EZForecourt

Developers

Manual

Version History

Version 1.0.0.0	August 2005
Version 1.2.0.0	November 2005
Version 1.3.0.0	July 2006
Version 2.1.0.0	May 2013
Version 2.3.0.0	May 2016
Version 2.3.0.1	August 2016

1. Introduction

The EZForecourt product is a forecourt controller designed to work in conjunction with third party windows XP/7/8 based POS systems, and possesses the following characteristics.

USB interface.

It utilizes a simple and small, USB based, forecourt interface module (EZMod) to easily connect a windows XP/2000 based server to common gas station forecourt devices. The use of the USB standard does away with the need to install any additional proprietary hardware in the host server.

Self-reliant device.

The EZMod module is a self-powered USB device which has the ability to buffer up to 2950 deliveries if the connection to the host is lost, hence providing redundancy and independence from the server.

Generic high level interface.

It can control and manage many common types and makes of Petrol pumps in a transparent fashion, all the host application sees is a generic pump device.

Electronic tank gauge support.

It also interfaces to many different types and makes of electronic tank gauges, providing a generic and simple interface for both alarms and tank statistics.

ActiveX .NET and DLL client interfaces.

The interface between EZForecourt and the host system/application is done via two ActiveX controls, EZClient.SO.1 and a .NET package EZTech.dll which contains EZTech.EZClient and EZTech.EZPump, or a standard windows a DLL, EZClient.DLL. Which of the three options is utilized will depend on the capabilities and requirements of the host system.

TCP/IP client server interface.

The interface between the EZForecourt server and client applications is done via TCP/IP sockets, hence the clients and the server need not be running on the same physical machine, but can be anywhere on the same local TCP/IP based network.

.

2. Concepts

Prior to starting an integration of EZForecourt, it is necessary that a few of the basic concepts be clarified, to ensure that the more technical information be interpreted correctly. The following terms will be used many times in the API and data base sections later on in this document.

Post-pay delivery

Deliveries on a forecourt generally follow the Post pay sequence of events, the hose is pulled on the pump and the pump is either authorized automatically or manually by a shop or forecourt attendant. When the delivery is completed the customer either pays the attendant for the delivery or pays for it in the shop. This is referred to as a post pay delivery, i.e. a delivery that is paid for after it is taken.

Prepay delivery

An alternative to paying for the delivery after it is taken is paying for it before. In this case the customer goes into the shop, pays a predetermined amount for fuel on a specified pump, this is usually an estimate on behalf of the customer, who then goes back to his car and takes the delivery. When the hose is pulled from the specified pump, it is automatically authorized to delivery up to the prepaid amount. If the delivery is terminated prior to the limit being reached, a refund for the difference is automatically generated and the customer is then required to return to the shop to collect the refund. If however there is no refund the customer is free to leave when the hose is returned.

Preauth delivery

These days many systems give the customer the option of paying for the fuel delivery at the pump, to do this, the pump must be fitted with a card terminal. The sequence of events in this situation is, the customer passes their card thru the card terminal on the respective pump, and then the card terminal starts a pre-authorization with the respective bank. Once this pre-authorization is granted, the pump is authorized to delivery up to the limit returned with the pre-authorization response. When the customer has completed the delivery the actual value of the delivery is sent as part of a pre-authorization advice message billing the customer’s card with the correct value. This is known as a preauth delivery.

Drive off delivery

Unfortunately there are also customers, who take a delivery at a pump and leave without paying, this only occurs in post pay environments. These are politely referred to as drive off deliveries.

Test delivery

At times it is necessary for the various certification authorities to carry out tests on the pumps to verify that their volume measurement conforms to the relevant regulations. When these deliveries are completed the fuel is returned manually returned to the tank from which it came. These deliveries must be treated differently and are referred to as test deliveries.

Monitor deliveries

In a lot of countries gas station forecourts are run fully attended and the customer does not even have to exit the car to fill up the tank. A forecourt attendant takes the delivery on behalf of the customer and the customer pays the attendant directly after the delivery is completed. In this mode the forecourt controller is in essence only monitoring the forecourt activity, and is usually used to prevent fraud or theft by the forecourt attendants. These deliveries are referred to as monitor deliveries.

Offline deliveries

At times the communications with the pumps are lost; this could be because the pumps are turned off, switched into local mode or because of technical faults. When the pumps are returned to normal function and communications are re-established, an offline delivery is generated which is the sum of all of the deliveries which were done while the pump was offline. This total is difference between the last known electronic totals before it stopped responding and the electronic totals retrieved immediately after the pump started to respond. The purpose of this delivery type is to ensure that the electronic and theoretical totals still tally.

Temp stop

Pump deliveries can be controlled remotely via EZForecourt, imagine a situation where someone is smoking near a pump, and due to safety reasons it is necessary to stop the fuel delivery, this is called a ‘Temp stop’. A temp stop remains in place until the pump is re-authorized manually via EZForecourt. All of the pumps can be stopped in single operation; this is called an ‘All Stop’, and can be reversed by an ‘All Re-authorize’.

Pump authorize

EZForecourt permits pumps to run in several authorization modes, the first and most simple is auto authorize. In this mode the pump will start delivering, without intervention, when the hose is pulled. When EZForecourt is running in monitor mode the pumps are also auto authorize. The other mode is compulsory authorize, in this mode when the hose is pulled, the pump will call for authorization, it will only start delivering when a forecourt or shop attendant manually authorizes the pump. If a limit is placed on the number of unpaid deliveries, auto authorize pumps will not automatically authorize, if this limit has been reached. One or more of the unpaid deliveries will need to be processed first. This does not however apply to monitor mode.

Electronic totals

Most pump types keep electronic totals in non-volatile memory; these totals are accumulative totals of both volume and value, for each of the hoses on the pump. These totals are zeroed when the pump is installed or when the pump electronics are reset. These totals (when present) are interrogated remotely by EZForecourt and are continually monitored to determine if deliveries are lost etc.

Mechanical totals

Most pump types also have mechanical totals for both volume and value for each hose present on the pump. Unlike the electronic totals these cannot be reset, or interrogated remotely. EZForecourt permits the manual entry of these totals. This is yet another total to be reconciled against the EZForecourt totals.

Theoretical totals

Besides the mechanical and electronic totals maintained by the pumps, EZForecourt also maintains a theoretical total for both value and volume for each of the hoses configured. This is maintained by simply accumulating the individual delivery totals as they are completed. Reconciling the theoretical, electronic and mechanical totals is ones best defense against theft or fraud.

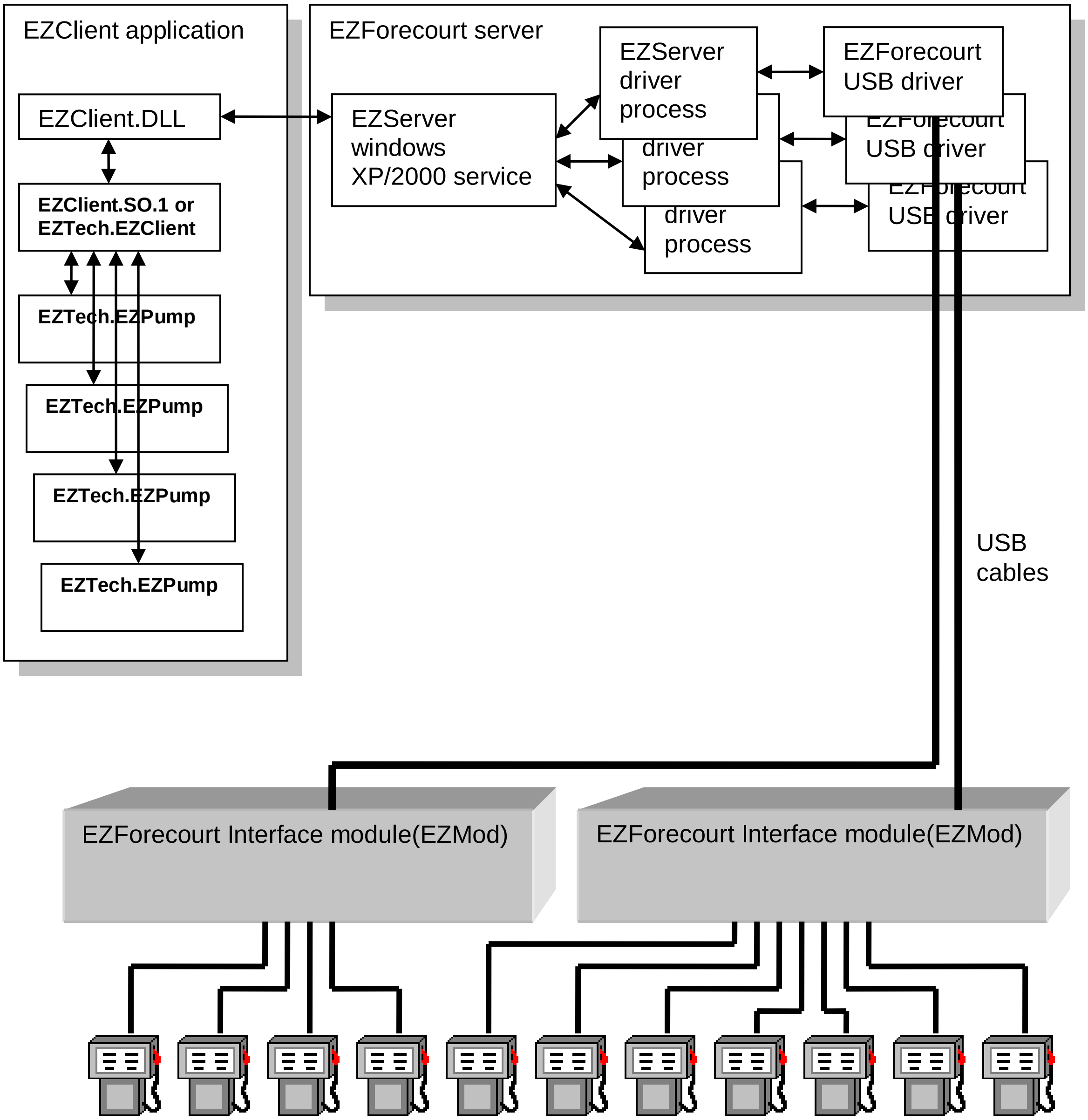
Price level

Most modern pump types today support more than one price per grade, this was originally intended to be used in credit or cash situations, however it can also be used for full vs. self-service etc. Even though most pumps do not display both prices on the pump displays, they are saved internally and can be changed and selected remotely. EZForecourt supports multiple prices per grade and up to two prices per hose.

It is assumed that the reader is familiar with windows XP/2000 and general programming concepts and as such there is no explanation of this here.

3. System Architecture

The EZForecourt system architecture is as follows:



As can be seen from the diagram above, EZForecourt supports more than one EZMod module. This may be required if there is more than eight physical pumps or a mixture of pump types (different manufacturers) that cannot be hosted on the same EZMod module.

It also clearly demonstrates how the optional back office and database fit into the overall structure. If the integrator decides to use their own existing back office or develop a new one, then a similar structure will be required.

It may not be evident here but the links between the BackOffice and EZServer are also via the EZClient.DLL, and in the interests of speed and data integrity it is a requirement that the BackOffice and EZServer be hosted on the same machine. Other EZClients (POSes etc.) do not need to be on the same machine, but can be anywhere on the same TCP/IP network.

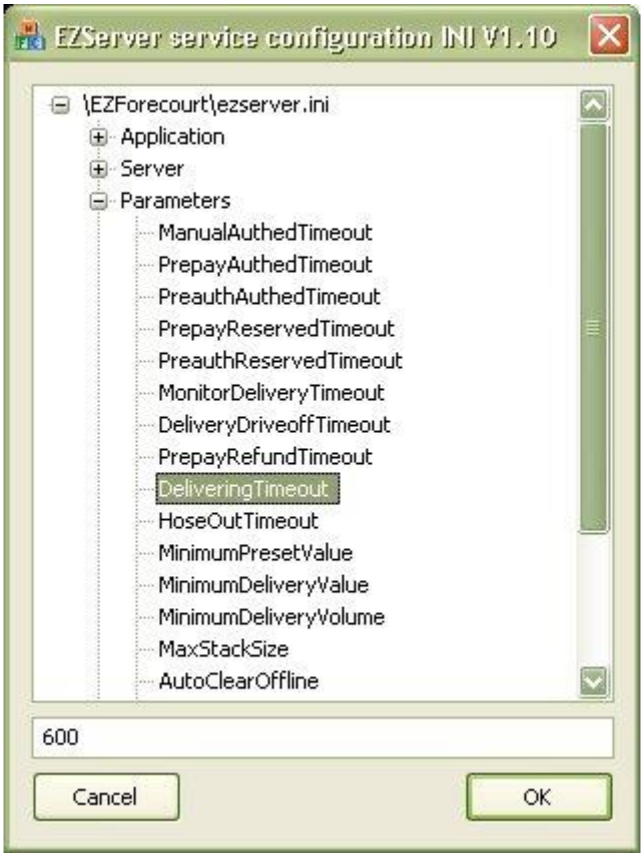
4. Installation

For a detailed description of how to install the EZForecourt product refer to the EZForecourt Installers Manual.

5. INI file configurations

Each of the EZForecourt software components is individually configured with a windows format INI file. These INI files all follow a standard format and are easily edited using the EZIniEditor utility provided. These INI files are installed per-configured and should only be edited under the direct instructions of EZTech support.

All of these INI files are located in the \EZForecourt directory. To edit an INI file simply open up a command window, change to the EZForecourt directory and then type EZIniEditor followed by the name of the INI file (with or without the INI extension). This will result in a screen similar to the following being presented:



```
\EZForecourt\ezserver.ini
    Application
    Server
    Parameters
        ManualAuthedTimeout
        PrepayAuthedTimeout
        PreauthAuthedTimeout
        PrepayReservedTimeout
        PreauthReservedTimeout
        MonitorDeliveryTimeout
        DeliveryDriveoffTimeout
        PrepayRefundTimeout
        DeliveringTimeout
        HoseOutTimeout
        MinimumPresetValue
        MinimumDeliveryValue
        MinimumDeliveryVolume
        MaxStackSize
        AutoClearOffline
600
```

If the INI file was located successfully, the name of the INI file will be displayed with a plus symbol beside the full file name, clicking the plus symbol will expand it to show all of the sections contain in the INI file. Clicking the plus symbol beside each of the sections will expand it to show all of the individual configuration parameters.

The value for the currently highlighted configuration parameter is displayed in the text box at the bottom of the window. To change the currently selected configuration value, simply edit it in the text box, saving is done automatically when a different configuration parameter is selected or the application is exited by clicking the OK button.

There are some sections and configuration values which are common to all of the EZForecourt INI files these are as follows.

Section	Parameter	Description
[Application]	Name	The name of the application to which this INI file applies.
	IniVersion	The version number for this INI file
[Server]	Name	The machine name of the machine hosting the EZServer service, if the service is on the same machine then the value <local> can be used.
	CallPort	The TCP/IP port number that is used by EZClients to access the server. This

		number must be the same for all EZClients and the EZServer.
	EventsPort	The TCP/IP port number that is used by the EZClients to receive events. This number must be the same for all of the EZClients and the EZServer.
	CallTimeout	The timeout value in milliseconds for clients when calling the server, if the server does not respond within this time the call will return with a SERVER_TIMEOUT result.
[Log]	Screen	A 0 or 1 (0=no,1=yes) flag used to determine if diagnostic information is sent to the screen, this is only of use when running the server in debug mode and is reserved for internal EZTech use.
	File	A 0 or 1 (0=no, 1=yes) flag used to determine if diagnostic information is sent to a log file. By default this is 1 and should be left that way, unless instructed to change it by EZTech. Diagnosing faults is considerably more difficult if this flag is turned off.
	Socket	A 0 or 1 (0=no, 1=yes) flag used to determine if diagnostic information is sent to a TCP/IP socket. This feature is currently unsupported, but is reserved for future use.

5.1. EZServer INI file

The EZServer INI file contains all of the configurable parameters for the EZServer service. This INI file contains the standard parameters listed above along with the specific parameters listed below.

Section	Parameter	Type	Description
[Parameters]	ManualAuthedTimeout	Numeric, Seconds	The amount of time, in seconds, that a compulsory authorized pump will remain authorized after manual authorization.
	TagAuthedTimeout	Numeric, Seconds	The amount of time, in seconds, that an EZID authorized pump will remain authorized after Mifare card is read authorization.
	PrepayAuthedTimeout	Numeric	The amount of time, in seconds, that a pump will remain authorized for a prepay delivery, before a prepay refund is

			automatically generated. It the pump starts to deliver before this time is up it will be treated as a prepay delivery.
	PreauthAuthedTimeout	Numeric	The amount of time, in seconds, that a pump will remain authorized for a preauth delivery, before the preauth reserve is automatically removed. It the pump starts to deliver before this time is up it will be treated as a preauth delivery.
	PrepayReservedTimeout	Numeric	The amount of time, in seconds, that a pump will remain reserved for a prepay delivery before the prepay reserve is automatically removed. If the pump is prepay authorized before this time is up, it will change to the prepay authorized state.
	PreauthReservedTimeout	Numeric	The amount of time, in seconds, that a pump will remain reserved for a preauth delivery before the preauth reserve is automatically removed. It the pump is preauth authorized before this time is up, it will change to the preauth authorized state.
	MonitorDeliveryTimeout	Numeric	The amount of time, in seconds, that a Post-pay delivery will remain on a pump, configured in monitor mode, before it is cleared automatically as a monitor delivery. It can be taken as a Post-pay delivery prior to timing out.
	DeliveryDriveoffTimeout	Numeric	The amount of time, in seconds, that a Post-pay delivery can remain on a pump until it is flagged as a potential drive off delivery.
	PrepayRefundTimeout	Numeric	The amount of time, in seconds, that a prepay refund remains on a pump before it is automatically

			cleared from the pump. If this delivery is cleared manually prior to this timeout expiring, the pump will remain locked out for the remainder of this timeout.
	DeliveringTimeout	Numeric	The amount of time, in seconds, that a delivering pump can remain delivering before a warning is generated.
	HoseOutTimeout	Numeric	The amount of time, in seconds, that a pump hose can be left out, not delivering, before a warning is generated.
	LeaveAuthedTimeout	Numeric	The time in seconds to leave a pump in the authorized state before canceling it.
	NotRespondingTimeout	Numeric	The maximum time in seconds that a pump can stop and start responding without being initialized.
	MinimumPresetValue	Numeric	The minimum value accepted by the server for a preset, prepay or preauth delivery.
	MinimumDeliveryValue	Numeric	The minimum delivery value accepted by the server for a valid delivery, deliveries with values smaller than this are discarded.
	MinimumDeliveryVolume	Numeric	The minimum delivery volume accepted by the server for a valid delivery, deliveries with volumes smaller than this are discarded.
	MaxStackSize	Numeric	The maximum number of unpaid deliveries permitted on a pump before the server will stop authorizing the pump. If the server is in standalone mode, or the pumps are in monitor mode, this limit is ignored.
	AutoClearOffline	Numeric	A yes or no value, that determines whether offline deliveries are automatically cleared by the server, or must be taken manually as a Post-pay delivery like any other.

	MinOfflineDeliveryVolume	Numeric	The minimum volume for an offline delivery, if an offline delivery is detected with less than this volume, it is ignored.
	MaxOfflineDeliveryVolume	Numeric	The maximum volume for an offline delivery, if an offline delivery very is detected with more than this it is ignored. This is used to discard offline deliveries generated when used pumps are first connected to the system.
	Standalone	Boolean	A yes or no value to determine if the forecourt interface modules are allowed authorize pumps and buffer deliveries when the server is not running. This is only of use in a pump monitoring environments, and hence is of little use in a self-service situation.
	NonVolObjects	Boolean	A yes or no value to determine if the objects in the EZServer are non-volatile or not.
	NonVolTotals	Boolean	A yes or no value to determine if the hose electronic totals and tank gauge values are saved to the non-vol store or not, it is recommended that it be yes.
	LogDeliveries	Boolean	A yes or no value to determine if the individual deliveries are logged to a separate log file.
	MaxDeliveryVolumeDiff	Numeric	The maximum delivery volume difference between the reported delivery volume and the volume electronic totals difference before an offline delivery is generated.
	MaxDeliveryValueDiff	Numeric	The maximum delivery value difference between the reported delivery value and the value electronic totals difference before an offline delivery is generated.
	MinimalPumpEvents	Boolean	If set to yes this suppresses various pump events including running total events. This should be no.

	ExtendedDeliveryProperties	Boolean	A flag to determine if the DeliveryEvent or DeliveryEventEx event is generated.
	RunningTotalRate	Numeric	The minimum running total update rate for pumps in milliseconds.
	ConfirmDeliveryLog	Boolean	Set this to Yes if you want to use EZLogger of equivalent to extract all of the cleared deliveries to a database etc.
	MinimumKeepDeliveries	Numeric	The minimum number of cleared deliveries that are retrained in the EZserver database, these can be seen in the EZMonitor deliveries history.
	ForceDeliveryLock	Boolean	Set to true to force the server to re-fire pump calling status event every 10 seconds.
	VolumetricDLL	String	The full path and file name for the DLL used to do the volumetric logging.
	ZigBeePanID	Numeric	The PAN ID used for the ZigBee network.
	GetDeliveryTimeout	Numeric	The time in seconds that the EZServer waits after the pump has finished delivering for the EZRemotes to log the delivery.
	GetETotalsTimeout	Numeric	The time in seconds the EZServer waits for the EZRemote waits for the EZRemote to log the e-totals after the delivery has been logged.
	DelMinimumDuration	Numeric	The minimum time in seconds that a pump must be in the delivering state to be considered a valid delivery.
	TagAuthType	Numeric	The type of tag authorization. 0 - Authorizes one fueling point only, and can be authorized prior to pulling the hose for TagAuthedTimeout seconds. 1 – Authorizes one fueling point only, and only if the hose is already pulled. 2 – Authorizes up to two

			fueling points (sides 1 and 3 or 2 and 4). Will authorize the un-authorized hose that has been pulled for the longest time.
	TagCacheOn	Boolean	The pump attendant cache in the EZRemotes enabled, default is yes.
	PortsReadOnly	Boolean	Whether the ports coinfiguration can be edited in the EZConfig application, the default is yes.
	GradesReadOnly	Boolean	Whether the grades coinfiguration can be edited in the EZConfig application, the default is yes.
	TanksReadOnly	Boolean	Whether the tanks coinfiguration can be edited in the EZConfig application, the default is yes.
	PumpsReadOnly	Boolean	Whether the pumps coinfiguration can be edited in the EZConfig application, the default is yes.
	ZigBeeReadOnly	Boolean	Whether the EZRemotes coinfiguration can be edited in the EZConfig application, the default is yes.
	HosesReadOnly	Boolean	Whether the hoses coinfiguration can be edited in the EZConfig application, the default is yes.
	PricesReadOnly	Boolean	Whether the fuel prices coinfiguration can be edited in the EZConfig application, the default is yes.
	AttendantsReadOnly	Boolean	Whether the attendants coinfiguration can be edited in the EZConfig application, the default is yes.
	ClientsReadOnly	Boolean	Whether the Card Clients coinfiguration can be edited in the EZConfig application, the default is yes.
	SensorsReadOnly	Boolean	Whether the Sensors coinfiguration can be edited in the EZConfig application, the default is yes.

5.2. EZClient *INI file*

The EZClient INI file contains all of the configurable parameters for all of the EZServer clients (EZClients). This INI file contains the entire standard parameters listed above but has no specific parameters. The main parameter of interest here is the name parameter in the Server section, in order to configure a remote client this parameter must be set to the machine name of EZServer service host machine.

Section	Parameter	Type	Description
N/A			

5.3. EZDriver INI file

This INI file is also used by the EZServer service; more specifically it is use by the driver DLLs loaded at start up. The parameters in this INI are all pre-configured and should only be altered under specific instructions from EZTech. The following documentation is provided for information purposes only. Under some of the protocol and pump type sections you will find parameters which are not listed here, these are driver specific parameters, for details of these parameters please contact EZTech technical support.

This INI file contains the standard parameters listed above along with the specific parameters listed below.

Section	Parameter	Type	Description
[Protocolnnn]	Name	String	The name of the protocol, nnn is from 001 to 999, and corresponds to the ProtocolID column in the Protocols table of the EZDB database.
	DAL	String	The name of the DAL (Driver Abstraction Layer) DLL, this will be PumpDrv.DLL or TankDrv.DLL depending on the device type.
	Driver	String	The name of the DLL file which is the driver for this protocol, note it must be a fully qualified file name, this is only used by TankDrv.DLL
	LoopType	String	The loop type for this connection COM or SOCKET, this is only used by TankDrv.DLL
	Baudrate	Numeric	The BPS (bits per second) communications rate use for this protocol, any value from 300 to 38400 is valid,

			only valid for LoopType COM.
	DataBits	Numeric	The number of data bits per character. The options are 7 or 8, only valid for LoopType COM.
	Parity	String	The parity bit type. The options are O, E or N, for odd, even or none respectively, only valid for LoopType COM.
	StopBits	Numeric	The number of stop bits trailing each character. The options are 1 or 2, only valid for LoopType COM.
	NotRepsondingPollingRate	Numeric	The delay in milliseconds inserted between polls to devices which are currently not responding.
	MaxNoResposes	Numeric	The maximum number of sequential no responses from a device, before this device is flagged as not responding.
	ResponseTimeout	Numeric	The response timeout in milliseconds.
	IntercharTimeout	Numeric	The inter character timeout in milliseconds.
	StatusPollRate	Numeric	The delay in milliseconds between polls for the tank status.
	AlarmsPollRate	Numeric	The delay in milliseconds between polls for the tank alarms.
	TCPIPPort	Numeric	The TCPIP port number for LoopType = SOCKET.
	AfterReadDelay	Numeric	The delay in seconds after each poll.
	SensorPollRate	Numeric	The rate in seconds that the leak detection sensors are polled.
[PumpTypennn]	Name	String	The pump type name, where nnn can be from 001 to 999, this is the Pump type name as shown in the EZConfig application.
	Protocol	Numeric	The ProtocolID used to communicate with this pump; this links this pump type to the appropriate [Protocolnnn] section. For the PumpDrv this is always 003.

	Driver	String	The name of the DLL file which is the driver for this protocol, note it must be a fully qualified file name.
	ProtocolType	Numeric	The protocol type ID, this number is hard coded and is used by the EZMOD and EZRemote to determine the type of polling for this device.
	LoopType	String	The type of physical connection and hence driver card necessary for this type of pump. The options are CL20 (20ma current loop), CL40 (40ma current loop), RS485, TOKHEIM and NZP.
	Baudrate	Numeric	The BPS (bits per second) communications rate use for this protocol, any value from 300 to 38400 is valid.
	DataBits	Numeric	The number of data bits per character. The options are 7 or 8.
	Parity	Numeric	The parity bit type. The options are O, E or N, for odd, even or none respectively.
	StopBits	Numeric	The number of stop bits trailing each character. The options are 1 or 2.
	MaxHoses	Numeric	The maximum number of hoses supported by this pump type.
	PriceLevels	Numeric	The maximum number of price levels supported by this pump.
	InterPollDelay	Numeric	The delay in milliseconds inserted between polls by the EZMods
	InterCharDelay	Numeric	The delay in milliseconds inserted between characters by the EZMods
	ForceValue	Boolean	Yes or No, to force the delivery value to the difference in the electronic totals.
	ForceVolume	Boolean	Yes or No, to force the delivery volume to the difference in the electronic totals.
	ResponseDelay	Numeric	A delay in milliseconds

			inserted before the MUX responds to a poll.
	TotalsRollOver		Yes or No, to maintain electronic the part of the totals greater than 1,000,000.00 in the EZServer.
	ExtendedTotals	Boolean	Does this protocol support e-totals greater than a million.
	4HoseProtocol	Boolean	Wayne specific flag whether it is the newer 4 hose or older three hose protocol
	BrazilCorrection	Boolean	Wayne specific flag to cover a bug in some Brazilian Wayne pumps.
	ValueMultiple	Numeric	The numeric multiple that is the step size in the delivery total, in cents.
	VolumeETotDecimals	Numeric	The number of decimals that the volume e-totals has for this pump type, if not present the pump volume display format is used.
	ValueETotDecimals	Numeric	The number of decimals that the value e-totals has for this pump type, if not present the pump value display format is used.
	VolumeDecimals	Numeric	The number of decimals that the delivery volume has for this pump type, if not present the pump volume display format is used.
	VolumeDecimals	Numeric	The number of decimals that the delivery value has for this pump type, if not present the pump value display format is used.
	PriceDecimals	Numeric	The number of decimals that the delivery price has for this pump type, if not present the pump price display format is used.

5.4. EZLicense INI file

The EZLicense INI file is used by various EZForecourt modules, it should contain a valid license key as issued by EZTech, this will determine which of the modules are licensed to run and until which date. This INI only contains one application specific configuration key.

Section	Parameter	Type	Description
[Application]	LicenseKey		A 24 hex digit license key as supplied by EZTech.
	SerialNo	String	The numeric part of the serial number used to generate this license key
	ExpirationDate	Date	The expiration date for this license key.

5.5. EZATG INI file

If the EZForecourt is licensed for ATG functionality the following table is to determine is functionality of the ATG software module.

Section	Parameter	Type	Description
[Parameters]	ConfirmEventLog	Boolean	A flag to determine if LogEvents require acknowledgement by a third party software, prior to deletion.
	MaxLogEvents	Numeric	The maximum number of log events stored in the EZForecourt.
	LogConfigChanges	Boolean	Log changes in the EZForecourt configuration.
	AutoClearTimeout	Numeric	The timeout in seconds for alarms to be auto-cleared.
	StoppedRespondingTimeout	Numeric	The timeout in seconds before a tank probe or pump not responding generates an alarm.
	PercentProductHiAlarm	Numeric	The percentage of the tank capacity that above which will generate a high product alarm.
	PercentProductHiWarning	Numeric	The percentage of the tank capacity that above which will generate a high product warning.
	PercentProductLowWarning	Numeric	The percentage of the tank capacity that below

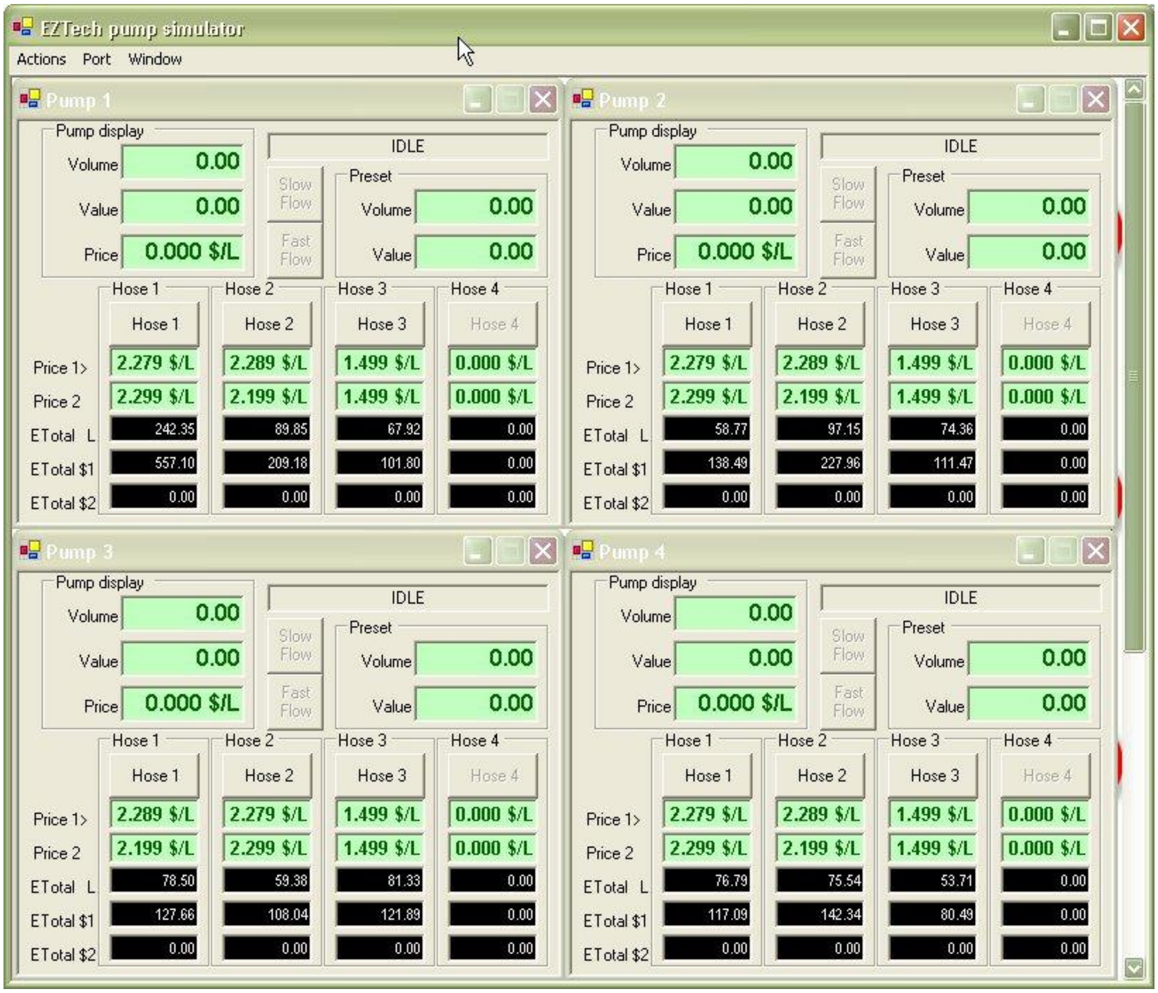
			which will generate a low product warning.
	PercentProductLowAlarm	Numeric	The percentage of the tank capacity that below which will generate a product low alarm.
	WaterLevelHiAlarm	Numeric	The height in mm, that above which will generate a high water alarm
	WaterLevelHiWarning	Numeric	The height in mm, that a water level above which will generate a high water alarm.
	TankReadingInterval	Numeric	The interval in minutes between logging the tank readings as a log event, a multiple of this value must be 60.
	StrappingTableStep	Numeric	The distance in meters between strapping table items.
	DampingFactor	Numeric	The number of sequential readings that are averaged together to get the weighted readings.
	WeightedReadings	Numeric	The number of sequential weighted readings that are used to detect tank drops etc.
	VolumeLevelTolerance	Numeric	The tolerance in meters between readings that the readings are treated as equal.
	MinimumLeakVolume	Numeric	The minimum discrepancy volume in liters that will generate a leak volume.
	MinimumCalibrationErrorVolume	Numeric	The minimum discrepancy volume in liters that will be treated as a tank calibration error.
	MinimumTankDropVolume	Numeric	The minimum volume in liters that will be treated as a tank drop.
	NotRespondingIsIdle	Boolean	Flag to determine of a non-responding pump is treated as idle or not.
	LeakTimeout	Numeric	The timeout in seconds for a static tank to clear a leaking alarm.
	ExtendedLogs	Boolean	Flag to turn extended logs on/off

6. Developer’s Kit contents

The developer’s kit is shipped with a full version of EZForecourt as well as sample programs, pump simulator, USB RS485 adapter, and cable to connect the USB RS845 adapter to the EZMod. The pump simulator requires the USB RS485 adapter and cable in order to function. To install the USB RS485 adapter simply plug it into any available USB port and when the new hardware found dialog appears point it to the \FTDI directory on the install CD.

6.1. The EZSim pump simulator

The Pump simulator is used to assist with developing the host system, when a real pump is not available. It is a windows application that simulates a real pump. The EZSim pump appears as follows:



Pump 1

Pump display: Volume 0.00, Value 0.00, Price 0.000 $/L

IDLE

Preset: Volume 0.00, Value 0.00

	Hose 1	Hose 2	Hose 3	Hose 4
Price 1>	2.279 $/L	2.289 $/L	1.499 $/L	0.000 $/L
Price 2	2.299 $/L	2.199 $/L	1.499 $/L	0.000 $/L
ETotal L	242.35	89.85	67.92	0.00
ETotal $1	557.10	209.18	101.80	0.00
ETotal $2	0.00	0.00	0.00	0.00

Pump 2

Pump display: Volume 0.00, Value 0.00, Price 0.000 $/L

IDLE

Preset: Volume 0.00, Value 0.00

	Hose 1	Hose 2	Hose 3	Hose 4
Price 1>	2.279 $/L	2.289 $/L	1.499 $/L	0.000 $/L
Price 2	2.299 $/L	2.199 $/L	1.499 $/L	0.000 $/L
ETotal L	58.77	97.15	74.36	0.00
ETotal $1	138.49	227.96	111.47	0.00
ETotal $2	0.00	0.00	0.00	0.00

Pump 3

Pump display: Volume 0.00, Value 0.00, Price 0.000 $/L

IDLE

Preset: Volume 0.00, Value 0.00

	Hose 1	Hose 2	Hose 3	Hose 4
Price 1>	2.289 $/L	2.279 $/L	1.499 $/L	0.000 $/L
Price 2	2.199 $/L	2.299 $/L	1.499 $/L	0.000 $/L
ETotal L	78.50	59.38	81.33	0.00
ETotal $1	127.66	108.04	121.89	0.00
ETotal $2	0.00	0.00	0.00	0.00

Pump 4

Pump display: Volume 0.00, Value 0.00, Price 0.000 $/L

IDLE

Preset: Volume 0.00, Value 0.00

	Hose 1	Hose 2	Hose 3	Hose 4
Price 1>	2.279 $/L	2.289 $/L	1.499 $/L	0.000 $/L
Price 2	2.299 $/L	2.199 $/L	1.499 $/L	0.000 $/L
ETotal L	76.79	75.54	53.71	0.00
ETotal $1	117.09	142.34	80.49	0.00
ETotal $2	0.00	0.00	0.00	0.00

In order for the pump simulator to work the server must be configure to use the EZSim protocol for the EZMod in question and the pump types must all be set to EZSim.

To start a delivery simply click on any of the enabled hose, to end the delivery simply click the hose button again.

6.1.1. EZSim INI file

This EZSim INI file is used by the EZSim pump simulator, as this is not an EZClient it does not have a [Server] or [Log] section. A lot of the parameters for the EZSim application are updated by the application when it is exited; as such editing them is of little benefit. The application specific configuration parameters are as follows.

Section	Parameter	Type	Description
[Parameters]	NumberOfPumps		The number of pumps currently configured for the simulator this can be from 1 to 16.
	PriceDecimals		The number of decimal places used for the price display and price fields in the communications protocol.
	ValueDecimals		The number of decimal places used for the value display and value fields in the communications protocol.
	VolumeDecimals		The number of decimal places used for the volume display and volume fields in the communications protocol.
	CommPort		The com port used by this simulator
	SlowRate		The amount added to the delivery volume every time a DelTimeInterval is passed when the pump is in slow flow mode.
	FastRate		The amount added to the delivery volume every time a DelTimeInterval is passed when the pump is in fast flow mode.
	DelTimerInterval		The amount of time in milliseconds that is used between updates of the volume delivered when the pump is delivering.
	PumpsPerRow		The number of pumps display per row in the

			EZSim window.
	HorizontalSpace		The number of pixels between each of the pump simulators horizontally.
	VerticalSpace		The number of pixels between each of the pump simulators vertically.
[PumpNNHoseY]	VolumeETot		The electronic volume total for pump NN hose Y
	Value1ETot		The electronic value total for pump NN hose Y price level 1
	Price1		The current price for price level 1
	Value2ETot		The electronic value total for pump NN hose Y price level 2
	Price2		The current price for price level 2

6.2. Sample applications

The developer’s kit is shipped with three demonstration applications to assist in the integration of the EZForecourt into the host system. These applications have no copyrights applied to them and can be copied or modified as desired. EZTech are adding example applications all the time, if your requirements are not covered by the following samples, enter into contact with EZTech support.

6.2.1. EZDemoPos

The EZDemoPos is a VB.net application that acts as a POS terminal and demonstrates all of the functions that are possible with EZForecourt. It uses the EZTech.DLL .NET package.

6.2.2. EZClientCSharp

The EZClientCSharp is a more basic example and was also developed in VS 2008 C#, it demonstrates how to call the EZClient DLL directly.

6.2.3. EZClientCpp.

The EZClientCpp application is an example developed in VS 2008 cpp and it calls the EZClient.DLL directly. There is also a Linux version which uses GNU Gcc and calls the EZClient.SO.1 shared object.

6.2.4. EZClientDelphi7 / EXClientDelphiXe2.

The EZClientDelphi applications are examples developed in Delphi 7 and Xe2 and call the EZClient.DLL directly.

7. API reference

7.1. API Components

The API (application interface) for the EZForecourt product consists of five components:

- EZClient.DLL, a DLL which exposes the complete API to all types of applications. This DLL is not a COM DLL and as such does not require registering or require COM support capabilities of its host. The API for this is defined in the EZClient.h and EZClient.lib files included with the developer’s kit.
- EZTech.EZClient .NET control contained in the EZTech.DLL .NET package this control exposes the complete API to .NET applications such as VB.NET, C#, and C++ .NET with the managed C++ extensions.
- EZTech.EZPump a visual .NET control contained in the EZTech.DLL .NET package, this control exposes the pump specific parts of the API. It requires that the EZTech.EZClient control also be hosted in the same application. This control is a visual control, and has a limited user interface of its own. The is also the EZTech.EZTank visual control which exposes the tank specific parts of the API
- There are 64 bit versions of the EZTech.DLL and EZClient.DLL, For the EZClient.DLL it is the EZCLient64.DLL for the EZTech.DLL it is the DLL found in the \EZForecourt\x64 directory.
- There is 32 bit EZTech.SO.1 for Linux applications. It exposes the same APIs as the EZClient.DLL
- There is also a Web Service interface as defined in the file EZSoap.wsdl

The API calls are largely common across all five components and will be addressed just once in this document; any differences between the five interfaces will be highlighted.

7.2. Data types

The API to the EZForecourt was designed using data types that are supported across as many platforms and development environments as possible. The basic data types are as follows:-

Int32	32 bit Integer, range –2,147,483,648 to 2,147,483,647
Int16	16 bit Integer, range –32,768 to 32,767
Int64	64 bit Integer, range -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807
DateTime	64 bit floating point value representing the number of days since January 1, 100. It has an approximate resolution of 1 millisecond up to December 31, 9999.
Double	64 bit standard floating point type
String	Variable length Unicode strings , i.e. 16 bit characters

The declaration of these data types varies depending on the development platform.

	C# Windows	C/C++ Windows	Delphi	VB.net	C/C++ Linux
Int64	Int64	__int64	Int64	ByVal Int64	long long
Int64*	ref Int64	__int64*	PInt64	ByRef Int64	long long*
Int32	Int32	long	Integer	ByVal Int32	long
Int32*	ref Int32	long*	PInteger	ByRef Int32	long*
Int16	Int16	short	Smallint	ByVal Int16	short
Int16*	ref Int16	short*	PSmallint	ByRef Int16	short*
double	Double	double	Double	ByVal Double	double
double*	ref Double	double*	PDouble	ByRef Double	double
String	String	BSTR	WideString	ByVal String	wchar_t*
String*	ref String	PBSTR	PWideString	ByRef String	wchar_t*
DateTime	DateTime	DATE	DateTime	ByVal Date	time_t
DateTime*	ref DateTime	DATE*	PDateTime	ByRef Date	time_t*

The API is object based, with all objects having a unique identifier. These IDs are of type long and a value of -1 is designated as the NULL ID. The IDs are only unique to the specific object type; as such to identify a specific object the type and ID are required.

7.3. Object based architecture

The EZserver software was developed from the ground up using an object based architecture. All object types have three keys, they are as follows.

- ID – a private unique 32 bit integer that is allocated to the object when it is created. This key cannot be changed during the life of the object. All relationships between objects are based on this key, and is hidden from the user.

- Number – a public 32 bit integer that is used by the user to identify the object, such as Pump number.

- Name – a public 20 character string this is used by the user to identify the object such as grade name.

All objects types have the following APIs to create, find, update and delete the objects, where ???? is the object type.

- Get????sCount – To retrieve the number of objects in the internal EZServer list.

- Get????ByOrdinal – To get the ID of an object in a specific position in the internal EZServer list.

- Get????ByNumber – To get the private ID of an object from is public number.

- Get????ByName – To get the private ID of an object from is public name.

- Get????Properties – To retrieve all of the properties of the object for a given ID, this includes the number and name.

- Set????Properties – To update the properties of an object for a given ID, if this ID does not exist, a new object of this type is created and appended to the internal EZserver list and its properties assigned.

- Delete???? – To delete an object with a given ID.

Some objects types such as *Deliveries*, *Card Reads*, *Logged events* are created dynamically by the EZServer, the ID for these object is generated automatically.

8. API Definition

8.1. Connection

The connection APIs are provided to facilitate the log on, log off from the server and verification of the connection.

8.1.1. ClientLogon(Ex)

Availability – EZClient.DLL, EZClient.SO.1 and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ClientID	Int32	All	A unique identifier to identify this client, this same value must be used by the EZClient.DLL in future calls to identify the source. This value must be between 1 and 99.
ClientType	Int16	All	Any combination of the following 0x01 = Calls client, i.e. can call all non-DB API functions 0x02 = Events client, i.e. will receive all non-DB events 0x04 = DB client, i.e. can call all DB API functions and will receive all DB related events, this identifies the client as the client who is performing all the DB operations on behalf of EZServer, see *Appendix 15 – Client type*
ServerName	String	All	The IP address or network name of the EZServer service host machine. This can be passed as '<local>' it is the on the same machine as the client.
CallPortNo	Int16	All	The IP Port number to use for calling the server, the default value for this is 5123. This value can be changed in the EZServer.ini file if required.
EventsPortNo	Int16	Ex	The IP Port number to use for receiving events from the, the default value for this is 5124. This value can be changed in the EZServer.ini file if required.
CallTimeout	Int32	Ex	The timeout in ms for the EZServer service to respond to calls. The default value for this is 10000.
EventHandle	Int32	All	A HANDLE to an event object that will be cleared when events have been received. Call ProcessEvents to retrieve the events for processing. Pass NULL or zero to disable this feature. (EZClient.DLL parameter only)
hWnd	Int32	All	A windows handle that will be posted with the wMsg message when an event is received. Call ProcessEvents to retrieve the events for processing. Pass NULL or zero to disable this feature. (EZClient.DLL parameter only)
wMsg	Int32	All	The message type that is posted to the hWnd handle when events are received, it must be a value greater than 1024, this value is ignored if hWnd is passed as NULL.

Return value

Error code	Error description
OK_RESULT	The call was successful.
ALREADY_LOGGED_ON_RESULT	A client with this client ID is already logged on.
SERVER_TIMEOUT	The call to the server timed out, although a calls socket was opened successfully.
CONNECTION_BROKEN	A connection could not be established with the server, it is most likely not running or inaccessible.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This call is required to logon to the forecourt server (EZServer) before calling any other API function. The last three parameters are provided as means for the DLL to signal the host application that an event has occurred, either an event object handle(EventHandle) or window handle (hWnd) and message type (wMsg) are passed, never both. Calling ProcessEvents after receiving notification of the events will process the waiting events. None of these three parameters are required for the EZClient.SO.1 or EZTech.EZClient APIs as all of the event processing is handled internally. For ClientLogon the server address and port numbers are retrieved from the ezclient.ini file, for EZClientLogonEx these values are passed directly.

See also

Erro! Fonte de referência não encontrada.

8.1.2. DllVersion

Availability – EZClient.DLL EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Version	String*		The returned value.

Return value

Error code	Error description
OK_RESULT	The call was successful.

Remarks

This API is called to get the software version of the EZClient.DLL.

See also

ServerVersion

8.1.3. ServerVersion

Availability – EZClient.DLL, EZClient.SO.1, Web Service, EZTech.EZClient

Parameters

Parameter	Type	API	Description
Version	String*		The returned value.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is called to get the software version of the EZServer service.

See also

DllVersion

8.1.4. ClientLogoff

Availability – EZClient.DLL, EZClient.SO.1, EZTech.EZClient

Parameters

Parameter	Type	API	Description
N/A			

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is called to terminate a client session with the forecourt server (EZServer).

See also

ClientLogon(Ex)

8.1.5. ClientStatus

Availability – EZClient.DLL, EZClient.SO.1, EZTech.EZClient

Parameters

Parameter	Type	API	Description
PumpsReserved	Int16*		The number of pumps currently reserved for a prepay or preauth operation by this client.
DeliveriesLocked	Int16*		The number of deliveries locked by this client.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This call should be made just after logging on, or when the terminal enters its idle state (between transactions), to ensure that no previous operations have left dependencies in the forecourt server, for this client. If either of these values is returned as non-zero, it is up to the client application to locate the pump and/or delivery objects with reserves and/or locks, and take the appropriate action. As this call has no effect on the server it can also be used to verify the server connection.

See also

TestConnection, LicenseStatus

8.1.6. TestConnection

Availability – EZClient.DLL, EZClient.SO.1, EZTech.EZClient

Parameters

Parameter	Type	API	Description
N/A			

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

Call this function to determine the status of the connection with the server.

See also

ClientStatus, LicenseStatus, GetLicenseType

8.1.7. LicenseStatus

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
N/A			

Return value

Error code	Error description
OK_RESULT	The license key is valid.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
SERVER_NOT_LICENSED_RESULT	The license key for the server is invalid or absent.
NO_EZMOD_RESULT	The EZModule for this license key cannot be found, it is most likely turned off or not plugged in.
LICENSE_EXPIRED_RESULT	The license key has expired.

Remarks

Call this function to determine the status of the license key for the server.

See also

ClientStatus, TestConnection, GetLicenseType

8.1.8. GetLicenseType

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
LicenseType	Int16*		16 bit flags determining which modules are licensed.

Return value

Error code	Error description
OK_RESULT	The license key is valid.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

Call this function to determine the status of the license key for the server.

See also

ClientStatus, TestConnection, LicenseStatus

8.1.9. GetIniValue

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Section	String		The section in the EZServer.ini file where the key is located.
Key	String		The key for the value being returned.
Value	String*		The returned value.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
INVALID_OBJECT_PARAMETER_RESULT	The key or section names are invalid (too short).

Remarks

This call is provided so that clients can read the EZServer.ini configuration values currently in use by EZServer.

See also

SetIniValue

8.1.10. SetIniValue

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Section	String		The section in the EZServer.ini file where the key/value is to be added / updated.
Key	String		The key for the value being updated/added.
Value	String		The value to be updated/added in this section and key.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
INVALID_OBJECT_PARAMETER_RESULT	The key or section names are invalid (too short).

Remarks

This call is provided so that clients can update the EZServer.ini configuration file. Once the update is completed the EZServer.ini values are re-loaded by the EZServer, hence alterations have an immediate effect. Note that there is no validation of the section, key or values passed other than length. Passing non-existent section or key values will result in the section/key being added. Care must be taken to ensure that the configuration value does not have undesired effects on the server operation.

See also

GetIniValue

8.1.11. GetClientsCount

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Count	Int32*		Returns the current number of clients logged onto EZServer.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

See also

ClientStatus, TestConnection

8.1.12. SetDateTime

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
DataTme	DateTime		The new date and time to be saved into the EZForecourt.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is used to update the date and time of the forecourt controller.

See also

GetDateTime

8.1.13. GetDateTime

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
DataTme	DateTime*		Returns the date and time from the EZForecourt.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

The API is used to read the date and time of the forecourt controller.

See also

SetDateTime

8.1.14. ResultString

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Res	Int32		The return value from any of the API calls listed above.

Return value

An English string describing the numeric error passed.

Remarks

This API can be used to convert an error value to an error description string in English.

See also

Appendix 13 – Error messages

8.1.15. CheckSocketClosed

Availability – EZClient.DLL, EZClient.SO.1, EZTech.EZClient, Web Service

Parameters

Parameter	Type	API	Description
Param	LPARAM	All	

Return value

Error code	Error description
OK_RESULT	The socket is open.
CONNECTION_BROKEN	The socket is closed.

Remarks

This API is used to verify whether the call socket for the current socket is closed or not.

See also

TestConnection

8.2. Events

The event APIs permit the processing of events, for a client application to receive events it must be logged on as an EVENTS_CLIENT_TYPE. Events are received via a separate TCP/IP socket, if a client is logged on as an EVENTS_CLIENT_TYPE and does not process the events, this may impact the forecourt controller. See ClientLogon(Ex).

8.2.1. ProcessEvents

Availability – EZClient.DLL, EZClient.SO.1

Parameters

Parameter	Type	API	Description
N/A			

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
CONNECTION_BROKEN	The connection with the server was lost.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call. It must include 'Events Client' and/or 'DB Client' before it will receive and events.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

ProcessEvents should be called when the EZClient.DLL posts a message to the windows handle, or when the associated event object is cleared. If neither a windows handle nor an event object handle were passed to the ClientLogon, then this routine can be polled, however this is not recommended as it is inefficient.

The ProcessEvents call processes all of the waiting events and places them in an internal event queue. The contents of this queue can be examined and manipulated by using the following calls GetEventsCount, GetNextEventType, DiscardNextEvent, and GetNext?????Event, where ???? is any of the various event types.

This call is only available on the EZClient.DLL because the EZClient.SO.1 and EZTech.EZClient controls do all of the event processing internally, firing the various events as the appropriate ActiveX or .NET events.

See also

GetEventsCount, GetNextEventType, DiscardNextEvent, GetNextPumpEvent(Ex, Ex2, Ex3)

8.2.2. GetEventsCount

Availability – EZClient.DLL, EZClient.SO.1

Parameters

Parameter	Type	API	Description
Count	Int32*		The number of events waiting in the events queue.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.

Remarks

This call is used to determine how many events are waiting in the internal events queue.

See also

ProcessEvents, GetNextEventType, DiscardNextEvent

8.2.3. GetNextEventType

Availability – EZClient.DLL, EZClient.SO.1

Parameters

Parameter	Type	API	Description
Type	Int16*		The type of the event waiting at the head of the internal events queue. See *Appendix 14 – Client event types* for more information.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.

Remarks

This call is used to determine what the type of the event at the head of the internal events queue is. Once you have the type you can then determine which of the GetNext????Event calls to use.

See also

ProcessEvents, *Appendix 14 – Client event types*

8.2.4. DiscardNextEvent

Availability – EZClient.DLL, EZClient.SO.1

Parameters

Parameter	Type	API	Description
N/A			

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
OBJECT_DOES_NOT_EXIST_RESULT	The internal events queue is empty.

Remarks

This call removes the event from the head of the internal events queue; it is provided so that those events of types that the host application is not interested in can be discarded.

See also

ProcessEvents, GetEventsCount, GetNextEventType

8.2.5. GetNextPumpEvent(Ex, Ex2, Ex3) / StatusEvent

Availability – EZClient.DLL and EZClient.SO.1 for GetNextPumpEvent, EZTech.EZClient for PumpStatusEvent with no return value, EZTech.EZPump for StatusEvent with no return value, these values are also accessible via individual control properties.

Parameters

Parameter	Type	API	Description
PumpID	Int32*	All	The pump object identifier. This parameter is not present for the StatusEvent on the EZTech.EZPump controls.
PumpNumber	Int16* Int32*	All Ex3,Ex3	The logical number of the pump. This parameter is not present for the StatusEvent on the EZTech.EZPump controls.
State	Int16*	All	The current state of the pump, see *Appendix 1 – Pump states* for explanation. EZTech.EZPump control property PumpState().
ReservedFor	Int16*	All	The reserve state of the pump, see *Appendix 2 – Pump reserves* for explanation. EZTech.EZPump control property ReservedFor().
ReservedBy	Int32*	All	The ID of the client that has the reserve in place or -1 if there is no reserve. EZTech.EZPump control property ReservedBy().
HoseID	Int32*	All	The ID of the currently pulled hose object or -1 if there is no currently pulled hose. EZTech.EZPump control property CurHoseID().
HoseNumber	Int16* Int32*	Ex Ex2,Ex3	The logical number of the currently pulled hose or 0 if there is no currently pulled hose. EZTech.EZPump control property CurHoseNumber().
HosePhysicalNumber	Int32*	Ex2,Ex3	The PhysicalHoseNumber of the currently pulled hose or -1 if there is no currently pulled hose.

GradeID	Int32*	All	The ID of the grade for the currently pulled hose or -1 if there is no currently pulled hose. EZTech.EZPump control property CurGradeID().
GradeNumber	Int32*	Ex3	The grade number for the currently pulled hose or empty string if there is no currently pulled hose.
GradeName	String*	All	The grade name for the currently pulled hose or empty string if there is no currently pulled hose. EZTech.EZPump control property CurGradeName().
ShortGradeName	String*	All	The short grade name for the currently pulled hose or empty string if there is no currently pulled hose. EZTech.EZPump control property CurShortGardeName().
PriceLevel	Int16*	All	The current price level on the pump. EZTech.EZPump control property CurPriceLevel().
Price	Double*	All	The selected price for the currently pulled hose or zero if there is no currently pulled hose. EZTech.EZPump control property CurPrice().
Volume	Double*	All	The in progress volume total of the currently delivering pump, if it is not currently delivering then it is zero. EZTech.EZPump control property CurVolume().
Value	Double*	All	The in progress value total of the currently delivering pump, if it is not currently delivering then it is zero. EZTech.EZPump control property CurValue().
StackSize	Int16*	All	The number of stacked deliveries waiting to be cleared on this fueling position. EZTech.EZPump control property StackSize().
PumpName	String*	Ex,Ex2,Ex3	The fueling position name.
PhysicalNumber	Int32*	Ex,Ex2,Ex3	The fueling position number.
Side	Int16*	Ex,Ex2,Ex3	The fueling position side.
Address	Int16*	Ex,Ex2,Ex3	The fueling position address.

PriceLevel1	Int16*	Ex,Ex2,Ex3	The grade price level associated with price level 1 on this fueling position
PriceLevel2	Int16*	Ex,Ex2,Ex3	The grade price level associated with price level 2 on this fueling position
Type	Int16*	Ex,Ex2,Ex3	The pump type
PortID	Int32*	Ex,Ex2,Ex3	The port that this pump is connected to 1 USB 1
AuthMode	Int16*	Ex,Ex2,Ex3	The current authorization mode for this fueling position
StackMode	Int16*	Ex,Ex2,Ex3	The current stack mode for this fueling position
PrepayAllowed	Int16*	Ex,Ex2,Ex3	A flag as to whether prepay deliveries are permitted at this fueling point.
PreauthAllowed	Int16*	Ex,Ex2,Ex3	A flag as to whether pre authorize deliveries are permitted at this fueling point.
PriceFormat	Int16*	Ex,Ex2,Ex3	The format of the price display, see *Appendix 6 – Pump display formats*
ValueFormat	Int16*	Ex,Ex2,Ex3	The format of the value/total display, see *Appendix 6 – Pump display formats*
VolumeFormat	Int16*	Ex,Ex2,Ex3	The format of the volume/total display, see *Appendix 6 – Pump display formats*
Tag	Int64*	Ex2,Ex3	An external tag associated with this delivery in progress, or -1 if none
AttendantID	Int32*	Ex2,Ex3	The ID of the pump attendant associated with this delivery in progress, or -1 if none.
AttendantNumber	Int32*	Ex2,Ex3	The number of the pump attendant associated with this delivery in progress, or 0 if none.
AttendantName	String*	Ex2,Ex3	The name of the pump attendant associated with this delivery in progress.
AttendantTag	Int64*	Ex2,Ex3	The tag of the pump attendant associated with this delivery in progress, or -1 if none.
CardClientID	String*	Ex2,Ex3	The ID of the card client associated with this delivery in progress, or -1 if none.
CardClientNumber	Int32*	Ex2,Ex3	The number of the card client associated with this delivery in progress, or 0 if none.
CardClientName	String*	Ex2,Ex3	The name of the card client associated with this delivery in progress.
CardClientTag	Int64*	Ex2,Ex3	The tag of the card client associated with this delivery in progress, or -1 if none.
CurFlowRate	Double*	Ex3	The current flow rate in liters per minute if the fueling point is delivering.
PeakFlowRate	Double*	Ex3	The peak flow rate in liters per minute of the fueling point during the current delivery.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit

	this call.
OBJECT_DOES_NOT_EXIST_RESULT	Either the internal events queue is empty, or the event at the head of the queue is not of this type.

Remarks

This event is fired whenever the state of the pump changes, this could either be the pump state, delivery state, delivery values, reserved state, and attendant or client details.

See also

ProcessEvents, GetEventsCount, GetNextEventType, Pumps

8.2.6. GetNextDeliveryEvent(Ex, Ex2, Ex3, Ex4) / DeliveryEvent

Availability – EZClient.DLL and EZClient.SO.1 for GetNextDeliveryEvent, EZTech.EZClient for DeliveryEvent with no return value, EZTech.EZPump for DeliveryEvent with return value,

Parameters

Parameter	Type	API	Description
DeliveryID	Int32*	All	The delivery object identifier, this is allocated by the server when the delivery is completed.
PumpID	Int32*	All	The pumps object identifier of the pump which dispensed this delivery. This parameter is not present for the StatusEvent on the EZTech.EZPump controls.
PumpNumber	Int16* Int32*	All Ex3,Ex4	The logical pump number of the pump which dispensed this delivery. This parameter is not present for the StatusEvent on the EZTech.EZPump controls.
PumpName	String*	Ex3,Ex4	The name of the fueling position for this delivery.
HoseID	Int32*	All	The hose object identifier for the hose which dispensed this delivery.
HoseNumber	Int16* Int32*	Base,Ex Ex3,Ex4	The hose number of the hose which dispensed this delivery.
HosePhysicalNumber	Int32*	Ex3,Ex4	The physical number of the hose for this delivery.
TankID	Int32*	Ex3,Ex4	The ID of the tank for this delivery.
TankNumber	Int32*	Ex3,Ex4	The number of the tank for this delivery.
TankName	String*	Ex3,Ex4	The name of the tank for this delivery.
GradeID	Int32*	All	The grades object identifier for the grade of this delivery.
GradeNumber	Int32*	All	The number of the grade for this delivery.
GradeName	Int32*	All	The full name of the grade for this delivery.
GradeShortName	String*	Ex3,Ex4	The short name of the grade for this delivery.
GradeCode	String*	Ex3,Ex4	The grade code for this delivery.
PriceLevel	Int16*	All	The price level used for this delivery.
Price	Double*	All	The price used for this delivery.
Volume	Double*	All	The volume dispensed for this delivery.
Value	Double*	All	The total value for this delivery.
DeliveryState	Int16*	All	The current state of this delivery, see *Appendix 4 – Delivery states* for explanation.

DeliveryType	Int16*	All	The type of the delivery, see *Appendix 3 – Delivery types* for explanation
LockedBy	Int32*	All	The ID of the client who has locked this delivery, if it is NULL ID then the delivery is unlocked.
ReservedBy	Int32*	All	The ID of the client who has reserved this delivery, if it is NULL ID then this delivery was not reserved.
Age	Int32*	All	The age of the delivery in seconds since it was completed.
CompletedDT	DateTime*	All	The date and time the delivery completed.
AttendantID	Int32*	All	The ID of the pump Attendant that authorized this delivery or was logged onto the pump at the time the delivery was done. If neither of these was the case then this is NULL ID.
VolumeETot	Double*	Ex	The electronic volume total for this hose at the end of this delivery.
Volume2ETot	Double*	Ex	Always returned as zero. Reserved for future use.
ValueETot	Double*	Ex	The electronic value total for this hose at the end of this delivery.
OldVolumeETot	Double*	Ex2,Ex3,Ex4	The electronic volume total for this hose at the start of this delivery.
OldVolume2ETot	Double*	Ex2,Ex3,Ex4	Always returned as zero. Reserved for future use.
OldValueETot	Double*	Ex2,Ex3,Ex4	The electronic value total for this hose at the start of this delivery
NewVolumeETot	Double*	Ex2,Ex3,Ex4	The electronic volume total for this hose at the end of this delivery.
NewVolume2ETot	Double*	Ex2,Ex3,Ex4	Always returned as zero. Reserved for future use.
NewValueETot	Double*	Ex2,Ex3,Ex4	The electronic value total for this hose at the end of this delivery
Duration	Int32*	Ex,Ex2,Ex3, Ex4	The duration of the delivery in seconds.
Tag	Int64*	Ex,Ex2,Ex3, Ex4	External authorization tag associated with this delivery.
AttendentNumber	Int32*	Ex3,Ex4	The number of the pump attendant associated with this delivery or 0
AttendentName	String*	Ex3,Ex4	The name of the pump attendant associated with this delivery
AttendentTag	Int64*	Ex3,Ex4	The tag of the pump attendant associated with this delivery or -1
CardClientID	Int32*	Ex3,Ex4	The ID of the card client that authorized this delivery. If this was not the case then this is NULL ID.
CardClientNumber	Int32*	Ex3,Ex4	The number of the card client associated with this delivery, or zero.
CardClientName	String*	Ex3,Ex4	The name of the card client associated with this delivery.
CardClientTag	Int64*	Ex3,Ex4	The card client tag associated with this

			delivery, or -1.
PeakFlowRate	Double*	Ex4	The peak flow rate obtained during the delivery in liters per minute.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	Either the internal events queue is empty, or the event at the head of the queue is not of this type.

Remarks

This event is fired whenever the state of a delivery changes, that could be the delivery state, type, locked by, reserved by properties, or when the delivery is completed.

See also

ProcessEvents, GetEventsCount, GetNextEventType, Deliver

8.2.7. GetNextServerEvent / ServerEvent

Availability – EZClient.DLL and EZClient.SO.1 as GetNextServerEvent,
EZTech.EZClient as ServerEvent (no return value)
For Events clients only

Parameters

Parameter	Type	API	Description
EventID	Int32*		The ID of the server generated event.
EventText	String*		The event description string.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	Either the internal events queue is empty, or the event at the head of the queue is not of this type.

Remarks

This event is generated by the server, it can be ignored or simply displayed on the screen.

See also

ProcessEvents, GetEventsCount, GetNextEventType

8.2.8. GetNextClientEvent/ClientEvent

Availability – EZClient.DLL and EZClient.SO.1 for GetNextClientEvent,
EZTech.EZClient for ClientEvent with no return value,
For Events/DB clients only

Parameters

Parameter	Type	API	Description
ClientID	Int32*		The ID of the client which generated the event.
EventID	Int32*		The ID of the client generated event.
EventText	String*		The event description string.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	Either the internal events queue is empty, or the event at the head of the queue is not of this type.

Remarks

This event is generated when another EZServer client calls FireClientEvent. The value of the EventID and EventText are simply passed through.

See also

ProcessEvents, GetEventsCount, GetNextEventType, FireClientEvent

8.2.9. FireClientEvent

Availability – EZClient.DLL, EZTech.EZClient, EZClient.SO.1

Parameters

Parameter	Type	API	Description
EventID	Int32		The ID of the client generated event
EventText	String		The event description string.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

Calling this API will result in all other EZServer clients receiving a ClientEvent. This API is provided so that a client can send notifications to other clients.

See also

GetNextClientEvent/ClientEvent

8.2.10. GetNextDBLogEvent/DBLogEvent

Availability – EZClient.DLL and EZClient.SO.1 for GetNextDBLogEvent,
EZTech.EZClient for DBLogEvent with no return value,
For DB clients only.

Parameters

Parameter	Type	API	Description
EventType	Int32*		This is the type of the event being generated. See *Appendix 5 – Event types*, for more information
DeviceID	Int32*		The ID of the device that generated this event, The event type will depend on the event type.
EventDT	DataTime*		The time and date the event occurred
EventText	String*		The event description text.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	Either the internal events queue is empty, or the event at the head of the queue is not of this type.

Remarks

This event is fired every time the server or an attached device/module generates an exception event. As this is a DB client event, it is expected that this client will log the event to the database.

See also

ProcessEvents, GetEventsCount, GetNextEventType

8.2.11. GetNextDBLogDeliveryEvent / DBLogDeliveryEvent

Availability – EZClient.DLL and EZClient.SO.1 for GetNextDBLogDeliveryEvent,
EZTech.EZClient for DBLogDeliveryEvent with no return value.
For DB clients only.

Parameters

Parameter	Type	API	Description
DeliveryID	Int32*		The ID of the delivery, as generated by the server.
HoseID	Int32*		The ID of the hose which dispensed the delivery.
DeliveryState	Int16*		The state of the delivery, see *Appendix 4 – Delivery states* for further explanation.
DeliveryType	Int16*		The type of the delivery, see *Appendix 3 – Delivery types* for further explanation.
Volume	Double*		The volume of the delivery dispensed.
PriceLevel	Int16*		The price level the delivery was done at.
Price	Double*		The price the delivery was done at.
Value	Double*		The total value of the delivery.
Volume2	Double*		The second volume of the delivery, this is a reserved field at this stage and is always zero.
ReservedBy	Int16*		The ID of the client who reserved this delivery.
AttendantID	Int32*		The ID of the attendant who authorized this delivery or NULL ID if not authorized by an attendant.
DeliveryDT	DateTime*		The date and time this delivery was stacked.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	Either the internal events queue is empty, or the event at the head of the queue is not of this type.

Remarks

This is event is fired every time a delivery is completed, as this is a DB client event, it is expected that the client will log this delivery to the database.

See also

ProcessEvents, GetEventsCount, GetNextEventType, Deliver

8.2.12. GetNextDBClearDeliveryEvent / DBClearDeliveryEvent

Availability – EZClient.DLL and EZClient.SO.1 for GetNextDBClearDeliveryEvent, EZTech.EZClient for DBClearDeliveryEvent with no return value.
For DB clients only.

Parameters

Parameter	Type	API	Description
DeliveryID	Int32*		The ID of the delivery that is being cleared
DeliveryType	Int16*		The type of the delivery when it was cleared, see *Appendix 3 – Delivery types* for further information.
ClearedBY	Int36*		The ID of the client who cleared the delivery.
ClearedDT	DateTime*		The time and date the delivery was cleared.
AttendantID	Int32*		The ID of the attendant who cleared the delivery, or NULL ID if it was not cleared by an attendant.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	Either the internal events queue is empty, or the event at the head of the queue is not of this type.

Remarks

This event is fired every time a delivery is cleared from the system. As this event is a DB client only event, the client is expected to flag the delivery as cleared in the database etc.

See also

ProcessEvents, GetEventsCount, GetNextEventType, Deliver

8.2.13. GetNextDBStackDeliveryEvent / DBStackDeliveryEvent

Availability – EZClient.DLL and EZClient.SO.1 for GetNextDBStackDeliveryEvent, EZTech.EZClient for DBStackDeliveryEvent with no return value.
For DB clients only.

Parameters

Parameter	Type	API	Description
DeliveryID	Int36*		The ID of the delivery that was stacked

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	Either the internal events queue is empty, or the event at the head of the queue is not of this type.

Remarks

This event is fired when the state of a delivery changes from CURRENT to STACKED. As this is a DB client only event, it is expected that the DB client will update the state of the delivery in the database.

See also

ProcessEvents, GetEventsCount, GetNextEventType, Deliver

8.2.14. GetNextDBHoseETotalsEvent(Ex) / DBHoseETotalsEvent(Ex)

Availability – EZClient.DLL and EZClient.SO.1 for GetNextDBStackDeliveryEvent,
EZTech.EZClient for DBStackDeliveryEvent with no return value.
For DB clients only.

Parameters

Parameter	Type	API	Description
HoseID	Int32*	All	The ID of the hose for which the electronics are being updated
Volume	Double*	All	The volume of the delivery which resulted in the change to the electronic totals, this value will be zero if the pump has just started responding.
Value	Double*	All	The value of the delivery which resulted in the change to the electronic totals, this value will be zero if the pump has just started responding.
VolumeETot	Double*	All	The new electronic volume total as retuned by the pump for this hose.
ValueETot	Double*	All	The new electronic value total as retuned by the pump for this hose.
HoseNumber	Int32*	Ex	The hose number (position on the felling point)
HosePhyscalNumber	Int32*	Ex	The hose physical number (global hose number)
PumpID	Int32*	Ex	The ID of the fueling position that this hose belongs to.
PumpNumber	Int32*	Ex	The number of the fueling position that this hose belongs to.
PumpName	String*	Ex	The name of the fueling position that this hose belongs to.
TankID	Int32*	Ex	The ID of the tank that this hose is connected to.
TankNumber	Int32*	Ex	The number of the tank that this hose is connected to.
TankName	String*	Ex	The name of the tank that this hose is connected to.
GradeID	Int32*	Ex	The ID of the grade that this hose dispenses.
GradeName	String*	Ex	The name of the grade that this hose dispenses.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

OBJECT_DOES_NOT_EXIST_RESULT	Either the internal events queue is empty, or the event at the head of the queue is not of this type.

Remarks

This event is fired at the completion of a delivery on the respective hose or when the pump starts responding. If the electronic totals are as a result of delivery completion, then the volume and value of the delivery is also passed, otherwise these values are zero.

See also

ProcessEvents, GetEventsCount, GetNextEventType, Deliver, Hose

8.2.15. GetNextDBTriggerEvent/DBTriggerEvent

Availability – EZClient.DLL and EZClient.SO.1 for GetNextDBTriggerEvent,
EZTech.EZClient for DBTriggerEvent with no return value.
For DB clients only.

Parameters

Parameter	Type	API	Description
TableID	Int32*		The ID of the table that has been updated
RowID	Int32*		The object ID of the table row that has been updated

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	Either the internal events queue is empty, or the event at the head of the queue is not of this type.

Remarks

This event is fired by the EZTrigger DLL which is called from various insert/update/delete triggers on the EZDB database. This is done so that the database client is aware of changes being made to the database by other applications, typically configuration/maintenance applications. For more information on this see the section regarding the EZTrigger DLL.

See also

ProcessEvents, GetEventsCount, GetNextEventType

8.2.16. GetNextDBAttendantLogonEvent / DBAttendantLogonEvent

Availability – EZClient.DLL and EZClient.SO.1 for GetDBNextAttendantLogon,
EZTech.EZClient for DBAttendantLogon with no return value.
For DB clients only.

Parameters

Parameter	Type	API	Description
AttendantID	Int32*		The ID of the attendant that has successfully logged on.
PumpID	Int32*		The ID of the pump that was logged onto.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	Either the internal events queue is empty, or the event at the head of the queue is not of this type.

Remarks

This event is fired by when an attendant successfully logs onto a specific pump.

See also

ProcessEvents, GetEventsCount, GetNextEventType, AttendantLogon, Attendants

8.2.17. GetNextDBAttendantLogoffEvent / DBAttendantLogonEvent

Availability – EZClient.DLL and EZClient.SO.1 for GetDBNextAttendantLogoff,
EZTech.EZClient for DBAttendantLogoff with no return value,
For DB clients only.

Parameters

Parameter	Type	API	Description
AttendantID	Int32*		The ID of the attendant that has successfully logged off.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	Either the internal events queue is empty, or the event at the head of the queue is not of this type.

Remarks

This event is fired when an attendant successfully logs off one or more pumps.

See also

ProcessEvents, GetEventsCount, GetNextEventType, AttendantLogoff, Attendants

8.2.18. GetNextDBTankStatusEvent(Ex,Ex2) / DBTankStatusEvent(Ex,Ex2)

Availability – EZClient.DLL and EZClient.SO.1 for GetNextDBTankStatusEvent,
EZTech.EZClient for DBTankStatusEvent with no return value.
For DB clients only.

Parameters

Parameter	Type	API	Description
TankID	Int32*	All	The ID of the tank that generated the status event.
GaugeVolume	Double*	All	The volume of fuel in the tank as reported by the tank gauge.
GaugeTCVolume	Double*	All	The temperature corrected volume of fuel in the tank as reported by the tank gauge.
GaugeUllage	Double*	All	The unused or available capacity in the tank as reported by the tank gauge.
GaugeTemperature	Double*	All	The temperature of the fuel in the tank as reported by the tank gauge.
GaugeLevel	Double*	All	The level of the fuel in the tank as reported by the tank gauge.
GaugeWaterVolume	Double*	All	The volume of water in the bottom of the tank as reported by the tank gauge.
GaugeWaterLevel	Double*	All	The level of the water in the bottom of the tank as reported by the tank gauge.
TankNumber	Int32*	Ex,Ex2	The tank number.
TankName	String*	Ex,Ex2	The tank name.
GradeID	Int32*	Ex,Ex2	The ID of the grade in this tank.
GradeName	String*	Ex,Ex2	The name of the grade in this tank.
Type	Int16*	Ex,Ex2	The type of tank measurement, see *Appendix 12 – Tank Types*
Capacity	Double*	Ex,Ex2	The total capacity of the tank.
Diameter	Double*	Ex,Ex2	The diameter of the tank.
GaugeID	Int32*	Ex,Ex2	The ID of the ATG that is reading this tank.
ProbeNo	Int16*	Ex,Ex2	The number of the probe inserted into this tank.
State	Ex2	Ex2	The state of the tank see *Appendix 23 – Tank State*
AlarmsMask	Int32*	Ex2	The state of all the alarms for this tank, in a bit mask format see *Appendix 24 – Alarms Mask*.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.

INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	Either the internal events queue is empty, or the event at the head of the queue is not of this type.

Remarks

This event is fired when an attendant successfully logs off one or more pumps.

See also

ProcessEvents, GetEventsCount, GetNextEventType, Tanks

8.2.19. GetNextCardReadEvent / CardReadEvent

Availability – EZClient.DLL and EZClient.SO.1 for GetNextCardReadEvent,
EZTech.EZClient for CardReadEvent with no return value,
For Events/DB clients only

Parameters

Parameter	Type	API	Description
CardReadID	Int32*		The ID of the card read, this is generated by the EZserver when the card read is detected.
Number	Int32*		The number for this card read, this will be the pump attendant or card client number depending on the type of card passed.
Name	String*		The name for this card read, this will be the pump attendant or card client name depending on the type of card passed.
PumpID	Int32*		The ID of the fueling position where this card read was registered.
Type	Int16*		The type of the card read see *Appendix 26 – Card Read Types*.
ParentID	Int32*		The ID of the pump attendant or card client or NULL ID, depending on the card read type.
Tag	Int64*		The hexadecimal RFiD tag ID as printed on the reverse of the card.
TimeStamp	DateTime*		The time and date that the card read was read at the reader.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This event is trigged whenever an RFiD card is read at one of the EZiD readers mounted on a pump.

See also

ProcessEvents, GetEventsCount, GetNextEventType, Card Read

8.2.20. GetNextLogEventEvent/LogEventEvent

Availability – EZClient.DLL and EZClient.SO.1 for GetNextLogEventEvent,
EZTech.EZClient for LogEventEvent with no return value,
For Events/DB clients only

Parameters

Parameter	Type	API	Description
LogEventID	Int32*		This ID of the Log Event, this is generated by the EZServer when the event is created.
DeviceType	Int16*		The type of the device that generated the event see *Appendix 20 – Log Event Device Type*
DeviceID	Int32*		The ID of the device that generated the event
DeviceNumber	Int32*		The number of the device that generated the event
DeviceName	String*		The name of the device that generated the event
EventLevel	Int16*		The level of the event see *Appendix 21 – Log Event Level*
EventType	Int16*		The type of the event see *Appendix 22 – Log Event Type*
EventDesc	String*		The event description.
GeneratedDT	DateTime*		The time and date the event was generated.
ClearedDT	DateTime*		The time and date the event was cleared.
ClearedBy	Int32*		The client ID that cleared the event see ClearLogEvent
AckedBy	Int32*		The client ID that acknowledged the event see AckLogEvent
Volume	Double*		A volume associated with this event, depends on the event type.
Value	Double*		A value associated with this event, depends on the event type.
ProductVolume	Double*		The volume of product in the tank if this is a tank related event.
ProductLevel	Double*		The level of product in the tank if this is a tank related event.
WaterLevel	Double*		The water level in the tank if this is a tank related event.
Temperature	Double*		The temperature in the tank if this is a tank related event.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This event is trigged whenever a log event is created, or changes.

See also

ProcessEvents, GetNextEventType, Logged

8.2.21. GetNextZeroDeliveryEvent

Availability – EZClient.DLL and EZClient.SO.1 for GetNextZeroDeliveryEvent,
EZTech.EZClient for ZeroDeliveryEvent with no return value,
For Events/DB clients only

Parameters

Parameter	Type	API	Description
PumpID	Int32*		The ID of fueling position that generated the event.
PumpNumber	Int32*		The number of fueling position that generated the event.
HoseID	Int32*		The ID of hose that generated the event.
Number	Int32*		The number of the hose that generated the event.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This event is generated if a hose was authorized and the returned without delivering, this is known as a zero delivery.

See also

ProcessEvents, GetNextEventType, Pumps, Hose

8.2.22. GetNextZB2GStatusEvent

Availability – EZClient.DLL and EZClient.SO.1 for GetNextZB2GStatusEvent,
EZTech.EZClient for ZB2GStatusEvent with no return value,
For Events/DB clients only

Parameters

Parameter	Type	API	Description
PortID	Int32*		The ID of the port that generated the event, usually 1 for USB1 port.
ZBAddress	Int64*		The 64bit MAC address of the device that generated the status event.
LQI	Int16*		The Link Quality Indicator, the higher the better.
RSSI	Int16*		The signal strength in dB, the higher the better.
ParZBAddress	Int64*		The 64 bot MAC address of the parent device, This can be used to determine how the network was formed.
ZBChannel	Int16*		The channel in use, 11 to 26
MemBlocks	Int16*		The number of memory blocks in use.
MemFree	Int16*		The number of free memory blocks.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This event is fired when the status of a specific ZigBee device changes.

See also

ProcessEvents, GetNextEventType, *ZigBee*

8.3. Pumps

The pump object is the main component for any forecourt controller, it is with this device that the majority of the communication occurs. The concept of a pump for the EZForecourt is in fact a fueling position, a physical pump may have more than one fueling position, and normally has two however it may have 4. For the rest of the document, pump will be referring to a fueling position.

8.3.1. GetPumpsCount

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Count	Int32*		The returned total number of pump objects configured in the server.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is used determine the total number of pumps currently configured in the server. Once this is known, the IDs of the individual pump objects can be obtained using GetPumpByOrdinal.

See also

GetPumpByName, GetPumpByGetPumpByNumber

8.3.2. GetPumpByName

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Name	String		The name of the pump object for which the ID is being requested.
ID	Int32*		The returned ID of the pump object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

Use this API to get the ID of the pump from the pump name.

See also

GetPumpsCount, GetPumpBy, GetPumpByNumber

8.3.3. GetPumpByNumber

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Number	Int32		The logical number of the pump for which the ID is being requested.
ID	Int32*		The returned ID of the pump object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

Use this API to get the pump ID from the logical pump number.

See also

GetPumpsCount, GetPumpByOrdinal, GetPumpByName

8.3.4. GetPumpByOrdinal

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Index	Int32		The index of the desired pump object, this can be between 1 and the total number of pumps, as returned by GetPumpsCount.
ID	Int32*		The returned ID of the pump.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used to retrieve the pump ID of a pump using its ordinal value in the EZServer's internal pump objects list. The pumps are ordered in this list by ID.

See also

GetPumpsCount, GetPumpByNumber, GetPumpByName, **Erro! Fonte de referência não encontrada.**

8.3.5. GetPumpProperties(Ex)

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient, EZTech.EZPump via individual properties.

Parameters

Parameter	Type	API	Description
ID	Int32	All	The ID of the pump object for which the properties are being requested. EZTech.EZPump property ID().
Number	Int32*	All	The logical pump number EZTech.EZPump property Number()
Name	String*	All	The pump name. EZTech.EZPump property Name()
PhysicalNumber	Int16*	All	The pump physical number. EZTech.EZPump property PhysicalNumber()
Side	Int16*	All	The side of the physical pump for this filling position. EZTech.EZPump property Side()
Address	Int16*	All	The electronic polling address, for the pump as configured in the pump. EZTech.EZPump property Address()
PriceLevel1	Int16*	All	The EZForecourt price level to be used as price level 1 on this pump. EZTech.EZPump property PriceLevel1()
PriceLevel2	Int16*	All	The EZForecourt price level to be used as price level 2 on this pump. EZTech.EZPump property PriceLevel2()
PriceDspFormat	Int16*	All	The format of the pump price display, see *Appendix 6 – Pump display formats* for more information. EZTech.EZPump property PriceDspFormat()
VolumeDspFormat	Int16*	All	The format of the pump volume display, see *Appendix 6 – Pump display formats* for more information. EZTech.EZPump property VolumeDspFormat()
ValueDspFormat	Int16*	All	The format of the pump value display, see *Appendix 6 – Pump display formats* for more information. EZTech.EZPump property ValueDspFormat()
Type	Int16*	All	The pump type identifier, this list is continually being

			added to. Consult EZTech support for the current list of supported pump types. EZTech.EZPump property Type()
PortID	Int16*	All	The ID of the communications port that this pump is connected to, the address of this pump must be unique for this port. EZTech.EZPump property PortID()
AttendantID	Int32*	All	The ID of the attendant currently logged onto this pump. EZTech.EZPump property AttendantID()
AuthMode	Int16*	All	The authorization mode of this pump. See *Appendix 7 – Pump authorization modes* for more information. EZTech.EZPump property AuthMode()
StackMode	Int16*	All	The pump delivery stack (or memory) mode for this pump, see *Appendix 8 – Pump delivery stack (memory) modes* for more information. EZTech.EZPump property StackMode()
PrepayAllowed	Int16*	All	Are prepay deliveries permitted on this pump. EZTech.EZPump property PrepayAllowed()
PreauthAllowed	Int16*	All	Are preauth deliveries permitted on this pump. EZTech.EZPump property PreauthAllowed()
SlotZigBeeId	Int16*	Ex	The slot or EZRemote ID that this pump is connected to. 1 to 4 are the slots, 5 and above are ZigBee IDs
MuxSlotBeeID	Int16*	Ex	Reserved must be NULL_ID
PriceControl	Int16*	Ex	Whether the EZserver is to control the price of this fueling point or not. See *Appendix 17 – Price Control*
HasPreset	Int16*	Ex	Dos this pump support remote presets.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API returns all the property values of a pump object for the given ID.

See also

SetPumpProperties(Ex)**Erro! Fonte de referência não encontrada.**

8.3.6. SetPumpProperties(Ex)

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient
For DB clients only.

Parameters

Parameter	Type	API	Description
ID	Int32	All	The ID for the pump object being queried.
Number	Int32	All	The logical pump number.
Name	String	All	The pump name.
PhysicalNumber	Int16	All	The pump physical number.
Side	Int16	All	The side of the physical pump for this filling position.
Address	Int16	All	The electronic polling address for the pump, as configured in the pump.
PriceLevel1	Int16	All	The EZForecourt price level to be used as price level 1 on this pump.
PriceLevel2	Int16	All	The EZForecourt price level to be used as price level 2 on this pump.
PriceDspFormat	Int16	All	The format of the pump price display, see *Appendix 6 – Pump display formats* for more information.
VolumeDspFormat	Int16	All	The format of the pump volume display, see *Appendix 6 – Pump display formats* for more information. This property corresponds to the
ValueDspFormat	Int16	All	The format of the value display see *Appendix 6 – Pump display formats* for more information.
Type	Int16	All	The pump type identifier, this list is continually being added to. Consult EZTech support for the current list of supported pump types.
PortID	Int32	All	The ID of the communications port that this pump is connected to, the address of this pump must be unique for this port.
AttendantID	Int32	All	The ID of the attendant currently logged onto this pump.
AuthMode	Int16	All	The authorization mode of this pump. See *Appendix 7 – Pump authorization modes* for more information.
StackMode	Int16	All	The pump delivery stack (or memory) mode for this pump, see *Appendix 8 – Pump delivery stack (memory) modes* for more information.
PrepayAllowed	Int16	All	Are prepay deliveries permitted on this pump.
PreauthAllowed	Int16	All	Are preauth deliveries permitted on this pump.
SlotZigBeeID	Int16	Ex	The slot or EZRemote ID that this pump is connected to. 1 to 4 are the slots, 5 and above are ZigBeeIDs
MuxSlotZigBeeID	Int32	Ex	Reserved for future use.
PriceControl	Int16	Ex	Whether the EZserver is to control the price of this fueling point or not. See *Appendix 17 – Price Control*
HasPreset	Int16	Ex	Dos this pump support remote presets.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
INVALID_OBJECT_PARAMETER_RESULT	One of the property values passed is out of range.
INVALID_OBJECT_LINK_RESULT	One of the IDs passed representing a related object references a non-existing object.

Remarks

This API is provided so that pump objects can be created and maintained on the server. If the given pump ID already exists then the properties for this object will be overwritten with the values passed, otherwise a new pump object with this ID is created and its properties set to the values passed. It is the responsibility of the caller to ensure that the pump address is unique for this port, and that the pump number etc. are unique.

See also

GetPumpProperties(Ex), DeletePump

8.3.7. DeletePump

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient.
For DB clients only.

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump object to be deleted from the EZServer.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
OBJECT_HAS_DEPENDANCIES_RESULT	This object cannot be deleted as there are other objects in the server which have references to it.

Remarks

This API is used to remove a pump object from the EZServer. If this pump has hoses or deliveries linked to it, it cannot be deleted, delete these objects first. If a pump ID of -1 is passed all the pumps will be deleted in a single operation.

See also

SetPumpProperties(Ex).

8.3.8. GetPumpHosesCount/GetHosesCount

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient for GetPumpHosesCount
EZTech.EZPump for GetHosesCount

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump in question. This parameter is not present for the StatusEvent on the EZTech.EZPump controls.
Count	Int32*		The returned number of hoses.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used to determine the quantity of hoses configured on this pump. Once this is obtained GetPumpHoseByNumber can be used get the IDs for each of the individual hoses.

See also

GetPumpHoseByNumber

8.3.9. GetPumpHoseByNumber/GetHoseByNumber

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient for GetPumpHoseByNumber. EZTech.EZPump for GetHoseByNumber.

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump that owns the hose. This parameter is not present for the GetHoseByNumber on the EZTech.EZPump controls.
Number	Int32		The logical number of the hose, ranging from 1 to the hoses count for this pump.
HoseID	Int32*		The returned ID of the hose.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used to get the hose IDs for hoses connected to a specific pump. This API along with GetPumpHosesCount can be used to get all of the hose IDs for a specific pump.

See also

GetPumpHosesCount/GetHosesCount

8.3.10. GetPumpStatus(Ex, Ex2)

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient.

Parameters

Parameter	Type	API	Description
ID	Int32	All	The ID of the pump for which the status is being requested.
State	Int16*	All	The state of the pump, see *Appendix 1 – Pump states* for more information.
ReservedFor	Int16*	All	The type of operation that the pump has been reserved for. See *Appendix 2 – Pump reserves* for more information.
ReservedBy	Int32*	All	The ID of the client who placed the reserve on the pump.
HoseID	Int32*	All	The ID of the current hose on the pump, or NULL ID if all the hoses are stowed.
HoseNumber	Int32*	All	The logical number of the current hose on the pump, or zero if all the hoses are stowed.
GradeID	Int32*	All	The ID of the grade for the current hose on the pump, or NULL ID if all the hoses are stowed.
GradeName	String*	All	The name of the grade for the current hose on the pump, or empty string if all the hoses are stowed.
ShortGradeName	String*	All	The short name of the grade for the current hose on the pump, or empty string if all the hoses are stowed.
PriceLevel	Int16*	All	The currently selected price level on the pump.
Price	Double*	All	The current price on the pump or zero if there is no hose pulled.
Volume	Double*	All	The current volume on the pump or zero if there is no hose pulled.
Value	Double*	All	The current value on the pump or zero if there is no hose pulled.
StackSize	Int16*	All	The number of untaken deliveries waiting on this pump.
AttendantID	Int32*	Ex,Ex2	The ID of the pump attendant that authorized the current delivery, or NULL ID if none is associated.
AttendantNumber	Int32*	Ex,Ex2	The number of the pump attendant.
AttendantName	Int32*	Ex,Ex2	The name of the pump attendant.
AttendantTag	Int64*	Ex,Ex2	The tag of the pump attendant.
CardClientID	Int32*	Ex,Ex2	The ID of the card client used to authorize the current delivery, or NULL ID if none is associated.
CardClientNumber	Int32*	Ex,Ex2	The number of the card client.
CardClientName	Int32*	Ex,Ex2	The name of the pump attendant.
CardClientTag	Int64*	Ex,Ex2	The tag of the pump attendant.
CurFlowRate	Double*	Ex2	The current flow rate in liters per minute.
PeakFlowRate	Double*	Ex2	The peak flow rate achieved during this delivery in liters per minute.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This call can be used to get the current status of the pump. This call returns the same information as that supplied by the PumpStatusEvent.

See also

GetNextPumpEvent(Ex, Ex2, Ex3) / StatusEvent, GetPumpDeliveryProperties(Ex, Ex2, Ex3, Ex4), GetAllPumpStatuses

8.3.11. PumpStateString

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
State	Int16		The numeric value of the pump state.

Return value

An English string describing the pump state value passed.

Remarks

This API can be used to convert a pump state value to a pump state description string in English.

See also

GetNextPumpEvent(Ex, Ex2, Ex3) / StatusEventGetPumpStatus(Ex, Ex2)

8.3.12. EnablePump

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient.

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump to enable.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used primarily to control CNG dispensers which by nature are auto authorize. Disabling and enabling these pumps does give some form of control, however the state of the hose (in or out) when the pump is disabled is not always available.

See also

DisablePump

8.3.13. DisablePump

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient.

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump to disable.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is the opposite of EnablePump, and is used primarily to block CNG dispensers.

See also

EnablePump

8.3.14. SetPumpDefaultPriceLevel

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient.

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the fueling position in question.
Level	Int16		The price level to be used as the default, this value must match one of the price levels configured for each grade.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used to set the default price level used for this fueling position. This is the price level used if none is specified in the authorization functions, or if it is auto authorization.

See also

GetPumpProperties(Ex), SetPumpProperties(Ex), LoadPreset, LoadPresetWithPrice, PaymentAuthorise

8.3.15. GetDensity

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient.

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the fueling position in question.
Density	Double*		The CNG density at this fueling point.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used to retrieve the density of the CNG at a specific CNG fueling point. If the fueling position is not for CNG this API will return 0.

See also

GetPumpStatus(Ex, Ex2).

8.3.16. ScheduleBeep

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The pump ID to which to action the beeper, or -1 for the EZForecourt.
Pitch1	Int16		The pitch of the first beep, 1500 = 1 KHz, 1000 = 1.5 KHz, 750 = 2Khz, 500 = 3Khz, 375 = 4Khz, 300 = 5 KHz, 250 = 6Khz, 188 = 8 KHz, 0 = off
Duration1	Int16		The duration the first beep in ms.
Pitch2	Int16		The pitch of the second beep.
Duration2	Int16		The duration the second beep in ms.
Pitch3	Int16		The pitch of the third beep.
Duration3	Int16		The duration the third beep in ms.
Pitch4	Int16		The pitch of the fourth beep.
Duration4	Int16		The duration the fourth beep in ms.
Pitch5	Int16		The pitch of the fifth beep.
Duration5	Int16		The duration the fifth beep in ms.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is used to action the beeper in an EZRemote installed in a pump, or in the EZForecourt. You can causa up to 5 different notes of different durations.

See also

FlashLEDS, GetDeviceDetails, ResetDevice

8.3.17. FlashLEDS

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The pump ID to where to flash the LEDs on the EZReader
Side	Int16		The side of the pump to flash the LEDs, 1 or 2
PeriodMs	Int16		The period in ms.
Cycles	Int16		The number of flashes

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is used to flash the LEDs of an EZreader connected to an EZRemote. You can causa up to flash a repeated number of times for a specific duration.

See also

ScheduleBeep, GetDeviceDetails, ResetDevice

8.4. Pump prepay deliveries

The Prepay APIs are provided to assist the POS system in performing prepay deliveries, this could result in the generation of prepay refund deliveries. See *Appendix 3 – Delivery types*.

8.4.1. PrepayReserve

Availability – EZClient.DLL, EZClient.SO.1, Web Service, EZTech.EZClient and EZTech.EZPump

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump for which the reserve is being requested. This parameter is not present for the EZTech.EZPump controls.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
INVALID_CLIENT_ID_RESULT	The current Client ID is not logged onto the server.
PUMP_NOT_RESPONDING_RESULT	The pump is not responding.
PUMP_NOT_AVAILABLE_RESULT	The pump is unavailable, it is either changing prices or timing out after a prepay refund.
PREPAYS_NOT_PREMITTED_RESULT	Prepays are not permitted on this pump.
PUMP_IN_USE_RESULT	The pump is not idle.
PUMP_IS_STOPPED_RESULT	The pump is temp stopped.
PUMP_ALREADY_RESERVED_RESULT	The pump already has either a prepay, preauth or payment reserve placed on it.
PUMP_NOT_AVAILABLE_RESULT	The pump has a current delivery which cannot be stacked automatically.

Remarks

The API is used to start a prepay delivery on a pump. This is the first step and must be done before the client is permitted to pay for the delivery. This guarantees that the pump is available and reserved for this prepay delivery.

See also

PrepayCancel, PrepayAuthorise

8.4.2. PrepayCancel

Availability – EZClient.DLL, EZClient.SO.1, Web Service, EZTech.EZClient and EZTech.EZPump

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump that is having the prepay reserve cancelled. This parameter is not present for the EZTech.EZPump controls.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
PUMP_NOT_RESERVED_FOR_PREPAY_RESULT	The pump was not reserved for a prepay delivery.
PUMP_NOT_RESERVED_BY_YOU_RESULT	The pump was not reserved by this client.
PUMP_NOT_RESPONDING_RESULT	The pump is not currently responding.
PUMP_IN_USE_RESULT	The pump is not idle.

Remarks

This API is used to cancel either a prepay reserve or prepay authorized pump. If the pump was already authorized for a prepay delivery then a prepay refund for the full amount will be generated.

See also

PrepayReserve, PrepayAuthorise

8.4.3. PrepayAuthorise

Availability – EZClient.DLL, EZClient.SO.1, Web Service, EZTech.EZClient and EZTech.EZPump

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump that is being prepay authorized. This parameter is not present for the EZTech.EZPump controls.
LimitType	Int16		The limit type being place on the delivery, DOLLAR_PREPAY_TYPE and VOLUME_PREPAY_TYPE are the only valid values here. See *Appendix 9 – Pump limit types* for more information
Value	Double		The dollar or volume limit being placed on the delivery.
Hose	Int16		The permitted hose/hoses for this prepay delivery. See *Appendix 10 – Permitted hoses mask* for more information.
PriceLevel	Int16		The selected price level for this prepay delivery.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
INVALID_PRESET_TYPE_RESULT	The preset type requested is not permitted for this type of authorize.
INVALID_PRESET_AMOUNT_RESULT	The preset amount is less than the configured minimum.
INVALID_HOSE_MASK_RESULT	None of the permitted hoses are configured on this pump.
PUMP_NOT_RESPONDING_RESULT	The pump is not responding.
PUMP_NOT_AVAILABLE_RESULT	The pump is performing a price change or a timeout after a prepay refund.
PUMP_IN_USE_RESULT	The pump is no longer idle.
PUMP_NOT_RESERVED_FOR_PREPAY_RESULT	The pump was not reserved for a prepay delivery.
PUMP_NOT_RESERVED_BY_YOU_RESULT	The pump was not reserved by this client.
PUMP_IS_STOPPED_RESULT	The pump is currently temp stopped.

Remarks

This API is the second step in doing a prepay delivery on a pump. Once the pump has been reserved correctly, the value and grade have been confirmed and paid for, the pump can be authorized to delivery up to the volume/value purchased.

Once this delivery is completed it is cleared automatically as a prepay delivery. If the value/volume was less than the limit specified then a prepay refund is generated. If the delivery is not started within the configured prepay authorized timeout, a prepay refund will be generated for the full amount.

See also

PrepayReserve, PrepayCancel

8.5. Pump preauth deliveries

The preauth APIs are provide to facilitate the integration with EFT payment terminals, these APIs are maintained for backward compatibility, and any new development should utilize the Payment APIs.

8.5.1. PreauthReserve

Availability – EZClient.DLL, EZClient.SO.1, Web Service, EZTech.EZClient and EZTech.EZPump

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump for which the reserve is being requested. This parameter is not present for the EZTech.EZPump controls.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
INVALID_CLIENT_ID_RESULT	The current Client ID is not logged onto the server.
PUMP_NOT_RESPONDING_RESULT	The pump is not responding.
PUMP_NOT_AVAILABLE_RESULT	The pump is unavailable, it is either changing prices or timing out after a prepay refund.
PREAUTHS_NOT_PREMITTED_RESULT	Preauths are not permitted on this pump.
PUMP_IN_USE_RESULT	The pump is not idle.
PUMP_IS_STOPPED_RESULT	The pump is temp stopped.
PUMP_ALREADY_RESERVED_RESULT	The pump already has either a prepay, preauth or payment reserve on it.
PUMP_NOT_AVAILABLE_RESULT	The pump has a current delivery which cannot be stacked automatically.

Remarks

The API is used to start a preauth delivery on a pump. This is the first step and must be done before the client’s credit rating is determined. This guarantees that the pump is available and reserved for this preauth delivery.

See also

PreauthCancel, PreauthAuthorise

8.5.2. PreauthCancel

Availability – EZClient.DLL, EZClient.SO.1, Web Service,
EZTech.EZClient and EZTech.EZPump

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump that is having the preauth reserve cancelled. This parameter is not present for the EZTech.EZPump controls.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
PUMP_NOT_RESERVED_FOR_PREAUTH_RESULT	The pump was not reserved for a preauth.
PUMP_NOT_RESERVED_BY_YOU_RESULT	The pump was not reserved by this client.
PUMP_NOT_RESPONDING_RESULT	The pump is not currently responding.
PUMP_IN_USE_RESULT	The pump is not idle.

Remarks

This API is used to cancel either a preauth reserved or authorized pump.

See also

PreauthReserve, PreauthAuthorise

8.5.3. PreauthAuthorise

Availability – EZClient.DLL, EZClient.SO.1, Web Service, EZTech.EZClient and EZTech.EZPump

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump that is being preauth authorized. This parameter is not present for the EZTech.EZPump controls.
LimitType	Int16		The limit type being place on the delivery, DOLLAR_PREAUTH_TYPE and VOLUME_PREAUTH_TYPE are the only valid values here. See *Appendix 9 – Pump limit types* for more information
Value	Double		The dollar or volume limit being placed on the delivery.
Hose	Int16		The permitted hose/hoses for this preauth delivery. See *Appendix 10 – Permitted hoses mask* for more information.
PriceLevel	Int16		The selected price level for this preauth delivery.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
INVALID_PRESET_TYPE_RESULT	The preset type requested is not permitted for this type of authorize.
INVALID_PRESET_AMOUNT_RESULT	The preset amount is less than the configured minimum.
INVALID_HOSE_MASK_RESULT	None of the permitted hoses are configured on this pump.
PUMP_NOT_RESPONDING_RESULT	The pump is not responding.
PUMP_NOT_AVAILABLE_RESULT	The pump is performing a price change or a timeout after a prepay refund.
PUMP_IN_USE_RESULT	The pump is no longer idle.
PUMP_NOT_RESERVED_FOR_PREAUTH_RESULT	The pump was not reserved for a preauth.
PUMP_NOT_RESERVED_BY_YOU_RESULT	The pump was not reserved by this client.
PUMP_IS_STOPPED_RESULT	The pump is currently temp stopped.

Remarks

This API is the second step in doing a preauth delivery on a pump. Once the pump has been reserved correctly, the value and grade have been confirmed and the credit rating of the client confirmed, the pump can be authorized to delivery up to the volume/value limit.

Once this delivery is completed is must be cleared by the EZServer client who reserved/authorized this delivery. If the delivery is not started within the configured preauth authorized timeout the pump reverts back to unreserved. It is then the responsibility of the EZServer client to reverse the preauth transaction.

See also

PreauthReserve, PreauthCancel

8.6. Pump payment deliveries

They pump payment APIs are provided to facilitate the integration with payment terminals. This supersedes the preauth delivery APIs.

8.6.1. PaymentReserve

Availability – EZClient.DLL, EZClient.SO.1, Web Service, EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump that is being payment authorized. This parameter is not present for the EZTech.EZPump controls.
TermID	Int32		The client ID that is making the reservation.
TermHash	String		Reserved for future use.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
INVALID_CLIENT_ID_RESULT	The current Client ID is not logged onto the server.
PUMP_NOT_RESPONDING_RESULT	The pump is not responding.
PUMP_NOT_AVAILABLE_RESULT	The pump is unavailable, it is either changing prices or timing out after a prepay refund.
PUMP_IN_USE_RESULT	The pump is not idle.
PUMP_IS_STOPPED_RESULT	The pump is temp stopped.
PUMP_ALREADY_RESERVED_RESULT	The pump already has either a prepay, preauth or payment reserve placed on it.
PUMP_NOT_AVAILABLE_RESULT	The pump has a current delivery which cannot be stacked automatically.

Remarks

This API starts the process of a payment type delivery, this the first step and is necessary to verify the availability of the fueling position in question. The TermID can be the same as the ClientID, and must be the same as the TermID used in the cancel and authorize steps of the transaction, and also the same as the client ID that locks and clears the delivery.

See also

PaymentCancel, PaymentAuthorise, ClearDelivery

8.6.2. PaymentCancel

Availability – EZClient.DLL, EZClient.SO.1, Web Service, EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump that is being payment authorized. This parameter is not present for the EZTech.EZPump controls.
TermID	Int32		The client ID that is cancelling the reservation.
TermHash	String		Reserved for future use.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
PUMP_NOT_RESERVED_FOR_CTF_RESULT	The pump was not reserved for a payment.
PUMP_NOT_RESERVED_BY_YOU_RESULT	The pump was not reserved by this client.
PUMP_NOT_RESPONDING_RESULT	The pump is not currently responding.
PUMP_IN_USE_RESULT	The pump is not idle.

Remarks

This API cancels a previous PaymentReserve.

See also

PaymentReserve, PaymentAuthorise

8.6.3. PaymentAuthorise

Availability – EZClient.DLL, EZClient.SO.1, Web Service, EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump that is being payment authorized. This parameter is not present for the EZTech.EZPump controls.
TermID	Int32		The client ID that is authorizing the payment delivery.
TermHash	String		Reserved for future use.
AttendantID	Int32		The ID of the pump attendant to be associated with this delivery, or NULL ID.
AttendantTag	Int64		The pump attendants RFiD tag to be associated with this delivery or -1. This tag must belong to a configured pump attendant.
CardClientID	Int32		The ID of the card client to be associated with this delivery, or NULL ID.
CardClientTag	Int64		The card clients RFiD tag to be associated with this delivery or -1. This tag must belong to a configured pump attendant.
AuthType	Int16		An optional authorization type to be saved with the delivery.
ExtTag	Int64		An optional external tag to be saved with the delivery.
PriceLevel	Int16		The predefined price level to be used for this delivery or zero for none.
Price	Double		The price to be used for this delivery, the treatment of this depends on the PriceType parameter.
GradeType	Int16		The grade type permitted, or 0 for all. See *EZATG INI file* and GetGradeProperties(Ex).
PriceType	Int16		The form that the Price field is interpreted see *Appendix 18 – Price Type*
PresetType	Int16		The form that the Value parameter is interpreted see *Appendix 9 – Pump limit types*
Value	Double		The preset limit for this delivery.
Hose	Int16		The hose mask for permitted hoses, see *Appendix 10 – Permitted hoses mask*
Odometer	Double		An optional odometer reading saved with the resulting delivery.
Odometer2	Double		A second optional odometer reading saved with the resulting delivery.
Plate	String		An optional vehicle number plate saved with the resulting delivery. Maximum 10 chars.
ExtTransactionID	String		An optional external transaction ID saved with the resulting delivery. Maximum 20 chars.
DriverID	String		An optional driver ID string saved with the resulting delivery. Maximum 10 chars.
Authorisation	String		An optional authorization ID string saved with the resulting delivery. Maximum 10 chars.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
INVALID_PRESET_TYPE_RESULT	The preset type requested is not permitted for this type of authorize.
INVALID_HOSE_MASK_RESULT	None of the permitted hoses are configured on this pump.
PUMP_NOT_RESPONDING_RESULT	The pump is not responding.
PUMP_NOT_AVAILABLE_RESULT	The pump is performing a price change or a timeout after a prepay refund.
PUMP_IN_USE_RESULT	The pump is no longer idle.
PUMP_NOT_RESERVED_FOR_CTF_RESULT	The pump was not reserved for a preauth.
PUMP_NOT_RESERVED_BY_YOU_RESULT	The pump was not reserved by this client.
PUMP_IS_STOPPED_RESULT	The pump is currently temp stopped.

Remarks

This API is the final step in a payment delivery, it permits limiting the hose, grade and specifying a temporary price. It also provides a means of saving additional information along with the delivery. These optional fields can be retrieved with the GetDeliveryExt API after the delivery is completed.

See also

PaymentReserve, PaymentCancel, GetDeliveryExt

8.7. Pump authorization

The pump authorization APIs provide the various methods by which fueling point can be authorized etc.

8.7.1. AttendantAuthorise

Availability – EZClient.DLL, EZClient.SO.1, Web Service, EZTech.EZClient and EZTech.EZPump

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump being authorized. This parameter is not present for the EZTech.EZPump controls.
AttendantID	Int32		The ID of the attendant authorizing the pump.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
PUMP_NOT_RESPONDING_RESULT	The pump is not responding.
PUMP_NOT_AVAILABLE_RESULT	The pump is unavailable, it is either changing prices or timing out after a prepay refund.
PUMP_IN_USE_RESULT	The pump is not idle.
PUMP_CANNOT_BE_AUTHED_RESULT	The pump mode does not permit post pay deliveries.
PUMP_ALREADY_RESERVED_RESULT	The pump is already reserved for a prepay, preauth or payment delivery.
PUMP_IS_STOPPED_RESULT	The pump is currently temp stopped.
HAS_CURRENT_DELIVERY_RESULT	The pump has a current delivery which cannot be automatically stacked.

Remarks

This is API is used to manually authorize a pump without a limit, the resultant delivery is logged against the authorizing attendant. If the delivery is not started within the configured manually authorized timeout the pump reverts locked.

See also

CancelAuthorise

8.7.2. Authorise

Availability – EZClient.DLL, EZClient.SO.1, Web Service, EZTech.EZClient and EZTech.EZPump

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump being authorized. This parameter is not present for the EZTech.EZPump controls.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
PUMP_NOT_RESPONDING_RESULT	The pump is not responding.
PUMP_NOT_AVAILABLE_RESULT	The pump is unavailable, it is either changing prices or timing out after a prepay refund.
PUMP_IN_USE_RESULT	The pump is not idle.
PUMP_CANNOT_BE_AUTHED_RESULT	The pump mode does not permit post pay deliveries.
PUMP_ALREADY_RESERVED_RESULT	The pump is already reserved for a prepay, preauth or payment delivery.
PUMP_IS_STOPPED_RESULT	The pump is currently temp stopped.
HAS_CURRENT_DELIVERY_RESULT	The pump has a current delivery which cannot be automatically stacked.

Remarks

This is API is used to manually authorize a pump without a limit. If the delivery is not started within the configured manually authorized timeout the pump reverts locked.

See also

CancelAuthorise, TempStop, TerminateDelivery, ReAuthorise

8.7.3. CancelAuthorise

Availability – EZClient.DLL, EZClient.SO.1, Web Service, EZTech.EZClient and EZTech.EZPump

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump having the manual authorization cancelled. This parameter is not present for the EZTech.EZPump controls.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
PUMP_NOT_RESPONDING_RESULT	The pump is not responding.
PUMP_NOT_AUTHED_RESULT	The pump was not manually authorized.
PUMP_ALREADY_RESERVED_RESULT	The pump has a prepay, preauth or payment reserved placed on it.

Remarks

The API is provided to cancel any of the manual authorizations which can be done on a pump.

See also

LoadPreset, AttendantAuthorise, Authorise

8.7.4. TempStop

Availability – EZClient.DLL, EZClient.SO.1, Web Service, EZTech.EZClient and EZTech.EZPump

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump being temp stopped. This parameter is not present for the EZTech.EZPump controls.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
PUMP_NOT_RESPONDING_RESULT	The pump is not responding.
PUMP_IS_STOPPED_RESULT	The pump is already temp stopped.

Remarks

This API is used to place a temp stop on a pump which is either delivering or authorized. A temp stop will stop the pump delivering or prevent it from starting. ReAuthorise is used to clear the temp stop.

See also

ReAuthorise

8.7.5. TerminateDelivery

Availability – EZClient.DLL, EZClient.SO.1, Web Service, EZTech.EZClient and EZTech.EZPump

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump for which to terminate the delivery. This parameter is not present for the EZTech.EZPump controls.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
PUMP_NOT_RESPONDING_RESULT	The pump is not responding.
PUMP_NOT_DELIVERING_RESULT	The pump is not currently delivering.

Remarks

This API is used to remotely terminate a delivery in progress. If the pump is currently delivering the pump is temp stopped first. The current value, volume, price and grade will be used to generate a delivery. Once the delivery has been terminated the pump cannot be authorized or re-authorized until the hose is returned. Once the hose is returned the electronic totals for the pump will be consulted and logged in the normal way.

See also

TempStop

8.7.6. ReAuthorise

Availability – EZClient.DLL, EZClient.SO.1, Web Service,
EZTech.EZClient and EZTech.EZPump

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump being re-authorized. This parameter is not present for the EZTech.EZPump controls.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
PUMP_NOT_RESPONDING_RESULT	The pump is not responding.
PUMP_NOT_STOPPED_RESULT	The pump is not temp stopped.
DELIVERY_TERMINATED_RESULT	A delivery on the pump has been terminated, and hence the pump cannot be re-authorized.

Remarks

The API is used to re-start a pump which has been temp stopped.

See also

ReAuthorise

8.7.7. LoadPreset

Availability – EZClient.DLL, EZClient.SO.1, Web Service, EZTech.EZClient and EZTech.EZPump

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump that is being authorized with a preset. This parameter is not present for the EZTech.EZPump controls.
LimitType	Int16		The limit type being place on the delivery, DOLLAR_PRESET_TYPE and VOLUME_PRESET_TYPE are the only valid values here. See *Appendix 9 – Pump limit types* for more information.
Value	Double		The dollar or volume limit being placed on the delivery.
Hose	Int16		The permitted hose/hoses for this delivery. See *Appendix 10 – Permitted hoses mask* for more information.
PriceLevel	Int16		The selected price level for this delivery.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
INVALID_PRESET_TYPE_RESULT	The preset type requested is not permitted for this type of authorize.
INVALID_PRESET_AMOUNT_RESULT	The preset amount is less than the configured minimum.
INVALID_HOSE_MASK_RESULT	None of the permitted hoses are configured on this pump.
PUMP_NOT_RESPONDING_RESULT	The pump is not responding.
PUMP_NOT_AVAILABLE_RESULT	The pump is unavailable, it is either changing prices or timing out after a prepay refund.
PUMP_IN_USE_RESULT	The pump is not idle.
PUMP_CANNOT_BE_AUTHED_RESULT	The pump mode does not permit post pay deliveries.
PUMP_ALREADY_RESERVED_RESULT	The pump is already reserved for a prepay, preauth or payment delivery.
PUMP_IS_STOPPED_RESULT	The pump is currently temp stopped.
HAS_CURRENT_DELIVERY_RESULT	The pump has a current delivery which cannot be automatically stacked.

Remarks

This API is used to manually authorize a pump with value limit, volume limit and/or hose mask. Once the delivery is completed it will need to be cleared as a Post-pay delivery by any logged on EZServer client. If the delivery is not started within the configured manually authorized timeout the preset is lost and the pump reverts locked.

See also

CancelAuthorise

8.7.8. TagAuthorise

Availability – EZClient.DLL, EZClient.SO.1, Web Service, EZTech.EZClient and EZTech.EZPump

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump that is being authorized with a preset.
Tag	Int64		The tag to be saved with the resulting delivery.
LimitType	Int16		The limit type being place on the delivery, DOLLAR_PRESET_TYPE and VOLUME_PRESET_TYPE are the only valid values here. See *Appendix 9 – Pump limit types* for more information
Value	Double		The dollar or volume limit being placed on the delivery.
Hose	Int16		The permitted hose/hoses for this delivery. See *Appendix 10 – Permitted hoses mask* for more information.
PriceLevel	Int16		The selected price level for this delivery.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
INVALID_PRESET_TYPE_RESULT	The preset type requested is not permitted for this type of authorize.
INVALID_PRESET_AMOUNT_RESULT	The preset amount is less than the configured minimum.
INVALID_HOSE_MASK_RESULT	None of the permitted hoses are configured on this pump.
PUMP_NOT_RESPONDING_RESULT	The pump is not responding.
PUMP_NOT_AVAILABLE_RESULT	The pump is unavailable, it is either changing prices or timing out after a prepay refund.
PUMP_IN_USE_RESULT	The pump is not idle.
PUMP_IS_STOPPED_RESULT	The pump is currently temp stopped.
PUMP_CANNOT_BE_AUTHED_RESULT	The pump mode does not permit post pay deliveries.
PUMP_ALREADY_RESERVED_RESULT	The pump is already reserved for a prepay, preauth or payment delivery.
PUMP_IS_STOPPED_RESULT	The pump is currently temp stopped.
HAS_CURRENT_DELIVERY_RESULT	The pump has a current delivery which cannot be automatically stacked.

Remarks

This API is used to authorize a pump when external device is used to capture the tag, this tag parameter can be retrieved when the delivery is complete along with its other properties.

See also

CancelAuthorise, GetPumpDeliveryProperties(Ex, Ex2, Ex3, Ex4)

8.7.9. LoadPresetWithPrice

Availability – EZClient.DLL, EZClient.SO.1, Web Service, EZTech.EZClient and EZTech.EZPump

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump that is being authorized with a preset. This parameter is not present for the EZTech.EZPump controls.
LimitType	Int16		The limit type being place on the delivery, DOLLAR_PRESET_TYPE and VOLUME_PRESET_TYPE are the only valid values here. See *Appendix 9 – Pump limit types* for more information
Value	Double		The dollar or volume limit being placed on the delivery.
Hose	Int16		The permitted hose/hoses for this delivery. See *Appendix 10 – Permitted hoses mask* for more information.
PriceLevel	Int16		The selected price level for this delivery.
Price	Double		The price that is to be used for this delivery.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
INVALID_PRESET_TYPE_RESULT	The preset type requested is not permitted for this type of authorize.
INVALID_PRESET_AMOUNT_RESULT	The preset amount is less than the configured minimum.
INVALID_HOSE_MASK_RESULT	None of the permitted hoses are configured on this pump.
PUMP_NOT_RESPONDING_RESULT	The pump is not responding.
PUMP_NOT_AVAILABLE_RESULT	The pump is unavailable, it is either changing prices or timing out after a prepay refund.
PUMP_IN_USE_RESULT	The pump is not idle.
PUMP_CANNOT_BE_AUTHED_RESULT	The pump mode does not permit post pay deliveries.
PUMP_ALREADY_RESERVED_RESULT	The pump is already reserved for a prepay, preauth or payment delivery.
PUMP_IS_STOPPED_RESULT	The pump is currently temp stopped.

HAS_CURRENT_DELIVERY_RESULT	The pump has a current delivery which cannot be automatically stacked.

Remarks

The API functions in same way as LoadPreset with the addition that it permits the application of a specific price, once the delivery is completed the original price at the pump will be restored.

See also

LoadPreset, CancelAuthorise

8.8. Global functions

The global pump functions will affect all defined fueling points or tanks as a whole.

8.8.1. AllStop

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
N/A			

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

The API is provided to place a temp stop on all pumps with one call. This can be cleared with AllReAuthorise, or each pump can be individually ReAuthorised.

See also

ReAuthorise, TempStop, AllReAuthorise, AllStopIfIdle

8.8.2. AllStopIfIdle

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
N/A			

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
PUMP_IN_USE_RESULT	One or more of the pumps is not idle, or has available deliveries.

Remarks

This API is provided to place a temp stop on all pumps with one call, if the pumps are all currently idle and do not have deliveries pending. This can be cleared with AllReAuthorise, or each pump can be individually ReAuthorised. This call is provided to assist with the shift end procedure, it is typically called when all the pumps become idle and then is ReAuthorised with the AllReAuthorise API.

See also

ReAuthorise, TempStop, AllReAuthorise, AllStop

8.8.3. AllAuthorise

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
N/A			

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is provided so that all pumps can be authorized in one operation. To reverse this CancelAuthorise must be called for each individual pump.

See also

CancelAuthorise, AllStop, AllReAuthorise, AllStopIfIdle

8.8.4. AllReAuthorise

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
N/A			

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is provided so that AllStop can be reversed in one operation. It has the same effect as calling ReAuthorise for each individual pump.

See also

ReAuthorise, AllStop, TempStop, AllStopIfIdle

8.8.5. GetAllPumpStatuses

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient.

Parameters

Parameter	Type	API	Description
States	String*		A string containing all of the current pump states, position 1 contains the state for pump number 1 and position n contains the state for pump n, the length of the string corresponds to the highest pump number. To convert the string element to the actual pump state, subtract '0' (48 decimal, or 0x30 hex). See *Appendix 1 – Pump states* for more information.
CurrentHoses	String*		A string containing all of the current pump hoses, position 1 contains the current hose for pump number 1 and position n contains the current hose for pump n, the length of the string corresponds to the highest pump number. To convert the string element to the actual pump hose number, subtract '0' (48 decimal, or 0x30 hex).
DeliveriesCount	String*		A string containing a count of all of the available deliveries, position 1 contains the available deliveries for pump number 1 and position n contains the available deliveries for pump n, the length of the string corresponds to the highest pump number. To convert the string element to the actual number of available deliveries, subtract '0' (48 decimal, or 0x30 hex).

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is used to quickly obtain a pump status summary for all of the installed hoses.

See also

GetPumpStatus

8.8.6. ReadAllTanks

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is used to force the ATG module initiate a read of all the tank statuses.

See also

8.9. Deliveries

Deliveries are the basic fuel delivery transaction. They may include links to the attendant and/or client associated with the delivery.

8.9.1. GetDeliveriesCount

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Count	Int32*		The returned total number of delivery objects currently in the server.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is used determine the total number of deliveries currently in the server. Once this is known, the IDs of the individual delivery objects can be obtained using GetDeliveryByOrdinal.

See also

GetDeliveryByOrdinal

8.9.2. GetDeliveryByOrdinal

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Index	Int32		The index of the desired delivery object, this can be between 1 and the total number of deliveries, as returned by GetDeliveriesCount.
ID	Int32*		The returned ID of the delivery object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used to retrieve the delivery ID of a delivery using its ordinal value in the server's internal deliveries objects list. The deliveries are ordered in this list by ID, the ordinal value is based on a filtered list of uncleared deliveries.

See also

GetDeliveriesCount

8.9.3. GetDeliveryProperties(Ex, Ex2, Ex3, Ex4)

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32	All	The ID of the delivery object for which the properties are being requested.
HoseID	Int32*	All	The ID of the hose which made the delivery.
State	Int16*	All	The current state of the delivery, see *Appendix 4 – Delivery states* for more information.
Type	Int16*	All	The type of the delivery, see *Appendix 3 – Delivery types* for more information.
Volume	Double*	All	The total volume for the delivery.
PriceLevel	Int16*	All	The price level this delivery was taken at.
Price	Double*	All	The unit price for the delivery.
Value	Double*	All	The total value of this delivery.
Volume2	Double*	All	This property is reserved for future use and will be retuned as zero.
CompletedDT	DateTime*	All	The date and time the delivery was completed.
ReservedBy	Int32*	All	The ID of the client who reserved the pump for this delivery, or NULL ID if it was not reserved.
LockedBy	Int32*	All	The ID of the client has locked this delivery or NULL ID if it was not reserved.
AttendantID	Int32*	All	The ID of the Attendant that authorized this delivery or was logged onto the pump at the time the delivery was done. If neither of these was the case then this is NULL ID.
Age	Int32*	All	The Age of the delivery in seconds since it was completed.
ClearedDT	DateTime*	Ex, Ex2, Ex3, Ex4	The date and time the delivery was cleared from the EZServer.
VolumeETot	Double*	Ex	The electronic volume total for this hose at the end of this delivery.
Volume2ETot	Double*	Ex	Always returned as zero. Reserved for future use.
ValueETot	Double*	Ex	The electronic value total for this hose at the end of this delivery.
OldVolumeETot	Double*	Ex2, Ex3, Ex4	The electronic volume total for this hose at the start of this delivery.
OldVolume2ETot	Double*	Ex2, Ex3, Ex4	Always returned as zero. Reserved for future use.
OldValueETot	Double*	Ex2, Ex3, Ex4	The electronic value total for this hose at the start of this delivery

NewVolumeETot	Double*	Ex2, Ex3, Ex4	The electronic volume total for this hose at the end of this delivery.
NewVolume2ETot	Double*	Ex2, Ex3, Ex4	Always returned as zero. Reserved for future use.
NewValueETot	Double*	Ex2, Ex3, Ex4	The electronic value total for this hose at the end of this delivery
Tag	Int64*	Ex, Ex2, Ex3, Ex4	The authorization tag as passed in TagAuthorise or -1 if not authorized by tag auth.
Duration	Int32*	Ex2, Ex3, Ex4	The duration of the delivery in seconds.
CardClientID	Int32*	Ex3, Ex4	The ID of the card client that authorized this delivery, or NULL ID.
PeakFlowRate	Double*	Ex4	The peak flow achieved during this delivery in liters per minute.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API returns all the property values of a delivery object for the given ID.

See also

SetDeliveryProperties, GetDeliveryExt

8.9.4. SetDeliveryProperties(Ex, Ex2, Ex3, Ex4)

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient
For DB clients only.

Parameters

Parameter	Type	API	Description
ID	Int32	All	The ID of the delivery object for which the properties are being set.
HoseID	Int32	All	The ID of the hose which made the delivery.
State	Int16	All	The current state of the delivery, see *Appendix 4 – Delivery states* for more information.
Type	Int16	All	The type of the delivery, see *Appendix 3 – Delivery types* for more information.
Volume	Double	All	The total volume for the delivery.
PriceLevel	Int16	All	The price level this delivery was taken at.
Price	Double	All	The unit price for the delivery.
Value	Double	All	The total value of this delivery.
Volume2	Double	All	This property is reserved for future use and will be retuned as zero.
CompletedDT	DateTime	All	The date and time the delivery was completed.
LockedBy	Int32	All	The ID of the client who has locked this delivery or NULL ID if it was not reserved.
ReservedBy	Int32	All	The ID of the client who reserved the pump for this delivery, or NULL ID if it was not reserved.
AttendantID	Int32	All	The ID of the Attendant that authorized this delivery or was logged onto the pump at the time the delivery was done. If neither of these was the case then this is NULL ID.
Age	Int32	All	The age in seconds of this delivery since it was completed.
ClearedDT	DateTime	Ex, Ex2, Ex3, Ex4	The date and time the delivery was cleared from the EZServer.
VolumeETot	Double	Ex	The electronic volume total for this hose at the end of this delivery
Volume2ETot	Double	Ex	Always returned as zero. Reserved for future use.
ValueETot	Double	Ex	The electronic value total for this hose at the end of this delivery.
OldVolume2ETot	Double	Ex2, Ex3, Ex4	The electronic volume total for this hose at the start of this delivery.
OldVolumeETot	Double	Ex2, Ex3, Ex4	Always returned as zero. Reserved for future use.
OldValueETot	Double	Ex2,	The electronic value total for this hose at the start of

		Ex3, Ex4	this delivery
NewVolume2ETot	Double	Ex2, Ex3, Ex4	The electronic volume total for this hose at the end of this delivery.
NewVolumeETot	Double	Ex4	Always returned as zero. Reserved for future use.
NewValueETot	Double	Ex4	The electronic value total for this hose at the end of this delivery
Tag	Int64	Ex, Ex2, Ex3, Ex4	The authorization tag as passed in TagAuthorise or -1 if not authorized by tag auth.
Duration	Int32	Ex2, Ex3, Ex4	The duration of the delivery in seconds.
CardClientID	Int32	Ex3, Ex4	The ID of the card client that authorized this delivery, or NULL ID.
PeakFlowRate	Double	Ex4	The peak flow achieved during this delivery in liters per minute.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
INVALID_OBJECT_PARAMETER_RESULT	One of the property values passed is out of range.
INVALID_OBJECT_LINK_RESULT	One of the IDs passed representing a related object references a non-existing object.

Remarks

This API is provided so that delivery objects can be created and maintained on the server. If the given delivery ID already exists then the properties for this object will be overwritten with the values passed, otherwise a new delivery object with this ID is created and its properties set to the values passed. The HoseID must represent an object which already exists on the server or the call will fail.

Normally when deliveries are completed the EZServer will automatically create a new delivery object with a generated ID. However if the EZServer has just been started up it is the responsibility of the DB Client to pre-load all the un-cleared deliveries which are still outstanding. Note that there is no DeleteDelivery API as ClearDelivery will automatically clear the deliveries from the system, and has the same effect.

See also

GetDeliveryProperties, SetNextDeliveryID

8.9.5. LockDelivery

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the delivery that is being locked.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
DELIVERY_ALREADY_LOCKED_RESULT	The delivery has already been locked, by you or another client.
DELIVERY_IS_RESERVED_RESULT	The delivery was done as a prepay or preauth delivery and is reserved for another EZClient.
INVALID_CLIENT_ID_RESULT	The ID of this client is not a valid EZServer client.

Remarks

This API is used to lock a delivery; this will prevent other terminals from locking or clearing the delivery while this client has it included in a sale. Once the delivery is successfully locked it can then be cleared which will remove it from the EZServer.

See also

UnlockDelivery, ClearDelivery, LockAndClearDelivery

8.9.6. UnlockDelivery

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the delivery that is being locked.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
DELIVERY_NOT_LOCKED_RESULT	The delivery is not currently locked.
DELIVERY_NOT_LOCKED_BY_YOU_RESULT	The delivery is locked by another EZServer client.

Remarks

The API performs the opposite of LockDelivery. It is provided so that a delivery already included into a sale (locked) can be voided (unlocked) making it available for other clients.

See also

LockDelivery, ClearDelivery, LockAndClearDelivery

8.9.7. ClearDelivery

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the delivery being cleared.
Type	Int16		The type the delivery is being cleared as. See *Appendix 3 – Delivery types* for more information.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
DELIVERY_NOT_LOCKED_RESULT	The delivery is not currently locked.
DELIVERY_NOT_LOCKED_BY_YOU_RESULT	The delivery is locked by another EZServer client.
DELIVERY_CANNOT_BE_CLEARED_AS_THIS_TYPE_RESULT	The original type of the delivery does not permit it to be cleared as this type.

Remarks

This API is used after the delivery has been locked and the sale is finalized, this will result in the delivery being removed from the EZServer. If the original type of the delivery was POSTPAY it may be cleared as either MONITOR, TEST, DRIVEOFF or OFFLINE delivery types, otherwise it must be cleared with same type as its original delivery type.

See also

LockDelivery, UnlockDelivery, LockAndClearDelivery

8.9.8. LockAndClearDelivery

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the delivery being locked and cleared.
Type	Int16		The type the delivery is being cleared as. See *Appendix 3 – Delivery types* for more information.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
DELIVERY_NOT_LOCKED_BY_YOU_RESULT	The delivery is locked by another EZServer client.
DELIVERY_CANNOT_BE_CLEARED_AS_THIS_TYPE_RESULT	The original type of the delivery does not permit it to be cleared as this type.

Remarks

This API is used to lock and clear a delivery as a single operation; this will result in the delivery being removed from the EZServer. If the original type of the delivery was POSTPAY it may be cleared as either MONITOR, TEST, DRIVEOFF or OFFLINE delivery types, otherwise it must be cleared with same type as its original delivery type. This call is the equivalent to calling LockDelivery followed by a ClearDelivery. This call is intended to be used by host systems, which clear the deliveries as they are completed, passing the control/responsibility for the delivery to the host system.

See also

LockDelivery, UnlockDelivery, ClearDelivery

8.9.9. SetNextDeliveryID

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient
For DB clients only.

Parameters

Parameter	Type	API	Description
ID	Int32		The ID used to save the next delivery completed.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

As the EZServer captures the deliveries done on the pumps, they are associated with a unique ID. This ID is generated by simply incrementing in internal long variable. However as the EZServer has no knowledge of the deliveries already logged into the database, it is necessary for the DB Client to give it a starting value. This would normally be the maximum logged delivery ID plus one.

See also

SetDeliveryProperties

8.9.10. AckDeliveryDBlog

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the delivery to be acknowledged as logged to the database.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

The API is used to acknowledge that a delivery has been logged to data base or third party system. For this to work the **ConfirmDeliveryLog** flag in the EZServer.ini must be set to true. The concept behind this that there may be a third party application that want to log all of the deliveries to an independent database. Deliveries will be retained in the EZserver local database until they have been logged.

See also

GetDeliveryIDByOrdinalNotLogged, GetDeliveriesCountNotLogged, *EZServer INI file*

8.9.11. GetDeliveryIDByOrdinalNotLogged

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Ordinal	Int32		The index of the desired delivery object, this can be between 1 and the total number of deliveries, as returned by GetDeliveriesCountNotLogged
ID	Int32*		The returned ID of the delivery object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API returns the ID of a delivery which has not already been logged. The deliveries are ordered in this list by ID, the ordinal value is based on a filtered list of unlogged deliveries.

See also

GetDeliveriesCountNotLogged, AckDeliveryDBlog

8.9.12. GetDeliveriesCountNotLogged

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Count	Int32*		The returned count of deliveries not logged.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

The API returns the count of deliveries in the filtered list of unlogged deliveries.

See also

AckDeliveryDBlog, GetDeliveryIDByOrdinalNotLogged

8.9.13. AckDeliveryVolLog

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the delivery to be acknowledged as logged to volumetric logging system.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

The API is used to acknowledge that a delivery has been logged to volumetric logging system. For this to work the **ConfirmVolumetricLog** flag in the EZServer.ini must be set to true. The concept behind this that there may be a third party volumetric logging application that wants to log all of the deliveries to an independent database. Deliveries will be retained in the EZserver local database until they have been logged.

See also

GetDeliveryIDByOrdinalNotVolLogged, GetDeliveriesCountNotVolLogged

8.9.14. GetDeliveryIDByOrdinalNotVolLogged

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Ordinal	Int32		The index of the desired delivery object, this can be between 1 and the total number of deliveries, as returned by GetDeliveriesCountNotVolLogged.
ID	Int32*		The returned ID of the delivery object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API returns the ID of a delivery which has not already been volumetric logged. The deliveries are ordered in this list by ID, the ordinal value is based on a filtered list of unlogged volumetric deliveries.

See also

AckDeliveryVolLog, GetDeliveriesCountNotVolLogged

8.9.15. GetDeliveriesCountNotVolLogged

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Count	Int32*		The returned count of deliveries not volumetric logged.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

The API returns the count of deliveries in the filtered list of unlogged volumetric deliveries.

See also

AckDeliveryVolLog, GetDeliveryIDByOrdinalNotVolLogged

8.9.16. GetAllDeliveriesCount

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Count	Int32*		The returned count of all deliveries, regardless of cleared or logged etc.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

The API returns the total number of deliveries regardless of logged or cleared status.

See also

GetAllDeliveryByOrdinal

8.9.17. GetAllDeliveryByOrdinal

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Ordinal	Int32		The index of the desired delivery object, this can be between 1 and the total number of deliveries, as returned by GetAllDeliveriesCount.
ID	Int32*		The returned ID of the delivery object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API returns the ID of a delivery from the unfiltered delivery list. The deliveries are ordered in this list by ID.

See also

GetAllDeliveriesCount

8.9.18. GetDeliverySummary(Ex, Ex2, Ex3)

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32	All	The ID of the delivery object for which the properties are being requested.
HoseID	Int32*	All	The hose object identifier for the hose which dispensed this delivery.
HoseNumber	Int32*	All	The number of the hose which dispensed this delivery.
HosePhysicalNumber	Int32*	All	The physical number of the hose for this delivery.
PumpID	Int32*	All	The pumps object identifier of the pump which dispensed this delivery. This parameter is not present for the StatusEvent on the EZTech.EZPump controls.
PumpNumber	Int32*	All	The logical pump number of the pump which dispensed this delivery. This parameter is not present for the StatusEvent on the EZTech.EZPump controls.
PumpName	String*	All	The name of the fueling position for this delivery.
TankID	Int32*	All	The ID of the tank for this delivery.
TankNumber	Int32*	All	The number of the tank for this delivery.
TankName	String*	All	The name of the tank for this delivery.
GradeID	Int32*	All	The grades object identifier for the grade of this delivery.
GradeNumber	Int32*	All	The number of the grade for this delivery.
GradeName	Int32*	All	The full name of the grade for this delivery.
GradeShortName	String*	All	The short name of the grade for this delivery.
GradeCode	String*	All	The grade code for this delivery.
State	Int16*	All	The current state of the delivery, see *Appendix 4 – Delivery states* for more information.
Type	Int16*	All	The type of the delivery, see *Appendix 3 – Delivery types* for more information.
Volume	Double*	All	The total volume for the delivery.
PriceLevel	Int16*	All	The price level this delivery was taken at.
Price	Double*	All	The unit price for the delivery.
Value	Double*	All	The total value of this delivery.
Volume2	Double*	All	This property is reserved for future use and will be retuned as zero.
CompletedDT	DateTime*	All	The date and time the delivery was completed.
LockedBy	Int32*	All	The ID of the client has locked this delivery or NULL ID if it was not reserved.
ReservedBy	Int32*	All	The ID of the client who reserved the pump for this delivery, or NULL ID if it was not reserved.

AttendantID	Int32*	All	The ID of the Attendant that authorized this delivery or was logged onto the pump at the time the delivery was done. If neither of these was the case then this is NULL ID.
Age	Int32*	All	The Age of the delivery in seconds since it was completed.
ClearedDT	DateTime*	All	The date and time the delivery was cleared from the EZServer.
VolumeETot	Double*	Base	The electronic volume total for this hose at the end of this delivery.
Volume2ETot	Double*	Base	Always returned as zero. Reserved for future use.
ValueETot	Double*	Base	The electronic value total for this hose at the end of this delivery.
OldVolumeETot	Double*	Ex, Ex2, Ex3	The electronic volume total for this hose at the start of this delivery.
OldVolume2ETot	Double*	Ex, Ex2, Ex3	Always returned as zero. Reserved for future use.
OldValueETot	Double*	Ex, Ex2, Ex3	The electronic value total for this hose at the start of this delivery
NewVolumeETot	Double*	Ex, Ex2, Ex3	The electronic volume total for this hose at the end of this delivery.
NewVolume2ETot	Double*	Ex2, Ex3, Ex4	Always returned as zero. Reserved for future use.
NewValueETot	Double*	Ex, Ex2, Ex3	The electronic value total for this hose at the end of this delivery
Tag	Int64*	All	The authorization tag as passed in TagAuthorise or -1 if not authorized by tag auth.
Duration	Int32	Ex, Ex2, Ex3	The duration of the delivery in seconds.
AttendentNumber	Int32*	Ex2, Ex3	The number of the pump attendant associated with this delivery or 0
AttendentName	String*	Ex2, Ex3	The name of the pump attendant associated with this delivery
AttendentTag	Int64*	Ex2, Ex3	The tag of the pump attendant associated with this delivery or -1
CardClientID	Int32*	Ex2, Ex3	The ID of the card client that authorized this delivery. If this was not the case then this is NULL ID.
CardClientNumber	Int32*	Ex2, Ex3	The number of the card client associated with this delivery, or zero.
CardClientName	String*	Ex2, Ex3	The name of the card client associated with this delivery.
CardClientTag	Int64*	Ex2, Ex3	The card client tag associated with this delivery, or -1.
PeakFlowRate	Double*	Ex4	The peak flow rate obtained during the delivery

			in liters per minute.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is functionally equivalent to GetDeliveryProperties(Ex, Ex2, Ex3, Ex4), however it expands and includes the pump, hose, tank and grade details for the delivery.

See also

GetDeliveryProperties(Ex, Ex2, Ex3, Ex4)

8.9.19. GetDeliveryExt

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the delivery in question.
Plate	String*		The vehicle number plate associated with the delivery.
Odometer	Double*		The main odometer reading associated with the delivery.
Odometer2	Double*		The second odometer reading associated with the delivery.
TransactionID	String*		The transaction ID string associated with the delivery.
DriverID	String*		The driver ID string associated with the delivery.
AuthID	String*		The authorization ID string associated with the delivery.
AuthType	Int16*		The authorization type associated with the delivery.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API retrieves the extended delivery properties, these properties are associated with the delivery via the PaymentAuthorise API.

See also

PreauthAuthorise, SetDeliveryExt

8.9.20. SetDeliveryExt

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the delivery in question.
Plate	String		The vehicle number plate associated with the delivery.
Odometer	Double		The main odometer reading associated with the delivery.
Odometer2	Double		The second odometer reading associated with the delivery.
TransactionID	String		The transaction ID string associated with the delivery.
DriverID	String		The driver ID string associated with the delivery.
AuthID	String		The authorization ID string associated with the delivery.
AuthType	Int16		The authorization type associated with the delivery.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API set the extended properties for a delivery, this API is used internally in the EZserver as a result of the PaymentAuthorise API, and as such it should not be necessary to do this.

See also

PaymentAuthorise, GetDeliveryExt

8.9.21. GetPumpDeliveryProperties(Ex, Ex2, Ex3, Ex4)

Availability – EZTech.EZClient for GetPumpDeliveryProperties
EZTech.EZPump for GetDeliveryProperties

Parameters

Parameter	Type	API	Description
ID	Int32	All	The ID of the pump that made the delivery. This parameter is not present for the EZTech.EZPump controls.
Index	Int16	All	The index into the delivery stack on the pump. This can be from one to the size of the stack, see stack size parameter in GetNextPumpEvent(Ex, Ex2, Ex3) / StatusEvent.
DeliveryID	Int32*	All	The returned ID of the delivery.
Type	Int16*	All	The type of the delivery, see *Appendix 3 – Delivery types* for more information.
State	Int16*	All	The current state of the delivery, see *Appendix 4 – Delivery states* for more information.
HoseID	Int32*	All	The ID of the hose which made the delivery.
HoseNum	Int16*	All	The logical number of the hose which made the delivery.
GradeID	Int32*	Ex,Ex2, Ex3,Ex4	The ID of the grade for this delivery.
GradeName	String*	Ex,Ex2, Ex3,Ex4	The name of the grade for this delivery.
ShortGradeName	String*	Ex,Ex2, Ex3,Ex4	The short name of the grade for this delivery.
PriceLevel	Int16*	All	The price level this delivery was taken at.
Price	Double*	All	The unit price that this delivery was taken at.
Volume	Double*	All	The total volume of this delivery.
Value	Double*	All	The total value of this delivery.
LockedBy	Int32*	All	The ID of the client that has this delivery locked or NULL ID if it is unlocked.
ReservedBy	Int32*	All	The ID of the client who reserved the pump for this delivery, or NULL ID if it was not reserved.
Age	Int32*	All	The time in seconds since this delivery was completed.
CompletedDT	Double*	All	The date and time the delivery was completed.
AttendantID	Int32*	All	The ID of the Attendant that authorized this delivery or was logged onto the pump at the time the delivery was done. If neither of these was the case then this is NULL ID.
VolumeETot	Double*	Ex	The electronic volume total for this hose at the end of this delivery.

Volume2ETot	Double*	Ex	Always returned as zero. Reserved for future use.
ValueETot	Double*	Ex	The electronic value total for this hose at the end of this delivery.
OldVolumeETot	Double*	Ex2,Ex3, Ex4	The electronic volume total for this hose at the start of this delivery.
OldVolume2ETot	Double*	Ex2,Ex3, Ex4	Always returned as zero. Reserved for future use.
OldValueETot	Double*	Ex2,Ex3, Ex4	The electronic value total for this hose at the start of this delivery
NewVolumeETot	Double*	Ex2,Ex3, Ex4	The electronic volume total for this hose at the end of this delivery.
NemVolume2ETot	Double*	Ex2,Ex3, Ex4	Always returned as zero. Reserved for future use.
NewValueETot	Double*	Ex2,Ex3, Ex4	The electronic value total for this hose at the end of this delivery
Tag	Int64*	Ex,Ex2, Ex3,Ex4	The authorization tag as passed in TagAuthorise or -1 if not authorized by tag auth.
Duration	Int32*	Ex2,Ex3, Ex4	The duration of the delivery in seconds.
CardClientID	Int32*	Ex3 Ex4	The ID of the card client that authorized this delivery, or NULL ID.
PeakFlowRate	Double*	Ex4	The peak flow achieved during this delivery in liters per minute.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
NO_DELIVERY_AVAILABLE_RESULT	The delivery requested does exist.

Remarks

This API can be used to get all the details about a specific delivery by using the pump ID and position in the delivery stack.

See also

GetPumpStatus

8.9.22. ReserveTypeString

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Type	Int16		The pump reserve value.

Return value

An English string describing the pump reserve value passed.

Remarks

This API can be used to convert a pump reserve value to a pump reserve description string in English.

See also

Appendix 2 – Pump reserves

8.9.23. GetDuration

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the delivery in question.
Duration	Int32*		The duration of the delivery in seconds.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API retrieves the duration in seconds for a specific delivery.

See also

GetDeliveryProperties(Ex, Ex2, Ex3, Ex4)

8.9.24. StackCurrentDelivery

Availability – EZClient.DLL, EZClient.SO.1, Web Service, EZTech.EZClient and EZTech.EZPump

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the pump for which to stack the current delivery. This parameter is not present for the EZTech.EZPump controls.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
STACK_IS_DISABLED_RESULT	Stacking deliveries is not permitted on this pump.
STACK_FULL_RESULT	The delivery stack for this pump is full.
NO_CURRENT_DELIVERY_RESULT	This pump has no current delivery.
DELIVERY_TYPE_CANNOT_BE_STACKED_RESULT	This type of delivery cannot be stacked.

Remarks

This API pushes the current delivery onto the delivery stack. The state of the delivery will change from current to stacked and a PumpStatusEvent and DeliveryEvent will be received if it was successfully stacked.

See also

GetNextPumpEvent(Ex, Ex2, Ex3) / StatusEvent, GetNextDeliveryEvent

8.9.25. DeliveryTypeString

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Type	Int16		A delivery type value.

Return value

An English string describing the delivery type value passed.

Remarks

This API can be used to convert a delivery type value to a delivery type description string in English.

See also

Appendix 3 – Delivery types

8.9.26. DeliveryStateString

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
State	Int16		

Return value

An English string describing the delivery state value passed.

Remarks

This API can be used to convert a delivery state value to a delivery state description string in English.

See also

Appendix 4 – Delivery states

8.10. Hoses

Hoses are the point where the fuel is dispenser, they have two parent objects, tanks and pumps. As such the tanks and pumps must be configured prior to configuring the hoses.

8.10.1. GetHosesCount

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Count	Int32*		The returned total number of hoses configured in the server.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is used to determine the total number of hoses currently configured in the server. Once this is known, the IDs of the individual hose objects can be obtained using GetHoseByOrdinal.

See also

GetHoseByOrdinal

8.10.2. GetHoseByOrdinal

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Index	Int32		The index of the desired hose object, this can be between 1 and the total number of hoses, as returned by GetHosesCount
ID	Int32*		The returned ID of the hose object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used to retrieve the hose ID of a hose using its ordinal value in the server's internal hose objects list. The hoses are ordered in this list by ID.

See also

GetHosesCount

8.10.3. GetHoseProperties(Ex, Ex2)

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32	All	The ID of the hose object for which the properties are being requested.
Number	Int32*	All	The logical number of the hose, this number is with respect to the pump, i.e. a number 1, means the first hose on this pump.
PumpID	Int32*	All	The ID of the pump to which this hose is connected.
TankID	Int32*	All	The ID of the tank to which this hose is connected.
PhysicalNumber	Int32*	All	The physical number of the hose, this number is unique across all hoses and represents the public key for this object.
MtrTheoValue	Double*	All	The current theoretical total dollar value delivered on this hose, since it was first configured in the system.
MtrTheoVolume	Double*	All	The current theoretical total volume delivered on this hose, since it was first configured in the system.
MtrElecValue	Double*	All	The current electronic total dollar value delivered on this hose, as reported by the pump. This value is updated at the end of every delivery and when a pump starts responding.
MtrElecVolume	Double*	All	The current electronic total volume delivered on this hose as reported by the pump. This value is updated at the end of every delivery and when a pump starts responding.
UVEAntenna	Int16*	Ex	The RFiD antenna number associated with this hose.
Price1	Double*	Ex2	Specific level 1 price for this hose, or zero if the associated grades price is to be used.
Price2	Double*	Ex2	Specific level 2 price for this hose, or zero if the associated grades price is to be used.
Enabled	Int16*	Ex2	Is the hose enabled yes/no

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API returns all the property values of a hose object for the given ID.

See also

SetHoseProperties(Ex, Ex2), DeleteHose

8.10.4. SetHoseProperties(Ex, Ex2)

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient
For DB clients only.

Parameters

Parameter	Type	API	Description
ID	Int32	All	The ID of the hose object for which the properties set.
Number	Int32	All	The logical number of the hose, this number is with respect to the pump, i.e. a number 1, means the first hose on this pump.
PumpID	Int32	All	The ID of the pump to which this hose is connected.
TankID	Int32	All	The ID of the tank to which this hose is connected.
PhysicalNumber	Int32	All	The physical number of the hose, this number is unique across all hoses and represents the public key for this object.
MtrTheoValue	Double	All	The current theoretical total dollar value delivered on this hose, since it was first configured in the system.
MtrTheoVolume	Double	All	The current theoretical total volume delivered on this hose, since it was first configured in the system.
MtrElecValue	Double	All	The current electronic total dollar value delivered on this hose, as reported by the pump. This value is updated at the end of every delivery and when a pump starts responding.
MtrElecVolume	Double	All	The current electronic total volume delivered on this hose as reported by the pump. This value is updated at the end of every delivery and when a pump starts responding.
UVEAtenna	Int16	Ex	The RFiD antenna number associated with this hose.
Price1	Double	Ex2	Specific level 1 price for this hose, or zero if the associated grades price is to be used.
Price2	Double	Ex2	Specific level 2 price for this hose, or zero if the associated grades price is to be used.
Enabled	Int16	Ex2	Is the hose enabled yes/no

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
INVALID_OBJECT_PARAMETER_RESULT	One of the property values passed is out of range.
INVALID_OBJECT_LINK_RESULT	One of the IDs passed representing a related

	object references a non-existing object.

Remarks

This API is provided so that hose objects can be created and maintained on the EZServer. If the given hose ID already exists then the properties for this object will be overwritten with the values passed, otherwise a new hose object with this ID is created and its properties set to the values passed. It is the responsibility of the caller to ensure that the hose physical number is unique. The PumpID and TankID must represent objects which already exist on the server or the call will fail.

See also

GetHoseProperties(Ex, Ex2), DeleteHose

8.10.5. DeleteHose

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient
For DB clients only.

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the hose object to be deleted from the server.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
OBJECT_HAS_DEPENDANCIES_RESULT	This object cannot be deleted as there are other objects in the server which have references to it.

Remarks

This API is used to remove a hose object from the server. If this hose has deliveries linked to it, it cannot be deleted, delete the delivery objects first. If a hose ID of -1 is passed all the hoses will be deleted in a single operation.

See also

GetHoseProperties(Ex, Ex2)

8.10.6. GetHosePrices

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the hose object for which to get the prices.
DurationType	Int16*		The duration for this price see *Appendix 19 – Price Duration Type*
PriceType	Int16*		The type of price see *Appendix 18 – Price Type*
Price1	Double*		The price for level 1 or zero, to use the grade price.
Price2	Double*		The price for level 2 or zero, to use the grade price.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API can be used to get any temporary or permanent prices set on a specific hose.

See also

SetHosePrices

8.10.7. GetHoseSummary(Ex)

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32	All	The ID of the hose object for which the properties are being requested.
Number	Int32*	All	The logical number of the hose, this number is with respect to the pump, i.e. a number 1, means the first hose on this pump.
PhyscalNumber	Int32*	All	The physical number of the hose, this number is unique across all hoses and represents the public key for this object.
PumpID	Int32*	All	The ID of the pump to which this hose is connected.
PumpNumber	Int32*	All	The logical pump number of the fueling position for this hose.
PumpName	String*	All	The name of the fueling position for this hose.
TankID	Int32*	All	The ID of the tank for this hose.
TankNumber	Int32*	All	The number of the tank for this hose.
TankName	String*	All	The name of the tank for this hose.
GradeID	Int32*	All	The grades object identifier for the grade of this hose.
GradeNumber	Int32*	All	The number of the grade for this hose.
GradeName	String*	All	The full name of the grade for this hose.
GradeShortName	String*	All	The short name of the grade for this hose.
GradeCode	String*	All	The grade code for this hose.
MtrTheoValue	Double*	All	The current theoretical total dollar value delivered on this hose, since it was first configured in the system.
MtrTheoVolume	Double*	All	The current theoretical total volume delivered on this hose, since it was first configured in the system.
MtrElecValue	Double*	All	The current electronic total dollar value delivered on this hose, as reported by the pump. This value is updated at the end of every delivery and when a pump starts responding.
MtrElecVolume	Double*	All	The current electronic total volume delivered on this hose as reported by the pump. This value is updated at the end of every delivery and when a pump starts responding.
Price1	Double*	Ex	Specific level 1 price for this hose, or zero if the associated grades price is to be used.
Price2	Double*	Ex	Specific level 2 price for this hose, or zero if the associated

			grades price is to be used.
Enabled	Int16*	Ex	Is the hose enabled yes/no

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is functionally equivalent to GetHoseProperties(Ex, Ex2), except that it includes and expends on the pump, tank and grades linked to this hose.

See also

GetHoseProperties(Ex, Ex2), GetHosePrices

8.10.8. SetHoseETotals

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the hose object in question.
Volume	Double		The new volume for MtrElecVolume parameter.
Value	Double		The new value for MtrElecValue parameter.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

With some pumps types or where virtual e-totals are maintained in the server it is possible to write new e-totals back to the pump. Also in pump types where the e-totals coming from the pump are truncated at the 1 million, it is possible to write the above 1 million part of the e-totals to the server. From that point on it is maintained by the server. I.e. the part over one million is incremented when the e-totals coming from the pump wrap around.

See also

GetHoseProperties(Ex, Ex2)

8.10.9. SetHosePrices

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the hose object for which to set the prices.
DurationType	Int16		The duration for this price see *Appendix 19 – Price Duration Type*
PriceType	Int16		The type of price see *Appendix 18 – Price Type*
Price1	Double		The price for level 1 or zero to leave unchanged.
Price2	Double		The price for level 2 or zero to leave unchanged.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used to set a temporary or permanent price for a specific hose, this price will override the grade price associated with this hose. Setting it to zero will revert to the grade price.

See also

GetHosePrices

8.11. Grades

Grades are the various types of fuels that can be dispensed, they have no parent object and as such must be configured prior to configuring the tanks.

8.11.1. GetGradesCount

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Count	Int32*		The returned total number of grades configured in the EZServer.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is used to determine the total number of grades currently configured in the EZServer. Once this is known, the IDs of the individual hose objects can be obtained using GetGradeByOrdinal.

See also

GetGradeByNumber, GetGradeByName, GetGradeByOrdinal, GetGradeProperties

8.11.2. GetGradeByNumber

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Number	Int32		The number of the grade for which the ID is being requested.
ID	Int32*		The returned ID of the grade object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

Use this API to get the grade ID from the grade number.

See also

GetGradesCount, GetGradeByName, GetGradeByOrdinal, GetGradeProperties

8.11.3. GetGradeByName

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Name	String		The name of the grade object for which the ID is being requested.
ID	Int32*		The returned ID of the grade object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

Use this API to get the ID of the grade from the grade name.

See also

GetGradesCount, GetGradeByNumber, GetGradeByOrdinal, GetGradeProperties

8.11.4. GetGradeByOrdinal

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Index	Int32		The index of the desired grade object, this can be between 1 and the total number of grades, as returned by GetGradesCount.
ID	Int32*		The returned ID of the grade.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used to retrieve the grade ID of a grade using its ordinal value in the EZServer's internal grade objects list. The grades are ordered in this list by ID.

See also

GetGradesCount, GetGradeByNumber, GetGradeByName, GetGradeProperties

8.11.5. GetGradeProperties(Ex)

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32	All	The ID of the grade object for which the properties are being requested.
Number	Int32*	All	The grade number.
Name	String*	All	The grade name.
ShortName	String*	All	The grade short name.
Code	String*	All	The grade code.
Type	Int16*	Ex	The fuel type as defined in the EZATG.ini file.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API returns all the property values of a grade object for the given ID.

See also

SetGradeProperties

8.11.6. SetGradeProperties(Ex)

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient
For DB clients only.

Parameters

Parameter	Type	API	Description
ID	Int32	All	The ID of the grade object for which the properties are being requested.
Number	Int32	All	The grade number.
Name	String	All	The grade name.
ShortName	String	All	The grade short name.
Code	String	All	The grade code.
Type	Int16	Ex	The fuel type as defined in the EZATG.ini file.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
INVALID_OBJECT_PARAMETER_RESULT	One of the property values passed is out of range.

Remarks

This API is provided so that grade objects can be created and maintained on the server. If the given grade ID already exists then the properties for this object will be overwritten with the values passed, otherwise a new grade object with this ID is created and its properties set to the values passed.

See also

GetGradeProperties, DeleteGrade

8.11.7. DeleteGrade

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient
For DB clients only.

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the grade object to be deleted from the EZServer.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
OBJECT_HAS_DEPENDANCIES_RESULT	This object cannot be deleted as there are other objects in the server which have references to it.

Remarks

This API is used to remove a grade object from the EZServer. If this grade has tanks linked to it, it cannot be deleted, delete the tank objects first. This will also remove any prices associated with the grade.

See also

SetGradeProperties, SetGradePrice

8.11.8. SetGradePrice

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient
For DB clients only.

Parameters

Parameter	Type	API	Description
ID	Int32		The grade ID of the grade having its price set.
Level	Int16		The price level, must be between 1 and 8.
Price	Double		The price for the grade and price level.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
INVALID_OBJECT_PARAMETER_RESULT	One of the property values passed is out of range.

Remarks

This API is provided so that the, prices can be set/changed for each price level grade combination. To create the price level in the first place simply set a price for this level, if this grade did not have a price for this level, it will be created and added to the price list. To delete a price level, other than level 1, set the price to zero. Price level 1 cannot be deleted.

See also

GetGradePrice

8.11.9. GetGradePrice

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The grade ID of the grade having its price queried.
Level	Int16		The price level, must be between 1 and 8.
Price	Double*		The price for this grade ID and price level.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used to read the price for a specific grade and price level.

See also

SetGradePrice

8.12. Tanks

Tanks are connected to all of the hoses and contain a specific grade, they may have probes and sensors associated with them. Before configuring the tanks, the grades must be configured, and tanks must be configured prior to configuring any hoses.

8.12.1. GetTanksCount

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Count	Int32*		The returned total number of tanks configured in the EZServer.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is used to determine the total number of tanks currently configured in the EZServer. Once this is known, the IDs of the individual tank objects can be obtained using GetTankByOrdinal.

See also

GetTankByNumber, GetTankByName, GetTankByOrdinal, GetTankProperties

8.12.2. GetTankByNumber

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Number	Int32		The number of the grade for which the ID is being requested.
ID	Int32*		The returned ID of the tank object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

Use this API to get the tank ID from the tank number.

See also

GetTanksCount, GetTankByName, GetTankByOrdinal, GetTankProperties

8.12.3. GetTankByName

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Name	String		The name of the tank object for which the ID is being requested.
ID	Int32*		The returned ID of the tank object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

Use this API to get the ID of the tank from the tank name.

See also

GetTanksCount, GetTankByNumber, GetTankByOrdinal, GetTankProperties

8.12.4. GetTankByOrdinal

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Index	Int32		The index of the desired tank object, this can be between 1 and the total number of tanks, as returned by GetTanksCount.
ID	Int32*		The returned ID of the tank.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used to retrieve the tank ID of a tank using its ordinal value in the EZServer's internal tank objects list. The tanks are ordered in this list by ID.

See also

GetTanksCount, GetTankByNumber, GetTankByName, GetTankProperties

8.12.5. GetTankProperties(Ex)

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32	All	The ID of the tank object for which the properties are being requested.
Number	Int32*	All	The tank number.
Name	String*	All	The tank name.
GradeID	Int32*	All	The ID of the grade contained in this tank.
Type	Int16*	All	The type of the tank. See *Appendix 12 – Tank Types* for more information.
Capacity	Double*	All	The total capacity of the tank in liters or gallons.
Diameter	Double*	All	The Diameter of the tank in meters or feet.
TheoVolume	Double*	All	The theoretical volume of fuel currently in the tank as calculated by the EZServer.
GaugeVolume	Double*	All	The volume in the tank as returned by the tank gauge.
GaugeTCVolume	Double*	All	The temperature corrected volume in the tank as returned by the tank gauge.
GaugeUllage	Double*	All	The unused or available capacity in the tank as returned by the tank gauge.
GaugeTemperature	Double*	All	The temperature of the fuel contained in the tank as reported by the tank gauge.
GaugeLevel	Double*	All	The level of the fuel in the tank as reported by the tank gauge.
GaugeWaterVolume	Double*	All	The volume of water in the bottom of the tank as reported by the tank gauge.
GaugeWaterLevel	Double*	All	The level of the water in the bottom of the tank as reported by the tank gauge.
GaugeID	Int32*	All	The ID of the gauge which is monitoring this tank.
ProbeNo	Int16*	All	The number of the probe as configured in the tank gauge.
GaugeAlarmsMask	Int32*	Ex	A bit field representing the currently active alarms etc. see *Appendix 24 – Alarms Mask*

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API returns all the property values of a tank object for the given ID.

See also

SetTankProperties

8.12.6. SetTankProperties(Ex)

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient
For DB clients only.

Parameters

Parameter	Type	API	Description
ID	Int32	All	The ID of the tank object for which the properties are being set.
Number	Int32	All	The tank number.
Name	String	All	The tank name.
GradeID	Int32	All	The ID of the grade contained in this tank.
Type	Int16	All	The type of the tank. See *Appendix 12 – Tank Types* for more information.
Capacity	Double	All	The total capacity of the tank in liters or gallons.
Diameter	Double	All	The Diameter of the tank in meters or feet.
TheoVolume	Double	All	The theoretical volume of fuel currently in the tank as calculated by the EZServer.
GaugeVolume	Double	All	The volume in the tank as returned by the tank gauge.
GaugeTCVolume	Double	All	The temperature corrected volume in the tank as returned by the tank gauge.
GaugeUllage	Double	All	The unused or available capacity in the tank as returned by the tank gauge.
GaugeTemperature	Double	All	The temperature of the fuel contained in the tank as reported by the tank gauge.
GaugeLevel	Double	All	The level of the fuel in the tank as reported by the tank gauge.
GaugeWaterVolume	Double	All	The volume of water in the bottom of the tank as reported by the tank gauge.
GaugeWaterLevel	Double	All	The level of the water in the bottom of the tank as reported by the tank gauge.
GaugeID	Int32	All	The ID of the gauge which is monitoring this tank.
ProbeNo	Int16	All	The number of the probe as configured in the tank gauge.
GaugeAlarmsMask	Int32	Ex	A bit field representing the currently active alarms etc. see *Appendix 24 – Alarms Mask*

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact

	EZTech technical support for assistance.
INVALID_OBJECT_PARAMETER_RESULT	One of the property values passed is out of range.
INVALID_OBJECT_LINK_RESULT	One of the IDs passed representing a related object references a non-existing object.

Remarks

This API is provided so that tank objects can be created and maintained on the EZServer. If the given tank ID already exists then the properties for this object will be overwritten with the values passed, otherwise a new tank object with this ID is created and its properties set to the values passed. The GradeID must represent a grade which already exists on the server or the call will fail.

See also

GetTankProperties, DeleteTank

8.12.7. DeleteTank

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient
For DB clients only.

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the tank object to be deleted from the EZServer.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
OBJECT_HAS_DEPENDANCIES_RESULT	This object cannot be deleted as there are other objects in the server which have references to it.

Remarks

This API is used to remove a tank object from the EZServer. If this tank has hoses linked to it, it cannot be deleted, delete the hose objects first. If a tank ID of -1 is passed all the tanks will be deleted in a single operation.

See also

SetTankProperties

8.12.8. GetTankSummary(Ex)

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32	All	The ID of the tank object for which the properties are being requested.
Number	Int32*	All	The tank number.
Name	String*	All	The tank name.
GradeID	Int32*	All	The ID of the grade contained in this tank.
GradeNumber	Int32*	All	The number of the grade contained in this tank.
GradeName	String*	All	The name of the grade contained in this tank.
GradeShortName	String*	All	The short name of the grade contained in this tank.
GradeCode	String*	All	The code of the grade contained in this tank.
Type	Int16*	All	The type of the tank. See *Appendix 12 – Tank Types* for more information.
Capacity	Double*	All	The total capacity of the tank in liters or gallons.
Diameter	Double*	All	The Diameter of the tank in meters or feet.
TheoVolume	Double*	All	The theoretical volume of fuel currently in the tank as calculated by the EZServer.
GaugeVolume	Double*	All	The volume in the tank as returned by the tank gauge.
GaugeTCVolume	Double*	All	The temperature corrected volume in the tank as returned by the tank gauge.
GaugeUllage	Double*	All	The unused or available capacity in the tank as returned by the tank gauge.
GaugeTemperature	Double*	All	The temperature of the fuel contained in the tank as reported by the tank gauge.
GaugeLevel	Double*	All	The level of the fuel in the tank as reported by the tank gauge.
GaugeWaterVolume	Double*	All	The volume of water in the bottom of the tank as reported by the tank gauge.
GaugeWaterLevel	Double*	All	The level of the water in the bottom of the tank as reported by the tank gauge.
GaugeID	Int32*	All	The ID of the gauge which is monitoring this tank.
ProbeNo	Int16*	All	The number of the probe as configured in the tank gauge.
State	Int16*	Ex	The current state of this tank see *Appendix 23 – Tank State*
GaugeAlarmsMask	Int32*	Ex	A bit field representing the currently active alarms etc. see *Appendix 24 – Alarms Mask*

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.

SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is functionally equivalent to GetTankProperties(Ex) however the grade details are expanded out and included.

See also

GetTankProperties(Ex)

8.13. Ports

Ports are the physical connection via which devices are connected to the EZForecourt, be default there are up to 4 USB ports, one being internal with up to another three optional external ports, besides this there is up to 8 serial ports, 2 internal and an optional 6 external. These are typically used for communications with ATG, or serial emulation of other forecourt controllers. It is also possible to create Ethernet socket ports for ATGs.

8.13.1. GetPortsCount

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Count	Int32*		The returned total number of ports configured in the EZServer.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is used to determine the total number of ports currently configured in the EZServer. Once this is known, the IDs of the individual port objects can be obtained using GetPortByOrdinal.

See also

GetPortByNumber, GetPortByName, GetPortByOrdinal, GetPortProperties

8.13.2. GetPortByNumber

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Number	Int32		The number of the port for which the ID is being requested.
ID	Int32*		The returned ID of the port object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

Use this API to get the port ID from the port number.

See also

GetPortsCount, GetPortByName, GetPortByOrdinal, GetPortProperties

8.13.3. GetPortByName

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Name	String		The name of the port object for which the ID is being requested.
ID	Int32*		The returned ID of the port object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

Use this API to get the ID of the port from the port name.

See also

GetPortsCount, GetPortByNumber, GetPortByOrdinal, GetPortProperties

8.13.4. GetPortByOrdinal

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Index	Int32		The index of the desired port object, this can be between 1 and the total number of ports, as returned by GetPortsCount.
ID	Int32*		The returned ID of the port object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used to retrieve the ID of a port using its ordinal value in the EZServer’s internal port objects list. The ports are ordered in this list by ID.

See also

GetPortsCount, GetPortByNumber, GetPortByName, GetPortProperties

8.13.5. GetPortProperties

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the port object for which the properties are being requested.
Number	In32*		The port number.
Name	String*		The port name, this can also be the IP address for an Ethernet port.
ProtocolID	Int32*		The ID of the protocol that is being used for this port. The list of supported protocols is being updated continually. Consult EZTech for an up-to-date list.
DeviceType	Int16*		The device type for the devices connected to this port.
SerialNo	String*		The serial number of the EZForecourt module that is connected to this port. If this is a standard serial port then this property is ignored.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API returns all the property values of a port object for the given ID.

See also

SetPortProperties

8.13.6. SetPortProperties

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient
For DB clients only.

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the port object for which the properties are being requested.
Number	Int32		The port number.
Name	String		The port name, this can also be the IP address for an Ethernet port.
ProtocolID	Int32		The ID of the protocol that is being used for this port. The list of supported protocols is being updated continually. Consult EZTech for an up-to-date list.
DeviceType	Int16		The device type for the devices connected to this port. See *Appendix 11 – Device Types* for more information.
SerialNo	String		The serial number of the EZForecourt module that is connected to this port. If this is a standard serial port then this property is ignored.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
INVALID_OBJECT_PARAMETER_RESULT	One of the property values passed is out of range.
INVALID_OBJECT_LINK_RESULT	One of the IDs passed representing a related object references a non-existing object.

Remarks

This API is provided so that port objects can be created and maintained on the EZServer. If the given port ID already exists then the properties for this object will be overwritten with the values passed, otherwise a new port object with this ID is created and its properties set to the values passed. The ProtocolID must represent a protocol which is installed on the system or the call will fail.

See also

GetPortProperties, RemovePort

8.13.7. RemovePort

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient
For DB clients only. For EZTech.EZClient this API is called RemovePort.

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the port object to be deleted from the EZServer.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
OBJECT_HAS_DEPENDANCIES_RESULT	This object cannot be deleted as there are other objects in the server which have references to it.

Remarks

This API is used to remove a port object from the EZServer. If this port has devices linked to it, it cannot be deleted, delete the device objects first. If a port ID of -1 is passed all the ports will be deleted in a single operation.

See also

SetPortProperties

8.13.8. GetZB2GConfig

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the USB port in question, 1 for USB1.
PanID	Int64*		The PAN id for the network.
Channels	Int32*		A bit mask to determine which of the channels can be used. Only bits 11 thru 26 can be used, and maximum of 3 bits can be set.
KeyA	Int64*		The first half of the encryption key.
KeyB	Int64*		The second half of the encryption key.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is used to retrieve the ZigBee configuration from the EZForecourt.

See also

GetSerialNo, GetDeviceDetails

8.13.9. GetSerialNo

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the USB port in question 1 for USB1
SerialNo	String*		The return serial number string.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is used to retrieve the Serial number of the EZForecourt.

See also

GetDeviceDetails, ResetDevice, RequestVersion

8.13.10. GetDeviceDetails

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the USB port in question, 1 for USB1.
ZBID	Int32		The ZigBeeID of the device in question, it must be greater than 4.
SerialNo	String*		The returned serial number.
BootVersion	String*		The returned boot code version.
FirmwareVersion	String*		The returned firmware version.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is used to retrieve the Serial number and firmware of a responding EZRemote.

See also

GetSerialNo, ResetDevice, RequestVersion

8.13.11. ResetDevice

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the USB port in question, 1 for USB1.
ZBID	Int32		The ZigBeeID of the device in question, it must be greater than 4.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is used to request that a device performs a software reset.

See also

GetSerialNo, GetDeviceDetails, RequestVersion

8.13.12. RequestVersion

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the USB port in question, 1 for USB1.
ZBID	Int32		The ZigBeeID of the device in question, it must be greater than 4.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is used to initiate a version request from an EZRemote. It may take several seconds to retrieve the version, which can be retrieved with GetDeviceDetails.

See also

GetSerialNo, GetDeviceDetails, ResetDevice

8.14. Attendants

Pump attendants are commonly found in full service gas stations, the attendant objects provide us with the means to link a delivery to a pump attendant. This is useful for passing the responsibility of collecting the payment for these delivery to the attendant responsible. It can also be used for commission or performance monitoring. See *Appendix 7 – Pump authorization modes*.

8.14.1. GetAttendantsCount

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Count	Int32*		The returned total number of pump attendants configured in the EZServer.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is used to determine the total number of attendants currently configured in the EZServer. Once this is known, the IDs of the individual attendant objects can be obtained using GetAttendantByOrdinal.

See also

GetAttendantByOrdinal, GetAttendantByNumber, GetAttendantByName, GetAttendantProperties

8.14.2. GetAttendantByNumber

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Number	Int32		The number of the attendant for which the ID is being requested.
ID	Int32*		The returned ID of the attendant object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

Use this API to get the attendant ID from the attendant number.

See also

GetAttendantsCount, GetAttendantByName, GetAttendantByOrdinal, GetAttendantProperties

8.14.3. GetAttendantByName

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Name	String		The name of the attendant object for which the ID is being requested.
ID	Int32*		The returned ID of the attendant object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

Use this API used to get the ID of the attendant from the attendant name.

See also

GetAttendantsCount, GetAttendantByNumber, GetAttendantByOrdinal, GetAttendantProperties

8.14.4. GetAttendantByOrdinal

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Index	Int32		The index of the desired Attendant object, this can be between 1 and the total number of Attendants, as returned by GetAttendantsCount.
ID	Int32*		The returned ID of the attendant object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used to retrieve the ID of an attendant using its ordinal value in the EZServer's internal attendant objects list. The attendants are ordered in this list by ID.

See also

GetAttendantsCount, GetAttendantByNumber, GetAttendantByName, GetAttendantProperties

8.14.5. GetAttendantProperties(Ex)

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32	All	The ID of the attendant object for which the properties are being requested.
Number	Int32*	All	The attendant number.
Name	String*	All	The attendant name.
ShortName	String*	All	The attendant short name.
Password	String*	All	The attendant password used to logon to a pump.
Tag	String*	All	The RFiD tag value as programmed into the attendant card.
ShiftAStart	Int16*	Ex	The time that the first shift for this attendant begins, it is in minutes from midnight, for example, 00:00 am = 0, 12:00pm = 720, 23:59 = 1439.
ShiftAEnd	Int16*	Ex	The time that the first shift for this attendant ends, it is in minutes from midnight.
ShiftBStart	Int16*	Ex	The time that the second shift for this attendant begins, it is in minutes from midnight.
ShiftBEnd	Int16*	Ex	The time that the first shift for this attendant ends, it is in minutes from midnight.
Type	Int16*	Ex	The pump attendant type, see *Appendix 24 – Attendant Type*

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API returns all the property values of an attendant object for the given ID.

See also

SetAttendantProperties

8.14.6. SetAttendantProperties(Ex)

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient
For DB clients only.

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the attendant object for which the properties are being requested.
Number	Int32		The attendant number.
Name	Int32		The attendant name.
ShortName	Int32		The attendant short name.
Password	String		The attendant password used to logon to a pump.
Tag	String	Ex	The RFiD tag value as programmed into the attendant card.
ShiftAStart	Int16	Ex	The time that the first shift for this attendant begins, it is in minutes from midnight, for example, 00:00 am = 0, 12:00pm = 720, 23:59 = 1439.
ShiftAEnd	Int16	Ex	The time that the first shift for this attendant ends, it is in minutes from midnight.
ShiftBStart	Int16	Ex	The time that the second shift for this attendant begins, it is in minutes from midnight.
ShiftBEnd	Int16	Ex	The time that the first shift for this attendant ends, it is in minutes from midnight.
Type	Int16	Ex	The pump attendant type, see *Appendix 24 – Attendant Type*

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
INVALID_OBJECT_PARAMETER_RESULT	One of the property values passed is out of range.
TAG_ALREADY_IN_USE_ERROR_RESULT	The new tag value is already in use by another pump attendant or card client.

Remarks

This API is provided so that attendant objects can be created and maintained on the EZServer. If the given attendant ID already exists the properties for this object will be overwritten with the values passed, otherwise a new attendant object with this ID is created and its properties set to the values passed.

See also

GetAttendantProperties, SetAttendantProperties

8.14.7. DeleteAttendant

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient
For DB clients only.

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the attendant object to be deleted from the EZServer.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
OBJECT_HAS_DEPENDANCIES_RESULT	This object cannot be deleted as there are other objects in the server which have references to it.

Remarks

This API is used to remove an attendant object from the EZServer. If this attendant is currently logged onto one or more pumps, the call will fail.

See also

SetAttendantProperties

8.14.8. AttendantLogon

Availability – EZClient.DLL, EZClient.SO.1, Web Service, EZTech.EZClient and EZTech.EZPump

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the attendant logging on.
PumpID	Int32		The ID of the pump this attendant is logging onto, or NULL_ID. This parameter is not present for the EZTech.EZPump controls.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
PUMP_IN_WRONG_AUTH_MODE_RESULT	The pump that the attendant is attempting to logon to is not in one of the attendant authorization modes.
ATTENDANT_ALREADY_LOGGED_ON_RESULT	The pump the attendant is attempting to logon to, already has an attendant logged on to it.
PUMP_HAS_DELIVERIES_RESULT	The pump being logged on to has uncleared deliveries.
INVALID_LOGON_RESULT	This attendant is currently blocked.

Remarks

This API is used to log an attendant onto a pump. While the attendant remains logged onto this pump, all deliveries done on this pump will logged to this attendant. If the attendant is logged on without specifying a pump, this simply enables the card for the pump attendant.

See also

AttendantLogoff

8.14.9. AttendantLogoff

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the attendant logging off.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
ATTENDANT_NOT_LOGGED_ON_RESULT	The attendant being logged off, is not currently logged on.
PUMP_HAS_DELIVERIES_RESULT	The pump being logged off has uncleared deliveries.

Remarks

This API is used to log an attendant off a pump. This will log the attendant off all logged on pumps and close the open attendant report for this attendant.

See also

AttendantLogon, RemovePort, SetPortProperties

8.14.10. GetAttendantState

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the attendant object for which the state is being requested.
Type	Int16*		The type of pump attendant, see *Appendix 24 – Attendant Type*
LoggedOn	Int16*		A true or false value as to whether the pump attendant is currently logged on or not.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is provided to query the current logged on state of the given pump attendant.

See also

GetAttendantProperties(Ex)

8.15. Card Clients

The card client object is an object which defines and identifies a client for purchasing fuel. This could be a vehicle, motorist or company. Associated with the card client is an RFiD card that is used to identify the customer and optionally authorize a delivery. See *Appendix 7 – Pump authorization modes*.

8.15.1. GetCardClientsCount

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Count	Int32*		The returned total number of card clients configured in the EZServer.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is used determine the total number of card clients currently configured in the EZServer. Once this is known, the IDs of the individual card client objects can be obtained using GetCardClientByOrdinal.

See also

GetCardClientByOrdinal

8.15.2. GetCardClientByNumber

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Number	Int32		The number of the card client for which the ID is being requested.
ID	Int32*		The returned ID of the card client object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

Use this API to get the card client ID from the card client number.

See also

GetCardClientByName, GetCardClientByOrdinal, GetCardClientsCount

8.15.3. GetCardClientByName

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Name	String		The name of the card client object for which the ID is being requested.
ID	Int32*		The returned ID of the card client object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

Use this API to get the ID of the card client from the card client name.

See also

GetZigBeeByNumber, GetZigBeeByOrdinal, GetZigBeeCount

8.15.4. GetCardClientByOrdinal

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Index	Int32		The index of the desired card client object, this can be between 1 and the total number of card client, as returned by GetCardClientsCount.
ID	Int32*		The returned ID of the card client object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used to retrieve the ID of a card client using its ordinal value in the EZServer’s internal card client objects list. The card clients are ordered in this list by ID.

See also

GetCardClientsCount, GetCardClientByNumber, GetCardClientByName

8.15.5. GetCardClientProperties(Ex,Ex2)

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32	All	The ID of the card client being queried.
Number	Int32*	All	The card client number.
Name	String*	All	The card client name.
Tag	String*	All	The RFiD tag for this client as read form the RFiD card.
Enabled	Int16*	All	A true/false flag to enable/disable this client.
PriceLevel	Int16*	Ex,Ex2	The level of the price to be used for this card client, this must be one of the levels defined for all of the grades.
Plate	String*	Ex,Ex2	The vehicle number plate.
GradeType	Int16*	Ex2	The type of grade permitted for this card client, or zero for all, see *EZATG INI file* and GetGradeProperties(Ex)
CardType	Int16*	Ex2	The card type of this card client, see *Appendix 27 – Card Types*.
LimitType	Int16*	Ex2	The delivery limit type, see *Appendix 9 – Pump limit types*
Limit	Double*	Ex2	The delivery volume of value limit to be applied for this card, this depends on the LimitType field.
EntryType	Int16*	Ex2	The type of additional manual data entry required for this card, see *Appendix 28 – Entry Types*
ExpirationDate	Date*	Ex2	The expiration date for this card.
ParentID	long*	Ex2	If this is a secondary card type, this filed is the ID of the primary card to which this is linked.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API returns all the property values of a card client object for the given ID.

See also

SetCardClientProperties(Ex, Ex2), *EZATG INI file*, GetGradeProperties(Ex)

8.15.6. SetCardClientProperties(Ex, Ex2)

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient
For DB clients only.

Parameters

Parameter	Type	API	Description
ID	Int32	All	The ID of the card client being updated,
Number	Int32	All	The card client number.
Name	String	All	The card client name.
Tag	String	All	The RFiD tag for this client as read form the RFiD card.
Enabled	Int16	All	A true/false flag to enable/disable this client.
PriceLeve	Int16	Ex,Ex2	The level of the price to be used for this card client, this must be one of the levels defined for all of the grades.
Plate	String	Ex,Ex2	The vehicle number plate.
GradeType	Int16	Ex2	The type of grade permitted for this card client, or zero for all, see *EZATG INI file* and GetGradeProperties(Ex)
CardType	Int16	Ex2	The card type of this card client, see *Appendix 27 – Card Types*.
LimitType	Int16	Ex2	The delivery limit type, see *Appendix 9 – Pump limit types*
Limit	Double	Ex2	The delivery volume of value limit to be applied for this card, this depends on the LimitType field.
EntryType	Int16	Ex2	The type of additional manual data entry required for this card, see *Appendix 28 – Entry Types*
ExpirationDate	Date	Ex2	The expiration date for this card.
ParentID	Long	Ex2	If this is a secondary card type, this filed is the ID of the primary card to which this is linked.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
INVALID_OBJECT_PARAMETER_RESULT	One of the property values passed is out of range.
TAG_ALREADY_IN_USE_ERROR_RESULT	The new tag value is already in use by another pump attendant or card client.

Remarks

This API is provided so that card client objects can be created and maintained on the EZServer. If the given card client ID already exists the properties for this object will be overwritten with the values passed, otherwise a new card client object with this ID is created and its properties set to the values passed.

See also

GetCardClientProperties(Ex,Ex2), DeleteCardClient

8.15.7. DeleteCardClient

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the card client object to delete.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
OBJECT_HAS_DEPENDANCIES_RESULT	This object cannot be deleted as there are other objects in the server which have references to it.

Remarks

The ID of the card client object to be deleted from the EZServer.

See also

SetCardClientProperties(Ex, Ex2)

8.16. Card Reads

The card read object is a temporary object created in the EZServer whenever an RFiD card is read from one of the RFiD readers attached to the EZremotes or other devices. These objects remain in the system until they are deleted by DeleteCardRead or timeout after 60 seconds. These card reads can be received as events in GetNextCardReadEvent / CardReadEvent as events of type CARD_READ_EVENT, or polled for using the GetCardReadsCount API etc. When a card read is generated it is pre-consulted in the attendants and card clients lists to see if it is a configured card.

8.16.1. GetCardReadsCount

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Count	Int32*		The returned total number of card reads cached in the EZServer.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is used to determine of card reads currently waiting in the internal card reads cache in the EZServer.

See also

GetCardReadByOrdinal, GetCardReadProperties, DeleteCardRead

8.16.2. GetCardReadByNumber

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Number	Int32		The number of the card read in question.
ID	Int32*		The ID of the card read returned.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is provided convert the card read number into its corresponding card ID. The number parameter for a card read depends on the associated object, it will the pump attendant, card client number or 0.

See also

GetCardReadByOrdinal, GetCardReadByName

8.16.3. GetCardReadByOrdinal

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Index	Int32		The index of the desired card read object, this can be between 1 and the total number of card reads, as returned by GetCardReadsCount.
ID	Int32*		The returned ID of the card read object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used to retrieve the card read ID using its ordinal value in the EZServer's internal card reads objects list. The card reads are ordered in this list by ID.

See also

GetCardReadsCount, GetCardReadByNumber, GetCardReadByName

8.16.4. GetCardReadByName

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Name	String		The name of the card read object for which the ID is being requested.
ID	Int32*		The returned ID of the card read object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API returns the ID of a card read for a given name. The name parameter of a card is determined the parent object, either a pump attendant or card client name is required.

See also

GetCardReadByNumber, GetCardReadByOrdinal

8.16.5. GetCardReadProperties

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient
For DB clients only.

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the card read object being queried.
Number	Int32*		The number of the pump attendant or card client associated with this card read.
Name	String*		The name of the pump attendant or card client associated with this card read.
PumpID	Int32*		The ID of the fueling point, where this card was read.
Type	Int16*		The type of card read, see *Appendix 26 – Card Read Types*
ParentID	Int32*		The ID of the parent object of this card read, this will either be an Attendant or Card Client ID.
Tag	Int64*		The tag of the RFiD card as read from the card.
TimeStamp	DateTime*		The time and date that the card was read.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used to query all of the attributes of this card read. When a card read object is created the pump attendants and card client lists are consulted to see I this card is registered, if the number, name, parentID and type fields are populated.

See also

SetCardReadProperties

8.16.6. SetCardReadProperties

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the card read object being updated.
Number	Int32		The number of the pump attendant or card client associated with this card read.
Name	String		The name of the pump attendant or card client associated with this card read.
PumpID	Int32		The ID of the fueling point, where this card was read.
Type	Int16		The type of card read, see *Appendix 26 – Card Read Types*
ParentID	Int32		The ID of the parent object of this card read, this will either be an Attendant or Card Client ID.
Tag	Int64		The tag of the RFiD card as read from the card.
TimeStamp	DateTime		The time and date that the card was read.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
INVALID_OBJECT_PARAMETER_RESULT	One of the property values passed is out of range.

Remarks

This API can be used to create a new card read object in the internal list of the EZServer. This API is used internally when a card read is detected.

See also

GetCardReadProperties

8.16.7. DeleteCardRead

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of card read object to be deleted from the EZServer.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API removes the card read object form the internal list of the EZServer.

See also

SetCardReadProperties

8.16.8. GetCardType

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Tag	String		The RFiD card tag, to be consulted.
TagType	Int16*		The type of card tag see *Appendix 26 – Card Read Types*
ID	Int32*		The ID od the associated object, a pump attendant, card client or NULL_ID
Name	String*		The name of the associated object, either a pump attendant or card client.
Number	Int32*		The number of the associated object, either a pump attendant or card client.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is provided to consult the internal card client and pump attendants list for a specific RFiD card tag, if it is found, the card details etc. are returned.

See also

GetCardReadProperties, GetAttendantProperties(Ex), GetCardClientProperties(Ex,Ex2)

8.17. ZigBee devices

ZigBee device objects are devices connected to the EZserver via the ZigBee wireless network, at this stage they are limited to EZRemote devices. In order to connect a pump to an EZRemote a ZigBee device must be created and configured first.

8.17.1. GetZigBeeCount

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Count	Int32*		The returned total number of ZigBee devices configured in the EZServer.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is used determine the total number of ZigBee devices currently configured in the EZServer. Once this is known, the IDs of the individual ZigBee devices can be obtained using GetZigBeeByOrdinal.

See also

GetZigBeeByOrdinal, GetZigBeeByNumber, GetZigBeeByName

8.17.2. GetZigBeeByNumber

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Number	Int32		The number of the ZigBee device for which the ID is being requested.
ID	Int32*		The returned ID of the ZigBee device object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

Use this API to get the ZigBee device ID from the ZigBee device number.

See also

GetZigBeeByName, GetZigBeeByOrdinal, GetZigBeeCount

8.17.3. GetZigBeeByName

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Name	String		The name of the ZigBee device for which the ID is being requested.
ID	Int32*		The returned ID of the ZigBee device object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

Use this API to get the ID of the ZigBee device from the ZigBee device name.

See also

GetZigBeeByNumber, GetZigBeeByOrdinal, GetZigBeeCount

8.17.4. GetZigBeeByOrdinal

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Index	Int32		The index of the desired ZigBee device, this can be between 1 and the total number of ZigBee devices, as returned by GetZigBeeCount.
ID	Int32*		The returned ID of the ZigBee device object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used to retrieve the ID of a ZigBee device using its ordinal value in the EZServer's internal ZigBee device objects list. The ZigBee device are ordered in this list by ID.

See also

GetZigBeeByNumber, GetZigBeeByName, GetZigBeeCount

8.17.5. GetZigBeeProperties

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the ZigBee device being queried.
Number	Int32*		The number of the ZigBee device.
Name	String*		The name of the ZigBee device.
DeviceType	Int16*		The type of ZigBee device see *Appendix 16 – Remote device type*
SerialNumber	String*		The serial number of this device in the format nnnn/nn.
NodeIdentifier	String*		Reserved for CTF.
PortID	Int32*		The ID of the port that is associated with the ZigBee network, usually 1 for USB1.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API returns all the property values of a ZigBee device object for a given ID.

See also

SetZigBeeProperties

8.17.6. SetZigBeeProperties

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient
For DB clients only.

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the ZigBee device being updated.
Number	Int32		The number of the ZigBee device.
Name	String		The name of the ZigBee device.
DeviceType	Int16		The type of ZigBee device see *Appendix 16 – Remote device type*
SerialNumber	String		The serial number of this device in the format nnnn/nn.
NodeIdentifier	String		Reserved for CTF.
PortID	Int32		The ID of the port that is associated with the ZigBee network, usually 1 for USB1.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
INVALID_OBJECT_PARAMETER_RESULT	One of the property values passed is out of range.

Remarks

This API is provided so that ZigBee device objects can be created and maintained on the EZServer. If the given ZigBee device ID already exists the properties for this object will be overwritten with the values passed, otherwise a new ZigBee device object with this ID is created and its properties set to the values passed.

See also

GetZigBeeProperties

8.17.7. DeleteZigBee

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the ZigBee device object to delete.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
OBJECT_HAS_DEPENDANCIES_RESULT	This object cannot be deleted as there are other objects in the server which have references to it.

Remarks

This API is used to delete a ZigBee device object from the EZServer.

See also

SetZigBeeProperties

8.18. Sensors

Sensors are a device for detecting the presence of liquid, they are primarily used around fuel tanks to detect leaks. They are usually connected to the ATG and will generate an alarm when the presence of liquid is detected. There is no specific event directly associated with sensors, however when triggered they will generate a LogEvent alarm.

8.18.1. GetSensorsCount

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Count	Int32*		The returned total number of sensors configured in the EZServer.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is used to determine the total number of sensors currently configured in the EZServer. Once this is known, the IDs of the individual sensors can be obtained using GetSensorByOrdinal.

See also

GetSensorByOrdinal, GetSensorProperties

8.18.2. GetSensorByNumber

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Number	Int32		The number of the sensor for which the ID is being requested.
ID	Int32*		The returned ID of the sensor object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

Use this API to get the sensor ID from the sensor number.

See also

GetSensorByName, GetSensorByOrdinal, GetSensorsCount

8.18.3. GetSensorByName

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Name	String		The name of the sensor for which the ID is being requested.
ID	Int32*		The returned ID of the sensor object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

Use this API to get the ID of the sensor from the sensor name.

See also

GetSensorByNumber, GetSensorByOrdinal, GetSensorsCount

8.18.4. GetSensorByOrdinal

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Index	Int32		The index of the desired sensor, this can be between 1 and the total number of sensors, as returned by GetSensorsCount.
ID	Int32*		The returned ID of the sensor object.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used to retrieve the ID of a sensor using its ordinal value in the EZServer's internal sensor objects list. The sensors are ordered in this list by ID.

See also

GetSensorsCount, GetSensorByNumber, GetSensorByName

8.18.5. GetSensorProperties

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the sensor being queried.
Number	Int32*		The number of the sensor.
Name	String*		The name of the sensor.
PortID	Int32*		The ID of the port where the ATG is connected, see *Ports*
Type	Int16*		The sensor type, must be 1, reserved for future use.
Address	Int16*		The address of the sensor control card, usually 1.
SensorNo	Int16*		The sensor number on this control card.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API returns all the property values of a sensor object for a given ID.

See also

SetSensorProperties, GetSensorStatus

8.18.6. SetSensorProperties

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient
For DB clients only.

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the sensor being updated.
Number	Int32		The number of the sensor.
Name	String		The name of the sensor.
PortID	Int32		The ID of the port where the ATG is connected, see *Ports*
Type	Int16		The sensor type, must be 1, reserved for future use.
Address	Int16		The address of the sensor control card, usually 1.
SensorNo	Int16		The sensor number on this control card.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
INVALID_OBJECT_PARAMETER_RESULT	One of the property values passed is out of range.

Remarks

This API is provided so that sensor objects can be created and maintained on the EZServer. If the given sensor ID already exists the properties for this object will be overwritten with the values passed, otherwise a new sensor object with this ID is created and its properties set to the values passed.

See also

GetSensorProperties

8.18.7. GetSensorStatus

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the sensor being queried.
State	Int16*		A true/false flag for the state of the sensor.
Responding	Int16*		A true/false flag to indicate if the sensor is responding or not.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used to query the status of a specific sensor.

See also

SetSensorStatus, GetSensorProperties

8.18.8. SetSensorStatus

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the sensor being updated.
State	Int16		A true/false flag for the state of the sensor.
Responding	Int16		A true/false flag to indicate if the sensor is responding or not.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

The API is used to update the state of a sensor, this API is called internally in the EZServer and should not be called by a client application.

See also

GetSensorStatus

8.18.9. DeleteSensor

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the sensor object to delete.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_HAS_DEPENDANCIES_RESULT	This object cannot be deleted as there are other objects in the server which have references to it.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used to delete a sensor object from the EZServer.

See also

SetSensorProperties

8.19. Logged events

The Event Log is a list of events that are recorded to keep track of exception or normal events. These could alarms, warnings or simply information events regarding the forecourt controller or connected devices.

8.19.1. GetLogEventCount

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Count	Int32*		The returned total number of sensors configured in the EZServer that form to the filter passed in following parameters.
DeviceType	Int16		The device type that we are interested in or -1 for all device types. See *Appendix 20 – Log Event Device Type*
DiviceID	Int32		The Device ID that we are interested in, or NULL_ID for all devices. To specify a specific ID you must also specify the Device type.
EventLevel	Int16		The event level that we are interested in or -1 for all levels, see *Appendix 21 – Log Event Level*
EventType	Int16		The event type that we are interested in or -1 for all types, *Appendix 22 – Log Event Type*
ClearedBy	Int32		The client ID that cleared the events, or -2 for all events, passing NULL_ID will return all the uncleared events.
AckedBy	Int32		The client ID that acknowledged the events, or -2 for all events, passing NULL_ID will return all of the unacknowledged events.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.

Remarks

This API is used to determine the total number of logged events currently in the EZServer. Once this is known, the IDs of the individual events can be obtained using GetLogEventByOrdinal. The parameters to this API provide a way of filtering the entire events log to obtain only those events of interest.

See also

GetLogEventByOrdinal

8.19.2. GetLogEventByOrdinal

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
Index	Int32		The index of the desired log event, this can be between 1 and the total number of log events, as returned by GetLogEventByOrdinal.
ID	Int32*		The returned ID of the log event object.
DeviceType	Int16		The device type that we are interested in or -1 for all device types. See *Appendix 20 – Log Event Device Type*
DeviceID	Int32		The Device ID that we are interested in, or NULL_ID for all devices. To specify a specific ID you must also specify the Device type.
EventLevel	Int16		The event level that we are interested in or -1 for all levels, see *Appendix 21 – Log Event Level*
EventType	Int16		The event type that we are interested in or -1 for all types, *Appendix 22 – Log Event Type*
ClearedBy	Int32		The client ID that cleared the events, or -2 for all events, passing NULL_ID will return all the uncleared events.
AckedBy	Int32		The client ID that acknowledged the events, or -2 for all events, passing NULL_ID will return all of the unacknowledged events.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used to retrieve the ID of a log event using its ordinal value in the EZServer's internal filtered log event objects list. The log events are ordered in this list by ID. The filter parameters passed to this API must be the same as those passed to GetLogEventCount.

See also

GetLogEventCount, GetLogEventProperies

8.19.3. GetLogEventProperies

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		This ID of the Log Event, this is generated by the EZServer when the event was created.
DeviceType	Int16*		The type of the device that generated the event see *Appendix 20 – Log Event Device Type*
DeviceID	Int32*		The ID of the device that generated the event
DeviceNumber	Int32*		The number of the device that generated the event
DeviceName	String*		The name of the device that generated the event
EventLevel	Int16*		The level of the event see *Appendix 21 – Log Event Level*
EventType	Int16*		The type of the event see *Appendix 22 – Log Event Type*
EventDesc	String*		The event description.
GeneratedDT	DateTime*		The time and date the event was generated.
ClearedDT	Int32*		The time and date the event was cleared.
ClearedBy	Int32*		The client ID that cleared the event see ClearLogEvent
AckedBy	Int32*		The client ID that acknowledged the event see AckLogEvent
Volume	Double*		A volume associated with this event, depends on the event type.
Value	Double*		A value associated with this event, depends on the event type.
ProductVolume	Double*		The volume of product in the tank if this is a tank related event.
ProductLevel	Double*		The level of product in the tank if this is a tank related event.
WaterLevel	Double*		The water level in the tank if this is a tank related event.
Temperature	Double*		The temperature in the tank if this is a tank related event.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API returns all the property values of a log event object for a given ID.

See also

SetLogEventProperties

8.19.4. SetLogEventProperties

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient
For DB clients only.

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the Log Event being updated.
DeviceType	Int16		The type of the device that generated the event see *Appendix 20 – Log Event Device Type*
DeviceID	Int32		The ID of the device that generated the event
DeviceNumber	Int32		The number of the device that generated the event
DeviceName	String		The name of the device that generated the event
EventLevel	Int16		The level of the event see *Appendix 21 – Log Event Level*
EventType	Int16		The type of the event see *Appendix 22 – Log Event Type*
EventDesc	String		The event description.
GeneratedDT	DateTime		The time and date the event was generated.
ClearedDT	DateTime		The time and date the event was cleared.
ClearedBy	Int32		The client ID that cleared the event see ClearLogEvent
AckedBy	Int32		The client ID that acknowledged the event see AckLogEvent
Volume	Double		A volume associated with this event, depends on the event type.
Value	Double		A value associated with this event, depends on the event type.
ProductVolume	Double		The volume of product in the tank if this is a tank related event.
ProductLevel	Double		The level of product in the tank if this is a tank related event.
WaterLevel	Double		The water level in the tank if this is a tank related event.
Temperature	Double		The temperature in the tank if this is a tank related event.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
INVALID_OBJECT_PARAMETER_RESULT	One of the property values passed is out of range.

Remarks

Log events are created internally by the EZServer, this API is used internally and should not be called via an external client.

See also

GetLogEventProperies

8.19.5. DeleteLogEvent

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the log event object to delete.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_HAS_DEPENDANCIES_RESULT	This object cannot be deleted as there are other objects in the server which have references to it.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.

Remarks

This API is used to delete a log event object from the EZServer. Log event management and deletion etc. is managed by the EZServer, as such it is not necessary to all this API from an external client.

See also

SetLogEventProperties

8.19.6. ClearLogEvent

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the log event object to cleared.
ClientID	Int32		The ID of the client clearing the log event.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
LOG_EVENT_ALREADY_ACKED_RESULT	The log event object has already been cleared.

Remarks

Log events of a warning or alarm level, must be acknowledged and cleared by a user, active alarms will cause the internal beep to beep continuously. Clearing the alarm stops the beeper. Some alarms will auto clear when the condition is no longer critical.

See also

AckLogEvent

8.19.7. AckLogEvent

Availability – EZClient.DLL, EZClient.SO.1, Web Service and EZTech.EZClient

Parameters

Parameter	Type	API	Description
ID	Int32		The ID of the log event object to acknowledged.
ClientID	Int32		The ID of the client acknowledging the log event.

Return value

Error code	Error description
OK_RESULT	The call was successful.
NOT_LOGGED_ON_RESULT	The client is not currently logged on.
INVALID_CLIENT_TYPE	The currently logged on client type, does not permit this call.
SERVER_TIMEOUT	The call to the server timed out.
CONNECTION_BROKEN	The connection with the server was lost.
INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
LOG_EVENT_ALREADY_ACKED_RESULT	The log event object has already been cleared.

Remarks

This API is used to acknowledge the log event. The acknowledged functionality is provided so that a third party system can upload the log events to a data base or similar. The log events will be retained in the EZServer until they are acknowledged. For this to work the ConfirmEventLog flag in the *EZATG INI file* must be set to Yes, otherwise the events will be auto acknowledged when they are created.

See also

ClearLogEvent

9. Appendices

9.1. Appendix 1 – Pump states

Value	Name	Description
0	INVALID_PUMP_STATE	The EZTech.EZPump control has lost it connection with the server.
1	NOT_INSTALLED_PUMP_STATE	The pump not installed, it has most likely been deleted.
2	NOT_RESPONDING_1_PUMP_STATE	The pump is installed but is not responding to the server or to the EZRemote.
3	IDLE_PUMP_STATE	The pump is responding and is idle, all hoses are stowed, if a hose is pulled it will call for authorization, i.e. it will not start delivering without manual authorization.
4	PRICE_CHANGE_STATE	The pump is currently performing a price change. This state only exists for certain pump types.
5	AUTHED_PUMP_STATE	The pump is idle with all of its hoses stowed. If a hose is pulled it will start delivering automatically.
6	CALLING_PUMP_STATE	A hose was pulled while the pump was in the IDLE_PUMP_STATE, it will not start delivering until it receives a manual authorization.
7	DELIVERY_STARTING_PUMP_STATE	A hose has been pulled and the pump is authorized, however the volume in progress total is still zero.
8	DELIVERING_PUMP_STATE	A hose is pulled and the volume in progress is not zero.
9	TEMP_STOPPED_PUMP_STATE	The pump has been temp stopped while in one of the delivering states. To get it out of this state the pump will require re-authorizing or the pump delivery will need to be terminated.
10	DELIVERY_FINISHING_PUMP_STATE	The pump is delivering; however the volume in progress total has not changed in the last 5 seconds.
11	DELIVERY_FINISHED_PUMP_STATE	A hose has been returned after a delivering some fuel. The state is temporary and only exists while the server retrieves the delivery totals and pump electronic totals from the pump.
12	DELIVERY_TIMEOUT_PUMP_STATE	The pump has been in one of the delivering states for more than the ‘Delivering Timeout’ as configured in the EZServer.ini file.
13	HOSE_OUT_PUMP_STATE	The pump has been in the DELIVERY_FINISHING_PUMP_STATE for more than the ‘HoseOutTimeout’ as configured in the EZServer.ini file.
14	PREPAY_REFUND_TIMEOUT_STATE	A prepay delivery has finished and a refund was generated. The pump then remains locked out for

		'PrepayRefundTimeout' seconds as configured in the EZServer.ini file.
15	DELIVERY_TERMINATED_STATE	A delivery has been terminated on this pump and the hose has not been returned yet. See
16	ERROR_PUMP_STATE	The pump has entered an error state; This is pump type specific.
17	NOT_RESPONDING_2_PUMP_STATE	The pumps is installed via an EZRemote, and the EZRemote is not responding to the server.

9.2. Appendix 2 – Pump reserves types

Value	Name	Description
1	NOT_RESERVED	There are no reserves placed on the pump.
2	RESERVED_FOR_PREPAY	The pump has been reserved for a prepay delivery and is waiting for the prepay authorization.
3	AUTHED_FOR_PREPAY	The pump has been authorized for a prepay delivery and is waiting for the client to complete the delivery.
4	RESERVED_FOR_PREAUTH	The pump has been reserved for a preauth delivery and is waiting for the preauth authorization
5	AUTHED_FOR_PREAUTH	The pump has been authorized for a preauth delivery and is waiting for the client to complete the delivery.
6	RESERVED_FOR_CTF	Reserved for CTFs use.
7	AUTHED_FOR_CTF	Reserved for CTFs use.
8	RESERVED_FOR_PAYMENT	The pump has been reserved for a payment delivery and is waiting for the payment authorization.
9	AUTHED_FOR_PAYMENT	The pump has been authorized for a payment delivery and is waiting for the client to complete the delivery.

9.3. Appendix 3 – Delivery types

Value	Name	Description
1	POSTPAY	A normal delivery which was delivered and paid for after it was completed prior to clearing it from the system
2	PREPAY	A delivery which was paid for prior to delivering.
3	PREPAY_REFUND	A refund delivery generated when a prepay delivery is taken for less than the prepaid amount.
4	PREAUTH	A pre-authorized delivery that was paid for after it was completed and before it was cleared from the system.
5	MONITOR	A normal delivery that was cleared automatically from the system after the MonitorDeliveryTimeout value was reached, as configured in the EZServier.ini file.
6	TEST	A normal delivery that was returned to the tank after completion. This is usually occurs when the pump is being calibrated or tested.
7	DRIVEOFF	A normal delivery for which the driver has driven off without paying. In order to balance the shift totals it cannot be cleared as a Post-pay delivery.
8	OFFLINE	A pseudo delivery generated when the server detects that the pump has done deliveries while offline, this is detected by comparing the saved electronic totals last retrieved from the pump, with the current electronic totals. It could well be more than one delivery summed together.
9	CTF	Reserved for CTFs use.
10	CARD_CLIENT	The delivery was authorized using a clients RFiD card.
11	PAYMENT	The delivery was authorized using the Payment APIs

9.4. Appendix 4 – Delivery states

Value	Name	Description
1	CURRENT	The current delivery on the pump, i.e. the delivery displayed on the pump is guaranteed to be the same as this delivery. This pump will not delivery again until this delivery is cleared or stacked
2	STACKED	This is a stacked delivery; the pump is now free to deliver again.
3	CLEARED	The delivery has been paid for and is cleared from the system.

9.5. Appendix 5 – Event types

Event types are maintained for backward compatibility, for new versions use *Appendix 22 – Log Event Type*

Value	Name	Description
100	SERVER_INFO_EVENT	
102	SERVER_STARTED_EVENT	
103	SERVER_STOPPED_EVENT	
200	DISPENSER_INFO_EVENT	
201	DISPENSER_INSTALLED_EVENT	
202	DISPENSER_STARTED_EVENT	
203	DISPENSER_STOPPED_EVENT	
204	DISPENSER_E_TOTALS_ERROR_EVENT	
205	DISPENSER_OFFLINE_DELIVERIES_EVENT	
206	DISPENSER_REMOTE_PRICE_CHANGE_EVENT	
207	DISPENSER_LOST_DELIVERY_EVENT	
208	DISPENSER_TAG_READ_EVENT	
209	DISPENSER_ATTENDANT_TAG_READ_EVENT	
300	DISPENSER_DRIVER_INFO_EVENT	
301	DISPENSER_DRIVER_STARTED_EVENT	
302	DISPENSER_DRIVER_STOPPED_EVENT	
400	TANK_INFO_EVENT	
401	TANK_SETUP_DATA_WARNING_EVENT	
402	TANK_LEAK_ALARM_EVENT	
403	TANK_HIGH_WATER_ALARM_EVENT	
404	TANK_HIGH_WATER_ALARM_EVENT	
405	TANK_OVERFILL_ALARM_EVENT	
406	TANK_LOW_PRODUCT_ALARM_EVENT	
407	TANK_SUDDEN_LOSS_ALARM_EVENT	
408	TANK_HIGH_PRODUCT_ALARM_EVENT	
409	TANK_INVALID_FUEL_LEVEL_ALARM_EVENT	
410	TANK_PROBE_OUT_ALARM_EVENT	
411	TANK_HIGH_WATER_WARNING_EVENT	
412	TANK_DELIVERY_NEEDED_WARNING_EVENT	
413	TANK_MAXIMUM_PRODUCT_ALARM_EVENT	
414	TANK_GROSS_LEAK_TEST_FAILED_ALARM_EVENT	
415	TANK_PERIODIC_LEAK_TEST_FAILED_ALARM_EVENT	
416	TANK_ANNUAL_LEAK_TEST_FAILED_ALARM_EVENT	
417	TANK_PERIODIC_TEST_NEEDED_WARNING_EVENT	
418	TANK_ANNUAL_TEST_NEEDED_WARNING_EVENT	
419	TANK_PERIODIC_TEST_NEEDED_ALARM_EVENT	
420	TANK_ANNUAL_TEST_NEEDED_ALARM_EVENT	
421	TANK_LEAK_TEST_ACTIVE_INFO_EVENT	
422	TANK_NO_CSLD_IDLE_TIME_WARNING_EVENT	
423	TANK_SIPHON_BREAK_ACTIVE_WARNING_EVENT	
424	TANK_CSLD_RATE_INCREASE_WARNING_EVENT	
425	TANK_ACCUCHART_CALIBARTION_WARNING_EVENT	

426	TANK_HRM_RECONCILATION_WARNING_EVENT	
427	TANK_HRM_RECONCILATION_ALARM_EVENT	
428	TANK_COLD_TEMPERATURE_WARNING_EVENT	
429	TANK_MISSING_DELIVERY_TICKET_WARNING_EVENT	
430	TANK_LINE_GROSS_LEAK_ALARM_EVENT	
431	TANK_SENSOR_STATE_ALARM_EVENT	
500	TANK_GAUGE_INFO_EVENT	
501	TANK_GAUGE_INSTALLED_EVENT	
502	TANK_GAUGE_STARTED_EVENT	
503	TANK_GAUGE_STOPPED_EVENT	
600	TANK_GAUGE_DRIVER_INFO_EVENT	
601	TANK_GAUGE_DRIVER_STARTED_EVENT	
602	TANK_GAUGE_DRIVER_STOPPED_EVENT	
700	PRICE_SIGN_INFO_EVENT	
701	PRICE_SIGN_INSTALLED_EVENT	
702	PRICE_SIGN_STARTED_EVENT	
703	PRICE_SIGN_STOPPED_EVENT	
800	PRICE_SIGN_DRIVER_INFO_EVENT	
801	PRICE_SIGN_DRIVER_STARTED_EVENT	
802	PRICE_SIGN_DRIVER_STOPPED_EVENT	

9.6. Appendix 6 – Pump display formats

Value	Name	Description
1	PUMP_DISPLAY_4_3	9.999
2	PUMP_DISPLAY_4_2	99.99
3	PUMP_DISPLAY_4_1	999.9
4	PUMP_DISPLAY_4_0	9999
5	PUMP_DISPLAY_5_3	99.999
6	PUMP_DISPLAY_5_2	999.99
7	PUMP_DISPLAY_5_1	9999.9
8	PUMP_DISPLAY_5_0	99999
9	PUMP_DISPLAY_6_3	999.999
10	PUMP_DISPLAY_6_2	9999.99
11	PUMP_DISPLAY_6_1	99999.9
12	PUMP_DISPLAY_6_0	999999
13	PUMP_DISPLAY_4_N1	99990 one invisible significant digit.
14	PUMP_DISPLAY_4_N2	999900 two invisible significant digits.
15	PUMP_DISPLAY_4_N3	9999000 three invisible significant digits.
16	PUMP_DISPLAY_5_N1	999990 one invisible significant digit.
17	PUMP_DISPLAY_5_N2	9999900 two invisible significant digits.
18	PUMP_DISPLAY_5_N3	99999000 three invisible significant digits.
19	PUMP_DISPLAY_6_N1	9999990 one invisible significant digit.
20	PUMP_DISPLAY_6_N2	99999900 two invisible significant digits.
21	PUMP_DISPLAY_6_N3	999999000 three invisible significant digits.

9.7. Appendix 7 – Pump authorization modes

Value	Name	Description
1	NOT_AUTHABLE	The pump cannot be authorized for post pay deliveries, i.e. the pump can only be used for Prepay or Preauth deliveries.
2	COMP_AUTH	The pump must be manually authorized before it will start to deliver.
3	AUTO_AUTH	The pump will automatically authorize when the hose is pulled, however there must be a no current delivery or the stack must be set to auto stacking with space available on the stack.
4	MONITOR_AUTH	The pump will automatically authorize always, and completed deliveries will auto clear after a configured amount of time.
5	ATTENDANT_AUTH	This is the same as auto authorize with the added requirement that an attendant must be logged on to the pump.
6	ATTENDANT_MONITOR_AUTH	This is the same as monitor authorize with the added requirement that an attendant must be logged on to the pump.
7	CTF_AUTH	Reserved for CTFs use.
8	TAG_AUTH	The pump must be authorized by the TagAuthorise API.
9	OFFLINE_AUTH	Permits the EZRemotes or EZMOD authorize the pump without the EZserver.
10	ATTENDANT_TAG_AUTH	A valid pump attendant’s card must be read at the pump to authorize it.
11	CLIENT_TAG_AUTH	A valid card client’s card must be read at the pump to authorize it.
12	ATTENDANT_AND_CLIENT_TAG_AUTH	A valid card client’s and pump attendant’s card must be read at the pump to authorize the delivery.
13	ATTENDANT_OR_CLIENT_TAG_AUTH	A valid card client’s or pump attendant’s card must be read at the pump to authorize the delivery.
14	EXT_AUTH	The pump must be authorized using the PaymentAuthorise API.

9.8. Appendix 8 – Pump delivery stack (memory) modes

Value	Name	Description
1	STACK_DISABLED	The pump delivery stack is disabled.
2	STACK_MANUAL	Pump deliveries must be shifted onto the stack manually.
3	STACK_AUTO	Pump deliveries will automatically be shifted on the stack when a pump starts calling, if there is space available on the stack.

9.9. Appendix 9 – Pump limit types

Value	Name	Description
1	NO_PRESET_TYPE	There is not limit placed on the pump.
2	DOLLAR_PRESET_TYPE	There a preset value limit placed on the pump.
3	VOLUME_PRESET_TYPE	There a preset volume limit placed on the pump.
4	DOLLAR_PREPAY_TYPE	There a prepay value limit placed on the pump.
5	VOLUME_PREPAY_TYPE	There a prepay volume limit placed on the pump.
6	DOLLAR_PREAUTH_TYPE	There a preauth value limit placed on the pump.
7	VOLUME_PREAUTH_TYPE	There a preauth volume limit placed on the pump.

9.10. Appendix 10 – Permitted hoses mask

Value	Name	Description
0	No hoses	No permitted hoses.
1	Hose 1	Logical hose number 1 permitted.
2	Hose 2	Logical hose number 2 permitted.
4	Hose 3	Logical hose number 3 permitted.
8	Hose 4	Logical hose number 4 permitted.
16	Hose 5	Logical hose number 5 permitted.
32	Hose 6	Logical hose number 6 permitted.
64	Hose 7	Logical hose number 7 permitted.
128	Hose 8	Logical hose number 8 permitted.

To permit more than one hose simply add desired hoses together, for example 3 would permit hoses 1 and 2.

9.11. Appendix 11 – Device Types

Value	Name	Description
1	DISPENSER_DEVICE	A fuel dispenser or pump device.
2	DISPENSER_DRIVER_DEVICE	A software driver module for a fuel dispenser or pump device.
3	TANK_GAUGE_DEVICE	An electronic tank gauge device.
4	TANK_GAUGE_DRIVER_DEVICE	A software driver module for an electronic tank gauge device.
5	PRICE_SIGN_DEVICE	An electronic price sign device.
6	PRICE_SIGN_DRIVER_DEVICE	A software driver module for an electronic price sign device.

9.12. Appendix 12 – Tank Types

Value	Name	Description
1	MANUAL_DIP_TANK_TYPE	The tank is not gauged and has it level/volume read manually.
2	GAUGED_TANK_TYPE	The tank has a tank gauge probe fitted, and the volume and level are read automatically by a tank gauge.
3	CALCULATE_TANK_TYPE	The volume is calculated from the level as if the tank where a perfect horizontal cylinder.
4	CALIBRATE_TANK_TYPE	The volume is calculated from the level using a strapping table, the strapping is currently being calibrated against the deliveries from this tank.
5	RECONCILE_TANK_TYPE	The volume is calculated from the level using a strapping table, this strapping table is constantly being checked and verified against the deliveries from tank.

9.13. Appendix 13 – Error messages

Value	Name	Description
0	OK_RESULT	The call was successful
-1	INVALID_HEADER_VERSION	An internal error has occurred contact EZTech technical support for assistance.
-2	INVALID_INTERFACE_VERSION	An internal error has occurred contact EZTech technical support for assistance.
-3	INVLAID_INTERFACE_ID	An internal error has occurred contact EZTech technical support for assistance.
-4	INVALID_FUNCTION_ID	An internal error has occurred contact EZTech technical support for assistance.
-5	INVALID_SOURCE_ID	An internal error has occurred contact EZTech technical support for assistance.
-6	INVALID_DESTINATION_ID	An internal error has occurred contact EZTech technical support for assistance.
-7	INVALID_OBJECT_ID	An internal error has occurred contact EZTech technical support for assistance.
-8	INVALID_SEQUENCE_NO	An internal error has occurred contact EZTech technical support for assistance.
-9	MSG_IN_BUFFER_OVERFLOW	An internal error has occurred contact EZTech technical support for assistance.
-10	MSG_OUT_BUFFER_OVERFLOW	An internal error has occurred contact EZTech technical support for assistance.
-11	PARAMETER_TYPE_MISMATCH	An internal error has occurred contact EZTech technical support for assistance.
-12	PARAMETER_COUNT_MISMATCH	An internal error has occurred contact EZTech technical support for assistance.
-13	SERVER_TIMEOUT	The call to the server timed out, although a calls socket was opened successfully.
-14	CONNECTION_BROKEN	The connection with the server was broken.
-15	SOCKET_READ_ERROR	The read socket has been terminated.
-16	NO_MSG_ERROR	There are no waiting events/messages.
-17	SOCKET_WRITE_ERROR	The write socket has been terminated.
-18	SERVER_NONASYNC_CALL	An internal error has occurred contact EZTech technical support for

		assistance.
-19	SOCKET_NOT_CONNECTED	The call socket is not currently connected with the server.
-20	CLIENT_NOT_CONNECTED	The client is not currently connected/logged on to the server.
-21	OCX_NOT_CONNECTED	The OCX control is not currently connected/logged on to the server.
-22	INVALID_CLIENT_TYPE	The logged on client type does not permit this type of call.
-23	INTERNAL_SERVER_ERROR	An unspecified internal error occurred; contact EZTech technical support for assistance.
1	OBJECT_EXISTS_RESULT	The object being created already exists.
2	OBJECT_DOES_NOT_EXIST_RESULT	The object referenced does not exist.
3	OBJECT_HAS_DEPENDANCIES_RESULT	This object cannot be deleted as there are other objects which reference it
4	INVALID_INTERFACE_RESULT	An internal error has occurred contact EZTech technical support for assistance.
5	INVALID_EVENTS_SOCKET_RESULT	An internal error has occurred contact EZTech technical support for assistance.
6	INVALID_OBJECT_LINK_RESULT	One of the IDs passed representing a related object references a non-existing object.
7	INVALID_OBJECT_PARAMETER_RESULT	One of the property values passed is out of range.
8	NOT_LOGGED_ON_RESULT	The client is not currently logged on.
9	ALREADY_LOGGED_ON_RESULT	A client with this client ID is already logged on.
10	INVALID_LOGON_RESULT	An internal error has occurred contact EZTech technical support for assistance.
11	INVALID_CLIENT_TYPE_RESULT	The client type is not database client.
12	PUMP_NOT_RESPONDING_RESULT	The pump is not responding.
13	PUMP_IN_USE_RESULT	The pump is not idle.
14	PUMP_ALREADY_RESERVED_RESULT	The pump already has either a prepay, preauth or payment reserve placed on it.
15	PUMP_NOT_AVAILABLE_RESULT	The pump has a current delivery which cannot be stacked automatically.
16	PUMP_NOT_RESERVED_RESULT	The pump does not have a preauth or prepay reserved place on it.
17	PUMP_NOT_RESERVED_FOR_PREPAY_RESULT	The pump was not reserved for a prepay.
18	PUMP_NOT_RESERVED_BY_YOU_RESULT	The pump was not reserved by this client.
19	INVALID_PRESET_TYPE_RESULT	The preset type requested is not permitted for this type of authorization.
20	INVALID_HOSE_MASK_RESULT	None of the permitted hoses are configured on this pump.
21	PUMP_NOT_RESERVED_FOR_PREAUTH_RESULT	The pump was not reserved for a

		preauth.
22	PREPAYS_NOT_PREMITTED_RESULT	Prepays are not permitted on this pump.
23	PREAUTHS_NOT_PREMITTED_RESULT	Preauths are not permitted on this pump.
24	PUMP_CANNOT_BE_AUTHED_RESULT	The pump mode does not permit post pay deliveries.
25	PUMP_NOT_AUTHED_RESULT	The pump was not manually authorized.
26	NO_DELIVERY_AVAILABLE_RESULT	The delivery requested does not exist.
27	STACK_IS_DISABLED_RESULT	Stacking deliveries is not permitted on this pump.
28	NO_CURRENT_DELIVERY_RESULT	This pump has no current delivery.
29	STACK_FULL_RESULT	The delivery stack for this pump is full.
30	PUMP_NOT_STOPPED_RESULT	The pump is not temp stopped.
31	PUMP_NOT_DELIVERING_RESULT	The pump is not currently delivering.
32	INVALID_PRESET_AMOUNT_RESULT	The preset amount is less than the configured minimum.
33	PUMP_IS_STOPPED_RESULT	The pump is temp stopped.
34	DELIVERY_ALREADY_LOCKED_RESULT	This delivery has already been locked, by you or another client.
35	DELIVERY_IS_RESERVED_RESULT	The delivery was done as a prepay or preauth delivery and is reserved for another EZClient.
36	DELIVERY_NOT_LOCKED_RESULT	The delivery is not currently locked.
37	DELIVERY_NOT_LOCKED_BY_YOU_RESULT	The delivery is locked by another EZServer client.
38	DELIVERY_TYPE_CANNOT_BE_STACKED_RESULT	This type of delivery cannot be stacked.
39	DELIVERY_CANNOT_BE_CLEARED_AS_THIS_TYPE_RESULT	The original type of the delivery does not permit it to be cleared as this type.
40	DELIVERY_NOT_CURRENT_RESULT	The delivery is not the current delivery.
41	INVALID_CLIENT_ID_RESULT	The current Client ID is not logged onto the server.
42	DELIVERY_TERMINATED_RESULT	A delivery on the pump has been terminated, and hence the pump cannot be re-authorized.
43	HAS_CURRENT_DELIVERY_RESULT	The pump has a current delivery which cannot be automatically stacked.
44	ATTENDANT_NOT_LOGGED_ON_RESULT	The attendant being logged off, is not currently logged on.
45	ATTENDANT_ALREADY_LOGGED_ON_RESULT	The pump the attendant is attempting to logon to, already has a logged on attendant.
46	PUMP_IN_WRONG_AUTH_MODE_RESULT	The pump that the attendant is attempting to logon to is not in one of the attendant authorization modes.
47	PUMP_HAS_DELIVERIES_RESULT	The pump being logged on/off has uncleared deliveries.
48	SERVER_NOT_LICENSED_RESULT	The license key for the server is invalid or absent.
49	NO_EZMOD_RESULT	The EZModule for this license key cannot be found, it is most likely turned off or not plugged in.
50	LICENSE_EXPIRED_RESULT	The license key has expired.

51	CTF_NOT_PREMITTED_RESULT	Reserved for CTFs use.
52	PUMP_NOT_RESERVED_FOR_CTF_RESULT	Reserved for CTFs use.
53	ZIGBEE_MODULE_TYPE_ERROR_RESULT	The type of the ZigBee device is incompatible with the API being attempted.
54	DELAY_LOGON_RESULT	An attempt is being made to logon with a client ID already in use.
55	STANDALONE_AUTHMODE_ERROR_RESULT	The authorization mode being selected for the pump is incompatible with the standalone configuration of the EZmod.
56	SERVER_CLIENT_INCOMPATIBLE_ERROR_RESULT	The version of the EZClient.DLL is incompatible with the version of the EZServer.
57	TAG_ALREADY_IN_USE_ERROR_RESULT	The new tag value is already in use by another pump attendant or card client.
58	LOG_EVENT_ALREADY_ACKED_RESULT	The log event has already been cleared or acked.
59	BAD_SESSION_ID_RESULT	A client with this ID and a different session ID is already logged on.
60	SOCKETS_INUSE_RESULT	A client cannot be logged off because the sockets are still in use,
61	SOCKET_UNCHANGED_RESULT	A reconnection has failed to unchanged socket.
62	INVALID_SOCKET_RESULT	A reconnection has failed due to no current socket.

9.14. Appendix 14 – Client event types

Value	Type	Event retrieval function
0	NO_CLIENT_EVENT	**** the events queue is empty ****
1	PUMP_EVENT	GetNextPumpEvent(Ex, Ex2, Ex3) / StatusEvent
2	DELIVERY_EVENT	GetNextDeliveryEvent
3	SERVER_EVENT	GetNextServerEvent / ServerEvent
4	CLIENT_EVENT	GetNextClientEvent/ClientEvent
5	DB_LOG_EVENT	GetNextDBLogEvent/DBLogEvent
6	DB_LOG_DELIVERY	GetNextDBLogDeliveryEvent / DBLogDeliveryEvent
7	DB_CLEAR_DELIVERY	GetNextDBClearDeliveryEvent / DBClearDeliveryEvent
8	DB_STACK_DELIVERY	GetNextDBStackDeliveryEvent / DBStackDeliveryEvent
9	DB_LOG_ETOTALS	GetNextDBHoseETotalsEvent(Ex) / DBHoseETotalsEvent
10	DB_TRIGGER	GetNextDBTriggerEvent/DBTriggerEvent
11	DB_ATTENDANT_LOGON_EVENT	GetNextDBAttendantLogonEvent / DBAttendantLogonEvent
12	DB_ATTENDANT_LOGOFF_EVENT	GetNextDBAttendantLogoffEvent / DBAttendantLogonEvent
13	DB_TANK_STATUS	GetNextDBTankStatusEvent(Ex,Ex2) / DBTankStatusEvent(Ex,Ex2)
14	SERIAL_PORT_EVENT	Reserved for CTFs use
15	ZIGBEE_EVENT	Reserved for CTFs use
16	UVE_EVENT	Reserved for CTFs use
17	ZERO_DELIVERY_EVENT	GetNextZeroDeliveryEvent
18	ZB_STATUS_EVENT	Reserved for CTFs use
19	ZB_PAN_EVENT	Reserved for CTFs use
20	ZIGBEE_CMD_EVENT	Reserved for CTFs use
21	ZIGBEE_RAW_EVENT	Reserved for CTFs use
22	CARD_READ_EVENT	GetNextCardReadEvent / CardReadEvent
23	ZB2G_STATUS_EVENT	GetNextZB2GStatusEvent
24	LOG_EVENT_EVENT	GetNextLogEventEvent/LogEventEvent

9.15. Appendix 15 – Client type

Value	Type	Description
0x01	CALLS_CLIENT_TYPE	The client will simply call non event or configuration related APIs
0x02	EVENTS_CLIENT_TYPE	The client will receive events, and can call the event related APIs.
0x04	DB_CLIENT_TYPE	**The client can call**

9.16. Appendix 16 – Remote device type

Value	Type	Description
1	IRB_ZBDEVICE	Reserved CTFs use.
2	MTF_ZBDEVICE	Reserved CTFs use.
3	OTHER_ZBDEVICE	Reserved CTFs use.
4	TRM_ZBDEVICE	Reserved CTFs use.
5	EZREMOTE_WIRELESS_ZBDEVICE	An EZRemote connected wirelessly.
6	EZREMOTE_SLOT1_ZBDEVICE	An EZRemote connected via cable on slot 1.
7	EZREMOTE_SLOT2_ZBDEVICE	An EZRemote connected via cable on slot 2.
8	EZREMOTE_SLOT3_ZBDEVICE	An EZRemote connected via cable on slot 3.
9	EZREMOTE_SLOT4_ZBDEVICE	An EZRemote connected via cable on slot 4.

9.17. Appendix 17 – Price Control

Value	Type	Event retrieval function
1	REMOTE_PRICE_CONTROL	The price is set and controlled by the forecourt controller
2	LOCAL_PRICE_CONTROL	The price is set manually at the pump.

9.18. Appendix 18 – Price Type

Value	Type	Description
1	FIXED_PRICE_TYPE	The price is the actual price value to be applied for this delivery.
2	DISCOUNT_PRICE_TYPE	The price is to the current pump price less this discount value.
3	SURCHARGE_PRICE_TYPE	The price is to the current pump price plus this surcharge value.

9.19. Appendix 19 – Price Duration Type

Value	Type	Description
1	SINGLE_DURATION_TYPE	This new price is to be used for the next delivery only, after the delivery is complete the price will revert to the old price.
2	MULTIPLE_DURATION_TYPE	This new price is to be used for the permanently until the price is changed again.

9.20. Appendix 20 – Log Event Device Type

Value	Type	Description
1	SERVER_ALR	The EZserver service.
2	PUMP_ALR	A fueling position object.
3	TANK_ALR	A tank object.
4	HOSE_ALR	A hose object.
5	GRADE_ALR	A grade object.
6	EZID_ALR	An EZRemote device.
7	ATTENDANT_ALR	A pump attendant object.
8	CARD_ALR	A card client object.
9	PORT_ALR	A port object.
10	PROCESS_ALR	A server thread/process.
11	POS_ALR	A POS device.
12	ATG_ALR	An ATG device.
13	SENSOR_ALR	A sensor/leak detector device.
14	PRICE_SIGN_ALR	An electronic price sign device.
15	USERDEFINED_ALR	A user defined device.

9.21. Appendix 21 – Log Event Level

Value	Type	Description
0	INFORMATION_EVENT	Information level
1	WARNING_EVENT	Warning level
2	ALARM_EVENT	Alarm level

9.22. Appendix 22 – Log Event Type

Value	Type	Description
0	INITIALIZE_TALR	A process, driver or thread was initialized.
1	TERMINATE_TALR	A process, driver or thread was terminated.
2	START_TALR	A device started responding.
3	STOP_TALR	A device stopped responding.
4	STOP_START_TALR	A device stopped responding and shortly after started responding again.
5	ADD_TALR	An object has added to the configuration
6	DELETE_TALR	An object has been deleted from the configuration
7	EDIT_TALR	The configuration of an object has changed.
8	PRICE_TALR	A remote price change has been effected.
9	OFFPRICE_TALR	A local price change has been detected.
10	OFFLINE_TALR	An offline delivery has been detected.
11	ETOT_REVERSE_TALR	The electronic totals have reversed.
12	ETOT_ZEROED_TALR	The electronic totals have been zeroed.
13	MEMORY_TALR	The EZserver is running very low on memory and has reset.
14	RESET_TALR	The EZserver was reset due to a segment violation.
100	TANK_LOW_PRODUCT_WARNING_START_TALR	Tank product low warning started.
101	TANK_LOW_PRODUCT_WARNING_END_TALR	Tank product low warning ended.
102	TANK_LOW_PRODUCT_ALARM_START_TALR	Tank product low alarm started.
103	TANK_LOW_PRODUCT_ALARM_END_TALR	Tank product low alarm ended.
104	TANK_HI_PRODUCT_WARNING_START_TALR	Tank product high warning started.
105	TANK_HI_PRODUCT_WARNING_END_TALR	Tank product high warning ended.
106	TANK_HI_PRODUCT_ALARM_START_TALR	Tank product high alarm started.
107	TANK_HI_PRODUCT_ALARM_END_TALR	Tank product high alarm ended.
108	TANK_HI_WATER_WARNING_START_TALR	Tank water level high warning started.
109	TANK_HI_WATER_WARNING_END_TALR	Tank water level high warning ended.
110	TANK_HI_WATER_ALARM_START_TALR	Tank water level high alarm started.
111	TANK_HI_WATER_ALARM_END_TALR	Tank water level high alarm ended.
112	TANK_PROBE_STOPPED_RESPONDING_TALR	Tank probe started responding.

113	TANK_PROBE_STARTED_RESPONDING_TALR	Tank probe stopped responding.
114	TANK_ATG_STOPPED_RESPONDING_TALR	ATG started responding.
115	TANK_ATG_STARTED_RESPONDING_TALR	ATG stopped responding.
116	TANK_DROP_START_TALR	Tank drop started.
117	TANK_DROP_END_TALR	Tank drop completed.
118	TANK_DROP_DOCUMENT_TALR	Tank drop documentation.
119	TANK_LEAK_START_TALR	Tank leak started.
120	TANK_LEAK_END_TALR	Tank lean ended.
121	TANK_STATE_TALR	Saved tank status.
122	TANK_CONFIG_TALR	Tank configuration error detected.
123	TANK_CALIBRATION_ERROR_TALR	Tank calibration error detected.
200	ATTENDANT_LOGGED_ON_TALR	Pump attendant logged on.
201	ATTENDANT_LOGGED_OFF_TALR	Pump attendant logged off.
202	ATTENDANT_CARD_BLOCKED_TALR	Blocked pump attendant card used.
203	ATTENDANT_WRONG_SHIFT_TALR	Out of shift pump attendant card used.
204	CLIENT_CARD_BLOCKED_TALR	Blocked card client card used.
205	UNKNOWN_CARD_TALR	Unknown card used.
300	SENSOR_ON_TALR	Sensor alarm started.
301	SENSOR_OFF_TALR	Sensor alarm ended.

9.23. Appendix 23 – Tank State

Value	Type	Description
0	NOT_RESPONDING_TANKSTATE	Tank probe or ATG not responding.
1	IDLE_TANKSTATE	Tank in idle state, not receiving or dispensing fuel.
2	DELIVERING_TANKSTATE	At least one hose connected to this tank is currently dispensing fuel.
3	RECEIVING_TANKSTATE	The tank is currently receiving a tank drop, it is also possible that it is dispensing as well.

9.24. Appendix 24 – Attendant Type

Value	Type	Description
0	BLOCKED_ATTSTATE	This pump attendant is currently blocked.
1	ENABLED_ATTSTATE	This pump attendant is permanently enabled.
3	LOG_ON_OFF_ATTSTATE	This pump attendant is only enabled when logged on.

9.25. Appendix 24 – Alarms Mask

Value	Type	Description
0x00000001	TANK_HI_PRODUCT_ALARM_BIT	The tank product is currently above the high alarm limit.
0x00000002	TANK_HI_PRODUCT_WARNING_BIT	The tank product is currently above the high warning limit.
0x00000004	TANK_LOW_PRODUCT_ALARM_BIT	The tank product is currently below the low alarm limit.
0x00000008	TANK_LOW_PRODUCT_WARNING_BIT	The tank product is currently below the low warning limit.
0x00000010	TANK_HI_WATER_ALARM_BIT	The tank water level is currently above the alarm limit.
0x00000020	TANK_HI_WATER_WARNING_BIT	The tank water level is currently above the warning limit.
0x00000040	TANK_RESPONDING_BIT	The tank probe is currently responding.
0x00000080	TANK_DROP_BIT	A tank drop is in progress.
0x00000100	TANK_LEAK_BIT	The tank is currently leaking
0x00000200	TANK_CONFIG_BIT	There is a tank configuration error.
0x00000400	ATG_RESPONDING_BIT	The tank ATG is currently responding.
0x00000800	TANK_CALIBRATION_ERROR_BIT	There is a calibration error in the tank strapping table.

9.26. Appendix 26 – Card Read Types

Value	Type	Description
1	ATTENDANT_TAG_TYPE	A card linked to a pump attendant logged on or enabled.
2	BLOCKED_ATTENDANT_TAG_TYPE	A card linked to a blocked pump attendant
3	WRONG_SHIFT_ATTENDANT_TAG_TYPE	A card linked to a pump attendant used outside of his/her defined shift.
4	CLIENT_TAG_TYPE	A card linked to an enabled card client.
5	BLOCKED_CLIENT_TAG_TYPE	A card linked to a blocked card client.
6	UNKNOWN_TAG_TYPE	A card not linked to a pump attendant or card client.

9.27. Appendix 27 – Card Types

Value	Type	Description
1	VEHICLE_CARD_TYPE	A vehicle identification card.
2	ATTENDANT_CARD_TYPE	An attendant identification card.
3	DRIVER_CARD_TYPE	A driver identifaction card.
4	SECONDARY_VEHICLE_CARD_TYPE	A secondary vehicle card, it is linked to a vehicle identification card.
5	SECONDARY_ATTENDANT_CARD_TYPE	A secondary attendant card, it is linked to an attendant identification card.
6	SECONDARY_DRIVER_CARD_TYPE	A secondary driver card, it is linked to a driver identification card.

9.28. Appendix 28 – Entry Types

Value	Type	Description
1	NO_ENTRY_TYPE	No additional data entry required.
2	ODO_ENTRY_TYPE	Odometer entry required.
3	DRIVER_ENTRY_TYPE	Driver entry required.
4	ODO_DRIVER_ENTRY_TYPE	Odometer & Driver entry required.
5	OTHER_ENTRY_TYPE	Other data entry required.
6	OTHER_ODO_ENTRY_TYPE	Other data & Odometer entry required.
7	OTHER_DRIVER_ENTRY_TYPE	Other data & Driver entry required.
8	OTHER_ODO_DRIVER_ENTRY_TYPE	Other data, Odometer & Driver entry required.
9	LIMIT_ENTRY_TYPE	Delivery limit entry required.
10	LIMIT_ODO_ENTRY_TYPE	Delivery limit & Odometer entry required.
11	LIMIT_DRIVER_ENTRY_TYPE	Delivery limit & Driver entry required.
12	LIMIT_ODO_DRIVER_ENTRY_TYPE	Delivery limit & Odometer data entry required.
13	LIMIT_OTHER_ENTRY_TYPE	Delivery limit & Other data entry required.
14	LIMIT_OTHER_ODO_ENTRY_TYPE	Delivery limit, Other data & Odometer entry required.
15	LIMIT_OTHER_DRIVER_ENTRY_TYPE	Delivery limit, Other data & Driver entry required.
16	LIMIT_OTHER_ODO_DRIVER_ENTRY_TYPE	Delivery limit, Other data, Odometer & Driver entry required.