



Active Market API v3

Document Revision 1.1
Date of Issue: 16 January 2019
Date of revision: 18 April 2019

Nick Palmer
Product Manager

Table of Contents

1. Purpose	3
2. Glossary of Terms	3
3. Technical Standards	3
4. Request Header	4
5. API Listing	5
5.1 Active Markets service (POST method)	5
6. Response Codes	12
6.1 Request validation error codes	12
6.2 HTTP Status codes	13
7. Appendix 1 – Special contracts types	14
8. Appendix 2 – Longer API response	15

1. Purpose

To provide the API end point information and examples of the web services available for Active Markets.

2. Glossary of Terms

Term	Meaning
LWIN	LWIN - the Liv-ex Wine Identification Number – serves as a universal wine identifier for the wine trade. LWIN is a unique seven to eighteen-digit numerical code that can be used to quickly and accurately identify a product. LWIN allows wine companies to keep their preferred naming system, while introducing a new universal code.
Wine	The word wine below is referring to a specific wine (the producer and brand, grape or vineyard), vintage and unit size combination.
Bid	A buyer places a bid on the Exchange for buying a certain amount of wine.
Offer	A seller places an offer on the Exchange for selling a certain amount of wine.
Order	Order is a generic term for both bid/offer.
Market Price	The Market Price is based on the cheapest 6 and 12-pack prices advertised by leading merchants in the EU and Switzerland. (Where appropriate, alternative unit sizes are used for the calculation.) It provides a guide as to the price you are likely to pay for SIB-compliant stock in the market
SIB	Standard in Bond trade terms: http://www.liv-ex.com/staticPageContent.do?pageKey=Rules_and_Regulations
SEP	Standard En Primeur: http://www.liv-ex.com/staticPageContent.do?pageKey=Rules_and_Regulations
Special	Special contract trade terms: http://www.liv-ex.com/staticPageContent.do?pageKey=Rules_and_Regulations
Contract Type	Contract type is a generic term for SIB, SEP or Special (X).

3. Technical Standards

- Permitted users will be issued with a unique token (CLIENT_KEY) and password (CLIENT_SECRET) combination to control the access for all the web services covered under Exchange Integration.
- The web services will consume and produce both XML and JSON. The user can provide the content type in the request header. If the user does not provide any information, then the default content type will be JSON.
- The project will support ISO 8601.
- The project will only support HTTPS protocol for client and server communications.
- The API will support the following methods:
 - POST for create operation
- Pretty printing for output readability only is supported if required
- Compression for bandwidth savings are used
- Authentication mechanism will be custom based on CLIENT_KEY and CLIENT_SECRET
- The APIs will be accessible at <https://api.liv-ex.com/> followed by their specific base URIs

4. Request Header

This information will be used to authenticate valid access to the REST API. Each user will have to provide the following information in the request header.

Param

Name	Mandatory	Description
CLIENT_KEY	Y	A valid merchant GUID which will be unique for each merchant.
CLIENT_SECRET	Y	Password/Secret for the merchants CLIENT_KEY.
ACCEPT	Y	Accept header is a way for a client to specify the media type of the response content it is expecting. The values for the content type will be application/json or application/xml. If no/ invalid content type is found in the request, then JSON format will be used by default.
CONTENT-TYPE	Y for POST requests	Content-type is a way to specify the media type of request being sent from the client to the server. The values for the content type will be application/json or application/xml. If no/ invalid content type is found in the request, then JSON format will be used by default.

e.g.

CLIENT_KEY: 94B5CC70-BC3D-49C3-B636-C3C7552E543D

CLIENT_SECRET: merchantpasswd

ACCEPT: application/json

CONTENT-TYPE: application/json

Invalid header JSON response

```
{
  "status": "Unauthorized",
  "statusCode": "401",
  "message": "Request was unsuccessful",
  "livexCode": "R000"
  "apiInfo": {
    "version": "3.0",
    "timestamp": 1518524979121,
    "provider": "Liv-ex"
  }
}
```

Invalid header XML response

```
<Response>
  <Status>Unauthorized</Status>
  <HttpCode>401</Code>
  <Message>Request was unsuccessful.</Message>
  <LivexCode>R001</LivexCode>
  <ApiInfo>
    <Version>3.0</Version>
    <Timestamp>2017-11-04T11:12:30</Timestamp>
    <Provider>Liv-ex</Provider>
  </ApiInfo>
</Response>
```

5. API Listing

5.1 Active Markets service (POST method)

Description

This service will be used to request live exchange information for a given LWIN code. On receipt of one or more valid LWIN codes, the service will return information on live contracts and any associated bid or offer prices active on the Liv-ex platform.

Base URI

[exchange/v3/activeMarket](#)

Request Parameters

Name	Mandatory	Description
lwin	Y	LWIN11, LWIN16 or LWIN18. A maximum of 50 LWINs are permitted per request. Type: integer
priceType	Y	Values: 'all', 'bid', 'offer', 'list' Type: alphanumeric
contractType	Y	Values: 'all', 'SIB', 'SEP', 'X' Type: alphanumeric
fullDepth	Y	Controls the level of bid/offer depth returned in the response. Type: boolean true/false true = all live bids, all live offers false = best bid and best offer only
timeframe	N Default = 'current'	Applies to priceType 'list' only. Determines the number of days to return list data for. Values = 'current', '15', '30', '45', '90' Type: alphanumeric
currency	Y	Values: 'gbp', 'eur', 'chf', 'usd', 'hkd', 'jpy', 'sgd', 'gbp/btt', 'eur/btt', 'chf/btt' Type: alphanumeric
fxType	N Default = 'spread'	Display prices using either the Liv-ex spread FX or spot rate FX. Applies to non-GBP currency requests only.

		Values: 'spread', 'spot' Type: alphanumeric
includeListMarkets	N Default = false	<i>A feature used by the Liv-ex UI.</i> If true, API will respond with an extended list of markets where there are also one or more valid list prices available. Type: Boolean true/false

Sample Request Body

JSON Request (single order)

```
{
  "lwin": ["10097692009"],
  "priceType": ["all"],
  "contractType": ["all"],
  "fullDepth": true,
  "currency": "GBP"
}
```

JSON Request (multiple orders)

```
{
  "lwin": ["10097692009", "10097692010", "10097692011"],
  "priceType": ["all"],
  "contractType": ["all"],
  "fullDepth": true,
  "currency": "GBP"
}
```

XML Request (single order)

```
<ActiveMarket>
  <lwin>10097692009</lwin>
  <priceType>all</priceType>
  <contractType>all</contractType>
  <fullDepth>true</fullDepth>
  <currency>GBP</currency>
</ActiveMarket>
```

XML Request (multiple orders)

```
<ActiveMarket>
  <lwin>10097692009</lwin>
  <lwin>10097692010</lwin>
  <lwin>10097692011</lwin>
  <priceType>all</priceType>
  <contractType>all</contractType>
  <fullDepth>true</fullDepth>
  <currency>GBP</currency>
</ActiveMarket>
```

Sample Response Body

The Active Markets service will respond with HTTP Code 200 OK in a successful response. The breadth of data returned will be dependent on the LWIN code length requested:

LWIN11

Response contains all active markets across all live contracts, pack and bottle combinations for the requested wine and vintage.

LWIN16

Response contains all active markets across all live contracts for the specified bottle size for the requested wine and vintage.

LWIN18

Response contains all active markets across all live contract for the specified bottle and pack size for the requested wine and vintage.

Response parameters

Name	Description
lwin	LWIN11 / LWIN18 Type: integer
wineName	The LWIN wine name description e.g. 'Grand Puy Lacoste' Type: alphanumeric
lwinRegion	The LWIN region description e.g. 'Bordeaux' Type: alphanumeric
vintage	Type: 4-digit integer
currency	The currency format of all pricing data contained in the response Type: alphanumeric
iwp	link to the IWP page of the LWIN returned. Type: alphanumeric
packSize	The LWIN pack size description e.g. '06' Type: 2-digit integer
bottleSize	The LWIN bottle size description e.g. '00750' Type: 5-digit integer
contractType	All, SIB, SEP, X Type: alphanumeric
dutyPaid	<i>null unless contractType = X</i> States whether stock offered on Special contract has a tax status of duty paid. If 'false' stock should be considered In Bond (IB). Type: Boolean 'true' or 'false'
minimumQty	States whether the seller has placed a minimum volume of units on the trade. . Standard Liv-ex terms (SIB) is a minimum quantity of 1. Type: integer
deliveryPeriod	States the lead time on the offer in weeks. Standard Liv-ex terms (SIB) are 2 weeks. If deliveryPeriod = 0, the offer is Special Now Type: integer
condition	<i>null unless contractType = X</i> Text field that states any issues with the stock or its packaging. Type: string

photoid	The photo resource ID Type: integer
lowResUrl	HTTP link to a low resolution version of the photo resource Type: alphanumeric
highResUrl	HTTP link to a high resolution version of the photo resource Type: alphanumeric
marketPrice	Type: double
lastTradePrice	Type: double
lastTradeDate	Type: date (epoch time if JSON)
priceType	Values: 'bid', 'offer', 'list' Type: string
price	Type: double
quantity	Type: integer
isBest	The position is the leading live bid/offer for the specific product and contract type. For priceType = 'list' this attribute will show as null. Type: Boolean true/false
myPosition	The bid/offer position is owned by the merchant calling the service. Type: Boolean true/false
orderGUID	The GUID of the bid/offer position. For priceType = 'list' this attribute will show as null. Type: 128-bit hexadecimal
priceDate	Type: date (epoch time if JSON)

JSON Response

The response is sent per request. The example below shows one live active market for the requested LWIN11 (SIB contract, 12x75 pack/bottle size) featuring one live bid position, one live offer position, and one list price.

A more complex response payload featuring multiple active markets is shown in Appendix 1.

```
{
  "status": "OK",
  "httpCode": "200",
  "message": "Request completed successfully.",
  "internalErrorCode": "R001",
  "apiInfo": {
    "version": "3.0",
    "timestamp": 1525969616721,
    "provider": "Liv-ex"
  },
  "activeMarket": [
    {
      "lwin": "10143072010",
      "wineName": "Pontet Canet",
      "lwinRegion": "Bordeaux",
      "vintage": "2010",
      "currency": "GBP",
      "iwp": "https://web.liv-ex.com/individualWine.do?vinWinId=58262",
      "market": [
        {
```

```

    "lwin": "101430720101200750",
    "packSize": "12",
    "bottleSize": "00750",
    "contractType": "SIB",
    "special": {
      "dutyPaid": false,
      "minimumQty": 1,
      "deliveryPeriod": 2,
      "condition": null,
      "photos": []
    },
    "marketPrice": 1500,
    "lastTradePrice": 1750,
    "lastTradeDate": 1524928616000,
    "depth": {
      "bids": {
        "bid": [
          {
            "priceType": "bid",
            "price": 1470,
            "quantity": 4,
            "isBest": true,
            "myPosition": false,
            "orderGUID": "c95e8b6f-e7c9-44fd-ba8a-68b6f765a1d6",
            "priceDate": 1525727448000
          }
        ]
      },
      "offers": {
        "offer": [
          {
            "priceType": "offer",
            "price": 1750,
            "quantity": 13,
            "isBest": true,
            "myPosition": false,
            "orderGUID": "8feebd4b-d987-4cfe-93be-f0b8b0199292",
            "priceDate": 1525969168000
          }
        ]
      },
      "lists": {
        "list": [
          {
            "priceType": "list",
            "price": 1800,
            "quantity": 2,
            "isBest": null,
            "myPosition": false,
            "orderGUID": null,
            "priceDate": "1525073812000"
          }
        ]
      }
    },
    "errors": null
  },
  "errors": null
}

```

Invalid JSON response

```
{
  "status": "Bad Request",
  "httpCode": "400",
  "message": "Request was unsuccessful.",
  "internalErrorCode": "R000",
  "apiInfo": {
    "version": "3.0",
    "timestamp": 1518539969741,
    "provider": "Liv-ex"
  },
  "activeMarket": [
    {
      "lwin": null,
      "wineName": null,
      "lwinRegion": null,
      "vintage": null,
      "currency": null,
      "iwp": null,
      "market": null,
      "errors": {
        "error": [
          {
            "code": "V064",
            "message": "invalid / incorrect lwin and vintage : [10097112009] combination."
          }
        ]
      }
    }
  ]
}
```

XML Response

The response is sent per request. The example below shows one live active market for the requested LWIN11 (SIB contract, 12x75 pack/bottle size) featuring one live bid position, one live offer position, and one list price.

A more complex response payload featuring multiple active markets is shown in Appendix 1 (JSON format).

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<activeMarketResponses xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="https://api.liv-ex.com/v1 https://api.liv-ex.com/schema/v1/services.xsd">
  <Status>OK</Status>
  <HttpCode>200</HttpCode>
  <Message>Request completed successfully.</Message>
  <InternalErrorCode>R001</InternalErrorCode>
  <ApiInfo>
    <Version>3.0</Version>
    <Timestamp>2018-02-13T16:39:09.960Z</Timestamp>
    <Provider>Liv-ex</Provider>
  </ApiInfo>
  <activeMarket>
    <lwin>10097692009</lwin>
    <wineName>Figeac</wineName>
    <lwinRegion>Bordeaux</lwinRegion>
    <vintage>2009</vintage>
    <currency>GBP</currency>
    <iwp>https://qa.liv-ex.com/individualWine.do?vinWinId=56560</iwp>
    <market>
      <lwin>100976920091200750</lwin>
      <packSize>12</packSize>
      <bottleSize>00750</bottleSize>
```

```
<contractType>SIB</contractType>
<special>
  <dutyPaid>>false</dutyPaid>
  <minimumQty>1</minimumQty>
  <deliveryPeriod>2</deliveryPeriod>
  <condition>>null</condition>
</special>
<marketPrice>1761.0</marketPrice>
<lastTradePrice>1700.0</lastTradePrice>
<lastTradeDate>2017-04-19T00:00:00Z</lastTradeDate>
<depth>
  <bids>
    <bid>
      <priceType>bid</priceType>
      <price>1400</price>
      <quantity>1</quantity>
      <isBest>>true</isBest>
      <myPosition>>false</myPosition>
      <orderGUID>50408474-5a04-4a49-b310-618fc8b45981</orderGUID>
      <priceDate>2017-12-01T12:16:30Z</priceDate>
    </bid>
  </bids>
  <offers>
    <offer>
      <priceType>offer</priceType>
      <price>1700</price>
      <quantity>1</quantity>
      <isBest>>true</isBest>
      <myPosition>>false</myPosition>
      <orderGUID>65e36ad7-13bf-4f71-8a05-aa770b899914</orderGUID>
      <priceDate>2018-02-13T16:24:03Z</priceDate>
    </offer>
  </offers>
  <lists>
    <list>
      <priceType>list</priceType>
      <price>1800</price>
      <quantity>2</quantity>
      <isBest>>null</isBest>
      <myPosition>>false</myPosition>
      <orderGUID>>null</orderGUID>
      <priceDate>2018-02-13T16:24:03Z</priceDate>
    </list>
  </lists>
</depth>
</market>
</activeMarket>
</activeMarketResponses>
```

Invalid XML Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<activeMarketResponses xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="https://api.liv-ex.com/v1 https://api.liv-ex.com/schema/v1/services.xsd">
  <Status>Bad Request</Status>
  <HttpCode>400</HttpCode>
  <Message>Request was unsuccessful.</Message>
  <InternalErrorCode>R000</InternalErrorCode>
  <ApiInfo>
    <Version>3.0</Version>
    <Timestamp>2018-02-13T16:47:51.544Z</Timestamp>
    <Provider>Liv-ex</Provider>
  </ApiInfo>
  <activeMarket>
    <errors>
      <error>
        <code>V064</code>
        <message>invalid / incorrect lwin and vintage : [10097112009] combination.</message>
      </error>
    </errors>
  </activeMarket>
</activeMarketResponses>
```

6. Response Codes

This section describes the response codes that will be returned by the Exchange Integration services.

Code	Message
R000	Request was unsuccessful
R001	Request completed successfully
R002	Request partially completed

6.1 Request validation error codes

Code	Message
V000	Mandatory field missing
V001	Merchant is not allowed to access the requested feed
V002	Invalid parameter(s)
V003	Wrong date format. Date should be 'yyyy-MM-dd'
V004	Invalid number parameter: positive number expected for <paramName>
V005	Merchant is not active
V006	Invalid LWIN number
V012	Invalid request headers. Please provide value for header <header name>
V013	Please provide valid vintage

V018	Mandatory field missing (<fieldName>).
V039	You do not have permission to access (<priceType>). Please contact Liv-ex
V058	Invalid / incorrect LWIN: [<value>]. Please provide a valid LWIN11, LWIN16 or LWIN18 code.
V059	Invalid / incorrect contractType: [<value>]. Possible values are 'sib', 'sep', 'x' or 'all'.
V060	Invalid / incorrect priceType: [<value>]. Possible values are 'bid', 'offer' or 'all'.
V061	invalid / incorrect currency: [<value>]. Possible values are 'gbp', 'eur', 'chf', 'usd', 'hkd', 'jpy', 'sgd', 'gbp/btt', 'eur/btt', 'chf/btt'."
V062	invalid / incorrect fxType: [<value>]. Possible values are 'spread' or 'spot'.
V063	invalid / incorrect fullDepth: [<value>]. Possible values are 'true' or 'false'.
V064	invalid / incorrect lwin and vintage: [<value>] combination

6.2 HTTP Status codes

HTTP defines a bunch of meaningful status codes that can be returned from our API. These can be leveraged to help our API Merchants/consumers route their responses accordingly:

Code	Message
200 OK	Response to a successful GET, POST, PUT, DELETE. Can also be used for a POST that doesn't result in a creation.
201 Created	Response to a POST that results in a creation.
202 Accepted	The request has been accepted and will be processed later. It is a classic answer to asynchronous calls (for better UX or performances).
204 No Content	Response to a successful request that won't be returning a body (like a DELETE request)
400 Bad Request	The request is malformed, such as if the body does not parse
401 Unauthorized	When no and/or invalid authentication details are provided. Can also be used to trigger an auth popup if API is used from a browser
403 Forbidden	When authentication succeeded but authenticated user doesn't have access to the resource
404 Not Found	When a non-existent resource is requested
405 Method Not Allowed	When an HTTP method is being requested that isn't allowed for the authenticated user

406 Not Acceptable	Nothing matches the Accept-* Header of the request. As an example, you ask for an XML formatted resource, but it is only available as JSON.
409 Conflict	Indicates one or more supplied parameters are triggering a validation error. A relevant TR code should be returned in the response.
410 Gone	Indicates that the resource at this end point is no longer available. Useful as a blanket response for old API versions
415 Unsupported Media Type	If incorrect content type was provided as part of the request
422 Unprocessable Entity	Used for validation errors. Should be used if the server cannot process the entity, e.g. if an image cannot be formatted or mandatory fields are missing in the payload.
429 Too Many Requests	When a request is rejected due to rate limiting
500 Internal Server Error	The general catch-all error when the server-side throws an exception. The request may be correct, but an execution problem has been encountered at our end.

7. Appendix 1 – Special contracts types

Special contracts (contractType = 'X') carry four attributes that define the tax status, minimum volume, lead time and condition of a specific offer. Attributes can be combined in various ways depending on the status of the stock.

Special attribute	Meaning
dutyPaid	States whether the stock offered on the Special contract has a tax status of duty paid or not. If set to 'false' stock should be considered In Bond (IB). Type: Boolean 'true' or 'false'
minimumQty	States whether the seller has placed a minimum volume of units on the trade. E.g the seller has 50 units on offer with a minimumQty value of 10. Type: integer
deliveryPeriod	States whether the lead time on the offer is different to the standard Liv-ex terms of 2 weeks. If deliveryPeriod = 0, the offer is Special Now i.e. the stock is in the Liv-ex warehouse, has been checked and is ready for immediate dispatch. Type: integer
condition	A free text field that states any issues with the stock or its packaging. Type: string

Some wine offered under a Special contract can match or exceed the Liv-ex SIB terms. Offers listed as 'Special – Now' on the exchange are the equivalent of Standard In Bond (SIB) but have the added benefit of being ready for immediate dispatch from Liv-ex warehouses.

8. Appendix 2 – Longer API response

The JSON response below shows a more complex response payload featuring multiple markets across multiple contract types for a single LWIN11 call. If the request was to contain multiple LWINs, the response below would feature similar payload blocks for each LWIN.

```
{
  "status": "OK",
  "httpCode": "200",
  "message": "Request completed successfully.",
  "internalErrorCode": "R001",
  "apiInfo": {
    "version": "3.0",
    "timestamp": 1525969616721,
    "provider": "Liv-ex"
  },
  "activeMarket": [
    {
      "lwin": "10143072010",
      "wineName": "Pontet Canet",
      "lwinRegion": "Bordeaux",
      "vintage": "2010",
      "currency": "GBP",
      "iwp": "https://web.liv-ex.com/individualWine.do?vinWinId=58262",
      "market": [
        {
          "lwin": "101430720101200750",
          "packSize": "12",
          "bottleSize": "00750",
          "contractType": "SIB",
          "special": {
            "dutyPaid": false,
            "minimumQty": 1,
            "deliveryPeriod": 2,
            "condition": null,
            "photos": []
          },
          "marketPrice": 1500,
          "lastTradePrice": 1750,
          "lastTradeDate": 1524928616000,
          "depth": {
            "bids": {
              "bid": [
                {
                  "priceType": "bid",
                  "price": 1470,
                  "quantity": 4,
                  "isBest": true,
                  "myPosition": false,
                  "orderGUID": "c95e8b6f-e7c9-44fd-ba8a-68b6f765a1d6",
                  "priceDate": 1525727448000
                }
              ]
            },
            "offers": {
              "offer": [
                {
                  "priceType": "offer",
                  "price": 1750,
                  "quantity": 13,
                  "isBest": true,
                  "myPosition": false,

```

```

        "orderGUID": "8feebd4b-d987-4cfe-93be-f0b8b0199292",
        "priceDate": 1525969168000
      },
      {
        "priceType": "offer",
        "price": 1775,
        "quantity": 1,
        "isBest": false,
        "myPosition": false,
        "orderGUID": "8feebd4b-d987-4cfe-93be-f0b8b01992aa",
        "priceDate": 1525969188000
      }
    ],
    "lists": {
      "list": [
        {
          "priceType": "list",
          "price": 1800,
          "quantity": 2,
          "isBest": null,
          "myPosition": false,
          "orderGUID": "272d3450-dad6-479a-b291-617b8ef8dbcb",
          "priceDate": "1525073812000"
        },
        {
          "priceType": "list",
          "price": 1900,
          "quantity": 20,
          "isBest": null,
          "myPosition": false,
          "orderGUID": null,
          "priceDate": 1525969168000
        },
        {
          "priceType": "list",
          "price": 2000,
          "quantity": 1,
          "isBest": null,
          "myPosition": true,
          "orderGUID": null,
          "priceDate": 1525969168000
        }
      ]
    }
  },
  {
    "lwin": "101430720100601500",
    "packSize": "06",
    "bottleSize": "01500",
    "contractType": "SIB",
    "special": {
      "dutyPaid": false,
      "minimumQty": 1,
      "deliveryPeriod": 2,
      "condition": null,
      "photos": []
    },
    "marketPrice": 1450,
    "lastTradePrice": 1500,
    "lastTradeDate": 1524928716123,
    "depth": {
      "bids": {
        "bid": [

```

```

        {
          "priceType": "bid",
          "price": 1470,
          "quantity": 4,
          "isBest": true,
          "myPosition": false,
          "orderGUID": "c95e8b6f-e7c9-44fd-ba8a-68b6f765a1d6",
          "priceDate": 1525727448000
        }
      ],
      "offers": {
        "offer": [
          {
            "priceType": "offer",
            "price": 1600,
            "quantity": 1,
            "isBest": true,
            "myPosition": false,
            "orderGUID": "8feebd4b-d987-4cfe-93be-f0b8b0199292",
            "priceDate": 1525969168000
          },
          {
            "price": 1750,
            "quantity": 1,
            "isBest": false,
            "myPosition": false,
            "orderGUID": "8feebd4b-d987-4cfe-93be-f0b8b01992aa",
            "priceDate": 1525969188000
          }
        ],
        "lists": {
          "list": [
            {
              "priceType": "list",
              "price": 1800,
              "quantity": 2,
              "isBest": null,
              "myPosition": false,
              "orderGUID": "272d3450-dad6-479a-b291-617b8ef8dbcb",
              "priceDate": "1525073812000"
            }
          ]
        }
      },
      {
        "lwin": "101430720100106000",
        "packSize": "01",
        "bottleSize": "06000",
        "contractType": "X",
        "special": {
          "dutyPaid": true,
          "minimumQty": 1,
          "deliveryPeriod": 4,
          "condition": null,
          "photos": []
        },
        "marketPrice": 1000,
        "lastTradePrice": 980,
        "lastTradeDate": 1524928716123,
        "depth": {
          "bids": {
            "bid": [

```

```

        {
          "priceType": "bid",
          "price": 1000,
          "quantity": 2,
          "isBest": true,
          "myPosition": false,
          "orderGUID": "c95e8b6f-e7c9-44fd-ba8a-68b6f7448d81",
          "priceDate": 1525727448000
        }
      ],
      "offers": {
        "offer": []
      },
      "lists": {
        "list": [
          {
            "priceType": "list",
            "price": 1500,
            "quantity": 1,
            "isBest": null,
            "myPosition": false,
            "orderGUID": "272d3450-dad6-479a-b291-617b8ef8dbcb",
            "priceDate": "1525073812000"
          }
        ]
      }
    },
    "errors": null
  }
],
"errors": null
}

```