solv

Rebase X 3D链社区 X BUIDLER DAO

ERC-3525设计思想及应用开发

Ethan 蔡一 | caiyi.eth

Solv协议核心开发者，ERC-3525联合作者

2023.5.24

内容

ERC-3525 是什么及其由来

ERC-3525 设计思想及规范内容

ERC-3525 参考实现介绍

ERC-3525 应用开发及案例

ERC-3525基础描述

https://eips.ethereum.org/EIPS/eip-3525

Standards Track: ERC

ERC-3525: Semi-Fungible Token

Defines a specification where ERC-721 compatible tokens with the same SLOT and different IDs are fungible.

Authors Will Wang (@will42w), Mike Meng <myan@solv.finance>, Yi Cai (@YeeTsai) <yee.tsai@gmail.com>, Ryan Chow <ryanchow@solv.finance>, Zhongxin Wu (@Nerverwind), AlvisDu (@AlvisDu)

Created 2020-12-01

Requires EIP-20, EIP-165, EIP-721

- **Abstract**: 介绍了<ID, SLOT, VALUE>三元组每个元素的含义，并说明协议在此基础上设计了相关的接口/功能
- **Motivation**: 通过阐述ERC-20、ERC-721在数字资产token化方面的不足，引出ERC-3525的解决方案

ERC-3525的由来

- 由于创业产品开发而提出的
- 定期存单/债券/票据等资产无法用ERC-20、721、1155方便的描述
- 每种Token标准（20、721、1155）都有其最佳使用范围
- 我们需要一种新的Token标准来建模类似存单这样的半匀质化的资产
- 从draft到final历时超过一年，不停在实际应用中打磨调整

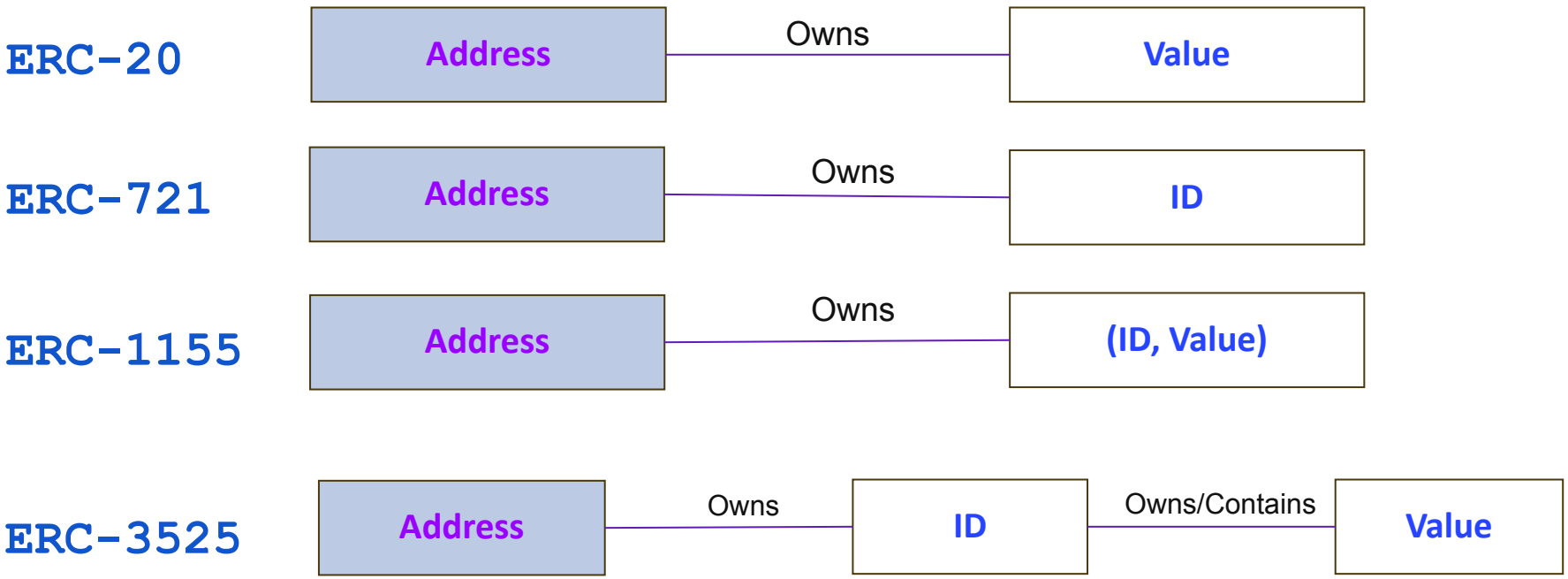
ERC-20,ERC-721,ERC-1155

- **ERC-20**
  - 对数量的描述很清晰，支持精度
  - 只能是同质化，如果要表达不同含义，必须部署多个合约
  - 适合描述货币类业务
- **ERC-721**
  - 有ID的概念，每个ID代表“一个”东西，各ID可对应完全不同的东西
  - 元数据仅有name, description, image，无数量等描述
  - 适合描述艺术品类业务
- **ERC-1155**
  - 引入ID的概念，但ID标识“一类”东西
  - 可多人共有一个ID
  - 适合描述游戏道具类无精度数量
  - 适合描述游戏道具

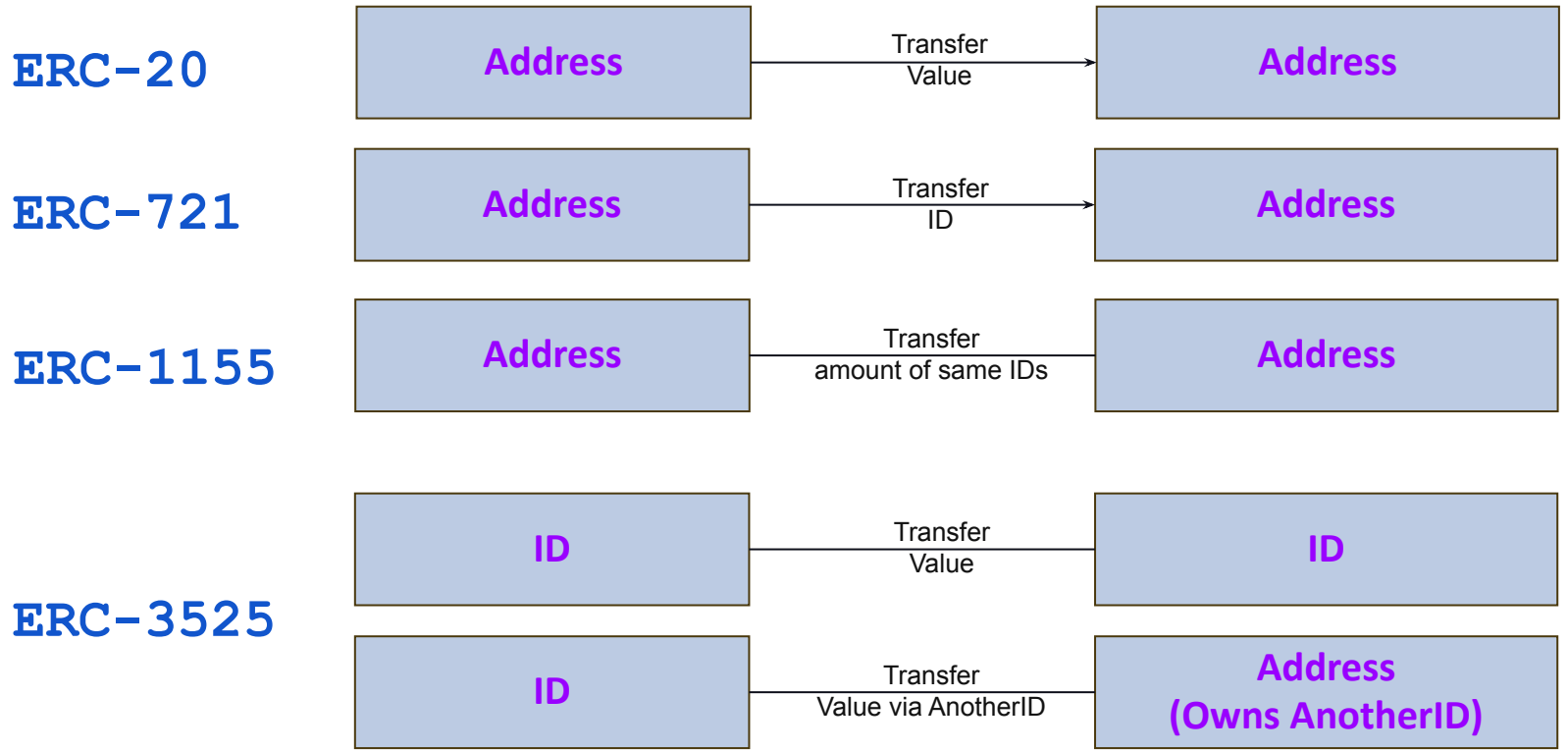
ERC-3525

- Address拥有的是ID，而不是amount，这是与EIP-1155最大的区别
- slot代表“一类”东西
- 每个ID是“一个”东西，兼容ERC-721
- 每个ID有自己的属性value
- 同一个slot下的不同ID可以split和merge，相当于分裂和合成
- 每个ID对应的slot可以变化，即从归属一个slot变成归属另一个

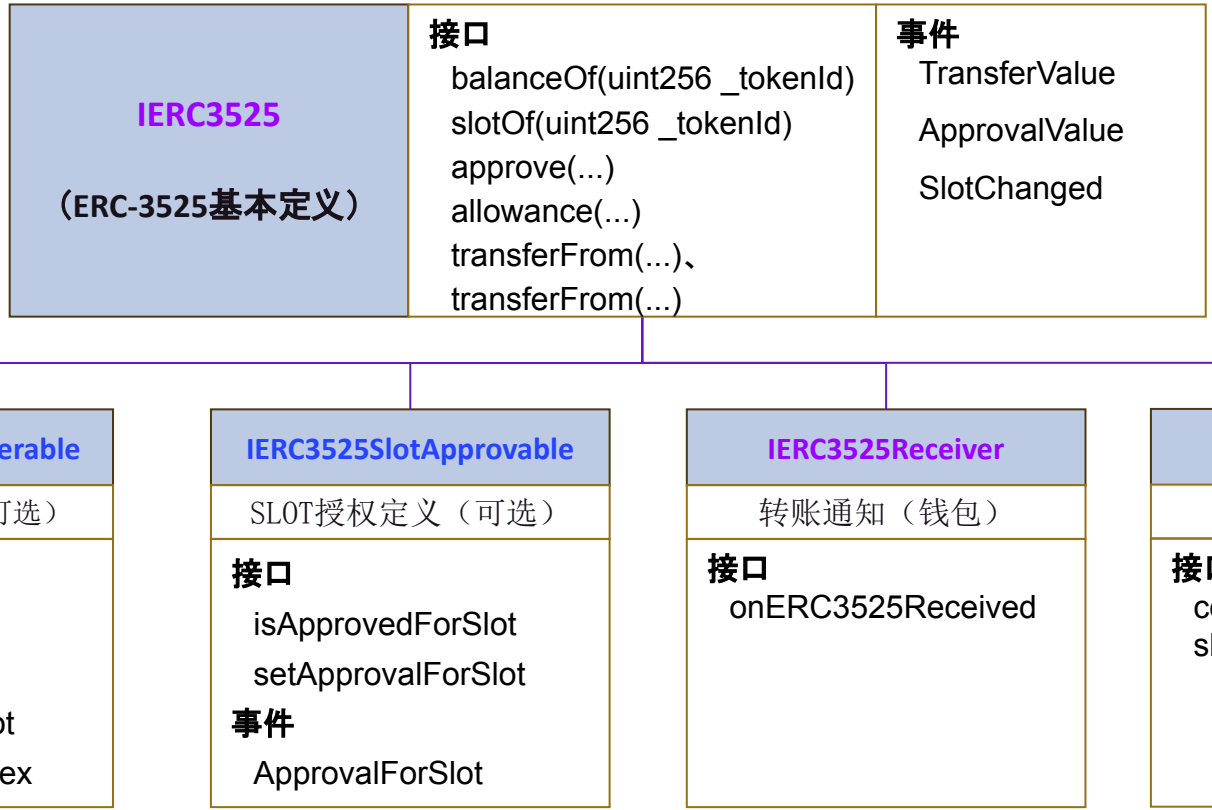
基础结构对比


ERC-20
Address
Owns
Value
ERC-721
Address
Owns
ID
ERC-1155
Address
Owns
(ID, Value)
ERC-3525
Address
Owns
ID
Owns/Contains
Value

转账模型对比


ERC-20
Address
Transfer
Value
Address
ERC-721
Address
Transfer
ID
Address
ERC-1155
Address
Transfer
amount of same IDs
Address
ERC-3525
ID
Transfer
Value
ID
ID
Transfer
Value via AnotherID
Address
(Owns AnotherID)

ERC-3525接口总体结构



基本接口定义(一)

interface IERC3525

function valueDecimals() **external view returns** (**uint8**);

function balanceOf(**uint256** _tokenId) **external view returns** (**uint256**);

function slotOf(**uint256** _tokenId) **external view returns** (**uint256**);

function approve(**uint256** _tokenId, **address** _operator, **uint256** _value) **external payable**;

function allowance(**uint256** _tokenId, **address** _operator) **external view returns** (**uint256**);

function transferFrom(**uint256** _fromTokenId, **uint256** _toTokenId, **uint256** _value) **external payable**;

function transferFrom(**uint256** _fromTokenId, **address** _to, **uint256** _value) **external payable returns** (**uint256**);

可选接口定义(一):枚举

interface IERC3525SlotEnumerable

```
function slotCount() external view returns (uint256);

function slotByIndex(uint256 _index) external view returns (uint256);

function tokenSupplyInSlot(uint256 _slot) external view returns (uint256);

function tokenInSlotByIndex(uint256 _slot, uint256 _index) external view returns (uint256);
```

可选接口定义(二):SLOT授权

interface IERC3525SlotApprovable

```
event ApprovalForSlot(address indexed _owner, uint256 indexed _slot, address indexed _operator, bool _approved);

function setApprovalForSlot(address _owner, uint256 _slot, address _operator, bool _approved) external payable;

function isApprovedForSlot(address _owner, uint256 _slot, address _operator) external view returns (bool);
```

支持SFT, Slot, ID, value四级授权模式

1. 用ERC-721的setApproveForAll来对整个SFT授权(用isApprovedForAll检查)
2. **可选**:用ERC-3525的setApprovalForSlot对某个Slot授权(用isApprovedForSlot检查)
3. 用ERC-721的approve(tokenId)来对某个Id授权(用getApproved检查)
4. 用ERC-3525的approve(tokenId, value)来对某个Id的value授权(用allowance来检查)

转账通知与反馈接口

interface IERC3525Receiver

```
function onERC3525Received(address _operator, uint256 _fromTokenId, uint256 _toTokenId,
uint256 _value, bytes calldata _data) external returns (bytes4);
```

ERC-3525协议规定：协议实现者需要去探测转账接收方有没有实现转账通知接口

1. 如果接收方（合约）实现该接口则调用该通知功能，由接收方在通知接口功能中进行响应
2. 如果接收方（合约）没有实现这个功能，则正常进行转账操作
3. 如果接收方不是合约，则正常进行转账操作（针对非合约地址转账，有另外的方案实现通知功能，与ERC-3525保持兼容，但未规定在ERC-3525协议中，未来可以扩展）

转账通知及反馈机制与现有协议的不同

- **ERC-20**
  - 没有规定SafeTransfer相关接口，有些扩展实现自行规定，与ERC-721情况类似
- **ERC-721**
  - 包含SafeTransfer与普通Transfer两套接口，带来了两个问题：
    - 1. 发起转账的钱包也不知到应该调用哪个接口最合适，只能选一个最简单、通用的，实际上没有达到协议设计的目标
    - 2. 一旦调用方采用SafeTransfer接口，接收方如果不实现转账通知接口，则调用必然失败，与旧合约兼容性差
- **ERC-1155**
  - 只保留SafeTransfer接口，消除了第1个问题，但是第2个问题更严重了，很多已有合约都会失败

元数据定义

function contractURI**() external view returns (string memory);**

```
{
  "title": "Contract Metadata",
  "type": "object",
  "properties": {
    "name": {
      "type": "string",
      "description": "Contract Name"
    },
    "description": {
      "type": "string",
      "description": "Describes the contract"
    },
    "image": {
      "type": "string",
      "description": "Optional. Either a base64 encoded imgae data or a URI pointing
    },
    "external_link": {
      "type": "string",
      "description": "Optional. A URI pointing to an external resource."
    },
    "valueDecimals": {
      "type": "integer",
      "description": "The number of decimal places that the balance should display -
    }
  }
}
```

function slotURI(**uint256** _slot) **external view returns** (**string memory**);

```
{
  "title": "Slot Metadata",
  "type": "object",
  "properties": {
    "name": {
      "type": "string",
      "description": "Identifies the asset category to which this slot represents"
    },
    "description": {
      "type": "string",
      "description": "Describes the asset category to which this slot represents"
    },
    "image": {
      "type": "string",
      "description": "Optional. Either a base64 encoded imgae data or a URI pointing to a resource wit
    },
    "properties": {
      "type": "array",
      "description": "Each item of `properties` SHOULD be organized in object format, including name,
      "items": {
        "type": "object",
        "properties": {
          "name": {
            "type": "string",
            "description": "The name of this property."
          },
          "description": {
            "type": "string",
            "description": "Describes this property."
          }
```

设计思路总结

- Semi-fungible：<ID，SLOT，Value>三元组，“Address-ID-Value”三层结构
- 转账与授权：使ID成为一个接近独立的概念，实现了ID-to-ID的转账，以及针对ID中部分value的授权
- 转账通知模型：改进了ERC-721/ERC-1155设计的问题，“接收者说了算”的最灵活模式
- 元数据描述：增加了与Opensea等主流平台兼容/部分兼容的描述，方便应用层获取和展示更多信息；针对SLOT有特殊的元数据描述，方便体现SLOT的设计理念

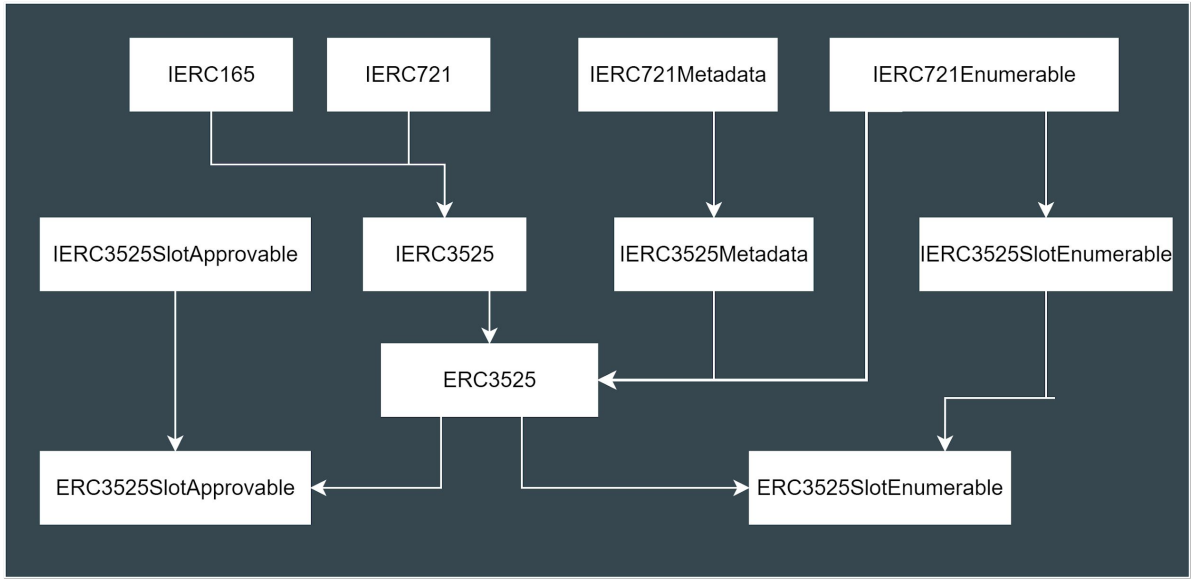
一个完整的Token标准协议（非应用协议），包含了迄今为止主流Token协议的主要功能与特点，并且针对Token标准发展过程中的各种问题做了最新的改进与适应；也使得整个标准比较复杂。

ERC-3525主要能力

- **同质化能力**
  - 同质化能力是指一种商品或物品与其他单个同类商品或物品互换的能力，这将直接决定资产的流动性以及金融效率
  - 在同一个种类（Slot）下可以对Value进行拆分合并，也可以对某个Slot通过AMM进行交易
- **多层次多维度结构能力**
  - 通过合约、Slot、ID三个维度，可以方便的描述现有世界中的大多数资产
  - 通过Slot的定义，可以通过struct包含多个关键、非关键属性，能对结构化资产方便建模
- **可视化能力**
  - 让复杂的资产结构也能通过图形动态直观的展现出来
  - 动态可视化可以根据合约状态实时变化
- **资产容器能力**
  - 每个Slot可以是一个Container，里面可以包含一种或多种底层资产（比如10ETH+20BTC），Value可以代表份额
  - 结合以上的三个能力，可以方便、透明、直观的形成可编程的资产容器

ERC-3525参考实现

https://github.com/solv-finance/erc-3525



- 内嵌ERC721实现
- 提供 ERC721Medata, ERC3525Metadata and ERC721Enumerable缺省实现
- 优化数据存储以节省gas
- 预留hook方便业务扩展
- 已完成合约代码审计

ERC-3525 GET STARTED for developer

https://medium.com/solv-blog/erc-3525-starter-kit-developer-edition-9d734ca62bd0

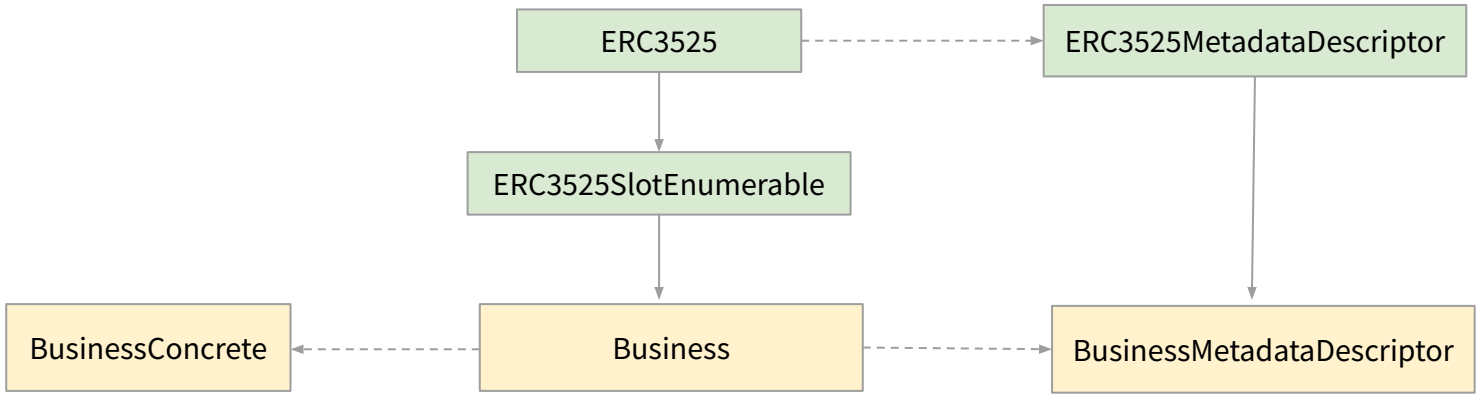
erc-3525 / examples / getting-started / contracts / ERC3525GettingStarted.sol

YeeTsai update README && add getting-started examples

Code Blame 44 lines (37 loc) · 2.05 KB

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.9;

import "@openzeppelin/contracts/utils/Strings.sol";
import "@solvprotocol/erc-3525/ERC3525.sol";

// Uncomment this line to use console.log
// import "hardhat/console.sol";

contract ERC3525GettingStarted is ERC3525 {
    using Strings for uint256;

    address public owner;

    constructor(address owner_) ERC3525("ERC3525GettingStarted", "ERC3525GS", 18) {
        owner = owner_;
    }

    function mint(address to_, uint256 slot_, uint256 amount_) external {
        require(msg.sender == owner, "ERC3525GettingStarted: only owner can mint");
        _mint(to_, slot_, amount_);
    }

    function tokenURI(uint256 tokenId_) public view virtual override returns (string memory) {
        return string(
            abi.encodePacked(
                '<svg width="600" height="600" xmlns="http://www.w3.org/2000/svg">',
                ' <g> <title>Layer 1</title>',
                '  <rect id="svg_1" height="600" width="600" y="0" x="0" stroke="#000" fill="#000000"/>',
                '  <text xml:space="preserve" text-anchor="start" font-family="Noto Sans JP" font-size="24" id="svg_2" y="340" x="200" stroke-width="0" stroke="#000" fill="#ffffff">TokeId: ',
                tokenId_.toString(),
                '</text>',
                '  <text xml:space="preserve" text-anchor="start" font-family="Noto Sans JP" font-size="24" id="svg_3" y="410" x="200" stroke-width="0" stroke="#000" fill="#ffffff">Balance: ',
                balanceOf(tokenId_).toString(),
                '</text>',
                '  <text xml:space="preserve" text-anchor="start" font-family="Noto Sans JP" font-size="24" id="svg_3" y="270" x="200" stroke-width="0" stroke="#000" fill="#ffffff">Slot: ',
                slotOf(tokenId_).toString(),
                '</text>',
                '  <text xml:space="preserve" text-anchor="start" font-family="Noto Sans JP" font-size="24" id="svg_4" y="160" x="150" stroke-width="0" stroke="#000" fill="#ffffff">ERC3252 GETTING STARTED</text>',
                ' </g> </svg>'
            )
        );
    }
}
```

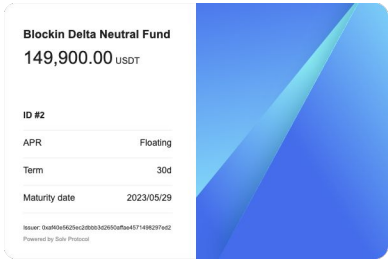
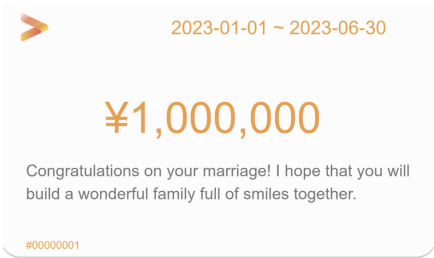
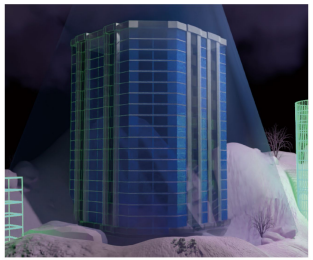
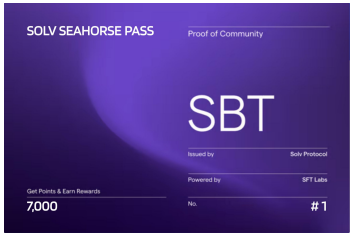
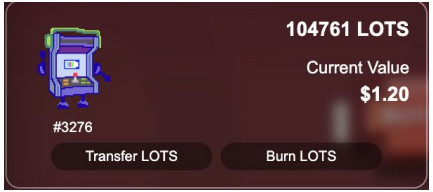
SFT应用典型合约架构



ERC-3525应用开发注意事项

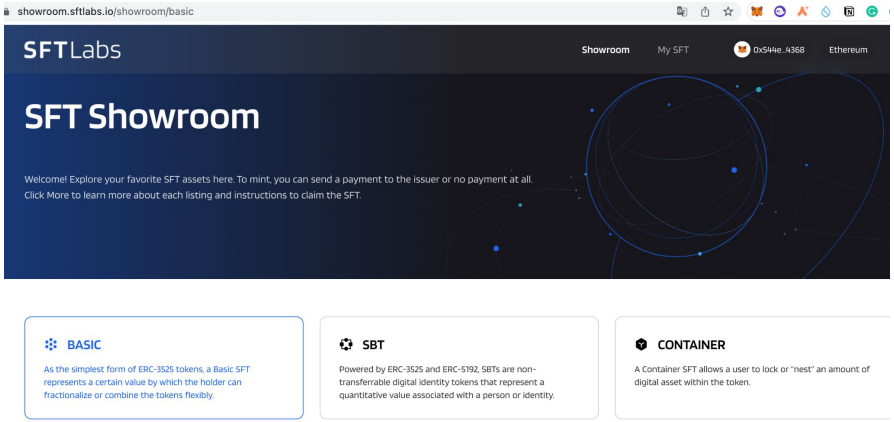
- 如果业务逻辑复杂，可能会遇到代码大小限制的问题，建议采用分合约的模式，将业务逻辑或者数据存储等合约单独部署
- Slot可以通过多个关键要素计算，类似uint256(keccak256(abi.encode(chainId, contractAddress, factorA, factorB, …)))
- 如果SVG中通过background的模式嵌入url，需要通过svg标签渲染，在opensea等用img渲染的平台可能无法显示url底图

ERC-3525部分应用案例（排名不分先后）

Solv Protocol 一站式基金投资平台		**CodeFox** Web3礼品卡	
InVar Finance RWA通证化		**SFT Labs** SBT门票/纪念卡	
Solv Seahorse 会员积分卡		**XWINNER** 去中心化游戏平台	

SFT Labs Showroom

https://showroom.sftlabs.io/



RWA Tokenization Model

Free to Mint

About RWA SFT

ERC-3525 release huge flexibility and re-innovation potential for real-world-asset tokenization, and a better solution for the NFT financialization. InVar Finance adopted ERC-3525 as novel medium to tokenized real data and value on-chain, also for further use cases development....

MINT NOW →

文理两开花播客听友见面会（香港）

Pay to Mint

《文理两开花》终于要和大家线下见面啦！

初次见面，我们特别选在了香港web3盛大节日期间，在香港中环地标"百年冰窖"——藝穗會（Fringe Vault）里，与听友相聚畅聊！

..

MINT NOW →

SOLV SEAHORSE PASS

Free to Mint

About this SBT:
Solv Seahorse Pass is like a unique NFT, but stores points you earn from participating in events in the Solv Protocol community -- think of it as a Solv loyalty card! Start building up your points to unlock exclusive benefits for core users like you.

MINT NOW →

参考资料

ERC-3525详细规范

ERC-3525参考实现

Awesome ERC-3525

ERC-3525 Starter Kit: Developer Edition

Explaining ERC-3525: What it is and How it Works

ERC-3525 通过倒计时丨SFT 是什么？有什么用？

万物研究院：从ERC20、721、1155到3525，详述RWA迈向Web3 Mass Adoption之路

solv

THANKS

- @SolvProtocol
- t.me/SolvProtocol/
- @Solv Protocol Tea
- @solv-finance
- @SolvProtocol

- https://github.com/YeeTsai
- https://twitter.com/yee2079