41. Контракт и наследяване. Проблем с ограничаване на областта в дъщерен обект.

Когато създаден контракт, реализиран с private метод - pure (непроменящ се), се

създава се метод, чиито наследник може да дефинира за да опище своите собствени инварианти. Той от своя страна вика public, abstract метод, чийто наследник може да дефинира за да опише своите собствени инварианти.

public abstract class DomainObject

{ public abstract Boolean IsValid();

[Pure]

private Boolean IsValidState()

{

return IsValid();

}

[ContractInvariantMethod]

private void ObjectInvariant()

{

Contract.Invariant(IsValidState());

}

**public class** Customer : DomainObject

{

**private** Int32 Id;

**private** String CompanyName, Contact;

**public** Customer(Int32 id, String company)

{

Contract.Requires(id > 0);

Contract.Requires(company.Length > 5); Contract.Requires(!String.IsNullOrWhiteSpace(company));

Id = id;

CompanyName = company;

}

...

**public override bool** IsValid()

{

**return** (Id > 0 && !String.IsNullOrWhiteSpace(CompanyName));

}

}

Има bug: вика се конструкторът на ‘DomainObject’, т.е инвариантът се проверява. IsValid() е виртуален, т.е. изпълнява се реализацията в ‘Custоmer’. Проверката, обаче, става преди инстанцията да е напълно инициализирана. Вдига се exception .

Добре е да спазате общо правило: не викайте виртуални методи в тялото на конструктор.