

# RESERVEDTICKET

Taxonomy Formula:  $tN\{\sim d, t, g, SC\}$

## Token Specification Summary

### Token Classification

<b>Template Type:</b>	SingleToken	This token has no sub or child tokens.
<b>Token Type:</b>	NonFungible	This token is not interchangeable with other tokens of the same type as they have different values.
<b>Token Unit:</b>	Whole	There can be many instances of this token, but they cannot be subdivided.
<b>Value Type:</b>	Intrinsic	This token is purely a digital token represents value directly, it represents no external physical form and cannot be a receipt or title for a material item or property.
<b>Representation Type:</b>	Common	This token is simply represented as a balance or quantity attributed to an owners address where all the balances are recorded on the same balance sheet, like a bank account. All instances can easily share common properties and locating them is simple.

*This is a Variable Supply Whole Non-Fungible where the total supply can vary. The tokens in this class will be of the same series, sharing those properties, but also have unique values like seat number. It is Whole by setting the Decimals property on the subdividable behavior = 0. A token instance can be burned.*

### Example

This token can be used to represent a unique item in a shared context, like a reserved seat at a concert.

### Analogies

Name	Description
<b>Reserved Seating</b>	All tokens in the class share the venue, date and time for the event, but have a

	unique seat number.
--	---------------------

## Comments

ReservedTicket is:

- Non-Subdividable
- Transferable
- Delegable

## ReservedTicket Details

Base: Whole Non-Fungible Token

<b>Type:</b>	Base
<b>Name:</b>	Whole Non-Fungible Token
<b>Id:</b>	3c05a856-c901-4c30-917e-df9feed1c8de
<b>Visual:</b>	$\tau_{N\{d\}}$
<b>Tooling:</b>	tN{~d}
<b>Version:</b>	1.0

## Definition

*Every non-fungible token is unique. A non-fungible token is not interchangeable with other tokens of the same class but have some shared properties while also having unique property values between instances. These tokens are whole tokens and can have quantities greater than 1 and also could support variable supply.*

## Example

CryptoKitties, Art, Reserved Seat for an event.

## Analogies

Name	Description
<b>Property Title</b>	The physical property title, land for example, have the identical look and feel from the paper, colors and seal. The difference between them are the values like property address, plot numbers, etc. These values make the title unique. There

	are some properties on a class of titles that are the same, like the county or jurisdiction the property is in. For titles that have some shared values and unique values, it may make more sense to define them in the same class.
<b>Art</b>	The valuable painting or other unique piece of art may not share any property values with other paintings, unless the artist is extremely prolific in generating tens of thousands of pieces of art, it would make sense to define each piece of art as its own class. Meaning there would be only a single piece of art represented by the token class. If the art cannot be sub-divided, meaning there can be no fractional owners, this token class can be a singleton if the quantity in the class is set to 1. A singleton has only one instance in the class, essentially meaning the class is the instance, and not be sub-dividable and no new tokens can be minted in the class.

## Comments

Non-fungible tokens require additional thought about how these tokens may or may not be grouped together in the same class.

## Dependencies

Artifact Type	Symbol	Description
<b>Base</b>	t	Base Token Definition

## Incompatible With

Artifact Type	Symbol	Id
<b>Behavior</b>	d	6e3501dc-5800-4c71-b59e-ad11418a998c

## Influenced By

Description	Symbol	Applies To
-------------	--------	------------

## Artifact Files

Content Type	File Name	File Content
--------------	-----------	--------------

Other	.DS_Store	
-------	-----------	--

## Code Map

Map Type	Name	Platform	Location
----------	------	----------	----------

## Implementation Map

Map Type	Name	Platform	Location
----------	------	----------	----------

## Resource Map

Map Type	Name	Location	Description
----------	------	----------	-------------

## Base Details

Token Name:	
Token Type:	NonFungible
Representation Type:	Common
Value Type:	Intrinsic
Token Unit:	Whole
Symbol:	
Owner:	
Quantity:	0
Decimals:	0
Constructor Name:	Constructor

## Behaviors

### Base: Non-Subdividable

Type:	Behavior
-------	----------

<b>Name:</b>	Non-Subdividable
<b>Id:</b>	d5807a8e-879b-4885-95fa-f09ba2a22172
<b>Visual:</b>	<i>~d</i>
<b>Tooling:</b>	~d
<b>Version:</b>	1.0

## Definition

*An ability or restriction on the token where it cannot be subdivided from a single whole token into fractions. Sets the base token Decimals property to 0 which will make the token non-sub-dividable and a whole token is the smallest ownable unit of the token.*

## Example

Non-subdividable is common for items where subdivision does not make sense, like a property title, inventory item or invoice.

## Analogies

Name	Description
<b>Non-Fractional</b>	It is not possible to own a fraction of this token.
<b>Barrel of Oil</b>	Barrels of Oil don't make sense to subdivide.

## Dependencies

Artifact Type	Symbol	Description
---------------	--------	-------------

## Incompatible With

Artifact Type	Symbol	Id
<b>Behavior</b>	d	6e3501dc-5800-4c71-b59e-ad11418a998c

## Influenced By

Description	Symbol	Applies To
-------------	--------	------------

## Artifact Files

Content Type	File Name	File Content
Control	non-subdividable.proto	
Uml	non-subdividable.md	
Other	.DS_Store	

## Code Map

Map Type	Name	Platform	Location
SourceCode	Code 1	Daml	

## Implementation Map

Map Type	Name	Platform	Location
Implementation	Implementation 1	ChaincodeGo	

## Resource Map

Map Type	Name	Location	Description
Resource	Regulation Reference 1		

## Specification Behavior

### Non-Subdividable

Taxonomy Formula: ~d

*An ability or restriction on the token where it cannot be subdivided from a single whole token into fractions. Sets the base token Decimals property to 0 which will*

*make the token non-sub-dividable and a whole token is the smallest ownable unit of the token.*

## Example

Non-subdividable is common for items where subdivision does not make sense, like a property title, inventory item or invoice.

## Analogies

Name	Description
<b>Non-Fractional</b>	It is not possible to own a fraction of this token.
<b>Barrel of Oil</b>	Barrels of Oil don't make sense to subdivide.

## Comments

<b>Is External:</b>	True
<b>Constructor:</b>	

## Non-Subdividable responds to these Invocations

### Properties

*Name: Decimals*

Value Description: Set to Zero, not allowing any subdivision, usually this is applied to the base token.

Template Value: 0

### Invocations

*GetDecimals*

Id: 2ca7fbb2-ce98-4dda-a6ae-e4ac2527bb33

Description: Should return 0

### Request

Control Message: GetDecimalsRequest

Description:

Parameters

Name	Value
------	-------

## Response

Control Message: GetDecimalsResponse

Description: Return 0

### Parameters

Name	Value
Decimals	0

## *GetDecimals*

Id: 2ca7fbb2-ce98-4dda-a6ae-e4ac2527bb33

Description: Should return 0

## Request

Control Message: GetDecimalsRequest

Description:

### Parameters

Name	Value
------	-------

## Response

Control Message: GetDecimalsResponse

Description: Return 0

### Parameters

Name	Value
Decimals	0

## Properties

### Base: Transferable

Type:	Behavior
-------	----------



<b>Name:</b>	Transferable
<b>Id:</b>	af119e58-6d84-4ca6-9656-75e8d312f038
<b>Visual:</b>	<i>t</i>
<b>Tooling:</b>	t
<b>Version:</b>	1.0

## Definition

*Every token instance has an owner. The Transferable behavior provides the owner the ability to transfer the ownership to another party or account. This behavior is often inferred by other behaviors that might exist like Redeem, Sell, etc. This behavior is Delegable. If the token definition is Delegable, TransferFrom will be available.*

## Example

## Analogies

Name	Description
<b>Analogy 1</b>	transferable analogy 1 description

## Dependencies

Artifact Type	Symbol	Description
---------------	--------	-------------

## Incompatible With

Artifact Type	Symbol	Id
<b>Behavior</b>	~t	a4fa4ca8-6afd-452b-91f5-7103b6fee5e5

## Influenced By

Description	Symbol	Applies To
<b>If the token is Delegable, TransferFrom should be enabled.</b>	g	[ ]

If Compliance is present, a CheckTransferAllowed request has to be made and verified before a Transfer request or a TransferFrom request.	c	[]
---	---	----

## Artifact Files

Content Type	File Name	File Content
<b>Control</b>	transferable.proto	
<b>Uml</b>	transferable.md	
<b>Other</b>	.DS_Store	

## Code Map

Map Type	Name	Platform	Location
<b>SourceCode</b>	Code 1	Daml	

## Implementation Map

Map Type	Name	Platform	Location
<b>Implementation</b>	Implementation 1	ChaincodeGo	

## Resource Map

Map Type	Name	Location	Description
<b>Resource</b>	Regulation Reference 1		

## Specification Behavior

### Transferable

#### Taxonomy Formula: t

*Every token instance has an owner. The Transferable behavior provides the owner the ability to transfer the ownership to another party or account. This behavior is often inferred by other behaviors that might exist like Redeem, Sell, etc. This behavior is Delegable. If the token definition is Delegable, TransferFrom will be available.*

#### Example

#### Analogies

Name	Description
<b>Analogy 1</b>	transferable analogy 1 description

#### Comments

<b>Is External:</b>	True
<b>Constructor:</b>	

#### Transferable responds to these Invocations

##### *Transfer*

Id: 5d4b8f10-7857-4a2f-9b8c-d61e367a6bcc

Description: >A transfer request will invoke a transfer from the owner of the token to the party or account provided in the To field of the request. For fungible or subdividable non-fungible tokens, this request may also include value in the Amount field of the request to transfer more than one token of the class in a single request.

#### Request Message:

TransferRequest

Description: The request

### Request Parameters

Name	Value
<b>To</b>	AccountId to transfer ownership to.
<b>Quantity</b>	Number of tokens to transfer.

### Response Message

TransferResponse

Description: The response

### Response Parameters

Name	Value
<b>Confirmation</b>	A confirmation receipt or error may be returned to the owner based on the outcome of the transfer request.

### *TransferFrom*

Id: 516b4e2f-4a14-4c4f-a6f2-1419d4af35c6

Description: >A transfer request will invoke a transfer from the owner of the token to the party or account provided in the To field of the request. For fungible or subdividable non-fungible tokens, this request may also include value in the Amount field of the request to transfer more than one token of the class in a single request.

### Request Message:

TransferFromRequest

Description: The request

### Request Parameters

Name	Value
<b>From</b>	AccountId to transfer ownership from.
<b>To</b>	AccountId to transfer ownership to.
<b>Quantity</b>	Number of tokens to transfer.

## Response Message

TransferFromResponse

Description: The response

### Response Parameters

Name	Value
Confirmation	A confirmation receipt or error may be returned to the owner based on the outcome of the transfer from request.

## Properties

### Base: Delegable

Type:	Behavior
Name:	Delegable
Id:	a3d02076-6009-4a65-9ed4-2deffe5291e1
Visual:	<i>g</i>
Tooling:	g
Version:	1.0

## Definition

*A token class that implements this behavior will support the delegation of certain behaviors to another party or account to invoke them on the behalf of the owner. When applied to a token, behaviors that are Delegable will enable delegated request invocations. This is useful to provide another party to automatically be able to perform the behaviors that can be delegated without seeking permission up to a certain allowance.*

## Example

## Analogies

Name	Description
<b>Broker</b>	You may allow a broker to transfer your tokens as a part of an investment strategy. Setting an allowance can cap the total number of tokens the broker is allowed to perform delegated behaviors, when exceeded a new allowance request will need to be granted.

## Comments

Applied to behaviors that are Delegable.

## Dependencies

Artifact Type	Symbol	Description
---------------	--------	-------------

## Incompatible With

Artifact Type	Symbol	Id
---------------	--------	----

## Influenced By

Description	Symbol	Applies To
-------------	--------	------------

## Artifact Files

Content Type	File Name	File Content
<b>Control</b>	delegable.proto	
<b>Uml</b>	delegable.md	
<b>Other</b>	.DS_Store	

## Code Map

Map Type	Name	Platform	Location
----------	------	----------	----------

SourceCode	Code 1	Daml	
------------	--------	------	--

## Implementation Map

Map Type	Name	Platform	Location
Implementation	Implementation 1	ChaincodeGo	

## Resource Map

Map Type	Name	Location	Description
Resource	Regulation Reference 1		

## Specification Behavior

### Delegable

## Taxonomy Formula: g

*A token class that implements this behavior will support the delegation of certain behaviors to another party or account to invoke them on the behalf of the owner. When applied to a token, behaviors that are Delegable will enable delegated request invocations. This is useful to provide another party to automatically be able to perform the behaviors that can be delegated without seeking permission up to a certain allowance.*

## Example

## Analogies

Name	Description
Broker	You may allow a broker to transfer your tokens as a part of an investment strategy. Setting an allowance can cap the total number of tokens the broker is allowed to perform delegated behaviors, when exceeded a new allowance

	request will need to be granted.
--	----------------------------------

## Comments

Applied to behaviors that are Delegable.

<b>Is External:</b>	True
<b>Constructor:</b>	

## Delegable responds to these Invocations

### *Allowance*

Id: 2e0fd8e5-2090-4c62-b094-232c32a78022

Description: A Request by a party or account to the owner of a token(s) to have the right to perform a delegated behavior on their behalf.

### **Request Message:**

AllowanceRequest

Description: The request

### Request Parameters

Name	Value
<b>Quantity</b>	Number of Tokens to be allowed.

### **Response Message**

AllowanceResponse

Description: The response

### Response Parameters

Name	Value
<b>Confirmation</b>	A confirmation receipt or denial be returned to the allowance requestor.



## Approve Allowance

Id: 6d5df99d-2f5e-4c7a-aea4-d2d54176abfd

Description: Same control message as the AllowanceRequest. This could allow for an AllowanceRequest to be forwarded to multiple parties needed to Approve and shield this from the requestor. When all Approvals are obtained, an AllowanceResponse could be sent.

### Request Message:

AllowanceRequest

Description: The request

#### Request Parameters

Name	Value
Quantity	Number of Tokens to be allowed.

### Response Message

ApproveResponse

Description: The response

#### Response Parameters

Name	Value
Confirmation	A confirmation response from the owner approving the an allowance request, indicating a allowance quantity the requestor has the option to invoke the Delegable behaviors on the token(s).

## Properties