

您的潜力，我们的动力

**Microsoft**  
微软(中国)有限公司

# C#面向对象设计模式纵横谈

## 18. Iterator 迭代器（行为型模式）

李建忠

[jianzhong.lee@gmail.com](mailto:jianzhong.lee@gmail.com)

设计模式论坛:

[forum.softcompass.com](http://forum.softcompass.com)

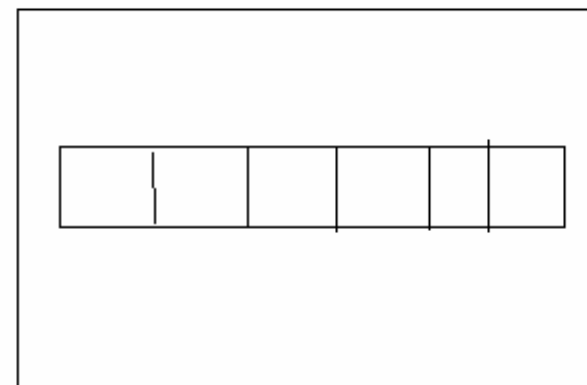
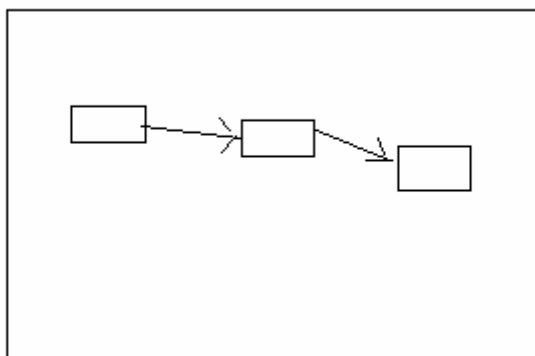
上海祝成科技 高级培训讲师

[www.softcompass.com](http://www.softcompass.com)

您的潜力，我们的动力

**Microsoft**  
微软(中国)有限公司

# 集合内部结构与外部访问



# 动机 (Motivation)

在软件构建过程中，集合对象内部结构常常变化各异。但对于这些集合对象，我们希望在暴露其内部结构的同时，可以让外部客户代码透明地访问其中包含的元素；同时这种“透明遍历”也为“同一种算法在多种集合对象上进行操作”提供了可能。

使用面向对象技术将这种遍历机制抽象为“迭代器对象”为“应对变化中的集合对象”提供了一种优雅的方式。

# 意图 (Intent)

提供一种方法顺序访问一个聚合对象中的各个元素，  
而又不暴露该对象的内部表示。

——《设计模式》GoF

# 例说Iterator应用

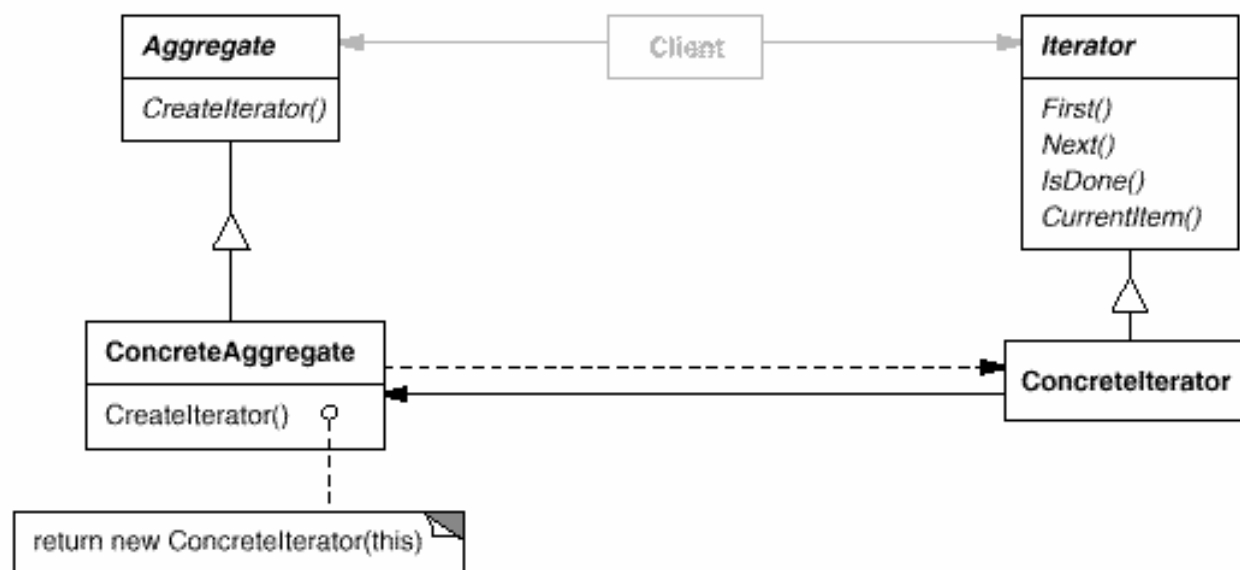
您的潜力，我们的动力

**Microsoft**  
微软(中国)有限公司

**Codes in VS.NET**



# 结构 (Structure)



# Iterator模式的几个要点

- 迭代抽象：访问一个聚合对象的内容而无需暴露它的内部表示。
- 迭代多态：为遍历不同的集合结构提供一个统一的接口，从而支持同样的算法在不同的集合结构上进行操作。
- 迭代器的健壮性考虑：遍历的同时更改迭代器所在的集合结构，会导致问题。

您的潜力，我们的动力

**Microsoft**<sup>®</sup>  
微软(中国)有限公司

# .NET 2.0 中的Iterator

Codes in VS.NET



# 推荐资源


- 《设计模式：可复用面向对象软件的基础》 GoF
- 《面向对象分析与设计》 Grady Booch
- 《敏捷软件开发：原则、模式与实践》 Robert C. Martin
- 《重构：改善既有代码的设计》 Martin Fowler
- 《Refactoring to Patterns》 Joshua Kerievsky

## 获取更多MSDN资源

- **MSDN中文网站**  
<http://www.microsoft.com/china/msdn>
- **MSDN中文网络广播**  
<http://www.msdnwebcast.com.cn>
- **MSDN Flash**  
<http://www.microsoft.com/china/newsletter/case/msdn.aspx>
- **MSDN开发中心**  
<http://www.microsoft.com/china/msdn/DeveloperCenter/default.mspx>


# Question & Answer

如需提出问题，请单击“提问”按钮并在随后显示的浮动面板中输入问题内容。一旦完成问题输入后，请单击“提问”按钮。

 **问题和解答 (无问题)** ▲ ×

在此会议中尚未解答任何问题。

要向演示者提问，请在此处键入问



提问(A)

删除(D)

问题管理器(Q)

您的潜力，我们的动力

**Microsoft®**  
微软(中国)有限公司

**Microsoft®**

msdn  


**MSDN Webcasts**