

# How to pay back

Ein paar einfache Wege, seinem Projekt zu helfen

Alexander Schmehl

Debian Projekt

29. Oktober 2005, Linux-Info-tag, Dresden

- 1 Motivation
- 2 Ein paar lächerlich einfache Jobs
- 3 Ein paar einfache Jobs, mit speziellerem Wissen
- 4 Thanks!

- Ich habe zu oft Phrasen wie „Ich würde ja gerne helfen, aber was kann *Ich* schon tun? Ich kann ja noch nicht mal programmieren!“, gehört
- Es ist ein geläufiges Vorurteil, dass mal ein „richtig k3wl h4x04“ sein muss, um helfen zu können
- Es ist eine ziemliche Verschwendung, nicht auch unerfahrenere aber lernfähige Benutzer einzubinden

Der beste Tipp, den ich geben kann, ist:

Seit nicht überempfindlich!  
Nehmt's nicht zu schwer.

Der zweitbeste Tipp, den ich geben kann, ist:

Gebt nicht zu schnell auf!

# Sagt ab und zu mal “Danke,,

Chris Higgins reichte einmal folgende Fehler gegen Debians OpenOffice.org ein:

<http://bugs.debian.org/245047>

Since the upgrade to 1.1.1-2 I've been unable to get my usual coffee intake.

This seems to be a new feature.. In previous versions I could launch OO, go for a coffee and when I came back it would have just finished loading.

Now I launch it, and presto-chango it appears on the screen

Why ? Why is someone putting effort into speeding up the loading time..

So this 'bug', is just to say \*thanks\* - and you can pass it upstream :-)

# Vielleicht kannst du ja etwas spenden?

- Projekte brauchen fast ständig Spiegel-Server und Bandbreite
- ... und Hardware
- ... und Geld nimmt man auch gerne an
- ... und natürlich ihre Zeit, ihre Ideen, ihre Hilfe

# Sag es allen weiter! Mach Werbung!

- Je mehr Leute freie Software benutzen, desto eher werden Fehler entdeckt, desto besser wird die Software, desto mehr werden bei der Entwicklung helfen und und und. . .
- Helf anderen ueber IRC / Mailinglisten / persönlich

# Hilf uns, besser zu werden!

- Entwickler haben einen anderen Blickwinkel auf ihre Software; sie wissen zu viel
- „normale“ Benutzer können zum Beispiel sinnvolle Hinweise zur Benutzerfreundlichkeit geben
- ... sagen, wenn sie etwas nervt
- ... sagen, wenn Programmfunktionen fehlen

*Dafür gibt es einfache Tools reportbug*

# Nerv ein paar Leute!

Aber nerv die richtigen!

Ein paar Beispiele:

- Hardware Hersteller
- Entwickler noch nicht portierter Software
- Entwickler unfreier Software
- Im Laden nach Software für Linux, Hardware Tauglichkeit und ähnlichem Fragen
- ... und warum man mit dem neuen Rechner W\*n\*o\*s mitkaufen muss, auch wenn man es nicht benutzt

*Erinner dich an das Gesetz von Angebot und Nachfrage*

# Sag uns Neuigkeiten!

- Viele Projekte haben (wöchentliche) Newsletter
- Oft auch in fremde Sprachen übersetzt und mit regionalen Infos erweitert

*Die Redakteure der Newsletter brauchen Input!*

# Sag uns Neuigkeiten!

- Viele Projekte haben (wöchentliche) Newsletter
- Oft auch in fremde Sprachen übersetzt und mit regionalen Infos erweitert

*Die Redakteure der Newsletter brauchen Input!*

# Mach dein Projekt für alle nutzbar!

- Die beste Software ist nutzlos, wenn die Benutzer sie nicht verstehen
- Die meiste Software ist bereits darauf vorbereitet, übersetzt zu werden  
Sogar in einige lokale Dialekte
- Meistenst muss man lediglich ein paar Zeichenkette in einer .po-Datei übersetzen
- Man kann seinen bevorzugten Text-Editor dafür benutzen, oder praktische Oberflächen wie zum Beispiel *kbabel*
- Im Regelfall gibt es Koordinations-Listen für Übersetzer (z.B. Einigung auf Übersetzungen für gebräuchliche Begriffe)
- gleiches gilt für längere Dokumentationen (Installationsanleitungen, Benutzerhandbücher, etc.)
- Wer sich das Übersetzen nicht zutraut, kann korrekturlesen
- Falsche oder fehlinterpretierbare Abschnitte bitte an der Author melden!

# Benutz deine Talente!

Es gibt Dinge, in denen sind Programmierer einfach nicht gut:

- Testen der Benutzerfreundlichkeit ihrer Programme
- Entwerfen von funktionierend, gut aussenden User-Interfaces
- Entwerfen von Banner, Artwork, DVD-labeln, etc.
- Gute Dokumentation
  - ▶ Verbessern und aktualisieren vorhandener Dokumentation
  - ▶ schreiben nicht vorhandener Dokumentation

# Melde Fehler!

Ein guter Fehlerbericht enthält die Antworten zu folgenden Fragen:

- Was hast du gemacht, als der Fehler auftrat?
- Was hast du erwartet, dass passieren würde?
- Was ist stattdessen passiert?
- Was könnte die Ursache des Problems sein?
- Kannst du den Fehler reproduzieren? Wie?
- Hast du noch irgend welche zusätzlichen Informationen?

*Dafuer gibt es Tools, reportbug, bug-buddy, u.ä.*

Vielen Dank für eure Aufmerksamkeit! Die Folien dieses Vortrags sind auch im Internet verfügbar:

<http://www.schmehl.info/publication/talks/payback/>

Bald wird dort auch eine ausführlichere Anleitung mit Beispielen verfügbar sein. Bitte haben sie ein wenig Geduld. . .

Offene Fragen? Zöger nicht, sie jetzt oder per E-Mail zu stellen:  
Alexander Schmehl <[alexander@schmehl.info](mailto:alexander@schmehl.info)>