

@Leisure Partner Guide

XML-RPC

Contact information

@Leisure

Web Development

Doornakkersweg 26

Eindhoven, The Netherlands

Tel: 040 - 216 36 00

Fax: 040 - 212 44 80

Web: www.leisure-affiliate.com

E-mail: webtech@eurorelais.com

Version history

<i>Date</i>	<i>Revision</i>	<i>Change</i>
07-09-2009	1.4	Payment details removed
04-09-2007		Photo section changed
05-07-2007		Photo link changed
05-01-2007	1.3a	Minor layout changes
01-08-2006	1.3	- Changed contact email address - Removed Search.Houses call
03-06-2006	1.2	Lastminute explanation added
09-05-2006	1.1	PaymentMethod and PaymentDetails in CheckBeforeOption and PlaceOption changed from mandatory to optional.
01-05-2006	1.1	Added calls: "Misc.BrochureRequest" (2.13) "Misc.PotentialHouseOwnerSignUp" (2.14) Changed: (Added optional fields: WebpartnerAgentCode, WebpartnerAgentPassword) "Booking.CheckBeforeBooking-v2" (2.8) "Booking.PlaceBooking-v2" (2.9) "Booking.CheckBeforeOption-v2" (2.10) "Booking.PlaceOption-v2" (2.11)
13-02-2006	1.0	Created

Table of contents

Contact information	2
Version history	3
Table of contents	4
1. General information	6
1.1 Components	6
1.2 Implementation guidelines	6
1.3 Commission	6
1.4 Testing	7
1.5 Picture server	8
1.6 Lastminutes	9
2. The XML-RPC files	10
2.1 introduction	10
2.2 common error codes	10
2.3 Search.PlanningOfOneHouse-v2	12
2.3.1 General	12
2.3.2 XML-RPC request structure	12
2.3.3 XML-RPC request structure description	13
2.3.4 XML-RPC answer structure	13
2.3.5 XML-RPC answer structure description	14
2.3.6 Error codes	14
2.4 Search.DetailsOfOneHouse	15
2.4.1 General	15
2.4.2 XML-RPC request structure	15
2.4.3 XML-RPC request structure description	15
2.4.4 XML-RPC answer structure	16
2.4.5 XML-RPC answer structure description	18
2.4.6 Error codes	18
2.5 Misc.RegisterForNewsletter-v2	19
2.5.1 General	19
2.5.2 XML-RPC request structure	19
2.5.3 XML-RPC request structure description	20
2.5.4 XML-RPC answer structure	20
2.5.5 XML-RPC answer structure description	20
2.5.6 Error codes	20
2.6 Booking.CheckCryptedString	21
2.6.1 General	21
2.6.2 XML-RPC request structure	21
2.6.3 XML-RPC request structure description	21
2.6.4 XML-RPC answer structure	22
2.6.5 XML-RPC answer structure description	22
2.6.6 Error codes	22
2.7 Booking.CheckBeforeBooking-v2	23
2.7.1 General	23
2.7.2 XML-RPC request structure	23
2.7.3 XML-RPC request structure description	26
2.7.4 XML-RPC answer structure	27
2.7.5 XML-RPC answer structure description	27
2.7.6 Error codes	28
2.8 Booking.PlaceBooking-v2	30
2.8.1 General information	30
2.8.2 XML-RPC request structure	30
2.8.3 XML-RPC request structure description	30
2.8.4 XML-RPC answer structure	30
2.8.5 XML-RPC answer structure description	31
2.8.6 Error codes	31
2.9 Booking.CheckBeforeOption-v2	33
2.9.1 General	33

2.9.2 XML-RPC request structure	33
2.9.3 XML-RPC request structure description	35
2.9.4 XML-RPC answer structure	37
2.9.5 XML-RPC answer structure description	37
2.9.6 Error codes	37
2.10 Booking.PlaceOption-v2	39
2.10.1 General	39
2.10.2 XML-RPC request structure	39
2.10.3 XML-RPC request structure description	39
2.10.4 XML-RPC answer structure	39
2.10.5 XML-RPC answer structure description	39
2.10.6 Error codes	40
2.11 Booking.DiscountCode	42
2.11.1 General	42
2.11.2 XML-RPC request structure	42
2.11.3 XML-RPC request structure description	42
2.11.4 XML-RPC answer structure	43
2.11.5 XML-RPC answer structure description	43
2.11.6 Error codes	43
2.12 Misc.BrochureRequest	44
2.12.1 General	44
2.12.2 XML-RPC request structure	44
2.12.3 XML-RPC request structure description	45
2.12.4 XML-RPC answer structure	46
2.12.5 XML-RPC answer structure description	46
2.12.6 Error codes	46
2.13 Misc.PotentialHouseOwnerSignUp	47
2.13.1 General	47
2.13.2 XML-RPC request structure	47
2.13.3 XML-RPC request structure description	49
2.13.4 XML-RPC answer structure	50
2.13.5 XML-RPC answer structure description	50
2.13.6 Error codes	51
3. Examples	52
3.1 Introduction	52
3.2 Example 2: House presentation	53
3.4 Example 3: making a booking	60

1. General information

1.1 Components

This document describes the XML-RPC interface that Euro Relais has available for their web partners. With this interface the web partner can do a pre-defined request on the server, and get the information he wants. The main goal for the XML-RPC interface is that you don't need to download the complete set of XML files.

This guide describes the following components:

- Detail descriptions of the pre-defined XML-RPC interface calls.
- Examples for implementing the XML-RPC interface calls.

1.2 Implementation guidelines

There are several ways to implement the XML-RPC interface. This document describes the implementation, but doesn't give code examples for particular programming languages. We chose for this solution, because there are several programming languages you can use. In this document we will explain the procedures and show how to search houses, display our house data and how to make your own bookings form. Of course you are free to use any programming language, but we cannot give support on them.

The following scheme is a rough layout of a XML-RPC setup:

1. Search houses within your database (using our XML files);
2. Display the search results;
3. Display house details with the "Search.DetailsOfOneHouse" call;
4. Display house planning with the "Search.PlanningOfOneHouse-v2" call;
5. Create your own bookings form with the "Booking.CheckBeforeBooking-v2", "Booking.PlaceBooking-v2", "Booking.CheckBeforeOption-v2", "Booking.PlaceOption-v2" calls, or link to our book engine.

With our XML-RPC interface it is possible to make your own booking form. It is also possible to show house availability and call our booking form with you partner code (Pt) to make a booking. The booking process will be handled by Euro Relais. You can alter the colors of the booking form so it matches your website design. You can specify a layout code (lo) so the engine will be displayed in the colors you specified in our web partner interface.

More will be explained in the next paragraphs.

1.3 Commission

Euro Relais pays a commission on each booking on a percentage basis. In order to assign all bookings correctly the web partner must submit a so called "partner code" (Pt) when making the XML-RPC call or redirecting to the booking engine. The partner code is assigned by your Euro Relais contact person when you become a web partner. It is of great importance that you submit your partner code in order to get your commission!

1.4 Testing

In order to test you implemented solution we have so called 'demo' (XX-) houses available with demo availability. When you book a 'demo' house no real booking will be made because the information entered will not be send to the financial administration.

1.5 Picture server

Euro Relais uses a picture server that contains all pictures of the available accommodations. We use our own picture server to guarantee that the pictures are updated when an accommodation changes. Another advantage is that you can just link to the pictures instead of downloading them all.

When a picture is available for an accommodation there will be one or more tags in the XML file that would look something like this:

The big picture:

```
<Pictures72dpi>  
  <Picture72dpi>http://www.houseimages.net/images/houses/FR/FR-59149-01/FR-59149-01-b-b1-170.jpg</Picture72dpi>  
</Pictures72dpi>
```

The small picture

```
<Pictures72dpiSmall>  
  <Picture72dpiSmall>http://www.houseimages.net/images/houses/FR/FR-59149-01/FR-59149-01-b-b1-84.jpg</Picture72dpiSmall>  
</Pictures72dpiSmall>
```

For each picture there will be a 'Picture72dpi' and 'Picture72dpiSmall' tag.

Note: You can also use the secure (https://) version of this link.

Explanation of letters used in photo name:

z=summerphoto
w=winterphoto
b=both summer and winter photo present

e=exterior
w=livingroom
d=diningroom
k=kitchen
s=bedroom
b=bathroom
t=patio/tuin
z=pool
f=facilities
a=otherwise
o=surroundings
(followed by sequencenumber, 1-2-3 etc)

1.6 Lastminutes

A house is a lastminute when the following criteria are met:

1. RentPrice is lower than RentPriceExclDiscount
2. The arrival date is maximum 6 weeks (42 days) in advance
3. The arrival date is not today
4. A house is not bookable on request (BookingOnRequest = 'no')

With above criteria you can create your own lastminute collection if you like to display them.

If a booking is within 2 weeks before the arrival date, online booking is not available. An option will be placed which has to be confirmed by telephone. The customer will be informed what to do if you use our booking screen.

2. The XML-RPC files

2.1 introduction

XML-RPC is a Remote Procedure Calling (RPC) protocol that works via the Internet. The protocol (the body) is written in eXtensive Markup Language (XML). RPC is a simple extension of the idea of the 'procedure call', with which connections are set up between procedures that run on different applications and/or on different computers. On a conceptual level, there is no difference between a Local Procedure and a Remote Procedure. However a Remote Procedure is implemented differently.

An XML-RPC message is a HTTP-POST request, with which a procedure can be called. The procedure to be called is executed on a server. The outcome of an executed procedure is sent back to the caller. This outcome (Response) is also in an XML format.

By using the XML-RPC files of Euro Relais, you as a partner are given freedom of choice. The Remote Call is put into a format that can be understood at the other end of the connection. This allows you, as a partner, to come up with customized solutions, based on the web services offered by Euro Relais.

Euro Relais, for instance, offers you the possibility to verify the availability of a home online, without first having to download all the XML files from Euro Relais. This also adds the possibility of verifying real-time availability to existing XML files. We refer to this web service as the Procedure Call 'Search.PlanningOfOneHouse'.

For more information on XML-RPC look at www.xmlrpc.com

To test the XMLRPC server put the XMLRPC call in a file 'input.xml' and then run the command:

```
curl -s -k --header "Content-Type: text/xml" --data-binary @input.xml  
https://www.eurorelais.net/cgi/lars/extern/xmlrpc/xmlrpc.htm
```

The call is only available in https (port 443).

If you run Linux the curl command should be on your system. If you run windows then contact Euro Relais or have a look at this site: <http://curl.haxx.se>

The XML-RPC server is offline every day for backup from 3:00 AM to 3:20 AM

Changes in XML-RPC definitions files will be announced by email 4 weeks in advance.

2.2 common error codes

Error code	Description
100	Input is not in XML format
101	methodName ... is missing or unknown
102	Parameter WebpartnerCode is missing
103	Parameter WebpartnerPassword is missing
104	Web partner Code / Password incorrect
105	Communication not in https / port 443
106	Web partner is not active
107	Parameter ... is unknown

Fault error structure.

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<methodResponse>
  <fault>
    <value>
      <struct>
        <member>
          <name>faultCode</name>
          <value><int></int></value>
        </member>
        <member>
          <name>faultString</name>
          <value><string> </string></value>
        </member>
      </struct>
    </value>
  </fault>
</methodResponse>
```

2.3 Search.PlanningOfOneHouse-v2

2.3.1 General

This call returns the planning records of one house. If there are no errors, the result is an XML-RPC message with the same record layout as in the planning.xml file. With this call you don't have to implement the complete planning.xml file into your database. Also, the planning you see for a house is in real-time.

2.3.2 XML-RPC request structure

```
<?xml version="1.0"?>
<methodCall>
  <methodName>Search.PlanningOfOneHouse-v2</methodName>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>WebpartnerCode</name>
            <value><string></string></value>
          </member>
          <member>
            <name>WebpartnerPassword</name>
            <value><string></string></value>
          </member>
          <member>
            <name>HouseCode</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>Period</name>
            <value><string></string></value>
          </member>
          <member>
            <name>FromDate</name>
            <value><string></string></value>
          </member>
          <member>
            <name>UntilDate</name>
            <value><string></string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodCall>
```

2.3.3 XML-RPC request structure description

<i>Parameter</i>	<i>Value</i>	<i>Mandatory/Optional</i>
WebpartnerCode	The partner code you received.	Mandatory
WebpartnerPassword	The password you received.	Mandatory
HouseCode	The code of the house you want to have the planning for.	Mandatory
Period	1w, 2w, 3w, mw, lw, wk or empty.	Optional
FromDate	Availability from date format DD-MM-YYYY	Optional
UntiDate	Availability from date format DD-MM-YYYY	Optional

2.3.4 XML-RPC answer structure

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<methodResponse>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>period</name>
            <value><string> </string></value>
            <name>arrival-date</name>
            <value><string> </string></value>
            <name>arrival-time-from</name>
            <value><string> </string></value>
            <name>arrival-time-until</name>
            <value><string> </string></value>
            <name>departure-date</name>
            <value><string> </string></value>
            <name>departure-time</name>
            <value><string> </string></value>
            <name>booking-on-request</name>
            <value><string> </string></value>
            <name>rent-price</name>
            <value><string> </string></value>
            <name>rent-price-excl-discount</name>
            <value><string> </string></value>
            <name>cs</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>period</name>
            <value><string> </string></value>
            <name>arrival-date</name>
            <value><string> </string></value>
            <name>arrival-time-from</name>
            <value><string> </string></value>
```

```
<name>arrival-time-until</name>
<value><string> </string></value>
<name>departure-date</name>
<value><string> </string></value>
<name>departure-time</name>
<value><string> </string></value>
<name>booking-on-request</name>
<value><string> </string></value>
<name>rent-price</name>
<value><string> </string></value>
<name>rent-price-excl-discount</name>
<value><string> </string></value>
<name>cs</name>
<value><string> </string></value>
</member>
</struct>
</value>
</param>
</params>
</methodResponse>
```

2.3.5 XML-RPC answer structure description

Parameter	Description
Period	The period of the planning record (1w = 1 week, 2w = 2 weeks, 3w = 3 weeks, wk = weekend, lw = long weekend, mw = midweek).
Arrival-date	The arrival date.
Arrival-time-from	The arrival time is between ArrivalTimeFrom and ArrivalTime Until.
Arrival-time-until	The arrival time is between ArrivalTimeFrom and ArrivalTime Until.
Departure-date	The departure date.
Departure-time	the time when you have to departure.
Booking-on-request	Value for identifying if the booking is on request.
Rent-price	The actual rent price.
Rent-price-excl-discount	The rent price exclusive discount.
Cs	The crypted string needed to make a booking.

2.3.6 Error codes

Error code	Description
1001	Parameter HouseCode is missing
1002	Parameter HouseCode is mandatory
1003	House ... is unknown
1004	Parameter Period is missing
1005	Period ... is unknown
1006	Parameter FromDate is missing
1007	FromDate ... is invalid
1008	Parameter UntilDate is missing
1009	UntilDate ... is invalid

2.4 Search.DetailsOfOneHouse

2.4.1 General

This call returns the details of one house. If there are no errors, the result is an XML-RPC message with the same record layout as in the house.xml file, but only the fields you select in the “FieldList”. With this call you don’t have to implement the complete house.xml file into your database.

2.4.2 XML-RPC request structure

```
<?xml version="1.0"?>
<methodCall>
  <methodName>Search.DetailsOfOneHouse</methodName>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>WebpartnerCode</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>WebpartnerPassword</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>HouseCode</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>FieldList</name>
            <value><string> </string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodCall>
```

2.4.3 XML-RPC request structure description

Parameter	Value	Mandatory/Optional
WebpartnerCode	The partner code you received	Mandatory
WebpartnerPassword	The password you received	Mandatory
HouseCode	The code of the house you want to have the details for.	Mandatory
FieldList	A comma separated list of all fields you want to have in the returning answer, names are the same is in	Mandatory

the house.xml file.

2.4.4 XML-RPC answer structure

The answer of the XML-RPC request is dependent of the field list that is given in the request. The next answer is given when the field list parameter is set to:

“Code,Name,City,introductionEN,descriptionEN,DepositEN,EnergyEN,petsEN,CleaningEN,TouristTaxEN,BedLinenEN,BathTowelsEN,KitchenLinenEN,EnvironmentEN,Pictures72dpi,ZipPostalcode,Properties,PropertiesDefinitionEN”.

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<methodResponse>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>Code</name>
            <value><string> </string></value>
            <name>Name</name>
            <value><string> </string></value>
            <name>ZipPostalCode</name>
            <value><string> </string></value>
            <name>City</name>
            <value><string> </string></value>
            <name>IntroductionEN</name>
            <value><string> </string></value>
            <name>DescriptionEN</name>
            <value><string> </string></value>
            <name>BathTowelsEN</name>
            <value><string> </string></value>
            <name>DepositEN</name>
            <value><string> </string></value>
            <name>EnergyEN</name>
            <value><string> </string></value>
            <name>PetsEN</name>
            <value><string> </string></value>
            <name>KitchenLinenEN</name>
            <value><string> </string></value>
            <name>BedLinenEN</name>
            <value><string> </string></value>
            <name>EnvironmentEN</name>
            <value><string> </string></value>
            <name>CleaningEN</name>
            <value><string> </string></value>
            <name>TouristTaxEN</name>
            <value><string> </string></value>
            <name>Pictures72dpi</name>
            <value><string> </string></value>
            <name>Properties</name>
            <value><string> </string></value>
            <name>PropertiesDefinitionEN</name>
            <value><string> </string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodResponse>
```



```
</struct>  
</value>  
</param>  
</params>  
</methodResponse>
```

2.4.5 XML-RPC answer structure description

Because the XML-RPC answer is dependent of the field list of the request, the description given is the answer when field list parameter is set to:

“Code,Name,City,introductionNL,descriptionNL,DepositNL,EnergyNL,petsNL,CleaningNL,TouristTaxNL,BedLinenNL,BathTowelsNL,KitchenLinenNL, EnvironmentNL,Pictures72dpi,ZipPostalcode”.

<i>Parameter</i>	<i>Description</i>
Code	The house code
Name	Name of the house
ZipPostalCode	Zip / postal code
City	City
IntroductionEN	House introduction in the English(EN) language.
DescriptionEN	House description in the English(EN) language.
BathTowelsEN	If the bath towels are inclusive, or if you have to pay for them, and how much.
DepositEN	If you need to make a deposit and how much.
EnergyEN	If you have to pay the energy and an indication of the costs.
PetsEN	If you have to pay for pets and how much you have to pay
KitchenLinenEN	If you can rent kitchen linen and how much the rent is.
BedLinenEN	If you can rent bed linen and how much the rent is.
EnvironmentEN	If you have to pay Environmental taxes and how much you have to pay.
CleaningEN	If you have to pay the cleaning of the house and how much you have to pay.
TouristTaxEN	If you have to pay the tourist tax and how much you have to pay.
Pictures72dpi	The location of pictures, separated by a comma.
Properties	A comma separated list of the properties of the house.
PropertiesDefinitionEN	A comma separated list of the properties in the English (EN) language.

2.4.6 Error codes

<i>Error code</i>	<i>Description</i>
1101	Parameter HouseCode is missing
1102	House ... is unknown
1103	House ... can not be booked
1104	Parameter FieldList is missing
1105	Parameter FieldList is mandatory

2.5 Misc.RegisterForNewsletter-v2

2.5.1 General

With this XML-RPC call you can register an email address for the Newsletter of Euro Relais.

2.5.2 XML-RPC request structure

```
<?xml version="1.0"?>
<methodCall>
  <methodName>Misc.RegisterForNewsletter-v2</methodName>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>WebpartnerCode</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>WebpartnerPassword</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>Email</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>ClientLanguage</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>ClientTitle</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>ClientName</name>
            <value><string> </string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodCall>
```

2.5.3 XML-RPC request structure description

All parameters are mandatory, but the value of some parameters may be empty (optional)

Parameter	Value	Mandatory/Optional
WebpartnerCode	The partner code you received	Mandatory
WebpartnerPassword	The password you received	Mandatory
Email	Email address you want to register for the newsletter	Mandatory
ClientLanguage	What language the newsletter must be (possible values: NL,FR,DE,EN,IT,ES).	Mandatory
ClientTitle	Title of the Client (possible values: Mr, Ms, Fam)	Optional
ClientName	The name of the client.	Optional

2.5.4 XML-RPC answer structure

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<methodResponse>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>RegisteredForNewsletter</name>
            <value><string>OK</string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodResponse>
```

2.5.5 XML-RPC answer structure description

Parameter	Description
RegisteredForNewsletter	If the Client added to the newsletter then Value = OK

2.5.6 Error codes

Error code	Description
3101	Parameter Email is missing
3102	Parameter Email is mandatory
3103	Email ... is invalid
3104	Parameter ClientLanguage is missing
3105	ClientLanguage ... is unknown
3106	Email ... is already registered
3107	Parameter ClientTitle is missing
3108	ClientTitle is invalid
3109	Parameter ClientName is missing

2.6 Booking.CheckCryptedString

2.6.1 General

With this XML-RPC call you can check the Crypted String you want to use for a booking. The Crypted String is the CS field in the Search.PlanningOfOneHouse-v2 XML-RPC call, and the CS field you find in the planning-v2.xml. If there are no errors, the result is an XML-RPC message with the value “valid” or “invalid”.

2.6.2 XML-RPC request structure

```
<?xml version="1.0"?>
<methodCall>
  <methodName>Booking.CheckCryptedString</methodName>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>WebpartnerCode</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>WebpartnerPassword</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>CryptedString</name>
            <value><string> </string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodCall>
```

2.6.3 XML-RPC request structure description

All parameters are mandatory, but the value of some parameters may be empty (optional)

Parameter	Value	Mandatory/Optional
WebpartnerCode	The partner code you received	Mandatory
WebpartnerPassword	The password you received	Mandatory
CryptedString	The CS code found in the planning-v2.xml or Search.PlanningOfOneHouse-v2	Mandatory

2.6.4 XML-RPC answer structure

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<methodResponse>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>CheckCryptedString</name>
            <value><string> </string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodResponse>
```

2.6.5 XML-RPC answer structure description

<i>Parameter</i>	<i>Description</i>
CheckCryptedString	Value = valid or invalid

2.6.6 Error codes

<i>Error code</i>	<i>Description</i>
2001	Parameter CryptedString is missing
2002	Parameter CryptedString is mandatory

2.7 Booking.CheckBeforeBooking-v2

2.7.1 General

With this XML-RPC call you can check if you have all the correct information for making a booking. If there are no errors, the result is an XML-RPC message with the value OK.

2.7.2 XML-RPC request structure

```
<?xml version="1.0"?>
<methodCall>
  <methodName>Booking.CheckBeforeBooking-v2</methodName>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>WebpartnerCode</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>WebpartnerPassword</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>WebpartnerBookingCode</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>WebpartnerAgentCode</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>WebpartnerAgentPassword</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>Brand</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>CryptedString</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>ClientTitle</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>ClientSurname</name>
            <value><string> </string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodCall>
```

```
<member>
  <name>ClientInitials</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientStreet</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientHouseNumber</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientHouseNumberSuffix</name>
  <value><string></string></value>
</member>
<member>
  <name>ClientZipCode</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientCity</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientCountry</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientTelephone</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientTelephoneMobile</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientEmail</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientLanguage</name>
  <value><string> </string></value>
</member>
<member>
  <name>NumberOfAdults</name>
  <value><string> </string></value>
</member>
<member>
  <name>NumberOfBabies</name>
  <value><string> </string></value>
</member>
<member>
  <name>NumberOfPets</name>
  <value><string> </string></value>
</member>
<member>
```

```
<name>RentPrice</name>
<value><string> </string></value>
</member>
<member>
<name>CancellationInsurance</name>
<value><string> </string></value>
</member>
<member>
<name>CancellationInsuranceInclGuarantee</name>
<value><string> </string></value>
</member>
<member>
<name>TravelInsurance</name>
<value><string> </string></value>
</member>
<member>
<name>TravelInsuranceWinterSports</name>
<value><string> </string></value>
</member>
<member>
<name>DiscountCode</name>
<value><string> </string></value>
</member>
<member>
<name>DiscountAmount</name>
<value><string> </string></value>
</member>
<member>
<name>TotalAmount</name>
<value><string> </string></value>
</member>
</struct>
</value>
</param>
</params>
</methodCall>
```

2.7.3 XML-RPC request structure description

All parameters are mandatory, but the value of some parameters may be empty (optional)

Parameter	Value	Mandatory/Optional
ClientCity	Name of the city	Mandatory
ClientCountry	2 letter country ISO code (NL, BE, DE,FR ,etc).	Mandatory
ClientEmail	E-mail address.	Mandatory
ClientLanguage	What language the client speaks (possible values: NL,FR,DE,EN,IT,ES).	Mandatory
ClientHouseNumber	House number, only integer.	Mandatory
ClientStreet	Name of the street without the house number.	Mandatory
CryptedString	The Field "CS" from the XML file, or the XML-RPC.	Mandatory
ClientSurname	Family name of the Client	Mandatory
ClientTelephone	Telephone number, don't include country codes.	Mandatory
ClientTitle	Title of the Client (possible values: Mr, Ms, Fam)	Mandatory
ClientZipCode	Zip code, with check for the Netherlands	Mandatory
NumberOfAdults	Amount of adults older then 3 years old. Only integer.	Mandatory
RentPrice	The amount of rent to be paid.	Mandatory
TotalAmount	The total amount a customer has to pay.	Mandatory
WebpartnerCode	The partner code you received	Mandatory
WebpartnerPassword	The password you received	Mandatory
CancellationInsurance	The amount of cancellation insurance contribution.	Optional
P = NumberOfAdults + NumberOfBabies if p <= 6 then contribution = round(RentPrice x 0.055) if p > 6 and <= 10 then contribution = round(RentPrice x 0.080) if p > 10 and <= 20 then contribution = round(RentPrice x 0.085) if p > 20 <= 40 then contribution = round(RentPrice x 0.095) if p > 40 then contribution = round(RentPrice x 0.100)		
CancellationInsuranceInclGuarantee	The amount of cancellation insurance contribution inclusive guarantee. Only if ClientCountry = NL or BE	Optional
	contribution = round(RentPrice x 0.01)	

ClientTelephoneMobile	Cell phone number.	Optional
ClientHouseNumberSuffix	House number suffix.	Optional
ClientInitials	The initials of the Client	Optional
DiscountCode	The code of a special offer.	Optional
DiscountAmount	The amount of a special offer (rounded to Euros with 0 decimals).	Optional
NumberOfPets	Amount of pets. Only integer.	Optional
NumberOfBabies	Amount of baby's younger then 3 years old. Only integer.	Optional
TravellInsurance	The amount of travel insurance Winter Sports contribution.	Optional
TravellInsuranceWinterSports	$P = \text{NumberOfAdults} + \text{NumberOfBabies}$ $\text{contribution} = \text{round}((\text{Departure date} - \text{Arrival date} + 1) \times p \times 1.45)$ The amount of travel insurance Winter Sports contribution.	Optional
WebpartnerBookingCode	$P = \text{NumberOfAdults} + \text{NumberOfBabies}$ $\text{contribution} = \text{round}((\text{Departure date} - \text{Arrival date} + 1) \times p \times 1.00)$ A code which the partner can use for his own administration	Optional
WebpartnerAgentCode	The code of a Callcenter agent	Optional
WebpartnerAgentPassword	The password of the Callcenter agent	Optional

2.7.4 XML-RPC answer structure

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<methodResponse>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>CheckBeforeBooking</name>
            <value><string>OK</string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodResponse>
```

2.7.5 XML-RPC answer structure description

<i>Parameter</i>	<i>Description</i>
CheckBeforeBooking	Value = OK

2.7.6 Error codes

<i>Error code</i>	<i>Description</i>
2101	Parameter CryptedString is missing
2102	Planning record with CryptedString 'CS' is unknown
2103	Parameter ClientTitle is missing
2104	ClientTitle ... is unknown
2105	Parameter ClientSurname is missing
2106	ClientSurname is mandatory
2107	Parameter ClientInitials is missing
2108	Parameter ClientStreet is missing
2109	ClientStreet is mandatory
2110	Parameter ClientHouseNumber is missing
2111	ClientHouseNumber is mandatory (integer)
2112	Parameter ClientHouseNumberSuffix is missing
2113	Parameter ClientZipCode is missing
2114	Parameter ClientZipCode is mandatory
2115	Parameter ClientCity is missing
2116	Parameter ClientCity is mandatory
2117	Parameter ClientCountry is missing
2118	ClientCountry ... is unknown
2119	Parameter ClientTelephone is missing
2120	Parameter ClientTelephone is mandatory
2121	Parameter ClientTelephoneMobile is missing
2122	Parameter ClientEmail is missing
2123	ClientEmail ... is invalid
2124	Parameter ClientLanguage is missing
2125	ClientLanguage ... is unknown
2126	Parameter NumberOfAdults is missing
2127	NumberOfAdults is mandatory (integer > 0)
2128	Parameter NumberOfBabies is missing
2129	ToManyPersons ... is more then the allowed ...
2130	Parameter NumberOfPets is missing
2131	ToManyPets ... is more then the allowed ...
2132	ArrivalDate - Today < 14 days, use PlaceOption
2133	Booking is on request, use PlaceOption
2135	Arrivaldate ... is in the past
2136	Parameter RentPrice is missing
2137	RentPrice ... is incorrect, should be
2138	Parameter CancellationInsurance is missing
2139	CancellationInsurance ... is incorrect, should be
2140	Parameter CancellationInsuranceInclGuarantee is missing
2141	CancellationInsuranceInclGuarantee ... is incorrect, should be
2142	CancellationInsuranceInclGuarantee only for customers from The Netherlands and Belgium
2143	CancellationInsuranceInclGuarantee only in combination with CancellationInsurance
2144	Parameter TravelInsurance is missing
2145	TravelInsurance ... is incorrect, should be
2146	TravelInsurance only for customers from The Netherlands and Belgium
2147	Parameter TravelInsuranceWinterSports is missing
2148	TravelInsuranceWinterSports ... is incorrect, should be
2149	TravelInsuranceWinterSports only for customers from The Netherlands and Belgium

2150	TravellInsuranceWinterSports only in combination with TravellInsurance
2160	Parameter DiscountCode is missing
2161	DiscountCode ... is invalid
2162	DiscountCode ... is not valid for this booking: Reason
2163	Parameter DiscountAmount is missing
2164	DiscountAmount ... is invalid, should be ...
2165	Parameter TotalAmount is missing
2166	TotalAmount ... is invalid, should be ...
2167	WebpartnerAgentCode ... / WebpartnetAgentPassword ... is unknown

2.8 Booking.PlaceBooking-v2

2.8.1 General information

With this XML-RPC call you place a definitive booking on a house. Before this call you can check if you have all the correct information to make the XML-RPC call with the “Booking.CheckBeforeBooking-v2” call. If there are no errors, the result is an XML-RPC message with the information about the booking you made. If the ClientEmail you send is already known by Euro Relais, the ClientNumber connected to this email address will be used, if the entered email address is not known, a new ClientNumber will be created. The needed information for making a booking is exactly the same as the “Booking.CheckBeforeBooking-v2” call.

2.8.2 XML-RPC request structure

See Booking.CheckBeforeBooking

2.8.3 XML-RPC request structure description

See Booking.CheckBeforeBooking

2.8.4 XML-RPC answer structure

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<methodResponse>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>BookingNumber</name>
            <value><string> </string></value>
            <name>ClientNumber</name>
            <value><string> </string></value>
            <name>WebPartnerCommissionBase</name>
            <value><string> </string></value>
            <name>WebPartnerBookingCode</name>
            <value><string></string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodResponse>
```

2.8.5 XML-RPC answer structure description

<i>Parameter</i>	<i>Description</i>
BookingNumber	The number of the booking
ClientNumber	The number of the client
PaymentFirstTermDate	The date before which the first payment must be payed.
PaymentFirstTermAmount	The amount of the first payment.
PaymentFirstTermStatus	Value=Payed when PaymentMethod is CreditCard Value=Open otherwise
PaymentSecondTermDate	The date before which the second payment must be payed. Value is empty if there is only 1 payment term.
PaymentSecondTermAmount	The amount of the second payment. Value is empty if there is only 1 payment term.
PaymentSecondTermStatus	Value is empty if there is only 1 payment term otherwise the value=Open
WebPartnerCommissionBase	The amount over which the web partner commission will be calculated

2.8.6 Error codes

<i>Error code</i>	<i>Description</i>
2202	Planning record with CryptedString 'CS' is unknown
2203	Parameter ClientTitle is missing
2204	ClientTitle ... is unknown
2205	Parameter ClientSurname is missing
2206	ClientSurname is mandatory
2207	Parameter ClientInitials is missing
2208	Parameter ClientStreet is missing
2209	ClientStreet is mandatory
2210	Parameter ClientHouseNumber is missing
2211	ClientHouseNumber is mandatory (integer)
2212	Parameter ClientHouseNumberSuffix is missing
2213	Parameter ClientZipCode is missing
2214	Parameter ClientZipCode is mandatory
2215	Parameter ClientCity is missing
2216	Parameter ClientCity is mandatory
2217	Parameter ClientCountry is missing
2218	ClientCountry ... is unknown
2219	Parameter ClientTelephone is missing
2220	Parameter ClientTelephone is mandatory
2221	Parameter ClientTelephoneMobile is missing
2222	Parameter ClientEmail is missing
2223	ClientEmail ... is invalid
2224	Parameter ClientLanguage is missing
2225	ClientLanguage ... is unknown
2226	Parameter NumberOfAdults is missing
2227	NumberOfAdults is mandatory (integer > 0)
2228	Parameter NumberOfBabies is missing
2229	ToManyPersons ... is more then the allowed ...
2230	Parameter NumberOfPets is missing
2231	ToManyPets ... is more then the allowed ...

2232	ArrivalDate - Today < 14 days, use PlaceOption
2233	Booking is on request, use PlaceOption
2235	Arrivaldate ... is in the past
2236	Parameter RentPrice is missing
2237	RentPrice ... is incorrect, should be
2238	Parameter CancellationInsurance is missing
2239	CancellationInsurance ... is incorrect, should be
2240	Parameter CancellationInsuranceInclGuarantee is missing
2241	CancellationInsuranceInclGuarantee ... is incorrect, should be
2242	CancellationInsuranceInclGuarantee only for customers from The Netherlands and Belgium
2243	CancellationInsuranceInclGuarantee only in combination with CancellationInsurance
2244	Parameter TravellInsurance is missing
2245	TravellInsurance ... is incorrect, should be
2246	TravellInsurance only for customers from The Netherlands and Belgium
2247	Parameter TravellInsuranceWinterSports is missing
2248	TravellInsuranceWinterSports ... is incorrect, should be
2249	TravellInsuranceWinterSports only for customers from The Netherlands and Belgium
2250	TravellInsuranceWinterSports only in combination with TravellInsurance
2260	Parameter DiscountCode is missing
2261	DiscountCode ... is invalid
2262	DiscountCode ... is not valid for this booking: Reason
2263	Parameter DiscountAmount is missing
2264	DiscountAmount ... is invalid, should be ...
2265	Parameter TotalAmount is missing
2266	TotalAmount ... is invalid, should be ...
2267	WebpartnerAgentCode ... / WebpartnerAgentPassword ... is unknown

2.9 Booking.CheckBeforeOption-v2

2.9.1 General

With this XML-RPC call you can check if you have all the correct information for making an option. If there are no errors, the result is an XML-RPC message with the value OK.

2.9.2 XML-RPC request structure

```
<?xml version="1.0"?>
<methodCall>
  <methodName>Booking.CheckBeforeOption-v2</methodName>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>WebpartnerCode</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>WebpartnerPassword</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>WebpartnerBookingCode</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>WebpartnerAgentCode</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>WebpartnerAgentPassword</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>Brand</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>CryptedString</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>ClientTitle</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>ClientSurname</name>
            <value><string> </string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodCall>
```

```
<member>
  <name>ClientInitials</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientStreet</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientHouseNumber</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientHouseNumberSuffix</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientZipCode</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientCity</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientCountry</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientTelephone</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientTelephoneMobile</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientEmail</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientLanguage</name>
  <value><string> </string></value>
</member>
<member>
  <name>NumberOfAdults</name>
  <value><string> </string></value>
</member>
<member>
  <name>NumberOfBabies</name>
  <value><string> </string></value>
</member>
<member>
  <name>NumberOfPets</name>
  <value><string> </string></value>
</member>
<member>
```

```
<name>RentPrice</name>
<value><string> </string></value>
</member>
<member>
<name>CancellationInsurance</name>
<value><string> </string></value>
</member>
<member>
<name>CancellationInsuranceInclGuarantee</name>
<value><string> </string></value>
</member>
<member>
<name>TravelInsurance</name>
<value><string> </string></value>
</member>
<member>
<name>TravelInsuranceWinterSports</name>
<value><string> </string></value>
</member>
<member>
<name>DiscountCode</name>
<value><string> </string></value>
</member>
<member>
<name>DiscountAmount</name>
<value><string> </string></value>
</member>
<member>
<name>TotalAmount</name>
<value><string> </string></value>
</member>
</struct>
</value>
</param>
</params>
</methodCall>
```

2.9.3 XML-RPC request structure description

All parameters are mandatory, but the value of some parameters may be empty (optional)

<i>Parameter</i>	<i>Value</i>	<i>Mandatory / Optional</i>
ClientCity	Name of the city	Mandatory
ClientCountry	2 letter country ISO code (NL, BE, DE,FR ,etc).	Mandatory
ClientEmail	E-mail address.	Mandatory
ClientHouseNumber	House number, only integer.	Mandatory
ClientLanguage	What language the client speaks (possible values: NL,FR,DE,EN,IT,ES).	Mandatory
ClientSurname	Family name of the Client	Mandatory
ClientStreet	Name of the street without the house number.	Mandatory
ClientTelephone	Telephone number, don't include country codes.	Mandatory
ClientTitle	Title of the Client (possible values: Mr, Ms,	Mandatory

ClientZipCode	Fam) Zip code, with check for the Netherlands	Mandatory
CryptedString	The Field "CS" from the XML file, or the XML-RPC.	Mandatory
NumberOfAdults	Amount of adults older then 3 years old. Only integer.	Mandatory
RentPrice	The amount of rent to be payed.	Mandatory
TotalAmount	The total amount a customer has to pay.	Mandatory
WebpartnerCode	The partner code you received	Mandatory
WebpartnerPassword	The password you received	Mandatory
NumberOfBabies	Amount of baby's younger then 3 years old. Only integer.	Optional
ClientHouseNumberSuffix	House number suffix.	Optional
ClientTelephoneMobile	Cell phone number.	Optional
NumberOfPets	Amount of pets. Only integer.	Optional
ClientInitials	The initials of the Client	Optional
CancellationInsurance	The amount of cancellation insurance contribution.	Optional
	P = NumberOfAdults + NumberOfBabies if p <= 6 then contribution = round(RentPrice x 0.055) if p > 6 and <= 10 then contribution = round(RentPrice x 0.080) if p > 10 and <= 20 then contribution = round(RentPrice x 0.085) if p > 20 <= 40 then contribution = round(RentPrice x 0.095) if p > 40 then contribution = round(RentPrice x 0.100)	
CancellationInsuranceInclGuarantee	The amount of cancellation insurance contribution inclusive guarantee. Only if ClientCountry = NL or BE	Optional
TravellInsurance	contribution = round(RentPrice x 0.01) The amount of travel insurance Winter Sports contribution.	Optional
TravellInsuranceWinterSports	P = NumberOfAdults + NumberOfBabies contribution = round((Departure date - Arrival date + 1) x p x 1.45) The amount of travel insurance Winter Sports contribution.	Optional
DiscountCode	P = NumberOfAdults + NumberOfBabies contribution = round((Departure date - Arrival date + 1) x p x 1.00) The code of a special offer.	Optional
WebpartnerBookingCode	A code which the partner can use for his own administration	Optional
DiscountAmount	The amount of a special offer (rounded to Euros with 0 decimals).	Optional
WebpartnerAgentCode	The code of a Callcenter agent	Optional
WebpartnerAgentPassword	The password of the Callcenter agent	Optional

2.9.4 XML-RPC answer structure

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<methodResponse>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>CheckBeforeOption</name>
            <value><string>OK</string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodResponse>
```

2.9.5 XML-RPC answer structure description

Parameter	Description
CheckBeforeOption	Value = OK

2.9.6 Error codes

Error code	Description
2301	Parameter CryptedString is missing
2302	Planning record with CryptedString 'CS' is unknown
2303	Parameter ClientTitle is missing
2304	ClientTitle ... is unknown
2305	Parameter ClientSurname is missing
2306	ClientSurname is mandatory
2307	Parameter ClientInitials is missing
2308	Parameter ClientStreet is missing
2309	ClientStreet is mandatory
2310	Parameter ClientHouseNumber is missing
2311	ClientHouseNumber is mandatory (integer)
2312	Parameter ClientHouseNumberSuffix is missing
2313	Parameter ClientZipCode is missing
2314	Parameter ClientZipCode is mandatory
2315	Parameter ClientCity is missing
2316	Parameter ClientCity is mandatory
2317	Parameter ClientCountry is missing
2318	ClientCountry ... is unknown
2319	Parameter ClientTelephone is missing
2320	Parameter ClientTelephone is mandatory
2321	Parameter ClientTelephoneMobile is missing
2322	Parameter ClientEmail is missing
2323	ClientEmail ... is invalid
2324	Parameter ClientLanguage is missing
2325	ClientLanguage ... is unknown

2326	Parameter NumberOfAdults is missing
2327	NumberOfAdults is mandatory (integer > 0)
2328	Parameter NumberOfBabies is missing
2329	ToManyPersons ... is more then the allowed ...
2330	Parameter NumberOfPets is missing
2331	ToManyPets ... is more then the allowed ...
2335	Arrivaldate ... is in the past
2336	Parameter RentPrice is missing
2337	RentPrice ... is incorrect, should be
2338	Parameter CancellationInsurance is missing
2339	CancellationInsurance ... is incorrect, should be
2340	Parameter CancellationInsuranceInclGuarantee is missing
2341	CancellationInsuranceInclGuarantee ... is incorrect, should be
2342	CancellationInsuranceInclGuarantee only for customers from The Netherlands and Belgium
2343	CancellationInsuranceInclGuarantee only in combination with CancellationInsurance
2344	Parameter TravelInsurance is missing
2345	TravelInsurance ... is incorrect, should be
2346	TravelInsurance only for customers from The Netherlands and Belgium
2347	Parameter TravelInsuranceWinterSports is missing
2348	TravelInsuranceWinterSports ... is incorrect, should be
2349	TravelInsuranceWinterSports only for customers from The Netherlands and Belgium
2350	TravelInsuranceWinterSports only in combination with TravelInsurance
2360	Parameter DiscountCode is missing
2361	DiscountCode ... is invalid
2362	DiscountCode ... is not valid for this booking: Reason
2363	Parameter DiscountAmount is missing
2364	DiscountAmount ... is invalid, should be ...
2365	Parameter TotalAmount is missing
2366	TotalAmount ... is invalid, should be ...
2367	WebpartnerAgentCode ... / WebpartnerAgentPassword ... is unknown

2.10 Booking.PlaceOption-v2

2.10.1 General

With this XML-RPC call you place an option on a house. Before this call you can check if you have all the correct information to make the XML-RPC call with the “Booking.CheckBeforeOption-v2” call. If there are no errors, the result is an XML-RPC message with the information about the option you made. If the ClientEmail you send is already known by Euro Relais, the ClientNumber connected to this email address will be used, if the entered email address is not known, a new ClientNumber will be created. The needed information for making an option is exactly the same as the “Booking.CheckBeforeOption-v2” call.

2.10.2 XML-RPC request structure

See Booking.CheckBeforeOption

2.10.3 XML-RPC request structure description

See Booking.CheckBeforeOption

3.10.4 XML-RPC answer structure

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<methodResponse>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>OptionNumber</name>
            <value><string> </string></value>
            <name>ClientNumber</name>
            <value><string> </string></value>
            <name>OptionExpireDate</name>
            <value><string> </string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodResponse>
```

2.10.5 XML-RPC answer structure description

Parameter	Description
OptionNumber	The number of the option
ClientNumber	The number of the client
OptionExpireDate	The date on which the option expires if it is not booked before

2.10.6 Error codes

Error code	Description
2401	Parameter CryptedString is missing
2402	Planning record with CryptedString 'CS' is unknown
2403	Parameter ClientTitle is missing
2404	ClientTitle ... is unknown
2405	Parameter ClientSurname is missing
2406	ClientSurname is mandatory
2407	Parameter ClientInitials is missing
2408	Parameter ClientStreet is missing
2409	ClientStreet is mandatory
2410	Parameter ClientHouseNumber is missing
2411	ClientHouseNumber is mandatory (integer)
2412	Parameter ClientHouseNumberSuffix is missing
2413	Parameter ClientZipCode is missing
2414	Parameter ClientZipCode is mandatory
2415	Parameter ClientCity is missing
2416	Parameter ClientCity is mandatory
2417	Parameter ClientCountry is missing
2418	ClientCountry ... is unknown
2419	Parameter ClientTelephone is missing
2420	Parameter ClientTelephone is mandatory
2421	Parameter ClientTelephoneMobile is missing
2422	Parameter ClientEmail is missing
2423	ClientEmail ... is invalid
2424	Parameter ClientLanguage is missing
2425	ClientLanguage ... is unknown
2426	Parameter NumberOfAdults is missing
2427	NumberOfAdults is mandatory (integer > 0)
2428	Parameter NumberOfBabies is missing
2429	ToManyPersons ... is more then the allowed ...
2430	Parameter NumberOfPets is missing
2431	ToManyPets ... is more then the allowed ...
2435	Arrivaldate ... is in the past
2436	Parameter RentPrice is missing
2437	RentPrice ... is incorrect, should be
2438	Parameter CancellationInsurance is missing
2439	CancellationInsurance ... is incorrect, should be
2440	Parameter CancellationInsuranceInclGuarantee is missing
2441	CancellationInsuranceInclGuarantee ... is incorrect, should be
2442	CancellationInsuranceInclGuarantee only for customers from The Netherlands and Belgium
2443	CancellationInsuranceInclGuarantee only in combination with CancellationInsurance
2444	Parameter TravelInsurance is missing
2445	TravelInsurance ... is incorrect, should be
2446	TravelInsurance only for customers from The Netherlands and Belgium
2447	Parameter TravelInsuranceWinterSports is missing
2448	TravelInsuranceWinterSports ... is incorrect, should be
2449	TravelInsuranceWinterSports only for customers from The Netherlands and Belgium
2450	TravelInsuranceWinterSports only in combination with TravelInsurance
2460	Parameter DiscountCode is missing
2461	DiscountCode ... is invalid
2462	DiscountCode ... is not valid for this booking: Reason
2463	Parameter DiscountAmount is missing
2464	DiscountAmount ... is invalid, should be ...
2465	Parameter TotalAmount is missing

2466	TotalAmount ... is invalid, should be ...
2467	WebpartnerAgentCode ... / WebpartnerAgentPassword ... is unknown

2.11 Booking.DiscountCode

2.11.1 General

With this XML-RPC call you can check the amount of discount you get with a defined DiscountCode. If there are no errors, the result will be the Amount of discount given with the entered DiscountCode.

2.11.2 XML-RPC request structure

```
<?xml version="1.0"?>
<methodCall>
  <methodName>Booking.DiscountCode</methodName>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>WebpartnerCode</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>WebpartnerPassword</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>CryptedString</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>DiscountCode</name>
            <value><string> </string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodCall>
```

2.11.3 XML-RPC request structure description

All parameters are mandatory, but the value of some parameters may be empty (optional)

Parameter	Value	Mandatory / Optional
CryptedString	The Field "CS" from the XML file, or the XML-RPC.	Mandatory
DiscountCode	The code of a special offer.	Mandatory
WebpartnerCode	The partner code you received	Mandatory
WebpartnerPassword	The password you received	Mandatory

2.11.4 XML-RPC answer structure

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<methodResponse>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>Amount</name>
            <value><string> </string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodResponse>
```

2.11.5 XML-RPC answer structure description

<i>Parameter</i>	<i>Description</i>
Amount	The amount of the discount

2.11.6 Error codes

<i>Error code</i>	<i>Description</i>
6101	Parameter CryptedString is missing
6102	Planning record with CryptedString 'CS' is unknown
6103	Parameter DiscountCode is missing
6104	DiscountCode ... is invalid
6105	DiscountCode ... is not valid for this booking: Reason

2.12 Misc.BrochureRequest

2.12.1 General

With this XML-RPC call you send a request to send a Brochure to the given e-mail address.

2.12.2 XML-RPC request structure

```
<?xml version="1.0"?>
<methodCall>
  <methodName>Misc.BrochureRequest</methodName>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>WebpartnerCode</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>WebpartnerPassword</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>ClientTitle</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>ClientSurname</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>ClientInitials</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>ClientStreet</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>ClientHouseNumber</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>ClientHouseNumberSuffix</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>ClientZipCode</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>ClientCity</name>
            <value><string> </string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodCall>
```

```

<member>
  <name>ClientCountry</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientTelephone</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientTelephoneMobile</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientEmail</name>
  <value><string> </string></value>
</member>
<member>
  <name>ClientLanguage</name>
  <value><string> </string></value>
</member>
<member>
  <name>Origin</name>
  <value><string> </string></value>
</member>
</struct>
</value>
</param>
</params>
</methodCall>

```

2.12.3 XML-RPC request structure description

All parameters are mandatory, but the value of some parameters may be empty (optional)

Parameter	Value	Mandatory / Optional
WebpartnerCode	The partner code you received	Mandatory
WebpartnerPassword	The password you received	Mandatory
ClientTitle	Title of the Client (possible values: Mr, Ms, Fam)	Optional
ClientSurname	Family name of the Client	Mandatory
ClientInitials	Client initials	Optional
ClientStreet	Name of the street without the house number.	Mandatory
ClientHouseNumber	House number, only integer.	Optional
ClientHouseNumberSuffix	Housenumber suffix	Optional
ClientZipCode	Zip code, with check for the Netherlands	Mandatory
ClientCity	Name of the city	Mandatory
ClientCountry	2 letter country ISO code (NL, BE, DE,FR ,etc).	Mandatory
ClientTelephone	Telephone number, don't include country codes.	Optional
ClientTelephoneMobile	Cell phone number	Optional
ClientEmail	E-mail address.	Mandatory
ClientLanguage	What language the client speaks (possible values: NL,FR,DE,EN,IT,ES).	Mandatory
Origin	Origin of the client. Possible values: 10 = through an advertisement 20 = through family of friends 30 = through a link from another website 40 = through a search engine (like Google/MSN etc.) 50 = through House Vision on television	Optional

2.12.4 XML-RPC answer structure

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<methodResponse>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>BrochureRequest</name>
            <value><string>OK</string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodResponse>
```

2.12.5 XML-RPC answer structure description

Parameter	Description
BrochureRequest	Value = OK

2.12.6 Error codes

Error code	Description
4104	ClientTitle ... is unknown
4105	Parameter ClientSurname is missing
4106	ClientSurname is mandatory
4108	Parameter ClientStreet is missing
4109	ClientStreet is mandatory
4113	Parameter ClientZipCode is missing
4114	Parameter ClientZipCode is mandatory
4115	Parameter ClientCity is missing
4116	Parameter ClientCity is mandatory
4117	Parameter ClientCountry is missing
4118	ClientCountry ... is unknown
4122	Parameter ClientEmail is missing
4123	ClientEmail ... is invalid
4124	Parameter ClientLanguage is missing
4125	ClientLanguage ... is unknown
4127	Origin ... is unknown
4128	To many Brochure Requests

2.13 Misc.PotentialHouseOwnerSignUp

2.13.1 General

With this XML-RPC call you sign up a potential house owner.

2.13.2 XML-RPC request structure

```
<?xml version="1.0"?>
<methodCall>
  <methodName>Misc.BrochureRequest</methodName>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>WebpartnerCode</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>WebpartnerPassword</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>PotentialHouseOwnerTitle</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>PotentialHouseOwnerSurname</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>PotentialHouseOwnerInitials</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>PotentialHouseOwnerStreet</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>PotentialHouseOwnerNumber</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>PotentialHouseOwnerHouseNumberSuffix</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>PotentialHouseOwnerZipCode</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>PotentialHouseOwnerCity</name>
            <value><string> </string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodCall>
```



```
<member>
  <name>PotentialHouseOwnerCountry</name>
  <value><string> </string></value>
</member>
<member>
  <name>PotentialHouseOwnerTelephone</name>
  <value><string> </string></value>
</member>
<member>
  <name>PotentialHouseOwnerTelephoneMobile</name>
  <value><string> </string></value>
</member>
<member>
  <name>PotentialHouseOwnerEmail</name>
  <value><string> </string></value>
</member>
<member>
  <name>PotentialHouseOwnerLanguage</name>
  <value><string> </string></value>
</member>
<member>
  <name>PotentialHouseOwnerWebsite</name>
  <value><string> </string></value>
</member>
<member>
  <name>Origin</name>
  <value><string> </string></value>
</member>
<member>
  <name>HouseCountry</name>
  <value><string> </string></value>
</member>
<member>
  <name>HouseRegion</name>
  <value><string> </string></value>
</member>
<member>
  <name>HouseCity</name>
  <value><string> </string></value>
</member>
<member>
  <name>HouseType</name>
  <value><string> </string></value>
</member>
<member>
  <name>HouseSituation</name>
  <value><string> </string></value>
</member>
<member>
  <name>HouseNumberOfPersons</name>
  <value><string> </string></value>
</member>
</struct>
</value>
</param>
</params>
</methodCall>
```


2.13.3 XML-RPC request structure description

All parameters are mandatory, but the value of some parameters may be empty (optional)

Parameter	Value	Mandatory / Optional
WebpartnerCode	The partner code you received	Mandatory
WebpartnerPassword	The password you received	Mandatory
PotentialHouseOwnerTitle	Title of the Client (possible values: Mr, Ms, Fam)	Optional
PotentialHouseOwnerSurname	Family name of the Client	Mandatory
PotentialHouseOwnerInitials	Client initials	Optional
PotentialHouseOwnerStreet	Name of the street without the house number.	Mandatory
PotentialHouseOwnerHouseNumber	House number, only integer.	Optional
PotentialHouseOwnerHouseNumberSuffix	Housenumber suffix	Optional
PotentialHouseOwnerZipCode	Zip code, with check for the Netherlands	Mandatory
PotentialHouseOwnerCity	Name of the city	Mandatory
PotentialHouseOwnerCountry	2 letter country ISO code (NL, BE, DE,FR ,etc).	Mandatory
PotentialHouseOwnerTelephone	Telephone number, don't include country codes.	Optional
PotentialHouseOwnerTelephoneMobile	Cell phone number	Optional
PotentialHouseOwnerEmail	E-mail address.	Mandatory
PotentialHouseOwnerLanguage	What language the client speaks (possible values: NL,FR,DE,EN,IT,ES).	Mandatory
PotentialHouseOwnerWebsite	The URL of the houseowners website	Optional
Origin	Origin of the houseowner. Possible values: 10 = through an advertisement 20 = through family of friends 30 = through a link from another website 40 = through a search engine (like Google/MSN etc.) 50 = through House Vision on television 99 = other	Optional
HouseCountry	The country of the holidayhouse. Values: see regions.xml in the xml documentation	Mandatory
HouseRegion	The region of the holidayhouse. Values: see regions.xml in the xml documentation	Mandatory
HouseCity	The city of the holidayhouse	Mandatory
HouseType	Type of the holidayhouse. Possible values: 10 = City apartment 20 = Castle 30 = Cottage 40 = Mansion 50 = Villa 60 = Chalet 70 = Farmhouse 80 = Extraordinary accommodation 90 = Boat 100 = Holiday home	Mandatory
HouseSituation	The situation of the holidayhouse. Possible values: 310 = Tranquil surroundings 320 = Close to/in wood	Mandatory

	330 = On a farm 340 = On a working farm 350 = On an island 360 = In the countryside 370 = In the mountains 380 = In the city 390 = Holiday Park	
HouseNumberOfPersons	How many persons can stay in the holidayhouse.	Mandatory

2.13.4 XML-RPC answer structure

```

<?xml version="1.0" encoding="iso-8859-1" ?>
<methodResponse>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>PotentialHouseOwnerSignUp</name>
            <value><string>OK</string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodResponse>

```

2.13.5 XML-RPC answer structure description

<i>Parameter</i>	<i>Description</i>
PotentialHouseOwnerSignUp	Value = OK

2.13.6 Error codes

Error code	Description
4204	PotentialHouseOwnerTitle ... is unknown
4205	Parameter PotentialHouseOwnerSurname is missing
4206	PotentialHouseOwnerSurname is mandatory
4208	Parameter PotentialHouseOwnerStreet is missing
4209	PotentialHouseOwnerStreet is mandatory
4213	Parameter PotentialHouseOwnerZipCode is missing
4214	Parameter PotentialHouseOwnerZipCode is mandatory
4215	Parameter PotentialHouseOwnerCity is missing
4216	Parameter PotentialHouseOwnerCity is mandatory
4217	Parameter PotentialHouseOwnerCountry is missing
4218	PotentialHouseOwnerCountry ... is unknown
4219	Parameter PotentialHouseOwnerTelephone is missing
4220	Parameter PotentialHouseOwnerTelephone is mandatory
4222	Parameter PotentialHouseOwnerEmail is missing
4223	PotentialHouseOwnerEmail ... is invalid
4224	Parameter PotentialHouseOwnerLanguage is missing
4225	PotentialHouseOwnerLanguage ... is unknown
4227	Origin ... is unknown
4228	Potential House Owner already exist
4230	Parameter HouseCountry is missing
4231	HouseCountry is unknown
4232	Parameter HouseRegion is missing
4233	HouseRegion is unknown
4234	Parameter HouseCity is missing
4235	HouseCity is mandatory
4236	Parameter HouseType is missing
4237	HouseType is unknown
4238	Parameter HouseSituation is missing
4239	HouseSituation is unknown
4240	Parameter HouseNumberOfPersons is missing
4241	HouseNumberOfPersons is mandatory

3. Examples

3.1 Introduction

This chapter is created to give an example of how to implement the XML-RPC files we have created. The examples give a good idea of what logic steps need to be taken for searching houses in our database. As we said in an earlier paragraph, the following scheme is needed for a simple search and book procedure:

- Display house details with the “Search.DetailsOfOneHouse” call;
- Display house planning with the “Search.PlanningOfOneHouse-v2” call;
- Create your own bookings form with the “Booking.CheckBeforeBooking-v2”, “Booking.PlaceBooking-v2”, “Booking.CheckBeforeOption-v2”, “Booking.PlaceOption-v2” calls, or link to our book engine.

We will give examples of how to implement the procedures for searching houses and showing the results (example 1), how to make a house presentation and show the planning (example 2), and how to create a booking form (example 3).

Schematic view of the examples and field connections of the XML-RPC calls:

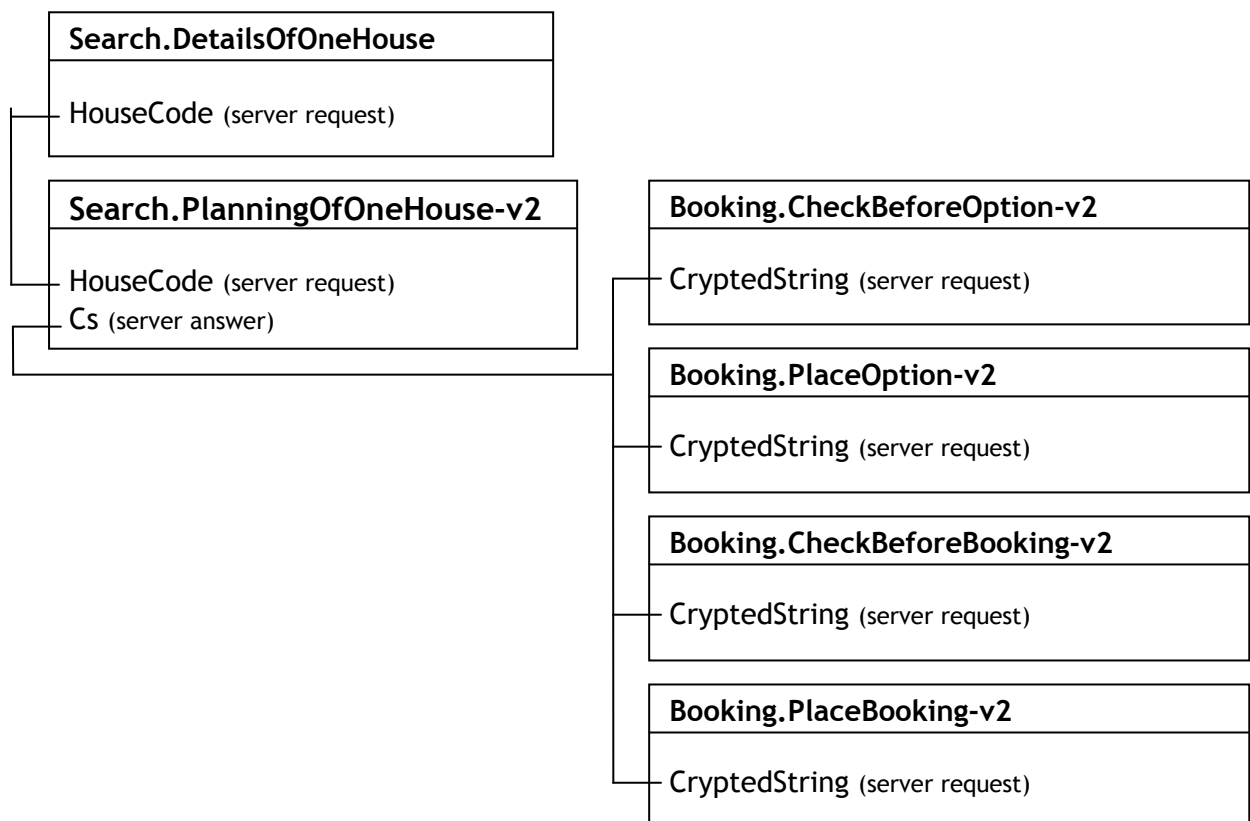


Figure 1: schematic view of the XML-RPC field connections

3.2 Example 2: House presentation

A simple explanation how to setup a house presentation page, and show the planning for the selected house.

Calls needed:

Search.DetailsOfOneHouse
Search.PlanningOfOneHouse-v2

Looking at the previous example, we are going to look at the house details of XX-1000-01.

This is the request we do:

```
<?xml version="1.0"?>
<methodCall>
  <methodName>Search.DetailsOfOneHouse</methodName>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>WebpartnerCode</name>
            <value><string>Partnercode</string></value>
          </member>
          <member>
            <name>WebpartnerPassword</name>
            <value><string>Password</string></value>
          </member>
          <member>
            <name>HouseCode</name>
            <value><string>DEMO-1000-01</string></value>
          </member>
          <member>
            <name>FieldList</name>
            <value><string>
Code,Name,FromNumberPersons,ToNumberPersons,NumberStars,ZipPostalCode,City,Region,Country,IntroductionEN,
DescriptionEN,MarketCity,MarketDistance,RestaurantCity,RestaurantDistance,RailwayCity,RailwayDistance,Air
portCity,AirportDistance,TouristInfoCity,TouristInfoDistance,ShopsCity,ShopsDistance,BathTowelsEN,Deposit
EN,EnergyEN,PetsEN,KitchenLinenEN,BedLinenEN,EnvironmentEN,CleaningEN,TouristTaxEN,Pictures72dpi,
PropertiesDefinitionEN</string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodCall>
```

As you can see, We need the House code from the previous example. So you need to request the “Code” field from the previous example. All fields in the “FieldList” are the same fields as are present in the house.xml file, except for the “PropertiesDefinitionEN” field. When requesting this field, the answer will contain a comma separated list with the definitions of the properties in the selected language.

PropertiesDefinitionNL = Dutch language
PropertiesDefinitionFR = French language
PropertiesDefinitionDE = German language
PropertiesDefinitionEN = English language
PropertiesDefinitionIT = Italian language
PropertiesDefinitionES = Spanish language

Because the answer from the server is dependant of the FieldList value, the answer will be like this:

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<methodResponse>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>Code</name>
            <value><string>XX-1000-01</string></value>
            <name>Name</name>
            <value><string>Demo house</string></value>
            <name>FromNumberPersons</name>
            <value><string>4</string></value>
            <name>ToNumberPersons</name>
            <value><string>5</string></value>
            <name>NumberStars</name>
            <value><string>3</string></value>
            <name>ZipPostalCode</name>
            <value><string>1111</string></value>
            <name>City</name>
            <value><string>Demo city</string></value>
            <name>Region</name>
            <value><string>nrdb</string></value>
          </member>
          <name>Country</name>
          <value><string>NL</string></value>
          <name>IntroductionNL</name>
          <value><string>Introduction about the house</string></value>
          <name>DescriptionNL</name>
          <value><string>Situatied close to the centre of the pretty village of Rosedale Abbey, in the
heart of the North York Moors National Park. etc.....</string></value>
          <name>MarketCity</name>
          <value><string>DEMO</string></value>
          <name>MarketDistance</name>
          <value><string>5 KM</string></value>
          <name>RestaurantCity</name>
          <value><string>DEMO</string></value>
          <name>RestaurantDistance</name>
          <value><string>5 KM</string></value>
          <name>RailwayCity</name>
          <value><string>DEMO</string></value>
          <name>RailwayDistance</name>
          <value><string>5 KM</string></value>
          <name>AirportCity</name>
          <value><string>DEMO</string></value>
          <name>AirportDistance</name>
          <value><string>5 KM</string></value>
          <name>TouristInfoCity</name>
          <value><string>DEMO</string></value>
          <name>TouristInfoDistance</name>
          <value><string>5 KM</string></value>
          <name>ShopsCity</name>
          <value><string>DEMO</string></value>
          <name>ShopsDistance</name>
          <value><string>5 KM</string></value>
          <name>BathTowelsNL</name>
          <value><string>No</string></value>
          <name>DepositNL</name>
          <value><string>0</string></value>
          <name>EnergyNL</name>
          <value><string>No</string></value>
          <name>PetsNL</name>
          <value><string>No</string></value>
          <name>KitchenLinenNL</name>
          <value><string>No</string></value>
          <name>BedLinenNL</name>
          <value><string>No</string></value>
          <name>EnvironmentNL</name>
          <value><string>No</string></value>
          <name>CleaningNL</name>
          <value><string>No</string></value>
          <name>TouristTaxNL</name>
          <value><string>No</string></value>
          <name>PropertiesDefinitionEN</name>
        </struct>
      </value>
    </param>
  </params>
</methodResponse>
```

```
<value><string>Holiday home,Detached,Jetty of pier,Table tennis,Boules
game,Stables,Waterski,Canoeing,Rowing,Sailing,Surfing,Diving,Marina,Cross country,Skating rink,Artificial
ski slope,City centre < 0 km.</string><value>
  <name>Pictures72dpi</name>
  <value><string>d/de/xx-1000-01-a-huis.jpg,d/de/xx-1000-01-i-hal.jpg,d/de/xx-1000-01-j-
zitkamer.jpg,d/de/xx-1000-01-k-keuken.jpg,d/de/xx-1000-01-l-slk.jpg,d/de/xx-1000-01-o-sauna.jpg,d/de/xx-
1000-01-o-terras.jpg,d/de/xx-1000-01-o-zwembad.jpg,d/de/xx-1000-01-p-omgeving.jpg </string></value>
    </member>
  </struct>
</value>
</param>
</params>
</methodResponse>
```

With this answer you can create a screen like the screen on the next page. To use the pictures shown in the screen, you have to comma separate the Pictures72dpi value, and add an URL in front of the values. The URL you need to put in front of the values is dependant of what size of picture you want to use.

For pictures with a dimension of 128 pixels x 85 pixels (thumbnail):

“http://www2.eurorelais.net/fotos/72dpiklein/”

Example:

“http://www2.eurorelais.net/fotos/72dpiklein/d/de/XX-1000-01-a-huis.jpg”

For pictures with a dimension of 255 pixels x 170 pixels (big picture):

“http://www2.eurorelais.net/fotos/72dpi/”

Example:

“http://www2.eurorelais.net/foto/72dpi/d/de/XX-1000-01-a-huis/jpg”



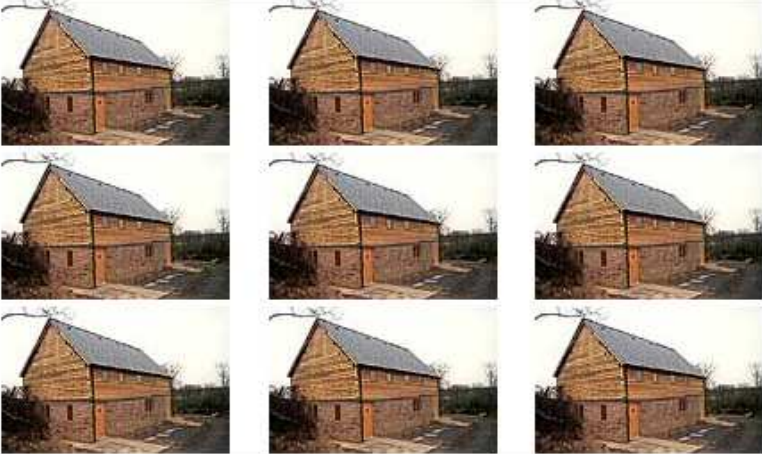
Main information			
Housecode:	DEMO-1000-01	City:	DEMO CTTY
Number of Persons:	4 / 5	Region:	nrdb
***		Country:	NL
Photo's			
			
House description			
Situatied close to the centre of the pretty village of Rosedale Abbey, in the heart of the North York Moors National Park. etc.....			
Surroundings			
Airport:	DEMO - 5 KM		
Restaurant:	DEMO - 5 KM		
Railway station:	DEMO - 5 KM		
Open-air market:	DEMO - 5 KM		
Tourist Information Office:	DEMO - 5 KM		
Shops:	DEMO - 5 KM		
Extra costs			
Deposit:	0	Bed linen:	No
Water/Energy:	No	Bath towels:	No
Pets:	No	Kitchen towels:	No
Cleaning:	No	Environmental tax:	No
Tourist tax:	No		
House facilities			
- Holiday home	- Detached	- Jetty of pier	- Table tennis
- Boules game	- Stables	- Waterski	- Canoeing
- Rowing	- Sailing	- Surfing	- Diving
- Marina	- Cross country	- Skating rink	- Artificial ski slope
- City centre < 0 km			
book now >>			

Figure 4: house details

After clicking on the “book now” button you have two options, the first option is to show all the records in the planning table, then you don’t need to put a value in the “Period” field. The second option is to select a period before you show a planning table. We are going to use the second option, and select a period before showing the planning. Here is a simple example:

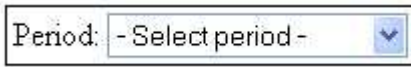


Figure 5: period selection

In this dropdown list you insert the values:

1w = 1 week
2w = 2 weeks
3w = 3 weeks
wk = weekend
lw = long weekend
mw = mid week

if you choose “1w” you will get the planning of 1 week.

To show the results we have to do a call to the server, this call will look like:

```
<?xml version="1.0"?>
<methodCall>
  <methodName>Search.PlanningOfOneHouse-v2</methodName>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>WebpartnerCode</name>
            <value><string>Partnercode</string></value>
          </member>
          <member>
            <name>WebpartnerPassword</name>
            <value><string>Password</string></value>
          </member>
          <member>
            <name>HouseCode</name>
            <value><string>XX-1000-01</string></value>
          </member>
          <member>
            <name>Period</name>
            <value><string>lw</string></value>
          </member>
          <member>
            <name>FromDate</name>
            <value><string></string></value>
          </member>
          <member>
            <name>UntilDate</name>
            <value><string></string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodCall>
```

We leave the FromDate and Until date empty, because we want to show the complete planning of a house for 1 week.

The answer you will get from the server will be like this:

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<methodResponse>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>period</name>
            <value><string>1w</string></value>
            <name>arrival-date</name>
            <value><string>05-05-2006</string></value>
            <name>arrival-time-from</name>
            <value><string>1500</string></value>
            <name>arrival-time-until</name>
            <value><string>1800</string></value>
            <name>departure-date</name>
            <value><string>12-05-2006</string></value>
            <name>departure-time</name>
            <value><string>1000</string></value>
            <name>booking-on-request</name>
            <value><string>No</string></value>
            <name>rent-price</name>
            <value><string>443</string></value>
            <name>rent-price-excl-discount</name>
            <value><string>443</string></value>
            <name>cs</name>
            <value><string>ldkjbeaaknWhcakjfxMVficlbeIclPt</string></value>
          </member>
          <member>
            <name>period</name>
            <value><string>1w</string></value>
            <name>arrival-date</name>
            <value><string>12-05-2006</string></value>
            <name>arrival-time-from</name>
            <value><string>1500</string></value>
            <name>arrival-time-until</name>
            <value><string>1800</string></value>
            <name>departure-date</name>
            <value><string>19-05-2006</string></value>
            <name>departure-time</name>
            <value><string>1000</string></value>
            <name>booking-on-request</name>
            <value><string>No</string></value>
            <name>rent-price</name>
            <value><string>388</string></value>
            <name>rent-price-excl-discount</name>
            <value><string>388</string></value>
            <name>cs</name>
            <value><string>ldkjbeaaknWacakjfxMVeficlbclPt </string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodResponse>
```

Each <member> is a separate date in the planning table.

With this answer you can make a screen like this:

05/05 - 12/05 € 343	12/05 - 19/05 € 338	19/05 - 26/05 € 388	26/05 - 02/06 € 388
------------------------	------------------------	------------------------	------------------------

Figure 6: date selection

With the previous call you have the required information to make a booking. You can choose to create your own booking page, then the next example is a good reference for making one. Or you can redirect to the booking page we provide. If you choose to link to our booking page, we refer to our search engine manual.

3.4 Example 3: making a booking

An explanation how to make your own booking page.
Keep in mind that making your own booking page is not simple.

Calls needed:

Booking.CheckBeforeBooking-v2
Booking.PlaceBooking-v2
Booking.CheckBeforeOption-v2
Booking.PlaceOption-v2

As you can see there are 2 types of booking calls. You have the ‘Booking’, and ‘Option’ version.
If the arrival date is less than 2 weeks from current date, the booking should be made as an option. If you try to make a booking that should be an option you will receive an error from the server. So this check must be done by you.

Checking if the information the client entered is consistent with the information needed in the fields is a thing you have to program yourself.

Looking at the previous examples, we are going to make a booking on house code XX-1000-01.
The 2 calls are completely the same, only the answer from the server is different.
Because you need to check if all information is correct, we first make the “Booking.CheckBeforeBooking-v2” call:

```
<?xml version="1.0"?>
<methodCall>
  <methodName>Booking.CheckBeforeBooking-v2</methodName>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>WebpartnerCode</name>
            <value><string>PartnerCode</string></value>
          </member>
          <member>
            <name>WebpartnerPassword</name>
            <value><string>Password</string></value>
          </member>
          <member>
            <name>WebpartnerBookingCode</name>
            <value><string> </string></value>
          </member>
          <member>
            <name>CryptedString</name>
            <value><string>ldkjbeaeknWacakjfMxMVeficlbclkPt </string></value>
          </member>
          <member>
            <name>ClientTitle</name>
            <value><string>Mr.</string></value>
          </member>
          <member>
            <name>ClientSurname</name>
            <value><string>Test</string></value>
          </member>
          <member>
            <name>ClientInitials</name>
            <value><string>T</string></value>
          </member>
          <member>
            <name>ClientStreet</name>
            <value><string>teststreet</string></value>
          </member>
          <member>
            <name>ClientHouseNumber</name>
            <value><string>123</string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodCall>
```



```
<member>
  <name>ClientHouseNumberSuffix</name>
  <value><string></string></value>
</member>
<member>
  <name>ClientZipCode</name>
  <value><string>5454aa</string></value>
</member>
<member>
  <name>ClientCity</name>
  <value><string>testcity</string></value>
</member>
<member>
  <name>ClientCountry</name>
  <value><string>NL</string></value>
</member>
<member>
  <name>ClientTelephone</name>
  <value><string>0123456789</string></value>
</member>
<member>
  <name>ClientTelephoneMobile</name>
  <value><string>0606060606</string></value>
</member>
<member>
  <name>ClientEmail</name>
  <value><string>test@demo.com</string></value>
</member>
<member>
  <name>ClientLanguage</name>
  <value><string>NL</string></value>
</member>
<member>
  <name>NumberOfAdults</name>
  <value><string>4</string></value>
</member>
<member>
  <name>NumberOfBabies</name>
  <value><string>0</string></value>
</member>
<member>
  <name>NumberOfPets</name>
  <value><string>0</string></value>
</member>
<member>
  <name>RentPrice</name>
  <value><string>388</string></value>
</member>
<member>
  <name>CancellationInsurance</name>
  <value><string></string></value>
</member>
<member>
  <name>CancellationInsuranceInclGuarantee</name>
  <value><string> </string></value>
</member>
<member>
  <name>TravelInsurance</name>
  <value><string> </string></value>
</member>
<member>
  <name>TravelInsuranceWinterSports</name>
  <value><string> </string></value>
</member>
<member>
  <name>DiscountCode</name>
  <value><string> </string></value>
</member>
<member>
  <name>DiscountAmount</name>
  <value><string> </string></value>
</member>
<member>
  <name>TotalAmount</name>
  <value><string> </string></value>
</member>
</member>
```

```

    <name>PaymentMethod</name>
    <value><string> </string></value>
  </member>
  <member>
    <name>PaymentDetails</name>
    <value><string> </string></value>
  </member>
</struct>
</value>
</param>
</params>
</methodCall>

```

To make sure you get all the information from the client, you need to make a form where the client must fill in his personal information. The form could look like this screen:


General information		
 <p>***</p>	Holiday house: Demo house City: Demo city House code: DEMO-1000-01 Length of stay: 1 Week Arrival: Friday 01-01-2006 15:00 hour and 18:00 hour Departure: Friday 08-01-2006 before 10:00 hour Rental price: € 388	
	Additional Remarks	
	As you are booking within 14 days of arrival at your holiday home, we are obliged to inform the home owner of your arrival prior to confirmation. On confirmation of the reservation you will receive an option number and further information as to the next steps to be made regarding the booking.	
	Personal information	
	Initials/Surname: <input type="text"/> Mr. <input type="text"/> <input type="text"/> Street: <input type="text"/> House number /Apartment number: <input type="text"/> <input type="text"/> Postal code / City: <input type="text"/> <input type="text"/> (e.g.) Country: <input type="text"/> Netherlands <input type="text"/> Telephone: <input type="text"/> (without countrycode, just numbers) Mobile: <input type="text"/> (without countrycode, just numbers) E-mail: <input type="text"/>	
	Promotion code (optional)	
	Promotion code: <input type="text"/> (optional)	
Number of persons		
Number of persons: <input type="text"/> 2 <input type="text"/> Maximum: 5 Number of children: <input type="text"/> 0 <input type="text"/> (Children under 3 years old) Number of pets: <input type="text"/> Pets are not allowed		
Extras		
Standard Cancellation Insurance <input checked="" type="checkbox"/> Costs: € <input type="text"/> 22 ? + Extended Cancellation Insurance * <input type="checkbox"/> Costs: € <input type="text"/> 0 ? Travel Insurance * <input type="checkbox"/> Costs: € <input type="text"/> 0 ? + Winter sports insurance * <input type="checkbox"/> Costs: € <input type="text"/> 0 ? * = Only available to residents of the Netherlands and Belgium		
<input type="button" value="Continue"/>		

Figure 7: booking form (step 1)

The part “Additional Remarks” is made for option bookings. So you can tell the client he is making an option booking, when making a normal booking, you can leave out the “Additional Remarks”. After filling in all the information, the only thing you are missing is the payment method. We chose to put this on the next page. On this next page you can show all the information the client submitted, and then ask for the payment method, at the end ask the client to accept the rental conditions. This page could look like:


General information																			
 <p>***</p>	<table> <tr> <td>Holiday house:</td> <td>Demo house</td> </tr> <tr> <td>City:</td> <td>Demo city</td> </tr> <tr> <td>House code:</td> <td>DEMO-1000-01</td> </tr> <tr> <td>Length of stay:</td> <td>1 Week</td> </tr> <tr> <td>Arrival:</td> <td>Friday 01-01-2006 16:00 hour and 19:00 hour</td> </tr> <tr> <td>Departure:</td> <td>Friday 08-01-2006 before 10:00 hour</td> </tr> <tr> <td>Rental price:</td> <td>€ 388</td> </tr> </table>	Holiday house:	Demo house	City:	Demo city	House code:	DEMO-1000-01	Length of stay:	1 Week	Arrival:	Friday 01-01-2006 16:00 hour and 19:00 hour	Departure:	Friday 08-01-2006 before 10:00 hour	Rental price:	€ 388				
	Holiday house:	Demo house																	
	City:	Demo city																	
	House code:	DEMO-1000-01																	
	Length of stay:	1 Week																	
	Arrival:	Friday 01-01-2006 16:00 hour and 19:00 hour																	
	Departure:	Friday 08-01-2006 before 10:00 hour																	
Rental price:	€ 388																		
Additional Remarks																			
<p>As you are booking within 14 days of arrival at your holiday home, we are obliged to inform the home owner of your arrival prior to confirmation. On confirmation of the reservation you will receive an option number and further information as to the next steps to be made regarding the booking.</p>																			
Personal information																			
<table> <tr> <td>Surname:</td> <td>Mr. T Test</td> </tr> <tr> <td>Street:</td> <td>test 12</td> </tr> <tr> <td>Postal code / City:</td> <td>nl-5454aa test</td> </tr> <tr> <td>Telephone:</td> <td>0413333333</td> </tr> <tr> <td>Mobile:</td> <td>0633333333</td> </tr> <tr> <td>E-mail:</td> <td>test@test.nl</td> </tr> </table>		Surname:	Mr. T Test	Street:	test 12	Postal code / City:	nl-5454aa test	Telephone:	0413333333	Mobile:	0633333333	E-mail:	test@test.nl						
Surname:	Mr. T Test																		
Street:	test 12																		
Postal code / City:	nl-5454aa test																		
Telephone:	0413333333																		
Mobile:	0633333333																		
E-mail:	test@test.nl																		
Number of persons																			
<table> <tr> <td>Number of persons:</td> <td>2</td> </tr> <tr> <td>Number of children:</td> <td>0</td> </tr> <tr> <td>Number of pets:</td> <td>0</td> </tr> </table>		Number of persons:	2	Number of children:	0	Number of pets:	0												
Number of persons:	2																		
Number of children:	0																		
Number of pets:	0																		
Costs																			
<table> <tr> <td>Rental price:</td> <td>€ 388</td> </tr> <tr> <td>Standard Cancellation Insurance:</td> <td>€ 25</td> </tr> <tr> <td>Extended Cancellation Insurance:</td> <td>-</td> </tr> <tr> <td>Travel Insurance:</td> <td>-</td> </tr> <tr> <td>Winter sports insurance:</td> <td>-</td> </tr> <tr> <td>Total amount:</td> <td>€ 413</td> </tr> </table>		Rental price:	€ 388	Standard Cancellation Insurance:	€ 25	Extended Cancellation Insurance:	-	Travel Insurance:	-	Winter sports insurance:	-	Total amount:	€ 413						
Rental price:	€ 388																		
Standard Cancellation Insurance:	€ 25																		
Extended Cancellation Insurance:	-																		
Travel Insurance:	-																		
Winter sports insurance:	-																		
Total amount:	€ 413																		
Extra Costs																			
<table> <tr> <td>Deposit:</td> <td>0</td> </tr> <tr> <td>Water/Energy:</td> <td>No</td> </tr> <tr> <td>Pets:</td> <td>No</td> </tr> <tr> <td>Cleaning:</td> <td>No</td> </tr> <tr> <td>Tourist tax:</td> <td>No</td> </tr> <tr> <td>Bed linen:</td> <td>No</td> </tr> <tr> <td>Bath towels:</td> <td>No</td> </tr> <tr> <td>Kitchen towels:</td> <td>No</td> </tr> <tr> <td>Environmental tax:</td> <td>No</td> </tr> </table>		Deposit:	0	Water/Energy:	No	Pets:	No	Cleaning:	No	Tourist tax:	No	Bed linen:	No	Bath towels:	No	Kitchen towels:	No	Environmental tax:	No
Deposit:	0																		
Water/Energy:	No																		
Pets:	No																		
Cleaning:	No																		
Tourist tax:	No																		
Bed linen:	No																		
Bath towels:	No																		
Kitchen towels:	No																		
Environmental tax:	No																		
Select your method of payment																			
<table> <tr> <td><input checked="" type="radio"/> Credit card (Visa)</td> <td>Credit card number:</td> <td><input type="text"/></td> </tr> <tr> <td><input type="radio"/> Credit card (Mastercard)</td> <td>Expiration Date:</td> <td><input type="text" value="YYMM"/></td> </tr> <tr> <td><input type="radio"/> Direct debit (Dutch bank account required) (Postbank (NL only))</td> <td>Owner name:</td> <td><input type="text"/></td> </tr> <tr> <td><input type="radio"/> Direct debit (Dutch bank account required) (Other bank)</td> <td>CVC code:</td> <td><input type="text"/></td> </tr> </table>		<input checked="" type="radio"/> Credit card (Visa)	Credit card number:	<input type="text"/>	<input type="radio"/> Credit card (Mastercard)	Expiration Date:	<input type="text" value="YYMM"/>	<input type="radio"/> Direct debit (Dutch bank account required) (Postbank (NL only))	Owner name:	<input type="text"/>	<input type="radio"/> Direct debit (Dutch bank account required) (Other bank)	CVC code:	<input type="text"/>						
<input checked="" type="radio"/> Credit card (Visa)	Credit card number:	<input type="text"/>																	
<input type="radio"/> Credit card (Mastercard)	Expiration Date:	<input type="text" value="YYMM"/>																	
<input type="radio"/> Direct debit (Dutch bank account required) (Postbank (NL only))	Owner name:	<input type="text"/>																	
<input type="radio"/> Direct debit (Dutch bank account required) (Other bank)	CVC code:	<input type="text"/>																	
Acceptance of the rental conditions is mandatory.																			
<input type="checkbox"/> I accept the rental conditions																			
<input type="button" value="Confirm your booking"/>																			

Figure 8: booking form (step 2)

After clicking the “Confirm your booking”, the XML-RPC call “Booking.CheckBeforeBooking-v2” or “Booking.CheckBeforeOption-v2” will be executed.

The answer will look like this:

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<methodResponse>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>CheckBeforeBooking-v2</name>
            <value><string>OK</string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodResponse>
```

When the answer is like above you automatically can call the “Booking.PlaceBooking-v2” or “Booking.PlaceOption-v2” function. The call has the same layout as the “Booking.CheckBeforeBooking-v2” or “Booking.CheckBeforeOption-v2”.

The answer of the “Booking.PlaceBooking-v2” call will look like :

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<methodResponse>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>BookingNumber</name>
            <value><string>4119157</string></value>
            <name>ClientNumber</name>
            <value><string>1608977</string></value>
            <name>PaymentFirstTermDate</name>
            <value><string>01-01-2006</string></value>
            <name>PaymentFirstTermAmount</name>
            <value><string>200</string></value>
            <name>PaymentFirstTermStatus</name>
            <value><string>Open</string></value>
            <name>PaymentSecondTermDate</name>
            <value><string>21-01-2001</string></value>
            <name>PaymentSecondTermAmount</name>
            <value><string>188</string></value>
            <name>PaymentSecondTermStatus</name>
            <value><string>Open</string></value>
            <name>WebPartnerCommissionBase</name>
            <value><string></string></value>
            <name>WebPartnerBookingCode</name>
            <value><string></string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodResponse>
```


With this answer from the server, you can make a screen where you can show the information the client entered. The screen can look like this:


General information		
 <p>***</p>	Holiday house:	Demo house
	City:	Demo city
	House code:	DEMO-1000-01
	Length of stay:	1 Week
	Arrival:	Friday 01-01-2006 16:00 hour and 19:00 hour
	Departure:	Friday 06-01-2006 before 10:00 hour
	Rental price:	€ 388
Thank you for booking with Euro Relais!		
Below you will find further information on your booking and what happens next. We advise you to print these pages so that you always have your booking number and customer number close at hand.		
Bookingsgegevens		
Your Booking Number:	4119251	
Your Customer Number:	1635831	
Full sum:	€ 413,00	
Full sum pay before:	01-01-2006	
Personal information		
Surname:	Mr. T Test	
Street:	Test 12	
Postal code / City:	nl-5454aa Test	
Telephone:	0413333333	
Mobile:	0633333333	
E-mail:	test@test.nl	
Number of persons		
Number of persons:	2	
Number of children:	0	
Number of pets:	0	
Costs		
Rental price:	€ 388	
Standard Cancellation Insurance:	€ 25	
Extended Cancellation Insurance:	€ 0	
Travel Insurance:	€ 0	
Winter sports insurance:	€ 0	
Discount:	€ 0	
Total amount:	€ 413	
payment methods		
Payment can be made via a direct-debit authorisation form or acceptgiro (only possible with a Dutch bank account), or by bank transfer.		

Figure 9: booking form for definite booking (step 3)

The answer of the “Booking.PlaceOption-v2” call will look like:

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<methodResponse>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>OptionNumber</name>
            <value><string>649269</string></value>
            <name>ClientNumber</name>
            <value><string>1608977</string></value>
            <name>OptionExpireDate</name>
            <value><string>31-03-2006</string></value>
            <name>WebPartnerBookingCode</name>
            <value><string></string></value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodResponse>
```

If you are going to make an option booking please keep in mind you have to display a warning that the booking you are making is no real booking but an option. You have to display something like: “As you are booking within 14 days of arrival at your holiday home, we are obliged to inform the home owner of your arrival prior to confirmation. On confirmation of the reservation you will receive an option number and further information as to the next steps to be made regarding the booking”.

With the given option number, the client has to call our service number to make the option booking a definite booking.

The screen could look like the one on the next page:


General information															
 <p>***</p>	<table> <tr> <td>Holiday house:</td> <td>Demo house</td> </tr> <tr> <td>City:</td> <td>Demo city</td> </tr> <tr> <td>House code:</td> <td>DEMO-1000-01</td> </tr> <tr> <td>Length of stay:</td> <td>1 Week</td> </tr> <tr> <td>Arrival:</td> <td>Friday 01-01-2006 16:00 hour and 19:00 hour</td> </tr> <tr> <td>Departure:</td> <td>Friday 08-01-2006 before 10:00 hour</td> </tr> <tr> <td>Rental price:</td> <td>€ 388</td> </tr> </table>	Holiday house:	Demo house	City:	Demo city	House code:	DEMO-1000-01	Length of stay:	1 Week	Arrival:	Friday 01-01-2006 16:00 hour and 19:00 hour	Departure:	Friday 08-01-2006 before 10:00 hour	Rental price:	€ 388
	Holiday house:	Demo house													
	City:	Demo city													
	House code:	DEMO-1000-01													
	Length of stay:	1 Week													
	Arrival:	Friday 01-01-2006 16:00 hour and 19:00 hour													
Departure:	Friday 08-01-2006 before 10:00 hour														
Rental price:	€ 388														
Additional Remarks															
<p>As you are booking within 14 days of arrival at your holiday home, we are obliged to inform the home owner of your arrival prior to confirmation. On confirmation of the reservation you will receive an option number and further information as to the next steps to be made regarding the booking.</p>															
Thank you for booking with Euro Relais!															
<p>Below you will find further information on your booking and what happens next.</p>															
Bookingsgegevens															
Your Option Number:	649274														
Your Customer Number:	1635831														
Personal information															
Surname:	Mr. T Test														
Street:	test 12														
Postal code / City:	nl-5454aa test														
Telephone:	0413333333														
Mobile:	0633333333														
E-mail:	test@test.nl														
Number of persons															
Number of persons:	2														
Number of children:	0														
Number of pets:	0														
Costs															
Rental price:	€ 388														
Standard Cancellation Insurance:	€ 25														
Extended Cancellation Insurance:	€ 0														
Travel Insurance:	€ 0														
Winter sports insurance:	€ 0														
Discount:	€ 0														
Total amount:	€ 413														
payment methods															
<p>Payment can be made via a direct-debit authorisation form or acceptgiro (only possible with a Dutch bank account), or by bank transfer.</p>															

Figure 10: booking form for making an option (step 3)