PostgreSQL-Installationsanleitung für Windows

# Benötigte Software

* Entpacker
  + Z.B. 7Zip <http://www.7-zip.org/>
* MinGW
  + <http://sourceforge.net/projects/mingw/files/>
  + Automated MinGW Installer
  + MSYS Base System
  + MinGW API for MS-Windows
* Perl
  + Z.B. ActivePerl <http://www.activestate.com/activeperl/>
* PostgreSQL Quellen
  + <http://www.postgresql.org/ftp/source/>
* pgAdmin – PostgreSQL GUI
  + <http://www.pgadmin.org/download/windows.php>
* Eclipse IDE for C/C++ Developers
  + <http://www.eclipse.org/downloads/>

# Installation

* MinGW
  + Benötigte Module
    - MinGW Base Tools
    - G++ Compiler (C++ Compiler)
    - Object C Compiler
    - MinGW Make
* MSYS Base System
  + Nachinstallationsprozedur durchlaufen, hier must der Pfad zur MinGW-Installation angegeben werden
* MinGW API for MS-Windows
  + Entpacken (ohne Wurzelordner) nach <MSYS\_INSTALL>\mingw
* Perl
  + Alle Module
  + Perl in die PATH-Umgebungsvariable eintragen (Installationsoption)
* PostgreSQL Quellen
  + In den Workspace entpacken (mit Wurzelordner)
* pgAdmin – PostgreSQL GUI
  + MSI entpacken und ausführen
* Eclipse
  + Entpacken

# Compilieren

* MSYS starten
* Ins Wurzelverzeichnis der PostgreSQL-Quellen wechseln
  + cd /c/<WORKSPACE>/postgresql-X.X.X
* Vorbereiten
  + ./configure --enable-debug --enable-cassert --enable-depend --without-zlib --prefix=<LATER\_POSTGRESQL\_INSTALL>
  + <LATER\_POSTGRESQL\_INSTALL> muss eine absolute Pfadangabe sein, z.B.: /c/<WORKSPACE>/postgresql-X.X.X-bin
  + Pfad variable setzen
    - <LATER\_POSTGRESQL\_INSTALL>/bin und <LATER\_POSTGRESQL\_INSTALL>/lib in Umgebungsvariable PATH eintragen
* Compilieren
  + make
* Installieren ins Verzeichnis <LATER\_POSTGRESQL\_INSTALL>
  + make install

# Starten und Stoppen

* In der Konsole (cmd)
* initdb -D <DATABASE\_DIR>
* Starten: pg\_ctl -D <DATABASE\_DIR> -l <LOG\_FILE> start
* Status: pg\_ctl -D <DATABASE\_DIR> status
* Stoppen: pg\_ctl -D <DATABASE\_DIR> stop
* Starten ist nur von einem unprivilegierten Nutzer aus möglich
  + Standardnutzer anlegen, z.B. postgresql
  + Nutzer Vollzugriff auf <WORKSPACE>/postgresql-X.X.X-bin geben
  + Mit runas starten:   
    runas /user:<POSTGRESQL\_USER> "pg\_ctl -D <DATABASE\_DIR> -l <LOG\_FILE> start"
* Datenbank erzeugen: createdb <DATABASE\_NAME>
* SQL-Konsole: psql <DATABASE\_NAME>
  + Anzeigen aller vorhanden Tabellen: \d
  + Beenden: \q

# In Eclipse einbinden

* Eclispe starten
* Neues C-Projekt erstellen
  + Project location: /<WORKSPACE>/postgresql-X.X.X
  + Project type: Makefile Project -> Empty Project
* Debuggen
  + Datei configure abpassen
  + -O2 durch -O0 ersetzen (Suchen/Ersetzen)
  + Neu vorbereiten (configure) und Kompilieren (make) und Installieren (make install)
  + In Eclipse Debug Konfiguration erstellen
    - Neue Konfiguration unter "Attach to C/C++ Application"
    - Applikation: <LATER\_POSTGRESQL\_INSTALL>\bin\postgres.exe
  + Debug ausführen
    - Eclipse als <POSTGRESQL\_USER> starten
    - Postgres Server als <POSTGRESQL\_USER> starten
    - PID ermitteln
      * PSQL starten
      * select pg\_backend\_pid(); ausführen
    - Debug starten und die postgres.exe mit der richtigen PID auswählen
    - Breakpoints einrichten und auf "Resume" klicken
    - SQL Statement in PSQL absetzen

# Dokumentation

* Kommentare im Quellcode
* Manual
  + <http://www.postgresql.org/docs/8.4/interactive/index.html>
  + Generelle Doku für die Benutzung
  + Zum Entwickeln interessant ist Abschnitt Internals
* Entwicklungsbezogene Präsentationen
  + <http://www.postgresql.org/developer/coding>
* Toturial: Introduction to Hacking PostgreSQL
  + <http://neilconway.org/talks/hacking/>
* Entwickler-Wiki
  + <http://wiki.postgresql.org/wiki/Development_information>