USR Scheme

❖ **修改历史**

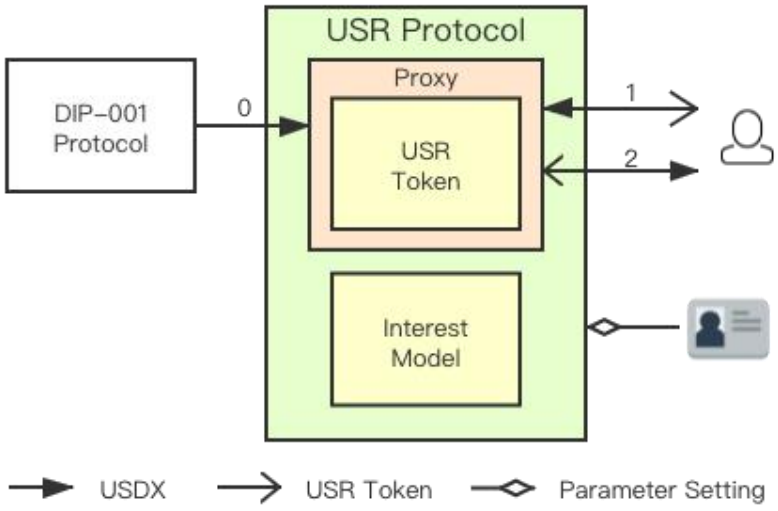
Date	Contents	Author
2020.2.25	v0.1	Horsen
2020.2.26	v0.2	Horsen
2020.2.27	v0.3	Skyge
2020.2.28	v0.4	Skyge
2020.3.4	v0.5	Skyge
2020.3.6	v0.6	Horsen

❖ 功能描述

USDx 是 dForce Network 开发的由多种成份币合成的稳定币，利用 DIP-001 协议可以将成份币存入去中心化借贷市场获得利息。这些利息会按设置的比例注入 USR 合约，转化为用户的 USDx 存款利息。

用户只需将 USDx 存入 USR（USDx Saving Rate）合约，得到存款凭证“USR Token”。USR Token 与 USDx 的兑换比率会根据年化利息不断增长，并可随时从 USR 合约中兑换出相应的 USDx 本金和利息。

❖ 整体架构



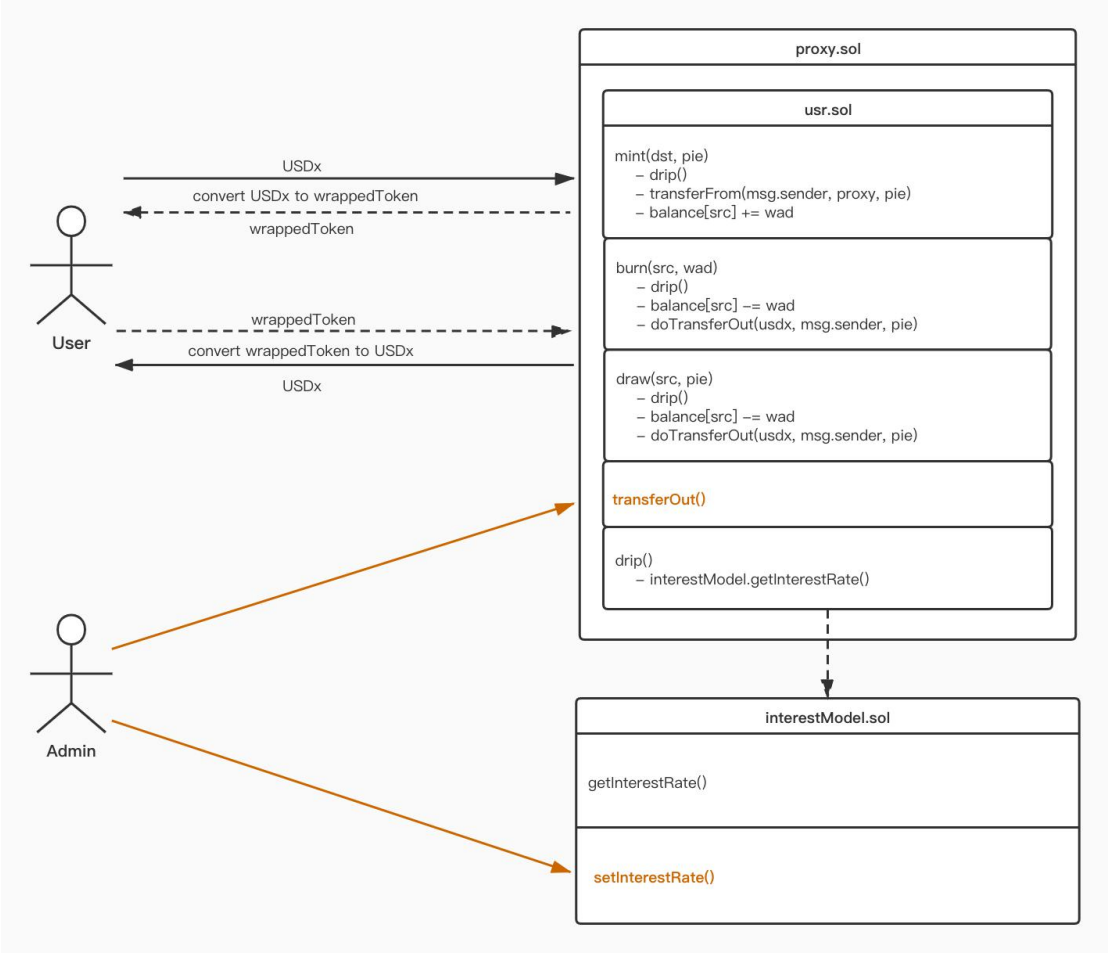
图中，USR Token 和 Interest Model 可升级，箭头表示 token 流转或兑换。

用户向 USR 存入 USDx 时，根据当时 USDx/USR Token 的兑换率 rate（rate 是根据 interest model 中的参数计算得到的，表示一个 USR Token 可以兑换多少 USDx。例如，存入 10000 USDx，当时 rate = 1.001，那么将获得 9990.01 USR Token）得到相应数量的 USR Token。同时，存入的 USDx 将转入 USR Core Proxy。

DIP-001 是 USDx 成份币的生息协议，其获得的利息会转入 USR Core Proxy。当用户执行存款或取款操作时，USR Protocol 会将收到成份币全额转入 USDx Protocol，铸成的 USDx 会自动转入 USR。

用户从 USDR 取回 USDx 时，根据当时的 USDx/USR Token 的兑换率 rate 计算出返还 USDx 的数量，从 USR Core Proxy 中取出相应数量的 USDx 给用户，同时销毁用户的 USR Token。

❖ 合约及接口函数



1.USR.sol

注：一、合约部署者享有初始 owner 权限

二、USR 合约是一个 ERC20 合约，所以它具备基本的 ERC20 合约的方法，包括：approve(), balanceOf(), transfer(), transferFrom()，此处不再对这些函数做说明。

1)、mint(_dst, _pie) external:

在合约未紧急关闭的情况下，方法调用者替用户（_dst）存入_pie 数量的 USDx 获得相应的 USR。

2)、burn(_src, _wad) external:

在合约未紧急关闭的情况下，方法调用者通过销毁用户（_src）_wad 数量的 USR 获得相应的 USDx。

3)、draw(_src, _pie) external note whenNotPaused:

在合约未紧急关闭的情况下，方法调用者想要获得用户（_src）_pie 数量的 USDx。

4)、getExchangeRate() public view:

计算获得此时此刻 USR 兑换 USDx 的比率。

5)、getTotalBalance(_account) external view:

计算用户（_account）此时此刻可以取出的最多的 USDx 数量（本息和-交易手续费）。

6)、share() external view:

表示此合约还可存入 USDx 生息的数量。

7)、equity() external view:

表示合约当前盈余情况，大于 0 表示盈余，小于 0 表示亏损，提示 manager 转入更多的 USDx 以支付用户存入 USDx 会获得的利息。

8)、drip() public:

更新最新的 USR 兑换 USDx 的比率。

9)、takeOut(address tokenAddress, address receiver, uint256 amount) external onlyManager whenNotPaused:

在合约未紧急关闭的情况下，具有 manager 权限的 account 将某个 token 取出 amount 个数量转给 receiver

10)、updateInterestModel(address _interestModel) external note onlyOwner:

具有 owner 权限的 account 更换读取的 interest model 合约的地址。

11)、setMaxDebtAmount(_newMaxDebtAmount) external onlyOwner:

具有 owner 权限的 account 更新一共可存入此合约的 USDx 的数量

12)、updateOriginationFee(uint _newOriginationFee) external onlyOwner

具有 owner 权限的 account 重新设置取款手续的比率。

13)、initialize(_name, _symbol, _decimals, _interestModel, _usdx, _originationFee, _maxDebtAmount):

相当于合约的 constructor()初始化函数，仅可调用一次，是为了配合 proxy 合约。

2. InterestModel.sol

注：合约部署者享有初始 owner 权限

1)、setInterestRate(uint _interestRate) external onlyManager;

具有 manager 权限的 account 更新 interest rate 的基准值。

2)、getInterestRate() external view returns (uint);

读取设定的 interest rate 的值

3.Ownable.sol

注：一、这是一个库合约，可供多个合约继承使用。

二、合约部署者享有初始 owner 权限

1)、 modifier onlyOwner()

修饰的函数方法只能由 owner 使用

2)、 modifier onlyManager()
凡是具有 manager 权限的账号都可以使用的方法

3)、 transferOwnership(address pendingOwner_) external onlyOwner:
原始的 owner 账户将权限转移到等待确认的 owner 账户(pendingOwner_)。

4)、 acceptOwnership() external;
等待确认的 owner 账户(pendingOwner)接收 onwer 权限，成为新的 owner。

5)、 setManager(address account) external onlyOwner
具有 owner 权限的 account 设置新的账户(account)具有 manager 权限。

6)、 removeManager(address account) external onlyOwner
具有 owner 权限的 account 将原来的 manager 账号(account)移除。

5.Pausable.sol
注：一、这是一个库合约，它本身继承了 Ownable.sol 合约，可供多个合约继承使用。
二、合约部署者享有初始 owner 权限
1)、 pause() public whenNotPaused onlyOwner
在合约未紧急关闭的情况下，具有 owner 权限的 account 将合约紧急关闭。

2)、 unpause() public whenPaused onlyOwner
在合约紧急关闭的情况下，具有 owner 权限的 account 将合约重新开启。

❖ 设计参考

1. dForce USDx
2. dForce DIP-001
3. Maker DSR
4. Chai
5. imBTC