Distributed Micro-Services Communication Problem Statements & Future Plans

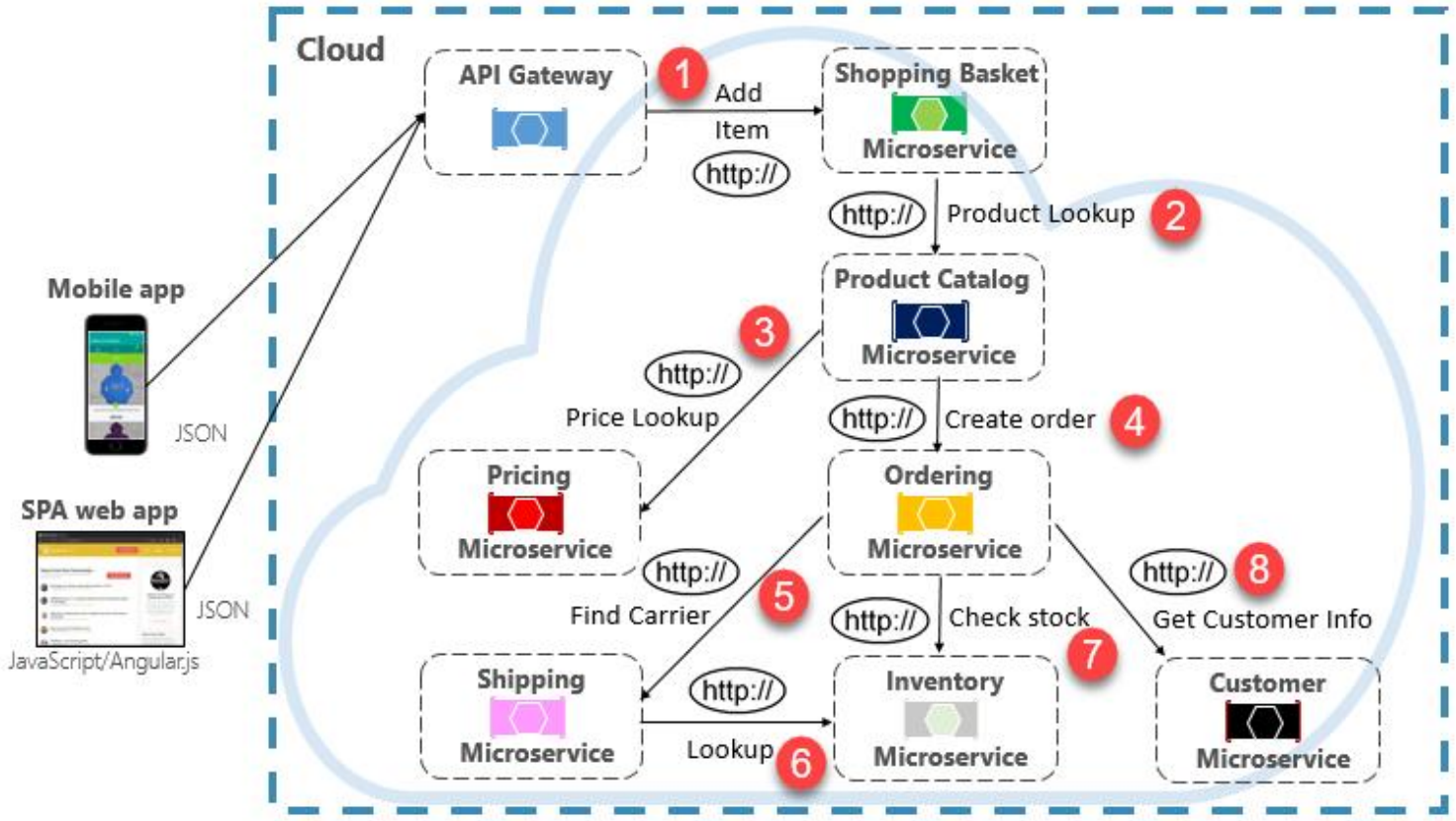
China Telecom

Main Contents

- **Problems of Micro service**
- Key ideas of the Distributed Micro Service Communication (DMSC) architecture
- Future plans

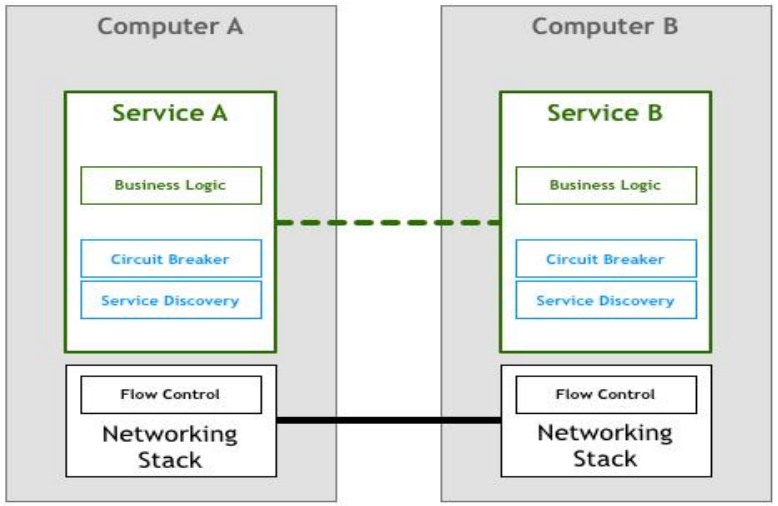
Requirements of Distributed Mirco Services from the Cloud

- Microservices – small, independent units of application
- General microservices communications have the following steps:
  - Service registration
  - Service discovery
  - Service measurement
  - Service scheduling

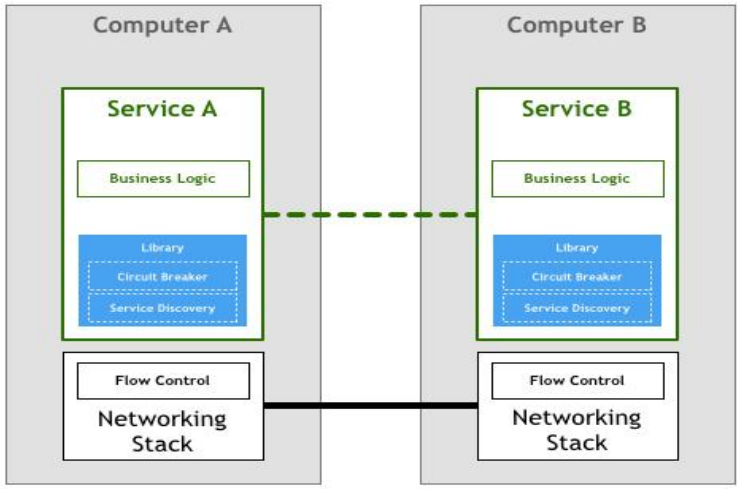


What is a Service Mesh

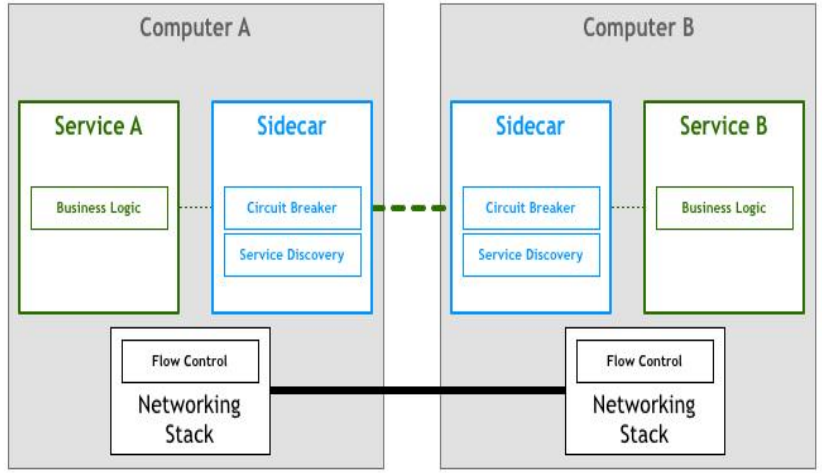
- A service mesh is a dedicated infrastructure layer for handling service-to-service communications
- A service mesh provides such capabilities like traffic management , security, observability etc.
- Below is the evolution history of service mesh



1: Standlone Micro Services



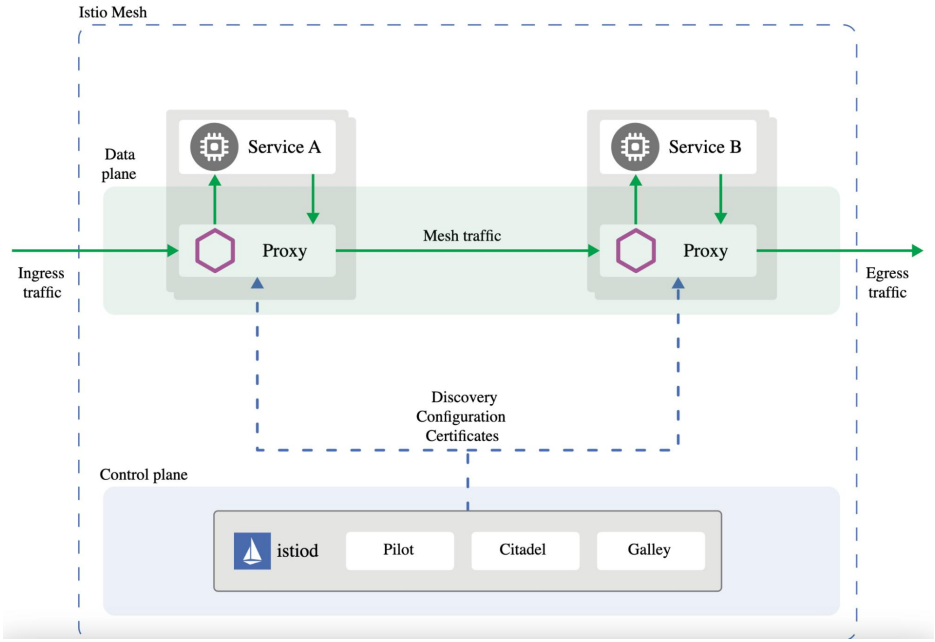
2: Embeded General Library



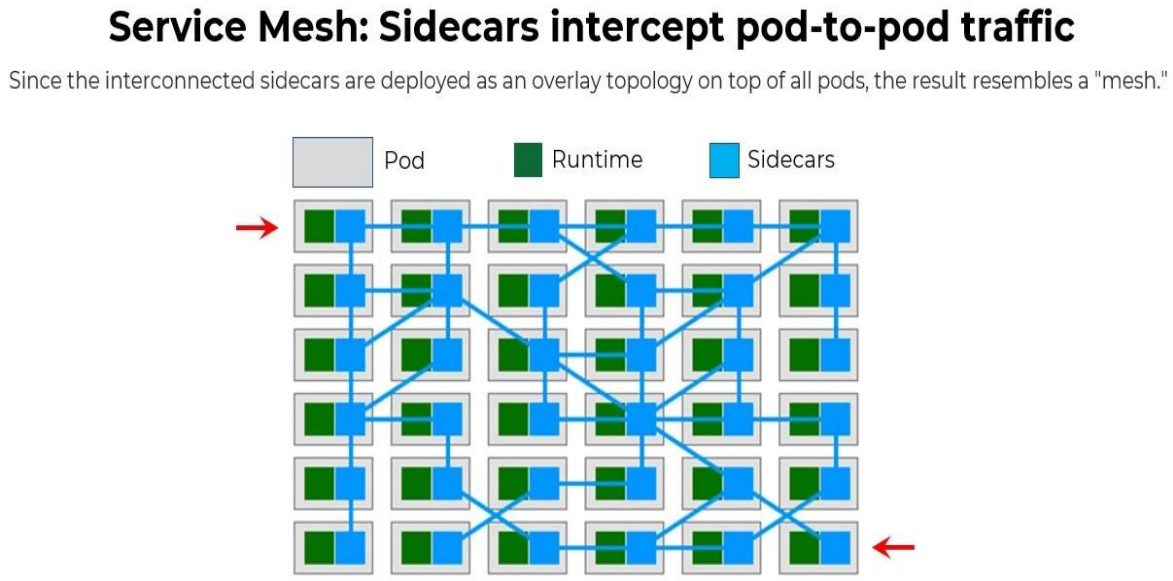
3: Sidecar

The aim of a Service Mesh

- **Service discovery:** Discover other services and route requests through the mesh accordingly.
- **Load balancing:** Distribute traffic evenly across instances of a service.
- **Security:** Authenticate and authorize access and encrypt data.
- **Observability:** Collect metrics, logs, and traces for monitoring.
- **Traffic control:** Use sidecars to get fine-grained control over routing and retries.

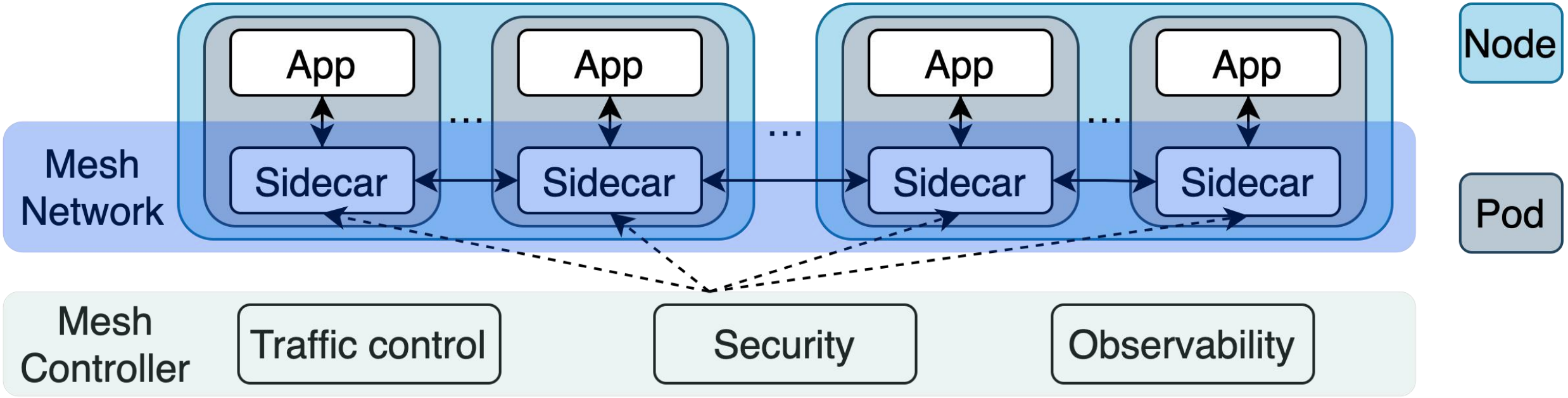


Istio Architecture



Service Mesh Topology

Problems of Per-Pod Sidecar



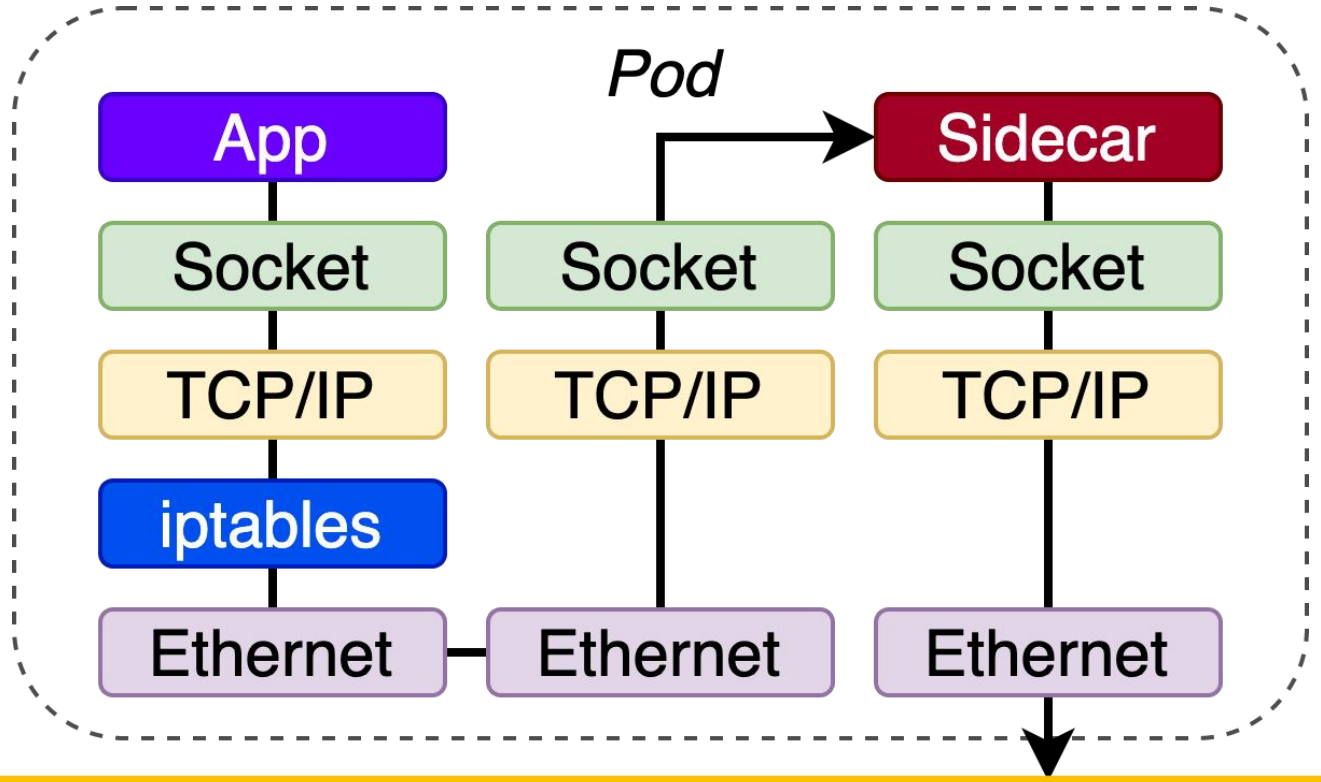
Problem 1：High intrusiveness and potential risks.

✕ Shared memory: memory leakage in the sidecar may cause the app to crash.

✕ Shared lifecycle: upgrading sidecar will require a pod restart, disrupting the app.

✕ Shared controller: potential misconfiguration risks.

Problems of Per-Pod Sidecar



Problem 2：Performance degradation due to extra steps.

✕ Extra steps: two extra context switch, memory copy, protocol stack processing[1].

✕ Performance degradation: throughput and latency degrade by 3x~7x[1]

[1] SPRIGHT: extracting the server from serverless computing! High-performance eBPF-based event-driven, shared- memory processing. In Proceedings of the ACM SIGCOMM 2022 Conference. 780–794.

Problems of Per-Pod Sidecar

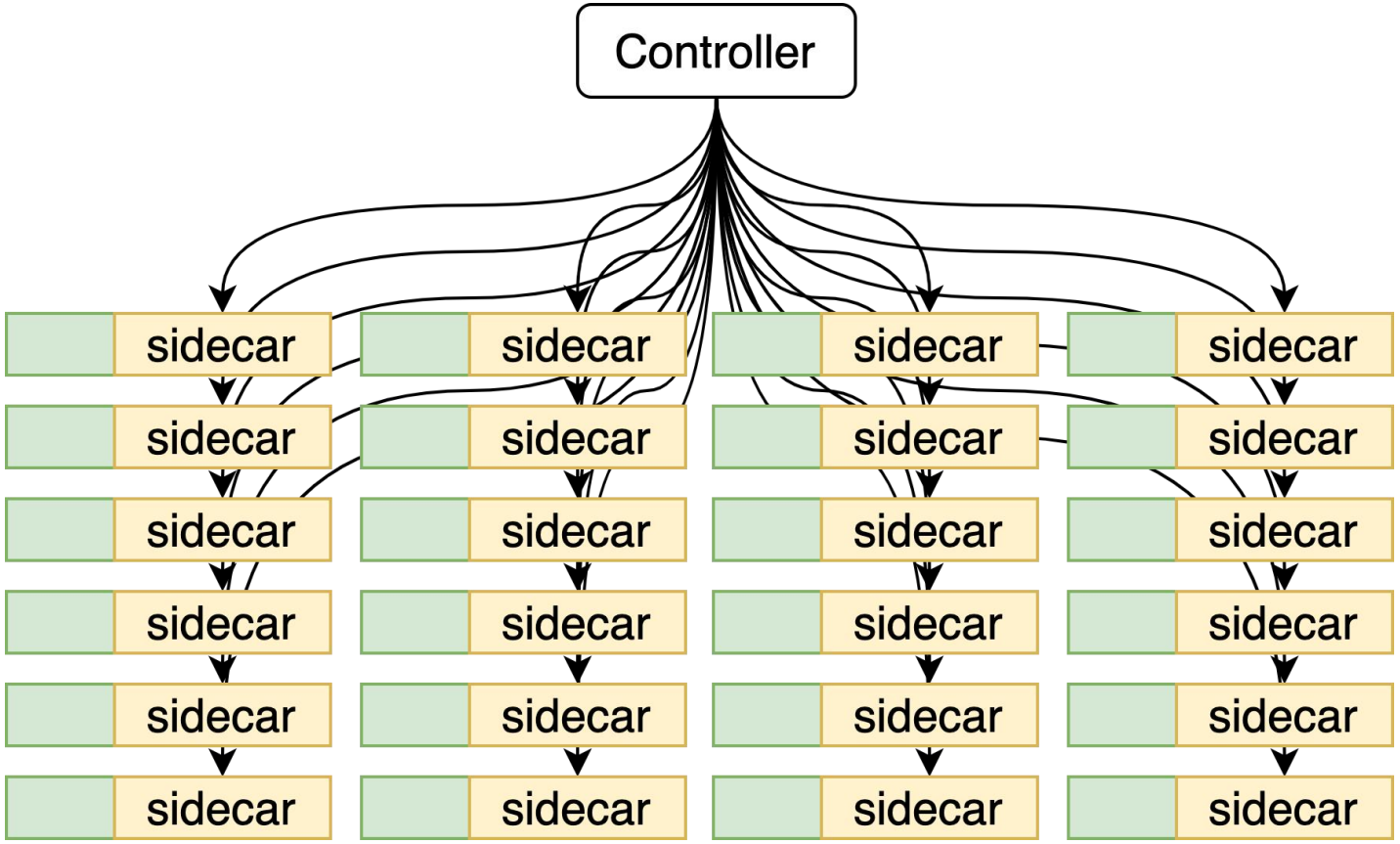
Cluster size		*Resource usage of sidecar*	
Node	Pod	CPU	Memory
500	15k	1500core, 10%	5000GB, 10%
200	8k	1000core, 8%	1200GB, 5%
100	1k	32core, 4%	150GB, 5%
60	2k	400core, 10%	300GB, 6%
60	400	150core, 30%	300GB, 25%

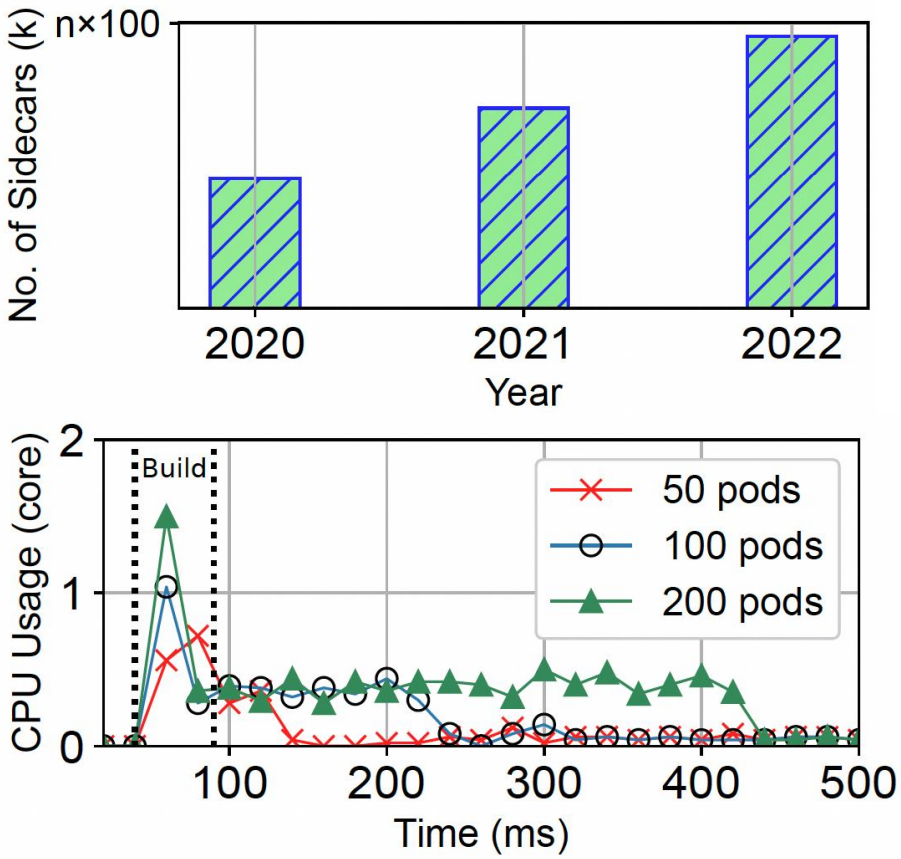
Problem 3：Excessive resource consumption.

✕ For a large K8s cluster: 1500 CPU cores and 5000 GB memory.

✕ Extreme case: 3x CPU and 5.54x memory more than app.

Problems of Per-Pod Sidecar





Problem 4：High control plane overhead.

✕ High orchestration overhead: O(100k) pods for a cloud service.

✕ High southbound overhead: configuration delays or even losses.

Main Contents

- Problems of Micro service
- **Key ideas of the Distributed Micro Service Communication (DMSC) architecture**
- Future plans

China Telecom Motivations

- Considering the above challenges, we require an innovative solution that:

  - Adapt to the continually growing demands of microservices communication.
  - Feature end-to-end service telemetry capabilities for real-time monitoring of communication and performance data between microservices.
  - Provide robust mechanisms to ensure the high availability and performance of critical microservices.
  - Offer flexible scheduling capabilities for dynamic resource allocation based on demand.
  - Support information-centric communication, facilitating the evolution of our network towards information-centric networking.

Features of Distributed Micro Service Communication

- **DMSC:** Distributed Micro Service Communication
- A novel approach to overcome the existing challenges by **leveraging of Information-Centric Networking (ICN) with the service mesh** communication architecture.

Content-Centric:

- prioritize content and services

Decentralization:

- distribute processing and storage capabilities

Dynamic Resource Allocation:

- optimize resource allocation
- enhancing network efficiency

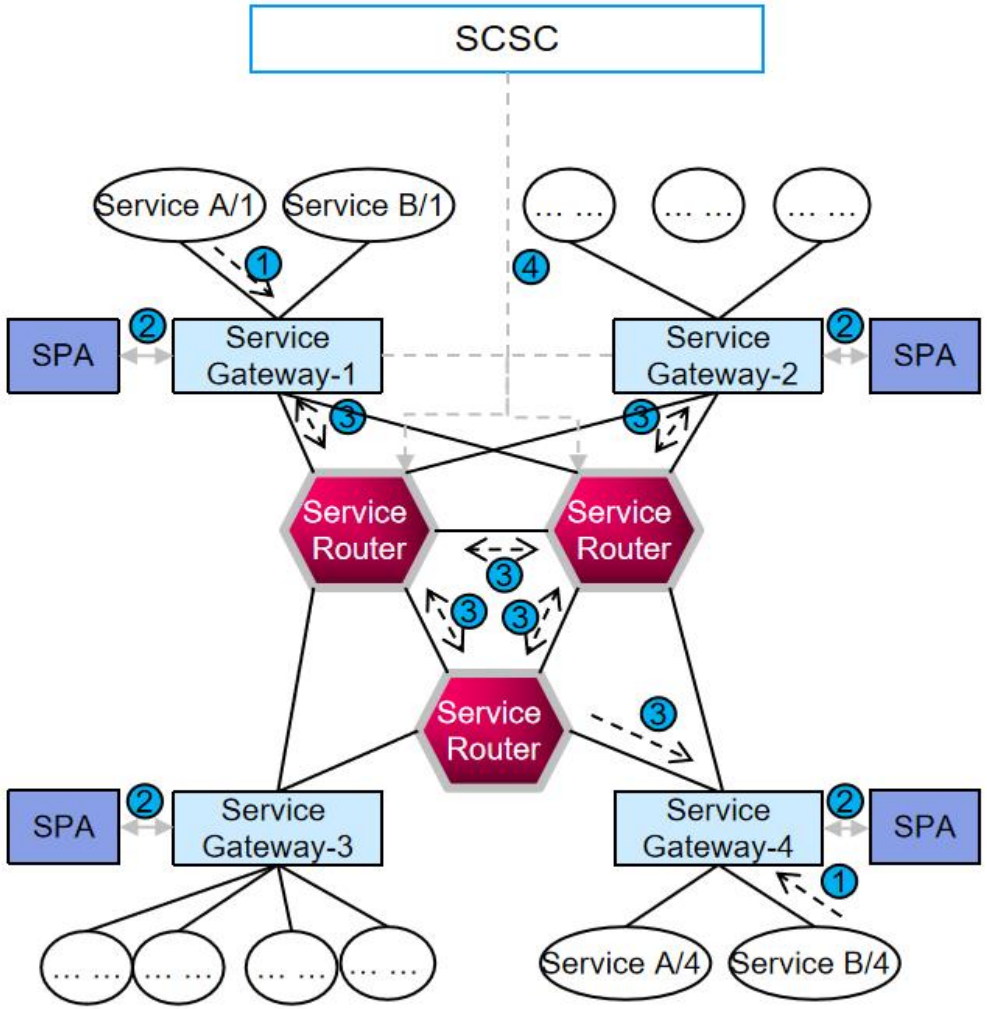
Scalability and Flexibility:

- accommodate the evolving demands of the network

Overview of DMSC

DMSC Purpose: Enhance microservice communication efficiency and reliability

- **Components:**
  - **Service Gateway (SG):** Default gateway for microservices； manages and controls communication traffic.
  - **Service Router (SR):** Optimizes routing based on Prefix and topology; exchange of reachable Prefixes and topology info.
  - **Service Prefix Authentication (SPA):** Validates Prefix usage by microservices.
  - **Service Mesh Communication Scheduling Center (SCSC):** Assist in optimizing communication policies.



- **Benefits:**
  - Decentralized routing decisions via SG and SR.
  - Routing Optimization based on SCSC.
  - Enhanced security via Prefix authentication.

Main Contents

- Problems of Micro service
- Key ideas of the Distributed Micro Service Communication (DMSC) architecture
- **Future plans**

Establish one new WG within IETF

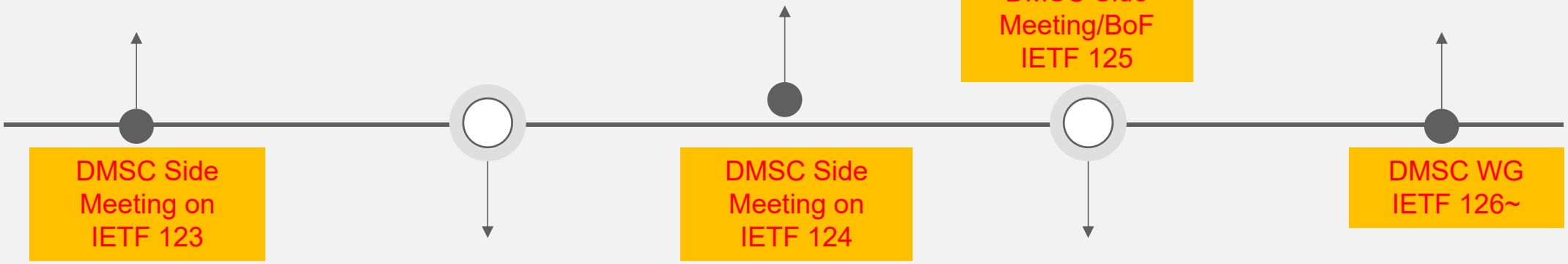
- IETF is the venue to industrilize the technology and related specification.
- Many experts from the vendors are gathered at the IETF.
- Service Mesh is the trigger point to initiate the new WG to accomplish DMSC.
- Published RFCs within IETF should be revisited to find the Gap between the existing standardizations work and implementation requirements from the vendors.
- The new WG(DMSC--Distributed Mirco Service Communication) may locate in WIT area of IETF.

IETF DMSC WG Planning

Investigate the current implementations and challenges of Service Mesh, and also other killer applications

Ecological chain partners reach a consensus and determine the progress and goals.

IETF DMSC WG Established



Define the architecture of Distributed Service Mesh.

Reviewed the published IETF RFCs

Communicate with the AD to define the goals and milestone plans of the new working group.

中国电信
CHINA TELECOM

5G hello

Thank you!