BỘ THÔNG TIN VÀ TRUYỀN THÔNG

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

BÁO CÁO
ĐỒ ÁN MÔN HỌC

ĐỀ TÀI: QUẢN LÝ KHO VẬT TƯ ĐA CHI NHÁNH
HỆ CƠ SỞ DỮ LIỆU PHÂN TÁN TRÊN BA SITE

Môn học: Cơ sở dữ liệu phân tán

Giảng viên hướng dẫn: [[Điền tên giảng viên]]

Nhóm thực hiện: [[Nhóm __]]

Thực hiện bởi nhóm sinh viên, bao gồm:

1. [[Họ tên thành viên 01]]	[[MSSV]]	Trưởng nhóm
2. [[Họ tên thành viên 02]]	[[MSSV]]	Thành viên
3. [[Họ tên thành viên 03]]	[[MSSV]]	Thành viên
4. [[Họ tên thành viên 04]]	[[MSSV]]	Thành viên
5. [[Họ tên thành viên 05]]	[[MSSV]]	Thành viên

TP.HCM, tháng 9 / 2026

MỤC LỤC

CHƯƠNG I. TỔNG QUAN 8
1. Giới thiệu đề tài 8
1.1. Đề tài được giao 8
1.2. Nhu cầu và tầm quan trọng của dự án 8
1.3. Sơ lược về dự án và các nhiệm vụ chính 8
1.4. Vì sao bắt buộc phải dùng cơ sở dữ liệu phân tán 9
1.5. Dữ liệu khi triển khai 10
1.6. Các đối tượng tham gia sử dụng 10
2. Cơ sở lý thuyết 11
2.1. Phân mảnh ngang nguyên thủy 11
2.2. Phân mảnh ngang dẫn xuất 11
2.3. Nhân bản dữ liệu 11
2.4. Giao tác phân tán và giao thức chuẩn bị hai pha 12
2.5. Điều khiển tương tranh 12
2.6. Các mức trong suốt 13
CHƯƠNG II. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 14
1. Phân tích 14
1.1. Các chức năng chính truy cập vào dữ liệu 14
1.2. Bảng tần suất truy cập tại các vị trí 14
1.3. Phân quyền cho các nhóm đối tượng 15
1.4. Phân tích chức năng của từng vị trí 15
1.5. Chức năng ở máy trạm và máy chủ 16
1.6. Phân tích cơ sở dữ liệu — mô hình thực thể liên kết 16
2. Thiết kế 18
2.1. Thiết kế cơ sở dữ liệu quan hệ 18
2.2. Lược đồ phân mảnh ngang nguyên thủy 19
2.3. Lược đồ phân mảnh ngang dẫn xuất 20
2.4. Lược đồ nhân bản 20
2.5. Lược đồ định vị 21
2.6. Lược đồ ánh xạ 22
2.7. Thiết kế kiến trúc hệ thống 23
2.8. Mô hình hệ thống tại chi nhánh và toàn hệ thống 23
CHƯƠNG III. TRIỂN KHAI HỆ THỐNG 25
1. Cài đặt mạng riêng ảo 25

2. Tạo đường liên kết mạng giữa các máy chủ 26
3. Cài đặt SQL Server 30
4. Kiểm tra dịch vụ Agent 36
5. Tạo liên kết cơ sở dữ liệu giữa các máy chủ 37
6. Tạo Publication 41
7. Thử các giao tác tương ứng với các chức năng 45
7.1. Nhập dữ liệu 45
7.2. Hiển thị dữ liệu 45
7.3. Thống kê 49
7.4. Trigger phân quyền bảo vệ các bảng 51
7.5. Thử các giao tác 55
CHƯƠNG IV. KẾT LUẬN 62
1. Kết quả đạt được 62
2. Những vấn đề đã gặp và cách khắc phục 62
3. Đánh giá 63
4. Hướng phát triển 63
TÀI LIỆU THAM KHẢO 65

DANH SÁCH HÌNH, BẢNG

DANH SÁCH HÌNH

Hình II.1: Sơ đồ quan hệ do SSMS sinh tại site KHO_A — chín bảng nghiệp vụ và bảng nhật ký kiểm toán

Hình III.1: Network riêng của nhóm vừa được tạo, kèm Network ID mười sáu ký tự dùng để mời hai máy còn lại

Hình III.2: Danh sách thành viên của network. Mỗi máy chỉ vào được sau khi người quản trị tích ô Auth

Hình III.3: Địa chỉ IP ảo mà ZeroTier cấp cho máy chủ, đối chiếu giữa trang quản lý network và lệnh ipconfig trên chính máy đó

Hình III.4: Kết quả tập lệnh cấu hình mạng: ba cổng đã được lắng nghe, bảy dịch vụ đều ở trạng thái Running và Automatic

Hình III.5: Giao thức TCP/IP được bật cho thể hiện SQL Server trong SQL Server Configuration Manager

Hình III.6: Gán cổng cố định tại mục IPAll, đồng thời xoá trắng ô cổng động

Hình III.7: Các luật tường lửa mở cho ba cổng thể hiện, cổng UDP 1434 của SQL Server Browser và dịch vụ MS DTC

Hình III.8: Kết nối qua địa chỉ IP ảo cho net_transport bằng TCP — dữ liệu thật sự đi qua mạng chứ không qua bộ nhớ chung

Hình III.9: Chọn phiên bản Developer — miễn phí và đầy đủ tính năng

Hình III.10: Chọn tính năng cài đặt. Bắt buộc tích SQL Server Replication, thiếu mục này là không làm được nhân bản

Hình III.11: Đặt tên thể hiện là KHO_A. Đây là bước quyết định ba site độc lập với nhau

Hình III.12: Đặt SQL Server Agent ở chế độ Automatic — điều kiện bắt buộc để các tiến trình nhân bản chạy được

Hình III.13: Chọn Mixed Mode để dùng được tài khoản sa. Linked Server giữa các site sẽ đăng nhập bằng tài khoản này

Hình III.14: Thư mục dữ liệu riêng của từng thể hiện

Hình III.15: Cài đặt thể hiện KHO_A hoàn tất, cả Database Engine và Replication đều Succeeded

Hình III.16: Cài thể hiện thứ hai. Bảng Installed instances đã có KHO_A, chọn Perform a new installation để tạo thêm thể hiện mới

Hình III.17: Cài thể hiện thứ ba KHO_C. Bảng dưới cho thấy KHO_A và KHO_B đã tồn tại — ba thể hiện chạy song song trên cùng một máy

Hình III.18: Hoàn tất cài đặt cả ba thể hiện

Hình III.19: Ba thể hiện hiện diện cùng lúc trong Object Explorer của SSMS — hình ảnh đại diện cho toàn bộ hệ thống phân tán

Hình III.20: Bảy dịch vụ của hệ thống đều ở trạng thái Running và Automatic

Hình III.21: Hai Linked Server tại site KHO_A trong Object Explorer

Hình III.22: Trang General — khai báo nguồn dữ liệu của liên kết

Hình III.23: Trang Security — ánh xạ đăng nhập sang tài khoản cố định ở máy đích

Hình III.24: Truy vấn phân tán đầu tiên — đứng tại KHO_A nhưng đọc được dữ liệu tồn kho của cả ba site
Hình III.25: Publication PUB_DanhMuc trong Object Explorer tại site KHO_A
Hình III.26: Ba bảng danh mục được chọn làm article của Publication
Hình III.27: Replication Monitor — hai Subscription đều ở trạng thái hoạt động
Hình III.28: Bằng chứng đồng bộ — sửa danh mục tại KHO_A, sau khoảng 20 giây cả KHO_B và KHO_C đều tự động cập nhật theo
Hình III.29: Tồn kho toàn hệ thống — 24 dòng gộp từ ba site, hiển thị như một bảng duy nhất
Hình III.30: Kiểm chứng ba tính chất của phân mảnh cùng phép thử vi phạm
Hình III.31: So khớp ba bản sao danh mục bằng vân tay CHECKSUM_AGG
Hình III.32: Danh sách liên kết, ánh xạ đăng nhập, kết quả thử kết nối và thông tin phiên bản đọc ngược từ hai site kia
Hình III.33: Thủ tục sp_TonKhoToanHeThong dựng câu lệnh tự động từ bảng Kho
Hình III.34: Thống kê toàn hệ thống theo kho, theo vật tư và theo tỷ trọng vốn
Hình III.35: Cảnh báo thiếu hàng kèm đề xuất kho cấp bù
Hình III.36: So sánh hai cách viết truy vấn phân tán qua thời gian thực thi
Hình III.37: Bảng tổng hợp quyền đã cấp cho bốn vai trò và danh sách trigger đang hoạt động
Hình III.38: Kết quả thử quyền — ai làm được gì, ai bị chặn
Hình III.39: Thử trigger tại site KHO_A — cột LỚP CHẶN chỉ rõ hàng phòng thủ nào đã bắt được vi phạm
Hình III.40: Cùng kịch bản chạy tại site KHO_B. Bốn dòng đầu giống hệt, riêng dòng thứ năm đổi từ được phép sang bị chặn — bằng chứng cho thấy trigger phân biệt được vai trò Publisher và Subscriber
Hình III.41: Nhật ký kiểm toán do trigger ghi lại
Hình III.42: Kịch bản T1 — tồn kho trước và sau khi điều chuyển, phiếu được ghi ở cả hai site
Hình III.43: Thông báo của kịch bản T1
Hình III.44: Kịch bản T2 — tồn kho hai site trước và sau đều y nguyên
Hình III.45: Thông báo từ chối của kịch bản T2
Hình III.46: Kịch bản T3 — tồn kho và số phiếu ở cả hai site đều y nguyên sau khi giao tác bị huỷ
Hình III.47: Thông báo quay lui của kịch bản T3
Hình III.48: Kịch bản C1 — hiện tượng mất cập nhật khi không dùng khoá
Hình III.49: Kịch bản C2 — khoá bi quan khắc phục triệt để
Hình III.50: Kịch bản C3 — khoá lạc quan bằng cột ROWVERSION
Hình III.51: Kịch bản C4 — khoá chết và cơ chế tự gỡ của SQL Server
Hình III.52: Quan sát các khoá đang tồn tại trong hệ thống

DANH SÁCH BẢNG

Bảng I.1: Ba site trong hệ thống
Bảng I.2: Ba nhóm bảng theo cách phân bố
Bảng I.3: Bốn nhóm người dùng và phạm vi quyền
Bảng II.1: Bảy chức năng chính của hệ thống
Bảng II.2: Tần suất truy cập theo chức năng và vị trí (lượt/ngày)
Bảng II.3: Bốn vai trò và quyền tương ứng
Bảng II.4: Nhiệm vụ riêng của từng site
Bảng II.5: Phân chia chức năng giữa máy chủ và máy trạm
Bảng II.6: Các mối kết hợp và bản số
Bảng II.7: Danh sách chín bảng của hệ thống
Bảng II.8: Quy ước sinh mã
Bảng II.9: Danh sách khoá ngoại
Bảng II.10: Ba mảnh ngang nguyên thủy của TonKho
Bảng II.11: Cấu hình nhân bản
Bảng II.12: Lược đồ định vị dữ liệu trên ba site
Bảng II.13: Hai đường đồng bộ hoá
Bảng III.1: Cổng TCP được gán cho ba thể hiện
Bảng III.2: Bốn tuỳ chọn của Linked Server
Bảng III.3: Năm thủ tục nghiệp vụ
Bảng III.4: Kết quả kiểm chứng phân mảnh
Bảng III.5: Số liệu thống kê toàn hệ thống
Bảng III.6: Ba lớp bảo vệ dữ liệu
Bảng III.7: Danh sách trigger bảo vệ
Bảng III.8: Kết quả thử năm hành vi vi phạm
Bảng III.9: Kết quả kịch bản T1
Bảng III.10: Tổng kết ba kịch bản giao tác phân tán
Bảng III.11: Tổng kết bốn kịch bản tương tranh
Bảng IV.1: Đối chiếu yêu cầu đề tài với kết quả
Bảng IV.2: Các sự cố đã xử lý

TÓM TẮT

Báo cáo trình bày quá trình phân tích, thiết kế và triển khai một **hệ cơ sở dữ liệu phân tán quản lý kho vật tư đa chi nhánh**, thực hiện trên ba site SQL Server độc lập.

Xuất phát từ khảo sát tần suất truy cập cho thấy **89% nghiệp vụ hằng ngày chỉ đụng tới dữ liệu của chính kho đó**, nhóm quyết định **phân mảnh ngang** bảng tồn kho và chứng từ theo mã kho, đồng thời **nhân bản** ba bảng danh mục vì chúng có tỷ lệ đọc trên ghi rất cao. Bảng chi tiết chứng từ không chứa mã kho nên được **phân mảnh ngang dẫn xuất** bằng phép nửa nối theo phiếu.

Hệ thống cài đặt đầy đủ các kỹ thuật cốt lõi của môn học: sáu **Linked Server** nối ba site thành đồ thị hai chiều; **Transactional Replication** đồng bộ danh mục với độ trễ đo được 15–20 giây; **giao tác phân tán hai pha** qua MS DTC cho nghiệp vụ điều chuyển vật tư; bốn kịch bản **điều khiển tương tranh**; và các **truy vấn phân tán** gộp dữ liệu từ cả ba máy chủ.

Yêu cầu trọng tâm của đề tài — *điều chuyển 50 sản phẩm từ Kho A sang Kho B, nếu thất bại giữa chừng thì dữ liệu phải nhất quán* — được kiểm chứng bằng ba kịch bản chạy thật. Trong kịch bản gây lỗi cố ý **sau khi kho nguồn đã bị trừ hàng**, giao thức hai pha đã quay lui toàn bộ ở cả hai máy chủ, tồn kho và số chứng từ trở về đúng trạng thái ban đầu, **không một đơn vị hàng nào bị mất**.

Dữ liệu được bảo vệ bằng **ba lớp chồng nhau**: ràng buộc kiểm tra ở tầng dữ liệu, sáu trigger cho những luật mà ràng buộc không diễn tả nổi, và hệ thống bốn vai trò buộc mọi thay đổi tồn kho phải đi qua thủ tục đã kiểm soát.

CHƯƠNG I. TỔNG QUAN

1. Giới thiệu đề tài

1.1. Đề tài được giao

Căn cứ bảng phân công đề tài đồ án cuối kỳ môn Cơ sở dữ liệu phân tán do giảng viên công bố, nhóm được giao **Đề tài 2 — Quản lý kho vật tư đa chi nhánh**.

Yêu cầu của đề tài là xây dựng một hệ cơ sở dữ liệu phân tán phục vụ doanh nghiệp có nhiều kho vật tư đặt tại các địa phương khác nhau, trong đó phải thể hiện được đầy đủ các kỹ thuật cốt lõi của môn học: phân mảnh dữ liệu, nhân bản dữ liệu, giao tác phân tán, điều khiển tương tranh và truy vấn phân tán.

Kịch bản bắt buộc mà đề tài nêu rõ: **Kho A chuyển 50 sản phẩm sang Kho B; nếu giao dịch thất bại giữa chừng thì dữ liệu phải được xử lý nhất quán.** Toàn bộ thiết kế của nhóm xoay quanh việc bảo đảm yêu cầu này.

1.2. Nhu cầu và tầm quan trọng của dự án

Một doanh nghiệp thương mại vật tư xây dựng thường có kho trung tâm và nhiều kho vệ tinh đặt tại các tỉnh thành. Mỗi kho hằng ngày phát sinh hàng trăm lượt nhập, xuất, tra cứu tồn. Nếu toàn bộ dữ liệu được đặt tập trung tại một máy chủ duy nhất ở trụ sở chính, hệ thống sẽ vấp phải ba vấn đề nghiêm trọng.

- **Phụ thuộc đường truyền.** Nhân viên kho ở tỉnh muốn lập một phiếu xuất cũng phải gửi yêu cầu qua Internet về trụ sở. Đường truyền chậm hoặc đứt là toàn bộ hoạt động của kho ngưng trệ, hàng hoá không xuất được cho khách.
- **Nghẽn cổ chai tại máy chủ trung tâm.** Toàn bộ tải của mọi chi nhánh dồn về một nơi, trong khi phần lớn nghiệp vụ chỉ liên quan tới dữ liệu của chính kho đó.
- **Rủi ro điểm hỏng duy nhất.** Máy chủ trung tâm gặp sự cố thì cả doanh nghiệp dừng hoạt động.

Cơ sở dữ liệu phân tán giải quyết cả ba vấn đề trên bằng cách đặt dữ liệu ở nơi nó được sử dụng nhiều nhất, đồng thời vẫn cho phép nhìn toàn bộ hệ thống như một cơ sở dữ liệu thống nhất.

1.3. Sơ lược về dự án và các nhiệm vụ chính

Hệ thống được xây dựng gồm **ba site độc lập**, mỗi site là một thể hiện (instance) SQL Server riêng biệt, quản lý dữ liệu của một kho.

Bảng I.1: Ba site trong hệ thống

Site	Thể hiện SQL Server	CSDL	Kho	Vai trò phân tán
S1	MIGNON\KHO_A	KhoA	Kho Trung tâm (Hà Nội)	Publisher + Distributor
S2	MIGNON\KHO_B	KhoB	Kho Miền Bắc (Bắc Ninh)	Subscriber
S3	MIGNON\KHO_C	KhoC	Kho Miền Nam (TP. Hồ Chí Minh)	Subscriber

Các nhiệm vụ chính mà hệ thống phải thực hiện:

1. Nhập vật tư vào kho tại chỗ, có kiểm soát chứng từ.
2. Xuất vật tư khỏi kho tại chỗ, có kiểm tra tồn trước khi xuất.
3. Tra cứu tồn kho tại chỗ phục vụ nghiệp vụ hằng ngày.
4. Kiểm tra tồn kho **toàn hệ thống** — dữ liệu gộp từ cả ba site.
5. **Điều chuyển vật tư giữa hai kho** bằng giao tác phân tán, bảo đảm không bao giờ xảy ra tình trạng kho nguồn đã trừ hàng mà kho đích chưa cộng.
6. **Đồng bộ danh mục vật tư** từ Kho Trung tâm xuống hai kho còn lại.
7. Cảnh báo vật tư xuống dưới mức tồn tối thiểu.

1.4. Vì sao bắt buộc phải dùng cơ sở dữ liệu phân tán

Đây là câu hỏi cốt lõi, và nhóm trả lời bằng **số liệu tần suất truy cập** chứ không bằng cảm tính. Bảng khảo sát tần suất (trình bày chi tiết ở Chương II) cho thấy:

- Ba chức năng **nhập kho, xuất kho, tra cứu tồn tại chỗ** chiếm **565 trên tổng số 634 lượt truy cập mỗi ngày, tương đương 89%**, và toàn bộ đều **chỉ đụng tới dữ liệu của chính kho đó**.
- Chức năng **sửa danh mục vật tư** chỉ phát sinh **6 lần mỗi ngày và chỉ tại Kho Trung tâm**, nhưng **mọi site đều phải đọc danh mục liên tục**.
- Chỉ hai chức năng **tra cứu toàn hệ thống (19 lượt)** và **điều chuyển vật tư (18 lượt)** là bắt buộc vượt ra ngoài phạm vi một site.

Ba con số trên dẫn thẳng tới ba quyết định thiết kế:

1. 89% nghiệp vụ cục bộ theo kho ⟹ **phân mảnh ngang** bảng tồn kho và chứng từ theo mã kho, để phần lớn công việc chạy tại chỗ, không đi qua mạng.
2. Danh mục có tỷ lệ đọc trên ghi rất cao ⟹ **nhân bản** thay vì phân mảnh, để mọi site đọc được tại chỗ mà vẫn thống nhất một nguồn.
3. Hai nghiệp vụ vượt site ⟹ cần **truy vấn phân tán** và **giao tác phân tán hai pha**.

Nếu dùng cơ sở dữ liệu tập trung, 89% lượt truy cập lẽ ra chạy tại chỗ sẽ bị đẩy hết qua đường truyền — đó chính là lý do kỹ thuật buộc phải chọn mô hình phân tán.

1.5. Dữ liệu khi triển khai

Cơ sở dữ liệu gồm **9 bảng**, chia làm ba nhóm theo cách phân bố dữ liệu:

Bảng I.2: Ba nhóm bảng theo cách phân bố

Nhóm	Các bảng	Cách phân bố
Danh mục	NhaCungCap, VatTu, Kho	Nhân bản — ba site giữ ba bản giống hệt nhau
Tồn kho	TonKho	Phân mảnh ngang nguyên thủy theo MaKho
Chứng từ	PhieuNhap, ChiTietNhap, PhieuXuat, ChiTietXuat	Phân mảnh ngang nguyên thủy và dẫn xuất
Điều chuyển	PhieuDieuChuyen	Nhân bản một phần — ghi ở cả hai kho liên quan

Dữ liệu mẫu khi triển khai: 4 nhà cung cấp, 10 mặt hàng vật tư xây dựng, và tồn kho phân bổ thành **24 dòng** — Kho Trung tâm giữ 10 mặt hàng, hai kho vệ tinh mỗi kho 7 mặt hàng.

1.6. Các đối tượng tham gia sử dụng

Bảng I.3: Bốn nhóm người dùng và phạm vi quyền

Nhóm người dùng	Làm việc tại	Quyền được cấp
Nhân viên kho	Kho sở tại	Đọc mọi bảng của kho mình; nhập và xuất kho thông qua thủ tục, không được ghi thẳng vào bảng
Trưởng kho	Kho sở tại	Toàn bộ quyền của nhân viên kho, cộng thêm quyền khởi tạo giao tác điều chuyển sang kho khác
Quản trị danh mục	Chỉ Kho Trung tâm	Toàn quyền trên danh mục vật tư và nhà cung cấp — nguồn của nhân bản
Ban giám đốc	Kho Trung tâm	Chỉ đọc, nhưng đọc được dữ liệu của cả ba site qua Linked Server

2. Cơ sở lý thuyết

2.1. Phân mảnh ngang nguyên thủy

Phân mảnh ngang là phép chia một quan hệ toàn cục thành nhiều tập con các bộ (dòng), mỗi tập con đặt tại một site. Phân mảnh ngang **nguyên thủy** dùng phép chọn với một vị từ đặt trên chính quan hệ đó:

```
R_i  =  σ (p_i) (R)
```

Một phép phân mảnh hợp lệ phải thoả **ba tính chất**:

- **Tính đầy đủ (completeness).** Mọi bộ của quan hệ toàn cục phải thuộc về ít nhất một mảnh, không được để sót dữ liệu.
- **Tính tái thiết (reconstruction).** Phải khôi phục lại được quan hệ gốc từ các mảnh. Với phân mảnh ngang, phép tái thiết là phép hợp: $R = R_1 \cup R_2 \cup \ldots \cup R_n$.
- **Tính tách rời (disjointness).** Các mảnh không được giao nhau, một bộ chỉ nằm ở đúng một mảnh.

Tập vị từ dùng để phân mảnh được rút gọn bằng thuật toán **COM_MIN**, nhằm thu được tập vị từ đầy đủ và cực tiểu, tránh sinh ra những mảnh thừa không ai dùng tới.

2.2. Phân mảnh ngang dẫn xuất

Khi một quan hệ **không chứa thuộc tính dùng làm tiêu chí phân mảnh**, ta không thể phân mảnh nguyên thủy cho nó. Lúc này phải dùng phân mảnh ngang **dẫn xuất**: mảnh của quan hệ con được xác định theo mảnh của quan hệ cha thông qua **phép nửa nối (semijoin)**:

```
S_i  =  S  ⋉  R_i
```

Điều kiện để áp dụng: đồ thị nối giữa các quan hệ phải có **dạng cây đơn giản**, mỗi quan hệ thành viên có đúng một quan hệ chủ. Tính tách rời được bảo đảm khi quan hệ chủ và quan hệ thành viên có liên kết **một–nhiều**.

2.3. Nhân bản dữ liệu

Nhân bản là việc giữ nhiều bản sao của cùng một dữ liệu tại nhiều site. Nhân bản làm tăng tốc độ đọc và tăng khả năng chịu lỗi, nhưng phải trả giá bằng chi phí giữ các bản sao nhất quán với nhau. Vì vậy nhân bản chỉ phù hợp với dữ liệu có **tỷ lệ đọc trên ghi cao**.

SQL Server cung cấp cơ chế **Transactional Replication** với ba vai trò:

- **Publisher** — site giữ bản gốc, nơi duy nhất được phép sửa dữ liệu.

- **Distributor** — site trung gian lưu các thay đổi đọc được từ nhật ký giao dịch và phân phát đi.
- **Subscriber** — site nhận bản sao, chỉ đọc.

Hai tiến trình nền đảm nhiệm việc truyền dữ liệu: **Log Reader Agent** đọc nhật ký giao dịch của Publisher, **Distribution Agent** đẩy thay đổi xuống Subscriber.

2.4. Giao tác phân tán và giao thức chuẩn bị hai pha

Một giao tác phân tán là giao tác tác động tới dữ liệu nằm trên nhiều site. Nó vẫn phải bảo đảm đầy đủ bốn tính chất **ACID**, trong đó khó nhất là **tính nguyên tố (Atomicity)**: hoặc mọi site cùng thành công, hoặc mọi site cùng quay lui.

Giao thức **chuẩn bị hai pha (Two-Phase Commit, 2PC)** giải quyết bài toán này:

- **Pha 1 — Chuẩn bị.** Bộ điều phối hỏi tất cả các site tham gia: "đã sẵn sàng kết thúc chưa?". Mỗi site thực hiện phần việc của mình, ghi nhật ký, rồi trả lời **YES** hoặc **NO**.
- **Pha 2 — Kết thúc.** Nếu **mọi** site trả lời YES, bộ điều phối ra lệnh commit đồng loạt. Chỉ cần **một** site trả lời NO hoặc mất kết nối, bộ điều phối ra lệnh **rollback toàn bộ ở mọi site**.

Trên nền Windows, vai trò bộ điều phối do dịch vụ **MS DTC (Microsoft Distributed Transaction Coordinator)** đảm nhiệm.

2.5. Điều khiển tương tranh

Khi nhiều giao tác cùng truy cập một dữ liệu, có thể xảy ra các dị thường, trong đó phổ biến nhất là **mất cập nhật (lost update)**: hai giao tác cùng đọc một giá trị, cùng tính toán trên giá trị cũ, rồi lần ghi sau xoá mất kết quả của lần ghi trước.

Có hai hướng xử lý:

- **Khoá bi quan (pessimistic).** Đặt khoá ngay lúc đọc, buộc giao tác đến sau phải chờ. Trong SQL Server dùng gợi ý `WITH (UPDLOCK, HOLDLOCK)`. Kết quả luôn đúng nhưng người đến sau phải đợi.
- **Khoá lạc quan (optimistic).** Không khoá, nhưng lúc ghi thì kiểm tra xem dòng dữ liệu có bị ai sửa trong lúc mình xử lý hay không, thường bằng cột phiên bản `ROWVERSION`. Nếu phát hiện xung đột thì từ chối ghi và báo người dùng làm lại.

Việc đặt khoá có thể dẫn tới **khoá chết (deadlock)** khi hai giao tác khoá hai tài nguyên theo thứ tự ngược nhau. SQL Server có bộ giám sát khoá tự phát hiện vòng tròn chờ,

chọn một giao tác làm **nạn nhân** và huỷ nó bằng lỗi **Msg 1205**, nhờ đó hệ thống tự gỡ mà không cần con người can thiệp.

2.6. Các mức trong suốt

Mục tiêu cuối cùng của một hệ cơ sở dữ liệu phân tán là làm cho người dùng **không cảm thấy sự phân tán**. Các mức trong suốt được cài đặt trong đồ án này gồm:

- **Trong suốt phân mảnh.** Người dùng viết truy vấn trên quan hệ toàn cục, không cần biết dữ liệu bị cắt thành mấy mảnh.
- **Trong suốt vị trí.** Người dùng không cần biết mảnh dữ liệu nằm ở máy nào.
- **Trong suốt nhân bản.** Người dùng không cần biết dữ liệu có bao nhiêu bản sao và bản sao nào đang được đọc.
- **Trong suốt giao tác.** Người dùng gọi một lệnh, hệ thống tự lo việc bảo đảm nguyên tố trên nhiều site.

CHƯƠNG II. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

1. Phân tích

1.1. Các chức năng chính truy cập vào dữ liệu

Bảng II.1: Bảy chức năng chính của hệ thống

Mã	Chức năng	Site thực hiện	Bảng truy cập	Kiểu truy cập
F1	Nhập vật tư vào kho	Kho sở tại	PhieuNhap, ChiTietNhap, TonKho	Ghi, cục bộ
F2	Xuất vật tư khỏi kho	Kho sở tại	PhieuXuat, ChiTietXuat, TonKho	Ghi, cục bộ
F3	Tra cứu tồn kho tại chỗ	Kho sở tại	TonKho, VatTu	Đọc, cục bộ
F4	Kiểm tra tồn kho toàn hệ thống	Kho Trung tâm	TonKho ở cả ba site	Đọc, **phân tán**
F5	Điều chuyển vật tư giữa hai kho	Kho nguồn	TonKho ở hai site, PhieuDieuChuyen	Ghi, **giao tác phân tán**
F6	Thêm, sửa danh mục vật tư	Chỉ Kho Trung tâm	VatTu, NhaCungCap	Ghi, **nhân bản xuống hai site**
F7	Cảnh báo dưới mức tồn tối thiểu	Mọi site	TonKho, VatTu	Đọc, cục bộ

1.2. Bảng tần suất truy cập tại các vị trí

Đây là căn cứ định lượng cho toàn bộ các quyết định thiết kế phía sau. Số liệu là ước lượng số lượt thao tác mỗi ngày trong điều kiện vận hành bình thường.

Bảng II.2: Tần suất truy cập theo chức năng và vị trí (lượt/ngày)

Chức năng	Kho A	Kho B	Kho C	Tổng
F1 — Nhập kho	40	25	30	95
F2 — Xuất kho	60	45	55	160
F3 — Tra cứu tồn tại chỗ	120	90	100	310
F4 — Tồn kho toàn hệ thống	15	2	2	19
F5 — Điều chuyển	8	5	5	18
F6 — Sửa danh mục	6	0	0	6
F7 — Cảnh báo tồn tối thiểu	10	8	8	26
Tổng cộng	**259**	**175**	**200**	**634**

Nhận xét dẫn tới thiết kế:

1. F1, F2, F3 chiếm **565/634 ≈ 89%** tổng lượt truy cập và hoàn toàn cục bộ theo kho ⟹ phải **phân mảnh ngang** bảng TonKho và chứng từ theo MaKho.

2. F6 chỉ xảy ra ở Kho Trung tâm 6 lần/ngày nhưng mọi site đều cần đọc danh mục ⟹ **nhân bản** ba bảng danh mục.
3. F4 và F5 bắt buộc vượt site ⟹ cần **truy vấn phân tán** và **giao tác phân tán**.

1.3. Phân quyền cho các nhóm đối tượng

Hệ thống thiết kế bốn vai trò, ánh xạ đúng theo bốn nhóm người dùng đã nêu ở Chương I.

Bảng II.3: Bốn vai trò và quyền tương ứng

Vai trò	Phạm vi	Quyền được cấp
NhanVienKho	Kho sở tại	SELECT trên mọi bảng của kho mình; EXECUTE trên các thủ tục nghiệp vụ. Bị DENY ghi thẳng vào TonKho, PhieuNhap, PhieuXuat
TruongKho	Kho sở tại	Kế thừa NhanVienKho, thêm EXECUTE trên sp_DieuChuyenVatTu và SELECT nhật ký kiểm toán
QuanTriDanhMuc	**Chỉ Kho Trung tâm**	Toàn quyền trên VatTu và NhaCungCap
BanGiamDoc	Kho Trung tâm	Chỉ SELECT, nhưng đọc được cả ba site qua Linked Server

Nguyên tắc thiết kế quan trọng nhất: nhân viên kho **không được cấp quyền ghi thẳng** vào bảng dữ liệu, mà chỉ có quyền chạy thủ tục. Nhờ cơ chế **chuỗi sở hữu (ownership chaining)** — thủ tục và bảng cùng thuộc lược đồ dbo — thủ tục vẫn ghi được vào bảng dù người gọi bị cấm. Hệ quả là **mọi thay đổi tồn kho bắt buộc đi qua đoạn mã đã kiểm tra tồn, đã đặt khoá và đã ghi chứng từ**, không tồn tại đường tắt nào.

Vai trò **QuanTriDanhMuc chỉ tồn tại ở Kho Trung tâm**. Đây là điểm cho thấy phân quyền được thiết kế bám sát kiến trúc phân tán: quyền sửa danh mục chỉ được cấp đúng ở nơi giữ vai trò Publisher.

1.4. Phân tích chức năng của từng vị trí

Bảng II.4: Nhiệm vụ riêng của từng site

Site	Nhiệm vụ nghiệp vụ	Nhiệm vụ trong hệ phân tán
S1 — Kho Trung tâm	Nhập, xuất, tra cứu tồn của kho mình. Quản lý danh mục vật tư và nhà cung cấp cho toàn doanh nghiệp. Điều phối việc cấp bù hàng cho hai kho vệ tinh	Publisher và Distributor của nhân bản danh mục. Nơi khởi tạo phần lớn các giao tác phân tán. Nơi Ban giám đốc xem báo cáo toàn hệ thống
S2 — Kho Miền Bắc	Nhập, xuất, tra cứu tồn của kho mình. Nhận hàng điều chuyển từ Kho Trung tâm	Subscriber nhận danh mục. Site tham gia trong giao tác phân tán

Site	Nhiệm vụ nghiệp vụ	Nhiệm vụ trong hệ phân tán
S3 — Kho Miền Nam	Nhập, xuất, tra cứu tồn của kho mình. Nhận hàng điều chuyển từ Kho Trung tâm	Subscriber nhận danh mục. Site tham gia trong giao tác phân tán

1.5. Chức năng ở máy trạm và máy chủ

Hệ thống chia làm hai tầng. Nguyên tắc xuyên suốt là **đẩy việc xử lý xuống máy chủ, máy trạm chỉ lo giao diện** — vì trong cơ sở dữ liệu phân tán, chi phí đắt nhất là dữ liệu chạy trên đường truyền.

Bảng II.5: Phân chia chức năng giữa máy chủ và máy trạm

Tầng	Chức năng đảm nhiệm
Máy chủ (SQL Server instance)	Lưu trữ mảnh dữ liệu của site; cưỡng chế toàn vẹn bằng ràng buộc CHECK, khoá ngoại và trigger; xử lý nghiệp vụ trong các thủ tục; điều khiển tương tranh bằng khoá; điều phối giao tác phân tán qua MS DTC; chạy nhân bản; thực thi truy vấn phân tán; quản lý phân quyền; ghi nhật ký kiểm toán
Máy trạm (máy nhân viên)	Giao diện lập phiếu nhập, phiếu xuất, phiếu điều chuyển; kiểm tra sơ bộ dữ liệu nhập; gọi thủ tục nghiệp vụ; hiển thị kết quả và cảnh báo; in chứng từ; bắt lỗi deadlock 1205 và tự thử lại giao tác

Ranh giới giữa hai tầng được cưỡng chế bằng hệ thống quyền, không phải bằng quy ước lập trình. Máy trạm bị DENY quyền ghi trực tiếp, chỉ có quyền EXECUTE thủ tục.

1.6. Phân tích cơ sở dữ liệu — mô hình thực thể liên kết

Cơ sở dữ liệu gồm sáu thực thể chính và ba mối kết hợp nhiều–nhiều có thuộc tính riêng.

Sáu thực thể: NhaCungCap, VatTu, Kho, PhieuNhap, PhieuXuat, PhieuDieuChuyen.

Ba mối kết hợp nhiều–nhiều — trong mô hình thực thể liên kết chúng **không phải là thực thể**, chỉ trở thành bảng khi chuyển sang mô hình quan hệ:

```
KHO       ◇────< TỒN KHO >────◇  VẬT TƯ     thuộc tính: SoLuong, NgayCapNhat
PHIẾU NHẬP ◇──< CHI TIẾT NHẬP >──◇ VẬT TƯ  thuộc tính: SoLuong, DonGia
PHIẾU XUẤT ◇──< CHI TIẾT XUẤT >──◇ VẬT TƯ  thuộc tính: SoLuong, DonGia
```

Bảng II.6: Các mối kết hợp và bản số

Mối kết hợp	Thực thể A	Bản số	Thực thể B	Ý nghĩa
cung cấp	NhaCungCap	1 : N	VatTu	Mỗi vật tư do một nhà cung cấp chính
TỒN KHO	Kho	M : N	VatTu	Có thuộc tính SoLuong
lập tại	Kho	1 : N	PhieuNhap	Mỗi phiếu thuộc đúng một kho
nhập từ	NhaCungCap	1 : N	PhieuNhap	Mỗi phiếu mua của một nhà cung cấp
CHI TIẾT NHẬP	PhieuNhap	M : N	VatTu	Có SoLuong và DonGia
lập tại	Kho	1 : N	PhieuXuat	Mỗi phiếu thuộc đúng một kho

Mối kết hợp	Thực thể A	Bản số	Thực thể B	Ý nghĩa
CHI TIẾT XUẤT	PhieuXuat	M : N	VatTu	Có SoLuong và DonGia
chuyển đi	Kho	1 : N	PhieuDieuChuyen	Vai trò **kho nguồn**
chuyển đến	Kho	1 : N	PhieuDieuChuyen	Vai trò **kho đích**

Điểm đáng chú ý: thực thể Kho tham gia vào PhieuDieuChuyen **hai lần với hai vai trò khác nhau** — kho nguồn và kho đích. Đây chính là thực thể khiến hệ thống **bắt buộc** phải dùng giao tác phân tán, vì một phiếu điều chuyển đụng tới dữ liệu nằm ở hai site khác nhau.

2. Thiết kế

2.1. Thiết kế cơ sở dữ liệu quan hệ

Bảng II.7: Danh sách chín bảng của hệ thống

#	Bảng	Khoá chính	Ý nghĩa
1	NhaCungCap	MaNCC	Nhà cung cấp vật tư
2	VatTu	MaVT	Danh mục vật tư
3	Kho	MaKho	Danh sách ba kho và tên máy chủ tương ứng
4	TonKho	MaKho + MaVT	Số lượng tồn của từng vật tư tại từng kho
5	PhieuNhap	MaPN	Phiếu nhập kho
6	ChiTietNhap	MaPN + MaVT	Dòng hàng của phiếu nhập
7	PhieuXuat	MaPX	Phiếu xuất kho
8	ChiTietXuat	MaPX + MaVT	Dòng hàng của phiếu xuất
9	PhieuDieuChuyen	MaDC	Phiếu điều chuyển vật tư giữa hai kho

Quy ước mã — điểm này rất quan trọng trong hệ phân tán:

Bảng II.8: Quy ước sinh mã

Bảng	Quy ước	Ví dụ
Kho	KHO_A, KHO_B, KHO_C	KHO_A
VatTu	VT + 3 chữ số	VT001
NhaCungCap	NCC + 2 chữ số	NCC01
PhieuNhap	PN + ký tự kho + 4 chữ số	PNA0001
PhieuXuat	PX + ký tự kho + 4 chữ số	PXB0001
PhieuDieuChuyen	DC + ký tự kho nguồn + 3 chữ số	DCA001

Mã chứng từ **có nhúng ký tự kho** để bảo đảm **khoá chính không đụng nhau giữa ba site**. Đây là kỹ thuật bắt buộc khi khoá chính được sinh độc lập ở nhiều nơi — nếu ba site cùng sinh mã PN0001 thì khi gộp dữ liệu sẽ xung đột.

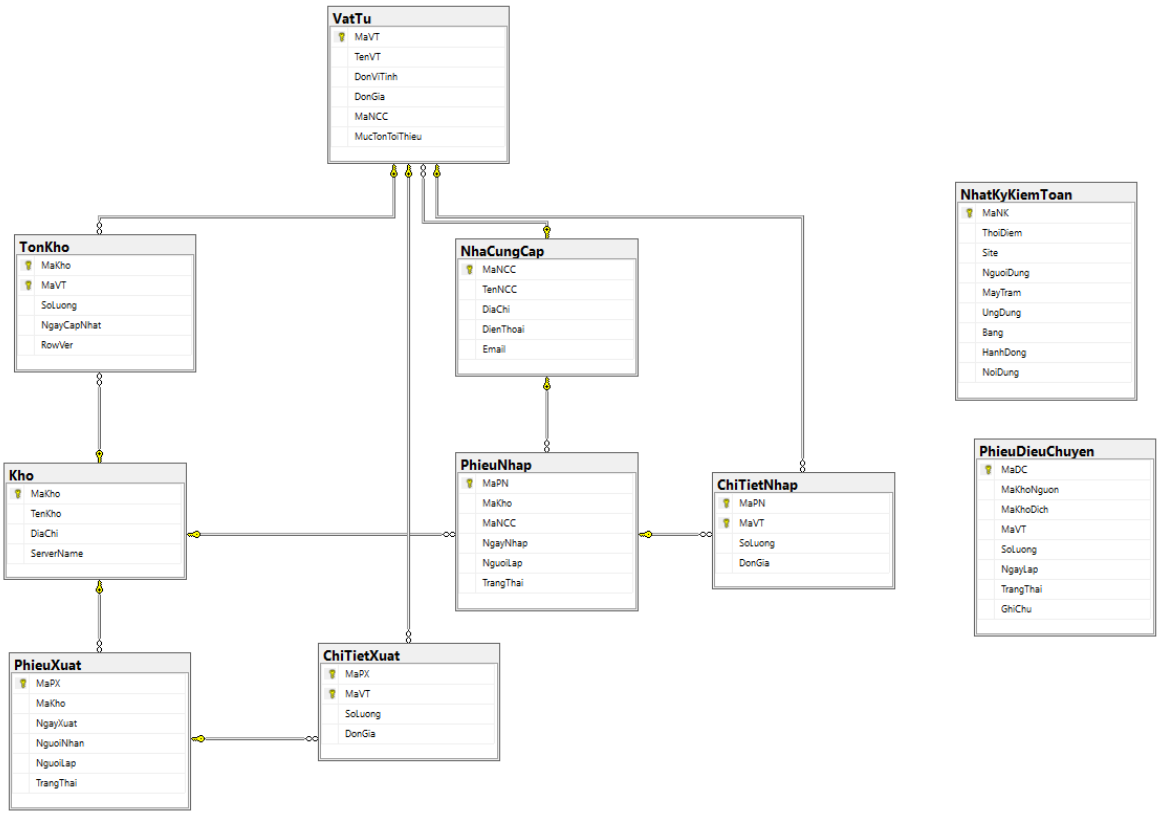
Mười ràng buộc khoá ngoại liên kết chín bảng với nhau:

Bảng II.9: Danh sách khoá ngoại

#	Tên ràng buộc	Bảng con	Cột	Trỏ tới
1	FK_VatTu_NCC	VatTu	MaNCC	NhaCungCap
2	FK_TonKho_Kho	TonKho	MaKho	Kho
3	FK_TonKho_VatTu	TonKho	MaVT	VatTu
4	FK_PN_Kho	PhieuNhap	MaKho	Kho
5	FK_PN_NCC	PhieuNhap	MaNCC	NhaCungCap
6	FK_CTN_PN	ChiTietNhap	MaPN	PhieuNhap

#	Tên ràng buộc	Bảng con	Cột	Trỏ tới
7	FK_CTN_VatTu	ChiTietNhap	MaVT	VatTu
8	FK_PX_Kho	PhieuXuat	MaKho	Kho
9	FK_CTX_PX	ChiTietXuat	MaPX	PhieuXuat
10	FK_CTX_VatTu	ChiTietXuat	MaVT	VatTu

Sơ đồ quan hệ dưới đây do SQL Server Management Studio sinh trực tiếp từ lược đồ đã cài đặt tại site KHO_A, nên nó phản ánh đúng những gì thật sự tồn tại trong cơ sở dữ liệu chứ không phải bản vẽ thiết kế trên giấy.



Hình II.1: Sơ đồ quan hệ do SSMS sinh tại site KHO_A — chín bảng nghiệp vụ và bảng nhật ký kiểm toán

Hai bảng **PhieuDieuChuyen** và **NhatKyKiemToan** đứng tách khỏi cụm quan hệ ở giữa sơ đồ. Điều này là **có chủ ý**, không phải thiếu sót thiết kế: PhieuDieuChuyen tham chiếu tới kho nguồn và kho đích nằm ở **hai site khác nhau**, mà ràng buộc khoá ngoại của SQL Server không vượt qua ranh giới máy chủ được, nên tính toàn vẹn của nó phải do giao tác phân tán và trigger bảo đảm thay cho khoá ngoại. NhatKyKiemToan là bảng nhật ký thuần tuý, cố tình không ràng buộc để một dòng nhật ký vẫn ghi lại được ngay cả khi bản ghi bị thao tác sai đã không còn tồn tại.

2.2. Lược đồ phân mảnh ngang nguyên thủy

Bảng **TonKho** được phân mảnh ngang nguyên thủy theo thuộc tính MaKho. Tập vị từ đơn giản sinh từ ba chức năng cục bộ F1, F2, F3:

```
p1 : MaKho = 'KHO_A'
p2 : MaKho = 'KHO_B'
p3 : MaKho = 'KHO_C'
```

Tập suy dẫn thoả **p1 ∨ p2 ∨ p3 ≡ TRUE** và ba vị từ đôi một loại trừ nhau. Áp dụng thuật toán COM_MIN, tập vị từ cực tiểu giữ lại là Pr' = {p1, p2}, vì p3 ⇔ ¬p1 ∧ ¬p2.

Bảng II.10: Ba mảnh ngang nguyên thủy của TonKho

Mảnh	Định nghĩa	Đặt tại	Số dòng
TonKho_A	σ (MaKho = 'KHO_A') (TonKho)	S1	10
TonKho_B	σ (MaKho = 'KHO_B') (TonKho)	S2	7
TonKho_C	σ (MaKho = 'KHO_C') (TonKho)	S3	7

Hai bảng **PhieuNhap** và **PhieuXuat** cũng được phân mảnh nguyên thủy theo cùng vị từ, vì bản thân chúng có chứa thuộc tính MaKho.

Định nghĩa mảnh được **cưỡng chế ngay tại tầng cơ sở dữ liệu** bằng ràng buộc kiểm tra, chứ không phải chỉ là quy ước:

```
CONSTRAINT CK_TonKho_Manh  CHECK (MaKho = 'KHO_A')
```

Nhờ ràng buộc này, dù người dùng có quyền quản trị cao nhất cũng không thể ghi nhầm dữ liệu của kho khác vào mảnh này.

2.3. Lược đồ phân mảnh ngang dẫn xuất

Hai bảng **ChiTietNhap** và **ChiTietXuat không chứa thuộc tính MaKho**, nên không thể phân mảnh nguyên thủy. Bắt buộc phải phân mảnh dẫn xuất theo phiếu bằng phép nửa nối:

```
PhieuNhap_i    =  σ (MaKho = 'KHO_i') (PhieuNhap)      (nguyên thủy)
ChiTietNhap_i  =  ChiTietNhap  ⋉  PhieuNhap_i          (DẪN XUẤT, nối theo MaPN)

PhieuXuat_i    =  σ (MaKho = 'KHO_i') (PhieuXuat)
ChiTietXuat_i  =  ChiTietXuat  ⋉  PhieuXuat_i          (DẪN XUẤT, nối theo MaPX)
```

Đồ thị nối: ChiTietNhap → PhieuNhap → Kho. Đây là đồ thị **dạng cây đơn giản**, mỗi quan hệ thành viên có đúng một quan hệ chủ, đủ điều kiện áp dụng phân mảnh dẫn xuất.

Tính tách rời được bảo đảm vì liên kết PhieuNhap : ChiTietNhap là **một–nhiều** — mỗi dòng chi tiết thuộc đúng một phiếu, phiếu đó lại thuộc đúng một kho.

2.4. Lược đồ nhân bản

Ba bảng danh mục **VatTu, NhaCungCap, Kho** được nhân bản thay vì phân mảnh, vì tỷ lệ đọc trên ghi rất cao (6 lượt ghi so với hàng trăm lượt đọc mỗi ngày).

Bảng II.11: Cấu hình nhân bản

Thành phần	Giá trị
Phương pháp	Transactional Replication một chiều
Publisher	MIGNON\KHO_A (Kho Trung tâm)
Distributor	MIGNON\KHO_A (gộp chung, vì hệ ba site nhỏ)
Subscriber	MIGNON\KHO_B và MIGNON\KHO_C
Kiểu đăng ký	Push Subscription — Publisher chủ động đẩy xuống
Tên Publication	PUB_DanhMuc
Số bài viết (article)	3 — VatTu, NhaCungCap, Kho
Độ trễ đồng bộ đo được	khoảng 15–20 giây

Chọn **một chiều** (Publisher sửa, Subscriber chỉ đọc) thay vì nhân bản hai chiều, vì chức năng F6 cho thấy danh mục chỉ được sửa tại Kho Trung tâm. Nhân bản một chiều đơn giản hơn nhiều và **loại bỏ hoàn toàn bài toán xung đột ghi** giữa các bản sao.

Bảng **PhieuDieuChuyen** dùng cơ chế khác: **nhân bản một phần** — một phiếu điều chuyển được ghi đồng thời ở cả kho nguồn lẫn kho đích, ngay trong cùng một giao tác phân tán, để hai bên cùng có chứng từ đối chiếu.

2.5. Lược đồ định vị

Bảng II.12: Lược đồ định vị dữ liệu trên ba site

Bảng / Mảnh	S1 KHO_A	S2 KHO_B	S3 KHO_C	Cơ chế
NhaCungCap	Gốc	Bản sao	Bản sao	Nhân bản
VatTu	Gốc	Bản sao	Bản sao	Nhân bản
Kho	Gốc	Bản sao	Bản sao	Nhân bản
TonKho_A	✓	—	—	Phân mảnh ngang
TonKho_B	—	✓	—	Phân mảnh ngang
TonKho_C	—	—	✓	Phân mảnh ngang
PhieuNhap_A + ChiTietNhap_A	✓	—	—	Nguyên thủy + dẫn xuất
PhieuNhap_B + ChiTietNhap_B	—	✓	—	Nguyên thủy + dẫn xuất
PhieuNhap_C + ChiTietNhap_C	—	—	✓	Nguyên thủy + dẫn xuất

Bảng / Mảnh	S1 KHO_A	S2 KHO_B	S3 KHO_C	Cơ chế
PhieuXuat_A + ChiTietXuat_A	✓	—	—	Nguyên thủy + dẫn xuất
PhieuXuat_B + ChiTietXuat_B	—	✓	—	Nguyên thủy + dẫn xuất
PhieuXuat_C + ChiTietXuat_C	—	—	✓	Nguyên thủy + dẫn xuất
PhieuDieuChuyen	✓	✓	✓	Ghi hai site qua giao tác phân tán

Ghi chú quan trọng: ba site dùng **cùng một tên bảng** `TonKho`, không đặt tên `TonKho_A`. Mảnh được xác định bởi **vị trí site** cộng với ràng buộc CHECK. Cách này giúp câu lệnh ứng dụng giống hệt nhau ở mọi site — đúng tinh thần **trong suốt phân mảnh**.

Căn cứ định vị — vì sao đặt mảnh nào ở đâu:

- Mảnh TonKho_i đặt tại site i, vì F1+F2+F3 chiếm 89% lượt truy cập và hoàn toàn cục bộ theo kho.
- Chứng từ đi theo kho lập ra nó, vì phiếu nhập và phiếu xuất chỉ được đọc lại bởi chính kho đó.
- Danh mục nhân bản thay vì phân mảnh, vì tỷ lệ đọc trên ghi rất cao.
- Publisher đặt ở KHO_A, vì F6 chỉ xảy ra tại Kho Trung tâm.

2.6. Lược đồ ánh xạ

Lược đồ ánh xạ mô tả đường đi từ cái nhìn của người dùng xuống tới dữ liệu vật lý, qua bốn tầng:

```
TẦNG 1 — LƯỢC ĐỒ TOÀN CỤC
   TonKho (MaKho, MaVT, SoLuong, NgayCapNhat)
   Người dùng nhìn thấy MỘT bảng duy nhất
                          |
TẦNG 2 — LƯỢC ĐỒ PHÂN MẢNH
   TonKho_A = σ(MaKho='KHO_A')(TonKho)
   TonKho_B = σ(MaKho='KHO_B')(TonKho)
   TonKho_C = σ(MaKho='KHO_C')(TonKho)
                          |
TẦNG 3 — LƯỢC ĐỒ ĐỊNH VỊ
   TonKho_A  →  S1   TonKho_B  →  S2   TonKho_C  →  S3
                          |
TẦNG 4 — LƯỢC ĐỒ ÁNH XẠ ĐỊA PHƯƠNG
   S1: KhoA.dbo.TonKho     S2: KhoB.dbo.TonKho     S3: KhoC.dbo.TonKho
```

Trong hệ thống, **khung nhìn v_TonKho_ToanHeThong** chính là hiện thực hoá của tầng 1 và tầng 2: nó thực hiện phép hợp ba mảnh, nhờ đó người dùng viết truy vấn như đang làm việc với một bảng duy nhất.

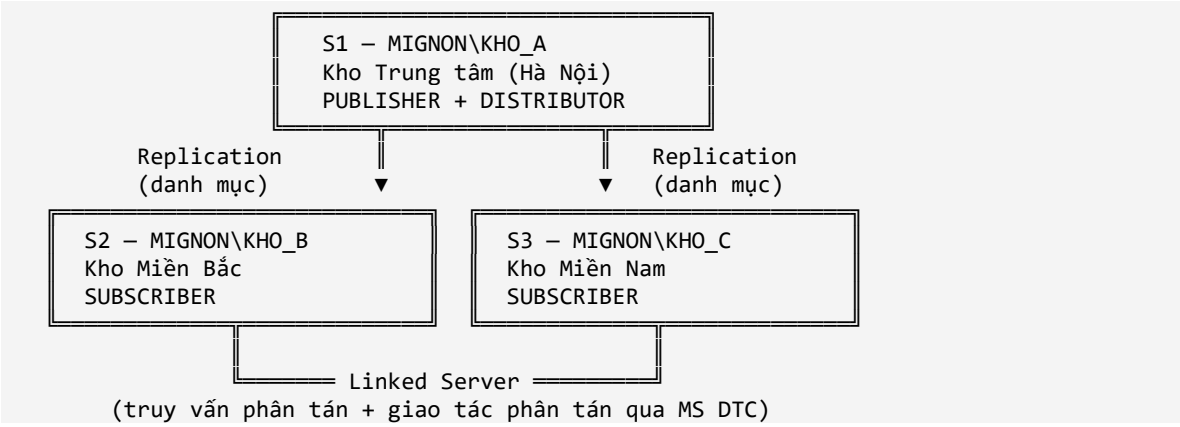
Thủ tục sp_TonKhoToanHeThong hiện thực hoá tầng 3: nó đọc tên máy chủ từ **cột ServerName của bảng Kho** rồi mới dựng câu lệnh, nên thêm một kho mới chỉ cần thêm một dòng vào bảng Kho, **không phải sửa một dòng mã nguồn nào**.

2.7. Thiết kế kiến trúc hệ thống

Hệ thống dùng kiến trúc **lai giữa Ngang hàng và Client/Server**, tuỳ theo loại dữ liệu:

- **Ngang hàng (peer-to-peer)** với dữ liệu **tồn kho và chứng từ**. Mỗi kho toàn quyền trên mảnh dữ liệu của mình, không kho nào phụ thuộc kho khác để nhập xuất hằng ngày. Ba site nối với nhau bằng **Linked Server hai chiều**, site nào cũng đọc được dữ liệu của site kia.
- **Client/Server** với dữ liệu **danh mục**. Kho Trung tâm là Publisher duy nhất được sửa, hai kho còn lại là Subscriber chỉ đọc. Cách này bảo đảm danh mục vật tư toàn hệ thống luôn thống nhất.

Sơ đồ kiến trúc tổng thể:



Đường đồng bộ hoá trong hệ thống có hai loại, phục vụ hai mục đích khác nhau:

Bảng II.13: Hai đường đồng bộ hoá

Đường	Cơ chế	Dữ liệu	Tính chất
Nhân bản danh mục	Transactional Replication	VatTu, NhaCungCap, Kho	Một chiều, bất đồng bộ, độ trễ 15–20 giây
Điều chuyển vật tư	Giao tác phân tán 2PC qua MS DTC	TonKho, PhieuDieuChuyen	Hai chiều, đồng bộ tức thời, nguyên tố tuyệt đối

Sự khác biệt này rất quan trọng: danh mục chấp nhận độ trễ vài giây vì giá vật tư đổi chậm; còn tồn kho thì **không được phép có độ trễ**, vì một giây sai lệch là có thể bán mất hàng không có thật.

2.8. Mô hình hệ thống tại chi nhánh và toàn hệ thống

Mô hình tại một chi nhánh:

```
┌──────────────── MỘT CHI NHÁNH KHO ────────────────┐
│ FRONT END                  BACK END               │
│                                                   │
│ ┌──────────────────┐  EXEC  ┌──────────────────┐  │
│ │ Máy nhân viên    │ ──────►│ SQL Server instance│ │
│ │  · lập phiếu     │        │  · 9 bảng        │ │
│ │  · tra tồn       │◄──────│  · 5 thủ tục     │ │
│ │  · in chứng từ   │ K.quả  │  · 6 trigger     │ │
│ └──────────────────┘        │  · 4 vai trò     │ │
│                             │  · SQL Agent     │ │
│ ┌──────────────────┐        │  · MS DTC        │ │
│ │ Máy trưởng kho   │ ──────►└──────────────────┘ │
│ └──────────────────┘                              │
└──────────────────────────┬────────────────────────┘
                           │
                ra mạng WAN / VPN ZeroTier
```

Mô hình toàn hệ thống:

```
TẦNG NGƯỜI DÙNG (FRONT END)

┌──────────┐  ┌──────────┐  ┌──────────┐  ┌──────────┐
│ NV kho A │  │ NV kho B │  │ NV kho C │  │ Ban giám đốc │
│ Trưởng A │  │ Trưởng B │  │ Trưởng C │  │ QT danh mục  │
└──────────┘  └──────────┘  └──────────┘  └──────────┘
═══╪═══════════╪═══════════╪═══════════╪═══
          MẠNG RIÊNG ẢO — ZeroTier (10.91.x.x)
═══╪═══════════╪═══════════╪═══════════╪═══
TẦNG CSDL (BACK END)                      │
   ▼           ▼           ▼              │
┌──────────┐ ┌──────────┐ ┌──────────┐   │
│KHO_A :1440│ │KHO_B :1441│ │KHO_C :1442│◄──┘
│Publisher │ │Subscriber│ │Subscriber│
└──────────┘ └──────────┘ └──────────┘
      └──── Linked Server + MS DTC ────┘
```

CHƯƠNG III. TRIỂN KHAI HỆ THỐNG

Chương này trình bày quá trình cài đặt vật lý thực tế của hệ thống, kèm ảnh chụp màn hình minh chứng cho từng bước.

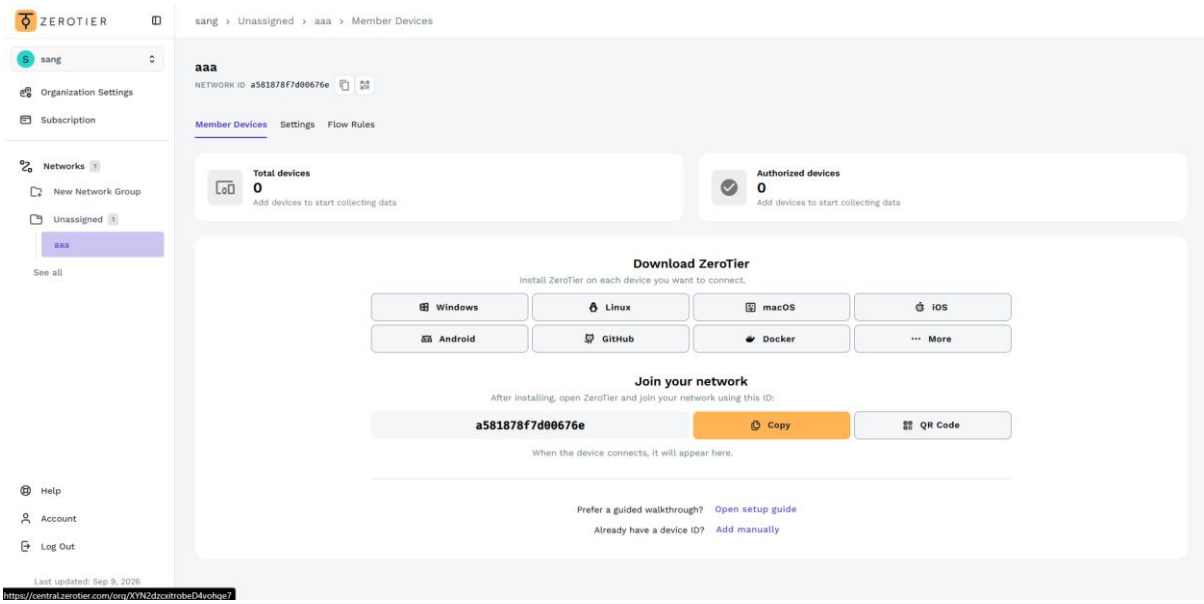
1. Cài đặt mạng riêng ảo

Ba kho trong bài toán nằm ở ba tỉnh thành khác nhau, nên phải có một mạng riêng ảo để ba máy chủ thấy nhau như trong cùng một mạng nội bộ. Nhóm chọn **ZeroTier** vì miễn phí, cài đặt đơn giản và tạo được mạng LAN ảo xuyên Internet.

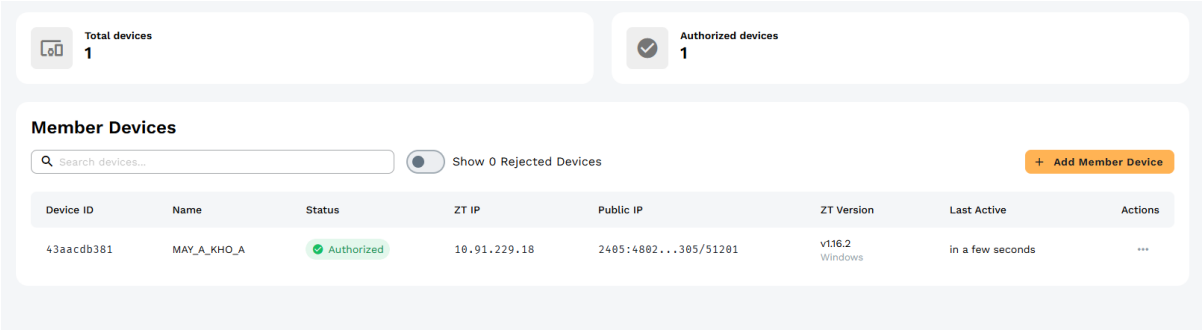
Các bước thực hiện:

1. Tải và cài ZeroTier trên cả ba máy.
2. Tạo một network riêng cho nhóm, thu được **Network ID** gồm 16 ký tự.
3. Mỗi máy tham gia network bằng Network ID đó.
4. Người quản trị network duyệt (Authorize) từng máy trên trang quản lý.
5. Mỗi máy được cấp một địa chỉ IP ảo riêng — network của nhóm được cấp dải **10.91.x.x**.

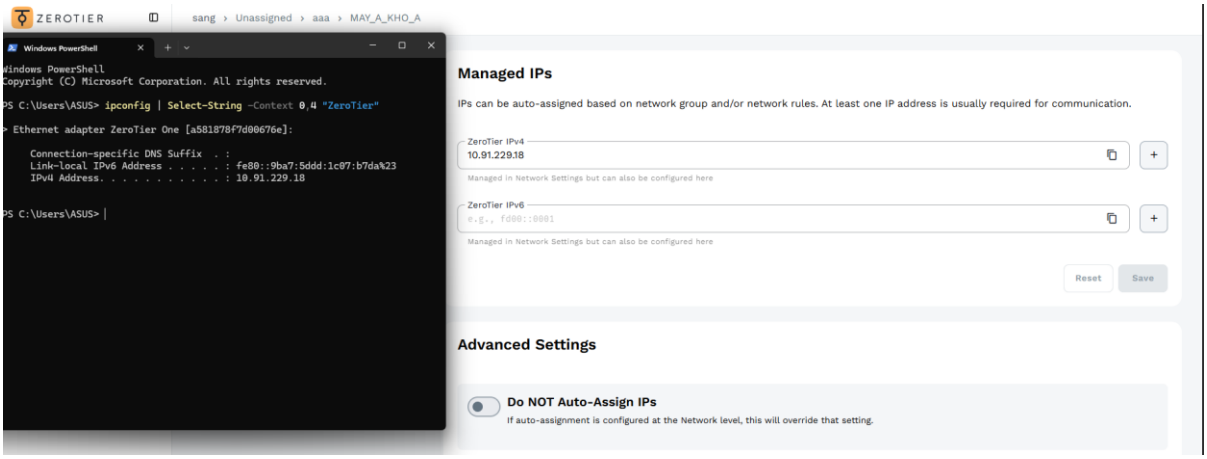
Sau bước này, ba máy chủ có thể gọi nhau bằng địa chỉ IP ảo, bất kể chúng đang ở mạng vật lý nào.



Hình III.1: Network riêng của nhóm vừa được tạo, kèm Network ID mười sáu ký tự dùng để mời hai máy còn lại



Hình III.2: Danh sách thành viên của network. Mỗi máy chỉ vào được sau khi người quản trị tích ô Auth



Hình III.3: Địa chỉ IP ảo mà ZeroTier cấp cho máy chủ, đối chiếu giữa trang quản lý network và lệnh ipconfig trên chính máy đó

2. Tạo đường liên kết mạng giữa các máy chủ

Sau khi có mạng ảo, phải mở đường cho SQL Server nghe trên mạng. Mặc định sau khi cài, ba thể hiện SQL Server chỉ giao tiếp qua **Shared Memory** — bộ nhớ chung trong cùng một máy — nên máy khác không thể kết nối tới.

Ba việc cần làm trên mỗi máy chủ:

1. **Bật giao thức TCP/IP** cho từng thể hiện SQL Server.
2. **Gán cổng cố định** cho từng thể hiện. Không dùng cổng động vì cổng động thay đổi sau mỗi lần khởi động, không thể mở tường lửa và máy khác không biết đường nào mà nối.
3. **Mở tường lửa** cho các cổng đó, cho cổng UDP 1434 của SQL Server Browser, và cho dịch vụ **MS DTC** — thiếu MS DTC thì giao tác phân tán giữa hai máy không chạy được.

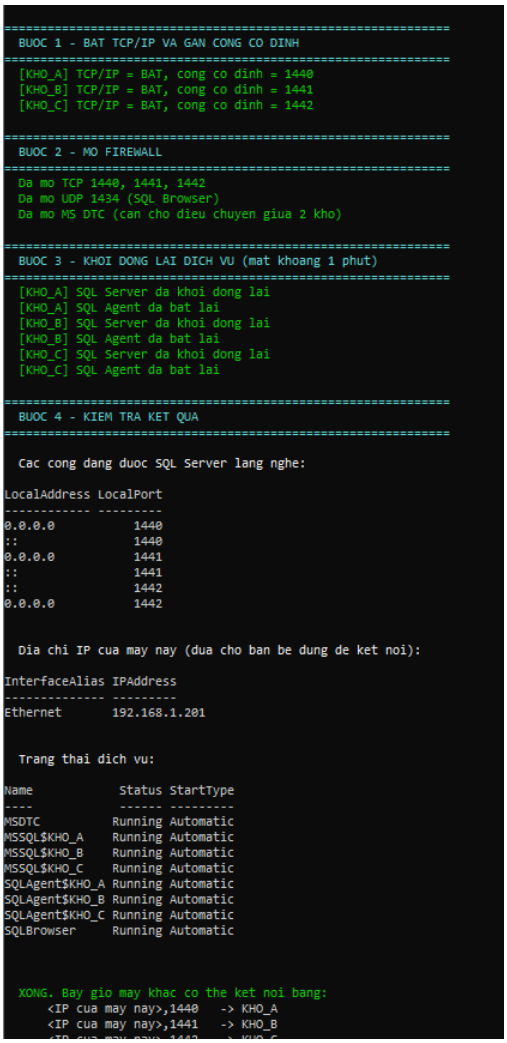
Bảng III.1: Cổng TCP được gán cho ba thể hiện

Site	Thể hiện	Cổng TCP	Vai trò
S1	MIGNON\KHO_A	1440	Kho Trung tâm — Publisher + Distributor

Site	Thể hiện	Cổng TCP	Vai trò
S2	MIGNON\KHO_B	1441	Kho Miền Bắc — Subscriber
S3	MIGNON\KHO_C	1442	Kho Miền Nam — Subscriber

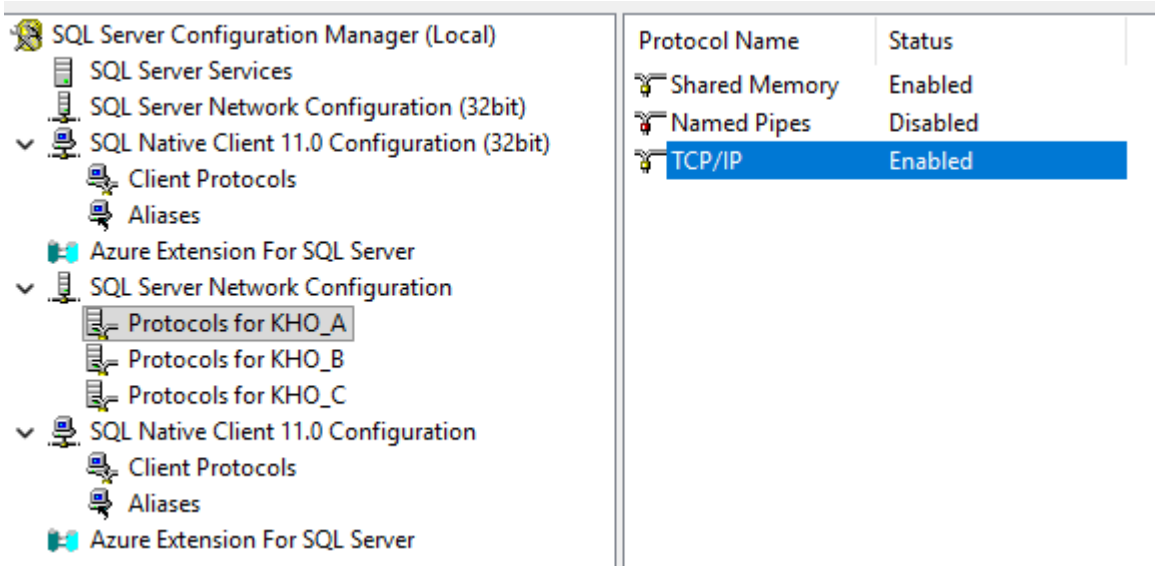
Việc kiểm chứng đường truyền được thực hiện bằng khung nhìn hệ thống `sys.dm_exec_connections`. Khi kết nối theo tên thể hiện trong cùng một máy, cột `net_transport` cho giá trị **Shared memory**; khi kết nối qua địa chỉ IP ảo, cột này cho giá trị **TCP** kèm đúng địa chỉ ảo — đó là bằng chứng bằng số liệu rằng dữ liệu thật sự đi qua mạng.

Ba việc trên được gộp vào một tập lệnh PowerShell chạy một lần với quyền quản trị, nhờ đó lặp lại được y hệt trên máy của từng thành viên mà không sợ sót bước hay gõ nhầm cổng.

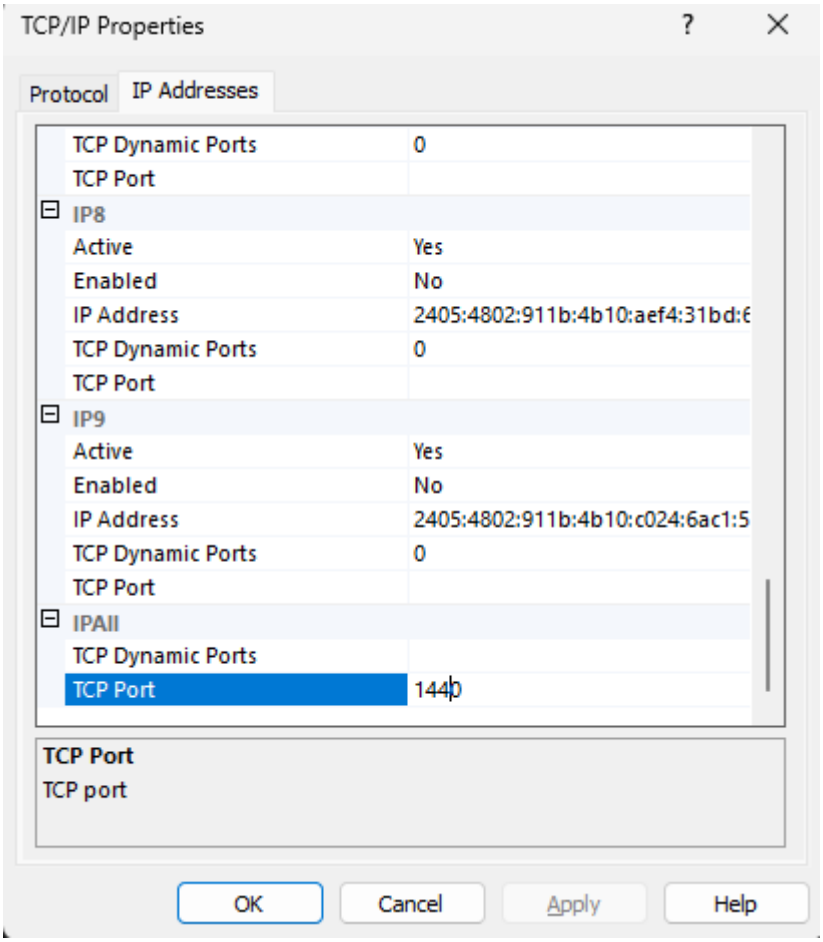


```
==============================================================
  BUOC 1 - BAT TCP/IP VA GAN CONG CO DINH
==============================================================
  [KHO_A] TCP/IP = BAT, cong co dinh = 1440
  [KHO_B] TCP/IP = BAT, cong co dinh = 1441
  [KHO_C] TCP/IP = BAT, cong co dinh = 1442

==============================================================
  BUOC 2 - MO FIREWALL
==============================================================
  Da mo TCP 1440, 1441, 1442
  Da mo UDP 1434 (SQL Browser)
  Da mo MS DTC (can cho dieu chuyen giua 2 kho)

==============================================================
  BUOC 3 - KHOI DONG LAI DICH VU (mat khoang 1 phut)
==============================================================
  [KHO_A] SQL Server da khoi dong lai
  [KHO_A] SQL Agent da bat lai
  [KHO_B] SQL Server da khoi dong lai
  [KHO_B] SQL Agent da bat lai
  [KHO_C] SQL Server da khoi dong lai
  [KHO_C] SQL Agent da bat lai

==============================================================
  BUOC 4 - KIEM TRA KET QUA
==============================================================

  Cac cong dang duoc SQL Server lang nghe:

LocalAddress LocalPort
------------ ---------
0.0.0.0           1440
::                1440
0.0.0.0           1441
::                1441
::                1442
0.0.0.0           1442


  Dia chi IP cua may nay (dua cho ban be dung de ket noi):

InterfaceAlias IPAddress
-------------- ---------
Ethernet       192.168.1.201


  Trang thai dich vu:

Name            Status StartType
----            ------ ---------
MSDTC           Running Automatic
MSSQL$KHO_A     Running Automatic
MSSQL$KHO_B     Running Automatic
MSSQL$KHO_C     Running Automatic
SQLAgent$KHO_A  Running Automatic
SQLAgent$KHO_B  Running Automatic
SQLAgent$KHO_C  Running Automatic
SQLBrowser      Running Automatic



  XONG. Bay gio may khac co the ket noi bang:
      <IP cua may nay>,1440   -> KHO_A
      <IP cua may nay>,1441   -> KHO_B
```

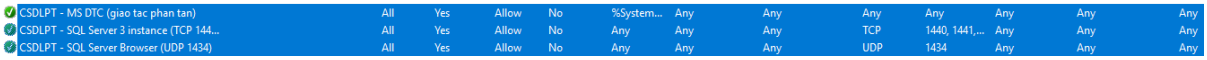
Hình III.4: Kết quả tập lệnh cấu hình mạng: ba cổng đã được lắng nghe, bảy dịch vụ đều ở trạng thái Running và Automatic



Hình III.5: Giao thức TCP/IP được bật cho thể hiện SQL Server trong SQL Server Configuration Manager



Hình III.6: Gán cổng cố định tại mục IPAll, đồng thời xoá trắng ô cổng động



Hình III.7: Các luật tường lửa mở cho ba cổng thể hiện, cổng UDP 1434 của SQL Server Browser và dịch vụ MS DTC

Results | Messages

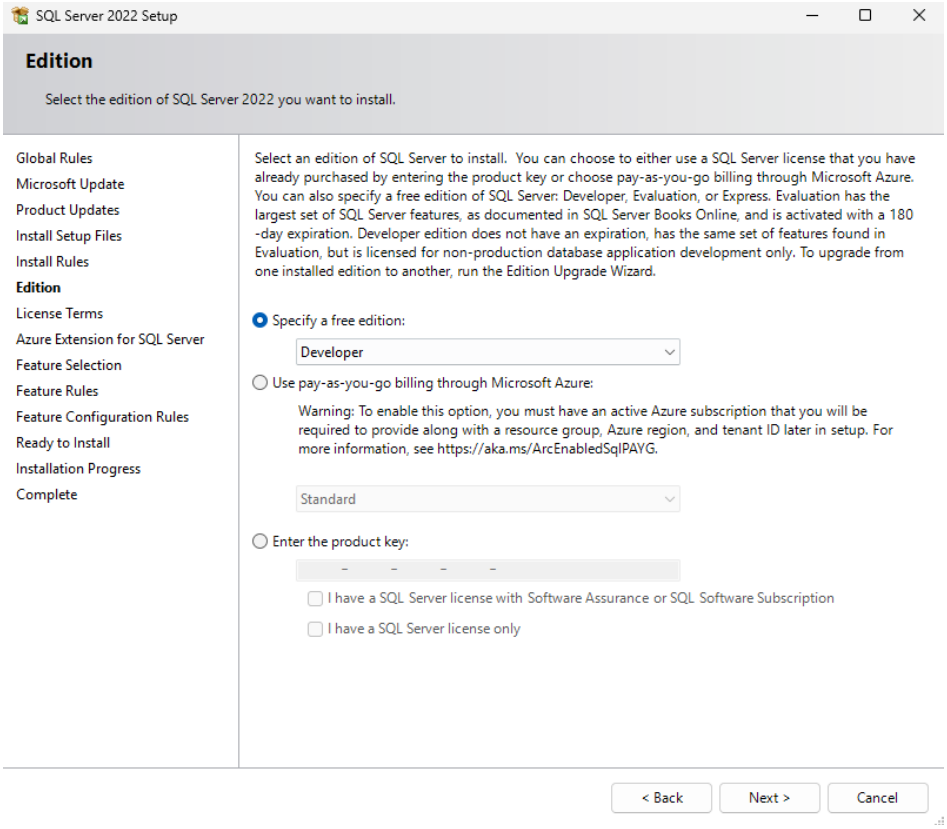
	MayChuDangKetNoi	GiaoThucKetNoi	DiaChi_MayChu	Cong	DiaChi_MayKhach
1	Mignon\KHO_A	TCP	192.168.1.201	1440	192.168.1.201

Hình III.8: Kết nối qua địa chỉ IP ảo cho net_transport bằng TCP — dữ liệu thật sự đi qua mạng chứ không qua bộ nhớ chung

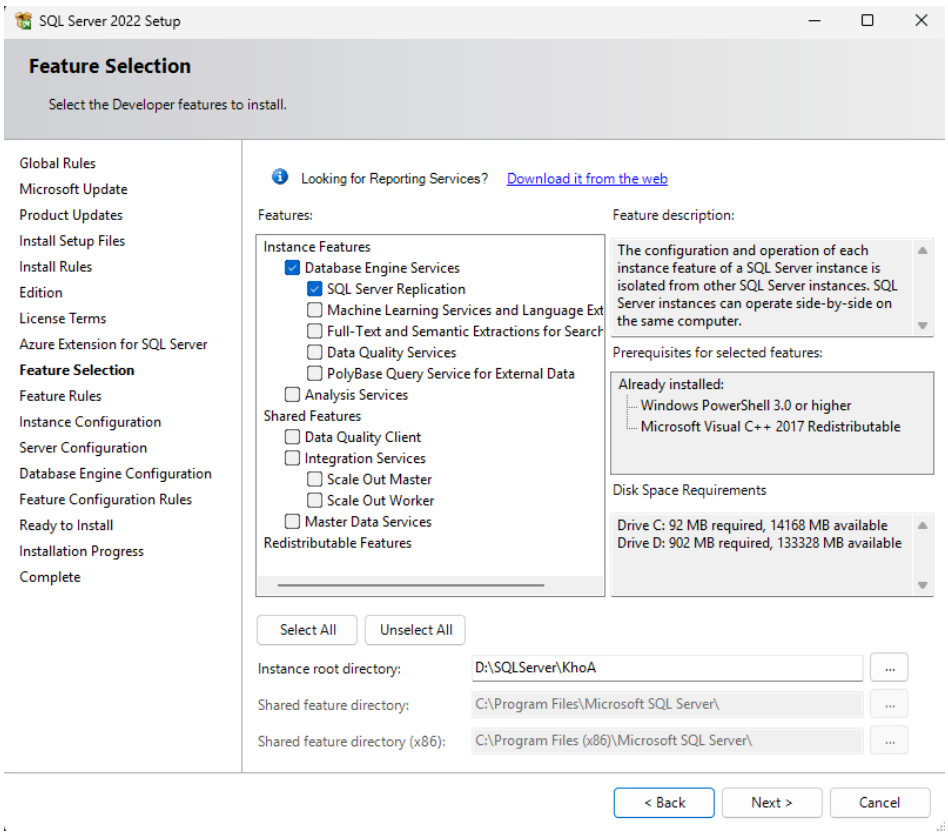
3. Cài đặt SQL Server

Nhóm cài **SQL Server 2022 Developer Edition** — bản miễn phí có đầy đủ tính năng của bản Enterprise, khác với bản Express vốn **không hỗ trợ Replication làm Publisher**, tức là không dùng được cho đồ án này.

Quá trình cài đặt được lặp lại **ba lần** để tạo ba thể hiện độc lập trên cùng một máy. Mỗi thể hiện có dịch vụ Windows riêng, SQL Server Agent riêng, thư mục dữ liệu riêng và cổng mạng riêng.

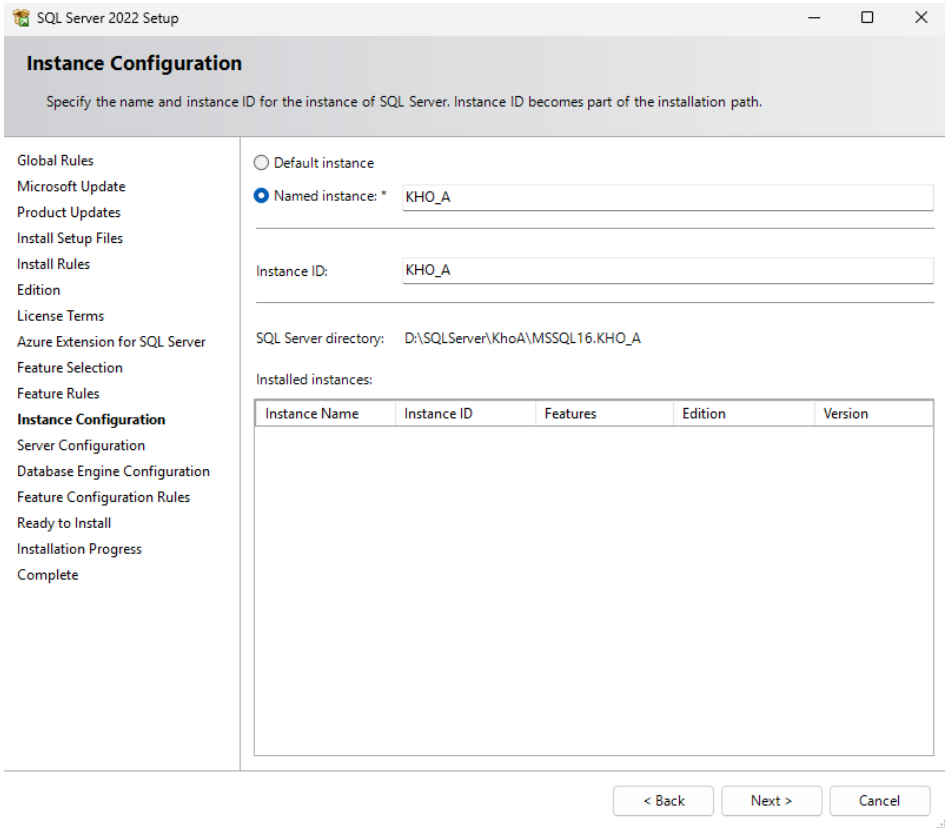


Hình III.9: Chọn phiên bản Developer — miễn phí và đầy đủ tính năng

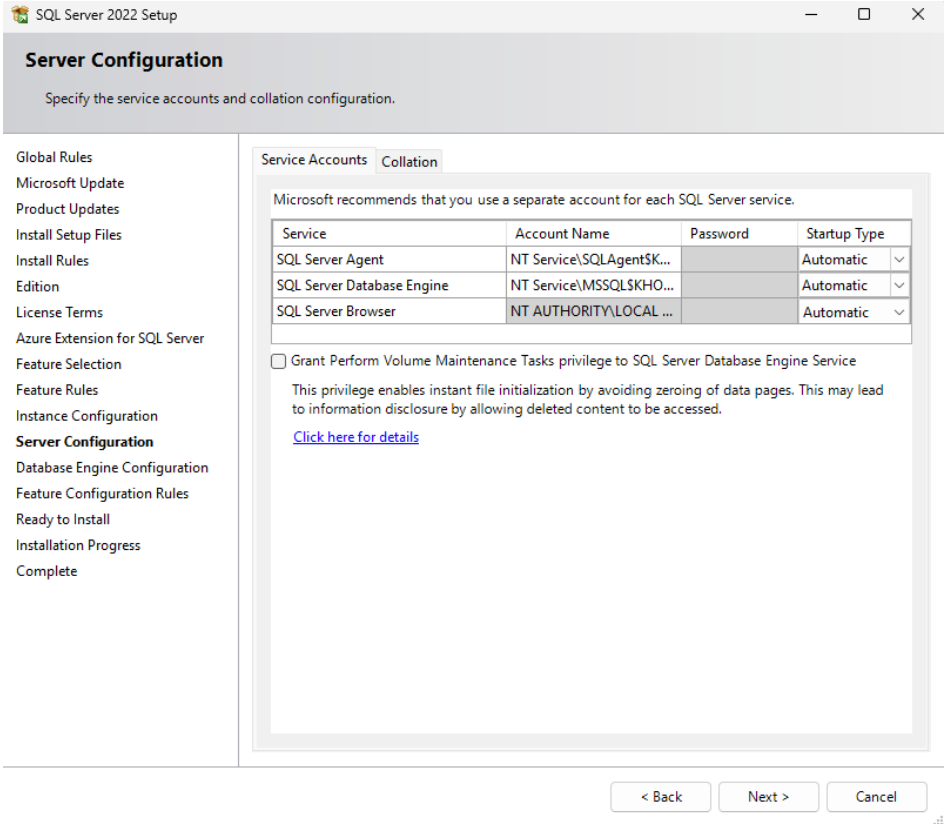


Hình III.10: Chọn tính năng cài đặt. Bắt buộc tích SQL Server Replication, thiếu mục này là không làm được nhân bản

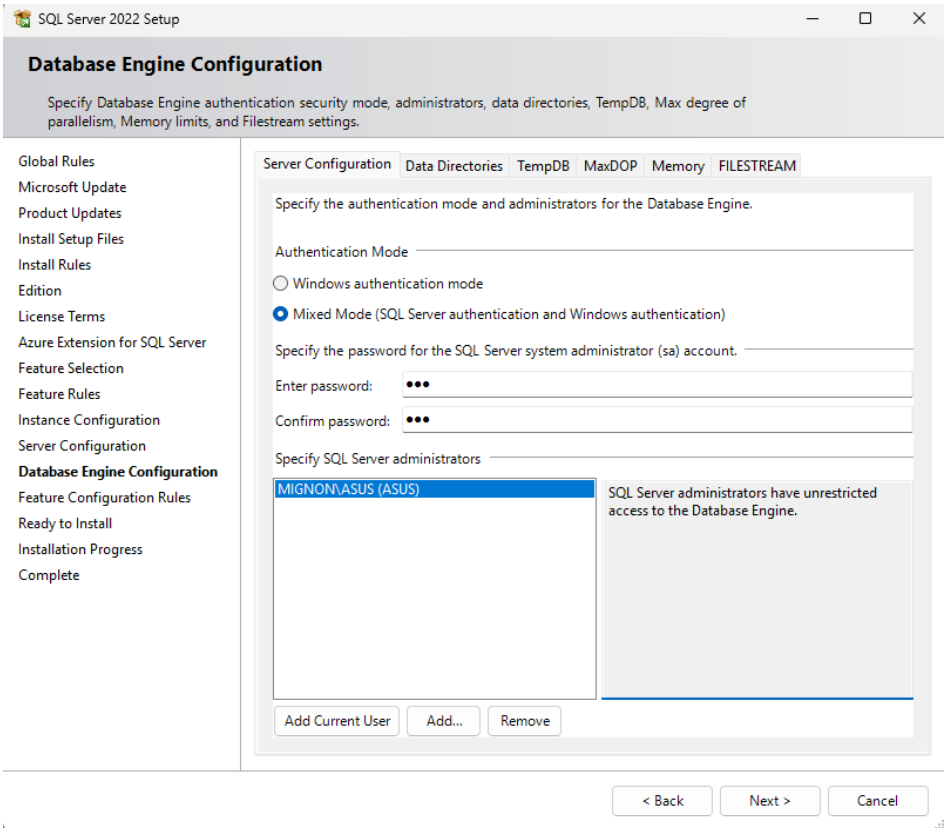
Tại màn hình này nhóm cũng đổi **Instance root directory** sang ổ D để tách dữ liệu khỏi ổ hệ thống.



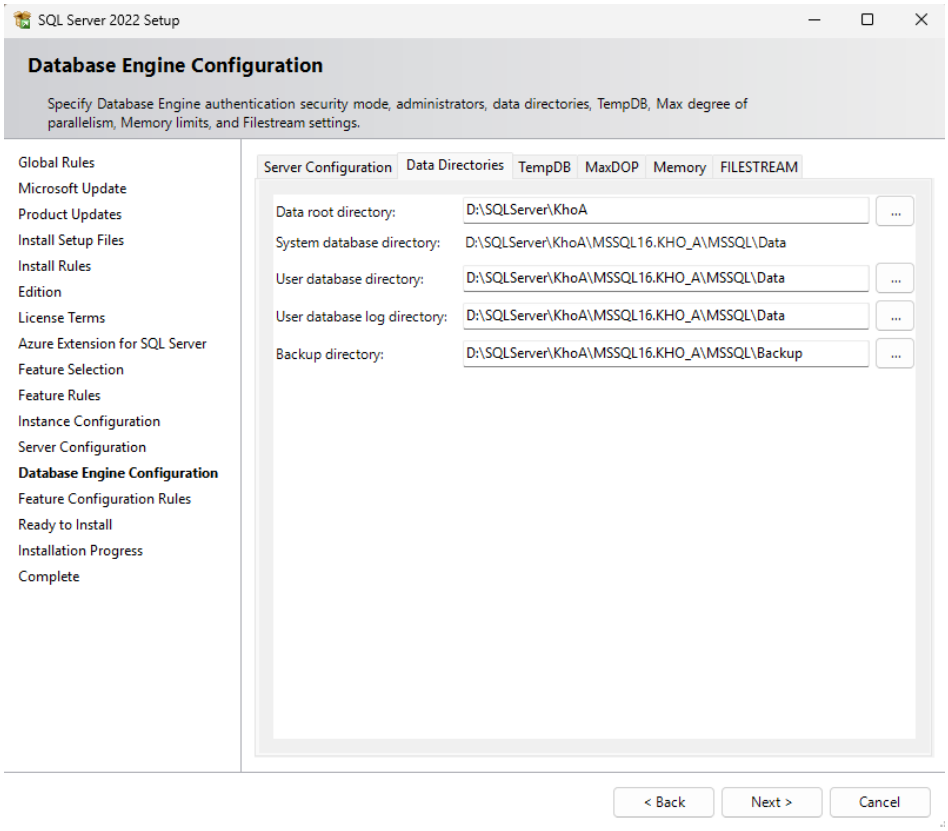
Hình III.11: Đặt tên thể hiện là KHO_A. Đây là bước quyết định ba site độc lập với nhau



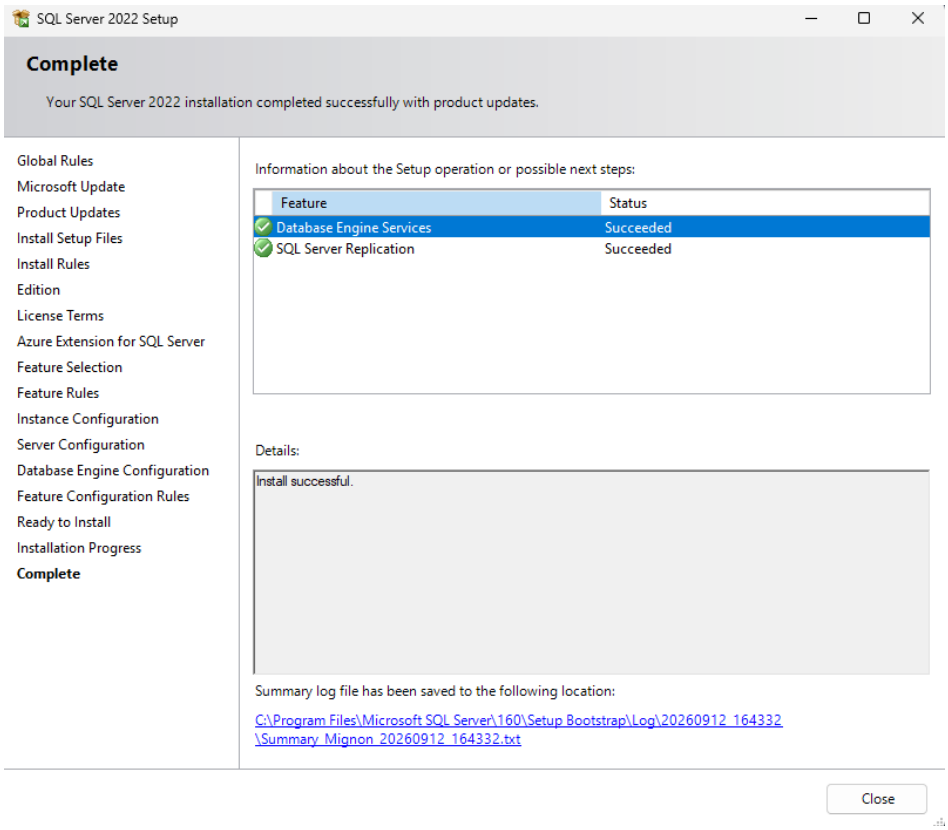
Hình III.12: Đặt SQL Server Agent ở chế độ Automatic — điều kiện bắt buộc để các tiến trình nhân bản chạy được



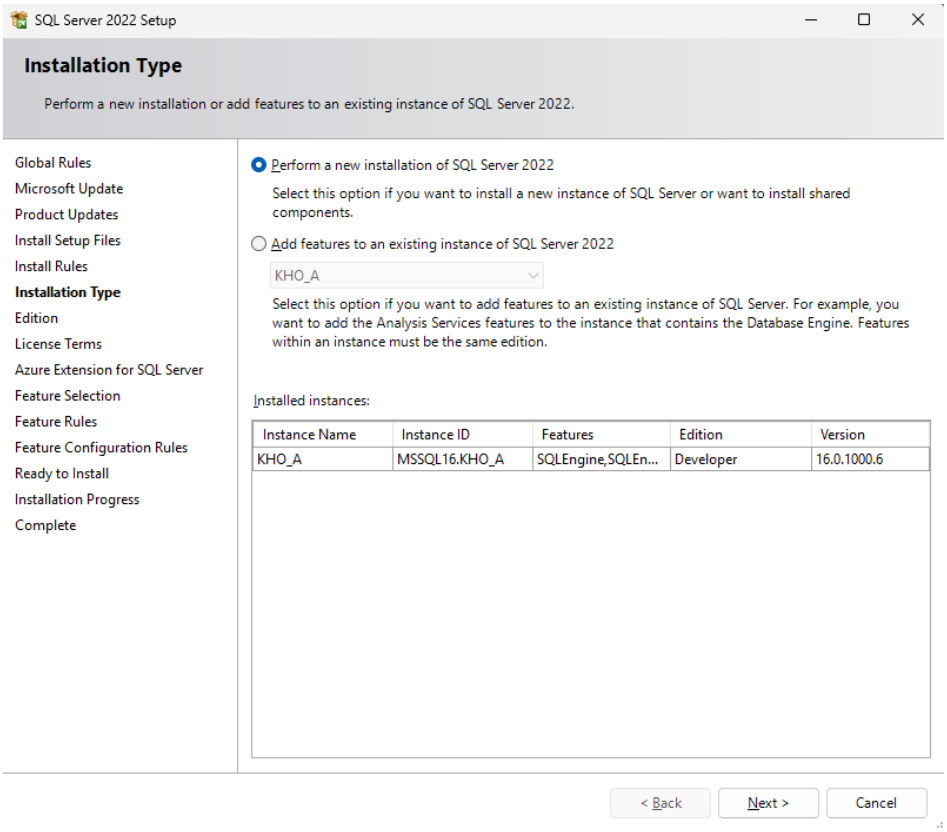
Hình III.13: Chọn Mixed Mode để dùng được tài khoản sa. Linked Server giữa các site sẽ đăng nhập bằng tài khoản này



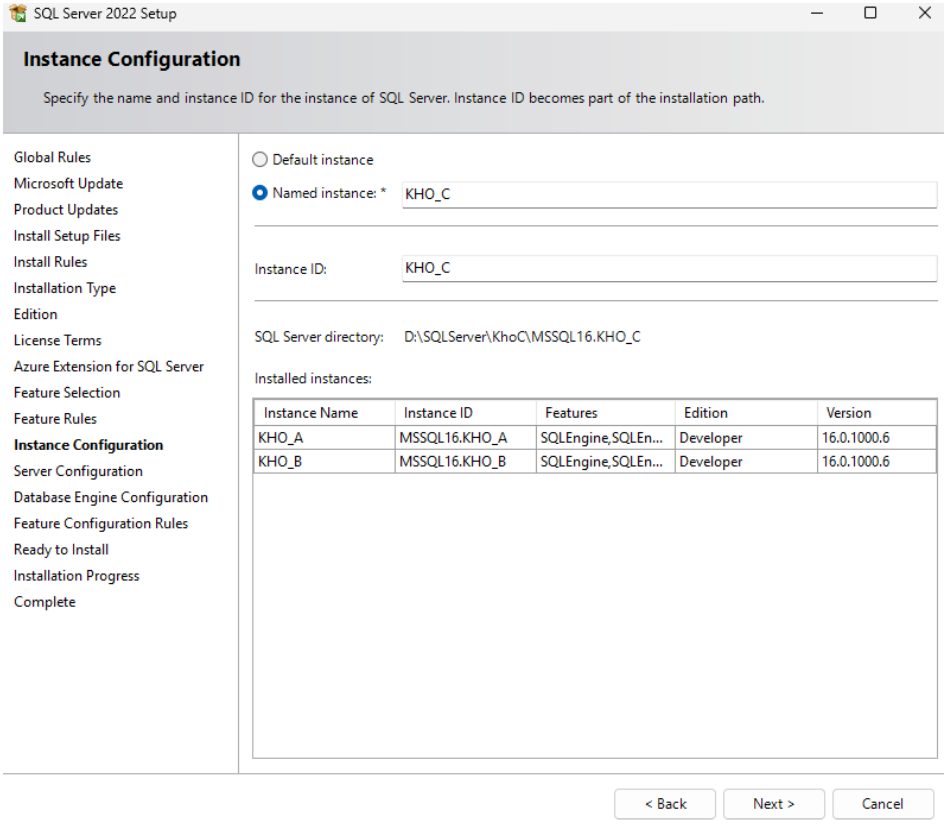
Hình III.14: Thư mục dữ liệu riêng của từng thể hiện



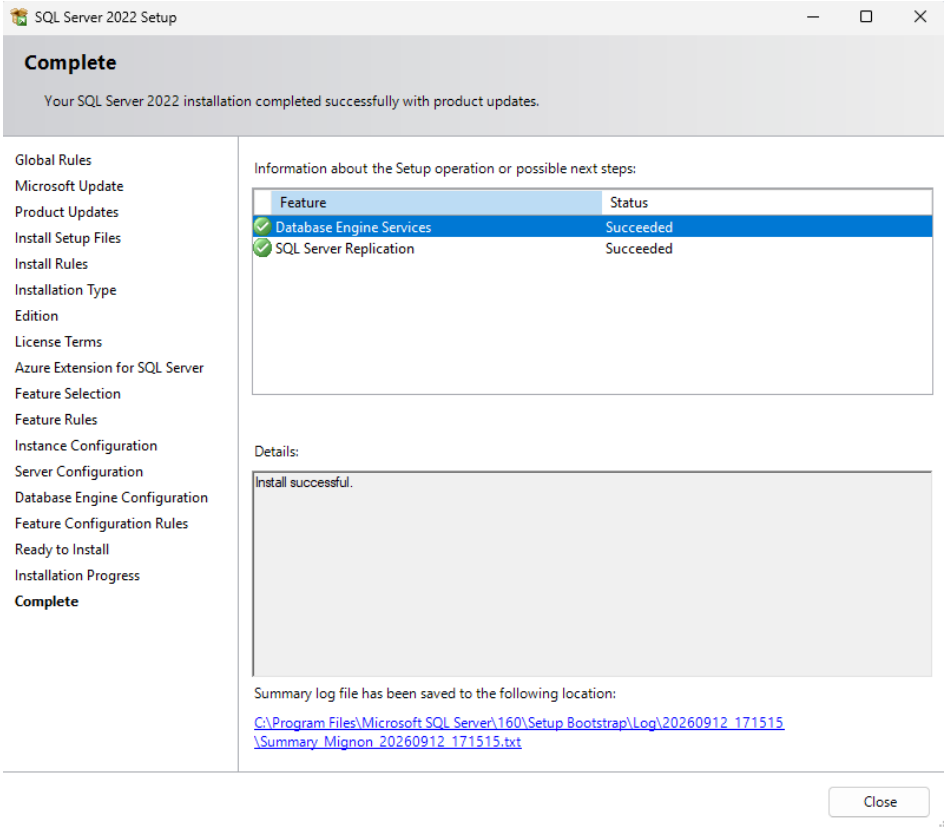
Hình III.15: Cài đặt thể hiện KHO_A hoàn tất, cả Database Engine và Replication đều Succeeded



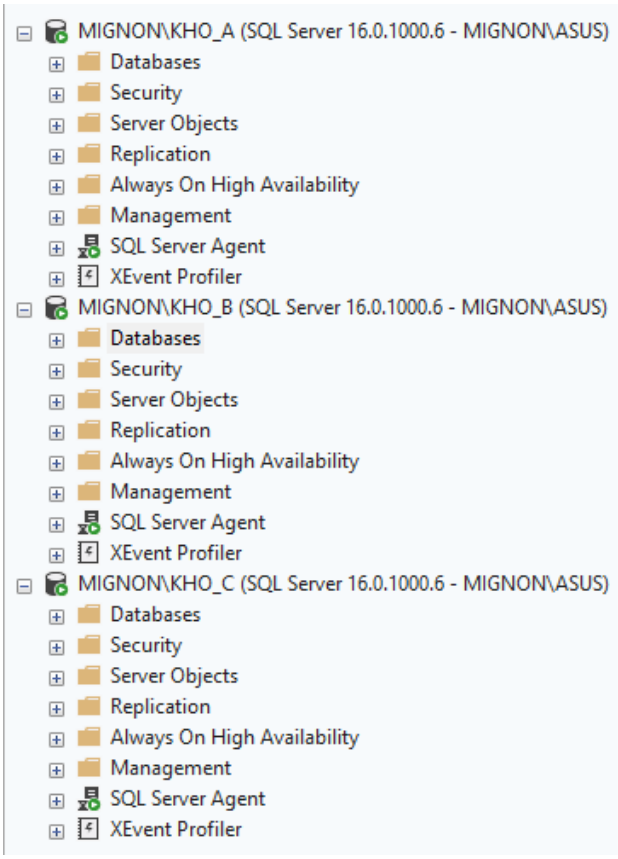
Hình III.16: Cài thể hiện thứ hai. Bảng Installed instances đã có KHO_A, chọn Perform a new installation để tạo thêm thể hiện mới



Hình III.17: Cài thể hiện thứ ba KHO_C. Bảng dưới cho thấy KHO_A và KHO_B đã tồn tại — ba thể hiện chạy song song trên cùng một máy



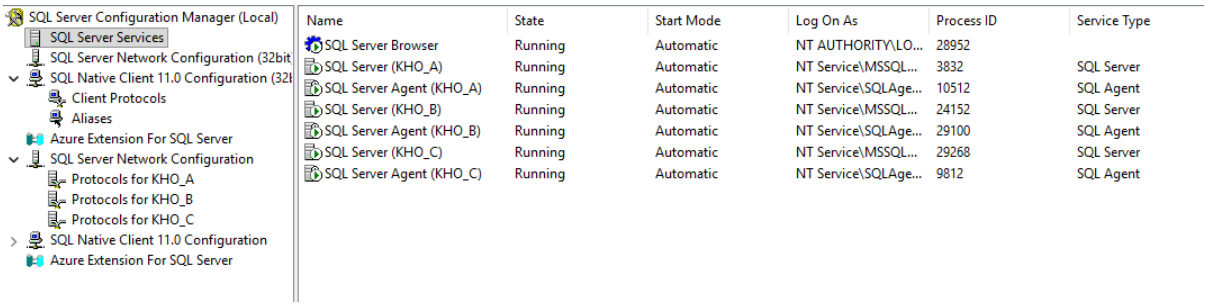
Hình III.18: Hoàn tất cài đặt cả ba thể hiện



Hình III.19: Ba thể hiện hiện diện cùng lúc trong Object Explorer của SSMS — hình ảnh đại diện cho toàn bộ hệ thống phân tán

4. Kiểm tra dịch vụ Agent

SQL Server Agent là tiến trình nền chạy các công việc theo lịch. Trong đồ án này nó **bắt buộc phải chạy**, vì hai tiến trình của nhân bản — Log Reader Agent và Distribution Agent — đều là công việc do Agent điều khiển. Agent dừng thì dữ liệu danh mục sẽ không được đẩy xuống hai kho vệ tinh.



Hình III.20: Bảy dịch vụ của hệ thống đều ở trạng thái Running và Automatic

Bảy dịch vụ trong ảnh gồm:

- **Ba dịch vụ SQL Server** — ba máy chủ cơ sở dữ liệu của ba kho.
- **Ba dịch vụ SQL Server Agent** — điều kiện để nhân bản hoạt động.
- **SQL Server Browser** — giúp máy khác tìm được thể hiện theo tên thay vì phải nhớ số cổng.

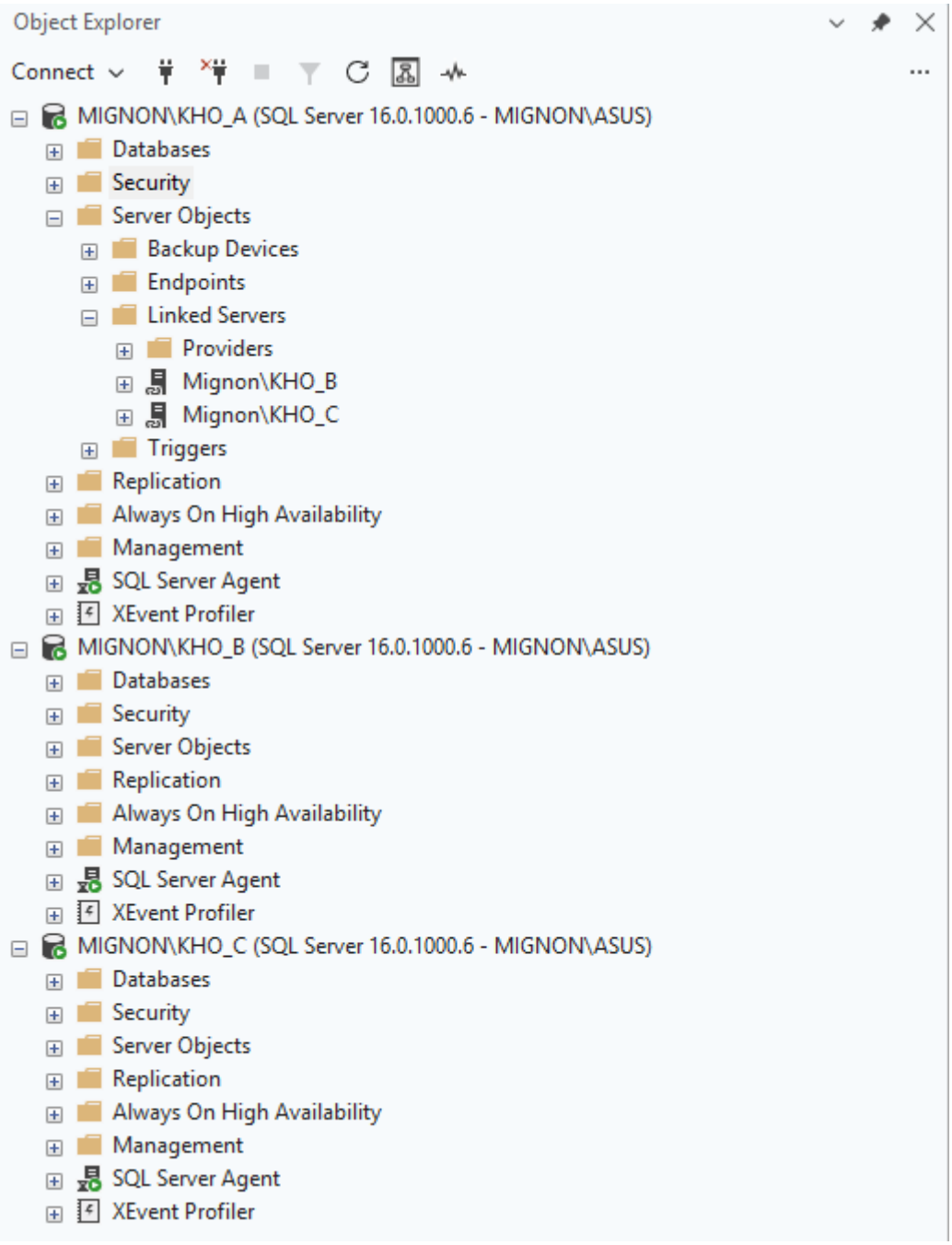
Cả bảy đều đặt **Start Mode = Automatic**, nghĩa là tự khởi động cùng máy, không cần người dùng bật tay.

5. Tạo liên kết cơ sở dữ liệu giữa các máy chủ

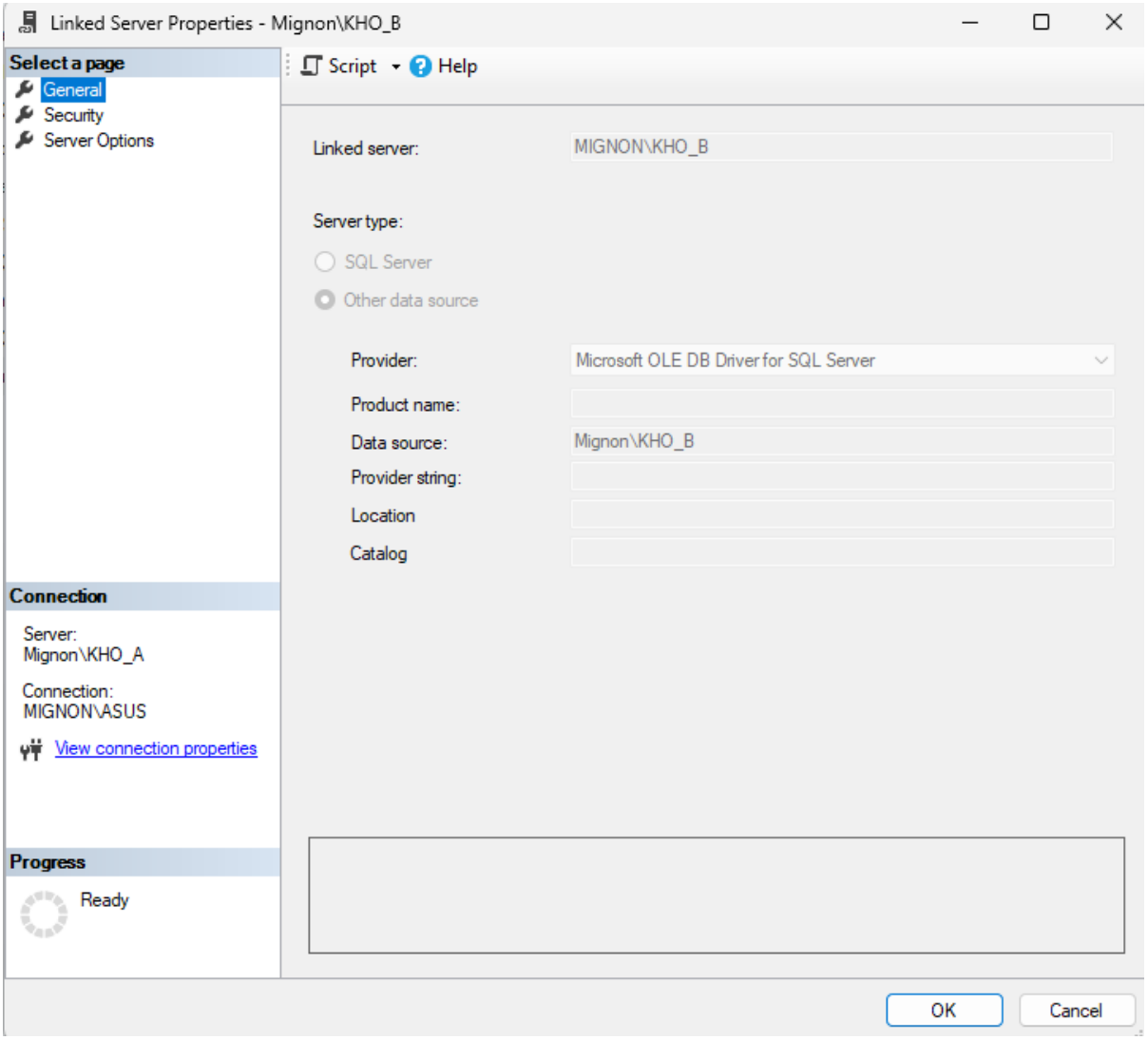
Linked Server là cơ chế cho phép một thể hiện SQL Server truy vấn dữ liệu nằm trên thể hiện khác như thể đó là bảng của chính mình. Đây là nền tảng kỹ thuật của **truy vấn phân tán** và **giao tác phân tán**.

Hệ thống tạo **sáu liên kết** — mỗi site nối tới hai site còn lại, tạo thành đồ thị hai chiều đầy đủ:

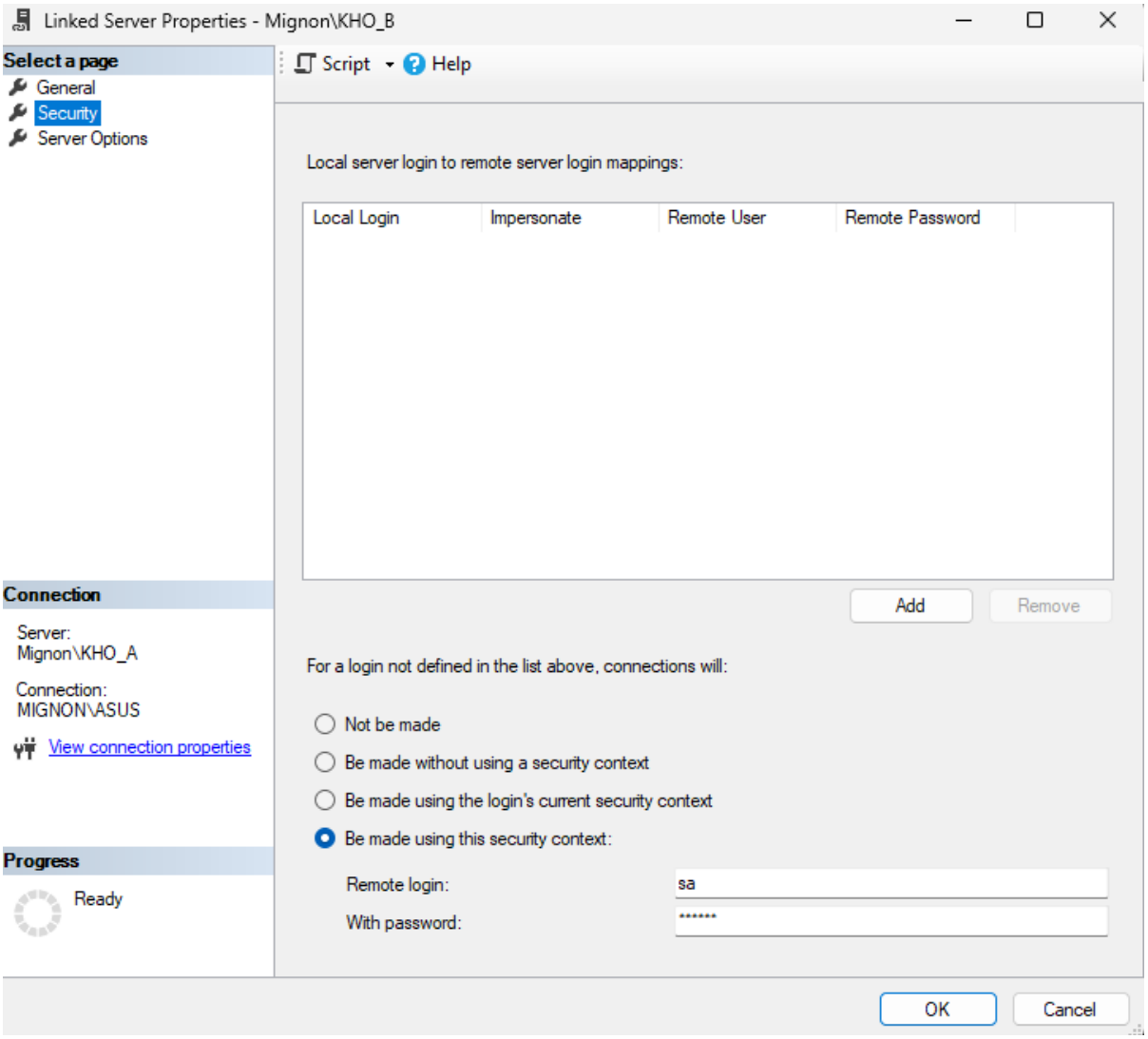
```
KHO_A ──► KHO_B        KHO_B ──► KHO_A        KHO_C ──► KHO_A
KHO_A ──► KHO_C        KHO_B ──► KHO_C        KHO_C ──► KHO_B
```



Hình III.21: Hai Linked Server tại site KHO_A trong Object Explorer



Hình III.22: Trang General — khai báo nguồn dữ liệu của liên kết



Hình III.23: Trang Security — ánh xạ đăng nhập sang tài khoản cố định ở máy đích

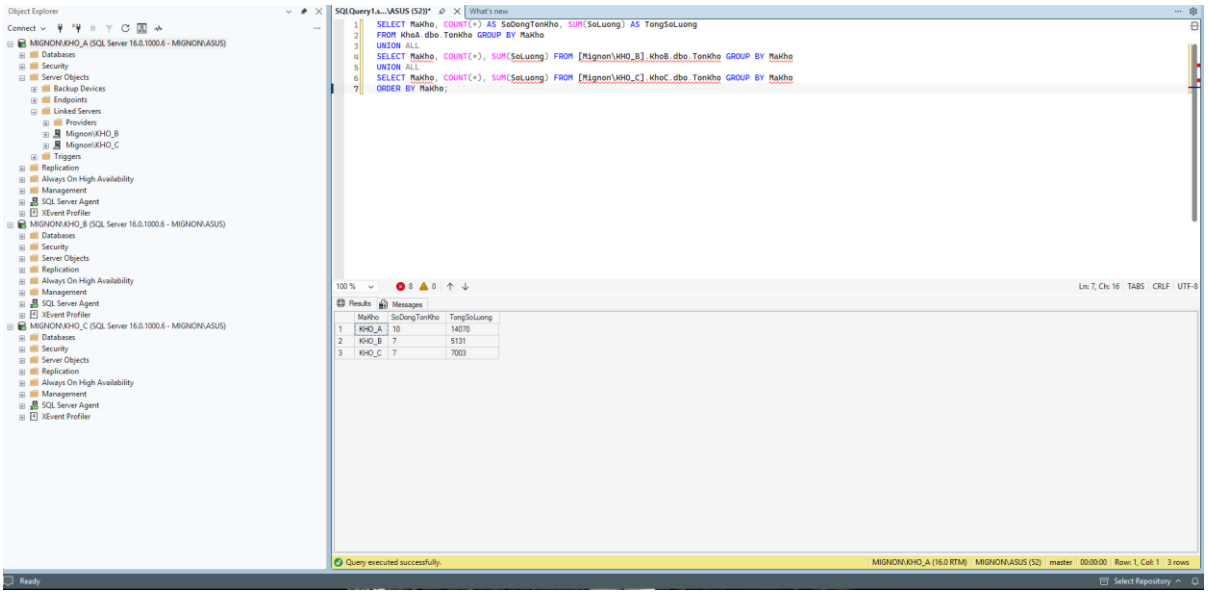
Bốn tuỳ chọn bắt buộc phải bật cho mỗi liên kết, thiếu cái nào hỏng chức năng tương ứng:

Bảng III.2: Bốn tuỳ chọn của Linked Server

Tuỳ chọn	Tác dụng	Thiếu thì sao
data access	Cho phép đọc dữ liệu qua liên kết	Lỗi Msg 7411 — not configured for DATA ACCESS
rpc	Cho phép nhận lời gọi thủ tục từ xa	Không gọi được thủ tục ở site kia
rpc out	Cho phép gọi thủ tục sang site kia	Không điều chuyển hàng được
remote proc transaction promotion	Tự nâng lời gọi thủ tục từ xa lên giao tác phân tán	Nửa việc ở máy kia không được quay lui cùng máy này

Một sự cố thực tế nhóm đã gặp và khắc phục: sau khi cấu hình nhân bản, thủ tục `sp_addsubscription` **ghi đè lại mục Linked Server** của Subscriber và **tắt tuỳ chọn `data access`**, vì bản thân nhân bản giao tiếp bằng cơ chế RPC riêng, không cần tới truy

vấn phân tán. Hậu quả là mọi truy vấn phân tán đang chạy tốt bỗng báo lỗi **Msg 7411**. Nhóm đã bổ sung một bước bật lại các tuỳ chọn này vào cuối kịch bản cấu hình nhân bản.

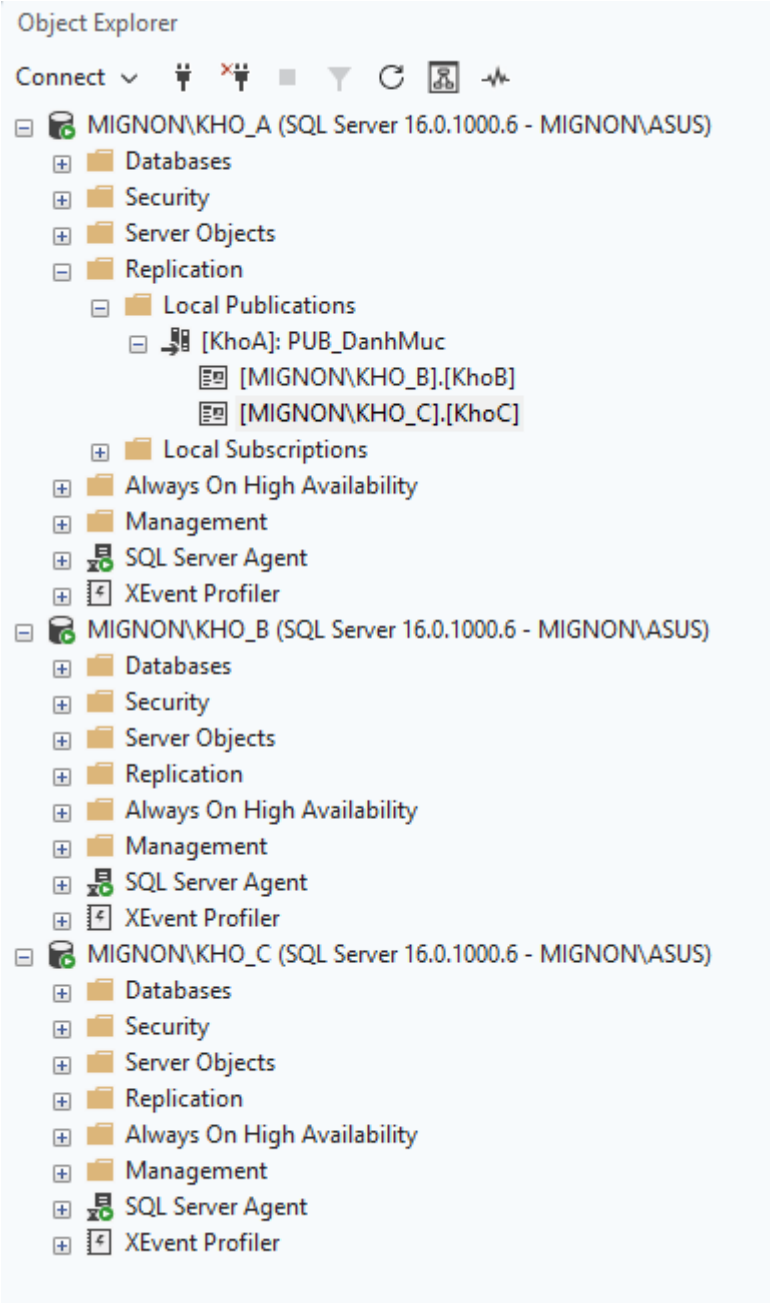


Hình III.24: Truy vấn phân tán đầu tiên — đứng tại KHO_A nhưng đọc được dữ liệu tồn kho của cả ba site

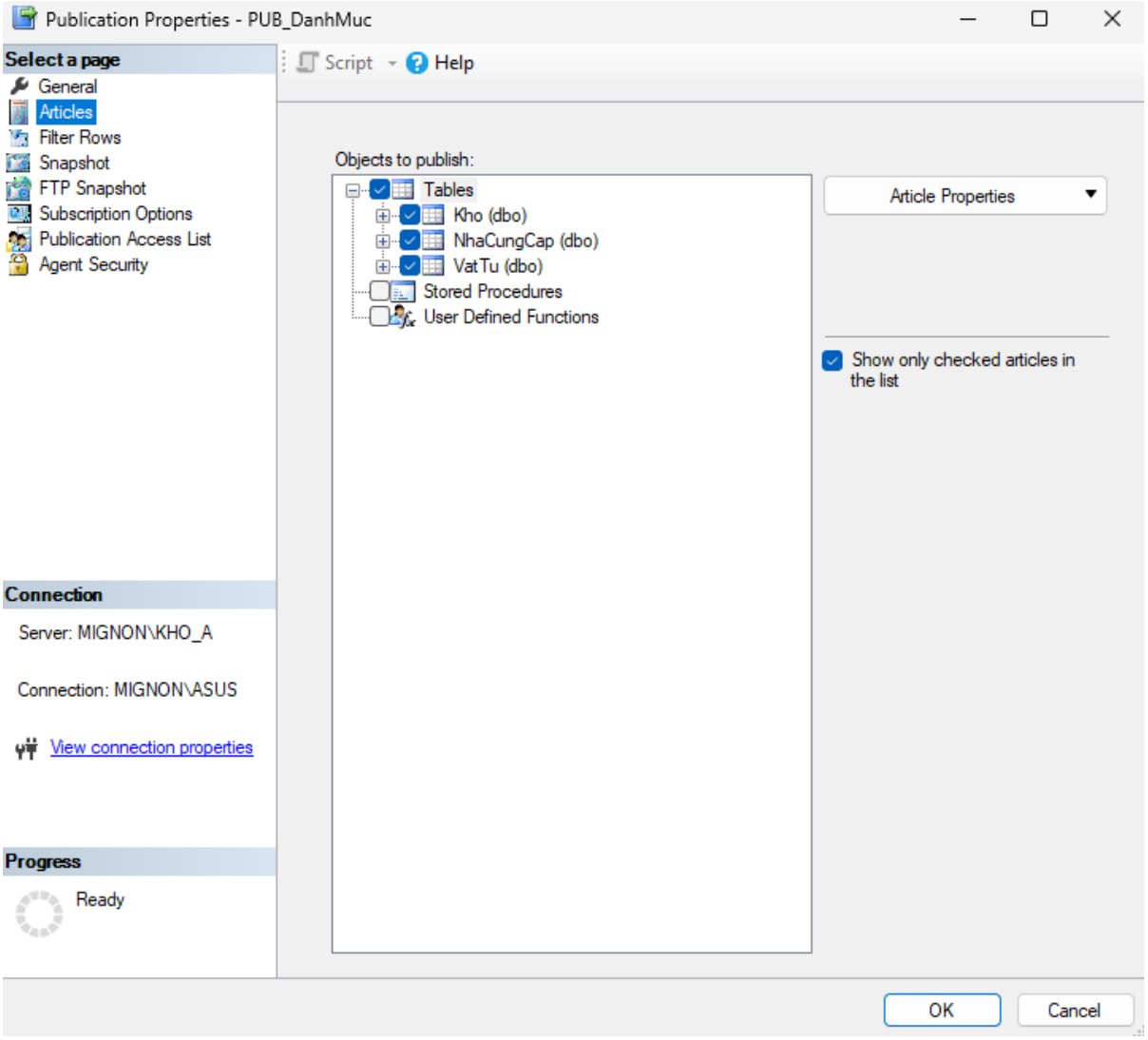
Kết quả trả về ba dòng tương ứng ba mảnh dữ liệu, chứng minh hệ thống thật sự phân tán và các liên kết hoạt động.

6. Tạo Publication

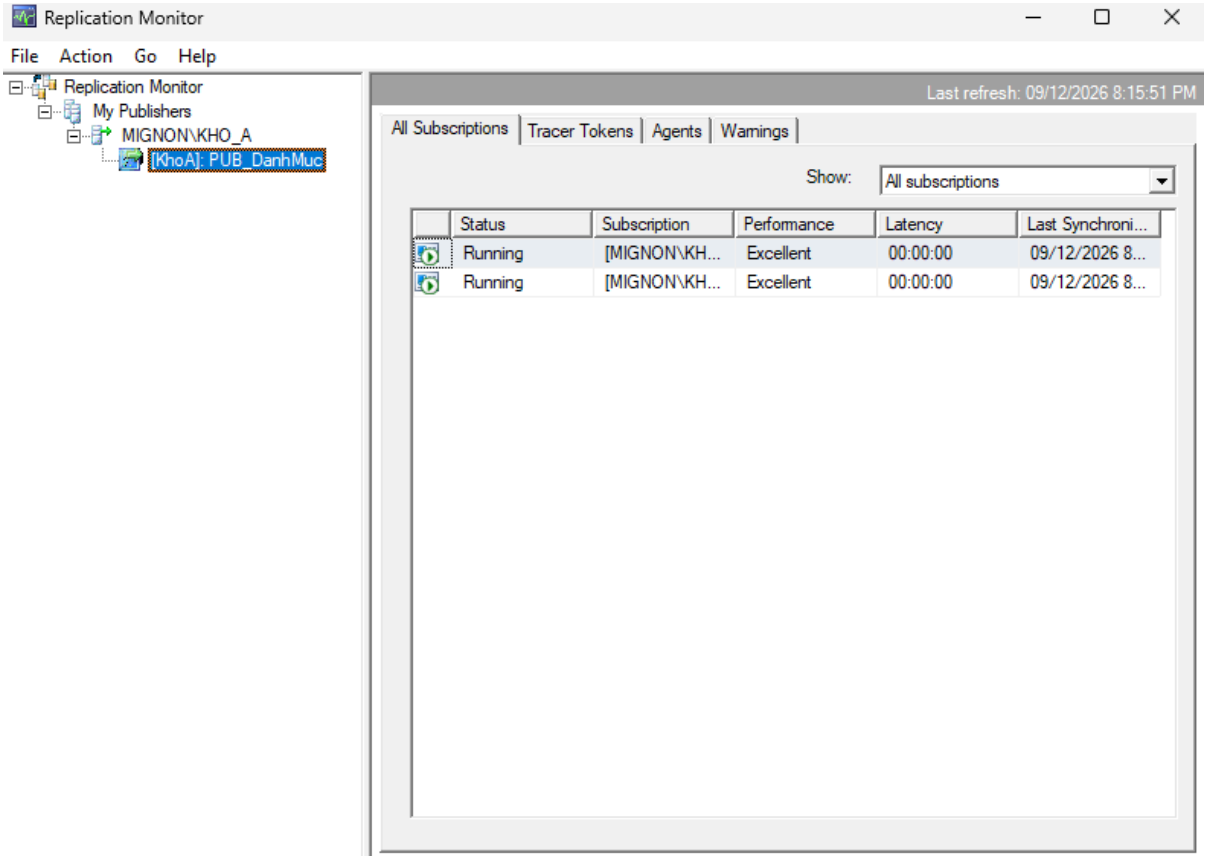
Publication là đơn vị cấu hình của Transactional Replication, mô tả **những bảng nào được nhân bản** và **nhân bản xuống đâu**. Nhóm tạo Publication tên **PUB_DanhMuc** tại KHO_A gồm ba bài viết (article): VatTu, NhaCungCap và Kho.



Hình III.25: Publication PUB_DanhMuc trong Object Explorer tại site KHO_A



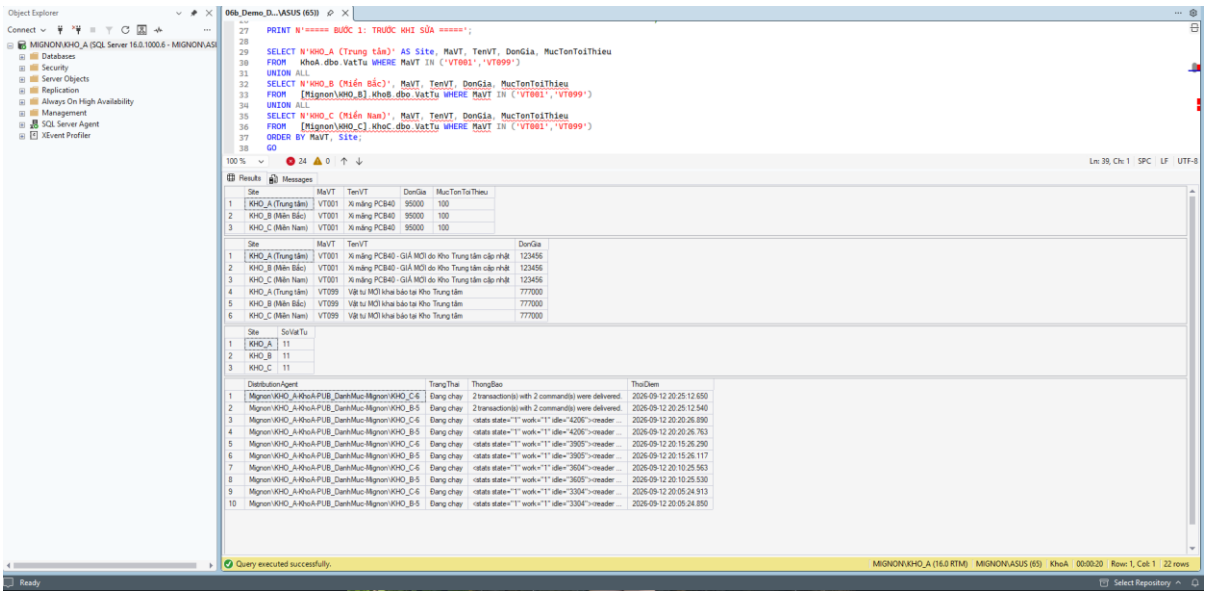
Hình III.26: Ba bảng danh mục được chọn làm article của Publication



Hình III.27: Replication Monitor — hai Subscription đều ở trạng thái hoạt động

Hai vấn đề kỹ thuật nhóm đã xử lý trong bước này:

1. **Xung đột khoá ngoại khi tạo snapshot.** Ba bảng danh mục có ràng buộc khoá ngoại với nhau và với các bảng khác ở Subscriber. Nếu để cơ chế snapshot mặc định, tiến trình sẽ cố xoá và tạo lại bảng ở Subscriber, gây xung đột. Nhóm dùng tham số `@sync_type = 'replication support only'` để chỉ tạo cơ chế nhân bản mà không đụng tới dữ liệu sẵn có.
2. **Distribution Agent không kết nối được tới Subscriber.** Tiến trình này chạy dưới tài khoản dịch vụ `NT Service\SQLAgent$KHO_A`, vốn không có quyền đăng nhập ở hai thể hiện kia. Nhóm khắc phục bằng cách đặt `@subscriber_security_mode = 0` và chỉ định tài khoản SQL đăng nhập vào Subscriber.



Hình III.28: Bằng chứng đồng bộ — sửa danh mục tại KHO_A, sau khoảng 20 giây cả KHO_B và KHO_C đều tự động cập nhật theo

Kịch bản kiểm chứng gồm ba thao tác để chứng minh cả ba loại lệnh đều được nhân bản: **UPDATE** đổi giá một vật tư, **INSERT** thêm một vật tư mới, và **DELETE** khi hoàn tác. Số lượng vật tư ở ba site luôn bằng nhau sau mỗi lần đồng bộ.

7. Thử các giao tác tương ứng với các chức năng

7.1. Nhập dữ liệu

Năm thủ tục nghiệp vụ được cài đặt và chạy **giống hệt nhau trên cả ba site**, không sửa một dòng nào. Điều này thực hiện được nhờ một hàm nhỏ tự nhận biết site đang chạy:

```
CREATE FUNCTION dbo.fn_MaKhoHienTai() RETURNS CHAR(5)
AS BEGIN
    RETURN 'KHO_' + UPPER(RIGHT(DB_NAME(), 1));
END
```

Bảng III.3: Năm thủ tục nghiệp vụ

Thủ tục	Chức năng	Kỹ thuật đáng chú ý
sp_NhapKho	F1 — Nhập vật tư vào kho	Dùng tham số kiểu bảng (TVP) để nhận nhiều dòng hàng trong một lần gọi; dùng MERGE để cộng tồn
sp_XuatKho	F2 — Xuất vật tư khỏi kho	Kiểm tra tồn với gợi ý khoá UPDLOCK, HOLDLOCK để chống mất cập nhật
sp_TraCuuTonKho	F3 — Tra cứu tồn tại chỗ	Truy vấn cục bộ, không qua mạng
sp_CanhBaoTonToiThieu	F7 — Cảnh báo thiếu hàng	So sánh tồn với mức tối thiểu trong danh mục
sp_DieuChuyenVatTu	F5 — Điều chuyển giữa hai kho	Giao tác phân tán hai pha qua MS DTC

Thủ tục xuất kho được thiết kế để **từ chối xuất quá tồn**: nếu số lượng yêu cầu vượt tồn hiện có, giao tác bị huỷ và trả về thông báo nêu rõ mã vật tư, số tồn còn lại và số cần xuất.

7.2. Hiển thị dữ liệu

Mục này kiểm chứng bốn điều: dữ liệu có được phân mảnh đúng không, có được nhân bản đúng không, các liên kết có hoạt động không, và có đồng bộ không.

a) Tồn kho toàn hệ thống

Results | Messages

	MaKho	TenKho	MaVT	TenVT	DonViTinh	SoLuong	GiaTriTon_VND
1	KHO_A	Kho Trung tâm	VT001	Xi măng PCB40	Bao	570	54,150,000
2	KHO_A	Kho Trung tâm	VT002	Thép cây phi 16	Cây	270	49,950,000
3	KHO_A	Kho Trung tâm	VT003	Gạch ống 8x8x18	Viên	12000	18,000,000
4	KHO_A	Kho Trung tâm	VT004	Cát xây dựng	Khối	60	19,200,000
5	KHO_A	Kho Trung tâm	VT005	Đá 1x2	Khối	55	20,900,000
6	KHO_A	Kho Trung tâm	VT006	Sơn nước nội thất 18L	Thùng	40	50,000,000
7	KHO_A	Kho Trung tâm	VT007	Ống nhựa PVC phi 90	Cây	150	11,700,000
8	KHO_A	Kho Trung tâm	VT008	Dây điện Cadivi 2.5mm	Cuộn	35	31,150,000
9	KHO_A	Kho Trung tâm	VT009	Bóng đèn LED 9W	Cái	700	31,500,000
10	KHO_A	Kho Trung tâm	VT010	Khóa cửa tay gạt	Bộ	90	31,500,000
11	KHO_B	Kho Miền Bắc	VT001	Xi măng PCB40	Bao	250	23,750,000
12	KHO_B	Kho Miền Bắc	VT002	Thép cây phi 16	Cây	140	25,900,000
13	KHO_B	Kho Miền Bắc	VT003	Gạch ống 8x8x18	Viên	4500	6,750,000
14	KHO_B	Kho Miền Bắc	VT004	Cát xây dựng	Khối	25	8,000,000
15	KHO_B	Kho Miền Bắc	VT006	Sơn nước nội thất 18L	Thùng	6	7,500,000
16	KHO_B	Kho Miền Bắc	VT007	Ống nhựa PVC phi 90	Cây	60	4,680,000
17	KHO_B	Kho Miền Bắc	VT009	Bóng đèn LED 9W	Cái	250	11,250,000
18	KHO_C	Kho Miền Nam	VT001	Xi măng PCB40	Bao	310	29,450,000
19	KHO_C	Kho Miền Nam	VT002	Thép cây phi 16	Cây	180	33,300,000
20	KHO_C	Kho Miền Nam	VT003	Gạch ống 8x8x18	Viên	6000	9,000,000
21	KHO_C	Kho Miền Nam	VT005	Đá 1x2	Khối	30	11,400,000
22	KHO_C	Kho Miền Nam	VT008	Dây điện Cadivi 2.5mm	Cuộn	18	16,020,000
23	KHO_C	Kho Miền Nam	VT009	Bóng đèn LED 9W	Cái	420	18,900,000
24	KHO_C	Kho Miền Nam	VT010	Khóa cửa tay gạt	Bộ	45	15,750,000

	MaKho	TenKho	TenVT	SoLuong	MucTonToiThieu	TinhTrang
1	KHO_A	Kho Trung tâm	Bóng đèn LED 9W	700	200	đủ
2	KHO_B	Kho Miền Bắc	Bóng đèn LED 9W	250	200	đủ
3	KHO_C	Kho Miền Nam	Bóng đèn LED 9W	420	200	đủ

	Muc	MaVT	TongSoLuong	SoKhoDangGiu
1	TỔNG TOÀN HỆ THỐNG	VT009	1370	3

Hình III.29: Tồn kho toàn hệ thống — 24 dòng gộp từ ba site, hiển thị như một bảng duy nhất

Một câu lệnh SELECT duy nhất, nhưng bộ tối ưu đã phải mở hai kết nối qua Linked Server, lấy dữ liệu từ KHO_B và KHO_C, gộp với dữ liệu tại chỗ rồi mới tính tổng. Người dùng hoàn toàn không thấy điều đó — **đây chính là trong suốt phân mảnh**.

b) Kiểm tra phân mảnh

Results | Messages

	PhepKiemTra	Site	SoMaKhoKhacNhau	MaKhoDuyNhat	SoDong
1	5.1 TÁCH RỜI	KHO_A	1	KHO_A	10
2	5.1 TÁCH RỜI	KHO_B	1	KHO_B	7
3	5.1 TÁCH RỜI	KHO_C	1	KHO_C	7

	PhepKiemTra	SoDongBiTrung	KetLuan
1	5.1 TÁCH RỜI - TÌM DÒNG TRÙNG GIỮA CÁC MẢNH	0	✔ ĐẠT - các mảnh hoàn toàn rời nhau

	PhepKiemTra	SoKhoTrongDanhMuc	SoMangCoDuLieu	KetLuan
1	5.2 ĐẦY ĐỦ	3	3	✔ ĐẠT - không kho nào bị bỏ sót

	PhepKiemTra	Manh_A	Manh_B	Manh_C	CongBaManh	QuanHeToanCuc	KetLuan
1	5.3 TÁI THIẾT	10	7	7	24	24	✔ ĐẠT nếu hai cột cuối bằng nhau

	PhepKiemTra	SoDongChiTietNhap	SoDongThuocDungManh	SoDongChiTietXuat	SoDongXuatDungManh	KetLuan
1	5.4 DẪN XUẤT - KHO_A	6	6	4	4	✔ ĐẠT nếu cột 1 = cột 2 và cột 3 = cột 4

	PhepKiemTra	SoHieuLoi	KetLuan	ThongBaoLoi
1	5.5 THỬ VI PHẠM	547	✔ ĐẠT - ràng buộc CK_TonKho_Manh đã CHẶN dữ liệu...	The INSERT statement conflicted with the CHECK c...

Hình III.30: Kiểm chứng ba tính chất của phân mảnh cùng phép thử vi phạm

Bảng III.4: Kết quả kiểm chứng phân mảnh

Phép kiểm tra	Kết quả thực tế	Kết luận
Tính tách rời	Mỗi site chỉ chứa đúng 1 mã kho; 0 dòng trùng giữa các mảnh	ĐẠT
Tính đầy đủ	3 kho trong danh mục = 3 mảnh có dữ liệu	ĐẠT
Tính tái thiết	10 + 7 + 7 = 24 dòng, bằng đúng số dòng quan hệ toàn cục	ĐẠT
Phân mảnh dẫn xuất	6 dòng chi tiết nhập đều thuộc phiếu của site mình; 4 dòng chi tiết xuất tương tự	ĐẠT
Thử ghi sai mảnh	Bị chặn bằng lỗi 547 — ràng buộc CK_TonKho_Manh	ĐẠT

c) Kiểm tra nhân bản và đồng bộ

Results | Messages

	Site	SoVatTu	TongDonGia	VanTayBang
1	KHO_A (Publisher)	10	3594500	347490305
2	KHO_B (Subscriber)	10	3594500	347490305
3	KHO_C (Subscriber)	10	3594500	347490305

	PhepKiemTra	SoVanTayKhacNhau	KetLuan
1	6.1 SO KHỚP VÂN TAY BẢNG VatTu	1	✔ ĐẠT - ba bản sao GIỐNG HỆT nhau, replication đang khoẻ

	TinhTrang	MaVT	TenVT	DonGia

	Bang	Tai_A	Tai_B	Tai_C
1	NhaCungCap	4	4	4
2	Kho	3	3	3
3	VatTu	10	10	10

	TenPublication	MoTa	TrangThai	SoBangDuocNhanBan
1	PUB_DanhMuc	Dong bo danh muc vat tu, nha cung cap va danh sa...	Đang hoạt động	3

	Subscriber	CSDL_Dich	TrangThai	KieuDongBo	SoBangNhanDuoc
1	MIGNON\KHO_B	KhoB	Đang hoạt động	Không snapshot	3
2	MIGNON\KHO_C	KhoC	Đang hoạt động	Không snapshot	3

Hình III.31: So khớp ba bản sao danh mục bằng vân tay CHECKSUM_AGG

Phương pháp kiểm chứng: tính **vân tay (checksum tổng hợp)** của toàn bộ bảng VatTu tại cả ba site. Chỉ cần một ô dữ liệu lệch nhau là vân tay đổi ngay. Kết quả thực tế: cả ba site cùng cho giá trị **347490305**, nghĩa là ba bản sao giống hệt nhau tới từng ô.

Phép kiểm tra thứ hai dùng toán tử EXCEPT để tìm những dòng có ở site này mà thiếu ở site kia — kết quả không trả về dòng nào, xác nhận đồng bộ hoàn toàn.

d) Kiểm tra Linked Server

Results | Messages

	LinkedServer	SanPham	TrinhCungCap	NguonDuLieu	ChoDocDuLieu	ChoGoiThuTucTuXa	NangLenGiaoTacPhanTan
1	Mignon\KHO_B		MSOLEDBSQL	Mignon\KHO_B	1	1	1
2	Mignon\KHO_C		MSOLEDBSQL	Mignon\KHO_C	1	1	1

	LinkedServer	TaiKhoanDangNhapTuXa	KieuAnhXa
1	Mignon\KHO_B	sa	Ánh xạ sang tài khoản cố định
2	Mignon\KHO_C	sa	Ánh xạ sang tài khoản cố định

	PhepKiemTra	LinkedServer	TrangThai
1	7.3 THỬ KẾT NỐI	Mignon\KHO_B	✔ THÔNG
2	7.3 THỬ KẾT NỐI	Mignon\KHO_C	✔ THÔNG

	Site	TenMayChu	PhienBan
1	KHO_A	Mignon\KHO_A	Developer Edition (64-bit)
2	KHO_B	Mignon\KHO_B	Developer Edition (64-bit)
3	KHO_C	Mignon\KHO_C	Developer Edition (64-bit)

Hình III.32: Danh sách liên kết, ánh xạ đăng nhập, kết quả thử kết nối và thông tin phiên bản đọc ngược từ hai site kia

Bảng cuối cùng trong ảnh đọc ngược `@@SERVERNAME` và phiên bản của hai site còn lại thông qua liên kết — bằng chứng trực tiếp rằng liên kết đang hoạt động thật.

e) Trong suốt vị trí

Results | Messages

	TenKho	MaKho	MaVT	SoLuong
1	Kho Trung tâm	KHO_A	VT009	700
2	Kho Miền Bắc	KHO_B	VT009	250
3	Kho Miền Nam	KHO_C	VT009	420

	TenKho	MaKho	MaVT	SoLuong
1	Kho Trung tâm	KHO_A	VT001	570
2	Kho Miền Bắc	KHO_B	VT001	250
3	Kho Miền Nam	KHO_C	VT001	310
4	Kho Trung tâm	KHO_A	VT002	270
5	Kho Miền Bắc	KHO_B	VT002	140
6	Kho Miền Nam	KHO_C	VT002	180
7	Kho Trung tâm	KHO_A	VT003	12000
8	Kho Miền Bắc	KHO_B	VT003	4500
9	Kho Miền Nam	KHO_C	VT003	6000
10	Kho Trung tâm	KHO_A	VT004	60
11	Kho Miền Bắc	KHO_B	VT004	25
12	Kho Trung tâm	KHO_A	VT005	55
13	Kho Miền Nam	KHO_C	VT005	30
14	Kho Trung tâm	KHO_A	VT006	40
15	Kho Miền Bắc	KHO_B	VT006	6
16	Kho Trung tâm	KHO_A	VT007	150
17	Kho Miền Bắc	KHO_B	VT007	60
18	Kho Trung tâm	KHO_A	VT008	35
19	Kho Miền Nam	KHO_C	VT008	18
20	Kho Trung tâm	KHO_A	VT009	700
21	Kho Miền Bắc	KHO_B	VT009	250
22	Kho Miền Nam	KHO_C	VT009	420
23	Kho Trung tâm	KHO_A	VT010	90
24	Kho Miền Nam	KHO_C	VT010	45

Hình III.33: Thủ tục sp_TonKhoToanHeThong dựng câu lệnh tự động từ bảng Kho

Thủ tục này đọc tên máy chủ từ **cột ServerName của bảng Kho** rồi mới dựng câu lệnh. Thêm một kho mới chỉ cần thêm một dòng vào bảng Kho, **không phải sửa một dòng mã nguồn nào** — đó là trong suốt vị trí.

7.3. Thống kê

Results | Messages

	MaKho	TenKho	SoMatHang	TongSoLuong	TongGiaTri_VND	SoMatHangThieu
1	*	>>> CỘNG TOÀN HỆ THỐNG	24	26204	539,700,000	2
2	KHO_A	Kho Trung tâm	10	13970	318,050,000	0
3	KHO_B	Kho Miền Bắc	7	5231	87,830,000	2
4	KHO_C	Kho Miền Nam	7	7003	133,820,000	0

	MaVT	TenVT	DonViTinh	TongToanHeThong	Tai_KHO_A	Tai_KHO_B	Tai_KHO_C	SoKhoDangGiu
1	VT003	Gạch ống 8x8x18	Viên	22500	12000	4500	6000	3
2	VT009	Bóng đèn LED 9W	Cái	1370	700	250	420	3
3	VT001	Xi măng PCB40	Bao	1130	570	250	310	3
4	VT002	Thép cây phi 16	Cây	590	270	140	180	3
5	VT007	Ống nhựa PVC phi 90	Cây	210	150	60	NULL	2
6	VT010	Khóa cửa tay gạt	Bộ	135	90	NULL	45	2
7	VT004	Cát xây dựng	Khối	85	60	25	NULL	2
8	VT005	Đá 1x2	Khối	85	55	NULL	30	2
9	VT008	Dây điện Cadivi 2.5mm	Cuộn	53	35	NULL	18	2
10	VT006	Sơn nước nội thất 18L	Thùng	46	40	6	NULL	2

	MaVT	TenVT	TongSoLuong	TongGiaTri_VND	TyTrongVon
1	VT002	Thép cây phi 16	590	109,150,000	20.22 %
2	VT001	Xi măng PCB40	1130	107,350,000	19.89 %
3	VT009	Bóng đèn LED 9W	1370	61,650,000	11.42 %
4	VT006	Sơn nước nội thất 18L	46	57,500,000	10.65 %
5	VT010	Khóa cửa tay gạt	135	47,250,000	8.75 %

	Site	SoPhieuNhap	SoPhieuXuat	SoPhieuDieuChuyen
1	KHO_A	3	3	2
2	KHO_B	3	3	2
3	KHO_C	3	3	0

Hình III.34: Thống kê toàn hệ thống theo kho, theo vật tư và theo tỷ trọng vốn

Bảng III.5: Số liệu thống kê toàn hệ thống

Site	Số mặt hàng	Tổng số lượng	Tổng giá trị (VND)
KHO_A — Kho Trung tâm	10	13.970	318.050.000
KHO_B — Kho Miền Bắc	7	5.231	87.830.000
KHO_C — Kho Miền Nam	7	7.003	133.820.000
Cộng toàn hệ thống	**24**	**26.204**	**539.700.000**

Results | Messages

	KhoThieu	TenKhoThieu	MaVT	TenVT	TonHienCo	MucToiThieu	CanBoSung	KhoCoTheCapBu	SoLuongCoTheCho	DeXuatXuLy
1	KHO_B	Kho Miền Bắc	VT003	Gạch ống 8x8x18	4500	5000	500	KHO_A	7000	✔ Điều chuyển 500 đơn vị từ KHO_A sang KHO_B
2	KHO_B	Kho Miền Bắc	VT003	Gạch ống 8x8x18	4500	5000	500	KHO_C	1000	✔ Điều chuyển 500 đơn vị từ KHO_C sang KHO_B
3	KHO_B	Kho Miền Bắc	VT006	Sơn nước nội thất 18L	6	10	4	KHO_A	30	✔ Điều chuyển 4 đơn vị từ KHO_A sang KHO_B

Hình III.35: Cảnh báo thiếu hàng kèm đề xuất kho cấp bù

Đây là truy vấn phân tán **có giá trị nghiệp vụ thật**, không chỉ để minh hoạ: nó tìm mặt hàng đang dưới mức tối thiểu ở kho này, đồng thời chỉ ra kho nào đang dư để điều chuyển sang. Kết quả của nó chính là đầu vào cho thủ tục điều chuyển.

```
PHẦN 8 - LINKED SERVER (4 PHẦN TÊN)  SO VỚI  OPENQUERY

SQL Server parse and compile time:
   CPU time = 0 ms, elapsed time = 238 ms.
--- CÁCH 1: bốn phần tên - kéo dữ liệu thô về rồi mới gom nhóm ---

 SQL Server Execution Times:
   CPU time = 0 ms,  elapsed time = 0 ms.

 SQL Server Execution Times:
   CPU time = 0 ms,  elapsed time = 6 ms.
SQL Server parse and compile time:
   CPU time = 0 ms, elapsed time = 6 ms.
--- CÁCH 2: OPENQUERY - máy từ xa gom nhóm sẵn, chỉ trả về kết quả ---

 SQL Server Execution Times:
   CPU time = 0 ms,  elapsed time = 0 ms.

 SQL Server Execution Times:
   CPU time = 16 ms,  elapsed time = 2 ms.
SQL Server parse and compile time:
   CPU time = 0 ms, elapsed time = 0 ms.

 SQL Server Execution Times:
   CPU time = 0 ms,  elapsed time = 0 ms.

Completion time: 2026-09-12T23:53:19.2252499+07:00
```

Hình III.36: So sánh hai cách viết truy vấn phân tán qua thời gian thực thi

Cùng một nhu cầu có hai cách viết. Cách dùng **tên bốn phần** buộc máy chủ phải kéo toàn bộ dòng dữ liệu thô về rồi mới gom nhóm. Cách dùng **OPENQUERY** gửi nguyên văn câu lệnh sang máy đích, máy đó tự gom nhóm rồi chỉ trả về kết quả đã tóm tắt. Trong cơ sở dữ liệu phân tán, chi phí lớn nhất là truyền dữ liệu qua mạng, nên với bảng lớn cách thứ hai luôn thắng.

7.4. Trigger phân quyền bảo vệ các bảng

Hệ thống dùng **ba lớp bảo vệ chồng lên nhau**, mỗi lớp bắt được loại vi phạm mà lớp kia không thấy.

Bảng III.6: Ba lớp bảo vệ dữ liệu

Lớp	Cơ chế	Bắt được gì
Lớp 1	Ràng buộc CHECK và khoá ngoại	Dữ liệu sai mảnh, dữ liệu mồ côi. Nhanh nhất, không thể vô hiệu hoá
Lớp 2	Sáu trigger	Những luật mà ràng buộc không diễn tả nổi, và ghi nhật ký kiểm toán
Lớp 3	Hệ thống quyền GRANT / DENY	Ngăn người dùng chạm thẳng vào bảng, buộc đi qua thủ tục

Sáu trigger được cài ở cả ba site:

Bảng III.7: Danh sách trigger bảo vệ

Trigger	Bảo vệ bảng	Nhiệm vụ
trg_TonKho_BaoVeManh	TonKho	Chặn ghi dữ liệu của site khác, cấm sửa MaKho để đẩy dòng sang mảnh khác, đồng thời ghi nhật ký mọi thay đổi tồn kho
trg_ChiTietNhap_BaoVeManhDanXuat	ChiTietNhap	Giữ bất biến của phép nửa nối: mọi dòng chi tiết phải thuộc một phiếu của chính site này
trg_ChiTietXuat_BaoVeManhDanXuat	ChiTietXuat	Tương tự cho phiếu xuất
trg_VatTu_ChiSuaTaiTrungTam	VatTu	Bản sao nhân bản là chỉ đọc — chặn người dùng sửa tay tại KHO_B và KHO_C
trg_NhaCungCap_ChiSuaTaiTrungTam	NhaCungCap	Tương tự cho danh mục nhà cung cấp
trg_PhieuNhap_CamXoa	PhieuNhap	Cấm xoá chứng từ đã hoàn tất — dùng INSTEAD OF DELETE

Vì sao cần trigger trong khi đã có ràng buộc CHECK? Ràng buộc CHECK chỉ nhìn được các cột trong cùng một dòng. Nó không làm được bốn việc sau:

1. Kiểm tra mảnh **dẫn xuất** — phải nối sang bảng phiếu mới biết dòng chi tiết có đúng mảnh hay không.
2. Phân biệt "người dùng sửa" với "tiến trình nhân bản đẩy dữ liệu về" — hai việc giống hệt nhau về câu lệnh nhưng khác hẳn về tính hợp lệ. Giải quyết bằng tuỳ chọn **NOT FOR REPLICATION**.
3. Ghi **nhật ký kiểm toán** ai sửa gì, lúc nào, từ máy nào.
4. Cấm xoá chứng từ đã hoàn tất — cần **INSTEAD OF DELETE** để chặn trước khi lệnh chạm vào dữ liệu.

Results | Messages

	VaiTro	DoiTuong	Loai	Quyen	TrangThai
1	BanGiamDoc	Kho	Bảng	SELECT	✔ CHO PHÉP
2	BanGiamDoc	NhatKyKiemToan	Bảng	SELECT	✔ CHO PHÉP
3	BanGiamDoc	PhieuDieuChuyen	Bảng	SELECT	✔ CHO PHÉP
4	BanGiamDoc	TonKho	Bảng	SELECT	✔ CHO PHÉP
5	BanGiamDoc	VatTu	Bảng	SELECT	✔ CHO PHÉP
6	BanGiamDoc	TonKho	Bảng	UPDATE	✗ CẤM (DENY)
7	BanGiamDoc	TonKho	Bảng	DELETE	✗ CẤM (DENY)
8	BanGiamDoc	TonKho	Bảng	INSERT	✗ CẤM (DENY)
9	NhanVienKho	ChiTietNhap	Bảng	SELECT	✔ CHO PHÉP
10	NhanVienKho	ChiTietVatTu_Type	Kiểu bảng (TVP)	EXECUTE	✔ CHO PHÉP
11	NhanVienKho	ChiTietXuat	Bảng	SELECT	✔ CHO PHÉP
12	NhanVienKho	Kho	Bảng	SELECT	✔ CHO PHÉP
13	NhanVienKho	NhaCungCap	Bảng	SELECT	✔ CHO PHÉP
14	NhanVienKho	PhieuDieuChuyen	Bảng	SELECT	✔ CHO PHÉP
15	NhanVienKho	PhieuNhap	Bảng	SELECT	✔ CHO PHÉP
16	NhanVienKho	PhieuXuat	Bảng	SELECT	✔ CHO PHÉP
17	NhanVienKho	sp_CanhBaoTonToiThieu	Thủ tục	EXECUTE	✔ CHO PHÉP
18	NhanVienKho	sp_NhapKho	Thủ tục	EXECUTE	✔ CHO PHÉP
19	NhanVienKho	sp_TraCuuTonKho	Thủ tục	EXECUTE	✔ CHO PHÉP
20	NhanVienKho	sp_XuatKho	Thủ tục	EXECUTE	✔ CHO PHÉP
21	NhanVienKho	TonKho	Bảng	SELECT	✔ CHO PHÉP
22	NhanVienKho	VatTu	Bảng	SELECT	✔ CHO PHÉP
23	NhanVienKho	NhaCungCap	Bảng	UPDATE	✗ CẤM (DENY)
24	NhanVienKho	NhaCungCap	Bảng	DELETE	✗ CẤM (DENY)
25	NhanVienKho	NhaCungCap	Bảng	INSERT	✗ CẤM (DENY)

	VaiTro	ThanhVien	Loai
1	BanGiamDoc	giam_doc	Là người dùng
2	NhanVienKho	TruongKho	Là VAI TRÒ con - kế thừa toàn bộ quyền của vai t...
3	NhanVienKho	nv_kho	Là người dùng
4	QuanTriDanhMuc	qt_danhmuc	Là người dùng
5	TruongKho	truong_kho	Là người dùng

	TenTrigger	BangDuocBaoVe	Kieu	TrangThai	PhamVi
1	trg_ChiTietNhap_BaoVeManhDanXuat	ChiTietNhap	AFTER	✔ Đang bật	Bỏ qua lệnh của replication
2	trg_ChiTietXuat_BaoVeManhDanXuat	ChiTietXuat	AFTER	✔ Đang bật	Bỏ qua lệnh của replication
3	trg_NhaCungCap_ChiSuaTaiTrungTam	NhaCungCap	AFTER	✔ Đang bật	Bỏ qua lệnh của replication
4	trg_PhieuNhap_CamXoa	PhieuNhap	INST...	✔ Đang bật	Áp dụng cho mọi lệnh
5	trg_TonKho_BaoVeManh	TonKho	AFTER	✔ Đang bật	Bỏ qua lệnh của replication
6	trg_VatTu_ChiSuaTaiTrungTam	VatTu	AFTER	✔ Đang bật	Bỏ qua lệnh của replication

Hình III.37: Bảng tổng hợp quyền đã cấp cho bốn vai trò và danh sách trigger đang hoạt động

Results | Messages

	STT	VaiTro	HanhDong	KetQua	ChiTiet
1	1	nv_kho	SELECT trên TonKho	✔ ĐƯỢC PHÉP	Đọc được 10 dòng tồn kho của kho mình
2	2	nv_kho	UPDATE thẳng vào TonKho	✔ BỊ CHẶN ĐÚNG	DENY có hiệu lực - buộc phải dùng sp_XuatKho / sp...
3	3	nv_kho	UPDATE danh mục VatTu	✔ BỊ CHẶN ĐÚNG	Danh mục là nguồn/bản sao nhân bản, nhân viên kh...
4	4	giam_doc	DELETE trên TonKho	✔ BỊ CHẶN ĐÚNG	BanGiamDoc là vai trò CHỈ ĐỌC toàn hệ thống
5	5	nv_kho	EXECUTE sp_DieuChuyenVatTu	✔ BỊ CHẶN ĐÚNG	Chỉ TruongKho mới được điều chuyển hàng sang kho...
6	6	truong_kho	EXECUTE sp_DieuChuyenVatTu	✔ ĐƯỢC PHÉP	Trưởng kho có quyền khởi tạo giao tác phân tán

Hình III.38: Kết quả thử quyền — ai làm được gì, ai bị chặn

Sáu phép thử đều cho kết quả đúng như thiết kế. Đáng chú ý nhất là phép thử thứ hai: nhân viên kho **đọc được** bảng tồn kho nhưng **không ghi thẳng vào được**, buộc phải đi qua thủ tục — chính là cơ chế chuỗi sở hữu đã trình bày ở Chương II.

Results | Messages

	STT	PhepThu	KetQua	SoHieuLoi	LopChan	ThongBao
1	1	Ghi tồn kho của site khác	✔ BỊ CHẶN	547	Lớp 1 - CHECK constraint	The INSERT statement conflicted with the CHECK constr...
2	2	Đổi MaKho để chuyển mảnh	✔ BỊ CHẶN	547	Lớp 1 - CHECK constraint	The UPDATE statement conflicted with the CHECK const...
3	3	Chi tiết nhập mồ côi (mảnh dẫn xuất)	✔ BỊ CHẶN	547	Lớp 1 - khoá ngoại	The INSERT statement conflicted with the FOREIGN KE...
4	4	Xoá chứng từ đã hoàn tất	✔ BỊ CHẶN	50000	Lớp 2 - TRIGGER (chỉ trigger làm được)	CẤM XOÁ CHỨNG TỪ: phiếu nhập PNA0001 đã ở trạng ...
5	5	Sửa danh mục VatTu tại site này	✔ ĐƯỢC PHÉP	0	Publisher - đúng thiết kế	Đây là Publisher, được sửa và sẽ nhân bản xuống 2 site kia

Hình III.39: Thử trigger tại site KHO_A — cột LỚP CHẶN chỉ rõ hàng phòng thủ nào đã bắt được vi phạm

Cùng một kịch bản này được chạy ở cả ba site và cho kết quả khác nhau đúng như thiết kế. Với phép thử cuối — sửa danh mục vật tư — tại **KHO_A** (Publisher) thì **được phép**, còn tại **KHO_B** và **KHO_C** (Subscriber) thì **bị trigger chặn** bằng lỗi 50000 kèm thông báo *"BẢNG NHÂN BẢN CHỈ ĐỌC"*. Đây là luật mà không một ràng buộc nào của SQL diễn tả nổi.

Results | Messages

	STT	PhepThu	KetQua	SoHieuLoi	LopChan	ThongBao
1	1	Ghi tồn kho của site khác	✔ BỊ CHẶN	547	Lớp 1 - CHECK constraint	The INSERT statement conflicted with the CHECK constra...
2	2	Đổi MaKho để chuyển mảnh	✔ BỊ CHẶN	547	Lớp 1 - CHECK constraint	The UPDATE statement conflicted with the CHECK constr...
3	3	Chi tiết nhập mồ côi (mảnh dẫn xuất)	✔ BỊ CHẶN	547	Lớp 1 - khoá ngoại	The INSERT statement conflicted with the FOREIGN KEY...
4	4	Xoá chứng từ đã hoàn tất	✔ BỊ CHẶN	50000	Lớp 2 - TRIGGER (chỉ trigger làm được)	CẤM XOÁ CHỨNG TỪ: phiếu nhập PNB0001 đã ở trạng t...
5	5	Sửa danh mục VatTu tại site này	✔ BỊ CHẶN	50000	Lớp 2 - TRIGGER (chỉ trigger làm được)	BẢNG NHÂN BẢN CHỈ ĐỌC: danh mục VatTu tại KHO_B ...

Hình III.40: Cùng kịch bản chạy tại site KHO_B. Bốn dòng đầu giống hệt, riêng dòng thứ năm đổi từ được phép sang bị chặn — bằng chứng cho thấy trigger phân biệt được vai trò Publisher và Subscriber

Cột **LỚP CHẶN** được suy ra từ số hiệu lỗi, cho thấy chính xác hàng phòng thủ nào đã bắt được hành vi sai:

Bảng III.8: Kết quả thử năm hành vi vi phạm

Hành vi thử	Số hiệu lỗi	Lớp chặn
Ghi tồn kho của site khác	547	Lớp 1 — ràng buộc CHECK
Sửa MaKho để đẩy dòng sang mảnh khác	547	Lớp 1 — ràng buộc CHECK
Ghi dòng chi tiết nhập mồ côi	547	Lớp 1 — khoá ngoại
Xoá chứng từ đã hoàn tất	50000	**Lớp 2 — trigger** (chỉ trigger làm được)
Sửa danh mục tại bản sao	50000	**Lớp 2 — trigger** (chỉ trigger làm được)

Results | Messages

	MaNK	ThoiDiem	Site	NguoiDung	MayTram	Bang	HanhDong	NoiDung
1	10	2026-09-13 00:52:51	KHO_A	MIGNON\ASUS	MIGNON	VatTu	UPDATE	Sửa danh mục -> sẽ NHÂN BẢN xuống KHO_B và KHO_...
2	9	2026-09-13 00:52:51	KHO_A	MIGNON\ASUS	MIGNON	PhieuNhap	DELETE-BLOCKED	Có người cố xoá phiếu nhập đã hoàn tất PNA0001 -> TRI...
3	8	2026-09-12 21:12:42	KHO_A	sa	MIGNON	VatTu	UPDATE	Sửa danh mục -> sẽ NHÂN BẢN xuống KHO_B và KHO_...
4	7	2026-09-12 21:12:10	KHO_A	sa	MIGNON	VatTu	UPDATE	Sửa danh mục -> sẽ NHÂN BẢN xuống KHO_B và KHO_...
5	6	2026-09-12 21:11:59	KHO_A	sa	MIGNON	VatTu	UPDATE	Sửa danh mục -> sẽ NHÂN BẢN xuống KHO_B và KHO_...
6	5	2026-09-12 21:11:59	KHO_A	sa	MIGNON	PhieuNhap	DELETE-BLOCKED	Có người cố xoá phiếu nhập đã hoàn tất PNA0001 -> TRI...
7	4	2026-09-12 21:11:04	KHO_A	sa	MIGNON	VatTu	UPDATE	Sửa danh mục -> sẽ NHÂN BẢN xuống KHO_B và KHO_...
8	3	2026-09-12 21:11:04	KHO_A	sa	MIGNON	PhieuNhap	DELETE-BLOCKED	Có người cố xoá phiếu nhập đã hoàn tất PNA0001 -> TRI...
9	2	2026-09-12 21:10:23	KHO_A	sa	MIGNON	VatTu	UPDATE	Sửa danh mục -> sẽ NHÂN BẢN xuống KHO_B và KHO_...

	Bang	HanhDong	SoLan	LanDauTien	LanGanNhat
1	VatTu	UPDATE	6	2026-09-12 21:10:23	2026-09-13 00:52:51
2	PhieuNhap	DELETE-BLOCKED	3	2026-09-12 21:11:04	2026-09-13 00:52:51

Hình III.41: Nhật ký kiểm toán do trigger ghi lại

Nhật ký ghi lại cả những hành vi **bị từ chối**, kèm tên tài khoản và tên máy trạm đã thực hiện.

Một chi tiết kỹ thuật nhóm phát hiện trong quá trình kiểm thử: ban đầu trigger ghi nhật ký rồi mới phát lỗi. Chạy thử thấy bảng nhật ký nhảy số thứ tự từ 1 sang 2 mà không có dòng nào — nguyên nhân là lệnh quay lui của trigger **cuốn theo cả dòng nhật ký vừa ghi**, chỉ số IDENTITY thì đã bị tiêu tốn. Nhóm đã chuyển việc ghi nhật ký sang phía gọi, trong khối bắt lỗi, nhờ đó dòng nhật ký mới tồn tại được.

7.5. Thử các giao tác

a) Giao tác phân tán — kịch bản bắt buộc của đề tài

Bối cảnh nghiệp vụ được dựng đúng theo yêu cầu đề bài: Kho Miền Bắc báo thiếu mặt hàng **VT009 — Bóng đèn LED 9W**, tồn chỉ còn 150 trong khi mức tối thiểu là 200, thiếu đúng 50 đơn vị. Kho Trung tâm điều chuyển sang 50 đơn vị.

Thủ tục `sp_DieuChuyenVatTu` thực hiện bốn việc trong **một giao tác phân tán duy nhất**:

1. Trừ tồn kho tại kho nguồn, có đặt khoá để chống tương tranh.
2. Cộng tồn kho tại kho đích thông qua Linked Server.
3. Ghi phiếu điều chuyển tại kho nguồn.
4. Ghi phiếu điều chuyển tại kho đích.

Bốn việc này nằm trên **hai máy chủ khác nhau**, nên bắt buộc phải dùng `BEGIN DISTRIBUTED TRANSACTION` với MS DTC làm bộ điều phối.

Kịch bản T1 — điều chuyển thành công

100 % 17 0

Results Messages

	Site	MaVT	SoLuong	NgayCapNhat
1	KHO_A (Trung tâm)	VT009	800	2026-09-12 20:46:20
2	KHO_B (Miền Bắc)	VT009	150	2026-09-12 20:46:20

	Site	MaVT	SoLuong	NgayCapNhat
1	KHO_A (Trung tâm)	VT009	750	2026-09-12 20:49:59
2	KHO_B (Miền Bắc)	VT009	200	2026-09-12 20:49:59

	LuuTai	MaDC	MaKhoNguon	MaKhoDich	MaVT	SoLuong	TrangThai
1	KHO_A	DCA001	KHO_A	KHO_B	VT009	50	HOAN_TAT
2	KHO_B	DCA001	KHO_A	KHO_B	VT009	50	HOAN_TAT

Hình III.42: Kịch bản T1 — tồn kho trước và sau khi điều chuyển, phiếu được ghi ở cả hai site

```
--- TRƯỚC KHI ĐIỀU CHUYỂN ---
✔ THÀNH CÔNG  DCA002 : KHO_A -> KHO_B  |  VT009 x 50
--- SAU KHI ĐIỀU CHUYỂN (A giảm 50, B tăng 50) ---
--- PHIẾU ĐIỀU CHUYỂN GHI Ở CẢ HAI SITE ---

Completion time: 2026-09-12T20:54:00.3910022+07:00
```

Hình III.43: Thông báo của kịch bản T1

Bảng III.9: Kết quả kịch bản T1

Thời điểm	Tồn tại KHO_A	Tồn tại KHO_B	Phiếu điều chuyển
Trước	800	150	chưa có
Sau	**750**	**200**	Ghi ở **cả hai site**, trạng thái HOAN_TAT

Tổng số lượng vật tư VT009 toàn hệ thống trước và sau đều bằng **1370** — không mất, không sinh thêm hàng.

Kịch bản T2 — không đủ hàng, bị từ chối trước khi thay đổi gì

Results | Messages

	Site	MaVT	SoLuong	NgayCapNhat
1	KHO_A (Trung tâm)	VT001	570	2026-09-12 20:27:51
2	KHO_B (Miền		250	2026-09-12 20:27:51

Site (nvarchar(17))
Click to select the whole column

	Site	MaVT	SoLuong	NgayCapNhat
1	KHO_A (Trung tâm)	VT001	570	2026-09-12 20:27:51
2	KHO_B (Miền Bắc)	VT001	250	2026-09-12 20:27:51

Hình III.44: Kịch bản T2 — tồn kho hai site trước và sau đều y nguyên

```
--- TRƯỚC ---
✗ THẤT BẠI - ĐÃ ROLLBACK TOÀN BỘ : KHÔNG ĐỦ HÀNG: kho KHO_A chỉ còn 570, cần chuyển 999999.
   >>> Đã bắt được lỗi đúng như mong đợi
--- SAU (phải GIỐNG HỆT bảng TRƯỚC) ---

Completion time: 2026-09-12T20:53:52.1237186+07:00
```

Hình III.45: Thông báo từ chối của kịch bản T2

Thủ tục kiểm tra tồn **trước khi** thực hiện bất cứ thay đổi nào, phát hiện không đủ hàng nên từ chối ngay. Tồn kho ở cả hai site giữ nguyên 570 và 250.

Kịch bản T3 — lỗi xảy ra giữa chừng, sau khi đã trừ kho nguồn

Đây là kịch bản **quan trọng nhất** của cả đồ án, trả lời trực tiếp yêu cầu *"nếu giao dịch thất bại giữa chừng thì dữ liệu phải được xử lý nhất quán"*.

Nhóm cố tình gây lỗi mô phỏng mất kết nối **sau khi kho nguồn đã bị trừ hàng**. Nếu hệ thống không có cơ chế bảo vệ, kết quả sẽ là: kho nguồn mất 50 đơn vị, kho đích không nhận được gì — hàng **biến mất khỏi hệ thống**.

Results | Messages

	Site	MaVT	SoLuong	NgayCapNhat
1	KHO_A (Trung tâm)	VT001	570	2026-09-12 20:27:51
2	KHO_B (Miền Bắc)	VT001	250	2026-09-12 20:27:51

	SoPhieu_KhoA
1	1

	Site	MaVT	SoLuong	NgayCapNhat
1	KHO_A (Trung tâm)	VT001	570	2026-09-12 20:27:51
2	KHO_B (Miền Bắc)	VT001	250	2026-09-12 20:27:51

	SoPhieu_KhoA
1	1

Hình III.46: Kịch bản T3 — tồn kho và số phiếu ở cả hai site đều y nguyên sau khi giao tác bị huỷ

```
--- TRƯỚC ---
✗ THẤT BẠI - ĐÃ ROLLBACK TOÀN BỘ : >>> LỖI MÔ PHỎNG: mất kết nối tới kho đích giữa chừng <<<
   >>> Giao tác đã bị huỷ, MS DTC rollback cả hai site
--- SAU (tồn kho PHẢI y nguyên, số phiếu PHẢI không đổi) ---

Completion time: 2026-09-12T20:53:04.9470914+07:00
```

Hình III.47: Thông báo quay lui của kịch bản T3

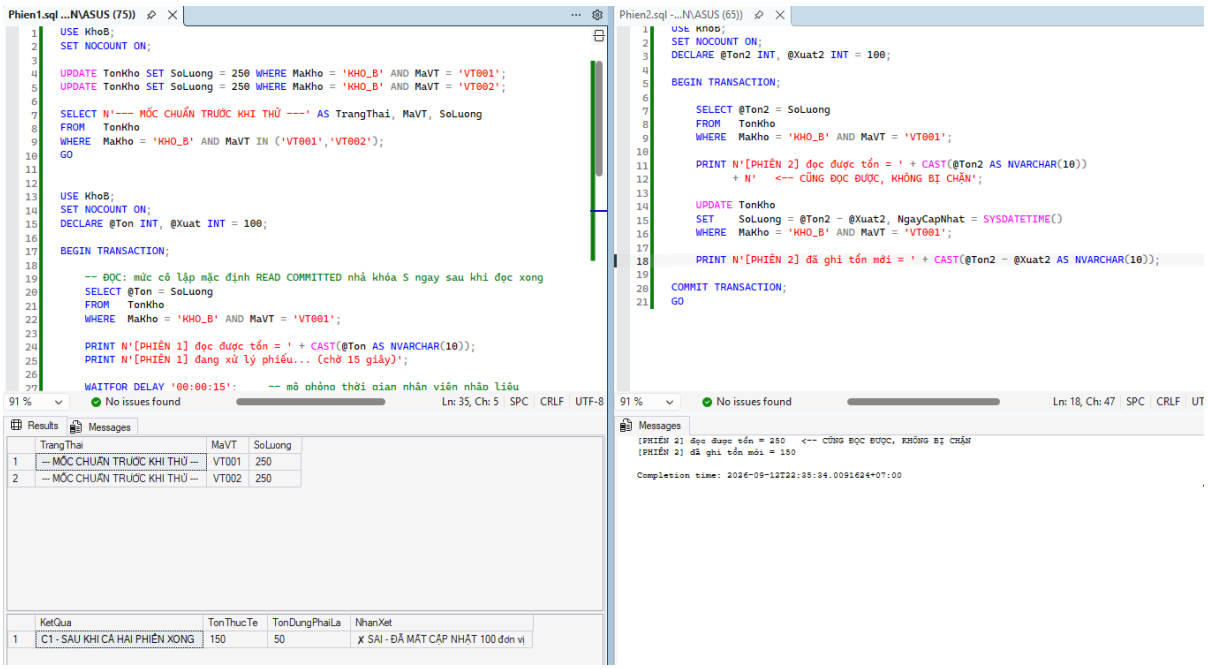
Kết quả thực tế: MS DTC phát hiện một site không hoàn thành được, ra lệnh **quay lui toàn bộ ở cả hai máy chủ**. Tồn kho trở về đúng giá trị ban đầu, số phiếu điều chuyển không đổi. **Không một đơn vị hàng nào bị mất.**

Bảng III.10: Tổng kết ba kịch bản giao tác phân tán

Kịch bản	Tình huống	Kết quả	Trạng thái dữ liệu
T1	Đủ hàng, mọi bước thành công	Commit ở cả hai site	KHO_A 800→750, KHO_B 150→200, phiếu ghi ở cả hai nơi
T2	Không đủ hàng ở kho nguồn	Từ chối trước khi thay đổi gì	Cả hai site y nguyên 570 và 250
T3	Lỗi giữa chừng, **sau khi đã trừ kho nguồn**	Quay lui toàn bộ ở cả hai site	Tồn kho và số phiếu **y nguyên** — không mất hàng

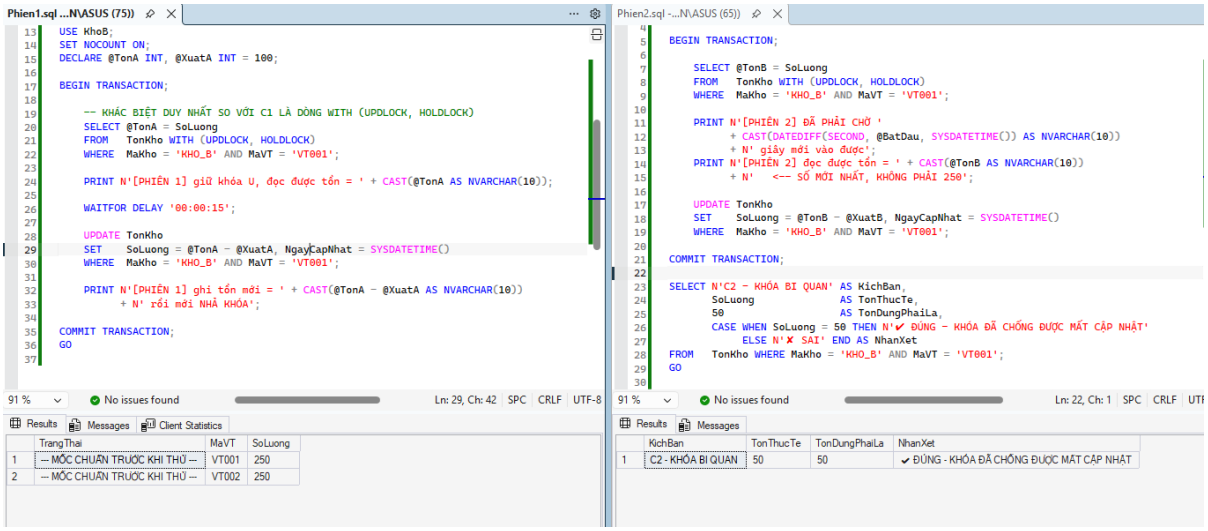
b) Tình huống tương tranh

Nhóm dựng bốn kịch bản, mỗi kịch bản chạy bằng **hai phiên làm việc song song** để mô phỏng hai nhân viên thao tác cùng lúc trên cùng một mặt hàng.



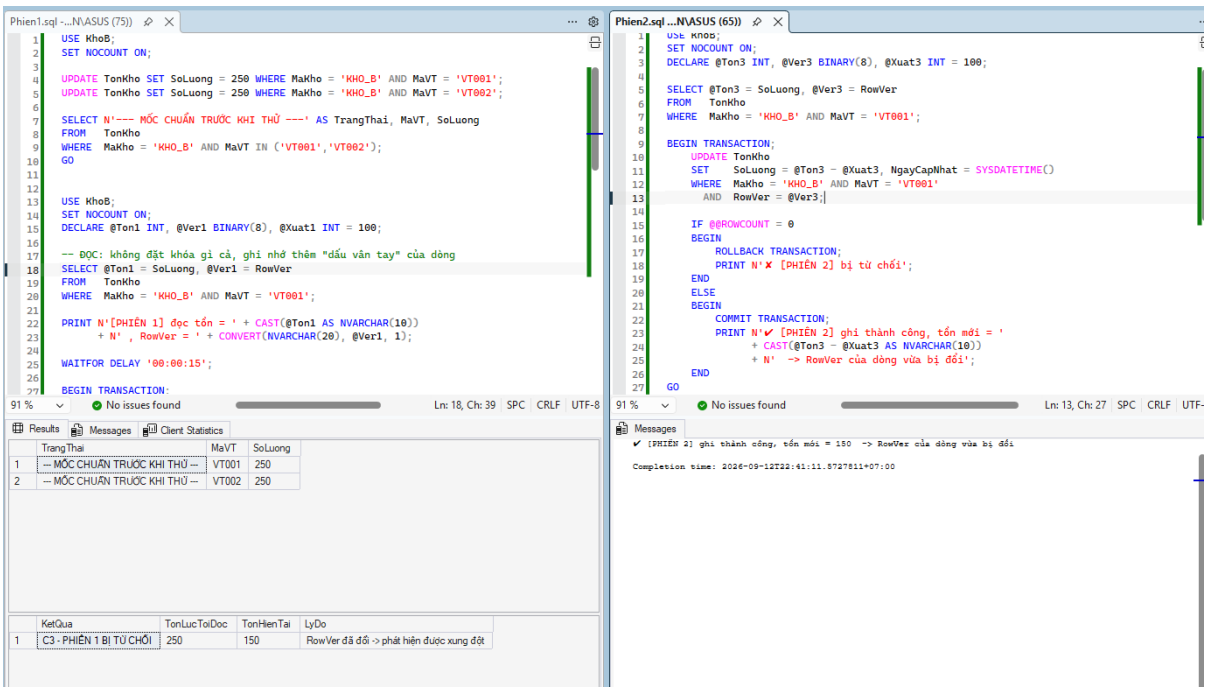
Hình III.48: Kịch bản C1 — hiện tượng mất cập nhật khi không dùng khoá

Hai phiên cùng xuất 100 đơn vị từ tồn 250, kết quả đúng phải là 50. Nhưng cả hai cùng đọc được 250, cùng tính 250 − 100 = 150 rồi cùng ghi đè. Lần ghi sau xoá sạch lần ghi trước. **Tồn cuối cùng là 150, mất đúng 100 đơn vị** — và nguy hiểm ở chỗ hệ thống hoàn toàn không hay biết.



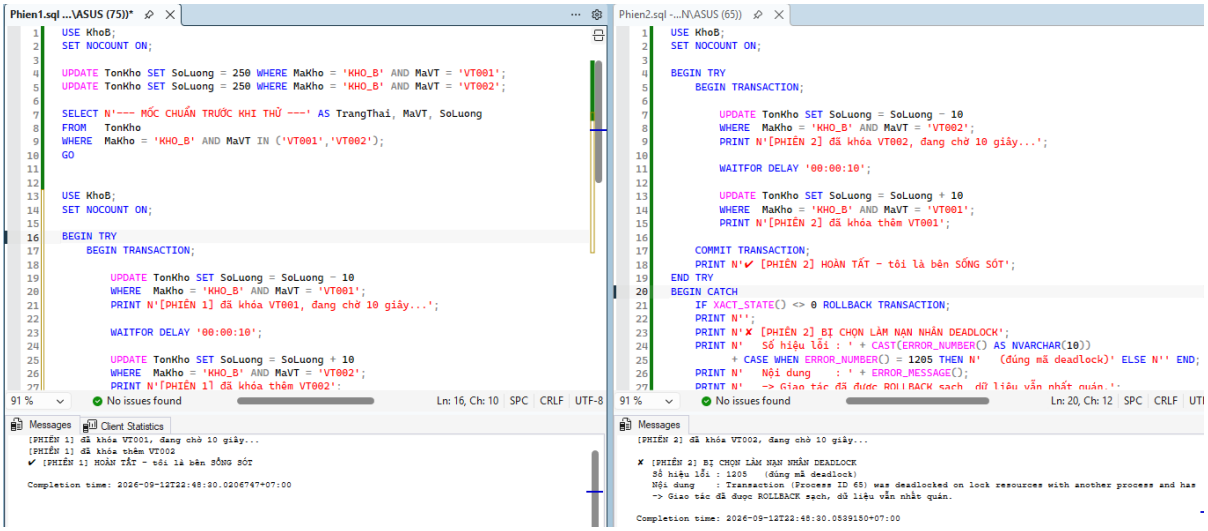
Hình III.49: Kịch bản C2 — khoá bi quan khắc phục triệt để

Chỉ thêm gợi ý `WITH (UPDLOCK, HOLDLOCK)` vào câu đọc, phiên thứ hai bị chặn lại và **phải chờ 13 giây** cho tới khi phiên thứ nhất kết thúc. Sau đó nó đọc được số mới nhất là 150 và ghi ra kết quả đúng **50**.



Hình III.50: Kịch bản C3 — khoá lạc quan bằng cột ROWVERSION

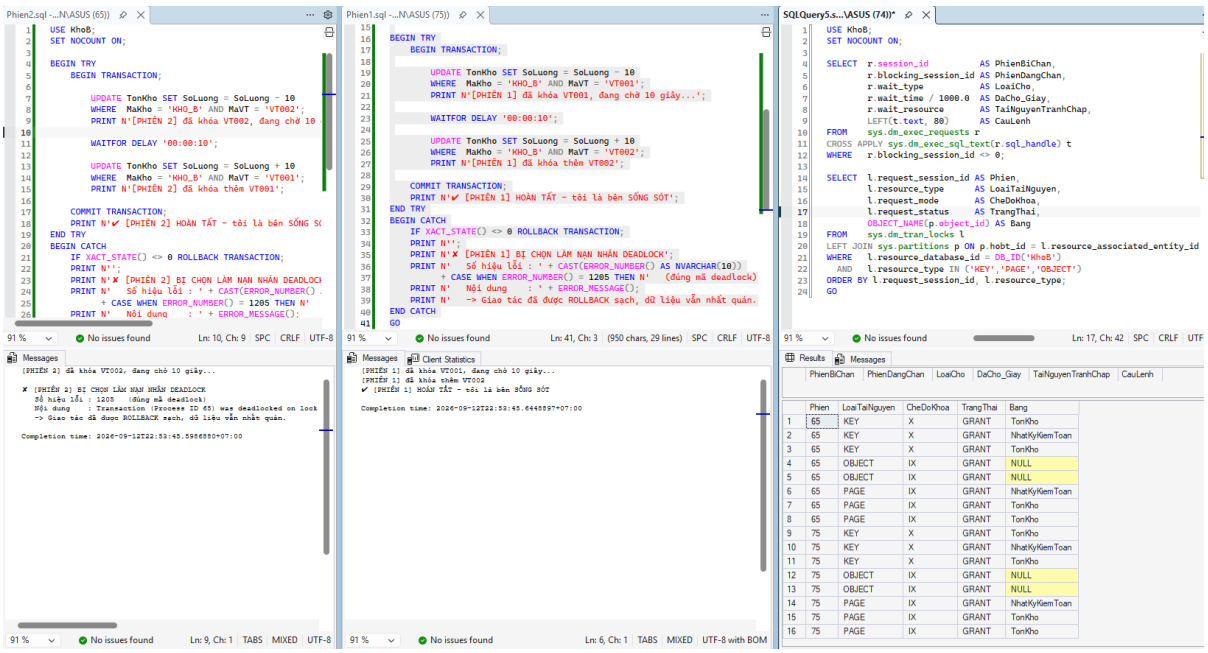
Cách thứ hai không khoá gì cả, nhưng lúc ghi thì kiểm tra cột phiên bản. Phiên thứ hai ghi thành công và làm đổi giá trị cột này. Phiên thứ nhất khi ghi phát hiện phiên bản đã khác, `@@ROWCOUNT` trả về 0, nên **từ chối ghi và báo lỗi xung đột** thay vì ghi đè âm thầm.



Hình III.51: Kịch bản C4 — khoá chết và cơ chế tự gỡ của SQL Server

Hai giao tác khoá hai mặt hàng theo **thứ tự ngược nhau**, mỗi bên giữ thứ bên kia cần nên chờ nhau vĩnh viễn. Bộ giám sát khoá của SQL Server phát hiện vòng tròn chờ, **chọn một giao tác làm nạn nhân** và huỷ nó bằng lỗi **Msg 1205**. Giao tác còn lại chạy tiếp bình thường.

Kiểm chứng sau đó cho thấy tổng số lượng hai mặt hàng vẫn đúng bằng 500 — **cơ sở dữ liệu vẫn nhất quán, không cần con người can thiệp**.



Hình III.52: Quan sát các khoá đang tồn tại trong hệ thống

Bảng III.11: Tổng kết bốn kịch bản tương tranh

Mã	Kịch bản	Kết quả	Ý nghĩa
C1	Không dùng khoá	Tồn 150 — **SAI**	Mất cập nhật, hệ thống không hề hay biết
C2	UPDLOCK + HOLDLOCK	Tồn 50 — **ĐÚNG**	Kết quả chính xác, phiên sau phải chờ 13 giây
C3	ROWVERSION	Tồn 150 — **ĐÚNG**	Phát hiện xung đột và báo lỗi cho phiên thua
C4	Deadlock hai chiều	Tổng 500 — **ĐÚNG**	SQL Server tự gỡ bằng Msg 1205

Lựa chọn của nhóm: dùng **khoá bi quan** cho các thao tác trong cùng một site, vì thao tác kho rất ngắn, tranh chấp ít, và khoá bi quan cho kết quả đúng tuyệt đối mà không bắt người dùng nhập lại phiếu. Cột phiên bản vẫn được giữ trong bảng tồn kho để dành cho tầng ứng dụng web sau này, nơi một màn hình có thể mở hàng phút trước khi bấm Lưu — lúc đó khoá bi quan sẽ chặn cả hệ thống quá lâu.

CHƯƠNG IV. KẾT LUẬN

1. Kết quả đạt được

Nhóm đã xây dựng hoàn chỉnh và **chạy kiểm thử thực tế** một hệ cơ sở dữ liệu phân tán quản lý kho vật tư trên ba site độc lập. Toàn bộ các yêu cầu của đề tài đều được đáp ứng và có minh chứng bằng ảnh chụp kết quả chạy thật.

Bảng IV.1: Đối chiếu yêu cầu đề tài với kết quả

Yêu cầu	Cách đáp ứng	Kết quả kiểm chứng
Có từ 2 đến 4 site	Ba thể hiện SQL Server độc lập	Ba dịch vụ riêng, ba Agent riêng, ba cổng riêng
Ít nhất một phép phân mảnh	Phân mảnh ngang **nguyên thủy** và **dẫn xuất**	10 + 7 + 7 = 24 dòng; 0 dòng trùng giữa các mảnh
Ít nhất một nhân bản	Transactional Replication ba bảng danh mục	Ba site cùng vân tay 347490305
Ít nhất một giao tác phân tán	sp_DieuChuyenVatTu qua MS DTC	T1 thành công, T2 và T3 quay lui sạch
Ít nhất một tình huống tương tranh	Bốn kịch bản C1 đến C4	Tái hiện mất cập nhật và khắc phục bằng hai cách
Ít nhất một truy vấn phân tán	Khung nhìn gộp ba site và thủ tục động	24 dòng gộp từ ba máy chủ
Đồng bộ dữ liệu	Nhân bản danh mục và giao tác phân tán	Độ trễ đo được 15–20 giây
Nhập tại Kho A, xuất tại Kho B	sp_NhapKho và sp_XuatKho	Chạy thật trên cả ba site
Kiểm tra tồn toàn hệ thống	v_TonKho_ToanHeThong	Tổng 26.204 đơn vị, giá trị 539.700.000 đồng
Điều chuyển giữa hai kho	sp_DieuChuyenVatTu	KHO_A 800→750, KHO_B 150→200
Thất bại giữa chừng phải nhất quán	Giao thức hai pha, quay lui toàn bộ	**Tồn kho và số phiếu ở cả hai site y nguyên**

2. Những vấn đề đã gặp và cách khắc phục

Quá trình triển khai gặp một số vấn đề thực tế mà tài liệu lý thuyết không nêu. Nhóm ghi lại đây vì đó là phần học được nhiều nhất.

Bảng IV.2: Các sự cố đã xử lý

Sự cố	Nguyên nhân	Cách khắc phục
Msg 7411 — not configured for DATA ACCESS	sp_addsubscription ghi đè mục Linked Server và tắt tuỳ chọn data access	Bổ sung bước bật lại các tuỳ chọn sau khi cấu hình nhân bản

Sự cố	Nguyên nhân	Cách khắc phục
Distribution Agent không kết nối được Subscriber	Agent chạy dưới tài khoản dịch vụ không có quyền ở site kia	Đặt subscriber_security_mode = 0 và chỉ định tài khoản SQL
Xung đột khoá ngoại khi tạo snapshot	Snapshot cố tạo lại bảng đã có dữ liệu ở Subscriber	Dùng sync_type = replication support only
Nhật ký kiểm toán bị mất khi trigger từ chối	Lệnh quay lui của trigger cuốn theo cả dòng nhật ký vừa ghi	Chuyển việc ghi nhật ký sang phía gọi, trong khối bắt lỗi
Mất dấu tiếng Việt trong câu lệnh động	Thiếu tiền tố N nên chuỗi bị hiểu là VARCHAR	Thêm tiền tố N vào chuỗi khi dựng câu lệnh
Tuỳ chọn nâng lên giao tác phân tán bị tắt	Cũng do sp_addsubscription ghi đè cấu hình liên kết	Bật lại remote proc transaction promotion trên cả sáu liên kết

3. Đánh giá

Ưu điểm của hệ thống:

- 89% nghiệp vụ hằng ngày chạy **hoàn toàn tại chỗ**, không phụ thuộc đường truyền.
- Định nghĩa mảnh được cưỡng chế ngay tại tầng cơ sở dữ liệu, **không thể ghi sai mảnh** kể cả với quyền quản trị cao nhất.
- Giao tác điều chuyển bảo đảm nguyên tố tuyệt đối, đã kiểm chứng bằng kịch bản gây lỗi cố ý.
- Mã nguồn của các thủ tục **giống hệt nhau ở cả ba site**, giảm chi phí bảo trì.
- Ba lớp bảo vệ chồng nhau, mỗi lớp bắt được loại vi phạm mà lớp kia không thấy.

Hạn chế còn tồn tại:

- Nhân bản danh mục là **một chiều**, nên khi Kho Trung tâm gặp sự cố thì không site nào sửa được danh mục.
- Distributor đặt chung máy với Publisher, chưa tách riêng như hệ thống thật.
- Chưa có cơ chế tự động thử lại giao tác khi gặp deadlock — hiện tại việc này để cho tầng ứng dụng.
- Chưa xây dựng phần mềm ứng dụng cho các trạm; hiện thao tác qua công cụ quản trị.

4. Hướng phát triển

1. Xây dựng phần mềm ứng dụng cho máy trạm theo đúng mô hình front-end đã thiết kế, dùng cột phiên bản để điều khiển tương tranh lạc quan.

2. Tách Distributor ra máy riêng để giảm tải cho Publisher.
3. Bổ sung cơ chế tự động thử lại khi gặp lỗi 1205.
4. Mở rộng sang mô hình nhiều kho hơn — thiết kế hiện tại đã sẵn sàng, vì thêm kho mới chỉ cần thêm một dòng vào bảng Kho.
5. Bổ sung sao lưu và phục hồi theo lịch cho từng site.

TÀI LIỆU THAM KHẢO

[1] M. Tamer Özsu, Patrick Valduriez. Principles of Distributed Database Systems, 4th Edition. Springer, 2020.

[2] Microsoft. SQL Server Replication Documentation. learn.microsoft.com/en-us/sql/relational-databases/replication

[3] Microsoft. Distributed Transactions and Microsoft Distributed Transaction Coordinator. learn.microsoft.com/en-us/sql/relational-databases/native-client-ole-db-transactions

[4] Microsoft. Linked Servers (Database Engine). learn.microsoft.com/en-us/sql/relational-databases/linked-servers

[5] Microsoft. Transaction Locking and Row Versioning Guide. learn.microsoft.com/en-us/sql/relational-databases/sql-server-transaction-locking-and-row-versioning-guide

[6] Microsoft. DML Triggers. learn.microsoft.com/en-us/sql/relational-databases/triggers/dml-triggers

[7] ZeroTier Inc. ZeroTier Manual. docs.zerotier.com

[8] Slide bài giảng môn Cơ sở dữ liệu phân tán, Học viện Công nghệ Bưu chính Viễn thông.