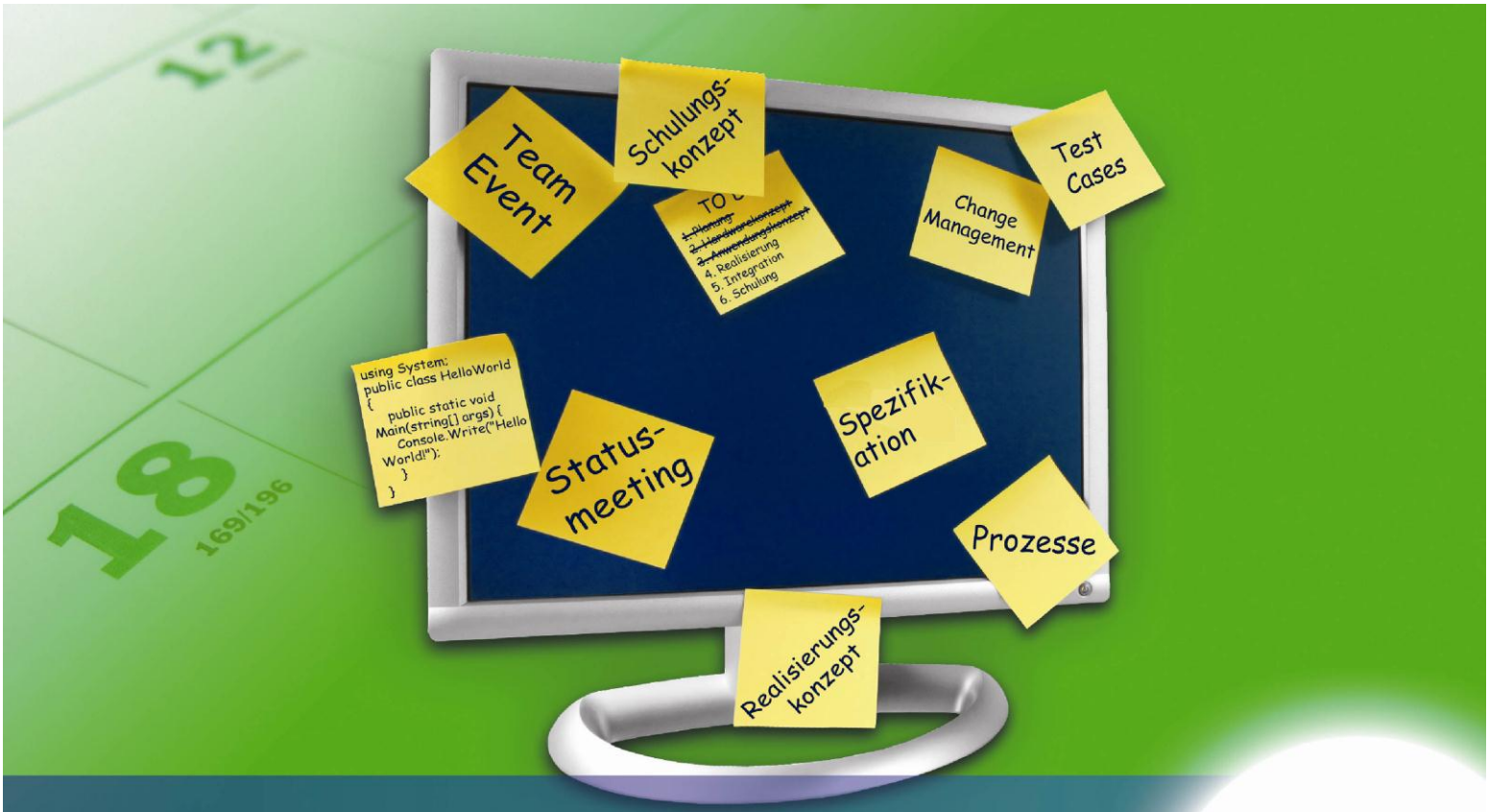


re-store: PhoneBook

Level 100 - Hands on Labs

rubicon
informationstechnologie gmbh

Table of Contents

1 Goal

Goal of this HOL is to teach the fundamentals of re-store, the data storage layer of re-motion. All exercises in this HOL can be solved step by step. Each exercise is designed to be solved with an executable result.

During this HOL you will run into sections with questions. Answers to some questions might be straight forward. Other questions could be tricky. In case you are doing this exercise with other developers, we encourage you to talk about your ideas and your proposed solutions.

After this exercise you will be able to understand the advantages of re-store and to compare them with other O/R mapping tools.

Please have also a look on the further readings section in the end of this HOL. They provide some suggestions on how you might want to proceed to learn more.

In this lab you will create a domain layer to a phone book application. You will start with a class without relations to other classes. You will learn how to automatically create your database script. You will not have to write a single sql script by yourself in the whole HOL. In a second step you learn how to add relation between classes.

In the second and third lab you will get a short introduction to re-linq and client transactions.

After this tutorial you will be able to implement re-store in a project. You will be ready to understand more complex topics and you will be prepared to continue with the re-bind tutorial.

Topics:

- Understand the concept of re-store
- Generate a simple domain layer with re-store
- Generate relations between domain classes
- Generate database scripts with dbschema.exe
- Introduction to client transactions
- Introduction to re-linq

Important:

For this HOL Visual Studio 2010 must be installed on your working PC. Additionally, access to a SQL Server 2008 is required.

We recommend installing the following two third party tools too:

- JetBrains Resharper (http://www.jetbrains.com/resharper/)
- RedGate Reflector (http://reflector.red-gate.com/Download.aspx)

Both tools will help you to understand the samples in this HOL better.

If you have no idea how to get re-motion, in the appendix “Getting re-motion” there is a guide how to get a version of re-motion. All samples were tested with version 1.13.94.

In this HOL we have C:\PhoneBook as basic directory for the HOL session. In case you want to use a different path, keep in mind to adapt the directory names from this HOL to your convention.

2 Lab 1: First steps with re-store

Estimated time: **30 Minutes**

2.1 Preparation

You will require a basic understanding of O/R Mapping technologies to be able to finish this HOL.

Read about O/R Mapping in Wikipedia, if you have not worked with O/R mappers before. Try to get an overview! What is their purpose? Which products are available? Which are the most common products for .NET?

2.2 Exercise – Ten Minutes SQL Script Generation

2.2.1 Task 1 – Ten Minutes Domain Object

1. Start Microsoft Visual Studio 2010.
2. Create a new C# Class Library project. Select platform .NET Framework 3.5. Call it 'PhoneBook.Domain'. This is the base of your domain layer.

   QuickCheck: Solution file should be under c:\PhoneBook, the project file under C:\PhoneBook\PhoneBook.Domain.
3. Press CTRL + SHIFT + A and select class to add a new class named Person to the domain project. Add the following code to the class!

```
using Remotion.Data.DomainObjects;
using Remotion.Data.DomainObjects.ObjectBinding;

namespace PhoneBook.Domain
{
  [DBTable]
  public class Person : BindableDomainObject
  {
    [StringProperty(MaximumLength = 60)]
    public virtual string FirstName { get; set; }

    [StringProperty(IsNullable = false, MaximumLength = 60)]
    public virtual string Surname { get; set; }

    public static Person NewObject()
    {
      return DomainObject.NewObject<Person>();
    }

    public static Person GetObject(ObjectID objid)
    {
      return DomainObject.GetObject<Person>(objid);
    }

    public override string DisplayName
    {
      get { return Surname; }
    }
  }
}
```

4. You will realize that some attributes and classes are marked red. We have to add the following references. If possible stick to the conventions of the chapter Adding references later in this HOL.

- Remotion
- Remotion.Interfaces
- Remotion.Data.Domain.Objects,
- Remotion.Interfaces.
- Remotion.ObjectBinding,
- Remotion.ObjectBinding.Interfaces

5. Build

You have created you first domain. In the next steps you will create your database script based on your domain class.

2.2.2 Task 2 – Using dbschema.exe

1. Open the command prompt.
2. Enter “cd C:\PhoneBook\PhoneBook.Domain\bin\Debug”
3. Call “..\..\..\Remotion\net-3.5\bin\debug\dbschema.exe /schema” (if you do not use the proposed directory schema, please adapt the path)
4. Look for the file SetupDB.sql

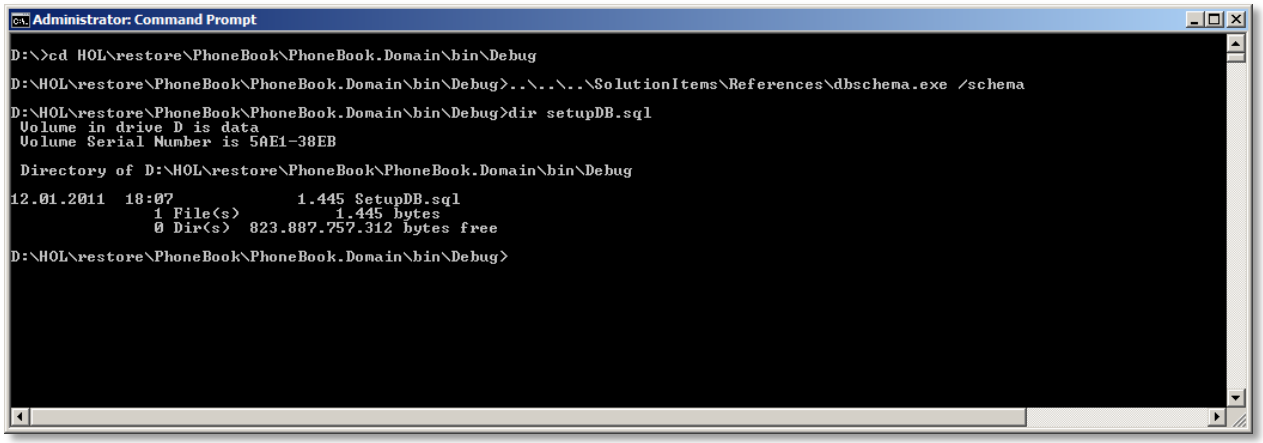


Illustration 1: Using dbschema.exe

2.2.3 Questions / Free Exercises / Discussions

- Try to remove one or two assemblies. Which error messages does the compiler return? Can you explain them?
- DomainObject.NewObject<Person>() is a factory method to create a new object instance. Can you explain the advantages of using factories to create objects via constructors within a few sentences? Does it make sense? If so, how would you try to convince a fellow developer to use factories instead of construction?
- Use Intellisense to investigate the StringProperty attribute. Find out what the meaning of MaximumLength and IsNullable is?
- What happens if you delete the StringProperties attribute, rebuild and rerun the dbschema.exe? Does it build? What has changed in the SetupDB.sql?
- Try to add an attribute Age or Birthday to the Person class.
- Explore the other parameters dbschema.exe provides? What does the verbose parameter do?
- The easiest way is to add the database and its tables with SQL Management studio. But is it also the fastest way? Do you want to open SQL Management Studio for every database change? How could you run this script on the console?
- Why do we call our domain project PhoneBook.Domain and not just Domain? Does this make sense?
- Automate dbschema: Changing to the command line and entering commands can get tedious. How can this be improved?
- **Expert question:** dbschema collects data from compiled assemblies not from source file. In theory, you could also read out c# files. Does this make sense? Discuss!
- **Expert question:** What would happen if you would use dbschema on assemblies compiled in release mode?

- **Expert question:** If you have the Redgate Reflector available, open it and explore the assemblies. Try to find BindableDomainObject and investigate it. Can you explain the purpose of this class?

2.3 Exercise – Adding Relations

In this example, we want to add relations to our sample. We add Country, Location and PhoneNumber as new domain classes.

2.3.1 Task 1 – Adding classes without relation

1. Add the following classes

```
namespace PhoneBook.Domain
{
  public enum Country
  {
    Austria = 0,
    Australia,
    Germany,
    Switzerland
  }
}
```

```
using Remotion.Data.DomainObjects;
using Remotion.Data.DomainObjects.ObjectBinding;

namespace PhoneBook.Domain
{
  [DBTable]
  public class Location : BindableDomainObject
  {
    [StringProperty(IsNullable = false, MaximumLength = 60)]
    public virtual string Street { get; set; }

    [StringProperty(IsNullable = true, MaximumLength = 12)]
    public virtual string Number { get; set; }

    [StringProperty(MaximumLength = 60)]
    public virtual string City { get; set; }

    public virtual int ZipCode { get; set; }

    public static Location NewObject()
    {
      return DomainObject.NewObject<Location>();
    }

    public static Location GetObject(ObjectID objid)
    {
      return DomainObject.GetObject<Location>(objid);
    }

    public override string DisplayName
    {
      get { return Street; }
    }
  }
}
```

```
using Remotion.Data.DomainObjects;
using Remotion.Data.DomainObjects.ObjectBinding;


namespace PhoneBook.Domain
{
  [DBTable]
  public class PhoneNumber : BindableDomainObject
  {
    [StringProperty(MaximumLength = 8)]
    public virtual string CountryCode { get; set; }

    [StringProperty(MaximumLength = 8)]
    public virtual string AreaCode { get; set; }

    [StringProperty(MaximumLength = 12, IsNullable = false)]
    public virtual string Number { get; set; }

    [StringProperty(MaximumLength = 8)]
```

```
    public virtual string Extension { get; set; }

    public static PhoneNumber NewObject()
    {
      return DomainObject.NewObject<PhoneNumber>();
    }

    public static PhoneNumber GetObject(ObjectID objid)
    {
      return DomainObject.GetObject<PhoneNumber>(objid);
    }

    public override string DisplayName
    {
      get { return Number; }
    }
  }
}
```

2. Build your classes and call dbschema.exe again
3. Add Location to Person. Rebuild and call dbschema. What has changed in your generated script file?

```
    public virtual Location Location { get; set; }
```

4. Add Country to Location

```
    public virtual Country? Country { get; set; }
```

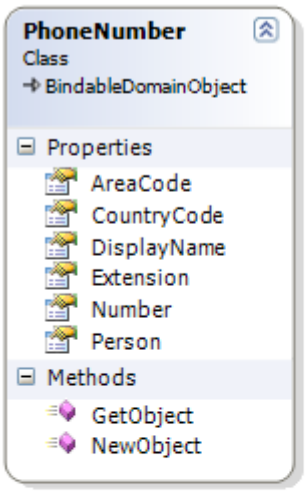
5. Now we add the next code to Person. Does it build? Does dbschema.exe return an error?

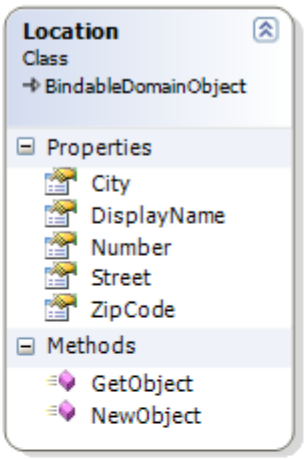
```
    [DBBidirectionalRelation("Person", SortExpression = "CountryCode,AreaCode,Number,Extension")]
    public virtual ObjectList<PhoneNumber> PhoneNumbers { get; set; }
```

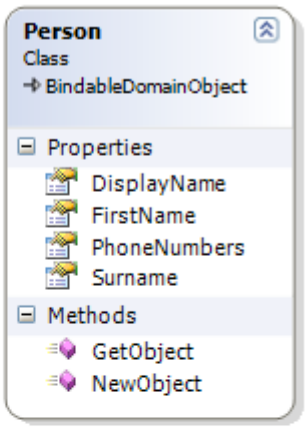
6. Now we add the code to PhoneNumbers. Does it build? Does dbschema.exe return an error?

```
[DBBidirectionalRelation("PhoneNumbers")]
        public virtual Person Person { get; set; }
```

We have now added classes and relations. We have created a domain model that contains the following classes:







2.3.2 Questions / Discussion / Exercises

- Does it make sense to organize the source code (Such as putting properties, relations or factory methods in regions)?
- There are two attributes to consider: Mandatory and StorageClassNone. Mandatory specifies that the property is mandatory and StorageClassNone that this property shall not be stored in the database. Apply them in the code, generate code and see what happens.
- If you open SetupSQL.db: Why is LocationID nullable?
- Which of the following is a “value object”? Person / Country? How would the other entity be called?
- If you have thought about “value type”, please look up the term “value object” with “DDD” or “Domain Driven Design”. Can you explain the difference between “value type” and “value object”?
- In the last example we saw that you might run into problems which won’t be detected by the compiler. The error gets obvious by generating classes with dbschema.exe. What can you do to make sure that this mapping validation error is detected as soon as possible? Does it make sense to add a dbschema generation to your buildscript?
- **Expert Question:** Re-store identity columns rely on GUIDs. Does this make sense? Is an INT in your opinion a better alternative?
- **Expert Question:** There are inheritance modes for databases: Single Table Inheritance / Class Table Inheritance / Concrete Table Inheritance. Find out what they do? Which model are we using?

2.4 Exercise – Sample Console App

1. Add a console application
2. Add new item “Application Configuration File”
3. If not done yet create your SQL Database for this sample. We recommend naming it “Phonebook”. Add tables and views with the generated SetupDB.sql script. Be aware that you have to adapt the use statement in the first line of the SetupDB.sql script to match the name of your database.
4. Add assembly references to the console. Also add a reference to the project of your domain object.
5. Add the following code for the program resp. application configuration file

```
using PhoneBook.Domain;
using Remotion.Data.DomainObjects;

namespace PhoneBook.Console
{
  class Program
  {
    static void AddSampleData()
    {
      using (ClientTransaction.CreateRootTransaction().EnterDiscardingScope())
      {
        var loc = Location.NewObject();
        loc.Street = "Microsoft Way";
        loc.Number = "4";
        loc.City = "Redmond";
        loc.ZipCode = 1110;

        var p1 = Person.NewObject();
        p1.FirstName = "Steve";
        p1.Surname = "Ballmer";
        p1.Location = loc;

        var pn1 = PhoneNumber.NewObject();
        pn1.CountryCode = "001";
        pn1.AreaCode = "425";
        pn1.Number = "705-1900";
        pn1.Person = p1;

        var p2 = Person.NewObject();
        p2.FirstName = "Scott";
        p2.Surname = "Guthrie";
        p2.Location = loc;
```

```
        var pn2 = PhoneNumber.NewObject();
        pn2.CountryCode = "001";
        pn2.AreaCode = "425";
        pn2.Number = "705-1901";
        pn2.Person = p2;

        ClientTransaction.Current.Commit();
      }
    }

    static void Main(string[] args)
    {
      AddSampleData();
    }
  }
}
```

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <configSections>
    <sectionGroup name="remotion.data.domainObjects"
type="Remotion.Data.DomainObjects.Configuration.DomainObjectsConfiguration,
Remotion.Data.DomainObjects">
      <section name="storage"
type="Remotion.Data.DomainObjects.Persistence.Configuration.StorageConfiguration,
Remotion.Data.DomainObjects" />
    </sectionGroup>
  </configSections>

  <remotion.data.domainObjects xmlns="http://www.re-
motion.com/Data/DomainObjects/Configuration/2.0">
    <storage defaultProviderDefinition="PhoneBookDB">
      <providerDefinitions>
        <add name="PhoneBookDB"
              type="Remotion.Data.DomainObjects::Persistence.Rdbms.RdbmsProviderDefinition"
              providerType="Remotion.Data.DomainObjects::Persistence.Rdbms.SqlProvider"
              connectionString="PhoneBook"
factoryType="Remotion.Data.DomainObjects::Persistence.Rdbms.SqlServer.Sql2005.SqlStorageObjectFac
tory"
            />
      </providerDefinitions>
    </storage>

  </remotion.data.domainObjects>

  <connectionStrings>
    <add name="PhoneBook" connectionString="Integrated Security=SSPI; Initial Catalog=PhoneBook;
Data Source=localhost" />
  </connectionStrings>
</configuration>
```

Run the sample and verify that the expected content is really stored in the database.

2.4.1 Questions / Free Exercises / Discussions

- What is required to add Intellisense to the app.config file?
- **Expert question:** Try to remove the virtual key word from one of the properties of Person and try to restart the app. There is an exception. Can you explain the reason for this?

3 Lab 2 – Client Transactions

Estimated time: **30 Minutes**

In this Lab we will dig deeper into Client Transactions. We will learn how Client Transactions are created and how objects can be enlisted into different transactions. We will also have a look on Sub Client Transactions as part of transaction handling.

3.1 Preparations

Database transactions are an essential mechanism to ensure data consistency. If you are new to this topic and have no experience with transactions, please try to get an overview before reading on. There are a large number of articles on transactions in the internet.

3.2 Exercise 1 – Understanding Client Transactions

3.2.1 Task 1 – Factories

Adapt the sample console application. Try to run the following code:

```
static void Main(string[] args)
{
  var loc = Location.NewObject();
}
```

What do you expect what will happen? Is an object of Location really instantiated?

3.2.2 Question

- To get rid of the exception at the factory method, you have to create a Client Transaction. Can you explain the difference between the following operations?

```
ClientTransaction.CreateRootTransaction().EnterDiscardingScope();
Location.NewObject();
```

and

```
using (ClientTransaction.CreateRootTransaction().EnterDiscardingScope())
{
  Location.NewObject();
}
```

- We have learned that a domain object cannot live outside a transaction. Do you think this approach is a good idea? Try to find pros and cons and discuss about them.

3.2.3 Task 2 – Adding members

Try to implement the following:

```
ClientTransaction.CreateRootTransaction().EnterDiscardingScope();
Location.NewObject();
int count1 = ClientTransaction.Current.EnlistedDomainObjectCount;
ClientTransaction.CreateRootTransaction().EnterDiscardingScope();
int count2 = ClientTransaction.Current.EnlistedDomainObjectCount;
```

- Find out, if count1 has the same value as count2? Can you explain, why they have different values?

3.2.4 Question

- What needs to be done so that count1 matches count2? **Hint**: Check out static methods of `ClientTransaction.Current`. Is there a method that will help you to solve the problem?
- The return value of `Location.NewObject()` is not assigned to a member or a variable. Is there a way to access the object without assigning the return value of `Location.NewObject()` to a variable?

3.2.5 Task 3 – Manipulation just on persisted

Please add the following code to your console application.

```
private static void Demo()
{
  ClientTransaction.CreateRootTransaction().EnterDiscardingScope();
  var l = Location.NewObject();
  // ClientTransaction.Current.Commit();

  int i1 = ClientTransaction.Current.EnlistedDomainObjectCount;
  using (ClientTransaction.CreateRootTransaction().EnterDiscardingScope())
  {
    ClientTransaction.Current.EnlistDomainObject(l);
    int i2 = ClientTransaction.Current.EnlistedDomainObjectCount;
    l.Street = "test";
  }
}
```

Try to explain the purpose of this code! The solution for the problem is obviously “marked”. Can you explain why uncommenting the fifth line solves the problem?

3.2.6 Task 4 – Discarding Scope

We have now a look the ClientTransactionScope. Try out the following example:

```
ClientTransaction.CreateRootTransaction().EnterDiscardingScope();
var l = Location.NewObject();
ClientTransaction.Current.Commit();

using (ClientTransaction.Current.EnterNonDiscardingScope())
{
  l.ZipCode = 11;
}
using (ClientTransaction.Current.EnterDiscardingScope())
{
  l.ZipCode = 12;
}

var obj = ClientTransaction.Current.GetEnlistedDomainObjects();
string res = l.ZipCode;
```

3.2.7 Question / Free Exercises / Discussions

- Try to explain: What do you think happens in the sample?
- There is also a third option to get into a scope. Discuss on the parameter list of ClientTransaction.Current.EnterScope(). Try also to use the Reflector to see what code does?
- Why do we need a class ClientTransactionScope anyway? Would it not be better just to work within scopes by usings?
- We might add something like

```
using (ClientTransaction.Current.EnterDiscardingScope())
{
   l.ZipCode = 12;
   DoCommit()
}


public void DoCommit()
{
   ClientTransaction.Current.Commit();
}
```

Do you think it is a good strategy to call commits in public methods? What could happen, if there is a commit in a public method?

3.3 Exercise 2 - SubTransactions

The following three code pieces are different in small details. Which results would you expect?

Try them out. What would you conclude?

```
using (ClientTransaction.CreateRootTransaction().EnterDiscardingScope())
{
  var loc = Location.NewObject();
  loc.City = "Vienna";

  using (ClientTransaction.Current.CreateSubTransaction().EnterDiscardingScope())
  {
    ClientTransaction.Current.EnlistDomainObject(loc);
    loc.City = "Berlin";
  }
  ClientTransaction.Current.Commit();
}
}

using (ClientTransaction.CreateRootTransaction().EnterDiscardingScope())
{
  var loc = Location.NewObject();
  loc.City = " Vienna ";

  using (ClientTransaction.Current.CreateSubTransaction().EnterDiscardingScope())
  {
    ClientTransaction.Current.EnlistDomainObject(loc);
    loc.City = "Berlin";
    ClientTransaction.Current.Commit();
  }
  ClientTransaction.Current.Commit();
}
```

```
using (ClientTransaction.CreateRootTransaction().EnterDiscardingScope())
{
  var loc = Location.NewObject();
  loc.City = "Vienna";

  using (ClientTransaction.Current.CreateSubTransaction().EnterDiscardingScope())
  {
    ClientTransaction.Current.EnlistDomainObject(loc);
    loc.City = "Berlin";
    ClientTransaction.Current.Commit();
  }
  ClientTransaction.Current.Commit();
}
```

4 Lab 2 – re-linq

4.1 Exercise 1 – First re-linq queries

4.1.1 Task – Report

We want to add a report to the sample console app. So we add the following code to the program file.

Be sure to add Remotion.Data.Linq to your project references and add `using System.Linq;`

```
static void Report()
{
  using (ClientTransaction.CreateRootTransaction().EnterDiscardingScope())
  {
    foreach (var l in Location.GetLocations())
    {
      System.Console.WriteLine("Location: {0}", l.DisplayName);
      foreach (var p in l.FindPersons())
      {
        System.Console.WriteLine(" ---> Person: {0} ", p.DisplayName);
        foreach (var pn in p.PhoneNumbers)
        {
          System.Console.WriteLine(" ------------------> Tel: {0}", pn.DisplayName);
        }
      }
    }
  }
  System.Console.ReadKey();
```

And we add the following lines to Location.cs:

```
public static Location[] GetLocations()
{
  var expr = from l in QueryFactory.CreateLinqQuery<Location>()
             select l;
  return expr.ToArray();
}

public Person[] FindPersons()
{
  var expr = from p in QueryFactory.CreateLinqQuery<Person>()
             where p.Location == this
             select p;
  return expr.ToArray();
}
```

We have created our first re-linq query.

4.2 Further studies

There is a whitepaper on re-linq on https://www.re-motion.org/download/re-linq.pdf. It explains what re-linq does and how it can help you to improve your queries.

5 Final Questions

- What are the advantages or disadvantages of working with a GUI designer, such as a GUI designer from the ADO.NET Entity Framework? Are you faster with a designer or with a code?

- Which alternative O/R Mapper do you know? Can you name some differences to re-store?
- Look up naked objects architecture in Wikipedia? Is re-motion based on a naked objects architecture?
- Re-store uses in many cases lazy loading. Can you explain the difference between lazy loading and eager loading?
- Do you think re-store would be compatible with Mono?
- Bidirectional relations:

  You might want to try the following

```
phoneNumber.Person = person;
person.PhoneNumbers.Add(pn);

person.PhoneNumbers.Add(pn);
phoneNumber.Person = person;
```

The first statement throws an exception: "Cannot add object 'PhoneNumber|a9b8188a-d290-45b0-926a-8bd821832bdf|System.Guid' already part of this collection."

The second does not throw an exception. Why?

- **Expert questions:** On http://madgeek.com/Articles/ORMapping/EN/mapping.htm you will find a selection guide for O/R Mapper. Which of these features are available in re-store?
- **Expert questions:** By default there is a MSSQL provider for re-store. What could be required to implement a provider for a different database?

6 HOL Summary

In this HOL you were able to

- Understand basic concepts of re-store
- Generate a domain layer
- Generate database scripts
- Understand basic concepts of client transactions
- Understand basic concepts of re-linq

Now you might want to

- Go to re-bind hol to generate a web application based on your domain
- Dig a little bit deeper into the topic .NET Enterprise Application. You probably want to read books like
  - http://www.amazon.com/dp/073562609X/ref=rdr_ext_sb_pi_sims_1#_
  - http://www.amazon.com/Professional-Enterprise-NET-Wrox-Programmer/dp/0470447613/ref=sr_1_4?s=books&ie=UTF8&qid=1295011417&sr=1-4
- Read Whitepapers to re-linq and have a look on the blogs.

7 Common Practices

7.1 Getting re-motion

For all examples, the re-motion binaries are required. You can either build them or download them via the build page (https://www.re-motion.org/builds/)

If you are not sure where to put the binaries, please read Adding references

If you want to build re-motion, you have to get a version from the following subversion repository: https://svn.re-motion.org/svn/Remotion/.

In most cases, the trunk build is sufficient. If you do not want to work with a trunk build, you might want to get the build with that this HOL was tested: https://svn.re-motion.org/svn/Remotion/tags/1.13.94.

Please read „How to build.txt“(found in the base directory of the repository) before your build.

7.2 Adding references

There are several ways to add binaries in folder hierarchy in a project.

In this sample, we copy the content of the zip file of the download of re-motion.org/builds into the directory C:\PhoneBook\remotion