Transactions of the Korean Nuclear Society Spring Meeting
Jeju, Korea, May 22-23, 2025

Development of Web-based Digital Core Twin System Using RAST-K/RAST-Q

Dongmin Yun[a], Sang-Rae Moon[b], Ji-woo Lee[c], Deokjung Lee[a*]
[a]*Department of Nuclear Engineering, Ulsan National Institute of Science and Technology, UNIST-gil 50, Eonyang-eup, Ulju-gun, Ulsan, 44919*
[b]*Core Analysis Group, Korea Hydro & Nuclear Power Central Research Institute (KHNP-CRI), Daejeon, 34101*
[c]*Expansion & Universal Co., Ltd, Techno 4-ro 17, Yuseong-gu, Daejeon, 34013*
[*]*Corresponding author: deokjung@unist.ac.kr*

***Keywords:** Web-based digital core twin, core adaptation, HDF5, web-based visualization

1. Introduction

A digital twin (DT) is a virtual representation that serves as a real-time digital replica of a physical object or process such as manufacturing, urban planning, or automotive plant industries [1][2]. The web-based digital core twin (WDCT) is a virtual model of the reactor core that utilizes real-time data for adaptive core monitoring and simulation, in the web environment. Real-time monitoring and twin-based simulation capabilities are critical for ensuring immediate access to plant information. The WDCT system is designed for nuclear power plant operators, reactor engineers, and nuclear safety analysts, providing an advanced tool for core condition assessment and operational decision-making.

The WDCT system consists of four specialized servers: the Design Core, As-Is Core, As-Will Core, and Quasi-Simulation Core. These servers communicate through a main controller server, ensuring mutual data exchange and integration via each server. The Design Core verifies core conditions based on nuclear design reports for each unit and cycle. The As-Is Core adapts real-time core conditions using snapshot data from the Plant Information System (PIS). The As-Will Core and Quasi-Simulation Core predict future reactor behavior based on user-defined scenarios, such as boron dilution, control rod maneuvers, and power level changes.

The development of the WDCT system involves four key components: front-end interface, back-end server, simulation engine, and database infrastructure. The front-end has been built using enuSpace, developed by Expansion & Universal Ltd., which enables the design of interactive graphical components and facilitates communication with the back-end via HTTP requests through Lua scripts. The enuSpace has been developed in Expansion & Universal Ltd., The back-end server, implemented in .NET 6.0 and hosted on IIS, manages controller APIs for input generation, simulation execution, data retrieval, and caching for multiple users. The simulation engine is based on RAST-K v2.0 [3], integrated as a dynamic link library (DLL) for core physics calculations. The database stores RAST-K simulation results along with various input and output datasets, ensuring efficient data management and retrieval. This paper presents the development process of the WDCT system, detailing its architecture, core functionalities, and potential applications in nuclear reactor operations and safety analysis.

2. Methods & Results

2.1. WDCT components

2.1.1. Design Core

The Design Core server performs verification calculations based on nuclear design reports (NDR) for each unit and cycle. Acting as an NDR result viewer, it displays pre-calculated HDF5 data derived from RAST-K simulations according to specified input parameters.

In addition to its core verification function, the server supports arbitrary burnup restart calculations, enabling users to select any restart point (t_a) and interpolate pin-wise burnup and number densities between two adjacent time points (t_i) and (t_{i+1}). It performs single-step criticality calculations and provides visualizations of the computed results, enhancing user interpretation and analysis.

2.1.2. As-Is Core

The As-Is Core dynamically adjusts core conditions in real-time by utilizing snapshot data obtained from the PIS. The database PostgreSQL continuously queries major core status parameters in every 5 seconds. As-Is Core server supports adaptation calculations with queried adaptation parameter X_{target}, such as T_{in}, mass flow rate, core relative power, control rod group positions, and ASI.

During this process, the adaptation factor (m_i) and the axial neutron cross-section (XS_i) are iteratively updated while applying weight from previous values of parameter.

$$m_i = m_{i-1} + \left(X_{target} - X_{i-1}\right) \times \frac{\Delta m_{i-1}}{\Delta X_{i-1}} \quad (1)$$

$$XS_i = XS_{i-1} + XS_{i-1} m_i f(z) \quad (2)$$

, where X_{target} represents the adaptation parameter provided by the PIS data server, and $f(z)$ denotes the user-defined shape tuning function for cross-section weighting. This iterative approach continuously refines the cross-section data axially, such tuning that the calculated X values match the real-time operational data (X_{target}) as closely as possible. Users can check which parameters are used for further adaptation calculations in simulation, shown in Fig. 1.

Fig. 1. GUI for checkboxes of adaptation parameters

2.1.3. As-Will Core

The As-Will Core server performs predictive simulations based on restart files obtained from either the Design Core or As-Is Core. Using these restart files as initial conditions, the As-Will Core simulates future reactor behavior according to user-defined scenarios. The predictive calculations are executed using RAST-K, which models future core conditions over various time intervals, such as hours, days, or months.

To configure predictive simulations, the As-Will Core requires a set of global input parameters that define the overall conditions for the simulation. These parameters include the cycle burnup for prediction, the number of burnup steps, the final burnup time, critical search options, and xenon treatment options. By initializing these settings, users can establish the fundamental conditions under which the predictive simulation will operate.

During the setting simulation scenario of As-Will, users must also define stepwise core conditions at each burnup step. The As-Will Core applies these user-specified conditions to accurately model the evolving core state. The key parameters used in this process include depletion step intervals, boron concentration, coolant mass flow rate, coolant inlet temperature, relative core power, and control rod bank positions. These parameters play a crucial role in determining how the core will behave under different operational scenarios.

2.1.4. Quasi Simulation (QS)

The Quasi Simulation (QS) server provides quasi-static simulation capabilities without depletion calculations, enabling rapid analysis of short-term core predictive behavior. The QS server utilizes results from the Design Core, As-Is Core, and As-Will Core to generate macroscopic cross-section (macro XS) data for RAST-Q. Users can directly upload locally generated macro XS files to the server.

To initialize a project of QS, the system requires global input settings similar to those used in As-Will Core. The interface of option controller of QS is shown in Fig. 2. These settings include the input scenario for prediction, the number of simulation steps, the final depletion time, critical search options, and xenon treatment options. These input parameters define the baseline conditions of all time-steps for the quasi-simulation.

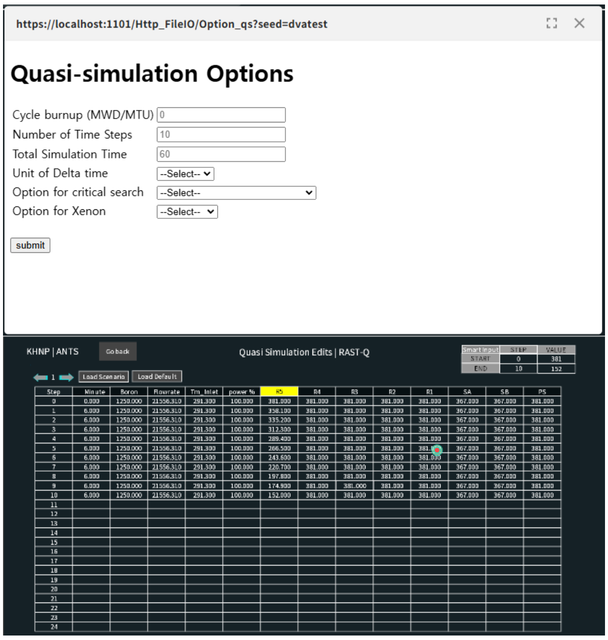

Fig. 2. GUI for global options and stepwise settings for QS

2.2. WDCT architecture

2.2.1 Front-end graphical user interface (GUI)

The front-end GUI components are visually designed using the Scalable Vector Graphics (SVG) format, which enables interactive and scalable visual elements on the web environment. The pre-compiled enuSpace control APIs enable dynamic async interactions from user's requests. When a user clicks a button component, an embedded Lua script and triggers HttpRequestPost API which implements HTTP POST request to the back-end server, transmitting specified URL, request body, and callbacks address.

2.2.2 Back-end web server

The data flow between the user interface, 4 core twin servers, and the simulation engine is done by back-end web server. The back-end server performs several key functions, including project setup and management for each simulation unit and cycle for multi-users, input interface management for each core twin server, and the generation and transfer of data to the simulation engine (RAST-K). It also handles the storage and parsing of the generated HDF5 output data and preprocesses it for front-end visualization purposes, ensuring that all data is properly serialized and ready for delivery to the front-end display components.

Additionally, the back-end server provides user-friendly features such as multi-user metadata management, including the creation, storage, loading, and deletion of simulation-related metadata for different users. It also includes monitoring functions that track the status of each core twin server, ensuring system stability and providing real-time feedback on server operations.

2.2.3 Simulation engine: RAST-K

The requested simulation from web server is calculated by functionalized RAST-K and RAST-Q kernel, a high-fidelity, nodal diffusion-based core analysis code designed for light water reactor simulations. It is utilized across multiple digital twin cores to provide accurate and efficient core calculations based on both real-time plant data and user-defined predictive scenarios.

For the As-Is Core simulation, RAST-K provides simulation results for the current depletion step by initializing with restart files derived from either real-time data or design-based conditions. Once initialized, RAST-K executes a predictor calculation to generate up-to-date core parameters for the twined simulation.

For the As-Will Core simulation, RAST-K implements prediction calculation based on the restart file, including the data of number densities of latest state of the As-Is Core. Another predictive Quasi Simulation utilizes the computation engine RAST-Q, which is a depletion-omitted light version of RAST-K for rapid core analysis. Instead of performing full neutron transport calculations, RAST-Q receives macro XS data as input and omitting micro XS feedback for faster simulation at the specific pre-calculated core conditions.

During predictive simulations, user-specified global options for these parameters are maintained as constraints throughout the calculation, enabling the system to predict the core's future state while preserving the adaptation conditions. The results from these predictive depletion calculations are then written to HDF5 files, where they are further processed for visualization and analysis within the WDCT framework.

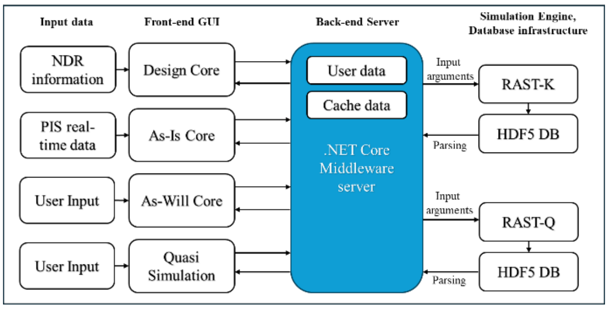

Fig. 3. The architecture of WDCT

2.3. Data Visualization

The result data saved in HDF5 available for extraction through these subroutines include the three-dimensional distribution data of normalized power, core burnup, fuel temperature, coolant temperature, coolant density, and fission products like xenon, iodine, and samarium. It is also extracted that the burnup-wise core parameters such as critical boron concentration, ASI, peaking factors, linear power density (LPD), and minimum departure of nucleate boiling ratio (m-DNBR).

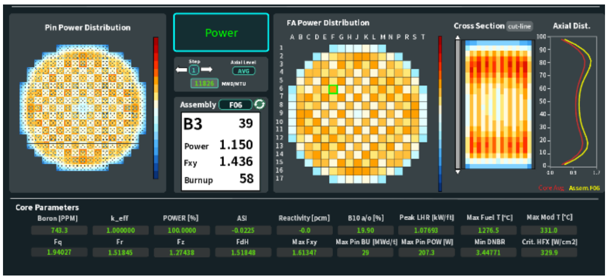
Fig. 4. The web page of overall data visualization

For data visualization to the front-end, the enuSpace platform requires serialized string data in CSV format to be compatible with its visualization components. The back-end server provides data serialization modules that process multidimensional and burnup-wise data extracted from HDF5 outputs and converts them into CSV-formatted strings. The visualization component at the front-end GUI shows color map with respect to given string data, shown in Fig. 5. and Fig. 6..

Every box and button are selectable for interactive visualization for local data. When clicking the fuel assembly (FA) in 2-D contour, the pin-wise data and assembly axial distribution for selected FA, are displayed. When clicking the y-axis from the line-chart component, users can reset the range of y-axis.

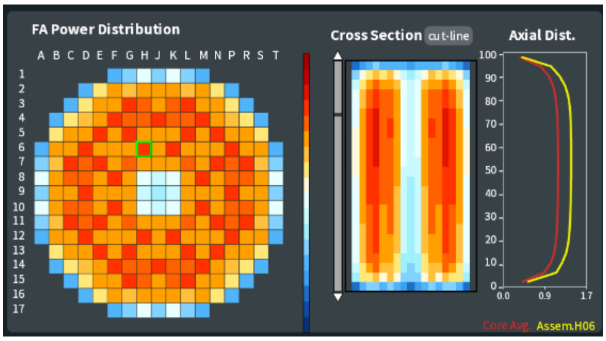

Fig. 5. The color-map visualization of multidimensional data (2-D, cross-sectional, and axial distribution)

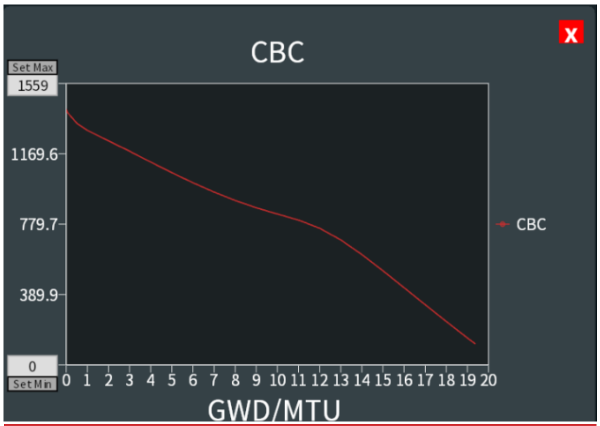

Fig. 6. The line-chart visualization of burnup-wise data

3. Conclusion

In this paper, the overall development of WDCT system has been explained. It is structured around four primary core server components —Design Core, As-Is Core, As-Will Core, and Quasi Simulation, enabling comprehensive simulation and visualization of various reactor core states, spanning from core design to real-time plant operations and future scenario predictions.

For the visualization of simulation outputs, the system utilizes the enuSpace Meta front-end toolkit, which is optimized for rendering data provided by the back-end server across multiple dimensions and channels in a web environment. The back-end server, developed using Microsoft .NET 6.0 and C#, systematically handles the entire data pipeline—from input file generation and HDF5 data parsing to output data serialization—ensuring both robustness and scalability of the digital twin system.

The system integrates with already existing core monitoring systems as unified platforms, leveraging a new centralized database PostgreSQL for mutual data management and inter-server communication between the WDCT and PIS server. This system is developed to help reactor operators and core analysts as its primary users, integrating a sophisticated environment for real-time core data monitoring and scenario-based prediction of core states. The WDCT facilitates more accurate and timely decision-making for users-friendly web-based interface, contributing to the enhancement of operational safety and efficiency in nuclear power plant management.

REFERENCES

[1] Moi, Torbjørn; Cibicik, Andrej; Rølvåg, Terje. "Digital twin based condition monitoring of a knuckle boom crane: An experimental study". Engineering Failure Analysis. 112: 104517. (2020) doi:10.1016/j.engfailanal.2020.104517. hdl:11250/2650461. ISSN 1350-6307.

[2] Haag, Sebastian; Anderl, Reiner. "Digital twin – Proof of concept". Manufacturing Letters. Industry 4.0 and Smart Manufacturing. 15: 64–66. (2018)

[3] Jinsu Park, Jaerim Jang, Hanjoo Kim, Jiwon Choe, Dongmin Yun, Peng Zhang, Alexey Cherezov, Deokjung Lee*, “RAST-K v2- Three Dimensional Nodal Diffusion Code for Pressurized Water Reactor Core Analysis”, Energies, 13: 6324,