

Semesterarbeit

für

Sebastian Ryffel

Betreuer: Lukas Ruf

Ausgabe: 19.03.2007Abgabe: 07.07.2007

Topsy v3 on Multi-Core Processors

1 Einführung

Chip-Multicore Prozessoren kommen heutzutage bereits in handelsüblichen Arbeitsplatzrechner zum Einsatz. Beispiele sind der Dual-Core Intel Xeon oder der Quad-Core AMD Opteron Prozessor.

Topsy v3 [3] – a Teachable Operating System [1] – wurde an der ETH Zürich entwickelt und wird an verschiedenen Universitäten und Fachhochschulen im Unterricht eingesetzt. Topsy v3 folgt einer Schichtenarchitektur und verfolgt ein sehr ausgeprägtes Mikrokern-Konzept, das sog. Protection Domains in den beiden Dimensionen *Privilegien* und *Ausführungspriorität* priorisieren kann.

Während Topsy v3 bereits schon für verschiedene Prozessorplattformen verfügbar ist (u.a. für die Prozessoren Intel XScale und MIPS R3000), existiert für ia32-kompatible Prozessoren mit Topsy386[4] nur eine Topsy v1-Architektur-kompatible Portierung.

2 Aufgaben: Topsy v3 on Multi-Core Processors

In dieser Semesterarbeit von Sebastian Ryffel soll Topsy v3 auf eine moderne ia32-kompatible Chip-Multicore Prozessorplattform portiert werden. Diese Portierung umfasst die Anpassungen des Hardware Abstraktionsschicht (HAL, Hardware Abstraction Layer) und, wo benötigt, die Implementierung und Anpassung geeigneter Mechanismen in der portablen Kernelschicht von Topsy v3.

3 Vorgehen

Die Portierung von Topsy v3 erfolgt in drei Phasen. In der ersten Phase wird der HAL Topsy386 auf einen binär kompatiblen ia32-Prozessor-Simulator und daran anschliessend auf die Ziel Hardware portiert. Diese Portierung gestattet das Anpassen der Basisfunktionalität, die für Topsy v3 ebenfalls benötigt wird, und erlaubt eine Einarbeitung in den Source Code und die Prozessorarchitektur.

In der zweiten Phase werden die Modifikationen für Topsy v3 am HAL und am Kernel in den dann existierenden Code eingearbeitet, so dass Topsy v3 für ia32-kompatible Plattformen zur Verfügung steht. Die dritte Phase beinhaltet die Umsetzung der benötigten Anpassungen, um Chip-Multicore-Eigenschaften soweit als möglich auszunutzen.

Zur Erreichung dieser Ziele wird die Arbeit nach folgendem Vorgehen durchgeführt:

- Richten Sie sich eine Entwicklungsumgebung (GNU Tools) unter Linux ein.
- Machen Sie sich vertraut mit den Unterlagen (Dokumentation und Source Code) zum bestehenden Topsy386 und Topsy v3, gleich wie zur ia32-Prozessorplattform.
- Erstellen Sie einen Zeitplan, in welchem Sie die von Ihnen zu erreichenden Meilensteine Ihrer Arbeit identifizieren.
- Analysieren Sie verschiedene Komponenten von Topsy386, um diejenigen zu identifizieren, die für eine moderne ia32-Prozessorplattform angepasst werden müssen.
- Identifizieren Sie die anzupassenden Funktionen und Parameter in Topsy v3, die für die Portierung angepasst werden müssen.
- Implementieren Sie diese für eine moderne ia32-kompatible Chip-Multicore Plattform.
- Beachten Sie insbesondere die Problematik des Startvorgangs und der Initialisierung der ia32-kompatiblen Prozessoren.

optional Entwickeln Sie Mechanismen, um spezifische Eigenschaften einer Chip-Multicore Architektur unter Topsy v3 zu kontrollieren und auszunutzen.

- Zeigen Sie die erfolgreiche Implementierung auf einem Chip-Multicore Prozessor durch eine Ausführung der Demo-Applikationen unter Topsy v3.
- Evaluieren Sie das Erreichte.
- Dokumentieren Sie das Erreichte.

Auf eine klare und ausführliche Dokumentation wird besonders Wert gelegt. Es wird empfohlen, diese laufend nachzuführen und insbesondere die entwickelten Konzepte und untersuchten Varianten vor dem definitiven Variantenentscheid ausführlich schriftlich festzuhalten.

4 Organisatorische Hinweise

- Am Ende der zweiten Woche ist ein Zeitplan für den Ablauf der Arbeit vorzulegen und mit dem Betreuer abzustimmen.
- Mit dem Betreuer sind regelmässige, zumindest wöchentliche Sitzungen zu vereinbaren. In diesen Sitzungen sollen die Studenten mündlich über den Fortgang der Arbeit und die Einhaltung des Zeitplanes berichten und anstehende Probleme diskutieren. Die Sitzungen können mittels geeigneten Telekommunikationsmitteln abgehalten werden, falls möglich.
- Am Ende des ersten Monates muss eine Vorabversion des Inhaltsverzeichnis zur Dokumentation dem Betreuer abgegeben und mit diesem besprochen werden.
- Nach der Hälfte der Arbeitsdauer soll ein kurzer mündlicher Zwischenbericht abgegeben werden, der über den Stand der Arbeit Auskunft gibt. Dieser Zwischenbericht besteht aus einer **viertelstündigen**, mündlichen Darlegung der bisherigen Schritte und des weiteren Vorgehens gegenüber Professor Plattner.
- Am Schluss der Arbeit muss eine Präsentation von **20 Minuten** im Fachgruppen- oder Institutsrahmen gegeben werden. Anschliessend an die Schlusspräsentation soll die Arbeit Interessierten praktisch vorgeführt werden.

- Ein einheitlicher Coding Style muss eingehalten werden.
- Bereits vorhandene Software kann übernommen und gegebenenfalls angepasst werden; Quellen müssen korrekt zitiert werden.
- Die Dokumentation muss mit dem Satzsystem \LaTeX Graphiken müssen mit Open Source Werkzeugen erstellt werden, während für die Präsentation auch Staroffice oder Microsoft PowerPoint verwendet werden kann.
- Quellcode der Arbeit muss regelmässig (mind. täglich) auf dem Topsy subversion Server gesichert werden (<https://svn.topsy.net>).
- Es ist ein mit Bindespinalen (am TIK vorhanden) gebundener Schlussbericht über die geleisteten Arbeit abzuliefern (2 Exemplare). Dieser Bericht besteht aus einer Zusammenfassung, einer Einleitung, einer Analyse von verwandten und verwendeten Arbeiten, sowie einer vollständigen Beschreibung der Konfiguration von den eingesetzten Programmen. Der Bericht ist in Deutsch oder Englisch zu halten. Die Zusammenfassung muss in Deutsch und Englisch verfasst werden.
- Die Arbeit muss auf CDROM archiviert abgegeben werden. Stellen Sie sicher, dass alle Programme sowie die Dokumentation und die Präsentationen sowohl in der lauffähigen, resp. druckbaren Version als auch im Quellformat vorhanden, lesbar und verwendbar sind.
Mit Hilfe der abgegebenen Dokumentation muss der entwickelte Code zu einem ausführbaren Programm erneut übersetzt und eingesetzt werden können.
- Diese Arbeit steht unter der GNU General Public License [2] (GNU GPL) v2.
- Diese Arbeit wird als Semesterarbeit an der ETH Zürich durchgeführt. Es gelten die Bestimmungen hinsichtlich Kopier- und Verwertungsrechte der ETH Zürich.

Literatur

- [1] G. Fankhauser, C. Conrad, E. Zitzler, and B. Plattner. *Topsy – A Teachable Operating System*. Computer Engineering and Networks Laboratory (TIK), 1997.
- [2] GNU General Public License. <http://www.gnu.org/copyleft/gpl.html>, Oct. 2005.
- [3] C. Jeker and B. Lutz. *Memory, IO and Process/Thread Management for Topsy v3*. Computer Engineering and Networks Laboratory (TIK), Mar. 2002.
- [4] L. Ruf. *Topsy i386 – A Teachable Operating System. The Port to the ia32 Architecture*. Computer Engineering and Networks Laboratory (TIK), 1998.

Zürich, den 19.03.2007

Lukas Ruf