Robot

- 游戏基本介绍
- DF开发初探
- DF开发进阶 之 零知识证明的应用
- DF开发进阶 之 EIP2535（diamond pattern）的应用
- 游戏机制设计的探讨
- ordengg VS dfdao 太空传奇
- 游戏运行展示

- fully on-chain game
- zkSNARKs to hide information
- MMO RTS
- permissionless interoperability

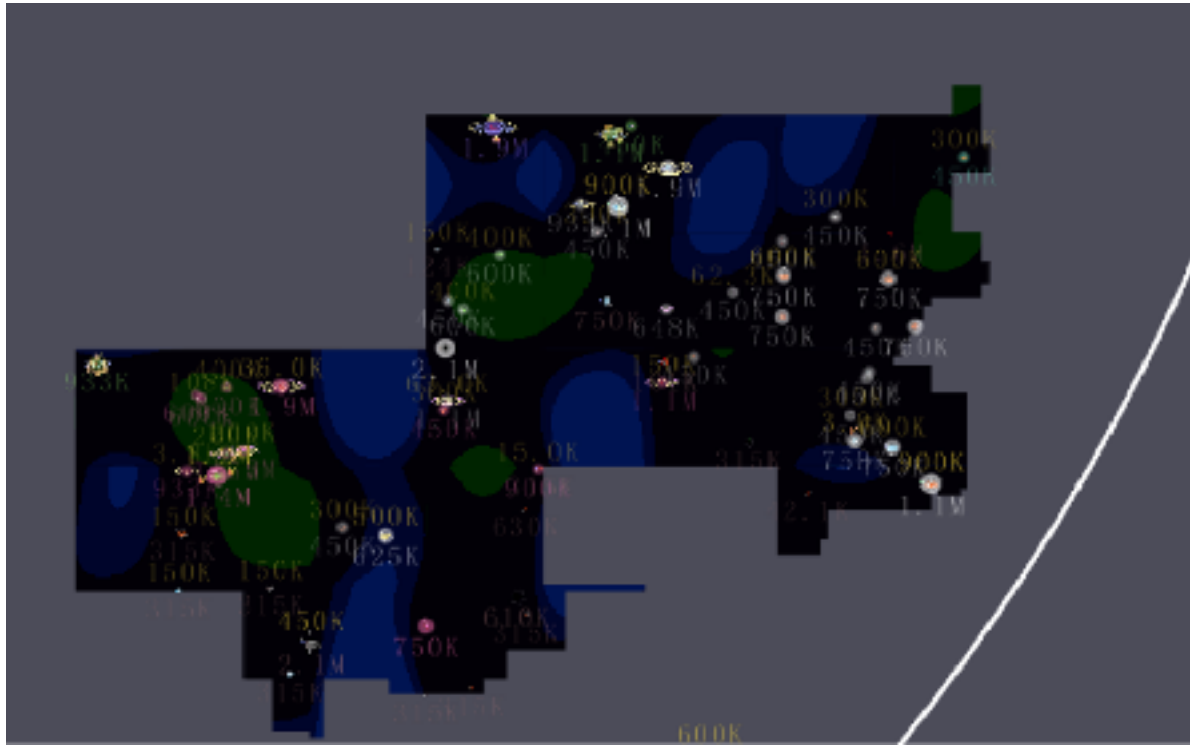
游戏的基本介绍

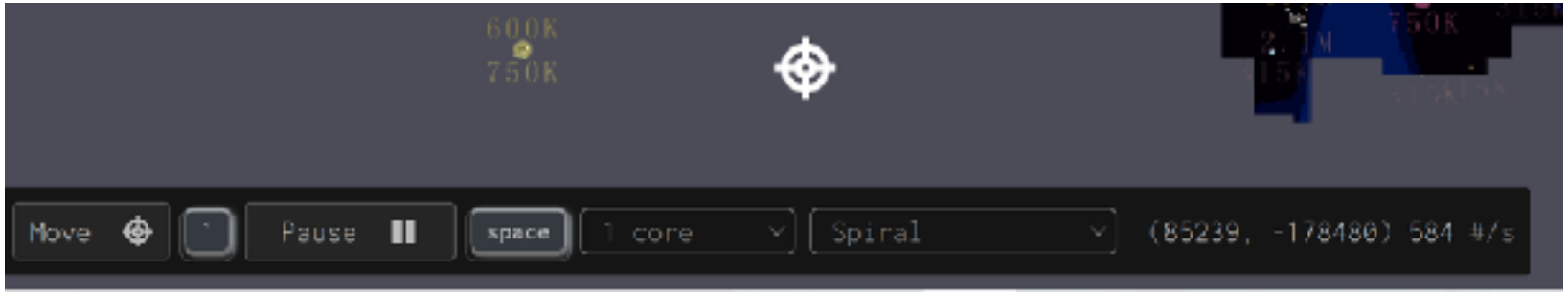
- 黑暗森林： 三年全链上游戏的启示录
- https://mirror.xyz/dfarchon.eth/XCJor0YF0lUMzB7B4xZSjkXhIWmEN8HOMqvLz8K_hqY

游戏的基本介绍

浏览器 挖地图
Rust 挖地图
GPU挖地图 强烈推荐df-explorer

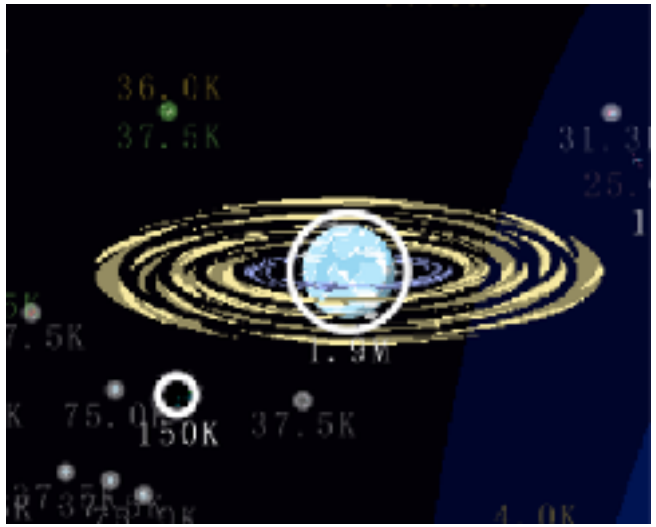
https://github.com/guild-w/df-explorer

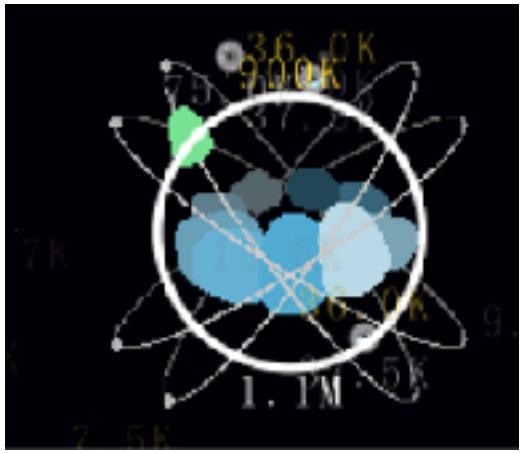


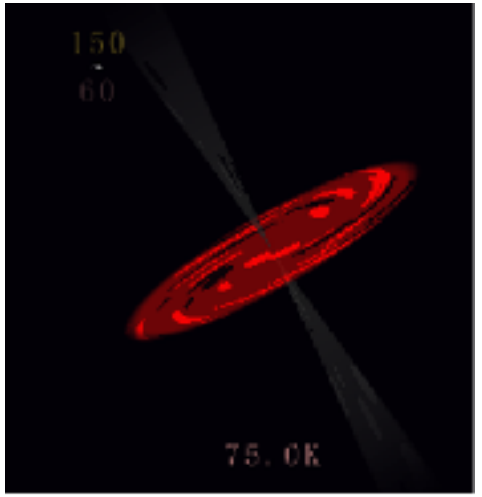


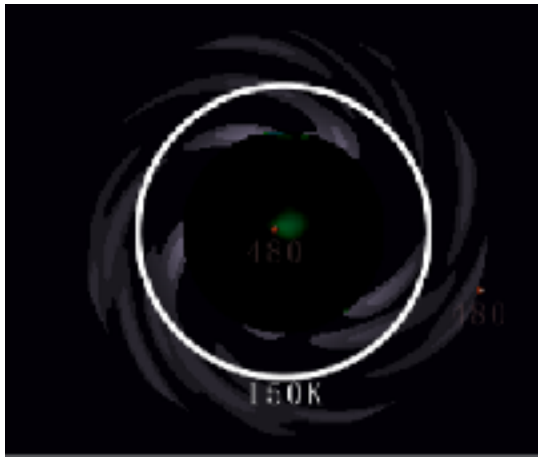
游戏基本介绍

- 星球种类





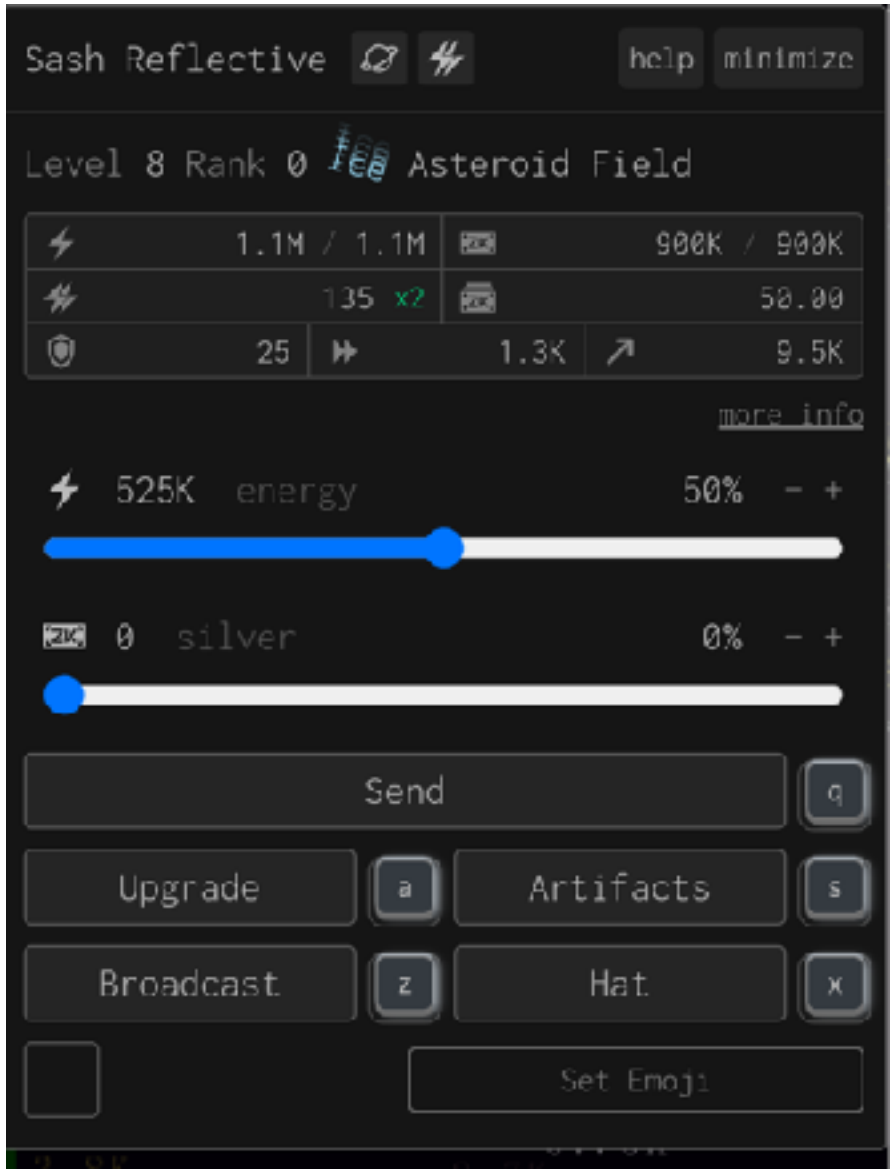




游戏基本介绍

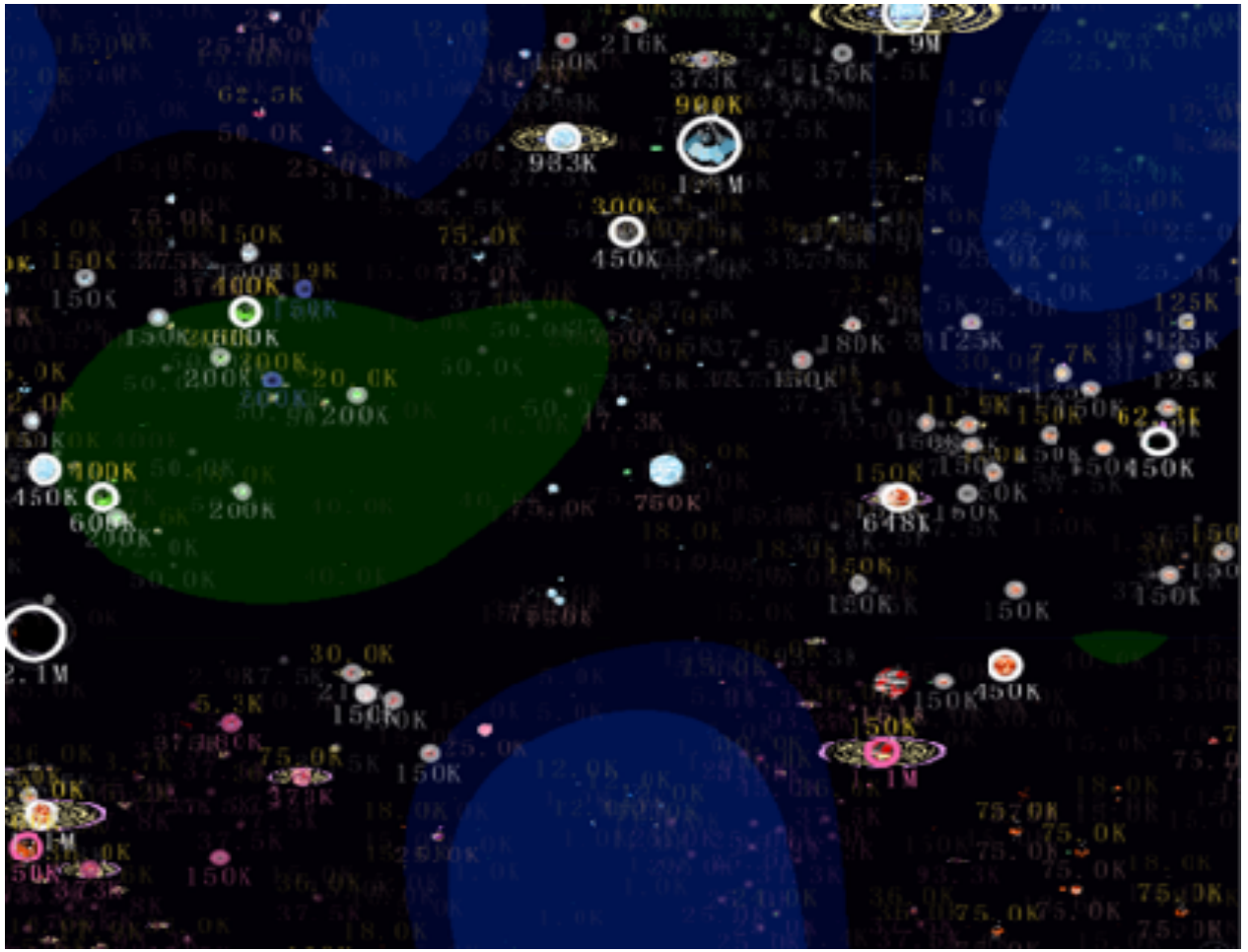
星球属性

- 能量
- 银矿
- 攻击范围
- 速度



游戏基本介绍

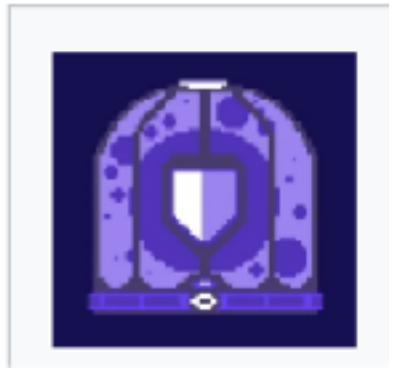
- 星球所在位置对于星球的影响



游戏基本介绍

- 神器



初探黑森开发

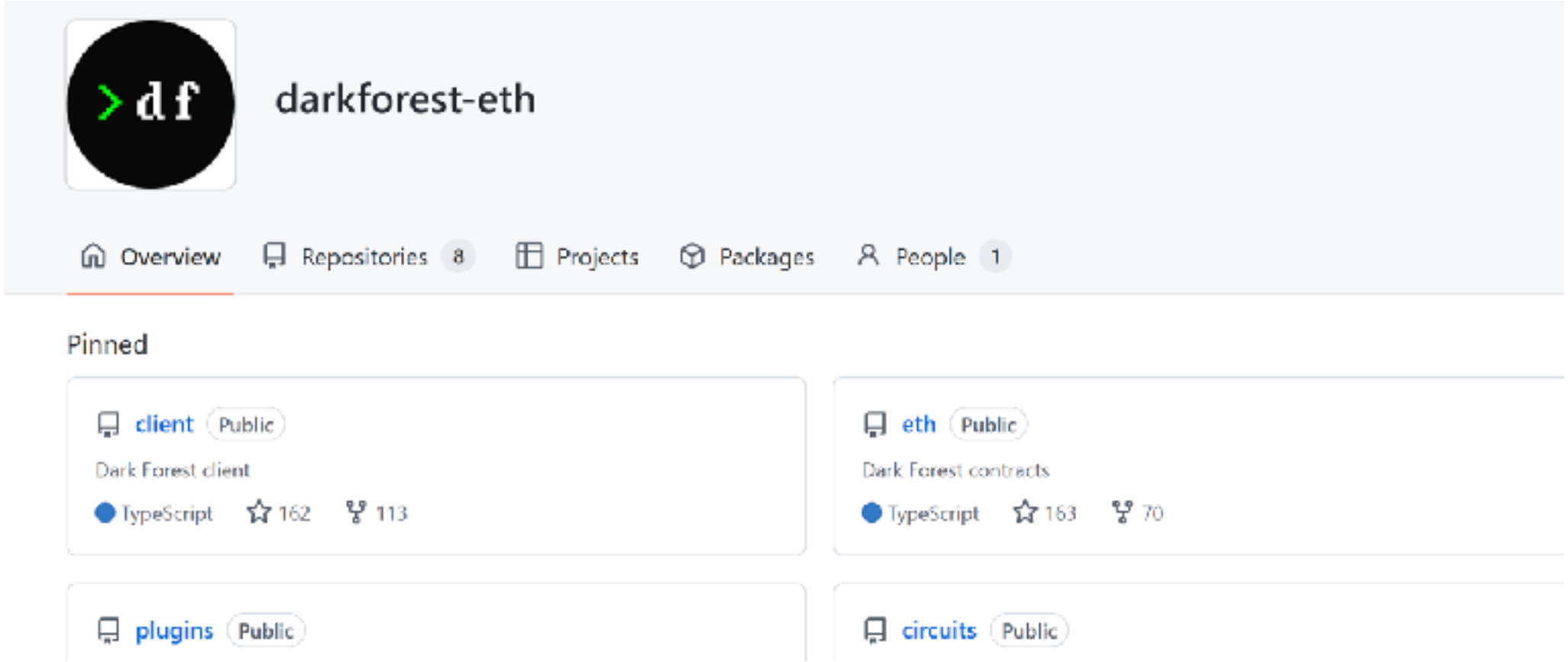
- plugins.zkga.me
- 社区维护的开源的插件

277dao维护的中文插件介绍

https://plugins.277dao.com/

初探黑森开发

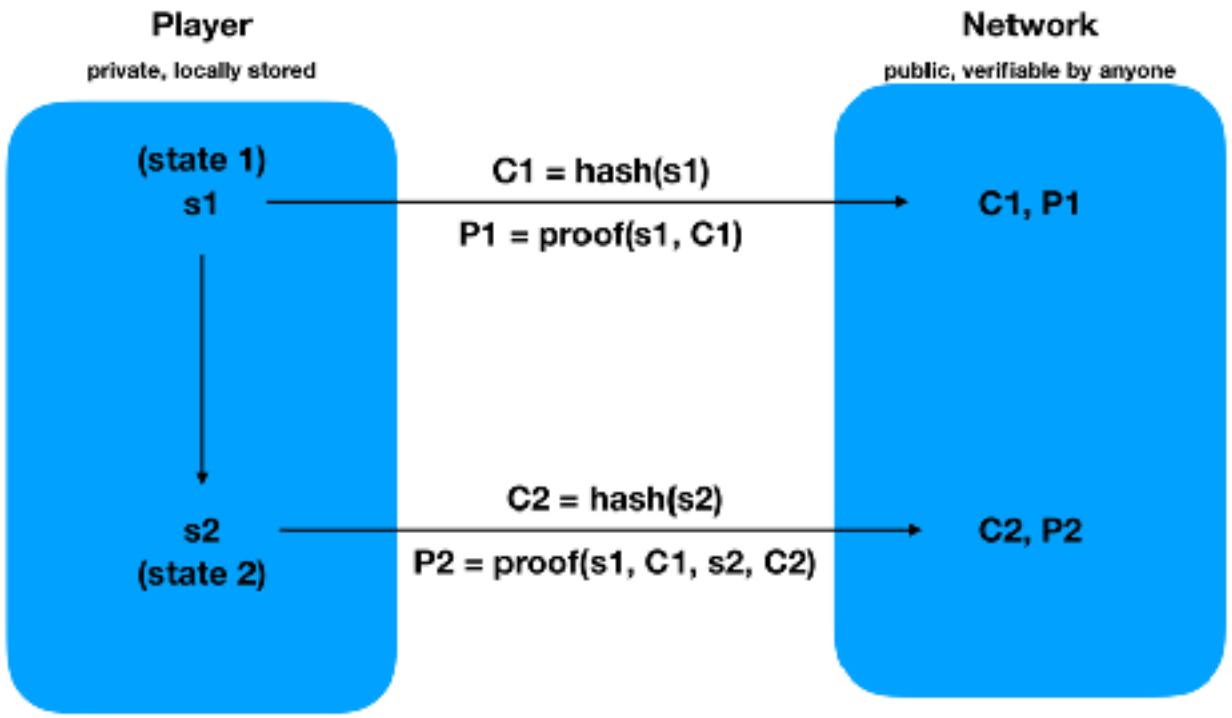
- https://github.com/darkforest-eth/



初探黑森开发

- https://dev-guides.zkga.me/
- https://blog.zkga.me/
- Join dark forest office discord
- https://github.com/darkforest-eth/darkforest-v0.6

零知识证明在黑森当中的应用



Private state on a decentralized system, with ZK Proofs

零知识证明在黑森当中的应用

在不泄露星球坐标的情况下，证明init 和 move 是有效的。

https://github.com/darkforest-eth/circuits

https://github.com/iden3/snarkjs

A zkSNARK is a gadget that can be used to generate a ZKP for any mathematical function

哈希函数 $h(x) = x^3 - x + 7$

我们可以使用zkSNARK生成一个h的ZKP

利用ZKP可以在不泄露*的情况证明我们知道值 * 使得h(*)=67,

Succinct: create proof in linear time, and other can verify it in constant time

non-interactive: without needing to ask additional question of us.

零知识证明在黑森当中的应用

locationId = MIMC（x，y）

MIMC 是有一个“SNARK-friendly”的哈希函数，

比较容易的能够生成ZKPs。

The init circuit

$$MiMC(x, y) = h$$

$$x^2 + y^2 \leq r^2$$

零知识证明在黑森当中的应用

- https://blog.zkga.me/announcing-darkforest
- https://blog.zkga.me/df-init-circuit
- https://blog.zkga.me/intro-to-zksnarks

- 黑森circuits的具体实现
- https://github.com/darkforest-eth/circuits

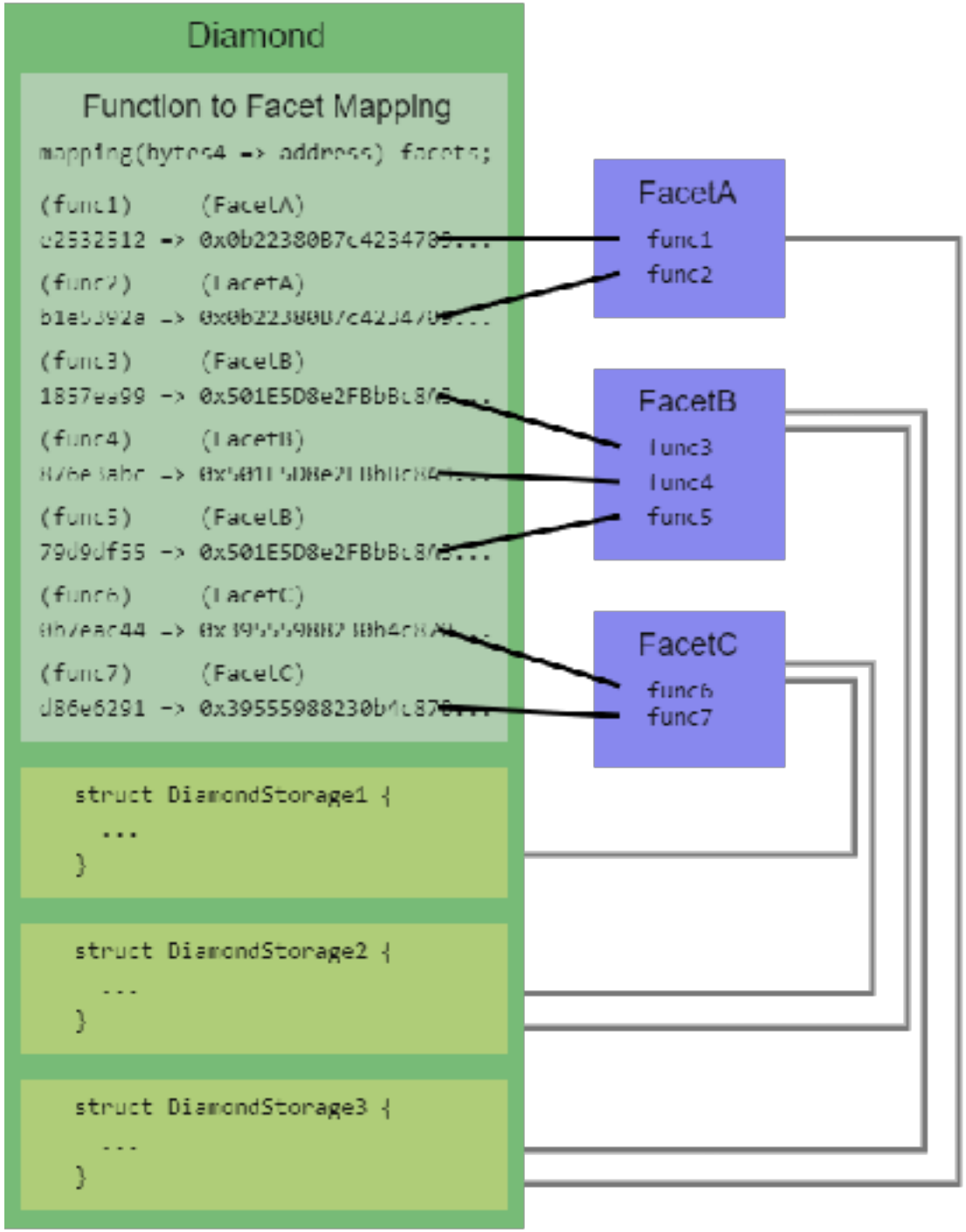
Diamond pattern在黑森当中的应用

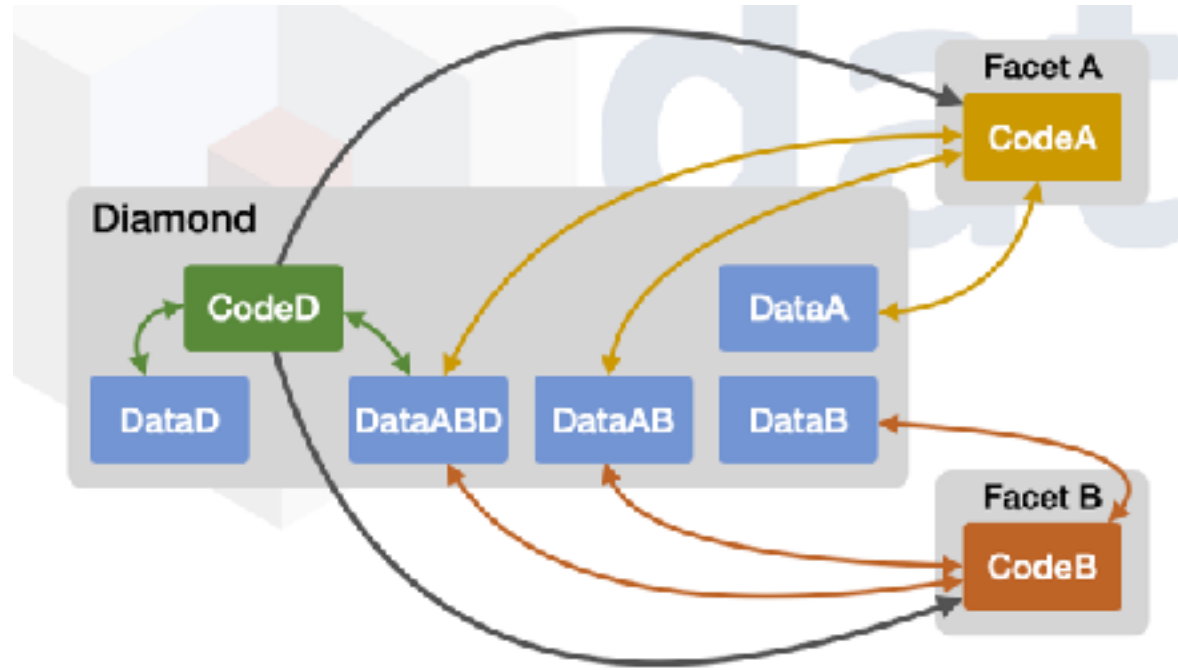
- v0.6.5 当中应用了diamond标准
- EIP-2535

- one storage space for all state variables
- One Ethereum address from which I could design and implement all functions without bytecode size limitation.
- All functions to read and write to state variables directly, easily and in the same way.
- Seemless upgrade functionality: to be able to replace functions, remove them and add new functionality without needing to redeploy everything.

Diamond在黑森当中的应用

- **Diamond**: a contract that implements the diamond standard is called a diamond.
- **Facets**: each contract that a diamond borrows functions from a different side or “facet”.
- **Diamondcut**: is used to add, replace, or remove facets and functions.
- **The Loupe**: return information about what facets and functions exist in a diamond.


Diamond
Function to Facet Mapping
mapping(bytes4 => address) facets;
(func1) (FacetA)
(func2) (FacetA)
(func3) (FacetB)
(func4) (FacetB)
(func5) (FacetB)
(func6) (FacetC)
(func7) (FacetC)
FacetA
func1
func2
FacetB
func3
func4
func5
FacetC
func6
func7
struct DiamondStorage1 {
...
}
struct DiamondStorage2 {
...
}
struct DiamondStorage3 {
...
}


Facet A
CodeA
Diamond
CodeD
DataA
DataD
DataABD
DataAB
DataB
Facet B
CodeB

Diamond在黑森当中的应用

- 黑科技： 在library当中使用状态变量

```
// This contract uses the library to set and retrieve state variables
contract ContractA {

  function setState() external {
    Library.setStateVariables(address(this), "My Name");
  }

  function getState()
    external
    view
    returns (address contractAddress, string memory name)
  {
    contractAddress = Library.contractAddress();
    name = Library.name();
  }
}
```

Diamond在黑森当中的应用

```solidity
// SPDX-License-Identifier: MIT
pragma solidity ^0.7.0;

// This library has the state variables 'contractAddress' and 'name'
library Library {

  // defining state variables
  struct DiamondStorage {
    address contractAddress;
    string name;
    // ... any number of other state variables
  }

  // return a struct storage pointer for accessing the state variables
  function diamondStorage()
    internal
    pure
    returns (DiamondStorage storage ds)
  {
    bytes32 position = keccak256("diamond.standard.diamond.storage");
    assembly { ds.slot := position }
  }

  // set state variables
  function setStateVariables(
    address _contractAddress,
    string memory _name
  )
    internal
  {
    DiamondStorage storage ds = diamondStorage();
    ds.contractAddress = _contractAddress;
    ds.name = _name;
  }

  // get contractAddress state variable
  function contractAddress() internal view returns (address)
    return diamondStorage().contractAddress;
  }

  // get name state variable
  function name() internal view returns (string memory) {
    return diamondStorage().name;
  }
}
```

Diamond在黑森当中的应用

- delegatecall
- https://eip2535diamonds.substack.com/p/understanding-delegatecall-and-how

Diamond在黑森当中的应用

- https://github.com/mudgen/Diamond

Implementation	diamondCut complexity	diamondCut gas cost	loupe complexity	loupe gas cost
diamond-1	low	medium	medium	high
diamond-2	high	low	high	high
diamond-3	medium	high	low	low

Diamond在黑森当中的应用

- https://github.com/darkforest-eth/eth

- solidstate库 和 diamond 匹配比较好
- https://github.com/solidstate-network/solidstate-solidity

Diamond在黑森当中的应用

- https://eips.ethereum.org/EIPS/eip-2535
- https://eip2535diamonds.substack.com/p/introduction-to-the-diamond-standard
- https://dev.to/mudgen/ethereum-s-maximum-contract-size-limit-is-solved-with-the-diamond-standard-2189

- https://dev.to/mudgen/solidity-libraries-can-t-have-state-variables-oh-yes-they-can-3ke9
- https://dev.to/mudgen/how-diamond-storage-works-90e

游戏机制探讨

V0.6.3 距离中心最近的星球来排名

V0.6.4 黑洞withdraw银矿的积分 + 开神器的奖励 来排名

v0.6.5 增加了飞船 spaceship

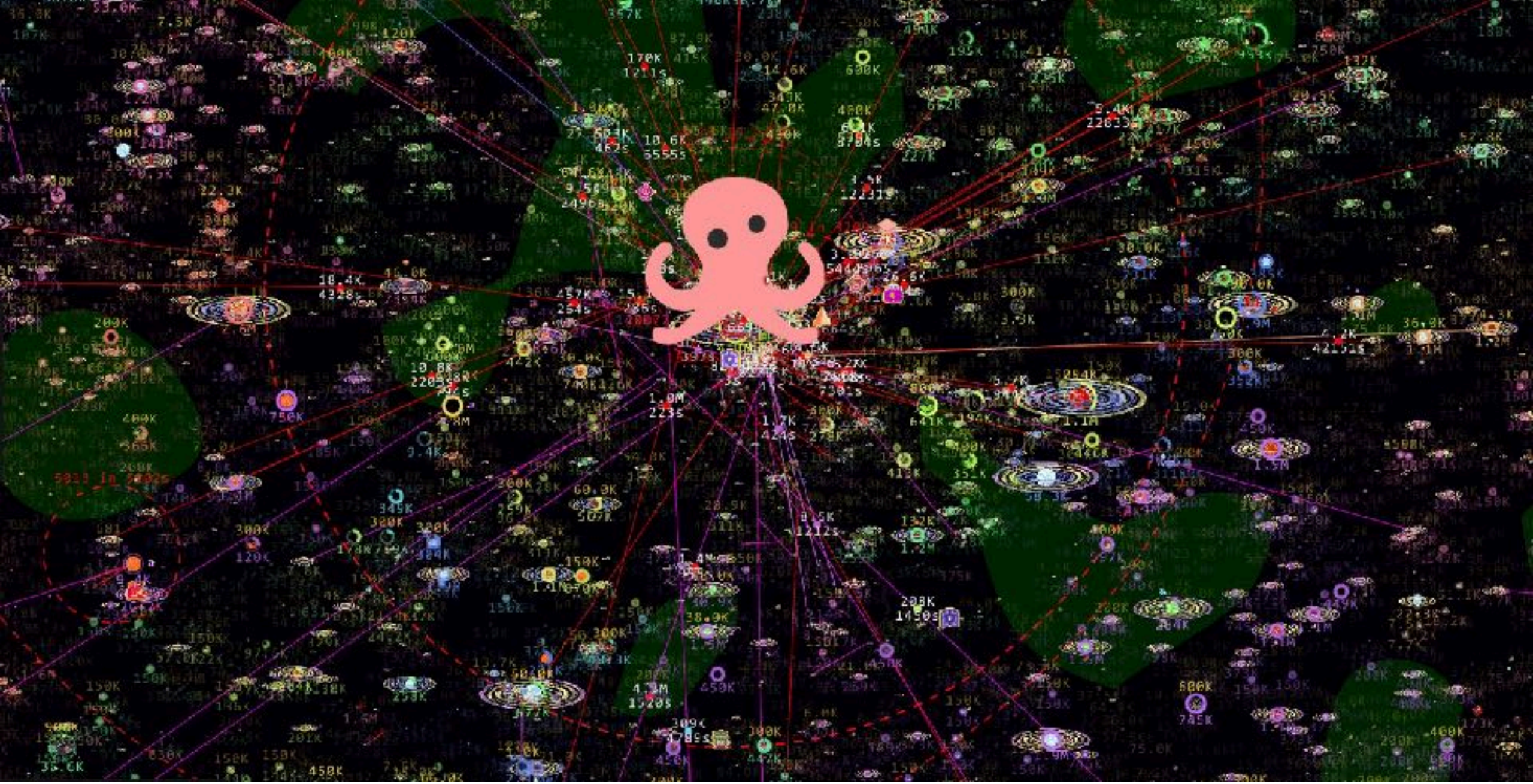
Mothership：使行星的能量恢复速度加倍

Whale：使行星的银生产速度加倍

Crescent：将一个无主的行星变成一个小行星场

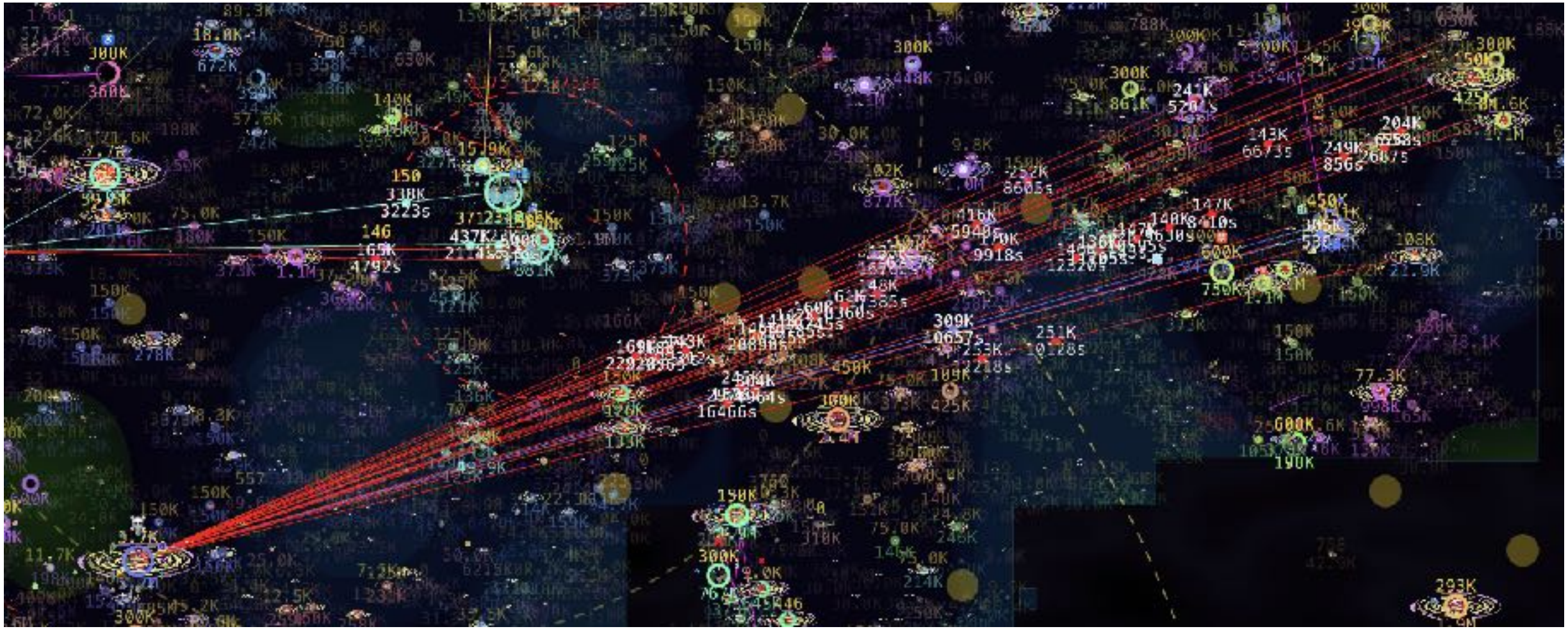
Gear：允许您在您控制的铸造厂上勘探和寻找文物

Titan：停止所在行星的能量/银再生（包括你自己的星球）



v0.6.3

游戏机制设计的探讨



v0.6.5

dfdao vs ordengg 太空传奇

https://mirror.xyz/vanishk.eth/TgEzC8MvAF2J5QBU61Xl1OsmgRVNX7mMOCA55GvRIh4

DF ARTEMIS

- **关于ARTEMIS的起源、现在和想象力的空间**https://mirror.xyz/dfarchon.eth/VoJQxsUPlBYJrLcgWutAFaZh0do5u_5hwAikTeZ9jSc

funder
manager
mercenary

rebuild **CHAIN Of SUSPICION** 😈


DF ARTEMIS
minimize
close
Welcome
Funder
Manager
Mercenary
Admin
Creator
DF-ARTEMIS
v0.0.2
the smart contract and plugin have not been audited
you may lost your xdai or private key
USE PLUGIN AT YOUR OWN RISK
Creator Fee: 5 %
Admin Fee: 5 %
manager Fee: 5 %
Balance: 0.0 xDai
Get Support
58.8K
410s
645K
164K
150K
75.0K
100K
2.0M
200K
16.1K
87.7K

And More

欢迎关注 DF ARES ——

Dark Forest Community Rounds hosted by DF Archon

目前公开的信息： https://dfares.notion.site/DF-ARES-Main-ca9238c3fee8418d933079d5d28eab3b

欢迎加入玩家群获取更多新鲜资讯

(强烈推荐）discord：https://discord.com/invite/vXCJKfT6a2

wechat group：👉👉👉

群聊：DF ARES 玩家群

该二维码7天内(5月12日前)有效，重新进入将更新