```
contract 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;

}
```

Eternal Storage

```
contract Proxy is EternalStorage {

address implementation;
address owner;

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

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

}
```

Proxy

```
contract Poll is EternalStorage {

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