

Praktické aspekty vývoje software

Integrovaná vývojová prostředí a nasazení aplikací

Ing. Jan Kouřil, Ing. Jaroslav Dytrych

Fakulta informačních technologií

29. února 2016

- 1 Úvod k vývojovým prostředím
- 2 Funkce IDE
- 3 Techniky nasazení aplikací
- 4 Nástroje pro nasazení

Uživatelská přívětivost

- Zvýrazňování syntaxe
- Doplnění kódu
- Refactoring
- Rychlejší navigace po projektu

Funkčnost

- Verzování kódu
- Debugging
- Automatická kompilace a linkování

Existující vývojová prostředí

Linux

- Anjuta
- KDevelop

Windows

- Visual Studio

Mac

- Xcode

Multiplatformní

- QtCreator
- Code::Blocks
- Eclipse
- Netbeans

Grafická část

- Editor zdrojového kódu
- Správce projektů
- Vestavěná nápověda

Funkce

- Debugger
- Kompilátor
- Linker
- Interpret
- GUI Designer

Příklad IDE

The image shows a Qt Creator IDE window with a C++ source file named `CanvasWidget.cpp`. The code defines a `CanvasWidget` class with several methods:

- `clearMatrices()`: Clears a 4x4 matrix.
- `paintEvent(QPaintEvent *)`: Draws a grid of rectangles on the canvas.
- `mousePressEvent(QMouseEvent *)`: Handles mouse clicks, updating the matrix and redrawing the canvas.
- `mat()`: Returns the current matrix.
- `prev()`: Moves to the previous matrix in the list.

The right-hand pane displays the **Qt Reference Documentation** for `QMouseEvent`. It includes the following sections:

- Qt Reference Documentation**: Home > Modules > QtGui > QMouseEvent
- Contents**: Public Functions, Detailed Description
- QMouseEvent Class Reference**: The QMouseEvent class contains parameters that describe a mouse event. More...
#include <QMouseEvent>
- Inherits:** QInputEvent
 - List of all members, including inherited members
 - Qt 3 support members
- Public Functions**
 - `QMouseEvent (Type type, const QPoint & position, Qt::MouseButton button, Qt::MouseButtons buttons, Qt::KeyboardModifiers modifiers)`
 - `QMouseEvent (Type type, const QPoint & pos, const QPoint & globalPos, Qt::MouseButton button, Qt::MouseButtons buttons, Qt::KeyboardModifiers modifiers)`
 - `Qt::MouseButton button () const`
 - `Qt::MouseButtons buttons () const`
 - `const QPoint & globalPos () const`
 - `int globalX () const`
 - `int globalY () const`
 - `const QPoint & pos () const`
 - `QPointF posF () const`
 - `int x () const`
 - `int y () const`
- Additional Inherited Members**
 - 1 public function inherited from `QInputEvent`
 - 6 public functions inherited from `QEvent`
- Detailed Description**: The QMouseEvent class contains parameters that describe a mouse event. Mouse events occur when a mouse button is pressed or released inside a widget, or when the mouse cursor is...

Automaticky

- Autoindentace
- Automatická tvorba závorek
- Zvýraznění odpovídajících si závorek
- Doplnování jmen proměnných a metod objektů
- Kontrola názvů proměnných

Funkce

- Vyhledání chyby
- Pokročilé vyhledávání
- Přepínání mezi definicí/deklarací
- Přeložit & Spustit

Cíle refactoringu

- Zlepšit interní kvalitu aplikace
- Lepší škálovatelnost
- Vylepšení čitelnosti a udržitelnosti kódu
- Zjednodušení struktury kódu

Metody refactoringu

- Zjednodušení podmínek cyklů a podmíněných výrazů
- Úpravy řetězců (kódování, překlad)
- Zkracování funkcí vytvářením nových
- Vhodná přejmenování funkcí a proměnných

Základy

- Nastavení breakpointů
- Úprava hodnot proměnných za běhu
- Sledování vláken

Funkce procházení

- Krokování
- Vstup/Výstup z funkce
- Pokračování k dalšímu breakpointu

Verzovací systémy

- Subversion
- CVS
- Mercurial
- Git
- Bazaar

Podpora v jednotlivých IDE

- QtCreator
- Netbeans
- Eclipse

Grafický diff

The screenshot shows a graphical diff tool interface. At the top, there is a 'Summary' tab and a 'Diff' button. Below that is a table with columns: Nodes, Time, User, and Message. The table lists three commits: 3494fab (Third commit), 82a1eb6 (Second commit), and 020148c (Initial commit).

Below the table, there are 'Graphical' and 'Textual' tabs. The 'Textual' tab is active, showing a diff between commit 82a1eb63050a722a5e47b909795170d535f38c2 (left) and commit 3494fab8e8255097268c6c30391f2421fc84917 (right). The diff shows changes in the 'private void start()' method. Line 67 is highlighted in green, indicating a change. Line 74 is highlighted in red, indicating a deletion. The code is as follows:

```
private void start(){
  currentIndex=0;
  if (images.isEmpty()){
    Object[] options = { "OK" };
    JOptionPane.showOptionDialog(null, "No images inserted", "Warning",
    JOptionPane.DEFAULT_OPTION, JOptionPane.WARNING_MESSAGE,
    null, options, options[0]);
    return;
  }
  actualImage=(BufferedImage)images.get(currentIndex);
  if (listener!=null)
    listener.actionPerformed(new ActionEvent(this,java.awt.event.ActionEvent.A
  coverMatrices();
  if (listener!=null)
    listener.actionPerformed(new ActionEvent(this,java.awt.event.ActionEvent.A
  }
}
```

Přehled verzí

The screenshot displays the Eclipse IDE interface. The main editor shows the source code for a Python Qt application named 'HelloPython'. The code includes imports for Qt and PyQt4, a class definition for 'HelloPython' with an 'init' method, and a main function that creates an application and displays the window.

```
#!/usr/bin/python
# coding: utf-8 -*-
from PyQt4 import Qt

class HelloPython(Qt.QWidget):
    def __init__(self, parent=None):
        super(HelloPython, self).__init__(parent)
        helloLabel = Qt.QLabel("Say hello to PyQt!")
        helloLineEdit = Qt.QLineEdit()

        mainLayout = Qt.QGridLayout()
        mainLayout.addWidget(helloLabel, 0, 0)
        mainLayout.addWidget(helloLineEdit, 0, 1)

        self.setLayout(mainLayout)

        self.setWindowTitle("My Python App")

if __name__ == '__main__':
    import sys

    app = Qt.QApplication(sys.argv)

    helloPythonWidget = HelloPython()
    helloPythonWidget.show()

    sys.exit(app.exec_())
```

The bottom panel shows the Git commit history for the project 'PYQT'. The table below summarizes the visible commits:

Commit	Author	Date	ID	Committer
Revert "Third commit"	ikauri@pckouri	49 seconds ago	2e910492...	ikauri@pckouri
Third commit	ikauri@pckouri	10 seconds ago	0907044	ikauri@pckouri
Second commit	ikauri@pckouri	3 minutes ago	378502cc...	ikauri@pckouri
Initial commit	ikauri@pckouri	25 hours ago	71767748...	ikauri@pckouri

The commit details for the selected 'Revert "Third commit"' are shown below the table:

```
commit 0f8f2d447ed71750a4fdb1b076d47105aa310
Author: ikauri <ikauri@pckouri> 2012-01-19 15:09:35
Committer: ikauri <ikauri@pckouri> 2012-01-19 15:09:35
Parent: 378502cc06bfc1b249c9e1e147f00baca195ca (Second commit)
Child: 2e9104921ba98538ea0c4d4f435a7c9ad4480d9 (Revert "Third commit")
Branches: master
Third commit
```

Přídavná funkčnost

- Podpora dalších programovacích jazyků
- Vývoj mobilních aplikací
- Nástroje pro testování
- Knihovny

Správce doplňků

The image shows two windows from the NetBeans IDE. The 'Plugins' window displays a list of available plugins, with 'NetBeans API Documentation' selected. The 'Available Software' window shows a tree view of software categories, with 'Web, XML, Java EE and OSGI Enterprise Development' selected.

Plugins Window:

Install	Name	Category	Source
<input type="checkbox"/>	Team Server	Base IDE	
<input type="checkbox"/>	C/C++	C/C++	
<input type="checkbox"/>	Gdbserver	C/C++	
<input type="checkbox"/>	Sakila Sample Database	Database	
<input type="checkbox"/>	JRebel NetBeans Plugin	Debugging	
<input type="checkbox"/>	Module Manager	Developing NetBeans	
<input type="checkbox"/>	Cobertura Module Test Coverage	Developing NetBeans	
<input checked="" type="checkbox"/>	NetBeans API Documentation	Developing NetBeans	
<input type="checkbox"/>	NetBeans Project Metadata Inspector	Developing NetBeans	
<input type="checkbox"/>	Log Management	Developing NetBeans	
<input type="checkbox"/>	CoffeeScript NetBeans	Editing	
<input type="checkbox"/>	Coherence Code Generator	Editing	
<input type="checkbox"/>	OperaPasteAndDependencies	Editing	
<input type="checkbox"/>	(N) for IBM-7.0 Update Center	Editing	
<input type="checkbox"/>	Copy and Paste History	Editing	
<input type="checkbox"/>	AutoSave Module	Editing	
<input type="checkbox"/>	RealTimeWindow	Editing	
<input type="checkbox"/>	Show and change line endings	Editing	
<input type="checkbox"/>	OperaPasteNetBeans	Editing, Tools	
<input type="checkbox"/>	nFeature	Editing, Tools	
<input type="checkbox"/>	Grassy and Goals	Grassy	
<input type="checkbox"/>	AngularJS Tools	HTML, XML	
<input type="checkbox"/>	jBuilder Project Importer	Infrastructure	
<input type="checkbox"/>	Are 1.8.2 Documentation	Java	
<input type="checkbox"/>	Perform Project Extras	Java	
<input type="checkbox"/>	Properties Editor	Java	
<input type="checkbox"/>	Organize Imports	Java	

NetBeans API Documentation Plugin Description:

Version: 2.26
Date: 2017.11
Source: Certified Plugins
Homepage: <https://www.netbeans.org/>

Includes a complete copy of the NetBeans API documentation for offline browsing and javadoc index searching. Covers all of the official API modules plus some APIs considered under development.

Available Software Window:

Work with: <http://download.eclipse.org/releases/indigo>

Find more software by working with the "Available Software Sites" preferences.

Type filter text:

Name	Version
Application Development Frameworks	
Business Intelligence, Reporting and Charting	
Collaboration	
Database Development	
EclipseRT Target Platform Components	
General Purpose Tools	
Linux Tools	
Mobile and Device Development	
Modeling	
Programming Languages	
SOA Development	
Testing	
Web, XML, Java EE and OSGI Enterprise Development	

Select Components Window:

Please [d]select the components you want to [u]install.

Component Name	Installed	New Version	Size
QT SDK	1.1.0-3	1.1.0-3	786.68 Kbytes
Experimental		0.0.3-1	0 Bytes
Hermitian		0.0.2-2	0 Bytes
Hermitian Platform API		10.2011.34-1	207.09 Mbytes
QT Components		1.0	0 Bytes
QT Quick components for Hermitian Simulator integration		1.0	11.92 Mbytes
QT Quick components for Hermitian Desktop integration		1.0	11.95 Mbytes
Hermitian Emulator		10.2011.34-1	517.62 Mbytes
MeCo-1.2 Hermitian API		10.2011.34-1	124.69 Mbytes
QT 4.8.0 RC		4.8.0-3	0 Bytes
Remote Compiler		1.1.0-1	0 Bytes
Documentation	1.1.0-1	1.1.0-1	0 Bytes
APIs	1.1.0-1	1.1.0-1	0 Bytes
Development Tools	1.1.0-1	1.1.0-1	0 Bytes
Miscellaneous	1.1.0-1	1.1.0-1	0 Bytes

Meta package needed to install the SDK.
Update Info: Changelog and README updated
This component will occupy approximately 786.68 Kbytes on your hard disk drive.

Možnosti nasazení aplikací

Windows

- MSI instalátor
- Spustitelný samorozbalující se archiv
- Vytváření binárních souborů ze skriptů

Linux

- Spustitelná binárka
- DEB
- RPM

Mac

- Obraz disku
- Balíky

Možnosti

- Vytváření binárních souborů ze skriptů
- Tvorba přenositelných verzí programu
- Bundlování knihoven
- Začlenění interpretu aplikace
- Využití nástrojů pro nasazení

Bundlované knihovny

- Liší se pro různé operační systémy
- Nutno zahrnout pluginy
- Cesty

C++

- Dynamické knihovny zvoleného frameworku
- Knihovna jazyka C++
- Knihovna GNU C (pokud není v základu OS)
- Spouštěcí skript (je-li třeba upravit cesty ke knihovnám)
- Pluginy v podadresářích

Obsah instalace - příklad

Mac

```
MyApp.app
|-> Contents
    |-> Resources
        | |-> *.png
    |-> Frameworks
        | |-> *.dylib
    |-> MacOS
        | |-> MyApp
```

Windows

```
MyApp.exe
*.dll
*.png
```

Mac

- Všechno v jednom spouštěcím adresáři
- Pluginy v podadresáři Frameworks
- macdeployqt

Podadresář s pluginy - Windows

```
imageformats/  
+----qico4.dll  
      qjpeg4.dll
```

Windows

- Nastavit proměnnou Path
- Spustit příčnou binárku
- Překodovat skript na exe (Vbs2Exe)

Podadresář s pluginy

```
Set objShell = CreateObject("WScript.Shell")
Set objEnv = objShell.Environment("System")
Set envProcess = objShell.Environment("Process")
oldSystemPath = objEnv("PATH")
pathToAdd = ";" & objShell.CurrentDirectory & "\lib"
newSystemPath = oldSystemPath & pathToAdd
envProcess("Path") = newSystemPath
objShell.Run(".\bin\app.exe")
```

Java

- Archiv jar
- Závislosti

Manifest

- Class-Path
- Main-Class

Obsah pracovního adresáře

```
lib/  
app.jar
```

Obsah souboru manifest v jar archivu

```
Manifest-Version: 1.0  
Ant-Version: Apache Ant 1.8.2  
Created-By: 1.6.0_23-b23 (Sun Microsystems Inc.)  
Class-Path: lib/swing-layout-1.0.4.jar  
X-COMMENT: Main-Class will be added ...  
Main-Class: appname.main
```

Python

- Zapouzdření skriptu do binárky
- PyInstaller

Zapouzdření - příklad

- `python ~/PyInstaller-3.1.1/makespec.py main.py`
- `python ~/PyInstaller-3.1.1/pyinstaller.py main.spec`

Nástroje pro nasazení

Windows

- InnoSetup
- NSIS
- Wix

Mac

- Installer for Mac OS X
- VISE X

Linux

- Balíčkovací systémy

Multiplatformní

- InstallJammer
- BitRock Installer

Využívání nástrojů

- Soubory instalátoru
- Licenční ujednání
- Akce instalátoru
- Oinstalátor

Typy balíčků

- Debian
- Red Hat

Převody balíčků

- Alien

Vytvoření debianího balíku

Stažení potřebných nástrojů

```
sudo apt-get install build-essential autoconf  
automake autotools-dev dh-make debhelper  
devscripts fakeroot cdb lintian pbuilder
```

Vytvoření podadresáře debian

```
dh_make -e <email> -n -s -c gpl
```

Důležité soubory podadresáře debian

```
control  
install  
rules
```

Obsah souboru control

- Build-depends

Obsah souboru install

- Soubory pro nakopírování do systému

Obsah souboru rules

```
#!/usr/bin/make -f  
include /usr/share/cdbs/1/rules/debhelper.mk  
include /usr/share/cdbs/1/class/qmake.mk
```

Kroky

- Nahrát vše do adresáře ve formátu packagename-packageversion
- Zabalit vše do nadřazeného adresáře
- *dh_make -e your@emailaddress -f ../yourpackage.tar.gz*
- Upravit soubory control a install
- Případně vytvořit trigger skripty pro instalaci/odinstalaci

Trigger skripty

- preinst
- postinst
- init.d
- prerm
- postrm

Obsah souboru install

```
src/app.xpm /usr/share/pixmaps  
src/app.desktop /usr/share/applications
```

Spouštěč .desktop

```
[Desktop Entry]  
Categories=Qt;Game;  
Exec=app  
GenericName=Simple Game  
Icon=/usr/share/pixmaps/app.xpm  
Name=App  
Type=Application
```

Vytvoření

```
dpkg-buildpackage -rfakeroot
```

Použití

- Software center
- GDebi
- QApt
- Apper
- dpkg
- yum

Vytvoření balíku RPM

Vytvoření sktruktury

```
yum install @development-tools
yum install fedora-packager
/usr/sbin/useradd makerpm
usermod -a -G mock makerpm
rpmdev-setuptree #when logged as makerpm user
```

Struktura

- | -> BUILD
- | -> BUILDROOT
- | -> RPMS
- | -> SPECS
- | -> SOURCES
- | -> SRPMS

Postup vytvoření balíku

Kroky

- Zabalit vše do adresáře SOURCES
- Zahrnout i všechny data, obrázky...
- Napsat skript PACKAGENAME.spec, kde je vše potřebné

Spec soubor










- Vytvoření šablony (rpmdev-newspec)
- Vyplnění sekcí instalátoru

Zkompilování balíku

```
rpmbuild -ba program.spec
```

Sekce instalátoru

```
%prep
#rozbalí vše z adresáře SOURCES do adresáře BUILD
%build
#kompilace adresáře BUILD do BUILDR00T
%pre
#co se má stát před instalací
%install
#kam se má co překopírovat při instalaci
%post
#co se má udělat po instalaci
%preun
#co se stane před odinstalací balíku
%postun
#po odinstalaci balíku
```

-  <http://doc.qt.nokia.com/qtcreator-2.8/>.
-  <http://netbeans.org/kb/index.html>.
-  <http://www.eclipse.org/documentation/>.
-  <http://www.installjammer.com/docs/InstallJammerUserGuide.pdf>.
-  <http://installbuilder.bitrock.com/docs/installbuilder-userguide.pdf>.
-  <http://standards.freedesktop.org/menu-spec/menu-spec-1.0.html>.
-  <http://www.pyinstaller.org/export/latest/trunk/doc/Manual.html>.
-  <http://www.debian.org/doc/manuals/maint-guide/>.
-  http://fedoraproject.org/wiki/How_to_create_an_RPM_package.