



Persistente identifiers:

Identificatiemechanismes in het digitale tijdperk

Daniel Steinmeier

Inleiding

Wat is, afgezien van conventies, feitelijk het verschil tussen een naam en een identifier? Beide zijn bedoeld om iets te identificeren. Maar waar we van een naam doorgaans niet verwachten dat deze gegarandeerd uniek is, verwachten we dit van een identifier wel, tenminste binnen de context van een identificatiesysteem.

Iedereen die werkzaam is bij een instelling met een eigen catalogus of database weet echter dat de context van identifiers weleens kan veranderen. Met name wanneer een instelling overgaat op een nieuw catalogussysteem, wanneer databases samengevoegd worden of wanneer records gepubliceerd worden op internet, zien we dat ontwikkelaars er vaak voor kiezen (of zelfs genoodzaakt zijn) om records te voorzien van een nieuwe identifier zodat ze ook binnen de nieuwe context gegarandeerd uniek zijn. De identifiers uit het oude systeem worden in zulke gevallen dan gedegradeerd tot een 'bron-id'. Zelfs wanneer nog op deze oude identifiers gezocht kan worden, zijn deze toch afhankelijk geworden van extra context om op de juiste manier geïnterpreteerd te kunnen worden. Anders gezegd, we moeten niet alleen weten wat de identifier is maar ook binnen welke context deze uniek is.

Een dergelijke manier van identificeren vereist echter dat het identificatienummer veel semantiek bevat om gegarandeerd uniek te kunnen zijn binnen een nieuwe context. Semantische waardes meegeven aan identifiers kan echter problematisch zijn aangezien dingen met betekenis, zoals collectionnamen of bedrijfsnamen, veranderlijk zijn. Met de komst van internet is dit probleem nog groter geworden omdat het kader waarbinnen objecten gebruikt en hergebruikt worden enorm is toegenomen. Het is dan ook van groot belang om identifiers te hebben die uniek zijn over de hele wereld en die zo min mogelijk semantiek bevatten zodat de noodzaak om de identifiers aan te passen minimaal is. Om te bereiken dat nummers gegarandeerd uniek zijn moet de uitgifte van identifiers internationaal gecoördineerd worden. Hiermee wordt voorkomen dat dezelfde nummers door verschillende partijen gebruikt worden.

Natuurlijk zijn dergelijke mechanismes sinds de jaren '70 gangbaar geworden. Het bekendste voorbeeld is misschien wel het nummer waarmee literaire uitgaves uniek kunnen worden geïdentificeerd namelijk ISBN (International Standard Book Number). Ook voor vele andere producten (van muziekopnames tot aan wijn) zijn identificatiemechanismen ontworpen, vaak ontwikkeld onder de hoede van de Internationale Organisatie voor Standaardisatie (ISO). Veel van deze mechanismes zijn echter bedacht voor objecten met een fysieke representatie. Hierdoor zijn ze niet geoptimaliseerd voor gebruik op het web of voor het verwijzen naar digitale objecten. Verwijzingen op het web zijn namelijk complexer dan men op het eerste gezicht zou denken. We zouden natuurlijk kunnen stellen dat naar een specifieke bron op het web altijd uniek verwezen kan worden door middel van het adres: een domeinnaam is tenslotte een centraal beheerd mechanisme dat naar een bepaalde server verwijst en alles na de domeinnaam specificiert de locatie van de bron op de webserver. Dit principe houdt echter geen rekening met domeinnamen die veranderen, pagina's die hernoemd of verhuisd worden, versiebeheer van objecten of het feit dat dezelfde objecten op verschillende webpagina's tegelijk kunnen staan.

Om deze reden zijn er de afgelopen jaren verschillende organisaties geweest die zich hebben bezig gehouden met het ontwikkelen van een systeem voor zogenaamde persistente identifiers.

Deze identifiers zijn speciaal gericht op het uniek identificeren van objecten op het web en bestemd voor het permanent doorverwijzen naar digitale representaties van objecten, of metadata hiervan, zonder afhankelijkheid van eventuele veranderingen aan onderliggende websites.

Echter, zoals het wel vaker gaat bij de ontwikkeling van technieken op internet is er uiteindelijk niet één gestandaardiseerde manier voor het beheer en het aanmaken van persistente identifiers ontwikkeld, maar is er een veelheid aan mechanismes ontstaan met gedeeltelijk verschillende en gedeeltelijk overlappende functionaliteit. Maar er is wel een aantal eigenschappen te benoemen dat kenmerkend is voor de meeste implementaties van persistente identifiers.

In dit artikel zal eerst kort geschetst worden welke vormen van identificatie en localisering gangbaar zijn op het web. Vervolgens zal beschreven worden welke voordelen persistente identifiers kunnen bieden en wat de hun kenmerkende eigenschappen zijn. Tenslotte zal het gebruik van persistente identifiers aan de hand van een usecase nader beschouwd worden en zal bekeken worden welke zaken nog te verbeteren zijn aan de techniek.

URI, URL en URN

Om te begrijpen hoe en waarom er verschillende mechanismes zijn ontstaan voor het aanmaken van persistente identifiers is het goed om eerst te kijken hoe binnen de techniek van het internet naar webpagina's verwezen wordt en welke problemen dit met zich meebrengt. In het kader van identificatie op het web worden vaak de termen URL, URI en URN gebruikt, soms ook op een wellicht wat verwarrende manier door elkaar. Maar wat is eigenlijk het verschil tussen deze drie termen, afgezien van de laatste letter?

Binnen de context van dagelijks internetgebruik is de term URL waarschijnlijk de bekendste als het gaat om verwijzingen naar webpagina's. Het is tevens de oudste van de drie hierboven genoemde afkortingen en was al een onderdeel van de eerste technische implementatie van websoftware in 1991.¹ De 'Uniform Resource Locator' wordt gebruikt als een manier om aan webpagina's te kunnen refereren door middel van een naam. Dit werkt dankzij DNS (Domain Name System) dat ervoor zorgt dat aan de hand van het opgegeven domein doorverwezen kan worden naar het IP-adres van de webserver. Dit adres moet uiteindelijk bekend zijn om communicatie met een webserver tot stand te kunnen brengen. De URL wordt ook wel adres genoemd en is een manier om te refereren aan bronnen die op het web te vinden zijn. Het doel van de URL was echter voornamelijk om te dienen als een systeem voor het adresseren van webpagina's, niet om deze te identificeren.

Om een overkoepelend concept te creëren dat zowel kan dienen als term voor het localiseren van objecten als voor het identificeren van objecten, werd in 1994 de term URI geïntroduceerd in een zogenaamde 'Request for comments' (RFC). Dit was een soort memo geschreven door Tim Berners-Lee, de man die een aantal jaar daarvoor het eerste voorstel schreef dat leidde tot de ontwikkeling van het web.

URI staat voor 'Uniform Resource Identifier' en kan geclassificeerd worden als een concept dat zowel een identificerende als een localiserende functie kan hebben.² De URL is daarmee een subset van een URI aangezien deze laatste slechts tot doel heeft om bronnen te localiseren, niet om deze te identificeren.

Een concept, daarentegen, dat alleen bedoeld is voor identificatie is tenslotte de URN (Uniform Resource Name).

¹ Hilse 2006, 4

² Hilse 2006, 4

Deze term is ook vastgelegd in een RFC in 1994. Het idee achter de URN is om een mechanisme te bieden voor het identificeren van bronnen zonder daarbij afhankelijk te zijn van de specifieke locatie. Deze techniek vervult dus slechts de identificerende functie van het URI-concept. In de memo wordt de functie van de URN beschreven als: het bieden van een unieke persistente identifier voor herkenning, toegang tot de bron en informatie over de bron. Hierbij worden ook al enkele eigenschappen genoemd die later ook bij implementaties van systemen voor persistente identifiers van belang zijn zoals uniciteit, schaalbaarheid en de mogelijkheid om aan de hand van een identifier te kunnen doorverwijzen naar een locatie.³ URN's zijn opgebouwd volgens een vast principe: elke URN begint met 'urn:', vervolgens komt een zogenaamde Namespace Identifier (NID) en hierna de identifier zelf, ook wel de Namespace Specific String (NSS) genoemd. De NID bepaalt uiteindelijk hoe de identifiers die na de NID volgen opgebouwd moeten worden. Elke NID moet geregistreerd zijn bij de Internet Assigned Numbers Authority (IANA), door middel van een RFC. In 2001 is bijvoorbeeld een NID geregistreerd door Juha Hakala van de universiteitsbibliotheek van Helsinki, voor het National Bibliography Number (NBN).⁴ Dit nummer wordt door Nationale Bibliotheken doorgaans toegekend aan publicaties zonder ISBN- of ISSN-nummer. De registratie van de NID maakt het mogelijk voor iedereen in de wereld die deze nummers gebruikt om een geldige URN te vormen op basis van hun al bestaande NBN-identifiers. Omdat elk land een eigen systeem gebruikt voor het vormen van deze nummers is in de RFC bepaald dat door middel van landcodes en organisatiecodes de uniciteit van URN's gewaarborgd moet worden.⁵ De RFC bevat ook een suggestie voor implementatie van een zogenaamde 'resolver' op het web zodat URN-NBN-identifiers ook gebruikt kunnen worden als een locatiemechanisme voor het doorverwijzen naar een object of informatie over een object. Op deze manier worden binnen een dergelijke implementatie beide functies van het URI-concept vervuld, namelijk identificeren en localiseren.

De voordelen van persistente identifiers

'Dode' links

Zoals het concept van de URN al laat zien is het voordeel van een persistente identifier boven een URL dat de identifier onafhankelijk is van een achterliggende locatie die tenslotte kan veranderen tijdens de levenscyclus van een digitaal object. In de literatuur over dit onderwerp is het vaak gehoord argument dat persistente identifiers er uiteindelijk voor moeten zorgen dat er geen 'dode' links ontstaan gedurende de tijd dat een object op het web aangeboden wordt.⁶ Er zijn namelijk verschillende redenen waarom links niet meer werken: domeinnamen veranderen, pagina's krijgen een andere plek op de webserver of documenten worden geheel verwijderd.⁷

Dit probleem werd door Tim Berners-Lee zelf ook al onderkend in 1998 in het invloedrijke artikel 'Cool URI's don't change'. Hij schetst in dit artikel veelgehoorde redenen voor het veranderen van links en stelt een aantal oplossingen voor om wijzigingen te voorkomen.

³ RFC 1373

⁴ RFC: 3188

⁵ Ibid.

⁶ Zie bijvoorbeeld Hilse 2006, 6 en Hakala 2010

⁷ Hilse 2006, 6

Hij doet bijvoorbeeld de suggestie om een aantal zaken die doorgaans snel veranderen zoals auteur, onderwerp en status niet op te nemen in een URL om zo te vermijden dat deze na verloop van tijd gewijzigd moet worden.⁸ Omdat zijn artikel vooral aanstuurt op het creëren van onveranderlijke verwijzingen naar websites wordt het idee van “Cool URI’s” vaak genoemd in verband met persistente identifiers. Er is echter geen consensus over de vraag in hoeverre “Cool URI’s” daadwerkelijk beschouwd moeten worden als een doeltreffend mechanisme voor implementatie van het concept van persistente identifiers. Juha Hakala bijvoorbeeld, stelt in zijn artikel over persistent identifiers dat dit *niet* het geval is omdat het aanmaken van “Cool URI’s” niet centraal gecoördineerd is zoals bij het aanmaken van ISBN-nummers en omdat deze niet kunnen voorzien in versiebeheer of in het leggen van een relatie tussen gelijke objecten die op verschillende locaties worden aangeboden.⁹

Herkomst

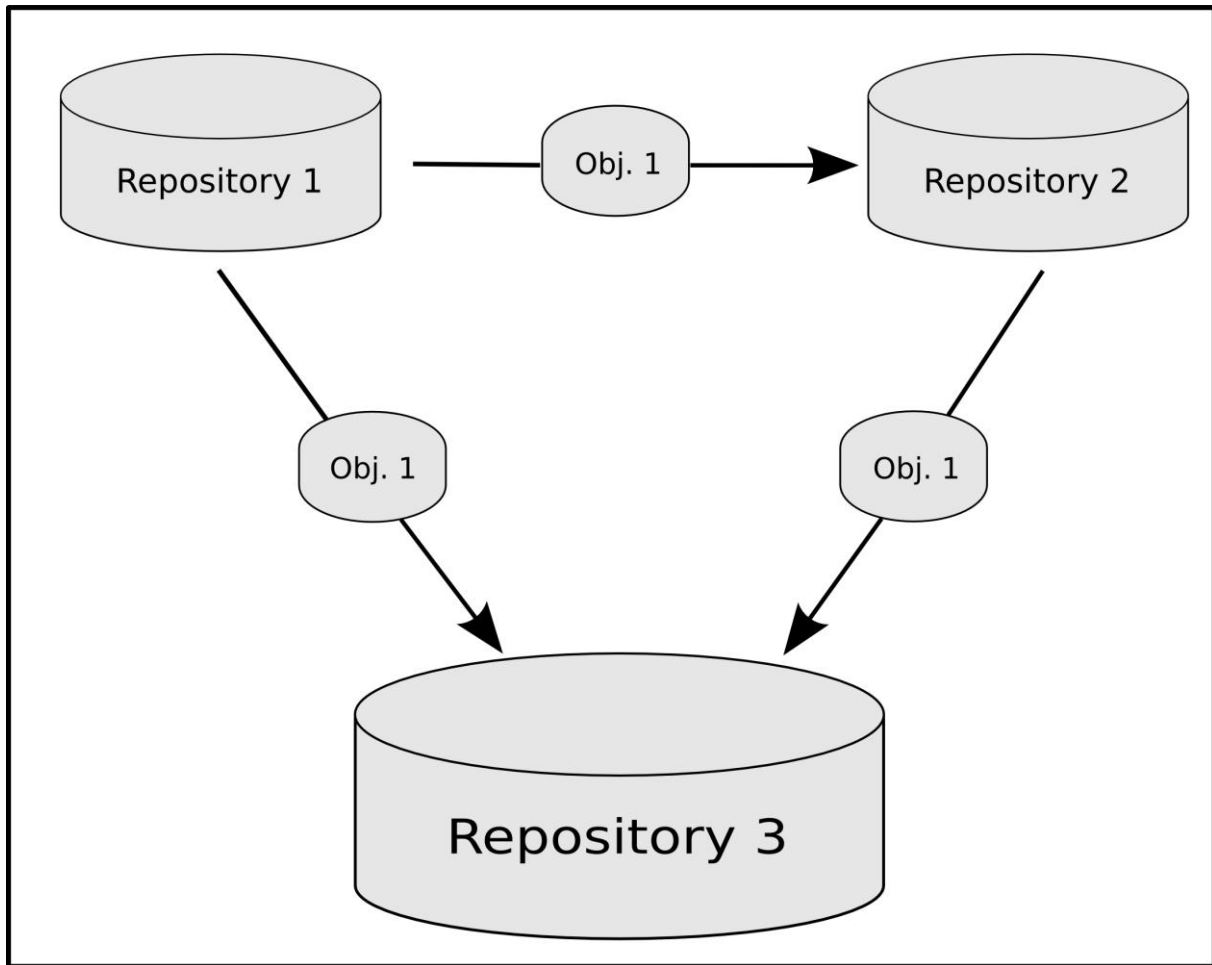
Dit laatste punt is een andere belangrijke reden om persistente identifiers te implementeren binnen een collectie digitale objecten. In de culturele wereld zien we vaak dat archieven, bibliotheken en andere instellingen met een eigen collectie hun gedigitaliseerde materiaal online plaatsen. Meestal gebeurt dit inclusief een manier om de metadata van deze objecten te kunnen ophalen, al dan niet door middel van OAI-PMH (Open Archives Initiative Protocol for Metadata Harvesting). Dit zorgt ervoor dat andere instellingen metadata van dezelfde objecten op hun eigen site beschikbaar kunnen stellen. Deze partijen kunnen vervolgens de metadata zelf ook weer doorgeven aan derden. Hierdoor kan het makkelijk gebeuren dat een object of in ieder geval de metadata daarbij op meerdere sites te vinden en te bekijken is zonder dat het voor de gebruiker onmiddellijk duidelijk is dat het om hetzelfde object gaat. Zeker wanneer er geen mogelijkheid is om te controleren wat de bron of de afkomst is van een digitaal object wordt het in dit soort scenario’s uiteindelijk zeer lastig om dubbelingen te voorkomen.

Er zijn op het web, onder andere door de inzet van OAI-PMH, veel sites te vinden die grote hoeveelheden digitale objecten uit verschillende bronnen binnenhalen en zelf beschikbaar stellen om zo centrale doorzoekbaarheid vanuit een bepaald thema te kunnen realiseren. De educatieve contentketen van Kennisnet bijvoorbeeld, is een aggregaat dat zich richt op het verzamelen van leermateriaal vanuit diverse instellingen in Nederland. Gebruikers kunnen bovendien zelf ook bronnen aandragen om toe te voegen aan deze verzameling digitale objecten. Binnen deze architectuur is de persistente identifier benoemd als de aangewezen manier om dubbelingen te voorkomen, die optreden doordat dezelfde objecten vanuit verschillende bronnen toegevoegd worden.¹⁰ Dit vereist uiteraard wel dat de applicatie die verantwoordelijk is voor het binnenhalen van nieuwe objecten ook ingebouwde logica bezit om te kunnen filteren op basis van persistente identifiers. Het is dan ook van belang dat persistente identifiers zo vroeg mogelijk in de keten worden aangemaakt, bijvoorbeeld voorafgaand aan publicatie van een object, en dat deze in de hele keten door alle partijen worden doorgegeven.

⁸ Berners-Lee 1998

⁹ Hakala 2010

¹⁰ Hamers et. al. 2011, 7



Illustratie 1: Hoe dubbelingen kunnen ontstaan: Repository 3 haalt objecten binnen van repository 1 en 2, maar repository 2 haalt ook objecten binnen van repository 1

Digitale preservering

Persistente identifiers spelen ook een belangrijke rol in het concept van digitale preservering. Digitale preservering heeft als doel om digitale objecten voor de lange termijn te behouden zodanig dat de integriteit en de authenticiteit van het object gewaarborgd blijft. 'Behouden' betekent in dit geval ook dat een digitaal object voor de gebruiker toegankelijk en afspeelbaar moet blijven ook al veranderen in de loop der tijd de onderliggende technieken die nodig zijn voor de juiste representatie van het object.

Het besef dat ook digitaal materiaal onderhevig is aan verval, en de erkenning van de noodzaak om hier maatregelen tegen te nemen, is de afgelopen tien jaar steeds sterker geworden en heeft geleid tot het opstellen van richtlijnen waaraan archieven en instellingen moeten voldoen om onbedoelde wijzigingen aan digitale objecten en verlies tegen te gaan. Het meest gangbare model voor digitale preservering is het functionele model OAIS (Open Archival Information System) waarbinnen abstracte onderdelen en processen van een digitaal archief zijn beschreven met daarbij behorende functies voor het behoud van de digitale objecten op de lange termijn.

De belangrijkste eisen waaraan de instelling moet voldoen om een dergelijke houdbaarheid te kunnen garanderen zijn vastgelegd in de ISO-standaard ISO 16363.

In deze standaard is aangaande identifiers van de objecten vastgelegd dat een instelling ieder object uniek moet kunnen identificeren en dat de instelling deze identifiers ook moet kunnen onderhouden zodat ze doorverwijzen ongeacht de fysieke locatie.¹¹ Juist wanneer het gaat om het toegankelijk houden van een digitaal object voor de lange termijn is het vanzelfsprekend van belang dat het archief ook in staat is om datgene wat het object identificeert eveneens voor de lange termijn te preserven. De wens om een betrouwbare digitale dienstverlening voor de lange termijn te kunnen garanderen kan dus ook een goede reden zijn om persistente identifiers te implementeren.

Versiebeheer

Zoals eerder genoemd is een belangrijk voordeel van de persistente identifier ten opzichte van de URL het feit dat eerstgenoemde versiebeheer mogelijk maakt. Zoals we in het voorgaande al zagen, bieden URL's geen oplossing voor het probleem dat de inhoud van een object kan veranderen. In deze gevallen moet, ofwel de URL aangepast worden om te kunnen blijven verwijzen naar een eerdere versie of de inhoud moet stilzwijgend veranderen waarbij de oude versie dan helemaal niet meer toegankelijk is. Er is natuurlijk ook nog een derde optie mogelijk namelijk dat de nieuwe versie een nieuwe URL krijgt en de oude versie op de oorspronkelijke URL beschikbaar blijft. Maar ook dan nog bestaat het probleem dat er in dat geval geen gestandaardiseerde manier is om de relatie tussen de twee versies aan te duiden. Aangezien persistente identifiers uniek verwijzen naar een bepaald object kan een instelling in het kader van versiebeheer ervoor kiezen om bij een verandering aan het object een nieuwe identifier toe te kennen aan de nieuwe versie. Op deze manier kan naar beide versies van het object uniek verwezen worden en blijven beide versies apart toegankelijk of in ieder geval apart te onderscheiden. De oude versie kan dan nog altijd gevonden worden aan de hand van de oorspronkelijke identifier.¹² Er kan voor gekozen worden om de relatie tussen beide versies te leggen door middel van een bepaalde benaming binnen de syntax van de identifier of door middel van metadata.

Eerstgenoemde strategie is bijvoorbeeld de manier waarop de UK Data Archive omgaat met versiebeheer van persistente identifiers. Binnen deze methode wordt de identifier opgedeeld in een aantal functioneel los te onderscheiden delen waarvan het laatste deel wordt gebruikt om versienummers mee aan te duiden.¹³

Het nadeel van de eerste methode is dat er toch enige afhankelijkheid van semantiek ontstaat bij het interpreteren van identifiers die op een dergelijke manier tot stand zijn gekomen, hetgeen voor mensen op zich geen probleem is, maar voor geautomatiseerde verwerking wel. Wanneer de relatie alleen gelegd wordt in de metadata bestaat deze afhankelijkheid niet. Binnen de uitgebreidere variant van de metadatastandaard Dublin Core, de zogenaamde Qualified Dublin Core, bestaat er bijvoorbeeld de mogelijkheid om naar een nieuwere versie te verwijzen door middel van het element 'isReplacedBy'.¹⁴ Zo kan de relatie tussen versies gestandaardiseerd vastgelegd worden zodat ook binnen applicaties logica ingebouwd kan worden om iets met deze informatie te doen bij de presentatie van objecten.

Overigens is het in beide gevallen van belang dat er tenminste binnen de instelling een eenduidig beleid is opgesteld rondom versiebeheer.

¹¹ Audit and certification of Trustworthy Digital Repositories, 4-9, 4-10

¹² Richards et. al. 2011, 17-18

¹³ Zie bijvoorbeeld de uitleg van dit mechanisme bij Corti 2012

¹⁴ <http://dublincore.org/documents/2012/06/14/dcmi-terms/?v=terms#terms-isReplacedBy>

Dit opdat voor medewerkers en gebruikers te allen tijde duidelijk is in welke gevallen een verandering leidt tot het aanmaken van een nieuwe identifier en in welke gevallen een verandering doorgevoerd kan worden zonder een nieuwe identifier aan te maken. In veel gevallen zal een keuze hieromtrent gebaseerd zijn op de vraag in hoeverre het van belang is voor gebruikers om nog te kunnen verwijzen naar een eerdere versie. Bij het herstellen van een eenvoudige typefout zal dit misschien niet het geval zijn maar bij een betekenisvolle verandering aan de inhoud is dit vaak wel van belang.¹⁵

Relaties

Op een vergelijkbare manier kunnen er door middel van metadata niet alleen relaties gelegd worden tussen versies van objecten maar ook tussen verschillende objecten onderling. Het toekennen van een identifier aan een object dwingt de instelling dan ook om na te denken over wat een object precies is binnen de context van het archief. Is het een geheel of bestaat het eigenlijk uit een samenstelling van losse objecten? Met het oog op het gebruik is het goed om vast te leggen in beleid in welke gevallen aparte onderdelen binnen een object ook apart te identificeren moeten zijn. Er kan dan voor gekozen worden om meerdere persistente identifiers aan te maken per object.

Binnen de Educatieve contentketen van Kennisnet is bijvoorbeeld afgesproken om niet alleen voor het object zelf een persistente identifier aan te maken maar ook voor de metadata van het object, zodat het mogelijk wordt om verschillende soorten metadata te kunnen onderscheiden, die ieder een ander doel dienen maar over hetzelfde object gaan.¹⁶

Bronvermelding

Tenslotte is een belangrijk voordeel van de persistente identifier dat deze bronvermelding eenduidiger maakt. Aangezien digitale objecten op meerdere plaatsen beschikbaar gesteld kunnen worden is het zowel voor de instelling als voor de gebruikers van belang dat naar de meest oorspronkelijke representatie van een object verwezen kan worden. Ervan uitgaande dat de instelling die de digitale objecten beheert ook de instelling is die verantwoordelijk is voor het aanmaken en onderhouden van de persistente identifiers voor haar collectie, kunnen we stellen dat het koppelen van persistente identifiers aan objecten een goede methode is om enige controle te houden over het eigenaarschap van haar digitale objecten op internet. Dit omdat de instelling zo zelf kan bepalen naar welke representatie van het object de gebruiker wordt doorverwezen bij het aanroepen van de persistente identifier op het web. In veel gevallen zal er bijvoorbeeld doorverwezen worden naar het digitale object op de website van de instelling zelf of naar metadata over het object zoals deze door de instelling verbonden is aan het object. Voor de gebruiker draagt deze koppeling tegelijkertijd bij aan het gemak waarmee de oorspronkelijke eigenaar van objecten gevonden kan worden en biedt ze een manier om bronnen op een ondubbelzinnige manier te kunnen identificeren en citeren. We zien dan ook dat persistente identifiers in de wereld van onderzoeksdata en wetenschappelijke publicaties veelal voor dit doeleinde worden ingezet, bijvoorbeeld in het geval van de dienst DataCite die instellingen de kans geeft om via hun dienst persistente identifiers aan te maken voor data.¹⁷

¹⁵ Zie voor een voorbeeld van een dergelijk beleid ook de presentatie van de UK Data Archive: Corti 2012

¹⁶ Hamers et. al. 2011, 6

¹⁷ Zie: <http://www.datacite.org/>

Overeenkomsten en verschillen

Er zijn dus voor instellingen met veel digitale objecten, en een grote verspreiding daarvan op het web, goede redenen om voor het archiefmateriaal persistente identifiers aan te maken. Zoals we al eerder hebben gezien zijn er vele soorten systemen die op een bepaalde manier het concept van de persistente identifier geïmplementeerd hebben. Maar wat is eigenlijk het gemeenschappelijke uitgangspunt van deze technieken? En hoe weten we welk systeem we het beste kunnen kiezen? Om daar achter te komen is het goed om eerst te kijken wat een persistente identifier precies kenmerkt en welke eigenschappen centraal staan binnen de meeste implementaties van dit concept.

Persistentie

Het eerste kenmerk ‘persistentie’ is meteen het meest voor de handliggende kenmerk. Het is niet onmiddellijk duidelijk wat hiermee bedoeld wordt. Betekent dit dat een identifier voor altijd beschikbaar moet zijn? En betekent beschikbaar zijn dan dat deze voor altijd moet doorverwijzen naar een internetpagina? Gezien de snelle ontwikkelingen op internet lijkt dit een bijna onhaalbare eis.

Meestal wordt dan ook gesteld dat de identifier tenminste zolang persistent moet zijn als het object waar het naar verwijst beschikbaar is.¹⁸ Volgens Nicholas is persistentie bovendien een begrip dat niet op zichzelf staat maar dat in relatie tot andere eigenschappen tot uiting moet komen. Bijvoorbeeld als garantie dat de identifier persistent doorverwijst naar de juiste informatie.¹⁹

Ook is het belangrijk om te beseffen dat persistentie niet met techniek alleen opgelost kan worden maar dat er vanuit de instelling de wil moet zijn om persistentie te garanderen door de identifiers gedurende een vastgestelde periode actief te onderhouden zodat deze altijd doorverwijzen naar de juiste informatie. Een dergelijke bereidheid kan bijvoorbeeld tot uiting komen door het belang van persistente identifiers vast te leggen in metadatabeleid. Hierdoor is binnen een instelling duidelijk dat wanneer er technisch iets verandert aan de opbouw van URL's binnen de website van de instelling, ook de verwijzingen bij de persistente identifiers aangepast moeten worden.

Aangezien dit onderhoud altijd een zekere mate van inspanning kost, is het goed om hiermee bij de uitgifte van persistente identifiers al rekening mee te houden. Dit kan door vooraf in beleid vast te leggen welke objecten persistent geïdentificeerd moeten worden en welke niet. Om hierover een zinnige keuze te maken is het dan van belang om het verwachte gebruik van de objecten bij een vastgestelde gebruikersgroep vooraf te inventariseren. In dit beleid zou ook vastgelegd moeten zijn hoe lang het onderhoud aan de identifiers gegarandeerd wordt, zodat gebruikers weten wat ze kunnen verwachten. Voor het idee van persistentie is betrouwbaarheid essentieel.²⁰

In die zin sluit dit idee over persistentie van identifiers ook aan bij de bredere discussie over wat duurzaam behoud van digitale objecten inhoudt. Binnen het internationale model voor digitale duurzaamheid OAIS wordt ook gesteld dat het voor betrouwbaar behoud noodzakelijk is om essentiële eigenschappen van een object te benoemen die onafhankelijk van veranderingen in techniek behouden moeten blijven.

¹⁸ Zie bijvoorbeeld Hakala 2010

¹⁹ Nicholas et. al. 2010

²⁰ Nicholas et. al. 2009

Deze eigenschappen moeten voorts getoetst worden aan een vastgestelde gebruikersgroep, de zogenaamde ‘Designated Community’ die uiteindelijk a) bepaalt of een object nog beschouwd kan worden als authentiek en b) of de diensten die door een instelling geleverd worden betrouwbaar zijn.

Centrale uitgifte, uniciteit en schaalbaarheid

Een ander kenmerk dat eigenlijk alle implementaties van persistente identifiers gemeen hebben is dat er voor het genereren van de identifiers een bepaald mechanisme bedacht is dat moet garanderen dat de identifiers wereldwijd uniek zijn. Juist het wereldwijd unieke karakter van persistente identifiers onderscheidt deze van ‘gewone’ identifiers. Binnen de architectuur van relationele databases is het bijvoorbeeld gebruikelijk om elk record in de database te voorzien van een unieke identifier, de zogenaamde *primary key*. Deze is binnen de context van de database uniek en zou op een bepaalde manier als persistent beschouwd kunnen worden in de zin dat het niet gebruikelijk is deze identifier zomaar te wijzigen. Buiten de context van de database is de identifier gewoon een nummer en bovendien ook niet gegarandeerd uniek. In veel databasesystemen is de identifier of primary key een simpel nummer dat per record automatisch opgehoogd wordt. Wanneer deze identifiers buiten de database worden gepubliceerd, bijvoorbeeld op een website waar ook data uit andere databases toegankelijk wordt gemaakt is de kans dus groot dat de identifiers niet uniek zijn en noodzakelijkerwijs aangepast moeten worden om dubbelingen te voorkomen.²¹

Bij veel systemen van persistente identifiers wordt dit probleem simpelweg voorkomen doordat uitgifte van de identifier, of een deel daarvan, centraal gecoördineerd wordt. Het kan bijvoorbeeld zo geregeld zijn dat elke afzonderlijke organisatie een eigen nummer krijgt dat in elke identifier terugkomt waarmee al op eenvoudige wijze voorkomen wordt dat twee verschillende instellingen hetzelfde nummer aanmaken. Dit eigen nummer van de organisatie wordt binnen veel systemen ook wel aangeduid als een ‘prefix’. Binnen de specificaties van dergelijke systemen voor persistente identifiers is er dan voor gekozen om door middel van een bepaalde opbouw (de syntax van de identifier in combinatie met een centraal register van aangesloten organisaties) ervoor te zorgen dat alle identifiers uit een bepaald systeem op dezelfde manier te interpreteren zijn. De manier waarop uitgifte van de identifiers is geregeld moet ook zorg dragen voor de schaalbaarheid van de identifiers. Als voorbeeld kunnen we het ‘Handle-system’²² noemen. Binnen dit systeem wordt schaalbaarheid gegarandeerd doordat naar extra servers kan worden uitgebreid, zowel wanneer een server een groot aantal aanvragen krijgt te verwerken als in het geval dat een server een zeer grote hoeveelheid identifiers moet opslaan voor een bepaalde partij.²³ Aangezien de software zelf ook de nummers aanmaakt en een register van organisaties bijhoudt, is de uniciteit van identifiers met dit systeem ook gegarandeerd.

Dat uniciteit en schaalbaarheid enkele van de belangrijkste eigenschappen van persistente identifiers zijn, blijkt ook uit het feit dat deze twee eigenschappen expliciet genoemd worden in de richtlijnen voor identifiers zoals deze beschreven zijn binnen het OAIS-model. Om een betrouwbare digitale dienst te leveren moet een archief volgens deze standaard een manier hebben om alle objecten in het digitale archief uniek te kunnen identificeren, voorkomen dat dubbelingen ontstaan en een systeem hebben dat ook met toekomstige hoeveelheden nog afdoende functioneert.

²¹ Richards et al, 7

²² <http://www.handle.net/>

²³ <http://www.handle.net/faq.html>

Implementatie van een systeem voor persistente identifiers kan een instelling dus helpen bij het vervullen van de OAIS-eisen aangezien bij dergelijke systemen al in ontwerp rekening is gehouden met deze eisen.

Echter, niet voor alle systemen geldt dat centrale uitgifte de manier is waarmee schaalbaarheid en uniciteit gegarandeerd worden. PURL²⁴ (Persistent Uniform Resource Locator) bijvoorbeeld werkt door middel van het installeren van software op een eigen server met een eigen domein waarbij degene die de software installeert zelf verantwoordelijk is voor uitgifte van identifiers.

Gerelateerd aan het kenmerk van centrale uitgifte, en voor sommige instellingen zelfs doorslaggevend in de keuze voor een bepaald systeem, is ook de manier waarop centrale uitgifte organisatorisch is ingericht en in hoeverre hier kosten aan verbonden zijn. Ook hier zijn de verschillen tussen de systemen groot. PURL kan zoals hierboven gezegd, licentievrij geïnstalleerd en gebruikt worden. DOI²⁵ (Digital Object Identifier) daarentegen wordt beheerd door The International DOI Foundation die ook de organisaties aanwijst die de uitgifte van DOI-prefixen beheren. Deze organisaties kunnen op hun beurt weer kosten berekenen aan organisaties die DOI's willen aanmaken.²⁶ DOI is afgeleid van Handle en ook bij Handle worden er kosten berekend voor het in stand houden van een centraal register van organisaties met bijbehorende prefixen.

Doorverwijzen

Een ander wezenlijk kenmerk van persistente identifiers is dat ze doorverwijzen naar een digitaal object of naar metadata over het object. Deze eigenschap hangt samen met een van de meest in het oog springende voordelen van persistente identifiers namelijk dat ze het 'rotten' van links op internet zouden moeten voorkomen. Het kunnen doorverwijzen, bijvoorbeeld door te verwijzen naar een internetpagina, is ook iets dat een persistente identifier voor heeft op het gebruik van een 'gewone' identifier zoals bijvoorbeeld een unieke code in een database. In zulke gevallen is het doorverwijzen naar een digitaal object iets dat is vastgelegd in de record zelf, afhankelijk van hoe de database is ingericht.

Persistente identifiers daarentegen staan los van de manier van opslag van de data zelf en zijn ontworpen om door te verwijzen naar externe objecten. Wanneer we stellen dat het kunnen doorverwijzen een essentiële eigenschap is van persistente identifiers betekent dat ook dat bepaalde systemen voor het identificeren van objecten feitelijk niet beschouwd kunnen worden als systemen voor *persistente* identifiers (hoewel ze misschien wel een systeem bieden voor het genereren van wereldwijd unieke identifiers).

Een voorbeeld is Universally Unique Identifier (UUID). Dit systeem biedt een methode voor het genereren van een identifier die in de praktijk als wereldwijd uniek beschouwd kan worden doordat de kans dat iemand precies dezelfde identifier per ongeluk genereert zeer klein is. Dit systeem biedt echter geen ingebouwde mogelijkheid om de identifier door te laten verwijzen naar een object of data over een object.²⁷

Maar wanneer we stellen dat doorverwijzen een essentiële eigenschap is kunnen we ons afvragen welke manier van doorverwijzen dan voldoet. Verschillende systemen voor persistente identifiers hebben deze vraag op verschillende manieren beantwoord.

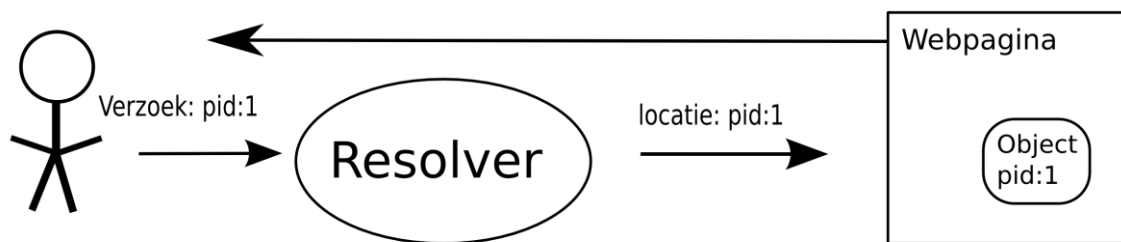
²⁴ <http://purl.oclc.org>

²⁵ <http://www.doi.org/>

²⁶ <http://www.doi.org/faq.html>

²⁷ Richards et. al. 2011, 7-8

In de praktijk is HTTP natuurlijk de meest voor de hand liggende invulling voor deze eigenschap, omdat identifiers vaak gekoppeld worden aan digitale objecten die op het internet te vinden zijn. Dat de identifier dus ook doorverwijst met behulp van internettechniek is in dat geval een logische keuze. Systemen zoals PURL zijn dan ook gebouwd op dezelfde technieken als internetpagina's. PURL maakt gebruik van DNS voor adressering van de identifier en gebruikt HTTP-redirects voor het doorverwijzen naar objecten of metadata over objecten. Hierin ligt ook meteen de beperking: als de domeinnaam om een of andere reden toch een keer moet veranderen dan werken de identifiers ook niet meer, net zoals bij andere internetpagina's. Bij het ontwikkelen van het Handle-systeem voor persistente identifiers is er bewust voor gekozen om een systeem te ontwerpen dat onafhankelijk is van DNS.²⁸ Er zijn weliswaar zogenaamde 'resolvers' die te benaderen zijn op een domeinnaam zoals <http://hdl.handle.net> maar dit is bewust bedacht als een extra toevoeging en niet als een afhankelijkheid besloten in de architectuur van het systeem.



Illustratie 2: Schematische weergave van hoe het doorverwijzen vaak werkt: een identifier wordt door de resolver omgezet in een locatie naar een digitaal object zodat de gebruiker het object kan ontvangen.

Semantiek of niet?

Een vraagstuk rondom persistente identifiers waarover de meningen verdeeld zijn, is of persistente identifiers semantiek mogen bevatten of niet. Zo ja, dan volgt meteen de vraag hoeveel semantiek dan acceptabel is. Duidelijk is wel dat eventuele semantiek vooral onveranderlijke kenmerken van het object moet betreffen. Het eerdergenoemde voorbeeld van de UK Data Archive waarbij versienummers verwerkt worden in de identifier kan bijvoorbeeld gezien worden als een acceptabele afhankelijkheid van semantiek. Maar een begrip als status (dat hier feitelijk heel dicht tegen aan ligt) is in de meeste gevallen geen semantische waarde die in de identifier vastgelegd moet worden, aangezien deze veranderlijk is. Dit is in ieder geval binnen het concept van Cool URI's een waarde die volgens de schrijver vermeden dient te worden.²⁹

Dat er geen overeenstemming is over of en hoeveel semantiek acceptabel is, blijkt ook uit de manier waarop de verschillende systemen omgaan met dit principe. De twee meest extreme posities hier worden ingenomen door de eerder genoemde implementaties die vanwege het ontbreken van een mechanisme voor centrale uitgifte met moeite tot persistente identifiers gerekend kunnen worden, namelijk Cool URI's en UUID's. Cool URI's bevatten veel semantiek omdat de identificering plaatsvindt aan de hand van leesbare URL's met een domeinnaam, een pad en een bestandsnaam die ook nog semantiek kan bevatten. UUID's daarentegen bevatten geen enkele semantiek aangezien het hier slechts een willkeurig gegenereerd nummer betreft dat verder op geen enkele manier afhankelijkheid kent met een inhoudelijk aspect van het object waar het indirect aan gekoppeld is.

²⁸ http://www.handle.net/overviews/system_fundamentals.html#dns

²⁹ Berners-Lee 1998.

De overige (meestgebruikte) systemen voor persistente identifiers zitten tussen deze twee posities in. Handles zijn bijvoorbeeld wel gekoppeld aan een organisatie door middel van een nummer, zodat Handles van een bepaalde organisatie voor een deel van de identifier eenzelfde nummer hebben. Het nummer is echter niet semantisch in die zin dat het niet hoeft te veranderen als de organisatie van naam verandert. Afhankelijk van de specifieke implementatie is dit bij PURL weer anders. Aangezien PURL's afhankelijk zijn van een bepaalde domeinnaam voor de doorverwijzing, kan het zijn dat wanneer een organisatie op zijn eigen domein een PURL-resolver heeft draaien, de identifier aangepast moet worden als de organisatie van naam verandert. Ook in het geval dat de algemene PURL-resolver op purl.org wordt gebruikt, gelden voor het aanmaken en onderhouden van PURL's veel van dezelfde principes als voor Cool URI's namelijk dat er goed nagedacht moet worden over de opbouw van de identifiers om te voorkomen dat deze afhankelijk worden van veranderlijke semantische betekenis. Als goed voorbeeld kunnen misschien de PURL's van het Dublin Core metadataschema genoemd worden, zoals die voor het veld datum:

<http://purl.org/dc/elements/1.1/date>

We zien dat hier verwezen wordt naar de algemene resolver op purl.org, dat een afkorting van de naam van het metadataschema wordt gebruikt als hoofdniveau, dat de verschillende elementen gegroepeerd zijn onder een subniveau en dat het versienummer verwerkt is in de opbouw om te voorkomen dat aanpassingen aan de velden van het metadataschema leiden tot aanpassingen aan de identifiers. Er zijn dus duidelijk alleen semantische waardes opgenomen die naar alle waarschijnlijkheid niet meer hoeven te veranderen.

In het algemeen kan gesteld worden dat de meeste systemen voor persistente identifiers in grotere of kleinere mate een afhankelijkheid kennen van inhoudelijke betekenissen die verbonden zijn aan de objecten die ze identificeren. Waar deze betekenis duidelijk zichtbaar is in de identifier zelf, zal de organisatie een goed doordacht beleid moeten formuleren voor de opbouw van identifiers zodat er een minimale afhankelijkheid bestaat van veranderlijke inhoudelijke kenmerken.

Vergelijking meest voorkomende PI-systemen

Zonder uitgebreid in te gaan op de ontstaansgeschiedenis, architectuur, ontwikkeling en mogelijkheden van alle systemen voor persistente identifiers zoals deze de afgelopen decennia zijn ontwikkeld, zouden we op basis van bovengenoemde kenmerken een vergelijking kunnen maken van de verschillende systemen zoals deze hierboven aan bod zijn gekomen. Het gaat hier om een globaal overzicht op basis waarvan de belangrijkste verschillen en overeenkomsten tussen de systemen overzichtelijk kan worden gemaakt.

	globaal uniek	centrale uitgifte	doorverwijzen	kosten	semantiek
Handle	Ja	ja	ja	ja	*
DOI	Ja	ja	ja	*	*
PURL	*	nee	ja	nee	ja
UUID	*	nee	nee	nee	nee
Cool URI	*	nee	nee	nee	ja
URN-NBN	ja	ja	*	nee	*
*afhankelijk van implementatie					

Feitelijk geeft bovenstaande tabel een licht vertekend beeld van de situatie in de zin dat sommige kwalificaties meer eenduidigheid suggereren dan in werkelijkheid bestaat. Anders gezegd: de ingevulde waarden zijn in strikte zin van toepassing, maar er zijn uitzonderingen te bedenken waarbij bepaalde PI-systemen geïmplementeerd zijn op een manier die deze kwalificaties zou veranderen. Zo bestaan er bijvoorbeeld implementaties die UUID's gebruiken als deel van een identifier in een 'resolver' zodat de UUID's wel degelijk doorverwijzen naar informatie over een object.³⁰ De specificatie van UUID zelf voorziet echter niet in deze functionaliteit. Aan de tabel is te zien dat met name Handle en DOI de meeste eigenschappen bezitten die in dit artikel gehanteerd worden als kenmerkend voor persistente identifiers. Als we stellen dat de eigenschap om door te verwijzen noodzakelijk is om een systeem te kunnen beschouwen als een implementatie van het concept 'persistent identifier', zien we ook dat Cool URI's en UUID's strict genomen niet voldoen aan de eisen.

Persistente identifiers in Europees verband

Zoals hierboven genoemd biedt het implementeren van persistente identifiers voordelen voor instellingen met objecten die ook in digitale vorm op het web te vinden zijn. Maar hoeveel van dergelijke instellingen binnen Europa hebben feitelijk een PI-systeem geïmplementeerd? Naar deze vraag is onderzoek gedaan binnen het Driver-project van SURFfoundation. Binnen het meest recente onderzoek dat is gedaan in 2008 bleek dat maar liefst 84.3 procent van de ondervraagde instellingen persistente identifiers had toegekend aan alle documenten onder hun beheer.³¹ Binnen de ondervraagde instellingen zijn de meest gebruikte PI-systemen Handle, URN, DOI en PURL met respectievelijk 42,7%, 35,3%, 15,3% en 12,7% van de ondervraagde instellingen.³²

Sinds die tijd zijn er ook overkoepelende initiatieven ontstaan waaruit blijkt dat er ook sinds 2008 nog steeds ingezet wordt op de techniek van persistente identifiers. Een dergelijk initiatief is bijvoorbeeld Persid. Binnen dit Europese project werken instellingen uit acht verschillende landen samen aan een internationale infrastructuur die URN:NBN-identifiers kan doorverwijzen naar de verschillende lokale organisaties die deze beheren.³³

Momenteel is het nog zo dat resolvers van URN-identifiers per land kunnen verschillen. De URN:NBN-resolver van Nederland is bijvoorbeeld ingericht voor het doorverwijzen van URN:NBN-identifiers uit Nederland, afkomstig van wetenschappelijke instellingen oftewel URN's van het type URN:NBN:NL:UI.³⁴ De URN-resolver van Finland daarentegen (<http://urn.fi>) kan zowel nationale URN:ISBN- als URN:NBN-identifiers doorverwijzen. De Duitse NBN-resolver kan naast URN ook DOI doorverwijzen. Nationale resolvers hebben dus verschillende mogelijkheden en zijn op verschillende url's te vinden.

³⁰ Het 3TU.Datacentrum heeft dit bijvoorbeeld gerealiseerd, zie: <http://datasupport.researchdata.nl/start-de-cursus/iv-gebruiksfase/data-citeren/persistent-identifiers-bij-3tudatacentrum-en-dans/>

³¹ Van Der Graaf 2009, p. 61

³² Ibid. p. 64

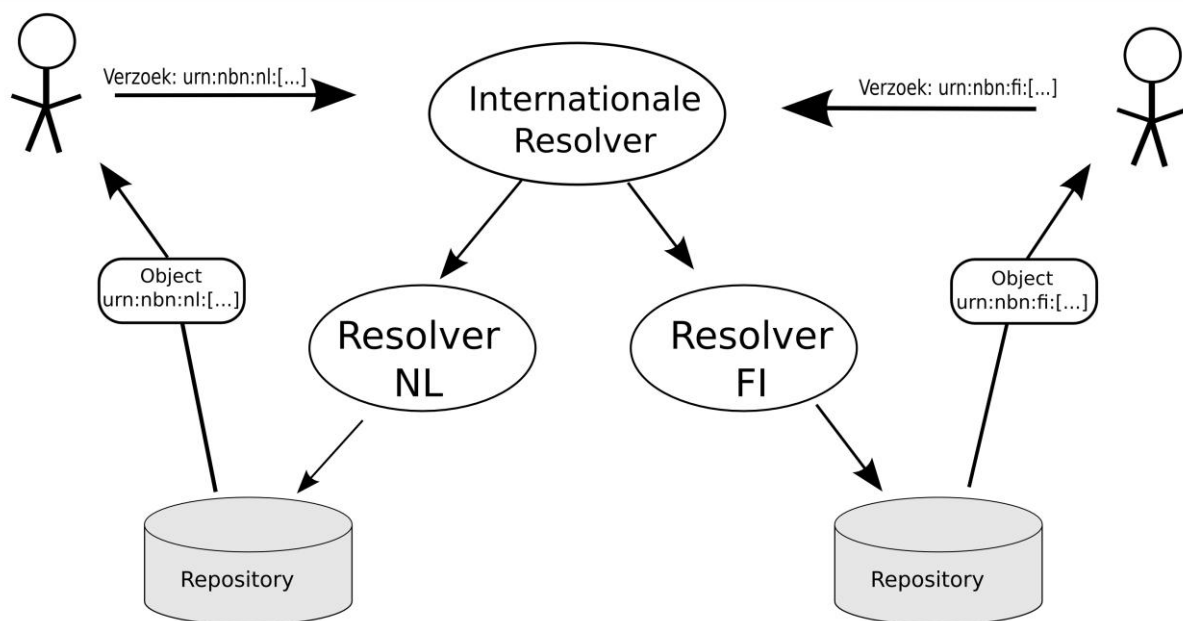
³³ <http://www.persid.org/>

³⁴ <http://persistent-identifier.nl>

Binnen Nederland is de Koninklijke Bibliotheek als nationale ‘Registration Agency’ verantwoordelijk voor het ‘NL’-domein van URN:NBN. Zij is daarmee ook verantwoordelijk voor het toekennen van ‘community-id’s’ (zoals ‘UI’ voor wetenschappelijke instellingen) en voor het toekennen van ‘repository-id’s’ voor de individuele instellingen die tot een bepaalde community behoren. DANS (Data Archiving and Networked Services) is verantwoordelijk voor de nationale resolver.³⁵

Zoals te zien is aan de verschillen tussen resolvers binnen verschillende landen is het van belang dat er internationale samenwerking is op het gebied van resolvers zodat gebruikers van URN:NBN-identifiers niet zelf hoeven te bedenken uit welk land een bepaalde URN afkomstig is om vervolgens de URL van de resolver te zoeken. Ook is er behoefte aan een resolver om identifiers uit verschillende PI-systemen met elkaar uitwisselbaar te maken, bijvoorbeeld DOI’s en URN:NBN’s. Een dergelijk initiatief is de ‘Europeana Resolution Discovery Service’ een prototype voor een zogenaamde ‘meta-resolver’ die in staat is om zowel URN:NBN-identifiers als DOI’s naar de juiste resolver door te verwijzen.³⁶

Een ander initiatief is de NBN-resolver van de Deutsche Nationalbibliothek die momenteel in staat is om URN:NBN-identifiers uit verschillende Europese landen door te verwijzen naar de nationale resolvers en bovendien ook in staat is om onder andere Handles en DOI’s door te verwijzen.³⁷



Illustratie 3: Schematische weergave van hoe een internationale urn:nbn-resolver kan werken: de internationale resolver herkent uit welk land een urn:nbn-identificer komt en stuurt deze door naar de nationale resolver die deze weer doorverwijst naar de repository waar het object te vinden is.

³⁵ <http://www.kb.nl/expertise/voor-bibliotheken/registration-agency-nbn>

³⁶ http://www.europeanaconnect.eu/documents/02_Europeana_Resolution_Discovery_Service.pdf

³⁷ <http://nbn-resolving.org>

Usecase Nederlands Instituut voor Beeld en Geluid

Binnen verschillende Europese samenwerkingsprojecten voor archiefmateriaal op internet zijn deelnemende organisaties tegenwoordig verplicht om hun objecten te voorzien van een persistente identifier, meegeleverd in de beschrijvende metadata. Dit was ook een eis vanuit het project Clarin-nl³⁸ waaraan Beeld en Geluid deelneemt. Clarin is een project waarbij een infrastructuur is ontwikkeld voor onderzoeksbronnen op het gebied van taal.

De bijdrage van Beeld en Geluid bestond uit het via een OAI-PMH-provider toegankelijk maken van online video's voor onderwijs en onderzoek. OAI-PMH is een protocol waarbij metadata in XML-formaat op een gestandaardiseerde manier kan worden uitgeleverd en opgevraagd. Als systeem voor persistente identifiers werd gekozen voor Handle. De betrouwbaarheid van centrale uitgifte gecombineerd met de onafhankelijkheid van semantiek waren enkele van de factoren die bij deze besluitvorming een rol speelden. Er werd tevens voor gekozen om zowel de objecten als de specifieke bestanden zoals deze op internet stonden, apart te identificeren door middel van Handles. Dit betekende dat voor elk record in de catalogus een Handle werd aangemaakt en dat voor elk bestand dat op internet beschikbaar was gemaakt nog een extra Handle werd aangemaakt. Hierdoor werd - voor elk mediabestand uit de onderwijscollectie - zowel een Handle gecreëerd voor de media zelf, als voor de record in het algemeen. Deze keuze was mede ingegeven door de manier waarop besloten was doorverwijzingen te implementeren. De Handle voor het record verwees door naar de metadata van de individuele record zoals deze beschikbaar was in de OAI-PMH-provider, terwijl de Handle voor het mediabestand doorverwees naar een pagina op de videoportal waar het bestand af te spelen was voor onderwijsinstellingen met een licentie.

Dat deze methode een voordeel had boven rechtstreeks verwijzen naar een afspeelpagina bleek toen de videoportal naar een nieuw domein verhuisde; na een kleine aanpassing van de verwijzingen binnen de Handle-service verwezen alle identifiers door naar de nieuwe locatie zonder dat er aan de identifiers zelf iets hoefde te veranderen.

Persistente identifiers vs. Linked Data

Natuurlijk zijn er ook een aantal beperkingen te benoemen aan het concept en de techniek van persistente identifiers. Deze beperkingen zijn enerzijds inherent aan de eigenschappen van deze techniek zoals hierboven beschreven. Ze hebben anderzijds te maken met nieuwe ontwikkelingen binnen de wereld van het web waarvan het niet onmiddellijk duidelijk is hoe deze te verenigen zijn met de uitgangspunten van persistente identifiers. Een voorbeeld van het laatste is bijvoorbeeld de ontwikkeling van 'Linked Data'. Linked Data is het concept om data die op internet op een open manier beschikbaar gemaakt is, aan elkaar te linken zodat er een web van aan elkaar verbonden data ontstaat waarbinnen relaties benoemd kunnen worden. Binnen de techniek van persistente identifiers is wel beschreven dat objecten uniek geïdentificeerd kunnen worden maar er wordt niet direct een manier geboden om objecten aan elkaar te linken en om relaties tussen objecten te kunnen beschrijven. Linked Data staat bovendien op gespannen voet met persistente identifiers omdat binnen het concept van Linked Data gekozen is om objecten te identificeren door middel van HTTP URI's. Zoals hierboven beschreven zijn persistente identifiers juist in het leven geroepen omdat de techniek van HTTP en URL's niet werd gezien als een afdoende manier om objecten uniek te kunnen

³⁸ <http://www.clarin.nl>

identificeren. De voorstanders van Linked Data daarentegen zien vaak niet het nut in van een apart schema voor persistente identifiers dat geheel los staat van HTTP, vooral wanneer deze toch vaak terugvallen op de techniek van HTTP voor het regelen van doorverwijzingen. Ze stellen dat veel van de zaken waar PI-systemen een oplossing voor moeten bieden, waaronder persistentie, ook mogelijk zijn met HTTP-URI's.³⁹ Zoals eerder gezegd erkennen voorstanders van het gebruik van persistente identifiers ook dat echte persistentie uiteindelijk meer gelegen is in het beleid van een instelling en de organisatorische inspanning die wordt gestoken in het onderhouden van doorverwijzingen dan dat persistentie altijd gegarandeerd is binnen de architectuur van de techniek zelf.

In andere aspecten lijken de twee groepen simpelweg uiteenlopende doelstellingen te hebben; persistente identifiers bevatten doorgaans zo min mogelijk semantiek zodat veranderingen niet noodzakelijk zijn terwijl binnen Linked Data juist voor mensen leesbare URI's de voorkeur hebben.

De twee technieken zijn ook organisatorisch behoorlijk verschillend. Linked Data is, net als het web zelf, volledig gedecentraliseerd terwijl persistente identifiers juist vaak in ieder geval ten dele steunen op een instantie die de uniciteit garandeert en uitgifte van nummers en/of prefixen regelt. Ondanks deze verschillen zijn er ontwikkelingen te zien waarbij er een aanzet wordt gedaan om een brug te slaan tussen deze twee technieken zodat de voordelen van beide technieken optimaal benut kunnen worden. Een van de nadelen van PI-systemen is zoals eerder gezegd dat er zoveel verschillende soorten zijn, die op afwijkende manieren werken zodat ze onderling niet makkelijk uitgewisseld kunnen worden. Het is bovendien mogelijk dat hetzelfde object meerdere soorten identifiers heeft, bijvoorbeeld een PURL en een Handle, zonder dat voor de gebruiker meteen duidelijk is dat het om hetzelfde object gaat. Dit is een bezwaar dat met behulp van Linked Data opgelost kan worden door de relatie tussen de twee objecten op een gestandaardiseerde manier vast te leggen. Dit punt is een van de bevindingen van de 'Den Haag Persistent Object Identifier – Linked Open Data-manifesto', een initiatief om uitgangspunten te beschrijven die een wisselwerking tussen PI-systemen en Linked Open Data mogelijk moeten maken.⁴⁰

Wanneer in de toekomst dergelijke initiatieven verder uitgewerkt worden kan dit beide technieken ten goede komen; de betrouwbaarheid van gecentraliseerde uitgifte van persistente identifiers met de structuur en de uitwisselbaarheid van Linked Data.

Bronnen:

Audit and certification of Trustworthy Digital Repositories, 2011, <http://public.ccsds.org/publications/archive/652x0m1.pdf>

Barbara Bazzanella, Stefano Bortoli, Paolo Bouquet, "Can persistent identifiers be cool?" in: International Journal of Digital Curation 2013, Vol. 8, No. 1, pp. 14-28, <http://dx.doi.org/10.2218/ijdc.v8i1.246>

T. Berners-Lee, Cool URIs don't change, 1998, <http://www.w3.org/Provider/Style/URI.html>

³⁹ Thompson et al. 2006

⁴⁰ Zie: <http://www.knowledge-exchange.info/Default.aspx?ID=462>

Louise Corti, [UKDA use of persistent identifiers](http://www.bl.uk/aboutus/stratpolprog/digi/datasets/workshoparchive/LouiseCortin_Identifier_sForTheUKDA_May2012.pdf): Case study, 2012, http://www.bl.uk/aboutus/stratpolprog/digi/datasets/workshoparchive/LouiseCortin_Identifier_sForTheUKDA_May2012.pdf

D5.4.1 Europeana Resolution Service, 2010, http://www.europeanaconnect.eu/documents/02_Europeana_Resolution_Discovery_Service.pdf

M. Van Der Graaf, *The European Repository Landscape 2008 : Inventory of Digital Repositories for Research Output*, Amsterdam University Press, 2009, <http://dx.doi.org/10.5117/9789089641908>

Juha Hakala, "Persistent identifiers: an overview", 2010, <http://metadaten-twr.org/2010/10/13/persistent-identifiers-an-overview/>

Jeroen Hamers, Wim Muskee, Edwin Verwoerd, Marjan Frijns en Jasper Roes, "Afspraak Unieke Persistente Identifier voor Leermateriaal en Metadatarecord", 2011, http://www.edustandaard.nl/fileadmin/contentelementen/kennisnet/EduStandaard/Documenten/Afspraak_Unieke_Persistente_Identifier_voor_Leermateriaal_en_Metadatarecord_v1.0.pdf

Hans-Werner Hilse and Jochen Kothe, "Implementing Persistent Identifiers: Overview of concepts, guidelines and recommendations", 2006, <http://nbn-resolving.de/urn:nbn:de:gbv:7-ISBN-90-6984-508-3-8>

Nick Nicholas, Nigel Ward, Kerry Blinco, "A Policy Checklist for Enabling Persistence of Identifiers" in: *D-Lib Magazine* January/February 2009 Volume 15 Number 1/2, <http://www.dlib.org/dlib/january09/nicholas/01nicholas.html>

Nick Nicholas, Nigel Ward, Kerry Blinco. "Abstract Modelling of Digital Identifiers" in: *Ariadne* Issue 62 January 2010, <http://www.ariadne.ac.uk/issue62/nicholas-et-al/>

RFC 1373: <http://www.ietf.org/rfc/rfc1737.txt>

RFC: 3188: <http://tools.ietf.org/html/rfc3188>

Kevin Richards, Richard White, Nicola Nicolson, Richard Pyle, GBIF (2011). *A Beginner's Guide to Persistent Identifiers*, version 1.0. Released on 9 February 2011. Copenhagen: Global Biodiversity Information Facility, 33 pp, accessible online at <http://www.gbif.org/resources/2575>

K. Sollins, L. Masinter, "RFC 1737: Functional Requirements for Uniform Resource Names", 1994, <http://www.ietf.org/rfc/rfc1737.txt>

Henry S. Thompson, David Orchard, *URNs, Namespaces and Registries*, 2006, <http://www.w3.org/2001/tag/doc/URNsAndRegistries-50-2006-08-17>

Emma Tonkin. "Persistent Identifiers: Considering the Options". July 2008, *Ariadne* Issue 56 <http://www.ariadne.ac.uk/issue56/tonkin/>