

Die technische Basis für M2M und das Internet der Dinge

Friedemann Mattern

Institut für Pervasive Computing, ETH Zürich

Vielen Dank, Herr Eberspächer, für Ihre freundliche Einführung! Ich möchte dazu anmerken, dass ich den Begriff „Internet der Dinge“ im Grunde genommen nicht erfunden habe; ich habe lediglich vor elf Jahren den deutschen Terminus eingeführt [Mat02], der als „Internet of Things“ im Englischen schon vorher auftauchte [Sch02, Ash09]. Bereits 1999 umschrieb Neil Gershenfeld vom MIT dies übrigens, ohne es beim heutigen Namen zu nennen, in poetisch-prophetischen Worten so: „Es kommt mir so vor, als sei das rasante Wachstum des WWW nur der Zündfunke einer viel gewaltigeren Explosion gewesen. Sie wird losbrechen, sobald die Dinge das Internet nutzen“ [Ger99].

Meine Damen und Herren, ich bin gebeten worden, heute einmal nicht über Visionen oder die Zukunft zu sprechen, sondern ganz nüchtern über die Technik und die Standards, die M2M – also „Machine to Machine“ – und dem Internet der Dinge zugrunde liegen sowie die sich daraus ergebenden Perspektiven. Beginnen wir mit der Bedeutung und der Historie der beiden Begriffe.

Ursprung und Bedeutung von „M2M“ und „Internet der Dinge“

Wir sehen in Bild 1 zunächst, was „Google Trends“ (www.google.com/trends) zu den beiden Begriffen meldet:

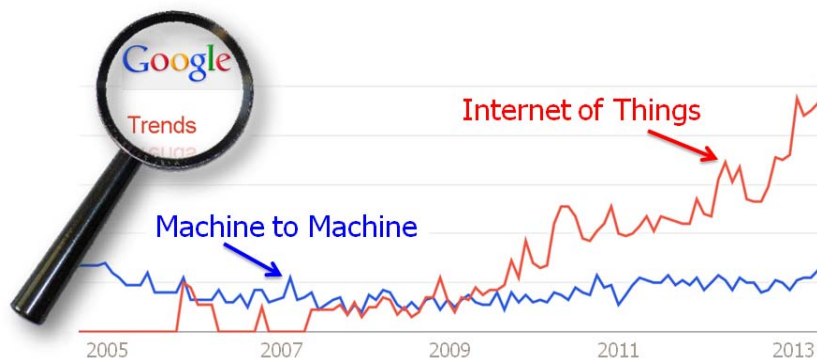


Bild 1: Popularität der Google-Suchbegriffe „Machine to Machine“ und „Internet of Things“.

Offenbar spricht man schon länger von M2M, wohingegen „Internet of Things“ (abgekürzt „IoT“) erst ab 2009 richtig populär wurde, auch wenn bereits 2008 die erste wissenschaftliche Konferenz unter dieser Bezeichnung stattfand [Flo08]. Man macht heute oft keinen großen Unterschied mehr, ich möchte allerdings zunächst doch die beiden Begriffe erst einmal auseinanderhalten. M2M kommt an sich aus der Telemetrie, ein ganz altes Gebiet, wo man sich

schon vor über 100 Jahren bemüht hat, mit elektromechanischen Mitteln die Messwerte mechanischer bzw. analoger Wetterstationen automatisch in Morsezeichen zu übersetzen und dann über Telegraphenleitungen zu versenden. Erst seit ca. 15 Jahren, erstmalig übrigens bei Siemens in München, wird die heutige Form praktiziert – man nutzt quasi Mobiltelefone in abgespeckter und abgewandelter Form, um Telemetrieaufgaben zu erledigen. Das hat dann im Telekommunikationsbereich neben der klassischen „Human-to-Human Communication“ zur „Non-Phone“, oder eben der Machine-to-Machine Communication geführt (Bild 2 links).

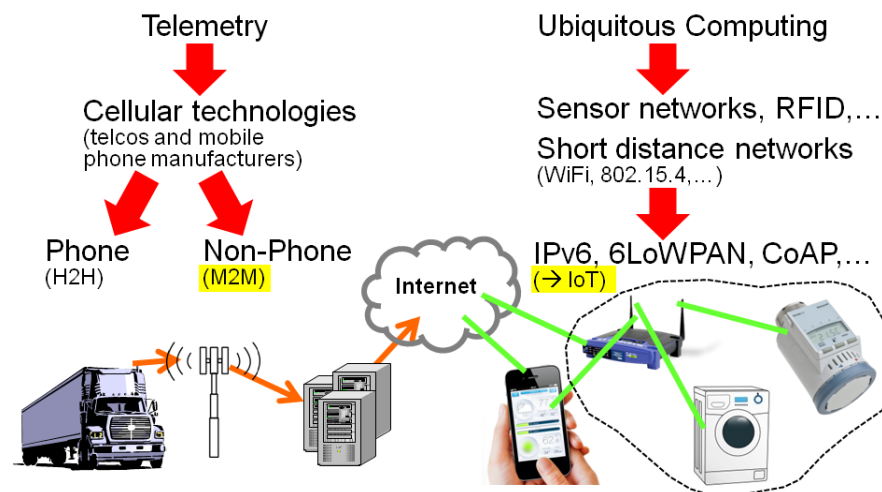


Bild 2: Die historischen Wurzeln von M2M und IoT.

Dem Internet der Dinge (Bild 2 rechts) liegt an sich eine ganz andere Historie zugrunde [MaF10]. Wie Herr Eberspächer vorhin bereits schilderte, hatte Marc Weiser Anfang der 1990er Jahre diese – damals noch sehr „akademische“ – Vision vom „Ubiquitous Computing“ [Wei91]. Die weitere Entwicklung verlief dann zunächst eher unspektakulär – in nüchternen Technikbegriffen so ausgedrückt [Mat05]: Sensornetze, RFID und natürlich die ganzen funkbasierten Kommunikationsmöglichkeiten im Kurzstreckenbereich. Mobiltelefone oder gar Smartphones spielten anfangs übrigens keine Rolle. Auch die eigentliche Internet-Technologie war beim „Ubiquitous Computing“ weitgehend außen vor, etwas genierlich versahen Elgar Fleisch und ich unser 2005 herausgegebenes Buch „Das Internet der Dinge“ [FM05] damals noch mit dem Untertitel „Ubiquitous Computing und RFID in der Praxis“. Tatsächlich verstand man in diesem Kontext den Begriff „Internet“ zunächst wohl eher als eine Metapher für eine wie auch immer geartete Kommunikation in smarten Umgebungen.

Das änderte sich dann allerdings schnell. Mit der stürmischen Entwicklung der letzten Jahre tauchten mit Bezug auf das Internet der Dinge neue Schlagworte und Abkürzungen wie IEEE 802.15.4, 6LoWPAN oder CoAP auf, die Kommunikationsprotokolle und Mechanismen bezeichnen, womit das Internet der Dinge im engeren Sinne Wirklichkeit werden soll. Auf diese Begriffe und die zugehörigen Techniken und Konzepte werde ich später noch auszugsweise eingehen. Jedenfalls hat das Internet der Dinge gegenüber M2M seinen Ursprung in einer ganz anderen Welt – bei M2M steht der zu unterstützende und zu optimierende Geschäftsprozess im Vordergrund, beim „Ubiquitous Computing“ war immer der individuelle Mensch der Bezugspunkt. Statt dem so wichtigen „tele“ bei M2M, das die technische Grundlage für die

Globalisierung bildet, schimmert bei den historischen Wurzeln des Internet der Dinge eine eher humanzentrierte, lokale Welt durch. Wo Menschen aktiv sind, gibt es heute aber auch WiFi, und wo WiFi ist, kann man dies auch für die „smarten“ Gegenstände [Mat03] in der Umgebung der Menschen nutzen und die Dinge damit an das Internet anschließen. Aber damit greife ich vor.

Inzwischen beobachtet man eine gewisse, ich möchte nicht direkt sagen Konvergenz, aber doch Annäherung der beiden Begriffe „M2M“ und „Internet der Dinge“ – beide bilden die Basis für ähnliche und sich mittlerweile überlappende Szenarien und Geschäftsmodelle. Und dennoch ist die zugrundeliegende Technik, insbesondere die Kommunikationstechnik, im Grunde genommen unterschiedlich. Ein intelligenter Stromzähler etwa kann als smartes „Ding“ im Haushalt angesehen werden, welches für die Bewohner per WiFi (oder einem anderen etablierten Funkstandard für lokale Bereiche wie z.B. ZigBee) und Internetstandards zugänglich ist, oder aber als „Maschine“, die mit dem viel größere Distanzen überbrückenden Mobilfunknetz (und Kommunikationsprotokollen wie GPRS) dem Elektrizitätsversorger zu Diensten steht.

Die Frage ist aber auch nicht so sehr, ob es sich bei „M2M“ und „IoT“ nun um eine begriffliche oder gar technische Konvergenz handelt, sondern es geht in pragmatischer Hinsicht einfach darum, welche Technologie für ein bestimmtes Szenario oder ein bestimmtes Geschäftsmodell die am besten geeignete ist – und für uns ist es natürlich auch spannend zu sehen, wie die jeweiligen Technologien mit Blick auf die stetig voranschreitende Informatisierung der Alltagswelt weiterentwickelt werden können.

Technologische Herausforderungen

Bei der dem Internet der Dinge zugrundeliegenden Technik gibt es eine ganze Reihe von Herausforderungen. Um nur einige aufzuführen:

- Kommunikation
- „Arrive and operate“
- Energieversorgung
- Sicherheit (Privatsphärenschutz, Datensicherheit, Betriebssicherheit / safety etc.)
- Zuverlässigkeit, Ausfallresistenz
- Skalierbarkeit
- Interoperabilität, Standardkonformität
- ...

Sicherlich ist das zunächst Interessante und Relevante die Kommunikation. Wie billig, schnell und wie viel kann unter welchen Voraussetzungen kommuniziert werden? Wie viel Energie wird dafür benötigt? Ist vielleicht ein eingebetteter WiFi-IPv6-HTTP-Protokollstack die Lösung? Auf Kommunikationsaspekte werde ich weiter unten etwas genauer eingehen. Zum Punkt „arrive and operate“ aus obiger Liste: Dies nannte man früher „plug and play“; es hat damit folgende Bewandnis: Fast alle Dinge, die wir besitzen, sind ja im Prinzip mobil, die meisten von uns können bestenfalls auf eine einzige Immobilie stolz sein. Nun kommt also ein mobiles, smartes Ding an und soll hier bei mir sofort funktionieren, sich also in der neuen

Umgebung einnisten. Nur hat man jetzt keinen „plug“ mehr, weil man drahtlos und damit „unplugged“ ist, und man will mit dem neuen Objekt nicht spielen, sondern ernste Dinge tun. „Arrive and operate“ ist eine essentielle Voraussetzung für viele interessante Szenarien im Bereich des Internet der Dinge, vor allem für Anwendungen in Massenmärkten. Die Herausforderung besteht darin, dies nicht nur „irgendwie“ zu ermöglichen, sondern das ganze muss sicher sein, soll interoperabel bleiben, skalierbar sein, die Standards müssen eingehalten werden – und Standards gibt es viele!

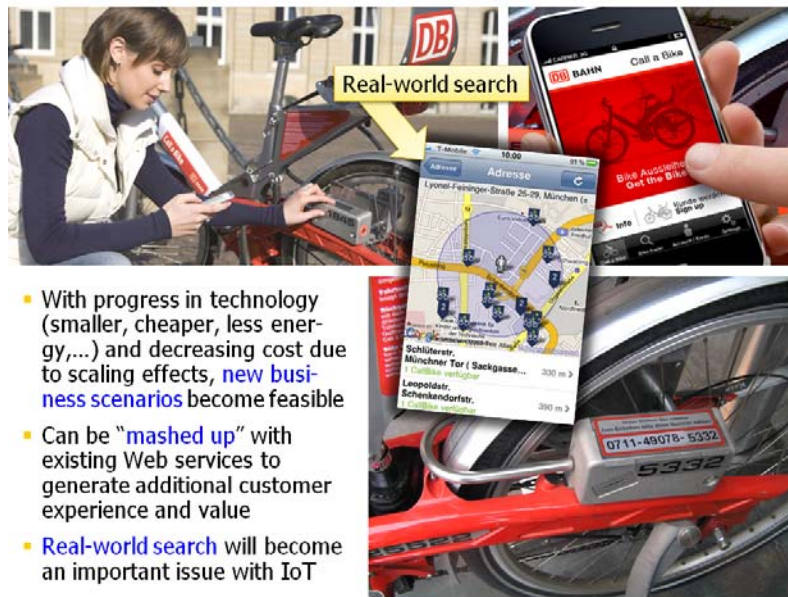
Was sticht noch aus obiger Aufzählung hervor? Energie ist natürlich auch ein sehr interessantes und spannendes Thema. Wie kommt man dran? Wie viel braucht man, damit alles noch verlässlich funktioniert? Wäre „energy harvesting“ eine Lösung? Auf die Punkte Sicherheit, Zuverlässigkeit und Interoperabilität möchte ich hier nicht weiter eingehen, wie man sowieso obige Liste noch fortsetzen könnte (z.B.: wie interagiert man mit smarten Dingen oder wie findet man das richtige smarte Ding in der physischen Welt?). Eine eingehendere Diskussion der vielfältigen technischen Gesichtspunkte erfordert wohl einen ganzen Vorlesungszyklus, und die Herausforderungen werden sicherlich noch für viele Dissertationen gut sein. Daher will ich mich in diesem Beitrag auf die Kommunikation und einige damit zusammenhängende Aspekte konzentrieren.

M2M: Anwendungsbereiche, Technik, Business

Betrachten wir zunächst die klassische M2M-Domäne. Damit verbunden sind typischerweise Business-to-Business-Anwendungen in Bereichen wie Flottenmanagement, Facility Management oder Warenlogistik [BMWi11], letztlich mit dem Ziel, Prozesse zu automatisieren und zu optimieren oder die Sicherheit in einem breit verstandenen Sinne zu erhöhen. Zur Technologie habe ich schon etwas salopp die abgespeckten Mobiltelefone erwähnt, teilweise umgebaut und mit neuen Schnittstellen zu Sensoren und externen Datenquellen versehen, oft auch mit GPS und Bewegungssensoren etc. ausgestattet, was wir ja auch von Smartphones her kennen. Die SIM-Karte spielt dabei eine zentrale Rolle, natürlich auch für das Geschäftsmodell der Telekommunikationsdiensteanbieter (kurz: Telcos) – denn wer verdient eigentlich an diesem Geschäftsprozess, für den der Mobilfunkprovider die SIM-Karte zur Verfügung stellt? Nur der Kunde oder kann man als Telco mit generischen Mehrwertdiensten daran teilhaben? Die starke Bindung an SIM-Karten und die daran gekoppelte proprietäre Kommunikationsinfrastruktur stellen für die Telcos auch bezüglich des Lock-in-Effektes Chance und Hindernis zugleich dar.

Indem die Hardware im Sinne des verallgemeinerten mooreschen Gesetzes im Laufe der Zeit quasi automatisch billiger, kleiner und besser wird, werden in technischer und ökonomischer Hinsicht nach und nach auch neue Geschäftsfelder für M2M möglich, bei denen es nicht mehr nur um Geräte und Maschinen geht, die besonders wertvoll oder kritisch sind, und wo in entsprechenden Szenarien dann Allerweltsdingen und „Allerweltsnutzern“ eine explizite Rolle zukommt. Als Beispiel mag ein öffentliches Fahrradverleihsystem (Bild 3, www.inside-m2m.de/deutsche-bahn.html) dienen. Mit all‘ dem, was das „Web-Ökosystem“ inzwischen an allgemein nutzbaren Tools und Möglichkeiten bietet, kann man das Ganze – zumindest als Proto-

typen – in Form von sogenannten Mashups relativ einfach zusammenbauen; in diesem Fall z.B. angedeutet durch die Einbindung von Google Maps.



- With progress in technology (smaller, cheaper, less energy,...) and decreasing cost due to scaling effects, **new business scenarios** become feasible
- Can be "mashed up" with existing Web services to generate additional customer experience and value
- **Real-world search** will become an important issue with IoT

Bild 3: Öffentliche Fahrradverleihsysteme als neue M2M-Szenarien.

In Klammern möchte ich passend zu diesem Beispiel noch anmerken, dass mit dem Internet der Dinge noch etwas anderes wichtig werden wird, nämlich das Suchen in der Realwelt. Natürlich stellt das eine phantastische Herausforderung für die Informatik dar, wenn man nicht mehr nur weitgehend statische und als Bitsequenzen vorliegende Webseiten mithilfe eines Index sucht, sondern physische Dinge und momentane Zustände der Realwelt [MDT12, Roe10].

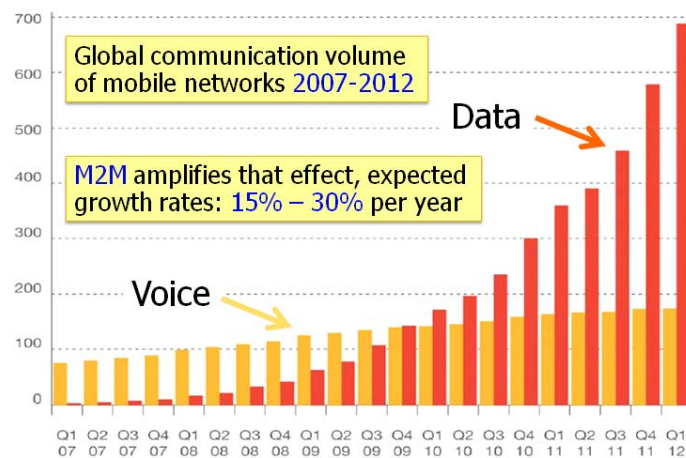


Bild 4: Wachstum der Sprach- und Datenkommunikation in mobilen Netzen.

Die Branche ist sich einig, dass das simkartenbasierte M2M-Geschäft weiter zunehmen wird, die Wachstumsraten dürften in den nächsten Jahren zwischen 15 und 30% liegen. Entsprechend wird das über die Mobilnetze versendete Datenvolumen anwachsen, der Trend ist unver-

kennbar (Bild 4). Die reine Sprachkommunikation ist dabei kaum mehr steigerbar, weil dazu Menschen gehören; andererseits aber können Maschinen immer schneller miteinander reden.

Wie sieht der weitere Fortschritt im Bereich des klassischen M2M aus? Natürlich werden die Hardwaremodule für die Mobilfunkkommunikation kleiner und billiger. Bild 5 zeigt links ein Beispiel, 2x2 cm im Quadrat und 3 mm hoch – mittlerweile gibt es schon Nachfolgesysteme davon. Allerdings täuscht das ein wenig, weil man noch eine externe SIM-Karte anschließen muss, eine Antenne benötigt und die Energie zur Verfügung stellen muss. Aber das eigentliche Kommunikationsmodul ist tatsächlich recht klein.

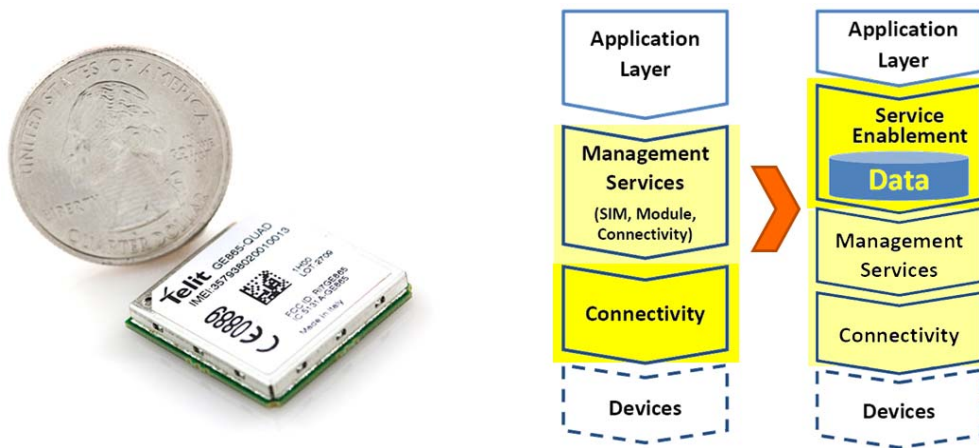


Bild 5: Links: ein M2M-Modul; rechts: M2M-Business-Säulen – jetzt und zukünftig.

Ganz generell kann man sagen, dass über die nächsten Jahre wahrscheinlich weiterhin das Energieproblem ein limitierender Faktor für viele interessante Szenarien sein wird, bei denen man eigentlich viele Jahre lang mit einer einzigen Knopfzelle auskommen möchte. Über lange Sicht werden Batterien im Mittel vielleicht 3 bis maximal 5% pro Jahr effizienter (bezogen auf Volumen oder Gewicht), wobei dies allerdings kaum kontinuierlich verläuft: Bisher gab es alle paar Jahre einmal einen Technologiesprung, dazwischen blieb die Energiedichte weitgehend konstant. „Energy harvesting“, also das „Absaugen“ oder Nutzen von Umweltenergie, ist (relativ zum Energiebedarf gegenwärtiger Hardware) meistens noch zu wenig ergiebig, auch wenn diese Technik bereits erfolgreich in speziellen Anwendungen eingesetzt wird (siehe z.B. die batterielose Funksensorik von EnOcean, www.enocean.com). Die einzige realistische Hoffnung ist, dass man aufgrund des mikroelektronischen Fortschritts immer weniger Energie für die Prozessoren, aber auch die Kommunikation, benötigt und dass es durch alle erdenklichen Tricks noch besser gelingt, möglichst lange in einem energiesparenden Tiefschlaf zu verharren, ohne dass Nachrichten oder wesentliche Ereignisse verpasst werden oder zu spät reagiert wird.

Der technische Fortschritt ermöglicht im M2M-Bereich jetzt auch sogenannte „embedded SIMs“, die noch kleiner, im Gegenzug allerdings nicht mehr einfach austauschbar sind (Bild 6, www.t-mobile.de/T-D1/img/display_image/0,3465,211955,00.jpg). Diese haben Vor- und Nachteile. Der entscheidende Vorteil ist, dass das gesamte M2M-Modul kleiner und stabiler wird. Der Nachteil ist, dass eine wesentliche Flexibilität, die man von der Mobiltelefonie her kennt, durch das Einlöten verloren geht: Bisher gehörte das Handset dem Benutzer und die auswech-

selbare SIM-Karte dem Provider, wodurch sich zum Vorteil der Kunden zwei flexibel kombinierbare Geschäftsfelder getrennt entwickeln konnten. Eine softwaremäßige Umkonfiguration der SIM-Karte durch „remote provisioning“ ist in dieser Hinsicht kein vollwertiger Ersatz für den physischen Austausch. Aber das ist eine andere Geschichte.



Bild 6: Mini-SIM-Karte und embedded SIM.

Es werden gegenwärtig auch einige Diskussionen zu radikaleren Technikentwicklungen geführt, man sollte hier aber ein bisschen vorsichtig sein – manche sind vielleicht eher bloße Hoffnungen als realistische Aussichten. So existieren Überlegungen (z.B. Sigfox, Frankreich), ein dediziertes funkzellenbasiertes M2M-Netz aufzubauen, das nicht mit der Sprachkommunikation oder anderen Datendiensten konkurriert – dies könnte dann auf die spezifischen Charakteristika von M2M-Datenverkehr hin optimiert werden. Ferner gibt es Pläne (z.B. Neul, Cambridge, UK), die durch die Digitalisierung des Fernsehens frei gewordenen Frequenzbänder für M2M zu gebrauchen oder generell derzeit ungenutztes Spektrum („white space“) zu verwenden, dabei auf tiefere Frequenzen auszuweichen, um mit weniger Funkzellen auszukommen und auch Gebäude besser durchdringen zu können. Spezielle Kommunikationsprotokolle, die besonders auf einen (für M2M-Anwendungen meist ausreichenden) sehr geringen Durchsatz und auf Energieeffizienz optimiert sind, stellen eine weitere Entwicklungsrichtung dar. Kann man in einem „One-Way-Modus“ auf Acknowledgements verzichten, können ebenfalls sehr energieeffiziente Protokolle eingesetzt werden – allerdings hat man dann natürlich keine Gewissheit, ob ein Datenpaket angekommen ist. Verkündungen wie „man kommt mit hundert Mal weniger Energie als bisher aus, man muss nur neue Netze aufbauen“ sollte man mit gesunder Skepsis begegnen – Wunder gibt es in der Physik selten, und einen „free lunch“ kann man eigentlich auch nicht erwarten: Verbesserungen in der einen Hinsicht gehen meistens auf Kosten eines (evtl. allerdings nicht ganz so relevanten) anderen Aspektes. Aber man darf natürlich gespannt bleiben und wird sehen, wohin solche Überlegungen und Bestrebungen führen werden.

Die generelle Frage im klassischen M2M-Business ist, wie und wohin sich die zugehörige Branche entwickeln wird. Bild 5 rechts zeigt in der linken Säule schematisch die heutige Situation: Zwischen den M2M-Geräten und den Anwendungen liegt die „Connectivity“. Das ist das, was die Telcos heute machen und womit sie bisher ihr Geld verdient haben – dazu kommen dann noch einige Basisdienstleistungen rund um die Verwaltung der SIM-Karten. Natürlich fragt man sich bei den Telcos zunehmend, weshalb andere viel mehr Umsatz machen, nämlich diejenigen, die Clouds betreiben und die Datenspeicherung sowie höherwertige Services anbieten, und ob man das nicht auch selbst tun könne. Daher geht das Bestreben vieler Telcos in eine Richtung, die in der rechten Säule von Bild 5 skizziert ist.

Spannend bleibt allerdings die Frage, ob die Telcos, die – wie ihr Name schon sagt – ja klassischerweise Kommunikation auf große Distanzen betreiben, auch im Nahbereich und bei der Verbindung der Menschen mit ihren Geräten und smarten Dingen ein erfolgreiches Business betreiben können oder ob dieses schnell wachsende Segment zunächst eher durch einige der emsigen jungen Start-up-Unternehmen des IoT-Bereichs angegangen werden wird.

IoT über WiFi

Wollen Menschen überhaupt mit ihren Geräten und smarten Dingen verbunden werden? Wozu? Tatsächlich ist dies jetzt, im Zeitalter von Smartphone und Tablet-Computer, die fast nie weiter als Armeslänge entfernt sind, ein überraschend schnell wachsendes Bedürfnis. Meine Jalousien und Markisen möchte ich über eine App meines Mobiltelefons steuern, nicht über eine extra Fernbedienung. Meine dimmbaren Lampen, meine Waschmaschine, meine Badezimmerwaage – dies alles gibt es schon „WiFi-enabled“ – sollen mir die gewünschte Informationen aufs Tablet servieren und meinem Kommandostab, dem Smartphone, gehorchen (Bild 7). Auch meine Wetterstation soll damit verbunden sein und natürlich mit Wetterinformationen aus dem Internet ergänzt werden. Oder der Thermostat an der Wohnzimmerwand – er soll sich nicht nur mit der Heizung, sondern auch mit meinem Smartphone verbinden. Dann kann ich nicht nur die Wunschttemperatur vom Sessel (oder vom Büro aus) einstellen, sondern Thermostat und Smartphone können sich zu meinem Nutzen verschwören – bin ich mit meinem Smartphone weit weg, kann die Heizung ja herunterfahren und Energie sparen, der Katze zuhause wird es schon nichts ausmachen.



Bild 7: Steuerung von Haushaltsgeräten mit Apps (Panasonic).

Vor 15 Jahren habe ich übrigens mit einem meiner Mitarbeiter gewettet, dass man innerhalb von 20 Jahren so etwas Banales wie Glühbirnen mit einem IP-Kommunikationsstack ausstatten wird, und ich habe meine Wette schon jetzt, fünf Jahre früher, gewonnen, denn mittlerweile kommen die ersten IP-Glühlampen auf den Markt. Man muss natürlich zugeben, dass das auch daran liegt, dass es sich bei den neuen „Leuchtmitteln“ streng genommen ja gar nicht mehr um glühende Birnen handelt, und dass die modernen LED-Birnen teurer (und langlebiger) geworden sind, so dass es sich jetzt eher lohnt, dort einen IP-Stack einzubauen, um die

Lampe kontrolliert zu dimmen oder eine komplexere Lichtsteuerung der Wohnung (ich bin in den Ferien, aber potentiellen Einbrechern wird eine automatisch gelernte Steuersequenz vorgegaukelt) realisieren zu können.

- 44 MHz **processor**
 - application programs stored in flash memory
 - 128 kB RAM, 512 kB ROM, 1 MB flash
- **WiFi**
 - wake up and join a secure network (WPA2): < 35 ms
- **TCP/IP stack** installed in ROM
- **Low power**
 - 4 μ A sleep mode (wake-up: sensor input or timer)
 - 40 mA receive data, 210 mA transmit data

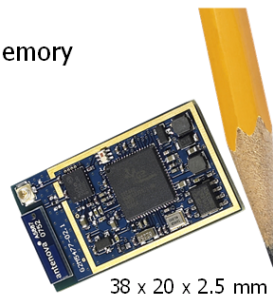


Bild 8: Ein Niedrigenergie-WiFi-System.

Wie sieht es in diesem Bereich mit Hardwareplattformen und Kommunikationsstandards aus? Darauf möchte ich nun kurz eingehen und von eigenen Erfahrungen berichten. Was ich jetzt vorstelle, gibt es so ähnlich auch von anderen Herstellern und auch in neueren Ausprägungen. Es dient hier also lediglich als Exempel. Das Modul (Bild 8) ist ca. 4x2 cm groß, 3 mm dick und hat einen mittelprächtigen Prozessor, der aber für viele Zwecke genügt. Die 128 kB RAM und 512 kB ROM scheinen nicht besonders viel zu sein, reichen aber für typische Anwendungen ebenfalls aus. Die integrierte WiFi-Einheit ist insofern interessant, als dass nach dem Aufwecken aus dem energiesparenden Tiefschlafmodus (mit ca. 4 Mikroampere Stromverbrauch), z.B. durch den Impuls eines externen Sensors, innerhalb von ca. 30 bis 40 Millisekunden eine nach dem WPA2-Standard gesicherte Kommunikationsverbindung aufgebaut wird. Außerdem befindet sich ein vollständiger TCP/IP-Stack im ROM.

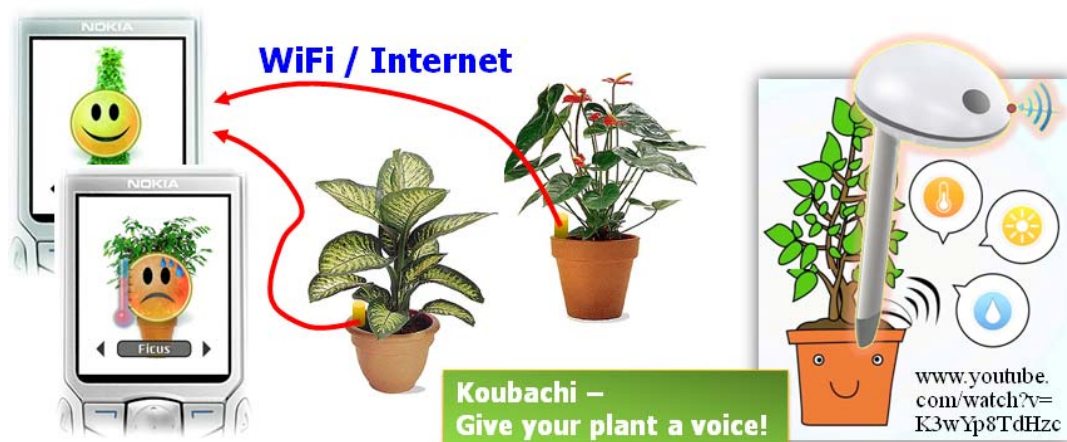


Bild 9: Affektiv kommunizierende Pflanzen und der Koubachi-Pflanzensensor.

Kurz gesagt: Man hat einen Computer, Sensoranschlüsse, WiFi und Internetprotokoll auf einem einzigen kleinen Modul integriert, das ein paar Dollar oder Euro kostet. Was kann man

damit nun alles anstellen? Wir haben die Einheit mit geeigneter Sensorik verbunden und in Blumentöpfe eingebaut. Damit kann man nun mit dem Smartphone des Pflanzenbesitzers kommunizieren und diesem mitteilen, ob es der Pflanze hinsichtlich Feuchtigkeit, Temperatur, Helligkeit etc. gutgeht, ob also alle Sensorwerte im grünen Bereich liegen. Macht man es, wie in Bild 9 dargestellt, mit einem netten Smiley-artigen Icon, das quasi als Avatar der Pflanze fungiert, dann braucht man keine Bedienungsanleitung – man sieht unmittelbar, ob alles stimmt, die Pflanze sich also wohlfühlt, oder was eventuell nicht in Ordnung ist [BO07]. Aus dieser Idee ist ein Start-up-Unternehmen (Koubachi) hervorgegangen – wir hoffen natürlich, dass es erfolgreich sein wird! Konkurrenzfirmen gibt es übrigens auch schon, aber wir vertrauen darauf, dass die Kunden das Schweizer Original schätzen.

Das Prinzip und die konkrete Technik sind aber natürlich nicht nur auf Blumentöpfe, sondern auf andere Gegenstände ebenso anwendbar. Es macht technisch kaum einen Unterschied, ob man misst, ob im Blumentopf noch genug Feuchtigkeit ist, oder ob in einem Badezimmer Wasser verbraucht wird, was sich durch einen Geräuschsensor irgendwo außen am Wasserrohr einfach feststellen lässt. Man ahnt, was damit möglich wird: Ist zum Beispiel in der Wohnung der betagten Verwandten das Bad heute schon benutzt worden? So etwas ließe sich dann aus der Ferne über eine Smartphone-App direkt feststellen – mit allen positiven wie negativen sozialen Nebeneffekten und Konsequenzen. Seien es nun Blumentöpfe oder Wasserarmaturen – jedenfalls verbindet man Alltagsdinge mit dem Internet und seinem „Ökosystem“ aus Apps und Cloud-Diensten – wir sind damit offenbar bereits im Zeitalter des Internet der Dinge angekommen!

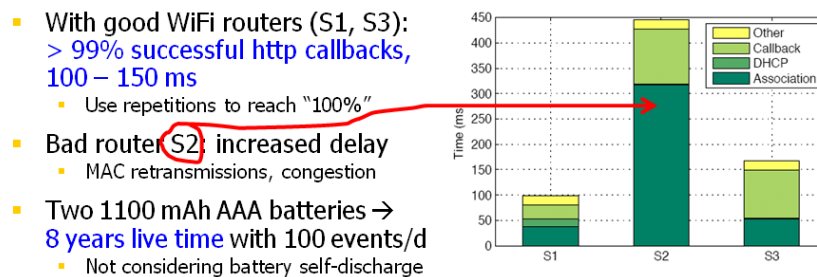


Bild 10: Evaluation der WiFi-Kommunikation.

Die Frage ist, ob so etwas in Wohnungen wirklich gut funktioniert. Kann man tatsächlich mit den Dingen im Haus über WiFi gut, stabil und energieeffizient kommunizieren? Kann man darauf ein Geschäft aufbauen oder muss man dauernd die Hotline bedienen, weil der Router in der Wohnung des Kunden oder irgendetwas anderes wieder einmal nicht funktioniert? Wir haben das natürlich evaluiert [OKS11]. Es geht ganz gut! Bild 10 zeigt drei typische Router aus unserem Test. Man sieht, wie schnell – in ca. 100 Millisekunden, in einem Fall in 500 Millisekunden – die Antwort auf einen HTTP-Request (gesendet von dem in Bild 8 gezeigten Modul) aus dem Internet da war. In über 99% der Fälle klappt es direkt beim ersten Versuch. Es gibt aber vereinzelt auch schlechte Router, wie S2 in Bild 10, mit höherer Fehlerrate (durch packet drops oder inexakte Zeitsynchronisation) und größeren Latenzzeiten. Das sind typischerweise kostengünstigere Modelle, die nicht ganz standardkonform sind oder bei denen etwas grundsätzlich nicht in Ordnung ist. Wenn man diese bei den Leuten zuhause austauscht, dann berichten sie, dass danach das Internet und auch andere Anwendungen endlich gut funk-

tionieren. Das hat sich in letzter Zeit stabilisiert – diese nicht völlig kompatiblen Router verschwinden langsam, und wir haben eigentlich kaum noch Probleme in dieser Hinsicht.

Es ist natürlich klar, dass WiFi einige grundsätzliche Beschränkungen hat. Zum Beispiel reicht das Funksignal zur nächsten Etage, aber manchmal keine zwei Etagen hoch zur dortigen Pflanze. Auch das Bootstrapping, also die Inbetriebnahme, ist leider gegenwärtig noch ein bisschen umständlich – „arrive and operate“, wie weiter oben beschrieben, ist so noch nicht möglich – ein Kunde muss sich evtl. mit SSIDs herumplagen, den Sicherheitsschlüssel eingeben etc. Das sind kleine Nachteile, aber wir stehen ja erst am Anfang der Entwicklung hin zum Internet der Dinge. Zum Beispiel ist mit dem neueren „WiFi Direct“-Standard das Setup einfacher und kundenfreundlicher geworden, es funktioniert ganz wörtlich auf Knopfdruck („Push-Button-Methode“) oder sogar in fast magischer Weise unter Nutzung von NFC (Near Field Communication) – dazu muss man ein smartes Ding wie den Pflanzensensor im Wesentlichen nur in die unmittelbare Nähe des heimischen Internetrouters bringen oder umgekehrt das bereits assoziierte Smartphone an das smarte Objekt halten.

Das Web der Dinge

Wir denken nun aber schon an den nächsten Entwicklungsschritt beim Internet der Dinge und nennen dies das „Web der Dinge“ [GTMW11]. Was ist der Unterschied zum klassischen Internet der Dinge? Zunächst einmal nutzt man beim Web der Dinge die Web-Architektur sehr konsequent, und es steht einem dadurch das ganze mächtige und flexible „Ökosystem“ des Web zur Verfügung; man kann beispielsweise relativ einfach verschiedene Services miteinander verbinden (zu einem „Mashup“), Standard-Browser zur Kommunikation mit Alltagsdingen nutzen oder die Aufrufschnittstellen (also APIs) von Plattformen sozialer Netzwerke direkt aus den smarten Dingen heraus verwenden. Genauer gesagt verwendet man das REST-Prinzip (die Abkürzung steht für „Representational State Transfer“), das als Architekturschema dem World Wide Web zugrundeliegt [FiT02] und ein wichtiger Grund für dessen Erfolg war. Charakteristisch für REST sind das zustandslose Client-Server-Modell sowie die Adressierbarkeit über URLs. Kommuniziert wird ähnlich wie bei HTTP über einfache Request-Primitive GET, POST, PUT und DELETE. Die aktuelle Temperatur eines Kühlschranks erhält man so zum Beispiel einfach durch ein „GET“, gesendet an die Adresse eines HomeGateways wie <http://mattern.myhome.com/fridge/temp> oder direkt an den Kühlschrank <http://fridge.mattern.myhome.com/temp>, sofern dieser „Web-ready“ ist.

Etwas pauschalisierend formuliert: Wir möchten die ganzen schönen und nützlichen Dienste des Web für die smarten Dinge nutzbar machen – und zwar in technisch möglichst einfacher Art und Weise, indem die ressourcenbeschränkten Geräte und Module direkt auf der Anwendungs- bzw. Web-Dienstebene in das Internet integriert werden. Umgekehrt können aber die Dienste im Internet bzw. Web auch die physischen Dinge und deren lokal gewonnenen Daten verwenden. Diese wechselseitige Nutzung kann zum symbiotischen Vorteil beider sein – durch den Beitrag vieler Dinge werden die globalen Dienste schlauer und mächtiger, und dies nützt wiederum den Dingen. Man denke beispielhaft an Navigationssysteme für Autos – wenn einige Fahrzeuge die gerade gefahrene Geschwindigkeit melden, dann kann der auf diese

Weise ermächtigte Dienst Staus erkennen und andere Autos davor warnen sowie das aktuelle Verkehrsaufkommen in seiner Routenberechnung berücksichtigen.

So vorteilhaft das Web auch für die smarten Dinge der Welt erscheint, das zugrundeliegende Internet hat leider einen wesentlichen Schönheitsfehler, wenn es um die lokale Anbindung kleiner, energiearmer Objekte geht: Die klassischen Internet-Protokolle, insbesondere TCP mit seinem Sliding-Window-Mechanismus, wurden für die Kommunikation über große Distanzen und für hohe Bandbreiten ausgelegt. Beides trifft für typische Anwendungen im Internet der Dinge nicht zu. Daher versucht man beim Web der Dinge direkt von der IP-Ebene auf die REST-basierte Anwendungsebene zu kommen, unter Auslassung von TCP. Präziser: Man ersetzt TCP durch ein Protokoll der Anwendungs- bzw. Dienstebene, welches das REST-Prinzip unmittelbar implementiert und damit quasi Web-kompatibel ist. Wir kommen darauf weiter unten noch zurück.

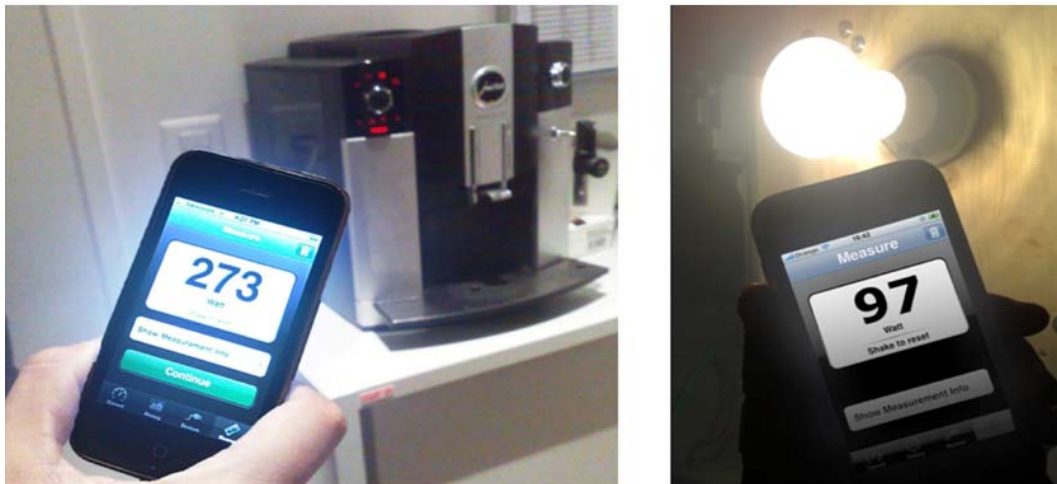


Bild 11: Energiemonitoring von Haushaltsgeräten mit Smartphones.

Schauen wir uns zunächst noch ein anderes Beispiel an, das Energiemonitoring von Haushaltsgeräten (Bild 11). Wollte man nicht schon immer einmal wissen, wie viel Strom die Espressomaschine verbraucht und was das im Endeffekt kostet? Elektrische Geräte sind über den Haushaltsstromzähler an das Elektrizitätsnetz angeschlossen (Bild 12). Der Stromzähler misst den insgesamt verbrauchten Strom einer Wohnung – früher in analoger Form, heute schon oft digital. Ein solcher „intelligenter“ Stromzähler (oder smart meter) hat digitale Schnittstellen, über welche die gemessenen Werte ausgelesen werden können. An eine derartige Schnittstelle kann ein WiFi-Modul angeschlossen werden, dieses kann man aber natürlich auch direkt in den smart meter selbst einbauen. Dies haben wir getan, und darauf haben wir außerdem noch einen Web-Server gepackt. Das heißt, dass der smart meter jetzt ein Gerät im Internet ist, das von überall her angesprochen werden kann, und zwar mit einem normalen Browser oder Smartphone [Wei09, Wei12].

Ich kann so für unsere Testhaushalte in Zürich von überall auf der Welt aus überprüfen, wie viel Strom sie gerade verbrauchen. Oder ich kann mir die Verbrauchshistorie anzeigen lassen – das sind dann teilweise recht merkwürdige Kurven, die bei der sekundengenauen Messung

angezeigt werden. Man fragt sich unwillkürlich, was die Bewohner gerade machen, ob sie etwa Musik hören oder die Waschmaschine laufen lassen, wenn sich der Stromverbrauch rhythmisch ändert (Bild 12, rechts unten) bzw. vielleicht Fernsehen schauen oder den PC nutzen, wenn der Verbrauch innerhalb eines Bandes hastig schwankt (Bild 12, rechts oben). Natürlich wird man hier unmittelbar mit der Privacy-Problematik konfrontiert – ein wichtiger Aspekt (auch weil sich schon mit wesentlich grobgranulareren Strommessungen im Halbstundentakt relevante sozio-ökonomische Aspekte über einen Haushalt ableiten lassen [BSS13]), der im Rahmen dieses Beitrags aber nicht weiter thematisiert werden kann.

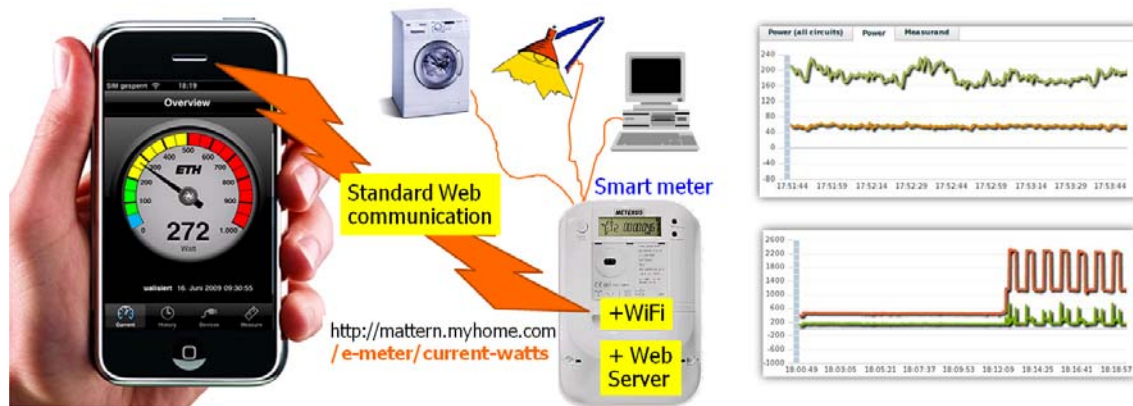


Bild 12: Ein intelligenter Stromzähler als Server im Web der Dinge.

Zurück zur oben aufgeworfenen Frage, wie viel Strom die Espressomaschine verbraucht. Bild 11 und Bild 12 illustrieren das zugehörige Szenario: Man muss nur kurz die Maschine ein- und ausschalten, dann wird die Differenz des Verbrauchs angezeigt, und es werden die monetären Kosten für einen sinnvollen Zeitraum hochgerechnet. Das Szenario stellt eine typische Anwendung im Web der Dinge dar: Adressiert werden die smarten Dinge (hier also der Stromzähler) über eine Web-Adresse, eine URL. Smarte Dinge implementieren entsprechend des REST-Prinzips einen einfachen Web-Server, der Anfragen von Clients beantworten kann. Clients sind im skizzierten Beispiel Web-Browser in einem Smartphone, wobei der Browser z.B. mittels HTML5 ein perfektes und ansprechendes GUI (Graphical User Interface) realisieren kann. Der Nutzer hat so das Gefühl, dass die Lampe oder die Espressomaschine dem Smartphone ihre Stromverbrauchswerte sendet – tatsächlich aber erledigt das der intelligente Stromzähler stellvertretend für die elektrischen Geräte.

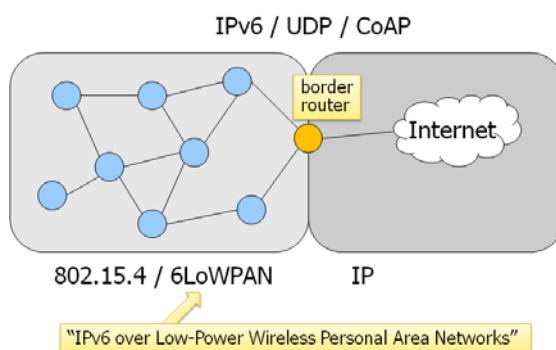


Bild 13: Der Anschluss kleinster Objekte an das Internet.

Durch all die oben geschilderten Prinzipien und Techniken wird die Entwicklung von Anwendungen im Internet der Dinge stark vereinfacht. Wenn man es einmal auf diese Weise gemacht hat, fragt man sich, warum man solche Anwendungen früher überhaupt anders realisiert hat und warum zum Beispiel bei der Gebäudeautomation noch immer so viele spezielle, oft etwas komplizierte, Protokolle und Standards verwendet werden und wieso man in diesem Bereich nicht zumindest heutzutage konsequent auf Internet-Protokolle setzt [KWG10].

Wie werden smarte Dinge mit ihren kleinen Kurzstrecken-Funkmodulen am besten an das globale Internet angeschlossen? Natürlich könnten die smarten Dinge einen klassischen TCP/IP-Protokollstack besitzen; bei dem oben erwähnten Pflanzensensor ist dies auch tatsächlich der Fall. Wie schon gesagt, ist TCP allerdings nicht immer gut geeignet, schon aus energetischen Gründen. In letzter Zeit hat sich daher alternativ eine auf Wireless Personal Area Networks (WPAN) beruhende Architektur herausgebildet [Ish13]: Ein „border router“ vermittelt zwischen der Außenwelt und dem WPAN, das typischerweise im Heimbereich installiert ist (Bild 13). Im Außenbereich haben wir die klassische TCP/IP-Protokollwelt des Internet, im Innenbereich ein im Allgemeinen stärker vermaschtes und sich teilweise selbstorganisierendes WPAN.

Layer	Protocol
Application	Constrained Application Protocol (CoAP)
Transport	UDP
Network	IPv6, RPL
Adaption	6LoWPAN
MAC	CSMA
Radio Duty Cycling	IEEE 802.15.4e, ContikiMAC
Physical	IEEE 802.15.4




Bild 14: Ein typischer Protokollstack im Internet der Dinge.

Der zugehörige Protokollstack ist in Bild 14 skizziert. Auf den unteren Ebenen findet man IEEE 802.15.4 als Kurzstrecken-Funkstandard, was man im Prinzip auch von ZigBee her kennt. Damit sind zwar nur Bandbreiten von einigen zig kbit/s möglich, allerdings mit geringer Verzögerung und vor allem wird vergleichsweise sehr wenig Energie zur Kommunikation benötigt. Statt IPv4 setzt man konsequent auf IPv6 – die vielen smarten Dinge, die man sich für die Zukunft vorstellt, benötigen ja auch tatsächlich einen viel größeren Adressraum als man bisher im Internet mit den 32-Bit-Adressen zur Verfügung hatte. Nun ist aber die Datenpaketstruktur von IPv6 ganz anders ausgestaltet als die Framestruktur bei 802.15.4 – IPv6 hat große Pakete mit mächtigen Headern und langen Adressen, in einen Frame passt aber nur eine Nutzlast von maximal 127 Byte. Daher wird eine Anpassungsschicht benötigt, 6LoWPAN (“IPv6 over low-power wireless personal area networks”). Durch verschiedene Mechanismen, auf die hier nicht genauer eingegangen werden kann (u.a. verlustfreie Komprimierung des Headers und Fragmentierung in Sequenzen kurzer Frames), bildet 6LoWPAN die IPv6-Pakete

erstaunlich effizient auf die 802.15.4-Infrastruktur ab [HuC08]. Für das Routing innerhalb des WPAN wurde ebenfalls ein eigenes Protokoll entwickelt, RPL (“routing protocol for low power and lossy networks”). Bei diesem spielt der border router eine entscheidende Rolle – er soll den Pfad zu jedem Knoten innerhalb des WPAN kennen, umgekehrt kann von jedem dieser Knoten der border router auf bekannten Pfaden einer baumartigen Struktur erreicht werden. Über IPv6 residiert dann nicht wie sonst meist TCP, sondern UDP, ein klassisches Internet-Protokoll. Und darauf setzt dann CoAP („Constrained Application Protocol“) auf, das gegenwärtig von der Internet Engineering Task Force (IETF) standardisiert wird [LHK12, Vil12].

Bild 15 führt einige Eigenschaften von CoAP auf. Es unterstützt direkt das oben erwähnte REST-Architekturprinzip sowie Web-basierte Anwendungen durch URIs und http-Mapping. Interessant ist auch das Observe-Paradigma im Zusammenhang mit dem Client-Server-Modell von REST: Damit kann man im Sinne des Publish-Subscribe-Prinzips erreichen, dass die Kontrolle insofern an den Server, also ein zu kontrollierendes Gerät oder ein smartes Ding, abgegeben wird, als dass dieses sich später wieder von sich aus meldet, wenn eine bestimmte Bedingung eingetreten ist – man vermeidet so ein sonst notwendiges aktives Polling. Das „resource discovery“ hilft, das weiter oben erwähnte Arrive-and-Operate“-Problem anzugehen. Daran und an der Unterstützung energiesparender, länger „schlafender“ Knoten erkennt man, dass mit CoAP ein Protokoll entwickelt wurde, das von vornherein für eine Welt konzipiert wurde, in der kleinste Geräte und smarte Dinge das Internet bzw. das Web nutzen. Die Open Mobile Alliance hat sich mit ihrer „Lightweight Machine to Machine Technical Specification“ (LWM2M) inzwischen stark an CoAP angelehnt.

- RESTful Web services for **resource-constrained** devices
- URIs, simple **http** mapping
- Asynchronous message exchange
- **Reliable UDP** transport («confirmable»)
- Alternative transport bindings (e.g., **SMS**)
- Push notifications (**observer paradigm**)
- **Multicast**
- **Resource discovery**
- Binary (optimized for parsing)
- Supports **sleepy nodes**, bulky data
- Datagram Transport Layer Security (DTLS)

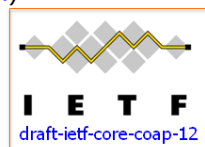


Bild 15: Eigenschaften von CoAP.

Der Vollständigkeit halber sei erwähnt, dass neben WiFi und 802.15.4 (bzw. dem darauf aufbauenden ZigBee) natürlich noch einige andere Funkkommunikationsprotokolle für den Nahbereich existieren, die sich unter gewissen Bedingungen für das Internet der Dinge qualifizieren, wie z.B. Z-Wave und Wireless M-Bus im Bereich der Gebäudeautomation oder das mittlerweile schon etwas betagte Bluetooth, eigentlich zur kabellosen Anbindung von Peripheriegeräten entwickelt. Hier gibt es aktuelle Weiterentwicklungen (z.B. Bluetooth Low Energy, Bluetooth SMART / 4.0); es werden aber auch neue Protokolle propagiert, wie zum Beispiel das auf der Technik der Schnurlostelefonie und digitaler Funk-Babyfone beruhende DECT ULE (Ultra Low Energy). Interessant erscheint vor allem auch WiFi Direct, womit per WiFi von einem Gerät zu einem anderen kommuniziert werden kann, ohne dass eine Basisstation

(Wireless Access Point) oder Router involviert sein muss. Es weist kurze Setup-Zeiten und diverse Pairing-Möglichkeiten auf. Genauer kann im Rahmen dieses Beitrages allerdings nicht auf die diversen Technologien bzw. Protokolle und deren Chancen in einem zukünftigen Internet der Dinge eingegangen werden.

Autos im Internet der Dinge

Ich möchte im Folgenden kurz ein Projekt, CloudThink, schildern, das am MIT durchgeführt wird und wo auch wir an der ETH Zürich ein Stück weit mitarbeiten. Man spricht zur Zeit ja viel von „Big Data“. Heutige Autos erzeugen viele Daten, sind also gewissermaßen Big-Data-Produzenten. Dafür verantwortlich sind die Sensor- und Bussysteme in den Autos. Wir greifen nun die Daten vom CAN-Bus über den OBD-Port eines Autos ab, der eigentlich für die Fahrzeugdiagnose vorgesehen ist – das Kürzel OBD steht ja für „on-board diagnostics“. Diese Daten werden per Mobilfunk – also entsprechend des klassischen simkartenbasierten M2M-Paradigmas – in die Cloud geschickt und füttern dort eine virtuelle Repräsentation („ghost car“) des jeweiligen Autos, worauf dann beliebige Applikationen zugreifen können. Das ganze funktioniert im Prinzip für beliebige Autotypen, und generell ist das Projekt „open data – open source“. Zwar bietet heute jeder bessere Fahrzeughersteller etwas an, um mit dem Smartphone tolle Dinge im Auto machen zu können, auf gewisse Daten des Autos zugreifen zu können etc., aber dies ist immer proprietär. Wir sind ganz bewusst rebellisch offen – offen für Daten von jedem Autotyp und von jedem Autobesitzer, sofern er zustimmt. Aber selbstverständlich ist es möglich, den Zugriff Dritter auf die eigenen Daten einzuschränken.

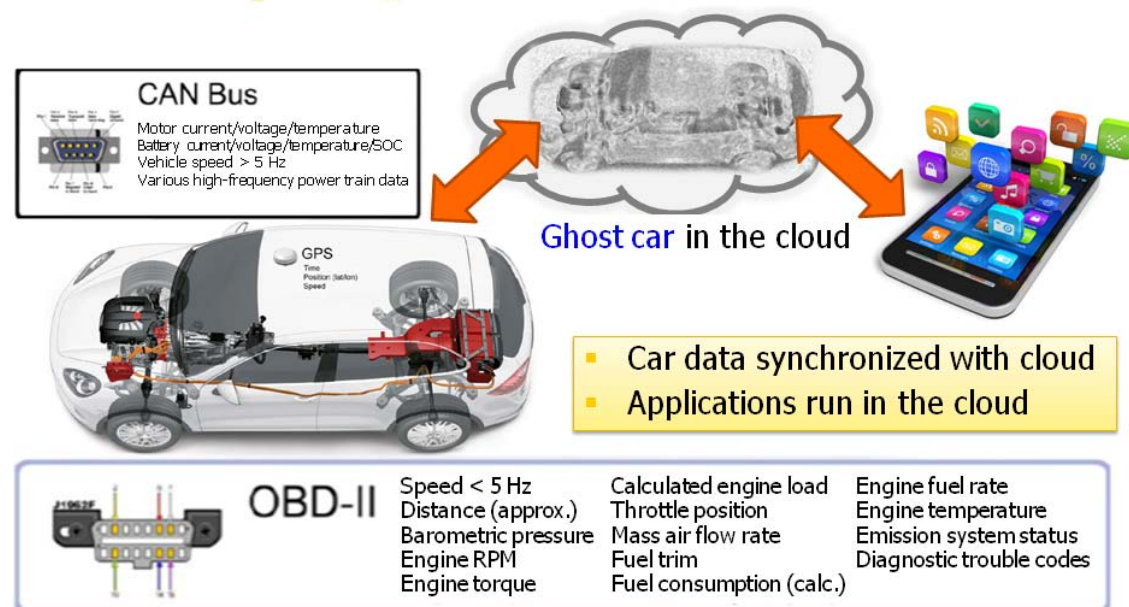


Bild 16: Das MIT-CloudThink-Projekt mit „ghost cars“ in der Cloud.

Wie Bild 16 zeigt, sind recht viele unterschiedliche Informationen über den OBD-Port abgreifbar. Man kann aber auch einige Steuerinformationen von außen setzen und beeinflussen. Wir konnten z.B. manche Autos aus der Ferne über das Internet auf- und zuschließen.

Wir konnten sogar die Bremsen aktivieren – nicht bei allen Modellen, aber immerhin in Einzelfällen. Es hat uns schon erstaunt, was damit alles machbar ist – es gibt einen Vorgehensschmack auf die Chancen und Herausforderungen der Cyber-Physical Systems kommender Zeiten, die in gewisser Weise eine Fortsetzung des Internets der Dinge in die Welt smarter Maschinen und großflächiger informatisierter Infrastrukturen darstellen, mit denen u.U. kräftig in die physische Welt hineingewirkt wird [CPS12].

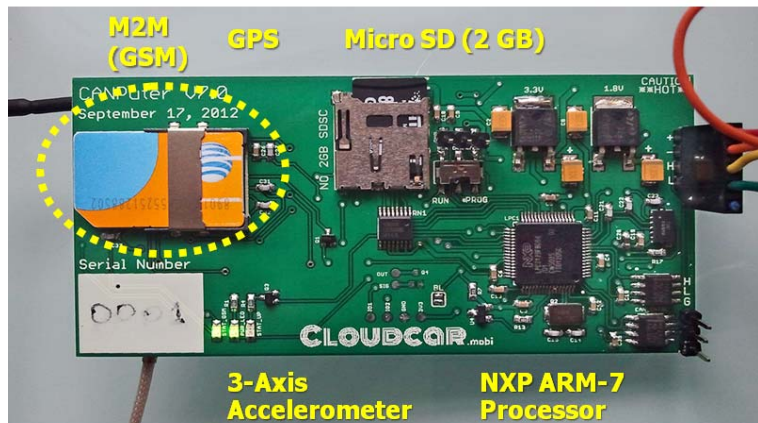


Bild 17: Hardware-Einheit, um die OBD-Daten vom Bus abzugreifen und in die Cloud zu schicken.

Bild 17 zeigt die Hardware-Einheit, die die OBD-Daten vom Bus abgreift und sekundlich über das Mobilnetz in die Cloud schickt. Zusätzlich sind noch Beschleunigungssensoren sowie ein GPS-Empfänger für Ortungs- und Navigationsanwendungen integriert. Das ganze sollte sich als Produkt für ca. 100 € herstellen lassen. Die Hardware-Einheit wird in ein steckerförmiges Gehäuse integriert, damit sie von Kunden nach dem „Plug-and-Forget-Prinzip“ verwendet werden kann (Bild 18).

- Connect device to OBD-port, then insert memory card and SIM card – if the light blinks, everything is OK



Bild 18: Installation des CloudThink-Systems im Auto.

Ich möchte an diesem Beispiel auch zeigen, wie nützlich es ist, dass man unmittelbar Web-Dienste nutzen kann und smarte Dinge in Mashups einbinden kann: Wir haben die Auto-Daten (insbesondere auch die geographische Position mit Zeitstempeln) im Internet, und es gibt natürlich Google Maps im Internet – genauer: Google Maps stellt seine Funktionalität in Form von APIs, also Aufrufchnittstellen, im Web zur Verfügung. Dann kostet es einen

halbwegs geübten Poweruser nur Minuten, oder bestenfalls wenige Stunden, bis er die Positionsdaten mit Google Maps verbunden hat; typischerweise werden solche Mashups dann im Internet frei verfügbar gemacht und andere können die Konstruktion gegebenenfalls verbessern oder erweitern.

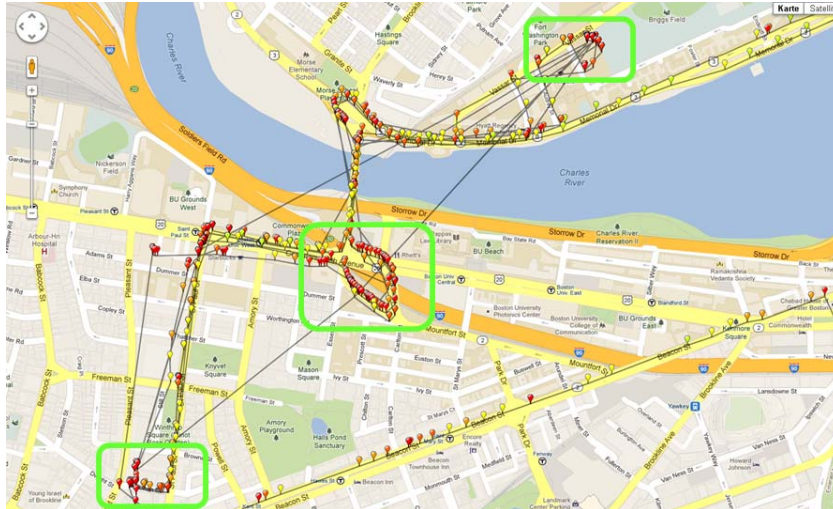


Bild 19: Mashup mit Google Maps: Kartenausschnitt von Boston, USA.

In Bild 19 sehen wir im Ergebnis des Mashups, wo das Auto eines Kollegen vom MIT in Boston herumgefahren ist. Ein kleiner roter Punkt heißt, es ging nur langsam vorwärts, gelb steht für eine mittlere Geschwindigkeit an dieser Stelle und grün (was in diesem Beispiel so gut wie nicht vorkommt) bedeutet schnelle Fahrt. Natürlich haben wir nun die volle Funktionalität von Google Maps zur Verfügung, wir bekommen diese in gewisser Weise geschenkt – dies illustriert die „Macht“ des Web-Ökosystems im Allgemeinen und von Mashups im Speziellen. Wollen wir genauer analysieren, was in auffälligen Bereichen, z.B. den auf der Karte markierten drei Ausschnitten, los ist, dann können wir dort googlemäßig hineinzoomen. Bild 20 zeigt dies beispielhaft für zwei der markierten Gebiete, wobei auf die Satellitenbild-Darstellung umgeschaltet wurde.

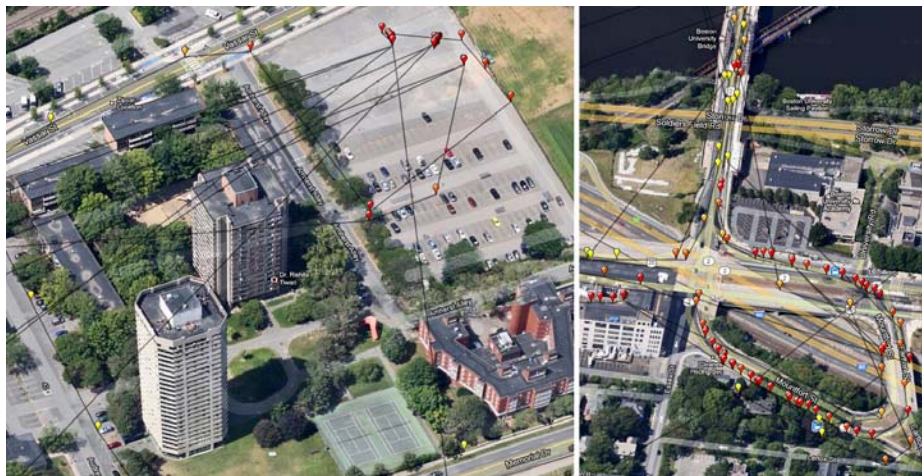


Bild 20: Satellitenbild-Zoom auf die Gebiete rechts oben und mitte von Bild 19.

Offenbar handelt es sich beim Gebiet rechts oben auf der Karte (Bild 20 links) um einen großen Parkplatz (zwischen MIT Westgate und dem Baseball-Platz des MIT), dies erklärt die vielen roten Punkte auf engem Raum bei der Karte. Wenn man noch die Zeitstempel betrachten würde, dann wird einem endgültig klar, dass hier in der Nähe der Arbeitsplatz des Autobesitzers liegen muss. Links unten auf der Karte in Bild 19 dürfte er zuhause sein, und das auffällige Gebiet in der Mitte der Karte ist in Bild 20 rechts als Satellitenbild dargestellt – der allmorgendliche Stau auf der Rampe zur Brücke über den Charles River.

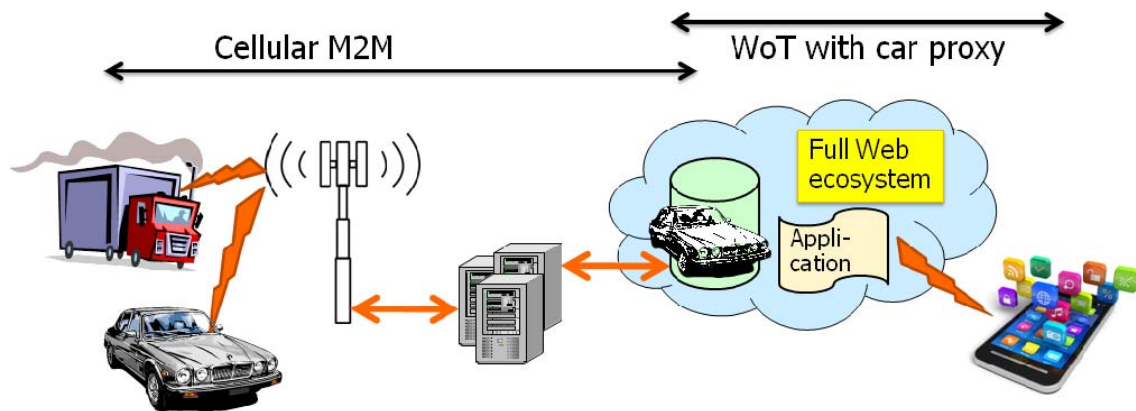


Bild 21: M2M und Web of Things mit „ghost car“ als Proxy.

Wie stellt sich nun das Gesamtsystem dar? Bild 21 skizziert dies: Wir haben klassische sim-kartenbasierte und zelluläre M2M-Technik genutzt, um die Daten sekundlich vom Auto abzugreifen und ein virtuelles Gegenstück des Fahrzeugs in der Cloud zu alimentieren. Ab dort ist das virtuelle Auto ein „Gegenstand“, der über Technologien des Internet/Web der Dinge zugreifbar ist und eine Vielzahl von Applikationen ermöglicht. Insofern verschwimmen hier die M2M- und IoT-Technologien bzw. werden jeweils in denjenigen Bereichen eingesetzt, wo sie besonders geeignet sind, um so zu einem Gesamtsystem kombiniert zu werden.

Proxies für Schokoladentafeln und andere dumme Dinge

Ich möchte abschließend noch erläutern, wie Smartphones als Proxy so wirken können, dass sie wirklich dumme Dinge smart erscheinen lassen. Tatsächlich brauchen längst nicht alle Objekte, die im Internet verkörpert werden sollen und uns gegenüber eine smarte Rolle einnehmen sollen, mit Elektronik ausgestattet zu werden. Es genügt oft, wenn wir einen geeigneten Proxy haben, der das Ganze stellvertretend für sie macht. Bild 22 zeigt ein entsprechendes Szenario: Eine Tafel Schokolade besitzt, wie jedes Supermarktprodukt, einen Strichcode. Smartphones sind ideale und mächtige Mediatoren zwischen dem Cyberspace und der dinglichen Welt – denn sie besitzen drei Schnittstellen: zum Menschen, zum Internet sowie (via Kameras und anderer Sensorik) zur Realität – jedenfalls zu demjenigen Ausschnitt davon, der für uns Menschen meist nur relevant ist, nämlich der unmittelbaren Umgebung. Mit der Kamera und geeigneter Software im Smartphone können wir den Strichcode des Produkts in null Komma nichts ablesen und auswerten, auch auf unscharfen und verwackelten Bildern [ALF06, SF12], gehen mit der eindeutigen Artikelnummer – ohne dass der Nutzer es richtig

merkt – schnell ins Internet, holen uns dort die relevante Information („wo wird das Produkt hergestellt?“) und zeigen schließlich an, ob es sich um ein Schweizer Produkt handelt oder nicht.

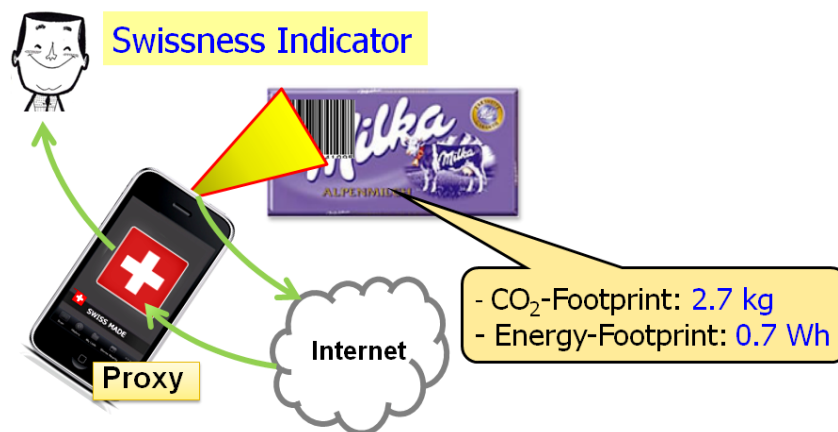


Bild 22: Das Smartphone als Proxy für „dumme“ Objekte.

Bei dem dargestellten Typ vermeintlicher Schweizer Suchard-Schokolade kann man übrigens eine Überraschung erleben – es wird nämlich „Hergestellt in Lörrach, Deutschland“ angezeigt – und tatsächlich ist der Markenname von Suchard an „Kraft Foods“ übergegangen! Aber mit der „Swissness“ ist es, das sei en passant noch angemerkt, sowieso nicht so einfach: Zwar kann Schokolade mit (und wegen) dem Schweizerkreuz 20% teurer verkauft werden (was auch dazu führt, dass immer mehr Schweizer ausländische Schokolade essen, über 30% wird schon importiert), aber künftig sollen Lebensmittel nach einer parlamentarischen Initiative nur dann als schweizerisch gelten, wenn mindestens 80% des Gewichts der Rohstoffe aus der Schweiz stammen. Milkschokolade hat allerdings 30% Kakaoanteile, und Bitterschokolade sogar über 50%. Ein Vertreter von CHOCOSUISSE, dem Verband der schweizerischen Schokoladefabrikanten, bemerkte dazu resignierend, dass in Flawil im Untertoggenburg, Standort einer Schokoladenfabrik, leider keine Kakaobäume wachsen würden.

Aber abgesehen von solchen Feinessen: Wer bevorzugt Erzeugnisse aus heimischer Produktion kaufen will, kann also nun das Smartphone mit der Swissness-App analog zu einem Geigerzähler nutzen – er bekommt beim Scannen im Supermarkt sofort angezeigt, ob es sich um einen heimischen Artikel handelt – im positiven Fall durch das vertraute und vertrauenerweckende Schweizerwappen („ein aufrechtes, freistehendes weisses Kreuz im roten Felde, dessen unter sich gleiche Arme je einen Sechstel länger sind als breit“). Natürlich könnte man, wie in Bild 22 ebenfalls angedeutet, „political shopping“ auch etwas weniger national und mehr globalisierend verstehen – als Anzeige von Fairtrade-Attributen oder der freigesetzten CO₂-Menge zusammen mit der benötigten Energie bei der Herstellung des Artikels. Die informationstechnische Realisierung wäre identisch, fraglich ist höchstens, ob sich solcherart Produktinformationen für genügend Supermarktartikel in allgemein zugänglichen Datenbanken oder im Internet finden lassen.

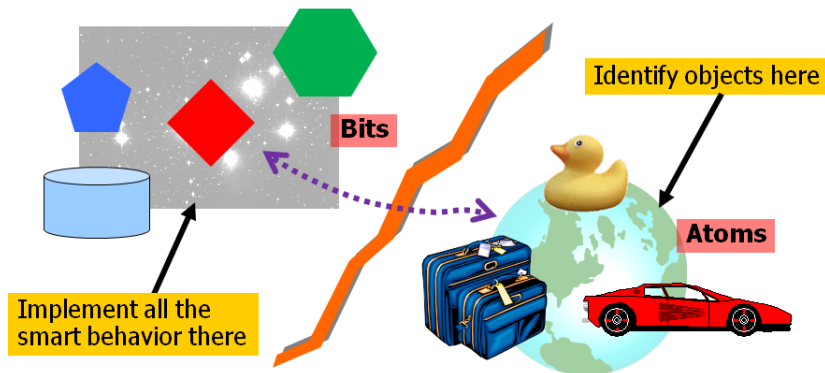


Bild 23: Auslagern der Smartness in das Internet.

Das Smartphone, das als Proxy und Mediator für ein an sich dummes Produkt agiert, macht das Produkt nicht wirklich smart, aber für den Nutzer sieht es so aus, als ob das Ding über den Mediator mit ihm kommuniziert. Diese scheinbare, über den Proxy vermittelte Smartness, die in das Internet – dort wo viel Prozessorleistung, Speicherplatz und Energie vorhanden ist – ausgelagert wurde, reicht zwar nicht für alle, aber doch für viele Anwendungsszenarien aus; der Gegenstand selbst muss nicht smart sein, sondern nur eindeutig identifiziert werden können (Bild 23). Der Trick besteht also darin, die dingliche Welt aus Atomen mit der virtuellen Welt des Cyberspace, der Cloud, dem Internet oder wie man es auch immer nennen mag, näher zusammenbringen. Atome mit Bits zu verbinden, das ist die Idee, die letztlich hinter dem Internet der Dinge steht.



Bild 24: Smarte und vernetzte Produkte informieren über sich selbst und preisen sich an.

Verfahren zur automatischen und berührungslosen Identifikation von Objekten aus der Ferne („Auto-ID“) werden im Logistikbereich schon länger eingesetzt, insbesondere die RFID-Technik („radio-frequency identification“) mit Reichweiten von einigen Metern spielt hier eine wichtige Rolle [FF05]. Im Prinzip funktionieren RFID-Anwendungen analog zu dem in Bild 23 geschilderten Szenario. Neuerdings bindet man aber auch Menschen in ihrer Rolle als Kunden in Kaufhäusern und Einzelhandelsgeschäften in solche Auto-ID-Szenarien mit ein: Ein elektronisches Display, an das man einen Artikel halten kann, damit Zusatzinformation angezeigt wird (Bild 24 links, www.techworks.be/images/gallery/big/digitalsignage_productinfo_big.jpg), vermittelt so gesehen ein Verkaufsgespräch des Artikels: „Ich bin so schön, kauf‘ mich, ich

bin so billig, andere haben mich auch gut gefunden, deine Freunde empfehlen mich...“. Bild 24 rechts (de-bug.de/mode/archives/95.html) stellt ein ähnliches Szenario dar: Man hält beim Shoppen den Anzug mit dem NFC-Chip im Kleiderbügel oder dem eingenähten RFID-Markenlabel (siehe z.B. www.textrace.ch) vor die Videowand und schon erscheint das passende Video mit der richtigen Stimmung, der Anzug dabei chic vorgeführt von einem Model. Zuhause sieht es dann nicht ganz so schön und elegant aus, aber man selbst hat ja auch keine Model-Figur...

Man kann dieses Darstellen von Zusatzinformation zu physischen Gegenständen auch als eine Art „augmented reality“ ansehen – elektronische Displays in Geschäften und Smartphones in der Hand werden in dieser Hinsicht bald ergänzt durch smarte Brillen; „Google Glass“ (Bild 25) ist hier wohl erst der Anfang einer spannenden Entwicklung hin zu einer allgegenwärtigen erweiterten Realität, bei der die dingliche Welt von einer (normalerweise unsichtbaren) „Infosphäre“ überlagert wird.



Bild 25: „Google Glass“ als prominenter Prototyp einer smarten Brille.

Stellen die Szenarien aus Bild 24 eher nüchterne und nützliche Produktinformationen dar oder vielmehr anpreisende Werbung zum Wohle des Verkäufers? Wohl wie man es nimmt. Mit der schnell voranschreitenden Informationstechnik – dem Internet der Dinge, der erweiterten Realität, den Cyber-Physical Systems –, die immer näher an uns Menschen heranrückt und uns in neue soziotechnische Strukturen einbindet, wird die Welt sicherlich nicht grundsätzlich besser, nur eben anders. So ganz überzeugt, dass wir nur Gutes bewirken mit diesen Entwicklungen, können wir daher nicht sein. Aber Informatik ist ja auch kein automatischer Heilsbringer, wie uns Gunter Dueck in seiner jüngsten Kolumne im Informatik-Spektrum in Erinnerung ruft [Due13]. Gerade uns technikaffinen Informatikern würde es seiner Ansicht nach schwerfallen zu verstehen, „dass alles Neue daraufhin abgeklopft wird, ob es als Waffe taugt, andere Menschen zu betrügen hilft oder anderen Menschen triviale Lüste aufschwätzen lässt, so dass man Geld wie Heu machen kann“. Und weiter meint er: „Informatik mündet in High-Tech, in Triallustbefriedigung, in Waffenkonstruktionen, in demütigende Incentive-Systeme, in neue Hochbildung, andere Zusammenlebensformen, neue Kulturen und Staatsformen. Informatik bewirkt.“ Sicherlich bewirkt das Internet der Dinge – und im weiteren Sinne die Informatisierung der physischen Welt – Gewaltiges. Auf dem Weg dorthin sollten wir Experten daher wachsam bleiben und auch unserer Verantwortung gegenüber der Gesellschaft nachkommen.

Ich bedanke mich für Ihre Aufmerksamkeit.

Literatur

- [ALF06] Robert Adelman, Marc Langheinrich, Christian Floerkemeier: A Toolkit for Bar-Code-Recognition and -Resolving on Camera Phones – Jump Starting the Internet of Things. GI Jahrestagung 2006 (2), pp. 366–373, 2006
- [Ash09] Kevin Ashton: That ‘Internet of Things’ Thing. RFID Journal, 22 June 2009
- [BMW11] Bundesministerium für Wirtschaft und Technologie (Hg.): Machine-to-Machine-Kommunikation – eine Chance für die deutsche Industrie. 2011
- [BO07] Philipp Bolliger, Benedikt Ostermaier: Koubachi – A Mobile Phone Widget to Enable Affective Communication with Indoor Plants. Adjunct Proceedings of MobileHCI, Mobile Interaction with the Real World Workshop (MIRW 2007), 2007
- [BSS13] Christian Beckel, Leyna Sadamori, Silvia Santini: Automatic socio-economic classification of households using electricity consumption data. Proc. Fourth Int. Conf. on Future Energy Systems (e-Energy 2013), ACM, pp. 75–86, 2013
- [CPS12] Eva Geisberger, Manfred Broy (Hg.): AgendaCPS – integrierte Forschungsagenda Cyber-Physical Systems. Deutsche Akademie der Technikwissenschaften acatech / Springer, 2012
- [Due13] Gunter Dueck: Vorstellungsbilder rund um die Informatik. Informatik-Spektrum 36(3), pp. 324–328, 2013
- [FiT02] Roy T. Fielding, Richard N. Taylor: Principled design of the modern Web architecture. ACM Transactions on Internet Technology 2(2), pp. 115–150, May 2002
- [FF05] Christian Floerkemeier, Elgar Fleisch: Augen und Ohren für die Logistik. Technology Review 4/05, pp. 34–36, April 2005
- [Flo08] Christian Floerkemeier, Marc Langheinrich, Elgar Fleisch, Friedemann Mattern, Sanjay E Sarma (Eds.): The Internet of Things (IoT 2008), First International Conference, Springer LNCS 4952, 2008
- [FM05] Elgar Fleisch, Friedemann Mattern (Hg.): Das Internet der Dinge – Ubiquitous Computing und RFID in der Praxis. Springer, 2005
- [Ger99] Neil Gershenfeld: Wenn die Dinge denken lernen. Econ, 1999
- [GTMW11] Dominique Guinard, Vlad Trifa, Friedemann Mattern, Erik Wilde: From the Internet of Things to the Web of Things: Resource Oriented Architecture and Best Practices. In: Dieter Uckelmann, Mark Harrison, Florian Michahelles (Eds.): Architecting the Internet of Things. Springer, pp. 97–129, 2011
- [HuC08] Jonathan W. Hui, David E. Culler: IP is dead, long live IP for wireless sensor networks. Proc. 6th Int. Conf. on Embedded Networked Sensor Systems (SenSys), ACM, pp. 15–28, 2008
- [Ish13] Isam Ishaq, David Carels, Girum K. Teklemariam, Jeroen Hoebeke, Floris Van den Abeele, Eli De Poorter, Ingrid Moerman, Piet Demeester: IETF Standardization in the Field of the Internet of Things (IoT): A Survey. J. Sens. Actuator Netw. 2(2), pp. 235–287, 2013
- [KWG10] Matthias Kovatsch, Markus Weiss, Dominique Guinard: Embedding Internet Technology for Home Automation. Proc. 15th IEEE Int. Conf. on Emerging Technologies and Factory Automation (ETFA 2010), 2010
- [LHK12] Christian Lerche, Klaus Hartke, Matthias Kovatsch: Industry Adoption of the Internet of Things: A Constrained Application Protocol Survey. Proc. 17th IEEE Int. Conf. on Emerging Technologies & Factory Automation (ETFA), 2012
- [MaF10] Friedemann Mattern, Christian Flörkemeier: Vom Internet der Computer zum Internet der Dinge. Informatik-Spektrum 33(2), 107–121, April 2010
- [Mat02] Friedemann Mattern: Vom Handy zum allgegenwärtigen Computer – Ubiquitous Computing: Szenarien einer informatisierten Welt. In: Analysen der Friedrich-Ebert-Stiftung zur Informationsgesellschaft 6, pp. 1–17, 2002
- [Mat03] Friedemann Mattern: From smart devices to smart everyday objects. Proc. Smart Objects Conference (SOC 2003, Symposium Objets Communicants, May 15–17, 2003, Grenoble, France), pp. 15–16, 2003
- [Mat05] Friedemann Mattern: Die technische Basis für das Internet der Dinge. In: Elgar Fleisch, Friedemann Mattern (Hg.): Das Internet der Dinge – Ubiquitous Computing und RFID in der Praxis. Springer, pp. 39–66, 2005

- [MDT12] Simon Mayer, Dominique Guinard, Vlad Trifa: Searching in a Web-based Infrastructure for Smart Things. Proc. 3rd Int. Conf. on the Internet of Things (IoT 2012), IEEE, pp. 119–126, 2012
- [OKS11] Benedikt Ostermaier, Matthias Kovatsch, Silvia Santini: Connecting Things to the Web using Programmable Low-power WiFi Modules. Proc. 2nd Int. Workshop on the Web of Things (WoT 2011), ACM, 2011
- [Roe10] Kay Römer, Benedikt Ostermaier, Friedemann Mattern, Michael Fahrmaier, Wolfgang Kellerer: Real-time search for real-world entities: A survey. Proceedings of the IEEE 98(11):1887–1902, 2010
- [SF12] Gábor Sörös, Christian Floerkemeier: Towards next generation barcode scanning. Proc. 11th Int. Conf. on Mobile and Ubiquitous Multimedia (MUM 2012), 47, ACM, 2012
- [Sch02] Chana R. Schoenberger: The internet of things. Forbes Magazine, March 18, 2002
- [Vil12] Berta Carballido Villaverde, Dirk Pesch, Rodolfo de Paz Alberola, Szymon Fedor, Menouer Boubekeur: Constrained Application Protocol for Low Power Embedded Networks: A Survey. Proc. Sixth Int. Conf. on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), IEEE, pp. 702–707, 2012
- [Wei91] Mark Weiser: The Computer for the 21st Century. Scientific American 265(9):66–75, 1991
- [Wei09] Markus Weiss, Friedemann Mattern, Tobias Graml, Thorsten Staake, Elgar Fleisch. Handy feedback: Connecting smart meters with mobile phones. Proc. 8th Int. Conf. on Mobile and Ubiquitous Multimedia (MUM 2009), ACM, 2009
- [Wei12] Markus Weiss, Claire-Michelle Loock, Thorsten Staake, Friedemann Mattern, Elgar Fleisch: Evaluating Mobile Phones as Energy Consumption Feedback Devices. In: Patrick Sénac, Max Ott, Aruna Seneviratne (Eds.): Mobile and Ubiquitous Systems: Computing, Networking, and Services (Mobiquitous 2010), Springer, pp. 63–77, 2012