

## **РОЗРОБКА ТЕХНІЧНИХ РІШЕНЬ ПОБУДОВИ ЄДИНОЇ БАЗИ ДАНИХ ТЕРИТОРІАЛЬНО РОЗПОДІЛЕНИХ ІНФРАСТРУКТУР**

*У статті розглянута структура зберігання даних, що дозволяє використовувати розподілене по різних комп'ютерах у мережі сховище даних, не прив'язуючись до конкретного місцезнаходження окремих файлів. Розроблена інфраструктура є простою у користуванні завдяки зрозумілому інтерфейсу користувача і, таким чином, поліпшує доступ до потужної розподіленої структури даних.*

*Запропоновані шляхи рівноправного децентралізованого об'єднання зацікавлених організацій для спільного використання ресурсів даних з метою розробки технічних рішень єдиної бази даних територіально розподілених інфраструктур.*

*Ключові слова: база даних, мережа, комп'ютерні системи, інфраструктура.*

**Вступ.** Бурхливий розвиток інформаційних технологій значно розширив можливості сучасної науки вцілому. Так, сучасний дослідник, окрім класичних підходів теоретичного та експериментального, отримав ще один принципово новий метод досліджень чисельне моделювання досліджуваних процесів та явищ.

Широке застосування моделювання в науці та на виробництві призвело також і до розвитку галузі паралельних обчислень. Паралельні комп'ютерні системи перестали бути рідкістю, величезні суперкомп'ютери змінили відносно дешеві обчислювальні кластери, що спричинило їх широке розповсюдження в світі.

Така тенденція спостерігається і в Україні.

В межах одного кластера на сьогодні проблему ефективного розподілу ресурсів між користувачами та їх програмами можна вважати вирішеною - створено програмне забезпечення, що відповідає визнаним стандартам.

Далі виникла ідея розподілу завдань та ресурсів між кластерами.

**Виклад основного матеріалу.** Цього можна досягти, наприклад, шляхом рівноправного децентралізованого об'єднання зацікавлених організацій для спільного використання ресурсів.

Ця ідея знайшла втілення у технології грид-систем.

Грид - це географічно розподілена інфраструктура, що надає можливість об'єднати існуючі в країні комп'ютери в єдиний величезний і могутній комп'ютер, шляхом об'єднання ресурсів різних типів - процесорів, довготривалої і оперативної пам'яті, сховищ і баз даних, доступ до яких користувач може отримати з будь-якої точки незалежно від місця свого розташування.

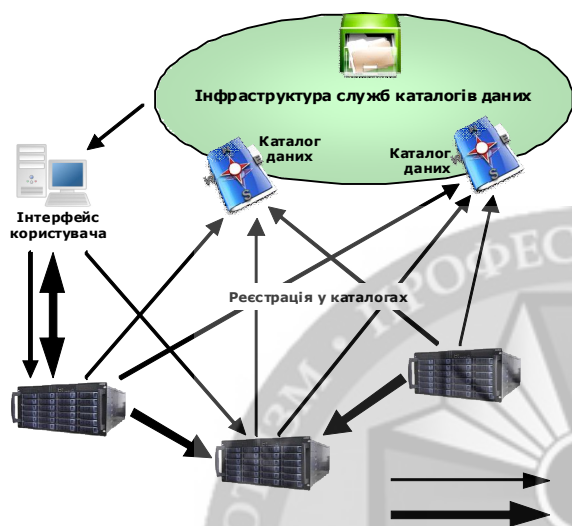


Рис. 1. Інфраструктура зберігання даних

До складу пакету проміжного програмного забезпечення ARC входить служба зберігання даних Smart Storage Element — SSE. Цей засіб розроблявся як автономна служба, що реалізує основні функції керування даними без втручання користувача.

За задумом розробників, SSE має функціонувати як складова частина загальної системи зберігання даних. Базуючись на SSE, можна побудувати інфраструктуру зберігання даних із досить широкими можливостями (рис.1).

Склад такої інфраструктури доповнюють служби каталогів даних. Каталоги даних можуть бути як загального користування, так і специфічними для певного застосування. Модульна архітектура дозволяє створювати інтерфейси до нестандартних чи невідтримуваних раніше каталогів даних.

Сам SSE не має внутрішньої структури для зберігання одиниць інформації. Файли розрізняються за ознаками, що використовуються у каталогу даних — за логічним іменем, шляхом, ідентифікатором GUID, тощо.

Всі операції над файлами (створення, реплікація, видалення репліки, тощо) виконуються на запит клієнта через SSE. Таким чином з точки зору клієнта вони є атомарними. Засоби роботи із службами каталогів клієнт повинен використовувати лише для операцій, у яких не задіяні конкретні одиниці інформації.

Контроль доступу до даних, що розмішені на SSE, здійснюється із застосуванням стандартного механізму аутентифікації користувачів Грид – посередницьких сертифікатів, та загальних списків контролю доступу (Generic Access Control Lists — GACL), що забезпечує значну гнучкість при визначенні політик.

Для передачі даних в SSE використовуються два протоколи — стандартний захищений протокол Інтернет HTTPS та його варіант із підтримкою Грид-аутентифікації HTTPg. Передача інформації між двома SSE здійснюється за запитом клієнта самими SSE. Це — так звана тристороння передача, що працює аналогічно із протоколом FTP.

Життєвий цикл одиниці інформації на SSE можна поділити на декілька стадій, які, в свою чергу, відносяться до двох паралельних процесів — наповненням вмісту одиниці та публікації її у каталогах даних (рис.2).

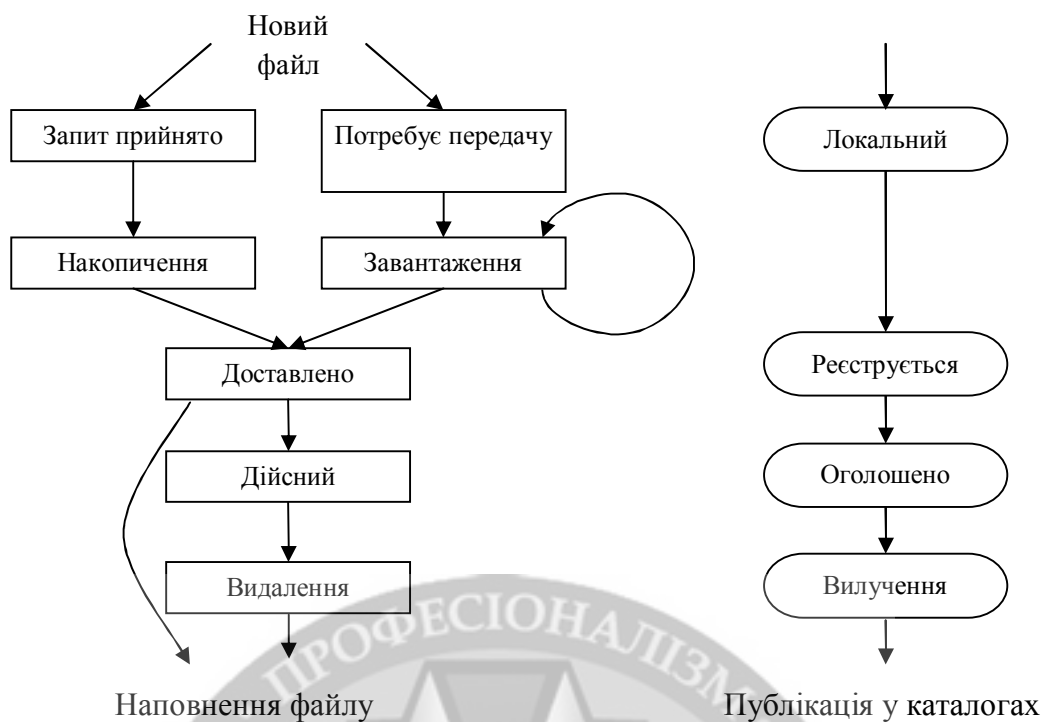


Рис. 2. Схема життєвого циклу одиниці інформації на SSE

Коли клієнт надсилає до SSE запит на створення файлу, під нього виділяється місце та заносяться мета-дані, після чого файл переходить у стан «**Запит прийнято**». У цьому стані завантаження змісту новоствореного файлу може розпочати будь-який клієнт, що пред'явить той самий сертифікат, що й клієнт-ініціатор. З моменту початку завантаження файл переходить стан «**Накопичення**». Після отримання службою останньої порції даних, SSE переводить його у стан «**Доставлено**».

Якщо ж служба отримала запит на завантаження файлу із іншого сховища, то при створенні він отримує стан «**Потребує передачу**». Цей стан відрізняється від стану «Запит прийнято» лише тим, що клієнтом надаються адреси сховищ-джерел, із яких слід отримати інформацію. Їх може бути декілька, у цьому випадку SSE вважає, що на кожному з них містяться ідентичні дані та завантажить файл із того джерела, що буде доступним на той момент. Якщо ж клієнтом не було надано жодного джерела, то SSE звернеться до каталогів даних для її отримання. Коли SSE остаточно визначився з адресою, він починає завантаження даних із неї — файл переходить у стан «**Завантаження**», а після завершення процесу — у стан «**Доставлено**». SSE може здійснити декілька спроб завантажити файл, якщо це не вдалося з першого разу.

Як тільки файл доставлено, перевіряється його достовірність шляхом порівняння контрольної суми із тою, що міститься у мета-даних. За вдалого результату файл отримує стан «**Дійсний**» і після цього може бути завантажений іншими клієнтами даного SSE за наявності відповідних повноважень. Це — нормальний стан для одиниць інформації, що містяться на SSE. Інакше файл та його мета-дані одразу ж видаляються.

При отриманні запиту на видалення, файл переводиться у стан «**Видалення**», а потім зникає із списку файлів.

Кожен SSE може бути налаштований на певний каталог даних. Відомості про кожен файл, що розміщується на SSE, разом із мета-даними надаються каталогу даних. До мета-даних відносять логічне ім'я, розмір, час створення та контрольну суму. Служба слідкує за станом реєстрації одиниці даних у каталозі одночасно із станом наповнення файлу.

До початку прийому даних файлу, про його існування відомо лише SSE — він має стан «**Локальний**». Після закінчення доставки файлу та його мета-даних, служба починає спроби його реєстрації у каталозі даних — файл отримує стан «**Реєструється**». За вдалої

спроби реєстрації відомості про файл стають доступними через каталог даних, а файл отримує стан «*Оголошено*». При видаленні файлу відомості про місцезнаходження цієї конкретної копії (репліки) вилучаються із каталогу даних — це відбувається у стані «*Вилучення*».

Серверні компоненти ARC реалізовані як модулі-розширення до демонів-мультиплексорів GridFTP та HTTPSD. Служба SSE, відповідно, є модулем HTTPSD. Таким чином її конфігурація визначається у секціях [httpsd] та [httpsd/se] загального файлу конфігурації `arc.conf`.

```
# Налаштування сервера HTTPSD
[httpsd]

# налаштування журналу
logfile="/var/log/httpsd.log"
logsize="100000 2"

# gsiport=port - порт для прийому з'єднань по HTTPg (варіант
Globus XIO)
gsiport="8000"

# gssapiport=port - порт для прийому з'єднань по HTTPg
(CERNівський варіант)
gssapiport="8002"

# sslport=port - порт для прийому з'єднань по HTTPS
sslport="8001"

# plainport=port - порт для прийому з'єднань по HTTP
plainport="8008"

# plugin=path - вказує модулі-розширення. Допускається декілька
разів.
plugin="/opt/nordugrid/lib/se.so"
. . .

# Налаштування модуля SSE
[httpsd/se]

# path=url_path - адреса для доступу ззовні (частина URL після
імені сервера)
path="/se"

# acl_create=group [group ...] - групи, кому дозволено створювати
файли
acl_create="users"

# групи, кому дозволено завантажувати без перевірки доступу.
acl_read="admins"

# storage=path - директорія локальної файлової системи, де
зберігати файли.
storage="/home/nordugrid/SSE"

# ns=configuration - адреса служби каталогів даних, настройка
реєстрації
ns="lrc://alien.univ.kiev.ua:39281"
registration="immediate,retry,showincomplete"
```

Служба SSE при доступі до каталогів даних може використовувати протоколи стандартних служб інструментарію Globus — каталогу реплік (Replica Catalog – RC), служби визначення місцезнаходження реплік (Replica Location Service – RLS) та локального каталогу реплік (Local Replica Catalog – LRC).

Оскільки служба RC вже застаріла, а підтримка RLS знаходиться у стадії розробки, то для налаштування каталогу даних було обрано протокол LRC. У складі інструментарію Globus, що розповсюджується разом із пакетом ARC, міститься сервер RLS/LRC. Для зберігання мета-даних він використовує відкритий інтерфейс доступу до баз даних (Open DataBase Connectivity – ODBC) засобами бібліотек iODBC чи unixODBC.

Конфігурація служби RLS/LRC задається у файлі **globus-rls-server.conf**.

```
# Параметри доступу до бази даних через ODBC
db_user          globus
db_pwd           globus
odbcini          /opt/globus/etc/odbc.ini
# адреса служби
myurl            rls://alien.univ.kiev.ua:39281

# негайне оновлення інформації в RLI із LRC
update_immediate true

# Опції LRC
lrc_server       true
lrc_dbname       globus_lrc
# Опції RLI
rli_server       true
rli_dbname       globus_rli
# Налаштування з'єднання
maxconnections   800
maxfreethreads   10
maxthreads       100
port              39281
```

Як систему керування базами даних було обрано MySQL, оскільки це найбільш проста у налаштуванні система, що має драйвер для ODBC. Налаштування ODBC задається у файлі **odbc.ini**. Обліковий запис для RLS/LRC потрібно створити заздалегідь засобами MySQL.

```
[ODBC Data Sources]
globus_lrc = MySQL LRC Database
globus_rli = MySQL RLI Database

[globus_lrc]
Driver      = /usr/lib/libmyodbc3_r.so
Socket      = /var/lib/mysql/mysql.sock
User        = globus
Password    =
Database    = globus_lrc

[globus_rli]
Driver      = /usr/lib/libmyodbc3_r.so
Description = RLI Database
Socket      = /var/lib/mysql/mysql.sock
User        = globus
Password    =
Database    = globus_rli
```

Після того, як служби RLS/LRC та HTTPSD запустились, система готова до використання у складі ARC. Доступ до сховища можна здійснювати за допомогою стандартних засобів інтерфейсу користувача ARC та Globus. До них належать такі утиліти командного рядка:

- **ngls** – отримання списку наявних файлів у каталозі чи на сховищі;

- **ngcp** – завантаження файлу на сховище;
- **ngrm** – видалення файлу;
- **ngtransfer** – формування запиту на тристоронню передачу із участю сховищ;
- **globus-url-copy** – аналогічний засіб інструментарію Globus;
- **globus-rls-cli** – отримання детальних відомостей про файли, що зареєстровані у каталозі.

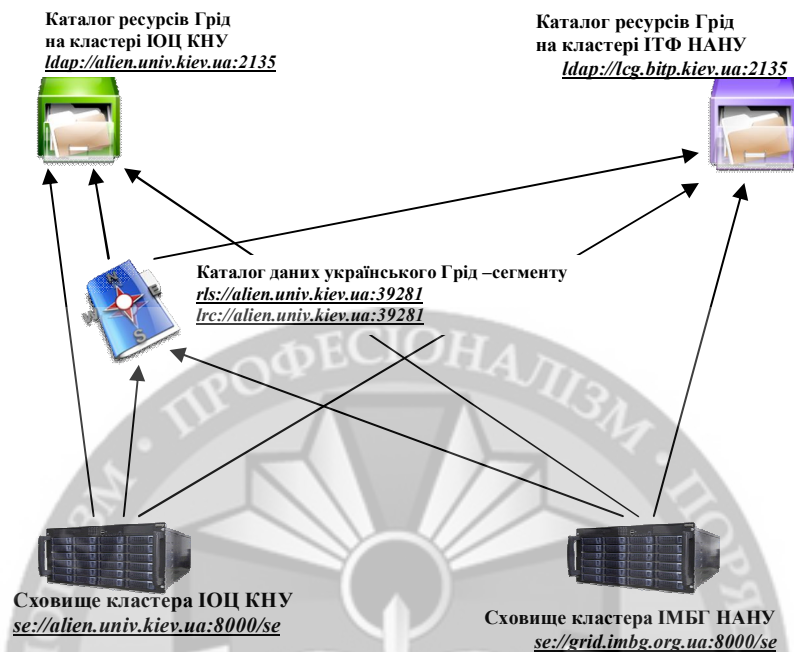


Рис. 3. Тестова інфраструктура зберігання даних українського Грід –сегменту

Також до сховища можна звертатись безпосередньо за допомогою Web-браузера за протоколом HTTPS, попередньо імпортувавши до нього сертифікат користувача.

В українському Грід –сегменті було успішно створено та протестовано інфраструктуру зберігання даних на базі ARC SSE та Globus RLS/LRC (рис. 3).

**Висновки.** Таким чином проаналізувавши все наведене вище можна зробити наступні висновки:

Розглянута структура зберігання даних дозволяє використовувати розподілене по різних комп'ютерах у мережі сховище даних, не прив'язуючись до конкретного місцезнаходження окремих файлів.

Розроблена інфраструктура є простою у користуванні завдяки зрозумілому інтерфейсу користувача і, таким чином, поліпшує доступ до потужної розподіленої структури даних.

#### ЛІТЕРАТУРА:

1. Швецов В.И. Базы данных. Учебное пособие /В.И. Швецов, А.Н. Визгунов, И.Б. Мееров. – Нижний Новгород: Изд-во ННГУ, 2004. – 217 с.
2. Марков А.С. Базы данных. Введение в теорию и методологию: Учебник /А.С. Марков, К.Ю. Лисовский – М.: Финансы и статистика, 2006. – 512 с.
3. Ситник В. Ф. та ін. Основи інформаційних систем: Навч. посібник. - Вид. 2-ге, перероб. і доп. / В. Ф. Ситник, Т. А. Писаревська, Н. В. Єрьоміна, О. С. Краєва; За ред. В. Ф. Ситника. - К.: КНЕУ, 2001. - 420с.
4. Стратегія формування та розвитку єдиного інформаційного простору України Сторінка доступу: [http://www.pravo.vuzlib.org/book\\_z1189\\_page\\_15.html](http://www.pravo.vuzlib.org/book_z1189_page_15.html)
5. Рейли Д. Создание приложений Microsoft ASP.NET - М.: Издательско-торговый дом "Русская редакция", 2002

**Рецензент: д.техн.н., проф. Ленков С.В.,** начальник науково-дослідного центру Військового інституту Київського національного університету імені Тараса Шевченка

к.т.н. Охрамович М.М.

## РОЗРОБКА ТЕХНІЧНИХ РІШЕНЬ ЄДИНОЇ БАЗИ ДАНИХ ТЕРИТОРІАЛЬНО РОЗПОДІЛЕНИХ ІНФРАСТРУКТУР

*В статье рассмотренная структура хранения данных, которая позволяет использовать распределенное по разным компьютерам в сети хранилище данных, не привязывая к конкретному местонахождению отдельных файлов. Разработанная инфраструктура простая в пользовании благодаря понятному интерфейсу пользователя и, таким образом, улучшает доступ к мощной распределенной структуре данных.*

*Предложены пути равноправного децентрализованного объединения заинтересованных организаций для общего использования ресурсов данных с целью разработки технических решений единой базы данных территориально распределенных инфраструктур.*

*Ключевые слова: база данных, сеть, компьютерные системы, инфраструктура*

Ph.D. Ohrmovych M.

## DEVELOPMENT OF TECHNICAL SOLUTIONS UNIFIED DATABASE OF GEOGRAPHICALLY DISTRIBUTED INFRASTRUCTURE

*In article rassmotrennaya storage of the structure of data, kotoraja pozvoljaet Require raspredelennoe to raznym computer storage of sets of data to be rolled pryvyazывaya particular mestonahozhdenyyu otdelnyh files. Razrabotannaya Infrastruktura Simple Thanks in polzovanyy Users and understandable interface, this way, uluchshaet Access to moschnoy raspredelenny Structure data.*

*Predlozheny path ravnopravnoho detsentralyzovannoho Combining zaynteresovannyh organizations for obshchego Data Using resources with a view tehnycheskyh Creative solutions edynoy bazy terrytoryalno raspredelennyh Data Infrastruktura.*

*Keywords: database, network, computer systems, Infrastructure.*