UNIVERSITY OF TARTU
FACULTY OF MATHEMATICS AND COMPUTER SCIENCE
Institute of Computer Science

Andres Traks

Visual IT-infrastructure Management

Bachelor's thesis

Supervisor: Egon Elbre
Co-supervisor: Ilja Livenson

Author: ..... "...." May 2013
Supervisor: ..... "...." May 2013
Co-supervisor: ..... "...." May 2013
Professor: ..... "...." May 2013

TARTU, 2013

Table of Contents

1 Introduction

Widespread availability of powerful computer hardware has enabled the adoption of large-scale IT-infrastructure. Scientific research that involves running complex simulations is carried out using hundreds or even thousands of computers working in parallel. Websites that serve many people in different geographical regions also require a large number of server machines to distribute their load. The load may change over time, for example, the load for a website typically drops at nighttime. Server virtualization technology allows a server instance to be seamlessly relocated to where resources are needed. Virtualization helps to prevent under-utilized web servers from wasting electricity and allows machines to be repurposed on the fly. This also implies that web servers are now contained in virtual machines (VMs) that are deployed in physical machines.

Having this many virtual and physical machines also poses the question of how to manage them efficiently. There are many solutions for managing machines in private networks from a single location. They all implement similar features: showing the list of machines on the network, starting, stopping, editing and migrating VMs, managing user accounts, etc.

With web browsers becoming well-standardized and able to provide a more visual user experience with technologies like HTML5 and CSS 3, it makes sense to investigate new ways in which information can be presented to the user. A more visual approach to infrastructure management can help to make the user interface more intuitive and more efficient to work with.

1.1 Structure

This thesis analyses the requirements for the graphical user interface (GUI) of IT-infrastructure management software and applies the analysis to the OpenNode server virtualization project. The work is aimed at system administrators that are evaluating OpenNode or other similar software from the perspective of how visual elements can help with management tasks. The resulting practical work is a component for OpenNode called VM-map. The tradeoffs of implementing system management tasks visually in a GUI and non-visually on the command line are also discussed. Basic knowledge of computer networks, hardware and server virtualization is assumed.

Chapter 2 introduces the concepts of infrastructure management and the problems that arise in user interface design. Chapter 3 describes specific requirements for the GUI and how to use visual elements to improve the user experience. Chapter 4 looks at how existing infrastructure management solutions have approached the given problems. Chapter 5 explains the practical work done in the OpenNode project. Chapter 6 discusses possible future extensions to OpenNode that weren't covered by the practical work herein.

2 Background

2.1 Infrastructure management

An increasingly important aspect of IT systems is infrastructure management. IT-infrastructure is comprised of hardware, such as physical computers, virtual machines and the connections between them, as well as the software for controlling the operation of the hardware. The infrastructure is designed to provide its users resources required for computation tasks. Resources include processing power (CPU time), volatile memory (RAM), persistent storage (hard disk space) and networking capabilities. As the demand for computer processing resources increases, so does the scale and complexity of IT-infrastructure. Therefore it is important for system administrators to have the appropriate tools to manage resources efficiently.

Management tasks include allocating resources for particular jobs, freeing resources when jobs are done, starting and stopping processes, starting and stopping virtual machines, monitoring resources and their usage, reviewing log messages and errors, grouping resources into logical units and handling the users of the IT system.

2.1.1 Remote management

IT-infrastructure is typically a distributed network of computer systems. The network can be spread into separate geographical as well as logical areas. A common requirement for such systems is the ability to manage the entire network from a single location, possibly from any location where Internet access is available. To satisfy this requirement, the nodes that the network consists of are typically connected over IP. Each node represents a computer with its resources. Infrastructure management software typically relies on the connectivity with the hardware nodes for propagating system administrator's requests.

2.1.2 Areas of application

Infrastructure management is an essential part of most large-scale IT systems. It has applications in areas where a large number of machines is required for operation. For example: cloud services, grid computing, websites, enterprise systems.

2.2 Infrastructure visualization

Developments in computer graphics processing have enabled information to be presented in a more visual way. For example, moving from the Command Line Interface (CLI) to the Graphical User Interface (GUI) means that people can interact with icons representing real world entities instead of issuing textual commands. Most system administrators rely on the CLI or use it as a fail-safe solution for when GUI-based management software doesn't work. The CLI is seen as faster, more reliable, more trustworthy and more established in the world of system administration [1]. It is important to note these perceptions when designing GUI-s for such systems. Trust for a GUI may be gained by increasing the amount of information available on the screen for administrators to verify the work that they do [2].

An important characteristic of infrastructure visualization is the ability to represent entities in the IT system as graphical symbols. These symbols can be icons that are shaped similarly to the entities that they represent. For instance, a computer can be shown as an icon of a computer. Icons can be accompanied by text to further differentiate two entities with the same icon or the same type.

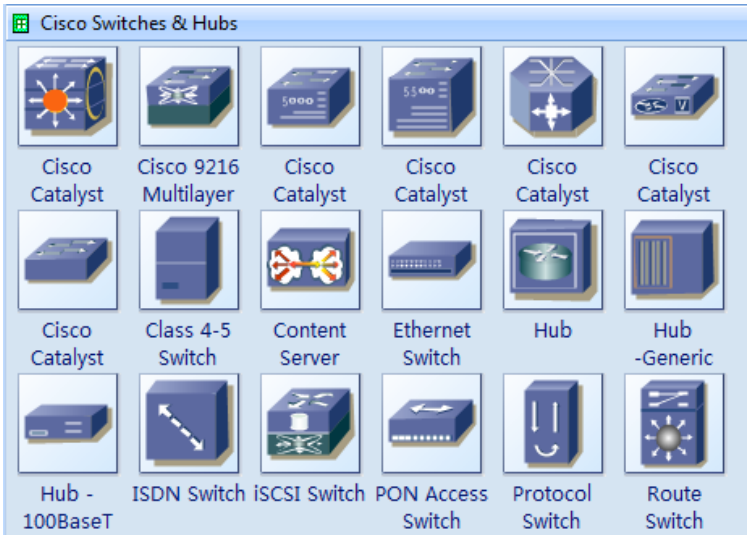

Figure 1. Various icons representing entities in a network topology [3].

In order to represent networks, the icons can be connected by lines to form a graph just as real entities are connected by cables to form a network.

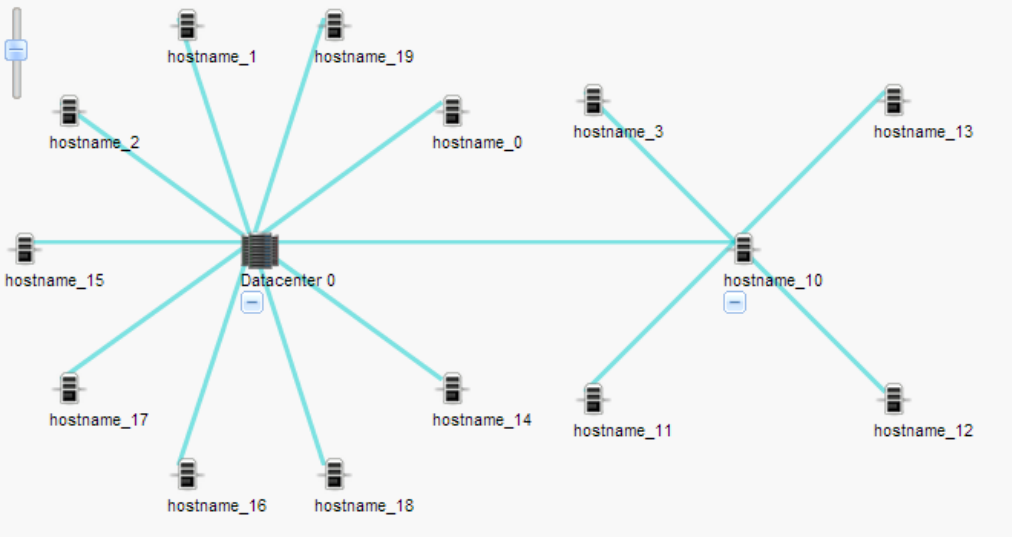

Figure 2. Network topology as displayed in an experimental version of OpenNode.

Icons can also help to visualize and distinguish entities in lists.

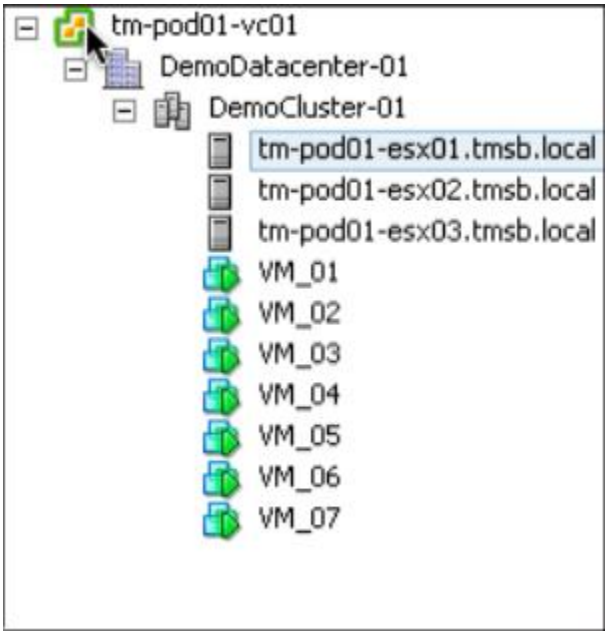

Figure 3. VMware vSphere test lab setup with 3 datastores and 7 VM-s [4].

3 Infrastructure management requirements

Infrastructure management software should have provisions for the following activities: monitoring, VM management, user management, lifecycle management. Additionally, scalability, security and fault tolerance need to be considered.

3.1 Monitoring

Monitoring is the process of gathering performance and resource usage data from a running system. This data can be provided via log files or by operating system APIs and can include both software and hardware metrics. Hardware metrics include the temperature of system components (CPU, HDD), network connectivity, power states. System administrators are typically interested in the following metrics: CPU usage, memory usage, hard disk usage and network usage. Administrators may want to define custom metrics to be monitored. For example, a website administrator might be interested in the combined disk usage of three storage devices. The metric could then be specified as “[d1] + [d2] + [d3]”. A common protocol for gathering metrics over the network is the Simple Network Management Protocol (SNMP).

Monitoring is important, because it is used for detecting system failures and warnings. Monitoring information is used for diagnostic purposes during troubleshooting. In visual monitoring tools, abnormal system behavior should be highlighted in order to catch attention.

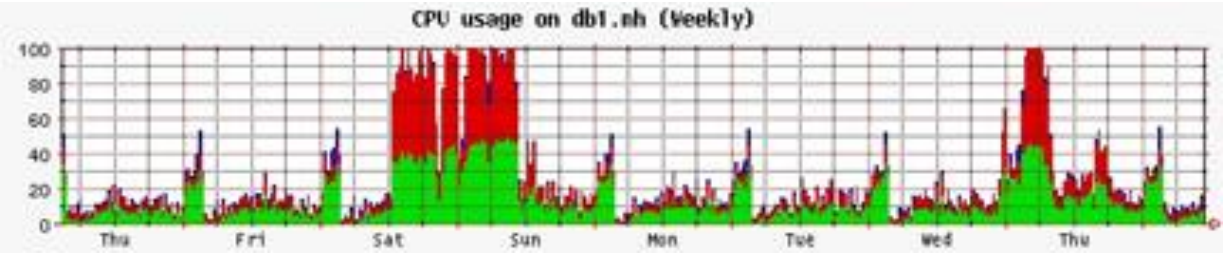

Figure 4. BBC Blogs Network CPU usage. The red spikes indicate spam attacks [5].

Activities related to monitoring:

- View current value of metric
- View history of metric
- Highlight abnormal values
- Notify of errors and warnings
- Filtering of metrics

3.2 VM management

Virtual machines require management features similar to those of physical machines. This includes monitoring of resource usage and performance. VMs, however, are deployed in containers that can be visualized as parents of the VMs. The containers usually support the following functions:

- Create VM
- Start VM
- Stop VM
- Edit VM configuration
- Migrate VM
- Destroy VM

VM configuration includes the IP address, hostname, nameservers, allocated memory and hard disk space, number of assigned CPU cores and whether to start the VM on boot.

VMs have to indicate their current status, which can be:

- Running
- Stopped
- Paused
- Transitioning

3.2.1 Templates

VM templates are VM images that are used to create VMs with a predefined configuration. Management software must provide means to select a template upon VM creation.

3.3 User management

Users of the infrastructure management software must have accounts for secure authentication and authorization. Users are assignable to groups. Possible rights that can be assigned on a per-user or per-group basis:

- Can manage users (add, block, remove)
- Can view infrastructure resources
- Can edit a resource
- Can delete a resource
- Can start/stop VMs

In addition to groups and per-user rights, users also have the following attributes that should be modifiable in the UI:

- Name
- Password (or data for some other type of authentication)
- Time zone
- Other required info (user description, contact info, timezone)

3.4 Security

Remote management presents an inherent security risk, because access to the system is granted to a user external to the system. The connection between the user and the system must be secured, meaning confidentiality and authenticity must be guaranteed. Text console based management can be done over Secure Shell (SSH). For web browser based management, Transport Layer Security (TLS) is common.

3.5 Visualizing systems of different scales

Private clouds and data centers are built for various purposes and in various scales. A user interface must therefore be able to handle both small and large numbers of infrastructure resources. As the number of resources increases, it becomes more difficult to visualize them. For example, a computer screen can only show a limited number of virtual machines at once. There are a number of ways to deal with this problem.

3.5.1 Filtering

A UI can apply a filter to a list of resources to show only those resources that the user is interested in. For example, by entering a domain name to a text field, the UI shows a list of VMs that have host names containing that domain name. More complex filtering can be achieved with regular expressions or similar parsing techniques.

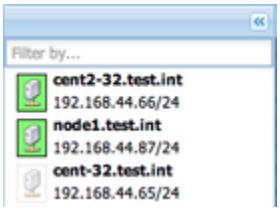

Figure 5. Filter bar in ONC.

3.5.2 Tagging

One way to differentiate resources is to enable tags to be assigned to them. For example, one VM can have the tags “linux” and “development” and another can have “linux” and “production”. This allows for a highly customized way of grouping or filtering VMs as the tags can be arbitrary.

In a graphical user interface, tags should also be visually distinguishable. This can be achieved by showing icons corresponding to tags together with the VM. The icon can be a solid color block, a more intricate pattern or an image. The benefit of a solid color icon is that it is easy to create programmatically. However, if there is a large number of tags, then colors are more difficult to tell apart than icons with patterns. Color perception is further reduced for users with color blindness. Patterns can also be generated programmatically, although it is more difficult. Icons with images are potentially the most distinguishable, but a set of predefined images may not suit all users and having users create their own icons is cumbersome.

3.5.3 Level of detail

To make good use of screen space, there must be an optimal balance between the number of resources shown and the amount of information displayed for each resource. Selecting a higher level of detail means displaying more details for fewer resources and vice versa. This becomes more important with a large number of resources.

3.6 Fault handling

In general, fault handling is the process of detecting errors in the IT-system and resolving these errors in response. Errors that may occur include:

- unexpected server downtime
- abnormal values of system parameters (high CPU usage, etc.)
- network connectivity errors
- errors and exceptions in software routines
- errors reported by other system components (via SNMP or log files)

To assist users in fault handling, the UI should indicate errors in a way that attracts attention. Warm colors such as red and yellow stand out from neutral colors such as blue, green, black and gray.

3.7 Considerations for mobile devices

Mobile devices such as smartphones and tablets have smaller screens, less functionality and different input mechanics compared to desktop computers. Having smaller screens means that less information can be displayed at once. Animations, which give useful visual feedback on activities, may have to be reduced in order to deal with the lower performance characteristics of these devices.

When managing IT-infrastructure on a mobile device, large operations should not be trivial to perform. For example, a simple panning gesture from a VM should not be enough to initiate a migration of that VM to another location. This is to prevent accidental gestures from having an effect that is difficult to revert.

On mobile devices, input is typically given to a touchscreen with fingers, which is a relatively imprecise input method. A stylus can be used for precise input, but it is not available on all devices. Since a touchscreen is often the only source of input, the device has to accurately differentiate between gestures such as tapping, panning and zooming without any additional input such as mouse keys or keyboard keys. There are techniques that can be used to reduce the chance of interpreting the various gestures incorrectly.

Delayed tapping (or “tap and hold”) is useful for preventing accidental taps on the touch screen. In this case, not any tap is registered immediately. Instead, the finger must be held in the same location on the screen for some time (about a second) for an action to begin. Delayed tapping is often used to bring up a context menu on a selected item.

Uncommon gestures can be used in some contexts. For example, the user can draw a circle or rectangle on the touchscreen to select multiple items.

Multi-touch enables two or more fingers to be used for gestures. For example, using two fingers to drag a VM may initiate migration while using three fingers may start a resize operation. This way, the large resize and migration operation have less chance to be initiated accidentally. Dragging with a single finger can be reserved for simple panning of the VM map.

4 Existing solutions

There are different tools available for infrastructure management. We look at VMware vSphere, CloudKick, Citrix XenServer and the current OpenNode solution to see how they match the requirements (given in chapter 3) and what ideas can be incorporated into and improved upon in OpenNode VM-map (in chapter 5). These solutions were chosen for their suitability for management of private clouds or data centers, relative popularity and previous experience with them.

4.1 VMware vSphere

VMware vSphere is a proprietary virtualization solution. It provides a graphical interface and several alternative command-line interfaces such as vCLI and PowerCLI.

VMware vSphere provides graphs showing the history of resource usage over time.

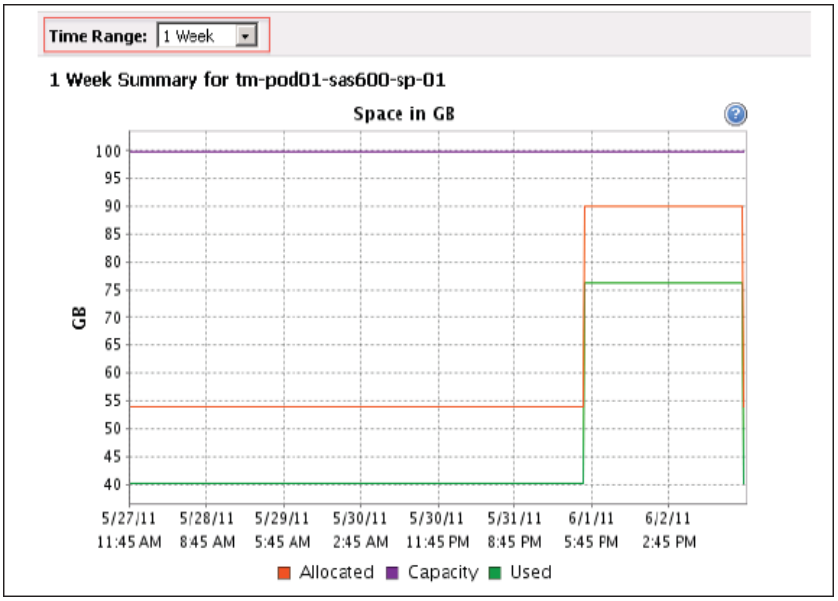

Figure 6. vSphere showing datastore space utilization history [4].

Red and yellow are used to indicate errors and warnings respectively. For example, a host failure is indicated by adding a red square with an exclamation mark to the host's icon. A warning for a host is indicated by adding a yellow triangle with an exclamation mark. Faults are displayed in several locations. In addition to adding red and yellow symbols to the host's icons, the issue is detailed in the host's summary page as well as in the host's event log. In all cases, warm colors are used to highlight the issues.

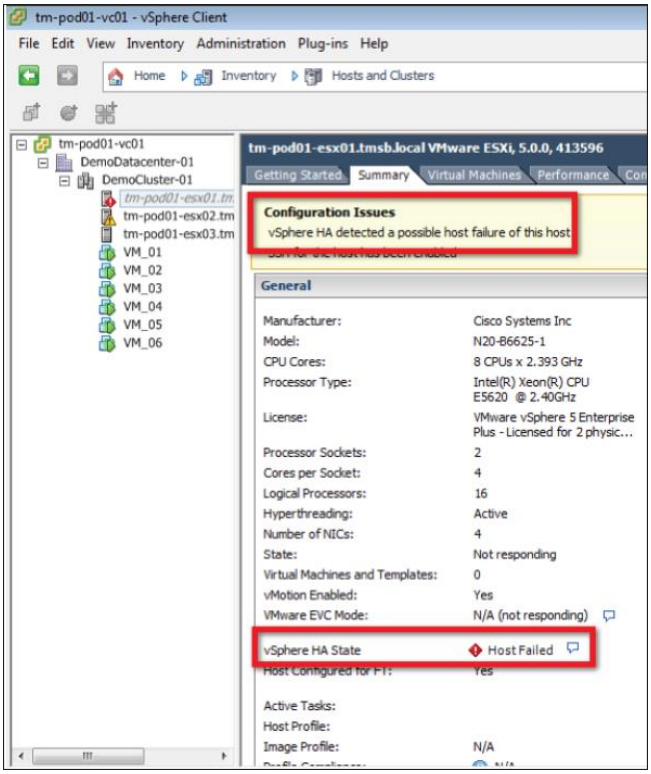

Figure 7. vSphere. Summary of a failed host [4].

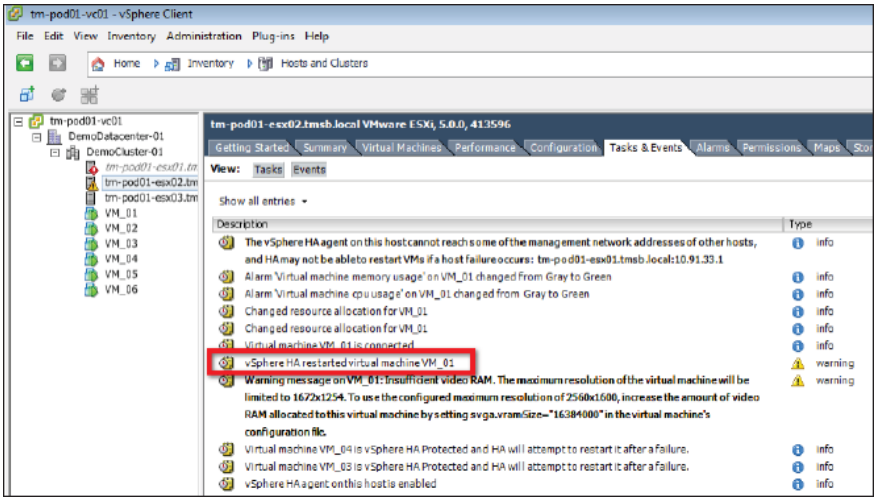

Figure 8. vSphere. Viewing log messages [4].

In contrast, PowerCLI as a command-line interface is only able to use text foreground and background colors to indicate issues.

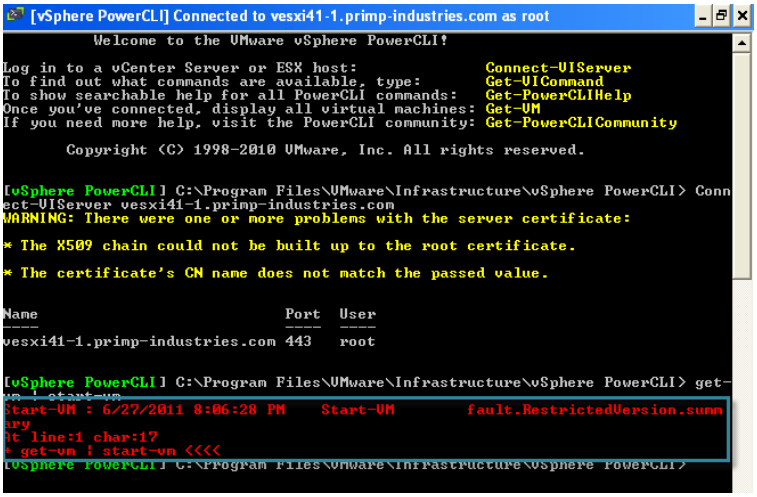

Figure 9. PowerCLI warning about server certificate and error when turning on a VM [6].

4.2 Cloudkick

Cloudkick is a proprietary cloud monitoring solution for use with Amazon EC2, SliceHost and Rackspace cloud hosting services.

On Cloudkick's dashboard, VM nodes are displayed as a list. Each list entry contains the VM's hostname, hosting provider, IP-address and textual tags. Cloudkick allows VM instances to be color coded, which is a variant of visually tagging VMs for easier identification. There are also action buttons for controlling the power state and for opening a terminal connection to the VM.

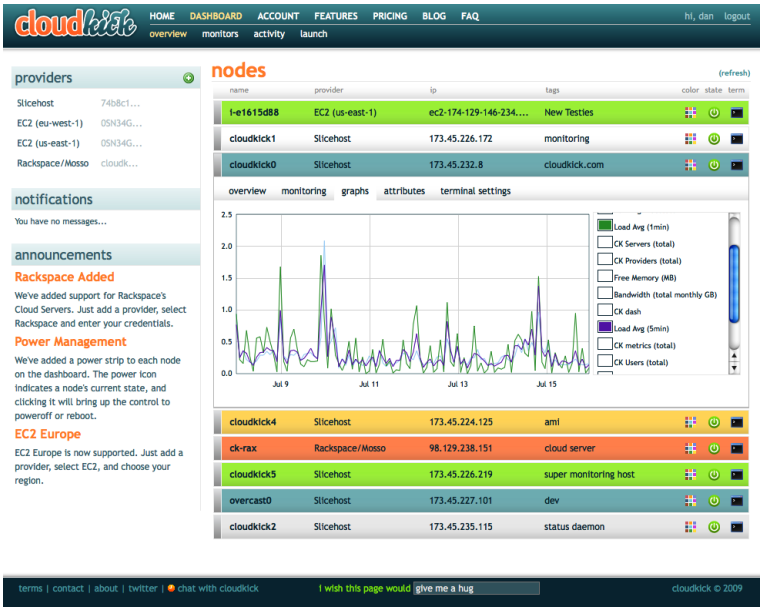

Figure 10. Cloudkick's dashboard [7].

Cloudkick can be set up to periodically check specific metrics of machines. If a metric is outside its normal range, an error or warning is displayed and highlighted in red or yellow.

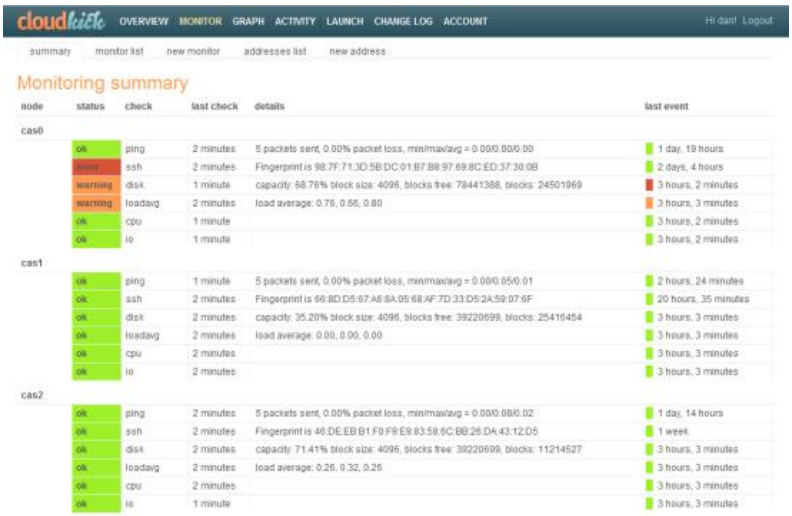

Figure 11. Clouckick monitoring summary [8].

4.3 Citrix XenServer

Citrix XenServer is a proprietary server virtualization solution. It has similar features to vSphere and CloudKick in terms of visualizing VMs and metrics.

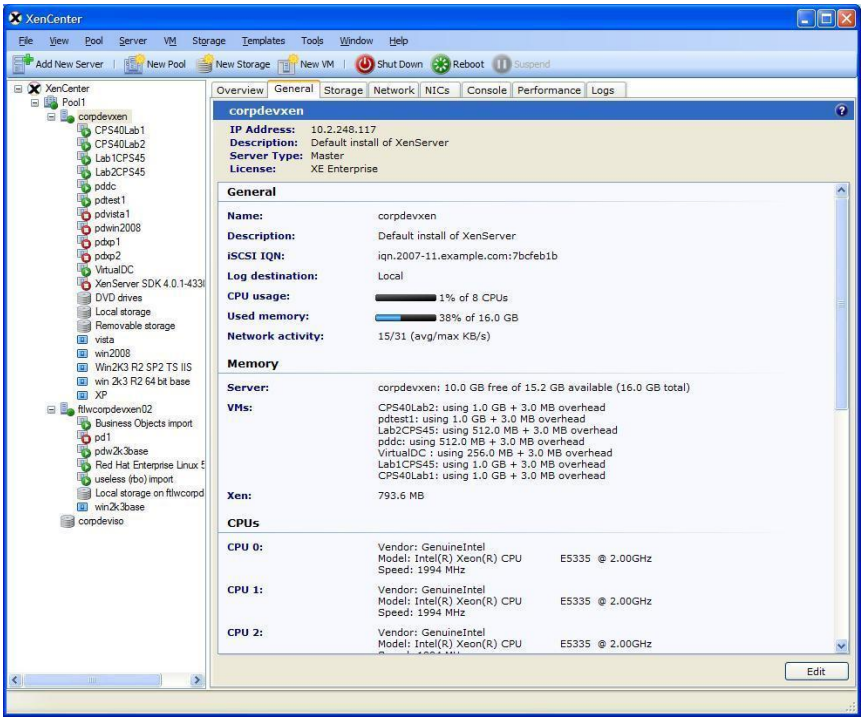

Figure 12. Citrix XenCenter management console [9].

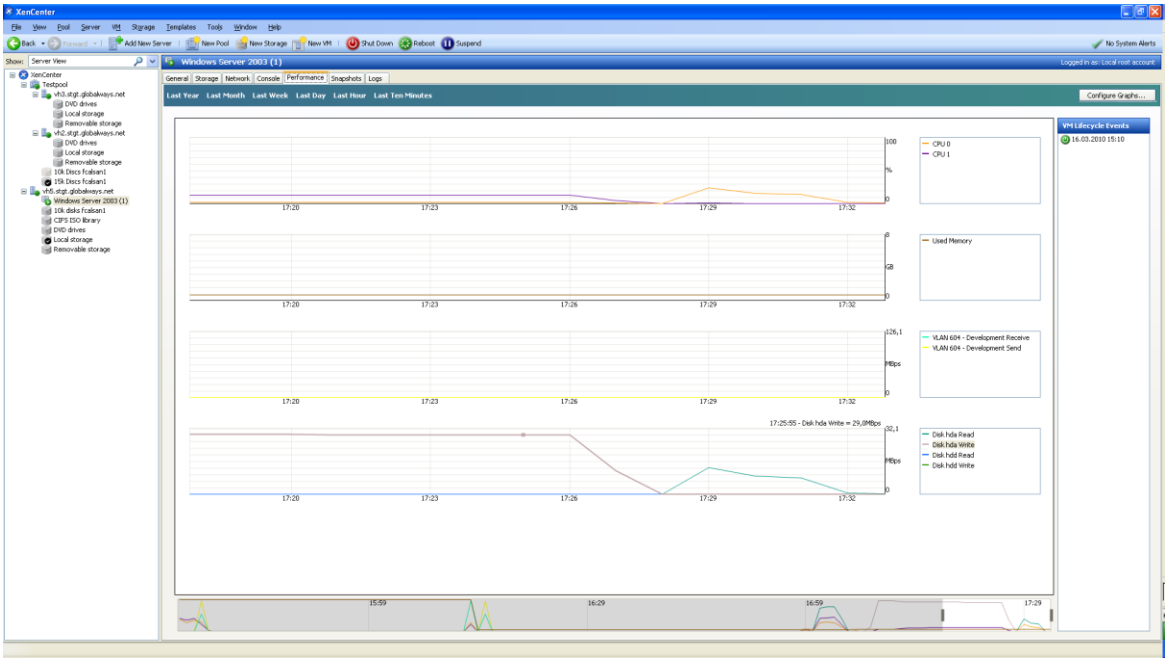

Figure 13. Citrix XenCenter performance graphs.

4.4 OpenNode Console (ONC)

OpenNode is a virtualization management solution built on open-source software [10]. It runs on the CentOS Linux distribution and makes use of virtualization software such as OpenVZ and KVM. Opennode consists of the OpenNode Management Server (OMS) and the OpenNode Management Console (ONC). There is also a Textual User Interface (TUI).

OMS is the backend that provides a Hierarchical Information System (HIS) from which to query information and initiate actions concerning the infrastructure that is being managed. OMS runs as a daemon process (omsd). Communication with OMS can be done through either a REST interface or a command-line interface (OMSH).

ONC is the frontend to OMS. It is a web application built using the ExtJS framework. It communicates with OMS through the REST interface, but also provides the OMSH interface for detailed control. ONC uses modern web technologies such as JavaScript and the Document Object Model (DOM).

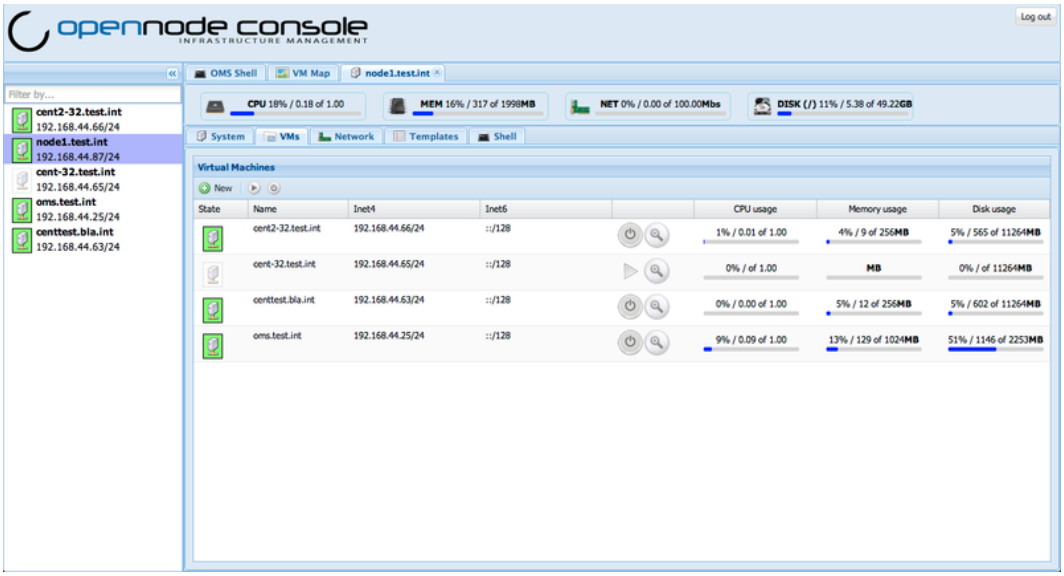

Figure 14. OpenNode Console.

5 OpenNode VM-map

5.1 Overview

VM-map is a component of the OpenNode Management Console. It provides a more visual representation of the infrastructure nodes than the standard ONC interface. VM-map was started as a result of involvement with the OpenNode project through the university, including the Software Project course. The features detailed here were implemented as part of this thesis. The code is written in JavaScript and publically hosted [11].

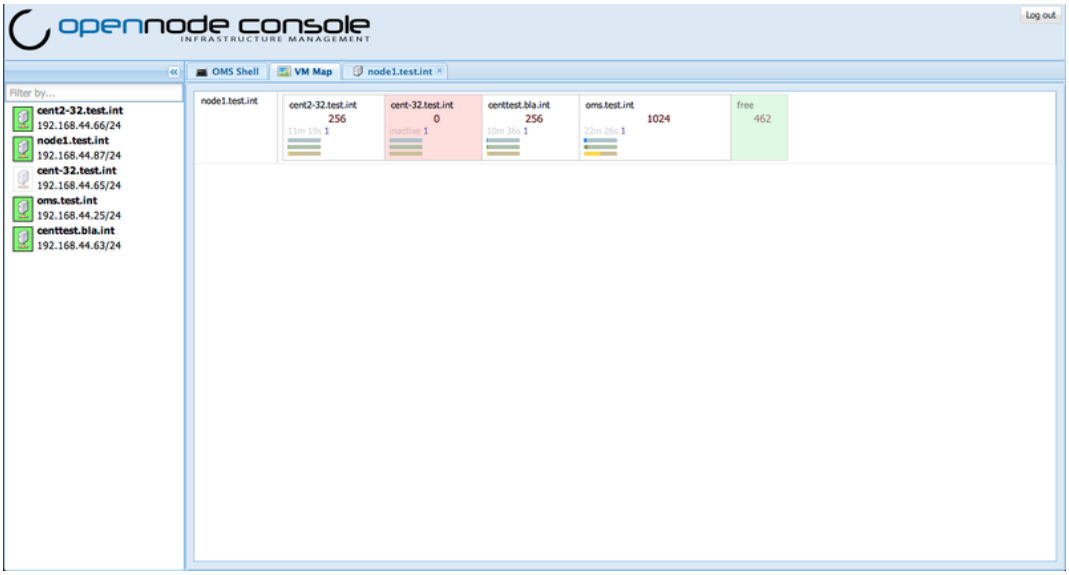

Figure 15. OpenNode Console with VM-map.

VM-map is implemented as a tab in ONC that contains a list of physical machines. A horizontal bar represents a single physical machine. The bar shows virtual machines that are contained within the physical machine. Each VM is represented as a block containing information such as the VM's hostname, allocated memory, power status, uptime and processor count. If the VM is offline, then the background of the corresponding block is red. Resource usage is displayed using smaller differently colored bars. The bars show the current CPU, memory and disk usage. A VM block is horizontally scaled roughly in proportion to the size of the memory that is allocated to the VM on the physical machine. There is an additional green block that shows the memory available for new VMs.

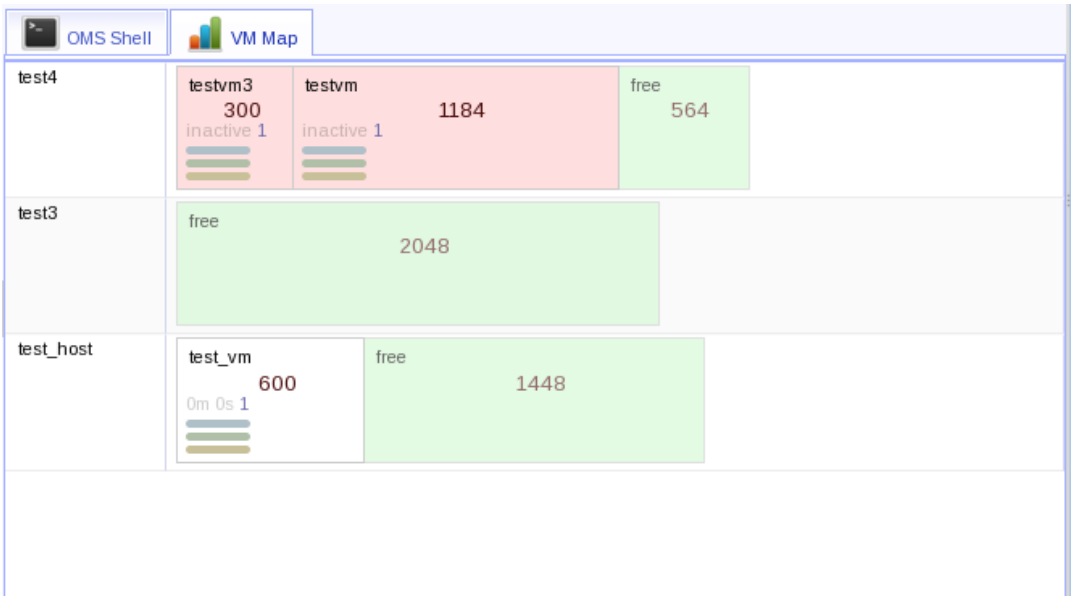

Figure 16. VM-map tab in OpenNode Console.

Figure 17. Active VM showing hostname, memory allocated, uptime, processor count and resource usage.

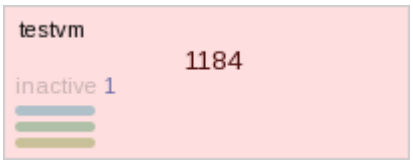

Figure 18. An inactive VM. The background of an inactive machine is red.

5.2 VM Selection

VMs can be selected using the mouse button. A selected VM is highlighted in light blue. Multiple VMs can be selected by holding down the Ctrl key. Clicking on two VMs while holding down the Shift key will select the two VMs and all VMs in between. Selecting multiple VMs enables actions to be performed on a group of VMs. These actions may include mass migration or deletion.

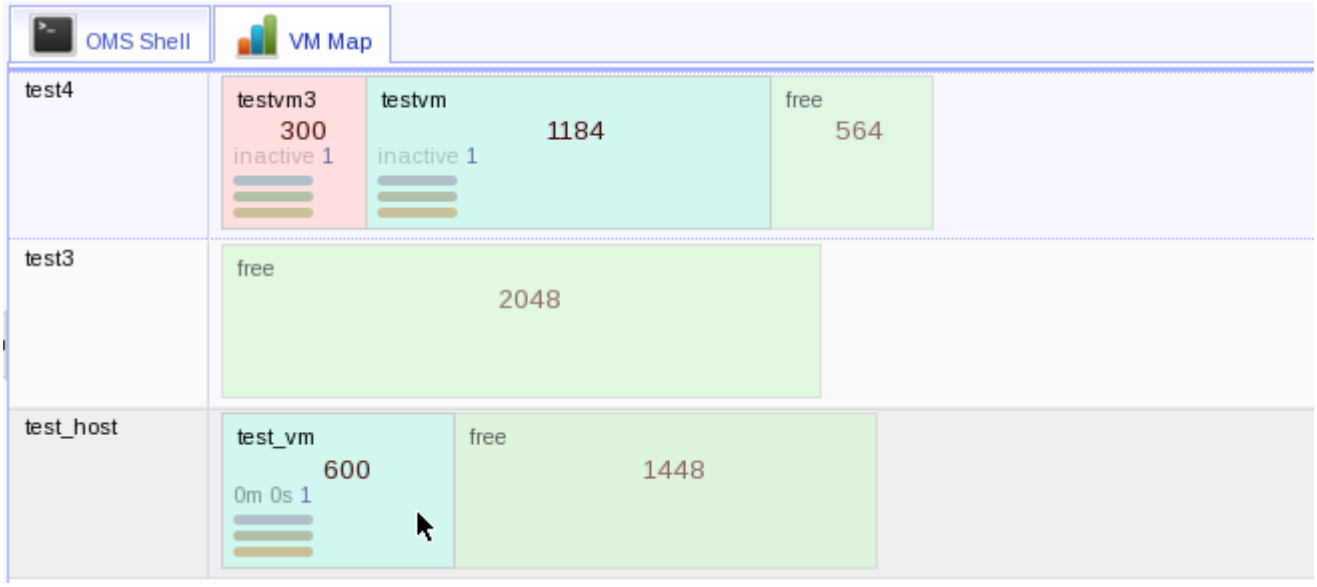

Figure 19. VM selection. The machines testvm and test_vm have been selected.

5.3 Migration

Migration of VMs is done in a special migration mode to prevent VMs from being migrated accidentally. Migration mode is enabled with a button on VM-map's menu bar.

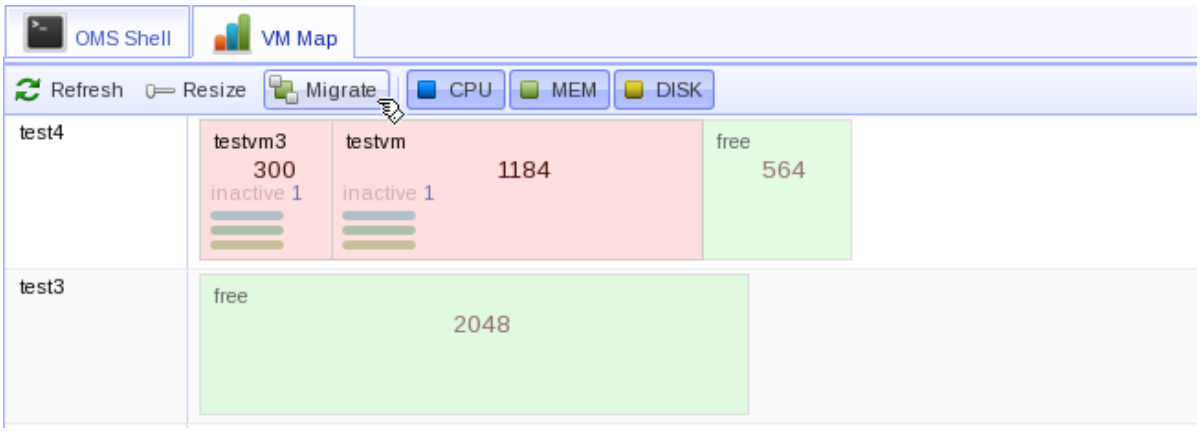

Figure 20. Entering migration mode (ONC development version).

Entering migration mode enables VM blocks to be dragged with the mouse cursor from one physical machine to another. While a VM block is being dragged, a box appears next to the cursor showing the name and other information about the VM as well as a sign indicating the status of the migration. The sign shows whether it is legal to drop the VM to the current mouse position. If the cursor is on a physical machine that meets requirements for migration, then the cursor shows a green tick. Otherwise, the cursor shows a red stop sign. The requirements for migration are that there are enough resources (memory) for the VM, the user has permission to perform the migration action and the VM isn't being migrated into its current location.

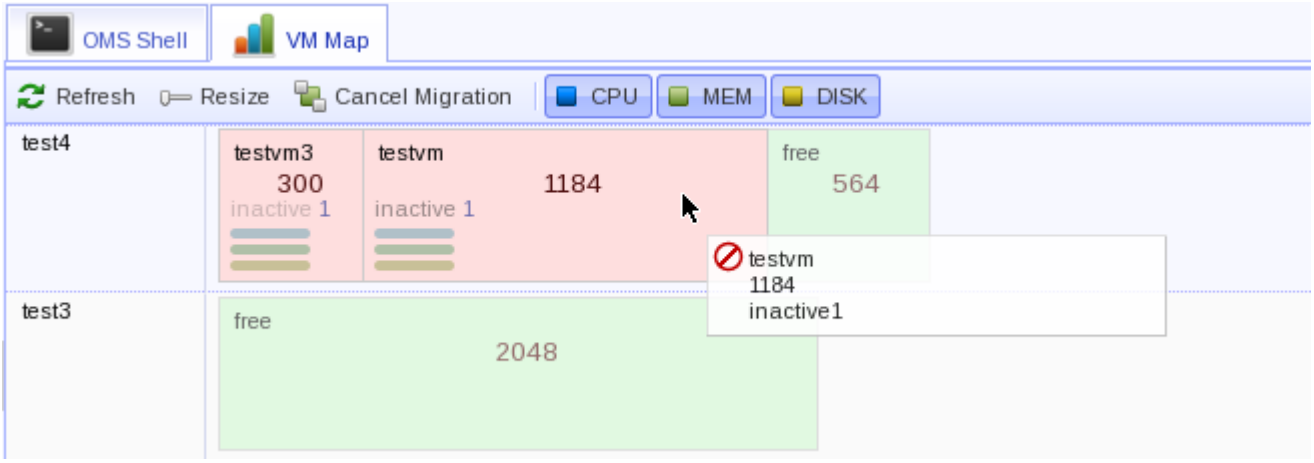

Figure 21. A VM cannot be migrated to the same physical machine, hence the red sign.

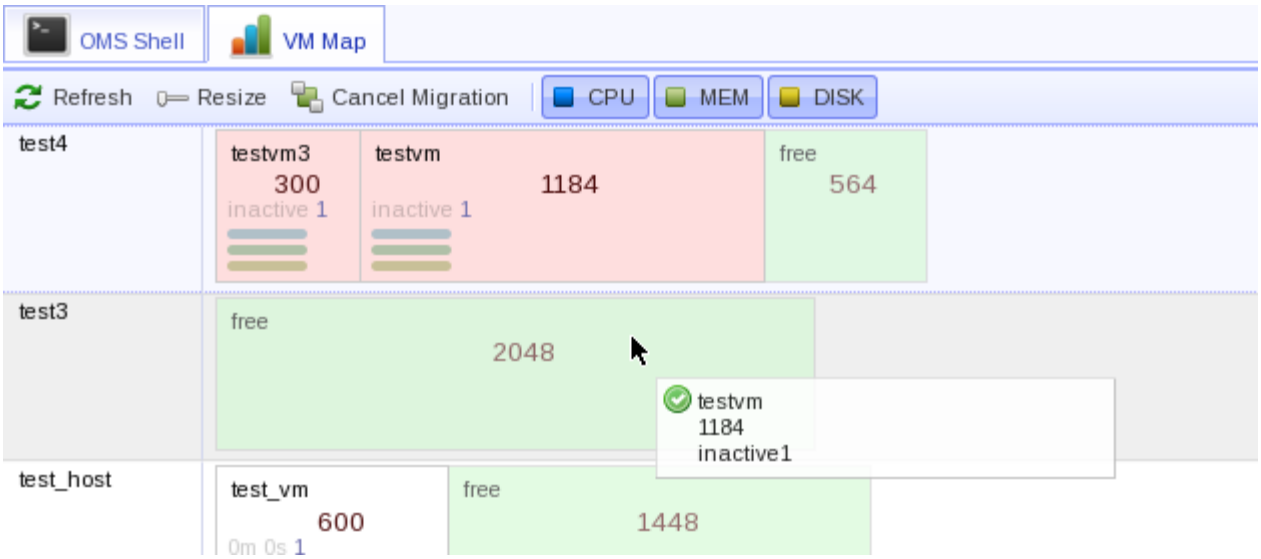

Figure 22. Test3 is a suitable target for the migration of testvm.

Once the VM block is dropped, a confirmation dialog box appears. The confirmation dialog shows the source and target machines. If the migration is confirmed, then the migration action is initiated. Confirmation is necessary, because migration is typically an expensive procedure and it should not be easy to initiate it by accident.

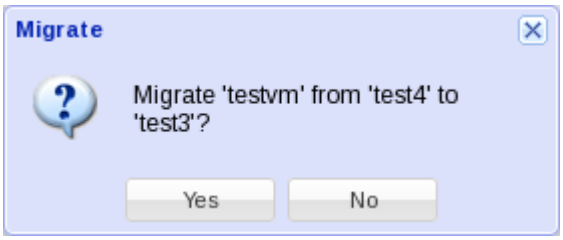

Figure 23. Migration confirmation dialog.

Alternatively, VMs can be moved freely while in migration mode, but the migration does not start immediately. Instead, the confirmation dialog appears when the user has made all of the required changes. The migrations are performed only once all migrations are confirmed and migration mode is exited. The confirmation dialog shows a list of all migration actions that are to be performed. This approach allows VMs to be migrated in bulk and to visualize the state of the machines after migration.

5.4 Resizing

Resizing of VMs is done in resizing mode. Resizing mode is enabled with a button on VM-map's menu bar.

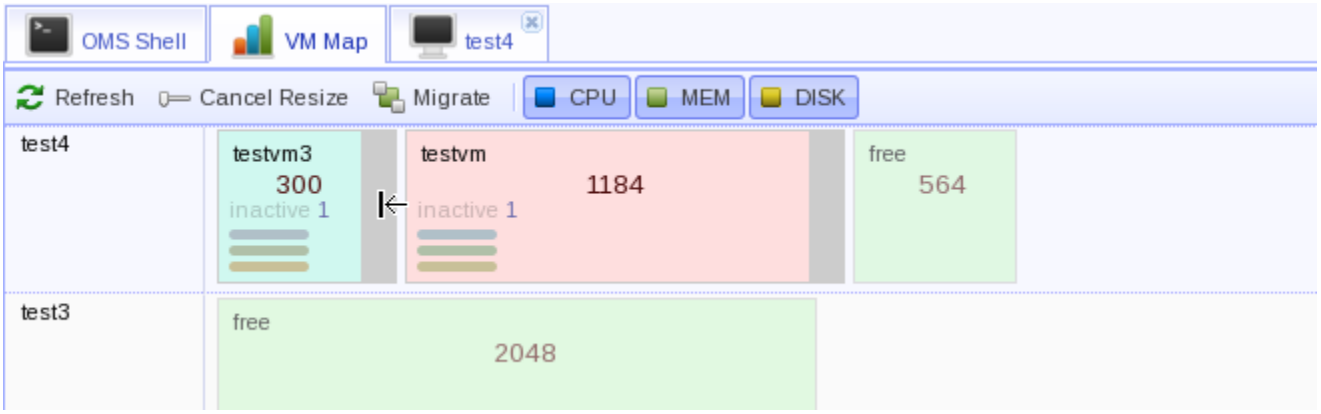

Figure 24. VM-map resize mode.

Resizing mode adds a small bar to the right of each VM block. The bar can be grabbed with the mouse button and moved left and right. Each VM block has its width proportional to how much memory it consumes. Therefore, moving the bar left decreases the memory allocated to the VM and moving the bar right increases it. When the mouse button is released, a confirmation dialog will appear showing the exact size of the new allocation. The resizing action will start once the user confirms the new size.

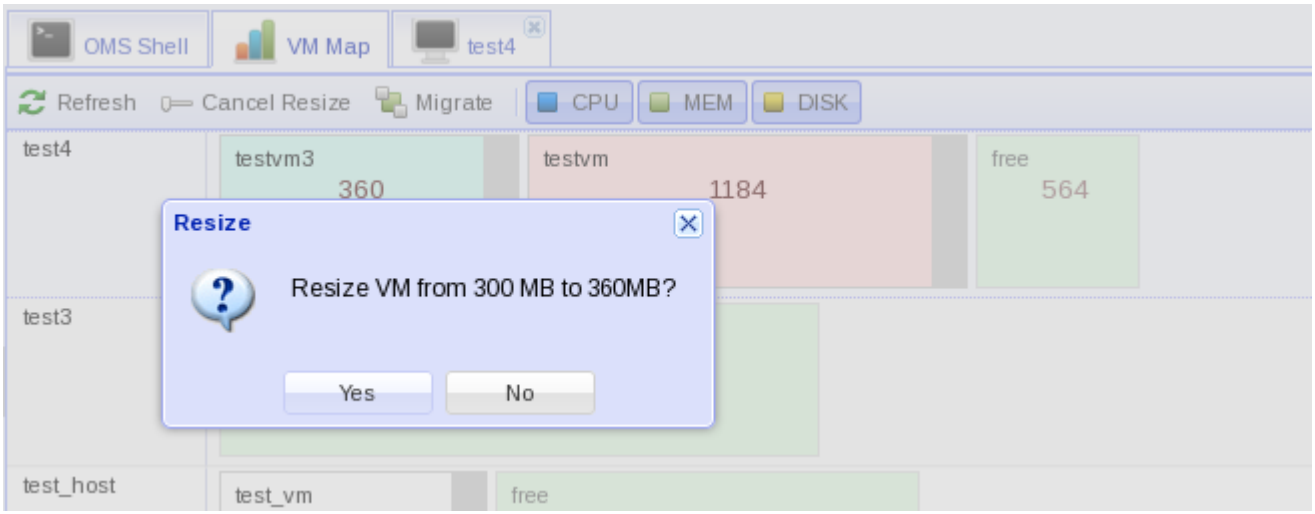

Figure 25. Resize confirmation.

5.5 Tagging

VM-map implements tagging of machines. If the machine has been tagged with a predefined text tag (e.g. “env:production”, “env:development”), then VM-map displays a small block with a solid color and the first letter of the tag (e.g. development - “D”).

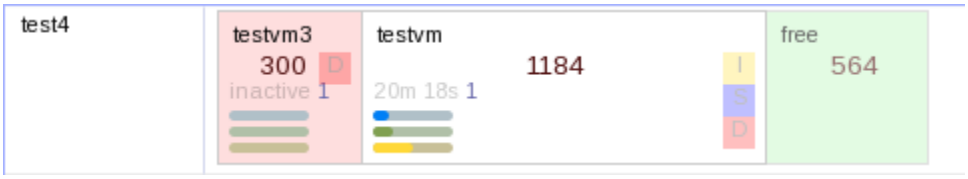

Figure 26. Tags of testvm3 (“env:development”) and testvm (“env:infrastructure, env:staging, env:development”).

Tagging can be used to filter and search for machines. If a tag is clicked on, then all machines with that tag are highlighted.

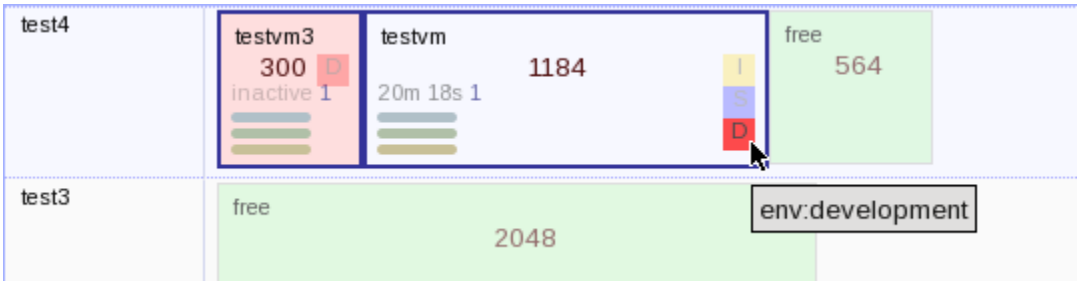

Figure 27. Testvm3 and testvm both have “env:development” and are therefore hightlighted.

6 Future work

There are more aspects of visual IT-infrastructure management that could be improved upon, but were too time-consuming to implement, had some unresolved design questions or had other outstanding issues.

6.1 Capacity planning

A feature that is missing from VM-map and other management solutions is an operating mode where changes made to machines are not stored immediately. In this mode, the changes would be visualized first and applied only after the user has verified that the configuration is acceptable. For example, this would allow to plan ahead the storage capacity of several machines and to know in advance if the amount of storage requested exceeds the storage available. This is similar to how most visual partition managers allow a hard drive to be partitioned. The partitions do not get written to disk until the user has finished specifying all of them, because a mistake may result in data loss.

6.2 Level of detail

VM-map currently displays all information about the VMs. It does not change the amount of information displayed depending on the amount of information and available screen space. With a large number of VMs, this may cause performance issues and make navigation in VM-map more cumbersome.

One solution is to have a detailed view and compact view that the user can switch between. The detailed view would take up more space, but provide more details. The compact view would be better for a general overview of the infrastructure and for quickly discovering machines with errors and warnings.

Another solution is iterative zooming where the amount of information displayed per machine changes linearly depending either on the total number of machines displayed or on the zoom level selected explicitly by the user.

A combination of compact and detailed views can be achieved by showing full details only for the machine that is currently selected. That is, when clicking on a VM, its box size is increased and more details are shown in it.

6.3 Tagging with symbols

VM-map currently allows tagging VMs with icons of solid color (see 5.5). With this it may become difficult to differentiate between a large number of tags. A solution is to automatically generate graphical symbols instead of the single-color icons. This is technically more challenging, but could be helpful for visualization.

7 Conclusion

This thesis analyzed the problems of IT-infrastructure management, specifically how to design a graphical user interface for managing virtual machines. Thanks to advancements in computer graphics technology, more visual elements can be incorporated into design. Visual elements help to bring attention to errors and warnings in the infrastructure and provide an alternative view of the machines.

We looked at existing management solutions such as VMware vSphere, CloudKick, Citrix XenServer and OpenNode. They all use visual elements to some extent, but there are aspects that can be improved upon and features to be added.

OpenNode VM-map was created as the practical part of this thesis to implement ideas for improvement. For example, VM-map visualizes VMs as boxes instead of the standard list, VMs can be tagged with graphical icons, migration can be done by dragging and dropping VMs, resizing can be done by stretching the VM box with the mouse instead of entering the size on the keyboard, etc. As of writing, VM-map hasn't been incorporated into OpenNode for a long time, but it is proving to be a useful alternative to the default view of the machines list.

In addition, there are more ideas for future work not yet implemented in VM-map such as provisions for capacity planning, level of detail and tagging with symbols. These may improve management workflow even more.

As IT-infrastructure continues to expand, the need for visual management tools increases. Therefore it is important to keep investigating new ways in which these tools can be made more intuitive and usable.

Visuaalne IT-infrastruktuuri haldamine

bakalaureusetöö
Andres Traks
Kokkuvõte

IT-infrastruktuur hõlmab IT-süsteemi ressursse: füüsilised arvutid, virtuaalmasinad ja nende vahelised ühendused, samuti arvutites olev tarkvara, protsessorid, mälu, kettaruum ja võrguliidesed. IT-süsteemi haldamiseks on vajalik tarkvara, mis võimaldaks ressursside olekut jälgida ning läbi viia ressurssidega seotud tegevusi. Jälgida saab näiteks mälu kasutust, kettaruumi kasutust, protsessori koormust või süsteemi veateateid. Ressurssidega seotud tegevused on näiteks mälu või kettaruumi jaotamine, virtuaalmasina käivitamine või peatamine, virtuaalmasina üleviimine ühest asukohast teise jms. Samuti on oluline haldamise võimalus kaugarvutist üle võrgu. Tüüpiline stsenaarium on privaatpilv teadusasutustes, kus hulk arvuteid vajavad koostöö koordineerimist.

Eeskätt keskendub töö eelkirjeldatud tarkvara graafilise kasutajaliidese (GUI) kavandamisele. Oluline on sealjuures tarkvara lihtsus, intuitiivsus ja efektiivsus. Süsteemiadministraatoritel on paljusid süsteemihalduse tegevusi harjumuspärasem teha käsurea kaudu. Seetõttu tuleb arvestada, et kõiki tegevusi ei pruugi olla mõtet GUI-sse sisse ehitada. Uuringutes on välja toodud, et GUI-d peetakse süsteemihalduse valdkonnas vähem usaldusväärseks ning vähem efektiivseks [12]. Küll aga pakub GUI võimaluse IT-süsteemi paremaks visualiseerimiseks. Näiteks saab kasutatud kettaruumi kuvada mitte ainult andmeühikutes tekstina, vaid ka graafilise ribana selliselt, et suuremale andmemahule vastab suurem osa ribast. Nii on info paremini hoomatav.

Üheks töö osaks on olemasolevate lahenduste uurimine. Virtuaalmasinate halduslahendusi pakuvad näiteks VMware, Citrix, Cloudkick, OpenNode jt. Üks juhendajatest, Ilja Livenson, on OpenNode'i arendaja. Lisaks funktsionaalsetele nõuetele võiksid lahendused vastata ka nõuetele, mis puudutavad turvalisust, skaleeritavust, tõrkekindlust ning standardite sobivust. Standardite puhul on oluline, et need oleks avatud ja laialt levinud. Samuti peaks lahendus olema kasutatav mobiiltelefonides, ka siis, kui tegemist on veebipõhise liidesega.

Töö praktiline osa toimub OpenNode'i projektis. OpenNode on avatud lähtekoodiga tarkvara virtualiseeritud serverite haldamiseks. See koosneb kesksest haldusserverist (OpenNode Management Server) ning veebipõhisest halduskonsoolist (OpenNode Management Console). Halduskonsoolil on seni puudu olnud paindlik graafilise visualiseerimise võimalus. Selle eesmärk oleks muu hulgas kuvada füüsilisi masinaid koos nendes olevate virtuaalmasinatega, näidata masinate kohta vajadustele vastavat infot ning luua uusi võimalusi haldustegevuste läbiviimiseks. Näiteks oleks süsteemi graafilises vaates võimalik ühe hiireliigutusega tõsta virtuaalmasin ühest füüsilisest masinast teise. Loodava komponendi nimi on VM-map (VM-kaart).

References

[1] R. Rachum, "GUIs kick CLIs' asses," Ram Rachum's blog, 6 September 2011. [Online]. Available: http://blog.ram.rachum.com/post/9875143930/guis-kick-clis-asses. [Accessed 5 May 2013].

[2] N. F. Velasquez, "Sysadmins and the need for verification information," in *2nd ACM Symposium on Computer Human Interaction for Management of Information Technology (CHiMiT '08)*, San Diego, California, 2008.

[3] EdrawSoft, "Cisco Network Topology Icons," 2013. [Online]. Available: http://www.edrawsoft.com/CiscoNetworkTopologyIcons.php. [Accessed 20 April 2013].

[4] VMware, "VMware vSphere 5.0 Evaluation Guide - Volume One," 2011. [Online]. Available: https://www.vmware.com/products/datacenter-virtualization/vsphere/resources.html. [Accessed 20 April 2013].

[5] R. Hamman, "18 Months of blogs, Part One: History & Technical challenges," BBC Internet Blog, 12 November 2007. [Online]. Available: http://www.bbc.co.uk/blogs/bbcinternet/2007/11/robin_post_part_i_1.html. [Accessed 20 April 2013].

[6] W. Lam, "The Dreaded fault.RestrictedVersion.summary Error," virtuallyGhetto WMware Scripts & Resources, 27 June 2011. [Online]. Available: http://www.virtuallyghetto.com/2011/06/dreaded-faultrestrictedversionsummary.html. [Accessed 20 April 2013].

[7] M. Greer, "Cloudkick Help You Keep Your Clouds in One Place," The Rackspace Blog, 10 December 2009. [Online]. Available: http://www.rackspace.com/blog/cloudkick-helps-you-keep-your-clouds-in-one-place/. [Accessed 1 May 2013].

[8] J. Barr, "Cloudkick - Server Management and Monitoring," Amazon Web Services Blog, 26 January 2010. [Online]. Available: http://aws.typepad.com/aws/2010/01/cloudkick-server-management-and-monitoring.html. [Accessed 1 May 2013].

[9] I. Informer Technologies, "Citrix XenCenter screenshots," Citrix, 2013. [Online]. Available: http://citrix-xencenter.software.informer.com/screenshot/284741/. [Accessed 7 May 2013].

[10] OpenNode, "OpenNode - About," 2013. [Online]. Available: http://opennodecloud.com/about/. [Accessed 9 May 2013].

[11] OpenNode, "VmMapTab.js in the OpenNode repository at GitHub," 2012. [Online]. Available: https://github.com/opennode/opennode-console/blob/master/app/view/tabs/VmMapTab.js. [Accessed 9 May 2013].

[12] L. Takayama and E. Kandogan, "Trust as an underlying factor of system administrator interface choice," in *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI '06)*, Montreal, Quebec, CA, 2006.

Index of figures

Non-exclusive licence to reproduce thesis and make thesis public

I, Andres Traks (19th January 1991),

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to:

1.1. reproduce, for the purpose of preservation and making available to the public, including for addition to the DSpace digital archives until expiry of the term of validity of the copyright, and

1.2. make available to the public via the web environment of the University of Tartu, including via the DSpace digital archives until expiry of the term of validity of the copyright,

Visual IT-infrastructure management,
supervised by Egon Elbre and Ilja Livenson,

2. I am aware of the fact that the author retains these rights.

3. I certify that granting the non-exclusive licence does not infringe the intellectual property rights or rights arising from the Personal Data Protection Act.

Tartus, **13.05.2013**