Network Powered by Computing

Ruslan Smelyanskiy

Prof. MSU, Corresponding Member Russian Academy of Sciences

APPLIED RESEARCH CENTER FOR COMPUTER NETWORKS

turing lecture

4.06.2018

DOI:10.1145/3282307

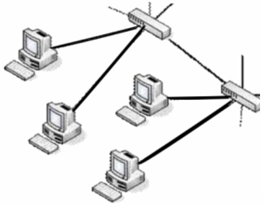
Innovations like domain-specific hardware, enhanced security, open instruction sets, and agile chip development will lead the way.

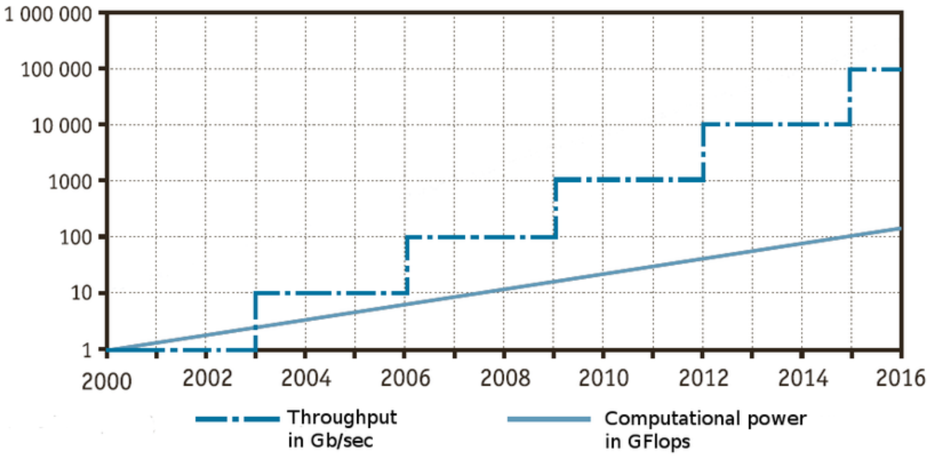
BY JOHN L. HENNESSY AND DAVID A. PATTERSON

A New Golden Age for Computer Architecture

New Golden Age of Computational Infrastructure

- the end 60-s – Computer installation with job packet processing;
- 70-s - mainframe computer center with terminal network;
- 80-s – Client-Server infrastructure with network access;
- 90-s – Servers Farm with Frontend server with access via LAN;
- 2000-s – monstrous DC with high speed WAN;
- Quo Vadis?





Application Requirements + Hardware Capabilities + Software Engineering

Applications suite of features

- **Distributed** –applications are composed of a set of functions/services that run in parallel on different nodes and have to integrate geographically distributed data;
- **Self-sufficient -** the application is no longer just code and source data, it is accompanied by a specification and orchestration of the components (application services), relationship topology, the determination of the required level of their performance, explicitly formulated requirements for the resources (computing, network, storage) and deadlines for their communication;
- **Elasticity** –the performance of the application changes automatically without interrupting its operation in accordance with the requirements of the SLA and the current load on it;
- **Real-Time mode** –applications are sensitive to delays and its response time is imitated;
- **Cross-platform** - it doesn't matter what software environment or hardware platform is available for the application;
- **Interaction and Synchronization -** combining the results of different stages of computations, regardless of their location, aggregation of service chains;
- **Maintainability** - updating the application does not require any action on the part of the user;

APPLIED RESEARCH CENTER FOR COMPUTER NETWORKS

The main force of computational infrastructure developments are applications needs!

Computational Infrastructure Requirements

- **Behavior predictability** – predictability of delays associated with computations, transfer and access to data during the application operation, in order to manage application's execution accordingly to the requirements of the SLA;
- **Security** – it does not pose unacceptable risks to the application and its data like Confidentiality, Integrity, Availability;
- **Availability, Reliability and Fault Tolerance** - the infrastructure should be robust enough to ensure a high level of availability and operability of its services, application components, recovery of lost data in case of failures and attacks, react in real time by changes in topology, traffic flows and shape routing to ensure the fulfillment of SLA requirements;
- **Efficiency and Fairness** -the infrastructure must ensure that the application runs, delivers and processes its data by infrastructure resources, reliably, without impair other applications and their traffic;
- **Virtualization** - virtualization of all types of resources (computing, storage, network)
- **Scalability** - it should be efficiently scalable depend on the number of data, services and applications points of presence in terms of performance;
- **Serverless** – the infrastructure should automatically place application components in a way that allows them to interact according to the application stricture, and in a way that ensures that the SLA requirements of the application are met, while minimizing infrastructure resources utilization.

- **The scaling range of the network service is huge and in real time, which put high demands on the algorithm time complexity.**
- **Only sub-optimal solutions are available using methods based on machine learning**

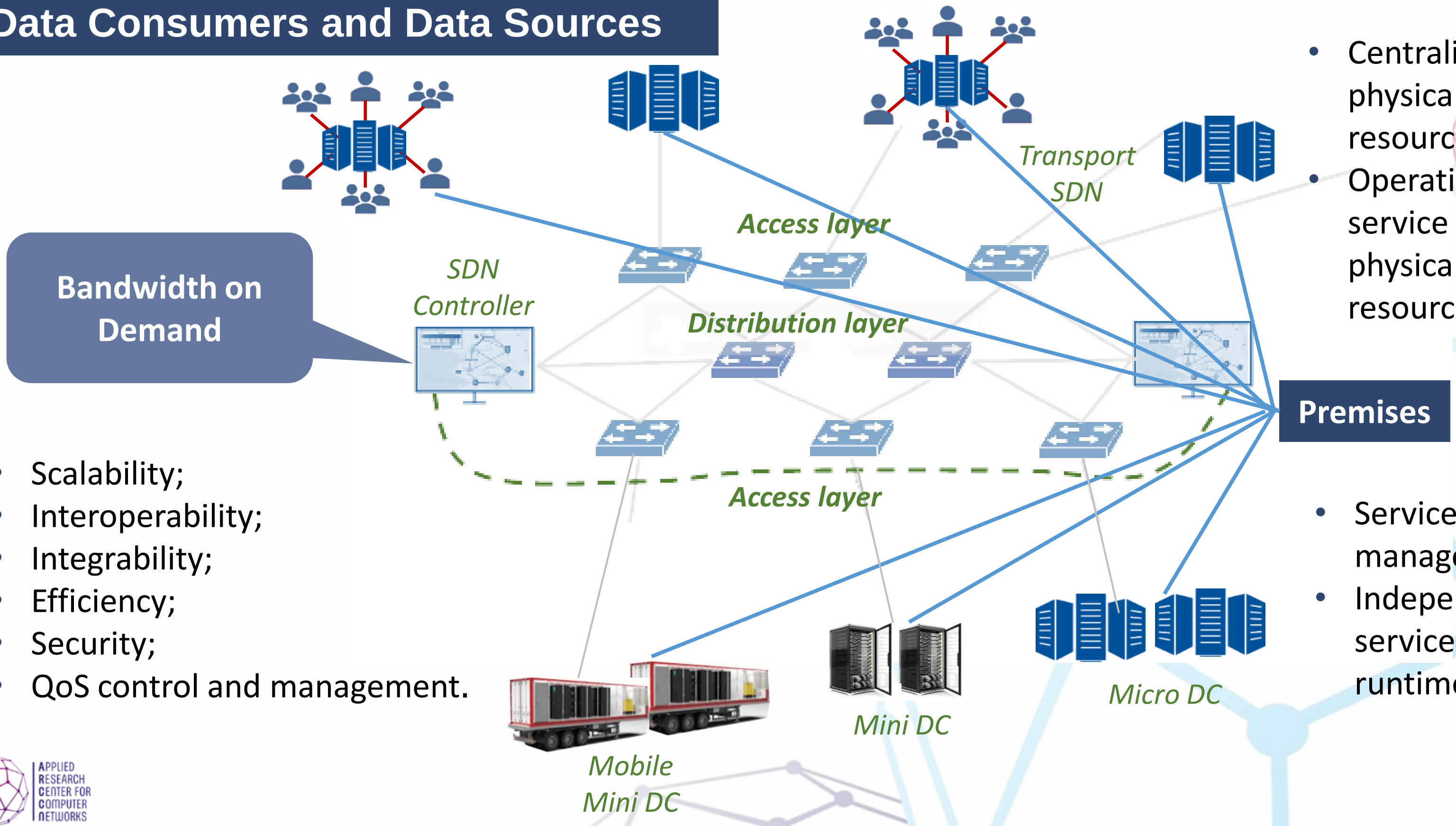
Network Powered by Computing is Super Large Scalable Computer

Fully Controllable Programmable Virtualized Infrastructure John Gage: SunMicrosystems

APPLIED RESEARCH CENTER FOR COMPUTER NETWORKS

Software Defined Network Network Function Virtualization

Data Consumers and Data Sources

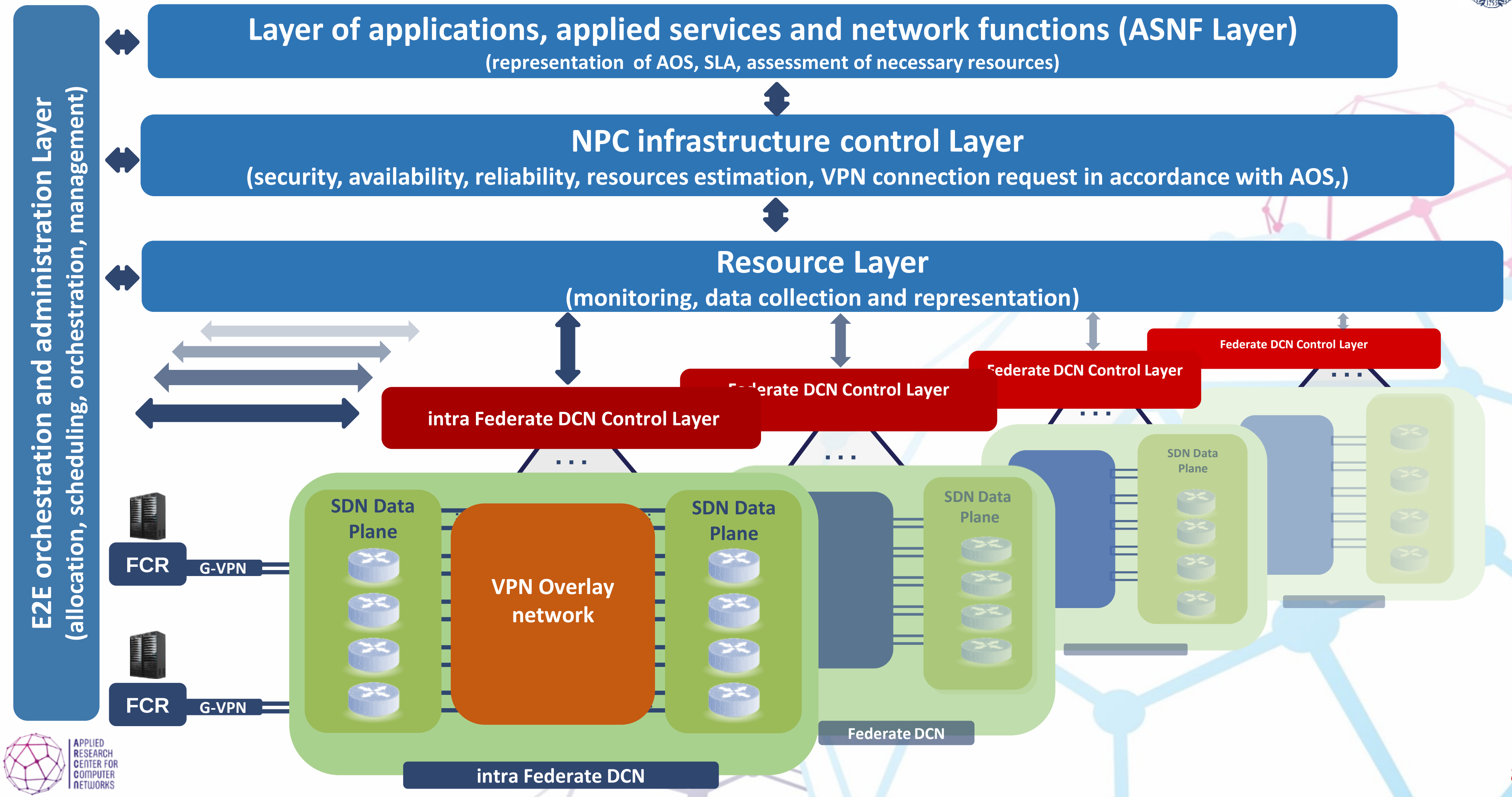


- Centralized management of physical infrastructure resources;
- Operational management of service deployment on physical infrastructure resources;

- Scalability;
- Interoperability;
- Integrability;
- Efficiency;
- Security;
- QoS control and management.

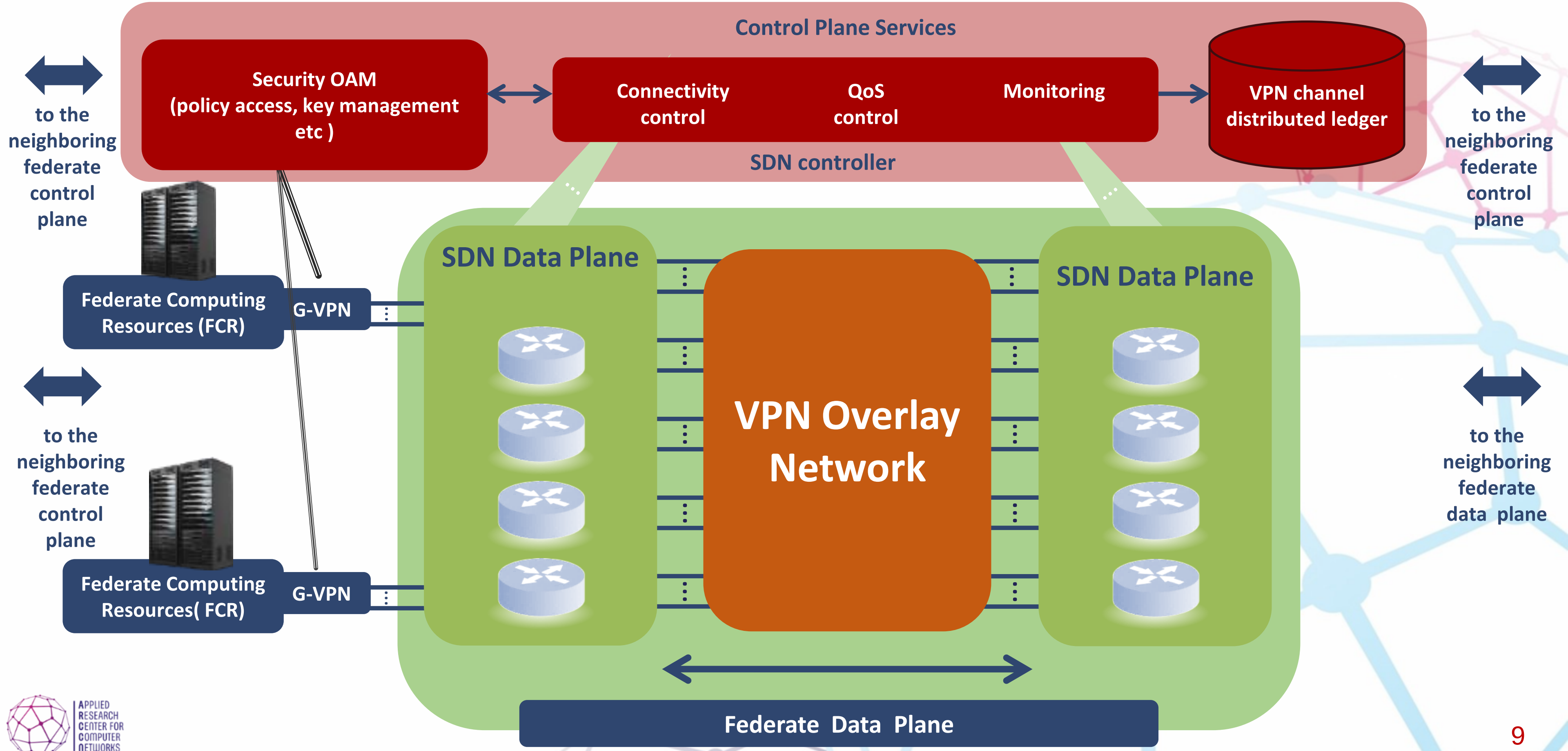
- Service life cycle management;
- Independence of the service logic from the runtime environment..

NPC Functional Architecture



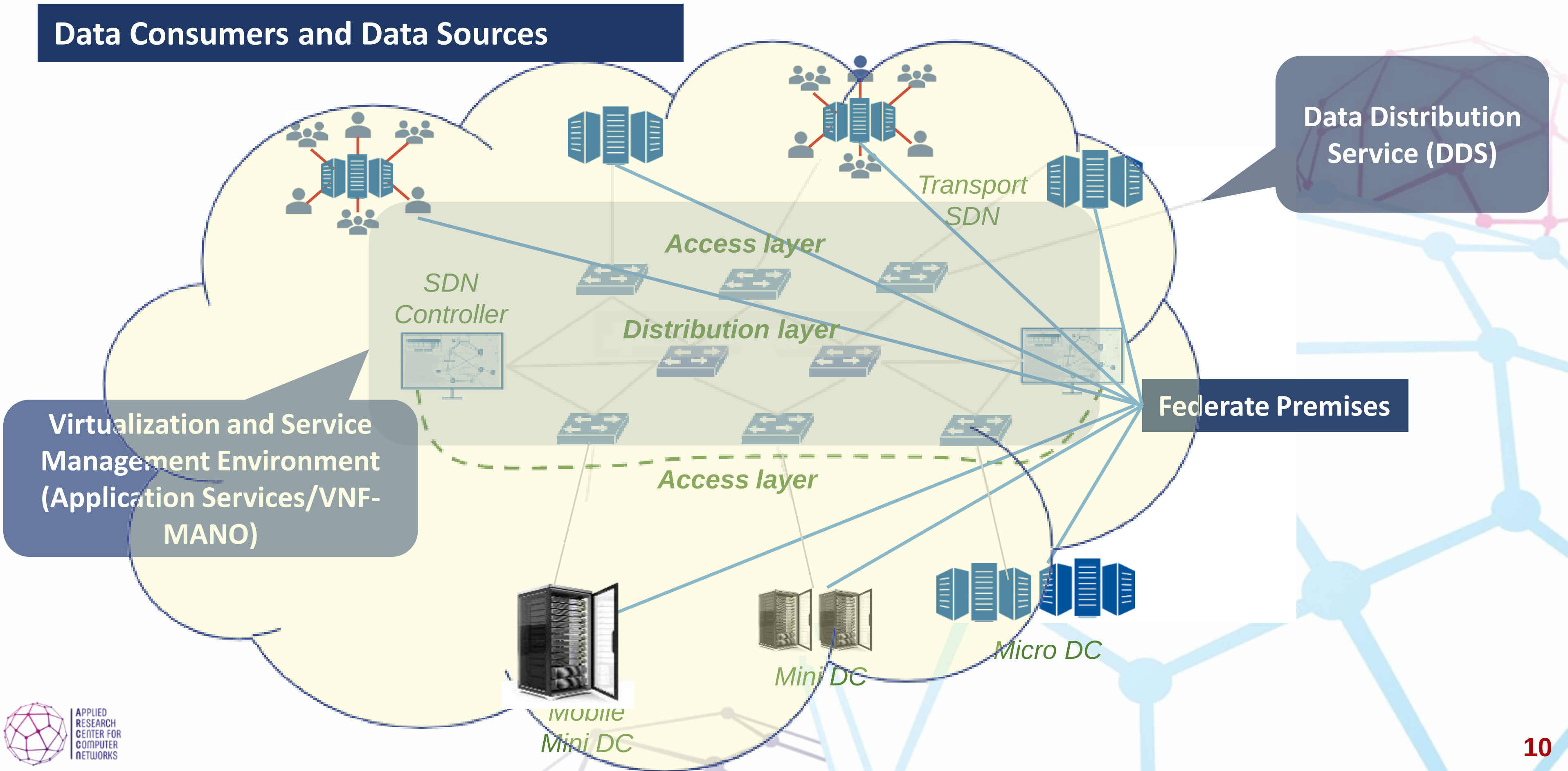
APPLIED RESEARCH CENTER FOR COMPUTER NETWORKS

NPC intra Federates DCN Layer



APPLIED RESEARCH CENTER FOR COMPUTER NETWORKS

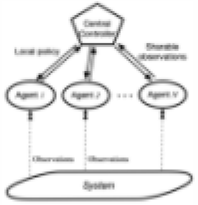
NPC: General View



Multi-agent optimal control

Efficiency → Distributed control
Accuracy → Centralized control

Centralized approach



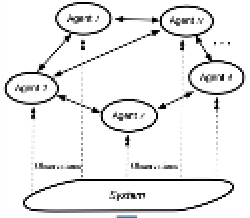
Each agent knows its local state.

The control center gathers the status of each agent.

The control center makes a decision based on the optimization policy.

Each agent is given a control action.

Distributed approach: Agent network

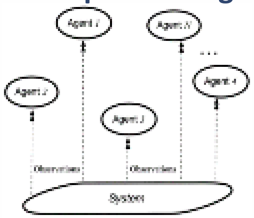


Each agent knows its local state.

Information exchange is limited to neighboring agents only.

Based on local information and information collected from neighbors, each agent decides on the optimal strategy for himself.

Distributed approach: Independent agents



Each agent knows its local state.

Each agent judges the control strategy and actions of other agents based on his experience.

The agent implements control decisions in accordance with its local optimization strategy and based on its observations.

Computing task scheduling → Dynamically tuned computing node (CN) scheduling

CN distribution: each CN decides to take a task or determines where to transfer it - a cooperative distribution of tasks between CNs.
Distributed and independent TE: each network node independently decides on the distribution of flows over available channels.

Service chain scheduling→Dynamic load of chain services in CN

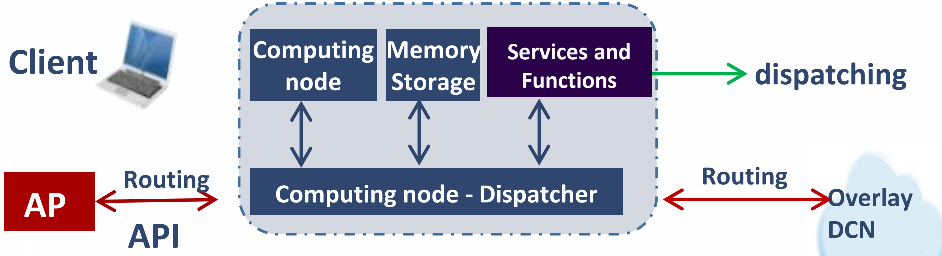
Distribution of chain services:
Accounting for time constraints and interaction logic.
Maximum load of CN resources (computing & storage).
Distributed and independent TE: each network node independently decides on the distribution of flows over available channels.

Problems of Multi-agent control

Poor scaling;
There are no mathematical models that guarantee convergence to the optimal solution;
Selection of the optimization functional;
The constraint of the deviation from the optimal solution is not guaranteed.

APPLIED RESEARCH CENTER FOR COMPUTER NETWORKS

Optimal SFC allocation for active mode



Problem: optimal distribution w ϵ ***W*** on NPC: $\{cn_i\}_w$

Necessary solutions:

- Minimizing the objective function for all $\boldsymbol{w_i}$ from ***W*** with given $p_i \in P$
- under SLA and available resource constraints

NPC = (V, A), where
V = C N U S N U P, where
CN = { cn**$_i$ **= <cr, m, h>} – set of computational nodes ,
SN – set of VPN gateway ,
P – set of ***NPC*** poles***.***
A= {l**$_{vi,vj}$**=(v**$_i$**,v**$_j$**) | v**$_i$**,v**$_j$ **ϵ V } - channels set of overlay network.
Q** (l$_{vi,vj}$,Δt)* **= (B, D, L, J)** is the function on ***A, *Δt* – interval of time;
W = {wi = (si1, ….., sik)}, set of SFC where ***sij ϵ AS U VNF,***
***sij = <cr, m, h, Q** (l$_{vi,vj}$,Δt)* **> ;**
ET: (AS U VNF) x CN → R - estimations of the execution time of ***sij ϵ AS U VNF,*** on ***cn**$_i$* ϵ ***CN***

objective function

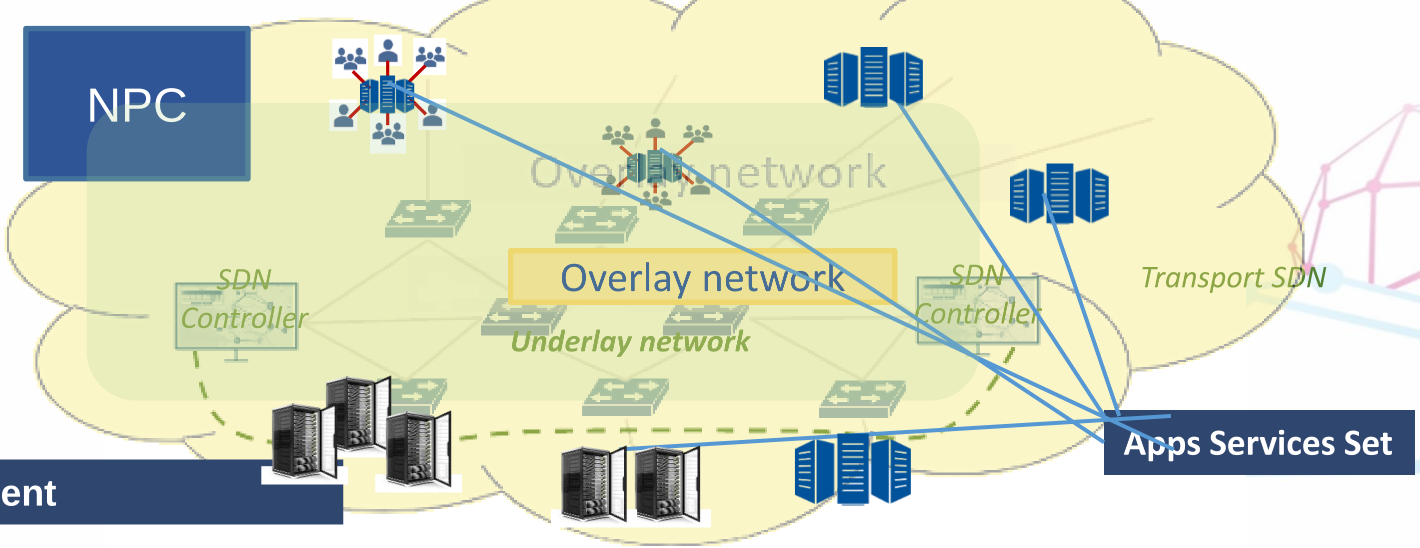
$$F = min \sum_1^{|CN|}\left[\alpha\frac{\bar{c}_i}{c_i} + \beta\frac{\bar{s}_i}{s_i} + \gamma\left(\left(\frac{\bar{c}_i}{c_i} - \Theta\right)^2 + \left(\frac{\bar{s}_i}{s_i} - \Delta\right)^2\right)\right]$$

, where:

α, β, γ – constant values;
c_i, s_i - ***cn**$_i$* resources are used
$\bar{c}_i, \bar{s}_i$ – ***cn**$_i$* resources and queue length averaged over usage time;
Θ, Δ – used resources of the entire NPC, averaged over time;
$(\boldsymbol{cn_i})_{\boldsymbol{w}}$ is a path in NPC correspond to SLA(w)

APPLIED RESEARCH CENTER FOR COMPUTER NETWORKS

Optimal proactive AS allocation problem



Problem statment

Given ***NPC***, ***AS***, ***W*** *and* ***P.***

It's required to build a matrix ***X*** **:** **|*X*| = |*AS*| *x* |*CN*|** *where*

$x_{ij}= 1$, if $\mathbf{s_i}$ can be located on $\mathbf{cn_j}$, otherwise $x_{ij}= 0$

under the following constraints:

1. $\neg\exists\ \mathbf{cn_j}$ and $\neg\exists\ l_{vi,vj} = (\mathbf{v_i},\mathbf{v_j})\ |\ \mathbf{v_i},\mathbf{v_j}\ \epsilon\ \mathbf{V}$, *incident to* $\mathbf{cn_j}$, $l_{vi,vk}\ \epsilon\ \mathbf{A}$ are violated;
2. $\forall\ k: s_{i_k}, s_{i_{k+1}} \in w_i: \exists cn_a, cn_b \in CN: \exists l_{cn_a,cn_b} \in A\ \&\ x_{i_k,a} = 1\ \&\ x_{i_{k+1},b} = 1;$
3. $\forall\ \mathbf{w_i}\ \epsilon \mathbf{W}$, *SLA always met for a given* ***P***.

Applied Research Center for Computer Networks

Every task needs a suitable computer

HPCG Benchmark June 2020

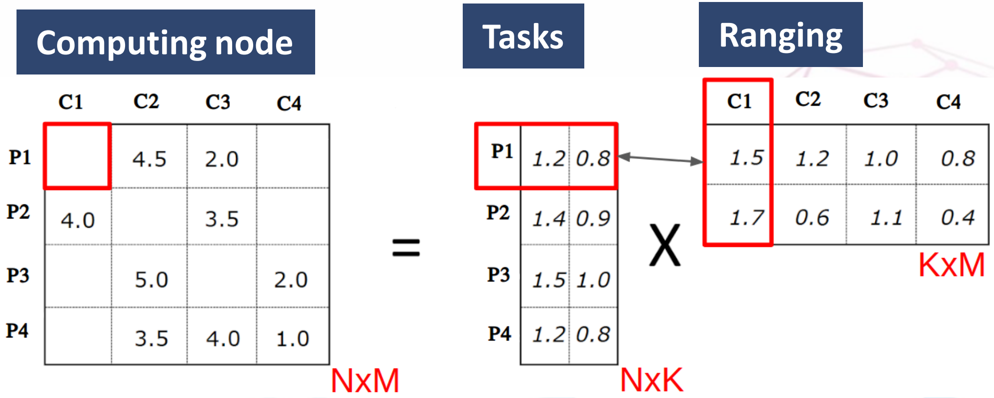
Rank	Site	Computer	Cores	HPL Rmax (Pflop/s)	TOP500 Rank	HPCG (Pflop/s)	Fraction of Peak
1	RIKEN Center for Computational Science Japan	**Fugaku**, Fujitsu A64FX, Tofu	7,299,072	415.53	1	13.4	2.5%
2	DOE/SC/ORNL USA	**Summit**, AC922, IBM POWER9 22C 3.7GHz, Dual-rail Mellanox FDR, NVIDIA Volta V100, IBM	2,414,592	143.50	2	2.926	1.5%
3	DOE/NNSA/LLNL USA	**Sierra**, S922LC, IBM POWER9 20C 3.1 GHz, Mellanox EDR, NVIDIA Volta V100, IBM	1,572,480	94.64	3	1.796	1.4%
4	Eni S.p.A. Italy	**HPC5**, PowerEdge, C4140, Xeon Gold 6252 24C 2.1 GHz, Mellanox HDR, NVIDIA Volta V100	669,760	35.45	6	0.860	2.4%
5	DOE/NNSA/LANL/SNL USA	**Trinity**, Cray XC40, Intel Xeon E5-2698 v3 16C 2.3GHz, Aries, Cray	979,072	20.16	11	0.546	1.3%
6	NVIDIA USA	**Selene**, DGX SuperPOD, AMD EPYC 7742 64C 2.25 GHz, Mellanox HDR, NVIDIA Ampere A100	277,760	27.58	7	0.5093	1.8%
7	Natl. Inst. Adv. Industrial Sci. and Tech. (AIST) Japan	**ABCI**, PRIMERGY CX2570M4, Intel Xeon Gold 6148 20C 2.4GHz, Infiniband EDR, NVIDIA Tesla V100, Fujitsu	391,680	16.86	12	0.5089	1.7%
8	Swiss National Supercomputing Centre (CSCS) Switzerland	**Piz Daint**, Cray XC50, Intel Xeon E5-2690v3 12C 2.6GHz, Cray Aries, NVIDIA Tesla P100 16GB, Cray	387,872	19.88	10	0.497	1.8%
9	National Supercomputing Center in Wuxi China	**Sunway** TaihuLight, Sunway MPP, SW26010 260C 1.45GHz, Sunway, NRCPC	10,649,600	93.01	4	0.481	0.4%
10	Korea Institute of Science and Technology Information Republic of Korea	**Nurion**, CS500, Intel Xeon Phi 7250 68C 563584C 1.4GHz, Intel Omni-Path, Intel Xeon Phi 7250, Cray	570,020	13.93	18	0.391	1.5%

https://hpcg-benchmark.org/custom/index.html?lid=154&slid=309

APPLIED RESEARCH CENTER FOR COMPUTER NETWORKS

Estimated task completion time (example)

P/C	C1	C2	C3	C4
P1, Arg_1	?	?	90	?
P1, Arg_2	?	?	45	?
…	…	…	…	…
P1, Arg_{A1}	5	10	15	20
P2, Arg1	10	12	?	40
P2, Arg2	?	?	?	30
…	…	…	…	…
P2, Arg_{A2}	?	5	?	10
…	…	…	…	…
Pi, Arg_1	25	35	?	56
Pi, Arg_2	45	?	67	100
…	…	…	…	…
Pi, Arg_{Ai}	60	75	96	?
…	…	…	…	…
P_N, Arg_1	?	34	67	?
P_N, Arg_2	?	23	36	200
…	…	…	…	…
P_N, Arg_{AN}	100	146	245	300

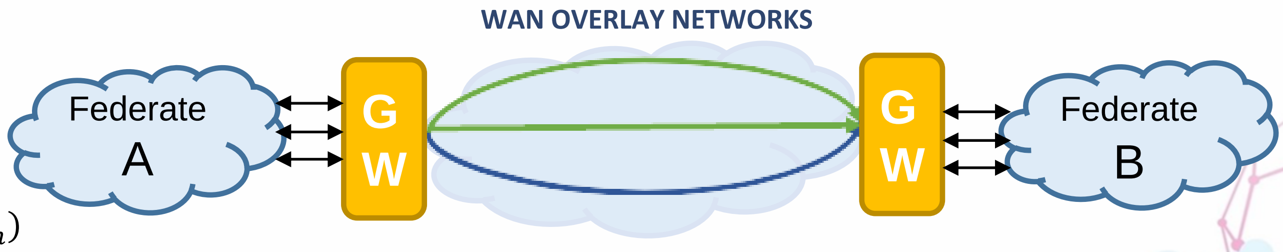


Name	Number of CN	Number of tasks	Benchmark type
MPIL2007	163	12	MPI
MPIM2007	396	13	MPI
ACCEL_OMP	25	15	OpenMP

APPLIED RESEARCH CENTER FOR COMPUTER NETWORKS

https://gitlab.arccn.ru/mc2e/mpi-prog-classification/-/blob/master/pipeline.ipynb

Intelligent DCN transport

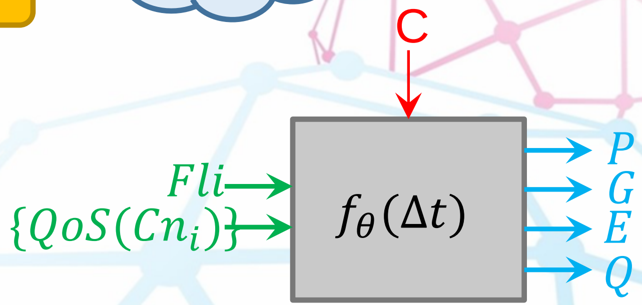


Flow SLA:

$$Fl_1 = (B_1, D_1, J_1, L_1)$$
$$Fl_2 = (B_2, D_2, J_2, L_2)$$
$$\dots$$
$$Fl_m = (B_m, D_m, J_m, L_m)$$

B – minimal admissible bandwidth,
D – maximal time delay,
J – maximal jitter,
L – maximal packet loss probability

Channel state prediction→ Dynamic FEC



Channel state:

$$QoS(Cn_1) = (\underline{R}_1, \hat{R}_1, \overline{R}_1, \mathrm{N}_1, \mathrm{l}_1, \hat{j}_1, \overline{j}_1, \widehat{b}_1, \overline{b}_1, \mathrm{h}_1)$$
$$QoS(Cn_2) = (\underline{R}_2, \hat{R}_2, \overline{R}_2, \mathrm{N}_2, \mathrm{l}_2, \hat{j}_2, \overline{j}_2, \widehat{b}_2, \overline{b}_2, \mathrm{h}_2)$$
$$\dots$$
$$QoS(Cn_n) = (\underline{R}_n, \hat{R}_n, \overline{R}_n, \mathrm{N}_n, \mathrm{l}_n, \hat{j}_n, \overline{j}_n, \widehat{b}_n, \overline{b}_n, \mathrm{h}_n)$$

$\underline{R}_r, \hat{R}_r, \overline{R}_r$ - min, average, max RTT
N_r - number of sent packets
l_r - packet loss rate
$\hat{j}_r, \overline{j}_r$ - average, max jitter
$\widehat{b}_r, \overline{b}_r$ - average, max bandwidth
h_r - current total load

Problems:

- Channel state prediction algorithm;
- Optimal channel selection;
- UDP based transport;
- FEC algorithm .

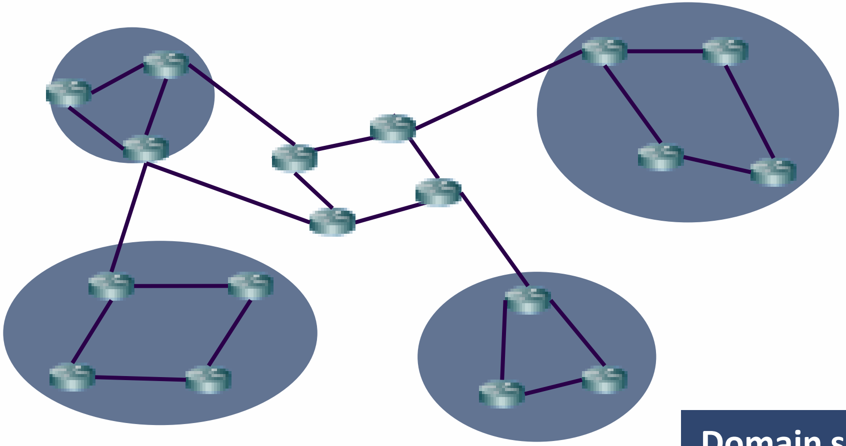
P – the probability of packet successful transmission
G – flow goodput;
E – total overhead for transmission of one byte;
Q – the packet delay.
C – the chosen channel and methods

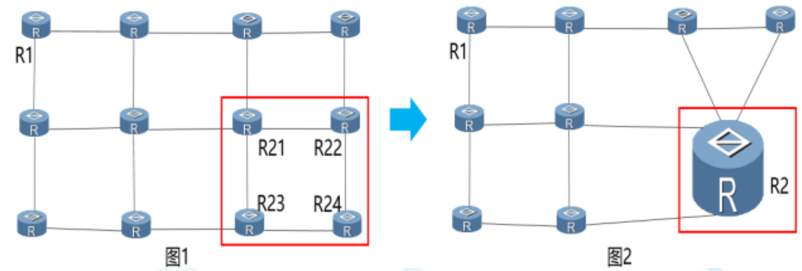
Methods:

FEC: to restore lost packets;
Packet replication;
Multipath transmission;
Congestion control algorithm.

APPLIED RESEARCH CENTER FOR COMPUTER NETWORKS

Federate DCN with auto-regulated domains





Domain size control

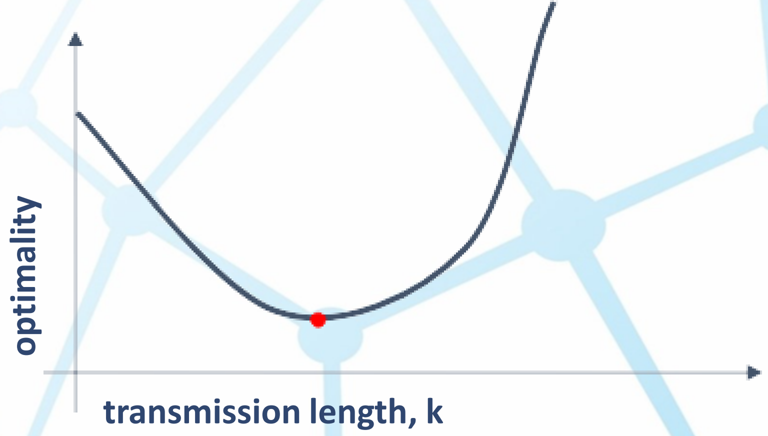
Let **k** be the length of the path along which routers will transmit information about their load:

k = 0 – they know only about their own load,

k = 1 – about themselves and neighbors,

k = k_max – know about everyone.

The more k, the more optimal the route, but the convergence time became longer.



APPLIED RESEARCH CENTER FOR COMPUTER NETWORKS

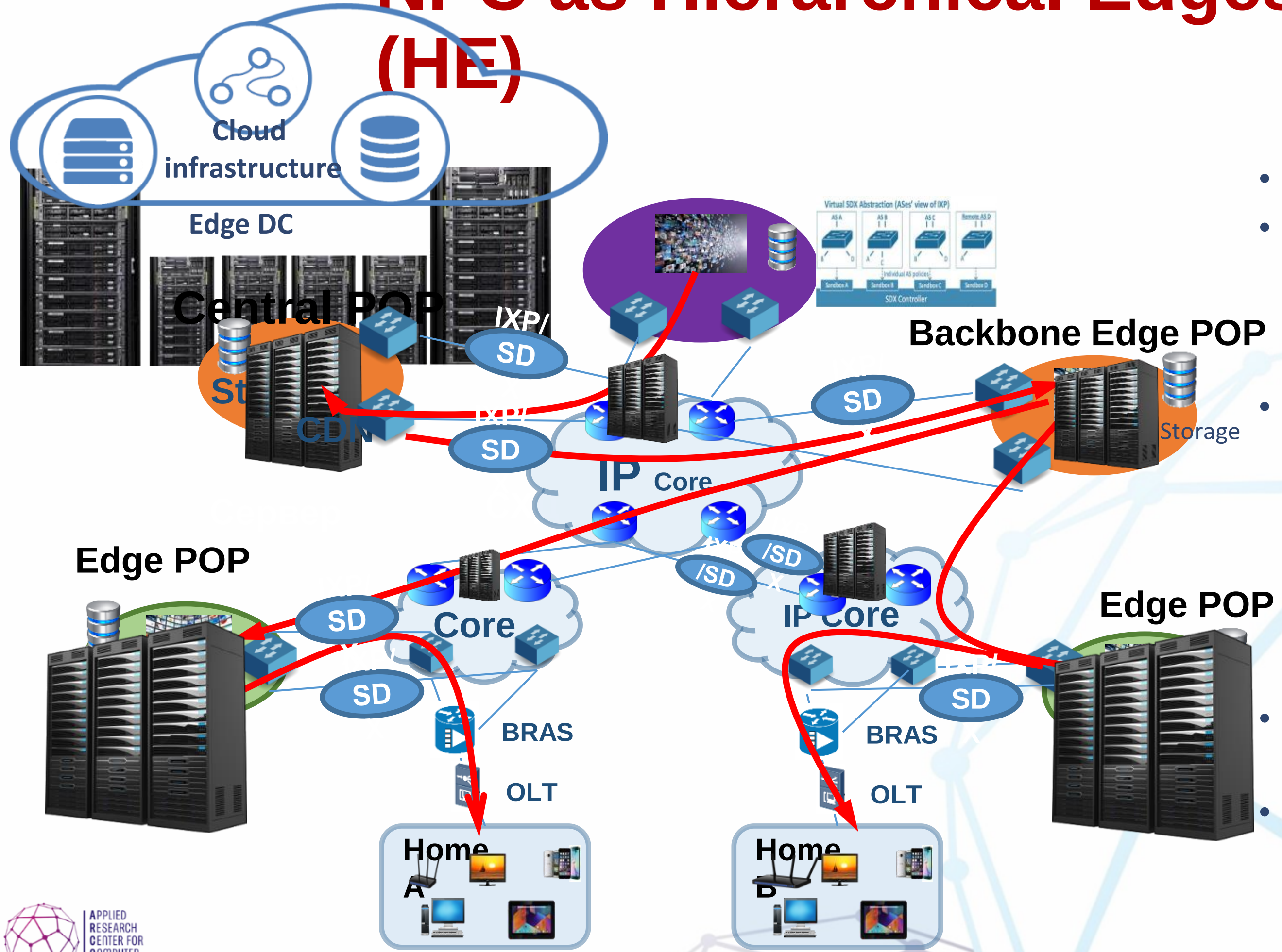
Bandwidth on Demand

The purpose of the Bandwidth on Demand service is to combine "on demand" two virtual local networks in different data centers into a single logical network with SDN-WAN management.

Implemented a fully automatic process of allocation of the required backbone network capacity "on demand", with payment for the actual time of its use.

The process of changing the allocated capacity is automated, both by creating channels with a different capacity, and by creating additional channels.

NPC as Hierarchical Edges (HE)

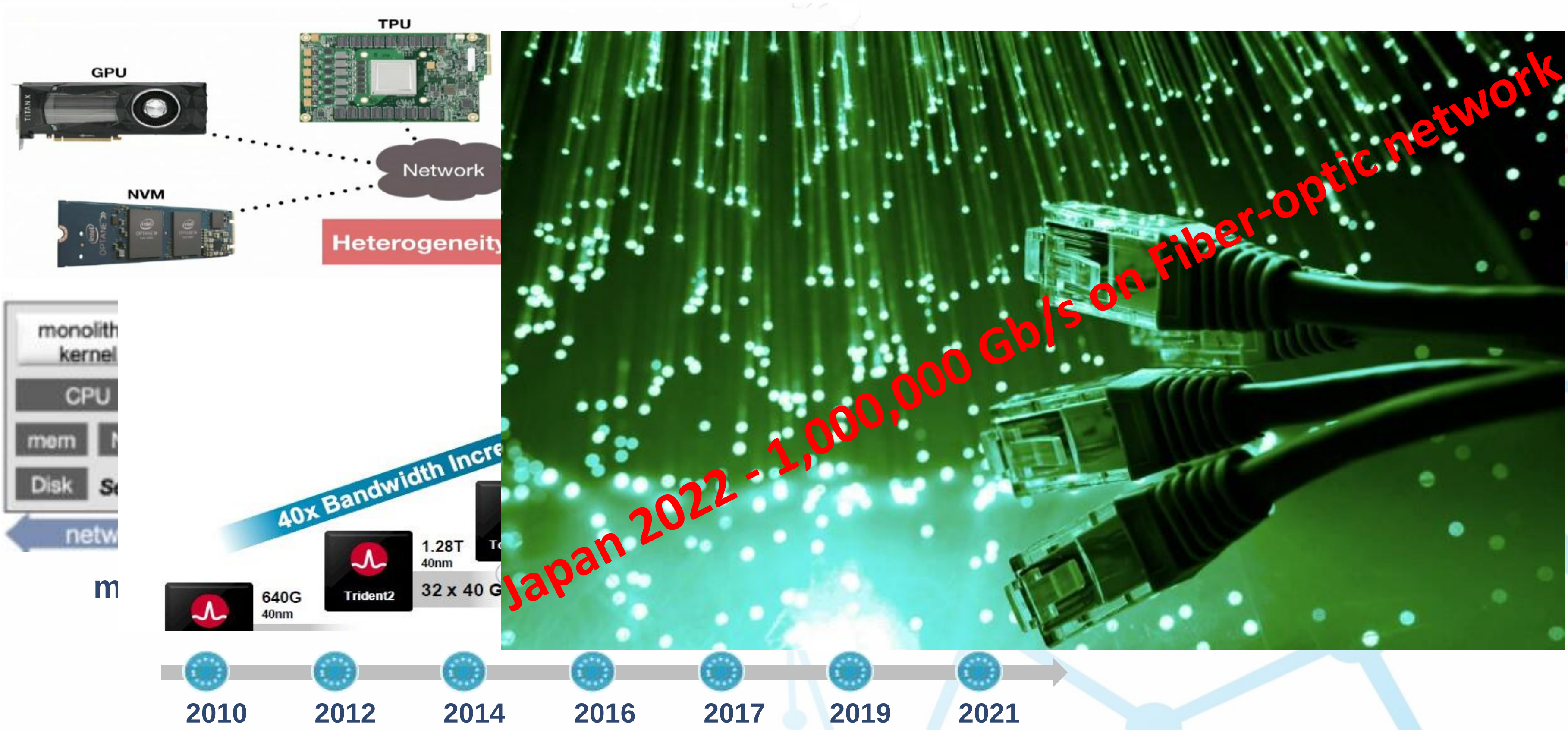


Benefits of the HE on mDC approach

- **Minimizing the latency**
- **Reduced DCN load** requirements for transport due to the proximity of the service instance to the end user;
- **Reducing energy consumption** 14% of the energy consumption in the Internet is due to the data transportation. The edge computation offloaded from traditional DC to mDC at the fringe of network
- **Easy scaling** by using a centralized cloud platform;
- **Increasing the efficiency of the network** due to a centralized management and orchestration Layer NPC

APPLIED RESEARCH CENTER FOR COMPUTER NETWORKS

Disaggregated Architecture



Conclusion

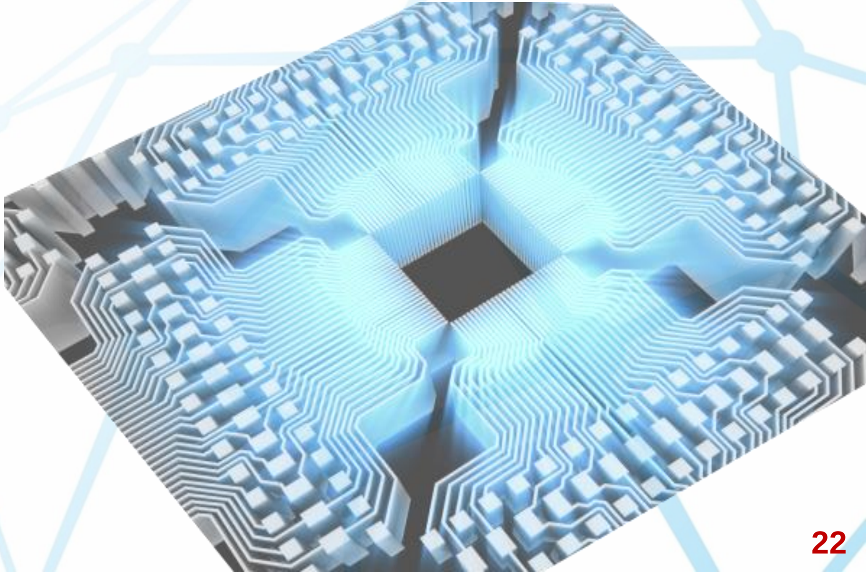
- **The Network Powered by Computing as the next generation of Computational Infrastructure was presented**
- **The Functional NPC Architecture was described**
- **Statements of the problem of SFC allocation for proactive and active modes of NPC operation were formulated.**

To bring NPCs to life we need:

- create DSL languages for Application Operation Specification;
- distributed hierarchical control methods based on ML ;
- coordinated routing on overlay and underlying networks;
- an adaptable tradeoff between centralized and decentralized control based on MA optimization;
- efficient intelligent transport in DCN (channels QoS prediction and control, FEC for packets loss);
- revise the concept of Operating System for NPC;

NPC will make our network to be Super Large Scalable Computer – with predictable behavior, secure, reliable, fault tolerant and scalable.

THANKS



Contacts: smel@cs.msu.su