AUTOMATION-TESTING FOR MONITORING THE NETWORK HEALTH OF MANUFACTURING WEB-BASED APPLICATION

Zakiah Ayop ¹, Nuridawati Mustafa ¹, Jonathan Chang ², Wong Boon Min ², Zaheera Zainal Abidin ¹, Lim Jie Hui ² and Syarulnaziah Anawar ¹

¹Faculty of Information and Communication Technology, Melaka National Technical University, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia
² Factory Integration, Infineon Technologies (Malaysia) Sdn. Bhd., Free Trade Zone, Batu Berendam, 75350 Melaka, Malaysia.

ABSTRACT: In this research, automation-testing approach is proposed to monitor the manufacturing web-based health application. Current monitoring system indicates that all components such as servers, applications, services, ports and URLs are not in critical condition, but operator on the production site could not operate the manufacturing application. The proposed approach will monitor the standard operation in manufacturing web-based application and determine the health state of the whole production. Automated script is executed and response time is captured as the performance indicator. A web-based reporting will display response times mapped in different graphs. From the preliminary testing result, graphs are compared and analyze. Finally, the comparison result will determine the abnormalities of the manufacturing application.

KEYWORDS: automation, monitoring, manufacturing application

1.0 INTRODUCTION

A collaborative monitoring system between system-level monitoring, application-level monitoring, end-to-end monitoring, custom monitoring, log analysis and with notifications integration is commonly adopted in global manufacturing companies to increase and improve the coverage of anomaly detection within short span of time. In distributed manufacturing systems such as oilfields, research has been conducted to monitor the steps of the productive process and to support decision-making into oilfields onshore. This specific ontology has been developed with the aim to provide agents with sufficient knowledge to manage components of oilfields [2]. This research is essential where some configuration changes in its components are required and due to complex decisions and too many intervening variables should be made by the managers and operators.

According to [3], this rating system and monitoring scheme is to monitor textile-stitching nonconformities conditions based on fuzzy weight and charting scheme. It is vital due to intensified competition in the global marketplace that has made organizations around the world realize that their survival highly depends on whether their provided products or services meet or surpass customer’s expectations. This demerit-fuzzy rating system, monitoring scheme and classification is essential to monitor online manufacturing processes.

A typical semiconductor backend assembly and test facility generally has a wide variety of products with each requiring different route specifications which resulting a considerable amount of process flows or routings. Monitoring is part of Modelling and Simulation Application component in backend assembly and test facility whereby monitoring agents monitor key performance indicators of the physical system. The status of the manufacturing system is collected by the monitoring agents to be updated into the database [1]. The monitoring agent will report any anomalies in the emulated system and triggers a what-if analysis to carry out simulation-based optimization to decide the best approach for handling the situation.

Automation testing is applied in managing product testing to minimize impact on manufacturing output [4]. Integrated test automation is used to run simulation of production wafers in a set of high-volume level prior to the production release. However, this study [1,4] does not monitor production in real time.

SOCRATES architecture [5], integrates web services from all corporate functions via Service-Oriented Architecture on the shop floor providing flexible processes with information visibility. This research focused on Internet-based process control has only resulted in small-scale capability.

Back-End operation in semiconductor manufacturing used integration from several software and tools for their high volume production. To ensure smooth operation, these tools and all components affected are monitored and managed 24x7 through customized web monitoring system. However, problems arise when software, servers, services and ports are healthy but the production sites could not operate the manufacturing application.

This paper will address the lack of monitoring process flow of the whole production system provided by the existing web monitoring system. The motivation of this paper is to monitor standard operation in manufacturing web-based application and determine the health state of the whole production. Automation-testing is proposed as an approach of process-health monitoring to monitor standard operation as an extension of the existing web monitoring system. In fact, automation-testing acts as a software test bed for analyzing, testing and validation of the whole process. Thus, automation-testing is applied to simulate the step-by-step
based on different variables (loading, operators, etc) for the next phase of analysis such as stabilizing the performance to an accepted time response.

This paper is organized as follows: The next section explains implementation details of automation-testing approach which entails theoretical framework of this study, methodology, automation testing design, data collection, data validation and data pre-processing. the process flow, performance indicator and experimental setup. Section 4.0 provides the preliminary result and discussion. Section 5.0 concludes some remarks and future work.

### 2.0 IMPLEMENTATION DETAIL

#### 2.1 Theoretical Framework

Automation-testing approach is based on web-based test automation framework in Figure 1 as it can be adapted to other tools. For this project, we added database (DB) for storing a mass of Test Data to pump into the test script and result data returned from the test. Section 2.3.2 explains the test script design proposed for this study.

\[
RT_A = (RTT \times \text{Turns}) + \frac{\text{Page Size}}{\text{Minimum Bandwidth}} + \text{SPT} + \text{CPT} - R_h
\]

*where*

\(RTT = \text{Round Trip Time};\ \text{SPT} = \text{Server Processing Time};\ \text{CPT} = \text{Client Processing Time};\ R_h = \text{hop-to-hop RTT} \)
Start
Open browser
Input ID and password
Click Login
Success?
Yes
Time\textsubscript{e,TA} = 0; Time\textsubscript{e,RTA} = 0

No

Display list of lots under the EquipmentID

Capture Time\textsubscript{e,RTA} = y

Input EquipmentID
Click WorkInProgress menu
Capture Time\textsubscript{e} = x

Store Time\textsubscript{e} and Time\textsubscript{e,RTA}

End
Close web browser

Figure 4: Proposed automation test flow chart

Click button using session "#Invoice" then wait until the browser is ready for input. Element search criteria is within document path "moduleFrame\#pageFrame\#identifier matches "ReferenceButton". If the step causes an error, continue the task.

Stop timer from session "IFXWIMPMainSession". Store elapsed time inside variable "IFXWIMPMain". The elapsed time is returned in number of Seconds.

Pause task for 2 seconds.

Set value of attribute "value" to "#Equipment25%", using session "mainwindow". Element search criteria is within document path "moduleFrame\#pageFrame\#identifier matches "Text1".

Figure 5: Partial automate script

Figure 6: (a) Login response time graph

Figure 6(b): WorkInProgress response time graph display
2.2 Methodology

Our methodology of carrying out this study is illustrated in Figure 2. Analysis of existing system consists of interviewing users, testing, manufacturing application and analysis of current monitoring system. Automation pilot test is done in staging environment where we are running automation test to applications that is not used for manufacturing chips in the production but mainly for simulation. Based on data collected in pilot test, we start to carry out the real automation testing in data collection phase. In this phase, configuration and scripting is done in real environment and data is collected will be validated to the next phase. Data collected is validated through a series of testing in the script to avoid noise data. Data pre-processed is where data collected is represented into a form of report. Lastly, evaluation is the phase where the engineers and IT admin analyse the information given and evaluate the current status of the system.

2.3 Automation-Testing Design

2.3.1 Performance Indicator

We derive response time of the manufacturing application web page based on the formula in [6]. Response time is measured by executing additional scripts from the client side that will include not only server processing time but it will also include time required to establish connection as well as transmission time over network. Although measuring response-time at client side will give accurate representation of quality of service experienced by web-users, it involves those components which are not governed by web-server and may not be helpful in gaining deep insight into current status of server and measuring web-server performance alone. On the other side, if we measure response-time at server side, it will incur some overhead on server and will carry out the real automation testing in data collection phase. In this phase, configuration and scripting is done in real environment and data is collected will be validated to the next phase. Data collected is validated through a series of testing in the script to avoid noise data. Data pre-processed is where data collected is represented into a form of report. Lastly, evaluation is the phase where the engineers and IT admin analyse the information given and evaluate the current status of the system.

2.3.2 Proposed Automation-Testing

The overall operating flowchart of the proposed approach is shown in Figure 3. Standard transaction in manufacturing web-based application is automated through a customized script. Although it does not monitor the current transaction running by human operator, it does monitor standard transaction and eliminate the idle time between actions like human do. In this case, it will represent the health of the Manufacturing Web-based application flow process.
MySQL and generate graph based on sites. It plots the data into graphs every 5 seconds and length of results display are set to daily or depends on the user. Figure 6(a) and (b) shows the graph modules display.

3.0 ANALYSIS AND EVALUATION
The preliminary results are illustrated in Figure 6(a) and (b). Graph in Figure 6(a) will determine the connection to the server. Y axis indicates time in seconds where as X axis is date and time of the script is tested. If the plot goes down to 0, it indicates the connection to the server is down and will notify the administrator of the server status through email or sms.

While in Figure 6(b), the graph determines the time taken to execute request to the database and displaying the result to webpage of WorkInProgress. If time of the script is plotted as 0 in login graph, the same logic is applied to the WorkInProgress graph.

By comparing both graphs average to the normal average time expected by the administrator, this graph will determine abnormalities of the manufacturing application. In this case, when login graph is normal, WorkInProgress graph is expected to be normal too. But if the WorkInProgress graph average does not reflect the login graph average, then there are abnormalities in the manufacturing application such as poor database query management or poor QOS management.

4.0 CONCLUSION AND FUTURE WORKS
An extensible approach to the web monitoring system of the backend manufacturing operation was suggested. The testing results conclude that automation-testing able to determine the issues arise in the standard operation transaction. Although the result is still in preliminary stage, the automation-testing contributed into two fold; a new approach in monitoring system by giving real representation of quality of service experience by the production worker. Secondly, automation testing can be used as troubleshooting network health by integrating php script to the automation tool and deploy one click away feature to the monitoring system. This features, integrate into mobile application as part of future works will enable real time monitoring accessible anywhere and anytime.

ACKNOWLEDGEMENTS
This project is part of Infineon Melaka University Collaboration Program under the Application “FI Application Overall Monitoring Framework”. The authors gratefully acknowledge the technical contribution of Dr. Jonathan Chang, Wong Boon Min, Lim Jie Hui, and BEAR Operation Team, Infineon Technologies. We thank our many partners from the Malacca and Singapore factories for their active participation.

REFERENCES