

Institut für Betriebssysteme und Rechnerverbund
Übungen zur Vorlesung “Verteilte Systeme”, WS 02/03

<http://www.ibr.cs.tu-bs.de/lehre/ws0203/vs/>

Dozent: Prof. Dr. Stefan Fischer <fischer@ibr.cs.tu-bs.de> · Übungsleiter: Frank Strauß <strauss@ibr.cs.tu-bs.de>

1 Einführung und Grundbegriffe

Übung am 06.11.2002

1.1 Verteilte Systeme und das Internet Protokoll

Das Internet Protokoll (IP) ist das Netzwerk-Protokoll, das bis heute mit großem Abstand die weiteste Verbreitung erlangt hat. Wieso begünstigt diese Tatsache die Entwicklung Verteilter Systeme? Welche Ausdehnungen Verteilter Systeme profitieren insbesondere vom Erfolg von IP?

1.2 Eng gekoppelte Systeme

Im Laufe der Vorlesung werden uns hauptsächlich *lose gekoppelte Systeme* interessieren, die im allgemeinen eine größere Ausdehnung haben und mittels eines Verbindungsnetzwerkes gekopplt sind. Daneben gibt es *eng gekoppte Systeme*, die eine sehr begrenzte räumliche Ausdehnung haben (meist innerhalb eines Gerätes) und die im allgemeinen über gemeinsamen Speicher kommunizieren.

Diskutieren Sie, welche der folgenden Systeme wir als eng gekoppelte Verteilte Systeme auffassen und welche Arten des Verbunds (Folien 1-15 – 1-16) hier vorliegen:

- (a) Ein SMP¹-System mit 2 oder mehr CPUs.
- (b) Ein lokales Netzwerk (LAN), in dem sich ein Fileserver und mehrere Clients befinden.
- (c) Ein “gewöhnlicher” Rechner, in dem das Betriebssystem auf einen Festplatten-Controller und ein Grafik-Subsystem zugreift.

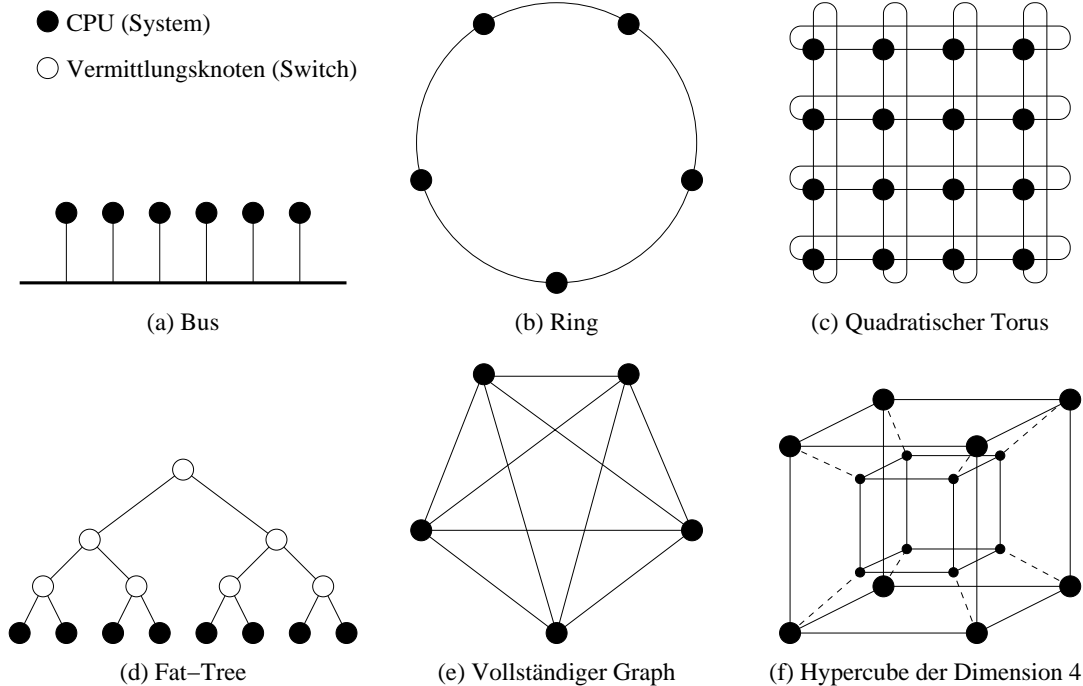
¹<http://www.webopedia.com/TERM/S/SMP.html>: SMP: Short for Symmetric Multiprocessing, a computer architecture that provides fast performance by making multiple CPUs available to complete individual processes simultaneously (multiprocessing). Unlike asymmetrical processing, any idle processor can be assigned any task, and additional CPUs can be added to improve performance and handle increased loads. A variety of specialized operating systems and hardware arrangements are available to support SMP. Specific applications can benefit from SMP if the code allows multithreading. SMP uses a single operating system and shares common memory and disk input/output resources. Both UNIX and Windows NT support SMP.

1.3 Topologien Verteilter Systeme

Insbesondere beim Einsatz Verteilter Systeme zum Zweck des Leistungsverbunds (Folie 1-16) ist die verwendete Netztopologie maßgeblich für die Effizienz des Gesamtsystems. Bestimmen Sie

- die gesamte Anzahl der Verbindungen
- die maximale Anzahl von Verbindungen zwischen zwei CPUs

als Funktion von der Anzahl der beteiligten CPUs für die folgenden Topologien. Bestimmen Sie anschließend die Vor- und Nachteile der Topologien (Aufwand der Hardware-Realisierung, Aufwand der Software-Implementierung, Bottlenecks, Skalierbarkeit, ...). Hierbei genügen begründete Abschätzungen.



1.4 Das Gesetz von Amdahl

Um eine kürzere Ausführungszeit eines Programmes als Vorteil aus dem Leistungsverbund eines Verteilten Systemes zu ziehen, muss der verwendete Algorithmus (zumindest teilweise) parallelisierbar sein (vgl. Folien 1-19 – 1-20). Nur der parallelisierte Teil des Programmes führt durch seine Nebenläufigkeit zu einer gleichzeitigen Verteilung von Teilaufgaben auf mehrere CPUs und somit zu einer Verkürzung der gesamten Ausführungszeit. Das *Gesetz von Amdahl* setzt diesen Leistungsvorteil in Beziehung zum parallelisierten Anteil des Programmes:

Es sei n die Anzahl der CPUs, $T(k)$ die Ausführungszeit des Programmes mit Parallelitätsgrad k , d.h. k CPUs können parallel arbeiten. Mit f werde der Anteil des Programmes bezeichnet, der nicht parallel ($k = 1$) ausgeführt werden kann. Es ist dann die *Ausführungszeit*

$$T(n) = fT(1) + \frac{(1-f)T(1)}{n}$$

der *Speedup*

$$S(n) = \frac{T(1)}{T(n)} = \frac{n}{1 + f(n-1)}$$

und die *Effizienz*

$$E(n) = \frac{S(n)}{n} = \frac{1}{1 + f(n-1)}$$

Verdeutlichen Sie diese Zusammenhänge, indem Sie insgesamt sechs Diagramme in Abhängigkeit von n (bei $f = 0.5$) und f (bei $n = 8$) jeweils für T , S und E anfertigen. Alternativ können Sie auch drei 3D-Diagramme ($T(f, n)$, $S(f, n)$, $E(f, n)$) für $f = 0 \dots 1$, $n = 1 \dots 8$) anfertigen².

2 Systemmodelle

2.1 Zusammenspiel mehrerer Client/Server Beziehungen

Gegeben sei das folgenden Szenario:

Ein Student ist stolzer Besitzer eines Notebooks mit einer Wireless LAN (WLAN) Netzwerkkarte. Während er sich im Informatikzentrum befindet schaltet er sein Notebook ein und meldet sich mittels dynamischer Konfiguration³ am WLAN des Instituts für Betriebssysteme und Rechnerverbund an. Er ruft Informationen vom Webserver des Instituts (www.ibr.cs.tu-bs.de) ab, um sie über einen Print-Spooler⁴ anschließend auszudrucken. Dabei betrachtet er eine Seite, die ein kleines eingebettetes Bild enthält:

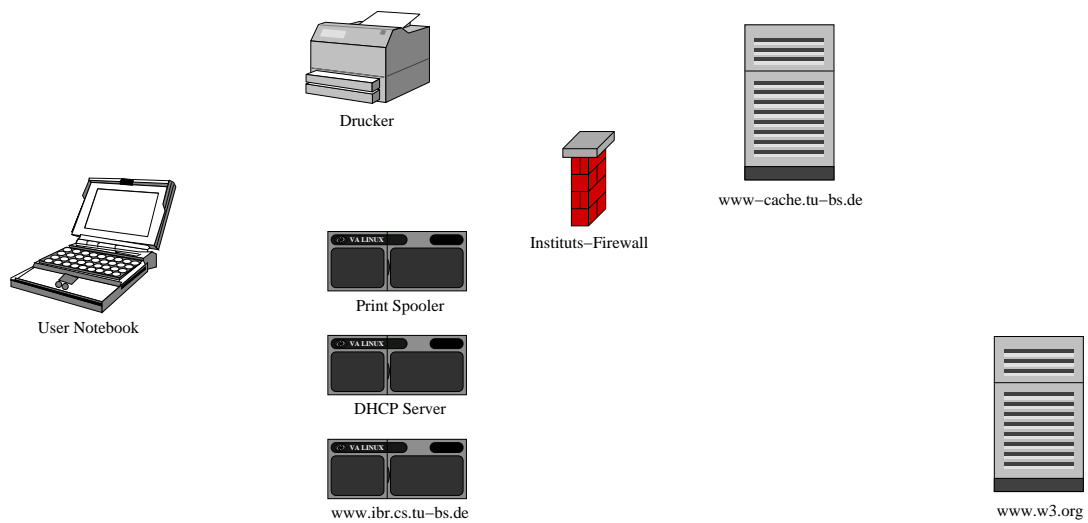
```
<html>
  ...
  <body>
    ...
    
  </body>
</html>
```

Schildern Sie den Ablauf der Kommunikation in diesem Szenario in Stichpunkten anhand folgender Skizze. Welche Komponenten agieren als Client, Server und Proxy?

²Das ist aber nur mit geeigneter Software, z.B. `gnuplot`, wirklich einfacher.

³DHCP: Dynamic Host Configuration Protocol

⁴<http://www.webopedia.com/TERM/s/spooling.html>: Spooling: Acronym for *simultaneous peripheral operations on-line*, spooling refers to putting jobs in a buffer, a special area in memory or on a disk where a device can access them when it is ready. Spooling is useful because devices access data at different rates. The buffer provides a waiting station where data can rest while the slower device catches up. The most common spooling application is print spooling. In print spooling, documents are loaded into a buffer (usually an area on a disk), and then the printer pulls them off the buffer at its own rate. Because the documents are in a buffer where they can be accessed by the printer, you can perform other operations on the computer while the printing takes place in the background. Spooling also lets you place a number of print jobs on a queue instead of waiting for each one to finish before specifying the next one.



2.2 Auswahl von Systemmodellen

Welche Eigenschaften (vgl. Folien 2-14 ff.) hätten oder haben

- (a) die Bereitstellung von Microsoft ServicePacks mittels Peer-to-Peer “Tauschbörsen” wie Gnutella?
- (b) der Austausch von Copyright-geschützten Inhalten mittels eines zentral betriebenen Servers?