1 Multi-Class Token System

Multi-class token system is the core mechanism in the DARC protocol, users can design different levels of tokens with different voting weights and dividend weights. With plugins as restrictions, the multi-class token system can be used to represent different components or assets in the organization.

1.1 Common Stocks

Common stocks are a central element in the mechanism of a joint stock company, and they serve as a means to raise capital and distribute ownership. Shareholders of common stock typically enjoy voting rights, allowing them to have a say in important company decisions during shareholder meetings. Additionally, common stockholders may receive dividends, which are a portion of the company's profits distributed to shareholders. This dual aspect of voting and dividends makes common stocks a key instrument for shareholders to participate in the governance and financial returns of the company.

If a token level is assigned a voting weight of 1 and a dividend weight of 1, and all essential matters in the DARC must be approved by all token holders at this level, then this token level can be considered equivalent to common stock in the DARC.

To instantiate such a token level, one must first establish it in the By-law Script, assigning a voting weight of 1 and a dividend weight of 1. Upon the successful execution of this script, the DARC protocol will initialize a level-0 token endowed with both a voting weight and dividend weight of 1.

```
batch_create_token_classes(
    ["TOKEN_0"],   // the token symbol in string
    [0],           // the level of token
    [1],           // the voting weight
    [1]            // the dividend weight
);
```

Consider about the complexity of corporation affairs, two voting rules are created for different purposes:

Voting rule 1: To approve an operation, the following conditions must be met: there must be more than 50% valid voting power in favor of the operation, and this voting needs to be in absolute majority mode, which means it must comprise at least 50% of the total token voting power. The time duration for voting rule 1 is 7 days(604800 seconds). After the program is approved, the operator needs to execute the program in 1 hour(3600 seconds).

Voting rule 2: To approve an operation, the following conditions must be met: there must be more than 75% valid voting power in favor of the operation, and this voting needs to be in relative majority mode, which means it must comprise at least 75% of the valid voting power. The time duration for voting rule 2 is 1 day(86400 seconds). After the program is approved, the operator needs to execute the program in 1 hour(3600 seconds).

The By-law Script below shows the initialization of voting rule 1 and 2. When adding a new plugin with voting rule index 1 or 2, the plugin will make sure that when the condition is triggered and the plugin is with the highest priority, a voting process will start with the selected voting rule.

```
batch_add_voting_rule([
    // add rule 1
    {
        // token level-0 is only allowed for the voting
        votingTokenClassList: [0],

        // approval threshold: 50%
        approvalThresholdPercentage: 50,

        // voting duration: 604800 seconds
        votingDurationInSeconds: 604800,

        // execution pending duration: 3600 seconds
        executionPendingDurationInSeconds: 3600,
        // is voting rule enabled or not: true
        isEnabled: true,

        // the note about the the voting rule
        note: "This is the voting rule index 1",

        // is absolute majority
        bIsAbsoluteMajority: true
    },

    // add rule 2
    {
        // token level-0 is only allowed for the voting
        votingTokenClassList: [0],

        // approval threshold: 75%
        approvalThresholdPercentage: 75,

        // voting duration: 604800 seconds
        votingDurationInSeconds: 86400,

        // execution pending duration: 3600 seconds
        executionPendingDurationInSeconds: 3600,

        // is voting rule enabled or not: true
        isEnabled: true,

        // the note about the the voting rule
        note: "This is the voting rule index 2",

        // is absolute majority
        bIsAbsoluteMajority: false
    }
]);
```

1.2 Class A/B Stocks

Class A and Class B stocks are fundamental tools in corporate finance. Class A provides voting rights and dividend advantages, typically held by founders. Class B, with limited voting rights, raises external capital while retaining control. This dual-class structure balances governance and growth needs, with implications for shareholders' decision-making and financial interests.

First initialize two token level: level-0 with voting weight 1 and dividend weight 1, and level-1 with voting weight 100 and dividend weight 1.

```
batch_create_token_classes(
    ["TOKEN_0", "TOKEN_1"],  // the token symbol in string
    [0, 1],           // the level of token
    [1, 100],           // the voting weight
    [1, 1]            // the dividend weight
);
```

Next we will mint 10000 level-0 tokens and 200 level-1 tokens. This will make the total voting power of level-0 tokens as 10000, and 20000 for level-1 tokens. In this way, the total voting power of the DARC is 30000, in which 66.7% in level-1 tokens. In this way, the co-founders and early stage investors will control the majority of the DARC for voting process which allows all level-0 and level-1 token holders.

We assume that addr1 and addr2 hold 10000 level-0 tokens in total, 5000 for each, and each of addr3, addr4, addr5 and addr5 holds 50 tokens. First mint tokens to these addresses in By-law Script.

```
batch_mint_tokens(
  [0, 0, 1, 1, 1, 1 ],
  [5000, 5000, 50, 50, 50, 50],
  [addr1, addr2, addr3, addr4, addr5, addr6]
);
```

Next add a plugin to limit the total supply of level-0 and level-1 token as 10000 and 20000.

Voting Rule 1: To approve the operation, 90% of total voting power of both level-0 and level-1 token must vote YES. The voting process needs to be in the absolute majority mode. The time duration for voting rule 1 is 7 days(604800 seconds). After the program is approved, the operator needs to execute the program in 1 hour(3600 seconds).

Before-Operation Plugin Rule 1: If the operation is BATCH_MINT_TOKENS and the level of token mint is 0 or 1, the operation needs to be executed in the sandbox before making the decision.

After-Operation Plugin Rule 1: If the operation is BATCH_MINT_TOKENS and the level of token mint is 0 or 1, the operation needs to be approved by voting rule 3.

Before-Operation Plugin Rule 2: If the operation is BATCH_DISABLE_PLUGIN and the index is before-operation plugin 1, before-operation plugin 2 or after-operation plugin 1, reject the operation. This plugin is with the highest priority.

In before-operation plugin 1 and after-operation plugin 1, when changing the shareholder structure by minting new level-0 or level-1 tokens, the operation needs to be approved by 90% of the total voting power. This will prevent the level-1 shareholders from diluting the value of level-0 tokens and protect those level-0 token holders, mostly retail investors.

The before-operation plugin 2 is a protection for the other plugins above. When any operator is trying to disable any of the three plugins, the operation will be rejected. This will permanently locked the mechanism and guarantee that the plugins cannot be disable.

```
batch_add_voting_rules([
    // add before-operation plugin 1
    {
        // token level-0 and level-1 are allowed for the voting
        votingTokenClassList: [0, 1],

        // approval threshold: 90%
        approvalThresholdPercentage: 90,

        // voting duration: 604800 seconds
        votingDurationInSeconds: 604800,

        // execution pending duration: 3600 seconds
        executionPendingDurationInSeconds: 3600,

        // is voting rule enabled or not: true
        isEnabled: true,

        // the note about the the voting rule
        note: "This is the voting rule 1(index 0)",

        // is absolute majority
        bIsAbsoluteMajority: true
    },
]);
```

Then add three plugins to the DARC.

```
batch_add_and_enable_plugins([
    // adding before-operation plugin with index 0
    {
        returnType: SANDBOX_NEEDED, // sandbox is needed
        level: 255, // level 255
        condition:
            operation_equals(BATCH_MINT_TOKENS) &
            (
                mint_token_class_equals(0) | mint_token_class_equals(1)
            ),
        votingRuleIndex: 0,  // no voting rule index needed
        note: "before-op plugin 1",
        bIsBeforeOperation: true  // the plugin will be added as before-op
    },

    // adding after-operation plugin with index 0
    {
        returnType: VOTING_NEEDED, // vote is needed
        level: 258, // level 258
        condition:
            operation_equals(BATCH_MINT_TOKENS) &
            (
                mint_token_class_equals(0) | mint_token_class_equals(1)
            )
        votingRuleIndex: 0,  // voting rule 1, index = 0
        note: "after-op plugin 1",
        bIsBeforeOperation: false  // the plugin will be added as before-op
    },

    // adding before-operation plugin 1
    {
        returnType: NO, // reject
        level: 257, // level 257
        condition:
            operation_equals(BATCH_DISABLE_PLUGIN) &
            (
                disable_before_op_plugin_index_equals(0) |
                disable_before_op_plugin_index_equals(1) |
                disable_after_op_plugin_index_equals(0)
            )
        votingRuleIndex: 0,  // no voting rule index needed
        note: "before-op plugin 2",
        bIsBeforeOperation: true  // the plugin will be added as before-op
    }
]);
```

1.3 Non-Voting Stocks

Non-voting stock, also known as preferred stock, is an ownership stake in a corporation that lacks voting rights but offers financial benefits like dividend priority, liquidation preference, and stability. It can be convertible or redeemable and is favored for its steady income. However, it provides no say in company decisions, making it a choice for income-focused investors seeking capital preservation with limited control over corporate matters. In the DARC protocol, when a token is initialized with a dividend weight of 1 and a voting weight of 0, it can be classified as non-voting shares.

1.4 Stocks with Stamp Duty

Stocks with stamp duty are securities subject to a government tax, known as a “stamp duty”, when they are bought or sold. This tax is applied to various financial instruments, including stocks and bonds, and is typically paid by the buyer or seller, depending on local regulations. People pay stamp duty when trading stocks because it serves as a source of government revenue and can also act as a tool to regulate financial markets and discourage excessive trading. The specific reasons and rates for stamp duty can vary by jurisdiction.

In the DARC protocol, we have the capability to create a mechanism that compels token holders to pay a transaction fee referred to as “stamp duty” when transferring tokens of varying levels. For instance, we can designate level-0 tokens as the standard stocks within the DARC protocol, with a fixed stamp duty of 1000 wei applied to each token transfer.

First, we must implement a plugin to disable all BATCH_TRANSFER_TOKENS operations when the token being transferred belongs to level-0. Next, we should develop a plugin that restricts the transfer of level-0 tokens exclusively through the BATCH_PAY_TO_MINT_TOKENS operation, provided each token’s transaction fee is greater than or equal to 1000 wei. Additionally, users are required to set the dividendable flag of the BATCH_PAY_TO_MINT_TOKENS operation to 0 (false), ensuring that the transaction fee (stamp duty) will not be considered as dividendable revenue for the DARC.

```
batch_add_and_enable_plugins([
    // adding before-operation plugin 1: disable trasnfer tokens for level-0 tokens
    {
        returnType: NO, // reject
        level: 255, // level 255
        condition:
            operation_equals(BATCH_TRANSFER_TOKENS) &
              transfer_tokens_level_equals(0)
        votingRuleIndex: 0,  // no voting rule index needed
        note: "disable trasnfer tokens for level-0 tokens",
        bIsBeforeOperation: true  // the plugin will be added as before-op
    },

    // adding before-operation plugin 2:
    // allow pay to transfer tokens for level-0 tokens,
    // with the transaction fee >= 1000 wei per token
    {
        returnType: YES_AND_SKIP_SANDBOX, // allow and skip sandbox
        level: 256, // level 256
        condition:
            operation_equals(BATCH_PAY_TO_TRANSFER_TOKENS) &
            (
                pay_to_transfer_tokens_level_equals(0)  &
                transaction_fee_per_token_GE(1000)   &
                pay_to_trasnfer_tokens_dividendable_flag_equals(0)
            )
        votingRuleIndex: 0,  // no voting rule index needed
        note: "allow pay to transfer level-0 token with transaction fee
              > 1000 and dividendable flag 0",
        bIsBeforeOperation: true  // the plugin will be added as before-op
    }
]);
```

1.5 Board of Directors

In the realm of DAOs, decisions are typically reached through voting using governance tokens. However, this process can become sluggish and inefficient when involving thousands of token holders in each decision. Furthermore, the absence of regulations or predefined rules for the pending execution of proposal smart contracts means that approved votes can potentially authorize a wide array of actions within the DAO.

Contrastingly, in traditional joint-stock companies, day-to-day operations and affairs are overseen by a Board of Directors, rather than relying on decision-making by all shareholders. Boards of Directors are known for their efficiency, driven by their expertise, strategic acumen, rapid decision-making capabilities, and their ability to maintain stability and consistency in corporate governance. While shareholder voting remains vital for accountability and representation, the Board system offers a structured and informed approach to decision-making, ensuring that the company's long-term interests are thoughtfully prioritized and effectively managed.

Below is an example of how the DARC Y experiment designs its board of directors using a multi-level token system. There are four types of tokens, each with its unique characteristics and roles within the organization, as illustrated in Table 1.

Level 0 (Class A Token): These tokens, with a total supply of 400, possess both voting and dividend weights of 1. They likely represent common stakeholders within the DARC Y protocol.

Level 1 (Class B Token): With a total supply of 60, Class B Tokens hold a substantial voting weight of 10, making them influential in decision-making. They also carry a dividend weight of 1, implying a potential share in revenue distribution.

Level 2 (Independent Director): There is only one Independent Director token, designed with no voting or dividend weight, suggesting a unique role in the governance structure.

Level 3 (Board of Directors): Comprising five tokens, the Board of Directors holds a voting weight of 1 but lacks dividend weight, indicating its primary function in decision-making.

Level	Name	Total Supply	Voting Weight	Dividend Weight
0	Class A Token	400	1	1
1	Class B Token	60	10	1
2	Independent Director	1	0	0
3	Board of Directors	5	1	0

Table 1: A structure of token distribution in DARC Y

Here are some guiding principles for designing the Board of Directors mechanism:

Principle 1: The Board of Directors should consist of a maximum of 5 members, and all board decisions must receive approval from more than 2/3 of its members.

Principle 2: If an address holds over 2/3 of the Class A tokens, it can represent the interests of retail investors and may be selected as a Class A board member. In the absence of any address holding over 2/3 of these tokens, there will be no Class A representative on the board.

Principle 3: The Board of Directors should always include one independent director, who has the option to appoint their successor. The removal of the independent director from the board is not permitted.

Principle 4: The board should comprise a maximum of 3 Class B board members, with the option to add any address holding more than 5% of the Class B tokens. Existing board members can nominate a Class B board member, subject to approval by at least 2/3 of all board members.

Principle 5: Existing board members have the authority to propose the removal of a Class B board member. Such a proposal can be approved under the following conditions: (1) The address to be removed holds a quantity less than or equal to 5% of the total supply of Class B tokens, or (2) The address does not possess more than 2/3 of the total supply of Class A tokens or the independent director token, and the proposal garners more than 2/3 of the voting power within the Board of Directors.

Principle 6: Class A Token directors, Class B Token directors, and the independent director should maintain independence from token holdings. This means that (1) Class A Token directors may only possess Class A tokens and Board of Directors tokens, and they are prohibited from holding Class B tokens or independent director tokens; (2) Class B Token directors may exclusively hold Class B tokens and Board of Directors tokens, refraining from ownership of Class A tokens or independent director tokens; and (3) The independent director is solely allowed to hold the independent director token.

Next, we will design the following plugins to implement the rules and processes defined above:

Before-Operation Plugin Rule 1: If an operator address holds more than 2/3 of the level-0 tokens and does not hold any level-1, level-2, or level-3 tokens, it can mint one level-3 token to itself without requiring a sandbox check. This plugin rule is designed for the nomination of Class A token board members.

Before-Operation Plugin Rule 2: When an operator address intends to transfer a level-2 token to another target address, and the target address does not hold any level-0, level-1, or level-3 tokens, this operation can be approved without a sandbox check. This plugin facilitates the transfer of the independent director position.

Before-Operation Plugin Rule 3: In cases where an operator address holds 1 level-2 token and 0 level-3 tokens and wishes to mint a level-3 token for itself, this operation can be approved without any sandbox check. This plugin serves the purpose of nominating an independent director.

Before-Operation Plugin Rule 4: Operations involving the minting or burning of level-2 tokens need to be rejected. This plugin restricts the number of independent directors.

Before-Operation Plugin Rule 5: If an operator address intends to mint 1 level-3 token and the target address holds more than 5% of the total supply of level-1 tokens, and the operator address holds 1 level-3 token, this operation must undergo a sandbox check. This plugin is utilized for the nomination of Class B token board members.

Before-Operation Plugin Rule 6: If an operator address intends to burn a level-3 token from a target address, and the target address holds less than or equal to 2/3 of the total supply of level-0 tokens, holds less than or equal to 5% of level-1 tokens, and has 0 level-2 tokens, this operation can be approved without a sandbox check. This plugin facilitates the removal of unqualified board members.

Before-Operation Plugin Rule 7: If an operator address intends to burn a level-3 token from a target address, and the target address holds less than or equal to 2/3 of the total supply of level-0 tokens, holds more than 5% of level-1 tokens, and has 0 level-2 tokens, and the operator address possesses at least 1 level-3 token, this operation must undergo a sandbox check. This plugin is employed for the removal of Class B token board members.

After-Operation Plugin Rule 1: If an operator address intends to burn a level-3 token from a target address, and the target address holds less than or equal to 2/3 of the total supply of level-0 tokens, holds more than 5% of level-1 tokens, has 0 level-2 tokens, and the operator address possesses at least 1 level-3 token, this operation must undergo a vote by voting rule 1. This vote involves all board members and requires an absolute majority mode, with an approval rate exceeding 66%. This plugin is utilized for Class B board member elections.

And next are the plugins implemented in the By-law Script.

```
const plugin_before_op_1 = {
    returnType: YES_AND_SKIP_SANDBOX,
    level: 255,
    condition: operation_equals(BATCH_MINT_TOKENS)
               & number_of_token_mint_equals(1)
               & level_of_token_mint_equals(3)
               & operator_owns_num_of_token_GE(0,267)
               & operator_owns_num_of_token_equals(1, 0)
               & operator_owns_num_of_token(2, 0)
               & operator_mint_to_itself() ,
    votingRuleIndex: 0,
    note: "before op 1",
    bIsBeforeOperation: true
},

const plugin_before_op_2 = {
    returnType: YES_AND_SKIP_SANDBOX,
    level: 255,
    condition: operation_equals(BATCH_TRANSFER_TOKENS)
               & transfer_token_level_equals(2) ,
    votingRuleIndex: 0,
    note: "before op 2",
    bIsBeforeOperation: true
```

```
},

const plugin_before_op_3 = {
    returnType: YES_AND_SKIP_SANDBOX,
    level: 255,
    condition: operation_equals(BATCH_MINT_TOKENS)
               & operator_owns_num_of_token_equals(2, 1)
               & number_of_token_mint_equals(1)
               & level_of_token_mint_equals(3) ,
    votingRuleIndex: 0,
    note: "before op 3",
    bIsBeforeOperation: true
},

const plugin_before_op_4 = {
    returnType: NO,
    level: 257,
    condition: (operation_equals(BATCH_BURN_TOKENS)
               & level_of_token_burned_equals(2)) |
               (operation_equals(BATCH_MINT_TOKENS)
               & level_of_token_mint_equals(2)) ,

    votingRuleIndex: 0,
    note: "before op 4",
    bIsBeforeOperation: true
},

const plugin_before_op_5 = {
    returnType: SANDBOX_NEEDED,
    level: 256,
    condition: operation_equals(BATCH_MINT_TOKENS)
               & level_of_token_mint_equals(3)
               & number_of_token_mint_equals(1)
               & operator_owns_num_of_tokens_equals(3,1)
               & batch_mint_tokens_target_address_ownes_num_of_tokens_greater(1,12),
    votingRuleIndex: 0,
    note: "before op 5",
    bIsBeforeOperation: true
},

const plugin_before_op_6 = {
    returnType: YES_AND_SKIP_SANDBOX,
    level: 255,
    condition: operation_equals(BATCH_BURN_TOKENS)
               & level_of_token_burned_equals(3)
               & batch_burn_token_target_address_owns_num_of_tokens_LE(1, 12)
               & batch_burn_token_target_address_owns_num_of_tokens_LE(0, 267)
               & batch_burn_token_target_address_owns_num_of_tokens_equals(2,0),
    votingRuleIndex: 0,
    note: "before op 6",
    bIsBeforeOperation: true
},

const plugin_before_op_7 = {
    returnType: SANDBOX_NEEDED,
    level: 256,
    condition: operation_equals(BATCH_BURN_TOKENS)
               & level_of_token_burned_equals(3)
               & batch_burn_token_target_address_owns_num_of_tokens_GE(1, 12)
               & batch_burn_token_target_address_owns_num_of_tokens_LE(0, 256)
               & batch_burn_token_target_address_owns_num_of_tokens_equals(2,0),
    votingRuleIndex: 0,
    note: "before op 7",
    bIsBeforeOperation: true
},

const plugin_after_op_1 = {
    returnType: VOTING_NEEDED,
    level: 253,
    condition: operation_equals(BATCH_BURN_TOKENS)
               & level_of_token_burned_equals(3)
               & batch_burn_token_target_address_owns_num_of_tokens_GE(1,12)
               & batch_burn_token_target_address_owns_num_of_tokens_LE(0, 267)
               & batch_burn_token_target_address_owns_num_of_tokens_equals(2,0)
               & operator_ownes_num_of_tokens_equals(3,1),
    votingRuleIndex: 1,
    note: "after op 1",
    bIsBeforeOperation: false
},

batch_add_and_enable_plugins([
    plugin_before_op_1,
    plugin_before_op_2,
    plugin_before_op_3,
    plugin_before_op_4,
    plugin_before_op_5,
    plugin_before_op_6,
    plugin_before_op_7,
    plugin_after_op_1
]);
```

1.6 Corporate Bonds

Company bonds are debt securities issued by corporations to raise funds. Investors lend money to the company in exchange for periodic interest payments and the return of their principal amount at maturity. They provide companies with a means to secure capital for various purposes and offer investors a predictable income stream.

In the context of the DARC protocol, the commands `BATCH_PAY_TO_MINT_TOKENS` and `BATCH_BURN_TOKENS_AND_REFUND` facilitate the creation of bond tokens available for purchase by any address during a specified time frame, with a designated price for both acquisition and redemption. Two rules govern the buying and refunding of these bond tokens:

Rule 1: Between January 1, 2020, and February 1, 2020, any address can purchase no more than 100 bond tokens(level-2) using `BATCH_PAY_TO_MINT_TOKENS` at a rate of 10,000 wei per token. The total supply of bond tokens should not exceed 10000.

Rule 2: Addresses have the option to redeem bond tokens at a rate of 15000 wei per token using `BATCH_BURN_TOKENS_AND_REFUND` after January 1, 2030.

```
batch_add_and_enable_plugins([
    {
        returnType: YES_AND_SKIP_SANDBOX,
        level: 253,
        condition: operation_equals(BATCH_PAY_TO_MINT_TOKENS)
                   & pay_to_mint_tokens_price_per_token_equals(10000)
                   & timestamp_greater(1577858400) // 2020-01-01-0-0-0
                   & timestamp_less(1580536800) // 2020-02-01-0-0-0
                   & pay_to_mint_tokens_level_equals(2) // level of token
                   & number_of_token_pay_to_mint_LE(100) // number to mint
                   & total_number_to_tokens_LE(2,99900), // total token supply
        votingRuleIndex: 0,
        note: "",
        bIsBeforeOperation: true
    },
    {
        returnType: YES_AND_SKIP_SANDBOX,
        level: 253,
        condition: operation_equals(BATCH_BURN_TOKENS_AND_REFUND)
                   & burn_tokens_and_refund_price_per_token_equals(15000)
                   & timestamp_GE(1893477600) // 2030-01-01-0-0-0
                   & burn_tokens_and_refund_level_equals(2), // level of token
        votingRuleIndex: 0,
        note: "",
        bIsBeforeOperation: true
    },
]);
```

1.7 Product Tokens and Non-Fungible Tokens

When we set both voting weight and dividend weight to 0, and forcefully require operators to mint tokens at a specific price, these tokens can be considered as product tokens representing paid products or services of the DARC instance.

When we further configure several levels of tokens to allow only one pay-to-mint one token per transaction, prohibit minting and burning operations, and disallow additional pay-to-mint transactions if one token of that level already exists. Simultaneously, we permit holders of these tokens to freely transfer them to other addresses, at this point, these level-specific tokens can be considered as NFTs (Non-Fungible Tokens) [?]. NFTs can be used and considered as unique digital assets that represent ownership or proof of authenticity for specific items or content in the DARC protocol.