Archer Developer Guide

Document Version 144

Notice

THE INFORMATION CONTAINED HEREIN IS PROVIDED TO RECIPIENT "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, OR WARRANTY OF TITLE OR NON- INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.

HEARTLAND PAYMENT SYSTEMS SHALL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM THE USE OF ANY INFORMATION CONTAINED HEREIN, WHETHER RESULTING FROM BREACH OF CONTRACT, BREACH OF WARRANTY, NEGLIGENCE, OR OTHERWISE, EVEN IF HEARTLAND PAYMENT SYSTEMS HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. HEARTLAND PAYMENT SYSTEMS RESERVES THE RIGHT TO MAKE CHANGES TO THE INFORMATION CONTAINED HEREIN AT ANY TIME WITHOUT NOTICE.

THIS DOCUMENT AND ALL INFORMATION CONTAINED HEREIN IS PROPRIETARY HEARTLAND PAYMENT SYSTEMS INFORMATION. UNDER ANY CIRCUMSTANCES, RECIPIENT SHALL NOT DISCLOSE THIS DOCUMENT OR THE SYSTEM DESCRIBED HEREIN TO ANY THIRD PARTY WITHOUT PRIOR WRITTEN CONSENT OF A DULY AUTHORIZED REPRESENTATIVE OF HEARTLAND PAYMENT SYSTEMS. IN ORDER TO PROTECT THE CONFIDENTIAL NATURE OF THIS PROPRIETARY INFORMATION, RECIPIENT AGREES:

> *A. TO IMPOSE IN WRITING SIMILAR OBLIGATIONS OF CONFIDENTIALITY AND NON- DISCLOSURE AS CONTAINED HEREIN ON RECIPIENT'S EMPLOYEES AND AUTHORIZED THIRD PARTIES TO WHOM RECIPIENT DISCLOSES THIS INFORMATION (SUCH DISCLOSURE TO BE MADE ON A STRICTLY NEED-TO-KNOW BASIS) PRIOR TO SHARING THIS DOCUMENT AND*

> *B. TO BE RESPONSIBLE FOR ANY BREACH OF CONFIDENTIALITY BY THOSE EMPLOYEES AND THIRD PARTIES TO WHOM RECIPIENT DISCLOSES THIS INFORMATION.*

RECIPIENT ACKNOWLEDGES AND AGREES THAT USE OF THE INFORMATION CONTAINED HEREIN SIGNIFIES ACKNOWLEDGEMENT AND ACCEPTANCE OF THESE TERMS. ANY SUCH USE IS CONDITIONED UPON THE TERMS, CONDITIONS AND OBLIGATIONS CONTAINED WITHIN THIS NOTICE.

THE TRADEMARKS AND SERVICE MARKS RELATING TO PRODUCTS OR SERVICES OF HEARTLAND PAYMENT SYSTEMS OR OF THIRD PARTIES ARE OWNED BY HEARTLAND PAYMENT SYSTEMS OR THE RESPECTIVE THIRD PARTY OWNERS OF THOSE MARKS, AS THE CASE MAY BE, AND NO LICENSE WITH RESPECT TO ANY SUCH MARK IS EITHER GRANTED OR IMPLIED.

TO VERIFY EXISTING CONTENT OR TO OBTAIN ADDITIONAL INFORMATION, PLEASE CALL OR EMAIL YOUR ASSIGNED HEARTLAND PAYMENT SYSTEMS CONTACT.

1. Table of Contents

2. Overview

2.1. Introduction

Archer is a simple, HTTP-based API for performing basic gift, loyalty, and rewards transactions against the Heartland Stored Value Platform. It is designed completely around standard HTTP features that are supported out-of-the-box by your favorite HTTP client, no matter what language or platform you are using.

2.2. Audience

This document is provided for software developers integrating software with the Heartland Stored Value Platform via the Archer API. This API provides a simple integration path that allows developers to build functioning clients quickly and easily. This is accomplished by:

- Leveraging standard HTTP protocols and clients
- Not requiring the use of complicated SOAP, XML or JSON schemas for requests or responses
- Providing a simplified view of the platform's available feature set
- Providing client libraries for certain languages and platforms

2.3. Who This is *Not* Intended For

The Archer API has been designed for simplicity and is not appropriate for everyone. It may not be suitable for your needs if one or more of the following are true:

- You require the ability to send single requests containing multiple activations, loads, redemptions, or combinations thereof
- You require line item (SKU) details, such as in a typical shopping cart
- You require a robust credit and/or debit processing interface
- You require richer support for features of the Heartland Stored Value Platform not exposed via the Archer API

In some cases, these limitations may be removed in future versions of the Archer API. In others, Heartland provides alternative API(s) to integrate with the Heartland Stored Value Platform.

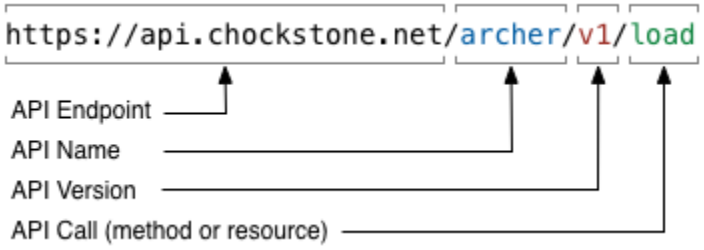
2.4. Other Limitations of the API

The Archer API was designed for use by merchant-oriented applications (including terminal, point of sale, e-commerce, host-to-host, and others) that are configured at runtime, either statically or dynamically, with merchant credentials provided by Heartland Payment Systems. It may be appropriate for some applications, most notably consumer-facing mobile or web applications, to use a different API (such as the Katana API) that is designed with a security and authentication model more appropriate for those types of applications.

3. Archer API Overview

3.1. Using the Archer API

The easiest way to explain how to use Archer is with an example. To send an Archer request, you POST form data to a URL like that shown below.



The URL is broken up into four distinct parts:

1. The **API Endpoint** is the address of the Archer host you are using. Multiple endpoints exist for development, testing, and production use.
2. The **API Name** identifies the fact that you are attempting to use the Archer API (as opposed to another Heartland Stored Value API).
3. The **API Version** denotes that you are using Version 1 of the Archer API.
4. Lastly, the **API Call** identifies what request type you are trying to perform. In this case, the client was performing a "load" to add value to a stored value account.

Each Archer API Call requires a set of parameters necessary to execute the request. These parameters are passed as URL-encoded form data or JSON map that can easily be provided using a standard HTTP client. To demonstrate, the following is a fully functional example of a `load` request using the `curl` command line utility.

The HTTP Content-type and Accept headers control the input and output encoding of parameters, respectively.

3.1.1. URL-encoded request example

```
curl https://api.cert.chockstone.com/archer/v1/load \
    -iu test_user:test_password \
    -X POST \
    -d domain=test \
    -d request=201212312359590001 \
    -d store=0001 \
    -d terminal=0001 \
    -d terminal.order=0001 \
    -d terminal.version=v17 \
    -d sva=6277200000000001 \
    -d acquired=MANUAL \
    -d amount=1000 \
    -d currency=USD
```

3.1.2. JSON request example

```
curl https://api.cert.chockstone.com/archer/v1/load \
    -iu test_user:test_password \
    -X POST \
    -H "Accept: application/json" \
    -H "Content-type: application/json" \
    -d '{
         "domain":"test",
         "request":"201212312359590001",
         "store":"0001",
         "terminal":"0001",
         "terminal.order":"0001",
         "terminal.version":"v17",
         "sva":"6277200000000001",
         "acquired":"MANUAL",
         "amount":"1000",
         "currency":"USD"
        }'
```

The Archer host will return a response that contains an HTTP status code identifying whether the transaction succeeded or failed, along with a URL or JSON-encoded parameters that can be parsed and used by the client.

3.1.3. URL-encoded response example

```
HTTP/1.1 200 OK
Date: Wed, 19 Dec 2012 23:06:33 GMT
Accept-Ranges: bytes
Server: chockstone
Vary: Accept-Charset, Accept-Encoding, Accept-Language, Accept
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
Content-Length: 278

notes=%2A%2A%2A%20Certification%20test%20server%20%2A%2A%2A
&order=135595839312510
&rewards=Points%208
&status.code=200
&status.description=The%20request%20has%20succeeded
&status.name=OK
&sva=XXXXXXXXXXXX0001
&sva.balances=USD%201000%2CPoints%201
&sva.registered=true
&sva.status=ACTIVE
```

Note that actual response data will not be sent with linebreaks or other formatting.

3.1.4. JSON response example

```
HTTP/1.1 200 OK
Date: Wed, 19 Dec 2012 22:59:45 GMT
Accept-Ranges: bytes
Server: chockstone
Vary: Accept-Charset, Accept-Encoding, Accept-Language, Accept
Content-Type: application/json;charset=UTF-8
Transfer-Encoding: chunked

{
   "notes":"*** Certification test server ***",
   "order":"135595798567700",
   "rewards":"Points 8",
   "status.code":"200",
   "status.description":"The request has succeeded",
   "status.name":"OK",
   "sva":"XXXXXXXXXXXX0001",
   "sva.balances":"USD 1000,Points 1",
   "sva.detailedBalances": [
      {
         "currency": "Points",
         "amount": 1,
         "threshold": 5,
         "flavor": "Visit"
      },
      {
         "currency": "USD",
         "amount": 1000,
         "threshold": 0,
         "flavor": ""
      }
   ],

   "sva.registered":"true",
   "sva.status":"ACTIVE"
}
```

Note that actual response data will not be sent with linebreaks or other formatting.

3.2. Errors

Archer uses standard HTTP response codes to indicate success or failure of an API request. In general, `200` means success, `4XX` means an application-level error, and `5XX` means a system-level error. Specific errors as they are translated from the Heartland Stored Value Platform to HTTP response codes are documented in Appendix C - Errors.

3.3. Case Sensitive

The Archer API is case sensitive! Code samples are provided throughout this document to specify currencies, parameter values, etc. If you do not match the cases exactly, the corresponding features will not work as intended.

3.4. Versioning and Backwards Compatibility

Every attempt is made to ensure that all changes made to a specific version of the API remain backwards compatible. Any changes that break backwards compatibility are implemented as a new version of the API, so that clients are unaffected and may migrate to the new version when appropriate.

Changes that are considered to break backwards compatibility include removing request or response parameters, and changing the syntactic structure or semantic meanings of existing parameters.

Note that both the introduction of new, optional request parameters that behave identically as before when not provided in a request, and the introduction of new response parameters that may be ignored by clients are not considered changes that break backwards compatibility. Additionally, it is expected that well-implemented clients use appropriate defensive coding techniques and are able to ignore unexpected response parameters and continue to function correctly.

3.5. Security

All Archer requests must be made using HTTPS. Attempting to use plain `HTTP` is not supported and will consequently fail.

3.6. Client Authorization

The Archer v1 API accepts an optional Heartland-supplied API Key with each request, using the HTTP header "Api-Key". An example API Key is provided below.

```
Api-Key: 8yEsbyXBsaU
```

The following snippet shows how to pass client authorization credentials using the command line tool curl:

```
curl -H "Api-Key: 8yEsbyXBsaU" ...
```

All client requests should still include authentication credentials even when sending the API Key.

3.7. Authentication

Archer utilizes a simple username/password authentication scheme implemented using the standard HTTP Basic Access Authentication protocol. This authentication scheme should be supported by nearly all HTTP client libraries. Because all Archer communication occurs over HTTPS (SSL), the authentication credentials are protected from eavesdropping and are never exposed as plain text.

Archer expects clients to provide the appropriate authentication credentials with every request. Details about passing authentication information to Archer using the Basic Access Authentication Protocol can be found here:http://en.wikipedia.org/wiki/Basic_access_authentication

A quick summary for manually constructing an Authorization HTTP header:

1. A username and password are combined into a string "username:password"
2. The resulting string literal is then encoded using Base64
3. Construct the header as follows (where `"dXNlcm5hbWU6cGFzc3dvcmQK"` is a sample Base64-encoded string from step 2):

```
Authorization: Basic dXNlcm5hbWU6cGFzc3dvcmQK
```

Manual construction of the Basic Authentication header is typically not required if using a standard HTTP client library for communicating with the Archer host. For example, the following code snippet shows how to pass the authentication credentials using the command line tool curl:

```
curl -u username:password ...
```

3.8. Archer API Endpoints

Archer provides three API Endpoints for the different execution environments:

Environment	Endpoint	Description
Certification	https://api.cert.chockstone.com/archer	A stateless sandbox environment for development and certification
Test	https://api.test.chockstone.com/archer	An environment for customer and user acceptance testing
Production	https://api.chockstone.net/archer	The environment for live, production use

The Certification environment does not store and track actual stored value accounts and balances. Instead, it is a completely stateless environment that mimics production behavior but is designed to provide developers the ability to control the responses received for the purpose of testing various transaction scenarios. The developer can force a desired response by sending the appropriate value in one of the request parameters, typically `amount` or `sva`, as documented in the section entitled Appendix D - Certification Host Response Matrix.

The Test environment is identical to the Production environment in function except that it is provided for testing only.

3.9. Summary of Archer API Calls

The following is a summary of the API Calls available within Archer via their URI path:

```
/archer/v1/activate
/archer/v1/load
/archer/v1/redeem
/archer/v1/inquiry
/archer/v1/void
/archer/v1/reverse
/archer/v1/transfer
/archer/v1/reward
/archer/v1/alias/add
/archer/v1/alias/remove
/archer/v1/alias/create
/archer/v1/deactivate
/archer/v1/rewardsinfo
/archer/v1/cashout
```

4. Archer API Reference

This section provides detailed information about each API Call supported by Archer.

Each subsection contains a table that provides a summary about the API call.

Description	A short description of the purpose of the call
Request Path	The path to invoke the call
Reversible	Indicates whether the call should be followed immediately by a `reverse` call on a network error condition
Voidable	Indicates whether the call can be voided

In addition, usage information is provided explaining when and why to use the API call, as well as the request and response parameters necessary to handle the request.

4.1. Activate

Description	Activates and adds an initial balance to a new stored value account
Request Path	`archer/v1/activate`
Reversible	Yes
Voidable	Yes

4.1.1. Usage

Used to activate a new stored value account and load it with an initial balance. If the specified stored value account is already active, the request will fail.

Note: stored value accounts can also be activated implicitly via the `load` method depending on the platform configuration for a customer's specific program.

4.1.2. Request Parameters

Parameter	Required	Description	Example
domain	Yes	Identifies the domain; see domain	hps
chain	Maybe	Used by global partners to send requests on behalf of a group of merchants; see chain	130909013158178
request	Yes	Identifies this unique request; see request	201212312359590001
store	Yes	Identifies the store; see store	0001
terminal	Yes	Identifies the terminal; see terminal	001
terminal.order	Yes	A terminal-generated order number; see terminal.order	1701
terminal.version	Yes	Identifies the terminal software version; see terminal.version	Cyberdyne T-1000
clerk	No	Identifies an individual clerk or server; see clerk	17
sva	Yes	Identifies the stored value account being activated; see sva	6277200000000001
pin	No	Specifies the PIN number associated with the account.	12345
acquired	Yes	Identifies how the sva was acquired; see acquired	SWIPE
amount	Yes	The amount to activate the account with; see amount	1000
currency	Yes	The currency of the activation amount; see currency	USD

4.1.3. Response Parameters

Parameter	Always Exists	Description	Example
status.code	Yes	Response status code; see status.code	200
status.name	Yes	Response status name; see status.name	AccountNotFound
status.description	Yes	Long, detailed description about the error; see status.description	
order	Yes	A host-generated order ID; Used for voids.	120100010000000063
sva	Yes	The masked version of the stored value account	XXXXXXXXXXXX0001
sva.status	Yes	The status of the stored value account; see sva.status	ACTIVE
sva.registered	Yes	Identifies whether the account is registered or not; see sva.registered	true

sva.balances	Yes	The account's balances; see sva.balances	"USD 1000,Points 1"
sva.detailedBalances	JSON-only	The account's balances and progress towards rewards; see sva.detailedBalances	
rewards	No	A list of any rewards added to the account; see rewards	"Points 5"
notes	No	Rewards notes; see notes	"You have earned an extra $1 for loading $10 onto your account!"

4.2. Load

Description	Loads value onto a stored value account
Request Path	`archer/v1/load`
Reversible	Yes
Voidable	Yes

4.2.1. Description

Loads a stored value account. In most cases (depending on the platform configuration for a customer) this request will also automatically activate the account if necessary.

This method is typically used to add actual tendered currencies such as `USD`, but can also be used to add `Points` (although this is discouraged in practice, as `Points` are typically awarded automatically by the host platform as defined by the customer's specific rewards program).

4.2.2. Request Parameters

Parameter	Required	Description	Example
domain	Yes	Identifies the domain; see domain	hps
chain	Maybe	Used by global partners to send requests on behalf of a group of merchants; see chain	130909013158178
request	Yes	Identifies this unique request; see request	201212312359590001
store	Yes	Identifies the store; see store	0001
terminal	Yes	Identifies the terminal; see terminal	001
terminal.order	Yes	A terminal-generated order number; see terminal.order	1701
terminal.version	Yes	Identifies the terminal software version; see terminal.version	Cyberdyne T-1000
clerk	No	Identifies an individual clerk or server; see clerk	17
sva	Yes	Identifies the stored value account being loaded; see sva	6277200000000001
pin	No	Specifies the PIN number associated with the account.	12345
acquired	Yes	Identifies how the sva was acquired; see acquired	SWIPE
amount	Yes	The amount to load; see amount	1000
currency	Yes	The currency being loaded; see currency	USD

4.2.3. Response Parameters

Parameter	Always Exists	Description	Example
status.code	Yes	Response status code; see status.code	200
status.name	Yes	Response status name; see status.name	AccountNotFound
status.description	Yes	Long, detailed description about the error; see status.description	
order	Yes	A host-generated order ID; Used for voids.	120100010000000063
sva	Yes	The masked version of the stored value account	XXXXXXXXXXXX0001
sva.status	Yes	The status of the stored value account; see sva.status	ACTIVE
sva.registered	Yes	Identifies whether the account is registered or not; see sva.registered	true

sva.balances	Yes	The account's balances; see sva.balances	"USD 1000,Points 1"
sva.detailedBalances	JSON-only	The account's balances and progress towards rewards; see sva.detailedBalances	
rewards	No	A list of any rewards added to the account; see rewards	"Points 5"
notes	No	Rewards notes; see notes	"You have earned an extra $1 for loading $10 onto your account!"

4.3. Redeem

Description	Redeems value from a stored value account
Request Path	`archer/v1/redeem`
Reversible	Yes
Voidable	Yes

4.3.1. Usage

Redeems value from a stored value account for the amount and currency specified. The corresponding account balance is decremented accordingly.

By default, if the account balance is non-zero but insufficient to cover the full redemption amount, the remaining balance will be drained and the amount still owed will be returned in the response for additional payment. This behavior is consistent with explicitly providing the partial parameter with a value of "true". To disable this behavior so that the full amount must be redeemed or else the entire transaction fails, the client should provide the partial parameter with the value of "false".

When partial authorization is being accepted and there is an amount still owed by the consumer, the merchant may accept any additional tender to cover the balance. If the account holder is unable to provide additional payment and the purchase is canceled, then this transaction should be voided to return the balance back to the account.

4.3.2. Request Parameters

Parameter	Required	Description	Example
domain	Yes	Identifies the domain; see domain	hps
chain	Maybe	Used by global partners to send requests on behalf of a group of merchants; see chain	130909013158178
request	Yes	Identifies this unique request; see request	201212312359590001
store	Yes	Identifies the store; see store	0001
terminal	Yes	Identifies the terminal; see terminal	001
terminal.order	Yes	A terminal-generated order number; see terminal.order	1701
terminal.version	Yes	Identifies the terminal software version; see terminal.version	Cyberdyne T-1000
clerk	No	Identifies an individual clerk or server; see clerk	17
sva	Yes	Identifies the stored value account being redeemed; see sva	6277200000000001
acquired	Yes	Identifies how the sva was acquired; see acquired	SWIPE
amount	Yes	The amount to redeem from the account; see amount	1000
currency	Yes	The currency of the redemption amount; see currency	USD
partial	No	Identifies whether a partial authorization is accepted or not; see partial	true
pin	No	The pin associated with the sva; see pin	123456
tax	No	The portion of the amount being redeemed that represents tax; see tax	80
tip	No	The portion of the amount being redeemed that represents a tip; see tip	200

4.3.3. Response Parameters

Parameter	Always Exists	Description	Example
status.code	Yes	Response status code; see status.code	200
status.name	Yes	Response status name; see status.name	AccountNotFound

status.description	Yes	Long, detailed description about the error; see status.description	
order	Yes	A host-generated order ID; Used for voids.	120100010000000063
sva	Yes	The masked version of the stored value account	XXXXXXXXXXX0001
sva.status	Yes	The status of the stored value account; see sva.status	ACTIVE
sva.registered	Yes	Identifies whether the account is registered or not; see sva.registered	true
sva.balances	Yes	The account's balances; see sva.balances	"USD 1000,Points 1"
sva.detailedBalances	JSON-only	The account's balances and progress towards rewards; see sva.detailedBalances	
rewards	No	A list of any rewards added to the account; see rewards	"Points 5"
notes	No	Rewards notes; see notes	"You have earned an extra $1 for loading $10 onto your account!"
owed	Yes	An amount still owed following the redemption. In cases of insufficient funds this will be non-zero; see owed	USD 278

4.4. Balance Inquiry

Description	Returns the balances of all currencies on a stored value account
Request Path	`archer/v1/inquiry`
Reversible	No
Voidable	No

4.4.1. Usage

Used to retrieve and display the balance(s) for each currency supported by a stored value account.

4.4.2. Request Parameters

Parameter	Required	Description	Example
domain	Yes	Identifies the domain; see domain	hps
chain	Maybe	Used by global partners to send requests on behalf of a group of merchants; see chain	130909013158178
request	Yes	Identifies this unique request; see request	201212312359590001
store	Yes	Identifies the store; see store	0001
terminal	Yes	Identifies the terminal; see terminal	001
terminal.version	Yes	Identifies the terminal software version; see terminal.version	Cyberdyne T-1000
clerk	No	Identifies an individual clerk or server; see clerk	17
sva	Yes	Identifies the stored value account; see sva	6277200000000001
acquired	Yes	Identifies how the sva was acquired; see acquired	SWIPE
pin	No	Specifies the PIN number associated with the account.	12345

4.4.3. Response Parameters

Parameter	Always Exists	Description	Example
status.code	Yes	Response status code; see status.code	200
status.name	Yes	Response status name; see status.name	AccountNotFound
status.description	Yes	Long, detailed description about the error; see status.description	
sva	Yes	The masked version of the stored value account	XXXXXXXXXXXX0001
sva.status	Yes	The status of the stored value account; see sva.status	ACTIVE
sva.registered	Yes	Identifies whether the account is registered or not; see sva.registered	true
sva.balances	Yes	The account's balances; see sva.balances	"USD 1000,Points 1"
sva.detailedBalances	JSON-only	The account's balances and progress towards rewards; see sva.detailedBalances	
rewards	No	A list of any rewards added to the account; see rewards	"Points 5"
notes	No	Rewards notes; see notes	"You have earned an extra $1 for loading $10 onto your account!"

4.5. Void

Description	Voids a prior transaction using the `order` parameter returned in its response
Request Path	`archer/v1/void`
Reversible	Yes
Voidable	No

4.5.1. Usage

Used to void (undo) a prior successful transaction. The original response from the host for the transaction to be voided contained a unique `order` parameter used to identify the transaction. This `order` parameter is specified to the host as part of the `void` request to identify which transaction to be voided.

When voiding a transaction, all changes to the account are reversed, including any additional value added by rewards programs or automated promotions.

Note: if a `void` transaction fails to return a complete response due to a network timeout condition, do not attempt to `reverse` the transaction. Instead, resend the same `void` transaction until you get a complete response back, whether the response indicates success or an error condition.

4.5.2. Request Parameters

Parameter	Required	Description	Example
domain	Yes	Identifies the domain; see domain	hps
chain	Maybe	Used by global partners to send requests on behalf of a group of merchants; see chain. Either store or chain must be supplied.	130909013158178
request	Yes	Identifies this unique request; see request	201212312359590001
store	Maybe	Identifies the store; see store. Either store or chain must be supplied.	0001
terminal	Yes	Identifies the terminal; see terminal	001
terminal.version	Yes	Identifies the terminal software version; see terminal.version	Cyberdyne T-1000
clerk	No	Identifies an individual clerk or server; see clerk	17
order	Yes	Identifies the transaction to void; see order	14000000000000000001

4.5.3. Response Parameters

Parameter	Always Exists	Description	Example
status.code	Yes	Response status code; see status.code	200
status.name	Yes	Response status name; see status.name	AccountNotFound
status.description	Yes	Long, detailed description about the error; see status.description	
order	Yes	A host-generated order ID; Used for voids.	120100010000000063
sva	Yes	The masked version of the stored value account	XXXXXXXXXXXX0001
sva.status	Yes	The status of the stored value account; see sva.status	ACTIVE
sva.registered	Yes	Identifies whether the account is registered or not; see sva.registered	true
sva.balances	Yes	The account's balances; see sva.balances	"USD 1000,Points 1"
sva.detailedBalances	JSON-only	The account's balances and progress towards rewards; see sva.detailedBalances	

notes	No	Rewards notes; see notes	"You have earned an extra $1 for loading $10 onto your account!"
void.action	Yes	The action voided; see void.action	redeem
void.amount	Yes	The amount voided; see amount	500
void.currency	Yes	The currency voided; see currency	USD

4.6. Reverse

Description	Reverses a prior transaction using the `request` parameter of the original request
Request Path	`archer/v1/reverse`
Reversible	No
Voidable	No

4.6.1. Usage

Used to reverse (undo) a prior transaction where a complete response was not received due to a network timeout. The `request` parameter of the original transaction is sent as the `reversal` parameter to identify the transaction to be reversed.

When reversing a transaction, all changes to the account are reversed, including any additional value added by rewards programs or automated promotions.

Note: if a `reverse` transaction fails to return a complete response due to a network timeout condition, continue sending the same `reverse` request until you get a complete response back.

4.6.2. Request Parameters

Parameter	Required	Description	Example
domain	Yes	Identifies the domain; see domain	hps
chain	Maybe	Used by global partners to send requests on behalf of a group of merchants; see chain. Either store or chain must be supplied.	130909013158178
request	Yes	Identifies this unique request; see request	201212312359590002
store	Maybe	Identifies the store; see store. Either store or chain must be supplied.	0001
terminal	Yes	Identifies the terminal; see terminal	001
terminal.version	Yes	Identifies the terminal software version; see terminal.version	Cyberdyne T-1000
clerk	No	Identifies an individual clerk or server; see clerk	17
reversal	Yes	Identifies the transaction to reverse; see reversal	201212312359590001

4.6.3. Response Parameters

Parameter	Always Exists	Description	Example
status.code	Yes	Response status code; see status.code	200
status.name	Yes	Response status name; see status.name	AccountNotFound
status.description	Yes	Long, detailed description about the error; see status.description	
reversal	Yes	Identifies the request ID of the transaction reversed; see reversal	201212312359590001

4.7. Transfer

Description	Transfers balances from one stored value account to another
Request Path	`archer/v1/transfer`
Reversible	No
Voidable	No

4.7.1. Usage

Transfers balances from one stored value account to another. This function is typically performed to replace a lost or stolen account with a new one or to consolidate two or more accounts into a single account.

After a transfer has completed, the source account will be closed and can no longer be used. The balance for the destination account will be returned in the response.

4.7.2. Request Parameters

Parameter	Required	Description	Example
domain	Yes	Identifies the domain; see domain	hps
chain	Maybe	Used by global partners to send requests on behalf of a group of merchants; see chain	130909013158178
request	Yes	Identifies this unique request; see request	201212312359590002
store	Yes	Identifies the store; see store	0001
terminal	Yes	Identifies the terminal; see terminal	001
terminal.version	Yes	Identifies the terminal software version; see terminal.version	Cyberdyne T-1000
clerk	No	Identifies an individual clerk or server; see clerk	17
from.sva	Yes	Identifies the stored value account to transfer from; see sva	6277200000000001
from.acquired	Yes	Identifies how the "from" sva was acquired; see acquired	MANUAL
from.pin	No	Specifies the PIN number associated with the "from" account.	12345
to.sva	Yes	Identifies the stored value account being transferred to; see sva	627720000000002
to.acquired	Yes	Identifies how the "to" sva was acquired; see acquired	SWIPE
to.pin	No	Specifies the PIN number associated with the "to" account.	12345

4.7.3. Response Parameters

Parameter	Always Exists	Description	Example
status.code	Yes	Response status code; see status.code	200
status.name	Yes	Response status name; see status.name	AccountNotFound
status.description	Yes	Long, detailed description about the error; see status.description	
sva	Yes	The masked version of the stored value account	XXXXXXXXXXXX0001
sva.status	Yes	The status of the stored value account; see sva.status	ACTIVE
sva.registered	Yes	Identifies whether the account is registered or not; see sva.registered	true
sva.balances	Yes	The account's balances; see sva.balances	"USD 1000,Points 1"

sva.detailedBalances	JSON-only	The account's balances and progress towards rewards; see sva.detailedBalances	

4.8. Reward

Description	Specifies a payment amount from some payment form other than from a stored value account for the purpose of earning rewards
Request Path	`archer/v1/reward`
Reversible	Yes
Voidable	Yes

4.8.1. Usage

When an account holder makes a payment using a payment form other than a stored value account (e.g. cash or credit card), the account holder may present their stored value account to earn Points or other loyalty rewards which would be added to their account. This may occur by swiping a loyalty card, entering an alias (e.g. a phone number) or scanning a mobile app (amongst other possibilities).

The purchase amount will *not* be deducted from the stored value account (as it would via a `redeem` transaction) or loaded onto the stored value account (as it would via a `load` transaction). Instead, the purchase amount is used to help determine what potential rewards may be added based on the merchant's loyalty and rewards program.

4.8.2. Request Parameters

Parameter	Required	Description	Example
domain	Yes	Identifies the domain; see domain	hps
chain	Maybe	Used by global partners to send requests on behalf of a group of merchants; see chain	130909013158178
request	Yes	Identifies this unique request; see request	201212312359590001
store	Yes	Identifies the store; see store	0001
terminal	Yes	Identifies the terminal; see terminal	001
terminal.order	Yes	A terminal-generated order number; see terminal.order	1701
terminal.version	Yes	Identifies the terminal software version; see terminal.version	Cyberdyne T-1000
clerk	No	Identifies an individual clerk or server; see clerk	17
sva	Yes	Identifies the stored value account being rewarded; see sva	6277200000000001
acquired	Yes	Identifies how the sva was acquired; see acquired	SWIPE
pin	No	Specifies the PIN number associated with the account.	12345
amount	Yes	The amount of the purchase; see amount	1000
currency	Yes	The currency of the purchase; see currency	USD
tax	No	The portion of the amount that represents tax; see tax	80
tip	No	The portion of the amount that represents a tip; see tip	200
exclude	No	The (non-tip, non-tax) portion of the amount that should be excluded from rewards; see exclude	1000

4.8.3. Response Parameters

Parameter	Always Exists	Description	Example
status.code	Yes	Response status code; see status.code	200
status.name	Yes	Response status name; see status.name	AccountNotFound

status.description	Yes	Long, detailed description about the error; see status.description	
order	Yes	A host-generated order ID; Used for voids.	120100010000000063
sva	Yes	The masked version of the stored value account	XXXXXXXXXXXX0001
sva.status	Yes	The status of the stored value account; see sva.status	ACTIVE
sva.registered	Yes	Identifies whether the account is registered or not; see sva.registered	true
sva.balances	Yes	The account's balances; see sva.balances	"USD 1000,Points 1"
sva.detailedBalances	JSON-only	The account's balances and progress towards rewards; see sva.detailedBalances	
rewards	No	A list of any rewards added to the account; see rewards	"Points 5"
notes	No	Rewards notes; see notes	"You have earned an extra $1 for loading $10 onto your account!"

4.9. Add Alias

Description	Adds an alias, such as a phone number, to an existing stored value account
Request Path	`archer/v1/alias/add`
Reversible	No
Voidable	No

4.9.1. Usage

This allows an account holder to add their phone number as an alias to a stored value account so that they may present their phone number to a merchant in place of a plastic card or other device.

4.9.2. Request Parameters

Parameter	Required	Description	Example
domain	Yes	Identifies the domain; see domain	hps
chain	Maybe	Used by global partners to send requests on behalf of a group of merchants; see chain	130909013158178
request	Yes	Identifies this unique request; see request	201212312359590001
store	Yes	Identifies the store; see store	0001
terminal	Yes	Identifies the terminal; see terminal	001
terminal.version	Yes	Identifies the terminal software version; see terminal.version	Cyberdyne T-1000
clerk	No	Identifies an individual clerk or server; see clerk	17
sva	Yes	Identifies the stored value account being aliased; see sva	6277200000000001
pin	No	Specifies the PIN number associated with the account.	12345
acquired	Yes	Identifies how the sva was acquired; see acquired	SWIPE
alias	Yes	The alias to add to the sva; see alias	5035551212

4.9.3. Response Parameters

Parameter	Always Exists	Description	Example
status.code	Yes	Response status code; see status.code	200
status.name	Yes	Response status name; see status.name	AccountNotFound
status.description	Yes	Long, detailed description about the error; see status.description	
sva	Yes	Identifies the account number that was aliased; see account	6277200000000001
alias	Yes	Identifies the alias that was added to the account; see alias	5035551212

4.10. Remove Alias

Description	Removes an alias, such as a phone number, from a stored value account
Request Path	`archer/v1/alias/remove`
Reversible	No
Voidable	No

4.10.1. Usage

This allows an account holder to remove their phone number as an alias from a stored value account. This is typically done because the account holder wants to change to a new phone number or because they want to associate their current phone number with a different stored value account.

4.10.2. Request Parameters

Parameter	Required	Description	Example
domain	Yes	Identifies the domain; see domain	hps
chain	Maybe	Used by global partners to send requests on behalf of a group of merchants; see chain	130909013158178
request	Yes	Identifies this unique request; see request	201212312359590001
store	Yes	Identifies the store; see store	0001
terminal	Yes	Identifies the terminal; see terminal	001
terminal.version	Yes	Identifies the terminal software version; see terminal.version	Cyberdyne T-1000
clerk	No	Identifies an individual clerk or server; see clerk	17
sva	Yes	Identifies the stored value account that is aliased; see sva	6277200000000001
pin	No	Specifies the PIN number associated with the account.	12345
acquired	Yes	Identifies how the sva was acquired; see acquired	SWIPE
alias	Yes	The alias to remove from the sva; see alias	5035551212

4.10.3. Response Parameters

Parameter	Always Exists	Description	Example
status.code	Yes	Response status code; see status.code	200
status.name	Yes	Response status name; see status.name	AccountNotFound
status.description	Yes	Long, detailed description about the error; see status.description	
sva	Yes	Identifies the account number that had the alias removed; see account	6277200000000001
alias	Yes	Identifies the alias that was removed from the account; see alias	5035551212

4.11. Create Alias

Description	Creates a new stored value account and associates an alias to it
Request Path	`archer/v1/alias/create`
Reversible	No
Voidable	No

4.11.1. Usage

This allows an account holder to add their phone number as an alias to a brand new stored value account. The new account does not have have a plastic card associated with it. This is typically referred to as a "cardless" account and can be used in cardless environments or by mobile applications where the mobile device serves as a virtual "card."

4.11.2. Request Parameters

Parameter	Required	Description	Example
domain	Yes	Identifies the domain; see domain	hps
chain	Maybe	Used by global partners to send requests on behalf of a group of merchants; see chain	130909013158178
request	Yes	Identifies this unique request; see request	201212312359590001
store	Yes	Identifies the store; see store	0001
terminal	Yes	Identifies the terminal; see terminal	001
terminal.version	Yes	Identifies the terminal software version; see terminal.version	Cyberdyne T-1000
clerk	No	Identifies an individual clerk or server; see clerk	17
alias	Yes	The alias to add to the new sva; see alias	5035551212

4.11.3. Response Parameters

Parameter	Always Exists	Description	Example
status.code	Yes	Response status code; see status.code	200
status.name	Yes	Response status name; see status.name	AccountNotFound
status.description	Yes	Long, detailed description about the error; see status.description	
sva	Yes	Identifies the account number that was created and aliased; see account	6277200000000001
alias	Yes	Identifies the alias that was added to the new account; see alias	5035551212
pin	Yes	Specifies the PIN number associated with the new account.	12345

4.12. Deactivate

Description	Deactivates an active stored value account
Request Path	`archer/v1/deactivate`
Reversible	Yes
Voidable	No

4.12.1. Usage

Used to undo an activation of a stored value account that otherwise has not been used. Deactivation resets the account so that it is brand new, with zero balance or activity. Attempts to deactivate an account that is not active or has had subsequent activity will fail.

4.12.2. Request Parameters

Parameter	Required	Description	Example
domain	Yes	Identifies the domain; see domain	hps
chain	Maybe	Used by global partners to send requests on behalf of a group of merchants; see chain. Either chain or store must be supplied.	130909013158178
request	Yes	Identifies this unique request; see request	201212312359590001
store	Maybe	Identifies the store; see store. Either chain or store must be supplied.	0001
terminal	Yes	Identifies the terminal; see terminal	001
terminal.order	Yes	A terminal-generated order number; see terminal.order	1701
terminal.version	Yes	Identifies the terminal software version; see terminal.version	Cyberdyne T-1000
clerk	No	Identifies an individual clerk or server; see clerk	17
sva	Yes	Identifies the stored value account being deactivated; see sva	6277200000000001
pin	No	Specifies the PIN number associated with the account.	12345
acquired	Yes	Identifies how the sva was acquired; see acquired	SWIPE

4.12.3. Response Parameters

Parameter	Always Exists	Description	Example
status.code	Yes	Response status code; see status.code	200
status.name	Yes	Response status name; see status.name	AccountNotFound
status.description	Yes	Long, detailed description about the error; see status.description	
sva	Yes	The masked version of the stored value account	XXXXXXXXXXXX0001
sva.status	Yes	The status of the stored value account; see sva.status	NEW
sva.registered	Yes	Identifies whether the account is registered or not; see sva.registered	false
sva.balances	Yes	The account's balances; see sva.balances	"USD 0,Points 0"
sva.detailedBalances	JSON-only	The account's balances and progress towards rewards; see sva.detailedBalances	

4.13. Greet

Description	Returns an account balance plus registered owner information.
Request Path	`archer/v1/greet`
Reversible	No
Voidable	No

4.13.1. Usage

Typically used in a restaurant scenario when a consumer first walks into the restaurant. This API will return an account balance, information about the registered owner of the card, register a "visit" event for reporting and also may cause promos to fire without there being a financial transaction involved.

4.13.2. Request Parameters

Parameter	Required	Description	Example
domain	Yes	Identifies the domain; see domain	hps
chain	Maybe	Used by global partners to send requests on behalf of a group of merchants; see chain	130909013158178
store	Yes	Identifies the store; see store	0001
terminal	Yes	Identifies the terminal; see terminal	001
terminal.version	Yes	Identifies the terminal software version; see terminal.version	Cyberdyne T-1000
clerk	No	Identifies an individual clerk or server; see clerk	17
sva	Yes	Identifies the stored value account being presented; see sva	6277200000000001
pin	No	Specifies the PIN number associated with the account.	12345
acquired	Yes	Identifies how the sva was acquired; see acquired	SWIPE

4.13.3. Response Parameters

Parameter	Always Exists	Description	Example
status.code	Yes	Response status code; see status.code	200
status.name	Yes	Response status name; see status.name	AccountNotFound
status.description	Yes	Long, detailed description about the error; see status.description	
sva	Yes	The masked version of the stored value account	XXXXXXXXXXXX0001
sva.balances	Yes	The account's balances; see sva.balances	"USD 1000,Points 1"
sva.detailedBalances	JSON-only	The account's balances and progress towards rewards; see sva.detailedBalances	
rewards	No	A list of any rewards added to the account; see rewards	"Points 5"
notes	No	Rewards notes; see notes	"You have earned an extra $1 for loading $10 onto your account!"
sva.status	Yes	The status of the stored value account; see sva.status	NEW
sva.registered	Yes	Identifies whether the account is registered or not; see sva.registered	false
sva.lastvisitstore	No	Last store at which this account was used.	“PITA3207”

sva.lastvisitdate	No	Last date this account was used.	"2014-03-17”
sva.dateregistered	No	Date on which this account was registered.	“2013-01-02”
sva.dateactivated	No	Date on which this account was first used.	“2014-03-12”
user.firstname	No	Consumers first name.	"John"
user.lastname	No	Consumers last name.	"Doe”
user.birthday	No	Day of the month of the consumers birthday.	"31”
user.birthmonth	No	Month of the year of the consumers birthday.	"11”
user.id.beanstalk	No	Consumer's ContactID if they are registered with Beanstalk.	"17139584”

4.14. Rewards Info

Description	Returns the details of a loyalty rewards program that can be communicated to a consumer.
Request Path	`archer/v1/rewardsinfo`
Reversible	No
Voidable	No

4.14.1. Usage

Typically used in a mobile app or by 3rd party developers to clearly communicate the details of a loyalty program to a consumer to promote usage and adoption. This API will return the program name, a short description, rewards structure (threshold and flavor), and URL to a full program description on the merchant's website.

4.14.2. Request Parameters

Parameter	Required	Description	Example
domain	Yes	Identifies the domain; see domain	hps
chain	Yes	Identifies the merchant; see chain	130909013158178
request	Yes	Identifies this unique request; see request	201212312359590001
terminal	Yes	Identifies the terminal; see terminal	001
terminal.version	Yes	Identifies the terminal software version; see terminal.version	Cyberdyne T-1000

4.14.3. Response Parameters

Parameter	Always Exists	Description	Example
status.code	Yes	Response status code; see status.code	200
status.name	Yes	Response status name; see status.name	AccountNotFound
status.description	Yes	Long, detailed description about the error; see status.description	
program.name	Yes	The name of the merchant's rewards program	Bob's rewards
program.description	Yes	A brief description of the rewards program that can be displayed inline	A fantastic rewards program brought to you by Bob
program.flavor	Yes	A hint for rendering the progress towards the next reward in a loyalty program; see flavor	VISIT SPEND LOAD PUNCH POINTS

program.threshold	Yes	Indicates the loyalty (non-tendered) balance amount needed to obtain a next reward in the loyalty program; see threshold	20
program.url	Yes	Link to full program details on the merchant's website	https://heartlandgiftcard.com/

4.14.4. Notes

Returns an error if no Rewards Info is available for the merchant.

4.15. Cashout

Description	Cashes out all remaining tendered currency from a stored value account, below a maximum amount.
Request Path	`archer/v1/cashout`
Reversible	Yes
Voidable	Yes

4.15.1. Usage

A cashout transaction removes the entire cash value of a stored value account, if it's below the given maximum amount. So a $10 cashout request would zero an account with a $9 balance (returning $9 to the consumer in cash). A $10 cashout request would fail and do nothing on an account with an $11 balance - the accounts balance is too high to cashout.

4.15.2. Request Parameters

Parameter	Required	Description	Example
domain	Yes	Identifies the domain; see domain	hps
chain	Maybe	Used by global partners to send requests on behalf of a group of merchants; see chain	130909013158178
request	Yes	Identifies this unique request; see request	201212312359590001
store	Yes	Identifies the store; see store	0001
terminal	Yes	Identifies the terminal; see terminal	001
terminal.order	Yes	A terminal-generated order number; see terminal.order	1701
terminal.version	Yes	Identifies the terminal software version; see terminal.version	Cyberdyne T-1000
clerk	No	Identifies an individual clerk or server; see clerk	17
sva	Yes	Identifies the stored value account being cashed-out; see sva	6277200000000001
acquired	Yes	Identifies how the sva was acquired; see acquired	SWIPE
amount	Yes	The maximum amount (the limit) to cashout. Any card with a balance below this amount will be zeroed. amount	1000
currency	Yes	The currency of the cashout amount; see currency	USD

4.15.3. Response Parameters

Parameter	Always Exists	Description	Example
status.code	Yes	Response status code; see status.code	200
status.name	Yes	Response status name; see status.name	AccountNotFound
status.description	Yes	Long, detailed description about the error; see status.description	
order	Yes	A host-generated order ID; Used for voids.	120100010000000063
sva	Yes	The masked version of the stored value account	XXXXXXXXXXXX0001

sva.status	Yes	The status of the stored value account; see sva.status	ACTIVE
sva.registered	Yes	Identifies whether the account is registered or not; see sva.registered	true
sva.balances	Yes	The account's balances; see sva.balances	"USD 1000,Points 1"
amount	Yes	The actual amount (balance) removed from the stored value account.	USD 800

5. Appendices

5.1. Appendix A - Parameter Reference

5.1.1. domain

Description	A Heartland-assigned identifier to distinguish processing environments.
Format	Alphanumeric
Max Length	16 chars
Examples	`hps`
Notes	Clients should make this a configurable parameter

5.1.2. chain

Description	A Heartland-assigned identifier to distinguish merchants
Format	Alphanumeric
Max Length	25

Examples	`301`
	`130909013158178`
	`DomainProgram`

Notes	Needed by global partners to send requests on behalf of a group of merchants

5.1.3. store

Description	Identifies a unique merchant location. This value may or may not be assigned by Heartland.
Format	Alphanumeric
Max Length	50 chars
Examples	`00001`
Notes	Clients should make this a configurable parameter

5.1.4. terminal

Description	Identifies a unique terminal within a store. This value may or may not be assigned by Heartland.
Format	Alphanumeric
Max Length	25 chars
Examples	`001`
Notes	Clients should make this a configurable parameter

5.1.5. request

Description	A client-generated token to uniquely identify an individual request. This token must be globally unique by store and terminal. That is, every request from an individual terminal in a store should have a unique request identifier or else the request will receive an error. In the event the client does not receive a response (timeout) or if the client receives a response with a status code of 5xx, this request identifier will be used to perform a timeout reversal.
Format	Alphanumeric
Max Length	50 chars
Examples	`20301231235959-0001`
Notes	To help ensure uniqueness, it is a good idea to utilize a timestamp as part of the request identifier.

5.1.6. sva

Description	Identifies the Stored Value Account. The value passed for this parameter can be a Stored Value Account Number, track data from the magnetic stripe on a swiped card, or an alias (such as a phone number) that references a Stored Value Account. Archer's intelligent parser knows how to automatically determine the type of SVA data presented and process the parameter correspondingly.
Format	Alphanumeric
Max Length	512

Examples	`6277200000000001`
	`%B6277200000000001^HEARTLAND PAYMENT SYSTE^37829821000123456789?`
	`B6277200000000001^HEARTLAND PAYMENT SYSTE^37829821000123456789`
	`;6277200000000001=391200012345?`
	`6277200000000001=391200012345`
	`5035551212`

Notes	

5.1.7. acquired

Description	Identifies how a Stored Value Account identifier was acquired. This parameter is used to control program behavior on the host.
Format	Enumerated set of alphanumeric values
Max Length	N/A
Examples	`SWIPE`
Notes	The complete set of values is provided in the section entitled Acquired Type Values.

5.1.8. amount

Description	Identifies the amount of a financial transaction. This parameter always expects an implied decimal point for the associated currency. For example, `1000` for the USD currency has an implied decimal point as if it were rendered as `10.00` representing ten dollars USD. The location of the implied decimal point is currency-specific.
Format	Numeric
Max Length	N/A
Examples	`1000`
Notes	

5.1.9. currency

Description	Identifies the currency of a financial transaction. Can be an ISO 4217 Currency Code identifier (a three letter code) such as `USD`, `CAD`, `MXN`, `EUR`, `GBP`, `AUD` or `NZD`, `Points`, or a coupon code.
Format	Alpha
Max Length	20
Examples	`USD` `Points`
Notes	

5.1.10. pin

Description	The numeric pin associated with a stored value account. The pin is typically only required when redeeming from an account using an alias such as a phone number.
Format	Numeric
Max Length	10
Examples	`123456`
Notes	Account pins are *typically* 4 to 8 digits long, but can be longer

5.1.11. tax

Description	The portion of the request's `amount` that represents tax, if applicable.
Format	Numeric
Max Length	N/A
Examples	`89`
Notes	

5.1.12. tip

Description	The portion of the request's `amount` that represents a tip, if applicable.
Format	Numeric
Max Length	N/A
Examples	`200`
Notes	

5.1.13. exclude

Description	The portion of the request's `amount` that should be excluded from rewards. Does *not* include tip or tax.
Format	Numeric
Max Length	N/A
Examples	`1000`
Notes	Some states require the exclusion of alcohol, tobacco, and certain other items.

5.1.14. order

Description	Identifies a transaction (aka order) that was successfully processed by the host platform. Every successfully processed transaction that changes the state of a stored value account that can be voided is assigned a unique `order` ID by the Heartland Stored Value Platform host.
Format	Numeric
Max Length	19
Examples	`140000000000000001`
Notes	

5.1.15. reversal

Description	Identifies a prior transaction that was not successfully processed because of a network timeout. The value specified in this field should be the value specified in the `request` field of the transaction to be reversed.
Format	Alphanumeric
Max Length	50
Examples	`20301231235959-0001`
Notes	

5.1.16. alias

Description	An alternate identifier used to reference a stored value account. The alias is typically a phone number in most merchant programs.
Format	Alphanumeric
Max Length	100
Examples	`5035551212`
Notes	

5.1.17. status.code

Description	The HTTP response code. Codes in the 2XX range represent success, codes in the 4XX range represent an application error, and codes in the 5XX range represent a system error. Every attempt is made to align the HTTP response code to our host errors as closely as possible.
Format	Numeric
Max Length	3

Examples	`200`
	`400`
	`500`

Notes	An exhaustive list of errors is provided in Appendix C - Errors

5.1.18. status.name

Description	A short, descriptive name for a specific error condition. More than one unique `status.name` may share the same `status.code`. For example, a large number of unique error conditions all share `status.code` `400`.
Format	Alphanumeric
Max Length	100

Examples	`Okay`
	`AccountNotFound`
	`InsufficientFunds`

Notes	An exhaustive list of errors is provided in Appendix C - Errors

5.1.19. status.description

Description	A long, descriptive name for a specific error condition. It is not intended to be machine parseable but is instead intended to be displayed to the end-user in some fashion, e.g. on a POS screen, on a receipt, on a webpage, etc.
Format	Alphanumeric
Max Length	256
Examples	`Unknown account [XXXXXXXXXXXX0001].` `The order [140000000000000001] has already been voided.`
Notes	

5.1.20. account

Description	Identifies a stored value account that was involved in an alias request.
Format	Numeric
Max Length	19
Examples	`6277200000000001`
Notes	

5.1.21. alias

Description	An alternate identifier used to reference a stored value account. The alias is typically a phone number in most merchant programs.
Format	Alphanumeric
Max Length	100
Examples	`5035551212`
Notes	

5.1.22. sva.status

Description	Identifies the status of a stored value account.
Format	Alphanumeric
Max Length	
Examples	`ACTIVE`
Notes	The complete set of values is provided in the section entitled Account Status Values.

5.1.23. sva.registered

Description	Identifies whether a stored value account is registered to a user profile or not.
Format	Boolean

Max Length	5
Examples	`true` `false`
Notes	

5.1.24. sva.balances

Description	Returns the balances for all currencies on the stored value account that are supported by a merchant. The balances are encoded as follows: `$currency $amount,$currency $amount` Each currency pair is separated by a comma, and the currency is separated from its amount by a single space.
Format	Alphanumeric
Max Length	100
Examples	`USD 1000,Points 7`
Notes	

5.1.25. sva.detailedBalances

Description	Returns the balances for all currencies on the stored value account that are supported by a merchant. Balances that are relevant to the rewards program will contain progress information towards that next reward.
Format	Array of JSON objects
Max Length	
Examples	`[ { "currency": "Points", "amount": 10, "threshold": 20, "flavor": "Visit"} ]`
Notes	This detailed information is only available in JSON responses.

5.1.25.1. currency

Description	Identifies the currency of a financial transaction. Can be an ISO 4217 Currency Code identifier (a three letter code) such as `USD`, `CAD`, `MXN`, `EUR`, `GBP`, `AUD` or `NZD`, `Points`, or a coupon code.
Format	Alpha
Max Length	20
Examples	`USD` `Points`
Notes	

5.1.25.2. amount

Description	Identifies the amount of a financial transaction. This parameter always expects an implied decimal point for the associated currency. For example, `1000` for the USD currency has an implied decimal point as if it were rendered as `10.00` representing ten dollars USD. The location of the implied decimal point is currency-specific.
Format	Numeric

Max Length	N/A
Examples	`1000`
Notes	

5.1.25.3. threshold

Description	Indicates the loyalty (non-tendered) balance amount needed to obtain the next reward in the loyalty program.
Format	Numeric
Max Length	N/A
Examples	20
Notes	Value is 0 for non-applicable currencies or when a progress hint is not applicable/available.

5.1.25.4. flavor

Description	Indicates a hint for rendering the progress towards the next reward in a loyalty program.
Format	Alphanumeric
Max Length	

Examples	`VISIT`
	`SPEND`
	`LOAD`
	`PUNCH`
	`POINTS`

Notes	Value is empty for non-applicable currencies. When present provides hints on how progress could be conveyed to the user, e.g. icons, progress bars, etc.

5.1.26. notes

Description	Contains rewards messages to be displayed on a receipt, mobile app, or web page to inform an account holder about special rewards or promotions that have executed on the account. The note text is simply a long, url-encoded string. The note text may contain zero or more instances of the two-character string "\n" (0x5C 0x6E). This is *not* a newline character (0x0A) but a backslash followed by the 'n' character. Well-written clients look for this two-character string and replace it with the appropriate line-break symbol for the target platform. For example, websites would turn those characters into a "` `" string while Unix clients would turn it into `'\n'` (0x0A) and Windows clients would turn it into `"\r\n"` (0x0D0A).
Format	Alphanumeric
Max Length	4096
Examples	`Congratulations! You have just earned $1\nfor loading $10 or more onto your account!`
Notes	

5.1.27. rewards

Description	An optional response parameter that identifies any currency-based rewards added to the account as a result of a transaction. The list of rewards are encoded as follows: `$currency $amount,$currency $amount` Each currency pair is separated by a comma, and the currency is separated from its reward amount by a single space.
Format	Alphanumeric

Max Length	100
Examples	`USD 100,Points 1`
Notes	

5.1.28. void.action

Description	Identifies the action of the transaction being voided, whether it was a `load`, `redeem`, etc.
Format	Alphanumeric
Max Length	100
Examples	`load` `redeem`
Notes	

5.1.29. terminal.order

Description	A terminal-generated order number. Used for reporting, to associate related transactions to each other (split tender, tip adjust, redeem/reward, etc.).
Format	Alphanumeric
Max Length	50
Examples	`0178` `09F`
Notes	

5.1.30. terminal.version

Description	The software version running on the terminal or POS.
Format	Alphanumeric
Max Length	50
Examples	`Cyberdyne T-1000` `HAL 9000`
Notes	

5.1.31. owed

Description	After a redemption, the amount still owed by the customer. Typically this will be `USD 0`, but in cases of insufficient funds it will be a non-zero amount. It will always be encoded as a `$currency $amount` pair.
Format	Alphanumeric
Max Length	100
Examples	`USD 278` `USD 0`
Notes	

5.1.32. partial

Description	Specifies whether the host should accept a partial redemption or not. A partial redemption is where the balance for an account is non-zero but insufficient to cover the full redemption amount. If partial redemption is enabled by setting the parameter value to true, the balance is drained and the amount owed is returned so that an additional payment in any acceptable form of tender may be accepted. If partial redemption is disabled, then the full amount must be covered by the balance or the entire transaction fails.
Format	Boolean
Max Length	N/A
Examples	`true` `false`
Notes	

5.1.33. clerk

Description	Identifies an individual clerk or server. Typically used on POS systems in which individual users must log in before processing transactions.
Format	Alphanumeric
Max Length	50
Examples	`17` `John Connor`
Notes	

5.2. Appendix B - Enumerated Values

5.2.1. Acquired Type Values

The complete set of supported Acquired Types are provided below and are case-sensitive.

Value	Description
`SWIPE`	Identifies acquisition by a magnetic stripe reader when "swiping" a card
`MANUAL`	Identifies acquisition by manually entering an account number via a number pad
`INTERNET`	Identifies acquisition directly from an account holder over the internet, such as through an e-commerce website
`TAP`	Identifies acquisition from an RFID or NFC tap
`BLUETOOTH`	Identifies acquisition via a bluetooth acquiring device
`SCAN`	Identifies acquisition from a 2D Barcode or QR Code scanning device
`WIRELESS`	Identifies acquisition from a wireless device not previously specified
`VOICE`	Identifies acquisition directly from an account holder over the phone
`OTHER`	Identifies acquisition from any other means not previously specified

5.2.2. Account Status Values

The complete set of supported Account Status values are provided below and are case-sensitive.

Value	Description
`NEW`	Identifies a new account that has no balance and is available for purchase
`ACTIVE`	Identifies an active account that has been activated and used
`FROZEN`	Identifies an account that is disabled from further use until unfrozen (typically because of suspicion of fraud)
`NEWFROZEN`	Identifies an account that is disabled from further use until unfrozen and returned to `NEW` status
`CLOSED`	Identifies an account with no balance that is permanently no longer available for use

5.3. Appendix C - Errors

5.3.1. HTTP errors

Standard HTTP errors, see HTTP spec for others. All 5xx error codes must be reversed (automatic with the Heartland-provided archer-client jar).

Status Code	Status Name
401	Unauthorized
403	Forbidden
404	NotFound
500	InternalServerError
503	ServiceUnavailable

5.3.2. Application errors

A collection of application and system errors that may be returned by the Archer host.

Status Code	Status Name
400	AccountCreationNotAllowed
400	AccountNotActivated
400	AccountNotFound
400	AccountOrderMismatch
400	ActivateAccountMaxCountExceeded
400	ActivateNotAllowed
400	AddValueMaxAmountExceeded
400	AddValueMaxCountExceeded
400	AddValueNotAllowed
400	AliasNotAvailable
400	AlreadyVoided
400	ApiError
400	AppAlert
400	BalanceLimitExceeded
400	BalanceNotApplicable
400	BankCardNotFound
400	BatchAlreadyReleased
400	BulkOperationNotAllowed
400	BulkOperationPending
400	BulkResolveTipLimitExceeded
400	CannotCashout
400	CannotReprocessOrder
400	CardAuthorizationFailed
400	CloseNotAllowedForUser

400	CurrencyConverterNotFound
400	CurrencyFactorOutOfRange
400	DeactivateNotAllowed
400	DebitCardVoidAuthorizationMissing
400	DereferenceFailureException
400	ForcedOrderFailure
400	ForcedOrderFailureAlert
400	FreezeNotAllowedForUser
400	GenericForcedOrderFailure
400	GroupNotFound
400	InsufficientActivationAmount
400	InsufficientFunds
400	InsufficientLoadAmount
400	InsufficientValueInStore
400	InvalidActivationAmount
400	InvalidBusinessDayBoundary
400	InvalidCardNumber
400	InvalidCardVerification
400	InvalidChainName
400	InvalidDomainName
400	InvalidExpiration
400	InvalidGroupName
400	InvalidLastFour
400	InvalidOptInType
400	InvalidPassword
400	InvalidPaymentType
400	InvalidPin
400	InvalidProgram
400	InvalidRange
400	InvalidSellerProfileId
400	InvalidTokenAction
400	LineItemNotFound
400	LoadCardWithCardDisallowed
400	MakePaymentMaxCountExceeded
400	ManagedThreadsException
400	MaxRequestDurationExceeded
400	MergeAccountStatusNotAllowed
400	MissingCvv2
400	MissingData

400	MissingOrderProcessingAccount
400	MissingSellerProfileId
400	NegativeAddValue
400	NoAccountForProfile
400	NoAccountsInRange
400	NoAssociatedCardAccount
400	NonPositiveStoredValuePayment
400	NonUniqueLineNumbers
400	ObjectNotFound
400	OrderEditFailed
400	OrderExists
400	OrderNotFound
400	PaymentTypeLimitExceeded
400	PinRequiredForRedemption
400	ProfileAlreadyExists
403	ProfileAuthorizationFailed
403	ProfileClosed
403	ProfileFrozen
403	ProfileNotFound
400	ProfileReferenceNotFound
400	ProgramDisallowsTransfer
400	ProgramDisallowsTransferToExistingAccount
400	ProgramMismatch
400	ProgramNotFound
400	PromoProgramChangeNotAllowed
400	PromoProgramExpired
400	RangeNotFound
400	RedemptionAcquiredPolicyNotMet
400	ReferenceUnavailable
400	RegistrationMismatch
400	RegistrationNotAllowed
400	RegistrationRequired
400	RegistrationRequiredToMerge
400	RegistrationRequiredToTransfer
400	RegistrationStatusNotAllowed
400	RequestNotFound
400	StateChangeNotAllowed
400	SvaRefParseException
400	TipBelowAmount

400	TokenExpired
400	TokenNotFound
400	TooLateToDeactivate
400	TooLateToVoid
400	TooManyOrderProcessingAccounts
400	TooManyPromosForProgram
400	TooManyStoredValuePayments
400	TooOldToVoid
400	TransferAccountStatusNotAllowed
400	TransferFrozen
400	TransferToSameAccountNotAllowed
400	UnableToCreateAccountReference
400	UnequalChargesAndPayments
400	UnfreezeNotAllowed
400	UnrecognizedCreditCardNumber
400	UnrecognizedCreditCardType
400	UnresolvableStore
400	ValueLimitExceeded
400	ValueStoreMismatch
400	ValueStoreNotActive
400	ValueStoreNotSupported
400	VelocityLimitExceeded

5.4. Appendix D - Certification Host Response Matrix

The Archer Certification Host provides a way to force responses based on user input, typically an `amount` or `sva`. This allows a client developer to test various transaction scenarios by simply using well-chosen input values.

5.4.1. Amount Response Matrix

Responses to `activate`, `load`, `redeem`, and `reward` requests can be controlled by the `amount` parameter.

All whole dollar amounts (e.g. `100`, `200`, `1000`, etc) will return a status code of `200` and status name of `Okay`. All non-whole dollar amounts (any amount that does not end in "00") will return some form of an error response (status code of 4XX or 5XX). The request amounts enumerated in the table below will cause the corresponding error response to be returned. These request amounts will return the corresponding response for all currencies, including Points.

Amount	Status Code	Status Name
101	503	ServiceUnavailable
201	403	ProfileAuthorizationFailed
202	403	ProfileClosed
203	403	ProfileNotFound
204	403	ProfileFrozen
301	400	InsufficientFunds
302	400	InsufficientActivationAmount
303	400	InsufficientLoadAmount
304	400	InvalidPaymentType
305	400	InvalidPin
306	400	InvalidSellerProfileId
307	400	OrderExists
308	400	RegistrationRequired
309	400	AccountNotActivated
500	200	OK (However, returned order.id will be "TooLateToVoid"
600	200	OK (However, returned order.id will be "OrderNotFound"

5.5. Appendix E - Certification Host Stored Value Accounts

All account numbers in the following ranges:

Start of Range	End of Range
`6277200000000001`	`6277200000000099`
`5022440000000000001`	`5022440000000000099`

All aliases (phone numbers) in the following ranges:

Start of Range	End of Range
`XXX5550100`	`XXX5550199`

You may use whatever area code (NPA) you would like, but the exchange (NXX) must be `555` and the line must be in the range `0100-0199` or the host will reject the alias with an `AccountNotFound` error.

5.6. Appendix F - Code Samples

This section provides some simple examples of using the Archer API to do a `load` transaction in a variety of languages.

5.6.1. Curl

```
curl https://api.cert.chockstone.com/archer/v1/load \
    -iu test_user:test_password \
    -X POST \
    -d domain=test \
    -d request=201212312359590001 \
    -d store=0001 \
    -d terminal=0001 \
    -d sva=6277200000000001 \
    -d acquired=MANUAL \
    -d amount=1000 \
    -d currency=USD
```

Note: In practice, parameters should be url-encoded, so the `--data-urlencode` argument should be used in place of `-d`. The version above was used to make the sample easier to read.

5.6.2. Java

The following Java code snippet demonstrates how easy it is to integrate with the Archer platform using the Heartland-provided archer-client.

```
import com.hps.archer.util.StatusException;
import com.hps.archer.v1.client.ArcherClient;
import com.hps.archer.v1.client.ArcherClient.Acquired;

import java.util.Map;
import java.util.List;

import java.net.MalformedURLException;

...

String tmpDir = "/var/tmp/archer-client";

String endpoint = "https://api.cert.chockstone.com";
String user = "test_user";
String password = "test_password";
String domain = "test";
String store = "1000";
String terminal = "001";
String terminalVersion = "testTerminal";
int terminalOrder = 0;

ArcherClient client = null;
try {
    client = new ArcherClient(tmpDir, endpoint, user,
        password, domain, store, terminal, terminalVersion);
} catch(MalformedURLException e) {
    // handle error
}

Map<String, String> response = null;
Map<String, Integer> balances = null;
Map<String, Integer> rewards = null;
List<String> notes = null;
try {
    // load $10.00 USD onto account 6277200000000001
    response = client.load("6277200000000001", Acquired.MANUAL, "USD",
        1000, String.valueOf(++terminalOrder));

    balances = ArcherClient.getBalances(response);
    rewards = ArcherClient.getRewards(response);
    notes = ArcherClient.getNotes(response);

    // print balances
    System.out.println("Balances:");
    for (String currency : balances.keySet()) {
        System.out.println(currency + ": " + balances.get(currency));
    }

    // print rewards
    System.out.println("Rewards:");
    for (String currency : rewards.keySet()) {
        System.out.println(currency + ": " + rewards.get(currency));
    }

    // print notes
    System.out.println("Notes:");
    for (String note : notes) {
        System.out.println(note);
    }

} catch (StatusException e) {
    // handle error
}

// print out all response data
for (String key : response.keySet()) {
    System.out.println(key + " = " + response.get(key));
}
```

5.7. Appendix G - HTTP headers

5.7.1. Content-type

Description	Specifies the encoding type used in an Archer *request*.
Format	Alphanumeric
Max Length	N/A
Examples	`application/x-www-form-urlencoded` `application/json`
Notes	If not specified, `application/x-www-form-urlencoded` is assumed.

5.7.2. Accept

Description	Specifies the encoding type desired in an Archer *response*.
Format	Alphanumeric
Max Length	N/A
Examples	`application/x-www-form-urlencoded` `application/json`
Notes	If not specified, `application/x-www-form-urlencoded` will be used.