

## Praktikum Spezifikation und Verifikation

### 1 Formalisierung des Unix File Systems

In dieser Aufgabe geht es darum als „realistischere“ Anwendung eine Formalisierung des Unix File Systems zu entwickeln.

Eine Beschreibung einer konkreten Implementierung des Unix File Systems finden Sie unter

<http://cm.bell-labs.com/cm/cs/who/dmr/cacm.html>.

Eine Formalisierung des File Systems, in der einige Vereinfachungen gegenüber der Realität vorgenommen wurden, finden Sie unter

<http://isabelle.informatik.tu-muenchen.de/library/HOL/Unix/document.pdf>

Das Ziel dieser Aufgabe ist es, diese Formalisierung um einige der Auslassungen zu ergänzen.

#### 1.1 Erweiterung der Formalisierung

Arbeiten Sie sich in die gegebene Formalisierung ein und überlegen Sie sich anhand der Beschreibung des File Systems eine sinnvolle Erweiterung der Formalisierung.

Nachfolgend finden Sie einige Erweiterungsmöglichkeiten der bisherigen Formalisierung, die Ihnen als Anregung dienen können.

**Berücksichtigung des Executable-Rechts bei Verzeichnissen:** In der bisherigen Formalisierung wurde die Vereinfachung getroffen, daß für alle Verzeichnisse das Executable-Recht gesetzt ist. In einer realistischeren Modellierung müsste beim Zugriff auf einen Pfad zusätzlich für jedes Präfix des Pfades geprüft werden, ob das Executable-Recht gesetzt ist. Dazu müsste die Funktion `access` entsprechend angepaßt werden.

**Erweiterung der File-Attribute** In der bisherigen Formalisierung haben Files nur die Attribute *owner* und *others*. Man könnte beispielsweise die Attribute eines Files um Gruppenzugehörigkeit erweitern und Rechte für *owner* und *group* einführen.

Entsprechend könnte man dann neue Operationen *chown* und *chgrp* hinzufügen.

**Symbolische Links** Links wurden in der bisherigen Formalisierung ganz ausgelassen. Zur Einführung symbolischer Links müsste der Typ *file* so verändert werden, daß Files entweder Text-Files (bestehend aus Attributen und dem Inhalt) oder symbolische Links (bestehend aus einem Pfad) sind.

Ein Problem ergibt sich hier durch mögliche zyklische Abhängigkeiten, die zur Nicht-Terminierung des Zugriffs (durch die Funktion *access*) führen können. In so einem Fall könnte die Funktion *access* einfach nach einer bestimmten Anzahl von rekursiven Aufrufen (also von Dereferenzierungen symbolischer Links) abbrechen. Das ist auch das Verhalten einer Unix Implementierung, die dann etwa die folgende Meldung ausgibt: *Number of symbolic links encountered during path name traversal exceeds MAXSYMthm sysmLINKS*

Entsprechend könnte man dann Operationen *symlink* und *readlink* zum Erzeugen bzw. Lesen eines symbolischen Links hinzufügen.

**Hard Links** Das File System ist in der bisherigen Formalisierung gegeben durch den Verzeichnisbaum *root*. Um Hard Links einführen zu können, könnte man eine Abbildung von File-Identifikatoren (*fid*) auf Files einführen und das File System um diese Abbildung erweitern. Entsprechend müßte dann die Übergangsrelation *transition* angepaßt werden. Im Gegensatz zu den symbolischen Links können hier keine Zyklen auftreten, da keine Verzeichnisse gelinkt werden können.

Entsprechend könnte man eine Operationen *link* zum Anlegen eines Hard Links hinzufügen.

**etc ...**

Erweitern Sie die Spezifikation und beweisen Sie einfache Simulationen (siehe Abschnitt 4.2), um sich von der Korrektheit der Spezifikation zu überzeugen. Die genannten Punkte sollten dabei nur als Anregung verstanden werden, es können auch andere Erweiterungen vorgenommen werden.

## 1.2 Beweis von Invarianz-Eigenschaften

Formulieren und beweisen die folgende Invarianz-Eigenschaft des File Systems:

*Sind an einer Folge von Operationen weder der **owner** eines Files, noch der **super user**, noch irgendein Benutzer, der auf dem File Schreibrechte hat, beteiligt, dann ändert sich der Inhalt des Files nicht.*

Es ist dabei Ihnen überlassen, ob sie Ihre erweiterte, oder die ursprüngliche Formalisierung verwenden.

Hinweise:

- Achten Sie beim Beweis auf ein Top-Down-Vorgehen, d.h. formulieren Sie einfache, offensichtliche Eigenschaften als (zunächst) unbewiesene Lemmata.
- Gegebenenfalls müssen Sie die Aussage noch präzisieren, indem Sie zum Beispiel noch weitere Annahmen an das System hinzunehmen.

▷ **Abgabe Erweiterung der Formalisierung: 4. Juli 2001**

▷ **Abgabe Beweis der Invarianz-Eigenschaft: 11. Juli 2001**

▷ **Abschließende Präsentation: 18. Juli 2001**

Bitte beachten Sie auch die Hinweise zur Präsentation auf unserer Homepage.