

**Einleitung**

Dieses **“Schnupperpraktikum”** soll Dir einen ersten Eindruck vom Verteilten Rechnen geben. Du lernst die Möglichkeiten des **High Performance Computing (HPC)** kennen. Weiterhin erhältst Du erste Informationen über Umsetzungsmöglichkeiten für verschiedene Berechnungsaufgaben. Als eine Art Schweizer Taschenmesser hat sich die **Message Parsing Interface (MPI)** Technologie in der IT-Welt etabliert. Diese Technologie bietet eine Schnittstelle um Berechnungsaufgaben an mehrere Rechner im Verbund (Cluster) zu verteilen, zu berechnen und die Ergebnisse zusammenzuführen.

**Beispiele**

Auf den Arbeitsplatzoberflächen der Laborrechner findest Du ein Beispielprogramm, welches in der Programmiersprache **C** umgesetzt ist – mach Dir keine Sorgen, wir helfen Dir beim Verstehen, frag einfach! Dieses Programm berechnet das Integral einer Kosinus-Funktion, Du kannst es Dir mit einem Editor Deiner Wahl (z.B. *kate*, *gedit*) öffnen und durchlesen:

```
parallel_integration.c
```

**Das Programm, “hacken” wir ne' Runde**

Das Programm *parallel\_integration.c*, arbeitet nach dem Senden-/Empfangen-Prinzip (Client-Server-Applikation). Diese Aufgaben übernehmen die Funktionen – *MPI\_Send(...)*, *MPI\_Recv(...)* – mit ihren dazugehörigen Parametern. Das Integral der Kosinus-Rechnung wird mit der Funktion *integral(...)* in den verschiedenen Prozessen durchgeführt, die Parameter *UNTERGRENZE*, *OBERGRENZE* und *UNTERTEILUNGEN* beschreiben ein Berechnungspaket. Zur Erinnerung -  $\int_{UNTERGRENZE}^{OBERGRENZE} \cos(x_i)$  - die Schritte zwischen  $\Delta GRENZEN = OBERGRENZE - UNTERGRENZE$  werden mit *UNTERTEILUNGEN* beschrieben -  $x_{i+1} = x_i + \frac{\Delta GRENZEN}{UNTERTEILUNGEN}$  - je mehr Unterteilungen, desto genauer kann das Ergebnis sein!

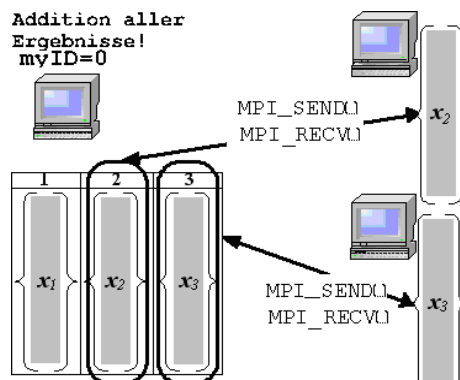


Abb 1: MPI Kommunikation - Senden/Empfangen

- Versuch Dir vorher klar zu machen was geschehen wird!
- Schau Dir die Quelltexte der Programme an und spiel einfach mal mit den Parametern!
- Führe (nächste Abschnitt) das Programm jeweils aus!

**Probiere u.a. folgende Werte:**

| Variablen      | Werte     |          |             |                    |             |
|----------------|-----------|----------|-------------|--------------------|-------------|
| UNTERGRENZE    | 0         | M_PI     | 0           | M_PI / 2.0f        | 0           |
| OBERGRENZE     | M_PI      | 2 * M_PI | 2.0f * M_PI | (3.0f/2.0f) * M_PI | M_PI / 2.0f |
| UNTERTEILUNGEN | 1 - 10000 |          |             |                    |             |

**Ausführen**

Das Programm übersetzt Du mit einem Skript (Programm), das auf der Arbeitsplatzoberfläche liegt. Mit einem Doppelklick der linken Maustaste auf die Datei *HierErstellen.sh*, kannst Du das Programm ausführen. Es öffnet sich ein Fenster in dem Du auch das Programm **starten** kannst!

Du hast zwei Möglichkeiten das Programm auszuführen, versuch beide und vergleiche :-)



- 1) Gebe im Fenster folgende Zeile ein und drücke danach **Enter/Return**: `./parallel_integration`
- 2) Nun tätige einmal die Eingabe: `mpixec -np 10 ./parallel_integration`

Huch, was ist geschehen? Hast Du eine Idee???  
Frag uns, wir beißen nicht...

Das Bild rechts könnte hilfreich sein :-)

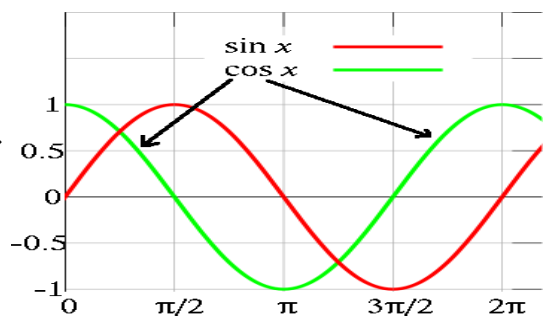


Abb 2: Sinus-Kosinus-Funktion