

# Integration of Ericsson OSS BSS solution for Multi-Country Telecom Operator

Eashin Matubber



<b>Author(s)</b> Eashin Matubber	
<b>Degree programme</b> Business Information Technology (BITE)	
<b>Report/thesis title</b> Integration of Ericsson OSS BSS solution for Multi-Country Telecom Operator	<b>Number of pages and appendix pages</b> <b>77 + 1</b>
<p>Integration Engineering for a multi-country Telecom operator to implement Telecommunication support software to adapt to technology and competitiveness changes is a demonstrable skill set. Daily, problem-Solving and planning to meet customer requirements with the right solution help develop professionalism by working in a team of the most skilled experts in the industry from many different countries.</p> <p>This writing proceeded along with developing ICT expertise in a short time, especially while working for a tech giant or global corporation. Integration engineer's team working scope is with Software Development tools, technologies, professional communication with a global team, and fast learning competencies.</p> <p>This has been written for 12 weeks, including many areas of software development, project management, Agile software development methodology Scrum, software development, and customization of software according to customer needs. From 09 November 2020 to 11 January 2021 of 10 weeks of writing shows professional performance and development through weekly analysis and continuous learning from individual and team perspectives.</p> <p>During the work of this writing, it was also individual agility through switching from one field to another of the project and collaboration with team as giving huge room to learn and improve. Each day, performance level enhancing was another significant part while building analytical strength and research ability to find the most appropriate and flawless solution. Professional progress and skill development details of this writing may add value to the objectives of any early-career starter and will help to create a more productive contribution to an ICT project.</p>	
<b>Keywords</b> Java Email API, Robot Framework, Ericsson Adaptive Inventory, Plain Old Java Object, Next Generation Object, Simple Mail Transfer Protocol	

## Table of contents

1	Introduction .....	1
2	Framework .....	3
2.1	Analysis of your current work .....	3
2.1.1	Description .....	3
2.1.2	Evaluation and Development .....	5
2.2	Interest groups at work.....	7
2.3	Interaction skills at work .....	7
3	Diary entries .....	8
3.1	Observation week 01 .....	8
3.2	Observation week 02 .....	15
3.3	Observation week 03 .....	22
3.4	Observation week 04 .....	30
3.5	Observation week 05 .....	36
3.6	Observation week 06 .....	43
3.7	Observation week 07 .....	50
3.8	Observation week 08 .....	56
3.9	Observation week 09 .....	62
3.10	Observation week 10 .....	67
4	Discussion and conclusions .....	73
	References .....	77
	Appendices .....	80
	Appendix 1. Abbreviations.....	80

# 1 Introduction

This writing is an analysis of integration tasks to implement Operation Support Systems (OSS) or Business Support Systems (BSS) provided by one of the leading providers of Information and Communication Technology (ICT) Telefonaktiebolaget LM Ericsson (publ) known as Ericsson.

These diary-like journals and analyses are based on job scope as an Integration Engineer of the Digital Business Services Finland & Baltics team of Ericsson Finland and collected data from real-life work sources during the project progress. The writing timeframe starts from week 44 of Autumn 2020 and continues for the next twelve weeks. Each week's outcome is an analysis consisting of various data from daily journal entries of ten observation weeks and another two weeks to discuss relevant issues, including the conclusion.

Operations Support Systems (OSS) and Business Support Systems (BSS) provide the infrastructure to manage the network, process the service orders, and bill the customer. This ongoing project is to integrate and implement the Ericsson Adaptive Inventory (EAI), a Network Domain Management solution to Automation & Orchestration of network operations. Ericsson Adaptive Inventory (One of Ericsson's OSS software) enables visibility of network, service, and application resources for service rollouts and network efficiencies. The integration team's main activity is to integrate the Ericsson Adaptive Inventory for a renowned multi-country telecom operator in the Nordic region.

As a member of the Integration Team, my job scope covered various software engineering activities, i.e., on-demand software development as part of the integration and testing of a developed feature. The work's nature and complexity need continuous consultation with the customer (referred to as The Telecom Operator, which names not necessary or recommended not to use).

The project is being managed using the Scrum Framework, followed by the implementation of Agile software development methodology. Scrum recognizes no sub-teams in the Development Team, regardless of domains that need to be addressed like testing, architecture, operations, or business analysis. (Schwaber & Sutherland, 2017). which is genuinely reflecting in the team. Our Development Team members find it in work activity in different fields as a developer. For example, I perform and contribute in both testing and development areas. To utilize Agile programming conveyance, we have to add iterative to

the steady ideas (Cohn, 2010). A three-week sprint allows our development team to accomplish development tasks, test, integrate & release features more efficiently without defects, empower self-organization of the team & ensure flexibility to members.

The current OSS project with a multi-country telecom operator of the Nordic region has a long-term aim to implement the same for different countries of the same operator. To meet the fulfillment and successfully deliver the project for all countries, it uses the Atlassian project management tools. The whole team follows Agile & scrum using Jira, project documentation & collaboration using Confluence, Bitbucket for version control, and Bamboo for CI & CD. The entire package of Atlassian tools makes the project perfect agile, and from a management perspective, Atlassian tools serving the big aim of the multi-country project goal effectively.

Highly professional, transparent, and on-time communication skill is mandatory, while team members work from many different geographic locations like India, United States, Italy, Sweden & Finland. From a technical perspective, my work required a sound understanding of software development analysis, software testing analysis, object-oriented programming language Java, JavaScript, & Spring Boot framework, test automation framework (RF), distributed version control system (Git), RDBMS, SQL, basic knowledge of DevOps, and other software development tools & technologies. Besides, a basic understanding of telecommunication networks & resources is also required as the whole project is about the automation of network operation and orchestration using the Ericsson Adaptive Inventory.

A significant transition or change to mention that in the middle of the project, Ericsson introduced the working from home (WFH) program globally during the Covid-19 pandemic, and surprisingly WFH program brought the whole team to a new level of cooperation and productivity. Remote working and collaborating tools like MS teams and Ericsson's internal digital working tools make teamwork more manageable and effective. According to many colleagues, it is even better collaboration than working from the office.

Specialists & experts in the fields are in the team & the learning together approach is very common in every Ericsson team. It is a powerful motivating fact that teammates and management highly appreciate continuous learning and individual agility as the new member. From the beginning, a new team member is guided by management to set personal goals & organization goals, which have been achieving along with the project proceeds.

## 2 Framework

### 2.1 Analysis of your current work

#### 2.1.1 Description

It has already been eight months since I started working as an Integration Engineer (Trainee) of Ericsson Finland. As an Integration team member of Ericsson's OSS products, I perform various types of roles such as analysis, plan, development, and test. Moreover, in those mentioned areas, individual tasks are the following:

- Analysis of customer (Telecom Operator) network & inventory data dump to model & develop the required feature.
- Create data files to model & develop templates of telecom network equipment.
- Analyze user stories/features in development.
- Test planning and create test cases using Jira.
- Executing tests, Document test execution results, and report defects in Jira.
- Track all defects and retest fixed defects, and report as resolved issues.
- Test automation planning & develop automation scripts with Robot Framework.
- Development activity of Resource Synchronization features with legacy systems to EAI.
- Troubleshooting & debugging tasks.

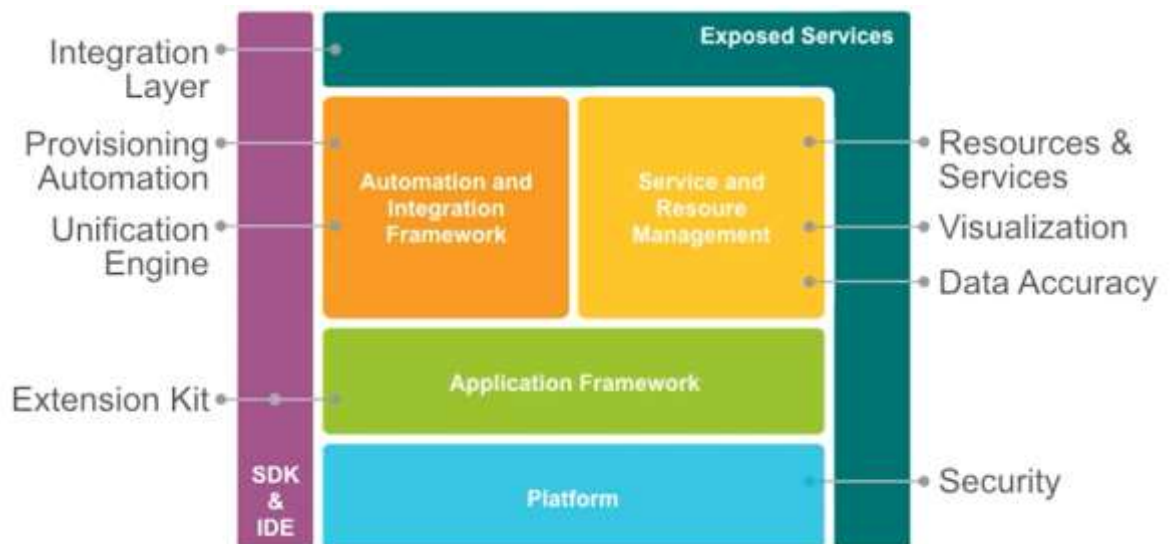


Figure 1. Ericsson Adaptive Inventory Framework

To model & develop templates of telecom network equipment required data analysis from the legacy system and extract the necessary data and ensure equipment creation accuracy using templates. In addition to dumped data analysis, it is another vital task to analyze vendor documentation of equipment as legacy system data often not reliable and not accurate sometimes. A data file is being created from the mentioned activity and deploy to the application layer and is used by the developed templates plugin to generate an equipment object and its child objects, which various EAI components will use. In figure 1, left side integration, provisioning, unification are areas of most work using SDK and extension kits.

Analyzing a user story in the development or analysis phase to understand the feature is a common initial step of testing activity. A good analysis often needs to have details, a deeper understanding, and a required analysis session with the respective developer. However, after completing the feature analysis, a test plan starts with creating test cases based on planned scenarios.

Required test data creation and execution tests and preparation test results documentation is an ongoing manual testing part. It is also important to outline the plan, which are the test cases or scenarios going to automated tests. After approval of selected test cases to automate from Test Manager, test automation scripts development, and commit to RF test automation repositories is a final phase of the task cycle. Test Automation, test tools, test management skills are needed, and it is an opportunity to have experience working with the latest tools and trends in this field.

Perform debugging of resource synchronization bugs raised by the client test team and involved in development activities in the same area, challenging and continuous learning tasks. To perform and contribute to the team, a member of this project team member should have a telecommunication network, equipment, Linux, JBoss, Oracle database in addition to a full-stack software engineering skill. I found myself lacking a skillset. I started learning many new tools and technologies like telecommunication network knowledge, L2/L3 VPN, JBOSS server, Soap API, Oracle DB, etc., with the project team as the required skill set while working.

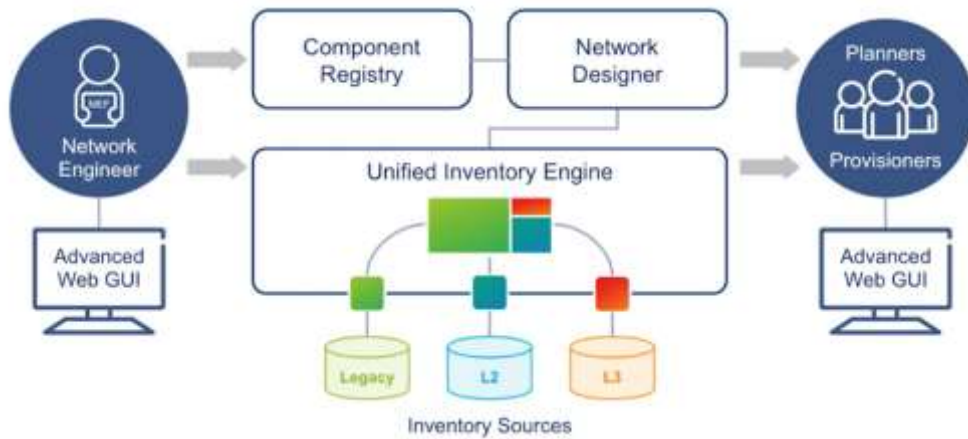


Figure 2. EAI Application Architecture

EAI architecture, application development libraries, and design documentation are topics all project members must be familiar with. I learned how to navigate Ericsson Library Explorer (Elex) and project confluence page thoroughly and find information quickly as it is always required during work. In figure 2, EAI architecture shows different network inventory sources and components providing services to the user level through a unified inventory system.

### 2.1.2 Evaluation and Development

Integration task of telecom OSS domain a new area for me and got tremendous training opportunity as a beginner and hands-on with all team experts. I considered myself the beginner stage performer in the project, so I started with simple tasks, and often I still required support from my teammates. Although we have many documentation and internal libraries on the application and its development areas for integration, most tasks seem difficult without my team's help.

I found many tasks required a lot of research and help from others to find solutions or develop, and some tasks were straightforward to complete as I have skills and previous experiences in that areas. I solved many minor issues and gained competencies to proceed with individual user story development with less help from others. I picked the working style, followed team meetings and customer meetings, and independently individual tasks quickly.

After a few months of work, I start participating in some decision-making meetings, joining an email chain for ongoing development instruction from solution architects, developer leads, or other stakeholders, and clearly understood or express opinions. I gained skills



through working, and at some point, I was able to perform systematically at a satisfactory level as previously spent on research to learn and apply it in my tasks.

During my employment period, my main objective was professional development which was achieved by gaining skills. My senior colleagues and another assigned mentor guide me as the team's trainee, and I used to get regular feedback to think further and improve. I learned the professional way of working, communication, and solution delivery to customers. I feel very privileged to start my career in such an international team of a corporate multinational company like Ericsson. This working opportunity made me a confident professional in the ICT field and built that mindset to learn anything required for work on the go.

I can analyze any user story to develop it for integration in the application and explain it to the customer and other relevant stakeholders. My contribution in the testing area was recognized from accepting scripts for automation to final test suites. Also, I showed my interest in learning any new areas of development, and I was allowed to start developing with senior developers. I have started with a new feature development of the application first with the help of other developers. Then I also optimized that and extended the functionalities of the feature. I learned to navigate product documentations from Ericsson's internal libraries and also know how to find a solution from external sources. At some point got feedback from senior team members that I am very much able to contribute to the project independently. I have also been participating in debugging and fixing defects of my code to provide a quality, flawless feature.

Integration engineering or development has an extensive work scope where I should have working space and technology flexibility. So, I must keep focused on learning any new skills whenever required. EAI (The product of Ericsson we are integrating for customers) used Java as a programming language. Still, it has a different development kit or SDK which supports other languages like JS and Groovy, and I think I should also be focused on those to have solid skills. We have started Camunda integration with the application for a specific country, and many things remain to learn in the future so that I will be more able to contribute to the current project and future projects. However, it is continuous learning with the team of agile project management and better delivery of the work to the customers.

## 2.2 Interest groups at work

I work with QA and Dev Team daily, and occasionally, I also work with the multi-country customer team for development and other decision-making works. Our developer's team has sub-teams for data migration development and integration developments, and I am a member of the integration development team. I am also working with the QA team for test automation, manual testing, and debugging-related tasks. I have many work events for deployment, access, different environment integration server, and issues with the infrastructure team from remote and onsite.



Figure 3. Interest Groups at Work

As shown in figure 3, the Ericsson EAI project team has a different group of teams or sub-team to work with. Ericsson's EAI project team members are located in a different location to work as a global team but belonged to a core team. We have an internal team of Developers, QA, Infrastructure, and another external team of multi-country telecom operators.

## 2.3 Interaction skills at work

Since the Covid-19 pandemic, Ericsson launched WFH (Work From Home) program for its employees, and we have all tools required for that. We communicate with many different tools like outlook, MS teams, Skype, Ericsson Digital workspaces, slack for external bodies.

As a multicultural team, we use English as our official language, and it was pretty good that most of my teammates are very easy to communicate efficiently. So we have a daily scrum where members from many different countries participate, and we communicate

with all members randomly according to time zone. There are windows for different purposes, and we all follow information flow about test, deployment, server issues, and document sharing.

Important parts of my communications are having code review with others, debugging sessions with the team, explain a newly developed feature or functionality to the team, participating in decision-making sessions with the team and external stakeholders. The most challenging part is explaining something technical to non-technical stakeholders. Besides, I struggle to speak as a new team member using all short or company-specific terminology; for example, Ericsson has all short naming conventions for different departments or teams. Another barrier of a global team is scheduling time or sessions with other team members working from a different geographical location, which is confusing for me sometimes. It took time to get used to it with the different time zone of my team members.

### **3 Diary entries**

#### **3.1 Observation week 01**

*Monday 09 November 2020*

Today is the first day of the last week of a three-week sprint, and we will have a demo day soon by this week. Today's primary goal is to test new features and expect several short discussion sessions with the developer to explain issues found and decide whether the feature passed tests and demo as completed or not.

From my previous analysis & flow of work, it is evident that data creation for some complex tests would be a critical task of the day as per test scenarios & test cases. There are two user stories of resource synchronization from the legacy system to Ericsson Adaptive Inventory about decommissioning a network trail between two network devices. The trail creation follows specific naming rules as acceptance criteria. This network trail has many prerequisites and rules to follow in both the source legacy system and the EAI, where it will be synchronized.

Execute test cases of these two user stories will be on my priority list and update manual test result documents as proof of fulfilling acceptance criteria to proceed for the demo & use that documents for further analysis of test automation. Also, expecting a pair or group working along with other developers or test engineers to accomplish some tasks as a team goal of the sprint.

After the daily scrum, start with a session with one of my test teammates to explain a few network equipment templates modeled and developed by me. To understand developed templates, expected device creation in EAI, and what acceptance criteria were presented to a test engineer.

Completed planned tasks, raised one defect in those two features, executed tests, and reported in Jira to track it further to debug. I have done a few short sessions of pair work to debug raised defect with the developer who developed the feature. As it is a minor issue, we decided that we may not block the demo but keep both as an open defect and fix them later.

Troubleshooting exercises and analytical thinking to simplify a complex issue were satisfactory as the day learning. Communicate efficiently and present a technical topic to understand and succeed together in a software engineering team that is more applicable when we turned to work from home, and we are making progress by days.

*Tuesday 10 November 2020*

At EOD of the previous day, I got an invitation and realized that we would have the current sprint demo today in the afternoon. I know some user stories have not yet passed the acceptance criteria to demo and have to help team members to accomplish that first as a priority. Besides, I have to prepare a presentation of the feature and create demo data & keep it handy to use during the resource synchronization demo.

Start with a discussion session about found defect issues with other developers and testers to determine one point from testers' observation. After a data analysis of the feature & design documentation, we decided that the issue is not a blocker & it won't affect any acceptance criteria.

In a pair-work session with another teammate, we have done our part of preparing for the demo of trail naming & trail decommissioning in a network. And as the demo day has replied to a few emails, the whole team has a clear view of any observations regarding my part of the entire sprint & no blocked issue.

I have connected to the customer's legacy system to demonstrate end-to-end data sync data from the customer's legacy system to EAI. During the presentation, my other teammate and I presented the network trail's resource synchronization feature, and those user stories were accepted by the product owner as resolved.

Another learning was on how to demonstrate a developed feature to the product owner and customer to be accepted as completed. Moreover, it also may skill to learn how to resolve any observations and disputes from a testing point of view to pass the work as per DoD or meet acceptance criteria.

*Wednesday 11 November 2020*

In addition to the ongoing pre-production regression test, there is an email chain going on about bugs raised by the customer's test team & both are on a priority now for the day. Involvement with debugging activity will require my presence and which is an excellent learning opportunity with the team to understand code flow and deep dive into complex technical issues. Besides VLAN management REST API tests are in the automation plan, and the development RF script is another exciting task to complete.

Start with manual testing of bugs and coordinating with developers simultaneously to find the root cause of bugs. It took an intense time and able to help resource synchronizer developers by analyzing log, bugs, code, which results in resolving a few bugs and defects as targeted to keep satisfactory progress in achieving the next milestone of the project.

EAI VLAN Object		Show/Hide	List Operations	Expand Operations
POST	/v1.0/idm/vlans/allocateNniVlanIds	Allocate NNI VLAN IDs for a NNI with anchor provider		
POST	/v1.0/idm/vlans/allocateVlanIds	Allocate VLAN IDs on network interfaces		
POST	/v1.0/idm/vlans/findVlanPools	Finds VLAN Pools for network interfaces		
GET	/v1.0/idm/vlans/getServiceClassDefs	Retrieve max and min VLAN number given Service Class		
GET	/v1.0/idm/vlans/queryVlanPools	Retrieve all allocated VLAN IDs from a VLAN Pool		
POST	/v1.0/idm/vlans/releaseVlanIds	Release VLAN IDs on network interfaces		

Figure 4. EAI VLAN Object REST APIs

Figure 4 shows several VLAN management API swagger documentation and a regression test on the pre-production environment required to ensure those APIs behave as intended. Telecom Network VLAN management is complex and has a physical & logical layer, including network infrastructures and devices. Hence test data creation is critical,

and I managed to do with my teammate using the Robot Framework script of EAI network design. We have covered test scenarios from essential to advanced VLAN management and developed Robot Framework scripts to automate so that any fallout events will be detected and notified.

```

*** Settings ***
Suite Setup      Set Selenium Implicit Wait  15s
Suite Teardown  Delete All Sessions
Library          SeleniumLibrary
Library          OperatingSystem
Library          XML
Library          String
Resource        ./Keywords/keywords.txt
Library        RequestsLibrary
Library        SeleniumLibrary  #Resource  ../ObjectRepository/dsmc-env.txt  Resource | ../ObjectRepository/sit-env.txt
Library        ConfigFiles/sit-env.txt
Library        JSOHLibrary

*** Test Cases ***

Verify Finds VLAN Pools for network interfaces
${auth}  Create List  svsada_to  svsada_to123
Create Session  findVlanPools  ${API_EndPoint_URL}  auth=${auth}
${Request_Header}  Create Dictionary  Content-Type=application/json  Source-System=FI  Source-User=FI
${findVlanPools_params}  Set Variable  ${Allocated_Vlan_ID_Suite}
${Response}  Post Request  findVlanPools  cll_connection_restapi/v1/sda/vlans?findVlanPools  data=${findVlanPools_params}
#assert 200 OK
Should Be Equal As Strings  ${Response.status_code}  200
#found Vlan pool
${found_Vlan_pool}  Get Value From Json  ${Response.json()}  1_vlanPoolBase
log  ${found_Vlan_pool}

```

Figure 5. Robot Framework Test Automation Script

Figure 5 illustrated a single test case in a test suite and its syntaxes, including all libraries to execute that test case. It is also another part of test automation to follow the test suites and their sequences, which is currently a senior automation developer fixing and forwarding the information to the respective developer. During the regression test execution, a defect is raised regardless of the pre-prod environment, so it is not blocking regression but reported in Jira to track and fix all environments.

Debug activity always has room to learn and widen the horizon of technical knowledge and code follow through browse the repository of application and server logs. As per my organization's expectation, one of the primary assignments to have experience in different project areas and various tasks of the day added good development & technical skills as an integration engineer.

*Thursday 12 November 2020*

Today's objective is to prepare all prerequisites and configuration for the regression test of network creation & resource synchronization from the customer's legacy system to EAI in the pre-production environment. Also, follow along debugging or troubleshooting & configuration with other developers and the person who is responsible for DevOps & infrastructure configuration as a learning activity. A development analysis session of new features is scheduled today with the customer, and I am required to participate.

We have an email from DevOps & the infrastructure team informs us that the required HTTP server to store data from the legacy system to a staging DB is up and running. I have followed along with developers to set up & configure the resource synchronization module in the application (EAI). It was not working as predicted, so a troubleshooting & debugging process continued. It was quite a learning of many different application components, and I found an issue by matching server log info with the module's code. We could not fix it immediately due to a holiday in India, and the person responsible for Infra & DevOps not available.

Have tested VLAN management defects, explained & reproduce the issue in the development environment to respective developers. Attended scheduled meetings with developers and Customer' legacy system stakeholders to analyze and finalize the resource synchronization module's planned feature. The planned feature establishes a network trail via sub-ports on the device sub-cards between two different devices installed in two separate technical rooms. It was another excellent learning event of making decisions to achieve the fulfillment of the use cases aligned with customer demand. I raised my observations and opinion from my perspective, as I am already familiar with the legacy system and relevant EAI application modules.

In today's scrum meeting, the Lead Developer assigned a new task of the "Task Management" module of EAI to me. I started installing and setting up my environment for that task by installing Oracle DB in a CentOS VM on local. I downloaded all required applications from a development server to run a stand-alone application on my local to develop and integrate locally without deploying in the development server or System Integration server. But to proceed further, I must study relevant module documentation from Ericsson Library, understand technical requirements, and set up my local development environment.

*Friday 13 November 2020*

As Team members from India are on holiday, I have to take responsibility for the retest & resolve bugs' status raised by the customer's test team. So, tracking all reported bugs in Jira and preparing test data for manual tests today are on my important task list. Besides, I must continue to join resource synchronization developers to fix configuration issues to continue learning activity. On this last day of the week, Ericsson Finland has virtual casual coffee and sharing meeting to update company information, achievement, and upcoming projects.

After the daily scrum, I started creating test data and prepared end-to-end test scenarios of resource synchronization functionality to manually test bugs on System Integration Environment. Able to report four bugs to resolve the status and two bugs reported to reopened status as those were reported as fixed but still exist. After the report in Jira and set the appropriate status of retested bugs, I have sent a status update email to all relevant stakeholders of our internal & customer side.

I have joined to follow along and learn the configuration session and debugging the previous day's issues as the resource synchronization module on the pre-production environment, not up & running yet. I have also sent another email to all relevant stakeholders explaining the session's issues and the same functionalities' regression test status. We are stuck on starting the test as the incomplete configuration and exist defect. It was excellent learning from this complex defect fixing issue, and participating in casual virtual coffee was fun. However, we were updated with a lot of information and shared knowledge in the event.

#### *Weekly Analysis of Observation Week 01*

Test data creation of a telecommunication network is critical and complex from a testing point of view. It needs analysis of features and scenarios, including positive and negative, to execute the test. This week, I have learned to apply effective and easier ways to create data in the System Integration environment and learned how to use Robot Framework tools for complex data creation. Introduce myself to the project's regression test framework and have understood to plan a testing framework for a telecom software test, regression test, and automation test while working on it with the team.

Developing API test automation scripts was another new part for me & it has complex prerequisites to run first. Both manual tests and manual test results are important to develop API test Robot Framework scripts, and I learned how to plan for scripts running sequences to run all the tests smoothly with valid data set. I also have a clear idea on another tricky part of selecting manual test sets to include in automation and not necessarily or not feasible to automate all manual test sets.

Participating in meetings with different stakeholders and demonstrating developed features to the product owner has many vital elements. It helps me continue learning to communicate better, explain complex technical things more straightforward way to others, and contribute to a critical decision-making event.



Previously I was not very confident about issues we handled, like debugging and finding a trace of HTTP server and resource synchronization issues. I expressed my motivation to follow and join experienced and expert engineers in the scrum meeting to deep dive into the activity to understand clearly. I started to understand more integration tasks where many different systems work together with an application like EAI by joining sessions and following others' work and activity.

Though I have also completed many testing activities, I struggled to find a time slot with the respective developers to clarify any doubts about the developed feature to create test cases. I figured that these communication issues are common while working with a team where members are located in different geographical locations. I started making notes of doubts and issues of any particular user story or all user stories in development by the same developers. It was easier to schedule a meeting for all relevant matters to clear up in one session.

Regression testing activity in a pre-production environment brings more new challenges from testing and debugging perspective. I was responsible for the test planning of three Epics of the whole software development project. During analysis and planning as per the testing framework, I also realize that all epics are interrelated, especially telecom network management and orchestration.

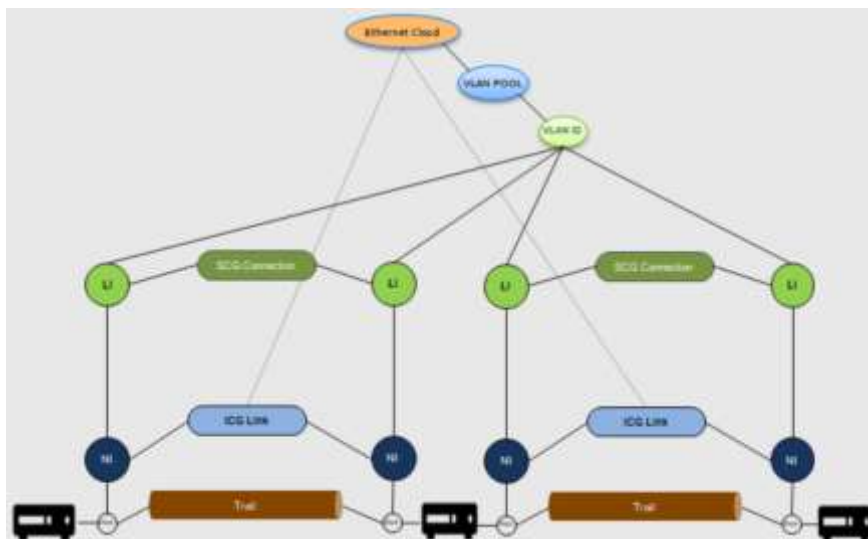


Figure 6. VLAN Id association architecture

An example illustrated in figure 6, VLAN Id allocation required network equipment, trails between equipment devices, Network Interface (NI) on device ports, and infrastructure connection group link (ICG) in the physical layer. The logical layer also uses a pre-created

logical interface (LI), service connection group (SCG), and Ethernet cloud in some cases. The network design & assign module must run before triggering the VLAN Id allocation API endpoint. It also needs to understand the network creation cycle for testing activities and tasks during the week to prepare test scenarios and debug.

There are many ways of sharing ideas during work and lots of individual research to find clarification of doubts and development topics. Still, I felt that communication among team members not at the best level yet, and maybe the blame might go to work from home. Apparently, I am continuously learning effective communication during remote work, and it is an experience to work in a team with different geographical locations and time zones. Refactoring, design patterns, customer focus groups, test-first development, and pair programming are not the tools of cowboy or cowgirl coders. These are the tools of developers who are exploring new ways of meeting the difficult goals of rapid product delivery, low defect levels, and flexibility. (Jim 2002, 150)

I also think that often in our task, many teammates engaged, and in some tasks, there are interdependencies to proceed, especially manual & automate tests. Though I have a way to work around when I know some team members won't be available during my working time, I could make them engage and responsive by applying the use of alternative communication like emails & digital workspace sharing. I also have it in mind to adjust my timing as my organization allows it.

### **3.2 Observation week 02**

*Monday 16 November 2020*

Ericsson Global training department will hold a pre-scheduled workshop today on the product we are integrating for a telecom operator in the project, and as a new team member, I am participating to have admin zone knowledge about EAI. Besides, also there were some pending issues in debugging & testing to complete during the day.

I got an invitation from the test team for a sync-up meeting in the morning and attended the daily scrum. We discussed issues we are facing in pre-production and also any blocker to complete testing activities. After the virtual meeting, we start preparing test data as the internal test team detects defects, and I knew the scenarios to test. After test execution, I found that two defects reported previously were fixed, and I have refactored test automation Robot Framework scripts accordingly. That means two tasks of Trail naming rules for an implemented feature and Trail creation in two different network devices in the

same site are working as intended in the implementation and met acceptance criteria to pass and successfully included in test automation.

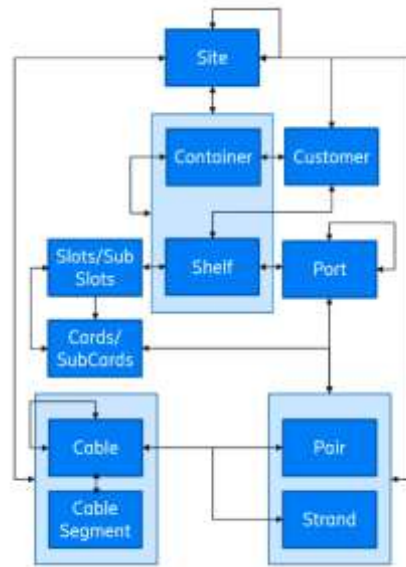


Figure 7. The base Inventory object model of the EAI

Participated in the workshop with few other teammates and learned details of many modules of EAI. For example, in figure 7, we have discussed and learned about a telecom network's base inventory model defined and the core of EAI inventory orchestration. Most participants were new to this particular product integration. The workshop was about the base Network object model of Ericsson Adaptive Inventory and the different functionalities. As an integrator, we know how to customize it according to customer demand.

Regarding testing issues, the sync-up meeting was advantageous, and the shared experience during the meeting helps to plan the quick test execution and mitigate blocking issues from test scenarios. The workshop was full of learning about the product's detailed functionalities that many of us were unaware of as we are working in a specific area of complete product integration. The underlying object and data modeling we went through helps to understand many development areas and interdependencies of different parts/modules of the product integration flow.

*Tuesday 17 November 2020*

There is a pending task of "Task Management" configuration and documentation analysis to start with this new feature with another developer. Though I have already studied related documentation from Ericsson Library, I also need to go through the new implementation documentation and design or customization details for our project's specific customer. Today, we have planned a knowledge-sharing session to present and share with my other teammates how to configure the resource synchronization job in EAI to pull resource data sent from an external source to our HTTP server. Besides, I got notification of a reported defect about VLAN, which was already fixed and deployed in a pre-production environment. I need to complete the testing part and close that defect.

I start analyzing all relevant documents about the Task Management module, including product documentation and specific customer design documentation. It is a new product (EAI) module in its latest release, and the customer wants to implement it. I found two-way integration of it, and one is from an external party to EAI another is from EAI to an external party. I have a session with my pair-work partner and decided that the configuration we need for it first is per customer customization design. It was not complete, and we send our observation to our Lead Developer for feedback. Both agreed to resume the activity tomorrow after the input from the Lead Developer of the team.

I also had another one-hour-long meeting with the templates modeling team and the customer representative responsible for it. We have discussed it thoroughly pending issues of template modeling and development. I have notes from the data dump & vendor documentation of the equipment we will model and presented decision-making points to confirm, for example, what will be the port access id and compatible cards of the equipment. So, we concluded the meeting with many confirmed decisions so that we will be able to proceed to those template development tasks now.

*Wednesday 18 November 2020*

Pre-production testing of external system integration with EAI is ongoing, and I have my task to complete if the end-to-end configuration works fine. As per yesterday's analysis, an initial configuration task needs to perform. Also need to have a session to clarify the feature and services integrated with other interdependent components developers. The project team will have a new resource joining the team, and the project manager informs me to arrange a knowledge-sharing session with him/her.

Start testing end-to-end resource synchronization testing, and a defect of load balancing cluster detected after several tests execution of same test case. It is creating two same objects in actuality when expected only one object. Reported a defect & immediately have an explanatory session with developers & Developer Lead and help them find the root cause by analyzing the server's log file. The root cause of the feature's strange behavior is from the load balancing cluster, which needs the customer infrastructure team to re-configure.

A new feature development of task management to handle unsuccessful synchronization cases to & from EAI needs to finalize the workflow and behavior with the synchronization feature. I have presented my analysis about developing the feature's framework to other developers who are developing different relevant components of the synchronizer. We concluded our session that we are not entirely clear and required our technical specialist and or Lead Developer's help to finalize the framework, and I have sent an email to all involved parties about the outcome.

As planned, have a kick-off session of knowledge sharing with the project manager and the new team member. It was just introducing each other, giving an overview of the project and details of the tasks I was working on. We have also discussed that ten different topics will be covered in ten sessions throughout the next two weeks. There were many things I am getting to know from senior developers from the feature analysis meeting and connecting many dots of real-life software integration.

*Thursday 18 November 2020*

Though I have a whole day-long training session today, I also have to join daily scrum to be updated and present that we have a problem to proceed with the new feature I started analyzing. After that, I have whole days in the lab to train the product & its different features and modules to integrate into a customer project.

We have hands-on experience in telecom network creation using Ericsson Adaptive Inventory (EAI) in the lab training. In practice, we have created objects like Site, Equipment, Shelf, Container, card, port, etc. We also created and learned how to create a template for shelves, cards, and other EAI objects and configure that in the admin zone of the application. After creating the required Network objects, added ports directly to a shelf and port wiring via Bulk Connections and Cross Connections. Today's lab training activities end with the successful creation of a telecom network. Five different sites are ready with all

equipment in the target network. One site is connected via multiple Strand Fiber between two provider sites, and a single Strand Fiber is connected from a provider site to the customer site. This training helps me understand the integrated network inventory software's whole scenarios and the importance of each component and object for a flawless network or service.

*Friday 18 November 2020*

I have a full-day training session in the lab with eight participants from different countries and advanced modules of EAI & develop new features to integrate main topics to learn in a lab in an instructor-led environment.

We start learning and creating in the lab all Adaptive Inventory objects of the telecom network. Today's topics covered advanced and complex network resources like Trail, Jumper, Trail Revisioning, Protection & Protected Trails, Network Topologies, Wireless Objects, etc. We have also learned, designed, and created a connectionless service object in the lab.

My favorite topic comes after the lunch break: developing, configuring, and integrating a plugin to customize functionalities and services offered by the product as per customer demand. We learned from scratch how to develop a plugin for EAI. We can develop it using many different programming languages, and recommended languages are JAVA, JS, Python, Groovy.

```
@Override
public SiteService_Create onEntry(SiteService_Create request) {
    logger.debug("SiteNameGeneratorPlugin --> onEntry INVOKED");
    String site = null;

    //SITE[1]. The name generated or inserted must be always in UPPERCASE
    site = request.getSite().getName();
    if (site != null && !site.equals("")) {
        site = site.toUpperCase();
        logger.info("Site Name >>:" + site);
    }
    request.getSite().setName(site);
    logger.info("Site Name before request return :" + request.getSite().getName());
    logger.debug("SiteNameGeneratorPlugin --> onEntry EXIT");
    return request;
}
```

Figure 8. Naming Generation Plugin code

Figure 8 shows our common developed simple plugin, which will be invoked while creating a Site object, and it will generate a site name to upper case. And configure in the admin zone with an extension point so that it will recognize at which point, for example, when this particular in will work. Using a plugin is one way of customizing the product and interface as customer demand is one of the most used development areas for the integration team, which is my job scope. Today's session was very fruitful for me, and I learned many complex integration areas.

### *Weekly Analysis of Observation Week 02*

According to James & Jim, Organisational Learning is about relationships. If a developer is extremely competent in technical areas but has no meaningful relationship with the customer, the overall results will be disappointing. (James & Jim 2000, 147). A customer-facing approach is always in practice in the integration process of any OSS BSS solution. We often have a decision-making meeting with relevant stakeholders from the customer side and the Ericsson side. This week I was the driver of such a meeting as the agenda was my development task previously, and we were making some changes according to customer demand. Effective communication is vital in this integration activity where a multi-country telecom operator may have many departments and issues to handle related to the solution implementation in progress. I am learning it continuously, and a customer-facing skill or mutual decision making is another learning in the integration engineering field.

It was super interesting to participate in the training about the EAI product. We dig down to a deeper level to learn the solution we are integrating for the customer. As an integration developer, it is necessary to know the underlying architecture of different application modules, which I learned throughout the week from the training sessions and lab exercises. During the three-day training session, we have a base knowledge of the application. We also applied theoretical & technical expertise gained from training to enhance EAI's customization and feature development skills from a developer's perspective.

An integration engineer's other required skill is configuration and troubleshooting, which was another main area we have hands-on learning in the training session. Through network creation in a lab environment, we have known tricky and critical parts. Many events occurred that everything was configured right and still not working as expected, so we went through troubleshooting and reconfiguration exercises during the training, which is essential in the real integration phase.

While working in previous sprints for integration development or testing, I was also facing another complicated issue: I was unsure about interdependencies between components.

After this week's training session, it is clearly understood that a very silly mistake of another feature or a different part of network creation may cause a problem for a different phase of the network in other application elements. One of the three components of the adaptive Development Life Cycle is learning. If we are to understand adaptive development, then we must learn about learning. (James & Jim 2000, 144).

Learning of plugin development and configuration of the application is a beneficial skill of integration work. Though it was a small part of the whole learning, I am more interested in it, and many doubts about plugin development are now clear, and I am thinking of working on some plugin development in the coming sprints. After the training, I think that I should have this skill development training more often in different areas of my work. By this week's practice, I have learned many admin zone configurations that teammates never fully clarified during work together, and the experienced trainer has made it clear during lab exercises.

Analysis of documents and requirements of a new module integration is another part of learning, and this was my new area of work I just started this week. It was quite a blurry topic or module, and I was unsure about how to proceed with this analysis. I found it very common for a new team member to start overthinking at the beginning of any new work area and tried to simplify things and break down the process to generate the idea to implement the framework for the development.

During this week's training session, I could not keep pace with other participants as they are more experienced in the telecom domain, where I am a new developer without the telecom domain's work or study experience. But after the first session on Monday, I started to study training materials to prepare myself for the next day to proceed along with the same pace in the lab. It was good that I was forced to quickly pick up the topic and learn some domain-specific searching from google and Ericsson Library. A quick learning practice opportunity was there to research and study a topic and apply it in practice immediately.

It was a week of learning in many ways, and I took up all new things this week, which will lead to perform better and contribute more to teamwork. I have gained more new skills, especially on EAI integration, but I was lost many times during training due to confusion and complexity, which I could mitigate by studying the course material earlier from Ericsson library. If I attend any new workshop or training in the future, I will research and explore the topics to get on it entirely. I am convinced from my learning and recent experience that integration development requires lots of analysis and research than maybe real



development tasks. As it is a complex area for a beginner, I will spend enough time analyzing the topic and researching it before a face customer meeting or starting a development task.

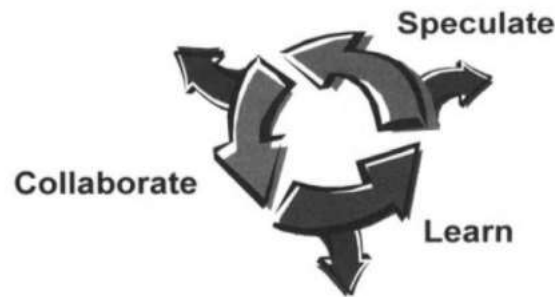


Figure 9. Adaptive Development Life Cycle (James & Jim 2000, 41)

as it shows in figure 9, it is very similar to Adaptive Software Development when widespread learning and collaboration occur in agile or scrum. This week I have a mixture of training and integration tasks, so I was speculating, learning, and collaborating.

### **3.3 Observation week 03**

*Monday 23 November 2020*

The Resource Synchronization module pre-production regression tests & related bugs are on my task list, and it will take time to accomplish the whole module and a continuous topic for a while. I should have arranged some time with the solution architect to clarify many of my doubts and understand the use case, data structure, and workflow to develop a Task Management module's common framework. In addition to those, I also have to talk with our infrastructure team lead to deploy the EAI application on my local VM. Also, I need to fix some jar issues for the Task Management project in eclipse.

After the daily scrum, I started to prepare a dataset to test data sync with EAI's legacy system in a pre-production environment. I also tested bugs we raised and fixed to determine that the feature can go to production after the regression test phase. While testing, bugs also tried to trace back the log files and the root cause of it. Because I am also closely working with data sync development, and it is complex to find the root cause of something when it involves legacy systems. The first half of the day and another session in 2nd half takes for the activity, and I planned for the next day to continue again so, I just switch as I have to keep doing other tasks.

I had a call with the Developer Lead & Solution Architect to understand the data flow, use cases, and the Task Management module's underlying architecture to start integrating the module to the application we are implementing for the customer. I understand how an application module is developed and how we should develop a common framework for every module on top of the real application layer from today's meeting. I made all notes and recorded the meeting for my future reference.

After the meeting, I immediately contacted the Lead of the infrastructure team to consult about fixing some jars that I think are required for my eclipse development project. We have a short call and found that some jar files got corrupted, and we need to replace them in project libraries. Infra Team Lead helps me find the right files from sever, and I managed to fix those issues and run a maven build successfully in my eclipse project. It was another learning & always a confusing thing, which only became more precise with doing similar libraries fixing and working experiences with similar topics.

*Tuesday 24 November 2020*

Sprint deadline approaching next week, and from this sprint accomplishment perspective, I have already planned to work on a few following topics today –

- Write code for EaiTaskService and EaiWorkOrderService classes for a common framework of the application.
- Write code for EaiTaskManager class for a common framework of the application.
- Refactor and ready development environment of Eclipse workspace and project cli\_common\_eai\_commom.
- Have a knowledge transfer session with the new team member.
- Check all pre-production regression tests & bugs status, fix the defect test again and send an update email to all stakeholders.

```

// Load Task
public Task loadTask(TaskKey key, ManagedObjectFetchSpec spec) throws AbstractException {
    try {
        return (Task) taskService.fetchByKey(key, spec);
    } catch (Exception e) {
        throw new SdkException(e);
    }
}

public Task loadTask(TaskKey key) throws AbstractException {
    ManagedObjectFetchSpec spec = new ManagedObjectFetchSpec(Mode.ASSOCIATIONS);
    return loadTask(key, spec);
}

// Create Task
public Task createTask(Task task) throws AbstractException {
    try {
        ManagedObjectFetchSpec spec = new ManagedObjectFetchSpec(Mode.ASSOCIATIONS);
        return (Task) taskService.createAndFetch(task, spec);
    } catch (Exception e) {
        logger.error(e, e);
        throw new SdkException(e);
    }
}

```

Figure 10. Task Service class method for Load & create tasks

I started writing code using Java and following similar service classes of the common project of application. I again found many third-party libraries of the application and the latest patch needed for my eclipse project to build with maven. I tried to fix it temporarily by Adding External Jars in the Java Build path in eclipse. I had to study documentation to understand and follow other similar projects and service classes to determine how to develop this and the service class's expected behavior. As shown in figure 10, I have developed common services from other components and integrate them into another module when needed.

Knowledge transfer session was another learning topic for me, too, while introducing different application cases of application modules to a new team member. We both thoroughly discussed many issues, have hands-on practice with some tasks, and use cases to understand better. I am a little surprised and come to know that the person is with ten years of experience working in Ericsson, and I am just a newbie having a KT (knowledge transfer) where I am leading the session.

In the later part of the day, I started checking regression test Jira tickets in the scrum board and related bugs logged in the Jira bug board. And it seems at least a few bugs have been fixed, which was blocking the pre-production regression tests. After retest, the fixed bug found new issues that again stopped some test cases and sent an update email to all pre-production stakeholders.

*Wednesday 25 November 2020*

Sprint target achievement is a priority now, and the sprint-developed user stories are going to deploy in SIT (System Integration Test). I have to concentrate on testing activities for a couple of days, and I have planned today for the following –

- Create test cases for this sprint use cases for those user stories I have analyzed, especially the Data Sync module.
- Refactor previously developed "Task Management" common service class of task and work order.
- Participate in a pre-scheduled meeting on Data Sync for user stories developed in this sprint.
- Host a scheduled knowledge transfer session for the new team member.
- Check all pre-production regression tests & bugs status, fixed defect test again, and sent an update email to all stakeholders.

I started to create test cases for some Data Sync user stories after the Scrum meeting. As I have analyzed and been involved with a few debug and studied the module's development, I understand the whole module, and it was not challenging to complete all test case creation. So that as soon as the deployment is done on SIT, the test team can start manual testing and plan for automation scripting if required.

I have done small refactoring of the Task Management module code in eclipse though there are some other issues to handle, which may complete when we finish with this sprint testing activity. This refactoring activity gives me further understanding of development issues and JAVA programming knowledge though I am still a learner of using application SDK.

I have participated with developers & relevant customer representatives to understand the new features of Data Sync. Developers presented a short demo, and we discuss all doubts and performance issues of use cases. It improves my further understanding of the application module, which will help me in testing and development areas.

I hosted the KT session similar previous way, and we proceed to present the next topic of the project and use cases of the developed features in this project. I found it interesting that I also start to know many things from that new team member.

Before wrap up, I again checked the Jira board and linked bugs with the regression test ticket. After careful check and a test run found another new defect, reported it to the Jira bug board, and similarly sent an update email.

*Thursday 26 November 2020*

From previous experience, I have already planned for the day like the following –

- Review code with a senior developer.
- Review bugs of this sprint with relevant developers and test fixed bugs.
- This sprint user stories test in SIT & participate in debug activities.

I have participated in a code review session with other developers for the Service class developed by me. I took note of some issues and suggestions from senior developers, which is vital for me. We have gone through other module codes to understand and compare and learned to browse through the project's repositories, which I was not confident to do before as instructed not to do alone.

Started review bugs with respective developers and reviewing fixed defects by running a test. A great activity to do together with a developer, so many unknown areas came to light, and I learned tricky software engineering things.

I got notification from the Jira board of stories already deployed in the SIT environment, and I started data creation of those user stories to execute test cases any time. I have performed some test cases and found those passed and join with another session where senior developers are doing the debugging activity. I am always interested to follow along to learn the more in-depth tricky topic of integration engineering.

*Friday 27 November 2020*

We will have a virtual meeting of Ericsson Finland after the lunch break, and I have a specific plan for the day to execute test cases and create critical data in the System Integration Test (SIT) environment for test purposes. I also have known and planned for fixed bugs and defect retesting.

The first half of the day was busy with all the testing activities. I was executing the remaining test cases from morning right after the scrum meeting. It is always crucial to create test data. After I planned for the data set, I manually created some data. I also seek help

from other team members to run automation scripts to create data in the database to have a fulfillment data set to perform test cases.

Meanwhile, I also helped my teammate execute another test and find issues in the templates model. I quickly found the issue as I was previously involved with template model development. As we did together, I also learned some template issues that I was not aware of before.

Later part of the day participated in a virtual team meeting of Ericsson Finland and got updates on many local company issues, pandemic measures of the company, work from home updates, etc.

### *Weekly Analysis of Observation Week 03*

In Ericsson, I found learning together approach among team members, as I mentioned before, and sometimes a team goes beyond rules to achieve a planned goal. An extensive regression testing of the pre-production environment was crucial to handle and learn many regression tests and final time development of automation scripts. It required excellent analyzing and planning skills to complete regression, especially in pre-production releases of solution effectively. From doing it, I learned how to do it and experience a different side of tests and regression tests for such a complex solution of network resources management solution.

A telecom network is a very complex area to maintain and create a solution for it. Developing a feature for a telecom network orchestration sound requires understanding the hierarchy and workflow of different network components. Sessions with the Lead Developer and Solution Architect of the project regarding the Task Management module of the solution facilitated me with the best opportunity to learn about the telecom network domain, network-related physical & logical resources, hierarchy, and workflow.

Setup and configure a java eclipse project and build it with all required EAI libraries were troublesome, and I had to do a lot of refactoring and troubleshooting at the beginning, which was interesting learning many small but very tricky issues of JAVA development and using Eclipse as IDE. Integration development also slightly different than typical software development as we are developing a feature to customize the standard original product to implement it as per customer demand. It is also new for anyone in an early career to work with Out of Box API (OOB) Services, which is enabling Ericsson Adaptive Inventory (EAI) more customizable. There are so many ways to customize and create new

services on top of the application layer. So it is a new way of development and learning by developing a common framework by using OOB APIs.

JAVA programming skills are another part of my interest, giving me a way of learning by research and finding the solution to many smaller programming issues of everyday development activities. I also started finding solutions from my senior teammates and the Ericsson developer community as many problems I can't find from google searching, and other public sources as most of the Ericsson's solutions and resources are internal & private. I realized that a developer needs to connect with an internal and external developer community is a must to overcome many challenges of software development.

Code review and bugs review is another part of a developer team of integration engineering to understand better and determine the best way of integration development and including debugging. My understanding always is more clear following along with my senior engineers as I found they are the best in the industry. Code reviewing allows expressing ideas for the same and also learn why is the chosen solution is better and the flaws of another similar way of the same solution.

Sharing knowledge with others was fantastic to practice my own to deliver to others, and also, I validate my knowledge with others from discussed ideas and ongoing feedback to each other. To be a host of the KT (Knowledge Transfer) sessions, I delivered it, but I was also receiving it from more knowledgeable participants in reality. From those sessions, I came to know many hidden areas of the telecom domain useful for the development areas. And after all, these activities are in many ways important for everyone in the industry as effective communication is a very important competency to succeed in the software development field.

I have challenges in understanding the documentation and architecture of a feature module as it is a completely new topic of configuring and developing from OOB APIs. I did some research from external and internal sources to have more knowledge on it, and senior members gave a hand and suggested ways to follow and resources to study, which made it easier for the first time and gave a lesson for the next time to do it independently.

Regression tests can be defined as the system's behavior remained the same as unit tests or acceptance tests to determine an application's desired behavior. Once the tests pass, they become part of a regression test suite to guard against any unexpected changes and should be automated to ensure continual feedback. (Gregory & Crispin

2014, 419). There were issues to solve and complete regression tests. All respective developers of different components were involved in fixing and planning tests, an excellent testing approach for a larger-scale project to be successfully released.

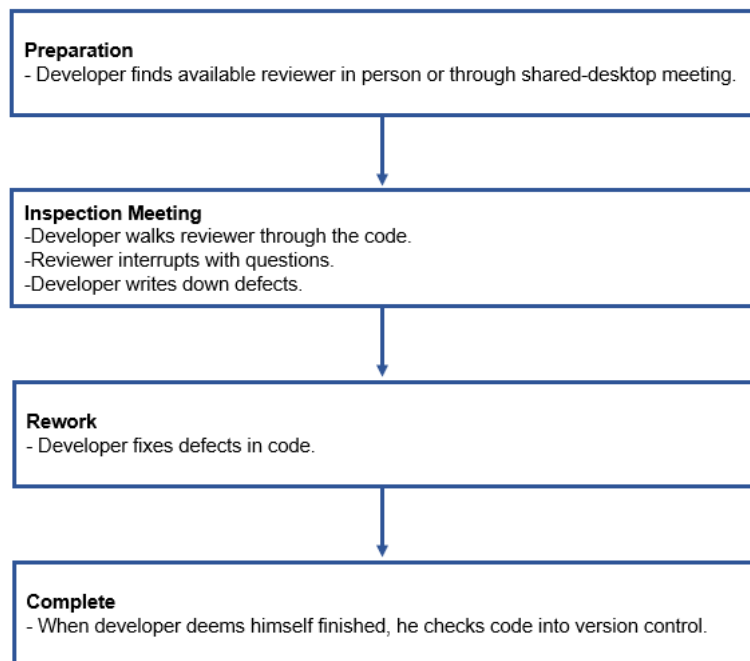


Figure 11. Over-the-Shoulder Review Process (Cohen, Teleki & Brown 2013, 28)

Code review and pair programming is another effective way I may follow in future sprints. As shown in figure 11, some studies found it controversial, but in modern software development like this project, successfully embracing it and getting results. By following my fellow team members, I also adapt and will try to be more useful in pair programming and code review to refactor code, enhance productivity, and minimize flaws and bugs. According to Cohen, Teleki & Brown (2013, 37), it is a controversial issue about whether pair programming reviews are better, worse, or complementary to more standard reviews. The reviewing developer can get deep insight, get more involved, and have a great thought about issues and ways of implementing.

Integration tasks of this project are a very agile way where many stakeholders are involved. I should include more team members to review any doubtful issues, at least in a short code review session, which will make me more confident while participating in a decision-making event with the customer as I have enough reviews on the topic. It also allows everybody to share knowledge, find a beneficial process to simplify complex issues, and establish collaboration among members, which is essential in this large project, unlike



a small software development. An arranged knowledge-sharing session, such as introducing a new member to the project, can be better with joint activity rather than delivering something and a roam to learn from each other.

### **3.4 Observation week 04**

*Monday 30 November 2020*

Tomorrow will be the sprint demo, and most of the developed user stories are in the status of System Integration Test, and without any pre-analysis, I know there will be testing & debugging activities: I started checking the Jira Kanban board & Bug board, and makes a plan to do following –

- Create data for test case scenarios.
- Test current sprint's Data Sync user stories, which I have analyzed before.
- Some defects block some user stories raised earlier and join the debugging session with respective developers.
- As I am responsible for the epic, I have to update all stakeholders of that epic status.

From the beginning of the day, I started to create data set for data Sync user stories, sync some data from legacy systems, and use previous data from the last sprints. It was not that complex this time, but to do it carefully so the data set's validity cannot be wrong to execute test cases. The data sync module of the application is becoming complex to implement at this stage, where legacy systems have many flaws to comply with the application. After test data creation, I analyzed bugs raised by the customer test team to determine the appropriate test scenarios and refine my test cases accordingly. A practical analysis skill is another critical area of competence an integration engineer needs. I am developing those skills from various tasks of testing.

I have executed test cases and passed to proceed for demo and started to have a few small sessions with different developers to fix and debug defects. A few we able to fix, test, and passed as those user stories meet acceptance criteria. Some defects were still not able to fix and update those tests as failed. I have checked all user stories and defects to send an update email to the whole team. I am somehow responsible for overseeing this data sync epic and make sure which are meeting acceptance criteria and proceed to demo to the product owner and customer representatives.

*Tuesday 01 December 2020*

As in the second half of the day, we will have the demo of the sprint, and I have on the priority to-do list like mentioned below –

- Left some defects to be fixed yet and will join code review to debugging for those defects.
- Update Jira board and move user stories to the appropriate status.
- Test defects & update user stories linked with those defects.
- Prepare demo data and participate in the demo to show data sync features.

I started the day with a very active mode with others as we have to debug defects and test user stories to pass and include in demo topics. Start the session with a debug session and retest defects in the SIT environment, and found some defects are fixed already but not all. I have executed some manual tests, passed a few user stories, and sent an update to the Developer Lead. For some defects blocking some features, I invited the test team to a meeting and decided on some user stories we divide and pass for the flawless part for demo and the rest part move to the next sprint development task. After the meeting, I have updated all Jira user stories boards with appropriate status, with comments where required sent an email to follow the latest update for the demo.

I have participated in a scheduled meeting with developers to recheck demo data creation and review presentation slides with others. In the meeting, one of our Solution Architects assigned a task to create demo data and join the feature demo presentation to show the feature to customer representatives and product owners. It was another exciting part of the day to show the demo's feature in front of everybody, which makes me again confident how to present a software feature from a developer's perspective and business perspective at the same time.

*Wednesday 02 December 2020*

After yesterday's demo, I have in plan to focus on the user story I am developing for the task management common framework, and today I will continue to do like below –

- Develop a Work Order service class and Work Order Manager class for the module's common framework.
- Find trace of a few critical bug and test those defects which could not pass acceptance criteria.

- Follow up regression test in the pre-production environment and execute test cases.
- End of Day update email to relevant bodies about regression test status.

```

package com.ericsson.common.service;

import org.apache.log4j.Logger;

public class EaiWorkOrderService extends AbstractService {
    private static Logger logger = Logger.getLogger(EaiWorkOrderService.class);
    protected WorkOrderService workOrderService;

    public EaiWorkOrderService(GraniteServiceFactory graniteServiceFactory) throws AbstractException {
        try {
            workOrderService = (WorkOrderService) graniteServiceFactory.getService(WorkOrderService.class);
        } catch (Exception e) {
            throw new AsiException(e);
        }
    }
}

```

Figure 12. Definition of the Work Order Service class

After last week's review, I figured that I need to write a new class for Work Order management, which will hold all tasks from failed scenarios of data sync. So, I started writing the work order service class (May refer to Figure 12) and following the same convention and structure as other common service classes I develop. Most importantly, the work order needs another manager class to generate a work order in the specific format demanded by the customer and designed by our solution architects. In the below figure, the initial step of service class definition, this class has crud functionality of work order as a service to use from this framework.

Also participated in some test execution in the latter part of the day, followed by debugging session of left defects in the last sprint. I still join those debug sessions and help developers find the cause of a defect and an interesting way of learning for a new integration engineer. Also, follow up regression test by getting updates from the test team, check database, and data sync history from the application itself. I have analyzed my findings and sent an update email to all relevant parties at the end of the day.

*Thursday 03 December 2020*

As there were pending task to complete Task Management common framework to implement, I intend to work on it and some other ongoing test & knowledge transfer session like the following –

- Complete the work order manager class to generate the naming of work orders via a common framework.

- Help test teams to their activities, which I was analyzed but not having time to proceed.
- Have a scheduled session for knowledge transfer with the new team member.
- Check test activities with the team and determining defects, and log them.
- Update team about regression test status of pre-production environment.

```

public String generateWorkOrderNaming(String countryLabel, String bundleName, String workOrderName)
    throws Exception {

    String tempWorkOrderName = null;
    String finalWorkOrderName = null;
    Integer sequence = null;
    if (sequence == null)
        sequence = 1;

    tempWorkOrderName = countryLabel + "_" + bundleName + "-" + workOrderName;

    if (tempWorkOrderName != null) {
        eaiLockMonitorService.lockByKey("WORKORDER_NAMING", LockScope.PERSISTENT);

        // Get last sequence no value
        try {
            WorkOrder workOrderDto = new WorkOrder();

```

Figure 13. Declaration of the Naming generation method of work order

I started writing the Work Order manager class for the common framework, which will enable all plugin callers to generate work order names as per specifications and design. Finally, the Task Management module will generate all work order name with Country Label, for example, for Finland - FI, Bundle Name for example, for core network- CNT, Work Order Name, for instance, XyzToEAI-exception and a sequence number. The above figure shows the class's method declaration only, but inside holding very long business logic about 124 Java code lines.

In the second half of the day, we have a knowledge transfer session similar to the last time and share many aspects of the project. I also participated in a session with the test team to help them in some test execution and introduce them to some user stories. Have checked and sync up with the test team to determine whether the developed feature's flaws can be considered defective. And in the same session, we also checked the history and progress of the regression test, and later I have drafted an email to send the latest info to all.

*Friday 04 December 2020*

Developing a common framework becomes a bit challenging. I assume it will take a lot of my time to complete because this framework will be linked with many network automation

and orchestration modules. Hence developing that business logic seems more complicated to figure out and implement. This will be my main focused area for today-

- Refactor the whole framework again to comply with relevant issues and exception handling
- A more in-depth analysis found that I need to implement more services to use.
- Implement business logic as required for the name generation of objects in the task management module.

I started refactoring the whole framework and modifying logics in different classes to work as expected for all network components as it is not only for the Data sync module to use but also other modules of the application. A new service class called Queue Service Class added to the framework, which took me more than the first half of the day. I also modified the logic of CRUD functionalities in all previous services classed developed previously and modified Work order Manger to generate naming and associating with other objects.

#### *Weekly Analysis of Observation Week 04*

Demonstration software or part of the software in a business meeting with participants from technical and non-technical backgrounds needs enough confidence, which comes with experience and repeatedly doing it. I think it is my continuous learning as a developer by participating in various meetings, especially presenting a tricky feature in the sprints' demo. Testing and debugging a bug always gives a deeper understanding of the software. As usual, I want to participate in a debug session, which allows me to understand better the telecom domain's network inventory application and network flow.

Analyze design, understand the need, and break down the process to turn in the programming language are must follow carefully to develop something involved complex business logic. A telecom network and its inventory is a complex domain to develop a solution for it. I found making many mistakes in programming making me more competent in the complex area of software development. I had to break down the process many times in the week to develop the framework I was working on. Solving one after another breakdown process gives me the result at the end, which was the aim to achieve.

Reanalysing the documentation and design of the developed framework for common use were essential as they may also affect other components. A proper reanalysis often ends up with the need for refactoring code. This week's common framework development tasks

and refactor it many times was a good start as a beginner, another study of the code repositories, further in-depth analysis of design, and even seek help from other senior developers. It was challenging to refactor the framework, and a wide range of research on the topic helped and managed to do the required modification. I also learned that refactoring code also needs to update the required libraries to build the project successfully.

More research and study allow me to solve the problem of parallel refactoring & modifying code, and update *pom.xml* file with the libraries. I learned to execute terminal commands quickly to install libraries and fix Maven configuration and know-how to create a command-line script, even to install libraries on the go. I am convinced that development is not just writing code but also fixing the configuration, updating the development environment, and for a java project update libraries as required with any extension of the project. It is the most critical part to update the development project and its libraries always and check if a modification required a new library or configuration.

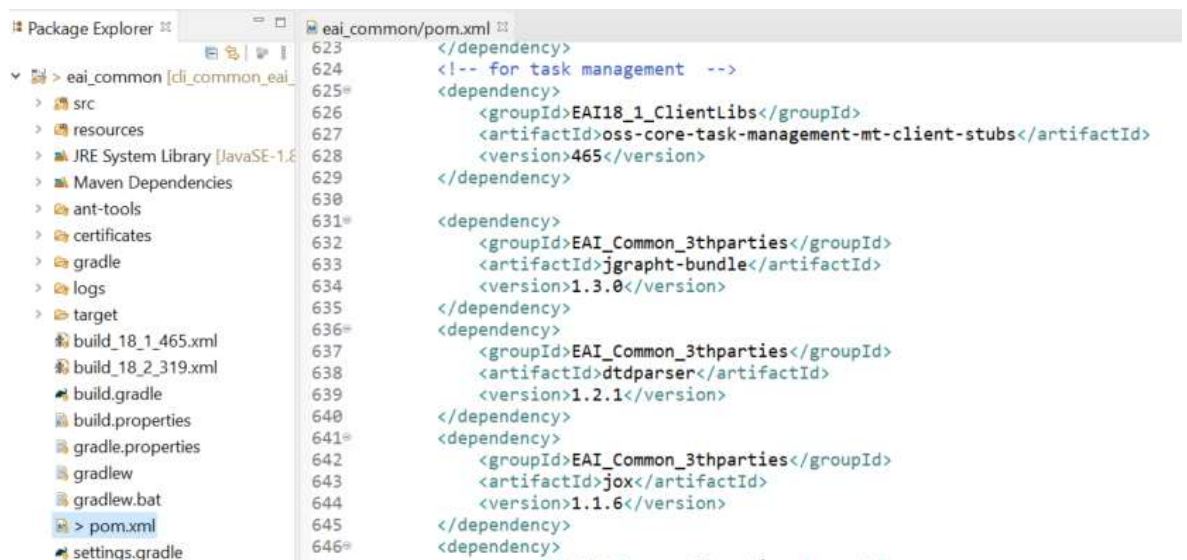


Figure 14. Declaration of the Naming generation method of work order

Figure 14 showing that the module framework also required core and third-party libraries, which can be added in *pom.xml* files of the project. At the root of our project, the *pom.xml* file, which is the Maven configuration file for our project. We can see the dependencies inside the file. To add more functionalities to our application, we will add more dependencies manually to the *pom.xml* file (Hinkula 2018, 15.). My modification often makes the project build failed and made errors, which can sometimes be avoided by updating the configuration and libraries.

I put my effort into following the project's activity and taking responsibility in some smaller areas from a project management perspective. I was continually overviewing the Jira Scrum board and Bug board to follow up on those areas. I was somehow helping the team leads and project manager to have the latest update. To do this, I also had to study and go through many project documentations from confluence pages and understand the business requirements of a specific epic or fulfillment target of the sprint. I have learned many project areas that were not needed for my tasks to complete but essential to know when following up the project objectives. So I also understand the need for accomplishment and teamwork that drives the project goal.

All those debug sessions, sync up session with the test team, and knowledge transfer activity of this week was another excellent part of satisfaction with full of learning. I was able to perform a leading role in a few smaller topics as I was keen and took it extra in addition to my task to overview and update others and determine the resolved status of user stories from the integration team.

I faced challenges throughout the week and overcame each problem required, even allocate out of work extra time sometimes. Again, I learned that breaking down the whole into pieces is the most effective and appropriate in software engineering tasks.

### **3.5 Observation week 05**

*Monday 07 December 2020*

From last week it seems that the developed common framework seems to work as expected, but I have comments from our Solution Architect that a careful test before deployment is necessary, and I will be focused on that today and have planned to do -

- Review my code with a senior developer, refactor if need, and debug if any flaw is found.
- Create a test project and write test classes to perform sanity tests of my code.
- Execute a few tests on the Pre-production environment and check the history and status of the regression test.

In the scrum, I have discussed the common framework development, and for the final time, I want other developers to review it and answer any queries or doubts. After the scrum, I have a review session along with two other data sync module developers, and we all agreed that it could be the final version to deploy.

As I have feedback comments in a meeting about the unit test of the developed framework and suggested creating a test project for it, it can be reusable to test further feature development. I have started to create a planned test project for testing, and for the first time, it seems not to have the required database connection class and other utilities for the project. I start research again about it and start fixing issues one by one and understanding how I should do it. I also applied the whole process's breakdown in many smaller pieces and figured it out one by one.

I have another responsibility task to check and follow up regression tests. I ended the day by performing a manual test of the data sync feature and analyzing how the pre-production environment works, particularly with the component concerned.

*Tuesday 08 December 2020*

Since yesterday I could not complete my test project set up to test the framework I developed; I need to fix all the troublesome issues of the project today and refactor or write my test classes again. I will keep working on those topics mentioned below –

- Troubleshoot and fix all unit test project issues to test developed by import or placing the required class from my framework project.
- Write test classes for different and straightforward tests of expected results of the task management framework.
- Test data sync epic user stories on SIT environment, close fixed defects, and update Jira accordingly.
- May involve in analyzing and writing test cases of data sync delete use case.

First things first, in the morning, I started fixing my test project in eclipse. Another research by google search and the internal developer community helps me understand Java eclipse project fixing. I have copied some utility and DB connection classes from common project repositories and placed them in the test project. I also refactored those classes to fit into my new eclipse project as required, and that initially fixes most of the error issues in the test project.

After fixing my eclipse test project, I started writing test classes to test the framework's functionalities, especially required naming generation as specified design documents. I also seek help from my senior developer as they also use the same testing project locally.



I was able to test the framework functions and verify it is creating the intended object instance in the application.

In the later part of the day, I have executed manual tests on the data sync topic and sync up with the test team about finding to determine if the team wants to include it in automation or not. This activity also included closing a few defects, update Jira bug boards accordingly, and sent an email to relevant persons for the latest update. I also rewrite a few test cases in Jira to make the test scenarios appropriate for quality assurance.

*Wednesday 09 December 2020*

After completing the development of the task management module framework, I will start analyzing the next user story of the same module as instructed by the Developer Lead. I am prepared to work on and proceed with the following tasks today –

- Start analyzing new use cases to develop the feature of mail notification of failure data synchronization job.
- Commit and push the final framework code to the remote bit bucket repository of the project.
- Keep analyzing test cases further and refine test cases as it seems a complicated feature.
- I Will also have a session with developers to determine critical issues of data sync.

A user story of task management notification handling was already in the design phase, which I have started analyzing today. I have gone through all available documentation regarding task management notification and integrating a third-party SMTP server to the EAI task notification feature. The idea is that failed data synchronization jobs from the legacy system to EAI, and vice versa cases will be notified via an email to people responsible for handling it manually. To solve the issue, we may need more information as there are not many materials on it in the internal documentation library. I have started to read all available documentation first and prepare documentation on further requirements to share with our solution architects, development lead, project manager to provide to the development team. The detail and careful reading of documentation is essential to make the intended document for a further requirement to implement the use case's integration.

Few user stories need further analysis as the application getting complexity with more features. I got an assignment from the test team to further analyze and refine test cases and prepare automation requirements. I have re-analyzed and found more issues to include in

the test scenarios, refined existing test cases, and added more test cases in Jira. I have tried to prepare a network service design flow to clarify the end-to-end testing flow and the requirement for initial automation where many external and internal components may work together. I also have made notes that for manual tests and to pass the acceptance criteria, we may stimulate different external components by API testing tools like postman or soapUI.

At the last part of the day, we have a scheduled session with the developer team of data sync to thoroughly discuss the issues we are all facing and how to proceed with the integration task. As this data sync module is working and syncing with very old legacy systems, it causes many problems to integrate it with our application. Still, we decided to merge our feature concerning that legacy system is the master source for us as that is the thumb rule for similar software integration.

*Thursday 10 December 2020*

From yesterday's session with developers and refine test cases of data sync, it came to light that the user story of the use case of data sync deletion is much more complicated than we thought it would be. Today I will be working on the topic and others like –

- Test data creation and perform test cases of complex scenarios of the deleting data sync feature.
- Analyze a test SMTP server for the mail notification feature of the task management module.

After the daily scrum, I started to determine test the data creation strategy for a complicated use case of network deletion user story. I came across that it needs more information to analyze and details of the legacy system's data structure and validation. I have logged in to the legacy system, started extracting information, and combined my previous knowledge and analysis with newly extracted information from the legacy system. I have planned the data creation and test execution strategies and call for an instant meeting with the test team to review the planned strategy. Members of the test team agreed to follow my plan and strategy for the whole use case test execution, and they understand better from my analysis result. They also commented that they can now refine or add more test cases to it to assure the feature quality.

In the second half, I again dived into the task management mail notification user story. I started researching SMTP servers using various sources, including Google search, the internal development community, and the internal documentation library. I intend to prepare myself to provide consultation and provide a suggestion for the integration implementation in a decision-making meeting with the customer as I know this is flagged as soon to decide the matter.

*Friday 11 December 2020*

The newly developed deleting feature of the data sync module is still doubtful for some test team members. I should take the lead today to make everyone understand as I have entirely understood from my analysis of the whole use case and relevant objects and their life cycle. To have a result-oriented week today will keep working on –

- Perform tests together with the team of deletion use cases of data sync epic.
- Again, re-analyze the task management interface and understand how to develop a plugin to be exposed to a third-party SMTP server.
- Conclude test execution and raise all defects, also accordingly update Jira status.

From previous progress and further checks on the test's status, I understand that end to end scenarios and how to stimulate it without external component is still not fully clear to all test team members. I took the lead in a combined manual test execution with the team and executed many test cases together with the team and found a few defects and logged them in Jira to track them. I have also explained the concerning issues that need extra care while testing and how to ensure the quality of the feature from testing and help developers minimize bugs.

I am in continuous research to solve this real-life problem of task notifications sent to an email. My main objective is to implement the task management module's system interface to enable an external mail server integration. I need to understand how an SMTP server works and how our application service will be connecting if the system interface has any event notification of task management.

At the later part of the day, we had a sync-up call with the test team and analyzed all our findings of a deleting feature's specific user story. We found a few defects and logged them in the Jira bug board, and linked them to block that feature.

Unit tests your code, and test the feature before deployment is a required part of the project, and I was struggling to do that for the first time. It was not simple as it seems to as it required connecting the application database and also should have access pass with proper authentication. I tried differently, and with the help of a senior developer, I managed to achieve what I intended to test.

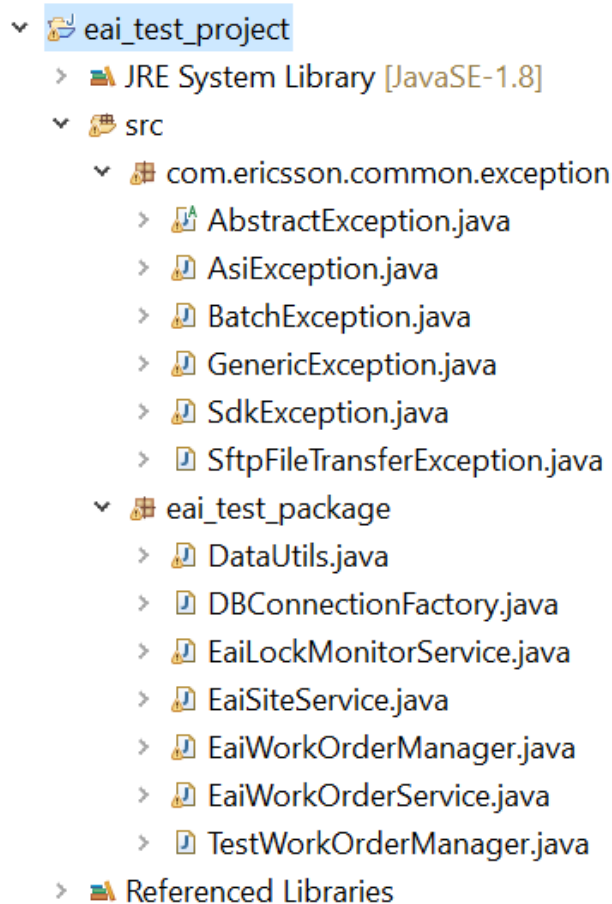


Figure 15. Test project setup for functional test

In figure 15, the test setup has required classes from the common repositories as those are required to connect databases, handle all exceptions, lock monitor, and other services. I learned a new way of testing in this project, and the approach was a nice workaround to avoid much complexity before deployment. The practice of testing makes the changes work in the server and the application without deploying it. So, there is no one straightway to solve a problem. I realized that even the same things might have many workarounds, and sometimes workaround works efficiently.

A troubleshooting or a problem-solving habit is the key to solving many problems in the software engineering field. It is a continuous process to build the practice & for example, in this project, I repeatedly realized each small issues matter and learned to consider all silly little things to consider when troubleshooting. Mindful learning from new problems when the same issues were different in the past also needs to be aware of while troubleshooting. The same issues do not happen for the exact similar cause every time, and to fix my test project, I was missing that and took time to figure out the root cause. So, At first, I started to refactor classes I place from the common project to work on my test project that was required, and it was now working as expected. After a long try found that the JDK version is not suitable for the libraries it used to authenticate and access the database, so I update the JDK version in my eclipse project, and it worked.

```
package eai_test_package;
import com.ericsson.granite.oss.core.taskmanagement.dto.WorkOrder;

public class TestWorkOrderManager {
    public static void main(String[] args) throws Exception {
        System.out.println("TEST MAIN");
        String countryLabel = "NO";
        String bundleName = "CNT";
        String workOrderName = "TeleMtoEAI-exception";

        DBConnectionFactory dbconn = new DBConnectionFactory();
        EaiWorkOrderManager eaiWorkOrderManager = new EaiWorkOrderManager(dbconn);
        String generatedWorkOrder = null;
        generatedWorkOrder = eaiWorkOrderManager.generateWorkOrderNaming(countryLabel, bundleName, workOrderName);
        System.out.println("Tested WO Name : " + generatedWorkOrder);

        WorkOrder testWorkOrder = new WorkOrder();
        testWorkOrder.setName(generatedWorkOrder);
        EaiWorkOrderService eaiWorkOrderService = new EaiWorkOrderService(dbconn);
        eaiWorkOrderService.createWorkOrder(testWorkOrder);
        System.out.println("Test Work Order Created : " + testWorkOrder);

        dbconn.shutdown();
    }
}
```

Figure 16. a workaround of a test in the main class.

As shown in figure 16, I Wrote a main class to test a method, and it tests how the method behaves in the application and creates a new work order instance in the database. The main method is connecting with the database and using services from the common framework I develop, generates work order using the method from common framework services, creates a work order instance in the application, and disconnected from the database and application. I was not even aware of this type of approach to test but learned this workaround. Also that we need to learn and maybe unlearn the old also continually.

Developers also need to think about real-life business use cases and issues related to that, and analysis of all issues of a feature needs to do carefully before development starts. Our Solution architects have designed the feature as the demand of customers and use cases but a feature of telecom network issue far complex and need to understand all

to develop it. For example, to send a notification email, I also need to learn about customers' SMTP servers and our exposed interface, and other application objects that may impact it. There were much learning and not everything I knew before, but I had to spend time figuring out and learning, for example, how to register a plugin, how to integrate the third-party server to EAI, as well as different objects, instance hierarchy, and relation or association to each other. I learned the importance of good analysis and research requirements for integration engineering tasks.

We had a complexity of tasks at the team level & struggling to find the approach to testing one user story, which was a real issue this week. I raised it strongly, even in team meetings, to have a collaborative approach and break down the whole topic into pieces. It was challenging to come up with a different approach than seniors and push to follow that, and it worked. I firmly believed that a team could solve complex problems to do solo; we face complex issues together. Everyone was innovative and keen to contribute to complete the task. We learned that teamwork and collaboration greatly value an agile team of software development projects. A study on the Significance of cooperation in agile software engineering found that by encouraging team members to take risks, the opposite of creativity is fear. It is then necessary to create an environment free from fear of failure: failures are a learning tool. The Extreme Programming methodology includes implicitly central aspects of creative teamwork ( Qureshi, Alshamat & Sabir 2014, 119.).

However, in this project, we have lots of team meetings and sharing issues with others, making the whole team productive where it is different from a typical scrum or agile team. For example, in the debug session and testing session, we cooperate, and it is maybe never a solo work in a group except for some development part where isolated coding is a way to work. When test and developer teams collaborate, I found it a very effective way of working and learning as a new integration engineer, making our project customers more collaborative. It motivated me to take the lead even am a new member, guiding me to learn with the whole team, which was a sort of my objective in this project to fulfill management expectations. As an example of confidently proposing work to do together, I determined it will deliver the best result & optimum solution to test and automation issues.

### **3.6 Observation week 06**

*Monday 14 December 2020*

Many design issues are ongoing with the customer, and there are most user stories from the Task Management epic are flagged as waiting for approval. I am having the plan to

find more information about those user stories even they are flagged and may follow the task to work on as below –

- Check details of the Jira board to extract more information to proceed with flagged issues.
- Follow up Regression test and keep testing and checking performance history.
- Investigate blocker bugs in-depth raised by the customer test team.
- Further, analyze and a session with the developer team about the mail notification from task management.

I start by checking the Jira board and keep notes of flagged issues on the user stories I am involved with and some bugs raised by me or in the area I am working. I already previously checked that Task Management Mail Notification user story was flagged. Before doing anything, I need to analyze more and send an email to our Solution Architect about its queries and the next action plan, so I did. Also, I checked bugs and defects and found a few are flagged and analyzed accordingly for flagging and sending a query email to all relevant people with my opinion and sought the next action plan.

Similarly, I performed a few tests manually in the pre-production environment and checked the history of tests conducted by other test team members to understand the current status and issues. I analyze the performed test and history and send an email to our team's relevant persons as the follow-up of the regression test assigned to me. Besides the regression test, further one particular bug of the data sync feature raised by the customer team and a short discussion of the cause and debugging issues.

Further, I have analyzed and found more information on implementing and sending mail notifications for any data sync use case that failed to sync. Though I have already got some info needed and progress my development breakdown, I will follow them during start development. I have joined the scheduled meeting with other data synchronization developers. The conference call showed my findings and breakdown development flow to update them with the latest info so that they can guide me on how to proceed.

*Tuesday 15 December 2020*

We will have a pre-scheduled customer-specific program meeting in 2nd half of the day, which is important to participate as an integration team member while we are the customer-facing team in a sense. However, in 1st half of today, I will have been working on some task like-

- Have set up a plugin project in eclipse and clone an existing plugin project from a common repository from the project bitbucket.
- Start configuration for the plugin development.
- A session with the test team to further discuss and conclude the test plan.

I started to set up a plugin project in eclipse to prepare for a plugin development environment. I cloned the plugin project from EAI common plugin and got many errors in the project as expected. I have started to configure the plugin project by checking what the missing libraries and other configurations are. I have discussed with my senior developers to understand more about the plugin development environment and know how to fix the missing configurations.

I could not fix it all, but at least proceeding with new plugin development is not a problem now. I had to participate with the test team to finalize further and conclude a test plan. We discussed the feature thoroughly and know that it is complex to plan for a complete test execution within a sequence. I proposed preparing a testing workflow for the test team and workflow design to automate & manual test creation in the discussion.

*Wednesday 16 December 2020*

Our customer is a renowned and leading multi-country mobile operator in the Nordic & Baltics area, and as per their requirement, the project goes by each product increment or PI. The idea also is that other Ericsson OSS or BSS software also whether newly implementing or updating at the same time with our project and they all having a combined package or product increment or a combined release of a version of all product. It is slightly different than traditional agile or scrum, but it suits perfectly all Ericsson ongoing projects as all products are working together for different areas of a mobile operator. Today, the day of all different project demo & it would demonstrate end to end so that audience will also understand how all product's workflow is. Today the whole day is for the demo, and all team members from different projects participate where lead solution architects will present the whole demonstration by turn.

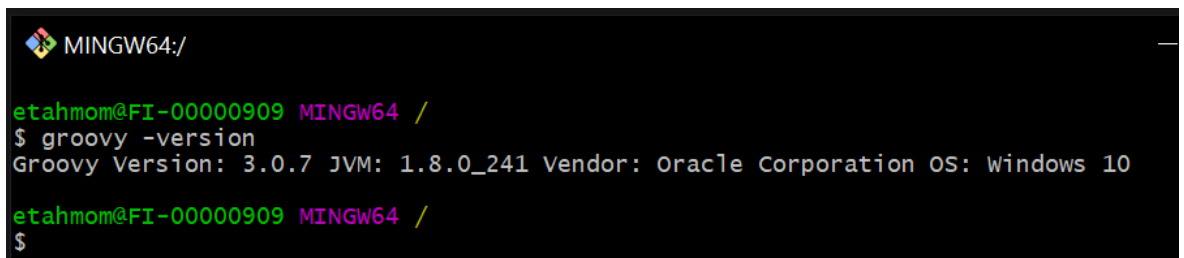
*Thursday 17 December 2020*

After the product increment demo, today's scrum meeting will be longer and assign different tasks to developers from backlog again. As I know, I have tasks for the sprint still left, and I will keep working on that task management mail notification feature. And my today's jobs are similar to below -



- I planned to fix the left part of the eclipse plugin project today and continue with POJO plugin development using the application's Middle-Tier SDK.
- Set up the development environment for groovy plugin development and create a new eclipse project for groovy development

I have little research on issues I have in the eclipse project of POJO plugin development using the application's Middle-Tier SDK. I found some information, and I call one of our senior developers to help me, and I did fix all with his help because when he was checking with me, I also helping him as I now have some idea from research. We together able to fix most of them this time, and there are still a few left, and we understand those cannot be fixed as it has a link with the Network Design Orchestration plugin project inside the project I cloned, but that should not be a problem for us in any way. I started developing the first Java POJO Plugin to send mail notifications.



```

MINGW64:/
etahmom@FI-00000909 MINGW64 /
$ groovy -version
Groovy Version: 3.0.7 JVM: 1.8.0_241 Vendor: Oracle Corporation OS: Windows 10
etahmom@FI-00000909 MINGW64 /
$

```

Figure 17. Groovy and JVM installed version

I have also installed groovy on my local and set up environment variables to make it ready for groovy development as I know I have to convert my POJO Plugin to groovy script using Run-Time SDK of application to deploy, run and test from EAI GUI. As shown in figure 17, it can be verified that Groovy Version 3.0.7 is installed with JVM 1.8, and it is a ready environment for groovy to develop. Also, create an eclipse project for groovy plugin development, and as groovy scripts run using Run-Time SDK, it does not require so much configuration and libraries like the Java POJO plugin project.

*Friday 18 December 2020*

Today is the Ericsson Finland annual party. This time it will go virtual as 2020, the year of pandemic, and the company follows local government laws for any social events, and all employees are under Work From Home (WFH) as well. Still, in the first half of the day, I have planned on the ongoing plugin project of task management mail notification.

I proceed with the POJO plugin and following the previously completed breakdown of development. I am proceeding slowly and at the same time doing some googling and on the internal developer's page to find some information on plugin development. I have achieved the first part and created the required main structure, and fix imports and injection.

After the first part of the POJO plugin, I created a package and a script in my groovy project and tried to achieve the same POJO plugin in the groovy plugin. It seems many groovy syntaxes are similar to Java syntax, but I am literally struggling and not able to proceed with it. I start googling and find information about groovy to learn it better with more understanding. I spent some time following some tutorials on it and found that the annual party has started and I could not resist myself to join my other colleague and celebrate it.

#### *Weekly Analysis of Observation Week 06*

Software Development Life Cycle (SDLC) in this agile project plays a significant role in each sprint. Some developed features take a higher amount of time in the system integration phase and seem more complex to verify the acceptance criteria. Even it comes late in the sprints. A quick and adaptive approach allows the team to achieve the target like this sprint having a critical user story to verify acceptance criteria and assure QA. Perry w. (2006, 47) describes effective testing is conducted throughout the SDLC in which software development activities must occur in sequence, and most of the issue with testing occurs from one of the following causes:

- Failing to define testing objectives
- Testing at the wrong phase in the life cycle
- Using ineffective testing techniques.

So, changing requirements on a small scale also should adapt to the team and focus on customer satisfaction and delivery promptly. Following agile SDLC, the team effort was in place while required. It allowed team members to develop new approaches like I did and performed a leading task to clarify the complexities and set objectives to complete the testing using effective testing techniques.

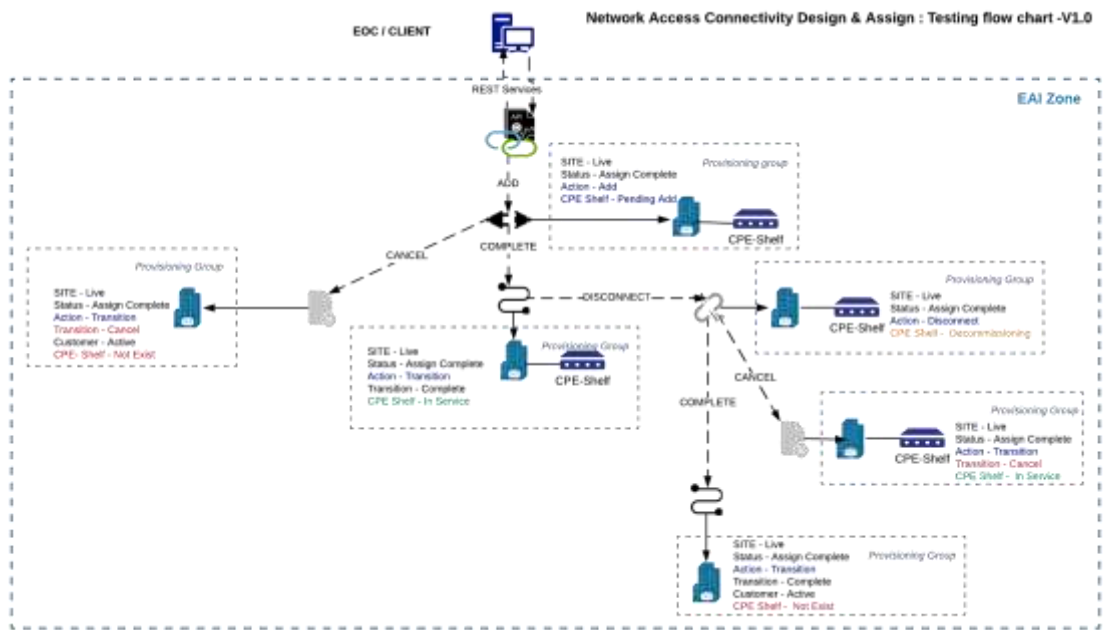


Figure 18. Network Access Connectivity Design & Assign: Testing flow chart -V1.0

In figure 18, a network connectivity design, assign, and Disconnect flow illustrated and demonstrated each step and its behavior with its scope of impact in the network inventory of a particular component. This extra clarifying testing flow chart refers to sequences need to follow for any test automation where the inventory object hierarchy is involved as well as manual test execution, including data creation and creating test scripts. For example, always follow the objects' hierarchy, and the connectivity sequences must follow the order accordingly to ensure the quality of the relevant features of the application.

Developing a plugin for the Ericsson Adaptive Inventory (EAI) application to integrate more functionality is tricky, and there are ways to do that as it offers Out Of Box (OOB) features of use. A plugin provides a customized element to integrate with the application and can be used in many different ways. Two primary way of doing it is using Middle-Tier SDK and Java POJO plugin and using Run-Time SDK and Javascripts, or Groovy scripts plugin and both methods are to do the same thing but in different ways. For example, the POJO plugin can be deployed only with application deployment as JAR files, but on the other hand, a scripted plugin can deploy and run on a running application, and no JAR files need to build or redeploy the application.

Deciding the way of a plugin development sometimes very confusing, and it is like determining tools for your development, which is not easy most of the time. I have developed some parts of the plugin using POJO and figure out that it's troublesome to test pro-

gressed and developed code as part of the development as it required to redeploy application JAR. Later, I decided to explore Run-Time scripted plugin development using groovy, but it requires more skills as a user of this Java-based scripting language. Groovy is an optional, dynamic language with many features for the Java platform inspired by Python, Ruby, and Smalltalk, making them accessible using a Java-like syntax for Java developers. Unlike other alternative languages, it planned to complement but not substitute Java ( König & al. 2015, 5.).

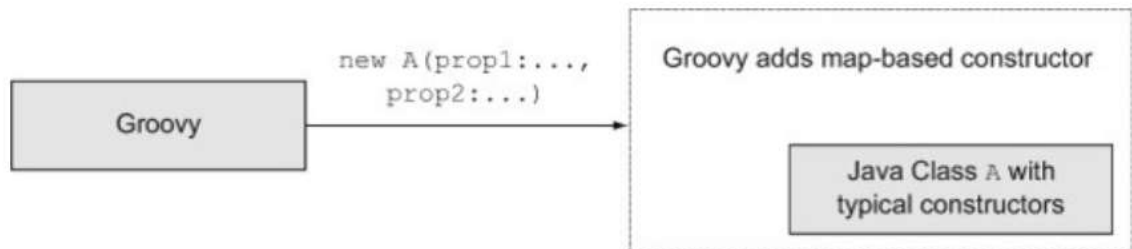


Figure 19. Groovy, a map-based constructor to Java classes (Kousen 2014, 66.)

Groovy scripting is impressive, and it has very co-existing features and relatable to all java developers to integrate faster if required. Learning & working with it for the first time makes a complete sense of suggestions from senior team members to understand it to develop a more flexible way and with versatile options to integrate along the way. According to Kousen (2014,65.) POGOs provide many more opportunities than POJOs. For, e.g., for fixing properties, all POGOs have a map-based constructor that is relatively easy. Moreover, If a class is written in Java, and if it is accessed from Groovy, it will have many similar ways in it. The above figure 19 represents that a Java persistence layer has many classes that map to relational DB tables, and Groovy initiated class use a map-based constructor refers as attributes set using the key-value notation to do and do not require any constructor.

A developer's attitude matters to have that confidence to take the challenge of starting learning new tools or techniques to solve a problem. I am observing my senior developers and learning from them. I positively received all team feedback, which motivates and improves my fast learning attitude, for example, groovy for this project.

Besides learning Groovy, analyzing, researching the topic, and finding information was another exciting part of the week. Searching for a solution to an entirely new problem always fills with a lot of relevant knowledge and even skills like troubleshooting many issues requiring research. Staring from the breakdown of the whole feature from the developer's

point of view and then fixing all issues like creating projects and configure the development environment gives an expression of progress to the individual member and the whole team.

### **3.7 Observation week 07**

*Monday 21 December 2020*

For Christmas and new year, our current sprint got extended till the holiday session ended. So, accordingly, I plan to proceed with existing tasks and follow-up tasks in the backlog and take some which are on priority to complete. Today I can work on the following tasks –

- Conclude test case of ongoing deletion feature of data sync.
- Update previously done common framework as found some missing services for task management.
- Continue to implement the next parts of the interface, TaskEventNotificationPlugin.

Have a joint execution of the test cases with three other test team members to conclude the manual test of previously analyzed & discussed deletion features. The designed testing flow chart helped create data and execute tests sequentially for many network equipment. Even it is a feature of the data sync module of the application. It is also linked with the network design module because both modules work for network creation and impact network inventory. After understanding the network component's workflow, we managed to execute all of our test cases. I have actively participated in the execution phase, over-viewed other testers' conducting, and determined that the feature meets acceptance criteria.

I received feedback from another senior developer of our team about the previously developed common task management framework for some missing services. I have rechecked and found that one more service should include the framework and other task management features. So, I have refactored my code and added a few more methods to use those methods from other application components.

Today I have further developed the next part of the task management notification plugin to access required data from objects like Work Order, Queue, and users data to sent notification email to users email. I think it is good progress till now where we have the recipient's email address list according to the business logic of the failed task of data sync from

the legacy system. Until now, when the plugin is invoked, it will have TaskEventContext object as the parameter in its notify() method and used previously developed framework services to access the created task instance and its Id. With the created task instance Id, this plugin can access the task object comments that contain its cause of failure and relevant objects like Queue objects and its other attributes like users associated with that particular queue.

*Tuesday 22 December 2020*

As I already proceeded with developing the intended plugin of mail notification and discussed it with one of our solution architects to understand what I analyzed regarding the implementation. He suggested keeping it in progress and come up with a POC type of version so that the development lead and all designers can review if any amendment is required. I will keep working on that topic with the below tasks-

- Review & propose POC of mail notification of task management integration.
- Proceed with plugin development and handle relevant issues like a work order, queue, tasks, and users.
- Learn more about Groovy scripting to develop & test with Run-Time SDK of the application.

I have a review session of the partially developed task management notification plugin with the Developer lead and the Solution architect. Along with the review, I also proposed creating a kind of POC first as we don't have the customer SMTP server's required information. It was an excellent workaround to proceed with development and complete the task according to all of us. I have shown my logic and plan how to do it behind this idea of POC first before the full integration of the customer SMTP server.

After the review and discussion, the proposed POC idea of the task management notification feature allows progress with the development task. I continue to proceed with coding until an email message is generated with each failed data sync job creates a task to handle manually by users associated with the queue where the task will be placed.

In the 2nd half of the day, I concentrate on learning groovy development more so that I can convert my POJO code to groovy and will be able to test it by deploying with Run-Time SDK of application and the plugin will be invoked and work as usual. To achieve that, I need to learn more about groovy scripting and how I can convert it from POJO to groovy or vice versa.

Wednesday 23 December 2020

Before the season holidays and Christmas time, I am thinking of proceeding with tasks related to the task management notification feature and testing some parts in EAI as the groovy script can test on the go without deploying it on the server POJO. To fulfill my today's objectives, I will concentrate on the following task today to accomplish –

- Start writing the groovy script based on the developed Java POJO plugin.
- Register the plugin to the octm: adapter in the EAI plugin section.
- Create an extension point and set the parameters of the plugin in EAI.
- Create system interface instance and configure EAI for a plugin deployment.

I have scripted the groovy plugin first in my eclipse groovy project following my Java POJO plugin of the same feature, and later, I will import and save it in EAI script zone as a Run-Time plugin. As I have learned basic groovy scripting, I am converting my POJO code to groovy by modifying syntaxes as groovy syntaxes are very similar to Java and easily convertible. My main intention is to test it using Run-Time SDK on the go by importing the script to the EAI script zone. So, I can test and develop each part of the feature simultaneously, which I broke down previously. To achieve that testing results for development further, I am placing `System.out.println()` method to log my variables and data fetched from objects or other services in the plugin and print in the console to make it easy debug in the development phase.

Once I have started my groovy plugin script writing, I have also done registering the plugin in the plugin section of the Admin Zone UI to the -

*octm:adapter: TaskManagementEventNotification\_ExternalSystem1* extension point.

We have standard practices for EAI developers to register a plugin using either the Admin UI Plugins screen or the `custom.plugins.xml` file. I used Admin Zone UI to register the plugin and made the above extension point on which it will be invoked. Also created a new Task Management System Interface object in EAI as required. Assigned previously created extension point in the External System Interface object and associated the External System Interface object to the pre-configured Queue instance for which we want task notifications sent.

The whole point of this configuration is to allow EAI to integrate a third-party system, and the plugin will find all users associated with the queue and their email address to send

email notifications to all of them about a failed task of the data sync module of the application. Task Management module will create a task under a Work Order according to the common task management framework, and all work order will be associated with a pre-configured queue. Each configured queue as per customer design will be assigned to one or more users who will email the task created regarding the failed data Sync job. So, now, all objects & instances configured as required, each task creation will invoke the plugin, and the development of each part of the plugin will be manageable and testable.

*Thursday 24 December 2020*

A national holiday for Christmas Eve

*Friday 25 December 2020*

A national holiday for Christmas Day

*Weekly Analysis of Observation Week 07*

To Integrate and develop a functional Task Management module, I have planned and get approval for a proof of concept (POC) approach. The POC approach's main reason is that we have a dependency on customer's SMTP server information to integrate with EAI. It was an issue to handle, and I thought we might develop the integration plugin instead of flagging and pause for the SMTP dependency. So the idea I came up to propose is to use our Ericsson SMTP server, and the SMTP server credentials will be passed as admin properties to EAI. We will develop additional admin properties for the SMTP server to configure manually anytime later. The concept also required another UI for admin properties that has to develop, but it extends more functionalities of the task management module and quality of the feature.

Most team members appreciated the POC concept because it brought to light that the extended functionalities of admin properties to configure SMTP server credentials may add more value to the feature. An SMTP server might change from time to time if the customer wants, so flexibility to configure any third-party server by user added to the feature. According to our solution architect, it can be a practice to follow in any similar situation where we pause a feature development for some dependency from the customer end.

So, the user story acceptance criteria are a configurable SMTP server using the admin zone UI of the application and a plugin that will send a notification to the users when a new task is created due to a failure or data sync job between EAI and the legacy system. Figure 20 shows a notify method declared in a groovy script and has TaskEventContext



parameter in which the TaskEventType will be passed to use by the plugin. The plugin will also use a previously developed common framework for task management; for example, “EaiTaskService” and “EaiQueueService” will be injected and used. The groovy script development was similar to Java, and I have learned many new areas of the development, like using another framework by injecting it into functions or methods.

```
def void notify(TaskEventContext tc) throws Exception {  
  
    System.out.println(" MAIL NOTIFY CALLED :: ");  
    // TaskEventContext - extract required data  
    System.out.println(" TC 1 :: " + tc);  
  
    EaiTaskService eaiTaskService = null;  
    EaiQueueService eaiQueueService = null;  
    ManagedObjectFetchSpec taskFetchspec = null;  
    ManagedObjectFetchSpec queueFetchspec = null;  
    Task taskObj = null;  
  
}
```

Figure 20. notify() function declaration in groovy script

Another area of the week was to configure required objects and instances to register a plugin so that plugin will be invoked on the application's intended event. Design documentation by the customer and requirements analysis of the feature made a clear view of how we should consider development when different application components like Queue, users, Work Order, and task are involved. Table 1 analyzed the task events in a different state of task management and our intention to capture “ADDED” type events. Whenever a task event “ADDED” to a queue interface in the next phase of task creation, our plugin will be invoked and do its notification sending task as per business logic defined by the customer.

Table 1. Different Task Notification Events

<b>Task Notification</b>	<b>Interface Associated to Work Order</b>	<b>Interface Associated to Queue</b>
Task created	<i>TaskEventType.ADDED</i>	None
Task Associated to a Queue	<i>TaskEventType.STATUS CHANGED</i>	<i>TaskEventType.ADDED</i>
Task association to Queue removed	<i>TaskEventType.STATUS CHANGED</i>	<i>TaskEventType.REMOVED</i>
Task Status Changes	<i>TaskEventType.STATUS CHANGED</i>	<i>TaskEventType.STATUS CHANGED</i>
Task deleted	<i>TaskEventType.REMOVED</i>	<i>TaskEventType.REMOVED</i>

Understanding the application feature of integrating the plugin to customize and extend the EAI feature was a study and learning by doing with the team again after the introductory training workshop. It was another challenge to register a plugin in EAI and configure the right extension point following Ericsson's internal libraries' documentation. It is never perfect for the first time doing something for software engineering, and there is room to develop in this field. For example, it was not working for the first time I configure and test the extension point. A troubleshooting session revealed that it needs to configure the extension point parameters to inject those frameworks and other plugin services to be used, and the plugin will work accordingly.

Also, the approach to use Run-Time SDK was mainly for the test of each part of the development and debugged at the development phase. So, I tried to use print methods in each part of the plugin and test it by creating a task and observing how my plugin behaves with the "ADDED" task at the queue interface (*TaskEventType.ADDED* events phase shows in table 1). Here to mention that this groovy script will be finally converted to Java POJO and will be deployed with a new build of the application. It can usually work as a plugin, but this current practice makes complete sense because it will require many different network components to work with. And each step and unit coded can verify by checking the debug log in the application server. It was exciting for the first time to develop something for a running application and test it on the go and observe after each unit code behavior.

I have shown the concept which I am planning to implement by a groovy scripted plugin using Run-Time SDK of EAI so that the Developer Lead and the Solution Architect will understand the whole idea, and after that, they agreed with this proof of concept style implementation though we will finally implement it with POJO plugin development using Middle-Tier SDK of EAI. Proof of the concept is an important stage in software development to

determine the project's value even before the development begins (Musienko 24 February 2021). I have practiced in this project on a small scale to learn the importance of POC in adding a new feature or module to the application.

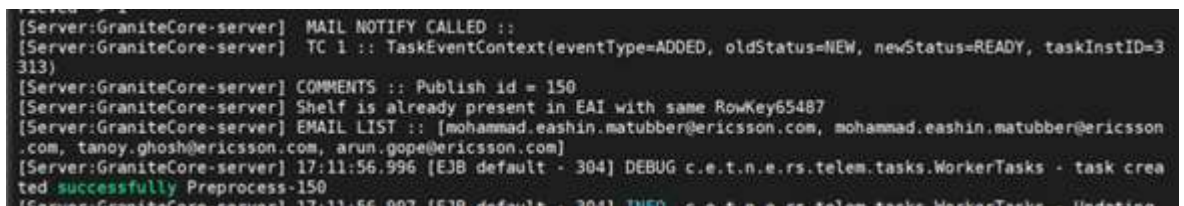
### 3.8 Observation week 08

*Monday 28 December 2020*

The Christmas season is ongoing, and our team members and customer representatives are still on holiday, but some of us keep working. For example, I will keep working on the following topics today –

- Groovy coding to further implement Task Notification Plugin in Run-Time SDK
- Test implemented plugin and debug issues before transfer to Middle-Tier SDK
- Research and analyze more how to achieve the intended feature of sending a task notification event.

I started coding for my scripted plugin using Groovy after the scrum meeting and proceeding first as it has a similar syntax to Java. It was the idea to move with a scripted plugin so that we are able to test it on the go and develop our POJO plugin without bugs and the right way. To achieve our goal, I have coded and tested to get all the data I need to get from all associate objects, for example, a Task object and the Queue object associated with the task. Also, all User object's email address attributes are associated with the Queue object.



```
[Server:GraniteCore-server] MAIL NOTIFY CALLED ::  
[Server:GraniteCore-server] TC 1 :: TaskEventContext(eventType=ADDED, oldStatus=NEW, newStatus=READY, taskInstID=3313)  
[Server:GraniteCore-server] COMMENTS :: Publish id = 150  
[Server:GraniteCore-server] Shelf is already present in EAI with same RowKey65487  
[Server:GraniteCore-server] EMAIL LIST :: [mohammad.eashin.matubber@ericsson.com, mohammad.eashin.matubber@ericsson.com, tanoy.ghosh@ericsson.com, arun.gope@ericsson.com]  
[Server:GraniteCore-server] 17:11:56.996 [EJB default - 304] DEBUG c.e.t.n.e.r.s.telem.tasks.WorkerTasks - task created successfully Preprocess-150  
[Server:GraniteCore-server] 17:11:56.997 [EJB default - 304] INFO c.e.t.n.e.r.s.telem.tasks.WorkerTasks - Updating
```

Figure 21. Server log of scripted plugin test of task creation to notify

Figure 21 observed from the server log that the initially developed script got all attributes data from different associated objects required to complete the mail notification feature's implementation.

I also started to research the functionalities we need to implement and relevant other plugins to work together for the mail notifications. I started studying one of the EAI plugins

called "Job Update Plugin," which is responsible for update the database upon any updates of resource synchronization job and make the status accordingly in the database column. I found that this plugin may trigger the notification plugin as it is updating job status so that it will trigger it upon a fail status.

*Tuesday 29 December 2020*

After handling the internal part of task management notification as tested, the developed plugin can extract & fetch required data and objects and need to address the next part of sending email to the SMTP server. So, today planning to proceed further to develop the next part of the feature and tried to complete the following tasks-

- Research about Java Email API and find a way to use it for mail notification.
- Study Ericsson online courses of 2020 as part of the organization's skill development.
- Progress in developing mail notification plugin further.

I started to research libraries and APIs of email sending of Java and different implementations of similar features using Java APIs. I also started to read Java Mail API documentation to understand offered packages and how to use them for our plugin project. In our implementation, we may need libraries of Javax mail API's message, session, exception, and I started to search for other similar examples of implementations for references.

I got an invitation to complete some internal online courses to learn about the organization's policies and practices. Courses are Data Privacy, Security-Awareness, Anti-corruption, and EGMS awareness( Ericsson Group Management System). I proceeded with studies and some tests on the go and completed a few, and the rest I decided to resume later to complete. These courses are mandatory to pass as an employee and before the year ending to have the necessary knowledge about my organization.

In the second part of the day, I started coding for the rest of the plugin and implementing mail notification sending using java mail API libraries. It is not done yet, but I tested that it can send emails to all addresses in my email list, which was my goal to achieve today. It will work with any SMTP server credentials to send emails from our application as per our implemented business use case.

*Wednesday 30 December 2020*

From the latest development of the Mail Notification Plugin's user story, there is another requirement to develop admin properties for users to configure, as discussed earlier. I also have some tests of templates modeled for EAI, so today's task list can be something like below-

- Further analysis & customization decision of admin properties for SMTP Server  
Test templates
- Code for admin properties for the SMTP server configurations in EAI
- Test templates deployed on the Pre-production environment

I research the application and its admin configuration properties for other features to find out more about its implementation. It is the best efficient way of implementing it for my mail notification feature. I also considered that these configuration properties are private and must have some sort of privacy from the system's general user. I found that there are more admin configuration properties, and I can extend one more similar way for the mail notification feature. I have also done the development part for a set of admin properties and start coding for it as it was easy to follow the previous implementation.

I developed for all required properties value of SMTP server configurations, for example, host, user, password, authentication, security, the sender's email, and plugin enable or disable picklist. After modification and required development, I also deployed it to the development server and have tested it from the GUI of the application.

*Thursday 31 December 2020*

As it has already been developed to send an email notification, I will review it for the final time with any of my senior team members and complete it for deployment.

I will deploy the feature in the development environment and test it thoroughly. So my task list for today will be similar as following-

- Done with admin properties MT project and any refactoring and formatting for the final time
- Build Jar & deploy MT project in Dev server to test it before release.
- Test templates and check acceptance criteria to release.

I have started with rechecking and formatting the mail notification plugin as it is completed. After that, also recheck and format code for admin objects. I have arranged a time

slot to review with our solution architect and reviewed all parts thoroughly, and we decided to make some changes. After the review session, I have refactored my code as we discussed. So, now Mail Notification from Task Management features & functionalities of EAI is ready to deploy.

Next, I have prepared my MT project and placed it in the development server to deploy in the development environment via the Bamboo pipeline. So, our pre-configured pipeline builds the deployment jar and deploys it. In the first run, it failed to build as it is a Gradle project, and I forgot to add all required libraries in gradle.build file. After adding all required libraries in that file, the deployment pipeline successfully builds and deployed the plugin. I have done some end-to-end tests and found it is working as expected, and every time there is a new task in Task Management, it is sending a mail to the respective user's email addresses.

In the later part of the day, I have started to test modeled templates for EAI. I have started to create shelf instances using templates in the EAI Inventory zone and manually check that it has created the intended shelf as per specifications.

*Friday 01 January 2021*

*OFF day*

*Weekly Analysis of Observation Week 08*

To develop the mail notification plugin, I also need to learn how to use java mail API in a java application. Java Mail API offers a framework for email services using Java programming language and can send and receive emails via SMTP, IMAP, POP, etc. The JavaMail API follows the abstract factory design pattern, allowing writing code based on the abstract superclasses without worrying about the lower-level details. Service providers implement particular protocols. A service provider is a group of concrete subclasses of the abstract JavaMail API classes that specialize the general API to a specific protocol and mail format. (Harold 2013, 2).

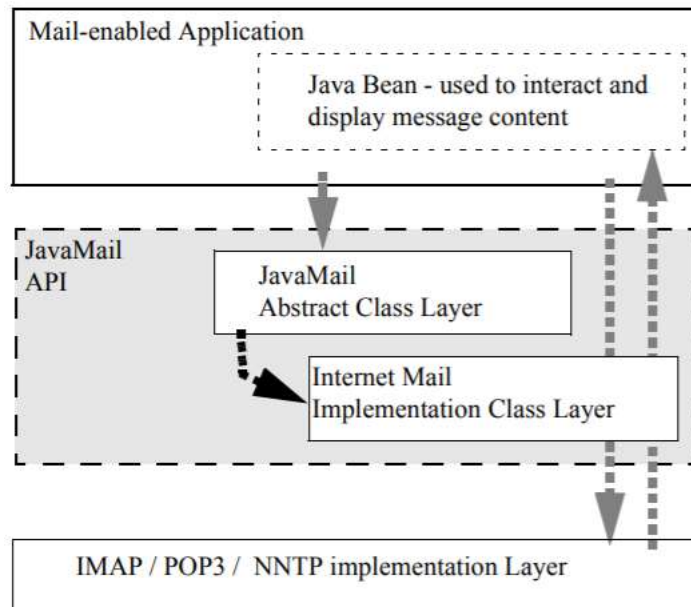


Figure 22. JavaMail Layered Architecture (JavaMail™ API Design Specification, 6)

JavaMail uses the JavaBeans Activation Framework (JAF) to encapsulate message data and handle commands intended to interact with that data. Interaction with message data should take place via JAF-aware JavaBeans, which are not provided by the JavaMail API. (JavaMail™ API Design Specification, 6). In figure 22 illustrated the JavaMail architecture and layer where Abstract Layer declares classes, interfaces, and abstract methods intended to support mail handling. The implementation layer is part of the abstract layer using internet standard – RFC822 and MIME.

```

for (String to : emailList) {
    try(SMTPTransport transport = (SMTPTransport) session.getTransport("smtp")) {
        Message message = new MimeMessage(session);
        message.setFrom(new InternetAddress(from));
        message.setRecipients(Message.RecipientType.TO, InternetAddress.parse(to));
        message.setSubject(emailSubject);
        message.setContent(emailBody, "text/html; charset=utf-8");
        transport.connect(host, username, password);
        transport.sendMessage(message, message.getAllRecipients());
        Logger.debug("Sent message successfully...");
    } catch (MessagingException e) {
        Logger.error(e.getMessage(), e);
    }
}

```

Figure 23. using javax.mail library implemented

In figure 23, we can see the implementation of send email, and it is content the session class which is a top-level class of the API. It creates mail sending sessions with configura-

tion settings, properties, and authentication. It also uses the message class and its subclass Mime Message, which is crafting the email's details as a Multipurpose Internet Mail Extension or MIME format.

The Mail Notification feature from Task Management of the EAI is also required to understand the SMTP server and client understanding to send emails via SMTP. Simple Mail Transfer Protocol we used as our customer's suggestion and says it is responsible for transferring emails from one user to another. There is no direct connection between them, but the users, the path is often broken up into hops, and each protocol session carries the message one hop further along the path to the recipient's post office. Also, a mail message is typically sent to a mail relay that sends it on to a number of intermediate relays until it arrives at its destination, usually a post office. (Rhoton 2000, 107).

Our infrastructure team takes care of all deployment and its pipeline. Still, before release, developers also deploy it in the development environment first, and in this project, we are using Gradle as the build tool to build the plugin project. And to deploy it, I also need to have a basic understanding of the Gradle build phase and build lifecycle. I have studied documentation and learned the main three build phases: initialization, configuration, and execution. According to Gradle's official documentation, Gradle determines the projects of the build during the initialization phase and creates project instances. During the configuration phase, project objects are configured, and in the execution phase, Gradle determines the subset of the tasks created and configured during the configuration phase to be executed (Gradle 6.8.3). Learned some basic command of Gradle and how to use commands to Gradle build.

In the development server, after building the project jar using Gradle, I also learned how to use a pre-created CI-CD pipeline of bamboo Atlassian tools for CI-CD. To deploy the Mail Notification plugin project in the development server, I have also researched, studied, and learned the bamboo CI server's basics and workflow to use it as suggested by our Developer Lead. I understand how different workflows like plans, jobs, and tasks of a pipeline works. For example, how in plans stages groups all jobs sequentially for execution, how in jobs step collects task requirements to map the capabilities, and in tasks step how it executes the commands pre-configured to the system. Finally, I have used a pre-configured bamboo pipeline and observed the pipeline logs from bamboo till the successful deployment.

In development and other tasks, I would achieve the same results in many ways, but as the integration team's trainee, I always try to have suggestions from my senior developers and follow previous examples. It was quite challenging to learn tools like bamboo and



Gradle in a concise period. I was not aware that I am learning it for understanding purpose only, which makes me stressed initially, and in the latter part, it was clear that it is a learning and there are everything preconfigured tools in place to use for developers, so it was another quick learning things throughout the week.

### **3.9 Observation week 09**

*Monday 04 January 2021*

Sickness leave

*Tuesday 05 January 2021*

Last week, we deployed the Mail notification feature in the development environment and ready to release it as all user story requirements and acceptance criteria were already developed. To release the feature in the system integration environment, I have to create release notes for the feature and having the plan to work on the following tasks -

- Create release notes for Mail Notification from the Task Management feature.
- Configure system integration environment to be ready to release Mail Notification feature.
- Have a meeting with the test team and debug if there are any bugs found in the mail notification.

After the scrum meeting, I started drafting release notes of the user story of Mail Notification on the confluence page for the current sprint. After the middle-tier deployment, some preparation is needed to have the Task Management system ready for email notifications. I have documented the complete configuration process to activate the application feature so everyone who deploys the latest EAI version will know the process without any confusion. After release notes documentation, I have completed the Mail notification feature's required configuration in the system integration environment. After the configuration, this environment is ready to deploy the developed feature and will work as expected, so we can say here that the feature has dependencies of some initial configuration in EAI.

In the 2nd half, I have a scheduled meeting with the test team to explain the Mail Notification feature's full functionalities and discuss issues like bugs they may already observing. I participated in the discussion and found that they already have many doubts, and I have

answered with an explanation. We have tested together and verified the feature's performance issues and found email message format issues too. I have logged a defect in the Jira bugs board to track the issue further and fix it with the optimization of the feature.

*Wednesday 06 January 2021*

National holiday

*Thursday 07 January 2021*

After the Mail Notification feature is integrated and deployed, there is a new development task of resource synchronization assigned to me. I will also follow up and optimize the Mail Notification feature to fix the defect raised previously. So, today I have the plan to work on the following tasks –

- Analysis of the reverse synchronization between the legacy system and EA
- Move all project plugin from EAI common to the project-specific repository for better maintainability.
- Debug, fix the defect and optimize the mail message formatting issue.

First of all, I started analyzing the user story for making reverse synchronization from EAI to the legacy system currently customer is using for their network inventory management. There is a mismatch in the port name between the legacy system and EAI. Therefore, it is required to update the Port label attribute with the EAI Port Access ID and DNS name with the Shelf hostName attribute in the legacy system. Ports directly on the Shelf or can have the Port Access ID (Interface Name) from EAI through the EAI to the legacy system interface using REST API. Also, Ports on Card/Sub-Card can be updated with PortAccessID (Interface Name) from EAI through the EAI to the legacy system interface using REST API. This mechanism is also will update the hostname besides updating Port Access Ids. It will be only applicable for the new core network equipment creation. The legacy system will update all the existing equipment using the post-dry run report published by Data Migration.

I have also moved all plugin projects code to a country-specific repository for more simplicity and better maintainabilities. I moved all java classes from the common plugin project to the country-specific project repo one by one. So, it was required to fix the current project where all classes will have resided now. It was a troublesome task to fix the project and replacing all the necessary jars and libraries for the new repo. So finally, I moved all country-specific plugins to repositories.

I have debugged one defect raised previously by our internal test team. The defect is about the Mail Notification and the email message of the service fulfillment Queue. The expected behavior is email content will have a hyperlink and send it via the Mail Notification feature, but the link sent to recipients was not the exact link it should send. I understood from the behavior and test using a different type of text format, and type "HTML/text" works fine for the expected behavior. I change the Mail Notification plugin code, push changed Java class to the bitbucket repository, and redeploy it to the development server for testing.

*Friday 08 January 2021*

I got another requirement to further optimize the feature from the admin configuration perspective. The main idea is that they want more flexibility and extendability of the feature with the current implementation by adding a component of configurable email templates for different Queues in EAI. So I will have to work on this, analyzing how to fulfill the acceptance criteria and breakdown the process to discuss with the solution architect for finalizing.

I start analyzing the mail templates' requirements and optimizing the Mail Notification feature along with the mail template feature. I start breaking down the whole idea to make a Next-Generation Object (NGO) for Queue base mail templates. So that NGO will be in the inventory zone in EAI and will be configurable by the admin user as required. So admin can create mail templates for each Queue object with different content. We can use Java Unified Expression Language to parse some expressions and fill in place holder in the templates.

I have scheduled a session with the solution architect for further discussion and present the idea I am thinking about the Mail Notification's extended requirements. I have explained the whole concept and process of the proposed extended feature and concluded with some points like - meaningful placeholder in the mail templates, a mail icon for the mail template object, and possible optimization of the previous implementation. We also discussed why we decided to use JUEL for the template and the possibility of optimization by breaking the plugin's responsibility. So in the new optimized implementation of the Mail Notification, a few different classes are responsible for composing, utility, mail sending, etc.

Later part of the day, I have started to research and find practicalities about JUEL. I spent some time studying and understand the documentation from the official library of JUEL.

After some understanding, I joined pair programming sessions with another senior developer to develop our analyzed reverse synchronization user story. We have created the project and developed the feature which will fetch equipment data by its name as a parameter and generate a JSON to send to the legacy system server to update the data for the same equipment in the system. We have also declared the main method to test the implementation's functionalities and remove the code's main method. After implementing level one port access ids update or reverse synchronization, we deployed it in the development environment to test further that reverse synchronization works and updates the legacy system's data fetched from EAI.

### *Weekly Analysis of Observation Week 09*

To release a feature this week, I have made a release note for the first time and realize a release note plays a significant role in software development. I have researched and understand the release notes' characteristics and how a developer should write a release note. It was quite tricky for the first time to write a release note for a feature where we are delivering it to a regional telecom leader. This note is something they will use to evaluate and understand and configure the feature. Several studies found that release notes must deliver in each sprint in agile software development practices. As release notes comprise the information related to new features, changes, and issues spanning the whole development life cycle, it is a vital artifact to development. The quality of release notes depends, among others, on its producers (Bi, T., Xia, X., Lo, D., Grundy, J. & Zimmermann, T. 2020, 8).

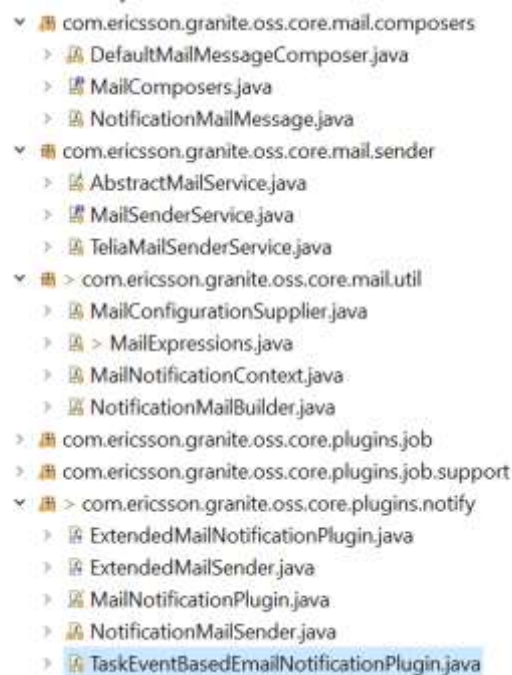


Figure 24. Maven build locally with warnings

Moving a plugin project from one repository to another, for example, in our case, from a common application repository to a country-specific repository, was a troublesome task to do. I faced challenges like fixing the project with proper libraries and resolving all issues arising from the moving repositories. Many dependency issues need to resolve, and from this week's experience, I must say it never goes by the book but requires trying different ways to resolve them sometimes. In figure 24, we can see that even resolving all mandatory issues still having many warnings after moving some projects to a new repo. So after having a review with a senior developer, I am satisfied with those warnings without resolved and maybe not mandatory to resolve all warnings. Our deployment pipeline uses Gradle, and locally we developers using maven for build and development, so some are handled by Gradle build in the deployment phase.

Understanding REST API and its behavior was another topic to research in my task. To develop a reverse synchronization feature, we have analyzed the issue to implement. We need to use REST APIs to fetch data from one system and update it in another. To conclude the issue and decide the development process, I have also researched RESTful services and their behavior. The difficult part is that we don't have documentation of the legacy system's API to update resources. To find the way, we did lots of tests and tried to find the API's behavior to know how we should send POST requests to the endpoint URL. According to a study, developers face difficulties like using the correct data types, correct data formats, and required HTTP headers and request body while performing API tasks using API documentation that lacks usage examples (Sohan, S. M., Maurer, F., Anslow, C., & Robillard, M. P. 2017, 60).

From one debugging task of this week, I found that there is always a better way to develop things or room to develop anything further. The defect was raised from a URL sent via an email message, and it is not behaving as expected. I researched and found that the better way of sending an email is HTML format content. I tried to understand the underlying process of the MIME message of Java Mail API and its set content method where we indicate the content type text/HTML. So, a more comprehensive use case of sending email also studied, for example, creating an email with an image or other attachments and not implemented any but kept aside for future use if required. '

We used JUEL (Java Unified Expression Language) to implement mail templates for our mail notification feature, and it was a new topic for me to do. I have struggled to find the way to use it for expression and manipulate the template placeholders with data fetched to

send as email content. I learned and used it for the development; I followed JUEL documentation and added JUEL JAR files to classpath and import classes. JUEL has three main parts those are –

1. Factory and context
2. Functions and Variables, and
3. Parse and evaluate

It uses `javax.el.ExpressionFactory` to create a thread-safe unlimited number of expressions of various types with a caching interface that allows applications to use their own caching mechanism. Value expression parses an expression string, builds an abstract syntax tree, and binds functions and variables using the context's mappers.

### **3.10 Observation week 10**

*Monday 11 January 2021*

We have developed and deployed the reverse synchronization and mail template in the previous week, and I have to start testing those this week simultaneously. Today I will focus on testing in the development server and begin analyzing new user stories and tasks. So the following would be similar tasks of today –

- Test previously deployed user stories to deploy in the System Integration environment.
- Start analyzing new user stories of NGO creation assigned to me.
- Start initial development of Firewall Cluster NGO.

I have started to test reverse synchronization to test that implemented feature is updating legacy system data as expected. I found it challenging to test, especially newly implemented validation that required proper input to synchronize first. Then, the synchronization process will reverse synchronize to the legacy system and its data as EAI system data.

I also have tested Mail NGO in the development environment to be deployed in SIT and proceed for demo. I found that it required permission for the Mail template NGO to use with the task management module for mail notification. I have configured permission so that another admin user can create a mail template and able to have an email notification template ready for different use cases.

I have analyzed new NGO creation user stories using design documentation from the confluence. It was not fully clear from documentations, and I have scheduled a meeting with our Solution Architect. In a call, it was a very explanatory discussion, and we clarify all issues. I started setting up my NGO development project using visual studio code IDE as it is a suitable IDE for XML development. Our EAI Next Generation Object model developed using XML. Later part of the day, I have developed a firewall cluster object model definition using Middle-Tier SDK documentations.

*Tuesday 12 January 2021*

Since yesterday, I have proceeded with the object model development, and today I plan to continue developing the Firewall Cluster object and possibly start the next NGO development. Tasks list of today might be like below –

- Develop Firewall Cluster NGO attribute definitions, Association definitions, and uniqueness definitions.
- Start development of VRRP (Virtual Router Redundancy Protocol) object model.

After the Scrum call, I started to develop the Firewall Cluster object model attributes definitions. I found that besides other attributes, it has a picklist that reuses previously develop another object's same picklist simply by defining picklist definition attribute Id to the intended component Id. I continued development tasks for other object model definitions like association and uniqueness. I defined that source object as Firewall Cluster, and target objects are many different other network objects in association definition. In uniqueness definition, it is defined that the object name will always be unique for the same object.

I also started to develop another new object model VRRP and started as usual from the object model definition. I have added object model definition in the Middle-Tier object model XML file. I have read the documentation on developing an NGO using Middle-Tier SDK of EAI and understand that defining object model definition is tricky to create a successful object in EAI. As I have done object model definition next parts of this, I can proceed smoothly.

*Wednesday 13 January 2021*

Yesterday I did develop the object model definition of VRRP. I may need to refactor that and start the development of the following parts of the object model, and my today's task list will look like below-

- Develop other definitions of the object like attributes, associations, and uniqueness.
- Done required refactoring if needed and validate XML to deploy.

I started working on the rest part of the VRRP object and developed attributes definition. I defined attributes like object name, description, UUID, and a picklist called status. For the status picklist, reused another component's picklist as both are the same. So referring to that component id in the attribute definition will generate a picklist for this object too. VRRP object also has a few associations to IP Addresses and the Logical Interface (LI) network. After developing the association definition, I have also done the name uniqueness definition for the object. The object's name must always be unique, and it won't allow users to create duplicate objects with the same name. I have achieved that by referring to the name attribute id of the object, which will have this uniqueness check.

After I have done both intended object modeling, I allocated time to refactor and review with a senior developer. I approached a senior developer and asked him to have a review of my code with me so that we both can identify if there are any issues. We have spent enough time going through all my code and added a few missing things; for example, the object should have lockability defined while modeling it at this stage. We have also validated the developed XML file with our Gradle build tools, found some invalid XML, and corrected them to make it 100% validated XML. Finally, it is ready to deploy as an MT (Middle-Tier) project in the development environment to test.

*Thursday 14 January 2021*

The new two Next-Generation Object model of EAI has developed and now need to deploy these and test thoroughly to find defects. So today, I will be focused on tasks like below-

- Move all changes in the development server and run the deployment pipeline
- Follow deployment logs while running it for possible errors (if any).



- Unit test manually to find defects, raise it in Jira, and create new features.

I have accessed the development server via WinSCP and moved that XML file in the project resources directory, generating an object in the GUI of EAI after a successful deployment. There are about 6000 lines of XML code in the object modeling XML file, which contains all new objects of EAI; I have replaced that file with the latest development. After carefully checking other directories, I again validate the XML in the development server and test the build using Gradle builds tool before final deployment. As all tests and validation passed, I have run the deployment pipeline, which will rebuild the project and deploy it to the development server.

I start following the deployment log while running the deployment as instructed by our Development Lead to fix any errors quickly and made the server up and running as soon as possible. It showed an estimated 40 minutes to complete deployment, and a log file was generated by the pipeline continuously for the process. Luckily there were no errors or any issues, and it deployed successfully.

I also have tested each unit of the deployed objects and tried to find possible defects. Found an issue with UUID generation of the objects. Firewall cluster objects are not required to generate UUID, but VRRP has the requirement to generate UUID while creating a new instance of the object. It is precisely not a defect but a requirement to fulfill. Found a defect that some field of the object should be "Read Only" values, but they are Read & Writable in the current implementation. So, I raised a defect in the bug board to track the defect and created a task in the Jira task in the respective user story to develop a plugin for UUID generation.

*Friday 15 January 2021*

There are issues like a new requirement to fulfill and debug a defect to meet the user story's acceptance criteria to release in the next Sprint demo. Today I have the plan to complete tasks like-

- Develop a simple plugin to generate the UUID of the VRRP object.
- Fix the defect raised in the development phase for Read-Only values.
- Start learning Camunda (BPM modeler) for the next user story development.

I started the development of a new plugin called `VrrpUuidPlugin.java`, a new java class in the common plugin project to generate UUID for VRRP object during creation. I injected

all dependency of Middle-Tier SDK and required service classes by importing and creating new injected instances. I developed the plugin and wrote code to check it will only work for country-specific projects, not for all countries. The pre-creation stage will generate a random UUID for the object, which will achieve the objective of the plugin. I have also changed the required code to make some fields read-only so that users can only read them but can't edit or modify it.

After the development tasks, I started learning Camunda for my next network creation tasks. Camunda is an automation tool for business process automation and end-end orchestration. It is decided to use this tool for network creation automation of EAI. I started from its documentation and EAI internal guide so that I understand and will be able to contribute to the team. I also asked for help from senior developers on which approach is better, particularly for learning the Camunda tool though there is a lot of good documentation.

#### *Weekly Analysis of Observation Week 10*

Most of the work I have done this week was all new development and relevant learning. Besides that, there was some research for analysis and information for work and a new tool like Camunda. For developing a new NGO, I have studied the internal guide for NGO development. This documentation of the EAI study made me even more knowledgeable about this type of integration development of the software.

```
<!--START: Uniqueness Definition for VRRP -->
<uniquenessDefinition>
  <id>vrrpName_uniqueness</id>
  <name>VRRP Name Uniqueness</name>
  <objectModelDefinition componentId="vrrp"/>
  <attributeDefinitions>
    <identifier componentId="name"/>
  </attributeDefinitions>
</uniquenessDefinition>
<!--END: Uniqueness Definition for VRRP-->
```

Figure 25. Uniqueness definition of VRRP NGO

Figure 25 shows the simple definition of the uniqueness of the object, and using this XML when it creates an object will validate the uniqueness of the name for componentId="vrrp" object and the name attribute of the object.

I also learn that we can integrate many customizations to the application via plugin development. And another new topic is that we are integrating network orchestration software for a multicountry telecom operator. Still, there may be some country-specific requirements too, which is another flexibility of the software and new learning of development area for me. The XML standards provide a crude scheme for validating that such a relationship exists using the ID and IDREF attribute types. Each attribute of type IDREF has a value that serves as reference to the value of another element that has an attribute of type Id. (Bradley 2002, 255).

```
@Override
public Command preCreate(Command arg0) throws GraniteException, Exception {
    // trigger the plugin for non Admin user only
    if (!PluginUtil.isAdminOrSuperUser(graniteSessionSupplier, graniteServiceFactory)) {
        logger.debug("VrrpCreatePlugin --> onEntry INVOKED");
        String teliaCountry = configurationProperties
            .getStringProperty(CommonConstants.COM_TELIACOMPANY_CLI_COUNTRY);
        // UUID generation only for Norway
        if (!("NO".equalsIgnoreCase(teliaCountry))) {
            VrrpService_Create vrrpService_create = (VrrpService_Create) arg0;
            Vrrp vrrp = vrrpService_create.getVrrp();
            // GENERATE UUID
            if (vrrp != null) {
                logger.info("Vrrp ## :: " + vrrp);
                String uuid_num = UUID.randomUUID().toString().toLowerCase();
                logger.info("UUID generated and to be set to Vrrp:!" + uuid_num);
                vrrp.setUuid(uuid_num);
            }
        }
    }
    return arg0;
}
```

Figure 26. Country-specific UUID generation for VRRP object

The previous figure shows that the requirement is achieved not for the whole application but only for a specific country. So the approach we took is if the country is the intended country, then only it will generate UUID; otherwise, not. It was a straightforward implementation, but it was not easy to analyze and select the approach. In the figure, we can see that a public method preCreate generating the UUID. A method definition that occurs in a class construct is made up of two distinct portions: the method signature header and its implementation code body surrounded by the braces{...} (Poo, Kiong & Ashok 2008, 19).

This week was another challenging quick learning activity for camunda BPM automation tool. Camunda is a Java-based framework for business process management workflow automation. It is using a modeler and file repository through REST/Java API, and workflow executes by tasks in the camunda cockpit, and this way, the automation flow from model to execute.

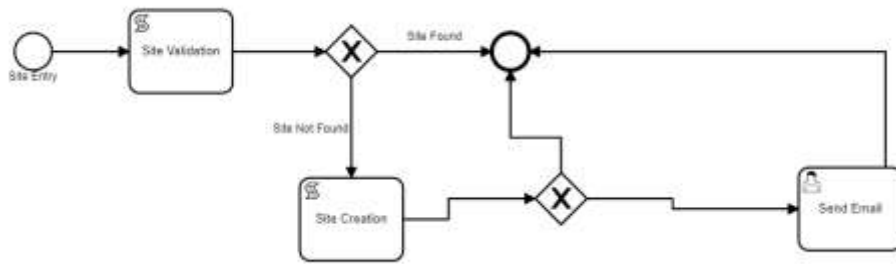


Figure 27. BPMN process diagram for a site object

Figure 27 shows the BPMN process diagram of a site creation of a network inventory workflow by Camunda Modeller. After triggering the start point event, it will execute the process's scripts, determine the next event of the flow, and finally end the process whether the site will be created or the intended site already exists in EAI. Camunda Platform is a system that is based on Java. It has a general emphasis on providing Java developers with the resources they need for developing, implementing, and running business processes and workflows on the JVM. The key components are written in Java. However, to make the process engine technology accessible to developers who aren't Java developers. As a result, Camunda Platform also includes a REST API. Camunda Platform can be used as a stand-alone process engine server or as part of custom Java apps. So, reasonably mutually decided with the customer to use Camunda for orchestration of use cases of resource synchronization between a third party software of customer network and the EAI for data sync and updating of the legacy system and implemented network inventory orchestration application. (Camunda Docs 2021).

## 4 Discussion and conclusions

I started my work just before the covid-19 pandemic and faced changes and challenges due to this extraordinary situation. I was focusing on different technologies previously, especially MERN Stack & others; the technologies I am mainly using in this project are Java, SQL, Oracle DB, Webservices, Red Hat tools, etc. It was a bit overwhelming and stressful position to adapt initially. Still, I was proactive and confident to follow along and learn all required technologies very first as some of the tools and technologies I studied and were did school projects in my university.

Fortunately, my management provided all I needed, such as a premium subscription to highly reputed online learning platforms, and sent me to the right person internally outside the team and instructor-based training sessions. While working on the project, I got

support from senior team members and spent time researching myself to understand and find many solutions and learned other required knowledge as the telecom domain has some new topics for me. After all, I found myself confident enough with other team members to contribute to many critical integration problems.

Individual empowerment in the organization made me more confident, and I was never afraid to ask anything or raise my voice in team meetings. I always ask for reviews with our Dev Lead, Solution Architects, or senior developers to learn and understand more by their valued feedback for any tasks. I also actively participated in critical debug sessions to learn about the root cause of any defect issues, which was a fascinating technique for newbies to learn more profound knowledge of any technology. Researching for any topic is a usual process many developers follow but at the same time, sharing with internal forums was a good way of finding solutions as we have developers forum internally who are working for similar projects of Ericsson.

I lacked experience in breaking down any development tasks like an engineer, which I found compelling and adapted to my work to progress for development tasks. In our agile team, I sometimes suggest modifying the requirements or acceptance criteria and design of the feature to provide flawlessness and an optimized solution. Others appreciated the development of software with a QA mindset in my work. This agile way of working also resulted in the acceptance of many customization features to the common solution. I think this was my job satisfaction, and an out of box approach initiated by a few other members and me followed later.

It was my desire to do my thesis work for a project that impacts many people. My current work project is an example of many residents in the region using our customer network, and I felt motivated to write this thesis. I have learned many new things throughout my working period, and I have become a professional communicator in a global environment. I also have learned a few new tools and technology like Groovy, Gradle, Camunda, new libraries of Java, and Robot Framework. Remarkably it was hands-on learning as I am learning those to complete my task, which made it an applicable skill of mine.

As part of the QA team, I have been working for test automation, where I have learned and gained experience of different testing approaches and test analysis and plan for a complex software project. Besides, in the development team, I learned many techniques of development and tools, for example, Camunda. I also discovered that a clear analysis

and plan for problem-solving is very important to follow, which was learned from the challenges I faced in the project. Groovy was another programming language I have learned during work to develop one integration plugin for a country-specific integration. Groovy has a similar syntax to Java and can be used as a scripting language for different purposes of Java-based applications and other use cases.

Dared to make strong suggestions to my seniors for some issues over the period; for example, I was able to attract management to having a pre-defined training model and materials for a new member of a similar project. I am confident that this initiative of preparing some learning material during my training for a future team member will create an impression of having overview level knowledge of the project quickly. I have also contributed to designing a new training session in every area of any similar project with different team members, which I believe will help a new team member easily sync with the team.

In my working period, I have developed a new feature for our customer's country-specific integration for sending mail notifications in the event of any manual task handled by a user. After the development of the feature, I found it interesting that this same feature can be implemented at the global level for all country our customer has operation in the region. After the solution architects and product owners' analysis, they have decided to implement it as a common application feature. It is undoubtedly will add tremendous value as a new feature of the software which also has more flexibility and customizability for many different use cases.

As a new member of the team, I was not allowed to use new technology for the project to solve problems we have to integrate Ericsson Operation Support Systems. I am aware that some internal research and trials ongoing within the company for moving to cloud-native, and having expertise and skills in cloud technologies will have demand in similar integration projects. I am very optimistic that in the near future. Ericsson will provide its OSS or BSS products ready for customer from own cloud applications. If I have any opportunity, I will try to have hands-on experiences in cloud-based application integration development.

Another futuristic area of the Ericsson Adaptive Inventory (EAI) will be intelligent network automation and inventory management system for 5G network operators. During the project progression, I found that telecom operators are already discussing artificial intelli-

gence and how it can be used for networks and other operations. Also, I agree that an intelligent system is the best solution for any complex telecom network and its management. It will be an exciting research opportunity for me as well as for Ericsson to work on artificial intelligence-powered features integration for the EAI or similar products of the company.

Every week's analysis of work provided me self-realizing feedback about my performance and a self-validation of my work. I was truly able to identify my drawbacks and further improvement key to follow in the coming weeks. Also, it was a good way of tracking my tasks during this thesis writing period; for example, I was able to plan efficiently for my daily task, keeping in mind the deadline of the sprint. I can say that the weekly writing analysis of work took some time, but it definitely made me more productive as I could understand my weakness from the previous week, and in the following week, my performance got better with the right plan.

Interestingly I was even able to produce my weekly analysis to the Head of Digital Service of Ericsson as my progress as a company trainee. I am convinced that my confidence and professionalism grew further because of this work analysis, where I could understand which way I should proceed to produce better results. Finally, I am happy that it was the right choice for my thesis as my Manager allowed me to do, and now my company is considering me to continue my employment, and this work analysis part played a role too.

## References

Boeg, J. 2012. Real Life Scrum. Trifork A/S. Denmark.

Bradley, N. 2002. XML Companion. 3<sup>rd</sup> Ed. Addison-Wesley. Boston.

Cohn, M. 2010. Succeeding with Agile Software Development Using Scrum. Addison-Wesley. USA.

Cohen, J., Teleki, S. & Brown, E. 2013. Best Kept Secrets of Peer Code Review. Smart-Bear Software. Beverly, MA.

Gregory, J. & Crispin, L. 2014. More Agile Testing. 1<sup>st</sup> ed. Addison-Wesley. NewYork.

Harold, E. 2013. Java Mail API. O'Reilly Media Inc. CA

Hinkula, J. 2018. Hands-On Full Stack Development with Spring Boot 2.0 and React. Packt Publishing Ltd. Birmingham.

James A., H. & Jim, H. 2002. Agile Software Development Ecosystems. Pearson Education, Inc. USA.

James, A. & Jim, H. 2000. Adaptive Software Development. Dorset House Publishing, New Yrok.

JavaMail™ API Design Specification Version 1.6. 2017. Oracle America Inc. CA.

König, D., King, P., Laforge, G., D'Arcy, H., Champeau, C., Pragt, E. & Skeet, J. 2015. Groovy in Action. 2<sup>nd</sup> ed. Manning Publications Co. NY.

Kousen, K. 2014. Making Java Groovy. Manning Publications Co. NY.

Poo, D., Kiong, D. & Ashok, S. 2008. Object-Oriented Programming and Java. 2<sup>nd</sup> Ed. Springer. Singapore.

Rhoton J. 2000. Programmer's Guide to Internet Mail. Digital Press. Boston.



Bi, T., Xia, X., Lo, D., Grundy, J. & Zimmermann, T. 2020. An Empirical Study of Release Note Production and Usage in Practice. IEEE Transactions on Software Engineering, 10.1109/TSE.2020.3038881, pp. 1-1.

Perry, W. 2006. Effective Methods for Software Testing. 3<sup>rd</sup> ed. Wiley Publishing Inc. Indianapolis.

Qureshi, R., Alshamat, S. & Sabir, F. 2014. Significance of the teamwork in agile software engineering. Science International-Lahore, 26, 1, pp. 117-120.

Sohan, S. M., Maurer, F., Anslow, C., & Robillard, M. P. 2017. A study of the effectiveness of usage examples in REST API documentation, IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC), 10.1109/VLHCC.2017.8103450, pp. 53-61.

About us. URL: <https://www.ericsson.com/en/about-us>. Accessed: 08 October 2020.

Adaptive Inventory. URL: <https://www.ericsson.com/en/portfolio/digital-services/automated-network-operations/orchestration/adaptive-inventory>. Accessed: 10 October 2020.

A multi-faceted language for the Java platform. URL: <http://groovy-lang.org/index.html>. Accessed: 29 December 2020.

Build Lifecycle. URL: [https://docs.gradle.org/current/userguide/build\\_lifecycle.html#sec:build\\_phases](https://docs.gradle.org/current/userguide/build_lifecycle.html#sec:build_phases). Accessed: 01 March 2021.

Camunda Docs. URL: <https://docs.camunda.org/manual/7.14/introduction/architecture/>. Accessed: 02 May 2021

Engineering higher quality through agile testing practices. URL: <https://www.atlassian.com/agile/software-development/testing>. Accessed: 19 December 2020.

Java – Sending Email. URL: [https://www.tutorialspoint.com/java/java\\_sending\\_email.htm](https://www.tutorialspoint.com/java/java_sending_email.htm). Accessed: 12 December 2020.

JUEL. URL: <http://juel.sourceforge.net/guide/start.html> Accessed: 02 March 2021.

Musienko, Y. 24 February 2021. WHAT IS PROOF OF CONCEPT (POC) IN SOFTWARE DEVELOPMENT. URL: <https://merehead.com/blog/proof-concept-software-development/>  
Accessed: 27 February 2021.

ROBOT FRAMEWORK/. URL: <https://robotframework.org/>. Accessed: 10 November 2020.

Understanding Bamboo Integration for CI/CD Pipeline. URL: <https://dev.to/successivetechblogs/understanding-bamboo-integration-for-ci-cd-pipeline-h79>. Accessed: 02 March 2021.

The Scrum Guide TM0. URL: <https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf#zoom=100>. Accessed: 12 October 2020.

## Appendices

### Appendix 1. Abbreviations

OSS	Operation Support Systems
BSS	Business Support Systems
EAI	Ericsson Adaptive Inventory
RF	Robot Framework
WFH	Work From Home
SDK	Software Development Kit
QA	Quality Assurance
VLAN	Virtual Local Area Network
REST	Representational state transfer
API	Application Programming Interface
DB	Database
HTTP	Hypertext Transfer Protocol
OOB	Out of Box
NGO	Next Generation Object
SMTP	Simple Mail Transfer Protocol
POJO	Plain Old Java Object
JVM	Java Virtual Machine
POC	Proof of Concept
GUI	Graphical User Interface
JAF	JavaBeans Activation Framework
MIME	Multipurpose Internet Mail Extensions
CI	Continuous Integration
CD	Continuous Deployment
JUEL	Java Unified Expression Language
JSON	JavaScript Object Notation
URL	Uniform Resource Locator
XML	Extensible Markup Language
VRRP	Virtual Router Redundancy Protocol
UUID	Universally Unique Identifier