



DOI:10.22144/ctu.jsi.2017.010

LỰA CHỌN NỀN TẢNG PHẦN MỀM NGUỒN MỞ XÂY DỰNG TRỰC TÍCH HỢP ỨNG DỤNG VÀ DỊCH VỤ

Nguyễn Hữu Vân Long¹, Nguyễn Thế Anh¹, Trương Hoàng Nguyên², Lý Minh Phương³,
Trần Công Án¹ và Ngô Bá Hùng¹

¹Khoa Công nghệ Thông tin và Truyền Thông, Trường Đại học Cần Thơ

²Sở Khoa học và Công nghệ tỉnh Sóc Trăng

³Sở Khoa học và Công nghệ tỉnh Vĩnh Long

Thông tin chung:

Ngày nhận bài: 15/09/2017

Ngày nhận bài sửa: 10/10/2017

Ngày duyệt đăng: 20/10/2017

Title:

Evaluating and selecting an appropriate open source software to build an enterprise service bus

Từ khóa:

Kiến trúc hướng dịch vụ, Trục tích hợp dịch vụ mã nguồn mở

Keywords:

Bus ESB, enterprise, Jboss, fuse SOA, service mule, WSO2

ABSTRACT

The need for application integration is increasing, especially when organizations shift to digital economy and e-government. Enterprise service bus is the most advantage application integration model today. Choosing an enterprise service bus solution that is suited to not only the current but also the long-term needs of an organization is a big challenge. This paper reviews several recently related research approaches and presents the results of evaluating and selecting an appropriate open-source software to build an enterprise service bus for An Giang province in the context of e-government building.

TÓM TẮT

Tích hợp ứng dụng là nhu cầu đang tăng đối với các tổ chức khi chuyển sang nền kinh tế và quản trị số. Trục tích hợp dịch vụ là mô hình tích hợp ứng dụng được đánh giá có nhiều ưu thế nhất hiện nay. Lựa chọn một giải pháp xây dựng trục tích hợp dịch vụ phù hợp yêu cầu hiện tại và lâu dài cho một tổ chức là một vấn đề khó. Nghiên cứu này tóm lược một số hướng nghiên cứu liên quan gần đây và trình bày kết quả việc đánh giá và chọn lựa một giải pháp phần mềm nguồn mở phù hợp để xây dựng một trục tích hợp dịch vụ trong bối cảnh xây dựng chính quyền điện tử tỉnh An Giang.

Trích dẫn: Nguyễn Hữu Vân Long, Nguyễn Thế Anh, Trương Hoàng Nguyên, Lý Minh Phương, Trần Công Án và Ngô Bá Hùng, 2017. Lựa chọn nền tảng phần mềm nguồn mở xây dựng trục tích hợp ứng dụng và dịch vụ. Tạp chí Khoa học Trường Đại học Cần Thơ. Số chuyên đề: Công nghệ thông tin: 72-79.

1 GIỚI THIỆU

Các tổ chức hiện nay thường chạy đồng thời hàng trăm ứng dụng có nguồn gốc rất khác nhau, chẳng hạn phần mềm tự phát triển, phần mềm mã nguồn mở, phần mềm thương mại... để đáp ứng đầy đủ quy trình nghiệp vụ của tổ chức. Mặc dù khác biệt nhau về nguồn gốc, các ứng dụng cần thiết phải giao tiếp và trao đổi được dữ liệu cho nhau để có thể hoàn thành các quy trình nghiệp vụ

chung của tổ chức. Các tổ chức vì thế có nhu cầu sử dụng một hoặc nhiều giải pháp để tích hợp các ứng dụng lại với nhau, nhờ đó các ứng dụng tuy khác nhau về ngôn ngữ lập trình, cơ sở dữ liệu, giao thức vẫn có thể phối hợp với nhau trong việc hoàn thành một quy trình nghiệp vụ của tổ chức (Wilem-Jan Van den Heuvel và Jon van Hilegersberg, 2001). Hiện tại, việc tích hợp ứng dụng (EAI – Enterprise Application Integration) có thể được chia thành 3 nhóm giải pháp là dựa

trên các phần mềm trung gian định hướng thông điệp (MOM – Message Oriented Middleware); dựa trên Kiến trúc hướng dịch vụ (SOA – Service Oriented Architecture) và công nghệ dịch vụ web (Web Services); và dựa trên một trục tích hợp dịch vụ (ESB – Enterprise Service Bus). Trong đó, giải pháp sử dụng ESB đang được sử dụng rộng rãi bởi các tổ chức lớn trong nhiều lĩnh vực khác nhau (Ebay, Logicoy...) tại nhiều quốc gia nhờ vào những tính năng tích hợp mạnh mẽ mà nó cung cấp vượt trội so với hai giải pháp đầu tiên (P.de Leusse et al., 2007). ESB là một hạ tầng tích hợp ứng dụng phân tán, đồng thời tận dụng được những điểm mạnh hướng thông điệp của MOM và sử dụng các chuẩn mở của công nghệ Web Services. ESB cung cấp các dịch vụ định tuyến (routing), triệu gọi (invocation) và điều chỉnh (mediation) để hỗ trợ cho việc tương tác giữa những ứng dụng và dịch vụ tách biệt phân tán trong một cách thức an toàn và tin cậy. Chúng có thể được triển khai bằng một giải pháp phần mềm thương mại hay một giải pháp dựa trên các phần mềm mã nguồn mở (FOSS- Free Open Source Software).

Bài viết này sẽ trình bày kết quả nghiên cứu về việc đánh giá và chọn lựa một giải pháp hoàn toàn dựa trên FOSS để xây dựng một trục tích hợp dịch vụ với nhiệm vụ kết nối, tích hợp các ứng dụng quản lý nhà nước của tỉnh An Giang trong bối cảnh xây dựng chính quyền điện tử của tỉnh.

Phần kết tiếp sẽ trình bày kiến trúc tổng quan, các tính năng cơ bản của một ESB. Một số giải pháp ESB dựa trên FOSS phổ biến nhất hiện nay là đối tượng được đánh giá và chọn lựa sẽ được giới thiệu ở phần 3. Bộ tiêu chí dùng cho việc đánh giá chọn lựa các giải pháp ESB dựa trên FOSS được đề xuất ở phần 4. Phần thứ 5 sẽ trình bày tiến trình so sánh, đánh giá và chọn lựa giải pháp ESB dựa trên FOSS cho tỉnh An Giang cùng với kết quả thử nghiệm ban đầu. Cuối cùng là kết luận và đề xuất hướng phát triển tiếp theo của nghiên cứu.

2 TRỤC TÍCH HỢP DỊCH VỤ ESB

Phần này sẽ mô tả về kiến trúc tổng quát và các tính năng chính của một trục tích hợp. Ngoài ra, một số giải pháp ESB dựa trên FOSS cũng được giới thiệu nhằm làm rõ tính phổ biến và tin cậy của giải pháp này.

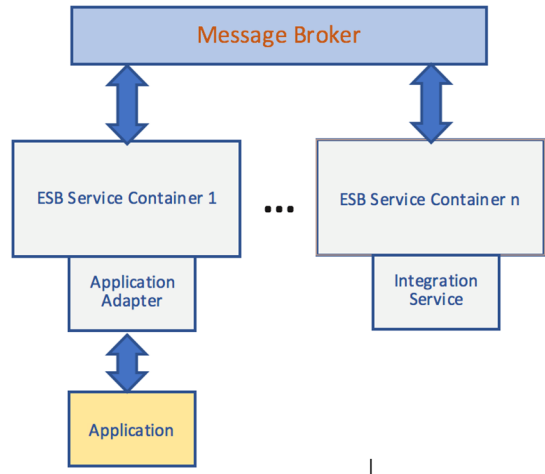
2.1 Kiến trúc trục tích hợp ESB

ESB là một hạ tầng tích hợp ứng dụng phân tán, dựa trên thông điệp (message) và các chuẩn mở. ESB cung cấp các dịch vụ định tuyến thông điệp (routing), triệu gọi (invocation), và điều chỉnh (mediation) để hỗ trợ cho việc tương tác giữa

những ứng dụng và dịch vụ tách biệt phân tán một cách an toàn và tin cậy.

Hình 1 mô tả kiến trúc tổng quát của một ESB, đó là sự kết hợp giữa mô hình của MOM và SOA. Trong đó, Message Broker đóng vai trò là Kênh (Bus) chuyển tiếp các thông điệp qua lại giữa các ứng dụng hay các dịch vụ tích hợp. ESB được xây dựng dựa trên một tập các Môi trường chứa đựng dịch vụ (ESB Service Container) được triển khai phân tán trên các máy tính trong mạng. Mỗi môi trường chứa đựng dịch vụ cho phép triển khai các dịch vụ như Bộ định tuyến thông điệp (Routers), các Bộ chuyển đổi thông điệp (Transformers), các Bộ điều biến ứng dụng (Application Adapter), và các tiện ích phục vụ truyền thông khác. Các ứng dụng hay các dịch vụ kết nối với Bus nhờ vào các Adapter. Để hỗ trợ SOA, các Service Container được cài đặt bằng các công nghệ của Web Services.

Mục đích tổng quát của ESB là có thể truyền tải các thông điệp và thực hiện việc tích hợp mà không cần phải viết lại mã nguồn phần mềm. Tùy vào mục đích sử dụng mà các thành phần tổng quát được cung cấp có thể được cấu hình lại theo các cách thức khác nhau.



Hình 1: Tích hợp ứng dụng doanh nghiệp với kênh dịch vụ doanh nghiệp ESB

2.2 Các tính năng chính trong ESB

Theo Falko Menge (2007), một giải pháp trục tích hợp dịch vụ cần cung cấp một bộ các tính năng cơ bản như được mô tả phía dưới đây.

2.2.1 Triệu gọi (Invocation)

Nhóm tính năng này cho phép ESB gửi các yêu cầu, nhận các trả lời từ các dịch vụ và tài nguyên được tích hợp, hỗ trợ các giao thức

truyền tải dữ liệu theo kiểu đồng bộ hoặc dị bộ.

2.2.2 Định tuyến (Message Mediation)

Nhóm tính năng này cho phép ESB xác định đích đến của một thông điệp trong quá trình vận chuyển thông điệp. Định tuyến là một tính năng cơ bản của một ESB bởi vì nó cho phép tách rời giữa nguồn gửi thông điệp với các nơi nhận thông điệp. ESB có những khả năng định tuyến như: chuyển tiếp thông điệp đến địa chỉ nhận căn cứ vào nội dung của thông điệp, chuyển cùng một thông điệp đến nhiều đích đến, chia nhỏ thông điệp ra các phần nhỏ và gửi mỗi phần đến các đích đến khác nhau, tổng hợp thông điệp từ nhiều nguồn và gửi cho đích đến.

2.2.3 Điều chỉnh (Message Mediation)

Nhóm tính năng này tham khảo đến tất cả các loại chuyển đổi hay dịch giữa các nguồn khác biệt, bao gồm việc chuyển đổi giao thức truyền tải, chuyển đổi định dạng thông điệp và chuyển đổi nội dung của các thông điệp. Đây là tính năng rất quan trọng và cần thiết trong việc thực hiện tích hợp các ứng dụng lên trực.

2.2.4 Bộ điều biến (Adaptor)

Nhóm tính năng này giúp một ESB có thể tích hợp, liên thông nhiều loại ứng dụng khác nhau. Bản chất của các Adapter là các gói phần mềm thực hiện nhiệm vụ nối kết đến các giao diện xử lý các giao dịch riêng rẽ, các API, các cấu trúc dữ liệu mà các ứng dụng đã công bố và thể hiện thông qua một giao diện chuẩn.

2.2.5 Chứng thực và bảo mật (Authentication and Security)

Nhóm tính năng này cho phép một ESB kiểm soát các ứng dụng được cấp phép tích hợp lên trực, được cấp phép truyền tải dữ liệu đến một đích đến cho trước thông qua cơ chế chứng thực và bộ lọc dữ liệu. Ngoài ra, ESB cũng cho phép tùy chọn mở rộng các mức độ bảo mật cho dữ liệu của các dịch vụ tham gia tích hợp.

2.3 Ứng dụng của ESB

Để xử lý cho một lượng lớn các giao dịch diễn ra hằng ngày, năm 2009, eBay đã triển khai sử dụng ESB trong các dịch vụ cơ bản (WSO2 Case Studies, 2016). Đại học Witwatersrand, Châu Phi đã sử dụng ESB để quản lý thông tin sinh viên trong ứng dụng quản lý nhà trường (MuleSoft Case Studies, 2016).

Việc ứng dụng ESB là giải pháp xây dựng chính quyền điện tử đã được thực hiện từ lâu trên thế giới. Marcello Castellano *et al.* (2011) đã giới thiệu một tiếp cận mới nhằm giúp chính quyền các

tiểu vương quốc Ả rập thực hiện tiến trình cải tổ để xây dựng chính quyền điện tử. Zakaria.I Saleh *et al.* (2013) cũng đã đề xuất một framework xây dựng chính quyền điện tử cho chính phủ Jordan thông qua việc tích hợp các ứng dụng khác nhau dựa trên công nghệ SOA, ESB.

3 CÁC ESB MÃ NGUỒN MỞ PHỔ BIẾN

Phần trên đã trình bày kiến trúc tổng quát các tính năng chính mà một giải pháp phần mềm cung cấp giải pháp ESB cần có. Phần này sẽ giới thiệu một số ESB dựa trên FOSS tiêu biểu nhất hiện nay như: Mule, RedHat JBoss Fuse, Open ESB, WSO2.

3.1 Mule

Mule là một trong những ESB mã nguồn mở đạt được nhiều thành công khi trở thành sự lựa chọn cho các doanh nghiệp, tổ chức trên nhiều lĩnh vực khác nhau như y tế cộng đồng, bảo hiểm, tài chính, các cơ quan chính phủ... Mule cung cấp phiên bản Community và Enterprise có chung đặc trưng như: đối tượng tích hợp vào trực là bất kỳ đối tượng nào; thông điệp truyền tải có định dạng tùy ý; sử dụng kiến trúc SEDA.

3.2 Red Hat JBoss Fuse

JBoss Fuse là một ESB sử dụng công nghệ JBI (Java Business Integration) làm nền tảng tích hợp ứng dụng. Nhờ vậy, JBoss Fuse thừa kế các tính năng trên JBI như: các bộ định tuyến và chuẩn hoá thông điệp, các tác vụ để quản lý và cài đặt các thành phần trong trực tích hợp... JBoss Fuse còn sử dụng Fuse Fabric giúp quản lý phân tán một lượng lớn các cluster trực, đồng thời cho phép một dịch vụ quảng bá tính khả dụng đến các dịch vụ khác trong cluster mà không cần thực hiện cấu hình nhiều lần.

3.3 Open ESB

OpenESB được phát triển bởi Sun Microsystems và có quá trình phát triển lâu đời nhất. OpenESB công bố 100% mã nguồn cho cộng đồng sử dụng và phát triển. Pymma và Logicoy là hai thành viên lớn đã thương mại hoá thành công OpenESB.

OpenESB áp dụng các công nghệ như: JBI, XML, WSDL, BPEL... giúp kết nối hệ thống cũ và phát triển hệ thống mới đơn giản, nhanh chóng, đồng thời duy trì được tính ổn định và hiệu quả trong thời gian hoạt động lâu dài.

3.4 WSO2

WSO2 được thiết kế với mục tiêu trở thành 1 trực tích hợp có hiệu suất cao nhất, an toàn nhất và tương thích với nhiều nền tảng tích hợp nhất. WSO2 là lựa chọn thường xuyên và tin cậy cho các tổ chức lớn như eBay, SUVA, Aspire Systems...

WSO2 vượt trội trong hỗ trợ các công cụ trực quan như: công cụ thiết kế luồng dữ liệu, công cụ ánh xạ dữ liệu...

WSO2 nổi bật ở năng suất hoạt động của các bộ định tuyến và chuyển đổi thông điệp nhờ engine của Apache Synapse. Hơn nữa, WSO2 cũng chứng tỏ tính ổn định, an toàn khi truyền lượng lớn dữ liệu với tốc độ cao (Shafreen Anfar, 2014). Ngoài ra, WSO2 cung cấp miễn phí 100% hàng loạt các dịch vụ kèm theo hỗ trợ cho quản lý, phân tích, lưu trữ dữ liệu...

4 XÂY DỰNG BỘ TIÊU CHÍ ĐÁNH GIÁ CÁC TRỰC TÍCH HỢP

Phần trên đã giới thiệu tổng quan bốn giải pháp ESB dựa trên FOSS phổ biến nhất hiện nay. Phần này sẽ giới thiệu một số tiêu chí đánh giá cho các trực tích hợp trong các nghiên cứu liên quan. Trên cơ sở đó sẽ đề xuất một bộ tiêu chí tổng hợp chung được dùng cho việc chọn lựa giải pháp ESB dựa trên FOSS.

4.1 Các nghiên cứu liên quan

David Kusák (2010) đã thực hiện đánh giá các giải pháp ESB dựa trên FOSS theo các tiêu chí như: Quản lý dự án, Tình hình phát triển phần mềm, Kiến trúc giải pháp mức ở thấp và mức cao. Tác giả đã trình bày các kịch bản kiểm thử và tổng hợp kết quả lần lượt cho từng tiêu chí bằng hình thức chấm điểm.

Một nghiên cứu khác của Sanjay P.Ahuja và Amit Patel (2011) đã tiến hành đánh giá các giải pháp ESB dựa trên FOSS bằng cách so sánh hiệu suất xử lý giao dịch (Throughput) và thời gian đáp ứng trung bình (Mean Response Time) cho việc xử lý thông điệp. Các kịch bản kiểm thử liên quan đến khả năng co giãn (Scalability Test), khả năng tải (Load Handling Test), khả năng chuyển đổi và định tuyến dữ liệu (Transformation and Routing Test).

Một số hướng nghiên cứu khác tập trung vào một hoặc một vài tiêu chí nhất định để so sánh các nền tảng tích hợp. Mikael Ahlberg (2010) đã tiến hành nghiên cứu cơ chế truyền tải thông điệp tin cậy trong trực tích hợp. Shafreen Anfar (2014) hay Tobia Kruessmann *et al.* (2009) đã trình bày về tính ổn định và hiệu suất hoạt động của các nền tảng tích hợp mà tập trung ở giai đoạn xử lý và chuyển đổi thông điệp. Báo cáo của Ken Vollmer *et al.* (2011) đã đưa ra các đánh giá và dự đoán xu hướng phát triển trong tương lai của các nền tảng tích hợp.

Trong một nghiên cứu gần đây, Robin Singh Bhadoria *et al.* (2016) đã thống kê đầy đủ các khảo sát từ năm 2007 đến 2015 trên các ESB khác nhau

với cách tiếp cận khác nhau, đồng thời cũng giới thiệu tổng quan về các ESB đang có mặt trên thị trường cùng một bộ tiêu chí rất tốt để so sánh với các thông số đặc trưng như: Monitoring and control, Open Source... Tuy nhiên, hạn chế của nghiên cứu này là các tiêu chí chỉ được đánh giá là Có, Không hoặc so sánh trên thang mức độ (High, Medium, Low) trên vài tiêu chí với mỗi ESB.

4.2 Xây dựng bộ tiêu chí đánh giá ESB

Từ kết quả của các nghiên cứu trên, bài viết đề xuất bộ tiêu chí được tổng hợp lại và có bổ sung tiêu chí mới kèm theo điểm số đề xuất trên từng tiêu chí nhằm lựa chọn một ESB FOSS thích hợp nhất. Sau đây là mô tả các tiêu chí trong bộ tiêu chí:

- *Khả năng sử dụng (Usability)*: được đánh giá qua mức độ thuận tiện trong việc cài đặt, môi trường phát triển thân thiện; các thiết lập để sử dụng; hướng dẫn và công cụ cài đặt ban đầu.
- *Khả năng quản trị (Maintainability)*: được đánh giá qua mức độ hỗ trợ của công cụ quản trị; giải pháp quản trị đơn giản hay không; giao diện quản lý dịch vụ gồm những tính năng gì.
- *Hỗ trợ cộng đồng (Community Support)*: đánh giá giữa phiên bản trả phí và miễn phí; tài liệu, khoá học, quy mô cộng đồng...
- *Chức năng (Functionality)*: được đánh giá qua các dịch vụ sẵn dùng trong tích hợp ESB.
- *Tính mềm dẻo (Flexibility)*: được đánh giá qua khả năng tùy chỉnh, phát triển thêm tính năng cho ESB.
- *Khả năng mở rộng (Expandability)*: được đánh giá qua khả năng và phạm vi mở rộng của ESB.
- *Các bộ kết nối (Connectors)*: được đánh giá qua số lượng các Adaptor cung cấp tương tích cho ứng dụng trên những nền tảng khác nhau được tích hợp.
- *Giá thành (Cost)*: được đánh giá bởi tổng chi phí để triển khai ESB; các chi phí phụ trội cho việc bổ sung dịch vụ, tính năng riêng lẻ từ nhà phát triển hoặc bên thứ ba.
- *Giấy phép (Licensing)*: được đánh giá qua loại chứng nhận được cấp; việc nâng cấp hoặc hạ cấp trực tích hợp có trả phí hay không.
- *An toàn dữ liệu (Security)*: được đánh giá qua các mức độ hỗ trợ về bảo mật, cơ chế chứng thực, cơ chế tích hợp dịch vụ vào trực.
- *Mức độ sử dụng trong doanh nghiệp (Industry wide acceptance)*: được đánh giá qua số lượng, quy mô và lĩnh vực của các tổ chức tham gia sử dụng trực tích hợp ESB.

– *Hiệu suất hoạt động (Performance)*: được đánh giá qua tốc độ xử lý các thông điệp trên trục (xử lý dữ liệu trong thông điệp, định tuyến thông điệp...)

– *Tính ổn định (Stability)*: được đánh giá qua khả năng hoạt động với dữ liệu lớn, khả năng chịu lỗi, khả năng chuyển dữ liệu độ trễ thấp.

5 LỰA CHỌN GIẢI PHÁP ESB

Để chọn lựa được một giải pháp ESB dựa trên FOSS, các giải pháp đã được giới thiệu ở phần 3 sẽ được đánh giá theo bộ tiêu chí đã được đề xuất ở phần 4.1. Sau đó sẽ tiến hành chọn lựa một trong số chúng dựa trên các yêu cầu thực tế.

5.1 Đánh giá các giải pháp ESB nguồn mở

Bảng 1 là kết quả so sánh đánh giá 4 giải pháp ESB đã được giới thiệu ở phần 3 với bộ tiêu chí đề xuất ở phần 4.2. Trong mỗi tiêu chí có đánh giá trên số sao (*), cao nhất là 5* và thấp nhất là 1*. Các đặc trưng riêng của từng giải pháp tích hợp sẽ được trình bày thêm trong mỗi tiêu chí để làm rõ sự khác biệt giữa chúng.

5.2 Lựa chọn trục tích hợp

Việc lựa chọn một ESB phù hợp là vấn đề cần được khảo sát và phân tích gắn liền với tổ chức triển khai thực tế. Vì vậy, chúng tôi đề xuất một bộ câu hỏi kết hợp với bộ tiêu chí đã giới thiệu để gợi ý lựa chọn: quy mô và khả năng mở rộng; nhu cầu trao đổi dữ liệu, các giao dịch; chi phí đầu tư cho từng chức năng, cho toàn bộ giải pháp; mức độ an toàn, bảo mật thông tin; tài liệu hướng dẫn, các khoá tập huấn; nguồn tài nguyên cần đảm bảo; chịu lỗi, cân bằng tải trong hệ thống; nghiệp vụ của tổ chức triển khai...

Từ kết quả điểm số trung bình (Overall Score), đặc trưng của từng tiêu chí trong Bảng 1 và các kết quả nghiên cứu liên quan, một số kết luận được rút ra như sau: *Về hiệu suất hoạt động và chỉ số đo lường ở kết quả đầu ra*, WSO2 và Mule đạt kết quả tốt và ổn định nhất, OpenESB trong một số kịch bản cũng đạt kết quả khả quan, riêng JBoss Fuse không gây nhiều ấn tượng về kết quả; *Về tính mở trong xây dựng trục*, WSO2 được đánh giá cao nhờ phiên bản 100% nguồn mở, trong khi Mule chỉ hỗ trợ tốt trong phiên bản Enterprise, tương tự với phiên bản thương mại OpenESB của Logicoy; *Về*

trình trạng của hệ thống cài đặt trục tích hợp mà ở đây là tổ chức hành chính tỉnh An Giang yêu cầu an toàn, khả năng chịu lỗi và mềm dẻo trong tích hợp (Ngô Bá Hùng, 2016) thì WSO2, Mule Enterprise và JBoss Fuse là có thể đáp ứng được. WSO2 cung cấp miễn phí 100% các kịch bản gia cố an ninh trên trục. Với WSO2, cơ chế an toàn được tổ chức thành các cấp rất chặt chẽ nhưng không quá phức tạp khi triển khai vì vậy cực kỳ phù hợp trong ngữ cảnh này. Vì vậy, WSO2 được chọn làm nền tảng tích hợp triển khai các ứng dụng quản lý nhà nước của tỉnh An Giang trong tiến trình xây dựng chính quyền điện tử của tỉnh.

5.3 Một số kết quả ban đầu

Hiện tại, trục đã tích hợp dịch vụ dựa trên WSO2 cho tỉnh An Giang đã triển khai thử nghiệm tại địa chỉ <http://113.161.213.226>. Đến thời điểm hiện tại, mô hình tích hợp phần mềm một cửa với phần mềm đăng ký đất đai đã dựa trên trục được xây dựng và thử nghiệm.

6 KẾT LUẬN VÀ ĐỀ XUẤT

Trục tích hợp dịch vụ ESB là một hạ tầng tích hợp ứng dụng phân tán cung cấp các dịch vụ như: định tuyến thông điệp, triệu gọi và điều chỉnh để hỗ trợ cho việc tương tác giữa những ứng dụng và dịch vụ tách biệt theo nhiều phương thức khác nhau. Trong số các ESB dựa trên FOSS tiêu biểu, bài báo này đã đánh giá Mule ESB, JBoss Fuse, OpenESB và WSO với mức độ hỗ trợ riêng của từng nền tảng có sự khác biệt rõ ràng. Để thuận tiện cho việc đánh giá, so sánh các ESB và lựa chọn một ESB dựa trên FOSS thích hợp, một bộ các tiêu chí đã được đề xuất dựa trên sự tổng hợp kết quả của các nghiên cứu liên quan đồng thời bổ sung một số tiêu chí mới và thực hiện chấm điểm trên tất cả tiêu chí. Trong đó, WSO2 ESB được đánh giá cao nhất dựa trên chỉ số tổng quát, các đặc trưng ấn tượng trong các tiêu chí quan trọng và thoả mãn các câu hỏi gợi ý theo khả năng thực tế của yêu xây dựng chính quyền điện tử của tỉnh An Giang. Mặc dù nghiên cứu chỉ dừng lại ở mức tổng hợp, bổ sung và chưa đề xuất công cụ thực nghiệm để đánh giá trên một số tiêu chí quan trọng như Performance, Stability nhưng với những kết quả bước đầu rất khả quan, chúng tôi tin rằng bộ tiêu chí sẽ hoàn thiện hơn nữa với những số liệu cụ thể bổ sung lý thuyết đã trình bày với kịch bản tích hợp đa dạng.

Bảng 1: Bảng các tiêu chí đánh giá bốn ESB dựa trên FOSS

	Mule ESB	WSO2 ESB	OpenESB	JBOSS FUSE
Usability	Enterprise: ***** Community: *** Bộ công cụ phát triển Anypoint Studio cho phép dùng thử 30 ngày trước khi trả phí.	Đánh giá: ***** Developer Studio tích hợp sẵn một số công cụ và có thể cài đặt riêng các công cụ khác dưới dạng gói thư viện.	Đánh giá: ***** OpenESB Studio có các công cụ cơ bản tuy nhiên khó tích hợp thêm các công cụ bổ sung.	Đánh giá: *** JBoss Developer Studio. Bản miễn phí phải tự build toàn bộ, bản Enterprise phát triển sẵn dùng
Maintainability	Đánh giá: ***** Mule Management Console cung cấp gói chức năng theo nhu cầu sử dụng, trả thêm phí nếu muốn mở rộng.	Đánh giá: ***** Quản trị trên nền web, 100% miễn phí, hệ thống thư viện phong phú, sẵn dùng để mở rộng.	Đánh giá: ***** Quản trị trên nền web gồm một số công cụ giám sát chuyên biệt: NMR monitoring, System Monitoring.	Đánh giá: *** Hỗ trợ bộ giám sát nhật ký web không đầy đủ tính năng, cần xây dựng bổ sung thêm.
Community Support	Enterprise: ***** Community: ***** Bản Community chủ yếu cộng đồng người dùng trợ giúp nhau. Bản Enterprise cung cấp các dịch vụ trợ giúp 24/7 hoặc giải quyết sự cố nhanh.	Đánh giá: ***** Hầu hết miễn phí. Hình thức hỗ trợ qua online form hoặc diễn đàn. Hỗ trợ chat trên diễn đàn giữa các thành viên.	Đánh giá: ***** Bản Community chủ yếu trợ giúp qua diễn đàn. Bản Enterprise thương mại hoá bởi Pymma thì kênh hỗ trợ phong phú, đa dạng.	Đánh giá: *** Khó tìm trợ giúp từ cộng đồng. Gói Subscription có phí trong vòng 1 năm sẽ có hỗ trợ 24/7, giải quyết sự cố, tư vấn xây dựng hệ thống.
Functionality	Đánh giá: ***** Các tính năng quản trị, phân tích phải trả phí.	Đánh giá: ***** Hỗ trợ 100% tính năng quản trị, phân tích.	Đánh giá: ***** Phải tự xây dựng thêm các tính năng quản trị theo quy mô sử dụng	Đánh giá: *** Mức độ tiếp cận để quản trị và phân tích không đơn giản
Flexibility	Đánh giá: ***** Rất mềm dẻo khi tích hợp các dịch vụ tự phát triển theo nhu cầu.	Đánh giá: ***** Có sẵn các dịch vụ miễn phí như: Data Services Server, Enterprise Mobility Manager...	Đánh giá: ***** Mã nguồn được công bố nên dễ tùy chỉnh dịch vụ tích hợp phù hợp với trực.	Đánh giá: ***** Cung cấp nhiều cách thức để xây dựng và triển khai các Endpoint thông qua Message Broker
Expandability	Đánh giá: ***** Khả năng mở rộng lên đến 13,000 Server trong môi trường phân tán lớn.	Đánh giá: ***** Cho phép tích hợp các cluster hoặc triển khai hệ thống theo hình thức nền tảng đám mây	Đánh giá: ***** Chủ yếu sử dụng gói dịch vụ mở rộng do cộng đồng phát triển và chia sẻ.	Đánh giá: *** Kiến trúc dễ mở rộng nhưng đòi nghiệp vụ người triển khai phải cao
Connectors	Đánh giá: ***** Tính năng Auto	Đánh giá: ***** Hỗ trợ SOAP,	Đánh giá: ***** Đầy đủ các chuẩn	Đánh giá: ***** Xây dựng sẵn

	Mule ESB	WSO2 ESB	OpenESB	JBOSS FUSE
	completion trong xây dựng các Adapter	REST, Java API và hầu hết các connectors phổ biến khác: Asana, Amazon, Beetrack...	phổ biến: HTTP, SOAP, FTP, JMS, LDAP, HL7, TCP/IP, Bespoke	dùng 150 Connectors
Cost	Đánh giá: **** Bản Community miễn phí. Tuy nhiên chi phí tăng khi bổ sung công cụ quản trị và công cụ quản lý dịch vụ.	Đánh giá: ***** Miễn phí sử dụng. Chi phí phát triển rất thấp, chủ yếu là chi phí thiết kế giải pháp cho hệ thống.	Đánh giá: ***** Bản Community 100% open source. Tồn chi phí tự phát triển các gói dịch vụ tương ứng.	Đánh giá: **** Mặc dù miễn phí sử dụng nhưng khó phát triển dịch vụ. Chi phí đắt nếu sử dụng bản Enterprise.
Licensing	CPAL license	WSO2 license	CDDL	ASL2.0
Security	Đánh giá: ***** Hỗ trợ bảo mật rất tốt từ Anypoint Enterprise Security, phải trả phí.	Đánh giá: ***** Hơn 20 kịch bản đảm bảo an ninh hệ thống sẵn dùng với các mức độ khác nhau	Đánh giá: **** Duy trì ở mức độ cơ bản. Phải tự thiết lập cơ chế an ninh cho hệ thống.	Đánh giá: **** Các gói an ninh nâng cao thường trả phí để thuê phát triển.
Stability	Đánh giá: ***** Khả năng chịu tải tốt với lượng dữ liệu lớn trong hàng triệu hoặc hàng tỷ giao dịch. Đặc biệt ấn tượng ở bản Enterprise	Đánh giá: ***** Engine sử dụng cho quá trình chuyển đổi thông điệp được tối ưu hoá giúp giảm thời gian delay trong xử lý thông điệp trên trục	Đánh giá: *****	Đánh giá: **** Công nghệ Fuse Fabric giúp quản lý tốt các cluster ứng dụng tích hợp.
Performan-ce	Đánh giá: **** Thời gian trung bình đáp ứng cho giao dịch ổn định. Lượng tài nguyên tiêu thụ thấp.	Đánh giá: ***** Trong nhiều kịch bản kiểm thử đạt hiệu suất cao nhất, đặc biệt khi đánh giá qua chỉ số đầu ra trên đơn vị thời gian.	Đánh giá: *** Trong một số bài kiểm tra, OpenESB có hiệu suất cao nhất nhờ vào việc xử lý chuyển đổi thông điệp nhanh.	Đánh giá: *** Kết quả ổn định, không ấn tượng trong các bài kiểm tra.
Industry Wide Acceptance	Đánh giá: ***** Khách hàng đa dạng, nhiều lĩnh vực: y tế, ngân hàng, giáo dục...	Đánh giá: ***** Đối tác tin cậy trong lĩnh vực tài chính, giao dịch ngân hàng.	Đánh giá: *** Bản miễn phí phù hợp với tổ chức có quy mô nhỏ hoặc vừa.	Đánh giá: *** Chỉ được đánh giá tốt với phiên bản Enterprise.
Overall Score	Đánh giá: ****	Đánh giá: *****	Đánh giá: ***	Đánh giá: ***

TÀI LIỆU THAM KHẢO

David Kusák, 2010. Comparision of Enterprise Application Integration Platforms, Department of Software Engineering, Charles University, Prague.

Falko Menge, 2007. Enterprise Service Bus. In: FrOSCon: Free and Open Source Software Conference, University of Applied Science Bonn-Rhein-Sieg.

JBoss Fuse ESB, <https://developers.redhat.com/products/fuse>

- Ken Vollmer, Mike Gilpin, Sander Rose., 2011. In: The Forrester Wave™: Enterprise Service Bus, Q2 2011. Software AG, Tibco Software, Oracle and Progress Software Lead The Way.
- Marcello Castellano, Politecnico di Bari, Nicola Pastore, Francesco Arcieri, Valerio Summo, Giuliano Bellone de Grecis., 2005. An e-Government Cooperative Framework for Government Agencies. In: Proceedings of the 38th Hawaii International Conference on System Sciences, USA. January 3-6. IEEE 2009: Volume 9.
- Mikael Ahlberg, 2010. Enterprise Service Buses: A comparison Regarding Reliable Message Transfer, KTH Computer Science and Communication, Royal Institute and Technology, Stockholm, Sweden.
- Mules ESB, <https://www.mulesoft.com>
- MuleSoft Case Studies, accessed on 17 September, 2017. Available from <https://www.mulesoft.com/case-studies/soa/university-witwatersrand>
- Ngô Bá Hùng, 2016. Thuyết minh đề tài nghiên cứu ứng dụng và phát triển công nghệ tinh An Giang.
- OpenESB, <http://www.open-esb.net>
- P. de Leusse, P. Periorellis, P. Watson., 2007. Enterprise Service Bus: An overview. In: TECHNICAL REPORT SERIES, No. CS-TR-1037, University of Newcastle upon Tyne, England, 9 pages.
- Robin Singh Bhadoria Narendra S Chaudhari, Geetam Singh Tomar., 2016., The Performance Metric for Enterprise Service Bus (ESB) in SOA System: theoretical underpinnings and empirical illustrations for information processing. In: Informations Systems. Elsevier 2017: Volume 65, 151-171.
- Sanjay P.Ahuja, Amit Patel., 2011. Enterprise Service Bus: A Performance Evaluation. In: Communication and Network, Vol.3, No.3, 133-140.
- Shafreen Anfar, 2014. In: WSO2 ESB Performance Round 7.5, February, 2014.
- Tobias Kruessmann, Arne Koschel, Martin Murphy, Ardrian Trenaman, Irina Astrova., 2009. High Availability: Evaluating Open Source Enterprise Service Bus. In: Vesna Luzar-Stiffler, Iva Jarec, Zoran Bekic. Proceedings of the ITI 2009 31st International Conference on Information Technology Interfaces, Cavtat/Dubrovnik, Croatia, June 22-25. IEEE 2009: 615-620.
- Wilem-Jan Van den Heuvel, Jos van Hillegersberg., 2001. Enterprise Application Integration: Introduction to the Minitrack, System Sciences. In: Proceeding of the 35th Annual Hawaii International Conference on System Sciences (HICSS'02), Maui, HI, USA, January 7-10. IEEE: Volume 8.
- WSO2 Case Studies, accessed on 17 September, 2017. Available from <http://wso2.com/casestudies/ebay-uses-100-open-source-wso2-esb-to-process-more-than-1-billion-transactions-per-day/>
- WSO2 ESB, <http://wso2.com>
- Zakaria I. Saleh, Rand A. Obei dat, Yaser Khamayseh., 2013. A Framework for an E-government Based on Service Oriented Architecture for Jordan. In: I.J. Information Engineering and Electronic Business. MECS: 3, 1-10.