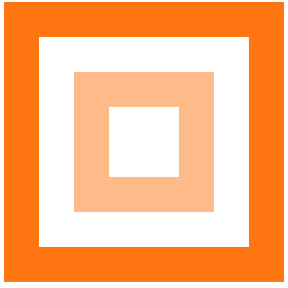
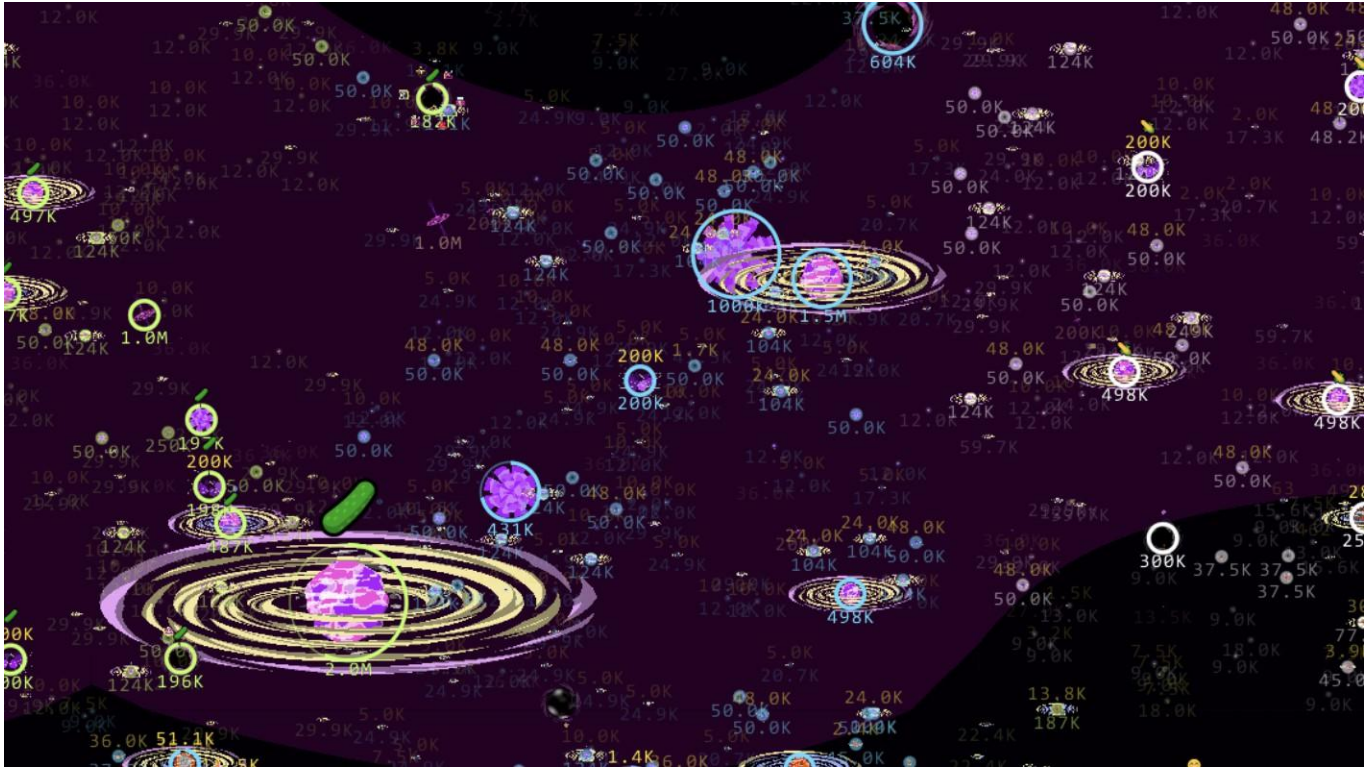


MUD

@_alvarius


50.0K
604K
124K
200K
497K
1.0M
48.0K
1000K
1.5M
104K
498K
1.7K
431K
197K
187K
2.0M
196K
300K
37.5K
13.8K
187K
36.0K
51.1K
1.4K

Game.sol

```
struct Coord {
  int32 x;
  int32 y;
}

mapping(bytes32 => Coord) positions;
```

Client.ts

Game.sol

```
struct Coord {
  int32 x;
  int32 y;
}

mapping(bytes32 => Coord) positions;
```

Client.ts

```
type Coord {
  x: number;
  y: number;
}

const positions = Map<string, Coord>();
```

Game.sol

```
struct Coord {
  int32 x;
  int32 y;
}

mapping(bytes32 => Coord) positions;

function getPosition(bytes32 entity)
  returns (Coord);
```

Client.ts

```
type Coord {
  x: number;
  y: number;
}

const positions = Map<string, Coord>();

function syncInitialPositions() {
  for(const entity of entities) {
    const coord = contract.getPosition(entity)
    positions.set(entity, coord);
  }
}
```

Game.sol

```
struct Coord {
  int32 x;
  int32 y;
}

mapping(bytes32 => Coord) positions;

function getPosition(bytes32 entity)
  returns (Coord);

event PositionChanged(
  bytes32 entity,
  Coord position
);

function move(bytes32 entity, Coord position) {
  emit PositionChanged(entity, position);
  positions[entity] = position;
}
```

Client.ts

```
type Coord {
  x: number;
  y: number;
}

const positions = Map<string, Coord>();

function syncInitialPositions() {
  for(const entity of entities) {
    const coord = contract.getPosition(entity)
    positions.set(entity, coord);
  }
}
```

Game.sol

```
struct Coord {
  int32 x;
  int32 y;
}

mapping(bytes32 => Coord) positions;

function getPosition(bytes32 entity)
  returns (Coord);

event PositionChanged(
  bytes32 entity,
  Coord position
);

function move(bytes32 entity, Coord position) {
  emit PositionChanged(entity, position);
  positions[entity] = position;
}
```

Client.ts

```
type Coord {
  x: number;
  y: number;
}

const positions = Map<string, Coord>();

function syncInitialPositions() {
  for(const entity of entities) {
    const coord = contract.getPosition(entity)
    positions.set(entity, coord);
  }
}

function onPositionChanged(
  entity: string,
  position: Coord
) {
  positions.set(entity, position);
}
```

Game.sol

```
struct Health {
  int32 health;
}

mapping(bytes32 => Health) healths;

function getHealth(bytes32 entity)
  returns (Health);

event HealthChanged(
  bytes32 entity,
  Health health
);

function heal(bytes32 entity, Health health) {
  emit HealthChanged(entity, health);
  healths[entity] = health;
}
```

Client.ts

```
type Health {
  health: number;
}

const healths = Map<string, Health>();

function syncInitialHealth() {
  for(const entity of entities) {
    const health = contract.getHealth(entity)
    healths.set(entity, health);
  }
}

function onHealthChanged(
  entity: string,
  health: Health
) {
  healths.set(entity, health);
}
```

Game.sol

```
struct Energy {
  int32 energy;
}

mapping(bytes32 => Energy) energies;

function getEnergy(bytes32 entity)
  returns (Energy);

event EnergyChanged(
  bytes32 entity,
  Energy energy
);

function boost(bytes32 entity, Energy energy) {
  emit EnergyChanged(entity, energy);
  energies[entity] = energy;
}
```

Client.ts

```
type Energy {
  energy: number;
}

const energies = Map<string, Energy>();

function syncInitialEnergy() {
  for(const entity of entities) {
    const energy = contract.getEnergy(entity)
    energies.set(entity, energy);
  }
}

function onEnergyChanged(
  entity: string,
  energy: Energy
) {
  energies.set(entity, energy);
}
```

Game.sol

```
struct Skill {
  bytes32 skill;
  uint256 level;
}

mapping(bytes32 => bytes32 => Skill) skills;

function getSkill(bytes32 entity, bytes32 name)
  returns (Skill);

event SkillChanged(
  bytes32 entity,
  Skill skill
);

function learn(bytes32 entity, Skill skill) {
  emit SkillChanged(entity, skill);
  skills[entity][skill.skill] = skill;
}
```

Client.ts

```
type Skill {
  skill: string;
  level: number;
}

const skills = Map<string, Skill>();

function syncInitialSkill() {
  for(const entity of entities) {
    const energy = contract.getSkill(entity,name)
    skills.set(entity+"/"+skill.skill, skill);
  }
}

function onSkillChanged(
  entity: string,
  skill: Skill
) {
  skills.set(entity+"/"+skill.skill, skill);
}
```

Game.sol

```
struct Item {
  bytes32 item;
  uint256 amount;
}

mapping(bytes32 => bytes32 => Item) inventory;

function getItem(bytes32 entity, bytes32 item)
  returns (Item);

event InventoryChanged(
  bytes32 entity,
  Item item
);

function spend(bytes32 entity, Item item) {
  emit InventoryChanged(entity, item);
  inventory[entity][item.item] = item;
}
```

Client.ts

```
type Item {
  item: string;
  amount: number;
}

const inventory = Map<string, Item>();

function syncInitialInventory() {
  for(const entity of entities) {
    for(const id of items) {
      const item = contract.getItem(entity, id)
      inventory.set(entity+"/"+id, item);
    }
  }
}

function onInventoryChanged(
  entity: string,
  item: Item
) {
  inventory.set(entity+"/"+id, skill);
}
```


Game.sol
struct Item {
bytes32 item;
uint256 amount;
}
mapping(bytes32 => bytes32 => Item) inventory;
function getItem(bytes32 entity, bytes32 item)
returns (Item);
event InventoryChanged(
bytes32 entity,
Item item
);
function spend(bytes32 entity, Item item) {
emit InventoryChanged(entity, item);
inventory[entity][item.item] = item;
}

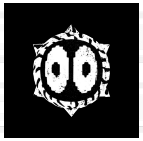
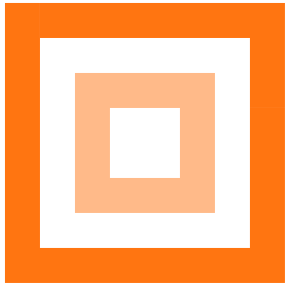

MUD



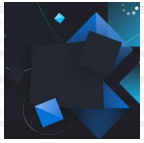




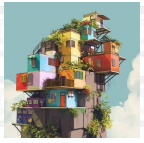
TX
MONSTER



MUD

EVE
FRONTIER

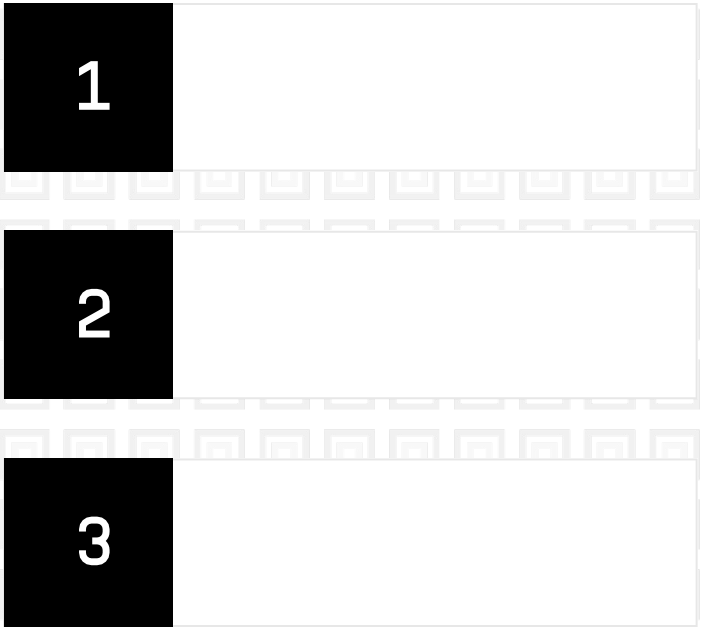



.NET



UD

CAFE
COSMOS



1
2
3

1 STATE SYNC

2

3

Game.sol

```
struct Item {
  bytes32 item;
  uint256 amount;
}

mapping(bytes32 => bytes32 => Item) inventory;

function getItem(bytes32 entity, bytes32 item)
  returns (Item);

event InventoryChanged(
  bytes32 entity,
  Item item
);

function spend(bytes32 entity, Item item) {
  emit InventoryChanged(entity, item);
  inventory[entity][item.item] = item;
}
```

Client.ts

```
type Item {
  item: string;
  amount: number;
}

const inventory = Map<string, Item>();

function syncInitialInventory() {
  for(const entity of entities) {
    for(const id of items) {
      const item = contract.getItem(entity, id)
      inventory.set(entity+"/"+id, item);
    }
  }
}

function onInventoryChanged(
  entity: string,
  item: Item
) {
  inventory.set(entity+"/"+id, skill);
}
```

Contract Position

Address Entity	uint256 x	uint256 y
0x00001	1	1
0x00002	0	3
0x00003	2	4

Client Position

Address Entity	uint256 x	uint256 y
0x00001	1	1
0x00002	0	3
0x00003	2	4

Contract Position

Address Entity	uint256 x	uint256 y
0x00001	1	1
0x00002	1	3
0x00003	2	4

Client Position

Address Entity	uint256 x	uint256 y
0x00001	1	1
0x00002	0	3
0x00003	2	4

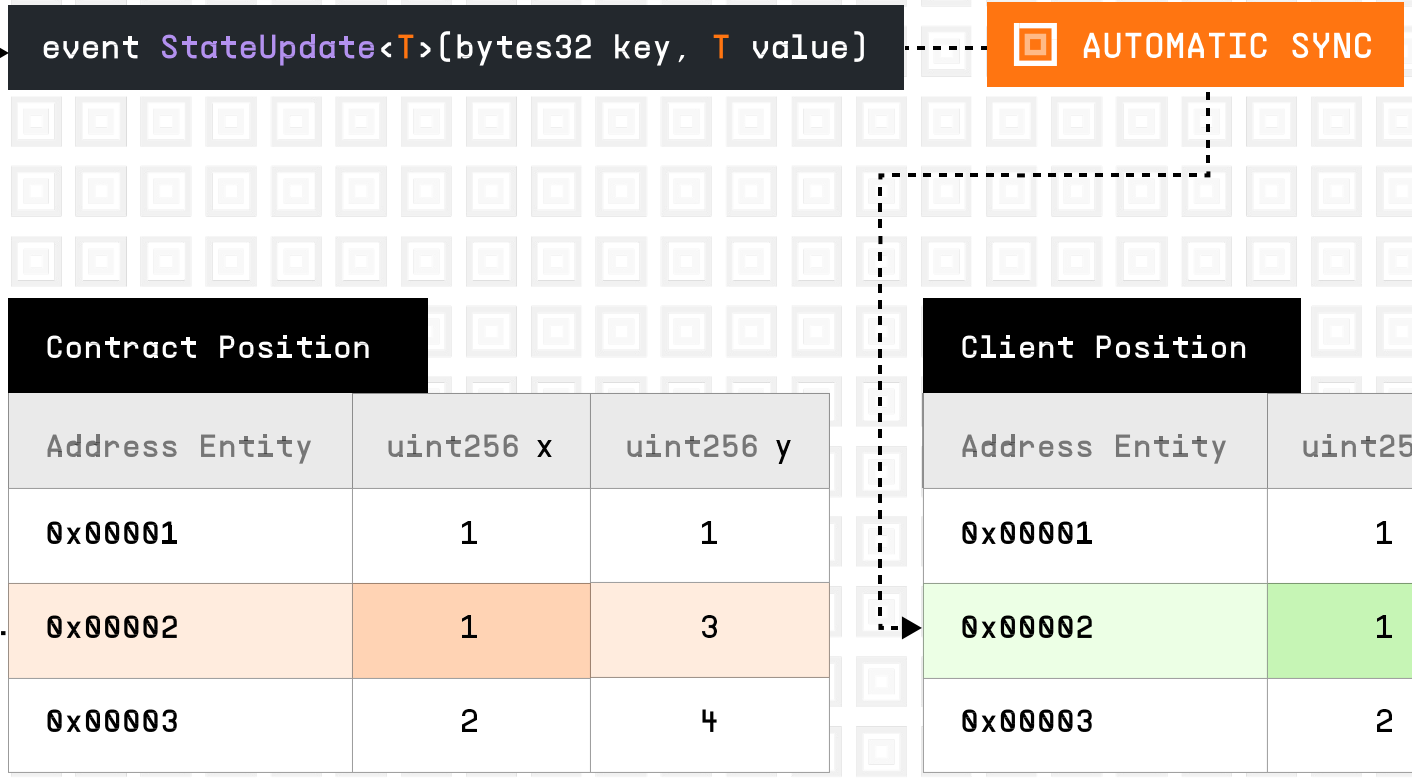
```
event StateUpdate<T>(bytes32 key, T value)
```

Contract Position

Address Entity	uint256 x	uint256 y
0x00001	1	1
0x00002	1	3
0x00003	2	4

Client Position

Address Entity	uint256 x	uint256 y
0x00001	1	1
0x00002	0	3
0x00003	2	4


event StateUpdate<T>(bytes32 key, T value)
AUTOMATIC SYNC
Contract Position
Address Entity
uint256 x
uint256 y
0x00001
1
1
0x00002
1
3
0x00003
2
4
Client Position
Address Entity
uint256 x
uint256 y
0x00001
1
1
0x00002
1
3
0x00003
2
4

```
event StateUpdate<T>(bytes32 key, T value)
```

AUTOMATIC SYNC

Contract Position

Address Entity	uint256 x	uint256 y
0x00001	1	1
0x00002	0	3
0x00003	2	4

Client Position

Address Entity	uint256 x	uint256 y
0x00001	1	1
0x00002	0	3
0x00003	2	4

```
event StateUpdate<T>(bytes32 key, T value)
```

AUTOMATIC SYNC

Game.sol

```
Position.set(entity, x, y);
```

Client.ts

```
const pos = Position.get(entity);
```

```
event StateUpdate<T>(bytes32 key, T value)
```

AUTOMATIC SYNC

Game.sol

```
Position.set(entity, x, y);
Health.set(entity, health);
```

Client.ts

```
const pos = Position.get(entity);
const health = Health.get(entity);
```

```
event StateUpdate<T>(bytes32 key, T value)
```

AUTOMATIC SYNC

Game.sol

```
Position.set(entity, x, y);
Health.set(entity, health);
Energy.set(entity, energy);
```

Client.ts

```
const pos = Position.get(entity);
const health = Health.get(entity);
const energy = Energy.get(entity);
```

```
event StateUpdate<T>(bytes32 key, T value)
```

AUTOMATIC SYNC

Game.sol

```
Position.set(entity, x, y);
Health.set(entity, health);
Energy.set(entity, energy);
Skill.set(entity, skill, level);
```

Client.ts

```
const pos = Position.get(entity);
const health = Health.get(entity);
const energy = Energy.get(entity);
const level = Skill.get(entity, skill);
```

```
event StateUpdate<T>(bytes32 key, T value)
```

AUTOMATIC SYNC

Game.sol

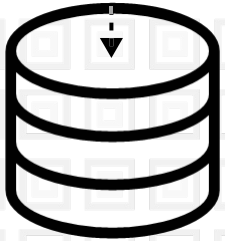
```
Position.set(entity, x, y);
Health.set(entity, health);
Energy.set(entity, energy);
Skill.set(entity, skill, level);
Inventory.set(entity, item, amount);
```

Client.ts

```
const pos = Position.get(entity);
const health = Health.get(entity);
const energy = Energy.get(entity);
const level = Skill.get(entity, skill);
const num = Inventory.get(entity, item);
```

```
event StateUpdate<T>(bytes32 key, T value)
```

AUTOMATIC SYNC

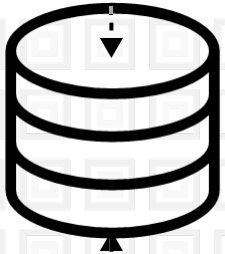


Contract Position

Address Entity	uint256 x	uint256 y
0x00001	1	1
0x00002	0	3
0x00003	2	4

```
event StateUpdate<T>(bytes32 key, T value)
```

AUTOMATIC SYNC



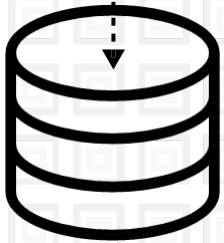
Contract Position

Address Entity	uint256 x	uint256 y
0x00001	1	1
0x00002	0	3
0x00003	2	4

```
SELECT p.entity, p.x, p.y, h.health
FROM position p, health h
WHERE p.entity = h.entity
  AND p.x > 10
  AND h.health > 0;
```

```
event StateUpdate(bytes32 table, bytes32 key, bytes value)
```

AUTOMATIC SYNC



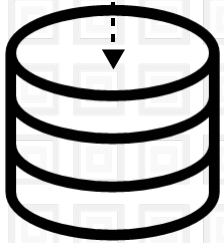
Contract Position

Address Entity	uint256 x	uint256 y
0x00001	1	1
0x00002	0	3
0x00003	2	4

```
event StateUpdate(bytes32 table, bytes32 key, bytes value)
```

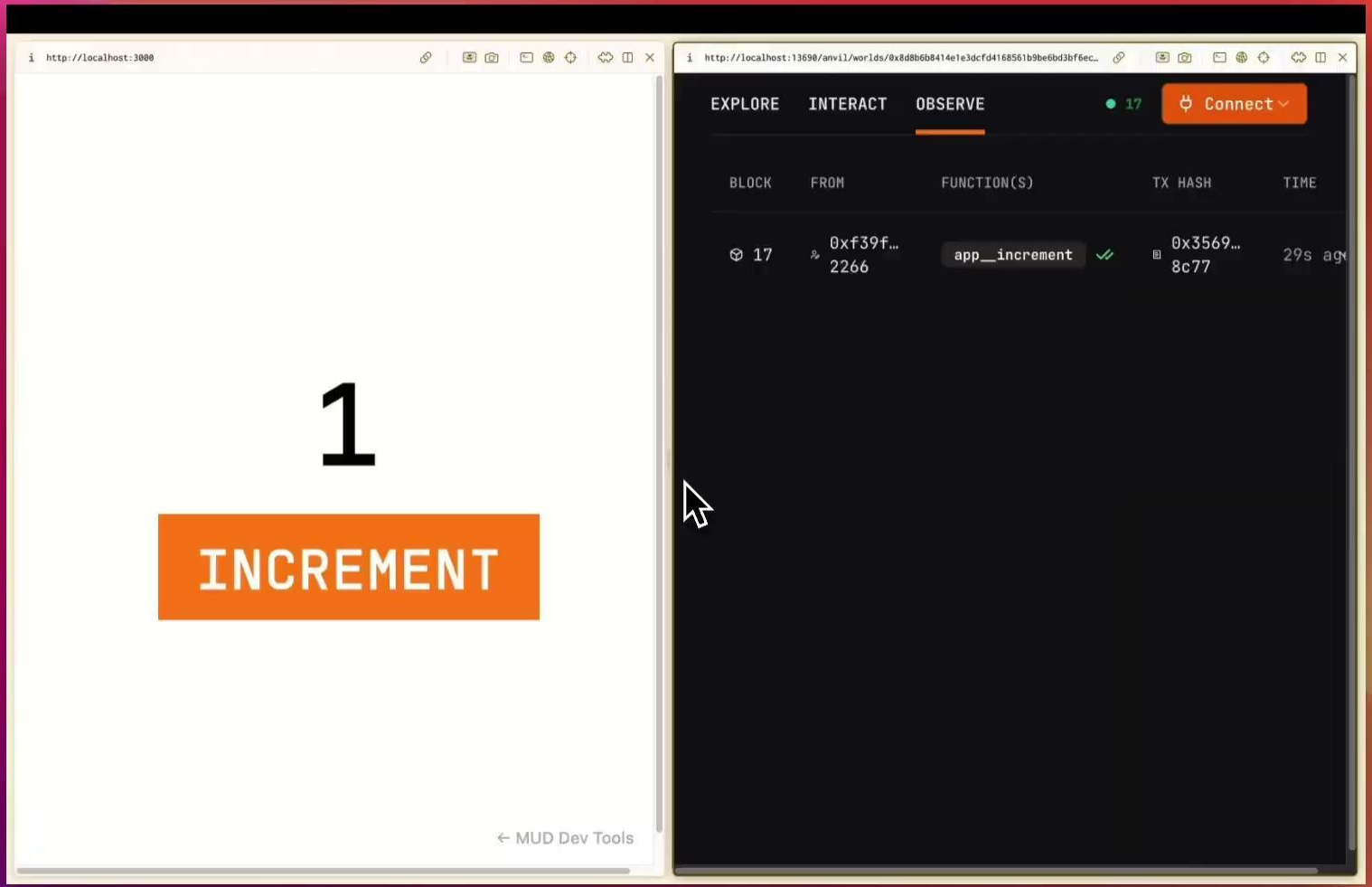
AUTOMATIC SYNC

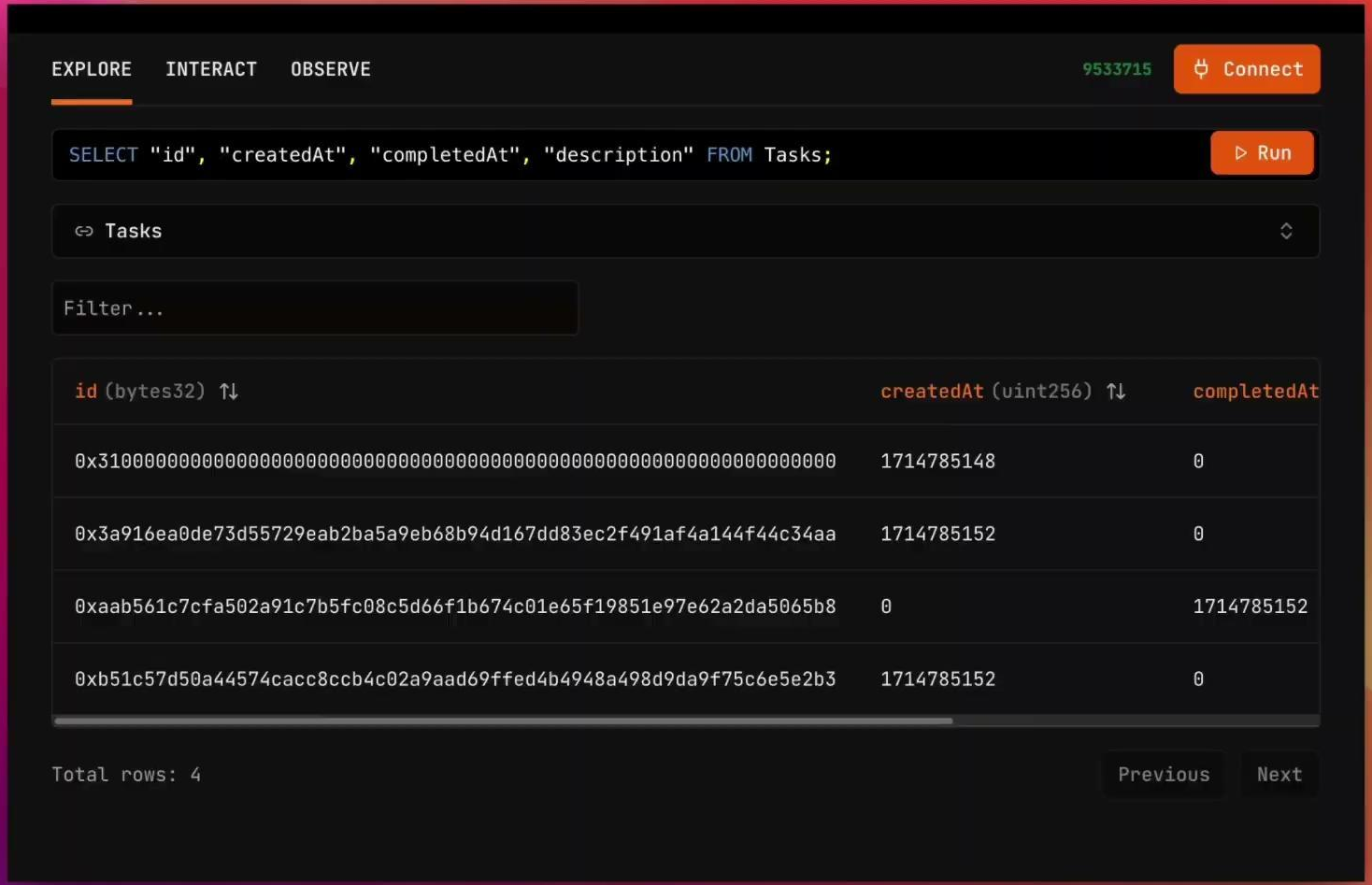
DECODER



Tables

bytes32 table	bytes32 schema
Position	uint256, uint256
Health	uint256
Inventory	bytes32, uint256


http://localhost:3000
1
INCREMENT
← MUD Dev Tools
http://localhost:13690/anvil/worlds/0x8d8b6b8414e1e3dcfd4168561b9be6bd3bf6ec...
EXPLORE
INTERACT
OBSERVE
17
Connect
BLOCK
FROM
FUNCTION(S)
TX HASH
TIME
17
0xf39f... 2266
app__increment
0x3569... 8c77
29s ag


EXPLORE
INTERACT
OBSERVE
9533715
Connect
SELECT "id", "createdAt", "completedAt", "description" FROM Tasks;
Run
Tasks
Filter...
id (bytes32)
createdAt (uint256)
completedAt
0x3100000000000000000000000000000000000000000000000000000000000000
1714785148
0
0x3a916ea0de73d55729eab2ba5a9eb68b94d167dd83ec2f491af4a144f44c34aa
1714785152
0
0xaab561c7cfa502a91c7b5fc08c5d66f1b674c01e65f19851e97e62a2da5065b8
0
1714785152
0xb51c57d50a44574cacc8ccb4c02a9aad69ffed4b4948a498d9da9f75c6e5e2b3
1714785152
0
Total rows: 4
Previous
Next

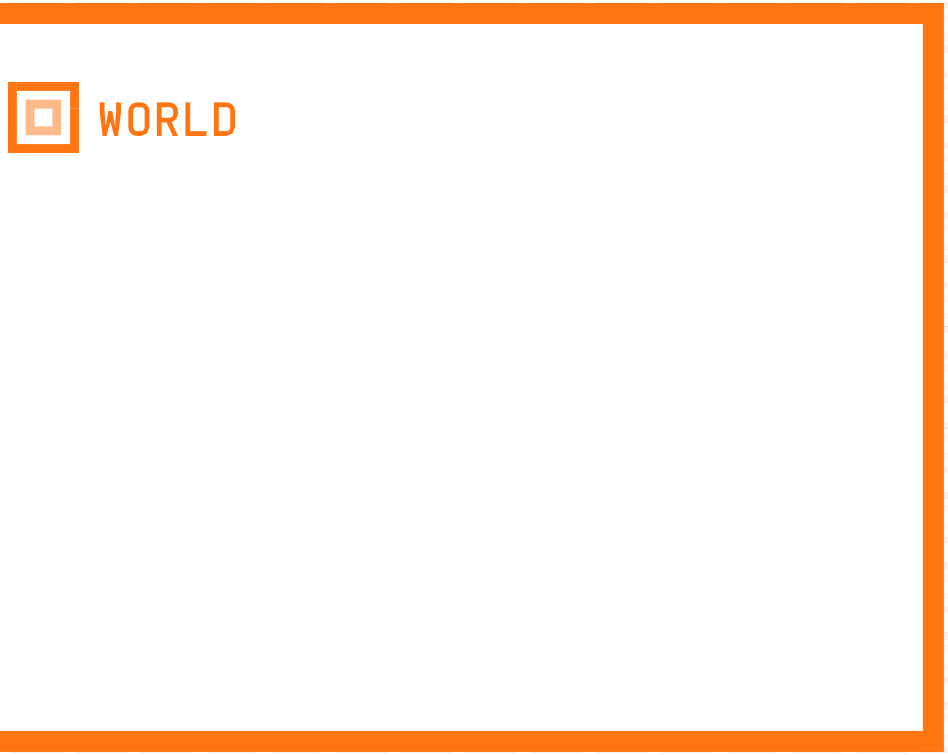

STORE

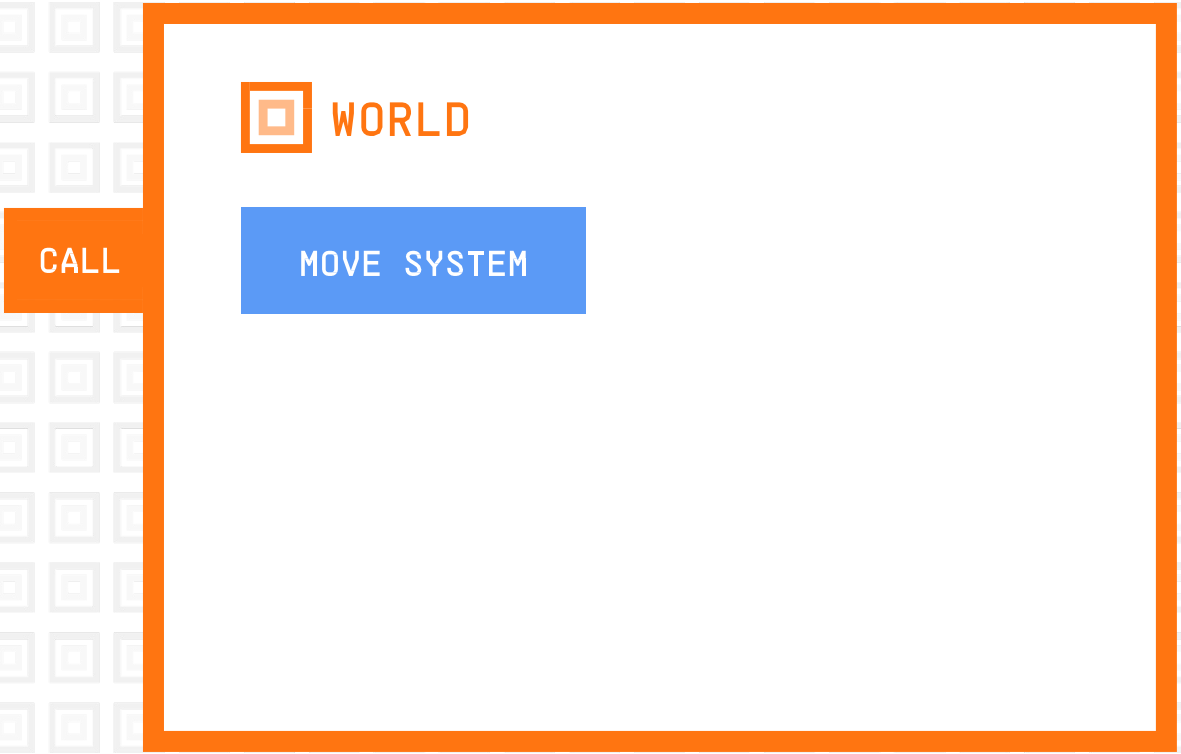
1 STATE SYNC

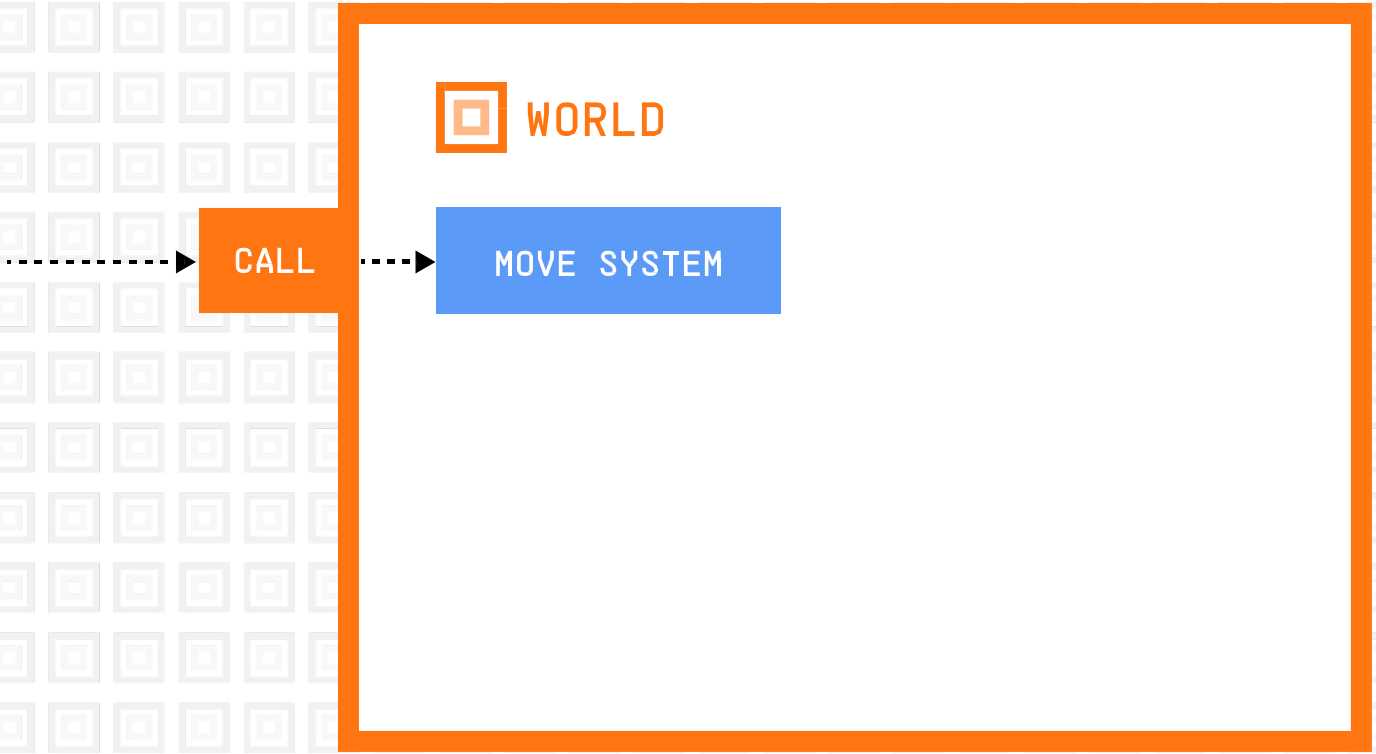
2 SCALE LOGIC

3

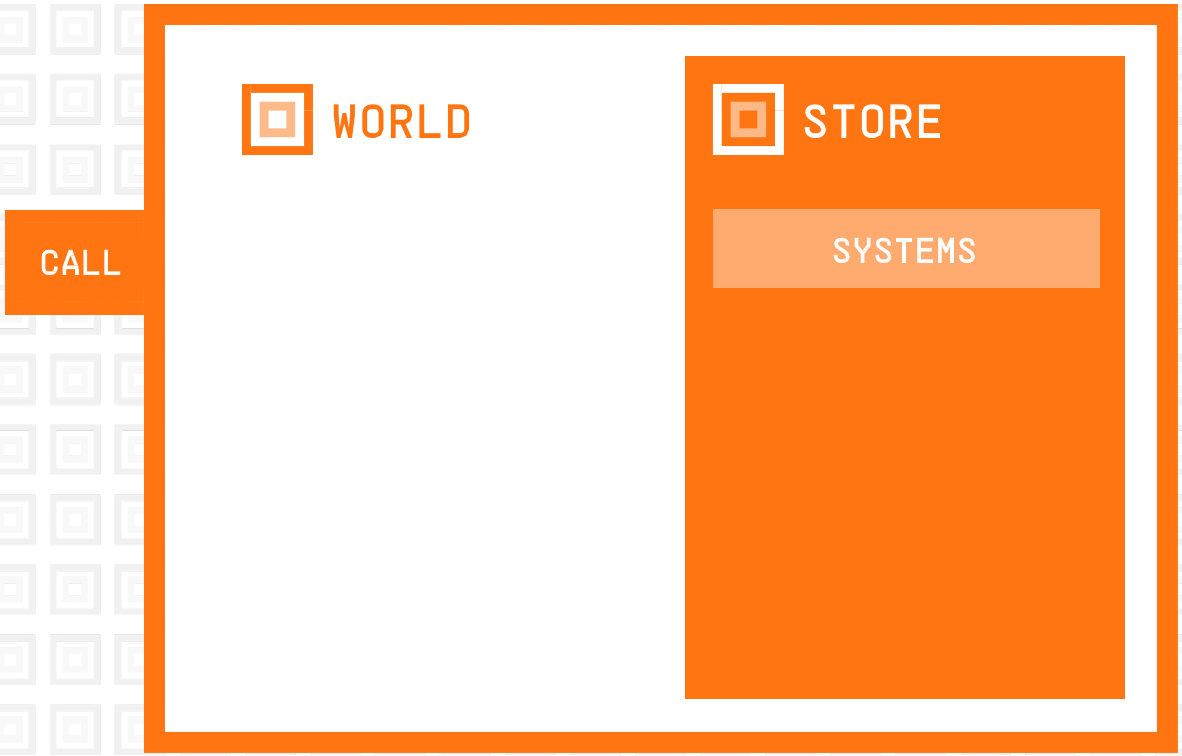
WARNING:
CONTRACT CODE SIZE EXCEEDS 24576 BYTES
(A LIMIT INTRODUCED IN SPURIOUS DRAGON)

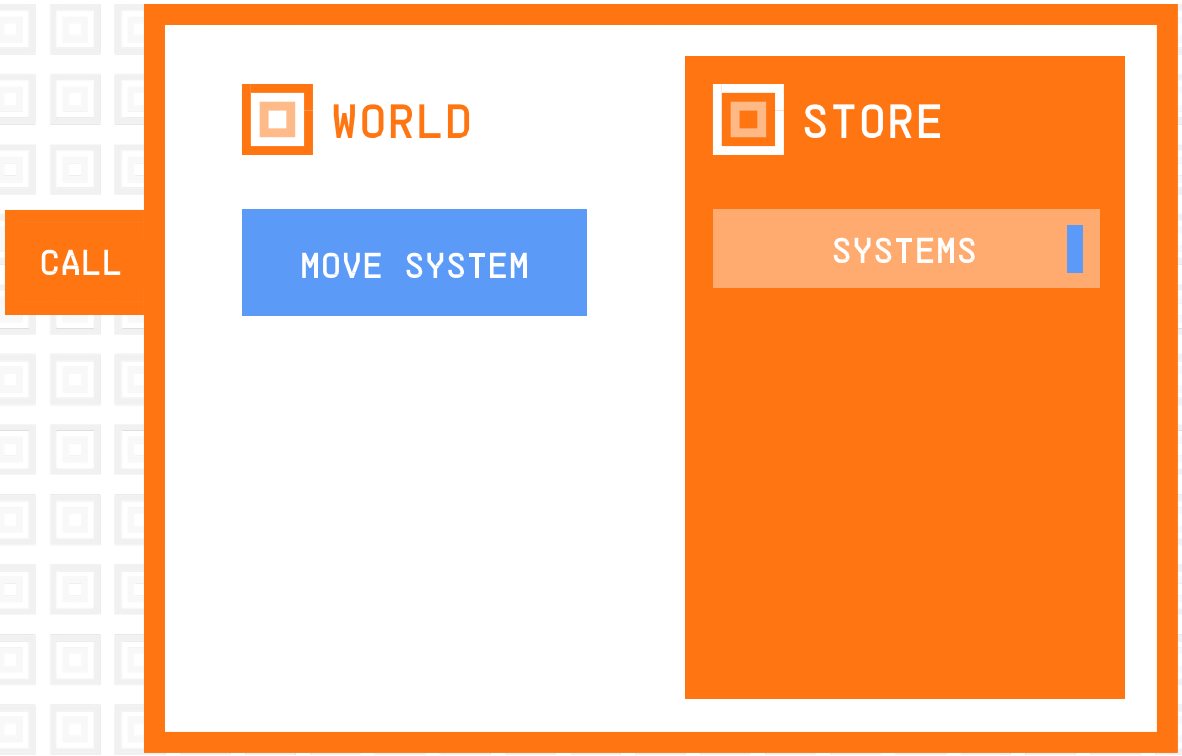

WORLD
CALL

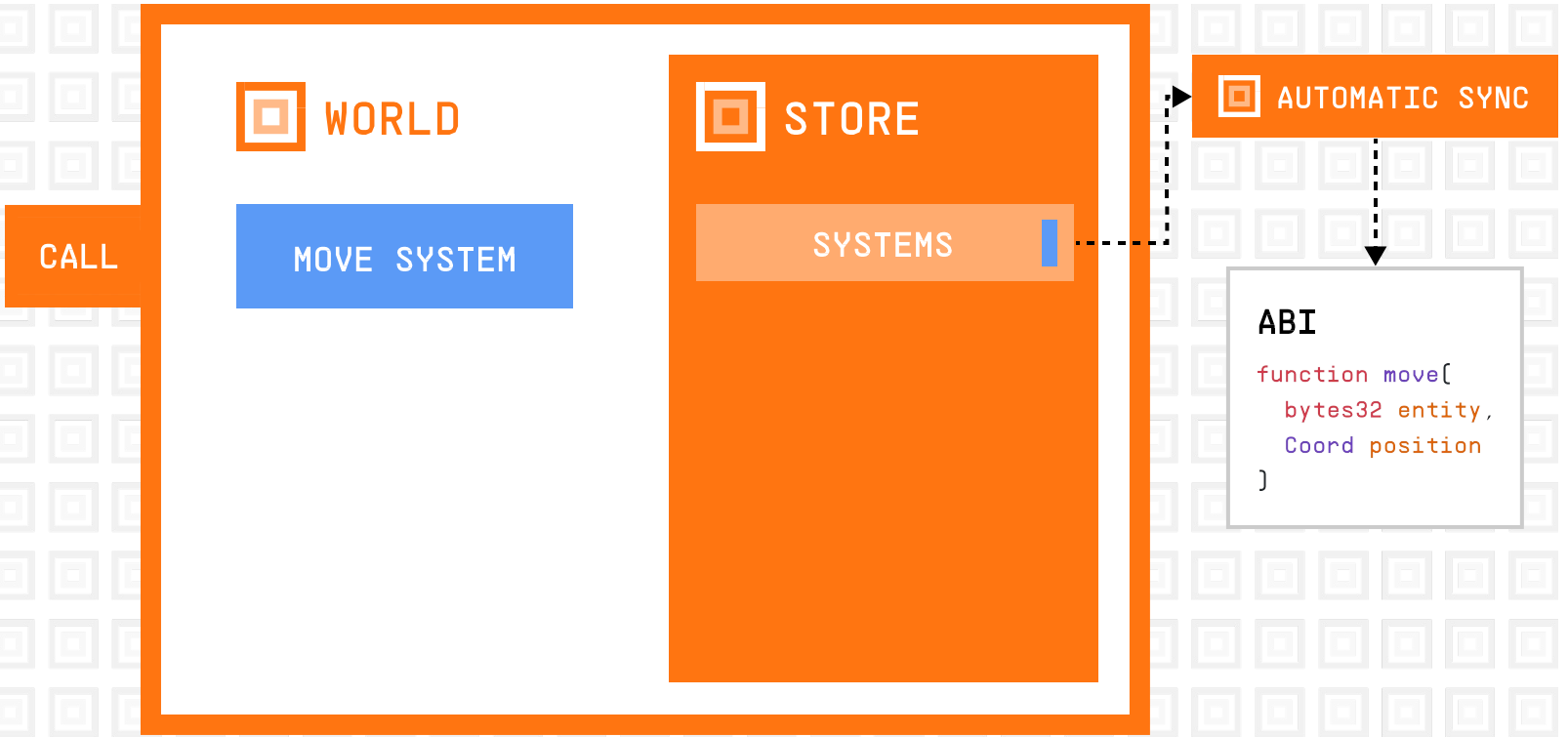

WORLD
CALL
MOVE SYSTEM

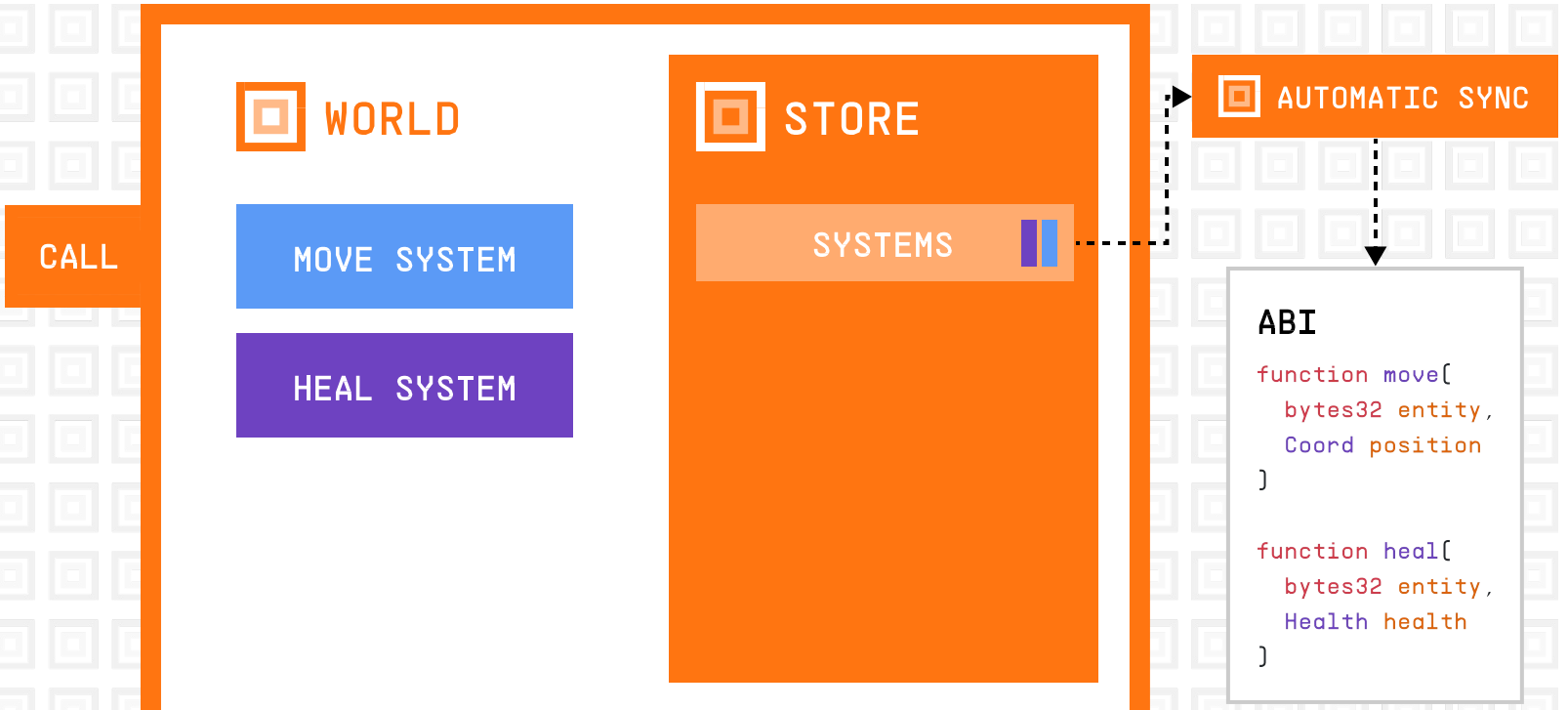

WORLD
CALL
MOVE SYSTEM

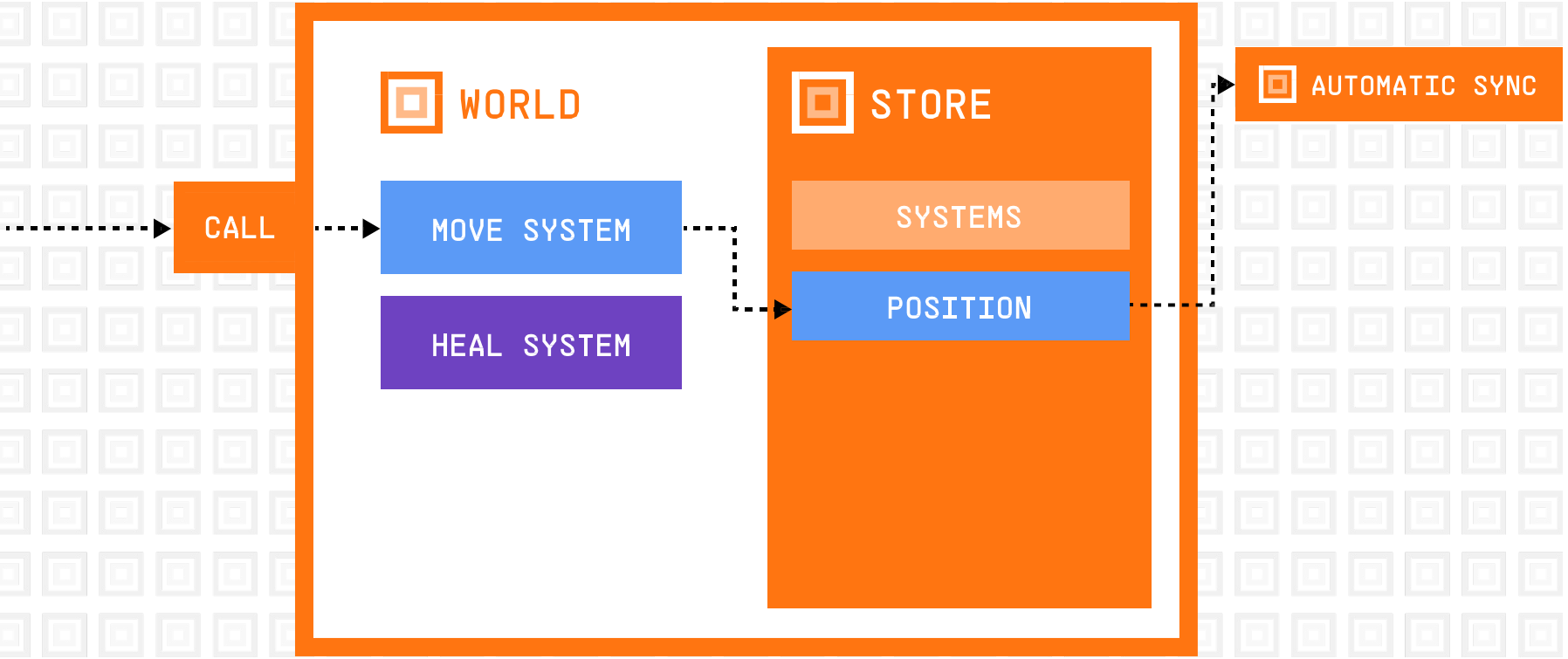

WORLD
STORE
CALL

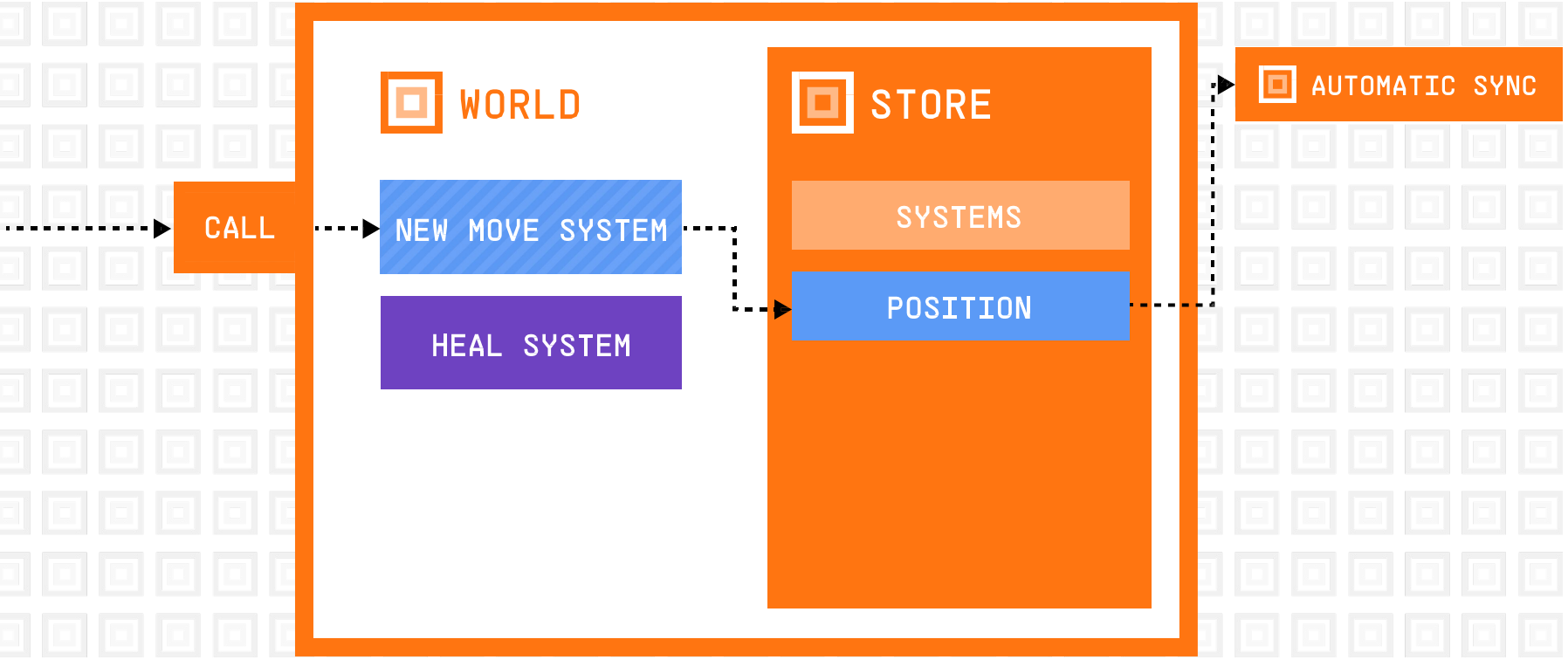

WORLD
STORE
SYSTEMS
CALL

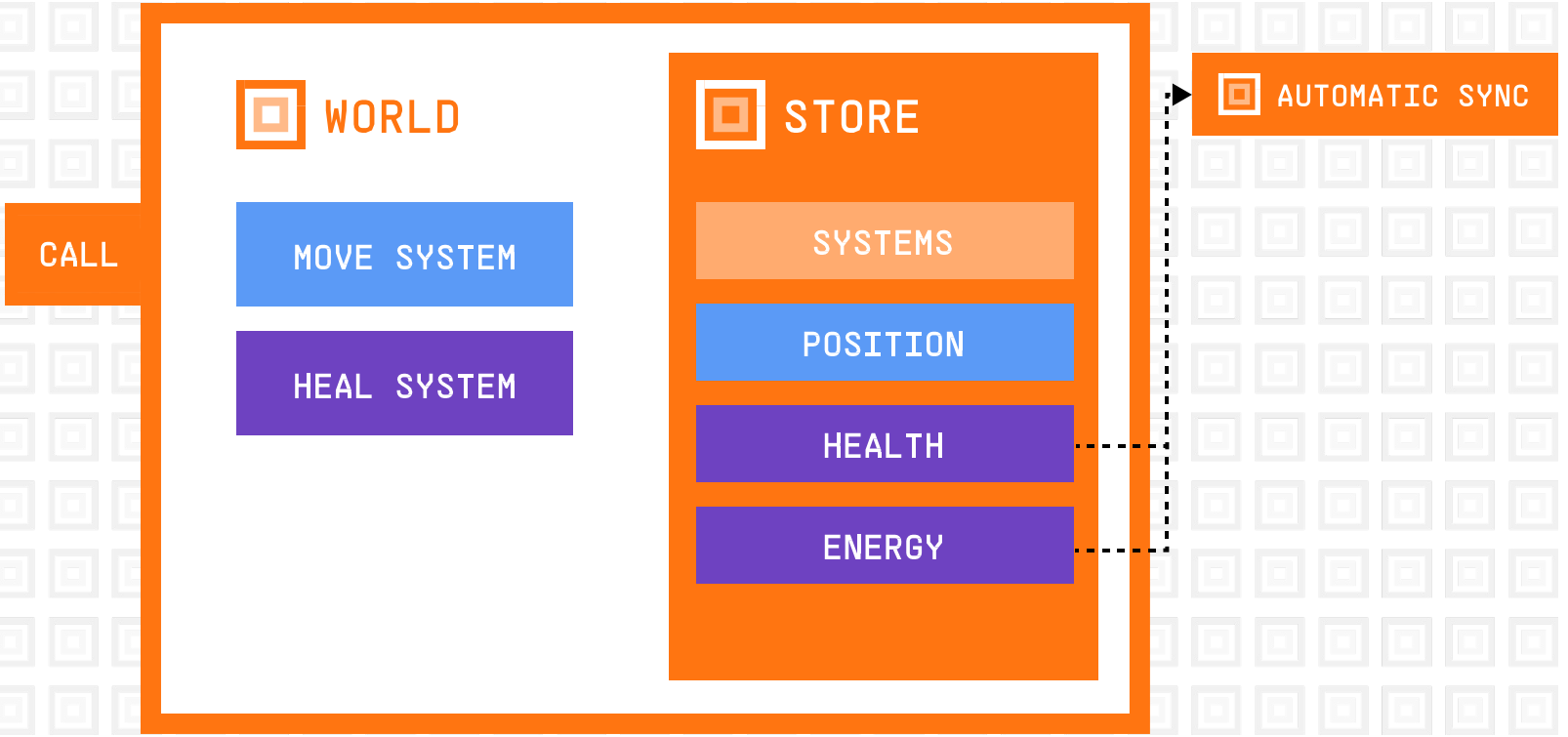

WORLD
STORE
CALL
MOVE SYSTEM
SYSTEMS

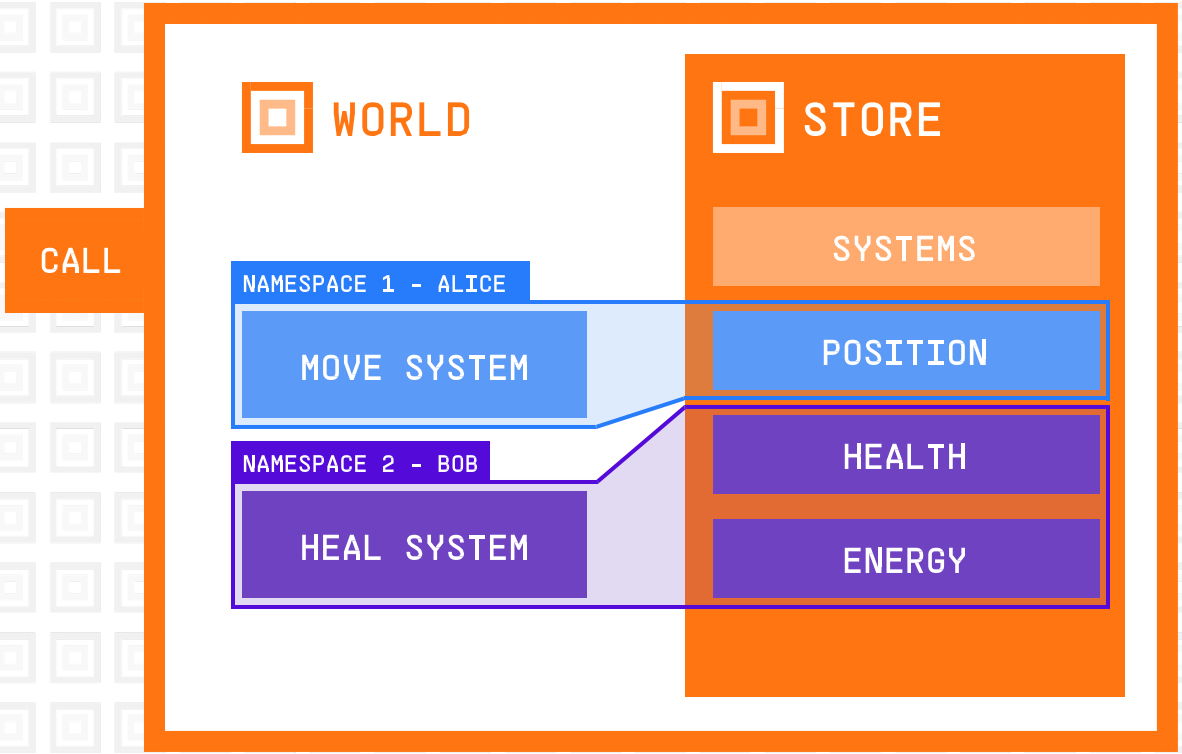

WORLD
STORE
AUTOMATIC SYNC
CALL
MOVE SYSTEM
SYSTEMS
ABI
function move(
bytes32 entity,
Coord position
)


CALL
WORLD
MOVE SYSTEM
HEAL SYSTEM
STORE
SYSTEMS
AUTOMATIC SYNC
ABI
function move(
bytes32 entity,
Coord position
)
function heal(
bytes32 entity,
Health health
)

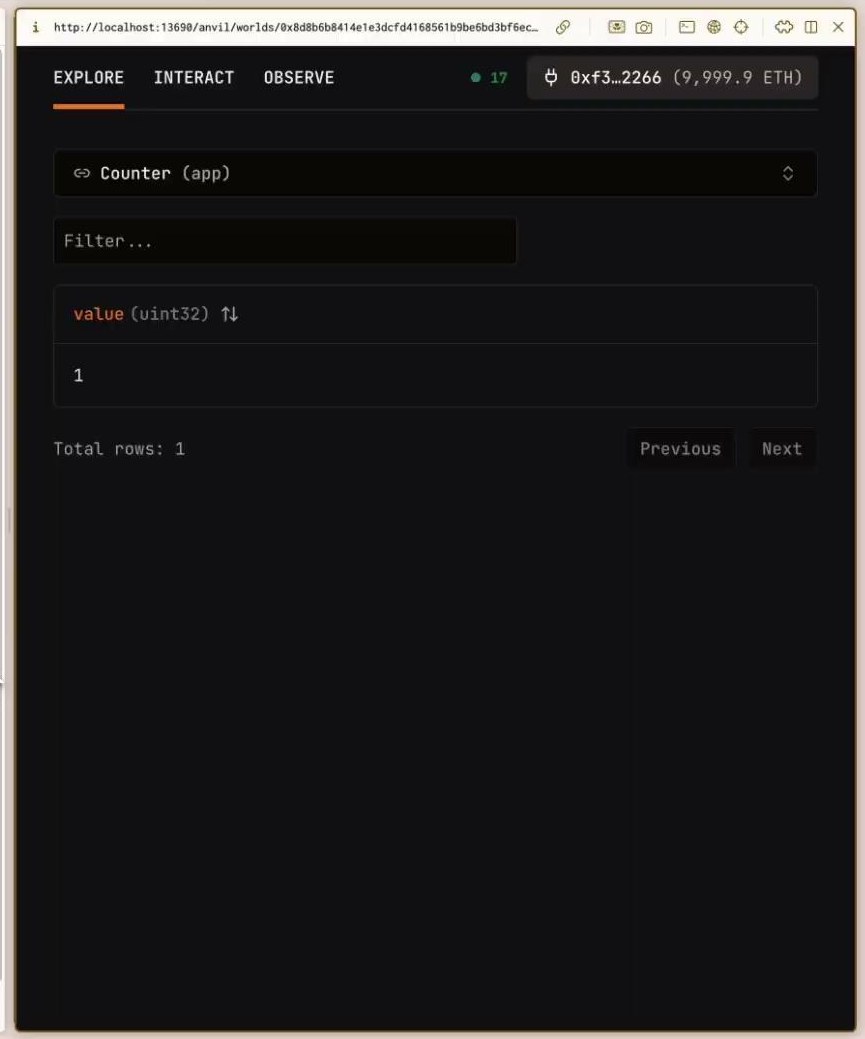

WORLD
CALL
MOVE SYSTEM
HEAL SYSTEM
STORE
SYSTEMS
POSITION
AUTOMATIC SYNC


WORLD
CALL
NEW MOVE SYSTEM
HEAL SYSTEM
STORE
SYSTEMS
POSITION
AUTOMATIC SYNC


WORLD
CALL
MOVE SYSTEM
HEAL SYSTEM
STORE
SYSTEMS
POSITION
HEALTH
ENERGY
AUTOMATIC SYNC


WORLD
STORE
CALL
SYSTEMS
NAMESPACE 1 - ALICE
MOVE SYSTEM
POSITION
NAMESPACE 2 - BOB
HEAL SYSTEM
HEALTH
ENERGY


http://localhost:3000
1
INCREMENT
← MUD Dev Tools


http://localhost:13690/anvil/worlds/0x8d8b6b8414e1e3dcfd4168561b9be6bd3bf6ec...
EXPLORE
INTERACT
OBSERVE
17
0xf3…2266 (9,999.9 ETH)
Counter (app)
Filter...
value (uint32)
1
Total rows: 1
Previous
Next

1 STATE SYNC

2 SCALE LOGIC

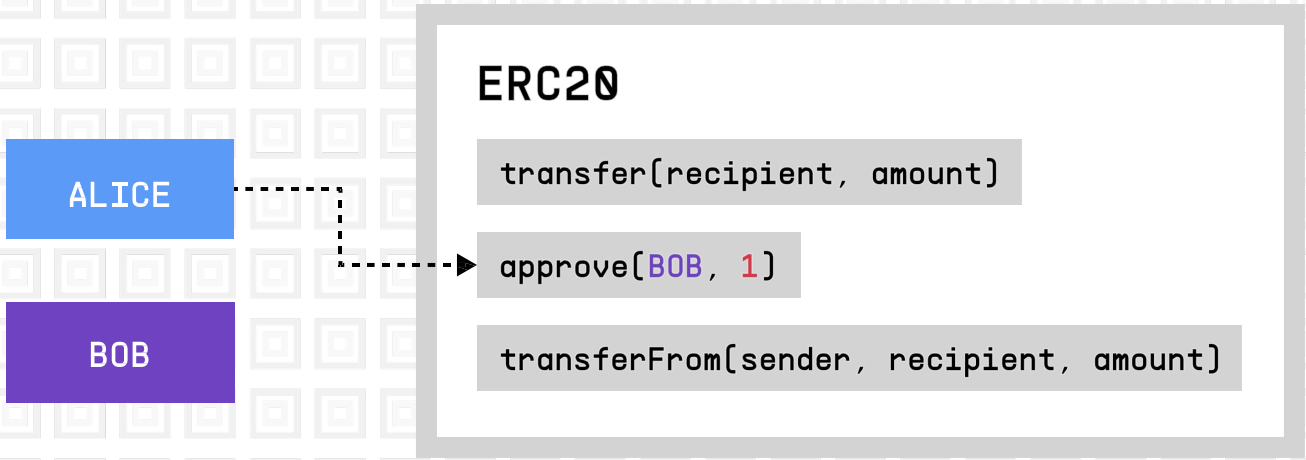
3 HIGHER LEVEL

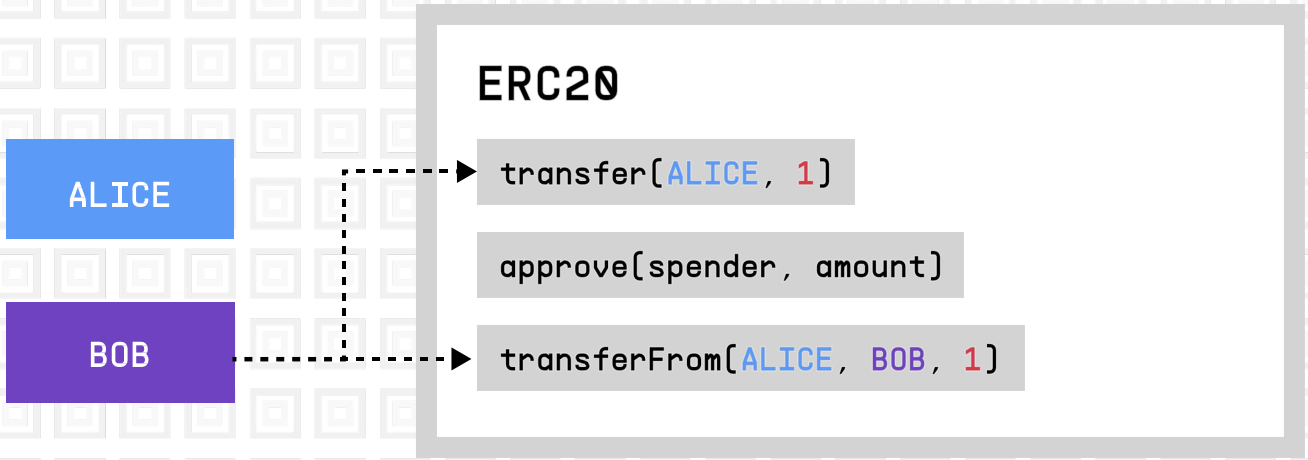
ERC20

```
transfer(recipient, amount)
```

```
approve(spender, amount)
```

```
transferFrom(sender, recipient, amount)
```


ALICE
BOB
ERC20
transfer(recipient, amount)
approve(BOB, 1)
transferFrom(sender, recipient, amount)


ERC20
ALICE
BOB
transfer(ALICE, 1)
approve(spender, amount)
transferFrom(ALICE, BOB, 1)

Game.sol

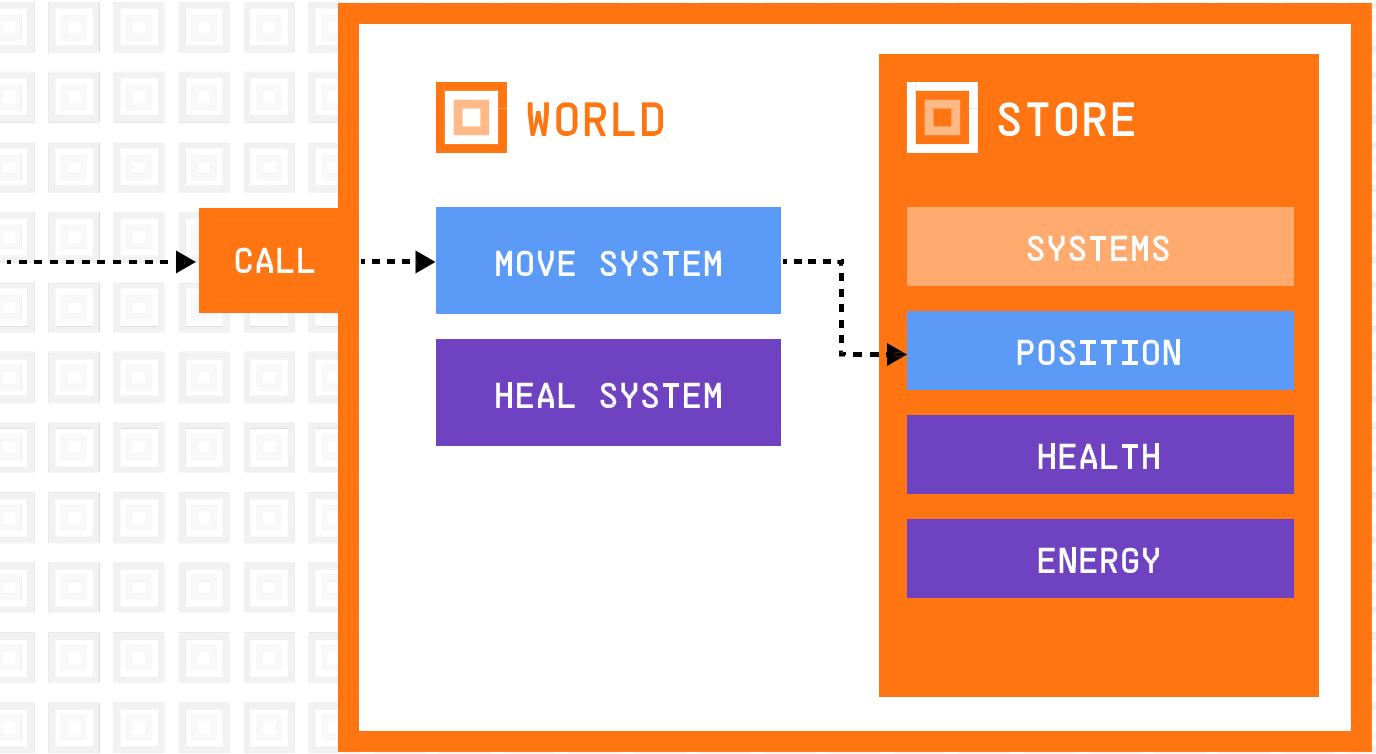
```
approveMove(mover);

moveFrom(owner, entity, coord);
```

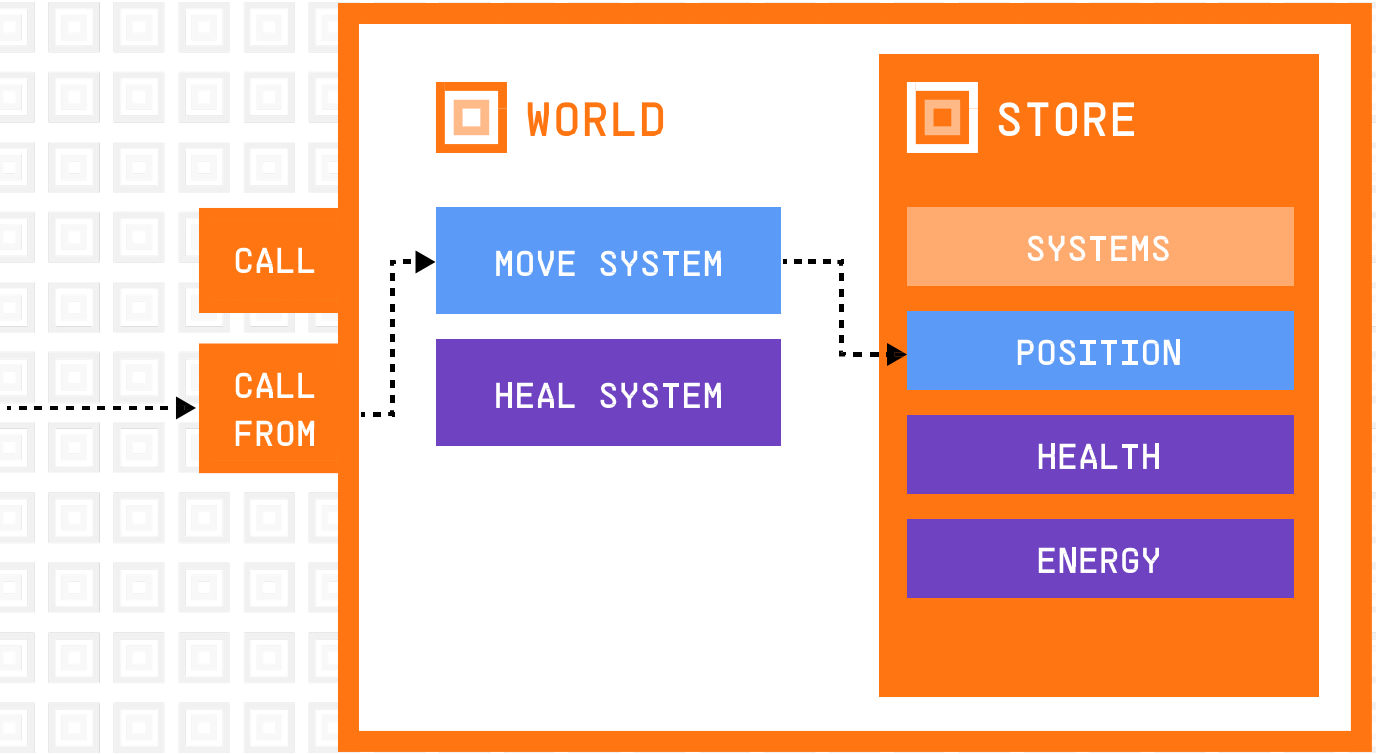
Game.sol

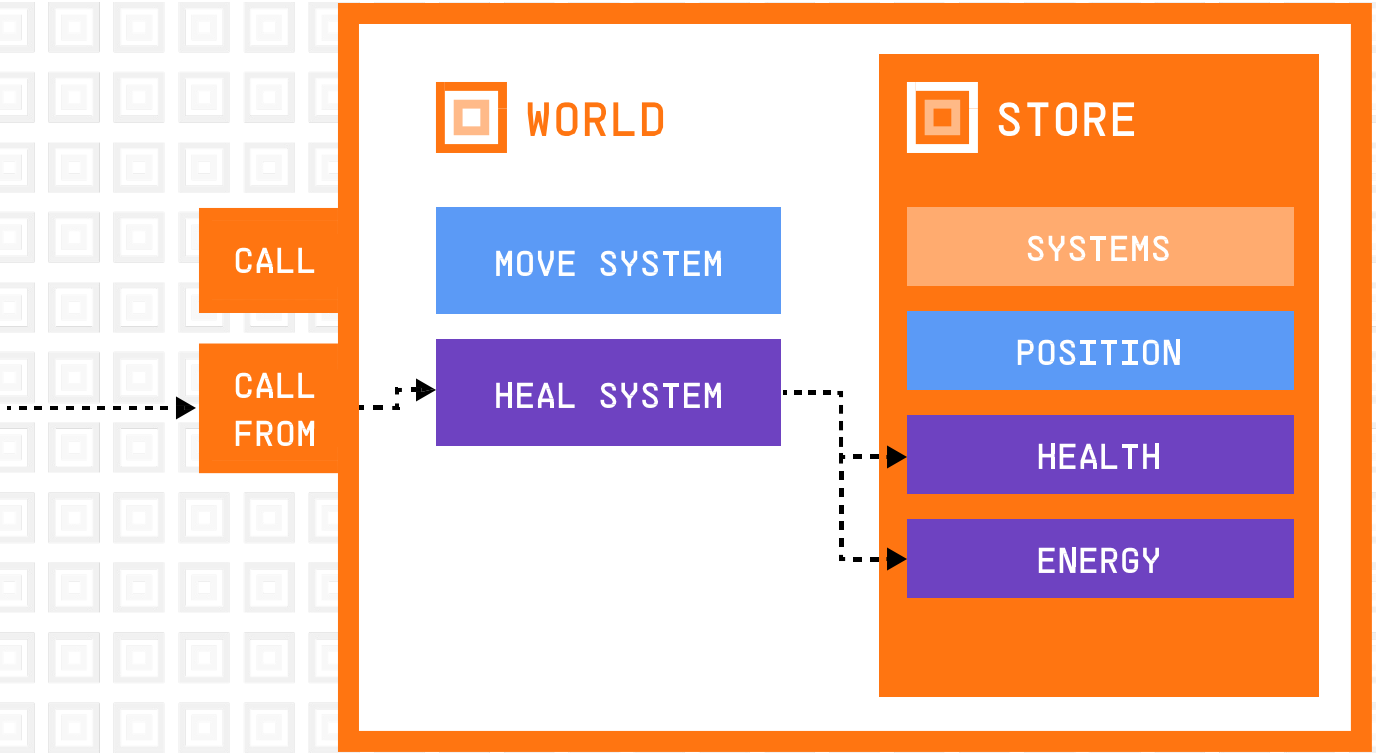
```
approveMove(mover);

moveFrom(owner, entity, coord);

approveAttack(attacker);

attackFrom(owner, entity, target);
```

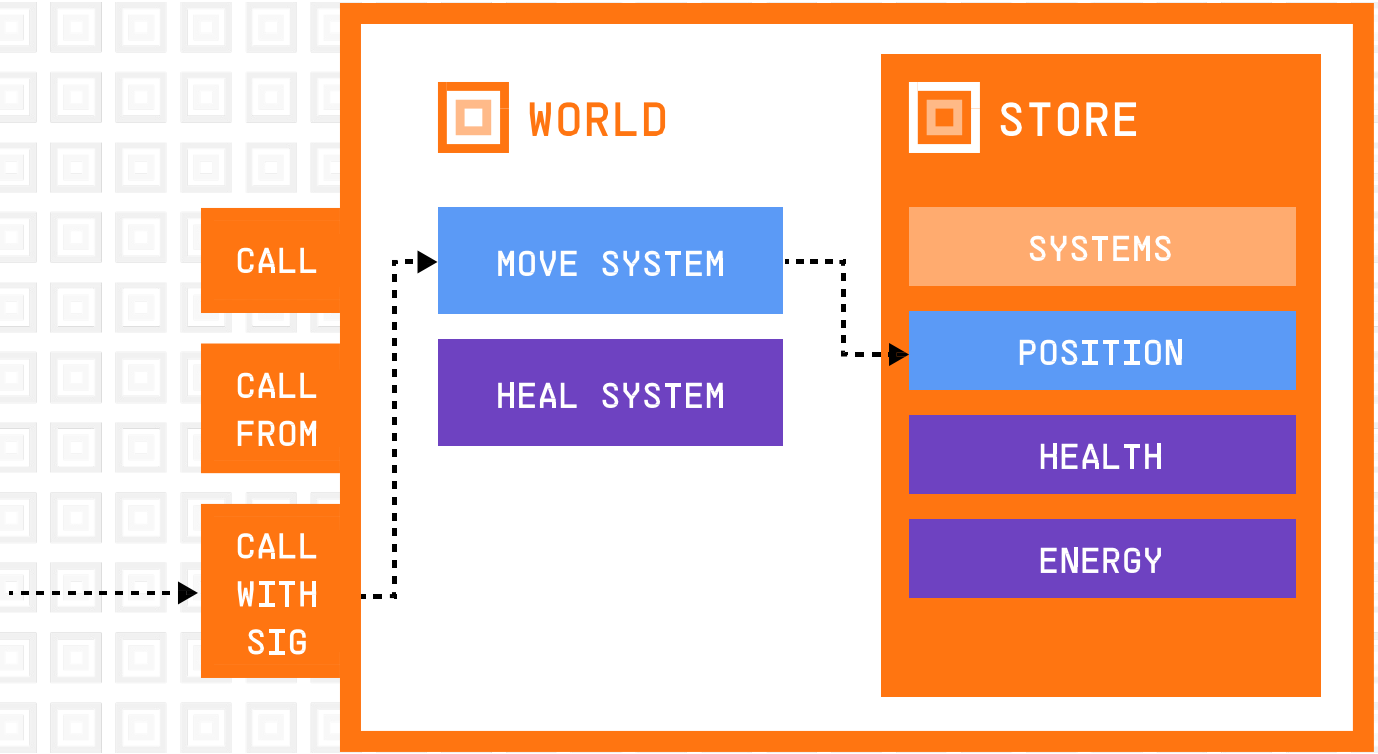
Game.sol

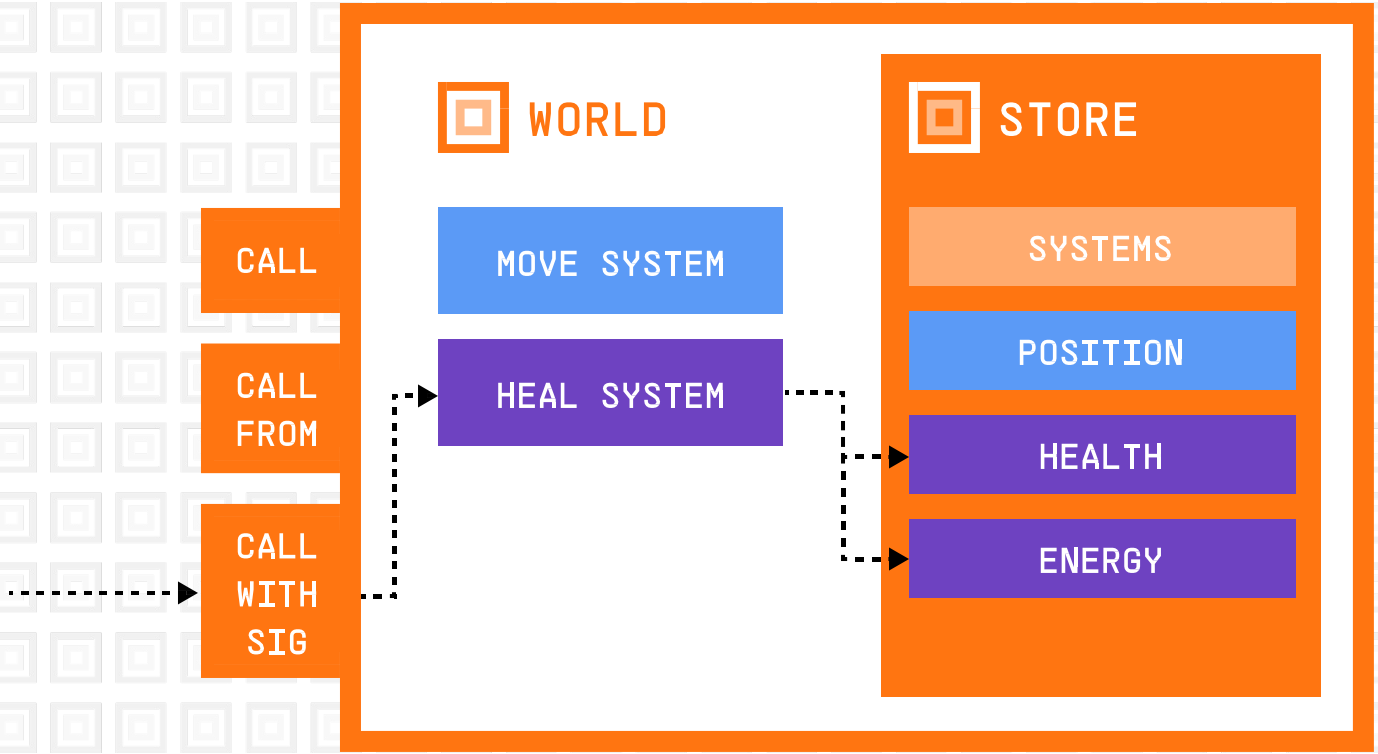
```
approveMove(mover);

moveFrom(owner, entity, coord);

approveAttack(attacker);

attackFrom(owner, entity, target);

moveWithSignature(sig, entity, coord);

attackWithSig(sig, entity, coord);

...
```

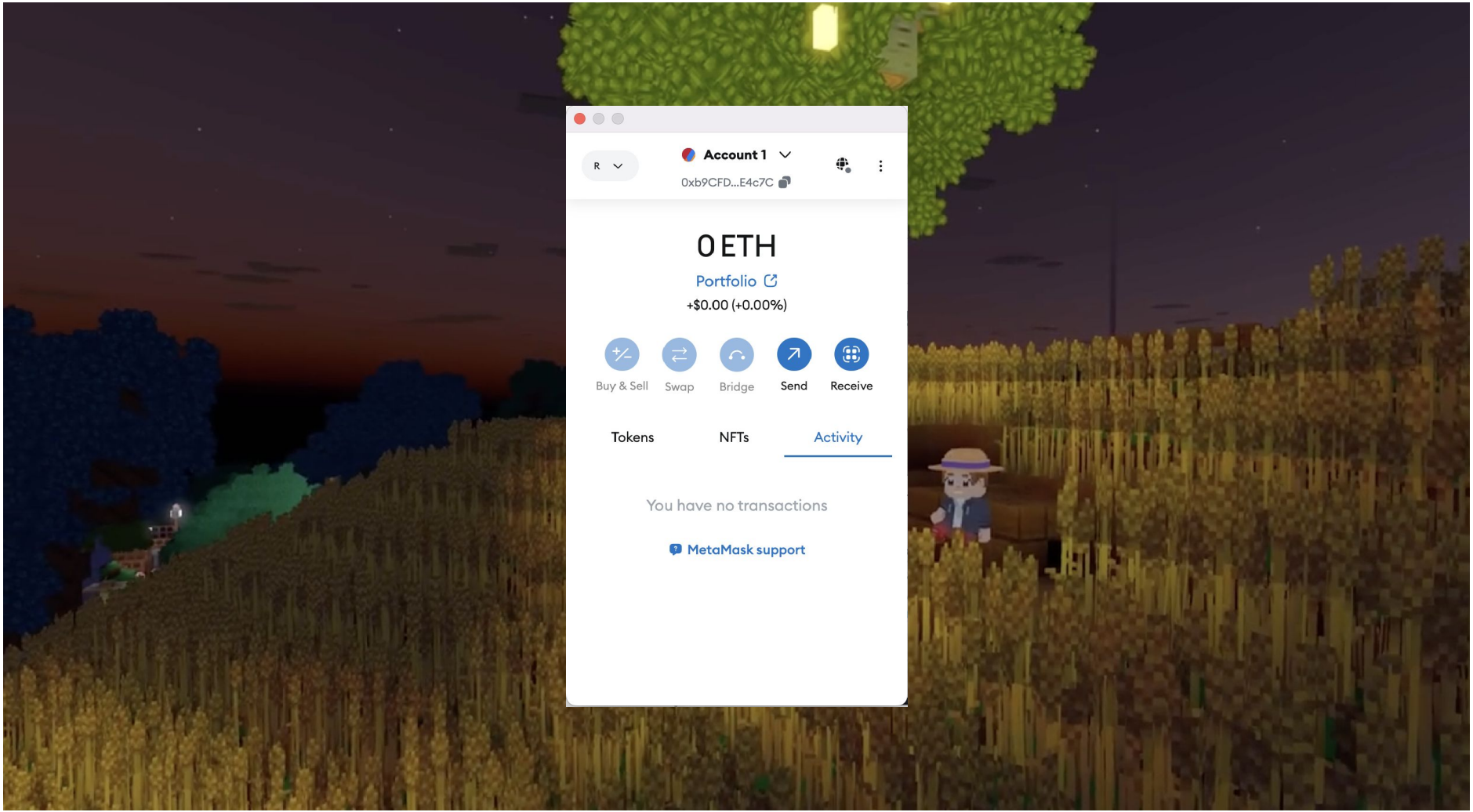

WORLD
STORE
CALL
MOVE SYSTEM
HEAL SYSTEM
SYSTEMS
POSITION
HEALTH
ENERGY

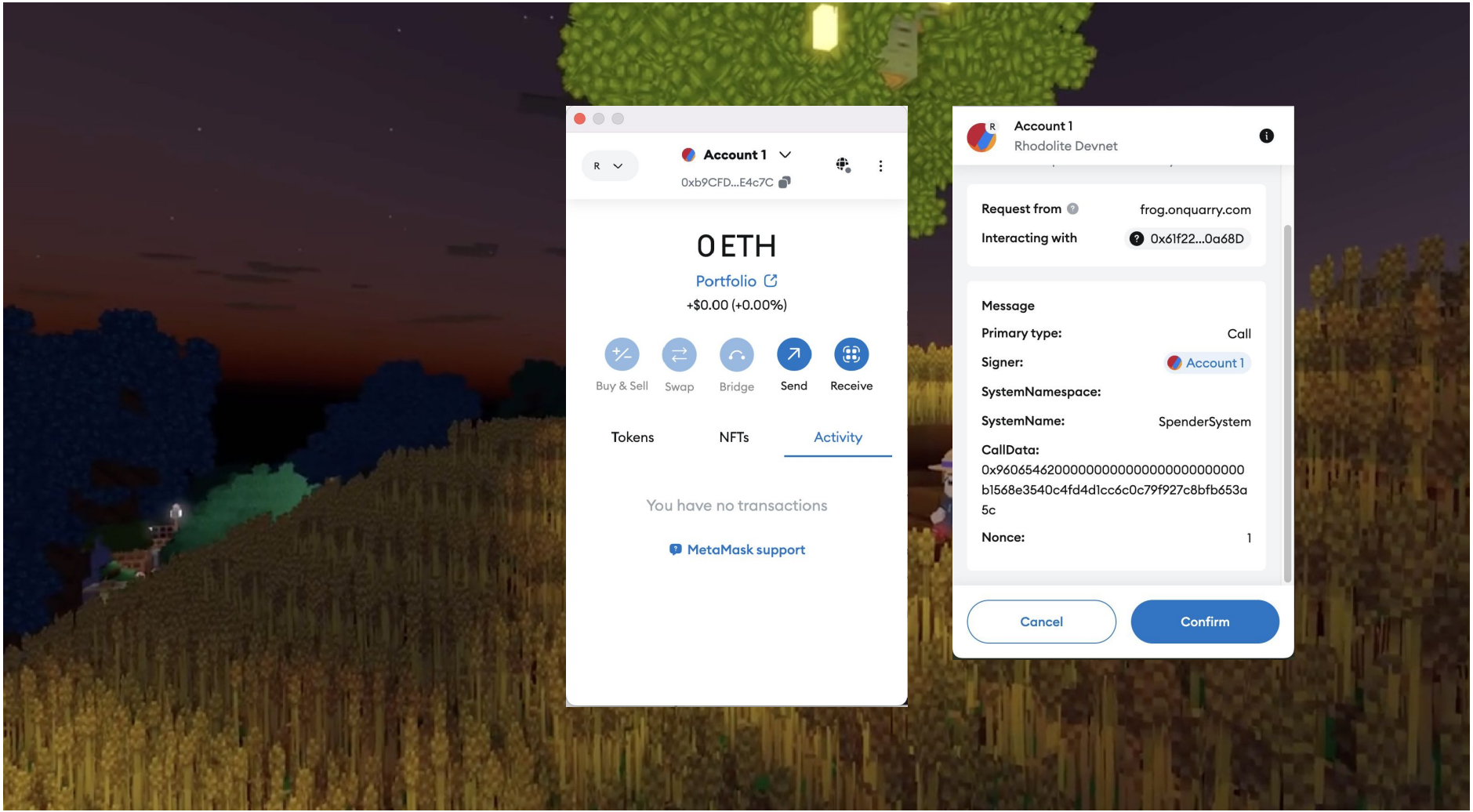

WORLD
STORE
CALL
CALL FROM
MOVE SYSTEM
HEAL SYSTEM
SYSTEMS
POSITION
HEALTH
ENERGY

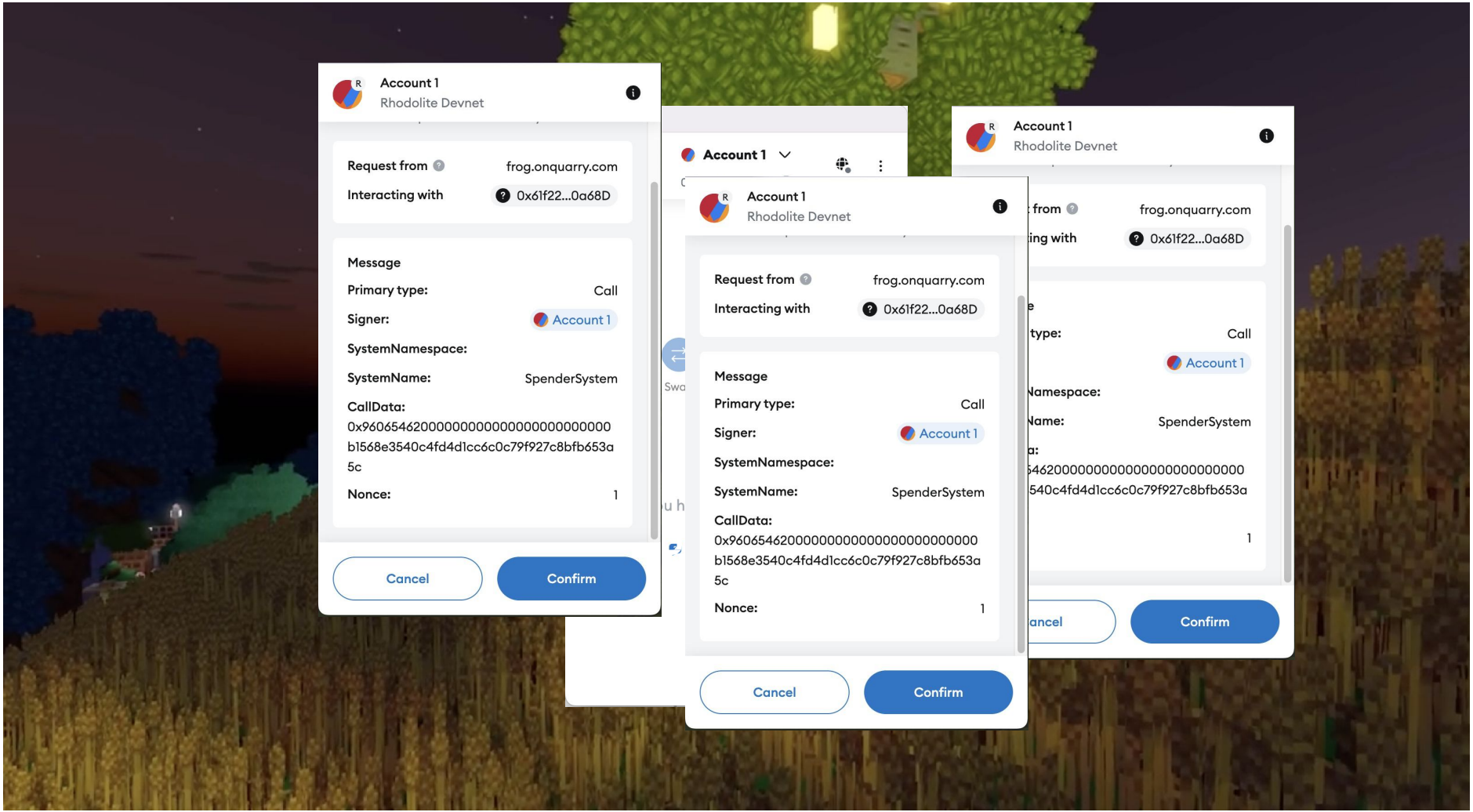

WORLD
STORE
CALL
CALL FROM
MOVE SYSTEM
HEAL SYSTEM
SYSTEMS
POSITION
HEALTH
ENERGY

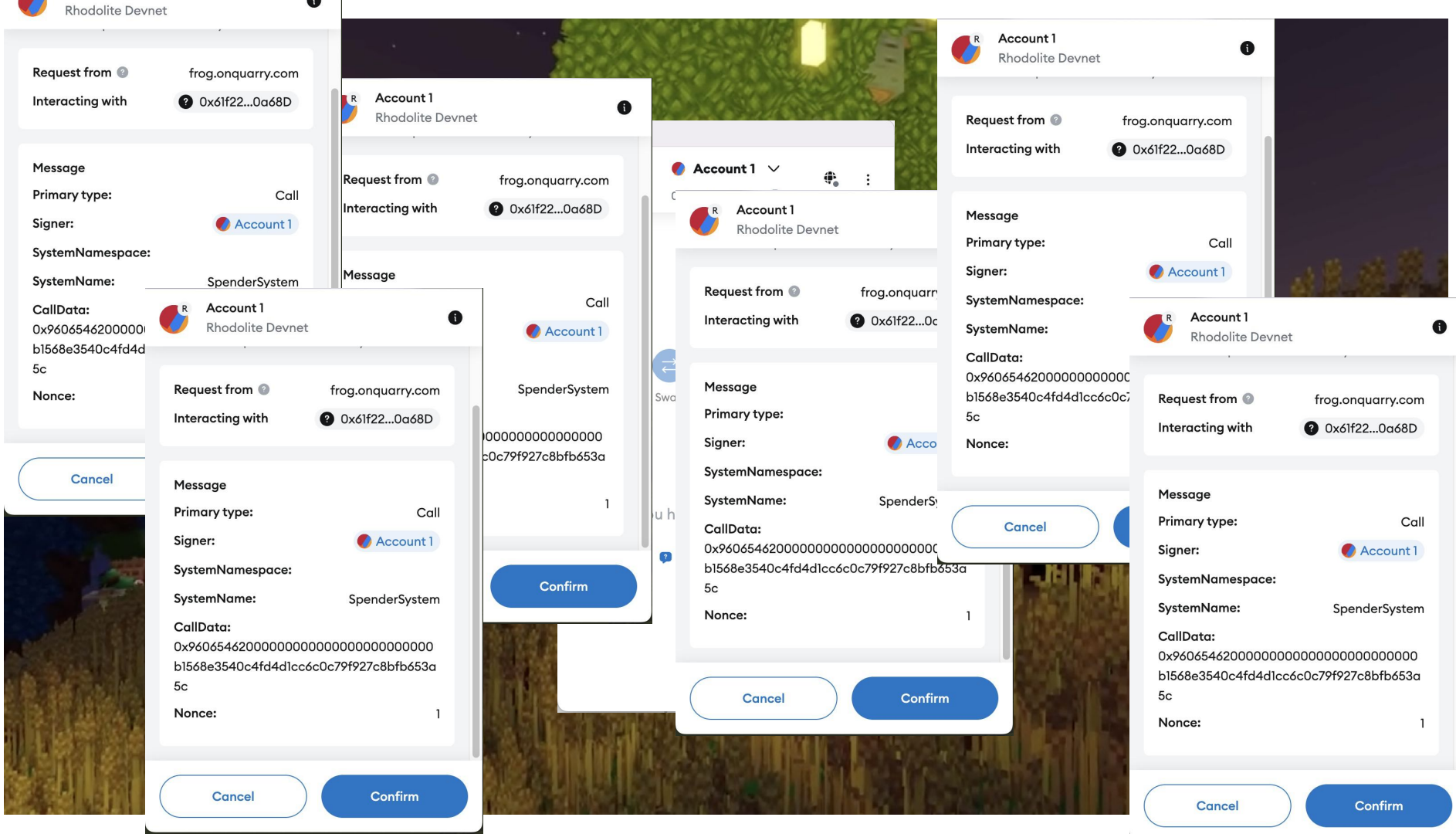

WORLD
STORE
CALL
CALL FROM
CALL WITH SIG
MOVE SYSTEM
HEAL SYSTEM
SYSTEMS
POSITION
HEALTH
ENERGY

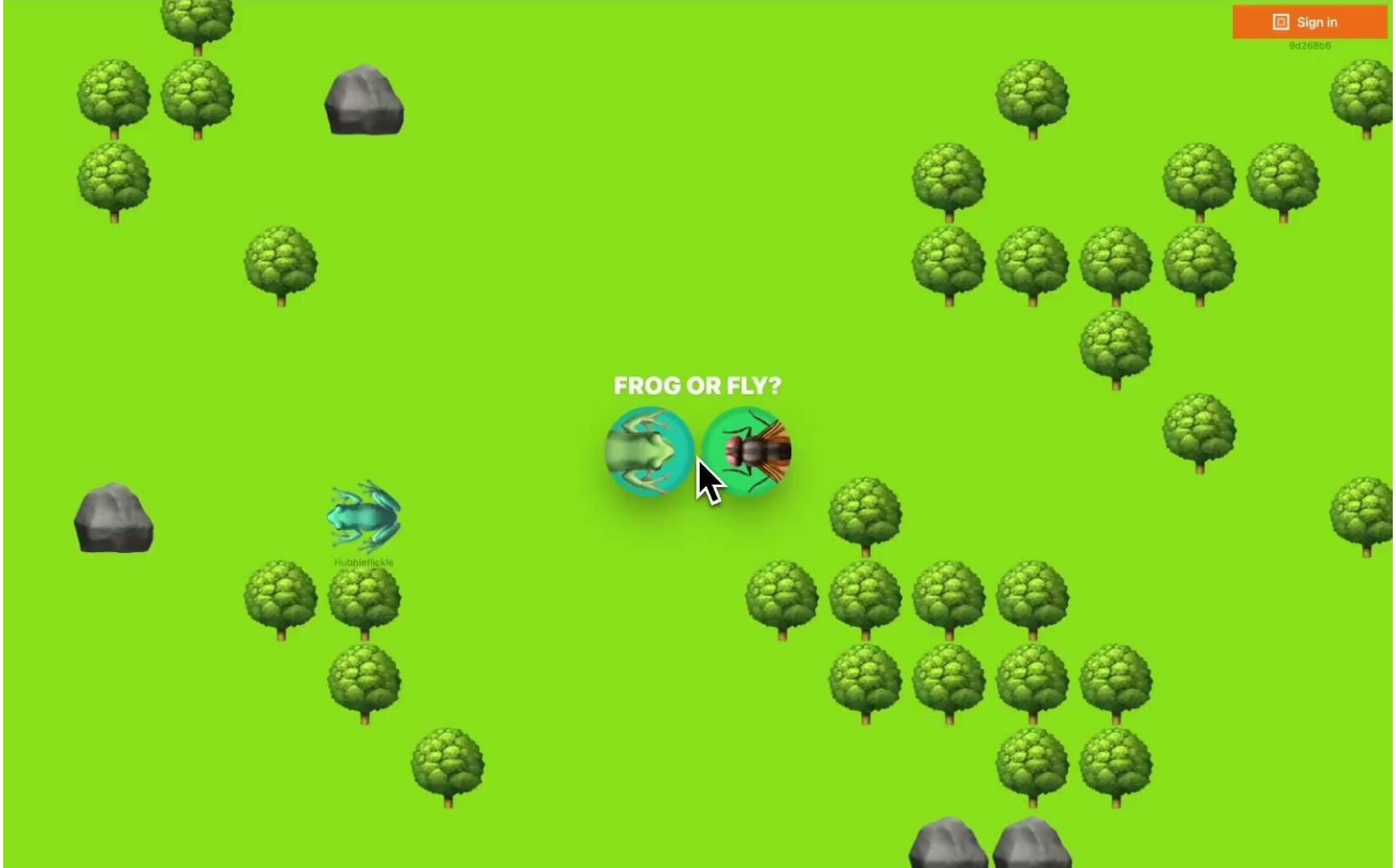

WORLD
STORE
CALL
CALL FROM
CALL WITH SIG
MOVE SYSTEM
HEAL SYSTEM
SYSTEMS
POSITION
HEALTH
ENERGY

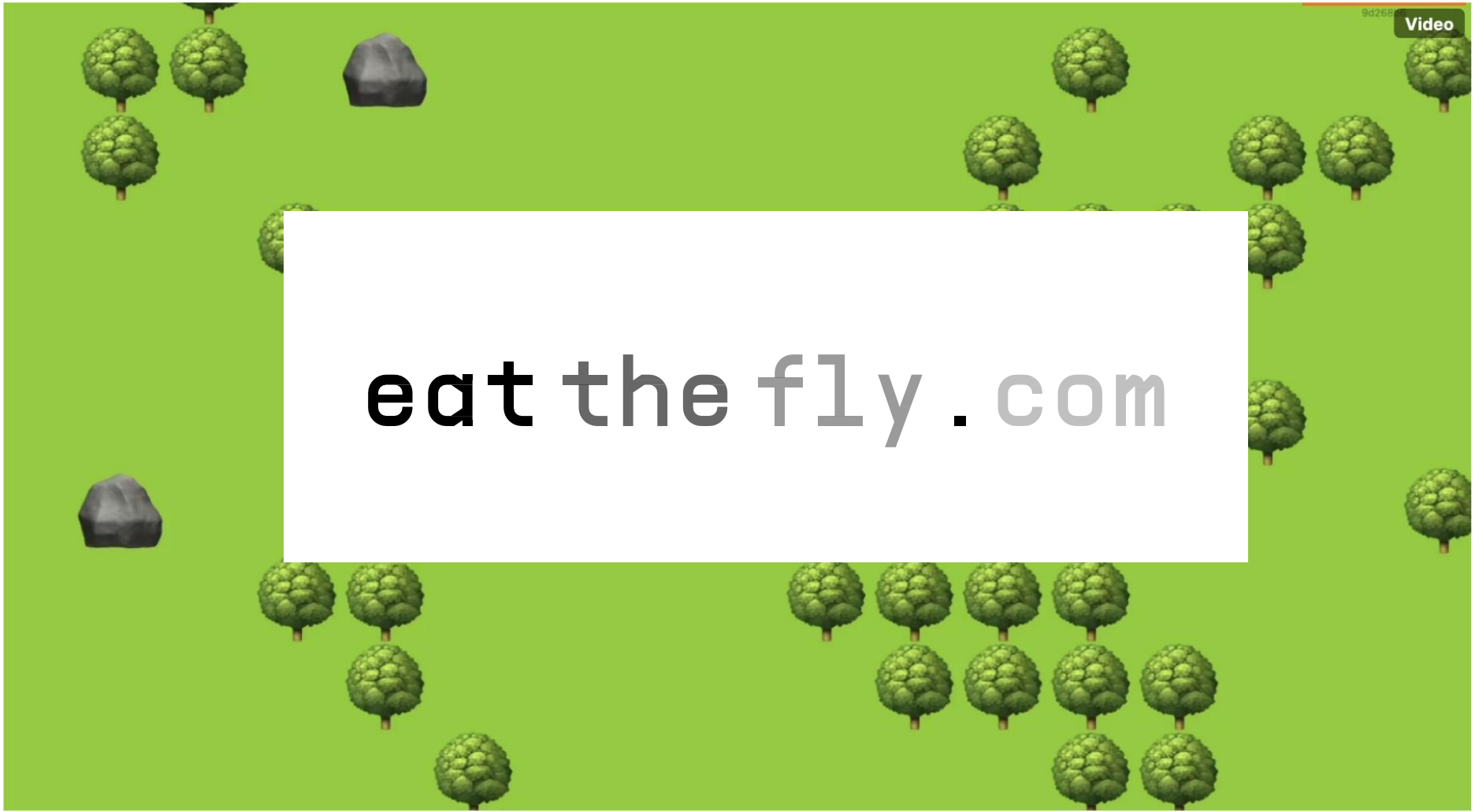

R
Account 1
0xb9CFD...E4c7C
0 ETH
Portfolio
+$0.00 (+0.00%)
Buy & Sell
Swap
Bridge
Send
Receive
Tokens
NFTs
Activity
You have no transactions
MetaMask support


Account 1
0xb9CFD...E4c7C
0 ETH
Portfolio
+$0.00 (+0.00%)
Buy & Sell
Swap
Bridge
Send
Receive
Tokens
NFTs
Activity
You have no transactions
MetaMask support
Account 1
Rhodolite Devnet
Request from
frog.onquarry.com
Interacting with
0x61f22...0a68D
Message
Primary type:
Call
Signer:
Account 1
SystemNamespace:
SystemName:
SpenderSystem
CallData:
0x960654620000000000000000000000000
b1568e3540c4fd4d1cc6c0c79f927c8bfb653a
5c
Nonce:
1
Cancel
Confirm


Account 1
Rhodolite Devnet
Request from
frog.onquarry.com
Interacting with
0x61f22...0a68D
Message
Primary type:
Call
Signer:
Account 1
SystemNamespace:
SystemName:
SpenderSystem
CallData:
0x960654620000000000000000000000000
b1568e3540c4fd4d1cc6c0c79f927c8bfb653a
5c
Nonce:
1
Cancel
Confirm


Account 1
Rhodolite Devnet
Request from
frog.onquarry.com
Interacting with
0x61f22...0a68D
Message
Primary type:
Call
Signer:
Account 1
SystemNamespace:
SystemName:
SpenderSystem
CallData:
0x9606546200000000000000000000000
b1568e3540c4fd4d1cc6c0c79f927c8bfb653a
5c
Nonce:
1
Cancel
Confirm


Sign in
9d268b6
FROG OR FLY?
Hubbleflickle

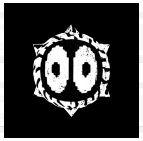
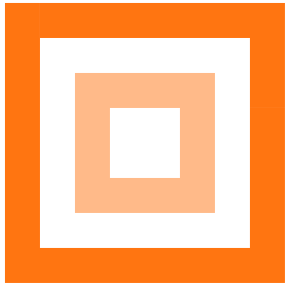

Video
eat the fly.com



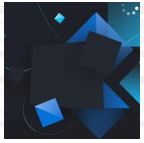




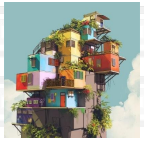
TX
MONSTER



MUD

EVE
FRONTIER

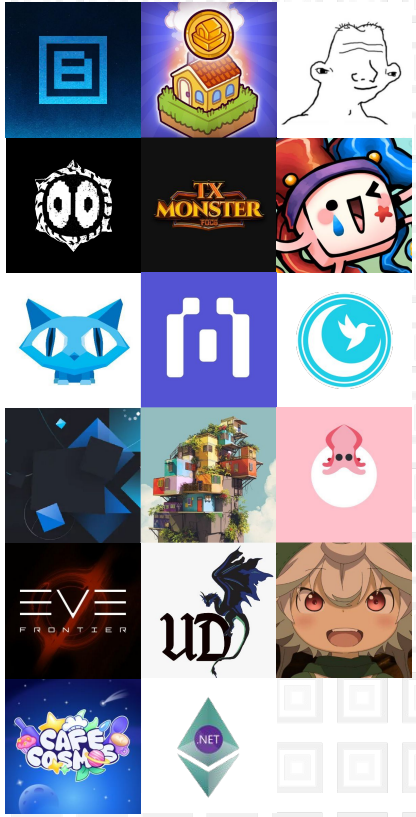



.NET



UD

CAFE
COSMOS

MUD DAY

TOMORROW (THURSDAY)

CLASSROOM A

MUD.DEV