Proceedings of the 2024 Winter Simulation Conference
H. Lam, E. Azar, D. Batur, S. Gao, W. Xie, S. R. Hunter, and M. D. Rossetti, eds.

A DATA-DRIVEN FRAMEWORK FOR LOW CODE SIMULATION DEVELOPMENT IN THE ALUMINUM INDUSTRY

Ju Jeon[1], Abdurrahman Yavuz[1], Adarsh Gadepalli[1], and Aristotelis Thanos[1]

[1]Simulation & Optimization Center of Excellence, Novelis, Atlanta, GA, USA

ABSTRACT

Eagle Simulate is a novel Simulation Data-Driven Framework for efficient capacity analysis and decision support in industrial aluminum plant operations. The framework is used internally in Novelis and offers a user-friendly interface for intuitive scenario creation and management, and a scenario-driven database for data organization. At the core of the framework lies a model built with AnyLogic to accurately simulate the complexities of aluminum plant operations. A key feature is the use of data schemas for auto-building the simulation model, positioning it as a low-code/no-code solution. This allows users to assemble the model predominantly from the data layer, accelerating development and reducing dependency on traditional coding. We will explore the framework's components, scenario management, and provide insights into the simulation model. This framework represents a significant advancement in swiftly deploying and iterating simulation models, fostering informed decision-making and enhancing efficiency in aluminum production facilities.

1 INTRODUCTION

As the aluminum industry evolves, the need for efficient, data-driven solutions becomes increasingly critical to optimize production and resource utilization. To address this challenge, we have built “Eagle Simulate”. This simulation framework does not only simplify the process of creating and managing operational scenarios but also revolutionizes the way typically data is used in simulation models and aims to tackle some of the most persistent challenges faced when building simulation models. Many simulation models lack longevity and become obsolete due to their specific design focus. They are often tailored to address particular cases, without sufficient flexibility for user adaptation, leading to their eventual disuse. For a large manufacturing company with multiple facilities, these types of models are extremely limited since they cannot be reused, extended, or used as a basis for the capacity analysis and decision support across other plants. To that end, in Novelis we have built this framework that we use internally to rapidly spin-off simulations based on the business needs with limited involvement of expert simulation engineers. In the following sections, we will delve into the framework components, explore the intricacies of scenario management, and provide detailed insights into the simulation model. Our goal is to demonstrate how this framework can streamline simulation creation by fostering accessible simulation development, adhering to standardized practices, and ensuring scalability.

2 FRAMEWORK ARCHITECTURE

This framework architecture is designed around a deployed simulation model that programmatically constructs the aluminum plant represented by input data, sourced from a Microsoft SQL Server database. The simulation model is designed to allow the model to adapt according to the provided data and uses a custom simulation library to create a virtual representation of a plant. The same SQL Server database is also utilized to store the output data from the simulation, ensuring a centralized and efficient data management system. The user interface is built using React, and is a key component of the architecture,

serving as a hub for user interaction. It allows users to edit and view the input data from the model, providing them with direct control over the simulation parameters. Additionally, the interface enables users to manage the simulation process and view its outputs.

3 DATA AND SCENARIO MANAGEMENT

The data design for the application was structured to ensure scalability and extensibility. It involves creating a robust data architecture that supports various data types including the custom aluminum library objects. The design includes the use of a relational database to facilitate not only the initialization of the model but also the development of the user interface. The data required for initializing simulation scenarios can be bifurcated into two categories: data specific to the scenario, and data that is not specific to the scenario.

Data that is not specific to the scenario comprises of feature enumerations that the user can choose from our existing library palette, which remain consistent across different scenarios. This could include options for machine/storage types and parameters. On the other hand, scenario specific data encompasses detailed configuration of the plant the user is creating. Most of the user-provided data includes details that define instances of machines, storage areas, and nodes.

Scenario management is integrated into the data design to support flexible use cases. It involves the creation of scenarios that represent different operational environments, user interactions, and system behaviors. These scenarios are stored in the database server and used to simulate and analyze various conditions under which the application may operate, allowing for comprehensive testing and optimization in a single platform.

Overall, the data design and scenario management components work in tandem to provide a robust foundation for the application, ensuring it is equipped to handle modifications in the event of library updates with minimal changes, if any.

4 SIMULATION MODEL

The simulation model developed is constructed using discrete-event and agent-based modelling principles. The model follows object-oriented programming and leverages the custom library objects to drive an operational virtual plant. The library incorporates machines, storages, transporters, and elements common to an aluminum plant such as hot mill, cold mill, overhead cranes, etc. and includes logical components to drive production, generate demand and route entities through the plant. Each of these library objects are designed to be instantiated with user defined parameters and built to be highly configurable, allowing for easy modification of parameters and conditions to suit different study requirements.

The primary environment is designed with the flexibility to initialize various facility configurations, acknowledging that even within the same business, operations can vary from one facility to another. The model's modular and highly adaptable design enables users to construct and personalize a virtual plant without the need for direct engagement with the code of the simulation model or specialized knowledge in simulation modeling.

5 CONCLUSION

The framework’s unique feature is its use of data schemas for auto-building simulation models enabling the creation of a model without the need for coding. This is particularly beneficial if a lower fidelity model is acceptable, and the library includes all the necessary functionality. Eagle Simulate focus on data-driven practices significantly accelerates the process of creating models, reduces dependency on traditional coding, and empowers the end user to conduct what-if analysis, manage, and compare their scenarios in a seamless way. It streamlines simulation creation, fostering accessible simulation development adhering to standardized practices and ensuring scalability.