

Programmierung von drahtlosen Sensornetzen mit UML2 Aktivitätsdiagrammen

Christoph Damm, Gerhard Fuchs

Universität Erlangen Nürnberg
Lehrstuhl Informatik 7

Hamburg-Harburg, 14. August 2009



Inhalt

- 1 Motivation
 - Trend / Fragestellungen
 - Resultierende Frage
 - Relevante Vorarbeiten
- 2 Unser Ansatz
 - UML2-Aktivitätsdiagramme
 - Beispiel-Programm
 - Prinzip
- 3 Zusammenfassung und Ausblick
 - Zusammenfassung
 - Ausblick



Wahrgenommener Stand der Forschung:

- Sensornetze sind ein aktueller Forschungszeitung;
 - "Kinderkrankheiten" wie z.B. von Routing, Clustering, Energieprobleme bei einzelnen Sensorknoten sind diskutiert ✓;
- ⇒ Es ist eine solide Basis vorhanden.

Offene Punkte:

- Wie setzt man die gewonnen Erkenntnisse zusammen?
- Wie geht man mit der große Menge an Sensorknoten um?
- Wie erstelle ich Anwendungen in der Praxis?



Wie programmiert man eine **(sehr)**
große Anzahl an Sensorknoten, so
dass Sie eine **gemeinsame Aufgabe**
ausführen?



Arbeiten anderer Gruppen:

- **Sugihara und Gupta** (2008), *Programming models for sensor networks: A survey*;
- **Guerrero, Jacobi und Buchmann** (2007), *Workflow Support for Wireless Sensor Actor Networks - A Position Paper*;



Studentische Arbeiten an unserem Lehrstuhl:

- **Ipek** (2006), *Aufgabenbeschreibung für mobile Roboterschwärme + ROBRAIN*;
- **Damm** (2008), *Implementierung und Bewertung eines RDF-basierten Frameworks zur Interpretierung und Ausführung von UML2-Aktivitätsdiagrammen auf Sensorknoten*;

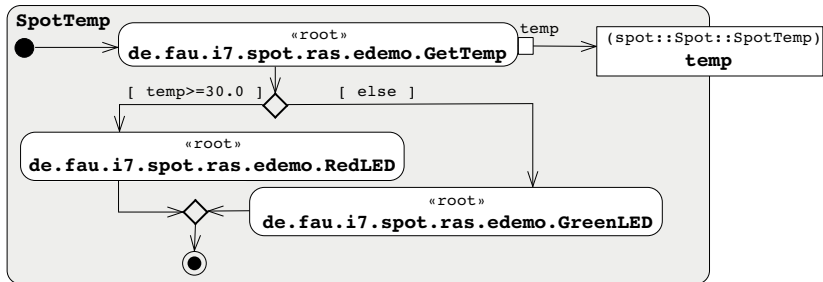


Wir verwenden **UML2-Aktivitätsdiagramme (UAD)** für die Programmierung von Sensornetzen.

- Ein UML2-Aktivitätsdiagramm ist durch verschiedene Arten von Knoten definiert, die durch Objekt- und Kontrollflüsse verbunden sind;
- Ein Kontrollfluss transportiert "Tokens";
- Ein Objektfluss transportiert Objekte;

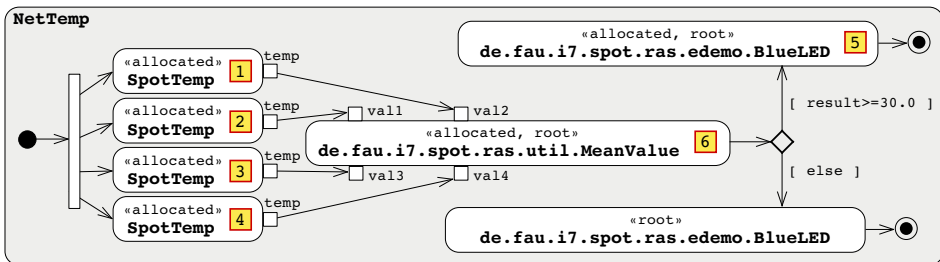
[nach Oestereich (2005), *Die UML 2.0 Kurzreferenz für die Praxis*]





- Kein `« allocated »` verwendet;
 - Wird lokal auf dem Spot ausgeführt;
- ⇒ Plattform-Zentrische Programmierung.





1 instr = random:uniform:a01,a02

2 instr = random:uniform:a03,a04

3 instr = random:uniform:a05,a06

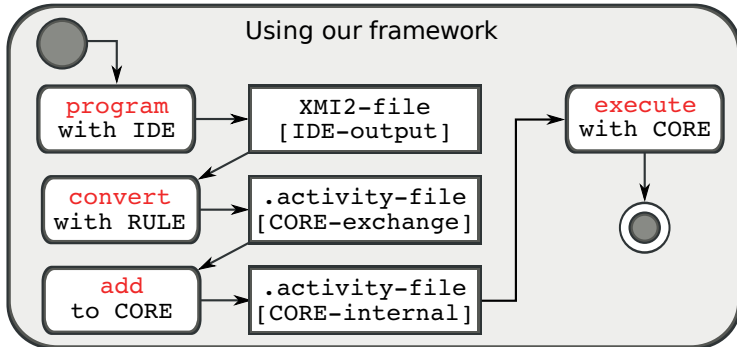
4 instr = random:uniform:a07,a08

5 instr = static::a10

6 instr = static::a09

- `« allocated »` verwendet;
- ⇒ **Netz-Zentrische** Programmierung.





- Als IDE verwenden wir Papyrus UML;
- CORE, RULE und ACCESS sind von uns implementiert;



DEMO



- Wir sehen einen Trend zur Untersuchung großer Sensornetze;
- Wie erstellt man Applikationen?
- Wie programmiert man große Sensornetze, in denen Spots kooperieren?
- **Wir untersuchen in wie weit UAD zur Lösung dieser Fragen verwendet werden können.**
- Wir haben einen prototypisches Framework realisiert und an einem kleinen Sensornetz getestet.





- **Aktuelle Arbeit:**

- Untersuchung intelligenterer Action-Allocation Mechanismen;
- Allgemeine Gedanken zur Erweiterung;
- Erweiterung des Sensornetzes;
- Aufbau und Integration von Robotern;

- **Offene Fragen:**

- Welche Funktionen können / müssen integriert werden?
- Was sind gute Szenarien für die Evaluation unseres Frameworks?



Danke für ihre Aufmerksamkeit

gerhard.fuchs@informatik.uni-erlangen.de

