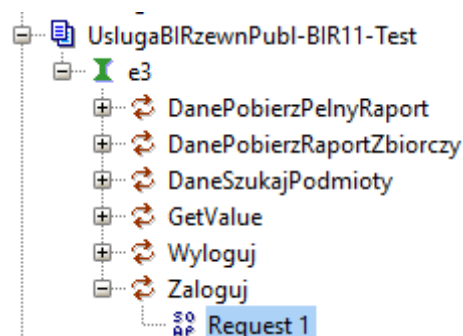


Informacje wstępne

- Usługa sieciowa BIR1 / 1.1 / 1.2 obsługuje żądania wykorzystujące protokół SOAP 1.2 i zwraca dane w formacie xml, zgodnie ze standardem MTOM. Inny protokół komunikacyjny (np. REST) czy format danych nie jest obsługiwany.
- Kodowanie znaków: charset=utf-8
- Content-Type zapytania (request) Usługi BIR1: application/**soap**+xml;
- Content-Type odpowiedzi (response) Usługi BIR1: application/**xop**+xml;
- O godz. 3:20 (w nocy) na serwerze www ma miejsce operacja „czyszczenia puli” (zalecenie producenta), co powoduje, że wszystkie aktywne o tej godzinie sesje http zostają zamknięte.
- Certyfikat publiczny domeny stat.gov.pl znajduje się na stronie <https://pki.stat.gov.pl> (plik https://pki.stat.gov.pl/stat_gov_pl.cer)

A) Sprawdzenie działania Usługi - kroki dla uzyskania danych z wykorzystaniem narzędzia SoapUI (przydatne podczas diagnostyki tworzonej aplikacji/ systemu)

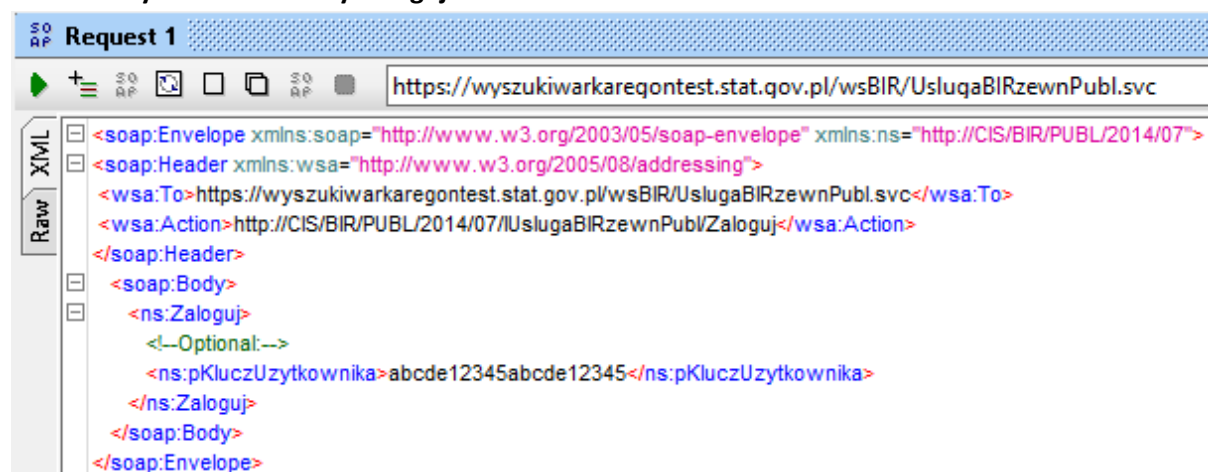
Metody usługi sieciowej (webservice'u) BIR1.1:



Uwaga

Narzędzie **SoapUI nie tworzy nagłówków** <soap:Header> automatycznie. (Pełne koperty SOAP dla BIR1.1– pkt 2).

Krok 1 – wywołanie metody Zaloguj

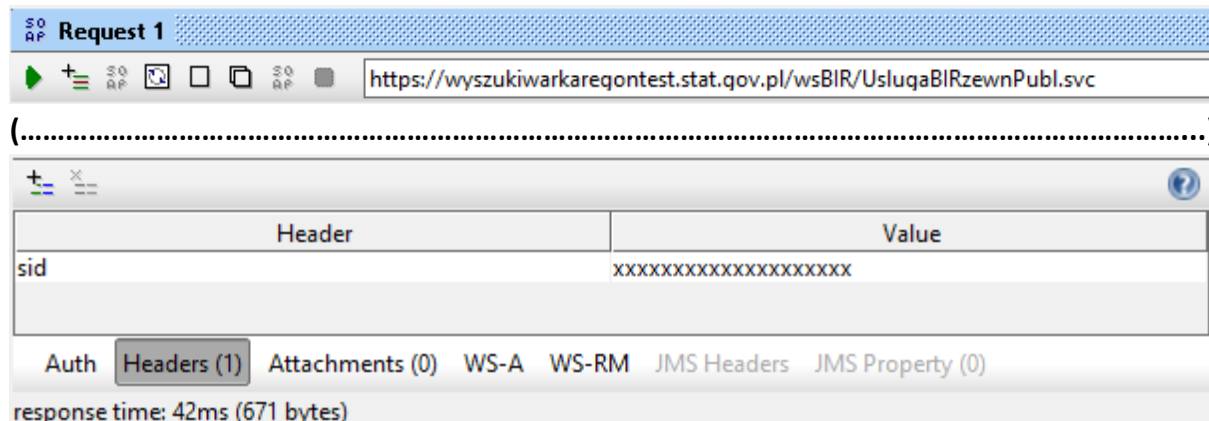


Odpowiedź:



Krok 2 - wywołanie metody DaneSzukajPodmioty

Krok 2.1 Dodanie uzyskanego w kroku 1 identyfikatora sesji jako header http o nazwie sid:



Request 1

https://wyszukiwarkaregontest.stat.gov.pl/wsBIR/UslugaBIRzewnPubl.svc

Header	Value
sid	XXXXXXXXXXXXXXXXXXXX

Auth Headers (1) Attachments (0) WS-A WS-RM JMS Headers JMS Property (0)

response time: 42ms (671 bytes)

Krok 2.2 Wywołanie metody DaneSzukajPodmioty



Request 1

https://wyszukiwarkaregontest.stat.gov.pl/wsBIR/UslugaBIRzewnPubl.svc

```
<?xml version='1.0' encoding='utf-8'>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope" xmlns:ns="http://CIS/BIR/PUBL/2014/07"
  xmlns:dat="http://CIS/BIR/PUBL/2014/07/DataContract">
  <soap:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
    <wsa:Action>http://CIS/BIR/PUBL/2014/07/UslugaBIRzewnPubl/DaneSzukajPodmioty</wsa:Action>
    <wsa:To>https://wyszukiwarkaregontest.stat.gov.pl/wsBIR/UslugaBIRzewnPubl.svc</wsa:To>
  </soap:Header>
  <soap:Body>
    <ns:DaneSzukajPodmioty>
      <ns:pParametryWyszukiwania>
        <dat:Nip>5261040828</dat:Nip>
      </ns:pParametryWyszukiwania>
    </ns:DaneSzukajPodmioty>
  </soap:Body>
</soap:Envelope>
```

Odpowiedź:



```
<?xml version='1.0' encoding='utf-8'>
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope" xmlns:a="http://www.w3.org/2005/08/addressing">
  <s:Header>
    <a:Action s:mustUnderstand="1">http://CIS/BIR/PUBL/2014/07/UslugaBIRzewnPubl/DaneSzukajPodmiotyResponse</a:Action>
  </s:Header>
  <s:Body>
    <DaneSzukajPodmiotyResponse xmlns="http://CIS/BIR/PUBL/2014/07">
      <DaneSzukajPodmiotyResult><![CDATA[<root>
<dane>
  <Regon>000331501</Regon>
  <Nip>5261040828</Nip>
  <StatusNip />
  <Nazwa>GŁÓWNY URZĄD STATYSTYCZNY</Nazwa>
  <Wojewodztwo>MAZOWIECKIE</Wojewodztwo>
  <Powiat>m. st. Warszawa</Powiat>
  <Gmina>Śródmieście</Gmina>
  <Miejscowosc>Warszawa</Miejscowosc>
</dane>
</root>
</CDATA>
</DaneSzukajPodmiotyResult>
</DaneSzukajPodmiotyResponse>
</s:Body>
</s:Envelope>
```

Krok 3.1 - dodanie sid i 3.2 - wywołanie metody DanePobierzPelnyRaport

Request 1

https://wyszukiwarkaregontest.stat.gov.pl/wsBIR/UsługaBIRzewnPubl.svc

Raw XML

```
<?xml version='1.0' encoding='utf-8'>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope" xmlns:ns="http://CIS/BIR/PUBL/2014/07">
  <soap:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
    <wsa:Action>http://CIS/BIR/PUBL/2014/07/UsługaBIRzewnPubl/DanePobierzPelnyRaport</wsa:Action>
    <wsa:To>https://wyszukiwarkaregontest.stat.gov.pl/wsBIR/UsługaBIRzewnPubl.svc</wsa:To>
  </soap:Header>
  <soap:Body>
    <ns:DanePobierzPelnyRaport>
      <!--Optional:-->
      <ns:pRegon>000331501</ns:pRegon>
      <ns:pNazwaRaportu>BIR110sPrawna</ns:pNazwaRaportu>
    </ns:DanePobierzPelnyRaport>
  </soap:Body>
</soap:Envelope>
```

Header	Value
sid	XXXXXXXXXXXXXXXXXXXX

Odpowiedź

Raw XML

```
<?xml version='1.0' encoding='utf-8'>
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope" xmlns:a="http://www.w3.org/2005/08/addressing">
  <s:Header>
    <a:Action s:mustUnderstand="1">http://CIS/BIR/PUBL/2014/07/UsługaBIRzewnPubl/DanePobierzPelnyRaportResponse</a:Action>
  </s:Header>
  <s:Body>
    <DanePobierzPelnyRaportResponse xmlns="http://CIS/BIR/PUBL/2014/07">
      <DanePobierzPelnyRaportResult><![CDATA[<root>
        <dane>
          <praw_regon9>000331501</praw_regon9>
          <praw_nip>5261040828</praw_nip>
          <praw_statusNip />
          <praw_nazwa>GŁÓWNY URZĄD STATYSTYCZNY</praw_nazwa>
          <praw_nazwaSkrócona>GUS</praw_nazwaSkrócona>
          <praw_numerWRejestrzeEwidencji />
          <praw_dataWpisuDoRejestruEwidencji>1975-12-15</praw_dataWpisuDoRejestruEwidencji>
          <praw_dataPowstania>1975-12-15</praw_dataPowstania>
          <praw_dataRozpoczęciaDziałalności>1975-12-15</praw_dataRozpoczęciaDziałalności>
          <praw_dataWpisuDoRegon />
          <praw_dataZawieszeniaDziałalności />
          <praw_dataWznowieniaDziałalności />
          <praw_dataZaistnieniaZmiany>2009-02-20</praw_dataZaistnieniaZmiany>
          <praw_dataZakończeniaDziałalności />
          <praw_dataSkreśleniaZRegon />
          <praw_dataOrzeczeniaOUpadłości />
          <praw_dataZakończeniaPostępowaniaUpadłościowego />
          <praw_adSiedzKraj_Symbol>PL</praw_adSiedzKraj_Symbol>
          <praw_adSiedzWojewodztwo_Symbol>14</praw_adSiedzWojewodztwo_Symbol>
          <praw_adSiedzPowiat_Symbol>65</praw_adSiedzPowiat_Symbol>
          <praw_adSiedzGmina_Symbol>108</praw_adSiedzGmina_Symbol>
          <praw_adSiedzKodPocztowy>00925</praw_adSiedzKodPocztowy>
          <praw_adSiedzMiejscowośćPoczty_Symbol>0919810</praw_adSiedzMiejscowośćPoczty_Symbol>
          <praw_adSiedzMiejscowość_Symbol>0919810</praw_adSiedzMiejscowość_Symbol>
          <praw_adSiedzUlica_Symbol>10013</praw_adSiedzUlica_Symbol>
          <praw_adSiedzNumerNieruchomości>208</praw_adSiedzNumerNieruchomości>
        </dane>
      </DanePobierzPelnyRaportResult>
    </DanePobierzPelnyRaportResponse>
  </s:Body>
</s:Envelope>
```

B) Przykłady prawidłowych kopert SOAP dla wywołania metod usługi BIR1.1

Uwaga

Narzędzie **SoapUI nie tworzy nagłówków** <soap:Header> automatycznie.

Korzystając z SoapUI należy nagłówki koperty SOAP dodać „ręcznie” korzystając z poniższych przykładów.

<!-- Metody usługi sieciowej BIR1.1 -->

<!-- Zaloguj ----->

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:ns="http://CIS/BIR/PUBL/2014/07">
  <soap:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
    <wsa:To>https://wyszukiwarkaregontest.stat.gov.pl/wsBIR/UslugaBIRzewnPubl.svc</wsa:To>
    <wsa:Action>http://CIS/BIR/PUBL/2014/07/IUslugaBIRzewnPubl/Zaloguj</wsa:Action>
  </soap:Header>
  <soap:Body>
    <ns:Zaloguj>
      <ns:pKluczUzytkownika>theUsersKey</ns:pKluczUzytkownika>
    </ns:Zaloguj>
  </soap:Body>
</soap:Envelope>
```

<!-- DaneSzukajPodmioty ----->

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:ns="http://CIS/BIR/PUBL/2014/07" xmlns:dat="http://CIS/BIR/PUBL/2014/07/DataContract">
  <soap:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
    <wsa:To>https://wyszukiwarkaregontest.stat.gov.pl/wsBIR/UslugaBIRzewnPubl.svc</wsa:To>
    <wsa:Action>http://CIS/BIR/PUBL/2014/07/IUslugaBIRzewnPubl/DaneSzukajPodmioty</wsa:Action>
  </soap:Header>
  <soap:Body>
    <ns:DaneSzukajPodmioty>
      <ns:pParametryWyszukiwania>
        <!--Optional:-->
        <dat:Regon>000331501</dat:Regon>
        <!--Optional:-->
        <dat:Nip></dat:Nip>
        <!--Optional:-->
        <dat:Krs></dat:Krs>
        <!--Optional:-->
        <dat:Nipy></dat:Nipy>
        <!--Optional:-->
        <dat:Regony9zn></dat:Regony9zn>
        <!--Optional:-->
        <dat:Krsy></dat:Krsy>
        <!--Optional:-->
        <dat:Regony14zn></dat:Regony14zn>
      </ns:pParametryWyszukiwania>
    </ns:DaneSzukajPodmioty>
  </soap:Body>
</soap:Envelope>
```

```
<!-- DanePobierzPelnyRaport ----->
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:ns="http://CIS/BIR/PUBL/2014/07">
<soap:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
<wsa:To>https://wyszukiwarkaregontest.stat.gov.pl/wsBIR/UslugaBIRzewnPubl.svc</wsa:To>
<wsa:Action>http://CIS/BIR/PUBL/2014/07/IUslugaBIRzewnPubl/DanePobierzPelnyRaport
</wsa:Action>
</soap:Header>
<soap:Body>
<ns:DanePobierzPelnyRaport>
<ns:pRegon>000331501</ns:pRegon>
<ns:pNazwaRaportu>BIR11OsPrawna</ns:pNazwaRaportu>
</ns:DanePobierzPelnyRaport>
</soap:Body>
</soap:Envelope>
```

```
<!-- DanePobierzRaportZbiorczy ----->
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:ns="http://CIS/BIR/PUBL/2014/07">
<soap:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
<wsa:Action>http://CIS/BIR/PUBL/2014/07/IUslugaBIRzewnPubl/DanePobierzRaportZbiorczy
</wsa:Action>
<wsa:To>https://wyszukiwarkaregontest.stat.gov.pl/wsBIR/UslugaBIRzewnPubl.svc</wsa:To>
</soap:Header>
<soap:Body>
<ns:DanePobierzRaportZbiorczy>
<ns:pDataRaportu>2014-11-07</ns:pDataRaportu>
<ns:pNazwaRaportu> BIR11NowePodmiotyPrawneOrazDzialalnosciOsFizycznych
</ns:pNazwaRaportu>
</ns:DanePobierzRaportZbiorczy>
</soap:Body>
</soap:Envelope>
```

```
<!-- GetValue ----->
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:ns="http://CIS/BIR/2014/07">
<soap:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
<wsa:To>https://wyszukiwarkaregontest.stat.gov.pl/wsBIR/UslugaBIRzewnPubl.svc</wsa:To>
<wsa:Action>http://CIS/BIR/2014/07/IUslugaBIR/GetValue</wsa:Action>
</soap:Header>
<soap:Body>
<ns:GetValue>
<ns:pNazwaParametru>StatusSesji</ns:pNazwaParametru>
</ns:GetValue>
</soap:Body>
</soap:Envelope>
```

```
<!-- Wyloguj ----->
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:ns="http://CIS/BIR/PUBL/2014/07">
<soap:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
<wsa:To>https://wyszukiwarkaregontest.stat.gov.pl/wsBIR/UslugaBIRzewnPubl.svc</wsa:To>
<wsa:Action>http://CIS/BIR/PUBL/2014/07/IUslugaBIRzewnPubl/Wyloguj</wsa:Action>
</soap:Header>
<soap:Body>
<ns:Wyloguj>
<ns:pIdentyfikatorSesji>xxxxxxxxxxxxxxxxxxxxxx</ns:pIdentyfikatorSesji>
</ns:Wyloguj>
</soap:Body>
</soap:Envelope>
```

C) Sekwencja wywołania metod usługi BIR1.1

Poniżej przedstawiono pseudokod dla przykładowej prawidłowej sekwencji pozwalającej na uzyskanie danych o podmiotach z rejestru REGON.

```
sid=Zaloguj();
do {
    DaneSzukajPodmioty (regon lub nip lub krs);
    i/lub DanePobierzPelnyRaport (regon, nazwaRaportu);
    i/lub DanePobierzRaportZbiorczy (data, nazwaRaportu);
    if(pusta odpowiedz) => sprawdź przyczynę metodą diagnostyczną GetValue();
    // Szczegóły nt. metody GetValue – „Instrukcja techniczna” str. 11.
    //najczęstsza przyczyna pustej odpowiedzi - wygasła sesja http
    if (sesja wygasła) => sid=Zaloguj();
} while(...);
Wyloguj(sid);
```

Uwaga: **w celu uzyskania adresu siedziby** wystarczy wywołanie tylko metody DaneSzukajPodmioty (inaczej niż w wersji 1.0)

Pusta odpowiedz to:

<DaneSzukajPodmiotyResult/>

lub <DanePobierzPelnyRaportResult/>

lub <DanePobierzRaportZbiorczyResult/>

Warunkiem jest podawanie w w/w pętli jako http header sid uzyskanego w Zaloguj

D) Często zadawane pytania dot. problemów

Pytanie

- Utworzyłem projekt w SoapUI korzystając z podanego adresu wsdl, lecz otrzymuję komunikat „DestinationUnreachable” oraz
<< The message with To " cannot be processed at the receiver, due to an AddressFilter mismatch at the EndpointDispatcher. Check that the sender and receiver's EndpointAddresses agree >>

Odpowiedź

- SoapUI nie dodaje nagłówków dla kopert SOAP (<soap:Header>). Szczegóły – plik BIR11_Przykłady, punkt B) „Przykłady prawidłowych kopert SOAP dla wywołania metod usługi BIR1.1”.

Pytanie

- Otrzymuję „Błąd 500 - Internal Server Error”.

Odpowiedź

- Prawdopodobnie nieprawidłowa jest treść koperty SOAP (lub całego żądania http).
Proszę przechwycić kopertę SOAP i porównać z przykładami z pliku BIR11_Przykłady.

Pytanie

- Otrzymuję komunikat:
“ Cannot process the message because the content type 'text/xml; charset=utf-8' was not the expected type 'multipart/related; type="application/xop+xml" “.

Odpowiedź

- Prawdopodobną przyczyną opisanego problemu jest sytuacja, że aplikacja oczekuje w zwracanej odpowiedzi - Content-Type:text/xml, a Content-Type odpowiedzi (response) Usługi BIR1 to: Content-Type: multipart/related; type="application/xop+xml"

Pytanie

- Otrzymuję odpowiedź: "Bad Request".

Odpowiedź

- Proszę sprawdzić czy Content-Type dla request i response jest prawidłowy. (Patrz str.1).

E) Konfiguracja dla środowiska .NET (Proszę zwrócić uwagę na ustawienie messageEncoding !)

```
<system.serviceModel>
  <bindings>
    <wsHttpBinding>
      <binding name="e3" messageEncoding="Mtom">
        <security mode="Transport">
          <transport clientCredentialType="None" />
        </security>
      </binding>
    </wsHttpBinding>
  </bindings>

  <client>
    <!-- Wersja testowa -->
    <endpoint address="https://wyszukiwarkaregontest.stat.gov.pl/wsBIR/UslugaBIRzewnPubl.svc"
      binding="wsHttpBinding" bindingConfiguration="e3"
      contract="ServiceRegon.IUslugaBIRzewnPubl" name="e3" />

    <!-- Wersja produkcyjna
    <endpoint address="https://wyszukiwarkaregon.stat.gov.pl/wsBIR/UslugaBIRzewnPubl.svc"
      binding="wsHttpBinding" bindingConfiguration="e3"
      contract="ServiceRegon.IUslugaBIRzewnPubl" name="e3" /> -->
  </client>
</system.serviceModel>
```

Przykłady kodu

Uwaga:

UsługaBIRpubl w poniższych przykładach to tzw. ServiceReference, który należy utworzyć samodzielnie

(w Visual Studio: "Add Service Reference") wykorzystując adres wsdl Usługi -

dla wersji testowej: <https://wyszukiwarkaregontest.stat.gov.pl/wsBIR/wsdl/UsługaBIRzewnPubl-ver11-test.wsdl> ;

dla wersji produkcyjnej: <https://wyszukiwarkaregon.stat.gov.pl/wsBIR/wsdl/UsługaBIRzewnPubl-ver11-prod.wsdl> .

1a) Przykład dot. VB.NET

```
Sub PolaczBIRpubl()

    'Create the binding.
    Dim myBinding = New WSHttpBinding
    myBinding.Security.Mode = SecurityMode.Transport
    myBinding.Security.Transport.ClientCredentialType =
HttpClientCredentialType.None
    myBinding.MessageEncoding = WSMessageEncoding.Mtom

    Dim ea As New
EndpointAddress("https://wyszukiwarkaregontest.stat.gov.pl/wsBIR/UsługaBIRzewnPubl.
svc") 'test

    Dim cc As New UsługaBIRpubl.UsługaBIRzewnPublClient(myBinding, ea)
    cc.Open()

    'ZALOGUJ (bez sida)
    Dim strSID As String = cc.Zaloguj("....klucz....")

    txtResult.Text &= "### Zaloguj:" & vbCrLf & strSID & vbCrLf & vbCrLf

    Using (New OperationContextScope(cc.InnerChannel))

        Dim requestMessage As Channels.HttpRequestMessageProperty = New
Channels.HttpRequestMessageProperty
        requestMessage.Headers("sid") = strSID

        OperationContext.Current.OutgoingMessageProperties(Channels.HttpRequestMessagePrope
rty.Name) = requestMessage

        'DANE SZUKAJ 1
        Dim objParametryGR1 As New UsługaBIRpubl.ParametryWyszukiwania
        objParametryGR1.Nip = "1111111111"
        txtResult.Text &= "### DaneSzukaj (grupa 1):" & vbCrLf &
cc.DaneSzukaj(objParametryGR1) & vbCrLf & vbCrLf

        ''DANE POBIERZ PEŁNY RAPORT
        txtResult.Text &= "### DanePobierzPelnyRaport:" & vbCrLf &
cc.DanePobierzPelnyRaport("39002176400000", "PublDaneRaportPrawna", "") & vbCrLf &
vbCrLf

        'WYLOGUJ
        txtResult.Text &= "### Wyloguj:" & vbCrLf & cc.Wyloguj(strSID) & vbCrLf &
vbCrLf

        'GET VALUE
        txtResult.Text &= "### GetValue(KomunikatKod):" & vbCrLf &
cc.GetValue("KomunikatKod") & vbCrLf
        txtResult.Text &= "### GetValue(KomunikatTresc):" & vbCrLf &
cc.GetValue("KomunikatTresc") & vbCrLf
        txtResult.Text &= "### GetValue(StatusSesji):" & vbCrLf &
cc.GetValue("StatusSesji") & vbCrLf

    End Using

    cc.Close()

End Sub
```

1b) Przykład dot. VB.NET

```
Sub SOAPpubl()

    'Create the binding.
    Dim myBinding = New WSHttpBinding
    myBinding.Security.Mode = SecurityMode.Transport
    myBinding.MessageEncoding = WSMessagesEncoding.Mtom
    myBinding.Security.Transport.ClientCredentialType =
HttpClientCredentialType.None

    Dim ea As New EndpointAddress(
"https://wyszukiwarkaregontest.stat.gov.pl/wsBIR/UslugaBIRzewnPubl.svc") 'test

    'Create the client.
    Dim cc As New UslugaBIRpubl.UslugaBIRzewnPublClient(myBinding, ea)

    Dim manualWebClient As New System.Net.WebClient()

    manualWebClient.Headers.Add("Content-Type", "application/soap+xml; charset=utf-
8")

    Dim strSOAP As String = String.Empty
    strSOAP &= "<soap:Envelope xmlns:soap=""http://www.w3.org/2003/05/soap-
envelope"" xmlns:ns=""http://CIS/BIR/PUBL/2014/07"">" & System.Environment.NewLine
    strSOAP &= "<soap:Header xmlns:wsa=""http://www.w3.org/2005/08/addressing"">" &
System.Environment.NewLine
    strSOAP &=
"<wsa:To>https://wyszukiwarkaregontest.stat.gov.pl/wsBIR/UslugaBIRzewnPubl.svc</wsa
:To>" & System.Environment.NewLine
    strSOAP &=
"<wsa:Action>http://CIS/BIR/PUBL/2014/07/IUslugaBIRzewnPubl/Zaloguj</wsa:Action></s
oap:Header>" & System.Environment.NewLine
    strSOAP &= "    <soap:Body>" & System.Environment.NewLine
    strSOAP &= "        <ns:Zaloguj>" & System.Environment.NewLine
    strSOAP &= "            <ns:pKluczUzytkownika>...klucz...</ns:pKluczUzytkownika>"
& System.Environment.NewLine
    strSOAP &= "        </ns:Zaloguj>" & System.Environment.NewLine
    strSOAP &= "    </soap:Body>" & System.Environment.NewLine
    strSOAP &= "</soap:Envelope>" & System.Environment.NewLine

    Dim bytArguments As Byte() = System.Text.Encoding.ASCII.GetBytes(strSOAP)
    Dim bytRetData As Byte() =
manualWebClient.UploadData("https://wyszukiwarkaregontest.stat.gov.pl/wsBIR/UslugaB
IRzewnPubl.svc", "POST", bytArguments)

    txtResult.Text = System.Text.Encoding.ASCII.GetString(bytRetData)

End Sub
```

2) Przekazanie wartości sid jako dodatkowego nagłówka żądania http (język C#).

```
string sid = client.Zaloguj(klucz);

OperationContextScope scope = new OperationContextScope(client.InnerChannel);

HttpRequestMessageProperty reqProps = new HttpRequestMessageProperty();
reqProps.Headers.Add("sid", sid);
OperationContext.Current.OutgoingMessageProperties[HttpRequestMessageProperty.Name] = reqProps;
```

Poniżej zamieszczamy informacje otrzymane od Usługobiorców i za ich zgodą dot. implementacji w innych środowiskach niż .NET. **Informacje, te przekazujemy „as is”, bez jakiegokolwiek weryfikacji zasobów zamieszczonych pod n/w adresami oraz bez wsparcia technicznego dla tych rozwiązań.**

3. Rozwiązanie w języku C#

„Dodałem publiczną implementację serwisu w wersji 1.1 na Githubie (wspiera .Net Core)”.

<https://github.com/Adlorem/RegonServiceNetStandard>

4. Informacja dot. dostępu do usługi BIR1 z poziomu Excela (VBA)

„Jeśli będą mieli Państwo zapytania dotyczące dostępu do API z poziomu Excela (VBA) to mogą Państwo przysłać link do przykładu napisanego przeze mnie w Excelu:

<https://github.com/MarekRod/Excel---GUS> ”

5. Pobieranie danych za pomocą curl z poziomu wiersza poleceń

```
curl -X --http1.1 -kvvvv "%URL%" ^-H "Content-Type: application/soap+xml;  
charset=utf-8" ^ --data-binary @login_pytanie_o_sid.xml > login_response.xml 2> nul
```

W celu uzyskania więcej szczegółów — prosimy o kontakt email: regon_bir@stat.gov.pl

6. Informacja dla twórców oprogramowanie klienckie do usługi BIR1 w języku PHP

„Jeśli będą mieli Państwo zapytania dotyczące dostępu do API z poziomu PHP to mogą Państwo przysłać link do biblioteki stworzonej przeze mnie w języku PHP <https://github.com/johnzruk/GusApi/>

7. Informacja dla twórców oprogramowanie klienckie do usługi BIR1 w języku JavaScript

„Stworzyliśmy paczkę do Node.js, którą śmiało mogą wykorzystać inni ludzie zainteresowani tematem: <https://github.com/Appsolutly/node-regon>

8. Informacja dla twórców oprogramowanie klienckie do usługi BIR1 w języku Python

„Jeśli będą mieli Państwo użytkowników, którzy będą korzystać z Państwa usługi z pomocą programów w Pythonie, możecie im Państwo przesłać link do naszej Open Source'owej biblioteki litex.regon: <https://pypi.python.org/pypi/litex.regon>

9. Informacja dla twórców oprogramowanie klienckie do usługi BIR1 w środowisku Delphi

Zostało udostępnione kompletne rozwiązanie (cała klasa do komunikacji).

Plik do pobrania znajduje się tutaj: <http://pliki.merco.pl/izzy.gus.zip>

10. Informacja dla twórców oprogramowanie klienckie do usługi BIR1 w języku Go

„Chciałbym poinformować, że przygotowałem pakiet (bibliotekę) dla języka Go umożliwiającą dostęp do usługi BIR1.1. Pakiet dostępny jest w repozytorium Github pod adresem

<https://github.com/orkanap/regonapi>

Licencja MIT.”

Press [PAGE DOWN] to continue reading...

11. Informacja dla tworzących oprogramowanie klienckie do usługi BIR1 w języku Java

„Klienta można w łatwy sposób wygenerować używając skryptu m.in. z pakietu Apache CXF wsdl2java. Rozwiązanie, z polisą AddressingFeature działa, lecz do zapewnienia pełnej funkcjonalności konieczne jest po uzyskaniu *sid'a*, dokleić go do nagłówków HTTP wychodzących requestów SOAPa”.

Przykład użycia:

```
UslugaBIRzewnPubl service = new UslugaBIRzewnPubl();

service.setHandlerResolver(new SoapHandlerResolver()); //<- doklejanie
SID'a do HTTP HEADER

IUslugaBIRzewnPubl port = service.getE3(new AddressingFeature());

String statusUslugi = "StatusUslugi";

String result = port.getValue(statusUslugi);

// logowanie, jezeli sesja wygasla, badx logowanie pierwszy raz

if ((SoapMessageHandler.sessionCookie.equals("")) || (!
result.equals("1"))){

    String sid = port.zaloguj("abcde12345abcde12345");

    SoapMessageHandler.sessionCookie = sid;

}

// przykład wyszukiwania po NIPie

ObjectFactory objectFactory = new ObjectFactory();

JAXBElement<String> nipParam =
objectFactory.createParametryWyszukiwaniaNip("1234567890");

ParametryWyszukiwania parametryWyszukiwania = new ParametryWyszukiwania();

parametryWyszukiwania.setNip(nipParam);

String raport = port.daneSzukaj(parametryWyszukiwania);
```

(W celu uzyskania więcej szczegółów — prosimy o kontakt email: regon_bir@stat.gov.pl).

Press [PAGE DOWN] to continue reading...

12. Informacja dla twórców oprogramowanie klienckie do usługi BIR1 przy użyciu Ruby on Rails

Link do naszej biblioteki Open Source'owej biblioteki gus_bir1:

https://github.com/espago/gus_bir1

Należy dodać do pliku Gemfile:

```
gem 'gus_bir1'
```

Skonfigurować dostępny

```
# /config/initializers/gus_bir1.rb
```

```
# TEST CONF
```

```
GusBir1.production = false
```

```
GusBir1.client_key = 'abcde12345abcde12345'
```

Wyszukiwać:

```
response = GusBir1.find_by(regon: '00033150100000')
```

Pobierać dane:

```
response = GusBir1.find_and_get_full_data(nips: '8992689516,5261040828')
```

Przykładowe użycie oraz pełny opis dostępny na stronie biblioteki.

13. Informacja dla twórców oprogramowanie klienckie do usługi BIR1 w języku R

Link do repozytorium: https://github.com/NaGrubie/API_REGON

Opis:

Biblioteka w języku R umożliwia zautomatyzowaną obsługę API BIR1.1 GUS — od uwierzytelnienia i zarządzania sesją, przez przetwarzanie wsadowe, aż po pobieranie pełnych raportów rejestrowych. Narzędzie zostało stworzone z myślą o użytkownikach, którzy chcą w prosty sposób przetworzyć dużą liczbę identyfikatorów (np. REGON) bez konieczności ręcznego składania zapytań.

Cały proces — od wczytania listy identyfikatorów z pliku, przez zapytania do API, aż po wygenerowanie końcowego wyniku — odbywa się automatycznie. Wynik dostarczany jest w formie gotowej tabeli zapisanej do pliku tekstowego, co umożliwia jego dalszą analizę lub import do innych systemów. Pełna informacja dostępna bezpośrednio w repozytorium.

14. Informacja dot. integracji z wykorzystaniem PL/SQL

Problem: PL/SQL oczekuje Content-Type: text/xml; charset=utf-8

Rozwiązanie znajduje się tutaj:

https://asktom.oracle.com/pls/apex/f?p=100:11:0:::P11_QUESTION_ID:9533637800346631296

Fragment odpowiedzi poniżej.

<<

We dont have anything native to parse the output [MTOM response], but you could easily just read the stream as raw and take what action your wanted, eg some psuedo-code below

```
declare
  l_req      utl_http.req utl_http.begin_request('your source');
  l_resp     utl_http.resp;
  l_blob     blob;
  l_raw      raw(4000);
  l_header   varchar2(4000);
begin
  l_resp := utl_http.get_response(l_req);

  --
  -- ie, read ascii until you get to a key phrase which means the binary is
  -- to follow. (This is not working code - just an indication of you might
  -- proceed)
  --
  loop
    utl_http.read_text(l_resp, l_header, 20);
    exit when l_header = 'binary data';
  end loop;

  --
  -- then start on the binary stream
  --
  dbms_lob.createtemporary(l_blob, true);
  begin
    loop
      utl_http.read_raw(l_resp, l_raw, 3999);
      dbms_lob.writeappend (l_blob, utl_raw.length(l_raw), l_raw);
    end loop;
  exception
    when utl_http.end_of_body then
      utl_http.end_response(l_resp);
  end;

  -- then do whatever you like with the blob

end;
/
>>
```