www.vineyard-h2020.eu

D2.3: System architecture

DOCUMENT ID	D2.3	**CONTRACT START DATE**	1st FEBRUARY 2016
DUE DATE	30/04/2017	**CONTRACT DURATION**	36 Months
DELIVERY DATE	18/05/2017		
CLASIFICATION	Confidential		
AUTHOR/S	C. Kachris, A. Bilas, N. Chrysos, H. Vandierendonck		
DOCUMENT VERSION	1.0		

PARTNERS
Institute of Communications and Computer Systems, Maxeler Technologies, Bull Systems, Queen's University of Belfast, Foundation for Research and Technology, The Hartree Centre / Science and Technologies Facilities Council, Neurasmus BV, Neurocom Luxembourg, Hellenic Exchanges SA, Holding, Clearing, Settlement and Registry, LeanXcale, Loba

European Commission

Co-funded by the Horizon 2020 Framework Programme of the European Union under Grant Agreement nº 687628

1 EXECUTIVE SUMMARY

VINEYARD aims at making accelerators easy and transparent to use for the purpose of significantly improving infrastructure efficiency while achieving application-side Quality of Service (QoS). VINEYARD proposes new approaches to integrating accelerators in datacenter environments and to dealing with heterogeneity and transparency issues.

This document describes the overall architecture of VINEYARD and specifically the hardware and the software components that are developed in VINEYARD. VINEYARD's goal is to both develop energy efficient hardware-accelerated servers and to develop the required framework for the seamlessly utilization of these servers in the programming frameworks that are widely used by the applications developers.

To this end, this document describes the VINEYARD platform and the VINEYARD framework. The VINEYARD platform consists of the hardware devices that are used and developed during the project while the VINEYARD framework consists of all the software, middleware, APIs, libraries and GUIs that are developed for the efficient integration of the hardware platforms.

The VINEYARD platform consists of the following hardware platforms:

- Developed in VINEYARD
    - Dataflow engines (Maxeler MAX5)
    - FPGA-based servers (Bull Purley servers using Xeon +FPGA based on QPI interface)
- Available for verification and validation
    - Dataflow engines (Maxeler MAX 4)
    - FPGA-based servers connected to the Intel processors through PCIe
    - MPSoC-based servers based on the Zynq all-programmable FPGAs
    - Xeon Phi accelerators (Knights Landing and Knights Corner)

The VINEYAD framework consists of the following components

- Resource Manager and Scheduler
- Virtualization (VineTalk and VineController)
- VineSim (Simulator for design space exploration and optimization)
- Vineyard Repository of hardware accelerators (VineStore)
- Programming model APIs and drivers
- VineFrame (web-based GUI for the control of the VINEYARD platform)

This document describes the overall architecture of the VINEYARD platform and the VINEYARD framework and how the software and hardware modules developed in the project are integrated into a unified system. As some of these modules are still under development the final integrated system for the VINEYARD framework and the VINEYARD platform will be reported in D6.1 (Integration of the VINEYARD framework, M30) and D6.3 (Integration of programmable accelerated servers, M33) respectively.

PARTNERS
Institute of Communications and Computer Systems, Maxeler Technologies, Bull Systems, Queen's University of Belfast, Foundation for Research and Technology, The Hartree Centre / Science and Technologies Facilities Council, Neurasmus BV, Neurocom Luxembourg, Hellenic Exchanges SA, Holding, Clearing, Settlement and Registry, LeanXcale, Loba
European Commission | Co-funded by the Horizon 2020 Framework Programme of the European Union under Grant Agreement nº 687628

CONTRIBUTORS

Name	Organization
Christoforos Kachris	ICCS
Angelos Bilas	FORTH
Hans Vandierendonck	QUB
Tobias Becker	MAX
Nikos Chrysos	FORTH
Christos Kozanitis	FORTH

PEER REVIEWERS

Name	Organization
Christoforos Kachris	ICCS
Harry Sidiropoulos	ICCS

REVISION HISTORY

Version	Date	Author/Organization	Modifications
0.1	9/05/2017	C. Kachris, ICCS	Initial version
1.0	18/5/2017	C. Kachris, ICCS	Final version with contributions

PARTNERS
Institute of Communications and Computer Systems, Maxeler Technologies, Bull Systems, Queen's University of Belfast, Foundation for Research and Technology, The Hartree Centre / Science and Technologies Facilities Council, Neurasmus BV, Neurocom Luxembourg, Hellenic Exchanges SA, Holding, Clearing, Settlement and Registry, LeanXcale, Loba
European Commission | Co-funded by the Horizon 2020 Framework Programme of the European Union under Grant Agreement nº 687628

Table of Contents

Table of Figures

PARTNERS
Institute of Communications and Computer Systems, Maxeler Technologies, Bull Systems, Queen's University of Belfast, Foundation for Research and Technology, The Hartree Centre / Science and Technologies Facilities Council, Neurasmus BV, Neurocom Luxembourg, Hellenic Exchanges SA, Holding, Clearing, Settlement and Registry, LeanXcale, Loba

European Commission | Co-funded by the Horizon 2020 Framework Programme of the European Union under Grant Agreement nº 687628

Table of Tables

No table of figures entries found.

PARTNERS
Institute of Communications and Computer Systems, Maxeler Technologies, Bull Systems, Queen's University of Belfast, Foundation for Research and Technology, The Hartree Centre / Science and Technologies Facilities Council, Neurasmus BV, Neurocom Luxembourg, Hellenic Exchanges SA, Holding, Clearing, Settlement and Registry, LeanXcale, Loba

European Commission | Co-funded by the Horizon 2020 Framework Programme of the European Union under Grant Agreement nº 687628

2 Related work – A survey on Hardware Accelerators for Cloud Computing

In the last couple of years there are several reconfigurable architectures that have been proposed for the acceleration of cloud computing applications in data centers. This deliverable presents a thorough survey of the FPGA-based accelerators for cloud computing that have been recently presented in the research literature.

The document first presents the frameworks that have been presented for the efficient deployment and virtualization of the FPGA-based hardware accelerators.

2.1 Integrated frameworks for FGPAs in the Data Centers

In the last years, several efforts have been presented for the efficient deployment of FPGAs in the Data Centers. This section presented the most promising integrated frameworks for the efficient deployment of FPGAs in the data centers both from industry and from academia.

2.1.1 IBM SupperVessel

FPGAs in the Cloud

In [1], a general framework is proposed for integrating FPGAs into the cloud by IBM. The framework proposes an accelerator pool (AP) that abstracts FPGA as a consumable resource while avoiding hardware dependencies of current FPGA technologies. In the AP abstraction, each FPGA has several pre-defined accelerators slots in which the hardware accelerators can be mapped. By utilizing the partial reconfiguration mechanism of the FPGAs, each slot can be considered a virtual resource that can be assigned for specific tasks. A cloud tenant can submit either pre-defined hardware accelerators that are hosted in central repository or can submit his own designs. However, in the latter case the cloud owner should perform the synthesis, place and route and generate the bitstreams for the FPGA slots.

The proposed framework supports two different methods for translating address between the guest physical address of the virtual machines and the host physical address. The first method copies data between the VM memory and the host buffers. This method (*VM-copy*) is easy to implement but a creates a data copy overhead. Another method, called *VM-nocopy*, maintains a fixed mapping between the virtual and the host address space. This method does not introduce a data copy overhead but it

[1] F. Chen, Y. Shan, Y. Zhang, Y. Wang, H. Franke, X. Chang, andK. Wang, "Enabling fpgas in the cloud," in Proceedings of the11th ACM Conference on Computing Frontiers, ser. CF '14. New York, NY, USA: ACM, 2014, pp. 3:1–3:10.

PARTNERS
Institute of Communications and Computer Systems, Maxeler Technologies, Bull Systems, Queen's University of Belfast, Foundation for Research and Technology, The Hartree Centre / Science and Technologies Facilities Council, Neurasmus BV, Neurocom Luxembourg, Hellenic Exchanges SA, Holding, Clearing, Settlement and Registry, LeanXcale, Loba

European Commission | Co-funded by the Horizon 2020 Framework Programme of the European Union under Grant Agreement nº 687628

requires several modifications to the host OS to reserve large blocks of physic memory and modifications to the memory allocation methods of VM.

A prototype of the framework is implemented on an x86-based Linux-KVM environment with attached FPGAs and deployed in a modified OpenStack cloud environment. Four different accelerators are used for prototyping: Encryption (AES), Hashing (SHA), Stereo matching and Matrix-Vector Multiply. The performance evaluation shows that proposed framework allows the efficient utilization of the FPGA resources by the cloud tenants with less than 4 microseconds latency overhead of the visualization.

HyperScale Data Centers

A similar framework is also proposed by IBM Zurich that allows cloud users to combine multiple FPGAs in the programmable fabric[2]. This allows cloud operators to offer an FPGA to users in a similar way as a standard server. In the proposed framework, multiple user applications can be hosted on a single physical FPGA, somehow similar to multiple VMs running on the same hypervisor. Each user can get a partition of the entire user logic and uses it to implement its own applications. This partitioning is achieved by utilizing partial reconfiguration. With partial reconfiguration, it is possible to dynamically reconfigure a portion of the FPGA while the rest of the regions remain untouchable.

The users first decide on the required number of vFPGAs and customizes them using their own custom hardware accelerators. The user can then define its fabric topology by connecting those customized vFPGAs. When a request from a cloud user arrives, the accelerator scheduler searches the FPGA pool to find a user logic resource that matches the vFPGA request. When the scheduler matches the request with the required FPGA then the interface for access to the vFPGA is offered to the user. Finally, the user rents the defined virtualized fabric from the IaaS vendor.

The proposed architecture has been integrated into the OpenStack framework and allows the renting of the FPGA resources on the cloud. The same architecture also allows the possibility to distribute their applications on a large number of FPGAs through an FPGA fabric. The integrated framework with multiple FPGAs is compared with a typical data center based on commodity processors. It is shown that if each system is based on 2048 module the FPGA-based system can provide 958 TFLOPS compared with the 442 TFLOPS offered by the commodity processors.

[2] J. Weerasinghe, F. Abel, C. Hagleitner, and A. Herkersdorf, "Enabling fpgas in hyperscale data centers," in 2015 IEEE International Conference on Cloud and Big Data Computing (CBDCom 2015), May 2015.

PARTNERS
Institute of Communications and Computer Systems, Maxeler Technologies, Bull Systems, Queen's University of Belfast, Foundation for Research and Technology, The Hartree Centre / Science and Technologies Facilities Council, Neurasmus BV, Neurocom Luxembourg, Hellenic Exchanges SA, Holding, Clearing, Settlement and Registry, LeanXcale, Loba
European Commission | Co-funded by the Horizon 2020 Framework Programme of the European Union under Grant Agreement nº 687628

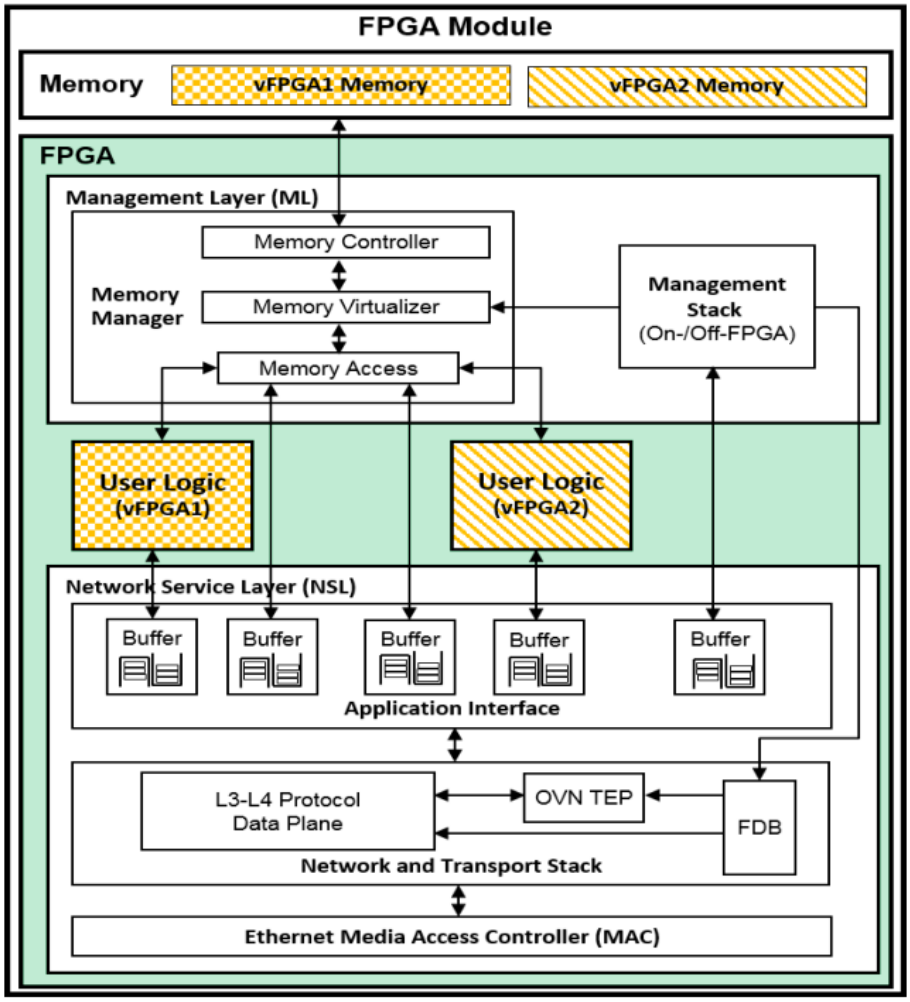

Figure 1. Hyperscale Data Centers from IBM

2.1.2 Virtualized Hardware accelerators, University of Toronto

In [3], a novel approach for integrating virtualized FPGA-based hardware resources into cloud computing systems with minimal overhead. The proposed framework allows cloud users to load and utilize hardware accelerators across multiple FPGAs using the same methods as the utilization of Virtual Machines. The reconfigurable resources of the FPGA are offered to the users as a generic cloud resources through OpenStack.

An agent is introduced in this framework that implements the resource management of the OpenStack. The proposed framework splits the FPGA into several reconfigurable regions, each of which is managed a s single resource. Therefore, instead of a single

[3] S. Byma, J. Steffan, H. Bannazadeh, A. Leon-Garcia, and P. Chow, "FPGAs in the cloud: Booting virtualized hardware accelerators with open-stack," in Field-Programmable Custom Computing Machines (FCCM), 2014 IEEE 22nd Annual International Symposium on, May 2014, pp. 109–116.

PARTNERS
Institute of Communications and Computer Systems, Maxeler Technologies, Bull Systems, Queen's University of Belfast, Foundation for Research and Technology, The Hartree Centre / Science and Technologies Facilities Council, Neurasmus BV, Neurocom Luxembourg, Hellenic Exchanges SA, Holding, Clearing, Settlement and Registry, LeanXcale, Loba

European Commission | Co-funded by the Horizon 2020 Framework Programme of the European Union under Grant Agreement nº 687628

FPGA bitstream, a collection of partial reconfigurable bitstreams corresponding to the user hardware is passed to the agent. Again, as in the case of the IBM, the cloud provider must generate the partial bitstream for each accelerator and for each partially reconfigurable slot since the current technology requires specific bitstreams for each region of the FPGA.

The proposed system can set up and tear down virtual accelerators in 2.6 on average. The static virtualization hardware on the physical FPGA has a minimum overhead (only three clock cycles). The proposed system is implemented and evaluated in a NetFPGA-10 platform. The prototype system implemented four virtual reconfigurable modules on one device.

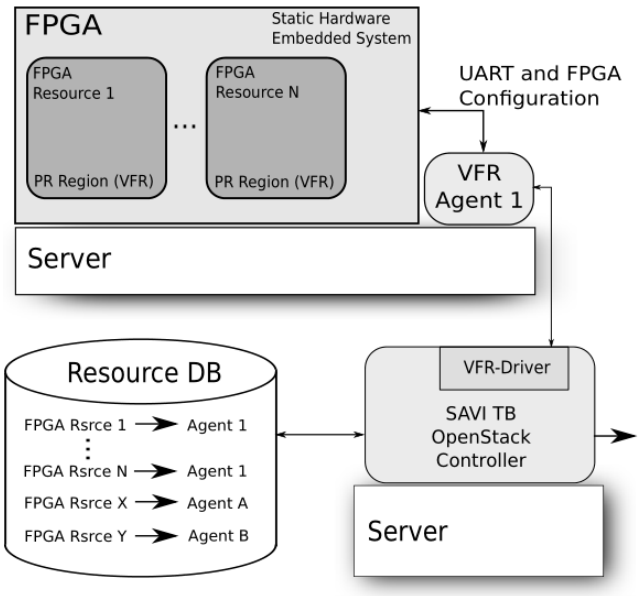

Figure 2. Toronto's system overview for virtualized FPGAs in the data centers.

2.1.3 RC3E: Reconfigurable Cloud Computing Environment

In [4], a cloud hypervisor is proposed by Technical University of Dresden that integrates virtualized FPGA-based hardware accelerators into the cloud environment. The hypervisor allows users to implement and execute their own hardware designs on virtual FPGAs. The hypervisor has access to a database containing all physical and virtual FPGA devices in the cloud system and their allocation status. Each device is assigned to its physical host system (node).

[4] O. Knodel and R. G. Spallek, "RC3E: provision and management of reconfigurable hardware accelerators in a cloud environment," in 2nd International Workshop on FPGAs for Software Programmers, 2015

PARTNERS
Institute of Communications and Computer Systems, Maxeler Technologies, Bull Systems, Queen's University of Belfast, Foundation for Research and Technology, The Hartree Centre / Science and Technologies Facilities Council, Neurasmus BV, Neurocom Luxembourg, Hellenic Exchanges SA, Holding, Clearing, Settlement and Registry, LeanXcale, Loba

European Commission | Co-funded by the Horizon 2020 Framework Programme of the European Union under Grant Agreement nº 687628

The user can allocate a complete physical FPGA, which has to be marked separately in the device database or can allocate portion of the vFPGA. In the case of vFPGA allocation, the configuration is performed by utilizing partial reconfiguration (PR). The proposed architecture included also a Reconfigurable Cloud Computing Framework (RC2F), that is used for the realizing the vFPGA concept and allows integration of user cores. The main part of the RC2F framework consists of a controller managing the configuration and the user cores as well as the monitoring of status information.

The proposed framework supports the required security by protecting the device files using access rights. This additional virtualization layer allows concurrent users to interact with their allocated devices without influencing each other. The proposed framework is prototyped using a matrix multiplication applications. The matrix multiplication offers both high amounts of data and computational complexity. The host application starts individual parallel user threads sending matrices to the cores, measures runtime and calculates the throughput.

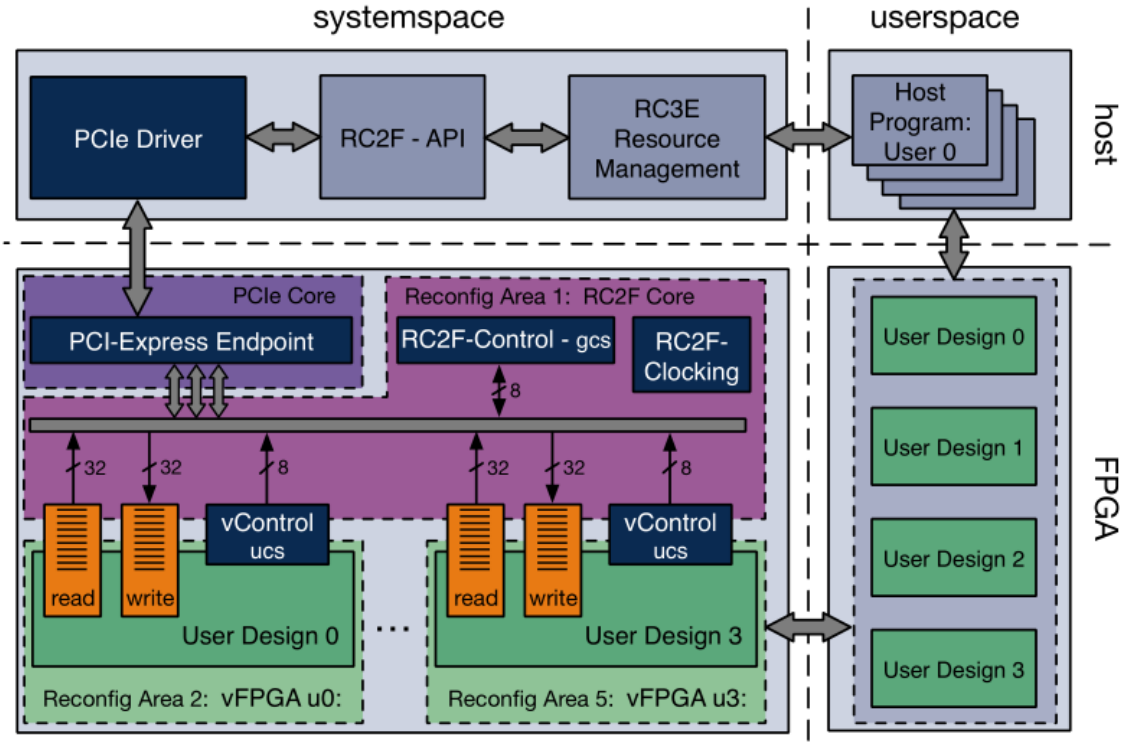

Figure 3. System overview of the RC3E architecture

PARTNERS
Institute of Communications and Computer Systems, Maxeler Technologies, Bull Systems, Queen's University of Belfast, Foundation for Research and Technology, The Hartree Centre / Science and Technologies Facilities Council, Neurasmus BV, Neurocom Luxembourg, Hellenic Exchanges SA, Holding, Clearing, Settlement and Registry, LeanXcale, Loba
European Commission
Co-funded by the Horizon 2020 Framework Programme of the European Union under Grant Agreement nº 687628

2.1.4 Virtualized FPGA accelerators, University of Warwick

In [5], a novel framework is presented that integrates reconfigurable accelerators in a standard server with virtualized resource management and communication. The proposed framework integrates a PCIe based FPGA board into a standard data center server. The FPGA is partitioned into separate accelerator slots. Accelerator functions are either stored in a library on the host machine as partial bitstreams or can be uploaded by the user.

Each physical FPGA is divided into multiple partially reconfigurable regions (PRRs), which act as virtual FPGAs (vFPGAs) for hosting accelerators. vFPGAs are interfaced with the host FPGA's PCIe and DRAM physical interfaces for communication and data storage. The logic to manage these physical interfaces is implemented in the FPGA static logic. A vFPGA can be configured with a compatible partial bitstream to implement a virtual FPGA accelerator (vFA). A vFPGA can host multiple vFAs. %To enable portability and simplify vFA design, the vFPGAs all have a standard interface: a single AXI4-Stream interface to the PCIe core and another AXI4-Stream interface to external DRAM.

In this framework, a hypervisor is implemented for the configuration and the scheduling of the user logic in the FPGA resources. When an accelerator is to be configured in the FPGA, the hypervisor decides on the optimal partial reconfiguration regions (PRRs) to host it and initiates reconfiguration. The hypervisor also maintains a list of the available PRRs and configured accelerators to avoid unnecessary reconfiguration when a required accelerator is already present in the FPGA and not in use. As a use-case study, it has been implemented a map-reduce accelerator for word counting, which finds the number of occurrences of a specified word in a large data set, useful in data mining applications. %In this use-case 8 mappers have been implemented in the proposed framework in a VC709 board consuming only 24W.

The virtualized FPGA implementation surpasses the software only computational efficiency once the FPGA is used over 12% of the time.

[5] S. A. Fahmy, K. Vipin, and S. Shreejith, "Virtualized FPGA accelerators for efficient cloud computing," in 7th IEEE International Conference on Cloud Computing Technology and Science, CloudCom 2015, Vancouver, BC, Canada, November 30 - Dec. 3, 2015, 2015, pp. 430–435.

PARTNERS
Institute of Communications and Computer Systems, Maxeler Technologies, Bull Systems, Queen's University of Belfast, Foundation for Research and Technology, The Hartree Centre / Science and Technologies Facilities Council, Neurasmus BV, Neurocom Luxembourg, Hellenic Exchanges SA, Holding, Clearing, Settlement and Registry, LeanXcale, Loba

European Commission | Co-funded by the Horizon 2020 Framework Programme of the European Union under Grant Agreement nº 687628

3 VINEYARD System architecture

The VINEYARD project is developing novel servers coupled with FPGA accelerators and programmable dataflow engines and that can be customized based on the data-centre's application requirements. These programmable FPGA accelerators and dataflow engines will be used not only to increase the performance of servers but also to reduce the energy consumption in data centres.

Furthermore, VINEYARD is also developing a programming framework that will hide the complexity of programming heterogeneous systems while at the same time providing the optimized performance of customized and heterogeneous architectures. VINEYARD is also developing a new programming framework that seamlessly leverage workload-specific accelerators based on the application requirements. In this suite, the user works with familiar programming frameworks (i.e. Spark) while a run-time manager selects appropriate accelerators based on application requirements such as execution time and power consumption.

This section provides the overall system architecture of the VINEYARD framework and the interfaces between the modules.

Figure 4 depicts the high-level overview of the VINEYARD platform. Applications that are targeting heterogeneous data centers using traditional servers or micro-servers are programmed using traditional data center frameworks, such as Spark, or more application specific frameworks such as the PyNN framework that is used for neural networks. In these applications, VINEYARD will provide the required APIs that will enable the utilization of the heterogeneous infrastructures without any other modifications in the sources codes.

However, some of the tasks are common across several applications such as sorting of data, key/value processing, encryption, compression, pattern matching, etc. and are extremely computationally intensive. These tasks can be implemented in hardware as customized intellectual-property (IP) accelerators that can achieve much higher performance with lower power consumption. These hardware accelerators are stored in an IP repository (VineStore) that interface with the VINEYARD resource manager and scheduler. For each application there are several version of the accelerators based on the available platform (FPGA, DFE, Xeon Phi, MPSoC).

The resource manager allocates resources from the heterogenous platform and dispatch the jobs to these nodes based on the application requirements. The resource manager communicates with the Maxeler Orchestrator and the FPGA/MPSoC orchestrator that keep the information of the status of these accelerators. The Resource manager also contains the IP Library controller that is used to fetch and dispatch the right hardware accelerator from the IP library based on the available resources (FPGA, DFE, Phi, or MPSoC).

Each node in the platform can host a typical high performance general purpose processor, a dataflow engine, an FPGA-based server or an MPSoC-based server. The

PARTNERS
Institute of Communications and Computer Systems, Maxeler Technologies, Bull Systems, Queen's University of Belfast, Foundation for Research and Technology, The Hartree Centre / Science and Technologies Facilities Council, Neurasmus BV, Neurocom Luxembourg, Hellenic Exchanges SA, Holding, Clearing, Settlement and Registry, LeanXcale, Loba
European Commission | Co-funded by the Horizon 2020 Framework Programme of the European Union under Grant Agreement nº 687628

specifications and the details for each platform that is used and the VINEYARD framework are described in detail in the following sections. The Software stack of each node contains the VMs that are running on the processor, the Local scheduler that dispatches the job to the local accelerators, the VineTalk that allows the virtualization of the underlying hardware resources, and the VineController that serializes the jobs to the hardware resources. VineController communicates with the accelerator' drivers that are based on the accelerators' vendors.

On top of the VINEYARD resource manager, a web-based GUI, called VineFrame, can be used to allow the control and the utilization of the available resources in an easy-to-use graphical interface. For example, the VineFrame can be customized to the applications for each heterogeneous platform (BrainFrame allows the utilization of the Pynn framework in the VINEYTARD heterogenous resources).

Each section, in this document describes in more detail the building blocks of the VINEYARD platform and the VINEYARD framework.

PARTNERS
Institute of Communications and Computer Systems, Maxeler Technologies, Bull Systems, Queen's University of Belfast, Foundation for Research and Technology, The Hartree Centre / Science and Technologies Facilities Council, Neurasmus BV, Neurocom Luxembourg, Hellenic Exchanges SA, Holding, Clearing, Settlement and Registry, LeanXcale, Loba
European Commission | Co-funded by the Horizon 2020 Framework Programme of the European Union under Grant Agreement nº 687628

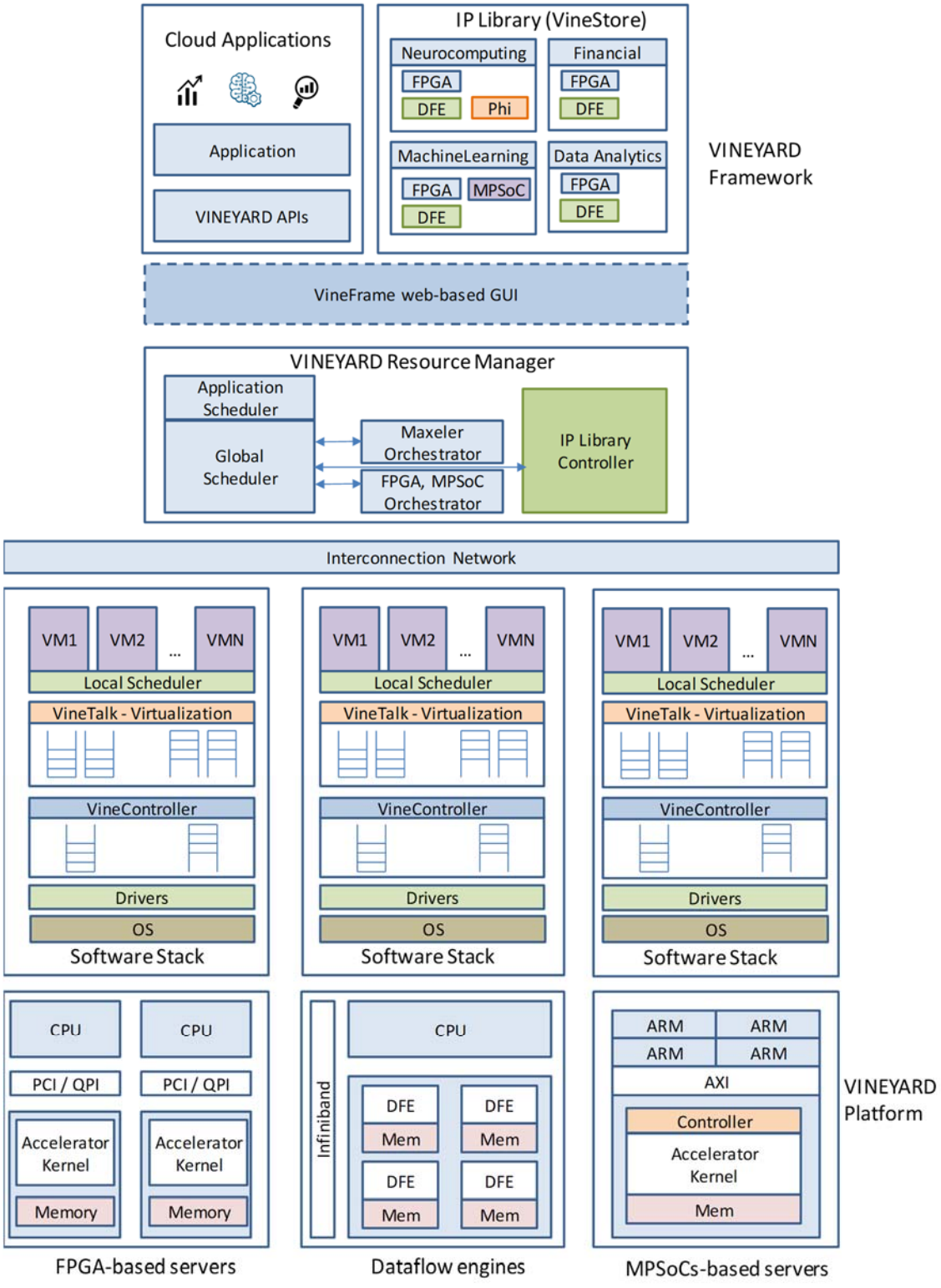

Figure 4. VINEYARD overall system architecture.

PARTNERS
Institute of Communications and Computer Systems, Maxeler Technologies, Bull Systems, Queen's University of Belfast, Foundation for Research and Technology, The Hartree Centre / Science and Technologies Facilities Council, Neurasmus BV, Neurocom Luxembourg, Hellenic Exchanges SA, Holding, Clearing, Settlement and Registry, LeanXcale, Loba
European Commission
Co-funded by the Horizon 2020 Framework Programme of the European Union under Grant Agreement nº 687628