

# Filezilla

Vrchlavský Petr

Open-source programming

OI FEL ČVUT 2012

# Záměr práce

- Podpora vývoje vytvořením dokumentace
- Absence podrobné dokumentace struktury aplikace
- Záměr
  - Vytvořit statický model (pohled) na architekturu aplikace
  - Nejlépe class diagramy
- Požadavky
  - Tvorba modelu nesmí přidělat práci navíc
  - Musí být aktuální
  - Model se musí generovat automaticky společně s pravidelnými buildy
  - Nástroje pro tvorbu modelů musí být open-source a multiplatformní

# Dostupná řešení

- Na trhu existuje 10+ nástrojů na tvorbu UML class diagramů
  - BOUML, StarUML, yUML, EnterpriseArchitect, Violet UML Editor, Umbrello UML Modeller, Doxygen, Argouml, Kivio, Dia, open modelshere, ...
- Pouze některé z nich jsou open-source
  - Violet UML Editor, Umbrello UML Modeller, Doxygen, Argouml, Kivio, Dia
- Ještě méně jich je multi-platformních
  - Umbrello UML Modeller, Doxygen, Argouml, Kivio

# Doxygen + Graphviz

- Generátor dokumentace
- Vstupem může být: C, C++, C#, Java, Python, ...
- Výstupem je dokumentace ve formátu HTML nebo TeX
- Lze spouštět z příkazové řádky
- Společně s Graphviz generuje class diagramy

# Ukázka 1

<a href="#">Main Page</a>	<a href="#">Namespaces</a>	<a href="#">Classes</a>	<a href="#">Files</a>
<a href="#">Class List</a>	<a href="#">Class Hierarchy</a>	<a href="#">Class Members</a>	

## Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

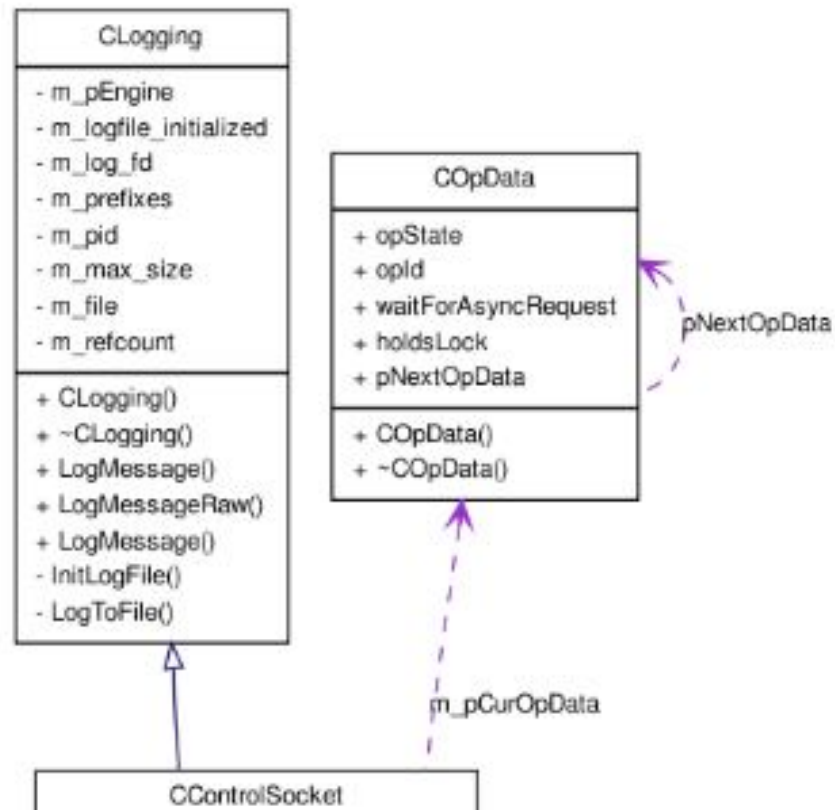
<a href="#">backref</a>	
<a href="#">backref_base</a>	
<a href="#">backref_holder</a>	
<a href="#">CBackend</a>	
<a href="#">CDirectoryCache::CCacheEntry</a>	
<a href="#">CCapabilities</a>	
<a href="#">CChangeDirOpData</a>	
<a href="#">CConnectOpData</a>	
<a href="#">CControlSocket</a>	
<a href="#">CDirectoryCache</a>	
<a href="#">CDirectoryListingParser</a>	
<a href="#">CFileTransferOpData</a>	
<a href="#">CFtpChangeDirOpData</a>	

# Ukázka 2

## CDirectoryListingParser Class Reference

```
#include <directorylistingparser.h>
```

Collaboration diagram for CDirectoryListingParser:



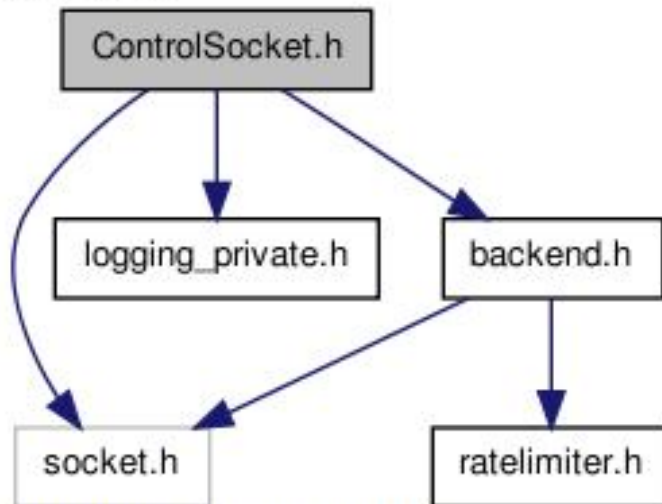
# Ukázka 3



## ControlSocket.h File Reference

```
#include "socket.h"  
#include "logging_private.h"  
#include "backend.h"
```

Include dependency graph for ControlSocket.h:



This graph shows which files directly or indirectly include this file:

# Ukázka 4

Main Page   Namespaces   Classes   Files

---

File List   File Members

---

## ControlSocket.h

[Go to the documentation of this file](#)

```
00001 #ifndef __CONTROL_SOCKET_H__
00002 #define __CONTROL_SOCKET_H__
00003
00004 #include "socket.h"
00005 #include "logging_private.h"
00006 #include "backend.h"
00007
00008 class COpData
00009 {
00010 public:
00011     COpData(enum Command op_Id);
00012     virtual ~COpData();
00013
00014     int opState;
00015     const enum Command opId;
00016
00017     bool waitForAsyncRequest;
00018     bool holdsLock;
00019
```



# Zhodnocení výsledků

- Systém pro generování dokumentace ještě není nasazen
- Popis systému a jeho využití byl zaslán administrátorovi projektu

# Spolupráce s komunitou

- Komunikace probíhá téměř výhradně přes fórum
- Vhodné pro dotazy ohledně vývoje a podpory
- Komunikace přes email je příliš zdlouhavá
  - V mém případě jsem čekal 9 dní na odpověď ohledně návrhu vytvoření dokumentace

# Zdroje

- <http://www.stack.nl/~dimitri/doxygen/>
- <http://www.graphviz.org/>
- <http://filezilla-project.org/>

**Děkuji za pozornost**