Cloud computing for big data infrastructures

DR. ZHIMING ZHAO

UvA Universiteit van Amsterdam

H2020 Project

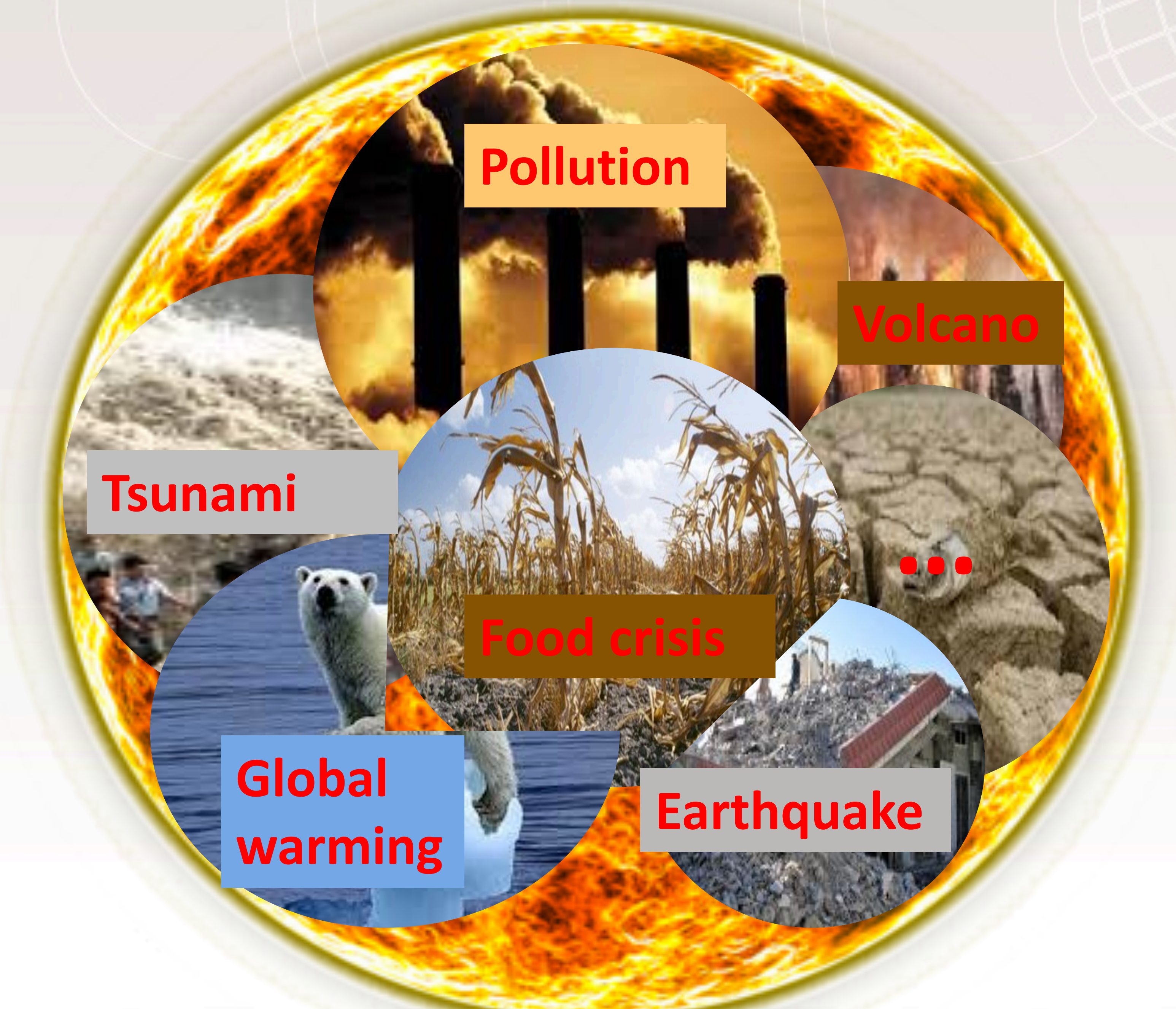
SWITCH

plus ENVRI

Environmental Research Infrastructures Providing Shared Solutions for Science and Society

VRE 4 EIC

Environmental challenges



Outline

- Infrastructures for data centric research
- Infrastructure interoperability
- Time critical cloud applications
- Summary

Outline

- **Infrastructures for data centric research**
- Infrastructure interoperability
- Time critical cloud applications
- Summary

From observation to data management

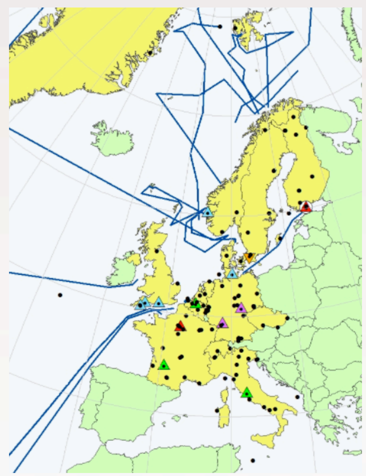
DATA

content

context

Description of the observation, and collected data etc........

Observation

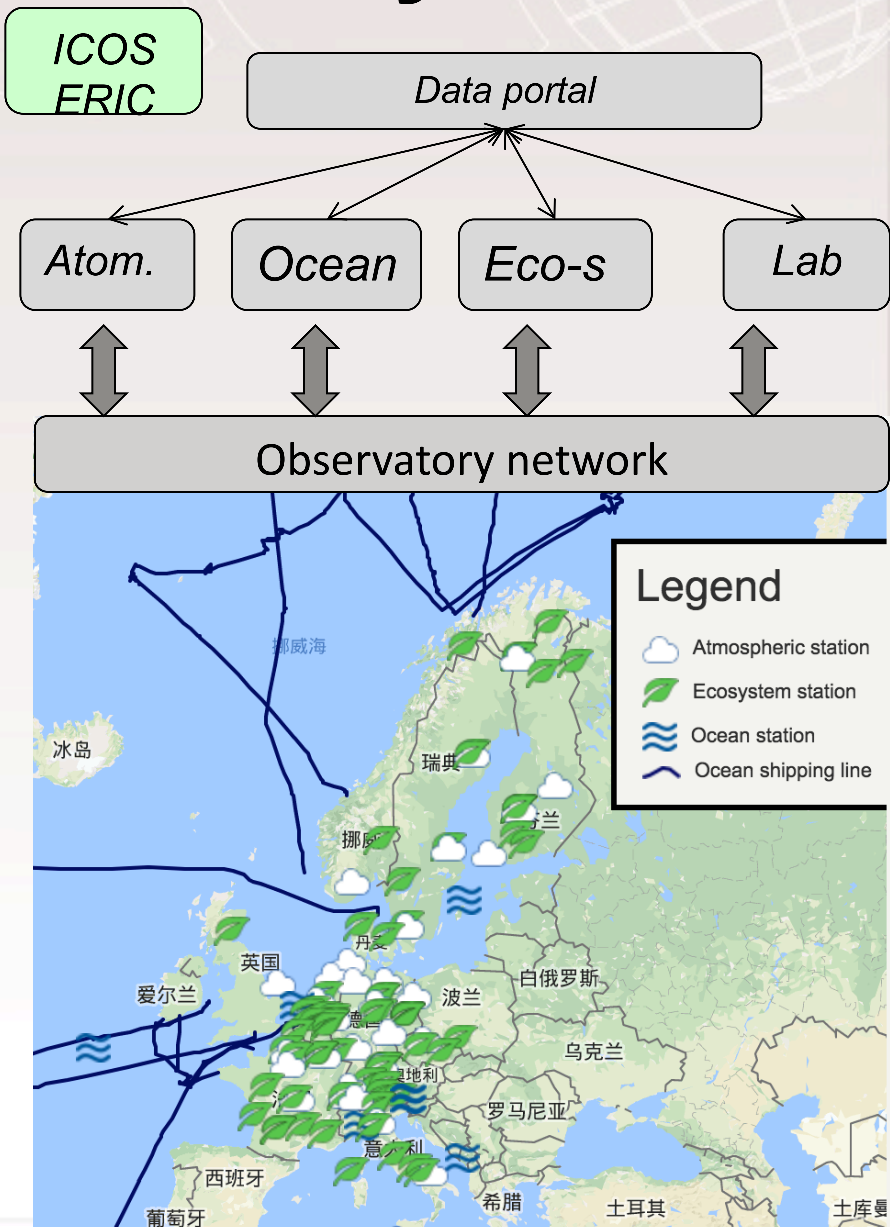


map / image / database

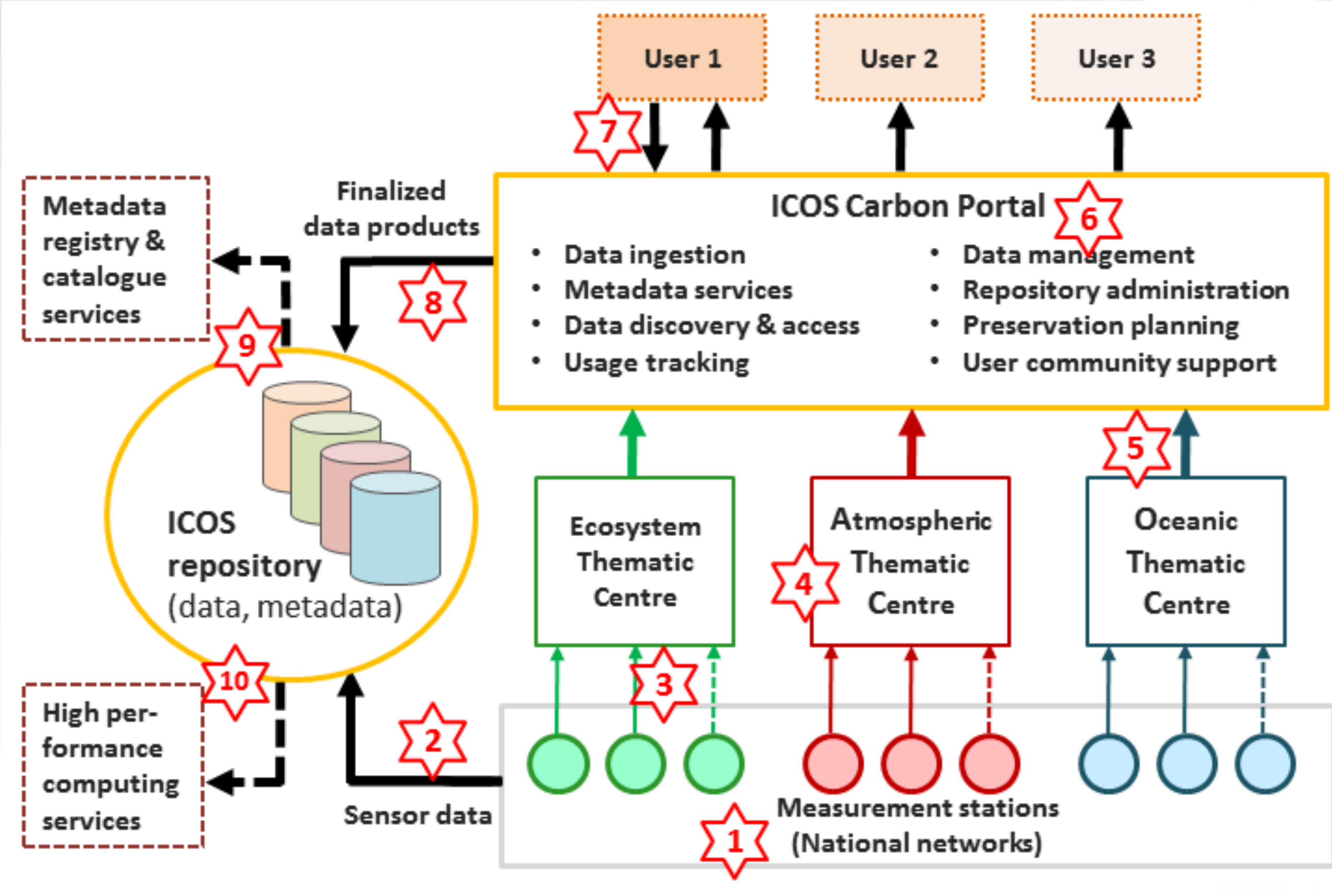
description

Integrated Carbon Observation System

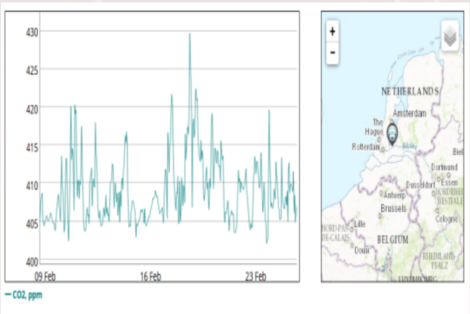
- More than 100 observation station in 12 EU countries
- Data acquisition:
  - Green gas
- Data management
  - 4 thematic centers
  - Via Data portal, users can also submit data results
- Governance
  - Legal entity: European Research Infrastructure Consortium

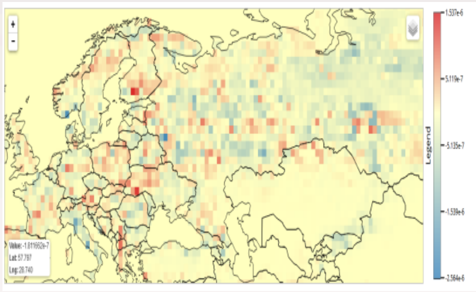


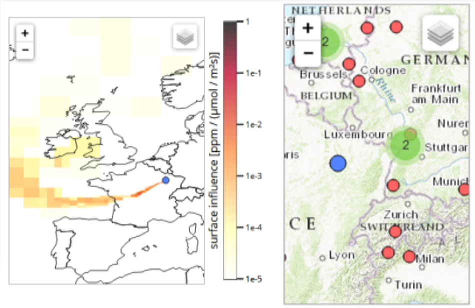
How ICOS manages data



ICOS data products







- Raw observational data (Level 0)

  For specialists; available on request
- Near Real-Time data (Level 1)

  For specialists; available on request

  Mainly automated quality control & processing

  Time delay & parameter scope varies
- QA/QC:ed & aggregated data (Level 2)

  Main data product – time series of 30-60 min means

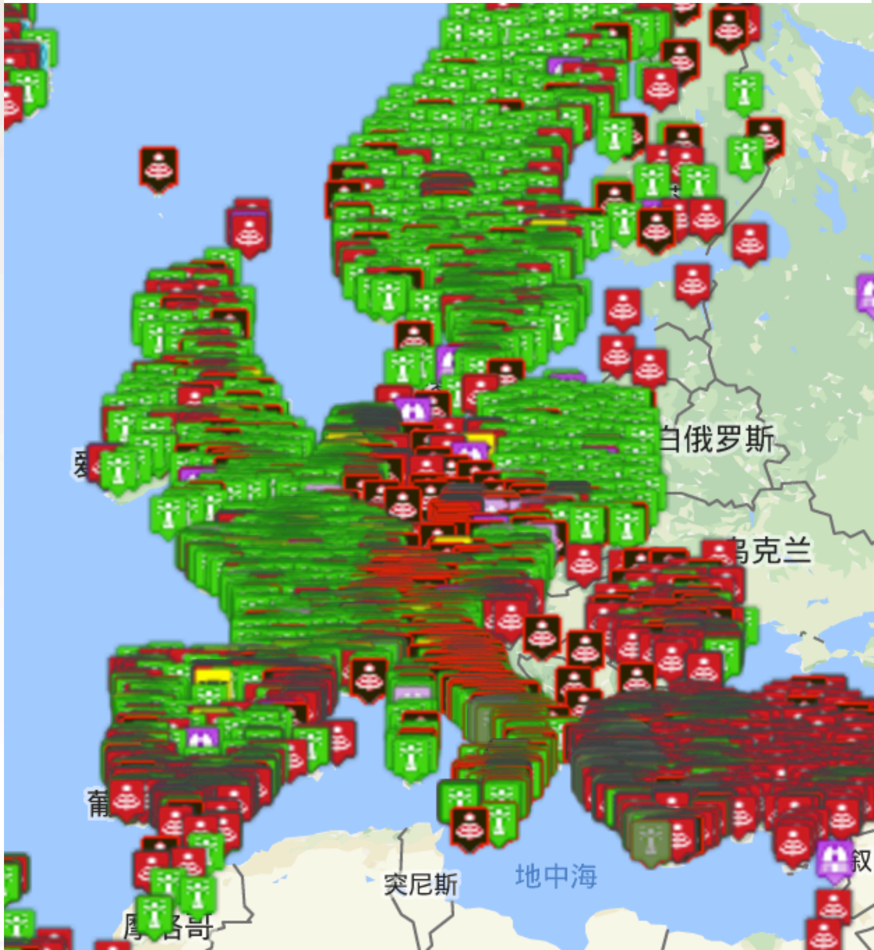
  Full parameter sets available
- Elaborated data (Level 3)

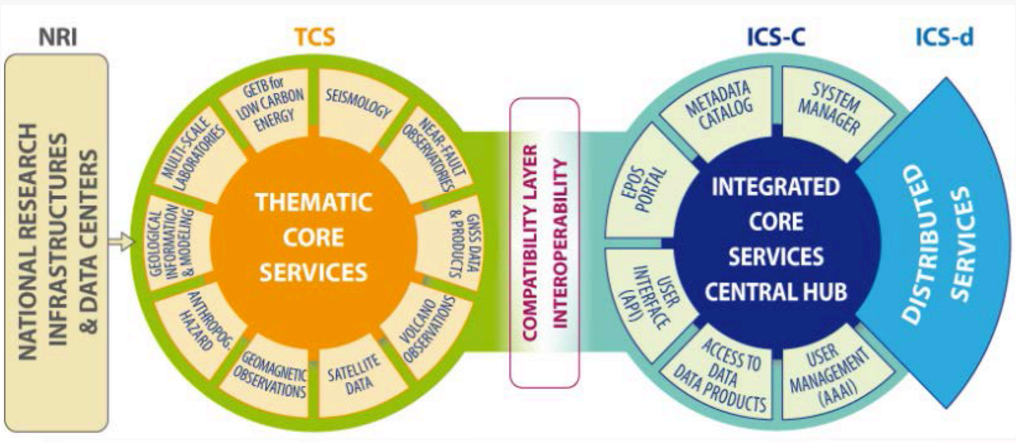
  Model calculation outputs (atmospheric & ecosystem)

  Mainly contributed by ICOS data end users

European Plate Observation System

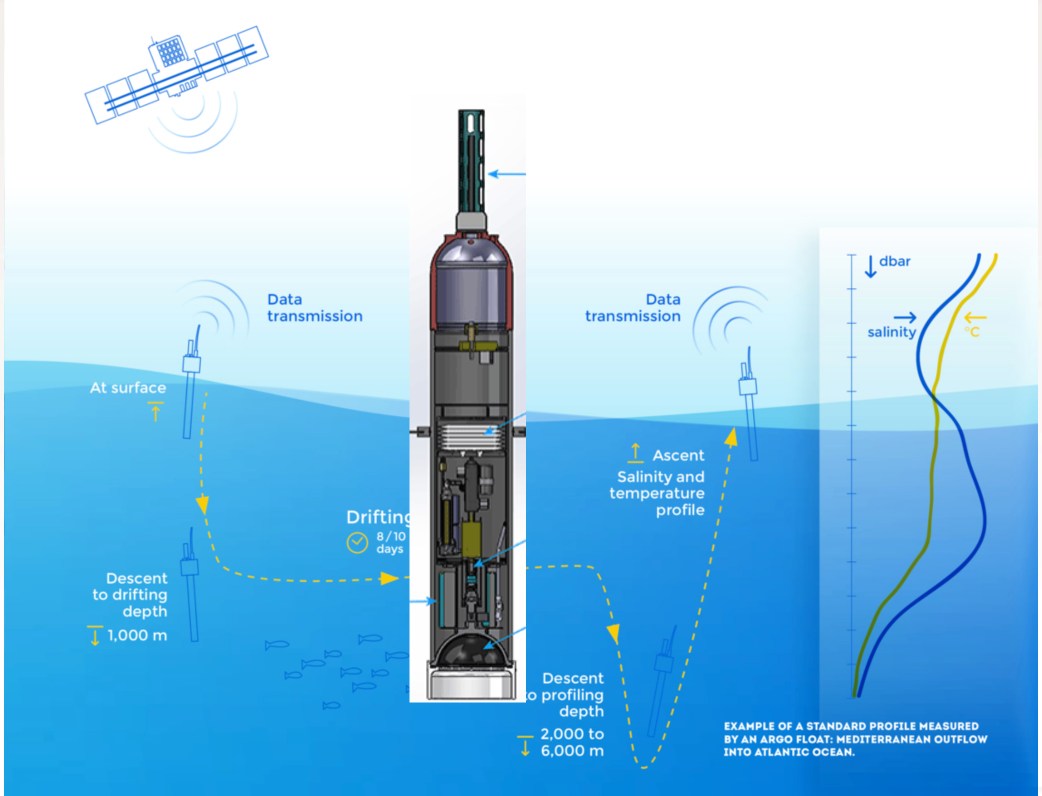
- More than 4000 monitoring station
- 10 key thematic services:
- Seismology, Near-Fault Observatories, GNSS Data and Products, Volcano Observations Satellite Data, Geomagnetic Observations, Anthropogenic Hazards, Geological Information and Modeling, Multi-Scale Laboratories, Geo-Energy Test Beds for Low Carbon Energy

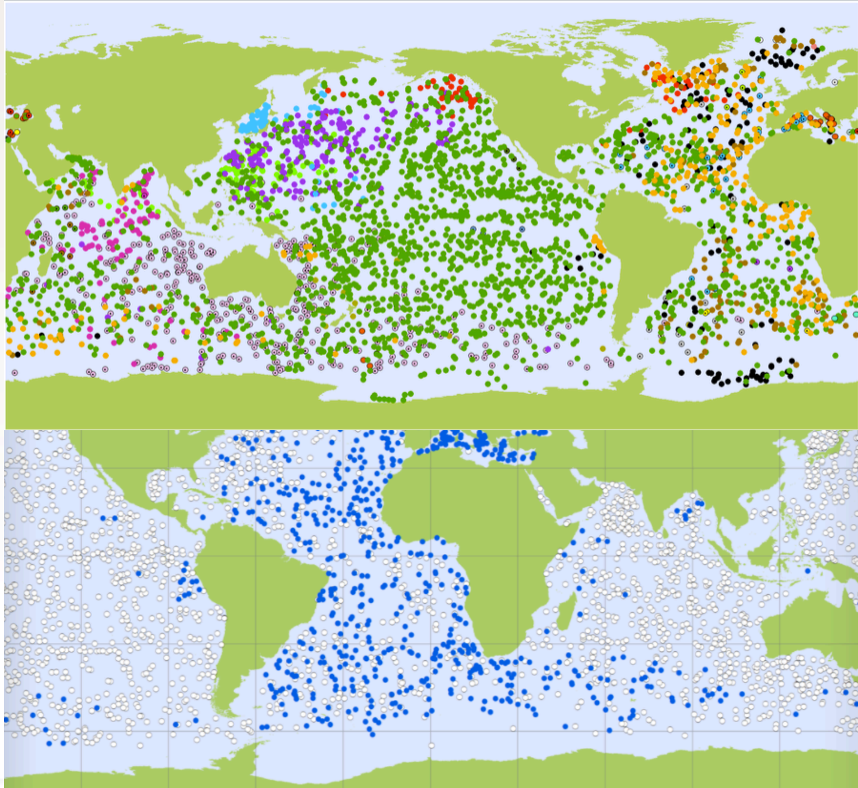




Euro-Argo, European contribution to the Argo program

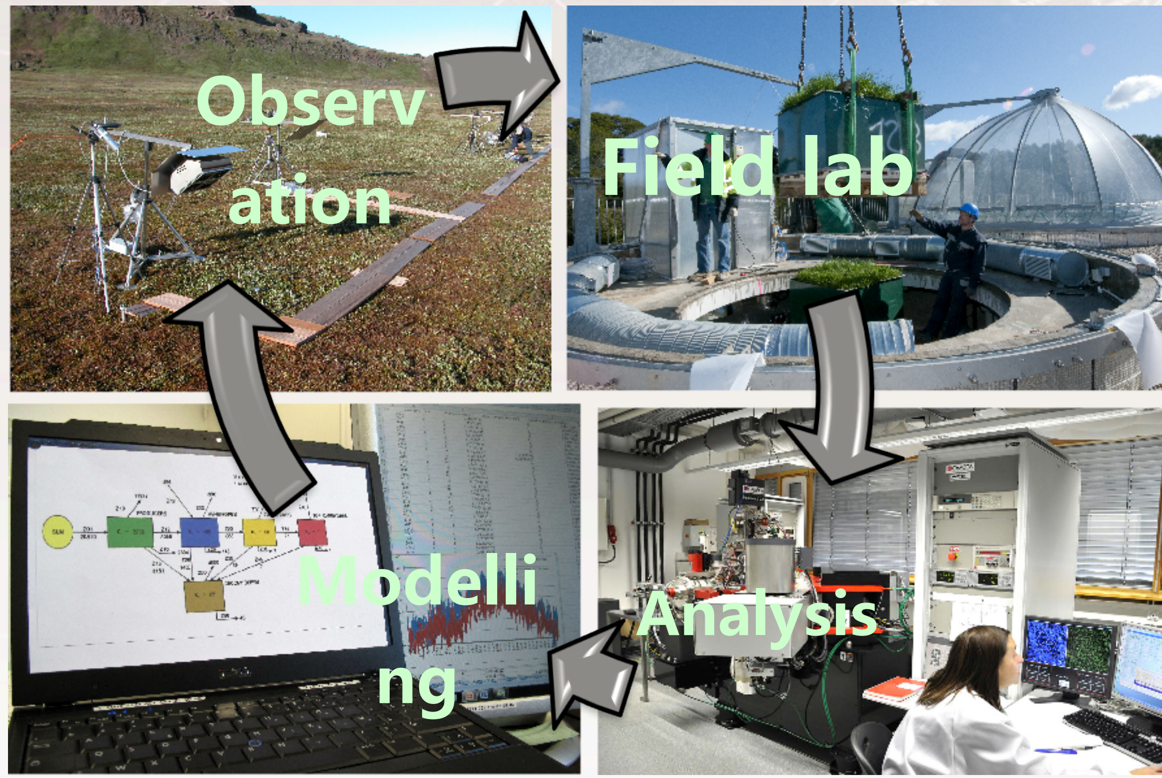
- The international Argo program was initiated in 1999
- The Argo network is a global array of more than 3500 autonomous instruments,
- 12 European countries gathered in 2008 within the Euro-Argo project
- European contribution to Argo by deploying 250 floats per year.





Analysis and experimentation on Ecosystems

- Different platforms across 17 EU countries
  - Open fields 109
  - In house 45
  - Data analysis 35
  - Modelling platform 9
- Food safety, agriculture



ENVRIplus Horizon 2020 project
Grant agreement No 654182

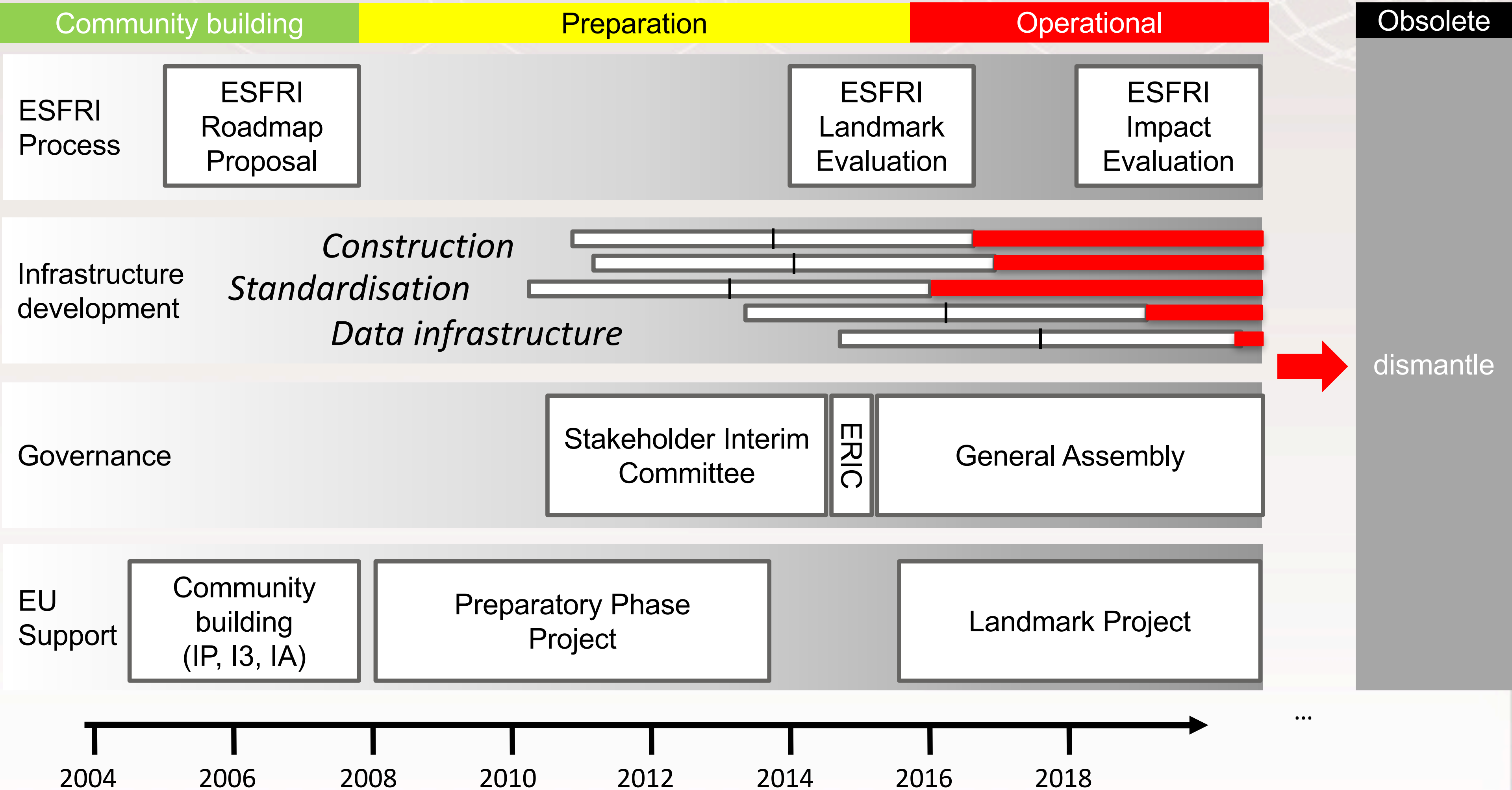
RISCAPE Horizon 2020 project
Grant agreement No 730974

Research infrastructure

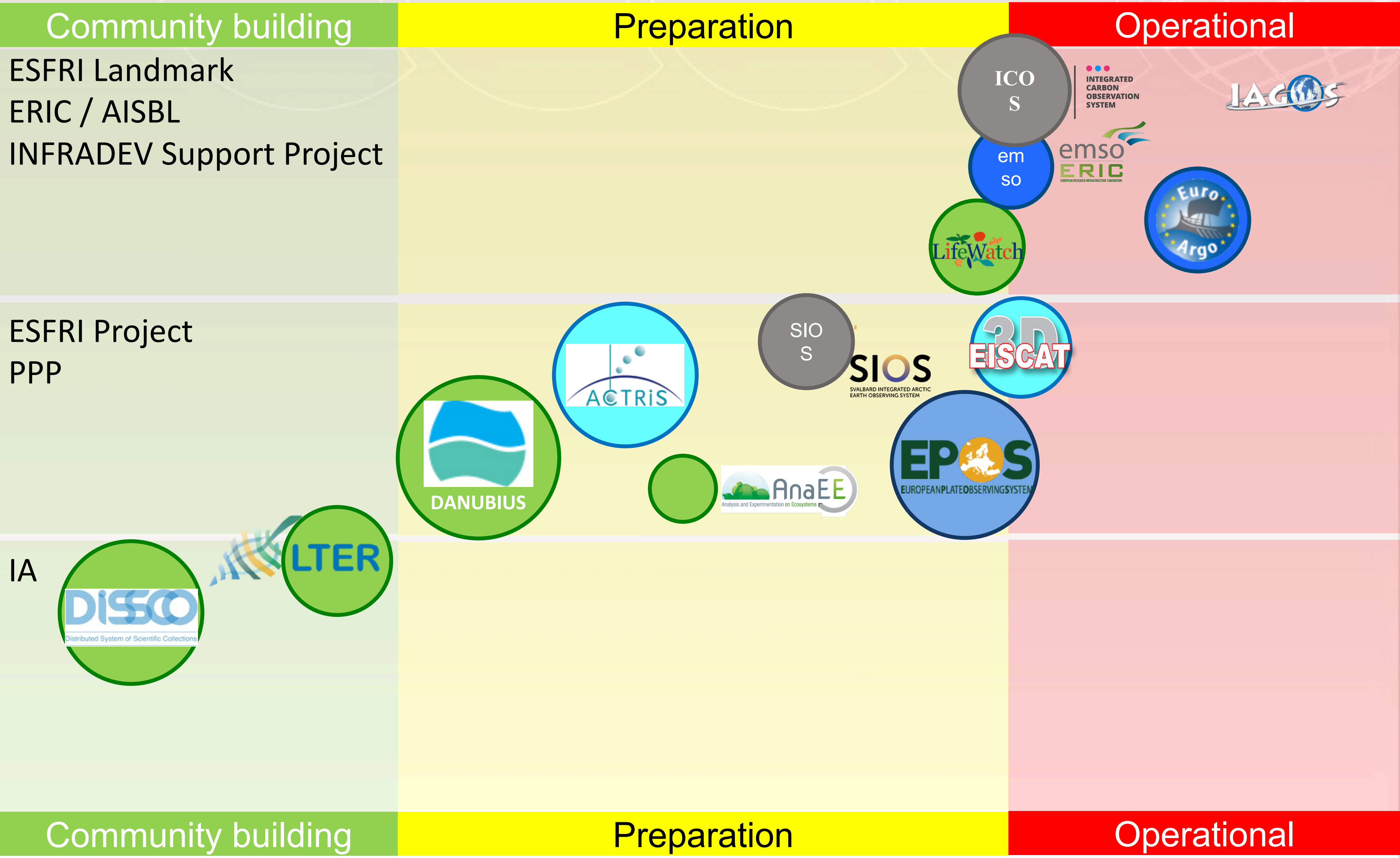
- **Research infrastructures (RIs)** are facilities, resources and services used by the science community to conduct research and foster innovation.
- Developed as single site, distributed or virtual (RI)

Humanity (5)	Life science(13)		Environmental sciences (9)		Energy (7)	Material (6)	Astronomy and physics (10)		e-Infra-structures (1)
SHARE	BBMRI	ELIXIR	ICOS	EURO-ARGO	ECCSEL	EUROFEL	ELI	TIARA*	PRACE
European Social Survey	ECRIN	INFRA FRONTIER	LIFEWATCH	IAGOS	Windscanner	EMFL	SPIRAL2	CTA	
CESSDA	INSTRUCT	EATRIS	EMSO	EPOS	EU-SOLARIS	European XFEL	E-ELT	SKA	
CLARIN	EU-OPENSCREEN	EMBRC	SIOS	EISCAT_3D	JHR	ESRF Upgrade	KM3NeT	FAIR	
DARIAH	Euro BioImaging	ERINHA BSL4 Lab		COPAL	IFMIF	NEUTRON ESS	SLHC-PP*	ILC-HIGRADE*	
	ISBE	MIRRI			HiPER	ILL20/20 Upgrade			
	ANAEE				MYRRHA				

The life cycle of a Research Infrastructure



The Research Infrastructure Landscape 2017



Outline

- Infrastructures for data centric research
- **Infrastructure interoperability**
- Time critical cloud applications
- Summary

Data for science theme: motivation

Data, services, workflows from multiple domains

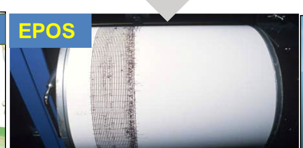
EISCAT 3D

• Upper space observing

EMSO

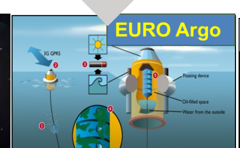
• Deep sea observing

EPOS



• Volcano, earthquake observing

EURO Argo



• Open sea observing

IAGOS

• Aircraft for global observing

ICOS

• Greenhouse observing

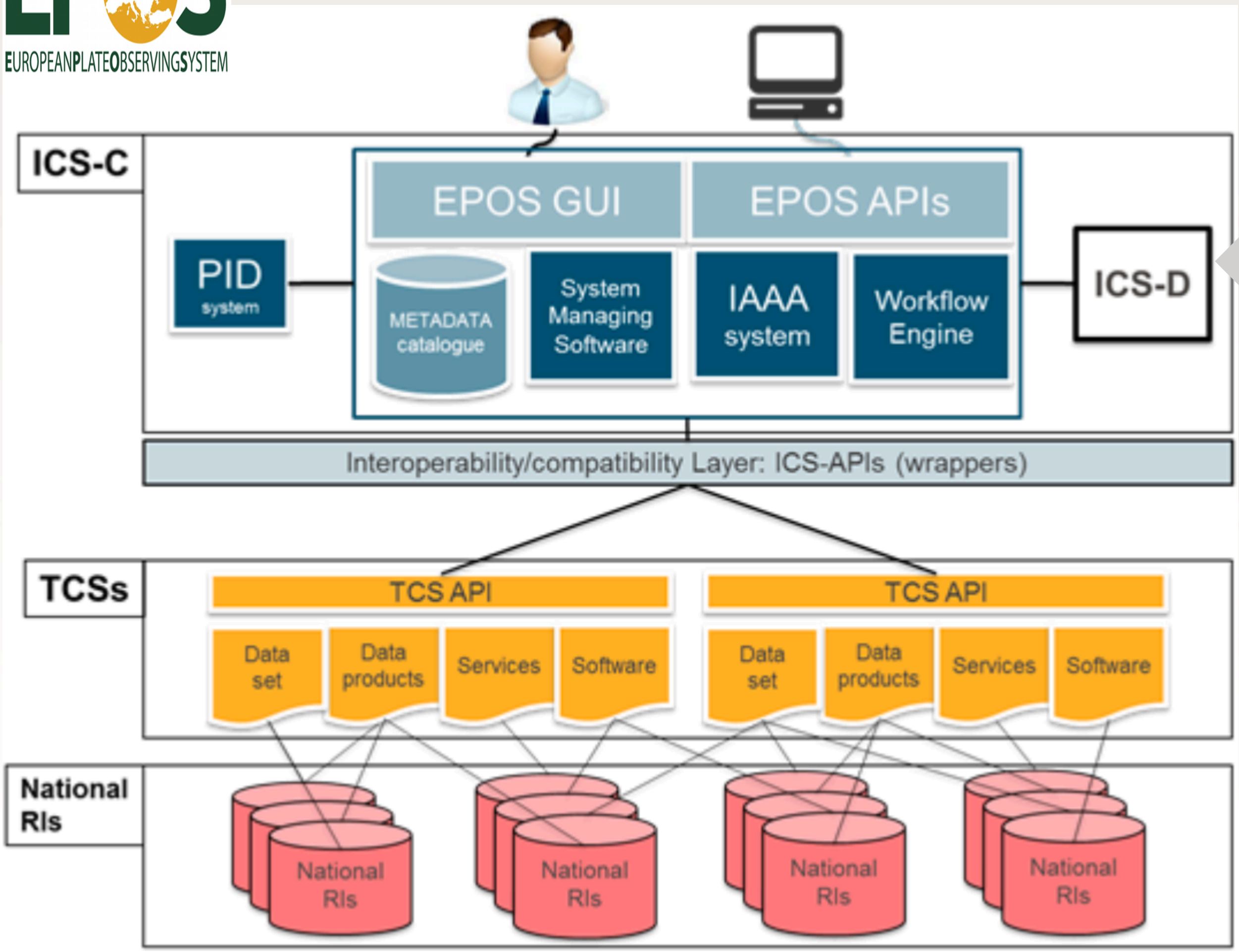
SIOS

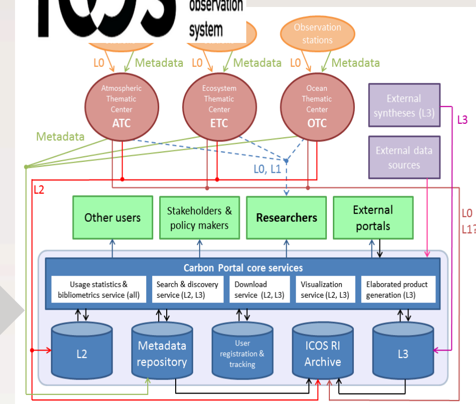
• Svalbard arctic Earth observing

……

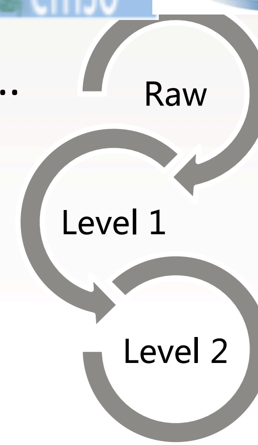
Curation: quality control, annotation …

Nearly real-time Data handling,

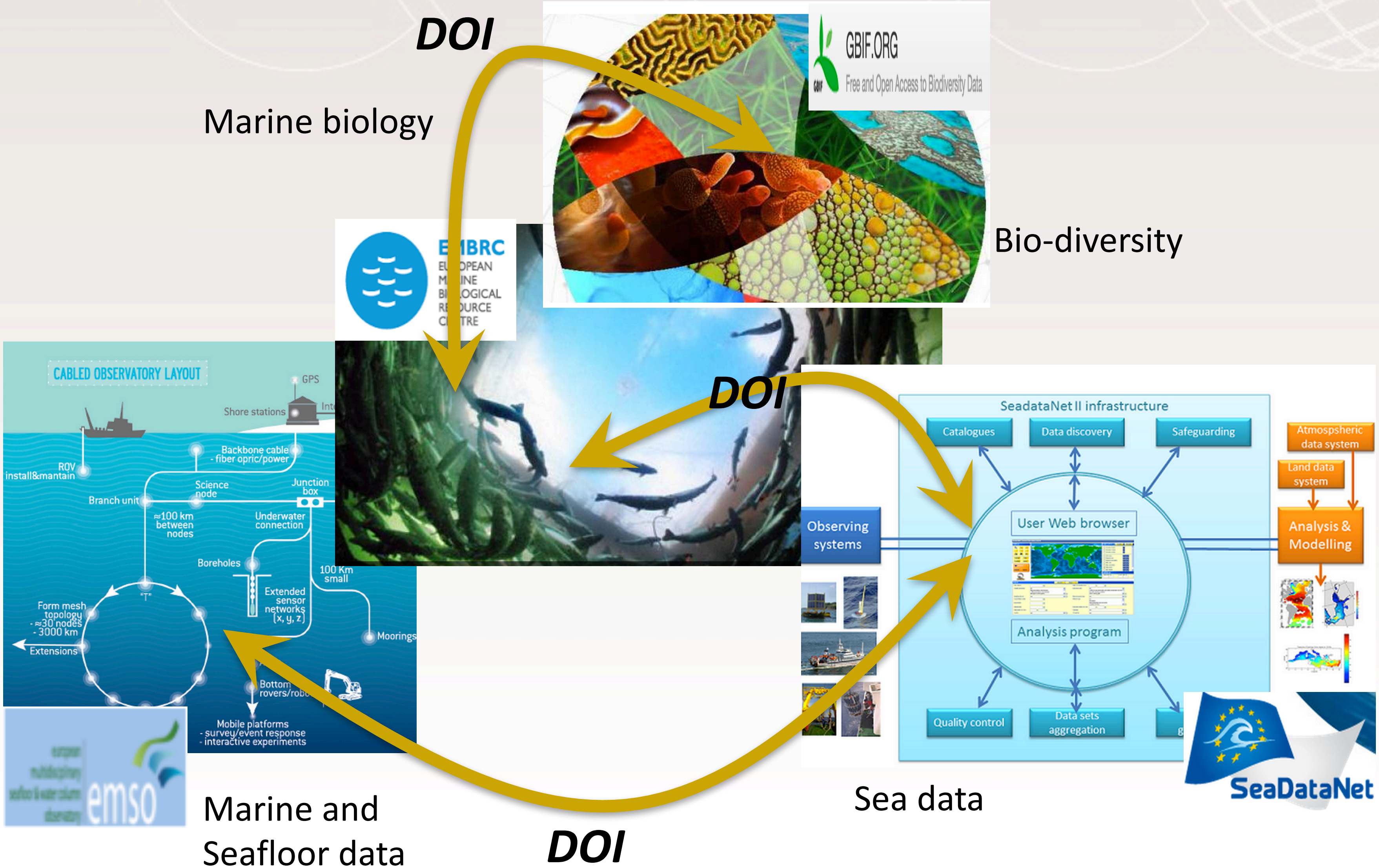




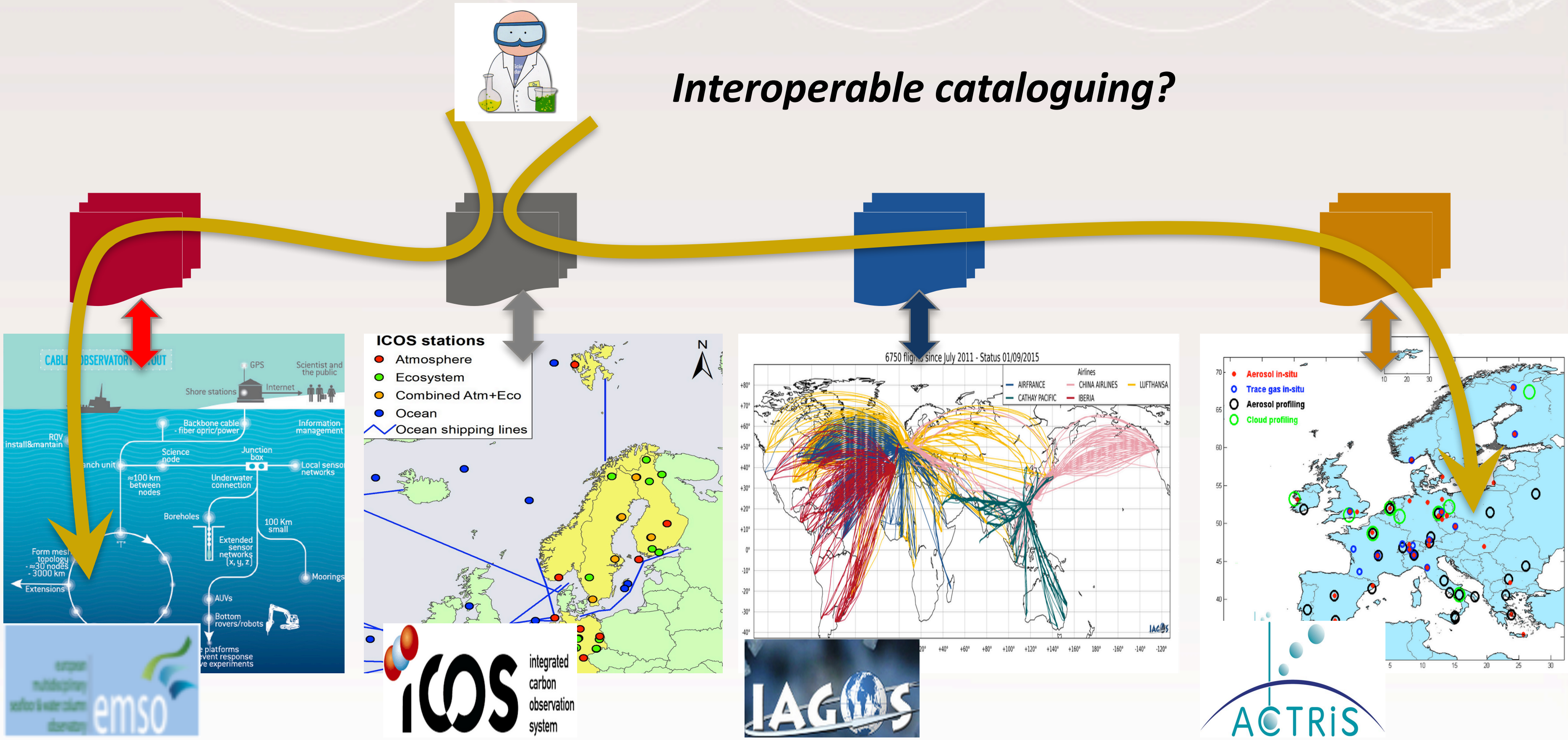
....



Data identification and citation



Data cataloguing



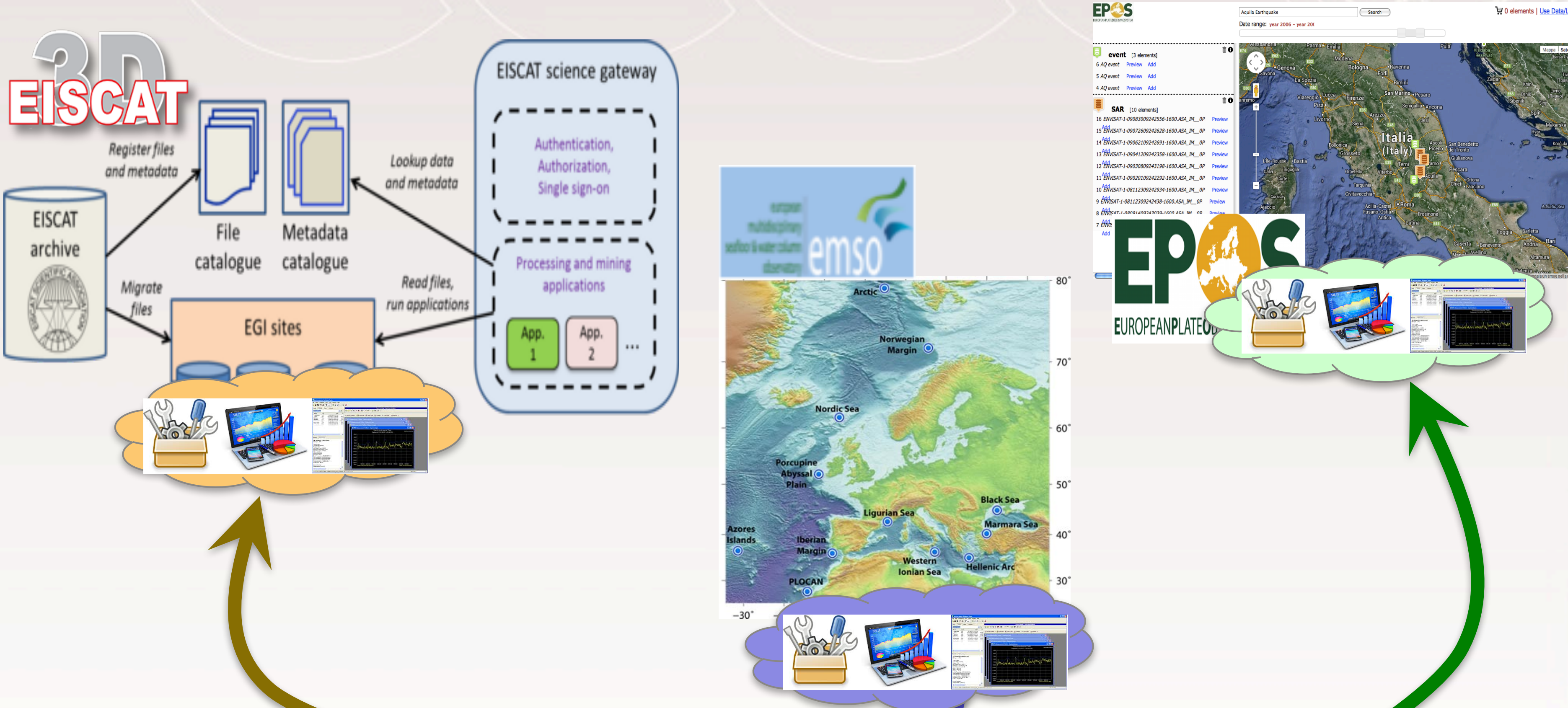
Marine and Seafloor data

Carbon, ocean, eco-sys

Cloud particles, atmosphere composition

Gas-sphere spices, etc.

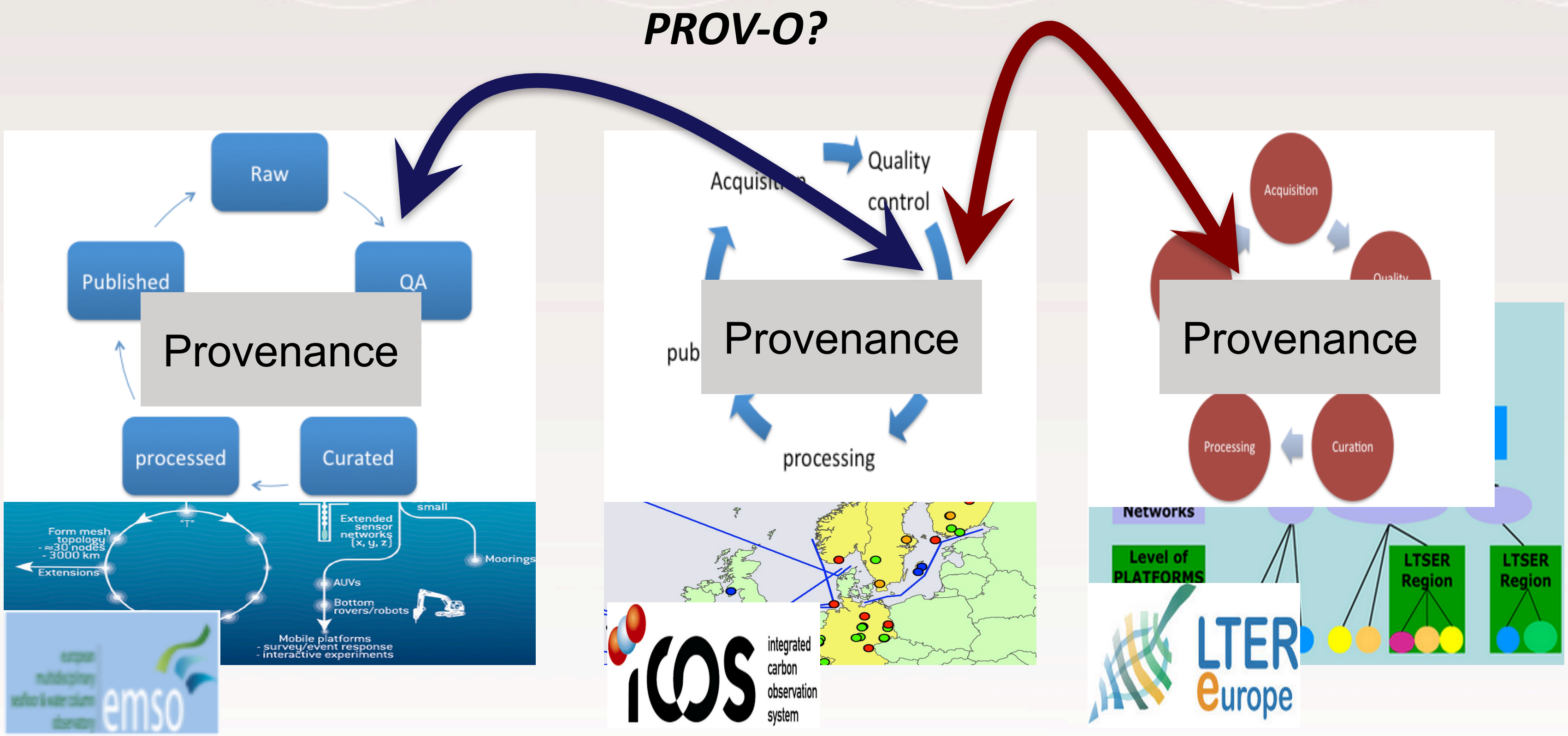
Processing, monitoring, diagnosis and optimization



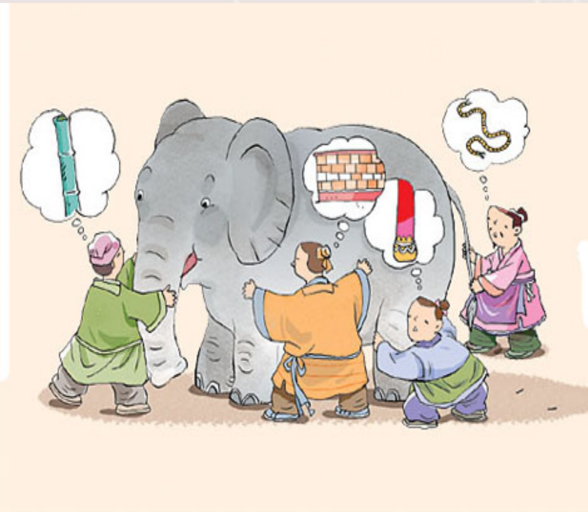
How to optimize: data discovery, access, delivery and processing.

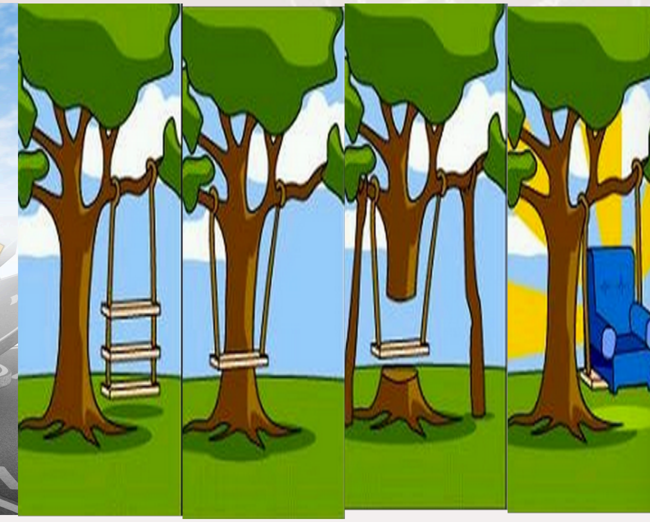
Tools: processing, monitoring, and diagnosis, virtual research environments

Data provenance



Interoperability challenges



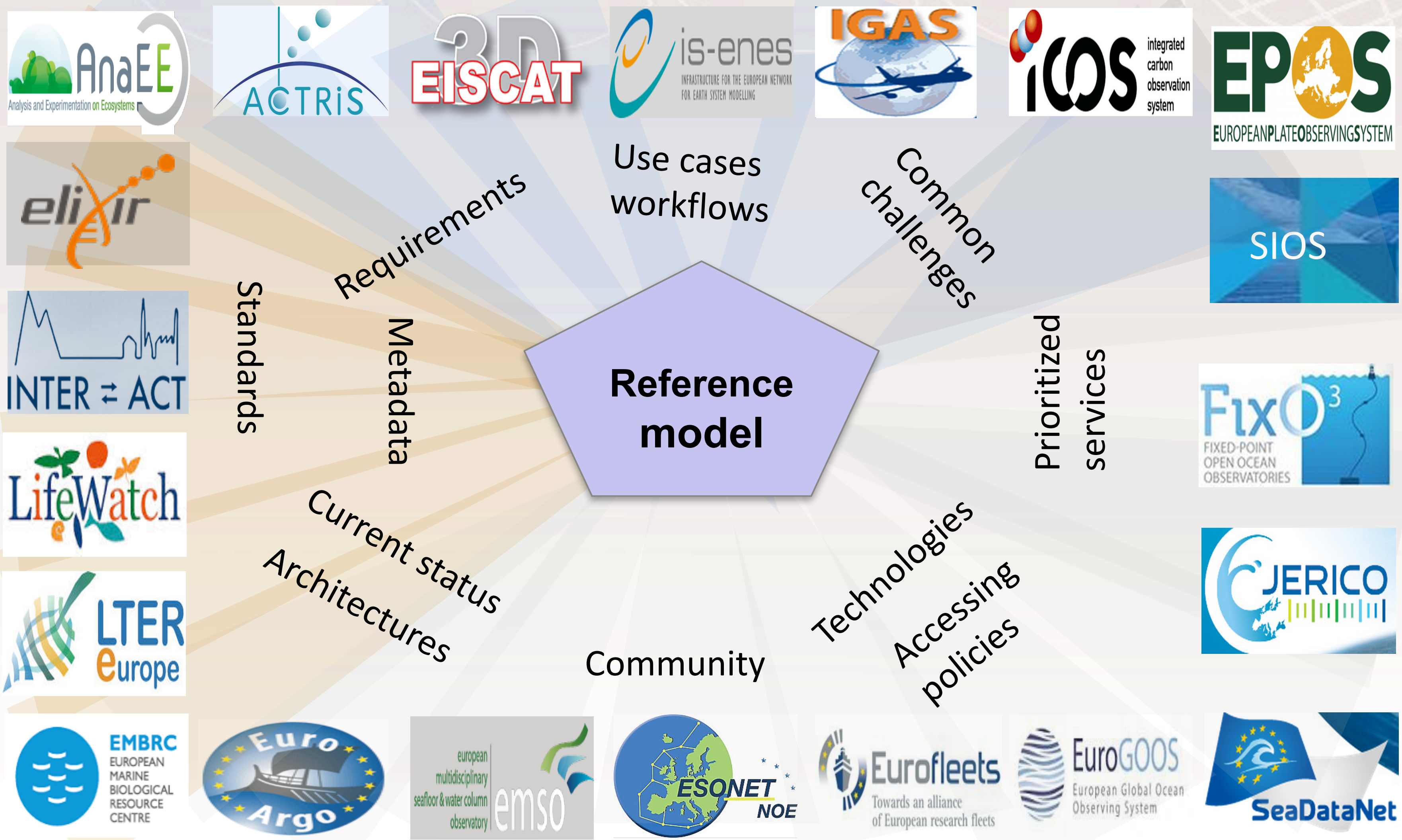


- **Diverse visions, different development agenda, and the lack of common vocabulary** make the data and functional components difficult to be interoperable.

Reference model

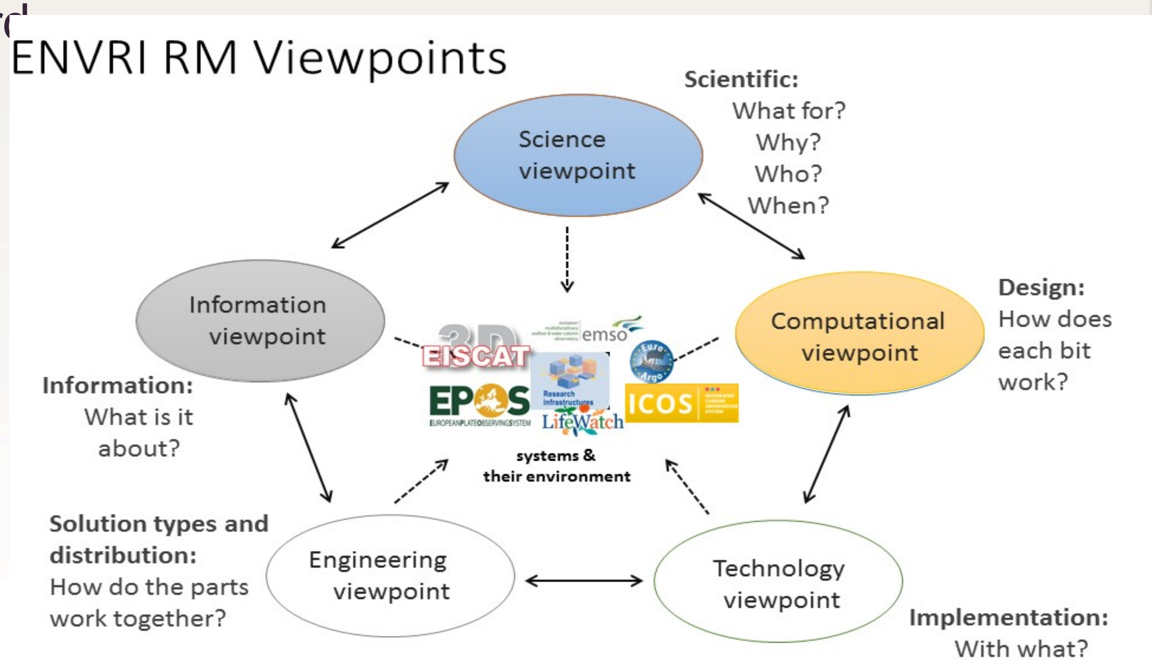
- ISO: **Open System Interconnection(OSI)** Reference model
- ISO: Software engineering, **Open Distributed Processing (ODP)** reference model
- OASIS: Service computing, **Service Oriented Architecture (SOA)**
- WfMC: **Workflow Management** Coalition reference model
- Data curation: **Digital Curation** Center Reference Model

Reference model



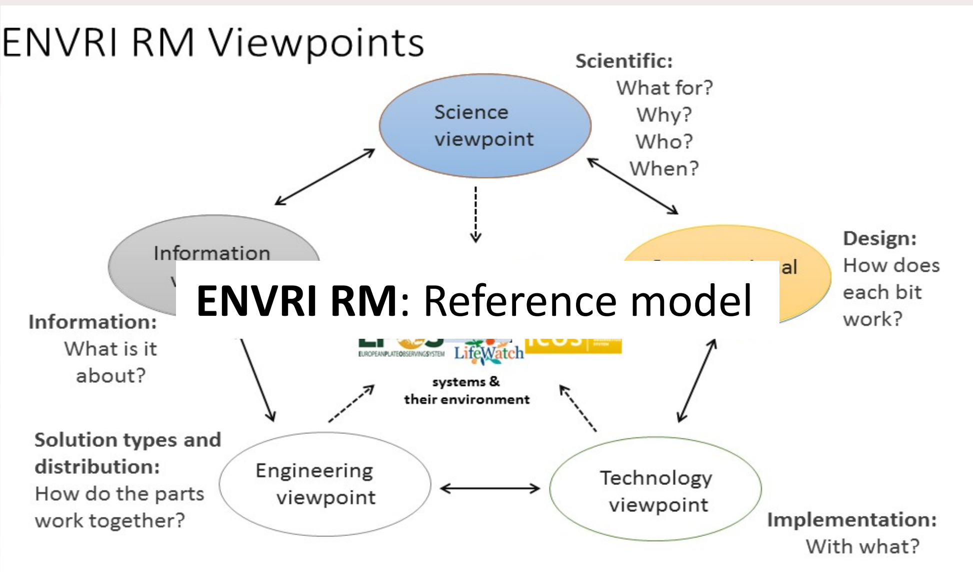
Reference model

- **The ENVRI Reference Model (RM)**, started in the former ENVRI project, is an ongoing development of ENVRIplus that can be applied to the informatics engineering design challenges faced by RIs.
- Based on the **Open Distributed Process** (RM-ODP, ISO 10746).
- **Like the design of buildings**, the RM decomposes the modelling procedure for a complex distributed system into different viewpoints.
- **The RM promotes** a standard vocabulary for **describing** environmental research infrastructures and for **modelling** their components and architecture from the ***scientific***, ***informational***, ***computational***, ***engineering*** and ***technology*** viewpoints.
- Current version: V2.1 http://envri.eu/rm

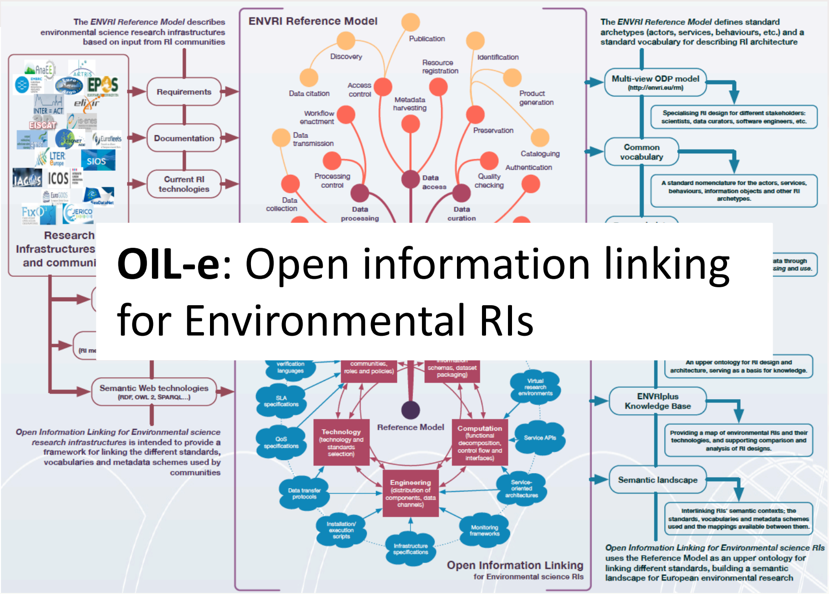


EU FP7 ENVRI: **Understand Common challenges and requirements**

Approach: multi viewpoint modelling, aims at a common ontological framework



ENVRI RM: Reference model



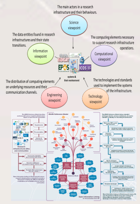
OIL-e: Open information linking for Environmental RIs

1. Chen, Y., et al., (2013) ***A common reference model for environmental science research infrastructures***. In Proceedings of EnviroInfo2013.

2. Zhao. Z, et al., (2015) ***Open Information Linking for Environmental Research Infrastructures***. In the proceedings of IEEE eScience [doi:10.1109/eScience.2015.66]

EU FP7 ENVRI:
Understand Common challenges and requirements

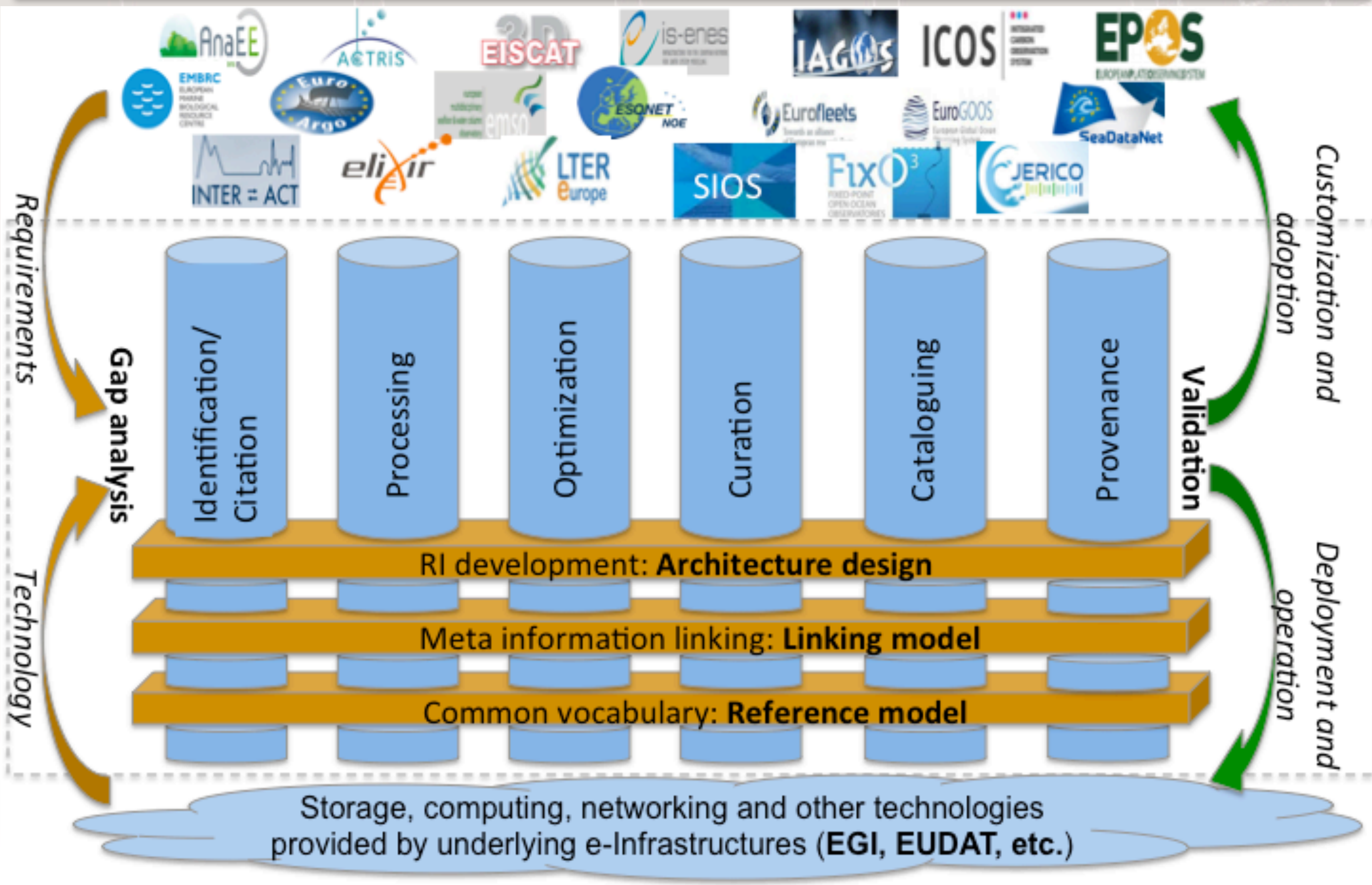
Approach: multi viewpoint modelling, aims at a common ontological framework



Output: ENVRI RM 1.0, OIL-e 1.0
Data query and processing prototype

EU H2020 ENVRIPLUS:
Data for science theme
Build reusable solutions to common development challenges.

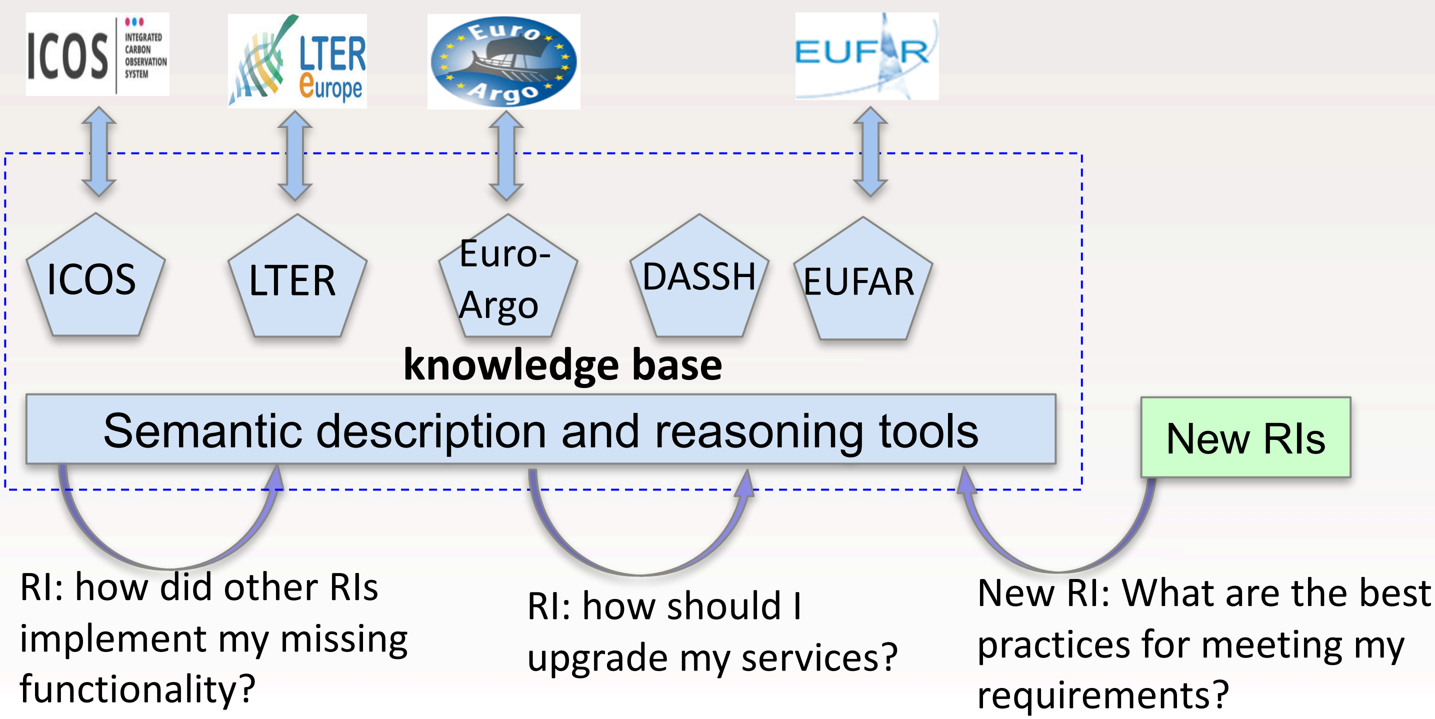
Approach: RM guided system design



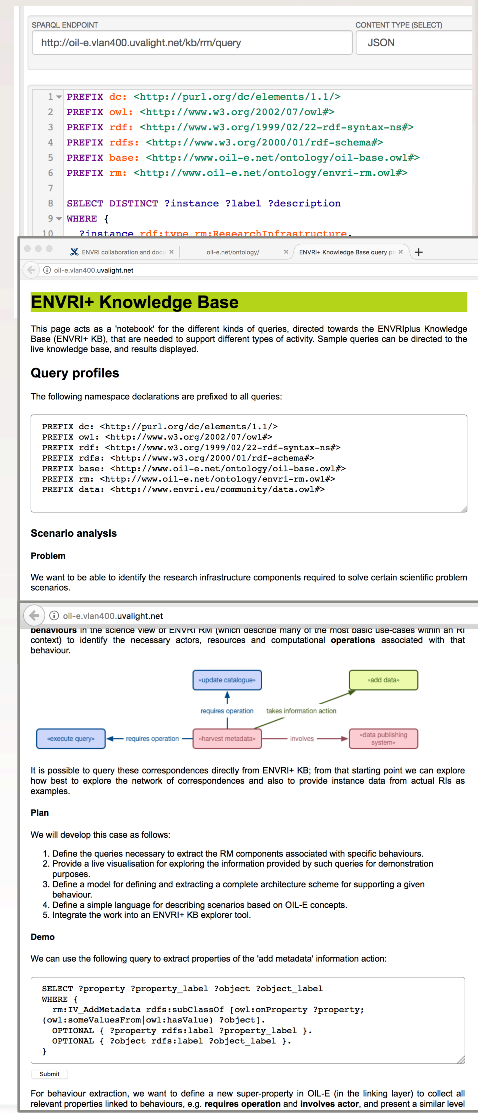
Zhao, Z., et al. (2015) ***Reference Model Guided System Design and Implementation for Interoperable Environmental Research Infrastructures***, proceedings of IEEE eScience, 2015 [doi:10.1109/eScience.2015.41]

Data for science theme knowledge base

- **Discover** reusable components among research infrastructures.
- **Design** new research infrastructures.
- **Optimise** the evolutionary path.



http://oil-e.vlan400.uvalight.net/



EU FP7 ENVRI: Understand Common challenges and requirements

Approach: multi viewpoint modelling, aims at a common ontological framework

Output: ENVRI RM 1.0, OIL-e 1.0

Data query and processing

EU H2020 ENVRIPLUS:

Current challenges:

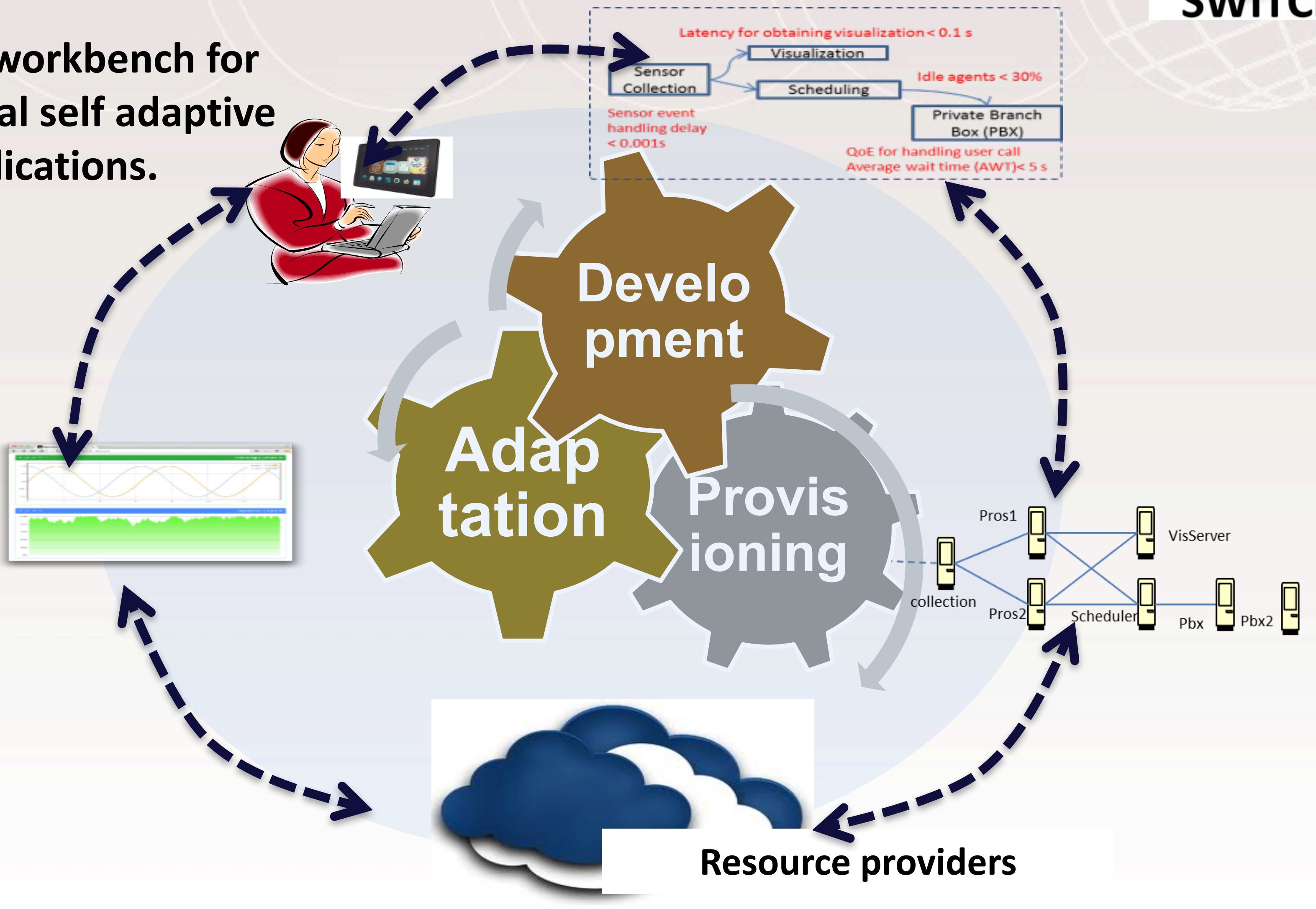
1) *Operational challenges*: AAI, deployment, maintenance
2) *Science support challenges*: effective VREs, discovery, optimization
3) *Sustainability challenges*: software upkeep, long term contracts for service provisioning?

Outline

- Infrastructures for data centric research
- Infrastructure interoperability
- **Time critical cloud applications**
- Summary

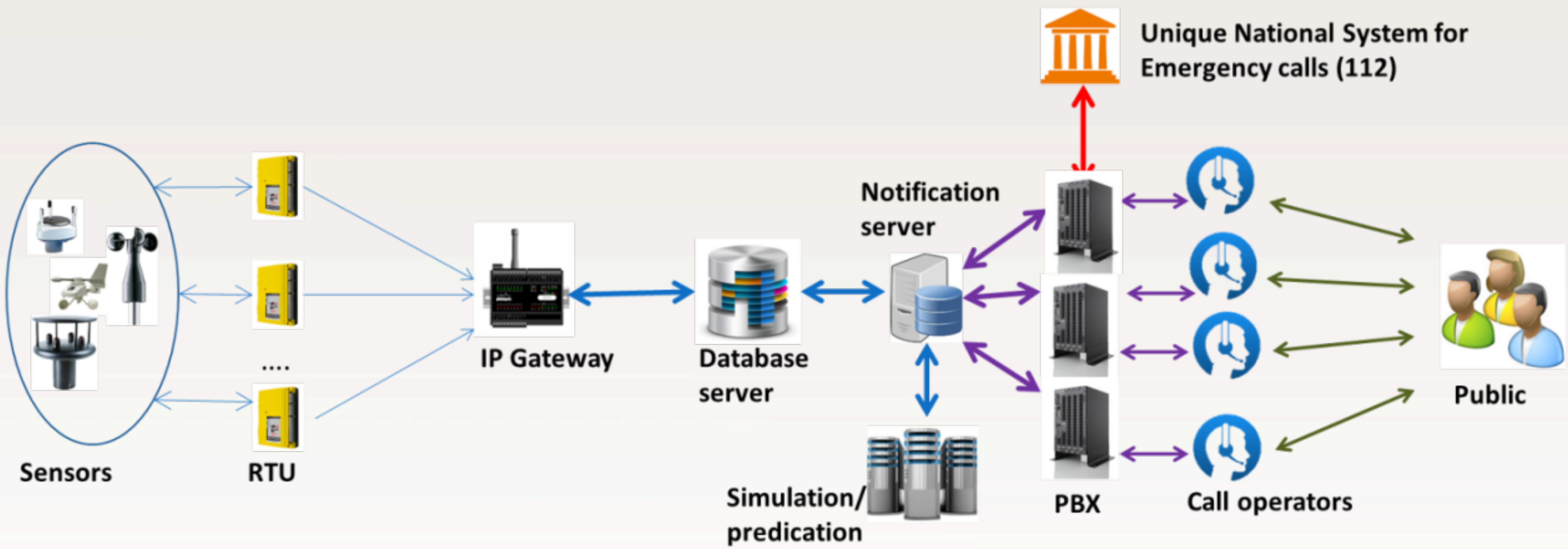
Quality critical data processing on virtualized infrastructures

Software workbench for time critical self adaptive cloud applications.



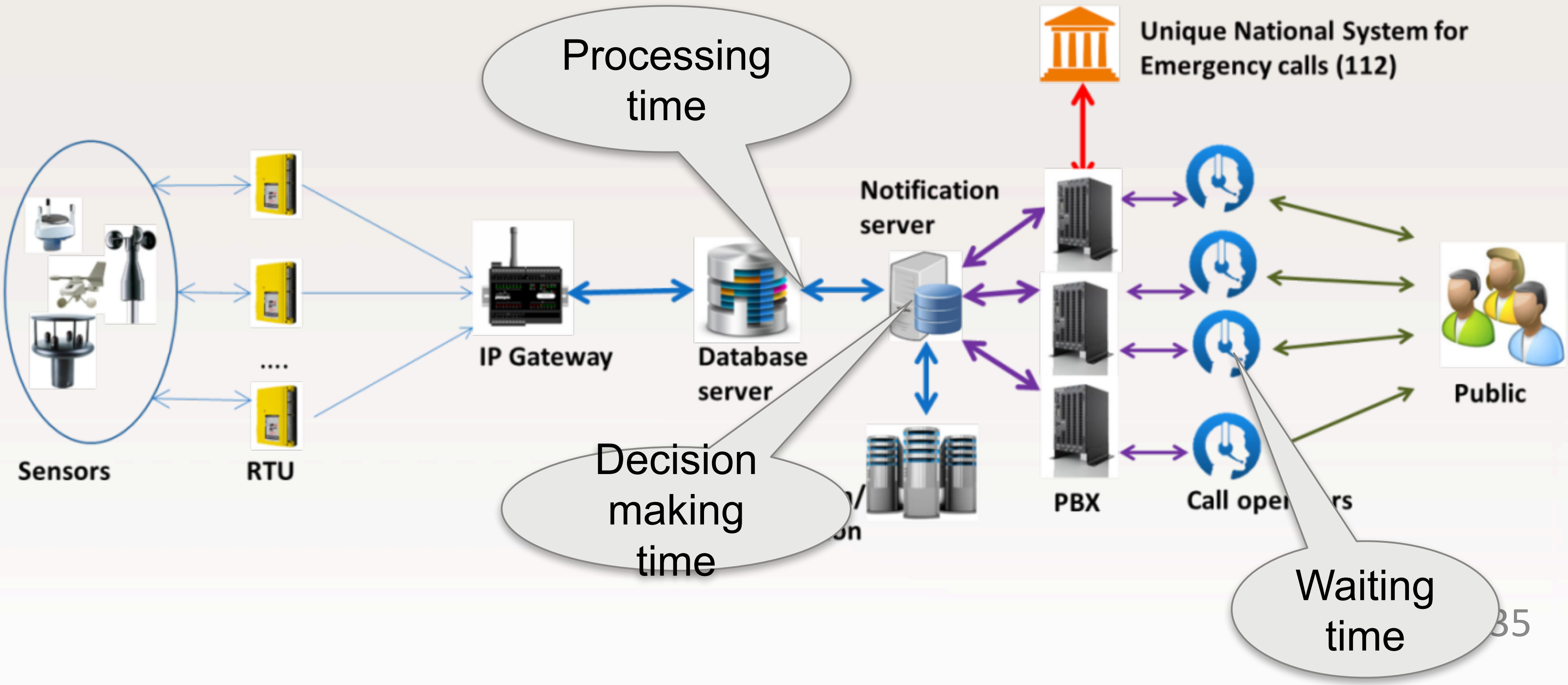
Zhao, Z., et al.,(2015), A software workbench for interactive, time critical and highly self-adaptive cloud applications (SWITCH), Cluster, Cloud and Grid Computing)

Data applications with critical time constraints

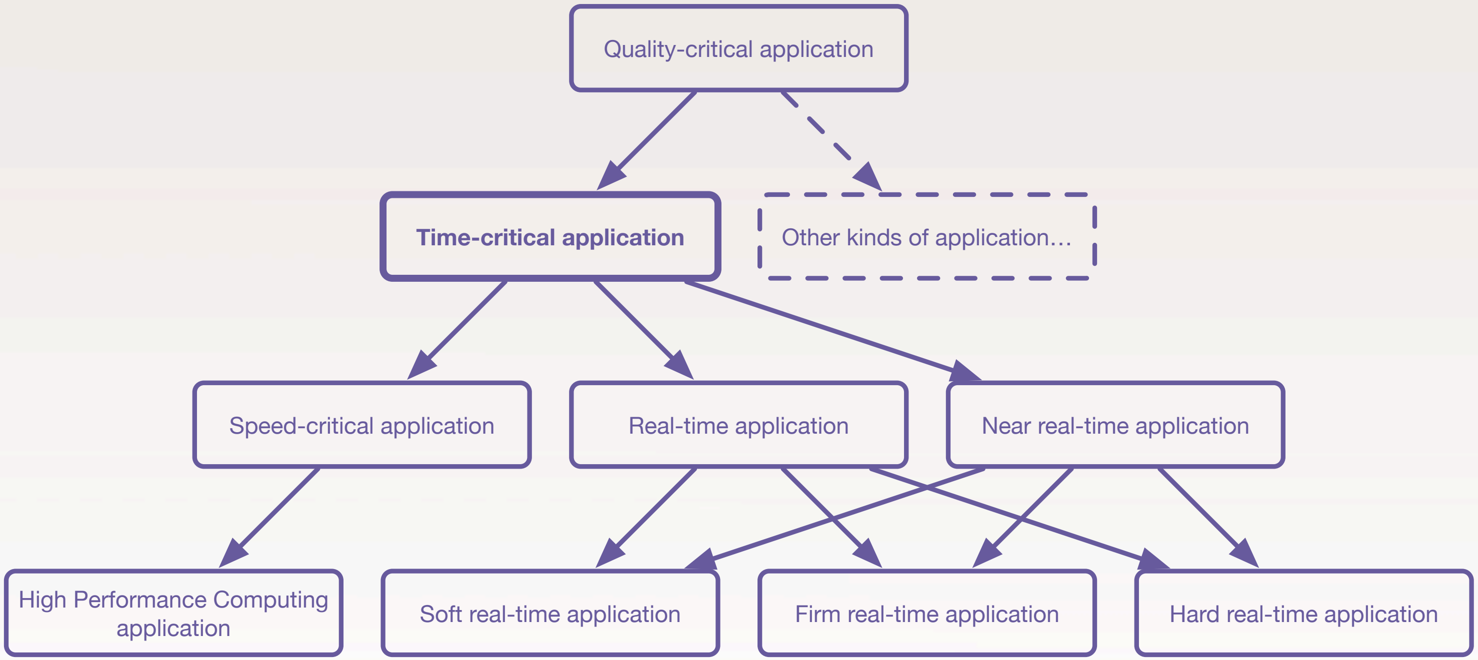


Disaster early warning system

Key time constraints



A simple taxonomy

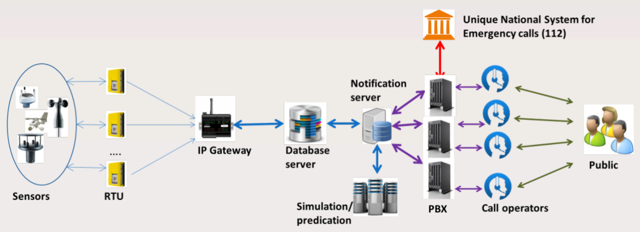


Z.Zhao et al., 2016, Time critical requirements and technical considerations for advanced support environments for data-intensive research, International workshop on IT4RIs, in the context of RTSS, Porto, Portugal. [https://doi.org/10.5281/zenodo.204756]

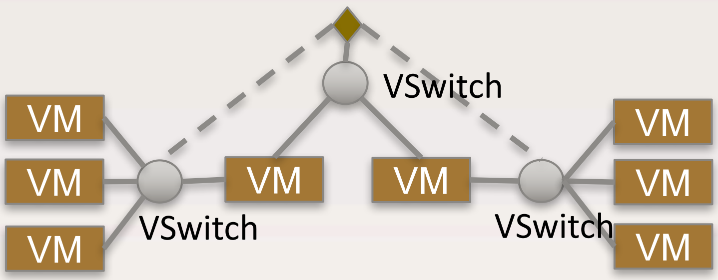
Why network aware application optimization

- Time critical applications are often distributed and require networked environments
- Network is often a bottleneck besides computation
- System level optimization is crucial to guarantee the time constraints
- **However, it requires vertical optimization from application to network:**
  - **not only programmable interfaces of network, but also effective programming and optimization models**

Virtual infrastructure planning



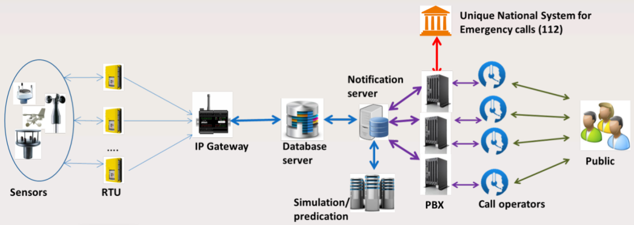
Time constraints, budget



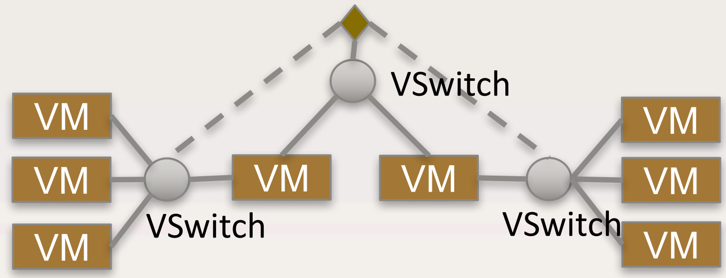
From application to VM

- **Problem**: how to select virtual machines based on quality of service, time constraints (deadlines), total budget and other constraints?
- **Related work**: scheduling (e.g., workflow in Grid [Yu 2008, Wang 2017])
  - Best effort scheduling
    - Immediate scheduling
    - List scheduling (batch, dependency, cluster and duplication)
  - QoS aware scheduling
    - Deadline aware
    - Budget aware
- **Approach**: extended from QoS aware (both deadline and budget) scheduling, but for multiple deadlines

From application logic to virtual infrastructure

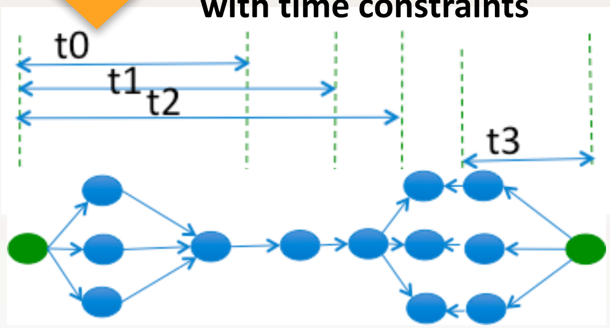


Time constraints, budget

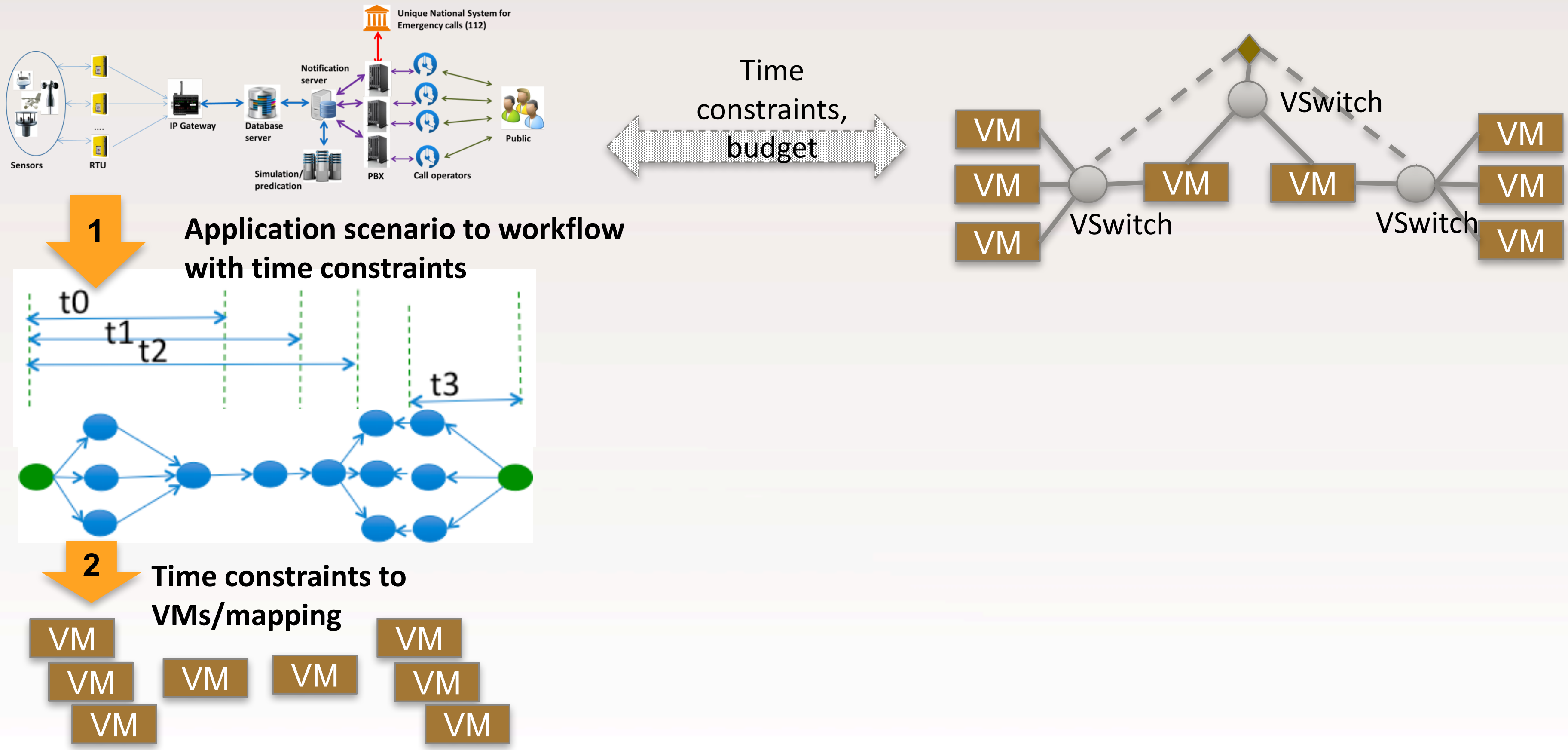


1

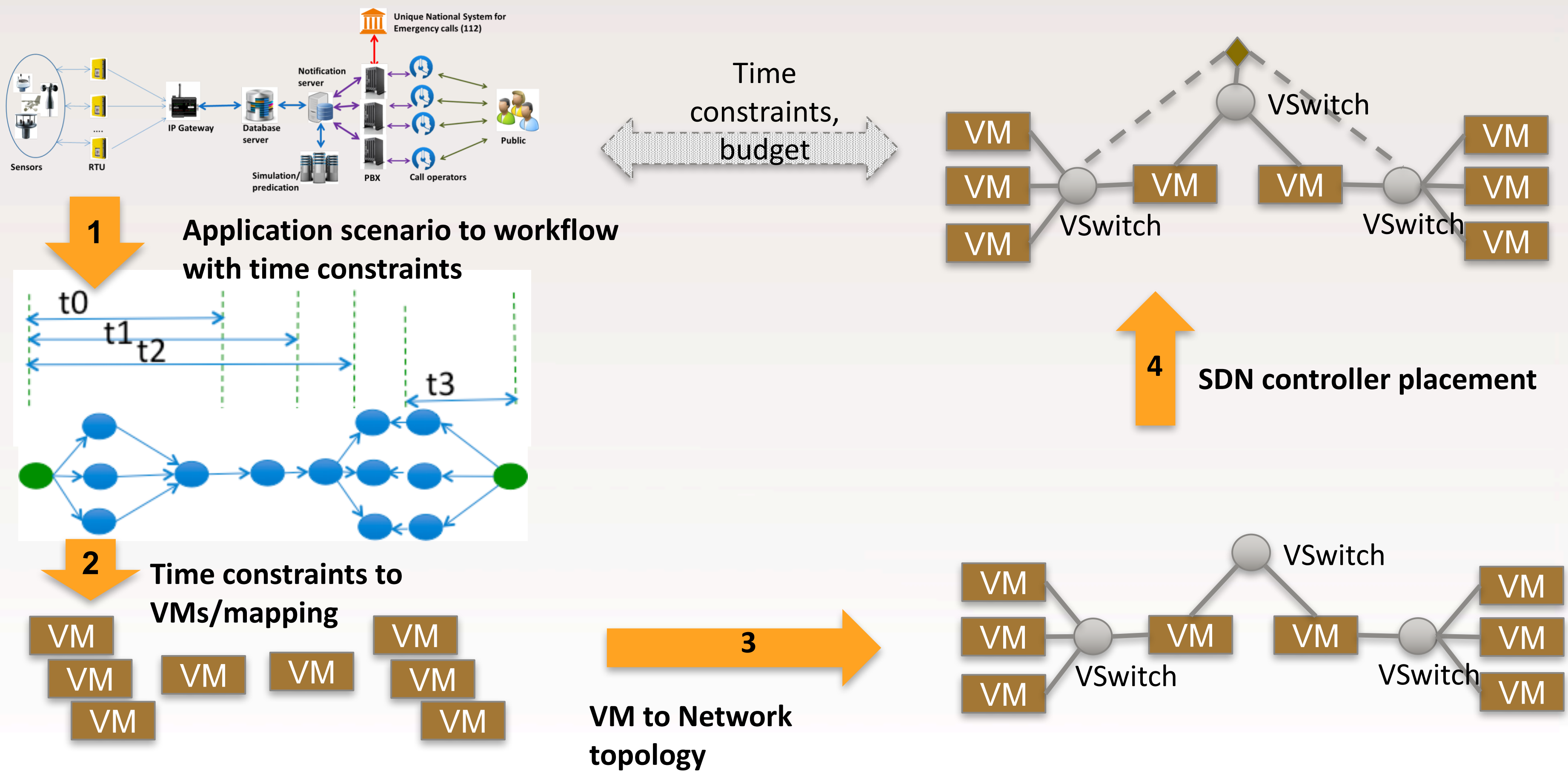
Application scenario to workflow with time constraints



From application logic to virtual infrastructure



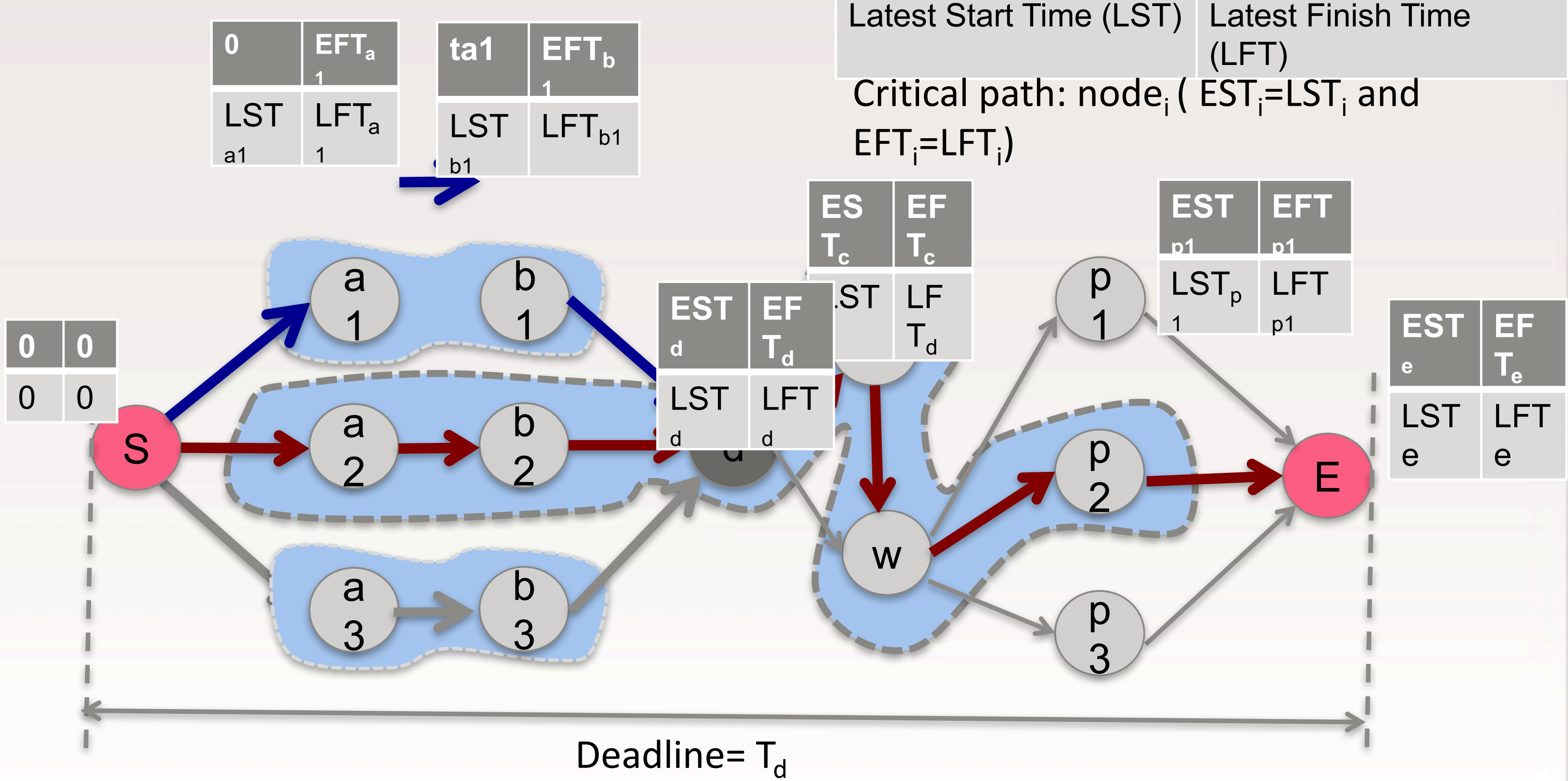
From application logic to virtual infrastructure



Critical path

Earliest Start Time (EST)	Earliest Finish Time (EFT)
Latest Start Time (LST)	Latest Finish Time (LFT)

Critical path: $node_i$ (EST_i=LST_i and EFT_i=LFT_i)



Results

- Wang, J., Taal, A., Martin, P., Hu Y., Zhou, H., Pang, J., de Laat, C., Zhao, Z. (*), (2017) *Planning Virtual Infrastructures for Time Critica Applications with Multiple Deadline Constraints* International journal of Future Generation Computer System [doi:10.1016/j.future.2017.02.001]
- Wang, J., de Laat, C., and Zhao, Z. (2017) *QoS Aware Virtual SDN Network Planning*, IFIP/IEE International Symposium on Integrated Networ Management, Lisbon, Portugal, May 8-12, 201

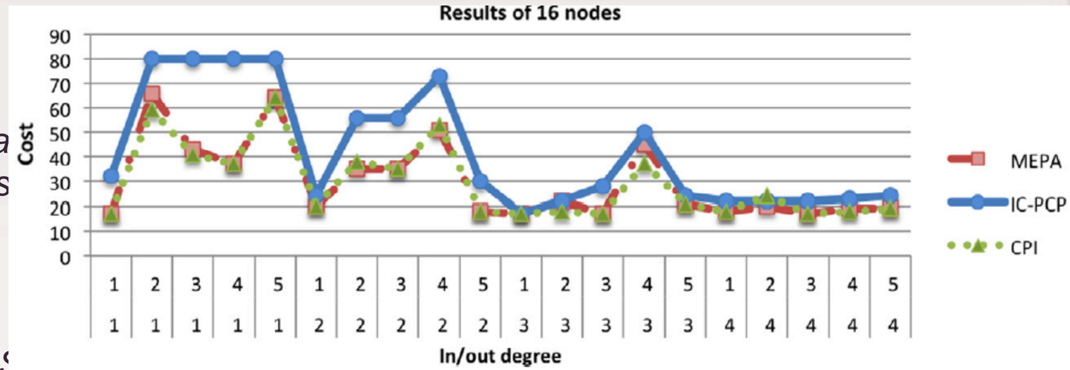
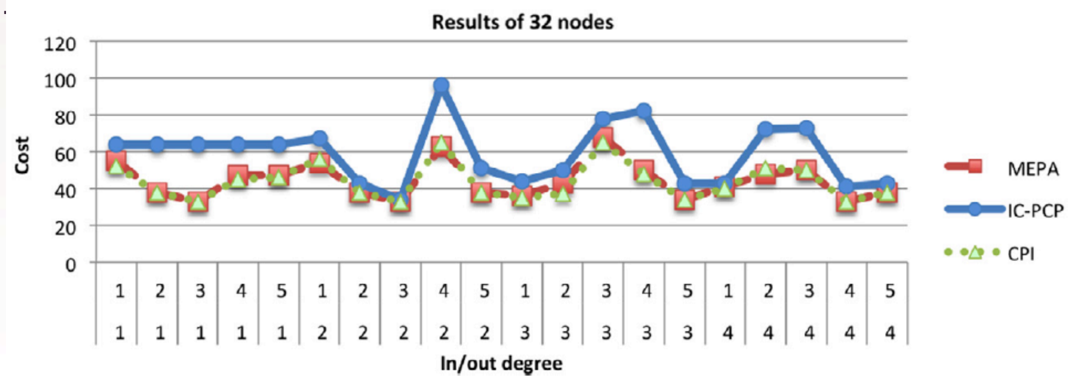
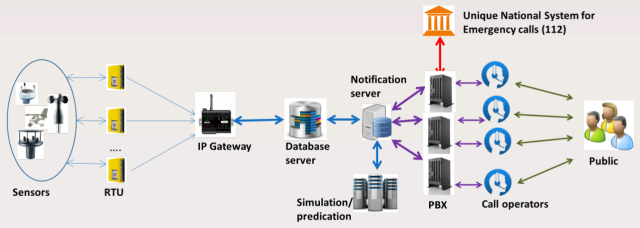
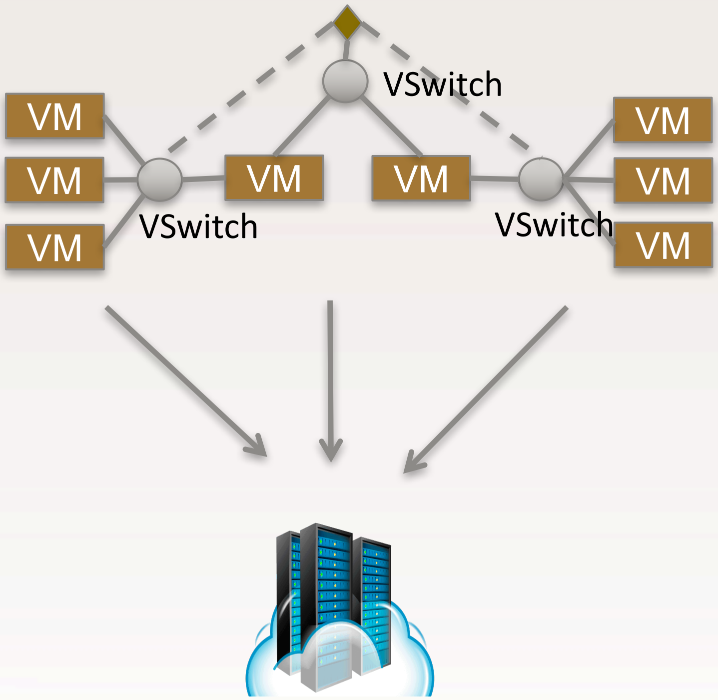


Fig. 3. Results of 16 nodes with IC-PCP, CPI and MEPA of single deadline.

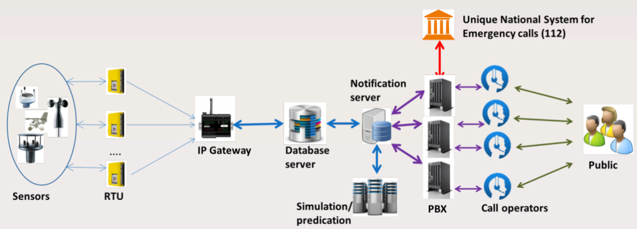


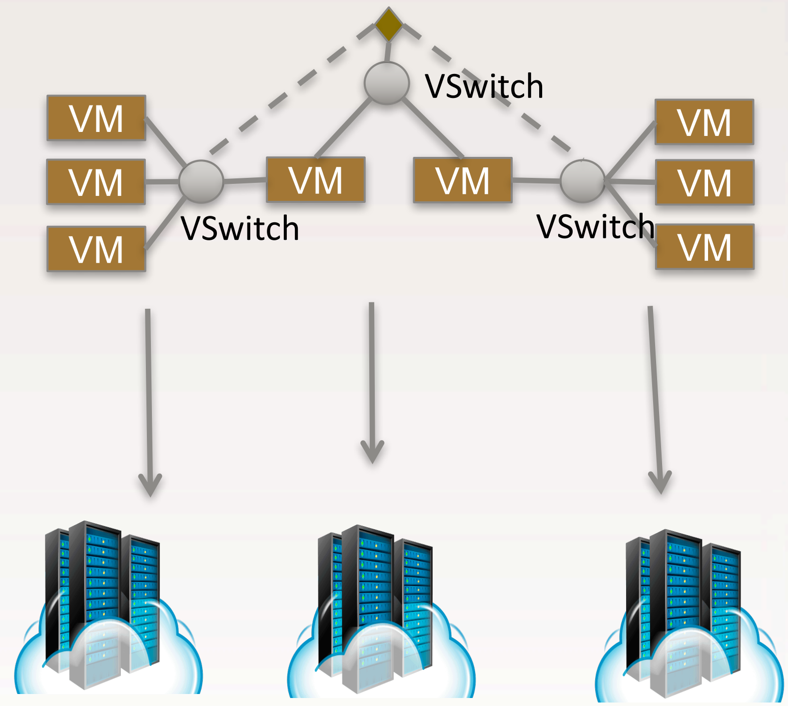
Provisioning virtual infrastructure



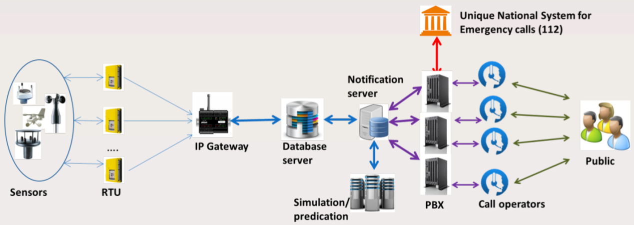


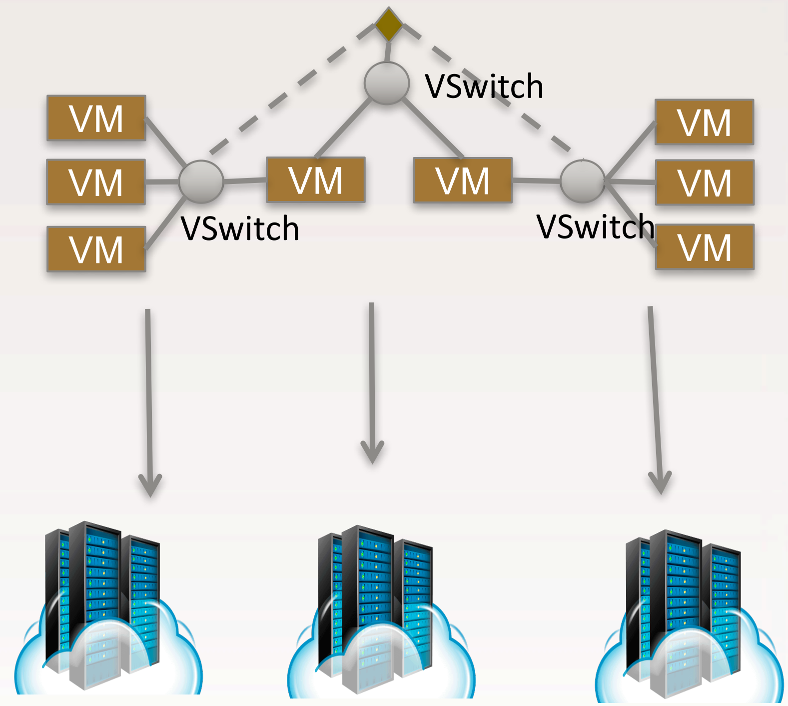
Why need multiple data centers?





Why need multiple data centers?





1) Geo-locations of sensors
2) Insufficient resources of a single data centre
3) Resilient purpose: some components may fail
4) SLA for the application
5) Etc.

Approaches

- Standardized Multi-level Infrastructure Description

```yaml
publicKeyPath: /Users/zh9314/.ssh/id_dsa.pub
userName: zh9314
topologies:
  - topology: zh_a
    cloudProvider: EC2
  - topology: zh_b
    cloudProvider: EC2
connections:
  - name: linkX
    source:
      component_name: zh_a.nodeB
      port: p3
      netmask: 255.255.255.0
      address: 192.168.10.10
    target:
      component_name: zh_b.nodeA
      port: p1
      netmask: 255.255.255.0
      address: 192.168.10.12
    bandwidth: 10000000
    latency: 200
```

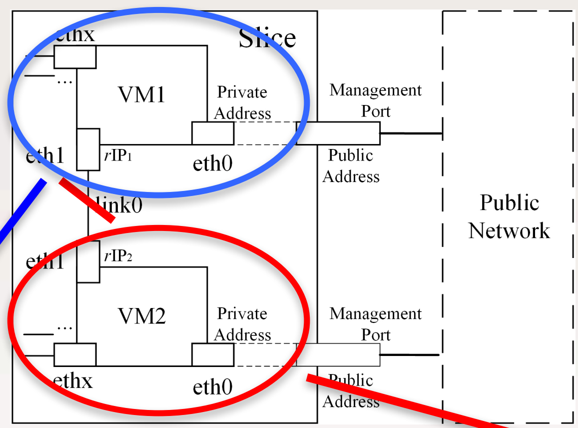
zh_all.yml

```yaml
components:
  - name: nodeA
    type: switch/compute
    nodetype: t2.medium
    OStype: "Ubuntu 14.04"
    domain: "ec2.us-east-1.amazonaws.com"
    script: /Users/zh9314/SWITCH_Provision/topology/t1/script/install.sh
    installation: /Users/zh9314/SWITCH_Provision/topology/t1/installation/Server
    public_address: TBD
    ethernet_port:
      - name: p2
        connection_name: c1.source
  - name: nodeB
    type: switch/compute
    nodetype: t2.medium
    OStype: "Ubuntu 14.04"
    domain: "ec2.us-east-1.amazonaws.com"
    script: /Users/zh9314/SWITCH_Provision/topology/t1/script/install.sh
    installation: /Users/zh9314/SWITCH_Provision/topology/t1/installation/Client
    public_address: TBD
    ethernet_port:
      - port_name: p2
        connection_name: c1.target
      - port_name: p3
        subnet_name: s1
        address: 192.168.10.10
subnets:
  - name: s1
    subnet: 192.168.10.0
    netmask: 255.255.255.0
connections:
  - name: c1
    source:
      component_name: nodeA
      port_name: p1
      netmask: 255.255.255.0
      address: 192.168.3.11
    target:
      component_name: nodeB
      port_name: p2
      netmask: 255.255.255.0
      address: 192.168.3.12
    bandwidth: 10000000
    latency: 200
```

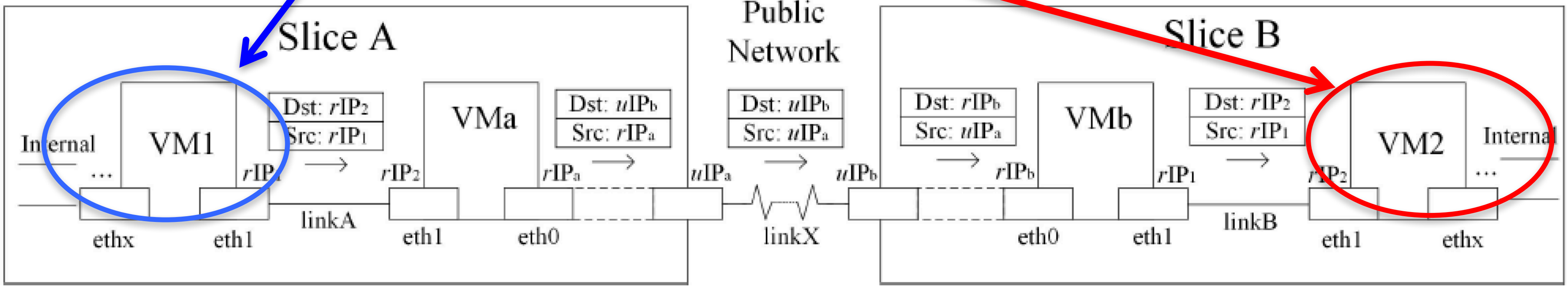
zh_a.yml

How it works

Original Topology

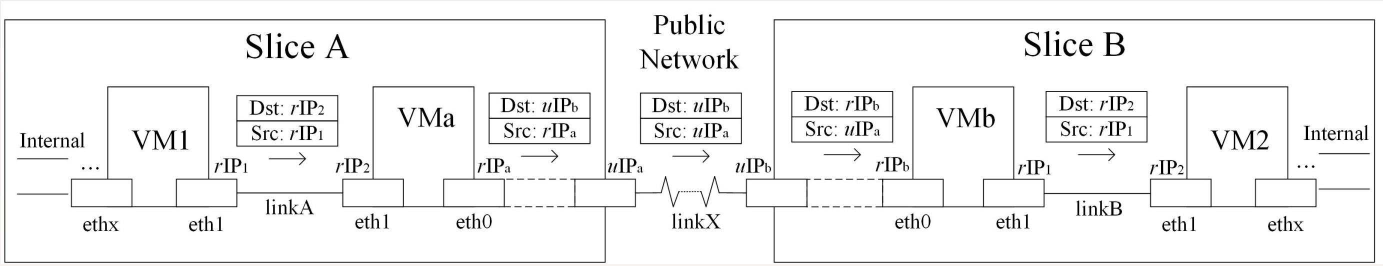


Partition

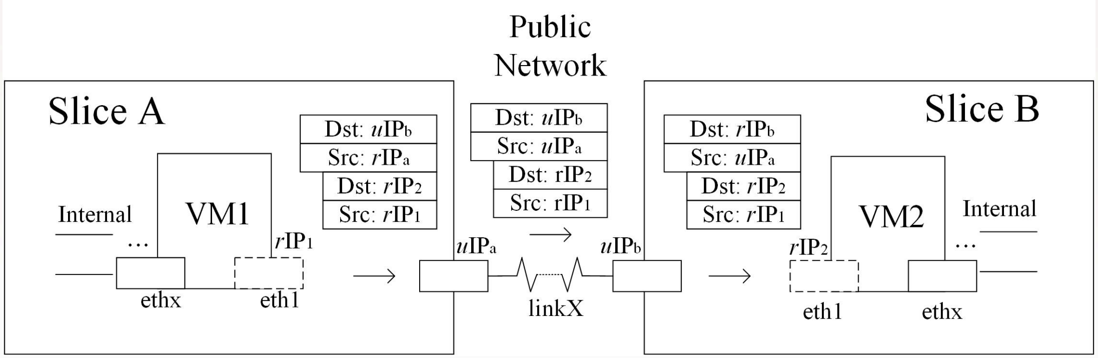


Approaches

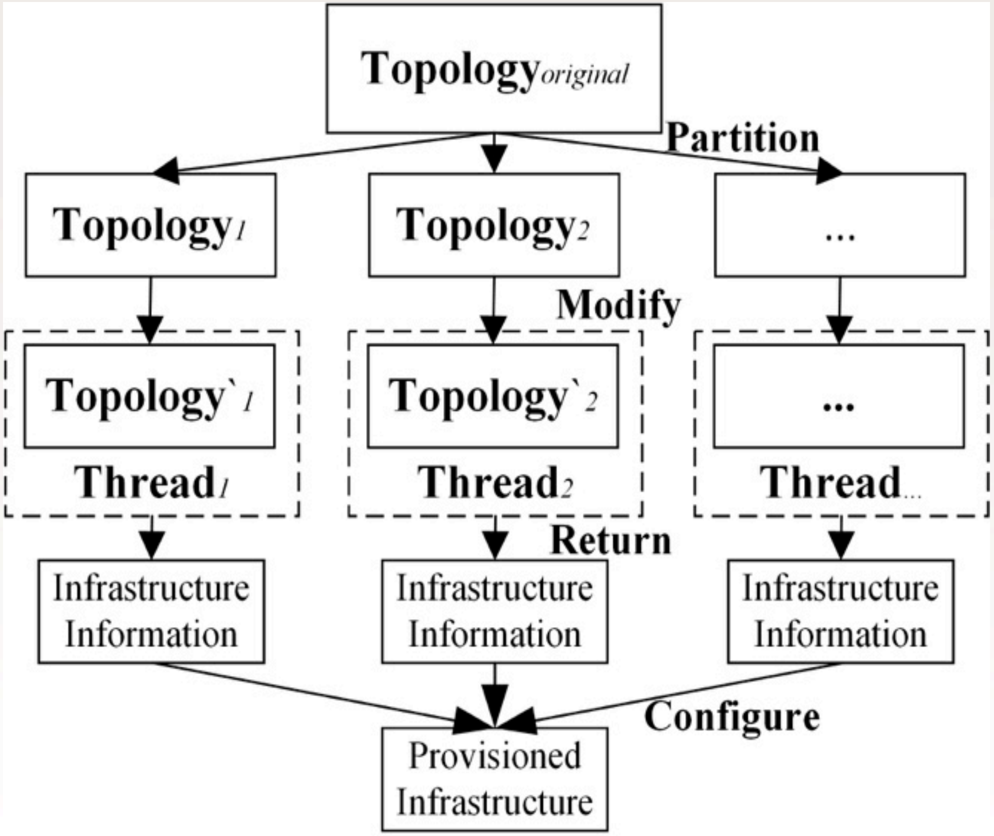
- Connection methods
  - NAT based



  - Tunnel based

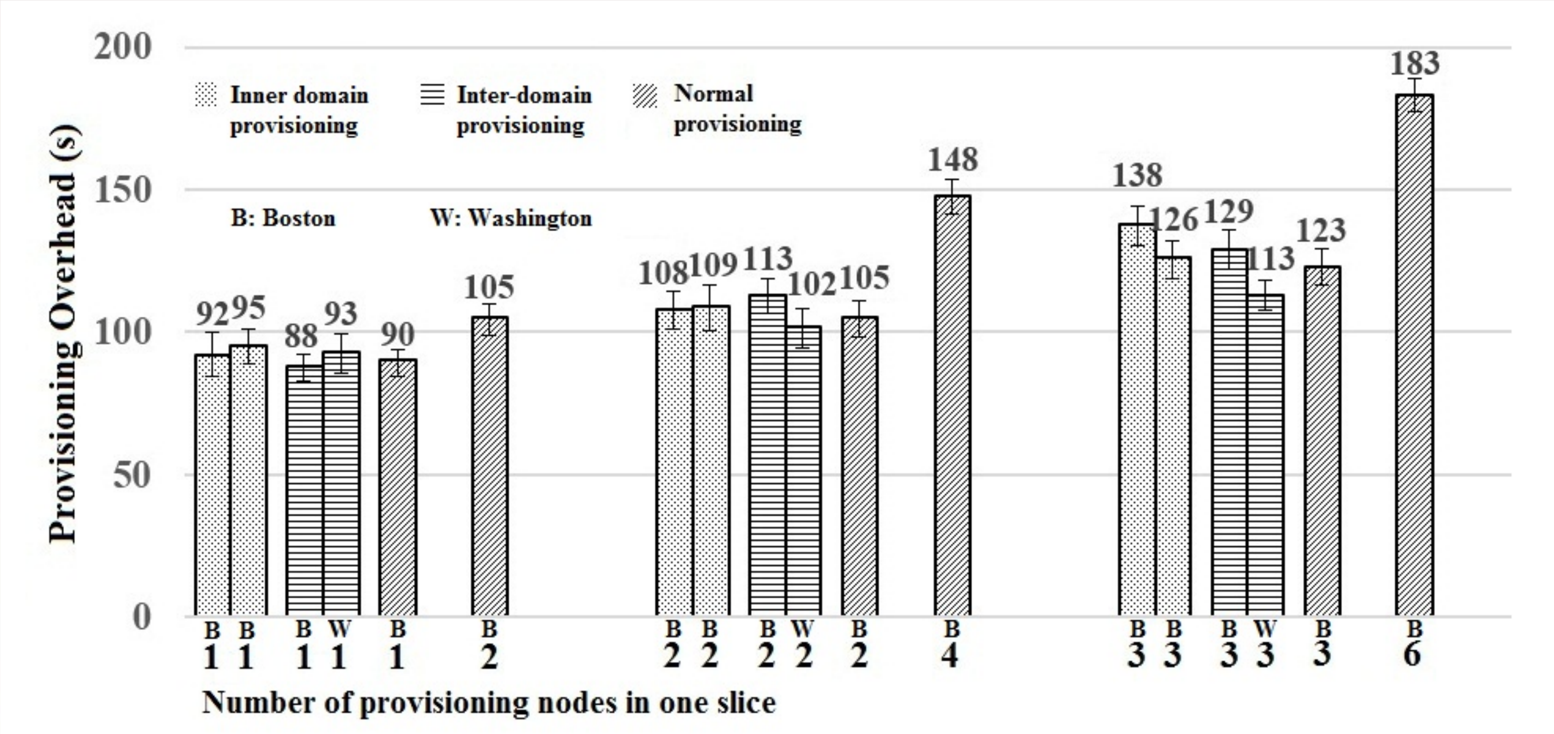


Basic idea



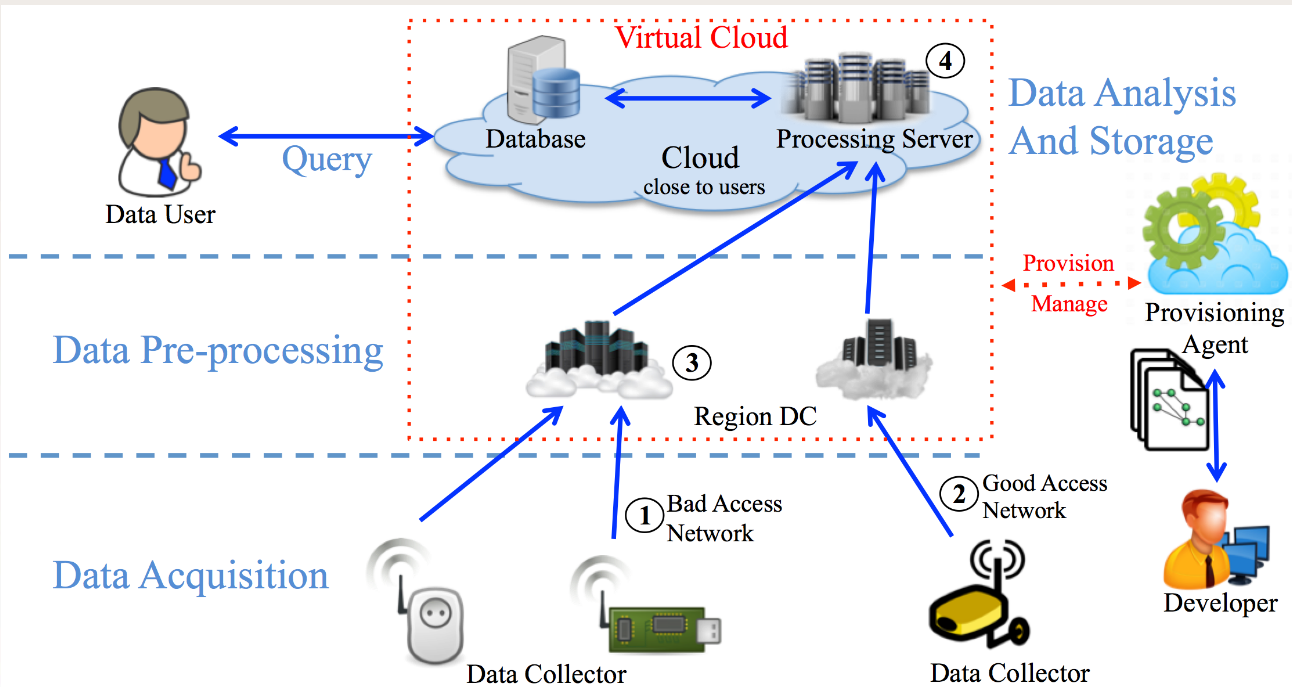
- Partition the original topology into sub-topologies.
- Provision them from different datacenters or Clouds.
- Make the network connection transparent to provision networked infrastructure.

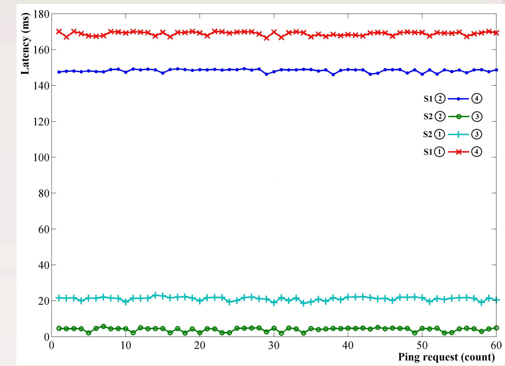
Evaluation Results

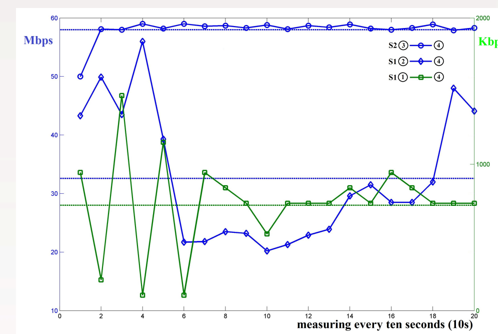


Zhou, H., Hu Y., Wang, J., Martin, P., Su, J., de Laat, C. and Zhao, Z., (2016) Fast Resource Co-provisioning for Time Critical Application Based on Networked Infrastructure, IEEE International Conference on CLOUD (CLOUD) 2016, San Francisco US.

QoS and geo-location aware provision

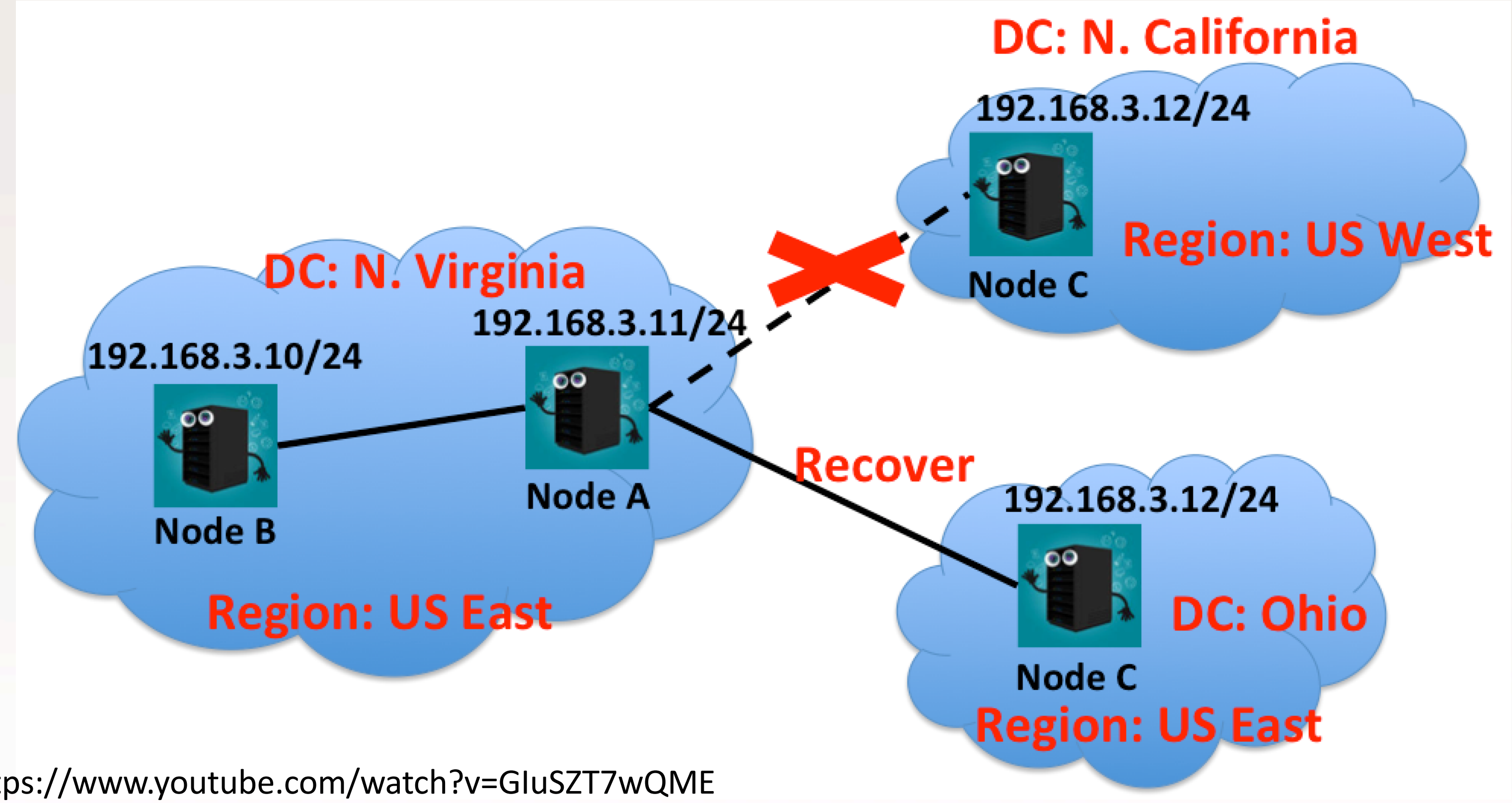






Zhou, H., Martin, P., Su, J., de Laat, C. and Zhao, Z. (2016) *A Flexible Inter-locale Virtual Cloud For Nearly Real-time Big Data Applications*, the 2nd International workshop on IT4RIs 16, in the context of IEEE Real-time System Symposium (RTSS)

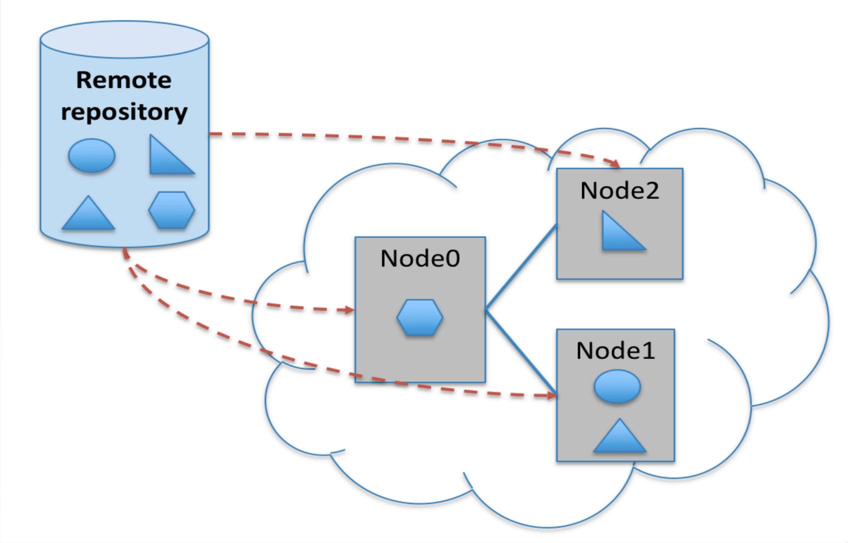
Failure recovery



https://www.youtube.com/watch?v=GIuSZT7wQME

Deploy components on virtual infrastructure

- Fetch components from remote repository
- Deploy the component



Deploy components on virtual infrastructure

- Fetch components from remote repository
- Deploy the component

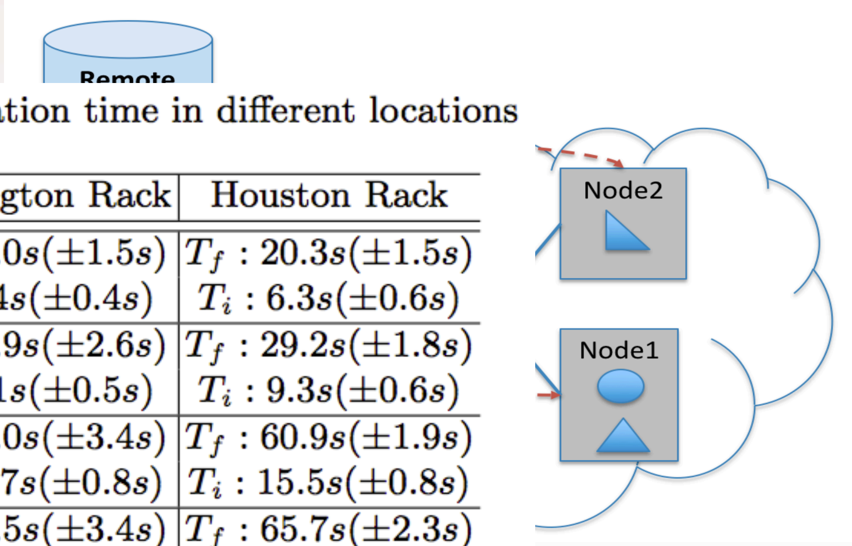
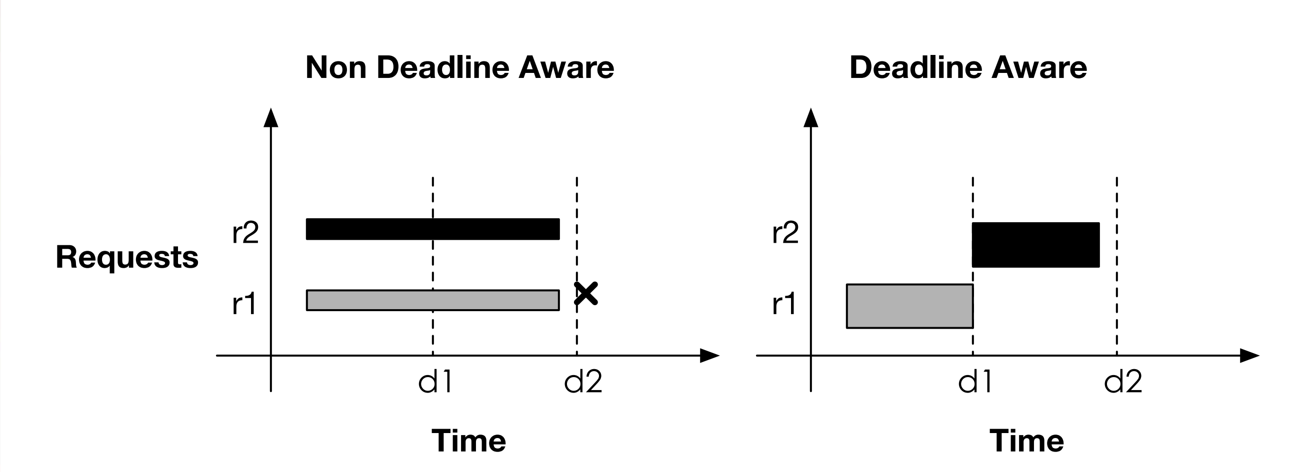


Table 1. Comparison of transmission time and installation time in different locations

Docker Image	Image Size	Boston Rack	Washington Rack	Houston Rack
ubuntu	400Mb	$T_f: 40.8s(\pm 2.2s)$ $T_i: 6.3s(\pm 0.5s)$	$T_f: 27.0s(\pm 1.5s)$ $T_i: 6.4s(\pm 0.4s)$	$T_f: 20.3s(\pm 1.5s)$ $T_i: 6.3s(\pm 0.6s)$
nginx	576Mb	$T_f: 58.7s(\pm 2.5s)$ $T_i: 9.3s(\pm 0.7s)$	$T_f: 38.9s(\pm 2.6s)$ $T_i: 9.1s(\pm 0.5s)$	$T_f: 29.2s(\pm 1.8s)$ $T_i: 9.3s(\pm 0.6s)$
mongodb	1200Mb	$T_f: 122.4s(\pm 3.0s)$ $T_i: 15.4s(\pm 0.5s)$	$T_f: 81.0s(\pm 3.4s)$ $T_i: 15.7s(\pm 0.8s)$	$T_f: 60.9s(\pm 1.9s)$ $T_i: 15.5s(\pm 0.8s)$
cassandra	1296Mb	$T_f: 132.2s(\pm 3.1s)$ $T_i: 17.1s(\pm 0.9s)$	$T_f: 87.5s(\pm 3.4s)$ $T_i: 17.3s(\pm 0.7s)$	$T_f: 65.7s(\pm 2.3s)$ $T_i: 17.4s(\pm 0.6s)$

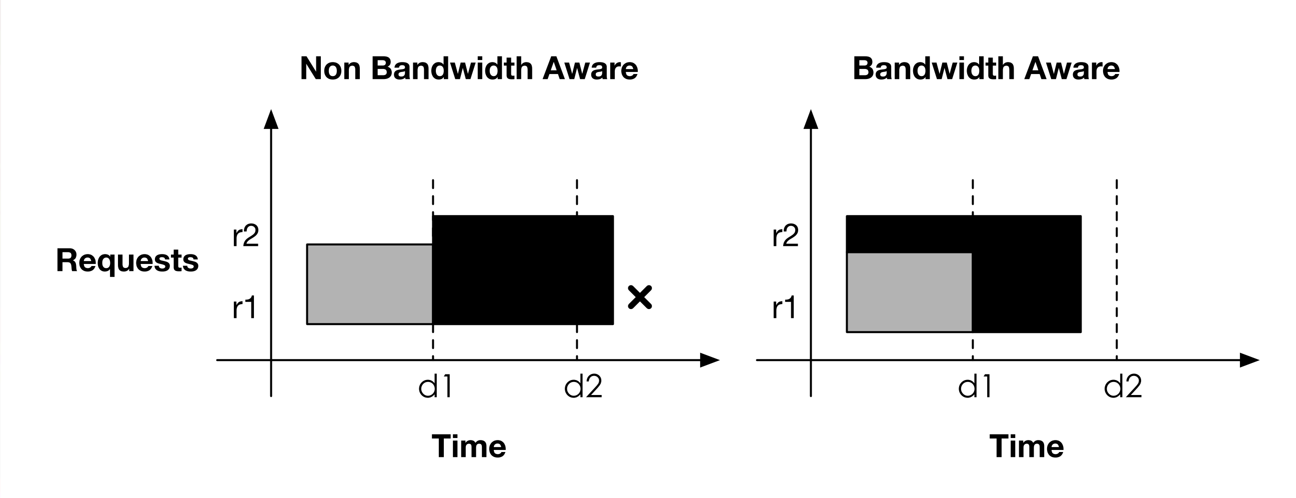
Bandwidth is the bottleneck

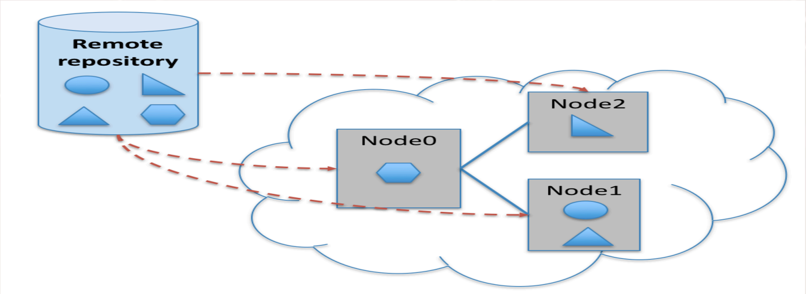
- How to meet the deployment deadlines?



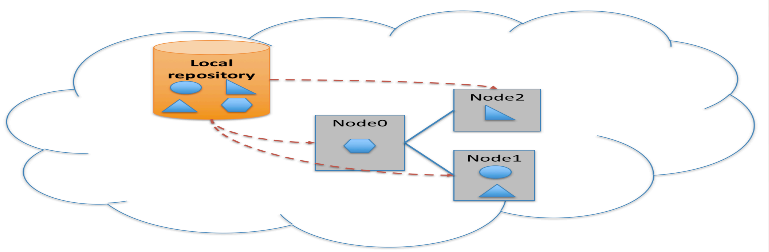
Bandwidth is the bottleneck

- Schedule network
- Optimize the utilization

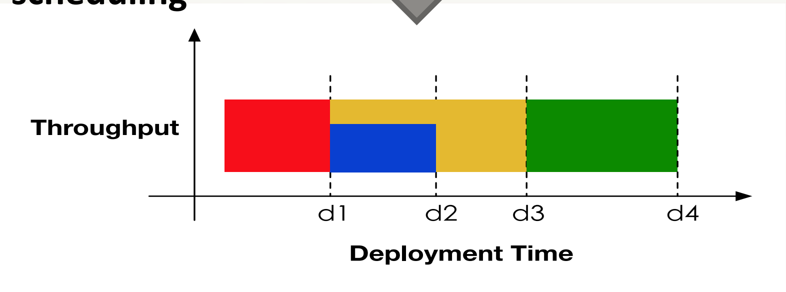




Create Local repository



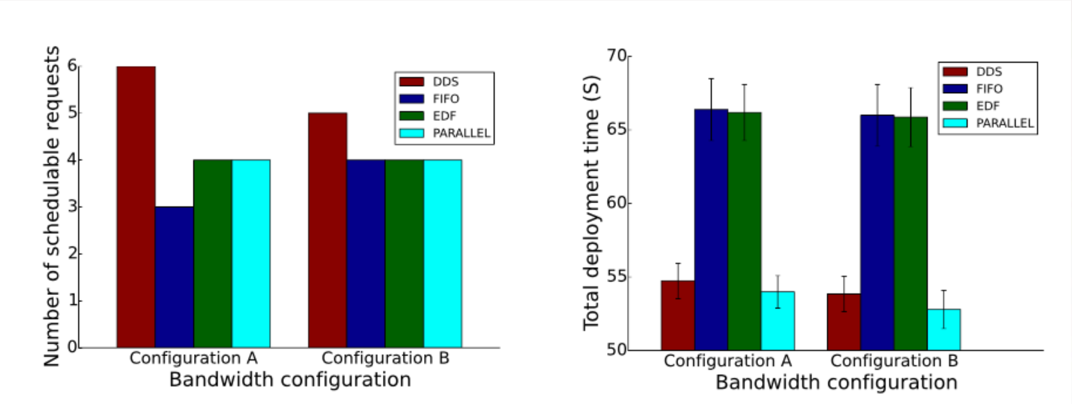
Bandwidth-aware EDF scheduling



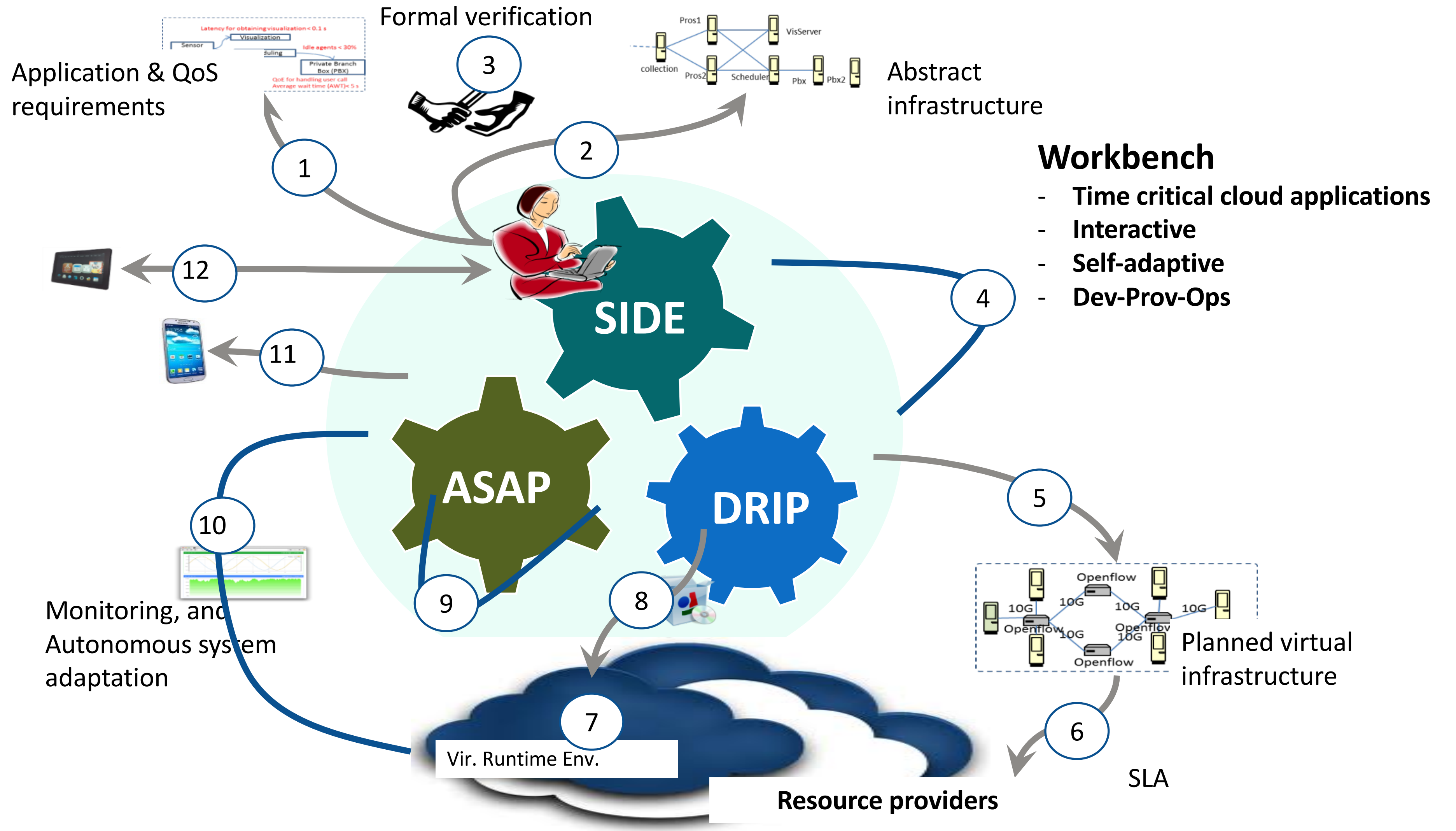
EDF + network scheduling for deployment

Scheduling options?

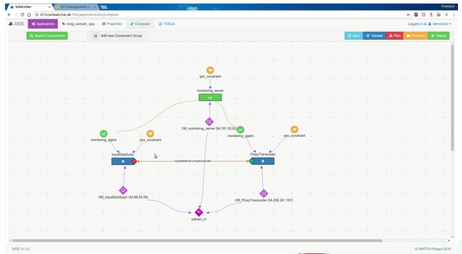
- **FIFO**: All the deployment requests are scheduled by the arrive time of the request in a sequential way.
- **PARALLEL**: All the deployment requests are scheduled immediately after arrival in a parallel way.
- **EDF**: All the deployment requests are scheduled by the EDF algorithm in a sequential way.
- **Bandwidth aware EDF**: including network bandwidth in the EDF scheduling



Hu, Y., Wang, J., Zhou, H., Martin, P., Taal, A., de Laat, C., and Zhao, Z.(2017) Deadline-aware Deployment for Time Critical Applications in Clouds has been accepted for presentation at the Euro-Par 2017 Conference in Santiago de Compostela

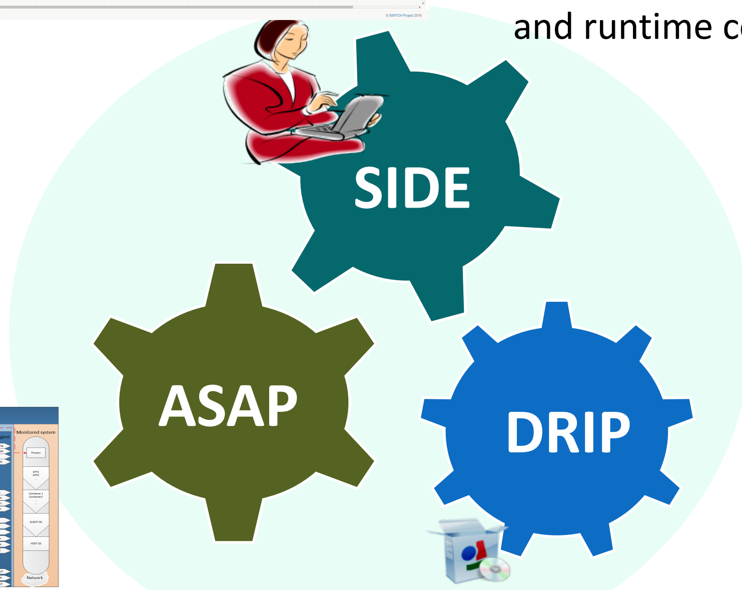


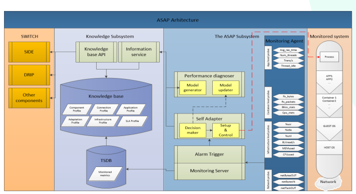
Formal verification
Application & QoS requirements
Abstract infrastructure
Workbench
- Time critical cloud applications
- Interactive
- Self-adaptive
- Dev-Prov-Ops
SIDE
ASAP
DRIP
1
2
3
4
5
6
7
8
9
10
11
12
Pros1
VisServer
collection
Pros2
Scheduler
Pbx
Pbx2
Openflow
10G
Planned virtual infrastructure
Monitoring, and Autonomous system adaptation
Vir. Runtime Env.
Resource providers
SLA

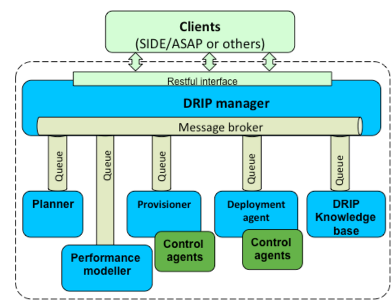


Multi view GUI

- Program application logic + virtual infrastructure
- Integration with DRIP/ASAP for provisioning and runtime control







Outline

- Infrastructures for data centric research
- Infrastructure interoperability
- Time critical cloud applications
- **Summary**

Summary

- The interoperability among big data infrastructures is crucial for system level of sciences
- It is very challenging to guarantee the deadlines of the time critical cloud applications
- There are increasing number of time critical applications identified from industrial and scientific computing domains
- Meta-heuristic approaches are playing important role in optimizing resource allocation and task scheduling for time critical applications; Dynamic Real-time Infrastructure Planner (DRIP) can automate the planning, provision and deployment of time critical cloud applications
- From software engineering point of view, time critical application oriented application-infrastructure programming and control models are needed

Acknowledgement and References

- **EU H2020 SWITCH** www.switchproject.eu
- **EU H2020 ENVRI^{PLUS}** www.envriplus.eu
- **EU H2020 VRE4EIC** www.vre4eic.eu
- Wang, J., Taal, A., Martin, P., Hu Y., Zhou, H., Pang, J., de Laat, C., Zhao, Z. (*), (2017) Planning Virtual Infrastructures for Time Critical Applications with Multiple Deadline Constraints, International journal of Future Generation Computer System
- Koulouzis, S., Belloum, A., Bubak, M., Zhao, Z., Zivkovic, M., de Laat, C. (2015) SDN-Aware Federation of Distributed Data, International journal of Future Generation Computer System.
- Zhu, H., van der Veldt, K., Zhao, Z.(*) , Grosso, P., Pavlov, D., Soeurt, J., Liao, X., de Laat, C., (2015) A semantic enhanced Power Budget Calculator for distributed computing using IEEE 802.3az, International Journal of Cluster Computing Volume 18, Issue 1, Page 61-77
- Hu, Y., Wang, J., Zhou, H., Martin, P., Taal, A., de Laat, C., and Zhao, Z.(2017) Deadline-aware Deployment for Time Critical Applications in Clouds has been accepted for presentation at the Euro-Par 2017 Conference in Santiago de Compostela
- Wang, J., de Laat, C., and Zhao, Z. (2017) QoS-Aware Virtual SDN Network Planning, IFIP/IEEE International Symposium on Integrated Network Management, Lisbon, Portugal, May 8-12, 2017.
- Zhou, H., Hu Y., Wang, J., Martin, P., Su, J., de Laat, C. and Zhao, Z., (2016) Fast Resource Co-provisioning for Time Critical Application Based on Networked Infrastructure, IEEE International Conference on CLOUD (CLOUD) 2016, San Francisco US.
- Zhou, H., Hu Y., Wang, J., Martin, P., de Laat, C. and Zhao, Z., (2016) Fast and Dynamic Resource Provisioning for Quality Critical Cloud Applications, IEEE International Symposium On Real-time Computing (ISORC) 2016, York UK

H2020 Project

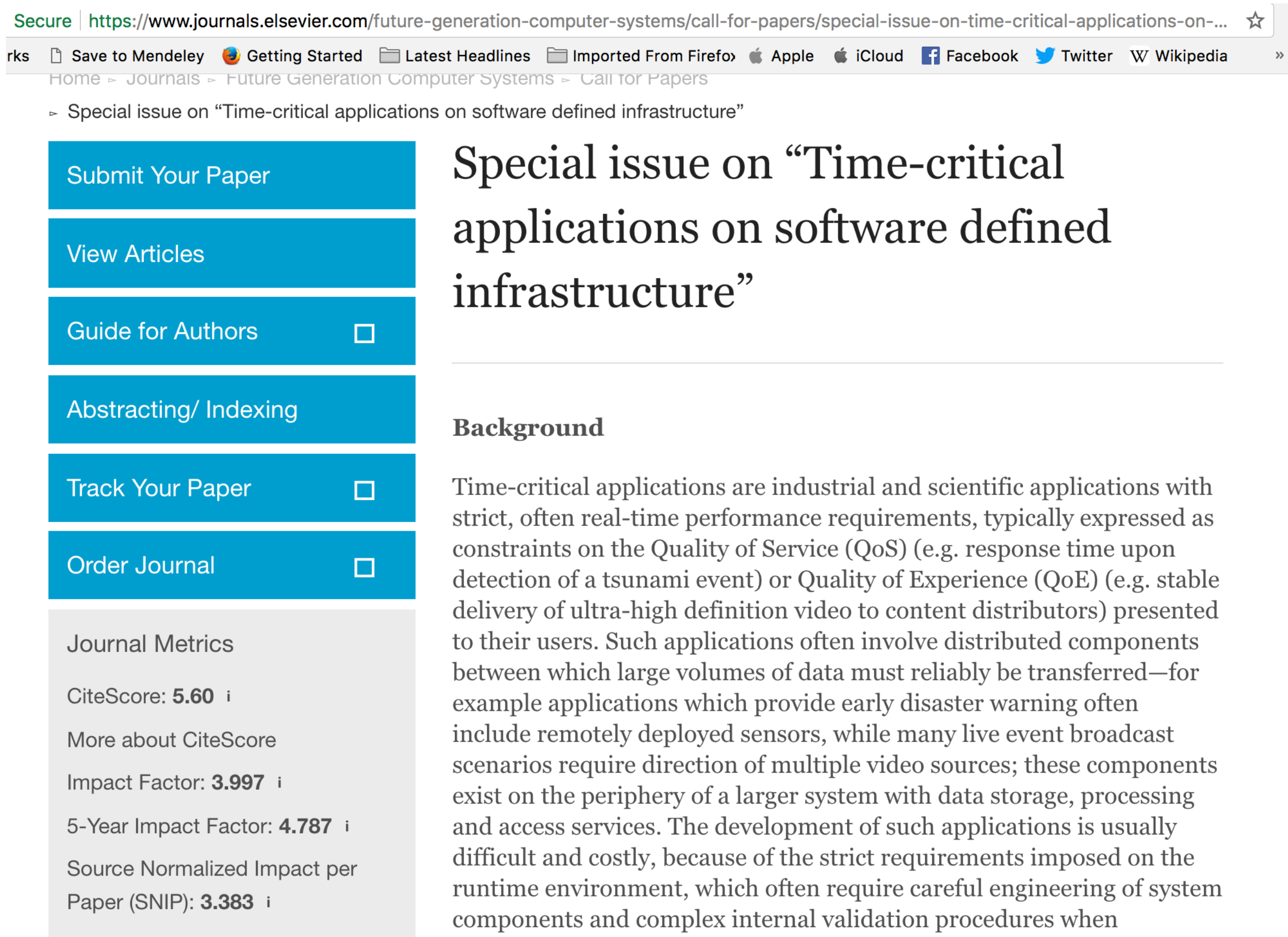
Environmental Research Infrastructures Providing Shared Solutions for Science and Society

Special issue on time critical applications on software defined infrastructure

Important dates:

- Submission April 10, 2018
- Notification September 30, 2018

Contact: z.zhao@uva.nl


Secure | https://www.journals.elsevier.com/future-generation-computer-systems/call-for-papers/special-issue-on-time-critical-applications-on-...

rks | Save to Mendeley | Getting Started | Latest Headlines | Imported From Firefox | Apple | iCloud | Facebook | Twitter | Wikipedia

Home ▹ Journals ▹ Future Generation Computer Systems ▹ Call for Papers

▹ Special issue on “Time-critical applications on software defined infrastructure”

Submit Your Paper

View Articles

Guide for Authors

Abstracting/ Indexing

Track Your Paper

Order Journal

Journal Metrics

CiteScore: **5.60**

More about CiteScore

Impact Factor: **3.997**

5-Year Impact Factor: **4.787**

Source Normalized Impact per Paper (SNIP): **3.383**

SCImago Journal Rank (SJR):

Special issue on “Time-critical applications on software defined infrastructure”

Background

Time-critical applications are industrial and scientific applications with strict, often real-time performance requirements, typically expressed as constraints on the Quality of Service (QoS) (e.g. response time upon detection of a tsunami event) or Quality of Experience (QoE) (e.g. stable delivery of ultra-high definition video to content distributors) presented to their users. Such applications often involve distributed components between which large volumes of data must reliably be transferred—for example applications which provide early disaster warning often include remotely deployed sensors, while many live event broadcast scenarios require direction of multiple video sources; these components exist on the periphery of a larger system with data storage, processing and access services. The development of such applications is usually difficult and costly, because of the strict requirements imposed on the runtime environment, which often require careful engineering of system components and complex internal validation procedures when integrating these components into a single functioning system