

Code Contracts

Mikael Lundin

Valtech

Design by Contract

- Bertrand Meyer, 1986



Design by Contract



Design by Contract

- Correctness

Correct software does what it is supposed to do ...

... and only that

- Rubustness

Rubust software acts acceptable in cases in which it cannot do what it was supposed to do

Design by Contract

- Pre-condition
- Post-condition
- Invariant

Hur har vi gjort fram tills nu?

```
public void Remove(Page page)
{
    if (page == null)
    {
        throw new ArgumentNullException("page");
    }

    for (int i = 0; i < data.Length; i++)
    {
        if (data[i] == page)
        {
            data[i] = null;
            return;
        }
    }
}
```

Hur har vi gjort fram tills nu?

```
System.Diagnostics.Debug.Assert(page != null,  
    "Page should not be null in this situation");
```

```
<configuration>  
  <system.diagnostics>  
    <assert assertuienabled="true"  
      logfile="c:\\myFile.log" />  
  </system.diagnostics>  
</configuration>
```

Microsoft Code Contracts

- Nedladdning

<http://research.microsoft.com/contracts>

- Add Reference...

- Microsoft.Contracts.dll

Microsoft Code Contracts

- VS2010 Premium / VS2008 Team Suite
Static verification
- Standard
Runtime verification
- Academic Licence
Static verification, non-commercial

Pre-conditions

```
public class PageCollection
{
    private Page[] data = new Page[10];
    private int count = 0;

    public void Add(Page page)
    {
        if (count == data.Length)
        {
            Expand();
        }

        data[count++] = page;
    }
}
```

Pre-conditions

```
public void Add(Page page)
{
    Contract.Requires<ArgumentNullException>(page != null);

    if (count == data.Length)
    {
        Expand();
    }
    data[count++] = page;
}
```

Pre-conditions demo

```
[Test]
public void ShouldRequireNonNullArgument ()
{
    /* Setup */
    var pageCollection = new PageCollection();

    /* Test */
    pageCollection.Add(null);

    /* Assert */
}
```

Error List					
0 Errors 2 Warnings 1 Message					
	Description	File	Line	Column	Project
1	CodeContracts: requires is false: page != null	Add.cs	19	13	CodeContractsExample
2	+ location related to previous warning	PageCollection.cs	13	13	CodeContractsExample

Pre-conditions

```
[Test]
public void ShouldRequireNonNullArgument ()
{
    /* Setup */
    var pageCollection = new PageCollection();

    /* Test */
    pageCollection.Add(null);

    /* Assert */
}
```

Unit Test Sessions - Session #1

Session #1

Group by: Projects and Namespaces

Tests failed: 1, passed: 0, ignored: 0

- <CodeContractsExample> (1 test) 1 test failed
 - CodeContractsExample.Tests.PageCollectionTest (1 test) 1 test failed
 - Add (1 test) 1 test failed
 - ShouldRequireNonNullArgument Failed: ContractException: Precondition failed: page 1 = null

Pre-conditions

```
public void Add(Page page)
{
    if (page == null)
    {
        throw new ArgumentNullException("page");
    }
    Contract.EndContractBlock();

    if (count == data.Length)
    {
        Expand();
    }

    data[count++] = page;
}
```

Post-conditions

```
public Page this[int index]
{
    get
    {
        Contract.Requires<IndexOutOfRangeException>(index > -1);
        Contract.Ensures(Contract.Result<Page>() != null);

        return data[index];
    }
}
```

Post-conditions

```
public void Add(Page page)
{
    Contract.Requires<ArgumentNullException>(page != null);
    Contract.Ensures(count == Contract.OldValue(count) + 1);

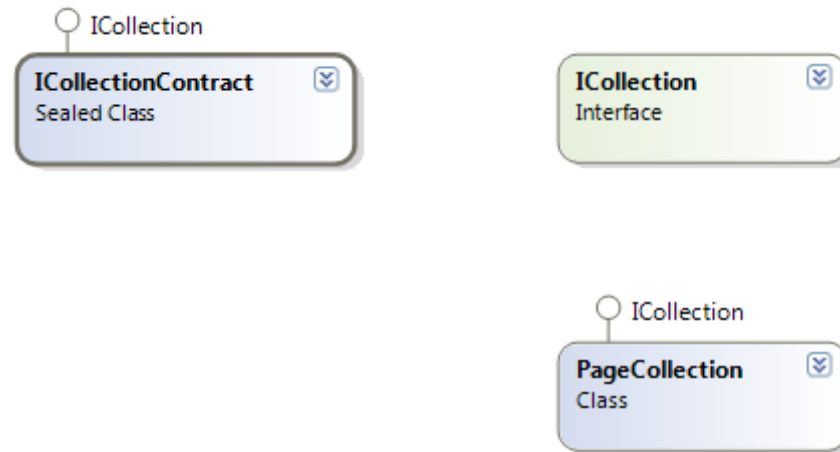
    if (count == data.Length)
    {
        Expand();
    }

    data[count++] = page;
}
```

Invariants

```
[ContractInvariantMethod]
private void ObjectInvariant()
{
    Contract.Invariant(count >= 0);
    Contract.Invariant(
        Contract.ForAll(0, count, index => data[index] != null));
}
```

Contracts on Interfaces



Contracts on Interfaces

```
[ContractClass(typeof(ICollectionContract))]  
public interface ICollection  
{  
    void Add(Page page);  
  
    void Remove(Page page);  
  
    Page this[int index] { get; }  
  
    int Count { get; }  
}
```

Contracts on Interfaces

```
[ContractClassFor(typeof(ICollection))]
public sealed class ICollectionContract : ICollection
{
    void ICollection.Add(Page page)
    {
        ICollection @this = this;
        Contract.Requires<ArgumentNullException>(page != null);
        Contract.Ensures(@this.Count == Contract.OldValue(@this.Count) + 1);
    }

    Page ICollection.this[int index]
    {
        get
        {
            Contract.Requires<IndexOutOfRangeException>(index > -1);
            Contract.Ensures(Contract.Result<Page>() != null);
            return default(Page);
        }
    }
}
```

När körs våra kontrakt?

- Vid kompilering
- Vid DEBUG

Contracts & Unit tests

- Kontrakt kan eliminera triviala tester
- Enhetstester kommer fortfarande behövas för att testa metodernas beteenden

Design by contracts

- Vilka problem löser det?

Tack!

- Microsoft Code Contracts
<http://research.microsoft.com/contracts>
- Kodexempel
<http://www.valtech.se/se/dotnetdays/code-contracts/>
- mikael.lundin@valtech.se
Twitter: @mikaellundin
Blogg: <http://mint.litemedia.se>