Advanced Models and Future Directions

- Beyond MCPs: full affordance-driven design
- Rich modeling for **intent, context, and interaction**
- Path toward **adaptive, agent-ready ecosystems**

Donella Meadows on Systems Thinking

- Anchor for systems modeling and leverage points
- Her voice brings clarity about how small shifts in design can change the behavior of whole systems — exactly what ALPS affords.

Why Move Beyond MCPs

- MCPs are **adapters**, not complete solutions
- Advanced models describe **system-level behavior**
- Long-term resilience comes from **affordance-first design**

Affordance Modeling with ALPS

- ALPS = **Application-Level Profile Semantics**
- Describes **states, transitions, and affordances**
- Lightweight, machine-readable, and intent-focused

TypeSpec and Smithy

- TypeSpec: modeling language for APIs, schemas, services
- Smithy: extensible specification with strong typing
- Both support **affordance and capability modeling**

Benefits of Advanced Models

- Improved **discoverability and adaptability**
- Clearer **contracts for machines and humans**
- Easier evolution*of large ecosystems

Challenges to Adoption

- Steeper learning curve*for teams
- Tooling still **maturing**
- Requires **ecosystem-wide buy-in**

Future of Adaptive Ecosystems

- APIs as **living environments**, not static products
- Continuous **evolution guided by usage telemetry**
- Inspired by **biological and ecological systems**

Vision of Agentic Systems

- Agents discovering and composing APIs on the fly
- Self-optimizing orchestration across domains
- Emergent behaviors enabled by **affordance modeling**

The Road Ahead for Designers

- Start with MCPs to bridge the gap
- Adopt ALPS, TypeSpec, or Smithy for deeper modeling
- Build API ecosystems that can **learn and adapt**

Looking Ahead

- Unit 9 wraps up the journey
- Summarizes the **five shifts + MCPs + advanced models**
- Prepares participants with **next steps and resources**