design by contract

sabre .net community

dariuszwozniak.net

kraków

sep 9
2014

agenda defensive programming design by contract

- What is: **Defensive Programming**
- What is: **Design by Contract** (DBC) and **Code Contracts**
- **Benefits** of DBC
- **History** of DBC
- **Code Contracts in C#**
  - Examples
- **Live Demo**
- **Summary**
- **Q&A**

agenda defensive programming design by contract

Syntax Correctness

- Verified by a compiler

Semantic Correctness

- Verified in a runtime
- Major cause of bugs
- Examples:
  - *Count()* >= 0
  - *age* must be in range [0; 122]
  - *Obj* cannot be Null

agenda defensive programming design by contract

GetRoom(Hotel hotel);

Problem:
How to check whether it is NULL or not?

- `if (hotel == null) throw new ArgumentNullException(”hotel”);`
- `Debug.Assert(hotel != null);`
- `Trace.Assert(hotel != null);`

- Configurable (DEBUG\RELEASE\etc.)
- Compile check

- `Contract.Requires<ArgumentNullException>(hotel != null, ”hotel”);`

agenda defensive programming design by contract

- design by contract is a **software correctness methodology**
- it uses preconditions and postconditions to **document** (or programmatically assert) **the change in state** caused by a piece of a program

source: http://c2.com/cgi/wiki?DesignByContract

agenda defensive programming design by contract

static (compile-time) and/or runtime checking

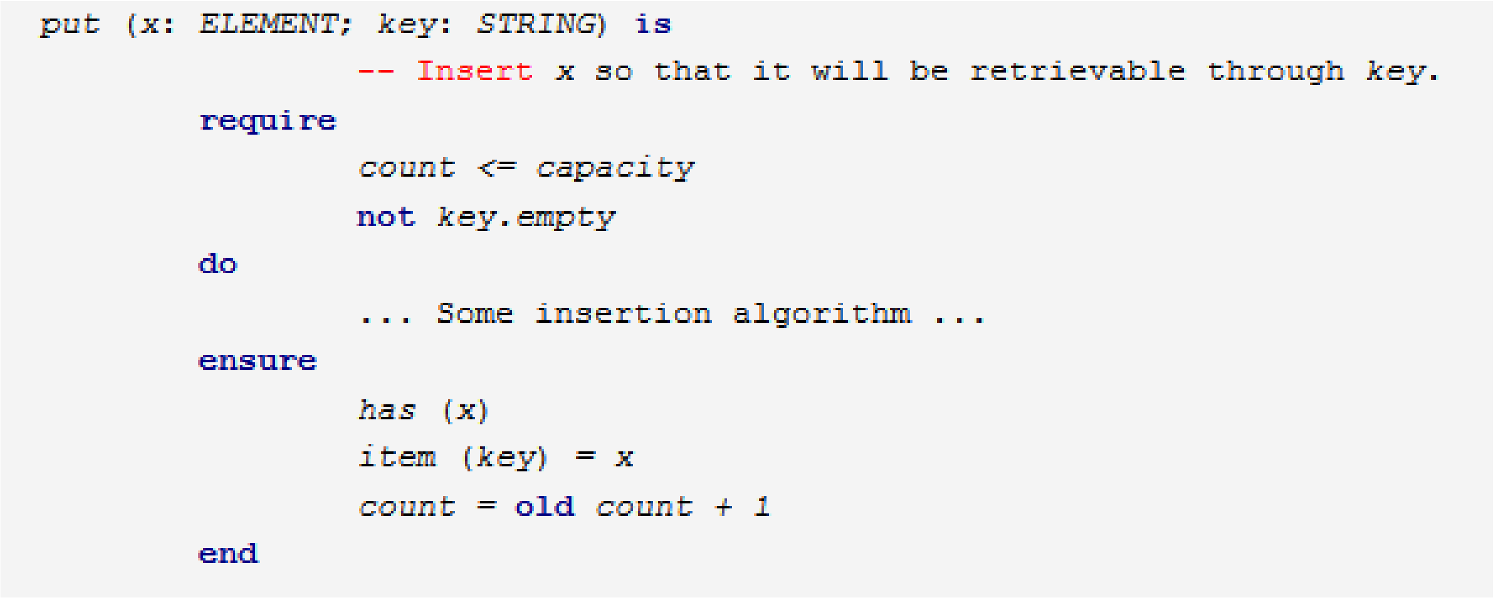
- precondition
  - condition checked on entry to method
- postcondition
  - condition checked on exit of method
- object invariant
  - condition that always should be true

benefits history examples summary references

- static verification
- automatic testing tools
- code documentation
  - contracts as documentation
  - contracts added to documentation
- cleaner code
- improved feedback loop
- short learning curve

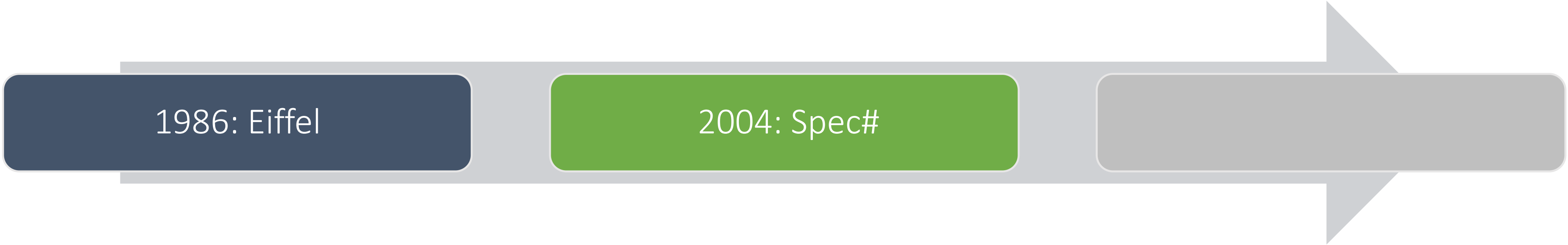
benefits history examples summary references

Eiffel Software



```
put (x: ELEMENT; key: STRING) is
                -- Insert x so that it will be retrievable through key.
        require
                count <= capacity
                not key.empty
        do
                ... Some insertion algorithm ...
        ensure
                has (x)
                item (key) = x
                count = old count + 1
        end
```

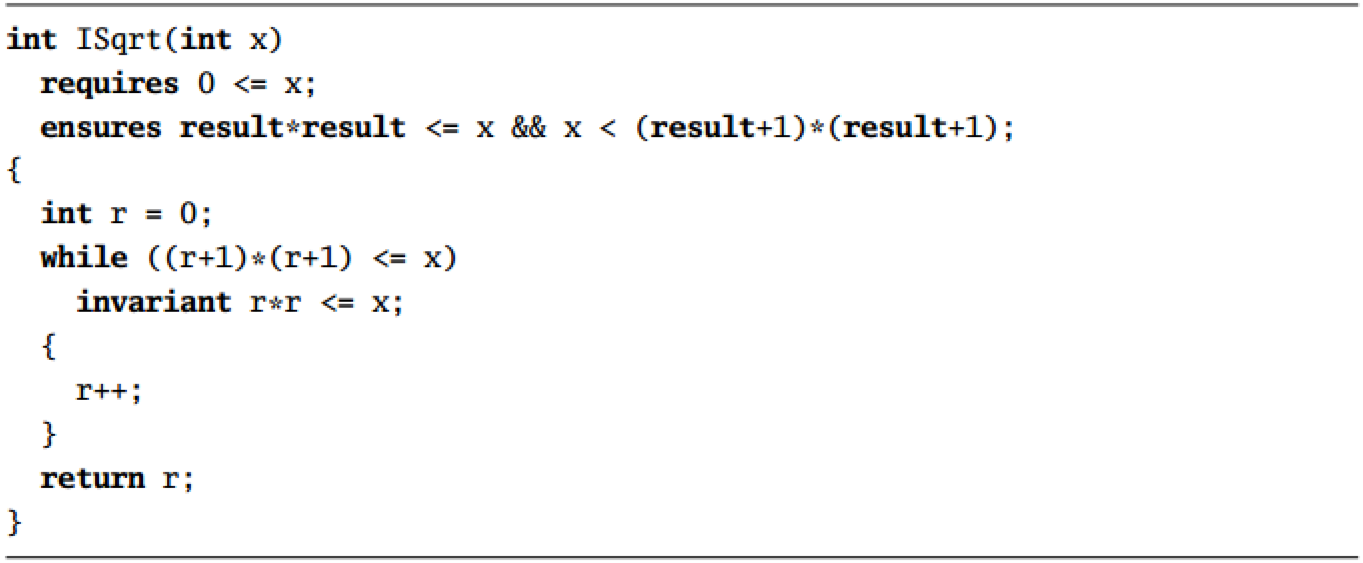
benefits history examples summary references



1986: Eiffel

2004: Spec#

Microsoft® Research



```
int ISqrt(int x)
  requires 0 <= x;
  ensures result*result <= x && x < (result+1)*(result+1);
{
  int r = 0;
  while ((r+1)*(r+1) <= x)
    invariant r*r <= x;
  {
    r++;
  }
  return r;
}
```

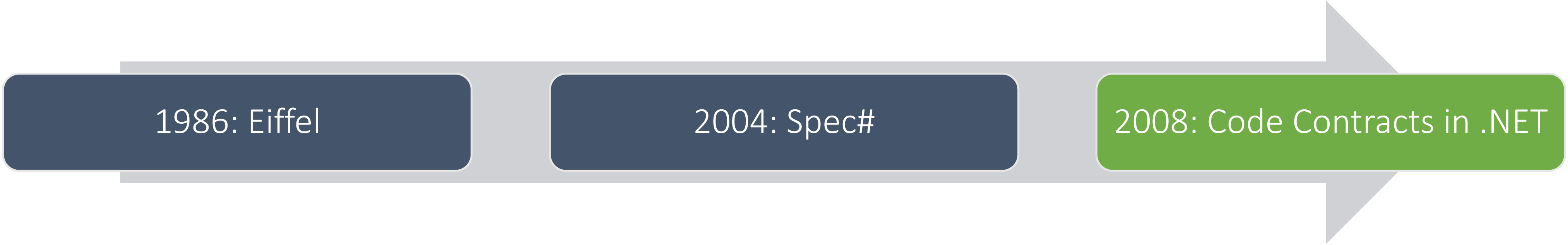
benefits history examples summary references

1986: Eiffel

2004: Spec#

2008: Code Contracts in .NET

benefits **history** examples summary references



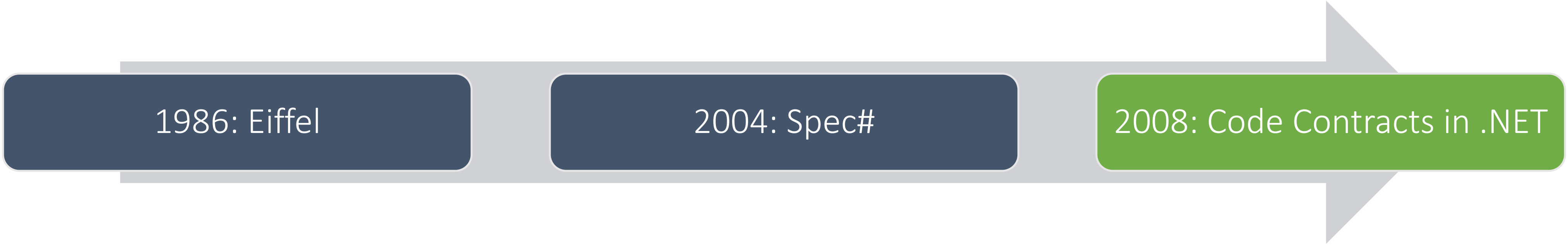
1986: Eiffel

2004: Spec#

2008: Code Contracts in .NET

- **part of the library** since .NET 4.0
- **static and runtime checking** (configurable per project)
- **inheritable** contracts
  - support for abstract classes and interfaces

benefits **history** examples summary references



1986: Eiffel

2004: Spec#

2008: Code Contracts in .NET

- generate **API documentation**
  - hooks into XML documentation and inserts contract requirements (requires, ensures)
- automatically suggests **missing contracts**
- **resharper** support

examples

preconditions

```
public int Add(int a, int b)
{
        Contract.Requires<ArgumentOutOfRangeException>(a >= 0);
        Contract.Requires<ArgumentOutOfRangeException>(b >= 0);
        // main logic
}
```

postconditions

```
public int Add(int a, int b)
{
        // pre-conditions
        Contract.Ensures(Contract.Result<int>() >= 0);
        // main logic
}
```

object invariants

```
[ContractInvariantMethod]
private void CheckIfLastResultIsInRange()
{
        Contract.Invariant(lastResult >= 0);
}
```

// demo

benefits history examples summary references

- defensive programming
- software correctness
- static and runtime checking of
  - preconditions
  - postconditions
  - object invariants
- documents and asserts changes in a state of a program

benefits history examples summary references

- MSDN: Code Contracts http://msdn.microsoft.com/en-us/library/dd264808%28v=vs.110%29.aspx
- Using the Spec# Language, Methodology, and Tools to Write Bug-Free Programs [2009]
- Mike Frederick: Code Contracts in .NET 4 — SVNUG Presentation [December 2011]
- Code Contracts is the next coding practice you should learn and use http://codebetter.com/patricksmacchia/2013/12/18/code-contracts-is-the-next-coding-practice-you-should-learn-and-use/
- Clarence Bakirtzidis: Code Contracts API In .NET
- http://c2.com/cgi/wiki?DesignByContract
- Jon Skeet: C# in Depth (2nd ed.)

questions

thank you

github.com/dariusz-wozniak/dbc-demo

dariusz.wozniak@sabre.com

dariuszwozniak.net