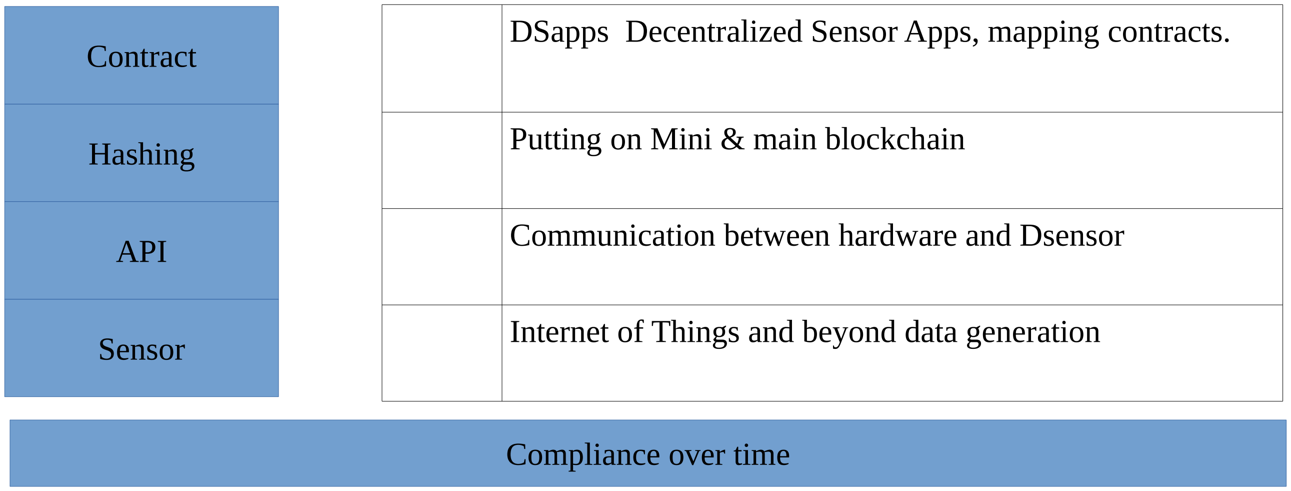
Dsensor – Decentralized Sensor Protocol

1 Introduction

The majority of data generated will come from sensors. Given this volume and priority position this paper makes the case for building a Dsensor protocol as part of the decentralized infrastructure for the Internet. Contracts are backed by data and this paper proposes to get the underlying sensor data in a trustworthy state as the main contract by applying blockchain/hashing methodologies. The Internet of Things race has been characterized initially by a number of independent Maker community projects but now we see the entrance of established tech giants to centralize the collection, storage and use applications. A Dsensor protocol provides a decentralized and secure option those capturing personal data and an open protocol for the grass roots Maker and IofT communities. Sensors by their nature are at the edge of all parts of the network and value can be delivered by DSapps (Decentralized Sensor applications), keeping the value and data in the hands of those that own it.

2 Dsensor Protocol



Contract		DSapps Decentralized Sensor Apps, mapping contracts.
Hashing		Putting on Mini & main blockchain
API		Communication between hardware and Dsensor
Sensor		Internet of Things and beyond data generation

Compliance over time

3 Sensors

The Dsensor protocol should be open and available to all sensors.

4 Connectivity API

Hardware connectivity   Ethernet, WiFi, Cellular, Bluetooth, Zigbee, Netmesh
Build industry strength API based on open source stack. Work with independent projects e.g. http://devices.wolfram.com/ or https://developers.ninja/libraries/nodejs.html http://cylonjs.com/ http://www.opengeospatial.org/

5 Hashing/blockchain

The goal is to collect data from a sensor and turn it in to a trustworthy state. This is the primary defense against data being made up i.e. by making the sensor owner incur the cost of hashing produces a disincentive to generate fictitious data on which to trade. Data successfully enters through hashing can not said to accurately represent any claimed contract description but does give proof the sensor data has adhered to the protocols hashing requirements. Secondary, data integrity checks can be made when data is combined into a pool. For example, google has to deal with millions of fictitious links trying to game their search engine ranking but the over whelming collective intelligence gained from aggregation can help weed out those fictitious links. A similar process can be envisioned for combined data pooling via a more advance Dsensor protocol.

5.1 Hashing ideas to search

Proof of capture: Mathematical proofs built into sensor firmware.
http://www.prismmodelchecker.org/lectures/pmc/ http://www.veriware.org/
http://www.cs.ox.ac.uk/marta.kwiatkowska/
Time limit to get sensor data on Blockchain. Use hashing algorithms of established blockchains.

5.2 Easy to get to blockchain:

The further 'distance' and time it takes for sensor data to hit a blockchain then the less certain we are on its authenticity. Direct hashing based on the electronics chip provides a close opportunity while data entering a blockchain via a third party API is much further way.

5.2.1 Hardware CHIP library

Every sensor will have a dedicated micro controller chip. A hashing and blockchain library will be conceived and built. The leading hardware chip to start research around is the Arm M chipset, http://www.arm.com/images/processor/Cortex-M3-chip-diagram-LG.png Building a universal library for all chip sets will be challenge. See list of hardware in the Appendix.

5.2.1.2 Data API

Two API data options are available:

5.2.1.2.1 Direct from Sensor App:

E.g. from amiigo.com developer API. Depending on how long and how open source the code of the application will impact the trustworthy state of the sensor data.

5.2.1.2.2 From a Data store API:

E.g. from apple health vault. There is very little know on what happens in these secure data stores. Is the data stored exactly as recorded from a sensor? Is any tidying up of the data performed automatically? The only real way to establish trust in such data is when data pool collective intelligence can make some commentary on the data.

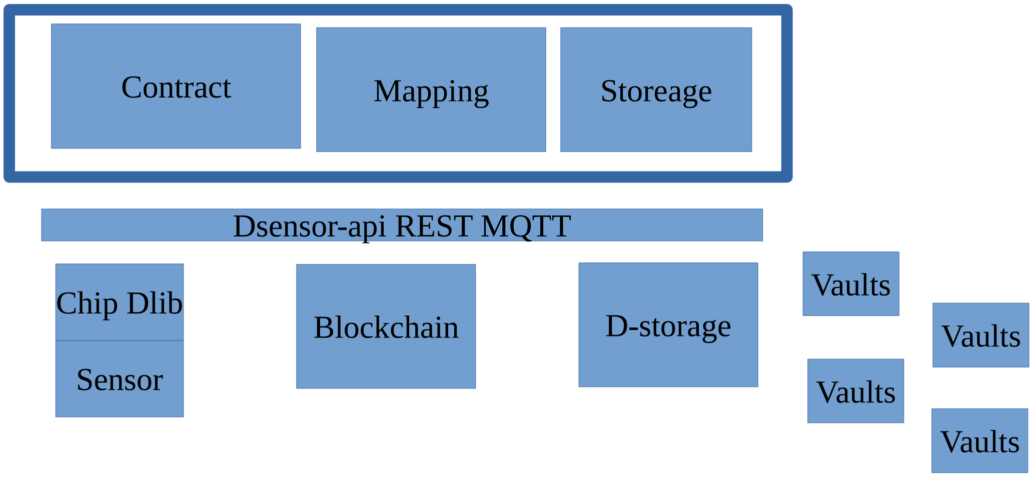
6 DSapps

Decentralized Sensor applications (DSapps) are a special kind of smart contact. The main new capability is **MAPPING** Dsensor data to contract terms in real time. This is a difficult challenge.

MAPPING – making contracts human readable

Sensor data is brittle and human language is more gray i.e. if every tiny breach of sensor data triggers a termination event in a contract then they may become impractical for daily use. To make contracts more robust DSapps will contain a Fuzzy Logic library in the contract builder. Developers will also be free to apply their own Fuzzy logic or other technique into the DSapp builder. See fuzz logic paper
http://www.sciencedirect.com/science/article/pii/S1570870511001326

6.1 DSapp – Dsensor Diagram



6.2 There are two main contract considerations:

6.2.1 getting sensor data on to Mini blockchain and reference made to a public blockchain.

6.2.2 two independent peers making a blockchain contract transaction to share data and/or monetary payments.

Of these the second involves the mapping data to the contract. This involves monitoring, verifying, compliance, security and computability of the data between the two parties. When all the above are enabled, the state is said to be contractable.

Note on outcomes: Two or more peers can come to agreement on a future state of being. Given an agreed sensor tracking conditions are adhered the parties may exchange a monetary consideration. From a purely data point of view the data recorded is the data record.

6.4 Example - Individual with a personal trainer (wellbeing)

An individual joins ABC gym, the DSapp is used to conclude an agreement between the individual and the personal trainer to reduce body fat percentage to 10%. The personal trainer sets out the sensor for measuring body fat percentage at the start and after 3 month period. The training programme is also set out in the contract and the individual agrees to verify completion of each task by sharing their wearable sensor data and attendance of attending a gym. The smart contract is set live.

At the conclusion of the contract the body fat is measured and the outcome is 12%. This is not as low as contracted. The data analysis shows this is due to both none-compliance to the training programme and type of activities suggested in the programme. Both parties accept the outcome and terminate the contract.

7 Estimate of Market

As the first sentence of the document states, sensor data will be the most denominate authoring type of content. The blockchain ecosystem will require a competitive option to the status quo offerings e.g. from Xively etc. The Dsensor protocol, libraries, mapping and DSapps will require a significant developer contribution. Developers need an strong incentive to contribute. Blockchain miners and storage farmers can be paid in crypt-transaction fees. The entire MAPPING, fuzzy logic builders or verification, compliance would be offered as a service fee i.e. the two peers entering into a Dsensor based contract will also pay an amount to a verification service. This could be a fixed fee or related to the outcome.

8 People/skills to reach out to

Blockchain/hashing mathematics
http://www.cs.ox.ac.uk/marta.kwiatkowska/
Bitcoin, Ethereum, Maidsafe etc. reps associated, quarterly review.

9 The Future

With a Dsensor protocol established the foundations will have been laid for more ambitious Smart Contracts applications, namely, DScapps: Decentralised Science Application where concept like the proof of Science can be explored.

10 Conclusions

A Dsensor Protocol will put sensor data on the blockchain in a smart and efficient way, establishing the data as trustworthy. DSapps bring the power of fuzzy logic computing to allow humans to build natural language based contract mapped to sensor data. The volume of sensors and the data they are producing will be vast and this should be acted upon at the edge of the network i.e. for each individual peer. Give peers this ability and the ability share data peer to peer provides a new economic model of value to be established and explored.

Appendix A

Chip sets

TsmarT Embedded C WiFi
SimpleLinks Linux, Android WiFi
IMX53QSB Linux, Android Ethernet
Hitex OM13031 FreeRTOS Ethernet
Beaglebone Linux, Android Ethernet
RX62N FreeRTOS Ethernet
Raspberry Pi Linux Ethernet
Android PC 8750 Android Ethernet
LPCXpresso FreeRTOS Ethernet Optional
ARM mbed LPC1768 Embedded C/C++ Ethernet
Zolertia Z1 Embedded C 6loWPAN
PIC32 Ethernet Kit Embedded C Ethernet
Arduino Uno Arduino Ethernet, WiFi, Cellular
Arduino Due Arduino Ethernet, WiFi, Cellular
Arduino Ethernet
Arduino Ethernet
Arduino Mega 2560
Arduino
Ethernet, WiFi, Cellular
Arduino Leonardo R3
Arduino
Ethernet, WiFi, Cellular
RedBack 1.0 Arduino WiFi
DiamondBack 1.0 Arduino WiFi