

# Wechsel auf Mercurial

---

## Zweck des Dokuments

Das Dokument soll die ersten Schritte beschreiben, wie der Wechsel von Subversion auf Mercurial stattzufinden hat und an welchen Speicherorten die einzelnen Repositories gespeichert werden.

## Speicherorte

Bis zum Release der nächsten Debianversion, wird Mercurial lokal auf den Rechner [\\dingo](#) gespeichert. Der Speicherort lautet `c:\Repos\`.

Sollte eines Tages die aktuelle stabile Debian-Version Mercurial 1.3 inkl. der Subrepos-Funktionalität beinhalten, wird dieses gesamte Repository auf den Server verschoben. Der neue Speicherort lautet dann:

```
ssh://mbrenn@code.depon.net/Repos/
```

Dieser Speicherort ist ein HG-Repository, das folgende Unterprojekte beinhaltet:

- Documentation
- Libraries
- Applications
- Projects

## Documentation

Dieses Projekt beinhaltet eine Sammlung verschiedener Schriftstücke, die im Laufe der Zeit projektübergreifend erstellt worden sind.

Dieses Unterprojekt enthält keine weiteren Unterprojekte und die Verzeichnisstruktur kann beliebig gestaltet werden.

## Libraries

Dieses Projekt beinhaltet eine Menge von Unterprojekten.

Der Name eines Unterprojektes ist folgendermaßen aufgebaut:

1. Name des genutzten Frameworks:
  - a. dotnet
  - b. php
  - c. java
  - d. python
  - e. etc...
2. Name des Projektes gemäß der Vorgabe des jeweiligen Frameworks:
  - a. BurnSystems.Parser
  - b. BParser
  - c. de.burnsystems.parser
  - d. etc...

Diese zwei Bestandteile werden durch einen Bindestrich getrennt:

- dotnet-Burnsystems.Parser
- php-BSParser
- java-de.burnsystems.parser

Ziel dieses Projektes ist es eine klare Übersicht über die verfügbaren Bibliotheken zu besitzen. Daher werden sich hier keine kleinen Testprojekte oder Wrapper zum Start einer Bibliothek finden.

Jedes der Projekte besitzt wiederum zwei Unterverzeichnisse:

- src: Hierin befinden sich die Quellcodedateien, die zur Kompilation des Projektes erforderlich sind. Ausgeschlossen sind dynamisch generierte Dateien, wie zum Beispiel das ‚bin‘ und ‚obj‘-Verzeichnis, sowohl die Caches für Stylecop.
- docs: Hierin befindet sich weiterführende Dokumentation, die zu diesem Projekt gehört.

## Applications

Im Unterprojekt Applications werden die Einsprungsprojekte gespeichert, die nicht sinnvoll nach obigem Namensschema benannt werden können. Hier bei sollten die Projekte folgendes Namensschema beinhalten:

- Name des Projektes:
  - Der Name des Projektes kann aus zwei Teilen bestehen:
    1. Allgemeiner Name: Hier sollte das Projekt selbst benannt werden
    2. Name der Datei: Wenn ein Projekt aus mehreren ausführbaren Dateien besteht, kann der zweite Teil zur Unterscheidung genutzt werden.
- Genutztes Framework
  - Siehe oben (dotnet, php, etc)
- Art der Applikation
  - server: Anwendung, die primär als Server dient
  - desktop: Anwendung, die primär vom Nutzer auf dem Desktop gestartet wird und dort auch genutzt wird.
  - website: Eine komplette Website
  - applet-silverlight: Ein Silverlight-Applet

Jeglicher Bestandteil wird durch einen Bindestrich getrennt. Gültige Namen wären demnach:

- JFS-dotnet- server
- FBK-dotnet- server
- FBK-Installer-dotnet-desktop
- Brueckenbauer-dotnet-website
- Brueckenbauer-dotnet-applet-silverlight

Jedes der Projekte besitzt wiederum zwei Unterverzeichnisse:

- src: Hierin befinden sich die Quellcodedateien, die zur Kompilation des Projektes erforderlich sind. Ausgeschlossen sind dynamisch generierte Dateien, wie zum Beispiel das ‚bin‘ und ‚obj‘-Verzeichnis, sowohl die Caches für Stylecop.
- docs: Hierin befindet sich weiterführende Dokumentation, die zu diesem Projekt gehört.

Mit der Unterscheidung zwischen ‚Libraries‘ und ‚Applications‘ sind die wiederverwendbaren von den nichtwiederverwendbaren Projekten getrennt.

## Projects

Ein Projekt ist ein für den Endnutzer geschlossen auftretendes Produkt. Zu einem Produkt gehören alle Unterprojekte, die der Endnutzer zu diesem Produkt zuordnen würde.

Beispielhaft werden Produkte aufgezählt:

- Jungfrauenspiel
- Flexibler Browsergamekern
- Another Spacegame
- Brückenbauer
- Bildabholer

Innerhalb des Produktes gibt es drei Unterverzeichnisse:

- docs
- src

Im ‚docs‘-Verzeichnis befinden sich alle Dateien, die das Projekt als Ganzes beschreiben.

Im ‚src‘-Verzeichnis befindet sich die Verlinkung auf die Unterprojekte, die Visual Studio-Arbeitsmappe und andere bibliotheksübergreifende und bibliotheksverbindende Dateien. Ziel ist es hier die maximale Flexibilität in den Projekten und den Applikationen zu behalten und ein Projekt so zu gestalten, dass es bei einem einfachen Checkout als Ganzes in einem kompilierfähigen Zustand exportiert wird.