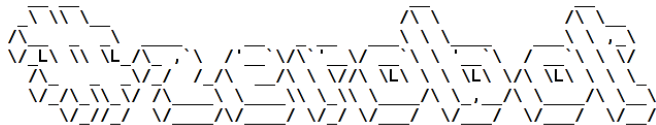


# Martin Wegner



# Features

- ▶ C++
- ▶ RFC<sup>1</sup>-konformer Parser des (relevanten) Client-Protokolls  
"Two-Level-Parser"
  - ▶ „Syntaktischer Parser“ nach (Pseudo-)BNF im RFC
  - ▶ „Semantischer Parser“ bildet Protokollnachrichten in Klassenhierarchie ab
- ▶ *Generator*: Einfache Serialisierung der Klassen über überladene Operatoren.
- ▶ *Flexible Plug-In-Infrastruktur*.
- ▶ *OpenSource* ;-)



---

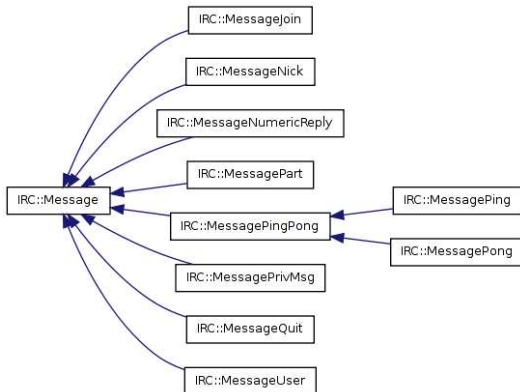
<sup>1</sup><http://tools.ietf.org/rfc/rfc1459.txt>

# Parser: BNF und Klassenhierarchie

```
<message> ::= [ ':' <prefix> <SPACE> ] <command> <params> <crlf>
<prefix>   ::= <servername> | <nick> [ '!' <user> ] [ '@' <host> ]
<command>  ::= <letter> { <letter> } | <number> <number> <number>
<SPACE>    ::= ' ' { ' ' }
<params>   ::= <SPACE> [ ':' <trailing> | <middle> <params> ]

<middle>   ::= <Any *non-empty* sequence of octets not including SPACE
               or NUL or CR or LF, the first of which may not be ':'>
<trailing> ::= <Any, possibly *empty*, sequence of octets not including
               NUL or CR or LF>

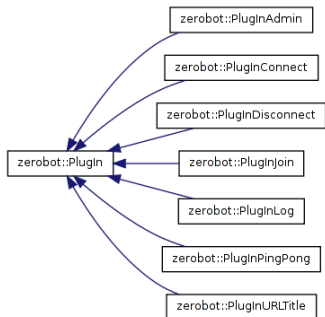
<crlf>     ::= CR LF
```



# Plug-Ins

- ▶ **Globaler Bot-Zustand:** connecting, connected, disconnecting, disconnected
- ▶ **Plug-In-Interface** mit vier Callbacks:
  - ▶ *Aktiv:*
    - ▶ `onConnect(state)`
    - ▶ `onTimeTrigger(state)`
    - ▶ `onDisconnect(state)`
  - ▶ *Reaktiv:*
    - ▶ `onPacket(state, receivedPacket)`
- ⇒ **Resultat:** Neuer Bot-Zustand, Liste zu versendender Nachrichten
- ▶ **Vorteile:** Einfach und mächtig (genug).
- ▶ **Nachteile:**
  - ▶ *Grob*, aber Verfeinerung der Zustände über weitere lokale Plug-In-Zustände.
  - ▶ *Lokal*, (noch) keine Kommunikation zwischen Plug-Ins möglich.

# Implementierte Plug-Ins



- ▶ **Verbindungsauf- und abbau**
- ▶ **PingPong**
- ▶ **Administration** Befehlsinterface für privilegierten Benutzer
- ▶ **Channel-Beitritt** (Join)
- ▶ **Logging von (Protokoll-)Nachrichten** in Channels in SQLite-Datenbank
- ▶ **Titel von URLs**, die im Channel in Nachrichten anderer Benutzer gefunden wurden

# URL-Titel

## Ausgangssituation:

- ⇒ Kein „einfacher“ (X)HTML-Parser für C/C++ gefunden
- ⇒ Gerade für nur teilweise geladene Dokumente nicht („Lazy-Parsing“)

## Lösung:



1. **curl** lädt z. B. die ersten 10 KiB von einem Dokument herunter
2. **TidyLib** erzeugt aus der Eingabe gültiges XHTML.
3. **eXpat** kann XHTML parsen, da XML-Dialekt.

# Libraries und Tools

## Benutzte Libraries:

- ▶ *libcurl*
- ▶ *TidyLib*
- ▶ *eXpat*
- ▶ *SQLite*
- ▶ *glibc* (Sockets, getopt, ...)

## Eingesetzte Tools:

IDE KDevelop4, vim

SCM git, subversion

Buildsystem CMake

Memchecker valgrind

Debugger gdb