

PDE - Abacus JSON Specification Contract Migration

Draft: Version 1.0
Author: John Getter

Introduction:

The purpose of this document is to explain the JSON specification used to migrate account, contract, and payment information from an external schema into the abacus contract, account, and payment schema. If the resultant JSON file meets the requirements within this specification the data will be imported into abacus without error and hopefully without bad data. It is up to the party writing the JSON to meet this specification with the most consistent and up-to-date data, and it is expected that the data is without error from the external system. Abacus and the importer can not validate contract terms, contract dates, reserve calculations, payment information, etc. other than ensuring the data meets the required types.

JSON Specification (ChatGPT):

JSON, short for JavaScript Object Notation, is a lightweight data interchange format that is easy for humans to read and write, and easy for machines to parse and generate. It is often used to transmit data between a server and a web application, as well as to store configuration information.

The JSON specification defines a syntax for representing structured data, including:

1. **Objects:** An unordered collection of key-value pairs. Each key is a string, and each value can be a string, number, boolean, object, array, or null. Objects are enclosed in curly braces `{ }`.
2. **Arrays:** An ordered list of values. Values can be strings, numbers, booleans, objects, arrays, or null. Arrays are enclosed in square brackets `[ ]`.
3. **Strings:** A sequence of characters enclosed in double quotes `" "`. Unicode characters can be represented using escape sequences `(\uXXXX)`.
4. **Numbers:** A sequence of digits that may include a decimal point, an optional exponent, and a leading minus sign.
5. **Boolean values:** Either `true` or `false`.
6. **Reference:** `$ref` elements refer to a definition within a nested object
7. **Null value:** Represented by the keyword `null`.

Guidelines To The Specification:

These are implied uses of the types within this specification.

- **Objects:** `$ref` - elements with a reference tag have nested objects/arrays that can be found towards the top of the specification within the `definitions` object.
- **Arrays:** with `minItems = 0` are for all inclusive arrays. For instance, WW can be `[]` an empty array to represent all countries for WW.
- **Strings:** with a `minLength = 1` require at least one char in the string, but they can be `null` if a string isn't set or relevant to the external schema.
- **Numbers** are considered decimal or floats and **Integers** are not decimal or floats
- **Required** most if not all elements are required within this specification. However, most of them also accept `null` for input.

Code will be highlighted below using `Courier` to clarify the description vs JSON.

The following description will start with root object/elements to be followed by the `Definitions` object for referenced nested objects.

JSON Object Description:

Root Properties Elements

Name	Type	Definition	Sample
account_id	integer	The account id that will be associated with the contract and payments	21989
account_currency	string (180)	The currency that account wants to be paid in	EUR
account_status	string (180)	The status of the account	signed
account_name	string (180)	The name for the account. This can not be <code>null</code> and it must contain at least one char.	Thirty Tigers
contract_id	integer	The contract id for the contract associated with the account. If set to -1 a contract id will be assigned to this contract by abacus.	-1
contract_name	string (180)	The name for the contract. This can be <code>null</code> but it must contain at least one char if set.	Contract for Thirty Tigers
contract_terms	array ref	See definition: <code>ContractTerm</code>	
term_start	date	The start date of the contract that can be <code>null</code> . *	2009-09-15
term_end	date	The end date of the contract that can be <code>null</code> . *	2029-12-31
payment_minimum	number	The minimum of revenue met to get paid	1000
payment_schedule	string (180)	Days after the royalty calculation that the account must be paid	30_days_after_month_end
reserves	object ref	See definition: <code>Reserves</code>	

* One of term_start or term_end must be set for the contract

Definitions Elements

Name	Type	Definition	Sample
ContractTerm	object	Nested object of contract terms	
contract_term_name	string (180)	The name for the contract. This can be <code>null</code> but it must contain at least one char if set.	Term 1 for Contract for Thirty Tigers
attachments	array	Array of IDs attached to the contract term - see term types. If type label this would be a label ID or UPC for a product type.	["UPC1223456789"]
attachments_relations	object	Parent label ID for product or track terms	{"label_ids": ["21989"]}
term_type	string	The term type: label, product or track of the contract term. These types determine the <code>attachments</code> and <code>relations</code> to the attachments	product
term_conditions	array ref	See definition: <code>ContractTermConditions</code>	
ContractTermConditions	object	Nest object of contract term conditions	
conditions	object	JSON object of conditions	{"stores": [], "countries": [], "transaction_types": []}
countries	array ref	Country conditions that might have different term rates. See definition: <code>Countries</code>	

stores	array	Stores that might have different term rates.	[]
transaction_types	array ref	Transaction type conditions that might have different term rates. See definition: <code>TransactionType</code>	
contract_term_condition_name	string (180)	The name for the contract term condition. This can be <code>null</code> but it must contain at least one char if set.	"US only"
term_rate	number	The term rate for the label's share of the revenue	87
commission	number	The commission for PDE's share of the revenue	13
priority	integer	The priority of the conditions. With multiple conditions the ascending order is how a sale is matched i.e. priority 1 is matched over priority 2	1
Reserves	object	Nested object of reserves	
reserve_conditions	object	Transaction types to calc reserves on i.e. PS or physical sales. See definition: <code>TransactionType</code>	"transaction_types": ["PS", "RE"]
reserve_installments_in_months	integer	Number of months to release reserves taken.	4
reserve_release_offset_in_months	integer	Number of months before reserves are released	3

reserve_rate	number	Rate of revenue taken for reserves	30.00
reserve_release_schedule	array	Scheduled release rates: 1 / installment of months	[0.167, 0.167, 0.1670, 0.167, 0.167, 0.167]
TransactionType	array	Array of transaction types supported by abacus. Any external transaction types will have to be mapped to these types - [] means all transaction types.	[]
Countries	array	Array of 3 char country codes abacus supports. Any countries that are not in this list from the external schema will have to be mapped to this list.	["USA", "DEU"]

Reference Transaction Type Table

Code	Transaction Type Name
AC	Ad-Supported Video Streaming (Direct - At Rate Card)
AEA	Ad-Enabled Audio Streams
AEV	Ad-Enabled Video Streams
AL	Airlines
AO	Ad-Supported Video Streaming (Direct - Over Rate Card)
AP	Ad-Supported Audio Publishing
AS	Ad-Supported Audio Streams
AV	Ad-Supported Video Streams
BR	Brand Partnership Revenue
CA	Audio UGC Matches
CAV	AudioVisual UGC Matches
CC	Cable Radio
CE	cEST
CL	Cloud Match Units
CR	cVOD
CV	Visual UGC Matches
CX	Copyright Term Extension
DA	Download Albums
DB	Download Bonus
DJ	DJ Mix Streams
DR	Downloaded Ringtones
DT	Download Tracks
DV	Video Download Purchases
ED	Educational
EH	Ephemeral
EQ	Equity
ER	Exclusive Rights
FA	Fitness Subscription Audio Streams
FS	Fraudulent Streams
FT	Free TV
LP	Physical Sales via Licensee

LR	Physical Returns via Licensee
MA	Making Available / Streaming
MD	Music Dubbing
ME	Metadata Revenue
MP	Physical Sales - Compilations
MR	Physical Returns - Compilations
MT	Mid-Tier Subscription Audio Streams
MV	Mobile Videos
NC	Non-Controlled Rights Owner Income
NR	Non-interactive Radio
NS	Non-Interactive Satellite Radio
NT	Non-Theatrical
OP	Physical Sales - Direct
OR	Physical Returns - Direct
PA	Private Copying - Audio
PF	Paid Features
PH	Physical Video
PM	Pure Merch
PP	Public Performance
PR	Pre Cut Ringtones
PS	Physical Sales
PV	Private Copying - Audiovisual
PX	Audiovisual
RB	Ringbacks
RC	Performance Rights
RD	Download Rentals
RE	Physical Returns
RL	Rental & Lending
RR	Radio Broadcast
RV	Video Ringtones
RW	Internet Radio and Webcasting
S	Subscription Audio Streams
SA	Spatial Subscription Audio Streams
SB	Streaming Bonus
SC	Sync Revenue

SE	Segments
SF	Short-form Video User Generated Revenue
SL	Satellite Radio
SP	Subscription Audio Publishing
SR	Sync Revenue
SV	Subscription Video Streams
TA	Tethered Albums
TD	Tethered Downloads
TH	Theatrical
TL	Term Licenses
TP	Physical Sales - D2C
TR	Physical Returns - D2C
TR	Trailers
TV	TV Broadcast
UA	Upgraded Download Albums
UN	Unidentified Public Performance and Broadcast
UP	Unidentified Private Copying
US	Ad-Disabled Audio Streams
UT	Upgraded Download Tracks
UV	Upgraded Download Videos
VA	Ad-Supported Video Streams via App
VB	Video Bundles
VP	Subscription Video Publishing
VR	Video Rentals
VS	Video Seasons
VU	Ad-Disabled Video Streams
WA	Spoken Word Content Deduction for Ad-Supported Audio Streams
WS	Spoken Word Content Deduction for Subscription Audio Streams
YP	Ad-Supported Video Publishing