

XDS Konfigurationsvejledning

Opsætning af udviklerversion af Microsofts open source XDS.b fra Codeplex

**Projekt: Net4Care
Version: V0.1, 2012-06-12**

Indholdsfortegnelse

Indledning	2
Miljø	2
Opsætning af XDS	3
Opsætning af database	3
Kodefiler	4
Konfiguration af services	6
Test	9

Indledning

Nærværende dokument indeholder en kort beskrivelse af det nødvendige miljø, og opsætning af en udviklerversion af Microsofts open source XDS.b implementering fra Codeplex (<http://www.codeplex.com/>).

Udvikleropsætningen er rettet imod hhv. Java og .Net udviklere, som ønsker at ændre, tilpasse og debugge XDS systemet. XDS standarden er ikke helt skarp defineret, derfor kan det være en hjælp at debugge XDS'en, for derigennem at undersøge, hvordan Microsoft har fortolket standarden i relation til deres implementering. Denne dokumentation er for "January 2012 Release" af Xds.b .

Miljø

Her beskrives et tilstrækkeligt miljø til opsætningen af udviklerversionen af XDS.b implementeringen.

- Windows 7
- SqlServer 2008 express med management tools
- Visual Studio 2010 med C#¹

Det skal noteres, at dette miljø udelukket er til test og debuggin, og kun i et mindre omfang kan benyttes til større test eller drift relaterede situationer. Der henledes til opsætningsdokumentationen, som den fremtræder fra den oprindelige dokumentation, som kommer med implementeringen, hvis man ønsker at se en vejledning til drift.

¹ Man kunne vælge at benytte Visual Studio Express, hvorved alle anvendte værktøjer og komponenter (udtagen operativsystemet) var omkostningsfrie, men da Express ikke kan benytte "solutions" er denne dokumentation ikke helt fyldestgørende, men angiver stadig, hvad der skal ændres. I Express udgaven skal det blot udføres projektvis.

Opsætning af XDS

Opsætningen af den implementering der ligger på CodePlex er default rettet mod en driftsopsætning. Da denne installation rettes mod et udviklingsopsætning benyttes standardopsætningen ikke.

Microsoft XDS.b implementering og dokumentation kan hentes ned fra CodePlex. Adressen er <http://ihe.codeplex.com/> . Materialet findes under DOWNLOADS, hvorefter man henter "Sync" versionen. Dette er en zipfil, som udpakkes et passende sted på et lokalt drev.

Opsætning af database

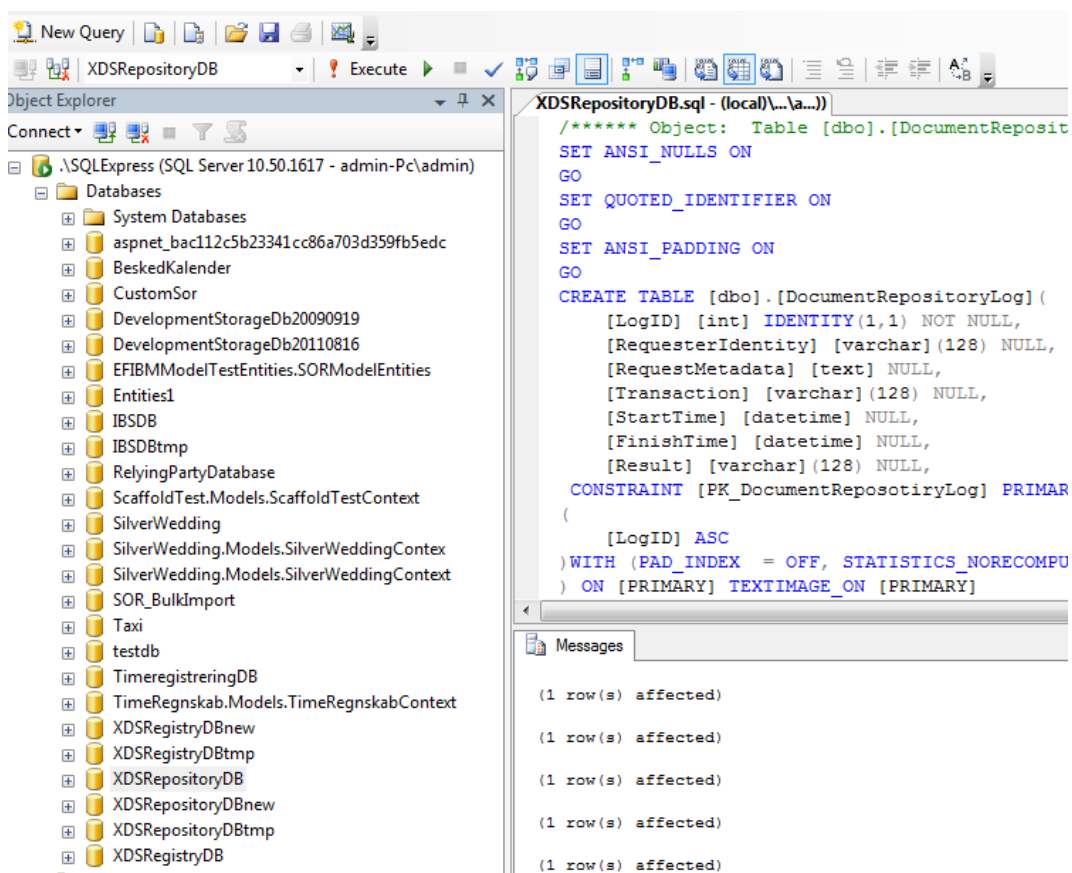
Der skal oprettes to databaser og indsættes data i databaserne. Til dette benyttes "Microsoft SQL Server Management Studio". Bemærk at I hele udviklingsopsætningen benyttes "Windows Authentication". Nu skal databaserne oprettes og data indsættes:

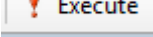
1. Start derfor SQL Server Management Studio
2. Højreklik på *Databases* og aktiver *New Database*
3. I "Database name" feltet skriv "XDSRegistryDB" og aktiver OK
4. Gentag dette med "XDSRepositoryDB"

Under *Databases* eksisterer nu de to databaser XDSRegistryDB og XDSRepositoryDB. Tabeller skal nu oprettes og data indsættes. I mappen source/DBScripts ligger filerne til dette.

1. Vælg XDSRepositoryDB databasen træk XDSRepositoryDB.sql over i SQL Server Management Studio.

Der skal nu være et billede ca. som nedenstående hvor XDSRepositoryDB er valgt:

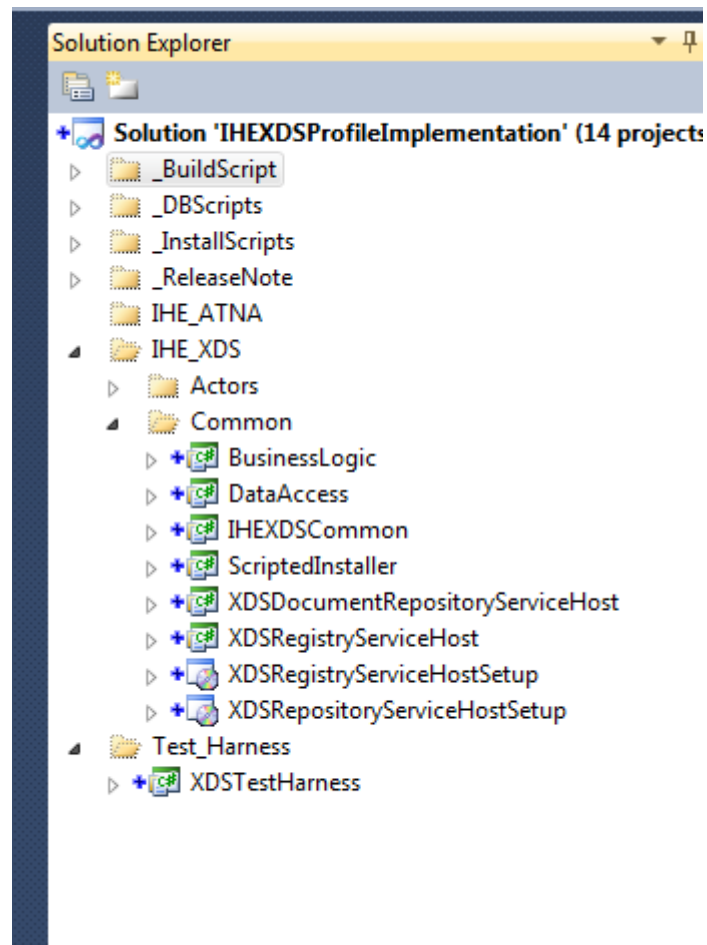


2. Tryk på Execute, led efter knappen 
3. Gentag dette med usp_IsQueryForSinglePatient.sql og PatientDocDetailsBySSID.sql
4. Herefter vælg XDSRegistryDB træk XDSRegistryDB.sql ind og aktiver Execute

Nu er databasen initialiseret.

Kodefiler

Start Visual og åben Solution IHEXDSProfileImplementation.sln som ligger i roden. Solution Explorer viser nu nedenstående:



Projekterne XSDDocumentRepositoryServiceHost og XDSRegistryServiceHost er services. De skal ændres til ikke at køre som services.

Under hvert projekt er en fil Program.cs.

Først redigeres XSDDocumentRepositoryServiceHost/Program.cs

Koden i Main() ændres til

```
try
{
    var svcHost = new
System.ServiceModel.ServiceHost(typeof(DocumentRepositoryService));
    svcHost.Open();
    System.Console.WriteLine("service up..");
    System.Console.ReadLine();
    svcHost.Close();
}
catch (System.Exception ex)
{
    System.Console.WriteLine( ex.Message );
}
```

Derefter redigeres XDSRegistryServiceHost/Program.cs

Koden i Main() ændres til

```
try
{
    var svcHost = new
System.ServiceModel.ServiceHost(typeof(DocumentRegistryService));
    svcHost.Open();
    System.Console.WriteLine("service up..");
    System.Console.ReadLine();
    svcHost.Close();
}
catch (System.Exception ex)
{
    System.Console.WriteLine( ex.Message);
}
```

Konfiguration af services

I filen (XSDSDocumentRepositoryServiceHost | XDSRegistryServiceHost)/App.config konfigureres services. Åben derfor App.Config, dette er en xml-fil.

Log

Er default sat til at ligge i C:\Log. Så enten sikres det at Log biblioteket eksisterer ellers rettes indholdet af

`configuration.system.diagnostics.sharedListeners.add.initializeData` til en passende sti.

DatabaseConnection

I filen "XSDSDocumentRepositoryServiceHost /App.config" under `configuration.connectionStrings`

Ændres

```
<add name="XDSRepository" connectionString="Initial
Catalog=XDSRepositoryDB;Data
Source=machinename;uid=admin;pwd=Enter321;Enlist=false;"
providerName="System.Data.SqlClient" />
```

til

```
<add name="XDSRepository" connectionString=" Data Source=.\SQLExpress;Initial
Catalog=XDSRepositoryDB;Integrated Security=True"
providerName="System.Data.SqlClient" />
```

og

```
<add name="ATNA" connectionString="Initial Catalog=XDSRepositoryDB;Data
Source=machinename;uid=admin;pwd=Enter321;Enlist=false;"
providerName="System.Data.SqlClient" />
```

til

```
<add name="ATNA" connectionString=" Data Source=.\SQLExpress;Initial
Catalog=XDSRepositoryDB;Integrated Security=True"
```

I filen "XDSRegistryServiceHost/App.config" under `configuration.connectionStrings`

Ændres

```
<add name="XDSRegistryOld" connectionString="Initial Catalog=XDSRegistryDB;Data Source=machinename;uid=admin;pwd=Enter321;Enlist=false;" providerName="System.Data.SqlClient"/>
<add name="ATNAOld" connectionString="Initial Catalog=XDSRegistryDB;Data Source=machinename;uid=admin;pwd=Enter321;" providerName="System.Data.SqlClient"/>
```

til

```
<add name="XDSRegistry" connectionString="Data Source=.\SQLEXPRESS;Initial Catalog=XDSRegistryDB;Integrated Security=True" providerName="System.Data.SqlClient"/>
<add name="ATNA" connectionString="Data Source=.\SQLEXPRESS;Initial Catalog=XDSRegistryDB;Integrated Security=True" providerName="System.Data.SqlClient"/>
```

HTTPS

I udviklingsopsætningen er HTTPS som transportlag fravalgt. Derfor skal HTTPS fjernes fra konfigurationerne. Ved "fjernes", menes enten en faktisk sletning eller en midlertidig udkommentering.

I "XSDDocumentRepositoryServiceHost/App.config" under `configuration.system.serviceModel.services.service` fjernes

```
<endpoint address="https://localhost:9081/XdsService/XDSRepository" binding="wsHttpBinding" bindingConfiguration="XDSRepository_HTTPS" name="XDSRepository_HTTPS_Endpoint" contract="Microsoft.IHE.XDS.DocumentRepository.IDocumentRepository" />
```

Under `configuration.system.serviceModel.services.service.host.baseAddresses` fjernes

```
<add baseAddress="https://localhost:9081/XdsService/XDSRepository" />
```

Under `configuration.system.serviceModel.behaviors.serviceBehaviors.behavior` erstattes

```
<serviceMetadata httpGetEnabled="true" httpsGetEnabled="true" />
```

med

```
<serviceMetadata httpGetEnabled="true"/>
```

og

```
<serviceCredentials>
  <serviceCertificate findValue="99e0fb6df06d82f2a802a9e5a1ca34c8e443f4fa" storeLocation="LocalMachine" storeName="Root" x509FindType="FindByThumbprint"/>
</serviceCredentials>
```

fjernes.

I "XDSRegistryServiceHost/App.config" under
`configuration.system.serviceModel.services.service` fjernes

```
<endpoint name="XDSRegistry_HTTPS_Endpoint"  
contract="Microsoft.IHE.XDS.DocumentRegistry.IDocumentRegistry"  
address="https://localhost:9080/xdsservice/xdsregistry" binding="wsHttpBinding"  
bindingConfiguration="XDSRegistry_HTTPS"/>
```

Under `configuration.system.serviceModel.services.service.host.baseAddresses`
fjernes

```
<add baseAddress="https://localhost:9080/xdsservice/xdsregistry"/>
```

Under `configuration.system.serviceModel.behaviors.serviceBehaviors.behavior`
erstattes

```
<serviceMetadata httpGetEnabled="true" httpsGetEnabled="true" />
```

med

```
<serviceMetadata httpGetEnabled="true"/>
```

og

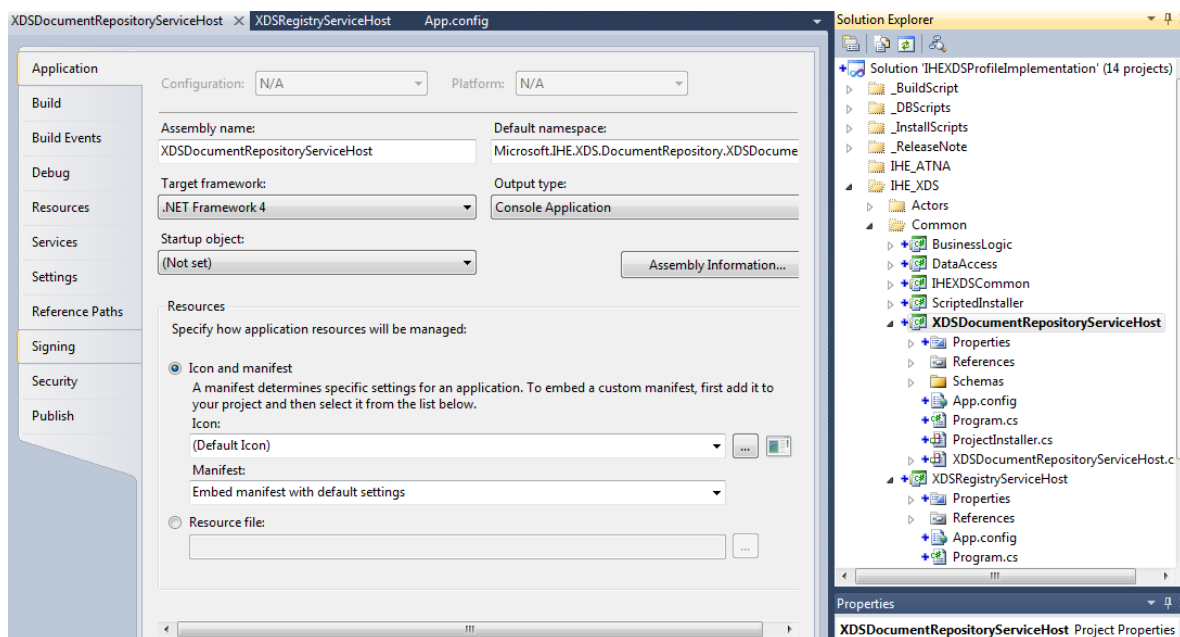
```
<serviceCredentials>  
  <serviceCertificate findValue ="f05be3965ae2ef0c7b296a8410b120fef39442a1"  
storeLocation="LocalMachine" storeName="Root" x509FindType="FindByThumbprint"/>  
</serviceCredentials>
```

fjernes.

Console Applikation

Det skal sikres at begge applikationer er konfigureret som Konsol applikationer. Dette sikres under properties for et projekt.

Sørg for at XDSDocumentRepositoryServiceHost er valgt under SolutionExplorer og tryk Alt+Enter . Nedenstående billede fremkommer.



Her skal det sikres at Output type skal være Console Applikation.

Det samme gennemføres med XDSRegistryServiceHost projektet. Her plejer default at være "Windows applikation" at være valgt. Dette skal rettes.

Test

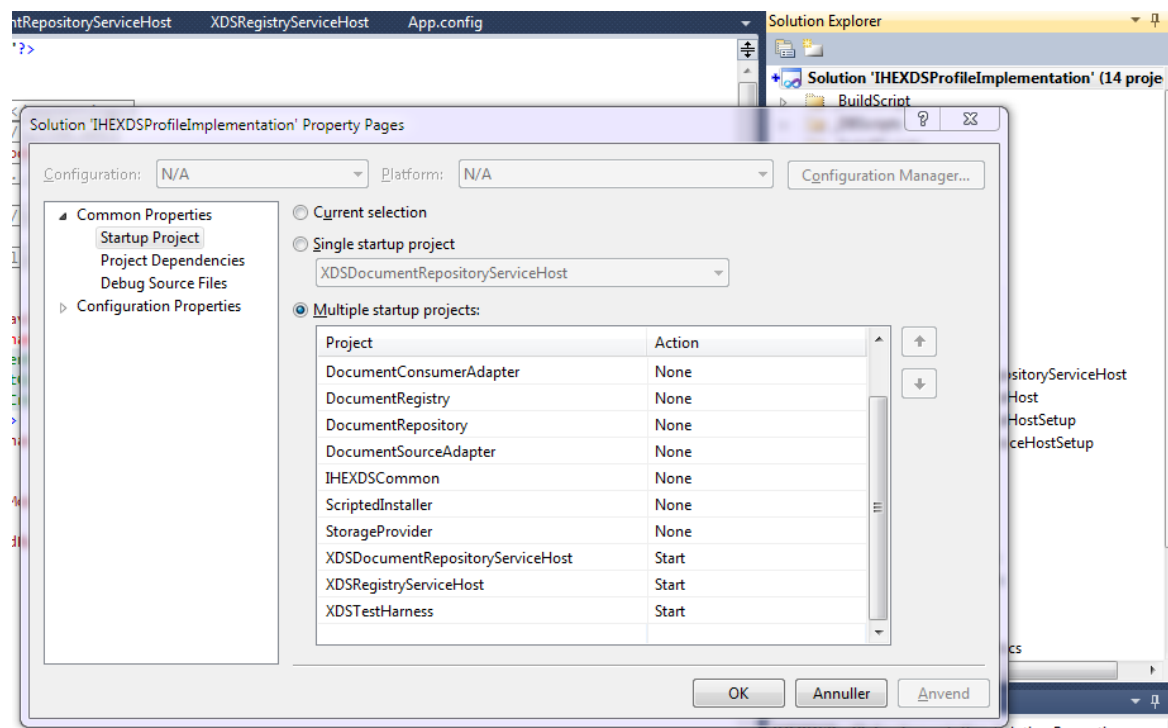
Testapplikationen XDSEstHarness er default kun konfigureret kun til http. Her skal under `configuration.system.serviceModel behaviors endpointBehaviors behavior`

fjernes følgende

```
<clientCredentials>
  <clientCertificate findValue="8c8bc95ec118ec6fbf4d0ec46f13f22e4da619cd"
storeLocation="LocalMachine" storeName="My" x509FindType="FindByThumbprint"/>
</clientCredentials>
```

Solution til test

Vælg Solution i Solution Explorer og tryk alt+Enter og nedenstående dialog fremkommer.



Vælg Multiple startup projekts og sæt XSDDocumentRepositoryServiceHost, XDSRegistryServiceHost og XDSTestHarness til Start.

Nu starter alle tre projekter under debug.