УДК 37.18.43(082)

Кисельов Г.Д.
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

Мироненко С.С.
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ІНФРАСТРУКТУРА ПЛАТФОРМИ ЕЛЕКТРОННОЇ ПІДТРИМКИ НАВЧАЛЬНОГО ПРОЦЕСУ

Дистанційне навчання, як і будь-яка інша технологія освіти або підвищення кваліфікації, містить три складові частини: організація процесу навчання, навчальний контент, моніторінг якості освіти. Таким чином, система дистанційного навчання або Learning Management System (LMS) – це сукупність програмних застосувань, які автоматизують всі ці складники дистанційної освіти. У статті розглянуто важливість застосування LMS поруч із традиційним процесом навчання. Розглянуто стандартизацію LMS, можливість і необхідність імплементації технології Big Data, а також технологічну схему процесу дистанційного навчання. Розглянуто стек технологій, що необхідний для побудуви засобів обробки даних в системі, а також їх можливості щодо надання результуючих даних для моніторінгу знань студентів дистанційної освіти. Розглянуто сильні та слабкі місця технологій обробки та збереження даних. Вказано на необхідність застосування в LMS методів аналізу великих масивів слабко структурованих даних і побудови архітектури повного аналізу даних. Наведено приклади застосування технології Big Data загалом і приклади продуктів сімейства Big Data, що можуть бути застосовані для побудування аналітичного складника в LMS.

Ключові слова: *LMS, Big Data, Hadoop, NoSQL, Data Lake, Map Reduce, Framework, змішане навчання, традиційні дані, навчальний процес.*

Постановка проблеми. Відомо, що засвоєння готових знань – це не ціль, а всього лише один із засобів інтелектуального розвитку людини. Педагогічні системи сьогодні не мають права, як в минулому сторіччі, будувати навчання на засвоєнні суми готових знань, тому практично у всіх розвинених країнах зроблений акцент на навчанні умінню самостійно добувати потрібну інформацію, відокремлювати проблеми і шукати шляхи їх раціонального рішення, уміти критично аналізувати одержувані знання і застосовувати їх для вирішення нових задач.

Постановка завдання. Розглянути важливість застосування LMS поруч із традиційним процесом навчання. Розглянути стандартизацію LMS, можливість і необхідність імплементації технології Big Data, а також технологічну схему процесу дистанційного навчання. Розглянути стек технологій, що необхідний для побудуви засобів обробки даних в системі, а також їх можливості щодо надання результуючих даних для моніторінгу знань студентів дистанційної освіти. Виявити сильні та слабкі місця технологій обробки та збереження даних.

Виклад основного матеріалу. Ідеальна система навчання має виконувати наступні функції [1]:

– сформувати у студента бажання вчитися і визначити ціль навчання;

– підтримувати мотивацію до навчання і творчої діяльності;

– забезпечити кожного студента індивідуально-адаптованими навчальними матеріалами;

– дати кожному студенту можливість вчитися за індивідуальним графіком;

– безупинно оцінювати результативність навчання.

Зрозуміло, що мова йде про ідеалізовану систему навчання, якою до останнього часу вважалась традиційна система. Однак, аналізуючи ці завдання, можна дійти висновку, що більшість з цих задач можна виконати, обравши за основу метод дистанційного навчання (е-навчання). Так, сформувати в студента бажання вчитися і визначити ціль навчання можна, використавши підхід ділової гри. Для підтримки мотивації можна ввести систему проміжних оцінок. Індивідуальний графік навчання надається інструментарієм дистанційного навчання. Досвід існування дистанційного навчання в вищій школі виявив цілий ряд його недоліків. Це підвищені вимоги до якості дистанційних навчальних курсів, складність, а в деяких

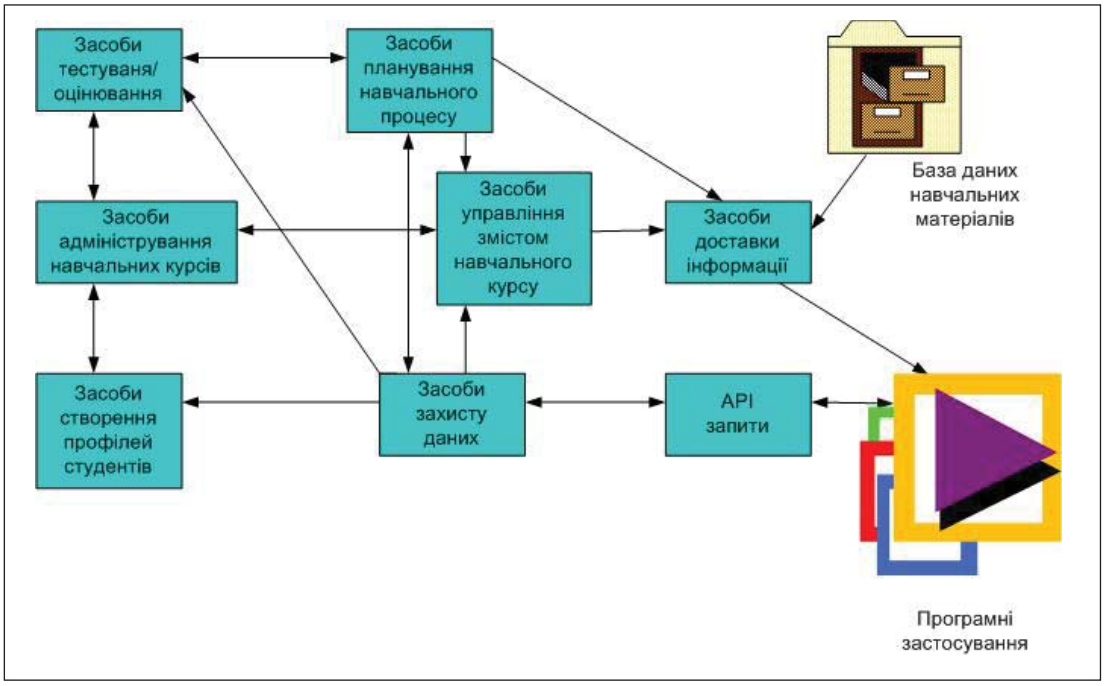

Рис. 1. Архітектура платформи е-навчання

випадках неможливість проведення практичних та лабораторних занять, складність дистанційного контролю знань. Новим підходом вирішення проблем навчання є використання технології змішаного навчання, коли поєднуються традиційна форма навчання і дистанційне навчання. Змішане навчання передбачає елементи самостійного контролю студентом освітнього маршруту, часу, місця і темпу навчання, а також інтеграцію досвіду традиційного навчання і дистанційного. Ефективність будь-якого виду навчання, особливо дистанційного і змішаного, залежить від його інструментальної (програмно-системної) підтримки або платформи електронного навчання [2].

Схема процесу навчання

Класична реалізація платформи е-навчання на рівні сервера включає засоби керування дистанційним навчанням (накопичення інформаційних ресурсів, розподіл прав доступу до навчальної інформації, контроль процесу навчання і засвоєння знань) і базу навчальних матеріалів (рис. 1). На клієнтському рівні створюється взаємодія користувачів (студентів і викладачів) з сервером [1].

Таке середовище має містити наступні компоненти:

– керування навчальним процесом, тобто планування і контроль навчання;

– контроль (діагностики) рівня підготовки студента;

– комплекс автоматизованих навчальних курсів, що охоплюють усі необхідні теми навчання;

– персональний кабінет студента;

– персональний кабінет викладача навчальної дисципліни;

– персональний кабінет викладача-автора навчальних матеріалів дисципліни;

– розподілені мережеві репозиторії навчальних матеріалів;

– експертну систему контролю і діагностики рівня знань.

Погоджена робота перелічених компонентів забезпечується системами дистанційного навчання (DLS – Distance Learning System).

Навчальні інформаційні ресурси створюються, починаючи від найпростіших текстових файлів і гіпертекстових систем допомоги і кінчаючи мультимедійними навчальними курсами, електронними підручниками, відеоматеріалами і засобами контролю / самоконтролю знань. Ефективність засобів комп'ютерного дистанційного навчання, що дозволяє підвищити «розуміння» проектувальниками інструментальних середовищ проектування, визначається насамперед наявністю продуманих LMS (Learning Management System) програмних засобів керування процесом навчання, що є невід'ємною частиною DLS. Під час створення LMS враховують досягнення технологій дистанційного навчання в класичних установах освіти, що закріплюються в стандартах. Розроблювачі LMS використовують, як правило, специфікацію IEEE LTSC P1484 комітету зі стандартів технологій навчання (LTSC – Learning

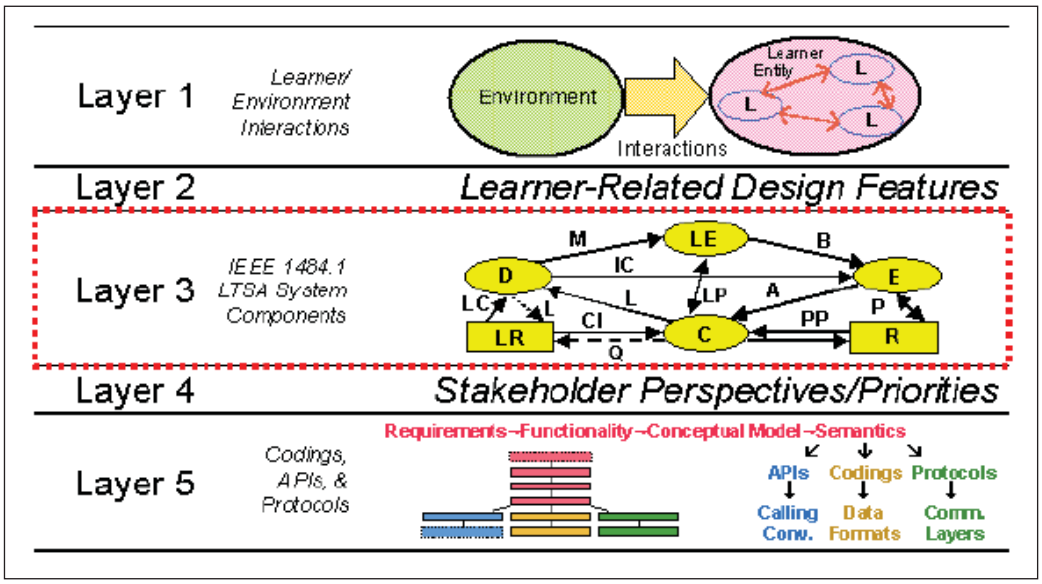

Рис. 2. П'ятирівнева модель системи дистанційного навчання по специфікації LTSA

Technology Standards Committe). Навчальні технології дистанційного навчання розвивають і вдосконалюють якість освіти, що здобувається, на основі методологічних стандартів університетської освіти, зберігаючи індивідуальність і авторитет (стиль навчання / рівень, що може прирівнюватись до бренду) вузу.

Стандартизація архітектури систем дистанційного навчання

Відповідно до специфікацій IMS для використання в освітній діяльності комітетом IEEE LTSC (P1484 – Learning Technology Standard Committee) рекомендована типова архітектура систем дистанційного навчання LTSA (Learning Technology Systems Architecture).

Стандарт IEEE P1484 охоплює достатньо широке коло систем зазвичай відомих як навчальні системи, тренінгові системи, системи комп'ютерного тестування, інтелектуальні навчальні системи. Стандарт є керівництвом для розвитку конфігурації інформаційних систем освітнього призначення, а також комунікаційних протоколів і методів взаємодії за спільної роботи в процесі навчання.

У системах, що реалізують технології навчання, виділяють п'ять рівнів опису архітектури (рис. 2) [1]:

Рівні опису системи дистанційного навчання:

Рівень 1. Взаємодія Студент– навколишнє середовище.

Рівень 2. Особливості проектування, пов'язані зі студентами.

Рівень 3. IEEE 1484.1, компоненти LTSA-системи.

Рівень 4. Перспективи і пріоритети учасників навчального процесу.

Рівень 5. Операційні компоненти і протоколи взаємодії.

Рівень 1. Виділяються два елементи – об'єкт навчання і середовище навчання. Розглядаються питання впливу середовища навчання на студента з позицій передачі знань, обміну інформацією за взаємодії з повчальним середовищем, представленим у вигляді Інтернету, лабораторії, комп'ютерів, бібліотек, книг, мультимедіа файлів, газет, телебачення, батьків, педагогів. Спільна робота студентів представляється у вигляді внутрішньої взаємодії, аналогічної взаємодії розподілених баз даних в процесі створення єдиної бази даних. Середовище навчання є чинником впливу на студента в процесі взаємодії.

Рівень 2. Формулюються завдання, пов'язані з особливостями інтерфейсу освітньої технологічної системи в процесі взаємодії зі студентами. Водночас акцентується увага на природі людини, що відрізняє її від комп'ютера.

Рівень 3. Найбільш інформативною частиною архітектури є 3-й рівень (компоненти системи), на якому аналізується система дистанційного навчання з позицій інформаційних технологій.

Рівень 4. Описуються інформаційні потоки між окремими компонентами системи дистанційного навчання залежно від моделей і технологій навчання. У LTSA Specification розглянуто 120 [1] варіантів архітектурних рішень, що служить, на думку авторів, доказом універсальності запропонованих архітектурних моделей.

Рівень 5. Забезпечує взаємодію системи дистанційного навчання, тобто описуються основні елементи, відповідальні за її взаємодію. Це насамперед інтерфейси прикладних програм (застосувань) – API (Application Program Interface), формати даних і протоколи обміну даними.

Кожна система дистанційного навчання, згідно зі специфікацією LTSA, має підтримувати чотири процеси, два сховища даних (репозиторію) і десять інформаційних потоків.

Процеси (Processes):

– виявлення знань студента (основним об'єктом цього процесу є студент (Learner entity);

– оцінка знань (Evaluation);

– координування – управління навчанням (основним об'єктом цього процесу є викладач-інструктор) (Coach);

– формування і доставка учбових матеріалів (процес доставки) (Delivery).

Сховища даних (Stores):

– записи успішності студента (база даних з результатами відповідей і успішності студентів) (Learner records);

– ресурси навчання (репозиторій з матеріалами для навчання) (Learning resources).

Інформаційні потоки між процесами і сховищами даних (Flows):

– спостереження за поведінкою учня (Behavior);

– інформація про тестування (Assessment);

– інформація про успішність (Preferences);

– запити (Query);

– інформаційні каталоги (Catalog Info);

– посилання (адреси) навчальних матеріалів (Locator);

– контент (навчальні матеріали) (Learning Content);

– мультимедіа (Multimedia);

– контекст взаємодії (Interaction Context);

– стилі, стратегії і методи навчання (Learning Preferences).

Процес навчання з використання СДН (Систем дистанційного навчання, LMS) включає наступні етапи [1]:

1. Стилі, стратегії і методи навчання обговорюються серед студентів і осіб, які зацікавлені в їх навчанні (керівники, спонсори, батьки та інші), і спільно виробляється найбільш переважна стратегія навчання.

2. Студент вивчає матеріал. В ході вивчення його знання постійно оцінюються автоматизованими засобами тестування з урахуванням персональних особливостей і вибраної стратегії навчання.

3. Після закінчення вивчення розділу або дисципліни знання студента оцінюються в процесі атестації або як сумарна тестова оцінка за виконання поточних завдань.

4. Інформація про виконання завдань запам'ятовується в базі даних (історія навчання).

5. Викладач-інструктор регулярно проглядає оцінки виконаних поточних завдань студентів, результати поточної атестації студентів, попередні атестації і коректує цілі навчання для кожного студента відповідно до його успіхів.

6. Викладач-інструктор досліджує ресурси навчання, набори тестів, питань й інформаційні каталоги і формує контент найбільш відповідний для конкретного студента.

7. Викладач-інструктор читає розділи навчальних матеріалів (використовуючи відповідні посилання) з інформаційних каталогів, визначає місцезнаходження вказаних ресурсів, передає інформацію про місце їх знаходження в службу доставки навчальних матеріалів і таким чином формує план навчання, орієнтований на конкретного студента.

8. Служба доставки дістає необхідний навчальний контент з БД за допомогою посилань і перетворює його в мультимедійну інтерактивну презентацію для конкретного студента.

На ринку представлені такі Системи дистанційного навчання, як WebCT, Learning Space, Lon-CAPA, Moodle та інші [8]. Вони відповідають вимогам стандартів, дозволяють створювати навчальні ресурси і контролювати успішність навчання.

Проте оцінювання тільки з боку LMS не достатні для оцінювання студента загалом, оскільки система враховує тільки активність в межах самої системи, не враховуючи оцінки за лабораторні роботи, контрольні тощо. Проте, якщо поєднати оцінки із системи та інші оцінки, можна отримати шлях до моніторингу успішності студента та якості начального матеріалу. Звичайно ж це потребуватиме деякого алгоритмічного підходу, який буде надавати послуги із побудови моделі успішності студента, проте для того щоб реалізувати цю систему, треба зробити деякі технічні рішення, необхідно спроектувати архітектуру, що буде містити в собі і LMS, і компоненти обробки даних та компоненти, що будуть організовувати процес обробки для даних.

Проте постає питання, яким чином отримувати генеровані дані, зберігати їх, аналізувати? Відповіддю на це питання буде система, що поєднає Big Streams, Big Data, Big Systems та логіку обробки.

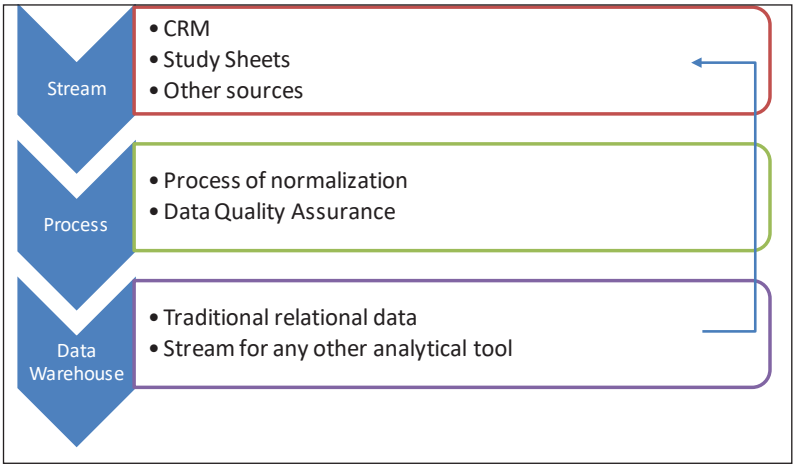

Рис. 3. Типовий процес обробки та аналізу даних

Результатом буде архітектура, що зможе поєднати велику кількість релевантних даних, зберігати їх та аналізувати. На перший погляд, задача потребує великих потужностей обчислювальної техніки, яких деякі наукові установи, на жаль, просто не мають. Проте можна використовувати технології масових паралельних обчислень, що дозволять застосувати ресурси комп'ютерного парку, що має установа, розподіляючи задачі за принципами навантаження.

Таким чином, маємо практичну інженерну задачу, яка потенційно дозволить покращити якість освіти та наукових досліджень за допомогою аналізу даних студента / курсу та виявлення слабких місць за допомогою алгоритмів аналізу. Вирішення цієї задачі полягає у створенні архітектурного рішення [2], що поєднає у собі обробку технології Big Data для аналізу і побудови моделей, що дозволить знаходити у курсах слабкі місця та поліпшувати їх. Для вирішення задачі необхідно обрати технології, що існують на ринку, сформувати технічну потребу для імплементації взаємодії обраних технологій, підключити як потік даних (Stream) базу даних присутньої LMS та написати клієнт-серверний додаток, що буде оброблювати дані, які були зібрані архітектурою. Надалі поверхнево розглянемо, як працюють елементи технологій, що будуть утворювати архітектурне рішення.

Основою для побудови процесу можуть стати технології та їх комбінації, що давно вже використовуються і перевірені часом у середовищі із високою конкуренцією – у бізнесі, тобто перевірені на стабільність за високих навантажень, де ціна помилки дуже висока.

Забезпечення ефективних інструментів та технологій бізнес-аналітики для підприємства є майже завжди пріоритетним напрямом. Ефективна бізнес-аналітика – від базової звітності до поглибленого аналізу даних та прогнозної аналітики – дозволяє аналітикам даних та бізнес-споживачам отримати висновки з корпоративних даних, які під час переведенні в дію забезпечують вищий рівень ефективності та прибутковості для підприємства.

Традиційні інструменти та технології керування даними та бізнес-аналітики змінюються під впливом Big Data, і тоді з'являються нові підходи, які допомагають користувачам отримати корисну інформацію з Big Data. Наприклад, фреймворк Hadoop, бази даних NoSQL, Cassandra, Accumulo і масово-паралельні аналітичні бази даних приймають принципово інший підхід до обробки даних, аналітики та Додатків, ніж традиційні інструменти та технології, від таких, як EMC Greenplum, HP Vertica та Teradata Aster Data. Це означає, що підприємства також повинні радикально переосмислити те, як вони підходять до аналізу даних бізнесу.

Застосування великих даних для більшості буде непростою задачею, проте компанії, що застосовують їх, стають більш конкуренто спроможними, а також зменшують час реакції на зміни, оскілки збирають та аналізують дані із декількох місць одночасно. Big Data у поєднанні зі складною бізнес-аналітикою призводить до розуміння поведінки клієнтів та нестабільних ринкових умов, що дає їм можливість робити бізнес-рішення на основі даних набагато швидше та ефективніше, ніж конкурентам.

Обробка та аналітика

Для виконання задачі із побудування інфраструктури оцінювання студента необхідно дослідити наявні продукти, що можуть забезпечити процес обробки потоку даних, у тому числі даних із властивостями історичних, поганою структуризацією, чіткою структуризацією логів і тому подібним. Тому розглянемо характерні особливості Big Data та пов'язаного з ними інструментарію.

Зазвичай для обробки даних в аналітичних цілях було достатньо використовувати наступний процес обробки. За звичайного процесу ведення бізнесу підприємства створюють достатньо скромні обсяги структурованих даних зі стабільними моделями даних через такі корпоративні програми, як CRM, ERP та фінансові системи. Інструменти інтеграції даних використовуються

для вилучення, перетворення та завантаження даних з корпоративних додатків та трансакційних баз даних у місце розташування, де відбувається нормалізація якості даних), а дані моделюються в структуровані рядки та таблиці. Модельовані, очищені дані потім завантажуються в корпоративне сховище даних. Ця процедура зазвичай відбувається на регулярній основі – щоденно або щотижня, іноді частіше.

Надалі адміністратори сховища даних створюють і запускають регулярні звіти, які працюють з нормалізованими даними, що зберігаються на сховищі даних. Вони також створюють інформаційні панелі та інші обмежені засоби візуалізації для користувачів.

Тим часом аналітики використовують інструменти аналізу даних, щоб запустити розширені аналітики зі сховища даних або проти зразкових даних, перенесених до локальних даних, через обмеження розміру. Неекспертні аналітики-користувачі виконують базову візуалізацію даних та обмежену аналітику порівняно зі сховищем даних через зовнішні інструменти бізнес-аналітики від постачальників, таких як SAP BusinessObjects та IBM Cognos. Обсяги даних в традиційних сховищах даних не часто перевищують кілька терабайт, оскільки великі обсяги даних займають ресурси сховища та знижують продуктивність.

Характер великих даних

Поява веб-сторінок, мобільних пристроїв та інших технологій призвела до істотних змін характеру даних. Великі дані мають важливі відмінні якості, які відрізняють їх від корпоративних «традиційних». Вона не централізована, не структурована, доволі поширена. Зокрема, вона має такі головні риси:

– об'єм – кількість даних, що створюється як всередині підприємства, так і поза локальною мережею – у веб, мобільних пристроях та IT інфраструктурах щороку збільшується експоненційно.

– тип – різноманіття типів даних збільшується, включає неструктуровані текстові дані та погано структуровані дані, як-то: дані соціальних платформ, місце розташування, логів тощо.

– швидкість – середня швидкість, за якої формується масив великих даних, та реальна потреба у аналітики реального часу для надання бізнесу цінної інформації із неї зростає завдяки кількості транзакцій та кількості пристроїв в цілому.

– Шляхи генерації Big Data:

– Соціальні мережі та медіа – сотні мільйонів сторінок у соціальних мережах, програмах обміну повідомленнями, блогах. Кожен пост у соціальній мережі, повідомлення чи оновлення блогу створюють «вихлоп даних» – погано структурованих чи не структурованих даних.

– Мобільні пристрої – на даний момент по всьому світу використовуються більше ніж 5 мільярдів пристроїв. Смартфони та планшеті, зокрема, надають мобільний доступ до соціальних мереж, а також збирають та передають дані місцезнаходження.

Традиційні сховища даних та інші інструменти керування даними під час обробки та аналізу великих даних не є ефективними в економічному та часовому аспекті. Зокрема, дані повинні бути організовані в реляційні таблиці – структуровані рядки та стовпці – до того, як традиційне корпоративне сховище даних може його отримати. Через кількість часу та надмірну потужність застосування такої структури до величезної кількості неструктурованих даних є непрактичним. Крім того, для розширення традиційного корпоративного сховища даних для розміщення потенційних петабайт даних можуть вимагатися нереалістичні фінансові інвестицій в нові, часто залежні від постачальника та обладнання. Також може постраждати продуктивність сховища даних через точку замикання для завантаження даних. Тому необхідні нові способи обробки та аналізу великих даних.

Таблиця 1

Порівняння Традиційних і не Традиційних баз даних

Традиційні дані	Big Data
Гігабайти	Петабайти та Екзабайти
Централізоване	Розподілений
Структурований	Напівструктурований та неструктурований характер даних
Стійка модель даних	Прості схеми
Складені взаємозв'язки	Майже відсутні взаємозв'язки

Підходи до обробки та аналітики

Існує ряд підходів до обробки та аналізу Big Data, але більшість поділяють деякі загальні характеристики, а саме: вони користуються перевагами товарного обладнання, щоб забезпечити масштабні технології паралельної обробки; використовують можливості не реляційних даних для обробки неструктурованих та нечітко структурованих даних; застосовують розширені аналітичні технології та технології візуалізації даних до великих даних для передачі статистики кін-

цевим користувачам. Надалі розглянемо технічні рішення, що присутні на ринку і покликані виконувати обробку даних.

Hadoop – відкритий фреймворк для обробки, зберігання та аналізу великої кількості неструктурованих, розподілених даних. Кластери Hadoop можуть працювати на недорогому фізичному обладнанні, що дозволить маленьким командам не збанкрутитися за раптового збільшення кластеру. Нині це продукт Apache Software Foundation, що означає, що продукт постійно оновлюється великою спільнотою. Фундаментальний концепт Hadoop – замість того щоб опрацьовувати за раз однією машиною один шматок даних, Hadoop розбиває дані на багато частин, щоб опрацьовувати і аналізувати одночасно.

Клієнт має доступ до даних, що не мають довіри, які являються слабко структурованими, наприклад, дані із соціальних мереж та внутрішніх сховищ. Система розбиває дані на частини, що потім завантажуються у файлову систему, що створена із багатьох нод.

Стандартний метод зберігання у Hadoop – Hadoop File System, що створена спеціально для того щоб зберігати великі об'єми неструктурованих даних, при цьому дані не зберігаються у реляційних форматах.

Кожна частина реплікується по декілька разів, а тільки потім завантажується у систему на випадок, якщо зв'язок із нодою буде втрачений. Головна нода або ж «іменна» (NameNode) працює як посередник, повідомляючи інформацію, як то: доступність вузлів, знаходження даних у кластері, ноди, що відмовили.

Як тільки дані були завантажені, можна починати процедуру MapReduce у відповідному фреймворку. Клієнт подає завдання "Map" – зазвичай запит, написаний на Java, на один із вузлів кластера, що називається Job Tracker. Відстеження роботи посилається на іменний вузол, щоб визначити, які дані йому необхідні для завершення роботи, а також, де знаходиться в кластері розташування даних. Після визначення, Job Tracker подає запит до відповідних вузлів. Замість того щоб повернути всі дані назад до центрального розташування, обробка відбувається на кожному вузлі паралельно. Це істотна характеристика Hadoop.

Коли кожен вузол закінчить обробку заданої роботи, він зберігає результати. Клієнт ініціює роботу "Reduce" через Job Tracker, в якій результати етапу Map зберігаються локалізовано на окремих вузлах, агрегуються, щоб визначити «відповідь» на початковий запит, а потім завантажується на інший вузол кластера.

Коли фаза Map-Reduce завершена, оброблені дані готові до подальшого аналізу з боку аналітиків та інших працівників, які мають навички аналізу даних. Аналітики можуть маніпулювати та аналізувати дані за допомогою будь-якої кількості інструментів для будь-якої кількості застосувань, у тому числі для пошуку прихованих фактів та моделей або для створення основи для створення аналітичних додатків, які використовуються користувачами. Дані також можна моделювати та переносити з кластерів Hadoop у наявні реляційні бази даних, сховища даних та інші традиційні інформаційні системи для подальшого аналізу та / або підтримки трансакційної обробки.

Компоненти Хадуп:

– Hadoop Distributed File System: HDFS – стандартне сховище у будь-якій реалізації кластера;

– Вузол Імен (NameNode) – нода у кластері, що надає клієнтові інформацію про розташування даних та працездатність вузлів;

– Job Tracker – вузол, що ініціює та координує MapReduce процеси, обробку даних;

– Slave Node – робочі ноди, що зберігають дані, та оброблюють їх, згідно Job Tracker.

На додаток до вищевказаного, екосистема Hadoop містить багато сумісних проектів. NoSQL Cassandra та HBase можуть зберігати результати MapReduce. Також присутній проект Hive, що дозволяє робити аналітичні моделі у Hadoop.

Аналіз необхідності застосування Hadoop

Головною перевагою Hadoop є те, що технологія дозволяє користувачам ефективно обробляти та аналізувати великі обсяги неструктурованих та нечітко структурованих даних, які до цього були недоступні для аналізу. Оскільки кластери Hadoop можуть масштабуватися до екзабайтів даних, підприємства більше не повинні спиратися на вибіркові набори даних, а можуть обробляти та аналізувати всі релевантні дані. Аналітики можуть застосовувати ітераційний підхід до аналізу, постійне вдосконалення та тестування запитів, щоб виявити раніше невідомі факти. Також певним плюсом є відносна мала ціна Hadoop у термінах ціни та інвестицій часу. Розробники можуть безкоштовно завантажити дистрибутив Apache Hadoop і почати експериментувати з ним менш ніж за день.

Недоліком є те, що все ще існує брак розробників Hadoop та аналітиків, а також доволі високі ціни на їх послуги, що робить обмеженим для багатьох підприємств застосування, підтримку та

використання великих кластерів Hadoop. Також Hadoop є пакетною системою, тобто не підтримує обробку та аналіз даних в реальному часі.

NoSQL

Новий формат зберігання у базі даних NoSQL (Not Only SQL) з'явився, подібно до Hadoop, для обробки великих обсягів багатоструктурних даних. Проте тоді як Hadoop чудово підтримує великомасштабний історичний аналіз пакетного стилю, база даних NoSQL спрямована здебільшого для надання дискретних даних для кінцевих користувачів та автоматизованих додатків великих даних, що зберігаються у великих обсягах багато-структурних даних. Ця можливість, на жаль, відсутня в технології реляційних баз даних, які просто не можуть підтримувати необхідний рівень продуктивності для додатків у масштабі Big Data.

У деяких випадках NoSQL та Hadoop працюють разом. Наприклад, HBase – це популярна база даних NoSQL, модельована за допомогою Google BigTable, яка часто використовується поверх HDFS (розподіленою файловою системою Hadoop), щоб забезпечити низький час затримки та швидкий пошук в Hadoop.

Нині наявні NoSQL БД:

– HBase;
– Cassandra;
– MarkLogic;
– Aerospike;
– MongoDB;
– Accumulo;
– Riak;
– CouchDB;
– DynamoDB

Недоліком більшості баз даних NoSQL є те, що вони знехтували принципом ACID (атомічність, послідовність, ізоляція, довговічність) для надбання продуктивності та масштабованості.

Масово-паралельні аналітичні бази даних.

Під час побудови системи також треба враховувати фактор внутрішніх порядків інституцій, де вона буде розгортатися, тобто тривалість часу зберігання даних. При цьому зберігати тимчасові файли немає жодної потреби, але файли, що містять лабораторні роботи, курсові роботи тощо – є обов'язковими для зберігання терміном мінімум у декілька років (наприклад у КПІ – три роки), тому необхідно застосувати систему, яка зможе легко бути реконфігурована таким чином, щоб зберігати дані у декількох окремих місцях. Цей інженерний прийом називається sharding (з англ. shard – частина) – процес, коли дані розподіляються за якоюсь ознакою по декількох серверах. Майже усі сучасні СУБД підтримують таку можливість, але найкраще ця технологія представлена у масово-паралельних аналітичних базах даних.

На відміну від традиційних сховищ даних, масово-паралельні аналітичні бази даних здатні швидко приймати великі обсяги структурованих даних з мінімальним моделюванням і можуть бути розширені для розміщення декількох петабайт, а іноді й екзабайт даних.

Що найголовніше для користувачів, масово-паралельні аналітичні бази даних підтримують результати майже в режимі реального часу для складних SQL-запитів (також називаються інтерактивними можливостями запитів) – помітна відсутність таких можливостей в Hadoop, а в деяких випадках – можливість підтримувати програми Big Data у режимі реального часу.

Наведемо основні характеристики масово-паралельної аналітичної бази.

Масово-паралельна обробка (MPP можливості). Як зазначено в назві, масово-паралельні аналітичні бази даних використовують масово-паралельну обробку або MPP, які одночасно підтримують прийом, обробку та запит даних на декількох машинах. Результатом є значно швидша продуктивність, ніж традиційні сховища даних, що працюють на одній великій коробці і обмежені однією точкою дроселя для зчитування даних.

«Shared-nothing» архітектура забезпечує відсутність єдиної точки відмови в деяких аналітичних середовищах бази даних. У цих випадках кожен вузол працює незалежно від інших, тому, якщо одна машина не працює, інші продовжують працювати. Це особливо важливо в середовищах MPP, в яких іноді сотні машин обробляють дані паралельно. Отже, випадковий збій однієї або більше машин неминучий.

Стовпчикові архітектури. Замість того, щоб зберігати та обробляти дані в рядках, як це характерно для більшості реляційних баз даних, в більшості масивних паралельних аналітичних баз даних використовуються стовпчикові архітектури. У стовпчикових середовищах обробляються лише стовпці, які містять необхідні дані для визначення «відповіді» на певний запит, а не цілі рядки даних, що приводить до результатів розділених запитів другого запиту. Це також означає, що дані не потрібно структурувати в акуратні таблиці, як у традиційних реляційних базах даних.

Розширені можливості стиснення даних. Вони дозволяють аналітичним базам даних приймати та зберігати більші обсяги даних, ніж це

Таблиця 2

Ринок продуктів, пов'язаних із Big Data[9]

Програмне забезпечення					
HADOOP	***NoSQL***	***NGDW***	***Аналітика***	***Додатки***	***Інструменти***
Hortonworks	DataStax	HP Vert.	Digital Reasoning	Google	Informatica
	Sprrl	EMC Greenpluum	Revolution Analytics	Tresata	talnd
Cloudera	Couchhase	IBM Netezza	Jaspersoft	Opera Solutions	Zettaset
MapR Hadapt	l0gen	SAP	Pentaho	DataXu	Syncrosoft
EMC Greenplum	Basho	Teradata aster	Datameer	SAP	Vmware
Сервіси					
Cloud Сервіси		***Технічні сервіси***		***Професійні сервіси***	
Amazon		Hortonworks		Thoink Big Analytics	
Google		Cloudera		IBM	
MapR		Cloudwick		EMC	
IBM		EMC		Accenture	
Microsoft		IBM		Deloitte	

можливо іншим способом, і робити це з значно меншими ресурсами обладнання, ніж традиційні бази даних. Наприклад, база даних з функцією стиснення «10-до-1» може стискати 10 терабайт даних до 1 терабайта. Стиснення даних та пов'язана з ними методика, що називається кодування даних, є критично важливими для ефективного масштабування масивних обсягів даних.

Апаратне забезпечення продукту. Як і кластери Hadoop, більшість, але не всі масово-паралельні аналітичні бази даних працюють на нестандартному обладнанні від Dell, IBM та інших, отже, вони можуть економічно розширюватись.

Обробка даних в пам'яті. Деякі масово-паралельні аналітичні бази даних використовують RAM для обробки деяких даних в реальному часі. Деякі, такі як SAP HANA і Aerospike, повністю вбудовані в пам'ять, тоді як інші використовують гібридний підхід, який поєднує менш дорогі, але менш ефективні на дисках накопичувачі для «холодніших» (менш використовуваних) даних з RAM для «більш гарячих» (використовуваних) даних.

Тим не менш, масово-паралельні аналітичні бази даних мають кілька слабких місць. Зокрема, вони не призначені для прийому, обробки та аналізу слабко структурованих і неструктурованих даних, які в основному відповідають за зріст популярності та розвиток Big Data.

Взаємодоповнюваність компонентів, пов'язаних із Big Data

Hadoop, NoSQL та масово-паралельні аналітичні бази даних не є взаємовиключними. Нині існують принаймні три популярні підходи, які є взаємодоповнюючими та можуть і мають співіснувати в багатьох підрозділах. Hadoop дуже добре підходить для обробки та аналізу великих об'ємів розподілених, неструктурованих даних в пакетному режимі в рамках історичного аналізу. Бази даних NoSQL допомагають зберігати та обслуговувати багатоструктурні дані в режимі реального часу для веб-додатків, пов'язаних із Big Data. Тоді як масово-паралельні аналітичні бази даних найкраще можуть забезпечити аналіз великих обсягів в основному структурованих даних у режимі майже реального часу.

Історичний аналіз, зроблений за допомогою Hadoop, може бути перенесений в аналітичні бази даних для подальшого аналізу або / та інтегрований зі структурованими даними на традиційних корпоративних сховищах даних. Наприклад, факти, отримані від аналізу Big Data, мають оброблятися у додатках Big Data. Користувач має прагнути до гнучких архітектур Big Data, щоб забезпечити ці три технології / підходи взаємодією та обміном даних.

Існує декілька коннекторів (поєднувачів), які призначені для легкої інтеграції технологічних рішень Big Data розробниками та адміністраторами із Hadoop. Водночас декілька постачальників, наприклад, Pivotal Initiative (раніше – EMC Greenplum, Cetas та ін.) та Teradata Aster пропонують рішення Big Data, що поєднують Hadoop і аналітичні бази даних із попередньо-налаштованим устаткуванням для швидкого розгортання з мінімальним необхідним налаштуванням. Також присутні технології на кшталт Hadapt, що пропонує єдину платформу, яка забезпечує обробку як SQL, так і Hadoop / MapReduce в одному кластері. Фірма Cloudera також переслідує цю стратегію проектами Impala та Hortonworks за допомогою ініціативи із відкритим сирцевим кодом Stinger Initiative.

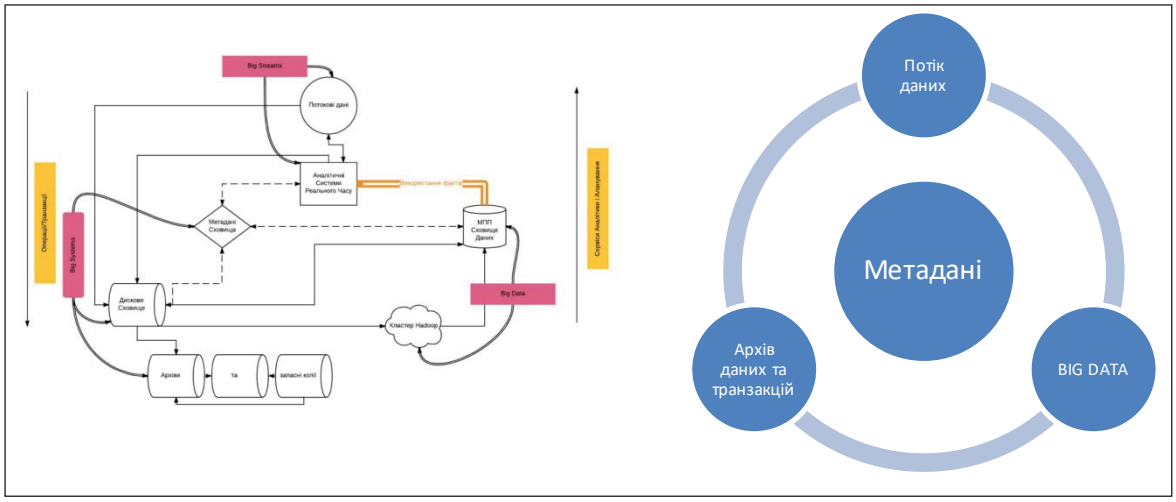

Рис. 4, 5. Процеси взаємодії Big Data [3, 6]

Приємною частиною є те, що підходи Big Data такі, як Hadoop, NoSQL та масово-паралельні аналітичні бази даних мають безкоштовні відкриті проекти, якими можна скористатись, проте у випадку MPP їх ефективність обмежена, що означає, що для реально великих об'ємів потрібне платне рішення. Проте у разі навчальних установ ці рішення часто є умовно-безкоштовними, наприклад IBM надає свій набір рішень DB2 warehouse навчальним установам безкоштовно.

Привабливість продуктів Big Data полягає в тому, що вони дозволяють підприємствам знаходити відповіді на питання, які навіть не поставали (в сенсі залежності між фактами, що породжує можливості для бізнесу і нові підходи у дослідах для науки). Це може призвести до розуміння, що призведе до нових ідей продуктів або допоможе визначити шляхи підвищення операційної ефективності. Тим не менш, існує вже цілий ряд ідентифікованих застосувань як для Big Data, Google, Facebook і LinkedIn, так і для більш традиційного підприємства. Вони включають, зокрема:

– дослідження. Такі підприємства, як науково-дослідницькі інститути та лабораторії використовують Hadoop для обробки великих обсягів даних текстових досліджень та інших історичних даних, щоб сприяти розробці нових продуктів;

– моніторинг мережі. Hadoop та інші технології Big Data використовуються для збору, аналізу та відображення даних, зібраних з серверів, пристроїв зберігання даних та інших пристроїв IT-обладнання, щоб дозволити адміністраторам здійснювати моніторинг роботи мережі та діагностувати вузькі місця та інші проблеми. Цей тип аналізу також можна застосувати до транспортних мереж з метою підвищення ефективності використання палива та інших мереж;

– моделювання ризику. Фінансові компанії, банки та інші підприємства використовують сховища даних Hadoop для аналізу великих обсягів транзакційних даних для визначення ризику та ризику фінансових активів, для підготовки до потенційних сценаріїв «що буде, якщо», заснованих на моделюванні ринкової поведінки, та для можливості врахування ризику потенційних клієнтів.

Побудова архітектури

Розглянувши у попередньому розділі наявні на ринку продукти, що забезпечують обробку великих даних, постає питання створення архітектурного рішення, яке покликане для зберігання та обробки даних, генератором якого є сам процес навчання. Основною проблемою під час створення таких рішень у рамках вітчизняного процесу навчання є лімітовані ресурси навчальних інституцій, оскільки сам процес обробки великих даних є доволі ресурсо-потребуючим. З цього факту постає пряма необхідність правильного вибору на користь тих продуктів, що будуть мінімально необхідними для побудови конвеєру обробки, проте будуть виконувати свої функції надійно та відносно швидко, але при цьому будуть мати можливість до вдосконалення та масштабування, що забезпечить максимально ефективне використання лімітованих ресурсів.

Для виконання цієї задачі насамперед треба порахувати можливий потік даних, що будуть зберігатися на носіях даних, для подальшої обробки.

За основу візьмемо процес навчання студентів кафедри ННК ІПСА «СП». У середньому у студентів є 4-5 предметів за один семестр навчання, що можуть бути вигідно покращені шляхом вико-

ристання систем дистанційного навчання для надання матеріалу, інтерактивів та інкорпорації матеріалу, що є продуктом надбання студентами практичних навичок, тобто лабораторних робіт, тестів, творчих завдань та презентацій. До того ж можна припустити, що загальний об'єм даних, що продукує один студент, в рамках одного семестру, за кількості лабораторних робіт, що в середньому становить 5-6 лабораторних робіт за семестр (за особистими спостереженнями авторів), та декількох творчих завдань, буде приблизно становити 60-100 Мб за семестр, не враховуючи об'єму даних, що генерується студентом під час сеансів роботи із системою. Водночас на одному потоці курсу ~70 студентів, отже, можна сміливо припустити, що навантаження на дискове сховище буде становити приблизно 100МБ*70+x*70, де X є об'ємом даних з одного потоку за семестр, що продукується самою системою. Попри це, статистичними даними є 500 Мб за один курс на одного студента [5]. При цьому необхідно буде враховувати «найгірший» випадок, коли даних багато, особливо при обмежених ресурсах. Тоді можливе навантаження на сховище буде буде $170Gb = \frac{70 * 500 * 5}{1024}$ за 1 семестр. Також дуже важливий характер даних. Оскілки переважна більшість лабораторних робіт являє собою комбінацію із масивів форматованого тексту та зображень, можна сказати, що дані мають приблизно однаковий характер. Попри те, треба врахувати додаткові фактори у вигляді даних із форумів, логів системи [2, 7], графів успішності студентів. Отже, можна говорити про слабко структурований характер даних, що становитимуть собою комбінацію із binary file, csv, метаданих [7] та текстових масивів. Проте метадані надаються LMS у вигляді опису «власника» даних, яким є користувач, що репрезентує студента, що суттєво полегшує роботу алгоритму, оскільки не треба встановлювати власника даних під час проведення аналізу.

Також треба врахувати, що важливим моментом у процесі навчання є контроль якості, найпростішим, але найважливішим аспектом якого є контроль плагіату. Реалізувати його можна, використовуючи third-party сервіси, що надають API для користування, проте цей компонент дозволить впливати на розмір навантаження на систему шляхом заборони завантаження файлу на сховище системи залежно від припустимого порогового значення.

Враховуючи огляд існуючих продуктів, що вказаний вище, пропонується побудувати datalake (масив даних та механізму обробки) для обробки даних на базі:

1. MariaDb, реляційна СУБД, що покликана обслуговувати обрану LMS та сховища історичних даних;

2. Apache spark, Hadoop як один із найпростіших та найпопулярніших продуктів;

3. MongoDB Nosql для забезпечення обміну статистикою між LMS та статистикою додатку оцінювання студента;

Вказані компоненти дозволять швидко та ефективно побудувати даталейк для обробки даних, а також із легкістю контейнеризувати, що забезпечить використання сучасної архітектури, що дозволяє легке оновлення компонентів та масштабування on demand.

Висновки. В роботі розглянуто технології та інструментальні засоби, які пропонуються для вдосконалення систем дистанційного навчання і дозволяють трансформувати традиційні технології навчання у змішані (mixed). Створення аналітичної підсистеми в складі LMS, в якій використовуються розглянуті технології обробки великих даних, дозволить підвищити ефективність використання дистанційного складника змішаного навчання і, як результат, якість отриманих знань. Для побудови LMS пропонуються програмні засоби, що є умовно безкоштовними, а також не потребують великих потужностей для інсталяції і обслуговування.

Список літератури:

1. Сучасний стан і світові тенденції розвитку дистанційної освіти, П.М. Таланчук, Г.Д. Киселев та ін., Київ 2010, Розділ 5. 470 стр.
2. "Evaluating Predictive Models of Student Success: Closing the Methodological Gap, arXiv: 1801.08494v2, 2018, 29 стр.
3. URL: http://www.dataversity.net/disrupting-metadata-management-metadata-automation/ (дата звернення 26.07.2018)
4. URL: https://www.nature.com/articles/s41539-017-0006-5 (дата звернення 26.07.2018)
5. URL: https://community.canvaslms.com/docs/DOC-10803-421473693 (дата звернення 26.07.2018)
6. URL: https://files.eric.ed.gov/fulltext/ED562284.pdf (дата звернення 26.07.2018)
7. URL: http://jasonpriem.org/self-archived/data-for-free.pdf (дата звернення 26.07.2018)
8. URL: https://www.researchandmarkets.com/research/v9bd9l/global_corporate?w=5 (дата звернення 26.07.2018)
9. URL: https://www.researchandmarkets.com/research/dkgxjz/global_big_data?w=4 (дата звернення 26.07.2018)

ИНФРАСТРУКТУРА ПЛАТФОРМЫ ЭЛЕКТРОННОЙ ПОДДЕРЖКИ УЧЕБНОГО ПРОЦЕССА

Дистанционное обучение, как и любая другая технология образования или повышения квалификации, содержит три составляющие: организация процесса обучения, учебный контент, мониторинг качества образования. Таким образом, система дистанционного обучения или Learning Management System (LMS) – это совокупность приложений, которые автоматизируют все эти составляющие дистанционного образования. В статье рассмотрены важность применения LMS рядом с традиционным процессом обучения. Рассмотрены стандартизации LMS, возможность и необходимость имплементации технологии Big Data, а также технологический процесс дистанционного обучения. Рассмотрен стек технологий, необходимый для выстраивания средств обработки данных в системе, а также их возможности к оказанию результирующих данных для мониторинга знаний студентов дистанционного образования. Рассмотрены сильные и слабые места технологий обработки и хранения данных. Указано на необходимость применения в LMS методов анализа больших массивов слабо структурированных данных и построения архитектуры полного анализа данных. Приведены примеры применения технологии Big Data в целом и примеры продуктов семейства Big Data, которые могут быть применены для построения аналитической составляющей в LMS.

Ключевые слова: *LMS, Big Data, Hadoop, NoSQL, Data Lake, Map Reduce, Framework, смешанное обучение, традиционные данные, учебный процесс.*

THE INFRASTRUCTURE OF THE E-LEARNING PLATFORM FOR THE LEARNING PROCESS

The article shows the importance of using LMS in the traditional learning process. The standardization of LMS as well as the possibility of implementing Big Data technology to LMS is exemplified, as well as the scheme of learning process for LMS. The interaction of the components of the system and the processing of the data it produces is demonstrated. The technologies stack that is necessary for the construction of the data processing system, as well as its capabilities, to provide the resulting data for the evaluation system, is demonstrated. The article describes strengths and weaknesses of processing and data storage technology. The need to use LMS to improve the learning process and the ease of use for analyzing large arrays of poorly structured data, as well as the need to build a complete data analysis architecture, also highlights the work of Big Data, as well as examples of the use of technology as a whole and examples of products representing the Big Data family that can be used to build the analytical platform for the LMS data analysis system is demonstrated, as well as the products that are worthy to start construction of own analytical platform for LMS data analysis and evaluation. The author exemplifies other usages of Big Data technology, that proved worthy in another spheres of IT.

Key words: *Learning Management System, Big Data, Hadoop, NoSql, Data Lake, Map Reduce, Framework, Mixed Learning, traditional data, studying process.*