

# Monitoring und Debugging

Azeddine Ayadi, Marc Lörner, Martin Schauer

June 6, 2003

## 1 Domänenlexikon

### Monitoring

Ist eine Systembeobachtung, um interessante Ereignisse, Operationen und Zustände von Funktionseinheiten aufzuzeichnen.

Ziel: Leistungsengpässe identifizieren, um Systemoptimierung ansetzen zu können.

#### Timestamps

- Protokollieren von Systemereignisse
- Laufzeitmessung von Funktionen

### Debugging

Ist Synonym für Fehlerbeseitigung, Fehlerverfolgung oder Dienstprogramm.

Debugger kann vom Betriebssystem beim Programmabsturz gestartet werden, dann nennt man ihn „Post-Mortem-Debugger“.

Wird ein Debugger vom Programmierer aufgerufen so nennt man ihn „Dynamische Debugger“. Es wird benutzt um:

- Zeigervariablen zu verfolgen.
- Variablenwerte zu ändern.
- Maschineninstruktionen zu verfolgen.

#### Trace

- Verfolgen und/oder protokollierung des Kontrollflusses
- Abfolge der Funktionsaufrufe
- Ausgabe von Variablenwerten

#### Invariance

Zusicherung einer bestimmten Eigenschaft, die als mathematische Bedingung (Ungleichung, Gleichung) formuliert werden kann.

- Objekte einer Klasse müssen in stabilem Zustand sein, insbesondere nach:

- Erzeugung des Objekts.
- Zugriff und Änderung eines Attributes.
- *This\* ≠ NULL* muss immer gelten.
- Während der Programmausführung muss jeder Stackpointer *SP* die Bedingung *SP < 32KB* immer erfüllt sein.

## 2 Domänenanalyse

Zur Domäne "Debugging und Monitoring" gehören Funktionen, die den Entwickler zum einen bei der Fehlersuche unterstützen, zum anderen um Leistungsmessungen am laufenden System durchzuführen und zu protokollieren. Dahingehend sollen Funktionen zur Verfügung gestellt werden, die

- die Abfolge der Funktionsaufrufe/-ausführung verdeutlichen
- Variablenwerte aufzeichnen/anzeigen
- Variablenwerte auf Einhaltung eines bestimmten Wertebereichs überprüfen (Invarianzen, z. B. *this – Pointer ≠ NULL*)
- Zeitstempel aufzeichnen.
- Laufzeit von Funktionen bestimmen.
- Lebensdauer von Objekten bestimmen.

Zur Domäne gehören ebenfalls weitere Funktionen zur Realisierung eines "Post-Mortem"- oder eines "Dynamischen"-Debuggers (siehe Domänenlexikon):

- Setzen von Breakpoints (und damit Anhalten der Programmausführung).
- Schrittweise Programmausführung (auf Maschinenbefehlebene).
- Anzeigen und Ändern von Variablenwerten.
- Visualisierung des aktuellen Stackinhalts.
- Anzeigen von Werten an einer bestimmten Speicherstelle.

Zur Aufzeichnung/Visualisierung der Systembeobachtung, soll es möglich sein, Daten auf dem Display des *Lego Mindstorms* anzuzeigen oder die Daten über die Infrarotschnittstelle auf einen anderen Rechner zu übertragen ("Remote") und diese dort entweder auf dem Bildschirm auszugeben oder in eine Logdatei zu schreiben.