

# UML for .NET Developers

## Implementation Diagrams, Packages & Model Management

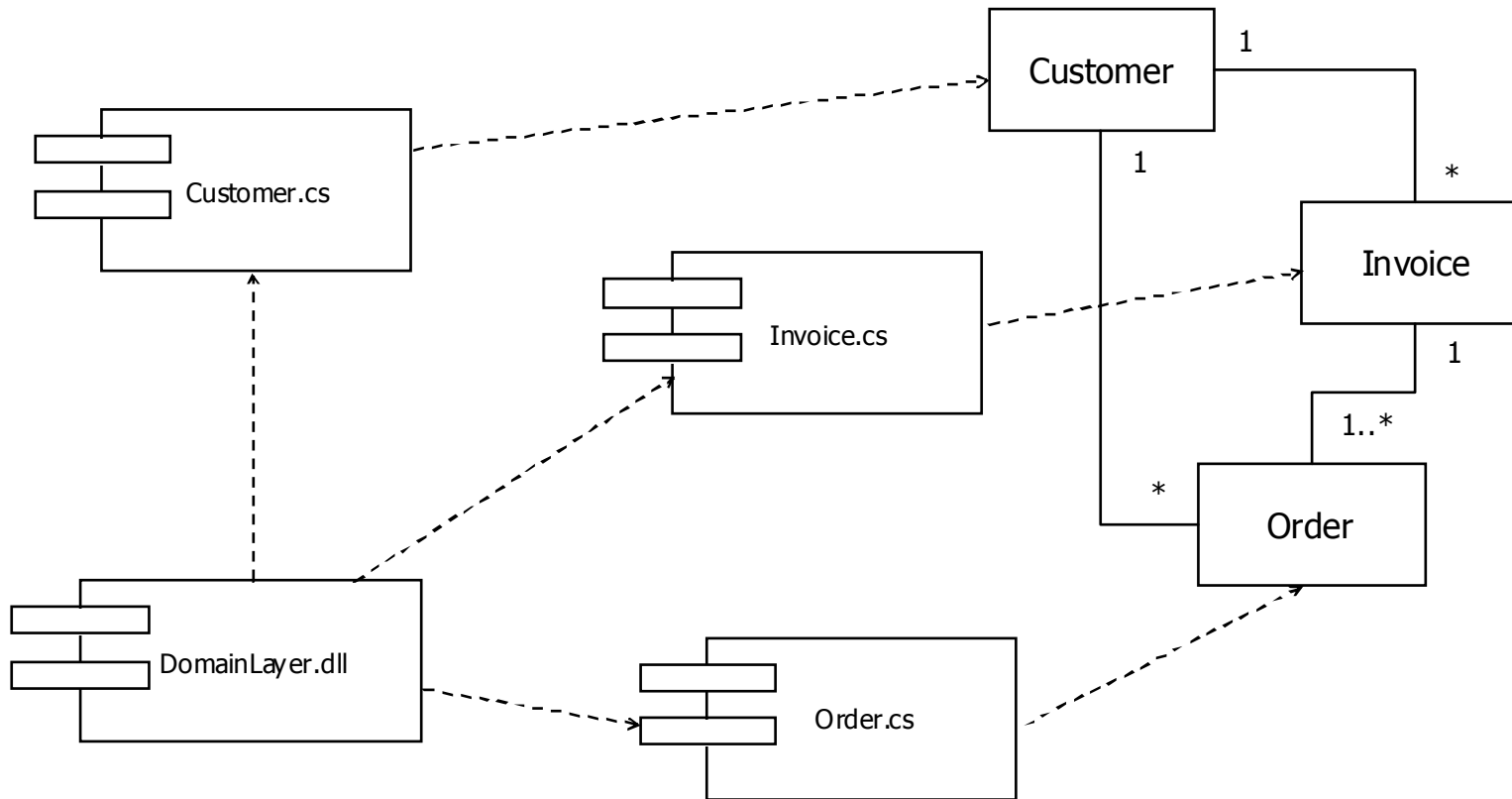
Jason Gorman



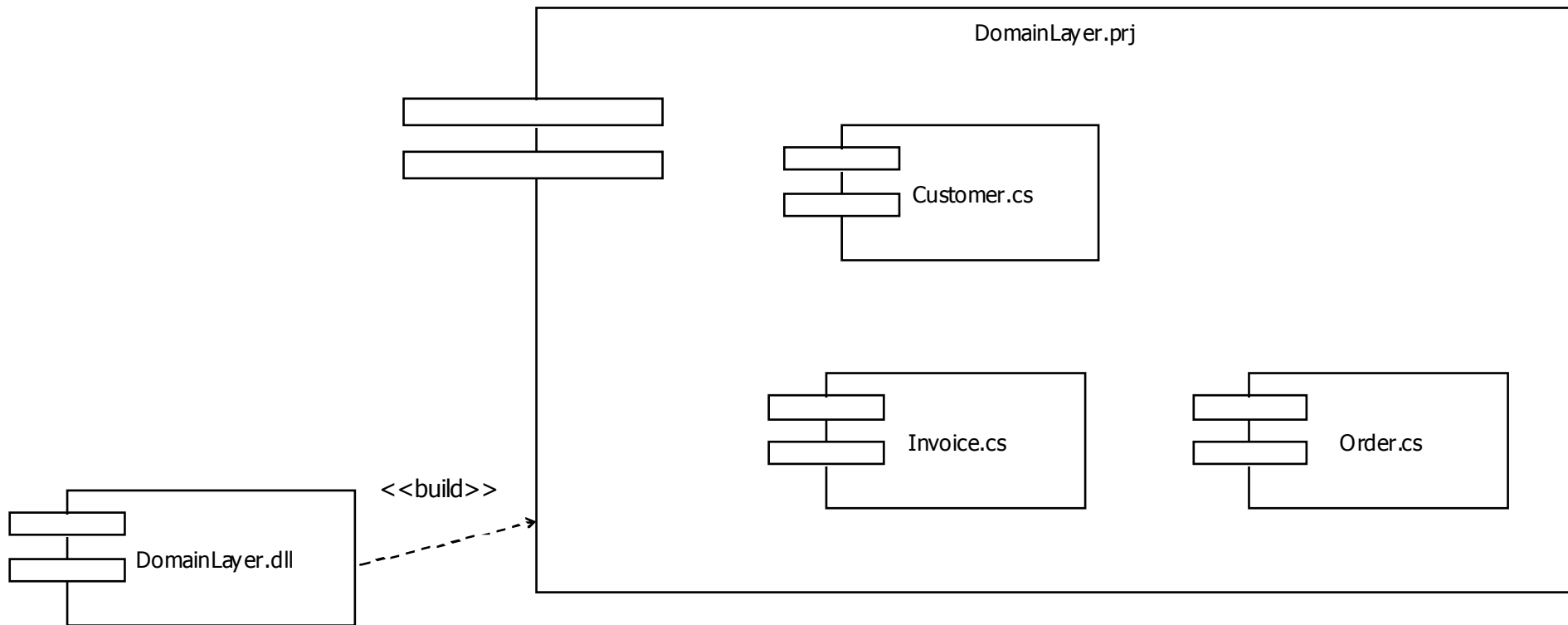
## Accelerated learning for C# developers

Get the UML for .NET Premium Package at  
<http://www.parlezuml.com/tutorials/umlfordotnet.htm>

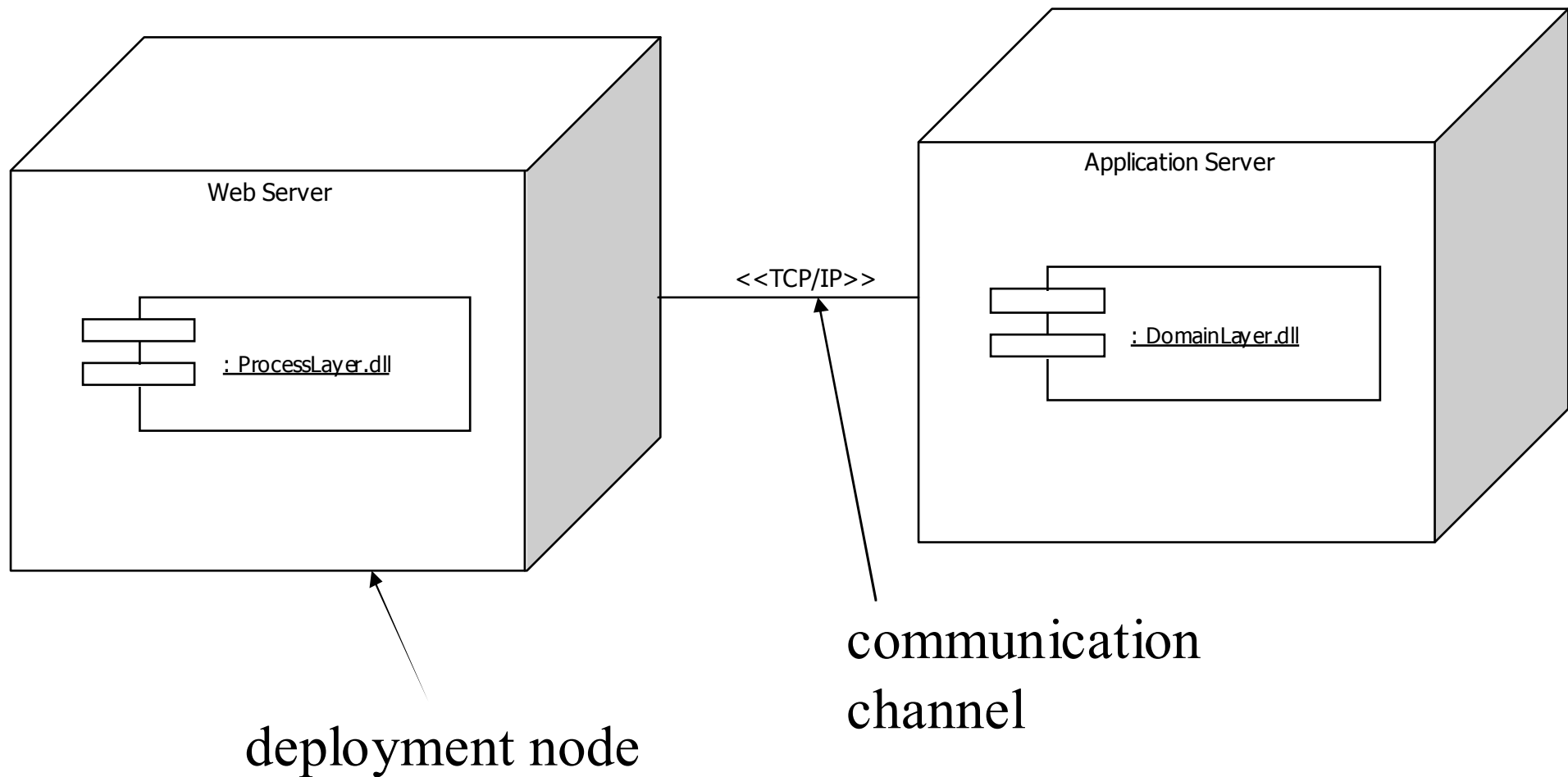
# Components Are Physical Files



# Components Can Contain Components



# Instances of Components Can be Deployed





## Accelerated learning for C# developers

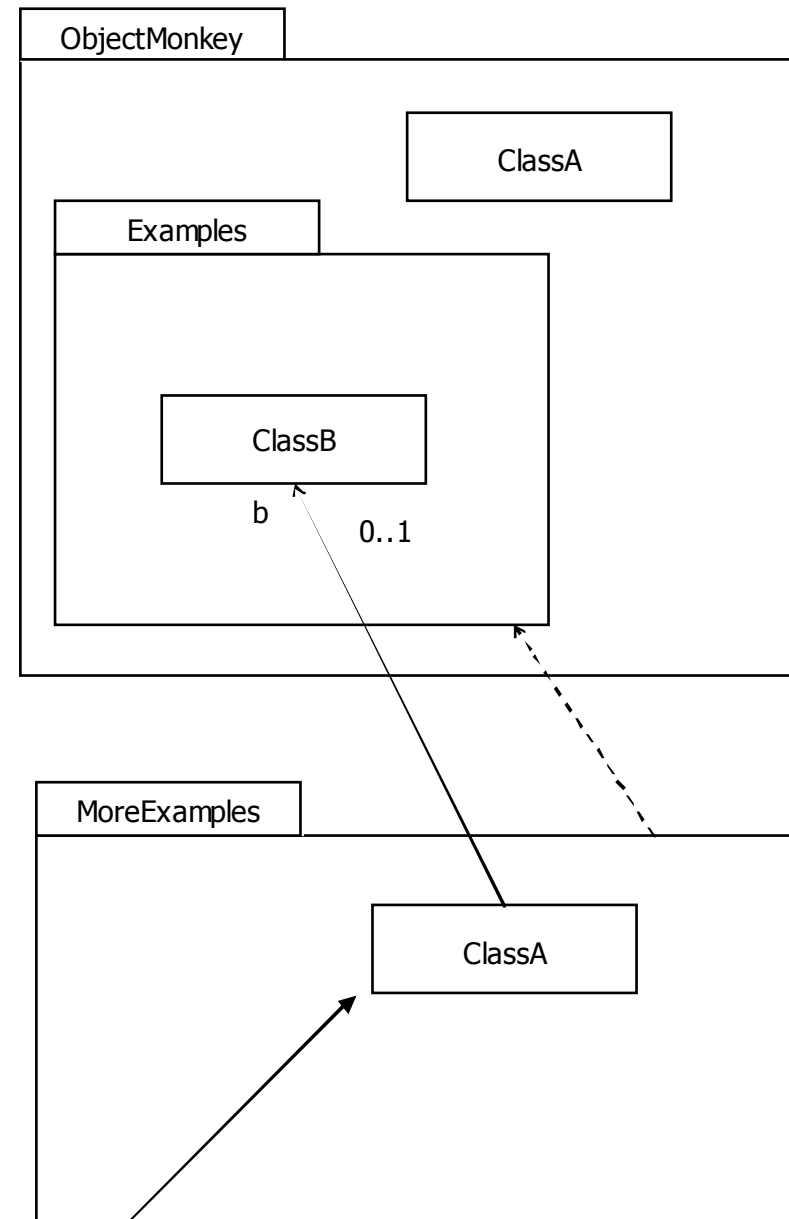
Get the UML for .NET Premium Package at  
<http://www.parlezuml.com/tutorials/umlfordotnet.htm>

# Packages & Namespaces

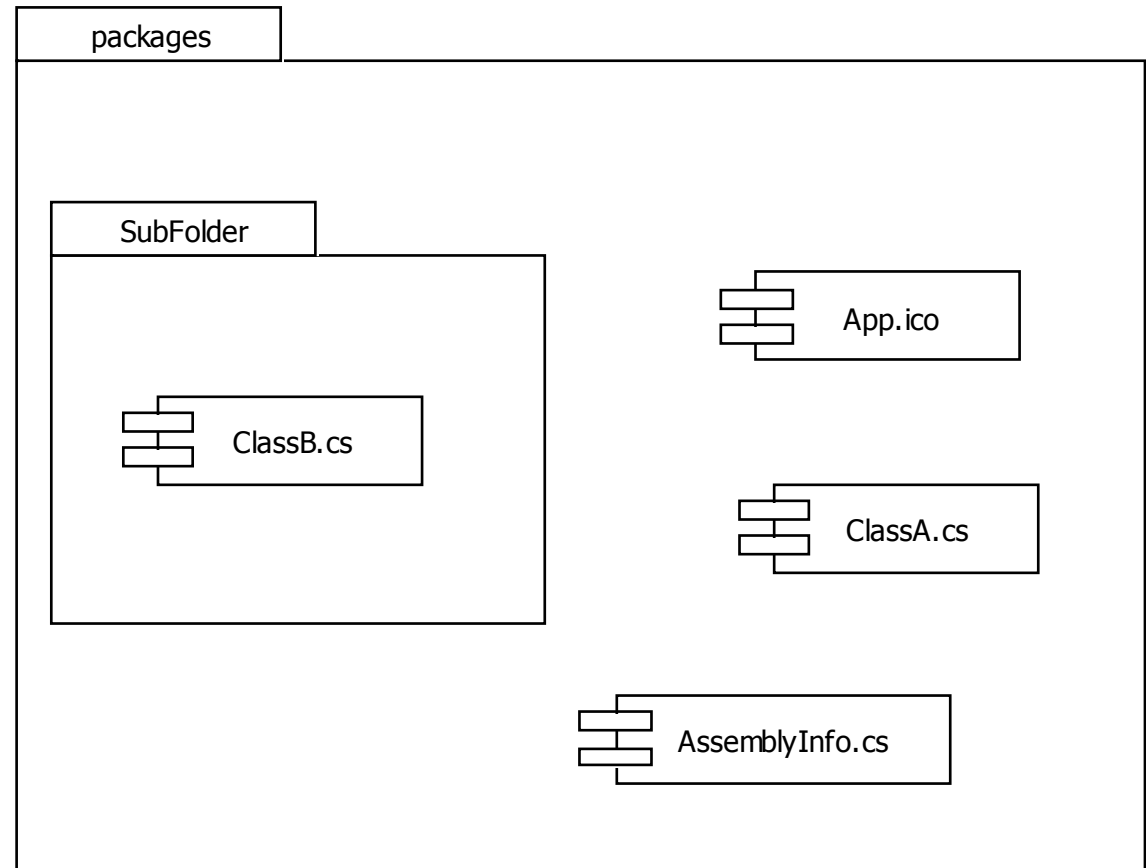
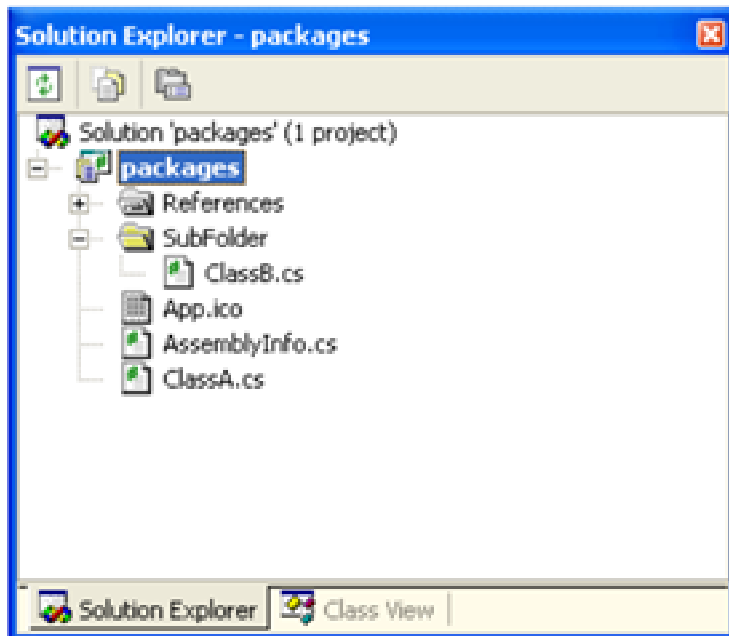
```
namespace ObjectMonkey
{
    class ClassA
    {
    }
    namespace Examples
    {
        class ClassB
        {
        }
    }
}

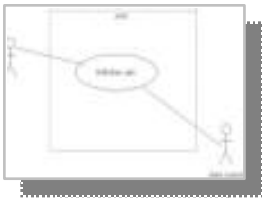
namespace MoreExamples
{
    using ObjectMonkey.Examples;

    class ClassA
    {
        private ClassB b;
    }
}
```



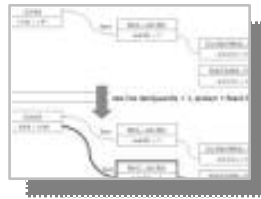
# Packages & Folders





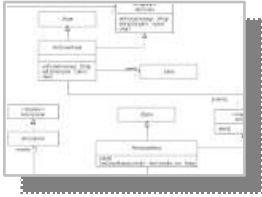
### Use Case Diagrams

Model the users of the system and the goals they can achieve by using it



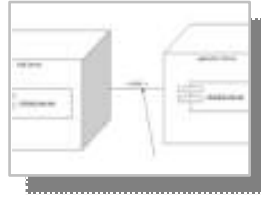
### Object Diagrams & Filmstrips

Model snapshots of the running system and show how actions change object state



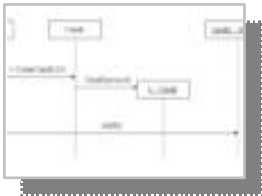
### Class Diagrams

Model types of objects and the relationships between them.



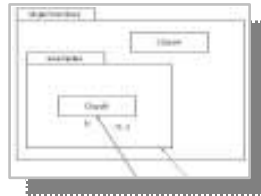
### Implementation Diagrams

Model the physical components of a system and their deployment architecture



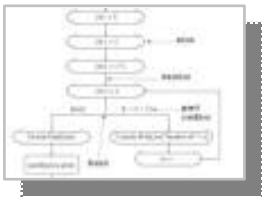
### Sequence Diagrams

Model how objects interact to achieve functional goals



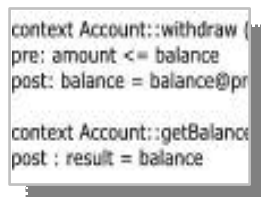
### Packages & Model Management

Organise your logical and physical models with packages



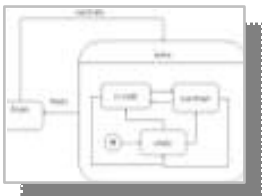
### Activity Diagrams

Model the flow of use cases and single and multi-threaded code



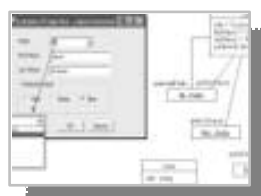
### Object Constraint Language

Model business rules and create unambiguous specifications



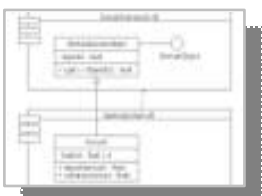
### Statechart Diagrams

Model the behaviour of objects and event-driven applications



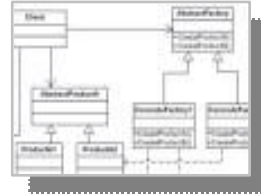
### User Experience Modeling

Design user-centred systems with UML



### Design Principles

Create well-designed software that's easier to change and reuse



### Design Patterns

Apply proven solutions to common OO design problems



# UML for .NET Premium Package

Available exclusively from [www.parlezuml.com](http://www.parlezuml.com)