



Grant Agreement No. 619572

COSIGN

Combining Optics and SDN In next Generation data centre Networks

Programme: Information and Communication Technologies

Funding scheme: Collaborative Project – Large-Scale Integrating Project

Deliverable D5.3 – Plan for Integration and Testing of COSIGN Demonstrators

Due date of deliverable: 15/6/2016

Actual submission date: 15/6/2016

Start date of project: January 1, 2014

Duration: 36 months

Lead contractor for this deliverable: Technical University of Denmark

Project co-funded by the European Commission within the Seventh Framework Programme		
Dissemination Level		
PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Executive Summary

The demonstrator plan for the remainder of the COSIGN project is described in this document. The plan assumes the approval of a 3-month extension of the project period. This additional time has not resulted in a significant shift in the original demonstrator plan but is rather used as an opportunity for additional exploitation and dissemination of the final demonstrator. Essentially, we will have time to harvest more results and do more comprehensive tests on the demonstrators. The added time will also increase the likelihood of including the Venture switch in the long-term demonstrator.

The demonstrator plan is described with a granularity of specific tasks for each demonstrator. In summary, three main demonstrators are planned; the ECOC demonstrator, the Industrial demonstrator at Interoute and the final demonstrator co-hosted by University of Bristol and the Technical University of Denmark. The demonstrators combined will achieve both broad dissemination of COSIGN results and detailed scientific analysis of features and performance. We are confident that with this plan we have mapped out a realistic path to successful realization of the demonstrators.

Legal Notice

The information in this document is subject to change without notice.

The Members of the COSIGN Consortium make no warranty of any kind with regard to this document, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The Members of the COSIGN Consortium shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Possible inaccuracies of information are under the responsibility of the project. This report reflects solely the views of its authors. The European Commission is not liable for any use that may be made of the information contained therein.

Document Information

Status and Version:	V1.1 Final version	
Date of Issue:	15/06/2016	
Dissemination level:	Public	
Author(s):	Name	Partner
	Yaniv Ben-Itzhak	IBM
	Michael Galili	DTU
	Salvatore Spadaro	UPC
	Albert Pagès	UPC
	Fernando Agraz	UPC
	Chris Jackson	UNIVBRIS
	Domenico Gallico	IRT
	Alessandro Predieri	IRT
Edited by:	Michael Galili	DTU
Reviewed by:	Chris Jackson	UNIVBRIS
	Adam Hughes	POLATIS
Checked by :	Sarah Ruepp	DTU

Table of Contents

Executive Summary	2
Table of Contents	4
1 Introduction.....	5
1.1 Reference Material	5
1.1.1 Reference Documents	5
1.1.2 Acronyms and Abbreviations	5
1.2 Document History	6
2 ECOC demonstrator.....	7
2.1 The data plane.....	7
2.2 The vApp use case.....	8
2.3 The VDC use case	9
2.4 Gantt chart for preparation of the ECOC demonstrator.....	10
3 Industrial Demonstrator	11
3.1 Plan for the industrial demonstrator	12
3.1.1 Test plan	12
4 Final Demonstrator.....	13
4.1 The mid-term data plane.....	13
4.2 The long-term data plane.....	14
4.2.1 Actions Required for Completion.....	15
4.3 The vApp use case.....	15
4.3.1 KPIs and indicative base values.....	16
4.4 The VDC use case	17
4.4.1 KPIs and indicative base values.....	18
4.5 The VM migration use case.....	18
4.5.1 KPIs and indicative base values.....	18
4.5.2 Experimental conditions that could be altered to obtain more results include:	19
4.6 Timeline for preparation of the large scale demonstrator.....	20
5 Novelty aspect of the VDC use case.....	21
6 Summary of KPIs.....	22

1 Introduction

This document reports the planning process for building the COSIGN project demonstrators. Tasks are listed with the assigned partner and estimates. The descriptions of most tasks are presented in D5.0 [2], but where necessary supplementary descriptions of demonstrator efforts are included.

The document also holds quantified base values of the KPIs against which the performance of the demonstrators will be compared.

At the end of the document there is a clarification of the novelty of the VDC use case.

1.1 Reference Material

1.1.1 Reference Documents

[1]	COSIGN FP7 Collaborative Project Grant Agreement Annex I - "Description of Work"
[2]	D5.0 "Definition, Design and Test Plan for Use Cases"
[3]	http://hadoop.apache.org/
[4]	http://spark.apache.org/
[5]	http://prof.ict.ac.cn/BigDataBench/
[6]	https://github.com/intel-hadoop/HiBench
[7]	Soudeh Ghorbani, Cole Schlesinger, Matthew Monaco, Eric Keller, Matthew Caesar, Jennifer Rexford, and David Walker. 2014. Transparent, Live Migration of a Software-Defined Network. In Proceedings of the ACM Symposium on Cloud Computing (SOCC '14). ACM, New York, NY, USA, , Article 3 , 14 pages. DOI= http://dx.doi.org/10.1145/2670979.2670982
[8]	D. Williams, H. Jamjoom, Z. Jiang, and H. Weatherspoon. VirtualWires for Live Migrating Virtual Networks across Clouds. Technical Report RC25378, IBM, 2013
[9]	K. Ye, X. Jiang, R. Ma and F. Yan, "VC-Migration: Live Migration of Virtual Clusters in the Cloud," 2012 ACM/IEEE 13th International Conference on Grid Computing, Beijing, 2012, pp. 209-218

1.1.2 Acronyms and Abbreviations

The most frequently used acronyms in this document are listed below. Additional acronyms may be defined and used throughout the text.

DoW	Description of Work
SDN	Software Defined Networks
VDC	Virtual Data Centre
TOR	Top of Rack
ODL	OpenDaylight
SDN	Software Defined Network
PoP	Point of Presence
TDM	Time Domain Multiplexing
RTT	Round Trip Time
ECOC	European Conference on Optical Communication
vApp	Virtual Application

1.2 Document History

Version	Date	Authors	Comment
0.1	18/05/2016	See the list of authors	TOC
0.2	04/06/2016		Partner contributions
0.3	08/06/2016		First integrated version
0.4	14/06/2016		Second integrated version with reviewer comments and additional partner contributions
1.0	14/06/2016		Integrated version for quality check
1.1	15/06/2016		Final version

2 ECOC demonstrator

This part of the plan concerns a COSIGN demonstrator for exhibition at the European Conference on Optical Communication (ECOC) in Düsseldorf during September 19th - 21st 2016. The demonstrator is described in detail in D5.0 [2]. A schematic of the demonstrator architecture is shown in Figure 1 for reference. The following sections list the major identified tasks remaining to complete the demonstrator. Immediately after ECOC the demonstrator hardware will be transported to IRT to be used in the Industrial Demonstrator discussed in section 3.

2.1 The data plane

The data plane of the ECOC demonstrator is illustrated below.

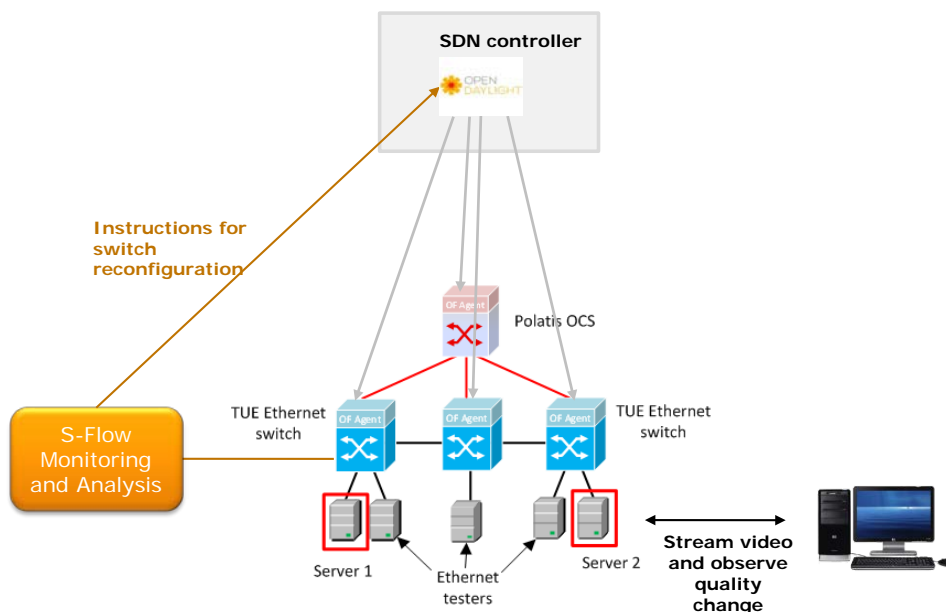


Figure 1 – schematic of ECOC demonstrator

The physical demonstrator will be built into two ‘half size’ 19 inch racks at DTU from hardware delivered by several partners. The racks will be transported to Düsseldorf fully assembled.

Currently, three servers for hosting the demonstrator (one hosting the controller and two others hosting the demonstrator applications) have been made available as part of the integration test bed hosted by DTU. The current topology allows testing using connections through either the Polatis switch or the TUE TOR (but not both) due to optical incompatibility of the two switches. The data plane shown in Figure 1 will be built as soon as additional TOR switches and optical converters to resolve the switch incompatibility arrive. The key tasks remaining to complete the demonstrator data plane are:

- Ethernet tester for generating ‘background’ traffic for the demonstrator.
 - The tester is available and will be programmed for use in the demonstrator
 - **Ongoing**, DTU, till mid-July.
- Inclusion of remaining TOR switches in data plane
 - A total of 3 TUE TORs are required. 1 is available, 2 more will be made available.
 - **Ongoing**, TUE, till end June.
- Optical interfacing of Polatis switch and TOR switch

Combining Optics and SDN In next Generation data centre Networks

- Single-mode/multi-mode converters are required in order establish optical connections that pass through both the Polatis and TOR switches.
 - **Ongoing**, TUE, till end July.
- Assembly of the final demonstrator.
 - Initial testbed built to allow server installation and integration testing of components. Will progress towards final demonstrator as components become available.
 - **Ongoing but awaiting delivery of remaining TOR switches and optical single-mode/multi-mode converters**, DTU, till start August.

2.2 The vApp use case

The vApp use case requires:

- Integration with TUE switch sFlow agent
 - **Ongoing**, PhotonX, end of June.
- Integration with ODL – capability to set up optical circuits (Polatis) and modifying forwarding rules (TOR)
ODL has two options:
 - Integrate the physical observer directly to current version of ODL, which includes optical circuit provisioning.
 - Extended ODL to support two types of optical circuits *private* (supported by current version) and *shared* – for shared optical circuits, we need to examine whether current APIs can support it or if it must be modified to configure the TOR accordingly.
 - **Ongoing**, IBM, NXW, End of June - ODL integration with the Path Computation Manager. Mid July - Integration with the physical observer.
- Integration of the physical and virtual observers to Horizon
 - **Ongoing**, IBM, i2cat, DTU, till mid of July.
- Integration of Horizon with Heat
 - **Not started**, IBM, I2cat, till end of July.

2.3 The VDC use case

The data plane architecture for the demonstration of the VDC use case is depicted below.

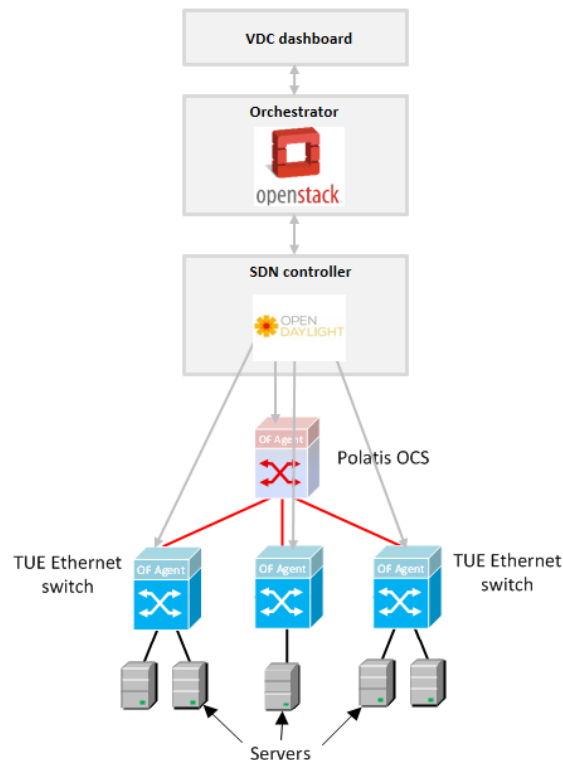


Figure 2– schematic of ECOC demonstrator for the VDC use case

The VDC use case requires:

- Integration of ODL with Polatis and TUE switches
 - Flow creation at Polatis and TUE switches.
 - **Ongoing**, UPC, NXW, UNIVBRIS, Polatis, TUE, beginning of September (dependent on the delivery of the data plane devices).
- Integration of all ODL controller modules
 - **Ongoing**, UPC, NXW, UNIVBRIS, DTU, i2cat end of June/mid July.
- Integration of the SDN controller and the Orchestrator
 - Integration of ODL with Neutron.
 - Finalize the interface/interaction between the algorithms module and ODL for the creation of virtual links.
 - Finalize extensions to Neutron to support the creation of a virtual link.
 - **Ongoing**, UPC, NXW, till mid/end of July.
- Integration of all the orchestrator modules.
 - Finalize OpenStack and ODL clients.
 - Integrate Horizon with the algorithms module.
 - Migrate developed algorithms to the orchestrator environment.
 - **Ongoing**, UPC, i2cat, till mid/end of July.
- Overall integration: Orchestrator + SDN controller + Data plane

- Integration and evaluation of the components of the complete test-bed and their operation as a whole. Evaluation of the end-to-end VDC creation workflow.
- **Ongoing**, UPC, NXW, DTU, i2cat, UNIVBRIS. September.

2.4 Gantt chart for preparation of the ECOC demonstrator

Task (responsible)	June		July		August		September	
Ethernet tester for generating 'background' traffic for the demonstrator (DTU)								
Inclusion of remaining TOR switches in data plane (TUE)								
Optical interfacing of Polatis switch and TOR switch (TUE)								
Assembly of the final demonstrator (DTU)								
Integration with TUE switch sFlow agent (PhotonX)								
ODL integration with the Path Computation (IBM, NXW)								
Integration with the physical observer (IBM, NXW)								
Integration of the physical and virtual observers to Horizon (IBM, i2cat, DTU)								
Integration of Horizon with Heat (IBM, i2cat)								
Integration of ODL with Polatis and TUE switches (UPC, NXW, UNIVBRIS, Polatis, TUE)								
Integration of all ODL controller modules (, UPC, NXW, UNIVBRIS, DTU, i2cat)								
Integration of the SDN controller and the Orchestrator (UPC, NXW)								
Integration of all the orchestrator modules (UPC, i2cat)								
Overall integration: Orchestrator + SDN controller + Data plane (UPC, NXW, DTU, i2cat, UNIVBRIS)								
ECOC demonstrator								

3 Industrial Demonstrator

The industrial demonstration will be performed at the Interoute PoP in Milano Caldera, where an area has been made available to host the COSIGN equipment. Interoute's co-location facility offers to customers either floor space in the common area or a private suite, into which customers can place their own equipment with connections both to the Interoute network and the local city. Each co-location area is provided with all the technical features available at each Interoute production facility, in particular:

- Power supply management: AC & DC power supply designed with N+1 resilience configuration and provisioned to UPS and rectifiers. Each power supply (AC and DC) has two separated and independent power feeds in order to ensure redundancy and guarantee 99.99% power availability.
- Cooling System: temperature and humidity are provided by a heat rejection system with N+1 redundancy configuration that guarantees a 23°C temperature and a humidity rate of 50%.
- Hands & Eyes service: all environmental conditions are monitored and controlled from within the facility with alarms being extended to the Network Operation Centre (NOC) that provides, among others:
 - Soft Reboot/power cycle of customer equipment
 - Provide a visual verification to assist during customers troubleshooting.
 - Relay status of customer equipment LEDs
 - Toggle, cycle a switch or push a button on customer equipment.

In Figure 3, a generic co-location area is represented:

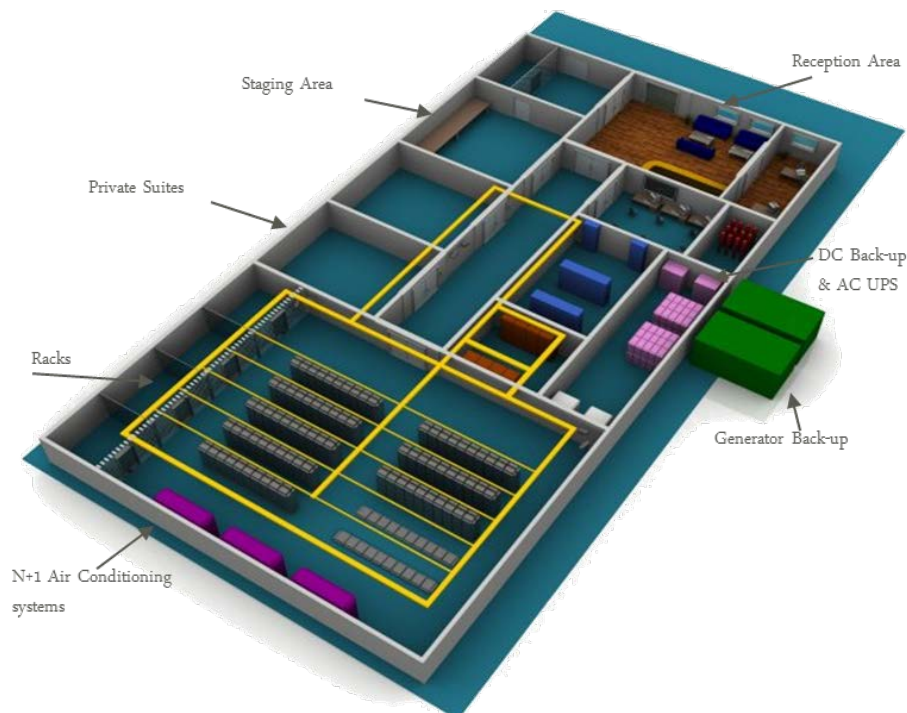


Figure 3 – Co-location site overview

3.1 Plan for the industrial demonstrator

The scope of the demonstration hosted at Interoute is to test the VDC use-case to evaluate the performance of the COSIGN infrastructure.

The demonstrator will rely on the topology and architecture already reported in D 5.0 section 3.2.2 [2]. The hardware provided by Tu/E and Polatis will be integrated with the virtualized infrastructure and deployed within the physical servers available in the Interoute co-location area.

After ECOC the same equipment will be available for the industrial demonstrator at the Interoute premises in Milano. In the agreed plan all the hardware components are delivered by the beginning of October, and the test phase will last approximately 4 weeks. After that period all the TU/e and Polatis switches will be available for the final integration and evaluation phase at DTU premises in Copenhagen.

The demonstration of the VDC use-case will test all the innovative features developed by COSIGN in comparison to services provided by standard VDCs.

The standard VDC service allows users to automatically deploy an Infrastructure-as-a-Service solution by virtualising the physical layer's resources available in a single or in multiple DCs. The on-demand procedure allows users to configure all the resources in terms of compute, storage, networks and other services in order to fulfil a wide set of possible application's requirements.

The COSIGN solution will bring an advanced level of automation that will have an impact on any level of the provisioning chain from the orchestration, to the control and data plane. The improvements in terms of performance in the deployment phase of a VDC instance and the related monitoring and management will be tested and evaluated.

At the orchestration level we shall verify the interaction and the mapping between the virtual and the physical resources, as well as the optimization process defined through the algorithms, and their implementation during the deployment phase.

The control plane functionalities that manage the underlying physical resources will be verified. In particular, attention will be given to the configuration of the optical data plane devices, their utilization parameters and monitoring data. This data is exposed to the orchestration level in order to keep track of runtime information and react accordingly.

The final and most important objective of hosting the VDC demonstrator at the Interoute facilities is the evaluation of the integrated platform with the main purpose of analysing and measuring the behavior of the COSIGN architecture in an industrial exploitation context.

3.1.1 Test plan

In order to have a complete overview of the performances of the COSIGN architecture the following measurements will be carried out:

- Provisioning time: this parameter will measure the time needed to provision a fully configured and running virtual instance on the COSIGN VDC platform. This value will allow the evaluation of the whole provisioning process that includes the orchestration layer, the control layer and the data plane layer.
- Intra-DC network performance: this is a set of parameters related to the evaluation of the COSIGN data plane in terms of RTT, jitter, latency, bandwidth and packet loss.
- Intra-DC Data transfer: different measurements will be conducted to evaluate the overall performance of the optical data plane infrastructure. Most importantly, we will determine throughput of the network as a fraction of its capacity.

4 Final Demonstrator

The final demonstrator in COSIGN will consist of two demonstrator test-beds. One will be a larger implementation of the mid-term scenario also used for the ECOC demonstrator discussed above. This test-bed will be established at DTU upon the completion of the Industrial demonstrator at Interoute. The other test-bed will be based on the “Cluster-based DCN” described in D5.0. This will be implemented at UNIVBIRS. Joint demonstrations will be carried out by interconnecting the two demonstrator sites. Each of the data planes and the use cases to be demonstrated are discussed below.

4.1 The mid-term data plane

For the final project demonstrator, the mid-term architecture, will be implemented on a larger scale compared to the one shown at ECOC. Figure 4 shows the architecture, details of the goals of the demonstrator are given in D5.0 [2]

At the completion of the industrial demonstrator (discussed above) the demonstrator platform will be shipped to DTU (November 1st) where it will be extended into the larger demonstrator platform shown below.

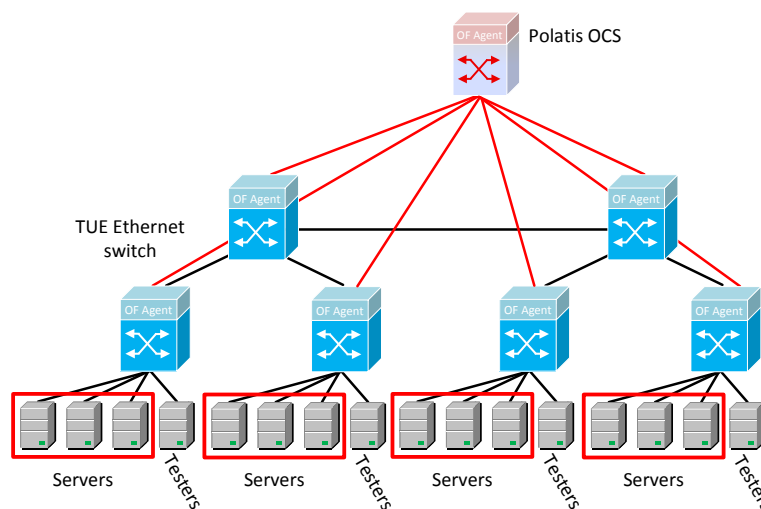


Figure 4- Mid-term architecture in the final demonstrator

This architecture will be used to test the use cases in a slightly larger network allowing performance testing as discussed for each use case below (sections 4.3, 4.4 and 4.5) and in [2].

Interconnecting the mid-term and long-term demonstrators will be achieved using the JANet network which will enable convenient access between DTU and UNIVBRIS demonstrator sites. The target is to have a 10Gbit/s connection between the sites for the duration of the demonstrations. Using this connection we will demonstrate migration of VMs in response to changing application requirements or resource constraints. In a 5G scenario VM migration between DCs in response to latency requirements of applications is expected to be a key functionality. The larger mid-term architecture will comprise a larger number of servers compared to the ECOC demo. Approximately 20 servers are currently available to the demonstrator with the option of extending to more if required. The technical aspects of realizing the demonstrator are identical to what is discussed for the ECOC demo above. The main difference is that a larger network comprising more servers will be constructed in this case to allow testing of the use cases at a greater scale.

Once the ECOC demonstrator is successfully realized the remaining tasks for scaling to the larger demonstrator are:

- Expanding the data plane with additional hardware components – mainly servers and switches. TUE, DTU, finalized by end October
- Preparing host servers – installation of Map-Reduce framework and other required components. DTU, finalized by mid-November
- Data connection between UNIVBRIS and DTU. DTU, UNIVBRIS, available by start November

4.2 The long-term data plane

An overview of the Long Term Demonstrator (LTD) infrastructure is displayed in Figure 5. The demonstrator is divided into clusters, with Polatis switches providing the Top-of-Cluster circuit switching devices. The Top-of-Rack ToR devices are combined Polatis switches and fast switches supporting TDM (this will be the Venture switch if available otherwise the modified optical packet switch from TUE), enabling both sub-wavelength bandwidth division and full-bandwidth circuit cut-through when required. This is supported by the NIC devices that support both TDM and optical circuit output.

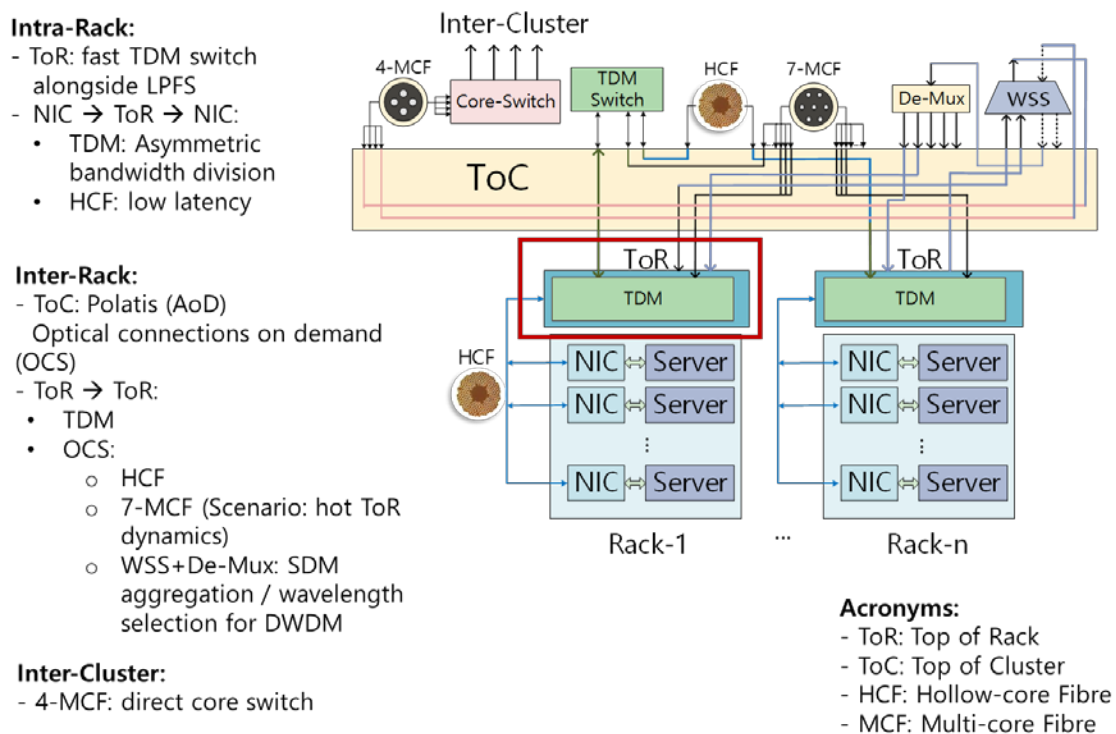


Figure 5 - UNIVBRIS all-optical Long Term Demonstrator infrastructure

In the event of the non-availability of the Venture device, a backup device with very similar functionality has been developed by DTU and TUE and this will be used instead. An OpenFlow agent for this device has already been developed, though TDM support must be added.

- If the FPGA-based NICs are to be used, specific transceivers (SFP+) in the C-band region, equipped with APD receivers will have to be utilized, in order for the optical paths to be maintained regardless of any potential insertion loss from the Hollow-core Photonic Band-Gap Fiber (HC-PBGF) links.
- Synchronization between NICs, especially in the case of fast-TDM might be an issue.

- The nanosecond optical switches that will eventually support fast-TDM (whether it will be synchronous or asynchronous) have to be primarily focused on intra-rack interconnection, where data flow exchanges tend to be shorter in duration but very frequent.
- Multi-core switching provides coarse granularity which is more useful in the aggregation levels of the network, i.e. for inter-cluster or even for inter-DCN communication.

4.2.1 Actions Required for Completion

- Completion of 4-MCF switching device – Southampton & Polatis - Device is ready but has not been tested in an integrated testbed, aiming to integrate by the end of August.
- Agree upon and test method of synchronisation between fast the TDM switches and NICs – UNIVBRIS, DTU and TU/E. The work is ongoing and design details of the TDM scheme are currently being finalised. The target is the beginning of August.
- Orchestration to add support for TDM slot allocation to VDC instances – NXW and UPC. Target for completion is mid-November.
- OpenDaylight Lithium extensions for programming flows with TDM slots, also Li monitoring of TDM allocation - UNIVBRIS. Aiming for completion and integration testing with the data plane by the end of August.

4.3 The vApp use case

In order to scale the demonstrator as compared to the ECOC demonstrator, we'll utilize a Map-Reduce task by using a well-known framework, such as Apache Hadoop [3] or Spark [4]. To that end, the servers will be installed with the Map Reduce framework and run a well-known task e.g., sort, word count, observed by a Map Reduce benchmark suite e.g., BigDataBench [5], HiBench [6].

Map-Reduce is a framework for processing parallelizable problems across huge datasets using multiple computers (nodes), collectively referred to as a cluster, if all nodes are on the same local network and use similar hardware. Processing can occur on either *unstructured* data stored in a file system or on *structured* data stored in a database.

Apache Hadoop is an open-source software framework for distributed storage and distributed processing of very large data sets on computer clusters built from commodity hardware. All the modules in Hadoop are designed with a fundamental assumption that hardware failures are common and should be automatically handled by the framework. The core of Apache Hadoop consists of a storage part, known as Hadoop Distributed File System (HDFS), and a processing part called MapReduce.

Apache Spark is an open source cluster computing framework. Originally developed at the University of California, Berkeley's AMPLab, the Spark codebase was later donated to the Apache Software Foundation that has maintained it since. Spark provides an interface for programming entire clusters with implicit data parallelism and fault-tolerance.

Apache Spark provides programmers with an application programming interface centered on a data structure called the resilient distributed dataset (RDD), a read-only multiset of data items distributed over a cluster of machines, that is maintained in a fault-tolerant way. It was developed in response to limitations in the MapReduce cluster computing paradigm, which forces a particular linear dataflow structure on distributed programs: MapReduce programs read input data from disk, map a function across the data, reduce the results of the map, and store reduction results on disk. Spark's RDDs function as a working set for distributed programs that offers a (deliberately) restricted form of distributed shared memory.

The availability of RDDs facilitates the implementation of both iterative algorithms, which visit their dataset multiple times in a loop, and interactive/exploratory data analysis, i.e., the repeated database-style querying of data. The latency of such applications (compared to Apache Hadoop, a popular MapReduce implementation) may be reduced by several orders of magnitude.

4.3.1 KPIs and indicative base values

The most straight forward and important KPI for Map Reduce tasks is the completion run-time of the task. Therefore, the demonstrator can present completion run-time comparison of base-line and COSIGN architectures for different Map Reduce tasks.

According to our emulated vApp demo, which was presented in the second year review, our initial results with a scaled down scenario suggest that both the RTT and throughput can be improved. Figure 6 presents the RTT improvement for private and shared optical circuits, which results in RTT improvement of 10% and 13% , respectively. Figure 7 presents the throughput improvement for private and shared optical circuits, which results in throughput improvement of 7% and 3%, respectively.

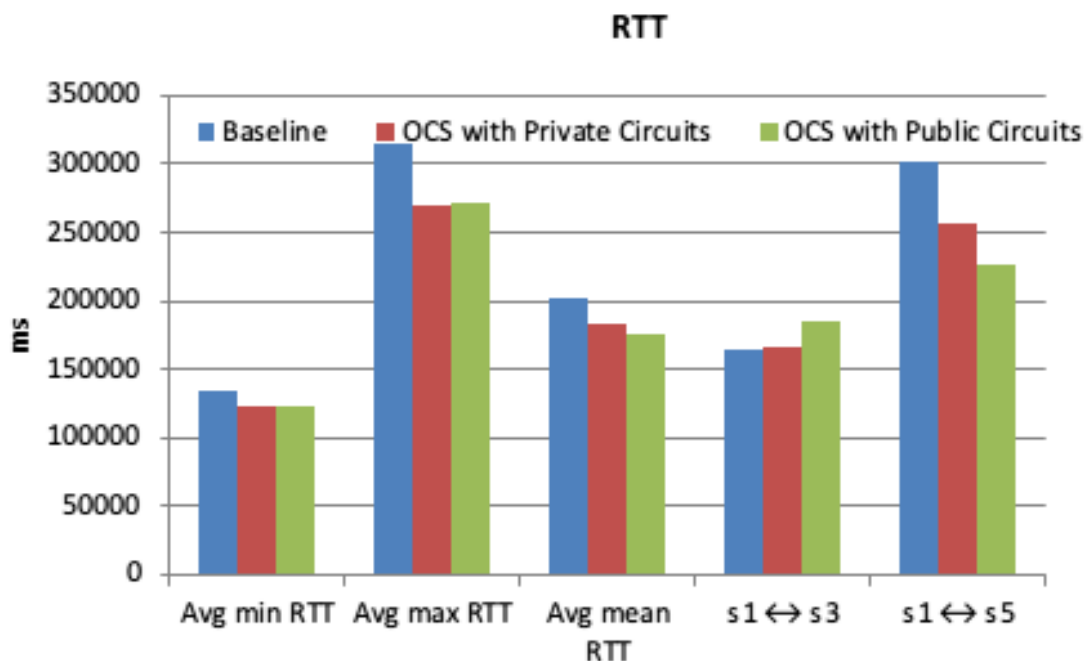


Figure 6 – RTT Improvement by vApp Architecture

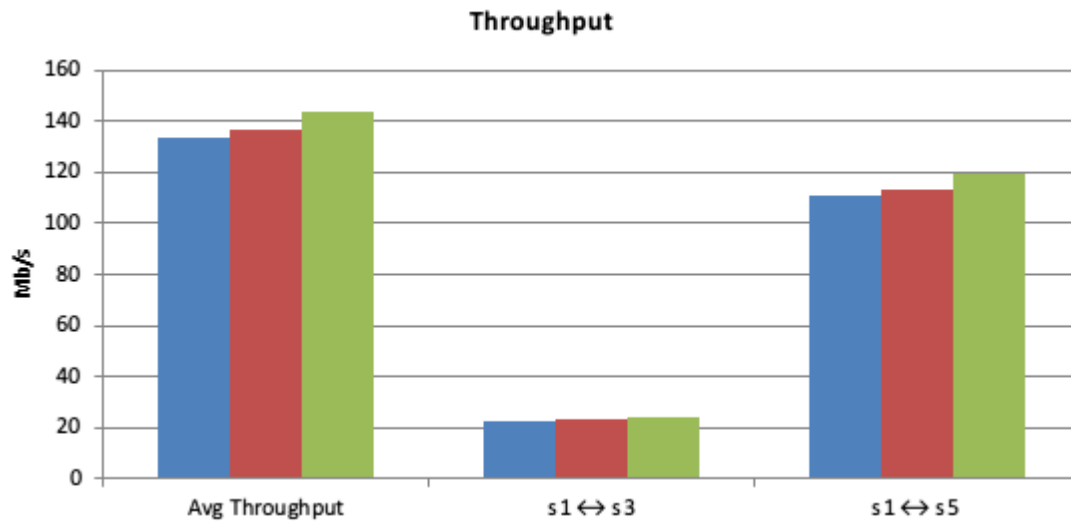


Figure 7 – Throughput Improvement by vApp Architecture (same legend as Figure 6)

4.4 The VDC use case

The VDC use case implementation for the large scale demonstrator is essentially the same as the ECOC demonstrator; the VDC use case benefits from the larger scenario to conduct more exhaustive tests. One especially interesting point is to demonstrate isolation between a significantly large number of VDC instances. One way to demonstrate this is to simultaneously execute high-definition video streams between VMs in every established VDC and assess that the quality of experience is not affected by neighboring VDC owned by other tenants. In order to conduct this experiment, it is necessary to install a video streaming application in the VMs employed to instantiate the VDCs. To this end, an already prepared image with the necessary software will be registered at the Glance service of OpenStack to have the image ready to be used by VDC requests at the dashboard graphical interface.

In the long-term data plane case, both the orchestrator and the SDN controller will change to support the Time Division Multiplexing (TDM)-based resource assignment of virtual links, instead of the fiber link granularity provided by the mid-term data plane.

Virtual links in VDC instances need TDM reservations to guarantee both co-existence with other VDCs and bandwidth:

- The TDM-based optical switches must advertise the frame and the time slot durations, and the resulting number of time slots per frame.
- This information will be needed by the SDN controller and orchestrator. The interface between the orchestrator and the SDN controller must be extended to report on and support TDM in the physical mapping of the virtual links. October-mid-November
- In the orchestrator layer, the mapping algorithms must calculate the optimal time slot scheduling per virtual link. October.
- In the SDN controller, TDM support will need changes to ODL components:
 - the OpenFlow plugin. October-November.
 - the Topology Manager. October-November.
 - the Optical Provisioning Manager. October-November.
 - the Optical Resource Virtualization Manager. October-November.
- Overall integration of the SDN controller, orchestrator and data plane, and testing. December.

Once all the extensions and modifications are implemented, similar performance tests as in the mid-term scenario will be conducted to assess the VDC creation. One additional test will involve the efficiency on the utilization of the optical resources, i.e. the available timeslots.

4.4.1 KPIs and indicative base values

- Provisioning time of a VDC instance (mid- and long-term data planes), considering required time at the orchestrator, SDN controller and the configuration of the physical equipment i.e., Polatis/TUE switches and virtual machines. **Baseline:** around 5 min.
- Spectral efficiency of the employed optical resources (long-term data plane). In the long-term scenario, it is possible to assign each virtual link of a VDC instance with the necessary set of time slots that exactly fit their bandwidth needs. Thus, one important performance indicator consists on determining how well used are the available time slots. **Baseline:** no current baseline is available. The improvement in terms of bandwidth usage provided by the long-term data plane compared to the mid-term case will be measured.

4.5 The VM migration use case

For detailed steps involved in executing the VM migration case we refer to D5.0, Sect. 3.3. Specifically, Sect. 3.3.1 describes the workflow required to trigger migration initialisation, migrate the VMs and virtual networks, then restart the VMs (live migration).

4.5.1 KPIs and indicative base values

KPIs when comparing the networks with and without circuit cut-through were also discussed in D5.0. The most important measure is the time between between halting and restarting the virtual network and VMs. However, the interaction between the orchestrator and hypervisor is key to ensure that faults are correctly detected and reacted to in a timely manner.

More specifically:

- Latency in fault detection and commencing recovery – Baseline below 10s
- Establishing a new virtual network – Baseline to match that of establishing a fresh VDC. In [7] thousands of flows can be migrated in sub-second times.
- Latency in restarting the VMs – Not something we have addressed or introduced novel methods for in COSIGN.
- Latency in migrating VMs to new hosts

Baselines for the latter two can be obtained from the literature, for example [7], [8] and [9]. These two works describe similar full or partial live migrations of virtual networks in cloud environments with purely packet-switched networks. The latency in restarting the VMs is dependent on the size of the VM cluster and the physical compute resources available to initialise the VMs.

We now give general analysis to allow us to obtain a lower bound on the baseline expected latencies for networks and VMs of different characteristics. To calculate the VM copy time, we can use an equation such as that given in Figure 8, which includes the head latency, wire latency and serialisation latency. N is the number of switches traversed between source and destination, r is the latency over a switch and l the latency over a link. We define $\text{Min}(\text{Bandwidth}(L))$ as the minimum bandwidth available on the set of links L that must be traversed, as this is the maximum bandwidth that the transfer will be able to utilise. This allows consideration of the case where there is contention on the links.

$$T_c = N \times r + (N - 1) \times l + \frac{\text{Size}(\text{VMs})}{\text{Min}(\text{Bandwidth}(L))}$$

Figure 8 - Latency for VM cluster copy.

The objective of applying an optical circuit cut-through via the LPFS is twofold. First, to reduce N and reduce the number of switches and links traversed. Secondly, to provide a contentionless route so that the term $\text{Min}(\text{Bandwidth}(L))$ should be the full capacity of the optical links.

However, the VM migration use case is also dependent on the time taken to allocate a fresh VDC network. The baseline given above for this about 5 minutes, which is likely to be significantly greater than the latency for the migration transfer.

4.5.2 Experimental conditions that could be altered to obtain more results include:

- The size and number of VMs
- The complexity of the virtual network
- The network congestion level

4.6 Timeline for preparation of the large scale demonstrator

Activity	Aug 16		Sep 16		Oct 16		Nov 16		Dec 16		Jan 17		Feb 17		Mar 17	
ECOC demonstrator																
Industrial demonstrator																
Additional servers and switches (mid-term data plane)																
Preparing host servers (mid-term demo)																
Completion of 4-MCF switching device (long-term demo)																
Synchronization between fast TDM switch and NICs (long-term demo)																
Orchestration to add support for TDM slot allocation to VDC instances (long-term)																
OpenDaylight Lithium extensions for programming flows with TDM slots, also Li monitoring of TDM allocation (long-term)																
Extended interface between orchestrator and SDN controller																
Mapping algorithm to calculate optimal time slot scheduling.																
Changes to ODL to enable TDM support in SDN controller																
Integration and testing of the SDN controller, orchestrator and data plane,																
Availability of final demonstrator for demonstrator events and academic tests and experimentation																

5 Novelty aspect of the VDC use case

The Virtual Data Centre (VDC) use case has previously been investigated within the LIGHTNESS project. However, the investigations in COSIGN represent a significant step further when compared to LIGHTNESS. We will now outline the main differences, concentrating on the important advances made by COSIGN.

In LIGHTNESS, some algorithms for the optimal/sub-optimal mapping of data center resources according to VDC requests were designed and evaluated. The designed algorithms focused on optimization of the networks resource usage but were customized for the specific hybrid data plane of the LIGHTNESS solution that is using OCS and OPS technologies respectively. The data plane of COSIGN uses different technologies and this is reflected in the design and implementation of the algorithms.

The Orchestration layer, developed in COSIGN WP4, was out of the scope and actually not examined in LIGHTNESS. In LIGHTNESS only the data and control layer have been designed, integrated and experimentally validated, so any provisioning was limited to these layers. Moreover, no equivalent of the algorithm module of COSIGN WP4 was implemented at all.

COSIGN is the first project to design and implement all the architecture layers, from the Horizon-based dashboard to the data plane. The COSIGN orchestrator is based on an OpenStack framework that has been properly extended with the Algorithms module to embed the different algorithms designed. Therefore, in COSIGN, the IT part of the Data Centre is being considered, not only in the optimization algorithms but also in the real deployment of the COSIGN architecture. Moreover, the algorithms/heuristics are customized to the specific COSIGN data plane.

6 Summary of KPIs

KPI	Use case/demonstrator description	Indicative base value	New functionality (no base value exists)
Provisioning time of a VDC instance	VDC	5 min	
Spectral efficiency of the employed optical resources (TDM connections)	VDC	Mid-term vs. long-term comparison in terms of bandwidth usage	No comparable functionality
Round trip time improvement	vApp	10%-13% see Figure 6	
Throughput improvement	vApp	3%-7% see Figure 7	
Latency in fault detection and commencing recovery	VM migration	<10s	
Establishing a new virtual network	VM migration	Establishing a fresh VDC	
Latency in migrating VMs to new hosts	VM migration	Figure 8	

This concludes the demonstrator plan for COSIGN. This document provides detailed information on the planning of demonstrators in the project.

We believe that with this plan we have a good overview of dependencies of components and required time and effort to complete the identified tasks. This planning will be key to ensuring successful completion of all demonstrator activities.