



Softwareschneiderei

Extreme Feedback Devices

Erfahrbare Qualität





Softwareschneiderei

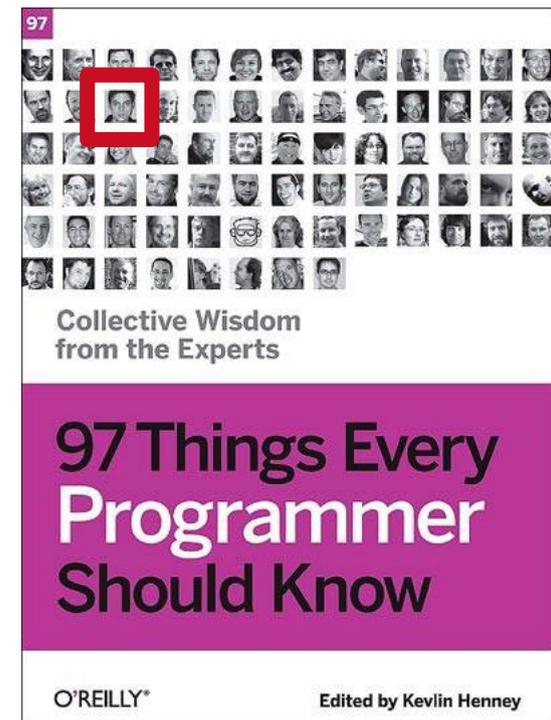
Softwareschneiderei GmbH





Daniel Lindner

- Diplom-Informatiker (KIT)
- Mitgründer der Softwareschneiderei GmbH
- Dozent für Softwaretechnik (DHBW)
- Softwareentwickler
- Fachbuchautor





Agenda

- Begriffserklärung
- Motivation
- Theorie
- Ausprägungen
- Beispiele
- Umsetzung
- Fragen und Diskussion



Hintergrund

Menschen nehmen ständig Informationen aus ihrer Umwelt auf





Information Radiators

Präsentieren Informationen so, dass sie im Vorbeigehen aufgenommen werden können





Daten werden physikalisiert

Informationen dringen so in die physikalische Welt ein und beeinflussen diese





Was ist „Feedback“?

Antwort eines Systems auf die Änderung eines äußeren Umstands





Feedback erfahren

Wir reagieren sehr sensibel auf unsere Umwelt





Was ist „Extreme Feedback“?

Unmittelbare Rückmeldung, die nicht ignoriert werden kann





Extreme Feedback

Normales Kommunikationsmittel zwischen Lebewesen





Was ist ein „Extreme Feedback Device“?

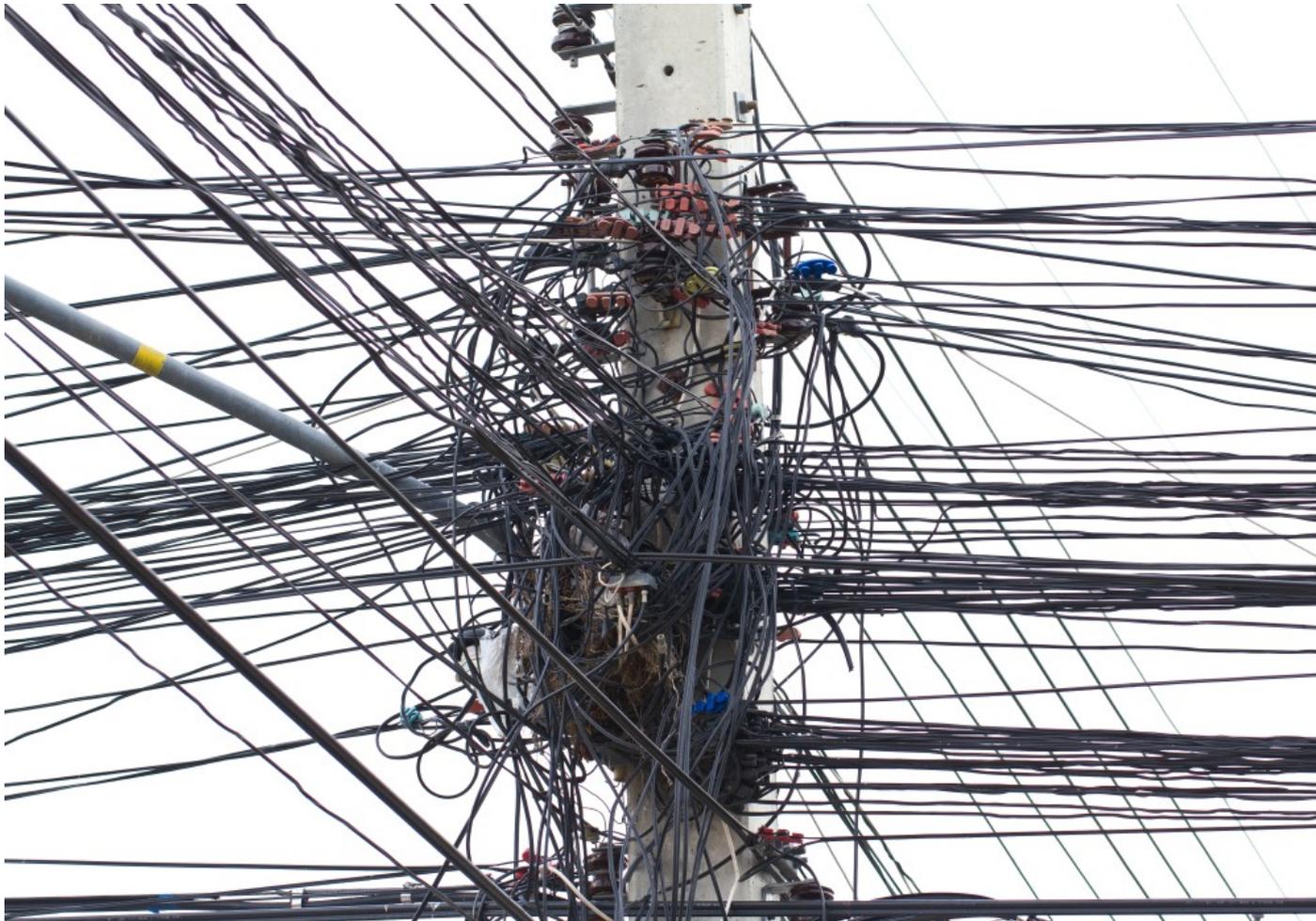
Ein Gerät zur unmittelbaren, nicht ignorierbaren (Rück-)
Meldung einer Änderung





Wozu brauchen wir Rückmeldungen?

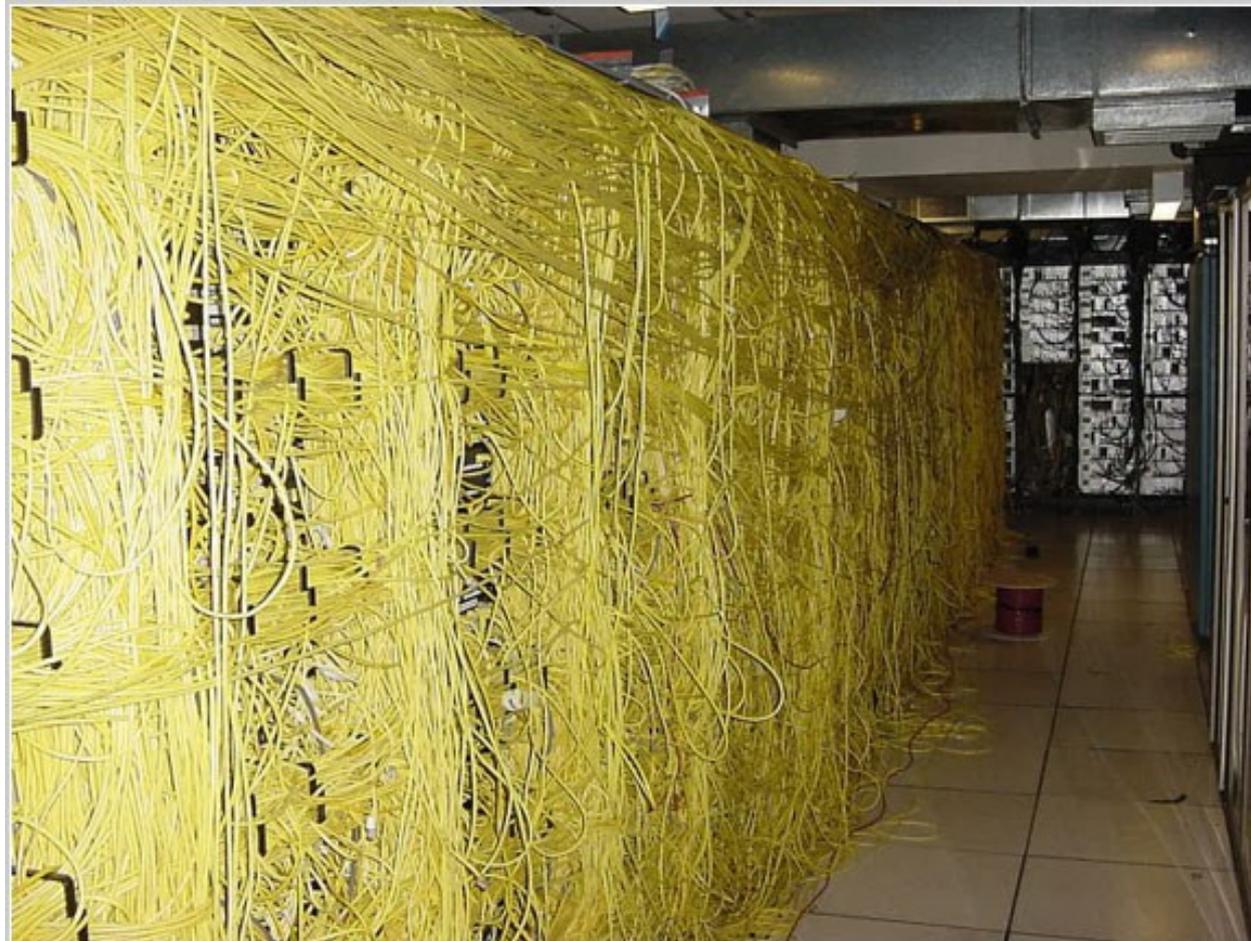
Team-Arbeit bedeutet: Keiner ist es gewesen





„Broken Window“-Effekt

Jeder fügt wirklich nur ein einziges Problemchen hinzu





Wer soll zurückmelden?

Der Betroffene selbst – unser Projekt.

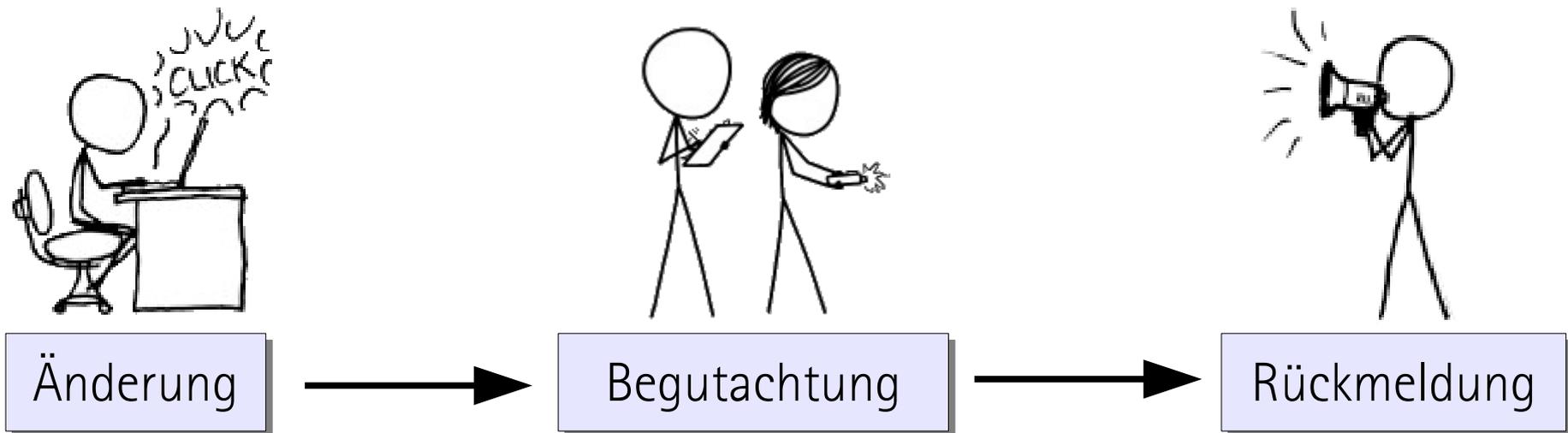
Es ist allerdings stumm. Geben wir ihm eine Stimme!





Das Projekt erhält eine Stimme

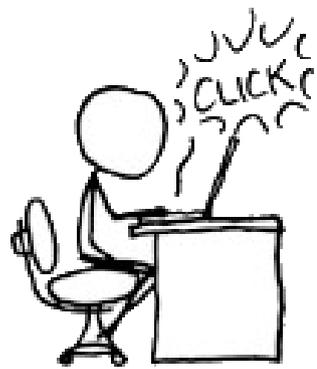
Das Projekt begutachtet jede Änderung an sich selbst und meldet das Ergebnis zurück





Continuous Integration (CI)

Wir haben bereits eine Vorkehrung zur Selbstprüfung der Projekte – den Build Server



Änderung



Begutachtung



Rückmeldung



Übliche Rückmeldung des CI

Der CI-Server gibt das Ergebnis per Webseite oder E-Mail bekannt





Extreme Rückmeldung

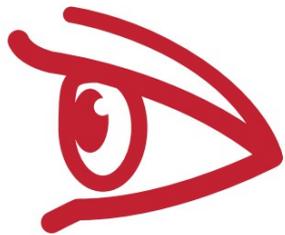
Wichtige Meldungen (z.B. gebrochener Test) erhalten durch die Art der Übermittlung eine „natürliche Wichtigkeit“





Ausprägungen der Rückmeldungen

Menschen besitzen meistens 5 Sinne, von denen der erste überproportional häufig verwendet wird



Sehen



Hören



Tasten



Riechen



Schmecken



Visuelle Wahrnehmung (Sehen)

Wichtigster Sinn für Software-Entwickler
Sehr hohe Aufnahmekapazität, ausrichtbar





Auditive Wahrnehmung (Hören)

Klassischer Kommunikationskanal (v.a. Sprache)
Mittlere Aufnahmekapazität, nicht ausrichtbar





Haptische Wahrnehmung (Tasten)

Findet durch „Force Feedback“ langsam Verwendung
Hohe Kapazität, ausrichtbar





Olfaktorische Wahrnehmung (Riechen)

Wird in der Medientechnik (fast) nicht verwendet
Mittlere Kapazität, nicht ausrichtbar, langsam





Gustatorische Wahrnehmung (Schmecken)

Keinerlei Verwendung außerhalb Mittagspause (Essen)
Sehr niedrige Kapazität





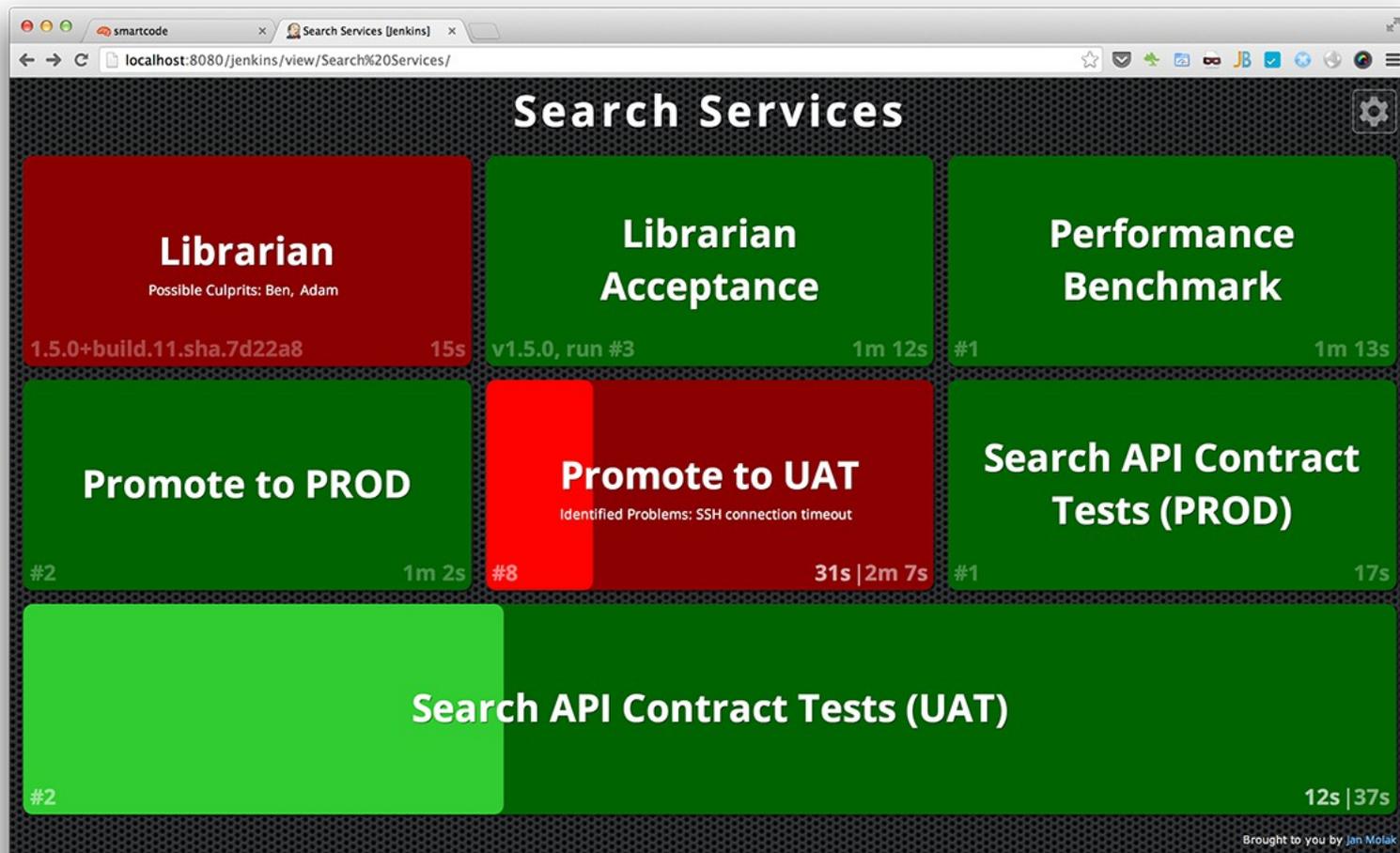
Nutzen für Extreme Feedback

Sehen		Universell, viele Codierungen: Schrift, Farbe, Symbole
Hören		Vor allem für überraschende Unterbrechung geeignet
Tasten		Dringt in persönliche Nahzone ein
Riechen		Zu langsame Informationsweitergabe
Schmecken		Ungeeignet



Beispiele für XFDs

Große Wand-Monitore zeigen Zustandsübersicht





Beispiele für XFDs

LED-Leuchtschriftenleiste zeigt textuelle Rückmeldung

<http://www.ziegelmeier.net/?q=XFD>





Beispiele für XFDs

Mehrere Lampen in Kombination

<http://www.thiswayup.de/2009/01/extreme-feedback-device/>





Beispiele für XFDs

Mehrere farbige Lampen in Kombination

<https://wiki.jenkins-ci.org/pages/viewpage.action?pageId=20250625>





Beispiele für XFDs

Mehrfarben-LED in Glaskugel („Orb“) für Farbwechsel

http://weblogs.java.net/blog/kohsuke/archive/2006/11/diiorb_my_own_e.html





Beispiele für XFDs

Lichterkette mit einzeln ansteuerbaren Lampen

<http://www.deepdarc.com/2010/11/27/hacking-christmas-lights/>

Arduino

RGB-LED





Beispiele für XFDs

Keyboard mit einzeln farbig beleuchtbaren Tasten





Beispiele für XFDs

Analoge Anzeige zeigt digital ermittelte Werte

<http://blog.last.fm/2008/08/01/quality-control>





Beispiele für XFDs

Zimmerbrunnen zeigt an, ob Bauvorgänge im CI laufen





Fast alles kann ein XFD sein

Jeder Gegenstand, der verschiedene Zustände anzeigen kann, ist ein potentielles XFD





Eigene XFDs ansteuern

Extreme Feedback Devices lassen sich erstaunlich schnell aufbauen





Signal to Steckdose



NET-PwrCtrl PRO, LAN, 8 Sockel, 160 €

<http://anel-elektronik.de/index.htm?src=SITE/produkte/pro/pro.htm>



Websteckdose, LAN & WLAN, 3 Sockel, 189 €

<https://www.woehlke-websteckdose.de/>

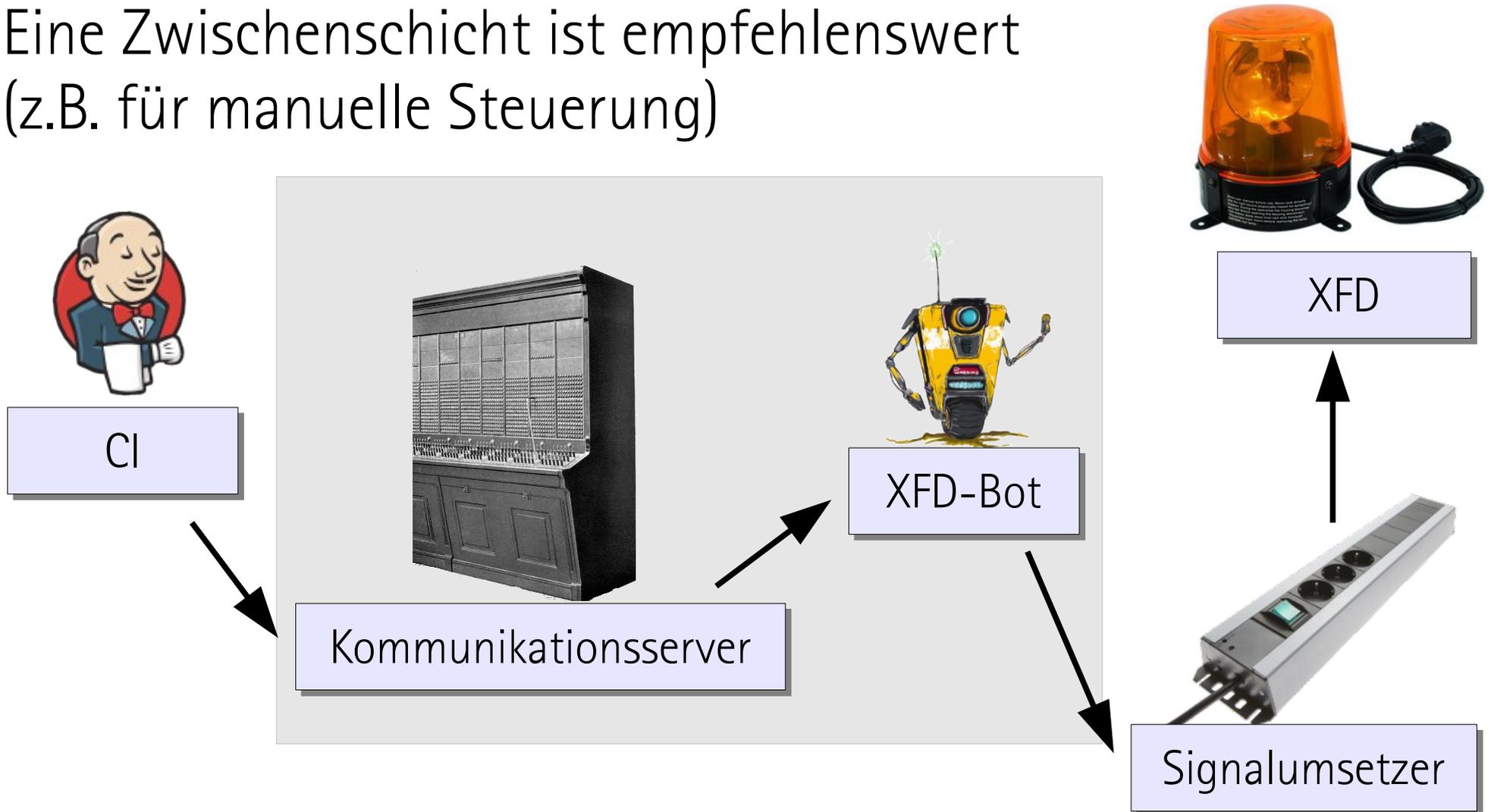


Gembird Silvershield, USB, 4 Steckdosen, ~30 €

<http://www.gembird.nl/item.aspx?id=3234&lang=de>

Systemarchitektur für XFD

Eine Zwischenschicht ist empfehlenswert
(z.B. für manuelle Steuerung)





XFD-Bot mit PircBot

Für das Kommunikationsprotokoll IRC gibt es eine gute Java-Bibliothek für IRC-Bots

Wir entwickeln einen Bot, um über eine Websteckdose eine Lampe zu steuern (an/aus)



XFD-Bot in Java mit PircBot #1

The screenshot shows an IDE with two main panes. On the left is the Package Explorer, and on the right is the code editor for Main.java.

Package Explorer:

- XFD-Bot
 - src
 - com.schneide.xfd
 - Main.java
 - JRE System Library [JavaSE-1.7]

Main.java Code:

```
package com.schneide.xfd;

public class Main {

    public static void main(String[] args) {
        // Bots be here
    }
}
```



XFD-Bot in Java mit PircBot #2

PircBot - Java IRC Bot Fram... x +

www.jibble.org/pircbot.php

jibble.org

PircBot Java IRC Bot

The most popular Java IRC API

[Download PircBot 1.5.0](#) (about 200kb)

PircBot Java IRC API Resources

- [PircBot Javadoc Documentation](#)
- [PircBot Tutorial Video](#) (New)
- [PircBot Developers' FAQ](#)
- [PircBot Change Log](#)
- [PircBot Implementations](#)
- [PircBot lecture slides](#) (PowerPoint)
- [PircBot lecture slides](#) (PDF 6-per-page)
- [PircBot Forums](#) (external link)

Java IRC API

PircBot is a Java framework for writing IRC bots quickly and easily. Its features include an event-driven architecture to h... included, and this page contains a 5-minute step-by-step guide to making your first IRC bot.

PircBot allows you to perform a... it is also used for more serious applications by the US Na

Main

- [www.jibble.org](#)
- [Contact Me](#)

My Books

- [IRC Hacks](#)
- [Impossible Sudoku](#)

Photography

- [DoF Calculator](#)
- [KiteCam](#)
- [Stacking Teleconverters](#)
- [F1 London](#)
- [Tethered Shooting](#)

IRC Bots

- [PircBot IRC API](#)
- [PieSpy IRC Bot](#)
- [Comic Bot](#)
- [SpeechBot](#)
- [Monty Quotes](#)
- [ReminderBot](#)
- [LogBot](#)

Misc. Articles

- [Meaning of Jibble](#)



XFD-Bot in Java mit PircBot #3

The screenshot shows an IDE window with two panes. The left pane is the Package Explorer, showing a project named 'XFD-Bot' with a source folder 'src' containing a package 'com.schneide.xfd' and a file 'Main.java'. Below this, the 'Referenced Libraries' section shows 'pircbot.jar' and a 'lib' folder. The right pane shows the code for 'Main.java'.

```
package com.schneide.xfd;

public class Main {

    public static void main(String[] args) {
        // Bots be here
    }
}
```



XFD-Bot in Java mit PircBot #4

The screenshot shows an IDE window with two panes. The left pane is the Package Explorer, showing a project named 'XFD-Bot' with a source folder 'src' containing a package 'com.schneide.xfd' and a file 'Main.java'. Below this are 'JRE System Library [JavaSE-1.7]', 'Referenced Libraries' (including 'pircbot.jar'), and a 'lib' folder. The right pane shows the code for 'Main.java'.

```
package com.schneide.xfd;

public class Main {

    public static void main(String[] args) throws Exception {
        LampBot bot = new LampBot();
        bot.connect("my.ircserv.er");
        bot.joinChannel("#jenkins");
    }
}
```



XFD-Bot in Java mit PircBot #5

```
package com.schneide.xfd;

import org.jibble.pircbot.PircBot;

public class LampBot extends PircBot {

    public LampBot() {
        setName("lampbot");
    }

}
```



XFD-Bot in Java mit PircBot #6

```
package com.schneide.xfd;

import org.jibble.pircbot.PircBot;

public class LampBot extends PircBot {

    public LampBot() {
        setName("lampbot");
    }

    on
}
```

onAction(String sender, String login, String hostname, String target, String action)

This method is called whenever an ACTION is sent from a user. E.g. such events generated by typing "/me goes shopping". Most IRC clients.

The implementation of this method in the PircBot class performs no actions and may be overridden as required.

Parameters:

- sender** The nick of the user that sent the action
- login** The login of the user that sent the action
- hostname** The hostname of the user that sent the action
- target** The target of the action, be it a channel or a user
- action** The action carried out by the user.

Press 'Ctrl+Space' to show Template Proposals

Press 'Tab' from proposal table to insert



XFD-Bot in Java mit PircBot #7

```
package com.schneide.xfd;

import org.jibble.pircbot.PircBot;

public class LampBot extends PircBot {

    public LampBot() {
        setName("lampbot");
    }

    @Override
    protected void onMessage(String channel, String sender,
        String login, String hostname, String message) {
        if (message.contains("FAILURE")) {
            switchLampOn();
        }
    }

    private void switchLampOn() {
        // change web power socket here
    }
}
```



XFD-Bot in Java mit PircBot #8

```
package com.schneide.xfd;

import java.io.IOException;

public class LampBot extends PircBot {

    public LampBot() {
        setName("lampbot");
    }

    @Override
    protected void onMessage(String channel, String sender,
        String login, String hostname, String message) {
        if (message.contains("FAILURE")) {
            switchLampOn();
        }
    }

    private void switchLampOn() throws IOException {
        final URL onCommand = new URL(
            new URL("http://lanpower1.intranet"),
            "cgi-bin/schalten?steckdose_nr=1&steckdose_soll=1");
        final URLConnection connection = onCommand.openConnection();
        connection.getInputStream();
    }
}
```



Der XFD-Bot im Einsatz

- Virtuelle Maschine mit CI- und IRC-Server
- IRC-Plugin im CI installiert
- XFD-Bot lokal in Eclipse gestartet
- Websteckdose im lokalen Netzwerk
- Lampe an Websteckdose
- Testprojekt im lokalen Eclipse
- Projekt wird vom CI überwacht



Vielen Dank





Softwareschneiderei

Softwareschneiderei GmbH





Bildnachweise

Time Square - Sean Pavone / Shutterstock

Three wise monkeys - Jakub Hatun / Wikipedia

Speed display - Paul Velgos / Shutterstock

Build bears - Dr. Simon Wiest (simonwiest.de)

Snow warning - Earl D. Walker / Shutterstock

Claptrap - Borderlands / Gearbox Software

Train departure - karnizz / Shutterstock

Multiple lamps - ABAS Software AG

Car dashboard - Ryszard Filipowicz / Shutterstock

Build dashboard - Jan Molak / SmartCode Ltd

Goosebumps - Sergieiev / Shutterstock

Analog display - last.fm

Alarm clock - artjazz / Shutterstock

Nabaztag - Catalarem / Wikipedia

Air horn - Volodymyr Krasyuk / Shutterstock

Stick people - Randall Munroe / xkcd.com

Power cable mess - benokky1972 / Shutterstock

Paper garbage - Aleks vF / Shutterstock

Chili pepper - Maks Narodenko / Shutterstock

Hot stove plate - THPStock / Shutterstock

Five senses - Subidubi / Shutterstock

Loudspeaker - alarich / Shutterstock

All other images: unknown / Internet

Ants on hand - spot-h / Shutterstock

Or by Softwareschneiderei GmbH (own production)