

*we need a better name than solar **corona**, which has not been a good name since this Jan....*

# Solar Corona (Edge AI) Intro

# why AI@Edge?

- Data Privacy
- Lower latency than cloud
- large volumes data on edge, expensive to transfer to cloud, and unnecessary sometimes due to temporal and spatial locality.

# Challenges of AI@Edge

- few-shot samples per edge for training, cold booting, hard to converge
- geo-distributed dataset across edges
- Non-I.I.D data across edges, the performance of universal AI model degraded on edge
- resource constrained on edge

# what we proposed

- ***It IS:***

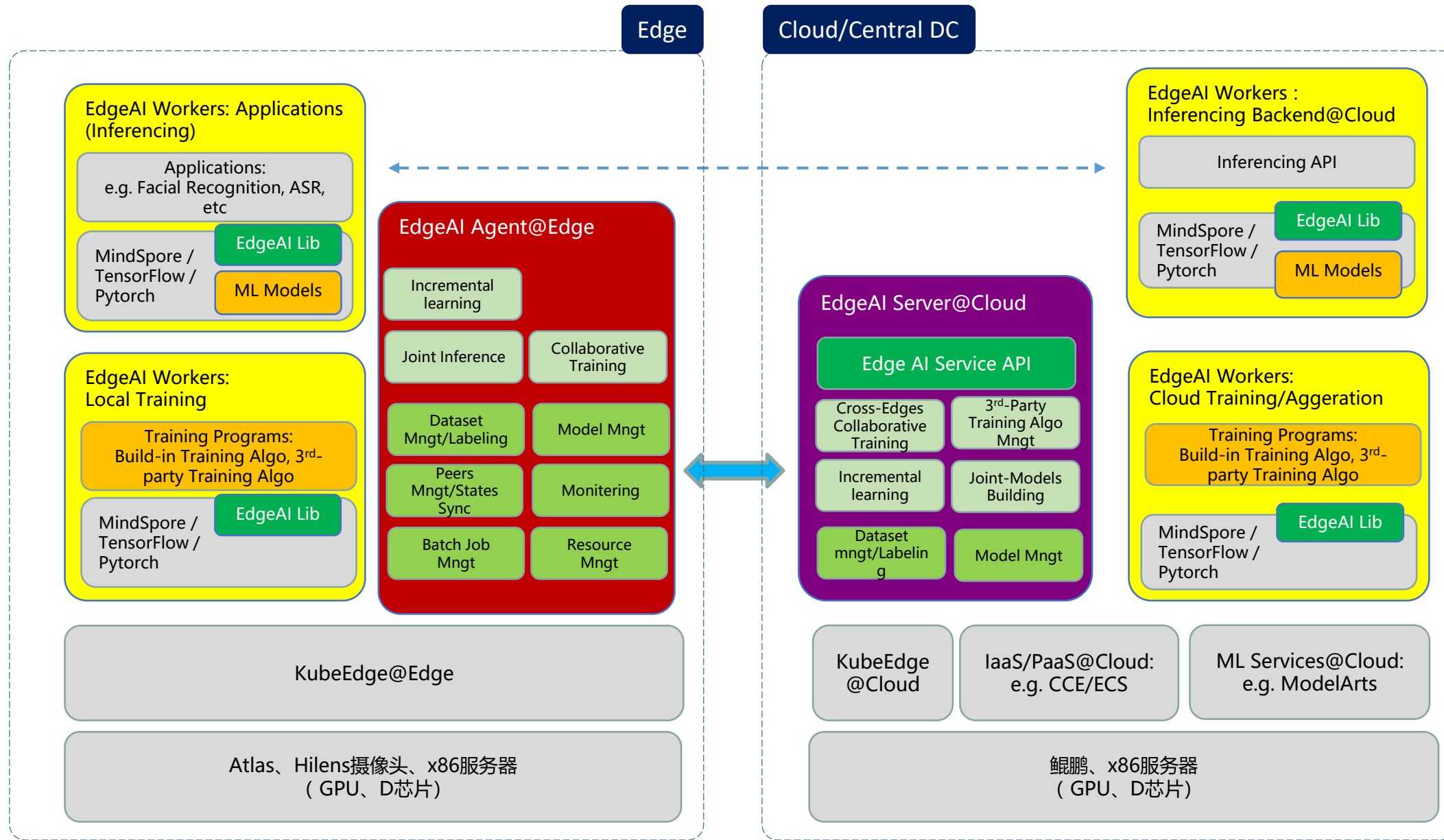
- **an edge-cloud collaborative ML service based on KubeEdge**, including a server on cloud, an agent at the edge, workers to run inferencing or training and a lib for interoperability with existing ML frameworks,
- **some edge-cloud collaborative ML features**, including: joint inferencing, incremental learning and collaborative training (aka federated learning)
- to help ***Domain-specific AI Developers*** to build and publish edge-cloud collaborative AI services/functions easily
- to help ***Application Developers*** to use edge-cloud collaborative AI capabilities.

- ***It's NOT:***

- to re-invent existing ML framework, i.e., tensorflow, pytorch, mindspore, etc.
- to re-invent existing edge platform, i.e., kubeedge, etc.
- to offer domain/application-specific algorithms, i.e., facial recognition, text classification, etc.

# Service Architecture

- **Server on Cloud:**
  1. uniportal of EdgeAI,
  2. across-edges coordination
- **Agent @ Edge:**
  1. local controller
  2. manage local dataset and models
- **Workers:**
  1. do inferencing or training, based on existing ML framework;
  2. launch on demand, imagine they are docker containers;
  3. different workers for different features;
  4. could run on edge or cloud.
- **Lib:**
  1. expose the Edge AI features to applications, i.e. training or inferencing programs.



# Service API and Lib interfaces (under discussion)

- Still under discussion
- 2 examples for incremental learning and Federated learning, which could be significantly changed at the end

CloudBU CTOOffice / edgeai / Federated\_Learning

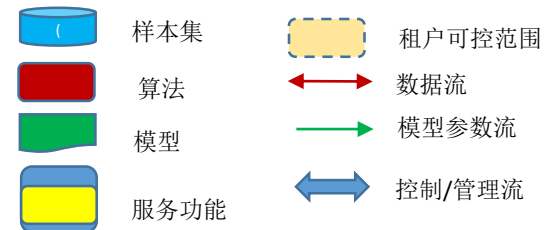
job
post /jobs
get /jobs
get /jobs/{job_id}
put /jobs/{job_id}
delete /jobs/{job_id}
get /jobs/{job_id}/dataset
get /jobs/{job_id}/models
post /jobs/{job_id}/join
post /jobs/{job_id}/start
post /jobs/{job_id}/stop
model
get /models
get /models/{model_id}
dataset
get /dataset
get /dataset/{dataset_id}
algorithm
post /algorithm
get /algorithm
get /algorithm/{algorithm_id}
put /algorithm/{algorithm_id}
delete /algorithm/{algorithm_id}

/ edgeai / Incremental\_Learning

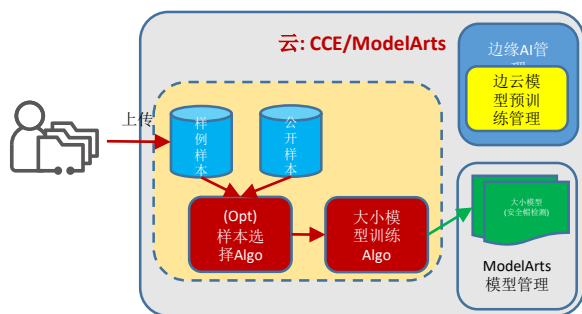
  

job
post /jobs
get /jobs
get /jobs/{job_id}
put /jobs/{job_id}
delete /jobs/{job_id}
get /jobs/{job_id}/status
get /jobs/{job_id}/models
post /jobs/{job_id}/start
post /jobs/{job_id}/stop
post /jobs/{job_id}/training
post /jobs/{job_id}/evaluation
post /jobs/{job_id}/deployment
model
get /models
get /models/{model_id}
post /models/{model_id}/evaluation
post /models/{model_id}/deployment
dataset
post /dataset
get /dataset
get /dataset/{dataset_id}
put /dataset/{dataset_id}
delete /dataset/{dataset_id}
get /dataset/{dataset_id}/versions

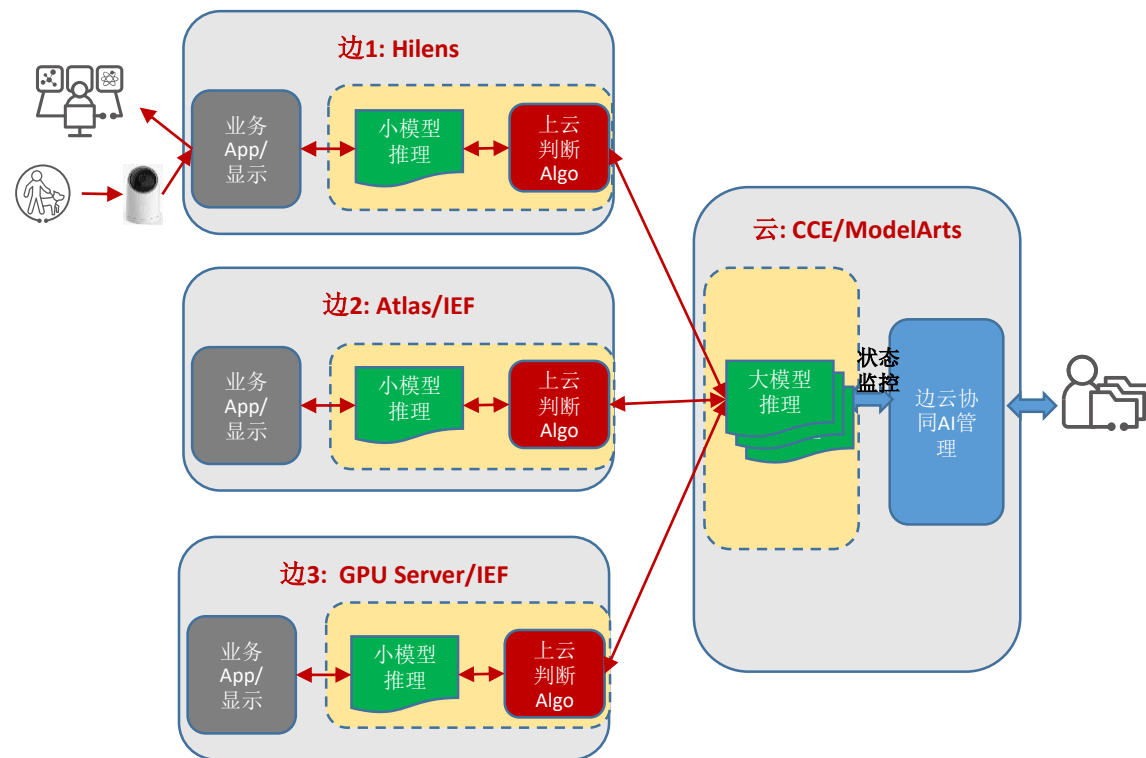
# Feature: joint inference



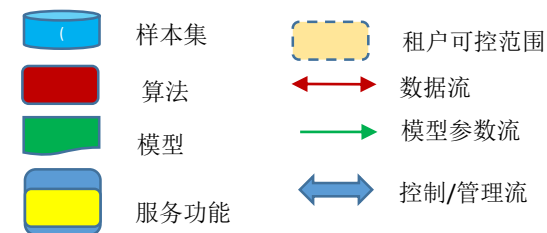
- 1 **预训练:** 用户开发者上传样例数据
1. **迁移学习:** 样本选择从公开数据集发现特征近似样本, 准备训练数据。
  2. **多优化目标模型训练:** 训练边云协同大小模型. 边云模型优化目标不同, 分别关注 recall和precision.



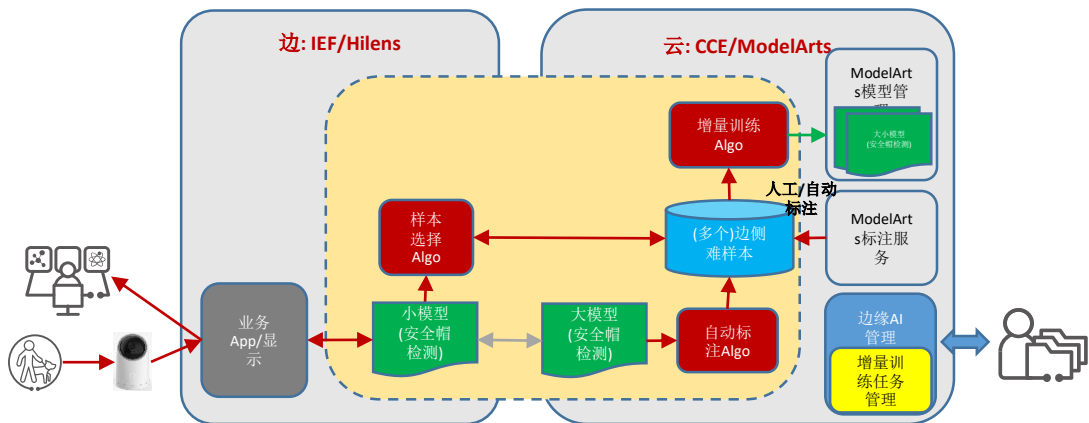
- 2 **边云协同推理:** 边云大小模型协同推理, 上云判断Algo决策将置信度低送到云上推理



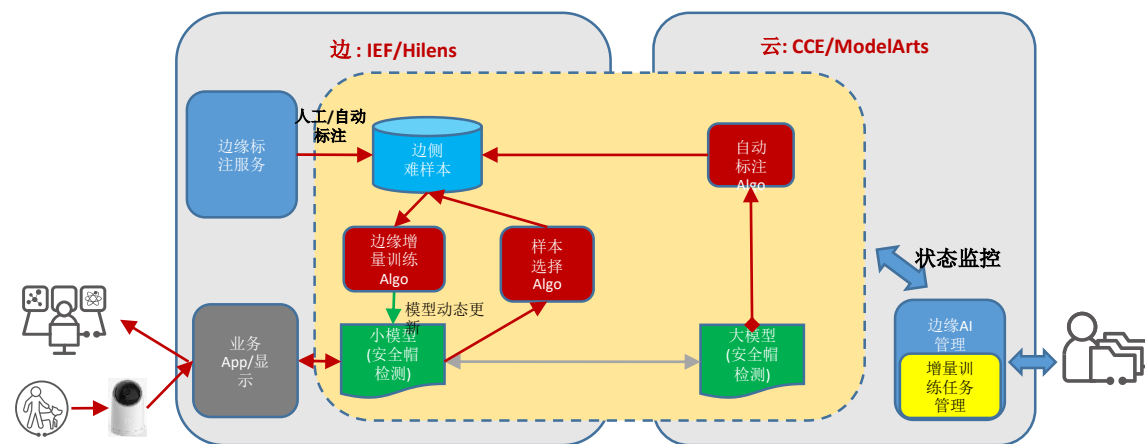
# Feature: incremental learning



**3A** 边云协同增量学习(云端训练):用户运维者云上标注; 云端增量训练大小模型, 完成后更新大小模型



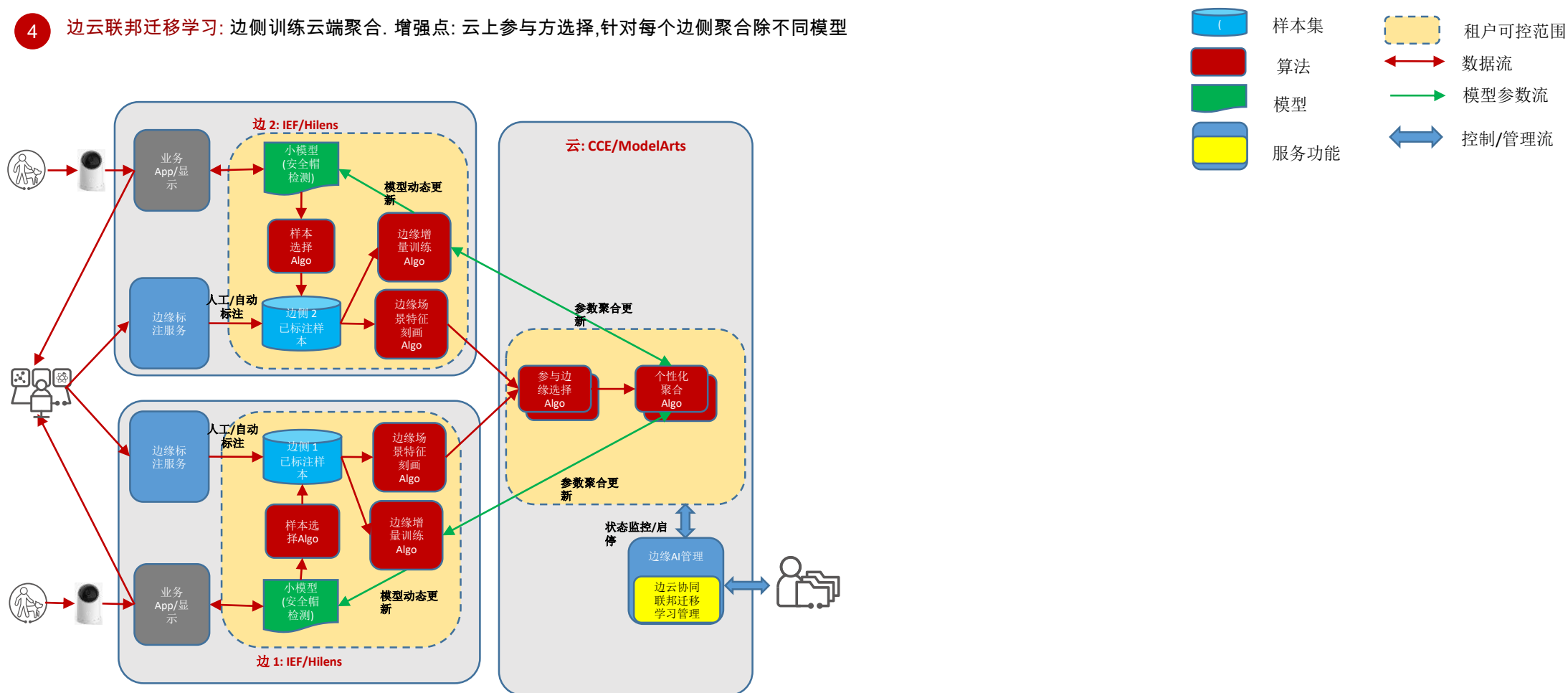
**3B** 边云协同增量学习(边侧训练):边侧标注, 边侧增量训练; 边侧自动更新





# Feature: collaborative training (fed learning)

4 边云联邦迁移学习: 边侧训练云端聚合. 增强点: 云上参与方选择, 针对每个边侧聚合除不同模型



Thanks

the workflow