

# **Robotik-Projekt WS 2005**

Verantwortlich: Prof. Hommel

Betreuer: Christian Fleischer

Raum: EN256

Tel.: 73114

[fleischer@cs.tu-berlin.de](mailto:fleischer@cs.tu-berlin.de)

# Kurzinfos:

1. Coding Style
2. Doxygen
3. SVN

# Coding Style I

- Keine Tabs
- Englische Worte
- Englische Dokumentation
  
- Worte ohne \_ zusammenschreiben:  
kein `void get_value();`
- Jeden Wortanfang groß:  
`void GetValue();`

# Coding Style II

- Möglichst einzelilige Kommentare (C++-style):

```
if (!IsValid())  
    return 0; // no valid calibration values
```

- Oder:

```
// Check if data is valid and  
// return 0 if invalid.  
if (!IsValid())  
    return 0; // no valid calibration values
```

# Coding Style III

- Eine Datei pro Klasse
- Dateiname = Klassenname
- Keine Variablen *public* deklarieren, sondern:

```
class EMGCalib
{
    protected:
        float Shape,Scale,Range;

    public:
        void Set(float Shape, float Scale);
        float CalcForce(float EMG) const;
        ...
};
```

# Coding Style IV

- Beispiel-Files in

`ExoProject/Projects/ForceControl/Source/EMGCalib.*`

# Doxygen I

- Erzeugen der Dokumentation:

```
~/ExoProject> make doc
```

- Dokumentation unter:

```
~/ExoProject/Documents/DoxyDocs/html/index.html
```

- Doxygen-Dokumentation im Internet:

```
http://www.stack.nl/~dimitri/doxygen/
```

Exoskeleton - Konqueror

Dokument Bearbeiten Ansicht Gehe zu Lesezeichen Extras Einstellungen Fenster Hilfe

Adresse: file:///home/fleischer/ExoProject/Documents/DoxyDocs/html/index.html

SuSE Exo Dienste Firmen Motoren Exoskeleton Google Medizin


**Exoskeleton**

- Main Page
- File List
- Compound List
- Class Hierarchy
- Compound Members
- File Members
- Related Pages

[Main Page](#) | [Class Hierarchy](#) | [Alphabetical List](#) | [Compound List](#) | [File List](#) | [Compound Members](#) | [File Members](#) | [Related Pages](#)

# Exoskeleton Documentation

---

Generated on Mon Nov 7 12:16:45 2005 for Exoskeleton by  **doxygen** \_1.3.2





# Doxygen IV

```
////////////////////////////////////  
/// \file  
/// \brief Implementation of EMGCalib.h  
////////////////////////////////////  
...  
  
////////////////////////////////////  
/// Calculates the force output of a muscle corresponding to the given...  
/// \param EMG The muscles activation.  
/// \return Force of the muscle for a given activation. Is always >= 0.  
////////////////////////////////////  
  
float EMGCalib::CalcForce(float EMG) const  
{  
...  
}
```

# SVN I

- Repository holen:

```
svn co svn://user@dontcry/ExoProject ExoProject
```

- Sourcen+Doku übersetzen:

```
~/ExoProject> make
```

- Eigenes Unterverzeichnis in `Projects` erstellen

# SVN II

- Neuerungen von anderen einspielen:

```
~/ExoProject/.../GeaendertesVerzeichnis> svn update
```

- Eigene Neuerungen ins Repository  
(erstes Mal)

```
~/ExoProject/Projects/MeinProjekt> svn add files dirs
```

# SVN III

- Eigene Neuerungen ins Repository (Änderungen)

```
~/ExoProject/Projects/MeinProjekt> svn commit -m "Text"
```

- **WICHTIG:** Wer SVN nicht kennt, bitte **vorher** Doku lesen **und** mich fragen!!

! Viel Spaß !

