

# Review of Cluster Computing for High Available Business Web Applications

Dubravko Miljković  
 Hrvatska elektroprivreda  
 Vukovarska 37, 10000 Zagreb  
[dubravko.miljkovic@hep.hr](mailto:dubravko.miljkovic@hep.hr)

**Abstract:** High availability is becoming requirement for ever increasing number of business web applications. Clustering emerges as a natural solution for delivering high availability for large number of users. In this paper clustering solutions for business web applications are reviewed. Achieving high availability and high capacity through the use of load balancing clusters at application server level and server clusters (including RAC concept) at database level is presented. Geographically dispersed clusters, metro clusters and extended distance clusters, impact of latency and corresponding synchronous and asynchronous replication solutions are described with short reference to grid computing. A benefit of using specialized hardware like blade servers and virtualization techniques on large servers and groups of servers for implementation of business clustering is presented. Final part of paper concentrates on brief description of few real world examples of clustering based on first hand experience.

**Index terms:** cluster computing, high availability, business applications, web applications, NLB, failover, geographically dispersed cluster

## I. INTRODUCTION

Today's businesses are becoming increasingly dependent of highly available web applications for large number of users, [1,2]. High availability of applications is needed for business critical applications (applications whose availability is critical to operation of business: banking, finance, internet sales, ticket reservation) but also for business application available to very large number of users (application per se is not business critical, but very large number of users make cost of downtime unacceptable). Highly available systems have set very tight criteria for permissible annual downtime (Table I).

TABLE I AVAILABILITY AND ANNUAL DOWNTIME

Availability	Annual downtime
99%	87.6 hours
99.9%	8.76 hours
99.99%	52.5 minutes
99.999%	5.25 minutes

Cost of downtime, per hour for various industrial sectors is shown in Table II. Business-critical applications that require maximum uptime are excellent candidates for clustering. Clustering can provide affordable availability and to ensure continuous access to business-critical applications and data, [3]. Web applications use three-tier architecture (Fig. 1). Three-tier is a client-server architecture in which the user interface, functional process logic, computer data storage and data access

TABLE II COST OF DOWNTIME FOR VARIOUS INDUSTRIAL SECTORS

Industrial sector	Cost of downtime per hour
Production	\$28.000
Transport	\$90.000
Sales, catalog	\$90.000
Sales, internet	\$113.000
Media, Pay Per View	\$1,100.000
Banking	\$2,500.000
Finance, credit cards processing	\$2,600.000
Brokerage	\$6,500.000

are developed and maintained as independent modules, most often on separate hardware platforms.

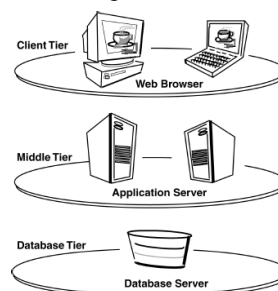


Fig. 1. Three-Tier architecture

The Three-Tier architecture has the following tiers:

- 1.) Client Tier: client is an application that runs on a personal computer and accesses by means of network a server to perform some operation.
- 2.) Middle Tier: consists of an application server that contains the bulk of the application logic. Application logic resides in a single tier and can be maintained easily at one location. The architectural design of the middle tier is optimized for server functions including access to a database.
- 3.) Database Tier: consists of one or more database servers, computers dedicated to database storage and retrieval. Upon request from application server, the database server searches the database for selected records and returns them over the network to middle tier.

## II. HIGH AVAILABILITY

Following are basic concepts related to availability, [4,5]:

### A. Reliability:

The probability that a system will perform its required functions under stated conditions for a stated period of time at a given confidence level.

$$0 \leq R(t) \leq 1 \quad (1)$$

### B. Mean time between failures (MTBF)

Mean time between failures is defined as the total functioning life of a population of an item during a specific measurement interval, divided by the total number of failures within the population during that interval. MTBF can be interpreted as the expected length of time a system will be operational between failures.

$$E(T) = \int_0^{\infty} R(t) dt \quad (2)$$

### C. Mean time to repair (MTTR)

Mean time to repair is the total corrective maintenance down time accumulated during a specific period divided by the total number of corrective maintenance actions completed during the same period

### D. Availability

Availability (inherent),  $A_I$ , is the probability that the system will be up and running correctly at a given time; this accounts for system operating time and corrective maintenance and excludes downtime associated with preventive maintenance, logistics, and administration.

$$A_I = \frac{MTBF}{MTBF + MTBR} \quad (3)$$

Operational availability,  $A_O$ , is the ratio of system uptime and total time:

$$A_O = \frac{Uptime}{Uptime + Downtime} \quad (4)$$

$$0 \leq A \leq 1 \quad (5)$$

Downtime includes both planned and unplanned downtime. Planned downtime includes scheduled maintenance activities, addition of hardware and software to improve availability and applying software and system upgrades. Unplanned downtime includes system failures, operating system crashes, application failures due to application bugs, unpredictable events such as viruses or power surges and human errors like unintentional application or system reset.

### E. Five levels of high availability

In Fig. 2 is presented stair-step solutions model to describe high availability across a range of levels, [6]. Solutions model ranges from simple to complex, from single systems to multiple systems, from single sites to multiple sites.

- Level 1 - Single system (ECC memory, Hot plug, redundant NICs)
- Level 2 - Single system (RAID Controllers and Drive Array)
- Level 3 - Cluster Fault Resilient (failover cluster)
- Level 4 - Cluster Shared Services (eg. parallel database, Real Application Cluster - RAC)
- Level 5 - Multi-Site Cluster (geographically dispersed cluster)

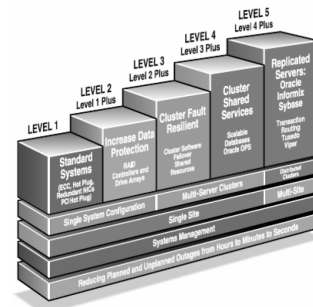


Fig. 2. Five levels of high availability

## III. HIGH CAPACITY

Beside providing high availability clusters are suitable to provide service for large number of users. Determining what hardware configuration will adequately meet the needs of your application is the process known as capacity planning. Gathering accurate performance requirements is an important part of the capacity planning process, [7]. The goal of capacity planning is to provide satisfactory service levels to users in a cost-effective manner. One must previously define maximal number of concurrent users and acceptable response time.

### A. Number of servers in cluster

When using cluster computing total number of users is generally sum of users per servers.

$$N_T = \sum_{i=0}^S N_S \quad (6)$$

$N_T$  is total number of users,  $N_S$  users per server and  $S$  number of servers available.

When sizing computer cluster for specific number of users it is good practice to add one more server in cluster than what is absolute necessity that will take load in case of another server failure:

$$S = \left\lceil \frac{N_T}{N_S} \right\rceil + 1 \quad (7)$$

With this type of sizing overall system will be able to accommodate all users even in case when one server is not functional. In spare solutions it is possible not to use this additional server, but then quick recovery of failed server must be assured by means of server self healing mechanisms that will quickly alleviate temporary capacity bottleneck or otherwise cluster servers must be run underutilized.

### B. Response time

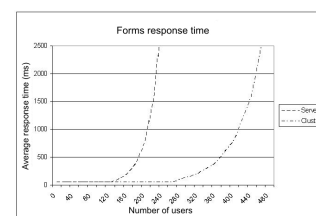


Fig. 3. Response time for user load, on single server and cluster

Response time is a performance measure defined as the round-trip delay to process a client request. An acceptable average response time may be defined as twice the average response time for one user.

### C. Behavior under load of users

Performance generally remains constant until the machine is saturated: the “hockey stick” or “knee” point on the graph. Once the saturation point is reached, performance degrades drastically with unacceptable user response time. When defining number of servers one should take into account maximal acceptable number of users per server with corresponding acceptable user response time and stay left of “hockey stick”, Fig. 3.

## IV. CLUSTER COMPUTING

A cluster is a group of independent computers that work together to run a common set of applications and provide the image of a single system to the client and application, [8]. Clustering provides high availability through elimination of single points of failure. Also provides scalability and manageability. Example of clustering applied to second and third level of three tier

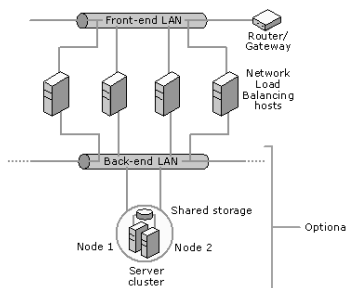


Fig. 4. Clustering applied to second and third tier

architecture is illustrated in Fig. 4. There exist two main types of clusters in business applications: load-balancing and server (failover) clusters, [8]. Load balancing clusters are suitable for middle tier where application servers reside and server clusters are suited for third tier where database servers reside.

## V. LOAD BALANCING

Load balancing is a technique to spread work between two or more computers in order to get optimal resource utilization. It supports multiple simultaneous live components and besides balancing of users load also provide way of implementing failover mechanism, ie. service continues to work despite the failure of one or more pieces of equipment. Load balancers exist in hardware and software versions.

### A. Hardware load balancers

Hardware load balancers are dedicated piece of hardware for partitioning of network traffic, commonly integrated with switch/router, Fig. 5.

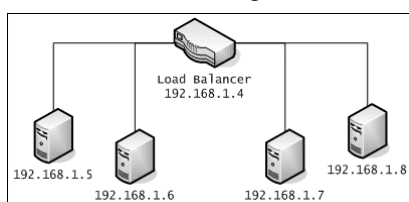


Fig. 5. Hardware load balancer

### B. Software load balancer

Software load balancer is either dedicated server with load balancing software or solution built into operating system (eg. Windows NLB).

Windows Network Load Balancing (NLB), Fig. 6, is a clustering technology offered by Microsoft as part of Windows 2000 Advanced Server and Windows Server 2003 family operating systems, [9]. To scale performance, Network Load Balancing distributes IP traffic across multiple cluster hosts. It also ensures high availability by detecting host failures and automatically redistributing traffic to the surviving hosts.

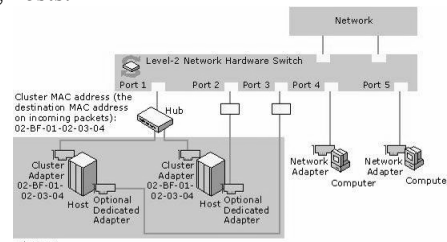


Fig. 6. Windows NLB

With load balancing it is possible to achieve:

- 1.) *Scalability* - if additional capacity is needed additional servers can simply be added to the existing cluster.
- 2.) *High availability* - NLB provides high availability by automatically detecting the failure of a server and repartitioning client traffic among the remaining servers within few seconds, providing users with continuous service.

Windows NLB is normally restricted to same physical network, for greater distances VLAN must be used.

## VI. SERVER CLUSTER

Server clusters can be realized in two main configurations:

### A. Active/passive clustering

One node in the cluster remains idle, while the other node (or nodes if running Datacenter Server) is active. If the active node fails, processing of cluster-aware applications will be switched to the passive node (failover). Once the failed node is restored, the application can revert back to the original node, so that it becomes the active node again (failback). The primary drawback with active/passive clustering is the cost associated with having a secondary system sitting idle.

Microsoft SQL Server is a relational database management system (RDBMS) produced by Microsoft and example of active/passive cluster (Fig. 7), [10].

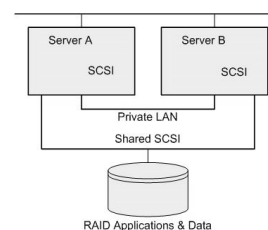


Fig. 7. Microsoft SQL Server

## B. Active/active clustering

All servers run their own workload simultaneously. Every computer in the cluster is available to do real work (is active), and each computer in the cluster is also available to recover the resources and workload of any other computer in the cluster. There is no need to have a secondary, idle server waiting for a failure. Drawback with active/active clustering is the risk of overloading the node that takes over for the failed node because it must now perform its own work plus that of the failed node.

Oracle RAC, Fig. 8, [11], is a cluster database with a shared cache architecture that overcomes the limitations of traditional shared-nothing and shared-disk approaches to provide highly scalable and available database solutions for all your business applications. Oracle RAC is example of active/active cluster. It also enables on-demand scalability by simply adding servers to cluster.

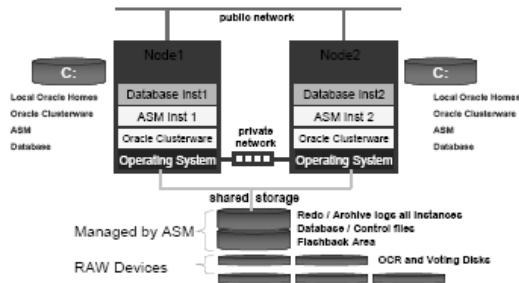


Fig. 8. Oracle Real Application Cluster (RAC)

## VII. GEOGRAPHICALLY DISPERSED CLUSTER

### A. Most common architectures

There exist three main types of disaster-tolerant geographically dispersed clusters [12, 13, 14, 15, 16]:

#### 1.) Extended distance cluster (corporate campus cluster)

- alternate nodes located in different datacenters
- RAC can run in active/active cluster
- host based replication
- Dark Fiber
- distance under 100 km

#### 2.) Metro cluster

- alternate nodes located in different parts of city or in adjacent cities (same metro area), Fig. 9.
- array based replication (storage)
- Dark Fiber
- use of arbitrators at third location
- distances under 350 km

#### 3.) Continental cluster (wan cluster)

- alternate clusters are separated by large distances connected via wide area networking (WAN)
- provides disaster recovery solution when disaster strike whole region
- replication is asynchronous and there is practically no limit on distance between clusters

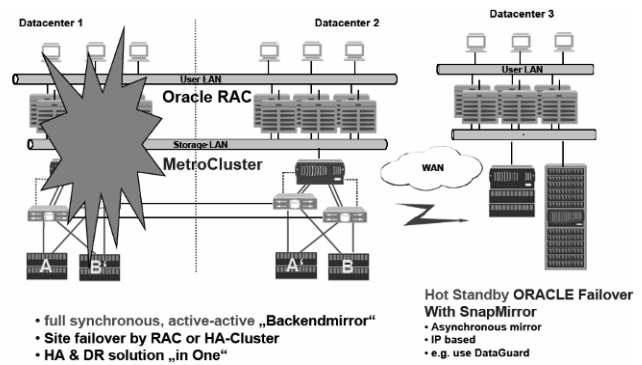


Fig. 9. Metro cluster with third datacenter (NetApp)

Distance is mainly problem for database clusters due to latency problems. At application server level with HTTP protocol, network latency is much less a problem. Clients can access application server via some kind of global load balancing (hardware solution or multiple DNS entries for same cluster name with use of subnet prioritization at client side).

### B. Three datacenter architecture

Protects against local and wide-area disasters by using both synchronous and asynchronous replication. Two datacenters form metro cluster with synchronous replication between them, and third datacenter is updated with asynchronous replication, Fig. 9.

### C. Replication

Replication is a set of technologies for copying and distributing data and database objects from one database to another and then synchronizing between databases to maintain consistency. Using replication, one can distribute data to different locations. Data Replication Mechanisms can be synchronous and asynchronous, [17]. Replication can be performed at server level, storage level and SAN level.

1.) *Synchronous Replication*: an I/O-update operation is not considered done until completion is confirmed at both the primary and mirrored sites. Synchronous replication ensures that a remote copy of the data, which is identical to the primary copy, is created at the time the primary copy is updated as long as the links between the two sites are up and running, [18].

2.) *Asynchronous Replication*: the primary write operation is disconnected from the remote write operation. The application writes the data to primary storage and continues with the next operation. There is no pause to wait for confirmation that data has been successfully written to the secondary site, as with synchronous replication. Asynchronous mode may or may not lose some committed transactions in the event of an unplanned failover to the secondary site, [19].

### D. Distribution of application code and configuration files

Beside replication of database content, distribution of application code and configuration files among application servers must also be provided, however this task is much easier to accomplish than with databases.

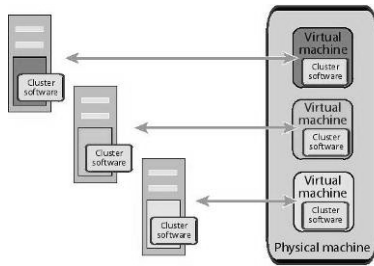


Fig. 10. Virtual clustering

### VIII. VIRTUAL CLUSTERING

Virtual clustering is accomplished by connecting Virtual Machines (VMs) into cluster, Fig. 10. Virtual machine is software layer which emulates the functionality of a certain machine or processor on a target machine. The benefit for using virtual nodes is that during hardware maintenance, all cluster nodes stay available. This can be achieved live migrating one Virtual Cluster Node from the Host System that must be maintained to some other Host System. Main environments for virtual machines are VMware ESX, [20], Windows Virtual Server 2005 and recently Oracle VM. Oracle VM is server virtualization software that fully supports both Oracle and non-Oracle applications. Most Oracle applications are certified to run on Oracle VM. VM is also another key grid technology, [21].

### IX. GRID COMPUTING

Grid computing is applying resources of many computers in a network to a single problem at the same time, [22, 23]. Instead of having each application running on its dedicated server, grid computing supports sharing the load among various systems. This offers flexibility when business processes are changing and additional server capacity is needed. Computer grids connect collections of computers which do not fully trust each other, or which are geographically dispersed. Each computer within grid has grid agent installed. These agents exchange status to enable efficient utilization of available resources. Reliability and availability is increased and serviceability is improved. Main benefits of grid computing are:

- flexibility to meet changing business needs
- high quality of service at low cost
- faster computing for better information
- investment protection and rapid ROI
- a shared infrastructure environment – ideal for Service Oriented Architecture (SOA)

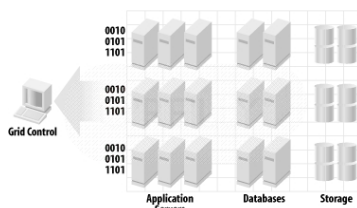


Fig. 11. Grid computing

Computer grid consists of three types of grids: application server grid, database grid and storage grid, Fig. 11. Resource allocation in grids ensures that all those that need or request resources are getting what they need, that resources are not standing idle while requests are going unserved.

### X. HARDWARE

Various hardware can be successfully used to build clusters:

#### A. Commodity servers

Conventional servers may be used as building blocks for clusters. Easy to find, but achieve low space utilization.

#### B. Blade servers

Blade servers are self-contained computer servers, designed for high density. A blade enclosure provides services such as power, cooling, networking, various interconnects and management. Blade server benefits are: reduced space requirements, reduced power consumption and improved power management, lower management cost, simplified cabling, future proofing through modularity and easier physical deployment, [24, 25].

#### C. Mid-range and high-end servers

These are large servers with numerous CPU-s (like 64 CPUs and 128 cores) and huge RAM memory (eg. 2 TB), [26]. Due to huge processing power and large memory this servers are suitable for implementation of virtual clusters.

### XI. EXAMPLES FROM REAL WORLD

#### A. Single site iAS NLB cluster and RAC database

Simple configuration with two iAS (Oracle Internet Application Server) in NLB cluster and database in Oracle RAC configuration is shown in Fig. 12. To minimize impact of possible downtime each application server is equipped with agent that monitors system and application health, manages NLB cluster start and stop, restarts failed components or even whole server, [27].

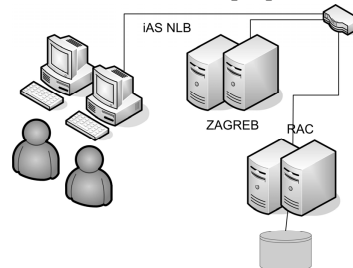


Fig. 12. iAS NLB cluster with database RAC

#### B. Geographic dispersed cluster

This is an extension of previous concept, [27, 28, 29]. Application servers and databases are set at four locations that correspond to company regional centers, Fig.13. Clusters from all locations periodically exchange short messages among themselves about cluster availability and load across

redundant network. Users access geographically dispersed cluster using single name (web address) common for all clusters. Geographical redirection takes into account location of user, current availability and load of clusters and directs users to appropriate cluster, usually to cluster of same region, but in case of its high load or failure evenly distribute users to remaining regional clusters.

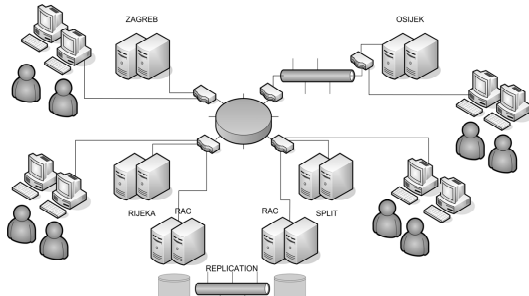


Fig. 13. Geographic dispersed cluster

There exist primary and secondary database site. Data is replicated between these two sites using synchronous replication. Failover and failback between sites is initiated manually (due to complex decision process when to make failover, etc).

### C. Blade servers

Compact and powerful solution for 1000 simultaneous users was realized with Fujitsu-Siemens Primergy BX600 S2 Advanced Blade EcoSystem, [30]. Blade chassis integrates 10 server blades, network fabric, NetApp storage and power supply with UPS, Fig. 14. Four server blades was configured as iAS NLB cluster. Two server blades constitute Oracle RAC database cluster. Remaining two server blades are spare blades.

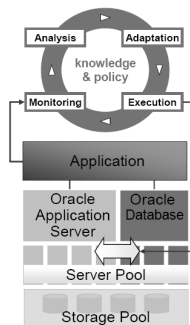


Fig. 14. NLB iAS and RAC on blade servers

Software failures are solved with self-healing mechanisms. Hardware failures are solved with spare blades that are booted with previously stored images of correct configurations.

## XII. CONCLUSION

With ever increasing reliance of business enterprises on web application for great number of users cluster computing can provide affordable solution to the problem. Choice of architecture is dependent on required availability and disaster tolerance. For smaller business single site cluster may suffice. For larger organization with business critical applications geographic dispersed

cluster is choice. Extended distance and metro clusters use synchronous replication for transparent failover and guaranteed database consistency. For increased disaster tolerance across region use of continental clusters is recommended. However, due to greater distance between clusters sometimes only asynchronous replication can be used. To achieve greater space utilization blade serves may be used. With powerful high end servers virtual hosts can also be realized that enable migration of virtual hosts.

## REFERENCES

1. Business Continuity Guideline, ASIS, 2005
2. IT Challenges in the Business Continuity Market, XOSoft; December 2003
3. Oracle9i Application Server High Availability Strategy Guide, Oracle, January 2002
4. N. Stojanović, Teorija pouzdanosti tehničkih sistema, BIGZ, Beograd, 1987
5. J. C. Conlon, W. A. Lilius and F. T. Tubessing, Test & Evaluation Of System Reliability Availability And Maintainability, A Primer, DoD, March 1982
6. Five Levels of High Availability, Compaq, 1998
7. How to Do Capacity Planning, Team Quest, 2007
8. Microsoft Windows 2000 Advanced Server Clustering Service Training Kit, Microsoft, 2001
9. Network Load Balancing Technical Overview, Microsoft, March 2000
10. I. Drakulić and V. Sviličić, SQL Server 2000 Failover Clustering, WinDays 2004
11. Oracle Real Application Clusters 11g, April 2007
12. Designing Disaster Tolerant HA Clusters Using Metrocluster and Continental clusters, HP, December 2006
13. Planning Considerations For Geographically Dispersed Clusters Using Ibm Hacmp/Xd: Hageotechnology, IBM, June 2004
14. Martin McClean, Stretching Microsoft Server Clusters with Geo-Dispersion, Microsoft TechNet, April 30, 2002
15. E. Petterson, Oracle Real Application Clusters on Extended Distance Clusters, Oracle, October 2006
16. Data Center High Availability Clusters Design Guide; CISCO, 2006
17. W. Fisher, Implementation of a Disaster Resilient Linux Cluster with Storage Subsystem Based Data Replication, Diplomarbeit, Hagenberg, Juni 2004
18. Synchronous Data Replication, Hitachi 2007
19. Asynchronous Long-distance Data Replication, Hitachi, 2007
20. VMware ESX Server, VMware, 2003
21. X. Zhang, K. Keahey, I. Foster and T. Freeman<sup>1</sup>, Virtual Cluster Workspaces for Grid Applications, ANL Tech Report ANL/MCS-P1246-0405
22. Grid Computing: Past, Present and Future An Innovation Perspective, IBM, June, 2006
23. Grid Computing with Oracle, Oracle, March 2005
24. Improving Economy of Blades with VMware, VMware, 2004
25. FlexFrame™ for Oracle Technical Overview, Fujitsu-Siemens, September 2006
26. HP Integrity Servers, Hewlett-Packard, February 2007
27. D. Miljković: Geografski distribuirani iAS NLB cluster, HROUG, Umag, 2005
28. D. Miljković: Geografski disperzirani cluster internet aplikacijskih servera za visoku raspoloživost i kapacitet, CASE, Opatija 2006
29. D. Miljković: Heterogeni geografski disperzirani cluster iAS-a 9i Rel. 1 i 9i Rel. 2 (10g), HROUG, Umag, 2006
30. D. Miljković: Primjena blade servera za poslovne web aplikacije koje zahtijevaju visoku raspoloživost i kapacitet, HROUG, Rovinj 2007