

## 1 成员资格

在每个公司中，除了董事会成员、股东、高管等成员外，还有许多员工、合同工、实习生、客户、供应商等。他们可能不持有任何一级代币或公司股份，但他们的工资、订单、订阅费和其他功能需要批量管理。因此，成员资格是另一套用于 DARC 协议中的便捷工具。

表 ?? 是 DARC X 咨询公司的一个分层成员资格结构示例。在这个表中，我们定义了 6 个级别：联合创始人、重要股东、高管、经理、员工和离职员工。我们可以基于这个成员资格表为 DARC X 定义一些插件。

规则 1：我们需要限制联合创始人在未通知高管的情况下出售大量股票。如果操作者是一个拥有“联合创始人”角色的地址，并且操作是出售超过 100000 级别-0 代币，这个操作需要得到所有董事会成员的批准。

规则 2：我们为员工级别和经理级别的地址（级别 4 和 5）提供每月 10 ETH 的工资。

级别	角色
1	联合创始人
2	重要股东
3	高管
4	经理
5	员工
6	离职员工

表 1: 一个分层成员资格结构

地址	角色	名称	是否暂停
0x0AC..03	1 (联合创始人)	Ann	否
0x156..21	2 (重要股东)	Banana Capital	否
0x918..1B	3 (高管)	Tom	否
0x4E1..90	4 (经理)	Jack	否
0x510..0B	5 (员工)	Bob	否
0x113..C7	6 (离职员工)	Tim	否

表 2: DARC X 的成员资格表

现在我们分析上述规则并设计 By-law 脚本中的插件。

对于规则 1，我们需要确保如果当前操作是“转移代币”，代币的级别是 0，并且代币数量超过 100000，并且地址在成员资格表中角色级别等于 1，程序将被暂停并等待投票过程。投票过程被定义并保存为投票规则 1，要求所有持有级别-3 代币的持有者在 1 小时内投票。总共有 5 个级别-3 代币，每个董事会成员持有 1 个代币。如果所有 5 个董事会成员在 1 小时内投票同意，此操作将被批准。否则，整个程序将被拒绝，操作将失败。

这里是为规则 1 设计的 By-law 脚本中的示例插件：

```
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
}
```

对于规则 2，我们需要确保如果操作者被分配的角色级别等于 4 或 5，操作是增加可提现余额，操作的总金额小于或等于 10 ETH（或 10000000000000000000 wei），并且操作者在超过 30 天（或 2592000 秒）内做了同样的操作，这个操作将被批准，并且可以跳过此操作的沙盒检查。

这里是为规则 2 设计的 By-law 脚本中的示例插件：

```
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: ""
}
```