



# Server Performance

*Ralf Martin*

28.11.2003



# Inhalt

---

- Warum Perforce bei Harman/Becker?
- Allgemeines zur Performance
- Erfahrungen bei Harman/Becker (1)
  - Das erste Jahr mit Perforce
  - Der Weg aus der Performance Krise
- Befehlsbearbeitung im Server
  - Locking der Tabellen
- Erfahrungen bei Harman/Becker (2)
  - Analyse mit p4gla
  - Geschwindigkeit erhöhen

# Übersicht

---

- Warum Perforce bei Harman/Becker?
- Allgemeines zur Performance
- Erfahrungen bei Harman/Becker (1)
  - Das erste Jahr mit Perforce
  - Der Weg aus der Performance Krise
- Befehlsbearbeitung im Server
  - Locking der Tabellen
- Erfahrungen bei Harman/Becker (2)
  - Analyse mit p4gla
  - Geschwindigkeit erhöhen

# Warum Perforce bei H/B

---

- Warum hat sich Harman/Becker für Perforce entschieden?
  - Hohe Stabilität/Sicherheit
  - Hohe Performance
  - Großer Funktionsumfang
    - Atomare Operationen/Changelists
    - Interfile Branching
    - ...

# Systemübersicht

---

- User: ca. 600
- Serveranzahl: 4
  - 1 Hauptserver
    - OS: HP-UX 11
    - CPU#: 2
    - RAM: 2GB
    - HD: Disk Array mit 41 Disks
  - 3 kleinere Server mit Linux
  - 5+ Proxies

# Übersicht

---

- Warum Perforce bei Harman/Becker?
- Allgemeines zur Performance
- Erfahrungen bei Harman/Becker (1)
  - Das erste Jahr mit Perforce
  - Der Weg aus der Performance Krise
- Befehlsbearbeitung im Server
  - Locking der Tabellen
- Erfahrungen bei Harman/Becker (2)
  - Analyse mit p4gla
  - Geschwindigkeit erhöhen

# Performance

---

- Jedes System hat eine physikalische Grenze
- Mehrere Bereiche beeinflussen die Performance
  - a) Serverleistung (CPU/RAM)
  - b) Festplattenleistung (Controller/Dateisystem)
  - c) Netzwerk (LAN/WAN)
  - d) Revision des Perforce Servers
  - e) Einstellungen des Clients
  - f) User Verhalten

## a) Serverleistung

---

- CPU
  - Perforce ist mehrprozessorfähig
  - Die CPU Anforderungen von Perforce sind aber gewöhnlich gering
- RAM
  - Der Server sollte nie in den *swap* Bereich gelangen



## b) Festplattenleistung

- Die Performance des Servers wird hauptsächlich durch die I/O Bandbreite limitiert
- Der eingesetzte RAID Level kann erheblichen Einfluss auf die Festplattenleistung haben
  - Das hängt aber auch von der verwendeten Hardware (Controller) ab → intensiv testen
- Das verwendete Dateisystem sollte nach Verfügbarkeit, Sicherheit, Geschwindigkeit gewählt werden
  - XFS ist bei UNIX eine gute Wahl

## c) Netzwerk

---

- LAN
  - nicht unnötigerweise Kompression einstellen
- WAN
  - Kompression benutzen
  - Proxy (P4P) benutzen
- Vielleicht einen zusätzlichen Server, aber
  - Remote Depots zeigen nicht die Metadaten des anderen Servers
  - Common Source ist nicht möglich (read only)

## d) Revision des Perforce Servers (1)

- Ab revision 2002.1 hat jeder Client ein mapState Attribut bekommen
  - Vorher war Index auf *db.have* in Client-Syntax
  - mit dem mapState ist nun ein Index auf Depot-Syntax vorhanden
  - Darauf achten, dass mapState den Wert 1 hat
    - Check dafür sollte automatisch durchgeführt werden
    - z.B. JAVA Programm, das Usern mit mapState 0 eine Email schickt (User entdecken dann manchmal überflüssige Clientspecs)

## d) Revision des Perforce Servers (2)

---

- mapState Beispiel von Michael Shields  
Perforce User Conference 2003 Las Vegas:
  - Befehl: user-fstat für eine Datei
    - mapState 0
      - Ausführungszeit: 6s
      - Userzeit: 4420ms; Systemzeit: 430ms
    - mapState 1
      - Ausführungszeit: 0s
      - Userzeit: 10ms; Systemzeit: 0ms

## e) Einstellungen des Clients

---

- Kompression korrekt einstellen
- Auf Korrekten mapState achten
  - Bei Änderungen in der Client View bietet P4Win ab Version 2002.2 einen Dialog an
    - Sync to head
    - Sync to have
    - Don't sync
  - Sync to have ist eine gute Wahl, wenn keine Änderung im Workspace erfolgen soll
- Ausgabe der P4 Befehle einschalten

## f) User Verhalten (1)

---

- Durch die Abstraktion von P4Win sind dem User nicht alle potentiellen Probleme bekannt
  - Manche Befehle haben als Parameter
    - `<command> //...`
  - Das hat zur Folge, dass **alle** Depots gescanned werden
    - Das ist besonders schlecht bei Remote Depots.  
Daher nur ausgewählten Usern read Rechte geben
    - Da auch andere Kommandos mehr Zeit für das Scannen brauchen ist es ratsam sehr sorgfältig vorzugehen

## f) User Verhalten (2)

- Typische Kommandos für //... werden i.d.R. dann ausgeführt, wenn kein Pfadkontext vorhanden ist
- Typischerweise mit einem Rechtsklick auf Inhalte des rechten Views in P4Win
  - Rechtsklick auf eine submitted Changelist
    - Sync to Changelist → Besser Rechtsklick auf Folder
  - Rechtsklick auf ein Label
    - Sync files → Besser Rechtsklick auf Folder
  - etc. (*//... Kommandos über Anzeige der Perforce Commands identifizieren*)

## f) User Verhalten (3)

---

- Revert großer Pending Changelists in P4Win
  - P4Win prüft bei einem Revert zunächst jede Datei, ob sich etwas geändert hat und warnt dann
  - Wenn feststeht, dass diese Pending Changelist ohne Nachfragen reverted werden soll, bietet sich die Konsole an
    - `p4 revert -c <changelist> <Pfad>/...`
- Große Pending Changelists zügig bearbeiten
  - P4Win erfragt häufig den Status offener Dateien



# Übersicht

---

- Warum Perforce bei Harman/Becker?
- Allgemeines zur Performance
- Erfahrungen bei Harman/Becker (1)
  - Das erste Jahr mit Perforce
  - Der Weg aus der Performance Krise
- Befehlsbearbeitung im Server
  - Locking der Tabellen
- Erfahrungen bei Harman/Becker (2)
  - Analyse mit p4gla
  - Geschwindigkeit erhöhen

# Erfahrungen bei Harman Becker

## Das erste Jahr mit Perforce (1)

---

- Import von MKS
  - Nicht über rcs conversion script, da branching völlig verschieden behandelt wird
  - Stattdessen über ausgesuchte Baselines (Labels), die sukzessiv übernommen wurden
  - Dadurch ein rasanter Anstieg der Filerevisions, die für Perforce aber kein Problem darstellen

# Erfahrungen bei Harman Becker

## Das erste Jahr mit Perforce (2)

- Die tägliche Arbeit und der mit Perforce überarbeitete Entwicklungsprozess forderte ein Umdenken
- Die Struktur bei gemeinsamen Komponenten musste noch gefunden werden
- Das Anlegen von Entwicklungscodelinien und die Integration führt zu einem rasanten Anstieg von Changelists
  - Wir haben innerhalb eines Jahres 100.000 submitted Changelists erreicht
- **Dann war die Performance plötzlich schlecht!**

# Erfahrungen bei Harman Becker

## Der Weg aus der Performance Krise (1)

---

- Die Suche nach dem Bottleneck (1)
  - Serverleistung: ok
  - Festplattenleistung: ok
  - Netzwerk: ok
  - Revision des Servers: ok (letzte Version)
  - Einstellungen des Clients:
    - Clients hatten überwiegend einfache Views
    - Manchmal war die Kompression über das WAN nicht eingeschaltet

# Erfahrungen bei Harman Becker

## Der Weg aus der Performance Krise (2)

---

- Bis dahin war keine nennenswerte Verbesserung zu erkennen
  - Userverhalten musste überprüft werden
- Die ersten Schritte zur Analyse
  - Support (Tony Smith) kontaktieren
  - Logfile (p4error) analysieren
  - Auffällige Userbefehle hinterfragen

# Erfahrungen bei Harman Becker

## Der Weg aus der Performance Krise (3)

- Die Suche nach dem Bottleneck (2)
  - Userverhalten:
    - Einige Befehle haben bis zu einer Stunde gedauert
    - Die Analyse der Parameter zeigte ein //...
      - → Durch Aufklärung der Ursache war auch der User erfreut, da seine Anfragen schneller wurden
    - Da alle User Leserechte auf alle Depots hatten, waren auch die Remote Depots eingeschlossen
      - → Das wurde durch Protections ausgeschlossen
- *Alle Maßnahmen brachten aber nicht den erhofften Durchbruch!*

# Erfahrungen bei Harman Becker

## Der Weg aus der Performance Krise (4)

---

- Tools zur Analyse des Server Logfiles
  - p4loga (Autor: Michael D. Shields, OS: jedes)
    - Zeigt die Maxima des Logfiles an
      - Maximum number of concurrent processes
      - Global Compute Phase Maximums
      - Global Elapsed Maximums
      - Elapsed Maximums per Function
  - p4gla (Autor: Ralf Martin, OS: Windows)
    - Visualisiert das Logfile grafisch

## Erfahrungen bei Harman Becker

### Der Weg aus der Performance Krise (5)

---

- Die ersten Probleme im Userverhalten konnten mit p4loga sehr gut analysiert werden
- Des Öfteren wurde aber ein Zeitpunkt gemeldet, an dem der Server nicht mehr reagierte
  - Das Logfile anzuschauen war extrem mühselig
  - Daher wurde p4gla entwickelt
  - Eine Analyse der beteiligten Befehle/Zeiten/User war nun möglich



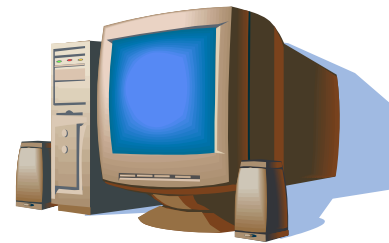
# Übersicht

---

- Warum Perforce bei Harman/Becker?
- Allgemeines zur Performance
- Erfahrungen bei Harman/Becker (1)
  - Das erste Jahr mit Perforce
  - Der Weg aus der Performance Krise
- **Befehlsbearbeitung im Server**
  - **Locking der Tabellen**
- Erfahrungen bei Harman/Becker (2)
  - Analyse mit p4gla
  - Geschwindigkeit erhöhen

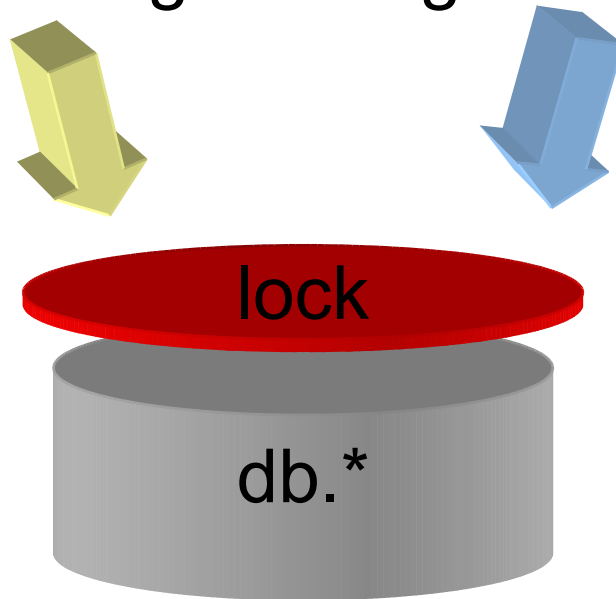
# Befehlsbearbeitung im Server (1)

- Ablauf aus Sicht des Servers
  - Befehl entgegen nehmen
  - Befehl ausführen
  - Ergebnis zusammenstellen
  - Ergebnis an Client senden



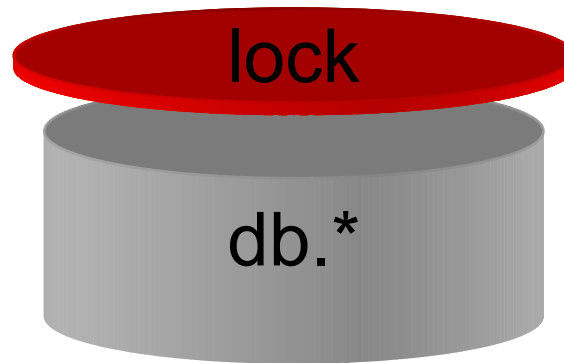
# Befehlsbearbeitung im Server (2)

Befehl entgegennehmen  
Befehlsanalyse  
Ergebnis an Client senden



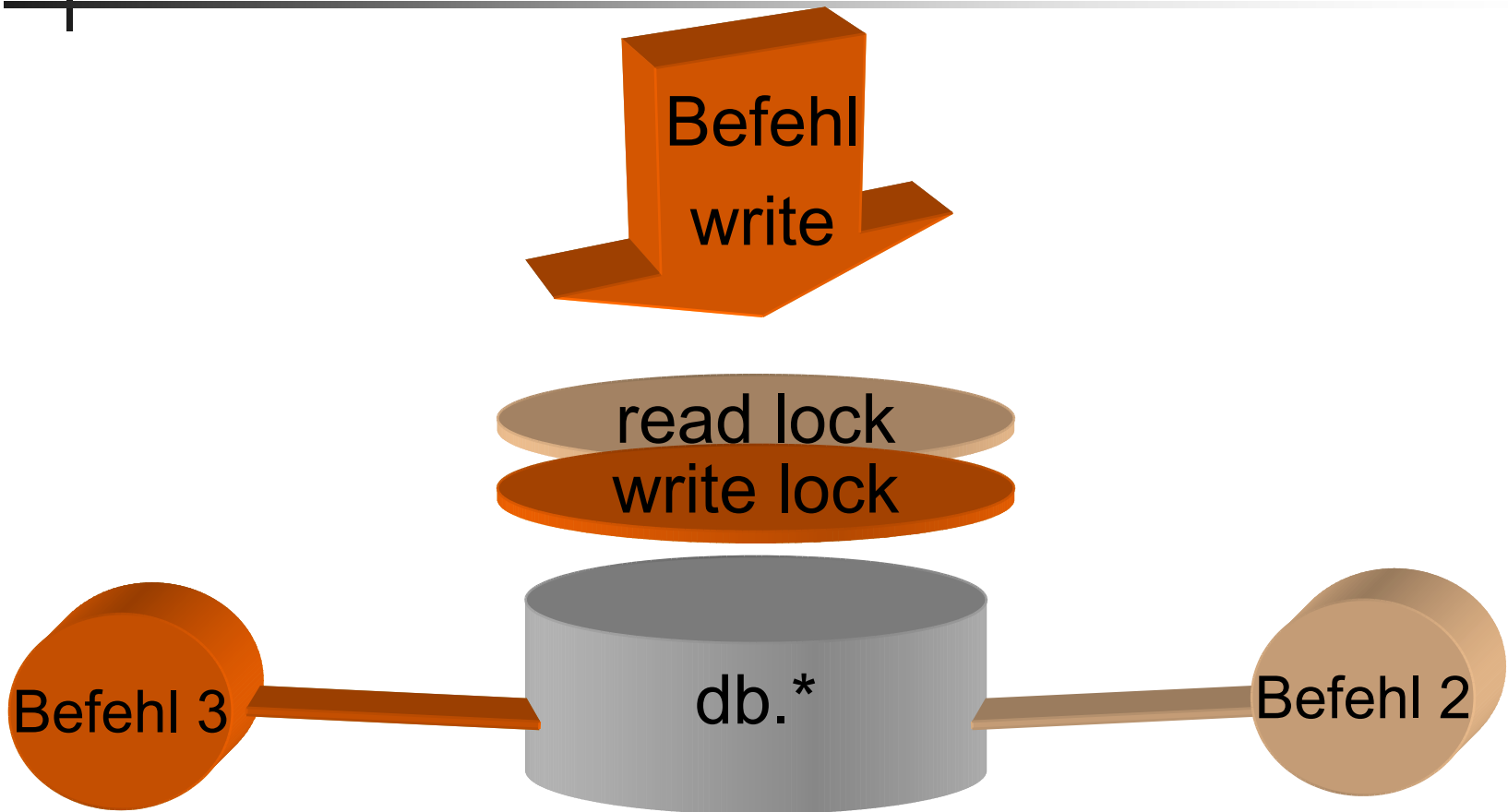
# Locking der Tabellen (1)

---



- Lock der db.\* Dateien muss unterschieden werden
  - read lock
  - write lock

# Locking der Tabellen (2)



## Locking der Tabellen (3)

---

- Zusammenfassung
  - Ein Lesezugriff wird nicht durch ein Read Lock geblockt
  - Ein Schreibzugriff wird durch einen Read Lock geblockt
  - Ein Lesezugriff wird durch ein Write Lock geblockt

# Locking der Tabellen (4)

---

- Erkenntnis
  - Ein langsamer Lesezugriff bewirkt, dass ein Schreibzugriff nicht erfolgen kann
  - Dieser Schreibzugriff wird in eine Queue gestellt
  - Ein weiterer Lesezugriff wird damit auch verhindert!

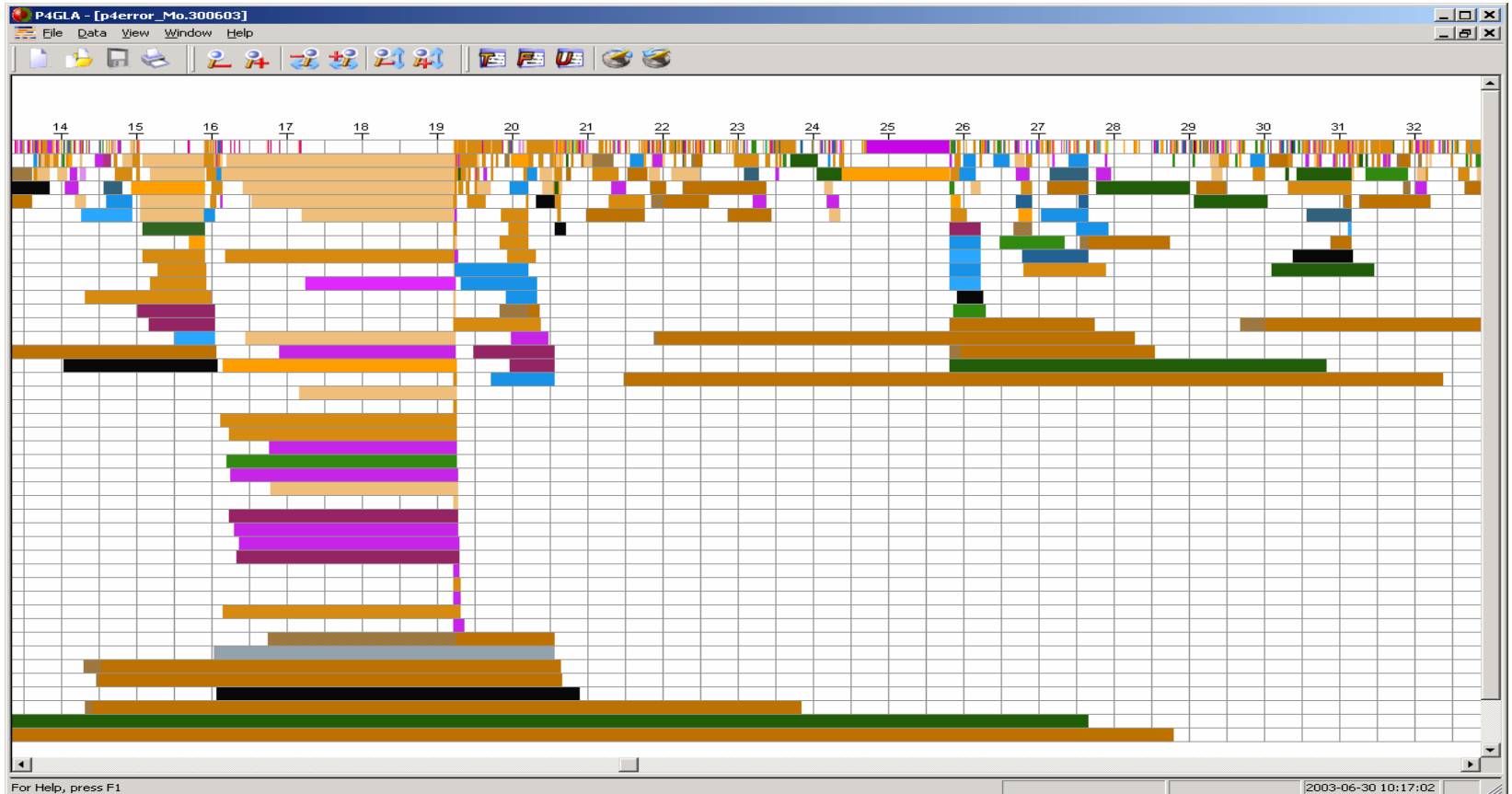
# Übersicht

---

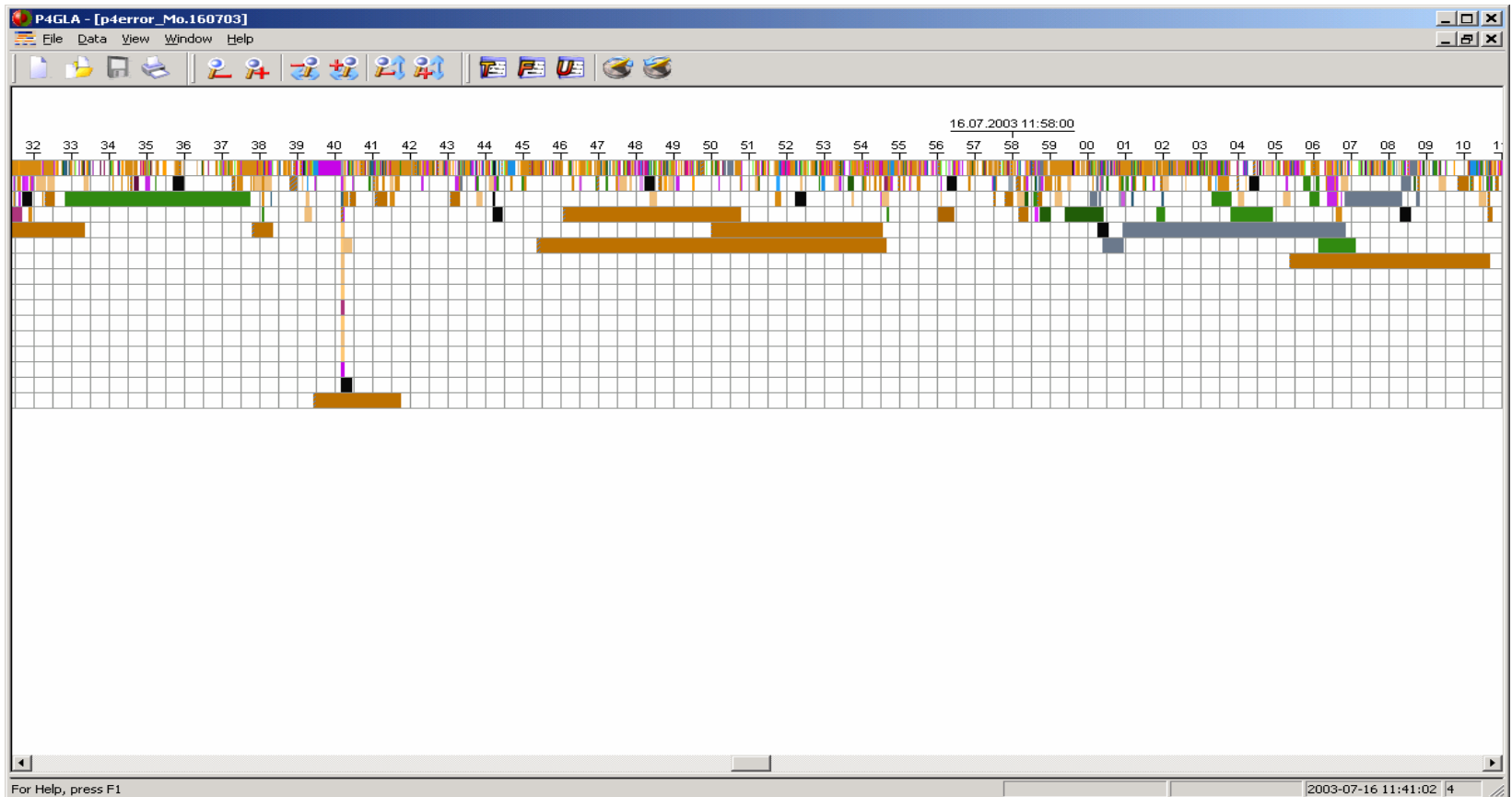
- Warum Perforce bei Harman/Becker?
- Allgemeines zur Performance
- Erfahrungen bei Harman/Becker (1)
  - Das erste Jahr mit Perforce
  - Der Weg aus der Performance Krise
- Befehlsbearbeitung im Server
  - Locking der Tabellen
- Erfahrungen bei Harman/Becker (2)
  - **Analyse mit p4gla**
  - Geschwindigkeit erhöhen



# Analyse mit p4gla (1)



# Analyse mit p4gla (2)



# Übersicht

---

- Warum Perforce bei Harman/Becker?
- Allgemeines zur Performance
- Erfahrungen bei Harman/Becker (1)
  - Das erste Jahr mit Perforce
  - Der Weg aus der Performance Krise
- Befehlsbearbeitung im Server
  - Locking der Tabellen
- Erfahrungen bei Harman/Becker (2)
  - Analyse mit p4gla
  - **Geschwindigkeit erhöhen**

# Geschwindigkeit erhöhen

- Ergebnis aus der Analyse
  - Vermeide auf jeden Fall langsame Zugriffe!
  
- Wie haben wir das erreicht?
  - db.\* werden als btrees gespeichert
  - btrees im fragmented Zustand sind ineffizient
  - Durch Reorganisieren der db.\* Dateien werden die btrees neu aufgebaut
  - → Verwerfen der db.\* und Zurücklesen des Journals