

# Component-based system for management of multilevel virtualization of networking resources

Robert Boczek

Dawid Ciepliński

March 28, 2011



# Contents

<b>1</b>	<b>Introduction</b>	<b>5</b>
<b>2</b>	<b>Context</b>	<b>7</b>
2.1	QoS-aware networking . . . . .	7
2.2	Resource virtualization approaches . . . . .	7
2.3	Multilevel network virtualization . . . . .	7
2.3.1	Virtual network resources . . . . .	7
2.3.2	Fine-grained QoS control . . . . .	7
2.3.3	Virtual appliances . . . . .	7
2.3.4	„Network in a box” concept . . . . .	7
2.4	Applications and benefits of virtual infrastructures . . . . .	7
2.4.1	Testing and simulations . . . . .	7
2.4.2	Improving server-side infrastructure scalability . . . . .	7
2.4.3	Infrastructure as a service . . . . .	7
2.4.4	The role of resource virtualization in the SOA stack . . . . .	7
<b>3</b>	<b>Requirements analysis</b>	<b>9</b>
3.1	Functional requirements . . . . .	9
3.1.1	Instantiation . . . . .	9
3.1.2	Discovery . . . . .	9
3.1.3	Accounting . . . . .	9
3.2	Non-functional requirements . . . . .	9
3.3	Underlying environment characteristics . . . . .	9
3.4	General approach and problems it imposes . . . . .	9
3.4.1	Load balancing / Deployment . . . . .	9
3.4.2	Infrastructure isolation . . . . .	9
3.4.3	Broadcast domain preservation . . . . .	9
3.4.4	Constraints . . . . .	9
<b>4</b>	<b>Solaris OS as a resource virtualization environment</b>	<b>11</b>
4.1	General information . . . . .	11
4.2	Lightweight OS-level virtualization with Solaris Containers . . . . .	11
4.3	Crossbow - network virtualization technology . . . . .	11
4.4	Resource access control . . . . .	11

<b>5</b>	<b>The system architecture</b>	<b>13</b>
5.1	High-level design . . . . .	13
5.2	System components and their responsibilities . . . . .	13
5.2.1	Assigner . . . . .	13
5.2.2	Supervisor . . . . .	13
5.2.3	Worker . . . . .	13
5.3	Crossbow resources instrumentation . . . . .	13
5.4	Domain model and data flows . . . . .	13
<b>6</b>	<b>Implementation</b>	<b>15</b>
6.1	Implementation environment . . . . .	15
6.2	Domain model transformation details . . . . .	15
6.3	Low-level functions access . . . . .	15
6.4	Building and running the platform . . . . .	15
<b>7</b>	<b>Case Study</b>	<b>17</b>
7.1	Clustered GlassFish . . . . .	17
7.1.1	Scenario description . . . . .	17
7.1.2	GlassFish cluster integration . . . . .	17
7.2	Multimedia server . . . . .	17
7.2.1	Scenario description . . . . .	17
7.2.2	Resource access requirements . . . . .	17
7.2.3	Providing tunable and scalable virtual infrastructure . . . . .	17
<b>8</b>	<b>Summary</b>	<b>19</b>
8.1	Conclusions . . . . .	19
8.2	Achieved goals . . . . .	19
8.3	Further work . . . . .	19

# Chapter 1

## Introduction



# Chapter 2

## Context

### Chapter overview

#### 2.1 QoS-aware networking

#### 2.2 Resource virtualization approaches

#### 2.3 Multilevel network virtualization

##### 2.3.1 Virtual network resources

##### 2.3.2 Fine-grained QoS control

##### 2.3.3 Virtual appliances

##### 2.3.4 „Network in a box” concept

#### 2.4 Applications and benefits of virtual infrastructures

##### 2.4.1 Testing and simulations

##### 2.4.2 Improving server-side infrastructure scalability

##### 2.4.3 Infrastructure as a service

##### 2.4.4 The role of resource virtualization in the SOA stack

### Summary





# Chapter 3

## Requirements analysis

### Chapter overview

#### 3.1 Functional requirements

##### 3.1.1 Instantiation

##### 3.1.2 Discovery

##### 3.1.3 Accounting

#### 3.2 Non-functional requirements

#### 3.3 Underlying environment characteristics

#### 3.4 General approach and problems it imposes

##### 3.4.1 Load balancing / Deployment

##### 3.4.2 Infrastructure isolation

##### 3.4.3 Broadcast domain preservation

##### 3.4.4 Constraints

### Summary



## Chapter 4

# Solaris OS as a resource virtualization environment

### Chapter overview

#### 4.1 General information

#### 4.2 Lightweight OS-level virtualization with Solaris Containers

#### 4.3 Crossbow - network virtualization technology

#### 4.4 Resource access control

### Summary



# Chapter 5

## The system architecture

### Chapter overview

#### 5.1 High-level design

#### 5.2 System components and their responsibilities

##### 5.2.1 Assigner

##### 5.2.2 Supervisor

##### 5.2.3 Worker

#### 5.3 Crossbow resources instrumentation

#### 5.4 Domain model and data flows

### Summary



## Chapter 6

# Implementation

### Chapter overview

#### 6.1 Implementation environment

#### 6.2 Domain model transformation details

#### 6.3 Low-level functions access

#### 6.4 Building and running the platform

### Summary





# Chapter 7

## Case Study

### Chapter overview

#### 7.1 Clustered GlassFish

##### 7.1.1 Scenario description

##### 7.1.2 GlassFish cluster integration

#### 7.2 Multimedia server

##### 7.2.1 Scenario description

##### 7.2.2 Resource access requirements

##### 7.2.3 Providing tunable and scalable virtual infrastructure

### Summary



## Chapter 8

# Summary

### Chapter overview

#### 8.1 Conclusions

#### 8.2 Achieved goals

#### 8.3 Further work



# Bibliography

- [1] Crossbow: From Hardware Virtualized NICS To Virtualized Networks,  
<http://conferences.sigcomm.org/sigcomm/2009/workshops/visa/papers/p53.pdf>
- [2] Virtual switching in Solaris, <http://hub.opensolaris.org/bin/download/Project+crossbow/Docs/virtual-switching-in-solaris.pdf>
- [3] Oracle Solaris 11 Express Network Virtualization and Network Resource Management,  
<http://www.oracle.com/technetwork/articles/servers-storage-admin/sol11ecrossbow-186794.pdf>