Proceedings of the 2021 Winter Simulation Conference
S. Kim, B. Feng, K. Smith, S. Masoud, Z. Zheng, C. Szabo and M. Loper, eds.

CASE STUDY: EMULATION OF AMAZON AIR HUB FOR MATERIAL FLOW CONTROL VALIDATION

Weilin Li
Hao Zhou
Ganesh Nanaware

Amazon
Worldwide Design Engineering
300 Boren Ave N
Seattle WA, 98109, USA

ABSTRACT

Discrete Event Simulation (DES) has been extensively used in Amazon Worldwide Design Engineering as a decision support tool to analyze and experiment with different warehouse layout, process, control logic, staffing and scheduling decisions. However, standalone simulation requires re-implementing all the algorithms and configurations of the actual warehouse flow control system within the simulation software. This results in some gaps between the real and "simulated" system no matter how closely the logic is being replicated. On top of that, there are still things that won't be captured by simulation such as network latency between hosts, databases and caches in production environment. This case study presents a high fidelity full system emulation developed for the Amazon KCVG Air Hub. It is used to debug, test, validate and optimize the flow control production software in a virtual environment.

1 BACKGROUND

The Amazon KCVG Air Hub, which is located at the existing Cincinnati/Northern Kentucky International Airport, is Amazon's largest, most advanced sortation facility ever. First phase of KCVG was launched in August 2021 while future phases will gradually add incremental sortation capacity to the final full build. KCVG allows Amazon to fulfill peak volume requirements faster and at a lower cost. Various automation mechanisms have been implemented at KCVG including sorters, robots, Amazon Robotics (AR) drives and so on. A combination of software from cross functional teams within Amazon contribute to the overall optimal flow of packages so that they can be sorted by their outbound destinations in an efficient manner. It is of vital importance to test and validate how individual software piece impacts the performance of each process as wells as the entire building.

2 SOLUTION

A Flexsim DES model was built to simulate hardware, vendor PLC that controls the physical sorters, and any manual processes within the building from Inbound dock to Outbound dock. The real time package routing and sortation decisions are made by the material flow control production software. In order to achieve that, a scalable emulator was developed based on C++ to integrate FlexSim simulation model with production software. The emulator is responsible for handling message receiving, message sequencing, data parsing, data translating, command execution and message replying between FlexSim and production software. The emulator was first tested through individual sorters, and eventually has been deployed in the KCVG full system emulation model.

3 USE CASES AND BENEFITS

As the first step, sorter level emulation testing were conducted to perform connectivity test, software integration reliability test, control logic and load balancing functionality test, item eligibility and lane full condition emulation test, sorter Anti-Grid Lock (AGL) test, as well as various sorter level failure mode what-if analyses. Sorter level emulation testing helped to identify multiple software issues including configuration script, timeout setting, flow control logic, etc.

Once all the production software were integrated, the end-to-end full system emulation was used to validate designed building throughput by running the baseline model with predicted nominal volume split. However, in real life the volume received and to be processed is drastically different by week of the year, by day of the week, even by hour of the day. Sensitivity analyses of volume swing scenarios (i.e., package type, package process path, volume split by outbound destination, etc.) were then conducted to understand system robustness to volume variation. Furthermore, emulation was used for various asset failure modes and mitigation strategies analyses at building level. It helped to quantify the throughput impact of each hardware failure, with and without reaction plans. Reaction plans are defined procedures to follow by both software side and operations side once asset is down, and are categorized by asset down time: short term (<10 minutes), medium term (10 minutes – 1 day), and long term (>1 day).

This full system emulation is a powerful tool that helps to improve first launch quality by letting engineers rapidly identify issues, fine tune algorithms, and implement changes ahead of the actual deployment in a virtual environment. It also allows stress testing the system against peak volume with various volume swing scenarios when it's difficult to acquire that many packages for testing purposes. It will continue to serve as a virtual validation platform as the site ramps up from launch and expands into the full-build in future years.

4 CHALLENGES

Because of the scale and complexity of KCVG, building a full system emulation requires cross functional efforts from all the software teams that are involved. We need to make sure that not only all the services are cleaned up and ready for emulation, but also the interface is handled properly between every two directly interacting services. Development timeline may be different for each team, some services are available for integration while others are still under design. If a certain service is not available, it needs to be mocked.

Second, since there's no clock synchronization between Flexsim and production software, it is critical to ensure that Flexsim can handle effectively such a large system model and be able to run at real time speed. We identified and optimized individual node or function in which CPU spends most time executing code, turned off 3D visual, and ran the emulation on a machine with the most powerful processor.

Third, in order to make the emulation environment more comparable to the production environment with real database connections and similar network latency, the material flow control software are currently set up manually on cloud environment. This means that only one emulation model can run at a time in real time speed. Automation in creating configurations, infrastructures, and new environment will make running multiple emulations feasible.

Lastly, it is harder to debug in an emulation compared to a standalone simulation. Since there are so many different software and services involved, model runs are hard to replicate. Having a centralized log system and KPI tracker from all software services is essential for tracking down errors during model run.