

# Big Data-nın Proqram Mühəndisliyinin Bəzi Problemləri

Məkrufə Hacırahimova<sup>1</sup>, Aybəniz Əliyeva<sup>2</sup>

<sup>1,2</sup>AMEA İnformasiya Texnologiyaları İnstitutu, Bakı, Azərbaycan

<sup>1</sup>makrufa@science.az, <sup>2</sup>aybeniz63@rambler.ru

**Xülasə—** Verilənlərin həcmnin sürətlə artması və emalı proqram təminatı sahəsindəki problemlərin dərinləşməsinə və yeni problemlərin meydana çıxmasına səbəb olmuşdur. Məqalədə bu problemlər və onların bəzi həlli yolları nəzərdən keçirilmişdir.

**Açar sözlər—** big data, proqram təminatı, proqram mühəndisliyi, onlayn diaqnostika, miqyaslanan arxitektura

## I. GİRİŞ

Müasir informasiya texnologiyalarının (İT) cəmiyyətin müxtəlif sferalarında yayılması, insanların İnternetə çıxış imkanlarının genişlənməsi, eləcə də İnternet vasitəsilə göstərilən xidmətlərin sayının artması verilənlərin yeni mənbələrinin yaranmasına səbəb olur. Sosial şəbəkələr, sensor qurğuları, biznes tranzaksiyaları və Əşyaların İnterneti böyük həcmdə verilənlərin rəqəmsal dalğasını yaradır. Bu verilənlərin toplanması, emalı, inteqrasiyası, analizi və arxivləşdirilməsi lazım gəlir [1,2]. Digər tərəfdən böyük verilənlər həcmində, intensivliyinə və mürəkkəbliyinə görə standart proqram platformalarının imkanlarını aşır, verilənlərin ənənəvi üsullarla emalı zamanı problemlər yaradır. Verilənlərin saxlanması və idarə edilməsinin paralel və paylanmış arxitekturası proqram təminatının Big Data (BD) paradigmasına uyğunluğunu tələb edir. Həmçinin, verilənlərin ölçüsünün artması ilə yanaşı, onların strukturunun da mürəkkəbləşməsi yeni proqram təminatı sistemlərinin yaradılmasını və klassik problemlərin aradan qaldırılması üçün yeni proqram mühəndisliyi prinsip və üsullarının tətbiqini zəruri edir [3,4].

Verilənlərin həcmnin sürətlə artması, əlyətərliyi və müasir texnologiyalar BD ətrafında fəaliyyətin intensivləşməsinə səbəb olur [5]. BD elmi kəşflərin və iqtisadi dəyərin yeni səviyyəsini vəd edir, biznes münasibətlərini sürətləndirir və biznes fəaliyyəti üçün böyük imkanlar yaradır. BD-nin səmərəli emalı və istifadəsi isə miqyaslanan texnologiyaların və alətlərin sayəsində mümkün ola bilər. Bu səbəbdən BD-nin proqram mühəndisliyi sahəsindəki problemlərin və bu sahədəki mövcud vəziyyətin araşdırılmasına ehtiyac vardır.

## II. İKT SAHƏSİNDƏ ƏSAS TENDENSİYALAR VƏ ONUN PROQRAM MÜHƏNDİSLİYİNƏ TƏSİRİ

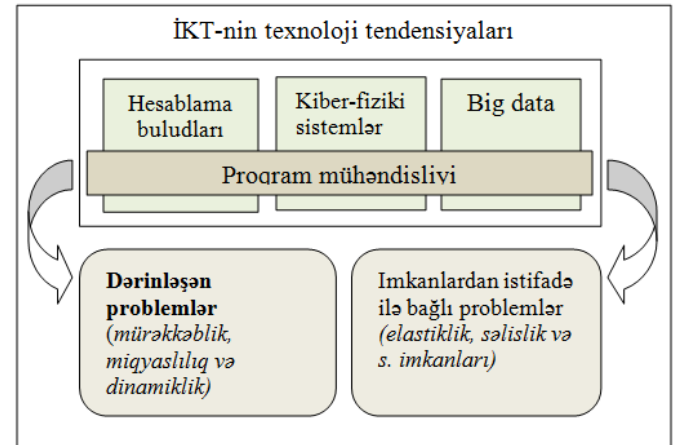
İKT sahəsində meydana çıxan müasir texnologiyalar proqram təminatı sistemlərinin təkmilləşdirilməsinə və uyğun olaraq, yeni proqram mühəndisliyi metodlarının yaranmasına səbəb olmuşdur. Son illər proqram mühəndisliyi sahəsindəki tədqiqatların uğurlu nəticələrinə baxmayaraq, İT-nin hesablama buludları (cloud computing), kiber-fiziki sistemlər və BD kimi yeni tendensiyaları bu sahədəki mövcud

problemlərin dərinləşməsinə və yeni problemlərin meydana çıxmasına səbəb olmuşdur (şəkil 1) [1].

Hesablama buludları cari və növbəti onilliyin xidmət və proqram təminatının bütün sahələrinə təsir edən innovativ texnologiyadır.

Kiber-fiziki sistemlər innovasiya məhsul və xidmətlərinin inkişaf etdirilməsinin əsasını təşkil edir.

BD biznes, elm və idarəetmə fəaliyyətində səmərəli qərarların qəbulunda istifadə üçün intellektual üsullarla verilənlərdən dəyərli informasiyanın çıxarılmasıdır [1,2, 4].



Şəkil 1. İnformasiya texnologiyaları tendensiyaları və proqram mühəndisliyi problemləri

Şəkildən görüldüyü kimi texnoloji tendensiyalar proqram mühəndisliyində 2 növ problemin yaranmasına səbəb olur. Bir tərəfdən, proqram mühəndisliyinin mövcud problemləri yeni İT-nin mürəkkəbliyi, ölçüsü (miqyası) və ya dinamikliyi üzündən dərinləşir. Nümunə üçün, buludlarda irsi (legacy) kodun mürəkkəb və dinamik şəbəkə sistemlərində effektiv miqyası, BD proqram əlavələrinin keyfiyyətinin təmini məqsədilə səmərəli testetmənin həyata keçirilməsi daha da mürəkkəbləşmişdir. Digər tərəfdən, proqram təminatının problemlərinin həlli üçün müasir İT-nin yaratdığı imkanların necə istifadə edilməsi ilə bağlı problemlər yaranır. Eenerjinin səmərəli istifadəsinə əsaslanan proqram təminatının yaradılmasında hesablama buludlarının, alıcıların tələbatını öyrənmək üçün sürətli əks əlaqənin qurulmasında buludların və ya korporativ acıq baza verilənlərinin daha yaxşı idarə

edilməsində BD analitikanın imkanlarından istifadə olunmaması və s. belə problemlərdəndir [1].

Hazırda BD-nin potensialının istifadəsi ilə bağlı bir çox problemlər mövcuddur. Bu problemlərin bəziləri BD-nin xüsusiyyətləri, bəziləri cari analiz model və üsulları, digər bir qismi isə verilənlərin müasir emal sistemlərinin məhdudiyətləri ilə bağlıdır. BD ənənəvi hesablama üsulları ilə emal olunması mümkün olmayan böyük həcmli verilənlərdir. BD-nin tələblərinə uyğun olaraq ənənəvi platformalar, arxitekturalar, metodologiyalar, alqoritmlər və alətlər təkmilləşdirilir və ya yeniləri yaradılır. Nümunə üçün, bu verilənlər paylanmış fayl sistemində saxlanılır, onların emalı isə paylanmış hesablama üsullarına əsaslanır. Aydın ki, strukturlaşmış və strukturlaşmamış böyük verilənlərin analitika alətləri, intellektual analiz üsulları, statistik yanaşmalar, alqoritmlər, vizuallaşdırma metodları verilənlərin ənənəvi üsul, alqoritm və yanaşmalarından fərqli olacaqdır. Bu fərqlilik proqram mühəndisliyi sahəsində də özünü göstərir.

### III. BIG DATA-NIN PROQRAM MÜHƏNDİSLİYİ PROBLEMLƏRİ

Verilənlərin həcm, sürət və müxtəlifliyi proqram təminatı sənayesində ciddi problemlərin yaranmasına səbəb olmuşdur. Problem ilk növbədə miqyas amili ilə bağlıdır və proqram təminatı arxitekturası bir qayda olaraq paylanmış sistemlərdə verilənlərin paralel emalı zamanı yaranan problemlərlə üzləşir [3, 5]. Ümumiyyətlə, BD-nin proqram mühəndisliyi sahəsindəki problemləri aşağıdakı kimi qruplaşdırmaq olar.

#### *1) Big data üçün miqyaslanan proqram təminatı arxitekturalarının kifayət qədər inkişaf etməməsi*

Klassik arxitekturalar böyük həcm, sürət, müxtəliflik və küyə malik Big Data sistemləri və proqramları ilə iş zamanı zəruri olan miqyaslanma, məhsuldarlıq və əlyetərliyin fundamental məsələlərini həll etmək üçün yararlı deyildir. Bu isə özlüyündə Big data sorğularını və axtarışını optimallaşdıran proqram təminatının yeni nəslinin yaradılmasını zəruri edir [3, 6].

Son dövrlərdə verilənlərin intensiv emalına əsaslanan sistemlərdə (data-intensive systems) vertikal miqyaslanmaya malik SQL (Structured Query Language) verilənlər bazası texnologiyalarından geniş istifadə olunurdu. Verilənlərin həcmi böyük olduğundan, onların saxlanması və ötürülməsi ənənəvi üsullardan fərqlənir. SQL verilənlər bazasının məhdudiyətləri (bütün sorğuların diskdə yerləşmiş verilənlər bazasında aparılması və sürətli olmaması) müxtəlif serverlərdə paylanmış böyük həcmli verilənlərin saxlanması və ötürülməsi tələblərinə cavab verən yeni modellərin yaradılmasını zəruri edir [6,7].

Hazırda Big Data ilə işləyən relyasion verilənlər bazasının inkişaf etdirilməsi ilə yanaşı, yeni sinif NoSQL sistemləri yaradılmışdır. NoSQL sistemləri (Cassandra, Riak, MongoDB və s.) proqramçılara verilənlərin idarəetmə mexanizmlərinə girişi təmin edirlər, ucuz və yüksək məhsuldarlıqlı server klasterləri vasitəsi ilə üfqi miqyaslanmanı həyata keçirirlər.

Belə sistemlərdə yüksək sürət, saxlamanın və əlyetərliyin elastikliyi klaster daxilində verilənlərin bölünməsi və tirajlanması hesabına təmin olunur [6,7,8].

BD-nin ötürülməsi üçün hazırlanmış Aspera kimi proqram təminatı verilənlərin sürətli ötürülməsi ilə yanaşı təhlükəsizliyini də təmin edir [9].

#### *2) Big data-nın proqram təminatının keyfiyyətinin proqram mühəndisliyi üsulları ilə təmini məsələləri*

Qeyd olunan problem aşağıdakı məsələlərin həllini tələb edir: BD-nin proqram tətbiqlərinin dayanıqlığı və etibarlılığının təmini üçün tipik test verilənlərinin yaradılması; BD-nin yoxlama üsulları vasitəsilə testetmənin tamamlanması; BD proqram tətbiqlərinin keyfiyyətinin yoxlaması üçün sürətli prototiplərin istifadəsi və s. [3].

BD-nin proqram təminatının test edilməsi zamanı verilənlərin emalına çox vaxt sərf olunması, çoxlu sayda serverlər arasında ötürülən və dəyişikliyə uğrayan verilənlərin hər bir keçid nöqtəsində yoxlanılması ilə bağlı problemlər meydana çıxır. Miqyaslanan test proqramlarından (test framework) istifadə etməklə bu problemləri həll etmək mümkündür. Belə ki, test proqramı giriş fəzasının bölmələrini testləşdirməklə ilkin böyük həcmli verilənlərdən tipik kiçik verilənlər yığımı yaradır. Testetmə üçün bu verilənlər yığımindan istifadə edilməsi sürətli proseslər zamanı onların aramsız inteqrasiya və ötürülməsinə mane olmayacaqdır. Həmçinin testetmə proqramı verilənlərin ötürüldüyü və dəyişikliyə uğradığı müxtəlif keçid nöqtələrində onların girişini və yoxlanmasını təmin edəcəkdir [10].

#### *3) BD-nin proqram təminatının onlayn diaqnostikası*

BD-nin proqram təminatı arxitekturalarının miqyaslanması son nəticədə onun testetmə və diaqnozu ilə bağlı problemlərin yaranmasına səbəb olur. Proqram tətbiqlərinin paylanması və böyük həcmdə verilənlər toplusunu idarə etməsi səbəbindən genişlənmədən əvvəl proqram təminatının dəyişikliklərini yoxlamaq üçün hərtərəfli test mühiti yaratmaq mümkün deyildir. İstehsalda qaçılmaz problemlər baş verən zaman, sürətli diaqnoz yalnız genişlənmiş monitoring və jurnal qeydləri (logging) vasitəsilə əldə edilə bilər. Monitoring loqlarının analizi özlüyündə BD problemlərindəndir. Belə ki, kompleks sistemlər üçün loqlar kiçik zaman aralıklarında asanlıqla böyüyə bilər. Ucuz qiymətə miqyaslanan infrastrukturun sistemə daxil edilməsi bu problemi həll etməyə imkan verə bilər [3, 6,11].

Əməliyyat zamanı BD sistemlərinə nəzarət edilməsi və keyfiyyətinin təmin edilməsində: 1) BD texnologiyalarından istifadə edilməsi; 2) böyük həcmdə meta-verilənləri analiz etməklə arzuolunmaz modellərin və yayınmaların aşkarlanması və s. mühüm rol oynaya bilər [3].

#### *4) Verilənlər axını və saxlanma üçün yeni və təkmilləşdirilmiş proqram təminatı alqoritmlərinin yaradılması*

Verilənlərin həcmi böyük olduğundan, onların saxlanması və ötürülməsi ənənəvi üsullardan fərqlənir. Hazırda çox sayda nisbətən ucuz qiymətə saxlama anbarlarının olmasına baxmayaraq, böyük həcmdə verilənlərin saxlanması və sorğusu problem kimi qalmaqda davam edir. Bulud saxlama xidmətləri, adətən, onlar üçün adekvat hesab olunmur. Çünki onlarda əməliyyatlar üçün aralıq sorğuların dəstəklənməsi kifayət deyildir [3, 12].

BD müxtəlif serverlərdə paylandığından onun emalı üçün Hadoop, NoSQL, MapReduce və s. kimi paralel paylanmış sistemlər lazımdır. Hadoop paylanmış fayl sistemi (Hadoop distributed file system – HDFS) klasterdə paylanmış şəkildə saxlanmış BD-nin əlyətərliyi problemini həll edir. HDFS böyük ölçülü klasterlərdə iri həcmli verilənlərin etibarlı saxlanması üçün nəzərdə tutulmuşdur. Ənənəvi paylanmış fayl sistemləri ilə müqayisədə HDFS iki əhəmiyyətli üstünlüyə malikdir [13]:

➤ *Nasazlıqlara yüksək dözümlülük (fault-tolerance).* Verilənlərin uzun müddət saxlanmasını təmin etmək üçün verilənlərin mühafizəsi mexanizmlərindən istifadə edən klassik paylanmış fayl sistemlərindən fərqli olaraq, HDFS böyük həcmdə verilənlərin bir neçə qovşaqlarda təkrar saxlanmasını təmin edir. Bir qovşaqlarda nasazlıq aşkarlandıqda digər qovşaqlardan istifadə olunur;

➤ *Böyük həcmli verilənlər.* Hazırda Hadoop petabaytlarla verilənləri saxlamaq imkanına malikdir. O, verilənlər yığımının yüksək ötürülmə qabiliyyətini və klasterin yüzlərlə qovşaqlarda miqyaslanmasını dəstəkləyir.

NoSQL və MapReduce Big Data-nın səmərəli saxlanması, təqdimi və sorğuları üçün geniş imkanlara malikdir [1,14]. NoSQL sistemləri server klasterlərində verilənlərin bölünməsi və tirajlanması hesabına saxlanmanın və əlyətərliyin səmərəliliyini təmin edirlər. MapReduce paralel paylanmış alqoritmlərin vasitəsilə klasterdə böyük verilənlər yığımının emal və generasiyasının həyata keçirən proqramlaşdırma modelidir. MapReduce böyük klasterdə BD-nin emalı zamanı miqyaslanma və nasazlıqlara yüksək dözümlülük kimi xüsusiyyətlərə malikdir [13]. Lakin böyük həcmdə verilənlərin sorğusu zamanı bu sistemlərdə bəzi problemlər meydana çıxır.

Beləliklə, BD-nin səmərəli saxlanması, təqdimi və sorğulanması ilə bağlı problemlərin həlli üçün aşağıdakıların həyata keçirilməsi zəruridir:

- dinamik verilənlər axını rejimində verilənlər obyektinin saxlanması və klasterləşməsinə həyata keçirən alqoritmlərin yaradılması;
- informasiyanın axtarışı və çıxarılmasını asanlaşdıran, həmçinin onu anlaşılan şəkildə təqdim edən alqoritmlərin yaradılması;
- verilənlərin emal vaxtını kifayət qədər azaltmağa imkan verən səmərəli saxlama üsullarının istifadəsi;
- klasterin bir çox qovşaqlarında saxlanmanın optimallaşdırılması və s. [1, 8].

5) *Proqram mühəndisliyində Big Data analitikadan istifadə məsələləri*

BD-nin tətbiqləri tez-tez kompleks problemlər yaradır. İstifadəçilərin ehtiyaclarının başa düşülməsi, proqram təminatının nöqsanlarının təhlili və s. kimi proqram mühəndisliyinin cari problemlərini hər hansı analitik üsulların köməyi ilə həll etmək mümkün deyil. Müasir verilənlər elmi (data science) data mining, statistik analiz, maşın təlimi (machine learning), paylanmış proqramlaşdırma, real vaxt rejimində analiz (real-time analysis), yaddaşda analiz (in-memory analysis) və insan-kompüter qarşılıqlı əlaqəsi (human-computer interaction) kimi üsul və yanaşmaları özündə birləşdirir. Bu üsullar BD-nin müxtəlif platformalarında istifadə olunur və proqram mühəndisliyinin problemlərinin həllində tətbiq edilir [14, 15].

6) *Big Data-nın mühəndislik üsul və strukturu ilə bağlı məsələlər*

BD verilənlərin saxlanma və emal üsullarını, saxlanma arxitekturasını, verilənlərə giriş mexanizmlərini köklü surətdə dəyişir. Bu dəyişiklik BD-nin proqram mühəndisliyi prinsip və üsullarında da özünü göstərir. Lakin yeni idarəetmə və analiz sistemlərinin yaradılmasına böyük həcmdə maliyyə vəsaitləri tələb edir. Uyğun metodlar və platformalar vasitəsilə BD proqram mühəndisliyini necə dəstəkləmək olar? Onlayn analitik emal (OLAP) sistemlərini BD-nin analizi strukturuna necə uyğulaşdırmaq? [3]. Bu suallar proqramçılar qarşısında böyük elmi-tədqiqat işlərinin aparılması məsələsini qoyur.

BD-nin proqram mühəndisliyinin yuxarıda qeyd olunmuş və qeyd olunmamış problemləri hələlik tam həllini tapmamışdır. Hazırda bu problemlərin həlli istiqamətində eimi-tədqiqat işləri aparılır və böyük həcmdə investisiya tələb edən innovativ layihələr həyata keçirilir [3].

## NƏTİCƏ

Aparılan tədqiqatlar BD-nin proqram mühəndisliyi sahəsində yaranan problemlərin əsasən verilənlərin miqyaslanması amili ilə bağlı olduğunu göstərir. Həmçinin miqyaslanmaya malik verilənlər sistemlərinin yaradılması və layihələndirilməsi üçün Big Data texnologiyalarından istifadə olunması proqram təminatı arxitekturasının layihəçiləri üçün əhəmiyyətli problemlər yaradır. Bu problemlərin həlli üçün layihəçilərin birgə işi və texniki biliyi əhəmiyyətli rol oynaya bilər. Lakin verilənlərin miqyaslanan idarəetmə və analiz sistemlərinin yaradılması böyük həcmdə investisiyaların cəlb edilməsi və risklərin artması səbəbindən problemləri daha da kəskinləşdirir. Ona görə də miqyaslanmanın və yeni texnologiyaların fundamental məsələlərinin nəzərə alınması proqram mühəndisliyinin uğuru üçün əsas şərtlərdəndir [7].

Bütün çətinliklərə baxmayaraq, BD-nin və İKT-nin digər gələcək tendesiyalarının imkanlarından səmərəli istifadə edilməsi, sürətlə dəyişən texnologiyaların və cəmiyyətdə baş verən dəyişiklərin tələblərinə cavab verməsi üçün proqram mühəndisliyinin prinsipləri, metodları və alətləri inkişaf etdirilməli və ya yeniləri hazırlanmalıdır.

ƏDƏBİYYAT

- [1] I. Gorton, “Addressing the Software Engineering Challenges of Big Data”, 2013, <https://insights.sei.cmu.edu/2013/10/addressing-the-software-engineering-challenges-of-big-data.html>
- [2] R.M. Əliquliyev, M. Ş. Hacırahimova, A.S. Əliyeva, "Big data-nin aktual elmi-nəzəri problemləri", *İnformasiya cəmiyyəti problemləri*, 2016, №2, səh. 37–49
- [3] NESSI – Software Engineering White Paper: Software engineering for and with Big Data, 2014, [http://www.nessieurope.eu/Files/Private/NESSI\\_SE\\_WhitePaper-FINAL.pdf](http://www.nessieurope.eu/Files/Private/NESSI_SE_WhitePaper-FINAL.pdf)
- [4] M. Sh. Hajirahimova, A.S. Aliyeva, “Review of statistical analysis methods of high-dimensional data”, *Eastern-European Journal of Enterprise Technologies*, Kharkov, 2015, no 5, pp. 23-30. <http://journals.uran.ua/eejet>
- [5] “DRAFT European Big Data Value Strategic Research & Innovation Agenda, Version 0.7,” 2014, <http://www.bigdatavalue.eu/index.php/downloads/>
- [6] I. Gorton, J. Klein, “Distribution, Data, Deployment: Software Architecture Convergence in Big Data Systems”, [https://resources.sei.cmu.edu/asset\\_files//2014\\_019\\_001\\_90915.pdf](https://resources.sei.cmu.edu/asset_files//2014_019_001_90915.pdf)
- [7] И. Гортон, Дж. Клейн, “Системы для Больших Данных: конвергенция архитектур”, *Открытые системы. СУБД*, 2015, № 03, стр. 24-28. <https://www.osp.ru/iz/bigdata/articles/13046898>
- [8] I. Gorton, A.B.Bener, A.Mockus, “Software Engineering for Big Data Systems”, 2016, *IEEE Software*, pp. 32-35, [www.computer.org/software](http://www.computer.org/software)
- [9] T. Huang, L. Lanc, X. Fanga and etc., “Promises and Challenges of Big Data Computing in Health Sciences”, *J.Big DataResearch*, 2015, vol.2, pp.2–11
- [10] N. Li, A. Escalona, Y. Guo; J. Offutt. “A Scalable Big Data Test Framework”, 2015, *IEEE 8th International Conference on Software Testing, Verification and Validation (ICST)*, <http://ieeexplore.ieee.org/abstract/document/7102619/>
- [11] K. M. Anderson, “Embrace the Challenges: Software Engineering in a Big Data World”, *IEEE/ACM 1st International Workshop on Big Data Software Engineering (BIGDSE)*, 2015
- [12] “Big Data: A New World of Opportunities,” 2012, [http://www.nessieurope.com/Files/NESSI\\_WhitePaper\\_BigData.pdf](http://www.nessieurope.com/Files/NESSI_WhitePaper_BigData.pdf).
- [13] P.L. Chen, C.Y. Zhang, “Data-intensive applications, challenges, techniques and technologies: A survey on Big Data”, *J.Information Sciences*, 2014, vol. 275, pp.314–347
- [14] S. Koussouris, E. D. Nitto, “Current and Future Challenges of Software Engineering for Services and Applications”, *4\_20161006\_Workshop\_Summary\_Challenges of Software Engineering*, 2916, <https://www.pac-online.com/>
- [15] X. Jina., B. W. Waha, X. Chenga, Y. Wanga, “Significance and Challenges of Big Data”, *J. Big Data Research*, 2015, vol.2, no.2, pp. 59–64