

Context-Aware Accessor

Marcus Chow, Michael Rodriguez and Anne H. Ngu
Department of Computer Science at Texas State University

Objectives

Our objective is to create an accessor that will be able to embrace the heterogeneity of middleware. Our Context-Aware Accessor will provide:

- a unified framework where a user can access different middleware without specific programming and protocol knowledge
- an easier way to develop IoT applications through Ptolemy's modeling system

Introduction

The Internet of Things (IoT) allows devices connected to sensors and software to produce data which can be stored and analyzed. IoT must be supported by middleware which is a software system designed to be the intermediary between physical devices and applications. A de facto middleware is difficult to obtain since there are many IoT middleware available and each has its own specifications and use of protocols to interface with. This makes developing an application that incorporates many middleware very tedious. We developed a Context-Aware Accessor that enables multiple middleware to be easily integrated into a single IoT application.

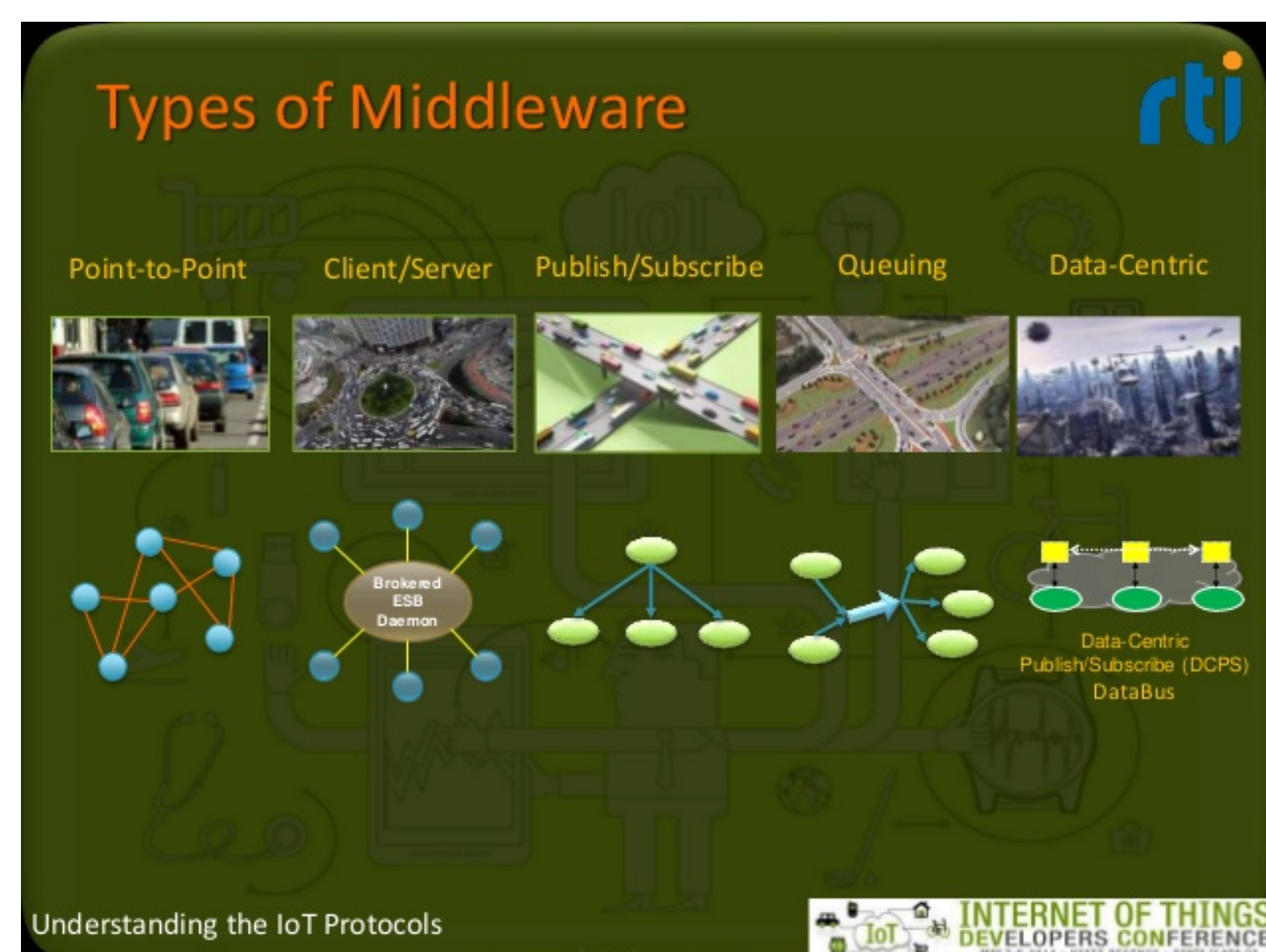


Figure 1: Different IoT protocols

Ptolemy II

- Ptolemy II is a modeling system that connects devices through an actor-oriented design.
- This system comes with preexisting actors/accessors which allow for a simple way to design and create an IoT application.

Approach

According to "A Vision of Swarmlets" an accessor is an actor that provides access to a service, sensor, or actuator. However, these accessors are static and can only be used for a single function. An accessor would need to be reprogrammed if the user needed a different functionality.

We propose the design and development of a Context-Aware Accessor which gives contextual information about IoT middleware and the corresponding physical devices deployed on it.

Context facilitates automatic configuration of the accessor at design time without customized coding of the accessor. The context-aware accessor can assume the behavior of a selected sensor in a selected middleware by a simple annotation.

Context-Aware Accessor

- Combines resources from different available middleware under one framework
- Has the ability to change functionality at run time according to the parameters given.
- GUI framework allows user to change the given context dynamically without any programming knowledge

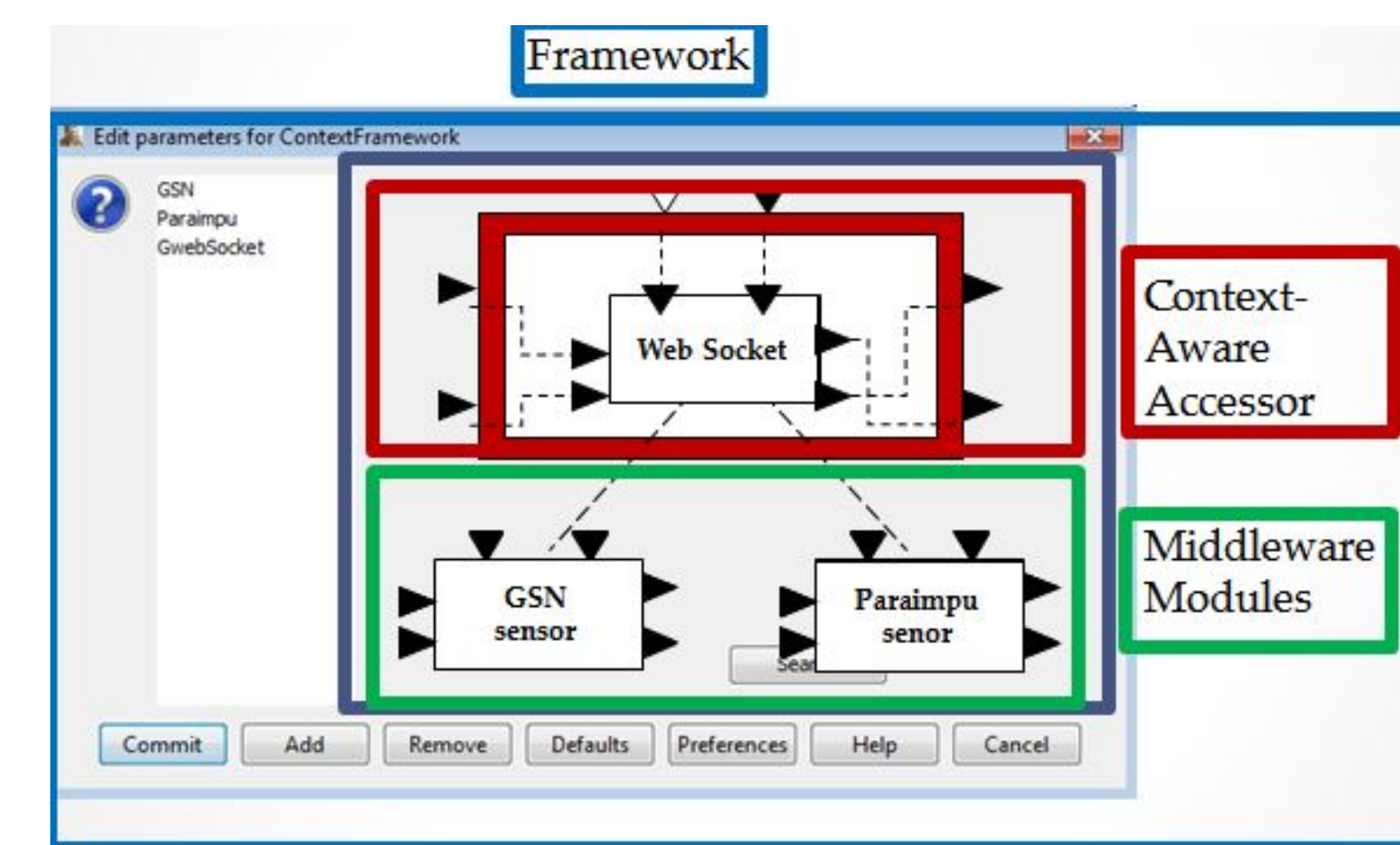


Figure 2: Context-Aware Accessor

Sound and Wind Sensor

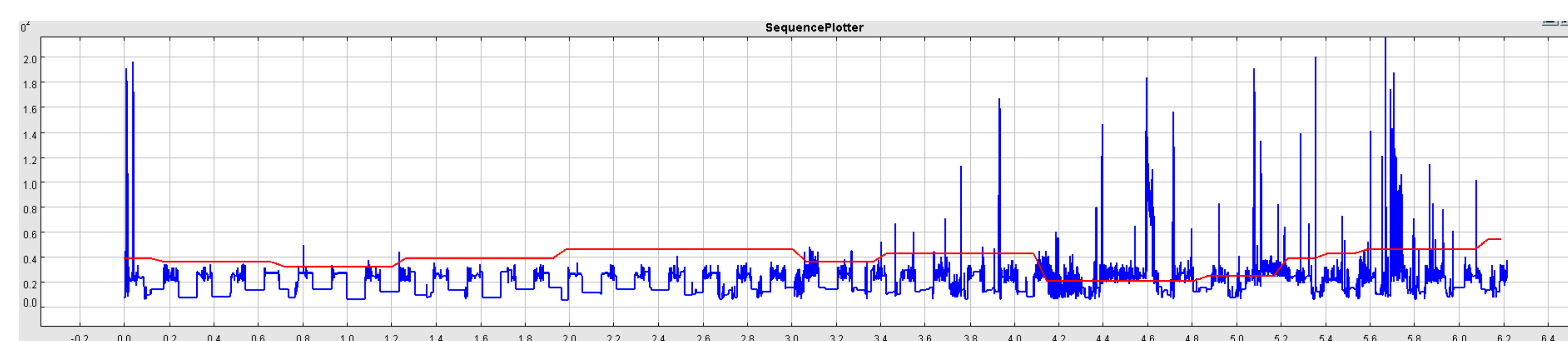


Figure 3: Data collected from our Sound and Wind sensors over a single day

IoT Application

We designed a simple IoT application as an example of how our Context-Aware Accessor can be used with different middleware and protocols within a single application. The Context-Aware Accessor can have different functions such as:

- Getting sound data from a GSM server using HTTP to track microwave usage
- A wind sensor that is connected to Paraimpu using HTTP
- Posting data to Twitter using Paraimpu
- Retrieving Clock signal through TCP web socket

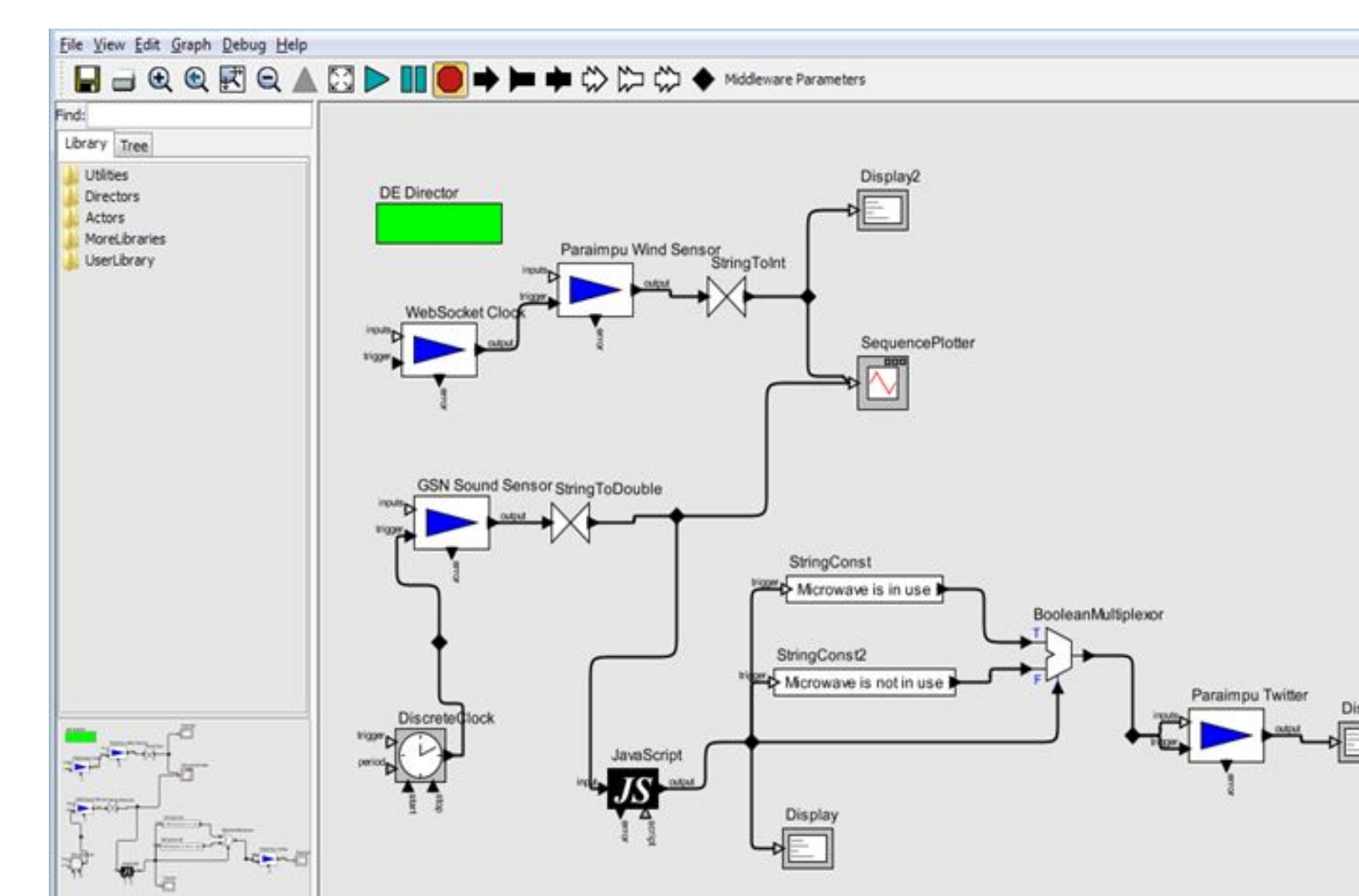


Figure 4: Ptolemy Model of an IoT application

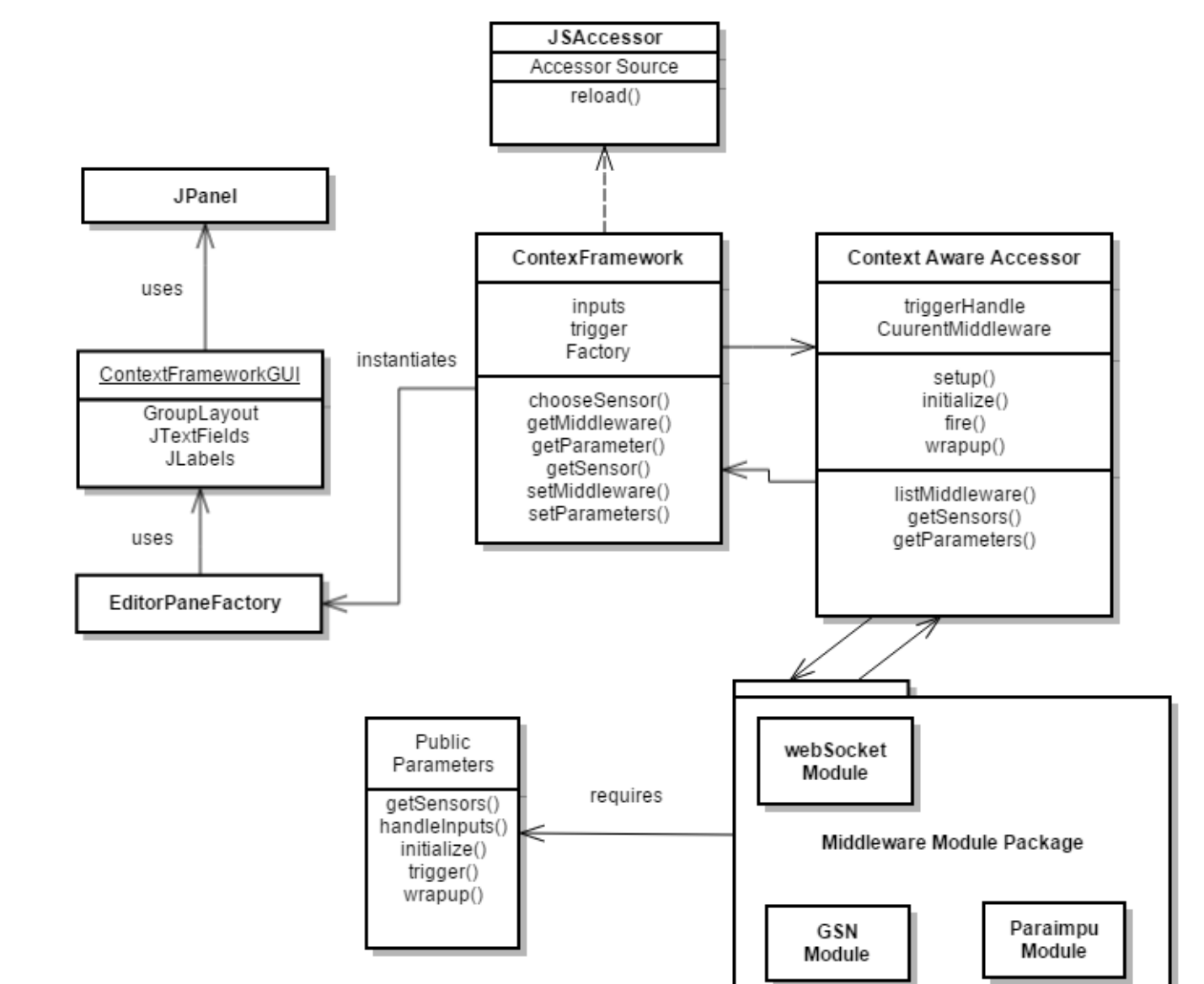


Figure 5: UML class diagram for Context-Aware Accessor

Conclusion

Developing IoT applications can be difficult because of the diversity of middleware. Current designs of accessing middleware are limited to what they can connect to because they must be hard coded for a specific device or service. Our Context-Aware Accessor can be dynamically altered to enable the use of any middleware or protocol.

References

Acknowledgements

We thank the National Science Foundation for funding the research under the Research Experiences for Undergraduates Program (CNS-1358939) at Texas State University to perform this piece of work and the infrastructure provided by a NSF-CRI 1305302 award.

Contact Information

- Email: mchow22@ucmerced.edu
- Email: mmr94@txstate.edu
- Email: angu@txstate.edu