What we are looking for in each paper:

1. Background info on IoT
2. Background info on any domain of IoT
3. Generic IoT patterns
4. Domain Specific IoT patters
5. Usages of non-IoT patterns in IoT

Add a commend on any line that is one of these saying which it is and any pertinent information.

Use <https://www.naturalreaders.com/online/> to read the articles aloud while following along.

Add at least one diagram for each of the subdomains.

Domains that I will write about:

Smart Water Management System

BCI

Health Care

-

Smart Metering

Smart Grids

Industrial IoT (such as Smart Cities)

Notes on the articles:

nf

Design Patterns for the Industrial Internet of Things:

* lots of general information about IoT and it’s structure
* several design patterns used in some domains

Fog and IoT- An Overview of Research Opportunities

* networking architecture pattern and reasons for it
* also includes information about general needs of IoT as it grows and limitations of current IoT

mp

An Integrated IoT Architecture for Smart Metering:

* information Smart meters (electricity, water, gas)
* the articles proposition on a system for it (including benefits for the utilities and customers)

IoT Architecture for Smart Grids:

* talks about why we need smart grids (benefits)
* lists lots of types of energy sources and how they connect to IoT (like distribution to local areas, micrograms, smart cities, building, etc.)
* ^as well as storing energy to fix the fluctuations in these energy generation types

Study of IoT - Understanding IoT Architecture, Applications, Issues and Challenges:

* has a lot of domains of IoT (also divided into sub-domains)
* general information about IoT
* pros and cons of IoT
* challenges of IoT

Decentralised IoT Architecture for Efficient Resources Utilisation:

* like the Fog one it talks about removing some of the work from the could due to the increasing work load on them
* ^offers solutions for that issue (such as a more intelligent edge device and more)

p

Dependable design for elderly health care

* Healthcare of the elderly, trustworthiness
* (check the main one to see what patterns are in it)

A simple security architecture for smart water management system:

* Issues with security regarding smart water (for instance the option of physically accessing the devices and not just hack into it computing)
* some general architectural patterns regarding their solution
* security solutions

BCI ontology- A context-based sense and actuation model for brain-computer interactions:

* general info on BCI
* design patterns for BCI
* ontology stuff… (conforming to existing things and so on …)
* integration with AI

Cataloging design patterns for Internet of Things artifact integration:

* a lot of design patterns in general and how they made IoT patterns from existing things
* (I think these can go in the general IoT category)

IoT design patterns- Computational constructs to design, build and engineer edge applications:

* several of general IoT patterns

The subject of this paper, the Internet of Things or IoT put simply is the technological field dealing with interconnected devices over a network. These devices range from a common electric kettle to the cutting edge of green energy wind turbine and everything in between. In fact, it is no exaggeration that the limits of IoT aren’t bound to the Earth itself and has taken flight to the bounds of spaces.

IoT is often thought of as a new up and coming technology and although it certainly has experienced growth spurt in the last decade or so, it can be seen as far back as the 1980’s, from before the term was even coined, in old vending machines. What most see and an internet buzz word some hail as a major component of the fourth Industrial Revolution. As such I would like to discuss why IoT is such an important in today’s world and even more so in the world of tomorrow. What can it do to further society, what fields does it progress?

Like any field especially one so large and diverse there many ways to get a job done, some efficient than others. We learn from our experiences and device better mays to design and develop in the field.

IoT hardware and software component and I would like to focus on the software side. Software design or soft architecture is a vast and highly discussed field in and of itself. It is imperative model our software in a way that is robust, easy to maintain and flexible among many other traits. Software as a general field has much experience with this since the Gang of Four and the original wide spread design patterns became well known. The job of these design patterns was to allow software to be built in such a way that would counter problems that has come up in somewhere to that point, improving the quality and maintainability by a significant margin, but it didn’t stop there. More and more patterns were recognized and as spread of software engineer spread specialized patterns showed up, that is to say, ways of designing software that fit a specific field or sub-field.

IoT is no stranger to this. As the field of IoT developed, software patterns that were specific to IoT appeared one after another, furthermore, patterns for the fields or domains within IoT developed patterns specific to themselves.

Note the term “pattern” that I have been using has been used somewhat ambiguously. Here I am using it to mean an identified design rather than specifically an architectural pattern or design pattern. The difference between the two being their level of abstraction. In fact, some go so far as to include architectural styles under the same umbrella. The difference between a pattern and a style is that a pattern is meant to solves a specific problem while a style is simply a preference for how to compose software. As both deal with software design they may be lumped together as long as it is understood what “pattern” may be referring to. As such there are three levels of abstraction of patterns where architectural styles are the most abstract deal with the general idea of the software but no specifics, architectural patterns are in the middle, dealing with the specific design of the larger systems of the software and design patterns the least abstract specifying the design of individual components.