Matryoshka: Realizing Hyperscale Data Center Network Design for the AI Era

NSDI'26 Spring · Experience Track

Yan Cai[1], Jialong Li[2], Kutalmis Akpinar[1], TianXiang Li[1], Hany Morsy[1], Jason Wilson[1], Sunil Khaunte[1], Yiting Xia[3], Ying Zhang[1]

1 - Meta · 2 - Shenzhen University of Advanced Technology · 3 - Max Planck Institute for Informatics

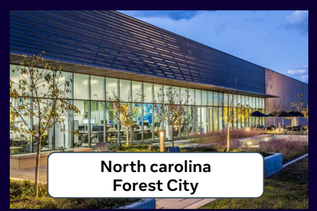
Meta

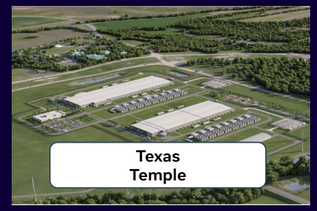
Gap: from Network Design to Working Configurations

- **Key question**: network design ⇒ a live running network
- **Design realization** (topology, IP addressing, routing, etc.)
- At Meta's scale **(900 DCNs)**: **A system problem,** rather than a scripting problem
- Missing from literature: **switch config generation for hyperscale DC networks**
- This gap is **especially acute in the AI era**

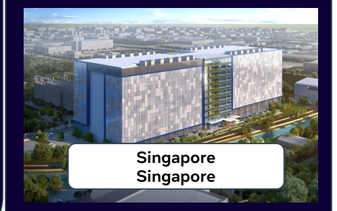
Meta DCN Infrastructure Overview

North America





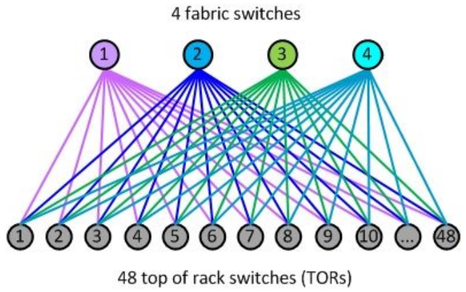
Europe & Asia

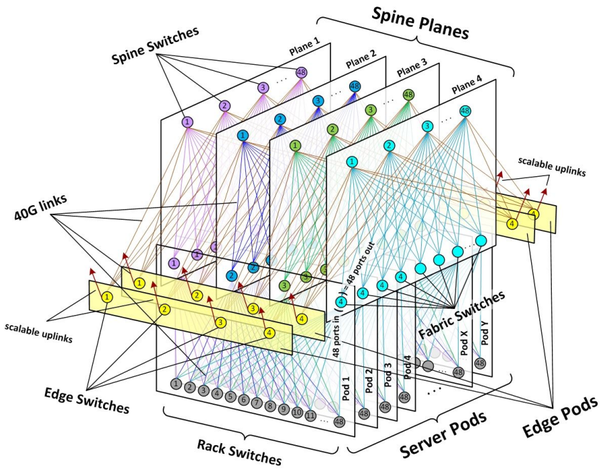


Meta DC regions (https://datacenters.atmeta.com/all-locations/)

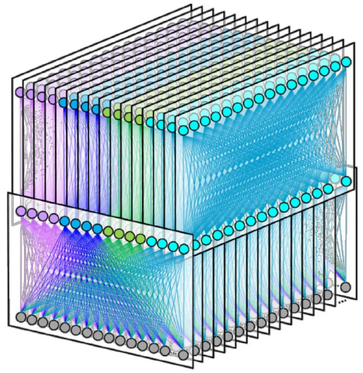
Timeline of Meta DC Evolution

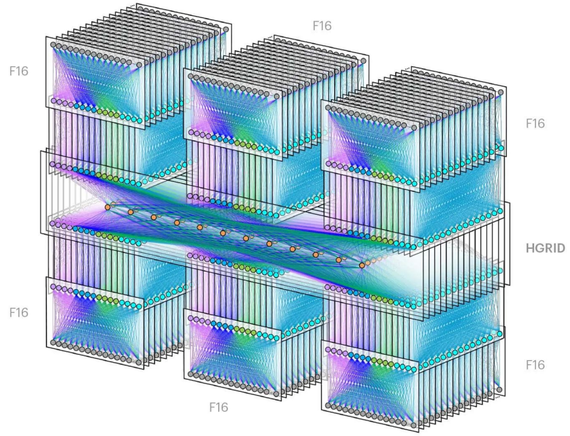
4-Plane Fabric
(https://shorturl.at/YCcd0)



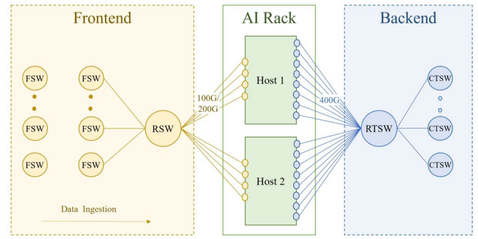


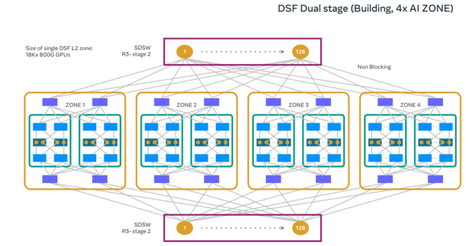
F16 + HGRID
(https://shorturl.at/sxEAV)



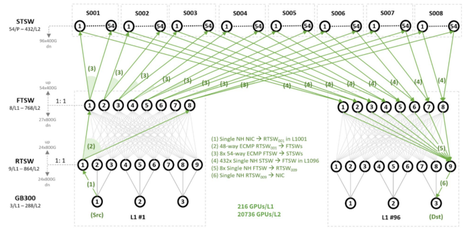


Emergence of AI BE Clusters
(https://shorturl.at/mG7lZ)





DSF
(https://shorturl.at/EGHGO)

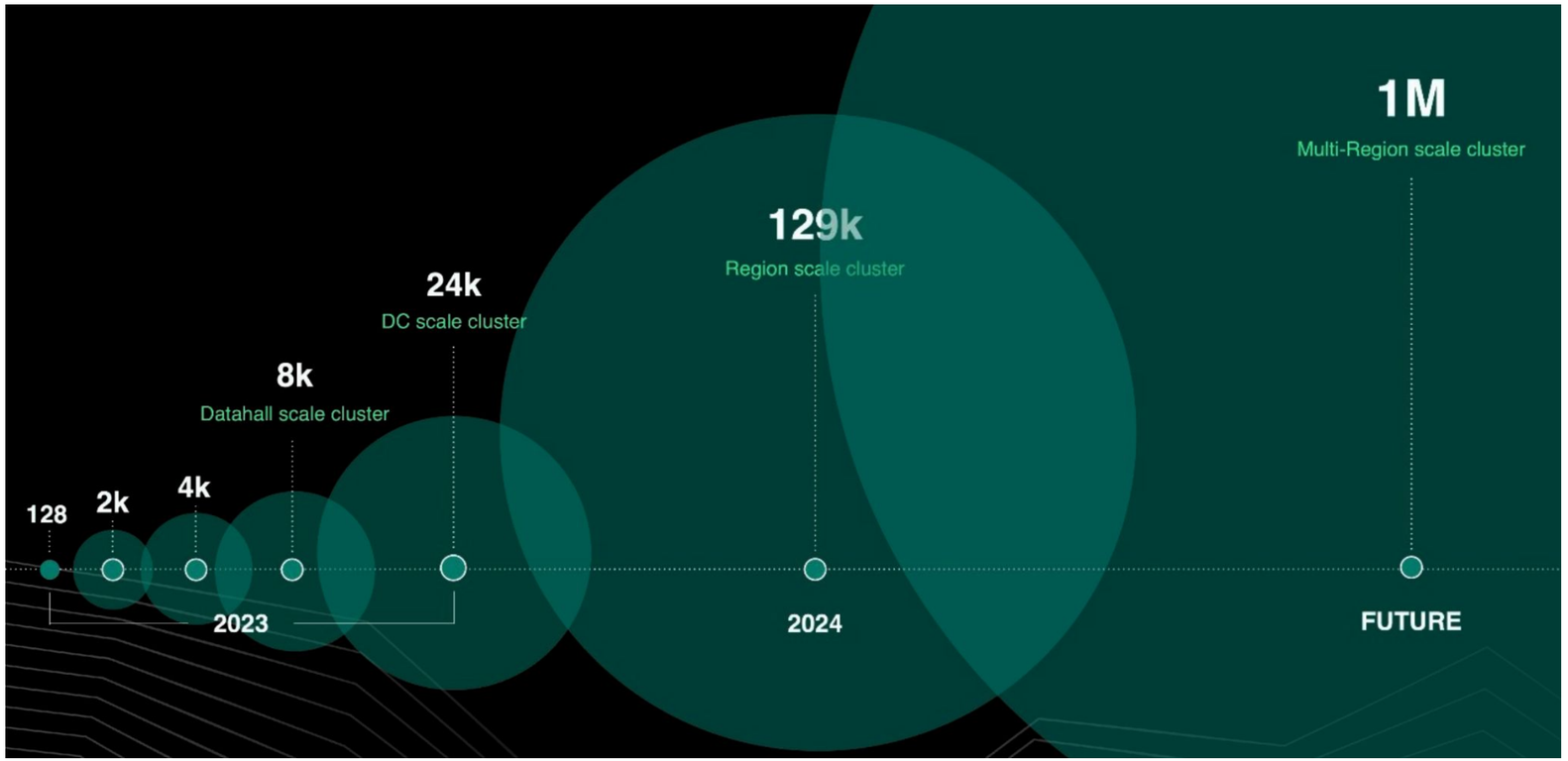


NSF
(https://shorturl.at/EGHGO)

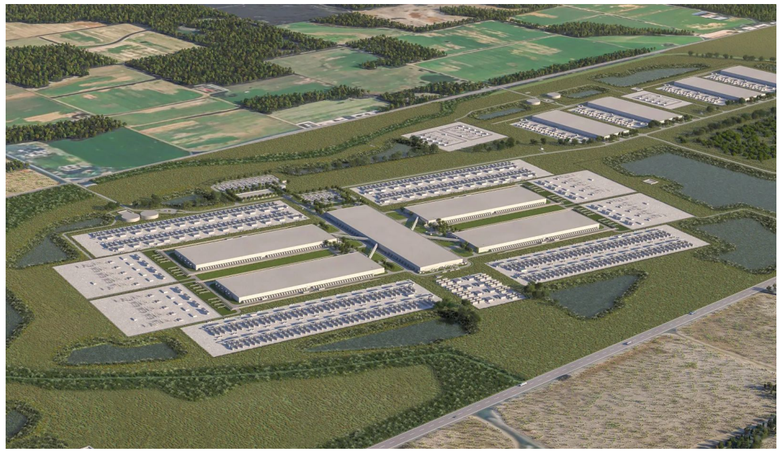
2014 2019 2020+

Challenge 1 - Hyperscale

- Horizontally: **900 DCNs**
- Vertically: AI clusters getting bigger and bigger
  - **100K -> 2 GW AI clusters**



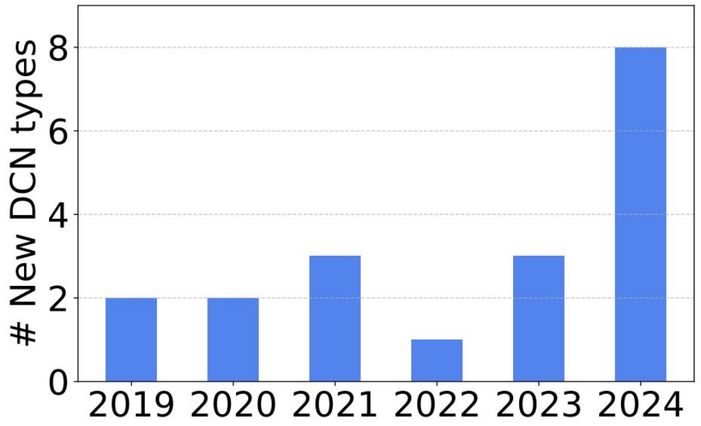
Prometheus 1GW AI Cluster (https://shorturl.at/ofrVd)

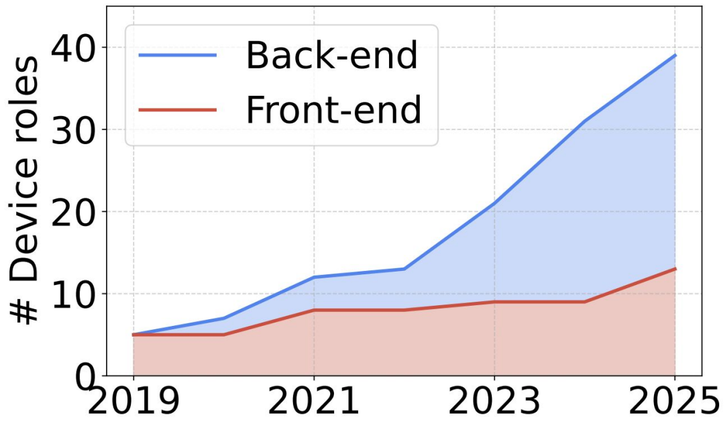


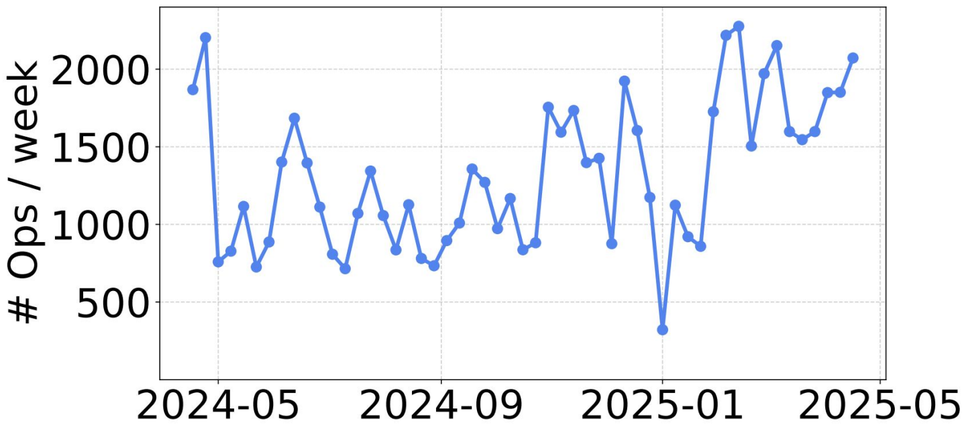
Hyperion 5GW AI Cluster (https://shorturl.at/asBUp)

Challenge 2 – Rapid Evolution and Growth

- Over the past 6 years: **16+ DC network designs** with **30+ new device roles**
- **800–2,000** configuration change operations per week (over 900+ DC networks)



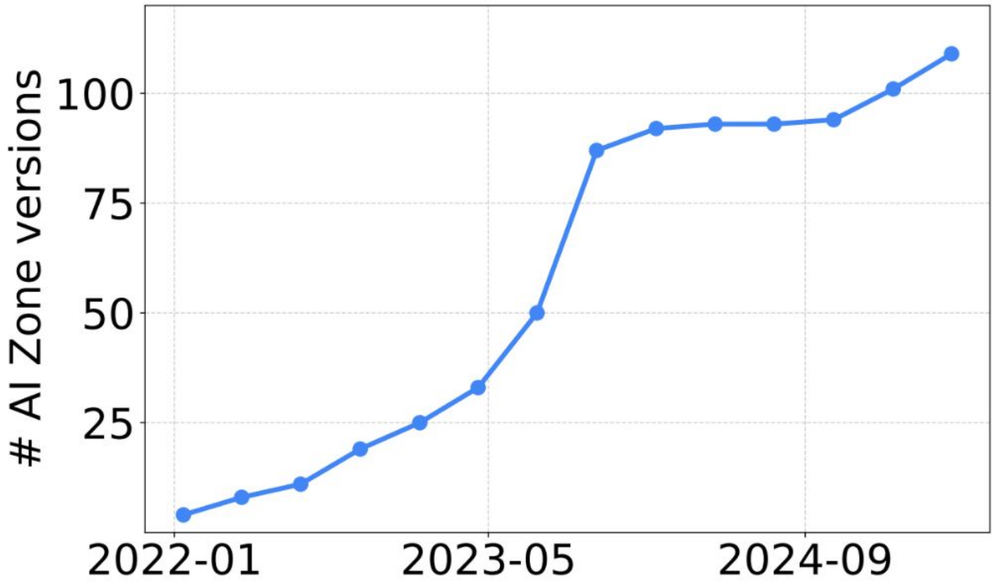




Meta DC Challenge 3 - Heterogeneity

- Diverse vendors, switch OSes, GPU platforms
- **109 AI Zone** versions

GPU Vendors in Meta DC (https://shorturl.at/ofrVd)



Matryoshka at a Glance

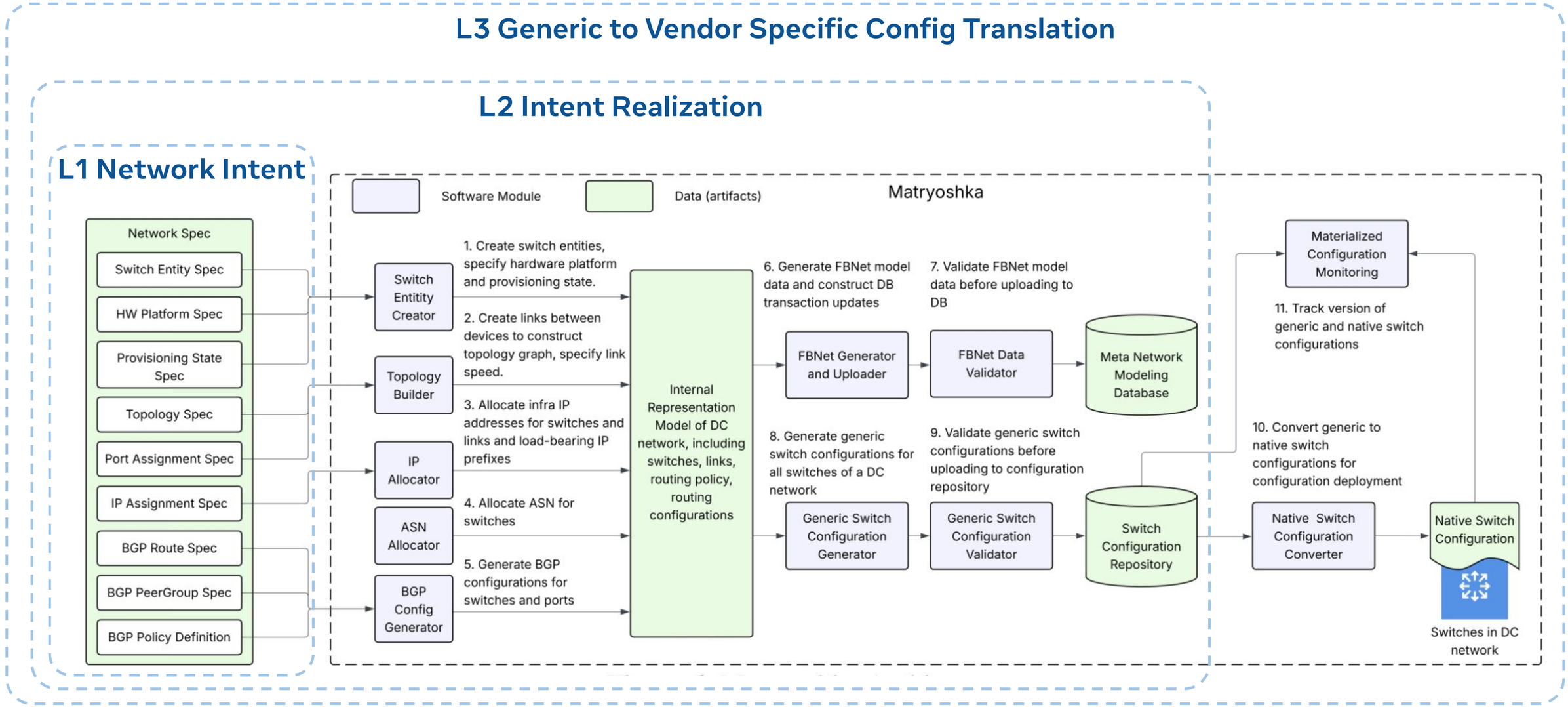
- An **intent-based, model-driven** system
  - Compile network design intents to switch configurations
- Like **nesting dolls**
  - Modular topologies nested within larger topologies;
  - Configurations progressively concretized layer by layer
- **In production over 6 years**: supporting **100%** of Meta's DCN infrastructure
  - Nearly **900 DCNs** across **18 DC design types**
  - **100K+-GPU AI supercluster** powering LLM training

Design Principles

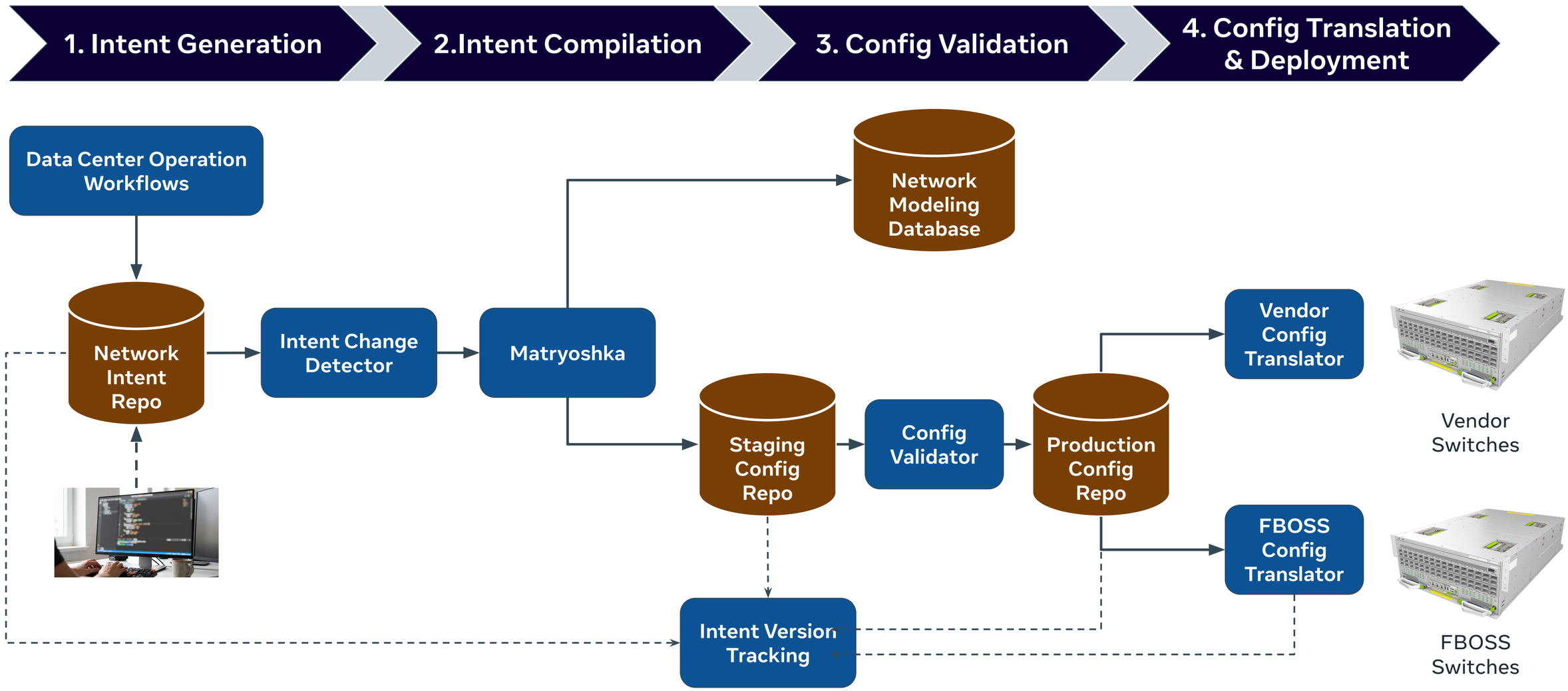
- **Modular Topology Generation**
  - Compose complex topologies from simple building blocks
- **Complete Intent Compilation Workflow**
  - End-to-end from intent to vendor-specific configs
- **Generic Modeling of Design Intent**
  - Thrift-based declarative network intent modeling
- **Deterministic and Stateless Compilation**
  - Target outcomes generated purely based on design intent
  - Scope of migration limited to DB updates
- **Generic (Switch) Configuration Interface**
  - Vendor-neutral interface that abstracts away vendor differences

Matryoshka Architecture

Three layers — like nesting dolls



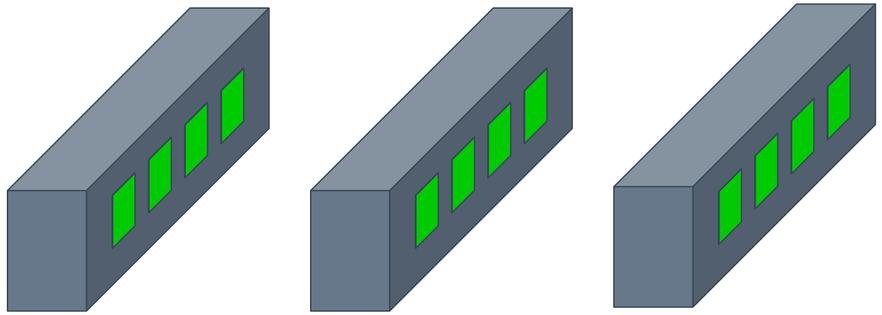
Matryoshka Powered Config Generation Pipeline



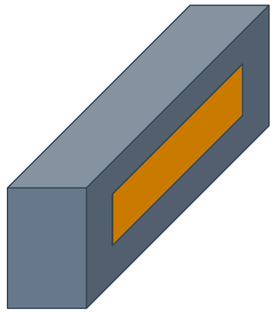
Use Case: 100K-GPU AI Supercluster

Before

3 regular DC building, each having FE services + 4 AI clusters



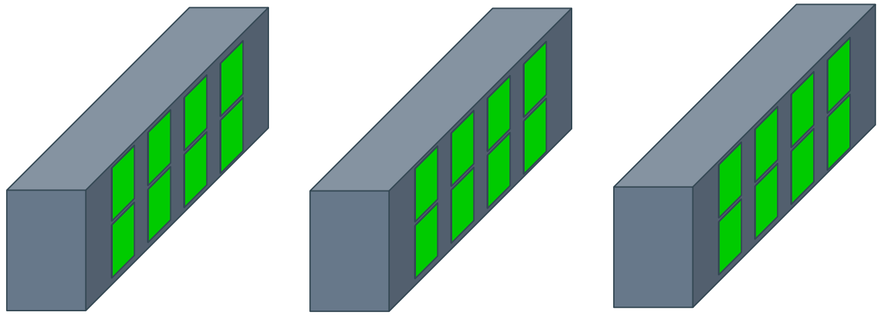
Front End Aggregation



1 storage building

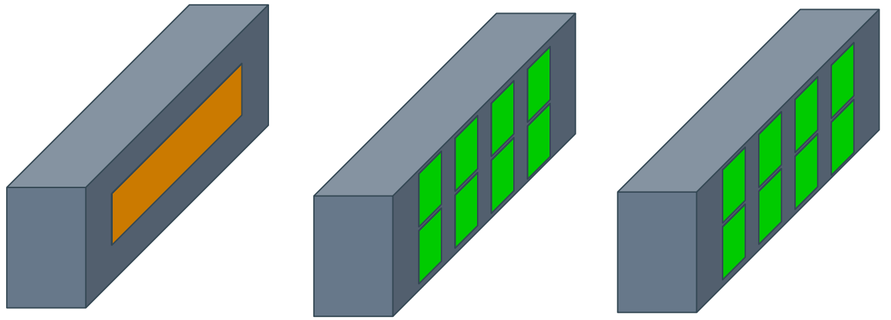
After

5 AI building, each having 8 AI zones. All FE services moved out



Front End Aggregation

Back End Aggregation



1 storage building dedicated to storing training data

Performance Evaluation

- **Correctness**
  - Matryoshka Operations: **95.4%** pass config validations
  - Matryoshka Daily Build: **96.4%** of daily builds released successfully
- **Efficiency**
  - Generic FBNet DB updating algorithm, leading to **70–99%** reduction in transaction time
- **Scalability**
  - Processing time grows **linearly** with network size
- **Extensibility**
  - Extended from 2 → 18 DCN types, 18 → 900 DCNs over past 6 years

Operational Experiences

- **Large Blast Radius**
  - A single bug tears down entire DC fleet -> regression test framework
- **System Complexity and Maintainability**
  - Migration-specific solutions never scale -> generic solutions win
- **Validation Aware Config Deployment**
  - Embed validation results in switch configurations
- **Evolving to Meet New Challenges**
  - Further reduce NPI timeline: 1 year → 1 quarter → 1 month

Conclusion and Takeaways

- **DCN config realization is a first-class systems problem**
- **Modular, stateless compilation scales**
- **The AI era makes this approach essential**

Thank You!

Matryoshka — the first production-grade DCN design realization system

- **6 years** in production · **900 DCNs** · **18 DC Types** · **100K GPUs**
- **100%** of Meta's DCN infrastructure

Questions?