Betfair Exchange Games
API v.1.0
User Guide

Table 1 Document History

Document version	**Date**	**Principal Changes**
Version 1.0	September, 2006	This was the original document.
Version 1.1	February, 2007	This is the version of the document that accompanied the first update-release of the API in 2007. Information was added to the document on: the Total-size Request betting facility, Exchange Omaha Hi Poker, “channelSnapshot” parameters, “accountStatement” parameters, XML POST requests, and the “nextId” attribute for betting markets.
Version 1.2	July, 2007	This version covers changes to the method available for calling the betSnapshot service
Version 1.3	September 2008	Added Exchange Card Racing
Version 1.4	August 2010	Added Exchange Turbo Card Racing
Version 1.41	October 2010	Removed Games test bed information

Contents

Chapter 1 Introduction to the Betfair Exchange Games API 8
1.1 This Document and Its Intended Audience 9
1.2 The REST Architecture of the Betfair Exchange Games API 9
1.3 XML and the Use of the ‘&’ (Ampersand) Character in URLs 10
1.4 Version Numbering for Releases of the Exchange Games API 11

Chapter 2 Submitting Valid User and Client Application IDs 12
2.1 Accessing the API without Supplying User or Agent/Instance IDs 12
2.2 Specifying a Valid Username and Password 13
2.2.1 Getting a Betfair Username and Password 13
2.2.2 Including the Username in the URLs for your Client’s API Requests 13
2.2.3 Including the Betfair Password in the Message Header 13
2.3 Specifying Valid Agent and Agent Instance IDs 13
2.3.1 Including an Agent ID in the Message Header 13
2.3.2 Including an Agent Instance ID in the Message Header 14
2.4 A Sample GET Request with Agent and Other Headers Included 15
2.5 A Sample POST Request with Agent and Other Headers Included 16

Chapter 3 Accessing the API before Viewing Any Games 17
3.1 The Hierarchy of XML Elements in the Exchange Games API 17
3.2 The Format of the API’s Responses to GET Requests 18
3.3 Requesting XML Listing the Services Available from the API (api) 19
3.4 Requesting the XML Schema Definition (XSD) for the API (schema) 20
3.5 Requesting XML that Reports a “Ping” of the API (ping) 20

Chapter 4 Accessing the Exchange Games Using the API 22
4.1 Requesting XML Listing All the Games You Can Bet On (channelLanding) 22
4.2 Requesting XML Describing the Current Game in a Channel (channelSnapshot) 24
4.2.1 Controlling How Much Data is Returned in a “channelSnapshot” (channelSnapshot) 28
4.2.2 Requesting XML Listing *all* Prices (odds) for a Selection (selectionTradeActivity) 32
4.2.3 Requesting XML Listing Your Potential Profit and/or Loss (channelSnapshot) 35
4.3 Requesting XML Listing the Previous Games in a Channel (channelHistory) 42
4.3.1 Limiting the List of Past Games Returned (channelHistory) 43
4.3.2 Requesting the Record of a Specific Past Game (channelHistory) 43

4.4 Requesting XML Stating When Betting is Permitted in a Game (channelInfo) 44

Chapter 5 Exchange Texas Hold'em Poker 46

5.1 The URLs for Accessing a Game of Exchange Texas Hold'em Poker 46

5.2 The Rules Followed by Exchange Texas Hold'em Poker 46

5.2.1 The Progress of a Game of Exchange Texas Hold'em Poker 46

5.2.2 The System for Ranking Hands in Exchange Texas Hold'em Poker................................ 48

5.2.3 The Tie-breaker for Exchange Texas Hold'em Poker.. 49

5.2.4 The Dead-heat Rule for Bet Settlement in Exchange Texas Hold'em Poker 49

5.3 The Main XML Elements Used to Represent a Game of Exchange Texas Hold'em Poker . 49

5.3.1 The Automated Players and Other "object" Elements for Texas Hold'em Poker (object) 50

5.3.2 The "marketType" Element for Exchange Texas Hold'em Poker (market) 50

5.3.3 The "selection" Element for Exchange Texas Hold'em Poker (selection)......................... 51

5.3.4 Requesting XML Stating How Texas Hold'em Poker is Represented (channelInfo) 53

Chapter 6 Exchange Omaha Hi Poker.. 54

6.1 The URLs for Accessing a Game of Exchange Omaha Hi Poker 54

6.2 The Rules Followed by Exchange Omaha Hi Poker 54

6.2.1 The Progress of a Game of Exchange Omaha Hi Poker 54

6.2.2 The System for Ranking Hands in Exchange Omaha Hi Poker....................................... 54

6.2.3 The Tie-breaker for Exchange Omaha Hi Poker.. 55

6.2.4 The Dead-heat Rule for Bet Settlement In Exchange Omaha Hi Poker 55

6.2.5 The Two Betting Markets for Exchange Omaha Hi Poker 55

6.3 The Main XML Elements Used to Represent a Game of Exchange Omaha Hi Poker 56

6.3.1 The Automated Players and Other "object" Elements for Exchange Omaha Hi Poker 56

6.3.2 The "marketType" Element for Omaha Hi Poker (marketType) 56

6.3.3 The "selection" Element for Omaha Hi Poker (selection).. 56

6.3.4 Requesting XML Stating How Omaha Hi Poker is Represented (channelInfo) 57

Chapter 7 Exchange Blackjack 58

7.1 The URLs for Accessing a Game of Exchange Blackjack.. 58

7.2 The Rules Followed by the Betfair Game of Exchange Blackjack 58

7.2.1 The Progress of a Game of Exchange Blackjack 58

7.2.2 The Logic Followed by the Automated Players in Exchange Blackjack 60

7.2.3 The System for Ranking the Winning Hands in Exchange Blackjack............................... 62

7.2.4 The Dead-heat rule for Bet Settlement in Exchange Blackjack 62

7.3 The Main XML Elements Used to Represent a Game of Blackjack (channelSnapshot) 62

7.3.1 The Automated Players and Other “object” Elements for Blackjack (object) 63
7.3.2 The “marketType” Element for Blackjack (marketType) 63
7.3.3 The “selection” Element for Blackjack (selection) 63
7.3.4 Requesting XML Stating How Blackjack is Represented (channelInfo) 67

Chapter 8 Exchange Baccarat 68
8.1 The URLs for Accessing a Game of Baccarat 68
8.2 The Rules Followed by the Betfair Game of Exchange Baccarat 68
8.2.1 The Progress of a Game of Exchange Baccarat 68
8.2.2 The System for Ranking the Hands in Exchange Baccarat 70
8.2.3 The Settlement of Bets in the Event of a Tie in Exchange Baccarat 71
8.2.4 The Two Betting Markets for Exchange Baccarat 71
8.3 The Main XML Elements Used to Represent a Game of Baccarat 72
8.3.1 The Automated Players and Other “object” Elements for Baccarat (object) 72
8.3.2 The “marketType” Element for the Baccarat “MainBets” Market (marketType) 73
8.3.3 The “marketType” Element for the Baccarat “SideBets” Market (marketType) 74
8.3.4 The “selection” Element for Baccarat (selection) 74
8.3.5 Requesting XML Stating How Baccarat is Represented (channelInfo) 79

Chapter 9 Exchange Hi Lo 81
9.1 The URLs for Accessing a Game of Exchange Hi Lo 81
9.2 The Rules Followed by the Betfair Game of Hi Lo 81
9.2.1 The Progress of a Game of Exchange Hi Lo 81
9.2.2 The Strategy Followed by the Automated Player in Exchange Hi Lo 82
9.2.3 The Two Betting Markets for Exchange Hi Lo 84
9.3 The Main XML Elements Used to Represent a Game of Hi Lo 85
9.3.1 The “object” Element for Hi Lo (object) 85
9.3.2 The “marketType” Element for the “CorrectPredictions” Market in Hi Lo (marketType) .. 86
9.3.3 The “marketType” Element for the “SideBets” Market in Hi Lo (marketType) 88
9.3.4 The “selection” Element for Hi Lo (selection) 88
9.3.5 Requesting XML Stating How Hi Lo is Represented (channelInfo) 90

Chapter 10 Exchange Card Racing 91
10.1 The URLs for Accessing a Game of Exchange Card Racing 91
10.2 The Rules Followed by the Betfair Game of Exchange Card Racing 91
10.2.1 Before the Race Starts 91
10.2.2 Handicaps 92

10.2.3 During the Race 92
10.2.4 The Winner 92
10.2.5 Rules 92
10.2.6 Market 92
10.2.7 Race schedule 92
10.2.8 Odds Ladder 93
10.3 The Main XML Elements Used to Represent a Game of Exchange Card Racing 93
10.3.1 The “object” Element for Exchange Card Racing (object) 94
10.3.2 The “marketType” Element for the “MainBets” Market in Exchange Card Racing (marketType) 95
10.3.3 The “selection” Element for Exchange Card Racing (selection) 97
10.3.4 Requesting XML Stating How Exchange Card Racing is Represented (channelInfo) 97

Chapter 11 Managing Your Games Account Using the API 98
11.1 Requesting XML for Managing Your Accounts (accountLanding) 98
11.2 Requesting XML Listing Your Exchange Games Account Balances (accountSnapshot) 99
11.3 Requesting XML Stating Your Main Betfair Balance (betfairAvailableBalanceSnapshot) 101
11.4 Requesting XML That Lists Your Games Account Transactions (accountStatement) 102
11.5 Adding Parameters Manually to the URLs for Listing Transactions 104
11.5.1 Limiting the Transactions List to a Particular Type of Game (accountStatement) 104
11.5.2 Listing Only Your Fund Transfers (accountStatement) 104
11.5.3 Requesting a Specific Number of Transactions (accountStatement) 105
11.5.4 Starting the List at a Specific Position in the Overall List (accountStatement) 105
11.6 Sending XML to Request a Transfer of Funds Between Accounts (postTransferOrder) 105

Chapter 12 Placing Bets Using the API 107
12.1 Requesting XML for Managing Bets (betLanding) 107
12.2 Sending XML to Place, Edit, and/or Cancel Bets (postBetOrder) 107
12.3 Sending XML to Make Total-size Bet Requests (postBetOrder) 110
12.3.1 Adjusting Your Stake after Sending a Total-size Request 111
12.3.2 How the Total-size Request Facility Works 111
12.3.3 Mixing Bet and Total-size Requests in a Single “postBetOrder” 111
12.3.4 Sending Bet and Total-size Requests Close Together in Time 112
12.3.5 The Application of Betfair’s “Best Execution” Policy to Total-size Requests 112
12.3.6 How the Exchange Server Calculates the Existing Stake-size 113
12.3.7 Using a Total-size Request to Lay (Instead of Back) a Selection 113
12.3.8 Total-size Request Examples and Scenarios 113

12.3.9 Sending XML Containing a Mixture of Betting Instructions 118

12.4 The Minimum Bets for Different Currencies 119

12.5 The Odds at which You Can Place and Edit Bets 120

Chapter 13 Viewing Your Current and Past Betting Activity 121

13.1 Requesting XML for Viewing Bets (betLanding) 121

13.2 Requesting XML That Lists Current Bets by Status (betSnapshot) 121

13.2.1 Listing Only Those Current Bets Whose Details Have Changed (betSnapshot) 123

13.2.2 Listing Current Bets for a Particular Channel (betSnapshot) 124

13.2.3 Listing Current Bets for a Particular Channel and Market (betSnapshot) 124

13.3 Requesting XML that Lists Past Bets by Status (betHistory) 125

13.4 Adding Parameters Manually to the URLs for Listing Bets 127

13.4.1 Listing Past Bets Placed at or Between Particular Times (betHistory) 127

13.4.2 Requesting a Specific Number of Bets from a List (betSnapshot and betHistory) 127

13.4.3 Starting the List at a Specific Position in the Overall List (betSnapshot and betHistory) 127

13.4.4 Listing Current Bets by Game or Time Placed (betSnapshot) 128

13.4.5 Specifying a Time Zone for the Reporting of Your Bets (betHistory) 128

Appendix A The Error Messages Returned by the API 130

A.1 Standard HTTP Error Messages Returned by the API 130

A.2 The XML “error” Element Returned by the Games API (error) 131

A.3 Error (and Other) Responses to Betting POST Requests (responseBetOrder) 136

Appendix B Statuses for “object” and “selection” Elements 139

Appendix C Rules and Good Practice for API Developers 149

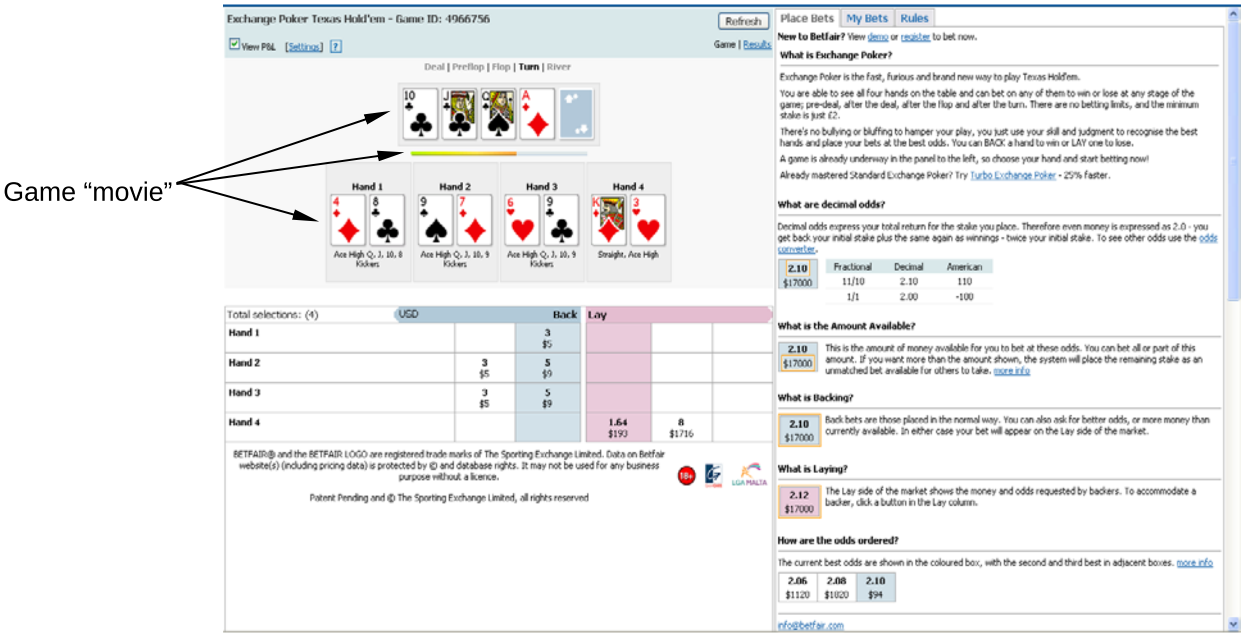
Chapter 1 Introduction to the Betfair Exchange Games API

Betfair Exchange Games is an online service provided by Betfair that enables members of the public to bet on the outcomes of automated card games. There are five games currently available to bet on: Exchange Texas Hold’em Poker, Exchange Omaha High Poker, Exchange Blackjack, Exchange Baccarat, and Exchange Hi-Lo, plus “turbo” versions of some of these games, in which the betting rounds are 25% faster. In addition, new games are being developed all the time.

The Betfair Exchange Games service is not an online casino. When you use the service you are not playing Poker or Blackjack (for example) yourself. Instead you are betting on which of a number of automated players will win or lose the game, or perhaps on which of a series of possible card combinations or other possible conditions, will prevail in the game. In Exchange Hi Lo, for instance, you are betting on the point in the game at which you think the automated player will fail to predict correctly whether the next card is higher or lower than the last.

The automated games follow rules that have in some cases been adapted by Betfair to make them suitable for exchange conditions. Before you place any bets, therefore, we recommend you read the rules followed by the Betfair versions of the games. These are given in the chapters below for each individual game.

The automated games are accessible via the Betfair Exchange Games website at games.betfair.com or you can go to the main Betfair website at www.betfair.com and click “Games” in the top right-hand corner of the home page.



Exchange Poker Texas Hold'em - Game ID: 4966756

Total selections: (4)	USD	Back	Lay		
Hand 1		3 $5			
Hand 2	3 $5	5 $9			
Hand 3	3 $5	5 $9			
Hand 4			1.64 $193	8 $1716	

BETFAIR® and the BETFAIR LOGO are registered trade marks of The Sporting Exchange Limited. Data on Betfair website(s) (including pricing data) is protected by © and database rights. It may not be used for any business purpose without a licence.

Patent Pending and © The Sporting Exchange Limited, all rights reserved

New to Betfair? View demo or register to bet now.

What is Exchange Poker?

Exchange Poker is the fast, furious and brand new way to play Texas Hold'em.

You are able to see all four hands on the table and can bet on any of them to win or lose at any stage of the game; pre-deal, after the deal, after the flop and after the turn. There are no betting limits, and the minimum stake is just £2.

There's no bullying or bluffing to hamper your play, you just use your skill and judgment to recognise the best hands and place your bets at the best odds. You can BACK a hand to win or LAY one to lose.

A game is already underway in the panel to the left, so choose your hand and start betting now!

Already mastered Standard Exchange Poker? Try Turbo Exchange Poker - 25% faster.

What are decimal odds?

Decimal odds express your total return for the stake you place. Therefore even money is expressed as 2.0 - you get back your initial stake plus the same again as winnings - twice your initial stake. To see other odds use the odds converter.

2.10
$17000

Fractional	Decimal	American
11/10	2.10	110
1/1	2.00	-100

What is the Amount Available?

2.10 $17000 This is the amount of money available for you to bet at these odds. You can bet all or part of this amount. If you want more than the amount shown, the system will place the remaining stake as an unmatched bet available for others to take. more info

What is Backing?

2.10 $17000 Back bets are those placed in the normal way. You can also ask for better odds, or more money than currently available. In either case your bet will appear on the Lay side of the market.

What is Laying?

2.12 $17000 The Lay side of the market shows the money and odds requested by backers. To accommodate a backer, click a button in the Lay column.

How are the odds ordered?

The current best odds are shown in the coloured box, with the second and third best in adjacent boxes. more info

2.06 $1120	2.08 $1020	2.10 $94

info@betfair.com

Figure 1 A “Movie” of a Game of Poker as Displayed by the Web Interface to the Exchange Games

However, the Exchange Games are now also available via an Application Programming Interface (API), and this means that you can write your own client application or user interface to access them. For example, you might want to write a 'Betting 'Bot'[1] that uses the API instead of Betfair's own web interface to place bets on the automated games.

It is the API that is the particular subject of this user guide. You might find it useful, while accessing the Games Exchange via the API, to watch the website "movie" representation of games that you are betting on (see Figure 1). The "movie" also includes dynamic information about the best back and lay prices for the possible betting selections.

NOTE: Unless otherwise specified, all references to the "API" in this document are to the Betfair Exchange Games API and not to the API for the Betfair Sports Exchange betting service.

1.1 This Document and Its Intended Audience

The purpose of this document is to describe the Exchange Games API in sufficient detail to enable you to write an application for it, assuming that you are a software developer with experience of an appropriate programming language such as Java, .NET, C#, or JavaScript.

The document does not tell you how to write a client application. It tells you what you need to know about the Exchange Games API if you already know how to write client applications.

Specifically, the document describes:

- The services available from the Games API,
- How to request each service (by making HTTP GET and POST requests using URLs provided by the Games API),
- And the form of the Games API's responses to HTTP GET and POST requests from a client application.

1.2 The REST Architecture of the Betfair Exchange Games API

The implementation of the Betfair Exchange Games API uses an architecture known as Representational State Transfer[2] (REST). This architecture is designed for communication between software applications across the internet. The API therefore - as an implementation of this architecture - works by responding to HTTP requests that it receives, for example, from a client application written by you.

REST services present a stateless interface - which, in this context, means that they do not remember the *state* of their client relationships from one interaction to the next. In other words, when a client application contacts the Exchange Games API, the API does not set up a session over which communication between itself and the client then takes place. The REST architecture has no concept of a session in which the server (in a client-server relationship) authenticates the client and then exchanges data freely with it until one or the other party terminates the session.

Instead, the Exchange Games API makes one-off replies to one-off requests. If it doesn't set up a communication session with a client application, however, it does the next best thing: in its responses to personalized HTTP requests it automatically includes the username associated with that client application in any URLs that it returns for the client application's subsequent use.

[1] For information about "Betting 'Bots" ("'Bots" is short for "Robots"), refer to the following URL: http://www.flatstats.co.uk/articles/betting_exchanges_3.html

[2] For information about the REST architecture, refer to the following URL: http://en.wikipedia.org/wiki/Representational_State_Transfer

The receiving client can then use these URLs to request further information without needing to modify them to incorporate the required username.

However, in developing your client application against the Games API, you need to make sure it incorporates the correct password (plus an agent ID and an agent instance ID) into the message header whenever it sends HTTP requests to the API. Although much of the information available from the API does not require a username, for information that does the API performs username and password authentication (and also agent-ID and agent-instance-ID authentication) on *every* HTTP request that it receives. For these requests, therefore, the username must be present in the body of the HTTP request, and the password, agent ID, and agent instance ID must be present in the header. (For information about incorporating the correct information into the message header, see Chapter 2.)

Most of the HTTP messages that the API can receive are simply requests for information: they use the HTTP GET command, which is for reading data.

However, a few are requests to write information to the Exchange Games API. These use the HTTP POST command, and it is these requests that enable you to place, edit, and cancel bets, and also to transfer money in and out of your Betfair Exchange Games account.

NOTE: For security reasons, the Exchange Games API only accepts HTTP messages that are encrypted as HTTPS. Therefore, in developing your client application, you will need to use a software library to manage SSL encryption between the client and the Exchange Games API.

The HTTP requests that the API supports, and the nature of the responses it can make, are all defined in an XML Schema[3] Definition document (XSD). (The schema document is available from the Exchange Games API itself; see Section 3.4.) The API's responses are therefore each in the form of XML, and they are structured according to the definitions in the schema. You will need to use the schema to enable your client to process the API's output.

It is assumed that most customers wishing to use the API will in the first instance run the schema through an object generator for their preferred programming or scripting language. Doing this creates a series of objects, and the client application you develop can then invoke these to communicate with the API.

1.3 XML and the Use of the '&' (Ampersand) Character in URLs

NOTE: If the software library you are using to process incoming XML does not convert XML entity reference strings back into the characters they represent, your client application must convert the string '`&amp;`' into the '&' character before it can use a URL that contains two or more parameters. For an explanation of why this is necessary, read the rest of this section.

The XML standard dictates that, within an XML page, certain data characters are represented by a string specifically defined for each and called an "entity reference". This is to distinguish them from the same characters in their function as XML special characters (in other words, in their function as part of an instruction to an application).

One of the characters that an XML entity reference is defined for is the '&' (ampersand) character and its entity reference is:

```
&amp;
```

When the Betfair Exchange Games API receives an HTTP request that contains an '&' character, it understands by this that any characters coming after the '&' character (until the next '&' or until the end of the string) identify and specify a parameter.

[3] For information about XML schemas, refer to the following URL: http://www.w3.org/XML/Schema

So, for example, if the Games API receives a request for the URL below, it interprets the string "`betStatus=MATCHED`" (at the end of the URL) as a parameter and, in this case, correctly returns to the requesting client application the user's list of currently matched bets:

`https://api.games.betfair.com/rest/v1/bet/snapshot?username=sampleuser&betStatus=MATCHED`

Now, although this is the form in which the Exchange Games API needs to *receive* and process URLs when a client application uses them to request a service, it is not the form in which the Games API *returns* URLs for a client application to use.

The Games API returns URLs for use by client applications inside the "href" attribute of an XML element called "resource" (see Section 3.3). Whenever it does this, if the URL includes a parameter, then the '&' character (which must precede the parameter when you use the URL) will in each case be represented by its entity reference ('`&amp;`'), as shown in bold type in the example below:

https://api.games.betfair.com/rest/v1/bet/snapshot?username=sampleuser**&amp**;betStatus=MATCHED

Thus, the form in which your client application receives some URLs from the Exchange Games API is different from the form in which it needs to use those URLs to access the API.

NOTE: Make sure that your client application does not send HTTP requests that contain XML entity references instead of ampersands.

1.4 Version Numbering for Releases of the Exchange Games API

From time to time, new versions of the Exchange Games API will be published as more games become available to bet on and as other enhancements are made to the product.

Because the version number of the API is included in the URLs that your client application will use to access the API, changes in the version number are likely to require some alteration in your client code.

Betfair will minimize any inconvenience of this kind caused to you by revisions of the API. They will do this by:

- Keeping the same version number in all URLs for accessing the API whenever incremental (and backwards-compatible) revisions of the API are published, and by
- Making both the new and the old API available at the same time, for a limited period, whenever a major new release of the API is published. This will give you time to develop new code against the new API without having to stop betting on games until you have modified your client to use the new API.

Chapter 2 Submitting Valid User and Client Application IDs

To place bets or access protected resources using the Games API, any client application you develop must submit certain information in the message header of its HTTP requests. It must also include a valid Betfair username in the URL that it sends the requests to.

The information that the client must include in the message header is: a Betfair password corresponding to the username included in the URL; an agent identifier for the client application (this is a name and version number for the software); and a unique identifier for the particular instance of the client application that the user is running.

This chapter tells you:

- Which API services you can access without submitting a username in the URL or specifying any of the header information required for accessing protected data.
- How to obtain a Betfair username and password and how to incorporate them into API requests.
- How to create acceptable agent and agent instance IDs and how to incorporate them into the message headers for API requests. (In this chapter the word “agent” is used interchangeably with “client” to designate a client application. This is because the header field entries you need to supply as identifiers of the client and client instance are called respectively “gamexAPIAgent” and “gamexAPIAgentInstance”.)

NOTE: When you log into the Betfair Games website at `http://games.betfair.com` using your Betfair username and password, you must accept the Betfair Games Terms and Conditions. The Exchange Games API will not accept your HTTP GET requests for protected data, or any POST requests, until you have accepted the Terms and Conditions on the Betfair Games website. You must also log out of the website after accepting the Terms and Conditions but before sending requests to the API. (There may be a delay of up to 15 minutes while Betfair process your acceptance.) Once you have started to use the API, you can log into the website to view the “movie” representations of the games that you are betting on.

2.1 Accessing the API without Supplying User or Agent/Instance IDs

You can access some services from the Betfair Exchange Games API without submitting any of the user and client ID information required for the protected areas of the API. However, you cannot perform any HTTP POST requests without submitting the necessary user and client ID information.

These are the services you can access without providing any user and client ID information:

- A list of the services available from the Games API (see 3.3).
- A copy of the XML Schema Definition (XSD) document for the Games API (see Section 3.4).
- A report of the operating status of the Games API (see Section 3.5).
- A list of the games currently available to bet on (see Chapter 4).
- A description of a current Exchange game of your choice (see Section 4.2).
- A list of the previous instances of a game for the last 24 hours (see Section 4.3).

To access all the other services available from the Games API, and in particular to place bets and manage your accounts, you need to submit both user and client application ID data.

NOTE: For the sake of consistency, all URLs in this document are printed with a sample username specified, even when the URL would be valid without one. If you intend to access the non-protected areas of the API and you do supply a username, then it must be valid and you must also supply a valid corresponding password in the header. Similarly, if you supply a password in the header, it must be valid and you must supply a valid corresponding username in the URL.

2.2 Specifying a Valid Username and Password

This section tells you how to get a Betfair username and password, and how to incorporate them both into the HTTP requests that your client sends to the API. The next section tells you how to complete the necessary header information by including agent and agent instance IDs.

NOTE: It is a condition of the use of the Games API that clients accessing it shall not store username and password details on static media. The user of the client must enter the user name and password manually each time he or she runs the client.

2.2.1 Getting a Betfair Username and Password

If you do not have a Betfair username and password, please register with Betfair on the website at www.betfair.com.

2.2.2 Including the Username in the URLs for your Client's API Requests

When the client you develop makes an HTTP GET request to the Games API for protected information or when it makes a POST request (to place, edit, or cancel a bet, for example), it must include a Betfair username in the target URL for the request, as shown below:

```
https://api.games.betfair.com/rest/v1?username=username
```

2.2.3 Including the Betfair Password in the Message Header

The Betfair password corresponding to the username submitted in the URL needs to be specified as a field entry in the message header for a request. Its format is as follows:

```
gamexAPIPassword: password
```

Where *password* is your Betfair password.

NOTE: Any HTTP GET requests that your client application makes and that include a password in the header must also include a valid username in the URL and a valid agent and instance ID in the header (see next section). In the same way, any HTTP GET requests that include a username must also include a password and a valid agent and instance ID (see next section). Finally, all HTTP POST requests that the client makes to the Games API must include a username, password, agent ID, and agent instance ID.

2.3 Specifying Valid Agent and Agent Instance IDs

This section tells you how to provide an agent ID and an agent instance ID in the message header of API requests.

2.3.1 Including an Agent ID in the Message Header

Each HTTP POST request or each HTTP GET request for protected information must include the following field entry in its message header:

```
gamexAPIAgent: application_ID_and_version_number
```

where *application_ID_and_version_number* is a URI (a Universal Resource Identifier) that identifies you as the creator of the application either by personal name (if you are a private individual) or by the name of your organization (if you have created the client application on behalf of an organization), and that includes at the end the version number of the software. Some sample agent IDs are:

```
ABCBets.com.GamesApp.1.0
ABCPartner1.com.GamesApp.1.1
123Games.co.uk.GamePal.2.1
maryBrown@AOL.com.myGames.4.0
```

2.3.2 Including an Agent Instance ID in the Message Header

The agent instance ID is a 32-character hexadecimal string that uniquely identifies each instance of a client. Your client application must generate a new agent instance ID each time it is installed.

The field entry header for the agent instance ID is as follows:

```
gamexAPIAgentInstance: application_instance_ID
```

where *application_instance_ID* is an MD5 digest formed according to the following procedure:

1. Take the current date and time expressed in the format below (note that the format includes hyphens and colon characters):

   YYYY-MM-DDThh:mm:ss.sTZD

   where:

   YYYY is a four-digit expression for the year (for example: 2006)
   MM is a two-digit expression for the month (for example: 01=January, 02=February, etc.)
   DD is a two-digit expression for the day in the month (for example: 01 through 31)
   T is a literal character indicating that the values after it specify the time
   hh is a two-digit expression for the hour (for example: 00 through 23)
   mm is a two-digit expression for the minute (for example: 00 through 59)
   ss is a two-digit expression for the second (for example: 00 through 59)
   s is one or more digits representing a decimal fraction of a second (for example: 34)
   TZD is a time-zone designator (for example, either Z *or* +hh:mm or -hh:mm)

   For example, the following string represents the time expressed in Coordinated Universal Time (UTC) at 9.13am on the morning of September 8, 2006:

   ```
   2006-09-08T09:13:53:26Z
   ```

   According to this expression of the time, it is 13 minutes, 53 seconds, and 26 milliseconds past 9 o'clock on the morning of September 8, 2006. The Z indicates that the time is being expressed in Coordinated Universal Time (UTC).

   If you are not expressing the time directly in UTC (and indicating that you are doing so by putting a Z at the end of the string), then you must include an offset instead of a Z at the end of the expression.

   For example, in the UK, the same date and time (during British Summer Time) would be expressed in the following way without using UTC:

   ```
   2006-09-08T10:13:53:26+01:00
   ```

   According to this expression, the time is 13 minutes past 10 a.m. on the same date as above, but in a local time-zone (British Summer Time) that is one hour ahead of Coordinated Universal Time (the offset is therefore +01:00).

   For a user in Canada, where the time-zone is Newfoundland Summer Time (NST), the same time would be expressed in the following way:

   ```
   2006-09-08T05:43:53:26-03:30
   ```

   According to this expression, the time is 17 minutes before 6 a.m. on the same date as above, but in a local time-zone (NST) that is three and a half hours behind Co-ordinated Universal Time (the offset is therefore -03:30).

For a user in Central Europe, where the time-zone is CET, the same time would be expressed in the following way:

```
2006-09-08T17:13:53:26+01:00
```

According to this expression, the time is 13 minutes after 5 p.m. on the same date as above, but in a local time-zone (CET) that is one hour ahead Co-ordinated Universal Time (the offset is therefore +01:00).

For more information about specifying the date and time in this format (which is defined by the World Wide Web Consortium), visit the following website: http://www.w3.org/TR/NOTE-datetime

2. Next append a five-digit randomly generated number to the string that contains the current date and time. For example, a five-digit number of 35932 appended to the expression of UTC from above would be as follows:

```
2006-09-08T09:13:53:26Z35932
```

3. Now append the string you have defined as your “gamexAPIAgent” ID (see Section 2.3.1) to the entire string from Step 2. For example:

```
2006-09-08T09:13:53:26Z35932maryBrown@AOL.com.myGames.4.0
```

4. Finally, take the string from Step 3 and use it as input to an MD5 hash generator to produce a unique 32 character hexadecimal string.

   The resulting string will form the value of the “gamexAPIApiAgentInstance” ID that you specify in the message header of your client’s API requests.

   The MD5 hash of the string from our example in Step 3 is as follows:

```
0d69ee8290ee2f9b336c1f060e3497a5
```

2.4 A Sample GET Request with Agent and Other Headers Included

An example of the GET request associated with an HTTPS call to the root element of the Games API is shown in Figure 2.

```
GET /rest/v1?username=sampleuser HTTP/1.1
Host: api.games.betfair.com
Accept-Encoding: gzip,deflate
Keep-Alive: 300
Connection: keep-alive
gamexAPIPassword: ********
gamexAPIAgent: maryBrown@AOL.com.myGames.4.0
gamexAPIAgentInstance: 0d69ee8290ee2f9b336c1f060e3497a5
```

Figure 2 A Sample GET Request (including headers) Associated with an Initial HTTPS Call to the API

For information about the format of the API’s responses, see Section 3.2.

For developer resources relating to MD5 hash generators, refer to the following website:

http://userpages.umbc.edu/~mabzug1/cs/md5/md5.html

2.5 A Sample POST Request with Agent and Other Headers Included

The format in which your POST requests are transmitted to the Games API will depend upon the software libraries you use to build your Games API client application. However, there are two basically different formatting methods. The first involves submitting the XML request as the body of an HTTP message (see Figure 3); and the second involves submitting XML as if it were part of an HTML form (see Figure 4). The first is the more robust and recommended method, but many software libraries implement the second.

Figure 3 illustrates the first method. It shows a POST request for performing a fund transfer (this service is described in Section 11.6).

```
POST /rest/v1/account/transferOrder?username=sampleuser HTTP/1.1
Host: api.games.betfair.com
Accept: */*
Content-type: application/xml; charset=UTF-8
gamexAPIPassword: ********
gamexAPIAgent: maryBrown@AOL.com.myGames.4.0
gamexAPIAgentInstance: 0d69ee8290ee2f9b336c1f060e3497a5
Content-Length: 214

<?xml version="1.0" encoding="UTF-8"?><postTransferOrder
xmlns="urn:betfair:games:api:v1" xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance"><amount currency="GBP"
action="TRANSFER">5</amount></postTransferOrder>
```

Figure 3 A Sample POST Request in which the XML is Submitted as the Body of an HTTP Message

NOTE: If your client application submits POST requests in the format shown in Figure 3, you must make sure that the `charset` value specified in the `Content-type` header corresponds with the type of encoding specified in the XML body (in this example the value is UTF-8).

Figure 4 illustrates the second method of formatting POST requests. In this method the body of the message is presented as a name=value pair, the name being `xmlRequest=` and the value being a URL-encoded version of the XML you are sending to the Games API. In Figure 4, therefore, the URL-encoded version of the XML in Figure 3 is shown.

```
POST /rest/v1/account/transferOrder?username=sampleuser HTTP/1.1 Host:
api.games.betfair.com
Accept: */*
Content-type: application/x-www-form-urlencoded; charset=UTF-8
gamexapipassword: ********
gamexAPIAgent: maryBrown@AOL.com.myGames.4.0
gamexAPIAgentInstance: 0d69ee8290ee2f9b336c1f060e3497a5
Content-Length: 309

xmlRequest=%3C%3Fxml+version%3D%221.0%22+encoding%3D%22UTF-
8%22%3F%3E%3CpostTransferOrder+xmlns%3D%22urn%3Abetfair%3Agames%3Aapi%3Av1%2
2+xmlns%3Axs%3D%22http%3A%2F%2Fwww.w3.org%2F2001%2FXMLSchema-
instance%22%3E%3Camount+currency%3D%22GBP%22+action%3D%22TRANSFER%22%3E5%3C%
2Famount%3E%3C%2FpostTransferOrder%3E
```

Figure 4 A Sample POST Request in which the XML is Submitted as if it were Part of an HTML Form

NOTE: If you submit POST requests in the format shown in Figure 4, you must make sure the values for the `Content-type` header are specified as “`application/x-www-form-urlencoded; charset=UTF-8`”, and that the `Content-Length` is specified accurately (in other words, that it correctly states the number of characters in the message body after it has been URL-encoded).

Chapter 3 Accessing the API before Viewing Any Games

This chapter provides an overview of the hierarchy of XML elements in the Exchange Games API. It also tells you:

- How to start using the API to view the services available from it,
- The format of HTTP responses from the API,
- How to retrieve the XML schema definition document (XSD) for the API, and
- How to “ping” the API to check its operating status and to test the client you are developing.

3.1 The Hierarchy of XML Elements in the Exchange Games API

The Games API works by returning XML pages in response to requests for information, or in response to requests to write information, to the Betfair Exchange Games API server. It is up to you, as the developer of a client application, to decide how to use the XML that comes back.

According to REST architecture good practice, the Betfair Exchange Games API is organized as a tree structure (see Figure 5). Its root element is called “api” and, when you access this root element by making an HTTPS request to its URL, it returns a list (in structured XML) of the main services available from the API.

This chapter tells you how to access the root element of the Exchange Games API and how to use it to:

- Request the API schema defining the structures of the data returned by the API. You can then pass the schema through an object generator to create classes of objects corresponding to the structures defined in the schema.
- Perform a “ping” of the Betfair Exchange Games API in order to: check the current operating status of the Games API, check your own username and password, and test the client application you are writing.

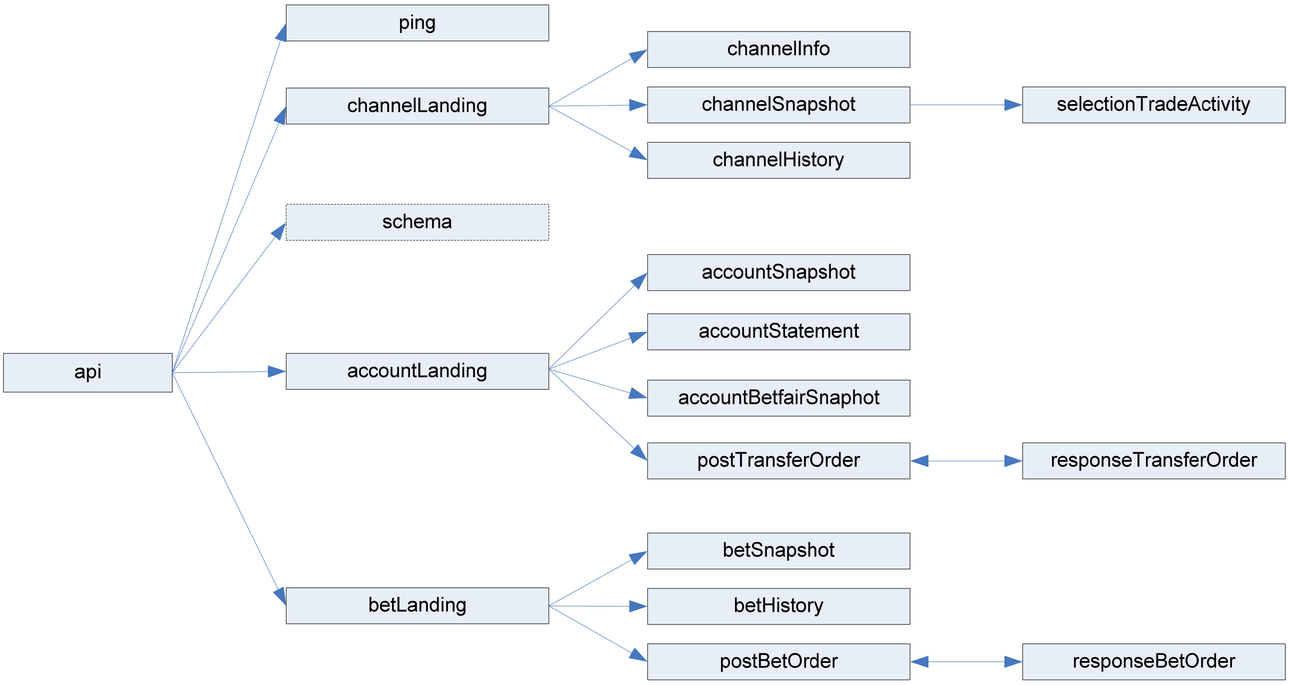


Figure 5 The Resource Hierarchy of the API (showing all the XML elements that you can request)

The three remaining URLs (in addition to the one for requesting the schema and the one for performing a “ping”) that are returned when you request the “api” root element enable you to:

- Access the games themselves,
- Manage your Betfair Exchange Games account, and
- Place and view bets.

These services are the subjects of the remaining chapters of this user guide.

Figure 5 shows the hierarchy of XML resources that comprise the Games API.

3.2 The Format of the API’s Responses to GET Requests

Figure 6 shows a sample response from the Games API to a GET request. It represents the root element of the Games API. This element is itself called “api”, and it is from this point in the Games API that you, or your client application, can access the API’s services (in other words, ping the API service, access current games, access your account, access your betting history and current bets, and place bets).

There are four things to note about the format of the sample response in Figure 6:

1. **This sample output is presented as HTTP and not HTTPS.**

   This is because the sample message is represented *after* decryption by some SSL library software.

2. **The sample output contains a series of “resource” elements, each one containing a URL.**

   It is by using URLs returned by the Games API (encapsulated as XML “resource” elements) that a client application navigates to the service or data required.

3. **The URLs listed in the sample output all include the `username` parameter with a sample value.**

   The Games API automatically includes the username parameter and value inside any URL that it returns to a client in response to an HTTP request that itself contained a valid username and password. This enables the client to use the URL without modification for a subsequent HTTP GET or POST request.

   If your client application requests services for which a username and password are not required and it does not specify a username and password, then the URLs contained in the responses that the Games API returns will themselves include no username parameter.

4. **The first 12 lines of sample output in Figure 6 represent standard HTTP header information**.

   All successful responses (including Betfair-specific error responses) sent by the Exchange Games API to a requesting client include HTTP header information. Note particularly that:

   - For all successful responses, the HTTP response code given in the header is 200.
   - All responses include a header field called “processTime”. This is a long integer value indicating the time taken (in milliseconds) for the Games API to process the request concerned. The “processTime” is measured from the time that a request is presented to the API after being decrypted, to the time that a response is presented by the API for encryption and transmission to the client. This piece of data can be useful for troubleshooting client applications: if there is an apparent delay in the network response times for a client application’s HTTP requests, you can check how much if any of the delay is accounted for by the time that the Games API is taking to process the client’s requests.

```
HTTP/1.1 200 OK

Date: Tue, 18 Apr 2006 09:47:02 GMT
Server: Betex HTTP Server
Cache-Control: no-store
Pragma: no-cache
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Content-Type: text/xml; charset=UTF-8
processTime: 3
Transfer-Encoding: chunked

29d
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
 <?xml-stylesheet type="text/xsl" href="/rest/apiStylesheet.xsl" ?>
<api xmlns="urn:betfair:games:api:v1">
<resource href="https://api.games.betfair.com/rest/v1/ping?username=sampleuser"
          responseType="ping" title="Ping the API server"/>
<resource href="https://api.games.betfair.com/rest/v1/channels?username=sampleuser"
          responseType="channels" title="View available channels"/>
<resource href="https://api.games.betfair.com/rest/v1/account?username=sampleuser"
          responseType="accountLanding" title="Manage your account"/>
<resource href="https://api.games.betfair.com/rest/v1/bet?username=sampleuser"
          responseType="betsLanding" title="Access your bets"/>
</api>
```

Figure *6* The Format of a Sample Response from the Games API to an Initial GET Request

In the remainder of the figures in this document that show sample output from the Games API, only the message body will be included (the XML content) unless there is specific reason to refer to details of the message header.

3.3 Requesting XML Listing the Services Available from the API (api)

The URL for the root element of the Games API is:

https://api.games.betfair.com/rest/v1?username=*username*

The call to this URL is an HTTP GET request. The name of the XML element returned is “api”.

```
<?xml version="1.0" ?>

<?xml-stylesheet type="text/xsl" href="/rest/apiStylesheet.xsl" ?>

<api xmlns="urn:betfair:games:api:v1">

    <resource href="https://api.games.betfair.com/rest/v1/schema?username=wnjayantha"
responseType="schema" title="View the schema"/>

    <resource href="https://api.games.betfair.com/rest/v1/ping?username=wnjayantha"
responseType="ping" title="Ping the API server"/>

    <resource href="https://api.games.betfair.com/rest/v1/channels?username=wnjayantha"
responseType="channelLanding" title="View available channels"/>

    <resource href="https://api.games.betfair.com/rest/v1/account?username=wnjayantha"
responseType="accountLanding" title="Manage your account"/>

    <resource href="https://api.games.betfair.com/rest/v1/bet?username=wnjayantha"
responseType="betLanding" title="Access your bets"/>

 </api>
```

Figure *7* The XML Returned When You Call the Root of the API

The XML returned by a request to this URL is shown in Figure 7. (The title of each URL is shown in bold.)

When you make an HTTP GET request for the root element of the Games API, you receive back a series of Betfair-defined XML elements, each of which is called a “resource”.

A “resource” element in any XML output from the Games API will always encapsulate a single URL, and this URL will provide access to another feature of the Games API.

The “resource” element contains no sub-elements but it has either three or four attributes:

- “href”, whose value is the URL that the “resource” exists to communicate;
- “title”, whose value is a text string describing what the URL is for;
- “responseType”, whose value is the name of the XML element that will be returned when you send an HTTP request to the URL,
- “postType” (only for “resource” elements that facilitate an HTTP POST request), whose value is the name of the XML element that your client application must send to the Games API when making the HTTP POST request associated with the URL.

  From Figure 7 you can see that, in the list of “resource” elements returned when you request the “api” root element, there are no URLs for posting data. However, later in this document you will find “resource” elements containing URLs that enable you to place bets and to transfer funds between Betfair accounts; in these cases the URLs provided are for sending POST requests.

When you request the “api” element of the Games API, the “resource” elements returned allow you to:

- Request the XML Schema Definition (XSD) document for the Games API
- Ping the Games API
- View the games available to bet on. (As you will see from Figure 7, the word “channels” is used in the title of the “resource” element that contains the URL for viewing available games. This concept is used in the sense of a TV channel: a games “channel” contains one current game at a time, just as a TV channel contains one programme at a time; shortly after one game finishes another starts.
- Manage your account
- Access your bets and place new ones

3.4 Requesting the XML Schema Definition (XSD) for the API (schema)

The URL that returns the XML schema for the API is:

`https://api.games.betfair.com/rest/v1/schema`

The call to this URL is an HTTP GET request. The name of the XML element returned is “schema”.

3.5 Requesting XML that Reports a “Ping” of the API (ping)

The URL that returns the operating status of the Betfair Exchange Games API is:

`https://api.games.betfair.com/rest/v1/ping`

The call to this URL is an HTTP GET request. The name of the XML element returned is “ping”.

When you send a GET request to this URL, the Games API returns its operating status; it also returns the results of a user credential test and if applicable a user status

The user credential test is still performed if you do not submit a username and password. However, the information returned in this case simply states “CREDENTIALS_NOT_SUBMITTED”.

No user status test is performed if you do not submit a username and password. Therefore in this case no “userStatus” element is included in the XML output.

These additional tests (for user credentials and status) mean that the “ping” element enables you to check there are no problems with your login data or account, as well as to check that the Games API is running and that your client can contact it successfully.

```
<ping xmlns="urn:betfair:games:api:v1">
<code>OK</code>
<credentialCheck>LOGIN_SUCCESS</credentialCheck>
<userStatus>ACTIVE</userStatus>
</ping>
```

Figure 8 Sample XML Reporting the Status of the Betfair Exchange Games API Server as OK

```
<ping>
<code>FAILED</code>
</ping>
```

Figure 9 *Sample XML Reporting the Status of the Betfair Exchange Games API Server as Not OK*

The “ping” element contains the following sub-elements:

- “code”, whose data content is one of the following strings: “OK”, “FAILED”, or “SITE_UPGRADE”. If it is “OK”, then the Games API is available and your client application can successfully contact it.

  If it is “FAILED”, the meaning is not exactly analogous to the meaning of a traditional IP ping failure. It means that the server has received and processed the HTTPS GET request, but it has been unable fully to complete the process of connecting the client to the Games API application.

  If it is “SITE_UPGRADE”, the Games API is temporarily unavailable while maintenance is in progress.

- “credentialCheck”, whose data content is one of the strings listed in Table 14 in (Appendix A).
- “userStatus”, whose data content is one of the text strings listed in Table 15 (Appendix A).

Finally, note that it is possible for a “ping” request to be successful but for the login to fail.

Figure 10 shows an example of a response in which the “ping” has been successful but the login has not completed because the user has not yet accepted the Betfair terms and conditions.

```
<ping xmlns="urn:betfair:games:api:v1">
   <code>OK</code>
   <credentialCheck>LOGIN_REQUIRE_TERMS_AND_CONDITIONS_ACCEPTANCE</credentialCheck>
   <userStatus>ACTIVE</userStatus>
</ping>
```

Figure *10* Sample Output When You Ping the Games API Successfully But the Login Does Not Complete

Chapter 4 Accessing the Exchange Games Using the API

This chapter tells you:

- How to use the Exchange Games API to request a list of the games currently available to bet on,
- How to use the API to retrieve details of a particular current or past game, and
- How to interpret the XML output that is returned for the different games.

The Betfair Exchange Games are played according to rules that will in some cases differ slightly from the rules with which you are familiar. Before you place any bets, therefore, we recommend that you read the rules adopted by the Betfair implementation of the games that you intend to bet on. These rules are explained in the chapters below which describe the XML for each game. They are also available from the Betfair Games website (`http://games.betfair.com`).

The XML elements that are used by the Exchange Games API to represent the different games are generic (they are the same for each game). However, they are used in subtly different ways for each game. The individual chapters (below) for each game explain how the elements are used to represent that game; this will help you design your client to process the XML correctly for each game.

4.1 Requesting XML Listing All the Games You Can Bet On (channelLanding)

The URL for requesting XML listing the games available to bet on is:

`https://api.games.betfair.com/rest/v1/channels`

The call to this URL is an HTTP GET request. The name of the XML element returned is “channelLanding”. Figure 11 shows some sample XML output provided by the Games API in response to a GET request to this URL.

The “channelLanding” element (as shown in Figure 11) contains an unbounded sequence of elements called “channel”. There is one for each available game, and there will always be a minimum of five: one each for Texas Hold’em Poker, Omaha Hi Poker, Blackjack, Baccarat, and Hi Lo. The sense in which the word “channel” is used in relation to games is analogous to the sense in which it is used in relation to TV programmes: a game takes place on a particular channel, just as a TV programme is shown on a particular channel; similarly when one game is over, another can - and will soon - start.

The “channel” element has three attributes and contains an unbounded sequence of “resource” elements. (For information about the “resource” element, see Section 3.3.) Its attributes are:

- “gameType”, whose value is a text string describing the type of game being played in the channel (examples are “POKER”, “BLACKJACK”, “BACCARAT”, “HILO”, “OMAHA_POKER”; the “gameType” is the same whether the channel is being used for the regular or the turbo version of a game),
- “id”, whose value is the identifying number of the “channel” (it will be an integer value);
- “name”, whose value is a text string giving the name of the particular version of the game whose generic name was given in the “gameType” attribute. Examples of the names of games are “Exchange Poker Texas Hold’em” and “Turbo Exchange Poker”, “Exchange Blackjack” and “Turbo Exchange Blackjack”, “Exchange Hi Lo”, and “Exchange Baccarat”. The word “Exchange” in these names acknowledges that each game is a unique implementation for the Betfair Games betting exchange.

  You can see from the sample “channelLanding” element shown in Figure 11 that for some Exchange games there are also “turbo” equivalents available. The rules and behaviour of the turbo versions of the games are exactly the same as those of the non-turbo versions. The only difference is that in the turbo versions the rounds of betting (referred to as betting windows) are 25% shorter. Therefore, if you play the turbo games you can bet on more games in a given space of time.

The “channel” element that is contained inside the “channelLanding” element is also *defined* inside the “channelLanding” element in the XML schema. This means that its definition is local to “channelLanding”. The reason for mentioning this here is that there are two other “channel” elements defined in the schema: one is local to “channelSnapshot” (see Section 4.2), and one is local to “channelHistory” (see Section 4.3). The function and structure of each are different and are determined by their different contexts.

However, all three elements called “channel” share the same three XML attributes listed above, including, most importantly, the “id” attribute (the number that identifies a channel with a particular game). The specific function of the “channel” inside the “channelLanding” element is to list URLs that enable you to:

- View a snapshot of a particular exchange game that is currently in progress; the URL for requesting this information is provided in a “resource” element whose “title” is “Channel Snapshot” (see Figure 11). For more information about the “channelSnapshot” resource, see Section 4.2.
- View details of all the games that have been played in the “channel” (in other words, all the games with the same number for the “id” attribute identifying the “channel” they were hosted by) over the last 90 days (see Figure 11). Games that are more than 90 days old are archived and will therefore no longer be available via the Exchange Games API. They are available from the following URL: http://stats.betfairgames.com.

```
<channelLanding xmlns="urn:betfair:games:api:v1">
 <channel gameType="POKER" id="1444074" name="Exchange Poker Texas Hold'em">
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444074/snapshot?username=sampleuser" responseType="channelSnapshot"
   title="Channel Snapshot"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444074/history?username=sampleuser" responseType="channelHistory"
   title="Channel history"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444074/info?username=sampleuser" responseType="channelInfo"
   title="Channel Definition"/>
 </channel>
 <channel gameType="POKER" id="1444080" name="Turbo Exchange Poker Texas Hold’em">
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444080/snapshot?username=sampleuser" responseType="channelSnapshot"
   title="Channel Snapshot"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444080/history?username=sampleuser" responseType="channelHistory"
   title="Channel history"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444080/info?username=sampleuser" responseType="channelInfo"
   title="Channel Definition"/>
 </channel>
 <channel gameType="BLACKJACK" id="1444077" name="Exchange BlackJack">
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444077/snapshot?username=sampleuser" responseType="channelSnapshot"
   title="Channel Snapshot"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444077/history?username=sampleuser" responseType="channelHistory"
   title="Channel history"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444077/info?username=sampleuser" responseType="channelInfo"
   title="Channel Definition"/>
 </channel>
 <channel gameType="BLACKJACK" id="1444082" name="Turbo Exchange BlackJack">
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444082/snapshot?username=sampleuser" responseType="channelSnapshot"
   title="Channel Snapshot"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444082/history?username=sampleuser" responseType="channelHistory"
   title="Channel history"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444082/info?username=sampleuser" responseType="channelInfo"
   title="Channel Definition"/>
 </channel>
 <channel gameType="BACCARAT" id="1444086" name="Exchange Baccarat">
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444086/snapshot?username=sampleuser" responseType="channelSnapshot"
   title="Channel Snapshot"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444086/history?username=sampleuser" responseType="channelHistory"
   title="Channel history"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444086/info?username=sampleuser" responseType="channelInfo"
   title="Channel Definition"/>
  <resource href=https://api.games.betfair.com/rest/v1/channels/1444086/snapshot?username=sampleuser&amp;selectionsType=MainBets
   responseType="channelSnapshot" title="Channel Snapshot - MainBets"/>
  <resource href=https://api.games.betfair.com/rest/v1/channels/1444086/snapshot?username=sampleuser&amp;selectionsType=SideBets
   responseType="channelSnapshot" title="Channel Snapshot - SideBets"/>
 </channel>
<channel gameType="HILO" id="1444093" name="Exchange Hi Lo">
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444093/snapshot?username=sampleuser" responseType="channelSnapshot"
   title="Channel Snapshot"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444093/history?username=sampleuser" responseType="channelHistory"
   title="Channel history"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444093/info?username=sampleuser" responseType="channelInfo"
   title="Channel Definition"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444093/snapshot?username=sampleuser&amp;selectionsType=CorrectPredictions"
   responseType="channelSnapshot" title="Channel Snapshot - CorrectPredictions"/>
  <resource href=https://api.games.betfair.com/rest/v1/channels/1444093/snapshot?username=sampleuser&amp;selectionsType=SideBets
   responseType="channelSnapshot" title="Channel Snapshot - SideBets"/>
 </channel>
<channel gameType="OMAHA_POKER" id="1444100" name="Exchange Omaha Hi">
  <resource href="https:// api.games.betfair.com /rest/v1/channels/1444100/snapshot?username=sampleuser " responseType="channelSnapshot"
   title="Channel Snapshot"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444100/history?username=sampleuser " responseType="channelHistory"
   title="Channel history"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444100/info?username=sampleuser " responseType="channelInfo"
   title="Channel Definition"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444100/snapshot?username=sampleuser&selectionsType=MainBets"
   responseType="channelSnapshot" title="Channel Snapshot - MainBets"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444100/snapshot?username=sampleuser&selectionsType=SideBets"
   responseType="channelSnapshot" title="Channel Snapshot - SideBets"/>
</channel>
</channelLanding>
```

Figure 11 Sample XML Response to a Request for the “channelLanding” Element Listing the Games

4.2 Requesting XML Describing the Current Game in a Channel (channelSnapshot)

For each game currently in progress on the Games API, you can request XML that reports the state of the game. To make your client application track the progress of the game, you need to include the URL for this request in a function that repeats the request every three seconds or so during a round of the game. The duration of each round in a game is presented in the “channelInfo” element, where it is called the “playingWindow”; see Section 4.4).

The URL for requesting XML describing the current state of the game in a particular channel is as follows:

`https://api.games.betfair.com/rest/v1/channels/channelId/snapshot?username=sampleuser`

where *channelId* is an integer value (automatically included in the URLs returned for each channel in the “channelLanding” page; see Figure 11 above).

The call to this URL is an HTTP GET request. The name of the XML element returned is “channelSnapshot”.

Figure 12 (on page 25) shows a sample “channelSnapshot” for the turbo version of a game of Poker (the turbo versions of games are the same as the non-turbo versions, except that the betting window times are 25% faster in the turbo versions).

The “channelSnapshot” element contains an element called “channel”, which in turn contains an element called “game”.

The “game” element has an “id” attribute (a unique number identifying the game independently from the betting markets associated with it) and it contains a “round” element (reporting the number of the current betting round), a “bettingWindowTime” (reporting the length of the betting round), and a “bettingWindowPercentageComplete” element (reporting how much of the betting round, as a percentage, has elapsed). For information about how to request a “channelSnapshot” that returns only this information about the betting round, see Section 4.2.1.4.

Most importantly, the “game” element also contains the “gameData” element and the “markets” element.

The “gameData” element contains the details of the current game in the form of a list of “object” elements, each “object” representing an automated player, dealer, or other significant participant in, or component of, the automated game; in the case of Exchange Hi Lo, there is only a single “object” and it represents the entire game.

For details of the different ways in which the “object” elements are implemented for Poker, Blackjack, Baccarat, and Hi Lo, see the sections in Chapter 5, Chapter 6, Chapter 7, Chapter 8, and Chapter 9, respectively, that describe the XML for the games.

The “markets” element in a “channelSnapshot” (see Figure 12 on page 25) contains the details of the game’s betting market or, if applicable, its markets (some of the games have two markets).

The market information is held inside the “market” element. Each “market” element has two attributes: an “id” attribute which uniquely identifies the current instance of the market and a “nextId” attribute which identifies the next instance of that market for the next game in the channel. This allows a client application to prepare “postBetOrder” requests for the next game before it has started.

The “market” element contains (as well as a list of betting “selection” elements) a “status” element, a “commissionRate” element, and a “marketType” element.

The possible values of a betting market’s “status” element are listed in Table 3 (on page 27).

The possible values of the “marketType” element are listed in Table 4 (on page 27).

Each betting selection whose details are returned in the “market” element of a “channelSnapshot” includes a facility for requesting comprehensive market data relating to the selection. For information about how to use this facility, see Section 4.2.2.

If you have any active bets on the game whose “channelSnapshot” you have requested, then each betting selection also tells you how much money you stand to win or lose if that selection wins (see 4.2.3).

```
<channelSnapshot xmlns="urn:betfair:games:api:v1">
  <channel gameType="POKER" id="1444080" name="Turbo Exchange Poker">
  <status>RUNNING</status>
    <game id="625414">
       <round>3</round>
       <bettingWindowTime>30</bettingWindowTime>
       <bettingWindowPercentageComplete>0</bettingWindowPercentageComplete>
       <gameData>
         <object name="Hand 1">
           <description>One Pair, Aces, Jack High</description>
           <status>IN_PLAY</status>
           <property name="Card 1" value="39"/>
           <property name="Card 2" value="27"/>
         </object>
         <object name="Hand 2">
            <description>One Pair, Jacks, Ace High, with 7 Kicker</description>
            <status>IN_PLAY</status>
            <property name="Card 1" value="32"/>
            <property name="Card 2" value="49"/>
         </object>
         ...
         <object name="Community Cards">
            <description/>
            <status>N/A</status>
            <property name="Card 1" value="13"/>
            <property name="Card 2" value="3"/>
            <property name="Card 3" value="23"/>
         ...
         </object>
       </gameData>
       <markets currency="GBP">
          <market id="2563238" nextId="2563261">
            <status>ACTIVE</status>
            <commissionRate>2.5</commissionRate>
            <marketType>WIN_ONLY</marketType>
            <selections type="MainBets">
              <selection id="658439">
                <name>Hand 1</name>
                <resource href="https://api.games.betfair.com/rest/v1/selection/tradeActivity
                 ?username=sampleuser&marketId=2563238&selectionId=658439"
                 responseType="selectionTradeActivity" title="Trade Activity"/>
                <status>IN_PLAY</status>
                <amountMatched>0.0</amountMatched>
                <bestAvailableToBackPrices>
                     <price amountUnmatched="160.0">10.9</price>
                     <price amountUnmatched="120.0">11</price>
                     <price amountUnmatched="200.0">11.1</price>
                </bestAvailableToBackPrices>
                <bestAvailableToLayPrices>
                     <price amountUnmatched="200.0">11.4</price>
                     <price amountUnmatched="180.0">11.5</price>
                     <price amountUnmatched="190.0">11.6</price>
                </bestAvailableToLayPrices>
              </selection>
               ...
             </selections>
          </market>
       </markets>
    </game>
  </channel>
</channelSnapshot>
```

Figure 12 Sample “channelSnapshot” Describing a Current Game of Turbo Exchange Poker [4]

[4] The dots in the sample XML output indicate the omission of repetitive data.

Table 2 (below) tells you the possible values for a channel's "status" element (in a "channelSnapshot").

Table 2 The Possible Values of the Channel's "status" Element (in a "channelSnapshot")

Channel Statuses: **strings of type "channelStatusEnum"**	**The meaning of the String**
RUNNING	The channel is operating normally.
ABANDONED	The Exchange Games engine has abandoned the channel. If a channel has this status, it is either because the Exchange Games engine has encountered an internal error that it cannot recover from, or it is because the Betfair System Administrator has sent an ABANDON instruction to the engine requiring it to abandon the channel immediately.
STOPPED	The Exchange Games engine is no longer running the channel. If you see this status for a channel, it means that the Betfair System Administrator has sent a STOP instruction to the Exchange Games engine, and that the engine has followed the instruction and stopped running the channel (after allowing the current game to complete and settle).
PENDING_RUNNING	This status means that the Betfair System Administrator has sent an instruction to the Exchange Games engine to start running the channel again, and that the engine is following the instruction.
PENDING_ABANDON	Although this status is not impossible, it is unlikely that you will see it. If you do, it means that the XML containing the "channelSnapshot" was sent to you by the Games Exchange API just a few milliseconds after the Exchange Games engine received an ABANDON instruction from the Betfair System Administrator to abandon the channel immediately.
PENDING_STOP	The Betfair System Administrator has sent a STOP instruction to the Exchange Games engine, and the engine will therefore stop running the channel when the current game completes and all current bets on the game are settled.

Table 3 The Possible Values of a Betting Market's "status" Element

Market Statuses: **strings of type "marketStatusEnum"**	**The meaning of the String**
ACTIVE	The betting market is running normally and you can place bets in it.
SUSPENDED_GAME_PLAYING	The market is running normally, but betting has been suspended while a particular phase of the game completes. (You will currently only see this status during the second - and possibly subsequent – "playingWindow" phases of Exchange Blackjack.)
SUSPENDED_GAME_SETTLING	Betting has been suspended because the bets are now being settled.
SUSPENDED_GAME_ROUND_OVER	Betting has been suspended because a round of cards has just ended.
SETTLED	The game has finished and all bets have been settled. This is the final status of a game, and it is this status that games (that finished normally) will have when viewed in the "channelHistory" element.
VOID	The game has been declared VOID, and all bets made on it have been made void.

Table 4 lists the different types of betting market associated with each of the Exchange Games.

Table 4 The Possible Types of Market for a Game (the Values for the "marketType" Element)

The Types of Betting Market (strings of type "marketTypeEnum")	**The Game and Betting Markets that these Types Apply to**
WIN_ONLY	The "MainBets" market for Exchange Poker is of this "marketType". For a description of it see Section 5.3.2. Also, both the "MainBets" and the "SideBets" markets for Exchange Omaha Hi Poker are of this "marketType" (see Section 6.3.2).
ANY_NUMBER_OF_WINNERS	The "MainBets" market for Exchange Blackjack and the "SideBets" markets for Exchange Baccarat and Exchange Hi Lo are of this "marketType". For a description of this "marketType" in its application to: • Blackjack, see Section 7.3.2 • Baccarat, see Section 8.3.3 • Hi Lo, see Section 9.3.3
SINGLE_WINNER_OR_TIE	The "MainBets" betting market for Exchange Baccarat is of this "marketType". For a description of it, see Section 8.3.2.

VARIABLE_HANDICAP	The “Correct Predictions” market for Exchange Hi Lo is of this “marketType”. For a description of it, see Section 9.3.2.

For details of the ways in which the betting “selection” elements are implemented for Poker, Blackjack, Baccarat, and Hi Lo, see the sections in Chapter 5, Chapter 6, Chapter 8, and Chapter 9 respectively, that describe the XML for those games.

4.2.1 Controlling How Much Data is Returned in a “channelSnapshot” (channelSnapshot)

To each URL listed in the “channelLanding” element (see Figure 11) for requesting a “channelSnapshot”, you can add other parameters manually. The available parameters allow you to suppress those elements of a complete “channelSnapshot” that are not immediately required. In other words, they enable you to minimise the quantity of redundant information that your client application needs to process.

For example, if you only want to know how much time is left in the betting window, you do not need to receive the information about the betting markets or the progress of the game that is normally returned in a “channelSnapshot”.

Similarly, if you only want to check the available back and lay prices for a betting selection, then you do not need the information about the progress of the game or about the betting round.

And, if you only want to check the progress of the game then you don’t need to receive the information that represents the state of the betting market or markets.

Finally, if you only want to receive information about a particular betting market, then information about the other betting market (if there is another one) and about the progress of the game and about the betting round is all redundant.

This section tells you how to use the different parameters that are available for the “channelSnapshot” URL to make sure you only receive the information you need about the games you are betting on.

To add one of the parameters, just put an ampersand character (&) at the end of the “channelSnapshot” URL you want to use, and then specify the parameter as described in the sections below.

To add another parameter after this, just add another ampersand character and specify the next parameter, and so on.

The following example is of a URL for requesting a “channelSnapshot” with the “gameData” element suppressed (as described in Section 4.2.1.1):

`https://api.games.betfair.com/rest/v1/channels/`*`channelId`*`/snapshot?username=sampleuser&game=false`

where *channelId* is an integer value (automatically included in the URLs returned for each channel in the “channelLanding” page; see Figure 11 above).

4.2.1.1 Suppressing all the Game Information (channelSnapshot)

To each of the URLs for requesting the “channelSnapshot” element, you can add a parameter manually to suppress the “gameData” element. Use this parameter if you are only interested in receiving data concerning the betting selections and their current prices in the market and you do not need to review the progress of the game.

The parameter (including the necessary ampersand character) for suppressing the “gameData” element is:

```
&game=false
```

4.2.1.2 Suppressing all the Market Information (channelSnapshot)

To each of the URLs for requesting the “channelSnapshot” element, you can add a parameter manually to suppress the “markets” element. Use this parameter if you are only interested in receiving data concerning the progress of the game and you do not need to review the betting selections and their current prices in the market. The parameter (including the necessary ampersand character) for suppressing the “markets” element is:

```
&market=false
```

4.2.1.3 Suppressing all the Betting Round Information (channelSnapshot)

To each of the URLs for requesting the “channelSnapshot” element, you can add a parameter manually to suppress the three XML elements in a “channelSnapshot” that give details of the current betting round. These elements are “round”, “bettingWindowTime”, and “bettingWindowPercentageComplete”. Use this parameter if you do not need to receive details of the current betting round. The parameter (including the necessary ampersand character) for suppressing these elements is:

```
&timing=false
```

4.2.1.4 Requesting the Betting Round Information Only (channelSnapshot)

You can use the parameters available for the “channelSnapshot” URL to request information about the current betting round only (in other words, suppressing the “gameData” and “markets” elements).

To do this, you need to include both the `game=false` parameter and the `market=false` parameter in the URL for the “channelSnapshot”. A sample URL for this is as follows:

https://api.games.betfair.com/rest/v1/channels/*channelId*/snapshot?username=sampleuser&market=false&game=false

where *channelId* is an integer value (automatically included in the URLs returned for each channel in the “channelLanding” page; see Figure 11 above).

```
  <channelSnapshot>
           <channel gameType="OMAHA_POKER" id="1444100" name="Exchange Omaha Hi">
                 <status>RUNNING</status>
                 <game id="157178">
                       <round>4</round>
                       <bettingWindowTime>25</bettingWindowTime>
                       <bettingWindowPercentageComplete>40</bettingWindowPercentageComplete>
                 </game>
           </channel>
     </channelSnapshot>
```

Figure 13 A “channelSnapshot” Showing Only Betting Round Data (all other data is suppressed)

4.2.1.5 Requesting the Game Information Only (channelSnapshot)

You can use the parameters available for the “channelSnapshot” URL to request information about only the progress of the game (in other words, suppressing the “round”, “bettingWindowTime”, and “bettingWindowPercentageComplete” elements, and suppressing also the “markets” element).

To do this, you need to include both the `timing=false` parameter and the `market=false` parameter in the URL for the “channelSnapshot”. A sample URL for this is as follows:

https://api.games.betfair.com/rest/v1/channels/*channelId*/snapshot?username=sampleuser&market=false&timing=false

where *channelId* is an integer value (automatically included in the URLs returned for each channel in the “channelLanding” page; see Figure 11 above).

```
<channelSnapshot>
         <channel gameType="OMAHA_POKER" id="1444100" name="Exchange Omaha Hi">
               <status>RUNNING</status>
               <game id="157184">
                     <gameData>
                           <object name="Hand 1">
                                 <description/>
                                 <status>IN_PLAY</status>
                                 <property name="Card 1" value="NOT AVAILABLE"/>
                                 <property name="Card 2" value="NOT AVAILABLE"/>
                                 <property name="Card 3" value="NOT AVAILABLE"/>
                                 <property name="Card 4" value="NOT AVAILABLE"/>
                           </object>
                           <object name="Hand 2">
                                 <description/>
                                 <status>IN_PLAY</status>
                                 <property name="Card 1" value="NOT AVAILABLE"/>
                                 <property name="Card 2" value="NOT AVAILABLE"/>
                                 <property name="Card 3" value="NOT AVAILABLE"/>
                                 <property name="Card 4" value="NOT AVAILABLE"/>
                           </object>
                           <object name="Hand 3">
                                 <description/>
                                 <status>IN_PLAY</status>
                                 <property name="Card 1" value="NOT AVAILABLE"/>
                                 <property name="Card 2" value="NOT AVAILABLE"/>
                                 <property name="Card 3" value="NOT AVAILABLE"/>
                                 <property name="Card 4" value="NOT AVAILABLE"/>
                           </object>
                           <object name="Hand 4">
                                 <description/>
                                 <status>IN_PLAY</status>
                                 <property name="Card 1" value="NOT AVAILABLE"/>
                                 <property name="Card 2" value="NOT AVAILABLE"/>
                                 <property name="Card 3" value="NOT AVAILABLE"/>
                                 <property name="Card 4" value="NOT AVAILABLE"/>
                           </object>
                           <object name="Community Cards">
                                 <description/>
                                 <status>N/A</status>
                                 <property name="Card 1" value="NOT AVAILABLE"/>
                                 <property name="Card 2" value="NOT AVAILABLE"/>
                                 <property name="Card 3" value="NOT AVAILABLE"/>
                                 <property name="Card 4" value="NOT AVAILABLE"/>
                                 <property name="Card 5" value="NOT AVAILABLE"/>
                           </object>
                     </gameData>
               </game>
         </channel>
   </channelSnapshot>
```

Figure 14 A "channelSnapshot" Showing Game Data (market and betting time data are suppressed)

4.2.1.6 Requesting the Market Information Only (channelSnapshot)

You can use the parameters available for the "channelSnapshot" URL to request information about the current betting market or markets only (in other words, suppressing the "gameData" element and the elements that provide betting round information).

To do this, you need to include both the `timing=false` parameter and the `game=false` parameter in the URL for the "channelSnapshot". A sample URL for this is as follows:

`https://api.games.betfair.com/rest/v1/channels/`*`channelId`*`/snapshot?username=sampleuser&timing=false&game=false`

where *channelId* is an integer value (automatically included in the URLs returned for each channel in the "channelLanding" page; see Figure 11 above).

```
  <channelSnapshot>
     <channel gameType="POKER" id="1444074" name="Exchange Poker Texas Hold'em">
       <status>RUNNING</status>
         <game id="159310">
         <markets currency="GBP">
           <market id="5164633" nextId="5164660">
             <status>ACTIVE</status>
             <commissionRate>2.5</commissionRate>
             <marketType>WIN_ONLY</marketType>
             <selections type="MainBets">
               <selection id="658439">
                 <name>Hand 1</name>
                 <resource href="https://api.games.betfair.com/rest/v1/selection/tradeActivity
                  ?username=sampleuser&marketId=5164633&selectionId=658439"
                  responseType="selectionTradeActivity" title="Trade Activity"/>
                 <status>IN_PLAY</status>
                 <amountMatched>0.0</amountMatched>
                 <bestAvailableToBackPrices/>
                 <bestAvailableToLayPrices/>
               </selection>
               <selection id="658440">
                 <name>Hand 2</name>
                 <resource href="https://api.games.betfair.com/rest/v1/selection/tradeActivity
                  ?username=hobbs&marketId=5164633&selectionId=658440"
                  responseType="selectionTradeActivity" title="Trade Activity"/>
                 <status>IN_PLAY</status>
                 <amountMatched>0.0</amountMatched>
                 <bestAvailableToBackPrices/>
                 <bestAvailableToLayPrices/>
               </selection>
               <selection id="658441">
                 <name>Hand 3</name>
                 <resource href=https://api.games.betfair.com/rest/v1/selection/tradeActivity?
                  username=sampleuser&marketId=5164633&selectionId=658441
                  responseType="selectionTradeActivity" title="Trade Activity"/>
                  <status>IN_PLAY</status>
                  <amountMatched>0.0</amountMatched>
                  <bestAvailableToBackPrices/>
                  <bestAvailableToLayPrices/>
               </selection>
               <selection id="658442">
                 <name>Hand 4</name>
                 <resource href="https://api.games.betfair.com/rest/v1/selection/tradeActivity?
                  username=hobbs&marketId=5164633&selectionId=658442"
                  responseType="selectionTradeActivity" title="Trade Activity"/>
                 <status>IN_PLAY</status>
                 <amountMatched>0.0</amountMatched>
                 <bestAvailableToBackPrices/>
                 <bestAvailableToLayPrices/>
               </selection>
             </selections>
           </market>
         </markets>
       </game>
    </channel>
  </channelSnapshot>
```

Figure 15 A “channelSnapshot” Showing Only Market Data (game and betting time data are suppressed)

4.2.1.7 Requesting Data about a Specific Market (channelSnapshot)

For games that have two betting markets associated with them, a Mainbets and a Sidebets market (or, in the case of Hi Lo, a Correct Predictions and a Sidebets market), there are URLs provided in the "channelLanding" page (see Figure 11) that enable you to request the "channelSnapshot" you require with one of the markets suppressed. These URLs contain the `selectionsType=` parameter.

The value of this parameter is the name of the betting market. So, for Exchange Baccarat, for example, it is either `MainBets` or `SideBets`, and for Exchange Hi Lo it is either `CorrectPredictions` or `SideBets`.

The parameter (including the necessary ampersand character) for specifying the betting market you want to see (and suppressing the other) is:

```
&selectionsType=betting market
```

where *`betting market`* is, for example, `MainBets`, `SideBets`, or `CorrectPredictions`. (Note that the betting market whose name you specify must be the correct name of an available market for the game.)

A sample URL for requesting an Exchange Baccarat "channelSnapshot" that contains market data for only the MainBets market is as follows:

https://api.games.betfair.com/rest/v1/channels/1444086/snapshot?username=sampleuser&selectionsType=MainBets

NOTE: Although the "channelSnapshot" returned by the above URL suppresses the data for the SideBets market, in all other respects it returns a complete "channelSnapshot" element (in other words, one that includes all the game and betting round information as well as the market data you specify).

For information about how to request a "channelSnapshot" that contains no data except that for the market you are interested in, see Section 4.2.1.8.

4.2.1.8 Requesting Data about a Specific Market and *Nothing* Else (channelSnapshot)

You can use the parameters available for the "channelSnapshot" URL to request information about a particular betting market and nothing else.

To do this, you need to include the `timing=false` parameter (to suppress the betting round data), the `game=false` parameter (to suppress the data about the automated players), and the `selectionsType=` parameter (to specify the market you want to examine). A sample URL for this is as follows:

https://api.games.betfair.com/rest/v1/channels/*channelId*/snapshot?username=sampleuser&game=false&timing=false&selectionsType=SideBets

where *channelId* is an integer value (automatically included in the URLs returned for each channel in the "channelLanding" page; see Figure 11 above).

4.2.2 Requesting XML Listing *all* Prices (odds) for a Selection (selectionTradeActivity)

In each "selection" element contained inside the "market" element (or "market" elements) in a "channelSnapshot" (see Figure 12), there is a "resource" element containing a URL.

You can use this URL to request a list of all the prices (odds) that have been offered by other bettors to lay or back the selection since the betting market opened.

The "market" element inside a "channelSnapshot" (see Figure 12) also contains elements called "bestAvailableToBackPrices" and "bestAvailableToLayPrices". As their names suggest, these elements provide the *best* prices available in the market for the selection (in fact, they provide the best three prices).

However, you might want to see what other prices are available. This information, for example, would be useful to you if you wanted to place a bet for a sum that was larger than the sums available to be matched at the best three prices.

In this case, if you simply placed a bet for the whole amount of your desired stake to back a selection at the best available price, then only that portion of your stake that was equal to the total amount available to be matched at that price would get matched. The balance would be added to the sum available at the best price to lay, but it might not get matched or it might not all get matched.

The alternative, then, is to analyse the distribution of funds at *all* the available prices and then choose which odds to place your bet at. It may well be that there are only a small amount of funds available at the best prices but that the “serious” money is available at slightly less favourable odds.

You can also use this facility to get an idea of how much of a large bet placed at prices less favourable than the best prices is likely to be matched at better prices, as a result of the exchange server’s “best execution” policy. (By this policy, the exchange server matches bets at the requested odds or at better odds if funds are available for them to be matched with at better odds than requested).

The “selectionTradeActivity” element (see Figure 16) contains the “selectionId”, the “marketId”, a figure for the total “volume” of matched and settled bets on the selection, a figure for the “lastPriceMatched”, and a series of price statistics in an XML array called “priceStats”.

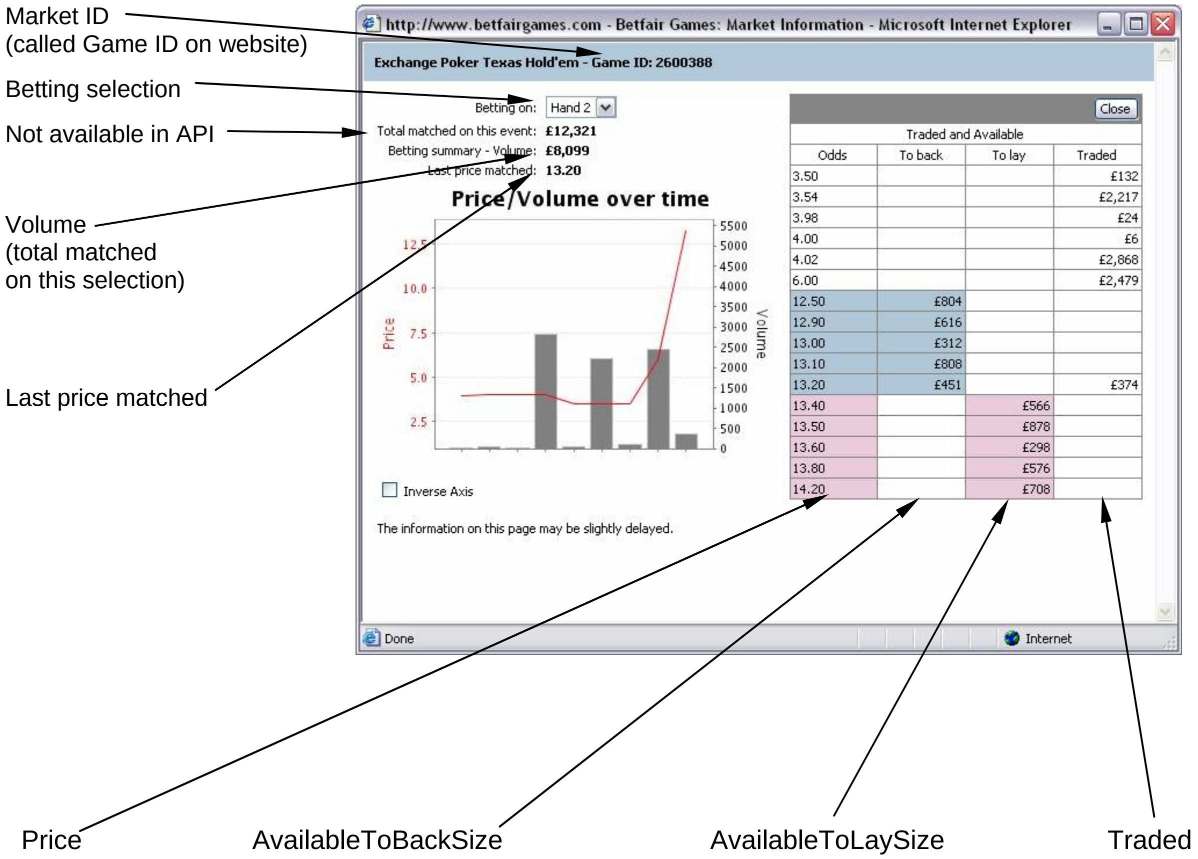
```
<selectionTradeActivity currency="GBP"
 xmlns="urn:betfair:games:api:v1">
    <selectionId>658439</selectionId>
    <marketId>2600388</marketId>
    <volume>8099</volume>
    <lastPriceMatched>13.20</lastPriceMatched>
    <priceStats>
     <priceStat>
       <price>3.50</price>
       <availableToBackSize>0.0</availableToBackSize>
       <availableToLaySize></availableToLaySize>
       <traded>132</traded>
     </priceStat>
      ...
     <priceStat>
       <price> 12.50</price>
       <availableToBackSize>804</availableToBackSize>
       <availableToLaySize></availableToLaySize>
       <traded></traded>
     <priceStat>

       ...
      <priceStat>
       <price> 13.40</price>
       <availableToBackSize>0.0</availableToBackSize>
       <availableToLaySize>566.0</availableToLaySize>
       <traded>374</traded>
     <priceStat>
       ...
    </priceStats>
</selectionTradeActivity>
```

Figure 16 The "selectionTradeActivity" Element with One Past Poker Game Shown[5]

[5] The dots in the sample XML output indicate the omission of repetitive data.

This array contains an unlimited number of “priceStat” elements, each of which in turn contains four sub-elements: “price”, “availableToBackSize”, “availableToLaySize”, and “traded”. These are equivalent respectively to the four columns in the Betfairgames website’s representation of market data for each selection. (To see this information for a selection on the website, just click on the selection you are interested in.) The word “traded” means the same as “matched”.

In Figure 17 the first six numbers in the “Traded” column are the sums matched at prices (odds) that are no longer realistic. They are from earlier rounds of betting and the game and the betting market have moved on. Any outstanding bets at those prices will have been lapsed and there are therefore now no funds available at them to be matched by anybody wanting to back or lay the selection. If anyone attempted to back the selection at a price of 3.5, that person’s bet would in fact be matched at the best available price of 13.2. At that price £374 has already been traded and there is still £451 available to be matched by back bets.



http://www.betfairgames.com - Betfair Games: Market Information - Microsoft Internet Explorer

Exchange Poker Texas Hold'em - Game ID: 2600388

Betting on: Hand 2

Total matched on this event: **£12,321**

Betting summary - Volume: **£8,099**

Last price matched: **13.20**

Traded and Available			
Odds	To back	To lay	Traded
3.50			£132
3.54			£2,217
3.98			£24
4.00			£6
4.02			£2,868
6.00			£2,479
12.50	£804		
12.90	£616		
13.00	£312		
13.10	£808		
13.20	£451		£374
13.40		£566	
13.50		£878	
13.60		£298	
13.80		£576	
14.20		£708	

□ Inverse Axis

The information on this page may be slightly delayed.

Figure *17* How “selectionTradeActivity” Data Appears for a Selection on the Betfair Games Website[6]

[6] There is no equivalent of the graph in the XML “selectionTradeActivity” element. It is the information in the table (plus the volume traded and the last price matched) that is paralleled in the Games API.

4.2.3 Requesting XML Listing Your Potential Profit and/or Loss (channelSnapshot)

This section describes how your potential profit and/or loss are expressed in the “channelSnapshot” page if you have logged in as a named user and have active bets matched[7] in the market for the game you are viewing. The XML you see will vary slightly depending on the type of betting market concerned. This section describes it for the different markets separately.

4.2.3.1 Profit OR Loss in a WIN_ONLY Betting Market (profitLoss)

If you have any bets placed in a WIN_ONLY betting market, for example, in the “MainBets” market for Poker, then the “channelSnapshot” element tells you what your potential profit or loss will be for each betting selection if that selection wins. The information is contained in the “profitLoss” element, which is returned inside every “selection” in the “market” element of a “channelSnapshot” (see Figure 18).

```
<market id="2563238" nextId="2563265">
            <status>ACTIVE</status>
            <commissionRate>2.5</commissionRate>
            <marketType>WIN_ONLY</marketType>
            <selections type="MainBets">
              <selection id="658439">
                <name>Hand 1</name>
                <resource href="https://api.games.betfair.com/rest/v1/selection/tradeActivity
                 ?username=sampleuser&marketId=2563238&selectionId=658439"
                 responseType="selectionTradeActivity" title="Trade Activity"/>
                <status>IN_PLAY</status>
                <amountMatched>0.0</amountMatched>
                <profitLoss>101.0</profitLoss>
                <bestAvailableToBackPrices>
                     <price amountUnmatched="160.0">10.9</price>
                     <price amountUnmatched="120.0">11</price>
                     <price amountUnmatched="200.0">11.1</price>
                </bestAvailableToBackPrices>
                <bestAvailableToLayPrices>
                     <price amountUnmatched="200.0">11.4</price>
                     <price amountUnmatched="180.0">11.5</price>
                     <price amountUnmatched="190.0">11.6</price>
                </bestAvailableToLayPrices>
              </selection>
              <selection id="658440">
                <name>Hand 2</name>
                <resource href="https://api.games.betfair.com/rest/v1/selection/tradeActivity
                 ?username=sampleuser&marketId=2563238&selectionId=658440"
                 responseType="selectionTradeActivity" title="Trade Activity"/>
                <status>IN_PLAY</status>
                <amountMatched>0.0</amountMatched>
                <profitLoss>-10.0</profitLoss>
                <bestAvailableToBackPrices>
                     <price amountUnmatched="160.0">10.9</price>
                     <price amountUnmatched="120.0">11</price>
                     <price amountUnmatched="200.0">11.1</price>
                </bestAvailableToBackPrices>
                <bestAvailableToLayPrices>
                     <price amountUnmatched="200.0">11.4</price>
                     <price amountUnmatched="180.0">11.5</price>
                     <price amountUnmatched="190.0">11.6</price>
                </bestAvailableToLayPrices>
                   ...
             </selections>
</market>
```

Figure 18 Excerpt from a “channelSnapshot”, Showing “profitLoss” for the “MainBets” Market for Poker

[7] The “profitLoss” element only appears when you have had bets matched in the relevant market. It does not appear when bets have been placed but not yet matched.

Figure 18 shows the potential profit on Hand 1 in a game of Poker for a user who has backed that hand with £10 at odds of 11.1. The potential profit is £101.0. For the next “selection”, Hand 2, the example shows the user’s potential loss of -£10.0 if that hand wins. The element that contains the potential profit in the “selection” data for Hand 1 and the element that contains the potential loss in the “selection” data for Hand 2 are the same: the “profitLoss” element is used (and if the value represents a loss it is prefixed by a minus sign).

Profit and loss are always represented in this way in a WIN_ONLY market.

4.2.3.2 Profit AND Loss in an ANY_NUMBER_OF_WINNERS Market (profitAndLoss)

For markets whose type is ANY_NUMBER_OF_WINNERS, for example, the “MainBets” market for Exchange Blackjack, and the “SideBets” markets for Exchange Baccarat and Exchange Hi Lo, the API returns a “profitAndLoss” element.

This is not the same as the “profitLoss” element described in the previous section, first, because the “profitAndLoss” element only appears for those selections you have bet on, and second, because the “profitAndLoss” element itself contains two sub-elements: “selectionWin” and “selectionLose”.

The “selectionWin” element tells you how much profit you will make if the selection wins. The “selectionLose” element tells you how much loss you will make if the selection loses.

```
<market id="2563079" nextId="2563106">
    <status>SUSPENDED_GAME_ROUND_OVER</status>
    <commissionRate>2.5</commissionRate>
    <marketType>ANY_NUMBER_OF_WINNERS</marketType>
    <selections type="SideBets">
     <selection id="1032040">
        <name>Pocket Pair In Any Hand</name>
        <resource href="https://api.betfair.com/rest/v1/selection/tradeActivity
          ?username=sampleuser&marketId=2563079&selectionId=1032040"
          responseType="selectionTradeActivity" title="Trade Activity"/>
        <status>LOSER</status>
        <amountMatched>0.0</amountMatched>
        <bestAvailableToBackPrices/>
        <bestAvailableToLayPrices/>
      </selection>
      <selection id="1032041">
        <name>Highest Hand Has Nine</name>
        <resource href="https://api.betfair.com/rest/v1/selection/tradeActivity
          ?username=sampleuser&marketId=2563079&selectionId=1032041"
          responseType="selectionTradeActivity" title="Trade Activity"/>
        <status>IN_PLAY</status>
        <amountMatched>0.0</amountMatched>
        <profitAndLoss>
           <selectionWin>10.0</selectionWin>
           <selectionLose>-5.0</selectionLose>
       </profitAndLoss>
       <bestAvailableToBackPrices/>
       <bestAvailableToLayPrices/>
       </selection>
     <selection id="1032042">
       <name>Highest Hand Is Odd</name>
       <resource href="https://api.betfair.com/rest/v1/selection/tradeActivity
         ?username=sampleuser&marketId=2563079&selectionId=1032042"
         responseType="selectionTradeActivity" title="Trade Activity"/>
       <status>IN_PLAY</status>
       <amountMatched>0.0</amountMatched>
       <bestAvailableToBackPrices/>
       <bestAvailableToLayPrices/>
     </selection>
     <selection id="1032043">
       <name>Natural Win</name>
        <resource href="https://api.betfair.com/rest/v1/selection/tradeActivity
         ?username=sampleuser&marketId=2563079&selectionId=1032043"
         responseType="selectionTradeActivity" title="Trade Activity"/>
       <status>LOSER</status>
       <amountMatched>0.0</amountMatched>
       <bestAvailableToBackPrices/>
       <bestAvailableToLayPrices/>
     </selection>
     <selection id="1032044">
       <name>Natural Tie</name>
       <resource href="https://api.betfair.com/rest/v1/selection/tradeActivity
         ?username=sampleuser&marketId=2563079&selectionId=1032044"
         responseType="selectionTradeActivity" title="Trade Activity"/>
       <status>LOSER</status>
       <amountMatched>0.0</amountMatched>
       <bestAvailableToBackPrices/>
       <bestAvailableToLayPrices/>
     </selection>
   </selections>
 </market>
```

Figure 19 Excerpt from a “channelSnapshot”, Showing “profitAndLoss” for the Baccarat “SideBets” Market

4.2.3.3 Profit OR Loss in a SINGLE_WINNER_OR_TIE Market (profitLoss)

If you have any bets placed in a SINGLE_WINNER_OR_TIE betting market, for example, in the “MainBets” market for Baccarat, then the “channelSnapshot” element tells you what your potential profit or loss will be for each betting selection if that selection wins. The information is contained in the “profitLoss” element, which is returned inside every “selection” in the “market” element of a “channelSnapshot” (see Figure 20), irrespective of whether you have a bet placed on the selection. The behaviour of the “profitLoss” element in a SINGLE_WINNER_OR_TIE market is the same as for a WIN_ONLY market.

```
<market id="3119655" nextId="3119682">
    <status>ACTIVE</status>
    <commissionRate>1.5</commissionRate>
    <marketType>SINGLE_WINNER_OR_TIE</marketType>
              <selections type="MainBets">
                 <selection id="1032010">
                    <name>Player</name>
                    <resource href=https://api.games.betfair.com/rest/v1/selection/
                     tradeActivity?username=sampluser&marketId=3119655&selectionId=1032010
                     responseType="selectionTradeActivity" title="Trade Activity"/>
                    <status>IN_PLAY</status>
                    <amountMatched>0.0</amountMatched>
                    <profitLoss>-20.0</profitLoss>
                    <bestAvailableToBackPrices/>
                    <bestAvailableToLayPrices/>
                 </selection>
                 <selection id="1032009">
                    <name>Banker</name>
                    <resource href="https://api.games.betfair.com/rest/v1/selection/
                     tradeActivity?username=sampleuser&marketId=3119655&selectionId=1032009"
                     responseType="selectionTradeActivity" title="Trade Activity"/>
                    <status>IN_PLAY</status>
                    <amountMatched>40.0</amountMatched>
                    <profitLoss>20.0</profitLoss>
                    <bestAvailableToBackPrices/>
                    <bestAvailableToLayPrices/>
                  </selection>
                  <selection id="1032011">
                    <name>Tie</name>
                    <resource href="https://api.games.betfair.com/rest/v1/selection/
                     tradeActivity?username=sampleuser&marketId=3119655&selectionId=1032011"
                     responseType="selectionTradeActivity" title="Trade Activity"/>
                    <status>IN_PLAY</status>
                    <amountMatched>0.0</amountMatched>
                    <profitLoss>0.0</profitLoss>
                    <bestAvailableToBackPrices/>
                    <bestAvailableToLayPrices/>
                 </selection>
              </selections>
</market>
```

Figure 20 Excerpt from a “channelSnapshot”, Showing “profitLoss” in the “MainBets” Market for Baccarat

You will notice that in Figure 20 the “profitLoss” value for the “Tie” selection is zero. This will always be the case in this type of betting market, which has been designed specifically for Exchange Baccarat, a game in which customers’ bets on the “Player” and “Banker” selections are returned in the event of a tie between the “Player” and “Banker” objects.

4.2.3.4 Profit OR Loss in a VARIABLE_HANDICAP Market (profitLoss)

If you have any bets placed in a VARIABLE_HANDICAP betting market, for example, in the "CorrectPredictions" market for Exchange Hi Lo, then the "channelSnapshot" element tells you what your potential profit or loss will be for each of four possible scenarios in relation to the selection or selections you have backed or laid.

The possible scenarios are that the game will:

- Not progress as far as an outcome whose selection you have backed or laid.
- Progress only as far as an outcome whose selection you have backed or laid.
- Progress to a point between two outcomes whose selections you have backed or laid.
- Progress beyond an outcome whose selection you have backed or laid.

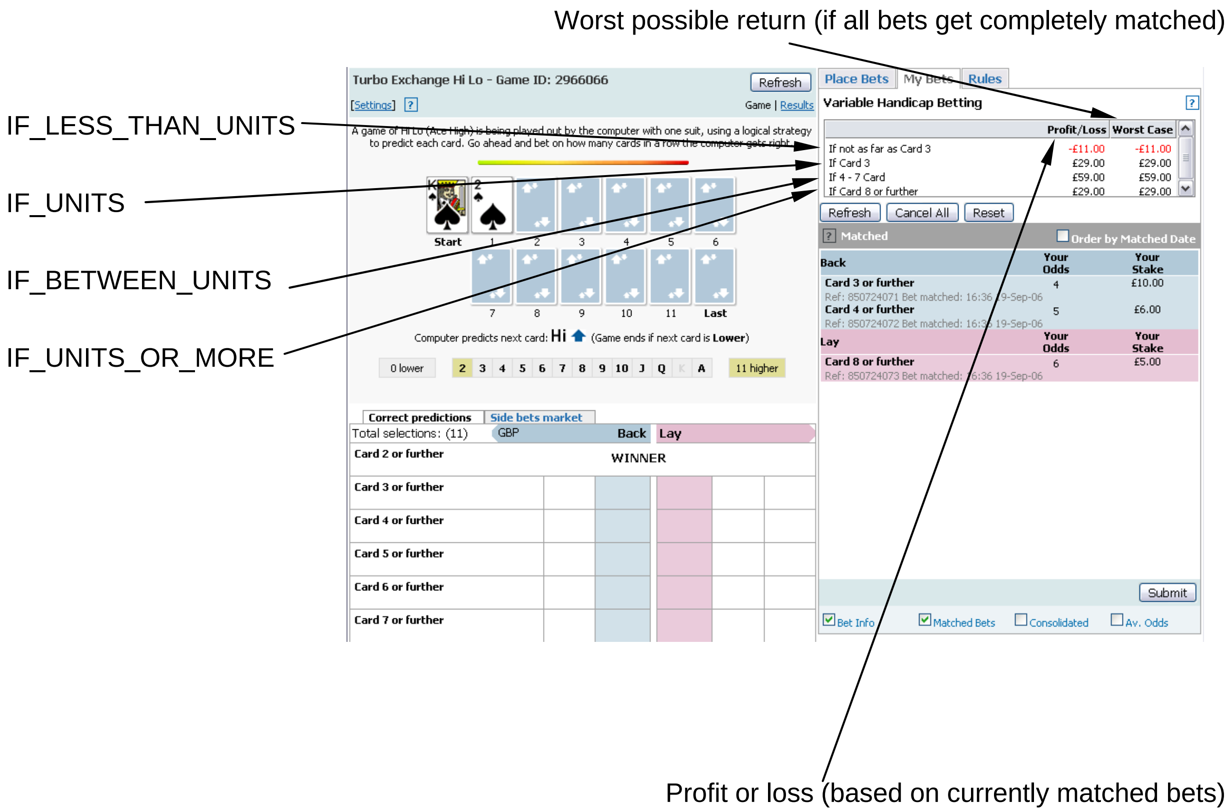
NOTE: If you are not thoroughly familiar with the game of Hi Lo, you may find it useful to read Chapter 9 before reading the rest of this section.

These four possible scenarios are each represented in the "channelSnapshot" for a game of Hi Lo (assuming that you have matched bets on it in the "CorrectPredictions" market). They appear inside the "market" element (see Figure 22) and each one is represented by an element called "variableHandicapProfitLoss".

The "variableHandicapProfitLoss" element contains the following five sub-elements:

- "variableHandicapOutcomeType": this element is a string that identifies the scenario represented by the particular "variableHandicapProfitLoss" element that contains it. It can have one of four possible strings:
    - "IF_LESS_THAN_UNITS",
    - "IF_UNITS",
    - "IF_BETWEEN_UNITS",
    - "IF_UNITS_OR_MORE".

  These correspond, respectively, to each of the four possible scenarios listed in the bullet points above. For example, if you backed the selection "Card 3 or further" in a game of Hi Lo, then the scenario represented by the "variableHandicapOutcomeType" called "IF_LESS_THAN_UNITS" would be the first in the list of bullet points above: it would be that in which the automated player did not predict correctly for Card 1, Card 2, or Card 3 with the result that the game progressed *less far* than the unit represented by your selection (3, since your selection was "Card 3 or further").
- "value1": this represents the card number in your selection; for example, if you select "Card 5 or further", the value for "value1" is 5; if you select "Card 8 or further", the value for "value1" is 8.
- "value2"; this element will only contain a value greater than 0.0 for the particular instance of the "variableHandicapProfitLoss" element called "IF_BETWEEN_UNITS". This instance corresponds to the third scenario in the bullet list above. It requires there to be a minimum of two bets placed. The value of the "value2" element is then the unit represented by your selection (for example, if your highest selection was "Card 10 or further", the second unit would be 10 and this would therefore be the value of "value2").
- "profitLoss": this element contains the profit or loss you will make, based on your currently matched bets, if the scenario represented by the "variableHandicapProfitLoss" element concerned is realised by events. This figure is based on the amounts of your current bets that have so far been matched.
- "worstCase": this element contains the figure for the worst possible return you can make if the scenario represented by the "variableHandicapProfitLoss" element concerned is realised by events. In other words, it tells you what your profit or loss will be if all your bets get matched. (If all your bets do get matched, then the "profitLoss" and "worstCase" values will equal each other.)



A game of Hi Lo (Ace High) is being played out by the computer with one suit, using a logical strategy to predict each card. Go ahead and bet on how many cards in a row the computer gets right.

Computer predicts next card: Hi (Game ends if next card is **Lower**)

	Profit/Loss	Worst Case
If not as far as Card 3	-£11.00	-£11.00
If Card 3	£29.00	£29.00
If 4 - 7 Card	£59.00	£59.00
If Card 8 or further	£29.00	£29.00

Back	Your Odds	Your Stake
Card 3 or further Ref: 850724071 Bet matched: 16:36 19-Sep-06	4	£10.00
Card 4 or further Ref: 850724072 Bet matched: 16:36 19-Sep-06	5	£6.00
Lay	**Your Odds**	**Your Stake**
Card 8 or further Ref: 850724073 Bet matched: 16:36 19-Sep-06	6	£5.00

Total selections: (11)	GBP Back	Lay
Card 2 or further	WINNER	
Card 3 or further		
Card 4 or further		
Card 5 or further		
Card 6 or further		
Card 7 or further		

Figure 21 How Variable Handicap Profit and Loss Appear for Hi Lo on the Betfair Games Website

In the table shown in the screenshot in Figure 21, potential profit and loss figures are listed for a user who has made the following bets:

Card 3 or further: backed for £10 @ odds of 4.0
Card 4 or further: backed for £6 @ odds of 5.0
Card 8 or further: laid for £5 @ odds of 6.0

This user's potential profit and loss are shown for each of the four possible scenarios listed in our bullet points at the top of the previous page, and arrows in Figure 21 also show which rows of the table in the website display correspond to which "variableHandicapOutcomeType" identifiers in the API.

In this sample user's case, the worst-case information is the same as the profit and loss information, because all of the user's bets have been matched.

The meaning of the first row of the table in Figure 21 ("If not as far as card 3") is that, if the automated player does not correctly predict for Card 1, Card 2, or Card 3, the user stands to lose £11.

The meaning of the second row ("If card 3") is that, if the automated player does correctly predict for Card 3, the user stands to win £29.

The meaning of the third row ("If 4 – 7 Card") is that, if the automated player predicts correctly for any card between Card 4 and Card 7, the user stands to win £59.

And the meaning of the fourth row ("If Card 8 or further") is that, if the automated player predicts correctly for Card 8 and/or any card beyond Card 8, the user stands to win £29. (The reason that the user stands to win less in this case than in the previous one is that is that the bet on "Card 8 or further" was a lay bet.)

A sample "market" element (from a "channelSnapshot" for Hi Lo) showing the "CorrectPredictions" data corresponding to Figure 21 is shown below (Figure 22).

```
<market id="2966066" nextId="2966093">
  <status>ACTIVE</status>
  <commissionRate>2.0</commissionRate>
  <marketType>VARIABLE_HANDICAP</marketType>
  <variableHandicapProfitLoss>
        <variableHandicapOutcomeType>IF_LESS_THAN_UNITS</variableHandicapOutcomeType>
        <value1>3.0</value1>
        <value2>0.0</value2>
        <profitLoss>-11.0</profitLoss>
        <worstCase>-11.0</worstCase>
  </variableHandicapProfitLoss>
  <variableHandicapProfitLoss>
        <variableHandicapOutcomeType>IF_UNITS</variableHandicapOutcomeType>
        <value1>3.0</value1>
        <value2>0.0</value2>
        <profitLoss>29.0</profitLoss>
        <worstCase>29.0</worstCase>
  </variableHandicapProfitLoss>
  <variableHandicapProfitLoss>
        <variableHandicapOutcomeType>IF_BETWEEN_UNITS</variableHandicapOutcomeType>
        <value1>4.0</value1>
        <value2>7.0</value2>
        <profitLoss>59.0</profitLoss>
        <worstCase>59.0</worstCase>
  </variableHandicapProfitLoss>
  <variableHandicapProfitLoss>
        <variableHandicapOutcomeType>IF_UNITS_OR_MORE</variableHandicapOutcomeType>
        <value1>8.0</value1>
        <value2>0.0</value2>
        <profitLoss>29.0</profitLoss>
        <worstCase>29.0</worstCase>
  </variableHandicapProfitLoss>
  <selections type="CorrectPredictions">
        <selection id="1032050">
           <name>Card 1 or further</name>
           <resource href="https://api.games.betfair.com/rest/v1/selection/
            tradeActivity?username=sampleuser &marketId=2966066&selectionId=1032050"
            responseType="selectionTradeActivity" title="Trade Activity”
           <status>WINNER</status>
           <amountMatched>0.0</amountMatched>
           <bestAvailableToBackPrices/>
           <bestAvailableToLayPrices/>
        </selection>
        ...
         </selections>
</market>
```

Figure 22 Excerpt from a “channelSnapshot”, Showing the “CorrectPredictions” Market for Hi Lo[8]

[8] The dots in the sample XML output indicate the omission of repetitive data.

4.3 Requesting XML Listing the Previous Games in a Channel (channelHistory)

Using the following URL, you can view the previous instances of the game you are interested in:

https://api.games.betfair.com/rest/v1/channels/*channelID*/history?username=sampleuser

where *channelID* is an integer value identifying the channel containing the game.

The call to this URL is an HTTP GET request. The name of the XML element returned is “channelHistory”.

The paging elements “nextPage” and “previousPage” (see Figure 23) contain a URL that enables you to navigate one page backward and then one page forward respectively (the sample output in Figure 23 is the first page of the channel’s history, therefore there is no previous page to view).

```
<channelHistory xmlns="urn:betfair:games:api:v1">
    <channel gameType="POKER" id="1444074" name="Exchange Poker Texas Hold'em">
        <games end="5" start="1" total="5">
            <nextPage>
                <resource href=https://api.games.betfair.com/rest/v1/channels/1444074/
                 history?username=sampleuser&amp;startRecord=6&amp;recordCount=5
                 responseType="channelHistory" title="Next Channel History Results"/>
            </nextPage>
            <game gameStartDate="2006-05-16T11:27:43+00:00">
                <gameData>
                    <object name="Hand 1">
                        <description>One Pair, Queens, Ace High</description>
                        <status>LOSER</status>
                        <property name="Card 1" value="28"/>
                        <property name="Card 2" value="4"/>
                    </object>
                    <object name="Hand 2">
                        <description>Three of a Kind, Queens, Ace High,
                          with K Kicker
                        </description>
                        <status>WINNER</status>
                        <property name="Card 1" value="51"/>
                        <property name="Card 2" value="37"/>
                    </object>
                    ...
                </gameData>
                <markets currency="GBP">
                    <market id="2604163">
                        <status>SETTLED</status>
                        <marketType>WIN_ONLY</marketType>
                        <selections type="Mainbets">
                           <selection>
                             <name>Hand 1</name>
                             <status>LOSER</status>
                           </selection>
                           <selection>
                             <name>Hand 2</name>
                             <status>WINNER</status>
                           </selection>
                        ...
                        </selections>
                    </market>
                </markets>
            </game>
           ...
        </games>
    </channel>
</channelHistory>
```

Figure *23* The "channelHistory" Element with One Past Poker Game Shown (dots indicate omitted data)

The sample output in Figure 23 illustrates the history of a game of Poker in the channel 1444074. The first set of dots in the illustration indicates that after Hand 2 the “object” elements for the players are omitted from the illustration. The second set of dots indicates that after Hand 2 the “selection” elements in the betting market are omitted. The third set of dots indicates that the list of other “game” elements (each one containing the record of an entire game and its market or markets) is omitted.

There will be a maximum of 5 games per page.

The “channelHistory” element contains an element called “channel”, which in turn contains an element called “games”. The “games” element contains a list of “game” elements. And the “game” elements each contain a “gameData” element and a “markets” element. The “gameData” element contains the final details of a single game: the details of the automated players’ final score or rank, their final hands of cards (if applicable), and their final status (this information being contained inside each “object” element inside the “gameData” element. For Baccarat, the “object” element representing the “shoe” contains only a single property, its “id”.

For details of the players’ possible final scores and rankings, see the sections in Chapter 5, Chapter 6, Chapter 8, and Chapter 9 that explain the rules adopted by Betfair for their implementation of each game.

The “markets” element contains the details of the game’s market or, if applicable, its markets (Baccarat has two markets). The market information is held inside the “market” element.

For each betting selection in “channelHistory”, only the selection’s name and whether it was a winning or losing selection are provided. There is no information stored about prices and commission rates. Therefore, clearly, the “markets” and “market” elements are not identical with the elements of the same name that are contained in the “channelSnapshot” element (see **Error! Reference source not found.**).

4.3.1 Limiting the List of Past Games Returned (channelHistory)

To each of the URLs for requesting the “channelHistory” element (see Figure 11 in Section 4.1), you can add a parameter manually to:

- Specify the start position (that is, the start position in the Games API’s overall list) for the list of games that you want to receive. The parameter (including the necessary ampersand character) for this is:

  ```
  &startRecord=number
  ```

  where *number* is the position (in the overall list of games played over the last 24 hours) of the first game record that you want to see returned.

- Specify the number of games that you want the list returned to contain. The parameter (including the necessary ampersand character) for this is:

  ```
  &recordCount=number
  ```

- Specify both the start position (in the overall list of games in “channelHistory”) for the list that you want to receive and the number of games to be included in the list. To do this, specify both of the above parameters in the URL when you send your request.

4.3.2 Requesting the Record of a Specific Past Game (channelHistory)

To each of the URLs for requesting the “channelHistory” element (see Figure 11 in Section 4.1), you can add a parameter manually to limit the output to information about a single past game. The parameter (including the necessary ampersand character) for this is:

```
marketId=number
```

where *number* is the integer value representing the “marketId” for the betting market (or one of the betting markets in the case of Baccarat) belonging to a particular past game.

The XML returned will list the game details and the details of the market you specified by using the “marketId” parameter.

4.4 Requesting XML Stating When Betting is Permitted in a Game (channelInfo)

The URL for finding out how much time is permitted for betting during a round of cards is:

https://api.games.betfair.com/rest/v1/channels/*channelID*/info?username=sampleuser

where *channelD* is the number of the channel containing the game you are interested in.

The call to this URL is an HTTP GET request. The name of the XML element returned is “channelInfo”.

```
<channelInfo xmlns="urn:betfair:games:api:v1">
  <round number="1">
    <bettingWindow>15</bettingWindow>
    <playingWindow>5</playingWindow>
  </round>
  <round number="2">
    <bettingWindow>30</bettingWindow>
    <playingWindow>5</playingWindow>
  </round>
  <round number="3">
    <bettingWindow>30</bettingWindow>
    <playingWindow>5</playingWindow>
  </round>
  <round number="4">
    <bettingWindow>30</bettingWindow>
    <playingWindow>5</playingWindow>
  </round>
  <round number="5">
    <bettingWindow>0</bettingWindow>
    <playingWindow>5</playingWindow>
  </round>
  <gameRules>NOT AVAILABLE</gameRules>
  <gameDescription>
   Object names will be presented as "Hand <id>", where <id> is a number
   greater than 0. The "Hand <id>" will also be the name of a corresponding
   selection in the betting market.

   The community cards will be presented as belonging to an object called
   "Community Cards".

   In all cases the card names will be presented as "Card <id>" where
   <id> is a number greater than 0. The value of each card will be presented
   as a number (between 0 and 51) which maps to a single card in the
   traditional deck according to the following matrix.

   Values Returned for 'Card' Objects:

            | Ace |  2 |  3 |  4 |  5 |  6 |  7 |  8 |  9 | 10 |  J |  Q | K  |
   ---------+-----+----+----+----+----+----+----+----+----+----+----+----+----+
   clubs    |  0  |  1 |  2 |  3 |  4 |  5 |  6 |  7 |  8 |  9 | 10 | 11 | 12 |
   diamonds | 13  | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |
   hearts   | 26  | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 |
   spades   | 39  | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 |

  When a card has not been turned over / dealt, the value "NOT AVAILABLE"
  is returned.
 </gameDescription>
</channelInfo>
```

Figure *24* The “channelInfo” Element Containing the Betting Window and Time Between Rounds

NOTE: The “channelInfo” element is also the one that contains a text description of the way in which the Exchange Games API uses the generic elements defined in the XML schema to represent the game that the channel is being used for.

The “bettingWindow” element inside “channelInfo” (see Figure 24) contains a number representing the time (in seconds) that is available for placing bets during the current round of the game. The difference between the regular and turbo versions of the game lies in the length of the “bettingWindow”.

The “playingWindow” element (also in “channelInfo”) contains a number representing the time (in seconds) for which betting is suspended while cards are dealt to the automated players.

The “playingWindow” is a minimum value: it represents the *minimum* time that the Exchange Games engine will spend completing that round of the game. Under a period of heavy load on the engine, the “playingWindow” might last longer than the time indicated by the value in this element (it will never be shorter).

If you are betting on a game, we recommend you configure your client application to request an updated “channelSnapshot” of the game every 3 seconds or so during each “playingWindow”. A “playingWindow” corresponds to a round of the game: it is when the cards are dealt to the automated players. The Exchange Games engine updates the game data approximately every three seconds during the “playingWindow”.

NOTE: During the “bettingWindow” (the time during which you can place bets on the possible outcomes of the game) the Exchange Games engine does not update the game at all. Therefore you will gain no new information about the game by requesting a “channelSnaphot” during the “bettingWindow”. Also, for many rounds of games, there is no need for your client application to poll more than once, because the game data is only updated once during the round. This is true for Poker and Baccarat, for example. But after round 1 of Blackjack, the game data is updated several times during the “playingWindow”, therefore it is worth polling every few seconds during the “playingWindow” for Blackjack after round 1.

You can tell from the “channelSnapshot” element (see Figure 12 and Figure 26) approximately when the next “playingWindow” opens (in other words, when the next round of the game begins), because the “channelSnapshot” contains both a “bettingWindow” element (stating the length of the current betting period) and a “bettingWindowPercentageComplete” element (stating what percentage of the current betting period has elapsed).

So you can configure your client application to start sending “channelSnapshot” requests when the “bettingWindowPercentageComplete” approaches 100%.

For information about how to request a “channelSnapshot” that contains only the betting round information, see Section 4.2.1.4.

For information about sending betting requests in bursts for a specified total stake at or around the beginning of a betting round, see Chapter 12.

```
<channelSnapshot xmlns="urn:betfair:games:api:v1">
  <channel gameType="POKER" id="1444074" name="Exchange Poker Texas Hold'em">
  <status>RUNNING</status>
    <game id="170382">
       <round>3</round>
       <bettingWindowTime>30</bettingWindowTime>
       <bettingWindowPercentageComplete>0</bettingWindowPercentageComplete>
       ...
```

Figure *25* Part of a “channelSnapshot” Element Showing the Betting Period Between Rounds of the Game

Chapter 5 Exchange Texas Hold’em Poker

This chapter describes the Betfair Exchange Games version of Texas Hold’em Poker and the XML that the Exchange Games API uses to represent it. The chapter tells you:

- Which URLs to use to request the current state of a game of Exchange Texas Hold’em Poker,
- The rules adopted by the Betfair implementation of Texas Hold’em Poker,
- How the generic XML elements defined in the Betfair Exchange Games API schema are used specifically to represent Texas Hold’em Poker, and
- Which URL to use to request information about the betting time available after each round of a game.

5.1 The URLs for Accessing a Game of Exchange Texas Hold’em Poker

The URLs for accessing a game of Exchange Texas Hold’em Poker are presented in the “channelLanding” page, described in Section 4.1. They are shown in Figure 26, with the elements identifying the channel and game printed in bold-type.

```
<channel gameType="POKER" id="1444074" name="Exchange Poker Texas Hold'em">
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444074/snapshot
   ?username=tex" responseType="channelSnapshot" title="Channel Snapshot"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444074/history
   ?username= sampleuser" responseType="channelHistory" title="Channel history"/>
  <resource  href="https://api.games.betfair.com/rest/v1/channels/1444074/info
   ?username=sampleuser" responseType="channelInfo" title="Channel Definition"/>
</channel>
<channel gameType="POKER" id="1444080" name="Turbo Exchange Poker">
  <resource  href="https://api.games.betfair.com/rest/v1/channels/1444080/snapshot
   ?username=sampleuser" responseType="channelSnapshot" title="Channel Snapshot"/>
 <resource  href="https://api.games.betfair.com/rest/v1/channels/1444080/history
  ?username=sampleuser" responseType="channelHistory" title="Channel history"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444080/info
  ?username=sampleuser" responseType="channelInfo" title="Channel Definition"/>
</channel>
```

Figure 26 Excerpt from a “channelLanding” Page Showing “resource” Elements for Texas Hold’em Poker

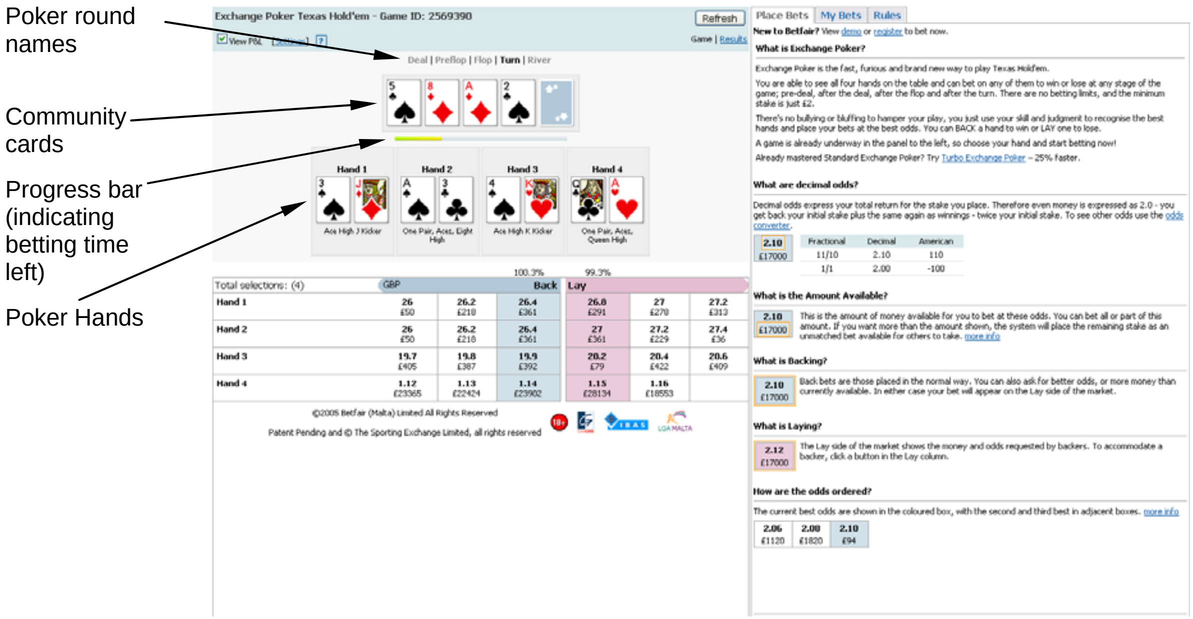
NOTE: Before you place any bets on this game, make sure you understand the rules that have been adopted for the Betfair version of it. They may differ from the rules that you are used to (see Section 5.2).

5.2 The Rules Followed by Exchange Texas Hold’em Poker

The Betfair game of Exchange Poker is based on the version of Poker known as 'Texas Hold'em'. This section describes the progress of the game, the rankings of the different winning hands that are possible, the rules for deciding the winning hand in the event of a tie between hands, and the dead-heat rule that is used for bet settlement when the tie-breaker logic has not yielded a single winner.

5.2.1 The Progress of a Game of Exchange Texas Hold’em Poker

Each game begins with 13 cards from one randomly shuffled pack being shown face-down. Eight of the 13 cards are then paired to form four separate hands. The cards in these pairs are called “hole cards”. The remaining five cards form a pool of Community Cards.



Exchange Poker Texas Hold'em - Game ID: 2569390

Deal | Preflop | Flop | **Turn** | River

Hand 1	Hand 2	Hand 3	Hand 4
Ace High J Kicker	One Pair, Aces, Eight High	Ace High K Kicker	One Pair, Aces, Queen High

Total selections: (4)	GBP		100.3% Back	99.3% Lay		
Hand 1	**26** £50	**26.2** £210	**26.4** £361	**26.8** £291	**27** £270	**27.2** £313
Hand 2	**26** £50	**26.2** £218	**26.4** £361	**27** £361	**27.2** £229	**27.4** £36
Hand 3	**19.7** £405	**19.8** £387	**19.9** £392	**20.2** £79	**20.4** £422	**20.6** £409
Hand 4	**1.12** £23365	**1.13** £22424	**1.14** £23902	**1.15** £28134	**1.16** £18553	

©2005 Betfair (Malta) Limited All Rights Reserved

Patent Pending and © The Sporting Exchange Limited, all rights reserved

New to Betfair? View demo or register to bet now.

What is Exchange Poker?

Exchange Poker is the fast, furious and brand new way to play Texas Hold'em.

You are able to see all four hands on the table and can bet on any of them to win or lose at any stage of the game; pre-deal, after the deal, after the flop and after the turn. There are no betting limits, and the minimum stake is just £2.

There's no bullying or bluffing to hamper your play, you just use your skill and judgment to recognise the best hands and place your bets at the best odds. You can BACK a hand to win or LAY one to lose.

A game is already underway in the panel to the left, so choose your hand and start betting now!

Already mastered Standard Exchange Poker? Try Turbo Exchange Poker – 25% faster.

What are decimal odds?

Decimal odds express your total return for the stake you place. Therefore even money is expressed as 2.0 - you get back your initial stake plus the same again as winnings - twice your initial stake. To see other odds use the odds converter.

2.10 £17000

Fractional	Decimal	American
11/10	2.10	110
1/1	2.00	-100

What is the Amount Available?

2.10 £17000 This is the amount of money available for you to bet at these odds. You can bet all or part of this amount. If you want more than the amount shown, the system will place the remaining stake as an unmatched bet available for others to take. more info

What is Backing?

2.10 £17000 Back bets are those placed in the normal way. You can also ask for better odds, or more money than currently available. In either case your bet will appear on the Lay side of the market.

What is Laying?

2.12 £17000 The Lay side of the market shows the money and odds requested by backers. To accommodate a backer, click a button in the Lay column.

How are the odds ordered?

The current best odds are shown in the coloured box, with the second and third best in adjacent boxes. more info

2.06 £1120	**2.08** £1820	**2.10** £94

Figure *27* A Representation of a Game of Texas Hold’em from the Betfair Exchange Games Website

The first round of betting occurs while all cards are face-down (this round of betting follows the first round of cards, which is called the “deal”). You can bet on the individual hand you think will win the game, or you can lay individual hands that you think will lose (you can place bets to back or lay hands).

Each round of betting lasts for a fixed period of time before unmatched bets are cancelled and the next cards turned face-up. The length of time left for betting is indicated in the “channelSnapshot” element by the value for “bettingWindowPercentageComplete” (see Figure 25 in Section 4.4). On the Betfair Exchange Games website the length of time left for betting is indicated by the progress bar just below the Community Cards, as shown in Figure 27.)

NOTE: Both the “bettingWindowPercentageComplete” value in “channelSnapshot” and the progress bar on the website provide an indication only.

At the end of the first round of betting, unmatched bets are cancelled and the pairs of cards forming the individual hands are simultaneously turned face-up.

There is then another round of betting and, again, you can back or lay individual hands within the game (this round of betting follows the second round of cards, which is called the “Pre-flop” round).

At the end of the second round of betting, unmatched bets are cancelled and the first three of the Community Cards are turned face-up (this event is called the “flop”).

A third round of betting then begins, at the end of which all unmatched bets are cancelled, and then the fourth card from the Community Cards is turned face-up (this card is sometimes itself called the '”turn”).

Another round of betting then commences. This fourth and final round of betting finishes with all unmatched bets being cancelled before the fifth and last of the Community Cards is turned face-up (the turning of this card is called the “river”).

The winner of the game is the hand “object” that has the highest value within the ranking system that the Betfair implementation of Texas Hold’em follows (see Section 5.2.2).

If two hands have an equal ranking, dead-heat rules apply (see Section 5.2.4).

5.2.2 The System for Ranking Hands in Exchange Texas Hold’em Poker

Throughout a game of Exchange Texas Hold’em Poker, and at the end of it too, the Exchange Games engine regards each hand as holding the best combination of cards possible to it, given its “hole” cards and the cards available from the Community Cards. (In other words, all the automated players are assumed to have perfect Poker skills and to be able to maximize their own chances of winning by spotting the best card combinations that conditions allow them at a given moment.)

At the end of a game of Exchange Texas Hold’em Poker, the card combination that each hand offers (after making its best effort to win throughout the game and without making any mistakes in this regard) will contain either:

- Two “hole” cards (the cards originally dealt to the hand) plus three Community Cards,
- One “hole” card plus four Community Cards, or
- All five Community Cards

The ranking that the Exchange Games engine uses to determine the winning hand is shown in Table 5.

Table 5 The Rankings for the Possible Winning Hands in Exchange Texas Hold’em Poker

Rank	Type of Winning Hand	Card Combination Forming Type of Winning Hand
1	Royal Flush	Ace, King, Queen, Jack, Ten of the same suit.
2	Straight Flush	Straight with all five cards in the same suit.
3	Four of a Kind	Four cards of the same number or face value (“quads”).
4	Full House	Three cards of one number or face value and two cards of another number or face value. If more than one player has a full house, the full house with the highest ranking three-of-a-kind (“trips”) wins.
5	Flush	Five cards of the same suit. If there is more than one flush, the hand with the highest card(s) wins.
6	Straight	Five cards in sequence. The cards can be in any suit. An Ace can be used in the highest straight (10, J, Q, K, A) and the lowest straight (A, 2, 3, 4, 5). If there is more than one straight, the hand with the highest card in the straight wins.
7	Three of a Kind	Three cards of the same number or face value (“trips”).
8	Two Pair	If two hands have a two pair, the hand with the highest pair wins. If they have the same high pair, then – of the two – the one that has the second highest pair wins. If both hands have the same two pair, then the one that has the highest fifth card (the “kicker”) wins.
9	One Pair	Two cards of the same number or face value. If two hands have the same pair, the one with the highest card (or cards) outside the pair wins.
10	High Card	The hand with the highest card (or cards) wins.

5.2.3 The Tie-breaker for Exchange Texas Hold’em Poker

If two or more hands hold cards or combinations of cards with equal rank and points value, then the best one of any cards that each holds outside of the cards forming their winning combinations is compared with the best that the other hands hold outside of their winning combinations. The card that each winning hand puts forward in this situation is called a “kicker”. The winner of the game is the hand with the highest value “kicker”. If there is still a tie after the comparison of “kicker” cards, a second comparison of “kicker” cards takes place (if cards in the affected hands are available) with each winning hand’s next best “kicker” card. If there is still a tie after this between the hands, then bets are settled according to the dead-heat rule.

5.2.4 The Dead-heat Rule for Bet Settlement in Exchange Texas Hold’em Poker

If two or more hands are still of equal value after the “kicker” has been tried, then dead-heat rules apply to the winning hands (the application of dead-heat rules is known as “splitting the pot”).

The dead-heat rule applies to bets on Exchange Texas Hold’em Poker when there is more than one winner, in other words, when two or more hands have tied.

In these cases, the odds received by the backers of each hand that won are divided by the number of hands that won, and bet settlement then takes place based upon the new odds.

For example, if Client A backs Hand 1 with £100 at odds of 4.0, and the outcome of the game is that Hand 1 and Hand 3 tie with each other, then Client A’s bet will be re-calculated as £100 placed at odds of 2.0; Client A will therefore receive £100 in winnings.

NOTE: The application of the dead-heat rule makes it possible for you to receive back less than your original stake when the bets are settled after a game. For example, if Client A had backed Hand 1 with £100 at odds of 1.2 and Hand 1 and Hand 3 had tied, then the total payout for the bet would be:

£100 x (1.2/2) = £60

This would represent a net loss of £40, even though Client A’s selection (Hand 1) did not lose the game.

When the dead-heat rule is applied, determining whether a particular bet is a net winner or a net loser depends on the number of hands involved in the dead heat and the odds at which the bet was matched.

5.3 The Main XML Elements Used to Represent a Game of Exchange Texas Hold’em Poker

The XML elements (principally the “object” and “selection” elements) that are used to describe the games available via the Exchange Games API are generic; in other words, they are the same for each game.

Figure 12 (on page 25) shows an XML snapshot of a game of Turbo Exchange Poker. The two main elements used are “object” (which is nested inside the element called “game”), and “selection” (which is nested inside the element called “market”).

Figure 28 (on page 51) shows how some of the data in the Exchange Games API’s “channelSnapshot” output for Poker is represented on the main Betfair website.

You also need to be aware of the “marketType” for Poker. This determines the number of winning selections that are possible in a betting market and the process involved in settling bets where more than one betting selection has won.

5.3.1 The Automated Players and Other "object" Elements for Texas Hold'em Poker (object)

For Exchange Texas Hold'em Poker, the "object" elements represent, respectively, the four automated players and the community cards. Each "object" has:

- A name ("Hand 1", "Hand 2", "Hand 3", "Hand 4", or "Community Cards"),
- A description (which, in the case of the "Community Cards" object, will always be empty and, in the case of the hand objects, will in the course of the game – and as applicable - describe the hand as one of the potentially winning card combinations recognised in Poker, for example, "straight", "flush", "four of a kind" etc),
- A status (which for the Betfair Poker game can be IN_PLAY, LOSER, WINNER, although the "Community Cards" object is an anomaly in this regard as it always has a status of N/A, meaning "not applicable"),
- And one or more "property" elements, each representing a card. These "property" elements each have two attributes: "name" and "value". The "name" is of the form "`Card` ***`number`***", where ***`number`*** is an integer value greater than zero (in other words, "Card 1", "Card 2", and so on). The "value" of the card will be a number from 0 to 51, each value corresponding in a matrix to one member of the traditional deck of cards. The mappings used by the Games API between the numbers 0 to 51 and the members of the deck are shown in Figure 29. (They are also published by Betfair at the following location on their Games website: http://stats.betfairgames.com/

  NOTE: From this URL (http://stats.betfairgames.com/) you can also download images of all the cards in a traditional deck of cards. These images remain the copyright of Betfair Ltd., but you are permitted to download them free of charge for personal and non-commercial use in the development of software that accesses the Betfair Exchange Games API.

Finally, in Exchange Texas Hold'em Poker, each hand object will always have two cards, and by the end of the game the "Community Cards" object will have a maximum of five cards and a minimum of three.

5.3.2 The "marketType" Element for Exchange Texas Hold'em Poker (market)

The betting market for Exchange Texas Hold'em Poker is a win-only market.

The value of its "marketType" element is "WIN_ONLY" (see Table 4 in Section 4.2).

A win-only market, by definition, has only one winning selection (all other selections being necessarily losing ones).

Bets backing the winner of a game or laying its losers are settled by the Exchange Games engine in favour of the bettor.

It is possible for a dead heat to apply in a win-only market. In this case, the odds received by the respective backers are each divided by the number of winners, and bet settlement is based upon the new odds. (For further details of the dead-heat rule for Exchange Texas Hold'em Poker, see Section 5.2.4.)

If a hand in a win-only market for a game of Exchange Texas Hold'em Poker is eliminated before the end of the game (in other words, if none of the cards remaining in the deck can give that hand a higher rank than a competing hand), then bets on the hand are settled immediately.

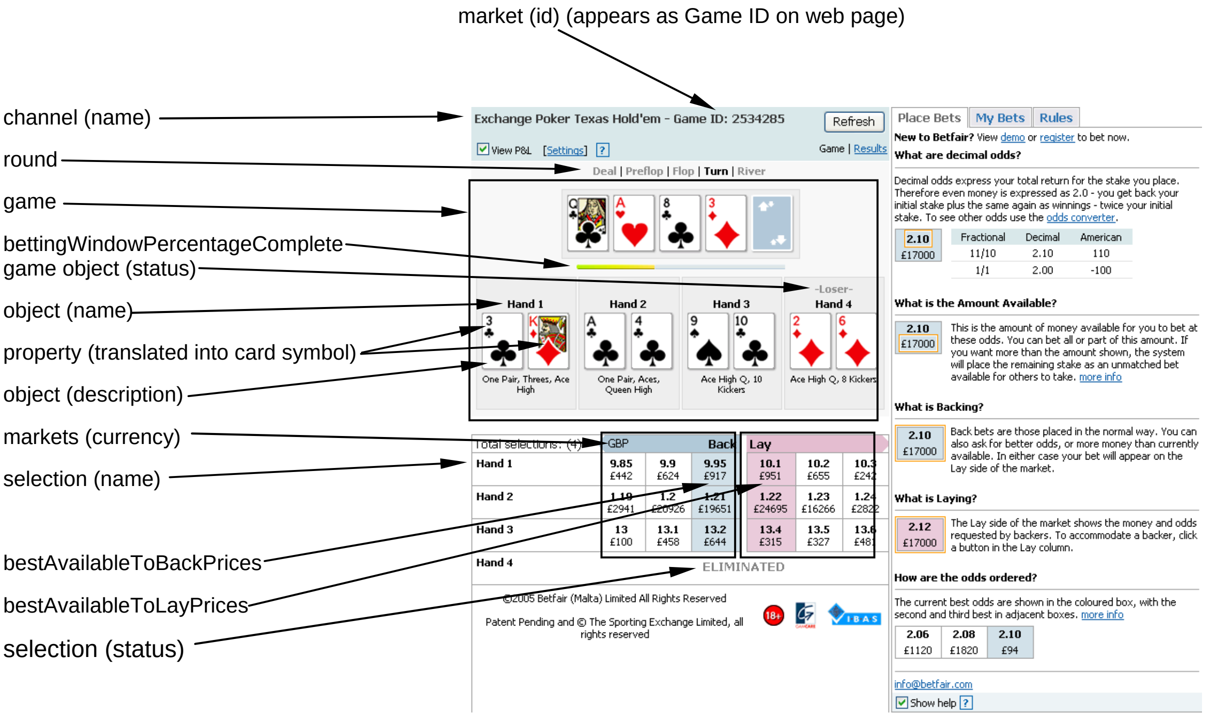
When only one hand can possibly win, the game ends (even if not all of the five Community Cards have been dealt).

5.3.3 The “selection” Element for Exchange Texas Hold’em Poker (selection)

Each betting “market” is represented as a series of elements called “selection”. In Betfair terminology, the word “selection” means a betting choice – that is, a possible outcome. So for Poker the betting market comprises the names of the four virtual hands that can possibly win the game.

Each “selection” element has:

- a name that corresponds to the name of one possible winner of the game, for example, “Hand 1”, “Hand 2”, and so on,
- a “selectionTradeActivity” resource for viewing all available prices for the selection (see Section 4.2.2),
- a “status” (which can be IN_PLAY, LOSER, TIED_DEAD_HEAT, or WINNER),
- a figure for the amount of the bet that has been matched (this is presented in the element called “amountMatched”)
- and figures for the best back and lay prices.



Exchange Poker Texas Hold'em - Game ID: 2534285 Refresh

☑ View P&L [Settings] [?] Game | Results

Deal | Preflop | Flop | **Turn** | River

-Loser-

Hand 1	Hand 2	Hand 3	Hand 4
One Pair, Threes, Ace High	One Pair, Aces, Queen High	Ace High Q, 10 Kickers	Ace High Q, 8 Kickers

Total selections: (4)	GBP		Back	Lay		
Hand 1	**9.85** £442	**9.9** £624	**9.95** £917	**10.1** £951	**10.2** £655	**10.3** £242
Hand 2	**1.19** £2941	**1.2** £20926	**1.21** £19651	**1.22** £24695	**1.23** £16266	**1.24** £2822
Hand 3	**13** £100	**13.1** £458	**13.2** £644	**13.4** £315	**13.5** £327	**13.6** £481
Hand 4			ELIMINATED			

©2005 Betfair (Malta) Limited All Rights Reserved

Patent Pending and © The Sporting Exchange Limited, all rights reserved

Place Bets | My Bets | Rules

New to Betfair? View demo or register to bet now.

What are decimal odds?

Decimal odds express your total return for the stake you place. Therefore even money is expressed as 2.0 - you get back your initial stake plus the same again as winnings - twice your initial stake. To see other odds use the odds converter.

2.10 £17000

Fractional	Decimal	American
11/10	2.10	110
1/1	2.00	-100

What is the Amount Available?

2.10 £17000 This is the amount of money available for you to bet at these odds. You can bet all or part of this amount. If you want more than the amount shown, the system will place the remaining stake as an unmatched bet available for others to take. more info

What is Backing?

2.10 £17000 Back bets are those placed in the normal way. You can also ask for better odds, or more money than currently available. In either case your bet will appear on the Lay side of the market.

What is Laying?

2.12 £17000 The Lay side of the market shows the money and odds requested by backers. To accommodate a backer, click a button in the Lay column.

How are the odds ordered?

The current best odds are shown in the coloured box, with the second and third best in adjacent boxes. more info

2.06 £1120	**2.08** £1820	**2.10** £94

info@betfair.com

☑ Show help [?]

Figure *28* How the Data Structures for Texas Hold’em Currently Appear on the Betfair Games Website

NOTE: The selection status “ELIMINATED” shown in the web site presentation of a game (Figure 28) is not a string returned by the Games API. The corresponding string in the Games API would be “LOSER”.

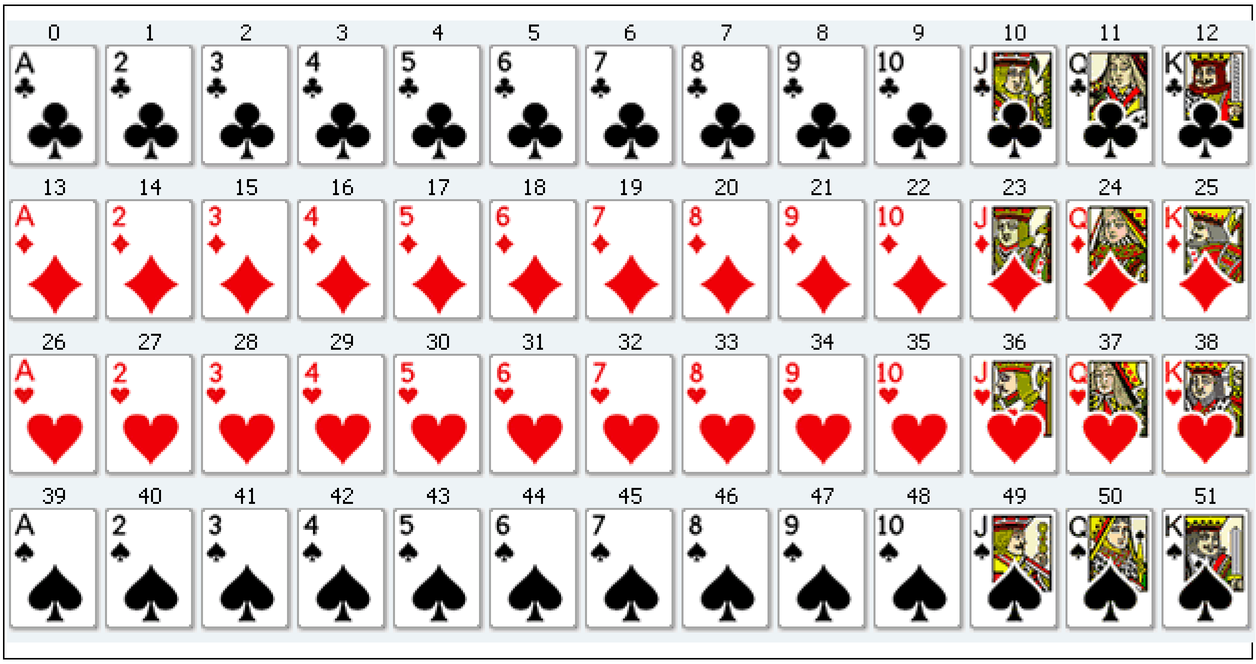


Figure *29* The Mappings Between Games-API-assigned Card “values” and the Cards in a Deck

NOTE: The value “NOT AVAILABLE” is used by the Betfair Exchange Games API to indicate a card that has not been turned over and whose suit and face value therefore cannot be identified.

5.3.4 Requesting XML Stating How Texas Hold’em Poker is Represented (channelInfo)

Using the following URL, you can read a brief description (contained inside the “gameDescription” element) of the way that the Games API uses the generic “object” and “selection” elements to represent a game of Texas Hold’em Poker and its betting market:

https://api.games.betfair.com/rest/v1/channels/*channelId*/info?username=sampleuser

where *channelID* is an integer value identifying the channel (this value is automatically included in the URLs returned for each channel in the “channelLanding” page, see Figure 11 above).

```
<channelInfo xmlns="urn:betfair:games:api:v1">
  <round number="1">
    <bettingWindow>15</bettingWindow>
    <playingWindow>5</playingWindow>
  </round>
  <round number="2">
    <bettingWindow>30</bettingWindow>
    <playingWindow>5</playingWindow>
  </round>
  <round number="3">
    <bettingWindow>30</bettingWindow>
    <playingWindow>5</playingWindow>
  </round>
  <round number="4">
    <bettingWindow>30</bettingWindow>
    <playingWindow>5</playingWindow>
  </round>
  <round number="5">
    <bettingWindow>0</bettingWindow>
    <playingWindow>5</playingWindow>
  </round>
  <gameRules>NOT AVAILABLE</gameRules>
  <gameDescription>
   Object names will be presented as "Hand <id>", where <id> is a number
   greater than 0. The "Hand <id>" will also be the name of a corresponding
   selection in the betting market.

   The community cards will be presented as belonging to an object called
   "Community Cards".

   In all cases the card names will be presented as "Card <id>" where
   <id> is a number greater than 0. The value of each card will be presented
   as a number (between 0 and 51) which maps to a single card in the
   traditional deck according to the following matrix.

   Values Returned for 'Card' Objects:

            | Ace |  2 |  3 |  4 |  5 |  6 |  7 |  8 |  9 | 10 |  J |  Q | K  |
   ---------+-----+----+----+----+----+----+----+----+----+----+----+----+----+
   clubs    |  0  |  1 |  2 |  3 |  4 |  5 |  6 |  7 |  8 |  9 | 10 | 11 | 12 |
   diamonds | 13  | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |
   hearts   | 26  | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 |
   spades   | 39  | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 |

   When a card has not been turned over / dealt, the value "NOT AVAILABLE"
   is returned.
  </gameDescription>
 </channelInfo>
```

Figure *30* The gameDescription (inside the “channelInfo” element) for Exchange Texas Hold’em Poker

For further information about the “channelInfo” element, refer to Section 4.4.

Chapter 6 Exchange Omaha Hi Poker

This chapter describes the Betfair Exchange Games version of Omaha Hi Poker and the XML that the Exchange Games API uses to represent it. The chapter tells you:

- Which URLs to use to request the current state of a game of Exchange Omaha Hi Poker,
- The rules adopted by the Betfair implementation of Omaha Hi Poker,
- How the generic XML elements defined in the Betfair Exchange Games API schema are used specifically to represent Omaha Hi Poker, and
- Which URL to use to request information about the betting time available after each round of a game.

6.1 The URLs for Accessing a Game of Exchange Omaha Hi Poker

The URLs for accessing a game of Omaha Hi Poker are presented in the “channelLanding” page, described in Section 4.1. They are shown in Figure 31, and the element identifying the channel and game is printed in bold-type.

```
<channel gameType="OMAHA_POKER" id="1444100" name="Exchange Omaha Hi">
  <resource href="https:// api.games.betfair.com/rest/v1/channels/1444100/snapshot
   ?username=sampleuser " responseType="channelSnapshot" title="Channel Snapshot"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444100/history
   ?username=sampleuser" responseType="channelHistory" title="Channel history"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444100/info
   ?username=sampleuser " responseType="channelInfo" title="Channel Definition"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444100/snapshot
   ?username=sampleuser&selectionsType=MainBets" responseType="channelSnapshot"
   title="Channel Snapshot - MainBets"/>
  <resource href="https://api.games.betfair.com/rest/v1/channels/1444100/snapshot
   ?username=sampleuser&selectionsType=SideBets" responseType="channelSnapshot"
   title="Channel Snapshot - SideBets"/>
</channel>
```

Figure 31 Excerpt from a “channelLanding” Page Showing “resource” Elements for Omaha Hi Poker

NOTE: Before you place any bets on this game, make sure you understand the rules that have been adopted for the Betfair version of it. They may differ from the rules that you are used to (see Section 6.2).

6.2 The Rules Followed by Exchange Omaha Hi Poker

The Betfair game of Exchange Omaha Hi Poker is based on the version of Poker known as ‘Omaha High’. Its rules are exactly the same as those followed by Exchange Texas Hold’em Poker (described in Section 5.2) except that in Omaha Hi:

- The total number of cards dealt is 21 (as opposed to 13 in Texas Hold’em)
- The number of “hole” cards allocated to each hand is four (as opposed to two in Texas Hold’em)
- The winning hand must consist of exactly two “hole” cards plus exactly three Community Cards. (In Texas Hold’em, the winning hand can be formed from five Community Cards, or from one “hole” card plus four Community Cards, or from two “hole” cards plus three Community Cards.)

6.2.1 The Progress of a Game of Exchange Omaha Hi Poker

The game progresses in the same way as a game of Exchange Texas Hold’em (see Section 5.2.1).

6.2.2 The System for Ranking Hands in Exchange Omaha Hi Poker

The system of ranking hands in Exchange Omaha Hi is the same as for Exchange Texas Hold’em (see Section 5.2.2).

6.2.3 The Tie-breaker for Exchange Omaha Hi Poker

The tie-breaker rules for Exchange Omaha Hi are the same as for Exchange Texas Hold'em (see Section 5.2.3).

6.2.4 The Dead-heat Rule for Bet Settlement In Exchange Omaha Hi Poker

And finally, the dead-heat rule for Exchange Omaha Hi is the same as for Exchange Texas Hold'em (see Section 5.2.4).

6.2.5 The Two Betting Markets for Exchange Omaha Hi Poker

For Exchange Omaha Hi Poker, there are two betting markets.

The first is the "MainBets" market, which contains the betting selections for the automated players: "Hand 1", "Hand 2", "Hand 3", and "Hand 4". In this market only one selection can win: the others must by definition lose. If there is a dead heat between two of the automated players, then the odds received by the backers of their respective selections are each divided by the number of winners, and bet settlement is based upon the new odds. (For full details of the dead-heat rule, see Section 5.2.4.)

The second betting market is the "SideBets" market. In this market, also, only one betting selection can win. However, in the "SideBets" market no dead-heat is possible and you need to understand the strict definition that applies to each selection[9]:

- Winner has Full House or Better
  If you choose this selection, it means you think that the winning hand will be a Full House, Four of a Kind, a Straight Flush, or a Royal Flush.
- Winner has Flush
  If you choose this selection, it means you think that the winning hand will contain five cards of the same suit. However, if the winning hand is a Royal Flush or a Straight Flush, this selection will lose and the winning selection will be "Winner has Full House or Better".
- Winner has Straight
  If you choose this selection, it means you think that the winning hand will contain five cards of any suit in ascending order. However, if the winning hand is a Straight Flush, this selection will lose and the winning selection will be "Winner has Full House or Better".
- Winner has Three of a Kind
  If you choose this selection, it means you think that the winning hand will contain three cards of the same face-value. However, if the winning hand is a Full House, or if it contains Four of a Kind, this selection will lose and the winning selection will be "Winner has Full House or Better".
- Winner has Two Pair or Worse
  If you choose this selection, it means you think that the winning hand will contain a High Card, Pair, or Two Pair. However, if the winning hand contains Three of a Kind, this selection loses and the winning selection will be "Winner has Three of a Kind". Similarly, if the winning hand contains Four of a Kind, this selection loses and the winning selection will be "Winner has Full House or Better".

[9] The definition of the selections is determined by the ranking of Poker hands in Table 5 (page 56).

6.3 The Main XML Elements Used to Represent a Game of Exchange Omaha Hi Poker

The XML elements that are used to represent a game of Exchange Omaha Hi Poker are broadly the same as those used to represent a game of Exchange Texas Hold'em Poker. Therefore, much of this section refers you to information in the previous chapter describing the implementation of Exchange Texas Hold'em. However, the implementation of Exchange Omaha Hi differs from the implementation of Exchange Texas Hold'em in that it has two betting markets associated with it (whereas Exchange Texas Hold'em has only one).

6.3.1 The Automated Players and Other "object" Elements for Exchange Omaha Hi Poker

In Exchange Omaha Hi, the "object" elements represent, respectively, the four automated players and the community cards. For a description of these "object" elements, see Section 5.3.1.

6.3.2 The "marketType" Element for Omaha Hi Poker (marketType)

The "marketType" element for the Omaha Hi "MainBets" market is the same as that described for the single betting market associated with Exchange Texas Hold'em (see Section 5.3.2).

The "SideBets" market for Omaha Hi is also a win-only market, and its "marketType" element is therefore also "WIN_ONLY" (see Table 4 in Section 4.2).

A win-only market, by definition, has only one winning selection (all other selections being necessarily losing ones).

Bets backing the winner of a game or laying its losers are settled by the Exchange Games engine in favour of the bettor.

It is not possible for a dead heat to occur in the Omaha Hi "SideBets" instance of the "WIN_ONLY" market type. For information about the way in which the selections "trump" each other according to the Poker hand rankings (so that only one can win), see Section 6.2.5.

When only one hand can possibly win, the game ends (even if not all of the five Community Cards have been dealt).

6.3.3 The "selection" Element for Omaha Hi Poker (selection)

For Omaha Hi, there are two betting markets and therefore the "markets" element itself contains two elements called "market". Each of these "market" elements in turn contains a single "selections" element, one of type "MainBets" and one of type "SideBets", and each contains a sequence of "selection" elements.

In the "MainBets" betting market, each selection has:

- a name that corresponds to the name of one possible winner of the game, for example, "Hand 1", "Hand 2", and so on,
- a "selectionTradeActivity" resource for viewing all available prices for the selection (see Section 4.2.2),
- a "status" (which can be IN_PLAY, LOSER, TIED_DEAD_HEAT, or WINNER),
- a figure for the amount of the bet that has been matched (this is presented in the element called "amountMatched")
- and figures for the best back and lay prices.

In the "SideBets" betting market, each betting selection has:

- a name ("Winner has Full House or Better", "Winner has Flush", "Winner has Straight", "Winner has three of a Kind", and "Winner has Two Pair or Worse"),

- a “selectionTradeActivity” resource for viewing all available prices for the selection (see Section 4.2.2),
- a “status” (which can be IN_PLAY, LOSER, or WINNER),
- a figure for the amount of the bet that has been matched (this is presented in the element called “amountMatched”),
- and figures for the best back and lay prices.

6.3.4 Requesting XML Stating How Omaha Hi Poker is Represented (channelInfo)

Using the following URL, you can read a brief description (contained inside the “gameDescription” element) of the way that the Games API uses the generic “object” and “selection” elements to represent a game of Omaha Hi Poker and its betting market:

`https://api.games.betfair.com/rest/v1/channels/`*`channelId`*`/info?username=sampleuser`

where *channelID* is an integer value identifying the channel (this value is automatically included in the URLs returned for each channel in the “channelLanding” page, see Figure 11 above). The description provided is the same as that provided for Exchange Texas Hold’em Poker (see 5.3.4).

Chapter 7 Exchange Blackjack

This chapter describes the Betfair Exchange Games version of Blackjack and the XML that is used by the Exchange Games API to represent it. The chapter tells you:

- Which URLs to use to request the current state of a game of Exchange Blackjack,
- The rules adopted by the Betfair implementation of Blackjack,
- How the generic XML elements defined in the Betfair Exchange Games API schema are used to represent a game of Blackjack, and
- Which URL to use to request information about the betting time available after each round of a game.

7.1 The URLs for Accessing a Game of Exchange Blackjack

The URLs for accessing a game of Exchange Blackjack or Turbo Exchange Blackjack are presented in the “channelLanding” page, described in Section 4.1. They are also shown in Figure 32, and the element identifying each channel and game is printed in bold-type.

```
<channel gameType="BLACKJACK" id="1444077" name="Exchange BlackJack">
<resource href="https://api.games.betfair.com/rest/v1/channels/1444077/snapshot
?username=sampleuser" responseType="channelSnapshot" title="Channel Snapshot"/>

<resource href="https://api.games.betfair.com/rest/v1/channels/1444077/history
?username=sampleuser" responseType="channelHistory" title="Channel history"/>

<resource href="https://api.games.betfair.com/rest/v1/channels/1444077/info
?username=sampleuser" responseType="channelInfo" title="Channel Definition"/>
</channel>

<channel gameType="BLACKJACK" id="1444082" name="Turbo Exchange BlackJack">
<resource href="https://api.games.betfair.com/rest/v1/channels/1444082/snapshot
?username=ampleuser" responseType="channelSnapshot" title="Channel Snapshot"/>

<resource href="https://api.games.betfair.com/rest/v1/channels/1444082/history
?username=sampleuser" responseType="channelHistory" title="Channel history"/>

<resource href="https://api.games.betfair.com/rest/v1/channels/1444082/info
?username=sampleuser" responseType="channelInfo" title="Channel Definition"/>
</channel>
```

Figure 32 Excerpt from a “channelLanding” Element Showing all the “resource” Elements for Blackjack

NOTE: The Betfair Exchange Games implementation of Blackjack uses rules that differ in some respects from the conventional rules. Before you place any bets on this game, make sure you understand the rules adopted for the Betfair version of the game (see Section 7.2).

7.2 The Rules Followed by the Betfair Game of Exchange Blackjack

This section describes the progress of a game of Exchange Blackjack, the rankings of the different winning hands that are possible, the rules for deciding the winning hand in the event of a tie between hands, and the dead-heat rule that is used for bet settlement when there are several winners in the betting market.

7.2.1 The Progress of a Game of Exchange Blackjack

This game is based on the casino game of Blackjack but in Exchange Blackjack the following rules apply to make the game practicable in an automated online form:

- There is an infinite deck of cards
- No splitting, doubling or surrender is permitted to the automated players

- The “Dealer” must stand when it holds cards with a points value (that includes an ace counted as 11) of 17 or higher
- Five-card Charlie beats a Blackjack (see Section 7.2.3)

The game begins with 10 cards from an infinite deck[10] being shown face-down. Eight of these cards are then paired to form four separate hands, one hand each for “Player 1”, “Player 2”, “Player 3” and “Player 4”. The remaining two cards are for the "Dealer".

The first round of betting occurs with no cards face-up. During it, you can place bets on any individual player to beat the “Dealer” or on any of the following four “exotic” selections:

- “Dealer wins or ties all” (if you back this selection, then you think the “Dealer” will either beat or tie with all four player hands);
- “One or more has 5 cards” (if you back this selection, then you think one or more of the players or the “Dealer” will achieve a "Five-card Charlie"; a “Five-card Charlie” is when five cards in a single hand have a total points value of less than 22);
- “All Cards Unique” (if you back this selection, then you think that, at the end of the game, in the set of cards comprising all the players’ hands plus the “Dealer” object’s hand, each card will be individually unique in respect of both suit and value; for example, this selection would lose if one player and the “Dealer” both held a Three of Clubs).
- “Only one has 21 pts” (this means that you think only one player or only the “Dealer” will have a hand worth 21 points at the end of the game).

During the betting windows at the end of each round of the game, you can place bets to back or lay each player or “exotic” selection.

At the end of the first round of betting, unmatched bets are cancelled and nine cards are randomly dealt to each of the “player” objects (four pairs of two) and one to the “Dealer”. There is then another round of betting, again bets can back or lay each individual hand or “exotic” selection within the game.

NOTE: If a selection is “resolved” before the end of the game (in other words, if a situation arises in which a selection can no longer represent a possible outcome of the particular game), bets on it will be settled immediately and bets will no longer be accepted on it.

There is then a second round of betting. After the second round of betting the players draw further cards. If a hand is bust then it is suspended as a betting selection and all bets on or against it are immediately settled.

The third round of betting now begins unless all players have bust. If all the players have bust, the game finishes without the “Dealer” taking a second card. After the third round of betting, the second of the “Dealer” object’s cards is dealt.

Whenever the “Dealer” object’s cards yield a points total of “Soft 17” or greater (from a maximum of 5 cards), the “Dealer” must stand. (“Soft” means that an ace is counted as worth 11 points to make up the points total.)

If the “Dealer” stands or is bust, the game is over and all bets are immediately settled.

If the “Dealer” does not stand or bust, betting continues until:

- The “Dealer” *does* stand or bust, or
- All of the players’ hands have either won, lost, or tied in relation to the “Dealer”, or

[10] With an infinite deck, an infinite number of duplicate cards is possible.

- it is known for each of the four hands whether it is impossible for it to win, lose or tie in relation to the “Dealer”. For example, if, prior to the “Dealer” taking a third card, three of the player hands have bust, the remaining player has a Blackjack, and the “Dealer” has no possibility of achieving a “Five-card Charlie” (because the “Dealer” object’s two cards already have a total points value of 15 or 16 and the “Dealer” in Exchange Blackjack must stand at “Soft 17” or above), then the Exchange Games engine knows the “Dealer” cannot win. In this scenario, therefore, the player holding the Blackjack becomes the only winner and the game finishes without the “Dealer” taking a third card.

7.2.2 The Logic Followed by the Automated Players in Exchange Blackjack

Betfair has commissioned the development of a “perfect player strategy” for the automated players and the “Dealer” to follow in Exchange Blackjack. The strategy has been produced by a company called The Wizard of Odds (whose website is at `www.wizardofodds.com`) and it can be viewed on the Betfair Games website. The strategy is also represented by the table in Figure 33.

Figure 33 shows you what decision each player takes for each possible hand value it can hold (shown in the leftmost vertical column in the coloured table) and in response to each possible “up card” belonging to the “Dealer” (shown in the top row of the coloured table). The “up card” is the card belonging to the “Dealer” that is turned face-up, and whose value is therefore known.

As with standard blackjack, the “Dealer” will hit on all scores of 16 or below and stand on all scores of 17 or above including 'soft' 17 (A6)

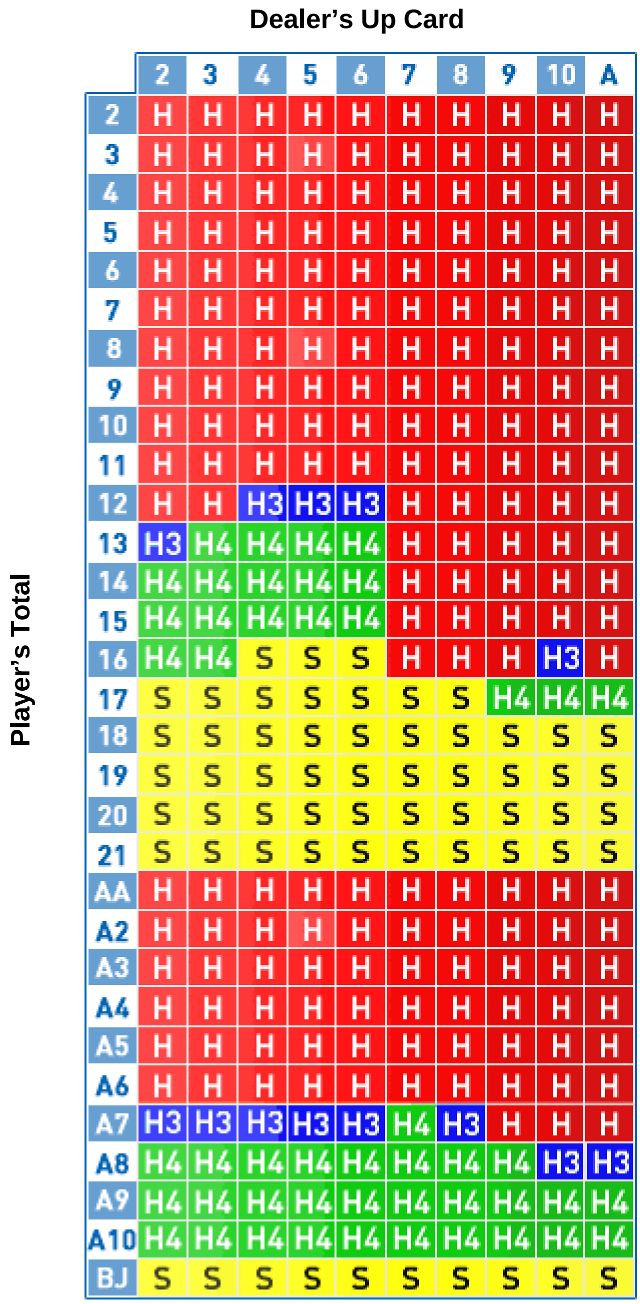
For the player's totals (in the leftmost column) the table (and the Exchange version of the game) treats a hand containing a 10 and a 6 (totalling 16) as equivalent to a hand containing a 6, a 5, a 3 and a 2 (also totalling 16).

“A7” in the strategy table means a hand containing an ace plus cards totalling 7 (for example, an ace and a 7, or an ace, plus a 2 and a 5).

When a hand has an ace plus further cards with a combined value of more than 10, the ace is always counted as 1 in the Exchange version of the game. So, for example, a hand containing an ace, a 6, and an 8 scores 15 and not A14.

Some sample look-ups in the Optimal Strategy Table in Figure 33 are:

- A player has a 10 and a 6 (totalling 16), and the card that the “Dealer” has face-up is 7: in this situation, the player’s action is “H” (player hits);
- A player has a 10, a 6 and a 3 (totalling 19), and the card that the “Dealer” has face-up is 7: the player’s action is “S” (player stands);
- A player has an ace and a 7 (totalling A7), and the card that the “Dealer” has face-up is 7: the player’s action is “H4” (the player stands because it has two cards; see the key in Figure 33);
- Player has an ace, a 2, a 3, and a 2 (totalling A7), and the card that the “Dealer” has face-up is 7: the player’s action is “H4” (the player hits because it has 4 cards; see the key in Figure 33).



Dealer's Up Card

Player's Total

	2	3	4	5	6	7	8	9	10	A
2	H	H	H	H	H	H	H	H	H	H
3	H	H	H	H	H	H	H	H	H	H
4	H	H	H	H	H	H	H	H	H	H
5	H	H	H	H	H	H	H	H	H	H
6	H	H	H	H	H	H	H	H	H	H
7	H	H	H	H	H	H	H	H	H	H
8	H	H	H	H	H	H	H	H	H	H
9	H	H	H	H	H	H	H	H	H	H
10	H	H	H	H	H	H	H	H	H	H
11	H	H	H	H	H	H	H	H	H	H
12	H	H	H3	H3	H3	H	H	H	H	H
13	H3	H4	H4	H4	H4	H	H	H	H	H
14	H4	H4	H4	H4	H4	H	H	H	H	H
15	H4	H4	H4	H4	H4	H	H	H	H	H
16	H4	H4	S	S	S	H	H	H	H3	H
17	S	S	S	S	S	S	S	H4	H4	H4
18	S	S	S	S	S	S	S	S	S	S
19	S	S	S	S	S	S	S	S	S	S
20	S	S	S	S	S	S	S	S	S	S
21	S	S	S	S	S	S	S	S	S	S
AA	H	H	H	H	H	H	H	H	H	H
A2	H	H	H	H	H	H	H	H	H	H
A3	H	H	H	H	H	H	H	H	H	H
A4	H	H	H	H	H	H	H	H	H	H
A5	H	H	H	H	H	H	H	H	H	H
A6	H	H	H	H	H	H	H	H	H	H
A7	H3	H3	H3	H3	H3	H4	H3	H	H	H
A8	H4	H4	H4	H4	H4	H4	H4	H4	H3	H3
A9	H4	H4	H4	H4	H4	H4	H4	H4	H4	H4
A10	H4	H4	H4	H4	H4	H4	H4	H4	H4	H4
BJ	S	S	S	S	S	S	S	S	S	S

Key to table:

H: Player always hits

H3: Player hits with 3 or more cards, otherwise the player stands

H4: Player hits with 4 or more cards, otherwise the player stands

S: Player stands

NOTE: Picture cards always count as 10

Figure 33 Blackjack Optimal Player Strategy Table (as displayed on the Betfair Games website)

7.2.3 The System for Ranking the Winning Hands in Exchange Blackjack

In Exchange Blackjack the cards are assigned points according to the following rules:

- Cards 2 to 10 each have a points value equal to their face value.
- Aces have a points value of 11 or 1, depending on the requirements of the “player” or “Dealer” object.
- Picture cards (that is, Jacks, Queens and Kings) each have a points value of 10.

The winning hands in Exchange Blackjack are ranked as follows:

1. 5 Card Charlie (this is achieved when a hand contains five cards whose total points value is less than 22)
2. Blackjack (this is achieved when a hand contains an ace plus a picture card or a card with a face value of 10)
3. Highest points value (this means that the hand with the highest points value wins)

7.2.4 The Dead-heat rule for Bet Settlement in Exchange Blackjack

The dead-heat rule applies to bets on Exchange Blackjack when the “Dealer” ties with one or more of the players. In these cases, the odds received by the backers of each selection are divided by two, and bet settlement then takes place based upon the new odds. (The odds for each selection are divided by two however many players the “Dealer” has tied with.)

For example, if Client A backs Hand 1 with £100 at odds of 4.0, and the outcome of the game is that Player 1 and Player 3 tie with the “Dealer”, then Client A’s bet will be re-calculated as £100 placed at odds of 2.0; Client A will therefore receive £100 in winnings.

NOTE: The application of the dead-heat rule makes it possible for you to receive back less than your original stake when the bets are settled after a game. For example, if Client A had backed Player 1 with £100 at odds of 1.2 and Player 1 and Player 3 had both tied with the “Dealer”, then the total payout for the bet would be:

£100 x (1.2/2) = £60

This would represent a net loss of £40, even though Client A’s selection (Player 1) did not lose against the “Dealer”.

When the dead-heat rule is applied, determining whether a particular bet is a net winner or a net loser depends on the number of selections involved in the dead heat and the odds at which the bet was matched.

7.3 The Main XML Elements Used to Represent a Game of Blackjack (channelSnapshot)

The XML elements (principally the “object” and “selection” elements) that are used to describe the games available via the Exchange Games API are generic; in other words, they are the same for each game.

Figure 35 shows an XML snapshot of a game of Blackjack. Just as with the Poker examples above, the two main elements are “object” (nested inside the element “game”) and “selection” (nested inside the element “market”). As stated previously, these elements are generic.

Figure 34 shows how the data returned in the “channelSnaphsot” element for Blackjack is represented on the main Betfair website.

7.3.1 The Automated Players and Other “object” Elements for Blackjack (object)

For Blackjack, the “object” elements represent, respectively, the four automated players and the dealer. Each object has:

- a name (“Player 1”, “Player 2”, “Player 3”, “Player 4”, or “Dealer”),
- a description (which, for each of the automated players and the dealer, will provide the total number of points represented by that player’s (or the dealer’s) current hand of cards, followed by the word “points” and, if applicable, a comment - such as “bust” - to gloss their significance),
- a status (which for the Betfair Blackjack game can be IN_PLAY, LOSER, STOOD, TIED, HIT, or WINNER; see Figure 79),
- And one or more property elements, each representing a card. These property elements each have two attributes: “name” and “value”. The “name” is of the form “`Card` *`number`*”, where *`number`* is an integer value greater than zero (in other words, “Card 1”, “Card 2”, and so on). The “value” of the card will be a number from 0 to 51, each value corresponding in a matrix to one member of the traditional deck of cards. The mappings that are used by the Games API between the numbers 0 to 51 and the members of the deck are shown in Figure 29. (They are also published by Betfair on their Games website: `http://stats.betfairgames.com/`.)

In Blackjack, the player objects each have between two and five cards at the end of the game.

7.3.2 The “marketType” Element for Blackjack (marketType)

The betting market for Exchange Blackjack is of the type “ANY_NUMBER_OF_WINNERS” (see Table 4 in Section 4.2).

In this type of market, wherever one selection wins there is no necessary implication that another loses (or that all other selections lose); the selections are independent of each other, and placing a bet on one selection will not affect your profit or loss on another.

In this type of market it is also possible for no selection to win.

A dead heat can occur in an Exchange Blackjack market when there is a tie between a Player and the Dealer. As with Exchange Poker, in the event of a dead heat the odds received by the backers of each are divided by two, and bet settlement is based upon the new odds (see Section 7.2.4).

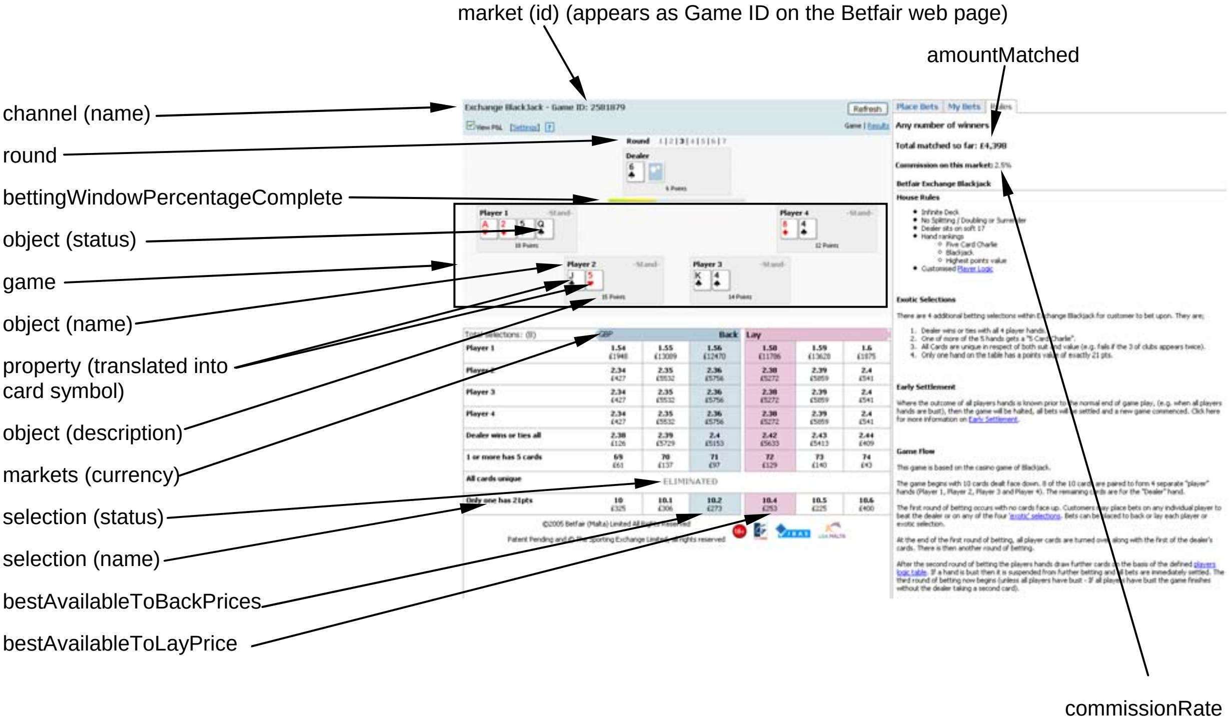
Finally, it is possible in an ANY_NUMBER_OF_WINNERS market for selections to be eliminated before the end of the game (for example, if a player is “bust”). Bets on selections that are eliminated before the end of the game are settled immediately by the Betfair Exchange Games engine.

7.3.3 The “selection” Element for Blackjack (selection)

For Blackjack, each betting selection has:

- A name that either corresponds to the name of one possible winner of the game (“Player 1”, “Player 2”, “Player 3”, “Player 4”) or that represents a possible outcome of the game (“Dealer wins or ties all”, “1 or more has 5 cards”, “All cards unique”, or “Only one has 21pts”),
- a “selectionTradeActivity” resource for viewing all available prices for the selection (see Section 4.2.2)
- A “status” (which can be IN_PLAY, LOSER, TIED_DEAD_HEAT, or WINNER; see Figure 80),
- A figure for the amount of the bet that has been matched (this is presented in the element called “amountMatched”)
- And figures for the best back and lay prices.

NOTE: In Figure 35 the dots indicate XML output for the “object” elements called “Player 3” and “Player 4” and for their corresponding betting “selection” elements. These elements are omitted from the sample output because their form is illustrated by the elements for “Player 1” and “Player 2”).



Exchange BlackJack - Game ID: 2581879

Total selections: (8)	GBP		Back	Lay		
Player 1	1.54 £1948	1.55 £13089	1.56 £12470	1.58 £11786	1.59 £13628	1.6 £1875
Player 2	2.34 £427	2.35 £5532	2.36 £5756	2.38 £5272	2.39 £5859	2.4 £541
Player 3	2.34 £427	2.35 £5532	2.36 £5756	2.38 £5272	2.39 £5859	2.4 £541
Player 4	2.34 £427	2.35 £5532	2.36 £5756	2.38 £5272	2.39 £5859	2.4 £541
Dealer wins or ties all	2.38 £126	2.39 £5729	2.4 £5153	2.42 £5633	2.43 £5413	2.44 £409
1 or more has 5 cards	69 £61	70 £137	71 £97	72 £129	73 £140	74 £43
All cards unique			ELIMINATED			
Only one has 21pts	10 £325	10.1 £306	10.2 £273	10.4 £253	10.5 £225	10.6 £400

©2005 Betfair (Malta) Limited All Rights Reserved

Patent Pending and © The Sporting Exchange Limited, all rights reserved

Any number of winners

Total matched so far: £4,398

Commission on this market: 2.5%

Betfair Exchange Blackjack

House Rules

- Infinite Deck
- No Splitting / Doubling or Surrender
- Dealer sits on soft 17
- Hand rankings
  - Five Card Charlie
  - Blackjack
  - Highest points value
- Customised Player Logic

Exotic Selections

There are 4 additional betting selections within Exchange Blackjack for customer to bet upon. They are;

1. Dealer wins or ties with all 4 player hands.
2. One of more of the 5 hands gets a "5 Card Charlie".
3. All Cards are unique in respect of both suit and value (e.g. fails if the 3 of clubs appears twice).
4. Only one hand on the table has a points value of exactly 21 pts.

Early Settlement

Where the outcome of all players hands is known prior to the normal end of game play, (e.g. when all players hands are bust), then the game will be halted, all bets will be settled and a new game commenced. Click here for more information on Early Settlement.

Game Flow

This game is based on the casino game of Blackjack.

The game begins with 10 cards dealt face down. 8 of the 10 cards are paired to form 4 separate "player" hands (Player 1, Player 2, Player 3 and Player 4). The remaining cards are for the "Dealer" hand.

The first round of betting occurs with no cards face up. Customers may place bets on any individual player to beat the dealer or on any of the four 'exotic' selections. Bets can be placed to back or lay each player or exotic selection.

At the end of the first round of betting, all player cards are turned over along with the first of the dealer's cards. There is then another round of betting.

After the second round of betting the players hands draw further cards on the basis of the defined players logic table. If a hand is bust then it is suspended from further betting and all bets are immediately settled. The third round of betting now begins (unless all players have bust - If all players have bust the game finishes without the dealer taking a second card).

Figure *34* How a “channelSnapshot” for Blackjack is Represented on the Betfair Games Website

```
<channelSnapshot xmlns="urn:betfair:games:api:v1">
   <channel gameType="BLACKJACK" id="1444077" name="Exchange BlackJack">
      <status>RUNNING</status>
      <game id="190675">
        <round>1</round>
        <bettingWindowTime>30</bettingWindowTime>
        <bettingWindowPercentageComplete>100</bettingWindowPercentageComplete>
          <gameData>
            <object name="Player 1">
              <description/>
              <status>IN_PLAY</status>
              <property name="Card 1" value="NOT AVAILABLE"/>
              <property name="Card 2" value="NOT AVAILABLE"/>
            </object>
            <object name="Player 2">
              <description/>
              <status>IN_PLAY</status>
              <property name="Card 1" value="NOT AVAILABLE"/>
              <property name="Card 2" value="NOT AVAILABLE"/>
            </object>
            ...
            <object name="Dealer">
              <description/>
              <status>IN_PLAY</status>
              <property name="Card 1" value="NOT AVAILABLE"/>
              <property name="Card 2" value="NOT AVAILABLE"/>
            </object>
          </gameData>
        <markets currency="GBP">
            <market id="2908368" nextId="2908395">
              <status>ACTIVE</status>
              <commissionRate>2.5</commissionRate>
              <marketType>ANY_NUMBER_OF_WINNERS</marketType>
              <selections type="MainBets">
                 <selection id="658442">
                    <name>Player 1</name>
                    <resource href="https://api.betfair.com/rest/v1/selection/tradeActivity
                     ?username=sampleuser&marketId=2908368&selectionId=658442"
                     responseType="selectionTradeActivity" title="Trade Activity"/>
                    <status>IN_PLAY</status>
                    <amountMatched>0.0</amountMatched>
                    <bestAvailableToBackPrices/>
                    <bestAvailableToLayPrices/>
                 </selection>
                 <selection id="658443">
                    <name>Player 2</name>
                     <resource href="https://api.betfair.com/rest/v1/selection/tradeActivity
                      ?username=sampleuser&marketId=2908368&selectionId=658443"
                      responseType="selectionTradeActivity" title="Trade Activity"/>
                    <status>IN_PLAY</status>
                    <amountMatched>0.0</amountMatched>
                    <bestAvailableToBackPrices/>
                    <bestAvailableToLayPrices/>
                    </selection>
                 ...
                 <selection id="658444">
                    <name>Dealer wins or ties all</name>
                    <resource href="https://api.betfair.com/rest/v1/selection/tradeActivity
                     ?username=sampleuser&marketId=2908368&selectionId=658444"
                     responseType="selectionTradeActivity" title="Trade Activity"/>
                    <status>IN_PLAY</status>
                    <amountMatched>0.0</amountMatched>
                    <bestAvailableToBackPrices/>
                    <bestAvailableToLayPrices/>
                 </selection>
                 <selection id="658445">
                    <name>1 or more has 5 cards</name>
                     <resource href="https://api.betfair.com/rest/v1/selection/tradeActivity
                      ?username=sampleuser&marketId=2908368&selectionId=658445"
                     responseType="selectionTradeActivity" title="Trade Activity"/>
                    <status>IN_PLAY</status>
                    <amountMatched>0.0</amountMatched>
                    <bestAvailableToBackPrices/>
                    <bestAvailableToLayPrices/>
                 </selection>
                 <selection id="658446">
                     <name>All cards unique</name>
                      <resource href="https://api.betfair.com/rest/v1/selection/tradeActivity
                      ?username=sampleuser&marketId=2908368&selectionId=658446"
                     responseType="selectionTradeActivity" title="Trade Activity"/>
                     <status>IN_PLAY</status>
                     <amountMatched>0.0</amountMatched>
                     <bestAvailableToBackPrices/>
                     <bestAvailableToLayPrices/>
                 </selection>
                 <selection id="658447">
                     <name>Only one has 21pts</name>
                      <resource href="https://api.betfair.com/rest/v1/selection/tradeActivity
                       ?username=sampleuser&marketId=2908368&selectionId=658447"
                       responseType="selectionTradeActivity" title="Trade Activity"/>
                     <status>IN_PLAY</status>
                     <amountMatched>0.0</amountMatched>
                     <bestAvailableToBackPrices/>
                     <bestAvailableToLayPrices/>
                 </selection>
              </selections>
            </market>
         </markets>
      </game>
   </channel>
</channelSnapshot>
```

Figure *35* Sample “channelSnapshot” Describing a Game of Blackjack (dots indicate omitted data)

7.3.4 Requesting XML Stating How Blackjack is Represented (channelInfo)

Using the following URL, you can read a brief description of the way that the Games API uses the generic elements “object” and “selection” to represent a game of Blackjack and its betting market:

https://api.games.betfair.com/rest/v1/channels/channelID/info?username=sampleuser

where *channelID* is an integer value identifying the channel being used for Blackjack (this number is automatically included in the URLs returned for each “channel” in the “channelLanding” page, see Figure 11 above). Figure 36 shows some sample output for the “channelInfo” element for a game of Blackjack. For further information about the “channelInfo” element refer to Section 4.4.

```
<channelInfo xmlns="urn:betfair:games:api:v1">
  <round number="1">
    <channelInfo>
       <round number="1">
         <bettingWindow>30</bettingWindow>
         <playingWindow>5</playingWindow>
       </round>
       <round number="2">
         <bettingWindow>30</bettingWindow>
         <playingWindow>20</playingWindow>
       </round>
       <round number="3">
         <bettingWindow>30</bettingWindow>
         <playingWindow>5</playingWindow>
       </round>
       <round number="4">
         <bettingWindow>30</bettingWindow>
       <playingWindow>5</playingWindow>
       </round>
       <round number="5">
         <bettingWindow>30</bettingWindow>
         <playingWindow>5</playingWindow>
       </round>
       <round number="6">
         <bettingWindow>30</bettingWindow>
         <playingWindow>5</playingWindow>
       </round>
       <round number="7">
         <bettingWindow>0</bettingWindow>
         <playingWindow>5</playingWindow>
       </round>
       <gameRules>NOT AVAILABLE</gameRules>
         <gameDescription>
          Object names will be presented as "Player <id>", where <id> is a number
          greater than 0.

         "Player <id>" will also be the name of a corresponding selection in the betting
          market.

          The dealer's cards will be presented as belonging to an object called "Dealer".

          All card names will be presented as "Card <id>", where <id> is a number
          greater than 0. The value of each card will be presented as a number (between 0 and
          51) which maps to a single card in the traditional deck according to the following
          matrix.

          Values Returned for 'Card' Objects:

                   | Ace |  2 |  3 |  4 |  5 |  6 |  7 |  8 |  9 | 10 |  J |  Q | K  |
          ---------+-----+----+----+----+----+----+----+----+----+----+----+----+----+
          clubs    |  0  |  1 |  2 |  3 |  4 |  5 |  6 |  7 |  8 |  9 | 10 | 11 | 12 |
          diamonds | 13  | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |
          hearts   | 26  | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 |
          spades   | 39  | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 |

          When a card has not been turned over/dealt, the value "NOT AVAILABLE" is returned.
         </gameDescription>
    </channelInfo>
```

Figure *36* The “gameDescription” Element (inside the “channelInfo” element) for Blackjack

Chapter 8 Exchange Baccarat

This chapter describes the Betfair Exchange Games version of Baccarat and the XML that the Exchange Games API uses to represent it. The chapter tells you:

- Which URLs to use to request the current state of a game of Exchange Baccarat,
- The rules adopted by the Betfair implementation of Baccarat,
- How the generic XML elements defined in the Betfair Exchange Games API schema are used specifically to represent Baccarat,
- Which URL to use to request information about the betting time available after each round of a game.

8.1 The URLs for Accessing a Game of Baccarat

The URLs for accessing a game of Exchange Baccarat are presented in the “channelLanding” page, which is described in Section 4.1. These URLs are shown in Figure 37.

```
<channel gameType="BACCARAT" id="1444086" name="Exchange Baccarat">
<resource href="https://api.games.betfair.com/rest/v1/channels/1444086/snapshot
?username=sampleuser" responseType="channelSnapshot" title="Channel Snapshot"/>

<resource href="https://api.games.betfair.com/rest/v1/channels/1444086/snapshot
?username=sampleuser&amp;selectionsType=Mainbets" responseType="channelSnapshot"
title="Channel Snapshot"/>

<resource href="https://api.games.betfair.com/rest/v1/channels/1444086/snapshot
?username=sampleuser&amp;selectionsType=Sidebets" responseType="channelSnapshot"
title="Channel Snapshot"/>

<resource href="https://api.games.betfair.com/rest/v1/channels/1444086/history
?username=sampleuser" responseType="channelHistory" title="Channel history"/>

<resource href="https://api.games.betfair.com/rest/v1/channels/1444086/info
?username=sampleuser" responseType="channelInfo" title="Channel Definition"/>
</channel>
</channel>
```

Figure 37 Excerpt from a “channelLanding” Element Showing “resource” Elements for Baccarat

NOTE: The Betfair Exchange Games implementation of Baccarat uses rules that differ in some respects from the conventional rules. Before you place any bets on this game, make sure you understand the rules adopted for the Betfair version of the game (see Section 8.2).

8.2 The Rules Followed by the Betfair Game of Exchange Baccarat

This game is Betfair’s version of the popular casino game Baccarat. This section describes the progress of a game and explains the ranking of winning hands. It also explains how bet settlement takes place in the event of a tie. And finally it introduces the two separate betting markets associated with this game.

8.2.1 The Progress of a Game of Exchange Baccarat

Exchange Baccarat is played with eight decks of cards and it applies a decision logic that conforms to standard Baccarat drawing rules. For more information about the drawing rules, see below in this section and refer to the Help available for Baccarat on the Betfair Games website (http://api.games.betfair.com).

Exchange Baccarat uses a shoe containing eight decks. The shoe persists over multiple games. It is reset either:

- When there are fewer than seven cards remaining in it before the start of a new game, or

- When the cards remaining in it yield a guaranteed outcome for either the “Player” or “Banker” or both Exchange Baccarat starts with one round of betting before any cards have been dealt to either the “Player” or the “Banker” object. At the end of this first round of betting, the “Player” and “Banker” are both dealt a card, and the second round of betting commences.

At the end of the second round of betting, the “Player” is dealt a second card, and the third round of betting commences.

At the end of the third round of betting, the “Banker” is dealt a second card, and the game will end at this point if neither the “Player” nor the “Banker” takes a third card.

The decisions of the “Player” and “Banker” at this stage of the game depend upon the points value of the hands they each hold. The logic applied by the Exchange Games engine to determine whether a third card will be dealt is as follows:

1. If either the “Player” or the “Banker” has a score of 8 or 9, then *both* of them stand. This rule overrides all others.
2. If the “Player” has a score of 6 or 7, then the “Player” stands.
3. If the “Player” stands, and the “Banker” has a score of 5 or less, the “Banker” hits.
4. If the “Player” has a score of 5 or less, the “Player” hits.

If the result of this decision logic is that either the “Player” or the “Banker” is to take a third card, then another round of betting commences *before* that card is dealt.

At the end of the game, the scores of the “Banker” and “Player” objects are compared, and the winner is the one with the highest score.

When a tie occurs between the Player and Banker in the game, the bets that backed the TIE selection win. But all back and lay bets on the other two selections (Banker and Player) are settled at odds of 1 (in other words, all backing stakes on them are returned and all lay liabilities associated with them are cleared).

8.2.2 The System for Ranking the Hands in Exchange Baccarat

Each card in Exchange Baccarat has a points value: picture cards are worth 10, an ace is worth 1, and all other cards are worth their face-value. The card suit is of no significance in the determination by the Exchange Games engine of the points value of a card.

The Exchange Games engine works out the score of a Baccarat hand by adding the points value of all cards in it, and, if the total is greater than 9, subtracting 10. If the score is still greater than 9, it continues to subtract 10 until the score is between 0 and 9. Sample scores are:

- Ten, Three, Five (subtract 10): this yields a score of 8
- Ten, Jack, Queen (subtract 10 three times): this yields a score of 0
- Five, Ace, Ace (no subtraction necessary): this yields a score of 7
- King, Eight (subtract 10): this yields a Natural 8 (a score of 8 resulting from two cards)
- Five, Four (no subtraction necessary): this yields a Natural 9 (a score of 9 resulting from two cards)
- Ten, Ten, Five (subtract 10 twice): this yields a score of 5
- King, Ace, Eight (subtract 10): this yields a score of 9

The best score in Baccarat is "Natural 9", which is a points total of 9 yielded by a hand of only two cards. The next best score is "Natural 8", which is a points total of 8 yielded by a hand of only two cards.

Table 6 Hand Rankings in Baccarat

Rank	Hand
1	Natural 9
2	Natural 8
3	9
4	8
5	7
6	6
7	5
8	4
9	3
10	2
11	1
12	0

8.2.3 The Settlement of Bets in the Event of a Tie in Exchange Baccarat

In the event of a tie between the “Banker” and “Player” at the end of the game, the stakes for all back bets on the “Banker” and “Player” are returned (in other words, these bets are settled at odds of 1.0).

8.2.4 The Two Betting Markets for Exchange Baccarat

For Exchange Baccarat, there are two betting markets. The first is the “MainBets” market, which contains the betting selections for the “Player”, the “Banker” or for a “Tie”.

In this market only one selection can win (see Section 8.3.2): the other two selections must by definition lose. In the event of a tie, therefore, between the “Banker” and “Player”, the “Tie” selection is treated as the winning selection and bets on a tie as the outcome of the game are settled accordingly.

NOTE: In the event of a tie, the stakes for bets placed on or against the “Player” and “Banker” selections are simply returned (they are settled at odds of 1.0).

The second betting market is the “SideBets” market. In this market any number of the selections can win (see Section 8.3.3). The selections in the “SideBets” market for Baccarat (and their meanings) are as follows:

- Pocket pair in either hand (if you choose this selection, it means you think that the first two cards dealt to either hand will be a pair)
- Natural Win (if you choose this selection, it means you think that the winning hand will be a “Natural 8” or a “Natural 9”)
- Natural Tie (if you choose this selection, it means you think that the game result in a tie between the “Player“ and the “Banker” and that both will have a “Natural 9” or a “Natural 8”)
- Highest hand is a 9 (if you choose this selection, it means you think that the highest scoring hand or hands will have a score of 9)
- Highest hand is odd (if you choose this selection, it means you think that the points value for the highest scoring hand or hands will be an odd number: 1, 3, 5, 7 or 9)

8.3 The Main XML Elements Used to Represent a Game of Baccarat

Figure 38 shows XML excerpted from the “channelSnapshot” element of a game of Baccarat. Just as with the Poker and Blackjack examples in earlier chapters, the two main structures are called “object” (nested inside the element “gameData”) and “selection” (nested inside the element “market”). As stated previously, these elements are generic for all games.

```
<gameData>
  <object name="Player">
    <description>1 point</description>
    <status>IN_PLAY</status>
    <property name="Card 1" value="26"/>
  </object>
  <object name="Banker">
    <description>3 points</description>
    <status>IN_PLAY</status>
    <property name="Card 1" value="28"/>
  </object>
  <object name="Shoe">
    <description>Baccarat Shoe Information</description>
    <status>N/A</status>
    <property name="cardsDealt" value="214"/>
    <property name="cardsRemaining" value="202"/>
    <property name="Card 1" value="13"/>
    <property name="Card 2" value="19"/>
    <property name="Card 3" value="15"/>
    <property name="Card 4" value="23"/>
    <property name="Card 5" value="15"/>
    <property name="Card 6" value="13"/>
    <property name="Card 7" value="16"/>
    <property name="Card 8" value="16"/>
    <property name="Card 9" value="15"/>
    <property name="Card 10" value="19"/>
    <property name="Card 11" value="15"/>
    <property name="Card 12" value="14"/>
    <property name="Card 13" value="9"/>
  </object>
  <object name="Scorecard">
     <description>Baccarat Scorecard Information</description>
     <status>N/A</status>
     <property name="Content"
      value="BN8TTTP962B798P96B6N8P3B6PN87N9BN8TN86N8PN8N8T3B7P7N8B9P6B47T7PN9BN8TTP5"/>
  </object>
</gameData>
```

Figure *38* Excerpt from a “channelSnapshot” Showing the “object” Elements for a Game of Baccarat

8.3.1 The Automated Players and Other “object” Elements for Baccarat (object)

For Baccarat, the “object” elements represent, respectively, the player, the banker, the shoe, and the scorecard. Each object has:

- a name (“Player”, “Banker”, “Shoe”, or “Scorecard”),
- a description (which, for the “Player” and “Banker”, will provide the total number of points represented by that object’s current hand of cards, and which for the “Shoe” and “Scorecard” simply says, respectively, “Baccarat shoe information” and “Baccarat scorecard information”),
- a status (which for the Betfair Baccarat game can be IN_PLAY, WINNER, LOSER, HIT, TIED, or STOOD),
- and one or more “property” elements.

For the “Player” and “Banker” objects in a “channelSnapshot”, the “property” elements represent the cards that each holds (the “property” elements for the “Player” and “Banker” are the same as for the automated players in Poker and Blackjack).

For the “Shoe”, however the property elements are different. There are fifteen of them. (This is not true of the “Shoe” element when viewed in the XML returned for a “channelHistory” request. In that case, the “Shoe” object contains just an “id” property element, whose value is an integer.)

The fifteen properties of the “Shoe” in a “channelSnapshot” represent:

- The number of cards dealt so far from the “Shoe” (the name of this property is “cardsDealt”, and its value is the number of cards dealt so far from the “Shoe”);
- The number of cards remaining to be dealt from the “Shoe” (the name of this property is “cardsRemaining”, and its value is the number of cards remaining to be dealt);
- The number of cards of each possible face-value that remain in the “Shoe”. These “property” elements each have the name “Card *number*“, where *number* is an integer value from 1 through 13 that corresponds to one of the thirteen possible face-values, irrespective of suit, in a deck of cards (from Ace through to King). In each case, the data in the “property” element’s value field is an integer representing the number of cards of each possible face-value that remain in the “Shoe” (so, for example, if the value of the property called “Card 1” were “7”, then there would be seven Aces remaining in the “Shoe”; if the value of the property called “Card 11” were “5”, there would be five Jacks remaining in the “Shoe”).

  There is one other aspect to note about the “Shoe”: its content persists across multiple games of Baccarat, and it is only reset (that is, filled with a new set of eight decks of cards) when there are fewer than seven cards in it (which is too few for a new game to start), or when a new game is about to start with either the “Player” or the “Banker” guaranteed to win.

Finally, the “Scorecard” object has only a single “property” element; this “property” reports the sequence of winning hands so far in the game. The “property”’ element’s name field contains the text “Content”, and its value field contains a long string of letters and numbers. When the “Banker” wins, a “B” appears in the string followed by a number representing the value of the winning hand. If the “Banker” wins again, only the value of the winning hand appears (the “B” being understood to apply to all the winning hand values reported until the next occurrence of a “P” indicates that the “Player” has won; the “P” is then similarly understood to apply to all the winning hand values reported until the next time the “Banker” wins). A Natural 8[11] is represented in the string by the two characters “N8”, a Natural 9 is represented by the two characters “N9”,and a tie is represented by the letter “T”.

In the example in Figure 38 the start of the string is: “`BN8TTTP962B7`”. This means that the “Banker” won the first game with a hand whose value was Natural 8; there followed three ties after which the “Player” won with a hand whose value was 9; the “Player” then won two games with hands whose values were 6 and 2, respectively; and the “Banker” then won again with a hand whose value was 7.

8.3.2 The “marketType” Element for the Baccarat “MainBets” Market (marketType)

The “MainBets” market for Baccarat is of type “SINGLE_WINNER_OR_TIE” (see Table 4 in Section 4.2). This type of market is like the WIN_ONLY market associated with Exchange Poker, but it differs from it by allowing no possibility of a dead heat between selections. Instead, the possible outcome of the Baccarat Player and Banker tying in the game is itself a “selection” in the market.

[11] A Natural 8 or 9 occurs when the sum of the face values of the *first two* cards in a hand is 8 or 9.

When a tie occurs between the Player and Banker in the game, then, in the SINGLE_WINNER_OR_TIE market, the bets that backed the TIE selection win. But all back and lay bets on the other two selections (Banker and Player) are settled at odds of 1 (in other words, all backing stakes on them are returned and all lay liabilities associated with them are cleared).

The nature of a SINGLE_WINNER_OR_TIE market means that the destiny of some selections (specifically, the Banker and Player selections in Baccarat) can become known before the end of the game.

If it becomes impossible for the Banker or Player to win in a game of Baccarat, betting on both the Banker and Player is suspended. But since it cannot yet be determined whether or not the Banker and Player will tie, bets on the Banker and Player are not settled until the end of the game. When this situation arises, the market “selection” elements representing the automated Banker and Player are given a “status” value of “CANNOT_WIN” or “CANNOT_LOSE”, as applicable.

Table 7 The "selection" Elements in Baccarat and their Possible Statuses

Status	Can this status apply to the Banker selection?	Can this status apply to the Player selection?	Can this status apply to the “Tie” selection?	Can bets be placed on a selection that has this status?	Have bets on the selection that has this status been settled?
IN_PLAY	Y	Y	Y	Y	N
CANNOT_WIN	Y	Y	N	N	N
CANNOT_LOSE	Y	Y	N	N	N
WINNER	Y	Y	Y	N	Y
LOSER	Y	Y	Y	N	Y
REFUND	Y	Y	N	N	Y

8.3.3 The “marketType” Element for the Baccarat “SideBets” Market (marketType)

The second market in Exchange Baccarat – whose “selectionsType” is “SideBets” – has a “marketType” of ANY_NUMBER_OF_WINNERS. This market behaves in the same way as the Exchange Blackjack market.

In this type of market, wherever one selection wins there is no necessary implication that another loses (or that all other selections lose). The selections are independent of each other. So placing a bet on one selection will not affect your profit or loss on another selection.

Also, in this type of market it is possible for no selection to win.

Finally, it is possible in an ANY_NUMBER_OF_WINNERS market for selections to be eliminated before the end of the game (for example, if no pocket pairs have been dealt). Bets on selections that are eliminated in this way are settled immediately by the Betfair Exchange Games engine.

8.3.4 The “selection” Element for Baccarat (selection)

For Baccarat, there are two betting markets and therefore the “markets” element itself contains two elements called “market”. Each of these “market” elements in turn contains a single “selections” element, one of type “MainBets” and one of type “SideBets”, and each contains a sequence of “selection” elements.

In the “MainBets” betting market, each selection has:

- a name (which can be “Player” , “Banker” or “Tie”),
- a “selectionTradeActivity” resource for viewing all available prices for the selection (see Section 4.2.2),

- a “status” (which can be IN_PLAY, LOSER, WINNER, REFUND, CANNOT_WIN, or CANNOT_LOSE),
- a figure for the amount of the bet that has been matched (this is presented in the element called “amountMatched”)
- and figures for the best back and lay prices.

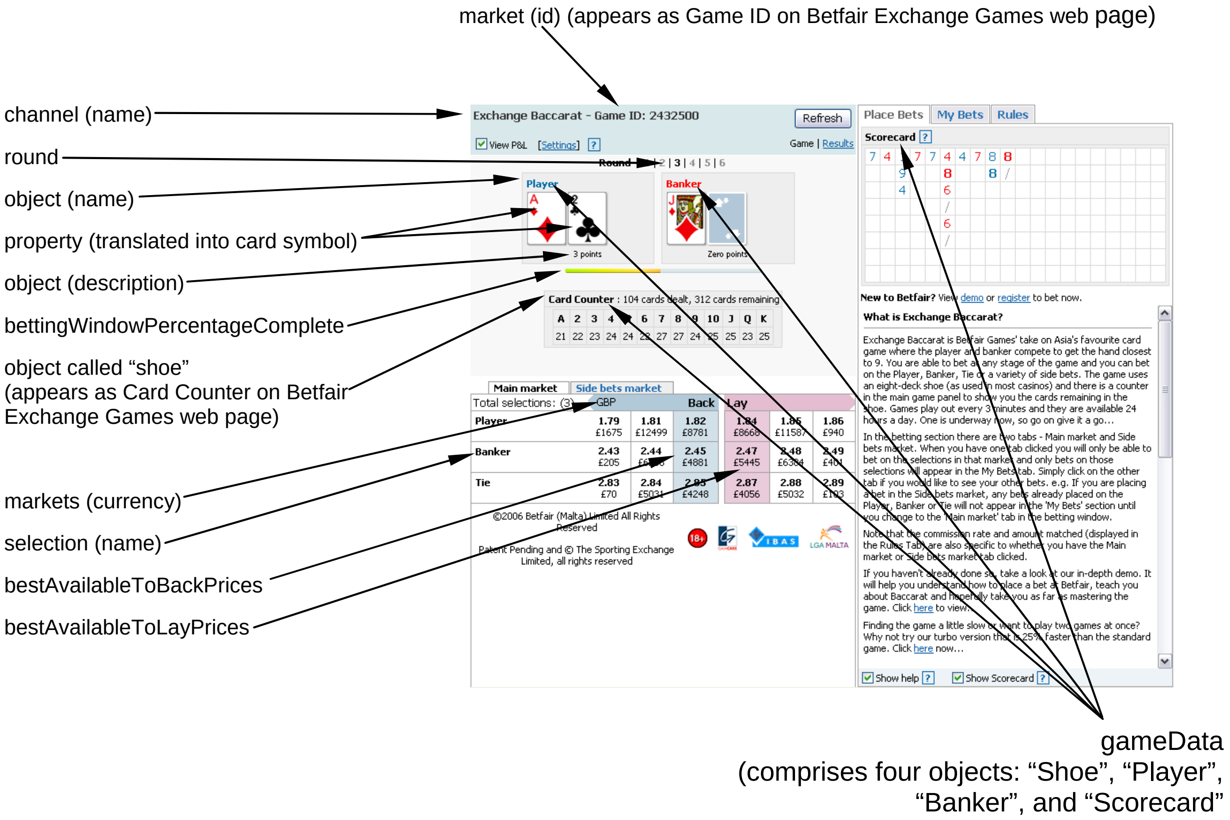
In the “SideBets” betting market, each betting selection has a:

- a name (which can be “Pocket Pair in Any Hand”, “Highest Hand Has Nine”, “Highest Hand is Odd”, “Natural Win”, or “Natural Tie”),
- a “selectionTradeActivity” resource for viewing all available prices for the selection (see Section 4.2.2)
- a “status” (which can be IN_PLAY, LOSER, WINNER, REFUND, CANNOT_WIN, or CANNOT_LOSE),
- a figure for the amount of the bet that has been matched (this is presented in the element called “amountMatched”)
- and figures for the best back and lay prices.

Figure 39 shows how data in a “channelSnapshot” for Baccarat is represented by the Betfair Games website. Figure 40 shows the same information except with the Rules tab clicked on the website, and this displays the data corresponding to the “amountMatched” and “commissionRate” elements in the Exchange Games API.

Figure 41 shows the “markets” element (excerpted from a “channelSnapshot”) for a game of Baccarat. It lists the orthodox (“MainBets”) betting selections and the extra selections (called “SideBets”).

NOTE: The “bestAvailableToBackPrices” and the “bestAvailableToLayPrices” elements in Figure 41 are shown empty for reasons of space. For examples of a “channelSnapshot” in which these elements are populated, see Figure 12.



market (id) (appears as Game ID on Betfair Exchange Games web page)

channel (name)

round

object (name)

property (translated into card symbol)

object (description)

bettingWindowPercentageComplete

object called “shoe”
(appears as Card Counter on Betfair Exchange Games web page)

markets (currency)

selection (name)

bestAvailableToBackPrices

bestAvailableToLayPrices

gameData
(comprises four objects: “Shoe”, “Player”, “Banker”, and “Scorecard”

Exchange Baccarat - Game ID: 2432500 Refresh

☑ View P&L [Settings] [?] Game | Results

Round | 2 | 3 | 4 | 5 | 6

Player
3 points

Banker
Zero points

Card Counter : 104 cards dealt, 312 cards remaining

A	2	3	4	[illegible]	6	7	8	9	10	J	Q	K
21	22	23	24	24	22	27	27	24	25	25	23	25

Main market | Side bets market

Total selections: (3)	GBP		Back	Lay		
Player	1.79 £1675	1.81 £12499	1.82 £8781	1.84 £8668	1.85 £11587	1.86 £940
Banker	2.43 £205	2.44 [illegible]	2.45 £4881	2.47 £5445	2.48 £6384	2.49 £401
Tie	2.83 £70	2.84 £5031	2.85 £4248	2.87 £4056	2.88 £5032	2.89 £103

©2006 Betfair (Malta) Limited All Rights Reserved

Patent Pending and © The Sporting Exchange Limited, all rights reserved

Place Bets | My Bets | Rules

Scorecard [?]

7	4		7	7	4	4	7	8	8
		9			8			8	/
		4			6				
					/				
					6				
					/				

New to Betfair? View demo or register to bet now.

What is Exchange Baccarat?

Exchange Baccarat is Betfair Games' take on Asia's favourite card game where the player and banker compete to get the hand closest to 9. You are able to bet at any stage of the game and you can bet on the Player, Banker, Tie or a variety of side bets. The game uses an eight-deck shoe (as used in most casinos) and there is a counter in the main game panel to show you the cards remaining in the shoe. Games play out every 3 minutes and they are available 24 hours a day. One is underway now, so go on give it a go…

In the betting section there are two tabs - Main market and Side bets market. When you have one tab clicked you will only be able to bet on the selections in that market and only bets on those selections will appear in the My Bets tab. Simply click on the other tab if you would like to see your other bets. e.g. If you are placing a bet in the Side bets market, any bets already placed on the Player, Banker or Tie will not appear in the 'My Bets' section until you change to the 'Main market' tab in the betting window.

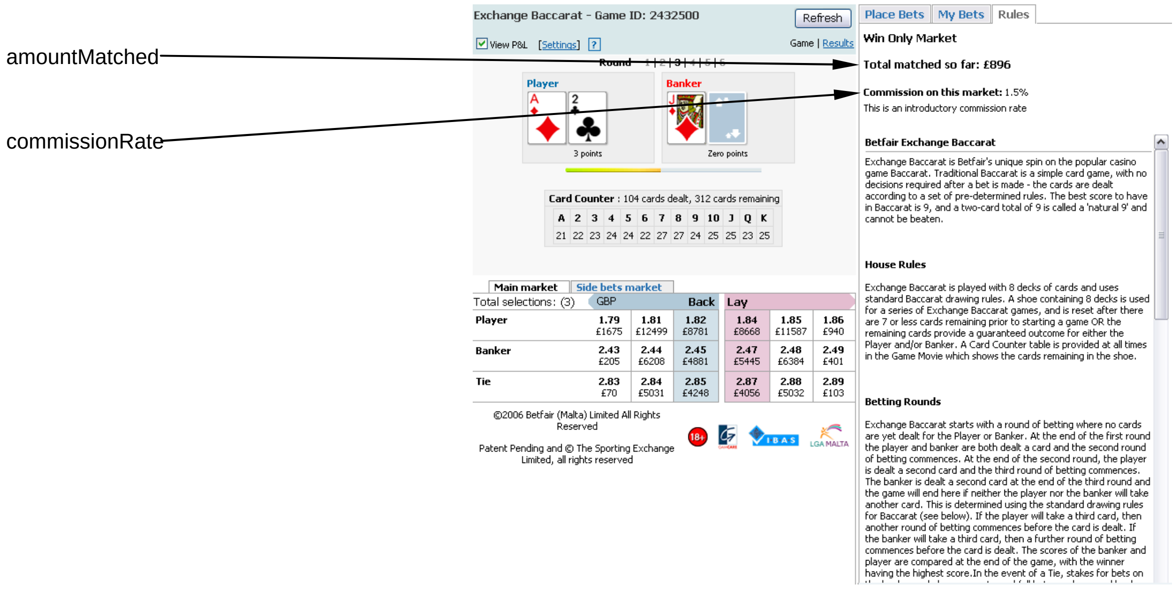
Note that the commission rate and amount matched (displayed in the Rules Tab) are also specific to whether you have the Main market or Side bets market tab clicked.

If you haven't already done so, take a look at our in-depth demo. It will help you understand how to place a bet at Betfair, teach you about Baccarat and hopefully take you as far as mastering the game. Click here to view…

Finding the game a little slow or want to play two games at once? Why not try our turbo version that is 25% faster than the standard game. Click here now…

☑ Show help [?] ☑ Show Scorecard [?]

Figure *39* How the Data Structures for Baccarat Appear on the Betfair Exchange Games Website



Exchange Baccarat - Game ID: 2432500

Card Counter : 104 cards dealt, 312 cards remaining

A	2	3	4	5	6	7	8	9	10	J	Q	K
21	22	23	24	24	22	27	27	24	25	25	23	25

Total selections: (3)	GBP		Back	Lay		
Player	**1.79** £1675	**1.81** £12499	**1.82** £8781	**1.84** £8668	**1.85** £11587	**1.86** £940
Banker	**2.43** £205	**2.44** £6208	**2.45** £4881	**2.47** £5445	**2.48** £6384	**2.49** £401
Tie	**2.83** £70	**2.84** £5031	**2.85** £4248	**2.87** £4056	**2.88** £5032	**2.89** £103

©2006 Betfair (Malta) Limited All Rights Reserved

Patent Pending and © The Sporting Exchange Limited, all rights reserved

Win Only Market

Total matched so far: £896

Commission on this market: 1.5%
This is an introductory commission rate

Betfair Exchange Baccarat

Exchange Baccarat is Betfair's unique spin on the popular casino game Baccarat. Traditional Baccarat is a simple card game, with no decisions required after a bet is made - the cards are dealt according to a set of pre-determined rules. The best score to have in Baccarat is 9, and a two-card total of 9 is called a 'natural 9' and cannot be beaten.

House Rules

Exchange Baccarat is played with 8 decks of cards and uses standard Baccarat drawing rules. A shoe containing 8 decks is used for a series of Exchange Baccarat games, and is reset after there are 7 or less cards remaining prior to starting a game OR the remaining cards provide a guaranteed outcome for either the Player and/or Banker. A Card Counter table is provided at all times in the Game Movie which shows the cards remaining in the shoe.

Betting Rounds

Exchange Baccarat starts with a round of betting where no cards are yet dealt for the Player or Banker. At the end of the first round the player and banker are both dealt a card and the second round of betting commences. At the end of the second round, the player is dealt a second card and the third round of betting commences. The banker is dealt a second card at the end of the third round and the game will end here if neither the player nor the banker will take another card. This is determined using the standard drawing rules for Baccarat (see below). If the player will take a third card, then another round of betting commences before the card is dealt. If the banker will take a third card, then a further round of betting commences before the card is dealt. The scores of the banker and player are compared at the end of the game, with the winner having the highest score.In the event of a Tie, stakes for bets on

Figure *40* The Baccarat Display on the Betfair Exchange Games Website (showing the rules tab)

```
<markets xmlns="urn:betfair:games:api:v1" currency="GBP">
  <market id="2563078" nextId="2563103">
    <status>SUSPENDED_GAME_ROUND_OVER</status>
    <commissionRate>1.5</commissionRate>
    <marketType>SINGLE_WINNER_OR_TIE</marketType>
    <selections type="MainBets">
     <selection id="1032038">
        <name>Player</name>
        <resource href="https://api.betfair.com/rest/v1/selection/tradeActivity
         ?username=tex&marketId=1032038&selectionId=1032070"
         responseType="selectionTradeActivity" title="Trade Activity"/>
        <status>IN_PLAY</status>
        <amountMatched>16.0</amountMatched>
        <profitLoss>12.0<profitLoss>
        <bestAvailableToBackPrices/>
        <bestAvailableToLayPrices/>
      </selection>
      <selection id="1032037">
        <name>Banker</name>
        <resource href="https://api.betfair.com/rest/v1/selection/tradeActivity
         ?username=tex&marketId=1032038&selectionId=1032080"
         responseType="selectionTradeActivity" title="Trade Activity"/>
        <status>IN_PLAY</status>
        <amountMatched>0.0</amountMatched>
        <profitLoss>-8.0<profitLoss>
        <bestAvailableToBackPrices/>
        <bestAvailableToLayPrices/>
      </selection>
      <selection id="1032039">
        <name>Tie</name>
        <resource href="https://api.betfair.com/rest/v1/selection/tradeActivity
         ?username=tex&marketId=1032038&selectionId=1032090"
         responseType="selectionTradeActivity" title="Trade Activity"/>
        <status>IN_PLAY</status>
        <amountMatched>0.0</amountMatched>
        <bestAvailableToBackPrices/>
        <bestAvailableToLayPrices/>
      </selection>
    </selections>
  </market>
  <market id="2563079" nextId="2563104">
    <status>SUSPENDED_GAME_ROUND_OVER</status>
    <commissionRate>2.5</commissionRate>
    <marketType>ANY_NUMBER_OF_WINNERS</marketType>
    <selections type="SideBets">
     <selection id="1032010">
        <name>Pocket Pair In Any Hand</name>
        <resource href="https://api.betfair.com/rest/v1/selection/tradeActivity
         ?username=tex&marketId=2563079&selectionId=1032010"
         responseType="selectionTradeActivity" title="Trade Activity"/>
        <status>LOSER</status>
        <amountMatched>0.0</amountMatched>
        <bestAvailableToBackPrices/>
        <bestAvailableToLayPrices/>
      </selection>
      <selection id="1032011">
        <name>Highest Hand Has Nine</name>
        <resource href="https://api.betfair.com/rest/v1/selection/tradeActivity
         ?username=tex&marketId=2563079&selectionId=1032011"
         responseType="selectionTradeActivity" title="Trade Activity"/>
        <status>IN_PLAY</status>
        <amountMatched>0.0</amountMatched>
        <bestAvailableToBackPrices/>
        <bestAvailableToLayPrices/>
        </selection>
      <selection id="1032012">
        <name>Highest Hand Is Odd</name>
        <resource href="https://api.betfair.com/rest/v1/selection/tradeActivity
         ?username=tex&marketId=2563079&selectionId=1032012"
         responseType="selectionTradeActivity" title="Trade Activity"/>
        <status>IN_PLAY</status>
        <amountMatched>0.0</amountMatched>
        <profitAndLoss>
```

```
             <profit>10.0</profit>
             <loss>-5.0</loss>
          </profitAndLoss>
          <bestAvailableToBackPrices/>
          <bestAvailableToLayPrices/>
        </selection>
        <selection id="1032013">
          <name>Natural Win</name>
          <resource href="https://api.betfair.com/rest/v1/selection/tradeActivity
           ?username=tex&marketId=2563079&selectionId=1032013"
           responseType="selectionTradeActivity" title="Trade Activity"/>
          <status>LOSER</status>
          <amountMatched>0.0</amountMatched>
          <bestAvailableToBackPrices/>
          <bestAvailableToLayPrices/>
        </selection>
        <selection id="1032014">
          <name>Natural Tie</name>
          <resource href="https://api.betfair.com/rest/v1/selection/tradeActivity
           ?username=tex&marketId=2563079&selectionId=1032014"
           responseType="selectionTradeActivity" title="Trade Activity"/>
          <status>LOSER</status>
          <amountMatched>0.0</amountMatched>
          <bestAvailableToBackPrices/>
          <bestAvailableToLayPrices/>
        </selection>
      </selections>
    </market>
  </markets>
```

Figure 41 Excerpt from a “channelSnapshot” Showing the Two “market” Elements for Baccarat

8.3.5 Requesting XML Stating How Baccarat is Represented (channelInfo)

Using the following URL, you can read a brief description of the way that the Games API uses the generic elements “object” and “selection” to represent a game of Baccarat and its betting markets:

https://api.games.betfair.com/rest/v1/channels/*channelID*/info?username=sampleuser

where *channelID* is an integer value identifying the channel being used for Blackjack (this number is automatically included in the URLs returned for each channel in the “channelLanding” page, see Figure 11 above). Figure 42 shows a sample “channelInfo” element for Baccarat, with the contents of the “gameDescription” element highlighted. The text describes how the Games API’s generic elements are used to represent the game of Baccarat.

```
<channelInfo xmlns="urn:betfair:games:api:v1">
  <round number="1">
    <channelInfo>
       <round number="1">
         <bettingWindow>33</bettingWindow>
         <playingWindow>8</playingWindow>
       </round>
       <round number="2">
         <bettingWindow>33</bettingWindow>
         <playingWindow>8</playingWindow>
       </round>
       <round number="3">
         <bettingWindow>39</bettingWindow>
         <playingWindow>8</playingWindow>
       </round>
       <round number="4">
         <bettingWindow>30</bettingWindow>
       <playingWindow>8</playingWindow>
       </round>
       <round number="5">
         <bettingWindow>30</bettingWindow>
         <playingWindow>8</playingWindow>
       </round>
       <round number="6">
```

```
         <bettingWindow>0</bettingWindow>
         <playingWindow>10</playingWindow>
       </round>
       <gameRules>NOT AVAILABLE</gameRules>
         <gameDescription>
          There are four objects: "Player", "Banker", "Shoe", and "Scorecard".

          "Player" and "Banker" are also the names of selections.

          All card names for cards dealt to the "Player" and "Banker" will be presented as
          "Card <id>", where <id>; is a number greater than 0. In each case, the card
          value will be presented as a number (between 0 and 51) which maps to a single card in
          the traditional deck according to the following matrix.

          Values Returned for 'Card' Objects:

                   | Ace |  2 |  3 |  4 |  5 |  6 |  7 |  8 |  9 | 10 |  J |  Q | K  |
          ---------+-----+----+----+----+----+----+----+----+----+----+----+----+----+
          clubs    |  0  |  1 |  2 |  3 |  4 |  5 |  6 |  7 |  8 |  9 | 10 | 11 | 12 |
          diamonds | 13  | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |
          hearts   | 26  | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 |
          spades   | 39  | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 |

          When a card has not been turned over / dealt, the value "NOT AVAILABLE" is returned.

          The contents of the "Shoe" persist across multiple games. In the "Shoe", there will
          be a property element that presents the number of cards dealt; this will be a number
          greater than or equal to 0. There will also be a property element that presents the
          number of cards remaining in the "Shoe"; this will be a number greater than or equal
          to 0. And finally in the "Shoe" there will be thirteen more property elements, one
          for each possible face value (Ace through to and including King) in a suit of cards.
          The value for each of these thirteen property elements is the number of cards of that
          face value (irrespective of suit) remaining in the "Shoe". Remember that, to begin
          with, the "Shoe" contains eight decks of cards.

          The "Scorecard" presents the points value of all winning hands so far achieved by the
          "Player" and "Banker" in the order in which they occurred. The value of the
          "Scorecard" property element is an ASCII string. In this string the characters P and
          B stand for "Player" and "Banker", respectively. The numbers that follow them are in
          each case the points value of a single winning hand. All the numbers following a P
          represent hands won by the "Player"; all numbers following a B represent hands won by
          the "Banker". When the character T occurs, it means that the "Player" and "Banker"
          have tied. In addition to the P, B and T characters, the "Scorecard" value string can
          contain the following characters and pairs of characters: 1,2,3,4,5,6,7,8,9,N8,N9,
          where N8 and N9 mean Natural Eight and Natural Nine, respectively. So, for example,
          B2349N8T34P4579 indicates that the "Banker" won with hands totalling 2,3,4,9,and N8
          points in succession. After that there was a Tie, and after that the "Banker" won
          hands worth 3 and then 4 points. Finally, the "Player" won hands totalling 4, 5, 7,
          and 9 points in succession.
         </gameDescription>
</channelInfo>
```

Figure *42* The “gameDescription” Element (contained within the “channelInfo” Element) for Baccarat

For further information about the “channelInfo” element refer to Section 4.4.

Chapter 9 Exchange Hi Lo

This chapter describes the Betfair Exchange Games version of the casino game Hi Lo and the XML that is used by the Exchange Games API to represent it. The chapter tells you:

- Which URLs to use to request the current state of a game of Exchange Hi Lo.
- The rules adopted by the Betfair implementation of Hi Lo.
- How the generic XML elements defined in the Betfair Exchange Games API schema are used to represent a game of Hi Lo, and
- Which URL to use to request information about the betting time available after each round of the automated game.

9.1 The URLs for Accessing a Game of Exchange Hi Lo

The URLs for accessing a game of Exchange Hi Lo are presented in the “channelLanding” page, which is described in Section 4.1. The URLs are shown in Figure 43, and the element identifying the channel and game is printed in bold-type.

```
<channel gameType="HILO" id="1444093" name="Exchange Hi Lo">
<resource href="https://api.games.betfair.com rest/v1/channels/1444093/snapshot
?username=sampleuser "responseType="channelSnapshot" title="Channel Snapshot"/>

<resource href="https://api.games.betfair.com rest/v1/channels/1444093/history
?username=sampleuser "responseType="channelHistory" title="Channel history"/>

<resource href="https://api.games.betfair.com rest/v1/channels/1444093/info
?username=sampleuser "responseType="channelInfo" title="Channel Definition"/>

<resource href="https://api.games.betfair.com rest/v1/channels/1444093/snapshot
?username=sampleuser&amp;selectionsType=Mainbets" responseType="channelSnapshot"
title="Channel Snapshot - Mainbets"/>

resource href="https://api.games.betfair.com rest/v1/channels/1444093/snapshot
?username=sampleuser&amp;selectionsType=SideBets" responseType="channelSnapshot"
title="Channel Snapshot - SideBets"/>
</channel>
```

Figure 43 Excerpt from a “channelLanding” Element Showing all the “resource” Elements for Hi Lo

NOTE: The Betfair Exchange Games implementation of Hi Lo uses rules that differ in some respects from the conventional rules. Before you place any bets on this game, make sure you understand the rules and logic followed for the Betfair version of the game (see Section 9.2).

9.2 The Rules Followed by the Betfair Game of Hi Lo

This game is Betfair’s version of the popular card game called Hi Lo. This section describes the progress of a game and explains the strategy of the automated player on whose success in a game it is up to you (or the users of your client application) to place bets.

This section also introduces the two separate betting markets associated with Hi Lo.

9.2.1 The Progress of a Game of Exchange Hi Lo

Exchange Hi Lo is Betfair’s implementation of the card game called Hi Lo in which a single player predicts whether the next card in the pack will be of a higher or lower value than the last. However, in the Betfair implementation, the player who predicts this is an automated player, and you can place back or lay bets on how far you think the automated player will progress in the game before it predicts incorrectly, or on a number of alternative betting selections available in a side bets market (see Section 9.2.3).

The automated player follows a simple strategy to predict whether the next card will be of a higher or lower value than the last. This strategy is described in the next section below.

For each game there is only a single suit of cards involved: Spades; and there are 13 cards in the suit. Also, in this game the Ace is high. In other words, the Ace has a higher value than the King and is, therefore, for the purposes of this game, the highest card in the pack. The cards in the suit (from lowest to highest) are therefore the 2, 3, 4, 5, 6, 7, 8, 9, 10, Jack, Queen, King, and Ace of Spades.

A game of Exchange Hi Lo starts before the first card has been dealt, because you can place bets in either or both markets (the Correct Predictions market or the Side bets market; for more details of which, see Section 9.2.3) before the first card has been dealt.

As with all Betfair Exchange games, Hi Lo proceeds until it reaches its logical conclusion even if the outcome becomes known before the end. Also as usual, any betting selection that represents a guaranteed outcome is eliminated from the betting market as soon as its probability becomes a certainty. Bets on selections that have become guaranteed outcomes are settled by the exchange server before the next round of betting starts.

The game of Exchange Hi Lo ends either when the automated player has predicted incorrectly or when the last card in the suit of 13 has been dealt.

Finally, if the automated player predicts correctly for Card 11, then Card 11 and the Last card will be revealed at the same time. This is because there is no betting round for the Last card, since (it being the only card whose value has not been revealed) its value is a certainty.

9.2.2 The Strategy Followed by the Automated Player in Exchange Hi Lo

The automated player makes its predictions according to the following logic. If:

1. The number of undealt cards with a higher value than the last dealt card is greater than
2. The number of undealt cards with a value lower than the last dealt card, then
3. The automated player will predict the next card's value to be higher than the last one's.
4. Otherwise it will predict the next card's value to be lower than the last one's.

For example, if the first dealt card is the 5 of Spades, then, out of the remaining possible cards, there are three with a value that is lower (the 2, 3, and 4 of Spades) and nine with a value that is higher (the 6, 7, 8, 9, 10, Jack, Queen, King, and Ace of Spades) than the value of the last card dealt. In this case, therefore, the automated player's prediction is "Higher".

NOTE: The automated player is a "perfect" player in the sense that it always predicts on the basis of complete knowledge of which cards have already been dealt, how many undealt cards remain that are of a lower value than the last dealt card, and how many undealt cards remain that are of a higher value than the last dealt card.

In other words, although the automated player's predictions can obviously turn out to be wrong, the calculations of probability on which they are based cannot be wrong, and neither can the data on which those calculations are themselves based.

In the "movie" representation for a game of Hi Lo on the Betfair Games website (www.games.betfair.com) a card counter is displayed (see Figure 44). This helps website users make their betting decisions by seeing the following information at a glance:

- The value of the last dealt card,
- How many undealt cards remain with a lower value than the last dealt card,
- How many remain with a higher value than the last dealt card,
- And which cards have already been dealt.

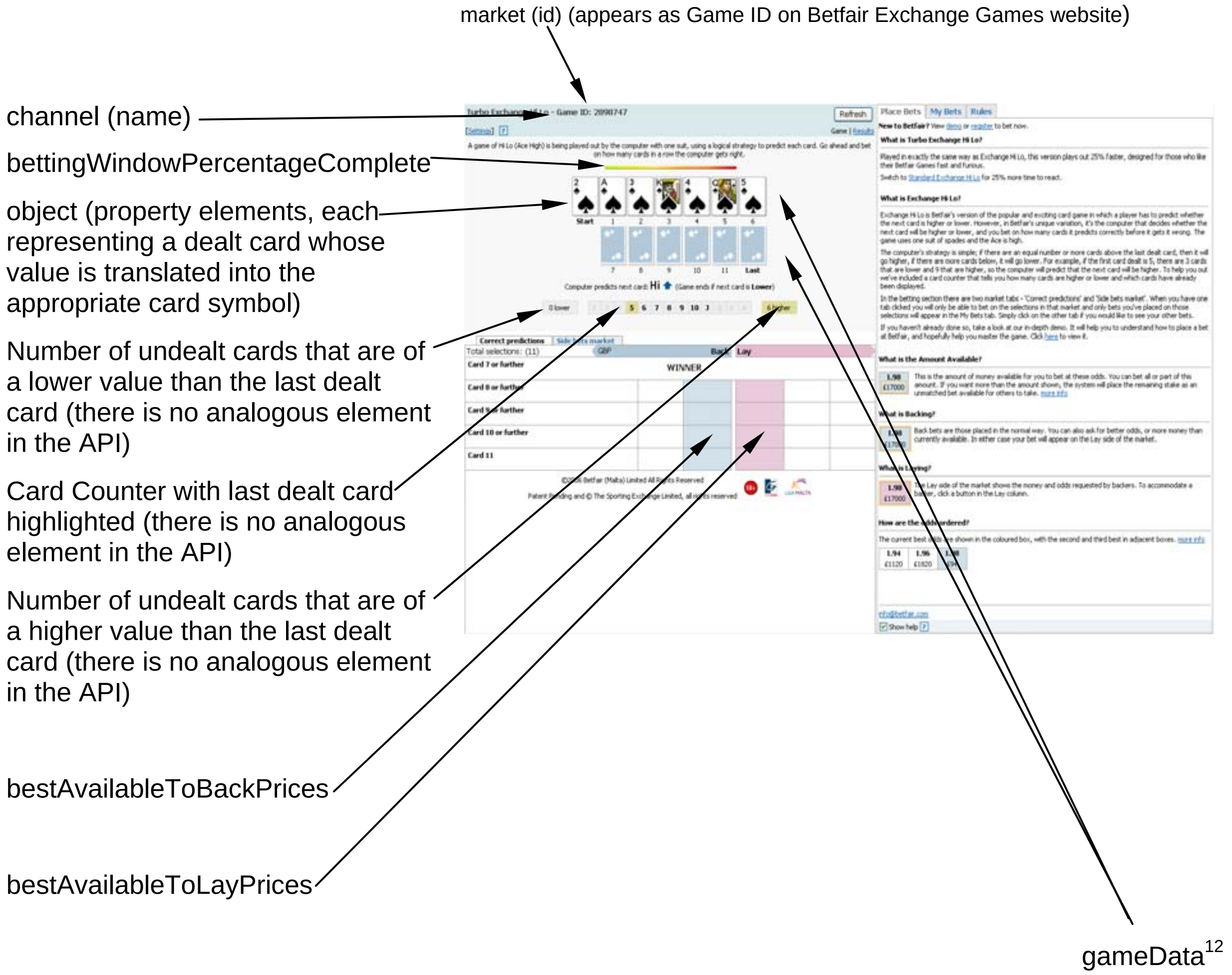

market (id) (appears as Game ID on Betfair Exchange Games website)
channel (name)
bettingWindowPercentageComplete
object (property elements, each representing a dealt card whose value is translated into the appropriate card symbol)
Number of undealt cards that are of a lower value than the last dealt card (there is no analogous element in the API)
Card Counter with last dealt card highlighted (there is no analogous element in the API)
Number of undealt cards that are of a higher value than the last dealt card (there is no analogous element in the API)
bestAvailableToBackPrices
bestAvailableToLayPrices
gameData[12]

Figure 44 The Movie Representation of a Game of Hi Lo on the Betfair Website

The card counter, therefore, gives website users the same information as is available to the automated player. The API, however, does not explicitly provide this information to client applications. Therefore, you will probably want to generate some simple code to enable your client either to include this information in its betting decisions (for example, if the client you are developing is a betting ‘bot) or to display the information (if the client you are developing is a user interface).

You can work out the figures for the remaining cards in the pack from the list of cards that have already been dealt. Each dealt card is represented in the API by a “property” element belonging to the Hi Lo’s main “object” element, which is called “board”. (For more information about these and the other XML elements used to represent a game of Hi Lo, see Section 9.3.)

[12] In the API the Hi Lo “gameData” element comprises: the single “object” called “board”, the “description” element containing the automated player’s current prediction, and the “property” elements representing each card dealt. Note that there is no XML element analogous to the Card Counter in the API’s representation of a game of Hi Lo.

9.2.3 The Two Betting Markets for Exchange Hi Lo

In Exchange Hi Lo there are two betting markets: the “Correct predictions” and “SideBets” markets.

9.2.3.1 Understanding How to Back and Lay in the Hi Lo “CorrectPredictions” Market

In the “Correct Predictions” market you are betting on whether the automated player will get to Card 1 or further, Card 2 or further, Card 3 or further, and so on, before predicting incorrectly whether the next card will be of a higher or lower value than the last dealt card.

If you think the automated player will predict:

- Correctly for at least the first dealt card after the Start card (but you are not confident of its performance after that), then you need to back the selection “Card 1”. If you do this and the automated player does indeed predict correctly for Card 1, then you will win your bet whatever happens subsequently in the game.
- Correctly for at least Card 1, Card 2, and Card 3, then you need to back the selection “Card 3 or further”. If you do this and the automated player does indeed predict correctly for Card 1, Card 2, and Card 3, then you will win your bet whatever happens subsequently in the game.
- Incorrectly for Card 1, you need to lay the selection “Card 1 or further”. If you do this and the automated player does indeed predict incorrectly for Card 1, then you will win your lay bet.
- Incorrectly during any one of its first four attempts, then you need to lay the selection “Card 4 or further”. Then, if the automated player fails to predict correctly for Card 1, Card 2, Card 3, or Card 4, you will win your lay bet.

9.2.3.2 Understanding How to Back and Lay in the Hi Lo “SideBets” Market

In the “SideBets” market for Hi Lo you are not betting explicitly on how far you think the automated player will progress in the game before it predicts incorrectly. Instead you are betting on the nature and length of the sequences of cards dealt or on whether particular cards or types of cards will appear in the list of cards dealt before the game finishes.

The betting selections are as follows:

- “2 card run”
  Back this selection if you think the list of cards dealt in the game will include two cards in a row in either ascending or descending order of value. Examples of two-card runs are: 2 followed by 3; 3 followed by 2; Queen followed by King; and King followed by Queen.
- “3 card run”
  Back this selection if you think the list of cards dealt in the game will include three cards in a row in either ascending or descending order of value. Examples of three-card runs are: 7 followed by 8 followed by 9; 9 followed by 8 followed by 7; 10 followed by Jack followed by Queen; and Queen followed by Jack followed by 10.
- “4 card run”:
  Back this selection if you think the list of cards dealt in the game will include four cards in a row in either ascending or descending order of value. Examples of four-card runs are: 6 followed by 7 followed by 8 followed by 9; and Ace followed by King followed by Queen followed by Jack.
- “No Ace”
  Back this selection if you think the list of cards dealt in the game will not include an Ace.
- “No picture cards”
  Back this selection if you think the list of cards dealt in the game will not include picture cards. The picture cards are the Jack, Queen, King and Ace.

In the Side bets market, the success of a bet is determined by the cards that are dealt in the game (not by whether the automated player predicts correctly for a given card, although this obviously bears on the number of cards dealt and therefore on the possible number of cards in a run).

An example of how it works is as follows: If the Start card was the King, then the automated player's prediction for Card 1 would be "lower", because only one card (the Ace) has a higher value than the King while eleven cards have a lower value than it. In this situation then, if Card 1 did indeed turn out to be an Ace, the game would end because the automated player had predicted incorrectly, and the settlement of bets in the "SideBets" market would be as follows: the "2 card run" selection would be settled as a winner because the King followed by the Ace was a two-card run; the other selections would all all be settled as losers, a King having been dealt (so that the "No picture cards" selection lost), an Ace having also been dealt (so that the "No Ace" selection lost), and there having been no three- or four-card runs (because only one card was dealt after the start card).

9.3 The Main XML Elements Used to Represent a Game of Hi Lo

Figure 45 shows XML excerpted from the "channelSnapshot" element of a game of Hi Lo. Just as with the Poker, Blackjack, and Baccarat examples in earlier chapters, the two main structures are called "object" (nested inside the element "gameData") and "selection" (nested inside the element "market", shown in Figure 46).

The "object" elements are generic for all games but the number of "object" elements used for a game, and the semantics of the sub-elements inside each "object" vary between the different exchange games. For an explanation of how the "object" element is implemented in Hi Lo, see Section 9.3.1

9.3.1 The "object" Element for Hi Lo (object)

For Hi Lo a single "object" element (called "board") represents the whole of the game. Hi Lo is unique among the Exchange games in that the automated player is not represented by its own "object". In fact, for practical purposes, the automated player is invisible, but its most recent prediction is contained in the "object" element's "description". The Hi Lo "object" element has:

- a name (which for this game is always "board"),
- a description (which holds the otherwise invisible automated player's most recent prediction of "higher", "lower", or "none"; "none" means that the automated player is not making a prediction, either because the Start card has not been dealt yet (and therefore the automated player has no information on the basis of which to make its prediction), or because the Last card is about to be dealt (and therefore the automated player does not make a prediction because the Last card's value is known as a certainty, all the other cards in the suit having been dealt already);
- a "status" (of IN_PLAY, LOSER, or WINNER, indicating respectively: whether the game is still in progress; whether the game has finished, with the automated player predicting incorrectly and therefore acquiring the status of LOSER; or whether the game has finished, with the automated player predicting correctly all the way through to the last card and therefore acquiring the status of WINNER);
- and from zero to 13 "property" elements representing the cards dealt as the game progresses. The first property element is named "Start", which is the card dealt before the automated player's first prediction. The eleven cards that the automated player makes its predictions on are named "Card <sequence>" where <sequence> is a number from 1 to 11. The final card, which is revealed if the 11th prediction is correct, is named "Last". No prediction is made for this card because it is the only undealt card in the deck at this point in the game (its value is therefore a certainty). The value of each card is presented as a number (between 0 and 51) which maps to a single card in the traditional deck according to the mappings given in Figure 29.

```
<gameData>
   <object name="Board">
     <description>Higher</description>
     <status>IN_PLAY</status>
     <property name="Start" value="40"/>
     <property name="Card 1" value="39"/>
     <property name="Card 2" value="41"/>
     <property name="Card 3" value="51"/>
     <property name="Card 4" value="42"/>
     <property name="Card 5" value="50"/>
     <property name="Card 6" value="43"/>
     <property name="Card 7" value="49"/>
     <property name="Card 8" value="44"/>
     <property name="Card 9" value="48"/>
     <property name="Card 10" value="45"/>
   </object>
 <gameData>
```

Figure 45 Excerpt from a “channelSnapshot” Showing the Single “object” Element in a Game of Hi Lo

9.3.2 The “marketType” Element for the “CorrectPredictions” Market in Hi Lo (marketType)

The “CorrectPredictions” market for Hi Lo is of the type VARIABLE_HANDICAP. This type of market is like the ANY_NUMBER_OF_WINNERS type in that it can have zero or more winners. However, it is unlike it in that, in a variable handicap market, there are dependencies between the selections.

For example, in a game of Hi Lo, for the selection “Card 4 or further” to win, the selections “Card 1 or further”, “Card 2 or further”, and “Card 3 or further” must themselves already have won. (Bets are settled as soon as a selection’s status as winner has been determined.)

Similarly, if the selection “Card 5 or further” loses, then the selection “Card 6 or further” will also lose, as will the selection “Card 7 or further”, and as will all the other selections in the market that have not already won.

The dependencies between the selections in a variable handicap market mean that it is possible for there to be no winners in the market. In Hi Lo, for example, if the automated player fails to predict correctly for card 1, then the selection “Card 1 or further” loses, none of the other selections in the market can possibly win, and the game ends.

```
<market id="2908881" nextId="2908906">
   <status>ACTIVE</status>
   <commissionRate>2.0</commissionRate>
   <marketType>VARIABLE_HANDICAP</marketType>
   <selections type="CorrectPredictions">
       <selection id="1032029">
           <name>Card 1 or further</name>
           <resource href="https://api.betfair.com/rest/v1/selection
            /tradeActivity?username=tex&marketId=2908881&selectionId=1032029"
            responseType="selectionTradeActivity" title="Trade Activity"/>
           <status>WINNER</status>
           <amountMatched>184.0</amountMatched>
           <bestAvailableToBackPrices/>
           <bestAvailableToLayPrices/>
       </selection>
       <selection id="1032030">
           <name>Card 2 or further</name>
           <resource href="https://api.betfair.com/rest/v1/selection
            /tradeActivity?username=tex&marketId=2908881&selectionId=1032030"
            responseType="selectionTradeActivity" title="Trade Activity"/>
           <status>WINNER</status>
           <amountMatched>296.0</amountMatched>
           <bestAvailableToBackPrices/>
           <bestAvailableToLayPrices/>
       </selection>
        . . .
       <selection id="1032037">
           <name>Card 9 or further</name>
            <resource href="https://api.betfair.com/rest/v1/selection
            /tradeActivity?username=tex&marketId=2908881&selectionId=1032037"
            responseType="selectionTradeActivity" title="Trade Activity"/>
           <status>WINNER</status>
           <amountMatched>58.0</amountMatched>
           <bestAvailableToBackPrices/>
           <bestAvailableToLayPrices/>
       </selection>
       <selection id="1032038">
           <name>Card 10 or further</name>
            <resource href="https://api.betfair.com/rest/v1/selection
            /tradeActivity?username=tex&marketId=2908881&selectionId=1032038"
            responseType="selectionTradeActivity" title="Trade Activity"/>
           <status>WINNER</status>
           <amountMatched>560.0</amountMatched>
           <bestAvailableToBackPrices/>
           <bestAvailableToLayPrices/>
       </selection>
       <selection id="1032039">
           <name>Card 11</name>
            <resource href="https://api.betfair.com/rest/v1/selection
            /tradeActivity?username=tex&marketId=2908881&selectionId=1032039"
            responseType="selectionTradeActivity" title="Trade Activity"/>
           <status>WINNER</status>
           <amountMatched>450.0</amountMatched>
           <bestAvailableToBackPrices/>
           <bestAvailableToLayPrices/>
       </selection>
   </selections>
</market>
```

Figure 46 Excerpt from a “channelSnapshot” Showing a “CorrectPrediction”s Market for Hi Lo[13]

[13] The dots in the sample XML output indicate the omission of repetitive data.

9.3.3 The “marketType” Element for the “SideBets” Market in Hi Lo (marketType)

The “SideBets” market for Hi Lo is of the type ANY_NUMBER_OF_WINNERS (see Figure 47). In this type of market, wherever one selection wins there is no necessary implication that another loses (or that all other selections lose). However, in the Hi Lo market there are dependencies between some of the selections: for the selection “4 card run” to win, the selections “3 card run” and “2 card run” must also have won. The other two selections are “No Ace” and “No picture cards”. For descriptions of all these selections, see Section 9.2.3.2.

In this type of market it is possible for no selection to win. In Hi Lo, for example, it is perfectly possible for the game to run its course without any cards being dealt in a run, without the Ace being dealt, and without any picture cards being dealt.

It is also possible in this type of market for selections to be eliminated before the end of the game. In Hi Lo, for example, if an Ace is dealt, then the “No Ace” selection is immediately eliminated. Similarly, if a Jack is dealt, then the “No picture cards” selection is immediately eliminated. Bets on selections that are eliminated in this way are settled immediately by the exchange server.

9.3.4 The “selection” Element for Hi Lo (selection)

For Hi Lo, there are two betting markets and therefore the “markets” element itself contains two elements called “market”. Each of these “market” elements in turn contains a single “selections” element, one of type “Correct predictions” and one of type “SideBets”, and each contains a sequence of “selection” elements.

In the “CorrectPredictions” betting market, each selection has:

- a name (“Card 1 or further”, “Card 2 or further”, through to “Card 11 or further”),
- a “selectionTradeActivity” resource for viewing all available prices for the selection (see Section 4.2.2),
- a “status” (which can be IN_PLAY, LOSER, or WINNER),
- a figure for the amount of the bet that has been matched (this is presented in the element called “amountMatched”),
- and figures for the best back and lay prices.

In the “SideBets” betting market, each betting selection has:

- a name (”2 card run”, 3 card run”, “4 card run”, “No Ace”, and “No picture card”),
- a “selectionTradeActivity” resource for viewing all available prices for the selection (see Section 4.2.2),
- a “status” (which can be IN_PLAY, LOSER, or WINNER),
- a figure for the amount of the bet that has been matched (this is presented in the element called “amountMatched”),
- and figures for the best back and lay prices.

```
 <market id="2904737" nextId="2904763">
    <status>ACTIVE</status>
    <commissionRate>2.0</commissionRate>
    <marketType>WIN_ONLY</marketType>
    <selections type="SideBets">
        <selection id="1032040">
            <name>2 Card Run</name>
            <resource href="https://api.games.betfair.com/rest/v1/selection
             /tradeActivity?username=tex&marketId=2904737&selectionId=1032040"
             responseType="selectionTradeActivity" title="Trade Activity"/>
            <status>WINNER</status>
            <amountMatched>150.0</amountMatched>
            <bestAvailableToBackPrices/>
            <bestAvailableToLayPrices/>
        </selection>
        <selection id="1032041">
            <name>3 Card Run</name>
            <resource href="https://api.games.betfair.com/rest/v1/selection
             /tradeActivity?username=tex&marketId=2904737&selectionId=1032041"
             responseType="selectionTradeActivity" title="Trade Activity"/>
            <status>IN_PLAY</status>
            <amountMatched>438.0</amountMatched>
            <bestAvailableToBackPrices/>
            <bestAvailableToLayPrices/>
        </selection>
        <selection id="1032042">
            <name>4 Card Run</name>
             <resource href="https://api.games.betfair.com/rest/v1/selection
             /tradeActivity?username=tex&marketId=2904737&selectionId=1032042"
             responseType="selectionTradeActivity" title="Trade Activity"/>
            <status>LOSER</status>
            <amountMatched>266.0</amountMatched>
            <bestAvailableToBackPrices/>
            <bestAvailableToLayPrices/>
        </selection>
        <selection id="1032043">
            <name>No Ace</name>
            <resource href="https://api.games.betfair.com/rest/v1/selection
             /tradeActivity?username=tex&marketId=2904737&selectionId=1032043"
             responseType="selectionTradeActivity" title="Trade Activity"/>
            <status>LOSER</status>
            <amountMatched>170.0</amountMatched>
            <bestAvailableToBackPrices/>
            <bestAvailableToLayPrices/>
        </selection>
        <selection id="1032044">
            <name>No Picture Cards</name>
            <resource href="https://api.games.betfair.com/rest/v1/selection
             /tradeActivity?username=tex&marketId=2904737&selectionId=1032044"
             responseType="selectionTradeActivity" title="Trade Activity"/>
            <status>LOSER</status>
            <amountMatched>960.0</amountMatched>
            <bestAvailableToBackPrices/>
            <bestAvailableToLayPrices/>
        </selection>
    </selections>
</market>
```

Figure 47 Excerpt from a “channelSnapshot” Showing the “SideBets” Market for a Game of Hi Lo

9.3.5 Requesting XML Stating How Hi Lo is Represented (channelInfo)

Using the following URL, you can read a brief description of the way that the Games API uses the generic elements “object” and “selection” to represent a game of Hi Lo and its betting markets:

`https://api.games.betfair.com/rest/v1/channels/channelID/info?username=sampleuser`

where *channelID* is an integer value identifying the channel being used for Hi Lo or (this number is automatically included in the URLs returned for each channel in the “channelLanding” page, see Figure 11 on page 23).

```
<channelInfo xmlns="urn:betfair:games:api:v1">
    <round number="1">
        <bettingWindow>5</bettingWindow>
        <playingWindow>5</playingWindow>
    </round>
    <round number="2">
        <bettingWindow>5</bettingWindow>
        <playingWindow>5</playingWindow>
    </round>
    <round number="3">
        <bettingWindow>5</bettingWindow>
        <playingWindow>5</playingWindow>
    </round>
   . . .

    <round number="13">
        <bettingWindow>5</bettingWindow>
        <playingWindow>5</playingWindow>
    </round>
    <gameRules>NOT AVAILABLE</gameRules>
    <gameDataDescription>The Object named "Board" will contain a "description" element
indicating the automated player's most recent high/low prediction ("none", "higher" or
"lower"), a "status" element indicating whether the game is still in progress ("IN_PLAY",
"WINNER" or "LOSER") and from zero to 13 "property" elements representing the cards displayed
on the screen as the game progresses. The first property element is named "Start", which is the
card revealed before the first prediction. The eleven cards that the automated player makes its
predictions on are named "card &lt;sequence&gt;" where &lt;sequence&gt; is a number from 1 to
11, and the final card, revealed if the 11th prediction is correct, is named "Last" (there is
no prediction for this card as it is the only undealt card in the deck at this point). The
value of each card will be presented as a number (between 0 and 51) which maps to a single card
in the traditional deck according to the following matrix:

Values Returned for 'Card' Objects:

         | Ace |  2 |  3 |  4 |  5 |  6 |  7 |  8 |  9 | 10 |  J |  Q | K  |
---------+-----+----+----+----+----+----+----+----+----+----+----+----+----+
spades   | 39  | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 |
</gameDataDescription>
</channelInfo>
```

Figure 48 The “gameDescription” Element (contained within the “channelInfo” Element) for Hi Lo[14]

Figure 48 shows a sample “channelInfo” element for Hi Lo, with the contents of the “gameDescription” element highlighted. The text describes how the Games API’s generic elements are used to represent the game of Hi Lo. Remember that for Hi Lo only one suit of cards is used (Spades).

For further information about the “channelInfo” element refer to Section 4.4.

For the mapping of card values to specific cards in the deck, see Figure 29 on page 52.

[14] The dots in the sample XML output indicate the omission of repetitive data.

Chapter 10 Exchange Card Racing

This chapter describes the Betfair Exchange Games version of the casino game Card Racing and the XML that is used by the Exchange Games API to represent it. The chapter tells you:

- Which URLs to use to request the current state of a game of Exchange Card Racing.
- The rules adopted by the Betfair implementation of Exchange Card Racing.
- How the generic XML elements defined in the Betfair Exchange Games API schema are used to represent a game of Exchange Card Racing, and
- Which URL to use to request information about the betting time available after each round of the automated game.

10.1 The URLs for Accessing a Game of Exchange Card Racing

The URLs for accessing a game of Exchange Card Racing are presented in the “channelLanding” page, which is described in Section 4.1. The URLs are shown in Figure 43, and the element identifying the channel and game is printed in bold-type.

```
<channel gameType="CARD_RACER" id="1444116" name="Card Racer">
      <resource href="https://api.games.betfair.com/rest/v1/channels/1444116/info"
responseType="channelInfo" title="Channel Definition"/>
      <resource href="https://api.games.betfair.com/rest/v1/channels/1444116/history"
responseType="channelHistory" title="Channel History"/>
      <resource href="https://api.games.betfair.com/rest/v1/channels/1444116/snapshot"
responseType="channelSnapshot" title="Channel Snapshot"/>
</channel>

<channel gameType="CARD_RACER" id="1444126" name="Turbo Card Racer">
      <resource href="https://api.games.betfair.com/rest/v1/channels/1444126/info"
responseType="channelInfo" title="Channel Definition"/>
      <resource href="https://api.games.betfair.com/rest/v1/channels/1444126/history"
responseType="channelHistory" title="Channel History"/>
      <resource href="https://api.games.betfair.com/rest/v1/channels/1444126/snapshot"
responseType="channelSnapshot" title="Channel Snapshot"/>
</channel>
```

Figure 49 Excerpt from a “channelLanding” Element Showing all the “resource” Elements for Exchange Card Racing and Turbo Exchange Card Racing

NOTE: The Betfair Exchange Games implementation of Exchange Card Racing uses rules that differ in some respects from the conventional rules. Before you place any bets on this game, make sure you understand the rules and logic followed for the Betfair version of the game (see Section 9.2).

10.2 The Rules Followed by the Betfair Game of Exchange Card Racing

Exchange Card Racing is a game of Virtual Racing offering a realistic racing experience. The players bet against each other on a bet exchange market. Just like betting on a horse race on Betfair.com, but here the race is simulated using a standard 52 card deck.

10.2.1 Before the Race Starts

Each race begins with four runners lined up at the starting line of a 32 segment race track. Each runner is represented by a suit (Spades, Hearts, Clubs and Diamonds). A randomly shuffled pack of 52 cards is used to simulate the race.

An exchange market where the players can place back and lay bets on the runners is available. The first round of betting occurs when all 52 cards in the deck are still unseen and players can back or lay any of the 4 runners.

10.2.2 Handicaps

At the end of the first round of betting any unmatched bets are cancelled. Now, before the race can start, the handicaps are determined. Three cards of each suit are randomly removed from the deck and turned face up. The remaining 40 cards in the deck will then be used for the race simulation. A second round of betting commences in which the players can place new bets based on the handicaps.

10.2.3 During the Race

After the end of the second betting round, the race starts. Three cards will now be drawn from the deck, one by one and turned face up. After each card is turned over, the runner of the corresponding suit moves forward according to the rank of the card (the card values are the same as in the game of Blackjack with Ace always ranked 11):

- If the card ranks **2 to 10** the runner moves as many segments as the value of the card.
- If it is a face card (**Jack, Queen or King**) the runner moves 10 segments forward.
- If the card is an **Ace** the runner moves 11 segments forward.

After all three cards have been drawn and the runners have been moved forward it is time for the next betting round.

10.2.4 The Winner

The race action will now continue with three cards being drawn between each betting round until one runner crosses the finishing line. Note that as soon as a runner crosses the finishing line (i.e. reaches the 32nd segment of the race track) the race is over and no more cards will be drawn. Therefore it is not possible for a dead heat to occur. The market for Exchange Card Racing is always settled with one single winner.

10.2.5 Rules

Card Racing is played subject to these market rules as well as the Terms and Conditions of Betfair and (where relevant) Betfair Exchange Games General Rules. The Terms and Conditions of Betfair and Betfair Exchange Games General Rules will apply to any situations not covered by these Card Racing market rules, but in the event of any inconsistency, these Card Racing Conditions will prevail.

10.2.6 Market

Commission on this market: 2.5% (no discounts apply)

There are four runners (four market selections).
The first runner to complete the 32 segment race track is immediately declared winner.
There can be only one winner (Win Only Market). No Dead Heat rule applies.
The game is played using a standard 52 card deck.

The card value determines how many segments the runner of that same suit is moved forward when the card is drawn; the rankings are:

- If the card ranks **2 to 10** the runner moves as many segments as the value of the card.
- If it is a face card (**Jack, Queen or King**) the runner moves 10 segments forward.
- If the card is an **Ace** the runner moves 11 segments forward.

10.2.7 Race schedule

The time for a race to start and available time for betting each round are predefined and may be subject to change.

10.2.8 Odds Ladder

Exchange Card Racing uses a different odds ladder from the other Exchange Games. The following table shows the allowed increments for each odds.

Odds From	To	Allowed Increment
1	2	0.01
2	3	0.02
3	4	0.05
4	5	0.05
5	6	0.1
6	10	0.1
10	20	0.2
20	30	0.5
30	50	1
50	100	2
100	500	5
500	1,000	10

For example: For odds between 1 and 2, you can request odds of 1.01, 1.02, 1.03, etc. Between odds from 2 to 3, you can request odds of 2.02, 2.04, 2.06, etc. Between odds of 100 and 500, you can request odds of 105, 110, 115, etc.

10.3 The Main XML Elements Used to Represent a Game of Exchange Card Racing

Figure 45 shows XML excerpted from the “channelSnapshot” element of a game of Exchange Card Racing. Just as with the Poker, Blackjack, and Baccarat examples in earlier chapters, the two main structures are called “object” (nested inside the element “gameData”) and “selection” (nested inside the element “market”, shown in Figure 46).

The “object” elements are generic for all games but the number of “object” elements used for a game, and the semantics of the sub-elements inside each “object” vary between the different exchange games. For an explanation of how the “object” element is implemented in Exchange Card Racing, see Section 9.3.1

10.3.1 The “object” Element for Exchange Card Racing (object)

The Exchange Card Racing “object” element represents the handicap determined at the start and, for each virtual runner:

- a name (Spades, Hearts, Clubs, Diamonds)
- a "status" (of IN_PLAY, LOSER, or WINNER, indicating respectively: whether the game is still in progress; whether the game has finished, with the automated player predicting incorrectly and therefore acquiring the status of LOSER; or whether the game has finished, with the automated player predicting correctly all the way through to the last card and therefore acquiring the status of WINNER);
- a “segment” representing the position on the board.
- A “place” showing which selection is in first, second, third, etc.

```
<gameData>
    <object name="Cards">
        <property name="Handicap" value="6,0,12,24,21,15,29,30,26,46,51,44"/>
        <property name="Used" value=""/>
        <property name="Current" value="41,32,31"/>
    </object>
    <object name="Spades">
        <status>IN_PLAY</status>
        <property name="Segment" value="3"/>
        <property name="Place" value="2"/>
    </object>
    <object name="Hearts">
        <status>IN_PLAY</status>
        <property name="Segment" value="13"/>
        <property name="Place" value="1"/>
    </object>
    <object name="Clubs">
        <status>IN_PLAY</status>
        <property name="Segment" value="0"/>
        <property name="Place" value="3"/>
    </object>
    <object name="Diamonds">
        <status>IN_PLAY</status>
        <property name="Segment" value="0"/>
        <property name="Place" value="3"/>
    </object>
</gameData>
```

Figure 50 Excerpt from a “channelSnapshot” Showing the “object” Elements in a Game of Exchange Card Racing

10.3.2 The “marketType” Element for the “MainBets” Market in Exchange Card Racing (marketType)

The “MainBets” market for Baccarat is of type “SINGLE_WINNER_OR_TIE” (see Table 4 in Section 4.2). This type of market is like the WIN_ONLY market associated with Exchange Poker, but it differs from it by allowing no possibility of a dead heat between selections.

```
                 <market id="6442110" nextId="6442135">
                                 <status>ACTIVE</status>
                                 <commissionRate>3.0</commissionRate>
                                 <marketType>SINGLE_WINNER_OR_TIE</marketType>
                                 <selections type="MainBets">
                                       <selection id="1032045">
                                             <name>Spades</name>
                                             <resource
href="https://api.games.betfair.com/rest/v1/selection/tradeActivity?marketId=6442110&selectionI
d=1032045" responseType="selectionTradeActivity" title="Trade Activity"/>
                                             <status>IN_PLAY</status>
                                             <amountMatched>0.0</amountMatched>
                                             <bestAvailableToBackPrices>
                                                   <price amountUnmatched="172.0">3.95</price>
                                                   <price amountUnmatched="628.0">3.9</price>
                                                   <price amountUnmatched="533.0">3.85</price>
                                             </bestAvailableToBackPrices>
                                             <bestAvailableToLayPrices>
                                                   <price amountUnmatched="169.0">4.05</price>
                                                   <price amountUnmatched="608.0">4.1</price>
                                                   <price amountUnmatched="506.0">4.15</price>
                                             </bestAvailableToLayPrices>
                                       </selection>
                         . . .
                                                   <selection id="1032043">
                                              <name>Diamonds</name>
                                              <resource
 href="https://api.games.betfair.com/v1/selection/tradeActivity?marketId=6442110&selectionId=10
 32043" responseType="selectionTradeActivity" title="Trade Activity"/>
                                              <status>IN_PLAY</status>
                                              <amountMatched>0.0</amountMatched>
                                              <bestAvailableToBackPrices>
                                                    <price amountUnmatched="172.0">3.95</price>
                                                    <price amountUnmatched="628.0">3.9</price>
                                                    <price amountUnmatched="533.0">3.85</price>
                                              </bestAvailableToBackPrices>
                                              <bestAvailableToLayPrices>
                                                    <price amountUnmatched="169.0">4.05</price>
                                                    <price amountUnmatched="608.0">4.1</price>
                                                    <price amountUnmatched="506.0">4.15</price>
                                              </bestAvailableToLayPrices>
                                        </selection>
                                  </selections>
                            </market>
```

Figure 51 Excerpt from a “channelSnapshot” Showing a “MainBets”s Market for Exchange Card Racing[15]

[15] The dots in the sample XML output indicate the omission of repetitive data.

10.3.3 The “selection” Element for Exchange Card Racing (selection)

In the “MainBets” betting market, each selection has:

- a name (one of the four suits in a deck of card “Spades”, Hearts”, “Clubs”, or “Diamonds”)
- a “selectionTradeActivity” resource for viewing all available prices for the selection (see Section 4.2.2),
- a “status” (which can be IN_PLAY, LOSER, or WINNER),
- a figure for the amount of the bet that has been matched (this is presented in the element called “amountMatched”),
- and figures for the best back and lay prices.

10.3.4 Requesting XML Stating How Exchange Card Racing is Represented (channelInfo)

Using the following URL, you can read a brief description of the way that the Games API uses the generic elements “object” and “selection” to represent a game of Exchange Card Racing and its betting markets:

`https://api.games.betfair.com/rest/v1/channels/`*`channelID`*`/info?username=sampleuser`

where *channelID* is an integer value identifying the channel being used for Exchange Card Racing or (this number is automatically included in the URLs returned for each channel in the “channelLanding” page, see Figure 11 on page 23).

For further information about the “channelInfo” element refer to Section 4.4.

For the mapping of card values to specific cards in the deck, see Figure 29 on page 52.

Chapter 11 Managing Your Games Account Using the API

This chapter tells you how to access the main page of the Games API for accessing account information. This involves requesting XML for the “accountLanding” element from the API. The chapter also tells you specifically how to retrieve your Betfair Exchange Games account balance, your main Betfair balance, a list of your account transactions, and finally how to transfer funds into your Games account from your main Betfair account or from your Games account into your main Betfair account.

11.1 Requesting XML for Managing Your Accounts (accountLanding)

The URLs for accessing your Betfair Exchange Games account, viewing your main Betfair account balance (but not any other details of your main Betfair account), for viewing a statement of your Games account, and for transferring money between the Games account and the main Betfair account are as shown in Figure 52.

https://api.games.betfair.com/rest/v1/account?username

The call to this URL is an HTTP GET request. The name of the XML element returned is “accountLanding”.

The “accountLanding” element contains a list of four “resource” elements. (For information about the “resource” element, see Section 2.2.3.) These four elements contain URLs that enable you to:

- View your Betfair Exchange Games account balance
- View your main Betfair account balance
- View your account statement (a list of transactions)
- Transfer money into and out of your Betfair Exchange Games account

```
<accountLanding xmlns="urn:betfair:games:api:v1">

<resource href="https://api.games.betfair.com/rest/v1/account/snapshot?username=sampleuser"
  responseType="accountSnapshot" title="View your Betfair Exchange Games account balances"/>

<resource href="https://api.games.betfair.com/rest/v1/account/betfair/snapshot?
  username=sampleuser" responseType="accountBetfairSnapshot"
  title="View your Betfair account balance"/>

<resource href=https://api.games.betfair.com/rest/v1/account/statement?username=sampleuser"
  responseType="accountStatement" title="Your account statement"/>

 <resource href="https://api.games.betfair.com/rest/v1/account/transferOrder
   ?username=sampleuser" postType="postTransferOrder" responseType="responseTransferOrder"
   title="Request money transfers between your Betfair and Betfair Exchange Games accounts"/>

</accountLanding>
```

Figure 52 The "accountLanding" Page Containing URLs for Managing Your Accounts

11.2 Requesting XML Listing Your Exchange Games Account Balances (accountSnapshot)

The URL for requesting XML listing the balances of your Betfair Exchange Games account is:

https://api.games.betfair.com/rest/v1/account/snapshot?username=sampleuser

The call to this URL is an HTTP GET request. The name of the XML element returned is "accountSnapshot".

The reason that (in Figure 52) the title of the URL refers to "balances" (in the plural) is that separate values are returned for:

- the available-to-bet balance,
- the available-to-transfer-out balance,
- and the Betfair Exchange Games balance.

The first two of these balances are always the same. The Betfair Exchange Games balance will be greater than the other two balances if there is any current exposure (in other words, if unsettled bets are in place) and if there is any figure currently displayed for retained commission. The balances that are available to bet with or to transfer out are net of current exposure and retained commission.

Current exposure is the amount that the customer stands to lose if his or her current (unsettled) bets are not successful.

Retained commission is a variable figure. If you place a bet (back or lay) on a selection which, in the course of the game, becomes guaranteed to win or lose, then bet settlement takes place immediately and the whole amount of your winnings is placed in your Games account. Each time this happens, a figure for the Betfair commission on the winning bet is added to the value in the "retainedCommission" element (see Figure 53). This figure represents the commission that will be payable to Betfair, calculated on the basis of your winnings so far in the game. If you lose any bets, the figure will be reduced, and at the end of the game if you have made a net loss no commission will be payable at all.

During the game, although the values for "currentExposure" and "retainedCommission" are not deducted from your Games account, their values are subtracted from the value in the "betfairGamesBalance" element to yield the values for the "availableToBetBalance" and the "availableToTransferBalance" elements.

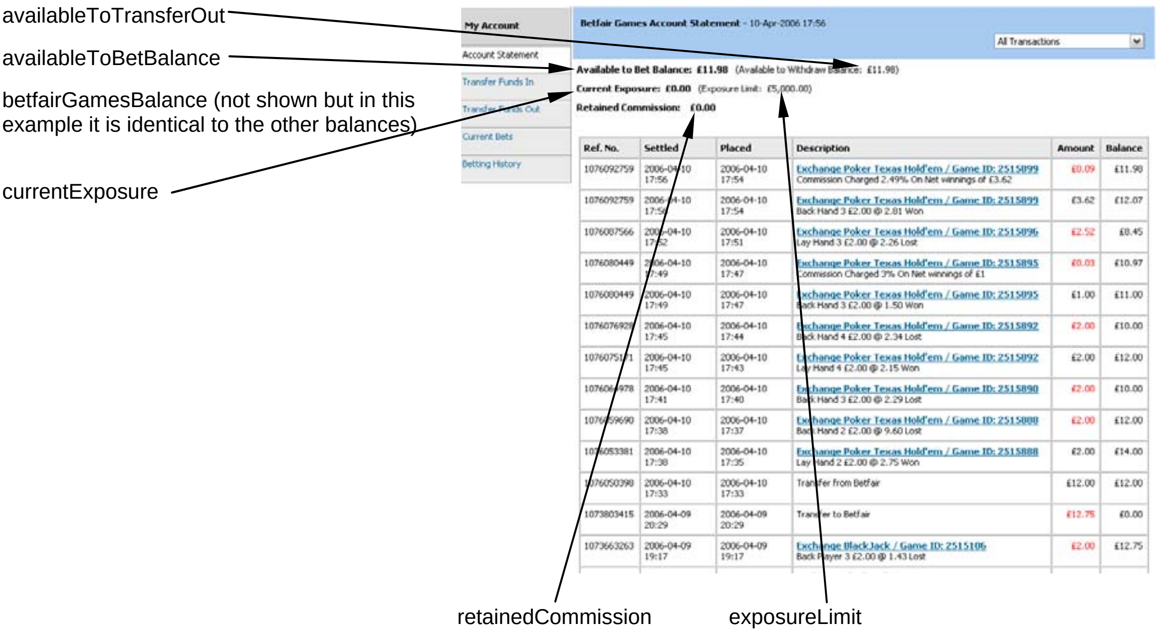
In other words, you cannot bet with funds in your Games account that may be required at the end of the game to pay for losses and commission.

Figure 53 shows XML describing a sample customer's Betfair Exchange Games account. This customer has £98.60 to bet with or transfer to his or her main Betfair account.

```
<accountSnapshot xmlns="urn:betfair:games:api:v1" currency="GBP">
   <exposureLimit>-50.0</exposureLimit>
   <availableToBetBalance>98.60</availableToBetBalance>
   <availableToTransferOut>98.60</availableToTransferOut>
   <betfairGamesBalance>105.0</betfairGamesBalance>
   <currentExposure>-5.75</currentExposure>
   <retainedCommission>0.65</retainedCommission>
 </accountSnapshot>
```

Figure *53* Sample "accountSnapshot" Element Showing Details of a User's Games Account Balances

Figure 54 shows how data from the “accountSnapshot” element is represented by the Betfair website.



Betfair Games Account Statement - 10-Apr-2006 17:56

Available to Bet Balance: £11.98 (Available to Withdraw Balance: £11.98)

Current Exposure: £0.00 (Exposure Limit: £5,000.00)

Retained Commission: £0.00

Ref. No.	Settled	Placed	Description	Amount	Balance
1076092759	2006-04-10 17:56	2006-04-10 17:54	Exchange Poker Texas Hold'em / Game ID: 2515899 Commission Charged 2.49% On Net winnings of £3.62	£0.09	£11.98
1076092759	2006-04-10 17:56	2006-04-10 17:54	Exchange Poker Texas Hold'em / Game ID: 2515899 Back Hand 3 £2.00 @ 2.81 Won	£3.62	£12.07
1076087566	2006-04-10 17:52	2006-04-10 17:51	Exchange Poker Texas Hold'em / Game ID: 2515896 Lay Hand 3 £2.00 @ 2.26 Lost	£2.52	£8.45
1076080449	2006-04-10 17:49	2006-04-10 17:47	Exchange Poker Texas Hold'em / Game ID: 2515895 Commission Charged 3% On Net winnings of £1	£0.03	£10.97
1076080449	2006-04-10 17:49	2006-04-10 17:47	Exchange Poker Texas Hold'em / Game ID: 2515895 Back Hand 3 £2.00 @ 1.50 Won	£1.00	£11.00
1076076928	2006-04-10 17:45	2006-04-10 17:44	Exchange Poker Texas Hold'em / Game ID: 2515892 Back Hand 4 £2.00 @ 2.34 Lost	£2.00	£10.00
1076075171	2006-04-10 17:45	2006-04-10 17:43	Exchange Poker Texas Hold'em / Game ID: 2515892 Lay Hand 4 £2.00 @ 2.15 Won	£2.00	£12.00
1076064978	2006-04-10 17:41	2006-04-10 17:40	Exchange Poker Texas Hold'em / Game ID: 2515890 Back Hand 3 £2.00 @ 2.29 Lost	£2.00	£10.00
1076059690	2006-04-10 17:38	2006-04-10 17:37	Exchange Poker Texas Hold'em / Game ID: 2515880 Back Hand 2 £2.00 @ 9.60 Lost	£2.00	£12.00
1076053381	2006-04-10 17:38	2006-04-10 17:35	Exchange Poker Texas Hold'em / Game ID: 2515888 Lay Hand 2 £2.00 @ 2.75 Won	£2.00	£14.00
1076050398	2006-04-10 17:33	2006-04-10 17:33	Transfer from Betfair	£12.00	£12.00
1073803415	2006-04-09 20:29	2006-04-09 20:29	Transfer to Betfair	£12.75	£0.00
1073663263	2006-04-09 19:17	2006-04-09 19:17	Exchange BlackJack / Game ID: 2515106 Back Player 3 £2.00 @ 1.43 Lost	£2.00	£12.75

Figure 54 Screen from the Betfair Exchange Games Web Site, Showing a Sample User's Balances

11.3 Requesting XML Stating Your Main Betfair Balance (betfairAvailableBalanceSnapshot)

The URL for requesting XML stating the balance of your main Betfair account is as follows:

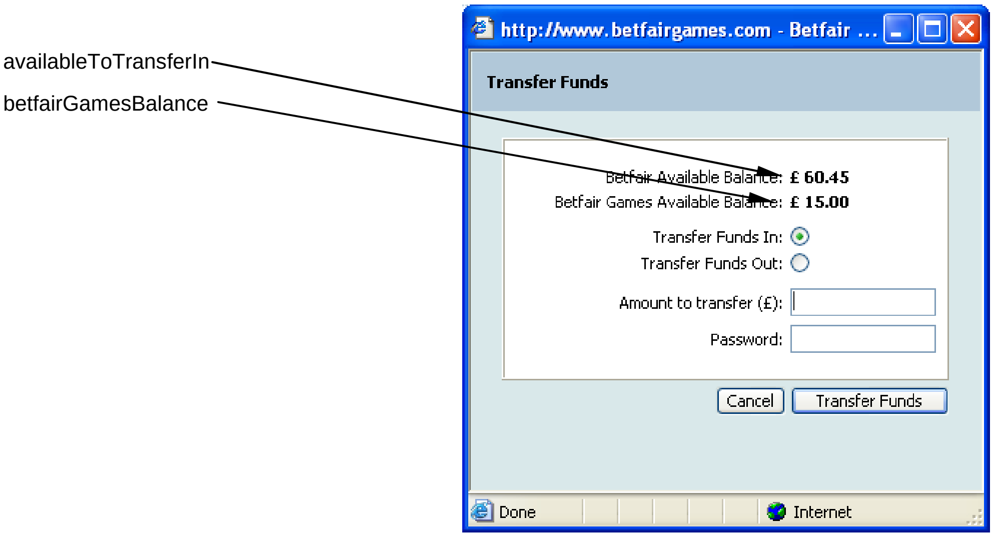
`https://api.games.betfair.com/rest/v1/account/betfair/snapshot?username=sampleuser`

The call to this URL is an HTTP GET request. The name of the XML element returned is “betfairAvailableBalanceSnapshot”. An example of the XML containing this element is given in Figure 55. And an example of how data corresponding to the data in the “betfairAvailableBalanceSnapshot” is presented on the Betfair Exchange Games website.

```
<accountBetfairSnapshot xmlns="urn:betfair:games:api:v1" currency="GBP">
   <availableToTransferIn>998.0</availableToTransferIn>
</accountBetfairSnapshot>
```

Figure *55* XML Containing Details of a Sample Main Betfair Account Containing £998.00

Note that you cannot retrieve any other details of your main Betfair account from the Games API. To do so, you must use the main Betfair website or the API available for the Betfair Sports Exchange.



Transfer Funds

Betfair Available Balance: £ 60.45
Betfair Games Available Balance: £ 15.00

Transfer Funds In: ●
Transfer Funds Out: ○

Amount to transfer (£):
Password:

Figure 56 The Use of the Main Betfair Balance and the Games Balances on the Betfair Games Web Site

11.4 Requesting XML That Lists Your Games Account Transactions (accountStatement)

The URL for requesting XML listing the transactions in your Betfair Exchange Games account is as follows:

https://api.games.betfair.com/rest/v1/account/statement?username=sampleuser

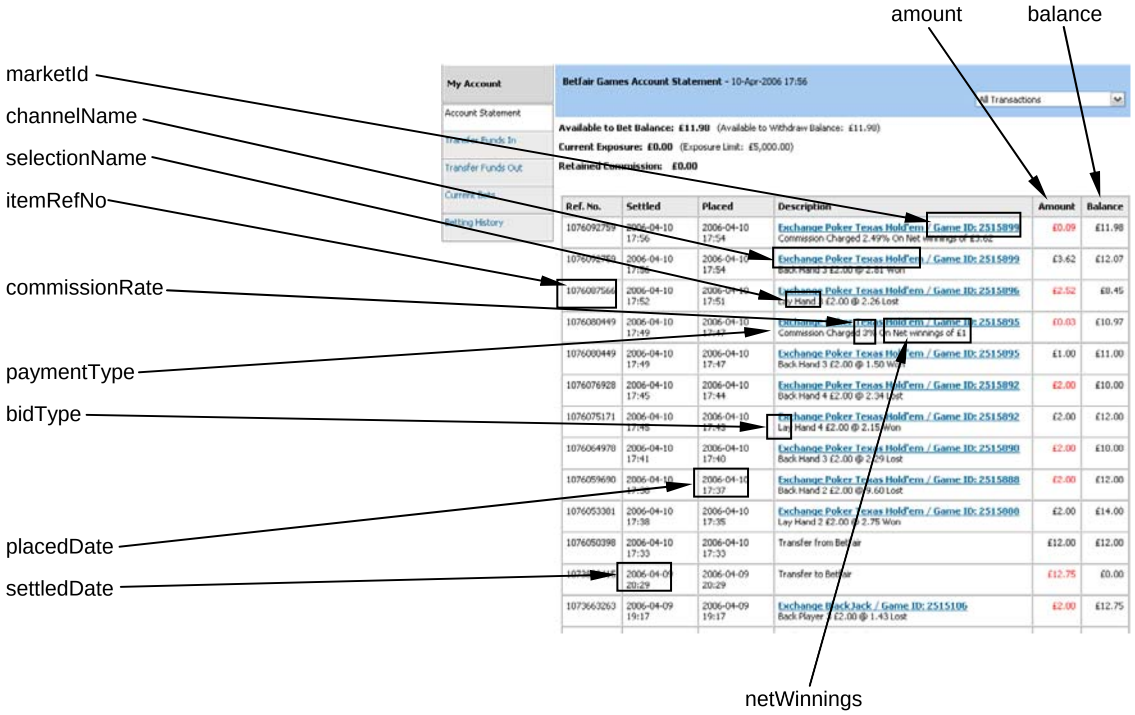
The call to this URL is an HTTP GET request. The name of the XML element returned is “accountStatement”.

The sample XML output in Figure 58 represents:

- first, a transaction into a sample Betfair Exchange Games account (in other words, a deposit),
- then a won bet,
- then a lost bet,
- and finally a commission payment.

The paging elements “nextPage” and “previousPage” (see Figure 58) contain a URL that enables you to navigate one page backward and one page forward, respectively (the sample output in Figure 58 is the first page of an account statement, therefore there is no previous page to view).

Figure 57 (below) shows a sample account statement as displayed by the Betfair Exchange Games web site. It also shows which data in the sample statement corresponds to particular sub-elements of a transaction “item” in the Games API’s “accountStatement” element.



Betfair Games Account Statement - 10-Apr-2006 17:56

Available to Bet Balance: £11.90 (Available to Withdraw Balance: £11.90)

Current Exposure: £0.00 (Exposure Limit: £5,000.00)

Retained Commission: £0.00

Ref. No.	Settled	Placed	Description	Amount	Balance
1076092759	2006-04-10 17:56	2006-04-10 17:54	Exchange Poker Texas Hold'em / Game ID: 2515899 Commission Charged 2.49% On Net winnings of £3.62	£0.09	£11.98
10760927?9	2006-04-10 17:56	2006-04-10 17:54	Exchange Poker Texas Hold'em / Game ID: 2515899 Back Hand 3 £2.00 @ 2.81 Won	£3.62	£12.07
1076087566	2006-04-10 17:52	2006-04-10 17:51	Exchange Poker Texas Hold'em / Game ID: 2515896 Lay Hand 3 £2.00 @ 2.26 Lost	£2.52	£8.45
1076080449	2006-04-10 17:49	2006-04-10 17:47	Exchange Poker Texas Hold'em / Game ID: 2515895 Commission Charged 3% On Net winnings of £1	£0.03	£10.97
1076080449	2006-04-10 17:49	2006-04-10 17:47	Exchange Poker Texas Hold'em / Game ID: 2515895 Back Hand 3 £2.00 @ 1.50 Won	£1.00	£11.00
1076076928	2006-04-10 17:45	2006-04-10 17:44	Exchange Poker Texas Hold'em / Game ID: 2515892 Back Hand 4 £2.00 @ 2.34 Lost	£2.00	£10.00
1076075171	2006-04-10 17:45	2006-04-10 17:43	Exchange Poker Texas Hold'em / Game ID: 2515892 Lay Hand 4 £2.00 @ 2.15 Won	£2.00	£12.00
1076064978	2006-04-10 17:41	2006-04-10 17:40	Exchange Poker Texas Hold'em / Game ID: 2515890 Back Hand 3 £2.00 @ 2.29 Lost	£2.00	£10.00
1076059690	2006-04-10 17:38	2006-04-10 17:37	Exchange Poker Texas Hold'em / Game ID: 2515888 Back Hand 2 £2.00 @ 9.60 Lost	£2.00	£12.00
1076053301	2006-04-10 17:38	2006-04-10 17:35	Exchange Poker Texas Hold'em / Game ID: 2515888 Lay Hand 2 £2.00 @ 2.75 Won	£2.00	£14.00
1076050398	2006-04-10 17:33	2006-04-10 17:33	Transfer from Betfair	£12.00	£12.00
[illegible]	2006-04-09 20:29	2006-04-09 20:29	Transfer to Betfair	£12.75	£0.00
1073663263	2006-04-09 19:17	2006-04-09 19:17	Exchange BlackJack / Game ID: 2515106 Back Player 3 £2.00 @ 1.43 Lost	£2.00	£12.75

Figure 57 A List of Transactions Belonging to a Sample User of the Betfair Exchange Games Web Site

```
 <accountStatement xmlns="urn:betfair:games:api:v1"
  currency="GBP" end="50" start="1" total="1114">
  <nextPage>
    <resource href="https://api.games.betfair.com/rest/v1/account/statement?username=sampleuser
     &amp;startRecord=51&amp;recordCount=50&amp;account=ALL" responseType="accountStatement"
    title="View the next 50 items in the statement" />
  </nextPage>
  <item refNo="794159438">
     <settledDate>2006-04-26T10:51:01+00:00</settledDate>
     <placedDate>2006-04-26T10:51:01+00:00</placedDate>
     <accountStatementDescription>
       <accountStatementTransferDescription>
         <paymentType>DEPOSIT</paymentType>
        </accountStatementTransferDescription>
     </accountStatementDescription>
     <amount>100.0</amount>
     <balance>900.2</balance>
  </item>
  <item refNo="794126966">
    <settledDate>2006-04-25T11:32:10+00:00</settledDate>
    <placedDate>2006-04-25T11:30:49+00:00</placedDate>
    <accountStatementDescription>
       <accountStatementBetDescription>
          <selectionReference>
              <marketReference>
                 <resource
href="https://api.games.betfair.com/rest/v1/channels/1444074/history?
                  marketId=2566351&amp;username=sampleuser" responseType="channelHistory"
                  title="Exchange Poker Texas Hold'em" />
                     <marketId>2566351</marketId>
                     <channelId>1444074</channelId>
                     <channelName>Exchange Poker Texas Hold'em</channelName>
              </marketReference>
              <selectionId>658441</selectionId>
              <selectionName>Hand 3</selectionName>
          </selectionReference>
          <size>10.0</size>
          <averagePrice>2.1</averagePrice>
          <winLose>WON</winLose>
          <bidType>LAY</bidType>
      </accountStatementBetDescription>
    </accountStatementDescription>
    <amount>10.0</amount>
    <balance>700.2</balance>
  </item>
  <item refNo="794126985">
     <settledDate>2006-04-25T11:32:10+00:00</settledDate>
     <placedDate>2006-04-25T11:31:31+00:00</placedDate>
     <accountStatementDescription>
       <accountStatementBetDescription>
          <selectionReference>
              <marketReference>
                  <resource
ref="https://api.games.betfair.com/rest/v1/channels/1444074/history?
                   marketId=2566351&amp;username=sampleuser" responseType="channelHistory"
                   title="Exchange Poker Texas Hold'em" />
                   <marketId>2566351</marketId>
                   <channelId>1444074</channelId>
                   <channelName>Exchange Poker Texas Hold'em</channelName>
              </marketReference>
              <selectionId>658440</selectionId>
              <selectionName>Hand 2</selectionName>
              </selectionReference>
              <size>10.0</size>
              <averagePrice>2.0</averagePrice>
              <winLose>LOST</winLose>
              <bidType>LAY</bidType>
       </accountStatementBetDescription>
     </accountStatementDescription>
     <amount>-10.0</amount>
     <balance>690.2</balance>
  </item>
  <item refNo="794169956">
     <settledDate>2006-05-03T08:34:21+00:00</settledDate>
     <placedDate>2006-05-03T08:32:46+00:00</placedDate>
     <accountStatementDescription>
       <accountStatementCommissionDescription>
        <marketReference>
            <resource href="https://api.games.betfair.com/rest/v1/channels/1444074/history?
              marketId=2576834&amp;username=sampleuser" responseType="channelHistory"
              title="Exchange BlackJack"/>
            <marketId>2576834</marketId>
            <channelId>1444074</channelId>
            <channelName>Exchange BlackJack</channelName>
          </marketReference>
          <paymentType>COMMISSION_CHARGED</paymentType>
          <commissionRate>1.5</commissionRate>
          <netWinnings>1000.0</netWinnings>
       </accountStatementCommissionDescription>
     </accountStatementDescription>
     <amount>-15.0</amount>
     <balance>150.0</balance>
  </item>
 </accountStatement>
```

Figure 58 Sample “accountStatement” Listing a User's Betfair Exchange Games Transactions

If you have won or lost bets that you made on a game that was for any reason voided, an element called “accountStatementGameStateDescription” will appear in the “accountStatement”, and it will contain the string “VOIDED”. When this element is present, it means that the game has been declared void and all bets made on it have also been made void.

11.5 Adding Parameters Manually to the URLs for Listing Transactions

To each URL that is returned in the “accountStatement” element, you can add other parameters manually. This section tells you what the parameters enable you to do and how to specify them. If you want to, you can use all of the parameters together.

To add a parameter, just put an ampersand character (&) at the end of the particular URL you want to use, and then specify the parameter. To add another parameter after this, add another ampersand character and specify the next parameter, and so on.

11.5.1 Limiting the Transactions List to a Particular Type of Game (accountStatement)

To each of the URLs for requesting the “accountStatement” element, you can add a parameter manually to to limit your list of transactions to those concerning a particular type of Exchange game. The parameter (including the necessary ampersand character) for this is:

`&account=`*`game`*

where *game* is one of the following values: `POKER`, `BLACKJACK`, `BACCARAT`, `HILO`, or `OMAHA_POKER`.

This parameter also enables you to list your fund transfers (see Section 11.5.2).

11.5.2 Listing Only Your Fund Transfers (accountStatement)

To each of the URLs for requesting the “accountStatement” element, you can add a parameter manually to list only those transactions that represent transfers of funds you have performed. The parameter (including the necessary ampersand character) for either of these options is:

`&account=TRANSFERS`

11.5.3 Requesting a Specific Number of Transactions (accountStatement)

To each of the URLs for requesting the “accountStatement” element, you can add a parameter manually to specify the number of transactions that you want the list returned to contain. The parameter (including the necessary ampersand mcharacter) for this is:

```
&recordCount=number
```

11.5.4 Starting the List at a Specific Position in the Overall List (accountStatement)

To each of the URLs for requesting the “accountStatement” element, you can add a parameter manually to specify the start position (that is, the start position in the Exchange Games API’s overall list) for the list of transactions that you want to receive. The parameter (including the necessary ampersand character) for this is:

```
&startRecord=number
```

11.6 Sending XML to Request a Transfer of Funds Between Accounts (postTransferOrder)

The URL for sending a request to transfer funds between your Betfair Exchange Games account and your main Betfair account is:

`https://api.games.betfair.com/rest/v1/account/transferOrder?username=sampleuser`

The call to this URL is an HTTP POST request. The name of the XML element you must send is called “postTransferOrder”.

```
<?xml version="1.0" encoding="UTF-8"?>
<postTransferOrder xmlns="urn:betfair:games:api:v1"
 xmlns:xs="http://www.w3.org/2001/XMLSchema-instance">
   <amount currency="GBP" action="TRANSFER">50</amount>
</postTransferOrder>
```

Figure *59* XML Showing the Format Required for Sending a Request to Transfer Funds

The name of the XML element returned is “responseTransferOrder” (see Figure 60).

If you specify a non-negative value for the amount (as in the example above, which specifies £50), then that amount of money will be transferred from your main Betfair account into your Betfair Exchange Games account.

If you specify a negative value, then that amount will be transferred from your Betfair Exchange Games account into your main Betfair account.

```
<responseTransferOrder xmlns="urn:betfair:games:api:v1">
   <message>
      SUCCESS
   </message>
</responseTransferOrder>
```

Figure 60 Sample XML Response from the Games API Server to a Request to Transfer Funds

The XML element returned by the Games API when you send a POST request to transfer funds is called a “responseTransferOrder”. This element contains a single element called “message”, and its content is a string of type “balanceResponseTypeEnum”. There are only three possible values it can have. They are:

- SUCCESS, which means that your balance transfer has been executed.

- INSUFFICIENT_FUNDS, which means that your balance transfer has not been executed, because you do not have sufficient funds in the source account to transfer the specified sum to the target account.
- FAILED, which means that your balance transfer has failed.

Chapter 12 Placing Bets Using the API

This chapter tells you how to use the Games API to place, edit, and cancel individual bets (see Section 12.1 and Section 12.2). It also tells you how to specify a total stake-size for a particular betting selection and odds combination (see Section 12.3).

The total stake-size facility enables you to send multiple duplicate bet requests (at or before the beginning of a betting round[16]) *for the precise sum you want to stake*, in the confidence that once one of the duplicate requests has been processed any others you have sent will be ignored. This means that, when you use the facility, the total stake-size for the selection and odds you specify will never exceed the sum you intend to stake however many times you send the duplicate request.

Before this facility was introduced, if you wanted to bet a certain total sum on a particular selection and at particular odds (while also maximising your chances of getting your bets placed early in the round), you needed to send multiple duplicate bet requests near to the beginning of the round, with each request representing *only a fraction of the sum you wanted to stake*. You then needed to monitor the matching of your bets and cancel any unmatched bets that took your total stake over the size you intended.

The total stake-size facility means that you no longer need to:

- Send multiple duplicate requests for amounts that are *smaller* than the sum you want to stake.
- Monitor your bets closely to see whether the sum matched exceeds the total you want to stake.
- Cancel unmatched bets to bring the total stake down to the size required.

For information about how to use the total-size facility, see Section 12.3.

Finally, this chapter tells you what the minimum bets you can place are, depending on the currency you are specifying your bets in (see Section 12.4), and what the valid odds and odds increments are for placing bets using the Games API (see Section 12.5).

12.1 Requesting XML for Managing Bets (betLanding)

The URL for requesting the XML resources that enable you to view and place bets is:

`https://api.games.betfair.com/rest/v1/bet?username=sampleuser`

The output returned when you request this information is longer than can be shown on a single page. Therefore for purposes of description it is dealt with in this chapter (and in the next chapter) in separate groups of the URLs it contains. The output returned is the “betLanding” XML element, which contains a list of “resource” elements, each containing a single URL. (The URL for placing bets appears at the bottom of the “betLanding” page. It is given in Section 12.2 below.)

12.2 Sending XML to Place, Edit, and/or Cancel Bets (postBetOrder)

The URL for sending XML when you want to place, edit, or cancel a bet appears at the bottom of the “betLanding” page. It is as follows:

`https://api.games.betfair.com/rest/v1/bet/order?username=sampleuser`

Because this URL is for sending a POST request, the “resource” element in the “betLanding” page that contains it includes the attribute “postType=postBetOrder”. The full contents of this “resource” element are shown in Figure 61. In it:

[16] This maximise your chances of getting your bets placed early in the round. And bets placed early in the betting round have a better chance of being matched than bets placed later.

- The ”postType” attribute indicates the name of the XML element (as defined in the schema) that you must send to the Games API to place, edit, or cancel bets (for the required format of this element, see Figure 62);
- The “responseType” attribute indicates the name of the XML element that the Games API will send back to you in response if you have sent the POST request correctly.

```
<resource href="https://api.games.betfair.com/rest/v1/bet/order?username=sampleuser"
 postType="postBetOrder" responseType="responseBetOrder"
 title="Place, cancel or edit a bet order"/>
```

Figure 61 The Contents of the "resource" Element (at the bottom of the “betLanding” page) for Betting

Figure 62 (below) contains sample XML requests for placing back and lay bets, editing a bet, and cancelling a bet. Your XML only needs to include those elements that are required for the particular request you are making: it needs one “betPlace” element for each bet you want to place, one “betEdit” element for each bet you want to edit, and one “cancel” element for each bet you want to cancel.

The exchange server performs the bet placement, edit, and cancel request instructions in exactly the order in which you are required to present the XML for them in the “postBetOrder” request. Before executing them, it checks the “postBetOrder” for a valid “marketId”, “round”, and “currency” value, and, if any of these are not valid, it rejects the entire “postBetOrder”. If the “postBetOrder” passes this initial test, however, the exchange server performs each of your instructions consecutively and independently of each other. This means that, if any one of them fails for any reason, the others before and after it in the “postBetOrder” are not affected. (The Betfair Games exchange server differs in this regard from the Betfair Sports exchange server.)

NOTE: If you need to cancel a bet *before* another bet gets placed (for example, because you will not have sufficient funds to cover the bet placement until the cancellation has taken effect), you must submit two separate “postBetOrder” requests, the first one cancelling the bet you want cancelled and the second one placing the bet you want to place. (If you put both requests in the same “postBetOrder”, the bet placement would not be successful unless sufficient funds to cover it were available before the cancellation.)

```
<?xml version="1.0" encoding="UTF-8"?>
<postBetOrder  xmlns="urn:betfair:games:api:v1"
  marketId="2568540" round="1" currency="GBP">
  <betPlace>
     <bidType>BACK</bidType>
     <price>2.5</price>
     <size>2.50</size>
     <selectionId>658440</selectionId>
  </betPlace>
  <betPlace>
     <bidType>LAY</bidType>
     <price>2.8</price>
     <size>5.50</size>
     <selectionId>658441</selectionId>
  </betPlace>
  <betEdit>
     <betId>794162680</betId>
     <price>2.5</price>
     <unmatchedSize>2.50</unmatchedSize>
     <bidType>BACK</bidType>
     <newPrice>8.6</newPrice>
     <newUnmatchedSize>8.50</newUnmatchedSize>
     <selectionId>658440</selectionId>
  </betEdit>
  <betCancel>
     <betId>794162675</betId>
  </betCancel>
</postBetOrder>
```

Figure 62 Sample XML for Sending a Request to Place Two Bets, Edit One, and Cancel Another

NOTE: From Figure 62 you can see that each “postBetOrder” is specific to a single betting market: you must specify the marketId as one of the attributes of the “postBetOrder”. In other words, in a single “postBetOrder” you can only place, edit, and cancel bets in a particular betting market.

```
 <?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<responseBetOrder xmlns="urn:betfair:games:api:v1">
     <betPlacementResult>
         <betId>794162685</betId>
         <resultCode>OK</resultCode>
     </betPlacementResult>

     <betPlacementResult>
          <betId>794162686</betId>
          <resultCode>OK</resultCode>
     </betPlacementResult>

     <betEditResult><betId>794162680</betId>
          <resultCode>OK</resultCode>
     </betEditResult>

     <betCancelResult>
          <resultCode>REMAINING_CANCELLED</resultCode>
     </betCancelResult>
</responseBetOrder>
```

Figure 63 *Sample XML Response to a Request to Place Two Bets, Edit One, and Cancel Another*

Figure 63 shows the format of the Games API’s response to the sample POST request in Figure 62.

12.3 Sending XML to Make Total-size Bet Requests (postBetOrder)

The previous section told you how to use the “postBetOrder” element to place, edit, and/or cancel individual bets. This section tells you how to use the “postBetOrder” element, in addition, to make total-size bet requests.

The total-size request facility has been designed for customers who send bursts of duplicate betting requests at or around the time that a betting window opens (in other words, when the current betting round for an Exchange game starts[17]). Customers who do this know that some of their requests will be rejected because they will arrive before the betting window has opened. But not knowing, therefore, how many will be accepted, and wanting to bet a certain total sum on or against a particular selection at specified odds (or better), they send bursts of duplicate bet requests, each one for a fraction of the total sum they want to stake. Then, when all their bet requests have been sent and some rejected but others processed and matched, they proceed (if necessary) to try to cancel any bets that remain unmatched so that they can get their stake down to the level they initially intended.

As explained in the introduction to this chapter, the total-size request facility means that you no longer need to place multiple duplicate bets, each representing a fraction of your required stake, only to cancel unmatched bets later to get your total stake down to the required size.

Instead you can now either:

- Send a single total-size request when the betting window has opened. The exchange server will then automatically place bets for you on the selection and at the odds you have specified (or at better odds if available) up to the value you specify in the total-size request.
- Or you can send multiple duplicate total-size requests at or before the time the betting window opens (in the same way as you would previously have sent multiple duplicate bet requests). If you do this the exchange server places bets for you only in response to the *first* instance of the total-size request that it receives after the betting window has opened (that is, after the betting market has opened). Any duplicate total-size requests that it receives after it has processed the first one are simply rejected[18].

NOTE: Total-size requests are betting-round-specific. This means that any unmatched bets that have been placed as a result of a total-size request will automatically be lapsed at the end of the betting round. It also means that any bets that were matched in a previous betting round will be ignored by the exchange server when it processes a new total-size request; this is so even if one or more bets matched in a previous round were matched for the same selection and odds as specified in the new total-size request.

For more explanation of and commentary on this facility, read the remainder of Section 12.3.

To use the total-size request facility, you need to include an element called “totalSizeRequest” in the “postBetOrder” that you send. The URL to send the “postBetOrder” to is the one that was provided in Section 12.2:

`https://api.games.betfair.com/rest/v1/bet/order?username=`*`username`*

For more information about sending “postBetOrder” requests, see Section 12.2.

For sample POST requests containing the “totalSizeRequest” element, see Figure 64 and Figure 68.

NOTE: For a very important warning about laying selections using the total-size request facility, see Section 12.3.7.

[17] Bets placed early in the betting round have the best chance of being matched.

[18] Total-size bet requests can therefore be described technically as “idempotent”.

12.3.1 Adjusting Your Stake after Sending a Total-size Request

If you want to adjust your total stake on the same selection and bid-type[19] and at the same odds as you had specified in a previous bet or total-size request during the same betting round, you can send a further total-size request specifying a new total-size for your stake.

If the new total size you request is:

- Higher than the previously requested one, then the exchange server places a new bet to increase your total stake to the size specified in the new request.
- Lower than the previously requested one, then the exchange server cancels any unmatched bets or unmatched portions of bets that have already been placed for you (for the specified selection, bid-type, and odds) until your total stake is reduced to (or as far as possible towards) the size specified in the new request. Where necessary, the exchange server will place a smaller bet than the one cancelled in order to adjust your stake to the precise size requested.

12.3.2 How the Total-size Request Facility Works

The exchange server's response to a total-size request depends on the circumstances. If your current stake for the specified selection, bid-type, and odds is:

- Already at the level requested in the latest total-size request, then no action is taken by the exchange server[20]. Therefore, if you send multiple duplicate requests, all but the first to be received will be ignored.
- Below the requested level, the exchange server will place an additional bet of the size necessary to bring it up to the requested level.
- Above the requested level, the exchange server cancels any unmatched bets or unmatched portions of bets that have already been placed by or for you (for that selection, bid-type, and odds) until your total stake is reduced to (or as far as possible towards) the requested level.

12.3.3 Mixing Bet and Total-size Requests in a Single "postBetOrder"

You can include a "totalSizeRequest" in a "postBetOrder" containing other instructions (as described in Section 12.2).

However, we do not recommend you mix total-size requests and bet placement instructions *for the same selection, bid-type, and odds* in the same "postBetOrder". The consequences of doing this can be difficult to predict.

Remember that the XML of a "postBetOrder" requires the bet placement instructions to appear first. This means that bet placement instructions are therefore processed first. And this in turn means that a total-size request affecting the same selection, bid-type, and odds as a bet placement instruction can be compromised by the bet placement instruction.

For example, if the bet got matched before the total-size request was processed, then the total-size request would be unable to reduce the stake (if that was what its purpose had been). On the other hand, if the bet did not get matched before the total-size request was processed, then the total size request would simply render the bet placement request redundant.

So it is not good practice to include bet placement instructions and total-size requests for the same selection, bid-type, and odds in a single "postBetOrder".

[19] Bid-type means whether the total-size request is to *back* or *lay* the specified selection.

[20] However, read Section 12.3.6 for important information about how the exchange server calculates and defines your current stake.

12.3.4 Sending Bet and Total-size Requests Close Together in Time

You can also obviously include bet requests and total-size requests in separate “postBetOrder” messages that nevertheless are transmitted very close together in time.

However, we do not recommend you send total-size requests and bet placement instructions *for the same selection, bid-type, and odds* in separate “postBetOrder” messages very close together in time. As described in the previous section, the consequences of doing this can be difficult to predict.

If you send a total-size request for a specified selection, bid-type, and odds immediately before or immediately after a bet placement request for the same selection, bid-type, and odds, then the outcome will depend upon the order in which the exchange server receives and processes the two requests.

For example, if you send a total-size request for £100 and a bet placement request for £50 (for the same selection, bid-type, and odds) in “postBetOrder” requests that are transmitted separately but very close together in time, then the outcome after both have been processed will either be:

a) A total stake of £150 if the total-size request was processed first and the bet placement request second, or

b) A total stake of £100 if the bet placement request was processed first and the total-size request second.

The behaviour in the first case is as follows: the exchange server receives a total-size request for £100 for a specified selection, bid-type, and odds. It processes this and therefore places a bet for £100 as specified. The exchange server then receives the bet placement request for £50 (for the same selection, bid-type, and odds); it processes this and therefore places a bet for £50 as specified. The outcome is a total stake of £150.

The behaviour in the second case is as follows: the exchange server receives a bet placement request for £50 for a specified selection, bid-type, and odds. It processes this and places the bet for £50, as specified in the request. The exchange server then receives the total-size request for £100 for the same selection, bid-type, and odds; it processes this, and, because there is already a stake of £50 on this selection (with the same bid-type and odds as specified in the total-size request), the exchange server places a bet of £50 to bring the stake up to the requested size of £100. The outcome in this case is therefore a total stake of £100.

NOTE: “postBetOrder” requests will not necessarily be processed in the order that you send them. They might be routed differently across the internet, with the result that one request sent later than another arrives at Betfair earlier than the other (the first request having incurred network latency that did not affect the second). And they might be processed by different servers at Betfair as a result of load balancing mechanisms.

12.3.5 The Application of Betfair’s “Best Execution” Policy to Total-size Requests

If you send a total-size request (for a specified selection, bid-type, and odds) and there are bets with better odds available to be matched at than you have requested, then the exchange server will automatically match bets for you at the better odds (up to the value of the sum available to be matched or up to the size of your requested total stake-size, whichever is the lower figure).

The exchange server does this because it interprets your request for particular odds as a request for those odds *or better*.

If you submit a total-size request for the same selection and bid-type as an earlier request that was partially matched at better odds than requested, but in the new request you specify the improved odds in your request, then the exchange server will process the new request without taking account of the stake you already have at those odds. This is because the existing stake resulted from a previous and distinct total-size request that specified different (lower) odds (see next section).

12.3.6 How the Exchange Server Calculates the Existing Stake-size

When the exchange server processes a total-size request, it analyses existing bets (for the specified selection, bid-type, and odds) to discover whether any already existing stake needs to be included in the calculation of the new total stake to be established for the selection, bid-type, and odds.

In this analysis, for each pre-existing bet (whether the bet was placed as the result of a bet placement request, a total-size request, or a bet edit request) the exchange server uses the *requested* price (odds) to identify whether the bet forms part of the current stake to be adjusted. In other words, it ignores the *matched* price when it makes its calculation.

Therefore, any stake you already have (for a given selection and bid-type) that results from a bet *matched* at better odds than you specifically requested will be ignored by the exchange server when it sets your stake in response to a total-size request that specifies those odds for the same selection and bid-type.

See Section 12.3.8.2 for an example of how the Games exchange server handles a total-size request in which the odds specified have been increased in relation to an earlier total-size request concerning the same selection and bid-type.

12.3.7 Using a Total-size Request to Lay (Instead of Back) a Selection

It is very important to keep in mind the semantics of lay betting when you place a total-size request to lay a selection.

The total stake-size you request for a lay bet specifies the size of the *backer's* stake that you are willing to lay. It does NOT specify the size of the total liability you are willing to incur. As with any lay bet, your liability is the product of the following formula:

Backer's stake x (price – 1) = layer's liability

Therefore, for example, if you submitted a total-size request of £500 to lay a betting selection at odds of 3.98, the liability you would incur would be £1490 calculated (according to this formula) as follows:

£500 x (3.98 – 1) = £1490

NOTE: It is only in the case of back bets that the stake-size and the liability are the same. For lay bets where the odds are higher than 2.0 the liability is always higher than the stake-size, because it is a multiple of the stake size.

12.3.8 Total-size Request Examples and Scenarios

This section contains sample XML for sending a total-size request, and sample scenarios that help to explain the behaviour of the Games exchange server in response to total-size requests. The scenarios will also help you understand how to use the total-size request facility.

12.3.8.1 Sample XML for Sending a Total-size Request

In Figure 64 there is a sample "postBetOrder" containing a total-size request. As you can see, the "postBetOrder" contains the element "totalSizeRequest", and this in turn contains the elements:

- "bidType" (which indicates whether the request is intended to back or lay the selection),
- "price" (which specifies the odds at which the stake is requested),
- "size" (which specifies the sum of money that you want set as your total stake),
- "selectionId" (which specifies the betting selection to which the request applies).

```
<?xml version="1.0" encoding="UTF-8"?>
<postBetOrder  xmlns="urn:betfair:games:api:v1"
  marketId="2568540" round="1" currency="GBP">

  <totalSizeRequest>
     <bidType>BACK</bidType>
     <price>2.5</place>
     <totalSize>150</totalSize>
     <slectionId>658443</selectionId>
  </totalSizeRequest>

</postBetOrder>
```

Figure 64 Sample XML for Submitting a Total-size Request

Figure 65 shows the format of the Games API’s response to the sample POST request in Figure 64.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<responseBetOrder xmlns="urn:betfair:games:api:v1">
     <totalSizeBetResult>
          <selectionId>658443</selectionId>
          <resultCode>OK</resultCode>
     </totalSizeBetResult>

</responseBetOrder>
```

Figure 65 Sample Response from the Games API to a Total-size Request

The effect of this request is to place a bet with a stake of £150 on the specified selection and at odds of 2.5. The API’s response of OK means that the total-size request has been accepted. The Games API does not report the bet or bets that have been placed in response to a total-size request. To monitor the bets that have been placed and matched in response to a total-size request, you must use the API facilities described in Section 12.3.

12.3.8.2 Scenario in which a Client Application Varies the Specified Stake-size

For the purposes of this scenario, certain assumptions are made as to the context in which the sample total-size requests are submitted. These assumptions are that:

- £50 is available to be matched at odds of 3.5.
- £150 is available to be matched at odds of 3.0.
- No other funds will enter the market between the first and the last total-size requests in the example.

In these hypothetical circumstances, then, suppose further that the user of a client application has no existing bets on a particular selection and that the client application submits a total-size request to place bets backing that selection to the value of £300 at odds of 3.0 (or better).

The exchange server will then place bets for the client application to this value and at the specified or better odds. Of the £300, therefore, £150 will be matched at the requested odds of 3.0, £50 will be matched at the better odds of 3.5, and £100 will remain unmatched. At this point, therefore, the user’s total stake-size and list of current bets for the specified selection, bid-type, and odds, will be as follows:

Bet 1: Matched bet of £50 @ 3.5
Bet 2: Matched bet of £150 @ 3.0
Bet 3: Unmatched bet of £100 @ 3.0
Total stake-size @ 3.0 (or better) = £300

If the client application now submits a second total-size request in the same betting round, specifying the same selection, bid-type, and odds, but this time specifying a total stake-size of £250, then the exchange server will act to reduce the user’s total stake-size (for the selection, bid-type, and odds) from £300 to £250. It will do this by editing the unmatched bet for £100 (Bet 3 in the list above) and reducing it to £50. At this point, therefore, the user’s total stake-size and list of current bets for the particular selection, bid-type, and odds, will be as follows:

Bet 1: Matched bet of £50 @ 3.5
Bet 2: Matched bet of £150 @ 3.0
Bet 3: Unmatched bet of £50 @ 3.0 (EDITED)
Total stake-size @ 3.0 (or better) = £250

If the client application next submits a third total-size request, specifying the same selection, bid-type, and odds, but this time specifying a total stake-size of £200, then the exchange server will act to reduce the total stake-size (for the selection, bid-type, and odds), this time from £250 to £200. It will do this by cancelling the unmatched bet of £50 at odds of 3.0 (Bet 3 in the list above). At this point, therefore, the user's total stake-size and list of current bets for the particular selection, bid-type, and odds, will be as follows:

Bet 1: Matched bet of £50 @ 3.5
Bet 2: Matched bet of £150 @ 3.0
~~Bet 3 - Unmatched bet of £50 @ 3.0~~ (CANCELLED)
Total stake-size @ 3.0 (or better) = £200

Finally, if the client application now submits a fourth total-size request, specifying the same selection, bid-type, and odds, but this time specifying a total stake-size of £400, then the exchange server will act to increase the total stake-size (for the selection, bid-type, and odds) from £200 to £400. It will do this by placing a new bet for £200 at odds of 3.0. At this point, therefore, the user's total stake-size and list of current bets for the particular selection, bid-type, and odds, will be as follows:

Bet 1: Matched bet of £50 @ 3.5
Bet 2: Matched bet of £150 @ 3.0

Bet 4 - Unmatched bet of £200 @ 3.0
Total stake-size @ 3.0 (or better) = £400

12.3.8.3 Scenario in which a Client Application Increases Both the Stake-size and Odds

For the purposes of this scenario, the same assumptions are made as in the previous example, the client application has sent the same first total-size request, and the exchange server has made the same bet placement in response. After this request has been processed, therefore, the user's total stake-size and list of current bets for the specified selection, bid-type, and odds, are the same as they were at the same point in our previous scenario:

Bet 1: Matched bet of £50 @ 3.5
Bet 2: Matched bet of £150 @ 3.0
Bet 3: Unmatched bet of £100 @ 3.0
Total stake-size @ 3.0 (or better) = £300

Suppose now that the client application submits a second request to set the user's total stake-size (for the same selection and bid-type as in the previous request) to £400, but that this time it specifies odds of 3.5 (or better).

In this case, the exchange server will entirely disregard the already existing stake of £50 at those odds (that was achieved by Bet 1), because this was the result of an earlier total-size request that had specified odds of only 3.0 (or better). In this situation the exchange server ignores any existing stake on the same selection that was not explicitly requested at the specified price (see Section 12.3.6). In response to this new request, the exchange server will simply place a bet for £400 at odds of 3.5.

At this point, therefore, the client application's total stake-size and list of current bets as a result of its two total-size requests will be as follows:

Bet 1: Matched bet of £50 @ 3.5
Bet 2: Matched bet of £150 @ 3.0
Bet 3: Unmatched bet of £100 @ 3.0
Total stake-size @ 3.0 (or better) = £300

Bet 4: Unmatched bet of £400 @ 3.5
Total stake-size @ 3.5 (or better) = £400

As you can see, although the second request (resulting in Bet 4) was for a total stake-size at odds for which the user of the client application already had a pre-existing stake, the exchange server took no account of this in its automatic betting decision. The outcome in this example was a de facto stake-size (including the already existing stake) of £450 at the odds specified in the second request, which is £50 more than was specified in that request.

Table 8 Summary of the Total Stake-sizes Requested and of the Sums Matched in this Scenario

	Total-size requested @ 3.0	Sum Matched @ 3.0	Total-size requested @ 3.5	Sum Matched @ 3.5	Sum Unmatched
Total-size request 1	300	150	▨	50	100
Total-size request 2	▨	▨	400	▨	400
Total Stake	300	▨	400	▨	▨

12.3.8.4 Scenario in which a Request to Reduce the Stake-size is Partially Successful

For the purposes of this scenario, the following assumptions are made:

- £100 is available to be matched at odds of 3.0.
- A further £50 enters the market between the first and the second total-size requests in the example.

Suppose that, in these hypothetical circumstances, a client application sends a total-size request for a total stake-size of £200, specifying odds of 3.0 (or better).

In response to this request, the exchange server will place a single bet for £200 on behalf of the user of the client application at odds of 3.0.

Part of this bet will be matched at the requested price (because £100 is available to be matched at those odds).

Therefore, at this point, the user's total stake-size and list of current bets as a result of the total-size request will be as follows:

Bet 1: Matched bet of £100 @ 3.0
Bet 2: Unmatched bet of £100 @ 3.0
Total stake-size @ 3.0 (or better) = £200

If the further £50 (assumed above) now enters the market, the exchange server will match half of the user's unmatched £100 bet with the new £50, leaving only £50 still unmatched. At this point, therefore, the user's total stake-size and list of current bets will be as follows:

Bet 1: Matched bet of £100 @ 3.0
Bet 2: Matched bet of £50 @ 3.0
Bet 3: Unmatched bet of £50 @ 3.0
Total stake-size @ 3.0 (or better) = £200

The total stake-size will therefore remain at £200, but there will now be £150 of it matched and only £50 unmatched.

However, if the client application now attempts to reduce its user's total stake for the selection, bid-type, and odds by submitting a second total-size request to the value of £100, then the exchange server will only be able cancel the unmatched bet of £50, leaving the user's total stake at £150, because £150 of the original £200 has already been matched. Therefore the API will return an error (see Figure 66) explaining that the request was only partially successful, and at this point the user’s total stake-size and list of current bets will be as follows:

Bet 1: Matched bet of £100 @ 3.0
Bet 2: Matched bet of £50 @ 3.0
~~Bet 3: Unmatched bet of £50 @ 3.0~~ (CANCELLED)
Total stake-size @ 3.0 (or better) = £150

In this scenario, therefore, the second total-size request, which sought to reduce the user’s total stake-size from £200 to £100, was only partially successful because by the time it was received £50 of the user’s previously unmatched £100 bet had already been matched (leaving an unmatched bet of only £50 available to be cancelled).

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

  <responseBetOrder xmlns="urn:betfair:games:api:v1">
        <totalSizeBetResult>
              <selectionId>658439</selectionId>
              <resultCode>
                  COULD_NOT_REDUCE_TO_TARGET_SIZE_REMAINING_UNMATCHED_EDITED_TO_ZERO
              </resultCode>
        </totalSizeBetResult>
  </responseBetOrder>
```

Figure 66 Sample Response Where Only Part of the Requested Stake-size Reduction Could be Achieved

12.3.8.5 Scenario in which a Client Application Fails to Reduce the Stake-size

For the purposes of this scenario, the following assumptions are made:

- £200 is available to be matched at odds of 3.0.
- No other funds will enter the market between the first and the last total-size requests in the example.

Suppose that, in these hypothetical circumstances, a client application sends a total-size request to the value of £200, specifying odds of 3.0 (or better).

In response to this request, the exchange server will place a single bet for the client of £200 at the odds of 3.0. This bet will be matched at the requested price (because £200 is available to be matched at those odds). Therefore, at this point, the user’s total stake-size and list of current bet as a result of the total-size request will be as follows:

Bet 1: Matched bet of £200 @ 3.0
Total stake-size @ 3.0 (or better) = £200

If the client application then attempts to reduce the user’s total stake by submitting a second total-size request to the value of £150 and specifying the same odds (and bid-type and selection), the exchange server will not be able to edit or cancel any bets to reduce the stake because the bet for the entire £200 has already been matched. Therefore it will return an error (see Figure 67) explaining that there are no unmatched bets available to be edited or cancelled and that the total stake-size cannot be reduced.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<responseBetOrder xmlns="urn:betfair:games:api:v1">
         <totalSizeBetResult>

               <selectionId>658439</selectionId>
               <resultCode>NO_UNMATCHED_BETS_TO_REDUCE_TOTAL_SIZE</resultCode>
         </totalSizeBetResult>
   </responseBetOrder>
```

Figure 67 Sample Response Where None of the Requested Stake-size Reduction Could be Achieved

12.3.9 Sending XML Containing a Mixture of Betting Instructions

It is possible to send a “postBetOrder” that contains assorted bet operation requests (placement, edit, total-size and/or cancellation requests, as in Figure 68 below) to be sent. However it may be only partially completed before the betting window for a round of the game you are betting on closes. Try to avoid sending “postBetOrder” requests containing multiple operations when the betting window is about to close. (For information about the “bettingWindow” and “bettingWindowPercentageComplete” elements, refer to Section 4.2 and Section 4.4.)

```
<?xml version="1.0" encoding="UTF-8"?>
<postBetOrder  xmlns="urn:betfair:games:api:v1"

  marketId="2568540" round="1" currency="GBP">

  <betPlace>
     <bidType>BACK</bidType>
     <price>2.5</price>
     <size>2.50</size>
     <selectionId>658440</selectionId>
  </betPlace>

  <betPlace>
     <bidType>LAY</bidType>
     <price>2.8</price>
     <size>5.50</size>
     <selectionId>658441</selectionId>
  </betPlace>

  <betEdit>
     <betId>794162680</betId>
     <price>2.5</price>
     <unmatchedSize>2.50</unmatchedSize>
     <bidType>BACK</bidType>
     <newPrice>8.6</newPrice>
     <newUnmatchedSize>8.50</newUnmatchedSize>
     <selectionId>658440</selectionId>
  </betEdit>

  <totalSizeRequest>
     <bidType>BACK</bidType>
     <price>2.5</place>
     <totalSize>150</totalSize>
     <slectionId>658443</selectionId>
  </totalSizeRequest>

  <betCancel>
     <betId>794162675</betId>
  </betCancel>
</postBetOrder>
```

Figure 68 Sample "postBetOrder" Request Containing Multiple Betting Instructions

NOTE: You can make a maximum of 100 bet placements, a maximum of 100 bet edits, a maximum of 100 total-size requests, and a maximum of 100 bet cancellations in any one “postBetOrder” request. Therefore your “postBetOrder” can contain a maximum of 400 betting instructions.

NOTE: As stated in 12.3.3, we do not recommend you mix bet placement requests with total-size requests concerning the same selection, bid-type, and odds in a single “postBetOrder” request.

12.4 The Minimum Bets for Different Currencies

There is a minimum amount that you can bet when you use the Exchange Games API. The minimum amount is given for various currencies in Table 9.

Table 9 The Minimum Bets for Different Currencies

Currency	Abbreviation	Minimum Bet
Danish Kroner	DKK	30
US Dollar	USD	4
UK Sterling	GBP	2
Australian Dollar	AUD	6
Euro	EUR	2
Canadian Dollar	CAD	6
Hong Kong Dollars	HKD	25
Norwegian Kronor	NOK	30
Singapore Dollars	SGD	6
Swedish Krona	SEK	30

12.5 The Odds at which You Can Place and Edit Bets

The Betfair Exchange Games API imposes certain limits on the odds at which you can place and edit bets. You cannot bet at odds of less than 1.01 or at odds of greater than 1000.00. You can bet at any odds of between 1.01 and 2.99 or edit a bet to new odds of any figure (with two decimal places) between 1.01 and 2.99.

However, if you want to bet (or edit a bet) at odds of between 3.00 and 5.98, you must specify odds in increments of 0.02 (so, for example, odds of 5.99 would not be permitted to you).

Similarly, if you want to bet (or edit a bet) at odds of between 6.00 and 9.95, you must specify odds in increments of 0.05 (so, for example, odds of 9.97 would not be permitted to you).

For a full list of the increments you must use to specify odds within different ranges, see Table 10.

Table 10 Odds Ladder

Odds range	**Valid increments for odds within different ranges**
0 – 1.00	You cannot bet at odds of less than 1.01.
1.01 – 2.00	0.01
2.00 – 3.00	0.02
3.00 – 5.00	0.05
5.00 – 10.00	0.10
10.00 – 20.00	0.20
20.00 – 30.00	0.50
30.00 – 50.00	1.00
50.00 – 100.00	2.00
100.00 – 500.00	5.00
500.00-1000.00	10.00
Over 1000.00	You cannot bet at odds of greater than 1000.00.

NOTE: The information in the above table does not apply to bets placed in a variable handicap market. For the entire odds range of 1.01 – 999.0 the valid increment is 1.01.

Chapter 13 Viewing Your Current and Past Betting Activity

This chapter tells you which URLs to use to view current and past bets.

13.1 Requesting XML for Viewing Bets (betLanding)

The URL for requesting XML that enables you to list your current and past bets (sorted into various categories) is the same as that which enables you to place bets:

`https://api.games.betfair.com/rest/v1/bet?username=sampleuser`

The output returned when you request this information is longer than can be shown on a single page. Therefore for purposes of description it will be dealt with in this chapter (and in the previous chapter) in separate groups of the URLs it contains. The output returned is the “betLanding” XML element, which contains a list of “resource” elements, each containing a single URL.

13.2 Requesting XML That Lists Current Bets by Status (betSnapshot)

The first group of URLs returned by the HTTP request for the “betLanding” element is a list of URLs for requesting the “betSnapshot” service.

You cannot request to view all current bets at once. You can only request to view current bets of a specific status. The possible statuses are: MATCHED, UNMATCHED, LAPSED, AND CANCELLED.

The URLs in the “betLanding” element for viewing current bets according to their status are as follows:

`https://api.games.betfair.com/rest/v1/bet/snapshot?username=sampleuser&betStatus=MATCHED`

`https://api.games.betfair.com/rest/v1/bet/snapshot?username=sampleuser&betStatus=UNMATCHED`

`https://api.games.betfair.com/rest/v1/bet/snapshot?username=sampleuser&betStatus=LAPSED`

`https://api.games.betfair.com/rest/v1/bet/snapshot?username=sampleuser&betStatus=CANCELLED`

Figure 69 shows a sample response from the Games API to a request for a list of current matched bets.

By default there will be a maximum of 50 bet records per page of the “betSnapshot” element. However, you can specify a lower number by including the `&recordCount` parameter in the URL when you request a “betSnapshot” (see Section 13.4.2).

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<betSnapshot currency="GBP" end="15" start="1" total="15" xmlns="urn:betfair:games:api:v1">
   <nextResultSet><resource href="https://api.games.betfair.com/rest/v1/bet/snapshot?
    username=sampleuser&startRecord=1&recordCount=50&after=1158045715470&betStatus=MATCHED"
    responseType="betSnapshot" title="Next Bet Snapshot Results Set"/> </nextResultSet>
    <betSnapshotItem>
        <betId>794201196</betId>
        <bidType>BACK</bidType>
        <placedDate>2006-05-09T10:49:08+00:00</placedDate>
        <matchedDate>2006-05-09T10:49:08+00:00</matchedDate>
        <price>4.0</price>
        <size>817.0</size>
        <selectionReference>
            <marketReference>
                <resource
                  href="https://api.games.betfair.com/rest/v1/channel/1444074/snapshot?
                  username=sampleuser" responseType="channelSnapshot"
                  title="Exchange Poker Texas Hold'em"/>
                <marketId>2594323</marketId>
                <channelId>1444074</channelId>
                <channelName>Exchange Poker Texas Hold'em</channelName>
                <gameStartDate>2006-05-09T10:47:32+00:00</gameStartDate>
            </marketReference>
            <selectionId>658440</selectionId>
            <selectionName>Hand 2</selectionName>
        </selectionReference>
        <averagePriceMatched>4.0</averagePriceMatched>
        <updateStamp>1158045715462</updateStamp>
    </betSnapshotItem>
    <betSnapshotItem>
        <betId>794201198</betId>
        <bidType>BACK</bidType>
        <placedDate>2006-05-09T10:49:08+00:00</placedDate>
        <matchedDate>2006-05-09T10:49:08+00:00</matchedDate>
        <price>2.0</price>
        <size>762.0</size>
        <selectionReference>
            <marketReference>
                <resource href="https://api.games.betfair.com/rest/v1/channel/1444074/snapshot?
                  username=sampleuser" responseType="channelSnapshot" title="Exchange Poker
                  Texas Hold'em"/>
                <marketId>2594323</marketId>
                <channelId>1444074</channelId>
                <channelName>Exchange Poker Texas Hold'em</channelName>
                <gameStartDate>2006-05-09T10:47:32+00:00</gameStartDate>
            </marketReference>
            <selectionId>658441</selectionId>
            <selectionName>Hand 3</selectionName>
        </selectionReference>
        <averagePriceMatched>2.0</averagePriceMatched>
        <updateStamp>1158045715470</updateStamp>
    </betSnapshotItem>
</betSnapshot>
```

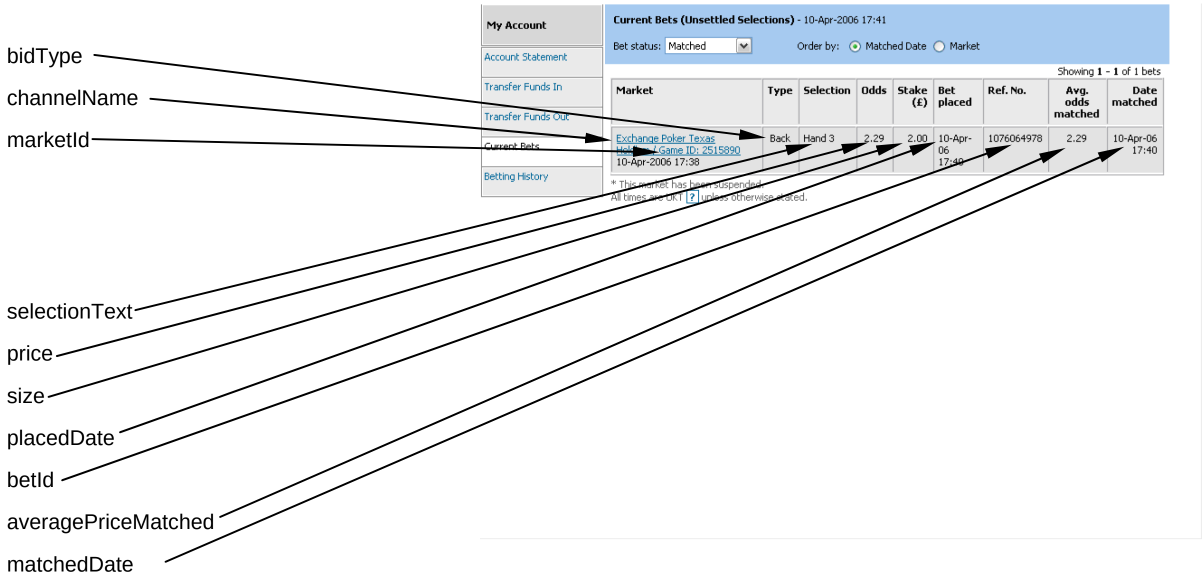
Figure 69 Listing of a Sample User's (Two) Currently Matched Bets (Sample Output of "betSnapshot")

If you examine Figure 69 you can see that, at the top of the XML page there is an element called "nextResultSet". This contains a URL that enables you to view details of any bets that have been updated since the "betSnapshot" page you are viewing was sent. (The type of updates listed is determined by the `betStatus` parameter in your original "betSnapshot" request.) This URL includes the `after` parameter, whose value is the highest "updateStamp" number in the "betSnapshot" page you are viewing (see Section 13.2.1).

Finally, in Figure 69 you can see that for each bet record returned when you request a list of your current bets, you are provided with a URL that enables you to view the game that you have bet on (that URL returns the "channelShapshot" element, described in Section 4.2).

NOTE: The information in a “betSnapshot” is time-sensitive, and the URL enabling you to view a “channelSnapshot” for a game that you have bet on will only return the details of that game if it is still the current game. If the game has finished and a new one has started, you need to use the “channelHistory” element to view details of the game you bet on (see Section 4.3).

Figure 70 shows how data corresponding to the content of the “betSnapshot” element is represented on the Betfair Exchange Games website.



Current Bets (Unsettled Selections) - 10-Apr-2006 17:41

Bet status: Matched

Order by: ● Matched Date ○ Market

Showing **1** - **1** of 1 bets

Market	Type	Selection	Odds	Stake (£)	Bet placed	Ref. No.	Avg. odds matched	Date matched
Exchange Poker Texas Hold'em / Game ID: 2515890 10-Apr-2006 17:38	Back	Hand 3	2.29	2.00	10-Apr-06 17:40	1076064978	2.29	10-Apr-06 17:40

* This market has been suspended.
All times are UKT ? unless otherwise stated.

Figure *70* Sample Screen from the Betfair Games Web Site, Illustrating the betSnapshot Element

13.2.1 Listing Only Those Current Bets Whose Details Have Changed (betSnapshot)

In URLs for requesting the “betSnapshot” element, you can include the `after` parameter to view only those current bets (of a specified `betStatus`) that have been updated since a particular “betSnapshotItem”.

The parameter is specified as follows:

```
        &after=updateStamp
```

where *updateStamp* is the numerical value of the “updateStamp” element in one “betSnapshotItem” listed in a “betSnapshot”. The “updateStamp” is like a time stamp; even though the number it contains does not denote a specific time, nevertheless the highest “updateStamp” is always the most recent one. Therefore to see a list of bet updates since a particular “betSnapshotItem” was returned, you need to specify the “updateStamp” number of that “betSnapshotItem” as the value of the `after` parameter.

The “nextResultSet” element returned at the top of each “betSnapShot” (see Figure 69) contains a ready-made URL that uses this parameter and sets its value to the highest “updateStamp” number on the current “betSnapshot” page. It therefore automatically provides a list of updates to the current “betSnapshot” page.

A sample URL that returns all bets whose details have been matched since the “betSnapshotItem” with “updateStamp” 1158045715462 is:

`https://api.games.betfair.com/rest/v1/bet/snapshot?username=sampleuser&after=1158045715462&betStatus=MATCHED`

The above URL does not specify a channel. When you do not specify a channel, you can use all `betStatus` values except ACTIVE. In other words, you cannot request to see a list of updates to all current bets - both matched and unmatched – across channels.

A sample URL that does specify a channel and that requests updates to all ACTIVE bets since the "betSnapshotItem" with "updateStamp" 1158251224572 is:

`https://api.games.betfair.com/rest/v1/bet/snapshot?username=sampleuser&startRecord=1&recordCount=50&after=1158251224572&channelId=1444080&betStatus=ACTIVE`

If you do specify a channel in your "betSnapshot" URL, you can view ACTIVE, MATCHED, and UNMATCHED bets (but not CANCELLED or LAPSED ones; see next section). (ACTIVE bets are the MATCHED plus the UNMATCHED bets.)

13.2.2 Listing Current Bets for a Particular Channel (betSnapshot)

The URL for requesting a list of current bets of a particular status and a particular game (that is, a particular channel) is as follows:

`https://api.games.betfair.com/rest/v1/bet/snapshot?username=sampluser&betStatus=ACTIVE&channelId=`*`channelId`*

where *`channelId`* is the number of the channel you are interested in. You can also use MATCHED or UNMATCHED as the value of the betStatus parameter in this URL. (ACTIVE bets are the MATCHED plus the UNMATCHED bets.) You cannot use CANCELLED or LAPSED as the value of the betStatus parameter in this URL.

Note that the "betSnapshot" element returns data associated with a betting market that is still open. When the betting market has closed, you need to use the URLs associated with the "betHistory" element to retrieve details of your bets.

13.2.3 Listing Current Bets for a Particular Channel and Market (betSnapshot)

The URL for requesting a list of current bets of a particular status and place in a particular market (that is, a particular selection of betting outcomes) is as follows:

`https://api.games.betfair.com/rest/v1/bet/snapshot?username=sampluser&betStatus=ACTIVE&channelId=`*`channelId`*`&selectionsType=`*`selectionsType`*

where *`channelId`* is the number of the channel you are interested in and *`selectionsType`* is "MainBets", "SideBets", or "CorrectPredictions". This is only applicable for a channel in which a game is running that has two or more lists of betting selections. Each list of betting selections is called a market. At the time of writing the Exchange games that have more than one betting market are Baccarat and Exchange Card Racing.

If you do not specify a *`selectionsType`* and the game in the channel specified has more than one betting market associated with it, you will be presented with both markets.

You can also use MATCHED or UNMATCHED as the value of the betStatus parameter in this URL. You cannot use CANCELLED or LAPSED.

13.3 Requesting XML that Lists Past Bets by Status (betHistory)

The second group of URLs returned by the HTTP request for the “betLanding” element is a list of four URLs for requesting the “betHistory” element.

You cannot request to view all past bets at once. You can only request to view past bets of a specific status. The possible statuses are: CANCELLED, LAPSED, SETTLED, AND VOIDED.

The URLs in the “betLanding” element for viewing past bets according to their status are as follows:

https://api.games.betfair.com/rest/v1/bet/history?username=sampleuser&betStatus=CANCELLED

https://api.games.betfair.com/rest/v1/bet/history?username=sampleuser&betStatus=LAPSED

https://api.games.betfair.com/rest/v1/bet/history?username=sampleuser&betStatus=SETTLED

https://api.games.betfair.com/rest/v1/bet/history?username=sampleuser&betStatus=VOIDED

Figure 71 shows a sample response from the Games API to a request for a list of past bets.

The paging elements “nextPage” and “previousPage” (see Figure 71) contain a URL that enables you to navigate one page backward and one page forward, respectively (the sample output in Figure 71 is the first page of an “betHistory” element, therefore there is no previous page to view). By default there will be a maximum of 50 bet records per page of the “betHistory” element. However, you can specify a lower number by including the `&recordCount` parameter in the URL when you request the “betHistory” element (see Section 13.4.2).

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<betHistory currency="GBP" end="50" start="1" total="167" xmlns="urn:betfair:games:api:v1">
    <nextPage>
        <resource href="https://api.games.betfair.com/rest/v1/bets/betHistory?
         username=sampleuser&amp;betStatus=SETTLED&amp;startRecord=51&amp;recordCount=50"
         responseType="betHistory" title="Next Bet History Results"/>
    </nextPage>
    <betHistoryItem>
        <betId>794170028</betId>
        <selectionReference>
            <marketReference>
                <resource
                 href="https://api.games.betfair.com/rest/v1/channels/1444074/history
                 ?username=sampleuser&amp;marketId=2577535" responseType="channelHistory"
                 title="Exchange Poker Texas Hold'em"/>
                <marketId>2577535</marketId>
                <channelId>1444074</channelId>
                <channelName>Exchange Poker Texas Hold'em</channelName>
                <gameStartDate>2006-05-03T13:20:45+00:00</gameStartDate>
            </marketReference>
            <selectionId>658439</selectionId>
            <selectionName>Hand 1</selectionName>
        </selectionReference>
        <bidType>BACK</bidType>
        <placedDate>2006-05-03T13:20:48+00:00</placedDate>
        <price>3.0</price>
        <size>10.0</size>
        <averagePriceMatched>3.0</averagePriceMatched>
        <profitLoss>-10.0</profitLoss>
        <winLose>LOST</winLose>
    </betHistoryItem>
    <betHistoryItem>
        <betId>794166480</betId>
        <selectionReference>
            <marketReference>
                <resource href="https://api.games.betfair.com/rest/v1/channels/1444074/history?
                 username=sampleuser&amp;marketId=2569736" responseType="channelHistory"
                 title="Exchange Poker Texas Hold'em"/>
                <marketId>2569736</marketId>
                <channelId>1444074</channelId>
                <channelName>Exchange Poker Texas Hold'em</channelName>
                <gameStartDate>2006-04-28T15:39:36+00:00</gameStartDate>
            </marketReference>
            <selectionId>658439</selectionId>
            <selectionName>Hand 1</selectionName>
        </selectionReference>
        <bidType>LAY</bidType>
        <placedDate>2006-04-28T15:41:43+00:00</placedDate>
        <price>1.3</price>
        <size>10.0</size>
        <averagePriceMatched>1.3</averagePriceMatched>
        <profitLoss>-3.0</profitLoss>
        <winLose>WON</winLose>
    </betHistoryItem>
</betHistory>
```

Figure 71 Sample XML Listing a User's Two Past Bets (one Lost, the other Won)

13.4 Adding Parameters Manually to the URLs for Listing Bets

To each URL that is returned in the “betLanding” element, you can add other parameters manually. This section tells you what the parameters enable you to do and how to specify them. If you want to, you can use all of the parameters together.

To add a parameter, just put an ampersand character (&) at the end of the particular URL you want to use, and then specify the parameter. To add another parameter after this, add another ampersand character and specify the next parameter, and so on.

The URLs returned by the Games API in the “betLanding”, “betSnapshot”, and “betHistory” pages already contain parameters, and where applicable these have been described in Section 12.3 and Section 13.3. This section therefore supplements the information provided above.

13.4.1 Listing Past Bets Placed at or Between Particular Times (betHistory)

To each URL that requests the “betHistory” element, you can add either or both of the following parameters that enable you to:

- Specify a start date for the list of bets to be returned. The parameter you need to add to the end of the URL to do this (including the necessary ampersand character) is:

  `&placedDateStart=`*`yyyy-MM-dd-HH:mm`*

- Specify an end date for the list of bets to be returned. The parameter you need to add to the end of the URL to do this (including the necessary ampersand character) is:

  `&placedDateEnd=`*`yyyy-MM-dd-HH:mm`*

If you use either or both of these parameters then, by default, the Games API returns the bets listed in groups for each date on which you placed them.

13.4.2 Requesting a Specific Number of Bets from a List (betSnapshot and betHistory)

To each of the URLs for requesting the “betSnapshot” element or the “betHistory” element, you can add a parameter manually to:

- Specify the number of bets that you want the list returned to contain. The parameter (including the necessary ampersand character) for this is:

  `&recordCount=`*`number`*

13.4.3 Starting the List at a Specific Position in the Overall List (betSnapshot and betHistory)

To each of the URLs for requesting the “betSnapshot” element or the “betHistory” element, you can add a parameter manually to:

- Specify the start position (that is, the start position in the Exchange Games API’s overall list) for the list of bets that you want to receive. The parameter (including the necessary ampersand character) for this is:

  `&startRecord=`*`number`*

13.4.4 Listing Current Bets by Game or Time Placed (betSnapshot)

To each of the URLs for requesting the “betSnapshot” element, you can add a parameter manually to:

- Order the returned list of bets by the type of game you are interested in. The parameter and value for this (including the necessary ampersand character) are:

  ```
  &orderBy=channelName
  ```

- Order the returned list of current bets by the time each was placed. The parameter and value for this (including the necessary ampersand character) are:

  ```
  &orderBy=placedDate
  ```

  If you use this parameter, the Exchange Games API will order your bets by time. All bets are date- and time-stamped and bets on games are placed and settled within minutes (the games themselves lasting only a few minutes). Therefore it is normally the *time* (and not the date) that will determine the order of current bets listed. The date will only make a difference if you place a bet shortly before the close of one day and then make a “betSnapshot” request with this parameter quickly enough on the next day to capture the bet before it is settled.

13.4.5 Specifying a Time Zone for the Reporting of Your Bets (betHistory)

To each of the URLs requesting the “betHistory” element, you can add a parameter to specify the time zone to be used to report your bets. The parameter you need to add to the end of the URL to do this (including the necessary ampersand character) is:

```
&timezone=time-zone code
```

where *time-zone code* is a code of your choice from the table below.

Table 11 Time-zone Codes

Time zone	**Time-zone Code**	**Description**
Africa/Johannesburg	RSA	South African
America/Indiana/Indianapolis	IEST	North America Indiana East
Asia/Bangkok	THAI	Bangkok
Asia/Calcutta	INT	Indian Time
Asia/Dubai	UAE	United Arab Emirates
Australia/Adelaide	ACST	Australian Central
Australia/Darwin	ANST	Australian Northern Territories
Australia/Perth	AWST	Australian Western
Australia/Queensland	AQST	Australian Queensland
Australia/Sydney	AEST	Australian Eastern
CET	CET	Central European
EET	EET	Eastern European

Etc/GMT-5	PKT	Pakistan Time
Europe/London	UKT	UK Time
Europe/Moscow	MSK	Moscow
GMT	GMT	Greenwich Mean
Hongkong	HK	Hong Kong
Japan	JPT	Japan
NZ	NZT	New Zealand
US/Alaska	AKST	North America Alaska
US/Arizona	AST	North America Arizona
US/Central	CST	North America Central
US/Eastern	EST	North America Eastern
US/Hawaii	HST	North America Hawaii
US/Mountain	MST	North America Mountain
US/Pacific	PST	North America Pacific

Appendix A The Error Messages Returned by the API

This appendix describes the standard HTTP error messages that are returned by the Betfair Exchange Games API, the (more numerous) Betfair-specific error messages that are returned by the Games API in the “error” element, and Betfair-specific error (and other) messages that are returned by the Games API in the “responseBetOrder” element whenever you send an HTTP POST request to place, edit, or cancel a bet.

A.1 Standard HTTP Error Messages Returned by the API

Table 12 lists the standard HTTP error codes that are returned by the Betfair Exchange Games API and explains what causes them.

Table 12 Standard HTTP Error Responses Returned by the Betfair Exchange Games API

HTTP Error Code	Reason for Error Response
401 Unauthorized	You have sent a request for a service that requires a password (in the message header) and a username (in the URL), and either or both are missing, invalid, or incorrectly specified. This condition will be indicated in the title element of the page returned with the error code by the following text: “`user+name+or+password+incorrect`”
	You have sent a request for a service that requires an agent ID and an agent instance ID, but you have not included these IDs in the header. This condition will be indicated in the title element of the page returned with the error code by the following text: “`gamexapiagent+or+gamexapiagentinstance+not+supplied`”
	You have sent a request for a service that does not require a username and password, but you have either supplied a password with no username or a username with no password, or you have specified the username or password incorrectly. You can access certain information without a username and password. But if you send any user credential information, the Games API will check it even though it is not required for some resources.
404 Not Found	You have attempted to connect to the Games API server on the wrong TCP/IP port. The required port is 443.
	You have sent an HTTPS request with an incorrect URL.
503 Internal Error	The Games API is currently unable to handle your request because of a temporary overloading of the server, or because essential maintenance work is being performed.

A.2 The XML “error” Element Returned by the Games API (error)

This section describes the Betfair-specific error messages returned by the Games API in the form of an XML element called “error”. Figure 72 shows an example (including header information) of an error response resulting from an unsuccessful login attempt. Notice that, although the body of the response contains error messages, the HTTP response code in the first line of the header is still 200, indicating that the HTTP exchange has been successful.

```
HTTP/1.1 200 OK

Date: Tue, 18 Apr 2006 09:47:02 GMT
Server: Betex HTTP Server
Cache-Control: no-store
Pragma: no-cache
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Content-Type: text/xml; charset=UTF-8
processTime: 3
Transfer-Encoding: chunked

<error>
  <errorCode>LOGIN_ERROR</errorCode>
  <credentialCheck>LOGIN_REQUIRE_TERMS_AND_CONDITIONS_ACCEPTANCE</credentialCheck>
  <userStatus>ACTIVE</userStatus>
</error>
```

Figure 72 Sample Error Message (Showing the HTTP 200 Response Code) Resulting From a Login Error

Figure 72 shows the error that would be returned by the Games API if the user identified by the username and password submitted in a request had not yet accepted the Betfair terms and conditions on http://games.betfair.com.

Figure 73 shows sample output resulting from an incorrect use of the Games API; in this case an attempt to place a bet without sending the correct XML in the POST request.

```
<error>
<errorCode>INVALID_BET_ORDER</errorCode>
</error>
```

Figure *73* A Sample Error (Illustrating the Format of Messages Resulting From Incorrect Use of the API)

The “error” element is returned by the Games API when it has successfully received an HTTP GET or POST request but has been unable to process it for a reason that is not explicitly covered by the function-specific elements of the Games API.

This “error” element will contain a minimum of one other element called “errorCode”. But it can also contain one or more other elements in addition to this. The contents of the “error” element are as follows:

- “errorCode”, whose content is a string of type “errorCodeEnum”, and whose possible values are shown in
- Table *13*. (There will always be one of these elements in the “error” element.)
- “credentialCheck”, whose content is a string of type “loginCredentialsStatusEnum”, and whose possible values are shown in Table 14. These values are the results of the authentication test performed by the Games API on the username and password that you submit with an HTTP GET or POST request. (There will be zero or one of these elements in the “error” element.)
- “userStatus”, whose content is a string of type “userStatusEnum”, and whose possible values are shown in Table 15. These values are the results of the user status test performed by the Games API when the username and password have been authenticated. The test checks whether the account in whose name a client application is logging in is in order. (There will be zero or one of these elements in the “error” element.)

- “errorDetails”, whose content is a string providing information to do with a Java exception. This element will only be included if the “errorCode” element contains the value “EXCEPTION”. (There will be zero or one of these elements in the “error” element.)

Table 13 Possible Values of the “errorcode” Element in an “error” Response from the Games API

Strings of type “errorCodeEnum”	**Meaning of String**
INPUT_CURRENCY_DOES_NOT_MATCH_ ACCOUNT_CURRENCY	You have attempted to perform a transfer of money in one currency but your account is held in another.
INVALID_OR_MISSING_BET_STATUS_CRITERIA	You have specified an invalid bet status in the URL for requesting the “betSnapshot” or “betHistory” elements. For the valid statuses, see Chapter 12.
INVALID_START_DATE	You have specified an invalid start date in your GET request for the “betHistory” element listing your past bets (see Section 13.4.1).
INVALID_END_DATE	You have specified an invalid end date in your GET request for the “betHistory” element listing your past bets (see Section 13.4.1).
INVALID_START_RECORD	You have specified an invalid start record in your GET request for the “betHistory” or “betSnapshot” element listing your past or current bets (see Section 13.4.2); or you have specified an invalid start record in a request for the “channelHistory” element listing previous games in a channel (see Section 4.3.1).
INVALID_RECORD_COUNT	In your GET request for the “betHistory” or “betSnapshot” element listing your past or current bets (see Section 13.4.2), or in your GET request for the “channelHistory” element listing previous games in a channel, you have specified an invalid total number of entries for the returned list to contain (see Section 4.3.1).
INVALID_TIME_ZONE	In your GET request for the “betHistory” or “betSnapshot” element, you have specified an invalid time zone (see Section 13.4.5).
INVALID_ORDER_BY_CRITERIA	You have specified an invalid criterion for ordering the list of current bets in your request for the “betSnapshot” element (see Section 13.4.4). Make sure you type the parameter correctly to have your bets ordered by channel name (that is, by game) or by place date (that is, by the time they at which they were placed).

INVALID_CHANNEL_ID	You have specified an invalid channel ID in your request for the "betSnapshot" element listing your current bets (see Section 13.2.1).
INVALID_MARKET_ID	You have specified an invalid "market Id" in your request for the "channelHistory" element describing a specific past game (see 4.3.2).
INVALID_BET_ORDER	You have posted invalid XML in your request to place, edit, or cancel a bet (see Section 12.2).
INVALID_CHANNEL_OR_SELECTIONS_TYPE	You have specified an invalid *channelId* or *selectionsType* in your request for the "channelSnapshot" element describing a current game (see Section 4.2).
EXCEPTION	An unexpected error has occurred in the middle-tier of the Games API. If possible, more specific information will be presented in the "errorDetails" element of the "error" response provided by the Games API.

As well as appearing in the "credentialsCheck" element inside the XML "error" element, the text strings in Table 14 are also returned inside the "ping" element when you request XML reporting the operating status of the Games API (see Section 3.5). The strings in Table 14 are the possible values of the element called "credentialCheck". They are of type "loginCredentialsStatusEnum".

Table 14 Possible Results of the User Credentials Test Performed by the Games API

Strings of type "loginCredentialsStatusEnum"	**Meaning of String**
CREDENTIALS_NOT_SUBMITTED	You made the HTTP GET request without including a username and password.
LOGIN_SUCCESS	The login succeeded.
LOGIN_ACCOUNT_NOW_LOCKED	Your account has been locked because you submitted your username and password unsuccessfully three times. Please contact Betfair. For contact details, go to www.betfair.com, click About Us, then, in the side-bar of the pop-up window, click Contact Us.
LOGIN_ACCOUNT_ALREADY_LOCKED	Your account is locked because, on an earlier occasion, you unsuccessfully submitted your username and password three times. Please contact Betfair. For contact details, go to www.betfair.com, click About Us, then, in the side-bar of the pop-up window, click Contact Us.

LOGIN_REQUIRE_REFERRAL_BETEX	There is a problem with your login details. Please contact Betfair. For contact details, go to www.betfair.com, click About Us, then, in the side-bar of the pop-up window, click Contact Us.
LOGIN_WITH_MESSAGES	If you receive this message, Betfair need you to answer some questions on-line about your account. Please contact Betfair. For contact details, go to www.betfair.com, click About Us, then, in the side-bar of the pop-up window, click Contact Us.
LOGIN_BANNED_COUNTRY	You are attempting to log in from a country in which on-line betting is forbidden.
LOGIN_BANNED_TRADING_ACCOUNT	You are attempting to log into a trading account that Betfair have banned. Please contact Betfair. For contact details, go to www.betfair.com, click About Us, then, in the side-bar of the pop-up window, click Contact Us.
LOGIN_BANNED_MARGIN_ACCOUNT	You are attempting to log into a margin account that Betfair have banned. Please contact Betfair. For contact details, go to www.betfair.com, click About Us, then, in the side-bar of the pop-up window, click Contact Us.

Table 15 Possible Results of the User Status Test Performed by the Games API

Strings of type "userStatusEnum"	**Meaning of String**
ACTIVE	Your account is active.
CLOSED	Your account is closed.
DUPLICATE_CARDS	Your credit card appears to be a duplicate of another card. Check that you have entered the numbers correctly. Consider whether you have already registered the card.
LOCKED	The account has been locked by Betfair because you have submitted your username and password incorrectly three times.
PENDING_AUTH	Your account is currently going through a process of authorization.
SUSPENDED	Your account has been suspended by Betfair. Please contact Betfair. For contact details, go to www.betfair.com, click About Us, then, in the side-bar of the pop-up window, click Contact Us.

Strings of type “userStatusEnum”	**Meaning of String**
TELBET_TCs_NA	This error string concerns customers who intend to take advantage of telephone betting, and therefore it is not a message that Games API users should expect to receive. It indicates that a customer has not accepted the terms and conditions (“TCs”) for telephone betting. Although you will see this string in the XML schema, the betting windows for the games on the Betfair Exchange Games Exchange are not open for long enough to make telephone betting practicable.
SEC_QUES_WRONG_X3	You have answered the login security questions on the Betfair web site wrongly three times.
KYC_SUSPEND	You have been requested to complete a “Know Your Customer” questionnaire by Betfair, and Betfair have not yet received or processed the answers to this questionnaire.

A.3 Error (and Other) Responses to Betting POST Requests (responseBetOrder)

When you place, edit, or cancel a bet by sending an HTTP POST request containing the XML element “postBetOrder” to the Betfair Exchange Games API, you will receive a response containing the XML element “responseBetOrder. (For more information about these two elements associated with placing, editing, and cancelling bets, see Section 0.)

The “responseBetOrder” element returned will contain a single element reporting the result of your POST request. Its content will either be a “betPlacementResult” element, a “bedEditResult” element, or a “betCancelResult” element, depending on the nature of the POST request you sent. In all cases, this element will itself contain a “betId” (which is a long integer) element and a “resultCode” element, which is a string of type “betResultEnum”.

Table 16 contains a list of all the possible values of the strings of type “betResultEnum” that report the success or failure of a bet placement, bet edit, or bet cancellation request.

Table 16 Possible Values of the “resultCode” Element in the XML Response to a Betting POST Request

Strings of type “betResultEnum”	The meaning of the String
OK	The POST request was successful.
UNKNOWN_ERROR	The problem that has occurred cannot be explained by the Games API. Please contact Betfair. For contact details, go to www.betfair.com, click About Us, then, in the side-bar of the pop-up window, click Contact Us.
REMAINING_CANCELLED	The amount of your bet that remained unmatched has been cancelled.
SITE_UPGRADE	The Betfair Exchange Games API is currently being upgraded.
ACCOUNT_CLOSED	Your account has been closed. Please contact Betfair. For contact details, go to www.betfair.com, click About Us, then, in the side-bar of the pop-up window, click Contact Us.
ACCOUNT_SUSPENDED	Your account has been suspended. Please contact Betfair. For contact details, go to www.betfair.com, click About Us, then, in the side-bar of the pop-up window, click Contact Us.
ACCOUNT_PENDING_AUTHORIZATION	Your account is going through the authorization process.
EVENT_CLOSED	This message indicates that the game you were attempting to bet on has finished.
BET_REJECTED_EVENT_CLOSED	Your bet has been rejected because the event you were attempting to bet on has finished.

SELECTION_ALREADY_LOST	You have attempted to bet on or against a selection that has already lost.
SELECTION_ALREADY_WON	You have attempted to bet on or against a selection that has already won.
BETTING_ROUND_STALE	The betting window closed before your bet could be placed.
BET_REJECTED_ROUND_STALE	The round specified in the request is not open.
INTERNAL_ERROR	An unexpected error has occurred. Please contact Betfair. For contact details, go to www.betfair.com, click About Us, then, in the side-bar of the pop-up window, click Contact Us.
UNMATCHED_BET_NOT_FOUND	You have attempted to edit or cancel a bet that has either lapsed or that has already been matched; or you have specified the wrong “bet Id” in your “postBetOrder” request.
ERROR_CALCULATING_EXPOSURE	The Games API has encountered an error while calculating your exposure and is unable to proceed with your betting request.
REMAINING_SIZE_CANCELLED_BET_ NOT_PLACED	The attempt you have made to edit a bet has been unsuccessful. The amount of the bet that remained unmatched has been cancelled, but the new bet that you requested has not been accepted.
EXPOSURE_CALCULATION_IN_PROGRESS	The Games API is calculating your exposure.
EXPOSURE_OR_AVAILABLE_BALANCE_ EXCEEDED	The Games API will not accept your bet because to do so would exceed your exposure limit or the balance in your Betfair Exchange Games account.
FROM_COUNTRY_ON_EVENT_FORBIDDEN	In the country you are attempting to place a bet from on-line betting is against the law.
INVALID_ASIAN_LINE_ID	You have specified an invalid selection for an Asian HSLV (handicap single-line variant) betting market.
INVALID_BET_TYPE	You have attempted to place a bet that was neither a Back or a Lay.
INVALID_INCREMENT	You have specified a size of bet that represents an invalid increment.
INVALID_MARKET	You have attempted to place a bet in a market that doesn’t exist. Make sure you specify the “marketId” correctly when you submit your “postBetOrder” request.
INVALID_SELECTION	You have specified the betting selection incorrectly.

	Check the contents of the "selectionid" element in your "postBetOrder".
INVALID_SIZE_OR_PRICE	You have specified the size or price of the bet incorrectly. For information about the minimum prices (in other words, the minimum bets for different currencies), see Section 12.4. For information about the odds that are valid within different ranges, see Section 12.5.
LOSS_LIMIT_EXCEEDED	You have exceeded your loss limit. To increase your limit, please contact Betfair. For contact details, go to www.betfair.com, click About Us, then, in the side-bar of the pop-up window, click Contact Us.
GAME_DOES_NOT_EXIST_OR_STALE	You have attempted to bet on a game that either does not exist (perhaps because you have specified the "channelId" incorrectly), or the game you were attempting to bet on is over.
GAMEX_EXCLUSION	Your bet could not be accepted because of restrictions that apply to your Betfair Exchange Games user account. Please contact Betfair. For contact details, go to www.betfair.com, click About Us, then, in the side-bar of the pop-up window, click Contact Us.

Appendix B Statuses for “object” and “selection” Elements

This appendix provides diagrams showing the possible statuses and status transitions for the automated players and betting choices associated with each game.

It also includes tables summarizing all the statuses possible for the “object” elements in each game and all the statuses possible for the “selection” statuses associated with each game.

The automated players are represented in the XML output as “object” elements, and the betting choices are represented in the XML output from the Exchange Games API as “selection” elements.

Tables summarizing the possible statuses for all “object” and “selection” elements are given below.

Table 17 Statuses Possible for "object" Elements During Games

Status of player “object”	**Poker (XPT)**	**Omaha Hi (XOH)**	**Blackjack (XBJ)**	**Baccarat (XBT)**	**Hi Lo (XHL)**	**Exchange Card Racing**
IN_PLAY	✓	✓	✓	✓	✓	
LOSER	✓	✓	✓	✓	✓	
WINNER	✓	✓	✓	✓	✓	
TIED	✓	✓	✓	✓		
DEALING			✓			
HIT			✓	✓		
STOOD			✓	✓		

Table 18 Statuses Possible for "selection" Elements During Games

Status of betting “selection”	**Poker (XPT)**	**Omaha Hi (XOH)**	**Blackjack (XBJ)**	**Baccarat (XBT)**	**Hi Lo (XHL)**	**Exchange Card Racing**
IN_PLAY	✓	✓	✓	✓	✓	
LOSER	✓	✓	✓	✓	✓	
WINNER	✓	✓	✓	✓	✓	
TIED_DEAD_HEAT	✓	✓	✓			
REFUND				✓		
CANNOT_WIN				✓		
CANNOT_LOSE				✓		
NOT_APPLICABLE						

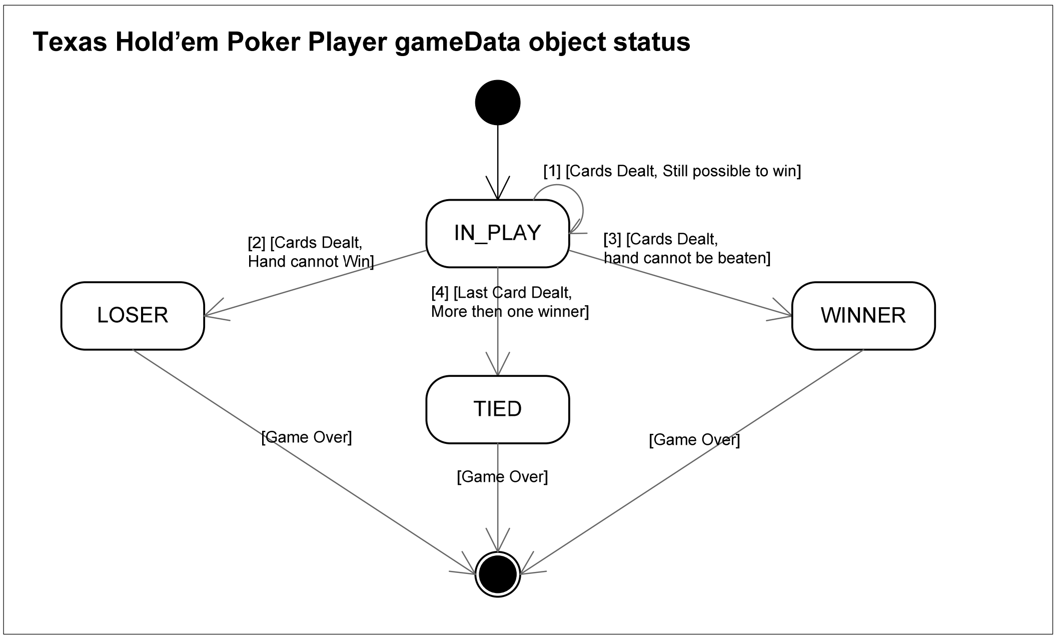


Figure 74 Possible Statuses for the Player "object" Elements in Texas Hold'em Poker

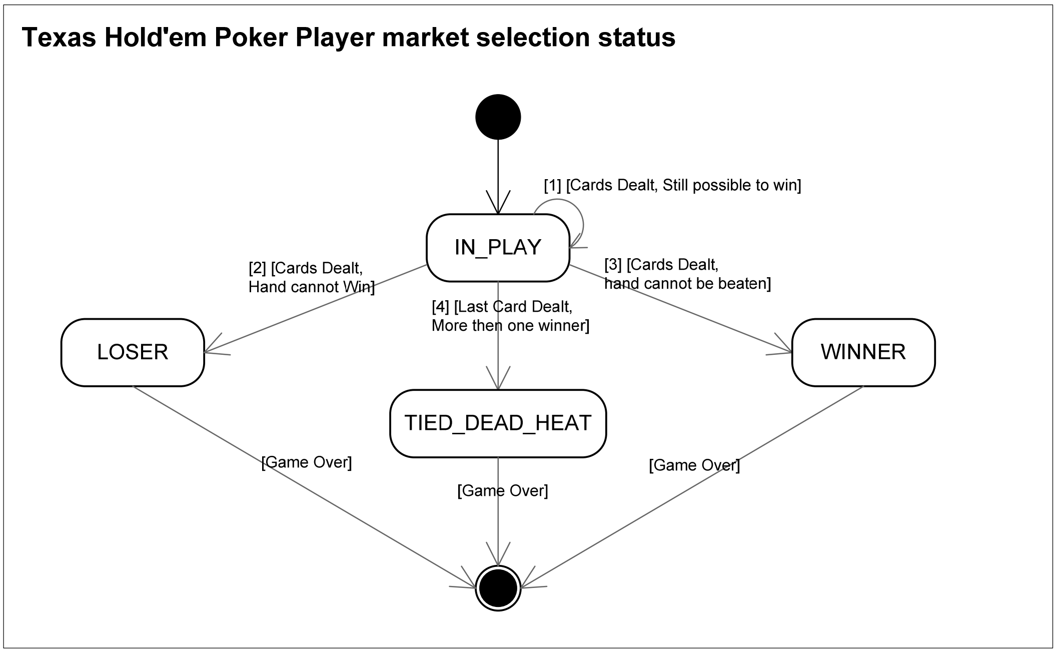


Figure 75 Possible Statuses for the Betting "selection" Elements in Texas Hold'em Poker

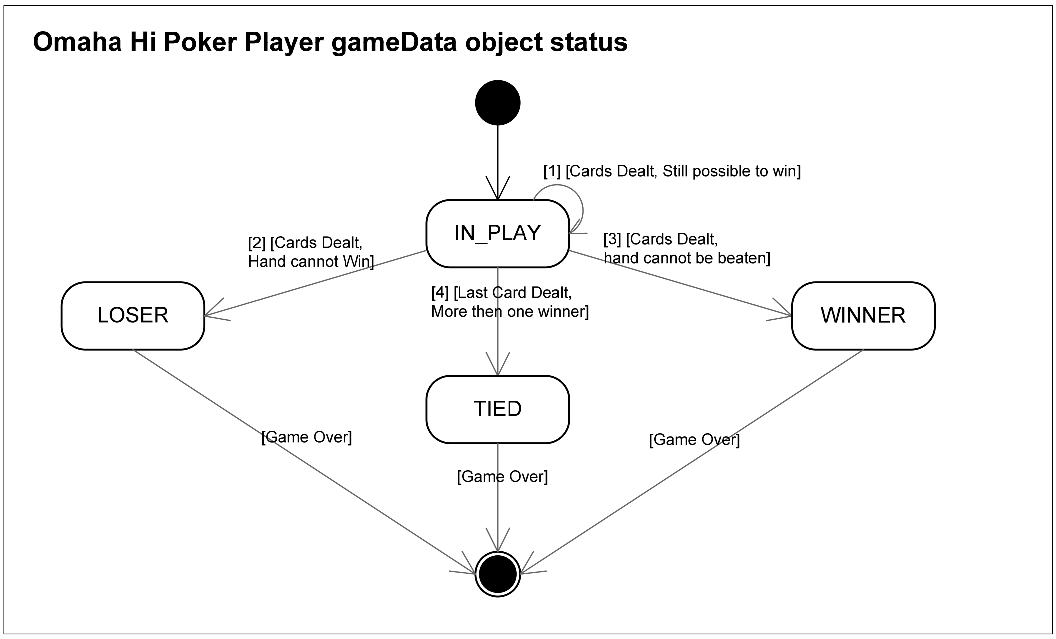


Figure 76 Possible Statuses for the Player "object" Elements in Omaha Hi Poker

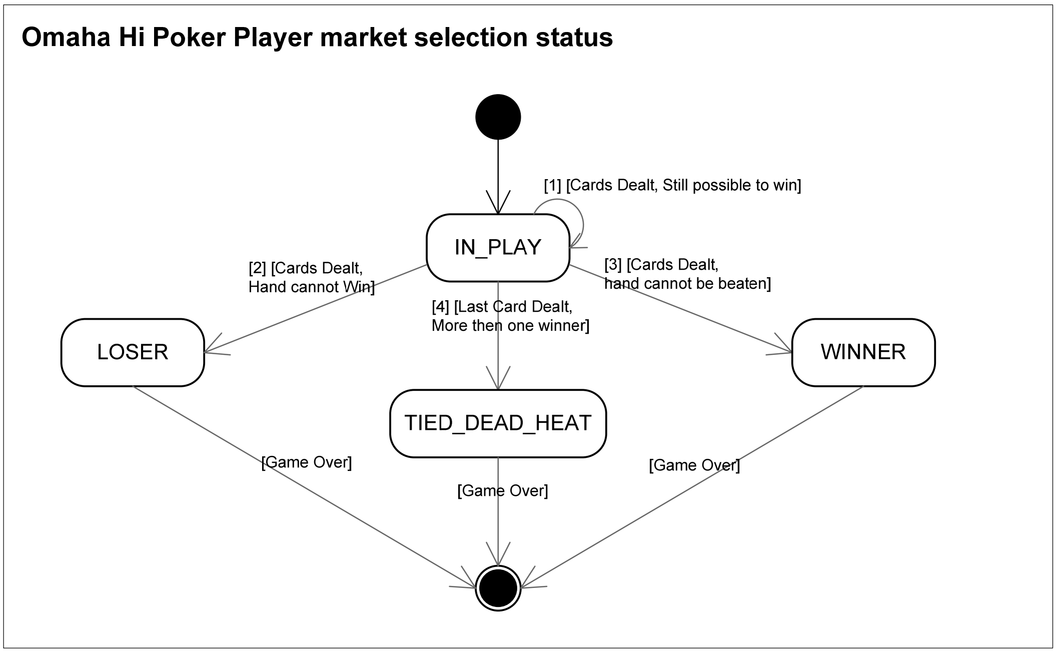


Figure 77 Possible Statuses for the MainBets Betting "selection" Elements in Omaha Hi Poker

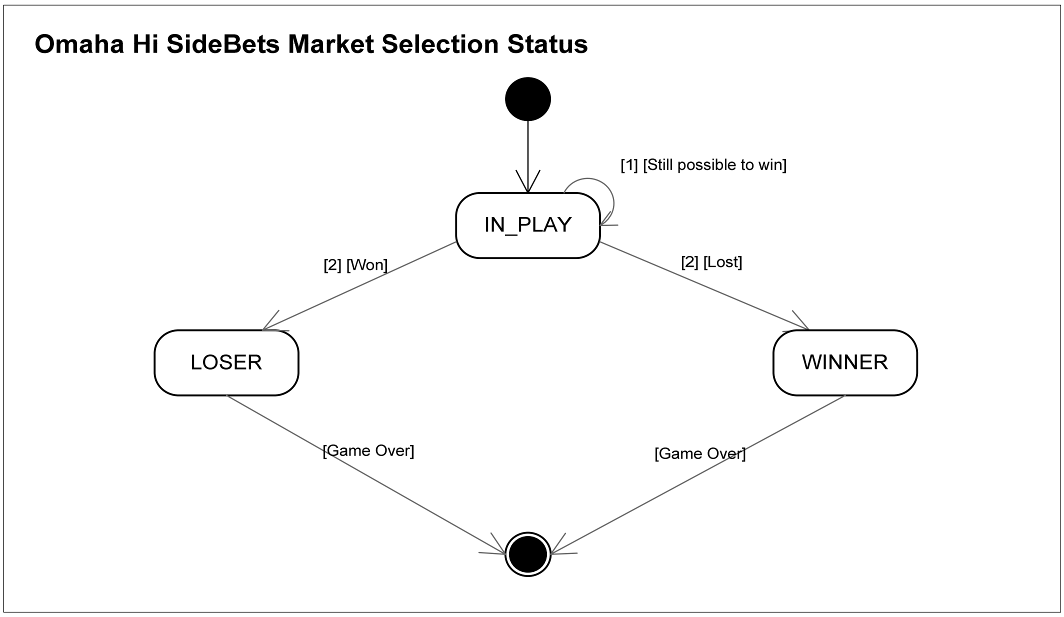

Omaha Hi SideBets Market Selection Status
[1] [Still possible to win]
IN_PLAY
[2] [Won]
[2] [Lost]
LOSER
WINNER
[Game Over]
[Game Over]

Figure 78 Possible Statuses for the SideBets Betting "selection" Elements in Omaha Hi Poker

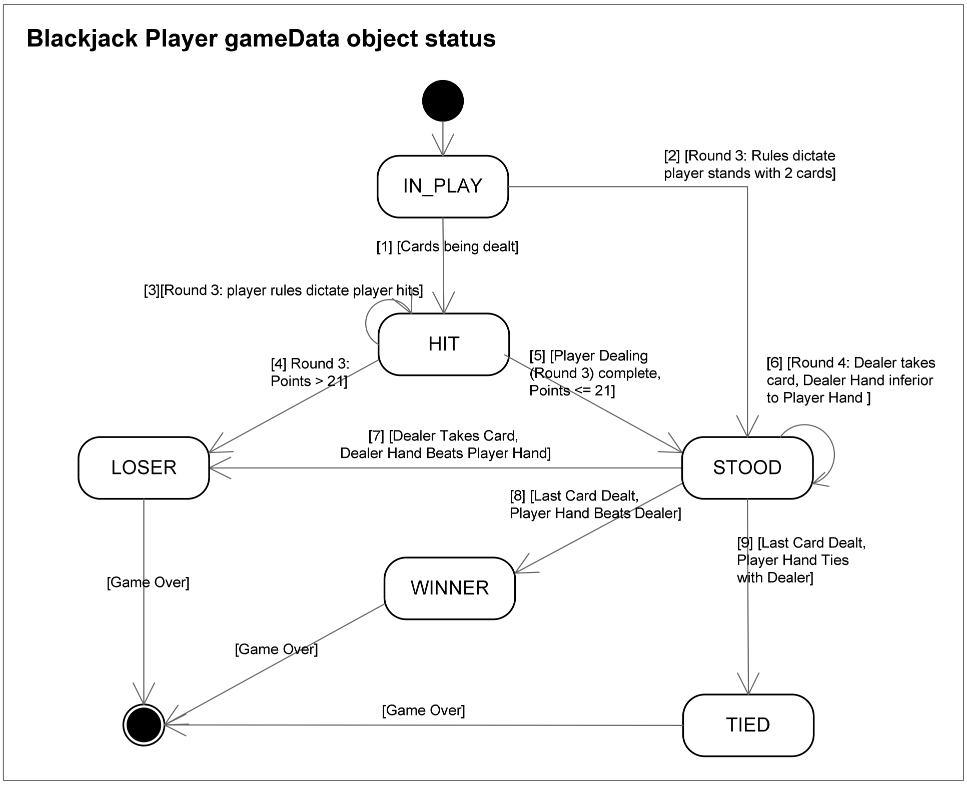

Blackjack Player gameData object status
IN_PLAY
[2] [Round 3: Rules dictate player stands with 2 cards]
[1] [Cards being dealt]
[3][Round 3: player rules dictate player hits]
HIT
[4] Round 3: Points > 21]
[5] [Player Dealing (Round 3) complete, Points <= 21]
[6] [Round 4: Dealer takes card, Dealer Hand inferior to Player Hand]
LOSER
[7] [Dealer Takes Card, Dealer Hand Beats Player Hand]
STOOD
[8] [Last Card Dealt, Player Hand Beats Dealer]
[9] [Last Card Dealt, Player Hand Ties with Dealer]
WINNER
[Game Over]
[Game Over]
[Game Over]
TIED

Figure 79 Possible Statuses for a Player "object" Element in Blackjack

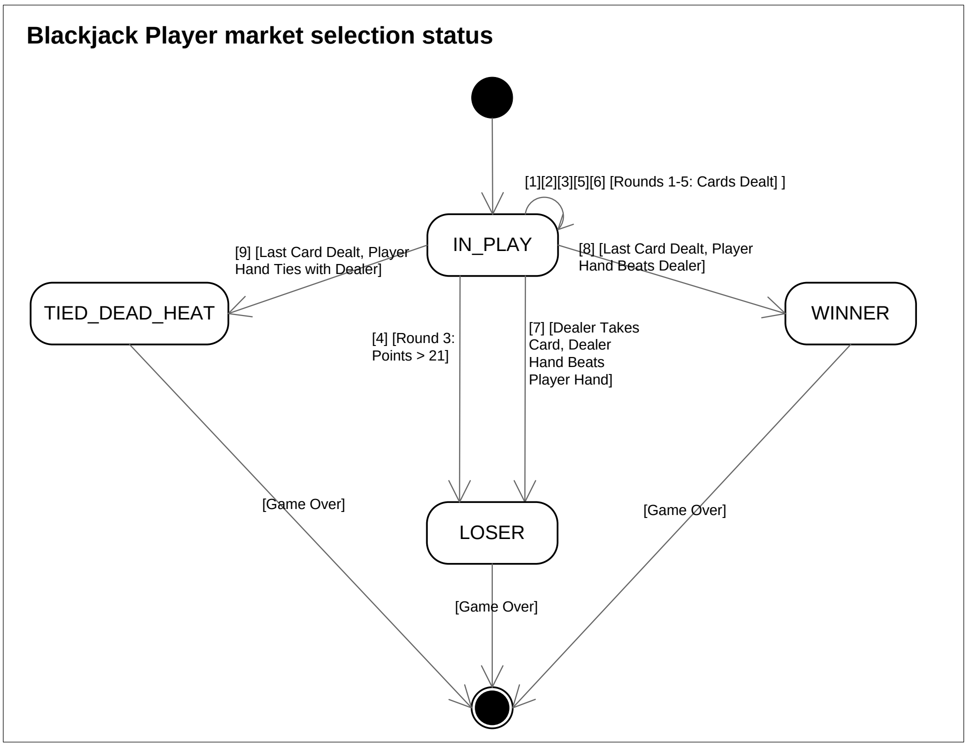

Blackjack Player market selection status
[1][2][3][5][6] [Rounds 1-5: Cards Dealt]]
IN_PLAY
[9] [Last Card Dealt, Player Hand Ties with Dealer]
[8] [Last Card Dealt, Player Hand Beats Dealer]
TIED_DEAD_HEAT
WINNER
[4] [Round 3: Points > 21]
[7] [Dealer Takes Card, Dealer Hand Beats Player Hand]
[Game Over]
LOSER
[Game Over]
[Game Over]

Figure 80 Possible Statuses for a Player "selection" Element in Blackjack

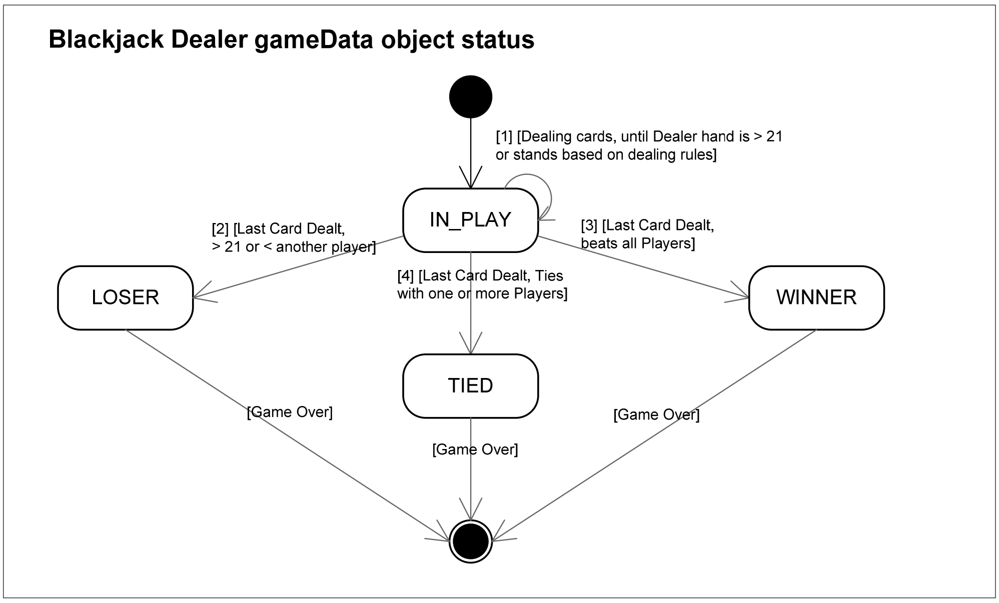

Blackjack Dealer gameData object status
[1] [Dealing cards, until Dealer hand is > 21
or stands based on dealing rules]
IN_PLAY
[2] [Last Card Dealt,
> 21 or < another player]
[3] [Last Card Dealt,
beats all Players]
[4] [Last Card Dealt, Ties
with one or more Players]
LOSER
WINNER
TIED
[Game Over]
[Game Over]
[Game Over]

Figure 81 Possible Statuses for the Dealer "object" Element in Blackjack

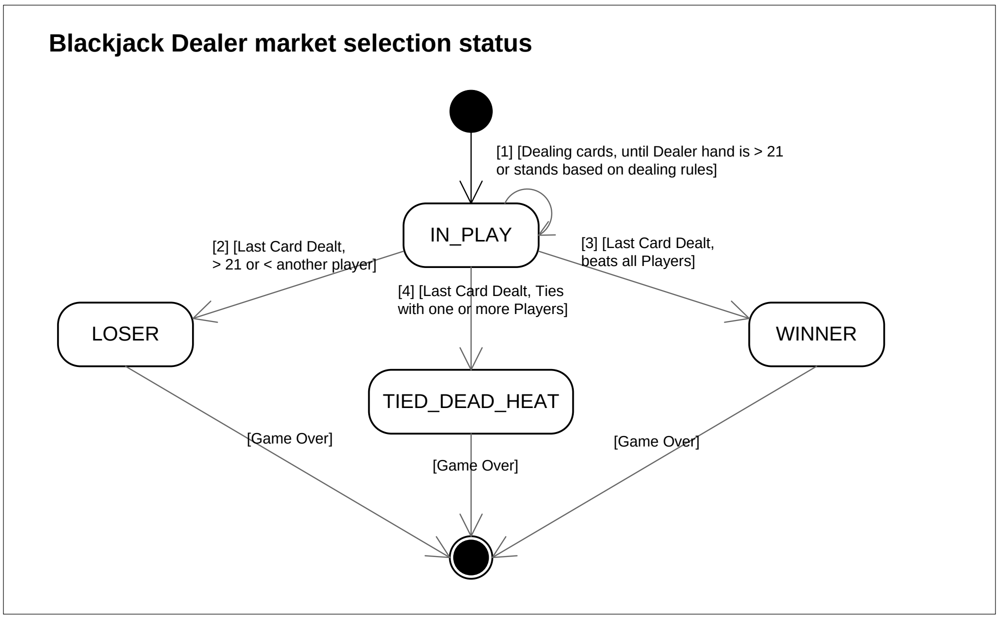

Blackjack Dealer market selection status
[1] [Dealing cards, until Dealer hand is > 21
or stands based on dealing rules]
IN_PLAY
[2] [Last Card Dealt,
> 21 or < another player]
[3] [Last Card Dealt,
beats all Players]
[4] [Last Card Dealt, Ties
with one or more Players]
LOSER
WINNER
TIED_DEAD_HEAT
[Game Over]
[Game Over]
[Game Over]

Figure 82 Possible Statuses for the Dealer "selection" Element in Blackjack

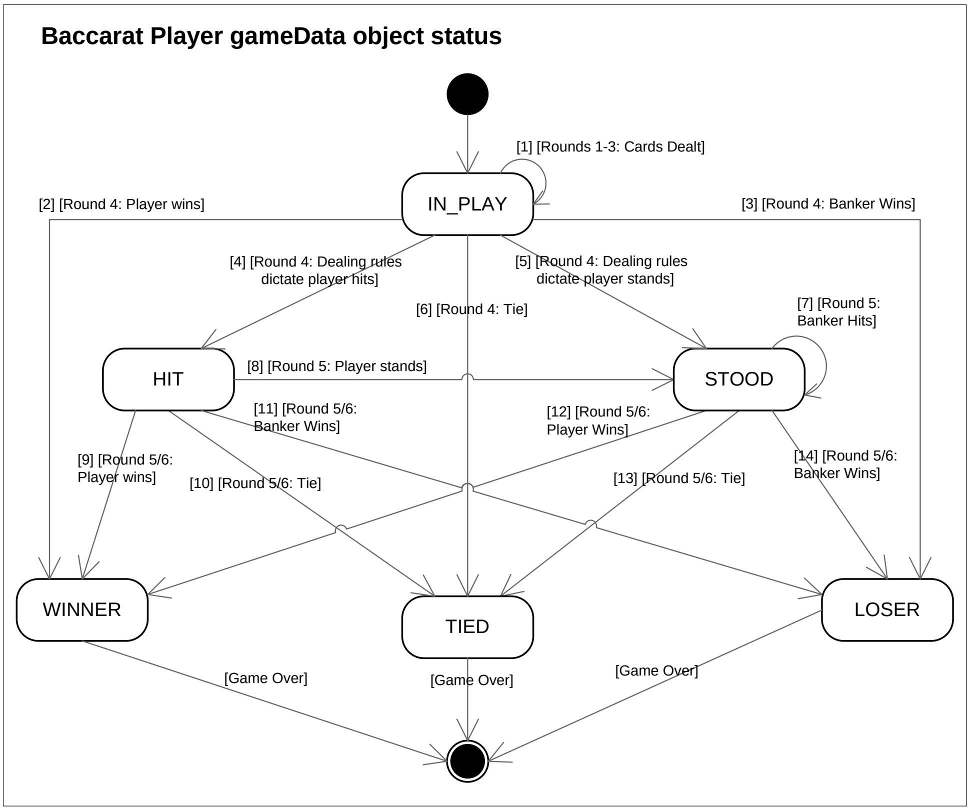

Baccarat Player gameData object status
[1] [Rounds 1-3: Cards Dealt]
IN_PLAY
[2] [Round 4: Player wins]
[3] [Round 4: Banker Wins]
[4] [Round 4: Dealing rules dictate player hits]
[5] [Round 4: Dealing rules dictate player stands]
[6] [Round 4: Tie]
[7] [Round 5: Banker Hits]
HIT
[8] [Round 5: Player stands]
STOOD
[11] [Round 5/6: Banker Wins]
[12] [Round 5/6: Player Wins]
[9] [Round 5/6: Player wins]
[10] [Round 5/6: Tie]
[13] [Round 5/6: Tie]
[14] [Round 5/6: Banker Wins]
WINNER
TIED
LOSER
[Game Over]
[Game Over]
[Game Over]

Figure 83 Possible Statuses for the Player "object" Element in Baccarat

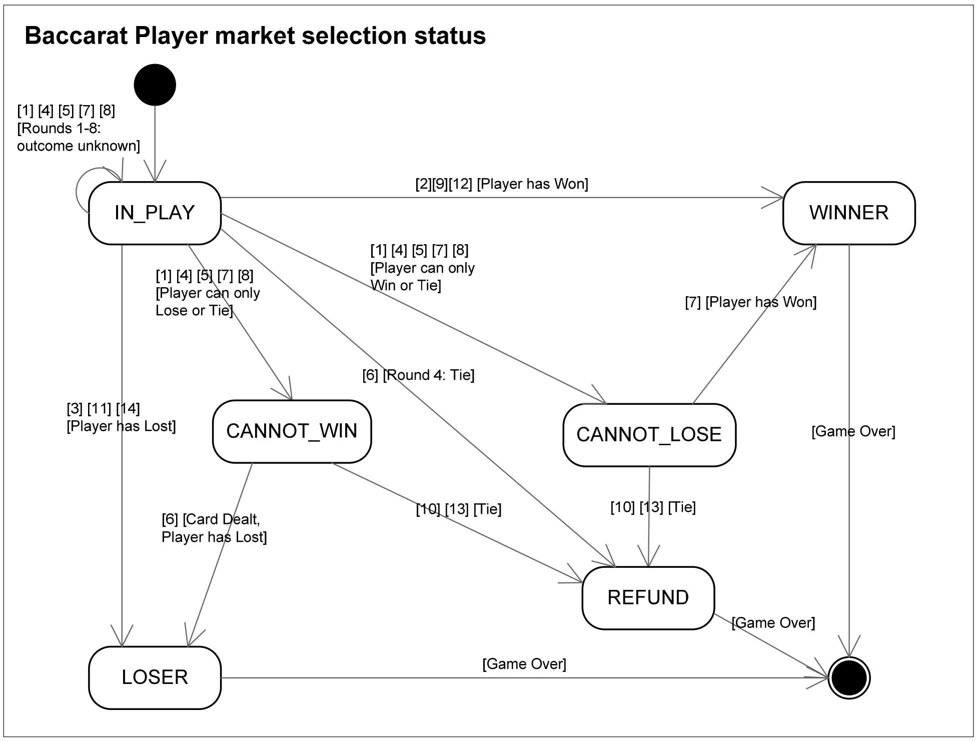

Baccarat Player market selection status
[1] [4] [5] [7] [8] [Rounds 1-8: outcome unknown]
IN_PLAY
[2][9][12] [Player has Won]
WINNER
[1] [4] [5] [7] [8] [Player can only Win or Tie]
[1] [4] [5] [7] [8] [Player can only Lose or Tie]
[7] [Player has Won]
[6] [Round 4: Tie]
[3] [11] [14] [Player has Lost]
CANNOT_WIN
CANNOT_LOSE
[Game Over]
[6] [Card Dealt, Player has Lost]
[10] [13] [Tie]
[10] [13] [Tie]
REFUND
[Game Over]
LOSER
[Game Over]

Figure 84 Possible Statuses for the Player Betting "selection" Element in Baccarat

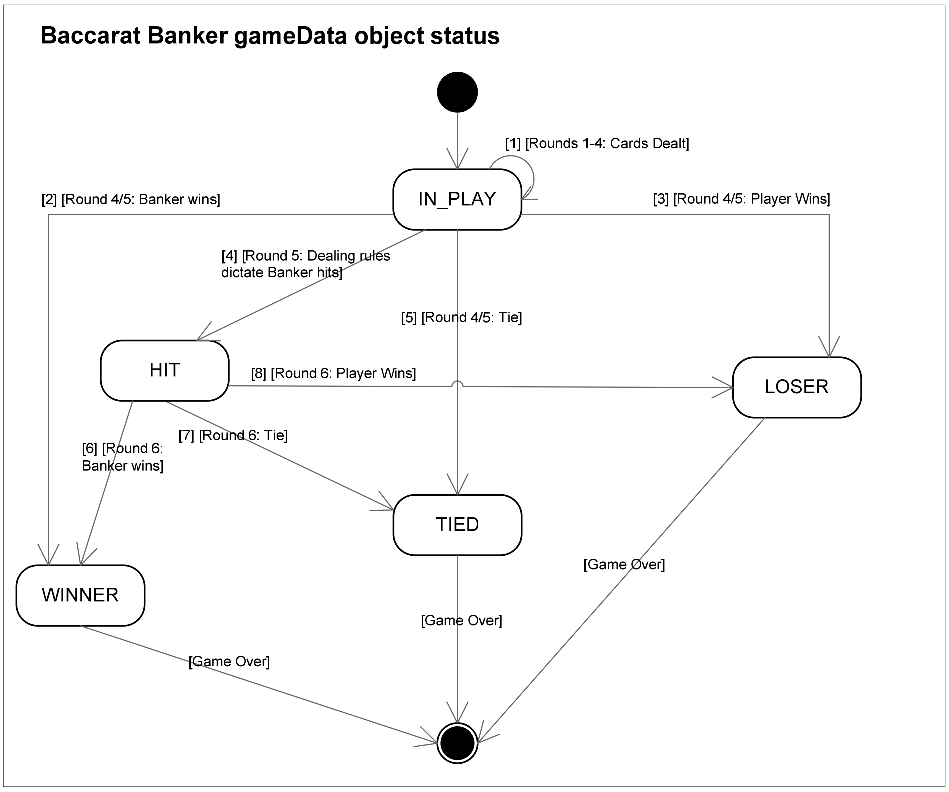

Baccarat Banker gameData object status
[1] [Rounds 1-4: Cards Dealt]
IN_PLAY
[2] [Round 4/5: Banker wins]
[3] [Round 4/5: Player Wins]
[4] [Round 5: Dealing rules dictate Banker hits]
[5] [Round 4/5: Tie]
HIT
[8] [Round 6: Player Wins]
LOSER
[7] [Round 6: Tie]
[6] [Round 6: Banker wins]
TIED
WINNER
[Game Over]
[Game Over]
[Game Over]

Figure 85 Possible Statuses for the Banker "object" Element in Baccarat

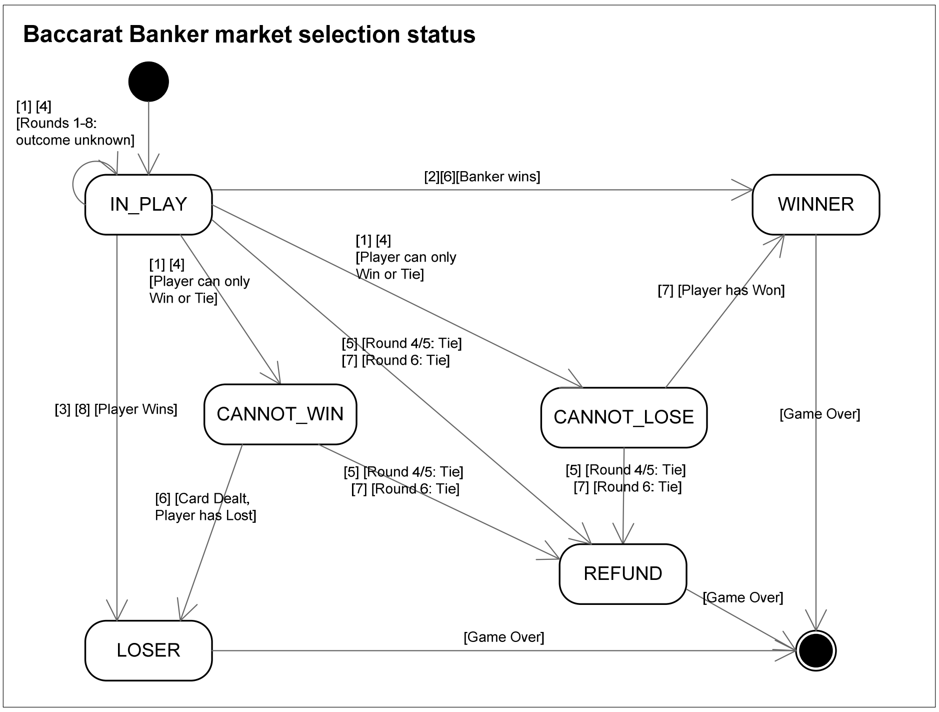

Baccarat Banker market selection status
[1] [4] [Rounds 1-8: outcome unknown]
IN_PLAY
[2][6][Banker wins]
WINNER
[1] [4] [Player can only Win or Tie]
[1] [4] [Player can only Win or Tie]
[7] [Player has Won]
[5] [Round 4/5: Tie] [7] [Round 6: Tie]
[3] [8] [Player Wins]
CANNOT_WIN
CANNOT_LOSE
[Game Over]
[5] [Round 4/5: Tie] [7] [Round 6: Tie]
[5] [Round 4/5: Tie] [7] [Round 6: Tie]
[6] [Card Dealt, Player has Lost]
REFUND
[Game Over]
LOSER
[Game Over]

Figure 86 Possible Statuses for the Banker Betting "selection" Element in Baccarat

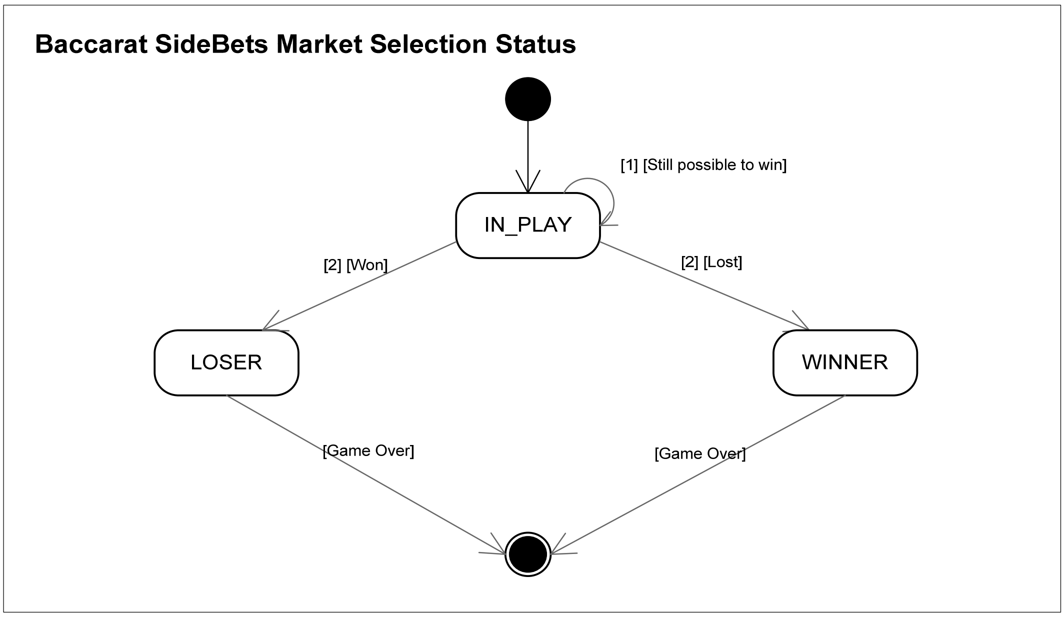

Baccarat SideBets Market Selection Status
[1] [Still possible to win]
IN_PLAY
[2] [Won]
[2] [Lost]
LOSER
WINNER
[Game Over]
[Game Over]

Figure 87 Possible Statuses for the SideBets Betting "selection" Elements in Baccarat

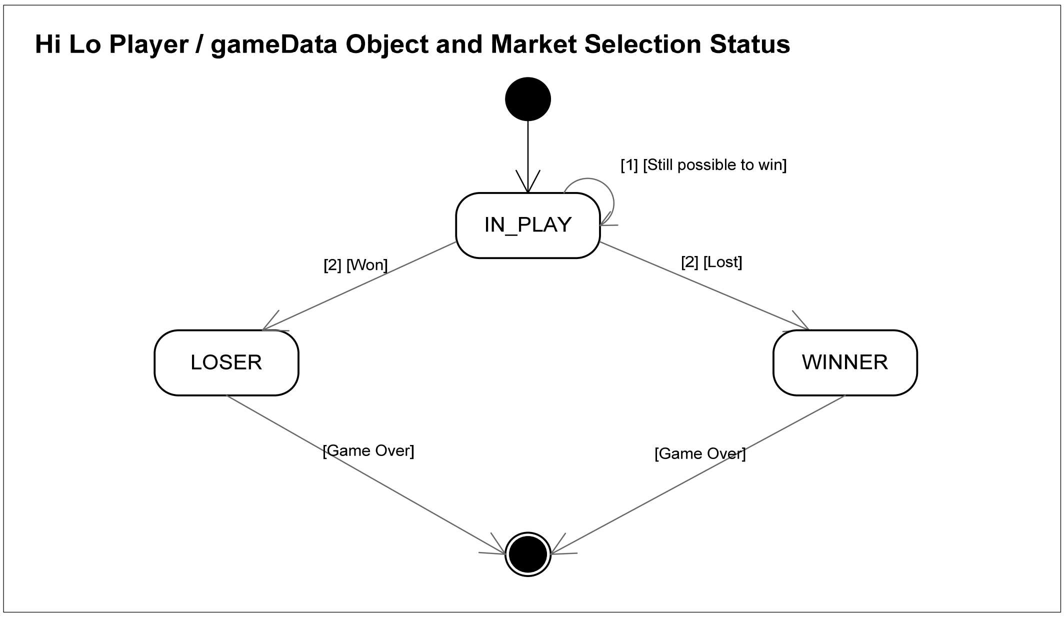

Hi Lo Player / gameData Object and Market Selection Status
[1] [Still possible to win]
IN_PLAY
[2] [Won]
[2] [Lost]
LOSER
WINNER
[Game Over]
[Game Over]

Figure 88 Possible Statuses for the "object" and "selection" Elements in Hi Lo

The diagram in Figure 88 applies to all “object” and “selection” elements that are not specifically illustrated in the previous diagrams in this appendix.

Appendix C Rules and Good Practice for API Developers

This appendix lists certain rules, requirements, and principles of good practice for customers who are developing client applications against the Betfair Games API. Some of the rules are legal requirements that bear on customer security; as such, these form part of the terms and conditions under which you are permitted to access the API. Others are practical recommendations designed to enable you to maximise the performance of your own client application while at the same time helping to maximise the performance of the Betfair exchange servers for all customers (by eliminating redundant traffic).

1. **Do not store Betfair username and password information on any static media**
   This is a legal requirement. Your client application must not store Betfair username and password information on any static media. The username and password must be supplied manually by the user of the client application each time he or she runs the application.

2. **Do not permit any intermediate processing of data between your client application and Betfair**
   This is a legal requirement. Your client application must communicate directly with the Betfair Games API. It must not pass API service requests or responses to any entity (such as a proxy server) that is capable of decrypting the traffic on its way to or from Betfair.

3. **Do not continue transmitting requests if you are receiving error messages**
   If you are receiving error messages, stop transmitting API requests, analyze the error messages, and take remedial action.

4. **Do not request more data than you require**
   Make use of the optional parameters that enable you to limit the data returned by certain API requests to only the data you require. For more information, refer to the sections of this document that describe parameters for controlling the data returned by:

   - “channelSnapshot” requests: see section 4.2.1
   - “channelHistory” requests: see Sections 4.3.1 and 4.3.2
   - “accountStatement” requests: see Section 11.5
   - “betSnapshot” and “betHistory” requests: see Sections 12.3, 13.3, and 13.4

5. **Do not request data updates more often than they are available**
   The Games exchange server, for example, updates the game data approximately every three seconds. Therefore, if you only want to track the progress of a game but do not necessarily need market information, do not send “channelSnapshot” requests to the Games API more requently than every three seconds.

6. **Do take advantage of the facilities in HTTP v1.1 for setting up persistent connections**
   For information about this facility, refer to Section 8.1 (entitled “Persistent Connections”) in the World-wide Web Consortium discussion document RFC2616, which is available on the following URL:

   http://www.w3.org/Protocols/rfc2616/rfc2616.html

7. **Do take advantage of the HTTP facilities for using compression**
   For information about using compression, again, refer the World-wide Web Consortium discussion document RFC2616 on the URL above; the relevant section is Section 14.3 (entitled “Accept-encoding”).

Index

accountBetfairSnapshot element, 102

accountLanding element, 99

Accounts

checking your main Betfair balance, 102

listing your Games balances, 99, 100, 101

listing your transactions, 99, 103, 104

managing your accounts, 99

transferring funds, 99, 106

transferring funds, 106

accountSnapshot element, 100

accountStatement

listing your transactions, 105, 106

accountStatement element, 103, 104, 105, 106

Agent ID, 20

Agent instance ID, 21

Ampersand characters, 16, 17, 51, 105, 128, 129

Anonymous access to the Games API, 20

API

root element, 26

services available, 26

Automated players

for Baccarat, 80

for Blackjack, 71

for Omaha Hi, 64

for Poker, 58

Hi Lo, 93

Baccarat

banker, 80

Default betting market, 82

player, 80

scorecard, 80

shoe, 80

Sidebets betting market, 83

balanceResponseTypeEnum strings, 106

Betfair login

How to get a username and password, 20

Including your login details in HTTP requests, 20

betHistory element, 124, 126, 128, 133

betHistory parameters, 124, 128

betLanding element, 108, 122

betPlacementResult element, 137

betResultEnum strings, 137

betSnapshot element, 123, 125, 129, 133, 134

betSnapshot parameters, 125, 128, 129

Betting

cancelling a bet, 108, 109, 119, 137, 138

editing a bet, 109, 119, 137, 138

listing your past bets, 124, 125, 126, 127, 128, 129, 133

placing a bet, 108, 109, 115, 119, 137, 139

viewing your current bets, 122, 123, 124, 125, 128, 129, 133, 134

Betting 'Bots, 15

Blackjack

cards, 71, 75

players, 71, 75

Card mappings, 60

Card properties

for Baccarat, 87

for Blackjack, 75

for Hi Lo, 93

for Omaha Hi, 65

for Poker, 52, 61

channel element, 30, 50

channelHistory

archived games, 30

channelHistory element, 50, 51, 133, 134

channelHistory parameters, 51

channelInfo element, 52, 53, 61, 65, 75, 87 channelLanding element, 29

channels, 27

channelSnapshot element, 32, 33, 34, 53, 70, 72, 73, 80, 83, 87, 93, 134

commission, 100

Element hierarchy, 24

entity references, 16

error element, 132

errorCodeEnum strings, 133

Five-card Charlie, 67

gameData element, 80

gameDescription

element for Baccarat, 88

element for Blackjack, 75

element for Hi Lo, 98

element for Omaha Hi. *See* element for Poker

element for Poker, 52, 61

Games

Baccarat, 29, 51, 76, 80, 81, 82, 83, 84, 85, 87, 88, 98

Blackjack, 14, 29, 70, 71, 72, 74, 75, 80, 81, 87

checking the betting time, 52, 61, 66, 75, 76, 87, 88, 98

Hi Lo, 89, 93, 96, 98

listing all current games, 29, 30, 31, 32, 36, 37, 38, 40, 54, 61, 62, 65, 66, 69, 75, 76, 87, 89, 98

Omaha Hi, 64, 65

Poker, 14, 29, 33, 50, 51, 54, 57, 58, 59, 61, 70, 80, 81, 104, 123, 127

viewing a particular game, 30, 32, 33, 34, 35, 51, 57, 66, 70, 72, 73, 74, 76, 83, 87, 123, 134

gamexAPIAgent, 20

gamexAPIAgentInstance, 21

GET request example, 22

Hi Lo, 89

automated player, 90, 93

board, 91, 93

cards, 93

CorrectPredictions betting market, 92, 94, 96

potential profit and loss, 47

rules, 89

Sidebets betting market, 96

SideBets market, 92

HTTP error codes, 131

HTTPS, 25

kicker, 57

Login problems, 132, 135

loginCredentialsStatusEnum strings, 134

market element, 45, 86

markets element, 86

Message headers, 16, 19, 20, 22, 25, 26, 131, 132

object

element for Baccarat, 80

element for Blackjack, 71

element for Hi Lo, 93

element for Omaha Hi. *See element for Poker*

element for Poker, 58

statuses for Baccarat, 80

statuses for Blackjack, 71

statuses for Hi Lo, 93

statuses for Omaha Hi. *See statuses for Poker*

statuses for Poker, 58

Omaha Hi, 62

Definitions of selections, 63

MainBets betting market, 64

SideBets betting market, 63, 64

Parameters

for the accountStatement URL, 105, 106

for the betHistory URL, 124, 128, 129

for the betSnapshot URL, 128, 129

for the channelHistory URL, 51

for the channelSnapshot URL, 36, 37, 38, 40

ping element, 27
Pinging the Games API, 24, 25, 26, 27, 28, 134
Poker
  cards, 52, 58, 61
  community cards, 52, 61
  dead heat rule, 57
  hands, 58
  tied games, 57
POST requests, 23, 106, 109
postBetOrder element, 108, 109, 137, 138
postTransferOrder element, 106
Potential profit/loss, 40
resource element, 27
responseTransferOrder element, 106
REST architecture, 15, 24
Root element, 22, 24, 25, 26, 27
Schema, 16, 19, 24, 27, 30, 52, 136
schema element, 27
selection
  element for Baccarat, 82
  element for Blackjack, 71
  element for Hi Lo, 96
  element for Poker, 59
  statuses for Baccarat, 83
  statuses for Blackjack, 71
  statuses for Hi Lo, 96
  statuses for Omaha Hi, 64, 65
  statuses for Poker, 59
selection element, 45, 86
selections element, 45, 86
selectionTradeActivity, 40
Services available from the Games API, 26
sidebets, 45, 86
SSL, 25
TCP/IP port, 131
Testing a client application, 28
Time-zone codes, 120, 121, 129
Total-size betting, 108, 111
Transferring money between accounts, 106
Turbo games, 29
Types of betting market
  ANY_NUMBER_OF_WINNERS, 35
  SINGLE_WINNER_OR_TIE, 35
  WIN_ONLY, 35
Usernames and passwords, 20, 22, 23, 24, 27, 28, 50, 99, 109, 131, 132, 134, 135
userStatusEnum strings, 135
variable handicap markets, 47
WIN_ONLY, 35
XSD document, 16

?username=*username*?username=*username*?username=*username*=sampleuser