Telerik Academy

Tools for Developers

Integrated Development Environments, Source Control Repositories, Automated Testing Tools, Bug Tracking, Code Analysis Tools, Build Tools, Project Hosting Sites, Deployment in the Cloud

Telerik Software Academy

Learning & Development

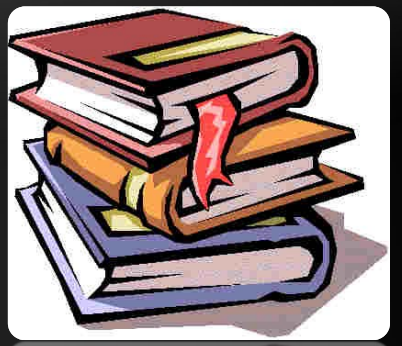
http://academy.telerik.com

Telerik Academy

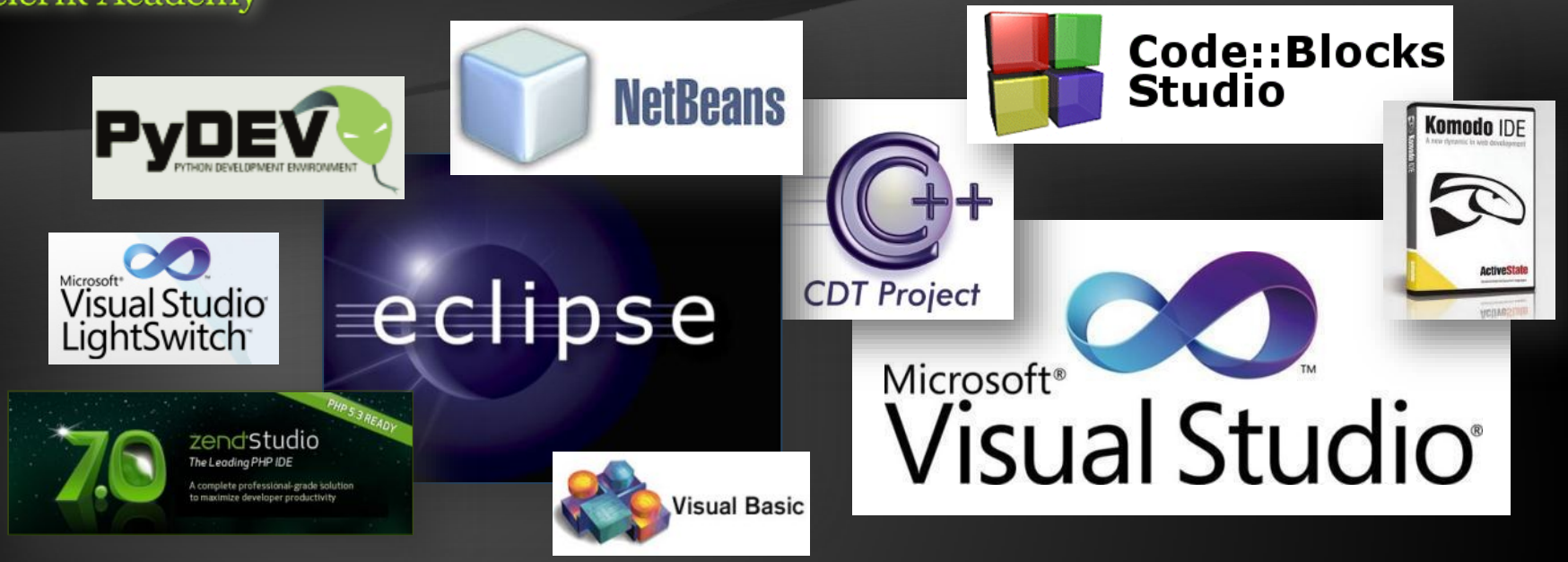
Table of Contents

- Integrated Development Environments (IDEs)
- Source Control Systems
- Code Generation Tools
- Logging Tools
- Unit Testing Tools
- Bug Tracking / Issue Tracking Systems
- Code Analysis Tools
- Code Decompilation Tools

Table of Contents (2)



Telerik Academy



Integrated Development Environments (IDEs)

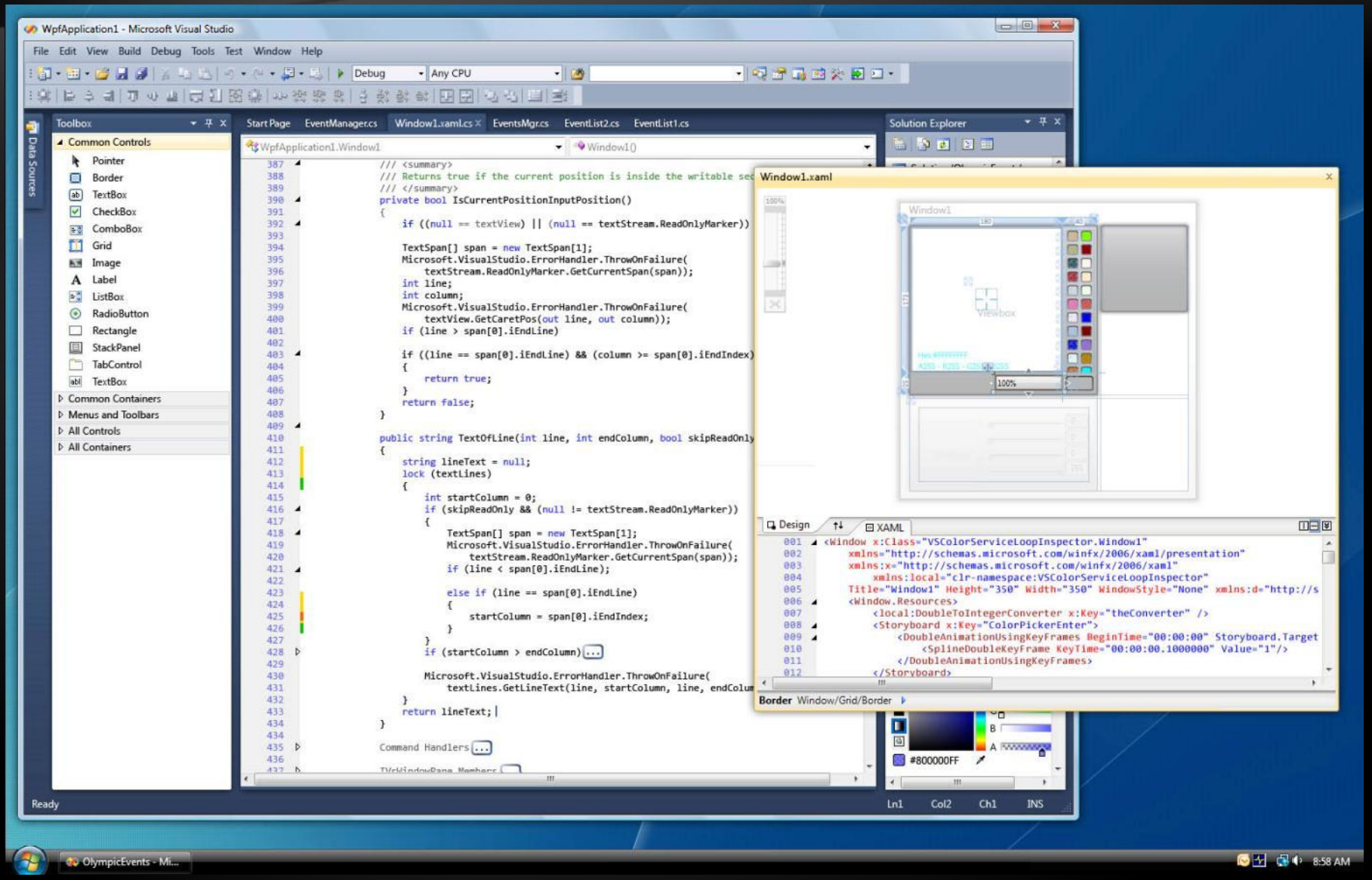
Visual Studio, Eclipse, IntelliJ IDEA, Netbeans, JDeveloper, Code::Blocks, Bloodshed Dev-C++

Visual Studio

- Visual Studio is official .NET development tool from Microsoft
  - Multiple languages: C#, VB.NET, C++, …
  - Multiple technologies and platforms: ASP.NET, WPF, Silverlight, WWF, WCF, Windows Mobile
  - Very powerful and feature rich
  - Write, compile, model, design GUI, forms, data, build, execute, debug, test, deploy, refactor, …
  - Commercial product, has free editions

Telerik Academy

Visual Studio – Screenshot



Visual Studio

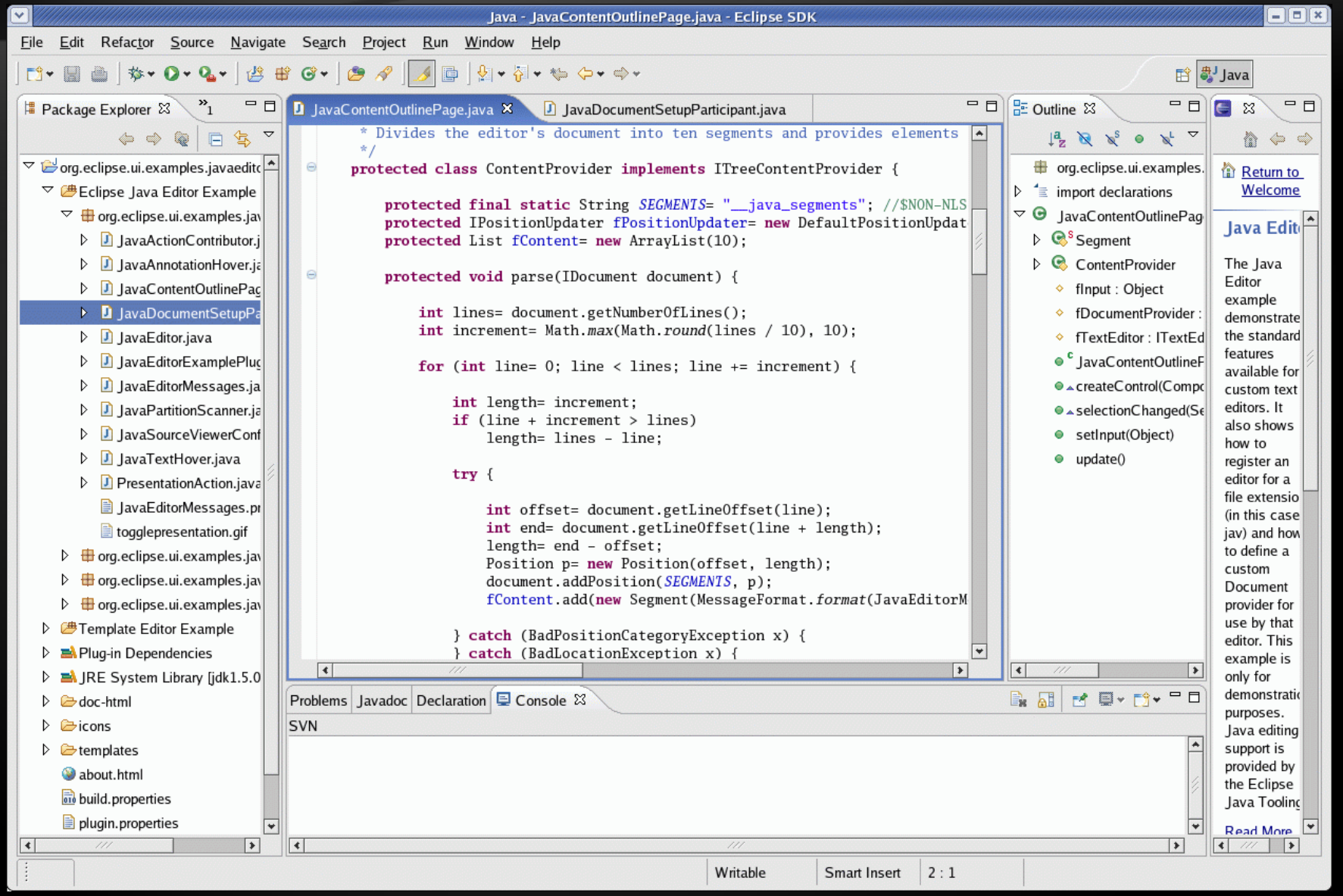
Live Demo

Eclipse

- Eclipse is the #1 Java IDE
  - Supports multiple languages and platforms: Java, Java EE, C++, PHP, Python, Ruby, mobile development, embedded development, …
  - Very powerful and feature rich
  - Write, compile, build, execute, debug, test, deploy, refactor, …
  - Thousands of plug-ins
  - Free, open-source product, very big community

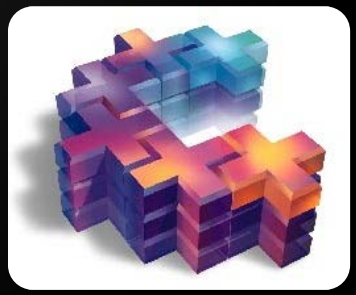
Telerik Academy

Eclipse – Screenshot



Other IDEs

- Other Java IDEs
  - Netbeans
    - Supports latest Java technologies
    - Nice GUI designer for Swing and JSF
  - IntelliJ IDEA – very powerful refactoring
  - JDeveloper – feature rich, supports UML
- Other C++ IDEs
  - Code::Blocks
  - Bloodshed Dev-C++



Telerik Academy

Source Control Systems

Subversion (SVN), Team Foundation Server (TFS), CVS, Git, Mercurial, Perforce, …

Telerik Academy

What is Source Control System?

- Source control systems (version control systems, source control repositories)
  - Hold the source code and project assets during the development process
  - Allow simultaneous changes in the source code and conflict resolution
  - Keep version history of the project assets
- Two versioning models:
  - Lock-Modify-Unlock and Copy-Modify-Merge

Lock-Modify-Unlock Model

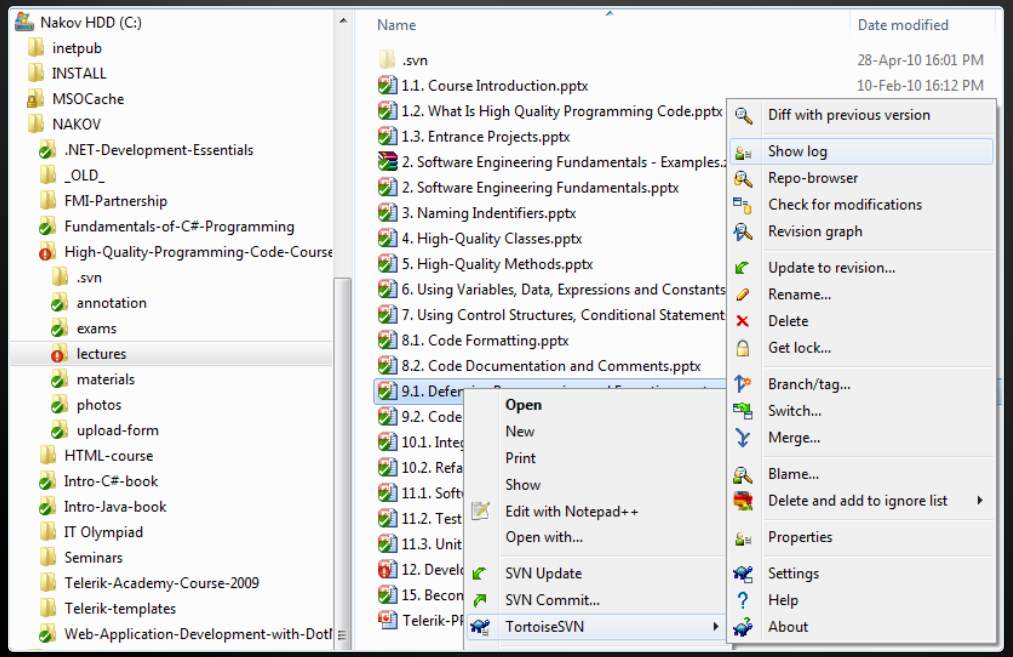
- The lock-modify-unlock model needs locking files before modification
  - One file is modified by at most one person in any given moment
  - No conflicts, no need of merge
  - Suitable for small teams
    - When changes almost don't need concurrency
  - Basic commands: check out, check-in
  - Implemented in: Visual SourceSafe, (TFS, SVN)

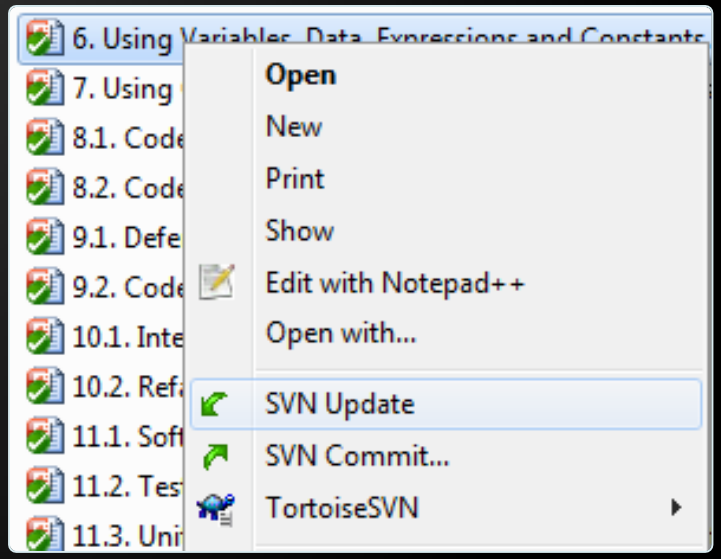
Copy-Modify-Merge Model

- Copy-modify-merge model does not hold locks during the source code modification
  - The same file could be simultaneously edited by multiple developers
  - Sometimes requires conflict resolution
  - Suitable for large teams and projects
    - High concurrency of source code modifications
  - Basic commands: update, commit
  - Implemented in: SVN, CVS, Git, Mercurial, TFS

Subversion (SVN)

- Subversion (SVN)
  - Popular and well established system
  - Free, open-source, very large community
  - TortoiseSVN – the most popular client





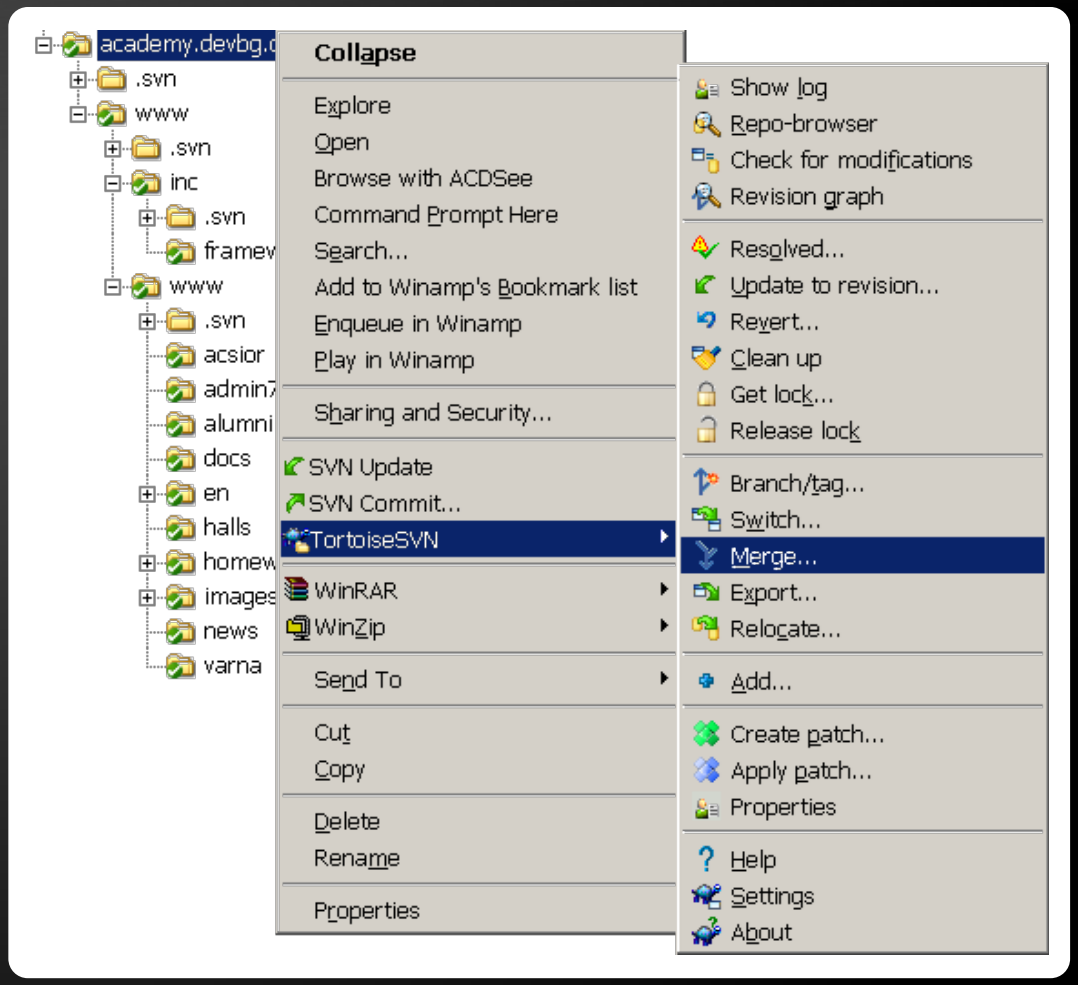
Telerik Academy

Subversion (SVN) (2)

- Subversion (SVN)
  - Official web site:
    - http://subversion.tigris.org/
  - Runs on UNIX, Linux, Windows
- Console client
  - svn
- GUI client
  - TortoiseSVN – http://tortoisesvn.tigris.org/

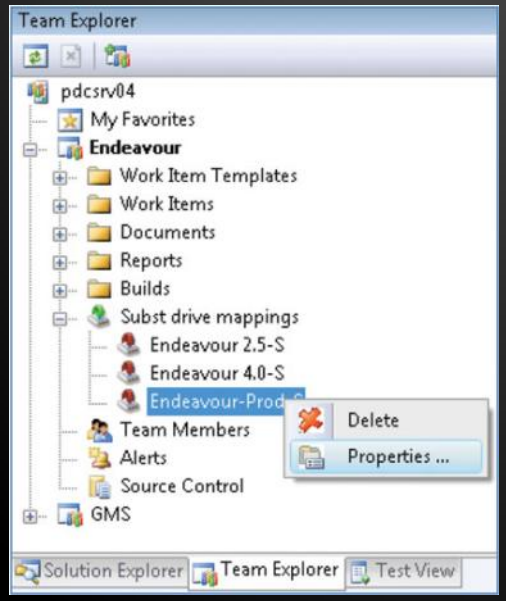
TortoiseSVN

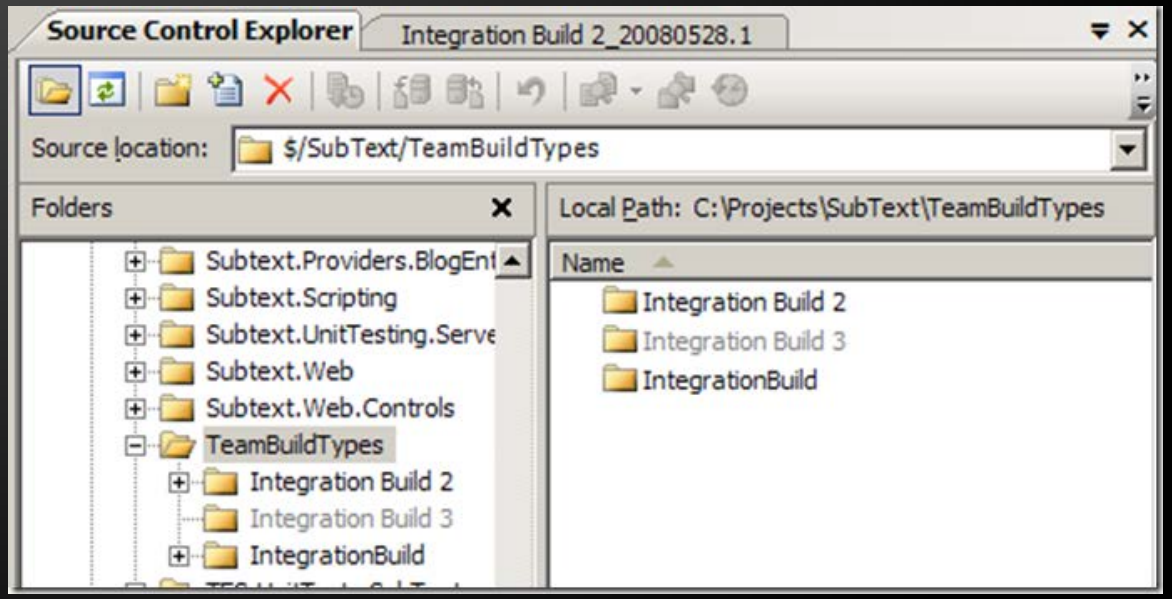
- TortoiseSVN
  - Open source GUI client for Subversion
  - Integrated in Windows Explorer
  - http://tortoisesvn.tigris.org/



Team Foundation Server

- Microsoft Team Foundation Server (TFS)
  - Works best with Visual Studio
    - Many problems outside of it
  - Commercial license





Telerik Academy

Other Source Control Systems

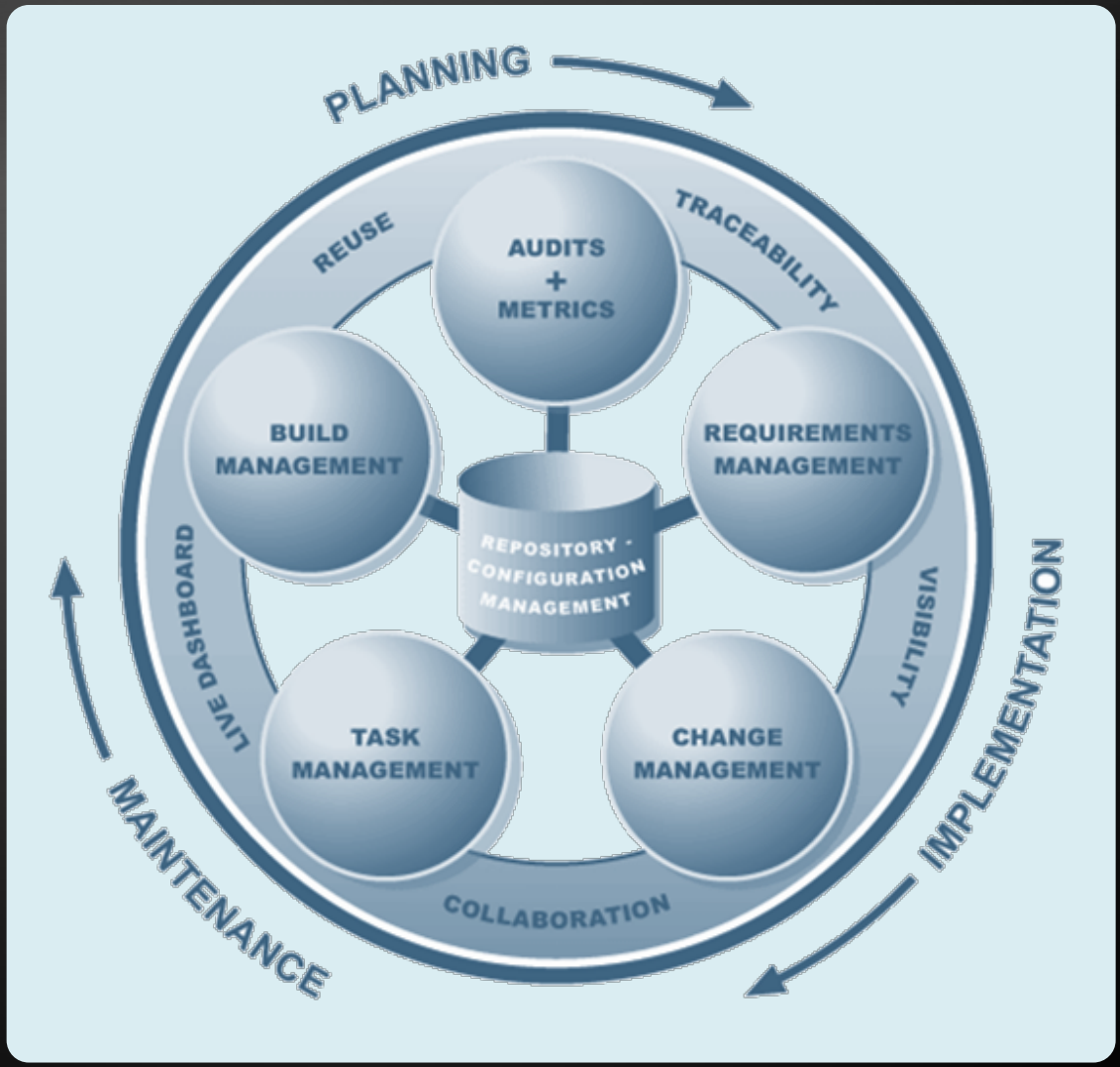
- CVS
  - Was extremely popular in the past
    - Now it is obsolete
  - Open source, mostly used in UNIX / Linux
- Git and Mercurial
  - Fast, distributed, open source
- Perforce
  - Very powerful and scalable (pentabytes of code)
  - Commercial product (used by SAP)

SCM and ALM Systems

- Software Configuration Management (SCM systems (e.g. Rational ClearCase, StarTeam)
  - Change management for requirements, documents, source code, etc.
  - Tools, policies, workflow, etc.
- Application Lifecycle Management (ALM) systems (e.g. VSTS + TFS, StarTeam, Polarion)
  - Covers the entire development process
  - Requirements, planning, project management, architecture, build, QA, test, integration, etc.

Telerik Academy

Application Lifecycle Management



Telerik Academy

Code Generation Tools

Visual Studio T4 Engine, CodeSmith, AndroMDA

Telerik Academy

Code Generators – Concepts

- Source code generation
  - Template based
    - Build data access layer by given database schema
      - E.g. Visual Studio Data Designer for LINQ-to-SQL
    - Build Web application by database schema
      - E.g. Django (Python based Web application platform)
  - Model based
    - Build entire application by high-level model

Code Generation Tools

- Microsoft T4 Engine, CodeSmith
  - Template based code generators
  - Can generate classes, Web pages and other project assets by set of templates
    - E.g. data access layer based on database schema
- AndroMDA
  - Application generator framework
  - Transforms UML models to Java EE applications based on Spring, Hibernate and JSF

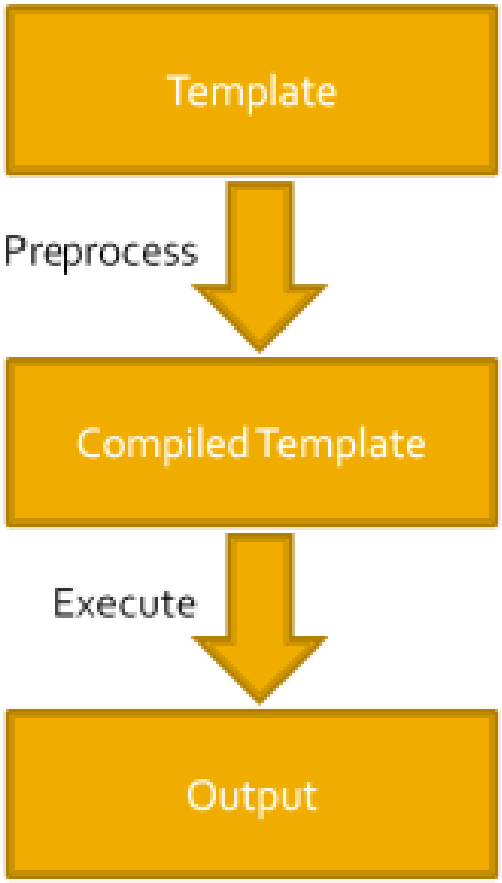
Telerik Academy

Microsoft T4 Template Engine

- Visual Studio T4 Engine
  - T4 == Text Template Transformation Toolkit
  - Integral part of Visual Studio
  - ASP.NET-like syntax
- T4 templates consist of:
  - Processing directives (e.g. include template)
  - Text blocks (static text)
  - Code blocks (in C#, VB.NET)
  - Compiled to C# and then to .NET assemblies

Telerik Academy

T4 Template Engine – Architecture




```
<#@ template language="C#" #>
Hello
<# Write("World!"); #>
```

```
public class GeneratedTextTransformation:
  Microsoft.VisualStudio.TextTemplating.TextTransformation
{
    public override string TransformText()
    {
        this.Write("Hello\r\n");
        Write("World!");
        return this.GenerationEnvironment.ToString();
    }
}
```

```
Hello
World!
```

Visual Studio T4 Example

```
<#@ template debug="false" hostspecific="false"
language="C#" #>
<#@ assembly name="System.Core" #>
<#@ import namespace="System.Text" #>
<#@ import namespace="System.Collections.Generic" #>
<#@ output extension=".cs" #>
<# for(int i = 0; i < 3; i++) { #>
        public int Item<#=i#> { get; set; }
<# } #>
```

```
public int Item0 { get; set; }
public int Item1 { get; set; }
public int Item2 { get; set; }
```

Telerik Academy

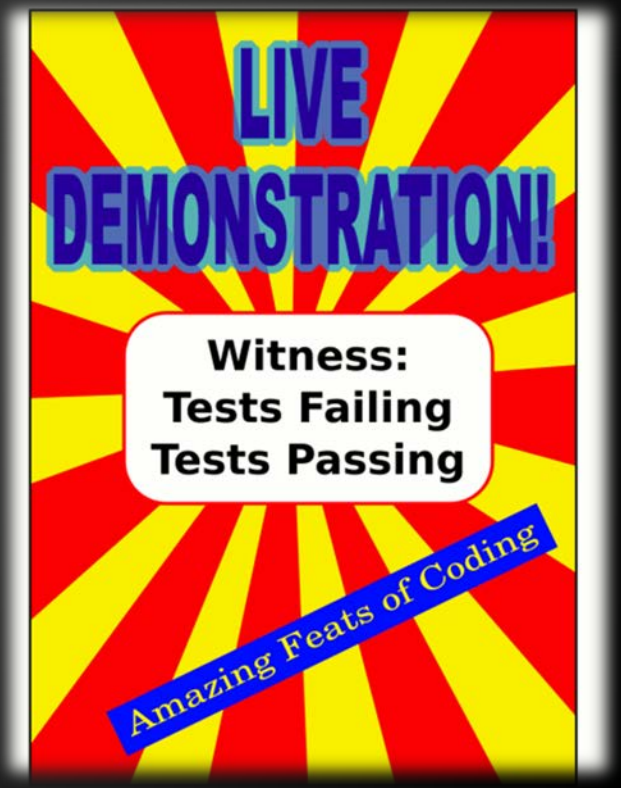
Visual Studio T4 Example (2)

```
<#@ template language="C#" hostspecific="true" #>
<#@ output extension=".xml" #>
<#@ import namespace="System.IO" #>
<#@ import
namespace="Microsoft.VisualStudio.TextTemplating" #>
<?xml version="1.0" ?>
<project>
<#  string currentDir = this.Host.ResolvePath(".");
    foreach (string dir in Directory.GetFiles(currentDir))
    {
        FileInfo fileInfo = new FileInfo(dir);
#>
      <file><#= fileInfo.Name #></file>
<#  } #>
</project>
```

Telerik Academy

Visual Studio T4 Template Engine

Live Demo



Telerik Academy

Logging Tools

Log4J, Log4Net

Logging

- Logging is chronological and systematic record of data processing events in a program
  - E.g. the Windows Event Log
- Logs can be saved to a persistent medium to be studied at a later time
- Use logging in the development phase:
  - Logging can help you debug the code
- Use logging in the production environment:
  - Helps you troubleshoot problems

Log4J / Log4Net

- Log4J / Log4Net are a popular logging frameworks for Java / .NET
  - Designed to be reliable, fast and extensible
  - Simple to understand and to use API
  - Allows the developer to control which log statements are output with arbitrary granularity
  - Fully configurable at runtime using external configuration files

Log4j / Log4Net Architecture

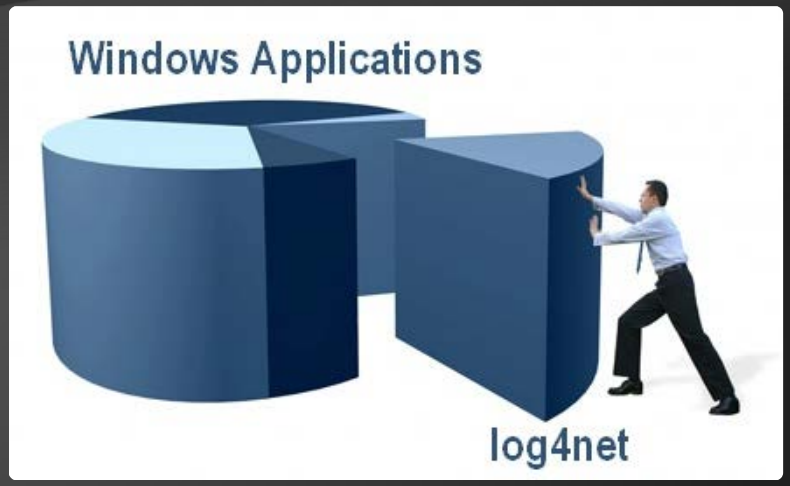
- Log4Net has three main components: loggers, appenders and layouts
  - Loggers
    - Channels for printing logging information
  - Appenders
    - Output destinations (e.g. XML file, database, …)
  - Layouts
    - Formats that appenders use to write their output

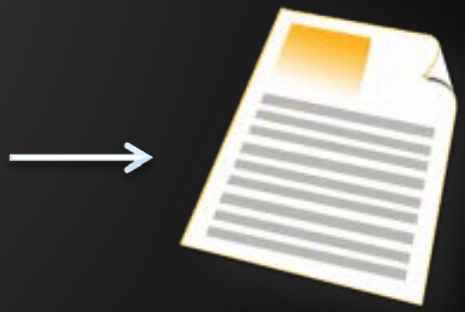
Hello Log4Net – Example

```
class Log4NetExample
{
  private static readonly ILog log =
    LogManager.GetLogger(typeof(Log4NetExample));
  static void Main()
  {
    BasicConfigurator.Configure();
    log.Debug("Debug msg");
    log.Error("Error msg");
  }
}
```

- Output from Log4NetExample

```
2010-12-16 23:25:08 DEBUG Log4NetExample – Debug msg
2010-12-16 23:25:08 ERROR Log4NetExample – Error msg
…
```

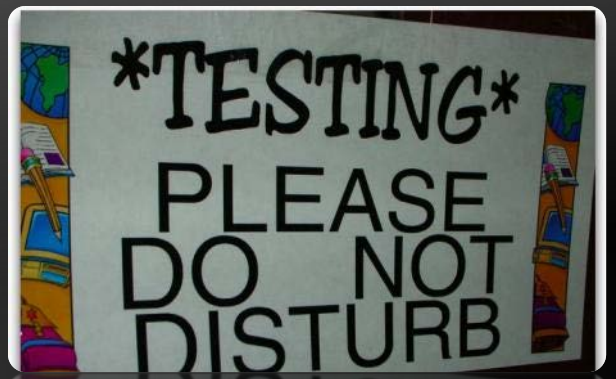




Log4Net

Live Demo

Telerik Academy



Unit Testing Tools

JUnit, NUnit, CppUnit, TestNG, JsUnit, …

Unit Testing

- Unit test is a program that tests pieces of code
  - Test a single method, class or component
  - Implement a common use case scenario
  - Confirm that the code works as expected
    - Or signal that the code is broken
- Unit tests should have high code coverage, e.g. 70-80%
  - Executed in the continuous integration process
- Unit tests dramatically decrease the number of defects in the code

Unit Test – Example

```
int sum(int[] array) {
  int sum = 0;
  for (int i=0; i<array.length; i++)
    sum += array[i];
  return sum;
}

void testSum() {
 if (sum(new int[]{1,2}) != 3)
    throw new TestFailedException("1+2 != 3");
 if (sum(new int[]{-2}) != -2)
    throw new TestFailedException("-2 != -2");
 if (sum(new int[]{}) != 0)
    throw new TestFailedException("0 != 0");
}
```

Telerik Academy

Unit Testing Frameworks / Tools

- Unit testing frameworks
  - Simplify design, implementation and execution of unit tests
- Popular unit testing frameworks
  - JUnit, TestNG – classical unit testing frameworks for Java
  - Visual Studio Team Test (VSTT), NUnit, MbUnit – for .NET development
  - cppUnit, UnitTest++ – for C++ developers
  - jsUnit – for JavaScript

Telerik Academy

Code Coverage & Mocking Tools

- Code coverage tools
  - Code coverage tools check what portion of the source code is covered by the unit tests
  - CodeCover for Eclipse – for Java
  - Visual Studio Team Suite – for C#
  - CoverageMeter – for C++
- Mocking tools
  - Allow testing functionality that is still not implemented9, e.g. through its interface
  - Moq, TypeMock, Rhino Mock, JustMock

Test Automation

- Test automation
  - Replaces manual tests (performed by people) with automated tests (performed by script)
  - Automatically run test scenarios and compare the actual outcomes to predicted outcomes
- Automated testing tools
  - Record and replay test scenarios
- Automated testing frameworks
  - Allow programmatically simulate user activity

Telerik Academy

Bug Tracking / Issue Tracking Systems

TRAC, Bugzilla, JIRA, TFS, SourceForge, Google Code, CodePlex, Project Locker

Bug Tracking Systems

- Bug tracking / issue tracking systems
  - Track bugs / issues related to software development, called tickets
- Tickets consist of:
  - Category: bug / feature request / task
  - State: new → assigned → fixed → closed
  - Priority: critical / high / low / etc.
  - Owner / responsible person
  - Summary, description, attachments

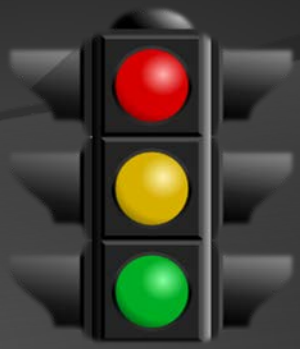
Telerik Academy

Code Analysis Tools

FxCop, StyleCop, Checkstyle, devAdvantage, FindBugs, BoundsChecker, …

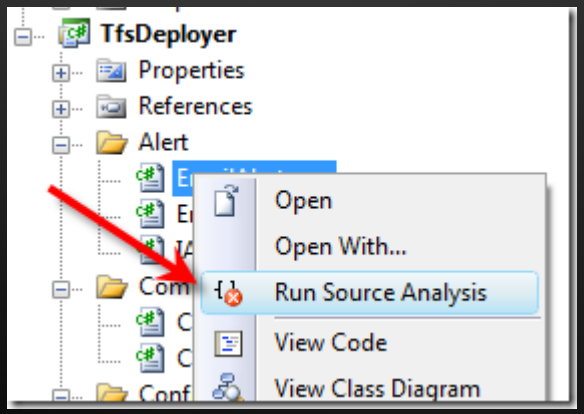
Code Analysis Tools

- Code analysis tools
  - Analyze the source code for bad coding style / unwanted coding practices
- Static analysis
  - Examine the source code at compile-time
  - Could work with the source code or with the compiled assemblies / JAR archives
- Dynamic analysis
  - Analyses the code at runtime (usually done by code instrumentation)



StyleCop

Live Demo



Telerik Academy

Code Decompilation Tools

JustDecompile, IL Spy

Code Decomplation

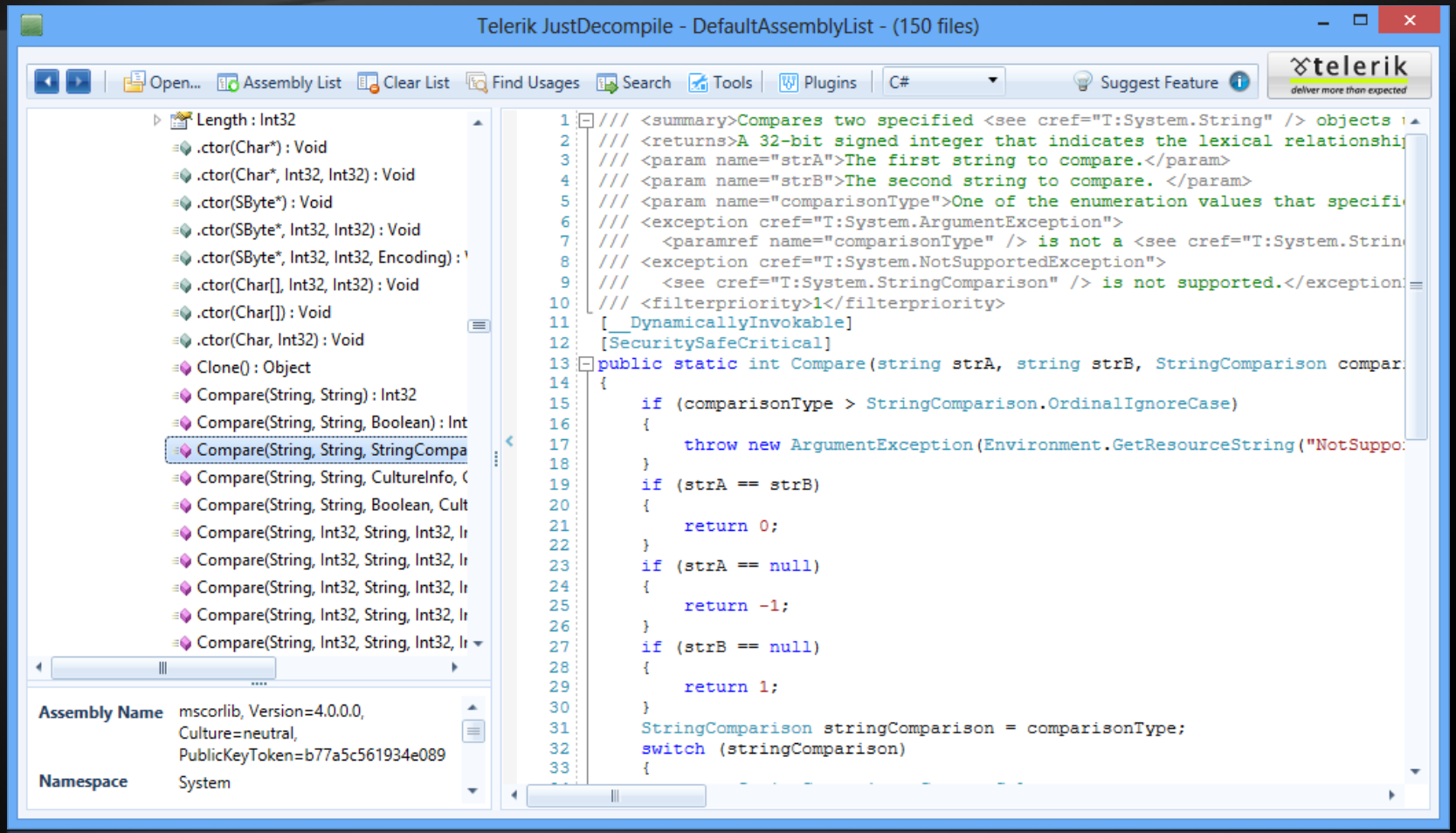
- Code decompilator / code disassembler
  - Reconstructs the source code (to some extent) from the compiled code
  - .NET assembly → C# / VB.NET source code
  - JAR archive / `.class` file → Java source code
  - .EXE file → C / C++ / Assembler code
- Reconstructed code
  - Is not always 100% compilable
  - Loses private identifier names and comments

Code Decomplation Tools

- Code decompilers
  - .NET
    - JustDecompile – free, powerful .NET decompiler
    - ILSpy – powerful, great usability, free
    - ILDASM – part of .NET SDK, decompiles to IL code
  - Java
    - DJ Java Decompiler
    - JD (JD-Core / JD-GUI / JD-Eclipse)
  - .EXE file
    - Boomerang Decompiler → outputs C source code
    - IDA Pro – powerful disassembler / debugger
    - OllyDbg, W32DASM, etc.

Telerik Academy

JustDecompiler



Telerik Academy

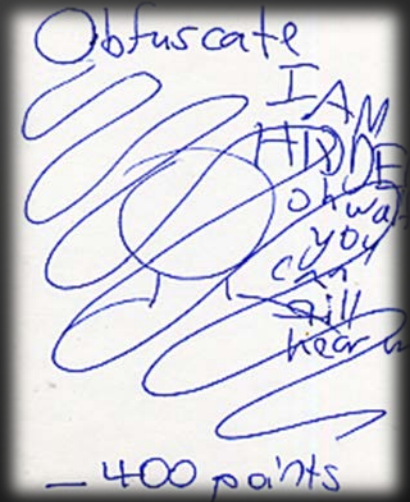
JustDecompile & ILSpy

Live Demo

Telerik Academy

Code Obfuscators

Making Reverse Engineering Difficult



Code Obfuscation

- Code obfuscation
  - Transform the source code or compiled .NET / Java code into a difficult to understand form
  - Obfuscated code has the same behavior
  - Sometimes is a bit slower due to changes and additions in the control logic
  - Obfuscated code is the opposite of the high-quality code
  - Obfuscation is a form of security through obscurity

Code Obfuscation Techniques

- Rewrite `for`-loops as `while`-loops followed by a series of cascading `if-else` statements
- Change iteration into recursion
- Obfuscate programming constructs (e.g. turn if-else statements into ?: operators)
- Introduce meaningless identifier names
- Remove intermediate variables and literals by repeating them as expressions in the code
- Remove literals (e.g. 0 and 1) – use expressions
- Randomize code formatting

Telerik Academy

Simple Obfuscation – Example

- Original source code in C#

```
long first = 1
long second = 1;
for (int i = 3; i <= N; i++)
{
  result = first + second;
  first = second;
  second = result;
}
return result;
```

- Obfuscated and decompiled

```
long _ = 1L;
long __ = 1L;
for (int ___ = 3; ___ <= _____;
___++)
{
    ____ = _ + __;
    _ = __;
    __ = ____;
}
return ____;
```

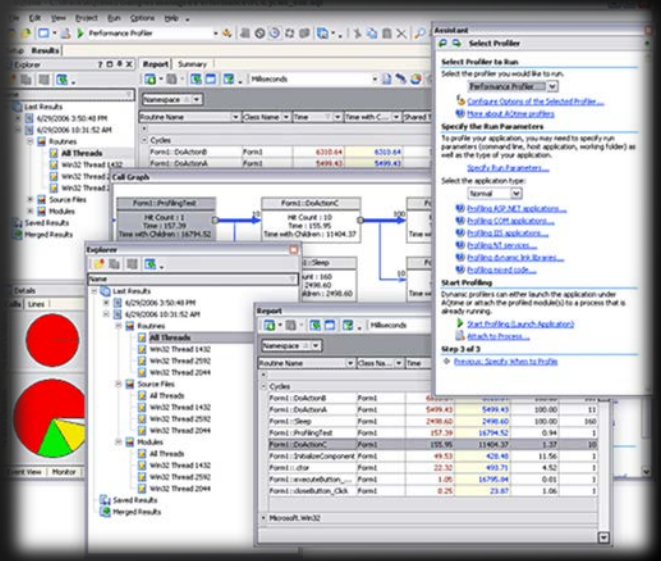
Obfuscation Tools

- .NET obfuscators
  - Eazfuscator.NET – free
  - {smartassembly} – commercial license, very powerful – assembly obfuscation + compression
- Java obfuscators
  - ProGuard – free, open-source
  - yGuard – free, open source
- C++ obfuscators
  - Stunnix C++ Obfuscator – commercial product

Telerik Academy

Code Profilers

ANTS Profiler for .NET, YourKit Profiler for .NET, Netbeans Profiler for Java, JProfiler, JProbe

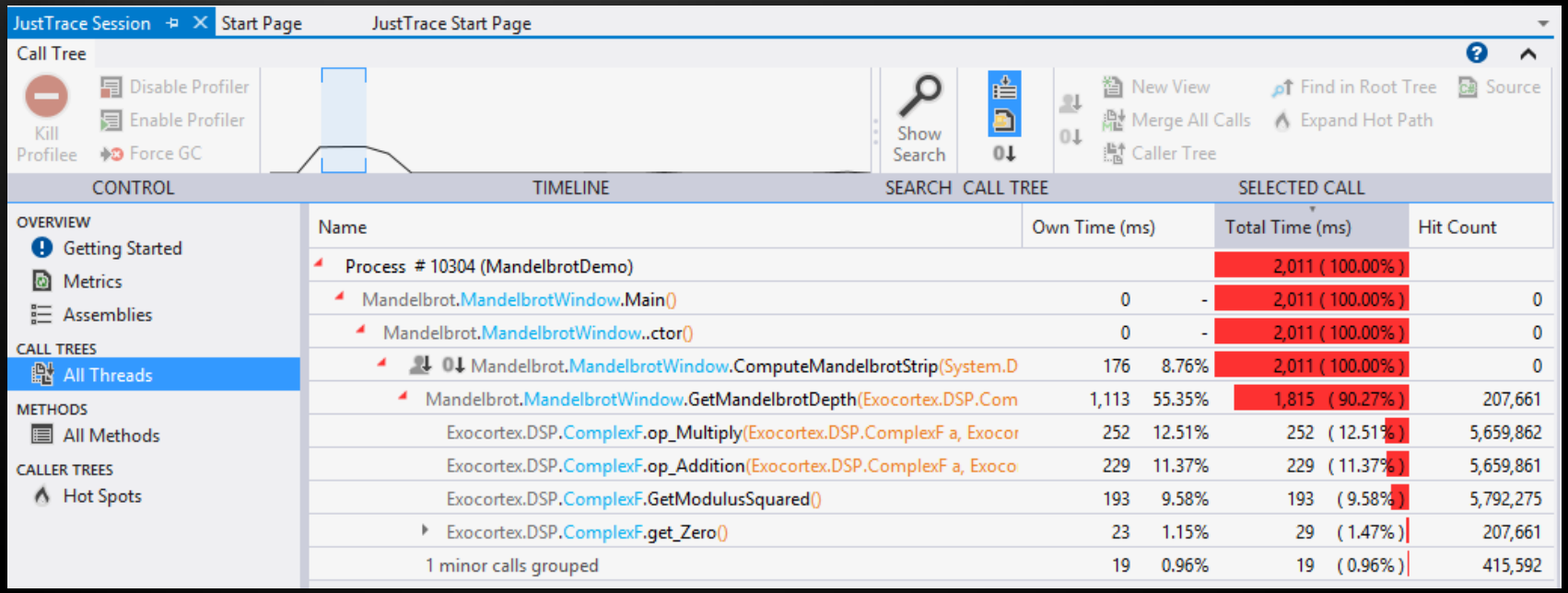


Profilers

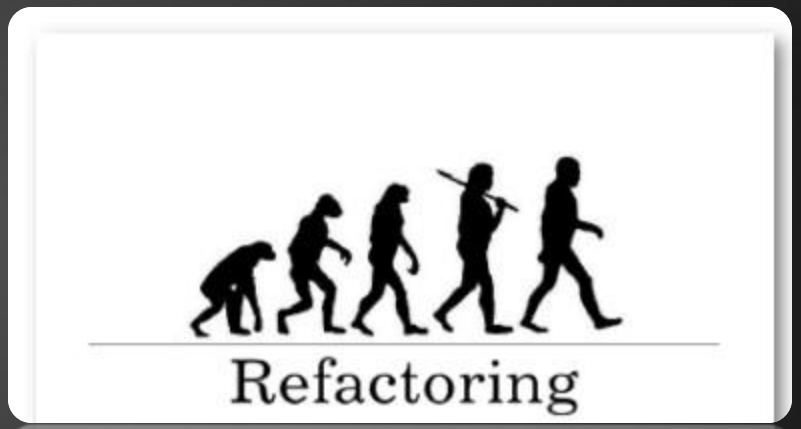
- Profilers are tools for gathering performance data and finding performance bottlenecks
  - Implemented by code instrumentation or based on built-in platform debugging / profiling APIs
  - Gather statistics for method calls, uses of classes, objects, data, memory, threads, etc.
- CPU profilers
  - Find performance bottlenecks
- Memory profilers
  - Find memory allocation problems

JustTrace Profiler

- What is JustTrace?
  - Designed for code and memory profiling
  - Measures the frequency and duration of function calls
  - Collects information about memory usage



Telerik Academy



Refactoring Tools

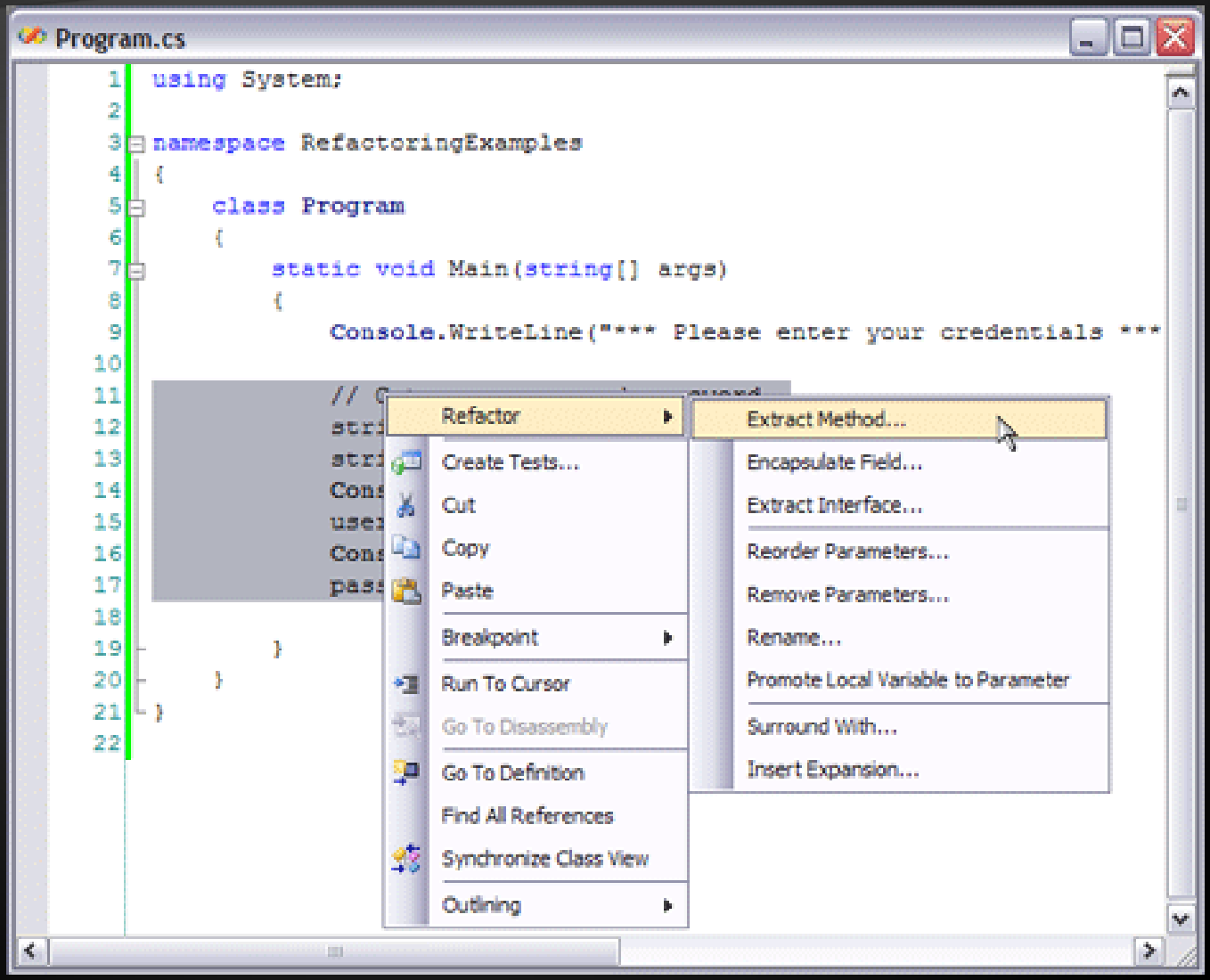
JustCode, ReSharper, IntelliJ IDEA, Visual Studio, Eclipse, Netbeans, JDeveloper

Refactoring

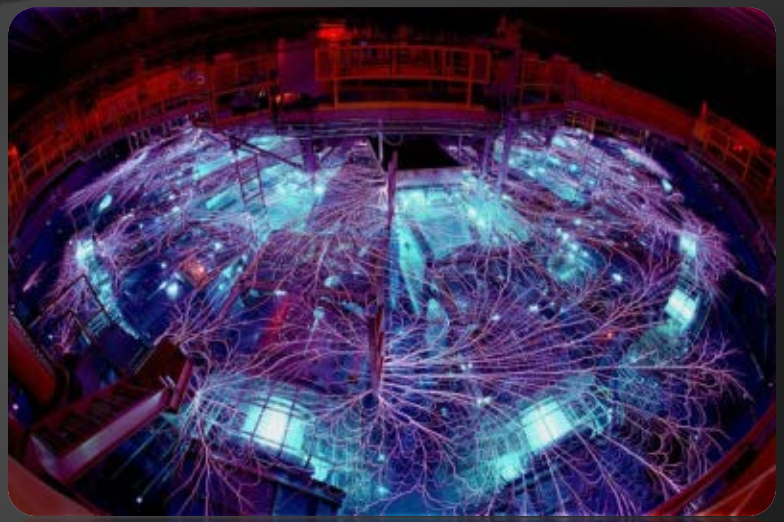
- Refactoring
  - Improving the design of the existing code without changing its behavior
- Typical refactoring patterns
  - Rename variable / class / method / member
  - Extract method
  - Extract constant
  - Extract interface
  - Encapsulate field

Telerik Academy

Refactoring in Visual Studio



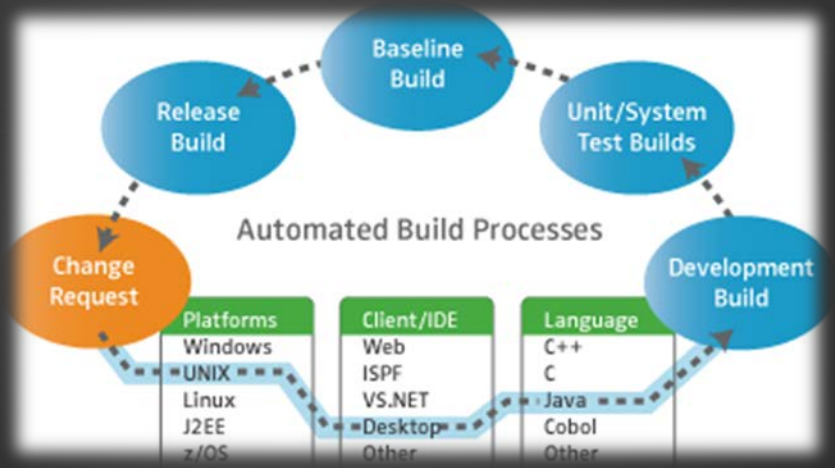
Telerik Academy



Refactoring in Visual Studio with / without JustCode

Live Demo

Telerik Academy



Automated Build Tools

CMake, Ant, Maven, MSBuild

Software Builds

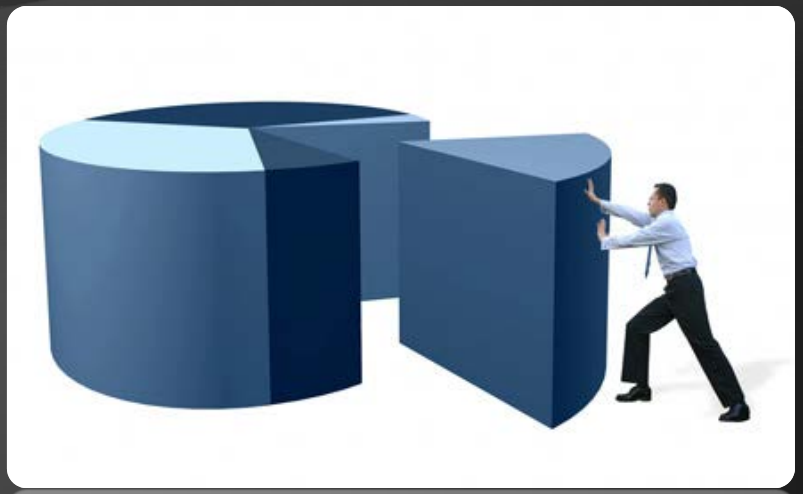
- What means to build software?
  - The process of compiling and assembling the system's modules to obtain the final product
  - Build activities can also include:
    - Getting the latest version from the source control repository
    - Linking external resources
    - Executing unit tests
    - Creating installation packages

Sample MS Build File

- .csproj file is actually a MS Build file

```
<Project xmlns="http://schemas.microsoft.com/developer/msbuild/2003">
  <PropertyGroup>
    <AssemblyName>MSBuildSample</AssemblyName>
    <OutputPath>Bin\</OutputPath>
  </PropertyGroup>
  <ItemGroup>
    <Compile Include="helloworld.cs" />
  </ItemGroup>
  <Target Name="Build">
    <MakeDir Directories="$(OutputPath)"
Condition="!Exists('$(OutputPath)')" />
    <Csc Sources="@(Compile)"
OutputAssembly="$(OutputPath)$(AssemblyName).exe" />
  </Target>
</Project>
```

Telerik Academy



Continuous Integration Tools

CruiseControl, CruiseControl.NET, Hudson, Team Foundation Server

Continuous Integration (CI)

- Continuous integration (CI)
  - Automating the build and integration process
  - Build the entire system each time any new code is checked in the source control repository
  - Run all the automated tests for each build
- What does "continuous" mean?
  - Ideally – build it after every check-in
  - Practically – for larger systems, every 1-2 hours
  - Or at least a couple of times a day

Components of the CI System

- Build server – separate machine (or pool)
- Source control repository
  - Subversion, Team Foundation Server (TFS), etc.
- Automated build system
  - Ant, NAnt, MSBuild, Cruise Control, TFS, etc.
- Status indicators / notifications to make problems visible right away
  - Email notifications / tray build notify utilities
  - Public build status monitors

Telerik Academy

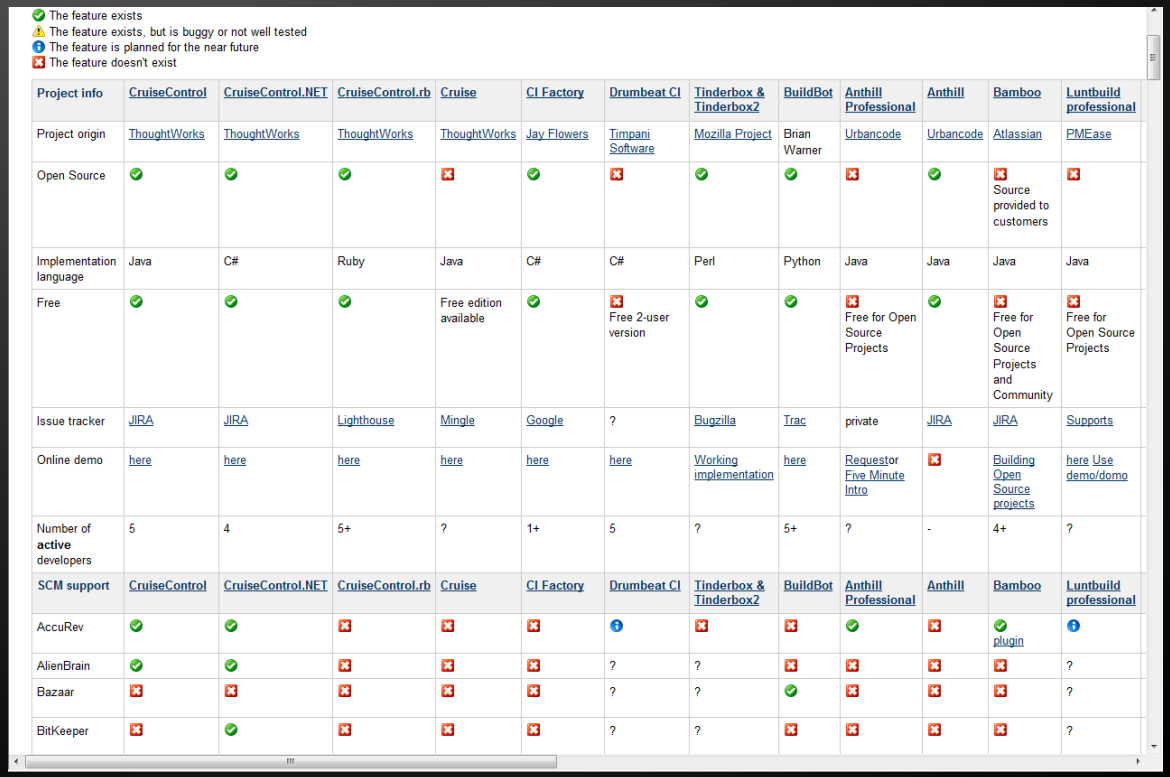
Continuous Integration Systems

- CruiseControl
  - Very popular, powerful, open source CI tool
  - Extensible, plug-in based, large community
- CruiseControl.NET
  - .NET cloning of CruiseControl
- Hudson
  - Powerful Java based CI server, open source
- Team Foundation Server (TFS)
  - TFS provides build-in continuous integration

Telerik Academy

CI Systems Comparison Matrix

- Continuous integration systems – comparison:
  - http://confluence.public.thoughtworks.org/display/CC/CI%2BFeature%2BMatrix



The feature exists
The feature exists, but is buggy or not well tested
The feature is planned for the near future
The feature doesn't exist

Project info	CruiseControl	CruiseControl.NET	CruiseControl.rb	Cruise	CI Factory	Drumbeat CI	Tinderbox & Tinderbox2	BuildBot	Anthill Professional	Anthill	Bamboo	Luntbuild professional
Project origin	ThoughtWorks	ThoughtWorks	ThoughtWorks	ThoughtWorks	Jay Flowers	Timpani Software	Mozilla Project	Brian Warner	Urbancode	Urbancode	Atlassian	PMEase
Open Source	✓	✓	✓	✗	✓	✗	✓	✓	✗	✓	✗ Source provided to customers	✗
Implementation language	Java	C#	Ruby	Java	C#	C#	Perl	Python	Java	Java	Java	Java
Free	✓	✓	✓	Free edition available	✓	✗ Free 2-user version	✓	✓	✗ Free for Open Source Projects	✓	✗ Free for Open Source Projects and Community	✗ Free for Open Source Projects
Issue tracker	JIRA	JIRA	Lighthouse	Mingle	Google	?	Bugzilla	Trac	private	JIRA	JIRA	Supports
Online demo	here	here	here	here	here	here	Working implementation	here	Requestor Five Minute Intro	✗	Building Open Source projects	here Use demo/domo
Number of **active** developers	5	4	5+	?	1+	5	?	5+	?	-	4+	?
SCM support	**CruiseControl**	**CruiseControl.NET**	**CruiseControl.rb**	**Cruise**	**CI Factory**	**Drumbeat CI**	**Tinderbox & Tinderbox2**	**BuildBot**	**Anthill Professional**	**Anthill**	**Bamboo**	**Luntbuild professional**
AccuRev	✓	✓	✗	✗	✗	ℹ	✗	✗	✓	✗	✓ plugin	ℹ
AlienBrain	✓	✓	✗	✗	✗	?	?	✗	✗	✗	✗	?
Bazaar	✗	✗	✗	✗	✗	?	?	✓	✗	✗	✗	?
BitKeeper	✗	✓	✗	✗	✗	?	?	✗	✗	✗	✗	?

Telerik Academy

Documentation Generators

Javadoc, Sandcastle, Doxygen, JSDoc

Documentation Frameworks

- The best source code documentation is the code itself
- Special types of comments are used in many platforms
  - Used to document the code classes, methods, parameters, return types, exceptions, etc.
  - Javadoc comments in Java
  - XML documentation in .NET
  - Doxygen-style documentation for C, C++, PHP, Python, Java, C#, etc.

Documentation Generators

- Javadoc
  - Transforms Javadoc comments used in Java into HTML documentation
- Sandcastle
  - Transforms the XML comments used .NET into CHM, HTML, PDF and other formats
- Doxygen
  - Transforms Doxygen comments into HTML
- JSDoc
  - Javadoc cloning for JavaScript

Telerik Academy

Project Hosting and Team Collaboration Sites

SourceForge, Google Code,
CodePlex, Project Locker

Project Hosting Sites

- SourceForge – http://www.sourceforge.net
  - Source control (SVN, Git, …), web hosting, tracker, wiki, blog, mailing lists, file release, statistics, etc.
  - Free, all projects are public and open source
- Google Code – http://code.google.com
  - Source control (SVN), file release, wiki, issue tracker
    - Very simple, basic functions only, not feature-rich
  - Free, all projects are public and open source
  - 1-minute signup, without heavy approval process

Project Hosting Sites (2)

- CodePlex – http://www.codeplex.com
  - Microsoft's open source projects site
  - Team Foundation Server (TFS) infrastructure
  - Source control (TFS), issue tracker, downloads, discussions, wiki, etc.
  - Free, all projects are public and open source
- Project Locker – http://www.projectlocker.com
  - Source control (SVN), TRAC, CI system, wiki, etc.
  - Private projects (not open source)
  - Free and paid editions

Project Hosting Sites (3)

- GitHub – https://github.com/
  - Web-based hosting service for software development projects
  - Uses the Git revision control system
- Bitbucket – http://bitbucket.org
  - Source control (Mercurial), issue tracker, wiki, management tools
  - Private projects, free and paid editions
- Others: Assembla, Unfuddle, XP-Dev, Beanstalk, etc.

Telerik Academy

GitHub

Live Demo

Telerik Academy

Project Deployment in the Public Clouds

AWS, GAE, Azure, AppHarbor, PHPFog, …

What is Cloud?

- Cloud ≈ multiple hardware machines combine computing power and resources
  - Share them between multiple applications
  - To save costs and use resources more efficiently
- Public clouds
  - Provide computing resources on demand
    - Publicly in Internet
    - Paid or free of charge (to some limit)
  - Azure, Amazon AWS, Google App Engine, AppHarbor, Rackspace, Heroku, …

AppHarbor

- AppHarbor – cloud platform for .NET apps
  - Supports a classical .NET development stack
    - C#, .NET Framework, ASP.NET (Web Forms and MVC), WCF, WWF, ADO.NET Entity Framework, …
  - Deployment through Git / SVN / TFS
    - Automated build process (compilation + unit tests)
  - Build-in load balancing
  - Rich set of add-on services

Telerik Academy

Tools for Developers

http://academy.telerik.com

Homework

1. Research and use the following tools on one of your projects and provide some output or screenshots for each tool to prove that you actually used the tools
   - One source control system (TFS, SVN or Git)
   - log4net
   - StyleCop or JustCode
   - JustDecompile
   - Sandcastle or Doxygen
   - Some obfuscation tool of your choice
2. Write a simple T4 template of your choice
3. Upload anonymously a project to a project hosting site (GitHub, Google Code, etc.) and provide a public link to the project. The project can be from your homework.

Telerik Academy

Free Trainings @ Telerik Academy

- C# Programming @ Telerik Academy
  - csharpfundamentals.telerik.com

- Telerik Software Academy
  - academy.telerik.com

- Telerik Academy @ Facebook
  - facebook.com/TelerikAcademy

- Telerik Software Academy Forums
  - forums.academy.telerik.com