Rethinking Pipelines, techniques, and Standards for AI-Native

Vishnu

Building an AI-native Future of Industry

Beijing, China , 21 September 2026

Aim of the talk

- Shares an evolving perspective on where AI-native technology stands today and where it may be headed
- Shaped by close involvement with FG-AINN
- Offered as a mentor's working notes
- Intent is to open a conversation
- You are encouraged to challenge the framing, push on the open questions.

- ***Not*** *an official account of the Focus Group's progress or a critique of its output, and several of the ideas raised are questions the speaker is still working through rather than settled answers.*

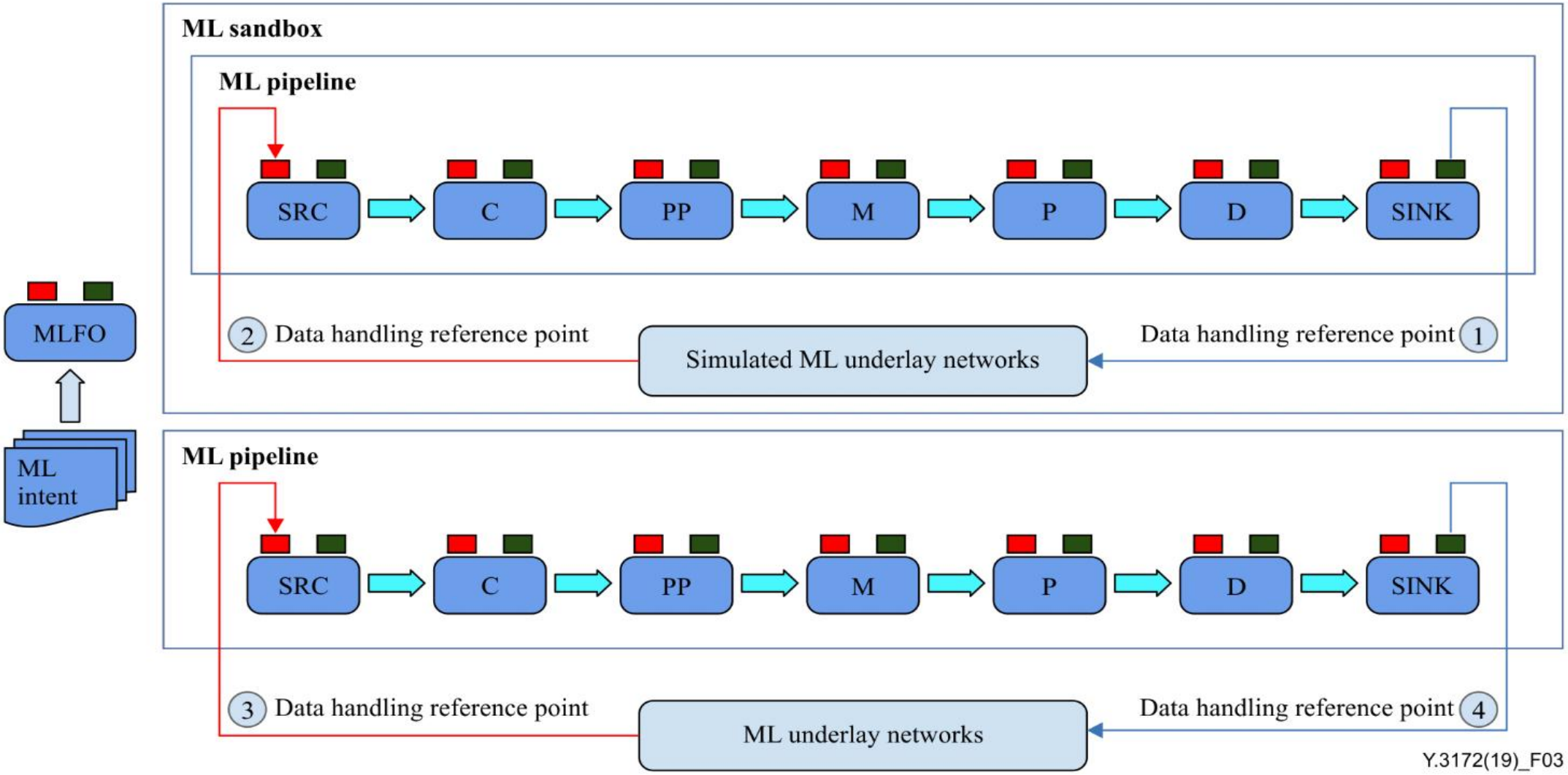
Covered topics

- What are the core principles in AI native networks?
- What is expected from each pipeline node?
- What are the AI native techniques to be applied in the network?
- How to look at current use cases, architectures (and PoC) with AI native techniques in mind?
- What are the considerations on standards and opensource for AI native?

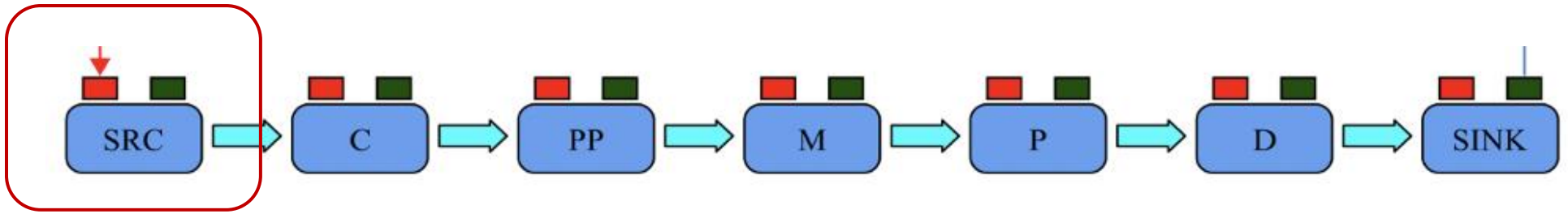
What are the core principles in AI native networks?

1. "AI native is a design principle”
2. ”At design stage, take existence AI for granted”
3. ”Design with an aim to (1) Provide AI and (2) consume AI”
4. ”Network functions should facilitate AI training and finetuning”
5. ”Programmability of AI" – in the network - dynamically create the pipeline nodes based on the intent (represents the use case)

What is expected from each pipeline node?

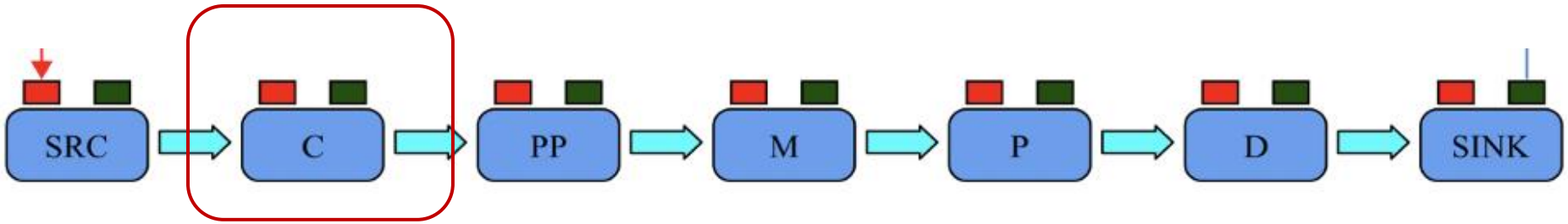


What is expected from each pipeline node?



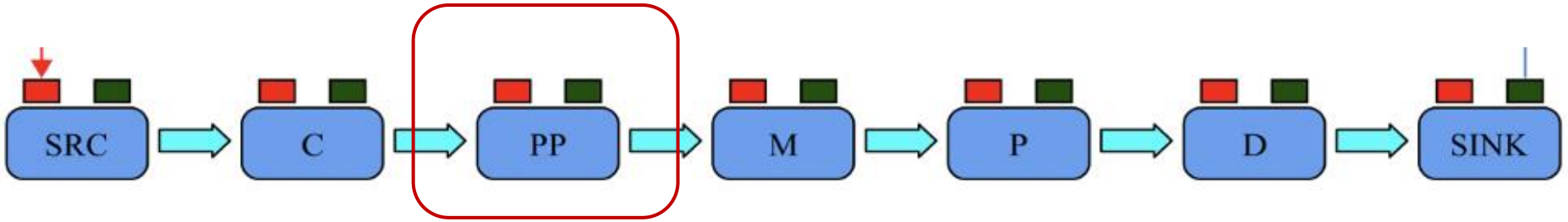
- SRC (source):
  - providing dynamic source of data
  - programmable (API based, request and response) source of data
  - discoverable source of data
  - (given the use case (intent), where is my data?)

What is expected from each pipeline node?



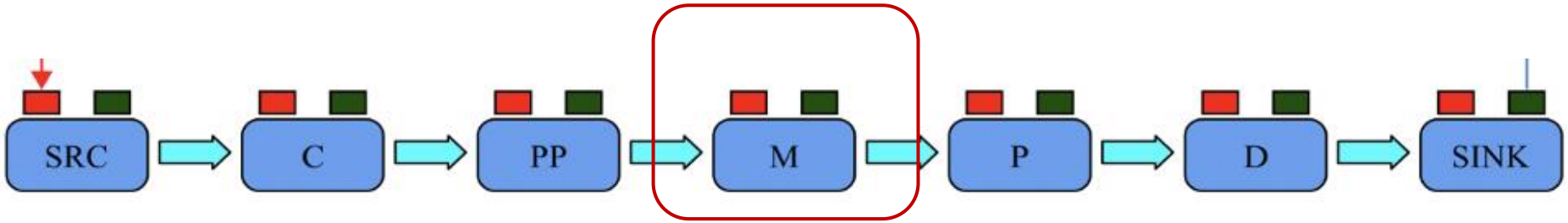
- Collector:
  - time alignment.
  - multi-modal fusion.
  - nothing specifically AI native about it.

What is expected from each pipeline node?



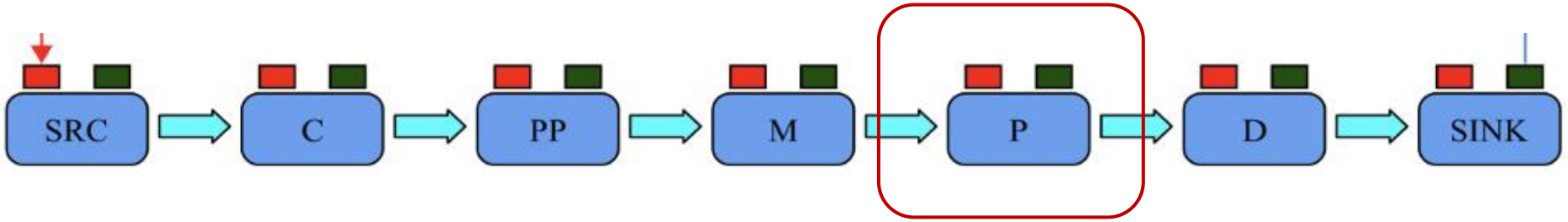
- Pre-processor:
  - no hardcoding
  - mapping between intent (use case) - src - policy (specific).
  - standard template - and - policy-to-code.

What is expected from each pipeline node?



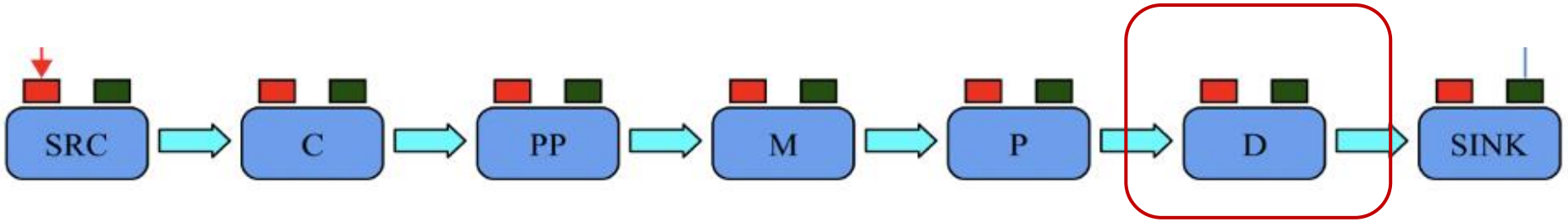
- Model node:
  - with added metadata
  - training/finetuning is enabled in the network
  - RAG KB (interfacing the M with the RAG)
  - create the RAG on the go (sandbox)
  - plug in the RAG dynamically (sandbox)
  - local context (with datasets, policies) e.g. operators from different regions, can use local RAGs.
  - serving/deployment : enabling scaling of models
  - integrated marketplaces (Y.3176)

What is expected from each pipeline node?



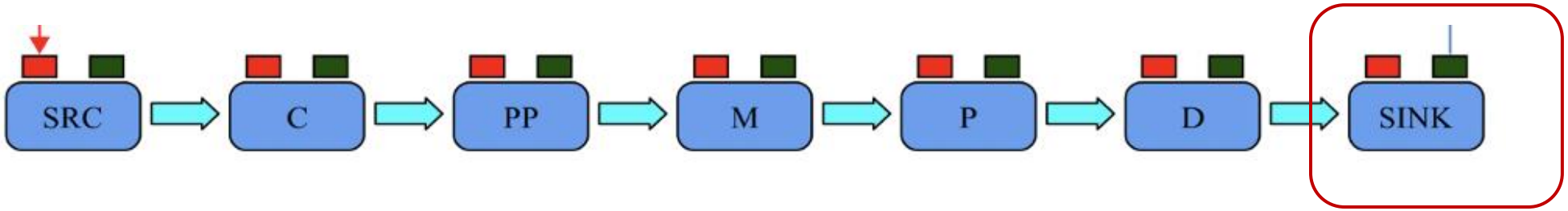
- Policy node:
  - no hardcoding
  - mapping between intent (use case) - src - policy (specific).
  - standard template - and - policy-to-code.

What is expected from each pipeline node?



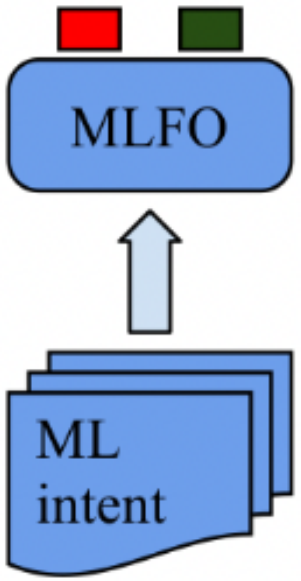
- Distributor:
  - plug in SINKs dynamically.
  - based on intent.
  - programmable
  - on-demand for the SINK.
  - metadata added (e.g. for trust)

What is expected from each pipeline node?



- SINK
  - create dynamically
  - based on intent.
  - 3173: demand mapped.
  - user action: configurable (standard template)
  - code gen

What is expected from each pipeline node?



- MLFO (ML Function Orchestrator)
  - monitoring
  - optimization.

Studying current use cases, architectures and PoC with ai native techniques in mind.

- FG-AINN-usecase-001:
- the proposed AI-native architecture is designed from the outset to integrate distributed intelligence across all layers—particularly through the synergistic operation of the Monitoring System (MS), Analytics Engine (AE), and Decision Engine (DE), and tight integration with actuators (ACT) for taking local closed loop decisions [b-Blanco2023]
- continuous real-time data collection from geo-distributed sensors, mobile applications, and IoT devices, enabling situational awareness even in volatile or degraded environments
  - applying context-aware AI models to detect anomalies, forecast risks, or assess resource availability
  - distributed AI architecture.

Studying current use cases, architectures and PoC with ai native techniques in mind.

- FG-AINN-usecase-002:
- Network for AI use cases, the network monetizes its available resources and related capabilities (i.e., data collection and storage) offering new value-added services to a consumer through a dedicated interface.
  - AI-native telecommunication networks have intelligent mechanism for determining the AI model (and its configuration) to use for a requested AI inference as a service offered by the AI-native telecommunication network as a value-added service.
  - AI-native telecommunication networks have intelligent mechanism for monitoring and optimizing the AI inference workload and its split within the AI-native telecommunication network and its environment;

Studying current use cases, architectures and PoC with ai native techniques in mind.

- FG-AINN-usecase-003:

- intelligent mechanism for dynamically split the AI Training workload between the AI-native telecommunication network and its environment, based on resource availabilities and AI Training;
- intelligent AI training mechanism that construct AI models able to work in different configuration to trade-off AI service performance with network resource usage
- AI-native telecommunication networks support data collection for AI Training

Studying current use cases, architectures and PoC with ai native techniques in mind.

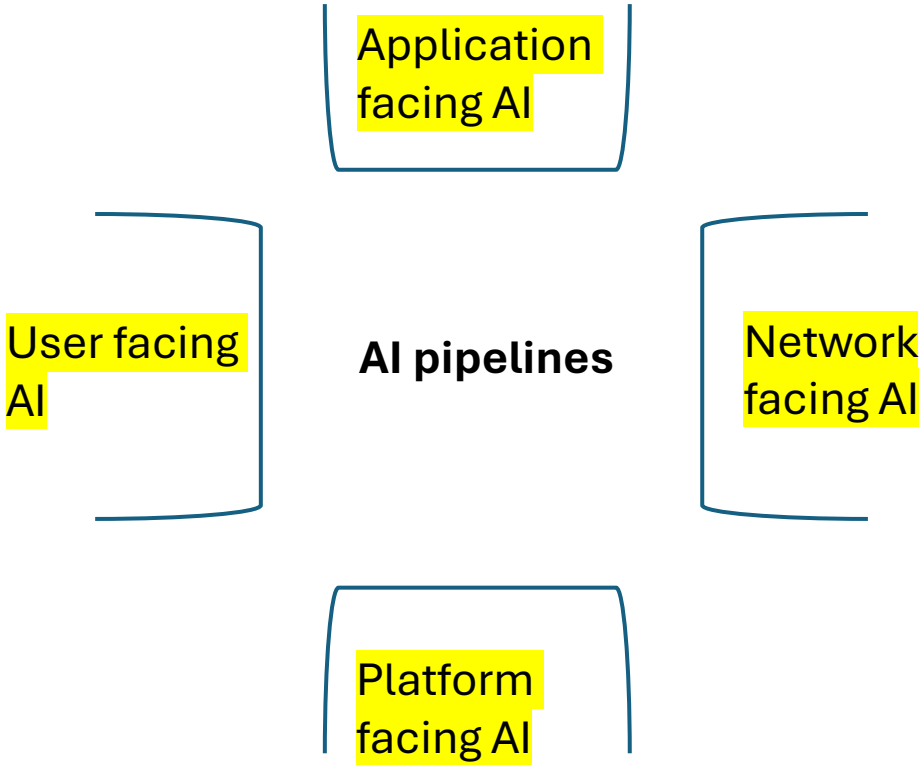
- FG-AINN-usecase-005:
- AI functions that can classify/fetch/pre-process/ allocate data are deployed at key nodes to undertake the tasks of resource allocation and transmission efficiency improvement, to support the deep integration of all aspects of the network and the optimization of analysis functions.
-

  - dynamic knowledge base and LLM analysis functions of AI-native networks can output analysis results that are more easily recognized and accepted by humans

Studying current use cases, architectures and PoC with ai native techniques in mind.

- FG-AINN-usecase-006:
- intent recognition and task decomposition
- transformation of multi-modal information
- provide AI models specific to different tasks, and regularly fine-tune or update the AI models
- provide knowledge bases containing knowledge from different fields to train AI models.

Studying current use cases, architectures and PoC with ai native techniques in mind.



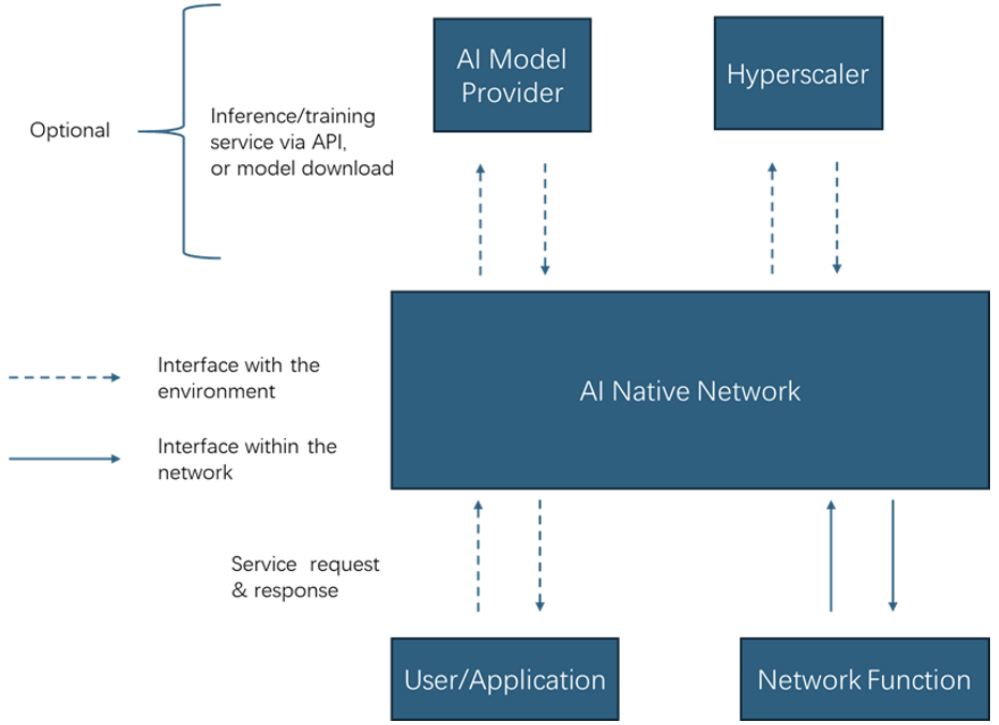
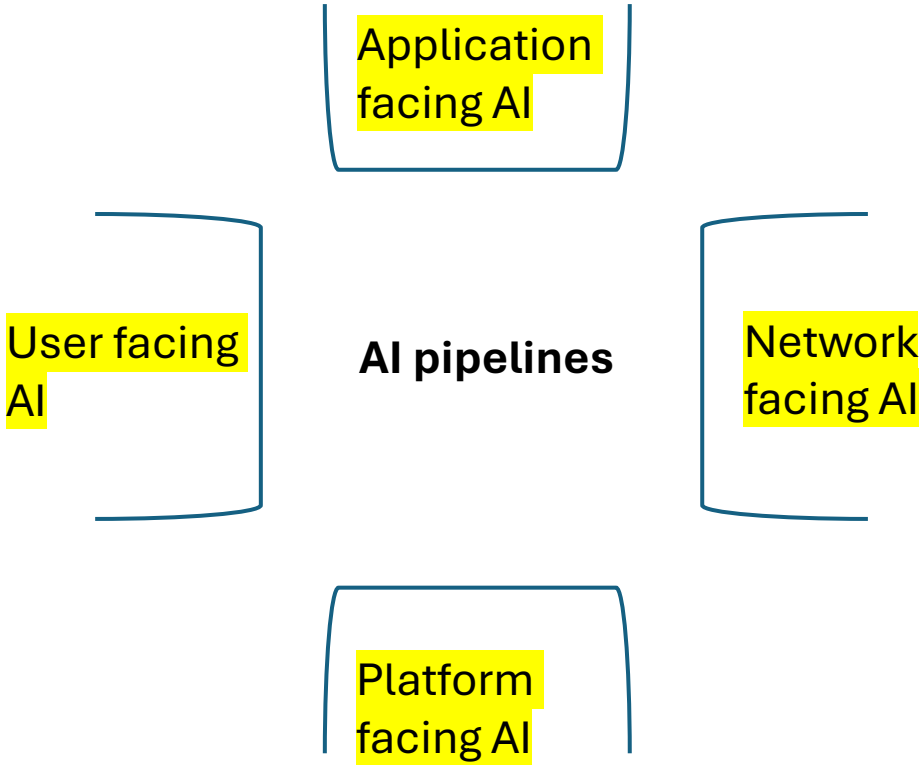


Figure 1. AI service deployment within the AI native network.

FG-AINN-O-034

AI native techniques



- User facing AI
  - (local language LLMs)
- Platform facing AI
  - (network 4 AI: scalable AI infrastructure)
- Network facing AI
  - (AI for network)
- Application facing AI
  - (Knowledge Bases, local RAG)

What are the considerations on standards and opensource for AI native?

	AI Native considerations	Standards and opensource gaps
1	Experimentation in sandbox	1. Mapping between intent (use case) – Source of data – specific policy to be applied. 2. Discovery of data source. 3. training/finetuning of models. 4. standard template - for - policy-to-code
2	Adaptation and evolution of AI agents	1. Metadata and repository of open agents. 2. Sandbox validation of agents 3. Closed loop monitoring and optimization of agents
3	Role of open knowledge base	1. create the RAG on the go 2. Validate and plug in the RAG dynamically
4	Role of natural language prompts (user facing)	1. Intent processing, 2. Configurable user actions with standard templates 3. code generation options and templates for actions.