

# Betriebssysteme

## *Kapitel 7: Files*

### *7.3: Pipes*

Stand: WS 08/09

Prof. Dr. Wolfgang Kuchlin

*Dipl.-Inform., Dr. sc. techn. (ETH)*

**Arbeitsbereich Symbolisches Rechnen  
Wilhelm-Schickard-Institut für Informatik  
Fakultät für Informations- und Kognitionswissenschaften**

**Universität Tübingen**

**Steinbeis Transferzentrum  
Objekt- und Internet-Technologien (OIT)**

**[Wolfgang.Kuechlin@uni-tuebingen.de](mailto:Wolfgang.Kuechlin@uni-tuebingen.de)  
<http://www-sr.informatik.uni-tuebingen.de>**



# Pipes

---

- Eine Pipe (Rohrleitung) ist eine FIFO Kommunikationsstruktur zwischen Prozessen.
  - Pipes wurden traditionell mittels des Dateisystems realisiert.
  - Eine neue Realisierung benutzt sockets (→ VL „Verteilte Systeme“).
- Nach dem Öffnen können sie logisch wie eine Datei beschrieben bzw. gelesen werden, was für Transparenz und Überschaubarkeit sehr wichtig ist.

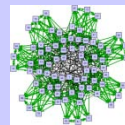


# Pipes

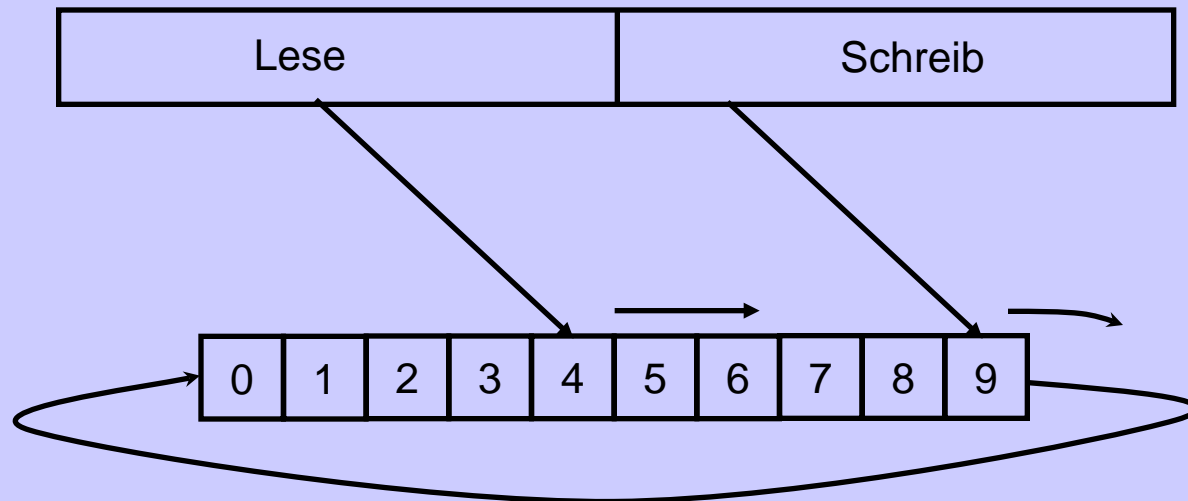
---

Mittels einer Datei kann eine Pipe wie folgt realisiert werden:

- Der Pufferspeicher der Pipe wird durch eine feste Anzahl von Blöcken ( in UNIX z.B. die 10 direkten Blöcke) der Datei realisiert, die logisch als Ringpuffer organisiert werden.
- Zudem gibt es einen Lesezeiger und einen Schreibzeiger, die auf die nächste zu lesende bzw. zu schreibende Position verweisen.
- Lese- und Schreibzeiger überholen sich nicht, sondern die entsprechenden Prozesse werden zuvor blockiert und dann wieder aufgeweckt, wenn wieder Platz ist.



# Pipes



Implementierung einer pipe über inode

- Natürlich können mehrere Prozesse in die pipe schreiben oder aus ihr lesen.
- Es gibt unbenannte (unnamed) und benannte (named) pipes zur Kommunikation zwischen verwandten bzw. nicht verwandten Prozessen.



# Pipes – Systemaufrufe

## ➤ unbenannte pipe mittels Filedeskriptoren

- `fds[0]` für das Lesen
- `fds[1]` für das Schreiben

```
int fds[2];  
pipe(fds);
```

```
mkfifo(path, access-mode)  
char* path, mode_t access-mode;
```

## ➤ pipe mit Namen `path`

- Liefert eine pipe mit Namen `path`.
- Die Lese- und Schreibdeskriptoren werden wie üblich durch nachfolgendes Öffnen der Datei zum Lesen bzw. Schreiben erhalten.
- Da die pipe einen Namen hat und explizit zum Lesen/Schreiben geöffnet wird, eignet sie sich zur Kommunikation nicht verwandter Prozesse.

