Pattern language for IoT applications

GULLENA SATISH CHANDRA, Tech Mahindra

Abstract - Internet of Things (IoT) is a discipline that is growing exponentially due to increasing availability of newer connected devices and application areas, day by day. It has been estimated that by 2020 there would be about 50 billion connected devices. Going forward, there would a need to build new applications or extend old applications that encompass IoT devices. Hence, it would help to have a pattern based approach to building IoT based applications. A pattern language is a method of describing good design practices or patterns of useful organization within a field of expertise. A pattern language can also be an attempt to express the deeper wisdom of what brings aliveness within a particular field of human endeavor, through a set of interconnected patterns. A variety of IoT related design patterns and programming idioms have been published over a period of time. As generic reference architecture for IoT based applications follows a layered architecture style, we have developed a catalog of design patterns and classified within architecture layers. The catalog is an initial effort of bringing together the published design patterns, as of date. However, we do not claim that it is exhaustive and would endeavor to enrich it based on feedback and further research in our future work. Further, catalogs of patterns are not enough because the designer does not know when and where to apply them, especially in a large complex system. We show a way to use these classifications through pattern diagrams where a designer can navigate to perform her pattern selection.

Keywords: Internet of Things, Design Patterns, Pattern Language, Pattern classification, Pattern Sequence

1. INTRODUCTION

As an emerging and fast growing discipline, the discipline of Internet of Things (IoT) has a wide range of applications, including smart cities, precision agriculture, environment monitoring, smart buildings, health care, industrial and military applications.

From a business perspective, IoT based applications address a variety of problem domains. However, from the technology perspective there are time tested approaches that have an underlying theme for development of IoT based applications from a technology perspective. These have been documented as design patterns.

A variety of IoT design patterns have been described in the literature, over a period of time. Several catalogs have been created but they vary in terms of their level of abstraction and in the rigor of defining the patterns. Further research needs to be done to develop more IoT design patterns and also enhance the existing ones so that they conform to the established pattern template documentation standards.

IoT based applications normally follow a layered architecture style. In this paper, we have classified the design patterns in accordance with the concern they address, which is represented by a particular layer.

There are huge opportunities but considerable challenges in designing IoT applications. These challenges range from the provisioning of ultra-low power operation and system design using modular, composable components to smart automation. Furthermore, the advancement in sensor instrumentation requires an efficient stream data processing. [1]

In response to such challenges, a variety of design patterns have been observed and documented. The design patterns help architects and designers build IoT applications of various levels of complexity.
The organization of the paper is as described in this paragraph. Section 2 defines the key concepts being discussed in the paper to set the context for the rest of the paper. Section 3 discusses the normally followed architecture style for solutions to IoT based

applications, i.e. the layered style. Section 4 classifies the existing design patterns in accordance with the layers of IoT based architectures. Section 5 discusses pattern language and sequence in the context of IoT based applications and provides a diagrammatic representation of a pattern sequence. Finally, Section 6 concludes the paper and presents thoughts on future work and research directions.

The objectives of this paper are threefold. First, do a literature survey to find existing published patterns. Second, to classify the patterns at a higher level of granularity i.e. in an abstraction space that is in between design patterns and architecture styles/patterns. Third, to present a pattern sequence for solutions to IoT based applications.

2. DEFINITIONS

2.1 Patterns and pattern language

Patterns describe a problem and then offer a solution.

A pattern language can also be an attempt to express the deeper wisdom of what brings aliveness within a particular field of human endeavor, through a set of interconnected patterns [2].

The development of complete pattern languages is an optimistic but worthwhile goal. Such languages provide solutions to all design problems that can occur in the respective domains. Christopher Alexander claims to have done this for areas in architecture.

Pattern languages already exist for small sub-domains of software design, for example the CHECKS pattern language for information integrity. It will be exciting to see how far the pattern community travels along this road. Even if we do not reach completeness in a strict sense, it would be very beneficial to have pattern languages that cover a substantial part of the design space of the respective domains [3].

A pattern language is both a process and a thing. The thing of a pattern language can be any domain in software. Here we have considered the IoT Applications as the domain of investigation for this paper.

The domain-specific process to create the thing of a pattern language is defined by the language's pattern network and the creation processes for the patterns within this network:

- One or more patterns define the entry point into the language.
- The creation process for the chosen entry point pattern describes how to resolve this pattern's problem, if feasible by using other patterns.
- Navigating the network, descending from the entry pattern, and applying the creation processes of the visited patterns defines a pattern sequence that creates the system under development.

The process defined by a pattern language gives concrete and precise guidance in developing systems for a specific domain:

- What are the key problems to be resolved?
- In what general order should the problems be tackled?

- What alternatives exist for resolving a specific problem?
- How are mutual dependencies between the problems to be handled?
- How is each individual problem resolved most optimally in the presence of its surrounding problems?

This concreteness and domain-orientation differentiates the processes defined by pattern languages from general purpose processes.

2.2 Pattern Language and Sequence

In [4] the relationship between pattern language and sequence has been described as “A sequence is one path through a pattern language. It is the process we follow to build something. After, we got a critical mass of patterns (in an area), we could arrange them in a language (static), and we could trace sequences through the language. The sequences we trace may well have come from the sequences that lodged themselves in our brains earlier.”

2.3 Internet of Things

The internet of things (IoT) is the network of physical devices, vehicles, buildings and other items—embedded with electronics, software, sensors, actuators, and network connectivity that enable these objects to collect and exchange data. The Global Standards Initiative on Internet of Things (IoT-GSI) defined the IoT as "the infrastructure of the information society." The IoT allows objects to be sensed and controlled remotely across existing network infrastructure, creating opportunities for more direct integration of the physical world into computer-based systems, and resulting in improved efficiency, accuracy and economic benefit; when IoT is augmented with sensors and actuators, the technology becomes an instance of the more general class of cyber-physical systems, which also encompasses technologies such as smart grids, smart homes, intelligent transportation and smart cities. Each thing is uniquely identifiable through its embedded computing system but is able to interoperate within the existing Internet infrastructure. [5].

3. ARCHITECTURE STYLE FOR APPLICATIONS BASED ON IOT

As in [6], in the Layered View the system is viewed as a complex heterogeneous entity that can be decomposed into interacting parts. The concerns addressed by this view are:

- What are the parts that make up the whole system?
- How do these parts interact with each other?
- How do the parts perform their functionality and still remain decoupled from each other?
- How are the quality attributes of modifiability, scalability, and integrability supported?

The individual parts of the system are components that are decoupled as much as possible from one another. The interaction mechanisms between the components are implemented through connectors that include appropriate interfaces, states, and interaction protocols. There is usually an overall control mechanism that maintains an overall organization scheme by orchestrating the various components.

Broadly, the layers of the architecture for IoT based applications should address the following features:

1. Connectivity, management and governance of devices and sensors
2. Communication of data gathered through the devices within a local network and/or propagating it over the internet to a remote server/cloud
3. Communication of messages and alerts
4. Security
5. Data ingestion and storage of data that is structured, semi-structured and unstructured
6. Perform analytics and predictive analytics
7. Expose data and functionality as web services for consumption by other layers as well as external systems
8. Enable application management
9. Infrastructure management
10. Rendering dashboards and reports through a User Interface layer
11. Store data in a centralized database on-premise or on cloud

4. IOT DESIGN PATTERNS

Literature survey of the design patterns for IoT based applications has shown that various researchers and organizations have developed and documented IoT based design patterns. However, they vary in terms of their level of abstraction and in the rigor of defining the patterns.

For instance, Michael Koster has documented a good number of IoT design patterns. But, there is scope to improve them to meet the rigor of design patterns' definition. Soheil Qanbari, et al. provides rigorous definitions related to the edge applications.

Further research needs to be done to develop more IoT design patterns and also enhance the existing ones so that they conform to the established pattern template documentation standards.

In the following table, we have mapped the IoT Design Patterns to related IoT Architecture Layers.

IoT Architecture Layer	Design Pattern(s)
Edge Devices - Device Integration, Management and Governance	Edge Provisioning Pattern, Edge Code Deployment Pattern, Edge Orchestration Pattern and Edge Diameter of Things (DoT) Pattern – [1]
Communication	Request/Response, Event Subscription, Asynchronous Messaging, Reliable Messaging, Multicasting, Publish/Subscribe, Queues, Message Brokers, Federation, Discovery and Delegation of Trust – [7] REST, Asynchronous Events, Resource Binding, Observer Pattern, Publish/Subscribe, Broker, Proxy, Protocol Bridge, Resource Discovery, Resource Registration and

	Sleeping/Non-reachable Endpoint – [8]
Security	Access control using data models, Social to physical graph relationship, PGP and asymmetric public-key cryptography on devices, DTLS over UDP, End-to-end encryption and Device Management. – [8]
Local Network	Peer-to-peer, Smartphone as gateway and Application gateway pattern – [8]
Integration	Middleware Platform - [8]
IoT Cloud	Cloud-based apps, Asset-based apps, Distributed IoT apps, Digital Twin and Social IoT – [9]
Infrastructure	6LowPAN edge router, WSN access point, Mesh routing, Application gateway, Behind-NAT connectivity and M2M WAN – [8]
Monitoring	Monitor a large number of devices over a large area – [8]
Application	Virtualization, Virtualization through middleware, Software connected to thing via a network and Thing-thing interaction – [8]
Information model	Structured data, Web Objects, Virtual Objects-Smart Objects, Composite Objects, Hypermedia-HATEOAS, Semantic Hyperlinks, Information model, Context model, Binding model, Resource Directory - Catalog, Resource constructor, and Access control model – [8]

Table I. Patterns classification based on Architectural Concerns/Layers

5. PATTERN LANGUAGE AND SEQUENCE FOR IOT APPLICATIONS

Patterns provide non-domain experts a means to access a domain expert's knowledge. Patterns can be applied at all levels of the software development lifecycle. Moreover, a single pattern may not be enough to describe the solution to a set of problems. Pattern languages provide a means to specify how a set of interconnected patterns can be used together to solve a set of related problems [3]. As discussed in Section 2.2, a pattern sequence is a path in a pattern language.

The following figure is a representation of the pattern sequence that is a path of the pattern language for solutions to IoT based applications.

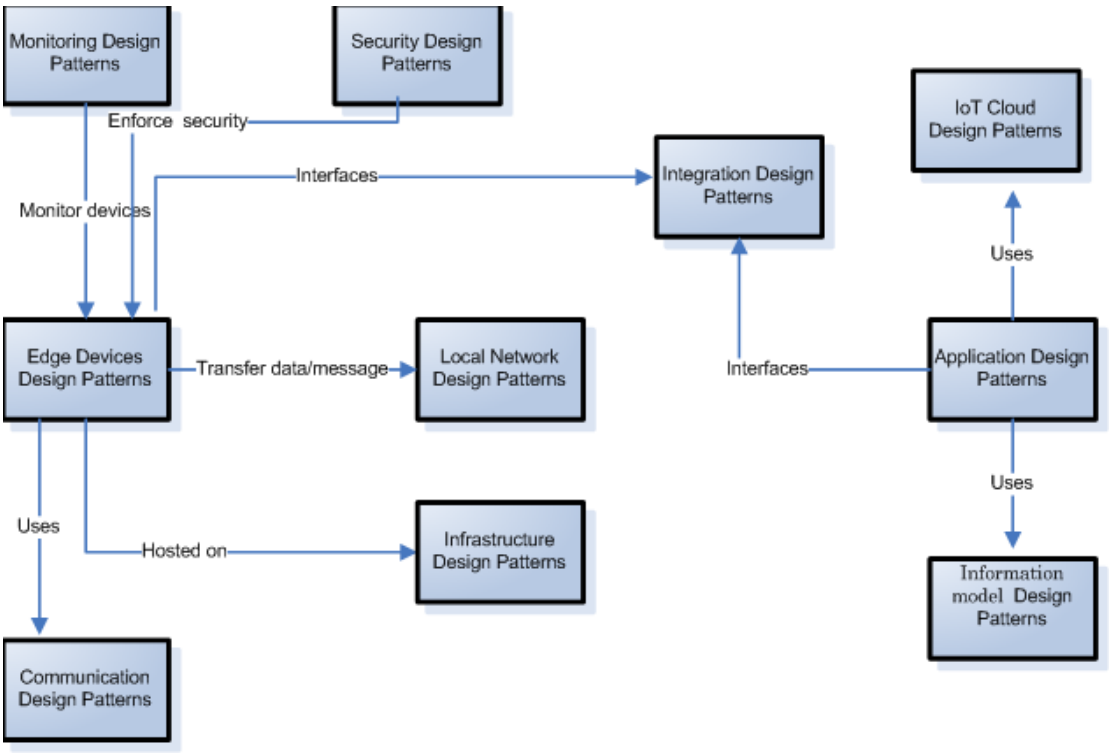

Fig 1 Pattern sequence for solutions to IoT based applications

The rectangles in the figure denote a class of design patterns. The arrows between the rectangles show the relationship between the rectangles along with the nature of relationship. A normal starting point for the navigation of the pattern sequence is the "Edge Devices Design Patterns".

We will take up an example of a connected car to illustrate at a high level how the pattern sequence for solutions to IoT based applications is navigated.

Various aspects of a connected car, ranging from the diagnostics and predictive features of the car itself to the behavior of the car's driver can be addressed by IoT based applications.

Data from vehicles, devices and systems can be combined into an integrated data platform to enable a range of innovative solutions that deliver greater safety, service and convenience. Large volumes of in-vehicle and repair data can be collected and analyzed in order to predict and prevent problems before they result in mechanical failure. Security can be infused into every step of the vehicle lifecycle – from design to production, and through supply chain and maintenance ecosystem. One can detect and mitigate attempted attacks on vehicle control and telematics systems, ensure secure integration with third-party service providers, and deliver the safety, privacy and convenience that consumers expect of todays connected vehicles. [10]

To address the connected car scenarios mentioned above, we can follow a sequence of selection of appropriate edge device(s) -> network connectivity -> infrastructure ->

integration -> application -> IoT cloud -> Information model. Further the patterns related to UI can be leveraged to select the appropriate user interface.

In this connection, the patterns presented in Table 1 can be considered during design. For e.g. the contenxt of “IoT devices are usually scattered geographically, sometimes hard to reach and large in number. Operation managers and developers must be able to reconfigure devices or provision new ones in an efficient way and have preconfigured nodes. ” would lead to the selection of the “Edge provisioning pattern” in [1].

6. CONCLUSIONS AND FUTURE WORK

Patterns can be classified according to many viewpoints. A good classification can make their selection easier and more precise [11]. We have taken an approach of classifying patterns based on the various layers of the Layered Architecture Layer that IoT based applications normally follow.

This approach imposes a top-down mechanism of ensuring that relevant design patterns have been identified for all the layers. As each layer deals with a set of specific concerns, for e.g. the focus of the edge devices layer is on devices and for the communication layer the focus is on is on communication mechanism. Hence, Subject Matter Experts (SMEs) can focus on identification of design patterns in their area of expertise. Thus, if SMEs are asked to comment on the patterns that form part of the layer corresponding to their expertise then additional design patterns could emerge.

We do not claim to have presented an exhaustive list of the design patterns or architecture layers in this paper. The primary aim of this paper is to present an approach that should help IoT architects and designers architect and design complex IoT based applications. Future work will include refining and development of further design patterns and classifications presented in the paper. We plan to extend this research and make the list as exhaustive as possible and also create an online repository that would be revised at periodical intervals.

REFERENCES

1. Soheil Qanbari, Samim Pezeshki, Rozita Raisi, Samira Mahdizadeh, Rabee Rahimzadeh, Negar Behinaein, Fada Mahmoudi, Shiva Ayoubzadeh, Parham Fazlali, Keyvan Roshani, Azalia Yaghini, Mozhdeh Amiri, Ashkan Farivarmoheb, Arash Zamani and Schahram Dustdar. IoT Design Patterns: Computational Constructs to Design, Build and Engineer Edge Applications, IEEE First International Conference on Internet-of-Things Design and Implementation, 2016.

2. https://en.wikipedia.org/wiki/Pattern_language

3. Frank Buschmann, Regine Meuneir, Hans Rohnert, Peter Sommerlad and Michael Stal - John Wiley & Sons.Pattern Oriented Software Architecture: A System of Patterns Vol 1.

4. Neil B. Harrison and James O. Coplien. Pattern Sequences. http://hillside.net/europlop/HillsideEurope/Papers/EuroPLoP2001/2001_HarrisonEtAl_PatternSequences.pdf

5. http://www.gartner.com/it-glossary/internet-of-things/

6. Paris Avgeriou and Uwe Zdun. Architectural Patterns Revisited - A Pattern Language. https://www.researchgate.net/publication/221034694_Architectural_Patterns_Revisited__A_Pattern_Language

7. Peter N. Communication Patterns for the Internet of Things, June 1, 2016. https://software.intel.com/en-us/articles/communication-patterns-for-the-internet-of-things

8. Michael Koster. Design Patterns for an Internet Of Things - A Design Pattern Framework for IoT Architecture. http://iot-datamodels.blogspot.in/2014/05/design-patterns-for-internet-of-things.html

9. Greg Gorbach. Five application patterns for the IoT Cloud, March 14, 2016. https://industrial-iot.com/2016/03/five-application-patterns-iot-cloud

10. http://www-935.ibm.com/industries/automotive/connected_car.html

11. Eduardo B. Fernández, Hironori Washizaki, Nobukazu Yoshioka and Yoshiaki Fukazawa: Classifying Security Patterns. Conference: Progress in WWW Research and Development, 10th Asia-Pacific Web Conference, APWeb 2008, Shenyang, China, April 26-28, 2008. Proceedings.