

# Parallele Programmierung mit MOSTflexiPL

*Christian Heinlein*  
*Studiengang Informatik*  
*Hochschule Aalen – Technik und Wirtschaft*

MOSTflexiPL (Modular, Statically Typed, Flexibly Extensible Programming Language, vgl. <http://flexipl.info>) ist eine Programmiersprache, deren Syntax vom Anwender nahezu beliebig erweitert und angepasst werden kann. Aufbauend auf einer kleinen Menge vordefinierter Grundoperatoren, können nach Belieben weitere Operatoren für unterschiedlichste Zwecke definiert werden. Da Operatoren beliebig viele Namen besitzen und auf beliebig viele Operanden angewandt werden können, decken sie neben den üblichen Präfix-, Infix- und Postfix-Operatoren (z. B.  $-x$  für Vorzeichenwechsel,  $x+y$  für Addition oder  $n!$  für Fakultät; die Operatornamen sind jeweils fett gedruckt) auch Mixfix-Operatoren (z. B. `a[i]` für Array-Elementzugriff), Kontrollstrukturen (z. B. `if ... then ... else ... end`), Typkonstruktoren (z. B. `List ...`) und Deklarationsformen (z. B. `A -> B`, um `A` als Untertyp von `B` zu deklarieren) ab.

Im Rahmen einer studentischen Arbeit an der Hochschule Aalen, wurden unlängst Grundoperatoren für parallele Programmierung zur Sprache hinzugefügt:

- `x | y` führt die Teilausdrücke `x` und `y` parallel aus und wartet auf die Beendigung von beiden.
- `||x` führt den Teilausdruck `x` asynchron aus, ohne auf seine Beendigung zu warten.
- `"s" : sem` deklariert ein Semaphor `s` mit Anfangswert 0.  
`+s` bzw. `-s` entsprechen den Operationen V und P von Dijkstra, d. h. sie erhöhen bzw. erniedrigen das Semaphor `s` um 1, wobei `-s` anschließend ggf. wartet, bis der Wert des Semaphors wieder mindestens 0 ist.

Aufbauend auf diesen Grundoperatoren, können ohne große Mühe „höherwertige“ Operatoren für parallele Programmierung realisiert werden, zum Beispiel:

- Eine `for`-Schleife, deren Schleifenrumpf für jeden Wert der Laufvariablen parallel ausgeführt wird
- Asynchron berechnete Werte (sog. futures)
- Variablen, die nur abwechselnd geschrieben und gelesen werden können (sog. MVars in Haskell)
- Pfadausdrücke zur deklarativen Formulierung zulässiger Ausführungsreihenfolgen von Prozeduren o. ä., deren Einhaltung automatisch durch eine Abbildung auf geeignete Semaphoroperationen gewährleistet wird
- Software Transactional Memory
- u. v. a. m.

Damit eignet sich MOSTflexiPL auch hervorragend zum Experimentieren mit neuen Sprachkonstrukten, weil sich diese häufig in wenigen Minuten implementieren lassen.