Table of Contents

Directo XML datu apmaiņas apraksts 3
Vispārīga informācija 3
Piemēri 3
Datu iegūšana no Directo 3
Datu nosūtīšana uz Directo 4
PHP piemēri 5
Datu izvade 5
Datu ievade 6
XML andmevahetusliidese üldine kirjeldus 7
Üldine 7
Näited 7
Andmete pärimine Directost ehk väljund 7
Praktiline näide - artiklite väljund 8
Andmete saatmine Directosse ehk sisend 8
Praktiline näide - müügiarvete sisend 8
PHP näited 9
Väljund 10
Sisend 10

IZLABO **Lapa nav pilnībā pārtulkota. Lūdzu palīdzi pabeigt tulkojumu!**
(Izdzēs šo rindkopu, kad tulkojums pabeigts!)

Directo XML datu apmaiņas apraksts

Vispārīga informācija

Jebkurai Directo DB ir iespējams konfigurēt datu apmaiņu, kas ļauj nodrošināt abpusēju datu plūsmu. Ir svarīgi ņemt vērā, ka datu apmaiņas saskarne nav standartizēta, bet tiek pielāgota katram gadījumam individuāli, lai atbilstu darba uzdevumā izvirzītajām prasībām. Datu apmaiņas uzstādīšana ir saistīta ar vienreizēju ieviešanas maksu, kas ir atkarīga no datu tipu skaita, kas tiks iesaistīts datu apmaiņā, tāpēc ir vēlams savlaicīgi sazināties ar Directo pārstāvjiem, lai precizētu projekta izmaksas.

Neatkarīgi no projekta specifikas, ir daži principi, kas visos gadījumos ir vienādi:

- Datu formāts ir XML
- Datu kodējums ir UTF-8
- Transporta protokols ir HTTPS
- Informācijas iegūšanai izmanto GET metodi
- Datu ievietošanai izmanto POST metodi
- Datu plūsmu vienmēr iniciē ārējs avots, neatkarīgi no tā, vai dati tiks sūtīti vai pieprasīti
- Datu struktūru nosaka XSD shēma (ja tas ir iespējams, tad labāk lai vērtības būtu atribūtos, nevis elementos)

Piemēri

Dotie piemēri ir veidoti tikai kā vadlīnijas.

Datu iegūšana no Directo

GET pieprasījumam vienmēr ir vismaz trīs parametri:

- `Get = 1`
- `What = ANDMELIIK`
- `Key = PREDEFINEDKEY`

Pārējie parametri ir atkarīgi no konkrēto uzdevumu specifikas.

Praktisks piemērs - artikulu izvade

Tāda veida izvadi bieži izmanto interneta veikali, piemēram, lai noskaidrotu pieejamos produktus un to atlikumus Directo noliktavā.

Piemērā lietotie URL ir tikai ilustratīvi un pat, ja šāds pieprasījums var rezultēties ar datu iegūšanu, dati ir fiktīvi un nav saistīti ar kāda klienta reālajiem datiem!

Šajā piemērā izmantosim parametru `what=item` un atslēgu `20DAF3F822B782AF12D1198C23F45604`
Ar artikulu xsd shēmu var iepazīties šeit: ws_artiklid.xsd

Informācijas par visiem artikuliem pieprasījums:
https://directo.gate.ee/xmlcore/demo_ee/xmlcore.asp?get=1&what=item&key=20DAF3F822B782AF12D1198C23F45604

Lai iegūtu informāciju par kādu konkrētu artikulu, piemēram, pēc artikula koda, mēs norādām papildus parametru `code=0000` un tagad links izskatīsies šādi:
https://directo.gate.ee/xmlcore/demo_ee/xmlcore.asp?get=1&what=item&code=0000&key=20DAF3F822B782AF12D1198C23F45604

Datu nosūtīšana uz Directo

Datus sūta, izmantojot POST metodi un norādot trīs parametrus:

- `put=1`
- `what=ANDMELIIK`
- `xmldata=<?xml version=„1.0“ encoding=„utf-8“?><PAYLOAD>`

Praktisks piemērs - pārdošanas reķina ievade

Šāda tipa ievade, piemēram, var tikt izmantota, lai replicētu kādu ārējās pārdošanas sistēmas rēķinu Directo.

Piemērā izmantotā adrese un xml saturs ir fiktīvi un nav izmantojami testēšanai!)

Pieņemsim, ka datu apmaiņas URL ir:
https://directo.gate.ee/xmlcore/this_is_the_worlds_greatest_company/xmlcore.asp

Un datu struktūra ir šāda: xml_in_arved.xsd

Piemēram, xml var izskatīties šādi (ne visi iespējamie atribūti ir izmantoti):

```
<? xml version = "1.0" encoding = "utf-8" ?>
 <Invoices>
   <INVOICE AppKey = "1234567879012345678790" InvoiceNr = "123"
 InvoiceDate = "2015-06-07" CustomerCode = "CUST001">
     <Lines>
       <line ProductID = "KILLERPRODUCT" Quantity = "3" UnitPriceWoVAT =
"123.45" />
       <line ProductID = "FREESAMPLE" Quantity = "1" UnitPriceWoVAT = "0.00"
/>
     </Lines>
   </INVOICE>
 </Invoices>
```

Pieņemsim, ka datu ievietošana ir noritējusi veiksmīgi un tiek sūtīts šāds sinhronais xml

```
<? xml version = "1.0" encoding = "UTF-8" ?>
 <Results>
   <Result Type = "0" desc = "OK" DocId = "123" doctype = "INVOICE" submit =
"Invoices" />
 </Results>
```

Ja `Result` elementa `Type` atribūts nav vienāds ar 0 tad ir notikusi kāda kļūda un šādā gadījumā `Desc` parādīsies precizējoša informācija.

Piemēram, ja rēķins ar šādu numuru jau eksistē, tad atbildes ziņojums būs šāds:

```
<? xml version = "1.0" encoding = "UTF-8" ?>
 <Results>
   <Result Type = "1" desc = "Duplicate" DocId = "123" doctype = "INVOICE"
submit = "Invoices" />
 </Results>
```

Bet ja vienā sūtījumā būtu bijuši vairāki rēķini:

```
<? xml version = "1.0" encoding = "UTF-8" ?>
 <Results>
   <Result Type = "1" desc = "Duplicate" DocId = "123" doctype = "INVOICE"
submit = "Invoices" />
   <Result Type = "0" desc = "OK" DocId = "124" doctype = "INVOICE" submit =
"Invoices" />
 </Results>
```

Atbildes ziņojuma apstrāde ir būtiski, ja nav nav saņemts apstiprinājums par veiksmīgu datu saņemšanu, tad dati nav ievietoti un ir jāveic atkārtots mēģinājums, bet ne ātrāk kā pēc 60s.

PHP piemēri

Datu izvade

```
$ url =
'https://directo.gate.ee/xmlcore/demo_ee/xmlcore.asp?get=1&what=item&key=20D
AF3F822B782AF12D1198C23F45604';

 $ ch = curl_init ();
 curl_setopt ($ ch, CURLOPT_HEADER, 0);
 curl_setopt ($ ch, CURLOPT_RETURNTRANSFER, true);
 curl_setopt ($ ch, CURLOPT_URL, $ url);
 curl_setopt ($ ch, CURLOPT_POST, 0);

 $ result = curl_exec ($ ch);

 header ( 'Content-Type: text / html; charset = utf-8');

 print '<textarea cols = 80 rows = 10>'.  $ result.  '</ textarea>';
```

```
 $ data = new SimpleXMLElement ($ result);

 print "<pre> ';

 foreach ($ data -> Items -> item as $ item) {
     print $ item [ 'Code'].  chr (9).  '<b>'.  $ item [ 'name'].  '</ b>'.
chr (9).  $ item [ 'class'].  'Estonian';
     foreach ($ item -> datafields -> DATA AS $ df) {
         print chr (9).  $ df [ 'code'].  ':'.  $ df [ 'content'].
'Estonian';
     }
 }
```

Datu ievade

```
$ XMLDATA = '<'.  '? xml version = "1.0" encoding = "UTF-8"?>
  <INVOICE AppKey = "123456787901234567890" InvoiceNr = "123"
      InvoiceDate = "2015-06-07" CustomerCode = "CUST001">
      <Lines>
        <line ProductID = "KILLERPRODUCT" Quantity = "3" UnitPriceWoVAT =
"123.45" />
        <line ProductID = "FREESAMPLE" Quantity = "1" UnitPriceWoVAT =
"00:00" />
      </ lines>
  </ INVOICE> ';

 $ url =
'https://directo.gate.ee/xmlcore/this_is_the_worlds_greatest_company/xmlcore
.asp';

 $ ch = curl_init ();
 curl_setopt ($ ch, CURLOPT_HEADER, 0);
 curl_setopt ($ ch, CURLOPT_RETURNTRANSFER, true);
 curl_setopt ($ ch, CURLOPT_URL, $ url);
 curl_setopt ($ ch, CURLOPT_POST, 1);

 curl_setopt ($ ch, CURLOPT_POSTFIELDS, 'XMLDATA = ". urlencode ($
XMLDATA).' 'put & = 1 & what = INVOICE');
 $ result = curl_exec ($ ch);

 print '<textarea cols = 80 rows = 10>'.  $ result.  '</ textarea>';

 curl_close ($ ch);
```

XML andmevahetusliidese üldine kirjeldus

Üldine

Iga Directo baasi külge saab soovi korral häälestada andmevahetusliidese, mis võimaldab mõlemasuunalist andmeliiklust. Oluline on tähele panna, et liides pole mitte vaikimisi olemas ja kättesaadav, vaid häälestatakse iga konkreetse juhtumi puhul eraldi konkreetse ülesandepüstituse vajadustest lähtuvalt. Liidese paigaldamine on seotud ühekordse häälestustasuga, mille suurus sõltub kummaski suunas liikuvate andmeliikude arvust, seega on enne tööde alustamist vaja ühendust võtta Directo müügiesindajaga ja kokku leppida täpsed vajadused ja kooskõlastada projekti maksumus.

Sõltumata konkreetse liidestusprojekti spetsiifikast on teatud printsiibid, millega peab igal juhul arvestama.

- Andmete formaat on XML
- Andmete kodeering on UTF-8
- Transpordiprotokoll on HTTPS (avalikult tunnustatud sertifikaadiga)
- Andmete pärimisel kasutatakse GET meetodit
- Andmete postitamisel kasutatakse POST meetodit
- Andmeliikluse algatab alati väline osapool sõltumata sellest, kas andmeid päritakse või saadetakse, Directo liides on passiivne
- POST päringule vastab liides sünkroonse XML sõnumiga, mille töötlemine võib protsessi edukuse huvides oluline olla
- Andmete struktuur peab olema kirjeldatav XSD schemana, võimaluse korral eelistada schemat, kus väärtuste kandjaks on atribuudid, mitte elemendid

Näited

Allpooltoodud näiteid tuleb käsitleda ainult umbkaudsete juhistena ja arvestada, et konkreetses loodavas liideses on peaaegu kindlasti kasutusel näites toodutest erinevad

- schemad
- URLid
- Parameetrid, piirangud
- Vastussõnumid
- Võtmed

Andmete pärimine Directost ehk väljund

GET päringu korral on alati täidetud vähemalt kolm parameetrit:

- `get=1`
- `what=ANDMELIIK`
- `key=PREDEFINEDKEY`

ja võimalikud lisaparameetrid. Kasutusel olevad andmeliikide verbid, võtme väärtused ja

andmeliigipõhised lisaparameetrid lepitakse kokku iga konkreetse installatsiooni käigus.

Praktiline näide - artiklite väljund

Seda tüüpi väljundit kasutab tihti näiteks veebipood, et toodete ja nende saadavuse infot Directost poodi laadida.

Käesolevas näites toodud URLid on mõeldud ainult võimaliku päringuformaadi illustreerimiseks. Ühelgi juhul ei ole tagatud, et need URLid päringule ka reaalselt vastavad. Juhul, kui päringu tulemuseks on andmed, on need 100% fiktiivsed ja pole seotud ühegi reaalse äriühingu andmetega.

Antud näiteks kasutatakse verbi `what=item` ja võtme väärtus on `20DAF3F822B782AF12D1198C23F45604`. Soovi korral võib tutvuda struktuuri kirjeldava schemaga ws_artiklid.xsd

Terve tootebaasi päring:
https://directo.gate.ee/xmlcore/demo_ee/xmlcore.asp?get=1&what=item&key=20DAF3F822B782AF12D1198C23F45604

Üksiku toote päring, toote koodi järgi, lisaparameeter `code=0000`:
https://directo.gate.ee/xmlcore/demo_ee/xmlcore.asp?get=1&what=item&code=0000&key=20DAF3F822B782AF12D1198C23F45604

Andmete saatmine Directosse ehk sisend

Andmete saatmine käib POST meetodil, `form/urlencoded` liidese URLi suhtes.

Täidetud on kolm vormimuutujat:

- `put=1`
- `what=ANDMELIIK`
- `xmldata=<?xml version=„1.0“ encoding=„utf-8“?><PAYLOAD>`

Praktiline näide - müügiarvete sisend

Seda tüüpi sisendit kasutatakse näiteks sellisel puhul, kui mingis välises süsteemis genereeritakse müügiarveid, mida on vaja Directos kajastada.

Käesolevas näites toodud URLid ja XML sõnumid on täiesti fiktiivsed ja pole teenuse testimiseks kasutatavad. Testimine toimub alati reaalse installatsiooni suhtes.

Oletame, et liidese URL on antud juhul
https://directo.gate.ee/xmlcore/this_is_the_worlds_greatest_company/xmlcore.asp

Andmeid kirjeldab schema xml_in_arved.xsd

Schemale vastav, aga sugugi mitte kõiki võimalikke atribuute kasutab XML võib välja näha näiteks selline:

```
<?xml version="1.0" encoding="utf-8"?>
<invoices>
  <invoice AppKey="123456787901234567890" InvoiceNr="123"
  InvoiceDate="2015-06-07" CustomerCode="CUST001">
    <lines>
      <line ProductID="KILLERPRODUCT" Quantity="3" UnitPriceWoVAT="123.45"/>
      <line ProductID="FREESAMPLE" Quantity="1" UnitPriceWoVAT="0.00"/>
    </lines>
  </invoice>
</invoices>
```

Postitusele vastab liides sünkroonse XML sõnumiga, kust saab teada, mis juhtus.

Näide, postitus oli edukas ja Directosse võeti vastu müügiarve number 123:

```
<?xml version="1.0" encoding="UTF-8" ?>
<results>
  <Result Type="0" Desc="OK" docid="123" doctype="INVOICE" submit="Invoices"
/>
</results>
```

Kui `Result` elemendi `Type` atribuudi väärtus pole 0, siis on tegemist mingi veaga ja sel juhul ütleb atribuut `Desc` lähemalt, mis häda oli.

Näide, arve number 123 vastuvõtmisest keelduti, kuna sellise numbriga arve oli Directos juba olemas:

```
<?xml version="1.0" encoding="UTF-8" ?>
<results>
  <Result Type="1" Desc="Duplicate" docid="123" doctype="INVOICE"
submit="Invoices" />
</results>
```

Ühest pakis võib olla korraga ka mitu arvet, igaühe kohta tuleb omaette `Result`:

```
<?xml version="1.0" encoding="UTF-8" ?>
<results>
  <Result Type="1" Desc="Duplicate" docid="123" doctype="INVOICE"
submit="Invoices" />
  <Result Type="0" Desc="OK" docid="124" doctype="INVOICE" submit="Invoices"
/>
</results>
```

Vastussõnumi töötlemine on oluline, juhul kui pole õnnestunud saada edukat vastuvõttu tähistavat sõnumit, tuleb saatmine ebaedukaks lugeda ja mingi intervalli järel (mitte vähem kui 60s) uuesti proovida.

PHP näited

Väljund

```
$url =
'https://directo.gate.ee/xmlcore/demo_ee/xmlcore.asp?get=1&what=item&key=20DAF3F822B782AF12D1198C23F45604';

$ch = curl_init();
curl_setopt($ch, CURLOPT_HEADER, 0);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt($ch, CURLOPT_URL, $url );
curl_setopt($ch, CURLOPT_POST, 0);

$result = curl_exec($ch);

header('Content-Type: text/html; charset=utf-8');

print '<textarea cols=80 rows=10>' . $result . '</textarea>';

$data = new SimpleXMLElement($result);

print '<pre>';

foreach ($data->items->item as $item) {
    print $item['Code'] . chr(9) .  '<b>' . $item['Name'] . '</b>' . chr(9) .  $item['Class'] . '<br>';
    foreach ($item->datafields->data as $df ) {
        print chr(9) . $df['code'] . ':' . $df['content'] . '<br>';
    }
}
```

Sisend

```
$xmldata = '<'.'?xml version="1.0" encoding="UTF-8"?>
<invoice AppKey="123456787901234567890" InvoiceNr="123"
    InvoiceDate="2015-06-07" CustomerCode="CUST001">
    <lines>
      <line ProductID="KILLERPRODUCT" Quantity="3" UnitPriceWoVAT="123.45"/>
      <line ProductID="FREESAMPLE" Quantity="1" UnitPriceWoVAT="0.00"/>
    </lines>
</invoice>';

$url =
'https://directo.gate.ee/xmlcore/this_is_the_worlds_greatest_company/xmlcore.asp';

$ch = curl_init();
curl_setopt($ch, CURLOPT_HEADER, 0);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt($ch, CURLOPT_URL, $url );
```

```
curl_setopt($ch, CURLOPT_POST, 1);

curl_setopt($ch, CURLOPT_POSTFIELDS,
'xmldata='.urlencode($xmldata).'&put=1&what=invoice' );
$result = curl_exec($ch);

print '<textarea cols=80 rows=10>' . $result . '</textarea>';

curl_close($ch);
```

From:
https://wiki.directo.ee/ - **Directo Help**

Permanent link:
https://wiki.directo.ee/lv/xmlcore?rev=1532096165

Last update: **2018/07/20 17:16**