

# How does the shop communicate with the ERP

## General

The shop is able to use different transports in order to connect to the ERP systems or other third parties. eZ Commerce (Advanced version only) supports different transport protocols such as

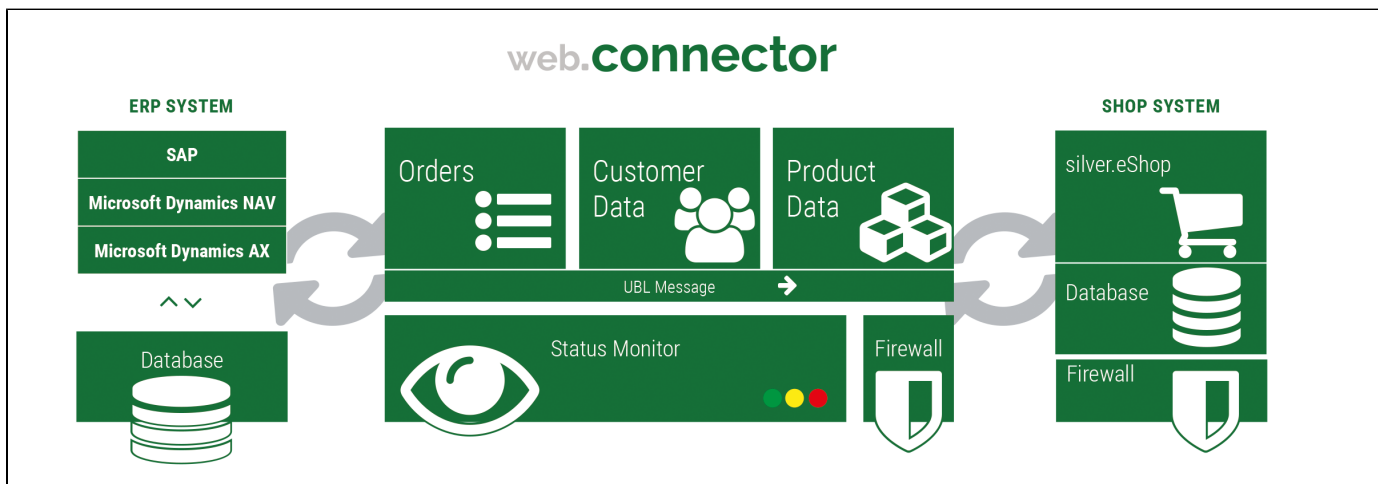
- Webservices/SOAP
- REST

The transport layer can be extended in project if there is a need for an additional protocol or way to access the ERP/CRM...

One way to connect to an SAP or Microsoft Dynamics NAV system is to use the Web.Connector (separately licensed product) which already comes with prepared interfaces to the stated ERP systems. There are other options besides using the Web.Connector:

- if the ERP offers a Webservice/REST interface it would be an option to connect eZ Commerce Advanced directly with the ERP
- An enterprise service bus (ESB) would be an option as well

The following picture describes how the shop communicates with the ERP system using the Web.Connector:



Since every ERP system uses its own standard how data is formatted and named eZ Commerce Advanced uses a standard format to describe data such as customer or orders. The standard used is **UBL (Universal Business Language)** an universal standard describing business processes.

## What happens when a request is sent to an ERP-System

Step	Who	What
1	Shop	Requests data using a request object
2	Shop	A mapping might map the UBL request to a specific request for the ERP (optional step)
3	Web.Connector	The Web.Connector receives the request and maps the request into an ERP request
4	Web.Connector	Sends the ERP request to the ERP
5	ERP	The ERP processes the requests and sends an answer
6	Web.Connector	Gets answer from ERP and maps it to an answer for the shop

7	Shop	Receives the request and maps it to UBL (optional step)
8	Shop	Application gets a proper UBL response object

**Specification for requests and responses**

The shop uses a specification which describes how the request to the ERP (our Web.Connector) and the answer from the ERP will look like.

The specification is written in XML and located in resources/specifications/xml. It includes always two files: a request and a response specification.

It is used to generate PHP objects which can be used to define a request or get the response back from the ERP.

## An example - get customer data

The request is initiated by the shop:

Step	Who	Details	Mapping
UBL-Request	Shop	<p>Sets request using the generated ERP-Classes for select customer.</p> <p>Customer now will be set in</p> <p>BuyerCustomerParty/Party/PartyIdentification/ID</p>	<p>Used specification:</p> <p>@EshopBundle/Resources/specifications/xml/request.selectC</p> <pre> &lt;?xml version="1.0" encoding="UTF-8" ?&gt; &lt;BuyerCustomerParty&gt;   &lt;Party ses_unbounded="Party"&gt;     &lt;PartyIdentification&gt;       &lt;ID&gt;10000&lt;/ID&gt;     &lt;/PartyIdentification&gt;   &lt;/Party&gt; &lt;/BuyerCustomerParty&gt; </pre>
maps request	Shop	<p>Request is mapped to</p> <pre> &lt;PARTY&gt;&lt;PARTY_ID&gt;0100000&lt;/PARTY_ID&gt;&lt;/PARTY&gt; </pre>	<p>Used mapping</p> <p>app/Resources/xsl/request.selectcustomer.xsl</p> <pre> &lt;?xml version="1.0" encoding="UTF-8" ?&gt; &lt;xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"&gt;   &lt;xsl:template match="/BuyerCustomerParty"&gt;     &lt;PARTY&gt;       &lt;PARTY_ID&gt;         &lt;xsl:value-of select="Party/PartyIdentification/ID"/&gt;       &lt;/PARTY_ID&gt;     &lt;/PARTY&gt;   &lt;/xsl:template&gt; &lt;/xsl:stylesheet&gt; </pre>

Receives request	<b>Web.Connector</b>	Converts the request in the language of the ERP, e.g. <pre>&lt;Params&gt;&lt;Read&gt;&lt;No&gt;0100000&lt;/No&gt;&lt;/Read&gt;&lt;/Params&gt;</pre>	Used mapping mapping/nav_ws/xsl/request.selectcustomer.xsl  <pre>&lt;xsl:stylesheet version="1.0"   xmlns:xsl="http://www.w3.org/1999/xsl"   &lt;xsl:param name="WEBSITEID"   &lt;xsl:template match="/"     &lt;Params&gt;       &lt;Read&gt;         &lt;No&gt;&lt;xsl:value-of select="       &lt;/Read&gt;     &lt;/Params&gt;   &lt;/xsl:template&gt; &lt;/xsl:stylesheet&gt;</pre>
Processes request	<b>ERP-System</b>	Gets the request and will send a result back	

The answer will be mapped one or more time depending on the configuration:

Step	Who	Details
------	-----	---------

receives  
customer  
data from  
ERP

**Web.Connector**

Data from ERP

```
<RawResponse>
  <Key>28;EgA...</Key>
  <No>0100000</No>
  <Blocked>_blank_</Blocked>
  <Name>Demo customer</Name>
  <Name_2>- Office -</Name_2>
  <Address>Demo street</Address>
  <Post_Code>12555</Post_Code>
  <City>Berlin</City>
  <Country_Region_Code>DE</Country_Region_Code>
  <Fax_No>0301212</Fax_No>
  <Phone_No>03012121</Phone_No>
  <E-Mail>info@test.de</E-Mail>
  <Contact>- -</Contact>
  <Customer_Posting_Group>SAMMEL</Customer_Posting_Group>
  <Gen_Bus_Posting_Group>NATIONAL</Gen_Bus_Posting_Group>
  <VAT_Bus_Posting_Group>NATIONAL</VAT_Bus_Posting_Group>

  <Kontakt_Email>info@test.de</Kontakt_Email>
  <Kontakt_Ansprechpartner>- -</Kontakt_Ansprechpartner>
  <PaymentMethodCode>EINZUG</PaymentMethodCode>
  <Ship_to_Address>
    <Cust_Ship_to_Address_WS>
      <Key>44;3gAAAAJ7BzAAMQAwADAAMAAwADAAAAACe/9MADAAMQ==9;1326
      <Customer_No>0100000</Customer_No>
      <Code>L01</Code>
      <Name>Demo customer</Name>
      <Name_2>Max Mueller</Name_2>
      <Address>Molkereistr. 1a</Address>
      <City>Berlin</City>
      <Post_Code>12555</Post_Code>
      <Blocked/>
    </Cust_Ship_to_Address_WS>

  </Ship_to_Address>
</RawResponse>
```

Maps data back	Web.Connector	Maps data for shop using a 1:1 vonversion
Receivers repsonse and maps it to UBL	Shop	Converts the answer into UBL using a mapping and the shop will convert the response to an UBL object as defined in the specific: pecifications/xml/response.selectCustomer.xml)











## Web.Connector setup - a short introduction

In eZ Commerce Advanced some config parameters have to be setup:

The URL to the Web.Connector:

webconnector\_url: [https://<ip>:<port>/webconnector/webconnector\\_opentrans.php5](https://<ip>:<port>/webconnector/webconnector_opentrans.php5)

and the configuration for each message:

```
silver_erp.config.messages:
    .....
    select_customer:
        message_class:
            "Silersolutions\\Bundle\\EshopBundle\\Message\\SelectCustomerMessage"
        response_document_class:
            "\\Silersolutions\\Bundle\\EshopBundle\\Entities\\Messages\\Document\\CustomerResponse"
        webservice_operation: "SV_OPENTRANS_SELECT_CUSTOMERINFO"
        mapping_identifier: "selectcustomer"
```

If a mapping\_identifier is defined eZ Commerce will use a mapping (XSLT based).

On the Web.Connector PC for each message a configuration describes which mapping shall be performed:

```
$cfg->setSetting('WebserviceHandler_Config', 'MethodSettings', array(
    .....
    'SV_OPENTRANS_SELECT_CUSTOMERINFO' => array(
        'remote_reference' => 'customer',
        'method' => 'Read',
        'mapping' => 'selectCustomer',
        'mapping_parameters' => array(
            'WEBSITE' => 'HOME',
        ),
    ),
),
```

## Example for developers

```

$erpService = $this->getContainer()->get('silver_erp.facade');

$customerNo = "1000";
$selectCustomerResponse = $erpService->selectCustomer($customerNo);

```

The answer will be an object as defined in @EshopBundle/Resources/specifications/xml/response.selectCustomer.xml:

```

Siversolutions\Bundle\EshopBundle\Entities\Messages\Document\CustomerResponse Object
(
    [BuyerCustomerParty] =>
    Siversolutions\Bundle\EshopBundle\Entities\Messages\Document\CustomerResponseBuyerCustomerParty Object
        (
            [Party] =>
            Siversolutions\Bundle\EshopBundle\Entities\Messages\Document\Party Object
                (
                    [PartyIdentification] => Array
                        (
                            [0] =>
                            Siversolutions\Bundle\EshopBundle\Entities\Messages\Document\PartyPartyIdentification Object
                                (
                                    [ID] =>
                                    Siversolutions\Bundle\EshopBundle\Entities\Messages\Document\StringObject Object
                                        (
                                            [value] => 0100000
                                        )
                                    )
                                )
                            [PartyName] => Array
                                (
                                    [0] =>
                                    Siversolutions\Bundle\EshopBundle\Entities\Messages\Document\PartyPartyName Object
                                        (
                                            [Name] =>
                                            Siversolutions\Bundle\EshopBundle\Entities\Messages\Document\StringObject Object
                                                (
                                                    [value] => Demo customer
                                                )
                                            )
                                    [1] =>
                                    Siversolutions\Bundle\EshopBundle\Entities\Messages\Document\PartyPartyName Object
                                        (

```

```

                                [Name] =>
Silverolutions\Bundle\EshopBundle\Entities\Messages\Document\StringObject
Object
                                (
                                    [value] => - Office -
                                )
                            )
                    [PostalAddress] =>
Silverolutions\Bundle\EshopBundle\Entities\Messages\Document\PartyPostalA
ddress Object
                            (
                                [StreetName] =>
Silverolutions\Bundle\EshopBundle\Entities\Messages\Document\StringObject
Object
                                    (
                                        [value] => Demo street 12
                                    )
                                [AdditionalStreetName] =>
Silverolutions\Bundle\EshopBundle\Entities\Messages\Document\StringObject
Object
                                    (
                                        [value] =>
                                    )
                                [BuildingNumber] =>
Silverolutions\Bundle\EshopBundle\Entities\Messages\Document\StringObject
Object
                                    (
                                        [value] =>
                                    )
                                [CityName] =>
Silverolutions\Bundle\EshopBundle\Entities\Messages\Document\StringObject
Object
                                    (
                                        [value] => Berlin
                                    )
                                [PostalZone] =>
Silverolutions\Bundle\EshopBundle\Entities\Messages\Document\StringObject
Object
                                    (

```

```
[value] => 12555
```

```
)
```

```
....
```