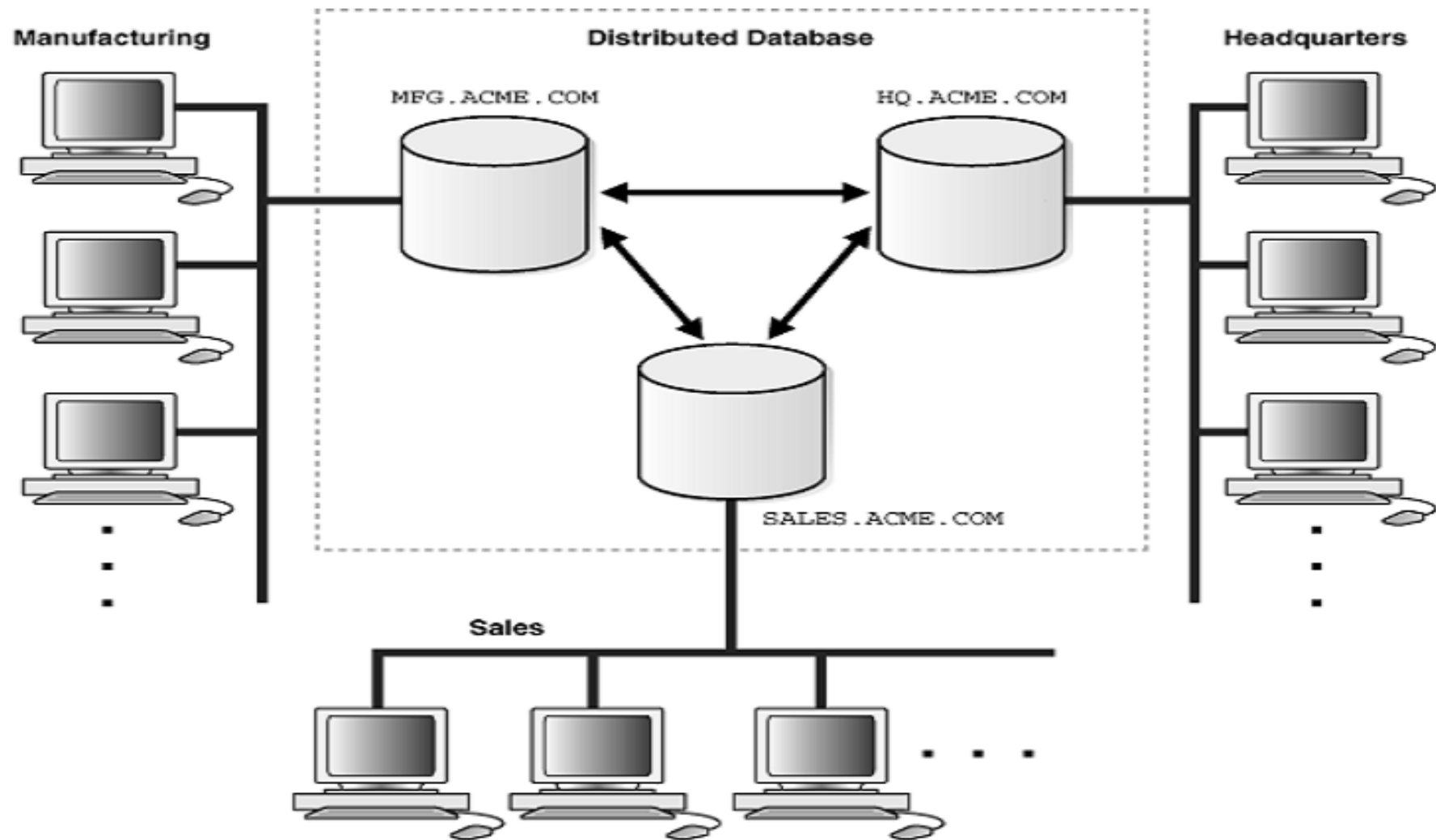

Chương IX:

CSDL PHÂN TÁN

I. Giới thiệu

- CSDL phân tán (**distributed database**) là một tập hợp nhiều CSDL có liên quan đến nhau **về mặt logic thuộc cùng một hệ thống** nhưng **về mặt vật lý được phân bố trên các trạm khác nhau** của một mạng máy tính.
- **Sự phân tán dữ liệu** (data distribution): dữ liệu phải được phân tán ở nhiều nơi.
- **Sự tương quan luận lý** (logical correlation): dữ liệu của các nơi được sử dụng chung để cùng giải quyết một vấn đề.

Ví dụ một CSDL phân tán



Hệ quản trị CSDL phân tán

■ Chức năng

- ❖ Hỗ trợ việc tạo và bảo trì cơ sở dữ liệu phân tán
- ❖ Có các thành phần tương tự như một hệ quản trị cơ sở dữ liệu tập trung
- ❖ Các thành phần hỗ trợ trong việc chuyển tải dữ liệu đến các trạm và ngược lại.

■ Thành phần của DDBMS:

- ❖ Quản trị dữ liệu (Database management): **DB**
 - ❖ Truyền thông dữ liệu (Data Communication): **DC**
 - ❖ Từ điển dữ liệu (Data Dictionary): **DD** dùng để mô tả thông tin về sự phân tán của dữ liệu trên mạng.
 - ❖ Cơ sở dữ liệu phân tán (Distributed Database): **DDB**
-

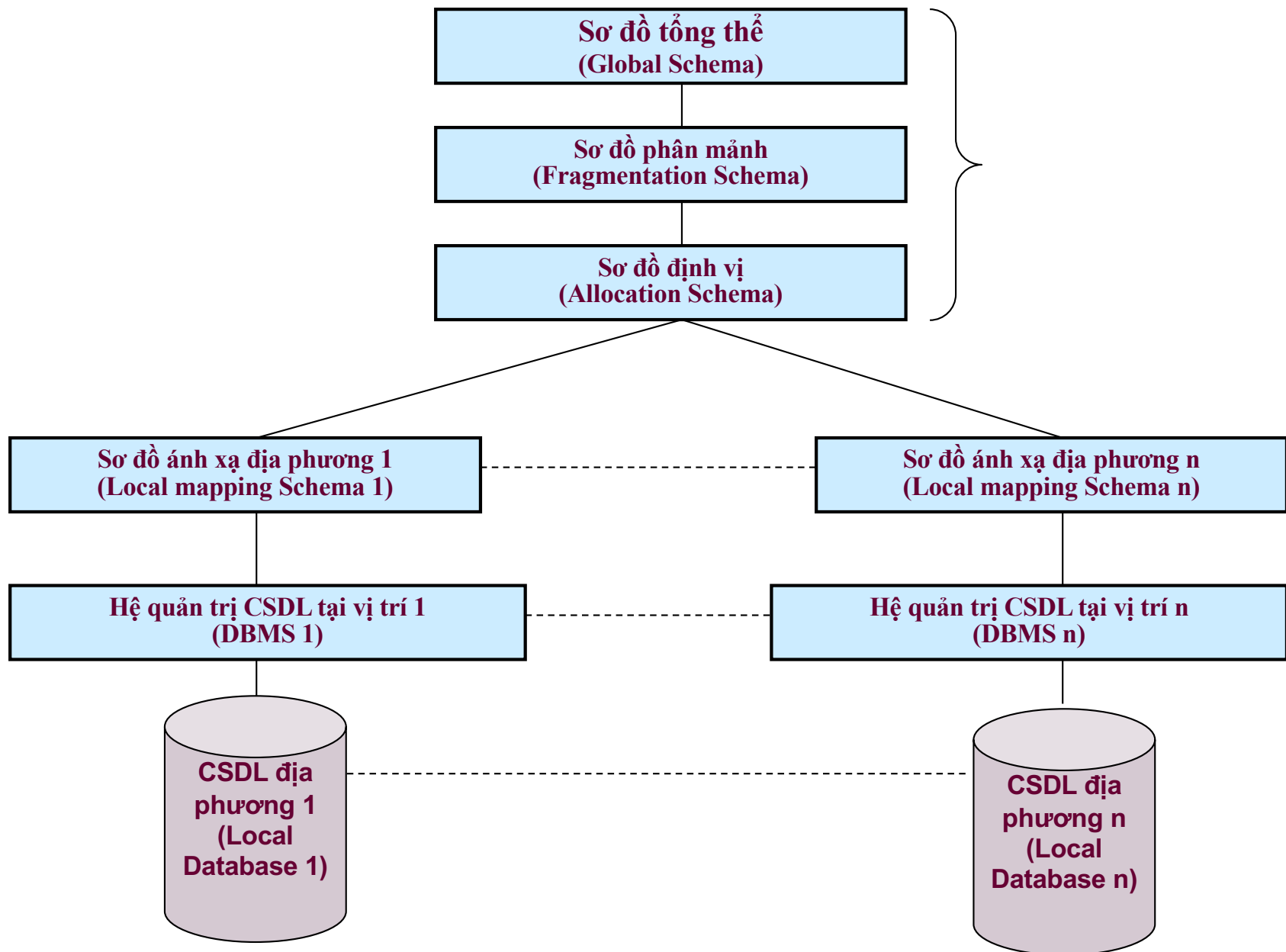
Ưu điểm của hệ phân tán

- ✧ Đáp ứng nhanh hầu hết các ứng dụng sử dụng dữ liệu tại các trạm
 - ✧ Tăng cường các đơn thể ứng dụng và CSDL mà không làm cản trở người sử dụng hiện tại.
 - ✧ Kiểm soát dữ liệu địa phương theo hướng hoàn thiện sự tích hợp và quản trị dữ liệu từ xa.
 - ✧ Tăng cường khả năng của hệ thống liên quan đến sự dư thừa dữ liệu.
-

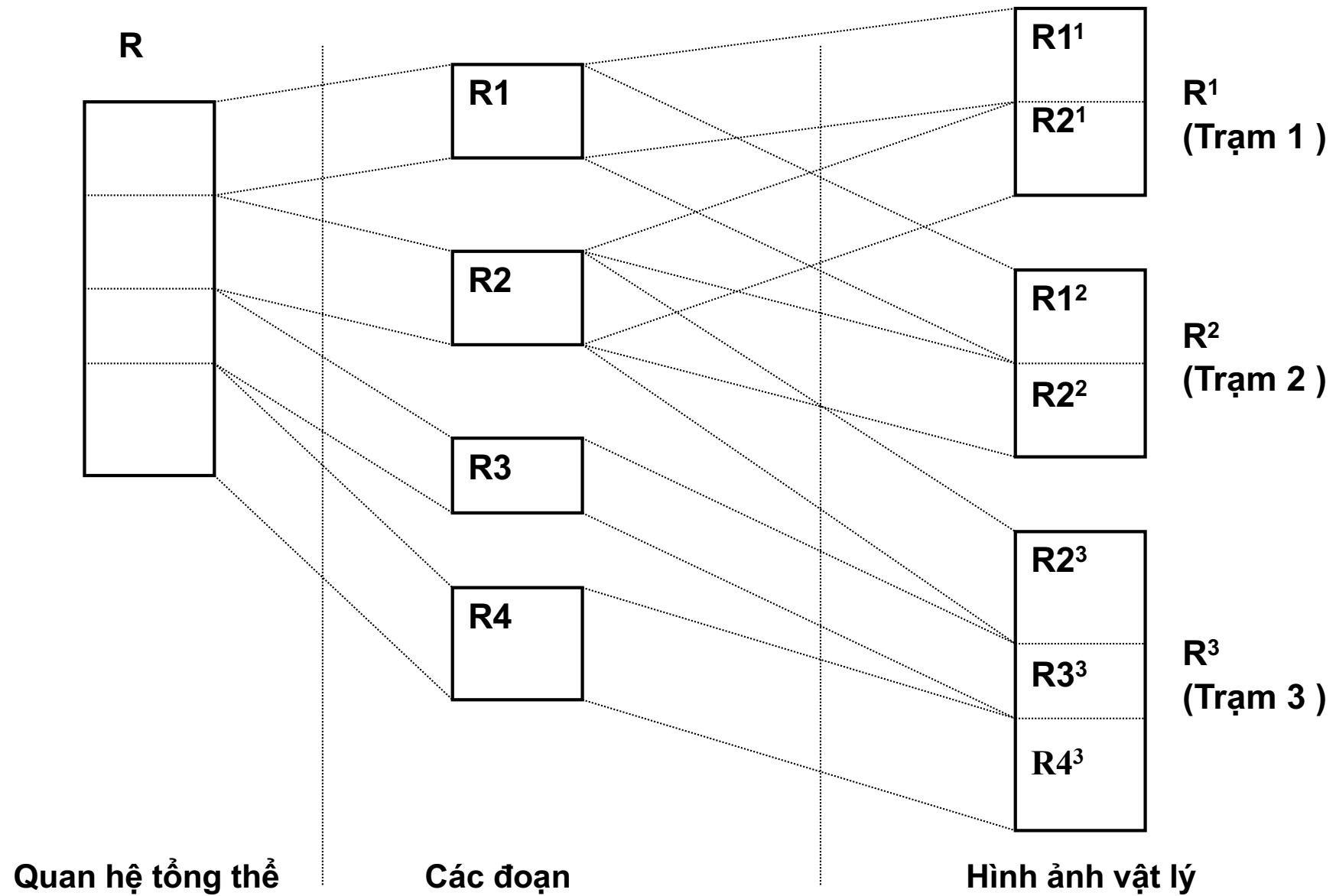
Nhược điểm của hệ phân tán

- ✧ Phần mềm đắt và phức tạp
 - ✧ Phải xử lý các thay đổi thông báo trong mọi địa điểm
 - ✧ Khó kiểm soát tính toàn vẹn dữ liệu với nhiều bản sao dữ liệu được phân bố khắp mọi nơi.
 - ✧ Đáp ứng chậm nhu cầu của các trạm trong trường hợp các phần mềm ứng dụng không được phân bố phù hợp với việc sử dụng chung.
-

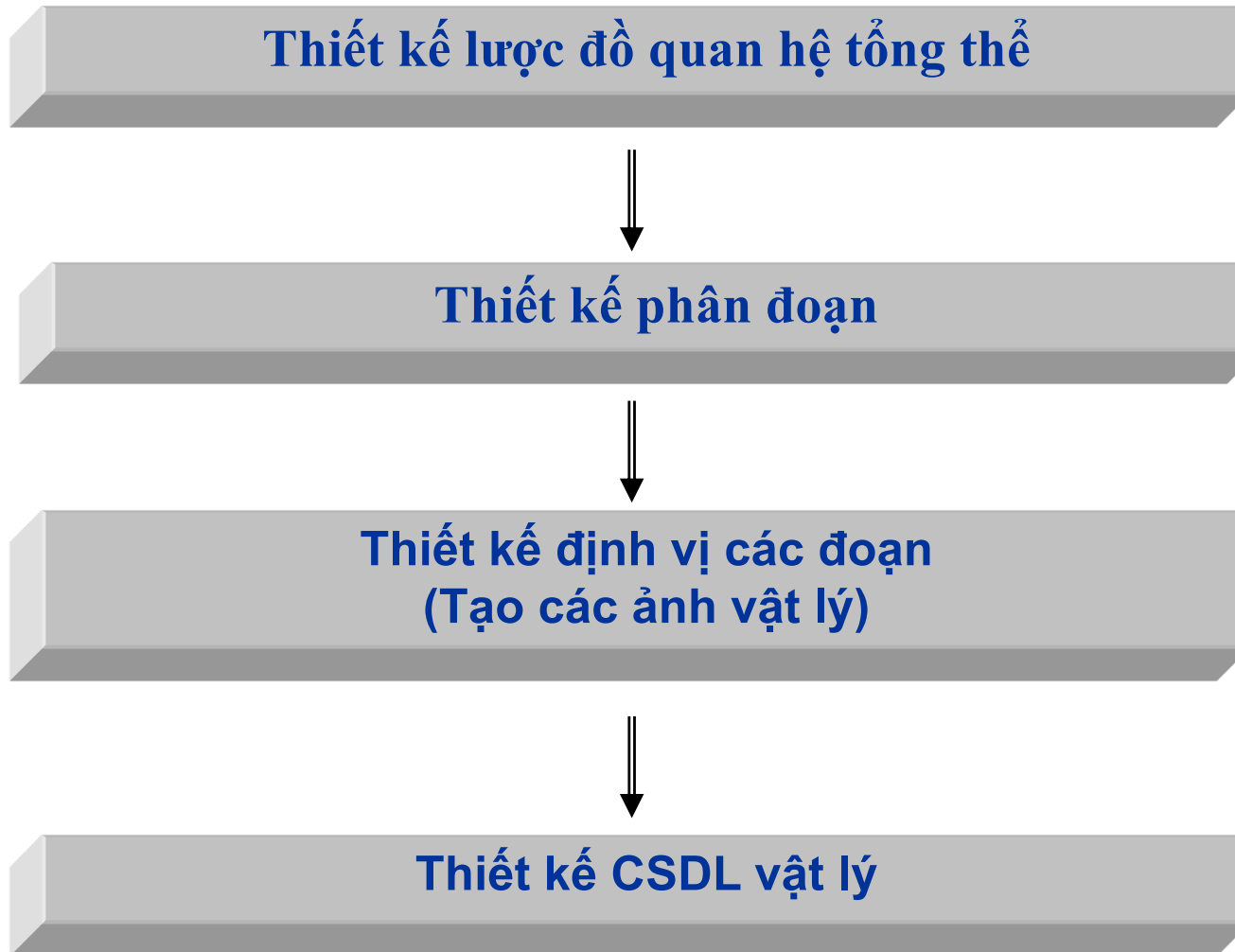
II. Kiến trúc của CSDL phân tán



Các đoạn và hình ảnh vật lý của một quan hệ tổng thể



III. PHƯƠNG PHÁP THIẾT KẾ CSDL PHÂN TÁN



Sơ đồ thiết kế tổng thể

■ **Thiết kế lược đồ quan hệ tổng thể**

- ❖ Thiết kế các quan hệ tổng thể

- ❖ Mô tả toàn bộ dữ liệu sẽ được dùng trong hệ thống

■ **Thiết kế phân đoạn:** thực hiện chia nhỏ dữ liệu thành các phần.

■ **Thiết kế định vị các đoạn**

- ❖ là quá trình thực hiện ánh xạ các đoạn vào các trạm khác nhau

- ❖ Tạo các ảnh vật lý tại các trạm.

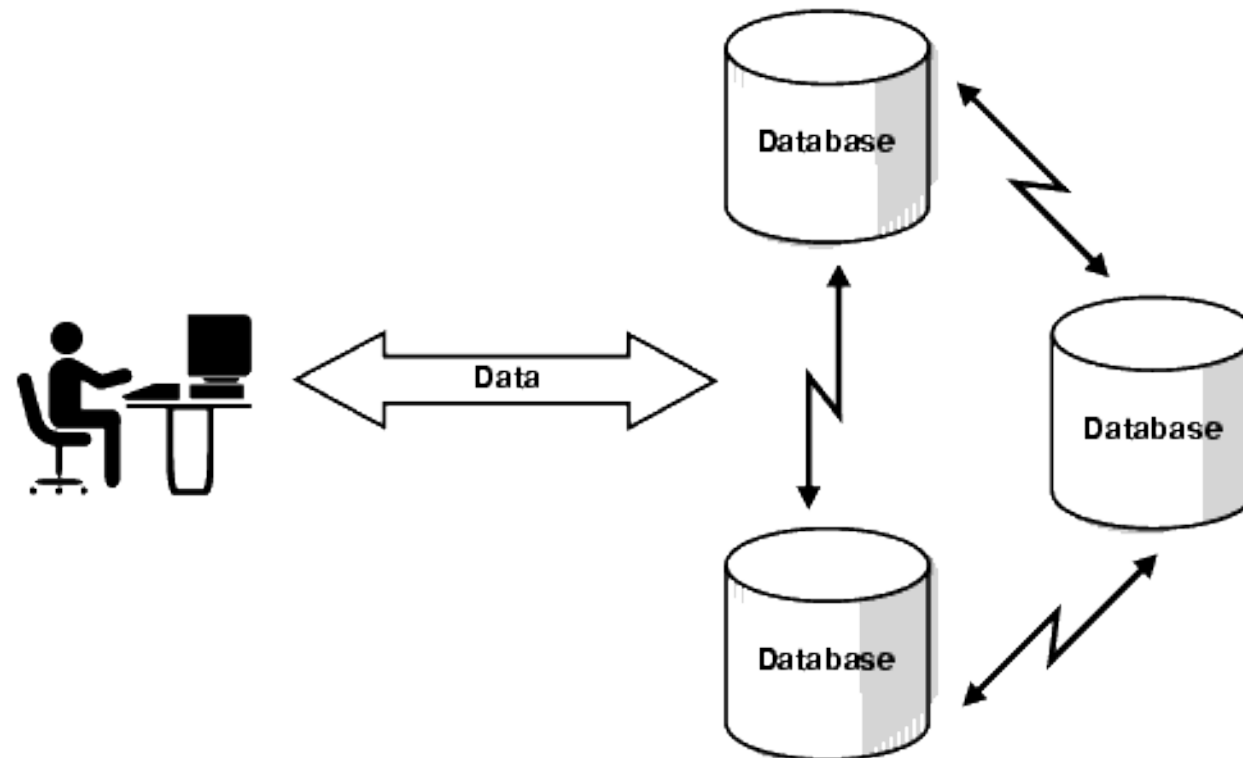
- ❖ Các đoạn dữ liệu được đưa vào các vị trí lưu trữ thích hợp với yêu cầu hoạt động thực tế của hệ thống.

■ **Thiết kế cơ sở dữ liệu vật lý:** thiết kế dữ liệu vật lý cho các quan hệ tại các trạm



IV. PHÂN MẢNH DỮ LIỆU

● **Phân mảnh dữ liệu:** Việc chia một quan hệ thành nhiều quan hệ nhỏ hơn.



**Các loại phân mảnh*

- ❑ **Phân mảnh ngang** (horizontal fragmentation)
- ❑ **Phân mảnh dọc** (vertical fragmentation)
- ❑ **Phân mảnh hỗn hợp** (hibrid fragmentation)

** Tùy vào từng yêu cầu của bài toán để xác định kiểu phân mảnh cho CSDL.*



Phân mảnh ngang

- Chia quan hệ ban đầu R thành các mảnh (đoạn), mỗi mảnh chứa một số bộ giá trị của R thỏa mãn một tính chất nào đấy (**điều kiện phân mảnh**).

Phân đoạn ngang thỏa mãn 3 điều kiện phân mảnh:

- Đầy đủ:** Mỗi bộ giá trị của R đều được đặt vào 1 R_i nào đó
- Xây dựng lại:** $R = R_1 \cup R_2 \dots \cup R_n$
- Tách rời:** $R_i \cap R_j = \phi$

*(Thực chất là **phép chọn** quan hệ trên một điều kiện xác định)*

Phân mảnh ngang

PROJ

| <i>PNo</i> | <i>PName</i> | <i>Budget</i> | <i>Loc</i> |
|------------|--------------|---------------|------------|
| P1 | Thiết bị | 150000 | Toronto |
| P2 | CSDL | 125000 | NewYork |
| P3 | Games | 75000 | NewYork |
| P4 | CAD | 100000 | Paris |

PROJ1

| <i>PNo</i> | <i>PName</i> | <i>Budget</i> | <i>Loc</i> |
|------------|--------------|---------------|------------|
| P1 | Thiết bị | 150000 | Toronto |
| P2 | CSDL | 125000 | NewYork |

PROJ2

| <i>PNo</i> | <i>PName</i> | <i>Budget</i> | <i>Loc</i> |
|------------|--------------|---------------|------------|
| P3 | Games | 75000 | NewYork |
| P4 | CAD | 100000 | Paris |

$\text{PROJ1} = \sigma_{\text{Budget} > 100000} \text{PROJ}$

$\text{PROJ2} = \sigma_{\text{Budget} \leq 100000} \text{PROJ}$



Phân mảnh dọc

● Là tách quan hệ ban đầu R thành các quan hệ R_i **có số cột nhỏ hơn**, (tập con của R) nhưng mỗi R_i đều phải chứa khóa chính của R .

- Phân mảnh dọc không thỏa mãn điều kiện tách rời.
- Xây dựng lại: $R = R_1 * R_2 * \dots * R_n$

(Thực chất là phép chiếu quan hệ R lên tập các thuộc tính)

Phân mảnh dọc

PROJ

| <i>PNo</i> | <i>PName</i> | <i>Budget</i> | <i>Loc</i> |
|------------|--------------|---------------|------------|
| P1 | Thiết bị | 150000 | Toronto |
| P2 | CSDL | 125000 | NewYork |
| P3 | Games | 75000 | NewYork |
| P4 | CAD | 100000 | Paris |

PROJ3

| <i>PNo</i> | <i>PName</i> | <i>Loc</i> |
|------------|--------------|------------|
| P1 | Thiết bị | Toronto |
| P2 | CSDL | NewYork |
| P3 | Games | NewYork |
| P4 | CAD | Paris |

PROJ4

| <i>PNo</i> | <i>Budget</i> |
|------------|---------------|
| P1 | 150000 |
| P2 | 125000 |
| P3 | 75000 |
| P4 | 100000 |

$\text{PROJ3} = \pi_{PNo, Pname, Loc} \text{PROJ}$

$\text{PROJ4} = \pi_{PNo, Budget} \text{PROJ}$

IV. Làm việc với CSDL phân tán trong SQL Server

1. Quy tắc khai thác dữ liệu

- Trong cùng một Server có thể truy xuất đến một đối tượng (bang, view,...) của CSDL khác
- Các thông tin phải xác định:
 - ❖ Tên CSDL - Tên người dùng - Tên đối tượng
- Cú pháp:
[database][.owner][.object Name]
- Ví dụ:
Select *
From QuanLyNhanVien.dbo.NhanVien

-
- **Với Server khác** (*Linked Server*) để truy xuất đến CSDL sử dụng cú pháp

[ServerName].[DatabaseName].[owner].[objectName]

- Ví dụ:

Select *

From **Server3.QuanLyNhanVien.dbo.tblNhanVien**

Select *

From **Server3.QuanLyNhanVien.user1.tblNhanVien**

2. Liên kết đến Server khác (Linked Server)

- Các máy chủ có thể được tham chiếu trong các truy vấn được gọi là **linked server**.
- Để liên kết đến một Server khác có thể sử dụng 2 cách
 - ❖ Sử dụng câu lệnh truy vấn
 - ❖ Thông qua cửa sổ **Management Studio**
- Một máy chủ định nghĩa được liên kết xác định các đối tượng sau đây:
 - ❖ *Một OLE DB provider*
 - ❖ *Một OLE DB data source*

Cách thức liên kết đến Server khác

- Thông qua câu lệnh
- Sử dụng công cụ

****Sử dụng truy vấn***





■ *Liên kết đến Server - Thêm một Linked Server*

EXEC **sp_addlinkedserver**

@server = <server_name> ,

@srvproduct = <product_name> ,

@provider = <provider_name> ,

@datasrc = <data_source path> ,

@catalog = <Database>

■ Trong đó:

- ❖ **@server = 'server name'** : Tên server dùng để tham chiếu khi bạn khai báo truy cập dữ liệu
- ❖ **@srvproduct = 'product_name'** : Tên sản phẩm ứng với cơ sở dữ liệu mà liên kết đến
 - SQL Server ⇔ không cần
 - Access ⇔ 'OLE DB Provider for Jet'
- ❖ **@provider = 'provider_string'** : Chuỗi ứng với trình điều khiển cơ sở dữ liệu
 - SQL Server ⇔ N' SQLOLEDB'
 - Access ⇔ 'Microsoft.Jet.OLEDB.4.0',
- ❖ **@datasrc = 'data_source'** : Tên CSDL (đường dẫn đầy đủ) được kết nối đến
- ❖ **@catalog = <Database>**: tên CSDL (SQL Server)

◆ Ví dụ Liên kết đến Server có CSDL SQL Server

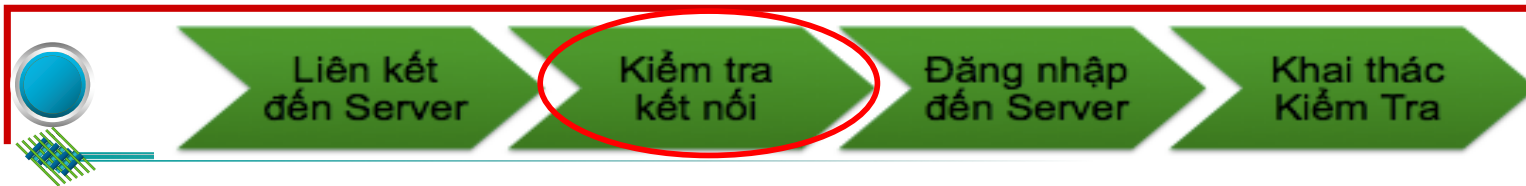
EXEC sp_addlinkedserver

```
@server = 'TrinhXuan_FIT',  
@srvproduct = 'SQLServer',  
@provider = 'SQLOLEDB',  
@datasrc = ' TrinhXuan_FIT\QuanLyDuAn';
```

◆ Ví dụ Liên kết đến Server có CSDL Access

EXEC sp_addlinkedserver

```
@server = 'TrinhXuan_FIT',  
@srvproduct = 'OLE DB Provider for Jet',  
@provider = 'OLE DB Provider for ACE',  
@datasrc = 'D:\DB_Access\QLyDuAn.accdb';
```



- **Kiểm tra kết nối với Server**

EXEC sp_linkedservers

Ví dụ: EXEC sp_linkedservers

| SRV_NAME | SRV_PROVIDERNAME | SRV_PRODUCT | SRV_DATASOURCE | SRV_PROVIDERSTRING | SRV_LOCATION | SRV_CAT |
|---------------|------------------|-------------|------------------------------|--------------------|--------------|---------|
| XUANTT\SQL2K8 | SQLLEDB | SQL SERVER | XUANTT\SQL2K8 | NULL | NULL | NULL |
| TRINHXUAN_FIT | SQLOLEDB | SQLServer | TrinhXuan_FIT\Quan LyDuAn | NULL | NULL | NULL |

-
- **Xóa kết nối với Server**

EXEC sp_dropserver <name-server>

- **Ví dụ**

EXEC sp_dropserver TrinhXuan_FIT

Liên kết
đến Server

Kiểm tra
kết nối

Đăng nhập
đến Server

Khai thác
Kiểm Tra

Đăng nhập đến Server được kết nối

EXEC sp_addlinkedserver

@rmtsrvname = <name-server> ,

@rmtuser = <user> ,

@rmtpassword = <password>

Trong đó:

- @rmtsrvname: tên server cần truy vấn giống như tên trong phát biểu **sp_addlinkedserver**
- @rmtuser: tên user để login vào server
- @rmtpassword: mật khẩu đăng nhập

EXEC dbo.sp_addlinkedserver

@rmtsrvname = 'TrinhXuan_FIT' ,

@rmtuser = 'sa' ,

@rmtpassword = '123456'



***Kiểm tra kết quả (sử dụng câu lệnh truy vấn)**

```
SELECT *  
FROM TrinhXuan_FIT. QuanLyDuAn.dbo.tblDUAN
```

3. Tạo Nhãn dữ liệu (trong suốt)

- Với các dữ liệu hỗn tạp ở nhiều nơi, để tập trung các dữ liệu hỗn tạp đó lại và thao tác như làm việc trên một máy cục bộ => gán nhãn tên cho dữ liệu (**trong suốt dữ liệu**)

- Lệnh tạo sự trong suốt dữ liệu:

Create synonym <synonym_name>

for <Object>

- Trong đó:

❖ **synonym_name**: là tên của trong suốt mới được đặt

❖ **Object**: là xác định đối tượng cho phép trong suốt,

<object> :: = {

[server_name.[database_name].[schema_name_2]. |

database_name . [schema_name_2]. |

schema_name_2.] object_name }

Ví dụ

- Thay vì với câu lệnh truy vấn

```
Select * from TrinhXuan_FIT.QuanLyDA.dbo.tblDuAn
```

- Thay bằng

```
Create synonym HSDuAn  
for TrinhXuan_FIT.QuanLyDA.dbo.tblDuAn
```

```
Select * from HSDuAn
```



V. BÀI TẬP ÁP DỤNG

● Cho CSDL **QLNhanvien**:

tblPhongBan(MaPB, TenPB, DiaChi, DienThoai)

tblNhanVien(MaNV, TenNV, Luong, MaPB)

- Phân mảnh dọc bảng **tblPhongban**
- Phân mảnh ngang bảng **tblNhanvien**

● Xây dựng các đối tượng (truy vấn, khung nhìn, thủ tục lưu, trigger,...) thao tác trên các bảng.

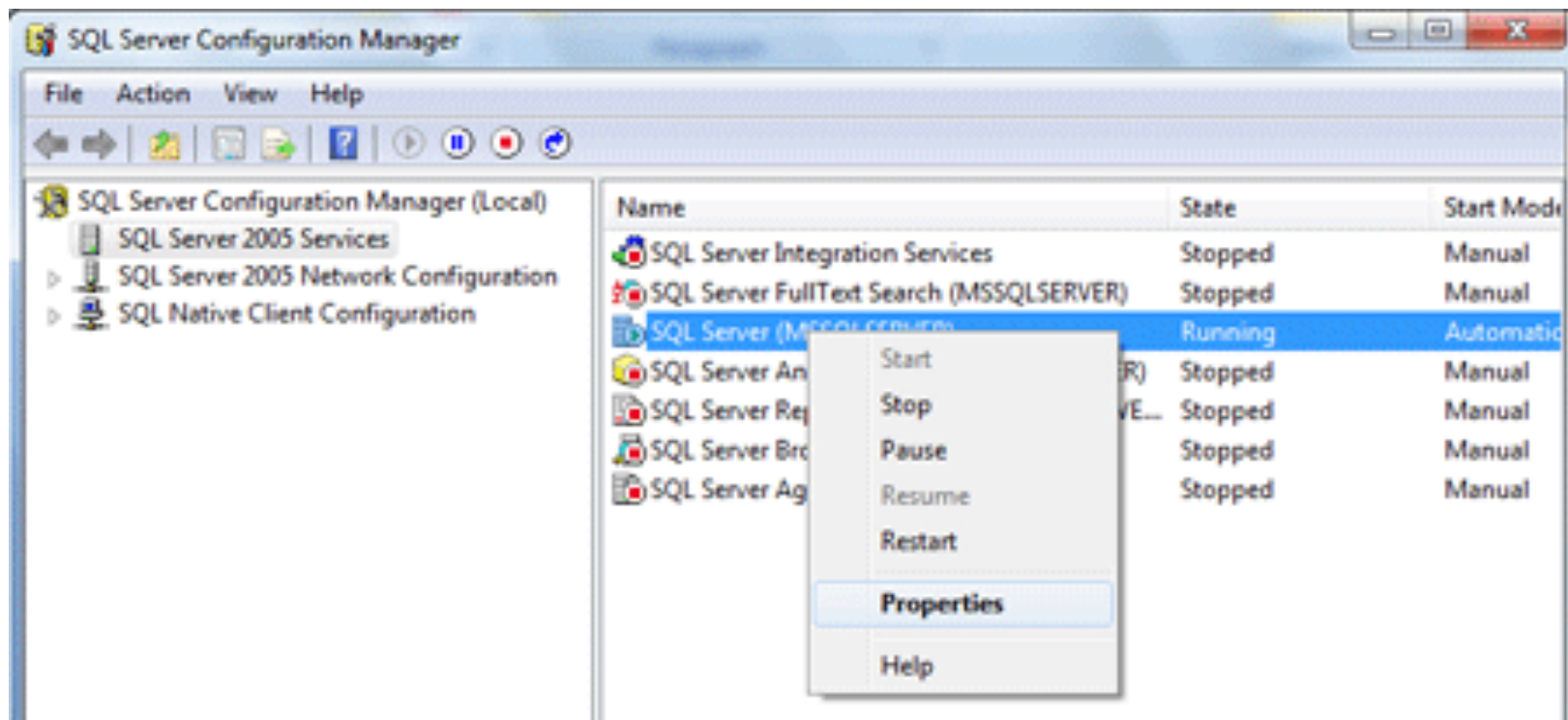


Chuẩn bị

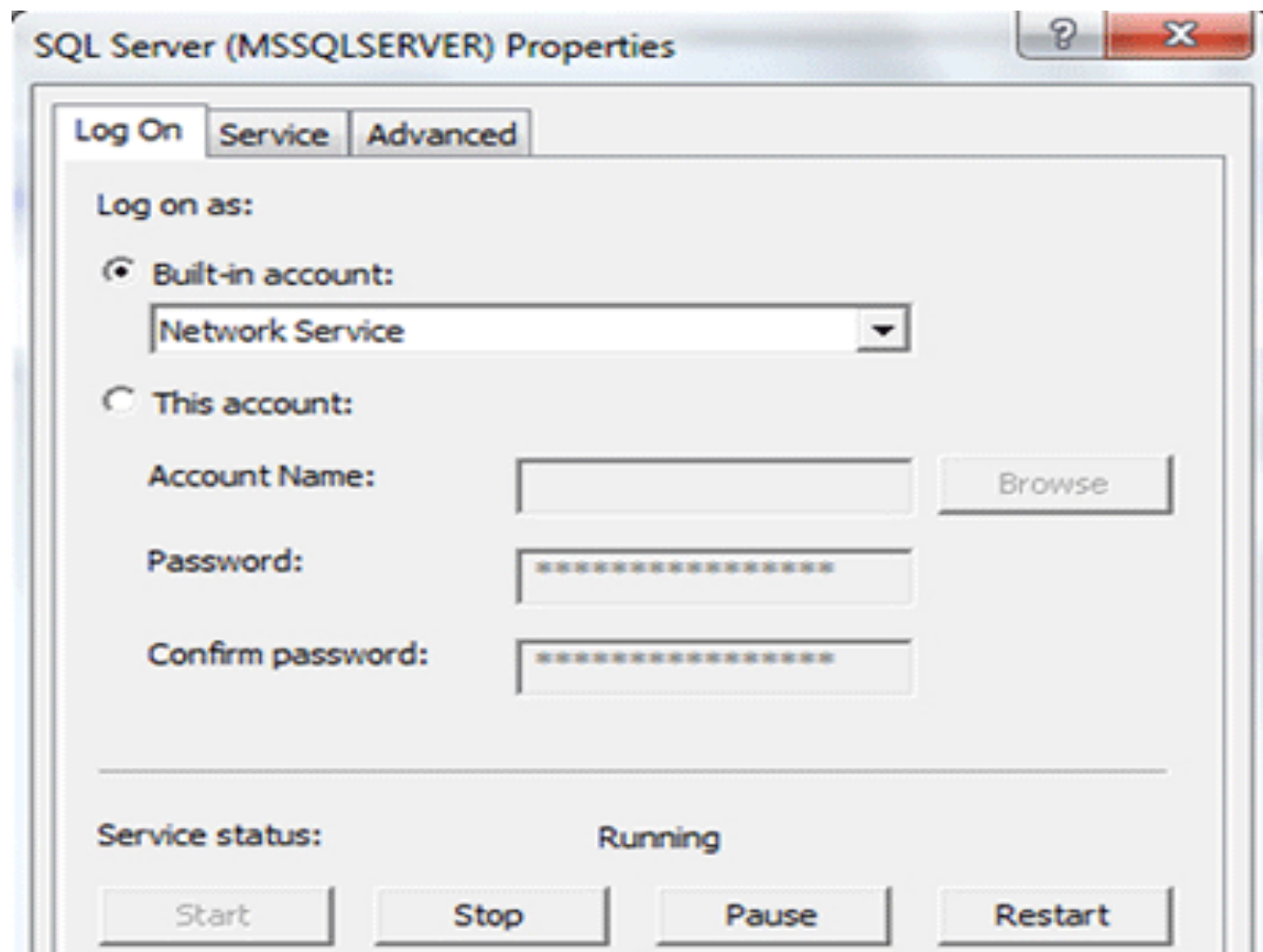
- Cài đặt thêm SQL Server trên máy ảo.
- Cấu hình SQL server ở các trạm sao cho các trạm có thể “làm việc – liên kết” được với nhau (sử dụng linked server):
 - **Config SQL Server cho phép kết nối từ xa*
 - **Tạo user để thực hiện kết nối từ xa*
 - **Cấu hình Firewall*
 - **Cấu hình Router cho phép kết nối qua cổng 1433*
 - **Thử kết nối*

*Config SQL Server cho phép kết nối từ xa

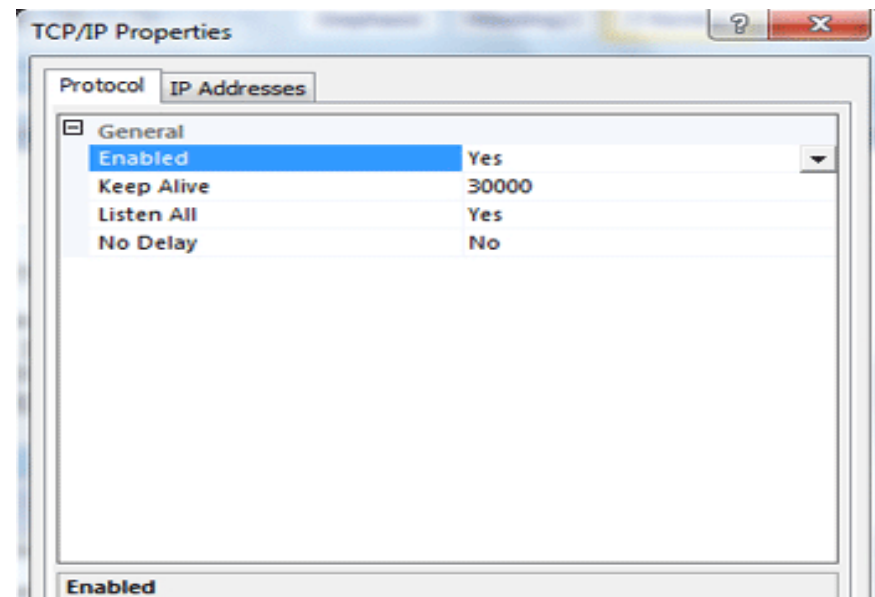
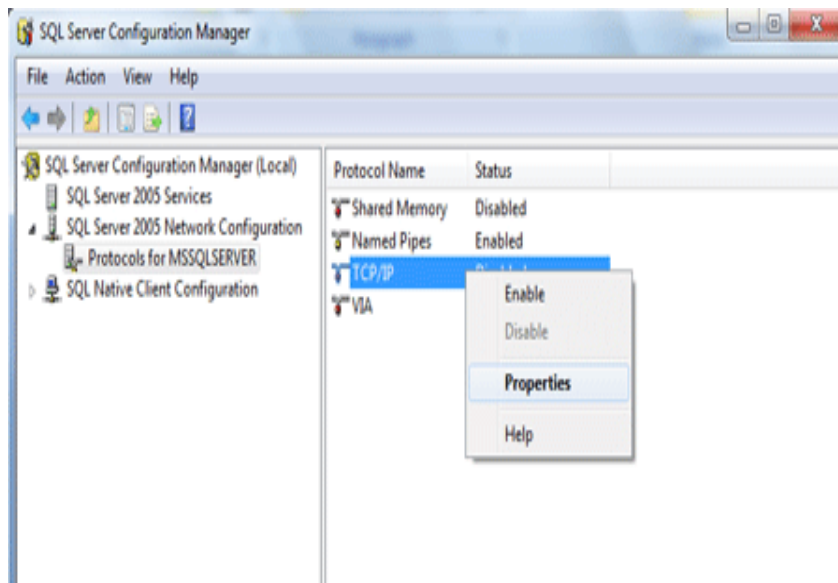
- Mở **SQL Server Configuration Manager**
- Chọn mục **SQL Server ... Services** -> tìm mục **SQL Server** (VD: **MSSQLSERVER** – *tên của instance*)
- Nháy phải chuột chọn **Properties**



- Trong thẻ **Log On**, click chọn vào **Built-In Account**, chọn vào **Network Service**, chọn **OK**



- Quay lại SQL Server Configuration Manager -> chọn **SQL Server ... Network Configuration** -> mở ra menu **Protocols for MSSQLSERVER** (tên trên máy bạn có thể khác), chọn vào mục này -> tìm mục **TCP/IP**, click phải và chọn **Properties**
- ở Tab **Protocols**, mục **Enabled** chọn vào **Yes**



■ Chọn **IP Address** -> Danh sách các IP (IP1, IP2,...)

- ❖ đây là danh sách các IP hình thành khi máy có kết nối vào các mạng LAN khác nhau

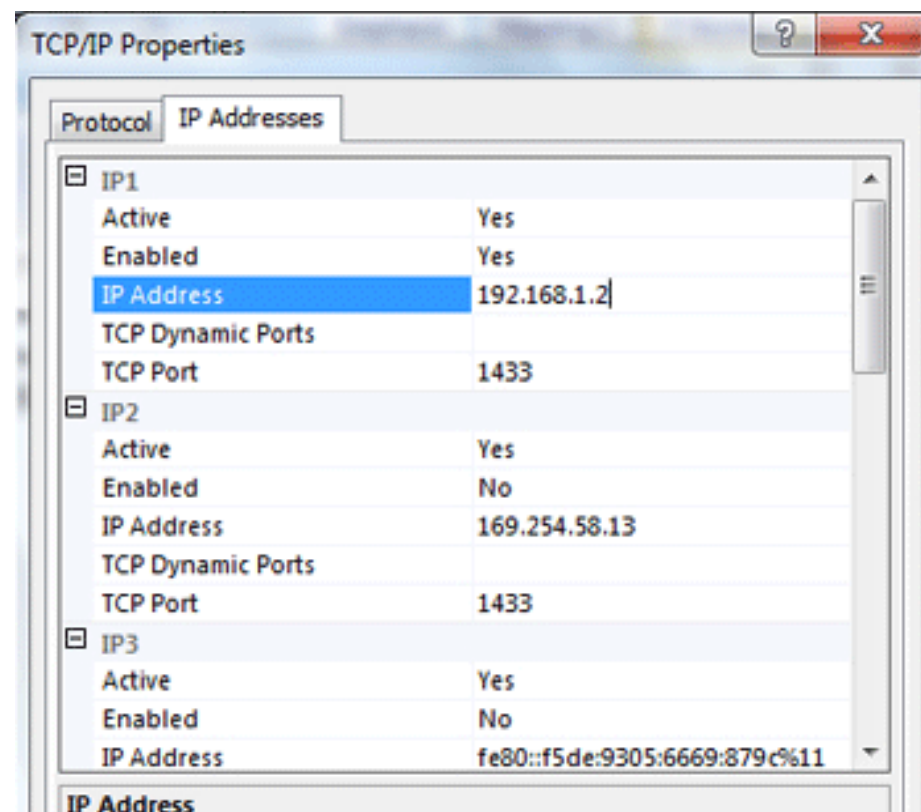
Hãy chọn một IP nào đó bất kỳ:

- ❖ Ở mục **IP Address** xóa đi và gõ lại địa chỉ IP của máy đang dùng trong mạng LAN

- ❖ Ở mục **Active** - chọn Yes, mục **Enabled** - chọn Yes

- ❖ Ở mục **TCP Port**, khai báo cổng share mặc định của SQL, để mặc định là **1433**

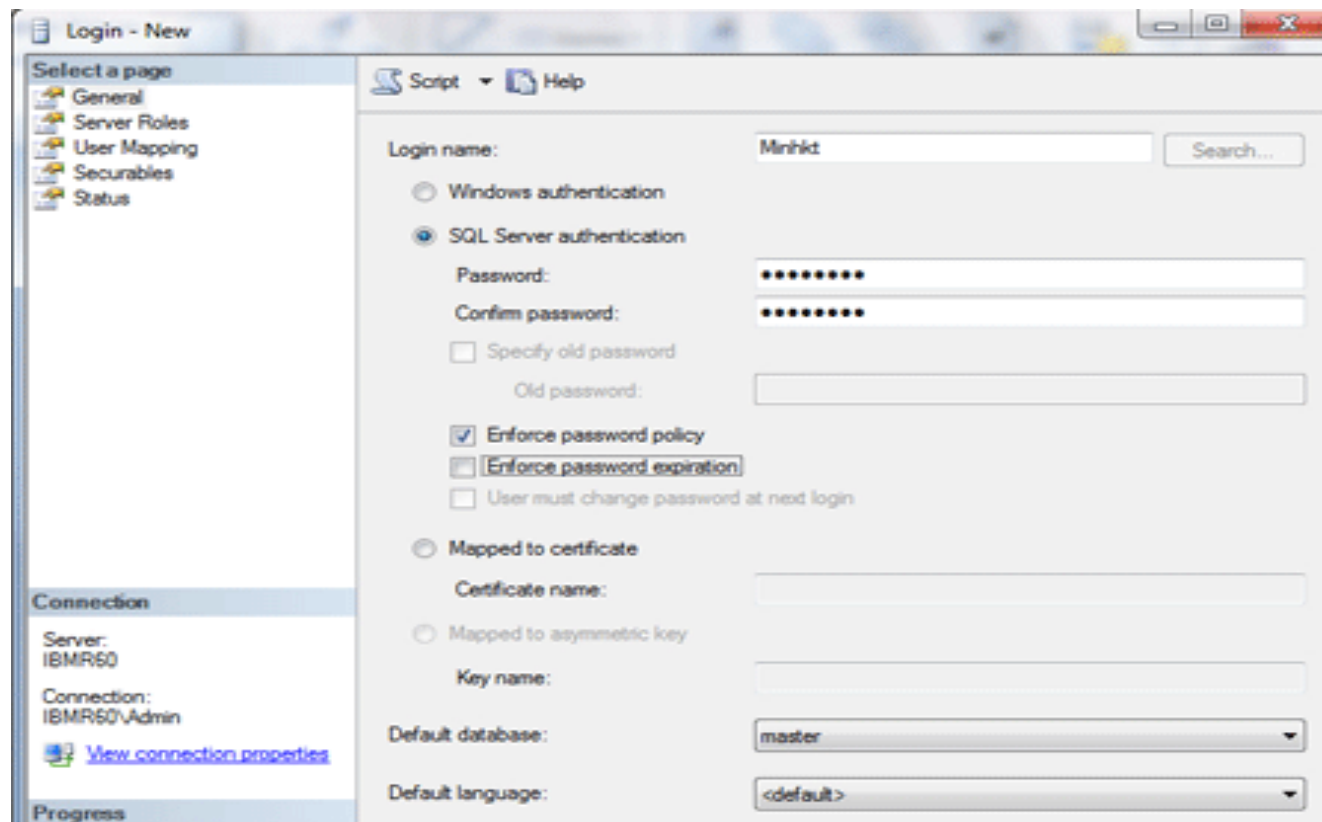
- ❖ Click OK



Restart lại SQL Server.

*Tạo user để thực hiện kết nối từ xa

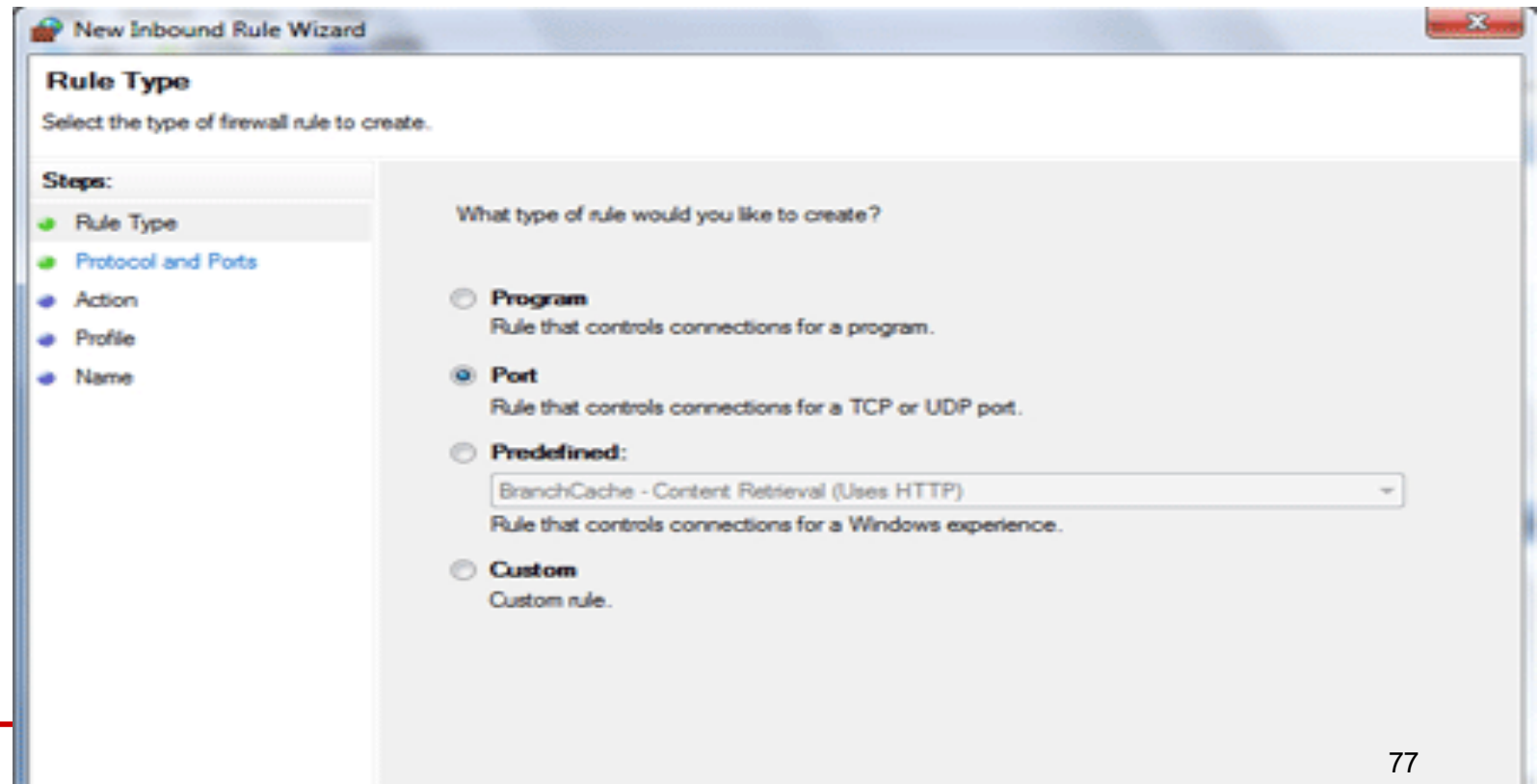
- Mở SQL Server Management Studio
- mở mục **Security** -> **Login** -> chuột phải và chọn **New Login**
- Gõ vào **User**, **Password** và bỏ dấu ở mục **Enforce password Expiration** -> Nhấn OK



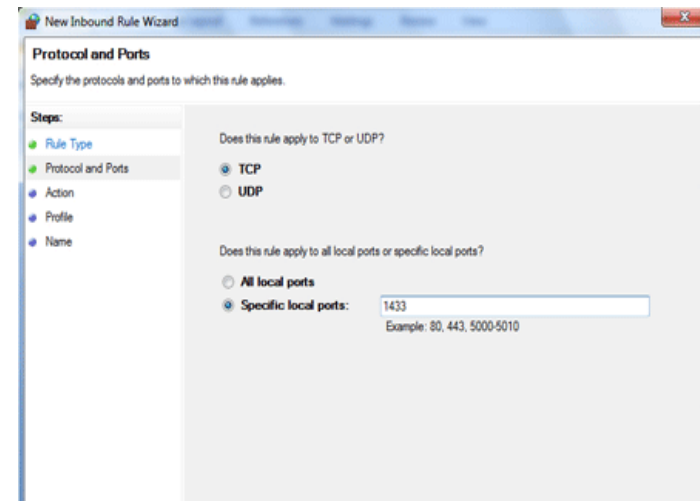
*Cấu hình Firewall

■ Win 7

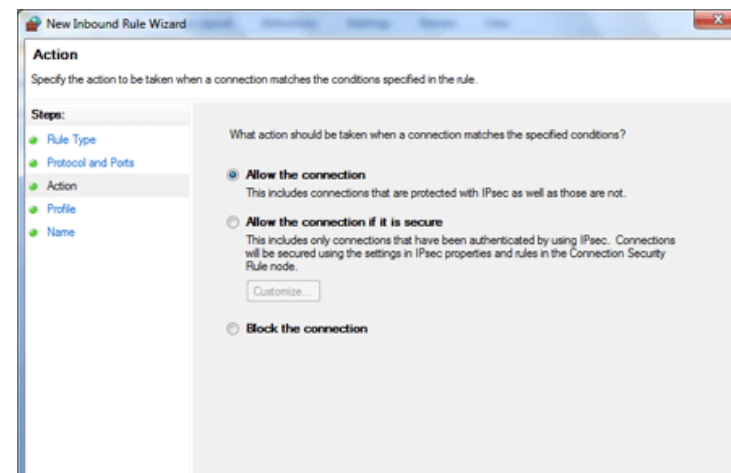
- ❖ Start -> Control Panel -> Windows Firewall -> chọn **Advanced settings** -> chọn **Inbound Rules** -> chọn **New Rule**
- ❖ Đánh dấu vào mục **Port** -> Nhấn **Next**



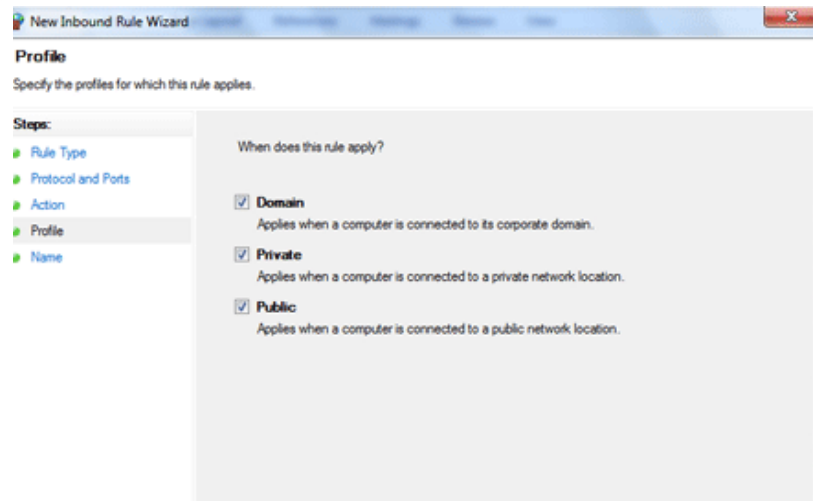
- ❖ Chọn vào mục **TCP** và **Specific local Ports**, gõ vào 1433 -> Nhấn Next



- ❖ đánh dấu chọn vào mục **Allow the connection** -> Next



- ❖ chọn kiểu mạng sẽ áp dụng mở cổng này, đánh dấu vào cả 3 mục **Domain, Private, Public** -> Nhấn Next



- ❖ gõ tên kết nối **Name** và **Description** tùy ý -> Nhấn Finish



Log Off hoặc Restart lại máy

*Cấu hình Router cho phép kết nối qua cổng 1433

- Tùy loại Router đang sử dụng việc cấu hình có thể khác nhau
- Các mục cơ bản:
 - ❖ **Custom Server:** Tên của thiết lập, do bạn tự đặt tùy ý
 - ❖ **Server IP Address:** Đây là địa chỉ IP của máy mà bạn cài làm máy chủ. Ví dụ của tôi là 192.168.1.2
 - ❖ **Port Start, Port End** là số hiệu của cổng nhận yêu cầu, đều khai báo là 1433
 - ❖ **Mục Protocol:** Chọn phương thức kết nối là TCP.

*Thử kết nối

- Mở *SQL Server Management Studio*
- Gõ các thông tin kết nối -> Connect



connect được thì đã cài đặt và thiết lập thành công.



* THỰC HÀNH PHÂN TÁN

tblPhongBan (MaPB, TenPB, DiaChi, DienThoai)

tblNhanVien (MaNV, TenNV, GT, Luong, MaPB)

- Phân mảnh dọc bảng **tblPhongban**
 - ✓ Server 1: MaPB, TenPB
 - ✓ Server 2(Linked Server): MaPB, DiaChi, DienThoai
- Phân mảnh ngang bảng **tblNhanvien**
 - ✓ Server 1: Các nhân viên “**Nữ**”
 - ✓ Server 2(Linked Server): Các nhân viên “**Nam**”

Server 2: **TrinhXuan_FIT**



Demo ví dụ

- Tạo CSDL QLNhanvien (chú ý thiết lập COLLATE cùng giá trị cho cả 2 server):

Create Database QLNhanvien COLLATE Vietnamese_CI_AS

Go

Use QLNhanvien

Go

Tại Server 1

-- *Tại Server 1 tạo bảng PhongBan: MaPB, TenPB*

Create Table tblPhongban

```
(      MaPB varchar(10) Primary key not null,  
      TenPB nvarchar(20) Unique not null  
)
```

-- *Tại Server 1 tạo bảng NhanVien*

Create Table tblNhanvien

```
(      MaNV varchar(10) Primary key not null,  
      TenNV nvarchar(30) Unique not null,  
      GT nvarchar(3) not null,  
      Luong float not null,  
      MaPB varchar(10) Foreign key(MaPB)  
          references tblPhongban(MaPB)  
          ON update Cascade  
          ON delete Cascade  
)
```

Tại Server 2 – Linked Server-TrinhXuan_FIT

-- Tại Server 2 tạo bảng PhongBan: MaPB, DiaChi, DienThoai

Create Table tblPhongban

```
(      MaPB varchar(10) Primary key not null,  
      DiaChi nvarchar(30) not null,  
      DienThoai nvarchar(13) not null  
)
```

-- Tại Server 2 tạo bảng NhanVien:

Create Table tblNhanvien

```
(      MaNV varchar(10) Primary key not null,  
      TenNV nvarchar(30) Unique not null,  
      GT nvarchar(3) not null,  
      Luong float not null,  
      MaPB varchar(10) Foreign key(MaPB)  
          references tblPhongban(MaPB)  
          ON update Cascade  
          ON delete Cascade  
)
```

Kết nối từ Server 1 đến Server 2

EXEC sp_addlinkedserver

```
@server = 'TrinhXuan_FIT',  
@srvproduct = 'SQLServer',  
@provider = 'SQLOLEDB',  
@datasrc = 'TrinhXuan_FIT\QLNhanvien';
```

EXEC dbo.sp_addlinkedsrvlogin

```
@rmtsrvname = 'TrinhXuan_FIT',  
@rmtuser = 'sa',  
@rmtpassword = '123456'
```




Tạo thủ tục thêm một bản ghi mới vào bảng tblPhongban:

Create Proc spInsertPhongban (@ma varchar(10), @ten nVarchar(20),
@dc nVarchar(30), @dt varchar(13))

AS

Begin

-- Kiểm tra tính hợp lệ của dữ liệu nhập

if Exists (Select * from tblPhongban where MaPB = @ma)

Begin

print N'Mã phòng ban đã có'

return

End

Insert Into tblPhongban **Values**(@ma, @ten)

Insert Into TrinhXuan_FIT. **QLNhanvien**.dbo.tblPhongban **Values**(@ma, @dc, @dt)

print N'Thêm thành công'

End

Exec spInsertPhongban 'PGD', N'Phòng giám đốc', N'Hà Nội', '043515265'

Exec spInsertPhongban 'PTH', N'Phòng tổng hợp', N'Nam Định', '043515678'



Thủ tục thêm dữ liệu bảng nhân viên

```
Create Proc spInsertNhanvien (@ma varchar(10), @ten nVarchar(20),  
                               @gt nvarchar(3), @luong float, @mapb varchar(10) )
```

AS

Begin

```
If (@gt = N'Nữ')
```

```
    Insert Into tblNhanvien
```

```
    Values(@ma, @ten, @gt, @luong, @mapb)
```

```
Else
```

```
    Insert Into TrinhXuan_FIT. QLNhanvien.dbo.tblNhanVien
```

```
    Values(@ma, @ten, @gt, @luong, @mapb)
```

```
    print N'Thêm thành công'
```

End

```
Exec spInsertNhanvien 'NV001', N'Nguyễn Thế Hùng', 'Nam', 1200, 'PGD'
```

```
Exec spInsertNhanvien 'NV002', N'Nguyễn Thu Hà', N'Nữ', 800, 'PGD'
```

```
Exec spInsertNhanvien 'NV003', N'Lê Văn Tuấn', 'Nam', 850, 'PTH'
```



Tạo View lấy thông tin tổng hợp đầy đủ cho phòng ban

**Create synonym HSPB
for TrinhXuan_FIT.QuanLyDuAn.dbo.tblPhongban**

Create view DSPhongBan

As

Begin

Select *

from tblPhongban, HSPB

where tblPhongBan.MaPB = HSPB.MaPB

end



Tạo View lấy thông tin tổng hợp đầy đủ cho tất cả nhân viên

**Create synonym HSNV
for TrinhXuan_FIT.QuanLyDuAn.dbo.tblNhanVien**

Create view DSNhanVien

As

Begin

Select *

from tblNhanVien

Union

Select *

from HSNV

end