

## პროგრამული სერვერების ორგანიზაცია კორპორატიულ ქსელებში

ვოლფგანგ რეისიგი, გია სურგულაძე, დავით გულუა  
ბერლინის ჰუმბოლდტის უნივერსიტეტი,  
საქართველოს ტექნიკური უნივერსიტეტი

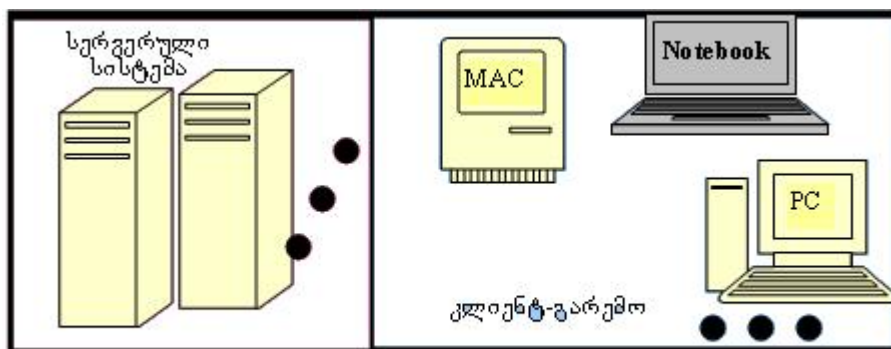
### რეზიუმე

განიხილება კორპორაციული სისტემების კლიენტ-სერვერული არქიტექტურა, რომლის სერვერ-ფარმი (ტერმინალ-სერვერი) მომხმარებელთა სამუშაო სესიების მართვასა და კლიენტთა მიმართვების ფიზიკური სერვერებზე თანაბარ განაწილებას უზრუნველყოფს, ხოლო კლასტერის (ფაილ-სერვერის) ამოცანას საერთო რესურსების (ვირტუალური IP-მისამართები და ქსელური სახელები, საერთო სერვისები და პროგრამები, განაწილებული კატალოგები და სხვა) საიმედო შენახვა და გარე მეხსიერებასთან (მაგ. SAN-დისკების მასივი) მიმართვის ორგანიზება წარმოადგენს.

### 1. შესავალი

**პროგრამული სერვერის (Application Server)** აგება და მართვა კორპორაციულ ქსელებში ერთერთ უმნიშვნელოვანეს და აქტუალურ ამოცანას წარმოადგენს. გასული საუკუნის 90-იან წლებამდე ამგვარი სერვერის განთავსება, კონფიგურირება და მართვა მხოლოდ სუპერ- და მინი კომპიუტერების ბაზაზე იყო შესაძლებელი, რადგან ზოგადად პროგრამული სერვერი ნებისმიერი სხვა ტიპის სერვერზე მეტ რესურსებს მოითხოვს. ბოლო ათწლეულში ტენდენცია შეიცვალა. პერსონალური კომპიუტერების გამოთვლით სიმძლავრეთა შეუქცევადი ზრდის შედეგად მათი ყველა ტიპის სერვერული ამოცანების გადასაწყვეტად გამოყენებაც გახდა შესაძლებელი და სადღეისოდ უკვე შეიძლება ითქვას, რომ სერვერ-ორიენტირებულმა პერსონალურმა კომპიუტერმა, აპარატურისა და ადმინისტრირების ბევრად მცირე დანახარჯების წყალობით, სერვერულ სექტორშიც სუპერ- და მინი მანქანების ხვედრითი წილი ფუნდამენტურად შეამცირა.

ამასთან უნდა აღინიშნოს ისიც, რომ დროთა განმავლობაში პერსონალური კლიენტ-მანქანების მრავალი ახალი ტიპის შექმნის მიუხედავად (რომელთაც ხშირად საკმარისი ოდენობის საკუთარი გამოთვლითი რესურსებიც გააჩნია და ქსელში სერვერებისგან დამოუკიდებელი შეუძლია იყოს), თანამედროვე პროგრამ-სერვერული სისტემების აგების ოპტიმალურ იდეოლოგიად ძველებურად ცენტრალიზებული მიდგომა მიიჩნევა, რაც სქემატურად 1-ელ ნახაზზეა ნაჩვენები.



ნახ.1 სერვერ-კლიენტის ცენტრალიზებული სისტემა

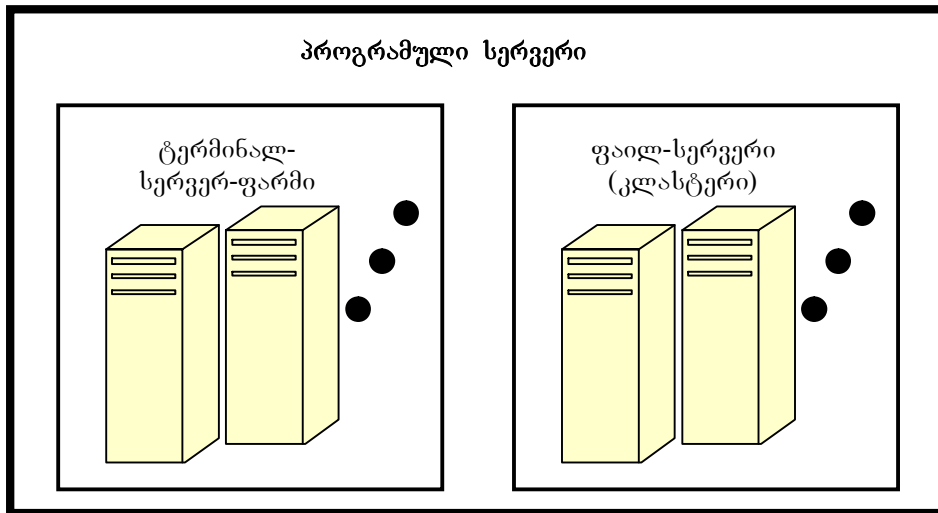
ამგვარი მიდგომა პროგრამული კომპლექსების მთლიანად სერვერის მხარეს განთავსებას გულისხმობს, ისევე როგორც სუპერ- და მინი მანქანების შემთხვევაში.

ჩამოვთვალოთ პროგრამული სერვერის ძირითადი მახასიათებლები:

- გამოყენებითი პროგრამების დიდი რაოდენობა;
- ერთდროულად შესრულებადი პროგრამების მაღალი რიცხვი;
- ცალკეული, უნიკალური მონაცემთა ბაზები გამოყენებითი პროგრამების უმრავლესობისთვის;

- ინტენსიური ქსელური ტრაფიკი.

ზემოთმოყვანილი სია პროგრამული სერვერების დიდ „პრეტენზიულობაზე“ მიუთითებს. შეიძლება ითქვას, რომ პროგრამული სერვერი ყველა სხვა სერვერზე მეტი ოდენობით მოითხოვს გამოთვლით რესურსებს. მას სჭირდება სწრაფი პროცესორებიც (უმჯობესია მულტიპროცესორული სისტემები), დიდი ოდენობით ოპერატიული და გარე მეხსიერება და მაღალი გამტარუნარიანობის ქსელური ინტერფეისები. რამდენადაც ცალკეული პერსონალური, თუნდაც სერვერ-სპეციფიკური კომპიუტერებისთვის ამგვარი მოთხოვნები ძნელი შესასრულებელია, პროგრამულ სერვერები სადღეისოდ კომპლექსური სახით აიგება (ნახ.2).



**ნახ.2. პროგრამული სერვერის ზოგადი სქემა**

## 2. ძირითადი ნაწილი

ოპტიმალური პროგრამული სერვერი სასურველია პირველი დონის ორ სერვერულ სისტემაზე იყოს დაფუძნებული: **ტერმინალ-სერვერსა** და **ფაილ-სერვერზე**. ორივე მათგანი, როგორც წესი, ერთზე მეტ ფიზიკურ სერვერს მოიცავს (მათ ლოგიკურ ერთობლიობას ტერმინალ-სერვერისათვის **სერვერ ფარმი**, ხოლო ფაილ-სერვერისთვის – **კლასტერი** ეწოდება), რაც მთლიანი სისტემის უსაფრთხოების უმთავრეს გარანტს წარმოადგენს, რადგან ორივე შემთხვევაში რომელიმე ფიზიკური სერვერის მწყობრიდან გამოსვლისას მის ამოცანებს ავტომატურად მისი „კოლეგა“ გადაიბარებს და სისტემის მუშაობის საერთო შეფერხება მხოლოდ მისი სისწრაფის შენელებით შემოიფარგლება.

სერვერ-ფარმი მომხმარებელთა საშუალო სესიების (**Client Sessions**) მართვასა და კლიენტთა მიმართების ფიზიკური სერვერებზე თანაბარ განაწილებას (**Load Balancing**) უზრუნველყოფს, ხოლო კლასტერის ამოცანას საერთო რესურსების (ვირტუალური IP-მისამართები და ქსელური სახელები, საერთო სერვისები და პროგრამები, განაწილებული კატალოგები და სხვა) საიმედო შენახვა და გარე მეხსიერებასთან (მაგ. **SAN**-დისკების მასივი) მიმართვის ორგანიზება წარმოადგენს.

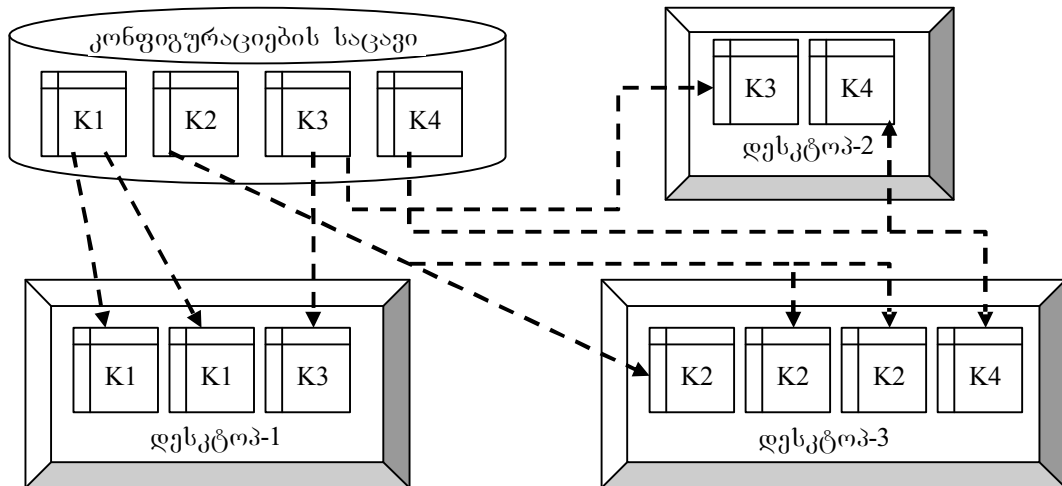
სერვერ-ფარმებზე პროგრამული სერვერის ორგანიზებისთვის საჭიროა იმ ინფრასტრუქტურის მოხაზვა, რომელშიც პროგრამული სერვერის კომპონენტები იმუშავებენ.

პროგრამული სერვერის ძირითად ამოცანათაგან შეიძლება დავასახელოთ:

- ერთიანი ინფრასტრუქტურის აგება სერვერზე განთავსებული პროგრამული პაკეტების მართვისთვის;
- პროგრამებისთვის აუცილებელი რესურსებია ცენტრალიზებული გაცემა და მართვა
- ლიცენზიების კონტროლი

პირველი ამოცანის გადასაწყვეტად განვსაზღვროთ პროგრამული სერვერის ფუნდამენტური ელემენტი, რომელსაც ფევსური კონტეინერი ანუ **კონფიგურაცია (Konfiguration)** ვუწოდოთ. კონფიგურაცია მმართველი და ინტერფეისის ელემენტების, პროგრამებზე, გამოყენებულ

რესურსებსა და დოკუმენტაციაზე მარჯვენა მხარეს ერთობლივად წარმოადგენს და ფოლდერების, გამოყენებითი პროგრამების, სისტემური პროცედურებისა და სხვა ინტერფეისული და არაინტერფეისული ელემენტებისგან აიგება. კონფიგურაცია ჩონჩხია პროგრამული პროფილების ასაგებად, სხვა სიტყვებით, კონფიგურაციების სია წარმოადგენს გამოყენებითი პროგრამების საცავს (რეპოზიტორს), სადაც კონფიგურაციების სხვადასხვა ქვესიმრავლეებით დამოუკიდებელი მომხმარებლის ინტერფეისები (დესკტოპები) აიგება (ნახაზი 3)



ნახ.3. პროგრამული სერვერის ზოგადი სქემა

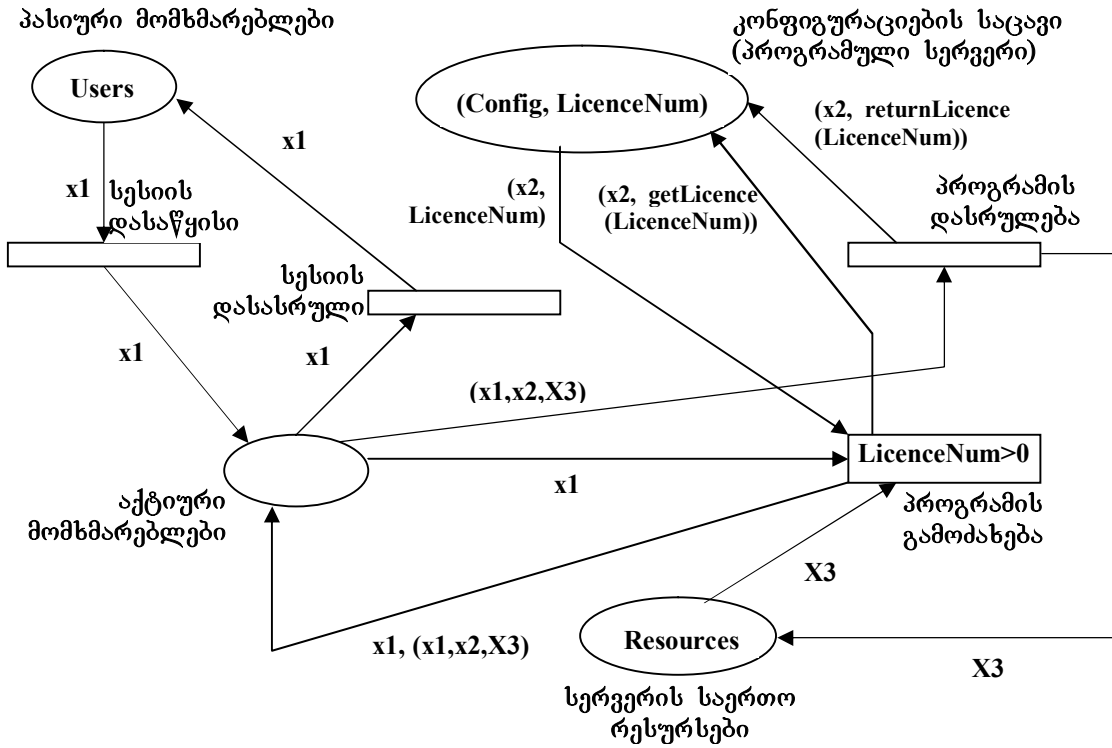
კონფიგურაციათა საცავი წარმოადგენს სიმრავლეს (დომენი)  $K = \{K_1, K_2, \dots, K_n\}$ , ხოლო პროგრამული პროფილი (დესკტოპი) არის მულტისიმრავლე (კომპლექტი) მოცემული სიმრავლის ელემენტებზე  $D_i = \{K_1, K_1, \dots, K_1, K_2, \dots, K_2, \dots\}$ , სადაც  $i=1..N$  და მასში ყოველი მოცემული კონფიგურაცია შეიძლება შედიოდეს ერთხელ, მეტჯერ ან არ შედიოდეს საერთოდ. დესკტოპების მულტისიმრავლეთა სიმრავლეს  $D = \{D_1, D_2, \dots, D_n\}$  პროგრამული სერვერი ვუწოდოთ.

მე-4 ნახაზზე წარმოდგენილია ამგვარი სერვერის მუშა მოდელის მაგალითი სისტემური პეტრის ქსელების გარემოში. სერვერ-ფარმების მართვის Windows-ორიენტირებული სისტემებიდან შეიძლება გამოვყოს Citrix Meta Frame და Windows 2003 Terminal Server, რომლებიც ნაწილობრივ პროგრამული სერვერების მართვის ინსტრუმენტებსაც მოიცავენ, თუმცა სერიოზული პროგრამული სერვერების ასაგებად სპეციალიზებული პროგრამული პროდუქტები გამოიყენება. მაგალითად, გერმანული ფირმა H+H-ს მიერ წარმოებული პროგრამული პაკეტი NetMan პროგრამულ სერვერებზე ან თუნდაც ლოკალურ კომპიუტერებზე გამოყენებითი პროგრამების სამართავად მეტად მოხერხებულია ინსტრუმენტი.

პროგრამული პაკეტი პროგრამული სერვერებისთვის ზემოთ აღწერილ ყველა აუცილებელ მოთხოვნებს აკმაყოფილებს. იგი შეიცავს მომხმარებელთა, სამუშაო ადგილების (Workstations), საერთო რესურსებისა და პროგრამული ბიბლიოთეკების მართვის ინსტრუმენტების სრულ ნაკრებს. მაგალითისთვის შეიძლება დავასახელოთ CD ROM-ებზე შენახული პროგრამული პროდუქტების სერვერზე მოქნილი გადატანის შესაძლებლობა (ვირტუალური კომპაქტ-დისკების შექმნა), რომელსაც NetMan იმავე ფირმის პროგრამულ პაკეტ Virtual CD-სთან ერთობლივად მუშაობით ახორციელებს.

### 3. დასკვნა

NetMan-ის ბოლო ვერსიების (NetMan XP, NetMan 3.0) უმნიშვნელოვანეს შენაძენად ინტერნეტ-ტექნოლოგიების ინტეგრირება უნდა მივიჩნიოთ. პროგრამის ვებ-კომპონენტები უზრუნველყოფს პირველ რიგში, პროგრამული სერვერის არქიტექტურის ინტერნეტისთვის გასაგებ ფორმატში (HTML) გადაყვანას, აგრეთვე ქმნის დამატებით ვებ-ინფრასტრუქტურას ინტერნეტ-გარემოში პროგრამული კომპლექსების ეფექტური აგებისა და მართვისთვის. ეს მეორე გარემოება NetMan-ის კორპორაციული ქსელების საზღვრებს გარეთ გატანის და პროგრამული სერვერების გლობალური (მაგალითად, კორპორაციათაშორისი) ქსელებისთვის აგების საშუალებასაც იძლევა.



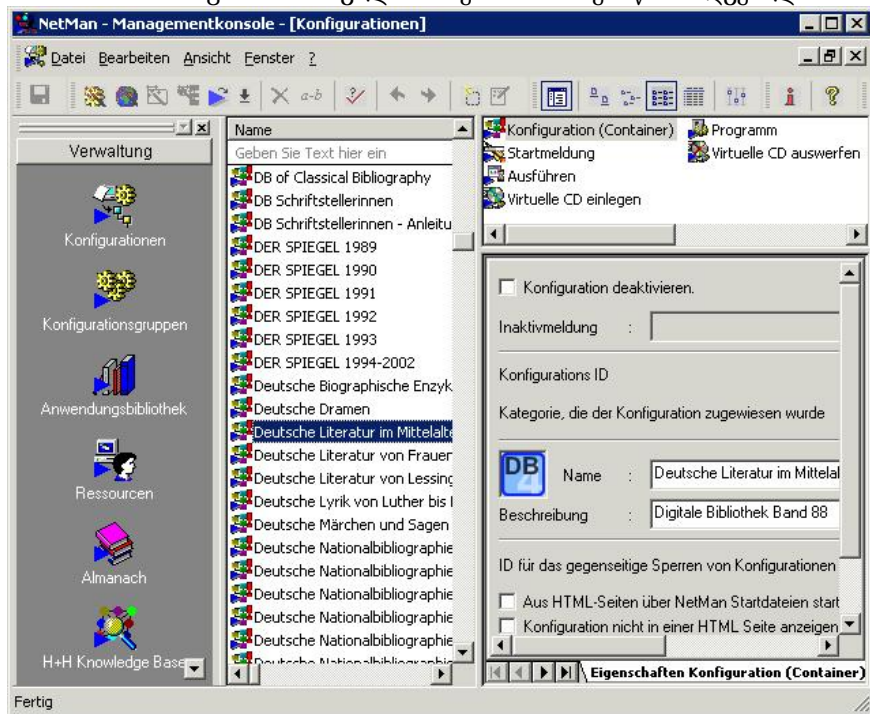
**System Schemata**

```

Users = {u1, u2, ..., um}; Configs = {c1, c2, ..., cn}; Licence-Num : N ; Resources = {r1, r2, ..., rk};
getLicence: Licence-Num -> Licence-Num; returnLicence: Licence-Num -> Licence-Num
var x1: Users; var x2: Configs; var X3: Subset of Resources;
getLicence(LicenceNum) = LicenceNum - 1; returnLicence(LicenceNum) = LicenceNum + 1;
    
```

ნახ.4: პროგრამული სერვერის ზოგადი სქემა

პროგრამა NetMan-ის მუშაობის მაგალითი მე-5 ნახაზზე წარმოდგენილი.



ნახ.5: პროგრამა NetMan-ის მართვის კონსოლი

**4. ლიტერატურა**

1. სურგულაძე გ., გულუა დ. განაწილებული სისტემების ობიექტ-ორიენტირებული მოდელირება უნიფიცირებული პეტრის ქსელებით. მონოგრაფია, სტუ, თბილისი. 2005.
2. NetMan 3.0 Handbuch, 2 Bände, Göttingen. 2005.

**ОРГАНИЗАЦИЯ ПРОГРАММНЫХ СЕРВЕРОВ В КОРПОРАТИВНЫХ СЕТЯХ**

Рейсиг В., Сургуладзе Г., Гулуа Д.  
Университет им. Гумбольдта, Берлин,  
Грузинский Технический Университет

**Резюме**

Рассматривается клиент-серверная архитектура корпоративных сетей, сервер-фарм (терминальный сервер) которой обеспечивает управление рабочими сессиями пользователей и равномерное распределение доступов клиентов к физическим серверам. Задачей кластера (файл-сервера) является надежное хранение общих ресурсов (виртуальные IP-адреса и сетевые имена, совместные программы и сервисы, распределенные каталоги и т.д.), а также организация доступа к внешней памяти (напр., массив SAN -дисков).

**ORGANIZATION OF PROGRAM SERVERS IN CORPORATE NETWORKS**

Reisig Wolfgang, Surguladze Gia, Gulua David  
Humboldt University Berlin, Germany  
Georgian Technical University

**Summary**

The client-server architecture of corporate networks, server-farm (terminal-server) which management of working sessions of users and uniform distribution access of clients to physical servers is considered provides. Task of a cluster (file-server) is a reliable storage of the common resources (virtual IP-addresses and share names, joint programs and the services, distributed catalogue, etc.), and also the organization of access to external memory (for example, a file of SAN-disks).