

# ***SOAP - INTRODUCTION***

SOAP is a simple and lightweight XML-based mechanism for creating structured data packages that can be exchanged between network applications. Our discussion on SOAP is based on the latest SOAP specification, SOAP 1.2, which passes the recommendation criteria of the W3C organization.

The SOAP specification defines the following fundamental components :

- An envelope that defines a framework for describing message structure
- A set of encoding rules for expressing instances of application-defined data types
- A convention for representing remote procedure calls (RPC) and responses
- A set of rules for using SOAP with HTTP
- Message exchange patterns (MEP) such as request “response, one-way, and peer-to-peer conversations

SOAP can be used in combination with a variety of network protocols, such as HTTP, SMTP, BEEP, FTP, and RMI/IIOP. As we have noted from previous implementation of Web services, SOAP is currently being used as the de facto standard for XML messaging including enveloping and exchanging messages.

SOAP provides a simple enveloping mechanism and is proven in being able to work with existing networking services technologies, such as HTTP. SOAP is also very flexible and extensible. It provides capabilities to add-on standards and application-defined extensions. The wide acceptance of SOAP is based on the fact that it builds upon the XML infoset.

The format of a SOAP message is formally defined in SOAP 1.2 Part 1; specifically , the Messaging Framework specification. <sup>[14]</sup> Figure 5.6 illustrates a simple SOAP message structure, as defined by this specification.

As illustrated in Figure, a SOAP message is packaged in a SOAP:Envelope, which consists of zero or more SOAP:Header and exactly one SOAP:Body. The SOAP body and header can be further divided into blocks. While the SOAP body blocks are intended for the final receiver, the header blocks can be interpreted by the SOAP intermediaries. Figure 3.6 illustrates such a message exchange pattern.

Figure 3.5. SOAP message formats.

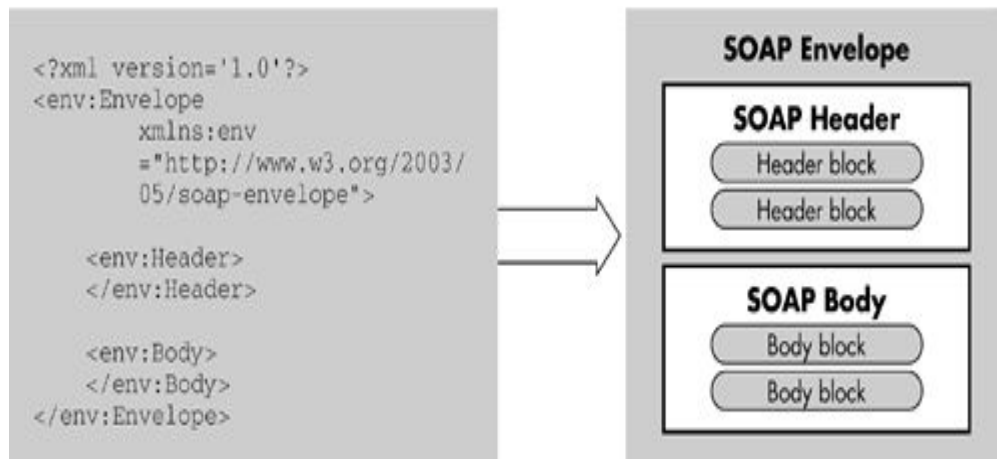
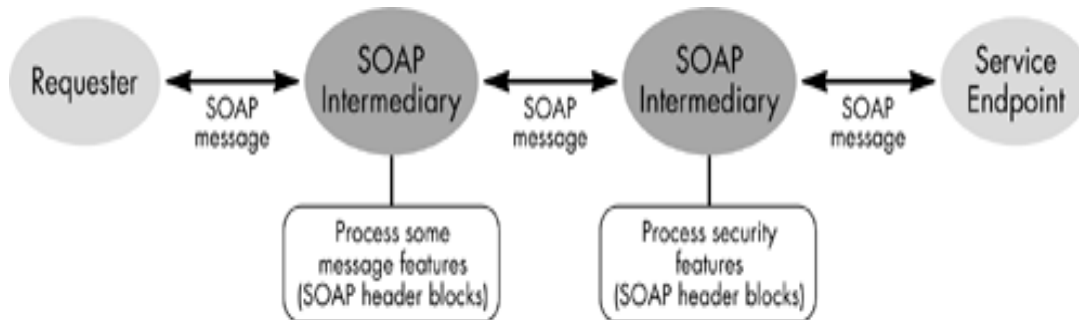


Figure 3.6. SOAP intermediaries.



The SOAP header blocks carry information such as security, transactional information, correlation, and so on. These intermediaries may act on these header blocks, add more blocks, change them, or leave them untouched. These header blocks can be targeted to some intermediary or final receiver by the use of "roles" attributes ("actors" in SOAP 1.1). The value (specified by URI) of the "roles" attribute can be an address or a standard role name as defined by the SOAP specification, which are:

- "next" – each SOAP intermediary and the ultimate receiver must act on this role
- "none" – the SOAP nodes must not act in this role
- "ultimateReceiver" – the ultimate receiver must act in this role

A sample SOAP message with roles is defined in Listing 1.

Listing 1. Sample SOAP message with roles.

```
<?xml version="1.0" ?> <env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-
envelope"> <env:Header> <a:firstheaderBlock xmlns:a="http://ph.com"
```

```
env:role="http://ph.com/example/role">      ...      </a:firstheaderBlock>      <q:secondBlock
xmlns:q="http://ph.com"  env:role="http://www.w3.org/2003/05/soap-envelope/role/next">  ...
</q:secondBlock> </env:Header> <env:Body > ... </env:Body> </env:Envelope>
```

Here <a:firstheaderBlock> is targeted to SOAP nodes who are acting on the role <http://ph.com/example/role>, whereas the <q:secondBlock> must need to be processed by all SOAP nodes in the path , and finally by the ultimate receiver. If no "role" is specified, it is assumed for the ultimate receiver.

We must be cautious in the distinction of these roles, in that we must identify the actor on a message header for the correct processing of the message. There are two other attributes that we must combine with the "role" attribute. They are "mustUnderstand" and "relay." We will note the details in the processing model in the following section.

Thus far, we are discussing topics regarding the format of the message. However, SOAP is much more than simply a message format, it also provides a simple framework for extensible messages and message processing features.

Source : <http://elearningatria.files.wordpress.com/2013/10/ise-viii-grid-computing-06is845-notes.pdf>