

Simple Object Access Protocol

Bartłomiej Świercz

Katedra Mikroelektroniki i Technik Informatycznych

Łódź, 11 grudnia 2005 roku

Czym jest SOAP?

- Akronim SOAP oznacza Simple Object Access Protocol.
- SOAP jest protokołem komunikacyjnym służącym do wymiany wiadomości pomiędzy aplikacjami.
- SOAP przeznaczony jest do wymiany danych w sieci internet.
- SOAP jest niezależny od platformy sprzętowej oraz języka programowania.
- Wiadomości SOAP są opisane za pomocą XML.
- SOAP jest standardem W3C: <http://www.w3.org/TR/soap/>

Dlaczego SOAP?

Komunikacja

Systemy wykorzystujące komunikację rozproszoną w postaci Remote Procedure Call jak CORBA lub DCOM napotykają na problemy bezpieczeństwa i dostępności serwisów w sieci internet. Przeszkodą są licznie wykorzystywane w sieci Internet systemy firewall i proxy.

Powszechność

SOAP jest wykorzystywany przez wiele firm i aplikacji komercyjnych. Jest podstawowym elementem architektury Microsoft .NET dla aplikacji internetowych.

Podstawowym założeniem projektu SOAP była prostota i rozszerzalność projektowanego protokołu. Oznacza to, że wiele elementów występujących w innych protokołach RPC nie pojawia się w SOAP np:

- Rozproszony odśmieczacz pamięci (Distributed garbage collection).
- Przekazywanie obiektów przez referencję (Objects-by-reference).

Protokół SOAP logicznie dzieli się na trzy części:

- Kopertę (SOAP envelope) - jest to definicja opisująca co znajduje się w wiadomości, dla kogo jest przeznaczona i czy wiadomość jest obowiązkowa lub opcjonalna.
- Zasady kodowania (SOAP encoding rules) - możliwość definiowania typów wykorzystywanych przez aplikację.
- Reprezentacja RPC (SOAP RPC representation) - określa reguły opisu zdalnych procedur i ich odpowiedzi.

Wiadomość jest podstawową komunikacją jednokierunkową pomiędzy nadawcą, a odbiorcą. Wiadomość SOAP może wykorzystać różne protokoły jako medium transportowe (np. HTTP).

Aplikacja SOAP, która odebrała wiadomość SOAP musi ją przetworzyć zgodnie z zasadami:

- Rozpoznać wszystkie części wiadomości.
- Zdecydować, czy wszystkie części wiadomości są wspierane przez aplikację.
- Jeżeli aplikacja nie jest adresatem wiadomości to musi ją przekazać do adresata.

Przykładowa wiadomość SOAP

Wiadomość SOAP osadzona w HTTP - rządanie

```
POST /StockQuote HTTP/1.1
Host: www.stockquoteserver.com
Content-Type: text/xml; charset="utf-8"
Content-Length: nnnn
SOAPAction: "Some-URI"
```

```
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <SOAP-ENV:Body>
    <m:GetLastTradePrice xmlns:m="Some-URI">
      <symbol>DIS</symbol>
    </m:GetLastTradePrice>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Przykładowa wiadomość SOAP

Wiadomość SOAP osadzona w HTTP - odpowiedź

HTTP/1.1 200 OK

Content-Type: text/xml; charset="utf-8"

Content-Length: nnnn

```
<SOAP-ENV:Envelope
```

```
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
```

```
  SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
```

```
    <SOAP-ENV:Body>
```

```
      <m:GetLastTradePriceResponse xmlns:m="Some-URI">
```

```
        <Price>34.5</Price>
```

```
      </m:GetLastTradePriceResponse>
```

```
    </SOAP-ENV:Body>
```

```
</SOAP-ENV:Envelope>
```


Wiadomość SOAP jest dokumentem XML zawierającym elementy:

- Wymagane elementy koperty, które identyfikują dokument XML jako wiadomość SOAP.
- Opcjonalny nagłówek.
- Wymagany element Body, który zawiera informacje żądającego lub odpowiadającego na wiadomość.
- Opcjonalny element Fault opisujący błędy mogące pojawić się podczas przetwarzania wiadomości.

Składnia wiadomości SOAP

- Wiadomość SOAP musi być zapisana przy użyciu XML.
- Wiadomość SOAP musi zawierać przestrzeń nazw SOAP Envelope.
- Wiadomość SOAP musi zawierać przestrzeń nazw SOAP Encoding.
- Wiadomość SOAP nie może zawierać definicji DTD.
- Wiadomość SOAP nie może zawierać instrukcji przetwarzania XML.

Szkielet wiadomości SOAP

```
<?xml version="1.0"?>
<soap:Envelope
xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
  <soap:Header>
    :
  </soap:Header>
  <soap:Body>
    :
    <soap:Fault>
      :
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```

Koperta wiadomości SOAP jest jednocześnie głównym (root) elementem dokumentu XML. Definiuje dokument XML jako wiadomość SOAP.

Struktura koperty

```
<?xml version="1.0"?>
<soap:Envelope
xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
  : :   Miejsce na wiadomość.
</soap:Envelope>
```

Koperta zawsze musi być skojarzona z przestrzenią nazw `http://www.w3.org/2001/12/soap-envelope`.

Atrybut `encodingStyle` jest używany do zdefiniowania typów używanych w wiadomości. Wiadomość SOAP nie ma domyślnego kodowania.

Opcjonalny element Header zawiera informacje specyficzne dla aplikacji (autoryzacja, transakcje itd.). Jeżeli element Header jest obecny w wiadomości to musi być pierwszym potomkiem elementu Envelope.

Element Body jest wymagany w każdej wiadomości SOAP. Elementy potomne elementu Body mogą być umieszczone w innej przestrzeni nazw. Specyfikacja SOAP definiuje tylko jeden element potomny umieszczony w domyślnej przestrzeni nazw. Jest to element Fault.

Przykład zapytania

```
<?xml version="1.0"?>

<soap:Envelope
xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">

  <soap:Body>
    <m:GetPrice
      xmlns:m="http://www.w3schools.com/prices">
      <m:Item>Apples</m:Item>
    </m:GetPrice>
  </soap:Body>

</soap:Envelope>
```


Przykład odpowiedzi

```
<?xml version="1.0"?>

<soap:Envelope
  xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
  soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">

  <soap:Body>
    <m:GetPriceResponse
      xmlns:m="http://www.w3schools.com/prices">
      <m:Price>1.90</m:Price>
    </m:GetPriceResponse>
  </soap:Body>

</soap:Envelope>
```

Element Fault

Informacja o błędzie dotyczącym wiadomości SOAP przekazywana jest wewnątrz elementu Fault. Element Fault musi być potomkiem elementu Body i może wystąpić tylko raz w dokumencie.

Przykład:

```
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SOAP-ENV:Fault>
      <faultcode>
        SOAP-ENV:MustUnderstand
      </faultcode>
      <faultstring>
        SOAP Must Understand Error
      </faultstring>
    </SOAP-ENV:Fault>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Element Fault

Informacja o błędzie dotyczącym wiadomości SOAP przekazywana jest wewnątrz elementu Fault. Element Fault musi być potomkiem elementu Body i może wystąpić tylko raz w dokumencie.

Przykład:

```
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SOAP-ENV:Fault>
      <faultcode>
        SOAP-ENV:MustUnderstand
      </faultcode>
      <faultstring>
        SOAP Must Understand Error
      </faultstring>
    </SOAP-ENV:Fault>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Elementy potomne elementu Fault

Element Fault może mieć zagnieżdżone elementy:

Element:	Opis:
<faultcode>	Kod identyfikujący błąd.
<faultstring>	Czytelny opis kodu.
<faultactor>	Informacja o tym co wywołało błąd.
<detail>	Dodatkowe informacje o błędzie.

Kod:	Opis:
VersionMismatch	Nieprawidłowa przestrzeń nazw dla elementu Envelope.
MustUnderstand	Odbiorca wiadomości nie zrozumiał elementu, który miał ustawiony atrybut mustUnderstand.
Client	Wiadomość została nieprawidłowo zbudowana lub zawiera niepoprawne dane.
Server	Wystąpił błąd podczas pracy serwera.