მონაცემთა სინქრონიზაცია

ირაკლი საბაშვილი

Email: irakli.sabashvili3@gmail.com

**რეზიუმე** — მონაცემთა სინქრონიზაცია აუცილებელი ფაქტორია მონაცემების მუდმივობის შესანარჩუნებლად, როცა მათზე ვახორციელებთ ნებისმიერი სახის ცვლილებას. მისი მიზანია უზრუნველყოს ამ ცვლილებების ერთდროულად მინიჭება ყველა სისტემაზე ან ლოკაციაზე, სადაც ეს მონაცემებია განთავსებული. ერთი შეხედვით, ეს პროცესი, თითქოს უპრობლემო ჩანს, მაგრამ მაინც გვხვდება უარყოფითი მხარეები, რომლებიც სინქრონიზაციის თანმდევია. ჩვენ განვიხილავთ რამდენიმე სახის სინქრონიზაციას: რეპლიკაციური სინქრონიზაცია და ორი სხვა სინქრონიზაციული მეთოდი, რომლებსაც DPM ტექნოლოგია იყენებს. ძირითდი აქცენტი გაკეთბულია რეპლიკაციურ სინქრონიზაციის მეთოდზე, რომელსაც, თავის მხრივ, ორი ტექნიკა გააჩნია. Master-Master და Master-Subordinate ტექნიკები. მათზე სინქრონიზაციის პროცესი სხვადასხვა სირთულით ხორციელდება. რეპლიკაციური სინქრონიზაციის მეთოდი უზრუნველყოფს იმ ძირითადი პრობლემების მოგვარებას, რაც დანარჩენ მეთოდებში მთავარ მახასიათებელ ნაკლს წარმოადგენს. ამ მეთოდის გამოყენება გვაძლევს ყველაზე ოპტიმალურ შედეგს, სხვა მეთოდებთან შედარებით.

**გასაღები სიტვვები -** ინკრემენტული, მუდმივობის შემოწმება, DPM, Master-Master, Master-Subordinate.

2017-2018 სასწ.წელი თსუ კომპიუტერულ მეცნიერებათა დეპარტამენტი

—————————— ◆ ——————————

# 1 შესავალი

დღესდღეობით, ხშირია შემთხვევა როდესაც რამდენიმე სისტემა ერთდროულად მუშაობს ერთსა და იმავე მონაცემზე. ასევე შეიძლება განვათვსოთ მონაცემები რამდენიმე ლოკაციაზე ერთდროულად და გვჭირდებოდეს მათი იდენტურობის შენარჩუნება, ანუ ყველანაირი ცვლილება ამ მონაცემებზე უნდა ისახებოდეს ყველა ლოკაციაზე ერთდროულად.

მონაცემთა სინქრონიზაციის მთავარი მიზანი ისაა, რომ დაამყაროს მუდმივობა სისტემებს შორის და შეინარჩუნოს ეს მუდმივობა მონაცემთა ინფორმაციის თანმიმდევრული, უწყვეტი განახლებით. სიტყვა „უწყვეტს“, ამ შემთხვევაში, ხაზი უნდა გაესვას, რადგან ეს პროცესი არ უნდა ჩავთვალოთ როგორც ერთჯერადად შესრულებადი ამოცანა. სინქრონიზაცია ფუნდამენტალური პროცესია აპლიკაციათა ფართო არჩევანში, რომელთა შორისაა ფაილების სინქრონიზაცია და მობილურ მოწყობილობათა სინქრონიზაცია (მაგ. PDA – Personal Device Assistant).

განვიხილოთ ერთი ზოგადი სიტუაცია, რომელიც საჭიროებს სინქრონიზაციის გამოყენებას. ნებისმიერ საწარმოში, ხშირ შემთხვევაში, არსებობს დაახლოებით 10 სისტემა მაინც, რომლებიც ერთი და იგივე მონაცემებზე მუშაობენ - კლიენტთა მონაცემები, პროდუქტთა მონაცემები, თანამშრომელთა მოანცემები, ანგარიშთა და ინვოისთა სისტემა და სხვა. იმისათვის რომ კომპანიის წარმოების პროცესი იყოს დაკვირვებადი, ყოველი სახის აქტივობა უნდა იყოს სწორად აღრიცხული. მაგალითად, თუ კომპანია ავტომობილებს აწარმოებს, ყოველი ავტომობილისთვის უნდა აღიწეროს: რომელი ნაწილებისგანაა აწყობილი იგი, ნაწილების სერიული ნომრები, თანამშრომლის ID რომელმაც იხმარა ეს ნაწილები ავტომობილის ასაწყობად, პირი რომელმაც იყიდა ეს ავტომობილი. თუმცა, კომპანიაში არსებულ თითოეულ სისტემას გააჩნია ნაწილობრივი ინფორმაცია. მაგალითად, ნაწილების დაყენების სისტემა, რომელიც ინახავს ინფორმაციას მომსახურე თანამშრომლის შესახებ, აგრეთვე საჭიროებს წვდომას ინფორმაციაზე, რომელიც ემსახურება ნაწილების მომწოდებელსა და მარაგში არსებულ ნაწილებს. მიუხედავად იმისა რომ რამოდენიმე სისტემა იყენებს ერთსა და იმავე მონაცემებს, ეს მონაცემები რეგისტრირდება ერთი სისტემის მიერ. ამის შემდეგ აუცილებელია მოხდეს მათი სინქრონიზაცია სხვა სისტემებთან. სინქრონიზაციის შედეგად კი, ყველა სისტემას ექნება წვდომა მათთვის საჭირო ინფორმაციაზე.

# 2 განხორციელებული სამუშაო (არსებული მდგომარეობა)

ზოგადად სინქრონიზაციის განმარტებაში შეიძლება ვიგულისხმოთ ისეთი უბრალო რამ, როგორიცაა MP3 ფლეიერიდან პერსონალურ კომპიუტერში (ან პირიქით), ან თუნდაც მობილურ მოწყობილობებს შორის .mp3 ფაილების გადატანა. მონაცემთა სინქრონიზაციას შეიძლება ჰქონდეს შემდეგი ფორმები:

* მობილური მოწყობილობების სინქრონიზაცია
* ფაილების სინქრონიზაცია
* ვერსიების კონტროლი
* განაწილებული ფაილური სისტემა

ფაილების სინქრონიზაცია გამოიყენება როდესაც ახლდება ფაილები USB ფლეშ დრაივზე ან გარე მყარი დისკის დაბექაფებისას. ეს პროცესი არის შეცდომებისგან უფრო მეტად დაზღვეული ვიდრე სხვა პროცესები და უზრუნველყოფს, რომ უკვე არსებული ფაილები ხელახლა არ გადაკოპირდეს, რითაც დაიზოგება ის დრო, რაც უნდა დახარჯულიყო ამ პროცესზე. ვერსიების კონტროლისას ვხვდებით იმ შემთხვევას, როდესაც რამდენიმე მომხმარებელი ერთდროულად ახდენს მოდიფიკაციას ერთსა და იმავე ფაილზე.

ამჯერად, ჩვენ არ გვაინტერესებს დიდ ფაილებზე შესრულებული ოპერაციები. ჩვენ ორიენტირებულები ვართ მონაცემებთან მუშაობაზე.

იმისათვის რომ განვახორციელოთ მოანცემთა სინქრონიზაცია, შეგვიძლია გამოვიყენოთ შემდეგი მეთოდები:

1. რეპლიკაცია და სინქრონიზაციის მეთოდი (ასევე ცნობილია როგორც რეპლიკაციური სინქრონიზაცია)

2. ორი მეთოდი, რომელებსაც იყენებს DPM (Data Protection Manager) ტექნოლოგია:

* ინკრემენტული სინქრონიზაცია
* სინქრონიზაცია მუდმივობის შემოწმებით.

მუდმივობის შემოწმების მეთოდი ითვლება უფრო სანდო მეთოდად. ის მონაცემთა ცვლილების ფაილური სერვერიდან DPM სერვერზე გადატანასთან ერთად, ასრულებს ნაბიჯ-ნაბიჯ შემოწმებას. ყოველივე ეს, კი უზრუნველყოფს იმას, რომ თითოეული წამოღბული მონაცემი იყოს მუდმივად იდენტური, სერვერში არსებულ დაცულ მონაცემებთან მიმართებაში. ეს პროცესი ინკრემენტულ სინქრონიზაციაზე უფრო ნელია, რადგან ხდება ყველა მონაცემის შედარება და არა პირდაპირ ცვლილების მინიჭება. ასევე გააჩნია ისეთი ნაკლები როგორიცაა: მუშაობის ხარისხის შემცირება და აპლიკაციური ლოდინის დროის გაზრდა, რომელიც გამოწვეულია ქსელურ ტრანზაქციებში კომუნიკაციის დაყოვნების ფონზე.

# 3 სისტემური მოდელი

როდესაც გვინდა, რომ შევარჩიოთ რაიმე მეთოდი, უნდა განვიხილოთ მოცემული ვარიანტები და გავითვალისწინოთ მათი გამოყენების დადებითი და უარყოფითი მხარეები, განსახორციელებელი სამუშაოს კონტექსტის შესაბამისად. ჩვენი მიზანია, რომ გავაკეთოთ სწორი არჩევანი დავალების ოპტიმალურად შესასრულებლად.

იმ შემთხვევაში, თუ ავირჩევთ მუდმივობის შეოწმების მეთოდს. გარანტირებული იქნება ჩვენი მონაცემების მუდმივობის სანდოობა, მაგრამ გვიწევს კონკრეტულ დათმობებზე წასვლა. მაგალითად, მუშაობის ხარისხის დაქვეითება და ლოდინის დროის გაზრდა. ძალიან იშვიათია შემთხვევა, როცა ეს პრობლემა შეიძლება უგულებელვყოთ.

თუ გამოვიყენებთ მეორე მეთოდს, ანუ ინკრემენტულ სინქრონიზაციას, ჩვენ აღმოვფხვრით იმ პრობლემებს, რომლებსაც ვაწყდებით მუდმივობის შემოწმების მეთოდში. თუმცა გვხვდება სხვა უარყოფითი მხარე. ეს უარყოფითი მხარე აისახება მონაცემთა არასანდოობაში, ვინაიდან ინკრემენტული სინქრონიზაციის მეთოდი, სერვერიდან წამოღებულ მონაცემთა ცვლილებებს პირდაპირ მიანიჭებს არსებულ მონაცემებს და აღარ ახორციელებს შედარების ოპერაციას.

# 4 ამოცანის დასმა

ჩვენი ამოცანა მდგომარეობს შემდეგში: განვახორციელოთ სამუშაო, ისე, რომ შევინარჩუნოთ მონაცემთა სანდოობა და ამასთანავე შევძლოთ თავიდან ავიცილოთ შემდეგი სახის პრობლემები: მუშაობის ხარისხის შემცირება, ქსელური შეფერხება, აპლიკაციური ლოდინი, არასაიმედო დაცულობა და ხელმისაწვდომობა. ის ორი სინქრონიზაციის მეთოდი, რომლებსაც DPM იყენებს, თავისი ნაკლოვანებების გამო, ჩვენი ამოცანის სრულად რეალიზების საშუალებას არ გვაძლევს. ამიტომ უნდა ვეძებოთ სხვა მეთოდი, რომელიც ამ მეთოდებზე ოპტიმალური იქნება.

# 5 პრობლემის გადაწყვეტა

ჩვენს წინაშე მდგომი პრობლემის გადასაწყვეტად, ყველაზე კარგ საშუალებად მესახება რეპლიკაცია და სინქრონიზაციის გამოყენება. ამ მეთოდის არჩევის შედეგად ჩვენ გადავჭრით იმ პრობლემებს, რა პრობელმებსაც ვაწყდებოდით DPM-ში იმპლემენტირებული ორი მეთოდის გამოყენებისას. პიველი დიდი პლიუსი, ამ მეთოდისა, არის ის, რომ იგი თავიდან გვაცილებს მონაცემთა დაკარგვისა და კოლიზიის რისკებს. რეპლიკაციური სინქრონიზაცია იყენებს ორ ტექნიკას:

Master-Master რეპლიკაციის მეთოდი უზრუნველყოფს თითოეულ ასლში დინამიკური და განახლებადი მოდელების არსებობას. ეს ტოპოლოგია ითხოვს ორმხრივი სინქზრონიზაციის მექანიზმს, რათა უზრუნველყოს ასლების განახლება და წამოჭრილი კონფლიქტების მოგვარება. იმისათვის, რომ ღრუბლურ აპლიკაციაში პასუხზე ლოდინის დრო დავიდეს მინიმუმამდე და შემცირდეს ქსელური შეფერხება, სინქრონიზაცია ხდება პერიოდულად.

Master-Subordinate რეპლიკაცია, სადაც მონაცემები დინამიკურია მხოლოდ ერთ ასლში (master ასლში) და დანარჩენი ასლები კი read-only-ა. ეს ტოპოპლოგია Master-Master-ის ტოპოლოგიაზე მარტივ სინქრონიზაციულ პროცესს მოითხოვს, რადგან აქ ნაკლებად მოსალოდნელია რაიმე კონფლიქტის შექმნის ვარიანტი.

# 6 ანალიზი

რეპლიკაციური სინქრონიზაციის არჩევისას ჩვენ აღარ გვიწევს ფიქრი იმ პრობლემებზე, რაც DPM-ის მეთოდების გამოყენებისას გვხვდებოდა. განვიხილოთ ის პრიორიტეტები რაც გააჩნია ამ მეთოდის გამოყენება:

მუშაობის ხარისხისა და მაშტაბურობის გაუმჯობესება - Master-Subordinate ტექნიკის გამოყენება read-only ასლებთან, გვაძლევს გაუმჯობესებულ პერფორმანსს. Master-Master ტექნიკა კი უზრუნველყოფს ჩაწერის ოპერაციების მასშტაბბურობას.

სანდოობა და ლოდინის დროის მინიმიზაცია - ასლების განთავსება აპლიკაციის ქსელში წვდომის საზღვრებში, უზრუნველყოფს ლოდინის დროის შემცირებას, რომელიც შეიძლება გამოწვეული იყოს ინტერნეტ სივრცეში მონაცემებზე წვდომის განხორციელებისას. აგრეთვე შენარჩუნებულია საიმედოობის მაღალი ხარისხი. ზოგადად, ხანგრძლივი ლოდინის დრო და ქსელური შეფერხებები, გამოწვეულია დაბალი დონის სანდოობის შემთხვევებში.

დაცულობა - ჰიბრიდულ აპლიკაციებში მხოლოდ არაკონფიდენციალური მონაცემების განთავსება და დანარჩენის on-prem-ზე (On-Premises Software) შენახვა უზრუნველყოფს მონაცემთა დაცულობა, ვინაიდან რეპლიკაციური სინქრონიზაცია განხორციელდება მხოლოდ ამ არაკონფიდენციალურ მონაცემებზე.

ხელმისაწდომობის გაუმჯობესება - Master-Master-ის განთავსება იმ რეგიონის დატაცენტრში, რომელშიც მუშაობს აპლიკაცია, აძლევს მას საშუალებას მიწვდეს ამ ინფორმაციას და შეამციროს მოანცემთა გადატანის ხარჯები. ხელსაყრელია გვქონდეს მთავარი დატაბაზიდან წამოღებულ მონაცემთა დამატებითი ასლები. ამ ასლების ყოველ ჯერზე განახლება, როდესაც რაიმე ცვლილება შევა ამ მონაცემებში, საშუალებას მისცემს აპლიკაციას წვდომა ჰქონდეს ძველ ვერსიებთან, იმ შემთხვევაში თუ მოხდა ორიგინალის დაზიანება. ამასთანავე, რაც უფრო მეტი ასლი გვაქვს, მით უფრო დიდი შანსია, რომ საჭირო მონაცემები მოიძებნოს იქვე, სადაც ხორციელდება ტრანზაქცია.

# დასკვნა

რეპლიკაციური სინქრონიზაციის საშუალებით, ჩვენ შეგვიძლია საიმედოდ და სწრაფად მოვახდინოთ მონაცემთა რეპლიკაცია და სინქრონიზირება, ამავდროულად შევინარჩუნოთ მუშაობის ხარისხი და შევამციროთ აპლიკაციური ლოდინის დრო. გვქონდეს მაღალი დაცულობა და მარტივი ხელმისაწვდომობა. თუმცა, როდესაც არჩევანს ვაკეთებთ ამ მეთოდზე უნდა გავითვალისწინოთ მისი უარყოფითი მხარეებიც:

განახლებებზე დახარჯული სამუშაო - განახლება აუცილებელი და საჭირო ფაქტორია, თუ ჩვენ ვმუშაობთ მონაცემებზე და მათ სინქრონიზაციაზე. ამ შემთხვევაში, ყოველი განახლებისას, მონაცემთა ბაზამ უნდა უზრუნველყოს თითოეული ასლის განახლება. ეს კი ითხოვს დიდ სამუშაო დანახარჯებს.

დისკზე დიდი მოცულობის დაკავება - ყოველი ახალი ასლის შექმნა იკავებს მოცულობის გარკვეულ ნაწილს. როდესაც ჩვენ ვქმნით უამრავ ასლს ერთსა და იმავე მონაცემზე, შესაბამისად იზრდება მოთხოვნა დისკის მოცულობაზეც.

ძვირადღირებულობა - რესურსების ერთდროულად გამოყენების (პარალელიზმის), მონაცემთა აღდგენის და სინქრონიზაციის ტექნიკები, დროთა განმავლობაში უფრო და უფრო იხვეწება და შესაბამისად, ახალ და გაუმჯობესებულ ვერსიებზე გადასვლა და ხმარებაში შემოღება ჯდება უფრო ძვირი.

აღსანიშნავია, რომ ამ მეთოდითაც ვერ შევასრულებთ ყველანაირი სახის დავალებას, მიუხედავად მისი დადებითი მხარეებისა. ამიტომ კვლევა გრძელდება ამ მეთოდის დასახვეწად და უკეთესი ხერხების საპოვნელად, რათა შევძლოთ ამდაგვარი სახის ამოცანების გადაწყვეტა უფრო ოპტიმალურად.

# ლიტერატურა

|  |  |
| --- | --- |
| [1] | "Msdn.Microsoft," [Online]. Available: https://msdn.microsoft.com/en-us/library/dn589787.aspx. |
| [2] | "Technet.Microsoft," [Online]. Available: https://technet.microsoft.com/en-us/library/cc161401.aspx. |
| [3] | O. Wolfson, "Tods.Acm," 25 10 2009. [Online]. Available: https://tods.acm.org. |