



PROTOCOLE.

# Cisco propose une alternative à Soap

*Une grande partie des protocoles de communication client serveur sont réputés pour être lourds et complexes. Pour s'en affranchir, l'équipementier prépare le sien, développé en open source.*

**LES FAITS**

Dans quelques semaines, Cisco dévoilera Etch, son protocole réseau de communication entre applications. Le constructeur souhaite ainsi imposer un nouveau standard.

**L'ANALYSE**

**S**oap est complexe car sa définition s'est voulue ambitieuse, avec la volonté affichée de répondre à tous les besoins possibles et imaginables. Avec Etch, nous proposons un protocole plus performant et plus simple », avance Pascal Delprat, consultant réseaux en charge de la sécurité chez Cisco. L'équipementier dévoilera cet été les spécifications et la version bêta de ce protocole de communication interapplicatif qu'il souhaite pousser en tant que standard.

Dans un premier temps, Etch sera intégré à la version 2.5 de la plate-forme Cisco Unified Application Environment (CUAE). Il pourra ensuite être exploité pour tout système d'interconnexion d'applications. Le nouveau protocole remplace le fichier WSDL (Web Service

Description Language), dont a besoin Soap, par un fichier IDL (Interface Definition Language) plus simple. Puis une bibliothèque transforme ce dernier en objet C# ou Java, les deux langages pris en charge pour l'instant. D'autres suivront : C, Ruby, Python, etc. Etch sera intégré à Visual Studio et à Eclipse. Le constructeur proposera son protocole Etch en open source, avec pour objectif la mise en place d'un écosystème de développeurs d'applications s'appuyant sur sa bibliothèque.

**Moins gourmand en bande passante**

Aujourd'hui encore, l'architecture Cisco s'appuie sur Soap, qui a pourtant atteint ses limites en termes de performance. Lors d'une demande d'activation de service au niveau d'un serveur, l'enveloppe du message généré par le protocole est volumineuse. L'appel (validation du formatage de la requête, appel de la procédure et réponse) nécessite de la puissance processeur, de la mémoire et de la bande passante réseau. « Etch permettra

2 QUESTIONS À...



**Philippe Mougin,**  
expert senior  
chez Octo Technology

**Pourquoi Soap déçoit-il ?**

« Après dix années d'existence, c'est une technologie toujours immature, lourde, et qui souffre de problèmes structurels majeurs. Soap et ses services web associés ont des spécifications complexes et imprécises, interprétées différemment par les fournisseurs. Ce qui entraîne des problèmes d'interopérabilité. De plus, la manière dont Soap est utilisé produit des systèmes distribués fortement couplés, difficiles à faire évoluer. »

**Existe-t-il un protocole prometteur ?**

« Le recours à HTTP et aux principes Rest conduit souvent à des systèmes plus simples, plus performants, moins couplés, et plus interopérables que Soap. Cette approche suscite de plus en plus d'intérêt dans l'industrie. Même Microsoft, l'un des principaux défenseurs de Soap, commence à intégrer Rest/HTTP dans ses plates-formes. »

*de consommer moins de bande passante réseau, car l'enveloppe des messages est nettement allégée, et moins de ressources CPU », affirme Pascal Delprat. D'après les premières indications fournies par Cisco, Etch ne sera pas très éloigné des Rest et autres Corba. Mais il rassemblera le meilleur de ces protocoles. Du moins, c'est ce que promet Cisco. ■ EDDY DIBAR*

**Les principaux protocoles de communication interapplicatifs**

	SOAP ET TECHNOLOGIES WS*	CORBA (IIOP)	RMI (JRMP)	REST/HTTP
<b>Description</b>	Middleware universel et extensible basé sur l'échange de messages XML.	Communication d'objets distribués multilingages.	Communication d'objets distribués Java.	Utilisation de HTTP et de ses principes architecturaux pour faire communiquer des applications entre elles.
<b>Organisme en charge des spécifications</b>	W3C, Oasis.	OMG.	Sun.	W3C pour HTTP.
<b>Points forts</b>	<ul style="list-style-type: none"> <li>• Support des transactions.</li> <li>• Echanges asynchrones.</li> <li>• Echanges sur internet.</li> </ul>	<ul style="list-style-type: none"> <li>• Maturité.</li> <li>• Performances.</li> <li>• Support de plusieurs langages.</li> </ul>	<ul style="list-style-type: none"> <li>• Maturité.</li> <li>• Performances.</li> <li>• Bonne intégration à Java.</li> </ul>	<ul style="list-style-type: none"> <li>• Support des échanges sur internet.</li> <li>• Couplage plus faible.</li> <li>• Interopérabilité.</li> </ul>
<b>Points faibles</b>	<ul style="list-style-type: none"> <li>• Couplage.</li> <li>• Lourdeur de maniement.</li> <li>• Complexité.</li> <li>• Problèmes d'interopérabilité.</li> </ul>	<ul style="list-style-type: none"> <li>• Peu adapté aux échanges sur internet.</li> <li>• Couplage.</li> <li>• Lourdeur de maniement.</li> </ul>	<ul style="list-style-type: none"> <li>• Peu adapté aux échanges sur internet.</li> <li>• Couplage.</li> <li>• Limité au monde Java.</li> </ul>	<ul style="list-style-type: none"> <li>• Peu d'outillage pour le développement.</li> <li>• Moins performant que Corba et RMI.</li> <li>• Pas de support natif de fonctionnalités comme le transactionnel distribué.</li> </ul>