

PORTAALTOOLS: SLEUTEL TOT EEN UNIEK TOEGANGSSCHERM?



BERT VANHALST

1. Inleiding

1.1. Probleemstelling

Doorgaans gebruiken medewerkers binnen een instelling een groot aantal verschillende toepassingen om hun taken uit te voeren. Deze toepassingen zijn gebaseerd op verschillende technologieën en hebben telkens een andere look & feel en een ander gedrag. Zo moet de gebruiker zich bijvoorbeeld ten opzichte van elke toepassing apart authenticeren. Bovendien is er weinig tot geen integratie tussen de toepassingen en moeten gegevens bijgevolg manueel overgetikt of gekopieerd worden. Ook is het geheel van toepassingen niet afgestemd op de specifieke rol van de gebruiker.

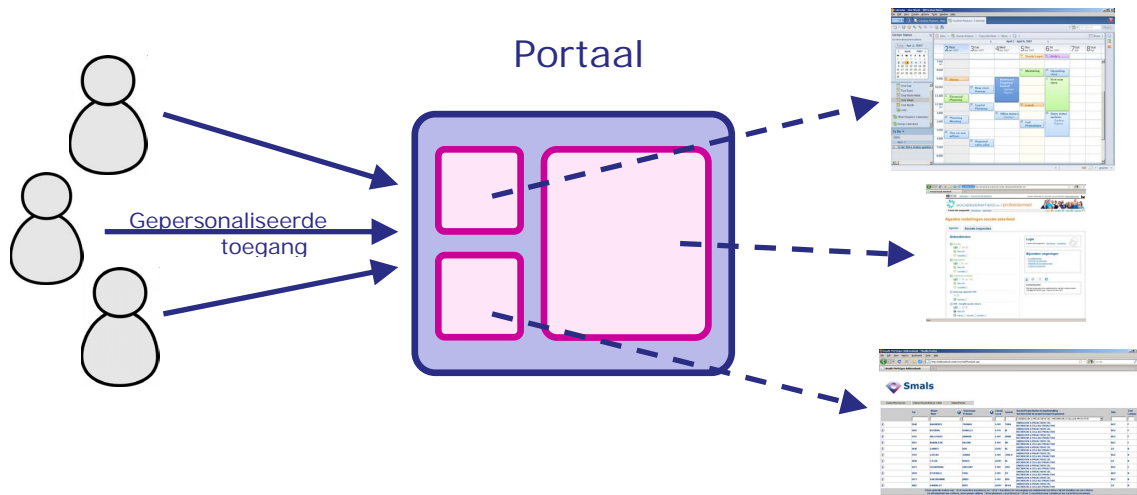
Voor velen zal deze situatie heel herkenbaar zijn. Een portaal kan een oplossing bieden voor deze ongemakken.

Een portaal is een unieke, beveiligde en gepersonaliseerde toegangspoort tot informatie en toepassingen.

De definitie is schematisch weergegeven in Figuur 1. Links op de figuur zien we de gebruikers die een beveiligde toegang krijgen tot het portaal. Het portaal zelf biedt op uniforme wijze toegang tot de verschillende toepassingen en informatiebronnen. Daarbij kan de inhoud en vormgeving van het portaal afgestemd worden op de gebruikersgroep of de identiteit en de rol van de individuele gebruiker.

Een portaal kan gericht zijn op verschillende doelgroepen:

- Interne werknemers: dit is de typische intranet-omgeving die toegang biedt tot toepassingen, documenten en informatie;
- Partner-instellingen (extranet): het creëren van een virtuele organisatie over de grenzen van de eigen organisatie heen;
- Ondernemingen en burgers (internet).



Figuur 1: Schematisch overzicht van de definitie van een portaal

Los van de doelgroepen kan een portaal ook verschillende gebruiksscenario's ondersteunen:

- Samenwerkingsplatform: uitwisselen van informatie, samenwerken aan documenten, voeren van discussies, enzovoort.
- Self-service: hier is het doel van het portaal om de gebruiker een aantal taken zelf te laten uitvoeren of informatie te verschaffen zodat minder een beroep moet gedaan worden op andere medewerkers. Denk aan HRM- of CRM-toepassingen.
- Uitvoeren van taken: hier ligt de nadruk op het aanbieden van een individuele takenlijst met aandacht voor de integratie van de toepassingen die nodig zijn om die taken uit te voeren.

De winst die men verkrijgt door het gebruik van een portaal kunnen we enerzijds zien vanuit het oogpunt van de gebruiker (medewerker, burger, ambtenaar, partner) en anderzijds vanuit het oogpunt van de aanbieder van een portaal (overheidsinstelling, Smals, iedere medewerker die betrokken is bij het opzetten en beheren van het portaal en zijn inhoud).

Intuïtief kunnen we zeggen dat een portaal de gebruikerservaring een stap vooruit helpt. Een aantal zaken kunnen het leven van de gebruikers heel wat eenvoudiger maken:

1. gebruikers vinden informatie en toepassingen op één unieke plaats;
2. gebruikers hoeven zich slechts eenmaal aan te melden om de verschillende toepassingen te gebruiken of informatie op te zoeken;
3. manueel overtypen van informatie kan vermeden worden door het automatisch doorgeven van informatie tussen toepassingen;
4. de inhoud is toegespitst op de specifieke rol van de gebruiker waardoor hij/zij enkel ziet wat voor hem/haar van belang is.

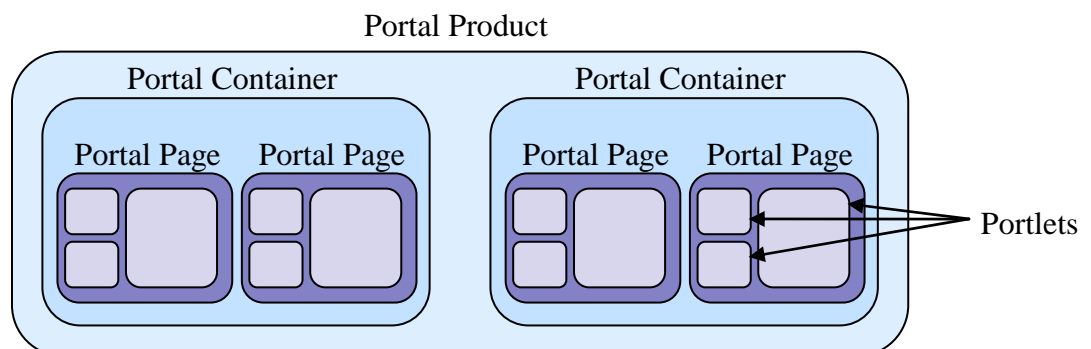
We kunnen dus stellen dat een portaal de productiviteit helpt verbeteren, iets wat de interesse van het management zeker kan wegdragen. Die productiviteitsverhoging is zeker van belang bij het uitbouwen van een *enterprise portal*, een portaal gericht op de interne medewerkers.

Het concept "portaal" spreekt tot de verbeelding van heel wat gebruikers en beslisningnemers. In dit rapport wordt dan ook nagegaan in welke mate portaaltools een oplossing bieden voor het gestelde probleem en dus tegemoetkomen aan dit ideaalbeeld van wat een portaal zou kunnen zijn.

1.2. Concepten

Vooraleer we de functionaliteiten van portaaltools bespreken, lichten we nog de belangrijkste concepten en termen toe.

Een portaalproduct bestaat uit een **portal container**. Technisch gezien is dat een toepassing die op een applicatieserver draait en een verzameling basisfunctionaliteiten aanbiedt. Uit Figuur 2 kan afgeleid worden dat de mogelijkheid bestaat om meerdere instanties van een portal container te draaien. Dat kan handig zijn voor load balancing en fail-over of voor het aanbieden van een intern en extern portaal die beide gebaseerd zijn op hetzelfde portaalproduct.



Figuur 2: Hiërarchische structuur van een portaal

Een portaal zelf is georganiseerd in pagina's die een set van componenten bevatten. Die componenten worden over het algemeen portlets genoemd, maar worden soms ook omschreven als web parts (Microsoft), gadgets, webmodules, blocks of connectors.

Portlets zijn de eigenlijke bouwstenen van een portaal die toegang bieden tot informatie en toepassingen. Bij een portaaltool is over het algemeen een catalogoog meegeleverd met portlets, bijvoorbeeld voor de authenticatie van gebruikers, zoekfunctionaliteit of het tonen van nieuwsberichten. Daarnaast bevatten portaaloplossingen ook de nodige ontwikkeltools om zelf portlets te ontwikkelen voor het integreren van specifieke toepassingen of informatie.

Een portaal kan een aantal subportalen bevatten, bedoeld voor bepaalde belangengroepen.

In de volgende paragraaf gaan we verder in op de verschillende functionaliteiten die zogenaamde *horizontale portaaltools* bieden. Een horizontaal portaal maakt integratie mogelijk tussen verschillende toepassingen zoals een content-managementsysteem, een zoekrobot en zelf-ontwikkelde toepassingen. Een verticaal portaal daarentegen integreert sterk met één bepaalde functie of één bepaald product, bijvoorbeeld SAP of Documentum.

2. Functionaliteiten van portaaltools

Al een hele tijd zijn er portaaltools op de markt die de basisfunctionaliteit leveren om een uniek toegangsscherm aan te bieden. In dit hoofdstuk bespreken we de functionaliteiten die we terugvinden in de meeste horizontale portaaltools.

2.1. Unieke toegang

Voor de portaaloperator is een unieke omgeving of architectuur of een uniek raamwerk voor het publiceren van nieuwe toepassingen en informatie eenvoudiger te beheren. Een uniforme technologie kan de volledige levenscyclus van een toepassing (analyse, ontwerp, ontwikkeling, ontplooiing en onderhoud) sterk vereenvoudigen en men moet slechts één keer de leerfase doorlopen om de functionaliteit die het portaal biedt onder de knie te krijgen.

Ook voor de gebruiker is een uniek toegangspunt tot alle voor hem relevante toepassingen en informatie zeker een vooruitgang, en duplicatie van informatie wordt vermeden. De grote schaduwzijde ligt meestal in de beperkingen van de webinterface. Mogelijkheden om de interface te verrijken moeten zeker worden onderzocht (zie verder).

Een standaard manier van werken wekt vertrouwen bij de gebruiker en beperkt de leerfase die hij moet doorlopen indien nieuwe functionaliteit wordt toegevoegd. Ook voor de aanbieder is standaardisatie een belangrijke winstfactor. De uniforme manier van werken kan in processen worden vastgelegd zodat iedereen kan bijdragen tot de informatie die wordt aangeboden op het portaal.

Workflows die gebruik maken van een integratieplatform en gebruikersinteractie vragen kunnen op een natuurlijke wijze worden geïntegreerd met een portaal.

Bemerk echter dat één toegangspunt tot alle toepassingen en informatie utopisch is. Heel wat gebruikers zijn best tevreden met de bestaande manier van werken op basis van clients met een rijke gebruikersinterface. Men moet de functionaliteit die men via een portaal aanbiedt gradueel opbouwen en enkel mogelijkheden toevoegen die echte meerwaarde bevatten.

Hierna bespreken we meer in detail welke bronnen ontsloten kunnen worden en op welke manier.

2.2. Integratie van toepassingen

Eén van de kerntaken van een portaaltool is toegang verschaffen tot toepassingen en informatie. Een portaal kan gezien worden als een integratietool waarbij de integratie “aan het glas” gebeurt, dit wil zeggen op het niveau van de gebruikersinterface en niet in de back-end.

Zoals in de inleiding beschreven, verloopt die integratie via portlets die moeten gezien worden als een venster op bestaande toepassingen en informatiebronnen. Integratie met back-endsystemen is mogelijk via de klassieke integratiemechanismen: API's, database-toegang, webservices, enzovoort. De nodige aandacht moet uitgaan naar het ontsluiten van de back-end. Daarbij is het belangrijk om een duidelijke scheiding te maken tussen de gebruikersinterface en de back-end. Een portaal is in die zin complementair met een service-gerichte architectuur (SOA): het portaal wordt de gebruikersinterface voor de toegang tot de services beschikbaar in een SOA-architectuur.

Het is aangeraden om zoveel mogelijk integratieproblemen op te lossen in de back-end, niet op niveau van het portaal. Zo kunnen aanpassingen aan het portaal beperkt worden bij upgrades en migraties van het portaal.

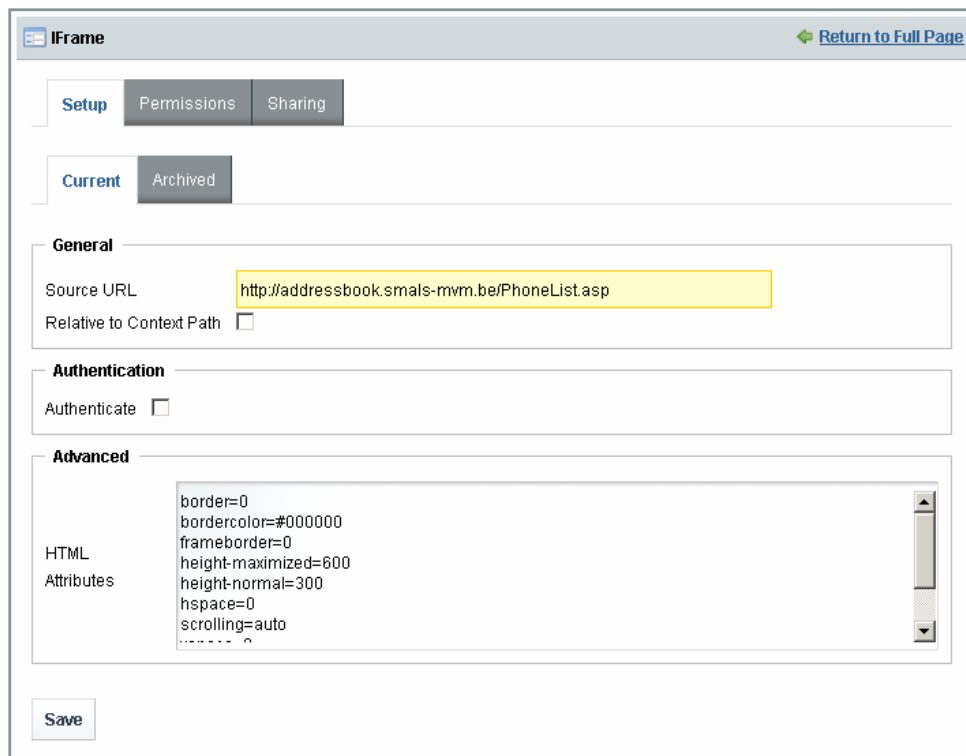
Integratie van webapplicaties

Integratie van webapplicaties kan gebeuren via het zogenaamde web-clippingmechanisme. Dat houdt in dat een bestaande webapplicatie getoond wordt in een portlet op basis van de URL van de webapplicatie. Technisch gezien wordt de bestaande webapplicatie dan getoond in een iframe.

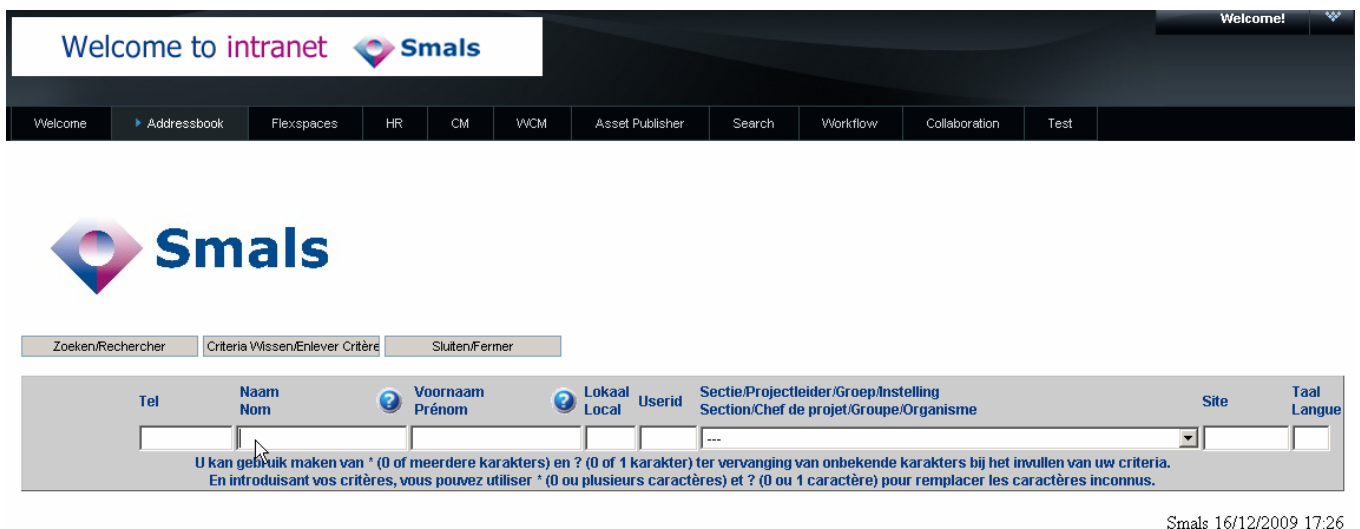
Om te vermijden dat we uit het kader van de portaalomgeving springen bij het aanklikken van een URL in de webapplicatie, kunnen de URL's herschreven worden door het portaal. Dit mechanisme heet URL rewriting.

Bijkomend bestaat de mogelijkheid om een subset van een bestaande pagina te tonen in een portlet aan de hand van specifieke HTML-tags. Web clipping portlets laten typisch ook toe om de gebruiker te authenticeren ten opzichte van de webtoepassing op basis van basic of form-based authenticatie.

Als voorbeeld zien we in Figuur 3 hoe we de bestaande adresboek-webapplicatie van Smals in Liferay Portal integreren. We hoeven enkel de URL van het adresboek op te geven. Het resultaat is te zien in Figuur 4.



Figuur 3: Configuratie van de iFrame portlet in Liferay Portal



Figuur 4: Integratie van de bestaande adresboek-applicatie via de iFrame portlet in Liferay Portal

Integratie van rich clients

In de praktijk zijn niet alle bestaande toepassingen webgebaseerd. We moeten ook rekening houden met zogeheten *rich clients*. Hiermee bedoelen we desktoptoepassingen met een rijke gebruikersinterface die verbinding maken met een back-end server. Rich clients kunnen

gebaseerd zijn op diverse technologieën, waaronder Java Swing¹, Adobe Flex/Air², Microsoft Silverlight³, PowerBuilder⁴ en WinDev⁵.

Toepassingen in Flex/Air en Silverlight kunnen eenvoudig geïntegreerd worden in een portaal. Toepassingen in Java Swing kunnen geïntegreerd worden in een portaalpagina door ze om te vormen tot een Java Applet. Dit houdt een kleine aanpassing in van de code. Bij toepassingen in PowerBuilder en WinDev daarentegen moet de front-end herschreven worden op basis van portlets.

Het is al meteen duidelijk dat de integratiekosten voor deze categorie van toepassingen al snel heel groot worden. De inspanningen kunnen weliswaar gefaseerd worden: de toepassingen kunnen stap voor stap geïntegreerd worden in de portaalomgeving.

We willen ook opmerken dat rich clients dikwijls offline werken mogelijk maken. Dat impliceert dat ze lokaal bestanden bijhouden op het werkstation van de gebruiker. Op dit ogenblik bestaat er nog geen goede ondersteuning om een dergelijke functionaliteit te voorzien in een webgebaseerde portaalomgeving. Mogelijk biedt HTML 5⁶ hier in de toekomst een oplossing, maar op dit ogenblik is dat nog niet matuur.

2.3. Inter-portletcommunicatie (IPC)

De term inter-portletcommunicatie (IPC) verwijst naar het mechanisme om gegevens uit te wisselen tussen portlets. Dit mechanisme is standaard aanwezig in de portaaltools en is ondertussen gestandaardiseerd voor Java-gebaseerde portalen (zie paragraaf 3.3). Als een gebruiker bijvoorbeeld klikt op de naam van een onderneming in een eerste portlet, kan het ondernemingsnummer doorgestuurd worden naar een tweede portlet die op basis daarvan details toont van de onderneming zoals het adres en de economische activiteit. Zo hoeft de gebruiker niet telkens gegevens over te typen of te kopiëren tussen toepassingen.

Dergelijke uitwisseling van gegevens kan niet alleen tussen portlets op dezelfde pagina plaatsvinden, maar ook tussen portlets op verschillende pagina's.

Een interessante toepassing van IPC is integratie tussen een persoonlijke takenlijst en de toepassingen die de gebruiker nodig heeft om die taken uit te voeren. In een workflow-systeem of Business Process Management systeem (BPMS) kunnen verantwoordelijken aangeven wie welke taken moet uitvoeren. Op basis daarvan kan een persoonlijke takenlijst

¹ http://nl.wikipedia.org/wiki/Swing_%28Java%29

² <http://www.adobe.com/nl/products/flex/> en <http://www.adobe.com/products/air/>

³ <http://www.microsoft.com/silverlight/>

⁴ <http://www.sybase.be/products/modelingdevelopment/powerbuilder>

⁵ <http://www.windev.com/>

⁶ <http://www.w3.org/TR/html5/>

opgesteld en gepresenteerd worden in het portaal. Als een gebruiker een taak selecteert, kan het portaal automatisch de toepassing tonen waarin de gebruiker de taak kan uitvoeren. Daarbij kunnen relevante gegevens die gekoppeld zijn aan de taak (zoals een ondernemingsnummer of een adres) vooraf ingevuld worden in de toepassing.

Opmerking: het uitwisselen van gegevens tussen portlets kan dan wel standaard ondersteund zijn in portaaltools, het is een stuk moeilijker om die gegevens daarna door te geven aan een bestaande webapplicatie die via web clipping geïntegreerd wordt in het portaal. Mits bijkomende ontwikkeling op niveau van het portaal kunnen parameters op twee manieren doorgegeven worden aan webapplicaties:

1. Via URL-parameters. Hieronder is een voorbeeld gegeven van het toevoegen van een parameter Empl_ID aan een URL:
http://www.socialsecurity.be/wrep/rep.do?Empl_ID=650420731
2. Door het automatisch laten invullen van parameters in een formulier van de webapplicatie.

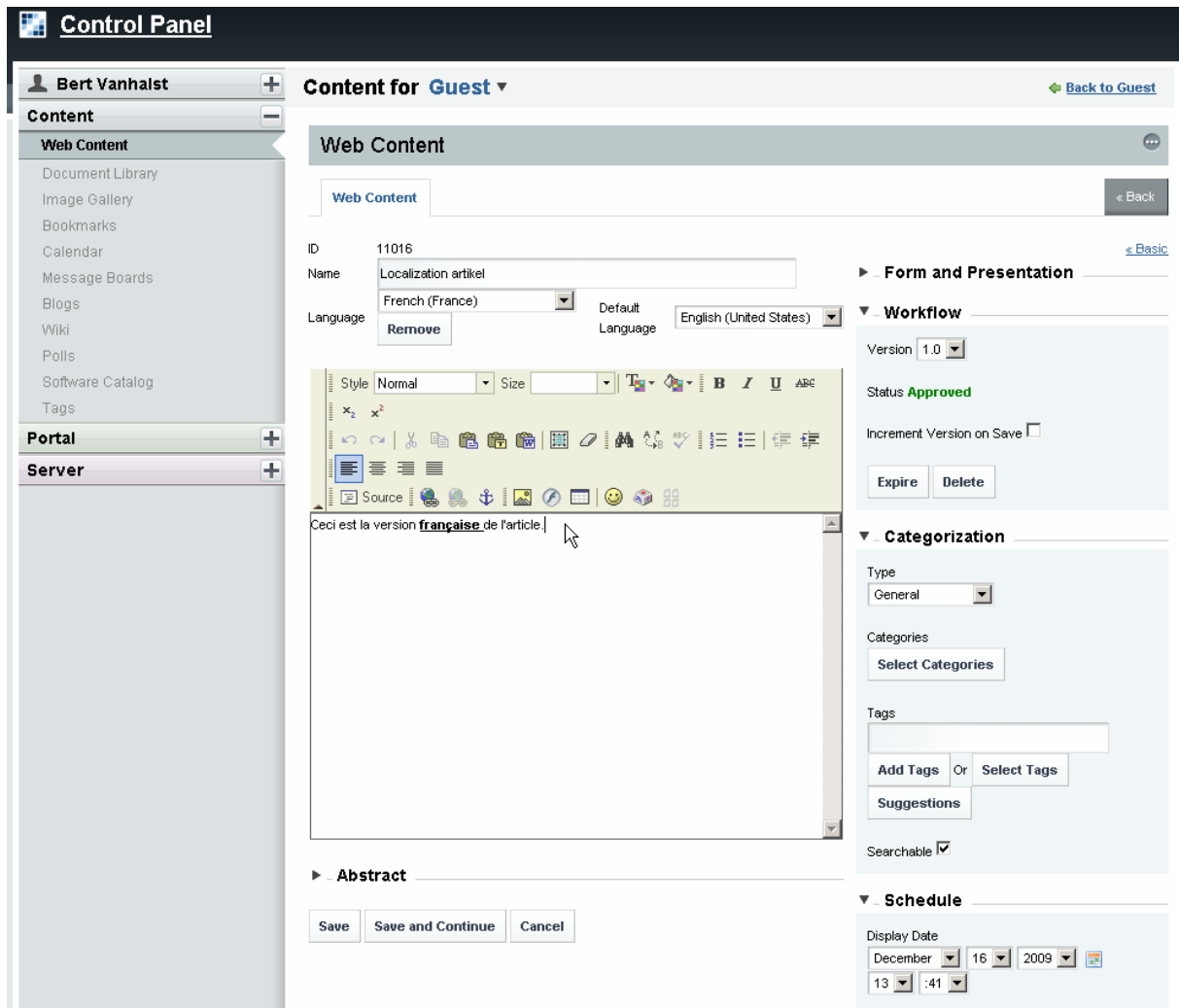
2.4. Contentbeheer en collaboratie

Een belangrijke functionaliteit van een portaal is het verschaffen van toegang tot content en collaboratiefunctiefunctionaliteit.

Bepaalde portaaltools leveren ingebouwde basisfunctionaliteit voor contentbeheer. Zo voorziet bijvoorbeeld Liferay Portal⁷ basisfunctionaliteit voor web content management (WCM), waaronder:

- een rich text editor voor het editeren van artikels vanop een centrale plaats of rechtstreeks in de pagina waar het artikel getoond wordt aan de gebruikers;
- de mogelijkheid om verschillende taalversies van een artikel centraal te beheren;
- de mogelijkheid om artikels aan te maken op basis van templates;
- een eenvoudig workflow-mechanisme voor het verkrijgen van een goedkeuring voor de publicatie;
- gedelegeerde publicatie: de verantwoordelijke van een subportaal of community kan zelf content publiceren binnen die context;
- de mogelijkheid om de publicatie van een artikel op voorhand in te plannen.

⁷ Liferay Portal werd binnen de sectie Onderzoek van Smals getest. Meer informatie is terug te vinden in het rapport "[Open Source Product Review – Liferay Portal 5.2.3](#)", december 2009, Bert Vanhalst, Sectie Onderzoek, Smals.



Figuur 5: Ingebouwde WCM-functionaliteit in Liferay Portal

Indien deze basisfunctionaliteit voldoet, moet er niet meer geïntegreerd worden met een complementair product van de leverancier of een extern pakket.

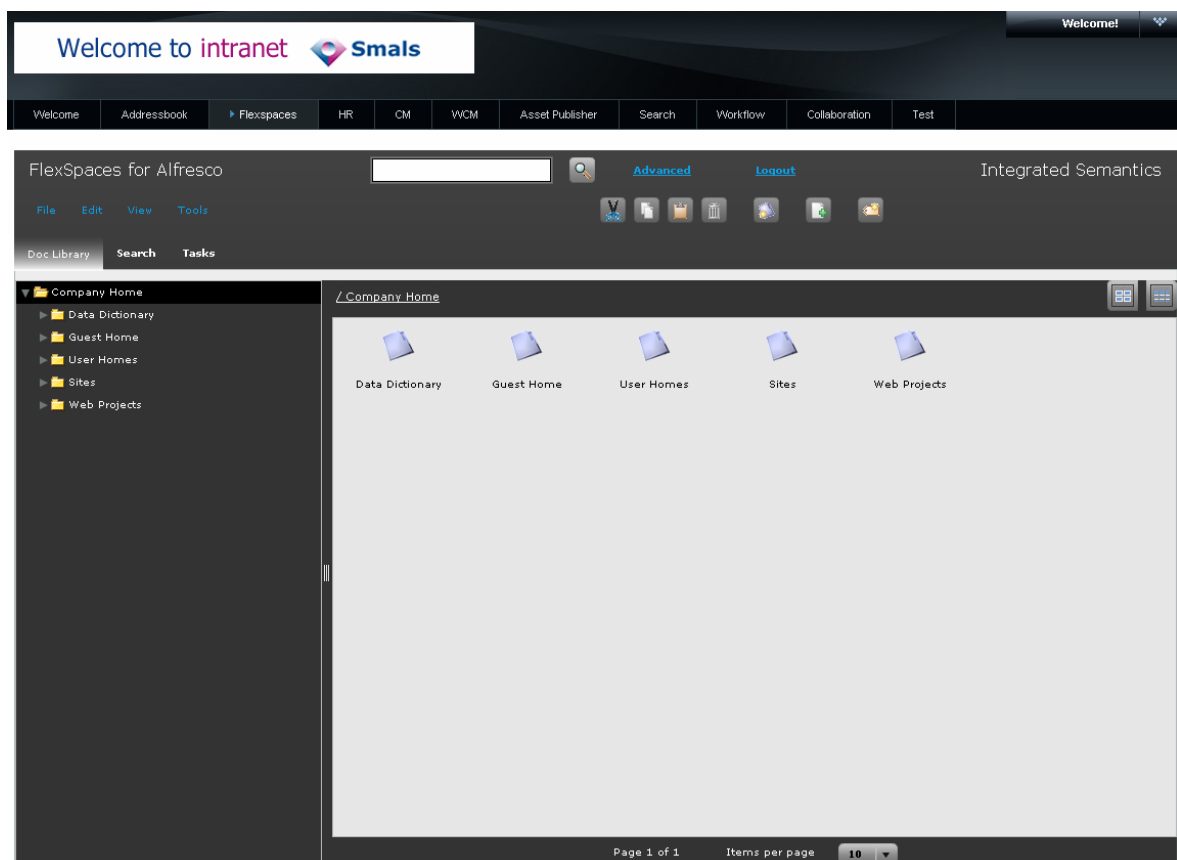
Dikwijls is er echter al een content-managementsysteem (CMS) in gebruik binnen de organisatie en is het dus een kwestie om dit te integreren in de portaalomgeving. De integratie van CMS-functionaliteit en de integratie van de content zelf kan op verschillende manieren gebeuren:

- via portlets die aangeleverd worden door de leverancier van het content-managementsysteem of een externe partner;
- via JCR (Java Content Repository): een gestandaardiseerde Java API voor toegang tot content repositories;
- via REST (Representational State Transfer), een integratiemechanisme gebaseerd op webprotocols;

- via WebDAV, een uitbreiding van het HTTP-protocol die toelaat om op afstand documenten aan te maken en aan te passen.

Bij de portlet-aanpak beschikken we al meteen over een gebruikersinterface. Indien we zelf een gebruikersinterface willen of moeten bouwen, dan kunnen we een beroep doen op de drie andere mechanismen.

Als voorbeeld is in Figuur 6 te zien hoe in Liferay Portal toegang kan gekregen worden tot het content-managementsysteem Alfresco⁸. De integratie werd gerealiseerd op basis van een portlet (FlexSpaces⁹) aangeleverd door Integrated Semantics.



Figuur 6: Integratie van Alfresco CMS in Liferay Portal op basis van de Flexspaces portlet

Gelijkaardig kunnen we via portlets de functionaliteit van collaboration tools in het portaal opnemen. Het is echter zo dat toegang tot e-mail in het portaal de volledige deskoptoepassing zoals Outlook of Lotus Notes niet

⁸ Meer informatie over Alfresco is te vinden in de Open Source Product Review "[Alfresco – Alternative for Enterprise Content Management](#)", Hervé Haut en Arnaud Hulstaert, september 2008, Sectie Onderzoek, Smals.

⁹ <http://forge.alfresco.com/projects/flexspaces/>

vervangt als primaire interface. Portlets leveren typisch een subset van de functionaliteit zoals de laatste ongelezen e-mails of de eerstvolgende afspraken in de kalender.

2.5. Beveiliging en single sign-on (SSO)

Een uniek toegangspunt tot gepersonaliseerde informatie betekent inherent dat men een mechanisme moet voorzien om deze informatie op een beveiligde manier te kunnen consulteren, dat men niet steeds gebruikersnamen en paswoorden moet voorzien om verder te gaan met zijn taken en dat men van de ene toepassing naar de andere kan gaan in een uniforme omgeving waarbij het systeem alle voorgaande handelingen kan bijhouden en gebruiken om de volgende ervaring te verrijken. Het portaal biedt hier geavanceerde mogelijkheden en kan integreren met bestaande oplossingen zoals LDAP, CAS (Central Authentication Service) en NTLM.

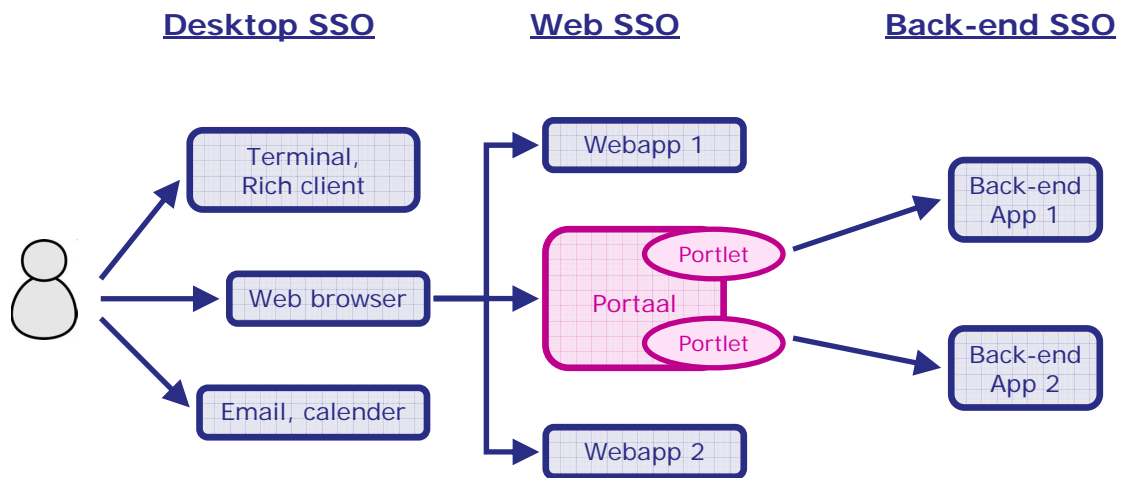
Huidige beveiligingssystemen zijn gebaseerd op rollen en een beveiligingsbeleid. Rollen worden toegekend aan een gebruiker op basis van de groep waartoe de gebruiker behoort en op basis van de waarde van bepaalde van zijn persoonlijke attributen. Deze informatie wordt gebruikt voor toegangscontrole. Toepassingen kunnen gebruik maken van dit beveiligingsmechanisme om transacties toe te laten of niet, of om bepaalde informatie te beschermen en te beperken tot bepaalde gebruikers. Voor de provider biedt dit een uniforme interface voor het beheren van beveiligingsinformatie.

Unieke authenticatie, ook single sign-on¹⁰ (SSO) genoemd, kan toegepast worden op verschillende niveaus.

1. Desktop SSO zorgt ervoor dat een gebruiker zich maar één keer moet aanmelden om toegang te krijgen tot verschillende desktoptoepassingen (terminal-clients, browsers, rich clients, e-mailtoepassingen, enzovoort).
2. Web SSO betekent specifieke SSO voor webgebaseerde toepassingen, waarvan het portaal er één is.
3. Back-end SSO verzorgt de authenticatie ten opzichte van de toepassingen en informatiebronnen die beschikbaar zijn via het portaal, eenmaal de gebruiker zich aangemeld heeft aan het portaal.

De belangrijkste vorm van SSO bij portaaltools is back-end SSO waarbij het portaal namens de gebruiker de authenticatie verzorgt ten opzichte van de achterliggende systemen.

¹⁰ Voor meer informatie over SSO, zie "[Desktop Single Sign-On / Enterprise Single Sign-On](#)", juli 2007, Bob Lannoy, Sectie Onderzoek, Smals en "[Le Single Sign On: Le problème de l'authentification unique](#)", maart 2004, Michel Laloy, Sectie Onderzoek, Smals.



Figuur 7: Verschillende niveaus van single sign-on (SSO)

Voor eenvoudige authenticatie op basis van een gebruikersnaam en paswoord kan het portaal de authenticatie verzorgen in naam van de gebruiker. Daarvoor dient de gebruiker één keer zijn gebruikersnaam en paswoord op te geven. Het portaal slaat deze credentials dan op in een zogeheten credential vault. Dit is een beveiligde opslagplaats waar de credentials geëncrypteerd bijgehouden worden en die enkel toegankelijk is voor het portaal. Naast individuele credentials van een gebruiker kunnen ook gemeenschappelijke credentials bijgehouden worden. Die worden dan op voorhand geconfigureerd door de portaaladministrator.

In bepaalde gevallen is het voor het portaal niet mogelijk om de authenticatie van een gebruiker te simuleren. Voorbeelden zijn niet-webgebaseerde applicaties en webapplicaties waarbij het authenticatiemechanisme gebruik maakt van redirectie (zoals bij het portaal van de sociale zekerheid). In die gevallen moet een specifieke integratie uitgevoerd worden met de externe authenticatiesystemen.

Er moet opgemerkt worden dat een automatische authenticatie ten opzichte van het portaal in combinatie met back-end SSO door het portaal een risico inhoudt. In dit geval staat toegang tot het werkstation gelijk aan toegang tot alle toepassingen en informatiebronnen die via het portaal toegankelijk zijn. Het is dus uiterst raadzaam om een automatische screenlock in te stellen op niveau van het besturingssysteem.

2.6. Personalisatie

Personalisatie betreft meer dan de mogelijkheid om het uitzicht of de vorm te kunnen aanpassen aan de wens van de gebruiker. Personalisatie moet toelaten het portaal van de gebruiker zodanig te organiseren dat het een aangename en uitnodigende werkomgeving wordt. De inhoud die gepubliceerd wordt, kan op basis van personalisatieregels aangepast worden aan de gebruiker om een optimale interactieve ervaring te bieden. Personalisatie moet opgezet worden vanuit het standpunt van de

gebruiker en de organisatie. Het mag zeker niet als invasief worden ervaren en moet op elk ogenblik aanpasbaar en herroepbaar zijn. Een organisatie moet haar identiteit kunnen etaleren via een portaal door het leggen van specifieke accenten (welke toepassingen beschikbaar stellen, specifieke navigatie, taxonomie, beheer, enzovoort).

Alle informatie die over de gebruiker en zijn context aanwezig is, kan gebruikt worden om de gebruikerservaring te verrijken. Dit kan zijn door hem mogelijkheden te bieden om in contact te treden met andere gebruikers die in eenzelfde segment thuishoren. Zoekopdrachten kunnen gepersonaliseerd worden en kunnen de tijd reduceren die nodig is voor het bekomen van resultaten. Links naar deskundigen en collega-gebruikers kunnen inzichten opleveren die primordiaal zijn voor de gebruiker. Toepassingen kunnen gebruik maken van de beschikbare persoonlijke gegevens om het aantal interacties te minimaliseren, bijvoorbeeld het vooraf invullen van formulieren.

Aanpassingen aan de taal en de rol van de gebruiker maken deel uit van de personalisatie. Met een portaal wordt dit mogelijk gemaakt op een gestandaardiseerde manier. Commerciële portaaltools beschikken doorgaans over een rules engine die toelaat om complexe personalisatieregels op te stellen.

Personalisatie belangt niet enkel de gebruiker aan. De provider bezit met personalisatie de mogelijkheid om de klant optimaal te bedienen en zo een grotere klantenbinding en klantentevredenheid te bekomen.

Over het algemeen zal een administrator bij het opzetten van een portaal een aantal algemene personalisatieregels doorvoeren op niveau van een groep of categorie van gebruikers (departement, jobfunctie, rol), niet zozeer op individueel niveau.

Daarnaast kan een administrator al dan niet de toelating geven aan gebruikers om verdere personalisatie door te voeren. Zo kan een gebruiker wegklicken wat hij niet interessant vindt, andere portlets toevoegen of de lay-out van een pagina aanpassen door het verplaatsen van portlets. De administrator kan er evenwel voor kiezen om bepaalde elementen uit te sluiten van personalisatie zodat ze altijd op het scherm verschijnen.

2.7. Zoekfunctionaliteit

Aangezien een portaal verschillende toepassingen en informatiebronnen integreert in één omgeving, kan het interessant zijn om die bronnen als gebruiker vanop één centrale plek te kunnen doorzoeken.

Een portaal kan een ingebouwde zoekmotor bevatten of kan integreren met een externe zoekmotor. Die zoekmotor verzorgt de indexatie van de verschillende bronnen (content-managementsystemen, websites, bestandssystemen, ...). Typisch worden er portlets aangeleverd voor het formuleren van een query en het tonen van de zoekresultaten. De zoekresultaten kunnen gepersonaliseerd worden in die zin dat een gebruiker enkel die resultaten te zien krijgt waar hij/zij ook effectief toegang toe heeft.

2.8. Web 2.0

Als portaaltools en Web 2.0¹¹ samenkomen, is het liefde op het eerste gezicht. Portaaltools omarmen de Web 2.0-wereld op een drietal vlakken:

1. Social computing

Mensen zijn sociale wezens en gedijen pas indien ze de kans hebben om met anderen te praten, te overleggen en om raad te vragen. Deze basisbehoefte kan gestimuleerd worden door een portaal. Een portaal moet toelaten om groepen of gemeenschappen te definiëren en de beschikbare inhoud en toepassingen aan te passen aan de noden van de groep.

Social computing tools zoals blogs, wiki's, social networking, social bookmarking en tagging ondersteunen het samenwerken op basis van computersystemen. Meer en meer worden dergelijke functionaliteiten ingebakken in portaaltools of geïntegreerd via complementaire producten.

2. Mashups

Mashup tools moeten eindgebruikers toelaten om eenvoudige samengestelde toepassingen in elkaar te steken. Uit eigen onderzoek¹² merken we dat toch een technische basiskennis vereist is om dergelijke toepassingen te maken. In Figuur 8 is een voorbeeld te zien van een mashup waarbij gemeentestatistieken gevisualiseerd worden op een kaart. Portaaltools leveren ofwel ingebakken functionaliteit voor het creëren van mashups, ofwel werken ze samen met complementaire producten waarbij mashups eenvoudig geïntegreerd kunnen worden in het portaal. Mashups zijn een alternatief voor portletgebaseerde integratie. De mashup widgets zijn conceptueel vergelijkbaar met portlets, maar zijn doorgaans gebaseerd op een propriëitair mechanisme.

3. Rijke gebruikersinterface

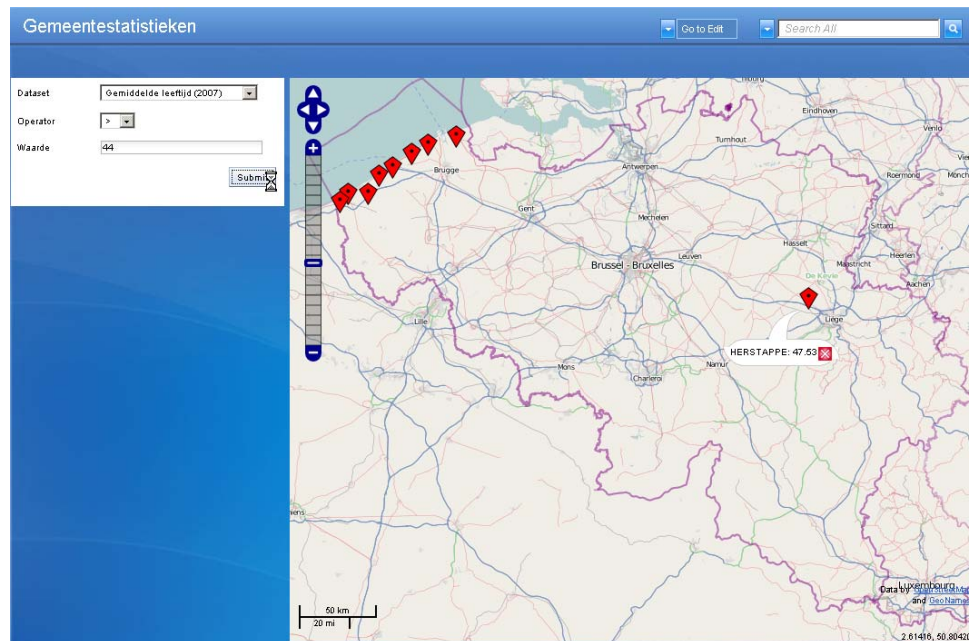
Met de opkomst van RIA's¹³ (Rich Internet Applications) is veel aandacht uitgegaan naar rijkere en responsievere gebruikersinterfaces. Gebruikers zijn niet langer tevreden met de eenvoudige webinterfaces waar de pagina volledig terug ingeladen moet worden bij elke actie. De kracht en gebruiksvriendelijkheid van desktopapplicaties worden met RIA-technologieën (zoals Adobe Flex/Air, Microsoft Silverlight en Ajax toolkits) nu ook beschikbaar in een webomgeving. Ook portaaltools bieden nu

¹¹ "Web 2.0: Enterprise 2.0 – Government 2.0", januari 2009, Sectie Onderzoek, Smals.

¹² Voor een neerslag van dit onderzoek, zie Quick Review "IBM Mashup Center 1.1 – Business Mashup Platform", juli 2009, Bert Vanhalst, Sectie Onderzoek, Smals.

¹³ "[Rich Internet Applications: bruikbare en optimaal toegankelijke applicaties](#)", juni 2008, Adelbert Groebbens, Sectie onderzoek, Smals.

dergelijke rijke interfaces aan en komen zo tegemoet aan de verzuchtingen van de gebruikers.



Figuur 8: Voorbeeld van een mashup: visualisatie van gemeentestatistieken op een kaart

In het algemeen kunnen portaaltools een goede basis zijn om ervaring op te doen met Web 2.0 tools.

De pure web 2.0 tools zijn heel sterk in wat ze doen. Maar als we die tools willen gebruiken als basis voor een portaal, moeten we rekening houden met hun beperkingen. Zo hebben ze een gebrek aan personalisatiemogelijkheden, beschikken ze slechts over beperkte integratiemogelijkheden (meestal propriëitair) en bieden ze nauwelijks ondersteuning voor contentbeheer.

3. Implementatie van een portaal

3.1. Administratie

De administratie van een gedistribueerde omgeving vraagt een enorme inspanning en kan problemen geven op het vlak van uniformiteit en consistentie. Een portaal biedt hier een oplossing door de administratie op te delen in verschillende niveaus. Bovenaan staan de administrators van het portaal die zorgen voor de beschikbaarheid en consistentie van de opbouwende componenten van het portaal (portlets, themes, ...). De eventuele subportalen, bedoeld voor bepaalde belangengroepen, kunnen door gedelegeerde administrators worden beheerd. Zij zijn beter in staat

in te spelen op de specifieke noden van hun gebruikerssegment en kunnen kiezen uit de ter beschikking gestelde componenten. Natuurlijk worden zij ook nauw betrokken bij het bepalen van de benodigde componenten.

De meeste portaaltools bieden een centrale administrator-interface aan. Via die interface kunnen de (gedelegeerde) administrators een aantal zaken beheren:

- Gebruikersrollen en toegangsregels. Bemerkt dat een portaal hiervoor ook kan integreren met een extern systeem.
- De structuur van het portaal (navigatie, pagina's, positionering van portlets). Dit kan ook in handen gegeven worden van ontwikkelaars.
- De look & feel. Doorgaans wordt de look & feel door ontwikkelaars opgesteld en is het de taak van de administrator om de gewenste look & feel te activeren.

3.2. Ontwikkeling

Ontwikkeling in de context van een portaal is voor een groot deel gebaseerd op klassieke webontwikkeling. Ontwikkelaars kunnen dan ook reeds gekende web frameworks zoals Struts of JavaServer Faces (JSF) blijven gebruiken. Verworven competenties zijn dus niet verloren.

Daarnaast moeten ontwikkelaars wel leren werken met specifieke portaalcomponenten zoals pagina's, portlets, lay-outs en themes. Vooral het eventing-mechanisme op niveau van portlets is even wennen. Er zijn specifieke API's die de ontwikkelaars moeten leren kennen. Deze API's zijn grotendeels gestandaardiseerd. In paragraaf 3.3 gaan we daar verder op in.

Voor de integratie vanuit portlets met de back-end kunnen dan weer de klassieke mechanismen gebruikt worden (API's, database-connecties, webservices, enzovoort).

Portaalleveranciers bieden de nodige tools om portlets te ontwikkelen, in veel gevallen ook Eclipse plugins waarin wizards aangeboden worden.

3.3. Portlet-standaarden

Eén van de belangrijkste functies van een portaal bestaat erin om content en toepassingen van verschillende bronnen en locaties samen te brengen. Om dat te kunnen realiseren, zijn standaarden heel belangrijk.

Leveranciers van third-party producten (bijvoorbeeld content-management-systemen) kunnen dan op standaarden gebaseerde portlets voorzien die naadloos integreren met portaaltools die die standaarden ondersteunen. Deze portlets worden dan overdraagbaar tussen portaalplatformen.

Zonder portlet-standaarden hebben we volgende problemen:

- Ontwikkelaars van pakketten (zoals content-management-systemen) moeten specifieke portlets bouwen voor meerdere portaalplatformen, wat bijkomende ontwikkelkosten betekent.
- Telkens de functionaliteit verandert, moeten de portlets voor de verschillende platformen aangepast worden, wat bijkomende onderhoudskosten met zich meebrengt.
- Eenmaal geïnvesteerd in portlets voor een specifiek portaalplatform is er een grote afhankelijkheid ten opzichte van dat platform. De kosten voor een overstap naar een ander platform zijn aanzienlijk groter.

Hieronder bespreken we de relevante standaarden, hun rol bij het uitbouwen van een portaal en hun onderling verband.

JSR-168 en JSR-286

Vanuit de Java community is een gemeenschappelijke set van API's ontstaan zodat portlets overdraagbaar zijn van het ene portaalplatform naar het andere.

Die Java portlet API is vastgelegd in Java Specification Request 168 (JSR-168), ook bekend als de Java Portlet Specification 1.0 (oktober 2003). De specificatie bepaalt onder andere de verschillende portlet modes (view, edit, help, configure), de set van window states (normaal, geminimaliseerd, gemaximaliseerd) en het URL-rewritingmechanisme.

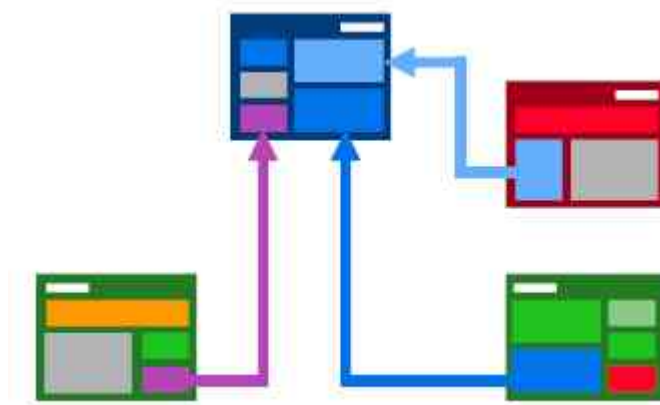
JSR-286 is de opvolger van JSR-168. De belangrijkste nieuwigheid is de standaardisatie van de inter-portletcommunicatie op basis van events.

WSRP – Web Services for Remote Portlets

WSRP (v2) werd een OASIS-standaard in april 2008. De standaard laat toe om portlets die draaien op een extern portaal te integreren in een eigen portaal aan de hand van webservices. Daar waar webservices typisch zorgen voor de overdracht van gegevens, wordt bij WSRP ook de presentatielogica teruggegeven. Zodoende moet de gebruikersinterface niet meer opnieuw ontwikkeld worden.

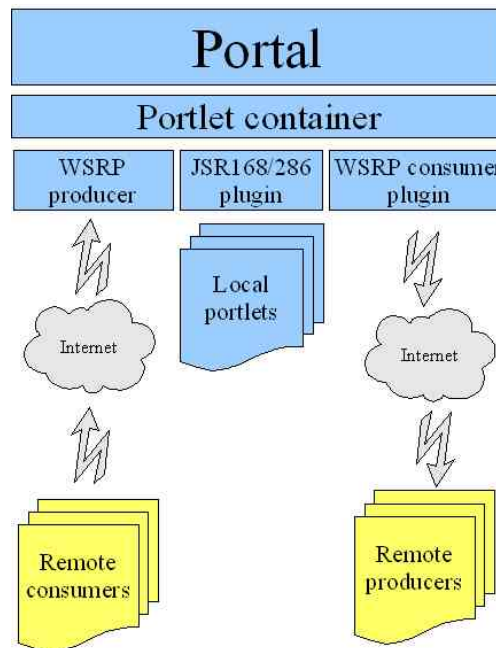
Figuur 9 illustreert een zogeheten federatie van portalen waarbij in een portaal verschillende toepassingen samengevoegd kunnen worden uit diverse andere portalen en op een overzichtelijke manier getoond aan eindgebruikers. We spreken hierbij van WSRP producers (leveranciers van portlets) en WSRP consumers (afnemers van portlets).

Omdat WSRP gebaseerd is op open standaarden (SOAP, WSDL, SAML), is het niet enkel van toepassing op Java-gebaseerde portalen.



Figuur 9: Federatie van portalen op basis van WSRP

In Figuur 10 zien we het onderlinge verband tussen de portletstandaarden. JSR168/286 zijn de standaarden voor lokale portlets in de Java-wereld en zorgen voor overdraagbaarheid van portlets tussen Java-gebaseerde portalen. WSRP daarentegen is de standaard voor het aanbieden en afnemen van remote portlets en is niet exclusief voor de Java-wereld. Het betreft dus geen concurrerende, maar complementaire standaarden.



Figuur 10: Onderling verband tussen de portletstandaarden

Het is zo dat niet alle portaalproducten de standaarden volledig ondersteunen. Vooral voor WSRP wordt dikwijls alleen de consumer-kant aangeboden, maar niet de producer-kant.

4. Huidige status van de portaalmarkt

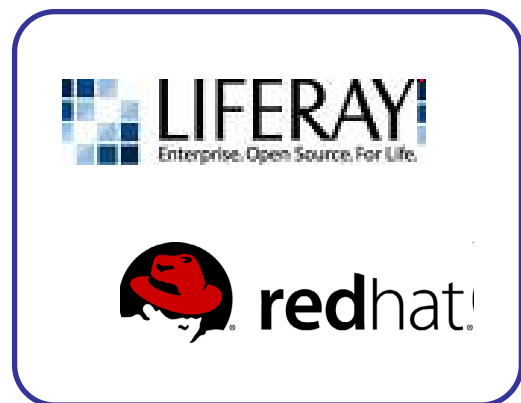
4.1. Marktoverzicht

De markt van horizontale portalen bestaat al een hele tijd en heeft al een paar consolidatierondes achter de rug. Op dit ogenblik is de markt geconcentreerd rond een beperkt aantal grote softwareleveranciers enerzijds en een paar valabele open source alternatieven anderzijds. De belangrijkste commerciële leveranciers zijn IBM, Microsoft en Oracle. De portaaloplossingen van Liferay en Red Hat zijn de belangrijkste open source alternatieven.

Commerciële oplossingen



Open source alternatieven



Figuur 11: Overzicht van de belangrijkste portaalleveranciers

IBM

IBM biedt met WebSphere Portal een complete bouwdoos met degelijke ontwikkeltools aan. De standaard portaalomgeving kan door middel van extra betalende modules (accelerators) uitgebreid worden met extra functionaliteit op vlak van workflow, content, collaboratie, dashboards, self-service, e-learning en mobiele toegang. Complementaire producten zoals Lotus WCM en Lotus Connections integreren naadloos met het portaal. Mashups worden ondersteund via het complementaire product IBM Mashup Center. Voor kleinere organisaties biedt IBM een "Portal Now" aanbod om sneller van start te gaan met een content-gebaseerd portaal.

Oracle

Door een reeks overnames beschikt Oracle over een aantal portaalproducten die sterk overlappend zijn: Oracle WebLogic Portal (vroeger BEA WebLogic Portal), Oracle Portal, Oracle WebCenter Interaction (vroeger BEA AquaLogic User Interaction), Oracle WebCenter Framework en Oracle WebCenter Services. Met de overname van Sun komt daar nog een portaalproduct bij, namelijk Sun GlassFish Web Space Server dat eerder gericht is op niet-kritische omgevingen.

Producten zoals Oracle WebLogic Portal maken deel uit van de “continue and converge” categorie van producten. Dat betekent dat die producten nog zeker 9 jaar ondersteund zullen blijven. Daarna zal zich hoogstwaarschijnlijk een migratie opdringen in de richting van het strategisch product, WebCenter Framework.

Microsoft

Met Microsoft Office Sharepoint Server (MOSS) heeft Microsoft een heel bekend en wijdverspreid portaalproduct in huis. Het biedt ingebouwde functionaliteit voor contentbeheer en collaboratie. Het geniet brede interesse, onder andere door het feit dat het eenvoudig op te zetten is op basis van templates. Tot nu toe is het vooral gebruikt als intern collaboratieplatform, maar naar eigen zeggen zou versie 2010 nu ook beter geschikt zijn als extern portaal. Alhoewel het platform een propriëitair portletmodel heeft (WebParts), bestaan er toch heel wat third-party webparts die geïntegreerd kunnen worden.

Liferay

Naast de bovengenoemde puur commerciële pakketten zijn ook een paar open source alternatieven het vermelden waard.

Een eerste leverancier van open source portals is Liferay. Hun product Liferay Portal is beschikbaar onder de MIT-licentie. Naast de gratis community-editie is er ook een enterprise-editie beschikbaar met ondersteuning en onderhoud tegen betaling. Het product bevat een ingebakken content-managementsysteem en collaboration tools. Daarnaast schenkt Liferay ook meer en meer aandacht aan sociale functies zoals het opvolgen van recente activiteiten van “vrienden” en leden van een community. Het is relatief snel op te zetten en te beheren. Voor meer gedetailleerde informatie verwijzen we naar de Open Source Product Review¹⁴. Ten opzichte van commerciële producten biedt Liferay minder uitgebreide mogelijkheden op vlak van personalisatie, dit door gebrek aan een rules engine. Er zijn minder portlets beschikbaar om te connecteren naar veelgebruikte zakelijke toepassingen, content-managementsystemen en collaboration tools. Liferay heeft minder ervaring met deployments met een groot aantal gebruikers en heeft ook een minder uitgebreid partnernetwerk.

¹⁴ [“Open Source Product Review – Liferay Portal 5.2.3”](#), december 2009, Bert Vanhalst, Sectie Onderzoek, Smals.

Red Hat

Red Hat JBoss Portal is een ander open source portaalproduct. Ook hier zijn er twee edities: de JBoss Portal community-editie en het Enterprise Portal Platform (EPP) met commerciële ondersteuning. In vergelijking met andere producten van Red Hat geniet JBoss Portal minder succes. JBoss Portal wordt gepositioneerd als een platform dat als basis dient voor eigen ontwikkeling en niet als een out-of-the-box portaal, wat minder interessant is voor organisaties die op zoek zijn naar een oplossing die snel op poten kan gezet worden. Onlangs werd een ander open source portalproject – eXo Portal – samengevoegd met JBoss Portal in het nieuwe GateIn-project¹⁵. Dit nieuwe portaalproject moet het beste uit beide initiatieven combineren en leiden tot een beter portaalproduct.

Portaaltools en cloud computing

Zoals andere softwaredomeinen ontsnappen ook portals niet aan het fenomeen *cloud computing*¹⁶. Bepaalde leveranciers bieden hun portaalproduct nu ook aan in de cloud. Met versie 2007 van SharePoint bood Microsoft al een stuk van de portaalfunctionaliteit aan in de cloud. Dit wordt verder uitgebreid met SharePoint Server 2010. IBM van zijn kant biedt de mogelijkheid om WebSphere Portal te draaien op Amazon Elastic Cloud Computing (EC2)¹⁷.

4.2. Selectiecriteria

Bij de keuze van een geschikt portaalproduct moet er rekening gehouden worden met de onderstaande criteria:

- Gebruiksscenario: zal het portaal dienen als samenwerkingsplatform of is het eerder gericht op de integratie van toepassingen?
- De snelheid van implementatie (eenvoudige configuratie) en de mogelijkheden voor ontwikkeling en integratie.
- De licentiekosten en kosten voor ondersteuning en onderhoud door de leverancier of een erkende partner.
- De compatibiliteit met de bestaande infrastructuur, bijvoorbeeld op vlak van applicatieservers en databanken.

¹⁵ Voor meer informatie omtrent het gemeenschappelijk portaalproject van JBoss Portal en eXo Portal (GateIn), zie <http://www.jboss.org/gatein/>

¹⁶ Voor meer informatie over cloud computing, zie Research Note “Software as a Service en Public Cloud Computing”, augustus 2009, Adelbert Groebbens, Sectie Onderzoek, Smals.

¹⁷ <http://www.ibm.com/developerworks/downloads/ls/wps-wcmse/ec2.html>

- De mate waarin de reeds aanwezige softwarepakketten (bijvoorbeeld content-managementsysteem) en toepassingen geïntegreerd kunnen worden.
- De mogelijkheid om te integreren met reeds aanwezige systemen voor Identity & Access Management (IAM).
- Eenvoud van configuratie en administratie.
- Kwaliteit en uitgebreidheid van de beschikbare ontwikkeltools.
- Ondersteuning voor de portletstandaarden (JSR-168/286 en WSRP).
- Aanwezigheid van een rules engine voor doorgedreven personalisatie op basis van profielinformatie en metagegevens.
- Productvisie en stabiliteit van de leverancier.

5. Prototype

5.1. Inleiding

In het kader van deze studie werd aan de hand van twee portaaltools nagegaan wat concreet de mogelijkheden en beperkingen zijn van portaaltools.

Het testscenario kadert in de inspectie-opdrachten van de RSZ. Hieronder is het scenario weergegeven. Daarbij zijn de betrokken toepassingen in vet aangeduid.

Een inspecteur krijgt een onderzoeksopdracht in verband met een werkgever in de bouwsector in **iBoss**. Ter voorbereiding van de enquête raadpleegt hij het **werkgeversrepertorium** en het **personeelsbestand** via de portaal-site van de sociale zekerheid; hij consulteert vervolgens **Genesis**, doet een beroep op documenten in het elektronisch werkgeversdossier (**DEE**) en raadpleegt informatie over de betreffende werkgever via **TEEPEE** (vb. consultatie werkgeversrekeningen (RE) en codebestand (FCD)). Daarnaast worden ook **DMFA-Monitoring** en **DMFCC** gebruikt. Ten slotte brengt hij gegevens in met betrekking tot het onderzoek in **iBoss**.

Doel van de testen is om na te gaan of de betrokken toepassingen geïntegreerd kunnen worden in één portaalomgeving, zonder aanpassing aan de toepassingen, met inbegrip van single sign-on en het automatisch uitwisselen van parameters tussen de toepassingen.

Dit is dus een specifiek scenario – maar niet het enige – voor het gebruik van een portaaltool. Andere scenario's kunnen eerder gericht zijn op het integreren van content-managementsystemen en collaboratietools.

De toepassingen zijn gebaseerd op verschillende technologieën: Java Swing, webtoepassingen beschikbaar op het portaal van de sociale

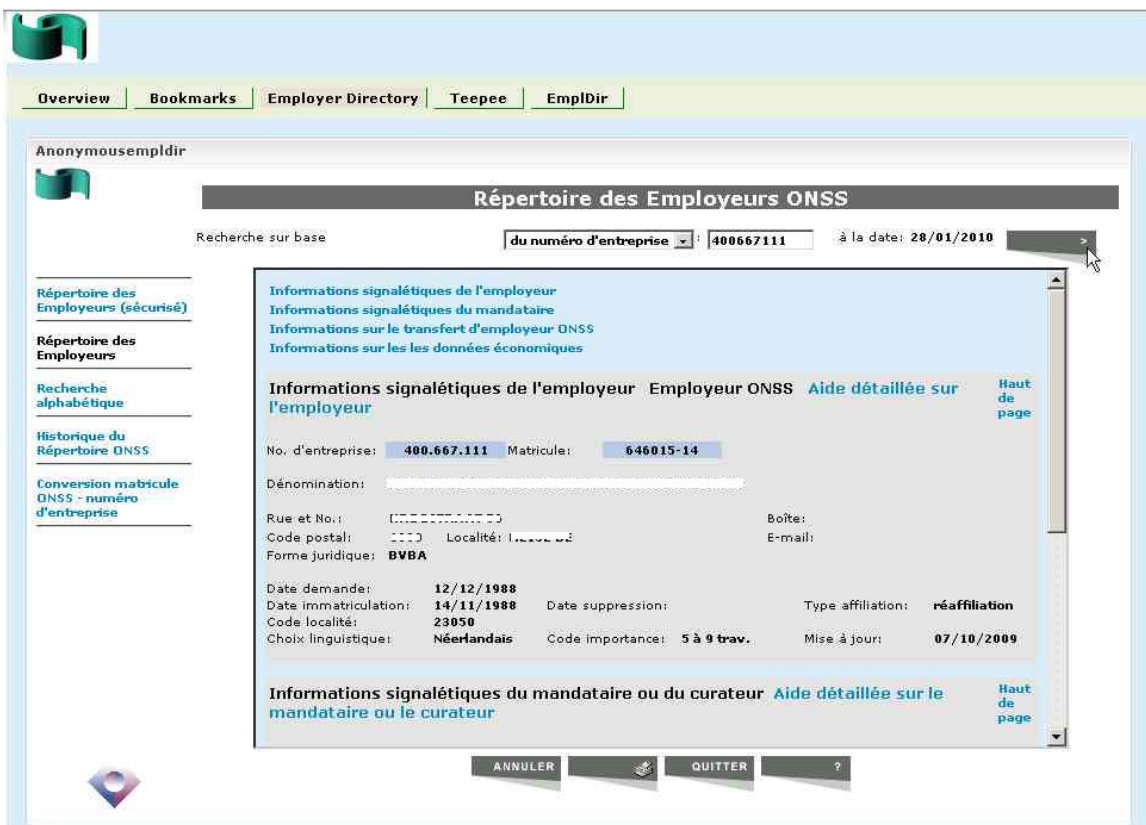
zekerheid, Visual Basic en PowerBuilder. Voor elk van de technologieën werd minstens één toepassing getest.

De testen werden uitgevoerd op basis van twee portaaltools: Oracle WebLogic Portal en IBM WebSphere Portal. Voor elk van de tools werden ongeveer drie dagen uitgetrokken om de testen uit te voeren, samen met consultants van Oracle en IBM.

5.2. Testresultaten

In eerste instantie werd getest of de toepassingen zonder aanpassingen geïntegreerd konden worden in de portaalomgeving.

De webtoepassing Werkgeversrepertorium kon eenvoudig geïntegreerd worden via web clipping (zie Figuur 12). Wel dienden de cascading style sheets (CSS) manueel gekopieerd te worden naar de portaalomgeving om de originele look & feel van de toepassing te behouden.



Figuur 12: Integrité van de webtoepassing Werkgeversrepertorium

De integratie van de rich clients is een ander paar mouwen. In het algemeen moet een nieuwe gebruikersinterface ontwikkeld worden om de functionaliteit te integreren in het portaal. Dat geldt zeker voor de niet-Java-toepassingen (Visual Basic, PowerBuilder). De Java Swing-toepassingen kunnen eventueel nog omgezet worden naar een Java

applet en op die manier geïntegreerd worden. Concreet werd dit uitgetest met de toepassing Teepee. De integratie kon binnen de gegeven tijd slechts gedeeltelijk gerealiseerd worden: de toepassing start op bij het selecteren van de "Teepee"-pagina, maar draait nog altijd in een apart venster. Dat kan verholpen worden mits bijkomende aanpassingen.

Voor een andere rich client (iBoss) werd een nieuwe gebruikersinterface gebouwd in de vorm van drie portlets die communiceren met de bestaande back-end via webservices:

1. de eerste portlet biedt een lijst met onderzoeken aan, specifiek voor de aangemelde gebruiker;
2. een tweede portlet toont details van een onderzoek dat aangeklikt wordt in de eerste portlet;
3. daarnaast is ook een "history" portlet voorzien die een chronologische lijst bijhoudt van werkgevers verbonden aan een geselecteerd onderzoek. Dat laat een gebruiker toe om eenvoudig terug te keren naar een vorige werkgever.

Het resultaat is een soort startpagina en is te zien in Figuur 13. Dit toont aan dat nieuwe ontwikkeling op basis van een portal framework veel mogelijkheden biedt om de inhoud van het portaal goed af te stellen op de noden van de gebruikers.

The screenshot displays a web portal interface with a navigation bar at the top containing 'Overview', 'Bookmarks', 'Employer Directory', 'Teepee', and 'EmplDir'. The main content area is divided into three portlets:

- Login:** Displays 'Bienvenue, VanhalstBe' and a 'Logout' button.
- Inquiry List:** A table with columns 'Nr', 'Inquiry ID', and 'Employer ID'.

Nr	Inquiry ID	Employer ID
1	8377807	2471692
2	8724064	4936603
- my History:** A list of Employer IDs: 4936603 and 2471692.
- Details:** A panel showing information for a selected inquiry.
 - KSZ: 170735616
 - BCE/KBO: 466260291
 - Denomination: [redacted]
 - Address: [redacted]
- Inquiry Data:** A light blue panel with the following details:
 - Inquiry number: 2010/004525
 - ED reference: WAITING
 - Creation date: 2010-01-28
 - Matter: MA6603S0101
 - Direction: D21S
 - Origin: O302C
 - Reason: M60S01
 - Priority: Prior-00
- Inquiry Status:** A light blue panel showing:
 - General: new
 - Current state: open

Figuur 13: Startpagina met lijst van onderzoeken, details van het geselecteerde onderzoek en een geschiedenis van geselecteerde werkgevers

Op vlak van inter-portletcommunicatie was de intentie om automatisch in het Werkgeversrepertorium (op de "Employer Directory"-pagina) de informatie van de werkgever te tonen die geselecteerd is in het startscherm ("Overview"-pagina). Dat is theoretisch mogelijk, maar kon niet gerealiseerd worden binnen de gegeven tijd, wat aangeeft dat de bijkomende ontwikkeling niet triviaal is.

Single sign-on werd gerealiseerd wat betreft de toegang tot de iBoss webservice voor het opvragen van de gegevens over de uit te voeren onderzoeken. Het portaal houdt de gebruikersnaam en het paswoord bij in een credential vault en kan op basis daarvan de authenticatie verzorgen in naam van de gebruiker.

Hetzelfde mechanisme kan niet toegepast worden voor SSO ten opzichte van de webtoepassingen op het portaal van de sociale zekerheid. Om die SSO te voorzien, is bijkomende integratie nodig in de back-end, wat technisch gezien neerkomt op de ontwikkeling van een authenticator en service provider.

Op vlak van personalisatie is het principe aangetoond om pagina's en portlets te tonen of te verbergen op basis van het profiel van de gebruiker. Ook de taalinstellingen van de browser konden gebruikt worden om de taal van de volledige portaalomgeving automatisch aan te passen.

5.3. Besluit

Uit de testen kunnen we besluiten dat een portaaltool interessante mogelijkheden biedt indien de gebruikersinterface specifiek ontwikkeld is op basis van het portal framework.

Het integreren van bestaande toepassingen daarentegen is niet altijd even eenvoudig. Zo moeten er extra inspanningen geleverd worden om parameters door te geven aan bestaande webtoepassingen. De integratie van rich clients is bovendien enkel mogelijk mits aanpassingen aan de bestaande toepassing of het volledig herschrijven van de gebruikersinterface.

Voor dit specifieke gebruiksscenario, sterk gericht op de integratie van maatwerk-toepassingen, moeten we dus rekening houden met behoorlijke integratiekosten.

6. Besluit en aanbevelingen

We kunnen besluiten dat portaaltools de basis kunnen vormen tot een uniek toegangsscherm. We moeten echter rekening houden met de beperkingen bij de integratie van bestaande toepassingen.

Het is interessant om de vraag te stellen wanneer we best kiezen voor een portaaltool ten opzichte van klassieke webontwikkeling. Hieronder zijn een aantal pluspunten gegeven van portaaltools:

- Een portaal is eenvoudiger uitbreidbaar op vlak van toepassingen en content en laat dus een beheersbare en gefaseerde evolutie toe van de applicaties.
- Daar waar klassieke websites eerder een statische structuur hebben, kunnen we de structuur van een portaal eenvoudiger aanpassen en laten evolueren.
- Een portaal biedt meer doorgedreven personalisatie op basis van het gebruikersprofiel.
- Het systeem van inter-portletcommunicatie biedt de mogelijkheid om parameters door te geven tussen toepassingen, wat de gebruiker een geïntegreerde gebruikerservaring biedt.

Tot slot geven we hieronder nog een aantal aandachtspunten en aanbevelingen mee die van belang zijn bij het voorbereiden en implementeren van een portaalomgeving.

- Ga na welke toepassingen en informatiebronnen geïntegreerd moeten worden in de portaalomgeving, zowel op korte als op langere termijn. Op basis daarvan kan de technische haalbaarheid nagegaan worden en kan ook een inschatting gemaakt worden van de integratiekosten. De integratiekosten kunnen al snel heel hoog oplopen.
- Leg de verschillende doelgroepen (gebruikerssegmenten) vast en bepaal de personalisatieregels.
- Bij een toenemend aantal gebruikerssegmenten en toepassingen wordt een portaal interessanter.
- Een portaalproject is een veranderingsproject. Het is belangrijk om de eindgebruikers van bij het begin te betrekken en in begrijpbare taal toe te lichten wat die verandering precies zal inhouden. Geef duidelijk de meerwaarde aan van het portaal en luister naar feedback op basis van pilootprojecten.
- De mogelijkheden van portaaltools (zoals inter-portletcommunicatie) kunnen pas echt benut worden indien de gebruikersinterface gebaseerd is op het portal framework.
- Niet alles moet geïntegreerd worden in de portaalomgeving: enkel die zaken die een meerwaarde bieden. Een portaal is meestal één van de front-end kanalen voor gebruikers en dus niet hét uniek kanaal.

- Een portaal moet als uniek toegangspunt altijd beschikbaar en performant zijn, zoniet krijgt de gebruiker geen toegang tot de systemen die via het portaal ter beschikking gesteld worden. Zorg voor een degelijk contract voor ondersteuning en onderhoud.
- Het uitbouwen van een portaal is een langetermijnproject. Indien mogelijk gebeurt het best gefaseerd. In een eerste fase kunnen de bestaande toepassingen ongewijzigd toegankelijk gesteld worden via het portaal. Daarbij kan al single sign-on voorzien worden. In een tweede fase kunnen de toepassingen dan stapsgewijs gemigreerd worden naar het portal framework. Dat houdt de ontwikkeling in van een nieuwe gebruikersinterface die communiceert met de bestaande back-end. Voorwaarde is dat de bestaande toepassingen gebaseerd zijn op een gelaagd model met een duidelijke scheiding tussen de gebruikersinterface, de business-logica en data.
- Op vlak van ontwikkeling moeten de portaalcomponenten (portlets, pagina's, themes) beheerd worden als applicaties. Dat wil zeggen dat ook hier de regels moeten toegepast worden voor life cycle management (versiebeheer, uitvoeren van testen, planning van releases, enzovoort).
- Naast kennis van de klassieke web frameworks moeten ontwikkelaars extra competenties verwerven.

De portaaltools en –technologieën zijn matuur en gestandaardiseerd. Er zijn zowel commerciële als open source oplossingen beschikbaar.

Het zijn niet zozeer de licentiekosten die zullen doorwegen in de totale kostprijs van een portaalproject, maar wel de kosten voor de integratie van de verschillende toepassingen in de portaalomgeving. De richtprijs bij commerciële oplossingen is 40.000 euro per cpu (+ 20 % voor onderhoud en ondersteuning). Bij open source oplossingen zijn er geen licentiekosten, maar is het bijzonder aangeraden om te kiezen voor de enterprise-editie waarbij men wel betaalt voor onderhoud en ondersteuning.

Bij de keuze van een portaaltool moeten we voornamelijk rekening houden met het beoogde gebruik (scenario) en de technische mogelijkheden. Voor scenario's die sterk steunen op technische integratie zijn de oplossingen van IBM, Oracle en Red Hat meer geschikt. Voor samenwerkingsplatformen en scenario's waarbij de snelheid van implementatie belangrijk is, zijn de oplossingen van Liferay en Microsoft meer geschikt.

De sectie Onderzoek van Smals brengt met regelmaat verschillende publicaties uit over een hele waaier aan topics in de huidige IT-markt. U kan deze publicaties opvragen via het extranet :

<http://documentatie.smals.be>

Of u kan rechtstreeks contact opnemen met het secretariaat van de afdeling "Klanten & Diensten", op het nummer 02/787 58 24.

7. Afkortingen

API – Application Programming Interface

CAS – Central Authentication Service

CMS – Content Management System

CRM – Customer Relationship Management

HRM – Human Resources Management

IAM – Identity & Access Management

IPC – Inter-Portlet Communication

JCR – Java Content Repository

JSF – JavaServer Faces

JSR – Java Specification Request

LDAP – Lightweight Directory Access Protocol

NTLM – NT Lan Manager

OASIS – Organization for the Advancement of Structured Information Standards

REST – Representational State Transfer

RIA – Rich Internet Applications

SAML – Security Assertion Markup Language

SOAP – Simple Object Access Protocol

SSO – Single Sign On

WCM – Web Content Management

WSDL – Web Services Description Language

WSRP – Web Services for Remote Portlets