DA-ITN: Data Aware Inference & Training Network

A data descriptor and topology enabled network for AI

Arashmid Akhavain arashmid.akhavain@huawei.com
Hesham Moussa hesham.moussa@huawei.com

Huawei Technologies Canada

IETF 123 - Madrid
July 2025

Background

- AI/ML heavily relies on data from various sources
  - In the centralised approach large volumes of data are collected and processed in central clouds.
  - Decentralised/distributed learning methods aim to provide model performance comparable to that of the centralised approach of learning while improving scalability, enhancing privacy, and potentially reducing computational/storage requirements.
  - Decentralised methods generally employ the model-follow-data paradigm over what is referred to as a knowledge-sharing network (KSN) where
    - The collective data in all network nodes constitutes the global data corpus (i.e. A global repository of all available data in the network).
    - Models are moved to relevant data nodes for training.

Background

- The model-follow-data paradigm attempts to establish optimal rendezvous points in KSNs where models and data interact based on
  - Data location and properties (e.g., age, size, type, quality, dynamics, distribution, etc.)
  - Network topology and availability (both at KSN and at their underlying communication network).
  - Resource availability.
  - Mutual trust between model and data (AI control WG???).
  - Model architecture (RNN, CNN, Transformers, etc.).

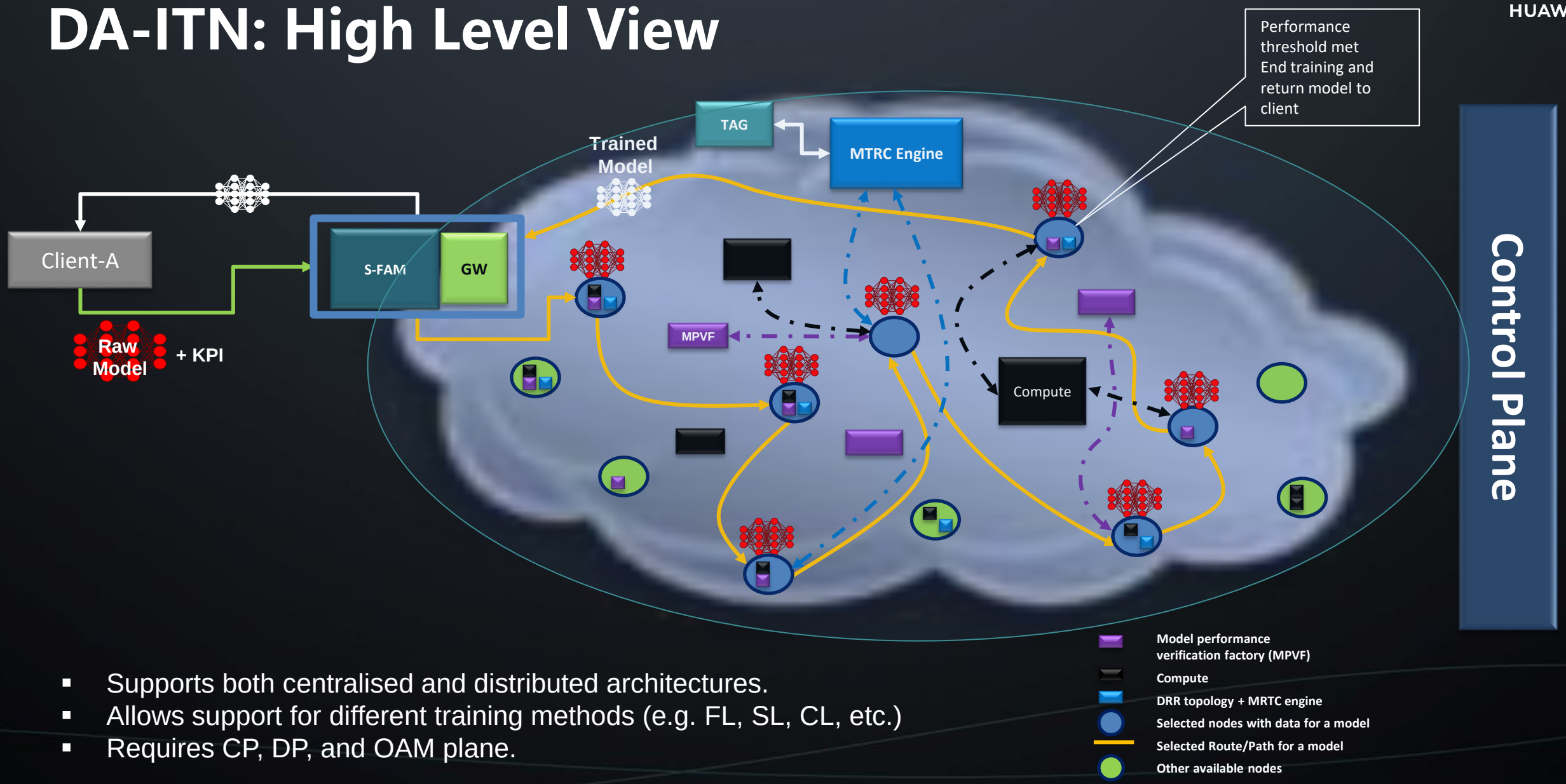
Initial Motivations

- Consider a large data corpus distributed across various equipment.
- Model requiring training usually come with different set of objectives and KPIs.
- It can be argued that there is an optimal subset of data that is sufficient for training each specific model.
  - A model can avoid training on two nodes with similar knowledge content.
- Therefore, there is a need to carve out this optimal subset from the global canvas for the purpose of efficient model training w.r.t communications, compute, storage, etc.

Data Aware Inference & Training Network (DA-ITN)

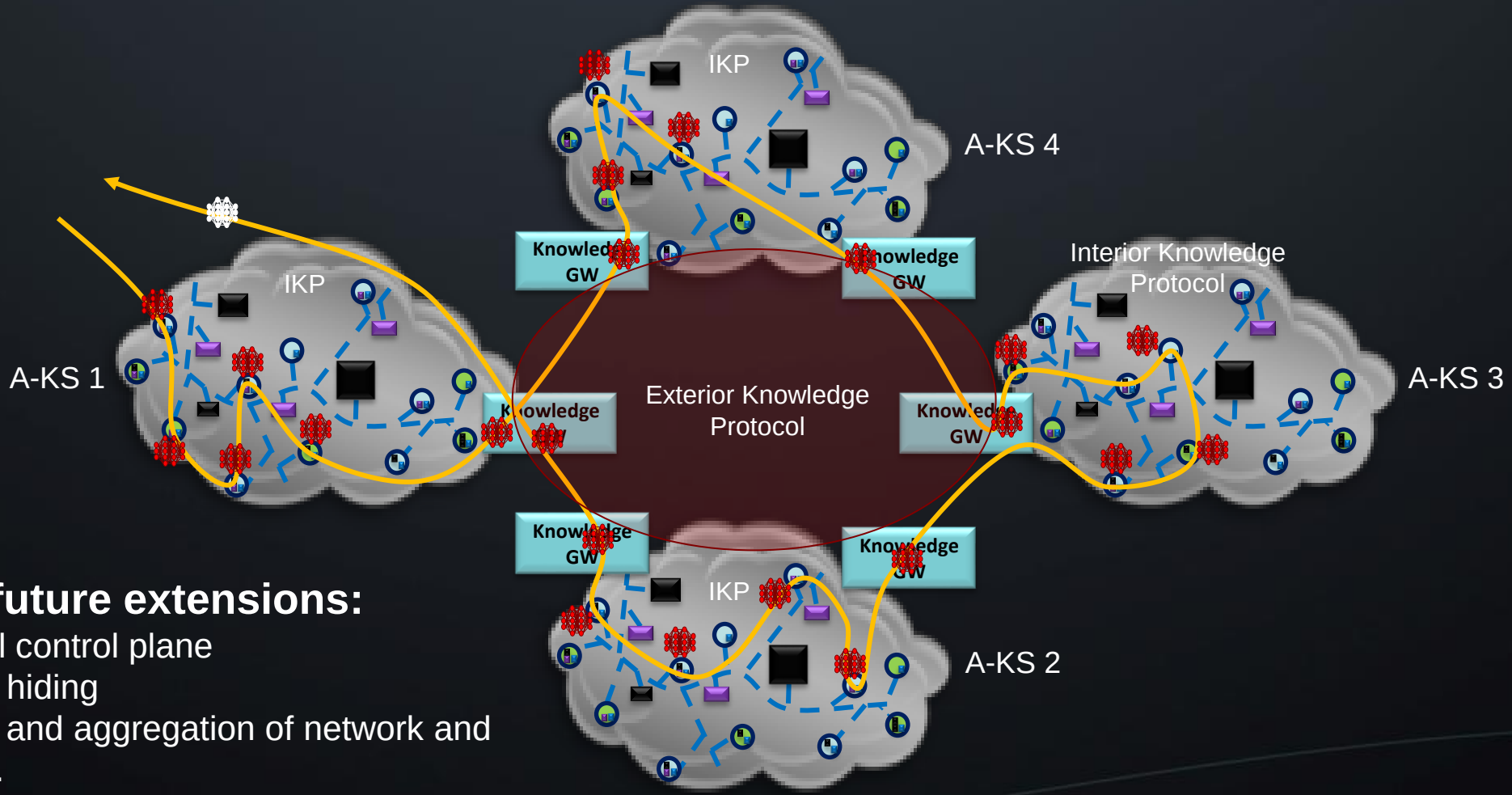
- Some of the **initial** envisioned objectives of DA-ITN for model training are:
  - Optimise distributed training services in KSNs by continuously adapting to the dynamic conditions of both KSN and its underlying communication network.
  - Compute optimum training route/path for AI models in KSNs.
  - Privacy protection – Providing data and model privacy standards for secure collaboration. i.e. mutual trust between models and data.
  - Service Feasibility Assessment - Determining whether the target model KPIs can be achieved under current constraints without starting the training.
  - Adaptive Training Methodology – Identifying suitable training method for a model.
  - Real-time Monitoring – Providing methods for tracking training progress and feedback.

DA-ITN: High Level View



- Supports both centralised and distributed architectures.
- Allows support for different training methods (e.g. FL, SL, CL, etc.)
- Requires CP, DP, and OAM plane.

DA-ITN: Hierarchy, Abstraction, and Aggregation



Potential future extensions:

- Hierarchical control plane
- Information hiding
- Abstraction and aggregation of network and information.
- Support for Autonomous Knowledge System (A-KS).

Few Observations

- Decision making based on attributes such as data size, data type, nodes' compute, storage, and communication capabilities partially address the problem, but they fail to consider the role and impact of the knowledge embedded within the data.

- Identifying relevant data w.r.t models is crucial regardless of the underlying employed training strategy (centralised or distributed). Some potential benefits include:

  - Optimal amount of training data and reduction in volume of data to be collected.
  - Lower pre processing, communications, memory and storage cost.
  - Improved training time, and model performance
  - Lower power consumption.
  - Better adaption to data dynamics over time.

How do we identify relevant data for each model?

Borrowing ideas from networking ???

- Link state information is a well understood idea in networking. It provides a unified method to network nodes enabling them to understand and digest link related information.
- Can we convey data related information in a similar fashion? Is there a unified construct that can enable us to express **information about data**?
- Is there a way to quantify data attributes such as data quality, data age, data dynamics/evolution and rate of change over time, data variance, and its relevance for a given AI model?

What would be useful as a set of data attributes?
How do we build such a construct?

Data descriptors/Data state information

- A data descriptor can be viewed as a function taking raw data as input and generating expressive output that enables us to differentiate datasets w.r.t each other based on identified essential attributes.
- Is there a function or as set of functions to serve this purpose?
- Is there a relationship between the model architecture/characteristics, data categories, and other data characteristics that can be used to derive this function?
- How do we do this without revealing too much or any of the raw data?
- A quantitative and qualitative data descriptor can be used to differentiate data from various nodes per model.

How do we determine an optimal knowledge set for a given model?

Essential factors

Data dynamics

- **Data evolution:** Changes in the state of data in datasets over time.
- **Data variance:** Understanding data scope w.r.t a per model-based reference point. (e.g. what does black cat mean? There is a range and ambiguity to be resolved here.)
- **Data age:** Relevance of data could be tightly coupled to data generation time.

➢ **To analyze and quantify data dynamics, it is essential to express and describe data from the models' perspectives.**

Model architecture

- Model characteristics such as model size, capacity, structure, and parameters are important and must be taken into account when interpreting data measurements.
- Importance of some factors become clear only after the training begins.
- Data characteristics cannot be considered in a vacuum and are coupled to models' architectures and training objectives.
- What factors need to be considered during training? **How do we discover this coupling with no or minimal training?** What is the measure of this coupling?

➢ **Mindful selection of training data from the corpus w.r.t model architecture can enhance performance. Data descriptors should serve a supportive and facilitating role.**

Essential factors

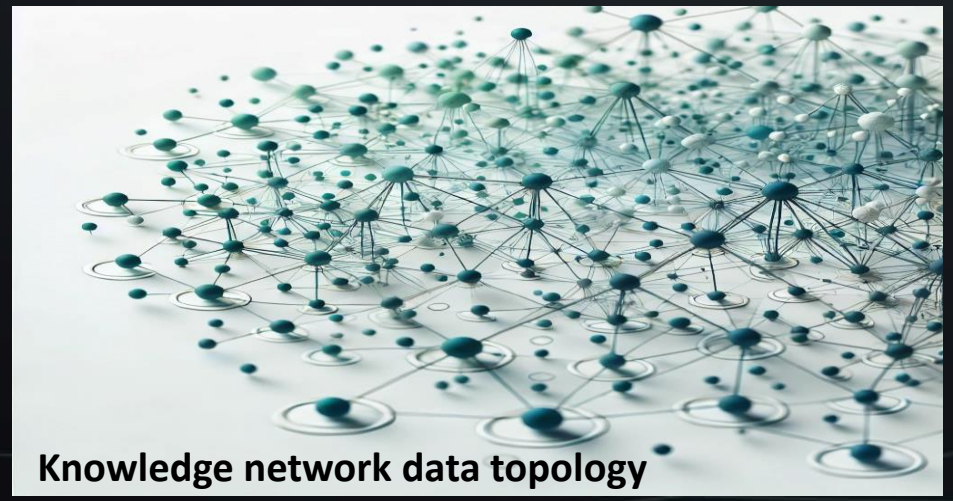
- **Training methods and algorithms**
  - There is also a coupling between training methods and data dynamics.
  - The rate of change in data for example can dictate the training method.
    - Federated learning lends itself better to faster data variation rate.
    - Sequential learning on the other hand adapts better to slower rate.

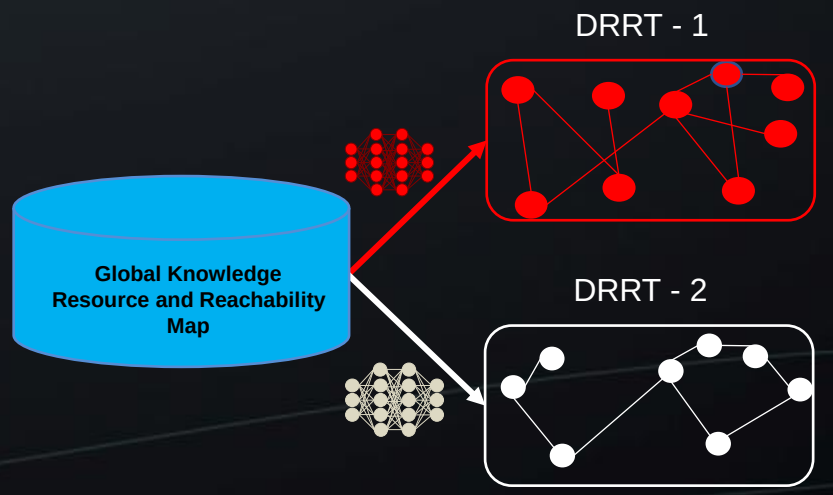
  - Performance of training algorithms is also dependent on data characteristics.
    - This relationship so far only reveals itself during the training.
    - How can data descriptors help expressing data characteristics that facilitate this task without or minimal training?

    - **Data descriptors should include attributes that facilitate the alignment of training methods and algorithms with the most suitable training datasets.**

From Data descriptors to Data Resource and Reachability Topology

- **DRRT**
  - A **dynamic** and **model specific** structure derived from a globally collected unstructured knowledge (data corpus), resource, and reachability database.
  - Holds data related information such as data type, **quality**, volume, age, and **dynamics**.
  - Monitors resource availability and reachability status of the nodes in the KSN.
  - MTRC engine in DA-ITN uses it to make accurate model steering decisions to fulfill model training requirements.
  - Centralised training can use it to target relevant data w.r.t the model under training.





Inference

- DA-ITN can have access to information that can help optimising inference performance such as: Training records, inference history, model performance, model QoE, network resource availability etc.
  - This information can be readily available in scenario where model is trained by DA-ITN.
  - Alternatively, the model can make this information available in scenarios where the model is trained outside of DA-ITN.
  - **We refer to this information as model descriptors.**
- DA-ITN can use model descriptors for
  - Inference load balancing
  - Inference routing
  - Inference chaining
  - Optimised model deployment, movement, instantiation, and mobility
  - Hot standby and model upgrade
  - Security, trust, privacy ????
- DA-ITN can provide a platform for billing, accounting, etc. associated with the inference purposes.
- DA-ITN can provide a mechanism for providing differentiated services (e.g. match models to query budget expressed by the application).

Agentic networks

- **DA-ITN can provide a platform for agent to agent collaboration, interactions, and communications.**

- **DA-ITN can facilitate agent instantiation, deployment, discovery, operations, status monitoring, life cycle management, etc.**

- **Having access to agent capabilities, DA-ITN can provide a ground for**
  - Implementing VPN like services for agents
  - Agent billboards and hourly usage cost
  - Agent operation scheduling and service queuing
  - Potential requirements for knowledge transfer and sharing between agents
  - Supporting wake up calls to dormant target agents

Concluding remarks

- **DA-ITN provides an all encompassing ecosystem for AI model training, inference, and agent to agent collaboration and interaction, operations, billing, and maintenance.**
- **DA-ITN is empowered by model descriptors, data descriptors and topologies.**

- **Data descriptors' design objectives:**
  - Convey data information from KSN nodes.
  - Help to pinpoint valuable knowledge for use by the model.
  - Enable the creation of comparison metrics allowing us to distinguish data across various nodes.
  - Lay the foundation for developing **Knowledge Network Data Topologies**.
  - Help optimize training and inference processes while reducing cost and preserving privacy.
  - Additional potential advantages and applications:
    - Data descriptors can provide a vehicle to express data producers' consents (AI-Control WG???)
    - Enable negotiations between data producer and consumers.
    - Can reduce cost for centralised training methods

- **Model descriptors' design objectives:**
  - Convey model and potentially agent information.
  - Help matching queries to appropriate models.
  - Assist in establishing the basis for providing differentiated services.
  - Provide a foundation for inference chain discovery
  - Help optimize training and inference processes while reducing cost

IEEE network magazine:
Distributed Learning and Inference System: A Network Perspective

Thank You.