

Overview: SOAP Messages

SOAP Messages

When a request is made to a Web service, the Web service automatically generates and sends a response back. There are several ways to handle these communications. For TANGO Web services, these requests and responses are SOAP messages.

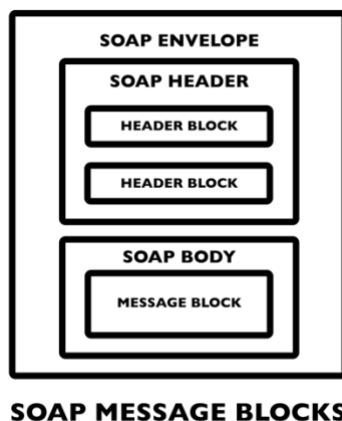
SOAP (Simple Object Access Protocol) defines how Web services communicate. SOAP ensures universal and easy integration for server (TANGO) to client (CMMS) and is best used in decentralized distributed environments.

All SOAP messages are written in XML (Extensible Markup Language). XML uses simple text files designed to be both human and machine-readable. XML is universally accepted like HTML, can easily format server-to-client data, and is an optimal way to store and transport data over the web.

SOAP Architecture

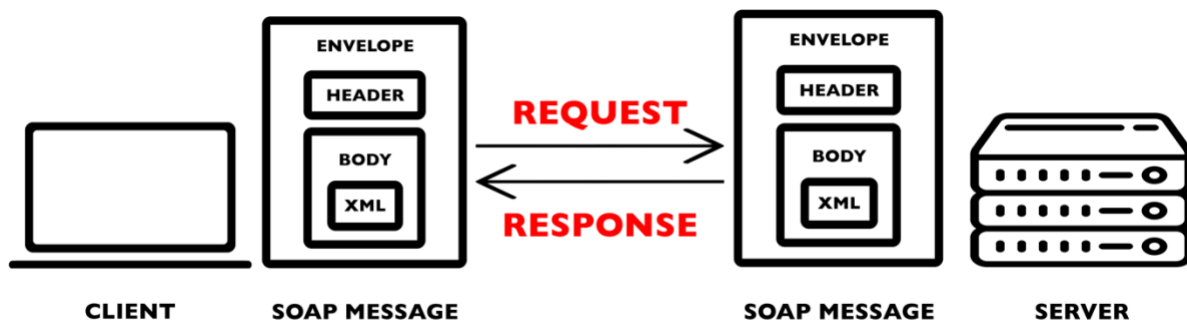
Each SOAP message is constructed with four main elements:

- **Envelope** is a mandatory root element for every SOAP message.
 - Encapsulates all necessary details of the message.
 - Defines the document as a SOAP message.
 - Indicates the start and end of the message.
- **Header** is an optional envelope sub element.
 - Contains message information.
 - Contains authentication credentials.
- **Body** is a mandatory envelope sub element that contains call and response information.
 - Contains the actual data.
- **Fault** is an optional body sub element.
 - Reports message processing errors and status information.



These elements provide:

- **Formatting conventions** represent remote procedure calls (RPC) and responses.
- **Encapsulating envelope** that provides routing directions to ensure the envelope's contents are only sent to and accessible by the desired recipient as well as a description of the contents and how the recipient should process them.
- **Encoding rules** that direct data formatting and translation to and from the XML format.
- **Protocol binding** that allows the envelope to be sent. In our case, via the web (HTTP protocol).



SOAP's most important feature is customization. Every block of a SOAP message may be customized to describe exactly what is needed and how. This flexibility ensures applications that communicate using SOAP, can do so regardless of operating system, technology, language, or transport (request/response) protocol.

SOAP Process

1. User executes a command through their application.
2. The application processes the command and sends it to the application API.
3. The API generates a request message which is translated in an XML format and sent to the XML parser for validation.
4. The requesting XML parser validates the XML document and sends as a SOAP message via HTTP.
5. The receiving XML parser validates the incoming SOAP message.
 - a. Accepted messages are processed further.
 - b. Rejected messages respond with an error message.
6. The accepted SOAP message is then routed to and translated by the application API.
 - a. Identify the parts of the SOAP message which correspond to the application.
 - b. Check that all the mandatory parts are supported by the application or discard the message.
 - c. Remove all the parts before transferring the message if the application is not the endpoint.
7. The application processes the command.
8. The response message follows the same steps (3-7).