WHAT ARE WEB SERVICES?

Web services, sometimes called application services, are standardized interfaces that use open standards for transmitting data, business logic and processes across a network or over the Internet. Because many of the standards used by Web Services are already widely implemented, the overall use of Web services is growing in popularity: eXtensible Markup Language (XML) is used to tag the data, Simple Object Access Protocol (SOAP) is used to transfer the data, Web Services Description Language (WSDL) is used to describe the service and Universal Description and Discovery Integration (UDDI) is used for listing what services are available.

Web services is a broad term that represents all the technologies it encompasses. The term Web service does not refer to services offered by an organization on the Web, such as those available on a banking Web site. Although such an organization may offer a service over the Web, it doesn't necessarily include a programmatic interface allowing multiple applications to be integrated – a technical requirement that defines a Web service.

The services or applications interface, not the users. With integration, developers can offer specific functionality to end users by adding the Web service to a graphical user interface (GUI), such as a Web page or an executable program, without having to "reinvent the wheel" and without needing to know anything about the business or complexity of the Web service being used.

You can think of an individual Web service as a piece of software that performs a specific task, or function, and makes that task available by exposing a set of operations that can be performed (known as methods or Web methods) with the task. Additionally, each of the methods exposes a set of variables that can accept data passed into the method. These variables are known as parameters or properties. Together, the properties and methods refer to a Web service's interface.

Example: Company A creates a Web service that provides currency rate functionality, which exposes a method called GetRate. Company B can then pass a parameter called CountryCode into the GetRate method. The GetRate method takes the CountryCode parameter, looks up the appropriate currency rate in a database and returns the rate back to the program that requested it. In this example, which database did Company A use to access the currency rate information? What was the name of the database server? What communication and security mechanisms were used to access the database server? The answer to these questions is, "It doesn't matter."

The key concept of Web services is encapsulation, the complexity of retrieving the actual currency rate is completely self-contained within the company that created the Web service (Company A). Company B knows only that it called a Web service to get a currency rate and it was given.

Web services provide seamless distributed computing across the entire network, as long as both sides know how to use a Web service. Because the application resides on a Web server, it can be called, or invoked, from any other computer on the network by using web protocols. Adherence to widely accepted standards means that a service written on any platform can work seamlessly with a service written on any other platform. Using Internet protocols mitigates security issues, as these issues are typically already resolved at the majority of sites and therefore no additional security risks are presented in using Web services.

Web services also make the notion of software as a service a real possibility. And because Web services provide integration between two systems, software as a service refers to the possibility of not having to install software on workstations or servers, but rather, being able to use it from across the Internet. (From Webopedia.com and Dummies.com).

WHAT IS THE RELATIONSHIP BETWEEN WEB SERVICES AND EDUNIFY?

In simple terms, EdUnify is a shared, neutral community-based registry of Web services that functions as a directory (like a phone directory or card catalog), connecting users — those looking for Web services with providers — those providing Web services. Providers can register or publish their Web services and manage their listings either publicly or privately on EdUnify.

EdUnify does not run a Web service itself and there is no hardware or software to install, manage or backup. EdUnify simply lists the Web service with a description of its purpose and overall operational characteristics. Users therefore access EdUnify, find Web services and then engage the providers directly. In addition to allowing users to search for Web services, users can also rate (or grade) services and monitor EdUnify for new ones.

Registering Web services in EdUnify is simple, straightforward and only takes a few moments. Some details though must be included in the registration process to ensure users understand applicable issues relating to licensing, authentication and the procedures required to engage the service. PESC administers the EdUnify registry and in so doing, offers an on-demand, cloud-based solution that enables a governance framework for service-oriented architecture (SOA).