**LAON IoT System**

|  |
| --- |
| Architecture Document |

LGE-CMU-LAON-IoT-V1.00

Team 4(Laon) LGE & CMU

Copyright © 2015 Laon Team, LG Electronics Inc. All Rights Reserved.

Though every care has been taken to ensure the accuracy of this document, LG Electronics Inc. cannot accept responsibility for any errors or omissions or for any loss occurred to any person, whether legal or natural, from acting, or refraining from action, as a result of the information contained herein. Information in this document is subject to change at any time without obligation to notify any person of such changes.

LG Electronics Inc. may have patents or patent pending applications, trademarks copyrights or other intellectual property rights covering subject matter in this document. The furnishing of this document does not give the recipient or reader any license to these patents, trademarks copyrights or other intellectual property rights.

No part of this document may be communicated, distributed, reproduced or transmitted in any form or by any means, electronic or mechanical or otherwise, for any purpose, without the prior written permission of LG Electronics Inc.

The document is subject to revision without further notice.

All brand names and product names mentioned in this document are trademarks or registered trademarks of their respective owners.

**Revision History**

| Document  Version | Date | Comment | Author | Reviewer |
| --- | --- | --- | --- | --- |
| 0.1 | Jun. 22, 2015 | Initial Version | YB.Choe | JY.Woo, SC.Kwon  SW.Lim, YM.Kim YJ.Jang |
| 0.2 | Jun. 22, 2015 | Merge first level system  Decomposition design. | YB.Choe | JY.Woo, SC.Kwon  SW.Lim, YM.Kim  YJ.Jang |
| 0.3 | Jun.23, 2015 | Add second level system decomposition design. | YB.Choe | JY.Woo, SC.Kwon  SW.Lim, YM.Kim  YJ.Jang |
| 0.4 | Jun.24,2015 | Add third system decomposition design. | YB.Choe | JY.Woo, SC.Kwon  SW.Lim, YM.Kim  YJ.Jang |
| 0.5 | Jun.24,2015 | Add fourth system decomposition design. | YB.Choe | JY.Woo, SC.Kwon  SW.Lim, YM.Kim  YJ.Jang |

**Contents**

[About This Documents 6](#_Toc423035596)

[Purpose of This Document 6](#_Toc423035597)

[Document Scope 6](#_Toc423035598)

[Audience 6](#_Toc423035599)

[Terminology 6](#_Toc423035600)

[Definition 6](#_Toc423035601)

[Acronyms 6](#_Toc423035602)

[Contact Information 6](#_Toc423035603)

[1 Project Overview 7](#_Toc423035604)

[2 Architectural Drivers 8](#_Toc423035605)

[2.1. Context 8](#_Toc423035608)

[2.1.1. Market context 8](#_Toc423035612)

[2.1.2. Organizational context 8](#_Toc423035613)

[2.2. Constraints 9](#_Toc423035614)

[2.2.1. Business constraint 9](#_Toc423035616)

[2.2.2. Technical constraints 9](#_Toc423035617)

[2.3. High Level Functionality 10](#_Toc423035618)

[2.2.3. Functional Requirements 10](#_Toc423035619)

[2.2.4. Use Case 12](#_Toc423035620)

[2.4. Quality Attributes 16](#_Toc423035621)

[2.4.1. Priority Scale 16](#_Toc423035622)

[2.4.2. Difficulty ranking scale 16](#_Toc423035623)

[2.4.3. Quality Attributes Priority 17](#_Toc423035624)

[2.4.4. Important Quality Attributes Priority 18](#_Toc423035625)

[2.4.5. Quality Attributes Scenarios 18](#_Toc423035626)

[3 System Context 22](#_Toc423035630)

[3.1. System Boundary 22](#_Toc423035632)

[3.2. System Scope 22](#_Toc423035633)

[4 Architecture Decomposition 23](#_Toc423035634)

[4.1. First Level Decomposition of Laon IoT system 23](#_Toc423035636)

[4.1.1. Dynamic Perspective – C&C view 23](#_Toc423035640)

[4.1.2. Static Perspective – Module view 24](#_Toc423035641)

[4.2. Second Level Decomposition 25](#_Toc423035642)

[4.2.1. SA Node 25](#_Toc423035644)

[4.2.2. Laon IoT System Server 29](#_Toc423035645)

[4.3. Third Level Decomposition 32](#_Toc423035646)

[4.3.1. Event Bus 32](#_Toc423035648)

[4.3.2. External entity 36](#_Toc423035649)

[4.4. Fourth Level Decomposition 39](#_Toc423035650)

[4.4.1. Web Application 39](#_Toc423035652)

[4.4.2. Heartbeat 40](#_Toc423035653)

[4.4.3. Event Tunnel 42](#_Toc423035654)

[4.4.4. Oracle 43](#_Toc423035655)

[5 Future works 45](#_Toc423035656)

**FIGURE**

[Figure 1 Conceptual IoT System 6](#_Toc423031840)

[Figure 2 System Context 21](#_Toc423031841)

[Figure 3 C&C View of Laon IoT system 22](#_Toc423031842)

[Figure 4 Module view of Laon IoT System 23](#_Toc423031843)

[Figure 5 C&C View of SA Node Initializing 24](#_Toc423031844)

[Figure 6 C&C View of SA Node Running 25](#_Toc423031845)

[Figure 7 C&C View of Running (call from Laon IoT Server) 25](#_Toc423031846)

[Figure 8 Module view of SA node 26](#_Toc423031847)

[Figure 9 Static perspectives (Generalization Style) of SA Node 27](#_Toc423031848)

[Figure 10 Static Perspective (Use and Decomposition style) of SA Node 28](#_Toc423031849)

[Figure 11 C&C View of Laon IoT Server 28](#_Toc423031850)

[Figure 12 Module View of Laon IoT System Server 29](#_Toc423031851)

[Figure 13 Allocation View of Laon IoT System Server 30](#_Toc423031852)

[Figure 14 Event Bus C&C View 31](#_Toc423031853)

[Figure 15 Sequence diagram of Registration request from external entities 32](#_Toc423031854)

[Figure 16 Sequence Diagram of "Report join / exit status from external entity" 33](#_Toc423031855)

[Figure 17 Sequence Diagram of reporting SA Node sensor values and user commands 33](#_Toc423031856)

[Figure 18 Sequence Diagram of notifying the situation to users 34](#_Toc423031857)

[Figure 19 Module view of event bus 34](#_Toc423031858)

[Figure 20 C&C View of External Entity 35](#_Toc423031859)

[Figure 21 Module View of External Entity (Use Style) 36](#_Toc423031860)

[Figure 22 Module View of External Entity (Class Diagram) 37](#_Toc423031861)

[Figure 23 C&C View of Web Application 38](#_Toc423031862)

[Figure 24 C&C View of Heartbeat 39](#_Toc423031863)

[Figure 25 Class Diagram of Heartbeat 40](#_Toc423031864)

[Figure 26 Class Diagram of Event Bus 41](#_Toc423031865)

[Figure 27 Class Diagram of Oracle 42](#_Toc423031866)

About This Documents

## Purpose of This Document

The purpose of the document is to describe the architectural and detail design of Laon IoT system.

## Document Scope

Scope of this document is limited to describe the architectural drivers and architecture of the Laon IoT system. It includes architectural drivers; high level functionality, quality attributes, quality attribute scenarios and constraints as well as architecture specification which include dynamic, static and physical perspectives.

## Audience

The target audience includes software designers and engineers responsible for designing and developing the Laon IoT system and maintaining it

## Terminology

Definitions, Acronyms used in this document are as follows.

## Definition

N.A

## Acronyms

|  |  |
| --- | --- |
| **Acronyms** | **Description** |
| IoT | Internet of Things |
| SMS | Short Message Service |
| SDK | Software Development Kit |
| QA | Quality Attribute |

## Contact Information

Contact us at ym.kim@lge.com. Please feel free to leave any comments, questions, suggestions, or problems you may have.

1. ****Project Overview****

**The key business goals** are to enter the IoT market successfully by providing following features.

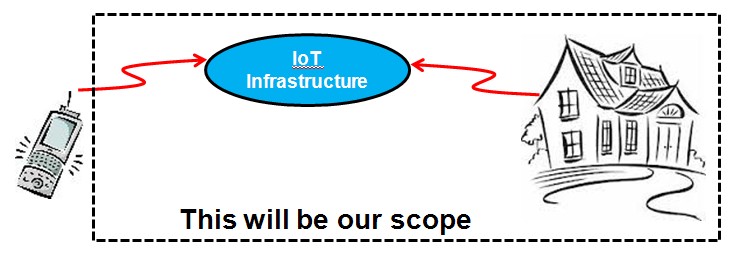
1. a system/infrastructure that enables users to communicate with sensors and actuators installed in a building via any mobile device or desktop system connected to the Internet
2. an infrastructure to support an open ecosystem of hardware device and software application developers, service providers, and installers and maintainers
3. a basic data centric infrastructure that will support the collection of data for future analytic operations and services  
     
   

Figure 1 Conceptual IoT System

1. Architectural Drivers

In this chapter, we describe architectural drivers which include context, constraint, high level functional requirements, use case of functional requirement, quality attributes and scenario of quality attributes.



## Context



### Market context

|  |  |
| --- | --- |
| Customer/Stake holders | * Product Manager * Team members * Sensor/Actuator producers. * Third party service providers – packaging, installing, maintaining, providing data services, etc. * Consumers. * Homebuilders. * Smart appliance producers. * Utilities companies. |
| Functional expectations | To build an IoT infrastructure that will support access to home sensors and actuators. |
| How quickly you must design and deliver new products(Time to market) | The target date is 25/06/2015. |
| Notions of quality | The infrastructure should support Do-It-Yourself (DIY) customers, or contractors enabling them to procure easily install IoT products for use in homes or businesses. |
| Price of products and  services | The competitors have their own price policies. Refer to following texts and the web-site.   * Apple App Store : 30% of the App price * <https://blog.profitbricks.com/top-49-tools-internet-of-things/> * [Dragino V2 MS14-S with M32 IoT Module](http://www.amazon.com/Dragino-MS14-S-M32-IoT-Module/dp/B00ODETFI2/ref=pd_sim_sbs_147_2?ie=UTF8&refRID=0703X7AWBXXYE18ZRV8A) : $65.90 |
| Product packaging | The package comprises SA nodes, Sensors, SDK for 3rd party vendors, user manual and the home server. These products will be available at DIY stores such as Lowes, Home Depot, and so forth. |

### Organizational context

|  |  |
| --- | --- |
| Structure | This team comprises following experts in their own fields. YUMI KIM(Leader), YONG JAE JANG, SEUNG CHAN KWON, WOO JUHYUNG, YONGBONG CHOE, SANG WON LIM  And, Jeffrey Gennari(Product Manager) |
| Culture | Our culture can be explained by one word, "Laon" means just "pleasure". |

## Constraints



### Business constraint

|  |  |
| --- | --- |
| ID | Description |
| BC-01 | You may use 3rd party open source SW but please check with the project manager and business developer (course instructor). |
| BC-02 | The system should be delivered in 5 weeks and the team is consisted of 6 members. |

### Technical constraints

|  |  |
| --- | --- |
| ID | Description |
| TC-01 | Permissible languages for this system (excluding the nodes) include Java and Python. |
| TC-02 | The node should be implemented on Arduino Uno and WIFI Shield. |

## High Level Functionality

In this section, we describe systems high level functional requirements that system must to do. Functional requirements are from project description document and several meetings with customers to clarify requirements.

### **Functional Requirement**s

Following functional requirements are recognized from the project description document.

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Descriptions | Priority | Status |
| FR-01 | Shall allow users to query the home to find out both 1) how many nodes are installed 2) what sensors/actuators are installed on each node | Essential | approved |
| FR-02 | Shall allow users to control 1) turn on and off lights 2) open and close the door 3) turn on and off the alarm | Essential | implemented |
| FR-03 | Shall allow users to determine 1) temperature 2) humidity 3) presence 4) door status 5) alarm status 6) light status | Essential | implemented |
| FR-04 | Shall log sensor values for one year. | Essential | implemented |
| FR-14 | Shall log all user commands for (configurable) one year. | Essential | implemented |
| FR-05 | Shall allow user applications to review their sensor and command history. | Essential | approved |
| FR-06 | Shall not allow unauthorized persons to access 1) the home sensors/actuators 2) any data generated by sensors/actuators 3) any data stored | Essential | implemented |
| FR-07 | Shall not allow unauthorized persons to register a sensor that they do not own | Essential | implemented |
| FR-08 | Shall send an emergency message (SMS) to registered users (i.e, house owner and/or police officer) when the door is manually opened while alarmed  SMS contains below  the house is manually opened while alarmed. The house address is 'xxx'. | Essential | implemented |
| FR-15 | Shall send an emergency message (SMS) to registered users (i.e., house owner and/or police officer) when the house is suddenly occupied while alarmed SMS contains below  the house is suddenly occupied while alarmed. The house address is 'xxx'. | Essential | implemented |
| FR-09 | Shall not allow automatic door opening while the house is alarmed. | Essential | implemented |

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Descriptions | Priority | Status |
| FR-10 | Shall send a message (SMS) to the user to inform them when the house is vacant and not alarmed. It should ask them if they want to alarm the home. Shall lock (door closed and alarmed) the house, if they do not respond within 5 minutes. If a door is open, it will close the door before alarming the house. SMS contains below the house is not alarmed and vacant. Please check it. The house address is 'xxx'. | Essential | implemented |
| FR-10 | Shall send a message (SMS) to the user to inform them when the house is vacant and not alarmed. It should ask them if they want to alarm the home. Shall lock (door closed and alarmed) the house, if they do not respond within 5 minutes. If a door is open, it will close the door before alarming the house. SMS contains below the house is not alarmed and vacant. Please check it. The house address is 'xxx'. | Essential | implemented |
| FR-11 | Shall turn off the lights when no one is home and 10 minutes elapses. | Essential | implemented |
| FR-12 | Shall add and remove nodes to and from the system without having to restart the system or other nodes. This includes registration and recognition of the type of sensors and actuators provided by the node. | Essential | implemented |
| FR-13 | Should support to build custom apps, services, and/or make mashups from existing available services. | optional | Deleted |
| FR-19 | Shall send an emergency message (SMS) to registered users (i.e., house owner and/or police officer) when the node is disconnected(i.e., physical removal) or network shut-down SMS contains below  the 'SA Node XXX' on the house is disconnected. The house address is 'xxx'. | conditional | implemented |
| FR-16 | When mailbox receives something, the system sends SMS message to registered number.  SMS contains below some mail is received on the house. The house address is 'xxx'. | conditional | implemented |
| FR-17 | Shall not allow setting alarm while door is opening. | conditional | defined |
| FR-18 | The system restores the last setting value when it's turned on. | conditional | implemented |

### Use Case

Describe detail use case scenario of each use case to get more details of functional requirements and to understand the requirement well.

#### UC-01 ‘Shall allow users to query the home to find out’

|  |  |
| --- | --- |
| Use case ID | UC-01 |
| Use case title | FR-01 - ‘Shall allow users to query the home to find out ’ |
| Stakeholders | User |
| Description | 1) The user would starts the application on own devices such as a laptop and a mobile phone. Then the node lists are shown on the screen. 2) The user selects a node to see what sensors/actuators are installed on the selected node. 3) Then the information which sensors/actuators are installed would be displayed. |

#### UC-02 ‘Shall allow users to control

|  |  |
| --- | --- |
| **Use case ID** | **UC-02** |
| Use case title | FR-02 - ‘Shall allow users to control ’ |
| Stakeholders | User |
| Description | After the information which sensors/actuators are installed would be displayed, its icon would indicate the current status (on/off, open/closed). Then, 1) The user would select the lights or the alarm to turn on/off it. 2) The user would select the door to open/close it. After the user selection, each actuator would work well and the icon on the user's screen would be toggled. |

#### UC-03 Shall allow users to determine 1) temperature 2) humidity 3) presence 4) door status 5) alarm status 6) light status

|  |  |
| --- | --- |
| Use case ID | UC-03 |
| Use case title | FR-03 - ‘Shall allow users to determine 1) temperature 2) humidity 3) presence 4) door status 5) alarm status 6) light status’ |
| Stakeholders | User |
| Description | 1) The user starts the application. 2) Login screen will be displayed and then the user inputs ID and password If it is correct, application display status of sensors and actuators on screen. |

#### UC-04 ‘shall store sensor values every for one year.

|  |  |
| --- | --- |
| Use case ID | UC-04 |
| Use case title | FR-04 - ‘Shall store sensor values for one year |
| Stakeholders | Server |
| Description | 1) SA node sends status of sensors and actuator whenever the status is changed. 2) Server updates this information in the repository. 3) Server checks that the information has passed for one year in the repository. If there are, Server will remove that information. |

#### UC-14 ‘Shall log all user commands for one (configurable) year.

|  |  |
| --- | --- |
| Use case ID | UC-14 |
| Use case title | FR-14 - ‘Shall log all user commands for one (configurable) year.’ |
| Stakeholders | User |
| Description | 1) The user change status of actuators on screen 2) The application send control message to the Server 3) Server update this information in repository. 4) Server checks that the information has passed for one year in repository. If there are, Server will remove that information. 5) Server send control message to the SA node |

#### UC-05 -'System shall allow user applications to review their sensor and command history'.

|  |  |
| --- | --- |
| Use case ID | UC-05 |
| Use case title | FR-05 -'System shall allow user applications to review their sensor and command history'. |
| Stakeholders | User |
| Description | 1) The application starts and main menu is displayed. 2) The user login to the system through the application. 3) The user can review their sensor value history on the application. 4) The user can review their command history on the application. 5) The user selects to exit the application. |

#### UC-06 'System shall not allow unauthorized persons to access'.

|  |  |
| --- | --- |
| Use case ID | UC-06 |
| Use  case title | FR-06 System Shall not allow unauthorized persons to access |
| Stakeholders | Malicious user |
| Description | 1) The application starts and main menu is displayed. 2) The user attempt to login to the system through the application but user fail to login. 3) The user attempt to access home sensors or actuators but the attempt failed. 4) The user attempt to generate data by sensors or actuators but the attempt failed. 5) The user attempt to store some data but the attempt failed. 6) The user elects to exit the application. |

#### UC-07 ‘Shall not allow unauthorized persons to register a sensor that they do not own’.

|  |  |
| --- | --- |
| Use case ID | UC-07 |
| Use  case title | FR-07 - ‘Shall not allow unauthorized person to register a sensor that they do not own’ |
| Stakeholders | User |
| Description | 1) The user would search a node that can be registered. 2) The user could not see a sensor if he is not unauthorized. |

#### UC-08 ‘Shall send an emergency message’.

|  |  |
| --- | --- |
| Use case ID | UC-08 |
| Use  case title | FR-08 - ‘Shall send an emergency message when the door is manually opened while alarmed’ |
| Stakeholders | User |
| Description | 1) The door is manually opened while alarmed. 2) The user will see emergency message. |

#### UC-15 ‘Shall send an emergency message’.

|  |  |
| --- | --- |
| Use case ID | UC-15 |
| Use  case title | FR-15 - ‘Shall send an emergency message when the house is suddenly occupied while alarmed’ |
| Stakeholders | User |
| Description | 1) The house is suddenly occupied while alarmed. 2) The user will see emergency message. |

#### UC-09 ‘Shall not allow automatic door opening while the house is alarmed.

|  |  |
| --- | --- |
| Use case ID | UC-09 |
| Use  case title | FR-09 Shall not allow automatic door opening while the house is alarmed. |
| Stakeholders | User |
| Description | 1) The user would like to open door from remote terminal such as smartphone or desktop via the system while alarmed. 2) The system should inform the user that house is alarmed. 3) The user disables alarm. 4) The user requests the door open again. 5) The system opens the door. |

#### UC-10 ‘Shall not allow automatic door opening while the house is alarmed.

|  |  |
| --- | --- |
| Use case ID | UC-10 |
| Use  case title | FR-10 Shall sends a message to the user to inform them when the house is vacant and not alarmed. |
| Stakeholders | User |
| Description | 1) The user goes out not enabling alarm. 2) The system checks vacant and alarm status. 3) If house is vacant and not alarmed, the system will send a message such as SMS or email to user(s) registered on the system. 4) The user can login the system and enable alarm of house after receiving the message. 5) If the user(s) don't respond within 5 minutes, the system shall close the door if it is open and will enable the alarm. |

#### UC-11 'Shall turn off the lights when no one is home and 10 minutes elapses.

|  |  |
| --- | --- |
| Use case ID | UC-11 |
| Use case title | FR-11 - 'Shall turn off the lights when no one is home and 10 minutes elapses.' |
| Stakeholders | Server |
| Description | 1) Server checks the status of presence or proximity sensor. 2) If no one is at home during 10 minutes, the server shall turn off the lights. |

#### UC-12 ‘Shall add and remove nodes to and from the system’.

|  |  |
| --- | --- |
| Use case ID | UC-12 |
| Use  case title | FR-12 - ‘Shall add and remove nodes to and from the system’ |
| Stakeholders | User |
| Description | 1) The user would search nodes. 2) The user would select nodes to add or remove to or from the searched lists. 3) The user could get the result from the system when nodes are added or removed to or from the node lists. |

#### UC-13 ‘Should support to build custom apps, services, and/or make mashups from existing available services.

|  |  |
| --- | --- |
| Use case ID | UC-13 |
| Use  case title | FR-13 - ‘Should support to build custom apps, services, and/or make mashups from existing available services.’ |
| Stakeholders | Server |
| Description | 1) Application/node developer requests the API documents to us. 2) Application/node developer make application or mashup service with the API documents provided by us. 3) The application/node works well with our system. |

## Quality Attributes

This chapter describes quality attributes and detail scenario of the each quality attributes

### **Priority Scale**

|  |  |  |  |
| --- | --- | --- | --- |
| Priority | Priority Value | Description | Score |
| H | Must Have | Must be present in the end product at all costs | 3 |
| M | Nice to Have | Customer would greatly appreciate implementation of these features. | 2 |
| L | If there is time | Reconsider them if customer deems them important enough. | 1 |

### **Difficulty ranking scale**

|  |  |  |
| --- | --- | --- |
| Priority | Description | Score |
| H | High complexity and large amount of effort required | 1 |
| M | High complexity or large amount of effort required | 2 |
| L | Moderate complexity and medium amount of effort required | 3 |

### **Quality Attributes Priority**

Make quality attribute priority using quality attribute utility tree.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | Quality Attribute | Attribute Refinement | Descriptions | Priority | Difficulty | Score |
| QA01 | Security | Confidentiality integrity Availability | The system shall allow user applications to review their sensor and command history in a secure and private way. | M | H | 3 |
| QA02 | Security | Authorization | The system shall provide secure registration of the type of sensors and actuators provided by the node. The system shall not allow unauthorized persons to register a sensor that they do not own | H | H | 4 |
| QA09 | Security | Authentication | The system shall provide some authentication method for user. | M | M | 4 |
| QA03 | Security | Authorization | The system shall not allow unauthorized persons to access 1) the home sensors/actuators 2) any data generated by sensors/actuators 3) any data stored | H | M | 5 |
| QA04 | Usability | Number of  tasks  accomplished | The system shall provide easy ways to add and remove nodes to and from the system within 5 step tasks. | M | M | 4 |
| QA05 | Availability | No  downtime | The system shall provide to add and remove nodes to and from the system without having to restart the system or other nodes. | M | L | 5 |
| QA06 | Scalability | Number  of nodes | Assume that a home can have one or more nodes up to 100. | H | H | 4 |
| QA07 | Inter-operability | Discover  service | The system shall be easy for application developers to build custom apps, services, and/or make mashups from existing available services. Then, percentage of information exchanges between the system and custom services correctly processed or rejected is over 99.99%. | M | H | 3 |
| QA08 | Modifiability | Coupling | SA nodes currently utilize 802.11 the system should make it easy to add emerging protocols with modifying only 1 element. (e g. it is expected that we will add Bluetooth 802.15 products in the future) | L | M | 3 |

### **Important Quality Attributes Priority**

From now, we consider QA02, QA03 and QA05 as key quality attributes, which have high score and high priority. To meet the selected Quality Attributes, Priority should be considered during system design

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | Quality Attribute | Attribute Refinement | Descriptions | Priority | Difficulty | Score |
| QA02 | Security | Authorization | The system shall provide secure registration of the type of sensors and actuators provided by the node. The system shall not allow unauthorized persons to register a sensor that they do not own | H | H | 4 |
| QA03 | Security | Authorization | The system shall not allow unauthorized persons to access 1) the home sensors/actuators 2) any data generated by sensors/actuators 3) any data stored | H | M | 5 |
| QA05 | Availability | No  downtime | The system shall provide to add and remove nodes to and from the system without having to restart the system or other nodes. | M | L | 5 |

### Quality Attributes Scenarios



#### QA-01 Security - Data packet security

|  |  |
| --- | --- |
| ID | QA-01 Security - Data packet security |
| Scenario (s) | The system shall allow user applications to review their sensor and command history in a secure and private way. |
| Stimulus | Attempt to display the data |
| Stimulus Source | Unauthorized user |
| Environmental Condition (s) | normal operation of the system |
| Artifact (if known) | Data produced / consumed by the system |
| Response | process data |
| Response Measures | The attempt to sniff the network packet is failed 100% of the time. |

#### QA-02 Security - Authorization

|  |  |
| --- | --- |
| ID | QA02. Security - Authorization |
| Scenario (s) | The system shall provide secure registration and recognition of the type of sensors and actuators provided by the node. The system shall not allow unauthorized persons to register a sensor that they do not own |
| Stimulus | A request for a new node registration |
| Stimulus Source | Unauthorized User |
| Environmental Condition (s) | User node should be connected to the internet through Wi-Fi |
| Artifact (if known) | The system. |
| Response | The registration is success or failure. |
| Response Measures | The attempt to register the new node is rejected 100% of the time. |

#### QA-03 Security - Authentication

|  |  |
| --- | --- |
| ID | QA03 Authentication (to be update) |
| Scenario (s) | The system shall not allow unauthorized persons to access 1) the home sensors/actuators 2) any data generated by sensors/actuators 3) any data stored |
| Stimulus | Request to login with random ID and password. |
| Source of Stimulus | Unauthorized user |
| Environmental Condition (s) | User node should be connected to the internet through Wi-Fi. |
| Artifact (if known) | The system. |
| Response | 1) If ID and password is wrong, message means login is refused. 2) If ID and password is right, message means login is permitted. |
| Response Measures | Estimate of the time needed to get permission is 3 years it is validated by formula. |

#### QA-04 Usability - Number of tasks accomplished

|  |  |
| --- | --- |
| ID | QA04 Usability |
| Scenario (s) | The system shall provide easy ways to add and remove nodes to and from the system within 5 step tasks.  1. User login Laon IoT system 2. Register new IoT device ID  3. Power on new IoT device  4. Enter the IoT device's PWD  5. Check IoT device status |
| Stimulus | attempt to add or remove nodes |
| Source of Stimulus | User |
| Environmental Condition (s) | The system is running on the network. |
| Artifact (if known) | The system. |
| Response | Steps to complete the task. |
| **Response Measures** | The task is done within 5 steps. |

#### QA-05 Availability

|  |  |
| --- | --- |
| ID | QA05 Availability |
| Scenario (s) | The system shall provide to add and remove nodes to and from the system without having to restart the system or other nodes. |
| Stimulus | Install new nodes in house or remove nodes. |
| Source of Stimulus | User |
| Environmental Condition (s) | The system is already deployed and working well. The network channel is working well. |
| Artifact (if known) | The system. |
| Response | To complete the process to install and remove nodes. |
| Response Measures | system downtime is under 12 minutes during 90 days |

#### QA-6 Scalability - Number of nodes

|  |  |
| --- | --- |
| ID | QA06 Scalability |
| Scenario (s) | The system shall support more than 100 nodes. |
| Stimulus | Installation of new SA node |
| Stimulus Source | User |
| Environmental Condition (s) | More than 100 nodes are installed in the system. |
| Artifact (if known) | The system |
| Response | Successful registration, communication and operation of SA node. |
| Response Measures | new nodes can be added to the system and operate without any additional software, recompilation, or special procedures |

#### QA-07 Interoperability - Discover service

|  |  |
| --- | --- |
| ID | QA07 Interoperability - Discover service |
| Scenario (s) | The system shall be easy for application developers to build custom apps, services, and/or make mashups from existing available services. |
| Stimulus | install SA nodes from 3rd party |
| Stimulus Source | user |
| Environmental Condition (s) | the system is operating |
| Artifact (if known) | The system |
| Response | successful registration, communication and operation of SA nodes |
| Response Measures | new 3rd party nodes can be added to the system and operate without any additional software, recompilation, or special procedures |

#### QA-08 Modifiability

|  |  |
| --- | --- |
| ID | QA08 Modifiability |
| Scenario (s) | SA nodes currently utilize 802.11 the system should make it easy to add emerging protocols with modifying only 1 element. (e g. it is expected that we will add Bluetooth 802.15 products in the future) |
| Stimulus | the system should support emerging protocols such as Bluetooth 802.15 |
| Stimulus Source | User |
| Artifact (if known) | The system |
| Environmental Condition (s) | The system has been developed to support 802.11 protocol the software is already deployed. |
| Response | Necessary man-month to support new protocol |
| Response Measures | less than 5MM |

1. System Context

In this section, we describe Laon IoT system context diagram and system scope.



## System Boundary

This system we create comprises SA node and a Server. This is shown on the following context diagram. The system interacts with sensors, actuators, terminal applications, 3rd party nodes and 3rd.

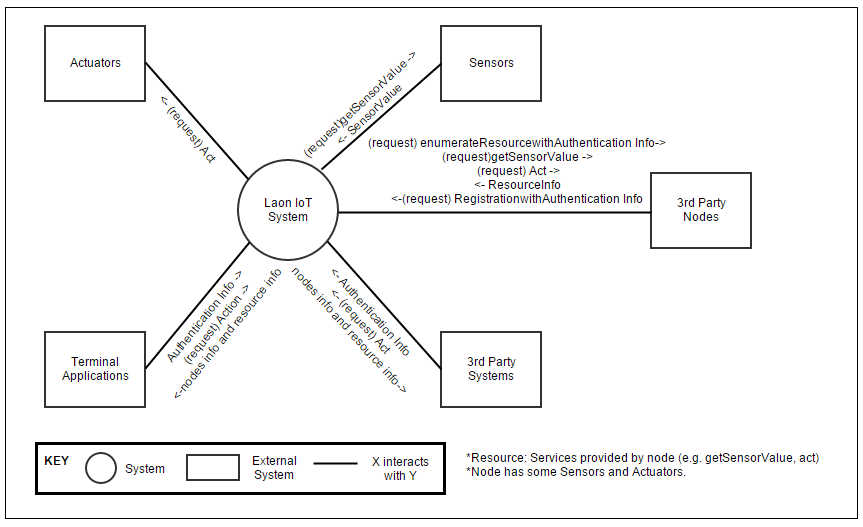


Figure 2 System Context

## System Scope

The scope of Laon IoT system development is as below

|  |  |
| --- | --- |
| Elements | Description |
| Laon IoT System | Laon IoT System which interact with actuator, sensor and 3rd party system. |
| Actuator/Sensor Controller | A controller part of Laon IoT system which control sensor and actuator |
| 3rd Party Node Interface | The Interface provided by Laon IoT system for 3rd party node. |
| 3rd Party System Interface | The Interface provided by Laon IoT system for 3rd party system |

1. Architecture Decomposition

## First Level Decomposition of Laon IoT system



### Dynamic Perspective – C&C view

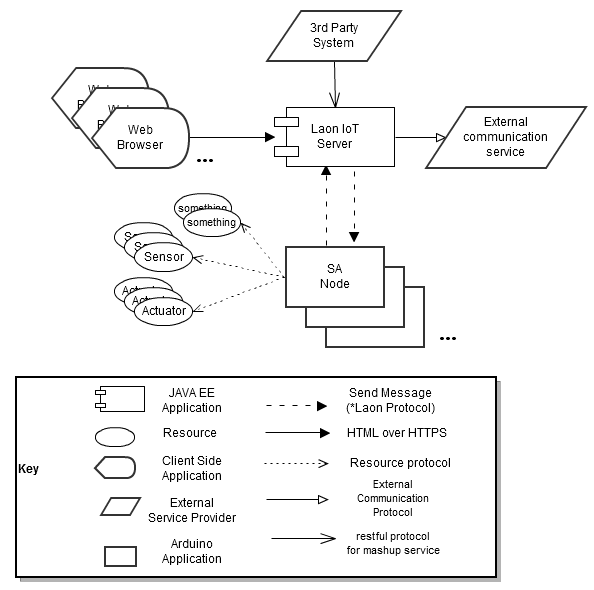


Figure 3 C&C View of Laon IoT system

#### Rationale

Server/client pattern promotes scalability and central service promotes security. Web Browser promote usability through omitting install application step

#### Responsibility

|  |  |
| --- | --- |
| Component Name | Responsibility |
| Laon IoT Server | * The Laon IoT Server Manage SA Nodes * The Server Control SA Node as agent of web browser * The Server is a proxy(Cache) of SA Node * The Server provides user authentication & authorization.  (E.g. Server permits controlling SA Node to authorized user only.) * The Server store SA node events * The Server stores user command history. * The Server stores sensor values. * The Server provides web-service for third parties. * The Server allows using AES encryption data. |
| SA Node | * SA Node allows controlling resources by Laon IoT Server. * SA Node communicates with many resources using resources protocol. (e.g. Bluetooth) * SA Node tries to register at the Laon IoT Server actively. * SA Node sends events to server actively. * SA Node sends description of resource service to Laon IoT server. * SA Node recognize some situation based on events and trigger some predefined action * SA Node provides AES encryption. |

### Static Perspective – Module view

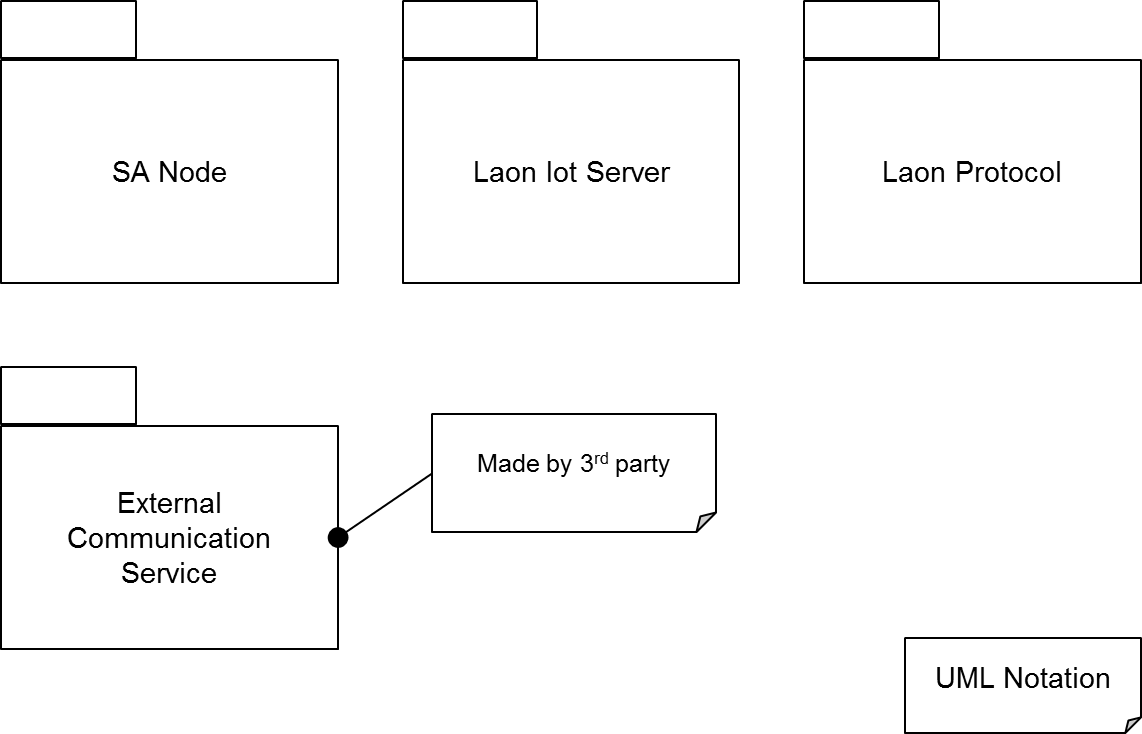


Figure 4 Module view of Laon IoT System

## Second Level Decomposition



### SA Node

#### Dynamic Perspective – C&C view

##### **C&C View of Initializing**

SA Node calls Register. And The Register processes SA node registration steps.

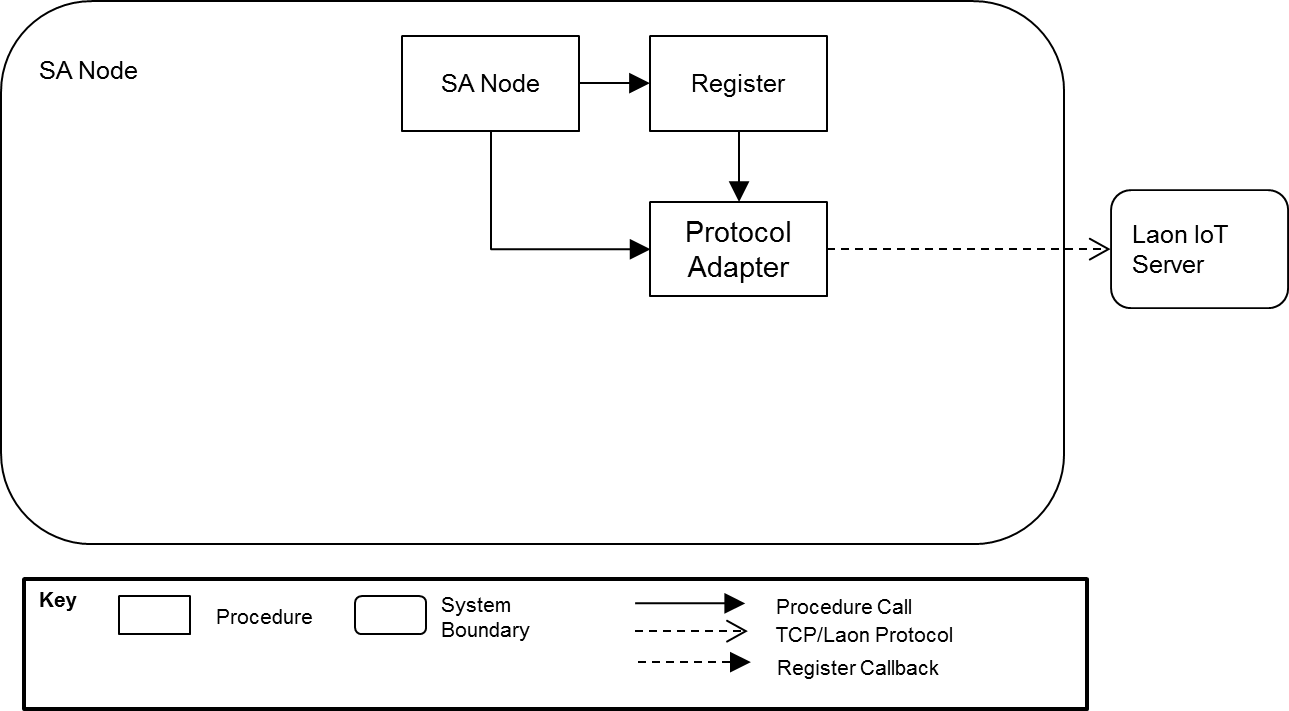


Figure 5 C&C View of SA Node Initializing

##### **C&C View of Running (Event Sending)**

The Event Handler calls Object Broker periodically.  
If Object’s status is changed, Event Handler call Protocol adapter to send event to Laon IoT server.   
Object broker call policy handler to check the policy when it calls objects. (E.g. Humidity, Presence…)

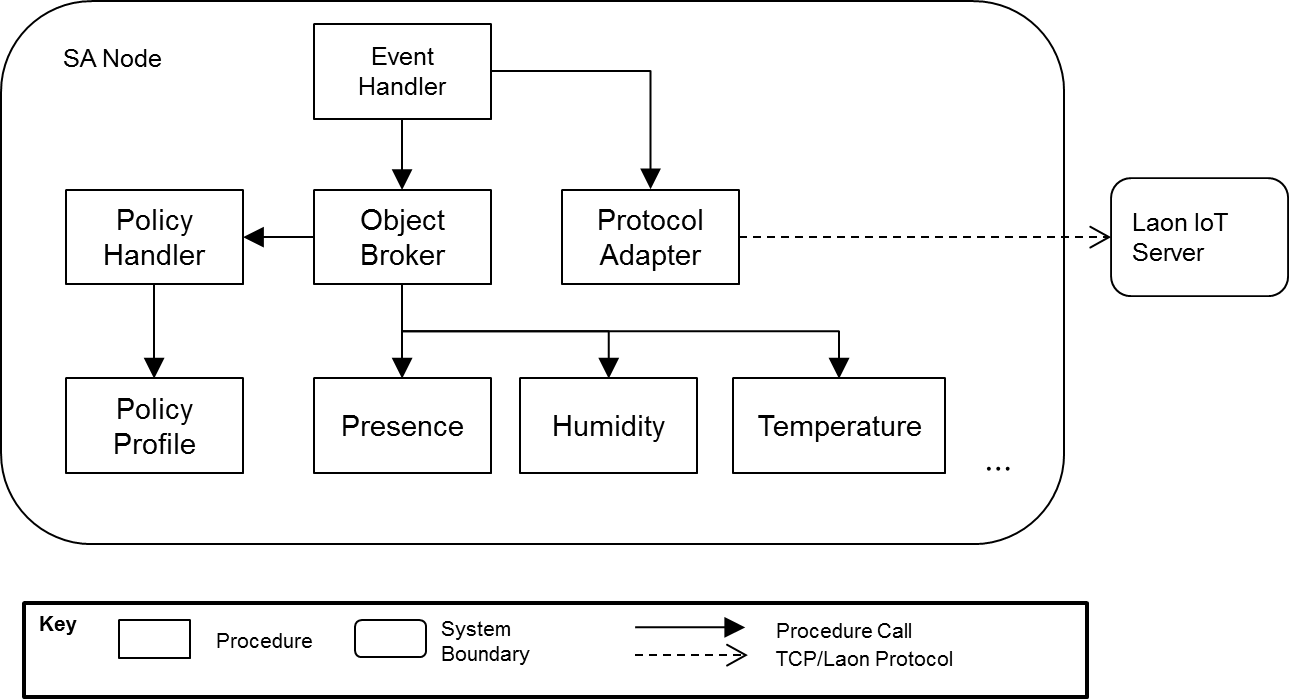


Figure 6 C&C View of SA Node Running

#### C&C View of Running (call from Laon IoT Server)

Laon IoT Server send request to Protocol Adapter. Protocol adapter interpret the request.   
SA Node read the request and call object broker to process the request.   
Object broker call specific object on the request and return result. Object broker call policy handler to check the policy when it calls objects. (E.g. Humidity, Presence…)

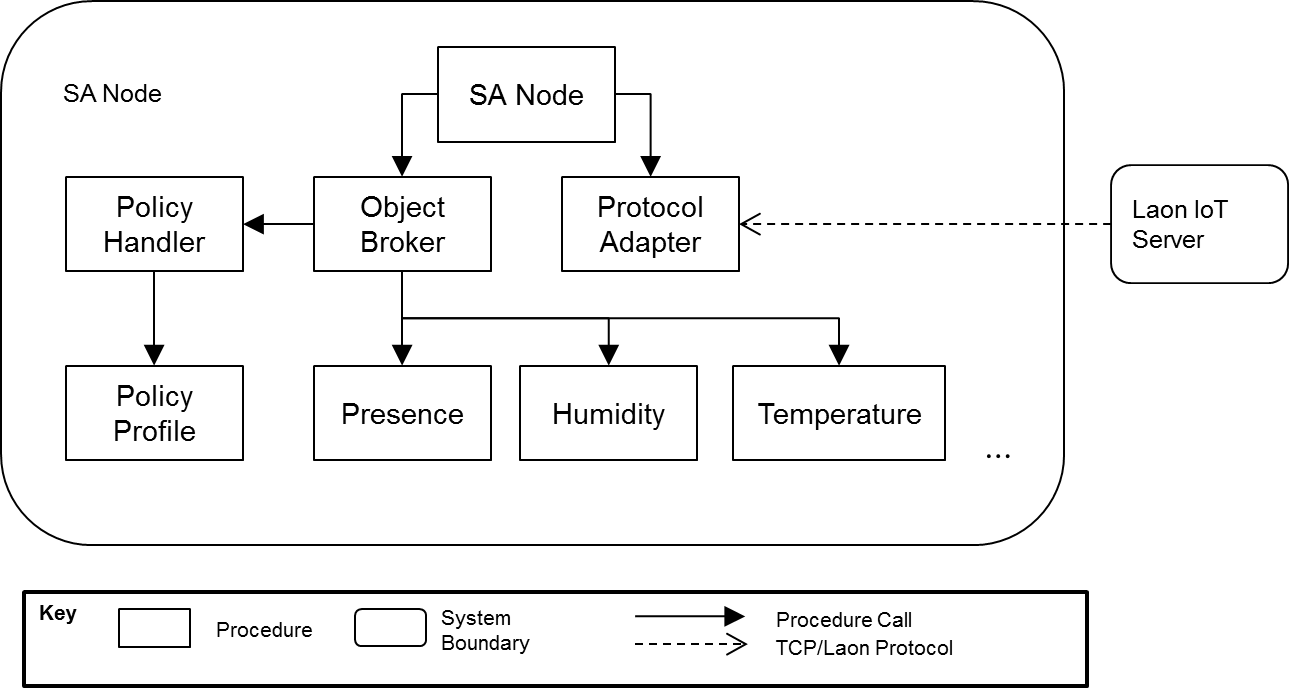


Figure 7 C&C View of Running (call from Laon IoT Server)

#### Static Perspective – Module view

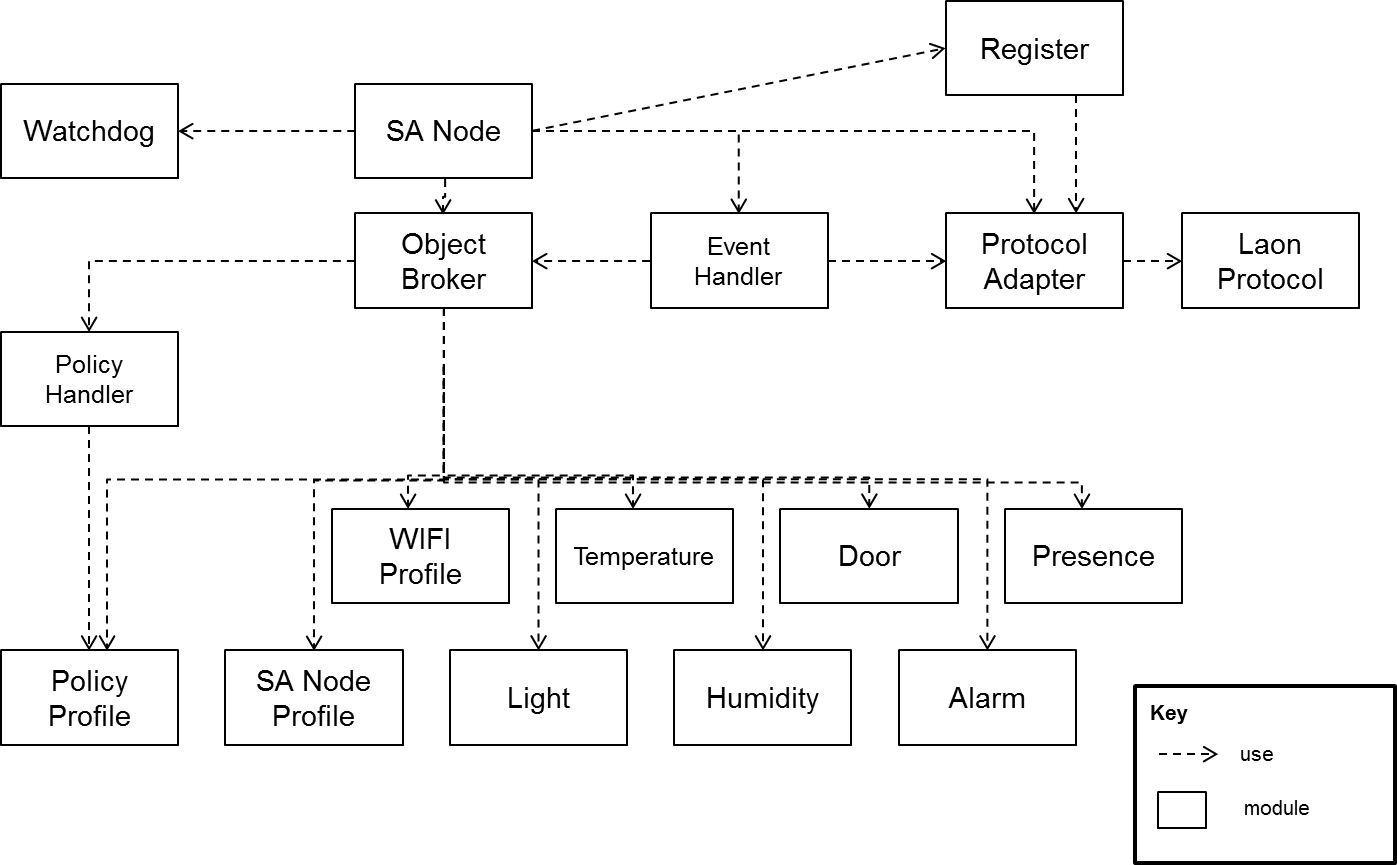


Figure 8 Module view of SA node

##### **Rationale**

Stable module is used by unstable module.   
Resource type modules are separated for reusability.  
Change effect is mitigated by adapter or broker.

##### **Responsibility**

|  |  |
| --- | --- |
| Component Name | Responsibility |
| SA Node | * SA Node module coordinates SA Node’s services. |
| Watchdog | * It checks system running condition and send confirm message to hardware watchdog. If the hardware watchdog doesn’t receive the message it reboots the system. |
| Register | * It register SA node at predefined event bus. It sends SA node’s authentication information to the event bus and receives the event bus’s authentication information. Register validates the event bus’s authentication information. |
| Object Broker | * It manages and controls resources. It’s an agency. It sends a command to specific resource. It calls back registered function point. |
| SA Node Profile | * SA Node profile manages MAC address, time, time zone, manufacturer name, and model name and software version. It provides schema of resources. |
| Policy Profile Handler | * It queries to object broker to get resource state and recognizes some condition based on policy profile and trigger predefined action on policy profile. |
| Policy Profile | * It manages policy profile. Policy profile can describe some condition based on resources state, time period and predefined action based on resources of SA Nodes. Policy profiles describe relationship between condition and action. |
| Event Handler | * Event Handler calls Object Broker periodically. If object’s status is changed, Event Handler call Protocol adapter to send event to the Laon IoT Server. |
| Protocol Adapter | * Adapting protocol. |
| Wi-Fi Profile | * It manages Wi-Fi configuration. |
| The others | * Module (Light, Temperature, Humidity, Door, Alarm, Presence) is manages real physical object’s status. |

#### Static Perspective (Generalization Style)

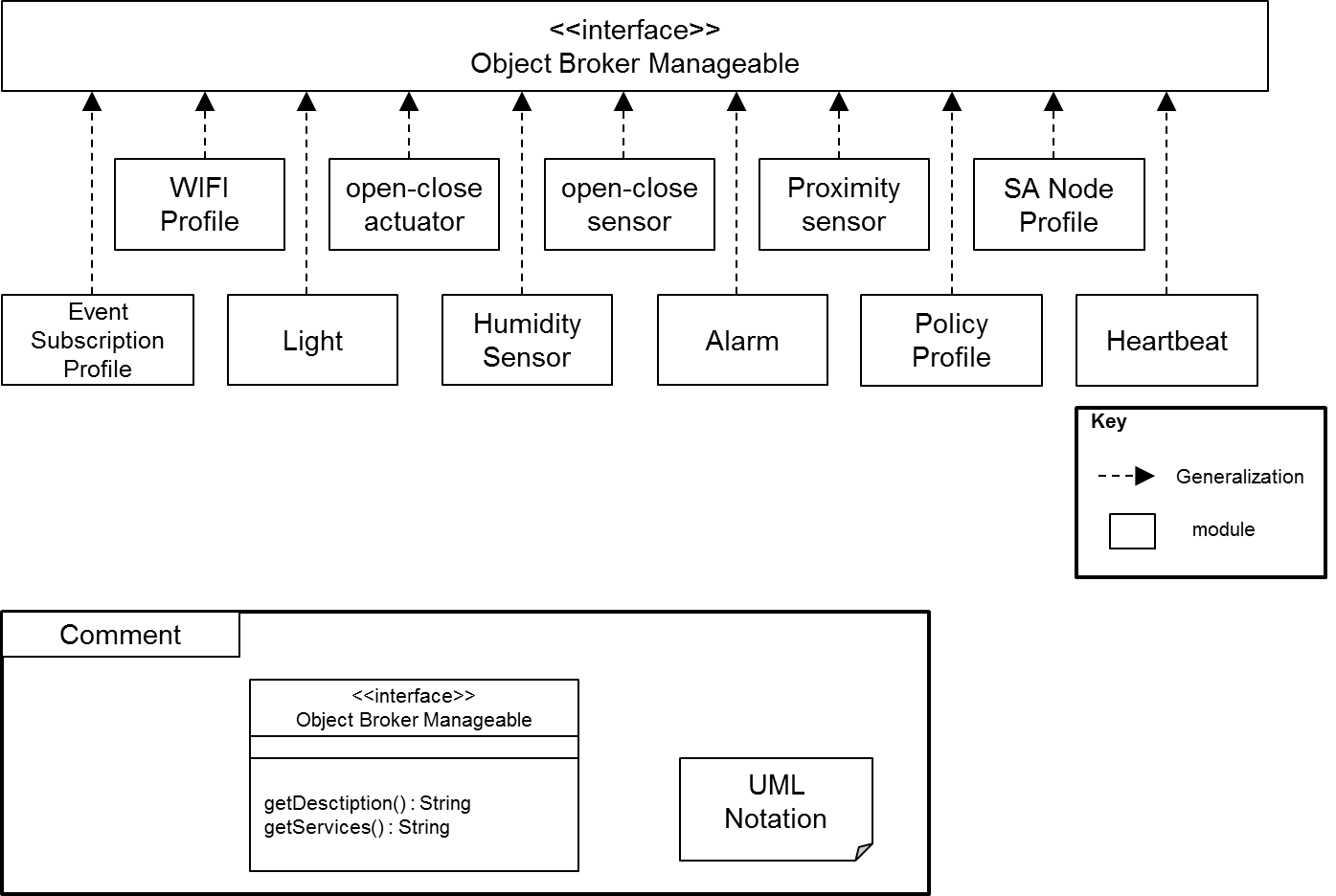


Figure 9 Static perspectives (Generalization Style) of SA Node

#### Static Perspective (Use and Decomposition style)

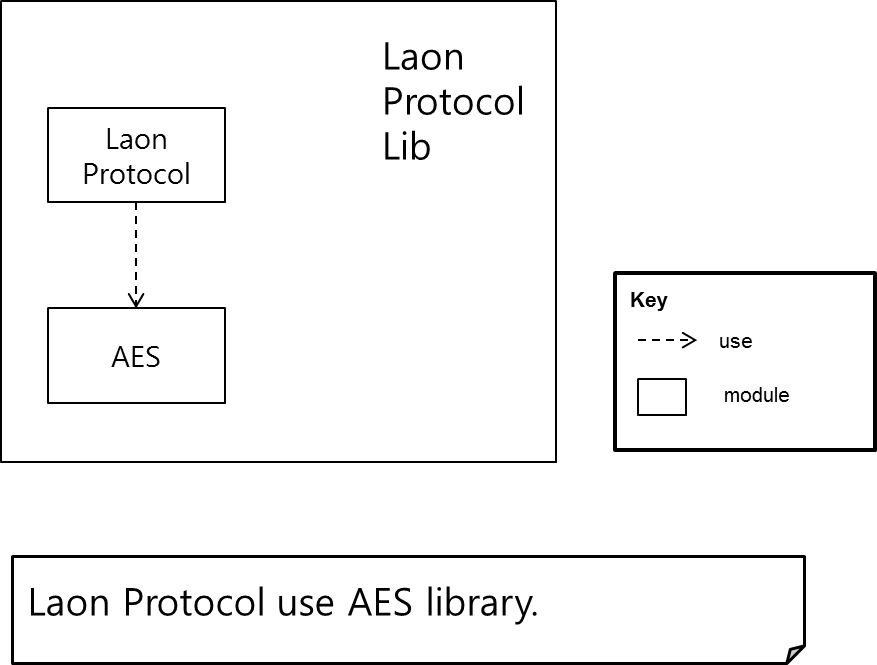


Figure 10 Static Perspective (Use and Decomposition style) of SA Node

### Laon IoT System Server

#### Dynamic Perspective – C&C view

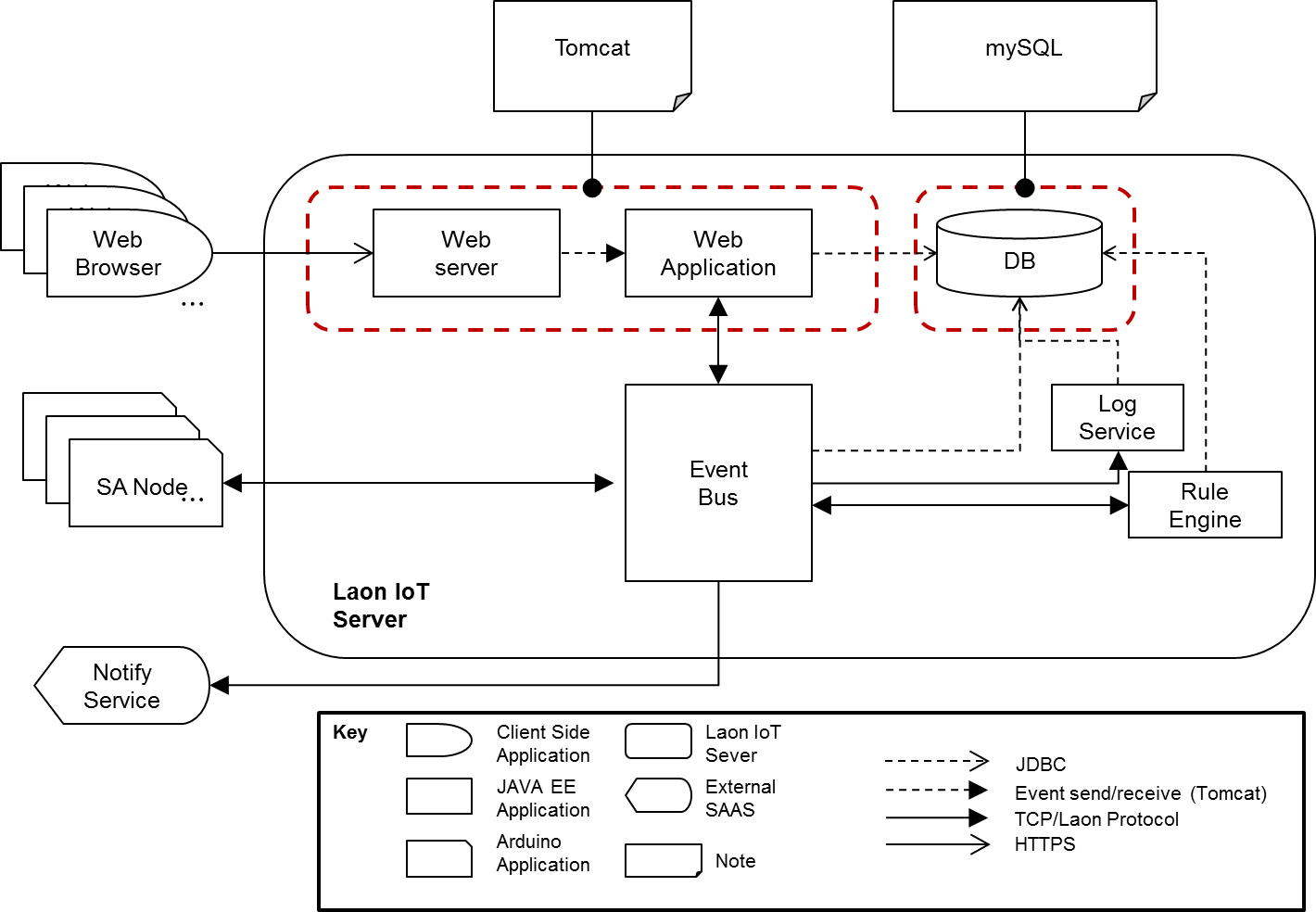


Figure 11 C&C View of Laon IoT Server

##### **Rationale**

Decouple between user service and node management. Proxy adapt quality attribute of SA Node.  
Event bus promotes scalability and extensibility. Limit Exposure tactic is applied

##### **Responsibility**

|  |  |
| --- | --- |
| Component Name | Responsibility |
| Web Application | * It provides authentication service. * It provides authorization service. (User can control only his/her nodes.) * It controls nodes through event bus for processing user request. * It provides to query logs from database. * It provide node managing service |
| Event Bus | * It sends events from publisher to subscribers. * It manages node connection referring connection permission in database. |
| Rule Engine | * It recognizes node disconnection based on heartbeat event and then it send predefined message to the SMS service. * Recognize malicious SA Node and request to cut off connection with the node to broker. * (E.g. some SA Node sends huge amount events.) |
| Log Service | * It stores every event as log in database. * It doesn’t provide retrieve and update service. (for security) |
| DB | * It stores each resources status of nodes. It is a proxy for web application server. * It store system information. (e.g. log, user, node, house and so on) |
| Notify Sender | * It provides to send a SMS Message. |

#### Static Perspective – Module view

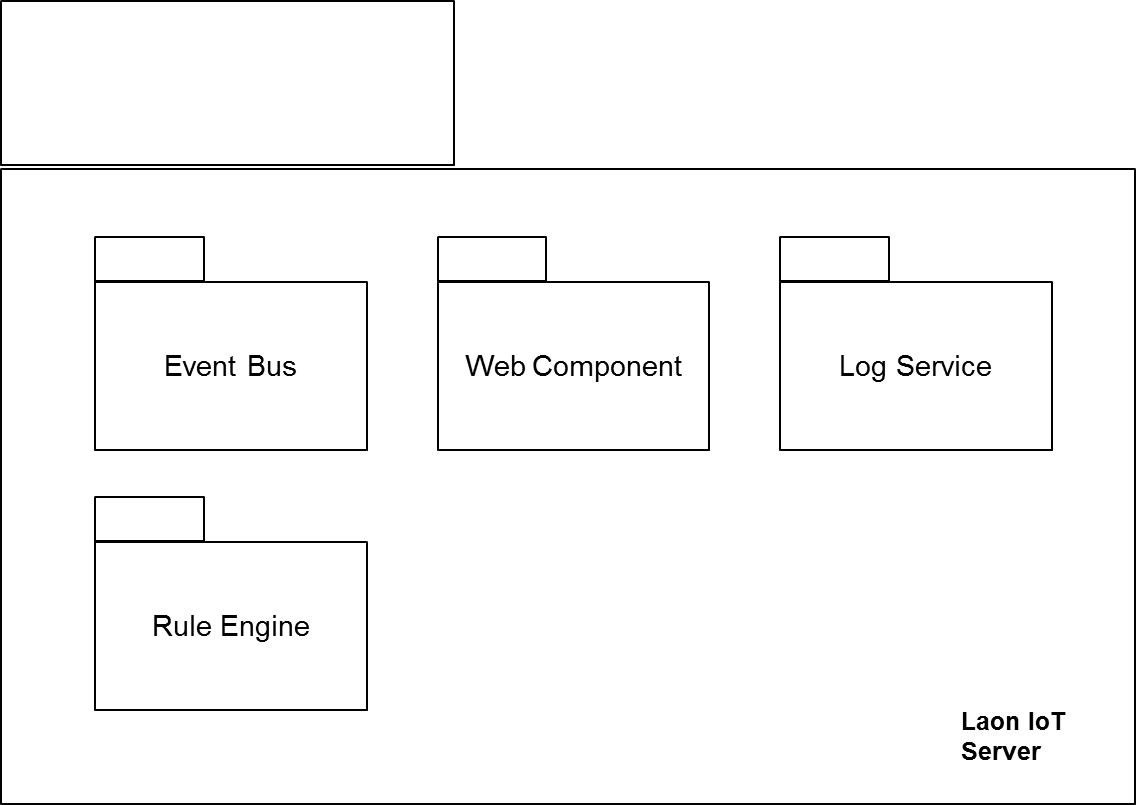
****

Figure 12 Module View of Laon IoT System Server

#### Physical Perspective – Allocation view

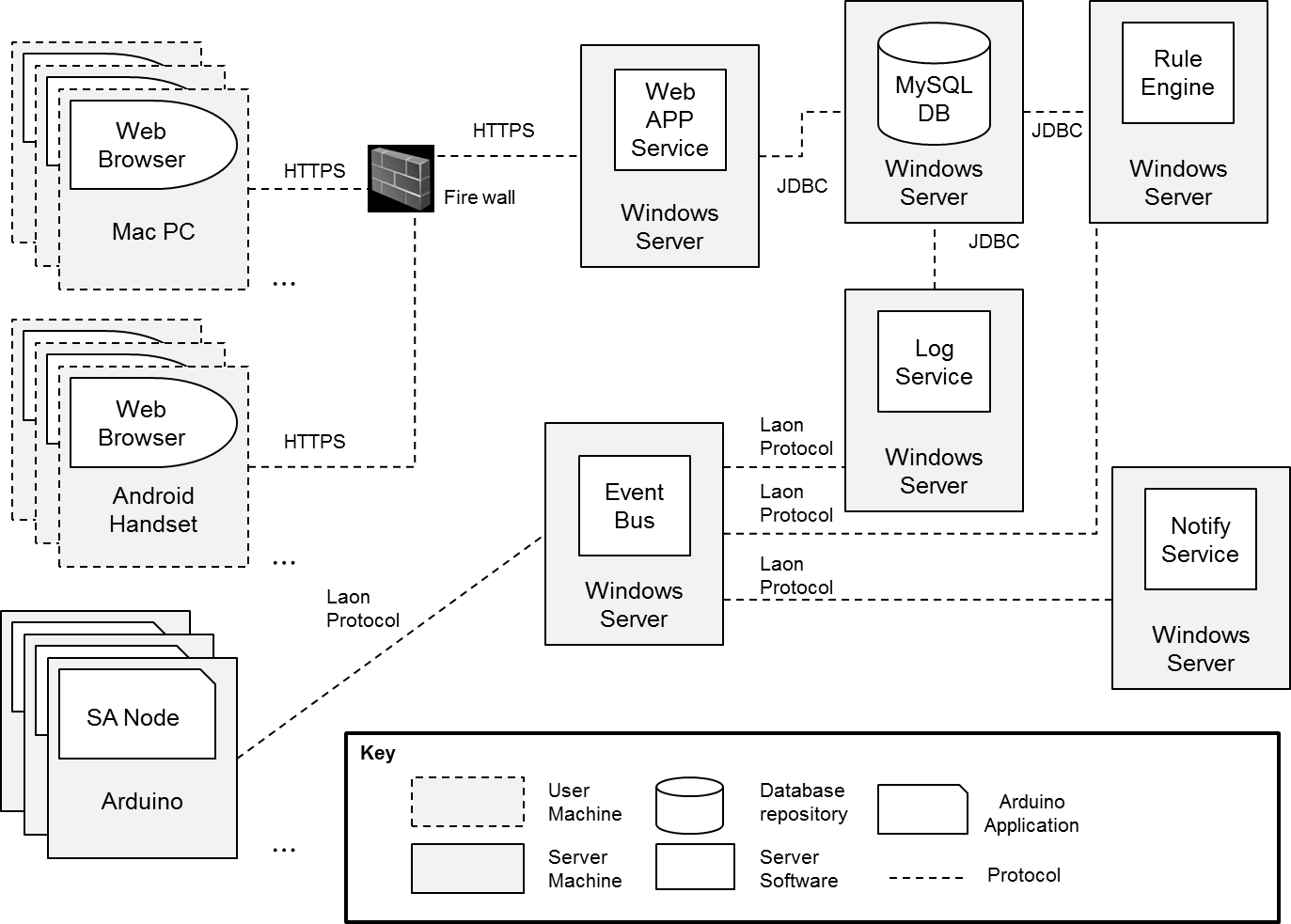


Figure 13 Allocation View of Laon IoT System Server

## Third Level Decomposition



### Event Bus

#### Dynamic Perspective – C&C view

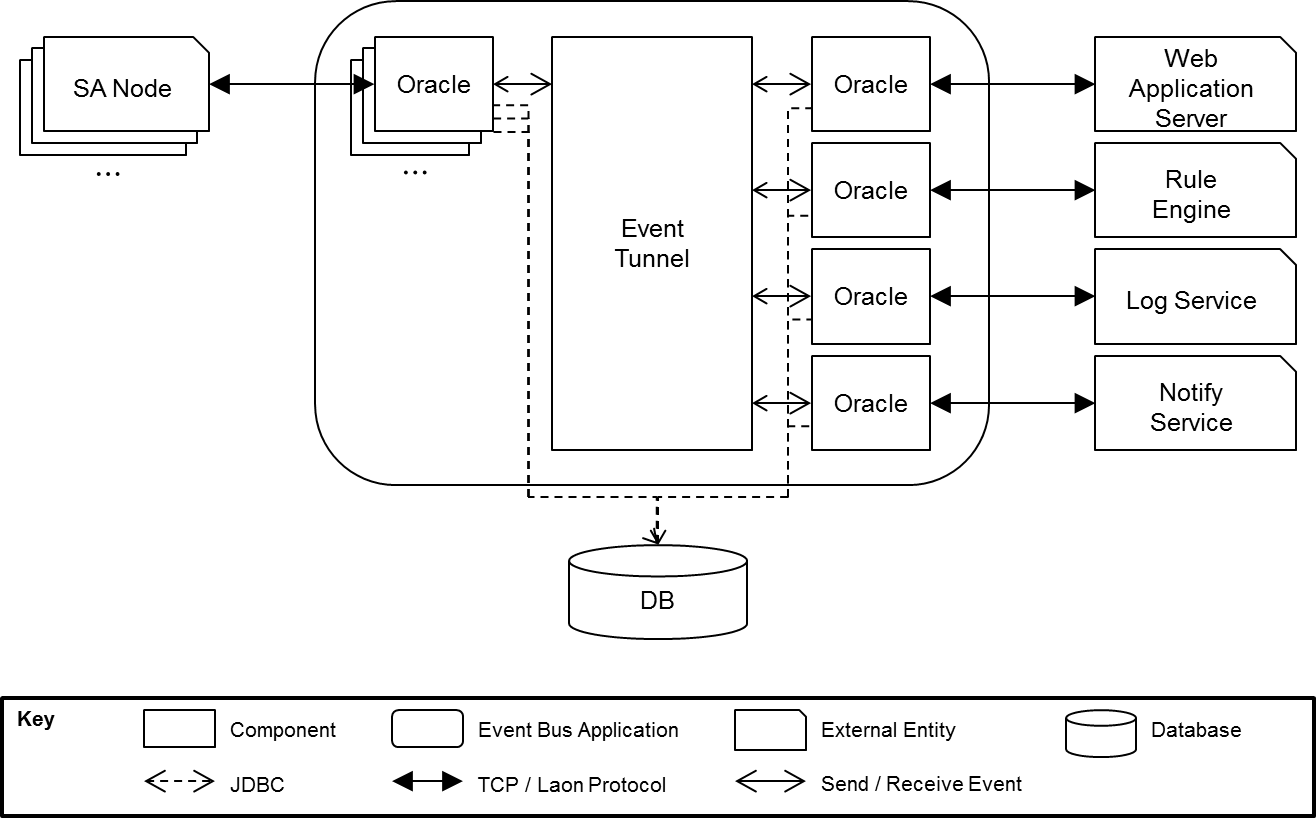


Figure 14 Event Bus C&C View

##### **Rational**

The event bus should be implemented to apply the Publish/Subscribe architecture style. All components in the event bus should run on each thread to ensure that events are deliverable during I/O operations. External entities should be connected to each adapter named Oracle to register, publish and subscribe events.

##### **Responsibility**

|  |  |
| --- | --- |
| Component Name | Responsibility |
| Event Tunnel | * Delivers events to each appropriate subscriber. * Guarantees all oracles are not blocked by other oracles. |
| Oracle | * Registers external entities to the event tunnel as a subscriber. * Bans the connection requested from unregistered external entities. * Blocks events sent from unregistered external entities. * Stores external entity’s schema to the database when nodes attempt to register to the event bus. * Conveys events sent from the external entity to the event tunnel. * Conveys events received from the event tunnel to the external entity. * Monitors and report the status if connected or disconnected   The term of external entity represents SA Nodes, Services and Web client. |

##### **Sequence Diagram (USE CASE: Registration request from external entities)**

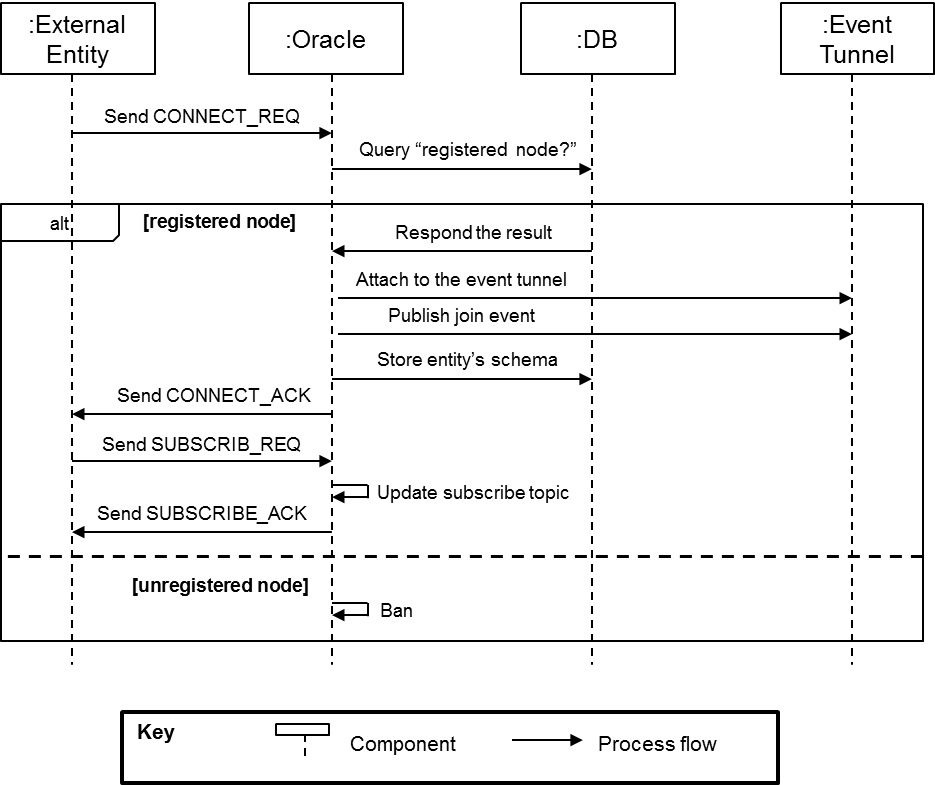


Figure 15 Sequence diagram of Registration request from external entities

##### **Sequence Diagram (USE CASE: Report join / exit status from external entity)**

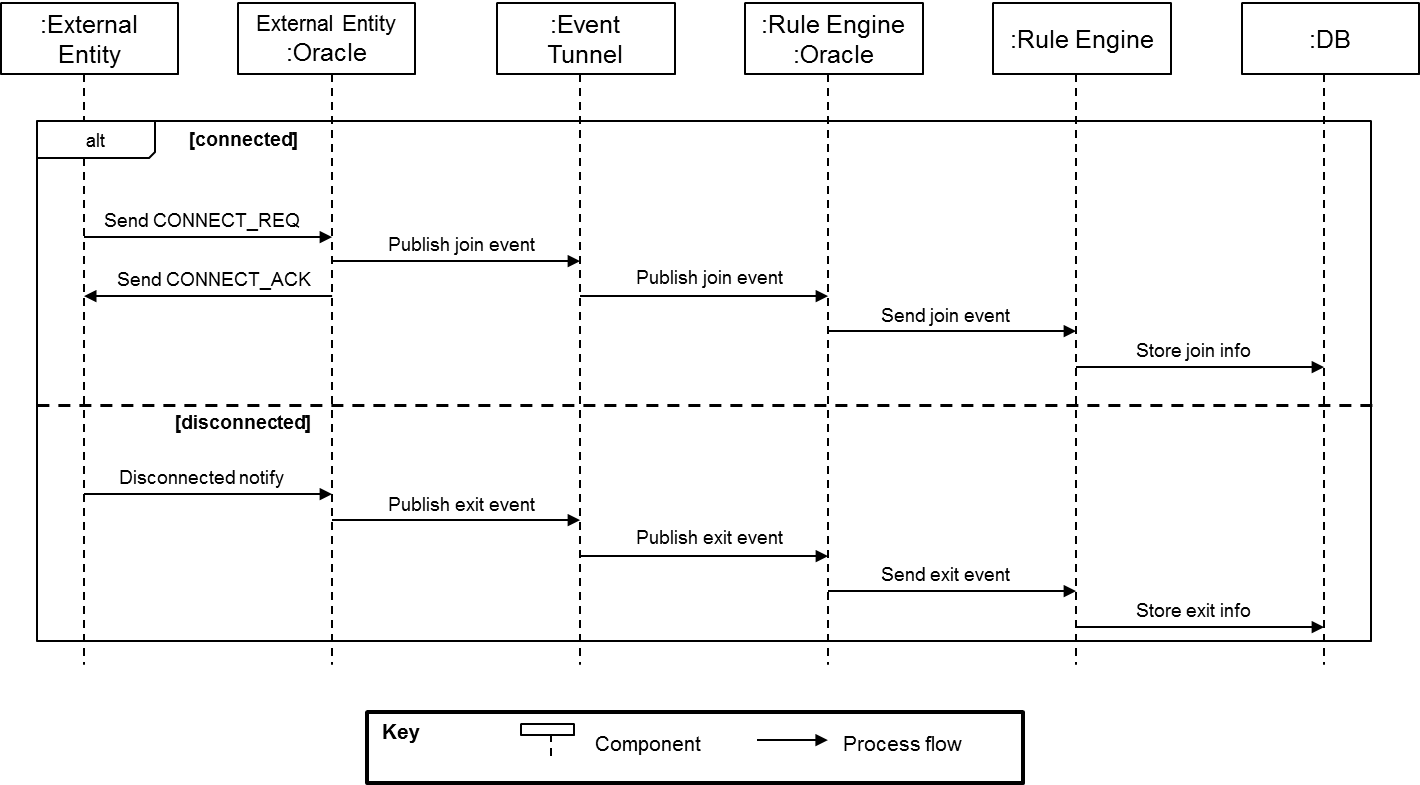


Figure 16 Sequence Diagram of "Report join / exit status from external entity"

##### **Sequence Diagram (USE CASE: Report SA Node sensor values and user commands)**

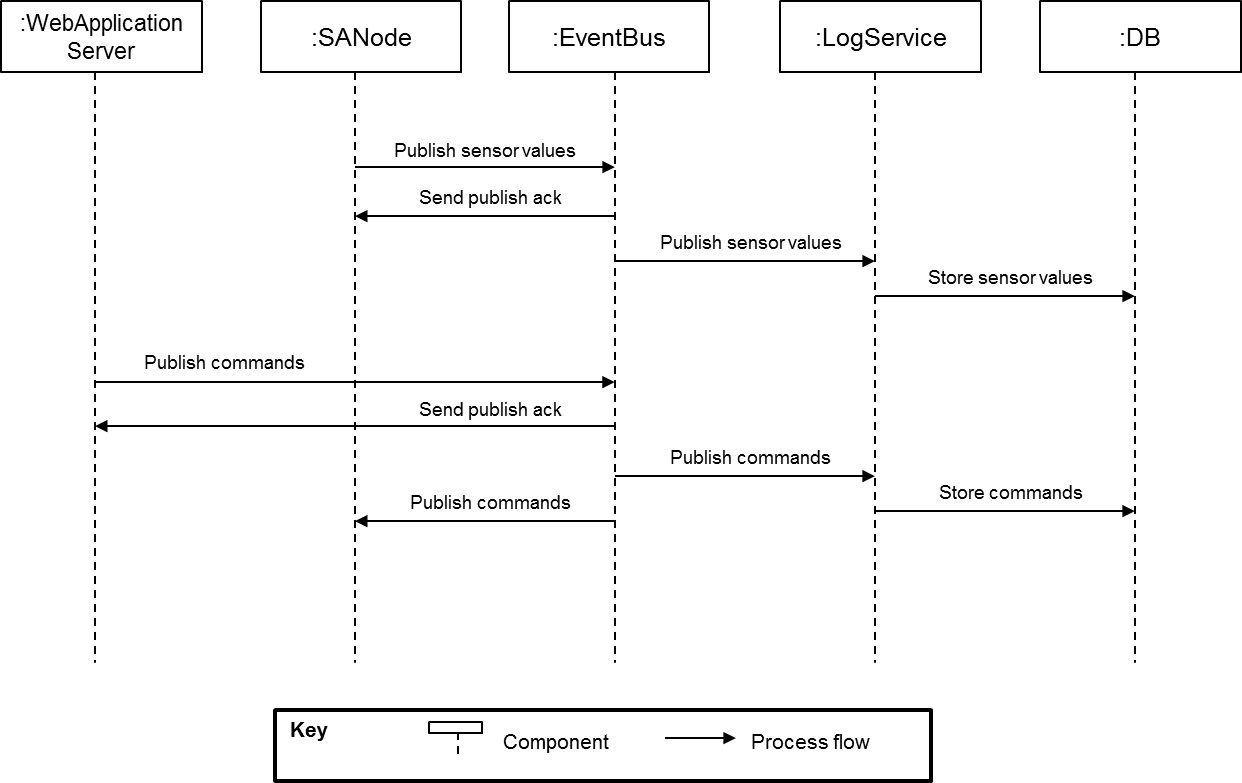


Figure 17 Sequence Diagram of reporting SA Node sensor values and user commands

##### **Sequence Diagram (USE CASE: Notify the situation to users (or security services) on emergency)**

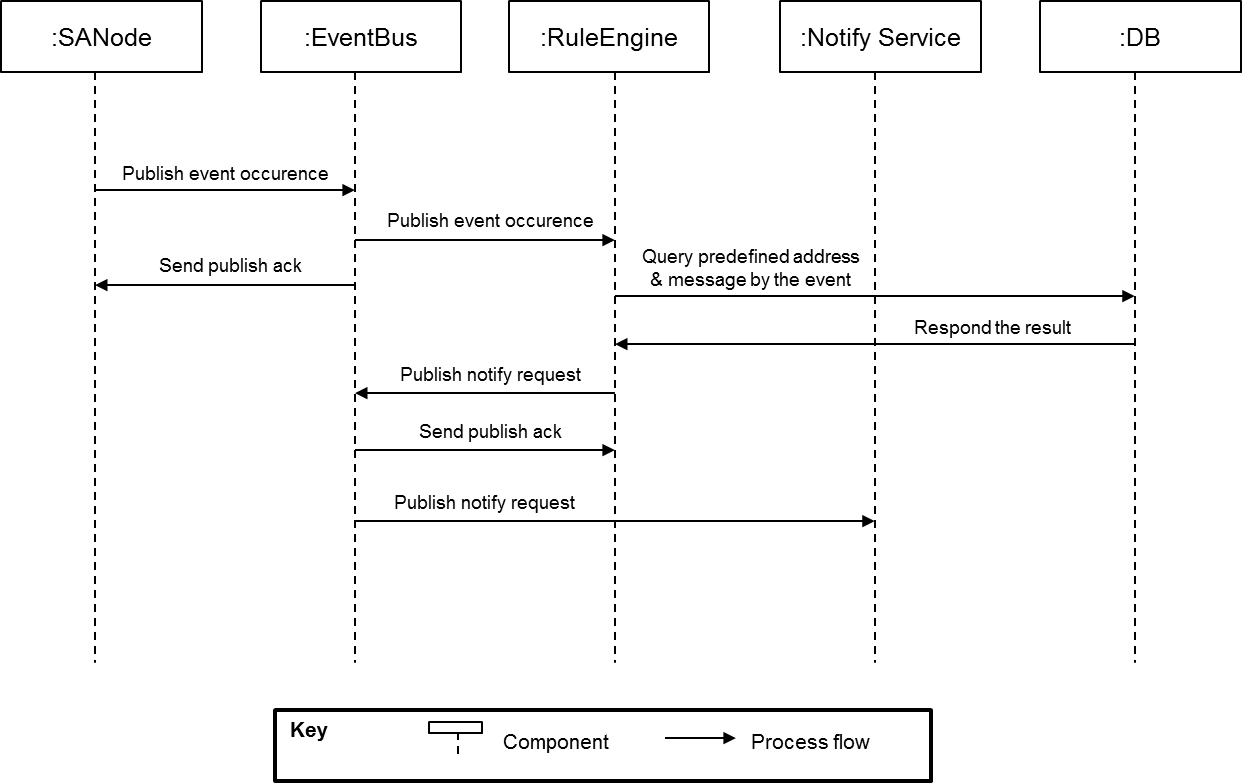


Figure 18 Sequence Diagram of notifying the situation to users

#### Static Perspective – Module view

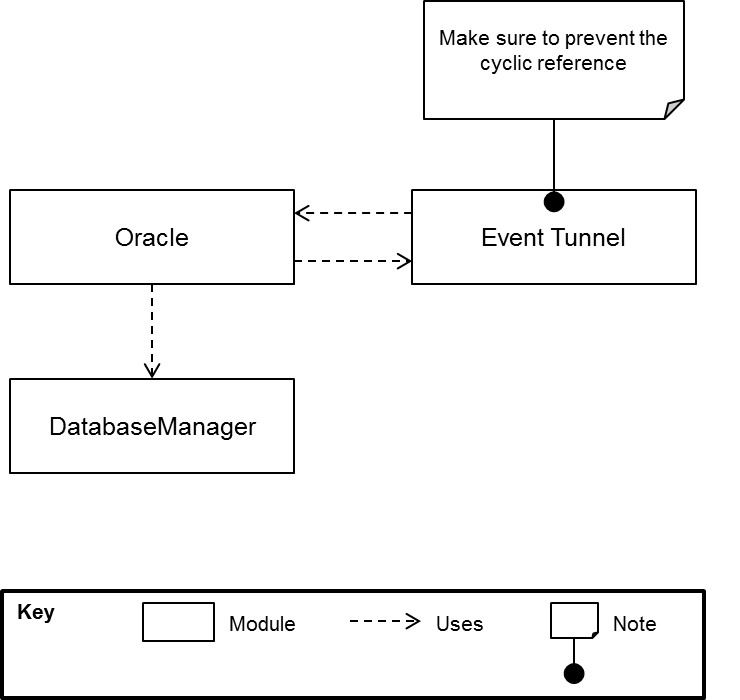


Figure 19 Module view of event bus

#### Physical Perspective (Allocation View) - Work assignment style

|  |  |  |
| --- | --- | --- |
| Module | Assignee | Note |
| Event Tunnel | Seunchan Kwon |  |
| Oracle | Seunchan Kwon |  |
| Data Base Manager | Yongjae Chang | Include JDBC connection |

### External entity

#### Dynamic Perspective – C&C view

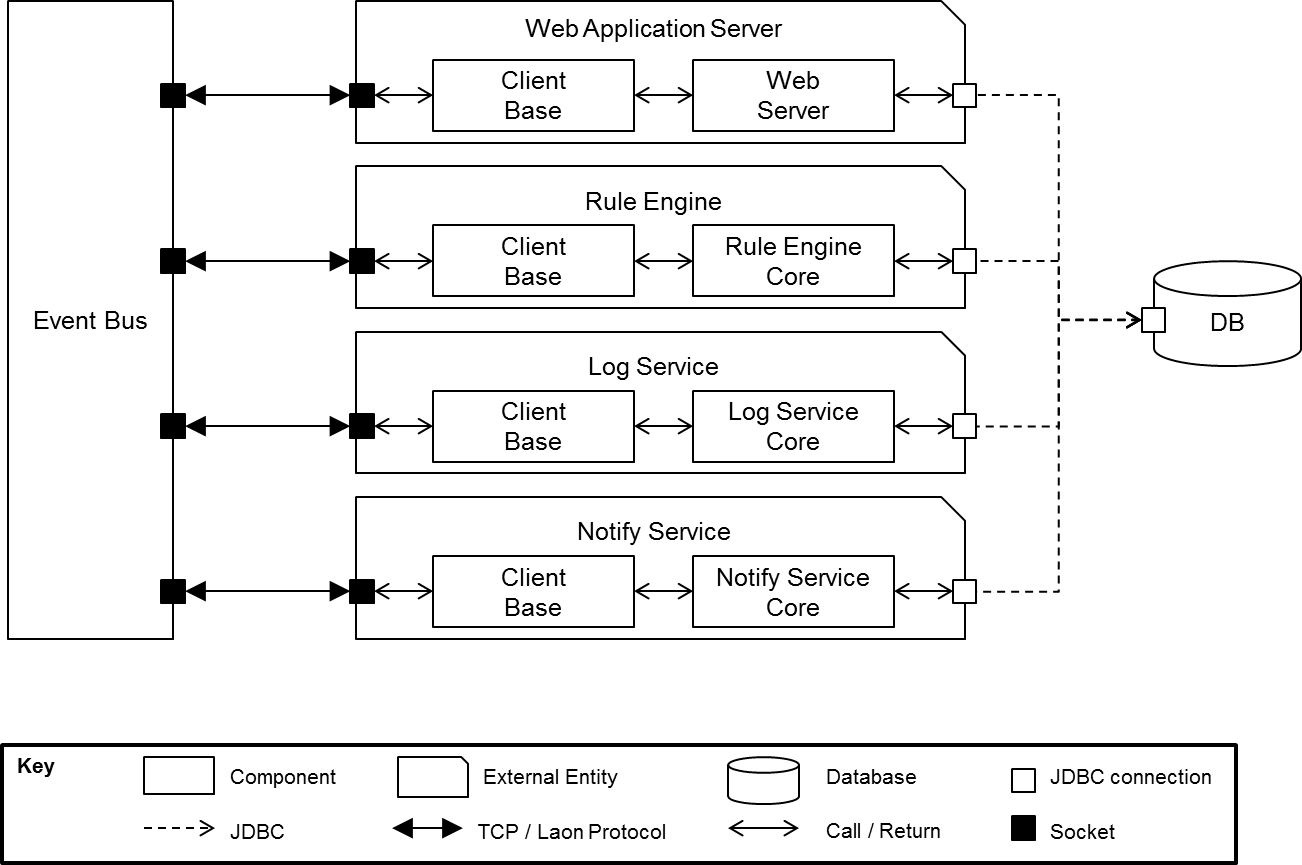


Figure 20 C&C View of External Entity

##### **Rationale**

The common protocol component should be built to share common functionalities about Laon Protocol among each external component. Then all external components can concentrate with their own core functionalities.

##### **Responsibility**

|  |  |
| --- | --- |
| Component Name | Responsibility |
| Client Base | * Provides common functionalities about Laon protocol to each external component. * Abstract the functions such as connection, registration, publish, subscribe request and heartbeat. |
| Web Application | * Provides web UI that includes functionalities about user sign-in, user registration and node management (control and monitor). |
| Rule Engine Core | * Make decision with received events on each event cases (connect, disconnect, emergency, mail received). * Send the notify request with address and text contents by the decision. * Store current connection status about nodes |
| Log Service Core | * Store control command logs and sensor value logs. |
| Notify Service Core | * Send messages when the notify event are subscribed. |

#### Static Perspective – Module view

##### **Use Style**

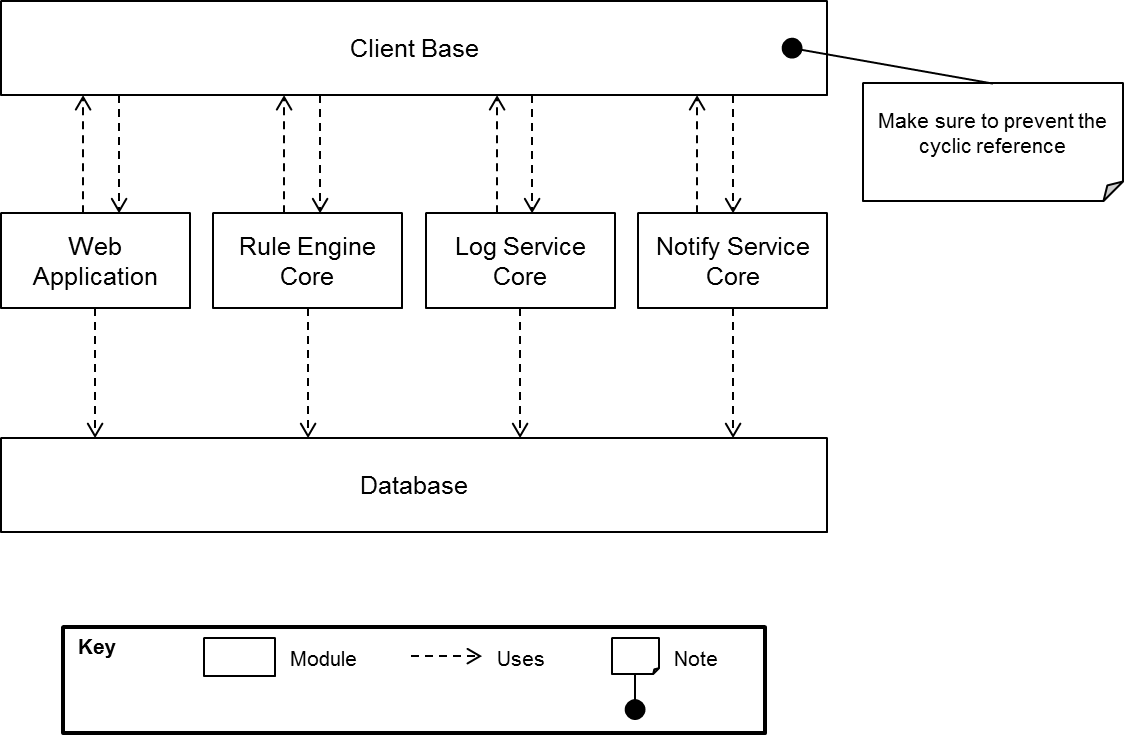


Figure 21 Module View of External Entity (Use Style)

##### **Class Diagram (Notation UML 2.0)**

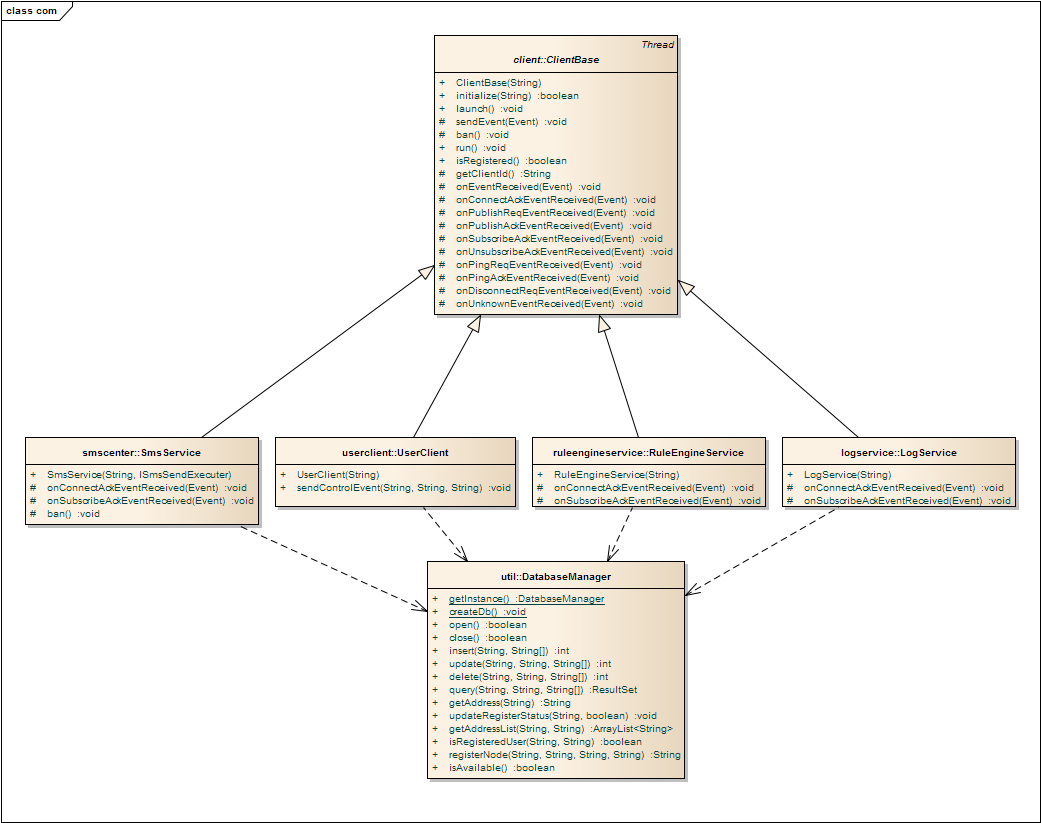


Figure 22 Module View of External Entity (Class Diagram)

**Rationale**

Definition of the relation among classes following component names correspond to modules in the module view. And they have same responsibility as in the module view.  
To hide specific & common logic, the Template Method pattern should be applied.

**Responsibility**

|  |  |
| --- | --- |
| Component Name | Responsibility |
| Client Base | * The class that corresponds to Client Base in the module view. * To apply the Template Method pattern, some method should be declared as abstract. But leave them as null method for the easy implementation. |
| UserClient | * The class that corresponds to a part of Web Application in the module view. Because of the border between implementation languages. * This should provide APIs to the external Web Server. |
| RuleEngineService | * The class that corresponds to Rule Engine Core in the module view. |
| Log Service | * The class that corresponds to Log Service Core in the module view. |
| Notify Service | * The class that corresponds to Notify Service Core in the module. |
| Database Manager | * The class that provides the database operations. |

## Fourth Level Decomposition



### Web Application

#### Dynamic Perspective – C&C View

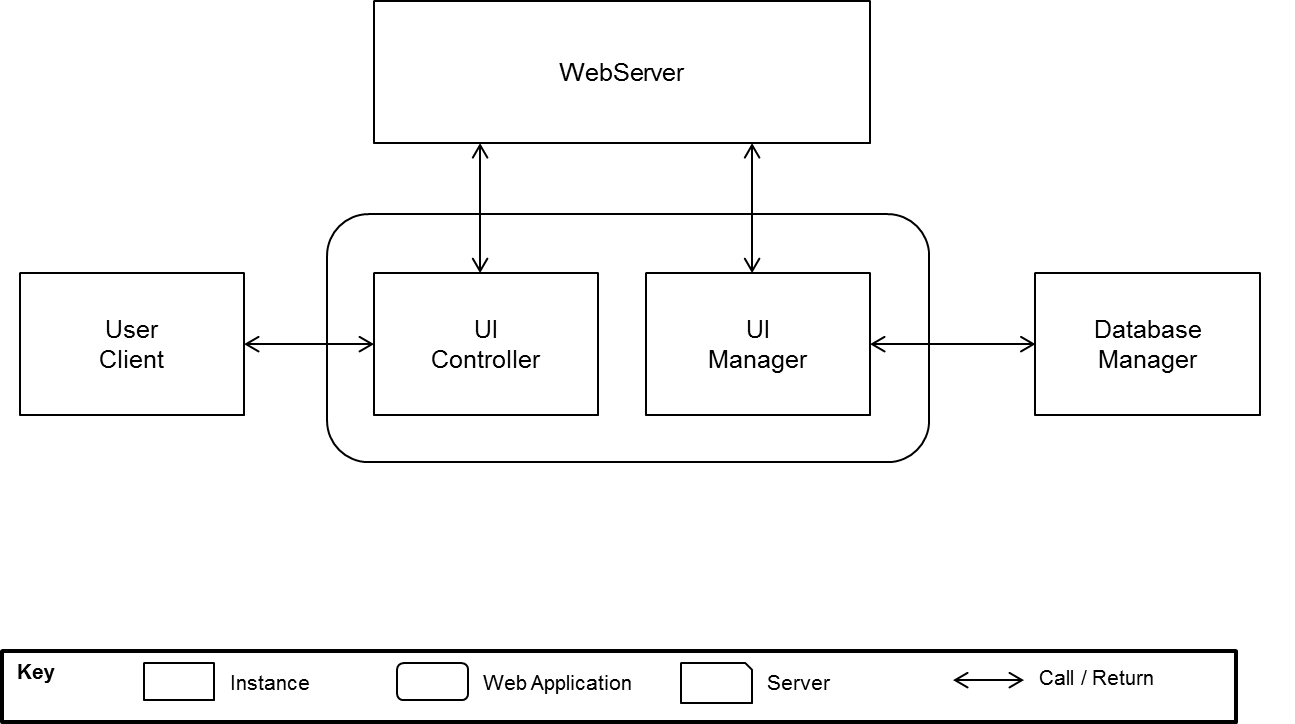


Figure 23 C&C View of Web Application

##### **Rationale**

Define the relation among classes. According to each component’s role and process flows, decompose web application to UI controller and UI manager.

##### **Responsibility**

|  |  |
| --- | --- |
| Component Name | Responsibility |
| UI Controller | * Calls the control function built from the user input |
| UI Manager | * Sends the display information built from the database to web server in HTML. |

### Heartbeat

#### Dynamic Perspective (C&C View)

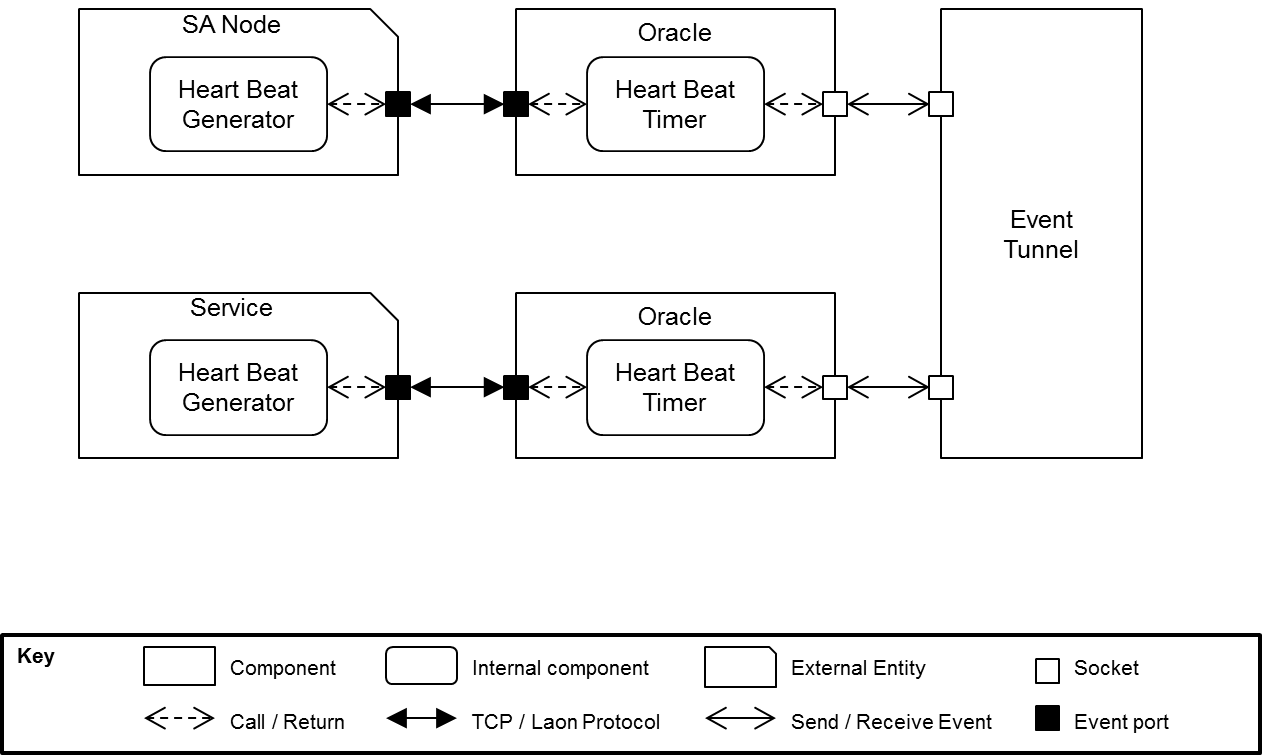


Figure 24 C&C View of Heartbeat

##### **Rationale**

The Heart Beat tactic should be applied to guarantee external entities are alive (still connected).

Less network load will be taken with the Heart Beat tactic than others such as Ping/Echo.

##### **Responsibility**

|  |  |
| --- | --- |
| Component Name | Responsibility |
| Heartbeat Generator | * Sends the heartbeat to a corresponding Oracle every predefined period. |
| Heartbeat Time | * Sends the exit event to the event tunnel when the external entity doesn’t send the heartbeat. |

#### Static Perspective - Class Diagram (notation UML 2.0)

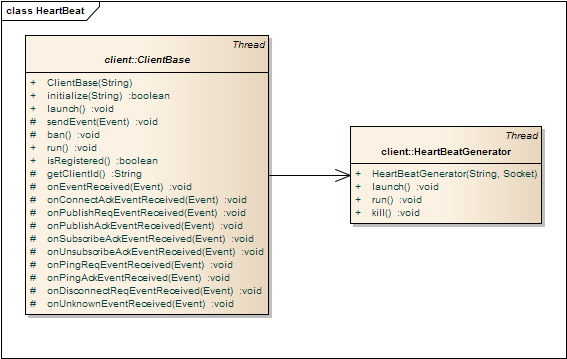


Figure 25 Class Diagram of Heartbeat

### Event Tunnel

#### Static Perspective – Class Diagram (notation UML 2.0)

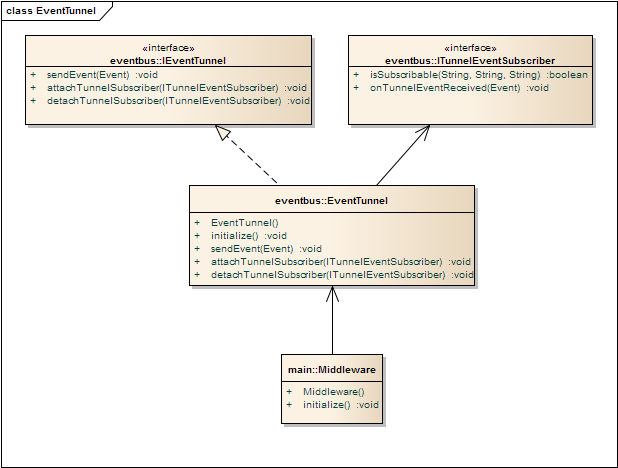


Figure 26 Class Diagram of Event Bus

##### **Rationale**

Define the relation among classes. To remove the cyclic reference, put the interface between the EventTunnel and the Subscribers. All interface state the accessible operations.

##### **Responsibility**

|  |  |
| --- | --- |
| Component Name | Responsibility |
| IEventTunnel | * This states the interface for the subscribers which want to use the Event tunnel. * Following operations should be stated.   public void sendEvent(Event event);  public void attachTunnelSubscriber(ITunnelEventSubscriber subscriber);  public void detachTunnelSubscriber(ITunnelEventSubscriber subscriber); |
| ITunnelEventSubscriber | * This states the interface for the subscribers which want to receive the event from the Event tunnel. * Following operations should be stated.   public boolean isSubscribable(String topicAddress, String topicName, String topicCommand);  public void onTunnelEventReceived(Event event); |
| EventTunnel | * Delivers events to each appropriate subscriber. * Guarantees all oracles are not blocked by other oracles. |
| Middleware | * Main thread for the EventTunnel and Oracles. * This should listen to the socket acceptance. |

### Oracle

#### Static Perspective – Class Diagram (notation UML 2.0)

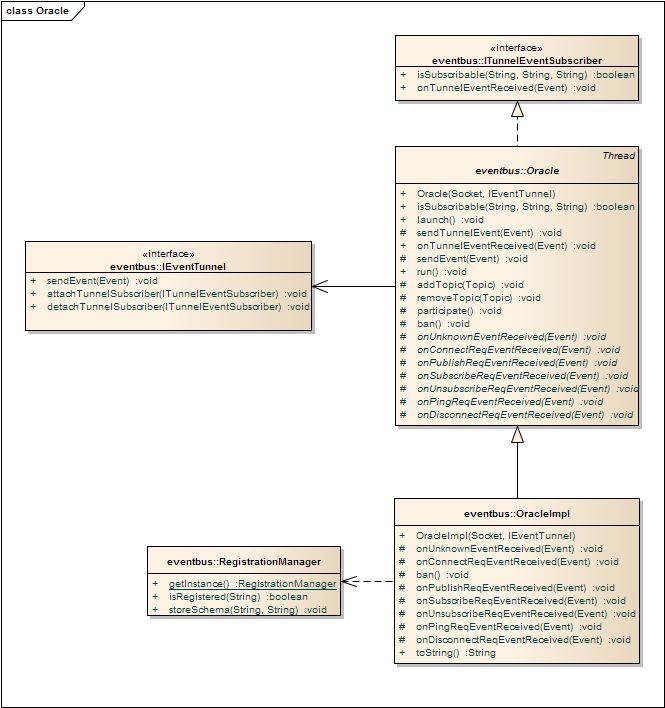


Figure 27 Class Diagram of Oracle

##### **Rationale**

The RegistrationManager should be provided to authorize nodes.   
Oracle should implement the ITunnelEventScriber interface because the Oracle is one of TunnelEventScribers. To hide specific & common logic, the Template Method pattern should be applied. Then, we can make the new system similar to this using this common (parent) classes

##### **Responsibility**

|  |  |
| --- | --- |
| Component Name | Responsibility |
| Oracle | * Registers external entities to the event tunnel as a subscriber. * Conveys events sent from the external entity to the event tunnel. * Conveys events received from the event tunnel to the external entity. |
| OracleImpl | * Bans the connection requested from unregistered external entities using RegistrationManager. * Blocks events sent from unregistered external entities. * Stores external entity’s schema to the database when nodes attempt to register to the event bus. * Monitors and report the status if connected or disconnected |
| RegistrationManager | * Provides the way to authorize nodes with DatabaseManager. |

#### Physical Perspective – Allocation View (work assignment style)

|  |  |  |
| --- | --- | --- |
| Module | Assignee | Note |
| EventBus | Seungchan Kwon |  |
| Log Service | Seungchan Kwon |  |
| Rule Engine | Seungchan Kwon |  |
| Notify Service | Seungchan Kwon | On the android phone |
| Web Application | Yumi Kim |  |
| DatabaseManager | Yongjae Jang | Including JDBC connection |
| Event Bus Protocol | Yongbong Choe |  |
| SA Node | Joohyung Woo |  |

1. Future works

**Enhance Security function of Laon IoT system**Almost IoT devices have low computing power and low memory so it’s very hard to build strong security function at the device. The SA node we use for IoT system also is in same shoes. Even though we implement protocol level security with AES encryption, we think that someday it will be vulnerable because it is a little bit weak secure logic for intended and focused attack. So we will think again how to build strong security system immutable to malicious attack at IoT device and Laon IoT middleware.

**Improve Performance of Laon IoT event bus**Laon IoT event bus is realized with publish-subscribe pattern. That mean that server running publish-subscribe function can be a slow-down when many IoT devices over 1000,000 are connected and there are heavy traffic. To mitigate the load of publish-subscribe server, we need to add network load balancer in front of event bus. Tightly coupled Oracle with event bus has a role authenticating IoT devices. The Oracle always checks whether incoming request is from valid IoT device with hardware ID and password. But to reduce the load, we need to add a function validating MAC address of IoT device before parsing and validation incoming request.

**Emerging network technology**we designed the architecture of Laon IoT which can embrace easily emerging protocol. But we don’t experiment with real device which has new protocol. Someday we will evaluate Laon IoT system architecture when the device or protocol is available.

**APIs for 3rd Party Service provider or application**Laon IoT system provides web service which can make 3rd party service or application developer to build custom apps, services, and/or make mashups from existing available service. But we have no API document available for 3rd party developer. we wil release the service APIs document and SDK to setup IoT ecosystem.