June 1, 2023

John Darrington
Lead Architect, Digital Engineering

Nuclear Rust

Fission powered code for the modern world.

Battelle Energy Alliance manages INL for the
U.S. Department of Energy's Office of Nuclear Energy

Idaho National Laboratory

Biography

Idaho National Laboratory
1955 N Fremont Ave
Idaho Falls, ID 83427

John Darrington
Architect, Software Engineer, Data Scientist

John currently serves as the Lead Architect within the Department of Digital Engineering at the Idaho National Laboratory based in Idaho Falls, ID. In this role John works closely with project stakeholders in identifying and building the software solutions necessary to accomplish their many goals in the Digital Twin and Digital Transformation space. With experience in designing software at scale for industries, John helps identify and deploy data management solutions across a wide variety of disciplines. John is also the primary author of the ontological graph data management warehouse DeepLynx and various tools in that ecosystem.

Fukushima Daiichi Accident (2011)

- 15-meter-high tsunami disabled power supply and cooling of three Fukushima Daiichi reactors
- All three cores largely melted within the first three days
- Took until December of that same year to reach official “cold shutdown condition”

“No industry is immune from accidents, but all industries learn from them.”

- World-nuclear.org

Nuclear Software Industry State

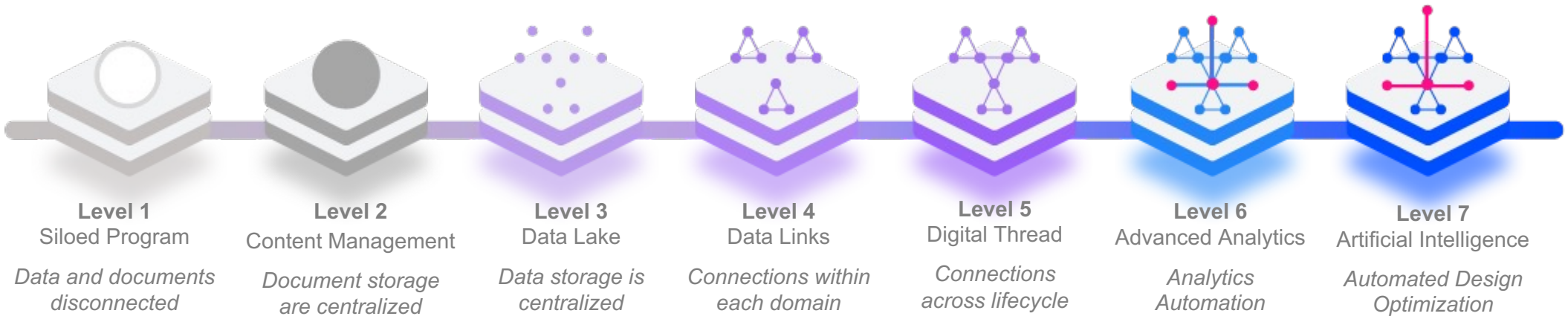
- Risk adverse
- Change is slow
- Committees
- Set way of doing things
- Ancient protocols

Why we feel we need to introduce a new language (original slide 2022)

- Embedded/Edge DeepLynx and other projects need a lower-level programming language than what we typically use
- Node.js is not suited for long-term use by the department in this area due to its performance issues, lack of well adopted scientific computing (at least at the lab), and runtime requirements
- Python, while in use, requires a runtime making it not suitable for all deployment pathways– performance also a potential concern for issues not covered by internal C libraries – extremely permissive and allows easy introduction of bugs
- Need to make conscious decisions before the decision is made for us or by accident
- C++ tooling is difficult to learn and causes issues
- Safety and lower-level languages necessary for operational control

Digital Twin Forms and Levels

- **Descriptive**: Visual replica
- **Informative**: Basic insights
- **Predictive**: Integration with operations
- **Comprehensive**: Integrated physics
- **Transformative**: Autonomous operation



Why Safety Matters

- Roughly 70% of all CVEs at Microsoft are memory safety issues.
- 2/3s of Linux kernel security vulnerabilities come from memory safety issues.
- An Apple study found that 60-70% of vulnerabilities in iOS and macOS are memory safety vulnerabilities.
- Google estimated that 90% of Android vulnerabilities are memory safety issues.
- 70% of all Chrome security bugs are memory safety issues.
- An analysis of 0-days that were discovered being exploited in the wild found that more than 80% of the exploited - vulnerabilities were memory safety issues.

- Okta.com

Rust
(original slide 2022)

- Sponsored by Mozilla 2010
- Statically typed, compiled language – can target many different chipsets and OSes easily
- No garbage collector – compiler enforces memory safety
- Lean standard library and large package ecosystem with centralized package manager (crates.io)
- Concurrency model built in, emphasis on memory safety in concurrent operations
- Complex, steeper learning curve (but has features such as generics, zero cost abstraction, pattern matching etc.)
- Community of 2.2 million developers (/Data) (active on messaging boards, GitHub etc.)
- Developer environment streamlined setup
- Excellent interop w/other languages

- Open-source ontology driven data lake with tabular/timeseries data support
- Originally pure Node.js
- Rust introduced 3 months after initial discussions – PostgreSQL fast loader module
- Rust drives new timeseries/tabular data model and Delta Lake integration
- RedisGraph fast loader
- LLM (Llama-rs) integration

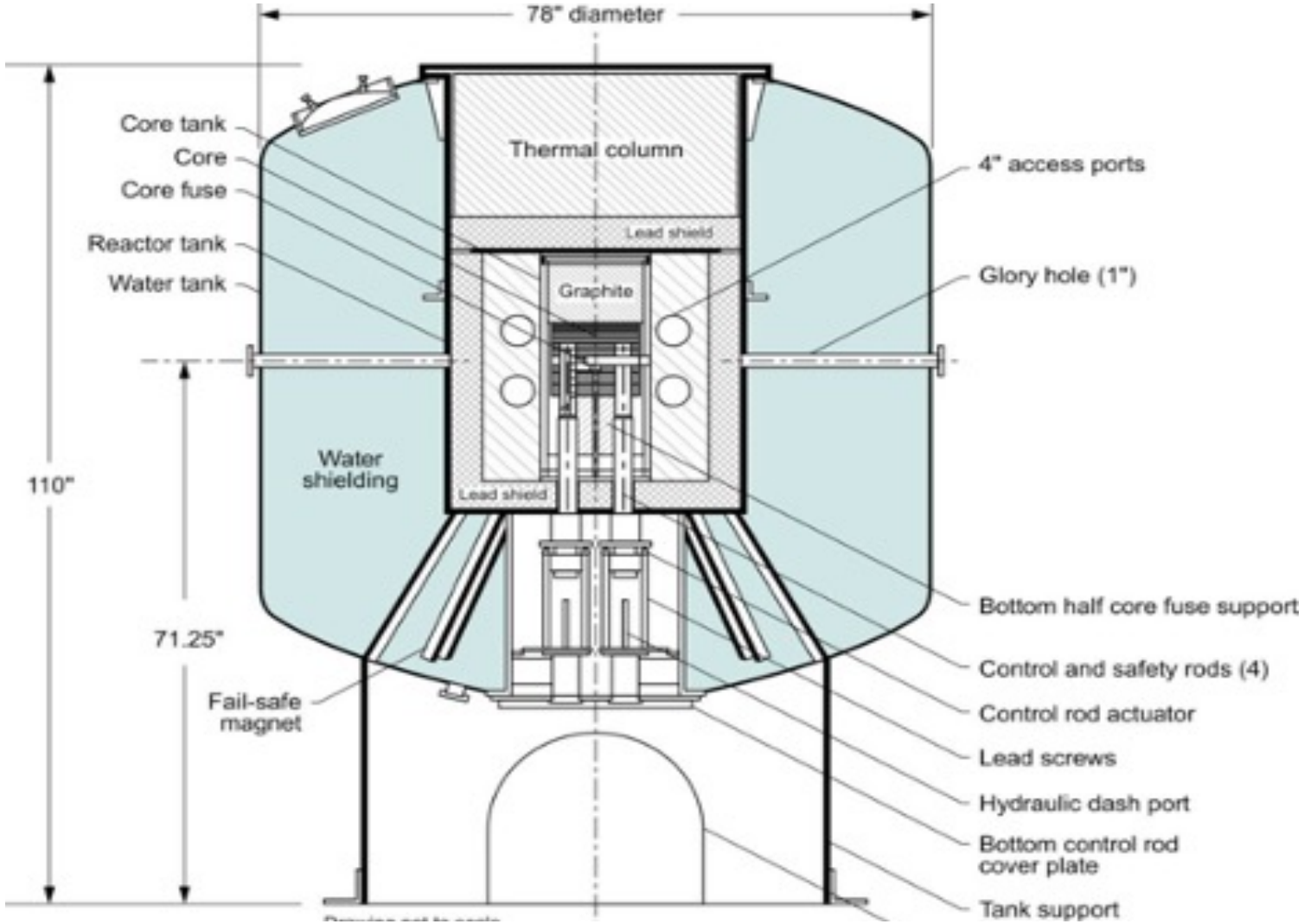
Serval

- Web-socket multiplayer game server
- Drives AR/VR multiplayer integrations for digital twins
- Drives multiplayer integrations for DeepLynx 3-D viewer

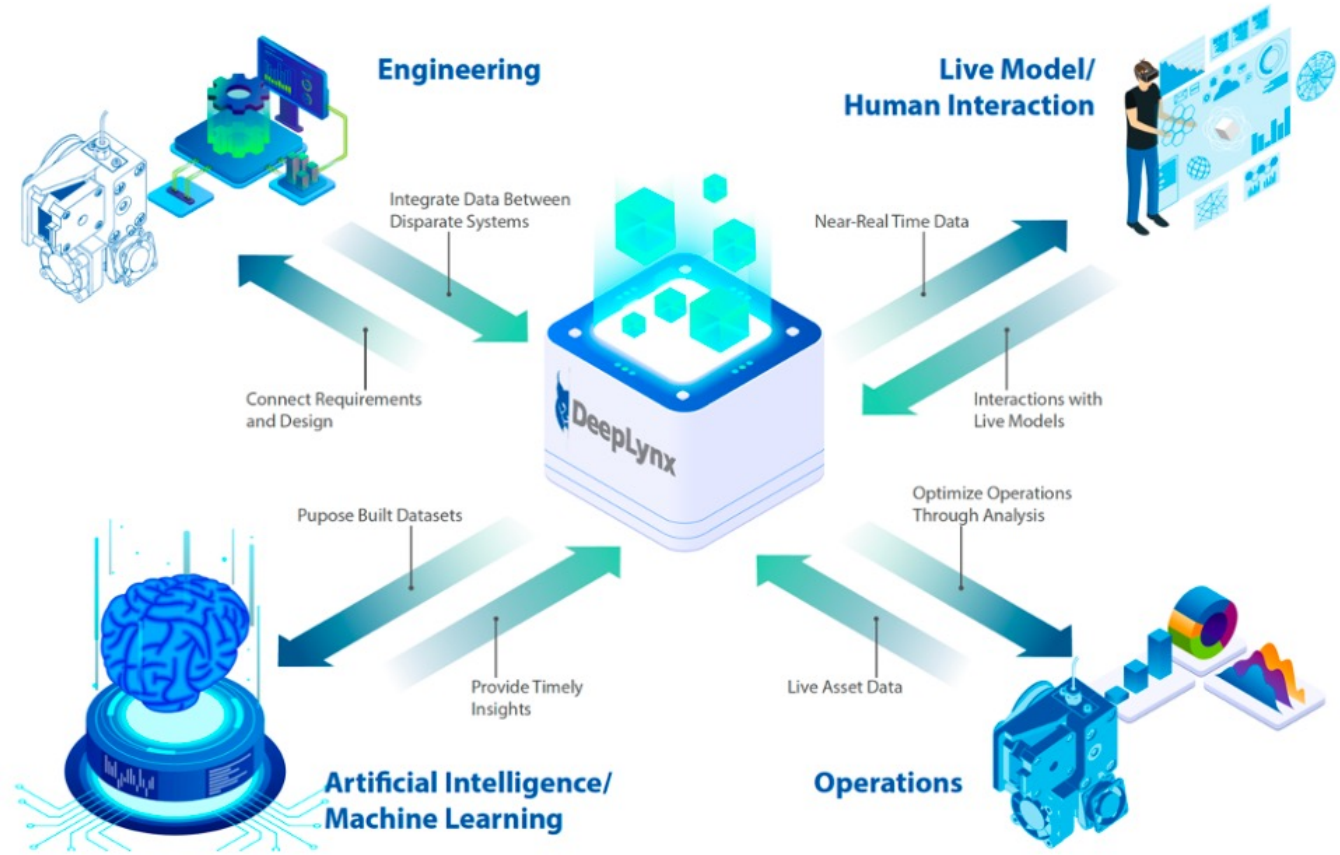
DeepLynx Loader

- Python & C# module for caching timeseries/tabular data
- Leverages DuckDB
- Utilized in many machine learning/ai pipelines

AGN-201 Nuclear Reactor



AGN-201 Digital Twin Basic Flow



Our Bet Paid Off

- Successfully launched the AGN-201 Digital Twin in less than 6 months
- Maintained speed and accuracy of data by using native modules and Rust
- Demonstrated the ability for Rust and current best practices to eventually become part of current nuclear control instruments and paradigms

Lessons Learned

- Find what drives management and those who make the big decisions – use those motivating factors to drive innovation
- Have a gameplan – but leave enough wiggle room for experimentation
- Prototype early and often

Idaho National Laboratory

Battelle Energy Alliance manages INL for the U.S. Department of Energy's Office of Nuclear Energy. INL is the nation's center for nuclear energy research and development, and also performs research in each of DOE's strategic goal areas: energy, national security, science and the environment.