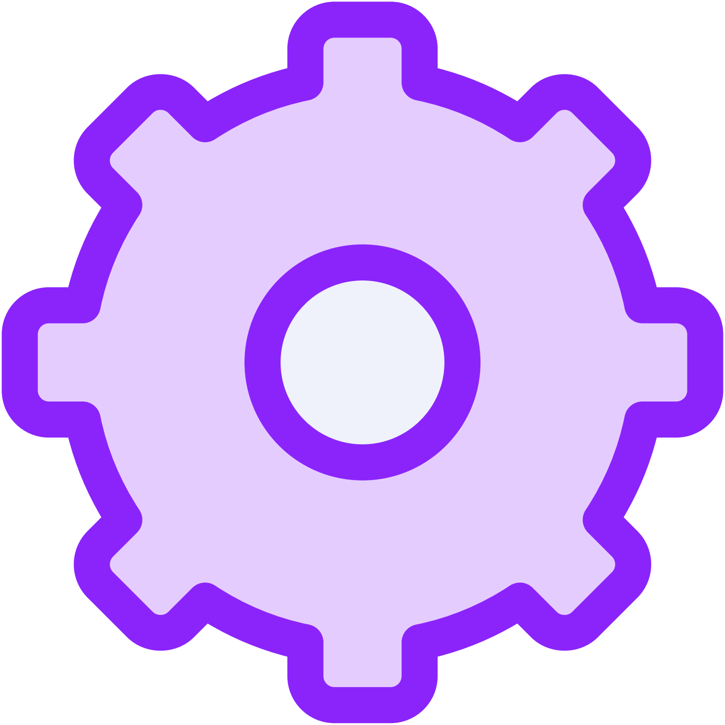
Proxy

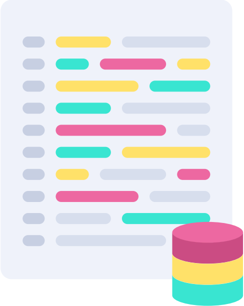
Logic

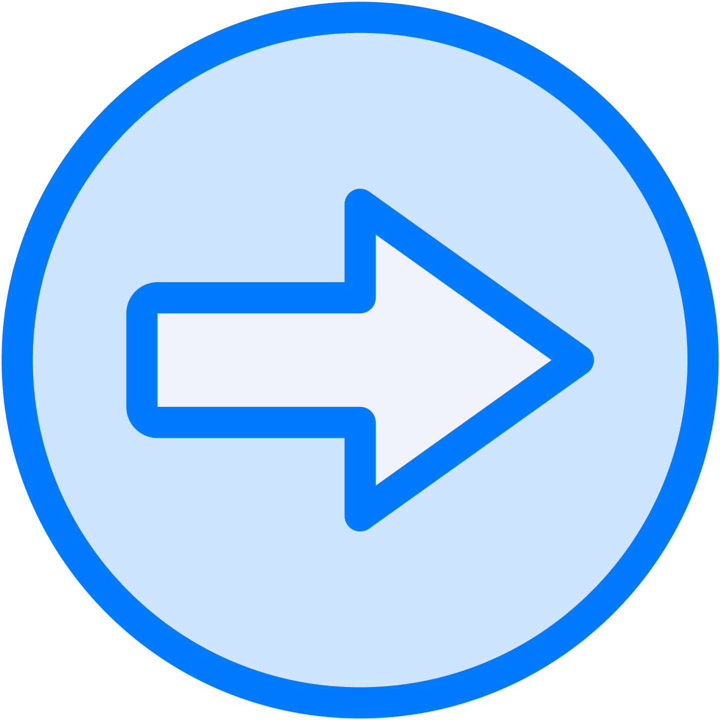
Eternal Storage

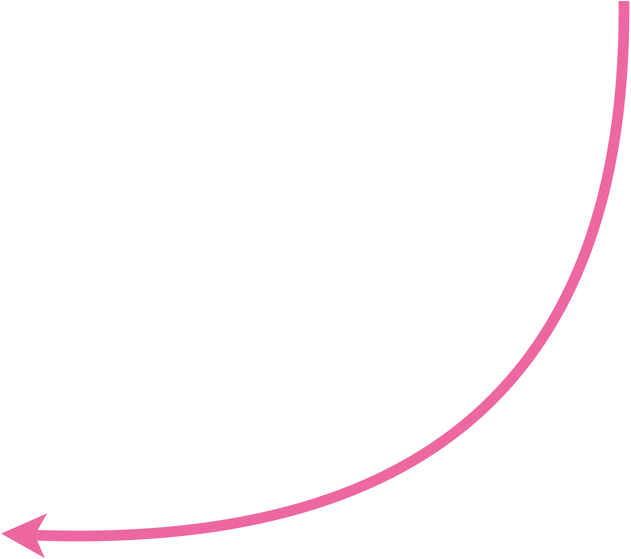
```
mapping(bytes32=>bool)
mapping(bytes32=>uint)
mapping(bytes32=>int)
mapping(bytes32=>address)
mapping(bytes32=>string)
mapping(bytes32=>bytes)

function voteForCoke() returns uint


function voteForPepsi() returns uint
```

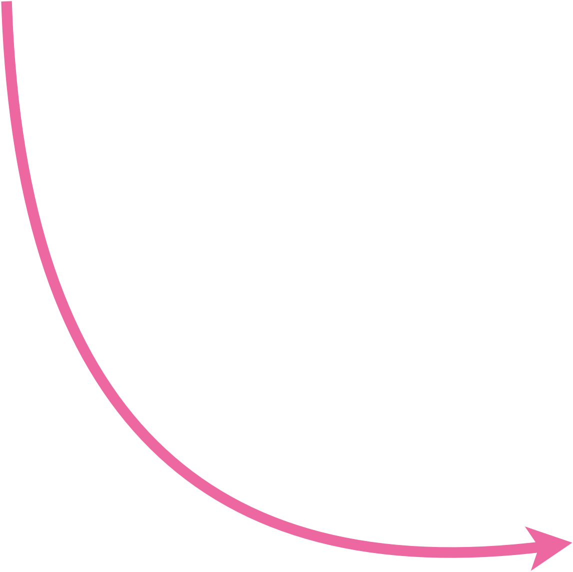


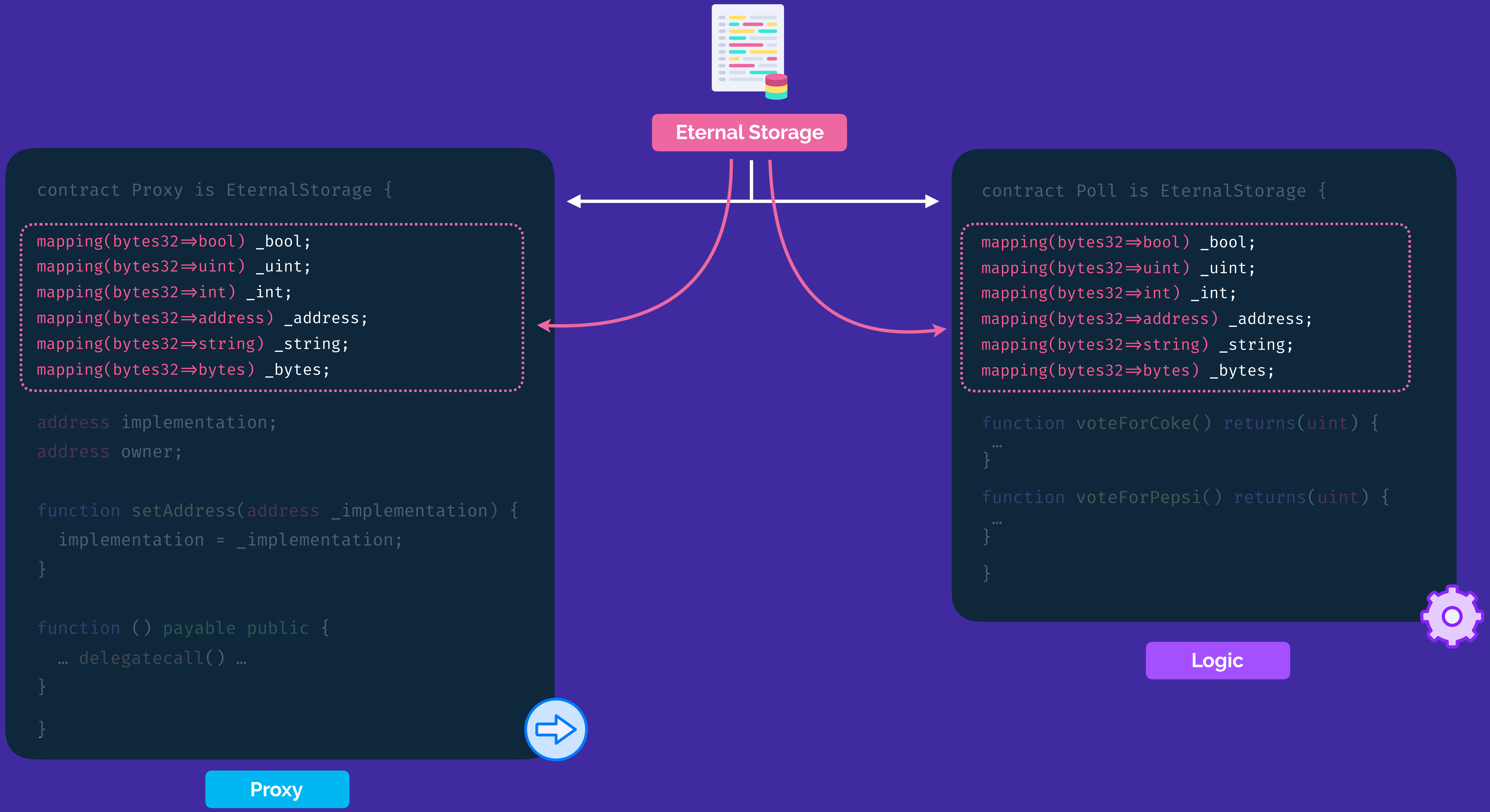
```
mapping(bytes32=>bool)
mapping(bytes32=>uint)
mapping(bytes32=>int)
mapping(bytes32=>address)
mapping(bytes32=>string)
mapping(bytes32=>bytes)

address
address

function setAddress(address                 )




function    payable public
    delegatecall
```











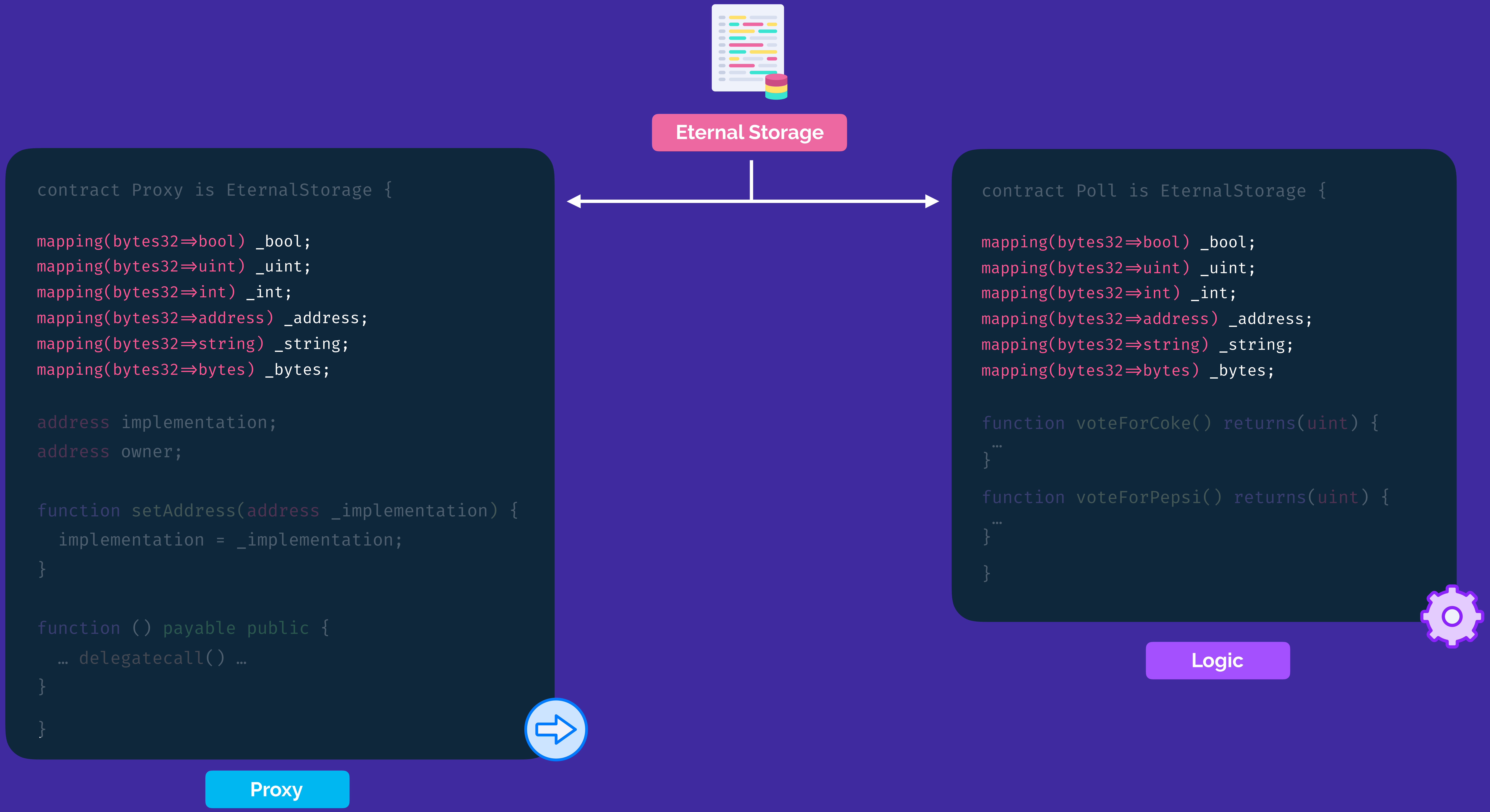
Eternal Storage

```
contract Proxy is EternalStorage {

mapping(bytes32=>bool) _bool;
mapping(bytes32=>uint) _uint;
mapping(bytes32=>int) _int;
mapping(bytes32=>address) _address;
mapping(bytes32=>string) _string;
mapping(bytes32=>bytes) _bytes;

address implementation;
address owner;

function setAddress(address _implementation) {
  implementation = _implementation;
}

function () payable public {
  … delegatecall() …
}

}
```

Proxy

```
contract Poll is EternalStorage {

mapping(bytes32=>bool) _bool;
mapping(bytes32=>uint) _uint;
mapping(bytes32=>int) _int;
mapping(bytes32=>address) _address;
mapping(bytes32=>string) _string;
mapping(bytes32=>bytes) _bytes;

function voteForCoke() returns(uint) {
  …
}

function voteForPepsi() returns(uint) {
  …
}

}
```

Logic



Eternal Storage

```
contract Proxy is EternalStorage {

mapping(bytes32=>bool) _bool;
mapping(bytes32=>uint) _uint;
mapping(bytes32=>int) _int;
mapping(bytes32=>address) _address;
mapping(bytes32=>string) _string;
mapping(bytes32=>bytes) _bytes;

address implementation;
address owner;

function setAddress(address _implementation) {
  implementation = _implementation;
}

function () payable public {
  … delegatecall() …
}

}
```

Proxy

```
contract Poll is EternalStorage {

mapping(bytes32=>bool) _bool;
mapping(bytes32=>uint) _uint;
mapping(bytes32=>int) _int;
mapping(bytes32=>address) _address;
mapping(bytes32=>string) _string;
mapping(bytes32=>bytes) _bytes;

function voteForCoke() returns(uint) {
  …
}

function voteForPepsi() returns(uint) {
  …
}

}
```

Logic