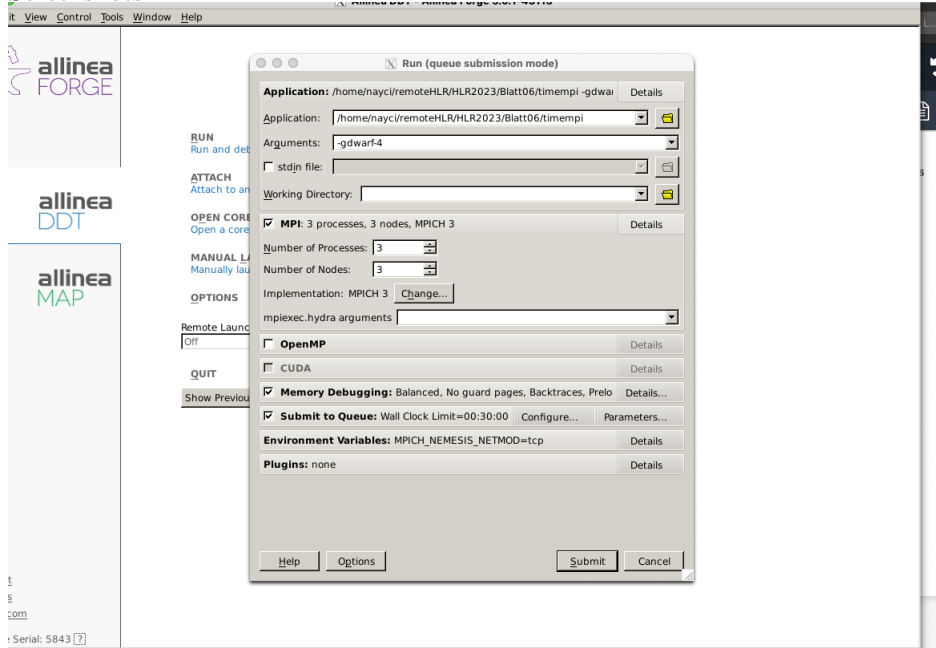


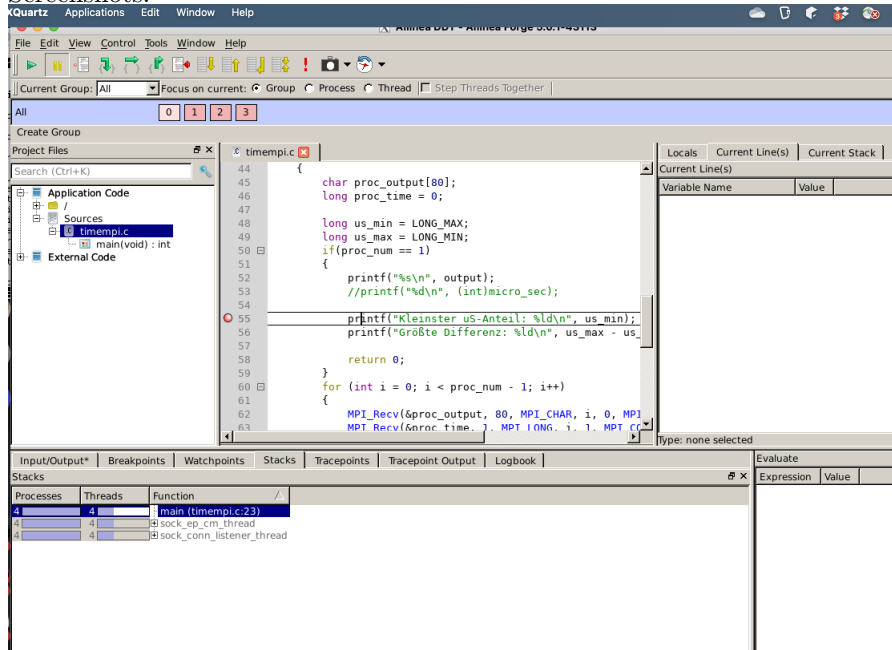
HLR #6

1. Screenshots:



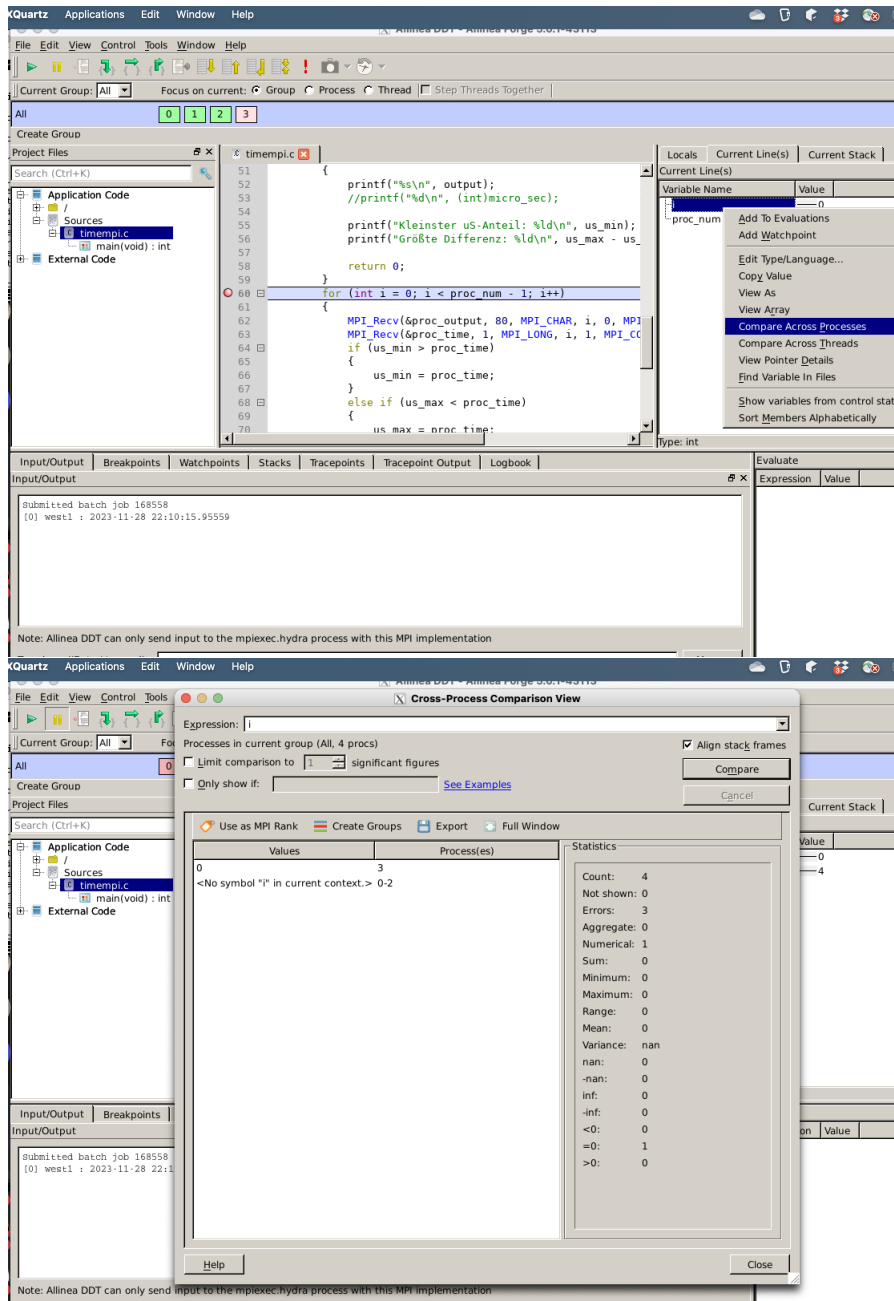
Argumente entweder über *ddt./programarg1arg2* oder im Argument-Feld, wie man auf dem Bild erkennt.

2. Screenshots:



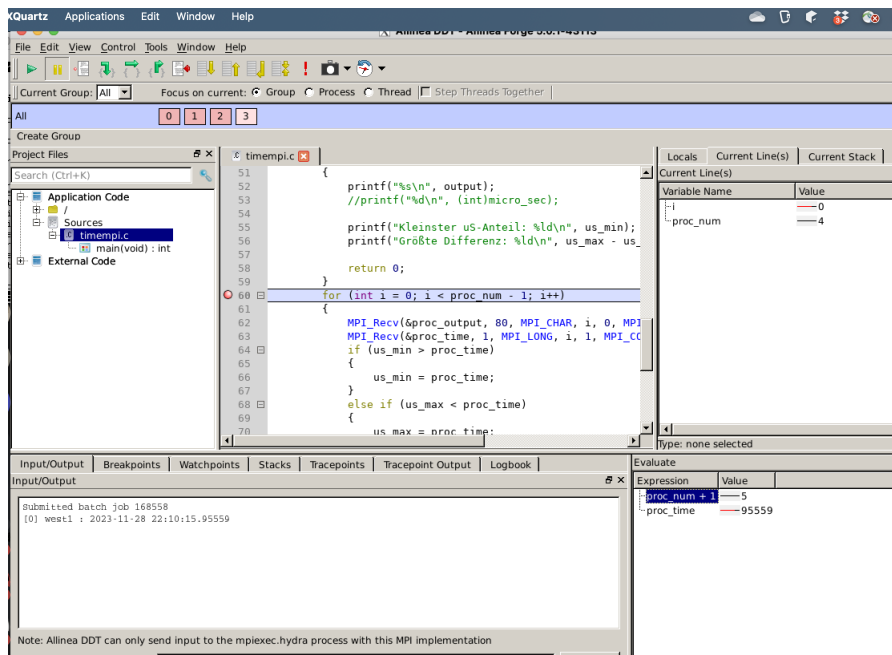
Step-Into (Springt in Funktion), Step-Over (führt Funktion aus, geht in nächste Zeile), Step-Out (Führt Funktion zu Ende, geht in Funktion, aus der Funktion ausgeführt wurde)

3. Screenshots:



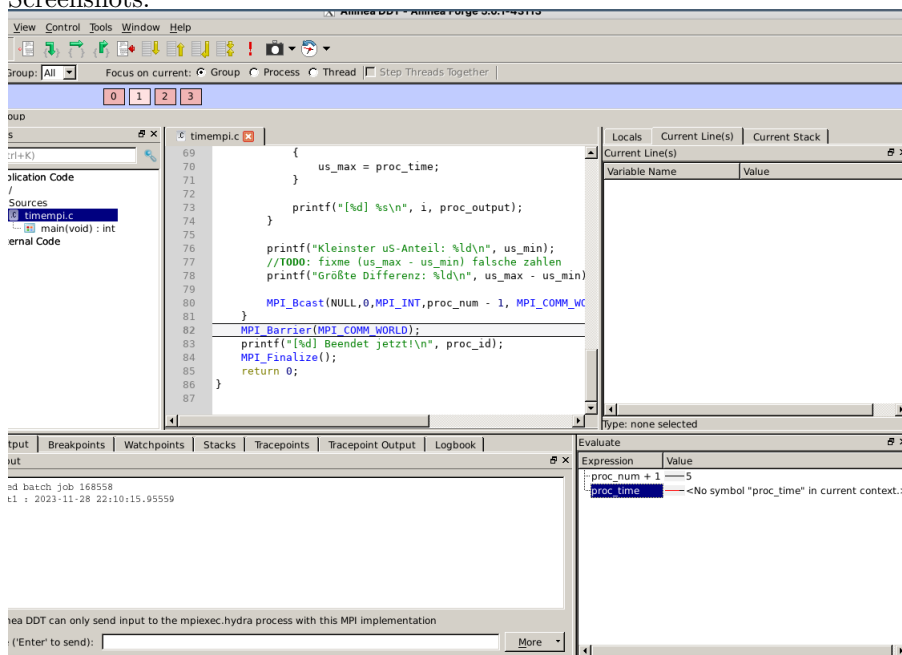
Wenn man auf die Linien klickt, kann man sich die Variable im Vergleich zwischen allen Prozessen anschauen. Im Beispiel hier ist die Laufvariable i die, die im Thread $n - 1$ anzeigt, von welchem Prozess die gesendete Nachricht stammt. Da die for-Schleife, also auch das i , nur in diesem Thread vorhanden sind, wird korrekter Weise angezeigt, dass die Variable in den anderen Prozessen nicht existiert.

4. Screenshots:



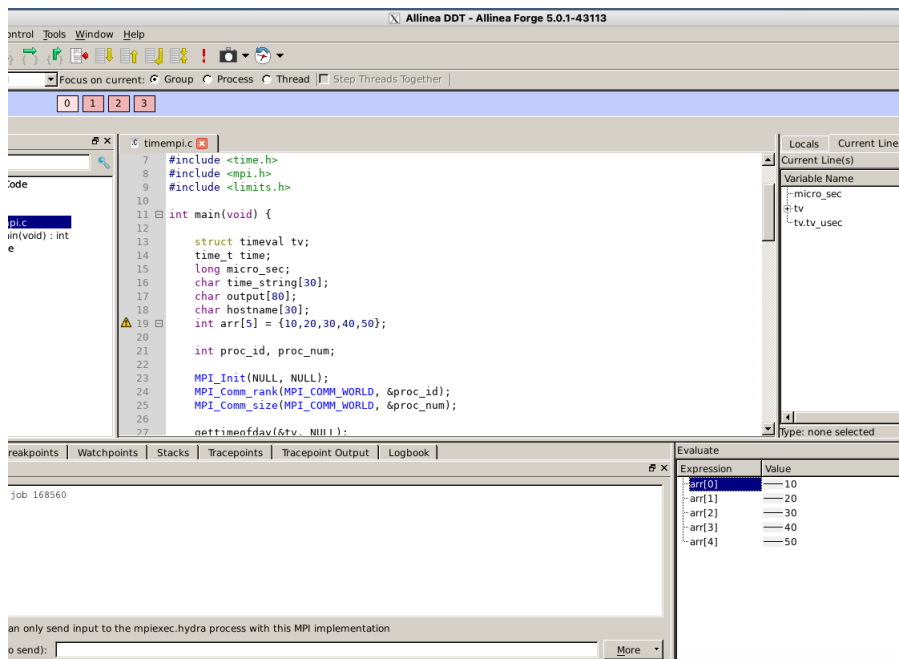
Das evaluate-Feld wie in vielen anderen Debuggern auch wertet Ausdrücke in C-Syntax aus. So kann man mit allen Variablen beispielsweise Rechenoperationen oder anderes durchtesten.

5. Screenshots:



Wie man erkennen kann, werden die Ausdrücke im evaluate-Feld für jeden Wechsel neu evaluiert, jeweils im Kontext des Threads. Da im Beispiel `proc_time` nur im Thread $n - 1$ definiert ist, ist es undefiniert im Thread 1

6. Screenshots:



Unten rechts im evaluate.Feld kann man über $arr[i]$ mit i als Index auf das initialisierte Array zugreifen, wenn die Initialisierung vom Debugger ausgeführt wurde.