

# **Restless Gambling API**

Kamil Banasik

2019/03/20 14:55

# Table of Contents

1. Overview .....	3
1.1 Purpose of this document .....	3
1.2 Role of Yggdrasil Restless API in E-gambling deployment .....	4
2. Concepts .....	5
2.1 General gambling flow .....	5
2.1.1 Launch and authentication .....	5
2.1.2 Making a bet .....	6
2.1.3 Ending a bet .....	6
2.2 Wagers .....	7
2.3 Balance and applicable bonus .....	7
2.4 Transaction categories .....	7
2.4.1 Game Tags .....	7
2.5 Bet details .....	8
2.6 Idempotency, Tombstones and Robustness .....	8
3. Protocol generics .....	9
3.1 Request format .....	9
3.2 Money data .....	9
3.3 Reply format .....	9
3.4 Successful reply data .....	9
3.5 Failure reply data .....	10
3.6 Fault handling .....	10
3.7 Prefix for transactions (from UK, RO, IT, ES, DK, SE, CZ...) .....	11
4. Protocol requests .....	11
4.1 playerinfo .....	11
4.2 wager .....	14
4.3 cancelwager .....	17
4.4 appendwagerresult .....	18
4.5 endwager .....	21
4.6 campaignpayout .....	23
4.7 getbalance .....	25
5. Launching games .....	26
6. Jackpot .....	28
6.1 Jackpot feed for Joker Milions (only available for clients with Maltese license) .....	28
6.2 Jackpot feed for all games .....	28
7. Bingo .....	30
8. Typical integration plan .....	30
9. Frequently asked questions .....	30

## Table of Contents

- [1. Overview](#)
    - [1.1 Purpose of this document](#)
    - [1.2 Role of Yggdrasil Restless API in E-gambling deployment](#)
  - [2. Concepts](#)
    - [2.1 General gambling flow](#)
      - [2.1.1 Launch and authentication](#)
      - [2.1.2 Making a bet](#)
      - [2.1.3 Ending a bet](#)
    - [2.2 Wagers](#)
    - [2.3 Balance and applicable bonus](#)
    - [2.4 Transaction categories](#)
      - [2.4.1 Game Tags](#)
    - [2.5 Bet details](#)
    - [2.6 Idempotency, Tombstones and Robustness](#)
  - [3. Protocol generics](#)
    - [3.1 Request format](#)
    - [3.2 Money data](#)
    - [3.3 Reply format](#)
    - [3.4 Successful reply data](#)
    - [3.5 Failure reply data](#)
    - [3.6 Fault handling](#)
    - [3.7 Prefix for transactions \(from UK, RO, IT, ES, DK, SE, CZ...\)](#)
  - [4. Protocol requests](#)
    - [4.1 playerinfo](#)
    - [4.2 wager](#)
    - [4.3 cancelwager](#)
    - [4.4 appendwagerresult](#)
    - [4.5 endwager](#)
    - [4.6 campaignpayout](#)
    - [4.7 getbalance](#)
  - [5. Launching games](#)
  - [6. Jackpot](#)
    - [6.1 Jackpot feed for Joker Millions \(only available for clients with Maltese license\)](#)
    - [6.2 Jackpot feed for all games](#)
  - [7. Bingo](#)
  - [8. Typical integration plan](#)
  - [9. Frequently asked questions](#)
- 

# 1. Overview

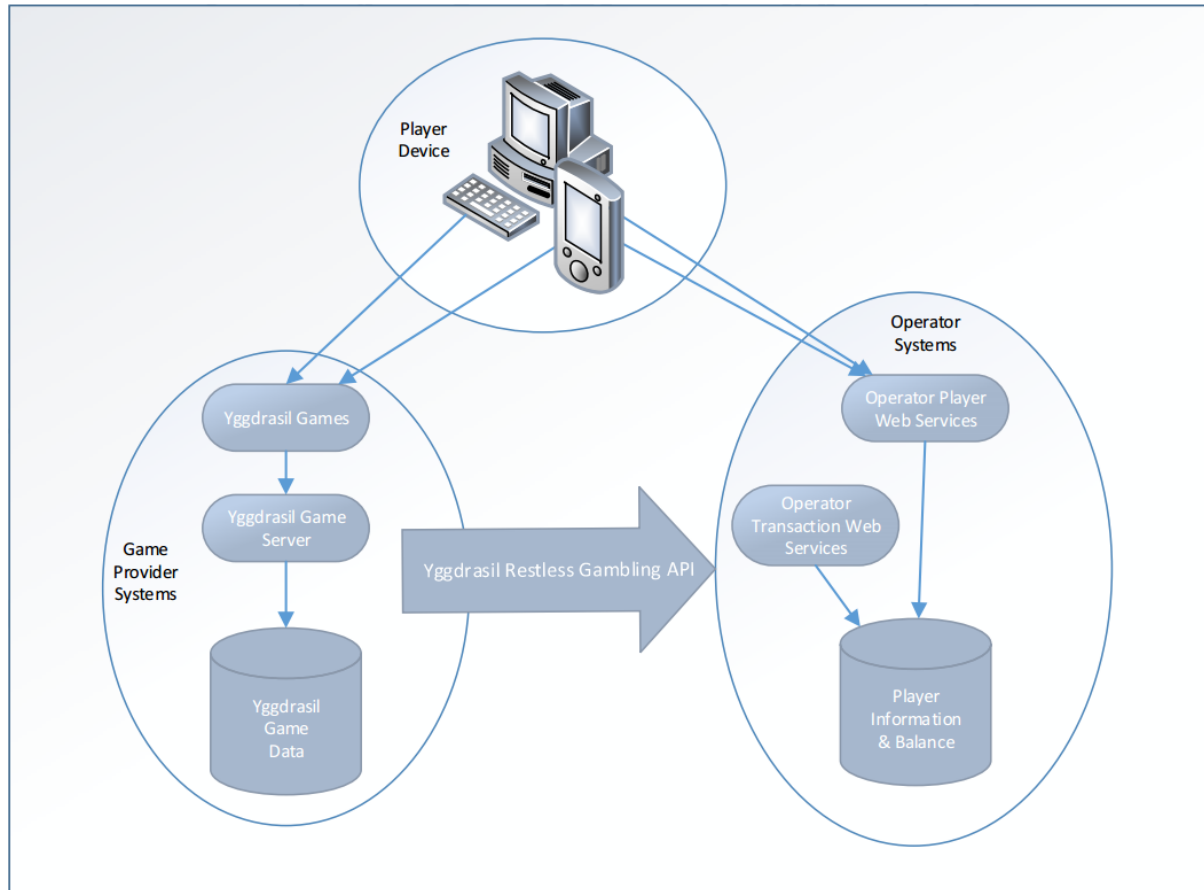
## 1.1 Purpose of this document

The purpose of this document is to describe how to integrate an Operator system with the Yggdrasil Game Server and the Yggdrasil Games.

The reader is expected to be familiar with internet gambling system architectures.

---

## 1.2 Role of Yggdrasil Restless API in E-gambling deployment

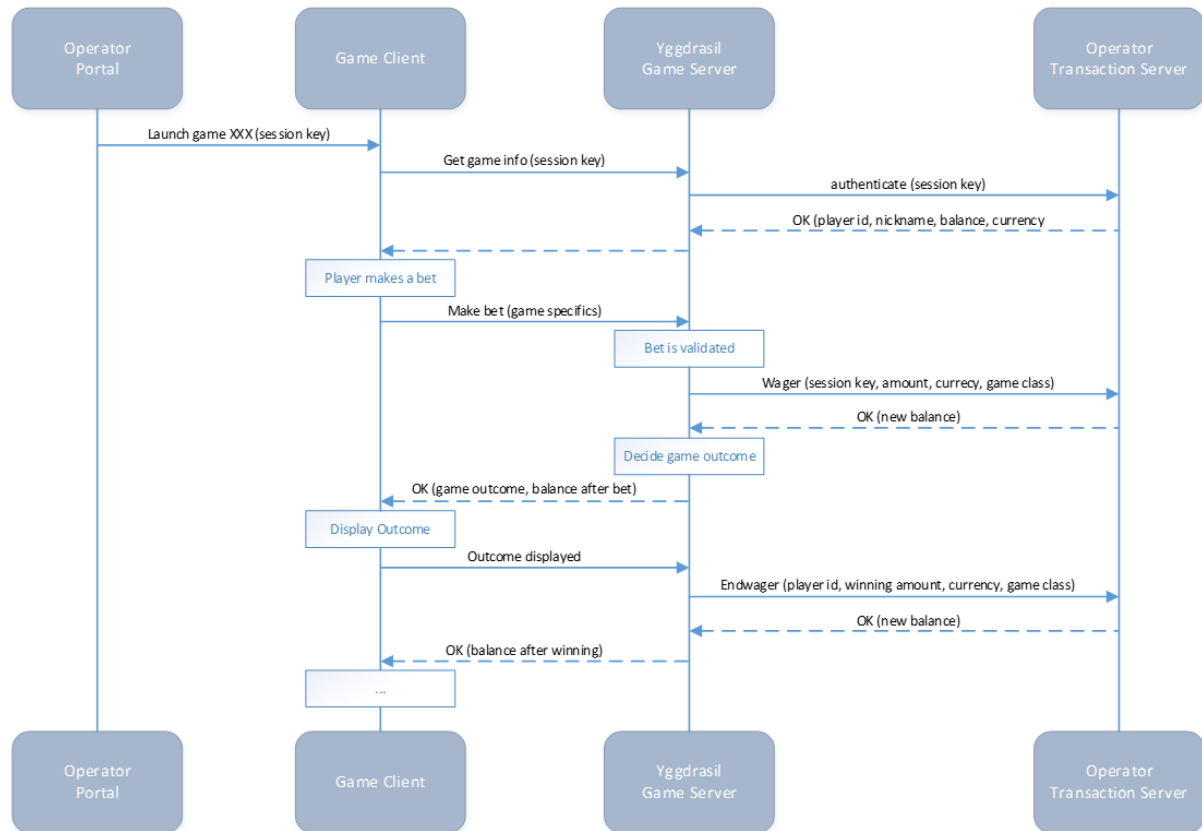


The Yggdrasil Restless Gambling API is a URL and Json based operator interface for handling player information and gambling transactions when connecting an operator's player database to the Yggdrasil Game Server.

The word "Restless" in the name of the API stems from the fact that the API is not a RESTful API but a only a simple protocol built on HTTP Get operation sending parameters as URL parameters and returning results as Json structures.

## 2. Concepts

### 2.1 General gambling flow



#### 2.1.1 Launch and authentication

The game play starts when the operator's web portal launches the actual game. How this is done may vary from game to game and is not part of this API description. But it is important that the web portal passes over a session key that can be used to validate the player login. This session key should be valid only for the current player session.

The Yggdrasil Game Server verifies the session key over the Restless call by the **playerinfo** URL, receiving some basic information about the player such as nickname, home currency and current balance.

Possible parameters:

Parameter	Example value	Default value	Scope limitation	Description
key	160421121051050000(...)			Session ID
currency	EUR, PLN, GBP			User's currency
lang	en, sv, pl	en		User's language
gameid	7301			Game identification number. Our game portfolio (with gameids) you can find in our Client Zone: <a href="https://client.yggdrasilgaming.com/docs-files/">https://client.yggdrasilgaming.com/docs-files/</a> Game catalogue file (ask Business for access)

<b>org</b>	Casino			Name of your brand organisation
<b>channel</b>	pc, mobile	pc		User's channel
<b>home</b>	https://yggdrasilgaming.com/		mobile only	Home button and Reality Check Stop button redirection URL (if realityCheckBackURL not specified)
<b>fullscreen</b>	yes, no	no	desktop only	Enables the option to play in fullscreen mode
<b>reminderElapsed</b>	30	0		Session time passed (minutes)
<b>reminderInterval</b>	100, 40, 0	60		Reality Check frequency (minutes). If the value is 0 then Reality Check is turned off. Otherwise, first info will appear after (reminderInterval-reminderElapsed) and each and every next will appear after reminderInterval
<b>topOrg</b>	CasinoGroup			only for pff launch client Your name of top organization
<b>clientHistoryURL</b>	https://yggdrasilgaming.com/		only for UK/GiB integrations	Game History link inside Reality Check Window. If not specified then it links to in-game Game History
<b>realityCheckBackURL</b>	https://yggdrasilgaming.com/			Reality Check Stop button redirection URL
<b>redirectType</b>	self, top	self		Redirection type (self - redirects iframe content, top, redirects iframe parent)
<b>share</b>	yes, no	yes		Enables or disables social features
<b>leaderboardRewatch</b>	yes, no	yes		Enables or disables watch functionality in tournaments
<b>license</b>	it, uk, ro, dk, it, schhol, agcc, cz, es, se		only for UK/RO/IT/DK/IT/SCHHOL/AGCC/CZ/ES integrations	Enables or disables additional features (for particular license) in game client. More information <a href="#">here</a>

### 2.1.2 Making a bet

When a bet has been made by the player and the Game Server has validated the bet parameters and prepared for the bet, a **wager** URL is sent to the operator. In case of success, the reply contains the updated player information.

### 2.1.3 Ending a bet

When the game has been played and the result has been determined and revealed to the player, the bet is ended by an **endwager** call that also informs the operator transactions system of the winnings. (For some games this happens before the game is actually revealed to the player).

Note that **endwager** is sent also on losing wagers.

## 2.2 Wagers

A wager is a sequence of bets (each sent by a **wager** call) terminated by one or more results, the last sent by an **endwager** call. All transactions that relate to the same wager are identified by sharing the same “reference”. This reference numbers are never reused.

E.g. a wager could start when a player buys an entry to a tournament and end when he loses or wins the tournament. There are a number of reasons why these transactions should be kept together, rather than seen as two entirely separate transfers. An important reason is to make it possible to keep track of the money that are “in play”, i.e. reserved for a bet (or other transaction) that has not been settled.

---

## 2.3 Balance and applicable bonus

There may be two amounts returned as a balance for the players.

Balance	Description
balance	The withdrawable amount of money that the player has available.
applicableBonus	The amount of bonus money that the player has and that may be used for a certain game as defined by the categorization.

Some games may display the available amount separately, but normally the sum of balance and applicableBonus is displayed, if anything is displayed at all.

The gambling transactions do not differentiate between balance and bonus. It is up to the operator transaction system to charge by using balance or bonus and to match and pay out the right kind when winnings are registered for the wager.

---

## 2.4 Transaction categories

Each gaming transaction is classified with a game category and optionally a number of tags.

The classification is a hierarchical structure that groups and identifies all Yggdrasil Games uniquely. The structure for lotteries is shown in the table below

Category	Example	Meaning
Cat1	Casino	The category of games
Cat2	Slot	The type within the category
Cat3	Seasons	The variant within the type
Cat4	Four Seasons	The name of the game
Cat5	7323	The number of the game

The exact contents of these fields is part of the game definition. The fields are supplied for two optional uses by the operator transaction system. There is NO REQUIREMENT on the operator transaction system to manage this data. To avoid complicated deployment of new or changed games, it is strongly recommended that the operator transaction system accepts all inputs and dynamically increases the set of different combinations received.

The first purpose of this structure is to make it possible for the operator to implement different bonuses and campaigns for different games or groups of games. Thus the operator transaction system could have bonus money available for all Lotteries, but not for other games. Or there could be a campaign to try a certain game for free.

The second purpose is of course to make possible intrinsic reporting.

### 2.4.1 Game Tags

There may also be a number of tags sent with each transaction. The operator transaction system can safely ignore them unless there is need to use the data for creating reports.

The actual tags depend on the game definition. The table below shows an example.

Tag group	Example	Meaning
GameName	GameName.Seasons	The name of the game (note that the same game name can appear as a category for many types of games)
Model	Model.M1	Special game model designator
Channel	Channel.PC	From where the game is played PC/Mobile

For formatting reasons, the tags are sent as TagGroup.Value in up to nine tag parameters called tag1 through tag9 (example: &tag1=GameName.Seasons)

## 2.5 Bet details

If detailed information about underlying bets is required, it can be configured in the wallet configuration within the game server.

Set the wallet property “betdetails=Yes” to enable additional bet information.

The information will be sent as a JSON array in the **bets** response field. Note that the format of this information is game specific. The information is intended mainly for logging and regulatory purposes.

## 2.6 Idempotency, Tombstones and Robustness

To get a robust connection between systems when the calling system does not get a proper reply to a request, it may need to redo the operation in case the first time did fail, or cancel it in case it did actually happen.

Such actions are supported by an idempotency scheme that makes repeating an operation into a no-op. The transactions are identified by a so called transaction reference. Thus if there is a failure, the calling service can safely repeat the request until it receives an OK reply, without risking double debits or credits.

An advantage of the “idempotency” approach to robustness in the API is that it is resilient also to messages that arrive very late. E.g. if a protocol instead asked “what happened to transaction N” and acted on that result it could well happen that the reply about what happened could be sent before transaction N arrives to the operator system. That way the game server could e.g. deduce that there is no need to refund money, but then the money is actually reserved for the failed bet.

To reconcile systems, the Restless Gambling API expects idempotency on the transaction reference. This means that if a transaction has already been made – as identified by the same **reference** and **subreference**, the repeated operation will be a no-op.

In some cases – such as a **wager**, normally the transaction will be “undone” in case of an unidentified failure. This is performed by the **cancelwager** call.

In other cases – such as **endwager** - the transaction will be repeated until an OK result is returned by the operator transaction system.

Further, in case a the Yggdrasil Game Server cancels a wager (by **cancelwager**) instead of retrying ( e.g. cancelling the bet and informing the player rather than trying again to perform it), the transaction should not only be cancelled, but also leave a mark (a tombstone) so that any delayed original **wager** request will not “undo the cancellation”.



## 3. Protocol generics

### 3.1 Request format

The request format is a general URL GET with parameters. E.g. as typed into a browser

<https://transaction.service.operator.com/playerinfo.json?sessiontoken=12101614403789000025BE113B9D0435889888BEBEC868F3C15D1E76BE0D91E70253D69BACB411BB4718&version=1&lang=en>

Data should be sent in UTF-8 format and standard url encoding should be applied where needed.

The request can also be sent using POST, whereas the parameters are sent as post data.

To enable the use of POST instead of GET, the wallet configuration property “method=POST” need to be set in the game server configuration

### 3.2 Money data

Monetary amounts are sent as strings up to 13 digits, a decimal point and then up to two decimals. E.g. 1234.56 or 9.00. in the URL parameter. In replies it should be sent as a json string value of the same format, as e.g. “1234.00” The decimals are optional so the previous example could also be sent as “1234”.

### 3.3 Reply format

Reply format should be a json structure returned as data, also in UTF-8. The media type should be “application/json”.

### 3.4 Successful reply data

Returned field	Required field	Format	Comment
code	yes	Integer	The return code (see below)
data		Structure	The data returned
data.playerId	yes	String	Id of the player
data.nickName	no	String	The nick name of the player, to be displayed in Table Games (optional).
data.organization	yes	String	Name of the operator to whom the player belongs.
data.balance	yes	String number	Amount of withdrawable money available to player
data.applicableBonus	yes	String number	Amount of bonus money available to player for the game referred in the request.
data.currency	yes	String	The three letter code of the balance and bonus returned.
data.homeCurrency	yes	String	The three letter code of the player's default currency.

Example:

```
{ "code": 0,
  "data": {
    "playerId": "40000001",
    "nickName": "Restless",
    "organization": "Demo",
```

```

    "balance":4000.00,
    "applicableBonus":0.00,
    "currency":"EUR",
    "homeCurrency":"EUR"
  }
}

```

### 3.5 Failure reply data

Returned field	Version	Format	Comment
code		Integer	The return code (see below)
msg		String	Message for internal purposes (it will show up in YGG internal logs), it doesn't affect a message presented to the player in any way.

Example:

```

{
  "code":1000,
  "msg":"Session expired. Please log in again."
}

```

### 3.6 Fault handling

Fault handling in the Restless Gambling API is rather simple. There are only a few return codes that the Game Server cares about, and in these cases there should be a translated message to display to the player.

Return code	Meaning	Comment	Error in-game message
0	Success	Expected for all calls	
1000	Not logged in	The session key is not valid.	A technical error occurred when processing the request. Please contact support
1006	Overdraft	The player is out of money in the currency requested. The message supplied is displayed, and if possible the player is directed to where he deposits money.	You do not have sufficient funds for the bet
1007	Blocked	The player is blocked from play, e.g. due to responsible gaming limits. The supplied message is displayed and the game client is closed.	The account is blocked and no bets can be performed
1008	Not authorized	The player is not authorized to make this bet. The supplied message is displayed to the player.	You are not allowed to perform the bet due to gaming limits
1013	Bonus limit	Restrict the max bet value when playing the game with bonus money.	You cannot place this bet due to max bet limit on bonus funds
1	Any other error	The operation will be cancelled or retired. The player is informed that there is "a technical problem". The message will be logged for system maintenance.	A technical error occurred when processing the request. Please contact support

In case of one of the recognized return codes (i.e. 1000, 1006, 1007 or 1008), the Yggdrasil Game Server **assumes that the operation did not go through**. In case of return code 1 (for synchronous gameplay) or any other return code that is non-zero, the game server will retry or cancel the transaction as appropriate.

---

### 3.7 Prefix for transactions (from UK, RO, IT, ES, DK, SE, CZ...)

For transactions from these jurisdictions: UK, RO, IT, ES, DK, SE, CZ, LV, EE we add an additional prefix (uk, ro, it, es, dk, se, cz) for methods: wager, endwager, cancelwager, appendwagerresult, campaignpayout.

Reference parameter contains an additional prefix. We add prefix only for 'reference', 'subreference' is without additional prefix.

Operator needs to have a possibility to distinguish the below two endwagers (and other calls: wager, cancelwager etc.) in your system (the same subreference and reference with prefix).

Example:

Environment MT: endwager, reference 1610111151170000003, subreference 1610111152240000013.1610111152240000015

Environment UK: endwager, reference uk1610111151170000003, subreference 1610111152240000013.1610111152240000015

Example of requests:

[https://host/wager.json?  
org=YourOrgName&sessiontoken=7d723bc2e501435ba0e671d67b3fbdca&playerid=e1\\_43859\\_eur&amount=0.20&currency=EUR&service=Mir+Game&cat1=Casino&cat2=Slot&cat3=Seasons&cat4=Four+Seasons&cat5=7323&tag1=GameName.Four+Seasons&tag2=Model.M1&tag3=Channel.pc&lang=en&version=5](https://host/wager.json?org=YourOrgName&sessiontoken=7d723bc2e501435ba0e671d67b3fbdca&playerid=e1_43859_eur&amount=0.20&currency=EUR&service=Mir+Game&cat1=Casino&cat2=Slot&cat3=Seasons&cat4=Four+Seasons&cat5=7323&tag1=GameName.Four+Seasons&tag2=Model.M1&tag3=Channel.pc&lang=en&version=5)

<https://host/cancelwager.json?>

[https://host/appendwagerresult.json?  
org=YourOrgName&playerid=cc\\_1\\_eur&amount=0.00&bonusprize=0.00&currency=EUR&tickets=1&service=Mir&reference=uk1603161511700000003.1610111152240000013.1610111152240000015&subreference=1610111152240000013.1610111152240000015&GameName.Incinerator&tag2=Model.M1&tag3=Channel.pc&lang=en&version=5](https://host/appendwagerresult.json?org=YourOrgName&playerid=cc_1_eur&amount=0.00&bonusprize=0.00&currency=EUR&tickets=1&service=Mir&reference=uk1603161511700000003.1610111152240000013.1610111152240000015&subreference=1610111152240000013.1610111152240000015&GameName.Incinerator&tag2=Model.M1&tag3=Channel.pc&lang=en&version=5)

<https://host/campaignpayout.json?>

---

## 4. Protocol requests

### 4.1 playerinfo

Operator's integration team is obliged to implement *playerinfo.json* in order to run gameplay. The parameters that are send by Yggdrasil Game Server and parameters that should be send back by Operator Transaction System are described in depth below.

The Operator Transaction System should validate if the provided session key is currently valid and return player information as defined above. In case the operator system is capable of currency exchange, it should return the available funds as converted to the currency sent. Otherwise, it should return the funds in the currency it can return.

In case there is any bonus money, that is applicable to a game according to the cat1..cat9 classification (see above), the bonus amount should be returned in return call.

As of version 3, playerinfo should return the extended information as it is described in section 3.5.

Parameters in playerinfo request send by the Yggdrasil Game Server:

## Restless Gambling API - Restless Gambling API

Parameter	Version	Format	Max Length	Comment
org		String	32	Organization identifier(s)
sessiontoken		String	32	The unique token key to identify and validate the player session
cat1..cat9		String	32	The category information describing the game for which any applicable bonus should be reported
lang		String	2	Two-letter language code used to identify language which will be utilized to translate messages presented to player
version		Integer	11	Protocol version
tag1..tag9		String	64	The tag information of the game

In the following way the game server will request an operator system:

[https://host/playerinfo.json?  
org=YourOrgName&sessiontoken=dc7e81db760c466a890274a9c153c349&lang=no&version=5&cat1=Casino&cat2=Slot&cat3=StickyJ&cat4=  
+Millions&cat5=7312&tag1=GameName.Joker+Millions&tag2=Model.M1&tag3=Channel.pc](https://host/playerinfo.json?org=YourOrgName&sessiontoken=dc7e81db760c466a890274a9c153c349&lang=no&version=5&cat1=Casino&cat2=Slot&cat3=StickyJ&cat4=+Millions&cat5=7312&tag1=GameName.Joker+Millions&tag2=Model.M1&tag3=Channel.pc)

Expected response parameters:

Parameter	Format	Max Length	Mandatory field	Comment
code	Int	5	yes	Should return 0 if request was valid. For error codes see chapter „Fault handling”
data.gender	String	1	optional	“M” for male or “F” for female. Other inputs will be ignored
data.playerId	String	20	yes	Identifies the player (User ID that is generated by partner’s system)
data.organization	String	32	yes	Organization identifier(s)
data.balance	String	20	yes	Amount of withdrawable money available to a player.
data.applicableBonus	String	20	optional	Amount of bonus money available to a player for the game referred in the request.
data.currency	String	3	yes	The three letter code of the player’s default currency.
data.homeCurrency	String	3	yes	The three letter code of the player’s default currency.
data.nickName	String	64	optional	The nick name of the player, to be displayed in Table Games (optional).
data.country	String	2	optional	The 2-letter country code of the player (ISO 3166-1 alfa-2)

## Restless Gambling API - Restless Gambling API

data.gameHistorySessionId	String	64	only for Italy	Required ONLY for ITALY (jurisdiction)
data.gameHistoryTicketId	String	64	only for Italy	Required ONLY for ITALY (jurisdiction)
data.popupMessage	JSON		only for UK/GIB	Optional ONLY for UK/GIB
data.popupMessage.title	String	30	optional	
data.popupMessage.content	String	86	optional	
data.popupMessage.buttons	String	11	optional	for example TERMS or CLOSE
data.popupMessage.buttonsStringType	StringType		optional	REDIRECT or CLOSE
data.popupMessage.buttons	String		optional	<a href="https://your.url.com/">https://your.url.com/</a> or null (to close the popup)
data.popupMessage.channel	String		optional	PC or MOBILE or BOTH
msg	String	100	yes	Only if code not equals to 0. Message with information about the error.

Example of response send back by the operator's system:

```
{
  "code": 0,
  "data": {
    "playerId": "40000001",
    "nickName": "Restless",
    "organization": "Demo",
    "balance": 4000.00,
    "applicableBonus": 0.00,
    "currency": "EUR",
    "homeCurrency": "EUR",
    "country": "DE",
    "gameHistorySessionId": "1234",
    "gameHistoryTicketId": "56789",
    "popupMessage": {
      "title": "Example Message Title!",
      "content": "This is Example message content!",
      "buttons": [
        {
          "label": "EXAMPLE",
          "actionType": "REDIRECT",
          "url": "https://your.url.com/"
        },
        {
          "label": "CLOSE",
          "actionType": "CLOSE",
          "url": null
        }
      ],
      "channel": "PC"
    }
  }
}
```

## 4.2 wager

Operator's integration team is obliged to implement *wager.json* in order to run gameplay. The parameters that are send by Yggdrasil Game Server and parameters that should be send back by Operator Transaction System are described in depth below.

The Operator Transaction System should reserve the amount unless the same wager (identified by reference and subreference) has already been processed or if cancelwager request has been already on it executed. The reservation may be of cash or of bonus or of any other kind.

The reply should reflect the same currency and any applicable bonus for the game according to the cat1..cat9 classification (see above) .

This could be the first transaction that belongs to the wager (identified by the reference), but it could also be a second or third bet in the same wager in case the game has several buy-ins.

If the wager is made by using a prepaid ticket, the amount will be equal to zero (unless part of the ticket is paid by the player for some reason), and the value of the bet will be in the prepaidvalue field and the actual cost of the bet (that may be different) will be in the prepaidcost field and the id of the used ticket will be in the prepaid ticket id. All three fields are for information only and may be ignored by the operator, as long as the zero-price wager is accepted.

Parameters in wager request send by the Yggdrasil Game Server:

Parameter	Version	Format	Max Length	Comment
sessiontoken		String	32	The unique token key to identify and validate the player session
org		String	32	Organization identifier(s)
playerid		String	20	The player id as returned from an authenticate call
amount		Money	22	The cost of the bet. (ex: 9999999999999999.99)
currency		String	3	The three letter code of the currency betted.
reference		String	20	Unique identifier of the wager
subreference		String	50	Unique identifier within the wager of the transaction
description		String	20	Description of the bet, translated and meant for the player to read.
prepaidticketid	2 or higher	String	20	Id of the ticket used to pay for wager
prepaidvalue	2 or higher	Money	22	The value paid for the wager by the prepaid ticket.
prepaidcost	2 or higher	Money	22	The cost as charged for the prepaid ticket
prepaidref	2 or higher	String	64	Reference that was supplied when prepaids were created
jackpotcontribution	4 or higher	String	22	Part of the players bet, which go to jackpot pool
cat1..cat9		String	64	The category information describing the game played

## Restless Gambling API - Restless Gambling API

tag1..tag9	String	64	The tag information of the game
lang	String	2	Two-letter language code used to identify language which will be utilized to translate messages presented to player
version	Integer	11	Protocol version

### Important!

*Description* field (case insensitive) has few possible values which need to be handled in order to make auto payout functionality work properly:

- "CashRace" - means that next endwager will be auto payout for participation in prize drop created in Yggdrasil's Back Office (in this case **only endwager** request contains description: "CashRace". Wager has a different description and amount of this wager is **greater than 0**)
- "Tournament" - means that next endwager will be auto payout for participation in tournaments created in Yggdrasil's Back Office
- "NetworkTournament" - means that next endwager will be auto payout for participation in network tournaments created in Yggdrasil's Back Office
- "NetworkMission" - means that next endwager will be auto payout for participation in network missions created in Yggdrasil's Back Office
- "Boost" - means that next endwager will be auto payout for participation in other boost tools that are not mentioned above

When *description* field contains one of above parameters, winnings from those bets should not be counted to Game Win (RTP). Keep in mind that there are other possible values in description field.

In the following way game server will request operator system:

[https://host/wager.json?org=YourOrgName&sessiontoken=7d723bc2e501435ba0e671d67b3fbdca&playerid=el\\_43859\\_eur&amount=0.20&currency=EUR&service=M+Game&cat1=Casino&cat2=Slot&cat3=Seasons&cat4=Four+Seasons&cat5=7323&tag1=GameName.Four+Seasons&tag2=Model.M1&tag3=Channel.pc&lang=en&version=5](https://host/wager.json?org=YourOrgName&sessiontoken=7d723bc2e501435ba0e671d67b3fbdca&playerid=el_43859_eur&amount=0.20&currency=EUR&service=M+Game&cat1=Casino&cat2=Slot&cat3=Seasons&cat4=Four+Seasons&cat5=7323&tag1=GameName.Four+Seasons&tag2=Model.M1&tag3=Channel.pc&lang=en&version=5)

Expected response parameters:

Parameter	Version	Format	Max Length	Comment
organization		String	32	Organization identifier(s)
playerId		String	20	The player id as returned from an authenticate call
currency		String	3	The three letter code of the currency betted.
applicableBonus		String	20	Amount of bonus money available to player for the game referred in the request.
homeCurrency		String	3	The three letter code of the player's default currency.
balance		String	20	Amount of withdrawable money available to player
nickName		String	64	The nick name of the player, to be displayed in Table Games (optional).
bonus		String	20	Bonus that player receive when make deposit to his wallet.

## Restless Gambling API - Restless Gambling API

gameHistorySessionId		String	64	Required ONLY for ITALY (jurisdiction)
gameHistoryTicketId		String	64	Required ONLY for ITALY (jurisdiction)
data.popupMessage		JSON	only for UK/GIB	Optional ONLY for UK/GIB
data.popupMessage.title		String	30 optional	Title of popupMessage
data.popupMessage.content		String	86 optional	Content of popupMessage
data.popupMessage.buttons		String	11 optional	for example TERMS or CLOSE
data.popupMessage.buttons.actionType		String	optional	REDIRECT or CLOSE
data.popupMessage.buttons		String	optional	for example <a href="https://your.url.com/">https://your.url.com/</a> or null (to close the popup)
data.popupMessage.channel		String	optional	PC or MOBILE or BOTH

Example of response send back by operator's system:

```
{
  "code": 0,
  "data": {
    "currency": "eur",
    "applicableBonus": 0.0,
    "homeCurrency": "eur",
    "organization": "orgDemo",
    "balance": 7.84,
    "nickName": "belles",
    "playerId": "el_43859_eur",
    "gameHistorySessionId": "1234",
    "gameHistoryTicketId": "56789",
    "popupMessage": {
      "title": "Example Message Title!",
      "content": "This is Example message content!",
      "buttons": [
        {
          "label": "TERMS",
          "actionType": "REDIRECT",
          "url": "https://your.url.com/"
        },
        {
          "label": "CLOSE",
          "actionType": "CLOSE",
          "url": null
        }
      ],
      "channel": "PC"
    }
  }
}
```

**Communication flow for automatic payouts works as follows (example of a scenario for Network Tournament):**

First, Yggdrasil Game Server sends wager request with *amount* = '0.00' and *description* = 'NetworkTournament'.

<https://host/wager.json?org=YourOrgName&sessiontoken=ddd&playerid=yyy&amount=0.00&currency=EUR&service=Mir&reference=1612031010460000155&subre>

Response send back by the operator's system:

```
{
```



```

"code":0,
"data": {
  "organization": "xxx",
  "playerId": "yyy",
  "balance": 961.20,
  "applicableBonus": 0.44,
  "currency": "EUR",
  "homeCurrency": "EUR"
}
}

```

In the last step, Yggdrasil Game Server sends endwager request with the prize money for the participation in Network Tournament. *Description* is also set to 'NetworkTournament' and prize is in the *amount* parameter.

<https://host/endwager.json?org=YourOrgName&playerid=yyy&amount=50.00&bonusprize=0.00&currency=EUR&tickets=1&service=Mir&reference=1612031010460000>

Response send back by the operator's system:

```

{
  "code": 0,
  "data": {
    "organization": "xxx",
    "playerId": "yyy",
    "balance": 1011.20,
    "applicableBonus": 0.44,
    "currency": "EUR",
    "homeCurrency": "EUR"
  }
}

```

### 4.3 cancelwager

Operator's integration team is obliged to implement *cancelwager.json* in order to run gameplay. The parameters that are sent by Yggdrasil Game Server and parameters that should be send back by Operator Transaction System are described in depth below.

The operator transaction system should cancel the wager transaction that has the same reference and subreference field. All reserved funds should be credited to the player.

If there has not been a wager with the given reference and subreference, the Operator Transaction System should leave a mark to prevent any further **wager** requests with the same reference and subreference.

#### Attention:

We send reference and subreference in cancelWager.

If you find this transaction in your system and cancel on your side succesfully - please return code 0.

If you couldn't find this transaction in your database - please return code 0 (the same like in above case).

In other cases - please return code: 1.

Parameters in cancelwager request send by the Yggdrasil Game Server:

Parameter	Version	Format	Max Length	Comment
playerid		String	20	The player id as returned from an authenticate call
reference		String	20	Unique identifier of the wager

## Restless Gambling API - Restless Gambling API

subreference	String	50	Unique identifier within the wager of the transaction
org	String	32	Organization identifier(s)
version	Integer	11	Protocol version

In the following way game server will request operator system:

[https://host/cancelwager.json?  
org=YourOrgName&playerid=D80187CDEF79499&reference=1603231304260100003&subreference=1&version=3](https://host/cancelwager.json?org=YourOrgName&playerid=D80187CDEF79499&reference=1603231304260100003&subreference=1&version=3)

Expected response parameters:

Parameter	Version	Format	Max Length	Comment
organization		String	32	Organization identifier(s)
playerId		String	20	The player id as returned from an authenticate call
currency		String	3	The three letter code of the currency betted.
balance		String	20	Amount of withdrawable money available to player
bonus		String	20	Bonus that player receive when make deposit to his wallet.

Example of response send back by operator's system:

```
{
  "code": 0,
  "data": {
    "playerId": "D80187CDEF79499",
    "organization": "Demo",
    "balance": 12.95,
    "currency": "EUR"
  }
}
```

---

## 4.4 appendwagerresult

Operator's integration team is have to implement *appendwagerresult.json* in order to run gameplay. Appendwagerresult is used **for new jackpot games such as Joker Millions and Mr Holmes and additionally for Boost tool: Prize Drop**. The parameters that are sent by Yggdrasil Game Server and parameters that should be send back by Operator Transaction System are described in depth below

The Operator Transaction System should pay out the amount unless the same appendwagerresult (identified by reference and subreference) has already been processed. The operator system should preferably credit bonus money in place of cash if the corresponding wager was paid for by bonus. And if there is a currency conversion, probably the same rate should be applied as when the bet was made.

## Restless Gambling API - Restless Gambling API

Bonusprize is used when the game has an extra payout that should normally not be withdrawable, such as an achievement bonus or the winnings from a prepaid ticket, on top of any bonus that may have resulted from the cash amount paid out. Note that the operator platform has to decide and manage any wagering requirements.

The reply should reflect the same currency and any applicable bonus for the game according to the cat1..cat9 classification (see above) .

This is an intermediary or otherwise extra result payout for a game that may yield more payouts. There will be an endwager call later.

This call is never made with zero payout, and would not have any meaning in that case.

Parameters in appendwagerresult request send by the Yggdrasil Game Server:

Parameter	Version	Format	Max Length	Comment
org		String	32	Organization identifier(s)
playerid		String	20	The player id as returned from an authenticate call when the game was launched
amount		Money	22	The winnings of the bet
isJackpotWin		Boolean	4	True if win is a jackpot
bonusprize	2	Money	22	The value of any bonus prize paid out, i.e. prizes that should not be withdrawable but only used for gambling.
currency		String	3	The three letter code of the currency won
reference		String	20	Unique identifier of the result payout
subreference		String	50	Unique identifier within the wager of the transaction
description		String	64	Description of the winning, translated and meant for the player to read.
cat1..cat9		String	64	The category information describing the game played
tag1..tag9		String	64	The tag information of the game
lang		String	2	Two-letter language code used to identify language which will be utilized to translate messages presented to player
version		Integer	11	Protocol version

In the following way game server will request operator system:

[https://host/appendwagerresult.json?  
org=YourOrgName&playerid=D80187CDEF79499&amount=1.01&bonusprize=0.00&currency=&tickets=1&service=Mir&reference=16032313+partial+result&cat1=Betting&cat2=SportsBook&cat3=SuperLive&cat4=BetPump&cat5=10000&tag1=GameName.BetPump&tag2=Model.M1&tag3=](https://host/appendwagerresult.json?org=YourOrgName&playerid=D80187CDEF79499&amount=1.01&bonusprize=0.00&currency=&tickets=1&service=Mir&reference=16032313+partial+result&cat1=Betting&cat2=SportsBook&cat3=SuperLive&cat4=BetPump&cat5=10000&tag1=GameName.BetPump&tag2=Model.M1&tag3=)

or

<https://host/appendwagerresult.json?org=YourOrgName&playerid=D80187CDEF79499&amount=0.00&bonusprize=100001.66&currency=&tickets=1&service=Mir&reference=161+and+the+Stolen+Stones&cat5=10000&tag1=GameName.Joker+Millions&tag2=Model.M1&tag3=Channel.web&lang=en&version=3>

Expected response parameters:

Parameter	Version	Format	Max Length	Comment
organization		String	32	Organization identifier(s)
playerId		String	20	The player id as returned from an authenticate call
currency		String	3	The three letter code of the currency betted.
applicableBonus		String	20	Amount of bonus money available to player for the game referred in the request.
homeCurrency		String	3	The three letter code of the player's default currency.
balance		String	20	Amount of withdrawable money available to player
nickName		String	64	The nick name of the player, to be displayed in Table Games (optional).
bonus		String	20	Bonus that player receive when make deposit to his wallet.

Example of response send back by operator's system:

```
{
  "code": 0,
  "data": {
    "currency": "eur",
    "applicableBonus": 12.95,
    "homeCurrency": "eur",
    "organization": "xxx",
    "balance": 105.05,
    "nickName": "xxx",
    "playerId": "el_54696_eur",
    "bonus": 0
  }
}
```

**When player wins jackpot, communication flow will look as follows (example of a scenario):**

Appendwagerresult request send by Yggdrasil Game Server to the operator system:

<https://host/appendwagerresult.json?org=YourOrgName&&playerid=0000000000000&amount=461.27&bonusprize=0.00&currency=EUR&tickets=0&service=Mir&reference=161+and+the+Stolen+Stones&cat5=7317&tag1=GameName.Holmes+and+the+Stolen+Stones&tag2=Model.M1&tag3=Channel.pc&lang=en&version=3>

Response from operator system:

```
{
  "code": 0,
```

```

"data": {
  "playerId": "0000000000000",
  "nickName": "xxx",
  "organization": "XXX",
  "balance": 492.35,
  "applicableBonus": 0.0,
  "currency": "EUR",
  "homeCurrency": "EUR"
}
}

```

## 4.5 endwager

Operator's integration team is obliged to implement *endwager.json* in order to run gameplay. The parameters that are sent by Yggdrasil Game Server and parameters that should be send back by Operator's Transaction System are described in depth below.

The Operator Transaction System should pay out the amount unless the same **endwager** (identified by reference and subreference) has already been processed. The payout, the amount and the reservation may be of cash or of bonus or of any other kind.

The operator system should preferably credit bonus money in place of cash if the corresponding wager was paid for by bonus. And if there is a currency conversion, probably the same rate should be applied as when the bet was made. The reply should reflect the same currency and any applicable bonus for the game according to the cat1..cat9 classification (see above) .

Bonusprize is used when the game has an extra payout that should normally not be withdraw-able, such as an achievement bonus or the winnings from a prepaid ticket, on top of any bonus that may have resulted from the cash amount paid out. Note that the operator platform has to decide and manage any wagering requirements.

Above added parameters marked with version 5 are related with super spin game play. They will be added to the operator request once the wallet version will be changed to 5 and super spin stop criteria will be achieved.

This is the last transaction relating to the wager.

Parameters in endwager request send by the Yggdrasil Game Server:

Parameter	Version	Format	Max Length	Comment
org		String	32	Organization identifier(s)
playerid		String	20	The player id as returned from an authenticate call when the game was launched
amount		Money	22	The winnings of the bet
isJackpotWin		Boolean	4	True win is a jackpot
bonusprize	2	Money	22	The value of any bonus prize paid out, i.e. prizes that should not be withdrawable but only used for gambling.
currency		String	3	The three letter code of the currency won
tickets		Integer	11	The number of tickets, spins or rounds played. This has different meaning for different games and is intended for campaigns and for statistics.

## Restless Gambling API - Restless Gambling API

reference		String	20	Unique identifier of the result payout
subreference		String	50	Unique identifier within the wager of the transaction, if campaignpayout is enabled or transaction is without a win, this field is empty
description		String	128	Description of the winning, translated and meant for the player to read.
cat1..cat9		String	64	The category information describing the game played
tag1..tag9		String	64	The tag information of the game
lang		String	2	Two-letter language code used to identify language which will be utilized to translate messages presented to player
version		Integer	11	Protocol version
prepaidref	2	String	64	Reference that was supplied when prepaids were created
prepaidticketid	2	String	20	Id of the ticket used to pay for wager
singleWin	5	String	22	Single win value in player currency
totalWin	5	String	22	Total win value in player currency
roundCount	5	String	11	Round count played to achieve stop criteria
ruleType	5	String	10	Rule type of the SuperSpin. Possible values are SINGLEWIN, TOTALWIN, FEATUREWIN

In the following way game server will request operator system:

[https://host/endwager.json?org=YourOrgName&&playerid=cc\\_1\\_eur&amount=0.00&bonusprize=0.00&currency=EUR&tickets=1&service=Mir&reference=16031615102+Game&cat1=Casino&cat2=Slot&cat3=Avalanche&cat4=Incinerator&cat5=7320&tag1=GameName.Incinerator&tag2=Model.M1&tag3=Chan](https://host/endwager.json?org=YourOrgName&&playerid=cc_1_eur&amount=0.00&bonusprize=0.00&currency=EUR&tickets=1&service=Mir&reference=16031615102+Game&cat1=Casino&cat2=Slot&cat3=Avalanche&cat4=Incinerator&cat5=7320&tag1=GameName.Incinerator&tag2=Model.M1&tag3=Chan)

Expected response parameters:

Parameter	Version	Format	Max Length	Comment
organization		String	32	Organization identifier(s)
playerId		String	20	The player id as returned from an authenticate call
currency		String	3	The three letter code of the currency betted.
applicableBonus		String	20	Amount of bonus money available to player for

				the game referred in the request.
homeCurrency		String	3	The three letter code of the player's default currency.
balance		String	20	Amount of withdrawable money available to player
nickName		String	64	The nick name of the player, to be displayed in Table Games (optional).
gameSessionBalance		String	64	Required ONLY for SPAIN (jurisdiction)
gameParticipation		String	64	Required ONLY for SPAIN (jurisdiction)
gamePrizes		String	64	Required ONLY for SPAIN (jurisdiction)

Example of response send back by operator's system:

```
{
  "code": 0,
  "data": {
    "currency": "eur",
    "applicableBonus": 980.0,
    "homeCurrency": "eur",
    "organization": "Demo",
    "balance": 1235.0,
    "nickName": "xxpq8n6ebekekuj",
    "playerId": "cc_360551_eur"
  }
}
```

## 4.6 campaignpayout

Operator's integration team is obliged to implement *campaignpayout.json* in order to be able to award winnings from prepaids to player in one transaction. The parameters that are sent by Yggdrasil Game Server and parameters that should be send back by Operator Transaction System are described in depth below.

In case of campaignPayout all win amount from prepaids will be sent as one transaction when player used all his prepaids(free spins). Without it, transaction will be sent after each used prepaid and win amount will be added to player immediately.

The operator transaction system should credit the player account with a combination of cash, bonus or goods as supplied in the api call. The "reference" parameter uniquely defines the campaignpayout.

Parameters in campaignpayout request send by the Yggdrasil Game Server:

Parameter	Version	Format	Max Length	Comment
org		String	64	Organization identifier(s)
playerid		String	20	The player id as returned from an authenticate call when the game was launched
cash		Money	22	Cash money to be paid out or zero

## Restless Gambling API - Restless Gambling API

bonus		Money	22	Bonus money to be paid out or zero
currency		String	3	The three letter code of the currency.
reference		String	64	Unique identifier of the payout
description		String	128	Description of the payout, translated and meant for the player to read.
cat1..cat9		String	64	The category information describing the game played
tag1..tag9		String	64	The tag information of the game
campaignref		String	64	A reference to the campaign that originated the payout(s)
last		String	1	Flag indicating if it was the last payout on the payouts associated with 'campaignref'. Values "Y" or "N".
lang		String	2	Two-letter language code used to identify language which will be utilized to translate messages presented to player
version		Integer	11	Protocol version
prepaidref	2	String	64	Reference that was supplied when prepaids were created
prepaidticketid	2	String	20	Id of the ticket used to pay for wager
singleWin	5	String	22	Single win value in player currency
totalWin	5	String	22	Total win value in player currency
roundCount	5	String	11	Round count played to achieve stop criteria
ruleType	5	String	10	Rule type of the SuperSpin. Possible values are SINGLEWIN, TOTALWIN, FEATUREWIN

In the following way Yggdrasil Game Server will request operator system:

[https://host/campaignpayout.json?org=YourOrgName&&playerid=cc\\_360471\\_eur&cash=0.00&bonus=1.02&currency=eur&service=Mir&reference=1603211008430800001&des+spin+win+ref%3A+1603211008150200001&campaignref=1603211008150200001&last=Y&cat1=Casino&cat2=Slot&cat3=Avalanche&cat4=Incinerator&cat5=7](https://host/campaignpayout.json?org=YourOrgName&&playerid=cc_360471_eur&cash=0.00&bonus=1.02&currency=eur&service=Mir&reference=1603211008430800001&des+spin+win+ref%3A+1603211008150200001&campaignref=1603211008150200001&last=Y&cat1=Casino&cat2=Slot&cat3=Avalanche&cat4=Incinerator&cat5=7)

Expected response parameters:

Parameter	Version	Format	Max Length	Comment
-----------	---------	--------	------------	---------



organization		String	32	Organization identifier(s)
playerId		String	20	The player id as returned from an authenticate call
currency		String	3	The three letter code of the currency betted.
applicableBonus		String	20	Amount of bonus money available to player for the game referred in the request.
homeCurrency		String	3	The three letter code of the player's default currency.
balance		String	20	Amount of withdrawable money available to player
nickName		String	64	The nick name of the player, to be displayed in Table Games (optional).

Example of response send back by operator's system:

```
{
  "code": 0,
  "data": {
    "currency": "eur",
    "applicableBonus": 1.02,
    "homeCurrency": "eur",
    "organization": "Demo",
    "balance": 0.0,
    "nickName": "a-2db3-a.8.2crx",
    "playerId": "cc_360471_eur"
  }
}
```

## 4.7 getbalance

Operator's integration team is obliged to implement `getbalance.json` in order to run gameplay. The parameters that are send by Yggdrasil Game Server and parameters that should be send back by Operator Transaction System are described in depth below.

Parameters in `getbalance` request send by the Yggdrasil Game Server:

Parameter	Version	Format	Max Length	Comment
org		String	32	Organization identifier(s)
sessiontoken		String	32	The unique token key to identify and validate the player session
playerid		String	20	The player id as returned from an authenticate (playerinfo) call
gameid		String	4	Game identifier
description		String	20	Description of the action

In the following way game server will request operator system:

<https://host/getbalance.json?org=YourOrgName&sessiontoken=1F31E37F7C87A74B9EC380373AE5DEA7&playerid=xxxxx&gameid=7330&description=getbalance>

Expected response parameters:

Parameter	Version	Format	Max Length	Comment
organization		String	32	Organization identifier(s)
playerId		String	20	The player id as returned from an authenticate (playerinfo) call
currency		String	3	The three letter code of the currency betted
homeCurrency		String	3	The three letter code of the player's default currency
applicableBonus		String	20	Amount of bonus money available to player for the game referred in the request.
balance		String	20	Amount of withdrawable money available to player
nickName		String	64	The nick name of the player, to be displayed in Table Games (optional).
bonus		String	20	Amount of bonus money available to player (alternative field to applicableBonus)

Example of response send back by operator's system:

```
{
  "code": 0,
  "data": {
    "currency": "EUR",
    "applicableBonus": 12.95,
    "homeCurrency": "EUR",
    "organization": "XXX",
    "balance": 105.05,
    "nickName": "XXX",
    "playerId": "XXX",
    "bonus": 0
  }
}
```

## 5. Launching games

In order to launch game please use one of below urls.

Environment name	URL	Game mode
Staging MT	<a href="https://staticstaging.yggdrasilgaming.com/init/launchClient.html?">https://staticstaging.yggdrasilgaming.com/init/launchClient.html?</a>	Play for real
Production MT	<a href="https://staticlive.yggdrasilgaming.com/init/launchClient.html?">https://staticlive.yggdrasilgaming.com/init/launchClient.html?</a>	Play for real
Staging Gib	<a href="https://staticstaginggib.yggdrasilgaming.com/init/launchClient.html?">https://staticstaginggib.yggdrasilgaming.com/init/launchClient.html?</a>	Play for real
Production Gib	<a href="https://staticlivegib.yggdrasilgaming.com/init/launchClient.html?">https://staticlivegib.yggdrasilgaming.com/init/launchClient.html?</a>	Play for real

## Restless Gambling API - Restless Gambling API

Staging CW	<a href="https://staticstagingcw.yggdrasilgaming.com/init/launchClient.html?">https://staticstagingcw.yggdrasilgaming.com/init/launchClient.html?</a>	Play for real
Production CW	<a href="https://staticlivecw.yggdrasilgaming.com/init/launchClient.html?">https://staticlivecw.yggdrasilgaming.com/init/launchClient.html?</a>	Play for real
Staging UK	<a href="https://staticstaginguk.yggdrasilgaming.com/init/launchClient.html?">https://staticstaginguk.yggdrasilgaming.com/init/launchClient.html?</a>	Play for real
Production UK	<a href="https://staticliveuk.yggdrasilgaming.com/init/launchClient.html?">https://staticliveuk.yggdrasilgaming.com/init/launchClient.html?</a>	Play for real
Staging MTIT (Italy)	<a href="https://staticstagingmtit.yggdrasilgaming.com/init/launchClient.html?">https://staticstagingmtit.yggdrasilgaming.com/init/launchClient.html?</a>	Play for real
Production MTIT (Italy)	<a href="https://staticlivemt.it.yggdrasilgaming.com/init/launchClient.html?">https://staticlivemt.it.yggdrasilgaming.com/init/launchClient.html?</a>	Play for real
Staging Asia	<a href="https://staticstagingsg.ygg-7ehd83n.com/init/launchClient.html?">https://staticstagingsg.ygg-7ehd83n.com/init/launchClient.html?</a>	Play for real
Production Asia	<a href="https://staticlivesg.ygg-7ehd83n.com/init/launchClient.html?">https://staticlivesg.ygg-7ehd83n.com/init/launchClient.html?</a>	Play for real
Staging Denmark	<a href="https://staticstagingdk.yggdrasilgaming.com/init/launchClient.html?">https://staticstagingdk.yggdrasilgaming.com/init/launchClient.html?</a>	Play for real
Production Denmark	<a href="https://staticlivedk.yggdrasilgaming.com/init/launchClient.html?">https://staticlivedk.yggdrasilgaming.com/init/launchClient.html?</a>	Play for real
Staging Sweden	<a href="https://staticstagingse.yggdrasilgaming.com/init/launchClient.html?">https://staticstagingse.yggdrasilgaming.com/init/launchClient.html?</a>	Play for real
Production Sweden	<a href="https://staticlivese.yggdrasilgaming.com/init/launchClient.html?">https://staticlivese.yggdrasilgaming.com/init/launchClient.html?</a>	Play for real
Staging Spain	<a href="https://staticstagingses.yggdrasilgaming.com/init/launchClient.html?">https://staticstagingses.yggdrasilgaming.com/init/launchClient.html?</a>	Play for real
Production Spain	<a href="https://staticlivees.yggdrasilgaming.com/init/launchClient.html?">https://staticlivees.yggdrasilgaming.com/init/launchClient.html?</a>	Play for real
Staging Czech	<a href="https://staticstagingcz.yggdrasilgaming.com/init/launchClient.html?">https://staticstagingcz.yggdrasilgaming.com/init/launchClient.html?</a>	Play for real
Production Czech	<a href="https://staticlivecz.yggdrasilgaming.com/init/launchClient.html?">https://staticlivecz.yggdrasilgaming.com/init/launchClient.html?</a>	Play for real
<b>Environment name</b>	<b>URL</b>	<b>Game mode</b>
PFF Malta	<a href="https://staticpff.yggdrasilgaming.com/init/launchClient.html?">https://staticpff.yggdrasilgaming.com/init/launchClient.html?</a>	Play for fun
PFF Curacao	<a href="https://staticpffcw.yggdrasilgaming.com/init/launchClient.html?">https://staticpffcw.yggdrasilgaming.com/init/launchClient.html?</a>	Play for fun
PFF UK	<a href="https://staticpffuk.yggdrasilgaming.com/init/launchClient.html?">https://staticpffuk.yggdrasilgaming.com/init/launchClient.html?</a>	Play for fun
PFF Gibraltar	<a href="https://staticpffgib.yggdrasilgaming.com/init/launchClient.html?">https://staticpffgib.yggdrasilgaming.com/init/launchClient.html?</a>	Play for fun
PFF MTIT (Italy)	<a href="https://staticpffit.yggdrasilgaming.com/init/launchClient.html?">https://staticpffit.yggdrasilgaming.com/init/launchClient.html?</a>	Play for fun
PFF Denmark	<a href="https://staticpffdk.yggdrasilgaming.com/init/launchClient.html?">https://staticpffdk.yggdrasilgaming.com/init/launchClient.html?</a>	Play for fun
PFF Sweden	<a href="https://staticpffse.yggdrasilgaming.com/init/launchClient.html?">https://staticpffse.yggdrasilgaming.com/init/launchClient.html?</a>	Play for fun
PFF Spain	<a href="https://staticpffes.yggdrasilgaming.com/init/launchClient.html?">https://staticpffes.yggdrasilgaming.com/init/launchClient.html?</a>	Play for fun
PFF Czech	<a href="https://staticpffcz.yggdrasilgaming.com/init/launchClient.html?">https://staticpffcz.yggdrasilgaming.com/init/launchClient.html?</a>	Play for fun
PFF Asia	<a href="https://staticpffsg.ygg-7ehd83n.com/init/launchClient.html?">https://staticpffsg.ygg-7ehd83n.com/init/launchClient.html?</a>	Play for fun

### IMPORTANT!

We assume each environment is connected to the separate database. Transactions i.e the pair of reference and subreference are unique per each environment.

Sample call for play for real mode on staging cw environment: <https://staticstagingcw.yggdrasilgaming.com/init/launchClient.html?gameid=7301&lang=sv&currency=EUR&org=YourOrgName&channel=pc&key=xxx>

One has to replace org & key parameters in the above url.

org - name of organization in the Yggdrasil system

key - session key provided by the operator

To launch mobile version of the games simply change "channel=pc" to "channel=mobile".

Sample call for play for fun mode: <https://staticpff.yggdrasilgaming.com/init/launchClient.html?gameid=7301&lang=en&currency=EUR&org=YourOrgName&channel=pc&key=>

---

## 6. Jackpot

There is available jackpot feed functionality that is able to return specified jackpot or all jackpots within current top organization, it is using two parameters:

- org – name of the organization (for the local progressive jackpots, Yggdrasil Game Server extracts top organization and use this to return jackpot value)
  - gameid – there can be different variances of jackpot games, thus gameid will be used and returned for different jackpot ids related with the same game
- 

### 6.1 Jackpot feed for Joker Milions (only available for clients with Maltese license)

The url addresses of the jackpot feeds are (please remember to replace org\_name value in the url with the correct brand name):

#### Production Malta

<https://production.yggdrasilgaming.com/game.web/services/feed/jackpot?org=YourOrgName&gameid=7312&currency=EUR>

If you prefer to use json instead you can use this service:

<https://production.yggdrasilgaming.com/game.web/service?fn=info&gameid=7312&what=jackpot&currency=EUR&org=YourOrgName>

### 6.2 Jackpot feed for all games

#### Production Malta

All jackpots <https://production.yggdrasilgaming.com/game.web/services/feed/jackpot?org=YourOrgName>

Jackpot/s related with particular game <https://production.yggdrasilgaming.com/game.web/services/feed/jackpot?org=YourOrgName&gameid=7317>

If you prefer to use json instead you can use this service:

<https://production.yggdrasilgaming.com/game.web/service?fn=info&gameid=7317&what=jackpot&currency=EUR&org=YourOrgName>

#### Production Curacao

All jackpots <https://productioncw.yggdrasilgaming.com/game.web/services/feed/jackpot?org=YourOrgName>

Jackpot/s related with particular game <https://productioncw.yggdrasilgaming.com/game.web/services/feed/jackpot?org=YourOrgName&gameid=7317>

If you prefer to use json instead you can use this service:

<https://productioncw.yggdrasilgaming.com/game.web/service?fn=info&gameid=7317&what=jackpot&currency=EUR&org=YourOrgName>

#### **Production United Kingdom**

All jackpots <https://productionuk.yggdrasilgaming.com/game.web/services/feed/jackpot?org=YourOrgName>

Jackpot/s related with particular game <https://productionuk.yggdrasilgaming.com/game.web/services/feed/jackpot?org=YourOrgName&gameid=7317>

If you prefer to use json instead you can use this service:

<https://productionuk.yggdrasilgaming.com/game.web/service?fn=info&gameid=7317&what=jackpot&currency=EUR&org=YourOrgName>

#### **Production Gibraltar**

All jackpots <https://productiongib.yggdrasilgaming.com/game.web/services/feed/jackpot?org=YourOrgName>

Jackpot/s related with particular game <https://productiongib.yggdrasilgaming.com/game.web/services/feed/jackpot?org=YourOrgName&gameid=7317>

If you prefer to use json instead you can use this service:

<https://productiongib.yggdrasilgaming.com/game.web/service?fn=info&gameid=7317&what=jackpot&currency=EUR&org=YourOrgName>

#### **Production Italy MT**

All jackpots <https://productionmtit.yggdrasilgaming.com/game.web/services/feed/jackpot?org=YourOrgName>

Jackpot/s related with particular game <https://productionmtit.yggdrasilgaming.com/game.web/services/feed/jackpot?org=YourOrgName&gameid=7317>

If you prefer to use json instead you can use this service:

<https://productionmtit.yggdrasilgaming.com/game.web/service?fn=info&gameid=7317&what=jackpot&currency=EUR&org=YourOrgName>

#### **Production Asia**

All jackpots <https://productionsg.ygg-7ehd83n.com/game.web/services/feed/jackpot?org=YourOrgName>

Jackpot/s related with particular game <https://productionsg.ygg-7ehd83n.com/game.web/services/feed/jackpot?org=YourOrgName&gameid=7317>

If you prefer to use json instead you can use this service:

<https://productionsg.ygg-7ehd83n.com/game.web/service?fn=info&gameid=7317&what=jackpot&currency=EUR&org=YourOrgName>

For **stage** environments please replace:

<https://production.yggdrasilgaming.com> with <https://staging.yggdrasilgaming.com> for Malta environment

<https://productioncw.yggdrasilgaming.com> with <https://stagingcw.yggdrasilgaming.com> for Curacao environment

<https://productionuk.yggdrasilgaming.com> with <https://staginguk.yggdrasilgaming.com> for United Kingdom environment

<https://productiongib.yggdrasilgaming.com> with <https://staginggib.yggdrasilgaming.com> for Gibraltar environment

<https://productionmtit.yggdrasilgaming.com> with <https://stagingmtit.yggdrasilgaming.com> for Italy environment

<https://productionsg.ygg-7ehd83n.com> with <https://stagingsg.ygg-7ehd83n.com> for Asia environment

<https://productiondk.yggdrasilgaming.com> with <https://stagingdk.yggdrasilgaming.com> for Danish environment

<https://productionse.yggdrasilgaming.com> with <https://stagingse.yggdrasilgaming.com> for Swedish environment

<https://productiones.yggdrasilgaming.com> with <https://staginges.yggdrasilgaming.com> for Spanish environment

<https://productioncz.yggdrasilgaming.com> with <https://stagingcz.yggdrasilgaming.com> for Czech environment

---

## 7. Bingo

Bingo does not support "data.popupMessage" sent in requests: playerinfo and wager. Bingo is always launched in the iframe. Therefore CMA requirement to inform the players if they are moving from playing with restricted funds to unrestricted and vice versa should be fulfilled on operator's site.

---

## 8. Typical integration plan

In the beginning, Yggdrasil prepares configuration for operator in order to make it possible to use stage environment. Yggdrasil developers/support team from this moment are at disposal of operator's integration team to resolve any issues, usually by email or skype (we prefer to contact by skype)

In the next step, operator's team prepares game play – implements playerinfo, wager, endwager, cancelwager and appendwagerresult (for jackpot games) in accordance to this document. Operator's team also prepares test accounts for Yggdrasil developers/testers (preferably each account has different currency)

Integration team on Yggdrasil side conducts idempotency tests, also the correctness of responses to Yggdrasil Game Server requests is being verified. Operator's developers fix any remaining issues.

Operator's integration team implements Prepays/Campaign API which is described here: <https://doc.yggdrasilgaming.com/xwiki/bin/view/Campaign+API/>

Yggdrasil's Quality Assurance Team conducts tests (Stage Environment) to check if:

- gameplay works properly on desktop and mobile devices (running on iOS and Android)
- prepaids are properly handed out
- Back Office works correctly for operator
- BOOST tools (Tournament, Cash Race & Missions) work properly

After successfully passing tests, operator's configuration will be released on production environment as soon as possible (usually when Yggdrasil releases changes to production environment – at the moment it is once per month)

At this point, both sides conduct hidden beta tests on production environment to ensure before going live, that everything (game play, BOOST, BackOffice) works properly. Finally, after passing tests, games go live on production environment.

---

## 9. Frequently asked questions

1. Is Cat5 the round number? Does all wagers and its matching appendwager requests (for a spin) have the same number? Is it unique in yggdrasil system or will it repeat between games or players?

Cat5 contains gameid. Wager and appendwagerresult calls have matching reference parameter.

2. What are possible values of Tag3 in game tags? Is it only Channel.PC and Channel.Mobile? Is it based on the value

Yes to all of these

3. Is "Sessiontoken" in wager request the same as "key" that was sent to the game launcher?

Yes, key = sessiontoken

4. Is "Reference" unique in the system? What is the difference between Reference and Subreference?

## Restless Gambling API - Restless Gambling API

Reference is wager's identifier and will look like this '1701170925270000003'. Reference should be unique, but you should look on reference and subreference as unique pair.

For subreference we have few possibilities

- \* in wager and cancelwager call, subreference is transaction reference and we create it by incrementing reference id by one and adding 'w' at the beginning (it will look like this 'w1701170925270000004'). If we cancel particular wager, reference and subreference will be exactly the same in cancelwager call.

- \* endwager and appendwagerresult

In endwager or appendwager call subsequent to wager call, the reference doesn't change (if wager had reference='1701170925270000003', endwager to this wager will also have reference='1701170925270000003'), }}

Subreference is different - usually when player has won it will consist of two parts: transaction reference and prize's identifier and will look like this '1701170845560000003.1701170845560000004'. In case player has lost it will be empty

5. "Bonusprize" field, under what condition will there be bonus payout?

There will be bonus payout when there are winnings from a prepaid ticket,

6. PlayerInfo response, can I assume below are the mandatory fields?

```
{ "code": 0, "data": { } "playerId": "40000001", "nickName": "Restless", "organization": "Demo", "balance": "4000.00", "applicableBonus": "0.00", "currency": "EUR", "homeCurrency": "EUR" }
```

Yes

7. Is nickname field required in the wager/end wager/cancelwager response?

You may leave there empty string.

8. When description field contains CashRace, Tournament, NetworkTournament, NetworkMission, Boost, how do we tag the

You shouldn't count the winning amount to game win when one of above is present in description field. It's up to you how to tag winning amount in your system.

However, you shouldn't categorize that as normal win, because it will negatively affect RTP. Any other type is fine. we don't have any preference.

9. Could you explain a bit more about campaignpayout?

It is used for freespins functionality.

Normally in freespins we send after every spin endwager call, but on request from operator we can turn campaignpayout functionality, which sends winnings from freespins in one request instead of multiple requests.

In order to have payout via 1 request from freespins, all you need to do is implement campaignpayout and ask for enabling this feature and we will set it up for you on our side.

10. How long does it take for sessiontoken to expire?

Sessiontoken expiration time is managed by operator's side. When sessiontoken expires and player tries to spin, game will present error and he will have to reload game. Every authentication call should provide new active sessiontoken.

11. Is lang and currency parameter mandatory in order to launch game?

Yes, both language and currency is obligatory when launching games.

12. Regarding cancel request, do we have to cancel all transactions with same reference or cancel only transactions w

All transactions with same reference and subreference.

13. What will happen in case of request timeout from our side, will you be resending the same request or cancel it?

If endwager timed out, we will resend same request. If it was wager, we will cancel it.

14. What is exactly prepaid ticket? How it will effect our system? Do we have to do any deduction from user balance?

Prepaidticketid is one of parameters connected with our prepaid api functionality. Prepaidticketid in wager request provides a way to identify prepaid ticket used by player.

When using our prepaid api you don't need to do any deduction from player's balance.

15. What is jackpotcontribution parameter in wager request? Does it have any impact on player's balance?

Jackpot contribution is a functionality that can be turned on per operator's request. This field is for informative purposes, shows how much of player's bet goes to prize pool. Works only for jackpot games.

## Restless Gambling API - Restless Gambling API

16. Is subreference unique?

Yes, it's unique, but you need to compare a pair of reference and subreference on your side. Wager and consecutive endwager have the same references, but different subreferences.

It is possible for endwager subreference to be empty when player has lost bet.

17. What should I do in case currency field is empty in wager request?

There is no possibility of this happening, there will be always currency in wager request.

18. What do you mean by "achievement bonus" in appendwagerresult?

What we meant by that is sometimes winning in appendwagerresult will be paid in amount parameter and sometimes in bonusprize field. It depends on game and our inner implementation.

19. What is SuperSpin, singleWin, totalWin, roundCount, ruleType?

Superspin is our type of prepaid. Please see prepaid documentation for more details.

singleWin, totalWin, roundCount, ruleType - those are extra parameters that can be turned on if operator requests for it. They are mainly for informative purposes,

20. What's your user inactivity timeout?

User session is active till the end of the game (to be specific: when player closes tab in browser or goes to another page)

21. What's your API-call timeout?

Yggdrasil Game Server API has two types of timeouts: RESTLESS\_CONNECTION\_TIMEOUT (specifies time limit during which our API needs to connect to operator's server) – 3s

and RESTLESS\_READ\_TIMEOUT (specifies how long our API awaits for input data to read) – 10s

22. Do you send secureToken(sessionToken) as part of every request like bet, win, cancel?

Most of our requests don't contain secureToken. Calls that do have secureToken: playerinfo (our authentication call), wager and getbalance. No, there is no possibility to configure it from operator's side.

24. What is the player experience in case of unfinished games?

When player launches game a pop-up window is shown in order to inform player that his session was interrupted and he can continue playing from where he stopped playing or play new session

25. Does the playerId need to be unique across labels?

Yes, the playerId must be unique across labels.

26. Do you send a win request in case Player didn't win?

Yes, we do send endwager calls even if player doesn't win anything.

27. Could you provide sample APIs call sequence needed in a gameplay of a normal game and jackpot game?

Regular gameplay: playerinfo -> response -> wager -> response -> wager .... -> endwager -> response -> end

For jackpot games: playerinfo -> response -> wager -> response -> appendwagerresult -> response -> endwager -> end