Razvoj na DotNetNuke platformi

Josip Šaban

crmT d.o.o

Microsoft
WinDays

Poslovno-tehnološka konferencija
Opatija, 22-25.04.2008.
21.04.2008. predkonferencijski dan

Microsoft
WinDays

Sponzori

GENERALNI SPONZOR

SPONZOR KOMUNIKACIJSKIH TEHNOLOGIJA

SPONZOR SISTEMSKE INTEGRACIJE

SPONZOR AUDIO VIZUALNIH TEHNOLOGIJA

GLAVNI SPONZOR

SPONZOR

MEDIJSKI POKROVITELJ

POKROVITELJ

SLUŽBENO VOZILO KONFERENCIJE

Zahvale na izboru

- PNP predavanje
- Osnovna verzija predavanja održana u veljači na SQL & Developers grupi

O čemu NEĆE biti govora

- O elementima koji su predavani na SQL & Developers grupi:
  - instalaciji DotNetNuke portala
  - “out-of-the box” mogućnostima
  - “kako izraditi modul”
- Predavanje je moguće vidjeti na http://www.mscommunity.net/Default.aspx?tabid=1190

Preduvjeti

- Osnovno poznavanje
  - Internet Information Server-a
  - SQL Server-a 2005
  - ASP.NET-a 2.0
  - Visual Studi-a 2005
  - principa funkcioniranja portala
  - DotNetNuke-a 4 (?)

O čemu ĆE biti govora

- Arhitekturi DotNetNuke portala
- Logiranju
- Arhitekturi modula
- Scheduler sustavu ugrađenom u DotNetNuke
- Lokalizaciji
- Razvoju “jednostavnog” modula
- ...

Statistika (www.dotnetnuke.com)

- 562207 registriranih članova (10.04.2008)
- Prosječno 450 novih članova dnevno
- Prosječno 5000 dnevnih preuzimanja
- 25+ uključenih modula

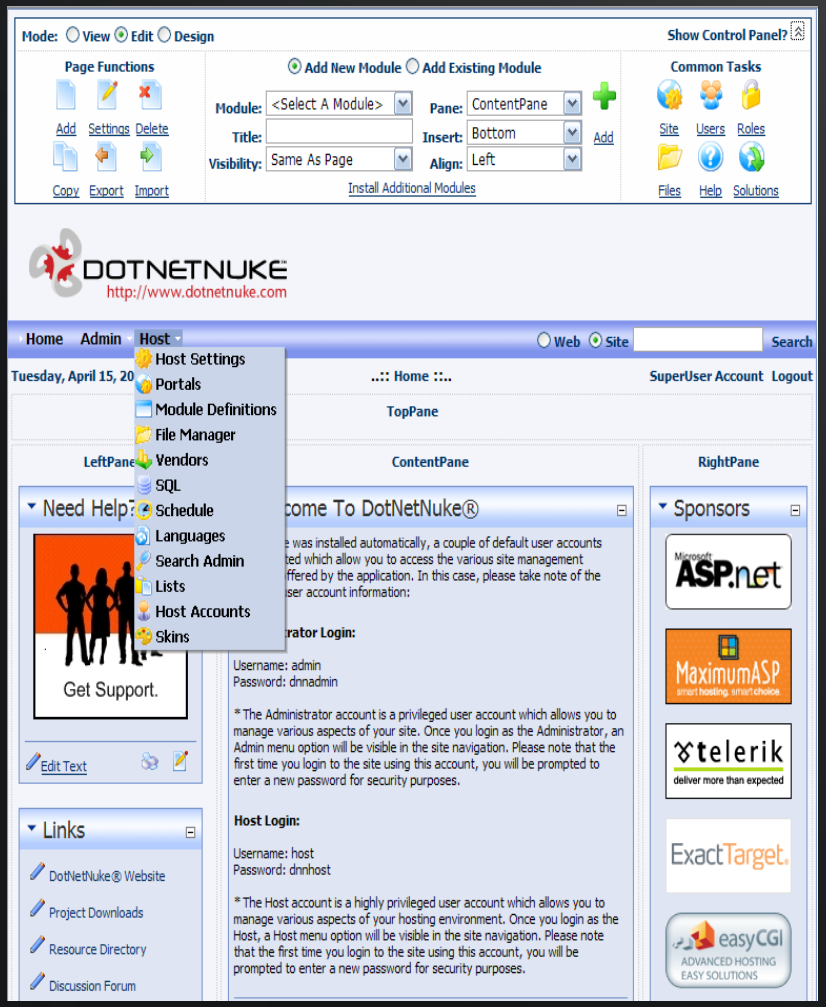
Izdano pod BSD licencom

- (B)erkeley (S)oftware (D)istribution
- Napravljena na University of California, Berkeley
- Implementacija portala mora sadržavati originalni *copyright* tekst
- Dozvoljava potpunu prilagodbu i implementaciju aplikacije u profitne i neprofitne svrhe

Što je portal?

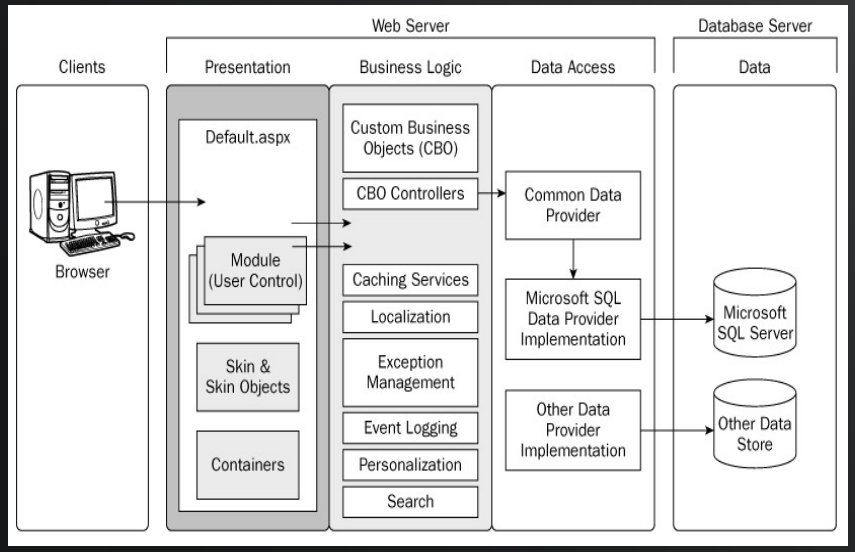
- Portal se može definirati kao web aplikacija koja služi prikazu i izradi sadržaja iz različitih izvora te (najčešće) služi kao prezentacijski sloj nekog informacijskog sustava – “IT” definicija
- Portal je web aplikacija kojom **u potpunosti** može upravljati **poslovni korisnik** (sa minimalnim ili bez ikakvog tehničkog znanja) te za čije korištenje ne treba nikakvo tehničko predznanje – “poslovna” definicija
- Svatko od nas može napraviti svoju definiciju ovisno o kontekstu iz kojeg promatra

Pogled unaprijed...



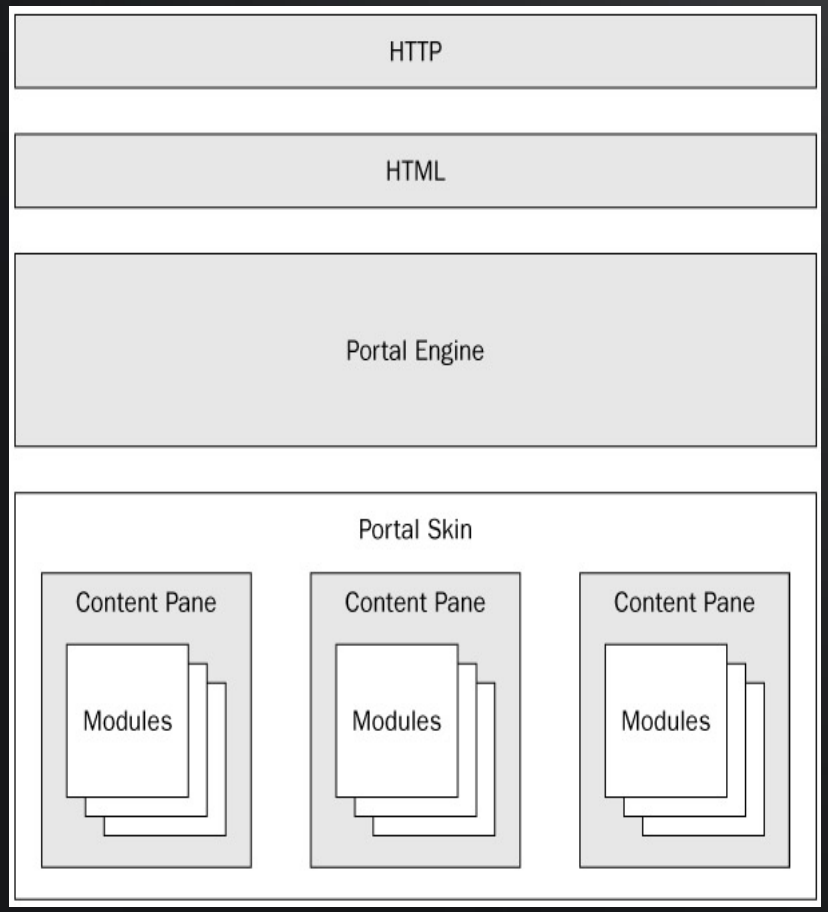
- Osnovni izgled portala nakon početne instalacije
- Vidljivi osnovni dijelovi arhitekture
  - Control Panel
  - Glavni izbornik
  - *Placeholder*-i za module
  - Postavljeni moduli
  - ...

Arhitektura portala



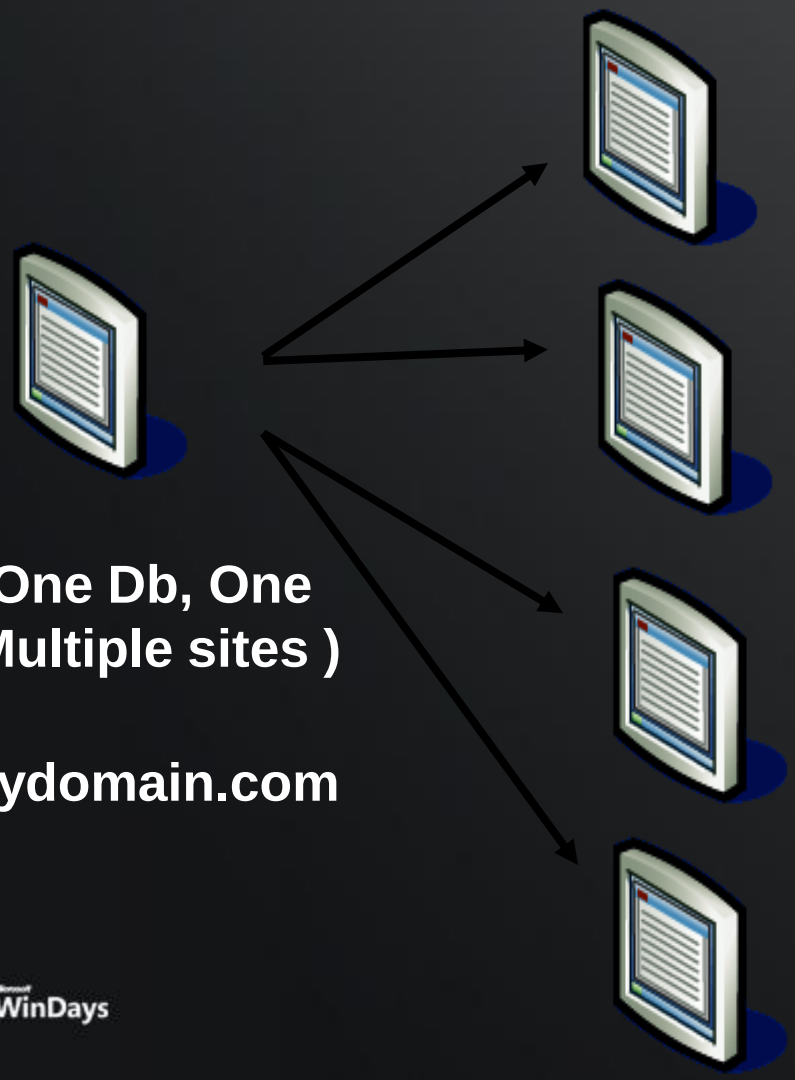
- Koncepti:
  - Model *provider-a*
    - Membership, Roles, Profile
    - Data
    - Scheduling
    - Logging
    - HTML Editor
    - Search
  - Custom Business Objects and Controllers
  - Lokalizacijski *framework* koji kopira ASP.NET 2.0 implementaciju
  - Friendly URL's
    - http://www.test.com/default.aspx?tabid=622 (3.x)
    - http://www.test.com/RoadMap/FriendlyURLs/tabid/622/default.aspx (4.x)

Arhitektura portala



- Dohvat stranice se obavlja u koracima:
  - Dohvat konfiguracije stranice
    - Uključuje dohvat modula na stranici, povezivanja instanci modula sa lokacijom na kojoj se trebaju pojaviti (sa njihovim *content pane*-om), te dohvaćanje sigurnosnih rola povezanih sa modulima
  - Provjera prava korisnika koji zahtjeva sadržaj
    - Povezivanje konteksta korisnika koji zahtijeva stranicu (bilo da je registrirani ili anonimni korisnik) te izrada liste “autoriziranih” modula za traženu stranicu
  - Dohvat sadržaja za autentificirane module (svaki modul se sam brine o svim aspektima svog prikaza)
    - Proces služi dinamičkom ubacivanju “autoriziranih” modula u pripadajuće *content pane*-ove
    - Nakon što su svi moduli učitani, svaki modul je tada odgovoran za dohvat i prikaz sadržaja (izvodi se definirani niz inicijalizacijskih događaja svakog modula)

Arhitektura portala



Host (One Db, One portal, Multiple sites)

www.mydomain.com

prod.mydomain.com

Dva IIS *web-site*-a

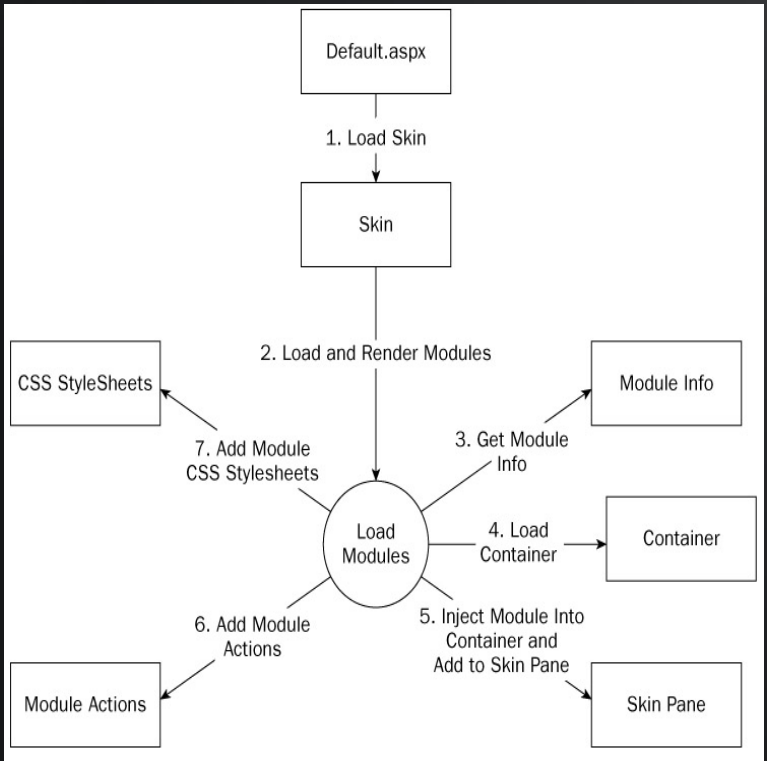
dev.mydomain.com

www.mydomain.com/prod

Jedan IIS *web-site*

www.mydomain.com/dev

Prezentacijski sloj



- Sastoji se od:
  - **Web formi**
  - **Skinova (korisničke kontrole)**
  - **Container-a (korisničke kontrole)**
  - **Modula (korisničke kontrole)**
  - **Klijentskih skripti**
    - Korisnički moduli i skinovi mogu sadržavati vlastite Javascript datoteke
    - U oba slučaja dotični se moraju nalaziti u istom direktoriju kao i skin/modul

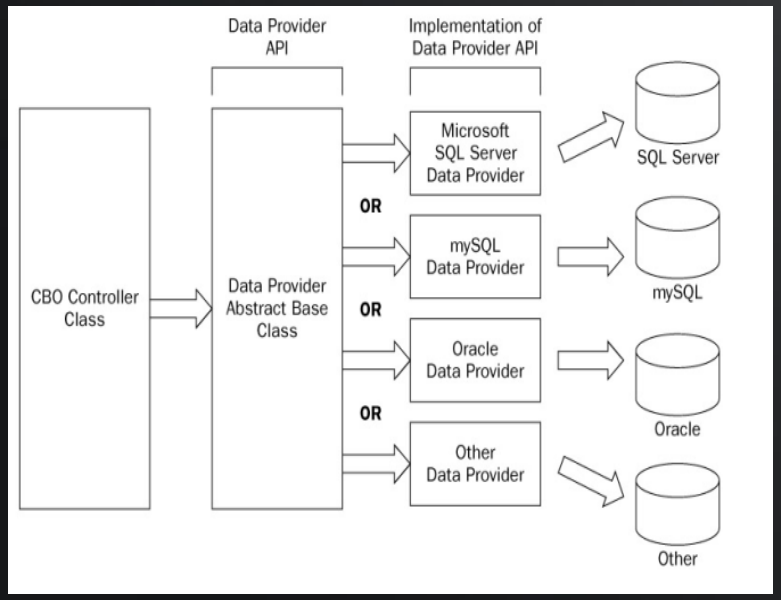
Poslovni sloj

- Služi za izvršavanje poslovne logike svih ključnih dijelova portala
- Preko ovog sloja dostupno je više servisa nužnih za funkcioniranje i portala i korisničkih modula:
  - Lokalizacija
  - Caching
  - Upravljanje iznimkama
  - Logiranje
  - Personalizacija
  - Pretraga
  - Instalacija i nadogradnja
  - Membership, roles, and profile
  - Sigurnosne dozvole
  - U ovom sloju se također nalaze CBO, čija je glavna uloga spremanje informacije o objektima

Podatkovni sloj

- Ovaj sloj pruža podatkovne usluge za sloj poslovne logike – služi protoku podataka iz i u podatkovno spremište
- Kao što je već rečeno, ovaj sloj izveden je korištenjem modela dobavljača (*Provider model*) u cilju podržavanja velikog broja podatkovnih izvorišta
- Podatkovni sloj sastoji se od dva elementa:
  - **Data Provider API:** Abstraktna bazna klasa koja predstavlja predložak "ugovora" koji implementacija mora ispuniti
  - **Izvedba Data Provider API-a:** Klasa koja nasljeđuje gore opisanu apstraktnu klasu i ispunjava "ugovor" implementacijom potrebnih članova i metoda

Data provider



- Data provider
  - Prvi implementiran
  - U početku samo podrška za SQL Server ugrađena u sam portal
  - Danas otvorena arhitektura definirana preko *web.config* datoteke

```
<data defaultProvider="SqlDataProvider">
    <providers>
        <clear />
        <add name="SqlDataProvider" type="DotNetNuke.Data.SqlDataProvider,
            DotNetNuke.SqlDataProvider" connectionStringName="SiteSqlServer"
            upgradeConnectionString="“ providerPath="~\Providers\DataProviders\SqlDataProvider\“
            objectQualifier="" templateFile="DotNetNuke_template.mdf"
            databaseOwner="dbo" />
    </providers>
</data>
```

Logiranje događaja

- *Logging Provider* pruža konfigurabilnu i proširivu arhitekturu koja uključuje
  - Logiranje iznimki
  - Auditing događaja
  - Sigurnosno logiranje
- Na početku nije uključeno
  - Host -> Host Settings -> Advanced -> Other Settings
    "Site Log Buffer (Items):" 1
    "Site Log History (Days):“ 60 (ili neki drugi iznos)
    Isključi "Enable Event Log Buffer”
  - Admin -> Site Settings -> Advanced Settings -> Host Settings
    Postavi "Site Log History (Days):“ 60 (ili neki drugi iznos)

Logiranje događaja

- Osnovni koncepti
  - **Klasifikacija log-a**
    - **Klasifikacija događaja –** bilo koji događaj unutar portala
    - **Klasifikacija iznimke** - možemo konfigurirati logiranje iznimki i metapodataka vezanog uz njih
    - Iako su ove dvije klasifikacije slične, postoje dvije kategorije isključivo zato jer logiraju različite informacije
  - **Vrsta loga**
    - Predstavlja vrstu događaja koja je stvorila unos u log – npr. LOGIN_FAILURE
    - *Logging provider* se može ponašati drugačije za različite vrste logova – također se može uključiti i isključiti za različite tipove logiranja
    - U ovisnosti o implementaciji, postoji mogućnost konfiguriranja različitih odredišta logiranja u odnosu na vrstu
  - **Konfiguracija vrste loga**
    - *Logging Provider* se konfigurira preko modula dostupnog iz *Log Viewer* modula

Scheduler

- Scheduler je mehanizam koji omogućava vremensko određeno izvršavanje zadataka
- Izvden je korištenjem *Provider* modela te se može lako zamijeniti nekom drugom implementacijom
- No, zbog prirode DotNetNuke portala potrebno je naglasiti
  - Izvršava se u kontekstu web aplikacije te je podložan *application recycle* procesu
  - U web-site hosting okruženju često je periodičko recikliranje radnog procesa radi čuvanja resursa
  - U tom času Scheduler stane sa radom – to nisu zadaci 24/7
  - Oni se izvršavaju po predefiniranom rasporedu u časovima kada je radni proces aktivan
  - Možemo definirati kako se često zadatak izvršava u periodičkim mjerama (svakih *x* sekundi, minuta, sati, ...)

Lokalizacija

- Lokalizacijski API služi za višejezično prevođenje u korisničkim modulima – puna podrška za lokalizaciju postojećih modula je već odavno ugrađena u portal
- U trenutku pripreme ove prezentacije postoji podrška samo za statičku lokalizaciju (to jest, moguće je prevoditi tekst i tekstualne stringove) – za specifične potrebe treba sam napisati podršku
- **Locales**
  - *Locale* je kombinacija koda države i koda jezika – u mnogo zemalja se govori više jezika i obrnuto, isti jezik ili njegov dijalekt se govori u više država
  - DotNetNuke portal podržava 4 tipa *locale*-a
    - **Sistemski –** (en-US) je već upisan u DotNetNuke i ne može se promijeniti
    - **Pretpostavljen portalski** – definiran u *Site Settings* modulu – automatski korišten za sve korisnike koji eksplicitno ne izaberu neki drugi u postavkama svoga korisničkog računa
    - **Korisnički** - izabran u modulu za promjenu podataka o korisniku
    - **Custom** - omogućava prilagodbe prijevoda za svaki portal posebno – lokalizacijski *framework* dodaje *PortalID* za svaki *custom locale*

Lokalizacija

- **Postoje tri nivoa resursa**
  - **Aplikacijski**
    - dijeljeni među modulima u implementaciji
    - spremište za generičke prijevode
    - spremljeni su u *App_Resources* direktoriju
  - **Lokalni**
    - specifični za pojedinu korisničku kontrolu (modul, skin, *container*)
    - spremljeni u pod-direktoriju od kontrole (u lokalni *App_Resources* poddirektorij)
    - treba probati koristiti što je više moguće postojeće prijevode iz aplikacijskih resursa
  - **Globalni**
    - koriste se za lokalizaciju modula koji nemaju lokalne resurse, ne koriste dijeljenje (aplikacijske) ili na bio koji način ne upadaju u jednu od dvije gornje kategorije
    - spremljeni su u glavnom *App_Resources* direktoriju
    - Pretpostavljeno ime je *GlobalResources.resx* sistemski *locale*, inače je konvencija *GlobalResources.[locale].resx*

Moduli

- Moduli su nezavisne cjeline koje obavljaju neku poslovnu aktivnost
- Služe proširenju funkcionalnosti portala
- Pisani su u nekom .NET kompatibilnom jeziku
- Kompilirani asembliji

Razvojni proces modula

Postavljanje

Početno stvaranje projekta - Visual Studio (programer)

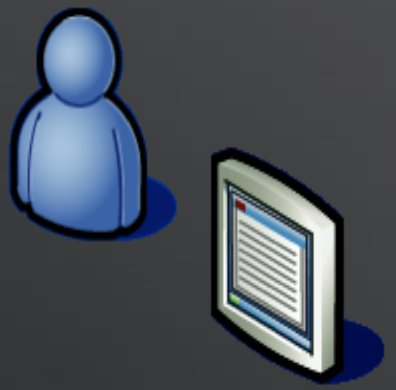
Razvoj

Razvoj potrebne funkcionalnosti (programer)

Pakiranje

Pakiranje modula u instalacijski paket (programer)

Instalacija



Instalacija modula u portal (administrator)

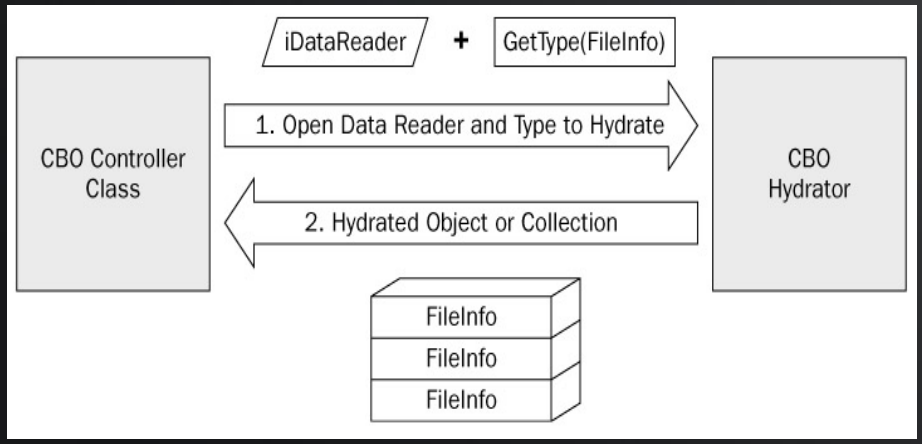
Skinovi

- Razdvajanje izgleda od sadržaja
- Više skinova na jednom portalu – svaka stranica može imati svoj skin
- Može se izrađivati korištenjem novog Microsoftovog dizajnerskog alata - *Expression Web*
- Sadrže HTML, CSS i slike
- Jednostavna distribucija preko ZIP datoteka

Spremnici (containers)

- Slično kao i skinovi, razdvajaju izgled od funkcionalnosti
- Ograničen je na jedan modul
- Stvara se na isti način kao i skin
  - HTML sa uputuma
  - XML sa *property*-ima objekta
  - Dodatne datoteke
- Kada se jednom napravi *upload* na portal, DNN ga pretvori u ASCX datoteku

CBO Hydrator



- Uključeni servis koji služi *hidraciji* (ovaj termin se odnosi na način *punjenja* instance neke klase, u ovom primjeru *DataReader* objekta) CBO-a ili kolekcije CBO-a
- Kada CBO Hydrator napuni *property*-e objekta, on također nalazi metapodatke korištenjem refleksije
- U tome času se ti podaci *cache*-iraju tako da iduci puta kada se objekt istog tipa treba hidrirati, neće morati dolaziti do ponovnog procesa otkrivanja

CBO Hydrator

- "Standardni" poziv

```
Dim dr As IDataReader
Try
        dr = DataProvider.Instance().GetFolder(PortalID, FolderPath)
        Dim f As New FileInfo f.ContentType = Convert.ToString(dr("ContentType"))
        f.Extension = Convert.ToString(dr("Extension"))
        f.FileId = Convert.ToInt32(dr("FileId"))
        f.FileName = Convert.ToString(dr("FileName"))
        f.Folder = Convert.ToString(dr("Folder"))
        f.Height = Convert.ToInt32(dr("Height"))
        f.PortalId = Convert.ToInt32(dr("PortalId"))
        f.Size = Convert.ToInt32(dr("Size"))
        f.Width = Convert.ToInt32(dr("Width"))
        Return f
Finally
        If Not dr Is Nothing Then dr.Close() End If
End Try
```

- CBO Hydrator

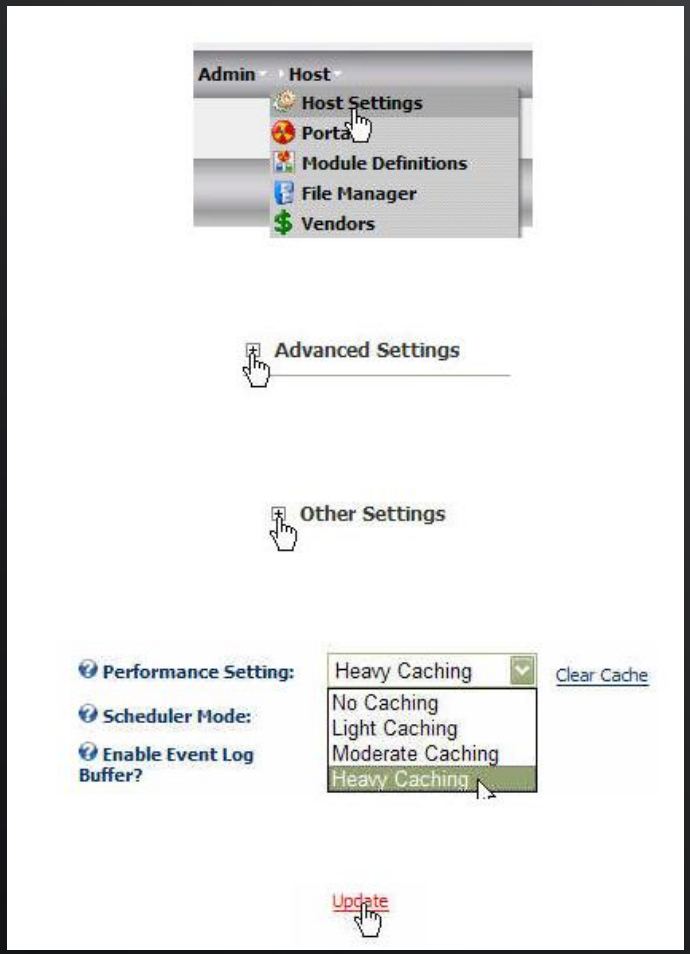
```
Return
CType(CBO.FillObject(
                DataProvider.Instance().GetFolder(PortalID, _ FolderPath),
                GetType(Services.FileSystem.FolderInfo)
                ), FolderInfo
        )
```

- Koristiti na "pravim" mjestima

WinDays

Cache u DotNetNuke-u

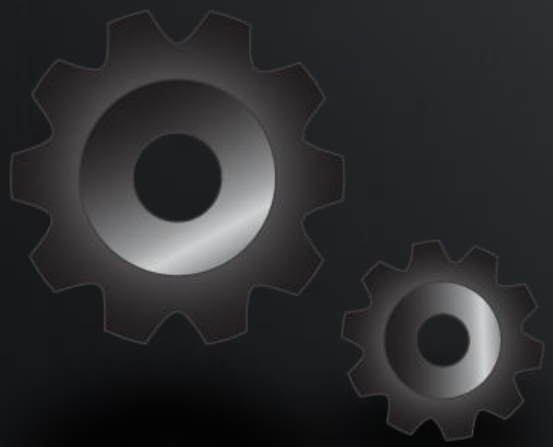
- DotNetNuke ima ugrađenu podršku za *cache*-iranje cijelog portala ili pojedinih stranica/modula (definiranje *cache timeout-a* – 60 je “zlatna sredina”)



Cache u DotNetNuke-u

- Koja je razlika između *Heavy*, *Moderate*, *Light* and *No caching?*
  - Razlika je u vremenu koji objekti ostaju u memoriji
  - “Best practice” kaže da je najbolje postaviti na *Heavy* jer čak i ako neki objekti ostanu u memoriji duže nego što je potrebno bolje je potrošiti više memorije nego izgubiti na performansama (čak i na ovoj postavci maksimalno vrijeme je 2 sata)
  - (Sa DotNetNuke foruma) – korisnik je pokazao statistiku da je uključivanjem *Heavy* cachinga sa *Moderate* nivoa dobio na performansama oko 20% (na serveru je bilo već od prije dostupno dovoljno memorije)

DEMO



- Prikazane neke od netom opisanih mogućnosti u sklopu razvijenog modula
- Korištena verzija 4.8.2
- Demo na:
  - Virtual PC 2007
  - SQL Server 2005 Express
  - Web Developer 2005 Express

Demo

“Konkurencija”

- Usporedbe funkcionalnosti između različitih (Microsoft) rješenja za izradu portala:

  http://www.dotnetnuke.com/Community/Blogs/tabid/825/EntryID/506/Default.aspx

Povezani sadržaji

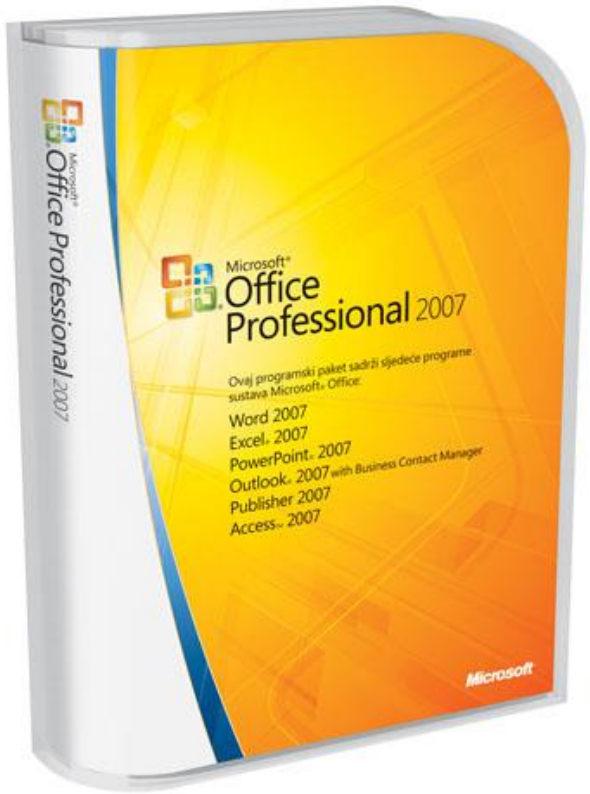
- Wrox - Professional DotNetNuke 4.0—Open Source Web Application Framework for ASP.NET 2.0
- Apress - Beginning DotNetNuke 4.0 Website Creation in C# 2005 With Visual Web Developer 2005 Express
- http://www.dotnetnuke.com

Razvoj – da ili ne?

- **Treba**
  - Odvagnuti između korištenja portala ili razvoja aplikacije “od nule”
  - Procijeniti da li je portal ono što vam stvarno treba
- **Ustvari...to je poslovni problem**
  - Da li ulaganje u razvoj može biti opravdano sa potencijalnom uštedom?
  - Da li se potrebni objekt (modul, skin, ...) može kupiti?
  - Da li je potrebna edukacija djelatnika?
  - Da li je potrebno uzeti vanjske konzultante?
  - Koji su infrastrukturni problemi?
  - Da li je potreban razvoj sučelja na različite izvore podataka?
  - Da li je potrebno podržavanje više verzija portala?
  - Što je potrebno za podršku razvijenih objekata?

Ispunite upitnike i osvojite nagrade

Petak: Microsoft Office Professional 2007



Microsoft WinDays

HVALA!

Microsoft
WinDays

Poslovno-tehnološka konferencija
Opatija, 22-25.04.2008.
21.04.2008. predkonferencijski dan