

NHibernate源代码浅读 PDF转换可能丢失图片或格式，建议
阅读原文

https://www.100test.com/kao_ti2020/472/2021_2022_NHibernate_c67_472939.htm 曾经了解过Hibernate, 印象很深,是个很不错得O/R Mapping FW. 在<http://nhibernate.sourceforge.net/>上有个从Java移植过来得.NET版本--NHibernate(以下称NH),不过目前还处于PreAlpha Build 2阶段. 出于兴趣以及学习.NET得目的, 花了两天仅看了很小一部分代码:一来源代码注释并不丰富,二来对于Hibernate/NHibernate的使用也很不熟悉,三来有些知识点还不熟悉.准备工作如下:1. NHibernate PreAlpha Build 22.

NUnit3. NHibernate配置文件 monitoring.dll.config 如下:可以看出以上是一个标准得Config文件,一般由System.Configuration.ConfigurationSettings.GetConfig方法来读取. 蓝色部分才是真正配置NH的地方,例子中我配置它使用SQL Server, 那些Key/Value的含义很好明白. 值得注意得是, 配置文件得文件名很重要,通常对于一个EXE得Assembly来说,是AssemblyName.Config,不过对于Dll Assembly来,对应的配置文件为AssemblyName.dll.config 例如:MyAssy.exe ->

MyAssy.configMyAssy.dll -> MyAssy.dll.config 我打算在我的monitoring.dll,一个用来监视性能东东中使用NH来持久化数据. 该类库包含了一个TestCase,由NUnit来调用4.将要被持久化的对象,即Business Object(BO)using System.namespace

```
Argus.Monitoring{ public class Monitor { //dbID & DBID  
是NH必须要求的主键 private int dbID. public int  
DBID{set{dbID=value.}get{return dbID.}} public string  
MonitorType {get{return "dummy monitor".}} public double Value
```

```
{get{return 12.34.} public string Category{get{return "this is category".}} public string Name{get{return "this is name".}} public string Instance{get{return "this is instance".}} public string Computer{get{return "this is computer".}} }
```

这是一个被极度简化的类,省略了Member Method,甚至Property的set方法,因为我打算先试试Insert功能,然后再尝试Load功能5. 写一个该BO对应的最简单的映射文件

100Test 下载频道开通,各类考试题目直接下载。详细请访问 www.100test.com