Trang lót bìa

Trang bìa

**NHẬN XÉT CỦA ĐƠN VỊ THỰC TẬP**

**NHẬN XÉT CỦA GVHD**

**LỜI CẢM ƠN**

**LỜI CAM ĐOAN**

**MỤC LỤC**

[LỜI MỞ ĐẦU 1](#_Toc164843475)

[1. Lý do chọn đề tài 1](#_Toc164843476)

[2. Mục đích nghiên cứu 1](#_Toc164843477)

[3. Đối tượng và phạm vi nghiên cứu 1](#_Toc164843478)

[4. Phương pháp nghiên cứu 1](#_Toc164843479)

[5. Bố cục đề tài 1](#_Toc164843480)

[CHƯƠNG 1: TỔNG QUAN 2](#_Toc164843481)

[1.1. Khái quát về Công ty Tài chính Ngân hàng TMCP Sài Gòn – Hà Nội SHB Finance 2](#_Toc164843482)

[1.1.1. Giới thiệu chung về SHB Finance 2](#_Toc164843483)

[1.1.2. Thành tựu, mục tiêu, tầm nhìn, sứ mệnh 3](#_Toc164843484)

[1.1.3. Cơ cấu tổ chức 3](#_Toc164843485)

[1.2. Mô tả bài toán 4](#_Toc164843486)

[1.2.1. Thực trạng 4](#_Toc164843487)

[1.2.2. Thách thức 6](#_Toc164843488)

[1.2.3. Giá trị mang lại 6](#_Toc164843489)

[1.2.4. Hướng giải quyết 7](#_Toc164843490)

[1.3. Giới thiệu về Oracle Data Guard 9](#_Toc164843491)

[1.3.1. Kiến trúc Oracle Database 9](#_Toc164843492)

[1.3.2. Sao lưu với công cụ Recovery Manager (RMAN) 11](#_Toc164843493)

[1.3.3. Khái niệm, kiến trúc và điểm chính của Oracle Data Guard 12](#_Toc164843494)

[1.3.4. Loại hình bảo vệ 14](#_Toc164843495)

[1.3.5. Cơ chế tương tác giữa các thành phần 19](#_Toc164843496)

[1.3.6. Oracle Data Guard Broker 26](#_Toc164843497)

[1.4. Kết luận chương I 30](#_Toc164843498)

[CHƯƠNG 2: TRIỂN KHAI GIẢI PHÁP ORACLE DATA GUARD CHO CƠ SỞ DỮ LIỆU HỆ THỐNG XẾP HẠNG TÍN DỤNG CỦA SHBFINANCE 31](#_Toc164843499)

[2.1. Lên kế hoạch xây dựng giải pháp Data Guard cho SHBFinance 31](#_Toc164843500)

[2.1.1. Xác định vấn đề 31](#_Toc164843501)

[2.1.2. Xác định phương thức chuyển đổi dự phòng 34](#_Toc164843502)

[2.1.3. Xác định về đường truyền và đồng bộ dữ liệu 36](#_Toc164843503)

[2.1.4. Xác định chế độ bảo vệ trong Data Guard 36](#_Toc164843504)

[2.1.5. Xác định việc giảm tải hệ thống chính 38](#_Toc164843505)

[2.1.6. Xác định yêu cầu phần cứng, phần mềm 38](#_Toc164843506)

[2.2. Thực nghiệm triển khai giải pháp Data Guard dựa trên RMAN Duplicate và nền tảng điện toán đám mây Oracle Cloud Infrastructure 39](#_Toc164843507)

[2.2.1. Kiến trúc tổng quan 39](#_Toc164843508)

[2.2.2. Môi trường Oracle Net và định danh CSDL 43](#_Toc164843509)

[2.2.3. Cấu hình tham số chung cho hệ thống chính 46](#_Toc164843510)

[2.2.4. Tạo hệ thống dự phòng dựa trên RMAN DUPLICATE 51](#_Toc164843511)

[2.2.5. Cấu hình môi trường Data Guard 55](#_Toc164843512)

[2.2.6. Mô hình giám sát tính khả dụng hệ thống 59](#_Toc164843513)

[2.3. Kiểm thử hoạt động hệ thống 68](#_Toc164843514)

[2.4. Phân tích sự cố mất ghi dữ liệu trong môi trường Data Guard 72](#_Toc164843515)

[2.4.1. Khái niệm 72](#_Toc164843516)

[2.4.2. Phát hiện vấn đề mất ghi với Lost Writes Protection 73](#_Toc164843517)

[2.4.3. Thực nghiệm cơ chế thông báo Lost Writes 75](#_Toc164843518)

[2.5. Kết quả đạt được sau khi triển khai 78](#_Toc164843519)

[2.6. Kết luận chương II 79](#_Toc164843520)

[CHƯƠNG 3: KẾT LUẬN 80](#_Toc164843521)

[3.1. Kết quả đóng góp 80](#_Toc164843522)

[3.2. Hạn chế và hướng phát triển 80](#_Toc164843523)

[3.3. Một số giải pháp dự phòng các hãng khác 81](#_Toc164843524)

[PHỤ LỤC 82](#_Toc164843525)

[TÀI LIỆU THAM KHẢO 83](#_Toc164843526)

**DANH MỤC CÁC CHỮ VIẾT TẮT**

|  |  |  |
| --- | --- | --- |
| **Chữ viết tắt** | **Chữ đầy đủ** | **Diễn giải** |
| TNHH MTV | Trách nhiệm hữu hạn, một thành viên | Loại hình công ty do một tổ chức/cá nhân làm chủ sở hữu |
| TMCP | Thương mại cổ phần | Loại hình công ty kinh doanh theo mô hình cổ phần – vốn góp của các cổ đông |
| RPO | Recovery Point Object | Chỉ số mục tiêu về lượng dữ liệu phục hồi |
| RTO | Recovery Time Object | Chỉ số mục tiêu về thời gian phục hồi |
| RMAN | Recovery Manager | Công cụ sao lưu và phục hồi của Oracle |
| OSB | Oracle Secure Backup | Tính năng giúp bảo mật bản sao lưu |
| MTTR | Mean Time To Recovery | Chỉ số thời gian trung bình phục hồi giữa các lần gặp sự cố |
| MAA | Maximum Availability Architeture | Khung kiến trúc giải pháp sẵn sàng cao của Oracle |
| ODG | Oracle Data Guard | Giải pháp khôi phục sau thảm họa của Oracle |
| ADG | Active Data Guard | Tính năng trong Data Guard cho phép truy vấn song song với quá trình đồng bộ |
| LNS | LogWriter Network Server | Tên chung cho các tiến trình thực hiện vận chuyển dữ liệu đồng bộ |
| RFS | Remote File Server | Tiến trình thực hiện nhận dữ liệu đồng bộ của CSDL dự phòng |
| MRP | Managed Recovery Process | Tiến trình thực hiện áp dụng thay đổi của CSDL dự phòng dạng vật lý |
| LSP | Logical Standby Process | Tiến trình thực hiện áp dụng thay đổi của CSDL dự phòng dạng lô-gic |
| CSDL | Cơ sở dữ liệu | Hệ thống lưu trữ dữ liệu. Trong phạm vi bài này là CSDL dạng quan hệ |
| ASM | Automatic Storage Management | Tính năng quản lý đĩa, vùng nhớ tự động thay vì File Systems trong Oracle |
| I/O | Input/Output | Hoạt động đọc/ghi dữ liệu của máy tính |
| FAL | Fetch Archived Log | Tính năng giúp CSDL dự phòng chủ động trong việc xử lý trễ/thiếu dữ liệu đồng bộ |
| ACK | Acknowledgement | Tín hiệu mà tiến trình RFS trả về CSDL chính khi xử dụng AFFIRM/NOAFFIRM |
| DML | Data Manipulation | Ngôn ngữ thao tác dữ liệu trên CSDL như INSERT, UPDATE |
| LGWR | Log Writer | Tiến trình ghi dữ liệu đồng bộ từ bộ nhớ xuống đĩa |
| DBWR | Database Writer | Tiến trình ghi dữ liệu từ bộ nhớ xuống đĩa |
| ARL | Archived Redo Log | Tệp tin được lưu cất của dữ liệu đồng bộ |
| NSS | Network Server Sync | Tiến trình vận chuyển dữ liệu đồng bộ trong chế độ đồng bộ |
| NSA | Network Server Async | Tiến trình vận chuyển dữ liệu đồng bộ trong chế độ bất đồng bộ |
| DGMGRL | Data Guard Command Line Interface | Công cụ giao diện dòng lệnh thao tác quản trị môi trường Data Guard |
| DMON | Data Guard Monitor | Tiến trình trong mô hình Broker thực hiện theo dõi tình trạng môi trường Data Guard |
| PMON | Process Monitor | Một trong sáu tiến trình quan trọng giúp Instance hoạt động của CSDL Oracle |
| DGB | Data Guard Broker | Thành phần quản lý môi trường Data Guard |
| BMM | Broker Management Model | Mô hình quản lý môi trường Data Guard |
| RAC | Real Application Cluster | Kiến trúc Oracle Database, trong đó một CSDL có thể được sử dụng bởi nhiều Instance (bản thể, multi-Instance) |
| OCI | Oracle Cloud Infrastructure | Nền tảng hạ tầng điện toán đám mây của Oracle, cung cấp các dịch vụ như IaaS, DaaS, Saas |
| OLTP | Online Transaction Processing | Hệ thống dùng để chuyển xử lý giao dịch, đảm bảo tính ACID của một CSDL |

**DANH MỤC CÁC BẢNG BIỂU**

[Bảng 1 Cấu hình đối số phương thức truyền/xác nhận 18](#_Toc164843417)

[Bảng 2 Tham số cấu hình cho cơ chế Fetch Archive Log 24](#_Toc164843418)

[Bảng 3 So sánh việc sử dụng Broker vào hệ thống 28](#_Toc164843419)

[Bảng 4 Các trường hợp chuyển đổi 35](#_Toc164843420)

[Bảng 5 Thiết lập cấu hình mạng lưới ảo OCI 42](#_Toc164843421)

[Bảng 6 Cấu hình máy chủ ảo cài đặt Oracle Data Guard trên OCI 42](#_Toc164843422)

[Bảng 7 Kết hợp đối số trong VALID\_FOR 50](#_Toc164843423)

[Bảng 8 Các thông tin cần để thiết lập chế độ bảo vệ 58](#_Toc164843424)

[Bảng 9 Các Testcase kiểm thử 72](#_Toc164843425)

**DANH MỤC HÌNH**

[Hình 1 Logo thương hiệu công ty SHB Finance 2](#_Toc164843355)

[Hình 2 Sơ đồ tổ chức của công ty Tài chính SHB Finance 4](#_Toc164843356)

[Hình 3 Minh họa khái niệm chỉ số RPO và RTO 5](#_Toc164843357)

[Hình 4 Minh họa khái niệm mức độ RLO 5](#_Toc164843358)

[Hình 5 Phân chia giải pháp theo khung tham chiếu Oracle MAA 8](#_Toc164843359)

[Hình 6 Kiến trúc tổng quan của Oracle Database 9](#_Toc164843360)

[Hình 7 Kết hợp RMAN, Oracle Secure Backup và sao lưu bằng lệnh hệ thống 12](#_Toc164843361)

[Hình 8 Kiến trúc tổng quan giải pháp Oracle Data Guard 13](#_Toc164843362)

[Hình 9 Tính năng Far Sync trong giải pháp Oracle Data Guard 19](#_Toc164843363)

[Hình 10 Luồng hoạt động của Oracle Data Guard với chế độ Ưu tiên bảo vệ 20](#_Toc164843364)

[Hình 11 Luồng hoạt động của Oracle Data Guard với chế độ Ưu tiên bảo vệ 22](#_Toc164843365)

[Hình 12 Minh họa cơ chế xử lý thiếu trong việc truyền thông tin thay đổi 23](#_Toc164843366)

[Hình 13 Kiến trúc của Oracle Data Guard trước khi có Standby Redo Log Files 25](#_Toc164843367)

[Hình 14 Kiến trúc Oracle Data Guard với tính năng Data Guard Broker 27](#_Toc164843368)

[Hình 15 Mối quan hệ giữa các thành phần trong mô hình Broker 29](#_Toc164843369)

[Hình 16 Giao tiếp các thành phần tại Server-Side trong Broker 30](#_Toc164843370)

[Hình 17 Tổng quan hệ thống xếp hạng tín dụng nội bộ của SHBFinance 31](#_Toc164843371)

[Hình 18 Minh họa bảng quy đổi đối chiếu xếp hạng tín dụng nội bộ 33](#_Toc164843372)

[Hình 19 Phương thức chuyển đổi toàn bộ 35](#_Toc164843373)

[Hình 20 Phương thức chuyển đổi chỉ hệ thống CSDL 35](#_Toc164843374)

[Hình 21 Minh họa hoạt động của DML Redirection 38](#_Toc164843375)

[Hình 22 Kiến trúc tổng quan thực nghiệm giải pháp Data Guard trên OCI 39](#_Toc164843376)

[Hình 23 Minh họa phân cấp giữa Region, AD và FD 41](#_Toc164843377)

[Hình 24 Minh họa phân cấp tên trong hệ thống CSDL cho SHBFinance 43](#_Toc164843378)

[Hình 25 Cấu hình Listener, Local Naming Method tại hai máy chủ CSDL 45](#_Toc164843379)

[Hình 26 Kết quả kiểm tra sau khi bật các chế độ cần thiết 47](#_Toc164843380)

[Hình 27 Dung lượng của các Online Redo Logs có trong CSDL hiện tại 49](#_Toc164843381)

[Hình 28 CSDL thực hiện sao lưu thông qua meta-data được RMAN quản lý 54](#_Toc164843382)

[Hình 29 CSDL dự phòng được nhân bản thông qua RMAN DUPLICATE 54](#_Toc164843383)

[Hình 30 Hoàn thành nhân bản CSDL phụ từ CSDL chính 55](#_Toc164843384)

[Hình 31 So sánh cơ chế Failover thủ công và tự động bằng Fast-Start Failover 59](#_Toc164843385)

[Hình 32 Minh họa máy chủ thứ ba chứa Observer trong môi trường Data Guard, dựa trên nền tảng điện toán đám mây OCI 60](#_Toc164843386)

[Hình 33 Minh họa tương tác của tiến trình DMON trong môi trường Broker Configuration 61](#_Toc164843387)

[Hình 34 Minh họa việc thiết lập Observer tại Windows 66](#_Toc164843388)

[Hình 35 CSDL chính bị mất ghi (nguồn: dbvisit) 73](#_Toc164843389)

[Hình 36 Áp dụng thay đổi tại CSDL dự phòng (nguồn: dbvisit) 74](#_Toc164843390)

[Hình 37 Tiếp tục thay đổi thông tin với block cũ (nguồn: dbvisit) 74](#_Toc164843391)

[Hình 38 Áp dụng sau khi xảy ra quá trình mất ghi ở CSDL dự phòng 75](#_Toc164843392)

# **LỜI MỞ ĐẦU**

1. Lý do chọn đề tài

2. Mục đích nghiên cứu

3. Đối tượng và phạm vi nghiên cứu

4. Phương pháp nghiên cứu

5. Bố cục đề tài

1. TỔNG QUAN
   1. Khái quát về Công ty Tài chính Ngân hàng TMCP Sài Gòn – Hà Nội SHB Finance
      1. Giới thiệu chung về SHB Finance

Công ty Tài chính Ngân hàng TMCP Sài Gòn – Hà Nội SHB Finance là một công ty hoạt động và cung cấp dịch vụ trong lĩnh vực Tài chính tiêu dùng phổ biến cho người dân và doanh nghiệp như cho vay nâng cấp thiết bị, sửa sang nhà cửa, mua xe hơi, du học, chăm sóc sức khỏe, bảo hiểm. “Công ty Tài Chính TNHH Ngân hàng TMCP Sài Gòn - Hà Nội (SHBFinance), được thành lập ngày 12/12/2016 (Mã số doanh nghiệp: 0107779290) với hình thức pháp lý ban đầu là Công ty Tài Chính TNHH MTV Ngân hàng TMCP Sài Gòn - Hà Nội. Ngày 2/6/2023, Ngân hàng Nhà nước Việt Nam đưa ra Quyết định chuyển đổi Công ty Tài chính TNHH MTV Ngân hàng TMCP Sài Gòn – Hà Nội thành Công ty Tài chính TNHH Ngân hàng TMCP Sài Gòn – Hà Nội.” (“Giới Thiệu | SHBFinance,” 2024).

* Tên công ty: Công ty Tài Chính TNHH Ngân hàng TMCP Sài Gòn - Hà Nội
* Tên tiếng Anh: SHB Consumer Finance Company Limited
* Tên gọi tắt: SHB Finance
* Địa chỉ: Tầng 6, Gelex Tower, 52 Lê Đại Hành, phường Lê Đại Hành, Quận Hai Bà Trưng, Hà Nội
* Điện thoại: 024 7109 8888
* Website: <https://www.shbfinance.com.vn/>
* Logo**:**



Hình 1 Logo thương hiệu công ty SHB Finance

Tiền thân là công ty Tài chính Vinaconex thuộc tập đoàn Viettel, hiện tại công ty hoạt động theo loại hình TNHH từ hai thành viên trở lên. Hiện tại, đang có hai tổ chức sở hữu 50% trên tổng số vốn điều lệ 1000 tỷ mỗi bên là Ngân hàng TMCP Sài Gòn – Hà Nội (SHB) và Ngân hàng TNHH Đại chúng Ayudhya (Krungsri) của Thái Lan. “Ngân hàng Sài Gòn – Hà Nội (SHB) vốn được biết đến thuộc top 5 ngân hàng tư nhân lớn nhất Việt Nam với 30 năm hình thành và phát triển. Tập đoàn tài chính Krungsri có 75 năm hình thành và phát triển, là thành viên chiến lược của Tập đoàn tài chính Nhật Bản MUFG, hiện mở rộng sự hiện diện tại 5 quốc gia trong khu vực”. (“Giới Thiệu | SHBFinance,” 2024)

* + 1. Thành tựu, mục tiêu, tầm nhìn, sứ mệnh

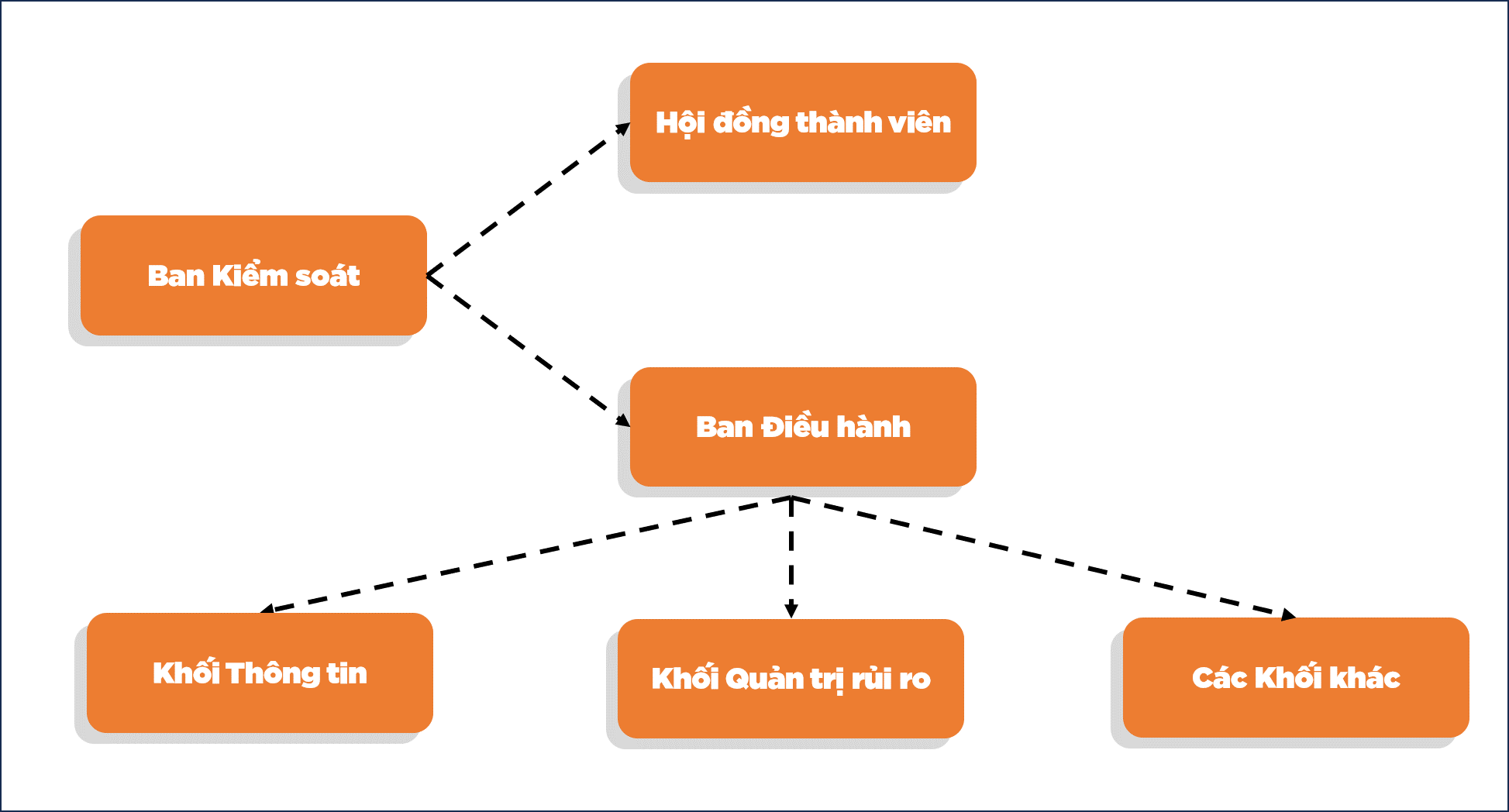
SHB Finance không chỉ có nhiều điểm nổi bật về việc cung cấp dịch vụ trong lĩnh vực hoạt động là Tài chính tiêu dùng, mà còn tạo công ăn việc làm cho gần 8000 người dân tại Việt Nam. Cụ thể là những ngành nghề như Bán hàng/Kinh doanh, Dịch vụ Ngân hàng và Tài chính cá nhân, Luật/Pháp lý, Ngân hàng, Quản trị rủi ro, Tài chính/Đầu tư. Với việc thấu hiểu rằng chăm sóc nhân viên tốt chính là cách để quảng bá sản phẩm tốt, SHB Finance cũng xây dựng chính sách đãi ngộ xứng đáng thông qua việc đào tạo, cung cấp bảo hiểm và cho vay vốn ưu đãi đối với CBNV. Thông qua việc điều hành hiệu quả, lấy khách hàng làm trung tâm, SHB Finance cũng đạt được một số thành tựu nhất định như:

* Top 6 Nhà tuyển dụng được yêu thích trong năm 2018
* Top 1 Nhà tuyển dụng được yêu thích nhất ngành nghề lĩnh vực Finance – Banking năm 2019
* Được Moody’s[[1]](#footnote-1) xếp hạng tín nhiệm lần đầu hạng Ba3 năm 2019
* Top 8 công ty tài chính tiêu dùng lớn nhất Việt Nam năm 2023
* Duy trì nợ xấu ở mức ổn định
* Phục vụ cho khoảng 300 nghìn hộ gia đình và 200 nghìn khách hàng

Cùng với đó là tầm nhìn “Trở thành Công ty Tài chính Thuận tiện và Tin cậy với người dân Việt Nam” và sứ mệnh “Cung cấp các giải pháp Tài chính tiêu dùng thông minh, dễ tiếp cận cho mọi người dân Việt”. Ngoài ra, SHB Finance cũng đưa ra bộ quy tắc ứng xử độc đáo, dựa trên năm chữ cái viết tắt “SHBFC” bao gồm: S – Smart, H – Honest, B – Brave, F – Friendly và C – Cooperative.

* + 1. Cơ cấu tổ chức

Cơ cấu tổ chức của công ty Tài chính SHB Finance là cơ cấu của công ty TNHH từ hai thành viên trở lên. Bao gồm: Ban Kiểm soát sẽ giám sát, miễn/bổ nhiệm đối với Hội đồng thành viên và giám sát với Ban Điều hành. Hội đồng thành viên sẽ bầu chọn Ban Điều hành và Ban Điều hành sẽ trực tiếp quản lý các Khối.

****

Hình 2 Sơ đồ tổ chức của công ty Tài chính SHB Finance

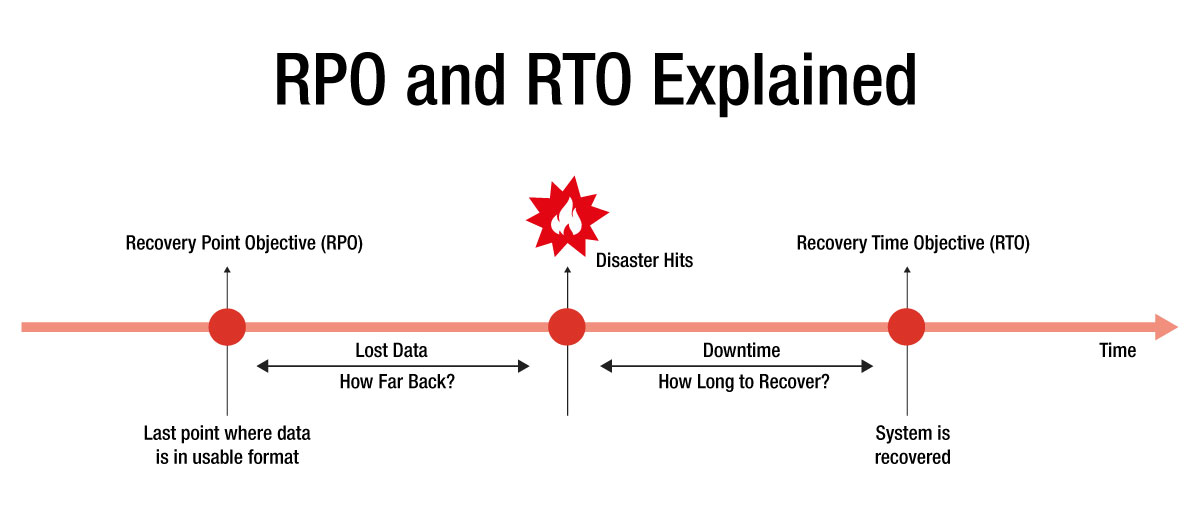
* 1. Mô tả bài toán
     1. Thực trạng

Trong hệ thống có số lượng giao dịch lớn, liên tục của các ngành, đặc biệt là ngành Tài chính – Ngân hàng, việc lựa chọn hệ quản trị cơ sở dữ liệu phù hợp là một phần quan trọng của việc xây dựng và duy trì ổn định các lô-gic, nghiệp vụ và nhận/trả yêu cầu của người dùng. Cơ sở luôn phải đảm bảo 4 đặc điểm – A (Atomicity, tính toàn vẹn), C (Consistency, tính nhất quán), I (Isolation, tính độc lập) và D (Durability, tính bền vững).

Oracle Database – hệ quản trị cơ sở dữ liệu quan hệ với bề dày lịch sử hơn 46 năm, hỗ trợ đa dạng nền tảng có thứ hạng đầu tính tới thời điểm tháng 9/2023 trên thị phần thế giới (“Most Popular Database Management Systems 2023 | Statista,” 2023). Với tính năng đa dạng, hiệu năng nổi trội và được chứng minh uy tín qua nhiều doanh nghiệp lớn, Công ty Tài chính tiêu dùng SHBFinance đã lựa chọn Oracle Database và sử dụng làm nền tảng cho các hệ thống tầng hỗ trợ nghiệp vụ, trong đó, có hệ thống Xếp hạng tín dụng nội bộ.

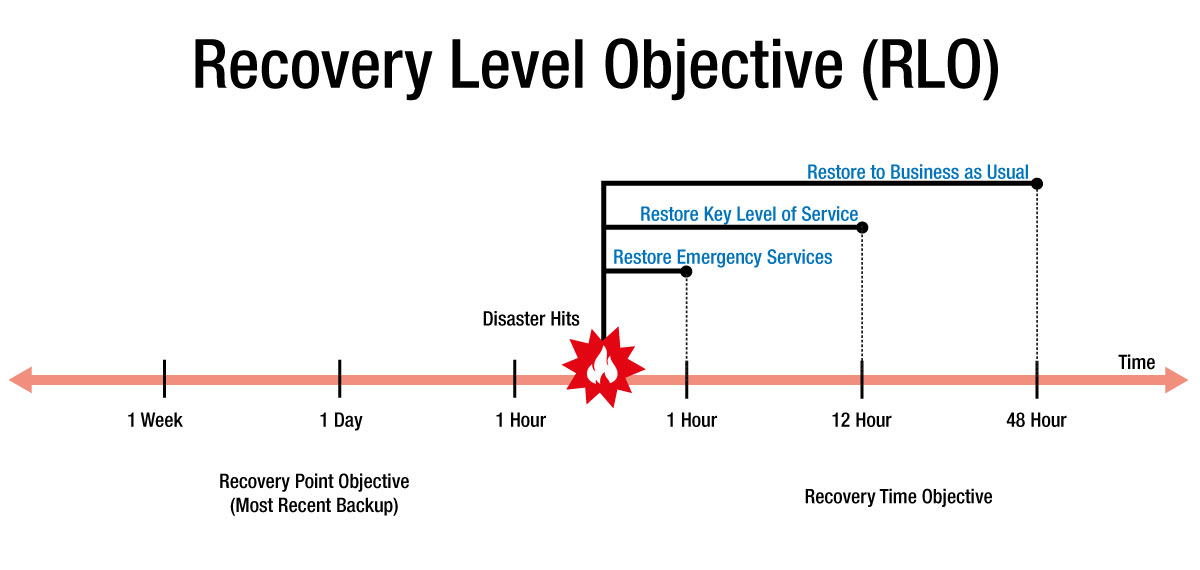
Trong SHB Finance, cũng như trong nhiều tổ chức tài chính khác, đều cần định hình chiến lược và quy trình đối phó với sự cố, cụ thể là xây dựng kế hoạch đảm bảo kinh doanh không gián đoạn TMCP (Business Continuity Plan) và kế hoạch khôi phục do sự cố nghiêm trọng gây ra DRP (Disaster Recovery Plan). Các kế hoạch đã nêu được ràng buộc bởi sự cân đối ngân sách (chi phí ban đầu, băng thông, lưu trữ) và mục tiêu với những chỉ số/mức độ như Recovery Level Objective (RLO), Recovery Time Object (RTO) và Recovery Point Object (RPO). Đây là các chỉ số quan trọng, nhưng không thể lúc nào cũng đảm bảo rằng các chỉ số này luôn hoàn hảo và vượt quá ngân sách của tổ chức, vì vậy, tổ chức cần đánh đổi giữa những lợi ích họ đạt được với chi phí.

* **Recovery Time Objective (RTO):** là chỉ số thể hiện thời gian tối đa mà một tổ chức đề ra nhằm thực hiện khôi phục hạ tầng CNTT và các hoạt động nghiệp vụ sau sự cố nghiêm trọng, phù hợp với nhu cầu của doanh nghiệp đó. Ví dụ, nếu RTO là 2 giờ, có nghĩa là tổ chức cần khôi phục hệ thống trong vòng nhiều nhất là 2 giờ.
* **Recovery Point Objective (RPO):** là chỉ số thể hiện mức độ tối đa dữ liệu mất mát trong quá trình khôi phục sau sự cố mà tổ chức đề ra. Được đo bằng thời điểm dữ liệu được sao lưu lần cuối và thời điểm sự cố xảy ra. Ví dụ, nếu RPO là 1 giờ, có nghĩa là tổ chức chấp nhận mất không quá 1 giờ dữ liệu sinh ra.



Hình 3 Minh họa khái niệm chỉ số RPO và RTO

* **Recovery Level Objective:** thang đo mức độ cần khôi phục dựa trên sự quan trọng của hệ thống. Được chia làm 03 mức độ: Immediate/Emergency, Near-Time, Deffered, với mức độ quan trọng giảm dần từ trái sang phải, từ vài giờ cho đến vài ngày/tuần hoặc hơn.



Hình 4 Minh họa khái niệm mức độ RLO

Để một hệ thống đạt được độ tin cậy cao, giảm thiệu rủi ro do gián đoạn và gây ảnh hưởng đến doanh thu, phải tiến tới việc đạt được giảm RTO, tăng RPO. Hiện tại, với cơ sở dữ liệu Oracle mà SHBFinance sử dụng cho các hệ thống hỗ trợ, đang được sử dụng các giải pháp sao lưu như công cụ chính Recovery Manager (RMAN), sử dụng Oracle Flashback và Oracle Secure Backup (OSB) để tăng tính bảo mật cho bản sao lưu trong môi trường phân tán.

Với công cụ sao lưu chính RMAN, có thể sao lưu từ toàn bộ hệ thống cho đến một phần hệ thống ở cấp độ vật lý. Tuy nhiên, thời gian khôi phục của RMAN được chỉ rõ lên tới hàng giờ cho đến hàng ngày, chưa đáp ứng được việc giảm chỉ số MTTR. Thực tế, có một số bài báo nghiên cứu khoa học cũng chỉ ra một số điểm hạn chế của giải pháp này như:

* **Hạn chế về băng thông:** RMAN gặp hạn chế khi không thể sao lưu các tập dữ liệu lớn tới một hệ thống nằm ngoại mạng. Khi cần sao lưu, sẽ tiêu tốn rất nhiều tài nguyên mạng cũng như gặp vấn đề về thời gian, ảnh hưởng hiệu suất và các “thời điểm vàng” của SHBFinance
* **Hạn chế về thời gian khôi phục:** Trong trường hợp một thảm họa làm cho trung tâm dữ liệu bị lỗi, việc khôi phục dữ liệu từ các bản sao lưu RMAN gặp nhiều khó khăn. Nếu cả trung tâm dữ liệu và bản sao lưu đều bị mất, quá trình khôi phục dữ liệu sẽ gặp nhiều vấn đề và có thể tốn thời gian đáng kể (Yu et al.)
  + 1. Thách thức

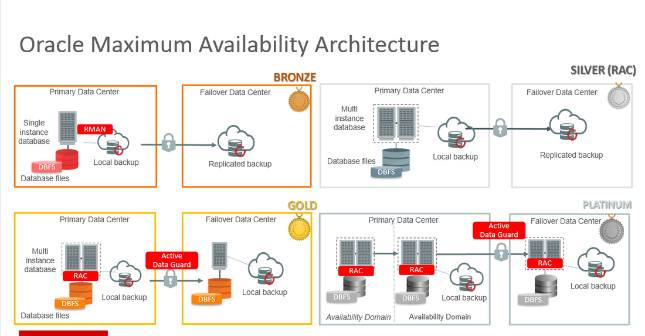
Khi hệ thống của một doanh nghiệp, đặc biệt là hoạt động về lĩnh vực Tài chính – Ngân hàng gặp vấn đề, như sụp đổ hoặc gián đoạn, không chỉ gây ảnh hưởng đến hoạt động kinh doanh hàng ngày mà còn mang theo nhiều thách thức lớn. Mất mát dữ liệu, không thể truy cập thông tin quan trọng để đưa ra quyết định kịp thời, và sự gián đoạn trong dịch vụ có thể dẫn đến mất lòng tin từ phía khách hàng, gây tổn thương về uy tín và doanh số của công ty. Hơn nữa, việc phục hồi hệ thống có thể đòi hỏi nhiều thời gian và chi phí, và không có kế hoạch hồi phục hiệu quả có thể làm gia tăng rủi ro và ảnh hưởng đến sức mạnh cạnh tranh của công ty trong ngành. Hơn hết, trong một môi trường kinh doanh đầy cạnh tranh như ngành tài chính, việc duy trì sự ổn định của hệ thống là một yếu tố quan trọng để bảo vệ uy tín và sự phát triển của doanh nghiệp.

* + 1. Giá trị mang lại

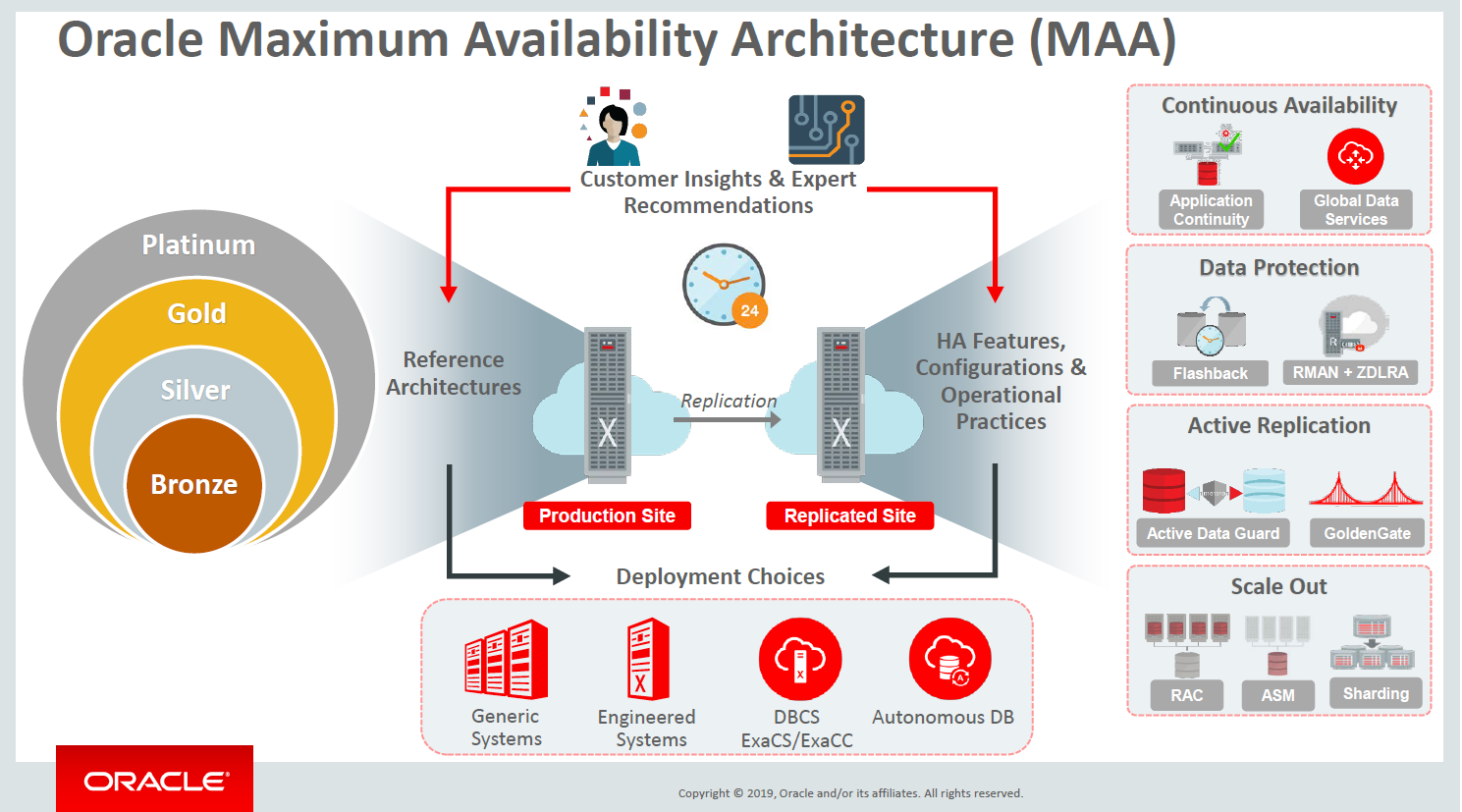
Để giải quyết thách thức về sao lưu và khôi phục dữ liệu tại SHBFinance, một quyết định cần được đưa ra với mục tiêu chính là đảm bảo rằng dữ liệu từ cơ sở dữ liệu luôn sẵn sàng cho hoạt động kinh doanh một cách nhanh chóng và đáng tin cậy. Điều này đòi hỏi một giải pháp hiệu quả có khả năng sao lưu và phục hồi dữ liệu tự động, giảm thiểu thời gian gián đoạn và mất mát dữ liệu. Một số giá trị lợi ích mà việc này mang lại có thể kể đến như:

* Khôi phục dữ liệu kịp thời: khôi phục dữ liệu nhanh chóng sau sự cố, giảm thiểu thời gian gián đoạn hoạt động kinh doanh và đảm bảo tính liên tục của các quy trình hoạt động.
* Tiến tới chuẩn Basel cao hơn trong lĩnh vực Tài chính – Ngân hàng: Bằng cách cung cấp khả năng khôi phục dữ liệu nhanh chóng và hiệu quả, giải pháp Data Guard giúp tổ chức tuân thủ các quy định và tiêu chuẩn của Basel liên quan đến quản lý rủi ro và bảo vệ dữ liệu, nâng cao uy tín và sự tin cậy của tổ chức trong mắt khách hàng và cơ quan giám sát.
* Tiết kiệm chi phí và thời gian: giảm thiểu chi phí và thời gian phục hồi sau sự cố, loại bỏ nhu cầu phải tiêu tốn nguồn lực nhân sự và vật lý để thực hiện các công việc sao lưu và khôi phục thủ công. Điều này giúp tổ chức tối ưu hóa hiệu suất và tài nguyên của họ, tập trung vào các hoạt động kinh doanh chính.
  + 1. Hướng giải quyết

Để giải quyết bài toán đã nêu, cần thay đổi loại hình sao lưu – khôi phục mà SHBFinance đang sử dụng. Tham chiếu với khung giải pháp *Oracle Maximum Availability Architecture (MAA)* của Oracle, hiện trạng về phần trung tâm cơ sở dữ liệu dự phòng của hệ thống Xếp hạng tín dụng nội bộ của SHBFinance chỉ đạt ở mức Silver trở xuống. Với mức Gold trở lên, giải pháp Oracle Data Guard (ODG, Active Data Guard là tính năng, có từ phiên bản 11g) được giới thiệu rằng giúp quá trình vận hành của doanh nghiệp được đảm bảo liên tục, bằng cách giảm thiểu các sự cố không có chủ đích, gây ra thời gian dừng lâu hoặc các nghiệp vụ có chủ đích nhằm nâng cấp hệ thống thông qua việc chuyển đổi quá trình hoạt động, vận hành giữa CSDL chính (Primary) sang CSDL phụ (Standby). Vì vậy, đây chính là giải pháp thích hợp để triển khai cho hệ thống Xếp hạng tín dụng nội bộ của SHBFinance.



Hình 5 Khung tham chiếu giải pháp Oracle MAA



Hình 6 Phân chia giải pháp theo khung tham chiếu Oracle MAA

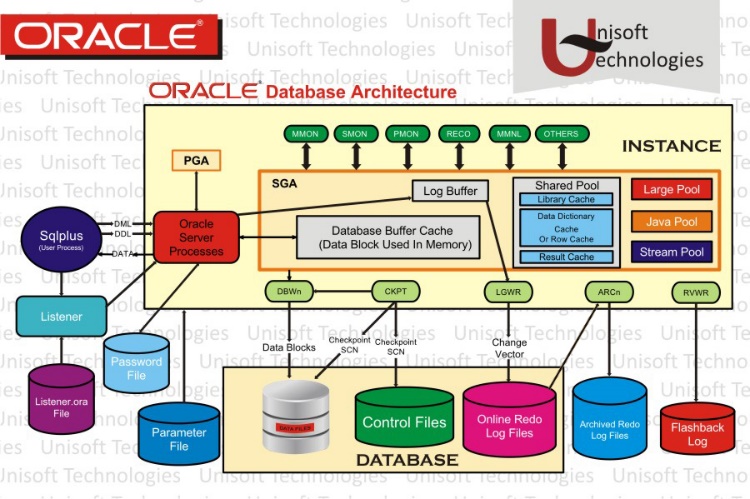
Giải pháp Oracle Data Guard có các đặc điểm như sau:

* Sao lưu dữ liệu theo thời gian thực: Oracle Data Guard cho phép sao lưu dữ liệu từ một cơ sở dữ liệu gốc (primary) sang một hoặc nhiều cơ sở dữ liệu dự phòng (standby) một cách đồng bộ và thời gian thực, đảm bảo rằng các bản sao luôn được cập nhật với dữ liệu mới nhất, đạt được tính toàn vẹn và tùy theo cấu hình hệ thống.
* Ứng biến kịp thời: Trong trường hợp sự cố xảy ra ở CSDL Primary, Oracle Data Guard tự động chuyển đổi sang một trong các CSDL Standby, giúp giảm thiểu thời gian chết hệ thống và tiếp tục phục vụ người dùng một cách liền mạch. Ngoài ra, ODG cho phép tạo dựng lên tới 30 CSDL Standby, giúp độ sẵn sàng của dữ liệu cao hơn.
* Phục vụ nhiều mục đích: Oracle Data Guard hỗ trợ nhiều loại CSDL ở chế độ dự phòng (gọi là standby), bao gồm physical standby, logical standby và snapshot standby, để phù hợp với nhu cầu cụ thể của doanh nghiệp và môi trường.
* Quản lý tự động: Oracle Data Guard cung cấp các công cụ quản lý tự động để giảm thiểu sự can thiệp của người quản trị, bao gồm cả việc tự động thực hiện các hoạt động như sao lưu, kiểm tra tính sẵn sàng và ghi nhật ký.
* Giảm tải cho hệ thống chính: kể từ phiên bản Oracle 11g trở đi, ODG cho phép các hệ thống máy người dùng truy vấn trực tiếp các dữ liệu được cập nhật theo thời gian ở chế độ chỉ đọc, và cũng cho phép sao lưu dữ liệu có trên CSDL Standby. Điều này giúp hệ thống CSDL Primary chỉ có nhiệm vụ vận hành, không cần phải tải thêm những nhiệm vụ phụ khác.

Nhờ một số tính năng liệt kê của Oracle Data Guard, đây là giải pháp nổi trội hơn các công cụ sao lưu nêu trước đang được dùng trong SHBFinance. Kết luận, đây là một giải pháp phù hợp với thực trạng cho cơ sở dữ liệu của hệ thống Xếp hạng tín dụng nội bộ của SHBFinacne.

* 1. Giới thiệu về Oracle Data Guard
     1. Kiến trúc Oracle Database

Tổng quan, kiến trúc của hệ quản trị cơ sở dữ liệu Oracle gồm hai phần chính,Instance – giao tiếp trực tiếp với người dùng, là cầu nối cho người dùng tương tác với Database – bao gồm các tập tin vật lý, lưu trữ dưới ổ đĩa cứng, được cấu thành bởi File Systems (hệ thống tệp tin).



Hình 7 Kiến trúc tổng quan của Oracle Database

Trong Instance, chứa hai cấu trúc thành phần lớn:

* *Cấu trúc tiến trình (Process Structure)*: tập hợp các tiến trình thực hiện trong bộ nhớ RAM, là yếu tố quan trọng trong việc điều tiết dữ liệu từ các tệp tin vật lý đến bộ nhớ, cũng như cho việc sao lưu và phục hồi. Ngoài các tiến trình được nêu gọi là Background Processes, thì cũng có một số tiến trình khác như Server Processes, thực hiện các hoạt động liên quan đến phản hồi yêu cầu từ hệ thống máy người dùng cuối
* *Cấu trúc bộ nhớ (Memory Structure)*: vùng nhớ được hệ thống cấp phát khi khởi động Instance, bao gồm vùng nhớ dùng chung (System Global Area - SGA) và vùng nhớ dùng riêng (Program Global Area - PGA). Với vùng nhớ SGA, toàn bộ hoạt động liên quan tới hệ thống điều tiết luồng dữ liệu sẽ hoạt động tại đây. Vùng nhớ PGA được cấp phát riêng cho mỗi hệ thống máy người dùng cuối khi họ kết nối tới (nếu được thiết lập cấu hình Dedicated Server), chứa thông tin về thông tin kết nối tới cơ sở dữ liệu (CSDL), con trỏ trong việc dùng truy vấn và nhiều thông tin khác

Với Database, chưa cấu trúc thành phần *Cấu trúc lưu trữ (Storage Structure)*. Tập hợp các tập tin quan trọng để lưu trữ dữ liệu và khởi động CSDL. Các tập tin này có thể kể đến một số như sau:

* *Tập tin điều khiển (Control Files):* đây là tập tin quan trọng, chứa các thông tin về địa điểm của các tập tin dữ liệu và tập tin sao lưu. Ngoài ra, chúng cũng chứa thông tin meta-data về điểm khôi phục như System Change Number (SCN)
* *Tập tin dữ liệu (Data Files):* lưu trữ toàn bộ dữ liệu của CSDL do người dùng sinh ra, cũng như bao gồm cả những đối tượng của CSDL, người dùng phải thông qua Instance thì mới có thể đọc/ghi chính xác trên tập tin này
* *Tập tin lưu trữ thay đổi (Redo Log Files):* lưu trữ những thông tin thay đổi về dữ liệu, được gọi là những véc-tơ thay đổi (information vector changes). Bất kể là những thay đổi được ghi nhận (COMMIT) hay chưa ghi nhận (UNCOMMITTED) đểu được lưu trữ xuống đây, cùng với thông tin về SCN nhằm mục đích có thể khôi phục CSDL sau những trường hợp lỗi hệ thống (ROLLFORWARD)
* *Tập tin mật khẩu (Password File):* lưu trữ thông tin được mã hóa về mật khẩu của tài khoản người dùng quản trị hệ thống với các đặc quyền sâu về quản trị hệ thống như SYS, SYSTEM, SYSOPER
* *Tập tin cấu hình tham số (Parameters File):* lưu trữ thông tin về tham số khởi động, cấp phát bộ nhớ cho Instance. Thường tập tin này sẽ có hai dạng, văn bản thô đơn thuần (text – init.ora) và dạng nhị phân máy đọc (spfile.ora)

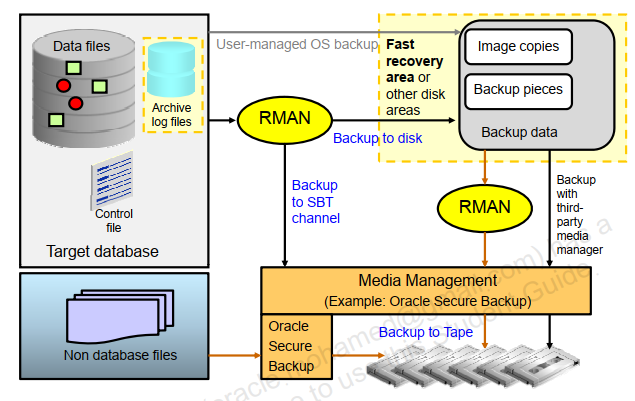
Để khởi động CSDL, cần vai trò của người quản trị thực hiện việc này, khi đăng nhập vào, hệ thống sẽ xác thực thông tin đăng nhập của người quản trị thông qua tệp tin mật khẩu. Khi khởi động CSDL với lệnh “startup” được yêu cầu từ SQL\*Plus hoặc bất kỳ công cụ nào dành cho người quản trị truy cập được vào môi trường hệ thống trạng thái IDLE Instance, CSDL sẽ khởi động theo các pha sau đây:

* Từ SHUTDOWN đến NOMOUNT: thực hiện tìm tệp tin cấu hình tham số và kích hoạt hệ thống cấp phát bộ nhớ SGA cho Instance, khởi động các Background Processes
* Từ NOMOUNT đến MOUNT: thực hiện tìm tệp tin điều khiển được xác định trong tệp tin cấu hình tham số. Từ đây, tệp tin điều khiển cung cấp cho hệ thống vị trí của các tệp tin dữ liệu cũng như các tệp tin lưu trữ thay đổi
* Từ NOMOUNT đến OPEN: tệp tin dữ liệu và tệp tin lưu trữ thay đổi sẵn sàng cho việc lưu trữ, hệ thống máy người dùng đã có thể kết nối tới và tương tác với CSDL. Các tiến trình Background Processes sẽ thực hiện kiểm tra tính đồng bộ giữa hai tệp tin và khôi phục nếu có lỗi
  + 1. Sao lưu với công cụ Recovery Manager (RMAN)

Recovery Manager (RMAN) là một công cụ quản lý sao lưu và khôi phục dữ liệu tối ưu của Oracle, được thiết kế đặc biệt để hỗ trợ việc quản lý và bảo vệ dữ liệu trong các cơ sở dữ liệu Oracle. Mục đích cơ bản của RMAN là cung cấp các tính năng và công cụ cho việc thực hiện các tác vụ sao lưu và khôi phục dữ liệu một cách linh hoạt, hiệu quả và an toàn.

Đặc điểm nổi bật chính của RMAN là ngoài việc sao lưu khi CSDL đang tắt (Offline/Consistent/Cold), thì cũng có thể sao lưu ngay cả lúc CSDL đang hoạt động (Online/Inconsistent/Hot). Ngoài ra, RMAN cũng cung cấp tính năng sao lưu một phần, bao gồm chỉ những thay đổi kể từ bản sao lưu trước đó. Tương tự đối với việc khôi phục, công cụ cho phép khôi phục một phần hoặc toàn phần.

Các thành phần mà RMAN sao lưu bao gồm: Data Files, Control Files, Archived Redo Log, Parameters File. Đối với mức độ block – đơn vị nhỏ nhất trong kiến trúc lưu trữ vật lý, khi sao lưu, RMAN tự động kiểm tra những block rỗng và sẽ bỏ qua block này. Công cụ này cũng được tích hợp lên giao diện quản trị dành cho doanh nghiệp là Oracle Enterprise Manager và cũng có thể sử dụng bởi lệnh SQL để tạo. Ngoài ra, chúng cũng được tích hợp thêm thành phần Oracle Secure Backup để mã hóa và sao lưu ra ổ đĩa dạng băng từ (tape) hoặc sao lưu lên đám mây một cách bảo mật, an toàn.



Hình 8 Kết hợp RMAN, Oracle Secure Backup và sao lưu bằng lệnh hệ thống

* + 1. Khái niệm, kiến trúc và điểm chính của Oracle Data Guard

***Khái niệm***

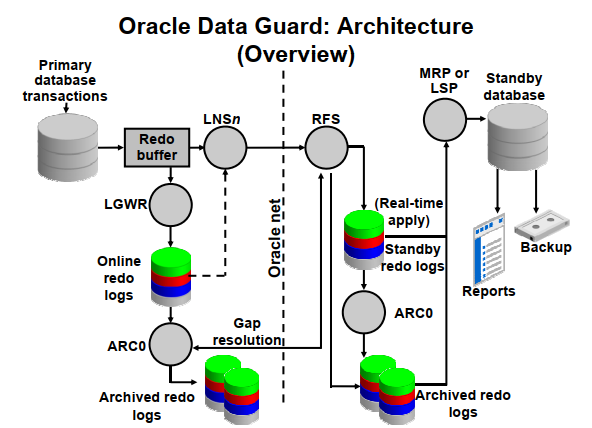
Giải pháp Oracle Data Guard được giới thiệu và triển khai kể từ phiên bản Oracle Database 7, kể từ đây, Oracle sử dụng thuật ngữ “Standby Database” để chỉ CSDL dự phòng. Tuy nhiên, phiên bản này có một hạn chế - các tệp tin ghi log được truyền tự động để sao lưu lại không thể lưu trữ (archive), nếu muốn, phải làm thao tác thủ công là viết mã hoặc lập trình để sao lưu. Kể từ phiên bản Oracle Database 8i, 9i, 9i R2 cho đến 10g thì các tính năng được cải thiện, đáp ứng nhu cầu gần như hoàn hảo của mọi doanh nghiệp, đó là không chỉ về truyền tải thời gian thực thông tin thay đổi và truy vấn trên CSDL Standby mà còn cho phép hành động ghi/đọc, sao lưu CSDL Standby và nhiều tính năng khác.

Trong các giải pháp phục hồi sau sự cố, Oracle Data Guard (ODG) là một công nghệ được đánh giá cao trong việc đảm bảo tính sẵn sàng và liên tục của CSDL Oracle. ODG được xây dựng và tích hợp trên CSDL Oracle, gồm nhiều mô-đun chức năng như quản lý, giám sát, tự động hóa quy trình, duy trì một hoặc nhiều CSDL Standby nhằm mục đích bảo vệ quy trình vận hành của doanh nghiệp khỏi sự cố, thảm họa, lỗi và các nghiệp vụ có chủ đích. CSDL Standby duy trì sự ổn định này bằng cách trở thành bản sao y của CSDL chính (Primary), phản ánh cùng một hiện trạng, trạng thái; CSDL Standby có thể đặt cách xa trung tâm dữ liệu của CSDL Primary để tăng thêm độ an toàn đề phòng trường hợp thiên tai ngay tại địa điểm đặt CSDL Primary. Khi CSDL Primary gặp sự cố, công nghệ ODG sẽ chuyển đổi sự vận hành (switch role) từ CSDL Primary – đang bị ảnh hưởng sang CSDL Standby, CSDL Standby sẽ đảm nhận vai trò thành CSDL Primary, từ đây giảm thiểu khỏi việc dữ liệu bị mất mát.

***Kiến trúc***

Trong kiến trúc của giải pháp ODG bao gồm 01 CSDL Primary và lên tới 30 CSDL Standby. Các hệ thống CSDL liên kết và giao tiếp với nhau thông qua một dịch vụ gọi là Oracle Network Service trong môi trường mạng của Oracle, các CSDL này có thể đặt cách xa nhau về mặt địa lý. Khoảng cách đặt hệ thống CSDL không phải là vấn đề, người triển khai giải pháp chỉ cần đảm bảo rằng các CSDL có thể giao tiếp với nhau thông qua môi trường mạng. Có nhiều cách triển khai CSDL dự phòng, có thể trong cùng một Server, xuất hiện đồng thời Primary và Standby, nhưng an toàn hơn thì cũng có thể có nhiều Server và mỗi CSDL thuộc về Server riêng.

CSDL Standby là bản sao y của CSDL Primary trong từng giao dịch (transactions). CSDL Standby được tạo dựng từ một bản sao lưu ban đầu của CSDL Primary. Ý tưởng của giải pháp ODG là tận dụng việc thông tin thay đổi được lưu dưới dạng véc-tơ có trong tệp tin lưu trữ thay đổi (redo log foles) để giữ cho CSDL Standby luôn được đồng bộ hóa nhờ việc chuyển và áp dụng những thay đổi đó của CSDL Primary.



Hình 9 Kiến trúc tổng quan giải pháp Oracle Data Guard

Trong kiến trúc giải pháp ODG sử dụng rất nhiều tiến trình phụ trợ, nhằm mục đích tự động hóa việc sao lưu và chuyển đổi vai trò. Có một số tiến trình thuộc về cấu trúc tiến trình chung của Oracle Database, và có một số tiến trình chỉ xuất hiện khi triển khai giải pháp ODG. Có thể liệt kê các thành phần và tiến trình khi phân loại theo Oracle Database so với giải pháp ODG như sau:

* Thành phần/Tiến trình chung trong Oracle Database
  + Redo Buffer: thành phần - bộ đệm lưu trữ thông tin véc-tơ thay đổi, các thông tin được gom nhóm và chuyển xuống Online Redo Logs theo từng đợt nhằm giảm thiểu việc I/O hệ thống
  + Online Redo Logs: thành phần - lưu trữ thông tin véc-tơ thay đổi trên đĩa, bao gồm cả COMMITED và UNCOMMITED TRANSACTION
  + Archived Redo Logs: thành phần – lưu trữ bản sao của Online Redo Logs mỗi khi có hiện tượng Log Switch xảy ra (khi một Online Redo Log Groups đã đầy các thông tin véc-tơ thay đổi, chuyển sang Groups khác)
  + Log Writer (LGWr): tiến trình – đưa các nhóm thông tin véc-tơ thay đổi từ Redo Buffer và ghi xuống Online Redo Logs ở đĩa
  + Archiver (ARCn): tiến trình – sao chép tệp tin lưu trữ thông tin thay đổi của Online Redo Log Groups khi xảy ra Log Switch
* Thành phần/Tiến trình riêng của giải pháp ODG
  + Standby Redo Logs: thành phần – bộ đệm lưu trữ thông tin véc-tơ thay đổi, nhưng là của CSDL Standby
  + Log Writer Network Server (LNSs): tiến trình – nhận thông tin véc-tơ thay đổi và chuyển cho CSDL Standby
  + Remote File Server (RFS): tiến trình – nhận thông tin véc-tơ thay đổi từ LNSs
  + Managed Recovery Process/Logical Standby Process (MRP/LSP): tiến trình áp dụng các thay đổi từ Standby Redo Log vào dữ liệu trên CSDL Standby. Tiến trình MRP cho CSDL dạng Physical và LSP sử dụng cho CSDL dạng Logical
    1. Loại hình bảo vệ

Giải pháp ODG cung cấp nhiều loại hình bảo vệ, mỗi loại hình có đặc điểm khác nhau, linh hoạt và phù hợp với nhu cầu hiện tại của doanh nghiệp. Có thể phân loại các loại hình theo hai góc độ: về loại hình CSDL Standby và về mức độ bảo vệ trong cơ chế truyền/đồng bộ hóa thông tin véc-tơ thay đổi giữa các CSDL.

***Phân loại theo loại hình CSDL Standby****:*

* *CSDL Standby vật lý (Physical):*
  + Là sao y của CSDL Primary có cấu trúc File Systems và dữ liệu giống với CSDL Primary
  + Được đồng bộ hóa với CSDL Primary thông qua việc áp dụng dữ liệu thay đổi (redo data) từ CSDL Primary
  + Cho phép thực hiện đồng thời tác vụ trả kết quả truy vấn cũng như việc nhận và áp dụng dữ liệu thay đổi vào CSDL. Chỉ mở CSDL ở chế độ chỉ đọc (read only)
* *CSDL Standby lô-gic (Logical):*
  + Chỉ sao y với CSDL Primary ở mức độ lô-gic, bộ nhớ vật lý có thể sử dụng các tính năng khác như Oracle Automatic Storage Management (ASM) để quản lý tập tin khác với File Systems theo mặc định. Ngoài ra, cũng sẽ thiếu một số cấu trúc vật lý như index, view.
  + Được đồng bộ hóa với CSDL Primary thông qua việc nhận, chuyển hóa dữ liệu thay đổi thành SQL và thực thi trên CSDL Standby để áp dụng thay đổi. Điều này được thực hiện nhờ công cụ phân tích tệp tin ghi trữ logs là LogMiner
  + Cho phép thực hiện đồng thời các tác vụ trả kết quả truy vấn, áp dụng dữ liệu thay đổi vào CSDL, đặc biệt hơn là cho phép chỉnh sửa đối với các bảng, đối tượng không nằm trong vùng được áp dụng thay đổi. Mở CSDL ở chế độ đọc/ghi (Read/Write)
* *CSDL Standby Snapshot:*
  + Là loại sao y CSDL Primary, được chuyển từ CSDL Standby Physical sang
  + Cho phép thực hiện đọc/ghi trên toàn bộ cơ sở dữ liệu với mục đích kiểm thử
  + Sẽ không nhận và áp dụng các thông tin thay đổi
  + Các thay đổi sẽ bị ROLLBACK lại nếu như chuyển về CSDL Standby Physical

***Phân loại theo chế độ bảo vệ***

Khi đi đến phân loại theo chế độ bảo vệ, cấu hình của các chế độ phụ thuộc vào các đối số được cài đặt (cụ thể là trong tham số *LOG\_ARCHIVE\_DEST\_n*), có 04 đối số chính như sau:

* SYNC: Xác nhận các redo data được gửi sang CSDL Standby thành công trước khi giao dịch gây ra thay đổi được đánh dấu là COMMIT; nếu không, hệ thống sẽ dừng hoạt động/tiếp tục tùy thuộc vào chế độ bảo vệ được chọn
* ASYNC: Không xác nhận việc redo data được nhận bởi CSDL Standby, do đó, giao dịch có thể COMMIT ngay lập tức trên CSDL Primary
* AFFIRM: Xác nhận việc redo data nhận được *sau khi* được ghi vào Standby Redo Log của CSDL Standby bằng một tín hiệu Acknowledgement (ACK), tổng quan là gửi ACK sau khi cơ chế Input/Output (I/O) thành công
* NOAFFIRM: Khác với AFFIRM ở chỗ sẽ gửi tín hiệu ACK, nhưng *gửi trước* khi được ghi vào Standby Redo Logs

Về phân loại theo chế độ bảo vệ, bằng cách tinh chỉnh cách truyền thông tin thay đổi để hỗ trợ. Các chế độ này giúp doanh nghiệp cân bằng việc toàn vẹn dữ liệu và hiệu năng của hệ thống:

* *Ưu tiên bảo vệ (max. protection):* 
  + Chế độ bảo vệ mà CSDL Primary sẽ đảm bảo rằng không có dữ liệu nào bị sót một cách tuyệt đối trong trường hợp CSDL Primary gặp sự cố như thảm họa thiên tai, bị lỗi mạng hoặc bị lỗi với CSDL Standby
  + CSDL Primary sẽ dừng hoạt động khi gặp sự cố khiến cho các tiến trình trong việc truyền tải/đồng bộ hóa thông tin thay đổi không thể ghi vào ít nhất một trong các CSDL Standby
  + Thông tin thay đổi phải được xác nhận đã ghi vào tệp tin lưu trữ thông tin thay đổi ở cả CSDL Primary và ít nhất một CSDL Standby (gọi là Standby Redo Log) trước khi giao dịch được đánh dấu là COMMIT
  + Cần thiết lập hai chế độ cho cách truyền thông tin thay đổi với hai đối số: SYNC – đồng bộ hóa (tuần tự theo quá trình gửi redo data, Standby nhận redo data, gửi lại tín hiệu), cần tín hiệu acknowledgement (ACK) và AFFIRM – xác nhận đã ghi xuống đĩa vật lý tại Standby Redo Logs, với ít nhất một Standby có Standby Redo Logs
* *Ưu tiên tính sẵn sàng (max. availabilty):*
  + Chế độ bảo về mà CSDL Primary sẽ đảm bảo không có dữ liệu nào bị sót nhưng không hoàn toàn tuyệt đối, vì không tác động tới việc vận hành của CSDL Primary trong một ràng buộc về thời gian cho trước
  + Thông tin về thay đổi cũng phải được ghi vào Online Redo Log của Primary và Standby (Standby Redo Log) và truyền thông qua các tiến trình LNSs và RFS
  + Tuy nhiên, CSDL Primary sẽ không dừng hoạt động nếu có sự cố không thể xác nhận CSDL Standby đã ghi/nhận hay chưa
  + Nếu có sự cố, CSDL Primary sẽ hoạt động theo cách thức bất động bộ (ASYNC, hoạt động không đợi xác nhận Standby đã nhận redo data hay chưa) cho đến khi ít nhất một CSDL Standby được đồng bộ về mặt thông tin thay đổi, xử lý các khoảng trống giữa dữ liệu bằng tiến trình Fetch Archived Log (FAL). Hiểu đơn giản, hệ thống sẽ chuyển sang chế độ *max. performance* để hoạt động
  + Cần thiết lập hai đối số: SYNC – đồng bộ hóa (tuần tự) và NOAFFIRM (không cần xác nhận đã ghi vào Standby Redo Logs) hoặc AFFIRM cho ít nhất một CSDL Standby có chứa Standby Redo Logs
* *Ưu tiên hiệu năng hệ thống (max. performance):*
  + Đây là chế độ mặc định của giải pháp ODG. Cung cấp việc bảo vệ dữ liệu thấp hơn hai mức còn lại về tính vẹn toàn, kịp thời nhưng hiệu năng hệ thống được tăng trên CSDL Primary
  + Giao dịch được xác nhận COMMIT, đồng thời thông tin thay đổi sẽ được lưu xuống tệp tin lưu trữ thông tin thay đổi ngay lập tức mà không cần quá trình xác nhận (ACK - acknowledgement) từ CSDL Standby
  + Thông tin thay đổi (redo data) được truyền tới CSDL Standby theo cách bất đồng bộ (ASYNC, các quá trình không phải đợi nhau lần lượt) với những thông tin thay đổi đã được COMMIT
  + Cần cấu hình đối số như sau: ASYNC – bất đồng bộ và NOAFFIRM – không xác nhận đã ghi cho CSDL Standby đã có Standby Redo Logs
  + Điều đặc biệt là khi băng thông mạng lớn, chế độ ưu tiên hiệu năng cung cấp việc truyền tải thông tin thay đổi gần tương tự với chế độ ưu tiên tính sẵn sàng mà vẫn đảm bảo hiệu năng cao

Bảng so sánh về các chế độ bảo vệ dưới đây sẽ có cái nhìn tổng quan và ngắn gọn hơn. Lưu ý rằng, đây là các kết hợp đối số có ý nghĩa, có một số trường hợp kết hợp đối số khác không được chấp nhận như ASYNC/AFFIRM:

|  |  |  |  |
| --- | --- | --- | --- |
| **Chế độ** | **Rủi ro** | **Chế độ truyền** | **Nếu tín hiệu ACK không gửi (AFFIRM/NOAFFIRM)** |
| Ưu tiên bảo vệ | Không mất đồng bộ dữ liệu và không mất dữ liệu, đảm bảo tính toàn vẹn ở cả hai CSDL | SYNC | CSDL Primary sẽ treo/dừng hoạt động cho đến khi nhận được tín hiệu ACK |
| Ưu tiên tính sẵn sàng | Không mất đồng bộ dữ liệu, đảm bảo tính sẵn sàng của CSDL chính và có thể đảm bảo toàn phần dữ liệu | SYNC/AFFIRM | CSDL Primary sẽ chờ trong một khoảng thời gian được xác định là Time-to-Live, nếu hết, sẽ tiếp tục hoạt động, xử lý phần thiếu với tiến trình FAL |
| Ưu tiên tính sẵn sàng | Rủi ro trong việc mất dữ liệu nếu các CSDL đều lỗi | SYNC/NOAFFIRM (Tính năng Fast-Sync) | CSDL Primary sẽ chờ trong một khoảng thời gian được xác định là Time-to-Live, nếu hết, sẽ tiếp tục hoạt động, xử lý phần thiếu với tiến trình FAL |
| Ưu tiên hiệu năng | Rủi ro trong việc mất đồng bộ dữ liệu cao nếu gặp vấn đề truyền và CSDL Primary lỗi | ASYNC | CSDL Primary không chờ tín hiệu ACK |

Bảng 1 Cấu hình đối số phương thức truyền/xác nhận

**Cấu hình bảo vệ nâng cao với Far Sync**

Tính năng Far Sync là một hệ thống CSDL Standby cho phép là điểm trung chuyển thông tin thay đổi tới nhiều các Standby khác nếu hệ thống mạng lưới có nhiều CSDL Standby. Thực tế, Far Sync Standby là một hệ thống có kích thước gọn nhẹ, tiêu thụ ít tài nguyên về lưu trữ cũng như xử lý.

Far Sync Standby giống với CSDL Standby thông thường ở điểm quản lý các Control File, nhận redo data vào Standby Redo Logs và lưu trữ khi Log Switch xuống Archived Redo Logs. Ngược lại, khác với CSDL Standby ở điểm không lưu trữ một số tệp tin không quan trọng như Data Files, không hỗ trợ chuyển đổi vai trò và chỉ thiết lập được tại 02 chế độ bảo vệ: *max. performance* hoặc *max. availability*.



Hình 10 Tính năng Far Sync trong giải pháp Oracle Data Guard

Có thể kể đến một số lợi ích mà tính năng Far Sync đem lại như”

* Trong chế độ *max. availability* (SYNC/AFFIRM hoặc SYNC/NOAFFIRM – FastSync), Far Sync Standby hoạt động gần giống như CSDL Primary bằng việc giảm độ trễ trong việc phản hồi tín hiệu ACK khi đồng bộ hóa bản sao (đối với CSDL Primary, khi là trung gian chuyển đến các Standby khác). Nếu CSDL Primary lỗi, các CSDL Standby sẽ lấy các redo data cuối cùng từ Far Sync Standby trong việc chuyển đổi vai trò, đảm bảo tính vẹn toàn
* Trong chế độ *max. performance* (ASYNC/NOAFFIRM), Far Sync Standby không bị đặt nặng vấn đề khoảng cách do cách truyền bất đồng bộ, không ảnh hưởng đến hiệu năng của CSDL Primary. Hơn hết, Far Sync Standby giúp giảm tải CSDL Primary rất nhiều trong việc truyền tải redo data đến hàng loại CSDL Standby
* Ngoài ra, việc bổ sung các tính năng như đóng gói, mã hóa dữ liệu được Far Sync Standby đảm nhận nhằm giảm tải cho CSDL chính
  + 1. Cơ chế tương tác giữa các thành phần

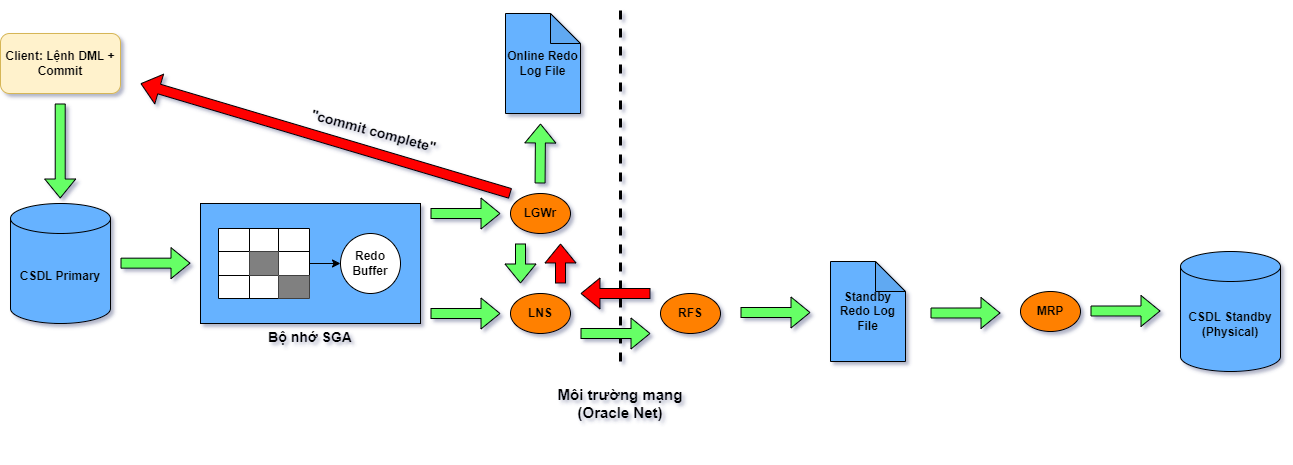
Các thành phần giao tiếp với nhau trong kiến trúc giải pháp Oracle Data Guard thông qua 03 cơ chế chính:

* Cơ chế vận chuyển thông tin thay đổi (Redo Transport Services): tự động hóa việc kiểm soát, điều khiển thông tin thay đổi từ CSDL Primary sang CSDL Standby
* Cơ chế áp dụng thông tin thay đổi (Log Apply Services): tự động hóa việc áp dụng thông tin và cách áp dụng thay đổi khi nhận được thông tin thay đổi. Với CSDL Standby Physical, thông tin thay đổi được áp dụng thông qua việc khôi phục với công nghệ Oracle Media Recovery (được sử dụng trong RMAN). Với CSDL Standby Logical, thông tin thay đổi được LogMiner phiên dịch thành SQL và thực thi SQL để áp dụng thay đổi.
* Cơ chế quản lý/chuyển đổi vai trò (Role Management Services): là một tiến trình nền (Background Process), thực hiện thay đổi vai trò của Standby sang Primary khi CSDL Primary gặp lỗi, hệ thống dừng hoạt động. Trong đó, Switchover – thay đổi có kế hoạch và Failover – thay đổi khi gặp sự cố không lường trước được. Trong đó, Switchover yêu cầu cả hai loại CSDL không bị lỗi và Failover chỉ được thực hiện ở Standby, không yêu cầu với Primary.

Cơ chế hoạt động của Oracle Data Guard thể hiện rõ nhất ở việc lựa chọn chế độ bảo vệ. Hai chế độ bảo vệ *Ưu tiên hiệu năng* và *Ưu tiên bảo vệ* có những đặc điểm nổi trội khác nhau, đặc biệt hơn so với *Ưu tiên về tính sẵn sàng*. Vì vậy, để hiểu được rõ được sự khác biệt và cơ chế hoạt động của luồng dữ liệu, hình vẽ cùng miêu tả các pha sẽ minh họa cơ chế hoạt động hai chế độ, hoạt động trong loại CSDL Standby vật lý (với tiến trình áp dụng thay đổi là MRP).

Với các hình vẽ minh họa, màu vàng be thể hiện hoạt động từ phía hệ thống máy người dùng, màu xanh thể hiện các thành phần chung hoặc CSDL, màu cam thể hiện tiến trình, màu xanh và đỏ thể hiện luồng đi thông tin.

**Chế độ bảo vệ *Ưu tiên bảo vệ***



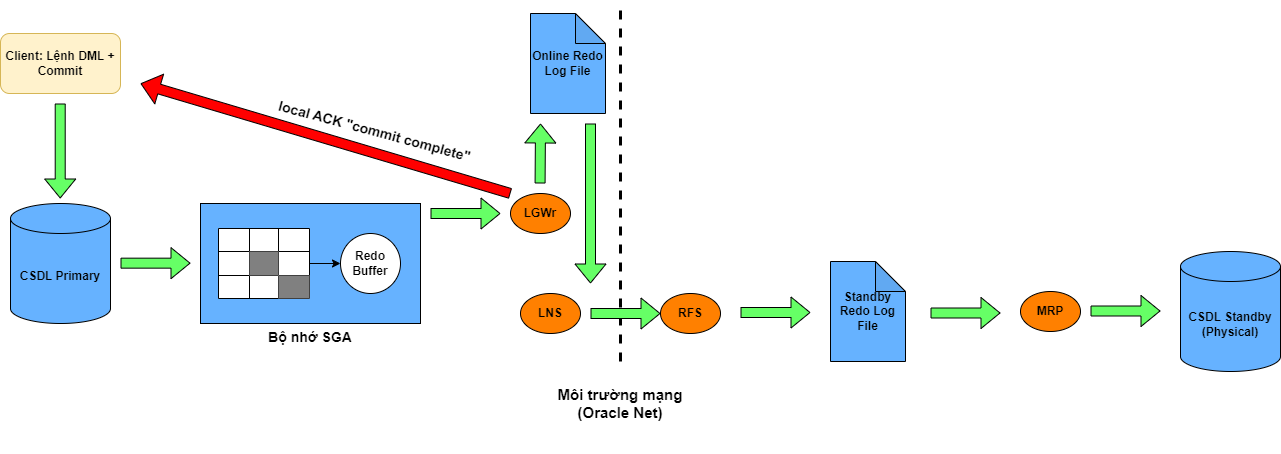
Hình 11 Luồng hoạt động của Oracle Data Guard với chế độ Ưu tiên bảo vệ

Các pha của cơ chế *Ưu tiên bảo vệ* được thể hiện tuần tự với các bước như sau, khi cấu hình SYNC/AFFIRM:

1. Người dùng sử dụng ứng dụng để giao tiếp với CSDL Primary. Người dùng ra lệnh dạng Data Manipulation Language (DML) như INSERT, UPDATE, DELETE và yêu cầu COMMIT. Vì với cấu hình SYNC/AFFIRM cho chế độ ưu tiên bảo vệ, cho nên, hệ thống chưa xác nhận COMMIT ngay lúc này do chưa nhận được xác nhận từ bên CSDL Standby đã áp dụng thay đổi vào đĩa
2. Trong CSDL Primary, tiến trình Server Process sẽ lấy dữ liệu cần thay đổi theo đơn vị block từ đĩa cứng lên Data Buffer Cache. Tại đây, các dữ liệu gốc (một block có thể là một dòng hoặc rất nhiều dòng dữ liệu) sẽ được thay đổi theo yêu cầu của lệnh DML. Block sau khi chỉnh sửa sẽ được đánh dấu là “dirty block”, ngoài ra, Redo Buffer Cache sẽ lưu lại các thông tin về thay đổi của dirty block theo dạng véc-tơ (gọi là redo data), có thể minh họa dễ hiểu như “Dòng ID xxx UPDATE từ zzz sang yyy, số SCN 1234, …”
3. Thông tin thay đổi sẽ được tiến trình LGWR đưa và lưu xuống Online Redo Log do có tín hiệu COMMIT, nhưng khi đã lưu, hệ thống chưa báo “commit complete” ngay. Lưu ý: lúc này, dữ liệu được thay đổi trên Data Buffer Cache chưa được lưu xuống đĩa, chỉ lưu khi xảy ra sự kiện Checkpoint thực hiện hoặc nguyên nhân khác
4. Tiến trình LNS sẽ nhận những thông tin thay đổi trong lúc LGWR xuất ra. Những thông tin thay đổi này sẽ được chuyển cho tiến trình RFS thuộc CSDL Standby, nhằm thực hiện sao lưu
5. Sau khi nhận được thông tin thay đổi, tiến trình RFS sẽ ghi những thông tin thay đổi xuống Standby Redo Log Files
6. Nếu CSDL Standby sử dụng tính năng Real-time Apply, thì ngay lập tức, các thông tin thay đổi này sẽ được áp dụng vào dữ liệu lưu trữ vật lý tại CSDL Standby với tiến trình khôi phục Media Recovery Process (MRP) dành cho loại Physical Standby
7. Với SYNC/AFFIRM, sau khi dữ liệu đã được áp dụng thành công, tiến trình RFS sẽ phản hồi lại cho tiến trình LNS thông tin (gọi là tín hiệu acknowledgement - ACK), lúc này, hệ thống mới tiến đến phản hồi người dùng “commit complete”. Tại đây, nếu gặp sự cố về môi trường mạng, khiến cho RFS không thể gửi cho LNS, CSDL Primary sẽ đợi tới khi nào nhận được thông tin dẫn tới hệ thống treo
8. LNS chuyển tiếp cho LGWR để hệ thống phản hồi lại cho người dùng thông báo COMMIT thực hiện thành công. Kết thúc quá trình của lệnh DML và COMMIT

Ngoài ra, khi xảy ra Log Switch trên CSDL Primary, sẽ kích hoạt một Trigger giúp CSDL Standby cũng thực hiện Log Switch đối với Standby Redo Log Files nhằm đảm bảo tính toàn vẹn. Đối với RFS, tiến trình này sẽ gửi trực tiếp redo data xuống Archive Log File nếu: Không có Standby Redo Logs (1), Standby Redo Log được cài đặt nhỏ hơn kích thước của Online Redo Logs (2), tất cả Standby Redo Logs đều chưa được lưu trữ (archived) (3) và nếu RFS thực hiện nhận redo data từ tiến trình ARCn trong cơ chế Gap Resolution (4).

**Chế độ *Ưu tiên hiệu năng***



Hình 12 Luồng hoạt động của Oracle Data Guard với chế độ Ưu tiên bảo vệ

Nhìn chung, trong chế độ ưu tiên hiệu năng, luồng dữ liệu không thay đổi nhiều. Tuy nhiên, với cơ chế ASYNC/NOAFFIRM, tức là hệ thống CSDL Primary sẽ không yêu cầu nhận bất kỳ tín hiệu xác nhận nào từ CSDL Standby trong việc ghi/áp dụng thay đổi thành công. Các redo data được LNS tiếp nhận và lấy tại Redo Buffer Cache hoặc lấy tại Online Redo Logs khi mà tốc độ xử lý của LNS không thể bắt kịp tốc độ hệ thống COMMIT, do đó, Log Switch xảy ra, các redo data được đưa xuống Online Redo Logs.

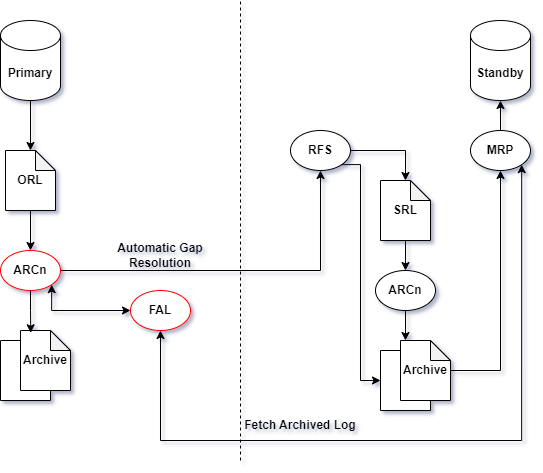
Khi người dùng gõ lệnh DML và yêu cầu COMMIT, hệ thống sẽ ngay lập tức trả lại tín hiệu COMMIT thành công – “commit complete”, do không phải chờ phản hồi từ CSDL Standby. Ở đây, tín hiệu ACK còn được gọi là “local ACK”, có nghĩa là tín hiệu này sẽ không xuất phát từ tiến trình RFS truyền qua môi trường mà CSDL Standby chứa đựng. Như vậy, hiệu năng hệ thống sẽ tăng lên đáng kể, nhưng rủi ro trong việc đảm bảo dữ liệu đồng bộ hóa và trong tính toàn vẹn là rất lớn.

Ngoài ra, để tăng hiệu năng hệ thống trong việc I/O tại đĩa, tiến trình LGWR cũng sẽ không chia sẻ thông tin thay đổi, mà LNS sẽ tự trích xuất từ Online Redo Logs và chuyển tiếp cho CSDL Standby.

**Cơ chế Log Delay và xử lý thiếu thông tin thay đổi**

Trong kiến trúc của Oracle Data Guard cũng thể hiện thêm những đường kết nối khác, thể hiện cho cấu hình tham số khác, tùy chỉnh theo yêu cầu của tổ chức, doanh nghiệp. Như cấu hình tham số liên quan tới ghi thay đổi nhưng tăng độ trễ, vì có thể liên quan tới vấn đề đường truyền, hiệu năng hạn chế. Lúc này, redo data thay vì từ Standby Redo Log File áp dụng trực tiếp tới CSDL Standby, sẽ lưu xuống Archived Log File rồi mới áp dụng thay đổi. Tham số này gọi là Log Delay.

Ngoài ra, cũng có thêm vấn đề về tính khác biệt, khoảng cách về tính cập nhật của dữ liệu bị chậm khi hệ thống vừa mới đồng bộ hóa lại sau khoảng thời gian mất tín hiệu mạng. Trong Oracle Data Guard, thuật ngữ để miêu tả tình trạng này là “Archive Redo Gap Sequence”. Tình trạng này thường xảy ra trong chế độ *Ưu tiên hiệu năng*, Oracle cung cấp 02 cơ chế để xử lý tình trạng này là “Automatic Gap Detection & Resolution with ARCH” và “Fetch Archive Log – FAL”.



Hình 13 Minh họa cơ chế xử lý thiếu trong việc truyền thông tin thay đổi

Khi hệ thống CSDL Primary quá tải, môi trường mạng gặp sự cố, và đặc biệt trong chế độ *Ưu tiên hiệu năng*, do cơ chế ASYNC, các redo data được COMMIT liên tục và không đợi xác nhận từ phía CSDL Standby. Với các vấn đề đã nêu, tiến trình LNS (NSA đối với cơ chế ASYNC) của CSDL Standby không thể bắt kịp những thông tin thay đổi trong Redo Buffer Cache hoặc Online Redo Logs như đã nêu ở phần cơ chế ưu tiên hiệu năng. Khi đó, Log Switch có thể xảy ra, các redo data trong Online Redo Logs được lưu xuống Archive Redo Logs nhằm lưu trữ lâu dài.

*Với cơ chế Automatic Gap Detection & Resolution:* Đây là cơ chế tự động, không cần cấu hình chi tiết. Tiến trình ARCn của CSDL Primary sẽ liên tục gửi lệnh ping đến cho tiến trình RFS của CSDL Standby để xác định trạng thái khi có phản hồi lẫn nhau. Khi có kết nối mạng hoặc hệ thống CSDL Primary khôi phục, ARCn sẽ tiến hành ping kèm truy vấn tệp tin Control Files nằm trong Standby Redo Logs, xác định thông tin về tệp tin lưu trữ thông tin thay đổi gần nhất mà RFS đã nhận. Từ đây, ARCn có cơ sở để xác định những tệp tin lưu trữ redo data nào đang bị thiếu, truyền gửi tự động qua RFS tới CSDL Standby để khôi phục. Ngoài ra, tiến trình LNS cũng sẽ hỗ trợ trong việc cập nhật/gửi đi các redo data mới nhất từ Redo Buffer Cache, Online Redo Logs, giúp CSDL Standby nhanh chóng khôi phục lại được trạng thái cập nhật gần nhất với CSDL Primary. Cho đến khi nào tiến trình MRP/LSP có thể đọc được Standby Redo Logs kịp thời, nó sẽ chuyển từ áp dụng thay đổi trong Archive Log sang Standby Redo Log. Cơ chế này được thực hiện bằng cách sử dụng đa tiến trình ARCn (với n mặc định là 4 tiến trình, lớn nhất là 30 tiến trình) nhằm giúp CSDL Standby thực hiện đồng bộ bắt kịp trạng thái dữ liệu với CSDL Primary, sau đó LGWR sẽ nhận được thông báo hoàn thành.

*Với cơ chế Fetch Archive Log (FAL):* đây là một phương pháp dành cho CSDL Standby vật lý, phương pháp này còn được gọi là phương pháp giảm thiểu trễ thông tin chủ động do hành động “fetch” – yêu cầu/lấy thông tin đang bị thiếu được thực hiện một cách chủ động bởi CSDL Standby thay vì như tiến trình ARCn của CSDL Primary quét dò. Sau khi được tiến trình ARCn gửi và RFS nhận vào, CSDL Standby sẽ cập nhật trong Control File của nó về tên và địa điểm của các Archive Log. Khi MRP thấy được những thay đổi mới trong Control File, hệ thống sẽ tiến hành áp dụng những thay đổi này vào CSDL Standby. Nhưng khi MRP thấy thông tin về tệp Archive Log từ CSDL Primary lỗi/hỏng/thiếu thì nó sẽ sử dụng cơ chế chủ động thông qua tiến trình FAL, yêu cầu gửi lại Archive Log. Cơ chế này cần thiết lập hai tham số chính trên CSDL Standby như sau:

|  |  |
| --- | --- |
| **Tham số** | **Miêu tả** |
| FAL\_SERVER | Tên trong môi trường Oracle Net, cấu hình trong tnsnames.ora, trỏ tới Listener của CSDL Primary |
| FAL\_CLIENT | Tên trong môi trường Oracle Net, cấu hình trong tnsnames.ora, trỏ tới Listener của CSDL Standby |

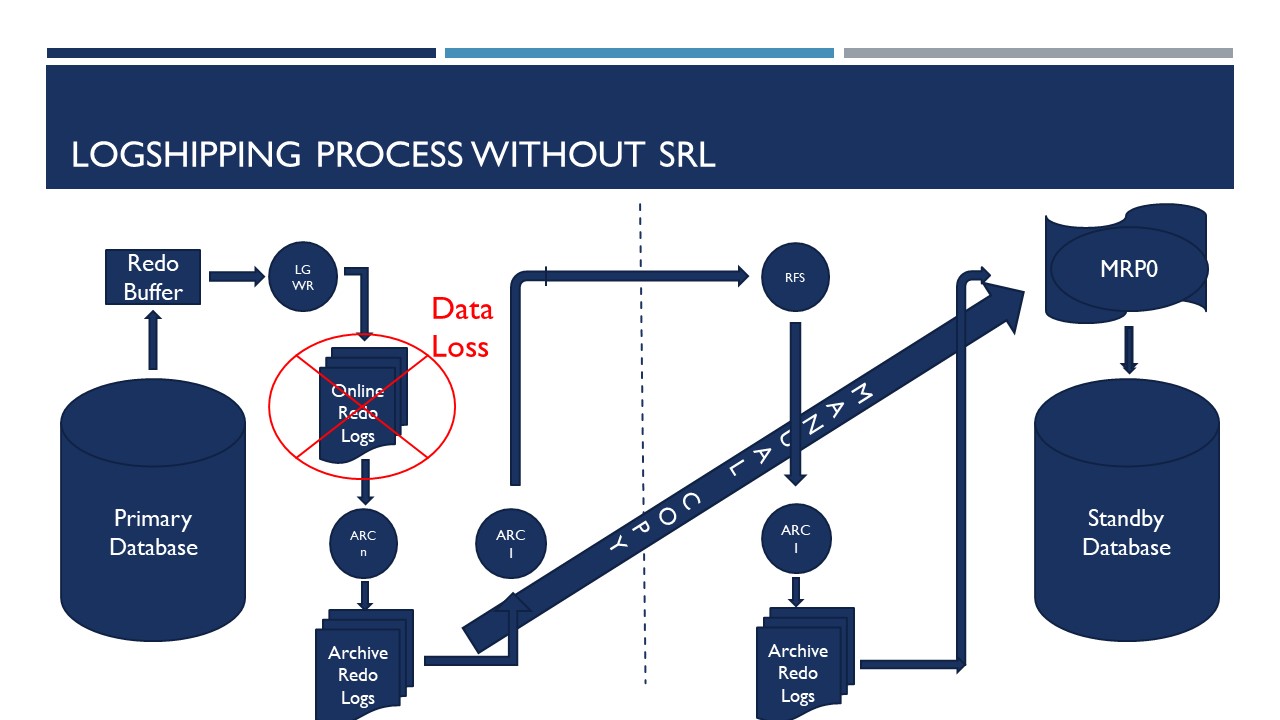
Bảng 2 Tham số cấu hình cho cơ chế Fetch Archive Log

Khi MRP cần giải quyết việc trễ gây ra thiếu dữ liệu, tiến trình này sẽ dựa vào tham số FAL\_SERVER để kích hoạt/gọi đến FAL của CSDL Primary. Kết nối được thiết lập giữa CSDL Primary và CSDL Standby, FAL\_CLIENT sẽ cho tiến trình ARCn của CSDL Primary biết rằng CSDL Standby nào đang cần nhận thêm dữ liệu để ARCn có thể gửi tới nó thông qua FAL Server. Nếu địa chỉ CSDL Primary trong FAL\_SERVER không thể giải quyết do không thể kết nối được, CSDL Standby sẽ dò và quét lần lượt từng địa chỉ trong FAL\_SERVER cho tới khi vấn đề được giải quyết hoặc một CSDL khác có thể kết nối tới.

**Tầm quan trọng của Standby Redo Log Files trong cơ chế hoạt động**

Trước khi xuất hiện Standby Redo Log Files được giới thiệu tại phiên bản Oracle 9i, thì tiến trình RFS sẽ đọc trực tiếp thông tin thay đổi trong Archived Redo Log Files của CSDL Primary với sự trợ giúp trích xuất của tiến trình ACRn và ghi thông tin này vào Archived Redo Logs của CSDL Standby. Sau đó, thông tin thay đổi này mới được sử dụng thông qua quá trình khôi phục CSDL với tiến trình MRP để thực hiện áp dụng thay đổi vào CSDL Standby.

Hạn chế của kiến trúc này là do chỉ có một luồng dữ liệu duy nhất đi từ CSDL Primary, thông tin thay đổi từ Online Redo Logs phải ghi xuống Archived Redo Logs rồi mới được vận chuyển qua CSDL Standby. Một khi CSDL Primary gặp sự cố, tiến hành chuyển đổi vai trò fail-over, những thông tin thay đổi trong Online Redo Logs chưa được lưu xuống Archived Redo Logs (VD: Log Switch chưa xảy ra), gây ra hiện tượng lệch dữ liệu, khiến việc vận hành hệ thống không được nhất quán, sai kết quả trong luồng nghiệp vụ. Ngoài ra, việc cập nhật dữ liệu sang CSDL Standby cũng sẽ không được giống kịp thời với CSDL Primary, do thông tin thay đổi lấy được là những thông tin thay đổi được lưu xuống, tuần tự chuyển sang CSDL Standby sau một khoảng thời gian lưu vào Online Redo Logs cho tới khi cơ chế Log Switch xảy ra.



Hình 14 Kiến trúc của Oracle Data Guard trước khi có Standby Redo Log Files

Sau khi Oracle bổ sung thêm Standby Redo Log Files, kiến trúc Oracle Data Guard hoạt động với việc vận chuyển thông tin thay đổi từ Online Redo Logs sang Standby Redo Logs, từ đây, hệ thống CSDL Standby có thể áp dụng thông tin thay đổi với phương thức “real-time apply” một cách nhanh chóng, kịp thời, đảm bảo tính vẹn toàn và đồng bộ. Khi này, cũng xuất hiện tiến trình LNS để phụ trợ và giảm tải tiến trình LGWR việc nhận và truyền redo data. LNS là tên gọi chung của hai tiến trình con Network Server Async (NSA) – dùng trong chế độ ASYNC/NOAFFIRM và Network Server Sync (NSS) – dùng trong chế độ SYNC/AFFIRM.

Standby Redo Log File có nhiệm vụ giống với Online Redo Log, lưu trữ dưới đĩa những thông tin thay đổi tạm thời (nếu có Archived Log Files, sẽ lưu lâu hơn) và xoay vòng khi các nhóm Standby Redo Log File hết không gian lưu trữ. Thuật ngữ Standby Redo Log dùng để chỉ thành phần riêng biệt để lưu trữ redo data được nhận từ CSDL khác.

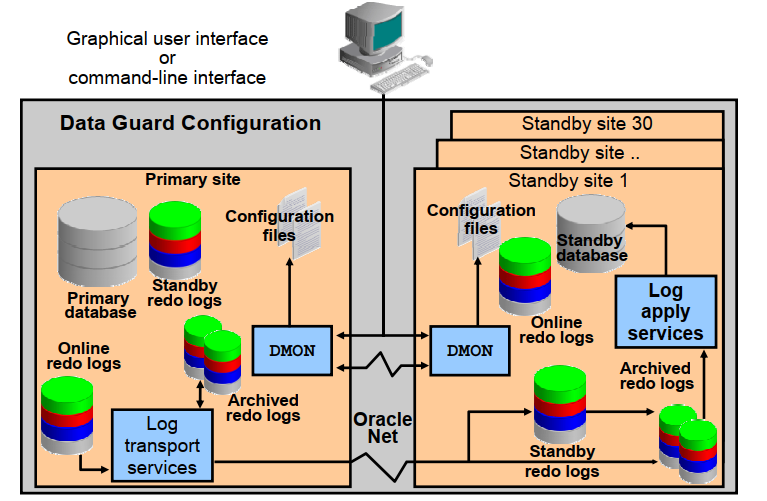
Standby Redo Logs đóng vai trò quan trọng trong chế độ *Ưu tiên bảo vệ* hoặc *Ưu tiên tính sẵn sàng* bằng việc làm giảm thời gian để hệ thống xác nhận một giao dịch đã được ghi lại thay đổi ở cả hai loại CSDL Primary và CSDL Standby sau khi COMMIT. Điều này được thể hiện ở chỗ, tính năng Real-Time Apply chỉ hỗ trợ Standby Redo Logs, các thông tin thay đổi sau khi được chuyển tới Standby Redo Logs bằng tiến trình RFS, sẽ được áp dụng ngay vào CSDL Standby, thay vì phải chờ Log Switch xuất hiện ở CSDL Primary, chuyển xuống Archived Redo Log rồi mới được chuyển sang CSDL Standby. Standby Redo Logs cũng góp phần giảm việc mất mát dữ liệu, giữ trạng thái CSDL Standby luôn “near real-time” trong chế độ *Ưu tiên hiệu năng*, cũng bởi thông tin thay đổi được gửi trực tiếp tới Standby Redo Log.

Trong tổ chức/doanh nghiệp, lượng dữ liệu rất lớn, dẫn tới kéo theo cấu hình Online Redo Logs lớn, vì vậy, Log Switch xảy ra có thể từ hàng chục phút cho tới hàng tiếng; nếu không có Standby Redo Logs để có thể lưu trữ trực tiếp thông tin thay đổi mà CSDL Primary gặp sự cố, tổ chức/doanh nghiệp cũng sẽ mất một lượng dữ liệu khổng lồ khi fail-over sang CSDL Standby. Vì vậy, vai trò của việc thiết lập Standby Redo Logs là cực kỳ quan trọng, đảm bảo tính toàn vẹn và tối ưu trong việc vận hành.

* + 1. Oracle Data Guard Broker

***Khái niệm và kiến trúc***

Data Guard Broker là một tính năng được tích hợp trong Oracle Database Server, dùng để hỗ trợ trong việc quản trị mô hình phân tán trong các vấn đề liên quan tới giải pháp Oracle Data Guard. Data Guard Broker – tên để chỉ một tính năng bao gồm rất nhiều thành phần, vì vậy tính năng này được triển khai bằng cách triển khai ở phía người dùng (client-side) với công cụ điều khiển/quản trị như Oracle Enterprise Manager/Broker’s Interfaces (DGMGRL – Data Guard Command Line Interface) và ở phía máy chủ (server-side) là tiến trình Data Guard Monitor (DMON) cùng với tập tin cấu hình.



Hình 15 Kiến trúc Oracle Data Guard với tính năng Data Guard Broker

Data Guard Broker (DGB) quản trị kiến trúc Data Guard với framework Broker Management Model (BMM). Framework BMM là một cấu trúc bao gồm thông tin cấu hình (Configuration), CSDL (Database) và tài nguyên CSDL (Database resourcec). DGB có thể quản lý 01 CSDL Primary và nhiều nhất 30 CSDL Standby đối với mỗi một thông tin cấu hình. Các CSDL Standby có thể là một trong các dạng Physical hoặc Logical, khác hơn có thể là một Far Sync Standby.

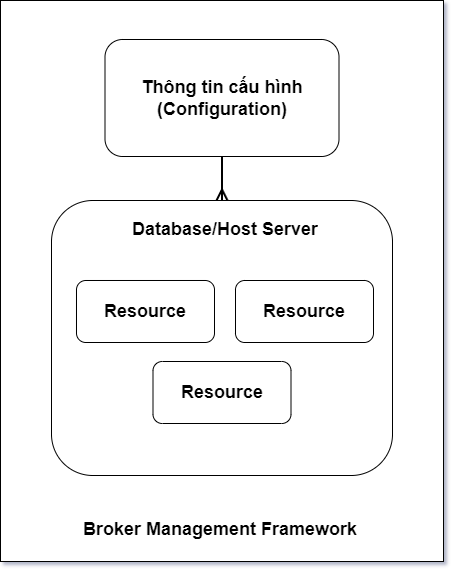
Một số lợi ích khi sử dụng DGB trong kiến trúc của ODG là:

* Khi tạo CSDL Standby, Broker sẽ tự động việc tạo các tệp tin như tệp tin cấu hình (spfile.ora), Oracle Net (tnsnames.ora)
* Là công cụ dùng để quản lý tập trung, có thể sử dụng thông qua giao diện (GUI) hoặc bằng giao diện dòng lệnh (CLI)
* Cung cấp cơ chế kiểm tra tình trạng hệ thống, giảm độ phức tạp khi thực hiện chuyển đổi vai trò giữa các CSDL

|  |  |  |
| --- | --- | --- |
|  | **Sử dụng Broker** | **Không sử dụng Broker** |
| **Quản trị** | Việc quản trị được tập trung hóa | Việc quản trị thực hiện riêng lẻ đối với từng CSDL |
| **Tạo CSDL Standby** | Sử dụng DGMRL hoặc OEMCC để đơn giản hóa và tự động việc tạo CSDL Standby (với GUI) (bao gồm control file, online redo log files, datafiles, và các tệp tin lưu trữ tham số) | Tạo thủ công (bằng cách sử dụng công cụ RMAN hoặc các công cụ khác):   * Sao chép các tệp tin của CSDL Primary cho CSDL Standby * Tạo control file trên CSDL Standby * Tạo file tham số trên CSDL Standby * Sao chép password file từ CSDL Primary sang CSDL Standby |
| **Cấu hình và quản lý** | Cho phép cấu hình và quản lý nhiều CSDL tập trung và quản lý cấu hình kết nối của các CSDL thông qua một tệp tin duy nhất | Cấu hình thủ công:   * Thiết lập Redo Transport Services và Log Apply Services tại mỗi CSDL * Quản lý CSDL riêng lẻ |
| **Điều khiển** | * Tự động thiết lập Redo Transport Services và Log Apply Services * Đơn giản hóa việc chuyển đổi vai trò và chuyển đổi Snapshot CSDL Standby về Physical | Sử dụng thủ công:   * Sử dụng SQL để quản lý * Sử dụng nhiều lệnh hệ thống để quản lý các CSDL cho việc chuyển đổi vai trò cũng như điều khiển các tiến trình |
| **Theo dõi** | * Cho phép theo dõi hiệu năng hệ thống, cấu hình và các tham số khác * Cung cấp báo cáo chi tiết về hệ thống | * Chỉ theo dõi cố định vào khoảng thời gian nhất định * Không tập hợp các tham số cần theo dõi cùng một lúc |

Bảng 3 So sánh việc sử dụng Broker vào hệ thống

***Framework quản lý Broker***

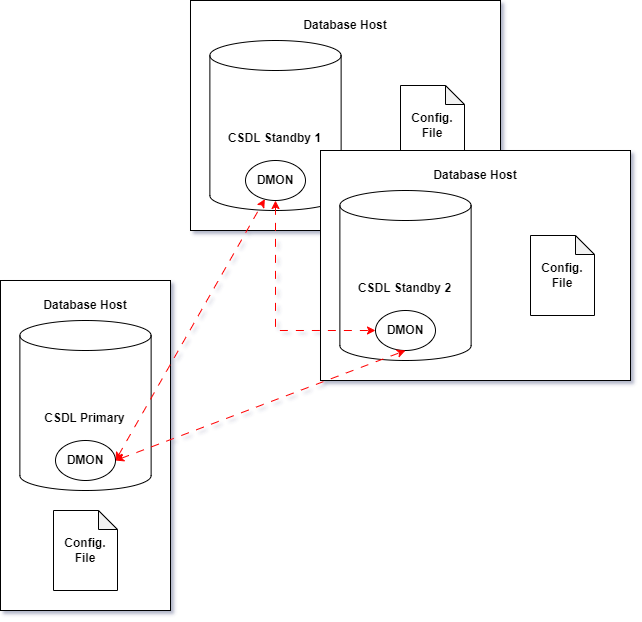


Hình 16 Mối quan hệ giữa các thành phần trong mô hình Broker

Trong mô hình quản lý của Broker, thông tin cấu hình Configuration được xem là nhóm tập hợp lô-gic các CSDL, bao gồm 01 CSDL Primary, tối đa 30 CSDL Standby hoặc có thể là Far Sync Standby.

Resource là đơn vị nhỏ nhất được quản lý bởi Broker, thành phần thể hiện một hoặc nhiều (đối với mô hình Real Application Clusters - RAC) Instance của CSDL. Ngoải ra, một Service như Log Apply Service (MRP hoặc LSP) cũng có thể được phân loại là Resource. Database hoặc Host Server là tập hợp nhiều Resources, hay chính là hệ thống CSDL Primary hoặc Standby mà Instance chạy trên chính nó. Configuration là đơn vị lớn nhất, bao gồm một hoặc nhiều CSDL (Database/Host Server). Tất cả thay đổi của một lớp sẽ gây các thành phần thuộc lớp đó thay đổi theo, được gọi là tính kế thừa của các thành phần. Ví dụ, nếu Database/Host Server có trạng thái tắt nguồn, thì các thành phần bên trong là Resource cũng sẽ không hoạt động.

Trong server-side, các thành phần của DGB gồm tiến trình DMON và tệp tin thông tin cấu hình. DMON là tiến trình nền, chạy ở mỗi Database Host khi Broker khởi động và được quản lý bởi Broker. Tệp tin thông tin cấu hình chứa các cài đặt về thuộc tính, trạng thái của Database Host. Tiến trình DMON thực hiện quản lý và sao chép các tệp tin thông tin cấu hình cho mỗi Database Host mà nó quản lý. Các tiến trình DMON ở mỗi CSDL khác nhau giao tiếp thông qua môi trường mạng Oracle Net để quản lý việc luân chuyển vai trò CSDL cũng như thống kê các chỉ số liên quan tới hiệu năng hệ thống.



Hình 17 Giao tiếp các thành phần tại Server-Side trong Broker

* 1. Kết luận chương I

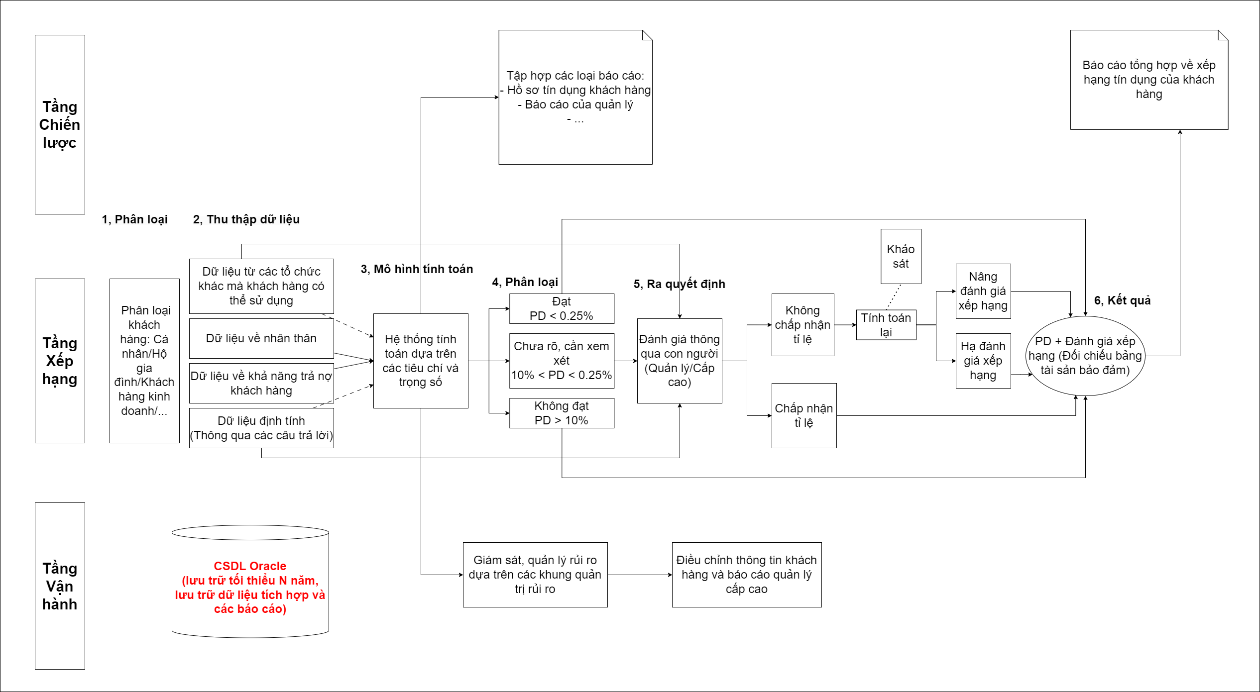
Chương I trình bày tổng quan về Công ty Tài chính tiêu dùng Ngân hàng TMCP Sài Gòn – Hà Nội (SHBFinance), bao gồm nhiệm vụ, chức năng và sứ mệnh của công ty, lịch sử hình thành, và cơ cấu tổ chức hiện tại. Sau đó, chương đi vào mô tả bài toán của công ty, tập trung vào thực trạng hiện tại, những thách thức mà doanh nghiệp này đang đối diện, giá trị mà doanh nghiệp sẽ đạt được và hướng giải quyết cho những thách thức đó. Phần tiếp theo của chương là sự giới thiệu về Oracle Data Guard (ODG), một giải pháp dự phòng dữ liệu cho hệ thống cơ sở dữ liệu Oracle. Bao gồm một số khái niệm quan trọng như kiến trúc của Oracle Database, việc sao lưu và phục hồi dữ liệu sử dụng công cụ Recovery Manager (RMAN), và các loại hình bảo vệ và cơ chế tương tác giữa các thành phần trong Oracle Data Guard.

1. TRIỂN KHAI GIẢI PHÁP ORACLE DATA GUARD CHO CƠ SỞ DỮ LIỆU HỆ THỐNG XẾP HẠNG   
   TÍN DỤNG CỦA SHBFINANCE
   1. Lên kế hoạch xây dựng giải pháp Data Guard cho SHBFinance
      1. Xác định vấn đề

Theo Thông tư số 11/2021/TT-NHNN: “mỗi ngân hàng nói riêng và tổ chức tín dụng nói chung, đều phải xây dựng: (i) Hệ thống xếp hạng tín dụng nội bộ và (ii) Quy định nội bộ về cấp tín dụng, quản lý nợ, chính sách dự phòng rủi ro.” (Nguyễn Nhật Vy, 2022)

Hệ thống xếp hạng tín dụng (nội bộ) là hệ thống gồm: “Các bộ chỉ tiêu tài chính và phi tài chính, các quy trình đánh giá khả năng trả nợ, thanh toán của khách hàng trên cơ sở định tính và định lượng về mặt tài chính, tình hình kinh doanh, quản trị, uy tín của khách hàng (1); Phương pháp đánh giá xếp hạng cho từng nhóm đối tượng khách hàng khác nhau, kể cả các đối tượng bị hạn chế cấp tín dụng và những người có liên quan của đối tượng (2)” (thuvienphapluat.vn, 2023)

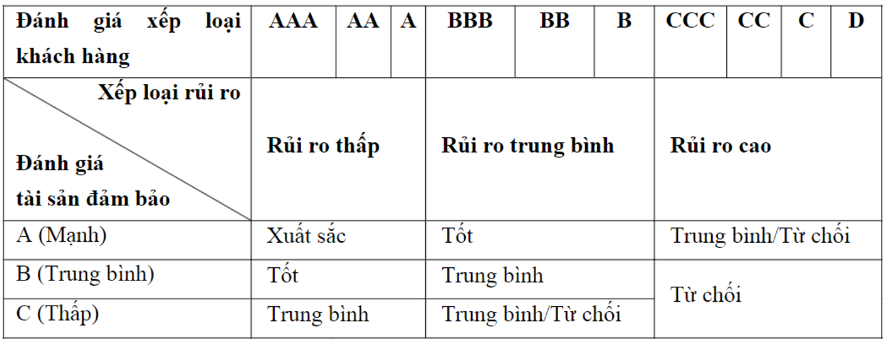
Với loại hình là tổ chức tín dụng phi ngân hàng, công ty Tài chính tiêu dùng SHBFinance tập chung cung cấp sản phẩm cho vay tiêu dùng đối với khách hàng cá nhân, cụ thể là nhân viên của SHBFinance, cá nhân kinh doanh nhỏ lẻ, công nhân và các khách hàng khác với mức thu nhập trung bình từ 3 triệu đồng. Về cơ bản, hệ thống đánh giá tín dụng nội bộ của công ty Tài chính tiêu dùng SHBFinance có quy trình được thể hiện như hình tiếp theo.



Hình 18 Tổng quan hệ thống xếp hạng tín dụng nội bộ của SHBFinance

Diễn giải quy trình hoạt động của hệ thống tín dụng nội bộ SHBFinance:

1. Phân loại khách hàng: SHB Finance phân loại khách hàng theo loại khách hàng. Gồm có 3 loại khách hàng chính mà SHB Finance cung cấp dịch vụ: Cá nhân, Hộ gia đình và Khách hàng là người kinh doanh
2. Thu thập dữ liệu: Dữ liệu chính được dùng là dữ liệu về nhân thân, dữ liệu trả nợ của khách hàng, loại dữ liệu này là dữ liệu được dùng để đưa vào mô hình tính toán. Ngoài ra, còn một số loại dữ liệu khác là dữ liệu từ các tổ chức mà khách hàng có thể sử dụng (Internet, viễn thông, bảo hiểm, …) và dữ liệu định tính (được thể hiện qua một số báo cáo bằng các biểu mẫu đặt câu hỏi dành cho khách hàng như dự định về tương lai, lối sống, …), kiểu loại dữ liệu này thường được dùng làm dữ liệu hỗ trợ cho quản lý ra quyết định khi điểm số chưa chắc chắn. Các dữ liệu này được thu thập và lưu trữ vào CSDL Oracle để thực hiện lưu trữ lâu dài cũng như thiết lập báo cáo
   * Dữ liệu về nhân thân: tuổi, nghề nghiệp, số năm công tác, số người phụ thuộc tài chính, …
   * Khả năng trả nợ: dựa trên phương pháp đánh giá uy tín, lịch sử trả nợ, số vòng quay vay nợ, ước lượng khả năng trả nợ dựa trên thông tin thu nhập, …
3. Tính toán: Mô hình thuật toán tính toán điểm tín dụng thông qua các dữ liệu chính và dữ liệu phụ (nếu có), đưa ra tỉ lệ nhất định trong 3 khoảng của tham số PD - Probability of Default (tỉ lệ vỡ nợ). Ví dụ: Trọng số được chia cho các tiêu chí đánh giá có thể kể đến như:
   * Lịch sử trả nợ, trọng số 35%
   * Dư nợ tại các tổ chức khác, trọng số 30%
   * Lịch sử tín dụng (càng dài càng uy tín), trọng số 15%
   * Số lần vay nợ mới, trọng số 10%
   * Các loại tín dụng sử dụng (loại tín dụng khác nhau sẽ có điểm số khác nhau), trọng số 10%
4. Phân loại: Khi có tỉ lệ, hệ thống sẽ quyết định dựa trên 3 khoảng. Ngoài hai khoảng Đạt (Rủi ro thấp) và Không đạt (Rủi ro cao), khoảng Chưa rõ (Rủi ro trung bình) cần sự quyết định của quản lý cấp cao. Lúc này quản lý cấp cao sẽ cần thông tin báo cáo về Dữ liệu của các tổ chức khác cũng như báo cáo về Kết quả biểu mẫu khảo sát định tính của khách hàng để đưa ra quyết định xếp hạng cho khách hàng.
5. Ra quyết định: Nếu cán bộ ra quyết định không đồng ý với tỷ lệ nằm trong khoảng Rủi ro trung bình, sẽ tiếp tục ra quyết định tăng hoặc giảm tỉ lệ dựa trên một biểu mẫu tiếp theo dành cho khách hàng để cán bộ căn cứ và có thể đánh giá tiếp.
6. Kết quả: Số điểm rủi ro PD sẽ được đối chiếu với giá trị của tài sản bảo đảm của khách hàng để đưa ra hạng của khách hàng trong thang điểm xếp hạng tín dụng. Mỗi hệ thống xếp hạng tín dụng nội bộ đều có thang điểm chuẩn khác nhau, dưới đây là hình minh họa về một bảng quy đổi đối chiếu hạng tín dụng nội bộ của khách hàng cá nhân.



Hình 19 Minh họa bảng quy đổi đối chiếu xếp hạng tín dụng nội bộ

CSDL Oracle được sử dụng để tích hợp thông tin từ nhiều nguồn, chuẩn bị số liệu, dữ liệu cho quy trình đánh giá tín dụng của khách hàng. Các dữ liệu này được đảm bảo lưu trữ trong khoảng thời gian tối đa mà thời gian lên đến hàng năm, tùy vào chính sách của doanh nghiệp cũng như chi phí lưu trữ đầu tư bỏ ra nhằm mục đích sử dụng thông tin để tạo lập các sản phẩm cho vay khách hàng.

Là một hệ thống CSDL quan trọng, tuy nhiên, hệ thống nói trên có một số vấn đề nổi bật như sau:

* Hiện tại, CSDL chính đã sử dụng các bản sao lưu khôi phục ở cả phạm vi nội bộ của trung tâm dữ liệu (local) và ở các nơi lưu trữ khác (điện toán đám mây, băng từ). Tuy nhiên, chưa đáp ứng được việc thời gian khôi phục khi chỉ số RTO lớn, dẫn đến việc khi gặp sự cố, hệ thống cần mất nhiều thời gian hơn gây gián đoạn tới quá trình hoạt động của doanh nghiệp
* Đối với vấn đề tính toàn vẹn của dữ liệu được thể hiện qua chỉ số RPO, khi hệ thống chính gặp sự cố, việc khôi phục lại dữ liệu có được đầy đủ hay không lại phụ thuộc vào bản sao lưu cuối cùng là bao lâu. Nếu bản sao lưu cuối cùng càng lâu, thì lượng dữ liệu mất/hỏng càng lớn
* Trong việc nâng cấp, bảo trì phần mềm/phần cứng của hệ thống CSDL chính, bắt buộc hệ thống chính sẽ phải dừng hoạt động, gây gián đoạn tới việc vận hành kinh doanh của doanh nghiệp
* Khi thực hiện hoạt động truy xuất thông tin trên hệ thống chính với khối lượng lớn sẽ làm giảm tải hiệu năng xử lý thông tin của hệ thống, cần tính toán đến khả năng mở rộng hệ thống để có thể truy xuất thông tin đồng thời

Vì vậy, việc bổ sung, nâng cấp thêm CSDL dự phòng với giải pháp Oracle Data Guard là cần thiết đối với hệ thống xếp hạng tín dụng nội bộ của SHBFinance trong việc đảm bảo quy trình nghiệp vụ được thực hiện không bị gián đoạn bởi sự cố tại trung tâm CSDL chính. Khi đó, CSDL dự phòng do được đồng bộ với CSDL chính thông qua Log Transport Services (tiến trình như RFS, LNS) và Log Apply Services (tiến trình MRP/LSP), có thể sử dụng CSDL dự phòng này để thực hiện các tác vụ như sao lưu dữ liệu, truy vấn dữ liệu (khi sử dụng Active Data Guard) hoặc để sử dụng làm CSDL thay thế trong trường hợp CSDL chính gặp sự cố hoặc cần bảo trì, nâng cấp.

* + 1. Xác định phương thức chuyển đổi dự phòng

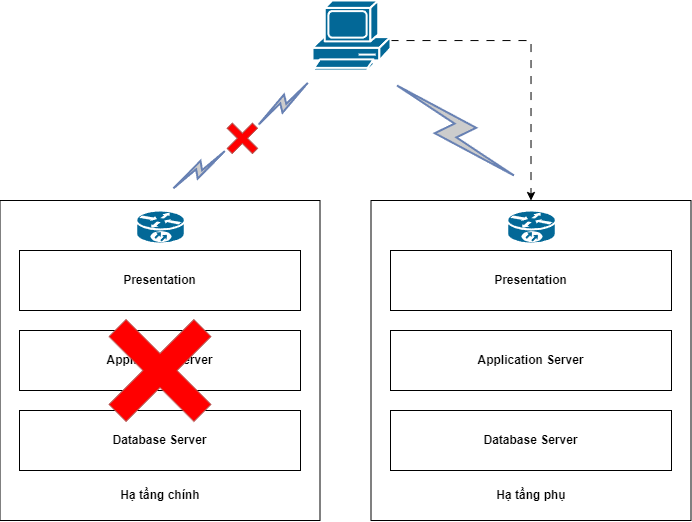
Khi có sự cố, sẽ có ba trường hợp xảy ra: toàn bộ hạ tầng hệ thống chính bị hỏng (bao gồm cả CSDL), một phần hạ tầng hệ thống chính bị hỏng và chỉ máy chủ CSDL bị hỏng. Tùy thuộc vào nhu cầu, chi phí mà doanh nghiệp sẵn sàng bỏ ra cũng như tầm quan trọng của hệ thống ở mức độ nào sẽ có các cách triển khai khác nhau.

Trong hệ thống xếp hạng tín dụng nội bộ của doanh nghiệp được triển khai theo kiến trúc 3 tầng (3-Tiers), bao gồm các tầng như CSDL, tầng xử lý ứng dụng (Application) và tầng giao diện (Client/Presentation). Tầng Presentation sẽ tương tác trực tiếp với người dùng, các yêu cầu từ lớp này sẽ được gửi tới tầng ứng dụng để xử lý nghiệp vụ, cũng như cung cấp các phương thức bảo mật khi giao tiếp với tầng CSDL. Sau đó, dữ liệu từ tầng CSDL sẽ được trả về lớp giao diện thông qua tầng ứng dụng nếu có.

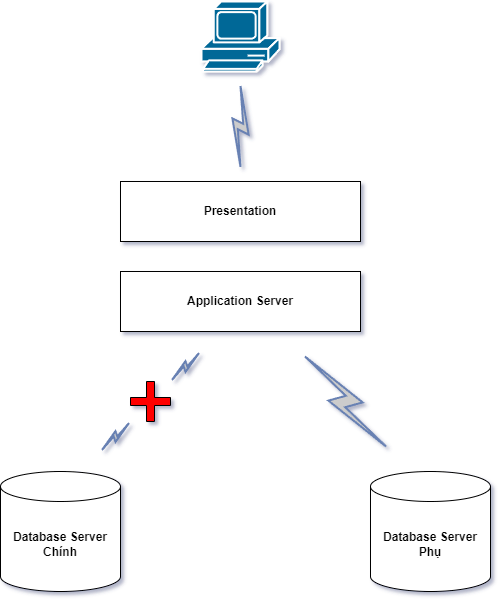
Khi xảy ra sự cố hệ thống, tùy vào việc dự đoán, tính toán sự cố ảnh hưởng cũng như chi phí sẵn sàng bỏ ra để lựa chọn. Có hai mức độ ảnh hưởng nổi bật để ra quyết định lựa chọn phương thức như sau:

|  |  |  |
| --- | --- | --- |
| **Loại sự cố** | **Mô tả** | **Phương thức chuyển đổi** |
| Toàn bộ hạ tầng hệ thống chính gặp sự cố | Hạ tầng hệ thống chính gặp sự cố, gây hỏng/mất kết nối đối với máy chủ chứa lớp ứng dụng với hai lớp giao diện và CSDL. Tất cả máy chủ chứa 3 lớp đều bị ảnh hưởng. | Chuyển đổi toàn bộ hạ tầng hệ thống chính sang hạ tầng hệ thống dự phòng |
| Chỉ hệ thống máy chủ chứa CSDL chính gặp sự cố | Máy chủ CSDL chính gặp sự cố. Các máy chủ khác hoạt động bình thường. Có thể điều hướng máy chủ ứng dụng kết nối tới CSDL khác để tiếp tục vận hành. Chấp nhận việc người dùng cần mất một khoảng thời gian để kết nối lại, và có thể độ trễ khi truy cập ứng dụng cao hơn. | Chỉ chuyển đổi kết nối của hệ thống máy chủ ứng dụng sang hệ thống máy chủ CSDL dự phòng. Phù hợp với mô hình đặt hệ thống máy chủ phân tán, khi hệ thống máy chủ ứng dụng và hệ thống máy chủ giao diện không chung một địa điểm với hệ thống máy chủ CSDL. |

Bảng 4 Các trường hợp chuyển đổi



Hình 20 Phương thức chuyển đổi toàn bộ



Hình 21 Phương thức chuyển đổi chỉ hệ thống CSDL

* + 1. Xác định về đường truyền và đồng bộ dữ liệu

Mô hình Data Guard sử dụng cơ chế đồng bộ dữ liệu thay đổi (redo data) thông qua môi trường mạng, từ CSDL chính tới CSDL phụ. Vì vậy, cần đảm bảo đường truyền hệ thống của doanh nghiệp SHBFinance có đủ khả năng để truyền dữ liệu đồng bộ, tránh gây ra độ trễ/lệch dữ liệu quá lớn.

Độ trễ/lệch dữ liệu đồng bộ xảy ra khi CSDL chính không thể truyền hoặc hạn chế trong việc truyền dữ liệu thay đổi tới CSDL phụ. Khi này, các thông tin dữ liệu thay đổi tại CSDL chính được tạo ra liên tục, nhưng CSDL phụ có thể không nhận được và có nguy cơ gây ra tình trạng mất hoàn toàn dữ liệu thay đổi khi hệ thống CSDL chính gặp sự cố mà không thể khôi phục.

Theo Oracle, để đánh giá mô hình mạng của hệ thống Data Guard, cần cân bằng dựa trên hai khía cạnh là độ tin cậy và băng thông có thể đáp ứng truyền của mạng. Có thể nhắc đến một số tiêu chí để doanh nghiệp cân bằng như:

* Tường lửa và bảo mật đường truyền: sử dụng mã hóa/tường lửa có thể làm chậm lưu lượng truyền/nhận dữ liệu thay đổi. Cần phải cân bằng việc bảo mật dữ liệu cũng như việc hạn chế mất mát dữ liệu.
* Sử dụng cơ chế nén thông tin thay đổi nhằm giảm dung lượng lưu trữ làm chậm việc truyền/nhận. CSDL chính sẽ phải thực hiện nén các tệp thông tin thay đổi trước khi gửi và CSDL phụ sẽ phải thực hiện giải nén trước khi áp dụng.
* Bảo mật dữ liệu với tính năng mã hóa dữ liệu dưới đĩa (Transparent Data Encryption) cũng sẽ làm ảnh hưởng tới tốc độ truyền/nhận trong môi trường Oracle Net. Do các dữ liệu được truyền cũng được mã hóa khi rời khỏi CSDL truyền và chỉ được giải mã khi đến đích.
* Tối ưu hóa các thông số liên quan tới hệ thống mạng như chỉ số Maximum Transmission Unit (MTU) – kích thước tối đa của một gói tin dữ liệu trong giao thức mạng TCP/IP (hoặc giao thức khác), đo lường số byte tối đa mà một gói tin có thể chứa trước khi gửi qua mạng, giảm thiểu tình trạng phân mảnh gói tin.
  + 1. Xác định chế độ bảo vệ trong Data Guard

Trong 03 chế độ bảo vệ của Data Guard, cần lựa chọn chế độ để phù hợp với yêu cầu vận hành của doanh nghiệp. Các chế độ có độ ưu tiên khác nhau về hiệu năng, tính sẵn sàng của hệ thống và mức độ mất mát dữ liệu.

*Với chế độ Ưu tiên bảo vệ (max. protection):*

Hệ thống đảm bảo sẽ không có dữ liệu đã thay đổi nào được lưu bị mất khi hệ thống CSDL chính gặp sự cố, kể cả khi kết nối giữa CSDL chính và CSDL phụ bị hỏng và hệ thống CSDL chính hỏng. Chế độ này thực hiện cơ chế chỉ xác nhận một giao dịch đã được COMMIT khi và chỉ khi ít nhất một CSDL phụ trả lại tín hiệu ACK cho CSDL chính rằng dữ liệu thay đổi đã được ghi và áp dụng vào CSDL phụ. Ngược lại, nếu không có bất kỳ tín hiệu nào trở về, CSDL chính sẽ treo và dừng hoạt động để đảm bảo tính toàn vẹn khi giao dịch chưa được COMMIT ở cả hai CSDL.

Để dự phòng trong trường hợp CSDL dự phòng không hoạt động, doanh nghiệp nên thực hiện triển khai tối thiểu hai CSDL dự phòng trong chế độ Ưu tiên bảo vệ. Với giải pháp này sẽ hạn chế việc CSDL chính rơi vào trạng thái chờ đợi, dẫn đến tự động dừng hoạt động khi ít nhất một trong hai CSDL phụ nhận và trả lại tín hiệu ACK cho CSDL chính. Chế độ này phù hợp với nhu cầu ưu tiên về tính toàn vẹn dữ liệu hơn là tính sẵn sàng của CSDL.

*Với chế độ Ưu tiên tính sẵn sàng (max. availability):*

Hệ thống đảm bảo rằng cũng sẽ không/hạn chế việc dữ liệu thay đổi được lưu bị mất khi CSDL chính gặp sự cố. Trong chế độ này, CSDL chính sẽ chờ đến thời gian tối đa được cấu hình trong biến NET\_TIMEOUT khi chờ tín hiệu ACK phản hồi lại từ CSDL phụ, nếu nhận được tín hiệu, CSDL chính có thể đánh dấu COMMIT và tiếp tục một giao dịch mới. Trong trường hợp không nhận được tín hiệu, CSDL chính vẫn hoạt động và COMMIT như bình thường, nhưng các tiến trình nền sẽ liên tục gửi và cập nhật trạng thái của CSDL phụ để có thể kết nối lại và đồng bộ hóa, khi này dữ liệu có thể bị mất khi CSDL chính gặp sự cố.

Để dự phòng trường hợp mất kết nối, doanh nghiệp nên kết hợp thêm tính năng Far Sync với một CSDL trung gian, đứng giữa CSDL chính và CSDL phụ để trung chuyển các dữ liệu thay đổi. CSDL Far Sync sẽ có đường truyền tốt hơn để đảm bảo không mất kết nối với CSDL chính, gây ra tình trạng mất dữ liệu. Ngoài ra, cũng luôn phải theo dõi các tiến trình, đường truyền để xử lý các sự cố gây ra độ trễ trong việc đồng bộ. Chế độ này phù hợp với nhu cầu muốn cân bằng về tính toàn vẹn của dữ liệu cũng như tính sẵn sàng của CSDL chính.

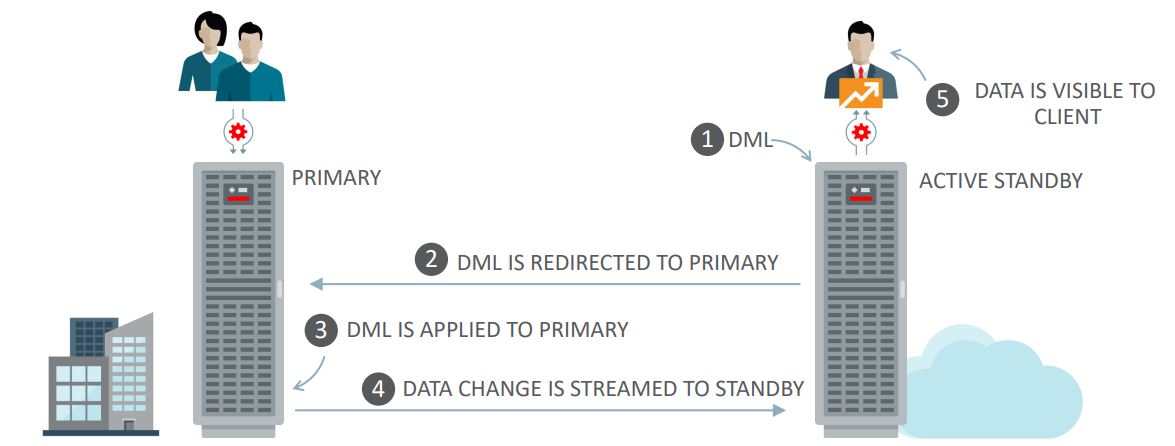
*Với chế độ Ưu tiên hiệu năng:*

Chế độ này hạn chế việc ảnh hưởng tới hiệu năng hệ thống, trong khi các dữ liệu thay đổi vẫn được chuyển đi và đồng bộ hóa. Một giao dịch được tạo ra và được COMMIT ngay khi có tín hiệu của người dùng, được khi vào Online Redo Log như khi không có CSDL phụ. Hệ thống Data Guard sẽ truyền dữ liệu thay đổi song song tới: Standby Redo Log của CSDL phụ, trực tiếp từ Log Buffer (đối với đường truyền tốt) (1), tới Archive Redo Log của CSDL chính (2) theo cơ chế bất đồng bộ với giao dịch được COMMIT, hạn chế việc mất mát dữ liệu khi CSDL chính xảy ra sự cố.

Chế độ này phù hợp đối với yêu cầu không đặt nặng vấn đề mất mát một lượng dữ liệu nhỏ và yêu cầu hiệu năng hệ thống chính cần hoạt động với hiệu năng cao.

* + 1. Xác định việc giảm tải hệ thống chính

Với tính năng Active Data Guard và DML Redirection trên giải pháp Data Guard, nên tận dụng và sử dụng để giảm tải khối lượng công việc cho hệ thống CSDL chính. Active Data Guard là tính năng cho phép CSDL phụ vừa hoạt động đồng bộ hóa dữ liệu thay đổi với CSDL chính và cho phép người dùng thực hiện hoạt động truy vấn mà không gây bất kỳ gián đoạn nào. DML Redirection thực hiện việc truyền tải và áp dụng lại những câu lệnh DML tại CSDL phụ tới CSDL chính, đây là một tính năng mới trong phiên bản Oracle Database 19c.



Hình 22 Minh họa hoạt động của DML Redirection

Tuy nhiên, để giảm tải, tận dụng có hiệu quả các tính năng trên, cần xem xét một số vấn đề sau:

* Lựa chọn, xem xét mô-đun nào của ứng dụng có công việc truy vấn dữ liệu với các câu lệnh truy vấn và tạo báo cáo là hoạt động chính
* Với các mô-đun của ứng dụng sử dụng nhiều câu lệnh DML, yêu cầu về tính kịp thời, đồng nhất thì không nên sử dụng đối với CSDL phụ trong tính năng DML Redirection, do yêu cầu cơ chế xử lý đồng nhất rất phức tạp trong trường hợp băng thông không tốt
* Đánh giá hiệu năng hệ thống sau khi đã giảm tải bớt khối lượng công việc từ CSDL chính, xem xét đã phù hợp với nhu cầu/tiêu chuẩn của doanh nghiệp. Giám sát sau khi đưa vào vận hành.
  + 1. Xác định yêu cầu phần cứng, phần mềm

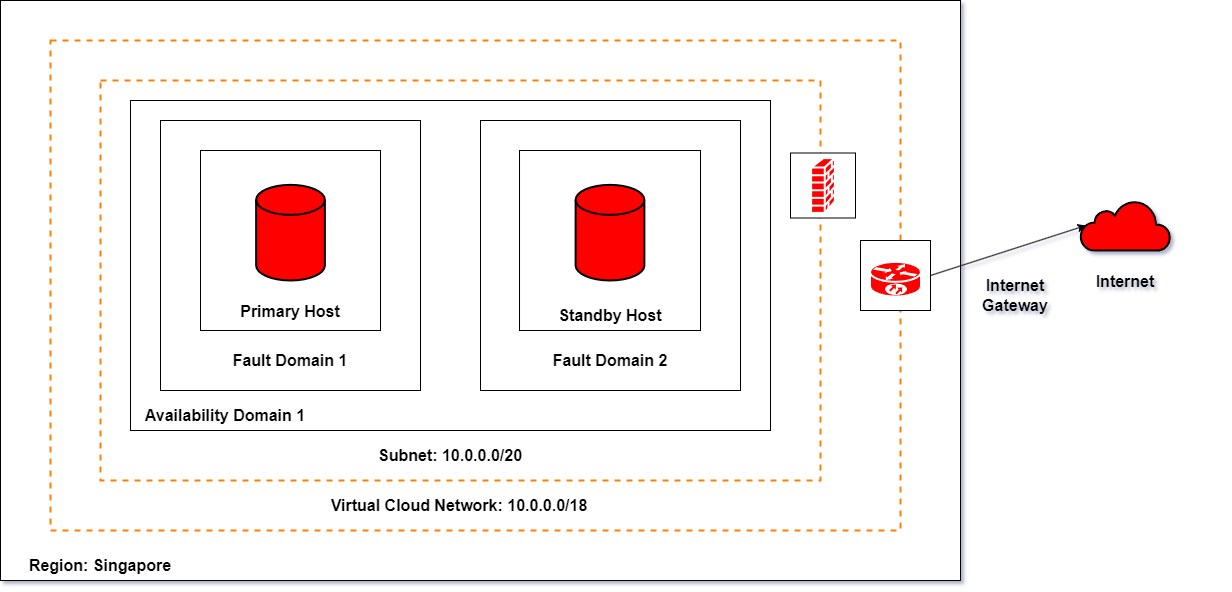
*Với phần cứng:*

* Phần cứng của CSDL chính và CSDL phụ có thể khác nhau về số lượng bộ xử lý chung tâm (CPU), kích thước bộ nhớ (Memory), và cấu hình lưu trữ (Storage)
* Cho phép hệ thống xử lý cũng như phiên bản cài đặt Oracle Database Software có kích thước đơn vị biểu diễn thông tin khác nhau (32-bit hoặc 64-bit)

Trong trường hợp CSDL chính và CSDL phụ cùng một máy chủ (local), cần đảm bảo rằng hệ thống được cấu hình đúng cách để hai CSDL có cùng DB\_NAME có thể hoạt động ổn định, không gây ra xung đột.

*Với phần mềm:*

* Yêu cầu phiên bản cài đặt cho Oracle Database từ Enterprise Edition trở lên cho cả hệ thống CSDL chính và CSDL phụ. Data Guard không hỗ trợ cho Oracle Database Standard Edition
* Nếu sử dụng công cụ quản lý bộ nhớ Automatic Storage Management (ASM) hoặc Oracle Managed Files (OMF) thì cần sử dụng giống nhau ở cả hai hệ thống CSDL chính và CSDL phụ. Đối với trường hợp kết hợp các phương thức thì cũng tương tự ở cả hai hệ thống
  1. Thực nghiệm triển khai giải pháp Data Guard dựa trên RMAN Duplicate và nền tảng điện toán đám mây Oracle Cloud Infrastructure
     1. Kiến trúc tổng quan

****

Hình 23 Kiến trúc tổng quan thực nghiệm giải pháp Data Guard trên OCI

*Region:* Là một khu vực địa lý, nơi đặt hạ tầng công nghệ thống tin trải dài trên lãnh thổ địa lý đó, cung cấp nền tảng mạng và tài nguyên cho các ứng dụng, dịch vụ trên điện toán đám mây mà người dùng có thể sử dụng. Mỗi khu vực chứa các hạ tầng công nghệ thông tin này hoàn toàn độc lập về mặt giao tiếp mạng cũng như về vị trí địa lý với các khu vực khác. Thông thường, các ứng dụng sẽ triển khai tại khu vực có nhiều lưu lượng sử dụng nhất để tăng tốc độ truy cập cho người dùng cuối. Tại các khu vực khác nhau, các dịch vụ có thể triển khai cũng có thể khác nhau do phụ thuộc nhu cầu và hạ tầng công nghệ thông tin, có thể kể đến một số dịch vụ như: Máy tính, Bộ nhớ, Mạng truyền thông, CSDL/Kho dữ liệu, Máy chủ biên dịch tên miền (DNS), Bảo mật.

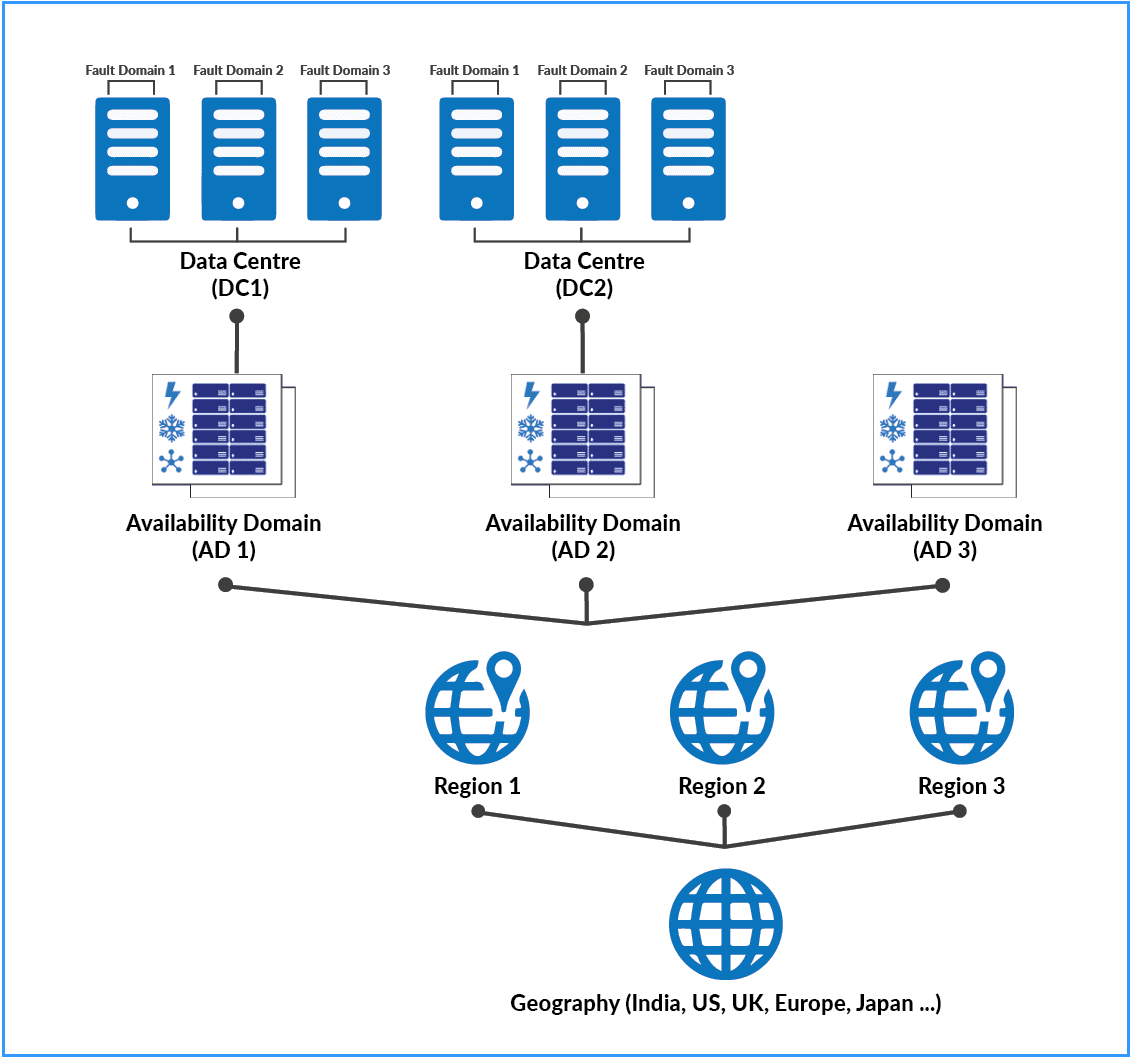
*Availability Domain (AD):* Là một hoặc nhiều tập hợp hạ tầng công nghệ thông tin trong cùng một khu vực. Mỗi khu vực thường có nhiều nhất 03 AD, mỗi AD được cách ly với nhau về mặt hạ tầng như hệ thống điện, hệ thống làm mát hoặc mạng nội bộ trong cùng một AD, nên có khả năng chịu lỗi và khó có thể gây ảnh hưởng tới các AD khác. Tất cả các AD trong cùng một khu vực được kết nối với nhau bằng đường truyền có băng thông cao, phù hợp cho việc xây dựng giải pháp dự phòng và sẵn sàng cao. Chỉ có một số khu vực Châu Mỹ và Châu Âu, Oracle mới thành lập nhiều AD, tại Châu Á, Singapore chỉ có 01 AD trong trường hợp này.

*Fault Domain (FD):* Là một tập hợp bao gồm các phần cứng và hạ tầng trong trong một AD. Mỗi AD có 03 FD. Được thiết kế để bảo vệ khỏi sự cố của các thành phần vật lý trong cùng một AD khi ứng dụng/dịch vụ được triển khai. Nếu một ứng dụng/dịch vụ được triển khai gặp lỗi hoặc phần cứng ứng dụng đó cần bảo trì, thì ứng dụng/dịch vụ khác được triển khai tại một FD khác sẽ không bị ảnh hưởng về mặt vật lý. Các thành phần vật lý của mỗi FD được tách biệt và đều có nguồn điện dự phòng, phòng chống việc mất điện ảnh hưởng tới các FD khác.

*Virtual Cloud Network (VCN) và Subnet:* là dịch vụ cung cấp loại mạng ảo, nội bộ giống với mạng truyền thống, có cung cấp tường lửa, các loại hình gateway, được thiết lập ảo hóa dựa trên hạ tầng công nghệ thông tin. Một VCN nằm trong một Region và có thể bao quát hết các đối tượng, dịch vụ/ứng dụng dựa trên tham số CIDR (Classless Inter-Domain Routing) – cấu hình tập hợp các địa chỉ IP có chung tiền tố mạng và số lượng máy đã thiết lập.

Mỗi mạng con – subnet bao gồm một dãy IP liền kề, các dãy IP này không trùng với nhau đối với các subnet khác trong cùng một VCN. Một subnet có thể được thiết kế để dùng cho một AD hoặc giữa các Region khác nhau. Các dịch vụ, ứng dụng trong cùng một VCN sử dụng chung một bảng định tuyến, quy định về bảo mật, tường lửa và máy chủ cấp phát IP động. Subnet có thể đặt ở chế độ nội bộ (private) hoặc công khai (public).

Với mạng công khai, các dịch vụ sẽ được gán một IPv4 công khai bên cạnh IPv4 nội bộ, và có thể giao tiếp với Internet. Ngoài ra, Subnet cũng có hai loại cấp độ về phạm vi Region hoặc AD. Với subnet cấp Region, các máy chủ tạo trong bất kỳ AD nào cùng một Region đều có thể giao tiếp với nhau; với AD, máy chủ chỉ có thể giao tiếp nội bộ khi trong cùng một AD.



Hình 24 Minh họa phân cấp giữa Region, AD và FD

Với Region là Singapore chỉ có một AD. Vì vậy để thực nghiệm giải pháp dự phòng Data Guard hợp lý, đảm tối thiểu về tính chống chịu lỗi, tách biệt vật lý, triển khai theo hướng mỗi máy chủ đặt tại một FD khác nhau trong cùng một AD là phù hợp với hoàn cảnh. Cấu hình về mạng lưới và các máy ảo được cài đặt như sau:

|  |  |
| --- | --- |
| **Thành phần** | **Thông tin** |
| **VCN** | IPv4 CIDR Blocks: 10.0.0.0/8 DNS Hostname: Yes |
| **Route Table** | Destination: 0.0.0.0/0 Gateway: Internet Gateway |
| **Security List** | *Ingress Rule:*   * Source: 0.0.0.0/0, IP Protocol: TCP, Destination Port Range: 22 (SSH) * Source: 0.0.0.0/0, IP Protocol: ICMP (ping) * Source: 0.0.0.0/0, IP Protocol: TCP, Destination Port Range: 1521 (Oracle Database) |
| **Subnet** | IPv4 CIDR Blocks: 10.0.0.0/20  Security Lists: Default  Subnet Access: Public Subnet  Subnet Type: Regional  DNS Hostname: Yes |

Bảng 5 Thiết lập cấu hình mạng lưới ảo OCI

|  |  |  |
| --- | --- | --- |
| **Máy chủ** | **Thông tin** | **Giá trị** |
| **Chính** | Oracle CPU | 1 |
| Memory (Gigabyte) | 6 |
| Availability Domain | AD-1 |
| Fault Domain | FD-1 |
| Subnet | 10.0.0.0/20 |
| Private IP | 10.0.12.202 |
| Public IP | 213.35.102.135 |
| Region | Singapore |
| Hostname | source |
| Operating System | Oracle Linux 7.9 |
| Storage (Gigabyte) | 50 |
| **Dự phòng/Phụ** | Oracle CPU | 1 |
| Memory (Gigabyte) | 6 |
| Availability Domain | AD-1 |
| Fault Domain | FD-2 |
| Subnet | 10.0.0.0/20 |
| Private IP | 10.0.15.63 |
| Public IP | 129.150.61.43 |
| Region | Singapore |
| Hostname | target |
| Operating System | Oracle Linux 7.9 |
| Storage (Gigabyte) | 50 |

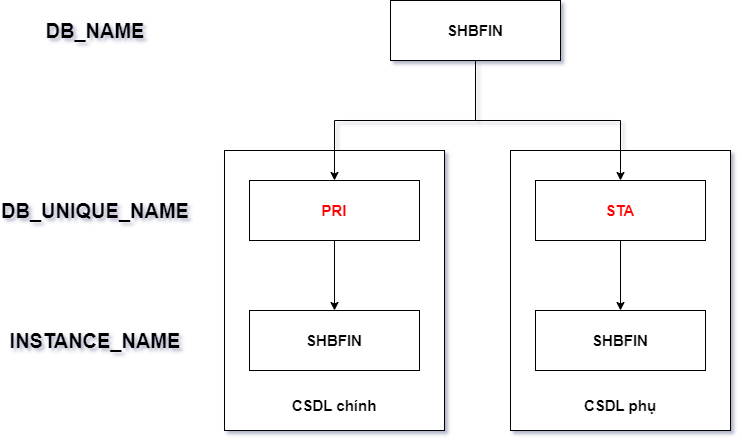
Bảng 6 Cấu hình máy chủ ảo cài đặt Oracle Data Guard trên OCI

Để triển khai giải pháp Oracle Data Guard, cả hai máy chủ cần được cài đặt Oracle Database Software phiên bản Enterprise Edition, phần cài đặt sẽ được để ở Phụ lục, hiện tại, bài đi theo phiên bản Oracle Database 19c nhằm tận dụng một số tính năng mới của Data Guard cũng như đảm bảo sự ổn định tới thời điểm năm 2024 này. Ngoài việc cài đặt thành công Software, máy chủ chính cũng cần cài đặt một Database trước. Trong phần thực nghiệm này, máy chủ chính đã được cài đặt Database dạng giao dịch (OLTP), kiến trúc Single Instance Database nhằm thực hiện đơn giản hóa việc thực nghiệm, thay vì kiến trúc Real Application Cluster (RAC). Ngoài ra, thiết lập phân quyền và đường dẫn ở cả hai hệ thống đều tương tự nhau.

Các phần tiếp theo sẽ đi sâu về cách triển khai và thiết lập Oracle Data Guard. Trong một số bước nhỏ, các câu lệnh xuất hiện trong bài sẽ không đầy đủ, thay vào đó là nêu lên ý tưởng, khái niệm để người đọc có thể hình dung rõ hơn. Lệnh và các cấu hình đầy đủ sẽ được đính kèm theo phụ lục tại cuối bài.

* + 1. Môi trường Oracle Net và định danh CSDL

Trong hệ thống áp dụng giải pháp Oracle Data Guard, các CSDL đều có một tên giống nhau được thiết lập trong tham số DB\_NAME. Tuy nhiên, khi tham gia vào môi trường mạng, để nhận diện đâu là CSDL chính/phụ, các CSDL cần có cấu hình tên định danh khác nhau trong tham số DB\_UNIQUE\_NAME. Khi này, hệ thống vừa đảm bảo các ứng dụng chỉ giao tiếp với CSDL thông qua tên chung là DB\_NAME, và vừa đảm bảo về mặt dự phòng chuyển đổi, giao tiếp giữa CSDL chính và phụ với nhau thông qua DB\_UNIQUE\_NAME, vẫn giữ được tính nhất quán và phân biệt.



Hình 25 Minh họa phân cấp tên trong hệ thống CSDL cho SHBFinance

Thiết lập và vận hành được kiến trúc Data Guard phụ thuộc chủ yếu vào cách cấu hình môi trường Oracle Net làm sao để cả hai hệ thống CSDL chính và phụ đều có thể giao tiếp với nhau. CSDL Oracle sử dụng dịch vụ Listener như một “gateway” trong thiết bị mạng, điều hướng kết nối từ các thiết bị liên lạc với nó tới CSDL để thiết lập phiên làm việc (sessions) dành cho người dùng.

Các CSDL trong môi trường Data Guard cũng sử dụng Listener để có thể kết nối với nhau và truyền tải thông tin, đồng bộ thay đổi dữ liệu, có thể hiểu khi CSDL chính cần truyền tài thông tin đi tới CSDL phụ, thì CSDL chính là đối tượng chủ động, một “client”, cần tìm Listener để kết nối và chuyển tới.

Thông thường, khi một client kết nối tới máy chủ CSDL, sẽ cần mô tả chuỗi kết nối, và chuỗi kết nối rất dài, bao gồm: địa chỉ IP hoặc tên host (nếu đã khai báo IP trong tệp hosts hoặc có DNS Name trong subnet), cổng port mở và tên nhận dạng của Instance kết nối tới. Ví dụ như:

*SQL> CONNECT* [*username/password@123.456.789.000:1521/shbfin*](mailto:username/password@123.456.789.000:1521/shbfin)

Các CSDL cũng như cấu hình liên quan trong Data Guard sử dụng một phương thức đơn giản hóa chuỗi kết nối trên bằng việc chứa thông tin chuỗi vào một tên bí danh (alias), tính năng này được gọi là Local Naming Method. Thông tin chuỗi kết nối này sẽ được cấu hình đưa vào bí danh trên mỗi máy khách, mỗi khi kết nối, chỉ cần sử dụng bí danh này là hệ thống có thể biên dịch sang chuỗi truyền thống:

*SQL> CONNECT username/password@pri*

*# pri chứa thông tin IP, port, instance cần kết nối*

Bước đầu trong quá trình triển khai kiến trúc Data Guard là cấu hình Listener cho mỗi hệ thống CSDL. Cùng với đó là cấu hình phương thức Local Naming Method để thực hiện việc đơn giản hóa, sử dụng bí danh để các CSDL giao tiếp với nhau. Tại cả hai máy chủ CSDL, thực hiện việc cấu hình Listener có dạng như sau:

*<listener\_name> =*

*(DESCRIPTION\_LIST =*

*(DESCRIPTION =*

*(ADDRESS = (PROTOCOL = TCP)(HOST = <host\_name>)(PORT = 1521))*

*(ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))*

*)*

*)*

*SID\_LIST\_<listener\_name> =*

*(SID\_LIST =*

*(SID\_DESC = (GLOBAL\_DBNAME = shbfin) (ORACLE\_HOME = /u01/app/oracle/product/19.0.0/dbhome\_1) (SID\_NAME = shbfin))*

*)*

*<listener\_name>*: tên của Listener

*DESCRIPTION\_LIST*: chứa danh sách mô tả các kết nối đến mà Listener sẽ xử lý. Tại đây quan tâm đến giao thức TCP dành cho kết nối từ các ứng dụng và giữa các Database với nhau. Với giao thức ICP (Inter-Process Communication), dành cho các ứng dụng cùng trên máy chủ chứa CSDL có thể kết nối nội bộ với nhau.

*SID\_LIST\_<listener\_name>:* chứa danh sách mô tả các CSDL mà Listener sẽ điều hướng kết nối của người dùng tới CSDL đó.

Cấu hình Local Naming Method cho hai máy chủ chứa CSDL có dạng như sau:

*<alias\_primary\_name> =*

*(DESCRIPTION = (ADDRESS = (PROTOCOL = TCP)(HOST = <host\_name>)(PORT = 1521))*

*(CONNECT\_DATA = (SERVER = DEDICATED) (SID = shbfin)))*

*<alias\_standby\_name> =*

*(DESCRIPTION = (ADDRESS = (PROTOCOL = TCP)(HOST = <host\_name>)(PORT = 1521))*

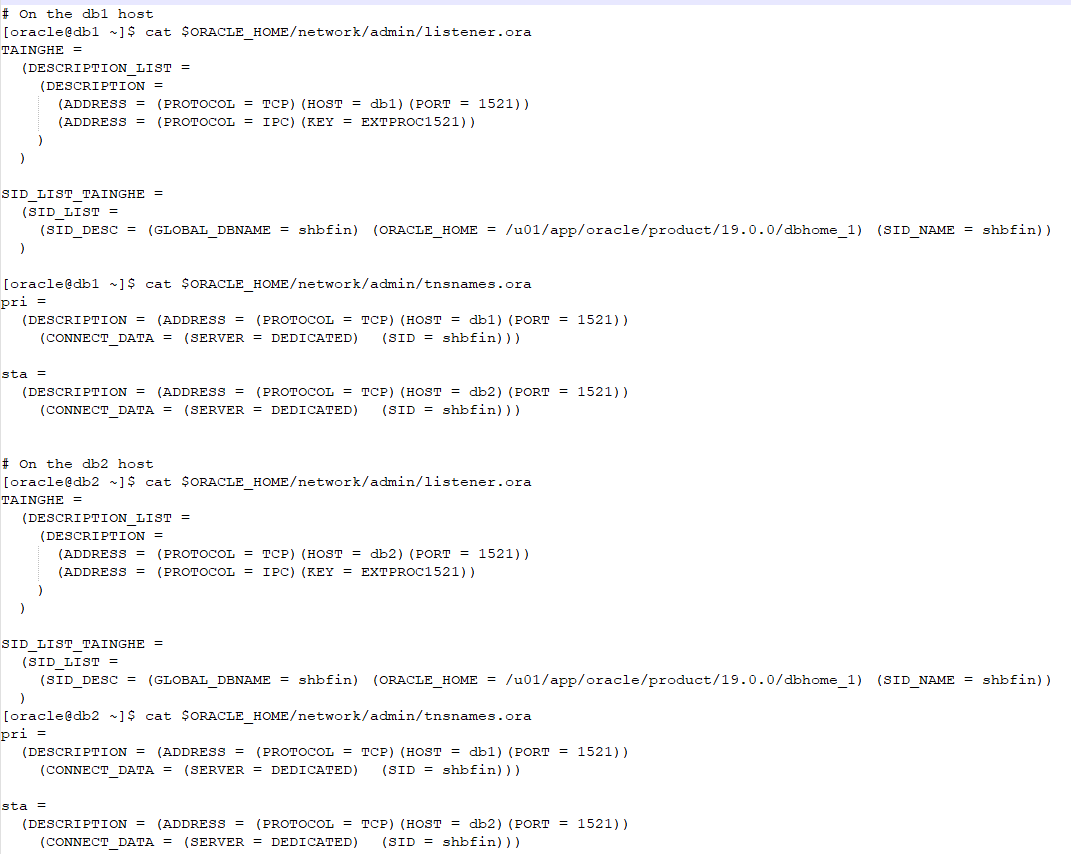
*(CONNECT\_DATA = (SERVER = DEDICATED) (SID = shbfin)))*

*<alias\_primary\_name>:* tên bí danh được sử dụng để mô tả kết nối

*DESCRIPTION*: mô tả kết nối, bao gồm địa chỉ và dữ liệu kết nối

*CONNECT\_DATA:* chỉ định kết nối sẽ theo phương thức nào và kết nối vào CSDL nào. Trong trường hợp này là phương thức DEDICATED – mỗi client kết nối vào sẽ có một tiến trình nền hỗ trợ riêng với bộ nhớ dành cho việc lưu trữ kết quả của SQL riêng và kết nối vào CSDL với Instance là shbfin.

Cấu hình cụ thể tại CSDL chính và phụ như sau:



Hình 26 Cấu hình Listener, Local Naming Method tại hai máy chủ CSDL

* + 1. Cấu hình tham số chung cho hệ thống chính

*Bật chế độ FORCE LOGGING:* Trong chế độ FORCE LOGGING, mọi thay đổi tại CSDL, cụ thể là trong Redo Buffer Cache đều được ghi xuống thiết bị đĩa cứng tại Online Redo Logs theo cơ chế xoay vòng (ghi đè khi hết) bằng tiến trình LGWr, bởi vậy mà CSDL có thể khôi phục được những thay đổi đã COMMIT sau khi xảy ra sự cố. FORCE LOGGING đảm bảo tính nhất quán của dữ liệu. Ngoài ra cũng có một tùy chọn cấu hình nữa là LOGGING, tuy nhiên, mức độ áp đặt lên các thay đổi của chế độ này không được ưu tiên đối với các đối tượng khác trong CSDL được đặt NOLOGGING (không ghi lại thay đổi). Kết quả là có rất nhiều thay đổi không được đồng bộ hóa sang CSDL phụ dẫn đến khi thực hiện chuyển đổi vai trò CSDL sẽ gây lỗi do chênh lệch thông tin.

*Bật chế độ ARCHIVELOG:* Khi một Online Redo Logs đầy về mặt dung lượng, chế độ ARCHIVELOG sẽ thực hiện lưu trữ tệp tin này thông qua việc sao chép bằng tiến trình ARCn với điều kiện Online Redo Logs thực hiện cơ chế Log Switch để thực hiện chuyển qua tệp tin khác lưu trữ. Trạng thái của Online Redo Logs (ORLs) thời điểm này sẽ chuyển từ CURRENT qua ACTIVE cho đến khi được lưu trữ thành công, trở về trạng thái INACTIVE.

Tại trạng thái ACTIVE, có thể hiểu đơn giản là Online Redo Logs chưa được lưu trữ bằng tiến trình ARCn, tiến trình checkpoint chưa xảy ra và các thông tin thay đổi trong Active Logs đó sẽ được sử dụng cho quá trình Crash Recovery – khôi phục hệ thống sau khi lỗi tắt nguồn đột ngột. Khi đã hoàn toàn được lưu trữ, trạng thái của ORLs này sẽ trở về InActive, tiếp tục được ghi đè và sử dụng lại cho những lần sau. Nhờ có Archive Log mà việc giải quyết độ trễ dữ liệu trong hệ thống Data Guard trở nên hiệu quả hơn thông qua tiến trình FAL như đã nêu ở khái niệm, cũng như việc khôi phục lại dữ liệu bị ghi đè ở ORLs.

*Bật chế độ FLASHBACK:* công nghệ Flashback là một phần không thể thiếu trong việc thiết lập chế độ chuyển vai trò tự động – Fast-start Failover trong trường hợp CSDL chính gặp sự cố không tính trước. Khi thực hiện chuyển đổi failover, CSDL chính (cũ) sẽ rơi vào trạng thái tắt nguồn, gặp lỗi và mất đồng bộ với CSDL chính (là CSDL phụ dự phòng trước đây), trạng thái hiển thị sẽ là “needs Re-instatement” mà không cần tạo lại CSDL chính (cũ).

Flashback dựa vào chỉ số SCN của checkpoint trước thời điểm gặp lỗi và lượng thời gian dự phòng sao lưu được cấu hình bằng tham số DB\_FLASHBACK\_RETENTION\_TARGET – khoảng thời gian mà hệ thống liên tục sao lưu và giữ lại các redo logs, undo data và flashback log để có thể khôi phục trong khoảng thời gian đã chỉ định. Nhờ công nghệ này mà quản trị viên không cần phải khôi phục lại nếu gặp sự cố trong thời gian ngắn, kể cả những sự cố nhỏ do lỗi người dùng như DROP.

*# Truy vấn CSDL đang ở chế độ nào*

*SQL> SELECT NAME, DB\_UNIQUE\_NAME, OPEN\_MODE, LOG\_MODE, FLASHBACK\_ON, FORCE\_LOGGING FROM V$DATABASE;*

*# Thực hiện bật chế độ FORCE LOGGING, ARCHIVELOG và FLASHBACK*

*-- Archive Log Mode*

*SQL> SHUTDOWN IMMEDIATE;*

*SQL> STARTUP MOUNT;*

*SQL> ALTER DATABASE ARCHIVELOG;*

*SQL> ALTER DATABASE OPEN;*

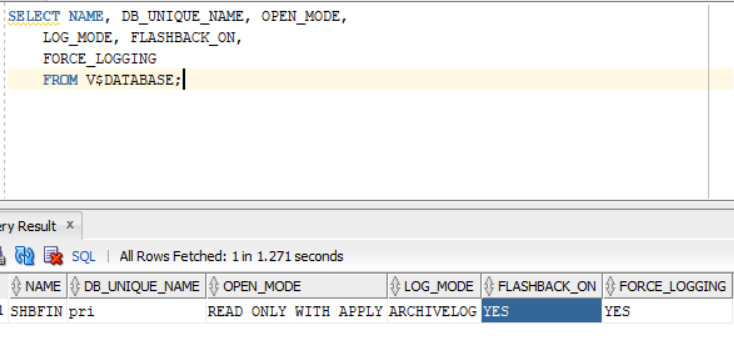
*-- Force Logging Mode*

*SQL> ALTER DATABASE FORCE LOGGING;*

*-- Flashback Mode*

*SQL> ALTER SYSTEM SET DB\_FLASHBACK\_RETENTION\_TARGET = 60 SCOPE=BOTH;*

*SQL> ALTER DATABASE FLASHBACK ON;*



Hình 27 Kết quả kiểm tra sau khi bật các chế độ cần thiết

*Tạo Standby Redo Logs:* Standby Redo Logs (SRLs) được dùng khi vai trò của CSDL là dự phòng/phụ, nhận thông tin đồng bộ dữ liệu thay đổi từ CSDL chính. Cần tạo SRLs ở cả hai CSDL chính và phụ, dự phòng trong việc chuyển đổi để chúng có thể nhận và áp dụng thay đổi vào CSDL. Có một số điều kiện bắt buộc khi tạo SRLs như sau: cần tạo nhiều hơn ít nhất 01 groups so với groups của ORLs tại CSDL chính (1), SRLs cần lớn hơn hoặc bằng ORLs của CSDL chính (2). Nếu SRLs được cấu hình sai với các điều kiện đã nêu, tiến trình RFS sẽ ghi vào Archive Redo Log (ARL), mất đi tính năng Real-Time Apply và gây ra hiện tượng trễ.

*# Kiểm tra dung lượng Online Redo Logs hiện tại theo megabyte*

*SQL> select GROUP#,THREAD#,SEQUENCE#,bytes/1024/1024 from v$log;*

*# Tạo Standby Redo Logs phù hợp với yêu cầu đề ra*

*ALTER DATABASE ADD STANDBY LOGFILE THREAD 1 GROUP 4 ('/u02/oradata/shbfin/stb\_redo04.log') SIZE 200M;*

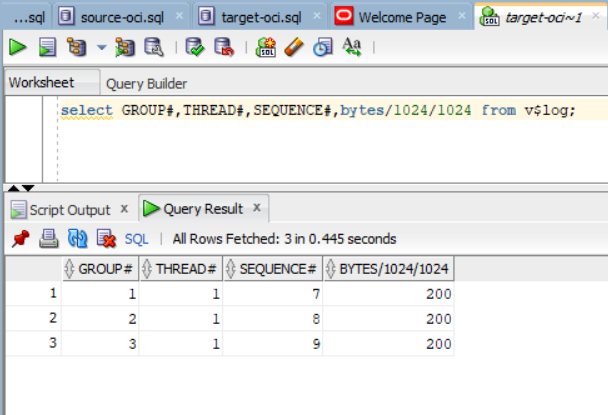
*ALTER DATABASE ADD STANDBY LOGFILE THREAD 1 GROUP 5 ('/u02/oradata/shbfin/stb\_redo05.log') SIZE 200M;*

*ALTER DATABASE ADD STANDBY LOGFILE THREAD 1 GROUP 6 ('/u02/oradata/shbfin/stb\_redo06.log') SIZE 200M;*

*ALTER DATABASE ADD STANDBY LOGFILE THREAD 1 GROUP 7 ('/u02/oradata/shbfin/stb\_redo07.log') SIZE 200M;*

*# Kiểm tra lại các loại logs hiện tại mà Oracle Database đang sử dụng*

*SELECT TYPE, MEMBER FROM V$LOGFILE ORDER BY GROUP#;*

**

Hình 28 Dung lượng của các Online Redo Logs có trong CSDL hiện tại

*Hình: Các loại file logs sau khi tạo Standby Redo Logs có trong CSDL hiện tại*

*Thiết lập vị trí lưu trữ Redo Log cục bộ:* Trong môi trường Data Guard, Redo Transport Services được cài đặt, điều khiển bằng tham số LOG\_ARCHIVE\_DEST\_n. Tham số này cho phép redo data vừa được gửi đồng bộ sang Standby Database, vừa được lưu trữ xuống đĩa. Cụ thể, dạng tổng quát thường dùng của tham số này như sau:

*LOG\_ARCHIVE\_DEST\_[1|2|3|…|31] =*

*‘ LOCATION=path\_name | SERVICE=service\_name*

*SYNC|ASYNC*

*AFFIRM | NOAFFIRM*

*VALID\_FOR=(redo\_log\_type,database\_role)*

*DB\_UNIQUE\_NAME=db\_unique\_name*

*…*

*’*

N là thông tin xác định Redo Transport Services sẽ chuyển redo data xuống cục bộ hay đi sang Standby Redo Logs. Trong trường hợp cấu hình cục bộ, n luôn phải đặt là 1, LOCATION sẽ được đặt giá trị là một đường dẫn của máy chủ cài đặt CSDL chính, SERVICE\_NAME dùng khi gửi sang Standby.

Cách truyền phụ thuộc rất nhiều vào kiểu truyền SYNC/ASYNC và AFFIRM/NOAFFIRM, mặc định khi thiết lập AFFIRM thì sẽ thiết lập SYNC. Với tham số VALID\_FOR gồm hai đối số đầu vào, khi CSDL có vai trò là *database\_role* thì sẽ lưu trữ *redo\_log\_type* xuống hoặc gửi redo\_log\_type đi cho CSDL phụ. Trong cài đặt cục bộ, thông tin cài đặt sẽ như sau:

*SQL> ALTER SYSTEM SET LOG\_ARCHIVE\_DEST\_1=*

*'LOCATION=/u02/oradata/shbfin/arch1/*

*VALID\_FOR=(ALL\_LOGFILES,ALL\_ROLES)*

*DB\_UNIQUE\_NAME=pri' scope=spfile;*

Với cài đặt này, dù CSDL ở vai trò chính hoặc vai trò phụ, các redo data được lưu trữ trong ORLs hoặc SRLs đều được sao chép và cất giữ theo đường dẫn đã cấu hình tại LOCATION.Bảng kết hợp cho tham số VALID\_FOR như sau, X là hợp lệ:

|  |  |  |  |
| --- | --- | --- | --- |
| **Kết hợp** | **CSDL Chính** | **CSDL Phụ (Physical)** | **CSDL Phụ (Logical)** |
| ONLINE\_LOGFILE, PRIMARY\_ROLE | X |  |  |
| ONLINE\_LOGFILE, STANDBY\_ROLE |  |  | X |
| ONLINE\_LOGFILE, ALL\_ROLES | X |  | X |
| STANDBY\_LOGFILE, STANDBY\_ROLES |  | X | X |
| STANDBY\_LOGFILE, ALL\_ROLES |  | X | X |
| ALL\_LOGFILES, PRIMARY\_ROLE | X |  |  |
| ALL\_LOGFILES, STANDBY\_ROLE |  | X | X |
| ALL\_LOGFILES,  ALL\_ROLES | X | X | X |

Bảng 7 Kết hợp đối số trong VALID\_FOR

Ngoài ra, còn một số tham số phụ trợ khác cho CSDL chính được cấu hình như sau:

*# Cấu hình CSDL sẽ sử dụng Listener nào làm bộ điều phối mạng*

*SQL> ALTER SYSTEM SET LOCAL\_LISTENER='(ADDRESS=(PROTOCOl=TCP)(HOST=source)(PORT=1521))' SCOPE=SPFILE;*

*# Cấu hình số lượng tiến trình ARCn*

*SQL> ALTER SYSTEM SET LOG\_ARCHIVE\_MAX\_PROCESSES=30 SCOPE=SPFILE;*

*# Cấu hình định dạng tên cho các Archive Log File*

*SQL> ALTER SYSTEM SET LOG\_ARCHIVE\_FORMAT='ora\_%t\_%s\_%r.arc' SCOPE=SPFILE; -- %t: thread, %s: log sequence number, %r: resetlog ID unique*

*# Cấu hình dung lượng tối đa không gian lưu trữ cho Fast Recovery Area*

*SQL> ALTER SYSTEM SET DB\_RECOVERY\_FILE\_DEST\_SIZE = 5G SCOPE=SPFILE;*

*# Cấu hình đường dẫn tới Fast Recovery Area*

*SQL> ALTER SYSTEM SET DB\_RECOVERY\_FILE\_DEST = '/u02/oradata/shbfin/fra/' SCOPE=SPFILE;*

*# Cấu hình định danh CSDL chính trong môi trường Data Guard*

*SQL> ALTER SYSTEM SET DB\_UNIQUE\_NAME=’pri’ SCOPE=BOTH;*

*# Cấu hình thời gian sao lưu, giữ flashback log tối đa, tính bằng phút*

*SQL> ALTER SYSTEM SET DB\_FLASHBACK\_RETENTION\_TARGET = 60 SCOPE=SPFILE;*

*# Cấu hình tệp tin mật khẩu chỉ dùng được trong phạm vi máy chủ cài đặt Oracle Database*

*SQL> ALTER SYSTEM SET REMOTE\_LOGIN\_PASSWORDFILE=EXCLUSIVE SCOPE=SPFILE;*

*# Cấu hình đồng bộ thao tác các tệp tin dữ liệu cùng với Standby Database*

*SQL> ALTER SYSTEM SET STANDBY\_FILE\_MANAGEMENT=AUTO SCOPE=SPFILE;*

*# Kết xuất tệp tin tham số dạng văn bản thô từ tệp tin tham số dạng nhị phân*

*SQL> CREATE PFILE FROM SPFILE;*

Trong cấu hình phụ trợ này, có thêm cấu hình về Fast Recovery Area (FRA). FRA là một tính năng của Oracle Database giúp quản lý và tự động hóa việc lưu trữ các thành phần quan trọng của cơ sở dữ liệu liên quan đến phục hồi và sao lưu. FRA được sử dụng trong hệ thống Data Guard để nhận những bản sao lưu bao gồm SRLs, ORLs và Archive Log được tạo ra bởi CSDL chính khi sử dụng RMAN với phương thức DUPLICATE.

* + 1. Tạo hệ thống dự phòng dựa trên RMAN DUPLICATE

*Sao chép và gửi file mật khẩu, tham số từ CSDL chính sang CSDL phụ:*Mọi CSDL trong kiến trúc Data Guard đều cần sử dụng một tệp lưu trữ mật khẩu có chung một mật khẩu giống nhau cho người dùng quản trị SYS. Việc redo data gửi có thành công hay không, một phần dựa vào việc hai hệ thống có xác thực được lẫn nhau hay không. Việc sao chép sang cả tệp tin mật khẩu, tham số đều là việc đảm bảo tính nhất quán dữ liệu trong các tệp. Một khi tệp tham số chuyển sang, CSDL phụ có thể khởi động tạm thời với chế độ NOMOUNT để làm nền tảng cho việc sử dụng RMAN DUPLICATE tiếp theo.

Khi thực hiện sao chép sang, hai hệ thống máy chủ đều cần thực hiện sở hữu khóa công khai của bên còn lại để xác thực qua phương thức SSH. Sao chép được dùng bằng công cụ Secure Copy (SCP), sử dụng SSH để mã hóa thông tin truyền đi, khi tệp tin nhận đến sẽ được giải mã bằng khóa bí mật mà hệ thống sở hữu. Cụ thể, việc sao chép từ máy chủ chính sang máy chủ phụ được thực hiện bằng cú pháp sau:

*> scp [other options] [source username@IP]:/[full file name] [destination username@IP]:/[directory]  
# source username@IP: thông tin định danh máy chủ chính*

*# full file name: tệp tin cần gửi, hiển thị dạng đầy đủ đường dẫn*

*# destination username@IP: thông tin định danh máy chủ phụ*

*# directory: thư mục nhận tệp tin được gửi*

*# Nếu đăng nhập và sử dụng luôn tại máy chủ chính, không cần truyền đối số source username@IP mà có thể bắt đầu luôn từ full file name*

Trong hệ thống hiện tại, tệp tin mật khẩu có tên là *orapwshbfin* và tệp tin tham số dạng văn bản thô đã được kết xuất có tên là *initshbfin.ora*, đều nằm ở thư mục *$ORACLE\_HOME/dbs*. Dưới đây là câu lệnh để sao chép các tệp tin trên từ máy chủ chính – source sang máy chủ dự phòng – target:

*> scp initshbfin.ora* [*oracle@10.0.15.63:$ORACLE\_HOME/dbs*](mailto:oracle@10.0.15.63:$ORACLE_HOME/dbs)

*> scp orapwshbfin* [*oracle@10.0.15.63:$ORACLE\_HOME/dbs*](mailto:oracle@10.0.15.63:$ORACLE_HOME/dbs)

*Dựng CSDL phụ bằng việc sao lưu dữ liệu từ CSDL chính bằng RMAN DUPLICATE:* công cụ RMAN tạo CSDL phụ bằng cách nhân bản các tệp tin được được sử dụng bởi CSDL chính, trong khi đó, việc vận hành của CSDL chính vẫn diễn ra bình thường mà không ảnh hưởng. Bằng việc khôi phục từ các Archive Redo Log được nhân bản từ CSDL chính, CSDL phụ được đồng bộ hóa với CSDL chính trong khoảng thời gian nhất định, phụ thuộc vào việc Archive Redo Log lưu trữ dữ liệu thay đổi ở thời điểm nào.

Ngoài ra, khi thực hiện bằng cách nhân bản – khôi phục này, các tham số được cấu hình tại CSDL chính cũng sẽ được đồng bộ sang cho CSDL phụ. Tuy nhiên, trong trường hợp này, tập tin tham số đã được sao chép bằng SCP sang trước, rút gọn thời gian trong việc thủ công tạo lại một tệp tham số mới. CSDL được sao chép gọi là TARGET, CSDL nhận và khôi phục bản sao chép gọi là AUXILIARY. Có thể kể đến một số dữ liệu được RMAN sao chép như: các datafiles hệ thống, control files, undo datafiles và tệp tin tham số cấu hình Instance của CSDL (spfile).

*# Khởi động Instance của CSDL dự phòng bằng tệp tin cấu hình tham số được sao chép từ hệ thống CSDL chính*

*SQL> STARTUP NOMOUNT PFILE=”$ORACLE\_HOME/dbs/initshbfin.ora”*

*# Thiết lập kết nối giữa hai CSDL cho công cụ RMAN*

**>** *rman TARGET sys/123@pri AUXILIARY sys/123@sta*

*# Thực hiện viết mã dành cho công việc DUPLICATE*

*RMAN> DUPLICATE TARGET DATABASE*

*FOR STANDBY*

*FROM ACTIVE DATABASE*

*DORECOVER*

*SPFILE*

*SET DB\_UNIQUE\_NAME=’sta’ COMMENT ‘IS STANDBY’*

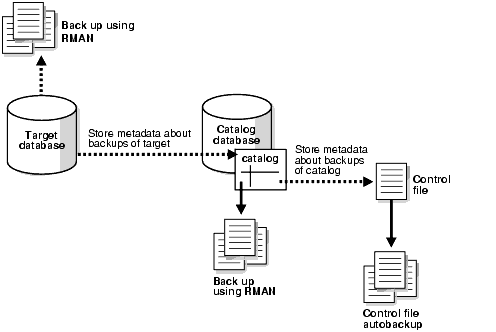
*SET LOCAL\_LISTENER=’(ADDRESS=(PROTOCOl=TCP)*

*(HOST=target)(PORT=1521))' SCOPE=SPFILE’*

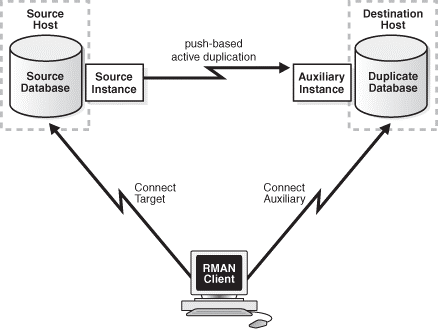
*NOFILENAMECHECK;*

Trong các lệnh của RMAN Duplicate được thể hiện ở trên, có rất nhiều cài đặt được đưa ra. Ý nghĩa của các cài đặt này như sau:

* *FOR STANDBY:* CSDL phụ/dự phòng, được tạo ra từ CSDL chính thông qua việc DUPLICATE không giống với các CSDL thông thường không ở trong môi trường Data Guard cũng được nhân bản từ CSDL ban đầu. Nếu không sử dụng thẻ này, CSDL dự phòng sẽ không nhận được mã số Database Identifier (DBID) mới. Mã số DBID này được một CSDL nhỏ tên là Recovery Catalog thực hiện lưu trữ tập trung siêu dữ liệu (meta-data) của các CSDL về các thông tin cho RMAN trong quá trình sao lưu khôi phục. Bởi DBID của CSDL chính và CSDL dự phòng cần phải giống nhau nên mới có thể nhận dạng và thực hiện giảm tải công việc sao lưu qua CSDL dự phòng khi CSDL chính hoạt động. Một số siêu dữ liệu mà Recovery Catalog chứa như sau: cấu trúc CSDL, thông tin về các data files, control files, archived redo logs.

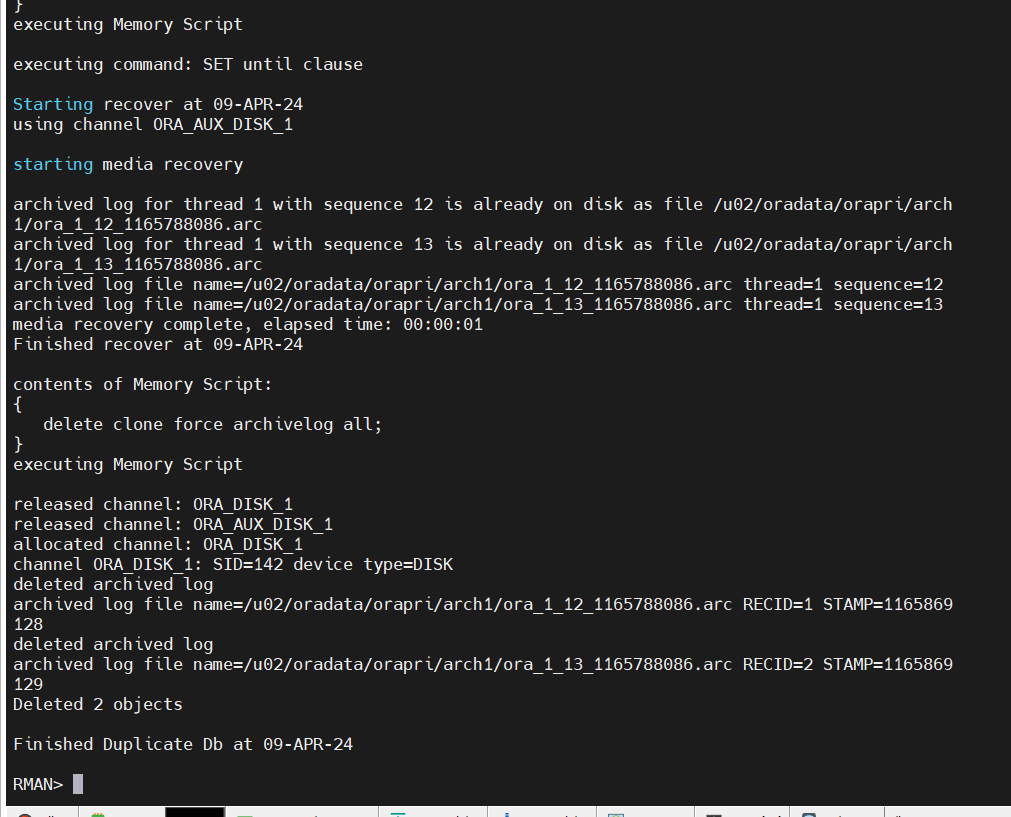


Hình 29 CSDL thực hiện sao lưu thông qua meta-data được RMAN quản lý



Hình 30 CSDL dự phòng được nhân bản thông qua RMAN DUPLICATE

* *FROM ACTIVE DATABASE:* RMAN thực hiện nhân bảo các data files trực tiếp từ CSDL chính tới CSDL phụ. Khi đó, bắt buộc CSDL chính phải khởi động từ mức MOUNTED trở lên, do tại mức độ này, control files được mở và cung cấp các thông tin về vị trí của các data files cũng như online redo logs. Nếu không sử dụng tùy chọn này, RMAN sẽ thực hiện nhân bản CSDL dựa trên bản sao lưu từ CSDL chính.
* *DORECOVER:* Khi tạo CSDL dự phòng bằng RMAN DUPLICATE, với tùy chọn này, RMAN sẽ dựng lại CSDL thông qua việc khôi phục các Archived Redo Logs, Online Redo Logs dựa trên SCN Checkpoint có trong control files được nhân bản qua. Nếu khong sử dụng tùy chọn này, RMAN sau khi nhân bản các tệp tin qua sẽ không thực hiện phục hồi và chỉ mở CSDL ở trạng thái MOUNTED.
* *SPFILE:* thực hiện nhân bản và chỉnh sửa thông tin tệp cấu hình tham số từ CSDL chính. Trong bài này, thông tin cần thay đổi là DB\_UNIQUE\_NAME và LOCAL\_LISTENER.
* *NOFILENAMECHECK:* khi đường dẫn chứa data files và online redo logs file tại CSDL chính giống với CSDL phụ, tùy chọn này bỏ qua việc kiểm tra tên tệp tin khu thực hiện sao lưu và không cần cung đường dẫn thư mục cụ thể.

**

Hình 31 Hoàn thành nhân bản CSDL phụ từ CSDL chính

* + 1. Cấu hình môi trường Data Guard

Sau khi CSDL dự phòng được nhân bản từ RMAN DUPLICATE, về cơ bản, các tham số chung của CSDL được cài đặt để có thể sử dụng bình thường. Tuy nhiên, Redo Transport Services tại CSDL chính chưa được thiết lập để có thể truyền redo data sang cho CSDL dự phòng, và Log Apply Services tại CSDL dự phòng chưa được thiết lập để có thể khôi phục thông qua việc áp dụng các redo data nhận được từ CSDL chính.

Ngoài ra, trong trường hợp CSDL chính chuyển giao vai trò vận hành sang CSDL phụ, cũng chưa có cơ chế nào để CSDL chính (cũ) thực hiện các công việc trước đây mà CSDL chính (phụ, mới) thực hiện. Vì vậy, việc cấu hình tham số trong môi trường Data Guard này rất quan trọng để đạt được thành công trong việc đảm bảo hoạt động switchover và failover không gặp lỗi. Hầu hết, các bước cấu hình môi trường Data Guard cho cả hai CSDL đều giống nhau, duy nhất chỉ có tại CSDL dự phòng, phải bật tiến trình MRPn hay Log Apply Services của CSDL dự phòng dạng vật lý để áp dụng các thay đổi, đồng bộ hóa cho CSDL.

*Cấu hình Redo Transport Services đối với CSDL chính:* cũng có thể nói đây là việc "*Thiết lập vị trí lưu Redo Log*" nhưng ở phạm vi toàn cục – gửi redo data sang CSDL dự phòng. Phần cấu hình này sẽ dùng tên bí danh – alias được cài đặt trong Local Naming Method thông qua tệp tin tnsname.ora, cho tham số SERVICE thay vì LOCATION với đường dẫn để lưu xuống như thông thường. Cùng với đó là kiểu truyền redo data. Đối với phần này, CSDL được thiết lập theo hướng *Ưu tiên hiệu năng*, nên cần kiểu truyền là ASYNC/NOAFFIRM. Các tham số như VALID\_FOR = (ALL\_LOGFILES, PRIMARY\_ROLE) và DB\_UNIQUE\_NAME=sta (bí danh của CSDL dự phòng) có ý nghĩa là: CSDL sẽ gửi redo data của ORLs cho CSDL dự phòng khi vai trò của CSDL là chính theo kiểu truyền bất đồng bộ ASYNC/NOAFFIRM.

*SQL> ALTER SYSTEM SET LOG\_ARCHIVE\_DEST\_2=*

*'SERVICE=sta ASYNC*

*VALID\_FOR=(ALL\_LOGFILES,PRIMARY\_ROLE)*

*DB\_UNIQUE\_NAME=sta' SCOPE=SPFILE;*

*Khai báo CSDL chính, dự phòng trong môi trường Data Guard:* thông qua tham số LOG\_ARCHIVE\_CONFIG, liệt kê CSDL chính và CSDL dự phòng bằng tham số con DG\_CONFIG. Theo mặc định, LOG\_ARCHIVE\_CONFIG cho phép CSDL chính gửi redo data cho CSDL dự phòng, tuy nhiên cũng có thể cài đặt không cho phép gửi từ CSDL chính hoặc không cho phép nhận từ CSDL dự phòng. Cấu trúc của câu lệnh như sau:

*LOG\_ARCHIVE\_CONFIG = {*

*[SEND | NOSEND] [RECEIVE | NORECEIVE]*

*[DG\_CONFIG = (remote\_db\_uniqu\_name1]*

*[, … remote\_db\_unique\_name9) | NODG\_CONFIG]}*

* SEND | NOSEND: xác định các redo data từ CSDL chính có được gửi đến CSDL dự phòng không, mặc định sẽ là SEND
* RECEIVE | NORECEIVE: xác định các CSDL dự phòng có nhận redo data từ CSLD chính không, mặc định sẽ là RECEIVE
* DG\_CONFIG: xác định danh sách bao gồm CSDL chính và các CSDL dự phòng được nhận redo data

Cấu hình dành cho hai CSDL thực hiện trong bài sẽ như sau:

*SQL> ALTER SYSTEM SET LOG\_ARCHIVE\_CONFIG='DG\_CONFIG=(pri,sta)';*

*# Kích hoạt các Redo Transport Services đã cấu hính*

*SQL> ALTER SYSTEM SET LOG\_ARCHIVE\_DEST\_STATE\_1=ENABLE;*

*SQL> ALTER SYSTEM SET LOG\_ARCHIVE\_DEST\_STATE\_2=ENABLE;*

*Cấu hình tiến tình xử lý trễ chủ động FAL cho vai trò CSDL dự phòng:* tiến trình FAL có hai thành phần cần cấu hình là FAL Client và FAL Server. Cả hai tiến trình con này đều được sử dụng cho CSDL dự phòng trong việc chủ động xử lý thiếu dữ liệu thay vì bị động với tiến trình ARCn của CSDL chính. Nếu tiến trình Log Apply MRP nhận thấy dấu hiệu thiếu dữ liệu, một yêu cầu bắt buộc CSDL chính gửi lại được sinh ra từ CSDL dự phòng. Khi này, FAL Client sẽ gửi yêu cầu tới FAL Server được đặt trên CSDL chính gửi lại Archived Log Files để xử lý trễ - được gọi là Archive Redo Gap Sequence. Điều đặc biệt là FAL chỉ hỗ trợ cho CSDL dự phòng dạng vật lý – Physcial Standby.

FAL\_CLIENT sẽ được cài đặt cho CSDL dự phòng mà CSDL chính đang trỏ tới. Khi tiến trình Log Apply MRP phát hiện độ trễ trong việc áp dụng thay đổi (thường là với điều kiện log sequence hiện tại > log sequence lần trước + 1), MRP sẽ gửi tín hiệu yêu cầu tiến trình ARCn thuộc CSDL chính được cấu hình bởi FAL\_SERVER và kèm theo đó là log sequence gây ra hiện tượng trễ dữ liệu. Ngoài ra, thông tin được gửi cũng kèm theo là giá trị trong FAL\_CLIENT. Sau đó, tiến trình ARCn sẽ gửi lại Archived Log Files có log sequence thấp hơn, thiếu so với log sequence mà MRP gửi sang.

*# Đối với CSDL chính, cài đặt cho việc khi bị chuyển sang thành vai trò dự phòng*

*SQL> ALTER SYSTEM SET FAL\_CLIENT='pri';*

*SQL> ALTER SYSTEM SET FAL\_SERVER='sta';*

*# Đối với CSDL dự phòng*

*SQL> ALTER SYSTEM SET FAL\_CLIENT='sta';*

*SQL> ALTER SYSTEM SET FAL\_SERVER='pri';*

*Cấu hình chế độ bảo vệ:* Chỉ cài đặt phương thức truyền của Redo Transport Services (như ASYNC/NOAFFIRM) là chưa đủ, phương thức truyền không thể đảm bảo được dữ liệu được bảo vệ theo cách nào. Ngoài ra, chế độ bảo vệ cũng cần cấu hình phương thức truyền của Redo Transport Service phù hợp. Ví dụ như đối với Maximum Protection đảm bảo rằng sẽ không có dữ liệu bị mất/lệch bằng cách dừng hoạt động của CSDL chính khi redo data không thể truyền/áp dụng thay đổi vào CSDL dự phòng. Bảng sau đây chỉ ra yêu cầu các thông tin cần để thiết lập chế độ bảo vệ:

|  |  |  |
| --- | --- | --- |
| **Maximum Availability** | **Maximum Performance** | **Maximum Protection** |
| AFFIRM/NOAFFIRM | NOAFFIRM | AFFIRM |
| SYNC | ASYNC | SYNC |
| DB\_UNIQUE\_NAME | DB\_UNIQUE\_NAME | DB\_UNIQUE\_NAME |

Bảng 8 Các thông tin cần để thiết lập chế độ bảo vệ

Thiết lập chế độ ưu tiên hiệu năng sau khi đã thiết lập Redo Transport Services theo phương thức bất đồng bộ ASYNC/NOAFFIRM:

*SQL> ALTER DATABASE SET STANDBY DATABASE TO MAXIMIZE PERFORMANCE;*

*Khởi động Redo Log Apply – tiến trình MRP trên CSDL dự phòng:* sau khi cấu hình thành công về môi trường Data Guard, bước cuối cùng là khởi động tiến trình MRP cho CSDL dự phòng dạng vật lý, để CSDL có thể bắt đầu áp dụng các redo data nhận được từ SRLs cũng như Archived Redo Logs.

Theo mặc định, MRP sẽ được tự động bật tính năng Real-Time Apply, hỗ trợ quá trình đồng bộ hóa diễn ra nhanh chóng, sát với CSDL chính thay vì xuất hiện độ trễ và áp dụng từ Archived Redo Logs. Tiến trình MRP sẽ áp dụng redo data từ SRLs sau khi tiến trình RFS hoàn thành việc ghi vào.

Sử dụng Real-Time Apply cùng với việc thiết kế SRLs có số lượng cũng như dung lượng lớn hơn ORLs luôn đảm bảo được việc CSDL dự phòng cập nhật “up-to-date” với CSDL chính vì Log Switch sẽ xảy ra chậm h ơn ít nhất là 1 log file so với ORLs. Nếu không muốn sử dụng Real-Time Apply, cần thêm tùy chọn DELAY cùng với khoảng thời gian giới hạn. Sử dụng thêm tùy chọn DISCONNECT sẽ đưa tiến trình này vào tiến trình chạy nền (background) thay vì theo trực tiếp phiên sử dụng của người dùng (foreground). Ngoài ra, có một số trường hợp redo data sẽ chưa được gửi tới CSDL dự phòng nếu chưa xảy ra Log Switch ở CSDL chính, lúc này cần thực hiện thủ công hoặc sử dụng đầy một ORLs.

*# Sử dụng Redo Apply với Real-Time Apply*

*SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT;*

*# Sử dụng Redo Apply với Delay Apply, thông qua Archived Redo Logs*

*SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE USING ARCHIVED LOGFILE DISCONNECT;*

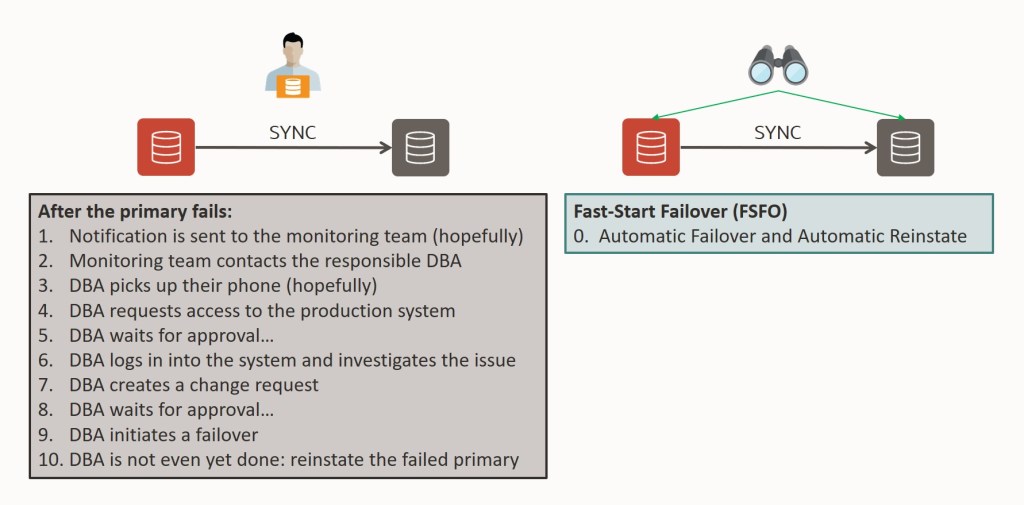
*# Dừng hoạt động tiến trình Redo Apply*

*SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE CANCEL;*

* + 1. Mô hình giám sát tính khả dụng hệ thống

Để đảm bảo hệ thống trong môi trường Data Guard vận hành ổn định, kết nối thông suốt với nhau, cần một chức năng giám sát trạng thái của các CSDL. Mô hình quản lý phân tán Oracle Data Guard Broker cung cấp các chức năng để có thể truy vấn thông tin về CSDL chính và các CSDL dự phòng. Broker tự động hóa được phần lớn các công việc của người quản trị trong môi trường Data Guard như switchover để phục vụ cho việc nâng cấp, bảo trì hệ thống, thay đổi các tham số cấu hình trong môi trường.

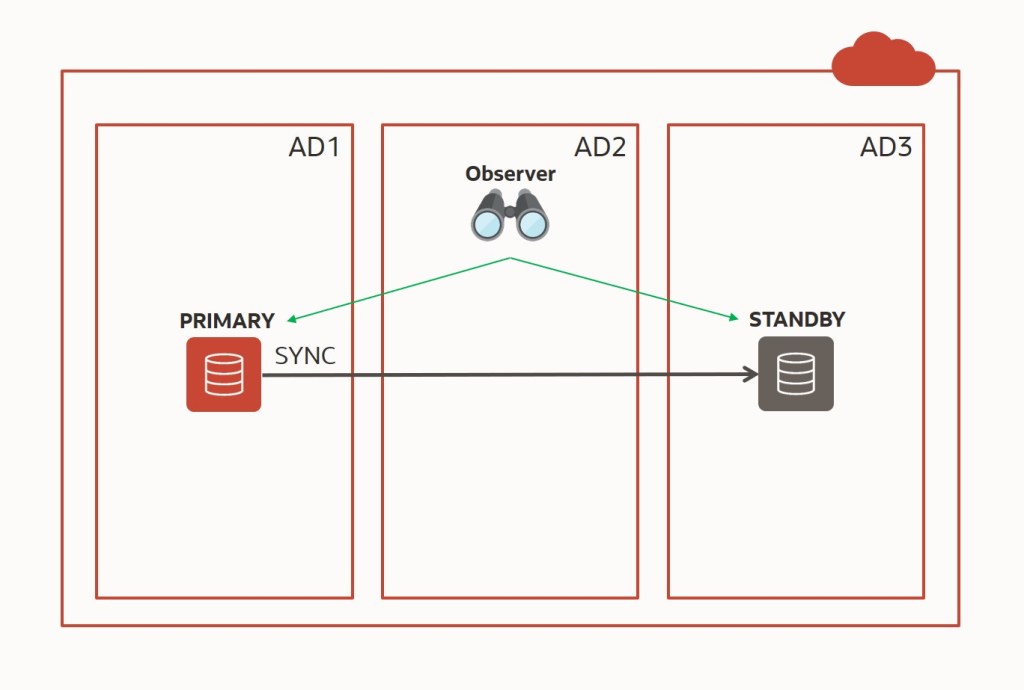
Ngoài việc cung cấp thông tin về trạng thái hệ thống, có một tính năng nữa mà mô hình Broker cung cấp là việc đảm bảo tính khả dụng của hệ thống CSDL luôn sẵn sàng – tính năng Fast-Start Failover. Tính năng Fast-Start Failover cho phép Broker thực hiện chuyển đổi vai trò tự động của các CSDL. Khi CSDL chính gặp sự cố đột ngột, CSDL dự phòng được chỉ định trước sẽ đảm nhận vai trò thay CSDL chính ngay lập tức (hoặc theo một lượng thời gian được chỉ định) thay vì đợi người quản trị (DBA) thực hiện thủ công, gây mất thời gian do thủ tục hạn chế.



Hình 32 So sánh cơ chế Failover thủ công và tự động bằng Fast-Start Failover

Tính năng Fast-Start Failover (FSFO) dựa vào một cơ chế giám sát khác (vẫn dựa trên mô hình Broker Framework) gọi là Observer, để thực hiện failover một cách tự động. Observer có thể coi là một thành phần thứ ba (third party) bên cạnh CSDL chính và CSDL dự phòng trong môi trường Data Guard, là một cơ chế được xây dựng dựa trên nền tảng Broker. Ngoài việc thực hiện nhiệm vụ failover tự động khi CSDL chính gặp sự cố, Observer còn tự động thực hiện việc khôi phục lại CSDL chính (cũ) sau khi gặp sự cố dựa trên công nghệ Flashback. Đây là một thành phần quan trọng, đảm bảo yếu tố dự phòng và chỉ số RTO, RPO.

Theo khuyến nghị của Oracle, Observer cần được cài trên một máy chủ thứ ba ngoài hai máy chủ cài đặt CSDL, dựa trên các thư viện và môi trường hỗ trợ từ Oracle Client. Khi này, bất kể các lỗi xảy ra tại hệ thống của CSDL chính hay dự phòng đều không ảnh hưởng đến máy chủ chạy Observer. Dựa vào mô hình phân tán như vậy, hạn chế được lỗi và đảm bảo được việc giám sát, hoạt động luôn chính xác.

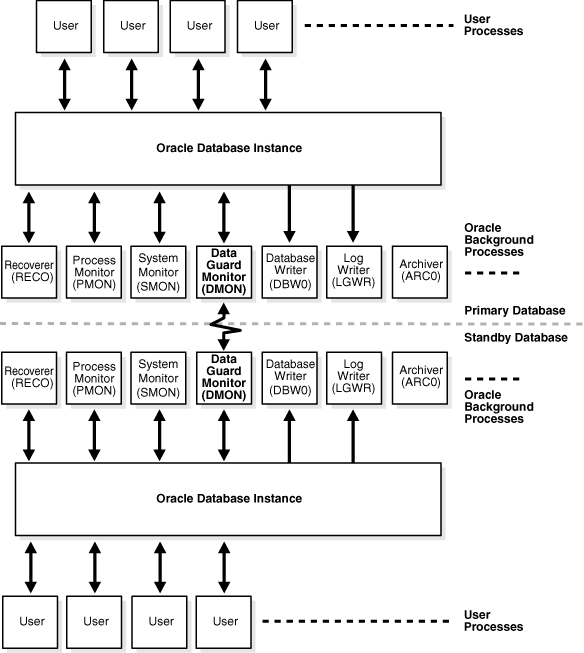


Hình 33 Minh họa máy chủ thứ ba chứa Observer trong môi trường Data Guard, dựa trên nền tảng điện toán đám mây OCI

Fast-Start Failover được kích hoạt khi Observer giám sát và nhận tín hiệu từ một số điều kiện sau:

* Máy chủ chứa Observer ổn định, Observer hoạt động
* Observer và CSDL dự phòng mất kết nối với CSDL chính. Nếu chỉ Observer mất kết nối với CSDL chính, thì Observer vẫn mặc định CSDL chính hoạt động thông qua CSDL dự phòng
* Observer vẫn có kết nối với CSDL dự phòng
* Thời gian chờ kết nối lại CSDL chính đã đạt giới hạn
* Các ràng buộc về hệ thống chứa CSDL như data files, các đối tượng, control files gặp lỗi, tiến trình LGWR không thể thực hiện ghi hoặc vùng nhớ chứa Archived Redo Logs bị đầy/không tồn tại
* Instance CSDL gặp lỗi bởi một hoặc nhiều các tiến trình quan trọng sau gặp lỗi: Process Monitor (PMON), System Monitor (SMON), Database Writer (DBWr), Checkpoint (CKPT), Log Writer (LGWr)
* Hoặc kích hoạt thủ công PL/SQL: DBMS\_DG.INITIATE\_FS\_FAILOVER

*Cấu hình cho mô hình Broker – Configuration:* Thực hiện đặt tham số DG\_BROKER\_START với giá trị TRUE - khởi động tiến trình nền Oracle Data Guard monitor (DMON), phải được chạy tại mọi CSDL được quản lý bởi Broker. DMON là tiến trình nền nằm ở phía máy chủ (server-side), là thành phần tương tác trực tiếp với CSDL và các tiến trình DMON của CSDL khác để thực hiện giám sát và nhận thông tin. Vì vậy, dù người quản trị thực hiện việc lấy thông tin từ Broker ở bất kỳ CSDL nào, đều nhận được thông tin của tất cả các CSDL chính và dự phòng, nằm trong phạm vi được Broker giám sát.



Hình 34 Minh họa tương tác của tiến trình DMON trong môi trường Broker Configuration

Ngoài ra, tiến trình DMON cũng tự động tạo, sao chép các tệp chứa thông tin cấu hình của các CSDL mà nó quản lý ở mỗi máy chủ. Tệp tin này chứa thông tin về trạng thái, thuộc tính của CSDL, vai trò của CSDL để thực hiện báo cáo tới cho người quản trị thông qua công cụ giao diện dòng lệnh DGMGRL hoặc giao diện đồ họa OEM Cloud Control.

Các thuộc tính liên quan tới môi trường Data Guard mà Broker Configuration quản lý có mối liên hệ chặt chẽ tới các tham số chung của CSDL. Chính vì điều này, mà các tham số chung đang được quản lý bởi tệp tin Server Parameter File (spfile), là nền tảng để Instance của CSDL khởi động và MOUNT với Database cũng như được quản lý bởi Broker Configuration, có nguy cơ xảy ra xung đột khi thực hiện chỉnh sửa thủ công từ một phía.

Để đảm bảo rằng Broker có thể cập nhật được giá trị của tham số của các tệp tin, người quản trị chỉ được thực hiện cấu hình trực tiếp thông qua Instance (tức là trên spfile, không phải pfile – dạng văn bản có thể đọc). Thông qua tệp tin cấu hình trên, Broker sẽ có cơ chế để xử lý xung đột thuộc tính. Ngoài ra, việc thực hiện chỉnh sửa trên Broker Configuration cũng tương tự, Broker cũng sẽ tự cập nhật ngược lại cho tệp tin cấu hình tham số spfile của Instance.

*# Thực hiện bật tiến trình DMON ở cả hai loại CSDL*

*SQL> ALTER SYSTEM SET DG\_BROKER\_START=TRUE SCOPE=BOTH;*

*Thực hiện xóa thông tin Redo Transport Service qua LOG\_ARCHIVE\_DEST\_n (n >= 2):* Với Redo Transport Service, Broker có cơ chế tự nhận biết CSDL chính và CSDL dự phòng, do đó, việc truyền tải sẽ được thực hiện tự động. Việc xóa thông tin cho tham số này đảm bảo cho việc Broker và các tiến trình được cài đặt không bị xung đột với nhau.

*# Thực hiện trên cả hai loại CSDL*

*ALTER SYSTEM SET LOG\_ARCHIVE\_DEST\_2='';*

*Thực hiện tạo Broker Configuration cho các CSDL:*

*# Thực hiện đăng nhập vào DGMGRL của CSDL (chính hoặc dự phòng đểu có thể)*

*> DGMGRL sys/123 as sysdba*

*# Thực hiện tạo Broker Configuration và thêm CSDL chính*

*DGMGRL> CREATE CONFIGURATION 'DRSHBfinSolution' AS PRIMARY DATABASE IS 'pri' CONNECT IDENTIFIER IS pri;*

*# Thực hiện thêm CSDL dự phòng vào Broker Configuration*

*DGMGRL> ADD DATABASE 'sta' AS CONNECT IDENTIFIER IS 'sta' MAINTAINED AS PHYSICAL;*

* ‘DRSHBfinSolution’ – tên cho Broker Configuration
* ‘pri’/’sta’ – tên thuộc DB\_UNIQUE\_NAME của CSDL chính và CSDL dự phòng
* pri/sta – là CONNECT-IDENTIFIER, là giá trị alias đặt trong tnsname.ora, thuộc phương thức phân giải chuỗi kết nối Local Naming Method. Broker sử dụng giá trị này để tương tác với các CSDL khác được cài đặt trong Broker Configuration
* MAINTAINED AS PHYSICAL – cung cấp thông tin cho Broker Configuration loại CSDL dự phòng đang dùng

Trong Broker Configuration, có rất nhiều tham số dùng để cấu hình, thường là liên quan đến việc hiển thị trong giám sát, đặt những tiêu chuẩn, giới hạn để cảnh báo sớm và liên quan đến các chuỗi dùng để kết nối. Trong phạm vi thực nghiệm, cấu hình Broker Configuration sẽ không đi quá sâu, mỗi vấn đề lại có các cách cấu hình khác nhau cũng như phân bổ thời gian hợp lý cho các thành phần khác.

*# Cấu hình lại chuỗi kết nối tĩnh cho Configuration để sau khi Switchover, Broker có thể kết nối lại, cho dù bất kể tình trạng của CSDL như thế nào (liên quan đến khái niệm về Oracle Net về SID và SERVICE\_NAME)*

*DGMGRL > EDIT DATABASE PRI SET PROPERTY STATICCONNECTIDENTIFIER = '(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(PORT=1521)(HOST=db1))(CONNECT\_DATA=(SERVICE\_NAME=shbfin)(INSTANCE\_NAME=shbfin)(SERVER=DEDICATED)))';*

*DGMGRL> EDIT DATABASE STA SET PROPERTY STATICCONNECTIDENTIFIER = '(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(PORT=1521)(HOST=db2))(CONNECT\_DATA=(SERVICE\_NAME=shbfin)(INSTANCE\_NAME=shbfin)(SERVER=DEDICATED)))';*

*# Cấu hình cảnh báo khi các điều kiện vượt quá giới hạn. Ở đây đặt các giá trị về 0 tuyệt đối (do trong cùng một subnet, độ trễ hiếm khi xảy ra)*

*# ApplyLagThreshold: Cảnh báo về CSDL dự phòng khi Redo Log Apply trễ vượt quá giá trị được chỉ định*

*# TransportLagThreshold: Cảnh báo về CSDL dự phòng khi việc truyền redo data bị trễ quá so với giá trị được chỉ định*

*DGMGRL> EDIT DATABASE PRI SET PROPERTY ApplyLagThreshold=0;*

*DGMGRL> EDIT DATABASE PRI SET PROPERTY TransportLagThreshold=0;*

*DGMGRL> EDIT DATABASE STA SET PROPERTY ApplyLagThreshold=0;*

*DGMGRL> EDIT DATABASE STA SET PROPERTY TransportLagThreshold=0*

*Cấu hình Fast-Start Failover:* Để thực hiện sử dụng tính năng Fast-Start Failover, CSDL phải được đặt ở chế độ *Ưu tiên hiệu năng* hoặc *Ưu tiên tính sẵn sàng*. Tham số LogXptMode – cấu hình phương thức truyền Redo Transport Services được đặt SYNC/ASYNC hoặc FASTSYNC tùy thuộc vào chế độ bảo vệ hoặc vai trò của CSDL. Tính năng Flashback phải được bật tại CSDL chính để thực hiện việc khôi phục nhanh (re-instate) tự động bởi Observer sau khi Failover bởi sự cố, cũng như việc cấu hình trước CSDL dự phòng nào sẽ đảm nhận vai trò chính.

*# Thiết lập sang chế độ bảo vệ ưu tiên tính sẵn sàng (tính khả dụng) của CSDL*

*DGMGRL> EDIT DATABASE PRI SET PROPERTY 'LogXptMode'='SYNC';*

*DGMGRL> EDIT DATABASE STA SET PROPERTY 'LogXptMode'='SYNC';*

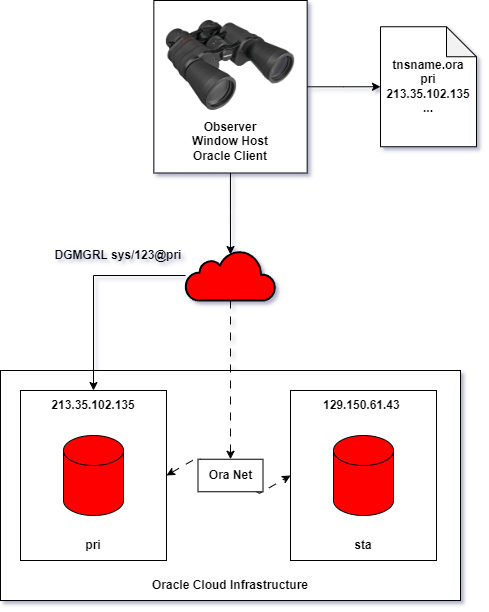
*DGMGRL> EDIT CONFIGURATION SET PROTECTION MODE AS MAXAVAILABILITY;*

Tính năng Fast-Start Failover có rất nhiều tham số, sau đây là một vài tham số mang tính chiến lược, quyết định tính khả dụng của CSDL và vận hành của doanh nghiệp:

* FastStartFailoverThreshold: thời gian giới hạn mà Observer thực hiện để kết nối lại CSDL chính trước khi thực hiện Fast-Start Failover (FSFO). Thời gian này bắt đầu khi Observer bị mất kết nối với CSDL chính. Nếu Observer không thể kết nối lại CSDL chính trong khoảng thời gian kể trên, Observer sẽ kích hoạt FSFO chuyển vai trò sang cho CSDL dự phòng. Mặc định là 30 giây cho tham số này, tuy nhiên, để có cái nhìn khách quan và cấu hình đúng nhất, thì người quản trị nên tham khảo bảng v$FS\_OBSERVER\_HISTOGRAM để xem dữ liệu thống kê mỗi lần Observer kết nối lại mất bao nhiêu thời gian. Oracle có một số khuyến nghị về lựa chọn thời gian như sau:
  + Với Single-Instance, mạng có độ trễ thấp và tin cậy: 10 – 15 giây
  + Với Single-Instance, mạng diện rộng có độ trễ cao: 30 – 45 giây
  + Với Multi-Instance (RAC): lớn hơn 24 – 40 giây
* FastStartFailoverLagLimit: thời gian giới hạn cho phép có độ trễ trong việc CSDL dự phòng áp dụng redo data so với CSDL chính. Nếu độ trễ có thời gian lớn hơn thời gian đã chỉ định, Fast-Start Failover sẽ không được sử dụng. Tham số này được cài đặt khi CSDL ở chế độ ưu tiên hiệu năng.
* FastStartFailoverAutoReinstate: thực hiện nhiệm vụ khôi phục lại trạng thái của CSDL chính cũ sau khi Fast-Start Failover xảy ra do gặp sự cố. Ngoài ra, Broker cũng không bao giờ tự động khôi phục lại trạng thái của CSDL chính nếu Fast-Start Failover được thực hiện thủ công hoặc được kích hoạt bằng thủ tục DBMS\_DG.INIIATE\_FS\_FAILOVER
* FastStartFailoverPmyShutdown: thực hiện dừng hoạt động CSDL chính sau khi Fast-Start Failover xảy ra, thực hiện dừng các hoạt động để chuyển sang cho CSDL dự phòng, không cho phép người dùng thông thường thực hiện truy vấn tại CSDL chính cũ
* CommunicationTimeount: giới hạn thời gian cho phép Broker chờ đợi trước khi ra quyết định cảnh báo mất kết nối giữa CSDL chính và CSDL dự phòng. Với giá trị bằng 0 cho biết các CSDL không bao giờ mất kết nối, mặc định là 180 giây
* ObserverReconnect: quy định chu kỳ mà Observer thiết lập kết nối mới tới CSDL chính. Với giá trị bằng 0, Observer duy trì kết nối với CSDL chính nhưng không định kỳ thiết lập kết nối mới. Việc thiết lập cũng có lợi trong việc phát hiện kịp thời khi không thể kết nối tới CSDL chính, tuy nhiên gây tốn kém về mặt hiệu suất và chi phí

*Cấu hình máy chủ chạy Observer cho Fast-Start Failover:* như theo khuyến nghị của Oracle, Observer nên được chạy trên một máy khác, và thường là máy khách với phần mềm Oracle Client (có môi trường giống với Oracle Database Software, nhưng giảm tải các thành phần không cần thiết đối với Client). Tại Oracle Client, sẽ thực hiện khởi động Observer trong giao diện dòng lệnh DGMGRL được kết nối tới bất kỳ CSDL nào, nhưng tiến trình, tệp lưu logs và cấu hình của Observer sẽ được chạy/lưu trên máy mà Observer sử dụng. Observer sẽ dựa vào thông tin mà Broker Configuration cung cấp để giám sát các CSDL. Vì Observer là một tiến trình trực tiếp (foreground), nên có thể đưa cả câu lệnh kích hoạt về tiến trình nền hoặc thiết lập một phiên kết nối mới tới công cụ DGMGRL.

Trong bài này, Oracle Client được cài đặt trên máy tính Windows, phục vụ khởi động Observer cùng với các tệp tin logs, dữ liệu được lưu trữ tại Windows để giám sát, chuẩn bị cho Fast-Start Failover của hai máy ảo trên nền tảng điện toán đám mây OCI chứa CSDL chính và CSDL phụ, đáp ứng được tính phân tán về mặt vật lý giữa các máy chủ. Tại Oracle Client cũng cần thiết lập Local Naming Method thông qua cấu hình tnsnames.ora để Oracle có thể biên dịch chuỗi mô tả kết nối.



Hình 35 Minh họa việc thiết lập Observer tại Windows

*# Thực hiện kích hoạt Fast-Start Failover trong Broker Configuration trước khi bật Observer trên Window Host*

*DGMGRL> ENABLE FAST\_START FAILOVER;*

*# Tại Windows Host, thực hiện việc kích hoạt Observer và cấu hình đường dẫn lưu các tệp tin của Observer thích hợp. Trước khi kích hoạt, đăng nhập vào Configuration tại bất kỳ CSDL nào được quản lý bởi Broker. Nếu không thực hiện cấu hình thư mục, Observer sẽ tự động tạo ở thư mục khác*

*> DGMGRL sys/123@pri*

*DGMGRL> START OBSERVER FILE IS D:\Workspace\2023-2024-Ki-I\Do-An-Tot-Nghiep\Bai-Lam\observer\obs.dat LOGFILE IS D:\Workspace\2023-2024-Ki-I\Do-An-Tot-Nghiep\Bai-Lam\observer\log\_obs.log;*

Ngoài ra, tệp ghi trữ log của Observer cũng rất quan trọng để người dùng quản trị có thể theo dõi hành động của Observer thực hiện đổi với các CSDL. Tuy nhiên, tệp log này cần mở thủ công mỗi khi người quản trị muốn thực hiện xác định lỗi. Để thuận tiện trong việc theo dõi, trong bài này, tệp log của Observer trên Windows Host được đọc tự động thông qua lập trình Bash scripts, sử dụng môi trường của Git Bash trong thời gian thực mỗi khi dòng dữ liệu mới về thông báo được thêm vào.

*# Sử dụng Bash để chạy lệnh dưới*

*#!/bin/bash*

*# Đường dẫn đến tệp log cần đọc*

*logfile="D:\Workspace\2023-2024-Ki-I\Do-An-Tot-Nghiep\Bai-Lam\observer\log\_obs.log"*

*# Thực hiện lấy ra số lượng ký tự hiện tại của tệp*

*lastsize=$(wc -c < "$logfile")*

*# Thực hiện vòng lặp với điều kiện nếu có dữ liệu mới thì in ra và cập nhật số lượng ký tự mới*

*while true; do*

*currentsize=$(wc -c < "$logfile")*

*if ((currentsize > lastsize)); then*

*# Lấy ra dòng mới từ vị trí cuối cùng đã đọc trước đó*

*newlines=$(tail -c +"$((lastsize + 1))" "$logfile")*

*# In ra thông báo*

*echo "$newlines"*

*lastsize=$currentsize*

*fi*

*# Thực hiện dùng 1 giây trước khi lặp tiếp*

*sleep 1*

*done*

* 1. Kiểm thử hoạt động hệ thống

|  |  |  |
| --- | --- | --- |
| **Số thứ tự** | **Danh mục** | **Giá trị** |
| **1** | Tên testcase | Đồng bộ dữ liệu |
| Yêu cầu người dùng | Dữ liệu thêm mới tại CSDL chính được cập nhật và đồng bộ hóa tại CSDL dự phòng sau khi được COMMIT |
| Dữ liệu kiểm thử | * Bảng test1(c1 INT PRIMARY KEY, c2 CHAR(6)) * Giá trị thêm: (52, ‘rap’), (56, ‘crap’) |
| Kết quả mong muốn | Chạy lệnh SELECT \* FROM test1 tại CSDL dự phòng sẽ trả về hai dòng giá trị: (52, ‘rap’), (56, ‘crap’) |
| Kết quả hệ thống | “2 rows returned” ((52, ‘rap’), (56, ‘crap’)) |
| Đánh giá | Đạt |
|  | | |
| **2** | Tên testcase | Log Switch CSDL chính |
| Yêu cầu người dùng | Khi hệ thống thực hiện Log Switch với điều kiện một Redo Log Groups đầy, Redo Logs Groups mới được sử dụng, Log Sequence tăng lên cho Groups tại CSDL chính, đảm bảo Standby Redo Logs bên CSDL dự phòng cũng phải tăng |
| Dữ liệu kiểm thử | Cho 03 Redo Log Groups với giá trị Log Sequence và tình trạng như sau:   * 13, CURRENT * 13, INACTIVE * 12, ACTIVE   Cho 04 Standby Redo Logs với giá trị Log Sequence và tình trạng như sau:   * 13, CURRENT * 0, UNASSIGNED * 0, UNASSIGNED * 0, UNASSIGNED |
| Kết quả mong muốn | Redo Log Groups:   * 13, ACTIVE * 14, CURRENT * 12, INACTIVE   Standby Redo Logs:   * 14, CURRENT * 0, UNASSIGNED * 0, UNASSIGNED * 0, UNASSIGNED |
| Kết quả hệ thống | Tại Redo Log Groups:   * 13, ACTIVE * 14, CURRENT * 12, INACTIVE (sai theo yêu cầu) |
| Đánh giá | Chưa đạt. Nguyên nhân do sau một khoảng thời gian không sử dụng hoặc log sequence số 12 đã hoàn thành việc lưu trữ (Archived) nên trạng thái đã thay đổi. Tuy nhiên các phần còn lại đúng như yêu cầu. |
|  | | |
| **3** | Tên testcase | Xử lý trễ dữ liệu sau khi tiến trình MRP được bật |
| Yêu cầu người dùng | Tiến trình áp dụng thay đổi CSDL dự phòng không được bật (Redo Log Apply). Sau khi tiến trình MRP được bật, các redo data áp dụng sang |
| Dữ liệu kiểm thử | * Bảng test1(c1 INT PRIMARY KEY, c2 CHAR(6)) * Dữ liệu thêm: (61, 'help'), (66, 'trapped') |
| Kết quả mong muốn | Kiểm tra không còn trễ với câu lệnh:  SELECT \* FROM GV$ARCHIVE\_DEST\_STATUS; |
| Kết quả hệ thống | NO GAP |
| Đánh giá | Đạt |
|  | | |
| **4** | Tên testcase | Xử lý trễ sau sự cố mất kết nối mạng |
| Yêu cầu người dùng | Các cơ chế như Automatic Gap Resolution hoặc Fetch Archived Log xử lý trễ do một số sự cố khách quan liên quan tới đường truyền mạng khiến redo data không thể gửi. Khi có kết nối mạng, yêu cầu xử lý trễ hoàn tất. Thực hiện ngắt kết nối và kết nối lại đường truyền mạng thông qua lệnh “ipconfig down” và “ipconfig up” |
| Dữ liệu kiểm thử | * Bảng test1(c1 INT PRIMARY KEY, c2 CHAR(6)) * Dữ liệu thêm: (12, 'le'), (14, 'vu') |
| Kết quả mong muốn | Kiểm tra không còn trễ với câu lệnh:  SELECT \* FROM GV$ARCHIVE\_DEST\_STATUS;  Không còn trạng thái “*TNS: Receive timeout occurred*” và “*RESOLVABLE GAP*” |
| Kết quả hệ thống | NO GAP |
| Đánh giá | Đạt |
|  | | |
| **5** | Tên testcase | Thực hiện Switchover cho kế hoạch |
| Yêu cầu người dùng | Thực hiện switchover chuyển đổi vai trò thành công cho CSDL chính và CSDL dự phòng |
| Dữ liệu kiểm thử | Sử dụng công cụ DGMGRL để theo dõi trạng thái   * CSDL chính (pri) với trạng thái: Primary * CSDL phụ (sta) với trạng thái: Standby   Truy vấn trên CSDL chính và phụ: *SELECT NAME, OPEN\_MODE, SWITCHOVER\_STATUS, DATABASE\_ROLE FROM V$DATABASE;*  Trả kết quả:   * pri | READ WRITE | TO STANDBY | PRIMARY * sta | READ ONLY | NOT ALLOWED | PHYSICAL STANDBY |
| Kết quả mong muốn | Sử dụng công cụ DGMGRL để theo dõi trạng thái   * CSDL chính (pri) với trạng thái: Standby * CSDL phụ (sta) với trạng thái: Primary   Truy vấn trên CSDL chính và phụ: *SELECT NAME, OPEN\_MODE, SWITCHOVER\_STATUS, DATABASE\_ROLE FROM V$DATABASE;*  Trả kết quả:   * sta | READ WRITE | TO STANDBY | PRIMARY * pri | READ ONLY | NOT ALLOWED | PHYSICAL STANDBY |
| Kết quả hệ thống | Tương tự kết quả mong muốn |
| Đánh giá | Đạt |
|  | | |
| **6** | Tên testcase | Thực hiện truy vấn trên CSDL dự phòng |
| Yêu cầu người dùng | Nhằm mục đích giảm tải CSDL chính trong tương lai, các truy vấn và tạo báo cáo cần được thực hiện bởi CSDL dự phòng với công nghệ Active Data Guard |
| Dữ liệu kiểm thử | Bảng test1 với 2 dòng dữ liệu |
| Kết quả mong muốn | Trả về 2 dòng dữ liệu của bảng test1 |
| Kết quả hệ thống | Trả về 2 dòng dữ liệu của bảng test1 |
| Đánh giá | Đạt |
|  | | |
| **7** | Tên testcase | Thực hiện Failover với Fast-Start Failover khi CSDL chính gặp sự cố |
| Yêu cầu người dùng | Observer được đặt ở máy riêng khác biệt với hai máy chủ chứa CSDL. Mô phỏng lại sự cố hỏng CSDL chính thông qua việc ngắt hoạt động của tiến trình PMON (quan trọng) tại Instance của CSDL chính. Sau khi đạt một khoảng thời gian nhất định tại CSDL chính, Failover xảy ra tự động và chuyển quyền cho CSDL dự phòng |
| Dữ liệu kiểm thử | * CSDL chính: Primary, CSDL dự phòng: Standby * Thực hiện ngắt hoạt động CSDL chính: kill -9 <UID>, với UID của tiến trình pmon trong hệ thống Oracle Linux * CSDL dự phòng được chỉ định Failover: sta |
| Kết quả mong muốn | * CSDL chính: sta * CSDL chính (cũ) được thực hiện khôi phục tự động (reinstate) * Thực hiện các tác vụ bình thường của CSDL vận hành trên CSDL chính mới |
| Kết quả hệ thống | Đạt các yêu cầu đề ra |
| Đánh giá | Đạt |
|  | | |
| **8** | Tên testcase | Thực hiện Switchover về CSDL chính cũ, trả lại vai trò sau Failover |
| Yêu cầu người dùng | Switchover hoạt động bình thường sau Failover, trả lại đúng vai trò của các CSDL |
| Dữ liệu kiểm thử | Hai CSDL sau Failover |
| Kết quả mong muốn | Trả về trạng thái cũ bình thường |
| Kết quả hệ thống | Trả về trạng thái cũ bình thường |
| Đánh giá | Đạt |

Bảng 9 Các Testcase kiểm thử

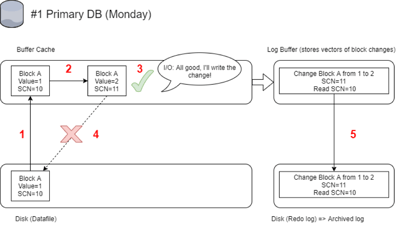
* 1. Phân tích sự cố mất ghi dữ liệu trong môi trường Data Guard
     1. Khái niệm

Doanh nghiệp xây dựng môi trường Data Guard, bao gồm 01 CSDL chính và 01 CSDL dự phòng (dạng vật lý). Lỗi mất ghi xảy ra tại CSDL chính và người quản trị không hề biết việc này bởi CSDL chính vẫn hoạt động bình thường. Đồng thời, tại CSDL dự phòng sẽ dừng hoạt động việc áp dụng thay đổi, thông báo mã lỗi ORA-600, đây là mã lỗi được đánh giá có tính nghiêm trọng cao, độ bao quát rộng, không đi sâu vào một lỗi cụ thể nào. Ví dụ như: lỗi logic, lỗi phân mảnh, lỗi tài nguyên, lỗi mạng, lỗi phần cứng. Vì vậy, khi gặp lỗi này, CSDL cần được xử lý bởi đội ngũ chuyên gia của Oracle thông qua Oracle Support. Để tránh mất nhiều thời gian cũng như cơ hội, các CSDL cần được bảo vệ và thông báo cụ thể về lỗi mất ghi thông qua tham số DB\_LOST \_WRITE\_PROTECT.

Mất ghi dữ liệu – Lost Writes, hay một block dữ liệu vật lý dưới thiết bị lưu trữ bị mất ghi khi các tiến trình thực hiện đọc ghi (I/O) trả về tín hiệu dữ liệu đã được ghi vào block dữ liệu, nhưng thực tế, việc ghi không thực sự xảy ra, dữ liệu tại thiết bị lưu trữ không thay đổi. Khi dữ liệu được tải lên khu vực Data Buffer Cache của Instance, thay đổi và được COMMIT, cần phải được giải phóng (flushing dirty blocks) – ghi xuống đĩa để dành lại không gian cho các hoạt động đọc ghi khác. Instance nhận được tín hiệu đã được ghi, tuy nhiên, có thể do tiến trình DBWr, bugs trong Oracle, lỗi ổ đĩa hoặc nhiều lý do khác mà dữ liệu cần được ghi lại không thay đổi, và CSDL vẫn tiếp tục hoạt động như chưa hề có vấn đề gì xảy ra.

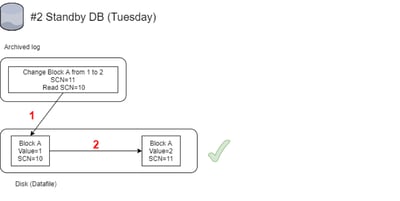
Lỗi mất ghi có thể xảy ra bất cứ lúc nào bởi những sự cố không thể lường trước. Trong môi trường Data Guard, lỗi mất ghi được dự báo sớm thông qua việc sử dụng tham số DB\_LOST\_WRITE\_PROTECT. Khi này, các redo data được gửi từ CSDL chính sang CSDL dự phòng sẽ thực hiện so sánh thông qua các thông tin bao gồm: giá trị đọc, giá trị ghi, chỉ số SCN. Khi đặt tham số DB\_LOST\_WRITE\_PROTECT với giá trị là TYPICAL (DB\_LOST\_WRITE\_PROTECT), CSDL dự phòng sẽ thực hiện so sánh trước khi áp dụng thay đổi. Giá trị này cũng yêu cầu redo data chứa thêm thông tin về chỉ số SCN của block khi được đọc từ đĩa. Khi lỗi mất ghi xảy ra, CSDL dự phòng sẽ thông báo lỗi với mã ORA-00756 (lỗi lost writes), dễ dàng cho người quản trị biết hướng để kịp thời sửa chữa hơn.

* + 1. Phát hiện vấn đề mất ghi với Lost Writes Protection



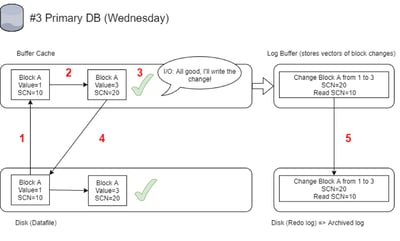
Hình 36 CSDL chính bị mất ghi (nguồn: dbvisit)

Diễn giải: (1) Block được đưa lên Data Buffer Cache theo yêu cầu của User thông qua Server Process; (2) câu truy vấn thực hiện cập nhật giá trị mới cho block, block trở thành dirty-block; (3) thực hiện Checkpoint, chỉ số SCN tăng từ 10 lên 11, tiến trình DBWr thực hiện ghi xuống và thông báo thành công; (4) bị lỗi mất ghi, block cũ không được cập nhật và nhất quán với giá trị cũ; (5) đồng thời mọi thay đổi của thông tin mới được lưu vào Redo Buffer Cache, bao gồm các hành động và chỉ số SCN, từ đây redo data được lưu xuống ORLs và được truyền sang CSDL dự phòng thông qua Redo Transport Services.



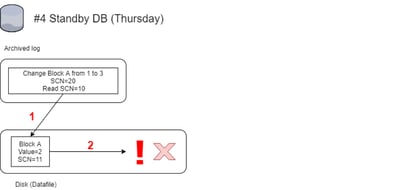
Hình 37 Áp dụng thay đổi tại CSDL dự phòng (nguồn: dbvisit)

Diễn giải: (1) CSDL dự phòng nhận Block với thông tin mới được tiến trình MRP nhận từ SRLs hoặc ARLs; (2) Tiến trình MRP áp dụng thay đổi thành công dưới đĩa, CSDL dự phòng được cập nhật với thông tin mới. Mặc dù CSDL dữ liệu bị lỗi mất ghi dưới đĩa, tuy nhiên, do vẫn nhận tín hiệu thành công, CSDL chính sẽ “tưởng nhầm” đã ghi thành công thông tin mới và gửi thông tin này cho CSDL dự phòng đồng bộ.



Hình 38 Tiếp tục thay đổi thông tin với block cũ (nguồn: dbvisit)

Diễn giải: (1) block cũ tiếp tục được tải lên Data Buffer Cache để thay đổi dữ liệu theo yêu cầu của người dùng thông qua Server Process; (2) block được thay đổi từ giá trị 1 sang giá trị 3; (3) thực hiện Checkpoint, chỉ số SCN sẽ tăng từ 10 lên 20, hệ thống nhận tín hiệu thành công khi được ghi; (4) đồng thời, tiến trình DBWr sẽ thực hiện ghi dirty-block xuống đĩa (flushing dirty-block) thành công; (5) Redo Buffer Cache nhận thông tin thay đổi cùng chỉ số SCN mới, lưu vào ORLs và gửi sang CSDL dự phòng



Hình 39 Áp dụng sau khi xảy ra quá trình mất ghi ở CSDL dự phòng

Diễn giải: (1) Block với SCN 11 và giá trị 2 sẽ được cơ chế Lost Write Protection thực hiện so sánh giữa các block trước kia đã áp dụng và block mới nhận trước khi chuyển cho tiến trình MRP áp dụng thay đổi vào CSDL dự phòng; (2) cơ chế phát hiện SCN trước khi của block lệch với block được ghi ở đĩa (10 và 11). Ngoài ra, giá trị gốc cũng có sự khác biệt khi giá trị cũ là 1 và giá trị trên đĩa là 2. Khi này hệ thống sẽ đưa ra lỗi cụ thể cho CSDL trong tệp tin cảnh báo (alert log file) với mã lỗi ORA-00752 thay vì lỗi mang tính bao quát, không cụ thể như ORA-600.

Khi gặp lỗi ORA-00752, để giải quyết vấn đề này, thông thường (đối với CSDL nhỏ, không quan trọng) sẽ thực hiện chuyển đổi vai trò failover qua CSDL dự phòng với mục đích khiến CSDL dự phòng trở thành CSDL chính và thực hiện dựng lại CSDL chính (cũ) thông qua RMAN DUPLICATE, công nghệ Flashback không được dùng để thực hiện khi không phải Fast-Start Failover xảy ra.

* + 1. Thực nghiệm cơ chế thông báo Lost Writes

Thực hiện việc tạo data files, bảng và chuẩn bị dữ liệu cho mục đích thực nghiệm cơ chế thông báo lỗi thông qua tham số DB\_LOST\_WRITE\_PROTECTION.

*# Tạo data files cho một tablespace với mục đích kiểm thử*

*SQL> CREATE TABLESPACE LOSTWRITE DATAFILE '/u02/oradata/shbfin/test.dbf' SIZE 100M;*

*# Kiểm tra lại các data files trong hệ thống, lúc này sẽ trả về kết quả bao gồm cả TEST.DBF*

*SQL> SELECT \* FROM DBA\_DATA\_FILES;*

*# Tạo một bảng mới trong tablespace đã tạo*

*SQL> CREATE TABLE LOSTTABLE*

*(ID NUMBER,*

*PAYLOAD VARCHAR2(100))*

*TABLESPACE LOSTWRITE;*

*# Chuẩn bị dữ liệu, thực hiện đồng bộ sang CSDL dự phòng*

*SQL> INSERT INTO LOSTTABLE(id,payload) VALUES (1, '2 trieu VND');*

*SQL> COMMIT;*

*# Tìm vị trí block của dòng dữ liệu đã được tạo nằm trong data files, ví dụ, dòng dữ liệu này nằm trong block số 133*

*SELECT ROWID, DBMS\_ROWID.ROWID\_BLOCK\_NUMBER(rowid), a.\* FROM LOSTTABLE a;*

Sau khi thực hiện chuẩn bị dữ liệu cũng như tìm được vị trí của dữ liệu trong data files với đơn vị là block, tiến hành thử nghiệm sự cố mất ghi thông qua việc cập nhật dữ liệu mới (với trạng thái CSDL đã biết), khôi phục lại data files trước khi cập nhật (tình trạng mất ghi) và thực hiện cập nhật tiếp dữ liệu sau khi đã bị mất ghi.

*# Sao lưu lại data files trước khi cập nhật, chứa giá trị ‘2 trieu VND’*

*> dd if=/u02/oradata/shbfin/test.dbf of=cpy\_test skip=133 count=1 bs=8192*

*# Tìm kiếm chuỗi ‘2 trieu VND’ trong data files, đảm bảo rằng đây là tệp chứa dữ liệu đã nhập. Ví dụ, kết quả trả về “Binray file cpy\_test.dbf matches”, thì có nghĩa đây là chính là dữ liệu đã nhập*

*> grep '2 trieu VND' cpy\_test.dbf*

*# Thực hiện cập nhật giá trị mới cho bảng losttable*

*SQL> UPDATE losttable*

*SET payload = '5 trieu VND'*

*WHERE id = 1;*

*COMMIT;*

*ALTER SYSTEM CHECKPOINT;*

*ALTER SYSTEM FLUSH BUFFER\_CACHE;*

*# Khôi phục lại data files với giá trị ban đầu (‘2 trieu VND’), khi này, CSDL vừa nhận trạng thái đã ghi, tuy nhiên, giá trị thực sự của data files lại là ‘2 trieu VND’. Như vậy, đã mô phỏng lại sự cố mất ghi thành công*

*> dd if=cpy\_test of=/u02/oradata/shbfin/test.dbf seek=135 count=1 bs=8192 conv=notrunc*

Tại CSDL dự phòng lúc này, tiến trình Redo Log Apply – MRP đối với CSDL dự phòng dạng vật lý sẽ bị tắt và không thể thực hiện được việc áp dụng các thay đổi. Tình trạng của tiến trình MRP trong các trường hợp là giống nhau, tuy nhiên, thông báo lỗi lại mang ý nghĩa rất khác đối với giá trị đặt trong tham số DB\_ LOST\_WRITE\_PROTECTION. Các lỗi sẽ được đưa vào tệp tin cảnh báo (alert.log) của CSDL dự phòng, ghi lại mọi hành động và thông báo của tiến trình MRP.

Với giá trị là MANUAL, CSDL dự phòng không biết được thông tin đọc ghi, chỉ có thông tin về chỉ số SCN, so sánh với nhau, nhưng đưa ra mã lỗi rất chung chung là ORA-600.

*<msg time='2024-04-23T10:26:56.186+07:00' org\_id='oracle' comp\_id='rdbms'*

*type='UNKNOWN' level='16' host\_id='db2'*

*host\_addr='192.168.137.102' pid='2468'>*

*<txt>Errors in file /u01/app/oracle/diag/rdbms/sta/shbfin/trace/shbfin\_mrp0\_2468.trc:*

***ORA-00600: internal error code****, arguments: [3020], [5], [135], [20971655], [], [], [], [], [], [], [], []*

*ORA-10567: Redo is inconsistent with data block (file# 5, block# 135, file offset is 1105920 bytes)*

*ORA-10564: tablespace LOSTWRITE*

*ORA-01110: data file 5: &apos;/u02/oradata/shbfin/test.dbf&apos;*

*ORA-10561: block type &apos;TRANSACTION MANAGED DATA BLOCK&apos;, data object# 73446*

*</txt>*

*</msg>*

*<msg time='2024-04-23T10:26:56.186+07:00' org\_id='oracle' comp\_id='rdbms'*

*type='UNKNOWN' level='16' host\_id='db2'*

*host\_addr='192.168.137.102' pid='2468'>*

*<txt>Background Media Recovery process shutdown (shbfin)*

*</txt>*

*</msg>*

Tuy nhiên, với giá trị là TYPICAL, mọi hành động đọc ghi được ghi theo redo data, CSDL dự phòng có thể thực hiện việc kiểm tra như theo trình bày phần khái niệm. Lúc này, CSDL dự phòng biết được thông tin gây ra hành động mất ghi khi các giá trị trước đó không khớp với nhau, người quản trị sẽ có thông tin rõ ràng hơn để chuẩn bị sửa chữa.

…

***ORA-00752****: recovery detected a lost write of a data block*

…

* 1. Kết quả đạt được sau khi triển khai

Sau khi thực nghiệm triển khai giải pháp Oracle Data Guard cho SHBFinance dựa trên CSDL dự phòng dạng vật lý, ứng dụng nền tảng điện toán đám mây Oracle Cloud Infrastructure, bài đã đạt được một số kết quả thực nghiệm cơ bản. Một số phần nâng cao sẽ tiếp tục được triển khai trong tương lai.

Thứ nhất, ứng dụng nền tảng điện toán đám mây Oracle Cloud Infrastructure, mô phỏng lại về việc tách bạch phần cứng vật lý giữa hai hệ thống CSDL, đảm bảo về ý tưởng dự phòng thay vì thực nghiệm ngay cùng một máy chủ.

Thứ hai, thử nghiệm việc dữ liệu đồng bộ thành công từ CSDL chính sang CSDL dự phòng, kiểm tra các tham số liên quan như các tiến trình chạy (Redo Log Apply, Log Transport Services) thông qua việc mô phỏng lại các cơ chế thông thường diễn ra trong một hệ thống xử lý giao dịch (OLTP).

Thứ ba, giám sát thông số, kích hoạt chuyển đổi vai trò cho cả hai trường hợp có kết hoạch và gặp sự cố đột ngột thông qua thành phần giám sát thứ ba – Observer, dựa trên mô hình giám sát Broker Framework. Khi muốn chuyển đổi vai trò cho các công việc có kế hoạch (bảo trì phần cứng, nâng cấp phần mềm) - Switchover; khi đạt đến điều kiện giới hạn trong thiết lập, Observer ra tín hiệu chuyển đổi trong trường hợp gặp sự cố - Fast-Start Failover. Thay vì thực hiện thủ công, các câu lệnh được thực hiện tự động giúp người quản trị cũng như doanh nghiệp tiết kiệm thời gian.

Thứ tư, kiểm thử các trường hợp xử lý trễ trong việc đồng bộ dữ liệu giữa CSDL chính và CSDL dự phòng trong điều kiện có thể xử lý được thông qua hai cơ chế: Automatic Gap Resolution (CSDL chính chủ động), Fetch Archived Log (CSDL dự phòng chủ động). Các cơ chế xử lý trễ có vai trò quan trọng khi hệ thống gặp các sự cố về đường truyền mạng, gây mất kết nối trong môi trường Oracle Net.

Thứ năm, nghiên cứu, thử nghiệm và chỉ ra sự cố đặc biệt quan trọng trong môi trường Oracle Data Guard – Lost Writes (mất ghi). Hệ thống chỉ ghi lại lỗi chi tiết cho người quản trị rõ ràng khi thực hiện cấu hình tham số DB\_LOST\_WRITE\_ PROTECTION.

* 1. Kết luận chương II

Chương II tập trung vào việc lên kế hoạch và triển khai giải pháp Data Guard cho hệ thống xếp hạng tín dụng nội bộ của công ty, nhằm tăng cường tính khả dụng và dự phòng của CSDL. Trước khi triển khai, cần phải làm rõ vấn đề, như hệ thống đang trong tình trạng nào và rủi ro trong việc khôi phục dữ liệu dựa trên hai chỉ số RPO và RTO. Tiếp theo, thực nghiệm dựa trên RMAN Duplicate và Oracle Cloud Infrastructure đã được áp dụng để mô phỏng tính lại sự chống lỗi thông qua việc tách biệt nhau về mặt vật lý.

Data Guard cung cấp các chế độ bảo vệ trong Data Guard đã được cấu hình để đảm bảo tính an toàn của dữ liệu, với sự lựa chọn phù hợp với yêu cầu của hệ thống. Để giảm tải hệ thống chính, sử dụng tính năng Active Data Guard để thực hiện truy vấn trên hệ thống dự phòng, và các yêu cầu về phần cứng và phần mềm đã được đáp ứng để đảm bảo tính phù hợp.

Không chỉ vậy, chương cũng kiểm thử hoạt động hệ thống đã được thực hiện để đảm bảo tính nhất quán và đáng tin cậy. Cuối cùng, chương đã phân tích sự cố mất dữ liệu và kết quả đạt được sau triển khai đã tập trung vào cải thiện khả năng phục hồi và đảm bảo tính an toàn của dữ liệu. Điều này đã giúp cải thiện hiệu suất và sự tin cậy của hệ thống xếp hạng tín dụng nội bộ của công ty.

1. KẾT LUẬN
   1. Kết quả đóng góp

Bài viết trình bày tổng quan về doanh nghiệp SHBFinance, hệ thống tổng quan của SHBFinance và điểm yếu khi công nghệ dự phòng chưa kịp thời đối với hệ thống xếp hạng tín dụng. Nhằm giải quyết vấn đề về tính dự phòng, giải pháp Oracle Data Guard được áp dụng để tạo nên môi trường có tính khả dụng tin cậy, đem lại lợi ích về đảm bảo vận hành kinh doanh cũng như tiết kiệm nguồn lực thời gian.

Bài viết cũng đã liệt kê những điểm lưu ý khi triển khai, trình bày chi tiết về từng thành phần trong kiến trúc Oracle Data Guard cũng như cách hoạt động, luồng đi dữ liệu đối với mỗi cách cài đặt khác nhau, giúp người hoạch định lựa chọn cách phù hợp nhất với nghiệp vụ và yêu cầu của tổ chức. Ngoài ra, sự cố nghiêm trọng – Lost Writes được chỉ ra ở mức cơ bản, nhằm cung cấp cho người triển khai có cái nhìn toàn diện hơn

Cuối cùng, với việc thực nghiệm, giới thiệu việc sử dụng nền tảng điện toán đám mây Oracle Cloud Infrastructure đã mô phỏng lại việc đặt hai CSDL có tính khác biệt về mặt vật lý trong cùng một hệ thống, tránh sự cố xảy ra đồng thời trên toàn CSDL.

* 1. Hạn chế

Trong quá trình thực nghiệm, cũng như trình bày, còn một số điểm hạn chế như sau, đồng thời, cũng là hướng trong tương lai mà em muốn phát triển cho giải pháp Oracle Data Guard:

Phạm vi tài liệu của doanh nghiệp SHBFinance cung cấp hạn chế, yêu cầu đảm bảo tính bảo mật của công ty do tác giả thuộc bên thứ 03 (giữa doanh nghiệp triển khai và doanh nghiệp được triển khai). Chưa cung cấp được hết tính thực tế trong dự án như hạ tầng công nghệ thông tin, quy trình làm việc, sử dụng hệ thống.

Hệ thống CSDL trong các doanh nghiệp lớn thuộc tổ chức tín dụng như Ngân hàng, Công ty tài chính như SHBFinance đều sử dụng mô hình Multi-Instance hay Real Application Cluster. Trên một máy chủ chứa CSDL, có thể có từ 02 bản thể (Instance) dùng chung một CSDL trở lên nhằm phục vụ mục đích cân bằng tải, đảm bảo tính sẵn sàng cao. Ngoài ra, kiến trúc Container Database cũng được sử dụng để phân chia quyền và dữ liệu theo phòng ban bộ phận. Trong phạm vi của bài khóa luận này chưa thể hiện được các mô hình, kiến trúc trên để đảm bảo thời gian phân chia cho các phần khác, tập trung vào cách hoạt động của Data Guard hơn.

Trong quá trình quản trị và giám sát CSDL Oracle, công cụ Oracle Enterprise Manager Cloud Control được sử dụng nhằm mục đích tối giản, tập trung trong việc quản lý cũng như cung cấp tính trực quan hóa cho người quản trị. Trong bài viết chỉ có công cụ DGMGRL được sử dụng, cung cấp giao diện dòng lệnh và phải thao tác thủ công nhiều.

Chưa thể hiện được hết các tính năng cũng như các tham số điều chỉnh do kiến thức của em còn hạn chế về mạng truyền thông, các kiến trúc vật lý về thiết bị lưu trữ. Ngoài ra, em cũng chưa có kinh nghiệm trong việc quản trị môi trường Data Guard để có thể phát hiện những dấu hiệu bất thường khi giám sát qua các tệp tin logs cũng như các chỉ số thể hiện trên Broker Configuration.

Chưa đưa ra được các cách thức bảo mật trong hạ tầng và trong môi tường Data Guard hay lên lịch cho sao lưu và khôi phục dự phòng bằng CSDL dự phòng. Có các phương thức bảo mật như bảo mật đường truyền trong môi trường Oracle Net, bảo mật thông tin dưới đĩa vật lý, bảo mật xác thực. Ngoài ra, CSDL dự phòng cũng được sử dụng để tạo bản sao lưu nhằm giảm tải cho CSDL chính khi đang vận hành.

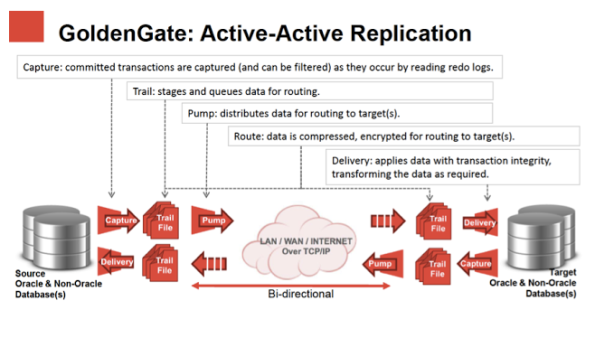
* 1. Hướng phát triển

Triển khai giải pháp Oracle Data Guard dành cho hệ thống xếp hạng tín dụng nội bộ của SHBFinance là một trong những cách thức giúp cho doanh nghiệp rút ngắn thời gian về mặt khôi phục cũng như giảm thiểu số lượng dữ liệu mất mát. Hiện nay, Data Guard được xem là giải pháp không thể thiếu cho các doanh nghiệp nếu muốn đạt tới việc khôi phục sau thảm họa, bên cạnh các giải pháp mang tính sẵn sàng cao đối với các doanh nghiệp sử dụng CSDL của Oracle.

Trong tương lai gần, có một số mục tiêu mà em đặt ra để tiếp tục phát triển đề tài này như sau:

* Thực hiện thử nghiệm, triển khai cơ chế hoạt động và các sự cố mang phạm vi rộng hơn đối với toàn hệ thống, dữ liệu lớn, trên các môi trường và kiến trúc khác như Real Application Cluster và Container Database thay vì Single Instance Database
* Triển khai tính năng DML Redirection, cho phép thực hiện các thao tác chỉnh sửa như INSERT, UPDATE, DELETE đối với CSDL dự phòng, thay vì chỉ có thể SELECT như trước đây đối với Active Data Guard
* Triển khai mô hình sử dụng CSDL dự phòng dạng Far Sync, thực hiện cơ chế truyền broad – cast tới những CSDL dự phòng khác. Mô hình này giúp CSDL chính có thể hoạt động ổn định hơn, đảm bảo dữ liệu hạn chế mất khi đặt các chế độ bảo vệ Maximum Protection và Maximum Availability
* Triển khai tích hợp giao diện quản trị Oracle Enterprise Manager Cloud Control trong môi trường Data Guard, đơn giản hóa nhiệm vụ quản trị của người quản trị viên
  1. So sánh với giải pháp Oracle Golden Gate

Giải pháp Oracle Data Guarsd và Oracle GoldenGate đều thuộc chung một loại trong kiến trúc MAA của Oracle là “replication” – nhân bản hệ thống, với mục đích là đồng bộ hóa dữ liệu giữa hai hoặc nhiều hệ thống, phục vụ cho việc dự phòng, di chuyển dữ liệu hoặc xây dựng hệ thống CSDL khác tương tự. Khi tìm hiểu Oracle Data Guard và Oracle GoldenGate, hai giải pháp này thường được xem là có điểm tương đồng, dẫn đến việc nhầm lẫn trong việc lựa chọn giải pháp phục vụ cho việc khôi phục hệ thống sau thảm họa.



Hình 40 Giải pháp Oracle GoldenGate

Trong Oracle Data Guard, các thay đổi được lưu vào redo buffer cache rồi đến redo log. Tùy thuộc vào chế độ cài đặt cho Redo Transport Services như SYNC hoặc ASYNC mà redo data được gửi và lưu xuống đĩa ngay lập tức hoặc có thể trễ hơn tới CSDL dự phòng. Nhờ vậy, các thay đổi tại CSDL chính đều được gửi đến CSDL dự phòng. Khi xảy ra sự cố, quá trình failover diễn ra tự động, kịp thời, việc vận hành được CSDL dự phòng đảm nhận.

Đối với Oracle GoldenGate, CSDL sử dụng Trail – file, lưu lại các thông tin thay đổi của Redo Log, gửi sang cho các CSDL khác, sử dụng kỹ thuật đảo ngược (reverse engineers) thành SQL và áp dụng thay đổi này vào CSDL nhận. Ngoài ra, các dữ liệu được truyền được cấu hình chỉ truyền các thông tin cần thiết mà không truyền toàn bộ dữ liệu thông qua bộ lọc (filters), các hoạt động riêng lẻ (INSERT hoặc UPDATE hoặc DELETE) hoặc bảng/cột cố định. Bảng so sánh dưới đây sẽ cung cấp thông tin tổng quan hơn để lựa chọn các giải pháp phù hợp:

|  |  |  |  |
| --- | --- | --- | --- |
| **Số thứ tự** | **Nội dung so sánh** | **Oracle GoldenGate** | **Oracle Data Guard** |
| **1** | Nền tảng hệ thống | Hỗ trợ nhiều loại CSDL khác nhau | Hỗ trợ chỉ cùng CSDL Oracle |
| **2** | Cách thức đồng bộ | Đồng bộ nhiều chiều | Đồng bộ một chiều |
| **3** | Loại dữ liệu hỗ trợ | Không hỗ trợ XML và BLOB | Không giới hạn |
| **4** | Sao lưu dữ liệu | Chỉ những dữ liệu giống nhau mới được sao lưu | CSDL chính và CSDL phụ chính là bản sao lưu của nhau |
| **5** | Giá cả, bản quyền | Active Data Guard trong phiên bản Oracle Database Enterprise Edition, cần mua bản quyền để sở hữu.  Data Guard cơ bản không cần mua bản quyền, nhưng thiếu tính năng truy vấn đối với CSDL dự phòng. | Cần mua bản quyền cho tất cả CSDL tham gia vào môi trường GoldenGate. Tuy nhiên, khi mua bản quyền, tính năng Active Data Guard cũng nằm trong gói này. |

Bảng 10 So sánh giữa GoldenGate và Data Guard

Như vậy, đối với giải pháp Oracle GoldenGate sẽ phù hợp với các nhu cầu như sau, thay vì dùng giải pháp Data Guard:

* Thực hiện việc đồng bộ dữ liệu giữa một hoặc nhiều bảng tới các CSDL có chế độ đọc ghi
* Thực hiện việc đồng bộ và chuyển hóa dữ liệu tới các bảng
* Đồng bộ thực hiện theo hai chiều, với các cơ chế xử lý xung đột
* Đồng bộ trong hệ thống với các nền tảng CSDL khác nhau (PostgreSQL, SQL Server, Teradata, TimesTen, MySQL, DB2)

PHỤ LỤC

**Cài đặt kết nối SSH, dùng MobaXterm điều khiển**

**Cài Oracle Database 19c Software, Database**

**Cài Oracle Client và Observer để giám sát**

TÀI LIỆU THAM KHẢO

thuvienphapluat.vn. (2023, June 27). *Thông tư 11/2021/TT-NHNN quy định về phân loại tài sản có, mức trích, phương pháp trích lập dự phòng rủi ro và việc sử dụng dự phòng để xử lý rủi ro trong hoạt động của tổ chức tín dụng, chi nhánh ngân hàng nước ngoài do Ngân hàng Nhà nước Việt Nam ban hành*. Retrieved April 24, 2024, from THƯ VIỆN PHÁP LUẬT website: <https://thuvienphapluat.vn/van-ban/Tien-te-Ngan-hang/Thong-tu-11-2021-TT-NHNN-su-dung-du-phong-de-xu-ly-rui-ro-hoat-dong-cua-to-chuc-tin-dung-483459.aspx>

Nguyễn Nhật Vy. (2022, October 2). *Hệ thống xếp hạng tín dụng nội bộ của tổ chức tín dụng phi ngân hàng là hệ thống gồm những gì? Hệ thống này được xây dựng dựa trên những nguyên tắc nào?* Retrieved March 24, 2024, from Thuvienphapluat.vn website: <https://thuvienphapluat.vn/phap-luat/he-thong-xep-hang-tin-dung-noi-bo-cua-to-chuc-tin-dung-phi-ngan-hang-la-he-thong-gom-nhung-gi-he-th-901385-40740.html>

Giới thiệu | SHBFinance. (2024). Retrieved March 23, 2024, from Shbfinance.com.vn website: <https://www.shbfinance.com.vn/ve-chung-toi/gioi-thieu-chung>

*Most popular database management systems 2023* | Statista. (2023). Retrieved April 24, 2024, from Statista website: <https://www.statista.com/statistics/809750/worldwide-popularity-ranking-database-management-systems/>

Liu Xiu-ju. (2010). *A brief analysis of the disaster recovery backup technology in Oracle database DataGuard*. IEEE. <https://doi.org/10.1109/indusis.2010.5565635>

Brief Information About the Oracle Data Guard. (2011, June 16). Retrieved April 3, 2024, from Talip Hakan Ozturk’s ORACLE BLOG website: <https://taliphakanozturken.wordpress.com/2011/06/16/brief-information-about-the-oracle-data-guard/>

Data Guard Broker Architecture Oracle 10g data guard. (2017). Retrieved April 4, 2024, from Rampant-books.com website: <http://www.rampant-books.com/art_kumar_dg_broker_architiecture.htm>

Duplicating Databases. (2019). Retrieved April 20, 2024, from Oracle Help Center website: <https://docs.oracle.com/en/database/oracle/oracle-database/19/bradv/rman-duplicating-databases.html>

Guide to Oracle Data Guard Fast-Start Failover. (2019). Retrieved April 24, 2024, from Oracle.com website: <https://www.oracle.com/technical-resources/articles/smiley-fsfo.html>

Li, T. (2024). *Reliable Business Continuity at an Entry-Level Price | Buffalo Americas*. Retrieved April 5, 2024, from Buffalotech.com website: <https://www.buffalotech.com/resources/reliable-business-continuity-at-an-entry-level-price-jm>

Onur ARDAHANLI. (2020, January 8). *Automatic GAP Resolution*. Retrieved April 4, 2024, from Database Tutorials website: <https://dbtut.com/index.php/2020/01/08/automatic-gap-resolution/>

Oracle Data Guard Best Practices. (2019). Retrieved April 24, 2024, from Oracle Help Center website: <https://docs.oracle.com/en/database/oracle/oracle-database/19/haovw/oracle-data-guard-best-practices.html>

Oracle Dataguard Architecture & Background processes. (2020, February 20). Retrieved April 4, 2024, from Doyensys Blog website: <https://doyensys.com/blogs/oracle-dataguard-architecture-background-processes/>

Stryker, C. (2023, May 12). *MTTR vs. MTBF: What’s the difference?* Retrieved April 1, 2024, from IBM Blog website: <https://www.ibm.com/blog/mttr-vs-mtbf/>

TekDBA. (2019, August 15). *Oracle Dataguard (DG) and Role of Standby Redo Logs (SRL) in Oracle Dataguard*. Retrieved April 3, 2024, from TekDBA website: <https://tekdba.wordpress.com/2019/08/15/oracle-dataguard-dg-and-role-of-standby-redo-logs-srl-in-oracle-dataguard/>

*What you need to know about Lost Writes*. (2021). Retrieved April 24, 2024, from Dbvisit.com website: <https://dbvisit.com/blog/what-you-need-know-about-lost-writes>

Yu, P., Zhou, N., & Sun, H. (2011). *The application of Oracle Data Guard in the Logistics Distribution Management Platform*. <https://doi.org/10.1109/iccsnt.2011.6182094>

Fuller, M. (n.d.). *Oracle Database 12c: Data Guard Administration*. Joseph Fernandez, Veena Narasimhan.

1. Moody's là một cơ quan đánh giá tín nhiệm tín dụng hàng đầu trên thế giới. Họ cung cấp các báo cáo và xếp hạng tín dụng cho các công ty, quốc gia và các tổ chức khác để đánh giá khả năng trả nợ của họ và độ rủi ro đầu tư. [↑](#footnote-ref-1)