# Assignment 2

## Assignment description

Individually or in a group of at most two, create a distributed heterogeneous database environment comprising three sites with three different participating database platforms, and at least two different operating systems.  
  
Use the above environment to demonstrate your grasp of fragmentation and reconstruction. Think of a domain area comprising of at least four distributed relations. Write out your applications/reports and use them to perform fragmentation. Write out the applications/reports into calculus queries. Come up with appropriate query access frequencies of your choice. After working out the fragmentation, allocate the fragments by implementing them physically in the participating databases. Choose one of the sites to be the decision site and perform reconstruction using either views, functions, stored procedures or any other technique.  
  
Submission outputs

* A video recording of at most seven (7 minutes). First few seconds of video to have your registration number(s) displayed
* A pdf document summarizing the work. Include your registration number(s) on the first page

## Group members

1. Wisdom Makokha – P15/81777/2017
2. Joel Ndeda - P15/103081/2017

## Project process

Containers were used to simulate the operating systems and to satisfy the capability of the databases running on different operating systems.

A container provides an isolated, consistent environment that can mimic the behavior of running applications on separate OS instances.

Containers also make the installation and running of the targeted

So we used containers to simulate the environment of the different operating systems.

### Container setup

The technologies used for the system were

1. MySQL used on site 1.
2. Postgres SQL used on site 2.
3. SQL Server used on site 3.

MySQL server container setup command:



Postgres server container setup command



SQL Server container setup command:



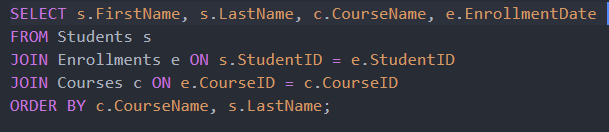
### Database design

|  |  |  |
| --- | --- | --- |
| TABLE\_NAME | COLUMN\_NAME | DATA\_TYPE |
| Courses | CourseID | int |
|  | CourseName | varchar |
| Department | varchar |
| Credits | int |
| Enrollments | EnrollmentID | int |
|  | StudentID | int |
| CourseID | int |
| EnrollmentDate | date |
| Grade | varchar |
| Professors | ProfessorID | int |
|  | FirstName | varchar |
| LastName | varchar |
| Department | varchar |
| Students | StudentID | int |
|  | FirstName | varchar |
| LastName | varchar |
| DateOfBirth | date |
| Gender | varchar |
| Major | varchar |

### Applications / reports

The applications, their equivalent calculus query and a sample of the results from each report.

1. Generate a report of all students enrolled in each course



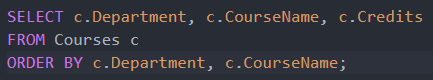


Fragmentation.

Horizontal fragmentation of Courses table by department.

Vertical fragmentation of Students table to separate student enrollment details from student personal details

1. List all courses offered by each department, including course names and credits



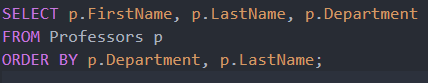


Fragmentation.

Horizontal fragmentation of courses table by department

Vertical fragmentation of courses table to separate credits details from other course details.

1. Generate a report of all professors and their respective departments

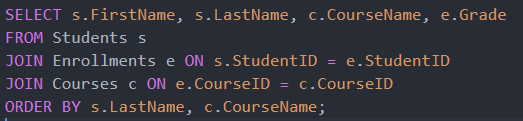




Fragmentation.

Vertical fragmentation of professors table to separate professor information from departmental information

1. Generate a report of student grades in each course





Fragmentation.

Vertical fragmentation of student table to separate enrollment details from personal student details.

### Access frequencies

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Report 1 | Report 2 | Report 3 | Report 4 |
| Site 1(MySQL) | 40 | 10 | 25 | 25 |
| Site 2 (SQL Server) | 15 | 50 | 15 | 20 |
| Site 3 (Postgres) | 40 | 10 | 10 | 40 |

Site 1

* Horizontal fragment students by their major for a specific group of majors - PHF
* Store enrollment dates - VF
* Store basic professor details – VF
* Store course credits - VF

Site 2

* Fragments from the course table separating the table horizontally by department – PHF
* General student info
* Store professor department information - VF

Site 3

* Horizontal fragmentation of students by major for a specific group of majors– PHF
* Horizontal fragmentation of enrollments by major - PHF

### Actual Fragment distribution

|  |  |
| --- | --- |
| Site | Fragments |
| Site 1 | F1 Student Major Science |
| F2 Course Credits |
| F3 Enrollment Dates |
| F4 Professor Basic Details |
| Site 2 | F1 Student General Info |
| F2 Courses Computer Science |
| F3 Courses Mathematics |
| F4 Courses Physics |
| F5 Courses Biology |
| F6 Courses Chemistry |
| F7 Professors Department |
| Site 3 | F1 Student Computer Math |
| F2 Science Enrollments |
| F3 Enrollment Computer Math |

### Decision site

Site 1 was selected as the decision site to perform reconstruction. Reconstruction will be performed programmatically to have more control over the result and the process