

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
KHOA CÔNG NGHỆ THÔNG TIN I



BÁO CÁO BÀI TẬP LỚN
KHO DỮ LIỆU

Nhóm học phần : Nhóm 02
Nhóm bài tập lớn : Nhóm 03
Thành viên nhóm :
Dương Xuân Hùng - B21DCCN409
Vũ Minh Kiên – B21DCCN469
Trương Quang Lập – B21DCCN073
Trần Quốc Khánh – B21DCCN457
Trần Sỹ Tiến – B21DCCN709
Giảng viên hướng dẫn : TS. Nguyễn Quỳnh Chi

Hà Nội – 2025

MỤC LỤC

Chương I: Giới thiệu	5
1. Mục tiêu	5
2. Phạm vi	5
Chương II: Yêu cầu nghiệp vụ	7
Chương III: Đặc tả yêu cầu chức năng	8
1. Đặc tả đầu vào	8
Chương IV: Thiết kế kho dữ liệu	10
1. Chuyển đổi lược đồ quan hệ sang mô hình thực thể liên kết mở rộng:	10
2. Tích hợp 2 mô hình thực thể liên kết thành một mô hình thực thể liên kết tích hợp IER:	17
3. Nghiên cứu yêu cầu của kho dữ liệu:	20
3.1. Nghiên cứu yêu cầu của kho dữ liệu:	20
3.2. Mô hình thiết kế kho dữ liệu:	21
3.3. Phân cấp cho các chiều dữ liệu:	24
Chương V: Cài đặt các khối dữ liệu	26
1. Tạo 2 cơ sở dữ liệu hiện tại của doanh nghiệp:	26
1.1. Cơ sở dữ liệu Văn phòng đại diện:	26
1.2. Cơ sở dữ liệu Bán hàng:	28
1.3. Sinh dữ liệu cho 2 CSDL:	31
2. Tạo cơ sở dữ liệu tích hợp:	35
3. Tạo kho dữ liệu:	38
3.1. Tạo CSDL KDL_DB_1:	38
3.2. Ánh xạ từ CSDL tích hợp sang kho:	41
3.3. Đổ dữ liệu vào kho	45
3.4. Thiết lập tự động cập nhật dữ liệu vào kho:	47
4. Tạo các khối dữ liệu:	49
Chương VI: Báo cáo phân tích trực tuyến	54
1. Tạo kết nối từ Power BI tới Database:	54

2. Tạo báo cáo:	57
2.1. Cube DoanhThu:	57
2.2. Cube TonKho:.....	67
Chương VII: Kiểm tra tính đúng đắn của dữ liệu	75
1. Tìm tất cả các cửa hàng cùng với thành phố, bang, số điện thoại, mô tả, kích cỡ, trọng lượng và đơn giá của tất cả các mặt hàng được bán ở kho đó.	75
2. Tìm tất cả các đơn đặt hàng với tên khách hàng và ngày đặt hàng được thực hiện bởi khách hàng đó.	76
3. Tìm tất cả các cửa hàng cùng với tên thành phố và số điện thoại mà có bán các mặt hàng được đặt bởi một khách hàng nào đó.	77
4. Tìm địa chỉ văn phòng đại diện với tên thành phố, bang của tất cả các cửa hàng lưu kho một mặt hàng nào đó với số lượng trên mức cụ thể.	78
5. Với mỗi một đơn đặt hàng của khách, liệt kê các mặt hàng được đặt cùng với mô tả, mã cửa hàng, tên thành phố và các cửa hàng có bán mặt hàng đó.	79
6. Tìm thành phố và bang mà một khách hàng nào đó sinh sống	80
7. Tìm mức độ tồn kho của một mặt hàng cụ thể tại tất cả các cửa hàng ở một thành phố cụ thể nào đó	80
8. Tìm các khách hàng du lịch, khách hàng đặt theo đường bưu điện và khách hàng thuộc cả hai loại:	81
9. Tìm các khách hàng du lịch, khách hàng đặt theo đường bưu điện và khách hàng thuộc cả hai loại	82
10. Tính tổng doanh thu của 1 người:.....	83
Chương VIII: Kết luận.....	84

Nhiệm vụ	Thành viên thực hiện
Tích hợp đúng dữ liệu từ hai nguồn thành một nguồn (đổi sang mô hình thực thể liên kết rồi mới tích hợp mô hình ER)	Hùng, Khánh, Lập, Kiên, Tiến
Thiết kế mô hình dữ liệu đã tích hợp	Hùng, Khánh, Lập, Kiên, Tiến
Sinh dữ liệu cho các nguồn dữ liệu	Hùng
Thiết kế mô hình dữ liệu cho kho	Hùng, Khánh, Lập, Kiên, Tiến
Cách ánh xạ để đổ dữ liệu từ CSDL tích hợp vào kho	Hùng, Khánh, Lập, Kiên, Tiến
Đổ dữ liệu và sinh khối	Hùng, Khánh, Lập, Kiên, Tiến
Tạo phân cấp để tính khối cho OLAP	Hùng, Khánh, Lập, Kiên, Tiến
Thiết kế các khối dữ liệu để thực hiện OLAP	Hùng, Khánh, Lập, Kiên, Tiến
Tính sẵn các khối cho OLAP	Hùng, Khánh, Lập, Kiên, Tiến
Trực quan hóa bằng Power BI	Hùng
Viết tài liệu đầy đủ bao gồm cả mô tả yêu cầu và giới thiệu chung	Hùng

Chương I: Giới thiệu

Để có thể thiết kế một kho dữ liệu cần bắt đầu xác định mục yêu cầu nghiệp vụ của kho dữ liệu. Chính vì vậy việc xác định mục tiêu và phạm vi của bài toán đóng vai trò vô cùng quan trọng.

1. Mục tiêu

- Thực hiện các câu hỏi phỏng vấn chiến lược với doanh nghiệp sẽ giúp chúng ta xác định được mục tiêu của doanh nghiệp đó khi xây dựng kho dữ liệu.
 - + Lĩnh vực hoạt động của doanh nghiệp: Lĩnh vực kinh doanh, chuyên cung cấp các mặt hàng cho 2 đối tượng khách hàng chính. Thứ nhất là khách du lịch. Loại khách hàng này thường được dẫn đến bởi các hướng dẫn viên các đoàn du lịch. Thứ hai là những khách hàng đặt hàng qua đường bưu điện.
 - + Các ứng dụng nghiệp vụ đang được doanh nghiệp sử dụng: Doanh nghiệp sử dụng nhiều hệ thống ứng dụng nghiệp vụ khác nhau như quản lý số lượng và lượng tiêu thụ mặt hàng, quản lý thông tin khách hàng, quản lý cửa hàng,...
 - + Vấn đề đang gặp phải hiện nay của doanh nghiệp: Mỗi khách hàng chỉ sống ở một tỉnh, thành phố và doanh nghiệp cần đáp ứng các mặt hàng trong đơn hàng của khách bằng cách lấy từ kho ở tỉnh, thành phố mà khách hàng đó sinh sống. Nếu không thể đáp ứng những yêu cầu đó thì công ty sẽ chuyển yêu cầu tới các thành phố đến khi mặt hàng có đủ để cung cấp. Một số thông tin xử lý là rất quan trọng đối với doanh nghiệp này. Ví dụ, tổng số lượng mỗi mặt hàng được lưu trữ tại mỗi thành phố. Sau mỗi lần mặt hàng được phân phối (được bán) trong một tỉnh, thành phố, doanh nghiệp cần biết tổng số lượng mỗi mặt hàng còn lại tại tất cả các cửa hàng trong một thành phố đó.
- Vì vậy, ban lãnh đạo doanh nghiệp cần giải pháp cho những vấn đề đang gặp phải, và phương pháp được chọn để giải quyết các vấn đề ở đây là kho dữ liệu.

2. Phạm vi

- Một trong những nguyên nhân dẫn đến thất bại trong việc xây dựng một kho dữ liệu là không xác định đúng phạm vi xây dựng của kho. Việc xây dựng một kho dữ liệu để đáp ứng nhiều nghiệp vụ khác nhau là rất khó khăn và tiềm ẩn nhiều rủi ro về mặt tài chính.
- Trong thực tế, người ta thường bắt đầu xây dựng dữ liệu cho một vài nghiệp vụ cụ thể trước, sau đó mở rộng dần nếu cần thiết.
- Trong bài toán xây dựng kho dữ liệu cho doanh nghiệp này, chúng ta đã xác định phạm vi xây dựng kho dữ liệu là quản lý tổng số lượng mỗi mặt hàng được lưu trữ tại mỗi tỉnh, thành phố. Sự lựa chọn này có căn cứ từ những yếu tố sau:
 - + Quản lý tổng số lượng các mặt hàng còn lại tại tất cả các cửa hàng trong một thành phố đóng vai trò quan trọng trong việc điều hành kinh doanh, nhằm cung cấp thông tin hỗ trợ cho lãnh đạo quyết định về việc đáp ứng đầy đủ các yêu cầu của khách

hàng từ các cửa hàng trong tỉnh, thành phố mà khách hàng sinh sống. Mỗi đơn đặt hàng của khách hàng có thể yêu cầu số lượng và các mặt hàng khác nhau.

- + Thông tin về tổng số lượng mặt hàng cung cấp sự hỗ trợ cho nhà lãnh đạo để đưa ra quyết định kinh doanh hiệu quả, giảm thiểu chi phí, và luôn đảm bảo cân đối, điều chỉnh nguồn cung và xuất hàng cho khách hàng một cách ổn định.
- + Áp dụng kỹ thuật OLAP (Online Analytical Processing) để xây dựng khối dữ liệu phân tích về khách hàng, mặt hàng và các yếu tố khác của doanh nghiệp theo nhiều chiều.

Chương II: Yêu cầu nghiệp vụ

- Để đáp ứng được nhu cầu của khách hàng, hệ thống kho dữ liệu trích lọc dữ liệu từ hai cơ sở dữ liệu hiện có để cho vào kho dữ liệu và cung cấp các xử lý phân tích trực tuyến với các thao tác cuộn lên, khoan xuống, chọn và chiếu dựa trên những yêu cầu chọn lựa của khách hàng và các bảng theo chiều để đáp ứng yêu cầu của người sử dụng. Thiết lập một chiều thời gian, hệ thống sẽ sinh một báo cáo Phân tích trực tuyến cho những yêu cầu sau:
 1. Tìm tất cả các cửa hàng cùng với thành phố, bang, số điện thoại, mô tả, kích cỡ, trọng lượng và đơn giá của tất cả các mặt hàng được bán ở kho đó.
 2. Tìm tất cả các đơn đặt hàng với tên khách hàng và ngày đặt hàng được thực hiện bởi khách hàng đó
 3. Tìm tất cả các cửa hàng cùng với tên thành phố và số điện thoại mà có bán các mặt hàng được đặt bởi một khách hàng nào đó
 4. Tìm địa chỉ văn phòng đại diện với tên thành phố, bang của tất cả các cửa hàng lưu kho một mặt hàng nào đó với số lượng trên mức cụ thể.
 5. Với mỗi một đơn đặt hàng của khách, liệt kê các mặt hàng được đặt cùng với mô tả, mã cửa hàng, tên thành phố và các cửa hàng có bán mặt hàng đó.
 6. Tìm thành phố và bang mà một khách hàng nào đó sinh sống
 7. Tìm mức độ tồn kho của một mặt hàng cụ thể tại tất cả các cửa hàng ở một thành phố cụ thể nào đó
 8. Tìm các mặt hàng, số lượng đặt, khách hàng, cửa hàng và thành phố của một đơn đặt hàng.
 9. Tìm các khách hàng du lịch, khách hàng đặt theo đường bưu điện và khách hàng thuộc cả hai loại

Chương III: Đặc tả yêu cầu chức năng

1. Đặc tả đầu vào

- Kho dữ liệu được xây dựng trên nền tảng Microsoft SQL Server và sử dụng công cụ Power BI để trực quan hóa. Dữ liệu được trích xuất từ hệ thống cơ sở dữ liệu giao dịch của công ty, bao gồm thông tin về đơn hàng, khách hàng, sản phẩm, nhân viên và khu vực bán hàng.
- Cơ sở dữ liệu nguồn gồm các quan hệ sau:
 - + Cơ sở dữ liệu Văn phòng đại diện bao gồm những quan hệ với các lược đồ như sau:
 - Khách hàng (Mã KH, Tên KH, Mã Thành phố, Ngày đặt hàng đầu tiên)
 - Khách hàng du lịch (*Mã KH, Hướng dẫn viên du lịch, Thời gian)
 - Khách hàng bưu điện (*Mã KH, Địa chỉ bưu điện, Thời gian)
 - + Cơ sở dữ liệu Bán hàng gồm những quan hệ với lược đồ như sau:
 - Văn phòng đại diện (Mã Thành phố, Tên Thành phố, Địa chỉ VP, Bang, Thời gian)
 - Cửa hàng (Mã cửa hàng, *Mã Thành phố, Số điện thoại, Thời gian)
 - Mặt hàng (Mã MH, Mô tả, Kích cỡ, Trọng lượng, Giá, Thời gian)
 - Mặt hàng_được lưu trữ (*Mã cửa hàng, *Mã mặt hàng, Số lượng trong kho, Thời gian)
 - Đơn đặt hàng (Mã đơn, Ngày đặt hàng, Mã Khách hàng)
 - Sau khi trích xuất, dữ liệu được làm sạch, chuyển đổi (ETL) và nạp vào kho dữ liệu để phân tích. Dữ liệu được tổ chức thành các chiều (*dimension*) và sự kiện (*fact*) để dễ dàng phân tích OLAP.

2. Đặc tả đầu ra

- Tất cả các cửa hàng cùng với thành phố, bang, số điện thoại, mô tả, kích cỡ, trọng lượng và đơn giá của tất cả các mặt hàng được bán ở kho đó.
- Tất cả các đơn đặt hàng với tên khách hàng và ngày đặt hàng được thực hiện bởi khách hàng đó.
- Tất cả các cửa hàng cùng với tên thành phố và số điện thoại mà có bán các mặt hàng được đặt bởi một khách hàng nào đó.
- Địa chỉ văn phòng đại diện với tên thành phố, bang của tất cả các cửa hàng lưu kho một mặt hàng nào đó với số lượng trên mức cụ thể.
- Với mỗi một đơn đặt hàng của khách, liệt kê các mặt hàng được đặt cùng với mô tả, mã cửa hàng, tên thành phố và các cửa hàng có bán mặt hàng đó.
- Thành phố và bang mà một khách hàng nào đó sinh sống.
- Mức độ tồn kho của một mặt hàng cụ thể tại tất cả các cửa hàng ở một thành phố cụ thể nào đó.

- Các mặt hàng, số lượng đặt, khách hàng, cửa hàng và thành phố của một đơn đặt hàng.
- Các khách hàng du lịch, khách hàng đặt theo đường bưu điện và khách hàng thuộc cả hai loại.
- Các báo cáo (bảng dữ liệu động) cho phép người dùng thực hiện các thao tác roll up, drill down, slice, dice, cung cấp thông tin để hỗ trợ người quản lý ra quyết định.
- Kho dữ liệu được chuyển hóa, phân ra thành những mục tiêu như khách hàng, bán hàng, sản phẩm,... nhờ công cụ OLAP (Online Analytical Processing) để hỗ trợ việc truy vấn.
- Bản báo cáo thống kê được trình bày ở dạng bảng và đồ thị (sử dụng Power BI).

3. Công cụ hỗ trợ phân tích

- Hệ thống sử dụng mô hình OLAP để xây dựng kho dữ liệu và hỗ trợ truy vấn.
- Báo cáo và phân tích dữ liệu được thực hiện qua công cụ Power BI, hỗ trợ người dùng dễ dàng tương tác, lọc, chọn dữ liệu và đưa ra quyết định kinh doanh hiệu quả.

Chương IV: Thiết kế kho dữ liệu

1. Chuyển đổi lược đồ quan hệ sang mô hình thực thể liên kết mở rộng:

- Ta có lược đồ quan hệ của hệ thống cơ sở dữ liệu quan hệ hiện tại của doanh nghiệp này như sau:
 - + Cơ sở dữ liệu Văn phòng đại diện bao gồm những quan hệ với các lược đồ như sau:
 - Khách hàng (Mã KH, Tên KH, Mã Thành phố, Ngày đặt hàng đầu tiên)
 - Khách hàng du lịch (*Mã KH, Hướng dẫn viên du lịch, Thời gian)
 - Khách hàng bưu điện (* Mã KH, Địa chỉ bưu điện, Thời gian)
 - + Cơ sở dữ liệu Bán hàng gồm những quan hệ với lược đồ như sau:
 - Văn phòng đại diện (Mã Thành phố, Tên Thành phố, Địa chỉ VP, Bang, Thời gian)
 - Cửa hàng (Mã cửa hàng, * Mã Thành phố, Số điện thoại, Thời gian)
 - Mặt hàng (Mã MH, Mô tả, Kích cỡ, Trọng lượng, Giá, Thời gian)
 - Mặt hàng_được lưu trữ (* Mã cửa hàng, * Mã mặt hàng, Số lượng trong kho, Thời gian)
 - Đơn đặt hàng (Mã đơn, Ngày đặt hàng, Mã Khách hàng)
 - Mặt hàng được đặt (* Mã đơn, * Mã mặt hàng, Số lượng đặt, Giá đặt, Thời gian)
- **Bước 1:** Xác định kiểu quan hệ, khóa và các trường
 - + Cơ sở dữ liệu Văn phòng đại diện:

Relation Name	RelType	Primary Type	KAP	KAG	FKA	NKA
Khách hàng	PR1	Mã KH				Tên khách hàng, Mã Thành phố, Ngày đặt hàng đầu tiên
Khách hàng du lịch	PR2	Mã KH	Mã KH			Hướng dẫn viên du lịch, Thời gian
Khách hàng bưu điện	PR2	Mã KH	Mã KH			Địa chỉ bưu điện, Thời gian

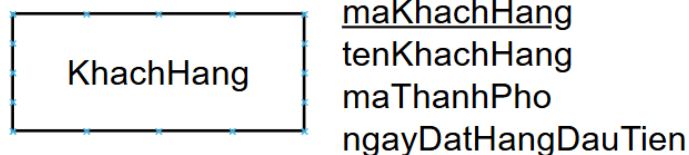
- + Cơ sở dữ liệu bán hàng:

Relation Name	RelType	Primary Type	KAP	KAG	FKA	NKA
Văn phòng đại diện	PR1	Mã Thành phố	Mã Thành phố			Tên Thành phố, Địa chỉ VP, Bang, Thời gian
Cửa hàng	PR1	Mã cửa hàng	Mã cửa hàng		Mã Thành phố	Số điện thoại, Thời gian
Mặt hàng	PR1	Mã MH	Mã MH			Mô tả, Kích cỡ, Trọng lượng, Giá, Thời gian
Mặt hàng được lưu trữ	SR1	Mã cửa hàng, Mã mặt hàng		Mã cửa hàng, Mã mặt hàng		Số lượng trong kho, Thời gian
Đơn đặt hàng	PR1	Mã đơn	Mã đơn			Ngày đặt hàng, Mã Khách hàng
Mặt hàng được đặt	SR1	Mã đơn, Mã mặt hàng		Mã đơn, Mã mặt hàng		Số lượng đặt, Giá đặt, Thời gian

– **Bước 2:** Chuyển đổi mỗi PR1 thành một thực thể

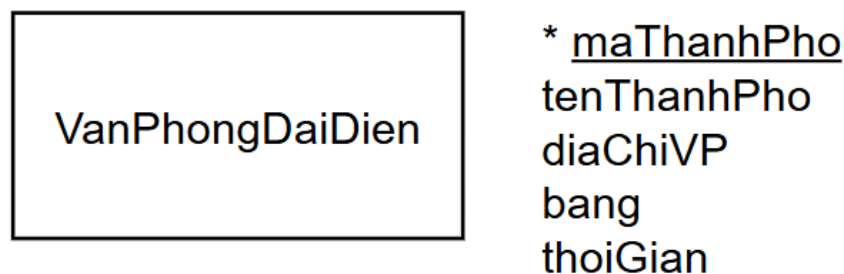
+ Cơ sở dữ liệu Văn phòng đại diện:

- Thực thể Khách hàng:

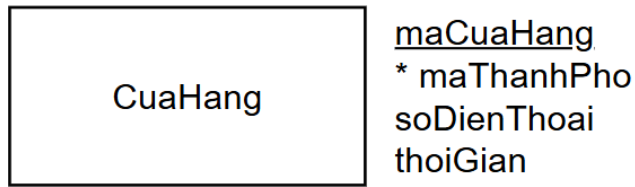


+ Cơ sở dữ liệu bán hàng:

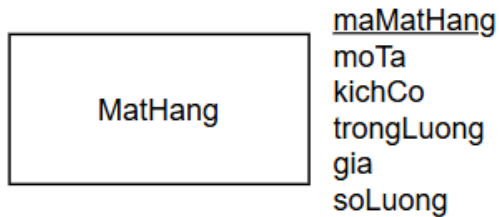
- Thực thể Văn phòng đại diện



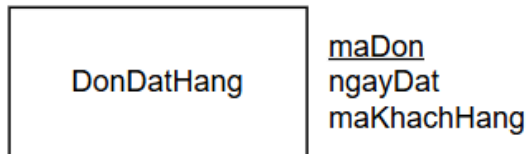
- Thực thể Cửa hàng:



- Thực thể MatHang

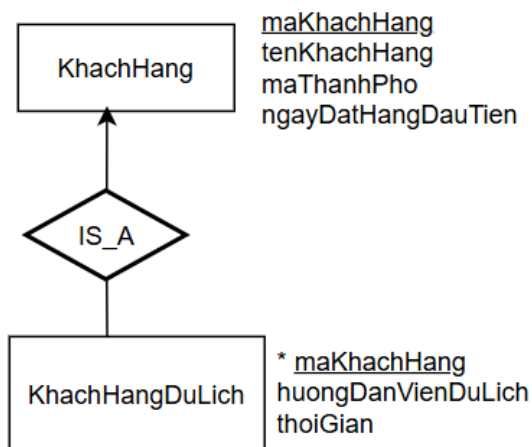


- Thực thể DonDatHang



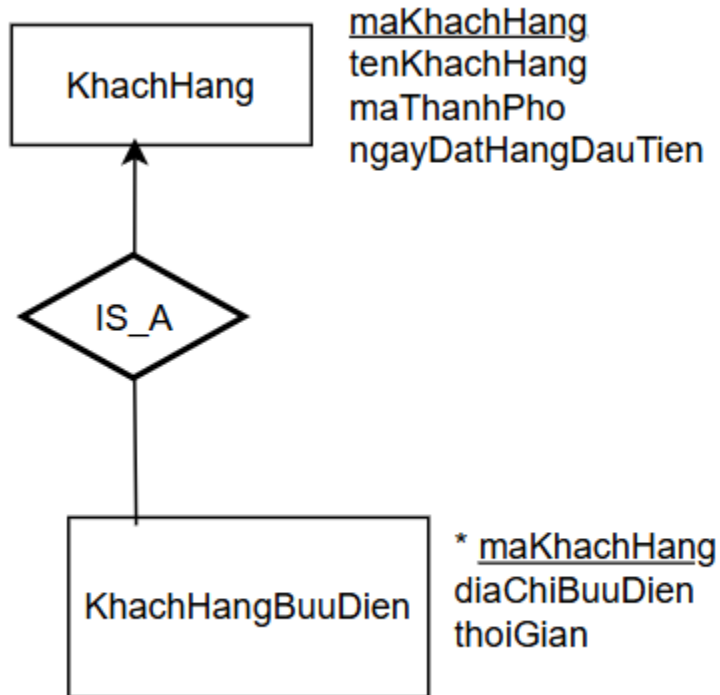
- **Bước 3** Ánh xạ mỗi PR2 sang một thực thể lớp con hoặc là một thực thể yếu
- + Cơ sở dữ liệu Văn phòng đại diện:

- Xét Khách hàng du lịch, ta có
 - o Khách hàng (Mã KH, Tên KH, Mã Thành phố, Ngày đặt hàng đầu tiên)
 - o Khách hàng du lịch (*Mã KH, Hướng dẫn viên du lịch, Thời gian)
 - ➔ Khóa của Khách hàng du lịch chính là khóa của Khách hàng, không có thuộc tính định danh riêng trong khóa
 - ➔ Khách hàng du lịch là thực thể con của Khách hàng:

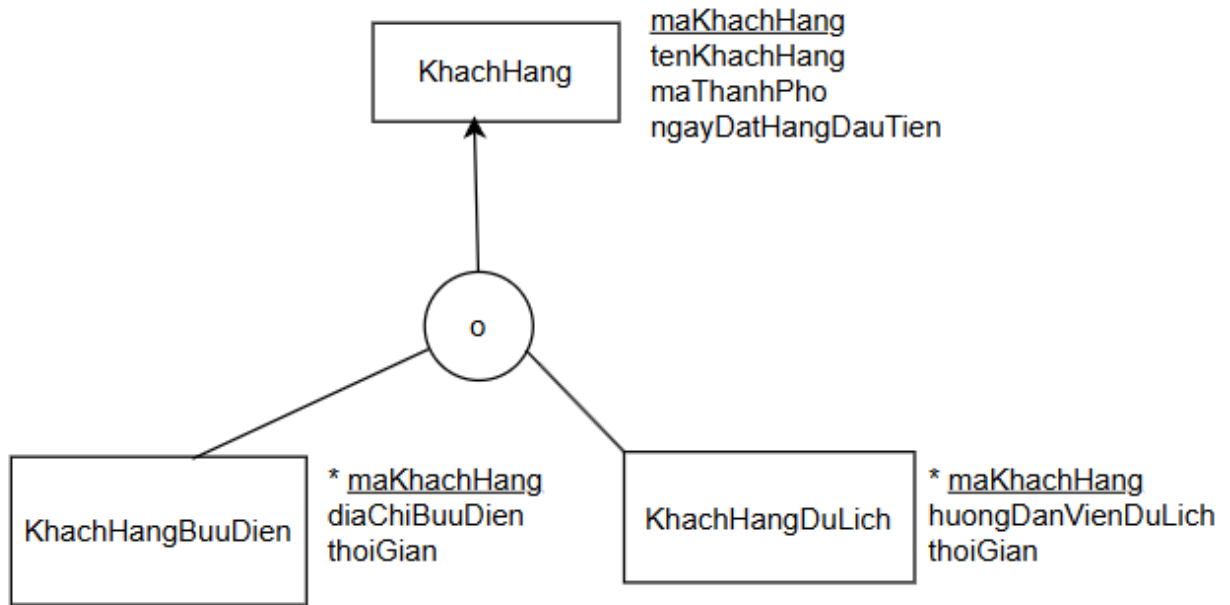


- Xét Khách hàng buro điện, ta có

- o Khách hàng (Mã KH, Tên KH, Mã Thành phố, Ngày đặt hàng đầu tiên)
- o Khách hàng bưu điện (* Mã KH, Địa chỉ bưu điện, Thời gian)
- ➔ Khóa của Khách hàng bưu điện chính là khóa của Khách hàng, không có thuộc tính định danh riêng trong khóa
- ➔ Khách hàng bưu điện là Thực thể con của Khách hàng:

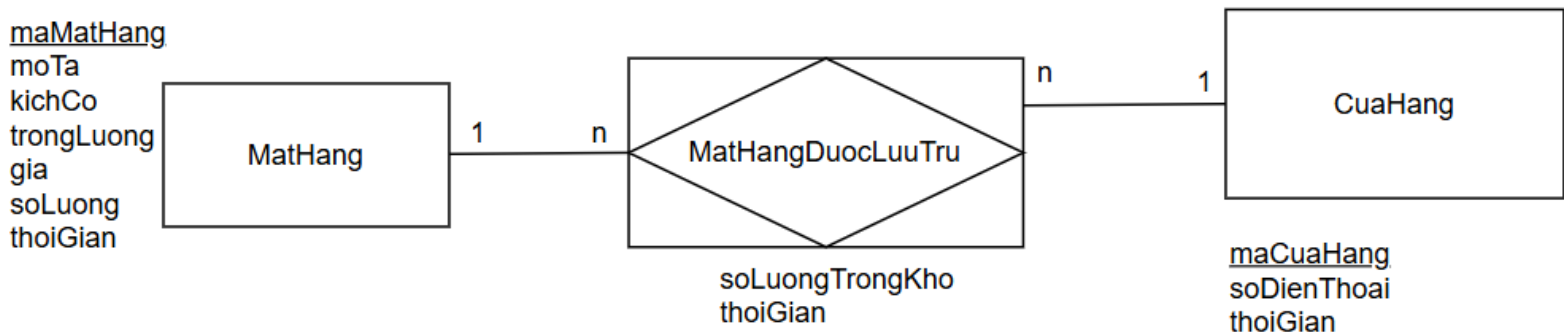


- Ta thấy 2 thực thể con Khách hàng bưu điện và Khách hàng du lịch đều là thực thể con của Khách hàng, Khách hàng bưu điện có thể là Khách hàng du lịch và ngược lại
- ➔ Ngữ nghĩa khái quát hóa giao nhau:



- + Cơ sở dữ liệu bán hàng: không có
- **Bước 4:** Ánh xạ mỗi SR1 sang quan hệ hai ngôi hoặc quan hệ nhiều ngôi
- + Cơ sở dữ liệu Văn phòng đại diện: không có
- + Cơ sở dữ liệu bán hàng:
 - Mặt hàng được lưu trữ:
 - o Cửa hàng (Mã cửa hàng, * Mã Thành phố, Số điện thoại, Thời gian)
 - o Mặt hàng (Mã MH, Mô tả, Kích cỡ, Trọng lượng, Giá, Thời gian)
 - o Mặt hàng_được lưu trữ (* Mã cửa hàng, * Mã mặt hàng, Số lượng trong kho, Thời gian)

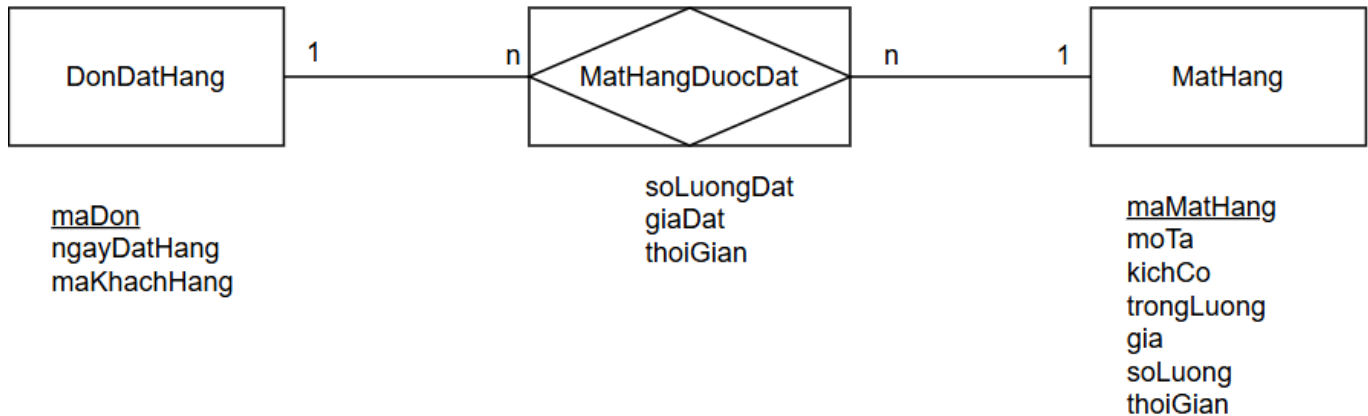
➔ Như vậy ta có quan hệ 2 ngôi:



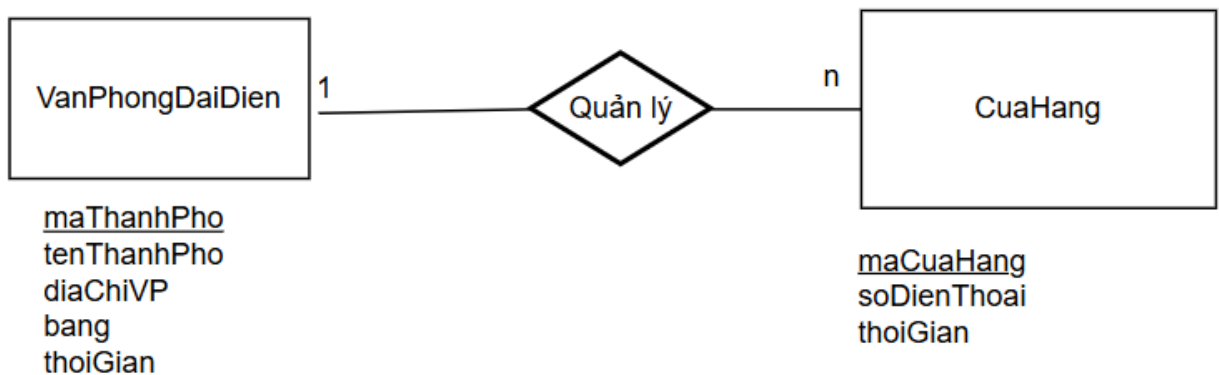
- Mặt hàng được đặt:
 - o Mặt hàng (Mã MH, Mô tả, Kích cỡ, Trọng lượng, Giá, Thời gian)
 - o Đơn đặt hàng (Mã đơn, Ngày đặt hàng, Mã Khách hàng)

- o Mặt hàng được đặt (* Mã đơn, * Mã mặt hàng, Số lượng đặt, Giá đặt, Thời gian)

➔ Như vậy ta có quan hệ 2 ngôi:



- **Bước 5:** Ánh xạ mỗi SR2 sang quan hệ hai ngôi hoặc quan hệ nhiều ngôi: không có
 - **Bước 6:** Ánh xạ mỗi FKA sang 1 mối quan hệ
 - + Cơ sở dữ liệu Văn phòng đại diện: không có
 - + Cơ sở dữ liệu bán hàng:
 - Cửa hàng (Mã cửa hàng, * Mã Thành phố, Số điện thoại, Thời gian) chứa FKA trở đến Văn phòng đại diện (Mã Thành phố, Tên Thành phố, Địa chỉ VP, Bang, Thời gian)
- ➔ Quan hệ quản lý: một Văn phòng đại diện quản lý nhiều cửa hàng



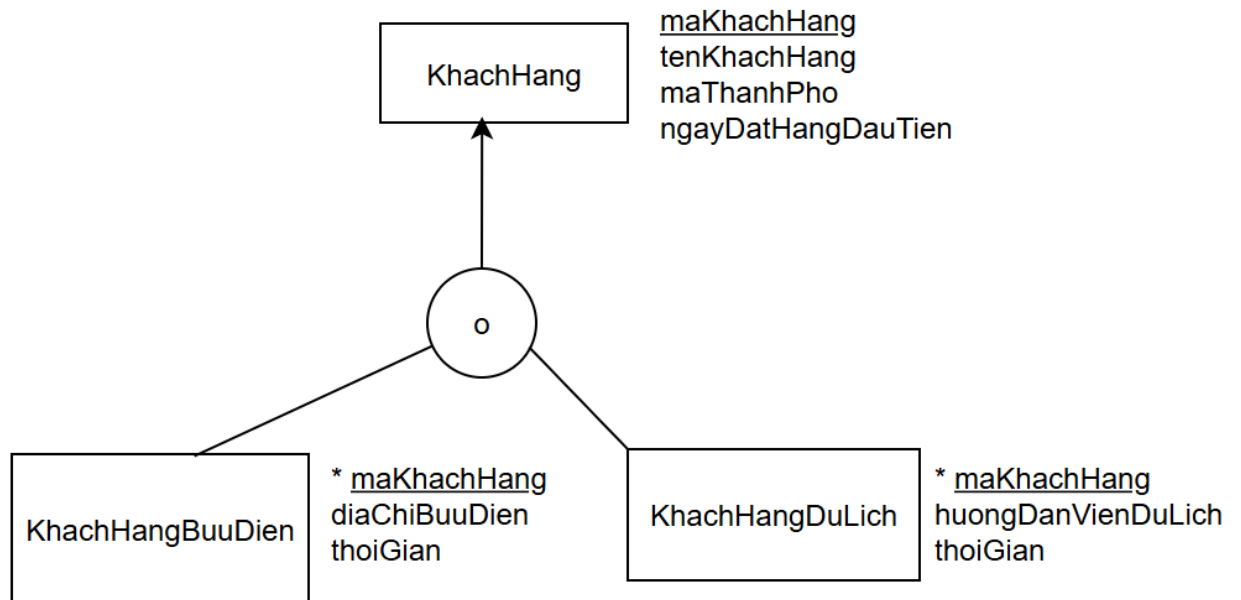
- **Bước 7:** Ánh xạ ý nghĩa phụ thuộc bao hàm sang ngữ nghĩa
 - + Cơ sở dữ liệu Văn phòng đại diện:
 - + Cơ sở dữ liệu bán hàng:

Các phụ thuộc bao hàm phái sinh	Ngữ nghĩa suy ra
---------------------------------	------------------

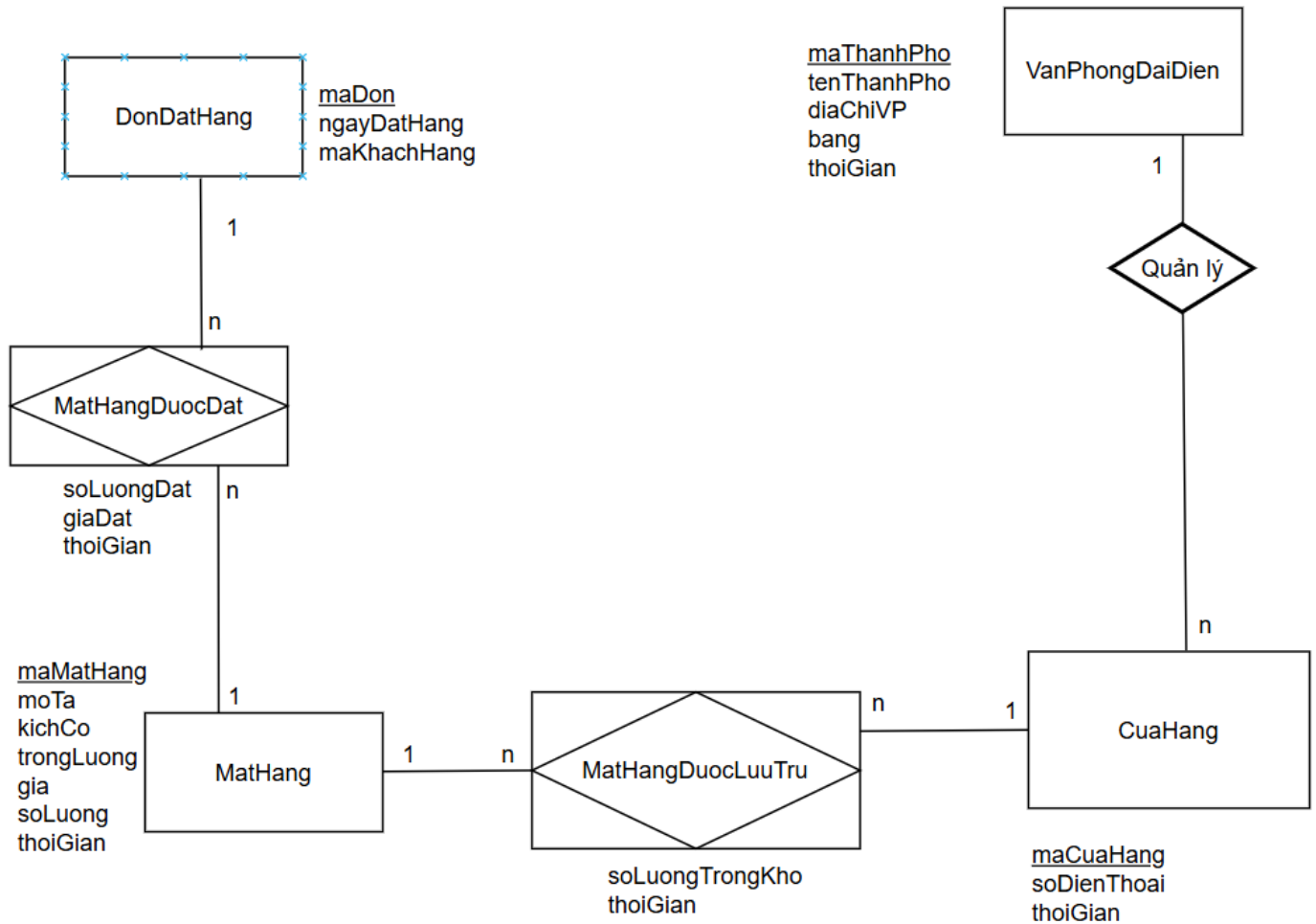
MatHangDuocDat.maDon \subseteq DonDatHang.maDon MatHangDuocDat.maHang \subseteq MatHang.maHang	DonHang và MatHang là quan hệ nhiều - nhiều
MatHangDuocLuuTru.maCuaHang \subseteq CuaHang.maCuaHang MatHangDuocLuuTru.maHang \subseteq MatHang.maHang	MatHang và CuaHang là quan hệ nhiều – nhiều

– **Bước 8:** Vẽ mô hình EER với những kết quả thu được từ các bước trên

+ Cơ sở dữ liệu Văn phòng đại diện:



+ Cơ sở dữ liệu bán hàng:



2. Tích hợp 2 mô hình thực thể liên kết thành một mô hình thực thể liên kết tích hợp IER:

– **Bước 1:** Giải quyết xung đột giữa các lược đồ ER:

+ Trường hợp 1: Giải quyết xung đột về đồng âm và đồng nghĩa

- MatHangDuocDat.thoiGian và DonDatHang.ngayDat có cùng ý nghĩa, vì các mặt hàng được đặt của 1 đơn hàng đều có ngày đặt giống nhau

➔ Thông tin DonDatHang.ngayDat là đủ, không cần

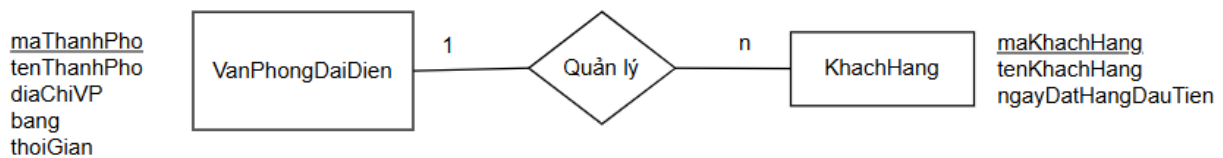
➔ Giữ lại DonDatHang.ngayDat, bỏ MatHangDuocDat.thoiGian

- Các thực thể Khách hàng du lịch, Khách hàng bưu điện, Văn phòng đại diện, Mặt hàng, Cửa hàng, Mặt hàng lưu trữ, Mặt hàng được đặt đều có thuộc tính Thời gian nhưng ý nghĩa khác nhau (hiện tượng đồng âm).

➔ Đổi tên thuộc tính:

- Khách hàng du lịch: Thời điểm du lịch
- Khách hàng bưu điện: Ngày mở tài khoản

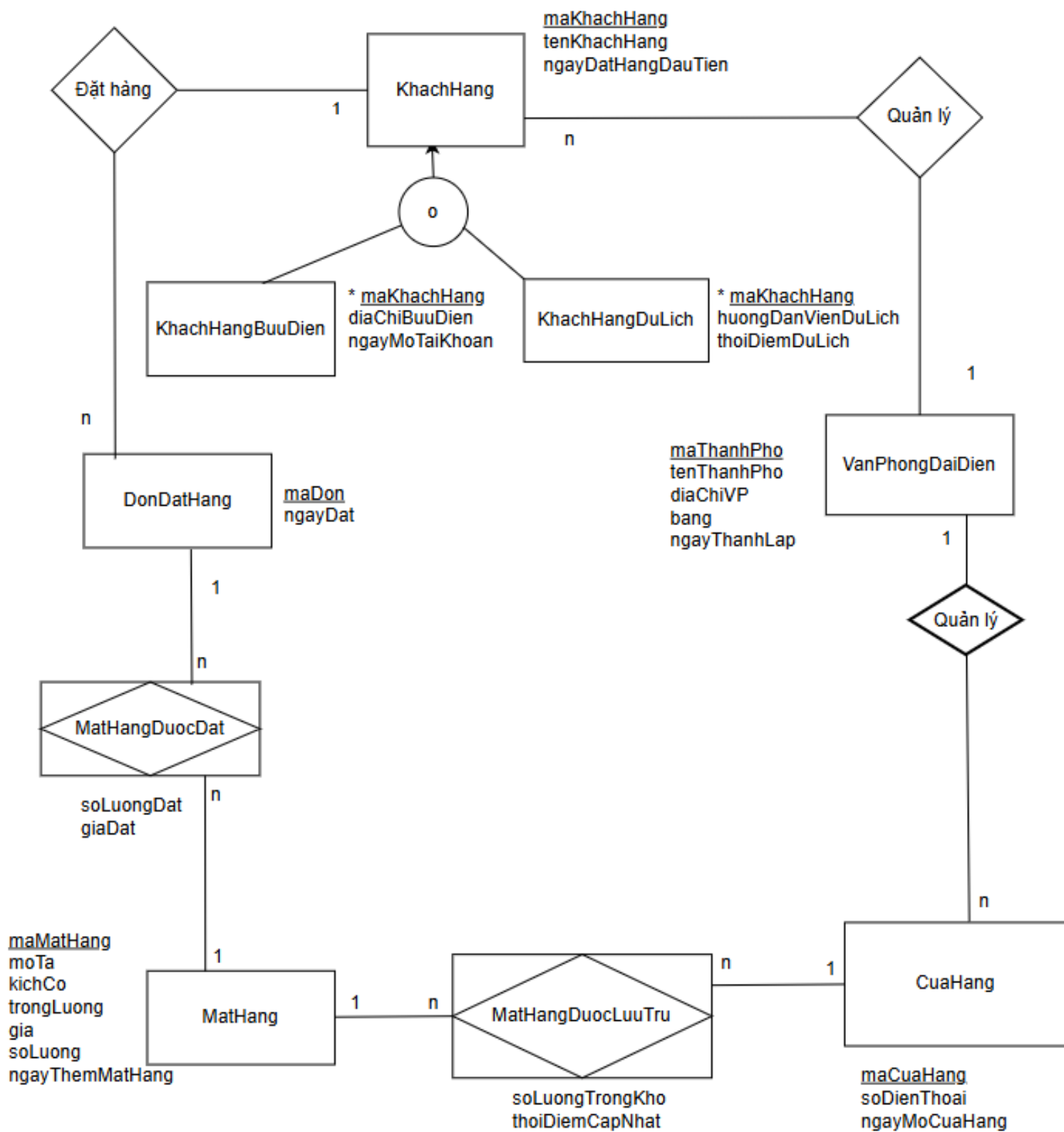
- Văn phòng đại diện: Ngày thành lập
- Cửa hàng: Ngày mở cửa hàng
- Mặt hàng: Thời điểm được thêm
- Mặt hàng được lưu trữ: Thời điểm cập nhật
- + Trường hợp 2: Giải quyết xung đột kiểu dữ liệu (coi như không có xung đột)
- + Trường hợp 3: Giải quyết xung đột khóa: không có
- + Trường hợp 4: Giải quyết xung đột về lực lượng: không có
- + Trường hợp 5: Giải quyết xung đột của thực thể yếu: không có
- + Trường hợp 6: Giải quyết xung đột trên thực thể kiểu con: không có
- **Bước 2:** Trộn các thực thể:
 - + Trường hợp 1: Trộn các tập thực thể bằng phép hợp: không có
 - + Trường hợp 2: Trộn EER bằng cách khái quát hóa: không có
 - + Trường hợp 3: Trộn EER bằng quan hệ loại con: không có
 - + Trường hợp 4: Trộn EER bằng cách dùng thực thể tích hợp (Aggregation): không có
 - + Trường hợp 5: Trộn EER bằng cách dùng phân loại: không có
 - + Trường hợp 6: Trộn EER bằng cách dùng quan hệ hai ngôi:
 - Khách hàng có chứa thuộc tính Mã TP không phải thuộc tính khóa, Mã TP lại là khóa chính của Văn phòng đại diện
 - ➔ Khách hàng và Văn phòng đại diện là quan hệ 1 nhiều:



- Đơn hàng chứa thuộc tính Mã KH không phải thuộc tính khóa, Mã KH là khóa của Khách hàng
- ➔ Khách hàng và Đơn hàng là quan hệ 1 nhiều:



- **Bước 3:** Trộn các quan hệ
 - + Trường hợp 1: Trộn các quan hệ bằng quan hệ cha con (không có)
 - + Trường hợp 2: Trộn các quan hệ bằng khái quát hóa trùng lặp : không có
 - + Trường hợp 3: Sát nhập các quan hệ ít ngôi thành một quan hệ có ngôi cao hơn: không có



3. Nghiên cứu yêu cầu của kho dữ liệu:

3.1. Nghiên cứu yêu cầu của kho dữ liệu:

1. *Tìm tất cả các cửa hàng cùng với thành phố, bang, số điện thoại, mô tả, kích cỡ, trọng lượng và đơn giá của tất cả các mặt hàng được bán ở kho đó.*

➔ Yêu cầu tìm kiếm thông tin, không cần xây dựng bảng fact

2. *Tìm tất cả các đơn đặt hàng với tên khách hàng và ngày đặt hàng được thực hiện bởi khách hàng đó*

➔ Yêu cầu tìm kiếm thông tin, không cần xây dựng bảng fact

3. *Tìm tất cả các cửa hàng cùng với tên thành phố và số điện thoại mà có bán các mặt hàng được đặt bởi một khách hàng nào đó*

➔ Yêu cầu tìm kiếm thông tin, không cần xây dựng bảng fact

4. *Tìm địa chỉ văn phòng đại diện với tên thành phố, bang của tất cả các cửa hàng lưu kho một mặt hàng nào đó với lượng trên mức cụ thể.*

4.1. *Chọn mặt hàng + nhập số lượng*

4.2. *Hiển thị địa chỉ văn phòng đại diện với tên thành phố, bang*

➔ Yêu cầu tìm kiếm thông tin, không cần xây dựng bảng fact

5. *Với mỗi một đơn đặt hàng của khách, liệt kê các mặt hàng được đặt cùng với mô tả, mã cửa hàng, tên thành phố và các cửa hàng có bán mặt hàng đó.*

➔ Yêu cầu tìm kiếm thông tin, không cần xây dựng bảng fact

6. *Tìm thành phố và bang mà một khách hàng nào đó sinh sống*

➔ Yêu cầu tìm kiếm thông tin, không cần xây dựng bảng fact

7. *Tìm mức độ tồn kho của một mặt hàng cụ thể tại tất cả các cửa hàng ở một thành phố cụ thể nào đó.*

➔ Mục tiêu: giúp người quản lý xác định được mặt nào thường xuyên bị hết tại 1 thành phố cụ thể để kịp thời điều chỉnh số lượng nhập tại cửa hàng đó; điều chuyển các đơn hàng sang các thành phố đang còn đơn hàng.

➔ Cần xây dựng bảng fact: FactTonKho

8. *Tìm các mặt hàng, số lượng đặt, khách hàng, cửa hàng và thành phố của một đơn đặt hàng.*

➔ Yêu cầu tìm kiếm thông tin, không cần xây dựng bảng fact

9. *Tìm các khách hàng du lịch, khách hàng đặt theo đường bưu điện và khách hàng thuộc cả hai loại*

➔ Yêu cầu tìm kiếm thông tin, không cần xây dựng bảng fact

Để hỗ trợ việc ra quyết định cho nhà quản lý: cần tập trung bán các loại mặt hàng nào tại thành phố nào cho khách hàng nào và tại thời điểm nào

➔ Cần xây dựng bảng fact: FactDoanhThu

- Các chiều: DimKhachHang, DimMatHang, DimVanPhongDaiDien, DimThoiGian

- Độ đo:
 - tongSoLuong: tổng số lượng của một mặt hàng được mua bởi một khách hàng, tại 1 thời điểm, tại một thành phố (tương ứng với 1 văn phòng đại diện)
 - tongDoanhThu: tổng doanh thu của một mặt hàng được mua bởi một khách hàng, tại 1 thời điểm, tại một thành phố (tương ứng với 1 văn phòng đại diện)
- Ý nghĩa của bảng fact: mỗi dòng thông tin trong bảng FactDoanhThu mô tả tổng số lượng và tổng doanh thu của một mặt hàng được mua bởi một khách hàng, tại 1 thời điểm, tại một thành phố (tương ứng với 1 văn phòng đại diện)

3.2. Mô hình thiết kế kho dữ liệu:

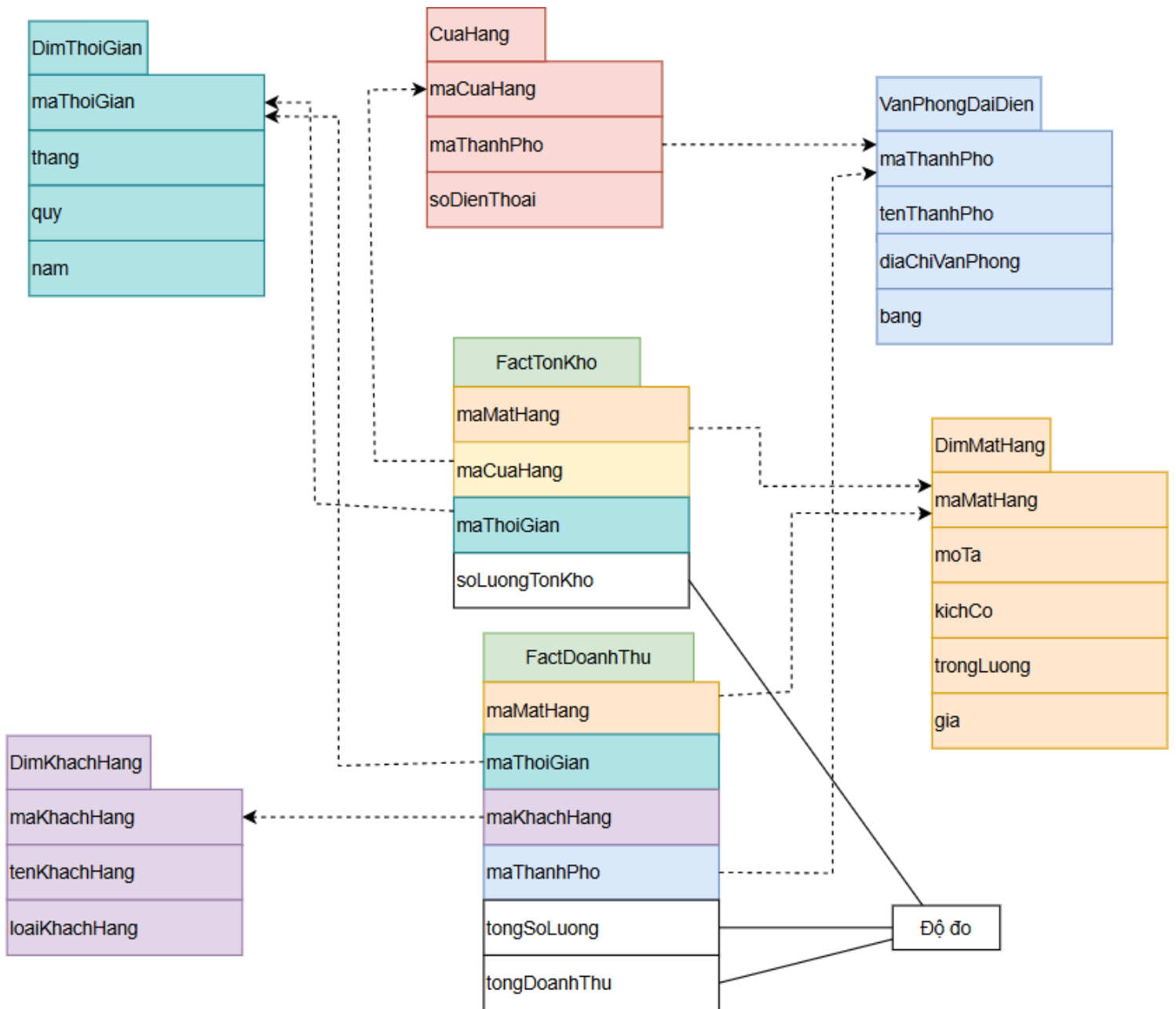
– Bảng FactTonKho

- + Các chiều chính: chiều mặt hàng, chiều cửa hàng, chiều thời gian
- + Chiều thứ cấp: chiều văn phòng đại diện
- + Thuộc tính:
 - maThoiGian
 - maCuaHang
 - maMatHang
 - Độ đo: soLuongTonKho: số lượng còn lại trong kho của một mặt hàng tại một cửa hàng tại một thời điểm
- + Ý nghĩa của bảng fact: mỗi dòng thông tin trong bảng FactMatHangTonKho mô tả số lượng còn lại của một mặt hàng tại một cửa hàng tại một thời điểm

– Bảng FactDoanhThu

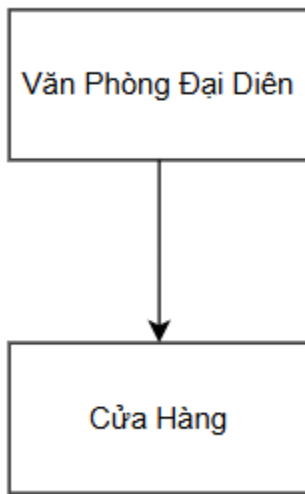
- + Các chiều: chiều mặt hàng, chiều khách hàng, chiều văn phòng đại diện, chiều thời gian
- + Thuộc tính:
 - maKhachHang
 - maThoiGian
 - maVanPhongDaiDien
 - maMatHang
 - Độ đo:
 - tongSoLuong: tổng số lượng của một mặt hàng được mua bởi một khách hàng, tại 1 thời điểm, tại một thành phố (tương ứng với 1 văn phòng đại diện)
 - tongDoanhThu: tổng doanh thu của một mặt hàng được mua bởi một khách hàng, tại 1 thời điểm, tại một thành phố (tương ứng với 1 văn phòng đại diện)

- + Ý nghĩa của bảng fact: mỗi dòng thông tin trong bảng FactDoanhThu mô tả tổng số lượng và tổng doanh thu của một mặt hàng được mua bởi một khách hàng, tại 1 thời điểm, tại một thành phố (tương ứng với 1 văn phòng đại diện)
- **DimMatHang:** chứa thông tin của MatHang bao gồm:
 - + moTa
 - + kíchCo
 - + trongLuong
 - + gia
- **DimThoiGian:** mỗi bản ghi trong bảng thể hiện 1 tháng trong năm bao gồm:
 - + Tháng
 - + Quý
 - + Năm
- **DimCuaHang:** mỗi bản ghi trong bảng thể hiện thông tin 1 CuaHang bao gồm:
 - + maCuaHang
 - + maThanhPho
 - + soDienThoai
- **DimVanPhongDaiDien:** mỗi bản ghi trong bảng thể hiện thông tin của 1 VanPhongDaiDien:
 - + maThanhPho
 - + tenThanhPho
 - + diaChiVanPhong
 - + bang
- **DimKhachHang:** Mỗi bản ghi trong bảng thể hiện thông tin 1 khách hàng bao gồm:
 - + tenKhachHang
 - + maKhachHang
 - + loaiKhachHang

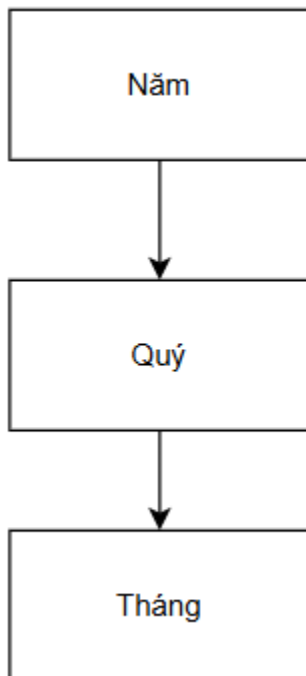


3.3. Phân cấp cho các chiều dữ liệu:

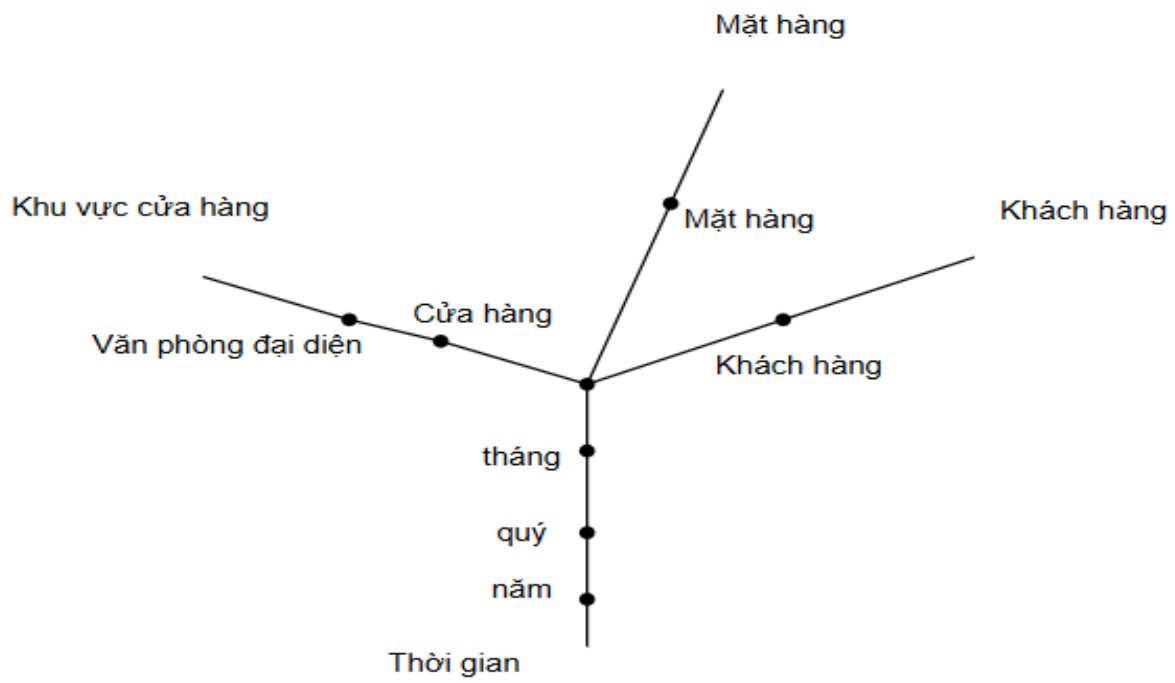
- Chiều khu vực cửa hàng: 2 cấp là văn phòng đại diện -> cửa hàng



- Chiều thời gian: 3 cấp là năm -> quý -> tháng



- Chiều khách hàng: 1 cấp: khách hàng
- Chiều mặt hàng: 1 cấp: mặt hàng



Chương V: Cài đặt các khối dữ liệu

1. Tạo 2 cơ sở dữ liệu hiện tại của doanh nghiệp:

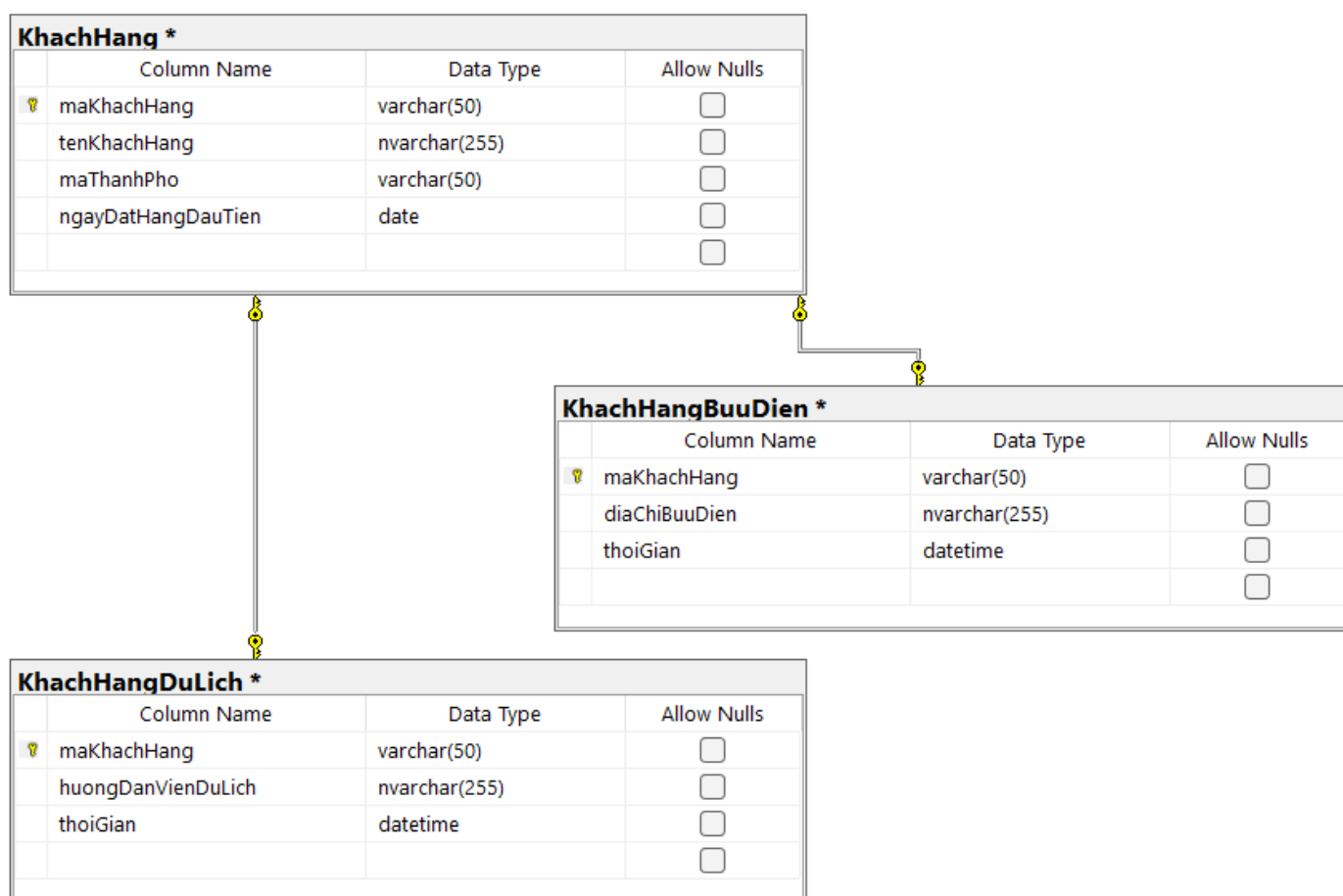
1.1. Cơ sở dữ liệu Văn phòng đại diện:

- Tạo Cơ sở dữ liệu Văn phòng đại diện:
 - + Tên cơ sở dữ liệu: VANPHONGDAIDIEN_DB_1
 - + Các bảng:

Tên Bảng	Tên Thuộc Tính	Kiểu Dữ Liệu	Khóa chính / Ngoại
KhachHang	maKhachHang	VARCHAR(50)	Khóa chính
	tenKhachHang	NVARCHAR(255)	
	maThanhPho	VARCHAR(50)	
	ngayDatHangDauTien	DATE	
KhachHangBuuDien	maKhachHang	VARCHAR(50)	Khóa chính, Khóa ngoại (KhachHang)
	diaChiBuuDien	NVARCHAR(255)	
	thoiGian	DATETIME	
KhachHangDuLich	maKhachHang	VARCHAR(50)	Khóa chính, Khóa ngoại (KhachHang)

	huongDanVienDuLich	NVARCHAR(255)	
	thoiGian	DATETIME	

– Cơ sở dữ liệu Văn phòng sau khi tạo xong:



1.2.Cơ sở dữ liệu Bán hàng:

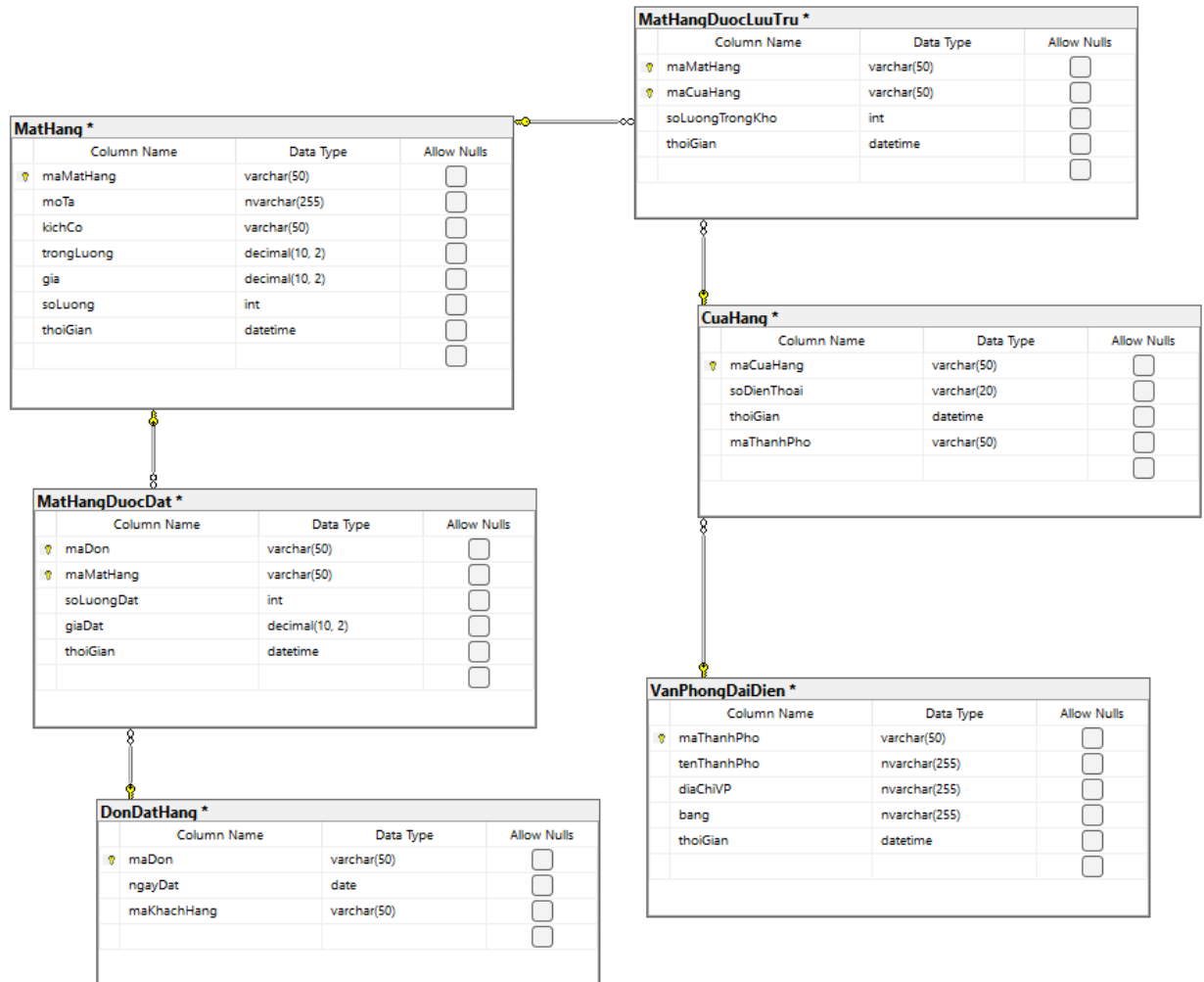
- Tạo Cơ sở dữ liệu Bán hàng
 - + Tên cơ sở dữ liệu: BANHANG_DB_1
 - + Các bảng:

Tên Bảng	Tên Thuộc Tính	Kiểu Dữ Liệu	Khóa chính / Ngoại	Ghi chú
DonDatHang	maDon	VARCHAR(50)	Khóa chính	Mã đơn đặt hàng
	ngayDat	DATE		Ngày đặt hàng
	maKhachHang	VARCHAR(50)	Khóa ngoại (KhachHang, nếu có)	Mã khách hàng
MatHang	maMatHang	VARCHAR(50)	Khóa chính	Mã mặt hàng
	moTa	NVARCHAR(255)		Mô tả mặt hàng
	kichCo	VARCHAR(50)		Kích cỡ mặt hàng
	trongLuong	DECIMAL(10, 2)		Trọng lượng
	gia	DECIMAL(10, 2)		Giá bán
	soLuong	INT		Số lượng có sẵn
	thoiGian	DATETIME		Thời điểm thêm mặt hàng

MatHangDuocDat	maDon	VARCHAR(50)	Khóa chính, Khóa ngoại (DonDatHang)	Mã đơn
	maMatHang	VARCHAR(50)	Khóa chính, Khóa ngoại (MatHang)	Mã mặt hàng được đặt
	soLuongDat	INT		Số lượng đặt
	giaDat	DECIMAL(10, 2)		Giá tại thời điểm đặt
	thoiGian	DATETIME		Thời điểm đặt hàng
VanPhongDaiDien	maThanhPho	VARCHAR(50)	Khóa chính	Mã thành phố
	tenThanhPho	NVARCHAR(255)		Tên thành phố
	diaChiVP	NVARCHAR(255)		Địa chỉ văn phòng đại diện
	bang	NVARCHAR(255)		Bang (state - nếu theo chuẩn US)
	thoiGian	DATETIME		Ngày thành lập

CuaHang	maCuaHang	VARCHAR(50)	Khóa chính	Mã cửa hàng
	soDienThoai	VARCHAR(20)		Số điện thoại cửa hàng
	thoiGian	DATETIME		Ngày mở cửa hàng
	maThanhPho	VARCHAR(50)	Khóa ngoại (VanPhongDaiDien)	Mã thành phố quản lý cửa hàng
MatHangDuocLuuTru	maMatHang	VARCHAR(50)	Khóa chính, Khóa ngoại (MatHang)	Mã mặt hàng được lưu trữ
	maCuaHang	VARCHAR(50)	Khóa chính, Khóa ngoại (CuaHang)	Mã cửa hàng lưu trữ mặt hàng
	soLuongTrongKho	INT		Số lượng hàng tồn trong kho
	thoiGian	DATETIME		Thời điểm cập nhật

– Cơ sở dữ liệu BANHANG_DB_1 sau khi tạo xong:



1.3.Sinh dữ liệu cho 2 CSDL:

– Mục tiêu:

- + Tạo câu lệnh SQL INSERT gồm dữ liệu giả để nạp vào hai cơ sở dữ liệu trên
- + Giữ các quan hệ logic, ví dụ như maKhachHang trong bảng DonDatHang của BANHANG_DB_1 phải tồn tại trong bảng KháchHang trong CSDL VANPHONGDAIDIEN_DB_1

– Để tạo ra các câu lệnh SQL:

- + Thay vì viết các câu lệnh SQL INSERT một cách thủ công (rất tốn thời gian và dễ mắc lỗi), ta sử dụng **Python** để tạo ra các câu lệnh SQL
- + Để sinh dữ liệu một cách ngẫu nhiên:
 - Dùng thư viện random
 - o Tạo các dữ liệu dạng số giả như khối lượng, ngày, ...
 - Thư viện Faker:

- o Áp dụng thư viện Faker để tạo ra dữ liệu giả nhưng có tính thực tế cao (ví dụ: tên người, địa chỉ, số điện thoại)
 - o Faker cung cấp nhiều "Provider" (nhà cung cấp) cho các loại dữ liệu khác nhau, giúp tạo ra dữ liệu đa dạng và có ý nghĩa.
 - o Trong code, Faker('vi_VN') được sử dụng để tạo dữ liệu tiếng Việt, thể hiện khả năng hỗ trợ đa ngôn ngữ của Faker.
- **Các bước tạo câu lệnh SQL:**
- + Sinh dữ liệu:
 - Tạo dữ liệu VanPhongDaiDien:

```

1 vanphongdaidien_banhang_data = []
2 for i in range(NUM_CITIES):
3     city_id = f"TP{i+1:03d}"
4     city_name = fake.city()
5     address = fake.street_address()
6     state = fake.state()
7     timestamp = (datetime.now() - timedelta(days=random.randint(2000, 2200))).date()
8     vanphongdaidien_banhang_data.append((city_id, city_name, address, state, timestamp))

```

- Tạo dữ liệu KháchHang có maThanhPho lấy từ VanPhongDaiDien:

```

1 khachhang_vanphong_data = []
2 city_ids_banhang = [city[0] for city in vanphongdaidien_banhang_data]
3 for i in range(NUM_CUSTOMERS):
4     customer_id = f"KH{i+1:05d}"
5     customer_name = fake.name()
6     assigned_city_id = random.choice(city_ids_banhang)
7     first_order_date = fake.date_between(start_date='-5y', end_date='today')
8     khachhang_vanphong_data.append((customer_id, customer_name, assigned_city_id, first_order_date))

```

- Tạo dữ liệu KháchHangBuuDien và KháchHangDuLich có maKhachHang lấy từ KháchHang


```

1 khachhangbuudien_vanphong_data = []
2 buudien_customers_vanphong = random.sample(khachhang_vanphong_data, k=min(int(NUM_CUSTOMERS * 0.3), NUM_CUSTOMERS))
3 for customer in buudien_customers_vanphong:
4     customer_id = customer[0]
5     address = fake.street_address() + ", " + fake.city()
6     timestamp = (datetime.now() - timedelta(days=random.randint(10, 2000))).date()
7     khachhangbuudien_vanphong_data.append((customer_id, address, timestamp))

```

- Tạo dữ liệu cho CuaHang có maThanhPho lấy random từ VanPhongDaiDien:

```

1 cuahang_banhang_data = []
2 for city_id, __, __, __ in vanphongdaidien_banhang_data:
3     for i in range(NUM_STORES_PER_CITY):
4         store_id = f"CH{city_id[2:]}{i+1:02d}"
5         phone = fake.phone_number()
6         timestamp = (datetime.now() - timedelta(days=random.randint(2000, 2200))).date()
7         cuahang_banhang_data.append((store_id, phone, timestamp, city_id))

```

- Tạo dữ liệu MatHang:

```

1 mathang_banhang_data = []
2 for i in range(NUM_PRODUCTS):
3     product_id = f"MH{i+1:05d}"
4     description = fake.word().capitalize() + " " + fake.bs()
5     size = random.choice(['Lon', 'Vua', 'Nho'])
6     weight = round(random.uniform(0.1, 10.0), 2)
7     price = round(random.uniform(10000, 500000), 2)
8     stock_quantity = random.randint(0, 1000)
9     timestamp = (datetime.now() - timedelta(days=random.randint(10, 2000)))
10    mathang_banhang_data.append((product_id, description, size, weight, price, stock_quantity))

```

- Tạo dữ liệu DonDatHang có maKhachHang lấy từ KhachHang:



```
1 dondathang_banhang_data = []
2 customer_ids_vanphong = [c[0] for c in khachhang_vanphong_data]
3 for i in range(NUM_ORDERS):
4     order_id = f"DH{i+1:06d}"
5     order_date = fake.date_between(start_date='-5y', end_date='today')
6     customer_id = random.choice(customer_ids_vanphong)
7     dondathang_banhang_data.append((order_id, order_date, customer_id))
8
```

- MatHangDuocDat lấy ngẫu nhiên 2 ID của DonDatHang và MatHang



```
1 mathangduocdat_banhang_data = []
2 product_ids_banhang = [p[0] for p in mathang_banhang_data]
3 for order_id, _ in dondathang_banhang_data:
4     num_items = random.randint(1, MAX_ITEMS_PER_ORDER)
5     selected_products = random.sample(product_ids_banhang, k=min(num_items, len(product_ids_banhang)))
6     for product_id in selected_products:
7         quantity = random.randint(1, 10)
8         product_price = next(p[4] for p in mathang_banhang_data if p[0] == product_id)
9         item_price = product_price
10        timestamp = order_date
11        mathangduocdat_banhang_data.append((order_id, product_id, quantity, item_price, timestamp))
```

- MatHangDuocLuuTru lấy ngẫu nhiên 2 ID của MatHang và CuaHang



```
1 print("Đang sinh dữ liệu cho MatHangDuocLuuTru (dành cho BANHANG_DB)...")
2 mathangduocluutru_banhang_data = []
3 store_ids_banhang = [s[0] for s in cuahang_banhang_data]
4 for store_id in store_ids_banhang:
5     num_products_in_store = random.randint(50, 200)
6     stocked_products = random.sample(product_ids_banhang, k=min(num_products_in_store, len(product_ids_banhang)))
7     for product_id in stocked_products:
8         stock_quantity = random.randint(10, 500)
9         timestamp = datetime.now() - timedelta(days=random.randint(0, 30))
10        mathangduocluutru_banhang_data.append((product_id, store_id, stock_quantity, timestamp))
```

+ Hàm sinh câu lệnh SQL:

```

1  def generate_insert_sql(table_name, columns, data):
2      sql_statements = []
3      column_names = ", ".join(columns)
4      for row in data:
5          formatted_values = []
6          for value in row:
7              if isinstance(value, str):
8                  escaped_value = value.replace("'", "'")
9                  formatted_values.append(f"'{escaped_value}'")
10             elif isinstance(value, datetime):
11                 formatted_values.append(f"'{value.strftime('%Y-%m-%d %H:%M:%S')}'")
12             elif isinstance(value, (int, float, complex)):
13                 formatted_values.append(str(value))
14             elif value is None:
15                 formatted_values.append("NULL")
16             else:
17                 formatted_values.append(f"'{value}'")
18             values = ", ".join(formatted_values)
19             sql_statements.append(f"INSERT INTO {table_name} ({column_names}) VALUES ({values});")
20     return sql_statements

```

- + Cách câu lệnh sql lưu vào vanphongdaidien_db_inserts.sql, banhang_db_inserts.sql
- **Thêm dữ liệu:** Chạy 2 file vanphongdaidien_db_inserts.sql, banhang_db_inserts.sql trong SQL để thêm dữ liệu

2. Tạo cơ sở dữ liệu tích hợp:

- Tên cơ sở dữ liệu: TICHHOP_DB_1
- Các bảng:

Tên Bảng	Tên Thuộc Tính	Kiểu Dữ Liệu	Khóa chính / Ngoại	Ghi chú
VanPhongDaiDien	maThanhPho	VARCHAR(50)	Khóa chính	Mã thành phố
	tenThanhPho	NVARCHAR(255)		Tên thành phố
	diaChiVP	NVARCHAR(255)		Địa chỉ văn phòng
	bang	NVARCHAR(255)		Bang (khu vực)
	ngayThanhLap	DATE		Ngày thành lập
CuaHang	maCuaHang	VARCHAR(50)	Khóa chính	Mã cửa hàng
	maThanhPho	VARCHAR(50)	Khóa ngoại	Thành phố của cửa hàng
	soDienThoai	VARCHAR(20)		Số điện thoại
	ngayMoCuaHang	DATE		Ngày mở cửa hàng

MatHang	maMatHang	VARCHAR(50)	Khóa chính	Mã mặt hàng
	moTa	NVARCHAR(255)		Mô tả
	kichCo	VARCHAR(50)		Kích cỡ
	trongLuong	DECIMAL(10, 2)		Trọng lượng
	gia	DECIMAL(10, 2)		Giá
	ngayThemMatHang	DATE		Ngày thêm mặt hàng
MatHangDuocLuuTru	maCuaHang	VARCHAR(50)	Khóa chính, Khóa ngoại	Mã cửa hàng
	maMatHang	VARCHAR(50)	Khóa chính, Khóa ngoại	Mã mặt hàng
	soLuongTrongKho	INT		Số lượng trong kho
	thoiDiemCapNhat	DATETIME		Thời điểm cập nhật
KhachHang	maKhachHang	VARCHAR(50)	Khóa chính	Mã khách hàng
	maThanhPho	VARCHAR(50)	Khóa ngoại	Thành phố
	tenKhachHang	NVARCHAR(255)		Tên khách hàng
	ngayDatHangDauTien	DATE		Ngày đặt hàng đầu tiên
KhachHangDuLich	maKhachHang	VARCHAR(50)	Khóa chính, Khóa ngoại	Mã khách hàng
	huongDanVienDuLich	NVARCHAR(255)		Hướng dẫn viên
	thoiDiemDuLich	DATE		Thời điểm du lịch
KhachHangBuuDien	maKhachHang	VARCHAR(50)	Khóa chính, Khóa ngoại	Mã khách hàng
	diaChiBuuDien	NVARCHAR(255)		Địa chỉ bưu điện
	ngayMoTaiKhoan	DATE		Ngày mở tài khoản
DonDatHang	maDon	VARCHAR(50)	Khóa chính	Mã đơn hàng
	ngayDat	DATE		Ngày đặt
	maKhachHang	VARCHAR(50)	Khóa ngoại	Mã khách hàng
MatHangDuocDat	maDon	VARCHAR(50)	Khóa chính, Khóa ngoại	Mã đơn hàng
	maMatHang	VARCHAR(50)	Khóa chính, Khóa ngoại	Mã mặt hàng
	soLuongDat	INT		Số lượng đặt
	giaDat	DECIMAL(10, 2)		Giá đặt

- Thêm dữ liệu vào TICHHOP_DB_1
 - + **VanPhongDaiDien**
 - Nguồn: BANHANG_DB_1.dbo.VanPhongDaiDien
 - Đích: TICHHOP_DB_1.dbo.VanPhongDaiDien
 - Ánh xạ:
 - o maThanhPho → maThanhPho
 - o tenThanhPho → tenThanhPho
 - o diaChiVP → diaChiVP
 - o bang → bang
 - o thoiGian → ngayThanhLap (dùng CONVERT(DATE, thoiGian) để chuyển kiểu về ngày)
 - + **KhachHang**
 - Nguồn: VANPHONGDAIDIEN_DB_1.dbo.KhachHang
 - Đích: TICHHOP_DB_1.dbo.KhachHang
 - Ánh xạ:
 - o maKhachHang → maKhachHang
 - o tenKhachHang → tenKhachHang
 - o maThanhPho → maThanhPho
 - o ngayDatHangDauTien → ngayDatHangDauTien
 - + **KhachHangBuuDien**
 - Nguồn: VANPHONGDAIDIEN_DB_1.dbo.KhachHangBuuDien
 - Đích: TICHHOP_DB_1.dbo.KhachHangBuuDien
 - Ánh xạ:
 - o maKhachHang → maKhachHang
 - o diaChiBuuDien → diaChiBuuDien
 - o thoiGian → ngayMoTaiKhoan (dùng CONVERT(DATE, thoiGian))
 - + **KhachHangDuLich**
 - Nguồn: VANPHONGDAIDIEN_DB_1.dbo.KhachHangDuLich
 - Đích: TICHHOP_DB_1.dbo.KhachHangDuLich
 - Ánh xạ:
 - o maKhachHang → maKhachHang
 - o huongDanVienDuLich → huongDanVienDuLich
 - o thoiGian → thoiDiemDuLich (dùng CONVERT(DATE, thoiGian))
 - + **CuaHang**
 - Nguồn: BANHANG_DB_1.dbo.CuaHang
 - Đích: TICHHOP_DB_1.dbo.CuaHang
 - Ánh xạ:
 - o maCuaHang → maCuaHang
 - o maThanhPho → maThanhPho
 - o soDienThoai → soDienThoai

- o thoiGian → ngayMoCuaHang (dùng CONVERT(DATE, thoiGian))
- + **MatHang**
 - Nguồn: BANHANG_DB_1.dbo.MatHang
 - Đích: TICHHOP_DB_1.dbo.MatHang
 - Ánh xạ:
 - o maMatHang → maMatHang
 - o moTa → moTa
 - o kichCo → kichCo
 - o trongLuong → trongLuong
 - o gia → gia
 - o thoiGian → ngayThemMatHang (dùng CONVERT(DATE, thoiGian))
- + **MatHangDuocLuuTru**
 - Nguồn: BANHANG_DB_1.dbo.MatHangDuocLuuTru
 - Đích: TICHHOP_DB_1.dbo.MatHangDuocLuuTru
 - Ánh xạ:
 - o maCuaHang → maCuaHang
 - o maMatHang → maMatHang
 - o soLuongTrongKho → soLuongTrongKho
 - o thoiGian → thoiDiemCapNhat
- + **DonDatHang**
 - Nguồn: BANHANG_DB_1.dbo.DonDatHang
 - Đích: TICHHOP_DB_1.dbo.DonDatHang
 - Ánh xạ:
 - o maDon → maDon
 - o ngayDat → ngayDat
 - o maKhachHang → maKhachHang
- + **MatHangDuocDat**
 - Nguồn: BANHANG_DB_1.dbo.MatHangDuocDat
 - Đích: TICHHOP_DB_1.dbo.MatHangDuocDat
 - Ánh xạ:
 - o maDon → maDon
 - o maMatHang → maMatHang
 - o soLuongDat → soLuongDat
 - o giaDat → giaDat

3. Tạo kho dữ liệu:

3.1.Tạo CSDL KDL_DB_1:

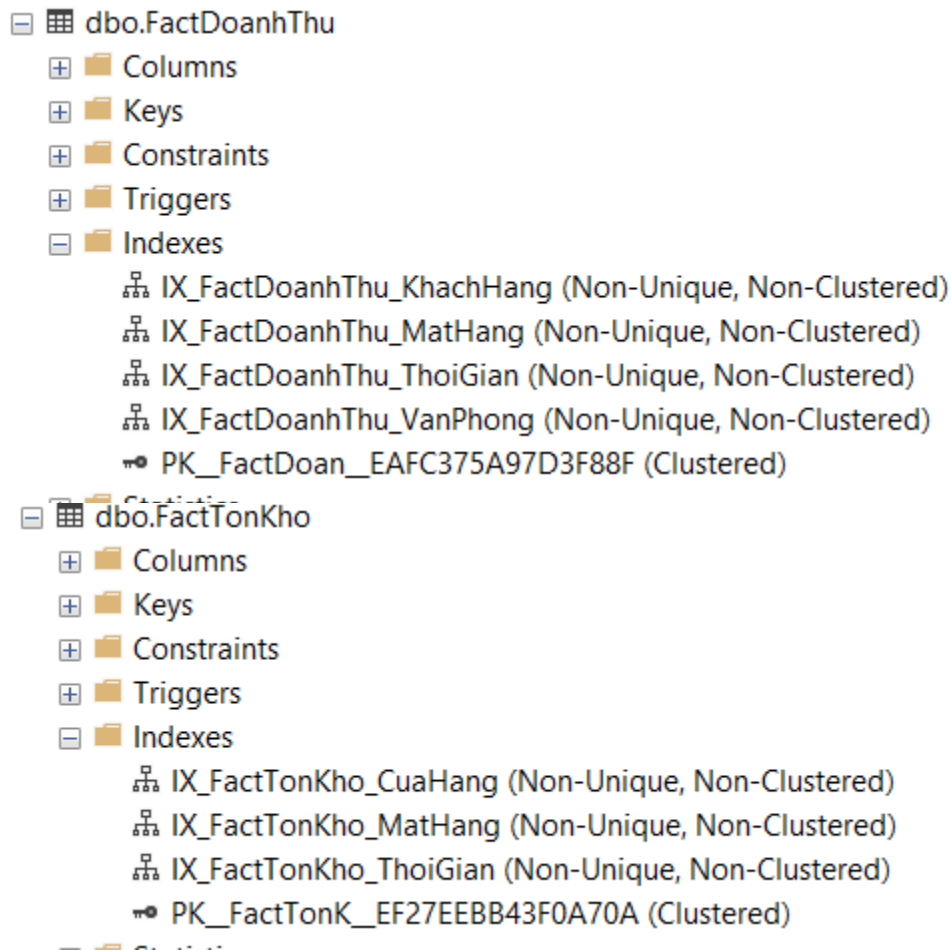
Tên Bảng	Tên Thuộc Tính	Kiểu Dữ Liệu	Khóa chính / Ngoại	Ghi chú
DimThoiGian	maThoiGian	int	Khóa chính (Primary Key)	Mã thời gian duy nhất

	thang	int		Tháng
	quy	int		Quý
	nam	int		Năm
DimKhachHang	maKhachHang	varchar(50)	Khóa chính (Primary Key)	Mã khách hàng
	tenKhachHang	nvarchar(255)		Tên khách hàng
	loaiKhachHang	nvarchar(255)		Loại khách hàng (Du lịch, Bưu điện, Khác)
DimMatHang	maMatHang	varchar(50)	Khóa chính (Primary Key)	Mã mặt hàng
	moTa	nvarchar(255)		Mô tả mặt hàng
	kichCo	nvarchar(255)		Kích cỡ
	trongLuong	float		Trọng lượng
	gia	float		Giá
DimVanPhongDaiDien	maThanhPho	varchar(50)	Khóa chính (Primary Key)	Mã thành phố
DimVanPhongDaiDien	tenThanhPho	nvarchar(255)		Tên thành phố
	diaChiVanPhong	nvarchar(255)		Địa chỉ văn phòng đại diện
	bang	nvarchar(255)		Bang (bang hành chính)
DimCuaHang	maCuaHang	varchar(50)	Khóa chính (Primary Key)	Mã cửa hàng
	maThanhPho	varchar(50)	Khóa ngoại (FK)	FK đến DimVanPhongDaiDien(maThanhPho)

	soDienThoai	varchar(20)		Số điện thoại
FactDoanhThu	maMatHang	varchar(50)	Khóa ngoại (FK)	FK đến DimMatHang(maMatHang)
	maThoiGian	int	Khóa ngoại (FK)	FK đến DimThoiGian(maThoiGian)
	maKhachHang	varchar(50)	Khóa ngoại (FK)	FK đến DimKhachHang(maKhachHang)
	maThanhPho	varchar(50)	Khóa ngoại (FK)	FK đến DimVanPhongDaiDien(maThanhPho)
	tongSoLuong	float		Tổng số lượng bán được
	tongDoanhThu	float		Tổng doanh thu
FactTonKho	maMatHang	varchar(50)	Khóa ngoại (FK)	FK đến DimMatHang(maMatHang)
	maCuaHang	varchar(50)	Khóa ngoại (FK)	FK đến DimCuaHang(maCuaHang)
	maThoiGian	int	Khóa ngoại (FK)	FK đến DimThoiGian(maThoiGian)
	soLuongTonKho	float		Số lượng tồn kho

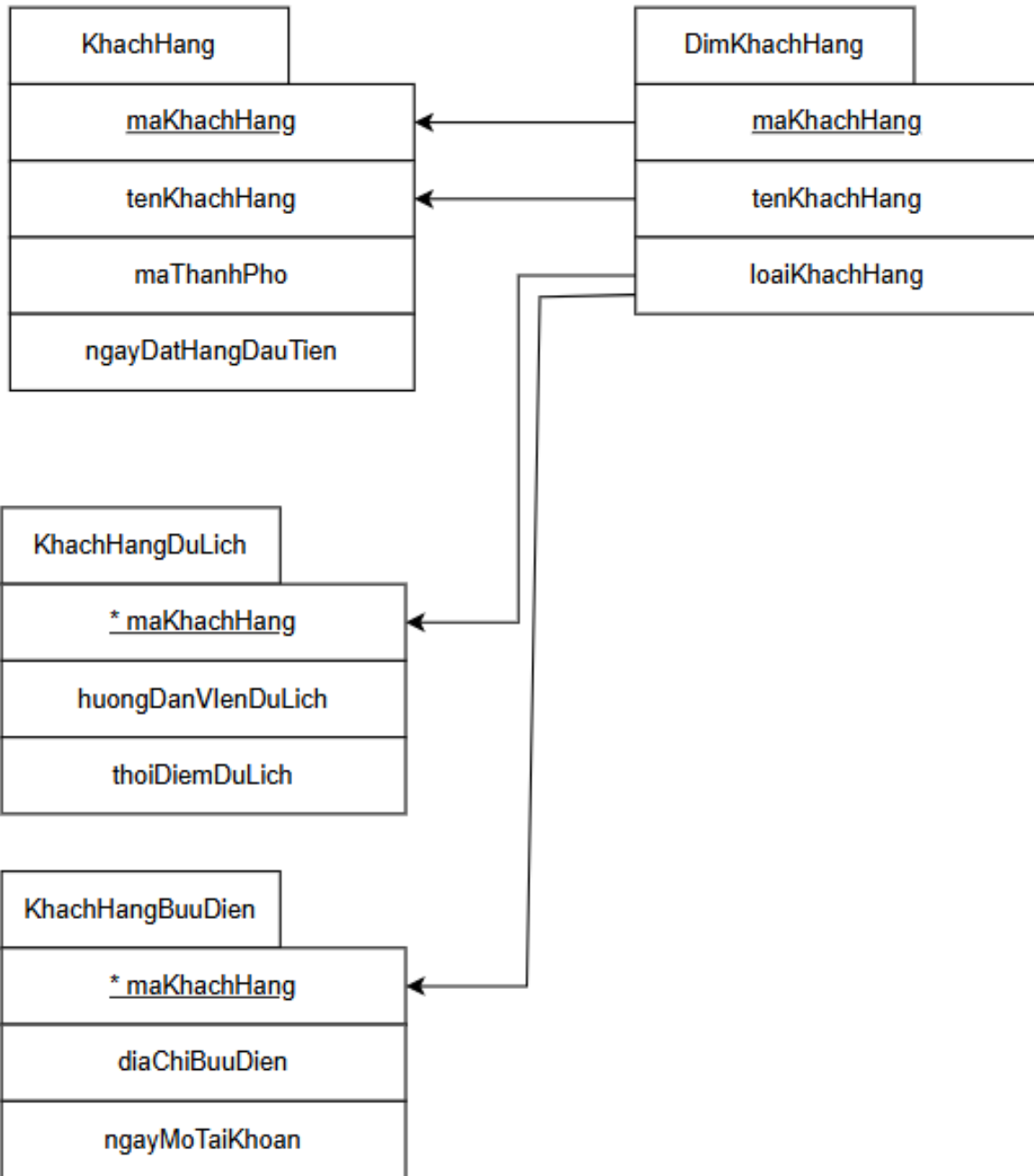
3.2. Tạo metadata và index:

- Metadata đã được tạo sẵn trong cơ sở dữ liệu
- Tạo index để tối ưu hóa trong việc truy vấn
 - + Các bảng DimKhachHang, DimMatHang, DimCuaHang, DimVanPhongDaiDien, DimThoiGian đều có clustered index cho các khóa chính của nó
 - + Tạo Non-Clustered Index cho 2 bảng fact:

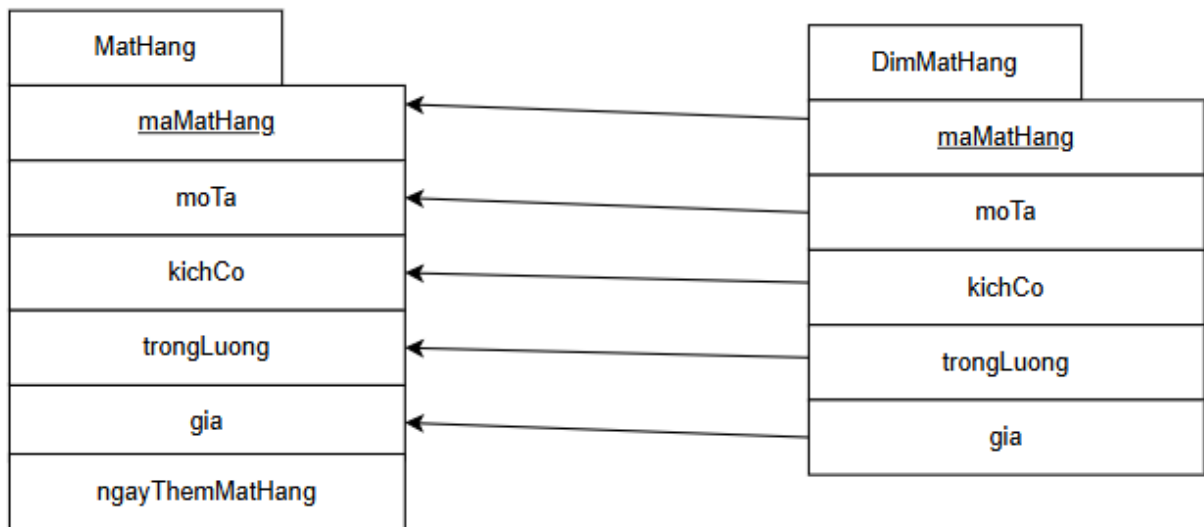


3.3. Ảnh xạ từ CSDL tích hợp sang kho:

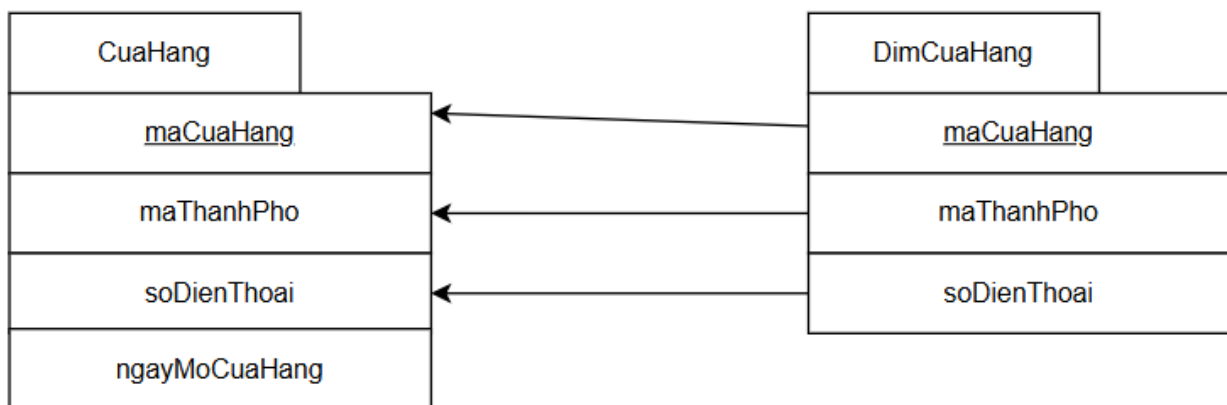
- DimKhachHang:



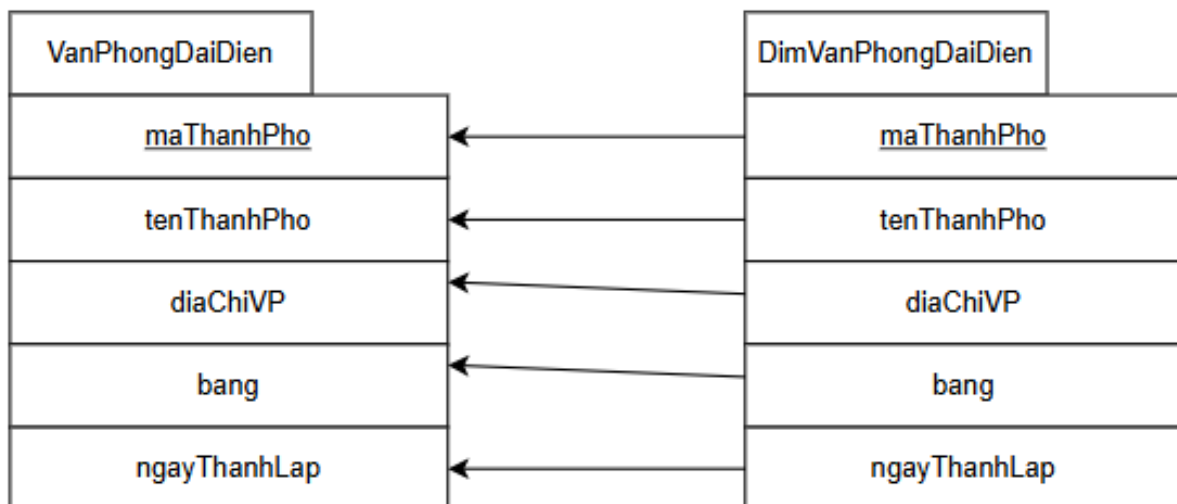
– DimMatHang:



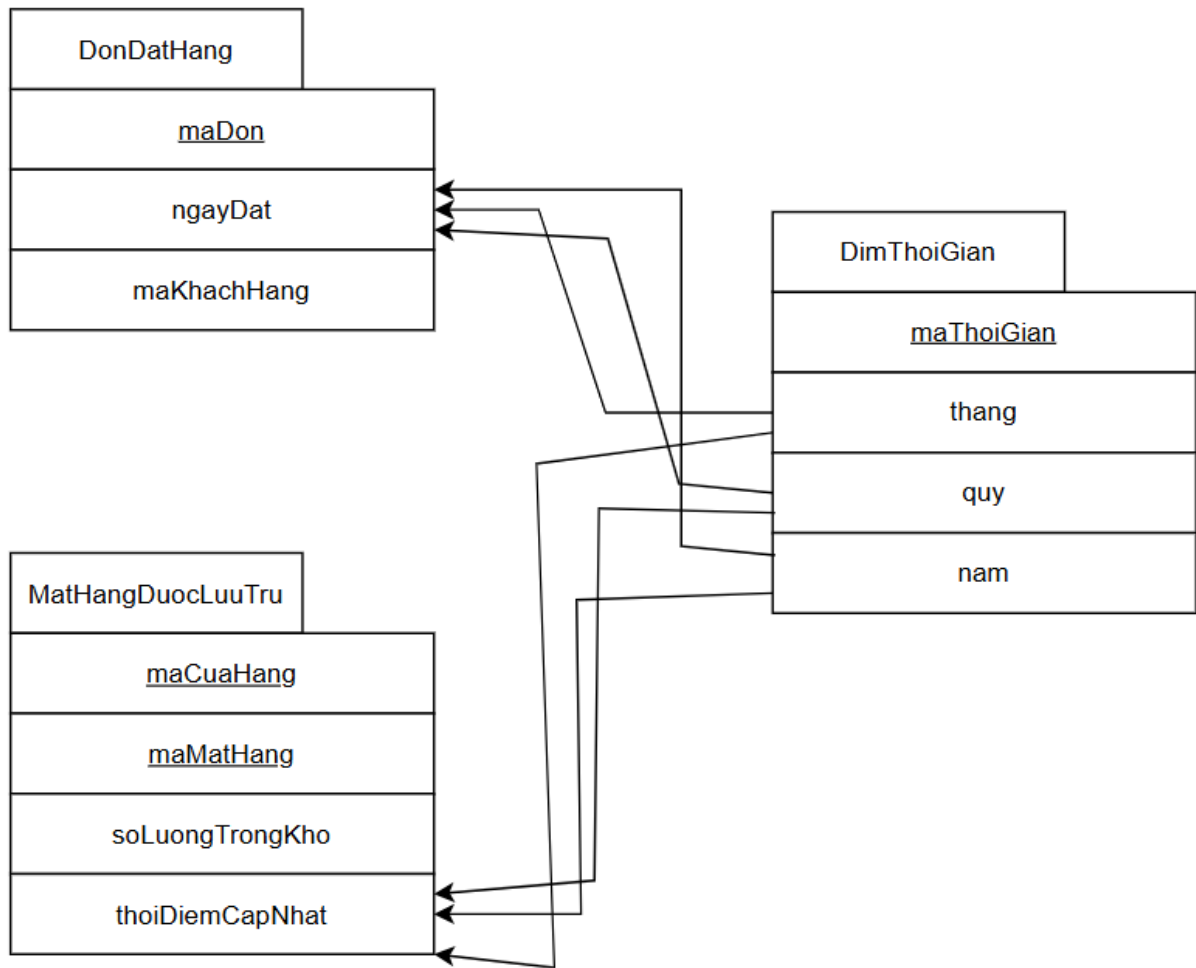
– DimCuaHang:



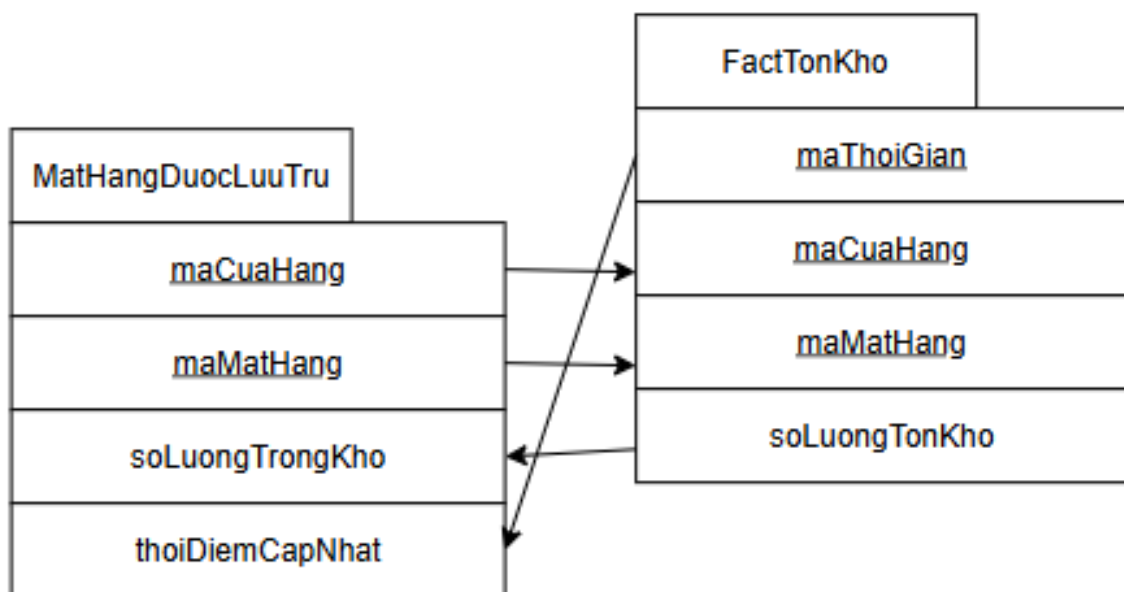
– DimVanPhongDaiDien:



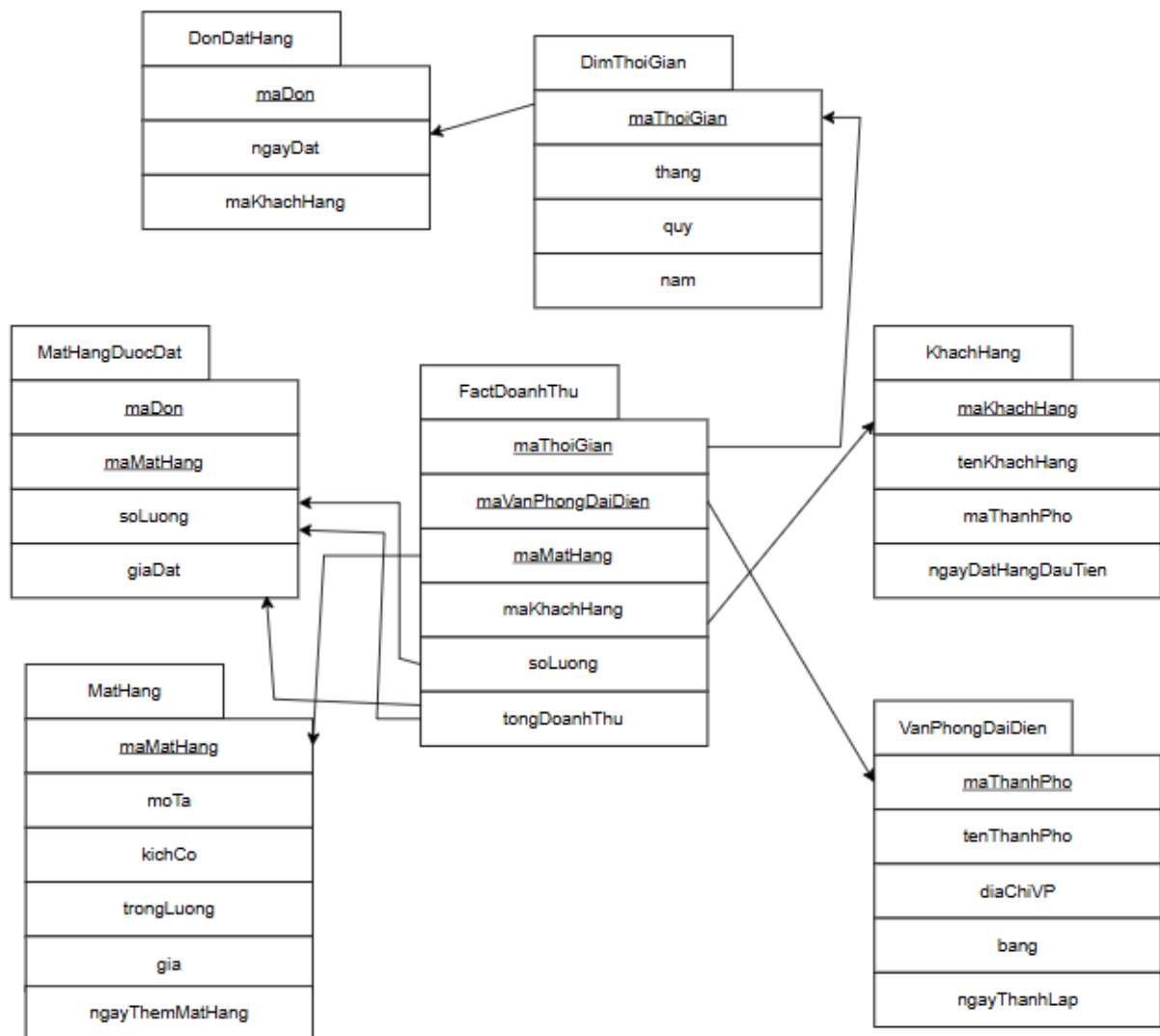
– DimThoiGian:



– FactTonKho:



- FactDoanhThu:



3.4.Đổ dữ liệu vào kho

- Sinh dữ liệu cho bảng DimThoiGian:

```

1  -- Thêm dữ liệu vào DimThoiGian cho 5 năm gần nhất
2  DECLARE @NamBatDau INT;
3  DECLARE @NamKetThuc INT;
4  DECLARE @NamHienTai INT;
5
6  SET @NamHienTai = YEAR(GETDATE());
7  SET @NamBatDau = @NamHienTai - 4; -- Tính từ 5 năm trước, bao gồm năm hiện tại
8  SET @NamKetThuc = @NamHienTai;
9
10 WHILE @NamBatDau <= @NamKetThuc
11 BEGIN
12     DECLARE @Thang INT = 1;
13     WHILE @Thang <= 12
14     BEGIN
15         DECLARE @Quy INT = (SELECT CASE
16                                WHEN @Thang BETWEEN 1 AND 3 THEN 1
17                                WHEN @Thang BETWEEN 4 AND 6 THEN 2
18                                WHEN @Thang BETWEEN 7 AND 9 THEN 3
19                                ELSE 4
20                                END);
21         -- Kiểm tra xem bản ghi đã tồn tại chưa trước khi chèn
22         IF NOT EXISTS (SELECT 1 FROM DimThoiGian WHERE thang = @Thang AND quy = @Quy AND nam = @NamBatDau)
23         BEGIN
24             INSERT INTO DimThoiGian (maThoiGian, thang, quy, nam)
25             VALUES ((SELECT ISNULL(MAX(maThoiGian), 0) + 1 FROM DimThoiGian), @Thang, @Quy, @NamBatDau);
26         END;
27         SET @Thang = @Thang + 1;
28     END;
29     SET @NamBatDau = @NamBatDau + 1;
30 END;
31 GO

```

- Tạo hàm GetMaThoiGian(thang, quy, nam): tự động lấy mã thời gian phù hợp

```

1  -- Hàm để lấy maThoiGian
2  CREATE FUNCTION dbo.GetMaThoiGian (@thang INT, @quy INT, @nam INT)
3  RETURNS INT
4  AS
5  BEGIN
6      DECLARE @maThoiGian INT;
7
8      SELECT @maThoiGian = maThoiGian
9      FROM DimThoiGian
10     WHERE thang = @thang AND quy = @quy AND nam = @nam;
11
12     RETURN ISNULL(@maThoiGian, 0);
13 END;
14 GO

```

- **Đổ dữ liệu cho các chiều còn lại:** DimKhachHang, DimCuaHang, DimMatHang, DimVanPhongDaiDien
- **Đổ dữ liệu vào các bảng fact:**
 - + FactDoanhThu:

```

1  INSERT INTO FactDoanhThu (maMatHang, maThoiGian, maKhachHang, maThanhPho, tongSoLuong, tongDoanhThu)
2  SELECT
3      MDD.maMatHang AS maMatHang,
4      dbo.GetMaThoiGian(MONTH(DDH.ngayDat), DATEPART(QUARTER, DDH.ngayDat), YEAR(DDH.ngayDat)) AS maThoiGian,
5      DDH.maKhachHang AS maKhachHang,
6      KH.maThanhPho AS maThanhPho,
7      SUM(MDD.soLuongDat) AS tongSoLuong,
8      SUM(MDD.giaDat * MDD.soLuongDat) AS tongDoanhThu
9  FROM TICHHOP_DB.dbo.MatHangDuocDat AS MDD
10 JOIN TICHHOP_DB.dbo.DonDatHang AS DDH ON MDD.maDon = DDH.maDon
11 JOIN TICHHOP_DB_1.dbo.KhachHang AS KH ON DDH.maKhachHang = KH.maKhachHang
12 GROUP BY MDD.maMatHang, DDH.maKhachHang, KH.maThanhPho, MONTH(DDH.ngayDat), DATEPART(QUARTER, DDH.ngayDat), YEAR(DDH.ngayDat);
13

```

- + FactTonKho:

```

1
2  -- Chuyển dữ liệu vào FactTonKho
3  INSERT INTO FactTonKho (maMatHang, maCuaHang, maThoiGian, soLuongTonKho)
4  SELECT
5      MDL.maMatHang,
6      MDL.maCuaHang,
7      dbo.GetMaThoiGian(MONTH(MDL.thoiDiemCapNhat), DATEPART(QUARTER, MDL.thoiDiemCapNhat), YEAR(MDL.thoiDiemCapNhat)) AS maThoiGian,
8      MDL.soLuongTrongKho
9  FROM TICHHOP_DB.dbo.MatHangDuocLuuTru AS MDL;

```

- **MatHangDuocLuuTru chỉ lưu giá trị tồn kho cuối cùng**
 - ➔ Fake giá trị tồn kho trong quá khứ -> đổ vào bảng FactTonKho

3.5. Thiết lập tự động cập nhật dữ liệu vào kho:

- **Bước 1:** Tạo procedure thêm dữ liệu (dbo.ProcessDataFromTICHHOP)
- **Bước 2:** kết nối SQL Server Agent
- **Bước 3:** Trong SQL Server Agent, tạo New Jobs
- **Bước 4:**
 - + Bước 3.1: Tại general, chọn name: Update_KDL_DB_1
 - + Bước 3.2: Tại Steps: chọn New...
 - + Điền Step name, type(T_SQL), Database, command (lệnh cập nhật)

e_ProcessDataFromTICHHOP

Script Help

Step name:
Execute_ProcessDataFromTICHHOP

Type:
Transact-SQL script (T-SQL)

Run as:

Database:
KDL_DB_1

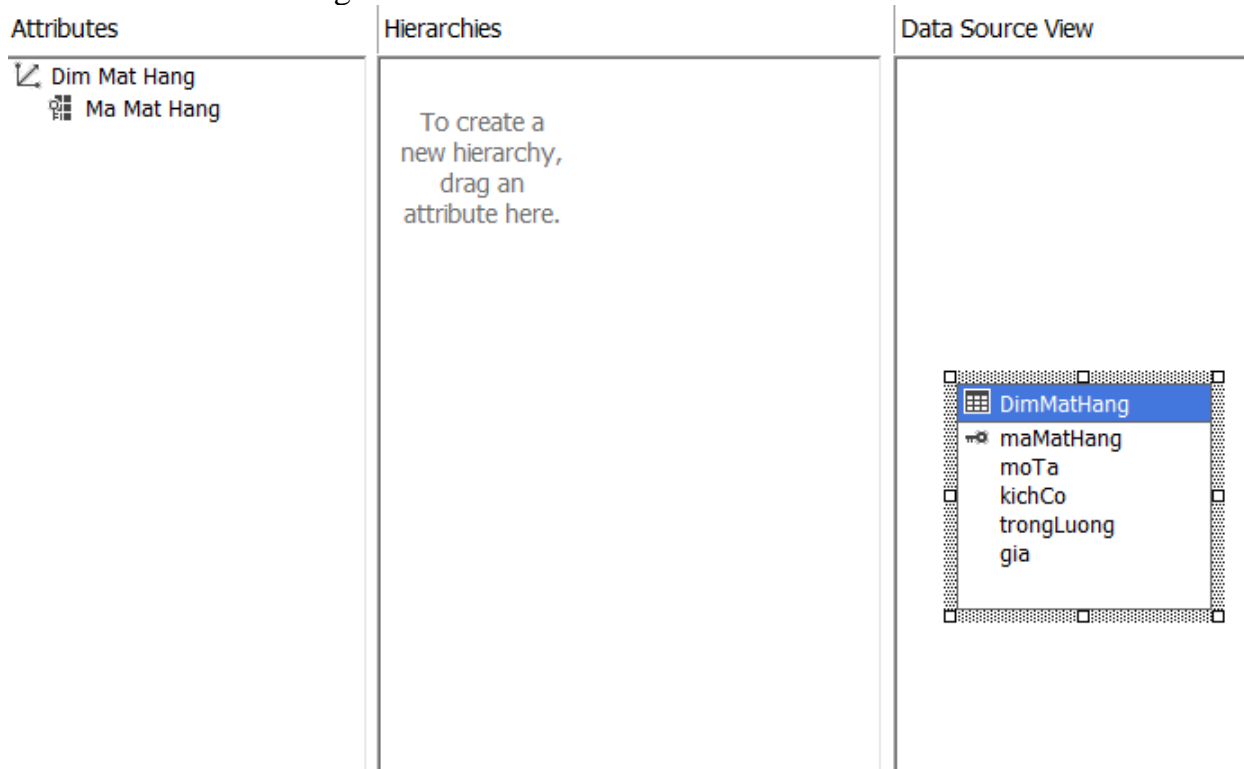
Command:
EXEC dbo.ProcessDataFromTICHHOP;

Open...
Select All
Copy
Paste
Parse

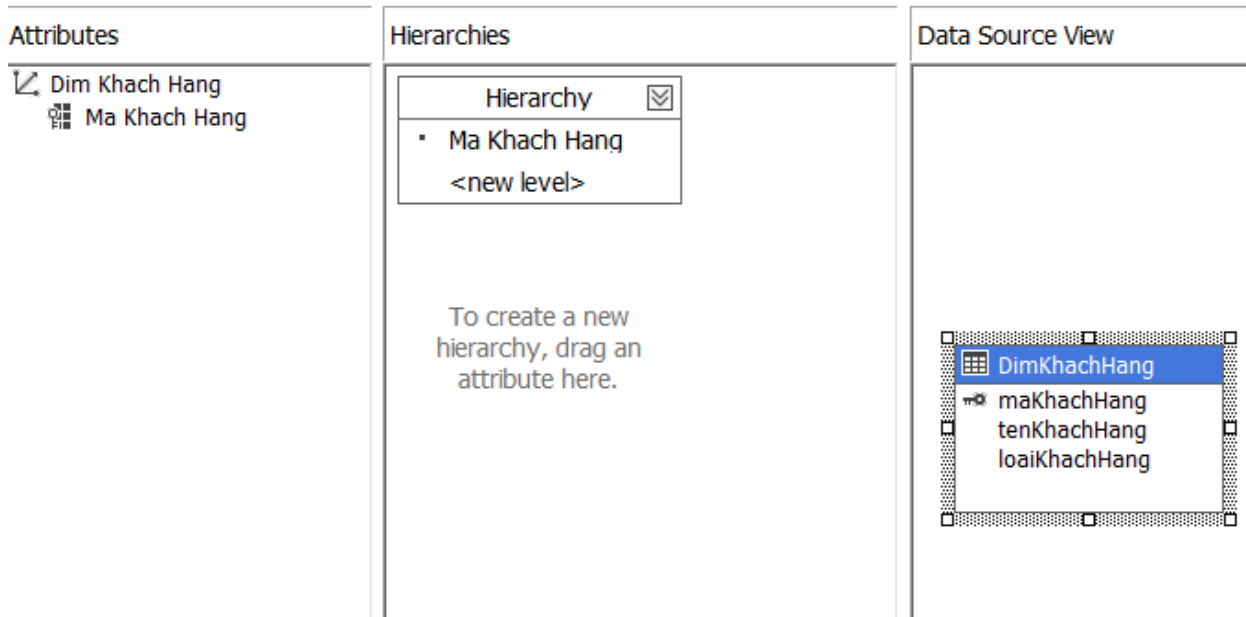
- + Trong schedules, chọn New.../daily, nhấn OK

4. Tạo các khối dữ liệu:

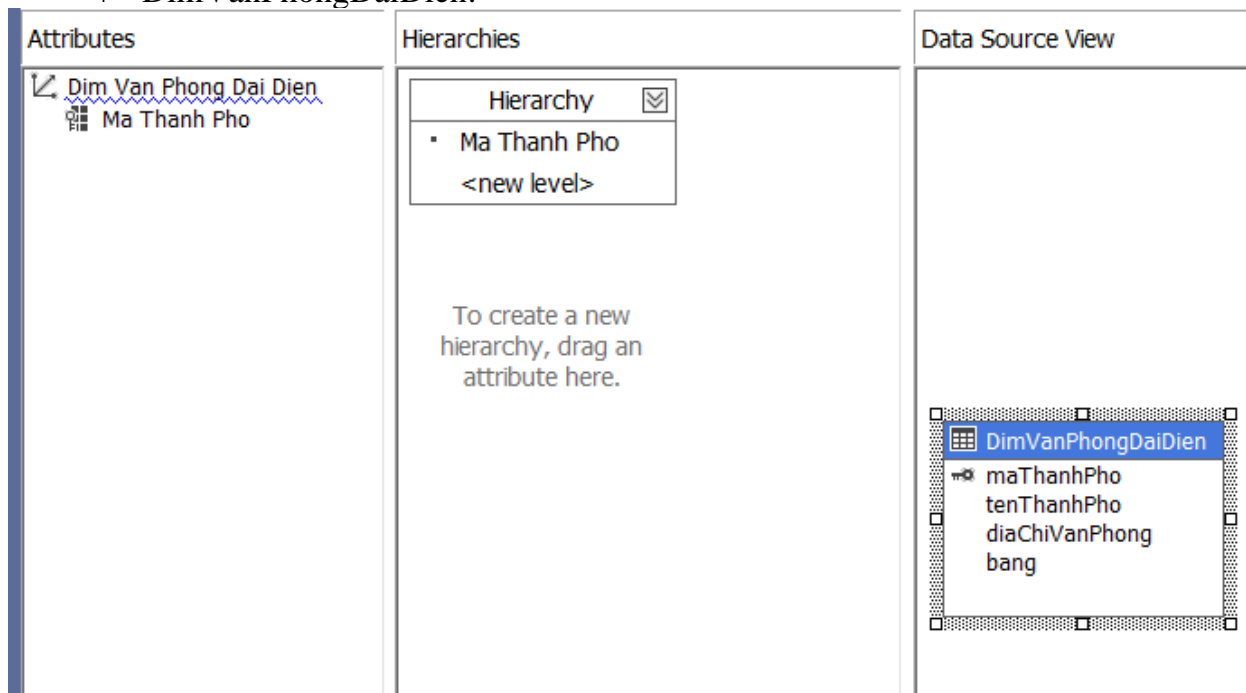
- Công cụ:
 - + Sử dụng SQL Server Analysis Services, tạo MOLAP
- Data source: KDL_DB_1
- Dimension:
 - + Dim Mat Hang.dim:



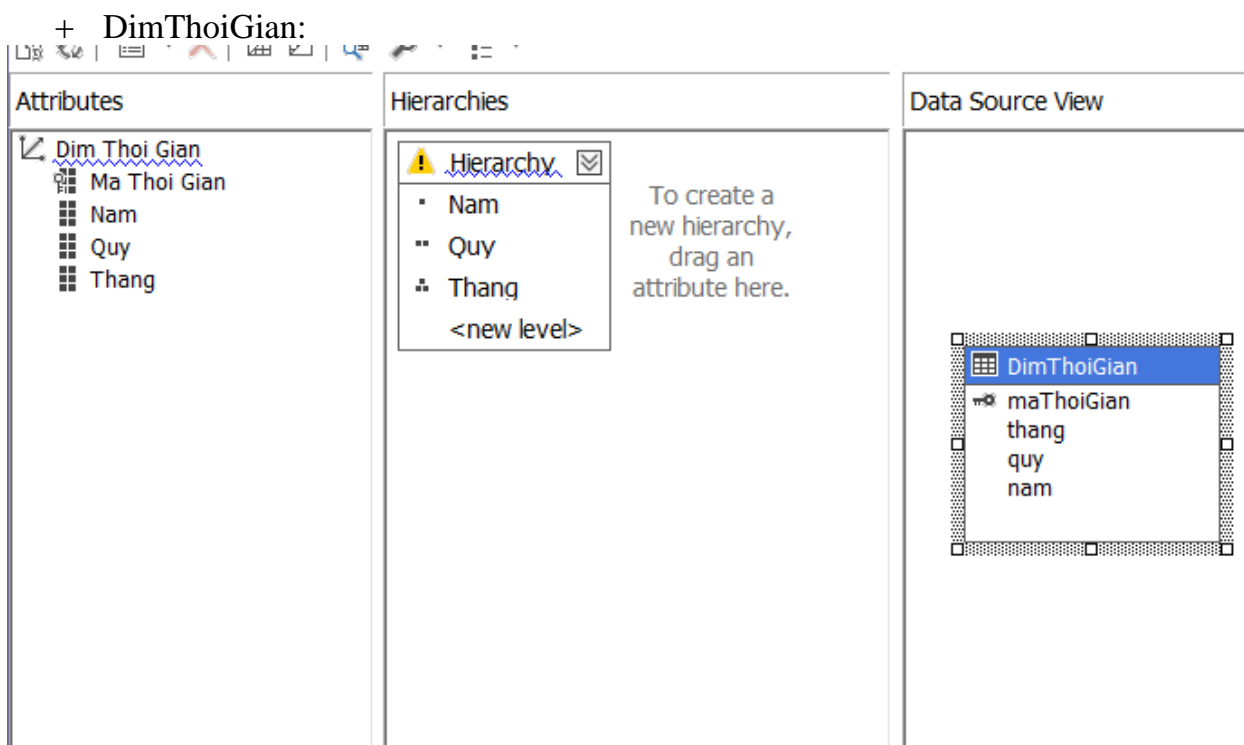
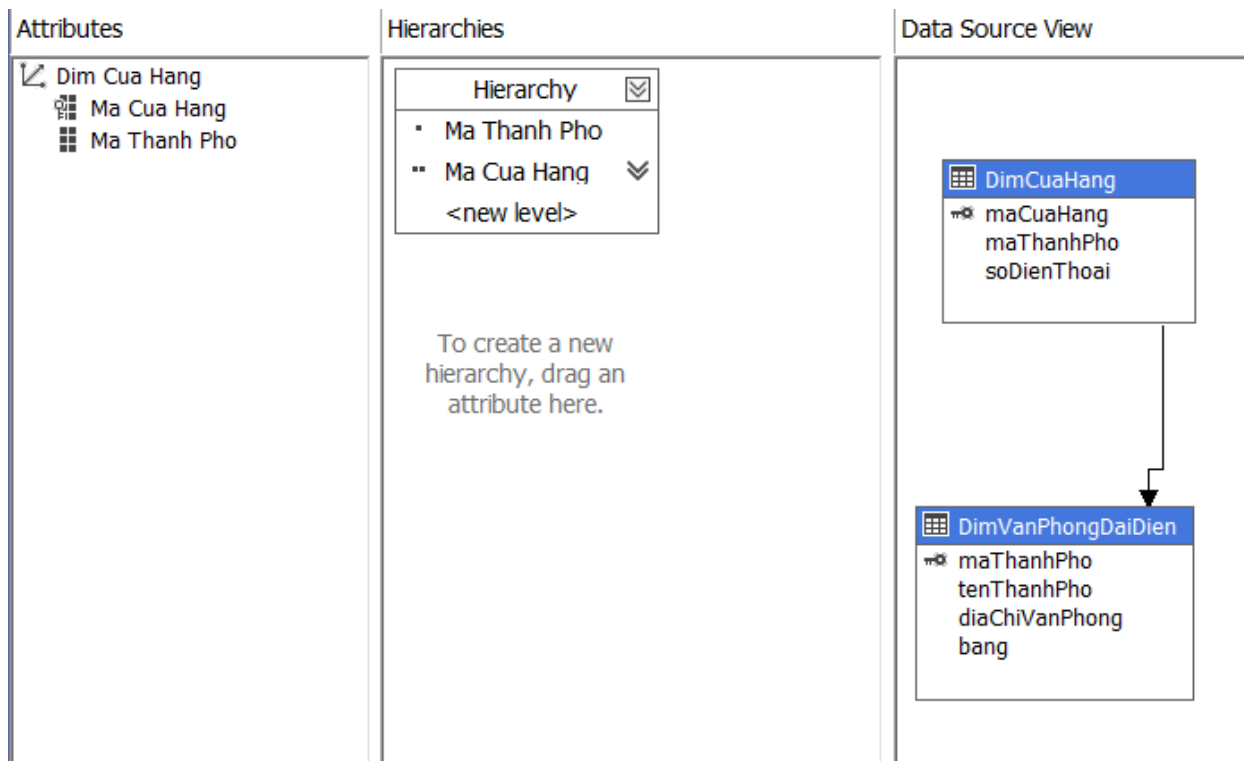
- + DimKhachHang:



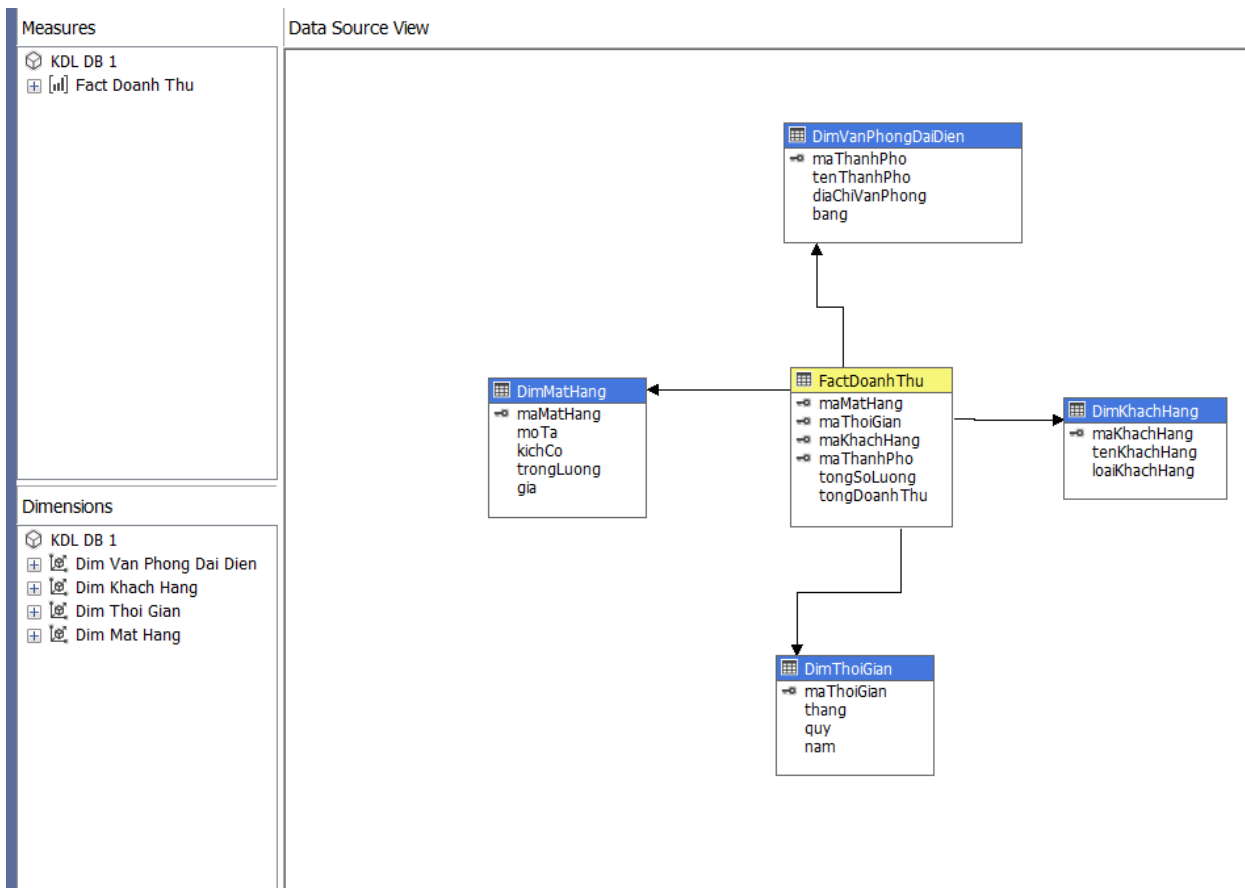
+ DimVanPhongDaiDien:



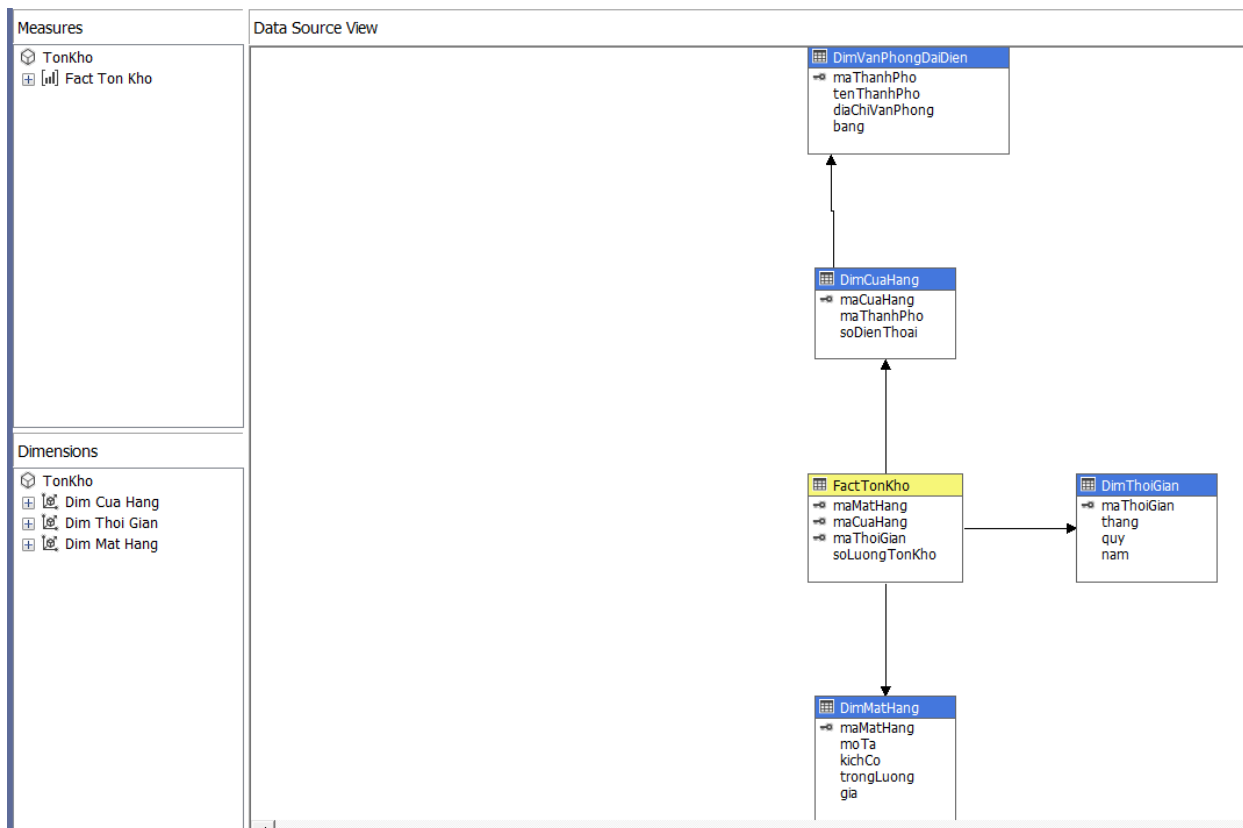
+ DimCuaHang:



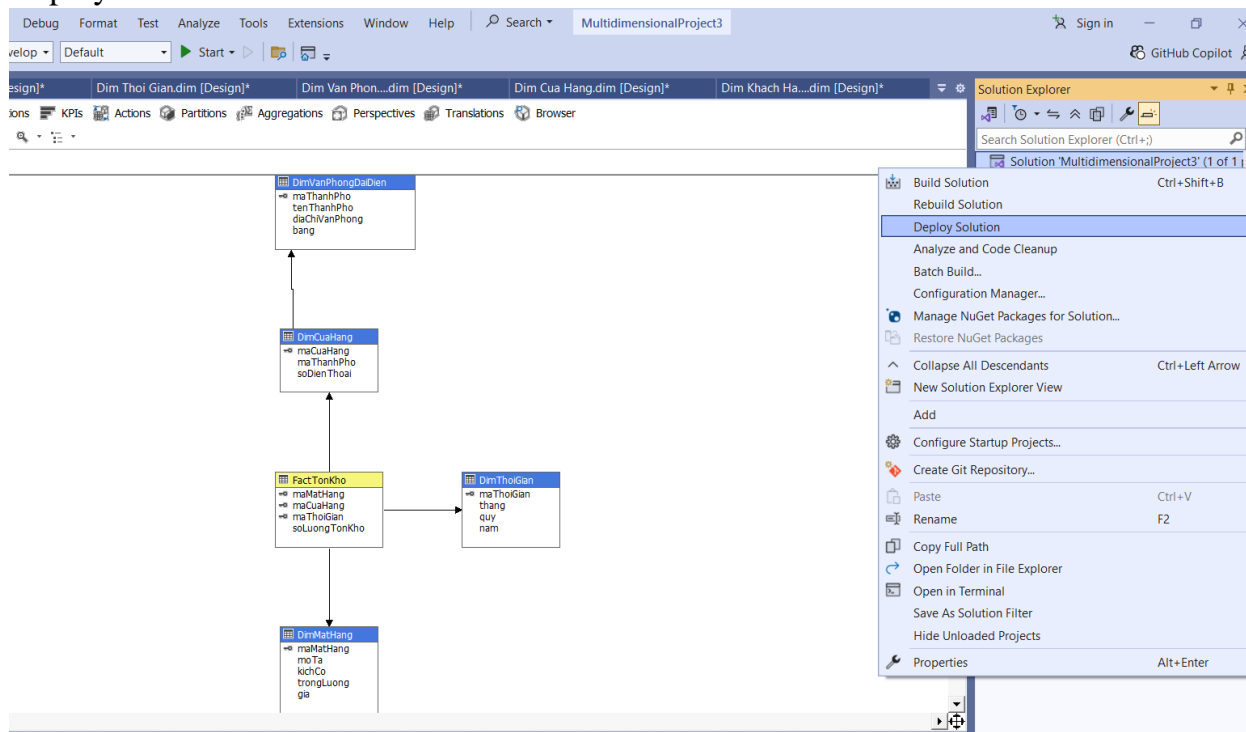
- Tạo các cube:
 - + Cube DoanhThu:



+ Cube TonKho:



– Deploy solution:

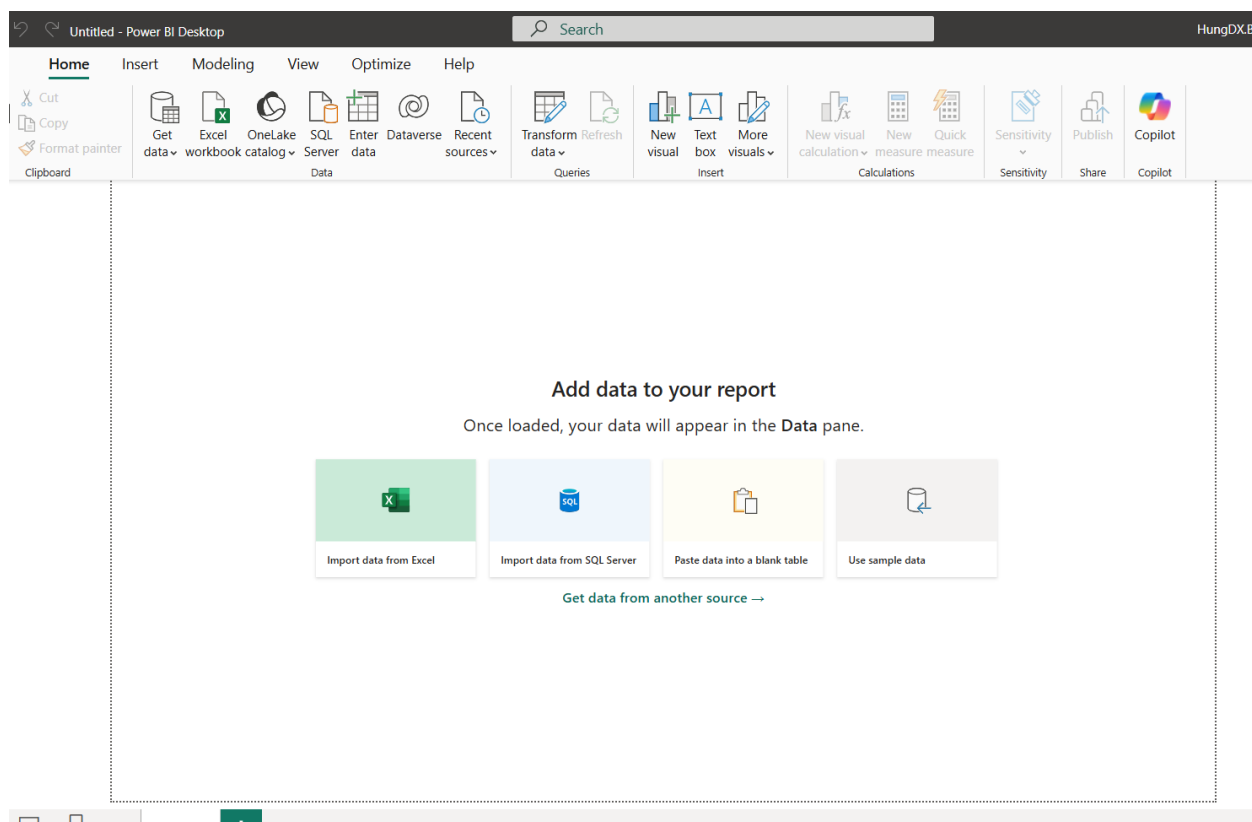


Chương VI: Báo cáo phân tích trực tuyến

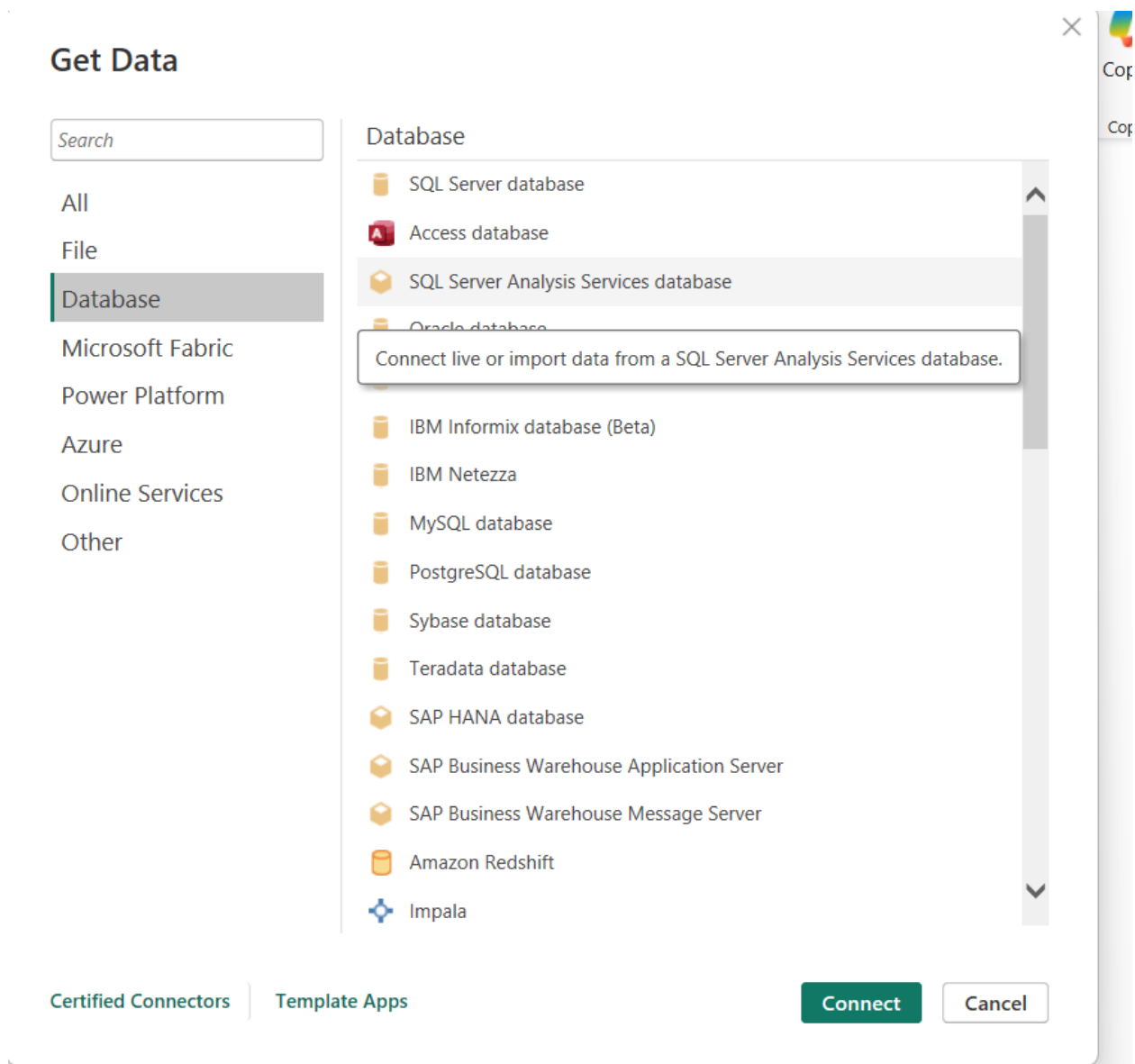
Sử dụng Power BI

1. Tạo kết nối từ Power BI tới Database:

Bước 1: Chọn Get Data:



Bước 2: Connect tới SQL Server Analysis Service Databases



Bước 3: Nhập tên SQL Server:

SQL Server Analysis Services database

Server ⓘ
LAPTOP-1V6NPRVI\BTL_KDL_1

Database (optional)

☐ Import

☒ Connect live

▸ MDX or DAX query (optional)

OK

Cancel

Bước 4: Chọn Cube

Navigator

🔍

📄

▴

LAPTOP-1V6NPRVI\BTL_KDL_1: Multidimensio...

📦 KDL DB 1

📦 KDL DB 12

📦 TonKho

KDL DB 1

Last Modified: 05/17/2025 16:31:18

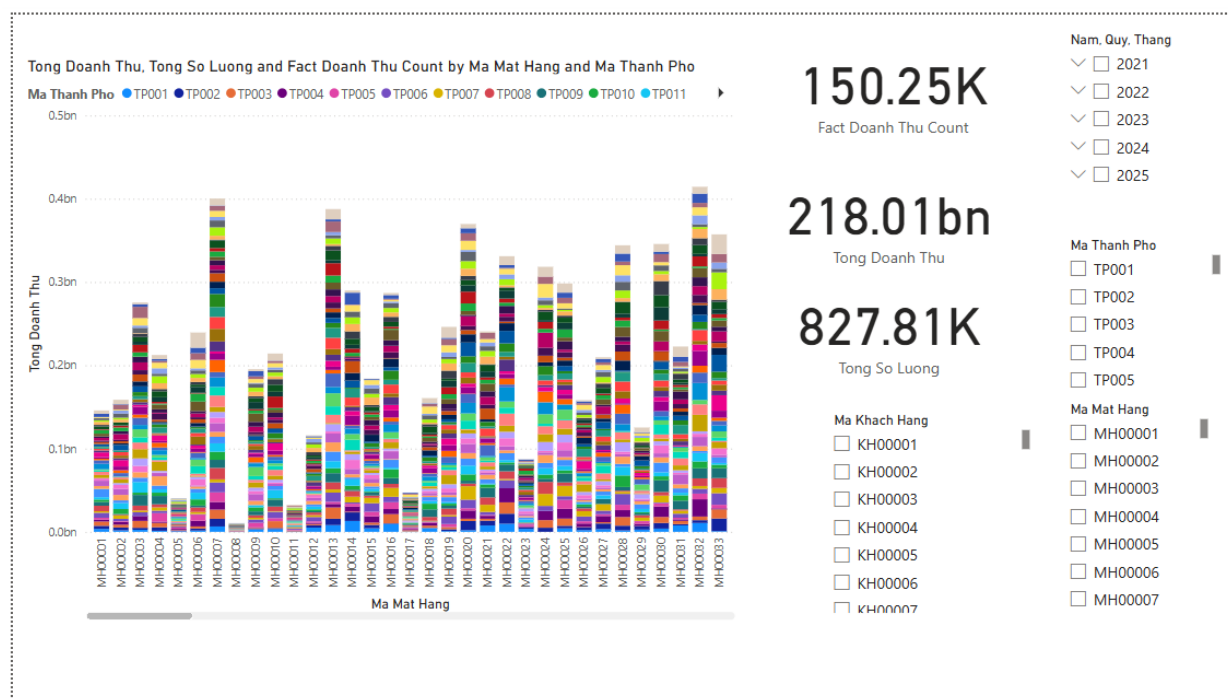
This model contains the following dimensions and measures
Dim Khách Hàng, Dim Mặt Hàng, Dim Thời Gian, Dim Văn Phòng Đại Diện, Tong So Luong, Tong Doanh Thu, Fact Doanh Thu Count

OK

Cancel

2. Tạo báo cáo:

2.1.Cube DoanhThu:



- Các card thể hiện giá trị của các độ đo:

150.25K
Fact Doanh Thu Count

218.01bn
Tong Doanh Thu

827.81K
Tong So Luong

- Slicer:

+ Thời gian

Nam, Quý, Tháng

✓ ☐ 2021

✓ ☐ 2022

✓ ☐ 2023

✓ ☐ 2024

✓ ☐ 2025

+ Khách hàng

Ma Khách Hàng

☐ KH00024

☐ KH00025

☐ KH00026

☐ KH00027

☐ KH00028

☐ KH00029

☐ KH00030

+ Thành phố

Ma Thanh Pho

☐ TP001

☐ TP002

☐ TP003

☐ TP004

☐ TP0   ...

+ Mặt hàng

Ma Mat Hang

☐ MH00001

☐ MH00002

☐ MH00003

☐ MH00004

☐ MH00005

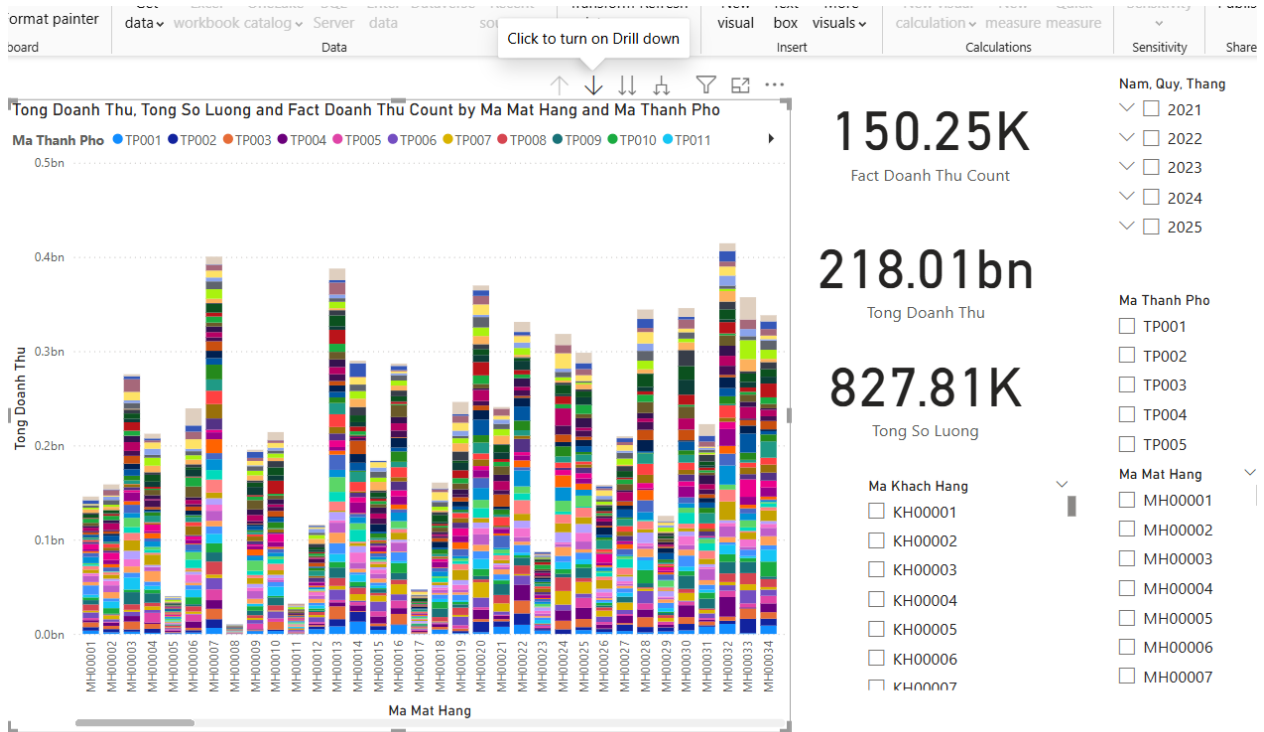
☐ MH00006

☐ MH00007

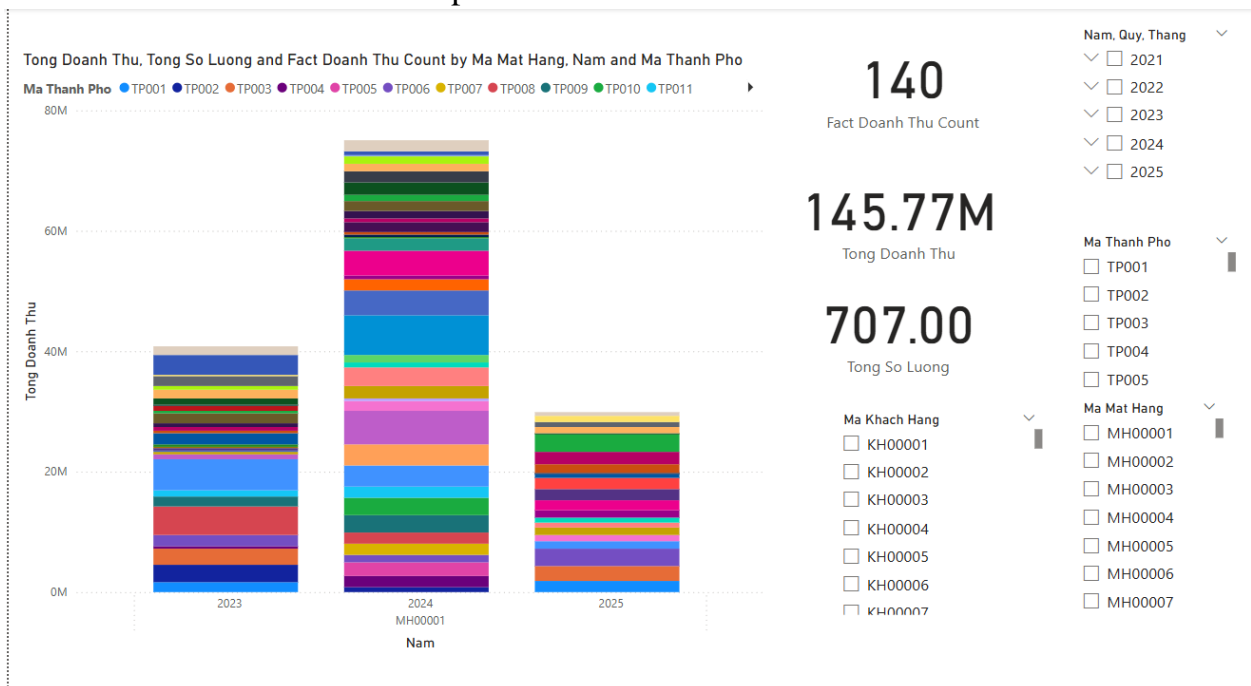
- Biểu đồ sử dụng: Stacked column chart (Biểu đồ cột chồng)
- Ý nghĩa các chiều trong báo cáo:
 - + Chiều Y: độ đo trong bảng fact, ví dụ ở đây em chọn Tong Doanh Thu
 - + Chiều X: Có thể chọn nhiều chiều và nó **phân cấp các chiều** dựa trên thứ tự của chúng (bên cạnh những chiều có phân cấp riêng của chính nó), ở đây em chọn theo thứ tự các chiều: Dim Mat Hang -> Dim Thoi Gian -> Dim Khach Hang
 - + Chiều Legend:
 - Biểu diễn độ đo của một chiều thành các cột được xếp chồng lên nhau
 - Ở đây em chọn Dim Thanh Pho, khi đó 1 cột chứa thông tin doanh thu của các thành phố và được xếp chồng lên nhau

Các phép toán OLAP:

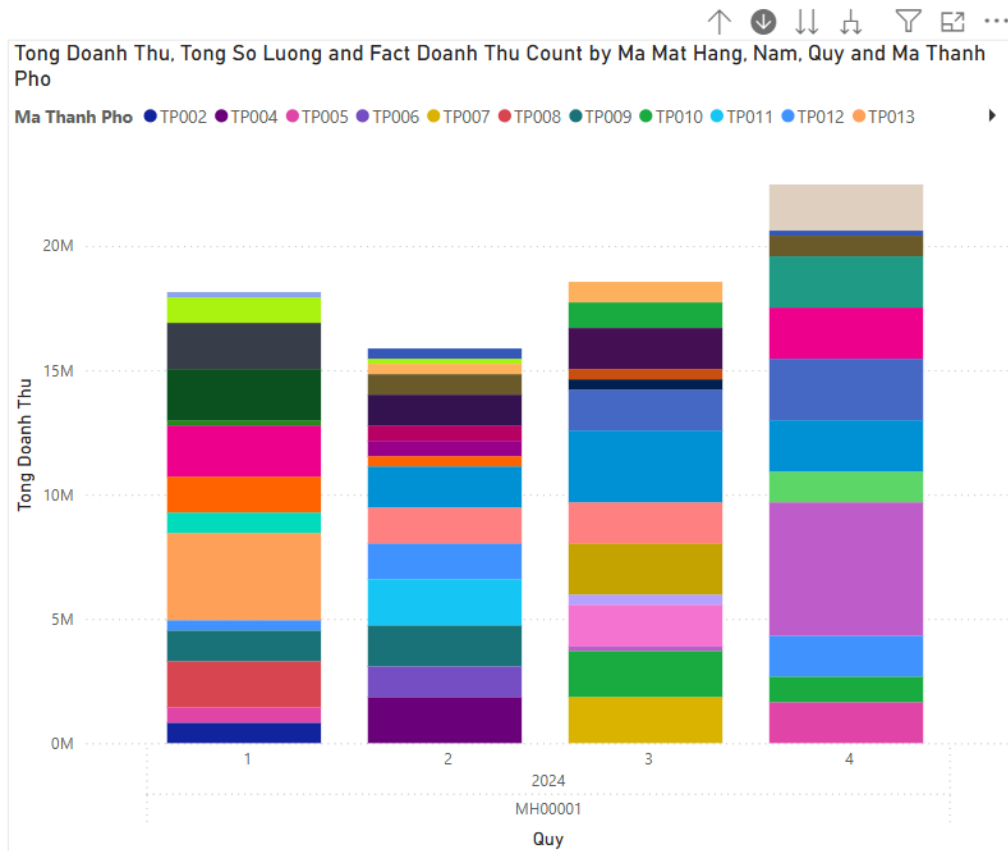
- Drill down/roll up:
 - + Click vào nút drill down:



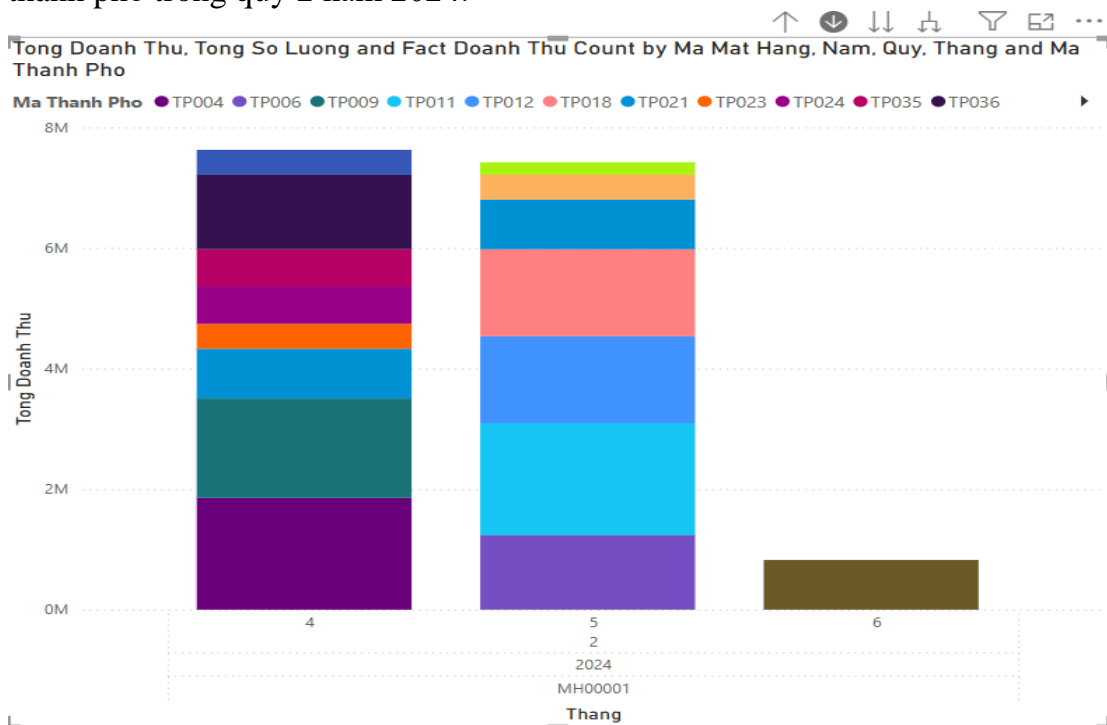
+ Chọn một cột MH0002 để drill down -> hiển thị doanh thu của mặt hàng MH0001 theo các năm của các thành phố:



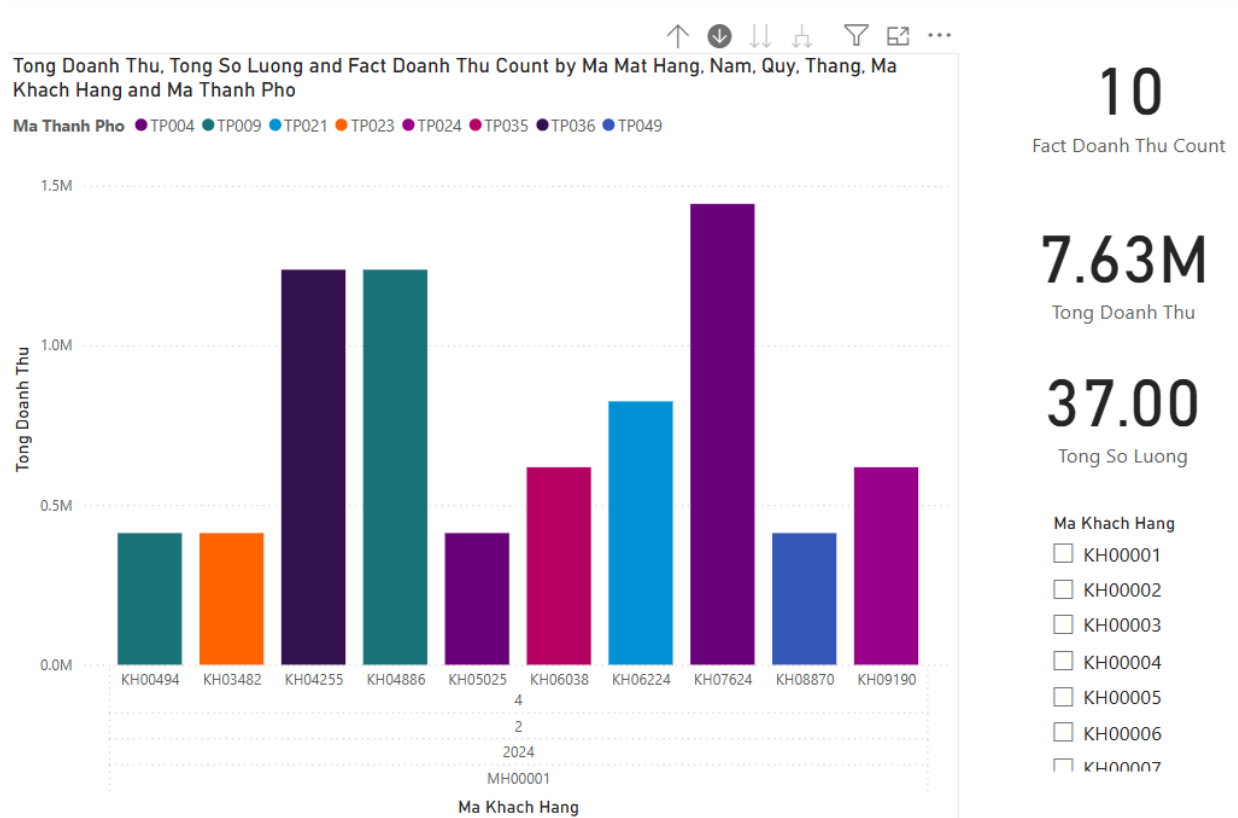
+ Drill down cột 2024 -> hiển thị tổng doanh thu của mặt hàng theo các quý của 1 thành phố trong năm 2024:



+ Drill down cột 2 -> hiển thị tổng doanh thu của mặt hàng theo các tháng của 1 thành phố trong quý 2 năm 2024:



- + Drill down cột tháng 4 -> hiển thị tổng doanh thu của mặt hàng theo các tháng của 1 thành phố trong tháng 4 năm 2024 theo các khách hàng



- + Roll up lên sẽ lần lượt tạo lại các bản đã được thực hiện
- Slice and dice:
 - + Phép Slice:
 - Chọn Ma Thanh Pho là TP001:

Ma Thanh Pho

☒ TP001

☐ TP002

☐ TP003

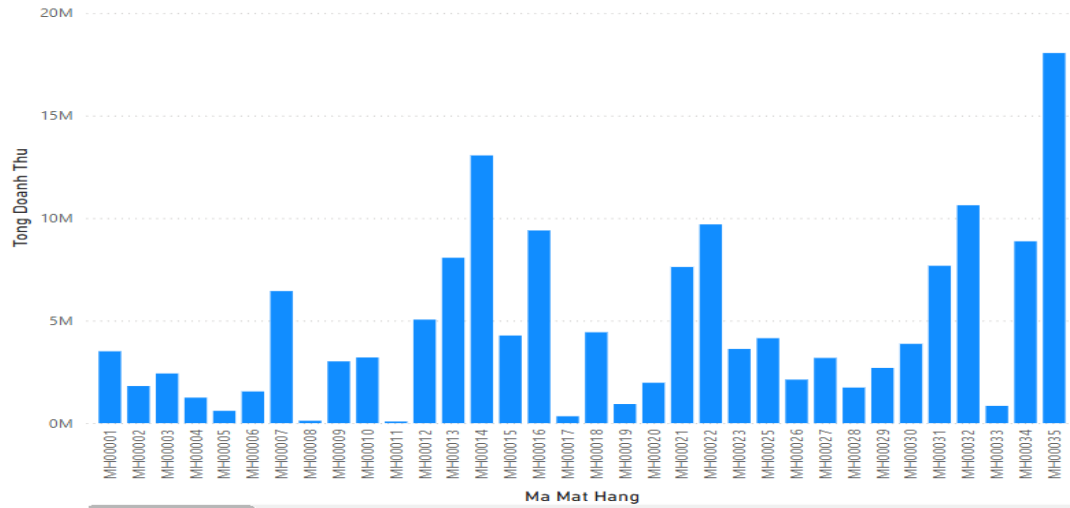
☐ TP004

☐ TP005

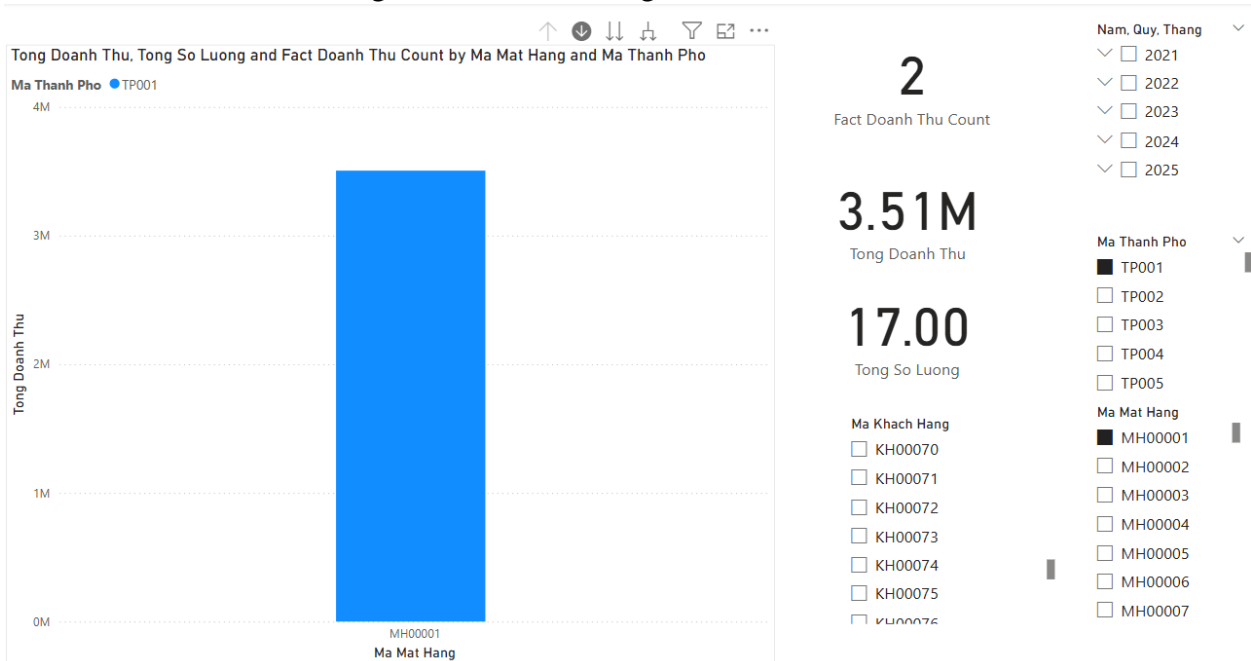
-> không còn chiều legend thể hiện doanh thu của từng thành phố nữa:

Tong Doanh Thu, Tong So Luong and Fact Doanh Thu Count by Ma Mat Hang and Ma Thanh Pho

Ma Thanh Pho ● TP001

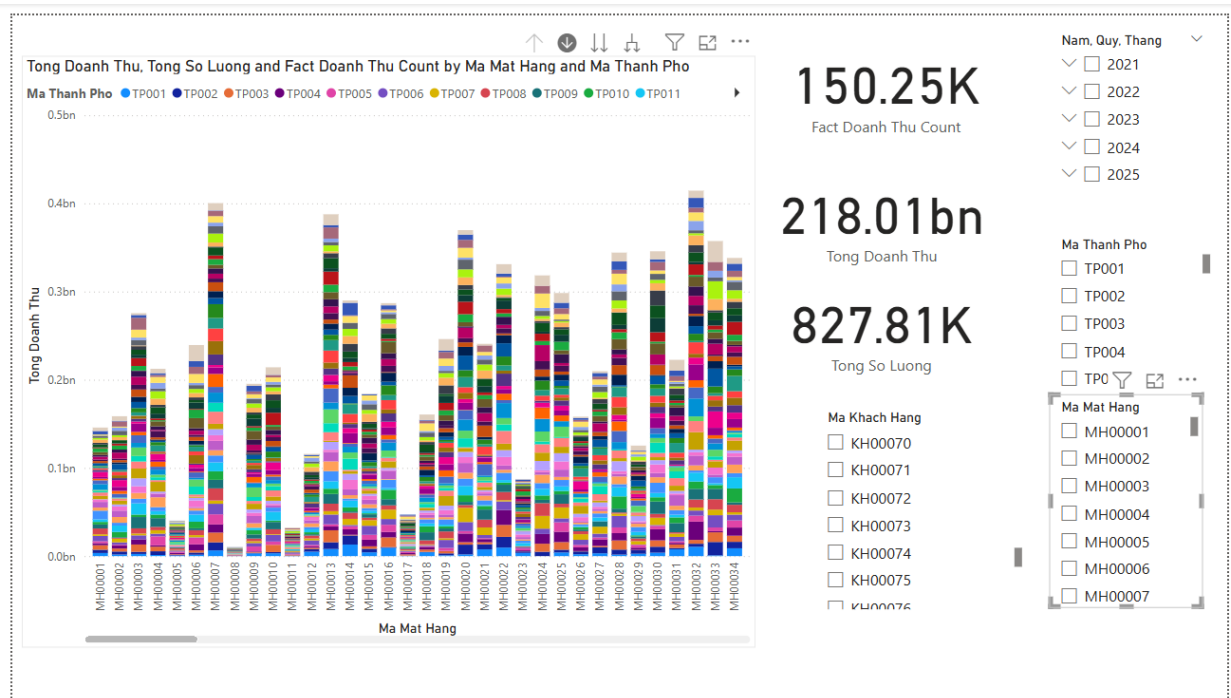


- Trong Slicer Ma Mat Hang, chọn MH00001 -> hệ thống chỉ hiển thị thông tin MH00001 -> không xét Dim Mat Hang nữa



+ Phép dice:

- Giao diện ban đầu:



- Trong bảng filters chọn Ma Mat Hang và thêm điều kiện:

Filters

Search

Filters on this visual

Fact Doanh Thu Count is (All)

Ma Khách Hàng is (All)

Ma Mat Hang starts with 'MH0000'

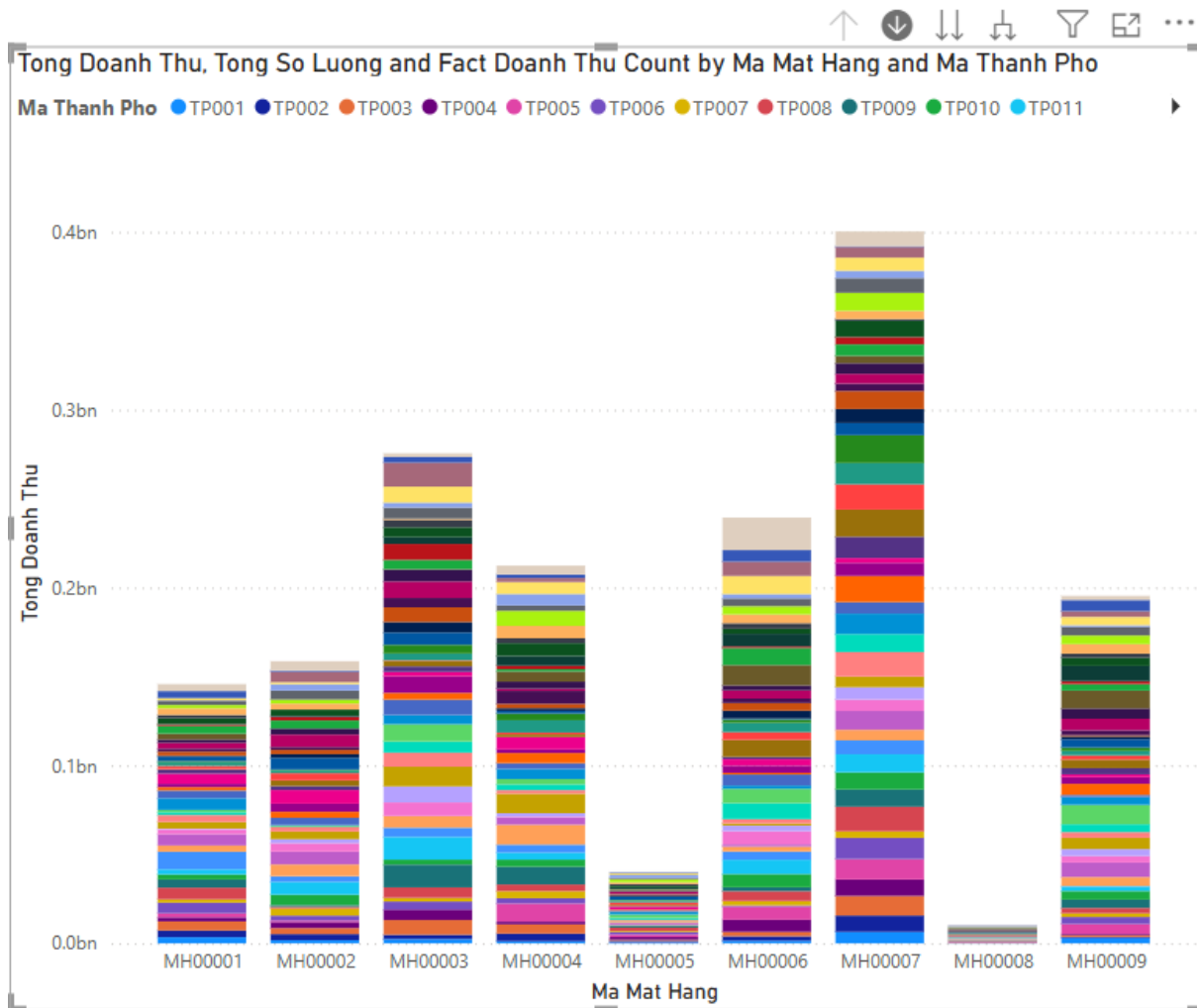
Filter type: Advanced filtering

Show items when the value starts with MH0000

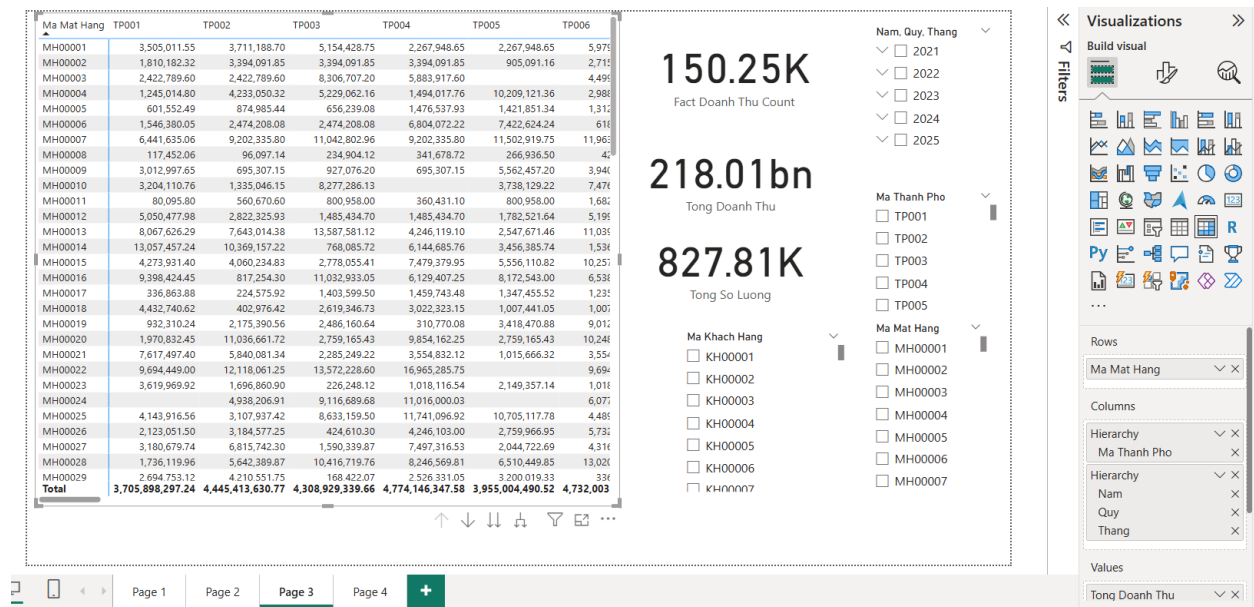
And Or

Apply filter

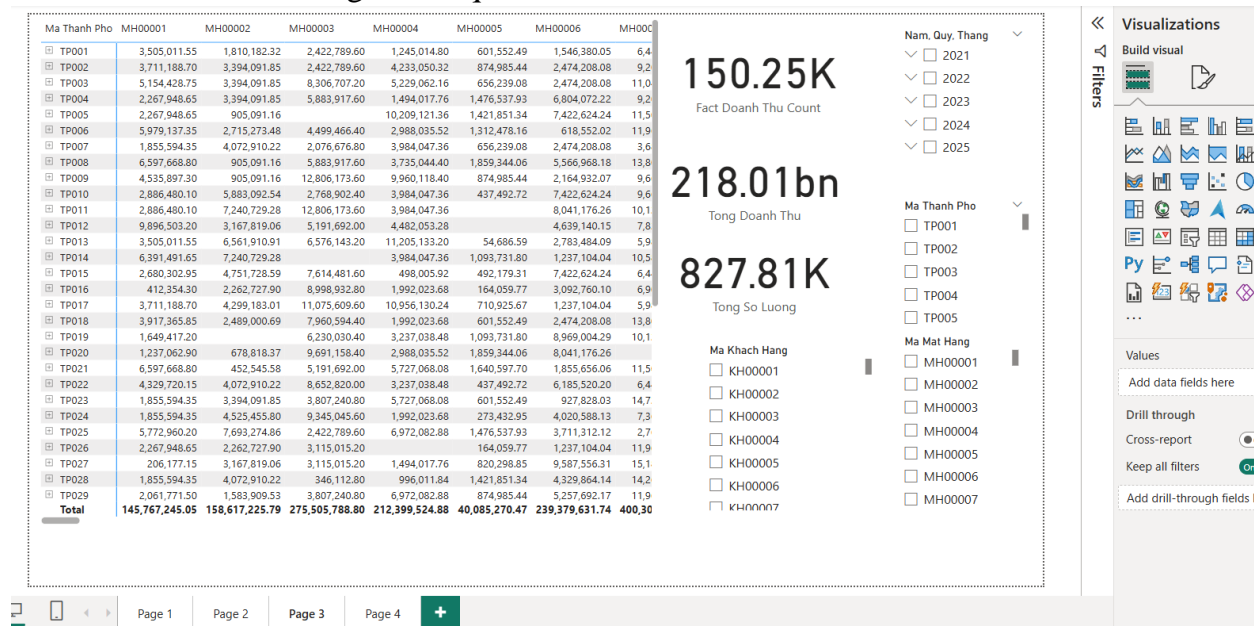
- Kết quả: số lượng các mặt hàng giảm đi nhưng vẫn đầy đủ các chiều dữ liệu:



- + Phép pivot: Sử dụng Matrix trong Power BI
 - Ban đầu tạo Matrix có row là Dim Mat Hang, column là Dim Thanh Pho và Dim Thoi Gian



- Ta xoay bằng cách đổi row là Dim Thanh Pho, Dim Thoi Gian và column là Dim Mat Hang -> kết quả:



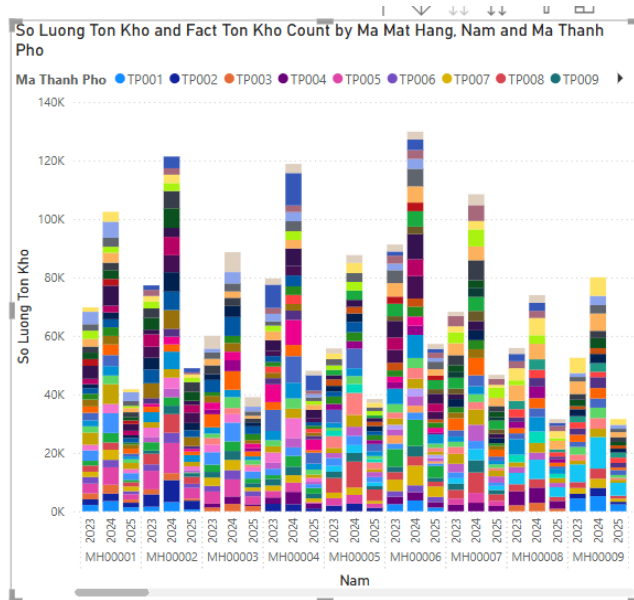
2.2.Cube TonKho:

812.25K

Fact Ton Kho Count

207.07M

So Luong Ton Kho



Ma Thanh Pho, Ma Cua Hang

☐ TP001

☐ TP002

☐ TP003

☐ TP004

☐ TP005

☐ TP006

☐ TP007

☐ TP008

☐ TP009

☐ TP010

☐ TP011

☐ TP012

☐ TP013

☐ TP014

☐ TP015

☐ TP016

☐ TP017

☐ TP018

☐ TP019

☐ TP020

☐ TP021

☐ TP022

☐ TP023

☐ TP024

☐ TP025

☐ TP026

☐ TP027

☐ TP028

☐ TP029

☐ TP030

☐ TP031

☐ TP032

☐ TP033

☐ TP034

☐ TP035

☐ TP036

☐ TP037

☐ TP038

☐ TP039

☐ TP040

☐ TP041

☐ TP042

☐ TP043

☐ TP044

☐ TP045

☐ TP046

☐ TP047

☐ TP048

☐ TP049

☐ TP050

☐ TP051

☐ TP052

☐ TP053

☐ TP054

☐ TP055

☐ TP056

☐ TP057

☐ TP058

☐ TP059

☐ TP060

☐ TP061

☐ TP062

☐ TP063

☐ TP064

☐ TP065

☐ TP066

☐ TP067

☐ TP068

☐ TP069

☐ TP070

☐ TP071

☐ TP072

- Các card thể hiện giá trị của các độ đo:

812.25K

Fact Ton Kho Count



207.07M

So Luong Ton Kho

- Slicer:

+ Thời gian

Nam, Quy, Thang

✓ ☐ 2021

✓ ☐ 2022

✓ ☐ 2023

✓ ☐ 2024

✓ ☐ 2025

+ Khu vực cửa hàng

Ma Thanh Pho, Ma Cua Hang

✓ ☐ TP001

✓ ☐ TP002

✓ ☐ TP003

✓ ☐ TP004

✓ ☐ TP005

✓ ☐ TP006

+ Mặt hàng

Ma Mat Hang

☐ MH00001

☐ MH00002

☐ MH00003

☐ MH00004

☐ MH00005

- Biểu đồ sử dụng: Stacked column chart (Biểu đồ cột chồng)
- Ý nghĩa các chiều trong báo cáo:
 - + Chiều Y: độ đo trong bảng fact, ở đây là So luong tồn kho
 - + Chiều X: Có thể chọn nhiều chiều và nó **phân cấp các chiều** dựa trên thứ tự của chúng (bên cạnh những chiều có phân cấp riêng của chính nó), ở đây em chọn theo thứ tự các chiều

Hierarchy	✓	✕
Ma Thanh Pho		✕
Ma Cua Hang		✕

Hierarchy	✓	✕
Nam		✕
Quy		✕
Thang		✕

+ Chiều Legend:

- Biểu diễn độ đo của một chiều thành các cột được xếp chồng lên nhau
- Ở đây em chọn Dim Mat Hang, khi đó 1 cột chứa thông tin số lượng tồn kho của các mặt hàng và được xếp chồng lên nhau

Các phép toán OLAP:

– Drill down/roll up:

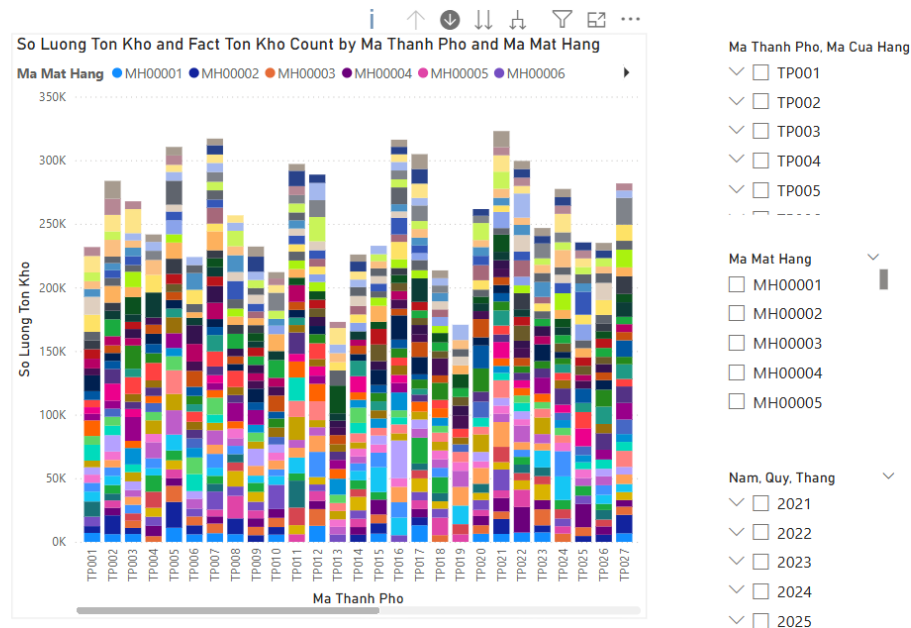
+ Ban đầu ta có báo cáo như sau.

812.25K

Fact Ton Kho Count

207.07M

So Luong Ton Kho



+ Chọn cột TP001 để drill down -> hiển thị số lượng tồn kho của các cửa hàng của Thành phố TP001

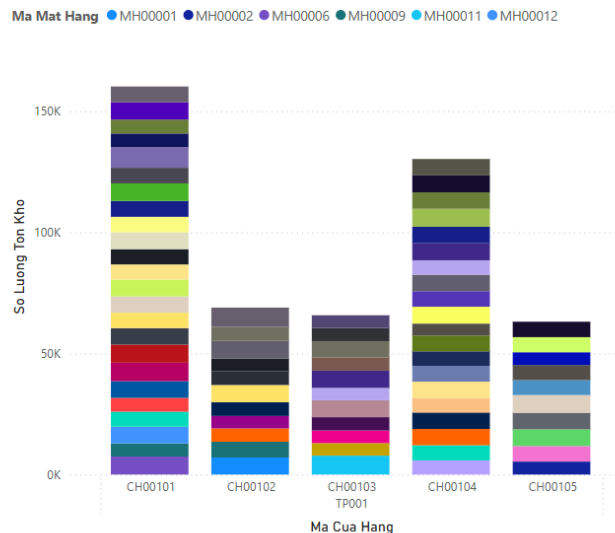
16.20K

Fact Ton Kho Count

4.16M

So Luong Ton Kho

So Luong Ton Kho and Fact Ton Kho Count by Ma Thanh Pho, Ma Cua Hang and Ma Mat Hang



Ma Thanh Pho, Ma Cua Hang

- ☐ TP001
- ☐ TP002
- ☐ TP003
- ☐ TP004
- ☐ TP005

Ma Mat Hang

- ☐ MH00001
- ☐ MH00002
- ☐ MH00003
- ☐ MH00004
- ☐ MH00005

Nam, Quy, Thang

- ☐ 2021
- ☐ 2022
- ☐ 2023
- ☐ 2024
- ☐ 2025

+ Chọn cửa hàng CH00101 để drill down -> kết quả:

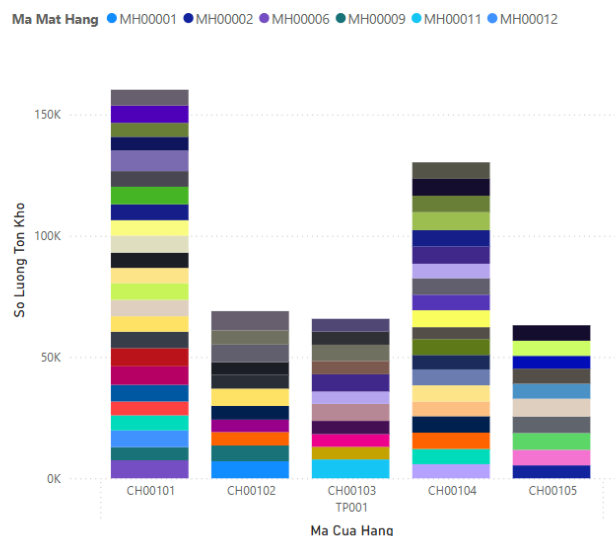
16.20K

Fact Ton Kho Count

4.16M

So Luong Ton Kho

So Luong Ton Kho and Fact Ton Kho Count by Ma Thanh Pho, Ma Cua Hang and Ma Mat Hang



Ma Thanh Pho, Ma Cua Hang

- ☐ TP001
- ☐ TP002
- ☐ TP003
- ☐ TP004
- ☐ TP005

Ma Mat Hang

- ☐ MH00001
- ☐ MH00002
- ☐ MH00003
- ☐ MH00004
- ☐ MH00005

Nam, Quy, Thang

- ☐ 2021
- ☐ 2022
- ☐ 2023
- ☐ 2024
- ☐ 2025

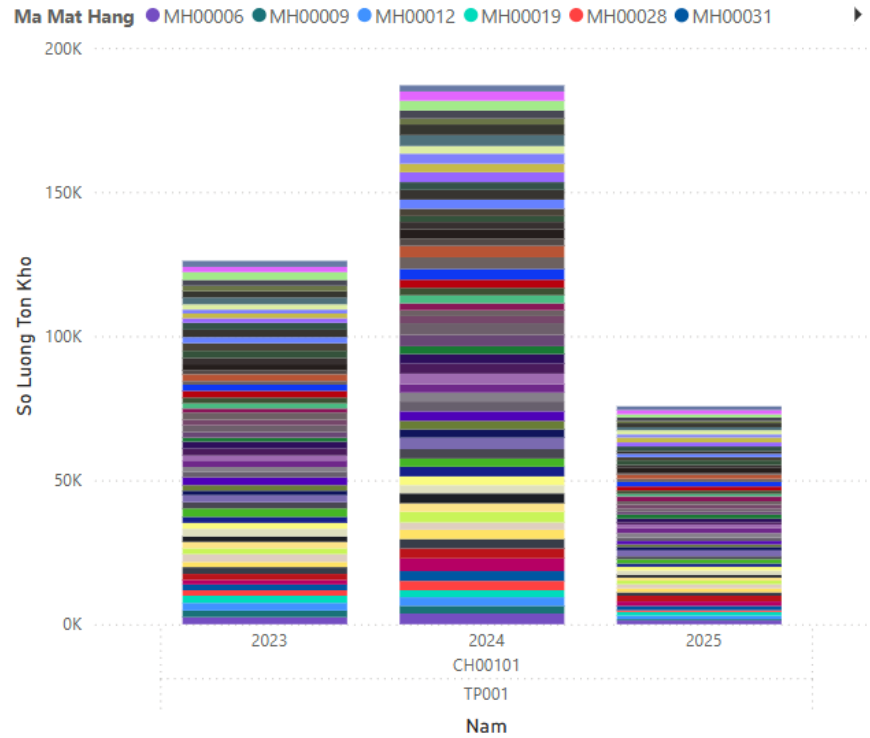
4125

Fact Ton Kho Count

1.07M

So Luong Ton Kho

So Luong Ton Kho and Fact Ton Kho Count by Ma Thanh Pho, Ma Cua Hang, Nam and Ma Mat Hang



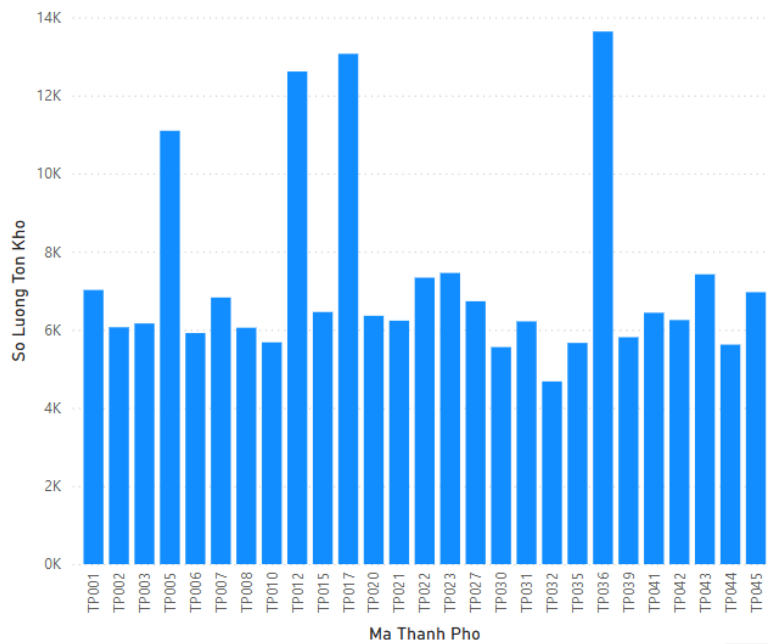
- + Sử dụng nút Drill up lên sẽ lần lượt tạo lại các bản đã được thực hiện
- Slice and dice:
 - + Phép Slice:
 - Chọn Ma Mat Hang là MH00001 -> không còn chiều Legend nữa -> giảm chiều dữ liệu:

850
Fact Ton Kho Count

213.97K
So Luong Ton Kho

So Luong Ton Kho and Fact Ton Kho Count by Ma Thanh Pho and Ma Mat Hang

Ma Mat Hang ● MH00001



Ma Thanh Pho, Ma Cua Hang

- ☒ TP001
- ☒ TP002
- ☒ TP003
- ☒ TP004
- ☒ TP005

Ma Mat Hang

- ☒ MH00001
- ☐ MH00002
- ☐ MH00003
- ☐ MH00004
- ☐ MH00005

Nam, Quy, Thang

- ☒ 2021
- ☒ 2022
- ☒ 2023
- ☒ 2024
- ☒ 2025

+ Phép dice:

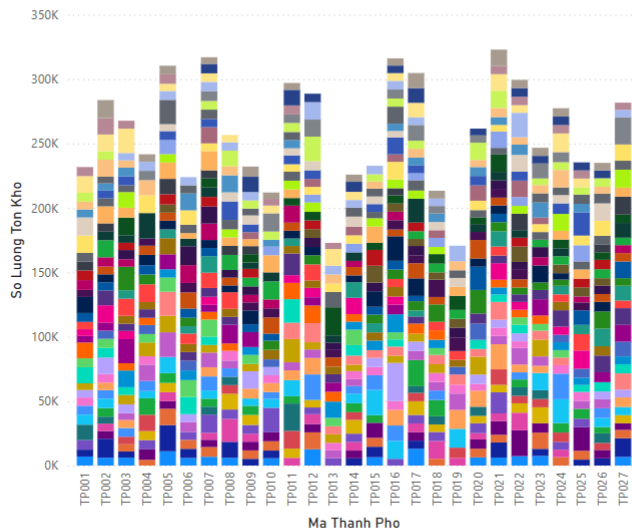
- Giao diện ban đầu:

812.25K
Fact Ton Kho Count

207.07M
So Luong Ton Kho

So Luong Ton Kho and Fact Ton Kho Count by Ma Thanh Pho and Ma Mat Hang

Ma Mat Hang ● MH00001 ● MH00002 ● MH00003 ● MH00004 ● MH00005 ● MH00006



Ma Thanh Pho, Ma Cua Hang

- ☒ TP001
- ☒ TP002
- ☒ TP003
- ☒ TP004
- ☒ TP005

Ma Mat Hang

- ☐ MH00001
- ☐ MH00002
- ☐ MH00003
- ☐ MH00004
- ☐ MH00005

Nam, Quy, Thang

- ☒ 2021
- ☒ 2022
- ☒ 2023
- ☒ 2024
- ☒ 2025

- Trong bảng filters chọn Ma Thang Pho và thêm điều kiện:

Ma Thanh Pho
contains 'TP00'

Filter type ⓘ

Advanced filtering ▼

Show items when the value

contains ▼

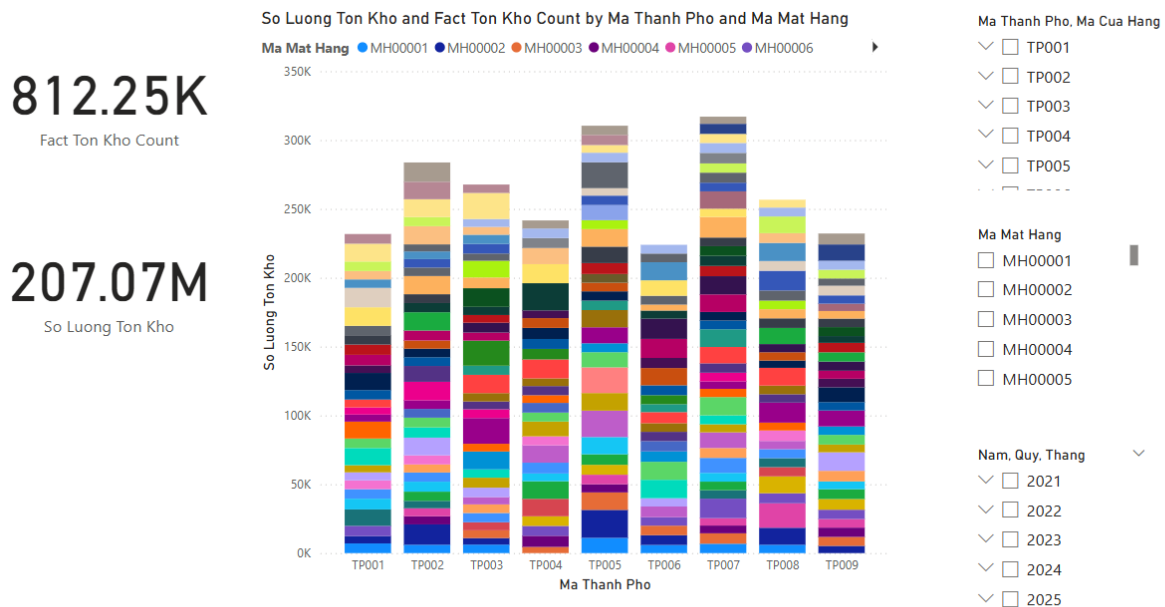
TP00

☒ And ☐ Or

▼

Apply filter

- Kết quả: số lượng các thành phố giảm đi nhưng vẫn đầy đủ các chiều dữ liệu:



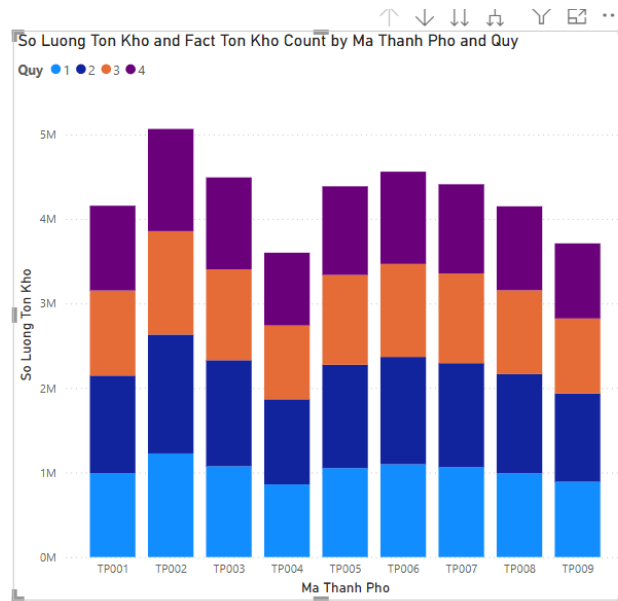
- + Phép pivot:
Ta có thể kéo các trường dữ liệu khác nhau vào các chiều x hoặc chiều legend để tạo ra báo cáo mới:

812.25K

Fact Ton Kho Count

207.07M

So Luong Ton Kho



Ma Thanh Pho, Ma Cua Hang

☐ TP001

☐ TP002

☐ TP003

☐ TP004

☐ TP005

☐ TP006

☐ TP007

☐ TP008

☐ TP009

☐ TP010

☐ TP011

☐ TP012

☐ TP013

☐ TP014

☐ TP015

☐ TP016

☐ TP017

☐ TP018

☐ TP019

☐ TP020

☐ TP021

☐ TP022

☐ TP023

☐ TP024

☐ TP025

☐ TP026

☐ TP027

☐ TP028

☐ TP029

☐ TP030

☐ TP031

☐ TP032

☐ TP033

☐ TP034

☐ TP035

☐ TP036

☐ TP037

☐ TP038

☐ TP039

☐ TP040

☐ TP041

☐ TP042

☐ TP043

☐ TP044

☐ TP045

☐ TP046

☐ TP047

☐ TP048

☐ TP049

☐ TP050

☐ TP051

☐ TP052

☐ TP053

☐ TP054

☐ TP055

☐ TP056

☐ TP057

☐ TP058

☐ TP059

☐ TP060

☐ TP061

☐ TP062

☐ TP063

☐ TP064

☐ TP065

☐ TP066

☐ TP067

☐ TP068

☐ TP069

☐ TP070

☐ TP071

☐ TP072

☐ TP073

☐ TP074

☐ TP075

☐ TP076

☐ TP077

☐ TP078

☐ TP079

☐ TP080

☐ TP081

☐ TP082

☐ TP083

☐ TP084

☐ TP085

☐ TP086

☐ TP087

☐ TP088

☐ TP089

☐ TP090

☐ TP091

☐ TP092

☐ TP093

☐ TP094

☐ TP095

☐ TP096

☐ TP097

☐ TP098

☐ TP099

☐ TP100

☐ TP101

☐ TP102

☐ TP103

☐ TP104

☐ TP105

☐ TP106

☐ TP107

☐ TP108

☐ TP109

☐ TP110

☐ TP111

☐ TP112

☐ TP113

☐ TP114

☐ TP115

☐ TP116

☐ TP117

☐ TP118

☐ TP119

☐ TP120

☐ TP121

☐ TP122

☐ TP123

☐ TP124

☐ TP125

☐ TP126

☐ TP127

☐ TP128

☐ TP129

☐ TP130

☐ TP131

☐ TP132

☐ TP133

☐ TP134

☐ TP135

☐ TP136

☐ TP137

☐ TP138

☐ TP139

☐ TP140

☐ TP141

☐ TP142

☐ TP143

☐ TP144

☐ TP145

☐ TP146

☐ TP147

☐ TP148

☐ TP149

☐ TP150

☐ TP151

☐ TP152

☐ TP153

☐ TP154

☐ TP155

☐ TP156

☐ TP157

☐ TP158

☐ TP159

☐ TP160

☐ TP161

☐ TP162

☐ TP163

☐ TP164

☐ TP165

☐ TP166

☐ TP167

☐ TP168

☐ TP169

☐ TP170

☐ TP171

☐ TP172

☐ TP173

☐ TP174

☐ TP175

☐ TP176

☐ TP177

☐ TP178

☐ TP179

☐ TP180

☐ TP181

☐ TP182

☐ TP183

☐ TP184

☐ TP185

☐ TP186

☐ TP187

☐ TP188

☐ TP189

☐ TP190

☐ TP191

☐ TP192

☐ TP193

☐ TP194

☐ TP195

☐ TP196

☐ TP197

☐ TP198

☐ TP199

☐ TP200

☐ TP201

☐ TP202

☐ TP203

☐ TP204

☐ TP205

☐ TP206

☐ TP207

☐ TP208

☐ TP209

☐ TP210

☐ TP211

☐ TP212

☐ TP213

☐ TP214

☐ TP215

☐ TP216

☐ TP217

☐ TP218

☐ TP219

☐ TP220

☐ TP221

☐ TP222

☐ TP223

☐ TP224

☐ TP225

☐ TP226

☐ TP227

☐ TP228

☐ TP229

☐ TP230

☐ TP231

☐ TP232

☐ TP233

☐ TP234

☐ TP235

☐ TP236

☐ TP237

☐ TP238

☐ TP239

☐ TP240

☐ TP241

☐ TP242

☐ TP243

☐ TP244

☐ TP245

☐ TP246

☐ TP247

☐ TP248

☐ TP249

☐ TP250

☐ TP251

☐ TP252

☐ TP253

☐ TP254

☐ TP255

☐ TP256

☐ TP257

☐ TP258

☐ TP259

☐ TP260

☐ TP261

☐ TP262

☐ TP263

☐ TP264

☐ TP265

☐ TP266

☐ TP267

<

Chương VII: Kiểm tra tính đúng đắn của dữ liệu

1. Tìm tất cả các cửa hàng cùng với thành phố, bang, số điện thoại, mô tả, kích cỡ, trọng lượng và đơn giá của tất cả các mặt hàng được bán ở kho đó.

```
1 SELECT
2     CH.maCuaHang,
3     VP.tenThanhPho,
4     VP.bang,
5     CH.soDienThoai,
6     MH.moTa,
7     MH.kichCo,
8     MH.trongLuong,
9     MH.gia
10 FROM
11     CuaHang AS CH
12 JOIN
13     VanPhongDaiDien AS VP ON CH.maThanhPho = VP.maThanhPho
14 JOIN
15     MatHangDuocLuuTru AS MDL ON CH.maCuaHang = MDL.maCuaHang
16 JOIN
17     MatHang AS MH ON MDL.maMatHang = MH.maMatHang;
```

10 %

Results Messages

	maCuaHang	tenThanhPho	bang	soDienThoai	moTa	kichCo	trongLuong	gia
1	CH00101	Thành phố JohnQuận	Hòa Bình	07 9388 6024	Sẽ repurpose front-end deliverables	L	2.03	309276.01
2	CH00101	Thành phố JohnQuận	Hòa Bình	07 9388 6024	Rất aggregate B2C channels	S	1.73	231769.05
3	CH00101	Thành phố JohnQuận	Hòa Bình	07 9388 6024	Không envisioneer virtual networks	L	5.48	148543.47
4	CH00101	Thành phố JohnQuận	Hòa Bình	07 9388 6024	Nhiều drive bricks-and-clicks interfaces	XXL	4.29	310770.08
5	CH00101	Thành phố JohnQuận	Hòa Bình	07 9388 6024	Của streamline transparent architectures	L	3.98	434029.99
6	CH00101	Thành phố JohnQuận	Hòa Bình	07 9388 6024	Gần revolutionize ubiquitous methodologies	L	0.24	307012.77
7	CH00101	Thành phố JohnQuận	Hòa Bình	07 9388 6024	Như repurpose cutting-edge paradigms	Free Size	1.75	462865.00
8	CH00101	Thành phố JohnQuận	Hòa Bình	07 9388 6024	Theo envisioneer end-to-end partnerships	XL	8.83	394304.53
9	CH00101	Thành phố JohnQuận	Hòa Bình	07 9388 6024	Vẫn facilitate clicks-and-mortar applications	XL	9.41	299862.41
10	CH00101	Thành phố JohnQuận	Hòa Bình	07 9388 6024	Để expedite interactive ROI	S	3.41	281596.48
11	CH00101	Thành phố JohnQuận	Hòa Bình	07 9388 6024	Tự optimize web-enabled e-services	S	0.99	223271.57
12	CH00101	Thành phố JohnQuận	Hòa Bình	07 9388 6024	Chứa matrix global methodologies	XXL	6.64	422769.69
13	CH00101	Thành phố JohnQuận	Hòa Bình	07 9388 6024	Cũng expedite cross-platform synergies	XXL	7.34	91472.90
14	CH00101	Thành phố JohnQuận	Hòa Bình	07 9388 6024	Tại deploy impactful channels	S	8.30	289972.15
15	CH00101	Thành phố JohnQuận	Hòa Bình	07 9388 6024	Có transform compelling channels	XXL	6.79	263075.21

2. Tìm tất cả các đơn đặt hàng với tên khách hàng và ngày đặt hàng được thực hiện bởi khách hàng đó.

```
1 SELECT
2     DDH.maDon,
3     KH.tenKhachHang,
4     DDH.ngayDat
5 FROM
6     DonDatHang AS DDH
7 JOIN
8     KhachHang AS KH ON DDH.maKhachHang = KH.maKhachHang;
```

) %			
Results Messages			
maDon	tenKhachHang	ngayDat	
DH000001	Phường Hoàng Mai	2025-01-25	
DH000002	Tú Xuân Lê	2023-09-09	
DH000003	Ánh Dương	2024-05-20	
DH000004	Bảo Xuân Phạm	2024-08-29	
DH000005	Lâm Mai Dương	2025-04-14	
DH000006	Anh Bùi	2025-04-13	
DH000007	Quý cô Phương Mai	2024-02-04	
DH000008	Khoa Đức Đặng	2025-04-16	
DH000009	Anh Tùng Trần	2024-03-16	
DH000010	Hạnh Mai Trần	2023-08-14	
DH000011	Hà Mai Bảo Lê	2023-10-19	
DH000012	Quý cô Mai Bùi	2023-08-15	
DH000013	Hà Dương	2024-05-08	
DH000014	Chị Dương Nguyễn	2023-10-31	
DH000015	Anh Trọng Vũ	2025-01-31	

3. Tìm tất cả các cửa hàng cùng với tên thành phố và số điện thoại mà có bán các mặt hàng được đặt bởi một khách hàng nào đó.

```

1 SELECT
2     DISTINCT CH.maCuaHang,
3     VP.tenThanhPho,
4     CH.soDienThoai
5 FROM
6     CuaHang AS CH
7 JOIN
8     VanPhongDaiDien AS VP ON CH.maThanhPho = VP.maThanhPho
9 JOIN
10    MatHangDuocLuuTru AS MDL ON CH.maCuaHang = MDL.maCuaHang
11 JOIN
12    MatHangDuocDat AS MDD ON MDL.maMatHang = MDD.maMatHang
13 JOIN
14    DonDatHang AS DDH ON MDD.maDon = DDH.maDon
15 WHERE
16    DDH.maKhachHang = 'KH00001';

```

10 %

Results Messages

	maCuaHang	tenThanhPho	soDienThoai
1	CH00701	Huyện JanePhường	03 9116825
2	CH00702	Huyện JanePhường	+84-98-807 5419
3	CH00703	Huyện JanePhường	+84-42-780601
4	CH00704	Huyện JanePhường	+84-64-801 7581
5	CH00705	Huyện JanePhường	01 7946666
6	CH04102	Huyện JanePhường	01 1486 5597
7	CH04104	Huyện JanePhường	+84-41-861877
8	CH04105	Huyện JanePhường	+84-77-952159
9	CH01201	JanePhường	+84-06-492 5159
10	CH01202	JanePhường	+84 28 2963625
11	CH01203	JanePhường	+84 70 3306622
12	CH01204	JanePhường	+84-66-082 8705
13	CH01205	JanePhường	+84-03-830064
14	CH01401	JanePhường	+84-83-184685
15	CH01402	JanePhường	+84 14 7058133

4. Tìm địa chỉ văn phòng đại diện với tên thành phố, bang của tất cả các cửa hàng lưu kho một mặt hàng nào đó với số lượng trên mức cụ thể.

```

1 SELECT
2     VP.diaChiVP,
3     VP.tenThanhPho,
4     VP.bang
5 FROM
6     VanPhongDaiDien AS VP
7 JOIN
8     CuaHang AS CH ON VP.maThanhPho = CH.maThanhPho
9 JOIN
10    MatHangDuocLuuTru AS MDL ON CH.maCuaHang = MDL.maCuaHang
11 WHERE
12    MDL.maMatHang = 'MH00001' AND MDL.soLuongTrongKho > 100;
13

```

Results			
Messages			
diaChiVP	tenThanhPho	bang	
9 Jane Số	Thành phố JohnQuận	Hòa Bình	
66 Vũ Dãy	Quận JohnThành phố	Đắk Lắk	
94 Jane Hẻm	JaneThị xã	Bắc Ninh	
18 John Số	Thị xã JohnQuận	Thừa Thiên Huế	
19 Jane Đường	Huyện JanePhường	Cần Thơ	
5 Bùi Ngõ	Thị xã JohnThành phố	An Giang	
352 Jane Số	Thành phố JohnXã	Hải Phòng	
75 Jane Số	JanePhường	Vĩnh Long	
75 Jane Số	JanePhường	Vĩnh Long	
4 Jane Ngõ	JohnQuận	Đồng Nai	
0 Trần Ngõ	Thành phố JohnQuận	Vĩnh Phúc	
547 Phạm Làng	JaneXã	Tuyên Quang	
16 John Hẻm	JohnXã	Hưng Yên	
2 Trần Ngõ	Thị xã JohnThị xã	Thừa Thiên Huế	
4 John Đường	JaneThành phố	Quảng Ngãi	

5. Với mỗi một đơn đặt hàng của khách, liệt kê các mặt hàng được đặt cùng với mô tả, mã cửa hàng, tên thành phố và các cửa hàng có bán mặt hàng đó.

```

1  SELECT
2      DDH.maDon,
3      MH.moTa,
4      CH.maCuaHang,
5      VP.tenThanhPho
6  FROM
7      DonDatHang AS DDH
8  JOIN
9      MatHangDuocDat AS MDD ON DDH.maDon = MDD.maDon
10 JOIN
11     MatHang AS MH ON MDD.maMatHang = MH.maMatHang
12 JOIN
13     MatHangDuocLuuTru AS MDL ON MH.maMatHang = MDL.maMatHang
14 JOIN
15     CuaHang AS CH ON MDL.maCuaHang = CH.maCuaHang
16 JOIN
17     VanPhongDaiDien AS VP ON CH.maThanhPho = VP.maThanhPho;

```

110 %

Results Messages

	maDon	moTa	maCuaHang	tenThanhPho
1	DH000001	Tại syndicate scalable experiences	CH01702	JohnQuận
2	DH000001	Tại syndicate scalable experiences	CH05003	JohnPhường
3	DH000001	Tại syndicate scalable experiences	CH04803	Thị xã JanePhường
4	DH000001	Tại syndicate scalable experiences	CH01403	JanePhường
5	DH000001	Tại syndicate scalable experiences	CH01202	JanePhường
6	DH000001	Tại syndicate scalable experiences	CH01005	Thành phố JohnXã
7	DH000001	Tại syndicate scalable experiences	CH01002	Thành phố JohnXã
8	DH000001	Tại syndicate scalable experiences	CH00903	JaneThành phố
9	DH000001	Tại syndicate scalable experiences	CH00601	Thị xã JohnQuận
10	DH000001	Tại syndicate scalable experiences	CH00105	Thành phố JohnQuận
11	DH000001	Tại syndicate scalable experiences	CH04604	Thị xã JohnThành phố
12	DH000001	Tại syndicate scalable experiences	CH04502	JohnThị xã
13	DH000001	Tại syndicate scalable experiences	CH04501	JohnThị xã
14	DH000001	Tại syndicate scalable experiences	CH03603	JohnHuyện
15	DH000001	Tại syndicate scalable experiences	CH03502	JohnHuyện

6. Tìm thành phố và bang mà một khách hàng nào đó sinh sống

```
1 SELECT
2     VP.tenThanhPho,
3     VP.bang
4 FROM
5     KhachHang AS KH
6 JOIN
7     VanPhongDaiDien AS VP ON KH.maThanhPho = VP.maThanhPho
8 WHERE
9     KH.maKhachHang = 'KH00001';
```

110 %

Results Messages

	tenThanhPho	bang
1	JohnThành phố	Lâm Đồng

7. Tìm mức độ tồn kho của một mặt hàng cụ thể tại tất cả các cửa hàng ở một thành phố cụ thể nào đó

```
1 SELECT
2     CH.maCuaHang,
3     MDL.soLuongTrongKho
4 FROM
5     MatHangDuocLuuTru AS MDL
6 JOIN
7     CuaHang AS CH ON MDL.maCuaHang = CH.maCuaHang
8 JOIN
9     VanPhongDaiDien AS VP ON CH.maThanhPho = VP.maThanhPho
10 WHERE
11     MDL.maMatHang = 'MH00001' AND VP.maThanhPho = 'TP001'; -- Thay 'maMatHang' và 'tenThanhPho' bằng giá trị cụ thể
12
```

%

Results Messages

maCuaHang	soLuongTrongKho
CH00102	321

8. Tìm các khách hàng du lịch, khách hàng đặt theo đường bưu điện và khách hàng thuộc cả hai loại:

```

1 SELECT
2     MH.moTa,
3     MDD.soLuongDat,
4     KH.tenKhachHang,
5     CH.maCuaHang,
6     VP.tenThanhPho
7 FROM
8     DonDatHang AS DDH
9 JOIN
10    MatHangDuocDat AS MDD ON DDH.maDon = MDD.maDon
11 JOIN
12    MatHang AS MH ON MDD.maMatHang = MH.maMatHang
13 JOIN
14    KhachHang AS KH ON DDH.maKhachHang = KH.maKhachHang
15 JOIN
16    CuaHang AS CH ON CH.maThanhPho = KH.maThanhPho
17 JOIN
18    VanPhongDaiDien AS VP ON CH.maThanhPho = VP.maThanhPho
19 WHERE
20    DDH.maDon = 'DH000001';

```

moTa	soLuongDat	tenKhachHang	maCuaHang	tenThanhPho
Tại syndicate scalable experiences	2	Phường Hoàng Mai	CH01901	JohnThành phố
Tự benchmark plug-and-play applications	10	Phường Hoàng Mai	CH01901	JohnThành phố
Từ orchestrate seamless e-markets	10	Phường Hoàng Mai	CH01901	JohnThành phố
Mỗi re-intermediate viral e-tailers	8	Phường Hoàng Mai	CH01901	JohnThành phố
Tại syndicate scalable experiences	2	Phường Hoàng Mai	CH01902	JohnThành phố
Tự benchmark plug-and-play applications	10	Phường Hoàng Mai	CH01902	JohnThành phố
Từ orchestrate seamless e-markets	10	Phường Hoàng Mai	CH01902	JohnThành phố
Mỗi re-intermediate viral e-tailers	8	Phường Hoàng Mai	CH01902	JohnThành phố
Tại syndicate scalable experiences	2	Phường Hoàng Mai	CH01903	JohnThành phố
Tự benchmark plug-and-play applications	10	Phường Hoàng Mai	CH01903	JohnThành phố
Từ orchestrate seamless e-markets	10	Phường Hoàng Mai	CH01903	JohnThành phố
Mỗi re-intermediate viral e-tailers	8	Phường Hoàng Mai	CH01903	JohnThành phố
Tại svndicate scalable experiences	2	Phường Hoàng Mai	CH01904	JohnThành phố

9. Tìm các khách hàng du lịch, khách hàng đặt theo đường bưu điện và khách hàng thuộc cả hai loại

```

1  -- Khách hàng du lịch
2  SELECT KH.* FROM KháchHang KH
3  JOIN KháchHangDuLich KHDL ON KH.maKhachHang = KHDL.maKhachHang;
4
5  -- Khách hàng đặt theo đường bưu điện
6  SELECT KH.* FROM KháchHang KH
7  JOIN KháchHangBuuDien KHBD ON KH.maKhachHang = KHBD.maKhachHang;
8
9  -- Khách hàng thuộc cả hai loại
10 SELECT KH.* FROM KháchHang KH
11 JOIN KháchHangDuLich KHDL ON KH.maKhachHang = KHDL.maKhachHang
12 JOIN KháchHangBuuDien KHBD ON KH.maKhachHang = KHBD.maKhachHang;

```

110 %

Results Messages

	maKhachHang	maThanhPho	tenKhachHang	ngayDatHangDauTien
1	KH00007	TP001	Cô Vi Bùi	2025-03-07
2	KH00020	TP033	Ông Anh Nguyễn	2025-04-18
3	KH00021	TP028	Huy Hải Hoàng	2024-08-25
4	KH00022	TP010	Chị Duyên Trần	2023-05-17
5	KH00027	TP013	Phương Mai Bảo Phạm	2024-01-29
6	KH00040	TP037	Quý ông Thành Mai	2024-10-26
7	KH00043	TP035	Xuân Nguyễn	2023-05-23
8	KH00052	TP011	Hưng Hữu Lê	2024-05-21
9	KH00054	TP034	Quý cô Hằng Lê	2023-05-17
	maKhachHang	maThanhPho	tenKhachHang	ngayDatHangDauTien
1	KH00001	TP019	Tùng Trí Bùi	2024-08-07
2	KH00003	TP046	Quý cô Nhật Nguyễn	2024-10-27
3	KH00004	TP015	Quý cô Yến Mai	2023-11-12
4	KH00005	TP032	Quý cô Vân Nguyễn	2023-12-11
5	KH00008	TP004	Linh Đặng	2023-06-18
6	KH00011	TP037	Châu Tấn Dương	2023-09-27
7	KH00014	TP030	Linh Hoàng	2024-08-18
8	KH00016	TP027	Ông Thành Hoàng	2023-06-02
	maKhachHang	maThanhPho	tenKhachHang	ngayDatHangDauTien

10. Tính tổng doanh thu của 1 người:

```

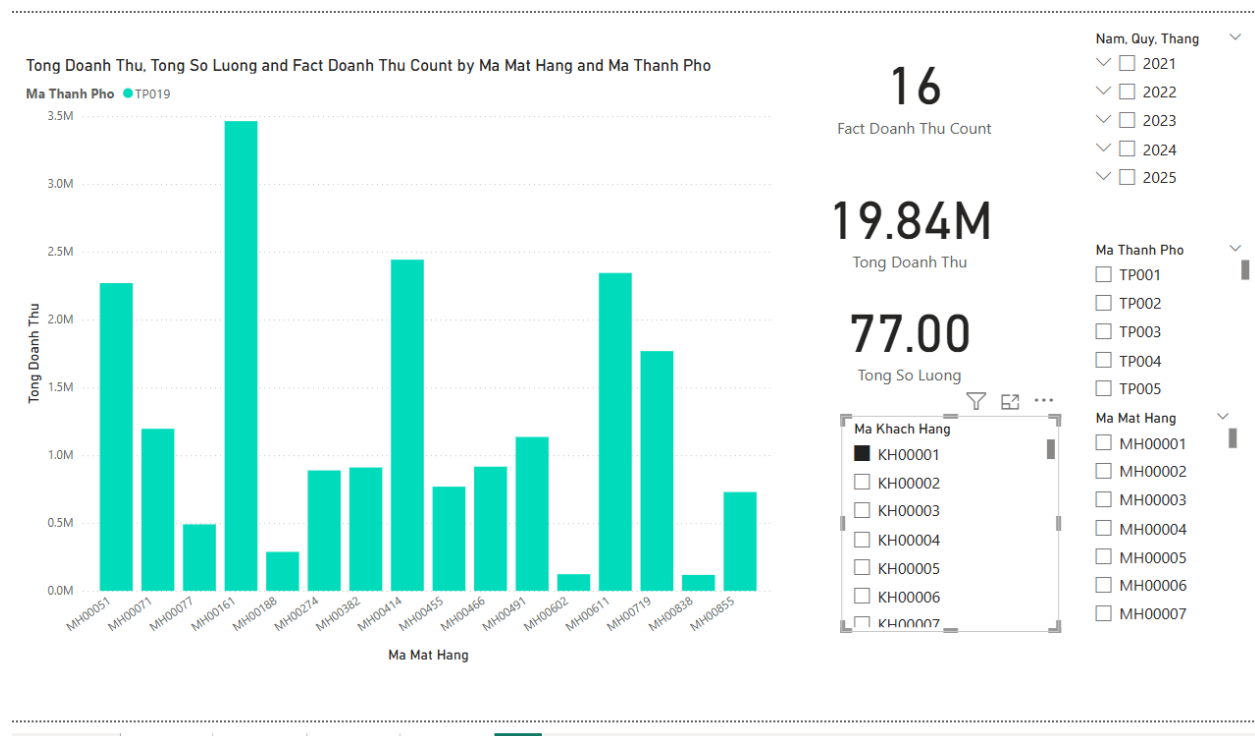
1 SELECT
2     KH.maKhachHang,
3     KH.tenKhachHang,
4     SUM(MDD.soLuongDat * MDD.giaDat) AS tongDoanhThu
5 FROM
6     TICHHOP_DB_1.dbo.MatHangDuocDat AS MDD
7 JOIN
8     TICHHOP_DB_1.dbo.DonDatHang AS DDH ON MDD.maDon = DDH.maDon
9 JOIN
10    TICHHOP_DB_1.dbo.KhachHang AS KH ON DDH.maKhachHang = KH.maKhachHang
11 WHERE
12     KH.maKhachHang = 'KH00001'
13 GROUP BY
14     KH.maKhachHang,
15     KH.tenKhachHang;

```

110 %

Results Messages

	maKhachHang	tenKhachHang	tongDoanhThu
1	KH00001	Tùng Trí Búi	19844734.77



Chương VIII: Kết luận

- Thông qua dự án này, nhóm đã xây dựng thành công một hệ thống kho dữ liệu hoàn chỉnh, từ khâu thiết kế mô hình, tích hợp dữ liệu, đến triển khai các công cụ phân tích trực quan, góp phần giải quyết bài toán quản lý thông tin khách hàng, sản phẩm và doanh thu cho doanh nghiệp. Giá trị cốt lõi mà hệ thống mang lại chính là khả năng tổng hợp và phân tích dữ liệu đa chiều theo thời gian, khu vực, mặt hàng và khách hàng, giúp ban lãnh đạo có được cái nhìn tổng quan và sâu sắc để đưa ra quyết định chính xác, nhanh chóng và hiệu quả hơn.
- Bằng cách chuẩn hóa dữ liệu từ nhiều nguồn, tự động hóa quá trình ETL, xây dựng các cube dữ liệu phục vụ phân tích OLAP và kết nối với Power BI để hiển thị trực quan, hệ thống không chỉ cải thiện hiệu quả quản trị dữ liệu mà còn tạo ra nền tảng dữ liệu ổn định cho việc triển khai các chiến lược phân tích nâng cao trong tương lai. Việc ứng dụng các phép toán như drill-down, roll-up, slice, dice hay pivot không những mang lại sự linh hoạt trong truy vấn thông tin, mà còn giúp người quản lý dễ dàng tiếp cận các chỉ số quan trọng như doanh thu, tồn kho, mức độ phân phối hàng hóa tại từng khu vực.
- Bên cạnh những giá trị nổi bật, nhóm cũng mạnh dạn thử nghiệm triển khai gateway để đồng bộ dữ liệu với nền tảng Power BI online nhằm hướng đến mô hình báo cáo thời gian thực. Mặc dù hiện tại còn gặp khó khăn kỹ thuật liên quan đến xung đột cổng PORT khi cài đặt gateway, đây là bước khởi đầu cho việc mở rộng hệ thống theo hướng hiện đại hóa và kết nối với hạ tầng đám mây. Điều này thể hiện tinh thần chủ động tiếp cận công nghệ mới và tư duy phát triển hệ thống lâu dài.
- Dự án đã chứng minh rằng, với cách tiếp cận bài bản, từ phân tích nghiệp vụ đến triển khai kỹ thuật, việc xây dựng kho dữ liệu không chỉ là giải pháp công nghệ mà còn là đòn bẩy chiến lược để doanh nghiệp nâng cao năng lực cạnh tranh, tối ưu vận hành và định hướng phát triển bền vững trong kỷ nguyên dữ liệu.