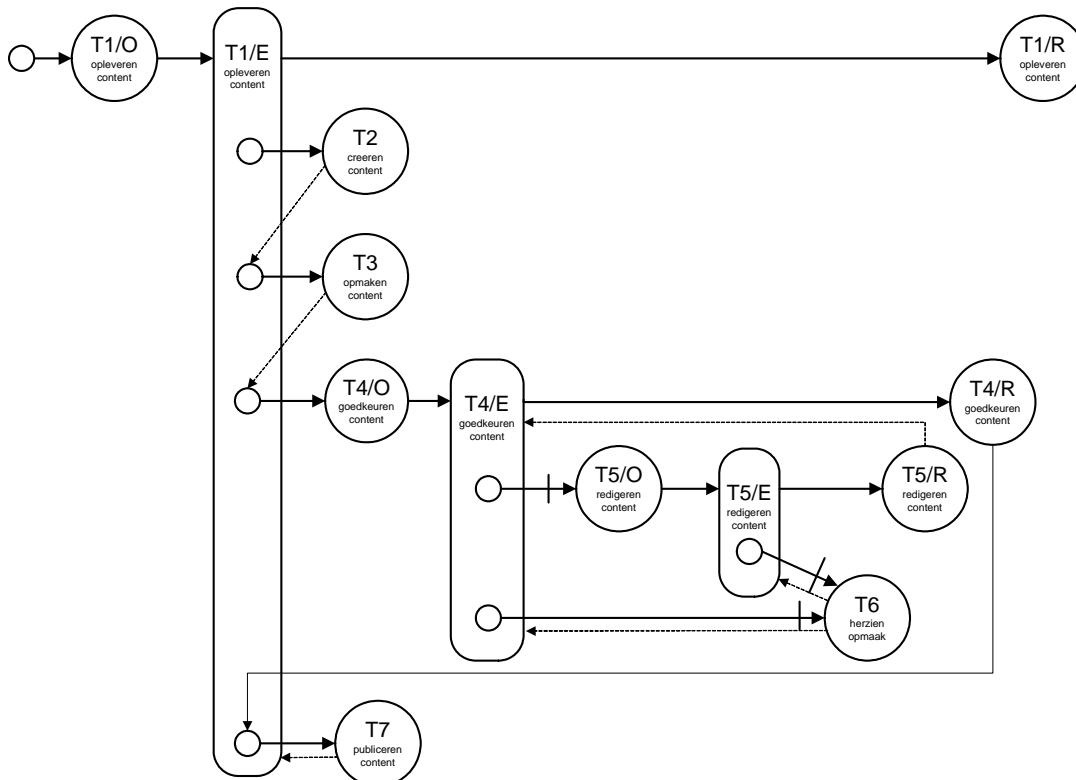
Tabel 2.1 toont de transactieresultaat tabel, waarin de bijbehorende resultaten van de transacties zijn opgenomen. Het publicatieproces wordt beschouwd als dusdanig generiek dat deze binnen iedere doelorganisatie voorkomt, hoewel de exacte organisatorische invulling van actoren kan verschillen. Voor de actoren die een rol spelen binnen het publicatieproces kunnen met behulp van de organisatorische invullingstabel van DEMO de verantwoordelijkheden worden toegekend aan de uitvoerende actor, zoals bijvoorbeeld in tabel 2.2.

Actor	Organisatorische invulling
S02 Opdrachtgever content	Eind- of hoofdredacteur
A01 Taakmanager content	Taakmanager
A02 Creator content	Redacteur
A03 Opmaker content	Vormgever
A04 Beoordelaar content	Eindredacteur
A05 Redigator content	Redacteur
A06 Reviewer opmaak	Vormgever
A07 Publicist content	Eindredacteur

Tabel 2.2: Voorbeeld van organisatorische invulling van actoren

De volgtijdelijkheid en afhankelijkheid van de transacties binnen het publicatieproces kunnen middels het DEMO procesdiagram (figuur 2.2) worden weergegeven. Hierin zijn causale afhankelijkheden te zien, die gerepresenteerd worden door normale pijlen. Een voorbeeld hiervan is het opstarten van transactie T1 vanuit het initiatiepunt linksboven. Enkele van de causale afhankelijkheden zijn optioneel, gerepresenteerd door het dwarsstreepje. Transactie T05 'redigeren content' wordt bijvoorbeeld slechts dan geïnitieerd wanneer de content wordt afgekeurd. Daarnaast bestaan er enkele conditionele afhankelijkheden, afgebeeld als stippelpijlen. Zo is het afronden van transactie T02 'creëren content' een noodzakelijke voorwaarde voor het initiëren van transactie T03.

Het publicatieproces kan beschouwd worden als een workflowproces, een proces waarbinnen een scheiding is aangebracht tussen de uitvoering en besturing van de procesactiviteiten. In hoofdstuk 4 zal hier uitgebreid op worden ingegaan, waarbij op de binnen het gehele M-CMS project gehanteerde UML modellering zal worden overgestapt, omdat deze op dat punt voorziet in de benodigde uitdrukkingvormen.



Figuur 2.2: DEMO procesdiagram van het publicatieproces

2.1.3 2.1.3 Beheeractiviteiten

De beheeractiviteiten betreffen alle activiteiten die na de contentpublicatie gehanteerd worden om de content te onderhouden en archiveren en het beheer te verzorgen van het systeem en templates. De beheeractiviteiten bestaan uit:

- Contentbeheer: het beheren van de content in zijn elementaire vorm van contentpagina's door de actor contentbeheerder. Hierbij worden activiteiten uitgevoerd als het aanpassen van live content en structuur, archivering en verwijdering van content;
- Sitebeheer: beheeractiviteiten die betrekking hebben op een samenhangend geheel van contentpagina's (site), die worden uitgevoerd door de actor contentbeheerder. Voorbeelden hiervan zijn het uitvoeren van back ups en de roll back: het vervangen van sites door een oudere versie bijvoorbeeld in geval van corrupte kritieke content;
- Systeembeheer: hieronder wordt verstaan het beheren van systeeminstellingen (voor het M-CMS in het algemeen en van de contentengine in het bijzonder), alsmede het gebruikers- en autorisatiebeheer van actorgroepen en actoren. Dit wordt uitgevoerd door de actor systeembeheerder;
- Templatebeheer: het creëren, aanpassen en verwijderen van standaard sjablonen (templates) die gebruikt worden voor de structuur van content in de vorm van XML-bestanden en de opmaak ervan in de vorm van XSL-stylesheets. Dit wordt verzorgd door de actor metacontentbeheerder.

In tabel 2.3 worden de actoren beschreven die de beheeractiviteiten uitvoeren. In paragraaf 2.2.3 zal de ondersteuning van deze activiteiten nader worden toegelicht.

Actor	Beschrijving	Verantwoordelijkheden
Content beheerder	Voert beheerstaken uit op gepubliceerde content	Content correctie (live) Content reductie: archivering, verwijdering Sitebeheer Uitvoeren roll back
Systeem beheerder	Voert beheerstaken uit op systeeminstellingen en contentengine	Systeembeheer Technisch beheer contentengine
Metacontent beheerder	Voert beheerstaken uit op metacontent	Templatenmanagement (wijzigen, aanpassen templates)

Tabel 2.3: Actoren met verantwoordelijkheden binnen de beheeractiviteiten

2.2 Globale beschrijving van M-CMS

Alvorens in de volgende hoofdstukken specifiek op enkele aspecten van het M-CMS in te gaan, is het nuttig om een globaal beeld van de functionaliteit te schetsen. Binnen het M-CMS is de belangrijkste functionaliteit te groeperen naar een drietal aandachtsgebieden, die voor een belangrijk deel in het prototype zijn geïmplementeerd:

- Contentengine: de technische realisatie binnen het systeem die de multi-channel distributie van de content mogelijk maakt;
- Workflowmanagement: de ondersteuning voor het publicatieproces vindt plaats in de vorm van workflowmanagement;
- Beheeractiviteiten, die binnen het M-CMS ondersteund worden, zijn onder te verdelen in contentbeheer, sitebeheer, systeembeheer, en templatebeheer.

2.2.1 Contentengine

De contentengine betreft de technische realisatie van de multi-channel distributie van content via diverse mediakanalen. Op basis van een contentaanvraag van een (site) bezoeker wordt middels een contentpijpleiding de betreffende content gegenereerd en aangepast tot de content het gewenste formaat, structuur en opmaak heeft en geleverd kan worden. In hoofdstuk 3 wordt de werking van de contentpijpleiding beschreven en geëvalueerd.

2.2.2 Workflowmanagement

Software systemen in het algemeen en contentmanagement systemen in het bijzonder worden nogal eens binnen organisaties geïntroduceerd, zonder dat er een adequate afstemming plaatsvindt op de bedrijfsprocessen van de organisatie. In het geval van CMS-producten die in eerste instantie aangeschaft worden om ondersteuning te bieden voor het publiceren en beheren van websites, is de afstemming op bedrijfsprocessen nogal eens onderbelicht. Hoewel de meeste CMS-producten ondersteuning bieden voor workflowmanagement [vdWouden], zijn dit soms starre en weinig flexibele oplossingen.

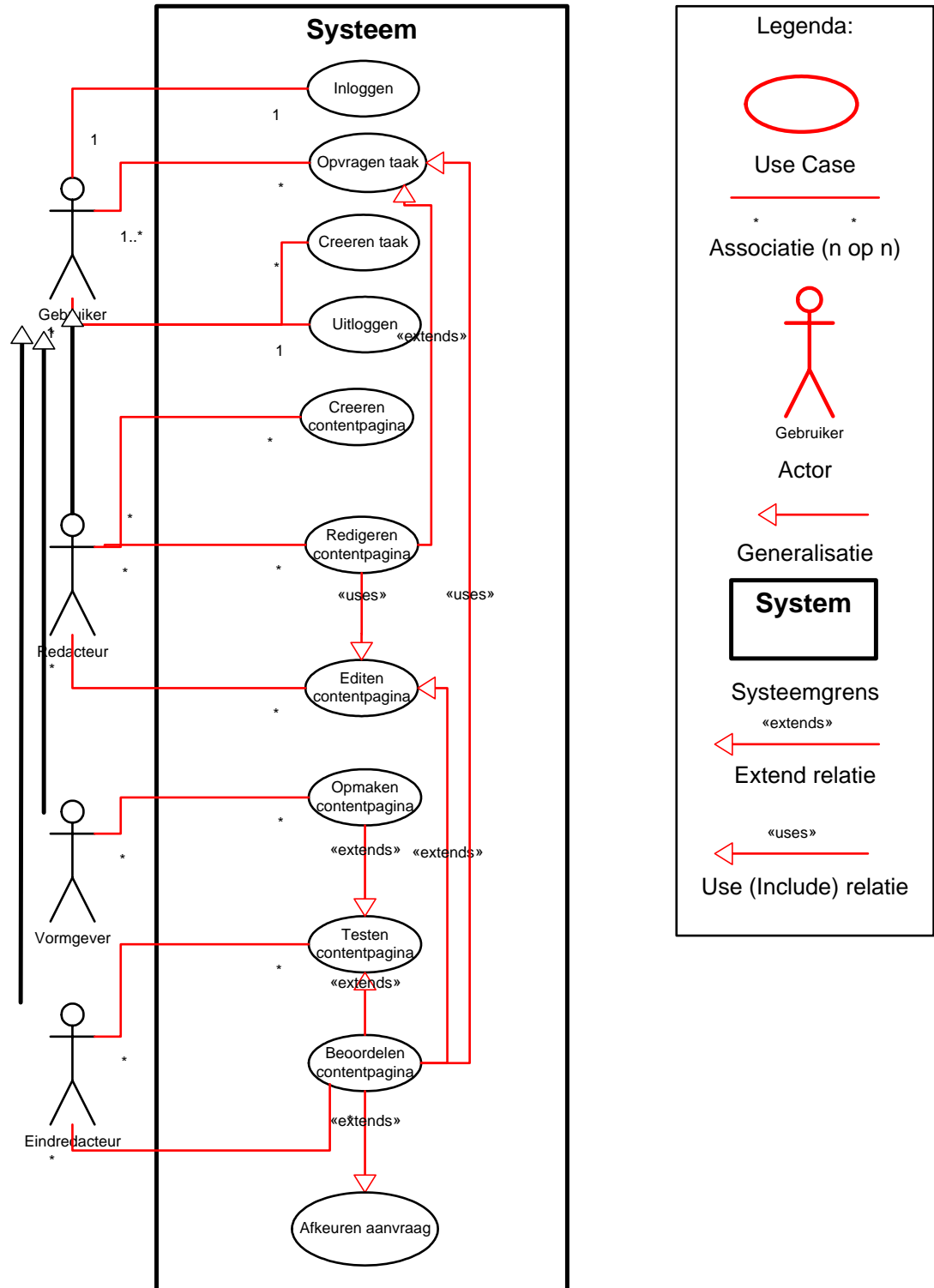
Het M-CMS tracht middels het onderzoeksdoel uit paragraaf 1.2.2 een goed en flexibel workflowmanagement te verkrijgen voor het publicatieproces, zodat het op maat gemaakt kan worden voor een specifieke organisatie.

Workflowmanagement betreft het managen van processen waarbinnen een scheiding is aangebracht tussen de uitvoering en besturing van de procesactiviteiten. Dit wordt gerealiseerd binnen het M-CMS door de activiteiten onder te verdelen in taken, die kunnen worden toegewezen aan, in behandeling genomen en uitgevoerd door individuele actoren of actorgroepen. Het workflowmanagement voor het publicatieproces wordt in hoofdstuk 4 uitgebreid toegelicht.

In figuur 2.3 is het use case diagram van de Unified Modelling Language (UML [OMG]) van de publicatie activiteiten gegeven dat een overzicht geeft van de functionaliteit van het M-CMS voor wat betreft de ondersteuning van het publicatieproces. Een use case is (de ovale vorm binnen de systeemgrens) een stukje functionaliteit waarin het systeem voorziet. De use cases zijn middels associaties verbonden met actoren, waarmee wordt aangegeven dat een actor gebruik kan maken van een use case. De actoren representeren de verschillende gebruikers van het systeem. De associaties bezitten een multipliciteit: bijvoorbeeld de 1..* op * relatie tussen de actor 'Gebruiker' en de use case 'Opvragen taak' wil zeggen dat 1 of meer gebruikers 0 of meerdere taken kunnen opvragen.

De actor 'gebruiker' is hierbij gehanteerd als generieke gebruiker, die eigenschappen bezit die iedere gebruiker heeft, hetgeen middels de generalisatie pijl van andere actoren naar de actor 'gebruiker' is aangegeven. Binnen het systeem, dat wordt afgebakend door de systeemgrens, bevinden zich tussen bepaalde use cases de 'extend' en 'use' relaties. De extend relatie geeft aan dat een use case hetzelfde gedrag kan vertonen als een andere use case op basis van bepaalde voorwaarden. Bijvoorbeeld bij de use case 'opmaken content' kan gebruik gemaakt worden van 'testen contentpagina'. De 'use' relatie (ook bekend als 'include' relatie) geeft aan dat een use case hetzelfde gedrag vertoont als een andere use case. In bijlage C zijn de volledige specificaties van de use cases opgenomen.

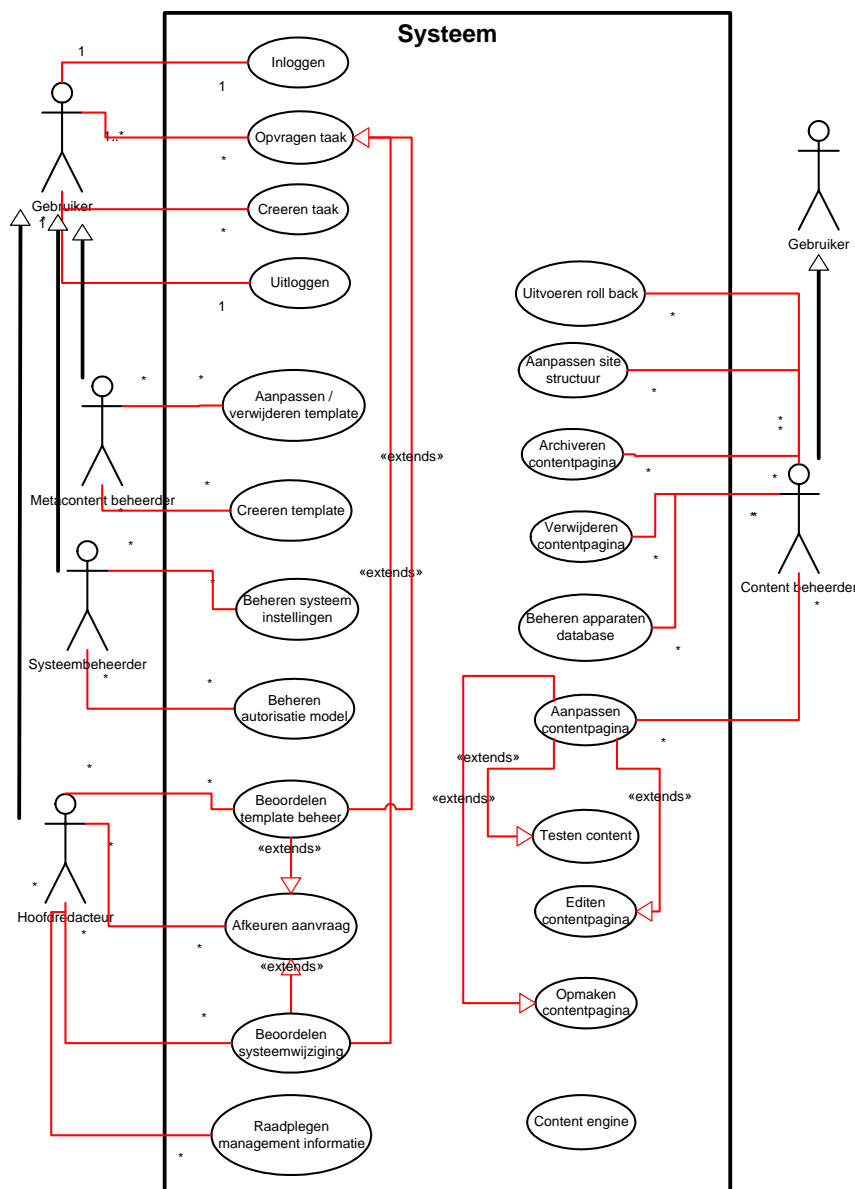
Het use case model beeldt zoals gezegd de functionaliteiten van het M-CMS uit, die het publicatieproces ondersteunen. Als zodanig is het use case model te vergelijken met de informationele ondersteuning van het DEMO communicatie diagram (CD). Een aantal transacties uit het DEMO CD worden nog wel direct gerepresenteerd, zoals 'creëren content' of 'opmaken content', hoewel inmiddels van een contentpagina wordt gesproken als kleinste eenheid van content. De use case diagrammen zijn hiermee wel een ondersteuning voor de DEMO modellen, maar een 1-op-1 relatie is niet aan te geven omdat er feitelijk sprake is van een ander abstractienivo. In hoofdstuk 4 zal de consistentie tussen de DEMO en de UML modellering worden duidelijk gemaakt middels het activity diagram van UML, dat een vergelijkbare uitdrukingsvorm als het procesdiagram heeft.



Figuur 2.3: Use case model van de publicatie activiteiten

2.2.3 Beheeractiviteiten

De beheeractiviteiten betreffen alle activiteiten die na de contentpublicatie gehanteerd worden om de content te onderhouden en archiveren (site en contentmanagement) en het beheer te verzorgen van het systeem en templates (systeembeheer respectievelijk templatebeheer). Het is mogelijk om deze activiteiten met de hulpmiddelen van het workflowmanagement te ondersteunen, bijvoorbeeld door taakmanagement in het geval dat deze activiteiten goedkeuring behoeven van de eind- of hoofdredacteur. In hoofdstuk 4 wordt hier nader aandacht aan besteed. In figuur 2.4 is het use case model van de beheeractiviteiten opgenomen, inclusief de use case 'contentengine' die los in het model is opgenomen omdat deze geen directe interactie heeft met de actoren.



Figuur 2.4: Use case model van de beheeractiviteiten

3 Multi-channel contentstrategie

In dit hoofdstuk wordt de multi-channel contentstrategie gepresenteerd zoals deze binnen het multi-channel contentmanagement systeem (M-CMS) wordt gehanteerd. Zodoende wordt richting gegeven aan de implementatie zoals deze heeft plaatsgevonden middels de contentpijpleiding. Hiermee wordt invulling gegeven aan het volgende onderzoeksdoel met betrekking tot de contentstrategie:

Het bepalen, toetsen en evalueren van de meest geschikte contentstrategie voor het beschikbaar stellen van content aan verschillende kanalen.

Binnen dit hoofdstuk zal tevens een antwoord worden gegeven op de onderzoeksvragen die bij dit onderzoeksdoel behoren:

- Kan er een praktisch werkbaar contentstrategie worden gevonden die content onafhankelijk houdt van formaat, structuur en vormgeving in zowel het publicatie- als het beheerproces?
- Welke combinatie van contentsselectie en contentcreatie verdient de voorkeur en hoe kan deze gecombineerd worden met de inrichting van de strategie gericht op contentaanpassing?
- Kan er een raamwerk voor prestatie-indicatoren worden opgesteld die toetsing, evaluatie en een eventueel vergelijk van contentstrategieën mogelijk maakt?

Bovendien zal hierbij impliciet aandacht worden besteed aan het templatemanagement: het creëren en beheren van basissjablonen die de structuur van content en vormgeving vastleggen. Hiertoe wordt het volgende onderzoeksdoel verwezenlijkt:

Het inrichten van het templatemanagement om enerzijds structuur aan te brengen in content- en vormgevingsjablonen en anderzijds deze flexibel te houden door aanpasbare systeeminstellingen te implementeren.

In paragraaf 3.1 zal daartoe eerst een aantal uitgangspunten van de contentstrategie worden behandeld. Hierbij komen een aantal standaard contentstrategieën aan de orde, het raamwerk met prestatie-indicatoren op basis waarvan de contentstrategie kan worden geëvalueerd en de van toepassing zijnde randvoorwaarden. In paragraaf 3.2 wordt de gehanteerde contentstrategie globaal beschreven op basis van de toepassing van XML en XSL een aantal toegepaste mechanismen die het beter behalen van de prestatie-indicatoren bewerkstelligen.

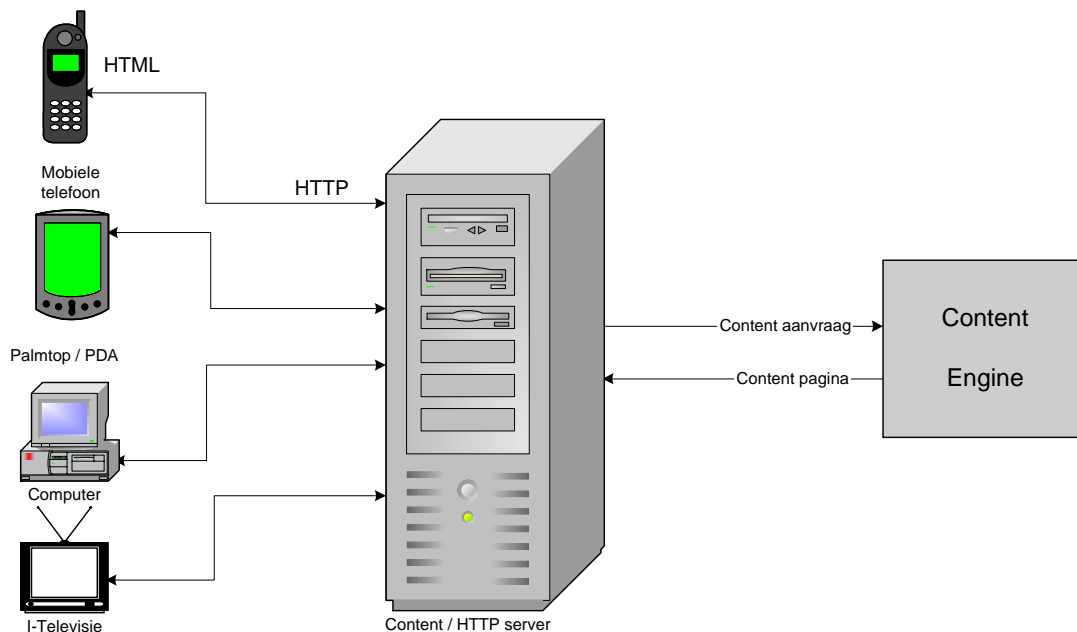
In paragraaf 3.3 zal op basis van de contentpijpleiding de implementatie van de contentstrategie binnen het M-CMS prototype worden beschreven. De contentpijpleiding is de (geautomatiseerde) serie van activiteiten die plaatsvindt om op basis van een HTTP request een stuk content in het juiste formaat en layout te leveren. In paragraaf 3.4 zal de casus Nieuwssite.nl worden beschreven ter illustratie van de opbouw van de contentstructuur en de werking van de contentpijpleiding. De casus levert tevens toetsingsresultaten op, die de basis vormen voor de evaluatie van de contentstrategie binnen het raamwerk van prestatie-indicatoren in paragraaf 3.5.

3.1 Uitgangspunten voor de contentstrategie

In deze paragraaf wordt een aantal uitgangspunten voor de contentstrategie beschreven. Allereerst worden een aantal standaard contentstrategieën beschreven. Vervolgens zal een raamwerk van prestatie-indicatoren worden beschreven, dat de bepaling, toetsing en evaluatie van de contentstrategie ondersteunt. Tenslotte worden enkele externe en projectmatige randvoorwaarden en een afbakening beschreven die bepalend zijn voor de keuze en de inrichting van de contentstrategie.

3.1.1 Standaard contentstrategieën

Het is nuttig om eerst te kijken naar hoe globaal de levering van content aan een apparaat in zijn werk gaat. Een apparaat dat een stuk content wil ontvangen, maakt contact via de user-agent (een applicatie die het mogelijk maakt contact te leggen met het internet) met de content of hypertext transfer protocol (HTTP) server van het betreffende internetadres. Het apparaat kan zijn een PC, een Personal Digital Assistant (PDA), een mobiel toestel of internet-tv. De contentaanvraag (in de vorm van een HTTP request) wordt ontvangen op de HTTP- of contentserver, waar de aanvraag in behandeling wordt genomen en vanaf waar de content wordt geleverd. In figuur 3.1 is hiervan een sterk vereenvoudigde weergave opgenomen, waarbij bijvoorbeeld HTML content wordt teruggegeven. Het proces waarbij op basis van een contentaanvraag een contentpagina wordt geleverd, wordt uitgevoerd door de zogenaamde contentengine, die binnen het M-CMS werkzaam is. We beschouwen de contentengine voor het moment even als een 'black box', die in paragraaf 3.3 nadere invulling krijgt.



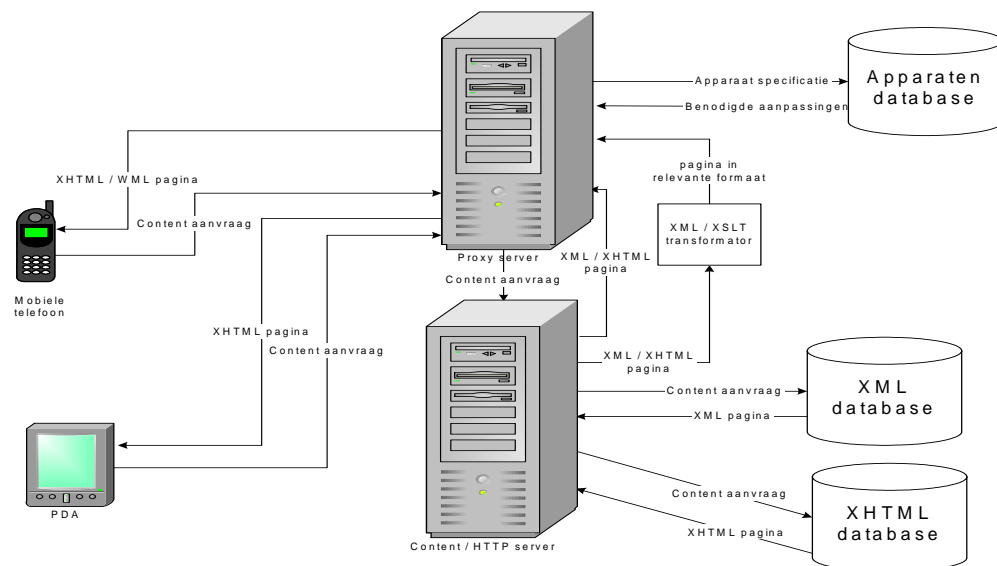
Figuur 3.1: Contentaanvraag door verschillende apparaten

Boumphrey beschrijft een aantal standaard contentstrategieën, die het leveren van content aan verschillende kanalen of media mogelijk maken, op basis waarvan de contentengine vorm kan worden gegeven:

- Contentselectie: het op voorhand creëren van meerdere eindformaten van een webpagina zoals XML, WML, XHTML (een willekeurige vorm of versie van HTML), waarbij de gewenste versie aan de user-agent van een web-browser of apparaat beschikbaar kan worden gesteld;
- Contentcreatie: het dynamisch genereren van het juiste formaat van een webpagina teneinde aan de aanvraag van het doelapparaat te voldoen. Normaliter kan een aantal basisformaten worden gecreëerd (bijvoorbeeld XML en XHTML) en worden minder voorkomende formaten met behulp van XSL transformatie (XSLT);
- Contentaanpassing: het aanpassen van een bestaande contentpagina in zijn oorspronkelijke vorm teneinde te voldoen aan de beperkingen van het doelapparaat. Contentaanpassingen kunnen variëren van het converteren van audio of video formaten tot het volledig aanpassen van de inhoud van de content.

In de onderzoekstaak [vdWouden] is een oplossingsrichting gegeven voor de problematiek (die in paragraaf 3.2 nader wordt beschreven) bij het multi-channel beschikbaar stellen van content middels een gecombineerde contentstrategie. Deze oplossingsrichting combineert de voornoemde strategieën als volgt (zie figuur 3.2, waar de web en i-tv media achterwege zijn gelaten):

- Contentselectie wordt toegepast voor XML en XHTML. Dit zijn standaarden waarmee het overgrote deel van de contentaanvragen kan worden afgehandeld;
- Contentcreatie voor andere standaarden dan XML en XHTML door middel van het transformeren van XML / XHTML contentpagina's door XSLT;
- Contentaanpassing van de getransformeerde of XML /XHTML eindformaten voor specifieke apparaateigenschappen op basis van een apparatendatabase.



Figuur 3.2: Oplossingsrichting voor gecombineerde multi-channel strategie

3.1.2 Raamwerk prestatie-indicatoren voor contentstrategie

In deze paragraaf wordt een raamwerk van prestatie-indicatoren beschreven, die van belang zijn bij de bepaling van de contentstrategie en later de toetsing en evaluatie van de contentstrategie mogelijk maakt. Het raamwerk is ingedeeld naar een vijftal categorieën, te weten de werking van de contentstrategie, de omgang met apparaatbeperkingen, de leveringssnelheid, de ondersteuningsmogelijkheden en de onderhoudbaarheid van het systeem. Het raamwerk is in tabel 3.1 opgenomen en toegelicht. De prestatie-indicatoren zijn niet allen van gelijk belang; de prioriteitstelling van de prestatie-indicatoren zal in paragraaf 3.1.4 aan de orde komen.

Prestatie-indicatoren	Toelichting
1. Werking van de strategie	De mate waarin de juiste content en formaten aan de doel-apparaten wordt geleverd.
a. Volledigheid content	Volledigheid van de content die wordt geleverd.
b. Bruikbaarheid formaat	Bruikbaarheid van het formaat dat wordt geleverd.
2. Apparaatbeperkingen	Hoe wordt omgegaan met de beperkingen aan het apparaat en zijn user-agent?
a. Schermresolutie	Hoe wordt omgegaan met de beperkte omvang van afbeeldingsoppervlakte en de grafische resolutie?
b. Standaarden	Hoe wordt omgegaan met de beperkte ondersteuning voor bepaalde technieken / standaarden van apparaten?
c. Bestandsformaten	Hoe wordt omgegaan met de beperkte ondersteuning voor bestandsformaten door de technische beperkingen?
3. Leveringssnelheid	De snelheid waarmee de content geleverd wordt aan het doelapparaat.
a. Vertraging pijpleiding	De vertragingfactor die de contentpijpleiding veroorzaakt.
b. Versnellingsmethoden	Ondersteunende methoden die zorgen voor snellere levering.
c. Client side speed-up	In hoeverre een snelle afbeelding aan de client side wordt ondersteund (client taken binnen de pijpleiding uitvoeren).
4. Ondersteuningsmogelijkheden	In hoeverre worden binnen de implementatie van de strategie bepaalde mogelijkheden ondersteund?
a. Huidig	In de vorm van wijzigbare systeeminstellingen?
b. Toekomstig	Ten aanzien van toekomstige ontwikkelingen.
5. Onderhoudbaarheid	Hoe goed zijn de verschillende onderdelen van de contentpijpleiding te onderhouden middels systeeminstellingen?

Tabel 3.1: Raamwerk voor prestatie-indicatoren voor de contentstrategie

3.1.3 Externe randvoorwaarde: HTTP request

Voor de invulling van de contentstrategie binnen het M-CMS zijn er enkele randvoorwaarden van belang die de keuze voor de te volgen contentstrategie bepalen. Deze randvoorwaarden betreffen enerzijds externe randvoorwaarden, in het bijzonder de HTTP request, die in deze paragraaf zal worden behandeld. Anderzijds zijn dit praktische randvoorwaarden binnen het project, die in de volgende paragraaf worden behandeld.

Ongeacht welk apparaat of mediatype de contentaanvraag verstuurt, deze komt altijd binnen in de vorm van een HyperText Transfer Protocol (HTTP) client-request header, in het vervolg aangeduidt als een HTTP request. De structuur hiervan is door het World Wide Web Consortium gedefinieerd in *[HTTPrequest]*. In figuur 3.3 is een voorbeeld opgenomen van een HTTP request vanuit een Netscape 2.0 webbrowser *[Boumphrey]*.

```
GET /style/index.htm HTTP 1.0
Accept: text/plain
Accept: text/html
Accept: text/xml
Accept: text/*
Accept: image/jpeg
Accept: image/*
Accept: */*
If-Modified-since: [Date]
User-Agent: Mozilla/2.00
```

Figuur 3.3: Voorbeeld van een webbrowser HTTP request

De eigenschappen zoals deze zijn opgenomen in figuur 3.3 zijn niet compleet, maar zijn wel de meest gebruikte en voor het M-CMS meest nuttige eigenschappen. De regels zoals deze in de HTTP request zijn opgenomen hebben de volgende betekenis:

- GET commando: bevat de locatie (URL) op de betreffende servers content directory voor de gewenste content en de gebruikte versie van het HTTP protocol;
- Accept: geeft aan welke contenttypen de user-agent in de vorm van Multipart Internet Mail Extensions (MIME)-types kan afbeelden. De MIME-types van de Internet Assigned Numbers Authority (IANA) bestaan uit twee onderdelen:
 - Het contenttype: bijvoorbeeld text, image, audio of video. In *[MIME]* is de volledige lijst van de contenttypen en -subtypen gegeven;
 - Het contentsubtype: gegeven achter het contenttype bijvoorbeeld voor text: plain, HTML of XML;**/** geeft aan dat niet genoemde formaten niet moeten worden meegegeven;
- If-Modified-since: wordt door sommige apparaten gegenereerd voor het raadplegen van content op het cache-geheugen van de server, indien de content niet eerder gegenereerd is dan de gegeven datum;
- User-agent: geeft het type user-agent en de versie aan.

Er is ten aanzien van de HTTP request een aantal opmerkingen te plaatsen:

- Figuur 3.3 bevat een voorbeeld van een webbrowser, maar user-agents van mobiele toestellen of PDA's zullen een vergelijkbare request genereren;
- Alle velden in de HTTP request zijn optioneel, waardoor er een zekere mate van onzekerheid bestaat over de parameters die worden meegegeven in een request;
- Er zal op basis van de HTTP request een aantal parameters afgeleid moeten worden, die benodigd zijn voor de contentengine. Indien de velden waarvan kritieke parameters van afgeleid moeten worden niet voorhanden zijn, zal eigenhandig een invulling gegeven moeten worden aan deze parameters.

De HTTP request levert nuttige parameters op basis waarvan de contentengine bepaalt welke contentlocatie wordt gezocht en welk contentformaat en/of mediatype van toepassing is. In paragraaf 3.3.1 zal worden beschreven hoe de velden van de HTTP request gebruikt worden voor het afleiden van de benodigde parameters.

3.1.4 Projectmatige randvoorwaarden en afbakening

De volgende praktische randvoorwaarden en afbakening zijn toegepast bij de ontwerpbeslissingen en de implementatie van het prototype:

- Het onderzoek is beperkt tot apparaten die digitale informatie kunnen ontvangen en verzenden, derhalve tot mediatypen die continue, visueel en interactief zijn, te weten: web, mobiele (WAP-gerelateerde) toestellen, PDA's en i-TV;
- De implementatie van de contentstrategie zal zich vooral richten op het tekst contenttype. Andere MIME contenttypes zoals image en video kenmerken zich binnen de contentdefinitie van [Lohr] nadrukkelijk door inhoud en formaat, maar minder duidelijk door structuur en layout hoewel deze wel af te leiden zijn. Er zal slechts zijdelings gekeken worden naar contenttypes zoals image en video, omdat tekst een alom aanwezig contenttype is en het interessantst is vanwege de uitdrukings- en manipulatiemogelijkheden voor wat betreft structuur en layout;
- Bij het te hanteren raamwerk zijn alle genoemde prestatie-indicatoren van belang, maar er moet een balans worden gevonden tussen deels conflicterende prestatie-indicatoren van bruikbaarheid, volledigheid en leveringssnelheid. Hierbij heeft de bruikbaarheid van de content de hoogste prioriteit; de leveringssnelheid en de volledigheid van de content (met name ten aanzien van contentonderdelen waarvan het formaat niet ondersteund wordt) zijn hieraan ondergeschikt;
- Het gebruik van de ontwerptool Intelligent Objects (IO) binnen het prototyping geeft beperkingen en mogelijkheden aan contentlevering:
 - Goede ondersteuning voor dynamische generatie van content; op basis van een contentaanvraag wordt de content niet uit een XML-database gehaald maar wordt er ter plaatse een XML-pagina gegenereerd in de vorm van een XML-Node op basis van een contentpagina;
 - IO kan (by choice) alleen XML genereren, waardoor de initiële creatie van content plaatsvindt in XML (wat in principe

voldoende is omdat andere contentformaten aansluitend door XSLT middels transformatie kunnen worden gegenereerd).

3.2 Gehanteerde contentstrategie

Op basis van de standaard strategieën en de geschetste randvoorwaarden wordt in dit hoofdstuk op hoofdlijnen de gehanteerde contentstrategie gepresenteerd. Allereerst zal worden beschreven hoe het multi-channel contentmanagement systeem (M-CMS) tot het toepassen van een scheiding tussen inhoud, structuur, layout en formaat komt door toepassing van de XML en XSL(T) standaarden. Daarna zullen enkele mechanismen worden beschreven, die worden toegepast om bij te dragen aan een betere prestatie van de contentstrategie.

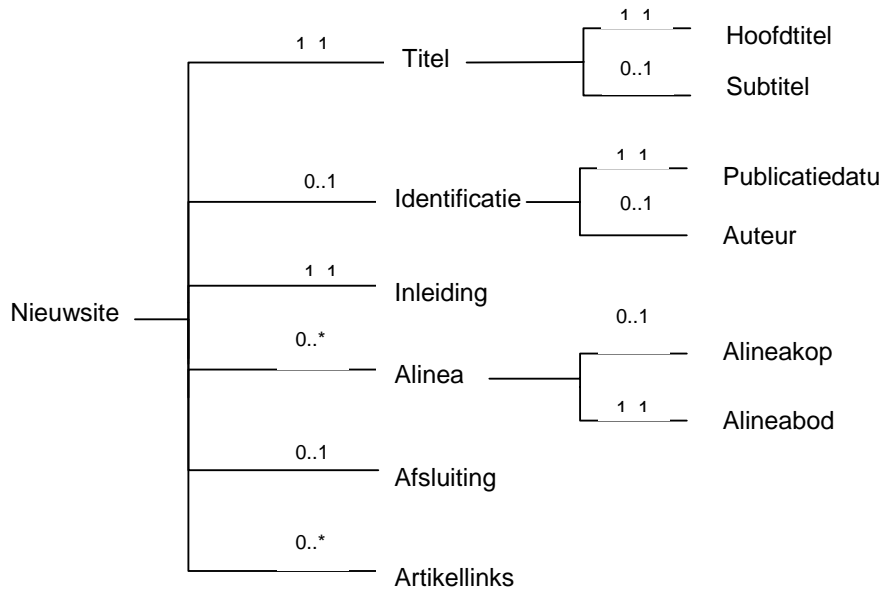
3.2.1 XML en XSL(T) toepassing bij contentcreatie

Zoals in de inleiding werd beschreven, wordt content gedefinieerd als informatie, die zich kenmerkt door inhoud, structuur, layout, mediaformaat en dragermedium (deze laatste wordt buiten beschouwing gelaten vanwege zijn onafhankelijkheid). Door toepassing van de de facto standaarden eXtensible Markup Language (XML) en eXtensible Stylesheet Language (XSL) kan worden gezorgd voor een scheiding van de inhoud, structuur en layout van de content. Bovendien kan de XSL-Transformations (XSLT) standaard zorg dragen voor een transformatie naar andere mediaformaten.

XML is een eenvoudige maar zeer flexibel tekstformaat [*W3C-XML*] dat het mogelijk maakt om informatie over data te definiëren. XML is een markeringstaal die de nadruk legt op het communiceren met behulp van regels en een syntaxis [Martin]. Met XML is het mogelijk de document structuur op te delen in componenten, attributen toe te kennen en onderlinge relaties aan te leggen. XML zelf doet feitelijk niets, slechts markeren en structureren. In combinatie met opmaak- en stijlelementen (XSL) geeft het evenwel enorm veel zeggenschap over de manier waarop pagina's worden weergegeven. XSL is een taal voor het uitdrukken van stylesheets. Net zoals Cascading StyleSheets (CSS) voor HTML doet, kan XSL beschrijven hoe een XML document moet worden afgebeeld. Daarnaast is XSLT beschikbaar: een taal voor het transformeren van XML naar andere formaten [*W3C-XSLT*].

De scheiding van structuur en inhoud kan bij het gebruik van XML door het definiëren van een document type definition (DTD) of een XML Schema [*W3C*], die formeel de contentstructuur beschrijven (bestaande uit componenten die attributen kunnen bevatten). Zowel het XML Schema als het formele DTD, hierna templates te noemen, vormen feitelijk een voorschrift (een redactionele afspraak als het ware) voor het vervaardigen van bijvoorbeeld contentpagina's.

Een voorbeeld van een DTD voor een Nieuwsitem in de vorm van een boomstructuur is opgenomen in figuur 3.4. Hierin zijn tevens multipliciteiten opgenomen: een 0..* relatie met alinea wil zeggen dat er 0 of meer alinea's in de XML-pagina mogen bestaan. Wanneer de inhoud van een contentpagina invulling krijgt, moet deze voldoen aan de voorgeschreven structuur van de DTD.

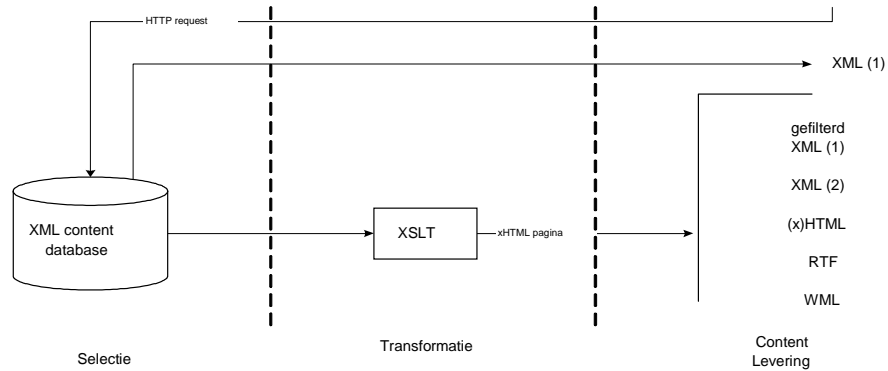


Figuur 3.4: DTD van een Nieuwsitem in de vorm van een boomstructuur

De templates definiëren bepaalde beperkingen maar ook vrijheidsgraden. Figuur 3.4 laat bijvoorbeeld zien dat een nieuwsitem altijd één titel en één inleiding moet bevatten. Anderzijds zijn er vrijheidsgraden aanwezig in de vorm van het aantal alinea's en het facultatief zijn van een subtitel of een alinea kop. De templates kunnen gebruikt worden om de XML contentpagina te valideren; hierbij kan worden gekeken of een 'goed gevormd' XML document (de XML voldoet aan de XML-syntaxisregels) overeenkomt met de template-regels.

Op de gegenereerde XML contentpagina kan vervolgens een XSL stylesheet worden toegepast. Met behulp van XSLT kan binnen de stylesheet het opgemaakte XML document worden getransformeerd naar een ander formaat. Overwegend resulteert er standaard een HTML pagina, maar andere formaten zoals WML, XHTML, cHTML, RTF maar ook XML zelf kunnen betrekkelijk eenvoudig worden gegenereerd. Daarna(ast) wordt met de (CSS) stylesheet per contentcomponent bepaald hoe deze wordt afgebeeld in termen van bijvoorbeeld lettertype, lettergrootte, kleur en andere layoutparameters.

In figuur 3.5 is grofweg het contentstructuur proces weergegeven, waarbij een XML basiscontent wordt getransformeerd door XSLT. De mogelijk eindformaten van de content zijn divers en kunnen verschillende vormen van XML zijn, zowel gefilterd als door toepassing van een ander XMLSchema tijdens de transformatie.

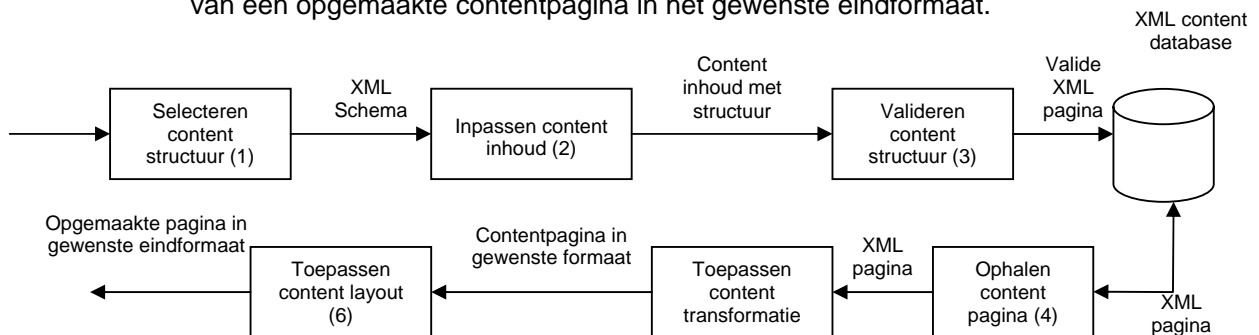


Figuur 3.5: Contentstructuur proces

De toepassing van XML en XSL levert het volgende op binnen het IO datamodel:

- Contentpagina in XML (binnen een site als verzameling van contentpagina's);
- Templates: XML Schema als flexibele voorbeeldstructuur en DTD ter validatie van de contentpagina's;
- Stylesheets: de XSL stylesheet die de transformatie kan bevatten naar een ander mediaformaat en de layout van de XML contentpagina bepaalt.

Daar waar XML de scheiding tussen inhoud en structuur mogelijk maakt, zorgt XSL(T) voor een afhandeling van de layout en het formaat van de content. Figuur 3.6 laat chronologisch de stappen zien die leiden tot het vervaardigen van een opgemaakte contentpagina in het gewenste eindformaat.



Figuur 3.6: Schematisch proces van contentvervaardiging en -levering

De blokken in figuur 3.6 zijn activiteiten, waarvan de eerste drie plaatsvinden binnen het publicatieproces (zie hoofdstuk 4, resulterend in een gepubliceerde contentpagina) en de andere activiteiten binnen de contentpijpleiding, die de volgende inhoud hebben:

- (1) Selecteren contentstructuur: de XML structuur, geformaliseerd door de DTD, wordt in een sjabloon gevat die een voorbeeldstructuur is voor een contentpagina. Binnen het DTD bestaan bepaalde vrijheidsgraden, maar om te voorkomen dat de structuur telkens handmatig moet worden ingegeven op basis van afspraken, wordt een XML Schema sjabloon aangereikt;
- (2) Inpassen contentinhoud: binnen de structuur kan de inhoudelijke content binnen de contentcomponenten worden ingepast, al dan niet in

combinatie met aanpassing van de structuur. De aanpassing van de structuur vindt plaats op basis van de vrijheidsgraden zoals het aanpassen van het aantal alinea's dat de pagina bevat en het verwijderen van niet-gebruikte componenten (hoewel dit niet verplicht is). Hieruit resulteert een XML-bestand waarbij de content inhoud en structuur heeft gekregen;

- (3) Valideren XML contentstructuur: optioneel is het valideren van de XML content op basis van een template (DTD of XML Schema), zodat een valide XML pagina resulteert die in de content database wordt opgeslagen;
- (4) Ophalen contentpagina: het ophalen (dynamisch genereren) van een XML pagina. Deze en de volgende activiteiten worden feitelijk getriggerd door de HTTP request en zullen binnen de contentpijpleiding plaatsvinden en nader worden beschreven in paragraaf 3.3.
- (5) Transformeren contentformaat: gelijktijdig met (6) kan met XSLT het contentformaat bepaald worden dat de resultante is van het toepassen van de stylesheet. Hierbinnen zal naar keuze een HTML, WML, of ander formaat document worden gegenereerd;
- (6) Toepassen content layout: middels een XSL / CSS stylesheet wordt aan de contentcomponenten een layout toegekend, zodat een van layout voorziene XML pagina resulteert (feitelijk een HTML of ander format pagina).

3.2.2 Mechanismen binnen contentstrategie

Gegeven de XML en XSL(T) toepassing en de ontvangst van de HTTP request zoals beschreven in de voorgaande paragrafen, zal een aantal mechanismen worden toegepast, die bijdragen aan een betere prestatie van de contentstrategie. In de volgende paragraaf zal uitvoerig de implementatie van deze mechanismen binnen de zogenaamde contentpijpleiding worden beschreven, die zorgt voor de creatie en levering van een dynamisch op maat gegenereerd stukje content. Figuur 3.1 in herinnering roepende, kunnen binnen de black box van de contentengine de volgende mechanismen worden toegepast:

- Contentselectie op het cache-geheugen;
- Dynamische creatie van een XML-contentpagina;
- Contentselectie van de XML-componenten op basis van mediatype;
- Contentformaat eliminatie op basis van MIME types.

Deze mechanismen worden in de volgende paragrafen beschreven.

3.2.2.1 Contentselectie op het cache-geheugen

Bij binnenkomst van de HTTP request op de server, zijn overwegend altijd wel mogelijkheden beschikbaar om een cache-geheugen te raadplegen dat de eigenschappen heeft van een kleine capaciteit dat snel doorzocht kan worden op het bevatten van een stuk content. Deze versnellingstechniek kan ervoor zorgen dat de contentengine niet behoeft te worden opgestart omdat de content (deels) direct uit de cache gehaald kan worden. De invulling hiervan is niet te geven, aangezien dit afhankelijk is van de specifieke implementatie van het M-CMS binnen een bepaalde organisatie. Het caching mechanisme zal dan ook niet expliciet in de opstelling van de contentpijpleiding worden geïntegreerd, maar wel kunnen de volgende opmerkingen worden geplaatst:

- Er vinden mogelijk op de content / HTTP server, de server aan de clientzijde en binnen de clientbrowser al cache mechanismen plaats. Voor specifiek cachen binnen de contentpijpleiding is het de vraag of dit toegevoegde waarde heeft;
- De cache van de content / HTTP-server is goed te raadplegen voor statische content: in het geval van een bedrijfslogo, dat op vrijwel iedere pagina wordt opgevraagd, is een cachemechanisme zeer nuttig. Voor geparametriseerde of dynamische content is een cache minder goed toe te passen. De mogelijkheid bestaat wel, maar de tests die gedaan moeten worden of en welke onderdelen van de content in de cache bruikbaar zijn weegt mogelijk niet op tegen het creëren van de content;
- Een contentaanvraag kan bestaan uit meerdere contentonderdelen, bijvoorbeeld een stuk tekst en een tweetal plaatjes. Hierbij kan scheiding worden gemaakt tussen onderdelen die zich in de cache bevinden en onderdelen die via de content-engine moeten worden gecreëerd; de plaatjes bevinden zich bijvoorbeeld op de cache terwijl de tekst door de contentengine moet worden gecreëerd. Onderdelen dienen uiteindelijk te worden samengevoegd en geleverd aan het apparaat.

3.2.2.2 Dynamische creatie van de XML-contentpagina

Ten aanzien van de IO tool waarbinnen het M-CMS prototype is gebouwd is in paragraaf 3.1.3 aangegeven dat twee randvoorwaarden van belang zijn: enerzijds wordt het dynamisch genereren van content door de IO tool goed ondersteund en anderzijds genereert de IO tool content in XML-formaat. Gegeven deze randvoorwaarden is het dynamisch laten genereren van contentpagina's door IO een logische keuze, waarbij de volgende argumenten zijn aan te voeren:

- Het gebruik van geparametriseerde, dynamische of interactieve content ten opzichte van statische content neemt steeds meer toe. De tendens is dat websites steeds minder gebruikt worden als statisch uithangbord en steeds meer als interactief medium om bijvoorbeeld klantcontacten te onderhouden of dynamisch medium voor bijvoorbeeld up-to-date reisinformatie of uitslagen;
- Ondersteuning voor geparametriseerde content (content waarin voor bepaalde parameters waarden worden meegegeven waarop bepaalde berekeningen of acties worden uitgevoerd) wordt steeds belangrijker. Dit kan bijvoorbeeld binnen een interactieve sessie zijn, waarbij telkens nieuwe informatie wordt uitgewisseld tussen de client en de server;
- Een mogelijk nadeel van het dynamisch genereren is dat dit voor statische content vertraging oplevert. Dit wordt echter in belangrijke mate ondervangen doordat standaard cache mechanismen (een deel van) de statische content bevatten.

3.2.2.3 Contentselectie van XML componenten binnen M-CMS

Het feit dat er een grote variatie aan apparaten bestaat die aanvragen doen voor content levert potentiële problemen op. De verschillen in apparaateigenschappen liggen ver uit elkaar tussen bijvoorbeeld een PC webbrowser en de browser van een mobiele telefoon, zie tabel 3.2. Die verschillen zijn zo groot dat grote bedrijven er soms toe overgaan om naast de website een aparte site voor mobiele toestellen te bouwen en onderhouden.

Eigenschap	PC webbrowser	Mobiele browser
Afbeeldingsoppervlak	Groot	Klein
Schermpixelresolutie	Hoog	Laag
Grafische ondersteuning	Complexe graphics	Eenvoudige iconen
Ondersteuning standaarden / formaten	In de basis vrij uitgebreid	Beperkt
Beperkingen mens-machine interface	Ondersteuning voor muis (scrollen) en toetsenbord	Beperkt invoermiddel: wel scrollen, geen toetsenbord

Tabel 3.2: Verschillen in eigenschappen webbrowser en mobiele browser

De belangrijkste multi-channel problematiek die hierbij komt kijken is:

- De grote verschillen in afbeeldingsoppervlak, schermresolutie en beperkingen aan mens-machine interface creëren een behoefte om eenzelfde stuk content binnen verschillende apparaten inhoudelijk anders af te beelden. Op een mobiel toestel zou je van een groot stuk tekst eigenlijk alleen de kernpunten willen afbeelden om een oneindig doorbladeren na volgende pagina-onderdelen te voorkomen. Dit onderscheid zal verder worden aangeduid als *organisatorische selectie* van content en is onderwerp van deze paragraaf;
- Door de verschillen in ondersteuning voor (grafische of tekstuele) standaarden en formaten is er behoefte om niet-ondersteunde standaarden niet mee te geven of te elimineren uit de contentpagina. Deze *technische selectie* komt in de volgende paragraaf aan de orde.

De organisatorische selectie van content kan plaatsvinden door:

- Het creëren van een attribuut 'modus' binnen de structuursjablonen van XML Schema, dat een onderscheid maakt in de categorieën mediatypen web, PDA en mobile. Dit attribuut wordt toegekend aan alle contentcomponenten met invulling van een van de toegestane waarden. Een redacteur kan nu bij de invulling van de inhoud naar eigen inzicht wijzigingen aan de attribuutwaarden aanbrengen. Als het bovenstaande in het DTD zou worden vastgelegd, wordt deze vrijheid weggenomen, afhankelijk van wat gewenst is;
- Bij het creëren van een contentpagina op basis van een HTTP request kan een selectie gerealiseerd worden, waarbij een apparaat ingedeeld wordt in een media-categorie en op basis daarvan bepaalde content-componenten ontvangt. Een mobiel toestel zal dan slechts de content-componenten ontvangen zoals in figuur 3.7 is opgenomen op basis van de waarde "all" (voor alle media geschikt) van het 'modus'-attribuut.

```

<nieuwsitem>
  <identificatie>""</identificatie>
  <titel modus="all">
    <hoofdtitel modus="all">""</hoofdtitel>
  </titel>
  <inleiding modus="all">""</inleiding>
</nieuwsitem>

```

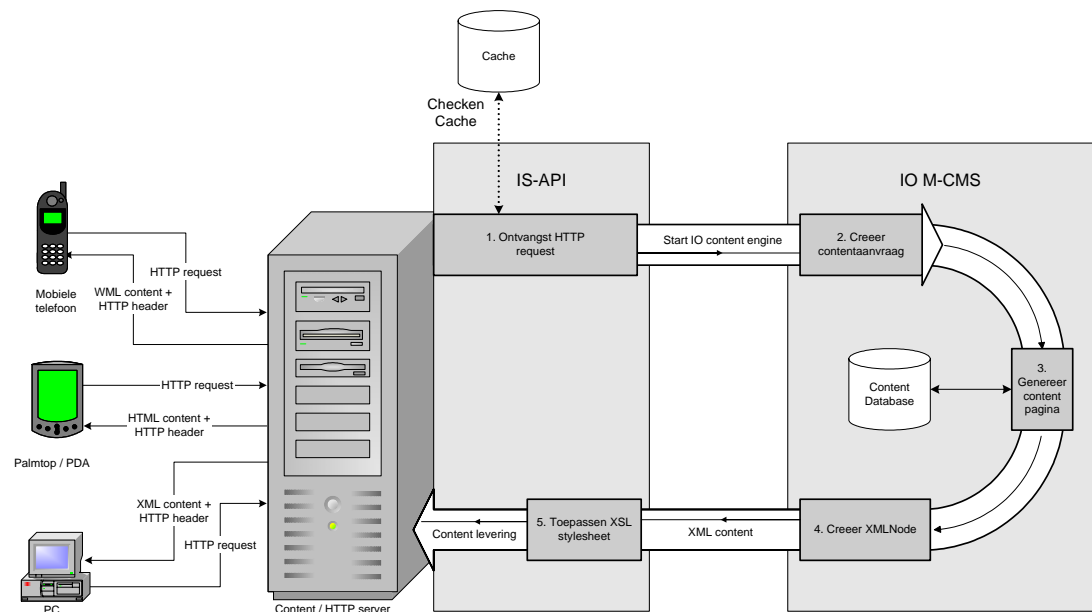
Figuur 3.7: Voorbeeld resultaat selectie Nieuwsitem voor een mobiel toestel

3.2.2.4 Contentformaat eliminatie

Het niet kunnen ondersteunen van verschillende standaarden of formaten kan door een apparaat in de HTTP request worden meegegeven middels de MIME types [MIME]. In dit geval kan XSLT bepalen naar welk formaat wordt getransformeerd en welke content op basis van MIME types wordt geëlimineerd omdat het apparaat dat niet ondersteund. Dit is de technische contentselectie waaraan in de vorige paragraaf werd gerefereerd. Deze keuze is gemaakt om te voorkomen dat (potentieel) niet ondersteunde contenttypes worden meegegeven en vervolgens niet kunnen worden afgebeeld op het apparaat. Dit is zowel een voordeel voor de bruikbaarheid van de content als de leveringssnelheid, maar gaat ten koste van de volledigheid van de content. Helaas is het meegeven van MIME types niet verplicht en is er derhalve geen zekerheid dat deze parameter wordt meegegeven. In de implementatie van de contentpijpleiding wordt nader naar deze problematiek gekeken.

3.3 Inrichting contentpijpleiding

In deze paragraaf zal in detail worden beschreven hoe de contentpijpleiding is gerealiseerd. In figuur 3.8 is de contentpijpleiding schematisch weergegeven. Voor het afhandelen van contentaanvragen worden twee belangrijke onderdelen beschreven, die samen zorgen voor een op maat gemaakte levering van de benodigde content. De ISAPI (Internet Server - Application Programming Interface) is een applicatie die draait op de server waar de HTTP requests binnen komen en die zorg draagt voor de communicatie met het M-CMS dat in de ontwerptool Intelligent Objects (IO) is gebouwd.



Figuur 3.8: Principe van de contentpijpleiding

In analogie met een workflowproces zou je kunnen zeggen dat de ISAPI de front office is voor de ontvangst van de aanvragen en het IO M-CMS de backoffice, die afhandeling van de aanvraag verzorgt. Binnen de contentpijpleiding wordt de benodigde content gecreëerd en aangepast op

basis van de parameters die in de HTTP request worden meegegeven middels de volgende vijf stappen (figuur 3.8):

1. Ontvangst HTTP request: ontleden van de HTTP request, het checken van het cache geheugen en het opstarten van het M-CMS met de ontleedde aanvraag;
2. Creeer contentaanvraag: het aanmaken van een object contentaanvraag in IO;
3. Genereer contentpagina: de contentpagina wordt vanuit de content database dynamisch gegenereerd naar een XML-contentpagina;
4. Creeer XML-Node: de contentpagina en contentaanvraag worden samen als XML-Node aangeboden aan de ISAPI;
5. Toepassen XSL-Stylesheet: het toepassen van een stylesheet waarmee de contentpagina een bepaalde layout (XSL) en formaat (XSLT) krijgt. Aansluitend kan de content worden geleverd met een HTTP header aan het apparaat.

In de volgende paragrafen zullen de stappen nader worden beschreven.

3.3.1 Stap 1: Ontvangst HTTP request

In paragraaf 3.1.3 is de structuur van de HTTP request beschreven. De ontvangst van de HTTP request vindt plaats in de ISAPI, die ervoor zorgt dat er een aantal acties worden ondernomen:

- Ontleden HTTP request en vertalen naar een aantal parameters die binnen de contentpijpleiding gebruikt worden. Tabel 3.3 geeft de vertaling van de HTTP velden naar de parameters zoals deze worden aangeboden aan het IO M-CMS;
- Checken cache: zoals in paragraaf 3.2.2.1 werd aangegeven, wordt er geen directe caching binnen de ISAPI toegepast, maar zijn er standaard mogelijkheden binnen servers die deze versnellingsstechniek toepassen;
- Opstarten van de contentengine van het M-CMS door het aanroepen van de procedure 'afhandelenContentAanvraag'. De parameters worden aangeboden aan het IO M-CMS in de vorm van een XML-bestand waarin de parameters als XML-nodes worden meegegeven.

HTTP veld	Bevat	ISAPI actie	Ontvangst op het IO M-CMS
GET	URL	Maak een domein en pagina locatie → IO M-CMS	Node Query, die bestaat uit nodes domein en pagina
	HTTP versie	Zorgt ervoor dat ISAPI de HTTP request goed kan lezen	-
Accept	MIME-types en -subtypes	Maak een lijst met geaccepteerde contenttypen → IO M-CMS	Node MIME-types met geaccepteerde contenttypen
If-modified-since	Datum	Check de cache op aanwezigheid van de content, indien geplaatst na [datum]	-
User-agent	User-agent type en versie	Maak een string useragent → IO M-CMS	Node user-agent

Tabel 3.3: Ontleding van de HTTP request

3.3.2 Stap 2: Creeer contentaanvraag

Wanneer de contentengine van het M-CMS wordt opgestart door de ISAPI, wordt allereerst een object contentaanvraag aangemaakt, dat een representatie is van de HTTP request binnen het M-CMS. De contentaanvraag wordt aangemaakt middels de procedure 'creeerContentaanvraag' die door de contentengine wordt aangeroepen. Aan de attributen van de contentaanvraag worden de waarden van de XML-Nodes toegekend die door de ISAPI worden meegegeven. De contentaanvraag heeft de volgende attributen:

- Contentformaat (String) met domein: {(XML),(HTML),(WML)};
- Mediatype (String) met domein: {"Mobile"},{"PDA"},{"web"};
- 'Ondersteunde MIME types' (List of Strings): ondersteunde contenttypes;
- Sitedomein (String): het serverdeel van de URL;
- URL (String): het URL van de contentpagina;
- User-agent (String).

3.3.3 Stap 3: Genereer contentpagina

Binnen de procedure afhandelenContentaanvraag van de contentengine is de volgende stap het dynamisch creëren van een contentpagina op basis van een XML-pagina in de contentdatabase. De XML-pagina bevat zowel de inhoud als de structuur van de content. De volgende stappen worden ondernomen:

- Het opzoeken van de contentpagina op basis van het URL;
- Het bepalen van het mediatype (dat is per definitie een af te leiden attribuut) op basis van de user-agent (direct) of via de look-up tabel die in de kennisbank wordt opgenomen (al dan niet met variabelen zoals resolutie);
- Het opspannen van een XML-boom waarbij de nodes dynamisch worden geëvalueerd; nodes kunnen platte tekst bevatten, maar ook complexere structuren als berekeningen (al dan niet op basis van meegegeven parameters);
- Voor de gegenereerde XML-Node een selectie van contentcomponenten laten plaatsvinden (of juist het verwijderen van nodes die niet voldoen aan het mediatype). Deze organisatorische selectie is een voorzorgsmaatregel voor het afbeelden van een te grote hoeveelheid content op apparaten die een kleine resolutie / schermgrootte hebben;
- Het resultaat is een XML node die de gecreëerde content bevat van de pagina.

3.3.4 Stap 4: Creeer XML-Node

In de vorige stap werd een XML-Node gecreëerd die de contentcomponenten van de pagina bevat. Hieraan moet een node worden bevestigd die de oorspronkelijke contentaanvraag bevat met mogelijk een aantal nader ingevulde attributen. Zo werd in stap 3 het mediatype op basis van de user-agent bepaald (gebruikt voor de selectie van contentcomponenten). Teneinde in de volgende stap de stylesheet toe te passen, is het tevens nodig om te bepalen wat het contentformaat moet worden. Dit kan bepaald worden aan de hand van de MIME-types die door het doelapparaat ondersteund worden of andere parameters die zijn meegegeven. Hierbij spelen voor het afleiden van het meest geschikte media formaat de volgende overwegingen een rol:

- De velden van de HTTP request zijn allen optioneel, dus er bestaat geen zekerheid over welke parameters zijn meegegeven. Wel kunnen parameters die potentieel beschikbaar zijn op geschiktheid voor het afleiden van het formaat worden afgegaan. Dit zijn:
- Het afleiden van de MIME types: bij de MIME types worden alle ondersteunde formaten meegegeven. Hierbij kan binnen het M-CMS op basis van een prioriteitsstelling bepaald worden welk formaat het meest geschikt is voor het betreffende mediatype. Bijvoorbeeld voor een mobiel toestel kan de volgorde zijn in aflopende prioriteit: WML, XHTML, HTML etc.;
- Het afleiden van de andere parameters zoals media type of user-agent: hoewel dit niet echt wenselijk is, kan in het geval dat geen MIME types zijn meegegeven een andere parameter dienen om het formaat af te leiden. Het mediatype kan hiervoor dienen op basis van de prioriteitsstelling van het vorige punt. Ook de user-agent zou hiervoor kunnen dienen, maar het gaat op dit punt te ver om dat uit te werken.

3.3.5 Stap 5: Toepassen XSL-stylesheet en leveren content

Wanneer het IO M-CMS de procedure 'afhandelenContentaanvraag' geheel heeft uitgevoerd, geeft deze aan de ISAPI een XML Node terug met daarin de content (inhoud en structuur) en de parameters van de contentaanvraag. De ISAPI kan met behulp van deze XML Node de volgende acties ondernemen:

- Het selecteren van de betreffende stylesheet, die over de content kan worden gelegd, waarbij gecombineerd kan worden uitgevoerd:
 - XSL stylesheet toepassing voor de opmaak van de content;
 - XSLT toepassing voor transformatie naar juiste formaat;
- Content geschikt maken voor versturing door toevoeging van een HTTP header naar het betreffende apparaat. Hierbij kan het zijn dan enkele contentonderdelen moeten worden samengevoegd, zoals de tekst uit het M-CMS en plaatjes die uit de cache kunnen worden gehaald.

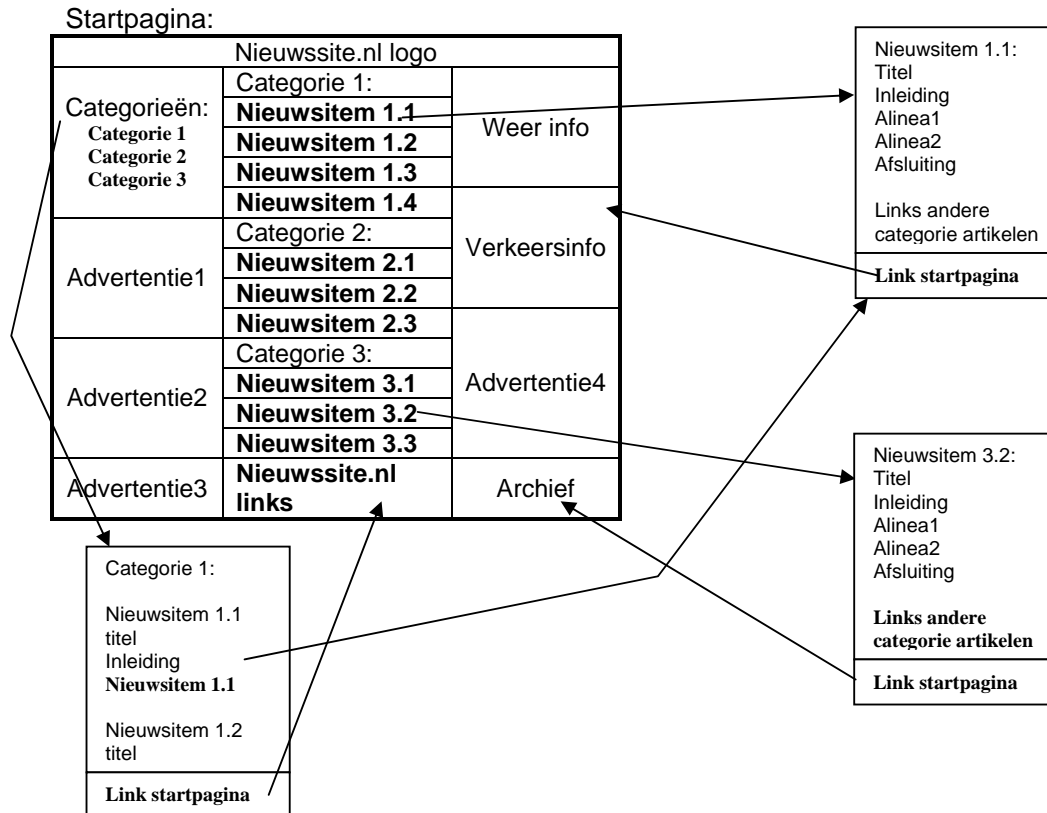
3.4 Casus Nieuwssite.nl

In deze paragraaf wordt aan de hand van de casus Nieuwssite.nl de opbouw van de contentstructuur en de werking van de contentpijpleiding geïllustreerd.

3.4.1 Beschrijving casus Nieuwssite.nl

Nieuwssite.nl is een fictieve site die door de gelijknamige fictieve organisatie van content wordt voorzien en onderhouden. De site is vergelijkbaar met een nieuwssite als nu.nl voor wat betreft content, structuur en organisatie.

De organisatie nieuwssite.nl heeft als kernactiviteiten het leveren van content aan en het onderhouden van de nieuwssite. Binnen de organisatie zijn dus zeer expliciet het publicatieproces en de beheeractiviteiten aanwezig. De organisatie wil dat de website tevens geschikt is voor publicatie naar andere kanalen dan het internet, zoals naar pocket PC's (PDA's) en mobiele telefoons via WAP of een vergelijkbaar protocol. In figuur 3.9 is een grove schematische opzet van de website opgenomen, waarbij een enkele link tussen webpagina's is opgenomen. De organisatie is per definitie zeer contentintensief en het contentverloop is hoog; nieuwsitems staan zelden langer dan een halve dag op de startpagina (hoewel ze wel in het archief behouden blijven).



Figuur 3.9: Sitestructuur van nieuwssite.nl

Linksboven in figuur 3.9 is de startpagina weergegeven, die nieuwsitems, categorieën en advertenties bevat. De links (vetgedrukte regels) naar andere webpagina's zijn door pijlen gevisualiseerd naar nieuwsitems (rechter blokken) en een categorie (onderste blok), die de samenvatting van de nieuwsitems bevat. De structuur van de content zal in paragraaf 3.4.2 worden toegelicht aan de hand van een XML Schema en het DTD voor het nieuwsitem. In paragraaf 3.4.3 zal aan de hand een voorbeeld HTTP request de werking van de contentpijpleiding worden geïllustreerd.

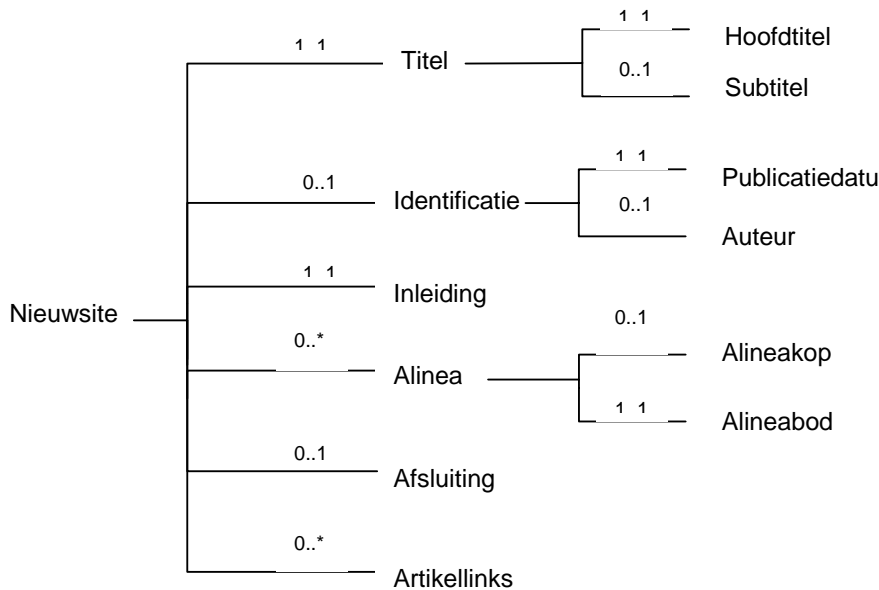
3.4.2 Inrichting van contentstructuur

De XML-contentstructuur, die ervoor zorgt dat contentpagina's in de database zowel inhoud als structuur hebben, wordt vorm gegeven door het XML Schema en de DTD. Het XML Schema is een sjabloon dat een valide structuur bevat, die evenwel een aantal vrijheidsgraden heeft binnen het DTD dat het formele voorschrift voor de structuur bevat. Hierbij kan het nieuwsitem goed dienen als voorbeeld; voor ieder nieuwsitem (de rechter blokken afgebeeld in figuur 3.9) kan als basis het XML Schema uit figuur 3.10 dienen, waarbij het bijvoorbeeld mogelijk is om alinea's toe te voegen of verwijderen en waarbij het facultatief is om bepaalde velden in te vullen. De regels die hierop van toepassing zijn, zijn middels het DTD van figuur 3.11 weergegeven. De DTD van het nieuwsitem is hierbij opgenomen als een XML-boom met multipliciteiten; een 0..* relatie met alinea wil zeggen dat er 0 of meer alinea's in de XML-pagina mogen bestaan.

```

<nieuwsitem>
  <titel modus="all">
    <hoofdtitel modus="all">""</hoofdtitel>
    <subtitel modus="PDA+web">""</subtitel>
  </titel>
  <identificatie></identificatie>
  <inleiding modus="all">""</inleiding>
  <body modus="PDA+web">
    <alineamodus="PDA+web">
      <alineakop>""</alineakop>
      <alineabody>""</alineabody>
    </alineamodus>
    <alineamodus="PDA+web">
      <alineakop>""</alineakop>
      <alineabody>""</alineabody>
    </alineamodus>
    <alineamodus="web">
      <alineakop>""</alineakop>
      <alineabody>""</alineabody>
    </alineamodus>
    <alineamodus="web">
      <alineakop>""</alineakop>
      <alineabody>""</alineabody>
    </alineamodus>
  </body>
  <afsluiting modus="web">""</afsluiting>
  <artikellinks>10 laatste artikelen</artikellinks>
</nieuwssitem>
  
```

Figuur 3.10: XML Schema van een nieuwsitem pagina



Figuur 3.11: Boomstructuur van de DTD van een nieuwsitem

```

<nieuwsitem>
  <titel modus="all">
    <hoofdtitel modus="all"> Nederlandse handlanger Saddam Hussein opgepakt.</hoofdtitel>
    <subtitel modus="PDA+web">""</subtitel>
  </titel>
  <identificatie>Zaterdag 18 december 2004, 17:02:34</identificatie>
  <inleiding modus="all"> ROTTERDAM - De Nationale Recherche heeft maandag in Amsterdam de
  62-jarige F. van A. aangehouden die wordt verdacht van betrokkenheid bij de door Saddam
  Hussein gepleegde oorlogsmisdrijven en genocide. </inleiding>
  <body modus="PDA+web">
    <alineamodus="PDA+web">
      <alineakop>""</alineakop>
      <alineabody>Het landelijk parket van het Openbaar Ministerie vermoedt dat de Nederlander
      duizenden tonnen grondstoffen voor chemische wapens tussen 1984 en 1988 aan het voormalige
      regime in Bagdad heeft geleverd.</alineabody>
    </alineamodus>
    <alineamodus="PDA+web">
      <alineakop> Chemische wapens</alineakop>
      <alineabody>De toenmalige Iraakse regering heeft de chemische wapens ingezet in de
      oorlog met Iran en tegen de Koerdische bevolking in Noord-Irak. De Nationale
      Recherche voert dit strafrechtelijk onderzoek in samenwerking met de FIOD-ECD uit op
      basis van de Wet Oorlogsstrafrecht en de Uitvoeringswet
      Genocideverdrag.</alineabody>
    </alineamodus>
    <alineamodus="web">
      <alineakop> Zaken</alineakop>
      <alineabody> Het onderzoek heeft volgens het landelijk parket aan het licht gebracht dat
      Van A. vermoedelijk rechtstreeks zaken deed met de toenmalige autoriteiten in Irak. Hij
      maakte daarbij gebruik van financiële schijnconstructies om de betrokkenheid van
      zichzelf en Irak buiten het zicht te houden. De Nederlander bediende zich van een
      Panamese onderneming die in het Zwitserse Lugano was gevestigd, laat het landelijk
      parket weten. </alineabody>
    </alineamodus>
    <alineamodus=["web"]>
      <alineakop> Zenuwgassen </alineakop>
      <alineabody> De grondstoffen voor mosterdgas en zenuwgassen waren afkomstig uit
      Japan en de Verenigde Staten. Het onderzoek richt zich op 36 leveringen, waaronder
      een tweetal zendingen fabrieksmaterialen voor Irak. Volgens de Verenigde Naties is de
      Nederlander een van de grootste tussenhandelaren in het verkrijgen van chemische
      materialen door Irak.</alineabody>
    </alineamodus>
    <alineamodus=["web"]>
      <alineakop>""</alineakop>
      <alineabody> In verband met de verboden export naar Irak stelde US Customs in
      Baltimore enkele jaren geleden al een strafrechtelijk onderzoek in. Uit het Amerikaanse
      onderzoek bleek dat de Nederlander betrokken was bij een viertal ladingen Thiodyglycol
      (TDG) die vanuit de Verenigde Staten naar Europa waren verscheept. Via de havens
      van Antwerpen en Aqaba in Jordanië bereikten de grondstoffen voor chemische wapens
      Irak. </alineabody>
    </alineamodus>
    <alineakop>Verzoek</alineakop>
    <alineabody> Op verzoek van de Verenigde Staten werd de Nederlander op 26 januari
    1989 aangehouden in Milaan. Maar nadat na twee maanden zijn uitleveringsdetentie
    was geschorst vluchtte de man naar Irak, waar hij verbleef tot de inval van de militaire
    coalitie in 2003. Daarna vertrok hij via Syrië naar Nederland. </alineabody>
  </alineamodus>
  </body>
  <afsluiting modus="web"> Uit verschillende bronnen valt af te leiden dat de Nederlander op de
  hoogte was van de bestemming en het uiteindelijke doel van de door hem geleverde grondstoffen,
  stelt het OM. Een van de bekendste aanvallen met chemische wapens is de vernietiging van het
  Koerdische stadje Halabja op 16 maart 1988. Tijdens deze aanval werden naar schatting 5000
  mensen gedood.</afsluiting>
  <artikellinks></artikellinks>
</nieuwsitem>

```

Figuur 3.12: XML nieuwsitem

De content, zoals deze in de contentdatabase binnen het M-CMS wordt opgenomen, is een XML document met inhoud en met structuur gebaseerd op het XML Schema. Een voorbeeld hiervan is opgenomen in figuur 3.12 voor het nieuwsitem “Nederlandse handlanger Saddam Hussein opgepakt”.

3.4.3 Werking en resultaat van de contentpijpleiding

Gegeven de beschreven content structuur kan nu op basis van een tweetal HTTP requests de werking en resultaten van de contentpijpleiding worden getoond. In figuur 3.13 is een voorbeeld van de HTTP request van een webbrowser opgenomen en in figuur 3.14 de HTTP request van een PDA browser. De URL (/nieuws.jsp?n=457584&c=23) die wordt meegegeven valt binnen het domein van de server waarop de HTTP request binnenkomt, <http://www.nieuwssite.nl> in dit geval. De URL is de identificatie van het nieuwsitem “Nederlandse handlanger Saddam Hussein opgepakt” dat in figuur 3.12 is beschreven.

```
GET /nieuws.jsp?n=457584&c=23 HTTP 1.1
Accept: text/plain
Accept: text/rtf
Accept: text/xhtmll
Accept: text/xml
Accept: text/*
Accept: image/gif
Accept: image/jpeg
Accept: image/tiff
Accept: image/*
Accept: */*
If-Modified-since: [Date]
User-Agent: Mozilla/4.0
```

Figuur 3.13: HTTP request voorbeeld van een webbrowser

```
GET /nieuws.jsp?n=457584&c=23 HTTP 1.0
Accept: text/plain
Accept: text/html
Accept: text/xhtmll
Accept: image/jpeg
Accept: text/*
Accept: */*
User-Agent: Microsoft Pocket Internet
Explorer/0.6
```

Figuur 3.14: HTTP request voorbeeld van een PDA browser

Als de HTTP request van de PDA browser als voorbeeld wordt genomen, worden de stappen van de contentpijpleiding als volgt uitgevoerd:

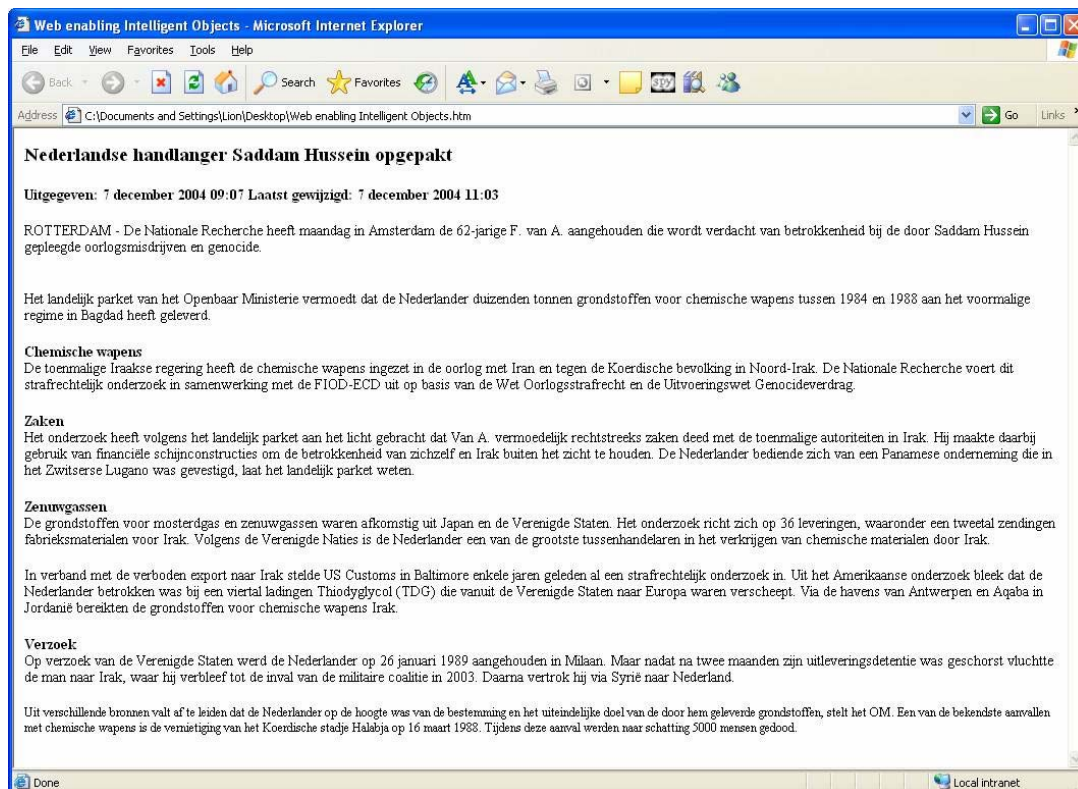
1. Ontvangst HTTP request: de ISAPI ontleed de HTTP request naar een XML Node waarmee de procedure `afhandelenContentaanvraag` binnen IO wordt opgestart. De cache wordt niet geraadpleegt vanwege het ontbreken van een `If-Modified-Since` veld. De XML Node bevat de volgende elementen:
 - o URL bestaande uit een domein "`http://www.nieuwssite.nl`" en een pagina "`/nieuws.jsp?n=457584&c=23`";
 - o MIME types bestaande uit `{("text/plain"), ("text/html"), ("text/xhtml"), ("image/jpeg")}`;
 - o User-agent "`Microsoft Pocket Internet Explorer/0.6`".
2. Creeer contentaanvraag: er wordt een object `Contentaanvraag` aangemaakt in IO middels het uitlezen van de XML Node, waarbij de attributen 'Ondersteunde MIME types', Sitedomein, URL en Useragent worden ingevuld, maar het contentformaat en mediatype worden later afgeleid;
3. Genereer contentpagina: de contentpagina wordt vanuit de content database dynamisch gegenereerd naar een XML-contentpagina. Het mediatype wordt op basis van de referentielijst bepaald als PDA, zodat enkele contentcomponenten worden verwijderd, hetgeen de XML van figuur 3.15 oplevert;
4. Creeer XML-Node: het contentformaat wordt bepaald op basis van de MIME-types: voor PDA wordt bijvoorbeeld een prioriteitstelling (XML, XHTML, HTML, anders) aangehouden zodat XHTML als contentformaat resulteert. Bovendien kan op basis van de contentpagina het adres van het bijbehorende XSL-stylesheet worden meegegeven. De contentpagina en nader ingevulde contentaanvraag worden samen als XML-Nodes aangeboden aan de ISAPI;
5. Toepassen XSL-Stylesheet: het toepassen van het XSL stylesheet waarmee de contentpagina een bepaalde layout en formaat krijgt aangemeten. Aansluitend kan de content worden geleverd met een HTTP header aan het apparaat.

In figuur 3.17 (links) staat van de PDA request (in dit geval een Pocket PC) de contentpagina afgebeeld voor het nieuwsitem "Nederlandse handlangers Saddam Hussein opgepakt". Ter vergelijking is in figuur 3.16 de webbrowsers request van de contentpagina van hetzelfde nieuwsitem afgebeeld. De afbeelding zoals deze in een mobile browser wordt getoond, staat rechts in figuur 3.17 opgenomen.

```

<nieuwsitem>
  <titel modus="all">
    <hoofdtitel modus="all"> Nederlandse handlangers Saddam Hussein
    opgepakt.</hoofdtitel>
    <subtitel modus="PDA+web">""</subtitel>
  </titel>
  <identificatie>Uitgegeven 7 december 2004 09:07</identificatie>
  <inleiding modus="all"> ROTTERDAM - De Nationale Recherche heeft maandag in
  Amsterdam de 62-jarige F. van A. aangehouden die wordt verdacht van betrokkenheid
  bij de door Saddam Hussein gepleegde oorlogsmisdrijven en genocide. </inleiding>
  <body modus="PDA+web">
    <alineamodus="PDA+web">
      <alineakop>""</alineakop>
      <alineabody>Het landelijk parket van het Openbaar Ministerie vermoedt dat de
      Nederlander duizenden tonnen grondstoffen voor chemische wapens tussen 1984 en
      1988 aan het voormalige regime in Bagdad heeft geleverd.</alineabody>
    </alineamodus="PDA+web">
    <alineamodus="PDA+web">
      <alineakop> Chemische wapens</alineakop>
      <alineabody>De toenmalige Iraakse regering heeft de chemische wapens
      ingezet in de oorlog met Iran en tegen de Koerdische bevolking in Noord-Irak.
      De Nationale Recherche voert dit strafrechtelijk onderzoek in samenwerking
      met de FIOD-ECD uit op basis van de Wet Oorlogsstrafrecht en de
      Uitvoeringswet Genocideverdrag.</alineabody>
    </alineamodus="PDA+web">
  </body>
  <artikellinks></artikellinks>
</nieuwssitem>
  
```

Figuur 3.15: XML pagina op basis van mediatype PDA



Figuur 3.16: Webbrowser screenshot voor nieuwssitem



Figuur 3.17: Browser shots van een Pocket PC (links) en een Mobile (rechts)

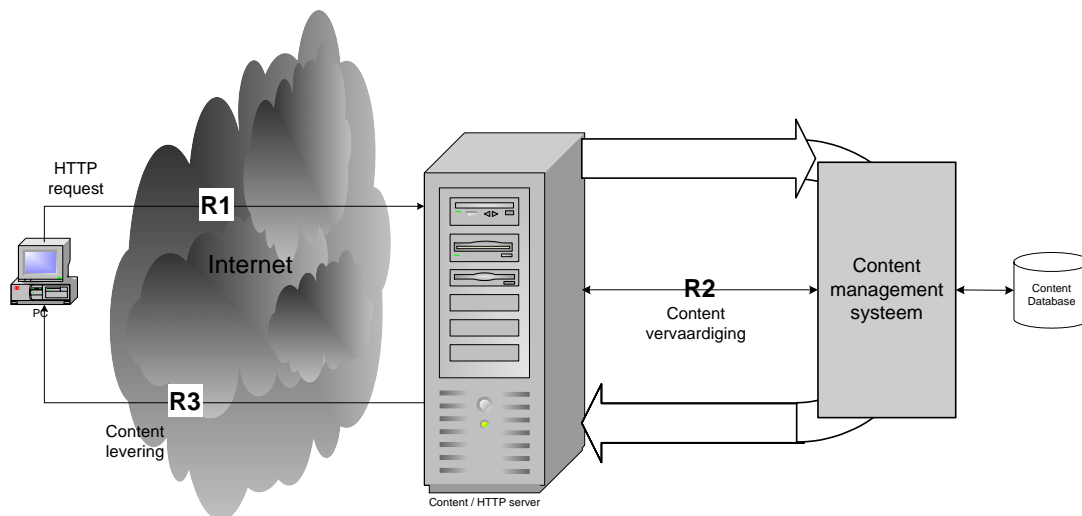
3.5 Evaluatie contentstrategie

In paragraaf 3.1.2 is het raamwerk met prestatie-indicatoren voor de toetsing van een contentstrategie gepresenteerd. Het raamwerk is geschikt om contentstrategieën te beschrijven, te evalueren en te vergelijken. Het vergelijken van de gehanteerde contentstrategie met andere strategieën zoals die beschreven zijn in paragraaf 3.1.1. is evenwel niet goed mogelijk. Deze strategieën zijn te globaal om uitspraak te kunnen doen over hun prestaties. Hiervoor zou een nadere invulling en implementatie van de strategieën moeten plaatsvinden. Er zijn bovendien geen vergelijkingscijfers beschikbaar die een vergelijking van de contentpijpleiding met een andere implementatie mogelijk maakt.

Het is evenwel goed mogelijk om de toegepaste contentstrategie te beschrijven en te evalueren, omdat de casus Nieuwssite een toetsing van presentatiemiddelen en resultaten van de contentpijpleiding heeft opgeleverd. In het bijzonder wordt in paragraaf 3.5.1 de vertraging van de pijpleiding toegelicht aan de hand van enkele meetresultaten. In paragraaf 3.5.2 worden mogelijkheden voor het fijnstellen van de pijpleiding toegelicht. In paragraaf 3.5.3 worden de resultaten samengevat in het raamwerk van prestatie-indicatoren.

3.5.1 Toetsing leveringsnelheid

In deze paragraaf zullen enkele restresultaten worden gegeven, die een indicatie geven van de mate van vertraging die de contentpijpleiding veroorzaakt. Dit wordt vergeleken met de startpagina van iCure en van de nieuwssite van de NOS. In figuur 3.18 is schematisch aangegeven wat de meetbare onderdelen zijn binnen de testopstelling. De metingen aan de testopstelling werd verricht met de tool Microsoft Application Center Test (MS ACT) en zijn voor de NOS en de iCure site over het volledige traject (R1+R2+R3) uitgevoerd. Voor het M-CMS kon vanwege de ISAPI realisatie en de simulatie van HTTP requests slechts route R2 worden gemeten, maar dit geeft desondanks een aardige indicatie.



Figuur 3.18: Schematische testopstelling

In figuur 3.18 is aangegeven dat er drie onderdelen zijn (in dit voorbeeld is enkel de PC als medium gegeven, maar mobiel en PDA zijn net zo goed mogelijk):

- R1. HTTP request: de verzending van de contentaanvraag over het internet;
- R2. Contentvervaardiging: de tijd die een server nodig heeft om de content te laten vervaardigen (met achterliggend CMS en database);
- R3. Contentlevering: het versturen van de content naar het doelapparaat.

De MS ACT tool is gebruikt om gedurende 10 minuten herhaald een contentpagina aan te vragen en deze kon de volgende bruikbare parameters meten:

- Gemiddeld aantal afgehandelde HTTP requests per seconde (A);
- Gemiddelde tijd tot eerste byte (msec): een benadering van de tijd van R1+R2 (B). Hierbij zit tevens de tijd voor de verzending van de eerste byte van R3, maar er wordt van uitgegaan dat dit beperkt is;
- Gemiddelde tijd tot laatste byte (msec): gelijk aan de tijd van R1+R2+R3 (C);

Daarnaast is de 'pingtijd' gemeten: het versturen van een lege request die direct van de server terugkomt (een benadering voor de tijd van R1+R3, hoewel de request iets kleiner is). Tabel 3.4 geeft de afleiding van de drie meetwaarden uit figuur 3.18. In tabel 3.5 zijn de metingen aangegeven van de testsites van de NOS (www.nos.nl/nieuws/index.html) en iCure. Bij iCure is

zowel statische (www.icure.nl/images.jpg) als dynamische content (www.icure.nl/icure/page.jsp) gemeten. Voor het M-CMS kon enkel in de simulatie de waarde van R2 worden bepaald, hetgeen staat weergegeven in tabel 3.6. Hierbij zijn metingen verricht met en zonder toepassing van XSLT transformatie, om tevens een beeld te scheppen van de tijdskosten van de transformatie. Dit ook voor een periode van 10 minuten.

Meetwaarde	Definitie / berekening
A	Gemiddeld aantal afgehandelde HTTP requests per seconde
B	Gemiddelde tijd tot eerste byte (msec) $\approx R1 + R2$
C	Gemiddelde tijd tot laatste byte (msec) $= R1 + R2 + R3$
P	Pingtijd $\approx R1 + R3$ (voor een lege request)
R1	$P / 2$ (uitgaande dat de pingtijd heen bij benadering even lang is als terug)
R2	$B - R1$
R3	$C - B$

Tabel 3.4: Bepaling van meetwaarden afgeleid van MS ACT resultaten

Site	P (ms)	A (#/s)	B (ms)	C (ms)	R1 (ms)	R2 (ms)	R3 (ms)
NOS	13	6,78	133,20	588,39	6,50	126,70	455,19
iCure stat	21	29,32	65,95	80,81	10,50	55,45	14,68
iCure dyn	21	11,07	204,70	360,14	10,50	194,20	155,44

Tabel 3.5: Gemiddelde tijden testsites

Site	A (#/s)	R2 (ms)
M-CMS transformatie, 4 users	109,64	15,93
M-CMS geen transformatie, 2 users	105,36	12,55
M-CMS geen transformatie, 4 users	126,43	12,42

Tabel 3.6: Gemiddelde tijden M-CMS

Uit de meetgegevens zijn enkele globale conclusies te trekken:

- De tijd die het M-CMS nodig heeft voor het vervaardigen van content (meetresultaat R2) is in vergelijking met de testsites zeer concurrerend;
- De tijd in het algemeen die het vervaardigen van content (R2) vergt in verhouding tot het verzenden van een request of een contentpakket (R3) is klein, waardoor het CMS niet de bottle-neck zal zijn van de leveringssnelheid. Wel wegen factoren als omvang van de content en het afbeeldingsgemak (client side speed-up) zwaar op de leveringssnelheid en afbeeldingssnelheid van het apparaat.
- De vertragingen over het Internet zijn significant, maar zijn afhankelijk van een diversiteit aan deels niet te beïnvloeden factoren, zoals bandbreedte, servereigenschappen en effectiviteit van cachingmechanismen;
- De tijd van de transformatie is 22% van de totale vervaardigingstijd, hetgeen te rechtvaardigen is door de verhoogde bruikbaarheid van de content en de client side speed-up (helaas niet te meten afbeeldingssnelheid) doordat een formaat wordt meegezonden dat direct (en het best) is af te beelden door de user agent.

Hierbij dient nog te worden opgemerkt dat bij de metingen van het M-CMS geen expliciete caching is gebruikt (anders dan op de browser client aanwezig), hetgeen nog additionele voordelen kan bieden.

3.5.2 Fijnstellen contentpijpleiding

Binnen het prototype is een aantal parameters beschikbaar waarmee het mogelijk is om de werking van de contentengine fijn te stellen. Deze parameters kunnen in wijzigbare vorm opgenomen worden als systeeminstellingen om af te stemmen voor een specifieke vorm van content, afhankelijk van het type organisatie, hun content, het type klanten of sitebezoekers en wat dies meer zij. De volgende parameters lenen zich hiervoor:

- Categorisering van mediatypes: deze is nu grof gekozen door de driedeling web-PDA-mobile. Geavanceerdere categorisering zou kunnen plaatsvinden op basis van de schermresolutie (indien voorhanden binnen de apparatendatabase) of van de useragents. Hiervoor is uitgebreid testen van een omvangrijke casus nodig;
- In het prototype is de keuze gemaakt voor eliminatie van contentcomponenten op basis van het mediatype. In het XMLSchema wordt hierin een basisstructuur gehanteerd, die door een individuele actor kan worden gewijzigd. Indien de contenteliminatie niet wenselijk is, kan dit achterwege worden gelaten;
- De prioriteitstelling van het meest geschikte contentformaat van de mediatypes is naar keuze te veranderen. Op basis van testen in een live omgeving kan bepaald worden welk formaat de minste problemen geeft bij een specifiek mediatype. Op dit moment is een 'educated guess' gemaakt door voor een mobiel toestel bijv. te kiezen voor in aflopende prioriteit: 1. WML (specifiek formaat voor mobiele toestellen), 2. XHTML (een zeer compatibel formaat), 3. XML (zeer compatibel, maar moet aan de client zijde worden vertaald) en 4. HTML (minder compatibel).

3.5.3 Evaluatie op basis van raamwerk prestatie-indicatoren

Tabel 3.7 laat zien hoe de prestaties van het M-CMS passen binnen het raamwerk van prestatie-indicatoren. Uit de voorgaande paragrafen kwam al naar voren dat de leveringssnelheid zonder meer concurrerend is, los nog van het feit dat het M-CMS zich richt op het aanbieden van zo compleet en compatibel mogelijke contentformaten, teneinde ook de afbeeldingstijd te minimaliseren (client side speed-up).

Daarnaast biedt het M-CMS mogelijkheden om op basis van specifieke karakteristieken van de content fijnstelling te laten plaatsvinden. Zo ligt binnen de implementatie de nadruk in aflopende prioriteit op de bruikbaarheid van de content, de snelheid van de levering en tenslotte op de volledigheid van de content. Dit komt tot uiting in het feit dat niet-ondersteunde formaten niet worden meegegeven, hetgeen ten koste gaat van de volledigheid maar ten goede komt aan de bruikbaarheid en snelheid. Indien gewenst kan door beïnvloeding van de parameters deze prioriteitstelling worden aangepast.

Binnen het M-CMS heeft tekstuele content veruit de meeste aandacht gekregen. Deze is immers alom aanwezig, de scheiding tussen inhoud, structuur, layout en formaat is met XML en XSLT standaarden goed te realiseren en mede hierdoor is dit contenttype goed te manipuleren. Andere contenttypes zoals image of video worden alleen meegegeven wanneer het gevraagde formaat beschikbaar is. Bovendien wordt er geen vorm van contentconversie of organisatorische contentselectie toegepast teneinde kleine

apparaten bijv. een kleiner imagepakket aan te bieden. Het toepassen van mechanismen om ook deze contenttypes op maat te kunnen aanbieden is een apart chapter. Dit kan evenwel zeer nuttig zijn bij organisaties waar de nadruk ligt op andere dan tekstuele contenttypes.

Prestatie-indicatoren	Implementatie / evaluatie M-CMS prototype
1. Werking strategie	
a. Volledigheid content	De content wordt voor kleinere apparaten slechts ten dele geselecteerd, waardoor een vorm van contentsamenvatting plaatsvindt voor tekst. En niet ondersteunde formaten worden niet meegegeven om problemen met afbeelding te voorkomen (voor andere contenttypes dan tekst).
b. Bruikbaarheid	Alleen die contentformaten die het apparaat ondersteunt worden op basis van de MIME-types meegegeven, dus de content zou altijd bruikbaar moeten zijn.
2. Apparaatbep.	
a. Scherm-resolutie	Organisatorische selectie van contentcomponenten binnen IO op basis van het mediatype levert content 'op maat'.
b. Standaarden	Slechts die contenttypes die worden ondersteund worden meegegeven op basis van MIME types uit de HTTP request.
c. Bestands-formaten	Slechts die bestandsformaten die worden ondersteund worden meegegeven op basis van MIME types.
3. Leveringssnelheid	
a. Vertraging	Goede prestatie, zoals bleek uit paragraaf 3.5.1.
b. Versnellings-methoden	Inrichting pijpleiding is zo gekozen dat de content zo snel mogelijk wordt gereduceerd en berekeningen alleen dan worden uitgevoerd wanneer van toepassing. Caching toepassen: met name nuttig bij veel voorkomende (complexe) plaatjes die veel ruimte kosten (standaard op server en browser, niet binnen het prototype toegepast).
c. Client side speed-up	Door levering diverse formaten hoeft er geen conversie plaats te vinden aan de client side (in tegenstelling tot levering van XML formaat bijv., die bij de client wordt geconverteerd). Verwijdering van niet ondersteunde formaten geeft versnelde maar onvolledige afbeelding.
4. Ondersteuning	
a. Huidig	Geïntegreerde ondersteuningsmogelijkheden: contentcreatie maakt dynamische / geparametriseerde content en interactief gebruik mogelijk.
b. Toekomstig	Door ondersteuning van alle de facto internetstandaarden is te verwachten dat nieuw ontwikkelde standaarden compatibel zijn en daarmee eenvoudig ondersteund kunnen worden
5. Onderhoudbaarheid	Apparatedatabase voor user-agents binnen IO. Systeeminstellingen voor fijnstellen uit paragraaf 3.5.2.

Tabel 3.7: Evaluatie contentpijpleiding binnen raamwerk prestatie-indicatoren

4 Workflowmanagement

Contentmanagement systemen (CMS-en) die in eerste instantie worden aangeschaft voor ondersteuning bij het publiceren en beheren van websites, missen nogal eens een goede afstemming op de bedrijfsprocessen van de organisatie. Hoewel de meeste CMS-producten ondersteuning bieden voor workflowmanagement [vdWouden], zijn dit soms weinig flexibele oplossingen. Het multi-channel contentmanagement systeem (M-CMS) tracht een goed ondersteund en flexibel workflowmanagement te bieden voor het publicatieproces, dat op maat gemaakt kan worden voor een specifieke organisatie. Hiermee wordt invulling gegeven aan het volgende onderzoeksdoel:

Het inrichten van het workflowmanagement van het M-CMS zodat een goede ondersteuning voor het publicatie- en beheerproces wordt verkregen, met behoud van een flexibele opzet zodat deze binnen verschillende organisatievormen toepasbaar is.

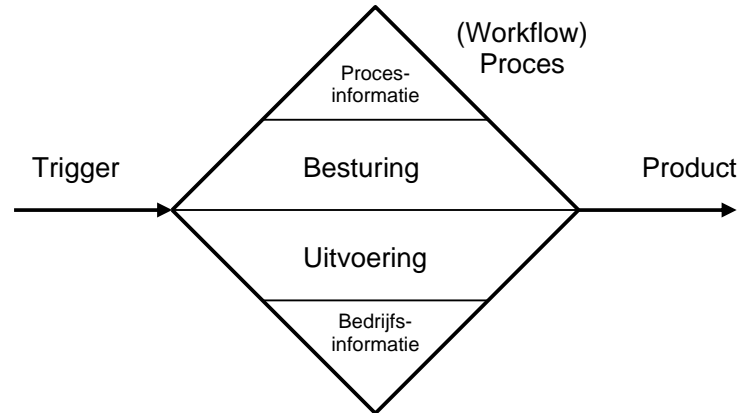
In dit hoofdstuk wordt aandacht geschonken aan de implementatie van het workflow-management binnen het M-CMS. Onder workflowmanagement wordt verstaan het managen van (bedrijfs)processen waarbinnen een scheiding is aangebracht tussen de uitvoering en de besturing van activiteiten [vdBerg]. Het workflowmanagement kan worden toegepast voor het ondersteunen van bedrijfsactiviteiten die binnen het publicatieproces en bij de beheeractiviteiten worden uitgevoerd.

In paragraaf 4.1 zal een plaatsbepaling van het workflowmanagement plaatsvinden op basis van enige theoretische achtergronden. In paragraaf 4.2 worden de verschillende onderdelen van het toegepaste workflowmanagement voor het publicatieproces beschreven. Hoewel de beheeractiviteiten niet als (workflow)proces beschouwd kunnen worden, kunnen de middelen van het workflowmanagement hiervoor toch ondersteuning bieden, hetgeen in paragraaf 4.3 wordt beschreven. Tenslotte wordt in paragraaf 4.4 een evaluatie gegeven van het workflowmanagement, zoals toegepast binnen het M-CMS.

4.1 Plaatsbepaling workflowmanagement

In een dynamische omgeving is de constante verandering van processen de enige zekerheid voor een organisatie. Het inspelen op veranderende omstandigheden vereist voor een organisatie een grote mate van flexibiliteit, welke gerealiseerd kan worden door onderscheid te maken tussen uitvoering en besturing van activiteiten. Hierbij is het van belang dat het management inzicht krijgt in de operationele beheersaspecten van de processen door procesinformatie te scheiden van bedrijfsinformatie. Het toepassen van workflowmanagement maakt dit mogelijk.

In figuur 4.1 is het workflow principe schematisch weergegeven. Een workflowproces heeft een duidelijk begin - een (start)trigger - dat ervoor zorgt dat een keten activiteiten uitgevoerd gaat worden. Een workflowproces heeft ook duidelijk eind, namelijk een eindproduct dat als resultaat van de activiteiten wordt opgeleverd. Elke keer dat een workflowproces wordt opgestart, wordt gesproken van een nieuwe instance.

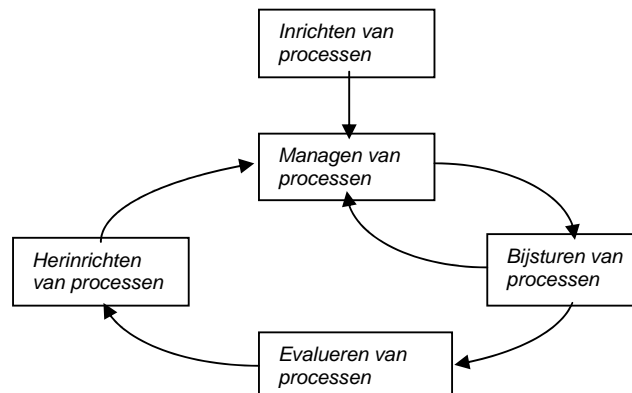


Figuur 4.1: Het workflow principe [vdBerg]

In de literatuur worden aanverwante concepten genoemd, die raakvlakken hebben met workflowmanagement. In het kader van het M-CMS wordt met name gedacht aan documentflow-management of in bredere zin objectflow-management. Dit concept staat voor het efficiënt en effectief routeren van een administratief object (een document of een digitale variant) door de organisatie. Objectflow-management verschilt van workflowmanagement in het ontbreken van registratie van gegevens over de voortgang en een adequate voortgangsbewaking. Omdat een uitgangspunt bij het M-CMS is dat het publicatieproces een tijdkritisch aspect heeft, zodat management en bijsturing van het proces gewenst is, zal hier verder van workflowmanagement gesproken worden.

Workflowmanagement kan beschouwd worden als een continue cyclus van het bijsturen en herinrichten van processen (de workflowlevenscyclus) [vdBerg] waarbij de volgende concepten van belang zijn (figuur 4.2):

- Inrichten van processen: dit wordt nader behandeld in paragraaf 4.1.1;
- Managen en bijsturen van processen in vorm van de besturingscyclus, die in paragraaf 4.1.2 wordt toegelicht;
- Evalueren en herinrichten van processen die samen met het managen en bijsturen de evaluatiecyclus vormen. Dit wordt in paragraaf 4.1.3 toegelicht.



Figuur 4.2: De cyclus van het continu bijsturen en herinrichten van processen

4.1.1 Procesinrichting

Het inrichten van een workflowproces of de workflowdefinitie betreft feitelijk het groeperen van activiteiten binnen een workflowproces. Activiteiten worden bepaald en gerelateerd aan de hand van het invullen van volgtijdelijkheid, voorwaarden voor het uitvoeren van de activiteiten en de criteria voor routebepaling in geval van beslissingen.

Binnen de procesinrichting wordt eenmalig een procesdefinitie gegenereerd, waarbij een invulling wordt gegeven aan:

- De procesgang: de procesactiviteiten, de volgtijdelijkheid en de eventuele afhankelijkheden;
- De actoren die een rol spelen binnen het proces met hun verantwoordelijkheden;
- De methode waarmee deze procesgang binnen het systeem wordt ondersteund. Dit is feitelijk de geautomatiseerde ondersteuning van het workflowproces;
- Het vaststellen van kenmerken van de procesactiviteiten die geregistreerd worden ten behoeve van het procesmanagement (zie tevens de volgende paragraaf).

Voor het modelleren van de workflow is een grote hoeveelheid aan modelleer technieken beschikbaar, waaronder flowcharts, activity diagrammen, structure charts, Petri Netten en dataflow diagrammen [Reijers]. De keuze voor de modellering dient te worden gemaakt op basis van het doel van het workflowmodel en de karakteristieken van de workflow.

4.1.2 Procesmanagement en procesbijsturing

Om een workflowproces te kunnen beheersen is er een integraal procesmanagement nodig. Dit heeft ten doel om suboptimalisatie te vermijden, de noodzakelijke coördinatie te laten plaatsvinden en te kunnen inspelen op een wisselende vraag naar producten en een variabele capaciteit van de resources. Indien noodzakelijk kan het workflowproces worden bijgestuurd, waardoor de besturingscyclus van figuur 4.2 wordt geëffectueerd (de wisselwerking tussen managen van processen en bijsturen van processen).

Ten behoeve van de besturingscyclus zijn de volgende aspecten van belang:

- Het definiëren van interne prestatie-indicatoren ten behoeve van de voortgangsbewaking;
- Het definiëren van normen voor de kwaliteitsbewaking van het proces;
- Het bieden van signaleringsmogelijkheden ter bewaking van de procesgang;
- Het bieden van escalatiemogelijkheden (bijsturing door afwijking van de normale procesgang) voor de workflowmanager teneinde het proces te kunnen bijsturen.

De prestatie-indicatoren, meetpunten en normen kunnen worden ontleend aan bepaalde kenmerken van de procesactiviteiten. Binnen de procesinrichting zal aandacht moeten worden besteed aan het registreren van deze kenmerken. De volgende groepen kenmerken komen daarvoor in aanmerking [vdBerg]:

- Tijd: bewerkingstijd, doorlooptijd en signaleringstermijn;

- Inzet van resources: verantwoordelijkheid, actoren, delegatie / vervanging, beslag op beperkte middelen, prestatie(meting);
- Voorwaardelijkheid van uitvoering: escalatiemogelijkheid, conditionaliteit, iteratie, facultatief / optioneel, redo-mogelijkheid;
- Typering van een activiteit ten behoeve van de herinrichting: als beslissing, controle, wachten, transport, documentatie, communicatie, bewerking. Op basis hiervan kan procesverbetering plaatsvinden (zie volgende paragraaf).

4.1.3 Procesevaluatie en procesherinrichting

Op basis van de procesinformatie kan men tot de conclusie komen dat het workflow-proces niet optimaal functioneert. Dit kan worden geconstateerd doordat er regelmatig moet worden bijgestuurd of dat de kwaliteit van de tussen- of eindproducten onvoldoende is. De proces- en kwaliteitsbewaking van het procesmanagement levert hiervoor de benodigde procesinformatie. In dit geval is er behoefte aan herinrichting van het workflowproces, hetgeen zal resulteren in een nieuwe workflow-procesdefinitie. Hiermee wordt een nieuwe herinrichtingscyclus opgestart, zoals in figuur 4.2 schematisch werd weergegeven.

Er is een aantal vormen van procesverbetering te onderkennen naar [vdBerg]:

- Aanpassingen aan de workflow-procesgang:
 - Eliminatie: het verwijderen van een workflow-procesactiviteit (ook inhoudelijk);
 - Integratie: de inhoud van een procesactiviteit wordt geïntegreerd in (een) andere procesactiviteit(en);
 - Verbreding: het opdelen van een procesactiviteit in meerdere activiteiten;
 - Parallellisatie: procesactiviteiten worden gelijktijdig uitgevoerd, waar deze voorheen sequentieel werden uitgevoerd;
- Volumevergroting: het volume heeft betrekking op het aantal instances dat door de workflow stroomt. De volgende manieren bewerkstelligen volumevergroting:
 - Vergroten van de bewerkingscapaciteit: inzetten van meer actoren;
 - Verminderen van de bewerkingstijd waardoor meer instances kunnen worden verwerkt. Dit kan door de uitvoering van de procesactiviteit te verbeteren of de bewerking te automatiseren;
- Effectiviteitsvergroting: de uitvoering van (een) workflow-procesactiviteit(en) wordt zo veranderd dat er een grotere toegevoegde waarde ten aanzien van het workflowresultaat ontstaat, bijvoorbeeld door een herverdeling van verantwoordelijkheden over minder organisatorische eenheden.

De methoden volgens welke de procesverbetering kan worden toegepast zijn legio. Deze variëren van specifieke workflowmethoden tot methoden voor Business Process Redesign / Reengineering (BPR) en worden per organisatie ook mogelijk anders gehanteerd.

4.2 Workflowmanagement van het publicatieproces

In deze paragraaf wordt allereerst aandacht geschonken aan de beargumentatie waarom het publicatieproces als een workflowproces kan worden beschouwd. Vervolgens zal op hoofdlijnen de realisatie van het workflowmanagement binnen het publicatieproces worden uiteengezet. Dit gebeurt op basis van de concepten uit de workflowlevenscyclus: de procesinrichting, het procesmanagement –en bijsturing en de procesevaluatie en -herinrichting.

4.2.1 Beargumentatie toepassing workflowmanagement

In paragraaf 4.1 werden de beginselen van het workflowmanagement uiteengezet. Om te beoordelen of een proces als een workflowproces beschouwd kan worden, zijn de volgende criteria bepalend [vdBerg]:

- Het gaat om een groot aantal instances dat het proces doorloopt;
- Er is behoefte aan voortgangsbewaking van de instances;
- Het proces heeft een hoge mate van complexiteit;
- Er is sprake een redelijke mate van standaardisatie;
- Er is onzekerheid vooraf met betrekking tot de bewerkings-en doorlooptijd.

Tabel 4.1 laat zien waarom deze criteria van toepassing zijn voor het publicatieproces zoals gehanteerd binnen het M-CMS. Hierbij worden de kenmerken van de doelorganisatie uit paragraaf 2.1 in herinnering geroepen, met name de content-intensiviteit van de organisatie met een hoog (al dan niet periodiek) content verloop waarbij sprake is van bepaalde randvoorwaarden (waaronder een deadline) bij de publicatie.

Criteria	Voorwaarde	Waarom van toepassing voor het publicatieproces?
Volume	Hoog	Contentintensieve doelorganisatie, met een hoog contentverloop (al dan niet periodiek)
Bewaking	Benodigd	Vanwege het tijdkritische aspect van publicatie is voortgangsbewaking en besturing noodzakelijk
Complexiteit	Complex	Het publicatieproces is op zich vrij triviaal, maar de mogelijke correctie-activiteiten die nodig zijn, maakt de levensloop van een instance weinig voorspelbaar
Standaardisatie	Redelijke mate	Het proces is vrij rudimentair en is ondanks beslispunten vrij standaard en generiek toepasbaar
Onzekerheid	Aanwezig	Content wordt bewerkt door verschillende actoren, waardoor bewerkings- en doorlooptijd onzeker is

Tabel 4.1: Workflowcriteria ingevuld voor het publicatieproces van het M-CMS

Tabel 4.1 laat zien dat het enige criterium waaraan in iets mindere mate wordt voldaan de complexiteit van het publicatieproces is; deze is niet zozeer

gelegen in de hoeveelheid activiteiten en / of actoren binnen het proces maar in het feit dat de potentiële herhaling(en) van activiteiten het proces onvoorspelbaar en weinig inzichtelijk maakt, indien een stuk content niet direct wordt goedgekeurd. Bovendien wordt het proces voor verschillende doelorganisaties gemodelleerd, hetgeen een belangrijke mate van flexibiliteit vergt binnen het proces.

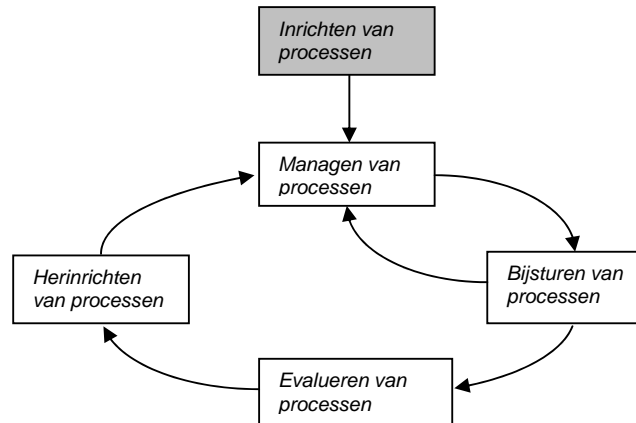
4.2.2 Procesinrichting publicatieproces

Wanneer het publicatieproces beschouwd wordt als een workflowproces kan deze worden gezien als een procesgerichte, formele workflow. De publicatieworkflow is procesgericht omdat het proces centraal staat en niet bepaalde taken binnen het proces. Bovendien is de publicatieworkflow een formele workflow, omdat de procesgang vooraf precies is te definiëren, al dan niet met toepassing van vaste of variabel terugkerende uitbreiding (tijdens het uitvoeren kunnen extra activiteiten verricht worden waarna - al dan niet - teruggekeerd wordt naar de normale procesgang).

[Reijers] maakt onderscheid tussen een taakgeoriënteerde benadering van het workflowproces, waarbij een workflow wordt beschouwd als een verzameling van gecorreleerde taken met procesinput en -output, en een taal / actie benadering die zich richt op conversaties en onderhandelingen tussen actoren. De taak-georiënteerde benadering is in het bijzonder geschikt voor gestructureerde workflows, terwijl de taal / actie benadering meer geschikt is voor ongestructureerde workflows. Aangezien het publicatieproces een procesgerichte, formele workflow is en daarmee een gestructureerde workflow, zal binnen het M-CMS een taakgeoriënteerde benadering worden gehanteerd.

In figuur 4.3 wordt de procesinrichting getoond binnen de workflowlevenscyclus. Binnen de procesinrichting wordt eenmalig een generieke procesdefinitie gegenereerd die voor verschillende organisaties toepasbaar moet zijn, waarbij een invulling wordt gegeven aan:

- De procesgang: de procesactiviteiten, de volgtijdelijkheid en de eventuele afhankelijkheden. De procesgang wordt in paragraaf 4.2.2.1 behandeld;
- De actoren die een rol spelen binnen het proces met hun verantwoordelijkheden, hetgeen in paragraaf 4.2.2.2 wordt toegelicht;
- De methode (de taakgeoriënteerde benadering) waarmee deze procesgang binnen het systeem wordt ondersteund, welke in paragraaf 4.2.2.3 wordt beschreven;
- Het vaststellen van kenmerken van de procesactiviteiten die geregistreerd worden ten behoeve van het procesmanagement, hetgeen binnen paragraaf 4.2.3 wordt behandeld.



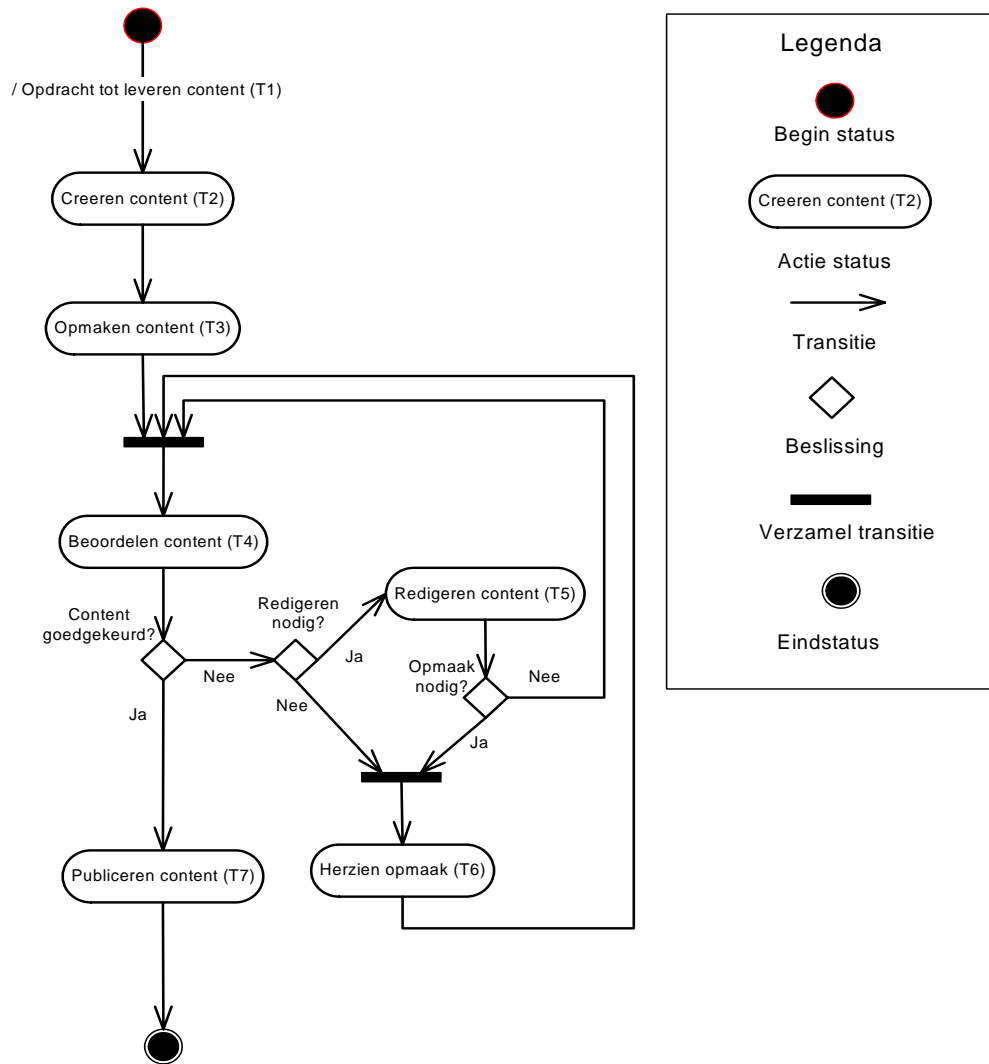
Figuur 4.3: De procesinrichting binnen de workflowlevenscyclus

4.2.2.1 De procesgang

De procesgang betreft het in beeld brengen van de publicatie-activiteiten, de volgtijdelijkheid en de eventuele afhankelijkheden. In paragraaf 2.1.1 werd bij de beschrijving van de doelorganisatie middels de DEMO modellering een beeld geschetst van het publicatieproces aan de hand van het procesdiagram van figuur 2.2. Nu werd in paragraaf 4.1.1 al vermeldt dat er veel modelleringsmogelijkheden zijn om de workflow in beeld te brengen. De keuze van de modellering is hierbij afhankelijk van het doel van het workflowmodel en karakteristieken van de workflow [Reijers]. Het publicatieproces is een betrekkelijk eenvoudige workflow, die ten doel heeft om voor een generieke doelorganisatie een flexibel workflowmanagement te definiëren. Hiertoe is het niet nodig om complexe modellering als de Petri Netten van [Reijers] te hanteren maar volstaat de UML modellering zoals deze binnen het RUP wordt gebruikt.

In het bijzonder kan het activity diagram worden gehanteerd, dat duidelijk een complexe workflow kan uitbeelden [Reed]. In figuur 4.4 wordt voor het publicatieproces de procesgang met de activiteiten, volgtijdelijkheid en afhankelijkheden in beeld worden gebracht. De actie statussen zijn hierin bijzondere statussen met een binnenkomende actie en een uitgaande transitie, die impliciet de afhandeling van de actie bevat.

Het activity diagram in figuur 4.4 beeldt op hoofdlijnen hetzelfde uit als het DEMO procesdiagram; ter referentie zijn de nummers van de DEMO transactietypes opgenomen in de action states. Hierbij zijn is een enkele wijziging aangebracht, namelijk het goedkeuren content is het beoordelen content geworden waarbij het redigeren content en herzien van opmaak na de beoordeling plaatsvinden (maar feitelijk wel binnen de goedkeuringsactiviteit).



Figuur 4.4: UML activity diagram van het publicatieproces

4.2.2.2 Actoren en verantwoordelijkheden

De actoren die een rol spelen binnen het proces met hun verantwoordelijkheden kunnen worden beschreven door aan de activiteiten een uitvoerende actor toe te kennen. Dit kan middels de organisatorische invullingstabel van DEMO uit tabel 2.1 van paragraaf 2.1.1. Op dit punt biedt het Rational Unified Process (RUP) evenwel de eerste hulpmiddelen om de actoren te beschrijven. Dit geschiedt op basis van de 'stakeholder en user requests' waarbij belanghebbenden en gebruikers worden geïdentificeerd, beschreven en hun interactie met het systeem wordt vastgelegd (zie ook bijlage C.1). Vanaf dit punt zal dan ook volledig op de RUP / UML modellering worden teruggevallen. In tabel 4.2 zijn de actoren opgenomen die binnen het publicatieproces een rol spelen, te weten de redacteur, vormgever en eindredacteur. Bovendien is hier de hoofdredacteur opgenomen, omdat deze als eindverantwoordelijke voor het publicatieproces wordt beschouwd. In tabel 4.2 zijn onderdelen die facultatief zijn tussen haakjes opgenomen.

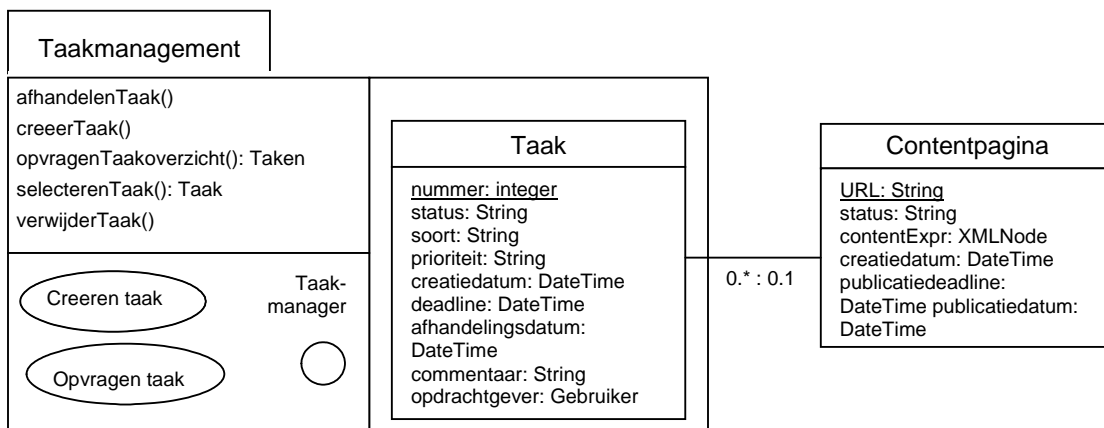
Actor	Beschrijving	Verantwoordelijkheden
Redacteur	Creeert en structureert content. Indien nodig wordt afgekeurde content herzien.	Content creëren Content structureren (middels XML) Content redigeren
Vormgever	Verzorgt en herziert de opmaak van de content.	Content opmaken: toepassen templates of handmatige opmaak
Eindredacteur	Beoordeelt content na creatie en opmaak en publiceert deze indien accoord.	Content beoordeling Content publicatie (Bewaking / bijsturing workflow)
Hoofdredacteur	Eindvoortwoordelijk voor publicatie workflow.	(Bewaking / bijsturing workflow) Beoordeling / herinrichting workflowproces

Tabel 4.2: Samenvatting van RUP user requests binnen het publicatieproces

4.2.2.3 Ondersteuningsmethode publicatieproces

Zoals hierboven reeds werd beschreven, wordt het workflowmanagement uitgevoerd op basis van een taakgeoriënteerd benadering omdat die voor gestructureerde workflow het meest geschikt is [Reijers]. Dit wordt in de vorm van taakmanagement gerealiseerd middels de volgende aanpak: de activiteiten, zoals die worden uitgevoerd binnen het publicatieproces, worden als taak behandeld. Een taak is een werkopdracht aan een individuele actor of actorgroep, die voor uitvoering in behandeling wordt gegeven. Om dit te realiseren is binnen het M-CMS het package taakmanagement ontwikkeld, zoals afgebeeld in figuur 4.5. Een package is een container voor enig deel van het ontwikkelwerk, dat als een eenheid beschouwd dient te worden [d'Souza].

Binnen de modellering van het M-CMS worden componenten weergegeven in de vorm van packages, naar [Arlow]. In figuur 4.5 zijn in het kader linksboven de functies (operaties die een resultaat teruggeven) en procedures (operaties die geen resultaat teruggeven) afgebeeld die binnen het taakmanagement kunnen worden uitgevoerd. Hierbij zijn de parameters die aan de operaties worden meegegeven niet opgenomen in de figuur. In het kader linksonder staan de use cases die door de package worden gerealiseerd en de interfaces.



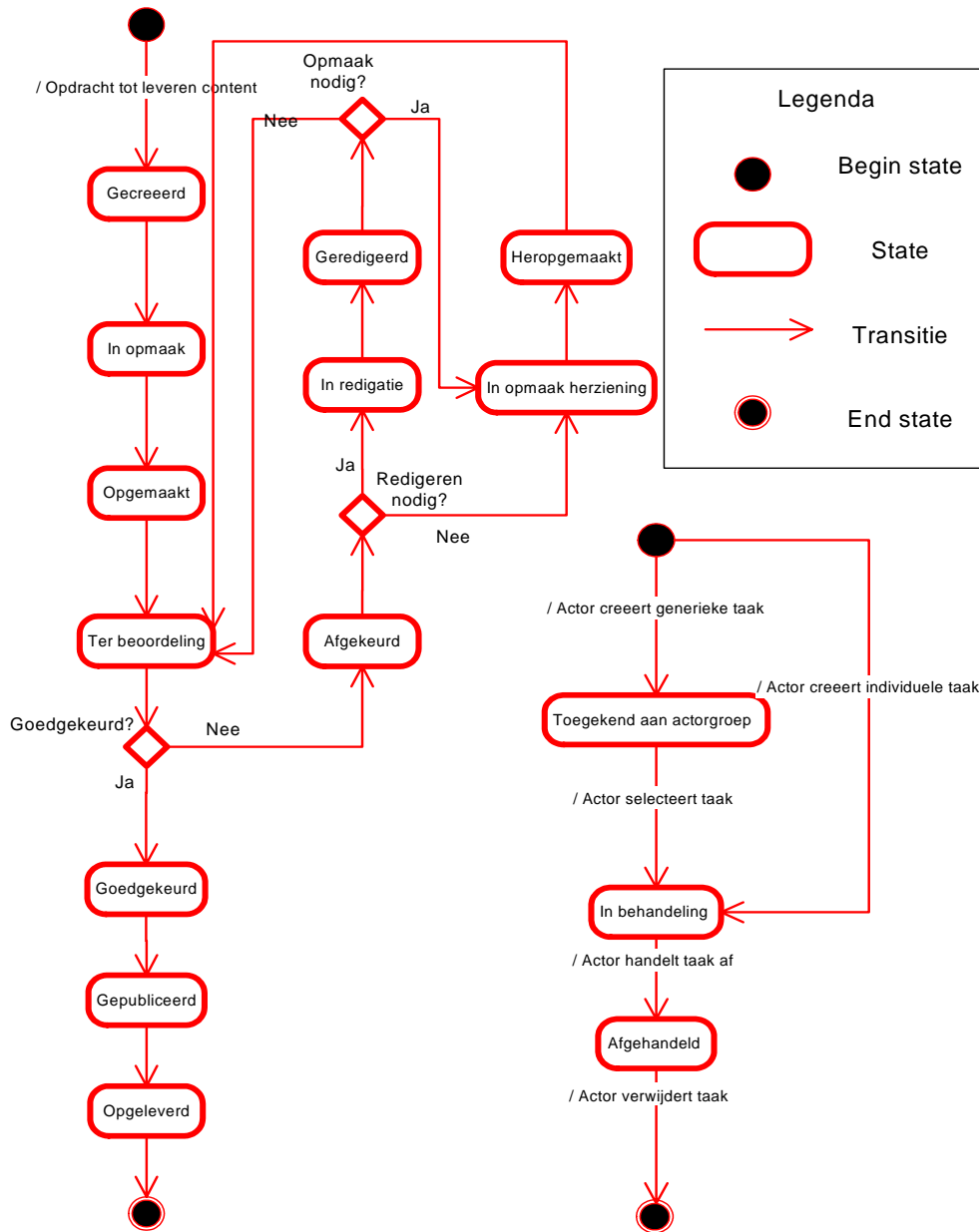
Figuur 4.5: Package taakmanager en relatie van de Taak met Contentpagina

In het rechter kader van figuur 4.5 staan de relevante klasse(n), waarvan de attributen zijn opgenomen. Hierbij is het taaknummer de unieke identifier van de taak. Het attribuut taaksoort is een aanduiding voor de activiteit die uitgevoerd moet worden, zoals “opmaken content” of “redigeren content”. De status van de taak (die bijv. kan zijn in behandeling of afgehandeld) wordt in figuur 4.6b nader toegelicht middels het statusdiagram van UML. De package is illustratief en is niet compleet in functionaliteit en functiedefinitie. In bijlage C zijn de complete diagrammen van de verschillende packages in het implementatiemodel opgenomen.

Op een taak kunnen de volgende handelingen worden toegepast (figuur 4.5):

- Creeer taak: het aanmaken van een taak om deze toe te wijzen aan een individuele actor of aan een actorgroep;
- Selecteer taak: het in behandeling nemen van een openstaande taak;
- Afhandelen taak: indien de betreffende activiteit is uitgevoerd kan de taak worden afgehandeld. De volgende activiteit (taak) in het proces kan hierbij worden aangemaakt, door de behandelend actor of automatisch door het systeem;
- Verwijderen taak: een onderhanden taak kan worden verwijderd indien een opschoning van de database moet plaatsvinden;
- Opvragen taakoverzicht: voor individuele actoren of voor actorgroepen kan een taakoverzicht opgevraagd worden van taken met de status ‘in behandeling door’ respectievelijk ‘toegekend aan actorgroep’. Zo kan een actor direct de voorraad onderhanden werk bekijken voor alle actorgroepen waarvoor hij gerechtigd is of die individueel aan hem zijn toegewezen.

Aangezien binnen het publicatieproces alle activiteiten betrekking hebben op het vervaardigen van een stuk content (hierna aangeduid als klasse Contentpagina), kunnen de statussen van de contentpagina als uitgangspunt dienen voor de procesinrichting naast het activity diagram van figuur 4.4. De toestanden van de contentpagina worden middels het UML statusdiagram weergegeven in figuur 4.6a. Ter verheldering zijn bij enkele transities de actie vermeld, die de transitie bepaald.

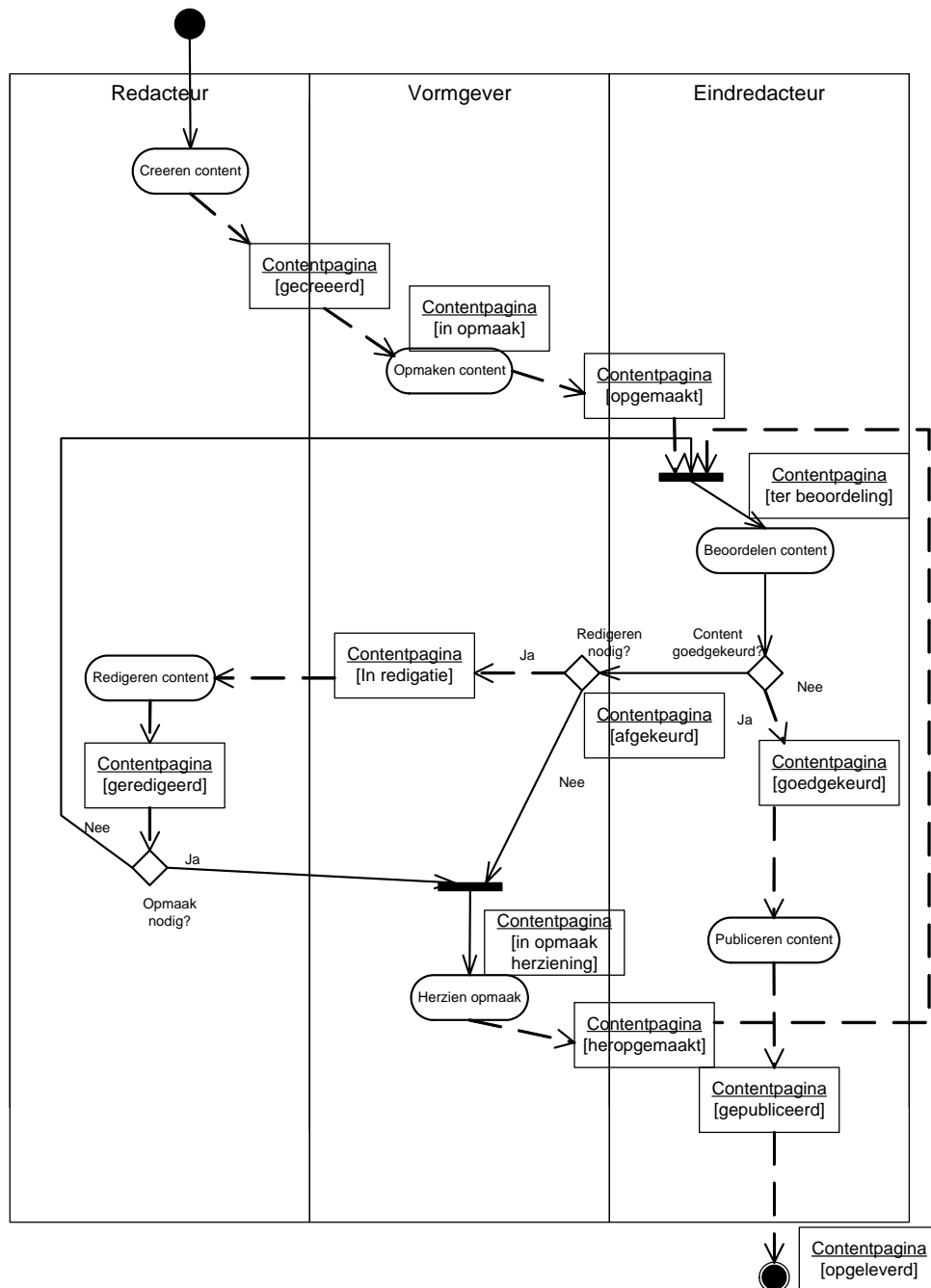


4.6a: Statusdiagram klasse Contentpagina

4.6b: Statusdiagram klasse Taak

Figuur 4.6: UML status diagrammen van de klassen Contentpagina en Taak

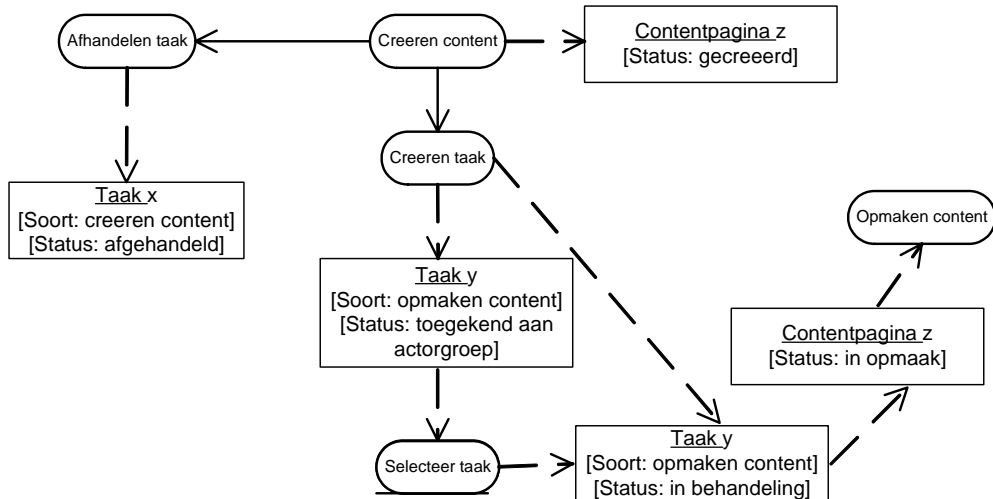
Voor het vastleggen van de procesgang kunnen de statuses van de contentpagina in het activity diagram worden geïntegreerd [UML] zoals in figuur 4.7 is weergegeven. Hierbij is tevens gebruik gemaakt van 'swinlanes', die de verantwoordelijkheid van de actiestatus bij een actor legt. De status van de contentpagina verandert bij het afronden van een actiestatus (bijv gecreeerd) of bij het aanvangen ervan (bijv in opmaak). Het gaat hierbij steeds om dezelfde contentpagina.



Figuur 4.7: Activity diagram met objectstatus Contentpagina

In figuur 4.7 zijn tevens gestreepte lijnen (de objectflow) opgenomen die gebruikt worden voor de statuswijzigingen van het object contentpagina. Het proces zoals beschreven is niet geheel volledig: per transitie tussen actiestatussen treedt het taakmanagement in werking, zoals in figuur 4.8 is uitgebeeld voor het taakmanagement dat plaatsvindt tussen het creëren van de content en het opmaken van de content. Zodra het creëren van de content is afgerond, kan de taak 'creëren content' (x) worden afgehandeld. Tevens wordt

een nieuwe taak (y) gecreëerd ten behoeve van het opmaken van de content, die wordt toegekend aan de actorgroep vormgevers of aan een individuele vormgever. In dit laatste geval (ook te bereiken na het selecteren van de taak door een vormgever) is de taakstatus 'in behandeling' geworden en de contentpagina is in opmaak. Dit gehele traject is de werkelijke route tussen 'creëren content' en 'opmaken content' zoals in figuur 4.7 gegeven en geldt ook voor de andere transities tussen actiestatussen.



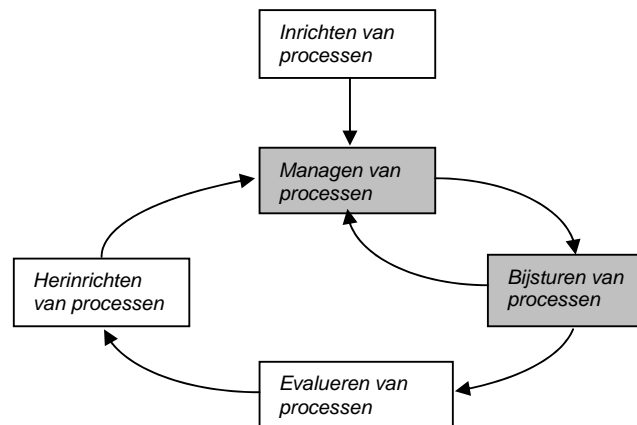
Figuur 4.8: Deel activity diagram voor taakmanagement

Bij het taakmanagement is een aantal randvoorwaarden en uitgangspunten van toepassing:

- Een taak kan enerzijds aan een individuele actor worden toegewezen, waarna de status van de taak is 'in behandeling'. Anderzijds kan een taak worden toegewezen aan een actorgroep. In dit geval zal een individuele, geautoriseerde actor de taak moeten selecteren om in behandeling te zijn genomen (figuur 4.8);
- Een redacteur van een contentpagina blijft verantwoordelijk voor zijn content en zal indien nodig later in het proces de content redigeren;
- Een vormgever die een contentpagina opmaakt is niet gebonden aan de content: het herzien van de opmaak kan door een andere vormgever geschieden;
- Het werken van meerdere redacteurs aan eenzelfde contentpagina vindt niet binnen het systeem plaats;
- Het vaststellen van kenmerken van de procesactiviteiten die geregistreerd worden zal plaatsvinden bij het procesmanagement in paragraaf 4.2.3.

4.2.3 Procesmanagement en -bijsturing publicatieproces

Na het bepalen en inrichten van het workflowproces, kan het workflowproces ten uitvoer worden gebracht en kan het workflowmanagement plaatsvinden. In figuur 4.9 zijn het procesmanagement en -bijsturing getoond binnen de workflowlevenscyclus.



Figuur 4.9: Procesmanagement en -bijsturing binnen de workflowlevenscyclus

Er is een aantal voorwaarden om workflowmanagement succesvol te laten zijn [vdBerg]:

- Inzicht in resources en werk: er is een actueel inzicht in beschikbare resources (met name beperkte middelen) en in de actuele werkvoorraad nodig;
- Self-management door de medewerkers: het plaatsen van verantwoordelijkheden op een lager niveau in de organisatie (hiermee wordt meer draagvlak gecreëerd voor het workflowmanagement);
- Attituden van betrokkenen: indien medewerkers zelf initiatief ontplooiën om de procesdoelstellingen te bereiken is er een attitude mogelijk die zich verlegt van reactief reageren naar proactief sturen;
- Vergroting van de flexibiliteit van de processen: voor continue aanpassingen van het proces is een grote flexibiliteit vereist, welke kan worden verkregen door een brede inzetbaarheid van mensen, een separate besturing van activiteiten en een modulaire opgezette informatievoorziening;
- Managementinformatie in de vorm van procesgegevens, te weten informatie over het werkaanbod, procesdefinitie, capaciteit (autorisaties en resources) en geleverde prestaties (doorlooptijden en geofferde capaciteit).

In tabel 4.4 zijn deze voorwaarden ingevuld voor de implementatie binnen het M-CMS.

Voorwaarde	Invulling M-CMS
Inzicht in resources en werk	<ul style="list-style-type: none"> • Taakoverzichten per individuele actor of voor actorgroepen geeft een overzicht van de actuele werkvoorraad, zowel voor uitvoerenden als voor het management; • De voornaamste resource binnen het publicatieproces zijn de medewerkers; een manager kan op basis van afwezigheid, ziekte en voorhanden werk zien of er voldoende medewerkers beschikbaar zijn voor de uitvoering.
Self-management	<ul style="list-style-type: none"> • Actoren kunnen na het uitvoeren van een taak deze zelf toekennen aan een individuele actor of actorgroep. Hiermee wordt op een lager organisatieniveau een stukje routing binnen het workflowproces geregeld; • Beperkingen aan de taaktoewijzing kan binnen het M-CMS op managementniveau worden vastgelegd middels systeeminstellingen. Deze restricties bepalen feitelijk de procesgang van het workflowproces.
Attitude van betrokkenen	<ul style="list-style-type: none"> • Het aanbrengen van flexibiliteit bij de toewijzing van een taak aan de volgende actor geeft al een mogelijkheid tot proactieve sturing. Hiermee kan bijvoorbeeld bestaande werkspecialisatie nuttig worden gebuikt; • Het ondersteunen van opnieuw toewijzen van taken maakt een milde vorm van bijsturing mogelijk. Bijvoorbeeld: een medewerker die op vakantie gaat kan zijn individuele taken aan de actorgroep toekennen. Op hoger organisatieniveau kan een manager (hoofdredacteur) bijvoorbeeld het toewijzen van taken aan deze medewerker blokkeren.
Vergroting flexibiliteit	<ul style="list-style-type: none"> • Separate besturing van activiteiten wordt door invulling van het procesmanagement en –bijsturing ondersteund door het M-CMS (zie verder in dit hoofdstuk); • Concrete invulling van de inzetbaarheid van medewerkers is sterk organisatie afhankelijk;
Management / procesinformatie	<ul style="list-style-type: none"> • Procesinformatie wordt middels parameters, meetpunten en prestatie-indicatoren betreffende de taken gegenereerd.

Tabel 4.4: Voorwaarden succesvol workflowmanagement binnen het M-CMS

Ten behoeve van het procesmanagement en de procesbijsturing binnen de besturingscyclus zijn de volgende aspecten van belang:

- Het definiëren van interne prestatie-indicatoren ten behoeve van de voortgangsbewaking: deze worden in paragraaf 4.2.3.1 behandeld;
- Het definiëren van normen voor de kwaliteitsbewaking van het proces: het bewaken van de kwaliteit is deels geïntegreerd in het publicatieproces door de beoordeling van de content door de eindredacteur. Kwaliteitsnormen voor content zijn verder moeilijk te

definiëren in generieke zin, vanwege de sterke afhankelijkheid van de specifieke organisatie waar het systeem gebruikt gaat worden en het karakter van de content die er gepubliceerd wordt. Er zal hier derhalve geen specifieke aandacht aan kwaliteitsnormen worden besteed;

- Het bieden van signaleringsmogelijkheden ter bewaking van de procesgang: deze worden in paragraaf 4.2.3.2 behandeld;
- Het bieden van escalatiemogelijkheden voor de workflowmanager teneinde het proces te kunnen bijsturen: deze worden in paragraaf 4.2.3.3 behandeld.

De invulling van het M-CMS voor het procesmanagement en de procesbijsturing is in tabel 4.4 globaal weergegeven op basis van de succesvoorwaarden. In de volgende paragrafen zal concreter worden ingegaan op de ondersteuningsmiddelen die beschikbaar kunnen worden gesteld voor het procesmanagement.

4.2.3.1 Voortgangsbewaking op basis van prestatie-indicatoren

In paragraaf 4.1.2 werden de kenmerken van een workflow-procesactiviteit genoemd. Deze kenmerken worden bij de procesinrichting nader ingevuld zodat een aantal interne prestatie-indicatoren voor het proces kunnen worden gegenereerd. Binnen dit project wordt evenwel van een generieke doelorganisatie uitgegaan, waardoor concrete invulling van deze meetpunten niet altijd mogelijk is. Binnen het publicatieproces kan in ieder geval worden aangegeven welke parameters van belang zijn en op welke wijze procesmanagement en bijsturing kan plaatsvinden.

De meetpunten die voor de voortgangsbewaking worden gehanteerd houden met name verband met de tijdskenmerken. Hierbij moet onderscheid gemaakt worden tussen de bewerkingstijd (de uitvoeringstijd van de activiteit) en de doorlooptijd (de tijd tussen het in behandeling nemen en het vrijgeven) binnen het publicatieproces:

- De *bewerkingstijd* is op basis van ervaringsgegevens binnen een organisatie redelijk nauwkeurig vast te stellen. Wel zal de bewerkingstijd afhankelijk zijn van bijvoorbeeld de omvang en complexiteit van een contentpagina (denk aan creatie en opmaak). Mogelijk kan de bewerkingstijd als afhankelijke variabele gedefinieerd worden, maar dit is organisatie afhankelijk;
- De *doorlooptijd* zou in het ideale geval de bewerkingstijd moeten benaderen. Indien de actoren de meest schaarse resource zijn (en daarmee de hoeveelheid onderhanden / af te handelen werk groot is), dan zullen de doorlooptijd en bewerkingstijd niet dicht bij elkaar liggen. Dit is voor het publicatieproces de verwachting, zeker in geval van publicatie op langere termijn (het uiteindelijk halen van een deadline is belangrijk). Indien de 'time to publication' een belangrijke factor is, zullen deze tijden dicht bij elkaar moeten liggen.

Naast tijdgerelateerde meetpunten, kunnen meetpunten op basis van de inzet van resources en de workload worden vastgelegd. Hierbij wordt gedacht aan het aantal taken of de leeftijd van de taken die een actorgroep of een individuele actor in zijn taaklijst heeft staan. Deze meetpunten kunnen een indicatie geven voor een te hoge workload en / of een gebrekkige inzet van resources.

Op basis van deze parameters kan aan het eind en tijdens de uitvoering van de activiteiten de prestatie van de workflow worden gemeten. Dit geeft gelegenheid om tijdens de uitvoering van het workflowproces pro-actief bij te sturen. Naast de bewerkingstijd en de doorlooptijd kan binnen een activiteit een signaleringstermijn worden gehanteerd. Dit wordt in de volgende paragraaf behandeld onder de signaleringsmogelijkheden voor de procesbewaking.

In tabel 4.5 wordt ter illustratie voor twee verschillende organisatievormen - te weten een webmagazine en een nieuwssite die in de inleiding als voorbeelden van doelorganisaties werden aangehaald - de mogelijke invulling binnen het M-CMS van de aspecten ten aanzien van procesmanagement en –besturing gegeven. Hierbij zijn tevens de signaleringstermijn en escalatiemogelijkheden ingevuld; deze onderwerpen komen in de volgende paragrafen aan de orde.

Aspect	Webmagazine	Nieuwssite
Bewerkingstijd	Mogelijke grote variatie in contentomvang, dus afhankelijk van omvang en complexiteit	Artikelen vergelijkbare omvang, dus bewerkingstijd vrij nauwkeurig vast te stellen
Doorlooptijd	Halen van deadline is belangrijk, dus doorlooptijd van totale workflow is belangrijker dan van separate activiteiten	Time to publication belangrijk, dus doorlooptijd moet bewerkingstijd benaderen
Signalerings- termijn	Afstemmen op deadline, relatief of absoluut signaleren en middels voorschriften richting geven	Richten op snelle publicatie, dus absolute verschil tussen tijden moet beheerst worden
Prestatiemeting	Op basis van resterende tijd tot de deadline, mogelijk in verhouding tot contentomvang en 'leeftijd' van de instance	Op basis van de 'leeftijd' van de instance, omdat een snelle publicatie belangrijk is
Escalatie- mogelijkheden	Opnieuw toewijzen, samennemen of 'claimen' van (sterk) vertraagde taken	Opnieuw toewijzen, samennemen of 'claimen' van taken die de publicatie van content ophouden

Tabel 4.5: Vergelijking procesmanagement tussen webmagazine en nieuwssite

4.2.3.2 Signaleringsmogelijkheden ter bewaking procesgang

Op basis van prestatie-indicatoren zoals de bewerkingstijd en doorlooptijd kunnen signaleringen worden ingebouwd ter bewaking van de procesgang. Dit kan bijvoorbeeld in de vorm van een signaleringstermijn, die is gedefinieerd als de termijn die vastgelegd wordt op een vast moment voor het eind van de activiteit [vdBerg]. Een voorbeeld is dat twee dagen voor de beoogde afronding van de activiteit een signalering wordt gegeven aan de uitvoerende. De bovenstaande definitie is echter vrij nauw. De signalering kan uitgebreider worden toegepast, waarbij de volgende aspecten zijn te onderscheiden:

- Organisatieniveau: een signalering kan plaatsvinden op operationeel niveau (bij de redacteur of vormgever) of op management niveau (eind/hoofdredacteur). Het laten plaatsvinden van signaleringen op

- operationeel niveau kan de betrokkenheid van de medewerkers bij de procesgang verbeteren, zolang de signalering niet als hinderlijk wordt ervaren (in het uiterste geval een 'big-brother' effect);
- Signaleringsaard: de aard van het signaleren kan een aantal vormen aannemen;
 - Informatief, bijvoorbeeld in de vorm van een herinnering;
 - Voorschrijvend, bijvoorbeeld in de vorm van een aanbeveling;
 - Dwingend, bijvoorbeeld in de vorm van een alarmering, voor een activiteit die onmiddellijke actie vereist.
 - Absolute (vaste) versus variabele (geparametriseerde) signalering: de signaleringstermijn kan vastgelegd zijn voor alle instances of variabel per instance worden ingesteld. Voorbeelden:
 - Absolute signalering op basis van een deadline, bijvoorbeeld: 48 uur voor de deadline moet de content gecreëerd zijn;
 - Relatieve signalering op basis van een deadline (na het opstarten van een instance). Bijvoorbeeld indien de deadline over 4 dagen is, en er 50% van de tijd voor een contentcreatie wordt aangehouden, dan moet deze binnen 2 dagen worden opgeleverd;
 - Absolute signalering op basis van snelle publicatie: richtlijnen als dat een uur na indienen van de content een pagina van opmaak moet zijn voorzien.
 - Het signaleringsbereik: signalering kan plaatsvinden (op basis van parameters) binnen een specifieke activiteit, maar ook ten aanzien van het volledige proces. Dit laatste heeft evenwel de nodige raakvlakken met de procesevaluatie die in paragraaf 4.2.4 wordt beschreven.

De bovenstaande signalering heeft betrekking op tijdsgerelateerd prestatie-indicatoren. Daarnaast kan signalering op basis van de inzet van resources en het meten van de workload plaatsvinden:

- Het aantal taken of de leeftijd van de taken die een actorgroep in zijn taaklijst heeft staan, kunnen een indicatie zijn van een te hoge workload en / of een gebrekkige inzet van resources;
- Dezelfde indicatoren gelden voor individuele actoren, waarbij een specifieke actor een onevenredig hoge workload heeft, bijvoorbeeld door verhoogde toewijzing van taken door zijn expertise.

De acties die moeten worden ondernomen om de inzet van resource te herzien of de workload aan te passen liggen echter al dicht aan tegen het herinrichten van processen. Een voorbeeld is dat individuele toewijzing aan vormgevers kan worden geblokkeerd omdat hun individuele workload sterk varieert. In plaats daarvan bepalen de vormgevers zelf binnen hun actorgroep of taken individueel worden toegewezen, waarmee feitelijk een verandering in de procesinrichting plaatsvindt.

4.2.3.3 Escalatiemogelijkheden voor procesbijsturing

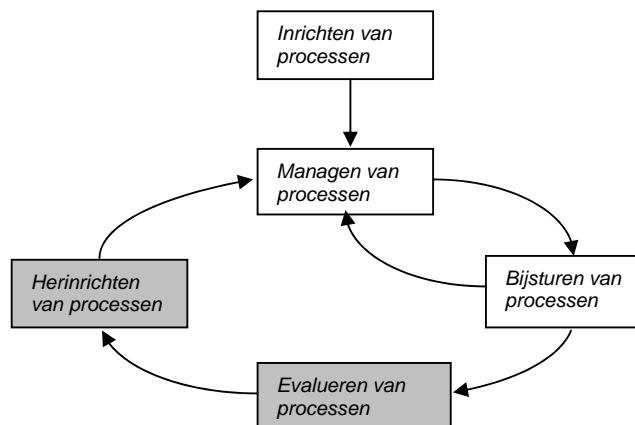
De signaleringen van de vorige paragraaf maken het mogelijk om enerzijds (potentiële) knelpunten of problemen in het proces te detecteren en anderzijds om binnen de grenzen van het workflowproces actie te ondernemen in de vorm van procesmanagement. Indien de te ondernemen actie betekent dat er bijsturing nodig is buiten de normale procesgang om, dan spreekt men van escalatie.

De escalatiemogelijkheden binnen het M-CMS zijn voorbehouden aan het management (door de eindredacteur of hoofdredacteur) en bestaan met name uit de flexibele omgang met taken:

- Taken opnieuw kunnen toewijzen aan andere actoren (op management niveau dus het opnieuw toewijzen van een aan een individu toegekende taak);
- Activiteiten in de procesgang kunnen worden samen genomen. Bijvoorbeeld kan een redacteur die ook kennis heeft van vormgeving bij het redigeren ook de opmaak verzorgen (voor zover herziening benodigd is) zodat de content direct kan worden beoordeelt en gepubliceerd. Het is binnen het M-CMS in beginsel niet vastgelegd dat de procesgang tot op de letter gevolgd moet worden;
- Activiteiten kunnen worden 'geclaimd' door een actor die daartoe geautoriseerd is (in ieder geval de manager). Een eindredacteur kan bijvoorbeeld het redigeren van content zelf verzorgen als de verantwoordelijke redacteur afwezig is of de aanpassingen marginaal zijn (in dit laatste geval vindt aanpassing evenwel niet plaats als gevolg van escalatie), waarbij hij de activiteit redigeren 'claimt' van de verantwoordelijke redacteur.

4.2.4 Procesevaluatie en procesherinrichting publicatieproces

Op basis van de procesinformatie die aan het procesmanagement wordt ontleend kan men tot de conclusie komen dat het workflowproces niet optimaal functioneert. Dit kan worden geconstateerd doordat er regelmatig moet worden bijgestuurd, dat normen niet worden gehaald of dat de kwaliteit van de tussen- of eindproducten onvoldoende is. Naast de bestaande procesinformatie zijn tevens externe prestatie-indicatoren benodigd die informatie bevatten met betrekking tot prestaties van de gehele workflow. De combinatie van deze gegevens maakt een effectieve procesevaluatie mogelijk. Bovendien moeten er ondersteuningsmiddelen bestaan voor het herinrichten van de workflow. Figuur 4.10 laat de procesevaluatie en –herinrichting zien binnen de workflowlevenscyclus.



Figuur 4.10: Procesevaluatie en –herinrichting binnen de workflowlevenscyclus

4.2.4.1 Prestatie-indicatoren ten behoeve van de procesevaluatie

In de voorgaande paragrafen is geschetst hoe op basis van de prestatie-indicatoren en de resulterende prestatiemeting van de workflowactiviteiten het procesmanagement en –bijsturing kan worden ingevuld. Deze procesinformatie of afgeleide parameters kunnen, al dan niet in aangepaste vorm, ook voor de procesevaluatie en de herinrichting worden gebruikt. Hierbij zal vooral gekeken worden naar de workflow in zijn geheel en kunnen de volgende externe prestatie-indicatoren een goed hulpmiddel zijn voor de procesevaluatie:

- Tijdtechnische indicatoren: dit kunnen de mate van het behalen deadlines of de ‘time to publication’ zijn;
- Het schetsen van een beeld van de tijd per activiteit binnen de workflow (zowel bewerkingstijd als doorlooptijd) en mogelijk op basis hiervan knelpunten bloot leggen indien bepaalde prestaties niet gehaald worden;
- Het niet behalen van bepaalde kwaliteitsnormen: kwaliteitsaspecten zijn weliswaar subjectief, maar het niet behalen van normen kan gelegen zijn in een niet optimale inrichting van de workflow. Indien er veel fouten in de opmaak van de content worden gemaakt als gevolg van een overhaaste uitvoering van de opmaakactiviteit (bewerkingstijd te laag ingeschat), dan kan de herinrichting (van onderdelen) van het proces noodzakelijk zijn.

In tabel 4.6 zijn nogmaals het webmagazine en de nieuwssite organisaties naast elkaar gezet, nu voor een vergelijking van de aspecten van de procesevaluatie.

Indicator	Webmagazine	Nieuwssite
Workflowtijd	De mate van het behalen van deadlines en de marge waarmee deze gehaald worden	De tijdsduur van instances: ‘time to publication’
Verdeling activiteiten	Knelpuntanalyse indien deadlines niet gehaald worden	Knelpuntanalyse indien ‘time to publication’ te hoog is
Kwaliteitsnormen	Kwaliteit van het magazine onvoldoende vanwege bijv. inschatting bewerkingstijd redigeren te laag	Onvoldoende kwaliteit van artikelen door hoge publicatiedruk

Tabel 4.6: Vergelijking procesevaluatie tussen webmagazine en nieuwssite

4.2.4.2 Ondersteuning voor de procesherinrichting

Op basis van de procesevaluatie kan besloten worden dat het workflowproces herinrichting behoeft. De diversiteit aan methoden voor procesverbetering en procesherinrichting is groot en het gaat te ver om deze hier te behandelen. Het is evenwel van belang dat de herinrichting op een algemene wijze ondersteund wordt binnen het systeem. Als de vormen van procesverbetering uit paragraaf 4.1.3 in herinnering worden geroepen, kan enerzijds gekeken worden naar de ondersteuning voor aanpassingen aan de procesgang. Anderzijds kunnen de volumevergroting en effectiviteitsvergroting worden beschouwd als middelen voor procesverbetering.

Aanpassingen aan de workflow-procesgang bestaan uit eliminatie, integratie, verbreding of parallelisatie van procesactiviteiten. De volgende aspecten van het M-CMS ondersteunen het aanpassen van de procesgang van de workflow:

- Het workflowproces is gerealiseerd door middel van het taakmanagement, waarbij de procesgang niet in steen is gebeiteld. Dit wil zeggen dat de procesgang zonder veel moeite op operationeel niveau kan worden (bij)gestuurd;
- Herinrichting van het proces zal worden geïmplementeerd op het niveau van systeeminstellingen en autorisaties, die de randvoorwaarden van de procesgang vorm geven. Het veranderen van de procesgang wordt feitelijk door de restricties van de taaktoekenning bepaald. Twee voorbeelden:
 - De groep redacteurs is geautoriseerd voor het toekennen van taken aan vormgevers (activiteit opmaken content) en aan de eindredacteur (activiteit beoordelen content, na redigeren). Indien binnen de procesgang een contentpagina altijd opnieuw moet worden opgemaakt door een vormgever, zou een redacteur na het afronden van een taak redigeren geen bevoegdheid mogen hebben tot het aanmaken van een taak beoordelen content, maar slechts tot een taak voor het opmaken van content. Op deze manier kan eliminatie van een procespad plaatsvinden;
 - Indien een nieuwe actor zorg gaat dragen voor de publicatie van de content, dan kan een nieuwe actorgroep (publicist) worden gecreëerd alsmede nieuwe individuele actoren die dezelfde autorisaties hebben als de groep. De autorisatie voor de eindredacteur om een contentpagina te publiceren wordt dan verwijderd en de mogelijkheid om taken toe te kennen aan de publicisten wordt toegevoegd.

De maatregelen die kunnen worden genomen voor de volume- en effectiviteitsvergroting zijn het vergroten van de bewerkingscapaciteit, het verminderen van de bewerkingstijd en het veranderen van de uitvoering van een workflowprocesactiviteit. Dit zijn maatregelen die in een live-omgeving worden toegepast en die te afhankelijk zijn van de organisatie waar het M-CMS wordt gebruikt om op voorhand ondersteuningsmiddelen voor aan te bieden. Maatregelen van deze aard gaan het bereik van de workflowondersteuning te buiten, alsmede de onderzoeksdoelstellingen binnen het afstudeerproject.

4.3 Ondersteuning voor beheeractiviteiten

In de voorgaande paragraaf is voor het publicatieproces expliciet invulling gegeven aan een aantal aspecten van het workflowmanagement en aanverwante concepten zoals het (her)inrichten en bijsturen van de workflow. Hoewel de beheeractiviteiten een aantal min of meer onafhankelijke activiteiten betreft, kunnen bepaalde hulpmiddelen van de workflowondersteuning ook toegepast worden op deze afzonderlijke activiteiten.

De beheeractiviteiten betreffen alle activiteiten die na de contentpublicatie gehanteerd worden om de content te onderhouden en archiveren (site- en contentmanagement) en het beheer te verzorgen van het systeem en templates (systeembeheer respectievelijk templatebeheer). De volgende beheeractiviteiten zijn geïntegreerd in het M-CMS:

- Contentbeheer: het beheren van de content in zijn elementaire vorm van contentpagina's door de actor contentbeheerder. Hierbij worden

- activiteiten uitgevoerd als het aanpassen van live content en structuur, archivering en verwijdering van content;
- Sitebeheer: beheeractiviteiten voor zover ze betrekking hebben op een samenhangend geheel van contentpagina's (site), uitgevoerd door de actor contentbeheerder. Het maken van back ups en het uitvoeren van een roll back van een volledige site zijn hiervan voorbeelden;
- Systeembeheer: hieronder wordt verstaan het beheren van systeeminstellingen (voor het M-CMS in het algemeen en van de contentengine in het bijzonder), alsmede het gebruikers- en autorisatiebeheer van actorgroepen en actoren. Een en ander wordt uitgevoerd door de actor systeembeheerder;
- Templatebeheer: het creëren, aanpassen en verwijderen van standaard sjablonen (templates) die gebruikt worden voor de structuur en opmaak van content in de vorm van XML-bestanden respectievelijk XSL-stylesheets. Dit wordt verzorgd door de actor metacontentbeheerder;

De beheeractiviteiten zijn zoals vermeldt vrij onafhankelijke activiteiten, die niet als proces laat staan als workflow beschouwd kunnen worden. Desalniettemin is het mogelijk om het flexibele taakmanagement op de volgende punten ondersteuning te laten bieden voor de beheersactiviteiten ter aanvulling van handmatige of communicatieve handelingen:

- Middels het creëren van taken voor de uitvoerende van een beheeractiviteit kunnen voorstellen gedaan worden voor of opdracht gegeven worden tot het aanpassen van bepaalde instellingen of standaarden. Hierbij kan gedacht worden aan:
 - Een medewerker die fouten ontdekt in de gepubliceerde content kan dit aan de contentbeheerder doorgeven in de vorm van een taak voor contentaanpassing;
 - Een systeemgebruiker die een verandering in zijn autorisaties wenst kan middels een taak een aanvraag doen bij de systeembeheerder;
 - De hoofdredacteur (of andere manager) kan een opdracht tot aanpassing van de templates aan de metacontentbeheerder geven middels een taak.
- Het taakmechanisme kan gebruikt worden voor goedkeuring van bepaalde beheersactiviteiten die een beheerder niet zelfstandig mag uitvoeren. Dit kan indien een systeem- of metacontentbeheerder een aanpassing willen maken die mogelijk organisatiebrede consequenties heeft en derhalve op een hoger organisatorisch niveau goedkeuring behoeft (van de hoofdredacteur bijvoorbeeld). Na goedkeuring kan de beheerder vervolgens de wijziging uitvoeren.

4.4 Evaluatie workflowmanagement

Voor de ontwikkeling van een generieke en flexibele workflowondersteuning is het hanteren van een taakgeoriënteerde benadering door middel van het implementeren van het package taakmanagement geschikt gebleken. Door binnen het M-CMS ondersteuning te bieden voor alle fasen van de workflowlevenscyclus is het zowel mogelijk om een generieke procesinrichting en brede ondersteuning voor het procesmanagement te bieden als een flexibele opzet te garanderen door het mogelijk maken van procesherinrichting.

In het bijzonder is de ondersteuning voor het publicatieproces vormgegeven binnen de verschillende onderdelen van de workflowlevenscyclus:

- Een generieke procesinrichting kan op basis van UML technieken zoals het activity- en status diagram worden gemodelleerd, waarmee de procesgang, actoren en verantwoordelijkheden worden beschreven. Het taakmanagement, dat de realisatie is van het workflowmanagement, wordt ingericht om middels restricties de procesgang te bepalen;
- Het procesmanagement en -bijsturing: hiervoor is het mogelijk gebleken om hulpmiddelen aan te reiken die enerzijds de benodigde procesinformatie verschaffen in de vorm van prestatie-indicatoren en signaleringen ter bewaking van de procesgang. Anderzijds biedt het M-CMS middels het taakmanagement voldoende mogelijkheden om op verschillende operationele nivo's pro-actief te sturen en reactief bij te sturen of te escaleren. Dit levert met name voordelen op het gebied van het self-management en de attitude van betrokken actoren;
- Procesevaluatie en herinrichting: door de ondersteuning van de herinrichting van het proces, met name ten aanzien van het wijzigen van de procesgang, is tevens de flexibele opzet benodigd voor inrichting van verschillende organisaties mogelijk gemaakt.

De afzonderlijke beheeractiviteiten kunnen slechts in beperkte mate gebruik maken van de workflowondersteuning. Dit komt voornamelijk omdat het hier geen proces betreft, maar enkele individuele activiteiten, die binnen het M-CMS vertaald kunnen worden naar separate functionaliteiten. Daar waar evenwel sprake is van interactie tussen actoren op basis van het geven van opdrachten of het vragen om goedkeuring kan het taakmanagement in enige mate ondersteuning bieden ter aanvulling van handmatige of communicatieve handelingen.

Een uitgebreide toetsing en evaluatie van het gepresenteerde workflowmanagement zou pas kunnen plaatsvinden indien een realistische casus wordt gebruikt waarbij alle fasen van de workflowlevenscyclus tenminste eenmalig worden doorlopen. Dit vereist evenwel een live omgeving waarin het workflowmanagement in al zijn facetten kan worden getoetst. Dit was echter niet mogelijk (en viel niet) binnen het bereik van het afstudeerproject.

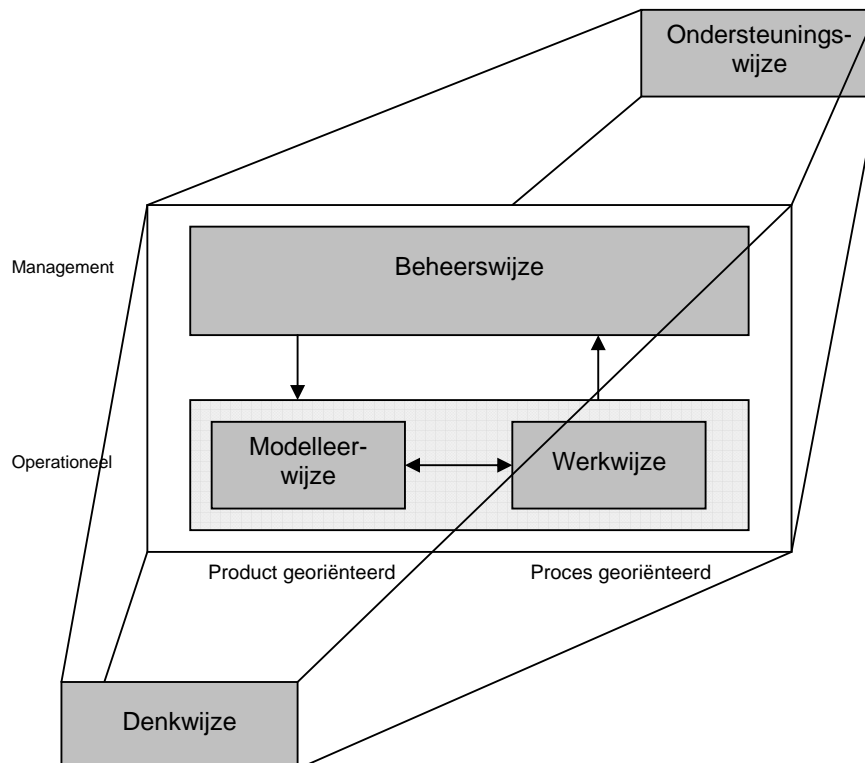
5 Het Rational Unified Process

Binnen het Multi-channel Contentmanagement Systeem (M-CMS) project is de ontwikkelmethode van het Rational Unified Process (RUP) toegepast. In dit hoofdstuk zal een invulling worden gegeven aan het onderzoeksdoel zoals geformuleerd in paragraaf 1.2 ten aanzien van het RUP en de Unified Modelling Language (UML):

Beoordeling en evaluatie van het RUP voor het M-CMS project zowel kijkend naar het incrementeel en iteratief karakter van de ontwikkelmethodiek als naar de bruikbaarheid en volledigheid van (eind- en tussen) resultaten.

Om een beeld te krijgen van een ontwikkelingsmethodiek in het algemeen en het RUP in het bijzonder kan het raamwerk voor de beschrijving van informatiesysteem methoden dienst doen [Wijers]. Hierbij worden de volgende concepten onderscheiden (figuur 5.1):

- Denkwijze: de beargumenteerde benadering en de verborgen aannamen die aan de ontwikkelwijze ten grondslag liggen;
- Werkwijze: de proces georiënteerde wijze waarmee systemen worden ontwikkeld;
- Modelleerwijze: de wijze waarop structuren worden gemodelleerd binnen de ontwikkelingsmethode;
- Beheerswijze: de manier waarop het (project) management van het ontwikkelingsproces plaats vindt;
- Ondersteuningswijze: de verzameling gereedschappen (applicaties), die de systeemontwikkeling ondersteunen.



Figuur 5.1: Raamwerk voor ontwerpmethodieken [Wijers]

In de eerste paragraaf zal een beschrijving van de component based development (CBD) benadering plaatsvinden en een argumentatie voor de initiële keuze om het RUP toe te passen. In paragraaf 5.2 zal de denkwijze van het RUP worden beschreven. In paragraaf 5.3 tot en met 5.6 zullen achtereenvolgens de werkwijze, de modelleerwijze UML, de beheerswijze en de ondersteuningswijze van het RUP worden toegelicht. In paragraaf 5.7 wordt op basis van het raamwerk van Wijers de toepassing van het RUP binnen het M-CMS project behandeld. Tenslotte wordt in paragraaf 5.8 het RUP geëvalueerd.

5.1 Component based development: de keuze voor RUP

Het multi-channel contentmanagement systeem (M-CMS) project is een software ontwikkelingsproject dat tot doel heeft een generiek CMS te bouwen, specifiek gericht op het bedienen van verschillende mediakanalen, dat inzetbaar is binnen uiteenlopende organisatievormen voor diverse contentproviders. Als voorbeeld kan men denken aan een nieuwssite als nu.nl, een webmagazine of een willekeurig bedrijf dat via verschillende kanalen zijn content wil presenteren. Hierbij zijn flexibiliteit, een generiek basisontwerp en op maat kunnen maken voor een specifieke klant de kernbegrippen.

Bij de keuze van een ontwikkelmethodiek voor dit project lag het gebruik van een component based development (CBD) methode voor de hand. CBD is een object georiënteerde (OO) benadering, die de nadruk legt op componenten: herbruikbare systeemdelen zijn die onderling via interfaces communiceren. In eerste instantie werden niet componenten maar objecten populair als middel om hergebruik van software te bevorderen [Allen]. Praktisch gezien zijn objecten evenwel van een te lage niveau van granulariteit en ontbraken standaarden voor uitgebreide toepassing. Componenten voorzien in een middel om samenhangende objecten in voorgefabriceerde stukken software samen te voegen van waaruit oplossingen worden geconstrueerd.

Een CBD paradigma waarbij gestandaardiseerde modellering wordt gehanteerd, biedt hierin goede toepassingsmogelijkheden. CBD zorgt voor een heldere en logische opzet van het systeem door een opdeling in packages, die een representatie zijn van de componenten. Een package is een container voor ontwikkelingsartefacten, die kan bestaan uit modellen, software of informatie over de structuur van andere (groepen) packages [d'Souza]. Een artefact is een door een projectmedewerker vervaardigd tussen-of eindproduct.

Hergebruik van bedrijfscomponenten is een sleutel tot het vervaardigen van flexibele software. Dit maakt een CBD methode zeer geschikt voor het M-CMS project. Er is in beginsel een keuze gemaakt voor de toepassing van het Rational Unified Process (RUP) omdat:

- RUP is CBD gebaseerde ontwikkelmethode, waardoor hergebruik van systeemonderdelen in belangrijke mate wordt ondersteunt;
- RUP is object georiënteerd en maakt gebruik van een gestandaardiseerde en breed geaccepteerde modelleringswijze, de Unified Modelling Language (UML);
- Van het RUP is voldoende literatuur en andere informatie beschikbaar, als ook voldoende ondersteuningsmiddelen voor modelleren en documentatie;

- RUP is een volwassen software ontwikkelingsmethodiek, gebaseerd op 'best practices', die zich leent voor een grote variëteit aan projecten en organisatievormen.

In de volgende paragrafen zal op basis van het raamwerk van [Wijers] de verschillende aspecten en de van de ontwikkelingsmethode RUP worden toegelicht.

5.2 De denkwijze van het RUP

De trend in software ontwikkeling is richting grotere, complexere systemen. Er is behoefte aan meer (krachtiger computers, internet informatie uitwisseling) en sneller (de time to market wordt steeds belangrijker). Het Rational Unified Process (RUP) is een ontwikkelmethode die de software ontwikkelaars een algemene benadering aanreikt die:

- Voorziet in begeleiding voor de volgorde van teamactiviteiten;
- De taken van individuele ontwikkelaars en het team als geheel aanstuurt;
- Specificeert welke artefacten moeten worden ontwikkeld;
- Criteria biedt voor het bewaken en meten van de producten en activiteiten van het project.

In de literatuur worden in plaats van RUP ook wel de termen Unified Process [Arlow] en [Reed] en Unified Software Development Process [Jacobson] gebruikt. Deze methoden zullen onder dezelfde algemene noemer RUP worden gebruikt, want het betreffen slechts marginale afwijkingen van het RUP.

Het Rational Unified Process heeft in een notepad de volgende kenmerken [Jacobson]:

- Het is een *software ontwikkelingsproces*; de verzameling activiteiten benodigd om de gebruikerseisen om te zetten in een software systeem. Meer dan dat is het RUP een generiek proces raamwerk dat kan worden gespecialiseerd voor een grote klasse software systemen, voor verschillende toepassingsgebieden, voor verschillende typen organisaties, verschillende competentienivo's en voor verschillende projectomvang;
- Het RUP is *component gebaseerd*, hetgeen betekent dat het software systeem opgebouwd wordt uit software componenten die via interfaces met elkaar communiceren;
- Het RUP gebruikt de *Unified Modelling Language* (UML) voor het prepareren van de blauwdruk van het software systeem. UML is zelfs een integraal onderdeel van het RUP;
- De onderscheidende kenmerken van het RUP betreffen de 'use-case driven', 'architectuur-centrisch', 'iteratieve en incrementele' benadering. Deze concepten worden in de eerstvolgende paragrafen nader beschreven.

5.2.1 Use case driven

Een software systeem wordt gecreëerd om de gebruikers te bedienen. Om derhalve een succesvol systeem te bouwen zijn de behoeften en eisen van de toekomstige gebruikers benodigd. Gebruikers moet in dit opzicht in breed perspectief worden gezien, want niet alleen menselijke gebruikers maar ook

andere systemen kunnen hierbij van toepassing zijn. Als beschrijving van de interactie tussen de gebruikers en het systeem worden use cases toegepast.

Een use case is een stukje functionaliteit in het systeem dat de gebruiker een waardevol resultaat geeft. De use cases samen vormen het use case model, dat de complete functionaliteit van het systeem beschrijft, vergelijkbaar met de traditionele functionele specificaties van een systeem. De use cases echter zijn niet alleen een middel om de systeemeisen te specificeren, maar ze sturen het ontwerp, de implementatie en het testen van het systeem aan. Hoewel de use cases het ontwikkelproces aansturen, worden ze niet geïsoleerd geselecteerd. De use cases worden in overeenstemming met de systeem architectuur ontwikkeld. Door de onderlinge beïnvloeding worden zowel de use cases als de systeemarchitectuur meer volwassen naarmate de levenscyclus vordert.

5.2.2 Architectuur-centrisch

Het concept van de software architectuur vertolkt de meest significante statische en dynamische aspecten van het systeem. De architectuur groeit uit de behoefte van de organisatie en wordt bovendien beïnvloed door factoren als het platform waar het systeem op draait (OS, DBMS, netwerk communicatie protocollen), beschikbaarheid van herbruikbare bouwstenen, deployment overwegingen, legacy systemen en niet-functionele systeemeisen. De architectuur en use cases vormen samen de balans van de vorm respectievelijk de functie van een software product. Bij de ontwikkeling van het systeem zullen de architectuur en use cases geleidelijk meer invulling krijgen op basis van onderlinge wisselwerking; enerzijds moeten de use cases binnen de architectuur passen, anderzijds moet de architectuur ruimte bieden voor de realisatie van de use cases.

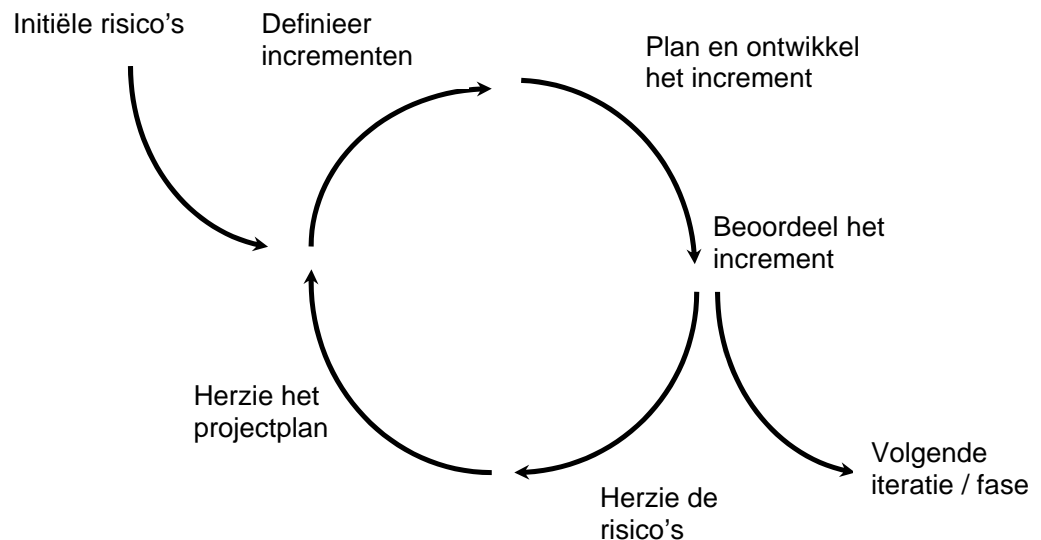
5.2.3 Iteratief en incrementeel

Het ontwerpen van een commercieel software product is een grote onderneming die maanden of zelfs jaren in beslag kan nemen. Het is hierbij praktisch om het werk te verdelen in kleinere delen. Een iteratieve en incrementele software ontwikkeling kan hierbij uitkomst bieden. Iteratief wordt hierbij gedefinieerd als de toepassing van taken op een repetitieve wijze teneinde te werken richting het verbeteren van een tussen- of eindproduct [Reed].

Binnen RUP resulteert iedere iteratie in een increment; iteraties verwijzen naar stappen binnen de workflow en incrementen naar de groei van het product. Een increment is een tussenresultaat, dat significante toegevoegde waarde heeft voor het project, dat op zichzelf staat of opereert op basis van een goed gedefinieerde interface of dat als subcomponent deel kan uitmaken van het eindproduct [Reed]. Hierin komt het CBD karakter van het RUP weer naar boven. Hierbij is een beheersing van de iteraties benodigd op basis van selectie en geplande uitvoering (zie ook paragraaf 5.5).

In figuur 5.2 staat het iteratief karakter schematisch toegelicht, in dit geval op basis van een risico-gebaseerde benadering naar [Reed]. Binnen de iteratie worden incrementen gedefinieerd op basis van initiële risico's. De incrementen worden gepland, ontwikkeld en beoordeeld en de iteratie wordt herhaald met herziene risico's en projectplan of er wordt naar de volgende fase / iteratie overgegaan. Een iteratie kan hierbij gezien worden als een deelproject,

waarvan er een aantal plaats kunnen vinden binnen een fase (zie tevens volgende paragraaf over de werkwijze).



Figuur 5.2: Iteratief en incrementeel proces RUP op basis van risicobeheersing

Resumerend is het Rational Unified Process component gebaseerd en gebruikt de visuele modelleringsstandaard UML. Het RUP is gebaseerd op drie gelijkwaardige concepten: use case driven, architectuur centrisch en iteratieve en incrementele ontwikkeling. De architectuur voorziet in een structuur waarbinnen het werk van de iteraties wordt begeleid, terwijl de use cases doelen definiëren en het werk van de iteraties invullen.

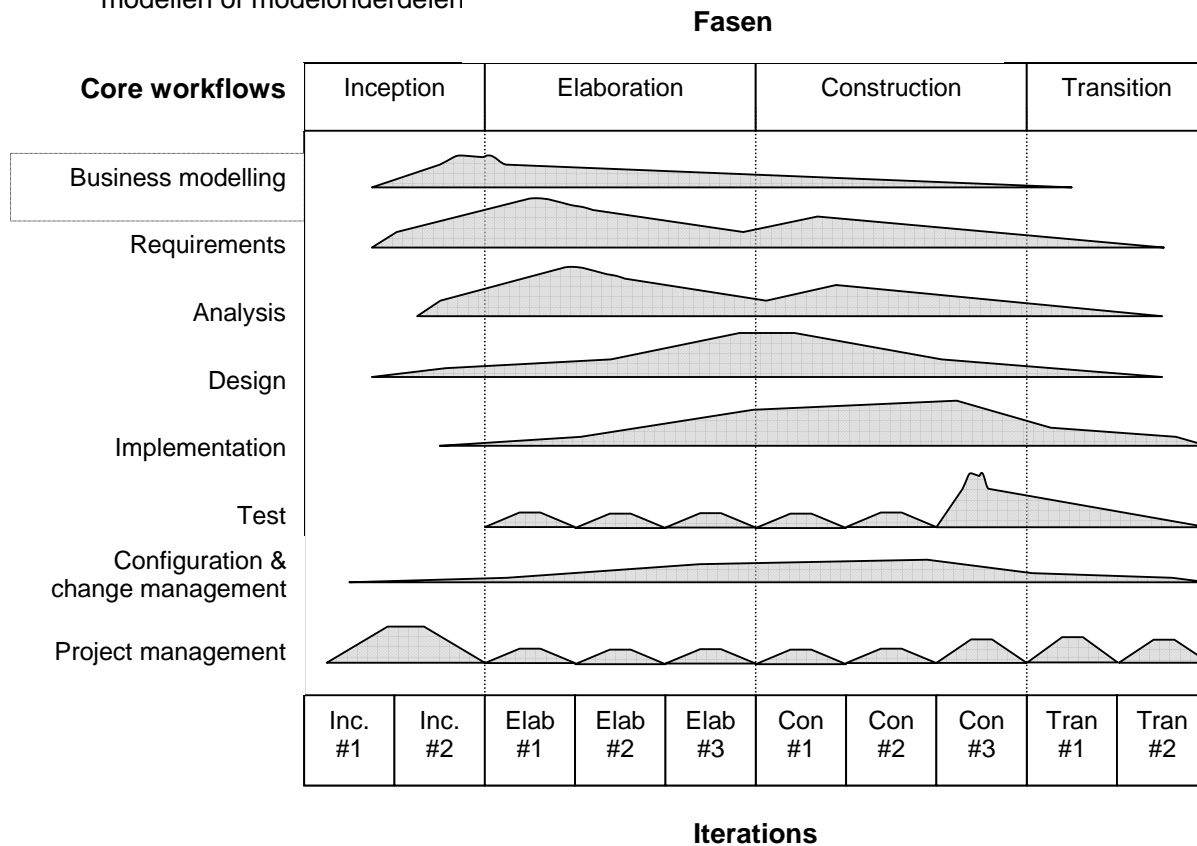
5.3 De werkwijze van het RUP

In deze paragraaf wordt de werkwijze van het RUP behandeld. De werkwijze en modelleerwijze van RUP hebben een directe onderlinge wisselwerking, waarbij de werkwijze de proceskant van het project beschrijft en de modelleerwijze de productkant vertolkt (zie ook figuur 5.1). De modelleerwijze wordt in de paragraaf 5.4 behandeld.

Binnen het RUP worden vier fasen gedefinieerd, opgedeeld in instantiëring (inception), uitwerking (elaboration), bouw (construction) en oplevering (transition). De fasen worden uitgevoerd middels zes hoofdactiviteiten (core workflows): bedrijfsmodellering (business modelling), systeemeisen (requirements), analyse (analysis), ontwerp (design), implementatie (implementation) en test. Een indicatie van fasen, workflows en iteraties binnen het RUP is gegeven binnen figuur 5.3 naar [Kruchten96]. Hierbij zijn tevens de ondersteunende workflows configuratie- en verandermanagement (configuration & change management) en projectmanagement weergegeven.

Figuur 5.3 geeft middels de curves indicatief per workflow aan in welke mate de workflow binnen de fasen wordt uitgevoerd. In de figuur zijn tevens een aantal iteraties per fase weergegeven. Aan het eind van de iteratie wordt een

increment opgeleverd dat artefacten bevat in de vorm van documenten, modellen of modelonderdelen



Figuur 5.3: Overzicht RUP in termen van fasen, workflows en iteraties

Een beschrijving van de werkwijze op hoofdlijnen [Kruchten99] kan het best worden ingedeeld naar de belangrijkste workflows, te weten bedrijfsmodellering, systeemeisen, analyse, ontwerp, implementatie en test. Deze workflows worden hieronder beschreven.

De workflow bedrijfsmodellering (in RUP: business modelling) heeft de volgende doelstellingen:

- Begrip van de structuur en dynamieken van de doelorganisatie;
- Begrip van bestaande problemen en definiëren van verbeteringspotentieel;
- Ervoor zorgen dat klanten, eindgebruikers en ontwikkelaars een gemeenschappelijk begrip van de doelorganisatie hebben;
- De eerste aanzet tot de systeemeisen te maken die de doelorganisatie nodig heeft.

Om deze doelen te behalen wordt binnen dit onderdeel een visie van de nieuwe doelorganisatie ontwikkelt en hierop worden de processen, rollen en verantwoordelijkheden van de organisatie gebaseerd in de vorm van een bedrijfs use-case model en een bedrijfs objectmodel.

De workflow systeemeisen (in RUP: requirements) heeft de volgende doelstellingen:

- Vaststellen en handhaven van overeenstemming met de belanghebbenden (in RUP stakeholders) ten aanzien van wat het systeem zou moeten doen;
- Systeem ontwikkelaars voorzien in een beter begrip van de systeemeisen;
- Zorgen voor een systeemafbakening;
- Voorzien in een basis voor het plannen van kosten en ontwikkelingstijd alsmede de technische inhoud van de iteraties;
- Het definiëren van een systeem UI, gericht op de (behoeften en doelstellingen van de) gebruikers.

Om deze doelstellingen te behalen dient allereerst een goede probleemdefinitie en –afbakening plaats te vinden, waarvoor de artefacten van de bedrijfsmodellering de input kunnen leveren. Daarnaast moeten belanghebbenden en hun interactie met het systeem worden gedefinieerd. Een visie document, een use-case model, use cases en toegevoegde systeemeisen (in RUP supplementary specifications) zorgen voor een volledige beschrijving van wat het systeem zal doen met de belanghebbenden als belangrijke bron van informatie.

De analyse- en ontwerpworkflow (in RUP: analysis and design) heeft ten doel:

- Het transformeren van de requirements (systeemeisen) naar een systeemontwerp;
- Het ontwikkelen van een robuuste systeemarchitectuur;
- Het aanpassen van het ontwerpmodel aan de implementatie omgeving, zodat het ontworpen wordt voor performance.

In bepaalde literatuur ([Jacobson] en [Arlow]) worden de analyse- en ontwerpfase strikt gescheiden gehouden, maar de grens tussen analyse en ontwerp is nogal vaag en het is in bepaalde mate ter beoordeling van de individuele analist om de streep te trekken [Arlow]. Bovendien wordt aangeraden om de analyse – en ontwerpfase door grotendeels dezelfde medewerkers te laten uitvoeren, omdat de fasen zo nauwverwant zijn. Het analysemodel en ontwerpmodel, de eindproducten van de respectievelijke fasen, vertonen dan ook op hoofdlijnen grote overeenkomsten, maar hebben als belangrijkste verschillen:

- Het ontwerp werkt in veel verder detail de UML modellen uit, vergelijkbaar met het onderscheid tussen functioneel en detail/technisch ontwerp;
- Het analyse model richt zich meer op de belanghebbenden en eindgebruikers, het ontwerpmodel meer op de ontwikkelaars en programmeurs van het systeem.

De implementatie workflow (in RUP: implementation) heeft de volgende doelstellingen:

- Het definiëren van de organisatie van de systeemcode in termen van implementatie subsystemen;
- Het implementeren van klassen en objecten in termen van componenten;
- Het testen van ontwikkelde componenten als eenheden;
- Het integreren van de individuele resultaten tot een executable systeem;

De implementatie workflow beperkt zich wat testen betreft tot het bereik van hoe individuele klassen worden getest als eenheid (het zogenaamde unit testen). De systeem- en integratietesten behoren tot de testfase.

Workflow	Actoren	Artefacten	UML modellen
Bedrijfs modellering	Bedrijfsproces analist Business ontwerper Business model reviewer	Bedrijfsproces (tekstueel) Bedrijfsvisie Bedrijfs use case model Bedrijfs objectmodel Begrippenlijst	Use case model
Systeemeisen - Requirements	Systeme analist Systeme architect Use case specificerder UI ontwerper	Visie statement (product) Stakeholder requests Actors and use cases Use case model Plan beheer systeemeisen Model user interface (UI) UI prototype	Use case specificaties Use case model
Analyse	Systeme architect Systeme ontwerper Arch. / design reviewers Database ontwerper	Analyse model: Use case model (analyse) Klasse model Architectuur analyse Database analyse	Klasse diagram Use case specificaties (definitief) Use case model (def)
Ontwerp	Systeme architect Systeme ontwerper Arch. / design reviewers Database ontwerper	Ontwerp model: Reference architectuur Use case realisatie Ontwerp packages Data model	Klasse diagram (def) Sequence diagram Collaboratie diagram State diagram Activity diagram
Implementatie	Systeme architect Integratoren Programmeurs	Implementatie model: Componenten Integratie bouwplan Implementatie subsystemen	Componenten diagram Deployment diagram
Test	Testers Ontwerpers Programmeurs Test ontwerper	Test model: Test plan / procedures Test resultaten Component testen Test evaluatie	N.v.t.

Tabel 5.1: Workflows met actoren, belangrijkste artefacten en UML modellen

De testworkflow heeft als doelstellingen:

- Verifiëren van de interactie tussen objecten;
- Verifiëren van de juiste integratie van alle componenten van het systeem;
- Verifiëren dat alle systeemeisen correct zijn geïmplementeerd;
- Identificeren en verhelpen van defecten voorafgaand aan deployment van het systeem.

Hierbij verdient het aanbeveling om zoveel mogelijk testen al vroegtijdig uit te voeren, opdat hier hoofdzakelijk de systeem- en integratietesten moeten worden uitgevoerd.

In tabel 5.1 worden de verschillende fasen nader toegelicht met betrokken actoren, de belangrijkste artefacten en UML modellen (zie paragraaf 5.4). De tabel is niet volledig en derhalve indicatief maar geeft een aanvullend beeld van de hoofdworkflows van RUP.

5.4 De modellerwijze van het RUP: UML

De modellerwijze wordt binnen het RUP in de UML specificatietaal [OMG] verzorgd. UML heeft de volgende grafische modellen beschikbaar, welke met name in de analyse- en ontwerpfasen worden toegepast en in mindere mate in de requirement en implementatie fasen (zie tevens tabel 5.1):

- Het use case model: een diagram voor het modelleren van de functionaliteiten van het systeem alsmede het gebruik ervan door externe interactoren;
- Het klasse (en object) model: een diagram voor het modelleren van de statische structuur van het systeem, die de klassen en objecten, hun interne structuur en hun onderlinge relaties modelleert;
- Diagrammen voor het modelleren van gedrag:
 - Het state(chart) diagram: beschrijft de levenscyclus van klassen aan de hand van diens reactie op externe gebeurtenissen;
 - Het activity diagram: een variatie op het statechart diagram waarbij de statussen de performance van acties representeren en de transities worden getriggered door de afhandeling van acties;
- Diagrammen voor het modelleren van interactie:
 - Het sequence diagram: beschrijft een use case realisatie middels interactie tussen objecten, in het bijzonder de lineaire volgtijdelijkheid;
 - Het collaboratie diagram: beschrijft feitelijk hetzelfde als de sequence diagrammen, maar met de focus op individuele objecten en hun interactie;
- Diagrammen ter ondersteuning van de implementatie:
 - Componenten diagram: beschrijft de onderlinge afhankelijkheden van software componenten op klasse niveau;
 - Deployment diagram: beschrijft de configuratie van run-time processing elementen en de software componenten, processen en objecten die 'execute on them'.

Voor het gebruik van de verschillende modellen wordt verwezen naar bijlage C, waar de belangrijkste UML modellering voor het M-CMS project is opgenomen en toegelicht.

5.5 De beheerswijze van het RUP

De beheerswijze van het RUP wordt vorm gegeven door de ondersteunende workflow projectmanagement. Het doel van het project management binnen RUP is:

- Een raamwerk te leveren voor het beheersen van software intensieve projecten;
- Praktische richtlijnen aan te bieden voor planning, bemannen, uitvoeren en bewaken van projecten;
- Een raamwerk te leveren voor het beheersen van risico's.

Het projectmanagement richt zich daarbij vooral op de belangrijkste aspecten van een iteratieve ontwikkelings workflow, te weten risico analyse, het plannen van een iteratief project (in zijn geheel als ook specifieke iteraties) en het bewaken van de voortgang. In beginsel wordt een projectplan geschreven (software development plan), dat het uitgangspunt vormt voor het project.

5.6 De ondersteuningswijze van het RUP

De ondersteuningswijze is de verzameling gereedschappen (overwegend applicaties), die de systeemontwikkeling ondersteunen. Er zijn verschillende applicaties op de markt die het gebruik van RUP ondersteunen. Rational Rose® [*Rational*] is een specifiek voor RUP ontwikkelde applicatie, die het creëren, onderhouden en transformeren van modellen ondersteunt. Voorbeelden van functionaliteiten zijn het automatisch genereren van collaboratie diagrammen op basis van sequence diagrammen en het genereren van code op basis van het klassediagram.

[*Rational*] biedt bovendien ondersteuning voor het RUP als commercieel product middels online handleidingen, sjablonen voor documentatie en diverse gereedschappen voor bijvoorbeeld ondersteuning van analyse- en ontwerpfase en het testen.

Andere applicaties die niet specifiek gericht zijn op het RUP, zoals Microsoft Visio, bieden tenminste een aardige ondersteuning voor UML middels tekensjablonen en ondersteuning voor de invulling van datatypen. Deze applicaties zijn een goedkoop doch minder compleet alternatief voor de commerciële RUP ondersteuningsmiddelen.

5.7 Toepassing van het RUP binnen het M-CMS project

In deze paragraaf zal gekeken worden naar de toepassing van het RUP binnen het M-CMS project. De toepassing van een ontwikkelingsmethode is voor ieder project verschillend. Binnen de methode dienen keuzes gemaakt te worden wat betreft mensen, modellen, artefacten en dergelijke. Het RUP is in het algemeen al veel te omvangrijk om in zijn geheel toe te passen, in het bijzonder binnen een éénmansproject. Het M-CMS project heeft nog enkele specifieke eigenschappen die bepalend zijn geweest voor de keuzen die binnen het project zijn gemaakt:

- Het project wordt door één persoon uitgevoerd, die verschillende actorrollen vervuld, zoals analist, ontwerper, programmeur en projectleider;
- Het eindproduct wordt niet voor een enkele organisatie ontwikkeld, maar moet een basisproduct zijn dat voor een generieke

doelorganisatie geschikt is. Hierdoor is met name de systeemarchitectuur beperkt ingevuld vanwege de (voor zover mogelijk) architectuuronafhankelijkheid van het product;

- Als uitgangspunt is gekozen voor een experimentele implementatie in de vorm van het bouwen van een prototype;
- De keuze van het bedrijf om geen grafisch ondersteuningsmiddel (Rational Rose) aan te schaffen, dat specifiek voor RUP geschikt is.

Het bovenstaande en de praktische overwegingen hebben de nodige gevolgen gehad bij de toepassing van het RUP. Hieronder worden achtereenvolgens de toepassing belicht van de denkwijze, de werk- en modelleerwijze en de beheers- en ondersteuningswijze.

5.7.1 Toepassing denkwijze

De denkwijze van het RUP is een gezonde benadering van software ontwikkeling en is in grote lijnen gevolgd. Op de volgende punten is echter afgeweken:

- Het architectuur centrische aspect is weinig belicht, omdat het basis M-CMS product geschikt moet zijn voor verschillende architecturen. Wel is binnen de ontwerpfase een basis systeemarchitectuur neergezet, die generiek is toe te passen voor verschillende organisatievormen en platforms;
- De toepassing van iteraties, deze worden in paragraaf 5.7.4 behandeld.

5.7.2 Toepassing werkwijze

Ten aanzien van de gevolgde werkwijze zijn de volgende aanpassingen ten opzichte van het RUP gesignaleerd (zie tabel 5.2 voor de opgeleverde artefacten per workflow):

- Vanwege het eenmanskarakter van het project kon overhead worden beperkt. Met name het gebruik van artefacten gericht op een goede communicatie en overdracht van tussenproducten tussen verschillende actoren is tot een minimum beperkt;
- Het toepassen van prototyping in plaats van een volledig implementatie- en testtraject uit te voeren, waardoor:
 - De workflow implementatie in beperkte mate aan de orde is gekomen in de vorm van het prototype en het unit testen ervan en de workflow testen slechts op de werking van de contentpijpleiding is toegepast;
 - De verschillende RUP hoofdworkflows een verschillende hoeveelheid aandacht kregen: bedrijfsmodellering en requirements vrij volledig, analyse en ontwerp voor de essentiële onderdelen van het systeem en de implementatie slechts voor die componenten die in het prototype worden opgenomen.

5.7.3 Toepassing modelleerwijze

Ten aanzien van de gevolgde modelleerwijze zijn de volgende aanpassingen gesignaleerd (in tabel 5.2 zijn de toegepaste artefacten per workflow weergegeven):

- Er is geen redundante modellering opgenomen, maar enkel de meest geschikte modellen zijn toegepast binnen het project. Een voorbeeld hiervan: de sequence en collaboratie diagrammen beelden hetzelfde uit, maar met een andere invalshoek. Aangezien sequence diagrammen de volgtijdelijkheid het beste verbeelden en beter aansluiten bij de implementatie is daarvoor gekozen;
- Bepaalde modellen zijn niet of slechts in beperkte mate uitgewerkt: activity diagrammen in het geheel niet, state diagrammen zijn voor een enkele klasse uitgewerkt. Het nut van de modellen was op dit punt in het project te beperkt;
- De UML modellering en RUP documentatie was nagenoeg overall toereikend, behalve ten aanzien van het modelleren van het publicatieproces. Om een goed beeld te schetsen anders dan tekstuele beschrijvingen zijn hiervoor het DEMO communicatiediagram en procesdiagram gebruikt ter aanvulling op de UML modellering.

In bijlage C zijn voor het ontwerp- en implementatiemodel de relevante UML modelleringen opgenomen die in tabel 5.2 vermeld staan.

Workflow	RUP documenten	UML modellen
Bedrijfs modellering	Bedrijfsprocessen Visie statement (mbt product) Begrippenlijst	Niet beschikbaar, derhalve is DEMO modellering gebruikt
Systeemeisen - Requirements	Stakeholder requests Software requirement specificatie (SRS) Niet functionele specificaties	Use case specificaties (1 ^e versie) Use case model (1 ^e versie)
Analyse	Analyse model: Architectuur analyse (basis)	Klasse model (1 ^e versie) Use cases (definitieve versie) Use case model (def. versie)
Ontwerp	Ontwerp model: Referentie architectuur (basis) Ontwerp packages UI prototype (systeemeis fase)	Klasse model (definitief) Use case realisatie (in de vorm van sequence diagrammen) State diagrammen (essentials)
Implementatie	Implementatie model: Componenten Prototype	Aanpassingen aan UML diagrammen voor zover van toepassing
Test	Test model: Test resultaten	N.v.t.

Tabel 5.2: Toepassing RUP bij het M-CMS project gerelateerd aan artefacten

5.7.4 Toepassing beheerswijze

De beheerswijze zoals deze door het projectmanagement van RUP wordt voorgeschreven is slechts beperkt toegepast. Dit is enerzijds het gevolg van het eenmans karakter en de beperkte omvang van het project. Anderzijds hebben de tijdfactor en haperingen bij de uitvoering enkele additionele randvoorwaarden opgeworpen. Wel is gebruik gemaakt van het Software development plan (SDP, een plan van aanpak voor het project). Tevens heeft de bijsturing van het project op basis van iteraties binnen de fasen plaatsgevonden. Deze iteraties vonden met name plaats in de vorm van het herzien van eerder opgeleverde documentatie / modellering op basis van voortschrijdende inzichten.

5.7.5 Toepassing ondersteuningswijze

Ten aanzien van de ondersteuningswijze kunnen de volgende uitspraken worden gedaan:

- Er is geen RUP-specifieke modelleringstool gebruikt. In plaats daarvan is gebruik gemaakt van Microsoft applicaties voor documentatie en van Microsoft Visio in het bijzonder voor de UML modellering;
- Er is gebruik gemaakt van een trialversie van het RUP voor wat betreft online handleidingen en sjablonen voor documentatie, die als uitgangspunt dienden voor documentatie die niet modelgerelateerd was;
- Bij de implementatie in de vorm van de bouw van een prototype was geen beschikking over Rational ondersteuningsmiddelen. Er is voor gekozen om de ontwikkeltool Intelligent Objects (IO) te gebruiken. Voor een beschrijving en evaluatie van IO wordt verwezen naar het volgende hoofdstuk.

5.8 Evaluatie van het RUP

In deze paragraaf zal het RUP worden geëvalueerd op basis van het raamwerk van Wijers. In zijn algemeenheid valt te constateren dat het RUP een prettige ontwikkelingsmethode is, die goed gedocumenteerd is en nuttige (commerciële) ondersteuningsmiddelen biedt. Het RUP is echter zeer omvangrijk en om te komen tot een effectieve ontwikkeling van het product, moet het project team slechts die iteraties selecteren die benodigd zijn om het doel van het project te behalen [Jacobson] en dit geldt ook voor de selectie van de diverse artefacten.

5.8.1 Denkwijze van het RUP

Het RUP biedt een heldere denkwijze, gebaseerd op de concepten use case driven, architectuur centrisch en een iteratieve en incrementele benadering. Hiermee wordt het traditionele lineaire waterval model van ontwerpmethodieken losgelaten, zoals dat binnen andere object georiënteerde en CBD methoden ook het geval is, waarmee dynamische ontwikkeling wordt ondersteunt.

Het use case driven concept zorgt ervoor dat de eindgebruikers en opdrachtgever(s) snel betrokken worden en vooral blijven (dankzij de iteraties) bij de systeemfunctionaliteit en –ontwikkeling. Een probleem dat zich nogal eens voordoet bij IT-projecten, dat het eindproduct (sterk) afwijkt van wat de opdrachtgever bedoeld heeft, wordt hiermee ondervangen.

Het architectuur centrische concept is binnen het M-CMS project nauwelijks belicht vanwege de architectuur / platform onafhankelijkheid van het eindproduct. Het is evenwel een zeer nuttige benadering, die in vroegtijdig stadium de functionaliteit (use cases) en technische realisatie (architectuur) op elkaar afstemt. Hiermee wordt voorkomen dat na maanden arbeid een schitterend ontwerp wordt opgeleverd, dat (architectuur) technisch niet of beperkt realiseerbaar blijkt.

De iteratieve en incrementele benadering ondersteunt het dynamisch modelleren. Enerzijds wordt bewerkstelligd dat stapsgewijs wordt gewerkt naar een kwalitatief goed tussen- of eindproduct. Anderzijds worden door de

onderverdeling in iteraties de RUP workflows overzichtelijk en beheersbaar gehouden (zie tevens paragraaf 5.8.4). Binnen het M-CMS project werden iteraties vooral gebruikt om resultaten uit een later stadium van het project met terugwerkende kracht te verwerken in eerdere documentatie / modellen, teneinde de consistentie van het project en product te waarborgen.

5.8.2 Werkwijze van het RUP

De werkwijze van het RUP is logisch, inzichtelijk en werkbaar gebleken. Het is zowel in de literatuur als middels Rational ondersteuningsmiddelen goed gedocumenteerd en van richtlijnen voorzien. Hierdoor komen de keuzes die gemaakt moeten worden binnen een ontwikkelingsproject goed naar voren. De keuzes richten zich op het hoe en welke workflows en iteraties toe te passen en welke artefacten op te leveren. Traditionelere ontwerpmethodieken zijn veelal voorschrijvend van aard, terwijl het RUP de activiteiten en het maken van keuzes beschrijft, waardoor een flexibele toepassing binnen verschillende project- en organisatievormen mogelijk wordt. Het RUP is veelomvattend en zeer uitgebreid, maar kan binnen kleinere projecten en organisaties nuttig worden toegepast, mits enige ervaring met de methode aanwezig is en de tijd genomen kan worden om de bovengenoemde keuzes weloverwogen te nemen.

5.8.3 Modellerwijze van het RUP

De modellerwijze van het RUP, UML, is een breed geaccepteerde modellerstandaard die object georiënteerd is en daarmee aansluit op de de facto standaarden. De modellen zelf zijn inzichtelijk, in belangrijke mate ook voor eindgebruikers en opdrachtgevers, ondanks de verschuiving van het zwaartepunt van generieke modellering aan het begin van het project naar meer specifieke en technische modellering richting de implementatie. De modellen vullen elkaar vanuit verschillende invalshoeken goed aan om tot een complete modellering van het product te komen. Bovendien sluiten de OO modellering goed aan op de facto standaarden als Java of .NET producten zoals Visual Basic. Het enige gebrek is, voortgekomen uit het systeemontwikkelkarakter van de RUP methode, dat er geen adequate modellering beschikbaar is voor het vastleggen van het publicatieproces, anders dan middels tekstuele beschrijvingen.

Een belangrijk speerpunt binnen het RUP is het laten aansluiten van documentatie en modellering van verschillende workflows, in het bijzonder als deze worden uitgevoerd door verschillende actoren. Binnen het M-CMS project is vanwege het éénmans karakter hiervan weinig gebruik gemaakt, maar de overhead kon worden beperkt door het maken van een adequate selectie van de artefacten. Dit aspect is binnen grotere projecten met meerdere medewerkers echter een belangrijke factor om het project beheersbaar te houden en om te voorkomen dat er (soms aanzienlijke) discrepanties in de benadering en tussenproducten van de verschillende workflows optreden.

5.8.4 Beheerswijze van het RUP

Het projectmanagement van het RUP biedt vele mogelijkheden om een project overzichtelijk en beheersbaar te houden. Het plannen, de risico analyse op basis van metrieken en de voortgangsbewaking geven goede handvaten voor een beheersbaar project. Bovendien zorgt het iteratieve en incrementele

benadering van het RUP, waaraan in paragraaf 5.8.1 al werd gerefereert, voor een heldere doch niet statische opdeling in logische onderdelen en systeemcomponenten die afzonderlijk te beheersen zijn.

5.8.5 Ondersteuningswijze van het RUP

De commerciële Rational producten bieden een goede ondersteuning voor het modelleren en documenteren van een ontwikkelingsproject. Dit is voor grotere projecten ongetwijfeld nuttig zo niet noodzakelijk. Minder specialistische alternatieven kunnen voor kleinere projecten al uitkomst bieden zoals Microsoft Visio, die een aardige ondersteuning biedt voor de analyse- en ontwerp workflows, hoewel de transitie naar de implementatie niet wordt ondersteunt. In de praktijk is dit geen groot gemis, daar de transitie naar implementatie vaak zelf door de programmeur uitgevoerd wordt, omdat anders een weinig flexibel implementatie model resulteert.

6 Intelligent Objects tool

In dit hoofdstuk wordt de ontwikkeltool Intelligent Objects (IO) beschreven en de toepassing ervan voor het prototyping van het multi-channel contentmanagement systeem (M-CMS). IO is een objectgeoriënteerde ontwikkeltool waarmee kennisgebaseerde systemen vervaardigd kunnen worden. Binnen het M-CMS project is bij het prototypen de keuze voor de toepassing van IO gemaakt, omdat de tool het mogelijk maakt om snel tot ontwikkeling van een (werkend) prototype te komen. Bovendien sluit de objectge-oriënteerde benadering van IO goed aan op het RUP en in het bijzonder het modelleren in UML, omdat IO ontwikkeld is op basis van de UML Specificatie. Het alternatief van het programmeren van een prototype in Java of dot.Net producten zoals Visual Basic zou minder effectief zijn. In paragraaf 6.1 zal bij de beschrijving van IO worden aangetoond waarom het M-CMS als kennisgebaseerd systeem kan worden gezien.

Bovendien levert het toepassen van IO binnen de ontwikkeling van een internet gerichte applicatie voor het bedrijf Thinkworks een testcasus op, waaraan de functionaliteiten van de tool gespiegeld kunnen worden. In het bijzonder is bij de toepassing van IO als ontwikkeltool aandacht besteed aan het volgende onderzoeksdoel:

De toetsing en evaluatie van de geschiktheid van de ontwikkeltool IO voor de toepassing bij systemen die zich richten op internettechnieken.

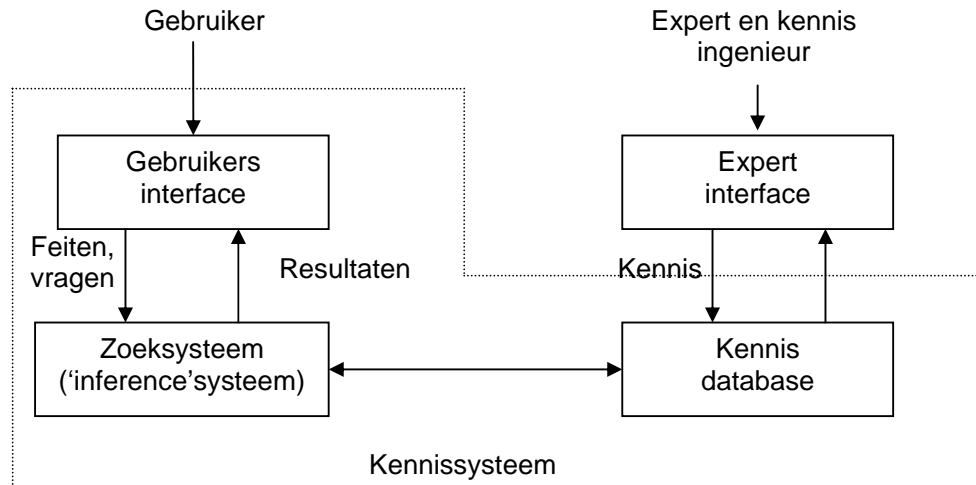
Tevens is de casus een goede manier om de bruikbaarheid en toegankelijkheid van de tool te evalueren door een ontwikkelaar die niet bekend is met de tool. In paragraaf 6.1 vindt een algemene beschrijving plaats van IO. In paragraaf 6.2 wordt vervolgens de toepassing van IO beschreven binnen het M-CMS project. Tenslotte wordt IO in paragraaf 6.3 geëvalueerd op basis van algemene kenmerken, aansluiting op UML en RUP en de ondersteuning voor internet-gebaseerde standaarden.

6.1 Beschrijving Intelligent Objects

In deze paragraaf wordt een generieke beschrijving gegeven van de ontwikkeltool Intelligent Objects (IO) die bij de prototyping van het M-CMS project is gebruikt. IO is een objectgeoriënteerde (OO) reflectieve ontwikkeltool om kennisgebaseerde systemen of expertsystemen mee te bouwen. Reflectief wil zeggen dat het systeem in run time herzien en uitgebreid kan worden, tevens op type niveau. Een kennisgebaseerde systeem of kortweg kennissysteem is een computersysteem dat kennis representeert en gebruikt om een taak uit te voeren [Stefik]. Kennissysteem is een wat algemener begrip dan expertstelsel (een computersysteem waarvan de performance wordt gestuurd door specifieke expertkennis bij probleemoplossen [Stefik]) omdat bij kennissysteem in het midden wordt gelaten of voor de kennis expertise nodig is.

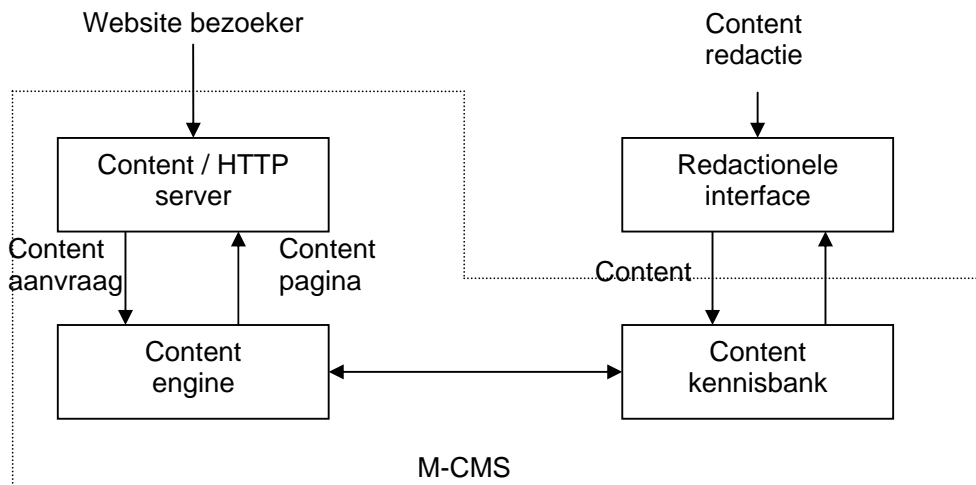
In figuur 6.1 is de klassieke karakterisering van een kennissysteem opgenomen. Hierin zijn een kennisdatabase, twee mens-computer interfaces en een zoek- of inferentie-systeem opgenomen. Een kennisdatabase of kennisbank (KB) is een opslagplaats voor de kennis die gebruikt wordt door het systeem: de regels en richtlijnen om het zoeken naar oplossingen te

begeleiden. De expertinterface is het systeemdeel waarbij kennis in het systeem kan worden ingegeven. Het zoekstelsel (of in het Engels 'inference system') bevat het mechanisme dat de probleemoplossingen berekent aan de hand van het doorzoeken van de inhoud van de kennisbank.



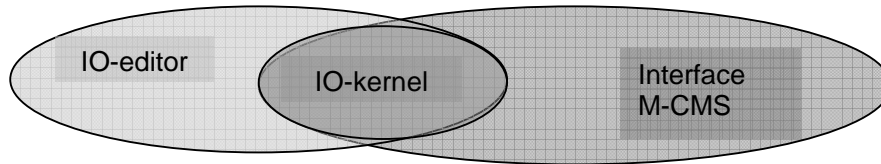
Figuur 6.1: Klassieke karakterisering van een kennisstelsel [Stefik]

Het begrip kennis heeft meerdere betekenissen, waaronder informatie als een verzameling feiten [Stefik]. Wanneer de content van het M-CMS als kennis wordt beschouwd, kan het M-CMS in de terminologie van het kennisstelsel van figuur 6.1 worden gevat. In figuur 6.2 is het M-CMS benaderd in de termen van het kennisstelsel. Aangezien het M-CMS op globaal nivo als kennisstelsel kan worden beschouwd, kan de ontwerptool IO gebruikt worden bij het ontwikkelen van het prototype.



Figuur 6.2: M-CMS gezien als een kennisstelsel

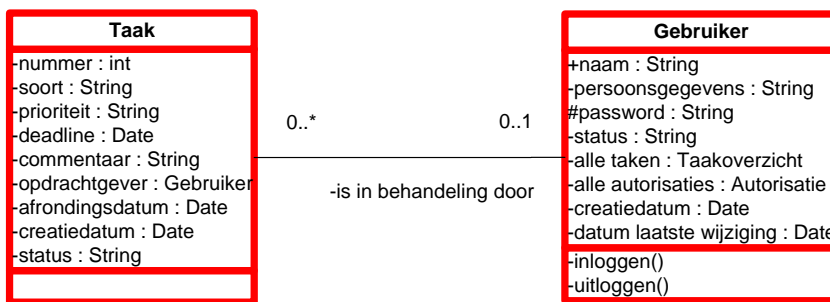
IO bestaat uit een IO-editor en een IO-kernel: de IO-editor is een userinterface die gebruikt wordt voor het modelleren van de kennisstelsels, in dit geval voor het M-CMS. De kernel is een compiler / interpretator, die elementen bevat en expressies en statements interpreteert en executeert.



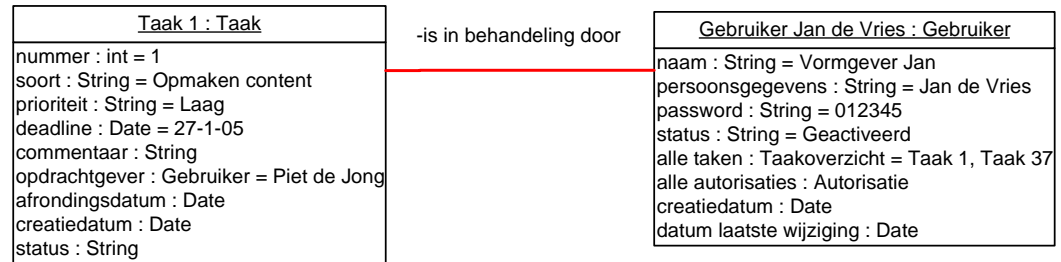
Figuur 6.3: Verschillende lagen in IO structuur

In figuur 6.3 zijn de verschillende lagen binnen IO aangegeven, waarbij de interface van het M-CMS een redactionele interface (zoals in figuur 6.2 geschetst) of beheersinterface kan zijn, die het mogelijk maakt om bijvoorbeeld content, stylesheets en templates te creëren of te wijzigen. De IO-editor is een object georiënteerde modelleeromgeving waarbinnen het ontwerp- en implementatiemodel van het Rational Unified Process (RUP) op basis van de Unified Modelling Language (UML) modellering is vast te leggen. De IO-editor en de M-CMS interface hebben in figuur 6.3 feitelijk geen overlap anders dan hun gezamenlijk gebruik van de IO-kernel.

Binnen IO wordt duidelijk onderscheid gemaakt tussen het type- en instantieniveau, die een directe representatie zijn van het klasse- respectievelijk objectniveau van UML. Een klasse is een statische datastructuur die (de eigenschappen en het gedrag van) een entiteit definieert [OMG]. Deze eigenschappen en het gedrag kun je volgens de OO benadering overerven. Een object is een specifieke instantie van een klasse. Een voorbeeld van een deel van het klassemodel is opgenomen in figuur 6.4. De klassen Gebruiker en Taak zijn hierin weergegeven met hun 0..* op 0..1 relatie ‘is in behandeling door’: iedere taak is in behandeling door nul of één gebruiker, maar een gebruiker kan 0 of meerdere taken in behandeling hebben. Het klassemodel definieert de structuur van objecten die tot de klasse behoren en kent tevens attributen, associaties en procedures toe. In figuur 6.5 is een objectdiagram weergegeven van een specifieke gebruiker en taak. Hierin is een specifieke taak 1 gedefinieerd, die door gebruiker Jan de Vries in behandeling is.



Figuur 6.4: Deel van klassemodel M-CMS

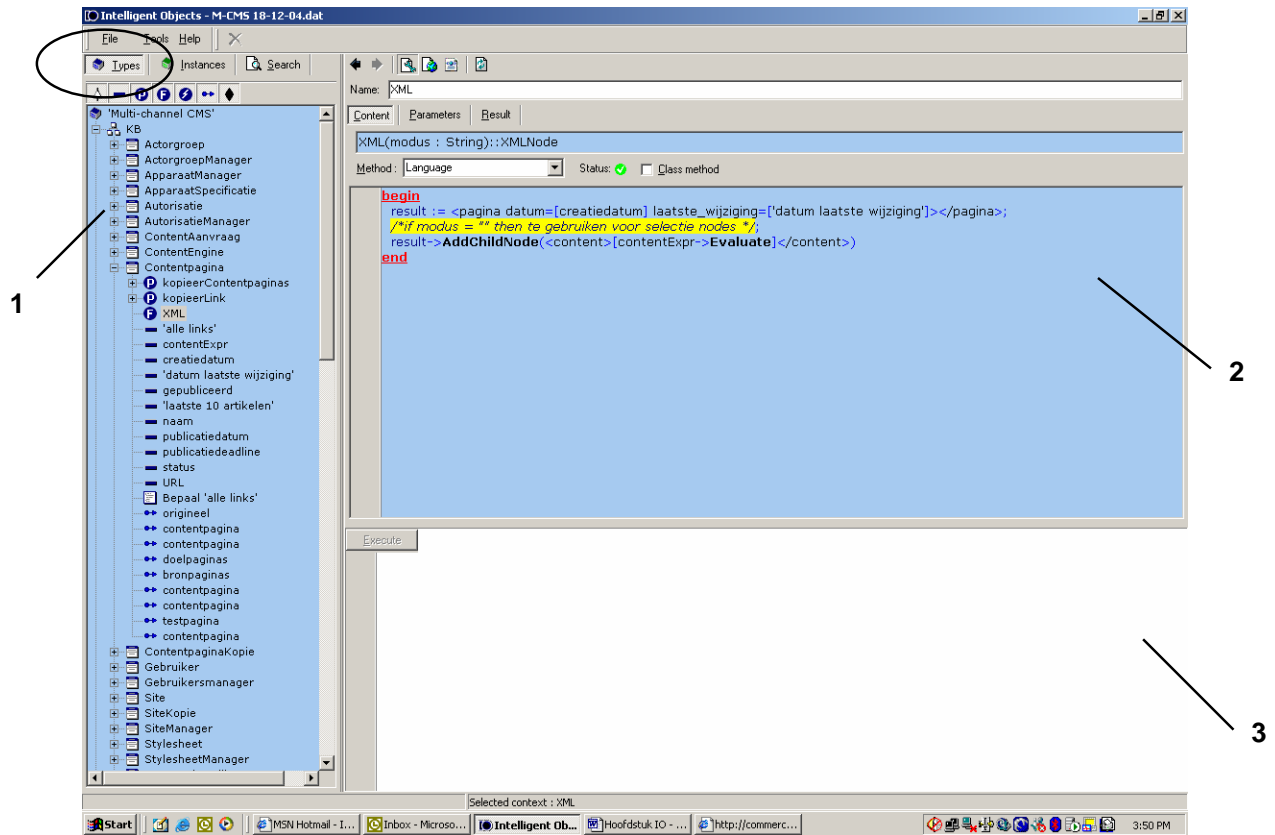


Figuur 6.5: Objectdiagram van taak 1 en gebruiker Jan de Vries

De IO editor kent op type-niveau een kennisbank (KB); de hoofdstructuur die de volgende structuren bevat:

- Klassen: de UML klassen met daarbinnen:
 - Attributen: de eigenschappen die objecten in de klasse kunnen hebben en mogelijk het domein van de attribuutwaarde;
 - Functie: een operatie met parameters die door de klasse wordt uitgevoerd, in het bijzonder een die een resultaat teruggeeft;
 - Procedure: een operatie die geen resultaat teruggeeft;
 - Composiet-relatie: de 1 op x relatie tussen klassen y en z, waarbij een object van de klasse y 'eigenaar' is van een object van de klasse z en daarmee verantwoordelijk is voor de creatie en verwijdering;
- Associatie: de x-op-y relaties tussen verschillende klassen;
- Containers, die objecten kunnen bevatten. De volgende vormen zijn aanwezig:
 - Set: een ongeordende verzameling van gelijksoortige, unieke entiteiten;
 - Bag: een ongeordende verzameling van gelijksoortige entiteiten (die vaker voor mogen komen);
 - Sequence: een geordende lijst van gelijksoortige entiteiten (die vaker voor mogen komen). Deze lijst wordt in het M-CMS voornamelijk gebruikt;

In figuur 6.6 is een voorbeeld opgenomen van de IO editor op het type niveau.

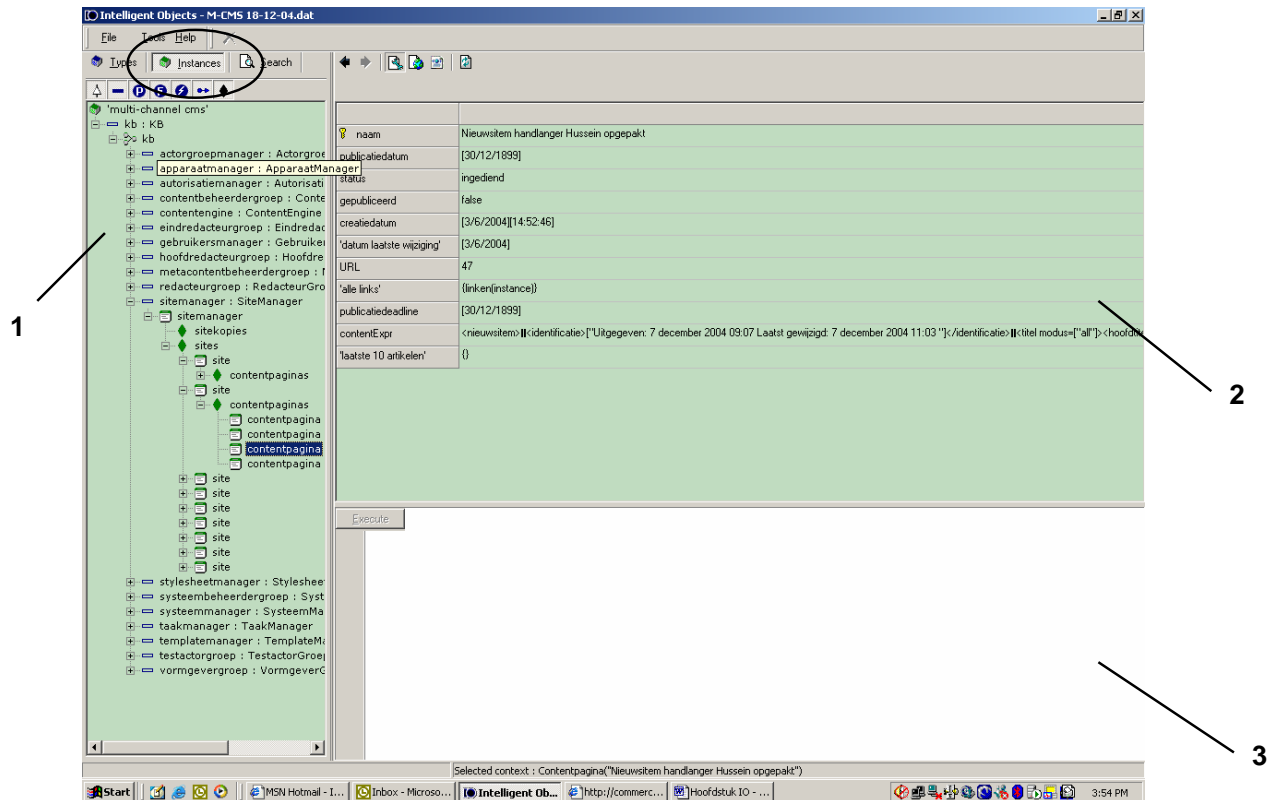


Figuur 6.6: screenshot van het type niveau van de IO editor

De verschillende genummerde kaders in figuur 6.6 bevatten de volgende onderdelen:

1. Het linker kader geeft het klassemodel weer van het M-CMS. Hier kunnen de klassen, attributen en functies worden aangemaakt en de onderlinge klasserelaties. De rondjes met een pijl (●→ symbolen) staan voor de rollen die een klasse kan vervullen binnen een bepaalde associatie ;
2. Het kader rechtsboven beeld van een geselecteerd onderdeel uit het linker kader een aantal eigenschappen af. Dit kan zijn voor een attribuut het type (bijv. String of Boolean) en domein of voor een functie (zoals in figuur 6.6 geselecteerd) de parameters die worden meegegeven, het resultaat van de functie en de methode en content (inhoud) van de methode die het resultaat bepaald. De methode kan zijn:
 - a. 'Language': IO-programmeertaal, gelijkend op de OCL expressietaal. Dit is de meest flexibele en meest gebruikte methode om een functie in uit te voeren. Andere methoden zijn in specifieke gevallen toepasbaar;
 - b. 'Decisiontable': met name in geval van een aaneenschakeling van geneste if-then-else statements kan dit een praktisch alternatief zijn;
 - c. 'Flowchart': met name in geval van afwisseling van acties en beslissingen kan een flowchart een praktisch alternatief zijn.
3. Het lege kader rechtsonder wordt gebruikt voor het evalueren of uitvoeren van tijdelijke expressies of statements. Dit betreft onder andere het aanmaken van objecten op instantienivo middels de OCL (Object Constraint Language) expressietaal, die in [OCL] staat beschreven.

Hierover volgt meer bij de toelichting van het instantienivo, zie tevens figuur 6.7.



Figuur 6.7: Screenshot van IO-editor instantie nivo

In figuur 6.7 is een voorbeeld van het instantie nivo van IO gegeven, waarbij de verschillende genummerde kaders de volgende onderdelen bevatten:

1. Het linker kader geeft een gelaagd overzicht van de instanties; attributen binnen de kennisbank op type nivo worden objecten op instantienivo. Weergegeven is dat de sitemanager (singleton) uit een aantal sites bestaat en de site uit 4 contentpagina's;
2. De contentpagina die in het linker kader is geselecteerd, wordt in het kader rechtsboven afgebeeld met zijn attributen en hun waarden. Hierbij kunnen een aantal afbeeldingsvormen worden geselecteerd:
 - a. Edit mode: laat de objecten met attributen en waarden zien (zoals in figuur 6.7 van een contentpagina instantie is weergegeven);
 - b. HTML mode: een dynamische view waarbij links staan weergegeven, die via relaties naar andere objecten kunnen worden gevolgd;
 - c. XML mode: de XML presentatie van het betreffende element;
3. Het rechtsonder kader wordt op instantienivo vaak gebruikt om met de OCL expressietaal instanties aan te maken, attributen een waarde te geven of functies uit te voeren. Hierbij dient een specifieke context te worden geselecteerd, die aangeeft op welke locatie binnen de kennisbank de expressie moet worden uitgevoerd. Feitelijk geeft OCL de mogelijkheid om alle onderdelen die op het typenivo zijn gedefinieerd binnen het instantienivo aan te maken en te manipuleren. Dit is derhalve een wat

grove interface te vergelijken met de expertinterface uit figuur 6.1 van [Stefik].

6.2 Prototyping van het M-CMS binnen IO

De beoordeling van de mate waarin IO aansluit op de implementatie van het M-CMS project kan ingedeeld worden naar 2 aspecten:

1. De ondersteuning van de UML-modellering en de mate waarin IO aansluit bij de werkwijze en beheerswijze van het RUP;
2. De geschiktheid van IO voor het toepassen bij systemen die gebruik maken van internettechnieken zoals gedefinieerd in het onderzoeksdoel.

Deze twee aspecten komen in de volgende paragrafen aan de orde.

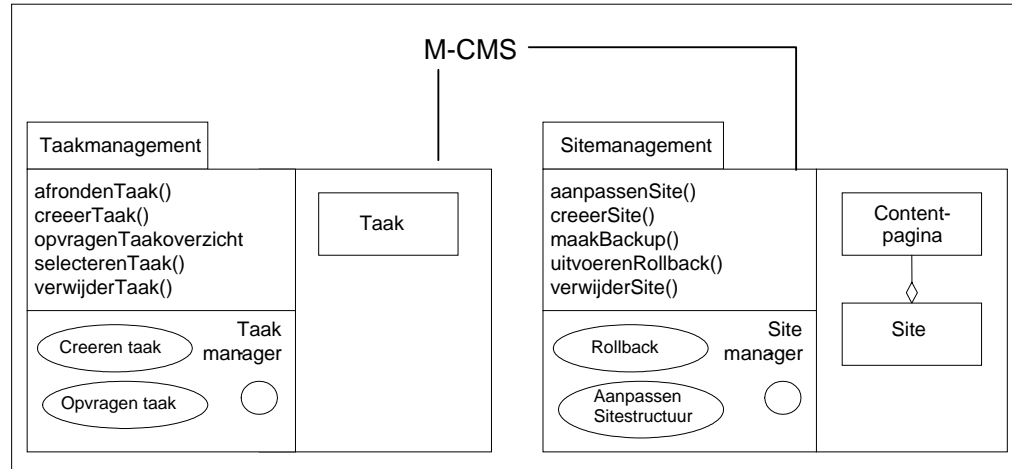
6.2.1 Aansluiting op UML en RUP

De objectgeoriënteerde benadering van IO sluit goed aan bij het modelleren van UML. Dit is ook niet vreemd, want IO is ontwikkeld op basis van de UML Specificatie 1.4. Het onderscheid binnen IO tussen het type- en instantienivo is hetzelfde als dat gehanteerd wordt bij UML op het klasse- en objectnivo. De modellering van het klassediagram of datamodel is daarom vrij snel te realiseren, wat voor een prototype erg prettig is.

De invulling en plaatsing van functies en procedures neemt meer tijd in beslag. Ten eerste is de keuze voor de plaatsing van een functie of procedure niet altijd triviaal; in UML is een keuze mogelijk bij het plaatsen van een functie bij een van de betrokken klassen. Een voorbeeld: een procedure selecterenTaak is gedefinieerd als een procedure waarbij een Gebruiker een Taak selecteert en daarmee zichzelf de Taak toewijst ter uitvoering. De uitvoering van de procedure kan afhankelijk van de ontwikkelaar worden geplaatst onder de klasse Gebruiker of onder de klasse Taak.

IO werkt het gebruik van zogenaamde 'control klassen' in de hand, die gerealiseerd worden door een composiet relatie met de betreffende klasse. Als voorbeeld: een control klasse Taakmanager wordt via een composiet relatie de 'eigenaar' van alle Taken en voert alle procedures en functies uit die op het beheer van de klasse Taak betrekking hebben. Dit voorkomt dat de modellering gekunseld wordt en geeft een component based development (CBD) gerichte implementatie. Uiteindelijk bestaat het systeem uit een aantal control klassen (die gezien kunnen worden als componenten) en een aantal gewone klassen die binnen de control klassen bestaan.

Een component is een modulair, fysiek en vervangbaar deel van het systeem, dat ontwerpklassen samenpakt en interfaces realiseert [Arlow]. In figuur 6.8 zijn twee componenten zijn in de vorm van twee mappen gerepresenteerd door packages naar [Reed]. Een package is een container voor enig deel van het ontwikkelwerk, dat als een eenheid beschouwd dient te worden [d'Souza]. Linksboven in figuur 6.8 zijn de functies en procedures afgebeeld. Linksonder staan de relevante use cases en de interfaces (control klassen worden beschouwd als interfaces in [UML]) en rechts staan de relevante klassen. De opgenomen packages zijn illustratief en zijn niet compleet in functionaliteit en functiedefinitie. In bijlage C zijn van de verschillende packages de complete diagrammen opgenomen, voor zover in het ontwerpmodel uitgewerkt.



Figuur 6.8: Voorbeeld van twee packages van het M-CMS

De wijze waarop IO aansluit op de werkwijze en beheerswijze van de gebruikte methodiek moet worden gezocht in de eerste handelingen binnen de workflows implementatie en testen van het RUP. Prototyping als een specifieke vorm van implementatie is niet expliciet in het RUP gedefinieerd, maar vormen als rapid prototyping [Rosenberg] en extreme programming [XProgramming] worden binnen het RUP wel aangehaald als nuttige vormen van experimentele implementatie.

Prototyping is een versnelde implementatie van (een deel van) de functionaliteiten van het systeem, zodat deze getest kan worden op high-level prestaties. Binnen het M-CMS project is specifiek een structureel en evolutionair prototype gebouwd. Structureel omdat met het opzetten van de contentpijpleiding een specifiek technisch probleem is opgelost en evolutionair omdat het M-CMS kan evolueren naar een volwaardig eindproduct. Bij het prototype wordt overwegend slechts een deel van de use cases geïmplementeerd en zijn use cases niet altijd volledig geïmplementeerd. In het bijzonder is prototyping gericht op het implementeren van de 'main flow of events' van de use cases, daarbij dikwijls alternatieve flows negerend [Rosenberg].

De RUP workflow implementatie is derhalve voor een beperkte set functionaliteiten uitgevoerd en het unit testen slechts tot die aspecten die vooraf zijn aangegeven als hoogste prioriteit. In het geval van het M-CMS zijn op basis van de onderzoeksdoelen de volgende componenten in het prototype verwerkt:

- Opzet van het taak- en gebruikersmanagement, teneinde aspecten van het workflowmanagement te implementeren en te kunnen toetsen;
- Opzet van de contentengine en contentaanvraag, die de realisatie van de contentpijpleiding bewerkstelligen;
- Opzet van het site-, template- en contentmanagement, voor zover dit betrekking heeft op de toetsing van de werking van de contentpijpleiding.

Het bovenstaande sluit goed aan op de implementatie workflow van het RUP zoals beschreven in paragraaf 5.3. Binnen het prototyping zijn de RUP

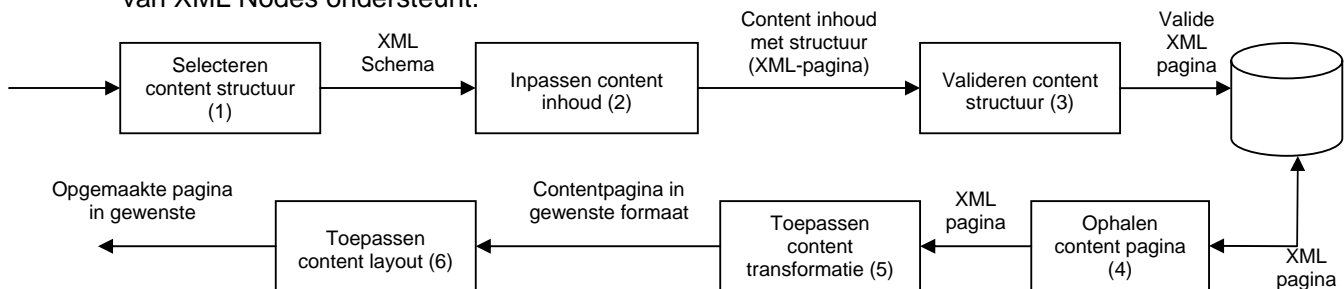
doelstellingen van de implementatie workflow zoals in paragraaf 5.3 beschreven werden bewerkstelligd, in het bijzonder het systeem opdelen in subsystemen, het implementeren van klassen en objecten in termen van componenten en het testen van componenten als eenheden.

6.2.2 Geschiktheid voor internettechnieken

Wanneer gekeken wordt naar de mate waarin IO aansluit op de internettechnieken die bij het M-CMS project van toepassing zijn, dan zijn de volgende aspecten van belang:

- De XML structuur van de content (inclusief XMLSchema en DTD);
- De toepassing van de XSL(-T) standaarden;
- De HTTP request van de apparaten.

Zoals in paragraaf 3.2.1 en werd aangegeven, ziet het proces voor de vervaardiging van de content er uit zoals hieronder weergegeven in de herhaalde figuur 3.4. In de content pijpleiding van paragraaf 3.3 werd getoond dat de onderdelen tot en met de creatie van een valide XML contentpagina binnen IO kunnen worden uitgevoerd omdat IO het genereren en manipuleren van XML Nodes ondersteunt.



Herhaald figuur 3.6: Schematisch proces van content vervaardiging

De onderdelen die met behulp van XSL(-T) worden uitgevoerd, het toepassen van een stylesheet voor de layout en de transformatie naar het geschikte formaat, kunnen niet binnen IO worden uitgevoerd. Hiervoor is een interface benodigd tussen de apparaten die de content behoeven en het IO M-CMS. In paragraaf 3.3 werd de implementatie van een ISAPI beschreven die een interface functie vervult bij:

- De ontvangst van de HTTP request en het subsekwente aanroepen van de procedure afhandelenContentaanvraag binnen het M-CMS;
- Het ontvangen van de XML Node met de XML contentpagina en het toepassen van de XSL stylesheet en XSL-T transformatie;
- Het leveren van de op maat gemaakte contentpagina met een HTTP header aan het betreffende apparaat.

Het gebruiken van een server die contentaanvragen in de vorm van een HTTP request ontvangt en met achterliggende (contentmanagement) systemen informatie uitwisselt is een de facto oplossing. Binnen de contentpijpleiding is dit middels een ISAPI gerealiseerd. Het verdelen van verschillende activiteiten tussen de ISAPI en het IO M-CMS levert ook extra mogelijkheden in de vorm van versnellingsmechanismen zoals het gebruik van de cache, het aanbieden van de contentaanvraag als XMLNode aan IO en het zo vroegtijdig mogelijk afvangen van fouten binnen de pijpleiding.

IO is uitermate geschikt gebleken om:

- De contentstructuur vast te leggen op basis van XMLSchema en DTD templates;
- Content in de vorm van XML-pagina's in de kennisbank op te slaan;
- Dynamisch de opgeslagen content te genereren, zodat geparametriseerde en interactieve content binnen het systeem kan worden gebruikt;
- XML nodes uit te lezen (aangeboden contentaanvraag) en te manipuleren (organisatorische contentselectie door node eliminatie).

6.3 Evaluatie IO

In deze paragraaf zal de ontwikkeltool IO worden geëvalueerd op basis van de ondersteuning voor internettechnieken, de aansluiting op de modelleerwijze UML en werkwijze van het RUP en op basis van enkele gebruikskenmerken. De eerste twee categorieën kwamen in de voorgaande paragraaf aan bod. De algemene gebruikskenmerken richten zich op persoonlijke ervaringen met de tool als iemand die niet bekend is met het werken met de IO ontwikkeltool, in het bijzonder als ontwikkelaar. In het volgende zullen eerst de algemene gebruikskenmerken worden toegelicht, waarna een evaluatie van al het voorgaande in een tabel zal worden samengevat.

In zijn algemeenheid is het werken met de IO ontwikkeltool prettig gebleken. Het meest problematische punt was het ontbreken van een gebruikershandleiding voor de IO-editor, die het zelfstandig aan de slag gaan en zelfstandig blijven werken met de tool bemoeilijkte. Nu wordt IO niet vaak door anderen dan de toolbouwers gebruikt om een (kennis)systeem te implementeren. De bouwers kennen de tool natuurlijk door en door en de eindgebruikers zijn tot dusver voornamelijk gebruikers van in IO gebouwde kennissystemen en gebruiken slechts sporadisch de editor om systematische wijzigingen door te voeren binnen het kennissysteem. Dit in eigen beheer aanpassen van het kennissysteem neemt evenwel steeds meer toe, waardoor het schrijven van een handleiding een nuttige zo niet noodzakelijke toevoeging is.

Daarentegen is de ondersteuning bij problemen voorbeeldig geweest zowel in directe zin als ten aanzien van het verder ontwikkelen van de tool om meer zelfstandig met de IO-editor te kunnen omgaan. Zo werd het foutmeldingsmechanisme verbeterd door de positie van de fout aan te wijzen, werden in de kernel XMLNode functies toegevoegd om XML te kunnen manipuleren en werden allerhande kleine hulpmiddelen aangebracht of verbeterd zoals pull down lijsten met de functies of entiteiten die op een specifieke punt ter beschikking staan.

Ten aanzien van de modelleermogelijkheden kan worden gesteld dat het type- en instantienivo prettig werkt en dat dit een groot aantal hulpmiddelen bevat om makkelijker te kunnen implementeren. De OCL expressietaal vereiste hierbij wat oefening, maar bleek na verloop van tijd goed toepasbaar. Een vertragende factor is wel het tijdconsumerende creëren van instanties met behulp van OCL, waarbij voor iedere functie of object een nieuwe OCL-expressie moet worden ingevoerd binnen de juiste context. Dit kan binnen een volledig implementatieproject opgevangen worden door tijdig een

expertinterface te vervaardigen die de creatie van objecten en het uitvoeren van functies ondersteunt. Binnen het prototyping traject was hiervoor evenwel geen tijd beschikbaar.

Een aantal hulpmiddelen verdient nog specifieke vermelding. Het kunnen aanroepen van een tracer, die stap voor stap de uitvoer van procedures volgt, is een bijzonder nuttige ondersteuning bij het detecteren van structurele fouten, in aanvulling op de melding van semantische en syntactische fouten binnen de editor. Daarnaast levert de overzichtelijke structuur van het typenivo in combinatie met de objectgeoriënteerde en modulaire inrichting van het kennissysteem een goede ondersteuning voor het unit testen van individuele operaties of van packages.

Kenmerk	Beschrijving	Beoordeling	Korte toelichting
1. Internet technieken			
a. XML	Ondersteuning voor de XML-standaard	++	XMLNodes lezen en manipuleren
b. XSL(T)	Ondersteuning voor de XSL(T) standaard	+	Niet in directe zin, maar technieken beschikbaar om dit te realiseren
c. Andere formaten	Ondersteuning voor andere formaten	+	
2. UML en RUP			
a. UML	Aansluiting op modelleerwijze van UML	++	1-op-1 overzetten OO-modellering
b. RUP	Aansluiting op RUP werkwijze	+	Stimuleert CBD door controlklassen
3. Algemeen gebruik			
a. Modelleer mogelijkheden	De mogelijkheden die bij het ontwikkelen aanwezig zijn om de modellering te realiseren	+	IO editor verdiende nog verbetering op bepaalde punten
b. Handleiding	De mate waarin men zelfstandig kan werken op basis van een handleiding	-	Geen gebruikers-handleiding beschikbaar
c. Begeleiding	De ondersteuning in die gevallen waarin de tool problemen geeft	++	Goede en directe ondersteuning

Tabel 6.1: Evaluatie IO

In tabel 6.1 wordt aan de kenmerken een globaal waarde oordeel toegekend op een schaal van zeer goed (++) tot zeer slecht (--) met een kernachtige toelichting. Ondanks het feit dat XSL(-T) en andere formaten niet direct door IO worden ondersteunt, zijn er voldoende mechanismen voorhanden die een relatief eenvoudig hanteren van andere formaten mogelijk maakt. Binnen dit project was dat de ISAPI, waardoor de beoordeling van de ondersteuning hiervoor positief is uitgevallen.

7 Conclusies en aanbevelingen

In de eerste paragraaf zullen op basis van de gestelde onderzoeksdoelen uit paragraaf 1.2 op een drietal gebieden de conclusies worden gepresenteerd. In de tweede paragraaf worden vervolgens enkele aanbevelingen gedaan.

7.1 Conclusies

In deze paragraaf worden op basis van de onderzoeksdoelen achtereenvolgens de conclusies gepresenteerd op het gebied van de geïmplementeerde contentstrategie, het workflowmanagement en tenslotte ten aanzien van de gehanteerde ontwikkelmethode RUP en de ontwerptool IO.

7.1.1 Conclusies contentstrategie en templatemanagement

In hoofdstuk 3 is de gehanteerde contentstrategie behandeld en de realisatie ervan binnen de contentpijpleiding. De contentstrategie is een specifieke combinatie van contentcreatie en –aanpassing, die zijn geïntegreerd in de contentpijpleiding. De contentcreatie betreft de dynamische generatie van XML-contentpagina's en aansluitend transformatie naar het juiste contentformaat. Hierbinnen vindt een globale contentaanpassing plaats door een organisatorische en technische contentselectie.

In de contentpijpleiding is een praktisch werkbare strategie gevonden die door het gebruik van XML als structuurtaal en XSL(-T) als taal voor layout en formaattransformatie een adequate scheiding mogelijk maakt tussen structuur, inhoud, layout en formaat van het tekstuele contenttype. Andere contenttypes zijn minder nadrukkelijk aan de orde geweest in de zin van toegepaste mechanismen, behalve dat ze geleverd worden indien in het juiste formaat aanwezig. Met XML is op basis van gerelateerde standaarden het templatemanagement ingericht. Hiertoe is XMLSchema gebruikt om basissjablonen van paginastructuren vorm te geven en kan het template worden gebruikt om de XML te valideren.

Met de contentpijpleiding is een multi-channel implementatie gerealiseerd, die op basis van het raamwerk van prestatie-indicatoren goede resultaten levert op het gebied van de bruikbaarheid en leveringssnelheid van de content. Wat betreft de volledigheid van de contentlevering zijn concessies gedaan ten faveure van de bruikbaarheid, hoewel voortdurend getracht wordt om de afbeelding bij de client zo compleet mogelijk en op maat aan te bieden.

7.1.2 Conclusies workflowmanagement

Voor de ontwikkeling van een generieke en flexibele workflowondersteuning is het hanteren van een taakgeoriënteerde benadering door implementatie van het taakmanagement zeer geschikt gebleken. Door binnen het M-CMS ondersteuning te bieden voor alle fasen van de workflowlevenscyclus, is het mogelijk om zowel een generieke procesinrichting en ondersteuning voor het procesmanagement neer te zetten als een flexibele opzet te garanderen door het mogelijk maken van procesherinrichting.

Het flexibele taakmanagement geeft voor het publicatieproces voldoende mogelijkheden om de procesinrichting van het workflowmanagement te verzorgen. De procesgang en actorgroepen met hun verantwoordelijkheden kunnen middels de systeeminstellingen worden gedefinieerd. De mogelijkheden beperken evenwel niet tot de inrichting, want middels prestatie-indicatoren, signalerings- en escalatiemogelijkheden wordt het procesmanagement en de procesbijsturing van de workflow ondersteunt. De prestatie-indicatoren van de procesinformatie of afgeleide parameters zijn tevens een goed uitgangspunt voor de procesevaluatie van de gehele workflow. Er moet gezegd dat dit laatste aspecten slechts op globaal niveau is belicht omdat de precieze invulling sterk afhankelijk is van de eigenschappen van de organisatie in kwestie.

7.1.3 Conclusies ontwikkelmethode RUP en ontwikkeltool IO

Het Rational Unified Process (RUP) is een logische, inzichtelijke en werkbare ontwikkelmethode gebleken. Het iteratief karakter van het RUP geeft goede mogelijkheden om tussen- en eindproducten stapsgewijs te ontwikkelen en aan te passen, hetgeen met name mogelijkheden biedt om eerdere tussenproducten op basis van voortschrijdend inzicht van opdrachtgever of ontwikkelaar aan veranderende eisen aan te passen. Dit in combinatie met het incrementeel karakter van het RUP, dat het overzicht en indelen naar componenten volgens een CBD methode ondersteunt, biedt het RUP goede mogelijkheden om ontwikkelprojecten beheersbaar te houden.

Wanneer de tussen- en eindproducten van het RUP adequaat worden geselecteerd, is de bruikbaarheid en volledigheid per definitie voldoende. Voor kleine projecten met een beperkte bezetting moet binnen het zeer omvangrijke RUP nauwkeurig een selectie gemaakt worden van ter zake doende activiteiten en producten. Dit is dan ook het zwakke punt van het RUP voor kleine projecten, waar een tekort aan ervaring met de methode een behoorlijke overhead kan veroorzaken en tot een uitvoering van ineffectieve activiteiten en artefacten kan leiden.

Het gebruik van de modelleringswijze UML binnen RUP resulteert in een vergaande mate van standaardisatie en compatibiliteit met programmeertalen en ondersteuningsmiddelen. Dit bleek tevens bij het prototypen in de ontwerptool IO, die in het bijzonder gebaseerd is op de UML Specificatie en daarmee zeer goed aansluit op de UML modellering. IO bleek tevens geschikt voor het toepassen bij internettechnieken, omdat de structuurtaal XML direct wordt ondersteund en het M-CMS met behulp van de ISAPI goed te ondersteunen bleek voor compatibele formaten en standaarden.

7.2 Aanbevelingen

Tijdens het afstudeerproject is het prototype neergezet van het M-CMS, dat een beperkte functionaliteit kent. In deze vorm is het M-CMS een aanzet tot een basis CMS-product dat voor specifieke klanten op maat gemaakt kan worden, maar waar nog veel ontwikkelwerk verricht zal moeten worden om een volwaardig product neer te zetten.

In het bijzonder is de contentpijpleiding een goede benadering om het hoofd te bieden aan de multi-channel problematiek, maar aangezien er alleen prototyping tests hebben plaatsgevonden, is het aanbevelenswaardig om een

volwaardige 'live' testcasus te ontwikkelen die de contentpijpleiding vergaand kan testen op een groot aantal prestatie-indicatoren uit het beschreven raamwerk. Eenzelfde overweging geldt, en misschien nog wel sterker, voor het workflowmanagement dat nu een brede ondersteuning lijkt te bieden voor de gehele workflowlevenscyclus, maar een serieuze casus behoeft om dit te toetsen.

De nadruk binnen het prototype heeft vooral gelegen op het contenttype tekst, waarbij slechts zijdelings aandacht is geschonken aan contenttypes zoals image, video en audio. Voor dit moment is de keuze gemaakt voor het elimineren van formaten en standaarden die niet worden ondersteunt. Deze contenttypes kunnen in een volgende beschouwing worden meegenomen, hoewel ze andere vereisten hebben dan het teksttype. Het onderscheid tussen de structuur, inhoud, formaat en layout van deze typen is wat minder scherp te leggen en ze zijn niet op structurele wijze (zoals met XML) manipuleerbaar. Een mogelijkheid om ook bij deze contenttypes toch de bruikbaarheid van de content te waarborgen is om een vorm van contentaanpassing toe te passen (bijvoorbeeld bij images) door:

- Een organisatorische reductie: de omvang van de content (image) te reduceren tot een variant met een minder hoge resolutie voor kleinere apparaten;
- Een technische conversie: een conversie naar ander formaat laten plaatsvinden indien het betreffende formaat niet wordt ondersteunt.

Hierbij zou met name de contentaanpassing uitgebreid kunnen worden door niet-ondersteunde standaarden te converteren naar standaarden die wel als MIME-type zijn meegegeven. Dit is met name interessant wanneer het karakter van de content minder tekstgericht is maar de nadruk legt op andere contenttypes zoals image en video omdat deze handelingen mogelijk sterke gevolgen kunnen hebben voor de leveringssnelheid.

Literatuurverwijzingen

De literatuurverwijzingen zijn onderverdeeld in verwijzingen naar boeken en rapporten respectievelijk verwijzingen naar websites. Verwijzingen naar boeken zijn in het document opgenomen als [auteur], website verwijzingen schuin gedrukt: [referentie].

Literatuurverwijzingen

- [Allen] Allen, P. en S. Frost, *Component-based development for enterprise systems*. Cambridge University Press, 1998.
- [Arlow] Arlow, J. en I. Neustadt, *UML and the Unified Process, practical object-oriented analysis and design*. Addison-Wesley, 2002.
- [vdBerg] Berg, A. van den en P. Pottjewijd, *Workflow, continue verbetering door integraal procesmanagement*, Schoonhoven Academic Service Informatica, 1997.
- [Boumphrey] Boumphrey, F. et al, *Beginning XHTML*. Birmingham UK, Wrox Press, 2000.
- [Connalen] Connalen, J., *Building web applications with UML*. Reading, Addison-Wesley, 1999.
- [Dietz] Dietz, J.L.G., *Introductie tot DEMO, van informatietechnologie naar organisatietechnologie*, Samson Bedrijfsinformatica 1996, Alphen aan den Rijn / Zaventem.
- [Jacobson] Jacobson, I. et al, *The Unified Software Development Process*. Addison-Wesley, 1999.
- [Koop] Koop, H.J. et al, *Erfolgsfaktor contentmanagement, vom Web content bis zum Knowledge management*, Braunschweig / Wiesbaden, Vieweg, 2001.
- [Kruchten96] Kruchten, P., 'A rational development process' in Crosstalk 9 (7) juli 1996, p.11-16.
- [Kruchten99] Kruchten, P., *The Rational Unified Process*. Addison-Wesley, 1999.
- [Martin] Martin, T., *Project Cool: guide to XML for web designers*. Wiley & sons, 2001.
- [OCL] Object Management Group, *Object constraint language version 1.6*, january 2003.
- [OMG] Object Management Group, *Unified Modelling Language Specification version 1.5*, maart 2003.

- [Reed] Reed, P., *Developing applications with Java and UML*. Addison-Wesley, 2002.
- [Reijers] Reijers, H.A., *Design and control of workflow processes*, Springer-Verlag, Berlin Heidelberg 2003
- [Rosenberg] Rosenberg, D. en K. Scott, *Use Case driven object modelling with UML, a practical approach*. Reading, Addison-Wesley Longman, 1999.
- [Russell] Russell, S. en P. Norvig. *Artificial Intelligence, a modern approach*, Prentice Hall, New Jersey 1995
- [d'Souza] d'Souza, D.F. en A.C. Wills, *Objects, components and frameworks with UML, the CatalysisSM approach*. Addison Wesley Longman Inc., 1999.
- [Stefik] Stefik, M. *Introduction to knowledge systems*, Morgan Kaufmann Publishers, San Francisco 1995.
- [Wijers] Wijers, G.M. et al, *Analysing the structure of IS Development methods: a framework for understanding*, Reader ontwerpmethodieken, TU Delft 1993.
- [vdWouden] Wouden, F van der, *XML toepassing bij een multi-channel CMS*, verslag onderzoekstaak, faculteit Informatie Technologie en Systemen, TU Delft, 2003

Website verwijzingen

- [DEMO] <http://www.demo.nl>: up-to-date informatie over de DEMO methodologie.
- [HTTPrequest] <http://www.w3.org/Protocols/HTTP/Object-Headers.html>: HTTP header formaat.
- [MIME] <http://www.iana.org/assignments/media-types/>: MIME type lijst.
- [Rational]: <http://www.rational.com/rup/>: voor Rational Unified Process producten en ondersteuningsmiddelen.
- [W3C] <http://www.w3.org>: World Wide Web Consortium home page.
- [W3C-XML]: <http://www.w3.org/xml>: informatie en formele specificaties van XML.
- [W3C-XSLT] <http://www.w3.org/TR/1999/REC-xslt-19991116>: W3C XSL-T versie 1.0
- [XProgramming]: <http://www.xprogramming.com/xpmag/whatisxp.htm>: informatie over extreme programming.

Bijlage A: Acroniemen

A B	API ASP BPR	Application Programming Interface Application Service Provider / Active Server Page Business Process Redesign / Reengineering
C	CBD CD CGI cHTMLcompact HTML CMS CSS	Component Based Design / Development Communicatie Diagram (DEMO) Common Gateway Interface Contentmanagement Systeem Cascading Style Sheets
D E	DCD DDML DEMO DHTML DOM DTD EWI	Document Content Description Document Definition Markup Language (Xschema) Dynamic Essential Modelling of Organisations Dynamic HyperText Markup Language Document Object Model Document Type Definition Elektrotechniek, Wiskunde en Informatica
F G	FTP GPRS GSM GUI	File Transfer Protocol General Packet Radio Services Global System for Mobile Telecommunication Grafische User Interface
H	HTC HTML HTTP	HyperText markup language Components HyperText Markup Language HyperText Transfer Protocol
I	IANA IO ISE ITS	Internet Assigned Numbers Authority Intelligent Objects Information System Engineering Informatie Technologie en Systemen
J K L	JSP k KB	Java Server Pages 1000 KennisBank
M	MathML M-CMS MIME MMA MMS MS ACT	Mathematical Markup Language Multi-channel Contentmanagement Systeem Multipart Internet Mail Extensions Metacontentmanagement Application Multimedia Messaging Service MicroSoft Application Center Test

N O	OO	Object geOriënteerd
	OPERA	Object Protocol Enabling Remote Access
	ORM	Operational Resource Planning
P R	PC	Personal Computer
	PDA	Personal Digital Assistant
	RDF	Resource Description Framework
	RPM	Requirement Management Plan
	RUP	Rational Unified Process
S	SDP	Software Development Plan (RUP)
	SGML	Standard Generalised Markup Language
	SMS	Short Messaging Service
	SMTP	Simple Mail Transfer Protocol
	SRS	Software Requirements Specification (RUP)
T U	UC	Use Case
	UIP	User Interface Prototype
	UMTS	Universal Mobile Telecommunications System
	UML	Unified Modelling Language
	URI	Uniform Resource Identifier
	URL	Uniform Resource Locator
V W	W3C	World Wide Web Consortium
	WAP	Wireless Application Protocol
	WFM	WorkFlowManagement
	WISP	Wireless Internet Service Provider
	WML	Wireless Markup Language
X - Z	XHTML	eXtensible Hypertext Markup Language
	XML	eXtensible Markup Language
	XSL(T)	eXtensible Stylesheet Language (Transformations)
	XQL	eXtensible Query Language

Bijlage B: Begrippenlijst

- API** Application Programming Interface: een verzameling routines, protocollen en hulpmiddelen voor het bouwen van software applicaties
- Artefact (RUP)** Een door een projectmedewerker vervaardigd RUP tussen- of eindproduct
- Bedrijfsinformatie** (management)informatie met betrekking tot onder meer de bedrijfsprocessen en organisatiedoelstellingen
- Besturingscyclus (wfm)** De cyclus die wordt gevormd door het managen van processen en het bijsturen van processen
- Componenten** Herbruikbare systeemdelen die onderling via interfaces communiceren. Componenten kunnen worden gepresenteerd als packages
- Content** Informatie, die zich kenmerkt door inhoud, structuur, layout, mediaformaat en dragermedium
- Contenttype** Onderdeel van het MIME-type, dat een application, audio, image, message, model, multipart, text, video kan zijn.
- Contentengine** De technische realisatie van de entiteit die zorgt voor de levering van een contentpagina op basis van de contentaanvraag van een user agent.
- Contentmanagement** De regelmatige, doelgerichte en systematische omgang met content om content daadwerkelijk waardevol te laten zijn
- Contentmanagement systeem** Een IT-gebaseerd systeem ter organisatie, beheer en doorvoering van contentmanagement
- Contentpagina** De kleinste eenheid van content die binnen het model van het M-CMS bestaat
- Contentpijpleiding** De (geautomatiseerde) serie van activiteiten die plaats vindt om op basis van een HTTP request een stuk content in het juiste formaat en layout te leveren
- DLL** Dynamic Link Library: een verzameling functies, die kunnen worden aangeroepen wanneer benodigd
- DTD** Document Type Definition; een formele beschrijving van de verzameling regels voor een specifiek type XML-document, in het bijzonder regels over de elementen, hun volgorde en hun attributen
- Escalatie (wfm)** Bijsturing van het workflowproces waarbij buiten de normale workflow procesgang wordt getreden
- Expertsysteem** Een computersysteem waarvan de performance wordt gestuurd door specifieke expertkennis bij probleemoplossen [Stefik]
- Functie** Een operatie die een resultaat teruggeeft
- Herinrichtingscyclus (wfm)** De cyclus die wordt gevormd door het managen, bijsturen, evalueren en herinrichten van processen
- HTTP request** Een contentaanvraag op basis van het Hypertext Transfer Protocol, die door ieder apparaat wordt gehanteerd

- Increment (RUP)** Een tussenresultaat, dat significante toegevoegde waarde heeft voor het project, dat op zichzelf staat, opereert op basis van een goed gedefinieerde interface of dat als subcomponent deel kan uitmaken van het eindproduct [Reed]
- Instance (wfm)** Een enkele instantie van een opgestart workflowproces
- Iteratief (RUP)** De toepassing van taken op een repetitieve wijze teneinde te werken richting het verbeteren van een tussen- of eindproduct [Reed]
- Kennisbank** Een opslagplaats voor de kennis gebruikt door het systeem: de regels en richtlijnen om het zoeken naar oplossingen te begeleiden
- Kennissysteem** Een computersysteem dat kennis representeert en gebruikt om een taak uit te voeren [Stefik]
- Metacontent** Informatie over content, zoals XML(Schema's) of een DTD
- Methode** Een reguliere en systematische manier om een doel te bereiken; de gedetailleerde, logisch geordende plannen of procedures die gevolgd worden om een taak uit te voeren of doel te bereiken.
- MIME** Multipart Internet Mail Extensions: document typen die in een HTTP Header worden meegegeven, bestaande uit een contenttype (text, image, audio, video e.a.) en een contentsubtype (bijv. text subtypes plain, html, xml, e.a.)
- Object** Een entiteit met een goed gedefinieerde afbakening en identiteit, die een status en gedrag bevat.
- Package** Een container voor ontwikkelingsartefacten, die kan bestaan uit modellen, software of informatie over de structuur van andere (groepen) packages
- Petrinet** Een modelleringstechniek, die zich in het bijzonder goed leent voor het modelleren van gestructureerde workflows.
- Procedure** Een operatie die geen resultaat teruggeeft
- Procesgang (wfm)** De procesactiviteiten, de volgtijdelijkheid en de eventuele afhankelijkheden binnen een (workflow) proces.
- Procesinformatie** Informatie met betrekking tot het workflowproces en de workflow-procesactiviteiten
- Reflectief (IO)** Een systeem dat in 'run time' herzien en uitgebreid kan worden, tevens op klasse niveau
- RMP (RUP)** Requirements Management Plan: een plan voor het beheren van systeemeisen
- Roll back** De geautomatiseerde ondersteuning voor het vervangen van een gepubliceerde ('live') site, bijvoorbeeld in reactie op corrupte kritieke content of onderdelen van de site
- Site** Een samenhangend geheel van contentpagina's
- Sitemanagement** Beheeractiviteiten die betrekking hebben op een volledige (of sitebeheer) site, waaronder het uitvoeren van back ups en de roll back

Starttrigger (wfm) De gebeurtenis die ervoor zorgt dat een keten van workflowactiviteiten wordt opgestart

Templatemangement Het creëren en beheren van templates die de structuur van de content en de vormgeving vastleggen

URI Uniform Resource Identifier: het mechanisme dat gebruikt wordt om resources op het internet te identificeren (een URL is een voorbeeld van een URI)

URL Uniform Resource Locator: unieke string die een resource (document, image, downloadable file e.d.) op het web identificeert, dat verkregen wordt door een samenvoeging van de resource (content pagina) en de host naam (domein)

Use case Een stukje functionaliteit in het systeem dat de gebruiker een waardevol resultaat geeft

User agent Een applicatie die het voor een apparaat (bijv PC, PDA, of WAP-toestel) mogelijk maakt contact te leggen met het internet.

Workflowlevenscyclus De cyclus van het continu bijsturen en herinrichten van workflowprocessen na het initiële inrichten van het proces

Workflowmanagement Het managen van processen waarbinnen een scheiding is aangebracht tussen de uitvoering en de besturing van activiteiten

xHTML Een willekeurige vorm of versie van HTML, bijvoorbeeld DHTML, XHTML of een HTML versie.

Bijlage C: Ontwerp- en implementatiemodel UML

C.1 Stakeholder en user profielen	103
C.1.1 Profiel belanghebbenden	104
C.1.2 Profiel van gebruikers	104
C.2 Use Case model (RUP requirements workflow)	107
C.2.1 Use case diagrammen	107
C.2.2 User Interface Prototype (UIP)	108
C.2.3 Use case specificaties	111
C.2.3.1 Gebruiker	111
C.2.3.2 Redacteur	113
C.2.3.3 Vormgever	114
C.2.3.4 Eindredacteur	116
C.2.3.5 Hoofdredeur	117
C.2.3.6 Contentbeheerder	119
C.2.3.7 Metacontentbeheerder	123
C.2.3.8 Systeembeheerder	124
C.3 Klassemodel (RUP analysis workflow)	126
C.3.1 Klassediagram	126
C.3.2 Associaties uit het klassediagram	127
C.4 Sequence diagrammen (RUP design workflow)	127
C.4 Overige diagrammen (RUP design workflow)	139
C.4.1 Globale systeemarchitectuur	140
C.4.3 Statusdiagrammen	142
C.5 Implementatiemodel	143
C.5.1 Overzicht van de componenten	143
C.5.2 Taakmanagement	144
C.5.3 Contentpijpleiding	144
C.5.4 Sitemanagement	144
C.5.4 Sitemanagement	145
C.5.5 Systeembeheer	146
C.5.6 Templatemanagement	147
C.5.7 Contentmanagement	148

C.1 Stakeholder en userprofielen

In het RUP worden stakeholders (belanghebbende) en users (gebruikers) in een vroegtijdig stadium geïdentificeerd en beschreven. Dit levert de aanzet tot de belanghebbenden- en gebruikersrequests aan het systeem die vertaald kunnen worden naar use cases (functionaliteiten binnen het systeem). Het onderstaande is ontleend aan het visie document zoals dat voor iCure is opgesteld.

Belanghebbenden:

Naam	Beschrijving	Verantwoordelijkheden
iCure directie (intern)	iCure opdrachtgevers voor de ontwikkeling van het M-CMS.	Accordeert projectplan en verzorgt begeleiding en input waar nodig.
iCure consultants (intern)	iCure uitvoerenden die aan het eindproduct op maat moeten maken voor een specifieke klant.	Zullen het systeem op maat maken en installeren bij de klant. Leveren derhalve ook input, al dan niet via directie.
iCure uitvoerende (intern)	iCure medewerker die het project gaat uitvoeren.	Verantwoordelijk voor alle fasen van het project t/m oplevering prototype.
Management doelorganisatie (extern)	Leidinggevende over de organisatie (-onderdeel) die het product gaat gebruiken (incl beheer).	Beslisser voor aanschaf, eindverantwoordelijk voor het goed functioneren van mensen en voor het voorzien in ondersteunings-middelen)

Gebruikers:

Naam	Beschrijving	Verantwoordelijkheden	Stakeholder
Redacteur	Creëert content en structureert deze.	Contentcreatie Content organisatie Content correctie	
Vormgever	Bewerkt content voor publicatie (opmaak).	Content bewerking (zoals in XML koppen definiëren, toepassen templates?).	
Eindredacteur	Accordeert content voor publicatie.	Content accordering; Content publicatie (plaatsing); Controle sites; Bewaking workflow.	(lager) management doelorganisatie.
Hoofdredeacteur	Eindverantwoordelijk voor content life cycle en publicatie workflow.	Metacontent accordering; Beoordeling en aanpassing workflowproces.	(hoger) management doelorganisatie.
Contentbeheerder	Voert beheerstaken uit op content (voor zover niet door het systeem gedaan).	Content correctie (live); Content reductie (archivering, verwijdering).	
Systeembeheerder	Voert beheerstaken uit op systeeminstellingen en contentengine.	Systeembeheer; Technisch beheer contentengine.	
Metacontent-beheerder	Voert beheerstaken uit op metacontent.	Templatenmanagement (wijzigen, uitbreiden templates); Sitemanagement (structuur).	

C.1.1 Profiel belanghebbenden

iCure directie

Vertegenwoordigt	Hans Leliveld, Martijn IJselmuiden
Beschrijving	Oprachtgever voor de ontwikkeling van het M-CMS
Type	Business consultants met technische achtergrond en tenminste globale kennis van ontwerpmethodieken en modelleertechnieken.
Verantwoordelijkheid	Oprachtgever voor M-CMS project.
Succes criteria	Goed product, werkend prototype dat de basis is voor het eindproduct.
Betrokkenheid	Eindverantwoordelijk, op gezette tijden beoordeling en begeleiding.
Deliverables	Participatie door input, feed-back en beoordeling van deliverables (met name in eerste fase en op hoog abstractieniveau).

iCure consultants

Vertegenwoordigt	Hans Leliveld
Beschrijving	Consultants van iCure die het eindproduct op maat gaan maken voor specifieke klantwensen.
Type	Business consultants met technische achtergrond en kennis van alle facetten van het systeem (materiedeskundigen van publicatieproces en multi-channel concept).
Verantwoordelijkheid	Maken het M-CMS eindproduct op maat voor een specifieke klant.
Succes criteria	Accoord met ontwerpbeslissingen en deliverables.
Betrokkenheid	Materiedeskundigheid levert input voor de ontwikkeling van het M-CMS, zij moeten inzicht krijgen en accoord gaan met de ontwerpbeslissingen.
Deliverables	Participatie door input, feed-back en beoordeling van deliverables (op een lager abstractieniveau dan de directie).

Management doelorganisatie

Vertegenwoordigt	Fictieve belanghebbende.
Beschrijving	Management van de fictieve doelorganisatie.
Type	Management, weinig affiniteit met techniek, maar beoordeelt het product op toegevoegde waarde en verbeteringen die het teweeg brengt.
Verantwoordelijkheden	Verantwoordelijk voor een goede performance van zijn (publicatie)afdeling, (mede-)beslissers voor aanschaf van M-CMS.
Succes criteria	Een CMS dat de problemen met site- en contentmanagement en de multi-channel beschikbaarstelling kan oplossen. Hiermee dient de efficiency van de betreffende organisatie te worden verbeterd alsmede multi-channel deployment te worden gerealiseerd.
Betrokkenheid	(Mede-)beslissers voor aanschaf van het M-CMS.

C.1.2 Profiel van gebruikers

Redacteur

Vertegenwoordigt	Fictieve gebruiker
Beschrijving	Redacteur van stukken content, die mogelijk door anderen wordt opgemaakt, geaccordeerd en geplaatst.
Type	Schrijver met enige omgangservaring met computersystemen.
Verantwoordelijkheid	Stelt content op (tekstueel en basisstructuur zoals alinea's en kopjes) zonder uitvoerige opmaak. Levert deze content met enkele parameters (wat voor content, waar plaatsen,...) aan het systeem.
Succes criteria	Gemak waarmee content kan worden meegegeven en aan het systeem voor verdere behandeling binnen de publicatie workflow. Flexibiliteit waarmee content kan worden aangeleverd.

Vormgever

Vertegenwoordigt	Fictieve gebruiker.
Beschrijving	Verzorgt de opmaak van stukken content op platte tekst aangeleverd door de redacteur. Mogelijk in de vorm van XML-tags toevoegen op basis van een bestaand DTD.
Type	Niet noodzakelijk een creatieve vormgever, als wel een ervaren tekstverwerker met de nodige ervaring met computerprogramma's.
Verantwoordelijkheid	Neemt aangeboden content voor opmaak in behandeling en biedt deze na behandeling aan de eindredacteur aan.
Succes criteria	Gemak van het verkrijgen en doorgeven van de content. Ondersteuning voor opmaakhandelingen (niet bewerkelijk).
Betrokkenheid	Mogelijk in het proces van op maat maken tav templates / DTD.

Eindredacteur

Vertegenwoordigt	Fictieve gebruiker.
Beschrijving	Management van de redacteurs, accordeert opgemaakte content, waarmee de plaatsing van de content wordt geëffectueerd of wordt indien niet accoord teruggegeven aan de redacteur(s) / vormgever(s).
Type	Vergelijkbaar met de redacteur, mogelijk wat meer ervaring met CMS-achtige producten.
Verantwoordelijkheid	Verzamelen van stukken content ter accordering, teruggeven aan redacteur(s) met commentaar dan wel accordering dmv plaatsing.
Succes criteria	Gebruikersgemak van verzamelen, plaatsen en retourneren van content. Ondersteuning workflow in de vorm van taakmanagement. Management informatie voor procesbewaking.
Betrokkenheid	Mogelijk bij op maat maken van systeem.

Hoofdredacteur

Vertegenwoordigt	Management doelorganisatie (belanghebbende).
Beschrijving	Eindverantwoordelijk voor de content workflow en publicatie.
Type	Manager, waarschijnlijk met redactionele achtergrond en met kennis van multi-channel deployment (hij is degene die het initieert).
Verantwoordelijkheid	Verantwoordelijk voor de prestaties van de gehele publicatie afdeling wat betreft inhoud, presentatie (bereik van de lezers) en distributie (stap naar multi-channel in dit bijzonder) en kosten die hierbij gemaakt worden.
Succes criteria	Lage kosten van het publicatie- en distributieproces, hoge kwaliteit van de inhoud en presentatie van de content (intern redactionele kwestie maar ook ondersteuning van de workflow door het systeem). Vergroten aantal lezers (door groot bereik met verschillende kanalen).
Betrokkenheid	Indirect, is (mede-)beslissers bij de aanschaf van een CMS.
Deliverables	-

Contentbeheerder

Vertegenwoordigt	Fictieve gebruiker.
Beschrijving	Voert beheer uit op de reeds geplaatste content door correctie, archivering en verwijdering, ook wel site – en contentmanagement.
Type	Meer technisch dan redactioneel gericht, ervaring met onderhoud (mogelijk ook constructie) van websites.
Verantwoordelijkheid	Onderhoud van de sites en beheer van de content. Ondersteuning van de multi-channel publicatie (nog verder uitwerken).
Succes criteria	Gemak waarmee onderhoud gepleegd kan worden, dus de mate van ondersteuning door het systeem. Feitelijk moet een deel van de bestaande ‘handmatige’ werkzaamheden worden weggenomen.
Betrokkenheid	-
Deliverables	-

Metacontent beheerder

Vertegenwoordigt	Fictieve gebruiker.
Beschrijving	Voert beheer uit op de metacontent, ook wel templatemanagement.
Type	Meer technisch dan redactioneel gericht, ervaring met onderhoud (mogelijk ook constructie) van templates.
Verantwoordelijkheid	Creatie of onderhoud van templates in DTD vorm of anders.
Succes criteria	Gemak waarmee templates kunnen worden gecreëerd, gewijzigd of verwijderd.
Betrokkenheid	Bij het op maat maken van templates door iCure.
Deliverables	Input voor op maat te maken templates door bestaande oplossing te beschrijven, huidige templates te overleggen.

Systeembeheerder

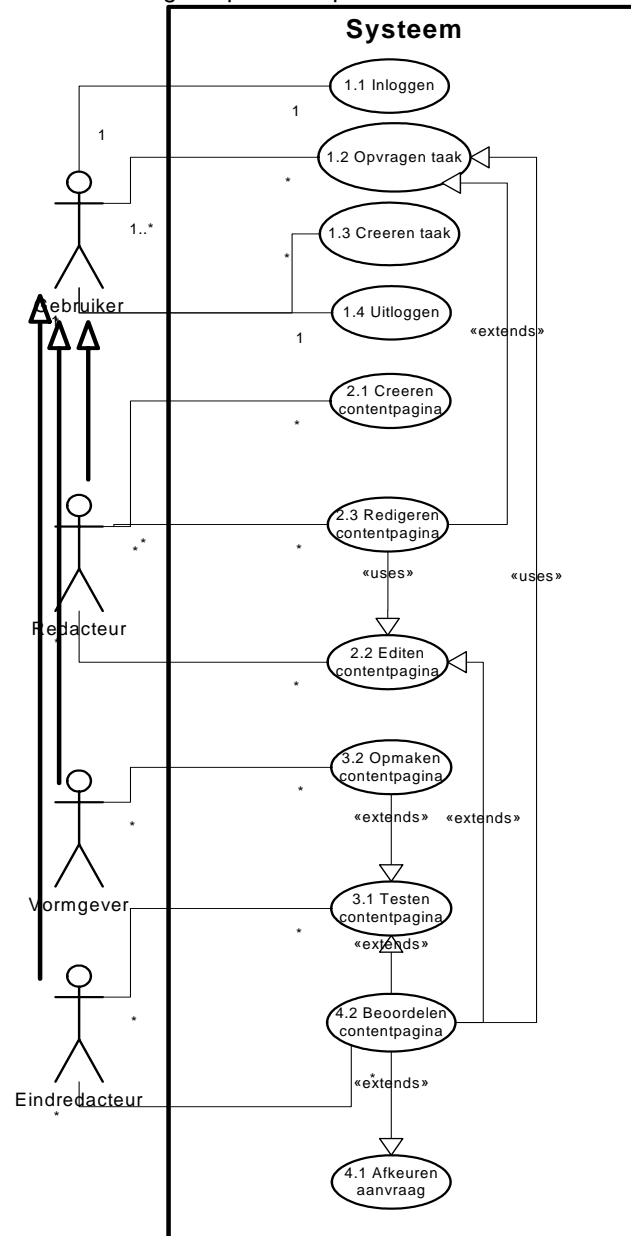
Vertegenwoordigt	Fictieve gebruiker.
Beschrijving	Voert beheer uit op de metacontent. Templatemanagement zagezegd.
Type	Voornamelijk technisch gericht, ervaring met onderhoud van computersystemen en -apparatuur.
Verantwoordelijkheid	Beheer van het CMS en contentengine en waar mogelijk verbeteringen aanbrengen en het beheren van het autorisatiemodel.
Succes criteria	Het presteren van het systeem en de contentengine voor zover binnen de reikwijdte van de beheerder.
Betrokkenheid	Bij het op maat maken van systeemkeuzes door iCure. Eea moet zoveel mogelijk aansluiten op de bestaande manier van werken, zolang dat overeen komt met het CMS-product.
Deliverables	Input voor op maat te maken van systeemkeuzes door bestaande oplossing te beschrijven.

C.2 Use Case model (RUP requirements workflow)

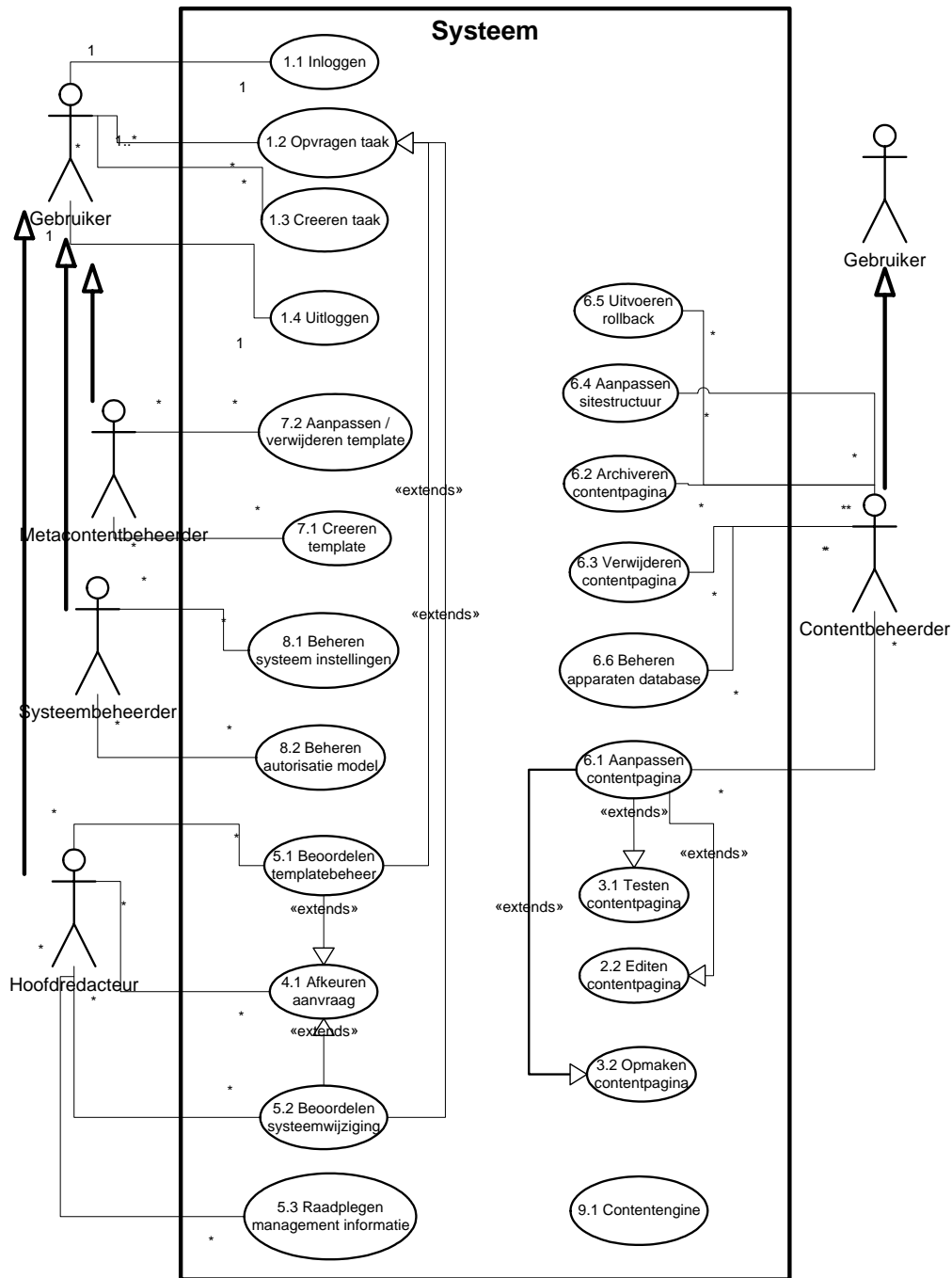
In dit onderdeel wordt het use case model gepresenteerd. In C.1.1 worden de use case diagrammen van het publicatieproces en de beheeractiviteiten gegeven, alsmede het User Interface Prototype omdat daar bij de use case specificatie regelmatig aan wordt gerefereerd. In C.1.2 worden de bijbehorende use case specificaties gegeven.

C.2.1 Use case diagrammen

Use Case diagram publicatieproces:

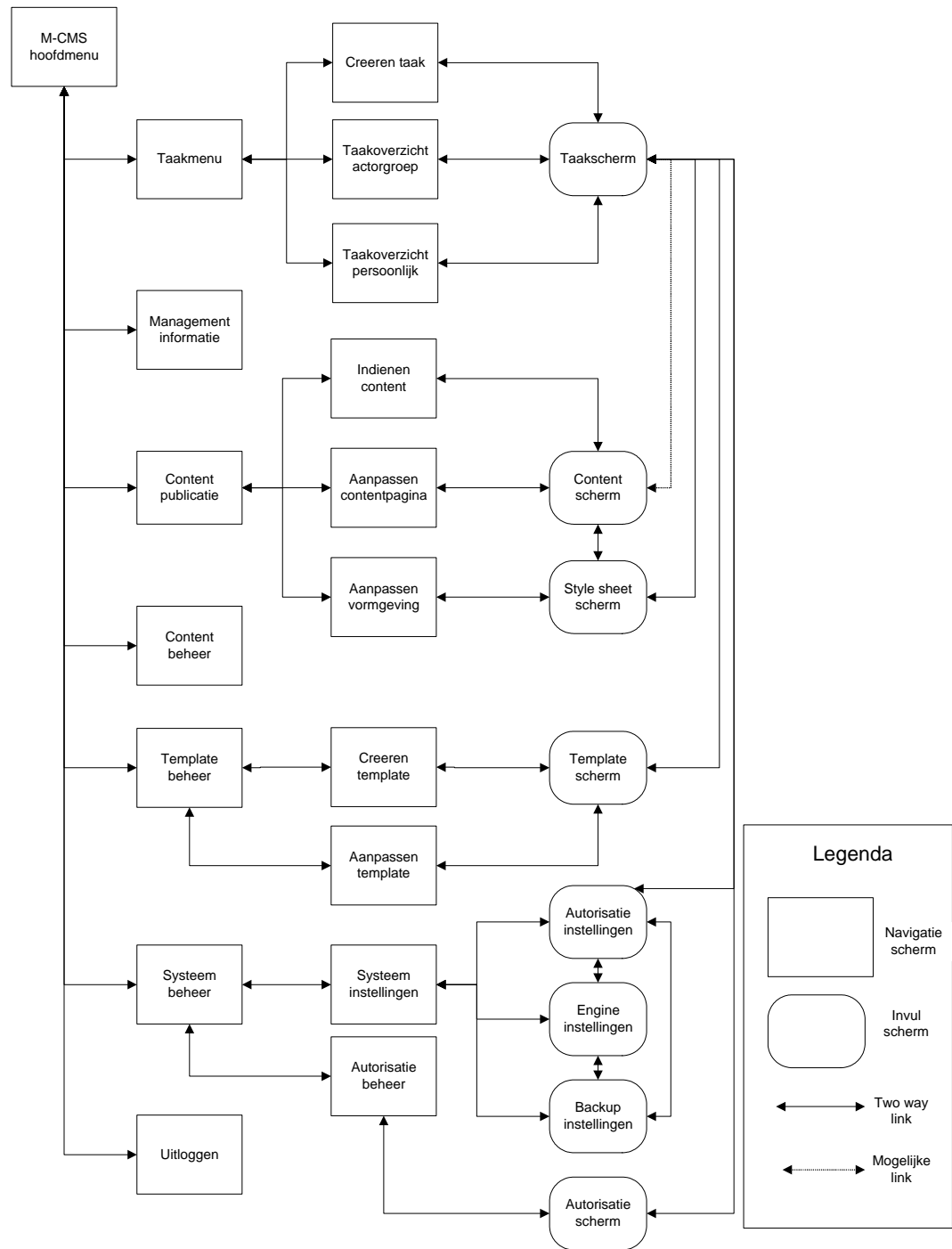


Use case diagram beheeractiviteiten en contentengine:

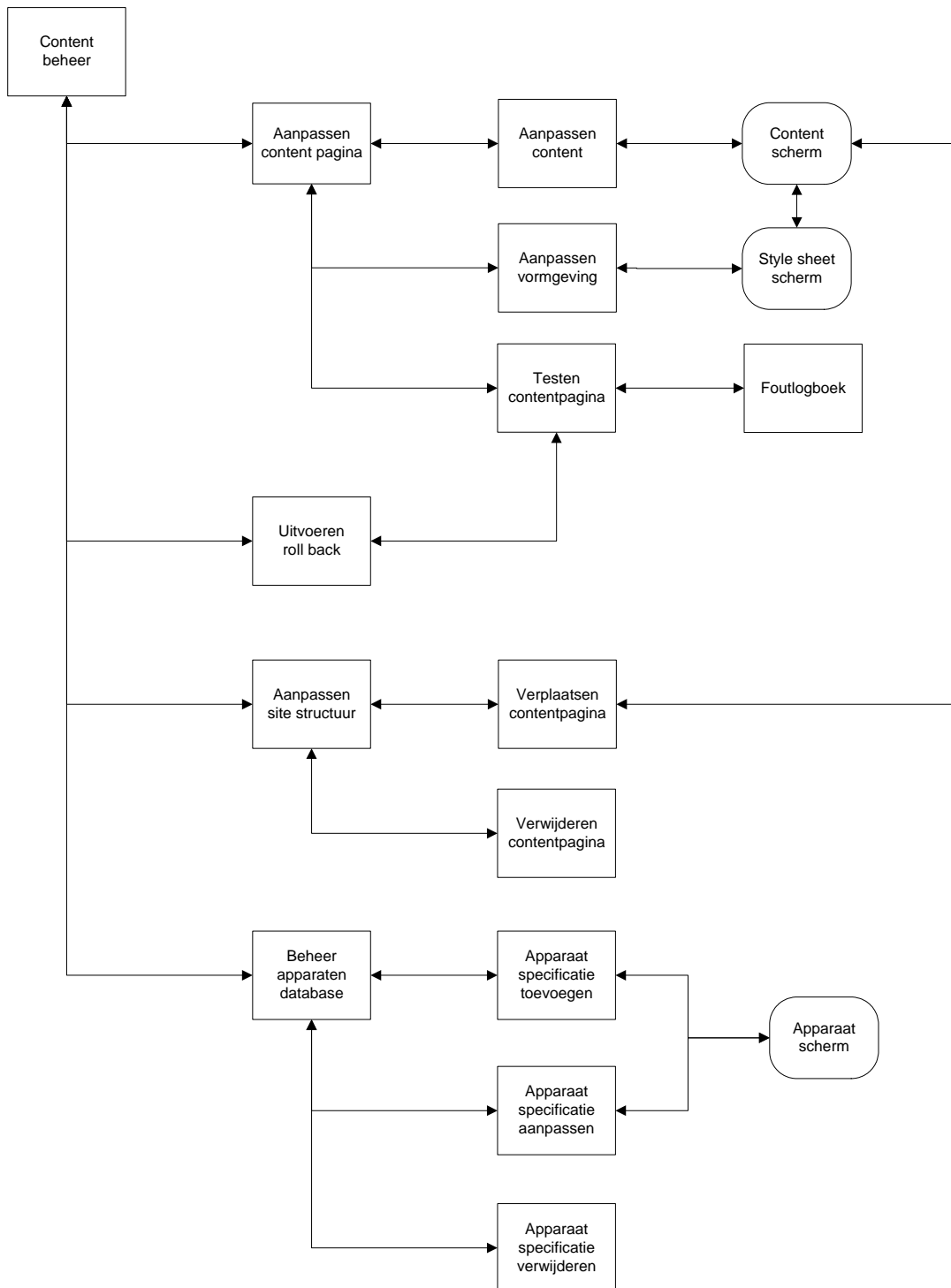


C.2.2 User Interface Prototype (UIP)

Het UIP bestaat uit een hoofdmenu en een menu voor contentbeheer. Dit is het prototype zoals de user interface uiteindelijk gestructureerd kan worden. Hieraan is in de implementatie geen nadere aandacht geschonken. Het hoofdmenu is als volgt gestructureerd:



Het menu voor het contentbeheer is als volgt vormgegeven:



C.2.3 Use case specificaties

In dit onderdeel zijn per actor de betreffende use cases opgenomen in de vorm van een use case specificatie naar [Arlow]. De use case specificatie beschrijft in de vorm van een 'flow of events' de acties die worden uitgevoerd om de use case te realiseren. Tevens zijn pre- en postcondities en eventuele alternatieve flow of events in opgenomen.

C.2.3.1 Gebruiker

Use cases behorend bij actor **gebruiker** (generalisatie, iedere actor kan deze activiteiten uitvoeren):

- 1.1. Inloggen;
- 1.2. Opvragen taak;
- 1.3. Creëren taak;
- 1.4. Uitloggen.

Use case: Inloggen
Use case id: UC 1.1
Actoren: Alle gebruikers van het systeem.
Precondities: De actor is niet ingelogd in het systeem.
Flow of events: <ol style="list-style-type: none"> 1. De actor start het systeem op; 2. Het systeem vraagt de actor om een gebruikersnaam en password; 3. De actor voert gebruikersnaam en password in; 4. Het systeem controleert de gebruikersnaam en password; 5. Gebruikersnaam en password worden correct bevonden; 6. Het systeem brengt de actor naar het hoofdmenu van het systeem. [the use case ends]
Postcondities: De actor is ingelogd in het systeem en de actor bevindt zich in het hoofdmenu.
Alternatieve flow: 5 Zolang gebruikersnaam en password niet accoord zijn: 5.1 Het systeem meldt dat 'gebruikersnaam of password is niet juist'; 5.2 Ga terug naar 2; (optioneel: herhaal 3 keer met dezelfde gebruikersnaam voor te stoppen met use case).
Postcondities: De actor is niet ingelogd in het systeem en mogelijk is de gebruikersnaam geblokkeerd.

Use case: Opvragen taak
Use case id: UC 1.2
Actor: Gebruiker
Precondities: De gebruiker is ingelogd en bevindt zich in het taakmenu.
Flow of events: <ol style="list-style-type: none"> 1. De actor selecteert de optie 'Taakoverzicht persoonlijk' uit het taakmenu; 2. Als het systeem op basis van de gebruikersnaam één of meer taken vindt dan: <ol style="list-style-type: none"> 2.1. Het systeem beeld een overzicht af van de openstaande taken. Hierbij worden parameters als taaksoort, prioriteit, datum en tijdstip van indienen opgenomen in het overzicht; 2.2. De actor selecteert een taak uit de lijst; 2.3. Het systeem beeld een overzicht van de taak [taakformulier] af (taaksoort, prioriteit, (afhandelingsdatum), opdrachtgever, commentaar, documentlink).

[the use case ends]
<p>Alternatief 1:</p> <ol style="list-style-type: none"> 1. De actor selecteert de optie 'Taakoverzicht groep' uit het taakmenu; 2. Het systeem checkt voor welke actorgroepen de gebruiker geautoriseerd is; 3. Als het systeem op basis van de actorgroep(en) één of meer taken vind dan: <ol style="list-style-type: none"> 3.1. Het systeem beeld een overzicht af van de openstaande taken voor de geautoriseerde actorgroepen; 3.2. De actor selecteert een taak uit de lijst; 3.3. Het systeem beeld een overzicht van de taak af (taaksoort, prioriteit, (afhandelingsdatum), opdrachtgever, commentaar, documentlink).
<p>Postcondities: De actor heeft een openstaande taak geselecteerd, welke wordt afgebeeld in het taak scherm.</p>
<p>Alternatief 2: Het systeem heeft geen taak gevonden (gebruiker of actorgroep), maakt hier melding van (pop-up box), de actor selecteert 'ok' en het systeem keert terug naar het taakmenu.</p>
<p>Postcondities: De actor heeft geen openstaande taken en bevindt zich in het taakmenu.</p>
<p>Alternatief 3: De actor kan op ieder moment afbreken middels de optie 'terug naar hoofdmenu'.</p>
<p>Postcondities: De actor bevindt zich in het hoofdmenu.</p>

Use case: Creëren taak
Use case id: UC 1.3
<p>Actoren: Alle gebruikers van het systeem.</p>
<p>Precondities: De actor is ingelogd in het systeem en heeft in het taakmenu de optie 'Creëren taak' geselecteert.</p>
<p>Korte typering: Soms is het nodig om een verzoek tot uitvoer van een taak te doen buiten de workflow; dit kan middels het creëren van een taak geschieden.</p>
<p>Flow of events:</p> <ol style="list-style-type: none"> 1. Het systeem beeld een taak scherm af met invulvelden voor een taak (taaksoort, prioriteit, (afhandelingsdatum), opdrachtgever (systeem vult de actor in), uitvoerende (gebruiker of actorgroep), commentaar, (documentlink) – tussen haken is facultatief). 2. De actor vult de taakgegevens in; 3. De actor selecteert de optie 'taak aanmaken'; 4. Het systeem beeld de gegevens af en vraagt om bevestiging; 5. Na accordering maakt het systeem de taak aan met opslag van datum en tijd van creatie; 6. Het systeem gaat terug naar het taakmenu. <p>[the use case ends]</p>
<p>Alternatieve flow: 5. De actor gaat niet accoord (selecteert 'taak aanpassen'). Het systeem beeld het ingevulde taak scherm af en gaat naar 3.</p>
<p>Postcondities: De actor heeft een taak gecreëerd en bevindt zich in het taakmenu.</p>

Use case: Uitloggen
Use case id: UC 1.4
Actoren: Alle gebruikers van het systeem.
Precondities: De actor is ingelogd en bevindt zich in het hoofdscherm.
Flow of events: 1. De actor selecteert de optie 'uitloggen' in het hoofdscherm. 2. Het systeem logt de actor uit en slaat daarbij persoonlijke gegevens op (welke?). [the use case ends]
Postcondities: De actor is uitgelogd.

C.2.3.2 Redacteur

Use cases behorend bij actor **redacteur**:

- 1.1 Inloggen;
- 1.2 Opvragen taak;
- 1.3 Creëren taak;
- 1.4 Uitloggen;
- 2.1 Creëren contentpagina;
- 2.2 Editen contentpagina;
- 2.3 Opmaken contentpagina.

Use case: Creëren contentpagina
Use case id: UC 2.1
Actoren: Redacteur
Precondities: De actor is ingelogd.
Flow of events: 1. De redacteur selecteert in het hoofdmenu de optie 'Content publicatie'; 2. De redacteur selecteert in het content publicatie-menu de optie 'Creëren contentpagina'; 3. Het systeem beeld een leeg contentscherm af met de invulbare velden: [nader te definiëren]; 4. De actor vult de velden in en selecteert de optie 'Content indienen'; 5. Het systeem slaat deze gegevens op samen met de datum en het tijdstip van creatie; 6. Include (Creëer taak); [met parameters actor en contentlink ingevuld] 7. Het systeem keert terug naar het hoofdmenu. [the use case ends]
Postcondities: Het systeem heeft de gecreëerde content met enkele parameters opgeslagen; De ontvangende actor (vormgever of eindredacteur) heeft een taak toegevoegd gekregen aan de taaklijst.

Use case: Editen contentpagina
Use case id: UC 2.2
Actoren: Contentbeheerder, vormgever of eindredacteur.
Precondities: De actor is ingelogd en heeft een contentpagina geopend.
Korte typering: De actor voert tekstuele aanpassingen aan de content uit en slaat deze op.
Flow of events: <ol style="list-style-type: none"> 1. De actor selecteert de optie 'Editen contentpagina'; 2. De actor verwerkt de aanpassingen handmatig in het contentscherm; 3. De actor selecteert de optie 'Aanpassingen opslaan'; 4. Het systeem vraagt onder welke naam de actor de aanpassingen wil opslaan (als default oude naam overschrijven, maar kan middels een nieuwe naam); 5. De actor geeft een documentnaam in en selecteert 'Opslaan'; 6. Het systeem slaat het document onder de opgegeven naam met de datum en tijd van opslag op; 7. Het systeem keert terug naar het vorige menu. [the use case ends]
Postcondities: De contentpagina is aangepast en opgeslagen in het systeem.

Use case: Redigeren content
Use case id: UC 2.3
Actoren: Redacteur
Precondities: De redacteur is ingelogd in het systeem en bevindt zich in het hoofdmenu. De redacteur heeft een openstaande taak in zijn taaklijst staan.
Korte typering: Aanpassen van de content door de redacteur nadat de eindredacteur deze heeft afgekeurd.
Flow of events: <ol style="list-style-type: none"> 1. Include (Opvragen taak); 2. De actor selecteert de contentpagina uit de taakomschrijving; 3. Include (Editen contentpagina); 4. Include (Creeer taak); [met parameters contentpagina, actor en opdrachtgever ingevuld] 5. De redacteur verwijdert de (redigeer) taak. [the use case ends]
Postcondities: De taak is afgehandeld en verwijderd uit de taaklijst van de redacteur. Er is een nieuwe taak gecreëerd voor de ontvangende actor (vormgever of eindredacteur).
Alternatief: De actor kan op enig moment de aanpassing afbreken. Hij kan hierbij kiezen uit de opties: 'stoppen en niet opslaan' en 'stoppen en opslaan'.
Postcondities: De taak is niet afgehandeld (mogelijk wel een nieuwe versie opgeslagen) en de actor bevindt zich weer in de taaklijst.

C.2.3.3 Vormgever

Use cases behorende bij de actor **vormgever**:

- 1.1 Inloggen;
- 1.2 Opvragen taak;
- 1.3 Creëren taak;
- 1.4 Uitloggen;

- 2.2 Editen contentpagina;
- 3.1 Testen contentpagina;
- 3.2 Opmaken contentpagina.

Use case: Testen contentpagina
Use case id: UC 3.1
Actoren: Vormgever, eindredacteur
Precondities: De actor is ingelogd en heeft een contentpagina in het contentscherm geopend.
Korte typering: Een contentpagina wordt als printvoorbeeld getoond. Hierbij kan de testomgeving fungeren als XML-parser, zodat foutmeldingen worden gegenereerd indien het document niet goed gevormd of niet valide is.
Flow of events: <ol style="list-style-type: none"> 1. De actor selecteert de optie 'Testen content'; 2. Het systeem test middels de XML-parser de contentpagina op XML goed gevormdheid en validiteit; 3. Het systeem laat een voorbeeldweergave zien van de contentpagina in de testomgeving; 4. Het systeem en maakt een foutlogboek aan waarin eventuele foutmeldingen zijn opgenomen; 5. De actor selecteert de optie 'afroonden testen'. [the use case ends]
Postcondities: De actor is terug in de contentpagina, waarvoor een foutlogboek is gecreëerd.

Use case: Opmaken contentpagina
Use case id: UC 3.2
Actoren: Vormgever
Precondities: De vormgever is ingelogd en heeft een openstaande taak in het persoonlijke / vormgevers taakoverzicht staan.
Flow of events: <ol style="list-style-type: none"> 1. Include (Opvragen taak); 2. De vormgever selecteert de te behandelen contentpagina; 3. Het systeem beeld de contentpagina af met achterliggende tabbladen (stylesheet schermen) voor diverse formaten; 4. De vormgever maakt het document op in een (XML?) editor; 5. Indien de vormgever de optie 'Testen contentpagina' selecteert: <ol style="list-style-type: none"> 5.1. Include (Testen contentpagina); [voor onderhavige style sheet] 6. De vormgever selecteert de optie 'Stylesheet opslaan'; 7. Het systeem vraagt onder welke naam het document moet worden opgeslagen; 8. De vormgever geeft een naam en bestandsmap op voor het opslaan; 9. Het systeem slaat de stylesheet op; 10. Include (Creeer taak); [met meegeven van documentlink(s), opdrachtgever=vormgever] 11. Het systeem gaat terug naar het taakscherm van de oorspronkelijke taak; 12. De vormgever selecteert 'Taak verwijderen'; 13. Het systeem verwijdert de taak na bevestiging van de vormgever. [the use case ends]
Alternatief: De vormgever kan voor [9] via de optie 'Editen contentpaginapagina' de contentpagina handmatig aanpassen via Include (Editen contentpaginapagina);
Postcondities: De geselecteerde taak van de vormgever is afgerond en verwijderd; Het opgemaakte document is opgeslagen in de database; Er is een taak gecreëerd voor beoordeling van de contentpagina (incl. Opmaak).

C.2.3.4 Eindredacteur

Use cases behorende bij de actor eindredacteur:

- 1.1 Inloggen;
- 1.2 Opvragen taak;
- 1.3 Creëren taak;
- 1.4 Uitloggen.
- 2.2 Editen contentpagina;
- 3.1 Editen contentpagina;
- 4.1 Afkeuren aanvraag;
- 4.2 Beoordelen contentpagina.

Use case: Afkeuren aanvraag
Use case id: UC 4.1
Actoren: Eindredacteur of hoofdredacteur, (redacteur, vormgever, Metacontentbeheerder)
Precondities: De actor (eindredacteur of hoofdredacteur) is ingelogd en heeft een beoordelingstaak geopend (dit kan zijn voor content-, template-, vormgeving- of een systeemaanpassing). De actor heeft de optie afkeuren geselecteerd;
Flow of events: 1. Include (Creeer taak); [met meegave van parameters opdrachtgever, uitvoerende, taaksoort] 2. Het systeem keert terug naar het taakscherm van de oorspronkelijke taak; 3. De actor selecteert de optie 'Verwijderen taak'; 4. Het systeem verwijdert de taak; 5. Het systeem keert terug naar het taakoverzicht van de actor(groep). [the use case ends]
Postcondities: De taak is verwijderd uit de lijst van de actor en er is een taak gecreëerd voor de indiener van de beoordeling.
Alternatief: De actor kan op enig moment (voor [6]) het afkeuren beëindigen door de optie 'Annuleren afkeuren'. Het systeem keert dan terug naar de taaklijst.
Postcondities: De actor bevindt zich in de taaklijst.

Use case: Beoordelen contentpagina
Use case id: UC 4.2
Actoren: Eindredacteur
Precondities: De eindredacteur is ingelogd en heeft een openstaande taak van de taaksoort beoordelen contentpagina in het individuele of actorgroep taakoverzicht.
Flow of events: 1. Include (Opvragen taak); 2. De eindredacteur selecteert 'Openen document' in het taakscherm; 3. Als de eindredacteur 'Editen contentpaginapagina' selecteert: 3.1. Include (Editen contentpagina); 4. De eindredacteur selecteert een bijbehorende stylesheet; 5. Als de eindredacteur 'Testen contentpagina' selecteert: 5.1. Include (Testen contentpagina); 6. Als de eindredacteur 'Contentpagina goedkeuren' selecteert: 6.1. Het systeem vraagt bevestiging van de goedkeuring; 6.2. De eindredacteur kiest de optie 'content plaatsen' indien accoord; 6.3. Het systeem slaat de content op op de contentserver met vermelding van plaatsingsdatum en -tijd;

<p>6.4. Het systeem keert terug naar het taakscherm; 6.5. De eindredacteur verwijdert zijn beoordelingstaak. [the use case ends]</p>
<p>Postcondities: De geselecteerde taak van de eindredacteur is afgerond en verwijderd;</p>
<p>Alternatieve flow 1: 5. Als de eindredacteur 'Contentpagina afkeuren' selecteert: 5.1. Include (Afkeuren aanvraag); [uitvoerende=redacteur, taaksoort=redigeren contentpagina]</p>
<p>Alternatieve flow 2: 5. Als de eindredacteur 'Vormgeving afkeuren' selecteert: 5.1. Include (Afkeuren aanvraag); [uitvoerende=vormgever, taaksoort=opmaken contentpaginapagina]</p>
<p>Postcondities: De geselecteerde taak van de eindredacteur is afgerond en verwijderd; Een redacteur of vormgever heeft een taak toegevoegd gekregen.</p>

C.2.3.5 Hoofdredacteur

Use cases behorende bij de hoofdredacteur:

- 1.1 Inloggen;
- 1.2 Opvragen taak;
- 1.3 Creëren taak;
- 1.4 Uitloggen;
- 4.1 Afkeuren aanvraag;
- 5.1 Beoordelen template beheer;
- 5.2 Beoordelen systeemwijziging;
- 5.3 Raadplegen management informatie.

<p>Use case: Beoordelen template beheer</p>
<p>Use case id: UC 5.1</p>
<p>Actoren: Hoofdredacteur</p>
<p>Precondities: De hoofdredacteur is ingelogd en heeft een openstaande taak in zijn taakoverzicht.</p>
<p>Flow of events: 1. Include (Opvragen taak); 2. De hoofdredacteur selecteert de templatebeheer aanvraag, die accordering behoeft; 3. Het systeem beeld de aanvraag af in het templatescherm (met wijzigingen highlighted oid?); 4. Als de hoofdredacteur de optie 'Aanvraag goedkeuren' selecteert: 4.1. Het systeem voert de template wijziging uit en slaat deze op; 4.2. De hoofdredacteur selecteert de optie 'Verwijderen taak'; 4.3. Het systeem verwijdert de taak; 4.4. Het systeem keert terug naar het taakoverzicht. [the use case ends]</p>
<p>Alternatieve flow 1: 4. Als de hoofdredacteur de optie 'Aanvraag afkeuren' selecteert: 4.1. Include (Afkeuren aanvraag) [met parameters: uitvoerende=Metacontentbeheerder, opdrachtgever=hoofdredacteur, taaksoort=template beheer]; 4.2. Het systeem keert terug naar het taakoverzicht.</p>
<p>Postcondities: De template aanvraag is afgehandeld en verwijderd.</p>
<p>Alternatieve flow 2: Op enig moment kan de hoofdredacteur de optie 'afbreken beoordelen' selecteren, waarna het systeem terugkeert naar het taakoverzicht.</p>
<p>Postconditie: De hoofdredacteur bevindt zich in het taakoverzicht.</p>

Use case: Beoordelen systeemwijziging
Use case id: UC 5.2
Actoren: Hoofdredacteur
Precondities: De hoofdredacteur is ingelogd en heeft een openstaande taak betreffende een systeemwijziging of autorisatie-aanvraag in zijn taakoverzicht.
Flow of events: 1. Include (Opvragen taak); 2. De hoofdredacteur selecteert de systeemwijziging, die accordering behoeft; 3. Het systeem beeld de aanvraag af in het systeeminstellingen scherm (met voorgestelde wijzigingen highlighted oid?); 4. Als de hoofdredacteur de optie 'Aanvraag goedkeuren' selecteert: 4.1. Het systeem voert de systeemwijziging uit en slaat deze op; 4.5. De hoofdredacteur selecteert de optie 'Verwijderen taak'; 4.2. Het systeem verwijdert de taak; 4.3. Het systeem keert terug naar het taakoverzicht; [the use case ends]
Postcondities: De systeemwijzigingsaanvraag is afgehandeld en de taak verwijderd. De hoofdredacteur bevindt zich in het taakoverzicht.
Postcondities: De hoofdredacteur heeft de taak afgehandeld, deze is uit zijn taaklijst verwijderd en hij bevindt zich in het taakoverzicht. De indiener van de aanvraag heeft in geval van afkeuren een taak in zijn taaklijst toegevoegd gekregen.
Alternatieve flow 1: 5. Als de hoofdredacteur de optie 'Aanvraag afkeuren' selecteert: 5.1. Include (Afkeuren aanvraag) [met parameters: uitvoerende=systeembeheerder, opdrachtgever=hoofdredacteur, taaksoort=systeemwijziging]; 5.2. Het systeem keert terug naar het taakoverzicht.
Alternatieve flow 2: Op enig moment kan de hoofdredacteur de optie 'afbreken beoordelen' selecteren, waarna het systeem terugkeert naar het taakoverzicht.
Postconditie: De hoofdredacteur bevindt zich in het taakoverzicht.

Use case: Raadplegen management informatie
Use case id: UC 5.3
Actoren: Hoofdredacteur
Precondities: De hoofdredacteur is ingelogd.
Korte typering: <i>Management informatie op basis van systeem en workflow prestatie-indicatoren. Aan de hand van de onderzoeksdoelen van het afstuderen kan hier een eerste aanzet toe gegeven worden, maar een uitgebreide management informatietool zal niet aan de orde zijn.</i>
Flow of events: [the use case ends]
Postcondities: De hoofdredacteur bevindt zich in het hoofdmenu.

C.2.3.6 Contentbeheerder

Use cases behorend bij de **contentbeheerder**:

- 1.1 Inloggen;
- 1.2 Opvragen taak;
- 1.3 Creëren taak;
- 1.4 Uitloggen;
- 6.1 Aanpassen contentpagina;
- 6.2 Archiveren contentpagina;
- 6.3 Verwijderen contentpagina;
- 6.4 Aanpassen site structuur;
- 6.5 Uitvoeren roll back;
- 6.6 Beheren apparatendatabase.

Use case: Aanpassen contentpagina
Use case id: UC 6.1
Actoren: Contentbeheerder
Precondities: De contentbeheerder is ingelogd en bevindt zich in het content beheer menu.
Beschrijving: De contentbeheerder wenst een contentpagina (tekstueel of layout) aan te passen die op de site staat. Hij past hiertoe de content aan en maakt deze direct actief op het net, al dan niet na testen.
Flow of events: <ol style="list-style-type: none"> 1. De actor selecteert de optie 'Aanpassen contentpagina'; 2. Het systeem beeld een lijst af met contentpagina's (gelaagd in mappenstructuur); 3. De actor selecteert een contentpagina; 4. Het systeem opent de contentpagina in het contentscherm en achterliggende XSL stylesheets; 5. Include (Editen contentpagina); 6. De actor selecteert een van de achterliggende XSL stylesheets; 7. De actor verzorgt de aanpassingen aan de stylesheet; 8. Als de actor 'Testen pagina' selecteert, dan: <ol style="list-style-type: none"> 8.1. Include (Testen contentpagina); 9. Als de actor 'Publiceer content' selecteert, dan: <ol style="list-style-type: none"> 9.1. Het systeem slaat de contentpagina en stylesheets op in de content database; 10. Het systeem keert terug naar het content beheer menu. <p>} [the use case ends]</p>
Postcondities: De content is aangepast en gepubliceerd.

Use case: Archiveren content pagina
Use case id: UC 6.2
Actoren: Contentbeheerder
Precondities: De contentbeheerder is ingelogd.
Beschrijving: Oude contentpagina moeten van de huidige locatie verplaatst worden naar een contentarchief (uitgaande van live; niet live wordt gevat onder verwijderen). Dit betreft feitelijk het verplaatsen van een content pagina, ofwel het aanpassen van links binnen de structuur van de pagina.
Flow of events: <ol style="list-style-type: none"> 1. De actor selecteert de optie 'archiveren contentpagina'; 2. Het systeem beeld een lijst af met contentpagina's (gelaagd in mappenstructuur); 3. De actor selecteert een contentpagina; 4. De actor selecteert de optie 'archiveren content'; 5. Het systeem vraagt om de locatie van waar de content gearchiveerd moet worden; 6. De actor geeft de locatie op; 7. Het systeem controleert of er zich problemen voordoen bij het verwijderen van de pagina (geen verwijzingsproblemen of andere probs?); <ol style="list-style-type: none"> 7.1. Het systeem maakt melding van de (potentiële) problemen en vraagt of de actor toch wil doorgaan.....(?) 8. Anders verwijdert het systeem links naar de betreffende pagina en verplaatst de content naar de archieflocatie. <p>{}</p> <p>[the use case ends]</p>
Postcondities: De content pagina is verplaatst naar een archieflocatie.

Use case: Verwijderen contentpagina
Use case id: UC 6.3
Actoren: Contentbeheerder
Precondities: De contentbeheerder is ingelogd en heeft een contentpagina geselecteerd die hij wenst te verwijderen.
Beschrijving: Check om te voorkomen dat er idle pagina's of links resulteren bij het uitvoeren van verwijdering.
Flow of events: <ol style="list-style-type: none"> 1. Het systeem maakt een overzicht van alle links naar de te verwijderen contentpagina; 2. Het systeem verwijdert deze links; 3. Het systeem controleert of de links op de te verwijderen pagina uniek zijn {als dit de enige link naar een contentpagina betreft mag deze niet zonder meer worden verwijderd}; 4. Als er unieke links zijn: <ol style="list-style-type: none"> 4.1. Het systeem vermeldt het bestaand van een unieke link en vraagt of deze pagina ook moet worden verwijderd; 4.2. Als de contentbeheerder 'ja' selecteert dan Include(Verwijderen Contentpagina); 5. Het systeem verwijdert de contentpagina. <p>{}</p> <p>[the use case ends]</p>
Postcondities: De geselecteerde content pagina is verwijderd (en mogelijk achterliggende pagina's).

Use case: Aanpassen site structuur
Use case id: UC 6.4
Actoren: Contentbeheerder
Precondities: De contentbeheerder is ingelogd.
Beschrijving: De structuur van de site wordt aangepast doordat contentpagina's worden verwijderd of verplaatst (links moeten worden verwijderd en / of toegevoegd). Uitgangspunt is dat voor browsers de oude URL behouden moet blijven.
Flow of events: <ol style="list-style-type: none"> 1. De actor selecteert de optie 'selecteren contentpagina'; 2. Het systeem beeld een lijst af met contentpagina's; 3. De actor selecteert een contentpagina; 4. Indien de actor 'Verplaatsen contentpagina' selecteert: <ol style="list-style-type: none"> 4.1. De actor vraagt een lijst op met links naar de contentpagina; 4.2. Het systeem genereert een lijst met alle links naar de contentpagina; 4.3. De actor selecteert een of meerdere links die verwijderd moeten worden; 4.4. Het systeem verwijdert de links op de betreffende contentpagina('s); 4.5. Include (Testen contentpagina); 4.6. De actor selecteert 'Opslaan en publiceren contentpagina'; 4.7. Het systeem informeert vanuit welke locatie('s) de contentpagina benadert moet kunnen worden (aanbrengen nieuwe link); 4.8. De actor selecteert een contentpagina, het systeem beeld deze af en de actor voegt een nieuwe link toe. 5. Het systeem slaat alle wijzigingen op en maakt daarmee de content 'live'. <p>[the use case ends]</p>
Alternatieve flow 1: <ol style="list-style-type: none"> 4. Indien de actor 'verwijderen content pagina' selecteert: <ol style="list-style-type: none"> 4.1. Include (Verwijderen contentpagina); 5. Het systeem slaat alle wijzigingen op.
Postcondities: De sitestructuur is aangepast en opgeslagen (live).

Use case: Uitvoeren roll back
Use case id: UC 6.5
Actoren: Contentbeheerder
Precondities: De contentbeheerder is ingelogd en bevindt zich in het contentbeheer menu.
Beschrijving: In geval van nood moet een volledige site vervangen worden door een back up als er ernstige misstanden op of aan de site worden geconstateerd.
Flow of events: <ol style="list-style-type: none"> 1. De actor selecteert 'Uitvoeren roll back'; 2. Het systeem toont een lijst met backup sites; 3. De actor selecteert een backup site; 4. Het systeem toont een lijst van contentpagina's van de site; 5. De actor selecteert een contentpagina; 6. Include 'Testen contentpagina'; 7. De actor selecteert 'Versie acoord': <ol style="list-style-type: none"> 7.1. Het systeem vraagt om bevestiging van de roll back; 7.2. Indien ja, dan maakt het systeem een back-up van de huidige live-site; 7.3. Het systeem vervangt de live site door de roll back-up; 7.4. De hoofdredacteur (of mogelijk alle gebruikers) ontvangt een notificatie van de roll back;

8. Het systeem keert terug naar het contentbeheer menu. [the use case ends]
Alternatieve flow 1: 6. De actor selecteert 'Versie niet accoord': 6.1. Ga terug naar 2;
Alternatieve flow 2: De actor kan op enig moment voor [6.2] de procedure afbreken door 'Afbreken roll back' te selecteren. Het systeem keert terug naar het contentbeheer menu.
Postcondities: De roll back is uitgevoerd of de actor heeft het proces gestopt. De actor bevindt zich in het contentbeheer menu.

Use case: Beheren apparaten database
Use case id: UC 6.6
Actoren: Contentbeheerder
Precondities: De contentbeheerder is ingelogd en bevindt zich in menu 'beheer apparatendatabase'.
Beschrijving: In een apparaten database aanpassingen maken zoals het toevoegen van apparaten met hun beperkingen en uit te voeren oplossingen (in de vorm van executables?) of het aanpassen / verwijderen van bestaande apparaten. Dit dient verder te worden uitgewerkt in de ontwerpfase.
Flow of events: {toevoegen apparaatspecificatie} 1. Als de contentbeheerder 'Apparaat toevoegen' selecteert: 1.1. Het systeem beeld een lijst af met in te vullen velden (apparaatscherm); 1.2. De contentbeheerder vult deze velden in; 1.3. De contentbeheerder selecteert 'gegevens opslaan'; 2. Het systeem verwerkt de gegevens in de apparatendatabase; 3. Het systeem keert terug naar het menu 'beheer apparatendatabase'. [the use case ends]
Alternatief 1: {aanpassen apparaatspecificatie} 1. Als de contentbeheerder 'Apparaat gegevens aanpassen' selecteert: 1.1. Het systeem beeld een lijst af met apparaten; 1.2. De actor selecteert een apparaat; 1.3. Het systeem beeld de wijzbare apparaatgegevens af (apparaatscherm); 1.4. De contentbeheerder past de gegevens aan; 1.5. De contentbeheerder selecteert 'gegevens opslaan'; 2. Het systeem verwerkt de gegevens in de apparatendatabase; 3. Het systeem keert terug naar het menu 'beheer apparatendatabase'.
Alternatief 2: {verwijderen apparaatspecificatie} 1. Als de contentbeheerder 'Apparaat verwijderen' selecteert: 1.1. Het systeem beeld een lijst af met apparaten; 1.2. De actor selecteert een apparaat; 1.3. Het systeem beeld de gegevens af (apparaatscherm); 1.4. De actor selecteert 'Apparaat uit database verwijderen'; 1.5. Het systeem vraagt om een bevestiging; 1.6. Als de actor bevestigd, dan verwijdert het systeem de gegevens uit de apparatendatabase; 2. Het systeem keert terug naar het menu 'beheer apparatendatabase'.
Alternatief 3: De actor kan op enig moment 'afbreken zonder opslaan' selecteren.
Postcondities: De apparaten database is aangepast, de actor bevindt zich in het menu 'beheer apparatendatabase'.

C.2.3.7 Metacontentbeheerder

Use cases behorend bij de Metacontentbeheerder:

- 1.1 Inloggen;
- 1.2 Opvragen taak;
- 1.3 Creëren taak;
- 1.4 Uitloggen;
- 7.1 Creëren template;
- 7.2 Aanpassen / verwijderen template.

Use case: Creëren template
Use case id: UC 7.1
Actoren: Metacontentbeheerder, hoofdredacteur.
Precondities: De Metacontentbeheerder is ingelogd en bevindt zich in het templatebeheer menu.
Beschrijving: Templates zijn de XML-structuren (bijv contentpagina structuur) of het meer generieke DTD dat dient ter validatie van de XML bestanden.
Flow of events: <ol style="list-style-type: none"> 1. De actor selecteert de optie 'Aanmaken template'; 2. Het systeem opent een editor waarin de actor zijn template kan maken (mogelijk openen bestanden indien al begin is gemaakt); 3. De actor selecteert de optie 'Opslaan bestand'; 4. Het systeem vraagt om een bestandsnaam; 5. De actor vult de bestandsnaam in; 6. Het systeem maakt een taak aan ter beoordeling van het template beheer: Include (Creer taak); 7. Als het template beheer wordt goedgekeurd dan: <ol style="list-style-type: none"> 7.1. Het systeem slaat het bestand op. <p>[the use case ends]</p>
Postcondities: Er is een nieuwe template gecreëerd en actief gemaakt.

Use case: Aanpassen / verwijderen template
Use case id: UC 7.2
Actoren: Metacontentbeheerder.
Precondities: De Metacontentbeheerder is ingelogd en bevindt zich in het templatebeheer menu.
Flow of events: <ol style="list-style-type: none"> 1. Het systeem vraagt de actor een bestaand template te selecteren; 2. De actor selecteert een template uit een aangeboden lijst; 3. Indien de actor de optie 'Aanpassen template' selecteert: <ol style="list-style-type: none"> 3.1. Metacontentbeheerder voert aanpassingen door; 3.2. De template wordt opgeslagen; 4. Het systeem maakt een taak aan ter beoordeling van het template beheer: Include (Creer taak); 5. Als het template beheer wordt goedgekeurd dan: <ol style="list-style-type: none"> 5.1. Het systeem slaat het bestand op. <p>[the use case ends]</p>
Alternatief 1: {verwijderen template, rest als main flow} <ol style="list-style-type: none"> 3. Indien de actor de optie 'verwijderen template' selecteert: <ol style="list-style-type: none"> 3.1. Het systeem vraagt om bevestiging van de verwijdering; 3.2. Het systeem verwijdert na bevestiging de template;
Alternatief 2:

Dit kan ook vanuit het taakmenu geschieden; 1 en 2 worden vervangen door opvragen taak en het selecteren van een templatebeheer taak.

Postcondities:

De template is aangepast en opgeslagen.

C.2.3.8 Systeembeheerder

Use cases behorend bij de systeembeheerder:

1.1 Inloggen;

1.2 Opvragen taak;

1.3 Creëren taak;

1.4 Uitloggen;

8.1 Beheren systeeminstellingen;

8.2 Beheren autorisatiemodel.

Use case: **Beheren systeeminstellingen**

Use case id: UC 8.1

Actoren:

Systeembeheerder, hoofdredacteur.

Precondities:

De systeembeheerder is ingelogd en bevindt zich in het menu systeembeheer.

Beschrijving:

Onder het systeembeheer kunnen taken vallen als het instellen van parameters binnen het systeem (backupfrequentie, het definiëren van rollen (beheer van actorgroepen) of het indelen van actorgroep autorisaties)

Flow of events:

1. De systeembeheerder selecteert de optie 'Wijzigen systeeminstellingen';
2. Het systeem biedt een lijst aan van (categorieën) parameters die gewijzigd kunnen worden;
3. De systeembeheerder selecteert parameter (of categorie);
4. Het systeem beeldt de betreffende instellingen af in wijzigbare velden;
5. De systeembeheerder voert de wijzigingen in;
6. De systeembeheerder selecteert de optie 'wijzigingen indienen';
7. Het systeem vraagt om bevestiging;
8. Het systeem creëert een beoordelingstaak voor de hoofdredacteur via Include (Creeer taak);
9. Als de wijziging wordt geaccordeerd worden de wijzigingen opgeslagen.

[the use case ends]

Postcondities:

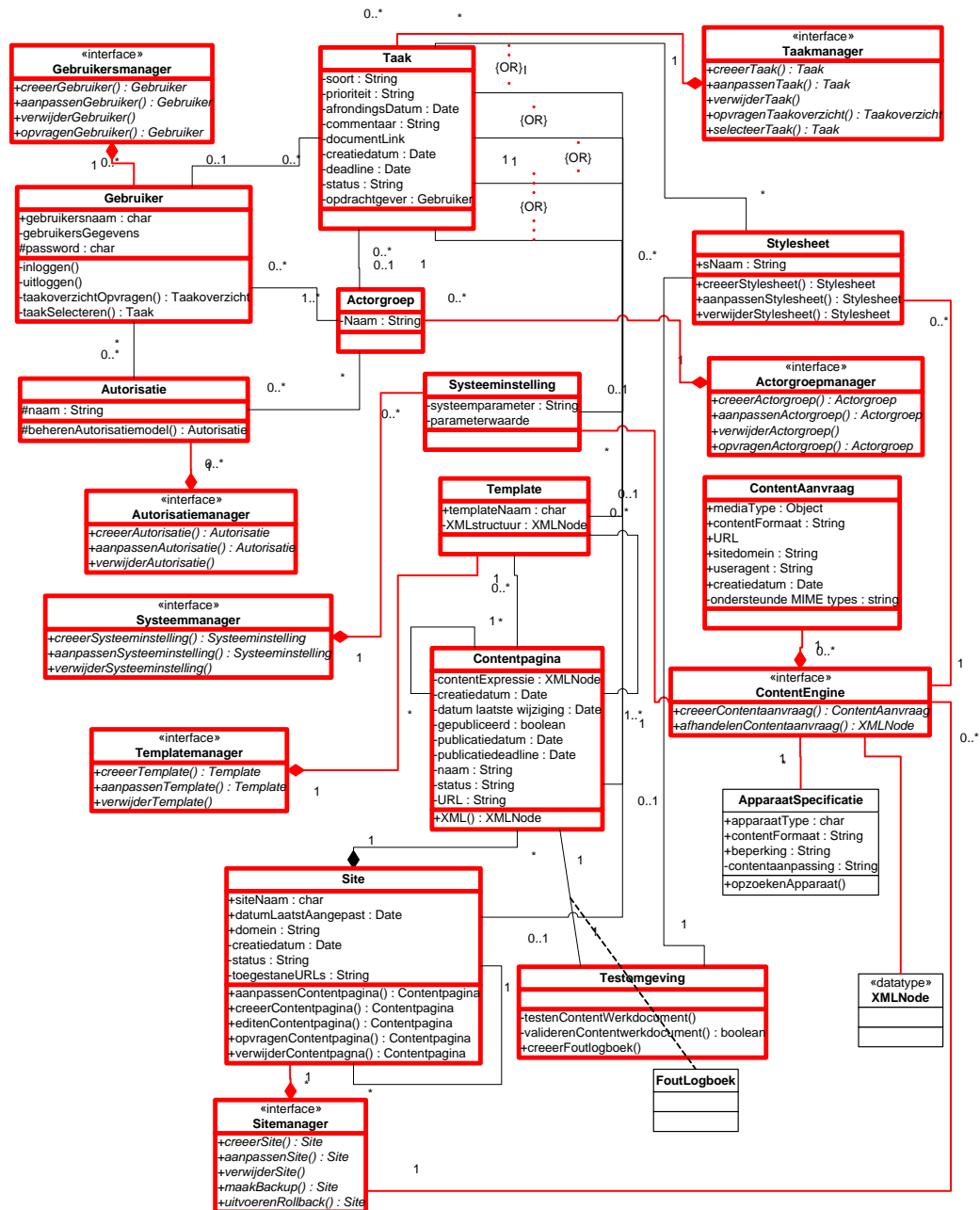
Het systeem heeft de gewijzigde instellingen in de vorm van een accorderingstaak aangeboden aan de hoofdredacteur en de systeembeheerder bevindt zich in het systeembeheer menu.

Use case: Beheren autorisatiemodel
Use case id: UC 8.2
Actoren: Systeembeheerder.
Precondities: De systeembeheerder is ingelogd en bevindt zich in het menu systeembeheer.
Beschrijving: Onder het autorisatie beheer vallen taken als het creëren of verwijderen van gebruikers en het instellen van autorisaties voor individuele actoren of voor actorgroepen. Deze behoeven niet te worden geaccordeerd door de hoofdredacteur.
Flow of events: <ol style="list-style-type: none"> 1. De systeembeheerder selecteert de optie 'Aanpassen autorisatiemodel'; 2. De systeembeheerder selecteert de optie 'Aanpassen bestaande autorisatie': <ol style="list-style-type: none"> 2.1. Het systeem beeld een lijst af met gebruikersnamen; 2.2. De systeembeheerder selecteert een gebruiker; 2.3. Het systeem beeld in het autorisatiescherm aanpasbare gegevens en af te vinken autorisaties af; 2.4. De systeembeheerder verzorgt de aanpassingen; 3. Het systeem slaat de autorisatiewijziging op. <p>[the use case ends]</p>
Alternatieve flow 1: <ol style="list-style-type: none"> 2. De systeembeheerder selecteert de optie 'Toevoegen nieuwe gebruiker': <ol style="list-style-type: none"> 2.1. De systeembeheerder maakt een nieuwe gebruiker aan; 2.2. De systeembeheerder vult een lijst in met in te vullen gegevens en af te vinken autorisaties; <p>Verder als main flow of events.</p>
Alternatieve flow 2: <ol style="list-style-type: none"> 2. De systeembeheerder selecteert de optie 'Verwijderen gebruiker': <ol style="list-style-type: none"> 2.1. Het systeem beeld een lijst af met gebruikersnamen; 2.2. De systeembeheerder selecteert een gebruiker; 2.3. Het systeem vraagt om bevestiging van de verwijdering; <p>Verder als main flow of events.</p>
Postcondities: Het systeem heeft de gewijzigde autorisaties opgeslagen en de systeembeheerder bevindt zich in het systeembeheer menu.

C.3 Klassemodel (RUP analysis workflow)

In de onderstaande figuur is het klassediagram opgenomen. De klassen zijn met verbonden door associaties, die in een tabel hieronder worden uitgewerkt. Klassen hebben een naam (1^e blok) en kunnen attributen (2^e blok) en operaties (3^e blok) hebben. Control klassen, die een composiet relatie hebben met een klasse (de dichte ruiten), zijn hier opgenomen als interfaces.

C.3.1 Klassediagram



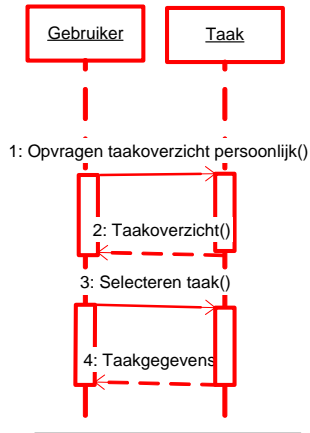
C.3.2 Associaties uit het klassediagram

Klasse (links)	Associatie:	Klasse (rechts)	Multipliciteit
Taak	Is toegekend aan	Actorgroep	[0*:1]
Taak	Is in behandeling door	Gebruiker	[0*:0,1]
Taak	Betreft werk aan	Template OR Systeeminstelling OR Site OR Contentpagina OR Stylesheet	[0*:0,1]
Gebruiker	Behoort tot	Actorgroep	[0*:1*]
Gebruiker	Is specifiek geautoriseerd voor	Autorisatie	[0*:0*]
Actorgroep	Is standaard geautoriseerd voor	Autorisatie	[0*:0*]
Template	Bepaalt structuur van	Contentpagina	[1:0*]
Contentpagina	Wordt op content getest binnen	Testomgeving	[1*,1]
ContentEngine	Creeert vormgeving met	Stylesheet	[1:0*]
Testomgeving	creert Foutlogboek voor	Contentpagina	[1:0,1:0*]
Site	Is backup van	Site	[1*:0*]
ContentEngine	Vraagt vormgeving op van	Stylesheet	[0,1:1*]
ContentEngine	Vraagt apparaataanpassing op van	ApparaatSpecificatie	[1:0*]
Contentpagina	Linken	Contentpagina	[0*,0*]
ContentEngine	Behoeft content van	Sitemanager	[1:1]
Systeeminstelling	Bepaalt werking van	ContentEngine	[0*,1]
Testomgeving	Test op vormgeving van	Stylesheet	[0*:1]
ContentEngine	Handelt af (composiet relatie)	ContentAanvraag	[1:0*]
Systeemmanager	Beheert (composiet relatie)	Systeeminstelling	[1:0*]
Autorisatiemanager	Beheert (composiet relatie)	Autorisatie	[1:0*]
Site	Omvat (composiet relatie)	Contentpagina	[1:0*]
Actorgroepmanager	Beheert (composiet relatie)	Actorgroep	[1:0*]
Gebruikersmanager	Beheert (composiet relatie)	Gebruiker	[1:0*]
Templatemanager	Beheert (composiet relatie)	Template	[1:0*]
Taakmanager	Beheert (composiet relatie)	Taak	[1:0*]
Sitemanager	Beheert (composiet relatie)	Site	[1:0*]

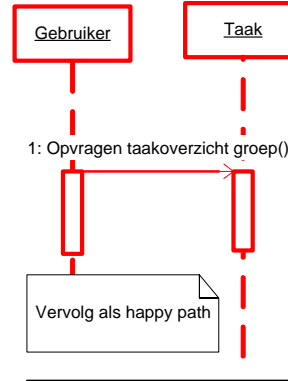
C.4 Sequence diagrammen (RUP design workflow)

Een sequence diagram beeld een interactie uit tussen verschillende klassen uit het klassediagram en een volgtijdelijkheid die ondersteund wordt door een nummering. De sequence diagrammen zijn de feitelijke realisatie van de use cases en bij de diagrammen wordt dan ook aan de betreffende use case gerealiseerd. Tussen twee klassen die met elkaar interacteren dient altijd een associatie te bestaan. Hieronder zijn de sequence diagrammen van de belangrijkste use cases opgenomen. De diagrammen kunnen ten opzichte van het implementatiemodel van het prototype iets achterhaald zijn, aangezien het prototype een nadere afbakening heeft gekregen na de ontwerp fase en derhalve kan afwijken van ontwerponderdelen.

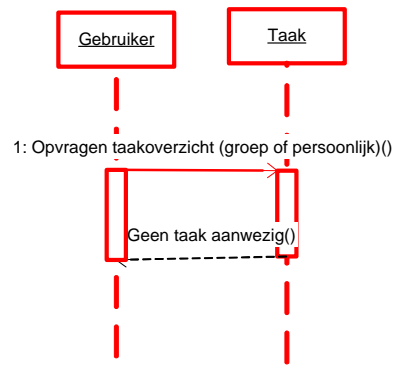
UC 1.2 Opvragen taak



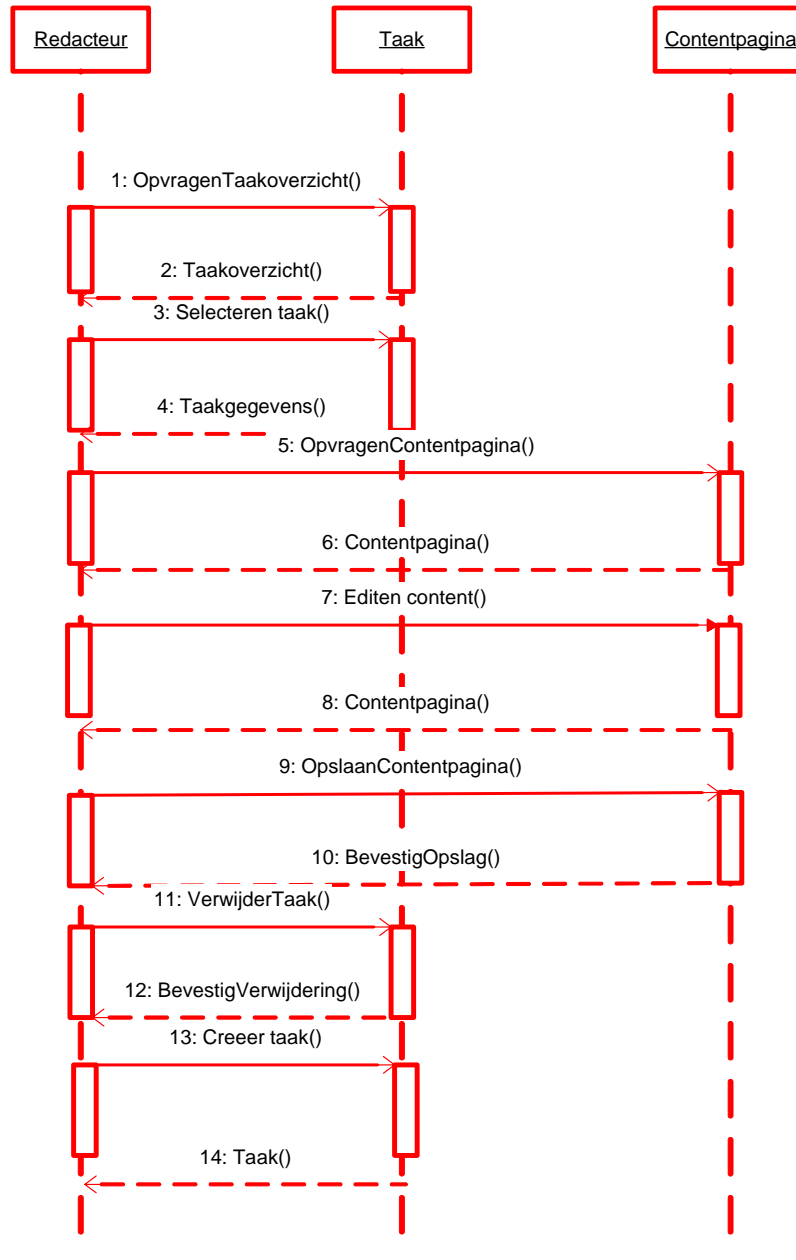
UC 1.2 Opvragen taak
Alternatief 1:
taakoverzicht groep



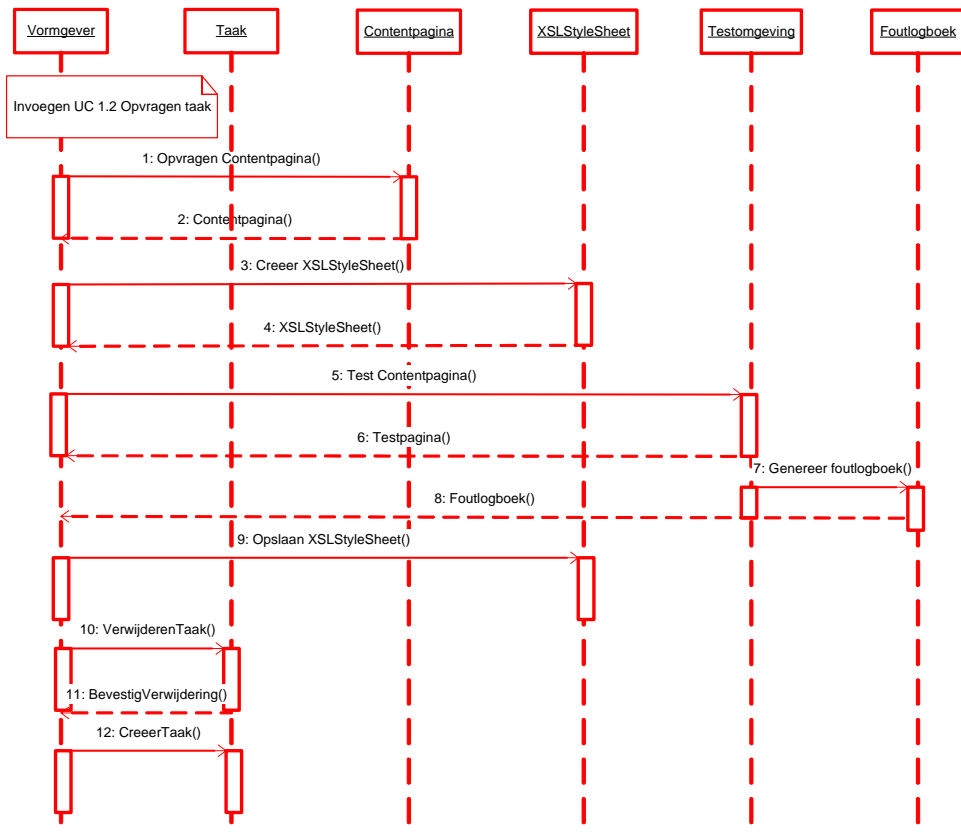
UC 1.2 Opvragen taak
Alternatief 2: geen taak
aanwezig



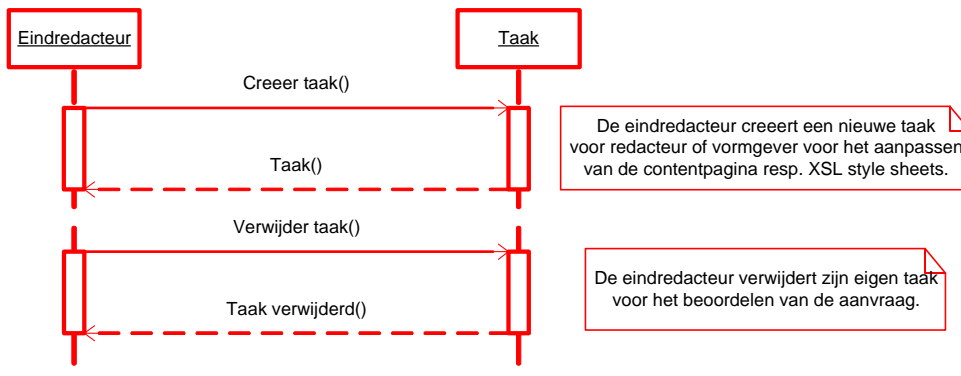
UC 2.3 Redigeren content



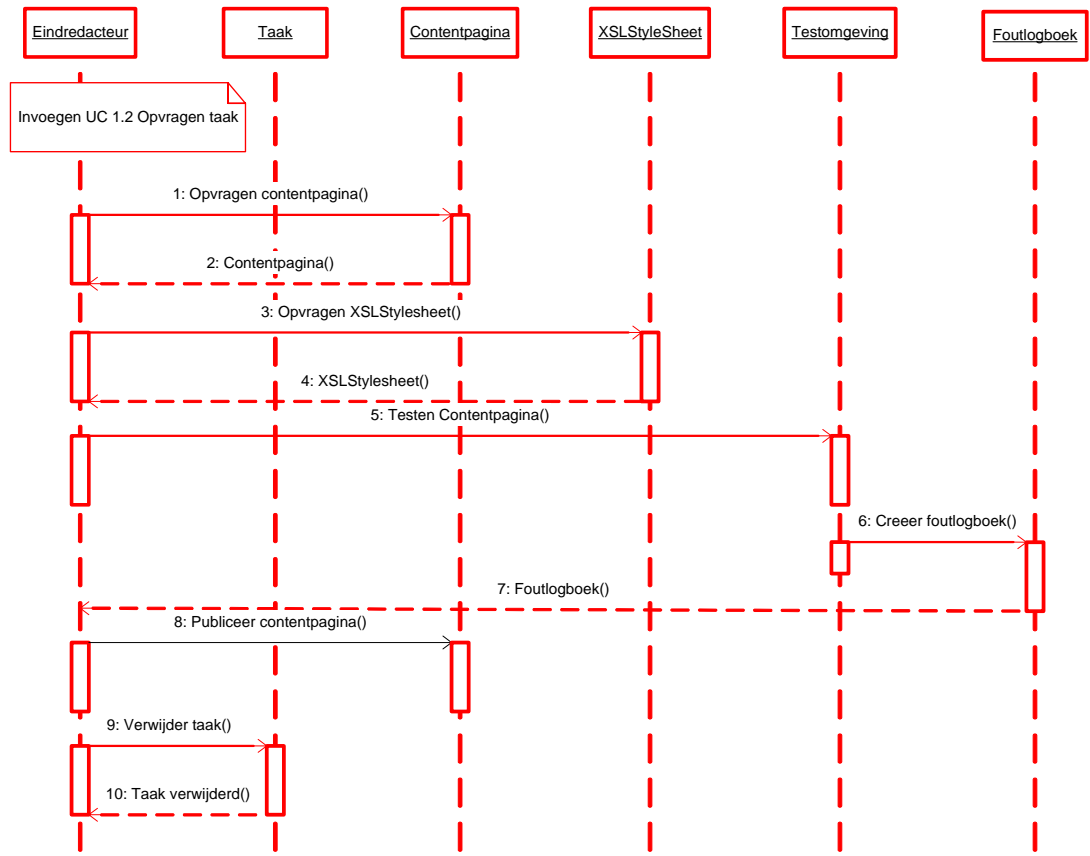
UC 3.2 Opmaken content



UC 4.1 Afkeuren aanvraag



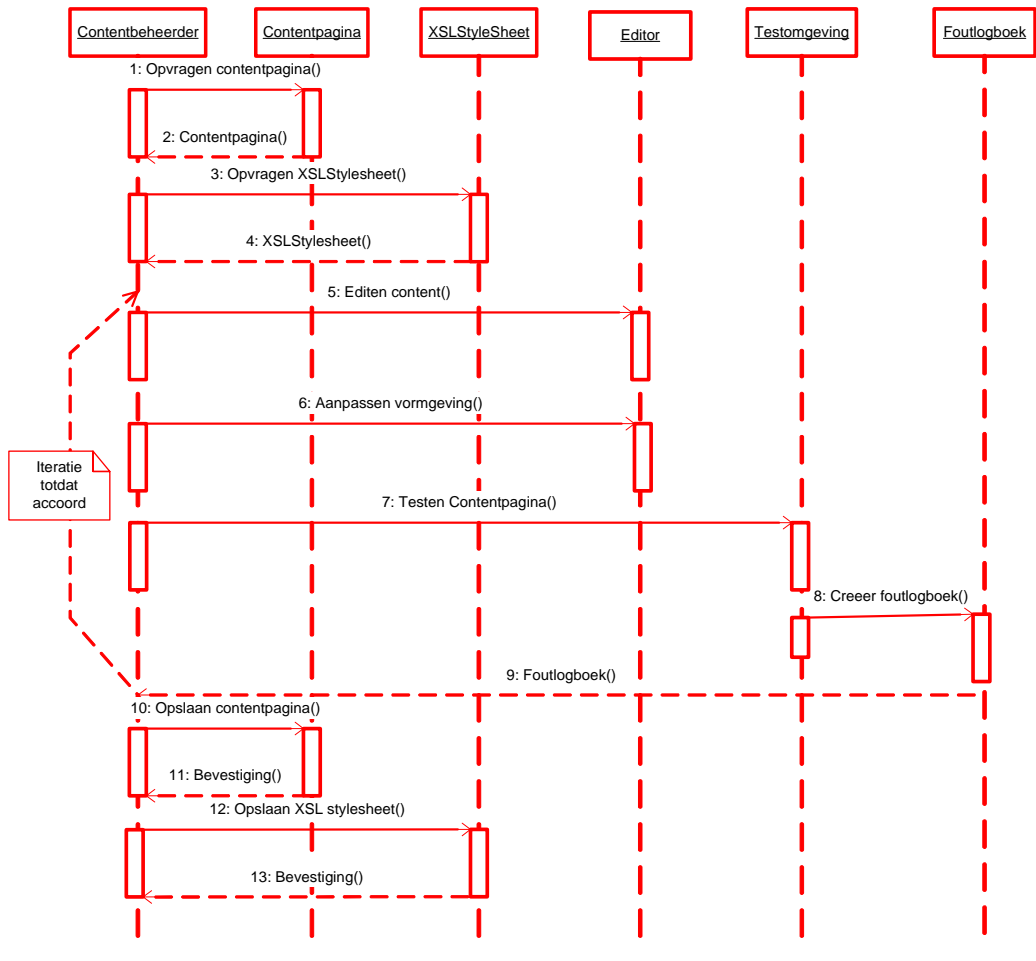
UC 4.2 Beoordelen contentpagina



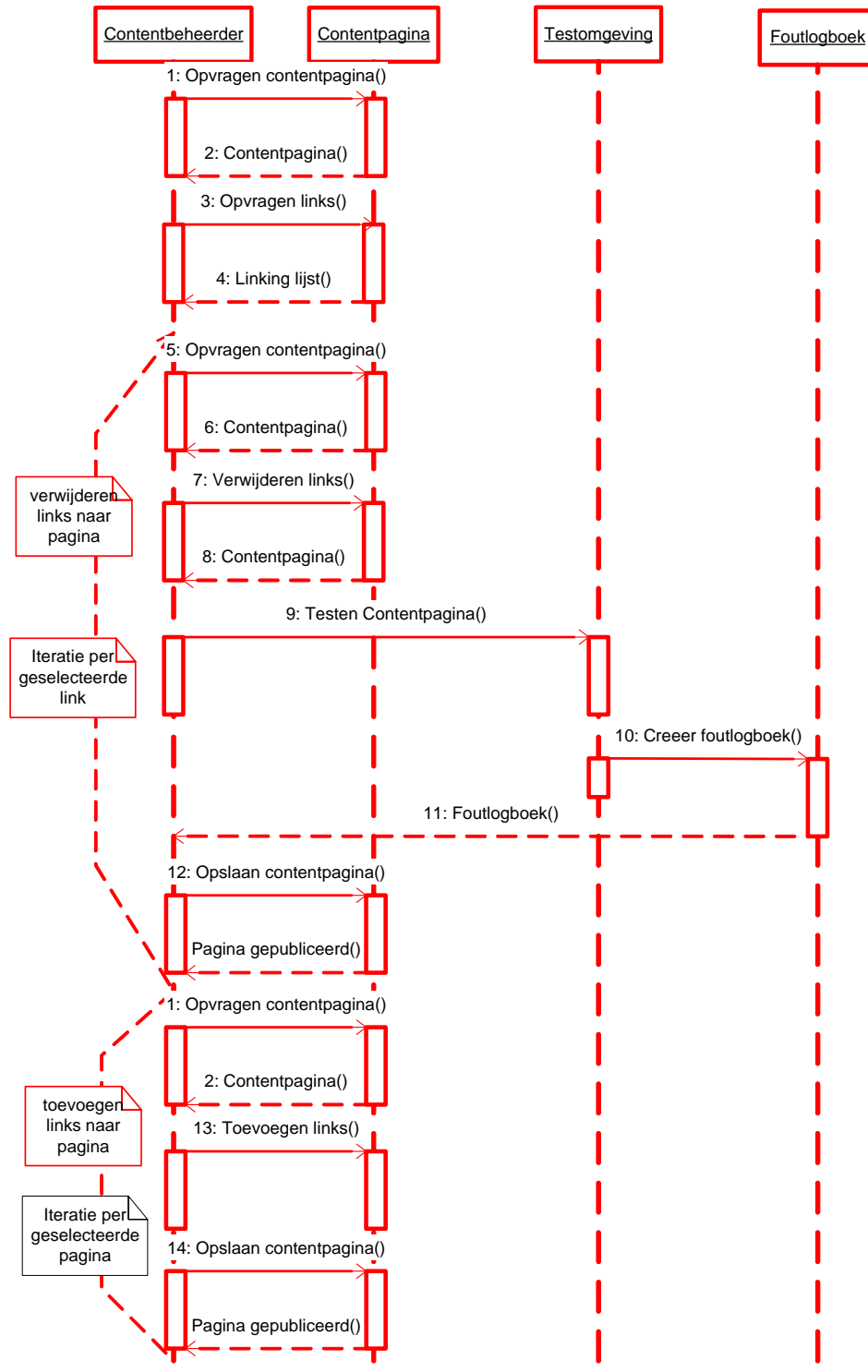
UC 4.2 Beoordelen contentpagina - alternatief 1: afkeuren aanvraag



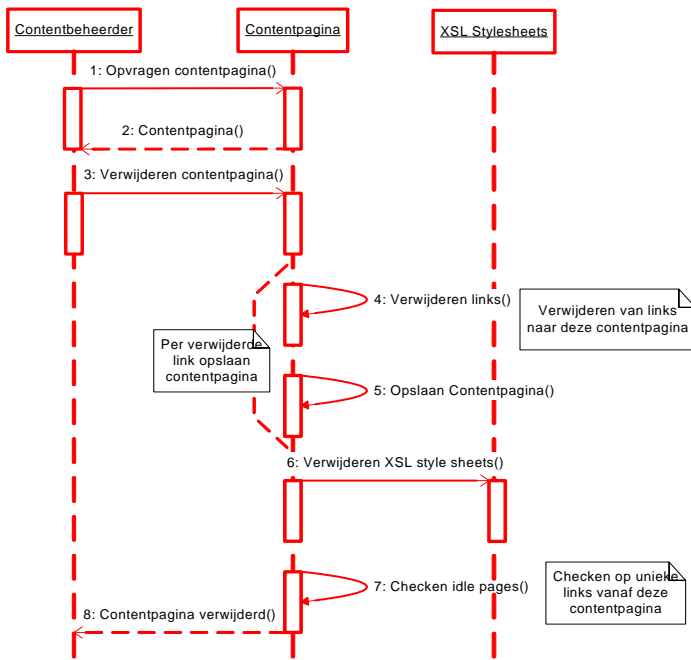
UC 6.1 Aanpassen contentpagina



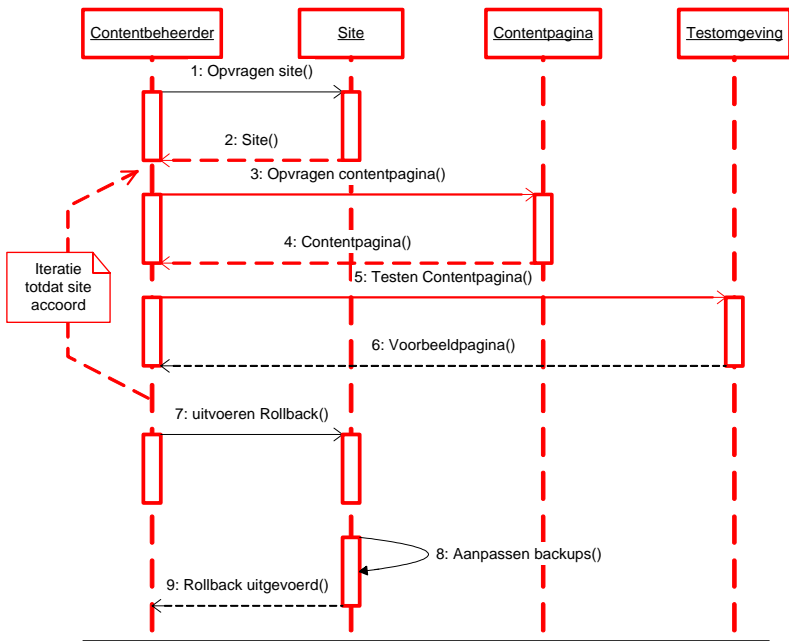
UC 6.4 Aanpassen sitestructuur
(verplaatsen contentpagina)



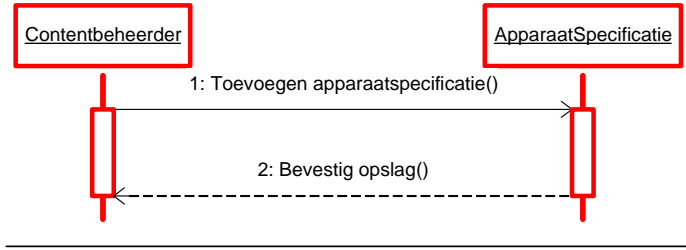
UC 6.4 Aanpassen sitestructuur
 Alternatief 1: verwijderen contentpagina



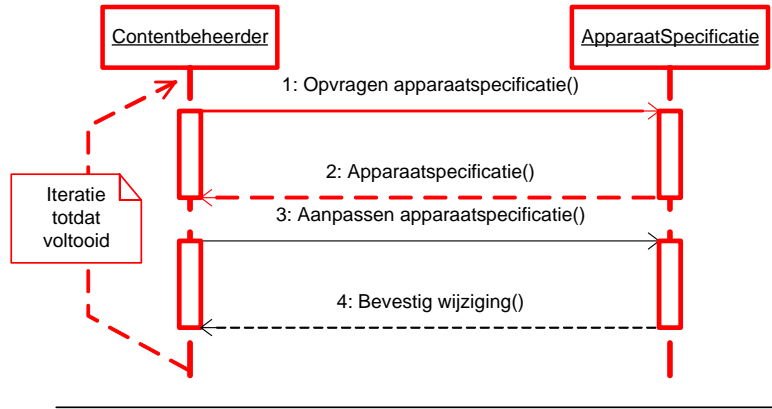
UC 6.5 Uitvoeren Rollback



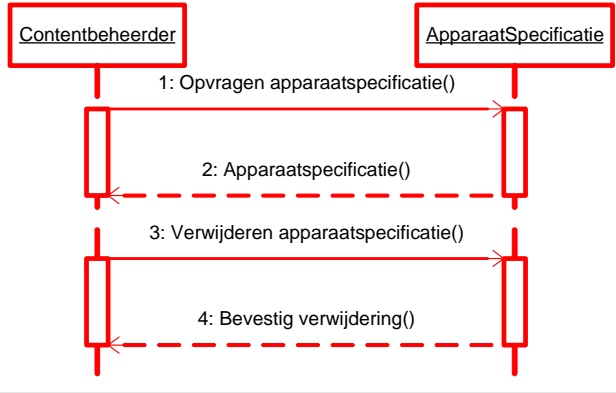
UC 6.6 Beheren apparaten database (toevoegen apparaat)



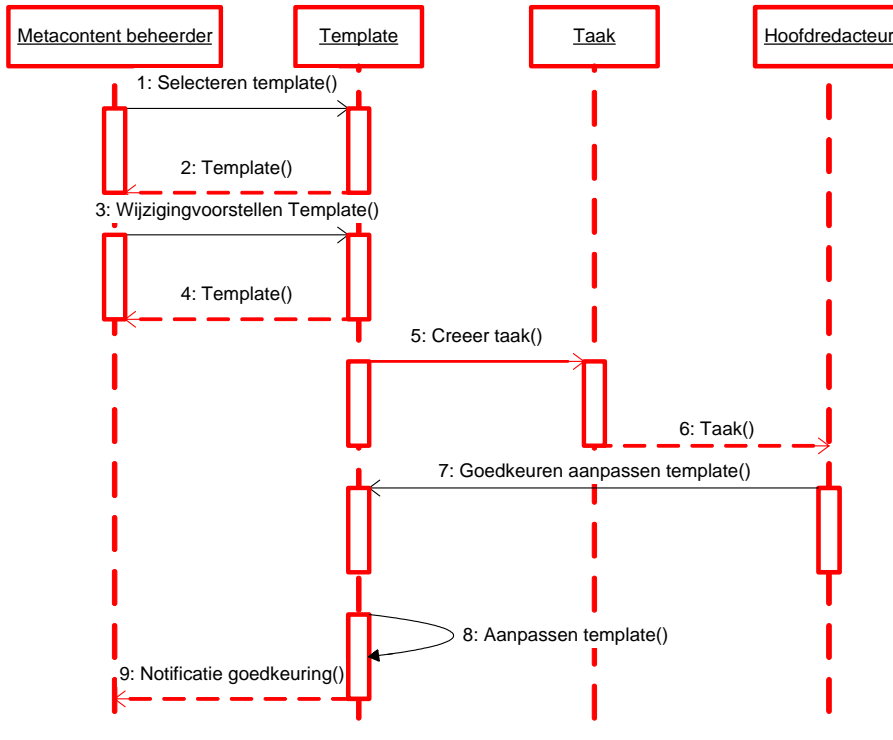
UC 6.6 Beheren apparaten database
Alternatief 1: aanpassen apparaat



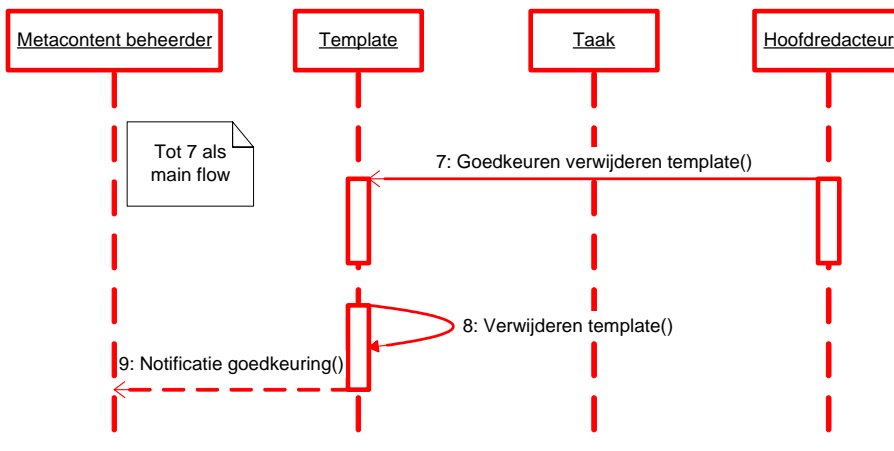
UC 6.6 Beheren apparaten database
Alternatief 2: verwijderen apparaat



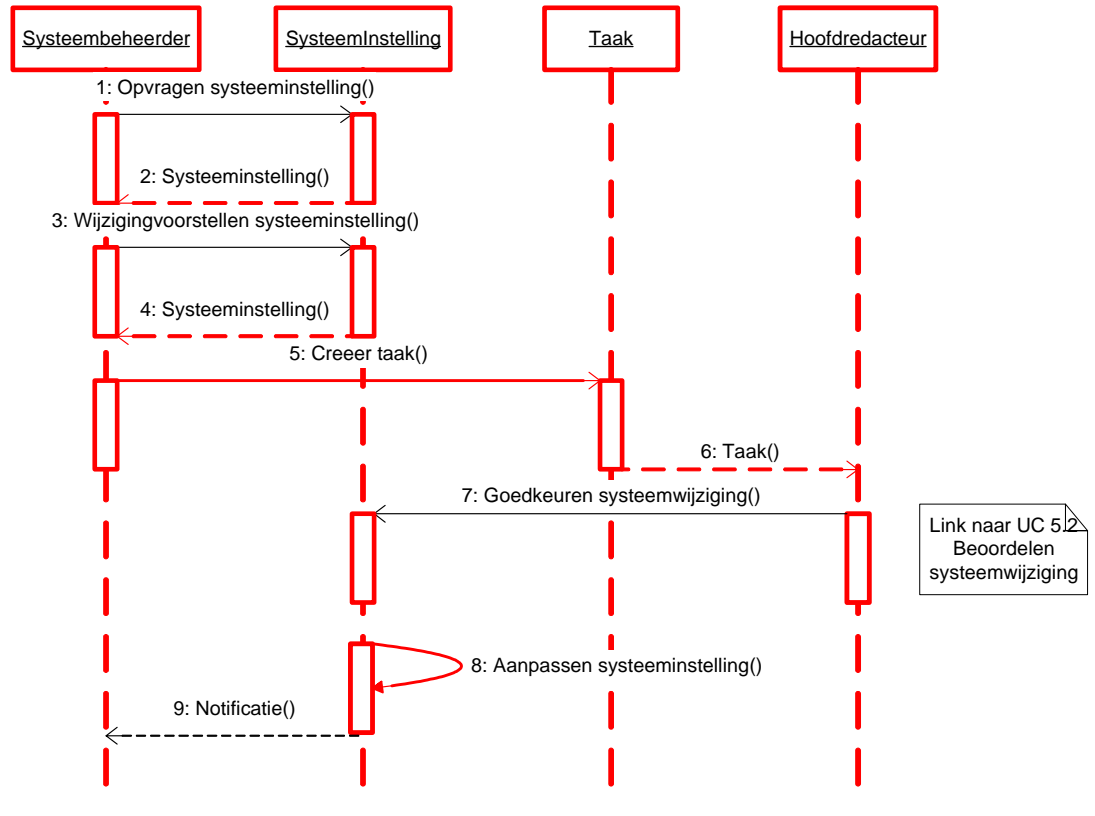
UC 7.2 Aanpassen / verwijderen template



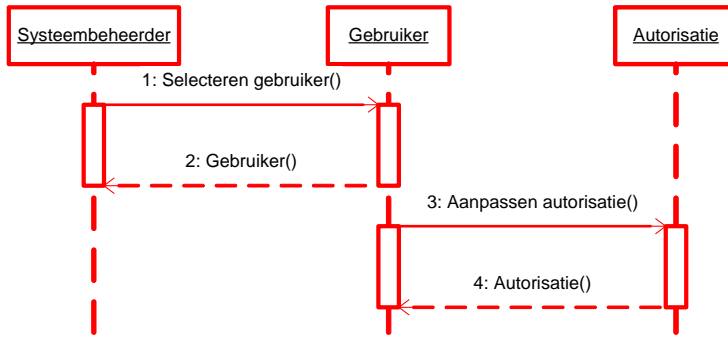
UC 7.2 Alternatief 1: verwijderen template



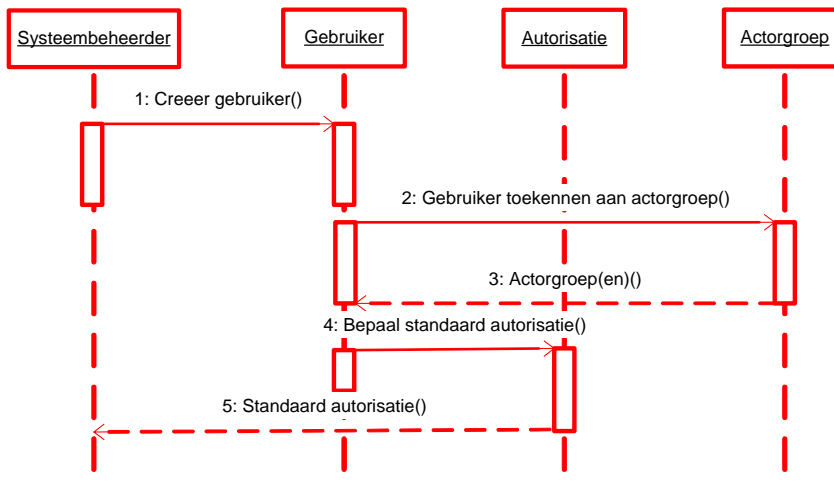
UC 8.1 Beheren systeeminstellingen



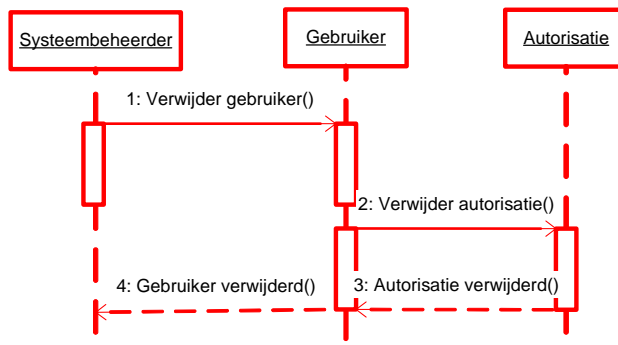
UC 8.2 Beheren autorisatiemodel
(aanpassen gebruikersautorisatie)

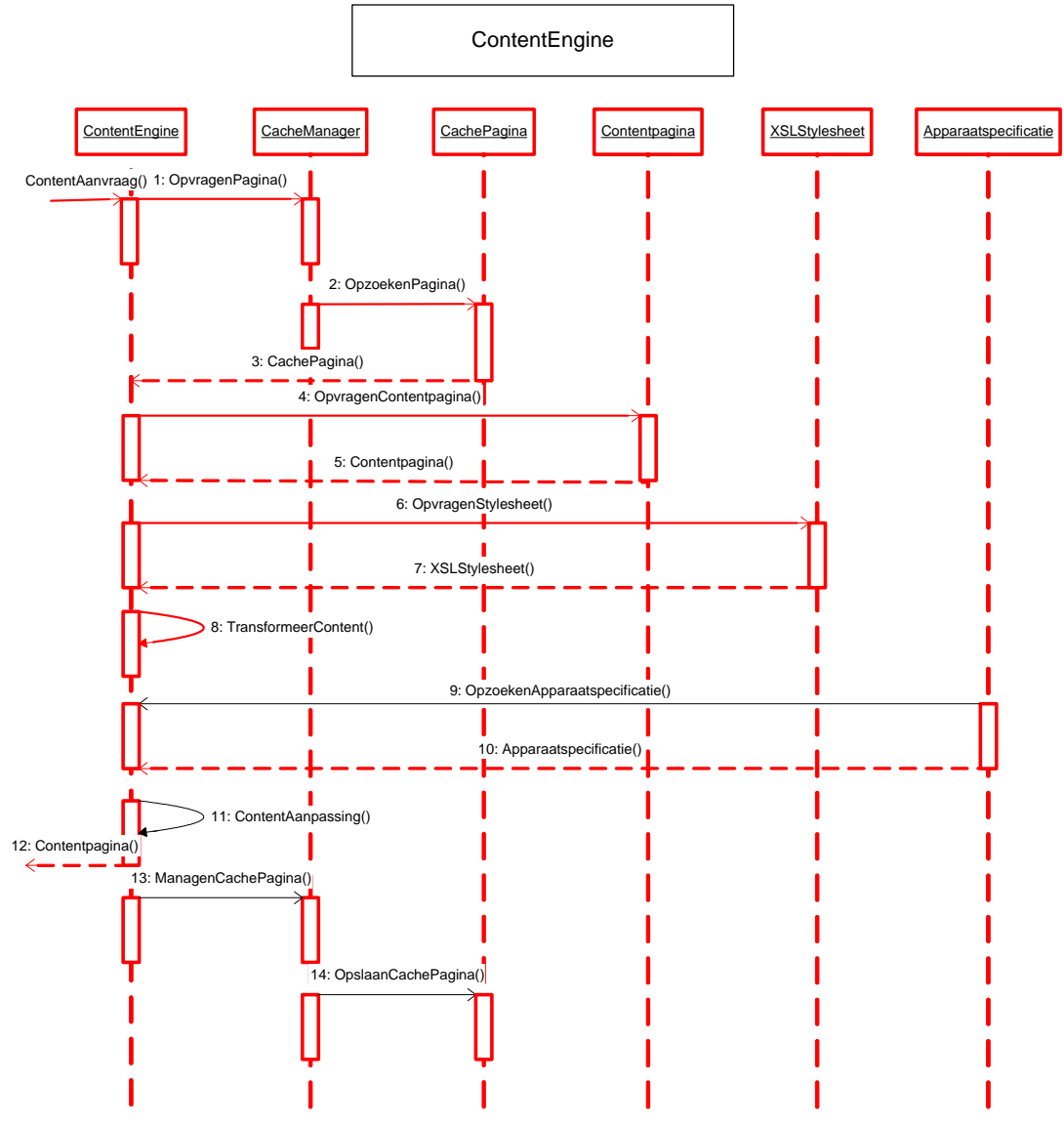


UC 8.2 Beheren autorisatiesmodel
Alternatief 1: Creer gebruiker



UC 8.2 Beheren autorisatiemodel
Alternatief 2: Verwijderen gebruiker



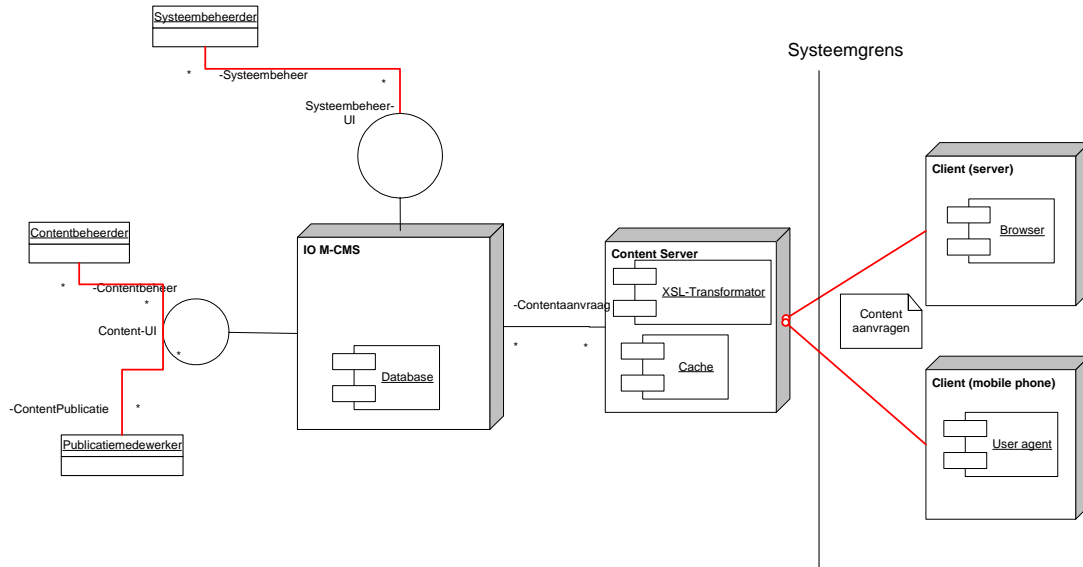


C.4 Overige diagrammen (RUP design workflow)

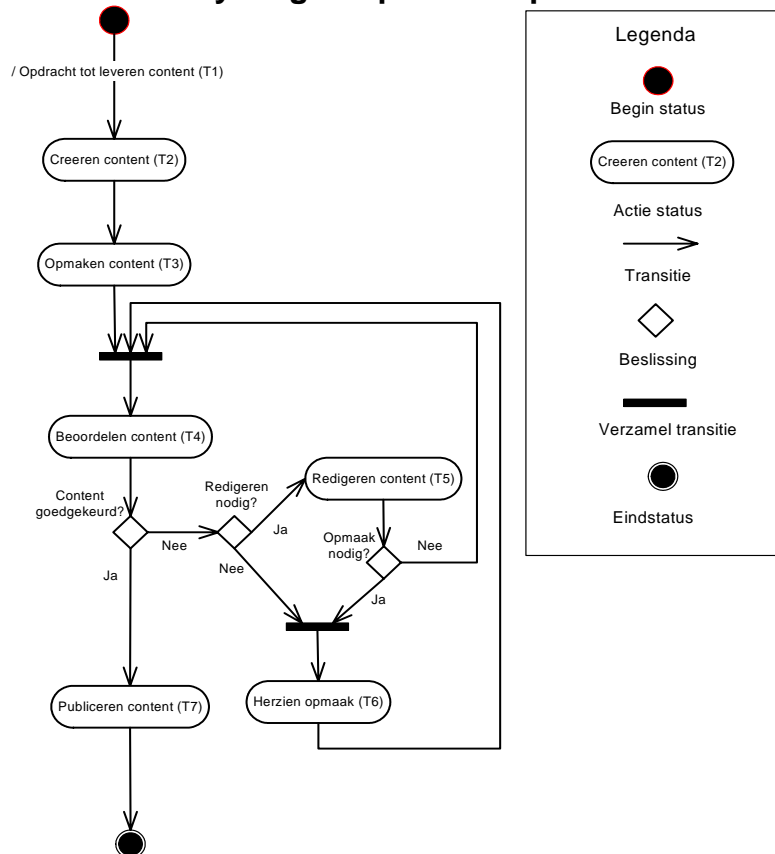
De overige diagrammen uit de ontwerpfase betreffende die diagrammen die op een bepaald nuttig werden geacht. Zo is van enkel de meest voorname en complexe klassen (taak en contentpagina) het statusdiagram opgenomen. Tevens is een globale systeemarchitectuur ingevoegd, zoals deze binnen een doelorganisatie zou kunnen worden toegepast. Deze verschilt van de opstelling van het prototype, omdat deze laatste op een ISAPI draaide en onderdelen werde gesimuleerd zoals de HTTP request.

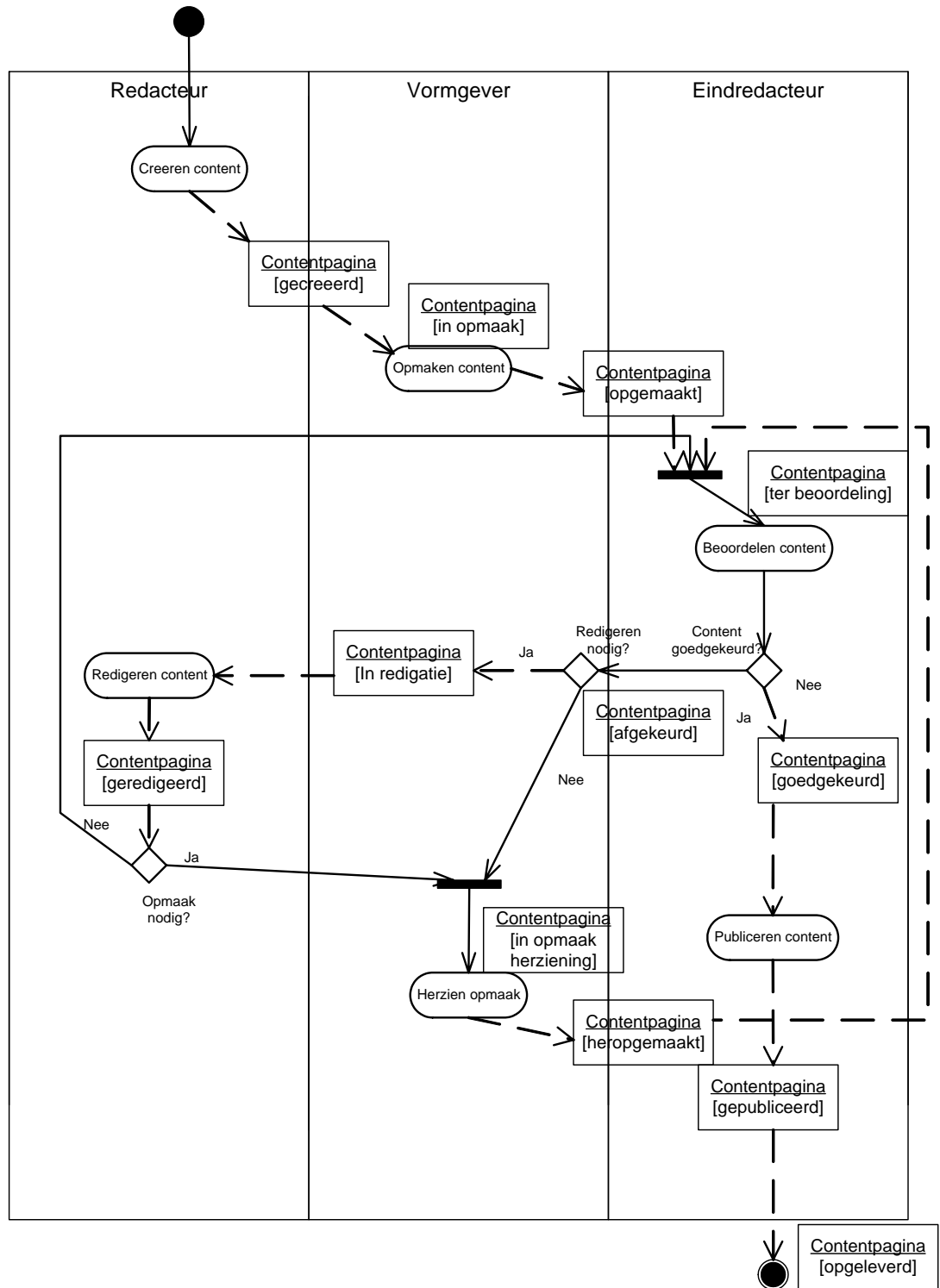
C.4.1 Globale systeemarchitectuur

Generieke systeemarchitectuur, zoals deze binnen een willekeurige doelorganisatie kan worden toegepast.

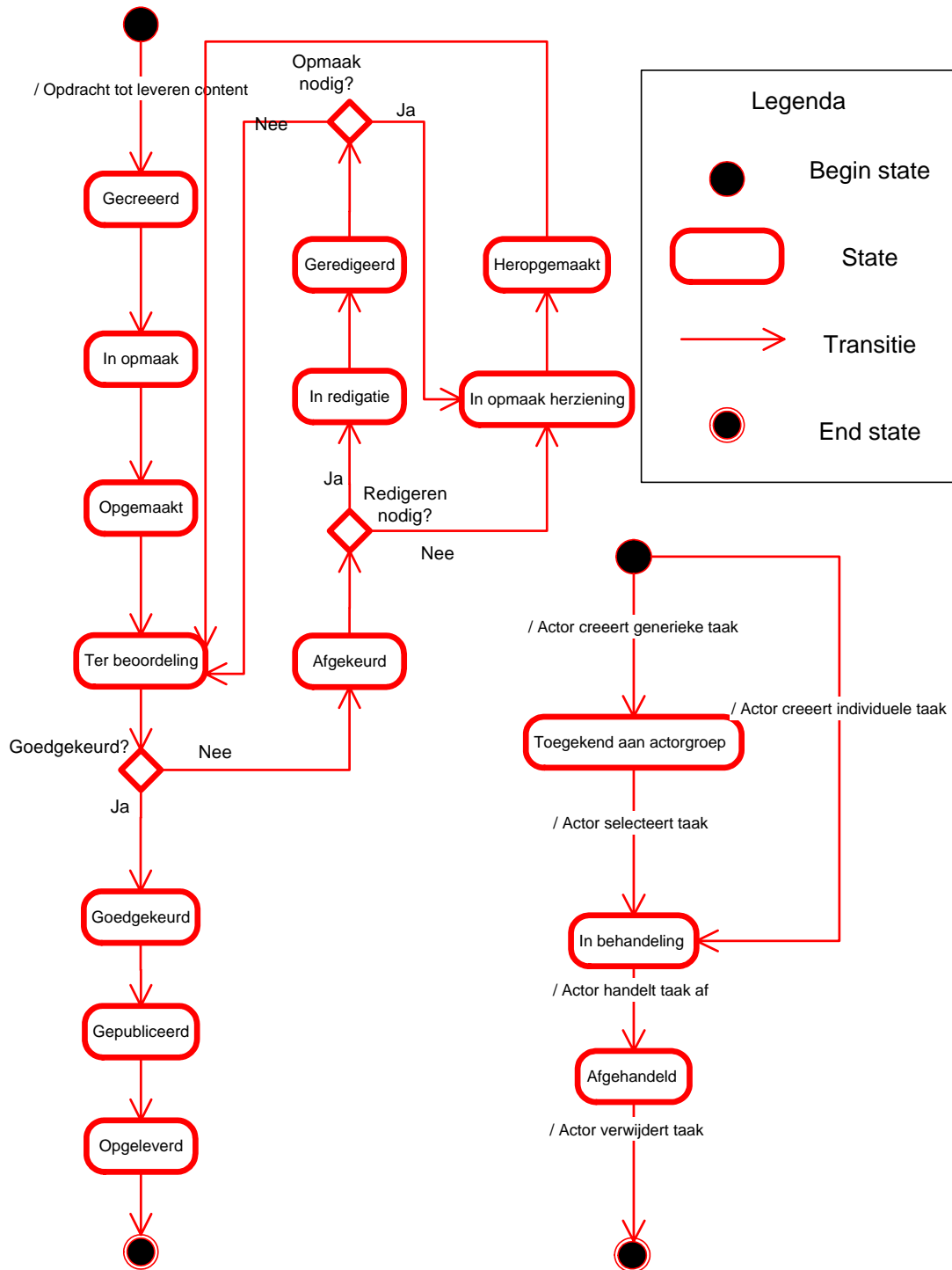


C.4.2 Activity diagram publicatieproces





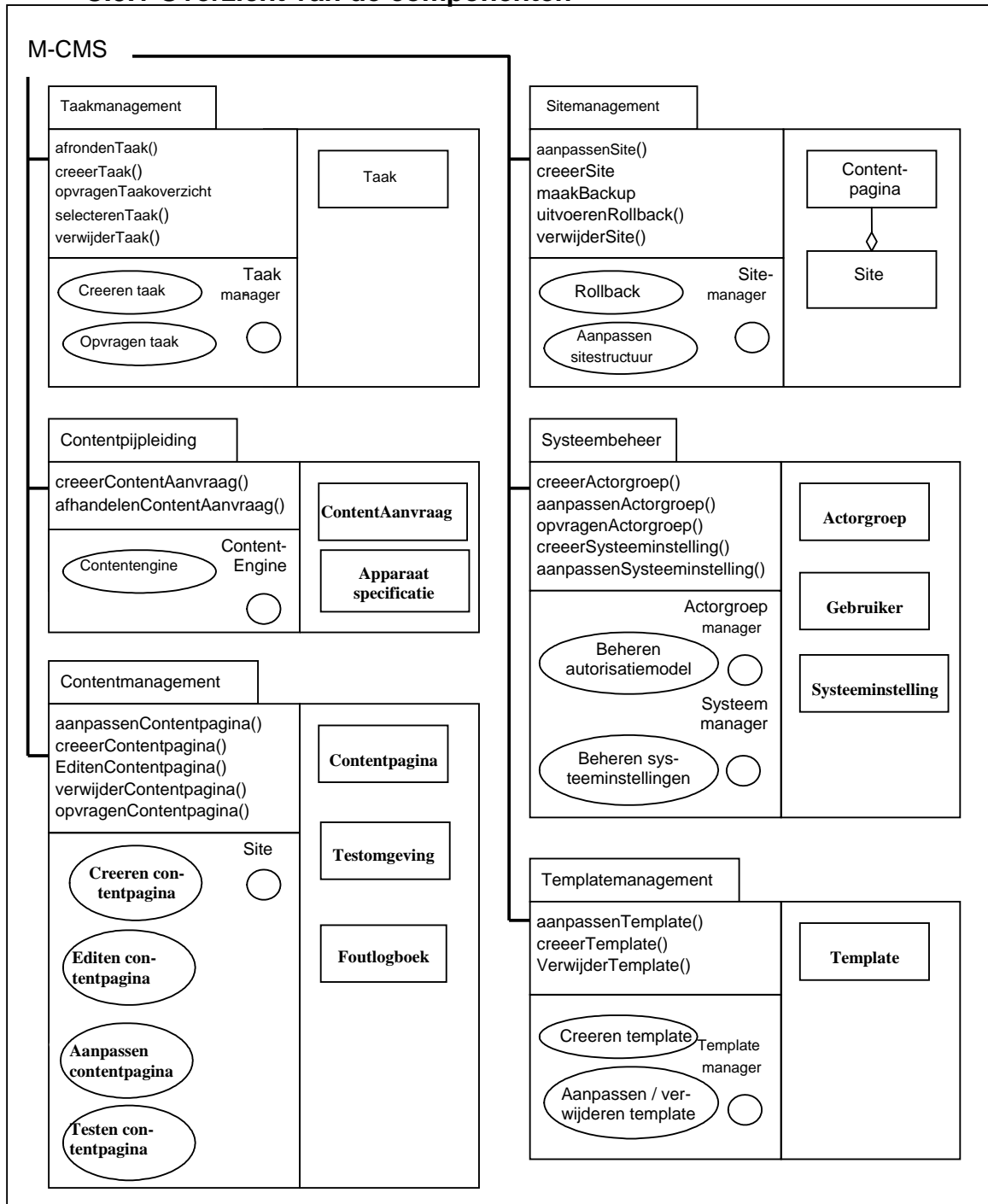
C.4.3 Statusdiagrammen



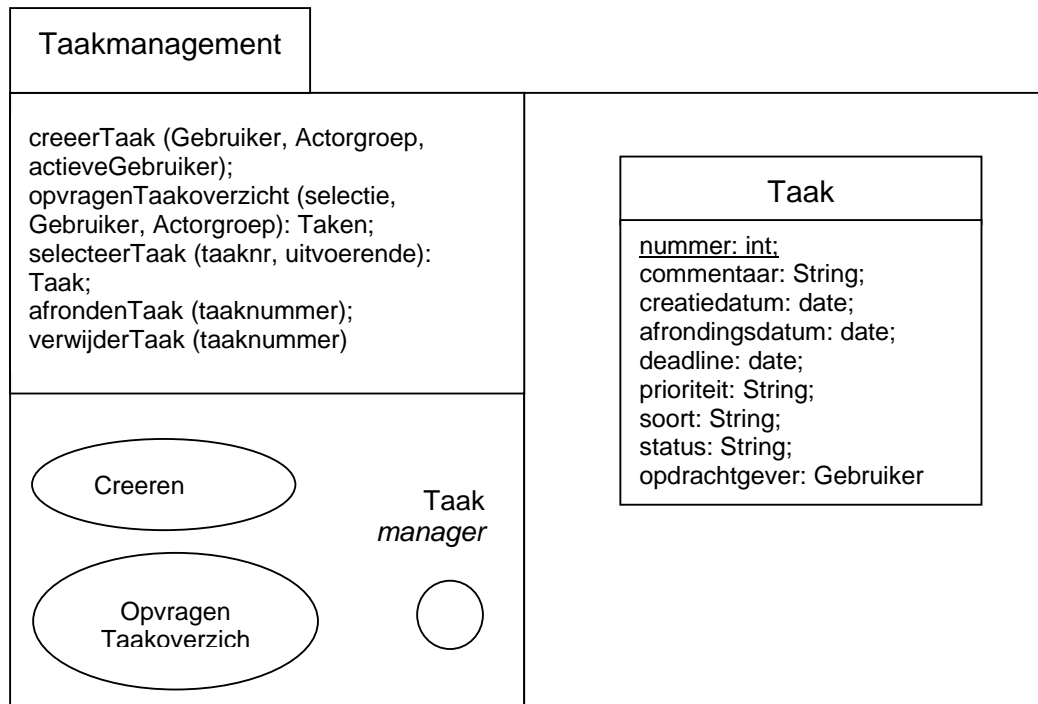
C.5 Implementatiemodel

In dit onderdeel wordt het implementatiemodel gepresenteerd aan de hand van de systeempackages. Eerst wordt een overzicht van de componenten van het M-CMS gegeven op basis van de geïmplementeerde packages. Vervolgens worden de verschillende individuele packages meer gedetailleerd gegeven.

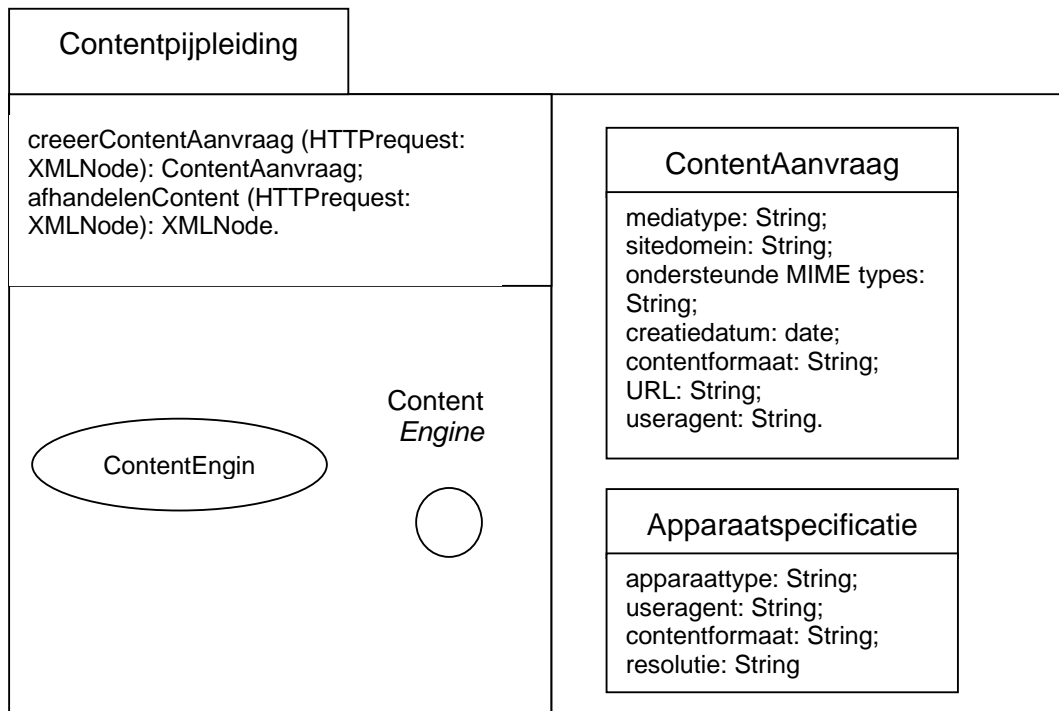
C.5.1 Overzicht van de componenten



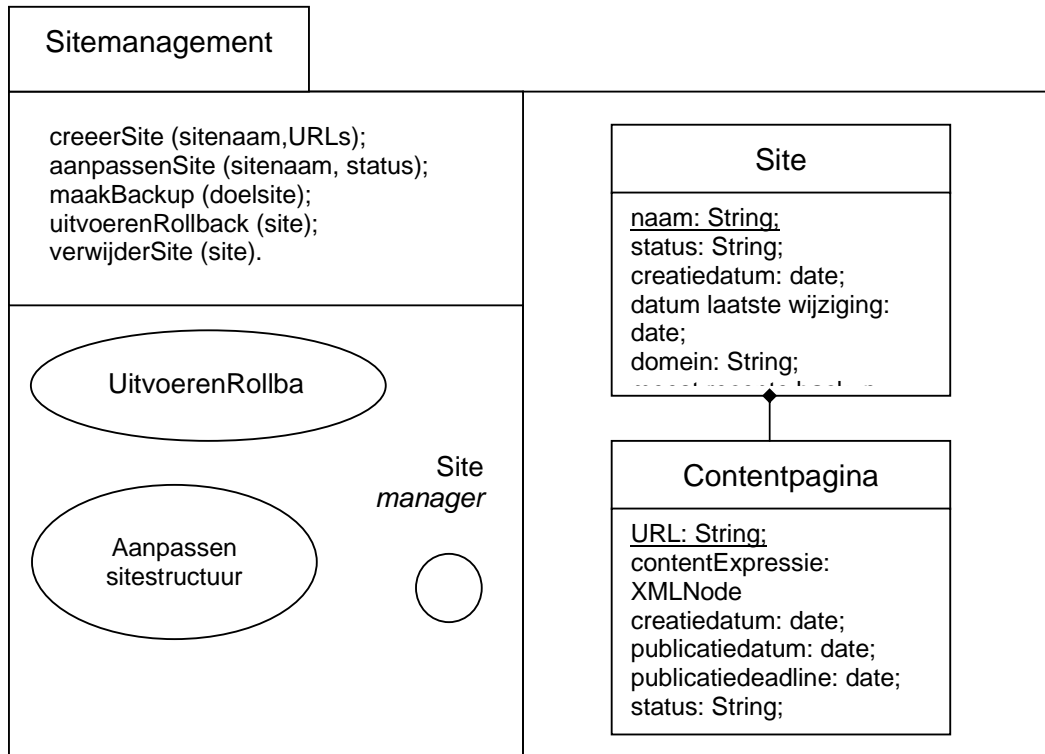
C.5.2 Taakmanagement



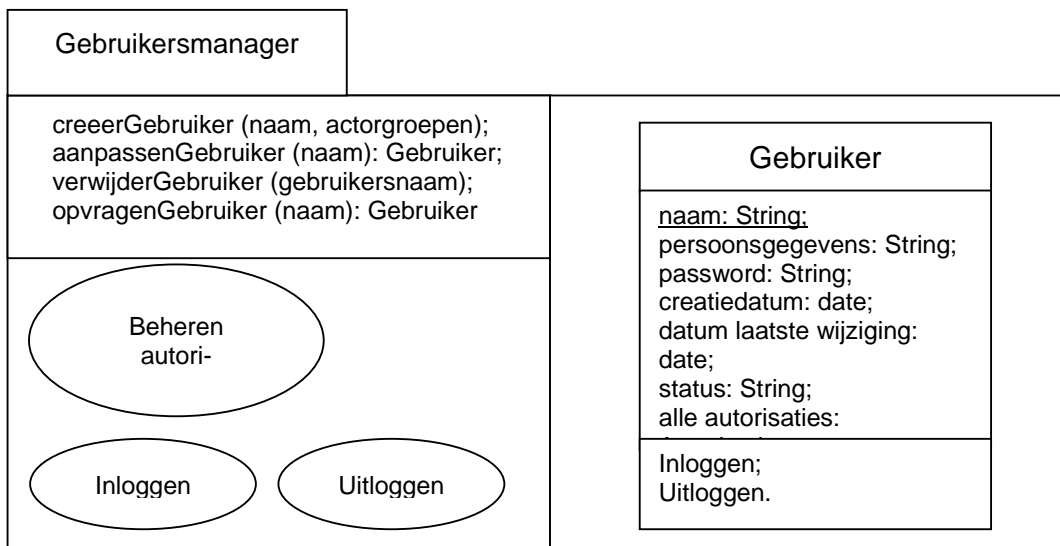
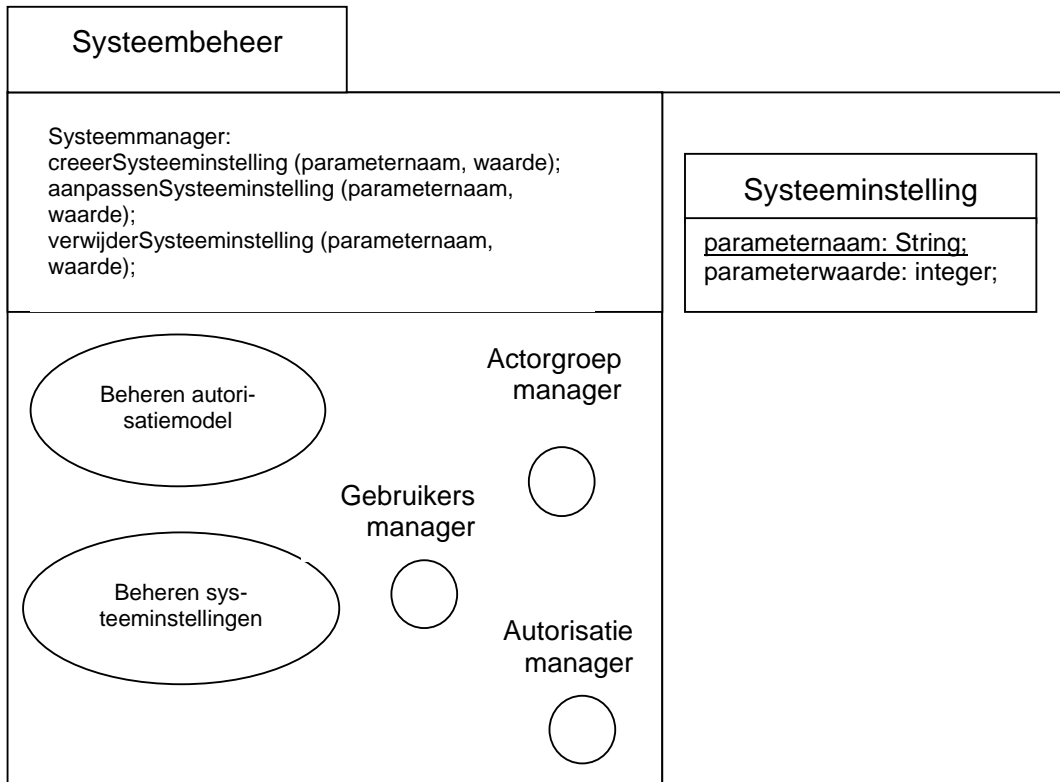
C.5.3 Contentpijpleiding

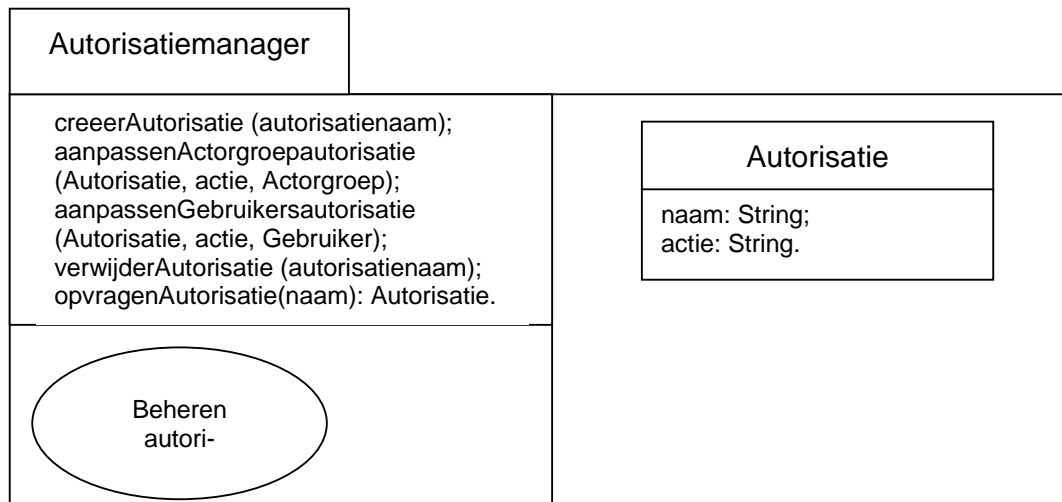


C.5.4 Sitemanagement

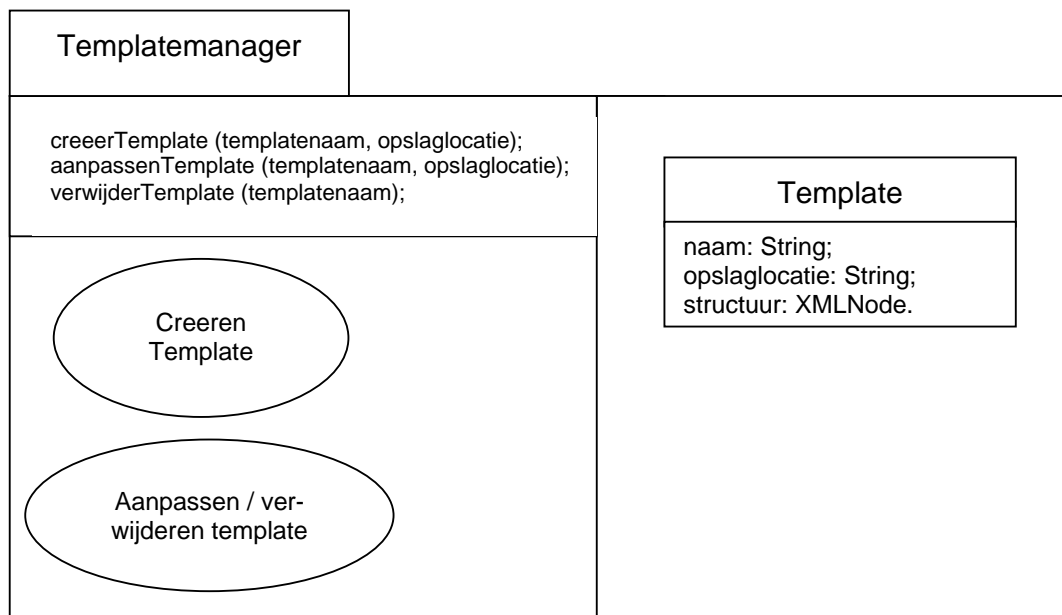


C.5.5 Systeembeheer





C.5.6 Templatemanagement



C.5.7 Contentmanagement

