

# 1 Memberships

In every company, in addition to members such as the board of directors, shareholders, executives, there are also numerous employees, contractors, interns, customers, suppliers, and so on. They may not hold any first-level tokens or company shares, but their payroll, orders, subscription fees, and other functionalities require batch management. Therefore, membership is another set of convenient tools for use in the DARC protocol.

Table 1 is an example of a hierarchical membership structure of the DARC X, a consulting company. In this table, we define 6 levels: co-founder, substantial shareholder, executive, manager, employee and departed employee. And we can define some plugins for DARC X based on this membership table.

**Rule 1:** We need to limit the co-founders from selling a large amount of stock without notifications to the executives. If the operator is an address with a "co-founder" role and operation is selling more than 100000 level-0 tokens, this operation needs to be approved by all board members.

**Rule 2:** We provide 10 ETH per month as the salary for both employee-level and manager-level address (level 4 & 5).

| Level | Role                    |
|-------|-------------------------|
| 1     | Co-founder              |
| 2     | Substantial Shareholder |
| 3     | Executive               |
| 4     | Manager                 |
| 5     | Employee                |
| 6     | Departed Employee       |

Table 1: A hierarchical membership structure

| Address   | Role                        | Name           | IsSuspended |
|-----------|-----------------------------|----------------|-------------|
| 0x0AC..03 | 1 (Co-founder)              | Ann            | No          |
| 0x156..21 | 2 (Substantial Shareholder) | Banana Capital | No          |
| 0x918..1B | 3 (Executive)               | Tom            | No          |
| 0x4E1..90 | 4 (Manager)                 | Jack           | No          |
| 0x510..0B | 5 (Employee)                | Bob            | No          |
| 0x113..C7 | 6 (Departed Employee)       | Tim            | No          |

Table 2: A membership table of DARC X

Now we analyze the rules above and design plugins in By-law Script.

For **Rule 1**, we need to make sure that if the current operation is "transfer token", the level of token is 0, and the number of token is more than 100000, and the address is in the membership table with a role level equals to 1, the program will be suspended and waiting for a voting process. The voting process is defined and saved as **voting rule 1**, which requires all the holders of level-3 token to vote in 1 hour. There are 5 level-3 tokens in total, and each board member holds 1 token. If all the 5 board members vote yes in 1 hour, this operation will be approved. Otherwise, the whole program will be rejected and the operation will fail.

Here is an example plugin for **Rule 1** designed in By-law Script:

```
const plugin_Rule_1 =
{
  condition: (operation_name_euqlas(BATCH_TRANSFER_TOKENS))
    & ( transfer_token_level_equals(0) )
    & ( transfer_token_amount_greater(100000))
    & ( operator_membership_level_equals(1)) ,
  return_type: VOTING_NEEDED,
  is_before_operation: false,
  return_level: 100,
```

```

    voting_rule_index: 1
}

```

For **Rule 2**, we need to make sure that if the operator is assigned with a role level equals 4 or 5, the operation is adding withdrawable balance, the total amount of operation is less than or equal to 10 ETH(or 10000000000000000000 wei), and the operator did the same operation in more than 30 days(or 2592000 seconds), this operation will be approved and sandbox check can be skipped for this operation.

Here is an example plugin for **Rule 2** designed in By-law Script:

```

const plugin_Rule_2 =
{
    condition: operation_equals(BATCH_ADD_WITHDRAWABLE_BALANCE)
    & total_add_withdrawable_balance_LE(10000000000000000000)
    & last_operation_by_operation_period_for_operator_GE(
        BATCH_ADD_WITHDRAWABLE_BALANCE, 2592000
    )
    & ( (operator_membership_level_equals(4))
        | (operator_membership_level_equals(5))
    ) ,
    returnType: YES_AND_SKIP_SANDBOX,
    bIsBeforeOperation: true,
    level: 100,
    votingRuleIndex: 0,
    note: ""
}

```