

# <You always have time for Coding>

A Product of Viblo Platform



**Do Ngoc Khanh** @do.ngoc.khanh

Follow

★ 164 👤 4 ✎ 16

Published Jun 26th, 2017 11:08 PM

👁 13.8K 💬 7 🔗 4

## Khái niệm cơ bản về SOAP, REST và cách phân biệt chúng

...

SOAP (Simple Object Access Protocol) và REST (Representational State Transfer) là đáp án cho cùng một câu hỏi: Làm thế nào để truy cập vào Web Services. Sự lựa chọn ban đầu dường như là dễ dàng, những đôi khi sẽ rất khó để cân nhắc chọn loại API nào phù hợp.

**SOAP** là một giao thức truy cập Webservice được sử dụng trong một thời gian và hưởng tất cả các lợi ích của việc sử dụng lâu dài. Được phát triển bởi Microsoft, SOAP không thực sự đơn giản như từ viết tắt của nó. Trong khi đó, **REST** là một chiến binh mới, tìm kiếm để sửa các vấn đề mà SOAP gặp phải và cung cấp các phương thức thực sự đơn giản để truy cập Webservice. Tuy nhiên, đôi khi SOAP thực sự dễ sử dụng hơn, đôi khi REST có vấn đề của riêng nó. Cả hai kỹ thuật đều có vấn đề cần xem xét khi quyết định sử dụng phương thức nào.

Cả **SOAP** và **REST** đều có những điểm tương đồng trên giao thức HTTP, **SOAP** là một tổ hợp các pattern truyền tin nghiêm ngặt hơn **REST**. Quy định trong **SOAP** là rất quan trọng bởi vì nếu thiếu chúng, bạn không thể đạt được bất kỳ level tiêu chuẩn hóa nào. **REST** mang một kiến trúc không yêu cầu quy trình và linh hoạt hơn. Cả SOAP và REST đều dựa trên những quy định đã tồn tại trong một thời gian dài mà mọi người đều đồng ý tuân theo do lợi ích mà nó mang lại

### Tổng quan ngắn gọn về SOAP

SOAP dựa hoàn toàn vào XML để cung cấp các services truyền tin. Microsoft ban đầu phát triển SOAP để thay thế cho các công nghệ cũ hơn không hoạt động tốt trên Internet như Distributed Component Object Model (DCOM) and Common Object Request Broker Architecture (CORBA). Những công nghệ này không thành công vì chúng dựa vào truyền tin nhị phân, cách truyền tin XML mà SOAP sử dụng làm việc tốt hơn qua Internet.

XML được sử dụng để tạo ra các requests và nhận responses trong SOAP, nó có thể trở nên cực kỳ phức tạp. Trong một số ngôn ngữ lập trình, bạn cần phải xây dựng các requests đó bằng tay, điều này trở nên mơ hồ vì SOAP không dung nạp được các lỗi. Tuy nhiên, một vài ngôn ngữ khác có thể sử dụng các phím tắt mà SOAP cung cấp; Có thể giúp bạn giảm bớt effort cần thiết để tạo request và phân tích responses. Thực tế, khi làm việc với .NET, bạn thậm chí chưa bao giờ gặp đến XML.

Một trong những ưu điểm quan trọng nhất của SOAP là tích hợp xử lý lỗi. Nếu có sự cố với requests của bạn, responses chứa thông tin lỗi mà bạn có thể sử dụng để khắc phục sự cố. Bạn có thể không sở hữu các Web service, tính năng này đặc biệt vô cùng quan trọng; Nếu không bạn sẽ thắc mắc tại sao mọi việc lại không thành công. Các báo cáo lỗi thậm chí cung cấp mã code chuẩn để có thể tự động hoá một số nhiệm vụ xử lý lỗi trong mã của bạn.

Một tính năng khác khá thú vị của SOAP là bạn không nhất thiết phải sử dụng nó với giao thức HyperText Transfer Protocol (HTTP). Có một chỉ định rõ ràng để sử dụng SOAP qua Simple Mail Transfer Protocol (SMTP) và không có lý do gì bạn không thể sử dụng nó trên các phương thức khác.

## Tổng quan ngắn gọn về REST

Nhiều developers thấy rằng SOAP cồng kềnh và khó sử dụng. Ví dụ, làm việc với SOAP trong JavaScript có nghĩa là viết một tấn các dòng code để thực hiện các task cực kỳ đơn giản bởi vì bạn phải tạo cấu trúc XML mỗi lần.

REST cung cấp giải pháp thay thế nhẹ hơn. Thay vì sử dụng XML để tạo request, REST dựa vào một URL đơn giản. Trong một số trường hợp, bạn phải cung cấp thông tin bổ sung theo những cách đặc biệt, nhưng hầu hết các Web service sử dụng REST đều dựa hoàn toàn vào việc thu lại các thông tin cần thiết bằng phương pháp URL. REST có thể sử dụng bốn hình thái HTTP 1.1 khác nhau (GET, POST, PUT, và DELETE) để thực hiện các tasks.

Không giống như SOAP, REST không phải sử dụng XML để cung cấp response. Bạn có thể tìm các Web service dựa trên REST đưa ra dữ liệu trong Command Separated Value (CSV), JavaScript Object Notation (JSON) và Really Simple Syndication (RSS). Vấn đề là bạn có thể thu được output bạn cần trong một form dễ phân tích trong ngôn ngữ bạn cần cho ứng dụng của bạn.

## Lựa chọn SOAP hay REST

Trước khi bạn dành hàng giờ đồng hồ để lựa chọn giữa SOAP và REST, hãy xem xét rằng một số Web service hỗ trợ một và một số khác. Trừ khi bạn có kế hoạch tạo ra Web service của riêng bạn, quyết định sử dụng giao thức nào đã được hình thành bởi bạn. Rất ít Web service, chẳng hạn như Amazon, hỗ trợ cả hai. Quyết định của bạn thường tập trung vào Web service nào phù hợp nhất với nhu cầu của bạn chứ không phải là sử dụng giao thức nào.

## So sánh giữa SOAP và REST

SOAP chắc chắn là lựa chọn nặng ký để truy cập Web service. Nó cung cấp những ưu điểm sau đây khi so sánh với REST:

- Ngôn ngữ, nền tảng và phương tiện độc lập (REST yêu cầu sử dụng HTTP)
- Hoạt động tốt trong môi trường doanh nghiệp (REST giao tiếp thẳng trực tiếp)
- Tiêu chuẩn hoá
- Cung cấp khả năng mở rộng đáng kể trước khi xây dựng dưới dạng các tiêu chuẩn WS \*
- Tích hợp xử lý lỗi
- Tự động hóa khi sử dụng với một số sản phẩm ngôn ngữ

Bên cạnh đó, REST dễ sử dụng hơn và linh hoạt hơn. Nó có những lợi thế sau khi so sánh với SOAP:

- Không có tools đắt tiền nào yêu cầu tương tác với Web service
- Smaller learning curve

- Hiệu quả (SOAP sử dụng XML cho tất cả các truyền tải, REST có thể sử dụng định dạng truyền tải ngắn gọn hơn)
- Nhanh (không yêu cầu xử lý rộng rãi) Gần gũi hơn với các công nghệ Web khác trong triết lý design.

Trên đây là một vài khái niệm cơ bản về SOAP và REST cũng như các đặc điểm của từng phương thức. Hi vọng chúng sẽ giúp ích cho các bạn để lựa chọn phương thức phù hợp nhất với ứng dụng của mình.

## Tài liệu tham khảo

<https://blog.smartbear.com/apis/understanding-soap-and-rest-basics/>



### Related

[Làm thế nào để kiểm thử API hiệu quả?](#)

[Hien Vu](#)  
👁 4402   🔒 4   💬 0   ⬆ 1

[Pattern - Microservices Architecture](#)

[Dao Van Tuan](#)  
👁 761   🔒 2   💬 0   ⬆ 1

[NoSQL Database](#)

[Rownak Jahan](#)  
👁 162   🔒 1   💬 0   ⬆ 0

[Ruby background jon framework nào phù hợp nhất với bạn ??](#)

[Hà Văn Đức](#)  
👁 117   🔒 1   💬 0   ⬆ -8

### More from Do Ngoc Khanh

[Tổng quan về Selenium](#)

[Do Ngoc Khanh](#)  
👁 372   🔒 0   💬 0   ⬆ 2

[Tìm hiểu về Wicket Framework \(Phần 1\).](#)

[Do Ngoc Khanh](#)  
👁 84   🔒 0   💬 0   ⬆ 0

[Một vài chia sẻ về sử dụng Map trong Java](#)

[Do Ngoc Khanh](#)  
👁 541   🔒 0   💬 0   ⬆ 1

[Tìm hiểu về Java 8 - Streams API](#)

[Do Ngoc Khanh](#)  
👁 438   🔒 1   💬 2   ⬆ 2

### Comments

🗨 Login to comment

[Dung Le](#) @dungle

Apr 14th, 2018 3:00 PM

Đọc không hiểu gì cả. Chắc người có kinh nghiệm nhiều rồi mới hiểu.

VD: Ngôn ngữ, nền tảng và phương tiện độc lập (REST yêu cầu sử dụng HTTP) => Mình thấy đều dùng ... link gọi mà. @.@ Hoạt động tốt trong môi trường doanh nghiệp (REST giao tiếp thẳng trực tiếp) => Còn gián tiếp là gì? Trong môi trường doanh nghiệp (mình hiểu là mạng nội bộ hay security gì gì để cho nội bộ tập đoàn sử dụng) thì mình cũng gọi link thẳng mà. Tiêu chuẩn hoá => Là sao? Cung cấp khả năng mở rộng đáng kể trước khi xây dựng dưới dạng các tiêu chuẩn WS \* => Mở rộng các tiêu chuẩn WS\* là sao? Tích hợp xử lý lỗi => Là khi gọi, nếu có lỗi thì SOAP báo lỗi luôn hả? Còn REST thì nếu có kết quả thì ra JSON, ko có thì không ra? Tự động hóa khi sử dụng với một số sản phẩm ngôn ngữ

...còn hàng cũng may có một [đồng nghiệp](#)...

0 | Reply Share ...

 [LE MINH DUC](#) @khoiminhtar

May 2nd, 2018 10:05 AM

đọc vài dòng cũng ko hiểu gì lun. haiz

0 | Reply Share ...

 [Dung Le](#) @dungle

Apr 14th, 2018 3:01 PM

Thấy cái link gốc rồi. Bạn tác giả chắc dùng google translate @.@

0 | Reply Share ...

VIBLO

Search Viblo



Sign In/Sign up

 [Nguyen Van Teo](#) @vanteo123

Jul 4th, 2018 12:16 PM

Bài viết thật sự chẳng mang lại hiệu quả học tập gì cả. Không biết tác giả có thật sự hiểu rõ vấn đề trước khi dịch không nữa?

0 | Reply Share ...

 [Quoc Nguyen](#) @quocnb

Jul 4th, 2018 2:40 PM

=)) sau lập cái nick fake thì đầu tư 1 tí, ít ra thì cũng phải Quang Teo chứ =))

0 | Reply Share ...

 [Nguyen Van Teo](#) @vanteo123

Jul 5th, 2018 7:17 AM 

Nick fake? Bộ trang này có kiểm chứng nick của ai fake hay ko à? Nick của bạn là thật vậy nhìn vào có ai biết Quốc Nguyễn là ai ko? =))

0 | Reply Share ...

 [Truong Ba Loc](#) @loctb

Jan 26th, 9:19 PM

Mình đã đọc kha khá bài của Viblo, nhưng bài này mình thật sự thất vọng, đọc xong 1 lần, đọc lại lần nữa vẫn không hiểu là "người dịch" muốn nói gì.

0 | Reply Share ...

RESOURCES

[Posts](#)

[Questions](#)

[Videos](#)

[Discussions](#)

[Tools](#)

[Organizations](#)

[Tags](#)

[Authors](#)

[Recommend System](#)

[Machine Learning](#)

LINKS

 [Facebook](#)

 [GitHub](#)

 [Browser extension](#)

 [Atom plugin](#)

MOBILE APP



© 2019 Viblo. All rights reserved.

[Feedback](#) [Help](#) [FAQs](#) [RSS](#) [Terms](#)

0

