

# **GPatch**

Ralf Gruner

Copyright © 1997, 1998 Ralf Gruner, Großschönau, Germany

---

<b>COLLABORATORS</b>
----------------------

	<i>TITLE :</i> GPatch		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Ralf Gruner	August 26, 2022	

<b>REVISION HISTORY</b>
-------------------------

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>GPatch</b>	<b>1</b>
1.1	GPatch Handbuch . . . . .	1
1.2	Vorwort . . . . .	2
1.3	GCompare . . . . .	2
1.4	GPatch . . . . .	3
1.5	Hinweise zur Benutzung . . . . .	3
1.6	Technische Details . . . . .	4
1.7	Beispiel . . . . .	5
1.8	Versionen für andere Betriebssysteme . . . . .	5
1.9	Copyright, Vertrieb und Haftungsausschluß . . . . .	5
1.10	Credits . . . . .	6
1.11	Stichwortverzeichnis . . . . .	6

---

# Chapter 1

## GPatch

### 1.1 GPatch Handbuch

\*\*\*RGR\*\*\*

GCompare / GPatch: Das Patch-System für Programm-Updates

Version 2.5

Autor: Ralf Gruner

Vorwort

GCompare

GPatch

Hinweise zur Benutzung

Technische Details

Beispiel

Versionen für andere Betriebssysteme

Copyright, Vertrieb und Haftungsausschluß

Danksagungen

Änderungsliste (nur englisch)

## 1.2 Vorwort

Vorwort

GCompare ist ein Programm zum Erzeugen von Patch-Dateien für den Vertrieb von Updates beliebiger Programme oder anderer Dateien.

Eine solche Patch-Datei enthält nur noch die Unterschiede zwischen den alten und den neuen Dateien, womit die zu vertreibenden Datenmengen erheblich reduziert werden.

GPatch dient dazu, diese Updates auszuführen.

Die Vorteile von GCompare/GPatch gegenüber vergleichbaren Programmen sind:

-Die Patch-Datei kann Änderungsdaten für eine beliebige Anzahl von Dateien enthalten. Dadurch ist es möglich, Updates für mehrere Versionen eines Programms zusammen zu vertreiben. Wenn sich die Verzeichnis-Struktur des Produktes nicht geändert hat, findet der Patcher selbständig die benötigten Änderungsdaten, ohne daß sich Ihr Skript darum kümmern muß. Außerdem können Sie die Patches für verschiedene Dateien in einer einzigen Patch-Datei unterbringen.

-Wenn die Patchdatei jeweils die Änderungen von einer Programmversion zur nächsten enthält, dann ist es nicht mehr notwendig, alle alten Versionen zu archivieren.

-Das Format der Patch-Datei ist bezüglich der Dateigröße aufwendig optimiert. GCompare kann mehrere verschiedene Kodierungsarten testen und diejenige auswählen, die das kürzeste Resultat liefert.

Obwohl ich nicht allzu viele ähnliche Programme kenne, möchte ich behaupten, daß GCompare in den meisten Fällen die kürzesten Patchdateien aller verfügbarer Patchprogramme erzeugt.

-Um zu verhindern, daß beim Anwender beschädigte Dateien ankommen, enthalten die Programme eine sehr zuverlässige Fehlererkennung (32-Bit-CRC-Signaturen für alle Dateien). Wenn keine Fehlermeldungen auftreten, dann können Sie sicher sein, daß die Ergebnisse des Patch-Vorgangs perfekt sind.

## 1.3 GCompare

GCompare

GCompare ist ein Shell-Befehl mit folgender Syntax:

```
GCompare <alte Datei> <neue Datei> <Patchdatei> [MODE <n> [VARY <n>] | AUTO [DEEP]] [VERBOSE]
```

GCompare vergleicht die alte Datei mit der neuen Datei und speichert alle Änderungen (Patches) in der Patchdatei.

Wenn GCompare eine bereits bestehende Patchdatei findet, dann hängt es die Änderungen an diese Datei an, nachdem es überprüft hat, daß für die zu bearbeitende alte Datei noch keine Änderungen in der Patchdatei enthalten sind.

Außerdem sucht es auch in der Patchdatei nach passenden Daten.

Aus der alten Datei und der Patchdatei kann GPatch dann die neue Datei erzeugen.

Bei Problemen gibt GCompare eine Fehlermeldung aus.

Schwerwiegende Fehler werden immer von einem Return-Code von 15 begleitet, und wenn das Ergebnis eine in irgendeiner Form beschädigte oder unvollständige Patchdatei ist, dann wird sie gelöscht.

Die Argumente MODE, VARY, AUTO und DEEP wählen die Betriebsart aus:

Ohne Angabe einer Betriebsart arbeitet GCompare in seiner schnellsten Betriebsart (der Quick-Modus der vorhergehenden Versionen von GCompare). Die sollten Sie aber nur benutzen, wenn es aus Zeitgründen nicht anders geht (für sehr große Dateien), oder wenn sich nur ein paar Bytes ihrer Dateien geändert haben, weil die Patchdatei sonst fast immer wesentlich größer ist als in den anderen Betriebsarten.

Mit dem Argument AUTO wählt GCompare selbständig das optimale Dateiformat aus. Das zusätzliche Argument DEEP bewirkt außerdem, daß die minimale Größe, ab der ein Datenbereich als passend gewertet wird, um -1 bis 1 verändert wird.

Mit AUTO DEEP erzeugt GCompare die kürzeste mögliche Patch-Datei.

---

Wenn Sie zusätzlich das Argument `VERBOSE` angeben, dann zeigt Ihnen `GCompare` nach dem Abschluß der Vergleichsfunktion die Einstellungen an, mit denen das beste Resultat erzielt worden ist.

Mit dem Argument `MODE` können Sie das Dateiformat direkt angeben. Die möglichen Werte für `<n>` nach `MODE` sind 1 bis 4.

Das Argument `VARY` verändert die internen Minimalwerte für die minimale Datenbereichsgröße, die in der `DEEP`-Betriebsart automatisch getestet werden. Der kleinste zulässige Wert ist -3.

Mit dem Argument `VERBOSE` zeigt `GCompare` in allen Betriebsarten außer `AUTO` eine ausführliche Liste der Patchdaten an.

`GCompare` braucht genug Arbeitsspeicher, um die alte, die neue und die Patchdatei gleichzeitig im Speicher halten zu können.

Die Suchgeschwindigkeit von `GCompare` hängt vom dann noch verfügbaren Speicher ab.

Der schnellste Suchalgorithmus benötigt insgesamt 9 mal die Dateigröße der alten Datei plus 514 KB.

Der einfache Listen-Suchalgorithmus benötigt insgesamt 9 mal die Dateigröße der alten Datei plus 2 KB.

Wenn `GCompare` auch diesen Speicher nicht findet, dann arbeitet es nur noch in der Betriebsart 3 mit linearer Suche.

Das gleiche gilt für die Suche in der möglicherweise vorhandenen Patchdatei.

Die schnellste Suche braucht 9 mal die Dateigröße der Patchdatei plus 514 KB und die einfache Suche 9 mal die Dateigröße plus 2 KB.

Wenn der Speicher auch dafür nicht ausreicht, dann wird nur in der alten Datei gesucht.

## 1.4 GPatch

GPatch

GPatch ist ein Shell-Befehl mit folgender Syntax:

```
GPatch <alte Datei> <Patchdatei> <neue Datei> [RECURSIVE] [NOVERSION | QUIET]
```

GPatch erzeugt aus der alten Datei und der Patchdatei die neue Datei.

Die passenden Patches wählt es anhand der Dateilänge und der CRC-Signatur der alten Datei aus.

Bei Problemen gibt GPatch eine Fehlermeldung aus.

Schwerwiegende Fehler werden immer von einem Return-Code von 15 begleitet, und wenn das Ergebnis eine in irgendeiner Form fehlerhafte neue Datei ist, dann wird sie gelöscht.

Das Argument `RECURSIVE` schaltet eine Betriebsart ein, in der GPatch, nachdem es erfolgreich eine neue Datei erzeugt hat, solange weitere Versuche zum Patchen unternimmt, bis keine passenden Daten mehr gefunden werden. Dabei wird immer die zuletzt erzeugte neue Datei als alte Datei benutzt.

Diese Betriebsart können Sie verwenden, wenn Sie nicht alle alten Versionen Ihrer Programme archivieren wollen.

Die Patchdatei sollte dann sinnvollerweise nicht die Änderungen aller alten Versionen auf die neueste Version enthalten, sondern jeweils die Änderungen von einer Version zur nächsten.

Mit `NOVERSION` läßt sich die Anzeige der Versionsdaten unterdrücken und mit `QUIET` arbeitet GPatch völlig ohne Textausgaben (Fehler werden natürlich immer angezeigt).

## 1.5 Hinweise zur Benutzung

Diskussion und Hinweise zur Benutzung:

Inzwischen habe ich einige Mails mit Kommentaren zur Geschwindigkeit und der Größe der resultierenden Patchdatei erhalten.

Manche fanden die Patchdatei viel kleiner als die anderer Patcher, andere wiederum nicht, und fast allen ist das Compare-Programm nicht schnell genug.

Das Thema Geschwindigkeit sollte sich mit Version 2.5 erledigt haben.

---

Dennoch ist bei größeren Dateien noch eine Steigerung möglich. Dazu würde GCompare einen Arbeitsspeicher von reichlich 128 MB brauchen. Wenn das irgendwann mal eine Selbstverständlichkeit ist, dann kann ich das Programm entsprechend erweitern.

Da es nicht möglich ist, ein Dateiformat für die Patchdatei zu entwickeln, das in allen Fällen das kürzeste Ergebnis liefert (die Größe der Datei hängt vom Abstand der passenden Daten ab und der Art und Weise, diesen Abstand zu adressieren), hat GCompare vier verschiedene Algorithmen.

Der Optimierer, der mit AUTO aktiviert wird, kann das beste Format auswählen, was aber etwa die 4-fache Rechenzeit braucht.

Die Verbesserungen, die DEEP zusätzlich bewirkt, hängen von den verglichenen Dateien ab. In vielen Fällen bewirkt DEEP keine Verbesserung mehr.

Wenn Ihnen der Optimierer zu lange rechnet, dann empfehle ich MODE 1. In den meisten Fällen liefert das ein gutes Ergebnis.

Wenn Sie die Ergebnisse von GCompare mit denen anderer Programme vergleichen wollen, sollten Sie das auch mit komprimierten Versionen der Patchdateien tun. GCompare besitzt keine eingebaute Kompression (im Gegensatz zu anderen Vergleichsprogrammen mit Lauflängenkompensation), weil ich denke, daß alle Vertriebsdateien vor ihrer Veröffentlichung ohnehin komprimiert werden und eine doppelte Kompression die Effektivität wieder verringert.

GCompare sucht passende Daten nicht nur in der alten Datei, sondern gegebenenfalls auch in der bereits vorhandenen Patchdatei. Die resultierende Patchdatei wird also verglichen mit einzelnen Patchdateien um so effektiver, je mehr Patches verschiedener Versionen des gleichen Programms sie enthält.

Wenn sich bei diesen Versionen immer nur wenig geändert hat, dann sollten Sie in der Patchdatei die Änderungen von einer Version zur nächsten speichern und GPatch im rekursiven Modus arbeiten lassen. So wird die Patchdatei besonders klein.

Um GPatch nicht zu groß werden zu lassen, ist das Bearbeiten der älteren Dateiformate nicht mehr vorgesehen. Damit Sie bei Bedarf auch diese Patchdateien benutzen können, sind die Versionen 1.4 und 1.6 ebenfalls noch im Lieferumfang.

Wenn GPatch eine alte Patchdatei findet, dann teilt es Ihnen mit, welche Version Sie dafür brauchen.

## 1.6 Technische Details

Technische Details

MODE 1

Modus 1 ist am effektivsten, wenn die meisten Änderungen der Dateien kleiner als 128 Byte sind.

MODE 2

Modus 2 ist bei vielen sehr kleinen Änderungen am effektivsten.

MODE 3

Modus 3 findet passende Daten nur in einem Relativbereich von 64 KB.

Dafür wird aber nur eine lineare Suche gebraucht, was besonders bei Speichermangel nützlich ist.

MODE 4

Modus 4 ähnelt Modus 1, wobei die Vorteile bei Dateien mit weniger aber größeren Änderungen liegen.

Schneller Modus (keine Option angegeben)

Der schnelle Modus produziert dasselbe Dateiformat wie Modus 3, es wird aber nur in einem Relativbereich von 254 Byte nach passenden Daten gesucht.

-----

Nachdem GCompare fertig ist, gibt es noch eine Liste der Patchdaten aus. Die angezeigten Werte sind alles, was in der Patchdatei über Ihre Dateien gespeichert ist.

Die Angabe "type" ist das Kodierungsverfahren, das mit MODE ausgewählt worden ist. Falls Sie außerdem wissen wollen, welche Änderung der minimalen Blockgröße GCompare im Modus AUTO DEEP benutzt hat, dann müssen Sie zusätzlich VERBOSE angeben.

Um die Änderungsdaten verschiedener Programme zu unterscheiden, benutzt GPatch die Dateigröße und die CRC-Signatur. Versionsangaben oder Dateinamen werden nicht berücksichtigt. Dieses Verfahren ist vollkommen sicher, da GCompare beim Erzeugen der Patchdatei auf dieselbe Weise überprüft, ob Änderungsdaten für eine bestimmte Datei bereits vorhanden sind, bevor es neue Daten anfügt.

## 1.7 Beispiel

### Beispiel

Als Beispiel für die Anwendung der Patchprogramme habe ich in der Schublade "ExampleScripts" Skripte für das Erzeugen und Anwenden einer Patch-Datei eines fiktiven Programms beigelegt.

MakePatchFile erzeugt die Patch-Datei und UpdateMyProgram führt das Patchen aus.

In dem Beispiel wird davon ausgegangen, daß verschiedene ältere Versionen des Programms "MyProgram" in einem Archiv namens "Archive" liegen. Der Patch für das Programm und eine Anleitung "MyProgram.readme" wird in diesem Beispiel direkt auf der Programmdiskette "MyProgram" ausgeführt.

Und zum schnellen Experimentieren gibt es noch das CompareScript, mit dem Sie GCompare benutzen können, ohne Shell-Befehle eintippen zu müssen.

Von Thomas Baust habe ich ein Beispiel-Skript für den Installer erhalten und im wesentlichen unverändert unter dem Namen UpdateMyProgram.installer mit beigelegt. Es führt einen Patch für ein Programm namens "MyProgram" in einem vom Anwender wählbaren Verzeichnis aus.

## 1.8 Versionen für andere Betriebssysteme

### Versionen für andere Betriebssysteme

Im Verzeichnis bin/IX befinden sich Versionen von GCompare und GPatch für einige andere Computersysteme. Die benutzte Compilerversion und das Betriebssystem können Sie dem Text Machines.readme entnehmen.

Diese Programme sind mit Ausnahme der Sparc-Version nicht ausführlich getestet, da mir die Hardware nicht zur Verfügung steht.

Wenn jemand damit Erfolge (oder Mißerfolge) erzielt, dann bitte eine E-Mail an [mich](#).

## 1.9 Copyright, Vertrieb und Haftungsausschluß

### Copyright

Das Copyright © 1997, 1998 von GCompare und GPatch liegt bei

Ralf Gruner, An der Sense 5a, D-02779 Großschönau

[ralf.gruner@t-online.de](mailto:ralf.gruner@t-online.de)

### Vertrieb

GCompare und GPatch für Amiga sind Freeware. Sie können beliebig eingesetzt werden, einschließlich der Anwendung für kommerzielle Programme.

Wenn Sie aber Wert auf zukünftige Updates legen, dann senden Sie mir bitte eine E-Mail, damit ich weiß, daß es überhaupt Benutzer der Programme gibt. Es gibt immer noch einiges zu verbessern, aber die Zeit dafür nehme ich mir nur, wenn es auch jemand braucht.

Die Versionen für andere Betriebssysteme sind nur für nichtkommerzielle Anwendungen Freeware. Für jeglichen kommerziellen Einsatz setzen Sie sich bitte mit mir in Verbindung.

Und (wer's lesen will):

### Rechtliches

Bei der Entwicklung der Software wurde mit allergrößter Sorgfalt vorgegangen. Trotzdem sind Fehler nicht vollständig ausgeschlossen. Der Autor übernimmt keine Haftung für Schäden, die direkt oder indirekt auf die Benutzung seiner Programme zurückzuführen sind.

---

## 1.10 Credits

Danksagungen

an Dirk Stöcker für seine ausführliche Fehlersuche, die Idee der rekursiven Betriebsart von GPatch und die Portierungen auf die diversen IX-Betriebssysteme

und an alle, die sich bei **mir** gemeldet haben, entweder um mitzuteilen, daß sie GCompare benutzen oder um Testberichte zu senden:

Thomas Baust

Christian Beck

Oliver Blumert

Domenic Gebauer

John Girvin

Matthew Gregan

Bernardo Innocenti

Richard Koerber

Michael Lünse

Tom Newton

Jürgen Reinert

Tobias Schaechtelin

Maik Schreiber

## 1.11 Stichwortverzeichnis

Stichwortverzeichnis A

Änderungsliste

B

**Beispiel**

C

**Copyright**

G

**GCompare**

**GPatch**

H

**Hinweise**

M

**mode**

U

**UNIX-Versionen**