Other Articles

- BMS Architecture
- Why Naming Conventions Are the Key to Successful Integration
- BMS 1.5: Out with the Old, In with the Sleek!

BMS Architecture

Once upon a time, in the cozy town of COG, there was a charming little restaurant called “BMS.” Imagine this: customers dining at tables practically next to the sizzling stove, with no clear division between the kitchen (back-end) and the dining area (front-end). The restaurant was run by Java, a talented chef who also took on the role of server. For years, everything seemed to work just fine, but there was a catch: if Java had a bad day with even one dish, the whole place would close. Need a fresh coat of paint? New chairs? You guessed it: shutters down!

To make things even trickier, Java was super picky about where and how to set up his restaurant. If he didn’t like the location or equipment available (hardware), he simply wouldn’t budge. Plus, he kept his secret recipes under lock and key, meaning no one else could lend a hand (no Docker containers). Renovations (adding more features)? Forget it! Even a simple expansion was impossible!

Then, one fateful day, a revolutionary idea came to light. Fancy little connectors, known as APIs, were introduced not only to help different parts of the restaurant communicate seamlessly but also allowed it to communicate with other restaurants. The kitchen (back-end) was separated and got a makeover with Laravel-PHP, while the dining room (front-end) was also rebuilt completely using Vue-JavaScript. Suddenly, chefs could focus solely on their delicious creations, while a dedicated team of servers managed orders and reservations like pros.

The entire operation became standardized, and training materials were now easily shared with new chefs and servers, making the business scalable. If there was an issue with a dish, the restaurant could stay open. Renovations? No problem! We could now make the restaurant work more efficiently in any way we chose. Everyone could focus on what they loved most and did best.

And so, in the newly thriving “BMS,” they all lived deliciously ever after!

Diagram 1: BMS Architecture

Diagram 2: Why do we need to separate the kitchen from the dining area and other parts?

image.png

Why Naming Conventions Are the Key to Successful Integration

Imagine merging three kitchens into one. Everyone agrees it'll save time and space - until you realize one person calls what's inside a jar "flour," another insists it's "powder," and the third just labels it "white stuff." But wait - it gets worse. The "flour" jar from kitchen one is not quite the same as the "powder" jar from kitchen two, and neither matches whatever kitchen three considers "white stuff." Now imagine trying to bake a cake under these conditions. Welcome to the chaos of merging systems across different LoBs without standardizing naming conventions.

This isn't just hypothetical - big players have faced this. When Microsoft acquired LinkedIn in 2016, both platforms had to merge data like contact information and user IDs. They had fields for user accounts and personal data, like "username" and "email," but their definitions weren't aligned. When American Airlines and US Airways merged in 2013, their reservation systems had to merge too. They used different terms and structures for customer IDs, booking references, and loyalty program numbers. This caused major headaches, from duplicate records to misplaced reservations - until they adopted standardization and reconciliation.

In the examples above, when seeking unity, each company found themselves stuck in a maze of mismatched concepts. But once they adopted a universal naming system, order was restored, and they became some of the strongest players in their industries.

Standardizing naming conventions isn't about stripping away identity but adding thoughtful layers that make the system inclusive and adaptable, so that no one's stuck in the middle of a system-wide food fight. Think of it as adding lanes to a highway- some drive on the left, others on the right, but the system supports both with clear signs and rules.

By adopting the majority's naming system (e.g., Program, Project, Work Packages, Purchase Order) as the foundation, we establish a common framework that most can relate to and understand. This reduces ambiguity, speeds up collaboration, and eliminates redundant or conflicting concepts. At the same time, introducing new terms (e.g., Program Area, Project Area, Purchase Request) ensures flexibility for LoBs with unique requirements.

Together, we can move from mismatched systems to a unified, efficient approach - because the best way forward is a road we all travel on together.

ChatGPT Image Jan 21, 2026, 10_37_21 AM.png

Image source: AI-generated created for illustrative purposes.

BMS 1.5: Out with the Old, In with the Sleek!

We began with DMS as the blueprint, and over time, BMS has evolved and been continuously improved. In recent years, we've been enhancing it with new technologies, and this April 2025, we're excited to unveil Version 1.5 - a sleek, modern redesign. This update not only marks a significant milestone in our organization's transformation from **CANDU Owners Group** to **Conexus Nuclear**, but it also brings a host of improvements. By leveraging cutting-edge technologies and refining the system's structure, we've boosted its performance, scalability, and user experience. With these upgrades, the system is now better equipped to meet the demands of today's fast-paced environment, all while preserving the core functionality that users depend on.

BMS version 1.0

Screenshot 2024-08-28 154804.png

BMS Version 1.5

image.png