

Enigma2 Plugin Entwicklung mit Eclipse



Inhaltsverzeichnis

1	ÜBER.....	3
2	INSTALLATION.....	4
2.1	INSTALLATION VON ECLIPSE.....	4
2.2	INSTALLATION VON PYDEV.....	4
3	EINBINDEN VON ENIGMA2.....	5
3.1	DOWNLOAD DER ENIGMA2 SOURCEN.....	5
3.2	PYTHON INTERPRETER IN ECLIPSE EINSTELLEN.....	5
3.3	ENIGMA2 PROJEKT IN ECLIPSE ERSTELLEN.....	6
4	PYTHON PLUGIN PROJEKT.....	9
4.1	ENIGMA2 PLUGIN IMPORTIEREN.....	9
4.2	ANSICHTEN.....	12
4.2.1	Fehlendes Externes Library.....	12
4.2.2	Fehlendes Source Directory.....	12
5	PLUGIN AUF DIE DREAMBOX KOPIEREN UND TESTEN.....	13
5.1	DATEIÜBERTRAGUNG MIT FTP.....	13
5.2	TELNET UND ENIGMA NEUSTART.....	13
6	VERSIONSKONTROLLE MIT ECLIPSE.....	14
7	WEITERE NÜTZLICHE PLUG-INS FÜR ECLIPSE.....	15
7.1	SUBCLIPSE SVN CLIENT.....	15
7.2	EGIT.....	15
7.3	CDT C/C++ IDE.....	15
8	NÜTZLICHE PROGRAMME.....	15
8.1	FILEZILLA.....	15
8.2	PUTTY.....	15
8.3	VISUAL SVN SERVER.....	15
8.4	TORTOISE WINDOWS EXPLORER INTEGRATION.....	15

1 Über

In diesem Dokument wird beschrieben, welche Programme und Tools benötigt werden, um Python Plugins für Enigma2 mit dem Source Code Editor Eclipse zu entwickeln.

Das Arbeiten mit Eclipse verhilft jeden Entwickler zu höherwertigen Source Code indem er von der Entwicklungsumgebung in allen Belangen unterstützt wird. Einfaches Refactoring sowie anzeige nicht verwendeter Imports oder Variablen. Schon alleine die Möglichkeit direkt an den Methodenaufruf zu springen erleichtert die Software Entwicklung um einiges.

Mit der Möglichkeit Eclipse direkt für die Versionsverwaltung zu nutzen und weitere Funktionen per Pluginsystem nach zu installieren bekommt man ein Entwicklungswerkzeug mit schier unglaublichen Möglichkeiten und Funktionen.

2 Installation

In diesem Kapitel wird beschrieben, wie alle benötigten Tools installiert werden.

2.1 Installation von Eclipse

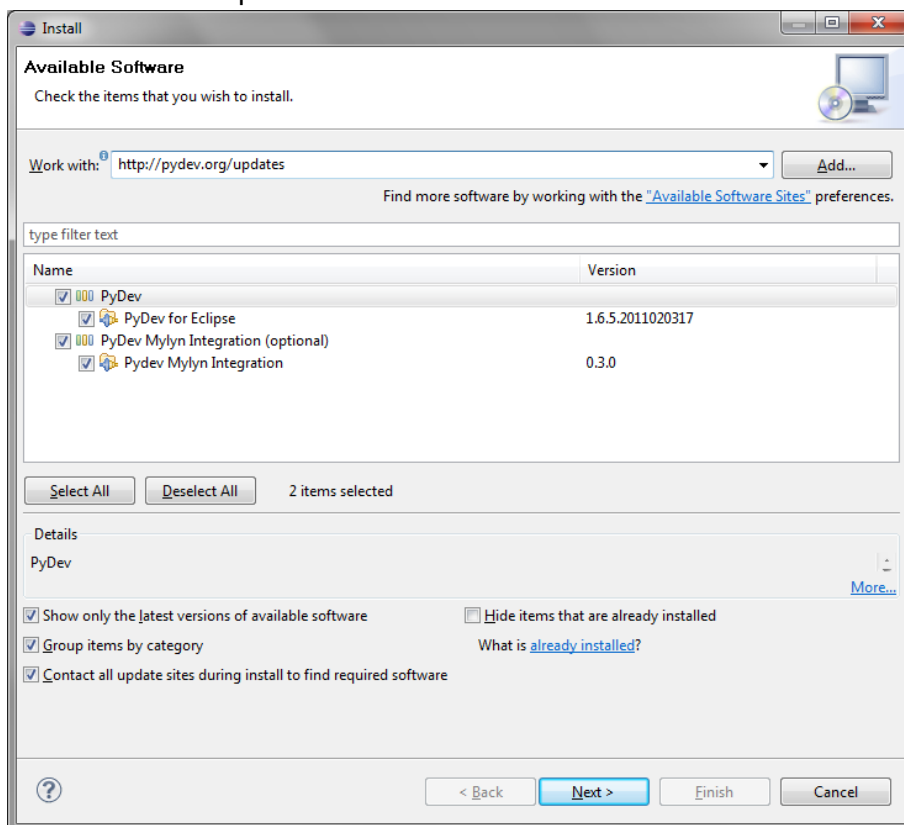
Eclipse ist ein quelloffenes Programmierwerkzeug zur Entwicklung von Software verschiedenster Art.

- Unter <http://www.eclipse.org> ein Eclipse Packet herunterladen. (z.B. Eclipse Classic)
- Anschließend das Paket entpacken.
- Eclipse starten.

2.2 Installation von Pydev

Pydev ist ein third-party plug-in für Eclipse.

- In Eclipse gehe zu “Help” -> “Install new Software...”
- Folgende Quelle zur Update Seite hinzufügen: <http://pydev.org/updates>
- Mit Next alle weiteren Schritte bestätigen.
- Anschließend Eclipse neu starten.



3 Einbinden von Enigma2

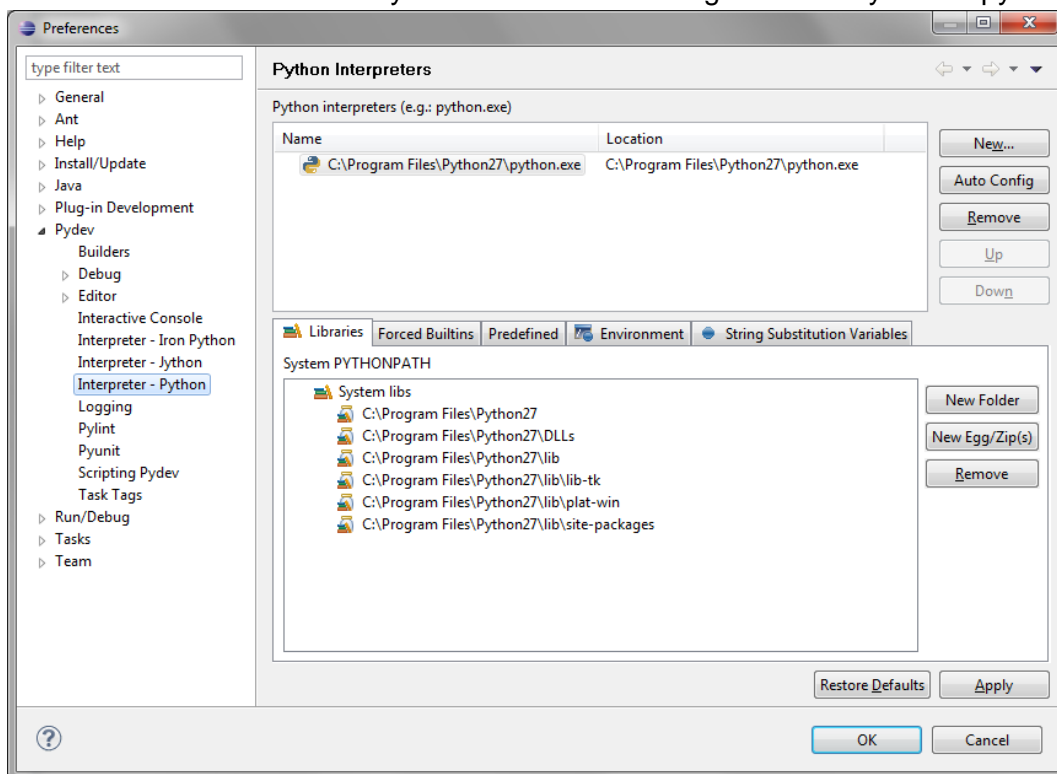
In diesem Kapitel wird beschrieben, wie die Python Files von Enigma2 in Eclipse eingebunden werden.

3.1 Download der Enigma2 Sourcen

- Mit einem ftp Programm z.B. Filezilla auf die Dreambox verbinden.
- Das Directory auf der Dreambox /usr/lib/Enigma2/python komplett herunterladen.
- Da wir die Kompilierten Python Files nicht haben möchten, gehen wir mit einem Dateimanager zu dem heruntergeladenen python Verzeichnis, suchen dort mit F3 nach allen *.pyo und löschen diese.

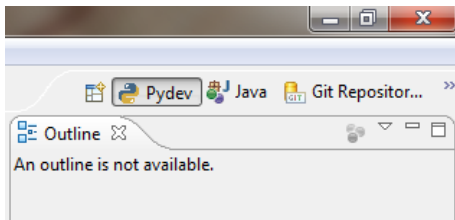
3.2 Python Interpreter in Eclipse einstellen

- Wir gehen in Eclipse auf: **Window -> Preferences**
- Anschließend auf: **Pydev -> Interpreter – Python**
- Auf der rechten Seite auf: New... und navigieren mit Browse... zum Installationsverzeichnis von Python. Normal in: C:\Program Files\Python27\python.exe

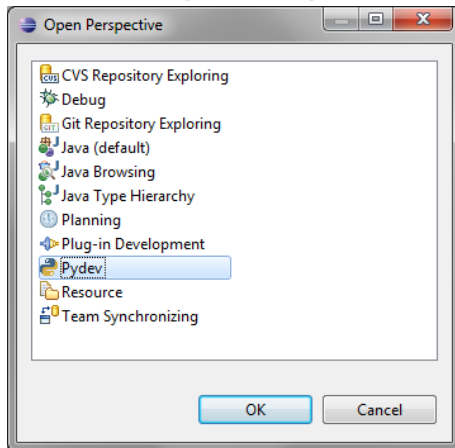


3.3 Enigma2 Projekt in Eclipse erstellen

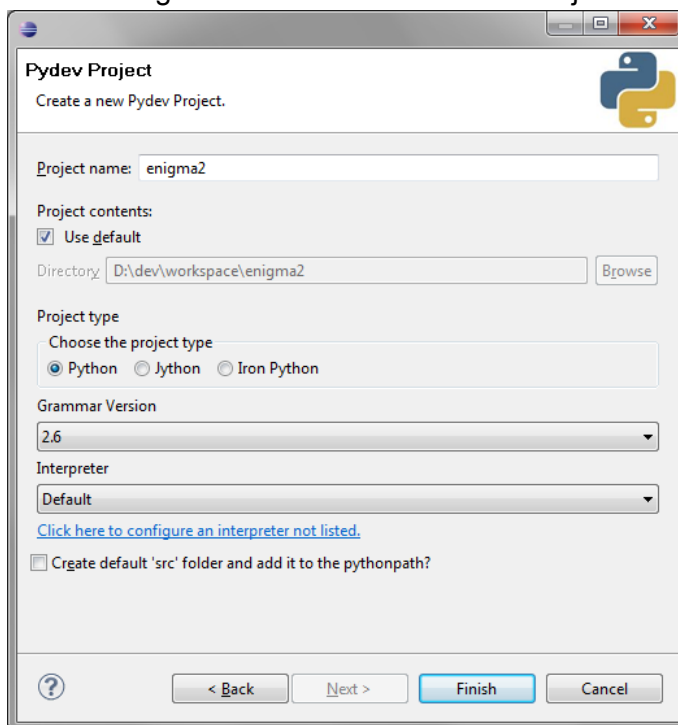
- Zuerst wechseln wir in die Ansicht von Pydev.



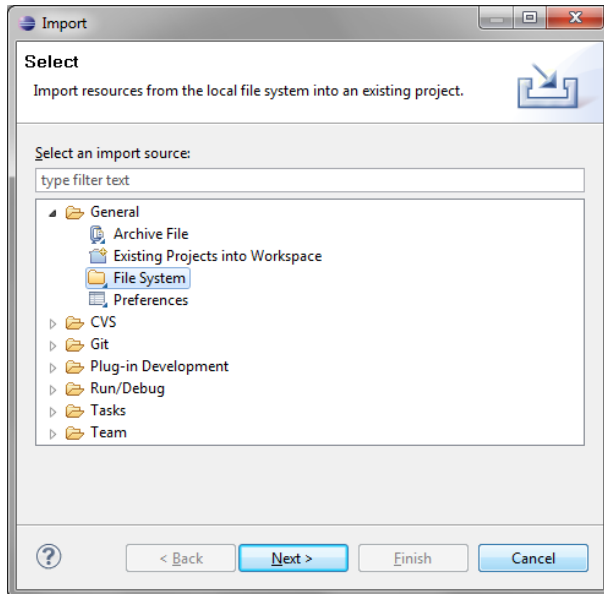
- Wenn es auf der rechten Seite nicht sichtbar ist:
Window -> Open Perspective -> Other...



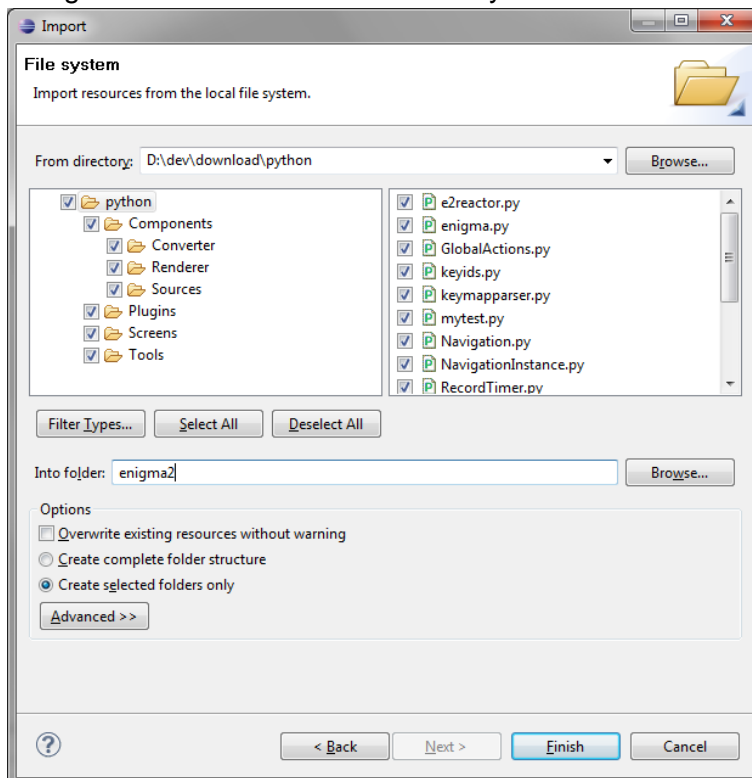
- Wir erstellen ein neues Pydev Projekt: **File -> New -> Pydev Project**
Den Haken bei „Create default 'src' folder“ kann man auch wegklicken. Dann befinden sich die Enigma2 Source Files direkt im Projektverzeichnis und nicht in 'src'.



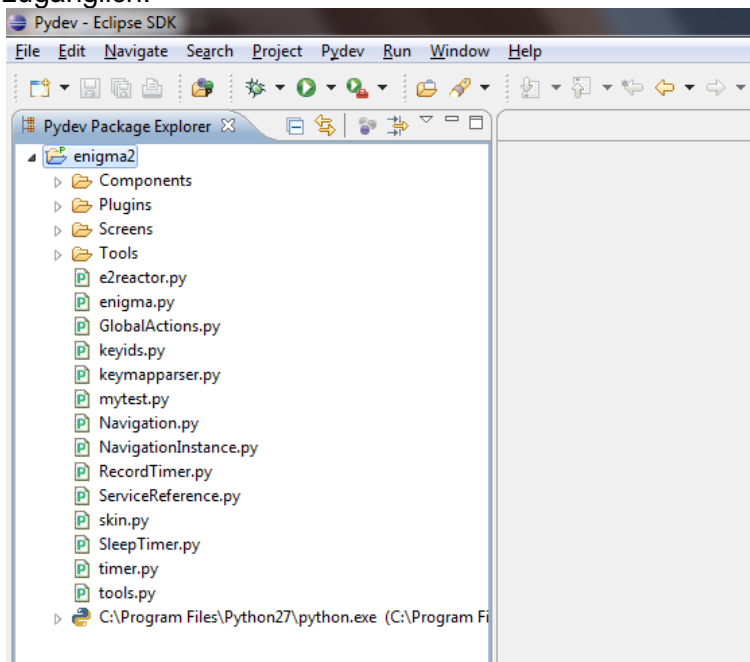
- Rechte Maus Taste auf das Projekt und auf Import: **General -> File System**



- Navigieren mit Browse zu unserem Python Verzeichnis von der Dreambox.



- Jetzt haben wir die Enigma Source Files in einem Projekt und damit für uns immer zugänglich.



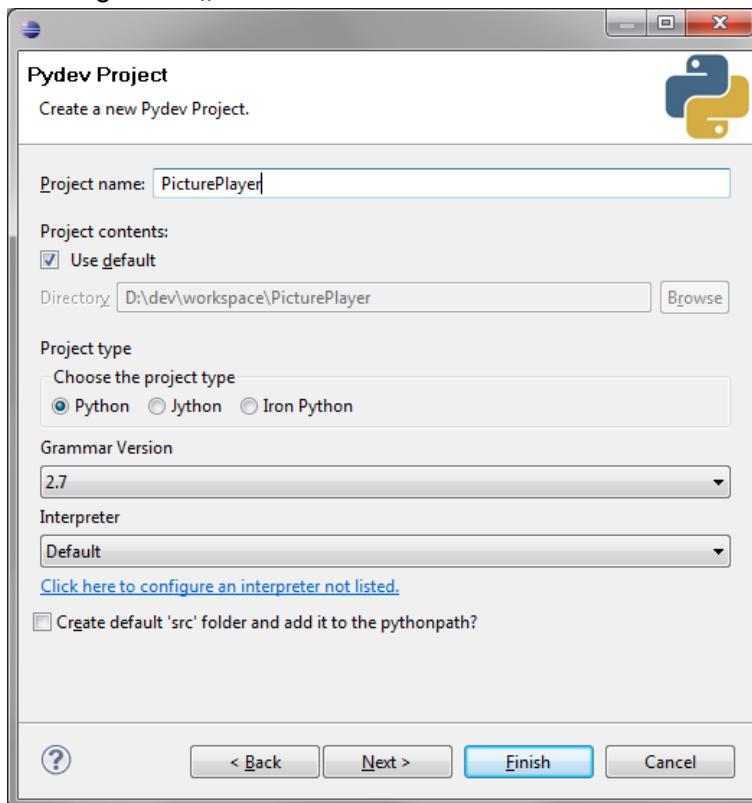
4 Python Plugin Projekt

In diesem Kapitel wird beschrieben, wie ein vorhandenes Plugin zum bearbeiten in Eclipse Importiert wird.

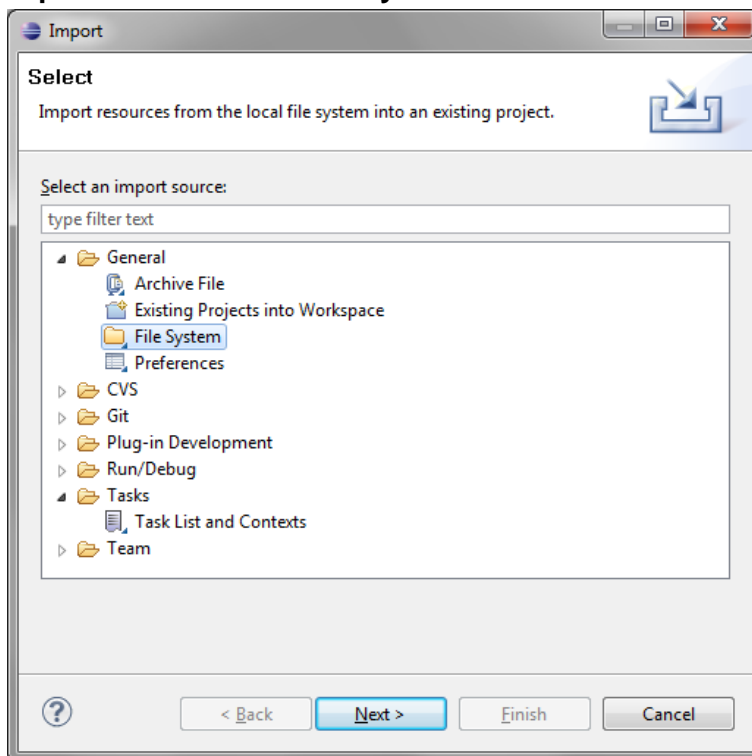
4.1 Enigma2 plugin importieren

Die Vorgehensweise für ein plugin ist genau gleich wie das einbinden von Enigma2 in Eclipse.

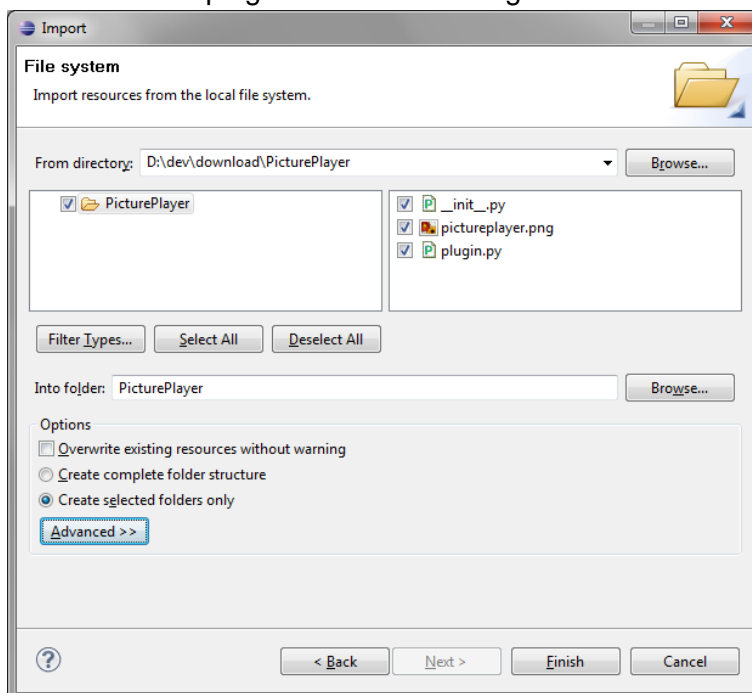
- Wenn das Plugin schon auf der Dreambox installiert war als man die enigma2 Sourcen heruntergeladen hat, kann es von dort als Filesystem importiert werden.
- Ansonsten das plugin installieren.
- Download des plugins mittels FTP Client von der Dreambox: z.B. **PicturePlayer** **/usr/lib/enigma2/python/Plugins/Extensions**
- Löschen der *.pyo
- Neues Projekt anlegen.
- Den Haken „Create default 'src' folder“ lassen wir wieder weg.
- Bestätigen mit „**Finish**“



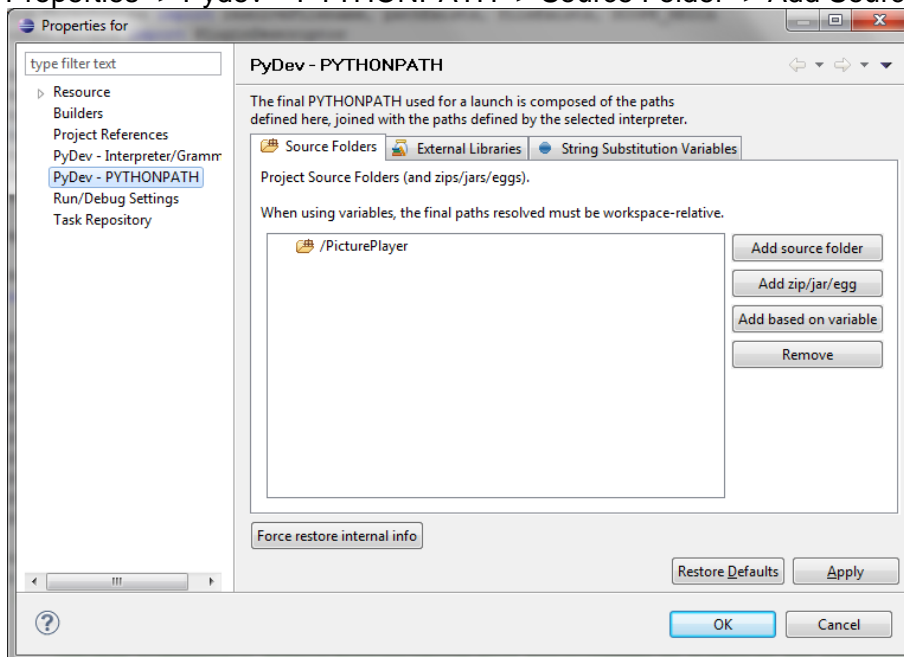
- Im „Pydev Package Explorer“ mit der rechten Maustaste auf das Projekt und auf **Import: -> General -> File System**



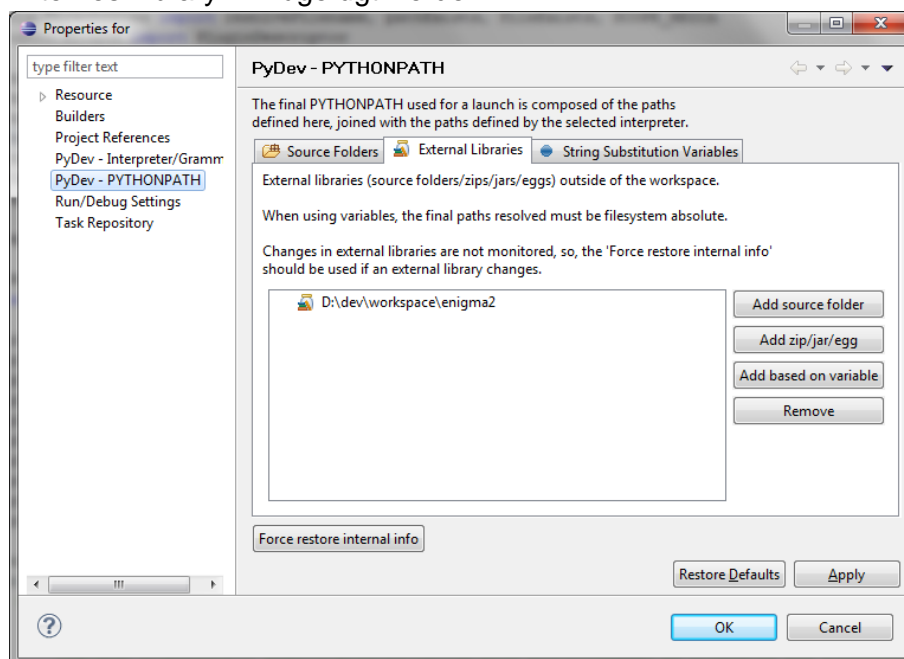
- Mit Brows zum plugin Verzeichnis navigieren und alles auswählen.



- Jeder der das 'src' Verzeichnis nicht gewählt hat muss jetzt im Projekt wieder ein „Source Folder“ auswählen. Das geht mit rechter Maustaste auf das Projekt -> Properties -> Pydev – PYTHONPATH -> Source Folder -> Add Source Folder



- Damit wir im Source Code jetzt noch Komfortabel zu den Imports und Methoden von Enigma2 springen können, muss jetzt nur noch der Enigma2 Python Source als Externes Library hinzugefügt werden.



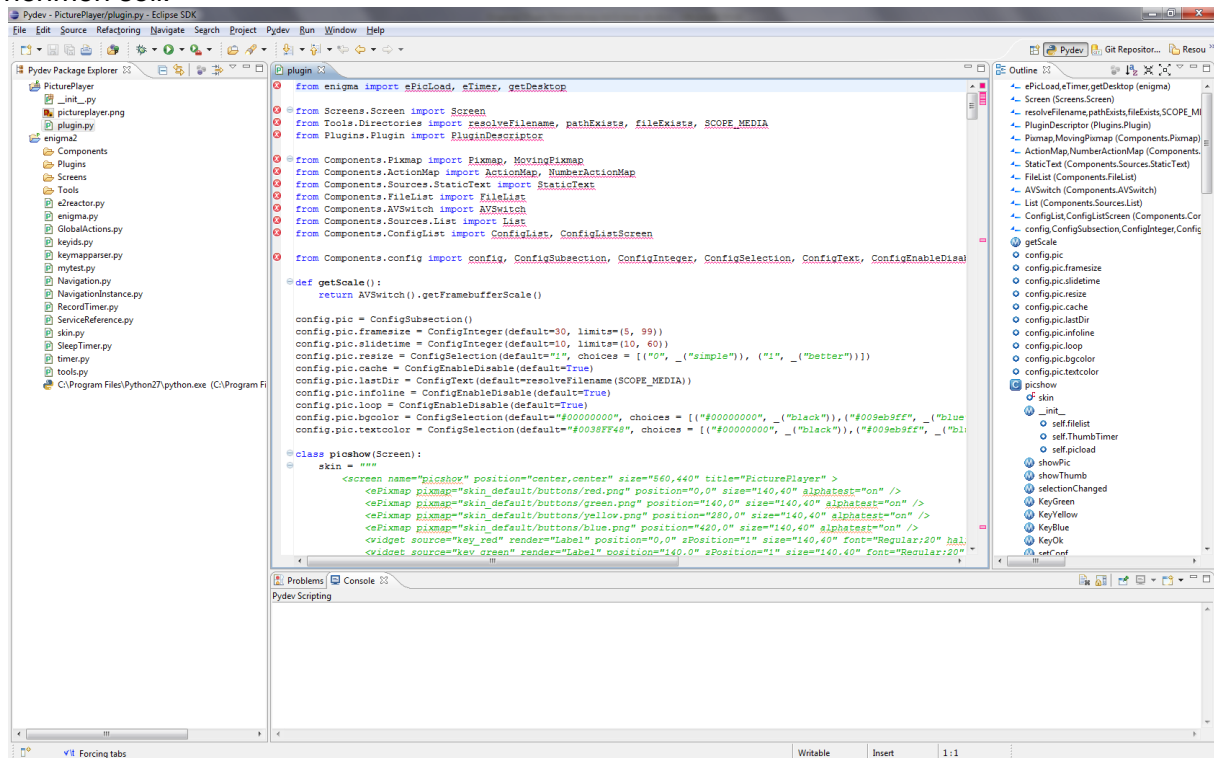
- Jetzt ist es sehr komfortabel möglich, plugins weiterzuentwickeln.

4.2 Ansichten

Hier ist zu sehen, ob alles richtig eingerichtet ist.

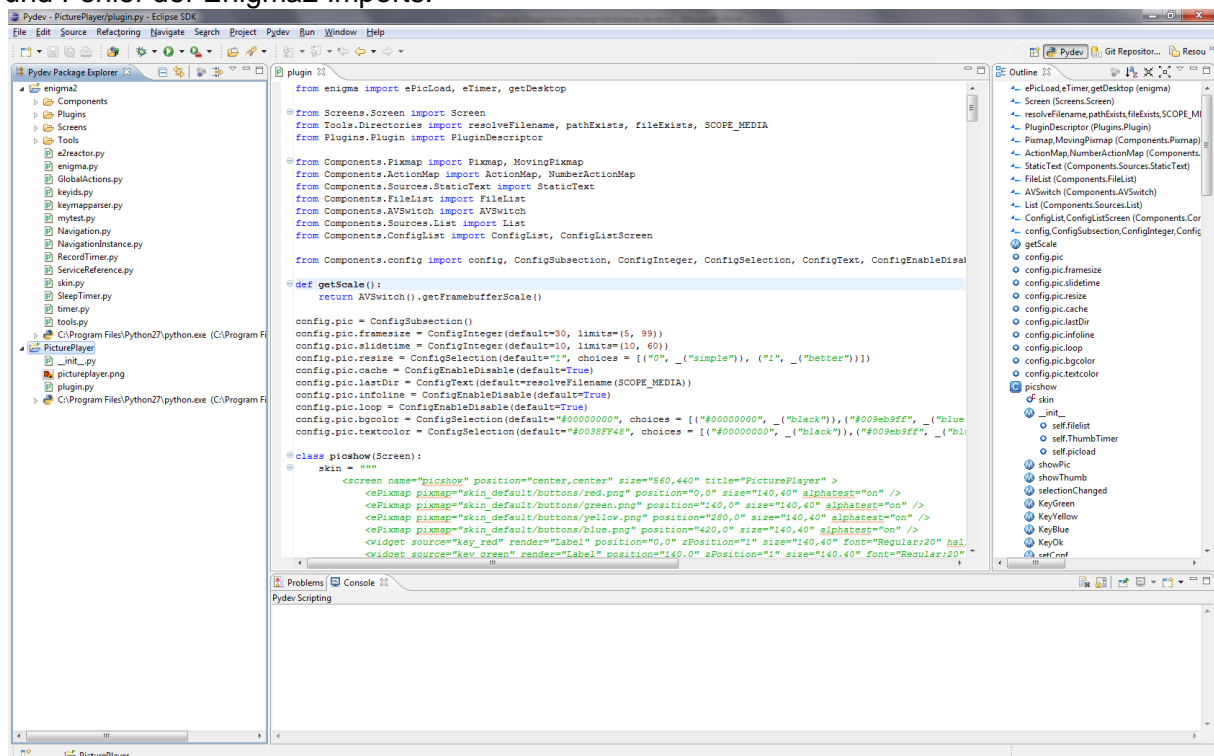
4.2.1 Fehlendes Externes Library

Jeder der das „Source Directory“ richtig gesetzt hat, aber kein `enigma2` als externes Library bekommt diese Ansicht. So ziemlich alles rot weil `Pydev` nicht weis, wo es die Imports her nehmen soll.



4.2.2 Fehlendes Source Directory

Jeder der das „Source Directory“ nicht gesetzt hat, bekommt diese Ansicht. Keine Warnungen und Fehler der `Enigma2` Imports.



5 Plugin auf die Dreambox kopieren und testen

In diesem Kapitel wird beschrieben, wie Plugins auf die Dreambox kommen und man sie Testet.

5.1 Dateiübertragung mit ftp

- Wir brauchen einen FTP Client z.B. Filezilla.
- Wir verbinden uns mit der IP der Dreambox und passwordroot.
- Kopieren am besten immer alle *.py Files auf die Dreambox.

5.2 Telnet und Enigma Neustart

- Wir brauchen einen Telnet Client z.B. Putty
- Host IP Adresse der Dreambox, Port 23
- Login als **root**
- Mit **init 4** wird enigma2 beendet.
- Mit **enigma2** starten wir wieder enigma und sehen auch den stdout mit eventuellen Fehlern.
- Jetzt das Plugin testen und eventuell die Debug Ausgabe beobachten.
- Am Plugin weiterarbeiten und mit dem FTP Client wieder alle *.py Files auf die Dreambox kopieren.
- Mit **CTRL + C** wird enigma wieder beendet.
- Anschließend wieder mit **enigma2** neu starten.
- Alles beginnt wieder von vorne.
- Nach dem Coden wird mit **init 3** enigma neu gestartet und wieder für den normalen Gebrauch verwendbar.

6 Versionskontrolle mit Eclipse

Um eine durchgängige Versionierung der Plugins zu bekommen und nicht an die 100 Kopien der Source Files auf seinem Rechner zu haben sollte die Verwendung eines Versionskontrollsystems in Erwägung gezogen werden. Ob jetzt CVS, SVN oder GIT zum Einsatz kommen soll, muss jeder für sich entscheiden.

In Eclipse ist standardmäßig ein CVS Client Adapter integriert. Es kann aber ohne weiteres jeder andere Versionskontrollsystem Client als Plugin nachinstalliert werden.

Natürlich muss man auch wie für CVS oder SVN einen Server installieren und einrichten. Z.B. muss/sollte angegeben werden wo auf der Festplatte alle Repositories abgelegt werden sollen. Man muss sich trotzdem um die Backups kümmern. Das übernimmt kein Versionskontrollsystem für einen. Das Repository selber muss auch angelegt werden und die User mit Berechtigungen müssen auch angelegt werden.

Eine Empfehlung wäre die Verwendung von SVN. Am besten den Visual SVN Server herunterladen und installieren. Anschließend ein Repository und Benutzer anlegen und im Webinterface nachschauen, ob man Zugriff auf das Repository hat. Die URL für das SVN Repository sieht ihr im VisualSVN Server Manager im Server Status.

Wenn der zu verwendende Client Adapter z.B. „**Subclipse SVN Client**“ installiert ist, geht man einfach auf: Window -> Open Perspective -> Other und dann auf **SVN Repository Exploring**. In der Ansicht unter „SVN Repositories“ klickt man mit der rechten Maustaste, dann auf: **New -> Repository Location** und fügt ein neues Repository hinzu. Die URL ist immer Server/Repository also z.B. so: <https://127.0.0.1:8443/svn/Python/>. Den Server alleine ohne Repository funktioniert nicht.

Wenn das alles ohne Fehler funktioniert hat, geht man wieder auf die Pydev Ansicht, mit der rechten Maustaste auf das Projekt, danach auf **Team** und **Share Project**. Danach muss man sich noch durch einige Abfragen hangeln aber im Grunde ist das kein Problem mehr.

Mit einem Commit (rechtsklick auf Projekt -> Team) werden dann alle Änderungen in das Repository übernommen. Ein Commit ohne Kommentar sollte nicht eingecheckt werden. Im Laufe der Zeit werden jede Menge Änderungen eingecheckt und ohne Kommentar ist die Historie eigentlich sinnlos.

Mit der Versionskontrolle kann jetzt jederzeit entweder für ein File oder das gesamte Projekt die Unterschiede der einzelnen Commits oder der „Working Copy“ angezeigt werden.

7 Weitere nützliche Plug-ins für Eclipse

In diesem Kapitel werden noch weitere nützliche Plug-ins für Eclipse gelistet.

7.1 Subclipse SVN Client

Subclipse ist ein SVN (Subversion) Team Provider Plugin für Eclipse.

- Update Site: http://subclipse.tigris.org/update_1.6.x

7.2 EGit

EGit ist ein git Team Provider Plugin für Eclipse.

- Update Site: <http://download.eclipse.org/egit/updates>

7.3 CDT C/C++ IDE

CDT ist ein Eclipse Plugin für die C/C++ Programmierung.

- Update Site: <http://download.eclipse.org/tools/cdt/releases/helios>

8 Nützliche Programme

In diesem Kapitel werden noch nützliche Programme gelistet. Mittels Suchmaschine jederzeit leicht zu finden.

8.1 Filezilla

FileZilla ist ein modernes und leistungsstarkes FTP-Programm für Windows, MAC und Linux Betriebssysteme.

8.2 PuTTY

Telnet SSH Client.

8.3 Visual SVN Server

Einen freien SVN Server für Windows.

8.4 Tortoise Windows Explorer Integration

Ein sehr nützliches Tool für die Windows Explorer Integration