

# COMS Architectural Design Enablers & Artefacts -I

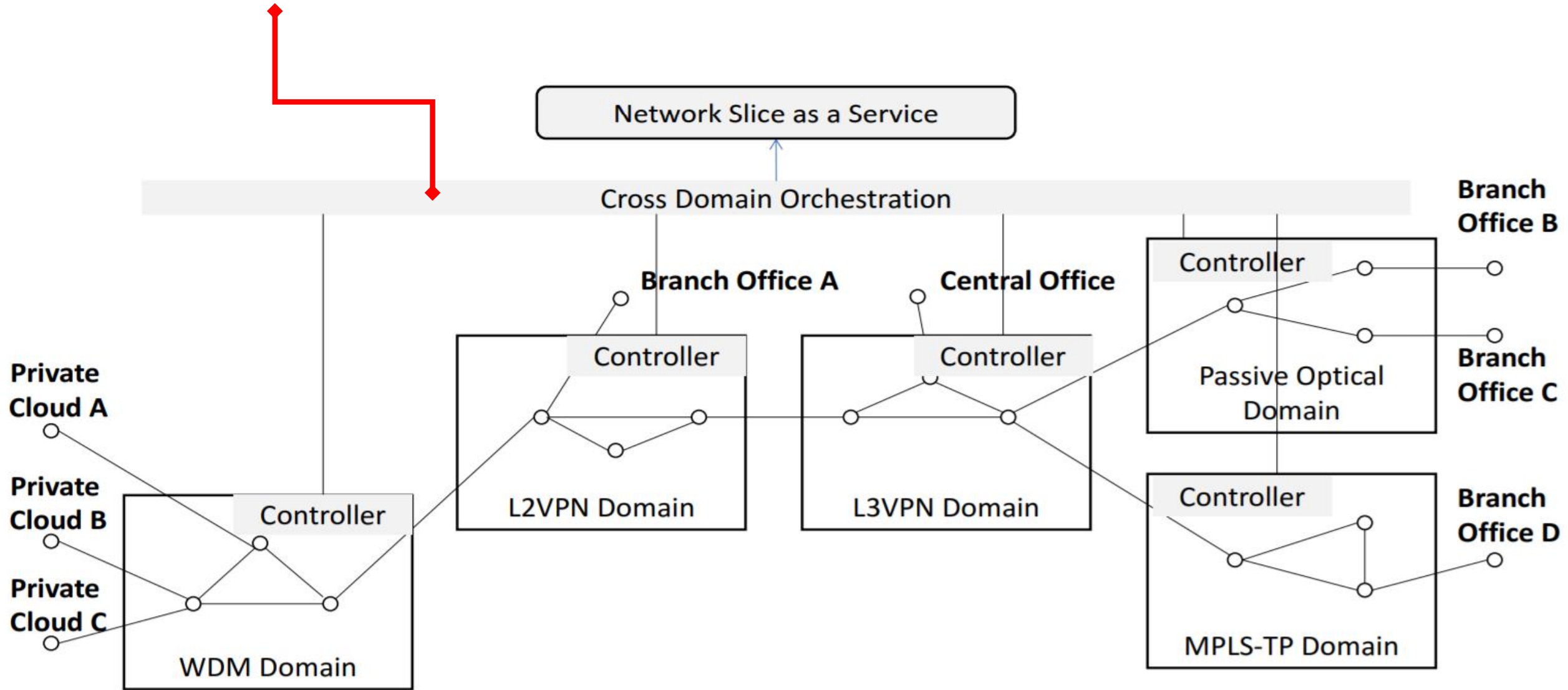
COMS Technology Independent Information Model

[draft-qiang-coms-netslicing-information-model-02](#)

Cristina QIANG – Huawei

# What can we see at this layer?

— no complete and unified view



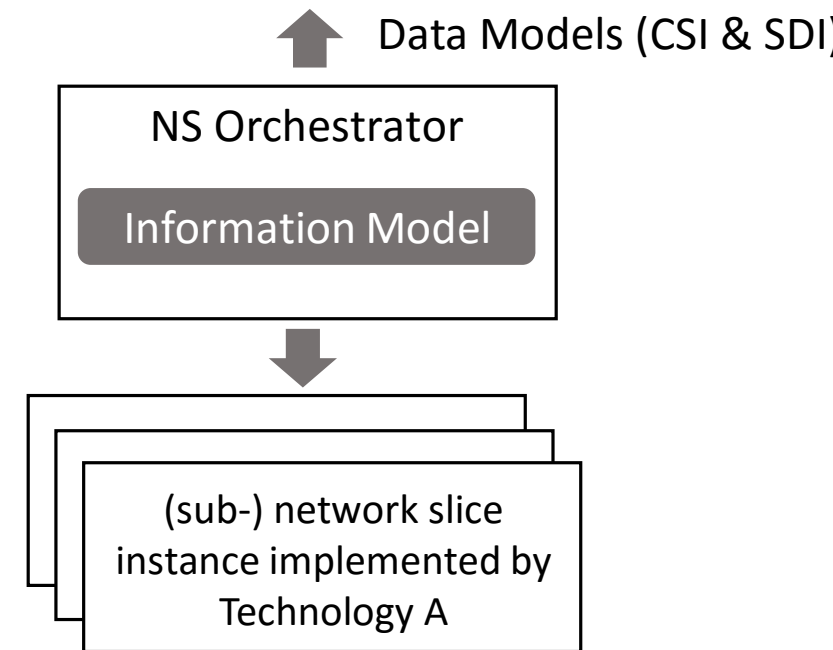
# COMS Information Model describes what network slice looks like, independent of NS implementation technology

- used in providing a complete slice view in cross-domain scenarios
- used in the mapping to different implementation technologies at data-plane
- used to enable the design of data models

Information Model (Apartment)

Attribute \ Entity	Material	Color	Price
Desk	Int 8 (1-wood, 2-metal, ...)	(Float R, Float G, Float B)	Float
Light	String 32	String 32	Enum (1-0~100, 2-100~200, ...)
Sofa	String 64	Int 8 (1-white, 2-black, ...)	String 64
...			

Data Model



Information Model will be followed by data models (**C**ustomer **S**ervice Interface & **S**ervice **D**elivery Interface) as a next step

# Various languages to describe the information model – UML, Yang, Pseudocode, Plain Text, etc.

Network Slice (in progress)

Technology-Independent Concept  
≠ L2VPN/L3VPN/SFC/MPLS/VLAN....

Connectivity

- Resources
- Node
- Link
- Storage-Units
- Computing-Units
- ...

- Generalized-Function-Block
- Tenant-Controller
- Load-Balancer
- Firewall
- ...

- Slice-Level Attribute
- ACL
- Service-Start-Time
- Service-Finish-Time
- Resource-Reservation-Level
- Reliability-Level
- ...

**None** – slice instances share and compete for resources

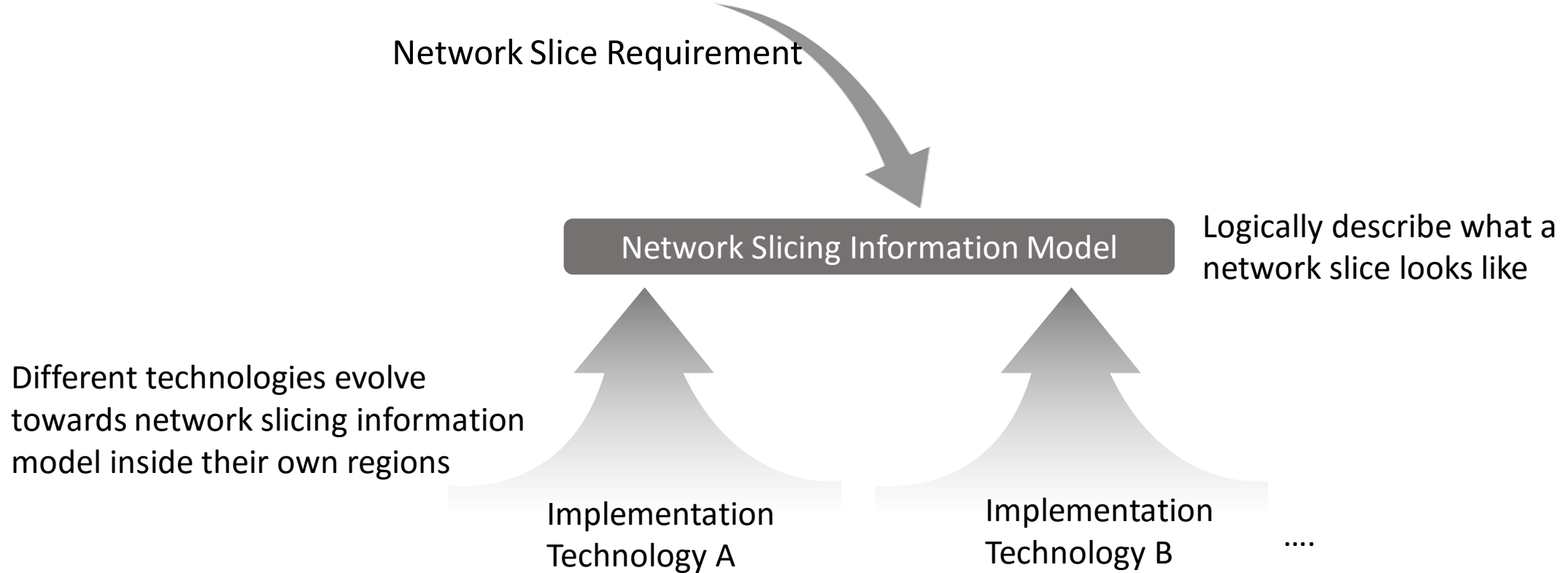
**Shared-non-Preemptive** – free reserved resources could be used by other slice instance, unable to be retrieved if other slice instance are using them

**Shared-Preemptive**

**Exclusive**

# Conclusion

- Provide a complete slice view in heterogeneous network infrastructures
- NS Top-down and Bottom-up approaches converge at Information Model



Thank You

```

|Network Slice Service Profile
|
*****
*Top Level Network Orchestrator based on COMS
*
* +-----v-----+
* |Common Information Model
* |
* | . ~~~~~ T(A->B)<=10ms; B(A->B)>=10M ~~~~~ .
* | . ~ A ~~~~~ B ~ S(B)=1G.
* | . ~~~~~
* | . | T(A->C)<=20ms; B(A->C)>=10M ~~~~~
* | . +-----v----- C ~ S(C)=2G.
* | .
* |
* +-----+
* |
* +-----v-----+
* |Split Service Profile into Domains
* |
* | . Domain 1 . .Domain 2
* | . T(A->D)<=2ms . . T(D->B)<=8ms S(B)=1G
* | . ~~~~~ B(A->D)>=10M ~~~~~ B(D->B)>=10M ~~~~~
* | . ~ A ~~~~~ D ~~~~~ B ~
* | . ~~~~~
* | . | T(A->E)<=2ms . . T(E->C)<=18ms S(C)=2G
* | . | B(A->E)>=10M ~~~~~ B(E->C)>=10M ~~~~~
* | . +-----v----- E ~~~~~ C ~
* | . ~~~~~
* |
* +-----+
* |
* +-----v-----+
* |Select Specific Implementation Technologies
* |
* | .Domain 1 . .Domain 2
* | . Flex-E . . VPN+NFV
* | .
* |
* +-----+
* |
* +-----v-----+
* |Map to Selected Technologies
* |
*****
|
*****v*****
* Flex-E Controller *
*****+*****
|
*****v*****
* Physical/Logical *
* Resources inside *
* Domain 1 *
*****

|
*****v*****
* VPN Controller *
*****+*****
|
*****v*****
* Physical/Logical *
* Resources inside *
* Domain 2 *
*****

|
*****v*****
* NFV Orchestrator *
*****+*****
|
*****v*****
* Physical/Logical *
* Resources inside *
* Domain 2 *
*****

```