



# Projecto Integrado

## LDAP

Jorge Mendes

2011-02-24

# Summary



- Introduction
- Implementation
  - Technologies
- Code Analysis
  - Tools
  - Results
- Conclusion

# Introduction

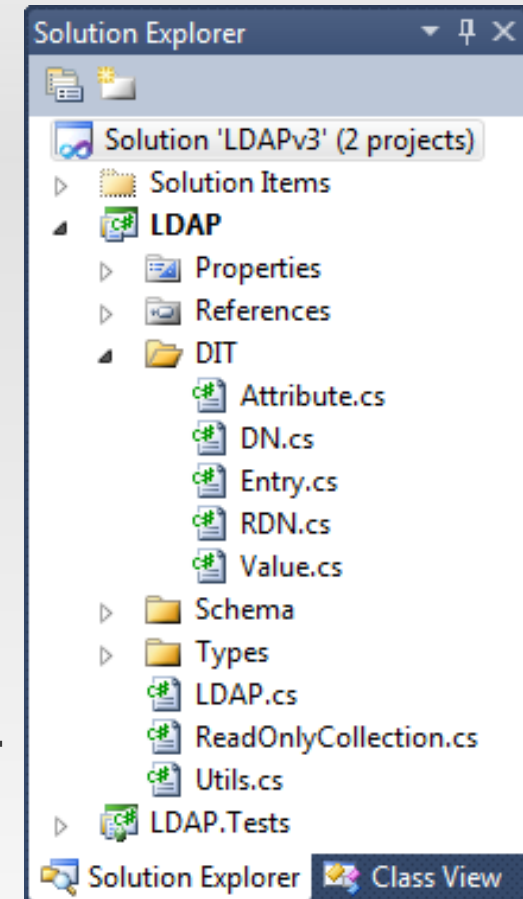


- LDAP: Internet protocol for accessing distributed directory services
- Difficult part:
  - the data model (entries, attributes, ...)
  - schema restrictions (object classes, types, ...)

# Implementation



- Decision: implement a library
- Functionalities:
  - operations on the data model
  - ensure data/schema consistency
- Environment:
  - Platform: Microsoft .NET Framework 4
  - Language: C#
  - IDE: Microsoft Visual Studio 2010 Ultimate



# Implementation Technologies



- Microsoft Code Contracts
  - specification of pre/post-conditions
  - specification of invariants
- Pex and Moles
  - automatic generation of test suites
  - facilitated unit testing

# Implementation

## Code Contracts



- Similar to methods
- Allows:
  - runtime checking
  - static checking

```
public class NOID : OID
{
    private List<uint> value;
```

```
[ContractInvariantMethod]
void ObjectInvariant()
{
    Contract.Invariant(this.value != null);
    Contract.Invariant(this.value.Count >= 2);
}
```



- Pre-conditions

```
public NOID(List<uint> value)
{
    Contract.Requires(value != null);
    Contract.Requires(value.Count >= 2);

    this.value = value;
}
```

- Post-conditions

```
public override string ToString()
{
    Contract.Ensures(
        !String.IsNullOrEmpty(Contract.Result<string>()));
}
```

# LDAP

## Unit Testing



- Useful to test:
  - individual methods
  - functionalities of libraries
- No need to always test the software as a whole

Hierarchy	Not Covered (Blocks)	Not Covered (% Blocks)	Covered (Blocks)	Covered (% Blocks)
▲ Jorge Mendes@PC-CASA 2011-02-22 20:22:24	795	35,08%	1471	64,92%
▲ LDAP.dll	795	35,08%	1471	64,92%
▲ {} LDAP	59	21,93%	210	78,07%
▶ LDAP	25	18,38%	111	81,62%
▶ ReadOnlyCollection1	30	38,96%	47	61,04%
▶ Utils	4	7,14%	52	92,86%
▶ {} LDAP.DIT	337	30,39%	772	69,61%
▶ {} LDAP.Schema	356	45,24%	431	54,76%
▶ {} LDAP.Types	43	42,57%	58	57,43%

Code Coverage Results window showing a table of code coverage data. The table has 5 columns: Hierarchy, Not Covered (Blocks), Not Covered (% Blocks), Covered (Blocks), and Covered (% Blocks). The data is organized in a tree structure under the project name 'Jorge Mendes@PC-CASA 2011-02-22 20:22:24'. The table shows coverage for the entire project, LDAP.dll, and various sub-components like LDAP, LDAP.DIT, LDAP.Schema, and LDAP.Types. The bottom of the screenshot shows a taskbar with several open windows: Error List, Pex Exploration Results, Code Metrics Results, Task List, Output, Code Coverage Results (active), and Test Results.



# Code Analysis



- Focus on Maintainability
- Evaluation of Code Metrics:
  - system volume
  - code duplication
  - unit size
  - unit complexity
  - unit testing

# Code Analysis

## Tools



- Microsoft Visual Studio 2010 Ultimate
  - code contracts static checking
  - unit testing (with code coverage)
- SourceMonitor
  - cyclomatic complexity per unit
  - unit size (in statements)
- Duplo
  - system volume
  - code duplication

# Code Analysis

## Metrics



- System Volume
  - 1477 physical lines of code
- Duplication of code
  - 6 lines (0,4%) – usings were ignored
- Unit size (average)
  - 5 statements (~lines of code)
- Unit complexity (average)
  - 2.18
- Unit testing (code coverage)
  - 64,92%

# Code Analysis

## Unit Size & Complexity



- Unit Size

	Risk			
	Low	Moderate	High	Very High
Relative LOC	732 (79%)	195 (21%)	0	0

- Unit Complexity

	Risk			
	Low	Moderate	High	Very High
Relative LOC	875 (94%)	17 (2%)	35 (4%)	0

# Code Analysis Results



	volume	duplication	complexity per unit	unit size	unit testing	
	5	5	4	4	3	
analysability	×	×		×	×	4
changeability		×	×			5
stability					×	3
testability			×	×	×	4

# Conclusion



- LDAP has lots of restrictions:
  - complex invariants
  - many pre/post-conditions
- LDAP implementations don't support all features
- Microsoft Code Contracts
  - provides a lot of help specifying restrictions
  - is not powerful enough to check all contracts statically
- Pex
  - easy way to check quickly many branches in methods
- Still some things to implement and test...



# Projecto Integrado

## LDAP

Jorge Mendes

2011-02-24