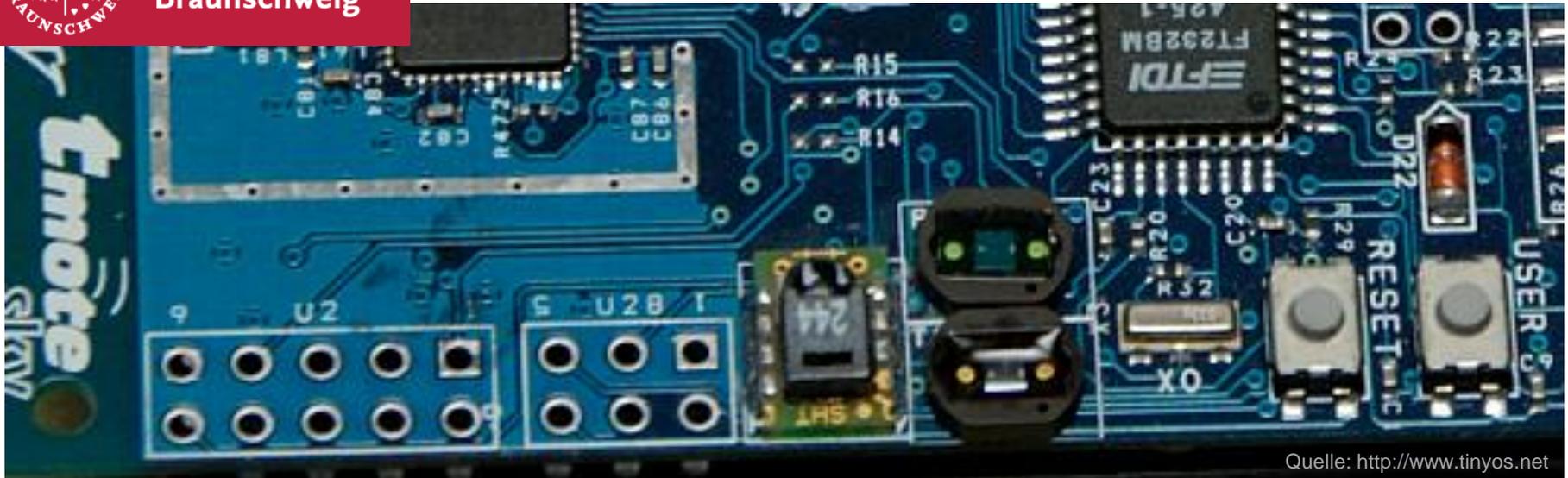




Technische
Universität
Braunschweig



Quelle: <http://www.tinyos.net>

Die Implementierung von Sensorschnittstellen in Wiselib, der Algorithmenbibliothek für Sensornetzwerke

Mario Wozenilek, 03.08.2011

Übersicht

- Aufgabe
- Umgebung
 - Wiselib
 - Hardware/Software
 - Sensoren
- Konzepte
- Aktueller Stand

Übersicht

- Aufgabe
- Umgebung
 - Wiselib
 - Hardware/Software
 - Sensoren
- Konzepte
- Aktueller Stand

Aufgabe

- Implementierung von Sensorschnittstellen
 - „Sensorklassen“ bereitstellen, die innerhalb der Wiselib verwendet werden können um Sensordaten (Temperatur, Druck, Feuchtigkeit, etc.) in Algorithmen und Anwendungen zu nutzen.
 - „Komfortfunktionen“ umsetzen, z.B. Messintervall setzen, Nachricht bei überschreiten von Grenzwerten etc.
- Portabilität, Effizienz, Heterogenität
 - Sensoren für mehrere „Betriebssysteme“ (Plattformen) nutzbar machen.
- Demoanwendung
 - Wiselib – Anwendung erstellen um neue Sensorschnittstellen zu demonstrieren
- ...ach ja
 - schriftliche Ausarbeitung

Übersicht

- Aufgabe
- Umgebung
 - Wiselib
 - Hardware/Software
 - Sensoren
- Konzepte
- Aktueller Stand

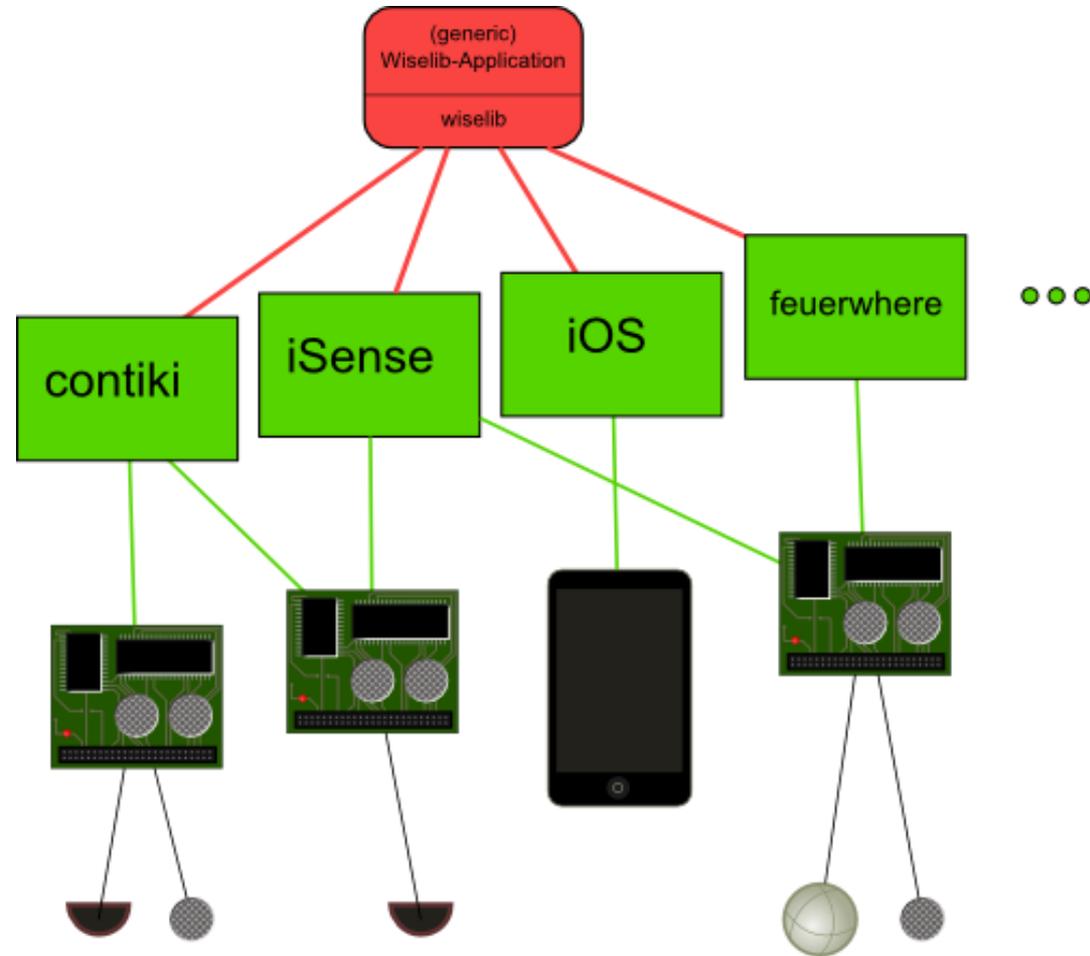
Umgebung – Wiselib

Die Wiselib...

- ...ist für **Sensornetzwerke** gedacht.
- ...bedient mehrere Plattformen.
- ...ermöglicht die Verwendung von vorgefertigten Algorithmen.
- ...erlaubt generische und plattformunabhängige Anwendungen.
- ...hat bisher nur sehr wenige Sensorschnittstellen.

Umgebung – Wiselib

- (Generische) Wiselib – Applikation
- Plattformen (Software)
- Plattformen (Hardware)
- Sensoren



Umgebung – Hardware / Software

Quelle: Wiselib: A Generic Algorithm Library for Heterogeneous Sensor Networks, T.Baumgartner et al.



iSense



iMote2



ScatterWeb MSB



Tmote Sky / TelosB

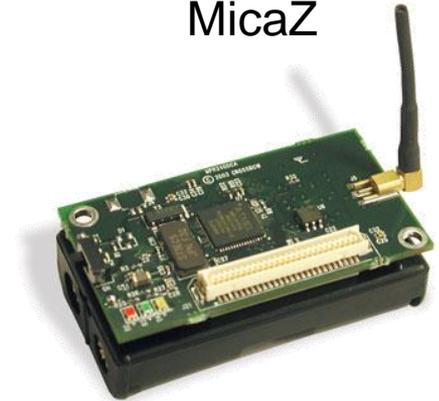
iPhone



Android-Phones



MicaZ



Quelle: <http://hypesrus.com/>

Quelle: <http://www.droidphonesblog.com>

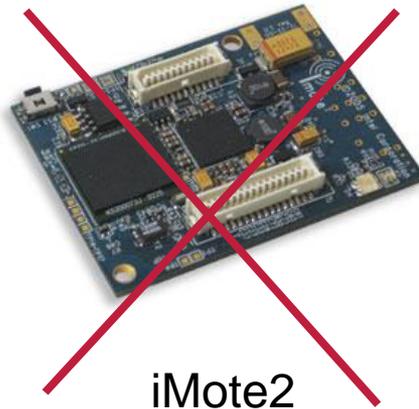
Quelle: <http://www.cmt-gmbh.de/>

Umgebung – Hardware / Software

Quelle: Wiselib: A Generic Algorithm Library for Heterogeneous Sensor Networks, T.Baumgartner et al.



iSense



iMote2



ScatterWeb MSB



Tmote Sky / TelosB

iPhone



Android-Phones



MicaZ



Quelle: <http://hypesrus.com/>

Quelle: <http://www.droidphonesblog.com>

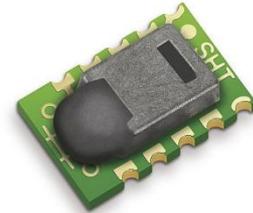
Quelle: <http://www.cmt-gmbh.de/>

Umgebung – Sensoren

- iSense
 - Environment Module
 - Lichtsensor
 - Thermometer
 - Security Module
 - Passiver Infrarot Sensor (PIR)
 - 3-Achsen-Beschleunigungssensor
- TelosB / Tmote Sky
 - Sensor für rel. Luftfeuchte
 - Temperatursensor
 - 2 x Lichtsensor (sichtbares Licht / sichtbares Licht + Infrarot)
 - Taster (Button sensor)



Quelle: <http://www.instructables.com>



Quelle: <http://img.directindustry.de>

Übersicht

- Aufgabe
- Umgebung
 - Wiselib
 - Hardware/Software
 - Sensoren
- **Konzepte**
- Aktueller Stand

Konzepte

Request Sensor

Erlaubt Zugriff auf Sensorwerte durch Verwendung des () Operators (Einfacher Methodenaufruf).

Rückgabetyt vorgegeben

Managed Sensor

Erlaubt Zugriff auf Sensorwerte durch Verwendung des () Operators (Einfacher Methodenaufruf).

Rückgabe encodiert (serialisiert)

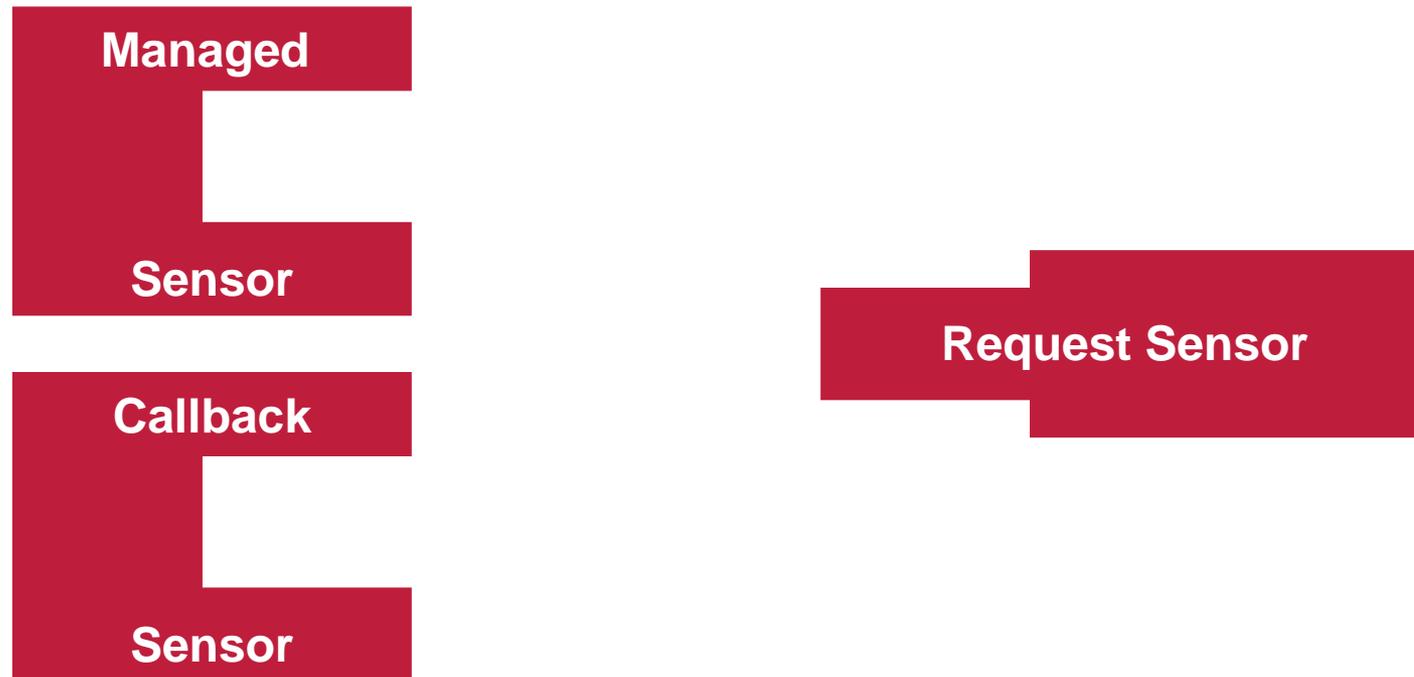
Callback Sensor

Ruft bei Überschreitung von Grenzwerten zuvor registrierte Funktion auf.

Rückgabetyt vorgegeben

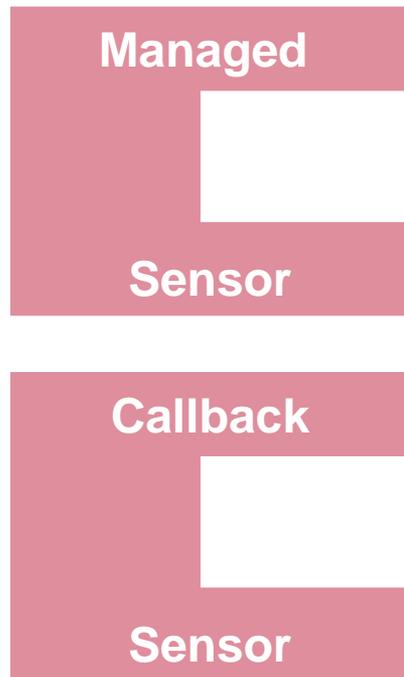
Implementierungen dieser drei Konzepte sollen für jede Sensorhardware verfügbar sein.

Konzepte



Durch „Wrapperklassen“ wird pro Sensor nur noch eine Implementierung nötig. (Ausnahme iSense)

Konzepte



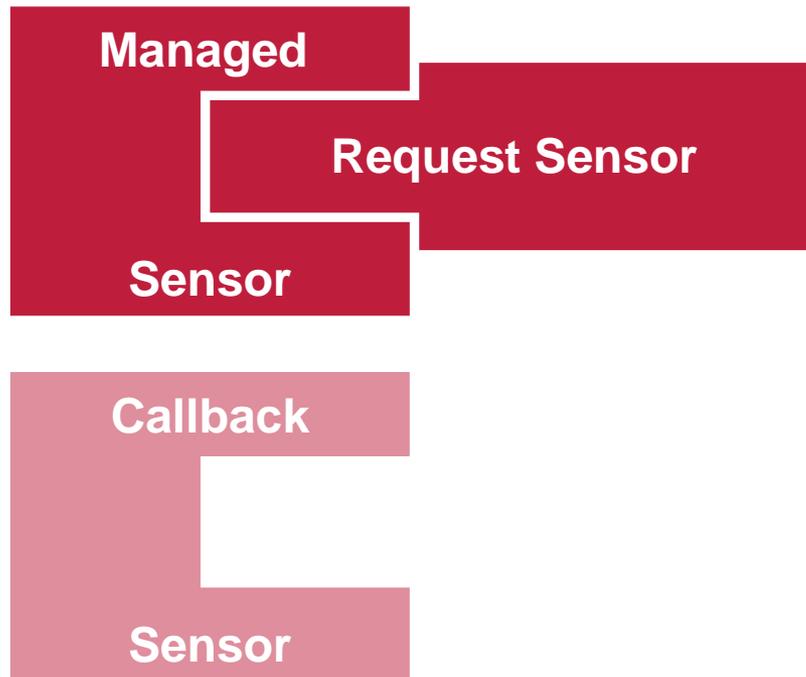
LightRequestSensor

Zugriff: operator()
Zugriffsart: Abfrage
Rückgabe: uint32 (Lux)



Beispiel Lichtsensor

Konzepte



LightManagedSensor

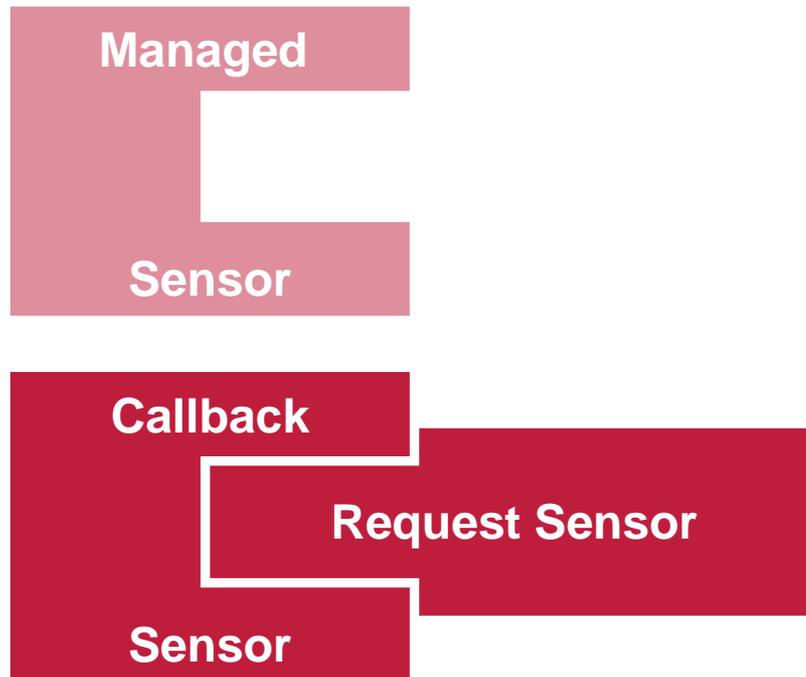
Zugriff: operator()

Zugriffsart: Abfrage

Rückgabe: char*
(serialisierter uin32 [Lux])

Beispiel Lichtsensor

Konzepte



LightCallbackSensor

Zugriff: registrieren von
Handlerfunktion

Zugriffsart: „Rückruf“
registrierter Funktion bei
Überschreiten des
Grenzwertes

Rückgabe: uint32 (Lux)
(Parameter der
Handlerfunktion)

Beispiel Lichtsensor

Übersicht

- Aufgabe
- Umgebung
 - Wiselib
 - Hardware/Software
 - Sensoren
- Konzepte
- **Aktueller Stand**

Aktueller Stand

		Humidity	Temperature	Light	PIR	Accelerometer	Battery	Button
iSense OS	Call.	N/A	✓*	✓	✓	✓	?	✗
	Req.	N/A	✓	✓	✓	✓	?	✗
Contiki		✓	✓	✓*	N/A	N/A	✗	✗
TinyOS		✗	✗	✗	N/A	N/A	✗	✗

- Wrapperklassen
 - Managed Sensor: ✓ (schon vor der Arbeit vorhanden)
 - Callback Sensor: ✗

- Demoanwendung: ✗

- schriftl. Ausarbeitung: ✗

Fragen?

