

# Week 17 - 10.31

## Day1 - EIP1967

delegatcall

EIP-1967

```
=>
bytes32(uint256( keccak256("eip1967.proxy.implementation") - 1))

event Upgraded(address indexed implementation);

=> beacon
bytes32(uint256( keccak256("eip1967.proxy.beacon") - 1))

event BeaconUpgraded(address indexed beacon);

=> admin
bytes32(uint256( keccak256("eip1967.proxy.admin") - 1))

event AdminChanged(address indexed previousAdmin, address newAdmin);
```

hash slot

slot

EIP1967 <https://eips.ethereum.org/EIPS/eip-1967>

## Day2 - ERC1967Proxy

ERC1967Proxy EIP1967 openzeppelin

[ERC1967Proxy](#) [ERC1967Proxy](#)

FUNCTIONS

```
constructor( _logic, _data)
```

```
_implementation()
```

## ERC1967UPGRADE

```
_getImplementation()
```

```
_upgradeTo( newImplementation)
```

```
_upgradeToAndCall( newImplementation, data, forceCall)
```

```
_upgradeToAndCallUUPS( newImplementation, data, forceCall)
```

```
_getAdmin()
```

```
_changeAdmin( newAdmin)
```

```
_getBeacon()
```

```
_upgradeBeaconToAndCall( newBeacon, data, forceCall)
```

## PROXY

```
_delegate( implementation)
```

```
_fallback()
```

```
fallback()
```

```
receive()
```

```
_beforeFallback()
```

## EVENTS

```
ERC1967UPGRADE
```

```
Upgraded( implementation)
```

```
AdminChanged( previousAdmin, newAdmin)
```

BeaconUpgraded( beacon)

# Day3 - Transparent Proxy

Openzeppelin

Transparent Proxy

EIP1967Proxy

- 1.
2. “admin cannot fallback to proxy t

ProxyAdmin

[https://docs.openzeppelin.com/contracts/4.x/api/proxy#transparent\\_proxy](https://docs.openzeppelin.com/contracts/4.x/api/proxy#transparent_proxy)

# Day4 - UUPS Proxy

upgradeTo( address newImpl)

Box

```
contract Box {
    uint256 private _value;

    function store(uint256 value) public { /*..*/ }

    function retrieve() public view returns (uint256) { /*..*/ }
}

contract BoxProxy {

    function _delegate(address implementation) internal virtual { /*..*/ }

    function getImplementationAddress() public view returns (address) { /*..*/ }
```

```

    fallback() external { /*..*/ }

    // Upgrade logic in Proxy contract
    upgradeTo(address newImpl) external {
        // Changes stored address of implementation of contract
        // at its slot in storage
    }
}

```

# UUPS Proxy

UUPS      EIP1822      UUPS      Transparent      Pro

UUPS      OpenZeppelin UUPSUpgradeable

```

contract Box is UUPSUpgradeable {
    uint256 private _value;

    function store(uint256 value) public { /*..*/ }

    function retrieve() public view returns (uint256) { /*..*/ }

    // Upgrade logic in Implementation contract
    upgradeTo(address newImpl) external {
        // Changes stored address of implementation of contract
        // at its slot in storage
    }
}

contract BoxProxy {

    function _delegate(address implementation) internal virtual { /*..*/ }

    function getImplementationAddress() public view returns (address) { /*..*/ }

    fallback() external { /*..*/ }

}

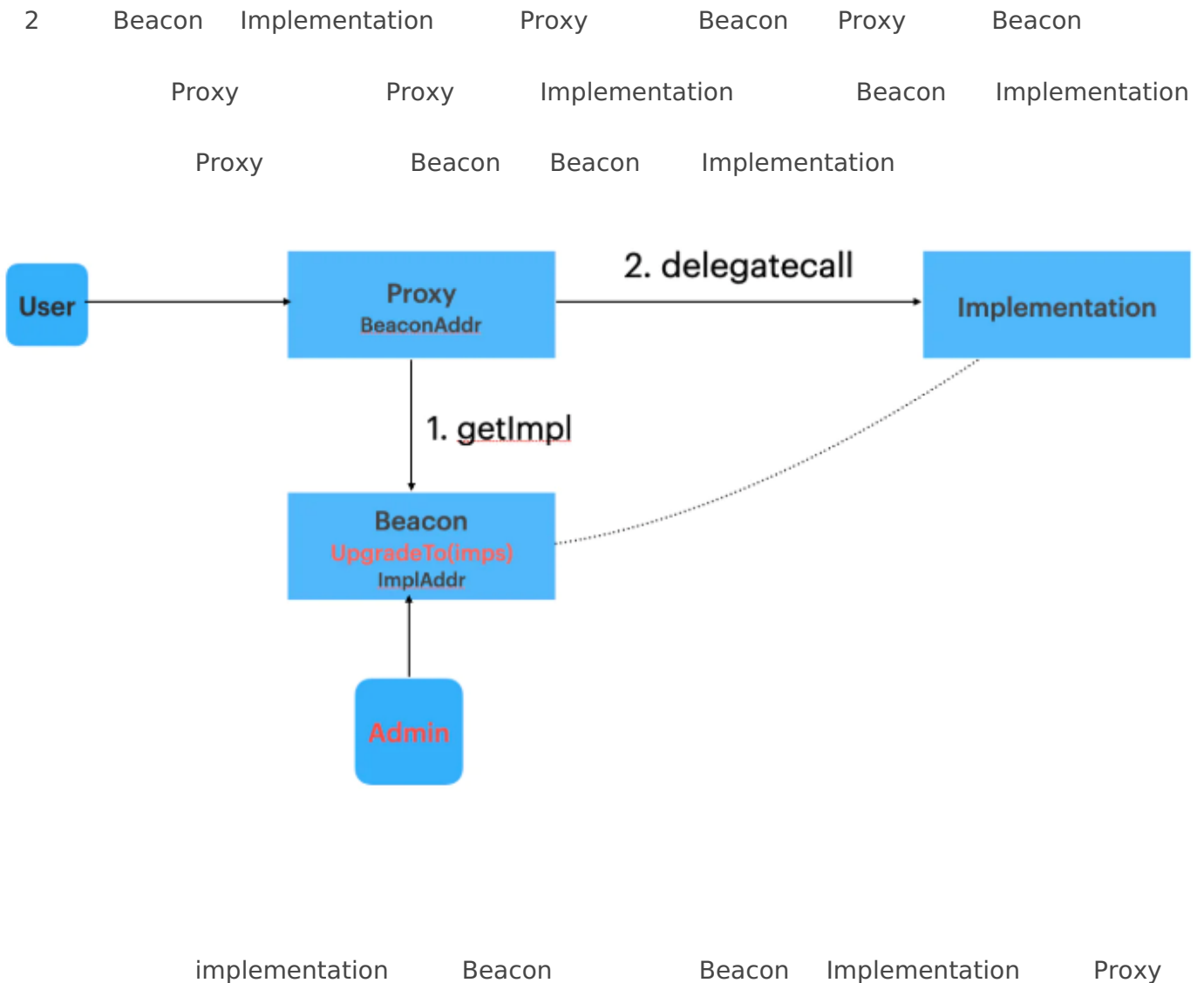
```

UUPS

UUPS

1. <https://learnblockchain.cn/article/4257>
2. <https://docs.openzeppelin.com/contracts/4.x/api/proxy#UUPSUpgradeable>

# Day5 - Beacon



1. <https://docs.openzeppelin.com/contracts/4.x/api/proxy#beacon>
2. [https://mirror.xyz/rbtree.eth/qDSQvenBZ\\_TWqZLUTlxiXVlhKQzPygRyv88EMFWxJro](https://mirror.xyz/rbtree.eth/qDSQvenBZ_TWqZLUTlxiXVlhKQzPygRyv88EMFWxJro)

Revision #5

Created 31 October 2022 11:03:06 by Dayong

Updated 4 November 2022 11:17:28 by Dayong