Code Contracts

Robert Haken [MVP ASP.NET, MCT]
Software architect, Owner at HAVIT, s.r.o.
knowledge-base.havit.cz

TECHNICAL EDUCATION & DEVELOPER DATABASE CONFERENCE 2010
GOPAS | DAQUAS | Microsoft

Tech·Ed
DevCon
Praha 2010

Contracts

- volaná strana deklaruje, jak se chová
- vzniká dohoda mezi volající a volanou stranou
- pokud volající splní vstupní podmínky volaného (pre-conditions)
- volaný se mu odmění splněním vlastních výstupních podmínek (post-conditions)
- při zachování podmínek konzistence (invariants)

Code Contracts v .NET

- Vyjádření contractů ve zdrojovém kódu
  - pre-conditions – vstupní podmínky metody
  - post-conditions – výstupní podmínky metody
  - object invariants – podmínky konzistence objektu
- získáme
  - runtime-checking
  - testování + statická analýza
  - generování dokumentace

```
public class RacionalniCislo
{
    public int Citatel
    {
        get{ return _citatel; }
    }
    private int _citatel;

    public int Jmenovatel
    {
        get
        {
            Contract.Ensures(Contract.Result<int>() != 0, "Jmenovatel nemůže být nikdy 0.");
            return _jmenovatel;
        }
    }
    private int _jmenovatel;

    public RacionalniCislo(int citatel, int jmenovatel)
    {
        Contract.Requires(jmenovatel != 0,"Nelze vytvořit racionální č. se jmenovatelem 0.");
        _citatel = citatel;
        _jmenovatel = jmenovatel;
    }

    [ContractInvariantMethod]
    void ObjectInvariant()
    {
        Contract.Invariant(_jmenovatel != 0, "Jmenovatel racionálního čisla není nikdy 0.");
    }
}
```

Intro

DEMO

Pre-conditions

- „stav světa“ před voláním metody
- Contract.Requires(condition, [message]);
- Contract.Requires<T>(condition, [message]);
  where T: Exception
- „Legacy Requires“
  if (condition) throw new XyException(…);
  Contract.*() nebo Contract.EndContractBlock();

Post-conditions

- „stav světa“ při skončení metody
- Contract.Ensures(condition, [message]);
- Contract.EnsuresOnThrow<T>(cond, [message]);
  where T: Exception

Pomocné metody do post-conditions

- Contract.Result<T>()
- Contract.OldValue<T>(value)
- Contract.ValueAtReturn(out value)

„Průběžné“ podmínky v kódu

- Contract.Assert(condition, [message]);
- Contract.Assume(condition, [message]);

Object-invariant podmínky

- [ContractInvariantMethod]
  private void ObjectInvariant()
  {
  Contract.Invariant(condition, [message]);
  Contract.Invariant(condition, [message]);
  ...
  }
- kontrolují se při výstupu z každé public metody

Kvantifikátory do podmínek

- Contract.ForAll(start, end, index => condition)
- Contract.ForAll(IEnumerable, item => condition)

- Contract.Exists(start, end, index => condition)
- Contract.Exists(IEnumerable, item => condition)

- také Linq.Enumerable.All(), Linq.Enumerable.Any()

Contracts pro interface/abstract

- [ContractClass(typeof(IFooContract))]
  interface IFoo {
      void DoSomething(int value);
  }
- [ContractClassFor(typeof(IFoo))]
  sealed class IFooContract : IFoo {
      void IFoo.DoSomething(int value) {
          Contract.Requires(value != 0);
      }
  }

Contracty a dědičnost/viditelnost

- dědí se i bez volání base.DoSomething()
- pre-conditions musí být v root metodě
  - nelze zpřísňovat contract potomka
- pre-condition smí používat jen „viditelné“ hodnoty (public, argumenty, ...)
  - volající strana musí být schopna podmínky sama ověřit
- post-conditions a invariants neomezeně
- v podmínkách lze volat jen [Pure] metody

Build, Rewriting
Dialog Code Contracts ve Visual Studiu

DEMO

Runtime Checking

	Legacy	Require<E>	Require	Ensure	Invariant	Assert Assume
Full	X	X	X	X	X	X
Pre and Post	X	X	X	X		
Preconditions	X	X	X			
ReleaseRequires	X	X				
None	!					

- Doporučené nastavení
  - **Debug** = Full + Assert on Contract Failure
  - **Release** = ReleaseRequires + Only Public Surface Contracts

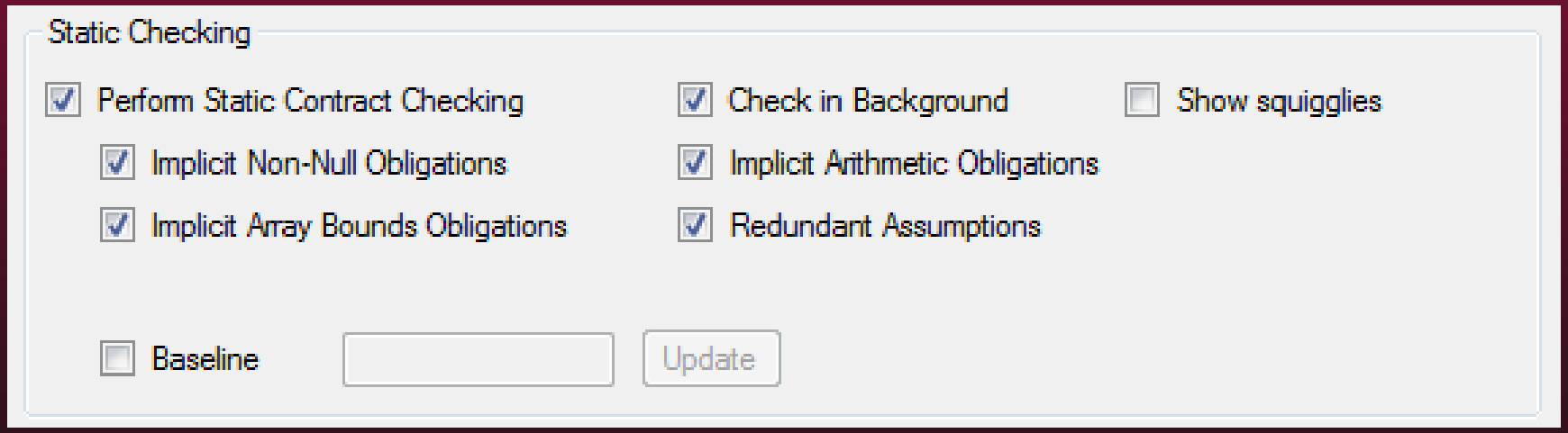
Runtime Checking Extensibility

- Contract.ContractFailed event
- Custom Rewriter Methods
  - Requires/Ensures/Invariant/Assert/Assume
    - ReportFailure
      - RaiseContractFailedEvent
      - TriggerFailure
- Call Site Requires pro design-time kontrolu plnění contractů API třetích stran

Runtime Checking

DEMO

Statická analýza



- Doporučení
  - Baseline na existujících projektech
  - [ContractVerification(true|false)] – opt in/out
  - Background checking (výkonově náročné!)

Statická analýza

DEMO

Generování dokumentace contractů

- doplní existující XML soubor generovaný compilerem C#/VB
- šablona pro Sandcastle

Generování dokumentace

DEMO

Použití ve webových projektech

- ERROR: Ambiguous class...
- MyAssembly.dll + MyAssembly.Contracts.dll
- řešení – odebrat assembly ve web.config

```
<system.web>
        <compilation>
                <assemblies>
                        <remove assembly="MyAssembly.Contracts"/>
                </assemblies>
        </compilation>
</system.web>
```

TECHNICAL EDUCATION & DEVELOPER DATABASE CONFERENCE 2010

GOPAS | DAQUAS | Microsoft

Q & A

Tech·Ed DevCon Praha 2010


GOPAS


DAQUAS


Microsoft