Fish4Knowledge Deliverable D4.2
Workflow Computational Platform

Principal Author: Fang-Pang Lin
Contributors: NARL
Dissemination: PU

Abstract: This document describes the computational platform built for the system of Fish4Knowledge(F4K) project, which allows video and image processing (VIP) components that constitute the tasks of the workflow to be composed and executed efficiently over a Tera-scale video dataset. The infrastructure of the platform includes 2 multicore machines with 48 and 96 cores for development and production purposes respectively, and one auxiliary 8 core PC cluster to support the migration between different systems. The required VIP components, specifically for the in-house developed 30 image processing (IP) executables, are installed and tested preliminarily. A process execution platform based on virtual machines is developed to assist F4K developers to migrate and integrate the codes from local development sites to the centralized production site.

Deliverable due: month 12

Table of Contents

1. Introduction

The system of Fish4Knowledge (F4K) project is designed to respond to the online query from marine scientists using the analysis of underwater video data from daily streaming and historical store[6,7]. In additional to the historical tera-scale video data, the size of underwater video data keeps growing approximately 6.21T/month. The scale of the data is the major challenge for the system to be developed. To meet the challenge the underlying infrastructure should be able to cope the problems of how to store, excess and how to analyze the large scale data. Such an infrastructure should incorporate data storage, compute facilities and networking [3,4]. In this report, the preparation of the computational platform for the system, which allows F4K workflow system to perform real-time video and image processing (VIP), will be presented.

F4K workflow system consists of VIP ontology, VIP components and workflow enactor and planner [1]. The performance of the system mainly depends on data I/O and data analysis, i.e. execution of the VIP components. The VIP components are currently a complete set of 30 independent executables for pre-processing, initialization and the calculations of predominant colors, texture features, detection, tracking and video classification etc. The F4K workflow system will use these executables as primitives and compose them into a solution that given by the workflow planner according to a set of rules, which are derived from higher level reasoning via VIP ontology.

The underlying compute infrastructure takes advantage of the architecture of general-purpose multicore processors, which are being accepted in all segments of the industry, including machine vision and image processing, as the need for more performance and general-purpose programmability has grown. The machine can easily scale out to increases performance by adding more parallel resources while maintaining manageable power characteristics. The architecture is well suited to the current computational need of the VIP.

In the following sections, the design of the F4K computational platform, which allows VIP components that constitute the tasks of the workflow to be composed and executed, will be presented first (Section 2). Then the required VIP components, specifically for the in-house developed 30 image processing (IP) executables, are installed and tested (Section 3). Finally, a virtual machine is built as a process execution platform to assist F4K developers to migrate and integrate the codes from their local development sites to the F4K production system.

2. Workflow Computational Platform Design

2.1 Basic Requirements

The F4K system aims at providing a production service. There are two phases of development to be considered. One is the development phase and the other production phase. In the development phase, the platform should be able to allow developers to work independently and able to access all video data for various tests, such as code verification and validation, scalability enhancement, sensitivity analysis and performance tuning. Also, the developers are able to migrate and integrate their codes into a unified workflow environment. In the production phase, the compute resources should be sufficient to meet the real-time VIP analysis.

2.2 Architecture Design

to the basic requirements, three-tier architecture system design are employed for F4K project (Figure 2.1), which includes Tier-1 for web portal, Tier-2 for computational service and Tier-3 for storage service[5]. In other words, presentation, application processing, and data management are logically separated into three-tier structure from Tier-1 to Tier-3 correspondingly [2]. The multi-tier architecture design is well known for its flexibility and reusability, which is crucial for F4K's system development from prototype to production. Internal communication in Tier-3 is through standard protocols, such as NSF/Samba and SSH/RPC. Externally, there are 10GB/s networks excepted be connect between the Tier-2 compute service and Tier-3 storage service to ensure the performance. The developers are allowed to login to VM server and the PC Cluster. But the domain users, e.g. marine scientists, are only allowed to access via specific web service in Tier-1. This report aims the development at the compute service and its web service from Tier-1 and Tier-3.

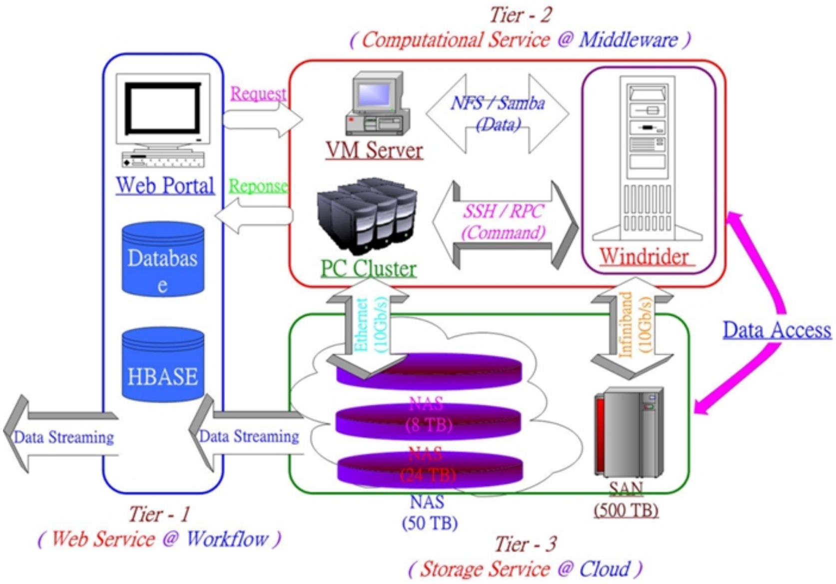

Figure 2.1: The three-tier architecture design for high performance compute and storage services.

2.3 Resources and Constrains

Following the three-tier architecture, three types of machines, a dedicated backend compute server, a virtual machine (VM) server and an auxiliary PC cluster, are considered in Tier-2. The backend computer server will be used as compute engine for production. Most development will be conducted in the VM server and the auxiliary PC cluster.

2.3.1 The Backend Computational Server

The backend compute server plays as a major computational power house in F4K system. NCHC Super Computer: “Advanced Large-Scale Parallel Supercomputer” ALPS (also known as WindRider) – AMDx86 64 platform with 12 core on-die. 2 nodes x 48 cores machine is used as a dedicated machine for compute. The configuration, status, access and software packages of the ALPS are described in details in the following tables.

Host & Compute Node	
Machine Name	ALPS — Acer AR585 F1 Cluster
Machine Structure	SMP Cluster (2 nodes, 96 cores)
Processors	AMD Opteron 6174, 12 cores, 2.2GHz (compute nodes)
Main Memory (per node)	128 GB (compute nodes) 4 HT Links with “25.6 GB/s” bandwidth (the speed of each HT Links is “6.4GT/s”)
Operating System	Novell SuSE Linux Enterprise 11 SP1

Job Scheduler & Queuing System	Platform LSF (Load Sharing Facility) 7.06

Table 2.1: Host and compute nodes of the ALSP for the F4K project.

Software Stack	
Installing Software	ABINIT, Amber, CASINO, CHARMM, Chemsoft, CPMD, DL_POLY, GAMESS, Gaussian, GROMACS, Molpro, NAMD, NWChem, octopus, OpenMX, Quantum ESPRESSO, siesta, VASP, WIEN2k
Development Tools	Intel Cluster Toolkit, PGI CDK/Server, x86 Open64, GCC
MPI/OpenMP	Platform MPI (formerly HP-MPI), Intel MPI, PGI MPI/OpenMP, MVAPICH/MVAPICH2
Math	Intel MKL (Math Kernel Library), AMD ACML (AMD Coe Math Library)

Table 2.2: Software stack provided in the ALPS.

2.3.2 The Virtual Machine Server

The VM server is used to cope with multiple requirements from various the F4K work packages, and also takes responsibility to provide operational services. The VM Server (also known as gad246) is AMDX86 64 platform with 12 cores on-die machine. It allows developers to invoke virtual machines and install their own development environments. The machine is close in location to the video data. Therefore, the benefit would be supporting incremental migration and integration of production codes as well as effective large data movement. The configuration, status, access and software packages of the VM Server are described in details in the following tables.

Host & Compute Node	
Machine Name	Gad246 — Tyan 8812F48W8HR
Machine Structure	SMP System
Processors	AMD Opteron 6176 SE, 12 cores, 2.3GHz
Main Memory (per node)	128 GB (compute nodes) 4 HT Links with “25.6 GB/s” bandwidth (the speed of each

	HT Links is “6.4GT/s”)
Operating System	Ubuntu 10.04.1 LTS

Table 2.3: Host and compute nodes of the F4K VM server.

Software Stack	
Development Tools	Intel Cluster Toolkit, PGI CDK/Server, x86 Open64, GCC, OpenCV
MPI/OpenMP	PGI MPI/OpenMP, MVAPICH/MVAPICH2
Math	Intel MKL (Math Kernel Library), AMD ACML (AMD Coe Math Library)

Table 2.4: Software stack installed in F4K VM server. Customized software stacks can be installed in personal VMs, hence there is no pre-installed software for users.

2.3.3 The PC Cluster

The PC cluster is a master node (also known as gad247) with Intelx86 64 platform with 8 cores on-die. It is used as an auxiliary uniform environment, which provides developers to migrate and test their codes from their own local environments to the production one. The configuration, status, access and software packages of the PC Cluster are described in details in the following tables.

Host & Compute Node	
Machine Name	Gad246 — ASUS MD-710
Machine Structure	SMP System
Processors	Intel core i7 2600, 4 cores, 3.4G Hz
Main Memory (per node)	4 GB (compute nodes)
Operating System	Ubuntu 10.04.1 LTS

Table 2.5: Host and compute nodes of the PC Cluster.

Software Stack	
Development Tools	Intel Cluster Toolkit, PGI CDK/Server, x86 Open64, GCC, OpenCV, MySQL, PostgreSQL
MPI/OpenMP	PGI MPI/OpenMP
Math	Intel MKL (Math Kernel Library), AMD ACML (AMD Coe Math Library)

Table 2.6: Software stack installed in the PC Cluster. The stack is

shared between developers. It can be updated when needed.

2.3.4 Constrains

There are a few hardware constrains required further attention.

- There is lack of holistic 10GB/s network interface card (NIC) support in storage and system servers. Currently most internal network connections are 1GB/s. Although parallel I/O can be used as one of the solutions to ease the problem, it is still required further investment to replace all 1GB/s NICs with 10 GB/s NICs.
- Hard disk space is limited in ALPS. The hard disk space in ALPS is 200TB shared by all users. Also, the connection between ALPS and other machines is limited due to security. Therefore, it is required to devise a method to ensure the performance of F4K system is not bottlenecked in the data I/O between ALPS and other machines during the execution.

3. Workflow Execution and Process Components

3.1 Install Essential Libraries for Video and Image Processing

The essential packages (Table 3.1) are installed for video and image processing (VIP). The VIP components of the F4K workflow system takes advantage of the rich functionality of VIP that is already developed in OpenCV. Top on OpenCV libraries, 30 image processing (IP) codes used in [1] are used as the basic building blocks VIP components.

Package & Library	
PKG Name	***PKG Version***
OpenCV	**OpenCV-2.3.1**
Ffmpeg	**ffmpeg-0.7.8**
Faac	**faac-1.28**
x264	**x264 core:115**
Xvidcore	**xvidcore-1.3.2**
Yasm	**yasm-1.1.0**
GTK+	**GTK+-2.0**
Cmake	**2.8.5**
Gcc	**4.3.4**

Table 3.1: Essential libraries installed for VIP components.

3.2 Process Components Preliminary Test

We have tested image processing program by experiments within WinRider (ALPS) machine. The experiment consisted of the major IP codes from Nadarajan [1]. The most IP programs in our preliminary test environment are running normally. IP team is producing new, more efficient code as well as some additional Matlab codes.

Here is the summary of testing in ALPS:

- 30 IP codes had been preliminarily tested
- Essential Libraries had been setup ad major IP codes run well.
- These codes are for preliminary test only.
- ALPS has Matlab installed and is accessed via SSH and SFTP only

4. Process Execution Interface

A web-based process execution interface is developed to allow developer to use the F4K computational resources that described in Section 3. The advantage of portability of web-based service is well received. It is also crucial for us to provide a conduit to allow F4K developers to use NCHC internal resources transparently. To tackle this problem is to couple web-service with virtual machine and to allow the virtual machine with specific security control to access the whole computational resources. Figure 2 shows the web-based VM portal prototype that we developed. The developers can click on the pre-installed system image, with customized environment, and spawn a virtual machine. When finished, return the resources back to the VM server.

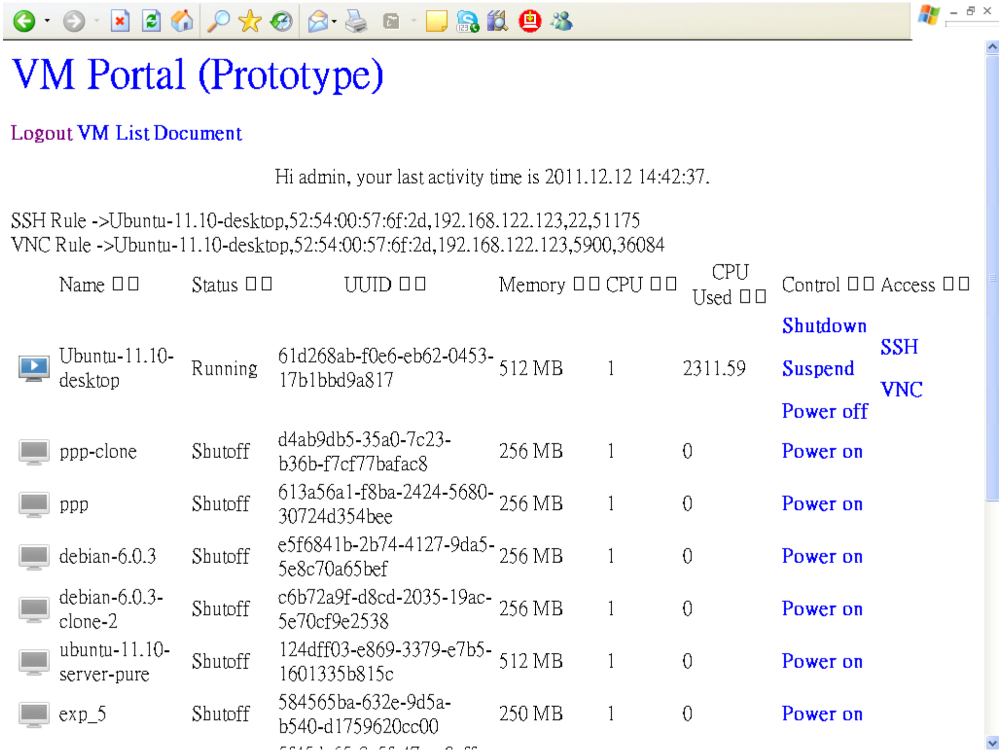

Figure 4.1: The Web-based VM portal prototype.

The design of the portal is shown in Figure 4.1, where the standard API “libvert” tool is used for VM management at front end and also responsible for communication between the portal and backend VM server. A port forwarding method is employed to make sure the specific private IP addresses can be access via external users. VM server will be responsible for the backend VM management. Currently, only Linux kernel is provided in the VM server.

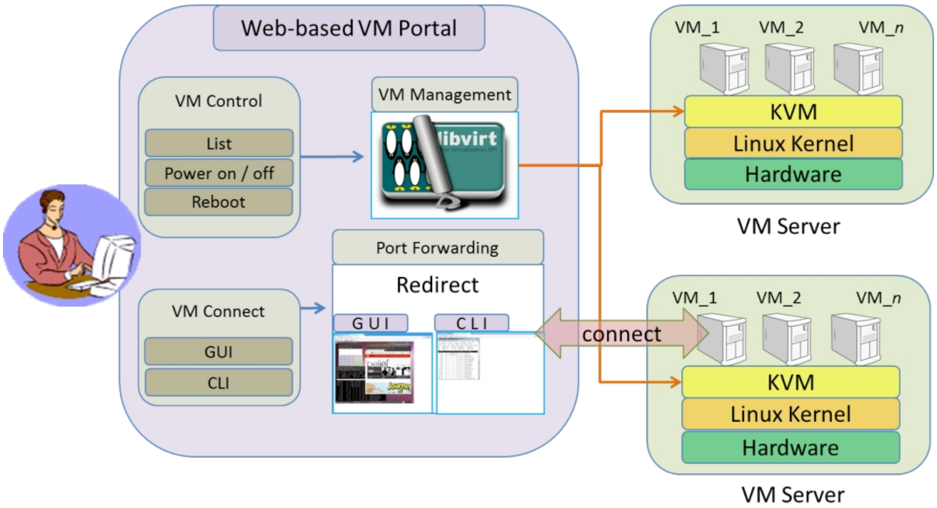

Figure 4.2: VM (Virtual Machine) Portal provides computational resources to developers.

There are some considerations in practice for using the VM.

- The VM Web-Portal is still evolving now. At this stage the light weight PC cluster (gad247) can be used as an alternative resource.
- The current VMs are bundled with private IPs. The system images of the VMs have to be deployed manually by administrator.
- The international partners, i.e. external developers, are suggested to build up the computing environment of VIP from European research network GEANET, which is already peering with Taiwan's research network TWAREN.

5. Conclusions

In this work, the workflow computational platform is designed and built for the system of Fish4Knowledge (F4K), which allows video and image processing (VIP) components that constitute the tasks of the workflow to be composed and executed efficiently over a Tera-scale video dataset. The work is concluded as the following:

- Totally 152 cores available for workflow computational platform service, including 48 cores server which provides virtual machine running platform, 96 cores from ALPS supercomputer used as dedicated compute service and 8 cores PC cluster to assist the use the VM server and the backend computational server.
- 30 image processing (IP) executables of the F4K workflow system are preliminarily tested.
- Customized-based VM Portal, web-based interface to access all computational resources, is developed A process execution platform based on virtual machines is developed to assist F4K developers to migrate and integrate their codes from local development sites to the centralized production site.

References

[1] G. Nadarajan. "Semantics and Planning Based Workflow Composition and Execution for Video Processing". Ph.D. Thesis, CISA, School of Informatics, University of Edinburgh, August 2010.

[2] Eckerson, Wayne W. "Three Tier Client/Server Architecture: Achieving Scalability, Performance, and Efficiency in Client Server Applications." Open Information Systems 10, 1 (January 1995): 3(20)

[3] T.-L. Chung, W.-Y. Chang, W.-F. Tsai, F.-P. Lin, E. Strandell, L.-C. Ku, J.-G. Lee, J.-Y. Chang, T.-H. Lee, J.-H. Wu, S.-C. Lin, M. Chen, Y.-H. Lee, K.-C. Chang, Y.-F. Wang, “Cyberinfrastructure for flood mitigation in Taiwan”, Proceedings of the ICE - Water Management, Volume 163, Issue 1,pages 3 – 11, 2010.

[4] Whey-Fone Tsai, Weicheng Huang, Fang-Pang Lin, Bonita Hung, Yao-Tsung Wang, Steven Shiau, Shyi-Ching Lin, Chang-Huain Hsieh, His-En Yu, Li-Lun Pan and Chien-Lin Huang “The Human-Centered Cyberinfratructure for Scientific and Engineering Grid Applications”, J. of the Chinese Institute of Engineers, Vol. 31, No.7, pp. 1127-1139, 2008.

[5] Jyh-Horng Wu, Te-lin Chung, Fang-Pang Lin, Whey-Fone Tsai, "A scalable middleware for multimedia streaming", HPC Asia 2009, Kaohsiung, Taiwan.

[6] Hsiu-Mei Chou, Yi-Haur Shiau, Shi-Wei Lo, Sun-In Lin, Fang-Pang Lin, Chia-Chen Kuo, Chuan-Lin Lai “A Real-Time Ecological Observation Video Streaming System Based on Grid Architecture” HPC Asia 2009, Kaohsiung, Taiwan.

[7] Nai-cheng LIN, Tung-Yung Fan, Fang-Pang Lin, Kwang-Tsao Shao, Tzong-Hwa Sheen, “Monitoring of of Coral Reefs at the Intake Inlet and Outlet Bay of the 3rd Nuclear Power Plant in southern Taiwan”, Annual Meeting of The Fisheries Society of Taiwan, 19-20, December, 2009.