CHAOS

Andrews & Arnold Ltd application interface for control systems and ordering

Adrian Kennard, Director

Version 2 as of Wednesday, 21 June 2017

Status

This system is still under development, and some of the features are not yet enabled or completed. Even when completed, features are subject to change (usually by adding more options). However, the system is now available for use by customers. Please let us have feedback.

Overview

CHAOS is an application interface (API) for machine to machine interaction with our control and ordering systems.

It provides a means to perform the following key operations:-

- Access information about an existing service
- Adjust settings on an existing service
- Obtain usage data relating to an existing service
- Order a cease of an existing service
- Order a change to an existing service
- Check availability of new services
- Order a new service, including creating a new account

The control systems covers a wide range of services, including broadband, ethernet, telephony, domains, email, mobile SIMs, and so on.

Whilst CHAOS may initially not provide access to all of these services, it is intended to be a general platform that can be extended over time to provide a comprehensive machine to machine interface to all of our services.

Whilst the design is for machine to machine interactions, an *options* system is included that makes it possible to use a generic front end to present options to an end user for interactive operations such as ordering.

Version

This is version 2. The version is part of the URL for the API.

https://chaos2.aa.net.uk/

We may add new objects and attributes without changing the version number.

This is an ongoing project and subject to change, but obviously we will aim to be backwards compatible, and announce any significant changes. This API is a free service, and can be withdrawn at any time. Like all of our free services it comes with a maximum of a money back guarantee, as per our normal terms.

Development/test

We may, at some point in the future, create a customer test platform, but at present this does not exist.

Authentication and security

The system make use of two separate authorisation systems - one is the accounts system which uses an account number and password to access invoicing and payment details. The other is a control pages login and password which allows access to existing services.

CHAOS makes use of both of these authentications systems.

The account number and password is considered the *master* and can be used to allow access to any of the services on that account.

The control login and password is considered more of a technical administration function and allows access only to the control pages services related to that login in the same way as a login to the control pages web interface. It cannot be used to place orders.

The principle is that access to service information, usage data, and means to adjust existing services can be done using either account or control authentication. However, any changes to billable services, such as ceasing, regrading, or ordering of new services, requires an account login.

Dealers and managers

A dealer for an account is permitted to place orders in relation to the account of their client. To do this, simply use the customer's account-number and the dealer's account password. The order will be logged as having been place by the dealer. Access to control pages will only for for the account-number specified.

A dealer/manager for the control pages simply means one login can access services of some other logins - there are a number of different levels to this. Simply use the dealer control-login and control-password to access services permitted by that login. Again, access is logged against the dealer login.

New accounts

It is also possible to order new services as a new account holder, causing an account to be created in the process. This does not require authentication, but does require billing and bank details and is heavily rate limited.

Rate limiting

To avoid abuse, the system contains a number of rate limiting systems. It is unlikely that these will impact any normal users, but if you encounter errors because of rate limiting, please contact support.

Basic operation

The basic operation involves requests made to the API and responses returned. Each request is independent, and there are no cookies or sessions involved at the http level.

In many cases a single request and response can complete the action required. However, the ordering process is designed to support an iterative sequence of requests such as availability, selection of options, checking for any errors, and further submission before actually placing an order. It is, however, possible to place a complete order in one request if you have all of the correct details.

HTTPS

The request is sent via HTTPS to https://chaos2.aa.net.uk/subsystem/command where *subsystem* and *command* depend on the specific request.

Unencrypted HTTP is not supported. Cleartext passwords are used via HTTPS but not included in responses

Request format

The design is to keep things simple, and as such the request consists of a number of *attributes*. There is no structure or order. Each attribute has a name (which may contain letters, numbers or hyphen/underscore) and a value (though the value is optional in the case of a Boolean attribute, simply being present means it is set). Each named attribute can only occur once in the request.

We recognise that CHAOS may be used by a variety of different systems from curl/wget, javascript/ajax, right through to VB/.NET systems. To support these the request can be submitted in a number of ways:-

- GET with URI encoded query string
  e.g. https://chaos2.aa.net.uk/broadband/quota?control_login=test@a&control_password=fred&service=0123456789
- POST using application/x-www-form-urlencoded
- POST using multipart/form-data
- POST using application/json
  e.g. {“control_login”:“test@a”,”control_password”:“fred”,”service”:“0123456789”}
- POST using text/xml with XML attributes
  e.g. <chaos control-login=“test@a” control-password=“fred” service=“0123456789”/>
- POST using text/xml with XML sub objects
  e.g. <chaos><control-login>test@a</control-login><control-password>fred</control-password><service>0123456789</service></chaos>
- POST using text/xml as SOAP
  i.e. <Envelope> containing <Body> where Body uses one of the above two XML formats for the request attributes

The XML handling does not care what order the attributes or objects are included, and ignores all namespaces. You are welcome to include namespaces though, for compatibility with systems that expect to send them. Attributes are case insensitive, and hyphens or underscores can be used interchangeably in requests. XML responses use hyphens, JSON responses use underscores.

The attributes to include depend on the subsystem but a number of standard attributes are defined for all subsystems.

Response format

The response is structured and contains a number of subordinate objects. The exact structure depends on the specific subsystem but a number of standard objects are defined for all subsystems.

The response can be in one of the following formats. The default is to use the same format as the request, or use JSON if form coding was used in the request. You can force the output format by appending the subsystem/command URL with /json, /xml, or /soap.

- JSON
- XML
- XML in SOAP
  This includes suitable XML namespaces, and includes the response objects in the <Body>. In addition, any error response is included in a standard SOAP style <Fault> object.

JSON encoding

Internally the system operates using XML. When producing output in JSON, hyphens are changed to underscores in attribute names. The top level object name is omitted and a JSON structure is returned. Each XML object is encoded as a structure with all attributes in the XML object included as string fields by the same name as the attribute.

However, subordinate objects can occur more than once in XML, so these are encoded using the name of the object as an array, maintaining the order from XML. So, if your top level JSON is in a variable called chaos, you could access chaos.options[0].option[0].name, for example. This is normally true even when there happens to be only one instance of the subordinate object so as to allow consistent coding in javascript.

There is an exception to this rule where we know that we will only include one instance of a subordinate object, such as <request>, <prices>, and <terms>. In this case the JSON is not encoded using an array, hence allowing access via chaos.prices.price[0].name rather than chaos.prices[0].price[0].name for example.

Subsystem and command

The URL used contains the *subsystem* and *command*. When used with SOAP, this can be omitted and used with a SOAPAction header. e.g. https://chaos2.aa.net.uk/ and a SOAPAction of *subsystem*/*command*, or https://chaos2.aa.net.uk/subsystem and a SOAPAction of *command*.

Authentication

Authentication can be provided in several ways. The simplest is the use of attributes *account-number* and *account-password* to authenticate as an account, or *control-login* and *control-password* to authenticate as a control system user. When authenticating as an account it is normal to include the *control-login* as well, and it must be one related to the account.

However, it is also possible to provide authentication using HTTP Basic authentication where the username is either an account number or control login and the corresponding password. This effectively sets the corresponding attributes.

When using SOAP it is possible to include <Header><Security><UsernameToken> with <Username> and <Password> to authenticate.

Standard commands

Command	Meaning
availability	Request availability and pricing for a new service
services	Request list of service IDs on this control-login for this subsystem
info	Request information about an existing service
adjust	Make changes to an existing service settings
check	Check an order for errors or warnings
order	Place an order for a new service or to change an existing service
cease	Place an order to cease a service
usage	Request usage details for a specific service

Standard Request Attributes

Attribute	Meaning
account-number	The account number for authentication, e.g. A1234A
account-password	The account password for authentication
control-login	The control systems login - used for authentication with control-password
control-password	The control systems password for authentication
service	The identifier for the specific service - the format depends on the service

Standard Response Objects

Object	Meaning
request	Contains all of the request attributes
error	If present, then there was an error and the request was not actioned
options	Details the possible values of attributes that could be used in a further request (see below)

Options system

A general mechanism is used to advise what attributes may be provided, and also to indicate if there is an issue with an attribute that was provided.

One or more <options> objects may be present in the response each containing one or more <option> objects, and each these provide details of one request attribute. The <options> object is simply to group <option> objects cosmetically to allow better user interface presentation.

The <option> objects are included in the following cases:-

- Where a request has missing or incorrect attributes and so an <error> is returned, it provides details of the attributes/values required.
- In response to an info request, detailing the attributes of a service that may be adjusted
- In response to an availability check, detailing the services that are available
- In response to an ordering check

The system is designed so that it could be used to create an interactive user interface, but also so that it provides information to developers on the options that are available, hence making the system self documenting. For this reason, this manual is not updated with every minor change to options that are available.

Where an option has a fixed choice of possible values, these are included in subordinate <choice> objects.

Options

<options>	Meaning
title	A single line description of the group of option objects
description	A longer description of the group of option objects
help	A URI for help on the group of option objects - typically this would be a pop-up on a web page
img	A URI for an image related to the group of option objects

Option

<option>	Meaning
name	The name of the attribute
type	The type of the attribute (see below for types)
optional	Set to “true” if this field is optional. For a text input field it would have to be non blank too. Note that a checkbox can be non-optional if required.
recheck	If present then changing this option may cause significant changes to other options, so it is recommended that the request be re-submitted after any such change.
value	The current or default value of the attribute
checked	Indicates that a *checkbox* type attribute is *checked*

<option>	Meaning
size	The suggested display size (characters) for an input value
max-length	The maximum number of characters for an input value - which may be enforced in a UI
title	A single line description of the attribute
description	A longer description of the attribute
help	A URI for help on the attribute - typically this would be a pop-up on a web page
img	A URI for an image related to the attribute
min	Minimum value acceptable (used for *quantity* and *date*)
max	Maximum value acceptable (used for *quantity* and *date*)
error	An error description indicating that the previously supplied value for this attribute was unacceptable
warning	A warning description indicating that the previously supplied value for this attribute is valid but seems unlikely and may need checking

Attribute types

type	Meaning
hidden	The attribute must be provided with the value specified.
fixed	As hidden, but value may be displayed to user
text	A simple text input
password	A password input - beware of auto-filled passwords on browsers
quantity	A quantity, default is positive integer.
checkbox	A boolean attribute. If *checked* it is included in the request attributes, and if not, then it is not. A value can be sent in the request but it is ignored.
choice	The value must be taken from a set of <choice> sub objects
select	The same as choice, but recommended that a pull down select is used when presented to a user
date	A date (YYYY-MM-DD).
datetime	A datetime (YYYY-MM-DD HH:MM:SS).
time	A time (HH:MM:SS)
email	An email address
telephone	A telephone number
postcode	A UK postcode

Choice

Where the attribute is a choice or select, there will be <choice> sub objects listing the values that are allowed.

<choice>	Meaning
value	The value for this choice
title	One line description
description	Longer description for this value
help	A URI for help on the attribute - typically this would be a pop-up on a web page
img	A URI for an image related to the attribute

Pricing information

It is important to ensure that any interactive ordering system presents the pricing details of options that are selected. To enable this <option> and <choice> can contain a number of *price-* fields which can be used to specify the price.

Pricing is, however, not simply a matter of one value. There may be several aspects to a price including *Installation*, *Equipment*, *Monthly*, *Early termination*, *Cease charges*, and so on.

To accommodate this in a general way, each type of price is given a simple tag, such as *install*, and a field is included in the <prices> object. This contains a <price> object for each type of price.

The <prices> object also has *vat-preferred* which can be “inc” or “exc” as a guide to how pricing should be shown to the customer.

<price>	Meaning
name	Type of pricing, e.g. “install”
title	One line description, e.g. “One-off installation charges”
suffix	If pricing to have a suffix, e.g. “/month”
inc	Total inclusive of VAT prices for current selected options in request
exc	Total exclusive of VAT prices for current selected options in request
help	A URI for help on the attribute - typically this would be a pop-up on a web page
img	A URI for an image related to the attribute

<options>, <option> and <choice> can then include a number of fields of the form *price-tag-inc* and *price-tag-exc* which specify the price in pounds both VAT inclusive and VAT exclusive, *tag* being the type of price such as “install”.

To work out the total for each type of price, add up any prices in <options>, <option>, and any prices in <choice> for the choice value that matches the option value. These need to be added up separately for VAT inclusive and exclusive amounts. Note where <option> is type checkbox, only add if checked.

Ordering

The ordering process usually starts by performing an availability check of some sort.

This returns <options>/<option> objects asking for more information, and can include a <prices> object with price headers.

Once enough details are provided to allow an order to at least be checked, the <prices> object will contain a *complete* field.

At this stage it is possible to run <check> which will simply confirm all attributes are sensible and that an order could be placed. If not, then an <error> is returned. It is also possible at this stage for individual options to have a warning provided even when no <error>.

The availability, and check functions also provide a <terms> objects which contains one or more <term> objects with text descriptions of key contract terms. The <terms> contains name/title/ description which is used for a check box - the named attribute should be sent to indicate agreement to the terms.

<terms>	Meaning
name	Name of attribute to be sent to confirm terms agreed
title	A single line description of the group of option objects
description	A longer description of the group of option objects
help	A URI for help on the group of option objects - typically this would be a pop-up on a web page
img	A URI for an image related to the group of option objects

Note that it is quite valid to place a complete order in one request if all details are known.

The response to *check* or *order* commands if all is well is no <error>, though the <options>, etc, are all still included as normal.

The response to an order includes one or more <OrderConfirmation> as well as <SalesInvoice> or <ProformaSalesInvoice> objects. These are formatted in accordance with our accounts system XML specifications.

New customers

For new customers the <option> objects will include a lot of attributes that start *account-* which are used to create the new account and define the invoice address.

In the case of services that include equipment, there may also be a number of attributes that start with *delivery-* which define the delivery address using similar fields.

Obviously the best way to manage these is present them all to the user. Some fields may only be included based on the selection made in other attributes.

The main suffixes used in these fields are :-

Suffix	Meaning
name	The contact person's name, usually done without Mr/Mrs prefix
company	The company name
type	Used as account-type, the type of the account
address1	First line of address - don't repeat company name here
address2	Second line of address
address3	Third line of address
posttown	The post town - note that county is not required
postcode	The UK post code
telephone	Contact telephone number
mobile	Contact mobile telephone number (typically for SMS)
regno	Company registration number

The account type is quite important and impacts some of the other fields that may be needed.

Type	Meaning
I	Individual (e.g. residential / personal)
M	Minor (under 18)
L	Small limited company (10 or fewer people working for company)
G	Larger limited company (more than 10 people working for company)
C	Communications provider as defined by The Communications Act
Q	Public Limited Company (plc)
P	Partnership (put the trading name in *company*)
S	Sole trader (put the trading name in *company*)
R	Registered Charity (put charity name in *company*)
F	Friendly Society (put society name in *company*)

Info/Adjust

One of the useful features of CHAOS is the ability to access the basic settings of services, and make changes.

This is designed to be as generic a possible, and hence allow us to expand the system quickly to all services on our control pages.

The *info* request uses the *service* attribute to specify the specific service. The response includes an <info> object with a simple list of all attributes which can be provided.

In addition the response includes <options> which lists all of the attributes (with current values) which may be changed. It includes the data type and if appropriate <choice> objects for the changes.

You can then use *adjust* with the attributes provide in the <options> response to make any changes. The response is either an error (with <options> making it clear where the error lies), or just the <info> object containing the new values.

The exact set of attributes which can be accessed or changed may change over time, so it is important to try and make use of the <options> response to know what is possible.

It is possible, though unlikely, that we might remove an attribute at some point in the future.

In some cases a change may cause some knock on effect or action, such as changing broadband line settings. This could mean losing sync, for example.

Login subsystem

The *login* subsystem us used simply to allow *info* and *adjust* of settings relating to a control login. The *service* identifier is the control-login in question, e.g. test@a

It is also useful with the *services* command which returns a list of the control system logins available.

Broadband subsystem

The *broadband* subsystem relates to ADSL, VDSL, and FTTP Internet access services.

The *service* identifier can be specified in one of a number of formats and relates to a specific *line* and not a set of lines in any way:-

- Telephone number of the line for broadband on a phone line
- A&A line ID (simply a number, usually 5 digits) as reported on the voice message on our PSTN lines
- Carrier circuit ID, e.g. BBEU something, as shown on graphs in some cases

The *info* request provides details relating to broadband services which can include quota information and line speed information. Being XML we may extend the fields included over time.

Availability

The availability checker can provide details of possible broadband services related to a phone line or address.

There are several ways to identify the service or address for which you want to check availability:-

- Provide *service* identifier for an existing A&A service that you have.
- Provide *postcode*, and *property* (house number or name). If this is not unique then <options> will give a list of possibilities using an *address-key*.
- Provide *postcode* and *address-key* - this will normally only be after you are offered a choice of addresses in <options>, but if you have other means to find an address key you can use this directly.
- Provide *postcode* and *number* (directory number, aka the phone number of a phone line).

For a new account it is a good idea to include *account-type* in the availability check to only offer appropriate services. If the specific type is not yet known, use account-type="I" to indicate an individual wanting residential or home/office services, and account-type="L" to indicate a business wanting home/office or office services. If omitted then all services are offered. It is useful to understand home/business as well so that VAT inclusive or exclusive pricing can be shown by default.

The response is by way of a number <options> objects for each main service that is available, and then <option>/<choice> for quota and other characteristics of the service.

Broadband specific commands

There are some specific additional commands that relate to the broadband subsystem.

Command	Meaning
kill	Perform a PPP kill/restart on the service
quota	Advise monthly quota and remaining quota on the service

Broadband order

There are a set of attributes that need to be sent to complete an order. The <options> systems provides full details of these in the various circumstances, but the following list represents the main fields and their meaning. These are in additional to account-number/control-login as above.

availability/check/order	Values	Meaning
api	N/A	Tells subsystem to allow some values to be skipped, such as customer-type and tech, allowing simpler availability checks.
token	Text	Optional token used to avoid duplicate orders. Interactive systems should simply include this as a hidden field as requested, and send in next request.
new-account	Y/N	Optional, and controls options for account-number/password
customer-type	H/O	Optional, restricts options for Home or Office customer
tech	ADSL/ VDSL	Type of service requested. Note VDSL also includes FTTP
new-line	Y/N/E	Controls options for if a new phone line needed (Y) or not (N), or existing service (E)
number	Telephone	The phone number for the order
service	ID	An ID for an existing service
property	Text	Property house number or name
postcode	Postcode	Postcode for order installation
address-key	DC+NAD	An address key for specific address wishing a postcode
package	Text	Selected package, e.g. H1ADSL, H1VDSL, H1VDSLTB, etc.
crd	Date	The Customer Required Date, optional, black or omit for ASAP
care	Y/N	Enhanced care option
annexm		If annex M required
premium		If 20CN premium option required
pstnto		If PSTN take over required
capt	4010	If capping to 40Mb/10Mbs is required on VDSL
router	N/R/M	No (N), Router (R), Modem (M)

availability/check/order	Values	Meaning
wifi		If 3 pack Unifi WiFi required
firebrick		If FB2700 required
quota	Bytes	Monthly quota required (not needed for H1VDSLTB/S1VDSLTB)
units	Units	Units required on units quota
block	Y/N/A	Yes/No/Auto blocking action for quota based service
lines	N	If multiple lines required on same login of same type
o1	A/V list	The lines required for an Office::1 order, e.g. VA is VDSL+ADSL, VVA is 2xVDSL+ADSL
filter	N	Non optional field, must be N
moving		If cease of old line on login is required on completion of install
move-account		If update of billing address is required on completion of install
login	login@a/q	The login where adding line to existing login or moving line
purchase-order	P/O No	Purchase order number
pruchase-reference	Test	Text for this line as reference on invoices
account-name/etc	Address	The address fields for a new account
delivery-name/etc	Address	The address fields for a separate delivery address
delivery-signed-for		If delivery is to be signed for
delivery-safe-place	Text	Alternatively if a safe place can be used for delivery
site-name/email/mobile	Text	Site contact details for installation
site-floor/room/position	Text	PSTN install extra details
site-engineer-notes	Test	PSTN engineer notes
terms-agreed		Must be sent to confirm agreed terms

When a broadband *order* command completes and <order> object is returned.

<order>	Meaning
account-number	The account number
account-password-url	If a new account, the URL to set the account password, valid for one use and today only
control-login	The control login
control-password-url	If a new control-login, the URL for setting control system password, valid for one use and today only
ripe-mic	The allocated RIPE handled
title	Heading for displayed ordering confirmation

<order>	Meaning
description	Text of order confirmation
<availability>	Some fields detailing availability
<InstallationAddress>	Installation address details
order-confirmation	The accounts system order confirmation number
order-confirmation-pdf-url	URL to access accounts order confirmation document
<OrderConfirmation>	Accounts system Order Confirmation
sales-invoice	The accounts system sales invoice number
sales-invoice-pdf-url	URL to access accounts system sales invoice document
<SalesInvoice>	Accounts system Sales Invoice
delivery-note	The accounts system delivery note number
delivery-note-pdf-url	URL ro access accounts system delivery note document
<DeliveryNote>	Accounts system Delivery Note

Domain subsystem

TBA

Email subsystem

TBA

Telephony subsystem

The telephone system includes an standard ordering system as describe above, and also the following commands.

Command	Meaning
ratecard	Provides list of numbers mapped to rate names and a list or rates and the charges for each rate.

SIM subsystem

TBA