

Umweltmonitoring mittels des Internets der Dinge

M. Volery

1 Übersicht

Umweltmonitoring mittels des Internets der Dinge bringt Probleme mit sich. Welche das sind oder sein können und mögliche Lösungen werde ich in dieser Ausarbeitung zu erklären versuchen.

Dazu werde ich zuerst auf die Umweltüberwachung mittels des Internets der Dinge allgemein eingehen und dann anhand von drei Beispielen Probleme und mögliche Lösungen aufzeigen.

2 Einleitung

Systeme, die Umweltdaten messen und auswerten, können in vielen Bereichen sinnvoll eingesetzt werden. Ein mögliches Beispiel wäre ein System, das die Wassermenge, Fließgeschwindigkeit und Wassertiefe eines Flusses an verschiedenen Stellen misst und so zur Vorhersage von Ort und Zeit von Überflutungen benutzt werden könnte.

Heute bestehen solche Systeme meistens aus wenigen Sensoren an statischen Punkten. Dies hat zur Folge, dass diese Systeme oft nur ungenaue Resultate liefern.

Fortschritte bei der Hardware ermöglichen nun aber immer kleinere, robustere, effizientere und billigere Komponenten. Dadurch werden vermehrt Systeme eingesetzt, die mit bedeutend mehr Sensoren arbeiten als bisherige. Allerdings erfordert diese grössere Anzahl an Sensoren auch neue Techniken, um zum Beispiel die Wartbarkeit zu gewährleisten. So ist es nicht sinnvoll, dass hunderte von Sensoren ihre gesammelten Daten unbearbeitet an einen zentralen Rechner senden. Je länger je mehr werden deswegen Sensoren die gesammelten Daten direkt bei sich verarbeiten. Ebenso müssen neue Kommunikationstechnologien her, schliesslich möchte man nicht kilometerweise Kabel verlegen, um die Daten zu übertragen und die Sensoren miteinander zu verknüpfen. Hier hilft der Fortschritt in der Wireless-Kommunikation.

Zunehmend werden so höhere Dienstleistungen möglich. Die Resultate werden aufgrund der grösseren Anzahl an Sensoren genauer. Durch die Verarbeitung im Sensor und der Kommunikation der Sensoren untereinander und des Systems mit weiteren Dingen vereinfacht sich die Arbeit des Endbenutzers.

3 Umweltmonitoring mittels des Internets der Dinge

Was ist nun das Internet der Dinge genau?

Das Internet der Dinge: Auf der deutschen Wikipedia-Seite [4] kann man Folgendes finden: "Das Internet der Dinge, englisch auch als "Internet of Things" bekannt, bezeichnet die elektronische Vernetzung von Gegenständen des Alltages. Ausgehend von der Vorstellung, dass Gegenstände (Dinge) mit einer eigenen Intelligenz ausgestattet sind, zielt das Internet der Dinge auf einen selbständigen Informationsaustausch der Dinge untereinander ab. [...] Das Ziel dieser Entwicklung ist es, dass Objekte mehr Informationen als nur ein Identifikationsmerkmal wie z.B. einen Barcode tragen. Zudem können sie in einem wesentlich grösseren Rahmen mit ihrer Umwelt kommunizieren. Diesen intelligenten Objekten soll es möglich sein, selbstständig ihren Prozessablauf zu organisieren."

Das Internet der Dinge geht also über die blosser Verknüpfung von Gegenständen hinaus und umschliesst auch smarte Dinge. Also Dinge, die selber eine gewisse Intelligenz besitzen. Das typische Beispiel ist wohl der Kühlschrank, der weiss, was er enthält und aufgrund dessen Rezeptvorschläge macht und fehlende Lebensmittel automatisch übers Internet nachbestellt.

Die Ideen des Internets der Dinge erlauben also "höhere" Dienstleistungen. Solche Dienstleistungen nehmen uns immer mehr Arbeit ab, indem die Dinge untereinander kommunizieren und wir uns nicht mehr um die Datenverarbeitung kümmern müssen.

Wie bereits erwähnt, werden im Moment Daten zur Umweltqualität meist für eine relativ grosse Region an meist wenigen, statischen Punkten gesammelt. Diese Daten werden dann von dort an eine zentrale Datenbank gesendet, wo sie bearbeitet werden. Dies hat natürlich einige Nachteile. Einerseits müssen alle gesammelten Daten übertragen werden, da die Sensoren keine Möglichkeit haben, uninteressante auszufiltern. Dies erfordert eine relativ grosse Bandbreite, oder aber es werden weniger Daten gesammelt. Andererseits ist auch die Aussagekraft von wenigen Sensoren für ein grosses Gebiet relativ klein. Besser wäre es natürlich, auf kleinerem Raum mehr Sensoren zu haben. So wäre es zum Beispiel möglich, die Luftqualität für einzelne Strassen einer Stadt zu bestimmen. Auch ist die Nachbearbeitung im Labor mühsam. Es würde die Arbeit des Benutzers stark vereinfachen, könnten die Sensoren ihre Daten untereinander austauschen und die Bearbeitung der Daten würde direkt im Sensor erfolgen, so dass der Benutzer direkt die gewünschten Daten bekäme.

Würden dann solche smarte und verknüpfte Sensoren noch mit anderen Dingen, zum Beispiel Mobiltelefonen, kommunizieren, hätte man ein Internet der Dinge, das uns das Leben um vieles einfacher machen würde. So würden ganz neue Applikationen möglich, auch im Bereich des Umweltmonitorings.

Die Luftqualität könnte zum Beispiel für ein Gebiet viel genauer und detaillierter gemessen werden, zudem eventuell mit weiteren Daten von anderen Sensoren, wie zum Beispiel der Temperatur und Windrichtung, verknüpft werden und dies nicht erst im Labor, sondern bereits in den Knoten/Sensoren. So gewonnene Daten könnten von anderen Dingen wiederum genutzt werden, um zum Beispiel einer Person mitzuteilen, ob sie an diesem Tag Heuschnupfenmedikamente zu sich nehmen muss, oder ob die Pollenkonzentration in der Luft nicht genügend hoch ist, als dass sie eine allergische Reaktion auslösen könnte.

Ich möchte nun auf drei Beispiele für Umweltmonitoring aus den Papers [1], [2] und [3] eingehen. Danach werde ich im nächsten Kapitel auf Probleme eingehen, die in solchen Systemen auftreten.

3.1 Argus

Argus ist ein System, das mit normalen Videokameras als Sensoren Daten über die Küstenzone von Meeren sammelt. Die so gewonnenen Bilder werden mit (Bild-)Algorithmen aufbereitet. Diese bearbeiteten Daten können dann verwendet werden, um zum Beispiel Aussagen über die Strömung, die Wellenhöhe oder die Lage von Sandbänken unter Wasser zu treffen.

Ein Argussystem besteht normalerweise aus mehreren Kameras, die an einen Computer angeschlossen sind. Dieser verfügt über eine Verbindung zu einem zentralen Rechner, wo die Daten verarbeitet werden. Dort können auch Informationen von verschiedenen Argusstationen miteinander verknüpft werden.

Ein Beispiel für den Einsatz von Argus kann man an der Ostküste Australiens finden. Dort mündet der Fluss Tweed ins Meer. Der von ihm transportierte Sand lagert sich regelmässig in dessen Mündung ab und verunmöglicht oder gefährdet damit die Durchfahrt von Schiffen. Um diesem Problem Herr zu werden, wurde eine Pumpe installiert, die den Sand absaugt. Dieser Sand wird durch Rohre transportiert und kann an vier verschiedenen Stellen wieder ins Meer eingespielen werden. Um zu entscheiden, wohin der Sand geleitet werden soll, werden vier Argusstationen genutzt. Die so gewonnenen Daten helfen den Personen, die diese Entscheidung treffen müssen, bei ihrer Arbeit.

Vernetzte intelligente Sensoren würden natürlich so ein System weit vereinfachen und womöglich wäre am Ende gar keine menschliche Intervention mehr nötig. Argussysteme könnten auch direkt mit anderen Systemen kommunizieren. So wäre es denkbar, dass die Lage von Sandbänken direkt an passierende Schiffe weitergeleitet wird. Diese könnten dann, falls nötig, ihren Kurs automatisch korrigieren.

3.2 CitySense

CitySense ist noch in der Planungsphase und wird voraussichtlich 2011 umgesetzt werden. Es ist "[...] the vision for an urban-scale wireless networking testbed, called CitySense with the goal of supporting the development and evaluation of

novel wireless systems that span an entire city” [3].

Man möchte ungefähr hundert embedded PCs, welche mit Funk und verschiedenen Sensoren ausgerüstet sind, über ganz Cambridge verteilt auf Gebäuden und Strassenlaternen installieren. Die einzelnen Knoten sollen direkt vom Endbenutzer programmiert werden können.

Das Ziel von CitySense ist es nicht, öffentlichen Internet-Zugang zu gewährleisten. CitySense soll Forschungszwecken dienen, zum Beispiel im Bereich von neuen Wireless mesh routing Protokollen oder Sensornetzen in städtischen Umgebungen.

Die Knoten untereinander werden über ein Mesh-Netz verbunden sein. Das ist ein Netz, das sich selbst organisiert und ohne feste Infrastruktur, wie zum Beispiel WLAN Access Points, auskommt. Daten werden von Knoten zu Knoten bis zu ihrem Ziel weitergereicht. Das so entstehende Netz muss sich, wenn Knoten ausfallen oder sich bewegen und ausser Reichweite geraten, immer wieder neu organisieren.

Ein solches Netz wie CitySense in einer Stadt würde zum Beispiel einen Gesundheitsassistenten ermöglichen. Eine Person mit Atembeschwerden, die sich bei schlechter Luftqualität verschlimmern, könnte so einen Assistenten tragen. Dieser würde sich automatisch mit dem Mesh-Netz verbinden und von ihm Daten zur Luftqualität beziehen. Würden Schadstoffe in der Luft einen bestimmten Grenzwert überschreiten, könnte er selbständig einen Alarm auslösen.

3.3 MobGeoSen

MobGeoSen ermöglicht einer Person mit ihrem Mobiltelefon Daten zu ihrer direkten Umgebung aufzuzeichnen, wie zum Beispiel Luftverschmutzung, Temperatur oder Luftfeuchtigkeit, sowie Aufzeichnungen über ihre Gesundheit und Aktivitäten zu sammeln.

MobGeoSen besteht aus verschiedenen Komponenten: dem Mobiltelefon selbst, dessen internen Sensoren (zum Beispiel Kamera und Mikrofon) und externen Wireless-Sensoren (hier ein Daten-Logger und ein GPS-Gerät). Das GPS-Gerät ermöglicht es dem Benutzer seine Bilder und Daten mit Ortsangaben zu verknüpfen. Mit dem Daten-Logger können bis zu vier weitere Sensoren mit dem Telefon verbunden werden.

Durch das Verknüpfen von Daten, die Einzelpersonen mit Hilfe von MobGeoSen gesammelt haben, könnte so eine sehr aufschlussreiche und genaue Karte von Daten zur Umgebung zusammengetragen werden. Denkbar wäre eine Karte, welche den Kohlenmonoxid-Gehalt der Luft für einzelne Strassen oder Quartiere einer Stadt, in Abhängigkeit der Tageszeit, sehr detailliert darstellt. Diese Daten könnten sich Benutzer dann wieder auf ihr Mobiltelefon laden. So könnte ein MobGeoSen-Mobiltelefon zum Beispiel eine Route nach Kriterien der besten Luftqualität der verschiedenen Strassen planen.

Ebenso wäre es denkbar, dass sich ein MobGeoSen-Mobiltelefon bei CitySense anmeldet, falls es in seiner Reichweite ist und so zwischenzeitlich als Knoten

fungiert, der ebenfalls Daten sammelt. Somit könnten die Daten, welche von CitySense-Knoten gesammelt wurden, mit zusätzlichen Daten der MobGeoSen-Geräte erweitert werden, wie zum Beispiel dem Lärm.

Auch wäre ein MobGeoSen-Mobiltelefon als Gesundheitsassistent, wie oben beschrieben, denkbar.

4 Schwierigkeiten

Das Internet der Dinge ermöglicht höhere Dienstleistungen. Die Probleme sind aber weiterhin auf den tieferen Ebenen zu finden. Schliesslich kann die höhere Dienstleistung nicht existieren, falls die tieferen Ebenen nicht zur Verfügung stehen. Schlägt zum Beispiel etwas in der Kommunikation der Dinge untereinander fehl, so kann die höhere Dienstleistung womöglich nicht mehr angeboten werden. Auf einige Probleme (der tieferen Ebene) möchte ich hier eingehen. Die Probleme werden die Beispiele aus dem vorherigen Kapitel betreffen. Dazu möchte ich dann jeweils erklären, wie in dem speziellen Fall das Problem gelöst wurde.

4.1 Argus

Datenmenge: Wenn die Bandbreite stark limitiert ist oder sehr viele Daten gesammelt werden, kann man womöglich nicht alle Daten unbearbeitet übertragen.

In solchen Fällen muss man die Daten reduzieren und nur diese behalten oder senden, die für eine Anfrage auch von Interesse sind.

In Argus werden dazu gewisse Techniken verwendet, um mehrere Bilder in einem zusammenzufassen, das dann über bestimmte Aspekte Auskunft gibt. Zwei solche Bildtypen wären (siehe dazu auch Fig.1):

- Time exposure image: Für diese Art von Bild wird eine bestimmte Zeit lang gefilmt und danach die einzelnen Bilder des Filmes wie folgt bearbeitet: Für jedes Pixel des Endbildes wird das Mittel der Helligkeit dieses Pixels auf den einzelnen Bildern des Filmes genommen. Als optimal hat sich eine Aufnahmezeit von 10 Minuten herausgestellt [1]. Diese Bildart ist sehr gut geeignet für die Bestimmung der Lage von Sandbänken unter Wasser, Fahrwasserrinnen oder der Küstenlinie.
- Variance image: Ein Pixel dieses Bildes setzt sich aus dem Unterschied der Helligkeit dieses Pixels auf den einzelnen Bildern des Filmes zusammen. Diese Bilder sind nützlich, um Regionen in denen viel passiert (zum Beispiel dort wo sich Wellen brechen) von Regionen in denen nicht viel passiert (zum Beispiel dem Strand) zu unterscheiden.

Für einige Informationen ist es allerdings gar nicht nötig, das ganze Bild zu betrachten. Ein einzelnes Pixel reicht zum Beispiel aus, um den zeitlichen Abstand aufeinanderfolgender Wellen zu bestimmen.



Fig. 1. Palm Beach, Australia: (a) snapshot, (b) 10-minute time exposure, and (c) variance image. [1]

Komplexität: Durch die zunehmende Anzahl Sensoren wird ein System automatisch auch komplexer in der Wartung und schwieriger zu überschauen und zu verstehen. Verarbeiten die Sensoren die Daten allerdings gleich bei sich und kommunizieren untereinander, so wird die Arbeit für den Endbenutzer leichter.

In Argus könnte dies zum Beispiel wie folgt aussehen:

- Smarte Kameras: Das Ziel wäre, dass die Kameras selber Kenntnisse über ihre in- (z.Bsp. Linsenverkrümmung) und extrinsische (z.Bsp. Position der Kamera) Parameter haben. Im Moment werden alle diese Informationen in einer zentralen Datenbank gespeichert und, zumindest einige, Berechnungen erst ausgeführt, nachdem die Daten ins Labor geschickt wurden, mit den Hilfsdaten (z.Bsp. Position der Kamera), die dort in der Datenbank vorhanden sind. Zumindest einige dieser Berechnungen könnten allerdings direkt in die einzelnen Kameras ausgelagert werden. So kann zum Beispiel die Korrektur der Linsenverkrümmung relativ einfach direkt in der Kamera ausgeführt werden.
- Full onboard analysis: Das Ziel wäre, die ganze Datensammlung und -analyse vollständig in die einzelnen Stationen auszulagern. Anstatt Rohdaten würden direkt Variablen, wie zum Beispiel der Wellenabstand oder die Strömungsgeschwindigkeit, als Funktion der Distanz zum Ufer, sowie aufbereitete Karten der Wassertiefe zurückgegeben. Der Benutzer müsste sich also um keine Datenbearbeitung mehr kümmern, sondern nur noch die richtigen Schlüsse aus den bereits aufbereiteten Daten ziehen. Vernetzte intelligente Sensoren würden zudem Auswertungen, die mehrere Argusstationen miteinbeziehen, ermöglichen und vereinfachen.

4.2 MobGeoSen

Visualisierung: Die Visualisierung von Daten ist nicht immer ganz einfach.

Es gilt verschiedenste Informationen zusammenzufassen. Hier sind das Daten der verschiedenen Sensoren, örtliche Information und womöglich

Bilder und Notizen.

Die gesammelten Daten werden dazu auf dem Mobiltelefon zuerst in ein Textfile gespeichert. Die so gespeicherten Daten können relativ einfach direkt auf dem Telefon mittels Diagrammen oder Kurven dargestellt werden. Allerdings fehlen in diesem Fall die zusätzlich gesammelten Informationen, wie zum Beispiel die Ortsinformation.

Die Daten aus dem Textfile (Sensorwerte, GPS-Daten, vom User gesetzte Wegpunkte, Photos und Notizen) können, nachdem die Datenaufzeichnung abgeschlossen wurde, aber auch in ein KML-File gespeichert werden ("Keyhole Markup Language, kurz KML, aktuelle Version 2.2, ist ein Austauschformat für die Client-Komponente des Programms Google Earth." [5]). Dieses KML-File kann dann auf einen Computer geladen werden und dort mittels Google Earth angezeigt werden.

4.3 CitySense

Physische Umgebung: Die Knoten werden zu jeder Jahreszeit den verschiedensten Verhältnissen ausgesetzt sein. Auch muss damit gerechnet werden, dass Knoten (mutwillig) beschädigt werden und deswegen ausfallen können.

Um dem Wetter standzuhalten, sind zum Beispiel wasserdichte Umschliessungen nötig und eventuell eine Heizung für den Winter oder eine extra Entlüftung für im Sommer, wenn es besonders heiss ist.

Die Knoten in CitySense werden laut Tests keine Heizung benötigen, da durch die Stromzufuhr und die Rechenleistung automatisch genug Wärme produziert wird. Im Sommer werden weitere Tests stattfinden, um zu sehen, ob ein Entlüften notwendig sein wird.

Auf den möglichen Ausfall von Knoten werde ich später zurückkommen.

Kommunikation und Sicherheit: In einer städtischen Umgebung muss mit vielen anderen Wireless-Netzen gerechnet werden, was unter Umständen stören kann.

Dass die Sensoren auf Strassenlaternen montiert werden sollen, bringt Vorteile mit sich. Zum einen werden die Knoten mit grosser Wahrscheinlichkeit untereinander Sichtkontakt haben und zum anderen ist 10 Meter über Boden die Reflexion des Signales am Boden oder die Störung durch andere WLANs geringer, was den Empfang verbessert.

Trotzdem wird CitySense bei weitem nicht das einzige Netz in der Stadt sein. Eine Kommunikation der Knoten untereinander sollte möglich sein, trotz grossen Schwankungen in der zur Verfügung stehenden Bandbreite, Veränderungen in der Umgebung oder Störungen durch andere Netze. Deswegen ist es von Vorteil omnidirektionale Funkverbindungen/Antennen mit relativ grosser Reichweite zu verwenden. Allerdings steigert dies natürlich wiederum ihre Sensibilität gegenüber fremdem Netzverkehr.

In Versuchen in Cambridge wurden zwei Knoten auf dem selben Dach in einer Entfernung von ca. 40 Metern aufgestellt. Die Bandbreite betrug

zwischen 20 Mbps und (nur!) 1 Mbps [3].

Damit beim Ausfall eines Knotens das Netz nicht unterbrochen wird, wird der Abstand zwischen zwei Knoten nicht grösser als die halbe Reichweite des Funkgerätes gewählt. Damit ist sichergestellt, dass zwei Knoten verbunden bleiben, auch wenn ihr Zwischenknoten ausfällt.

Auch Netzwerksicherheit ist ein wichtiger Aspekt in einer städtischen Umgebung mit vielen privaten Wireless-Netzen. Dazu will CitySense einerseits die Kommunikation auf dem Link Layer verschlüsseln und zusätzlich sollen die Anwendungen ein sicheres Transportprotokoll verwenden. Die Schlüssel zur Ver-/Entschlüsselung sollen ausserdem von Zeit zu Zeit ausgetauscht werden.

Netzwahl: Bei einem Netz mit vielen Knoten, die schwer zu erreichen sind (Strassenlaternen), möchte man möglichst wenig Wartungsaufwand haben. Sollte ein Knoten ganz ausfallen, kann man ihn womöglich weder reparieren noch ersetzen. Ebenso kann es von Vorteil sein, auf zentrale Anlaufstellen, wie WLAN Access Points, zu verzichten, da um sie herum relativ viel Netzverkehr herrschen wird und falls sie ausfallen, womöglich ein Grossteil der Knoten nicht mehr mit dem Netz verbunden sein wird. Deshalb ist es sinnvoll für solch eine Applikation ein sich selbst organisierendes Netz, zum Beispiel ein Mesh-Netz, das ohne zusätzliche Infrastruktur auskommt, zu wählen.

Allerdings hat ein Mesh-Netz auch einige Nachteile: Die Routingverfahren sind komplizierter. Jeder Knoten muss eine Routingtabelle speichern. Da jeder Knoten zum Routing gebraucht wird und, falls er ausfällt, das Netz neu organisiert werden muss, sollte er möglichst eingeschaltet bleiben, was, falls die zur Verfügung stehende Energie beschränkt ist, sich negativ auf seinen Energieverbrauch auswirken kann.

Energieverbrauch: Oft ist die Energie, die einem Knoten zur Verfügung steht, (stark) beschränkt. Hier ist dies allerdings nicht relevant, da die Knoten auf Strassenlaternen und Gebäuden montiert werden, wo eine Stromversorgung vorhanden ist.

Neuprogrammierung, Wartung und Fehlerbehebung: Über 100 Knoten in einem konsistenten Softwarezustand zu halten, erfordert einiges an Arbeit und bringt einige Schwierigkeiten mit sich. Was passiert mit Knoten, bei denen ein Fehler aufgetreten ist, was wenn Verbindungen unterbrochen wurden, was wenn viele User oft updaten, wie beeinflussen Software-Updates die Knotenverlässlichkeit? [3]

- **Wartung:** Damit sichergestellt ist, dass Experimente Updates, etc. nicht verhindern, soll jeder Knoten über zwei Funkgeräte verfügen. Wobei Eines davon zum Management-Netz verbunden bleiben soll.

- Fehlerbehebung Wenn ein Knoten abstürzt, muss sichergestellt werden können, dass er automatisch reboottet. Für das wird man einerseits auf der Softwareseite einen "Wachhund" [3] einbauen, der zum Beispiel merkt, wenn er keine Verbindung mehr zum Management-Netz hat, und andererseits auf der Hardwareseite einen Timer, der jeden Tag einmal den Knoten, unabhängig seines Zustandes, neu startet und damit in einen "guten Zustand" überführt. Dabei sollte natürlich darauf geachtet werden, dass sich nicht alle Knoten gleichzeitig neu starten.

Knoten, die sich neu starten, werden von einer separaten Partition booten, auf der ein "golden image" [3] aufbewahrt wird. In so einem Fall wird dem Knoten nur erlaubt, sich mit dem zentralen Server für "administrative control" [3] zu verbinden, jegliche Benutzerapplikation oder Kommunikation mit einem Benutzer wird unterbunden.

Diese Lösungsidee hat zur Folge, dass die Knoten etwas weniger zur Verfügung stehen werden, dafür ist sichergestellt, dass abgestürzte Knoten nach einem Fehler wieder verfügbar sein werden.

- Neuprogrammierung: Häufige Softwareupdates haben viel Netzwerkverkehr zur Folge, was die Kommunikation für Experimente oder zu Wartungszwecken beeinflussen könnte, zum Beispiel durch Störungen. Um viel Netzwerkverkehr aufgrund von Softwareupdates zu verhindern, will man mit Spanning Trees und rsync arbeiten: ein zentraler Server soll Updates sammeln und dann (zum Beispiel alle zwei Tage) zu gewissen (vorher bestimmten) Knoten senden und diese verteilen dann das Update an weitere Knoten weiter, die es wiederum an ihre Kinder weitergeben. Rsync ist ein Tool zur Synchronisation von einem File auf verschiedenen Rechnern. Übertragen werden nur tatsächlich geänderte Teile des Files. (Weitere Information zu rsync unter

http://www.samba.org/rsync/tech_report)

Jedes Update wird eine Nummer, aus einer monoton ansteigenden Folge, erhalten. So weiss jeder Knoten, ob er dieses Update schon erhalten hat, oder noch nicht.

Es ist fragwürdig, ob man dem User in seiner Applikation alles erlauben soll, oder ob man nicht besser ein API zur Verfügung stellt, das die wichtigsten Funktionen enthält (Position eines Sensors, Sensorwerte, Netzinformationen, etc.). Dieses Modell würde eine gewisse Kontrolle über die Applikationen liefern, die Intendanten von CitySense sind sich jedoch noch nicht sicher, ob sie die Art von Applikationen so stark einschränken wollen, oder ob eventuell eine Mischform möglich wäre, wo je nach Art der Applikation verschiedene (oder gar keine) APIs zur Verfügung stehen.

5 Zusammenfassung

Die Fortschritte bei der Hardware und in der Kommunikationstechnologie ermöglichen Systeme, die immer näher an die Idee des Internets der Dinge kommen, viel genauere Aussagen liefern als bisherige Systeme und dem Benutzer die Arbeit erleichtern oder gar abnehmen.

Viele Probleme liegen aber weiterhin auf tieferen Ebenen. Die höheren Dienste sind davon abhängig, dass die tieferen Ebenen korrekt funktionieren und auch bei einem Ausfall (z.Bsp. einzelner Knoten) der Dienst aufrecht erhalten werden kann.

Durch das Lösen dieser Probleme für einzelne Systeme werden so sehr mächtige Applikationen ermöglicht.

References

1. Rob Holman, John Stanley, and Tuba Özkan-Haller. Applying Video Sensor Networks to Nearshore Environment Monitoring. IEEE Pervasive Computing, Oktober-Dezember 2003.
2. Eiman Kanjo, Steve Benford, Mark Paxton, Alan Chamberlain, Danae Stanton Fraser, Dawn Woodgate, David Crellin, and Adrain Woorland. MobGeoSen: facilitating personal geosensor data collection and visualization using mobile phones. Springer, Oktober 2007.
3. <http://www.eecs.harvard.edu/~mdw/papers/citysense-techrept07.pdf>, 9.1.2008
4. http://de.wikipedia.org/wiki/Internet_der_Dinge, 9.1.2008
5. http://de.wikipedia.org/wiki/Keyhole_Markup_Language, 12.1.2008
6. Alan Mainwaring, Joseph Polastre, Robert Szewczyk, David Culler, John Anderson. Wireless Sensor networks for Habitat Monitoring.