NopCommerce适应多数据库方案

有时候一个项目需要连接多个数据库，以实现不同数据库的数据在同个项目的共享。

如果已经安装了nop，则需要在第二个数据库新建一个表，nop现在无法自动通过迁移来实现第二个或者更多数据库，所以这点需要我们手动去操作。

1、在SQLServer新建一个数据库，我这里新建的是TestDb，表为TestTable。

USE [TestDb]

GO

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[TestTable](

[Id] [int] IDENTITY(1,1) NOT NULL,

[Name] [nvarchar](255) NOT NULL,

[CreatedOnUtc] [datetime] NULL,

CONSTRAINT [PK\_TestTable] PRIMARY KEY CLUSTERED

(

[Id] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

2、添加实体，在Nop.Core下的Domain添加一个文件夹OtherDb，在OtherDb添加一个TestTable类，继承BaseEntity

TestTable：

namespace Nop.Core.Domain.OtherDb

{

public class TestTable : BaseEntity

{

public string Name { get; set; }

public DateTime CreatedOnUtc { get; set; }

}

}

3、添加映射，在Nop.Data下的Mapping添加一个OtherDb文件夹，在OtherDb添加一个TestTableMap映射类，继承NopEntityTypeConfiguration< TestTable >

TestTableMap：

namespace Nop.Data.Mapping.OtherDb

{

public class TestTableMap : NopEntityTypeConfiguration<TestTable>

{

public TestTableMap()

{

ToTable("TestTable");

HasKey(t => t.Id);

Property(t => t.Name).IsRequired().HasMaxLength(255);

}

}

}

4、添加新的DbContext，在Nop.Data添加一个OhterDb，里面添加一个OtherDbObjectContext类继承DbContext和IDbContext，添加一个OtherDbEfStartUpTask继承IStartTask

OtherDbObjectContext：

namespace Nop.Data.OtherDb

{

public class OtherDbObjectContext : DbContext, IDbContext

{

public OtherDbObjectContext(string nameOrConnectionString)

: base(nameOrConnectionString)

{

}

protected override void OnModelCreating(DbModelBuilder modelBuilder)

{

modelBuilder.Configurations.Add(new TestTableMap());

base.OnModelCreating(modelBuilder);

}

public string CreateDatabaseScript()

{

return ((IObjectContextAdapter)this).ObjectContext.CreateDatabaseScript();

}

public new IDbSet<TEntity> Set<TEntity>() where TEntity : Core.BaseEntity

{

return base.Set<TEntity>();

}

public IList<TEntity> ExecuteStoredProcedureList<TEntity>(string commandText, params object[] parameters) where TEntity : Core.BaseEntity, new()

{

throw new **NotImplementedException**();

}

public IEnumerable<TElement> SqlQuery<TElement>(string sql, params object[] parameters)

{

throw new **NotImplementedException**();

}

public int ExecuteSqlCommand(string sql, bool doNotEnsureTransaction = false, int? timeout = null, params object[] parameters)

{

throw new **NotImplementedException**();

}

public void Detach(object entity)

{

throw new **NotImplementedException**();

}

public bool ProxyCreationEnabled

{

get { return this.Configuration.ProxyCreationEnabled; }

set { this.Configuration.ProxyCreationEnabled = value; }

}

public bool AutoDetectChangesEnabled

{

get { return this.Configuration.AutoDetectChangesEnabled; }

set { this.Configuration.AutoDetectChangesEnabled = value; }

}

}

}

OtherDbEfStartUpTask：

namespace Nop.Data.OtherDb

{

public class OtherDbEfStartUpTask : IStartupTask

{

public void Execute()

{

Database.SetInitializer<OtherDbObjectContext>(null);

}

public int Order

{

get { return -1000; }

}

}

5、添加新的数据库连接字符串文本文件，用于连接TestDb数据库，在Nop.Web下面的App\_Data下面添加一个文本文件命名OtherDbSetting.txt里面添加连接字符串

DataProvider: sqlserver

DataConnectionString: Data Source=.;Initial Catalog=TestDb;Integrated Security=False;Persist Security Info=False;User ID=登陆身份;Password=登陆密码

改变数据库连接名即可

6、在autofac注入新的OtherDbObjectContext，在Nop.WebFramework新建一个OtherDb文件夹，里面添加一个DepedencyRegistrar类

DependencyRegistrar：

namespace Nop.Web.Framework.OtherDb

{

public class DependencyRegistrar : IDependencyRegistrar

{

protected virtual string MapPath(string path)

{

if (HostingEnvironment.IsHosted)

{

return HostingEnvironment.MapPath(path);

}

else

{

string baseDirectory = AppDomain.CurrentDomain.BaseDirectory;

path = path.Replace("~/", "").TrimStart('/').Replace('/', '\\');

return Path.Combine(baseDirectory, path);

}

}

public void Register(ContainerBuilder builder, Core.Infrastructure.ITypeFinder typeFinder, Core.Configuration.NopConfig config)

{

var dataSettingManager = new DataSettingsManager();

var dataProviderSettings = dataSettingManager.LoadSettings(Path.Combine(MapPath("~/App\_Data/"), "OtherDbSetting.txt"));

if (dataProviderSettings != null && dataProviderSettings.IsValid())

{

builder.Register<IDbContext>(c => new OtherDbObjectContext(dataProviderSettings.DataConnectionString))

.Named<IDbContext>("nop\_object\_context\_otherdb").InstancePerLifetimeScope();

//builder.Register<OtherDbObjectContext>(

// c => new OtherDbObjectContext(dataProviderSettings.DataConnectionString)).InstancePerLifetimeScope();

}

else

{

builder.Register<IDbContext>(

c => new OtherDbObjectContext(c.Resolve<DataSettings>().DataConnectionString)).InstancePerLifetimeScope();

//builder.Register(

// c => new OtherDbObjectContext(c.Resolve<DataSettings>().DataConnectionString))

// .InstancePerLifetimeScope();

}

builder.RegisterType<EfRepository<TestTable>>()

.As<IRepository<TestTable>>()

.WithParameter(ResolvedParameter.ForNamed<IDbContext>("nop\_object\_context\_otherdb"))

.InstancePerLifetimeScope();

}

public int Order

{

get { return -1; }

}

}

}

全部添加没问题后，添加其他业务实现代码，重新生成解决方案即可。 此方案试用Nop3.4版本及以上，如有跟nop前期或者更新某个版本有差异，请自行斟酌修改。

-- 鱼头鱼尾 QQ：875755898