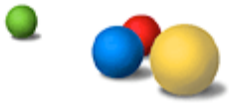
From E to EcmaScript and back again

Mark S. Miller and the Cajadores



Overview

- Object-Capabilities
  - Security as extreme modularity
- Securing JavaScript – Why and How?
  - E → Caja → ES5 → SES → Dr. SES
- Patterns of Safe Cooperation
  - In Secure EcmaScript (SES)
- Distributed Cryptographic Capabilities
  - In Distributed Resilient Secure EcmaScript (Dr. SES)

Security as Extreme Modularity

Modularity: Avoid needless dependencies

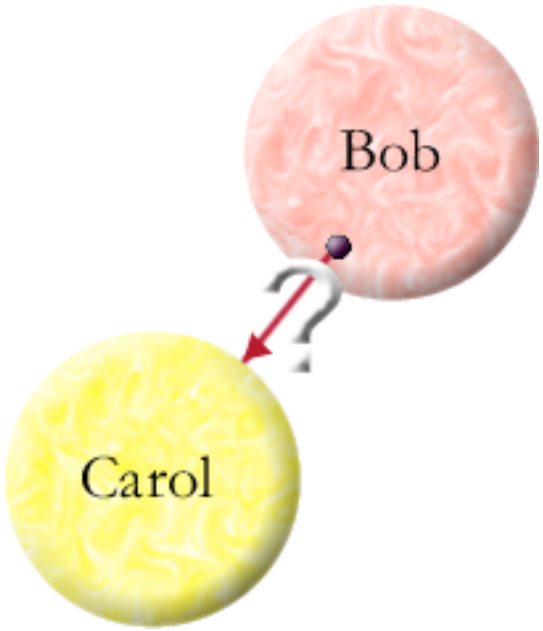
Security: Avoid needless vulnerabilities

Vulnerability is a form of dependency

Mod: Principle of info hiding - need to know.

Sec: Principle of least authority - need to do.

How do I designate thee?

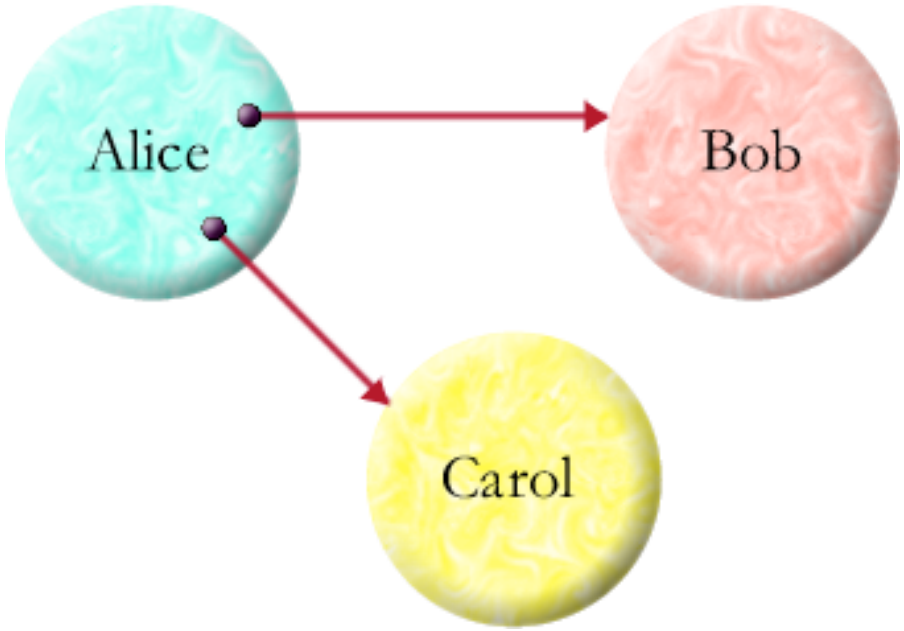


- by Introduction
  - ref to Carol
  - ref to Bob
  - decides to share
- by Parenthood
- by Endowment
- by Initial Conditions

How might object Bob come to know of object Carol?

How do I designate thee?

Alice says: bob.foo(carol)



by Introduction

- ref to Carol
- ref to Bob
- decides to share

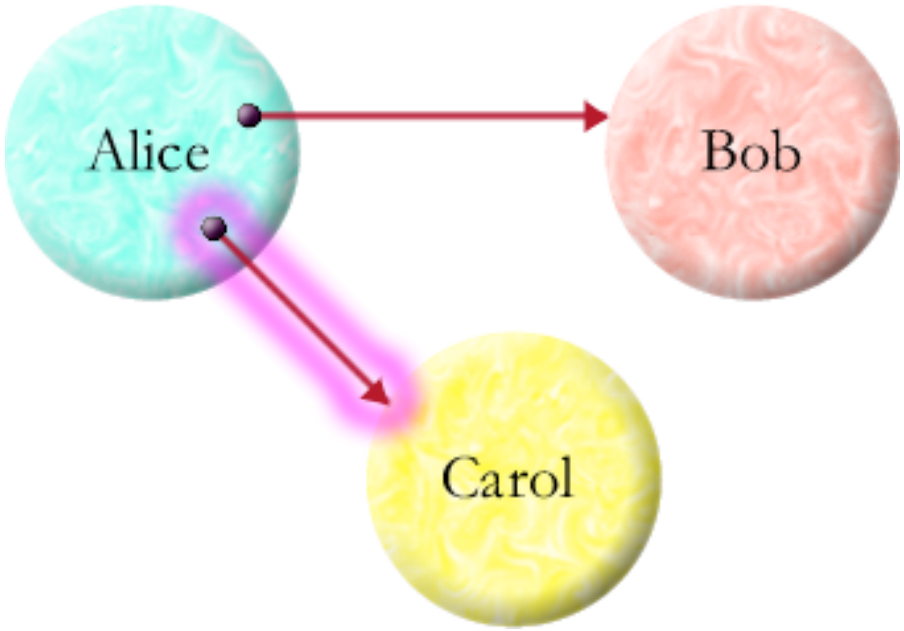
by Parenthood

by Endowment

by Initial Conditions

How do I designate thee?

Alice says: bob.foo(carol)



by Introduction

ref to Carol

ref to Bob

decides to share

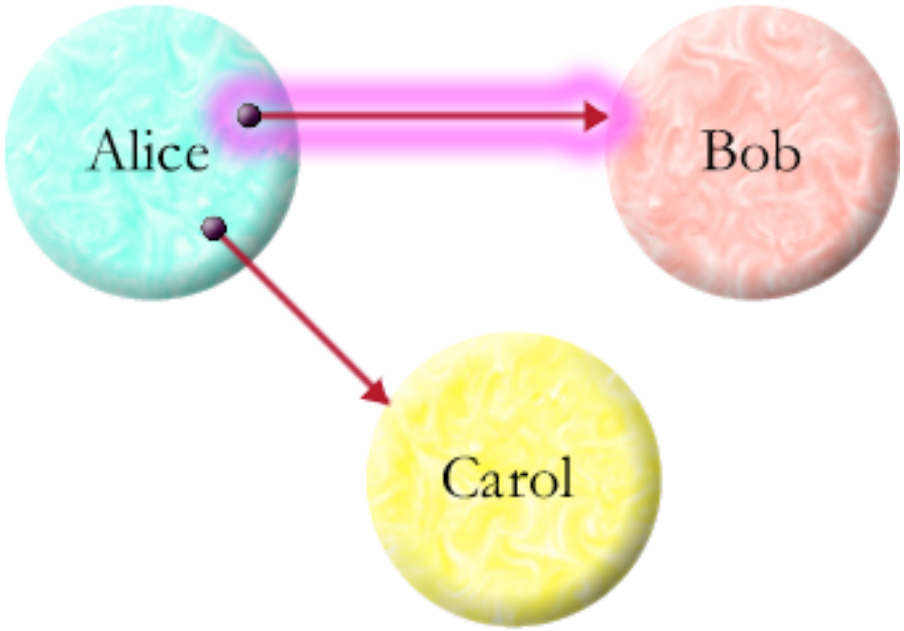
by Parenthood

by Endowment

by Initial Conditions

How do I designate thee?

Alice says: bob.foo(carol)



by Introduction

ref to Carol

ref to Bob

decides to share

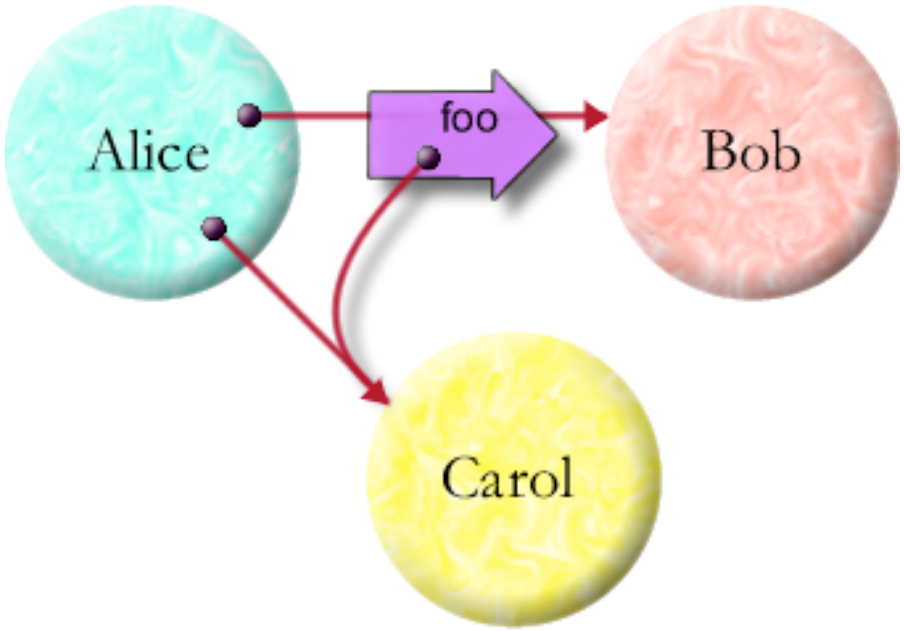
by Parenthood

by Endowment

by Initial Conditions

How do I designate thee?

Alice says: bob.foo(carol)



by Introduction

- *ref to Carol*
- *ref to Bob*
- ***decides to share***

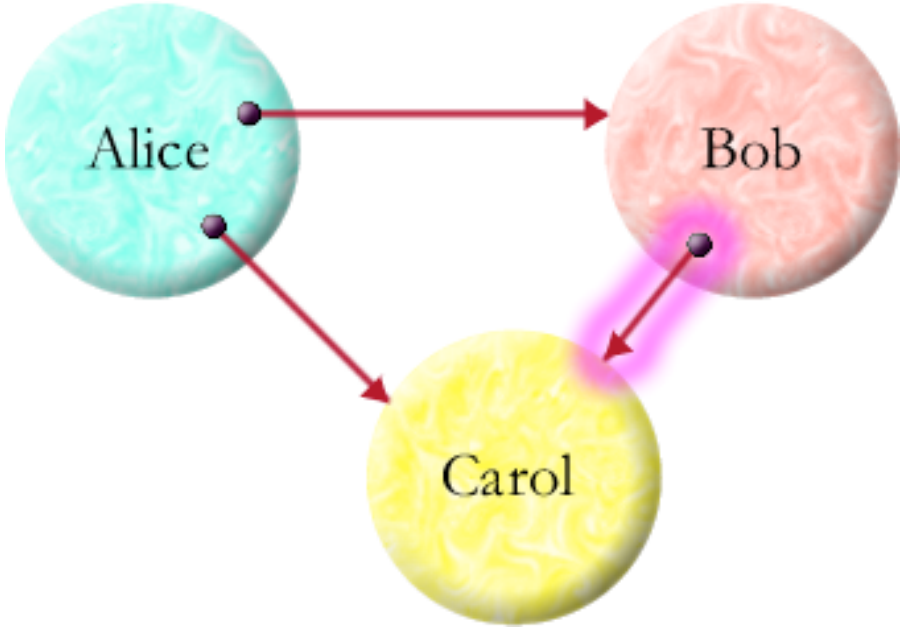
by Parenthood

by Endowment

by Initial Conditions

How do I designate thee?

Alice says: bob.foo(carol)



by Introduction

ref to Carol

ref to Bob

decides to share

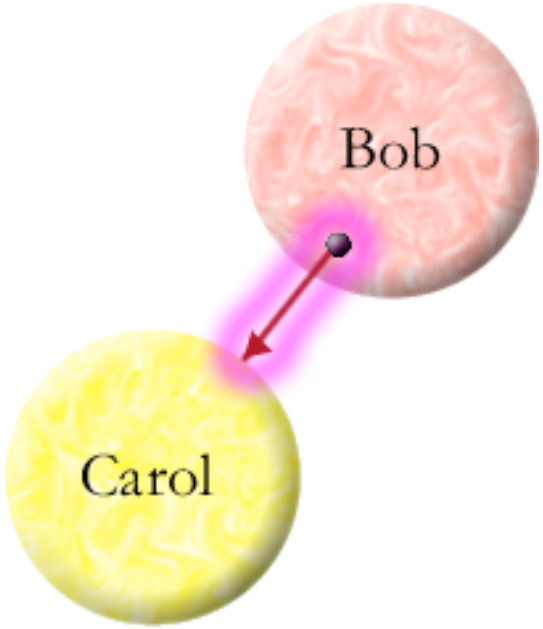
by Parenthood

by Endowment

by Initial Conditions

How do I designate thee?

Bob says: var *carol* = { ... };



by Introduction
- ref to Carol
- ref to Bob
- decides to share

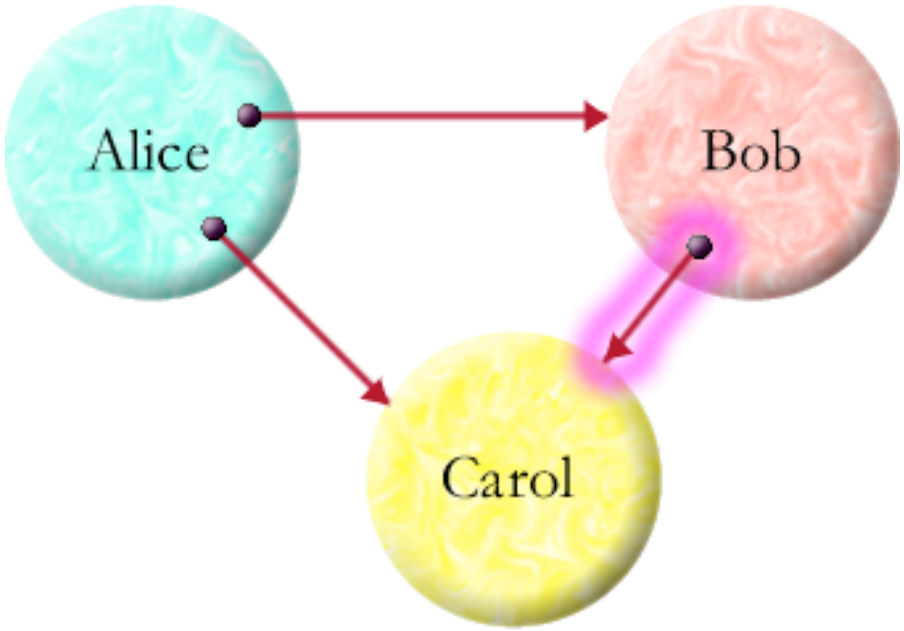
by Parenthood

by Endowment

by Initial Conditions

How do I designate thee?

Alice says: var *bob* = { ... carol ... };



by Introduction
- ref to Carol
- ref to Bob
- decides to share

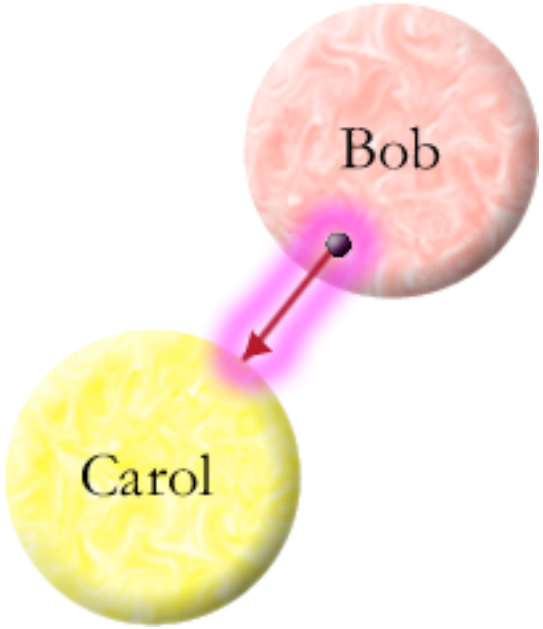
by Parenthood

by Endowment

by Initial Conditions

How do I designate thee?

At t_0:



by Introduction

- ref to Carol
- ref to Bob
- decides to share

by Parenthood

by Endowment

by Initial Conditions

OCaps: Small step from pure objects

Memory safety and encapsulation

\+ Effects ***only*** by using held references

\+ No powerful references by default

OCaps: Small step from pure objects

Memory safety and encapsulation

\+ Effects ***only*** by using held references

\+ No powerful references by default

Reference graph ≡ Access graph

Only connectivity begets connectivity

Natural *Least Authority*

OO expressiveness for security patterns

The Mashup problem: Code as Media

```
<html> <head> <title>Basic Mashup</title> <script>
   function animate(id) {
      var element = document.getElementById(id);
      var textNode = element.childNodes[0];
      var text = textNode.data;
      var reverse = false;
      element.onclick = function() { reverse = !reverse; };
      setInterval(function() {
         textNode.data = text = reverse ? text.substring(1) + text[0]
                  : text[text.length-1] + text.substring(0, text.length-1);
      }, 100);
   }
</script> </head> <body onload="animate('target')">
   <pre id="target">Hello Programmable World!  </pre>
</body> </html>
```

caja-corkboard.appspot.com

— kpreid.switchb.org, 2010-07-24 00:43:10.844801

View Source

...ggle about. -- David-Sarah

What version and OS? Can't reproduce on either machine I have handy. -- kpreid

— Anon, 2010-07-24 00:41:44.706661

Edit Delete

Unicode test. You should see two bullets and two (if you've got the font for it) U+1040E DESERET CAPITAL LETTER WU (interleaved).
•𐐎•𐐎

— kpreid.switchb.org, 2010-07-23 00:29:17.917977

View Source

Sean B. Palmer

— kpreid.switchb.org, 2010-07-22 17:05:53.107415

View Source

— erights@google.com (Logout), just now

Post This

This is a Caja demo. You can enter any HTML you like, and it will display as well as we currently support and yet not allow you to take over anyone else's postings or otherwise disrupt the application (other than by making the page load slower or hang).

This site is intended to demonstrate how to straightforwardly use Caja in a web application as a "better HTML sanitizer"; see CorkboardDemo on the Caja wiki for a tutorial.

Background image by Parée Erica (used under Creative Commons Attribution license).

caja-corkboard.appspot.com

— kpreid.switchb.org, 2010-07-24 00:43:10.844801

View Source

...ggle about. -- David-Sarah

What version and OS? Can't reproduce on either machine I have handy. -- kpreid

— Anon, 2010-07-24 00:41:44.706661

Edit Delete

Unicode test. You should see two bullets and two (if you've got the font for it) U+1040E DESERET CAPITAL LETTER WU (interleaved).
•𐐎•𐐎

— kpreid.switchb.org, 2010-07-23 00:29:17.917977

View Source

Sean B. Palmer

— kpreid.switchb.org, 2010-07-22 17:05:53.107415

View Source

```
<html> <head> <title>Basic Mashup</title> <script>
  function animate(id) {
    var element = document.getElementById(id);
    var textNode = element.childNodes[0];
    var text = textNode.data;
    var reverse = false;
    element.onclick = function() { reverse = !reverse; };
    setInterval(function() {
      textNode.data = text = reverse ? text.substring(1) + text[0]
          : text[text.length-1] + text.substring(0, text.length-1);
    }, 100);
  }
</script> </head> <body onload="animate('target')">
  <pre id="target">Hello Programmable World!  </pre>
</body> </html>
```

— erights@google.com (Logout), just now

Post This

This is a Caja demo. You can enter any HTML you like, and it will display as well as we currently support and yet not allow you to take over anyone else's postings or otherwise disrupt the application (other than by making the page load slower or hang).

This site is intended to demonstrate how to straightforwardly use Caja in a web application as a "better HTML sanitizer"; see CorkboardDemo on the Caja wiki for a tutorial.

Background image by Parée Erica (used under Creative Commons Attribution license).

caja-corkboard.appspot.com

Caja Corkboard Demo

grammable World! Hello Pro

— *erights@google.com, 2010-10-04 13:30:40.185506*

Edit Delete

(Error contacting Caja service)

— *kpreid.switchb.org, 2010-08-22 12:26:41.953037*

View Source

Greetings from Rosetta Code!

Not just a <marquee>:

World! Hello

— *kpreid.switchb.org, 2010-08-13 19:06:55.712467*

View Source

Cajoling-of-URLs test: you should see 2 links to google.com and 2 images.

Static	Dynamic
Link	Link
	

— *kpreid.switchb.org, 2010-08-13 00:27:22.459179*

View Source

Testing 123.

— *kpreid.switchb.org, 2010-08-10 22:21:44.542621*

View Source

— *kpreid.switchb.org, 2010-07-24 00:43:10.844801*

View Source

Improving JavaScript in Stages

EcmaScript 3:

One of the hardest oo languages to secure.

Caja: Complex server-side translator. Runtime overhead.

EcmaScript 5:

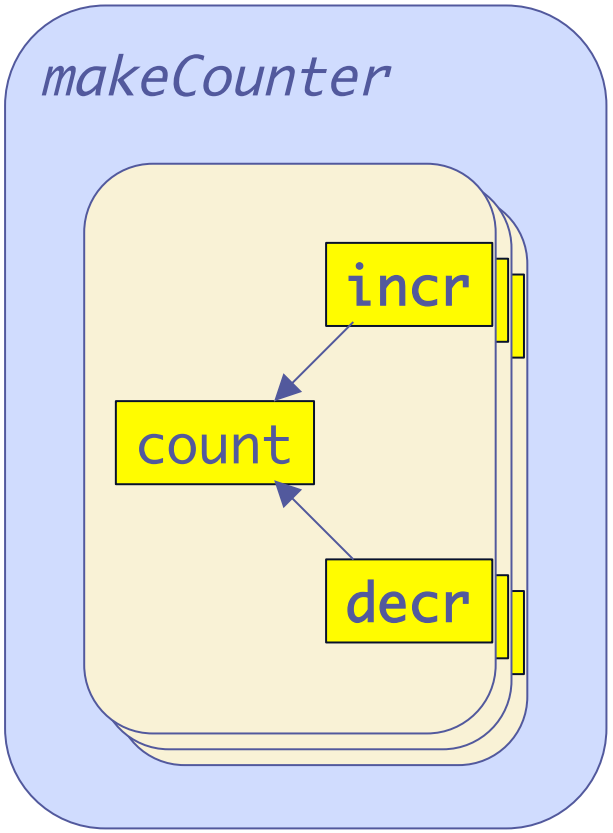
One of the easiest oo languages to secure.

<script src="initSES.js"></script>

Simple client-side init and verifier. No runtime overhead.

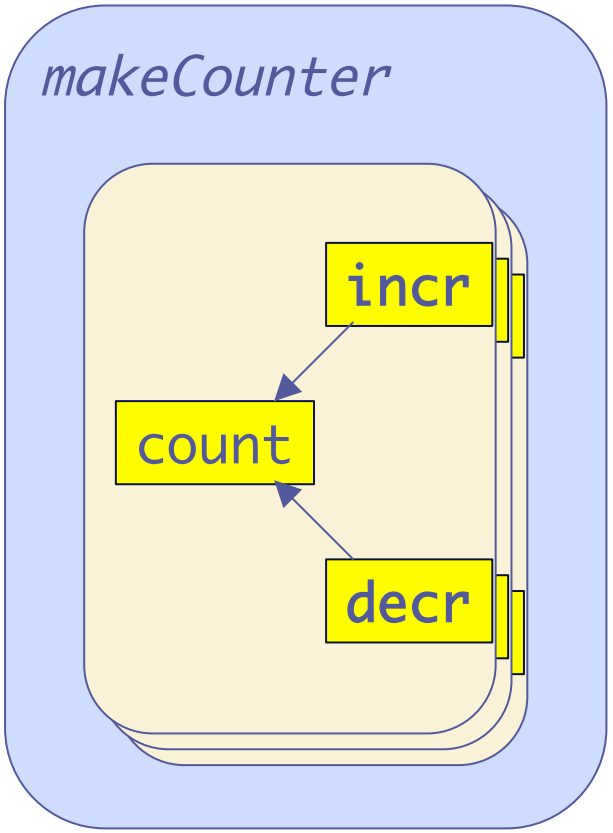
Approx 3K download compressed.

Objects as Closures




```
function makeCounter() {
    var count = 0;
    return {
        incr: function() { return ++count; },
        decr: function() { return –count; }
    };
}
```

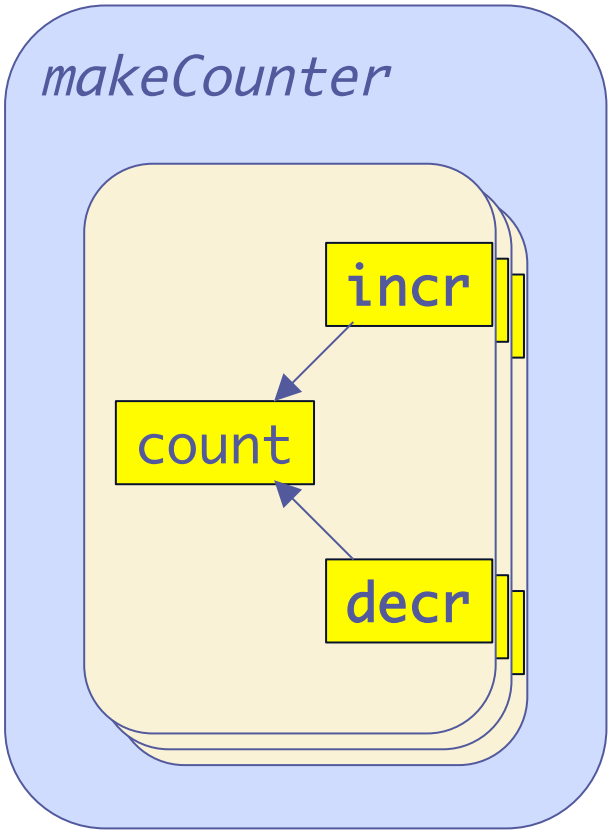
Objects as Closures




```
function makeCounter() {
    var count = 0;
    return {
        incr: function() { return ++count; },
        decr: function() { return –count; }
    };
}
```

A *record* of *closures* hiding *state*
is a fine representation of an
object of *methods* hiding *instance vars*

Objects as Closures in ES5/strict




```
"use strict";
function makeCounter() {
    var count = 0;
    return def({
        incr: function() { return ++count; },
        decr: function() { return –count; }
    });
}
```

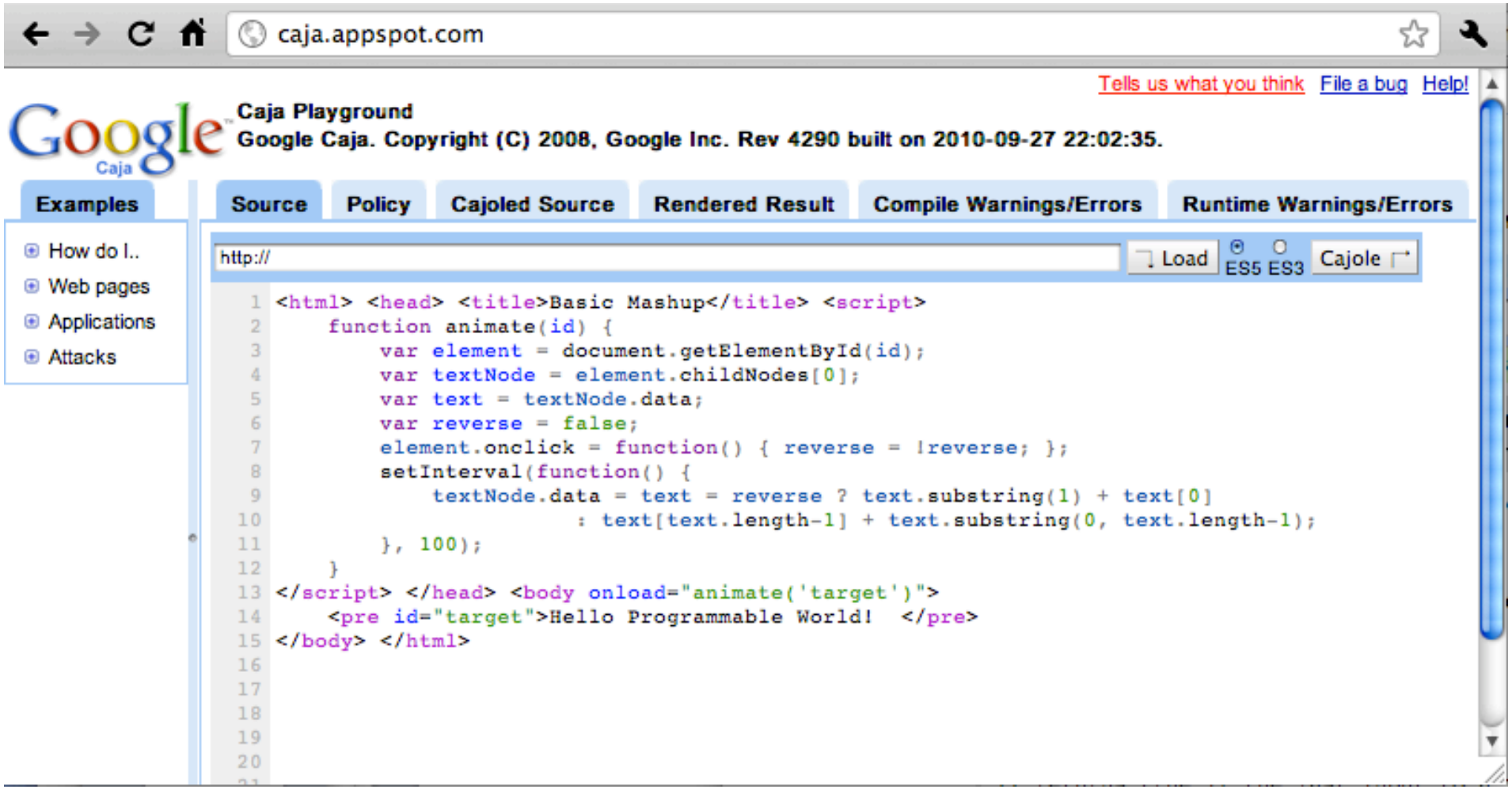
A *tamper-proof* record of *lexical* closures *encapsulating* state is a *defensive* object

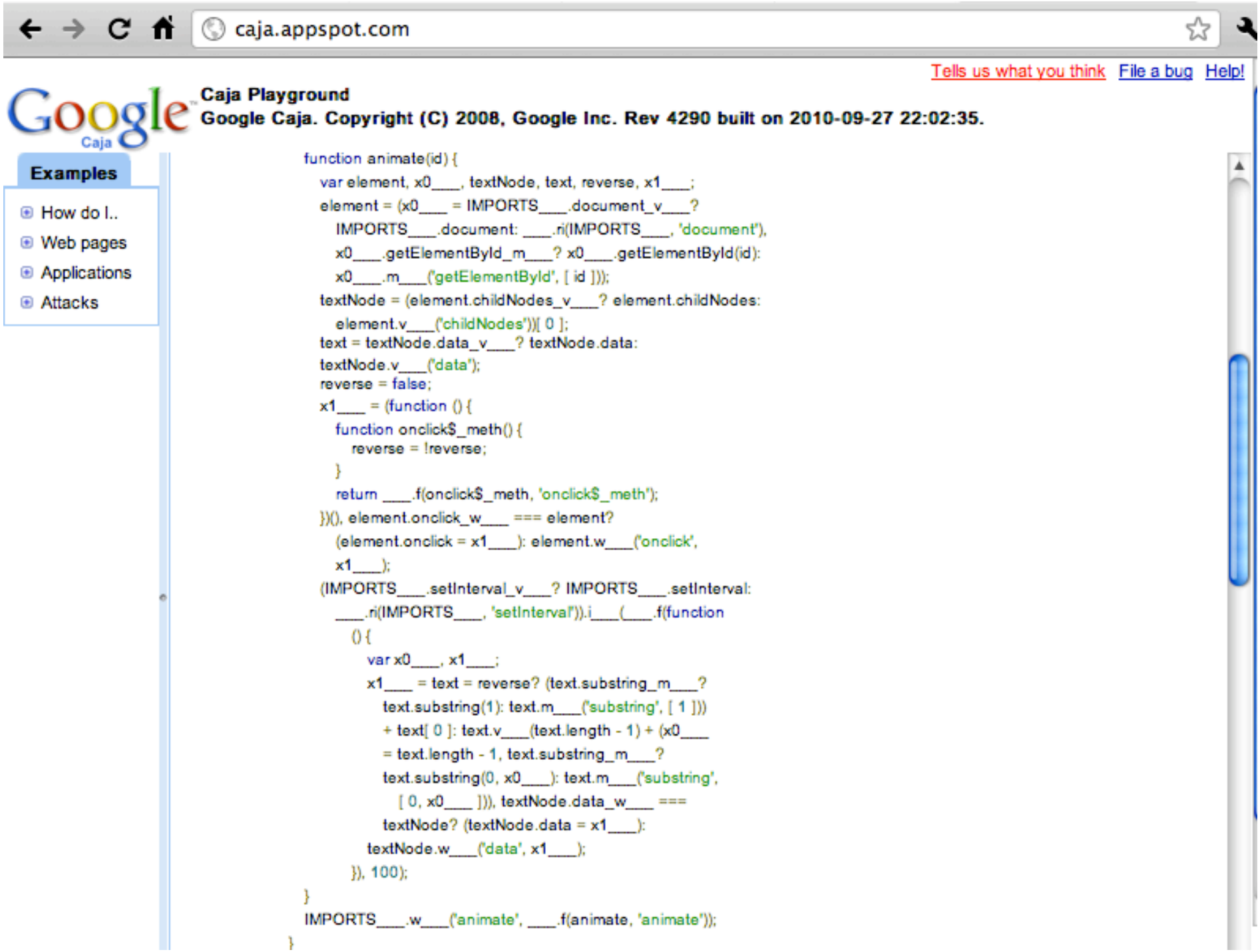
Turning ES5 into SES

```
<script src="initSES.js"></script>
```

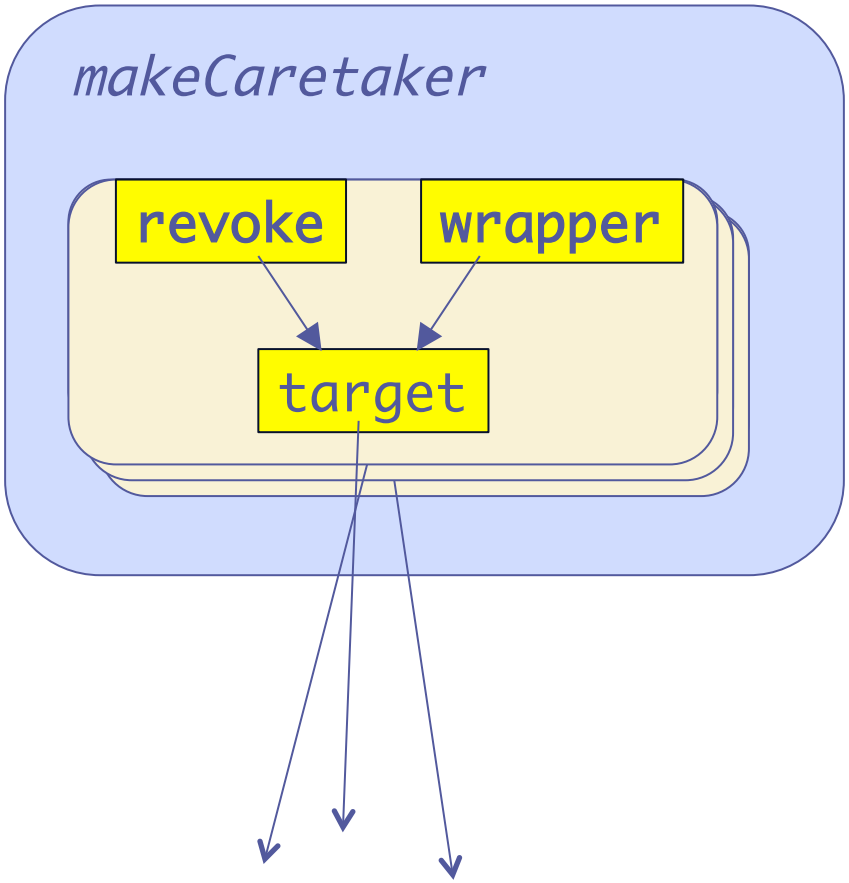
- Monkey patch away bad non-std behaviors
- Remove non-whitelisted primordials
- Install leaky **WeakMap** emulation
- Make virtual global **root**
- Freeze whitelisted global variables
- Replace **eval** & **Function** with safe alternatives
- Freeze accessible primordials

Running ES5 & SES on old browsers



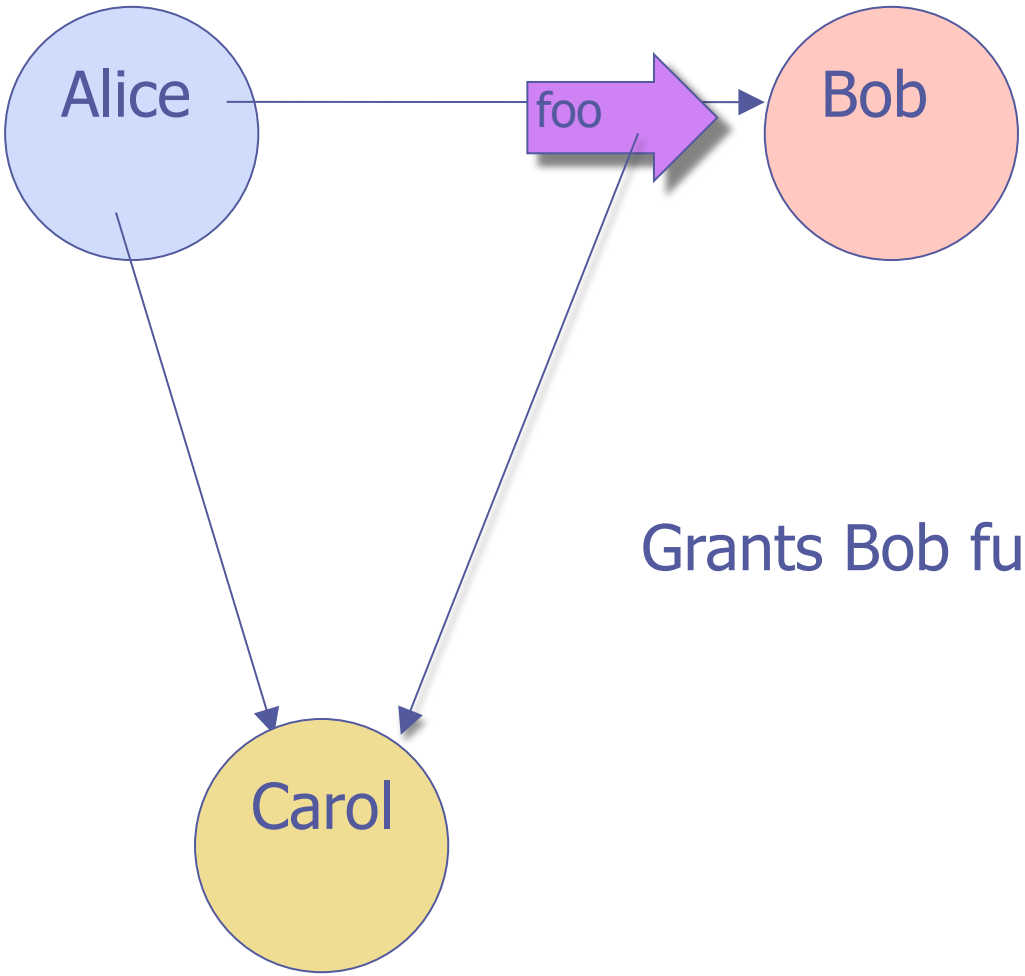

caja.appspot.com
Tells us what you think
File a bug
Help!
Google Caja
Caja Playground
Google Caja. Copyright (C) 2008, Google Inc. Rev 4290 built on 2010-09-27 22:02:35.
Examples
How do I..
Web pages
Applications
Attacks
function animate(id) {
var element, x0___, textNode, text, reverse, x1___;
element = (x0___ = IMPORTS___.document_v___?
IMPORTS___.document: ___.ri(IMPORTS___, 'document'),
x0___.getElementById_m___? x0___.getElementById(id):
x0___.m___('getElementById', [id]));
textNode = (element.childNodes_v___? element.childNodes:
element.v___('childNodes'))[0];
text = textNode.data_v___? textNode.data:
textNode.v___('data');
reverse = false;
x1___ = (function () {
function onclick$_meth() {
reverse = !reverse;
}
return ___.f(onclick$_meth, 'onclick$_meth');
})(), element.onclick_w___ === element?
(element.onclick = x1___): element.w___('onclick',
x1___);
(IMPORTS___.setInterval_v___? IMPORTS___.setInterval:
___.ri(IMPORTS___, 'setInterval')).i___(___.f(function
() {
var x0___, x1___;
x1___ = text = reverse? (text.substring_m___?
text.substring(1): text.m___('substring', [1]))
+ text[0]: text.v___(text.length - 1) + (x0___
= text.length - 1, text.substring_m___?
text.substring(0, x0___): text.m___('substring',
[0, x0___])), textNode.data_w___ ===
textNode? (textNode.data = x1___):
textNode.w___('data', x1___);
}), 100);
}
IMPORTS___.w___('animate', ___.f(animate, 'animate'));
}

Revocable Function Forwarder




```
function makeFnCaretaker(target) {
    return def({
        wrapper: function(…args) {
            return target(…args);
        },
        revoke: function() { target = null; }
    });
}
```

Unconditional Access

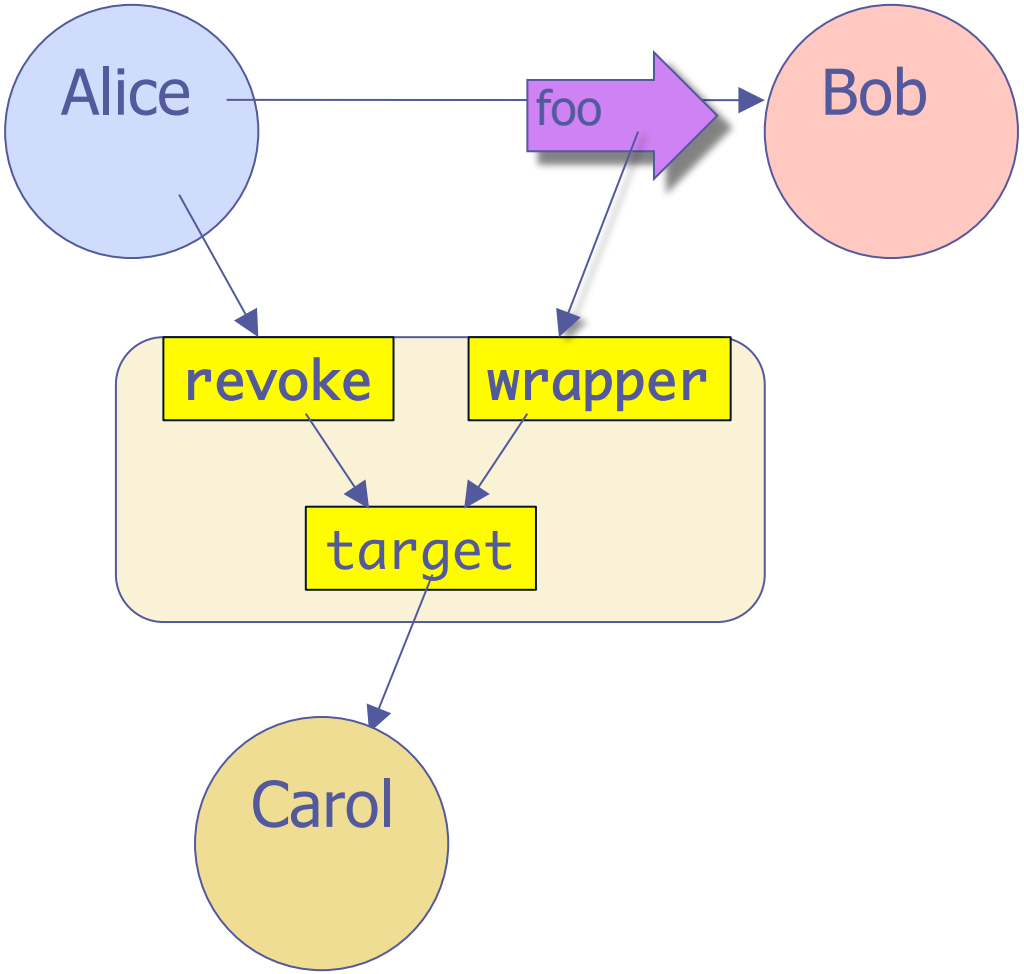


Alice says:

bob.foo(carol);

Grants Bob full access to Carol forever

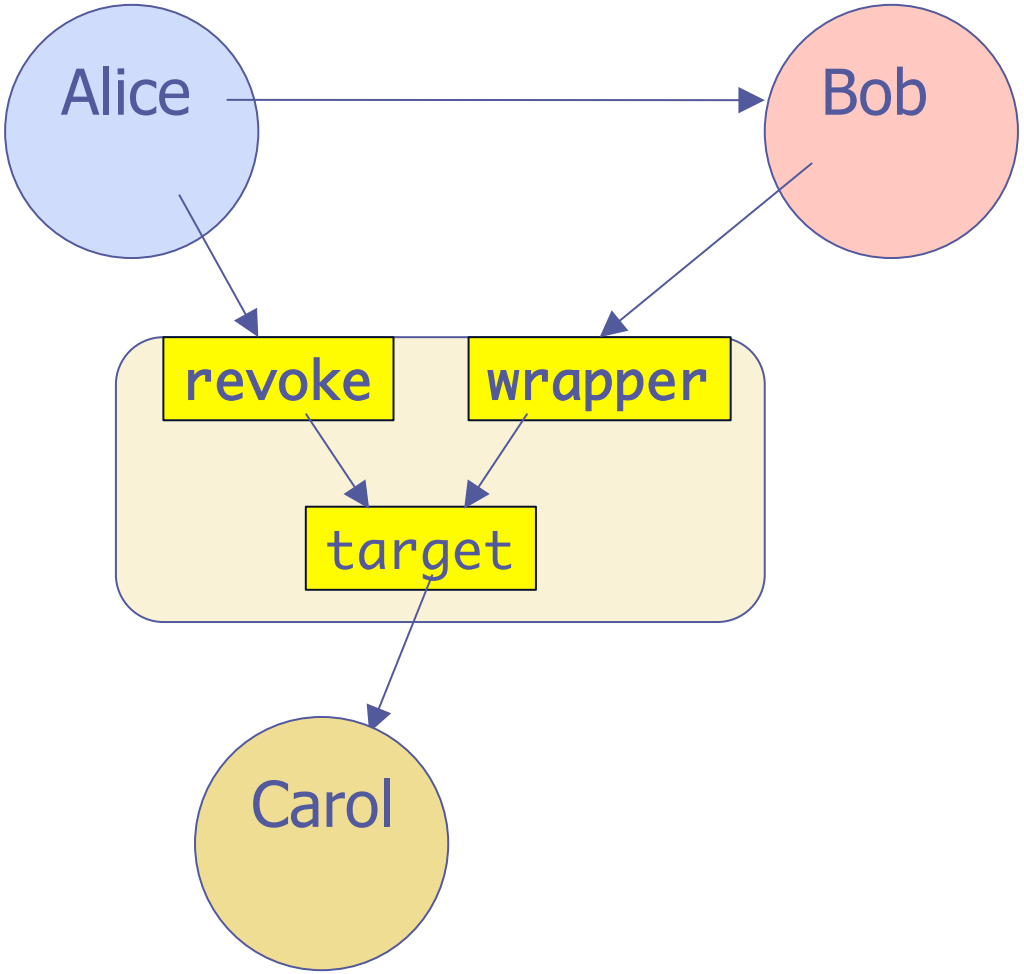
Revocability ≡ Temporal attenuation



Alice says:

```
var ct = makeCaretaker(carol);
bob.foo(ct.wrapper);
```

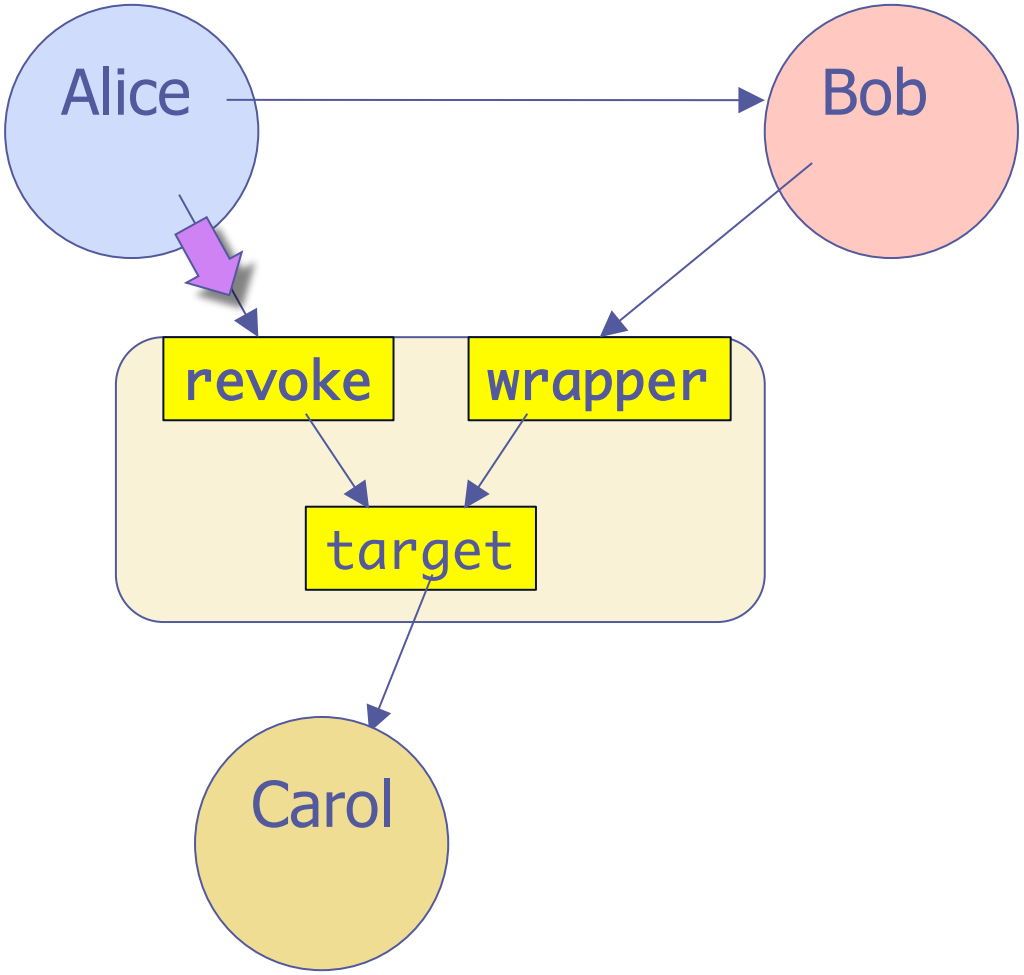
Revocability ≡ Temporal attenuation



Alice says:

```
var ct = makeCaretaker(carol);
bob.foo(ct.wrapper);
//…
```

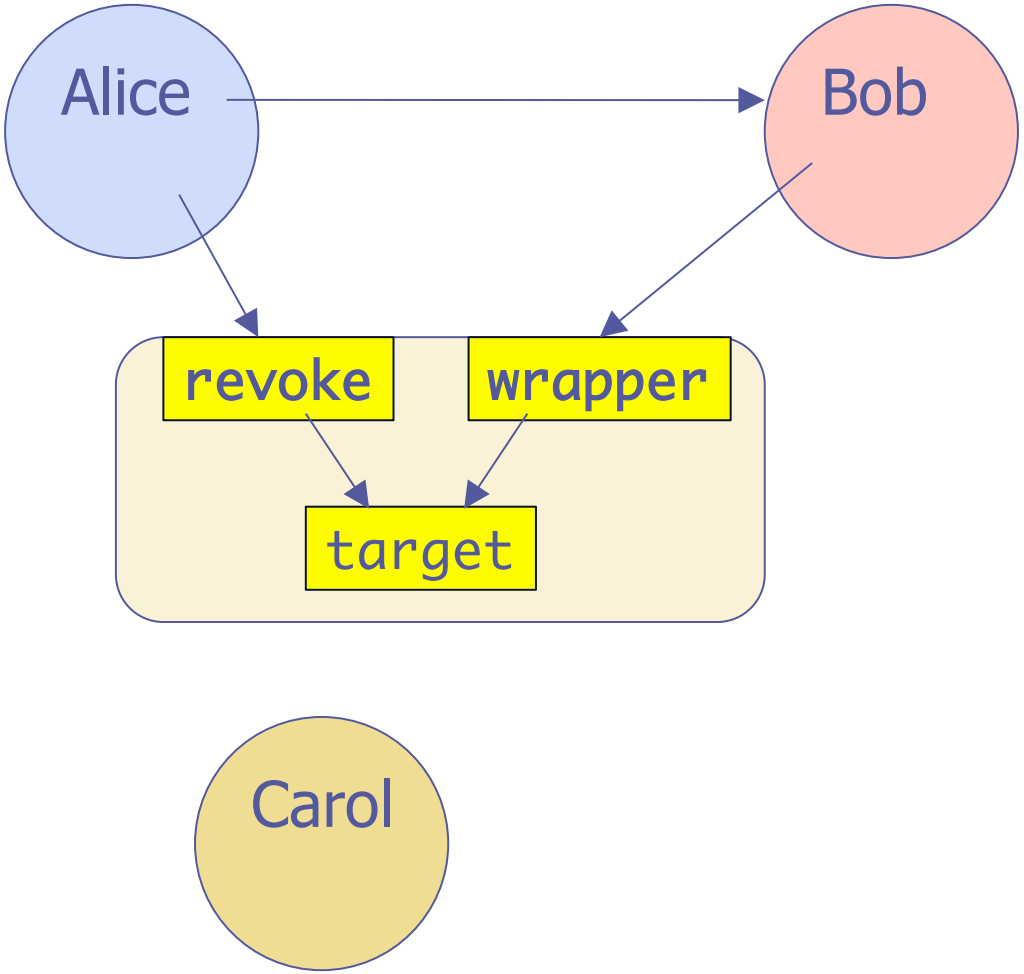
Revocability ≡ Temporal attenuation



Alice says:

```
var ct = makeCaretaker(carol);
bob.foo(ct.wrapper);
//…
ct.revoke();
```

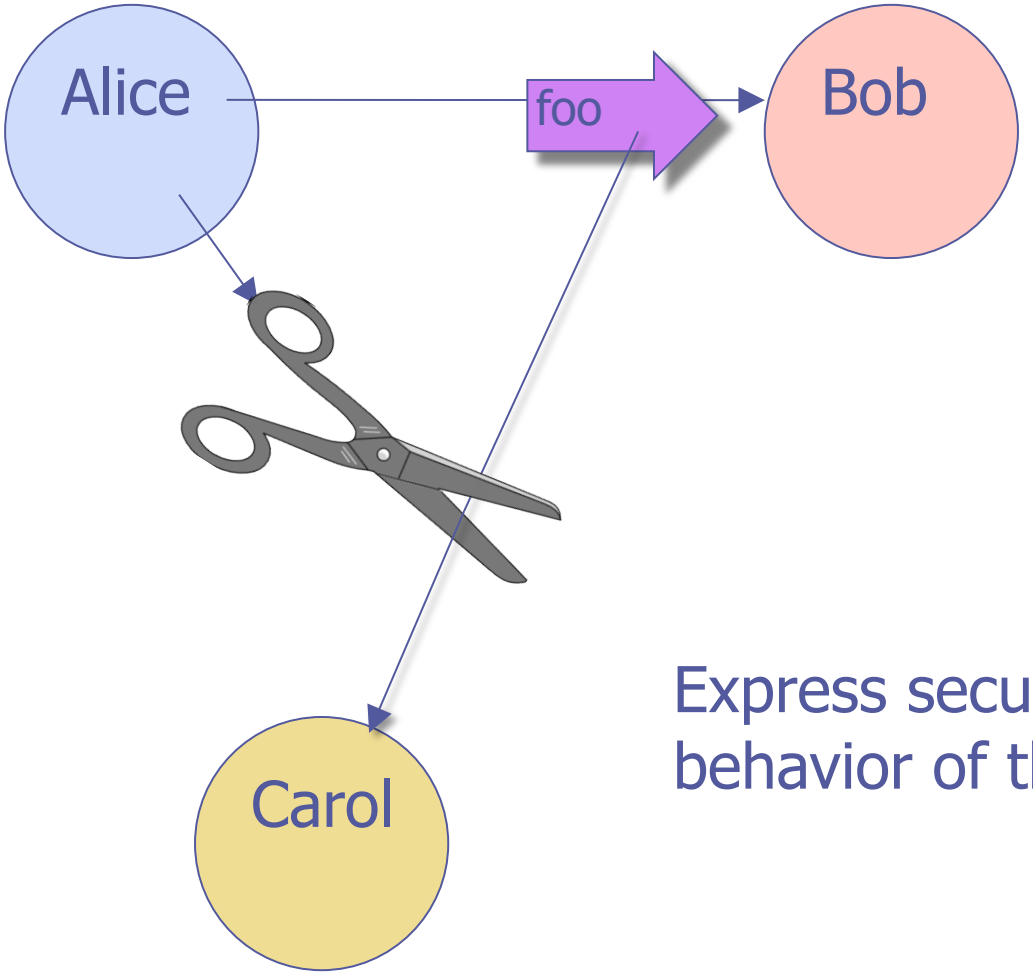
Revocability ≡ Temporal attenuation



Alice says:

```
var ct = makeCaretaker(carol);
bob.foo(ct.wrapper);
//…
ct.revoke();
```

Attenuators ≡ Access Abstractions



Alice says:

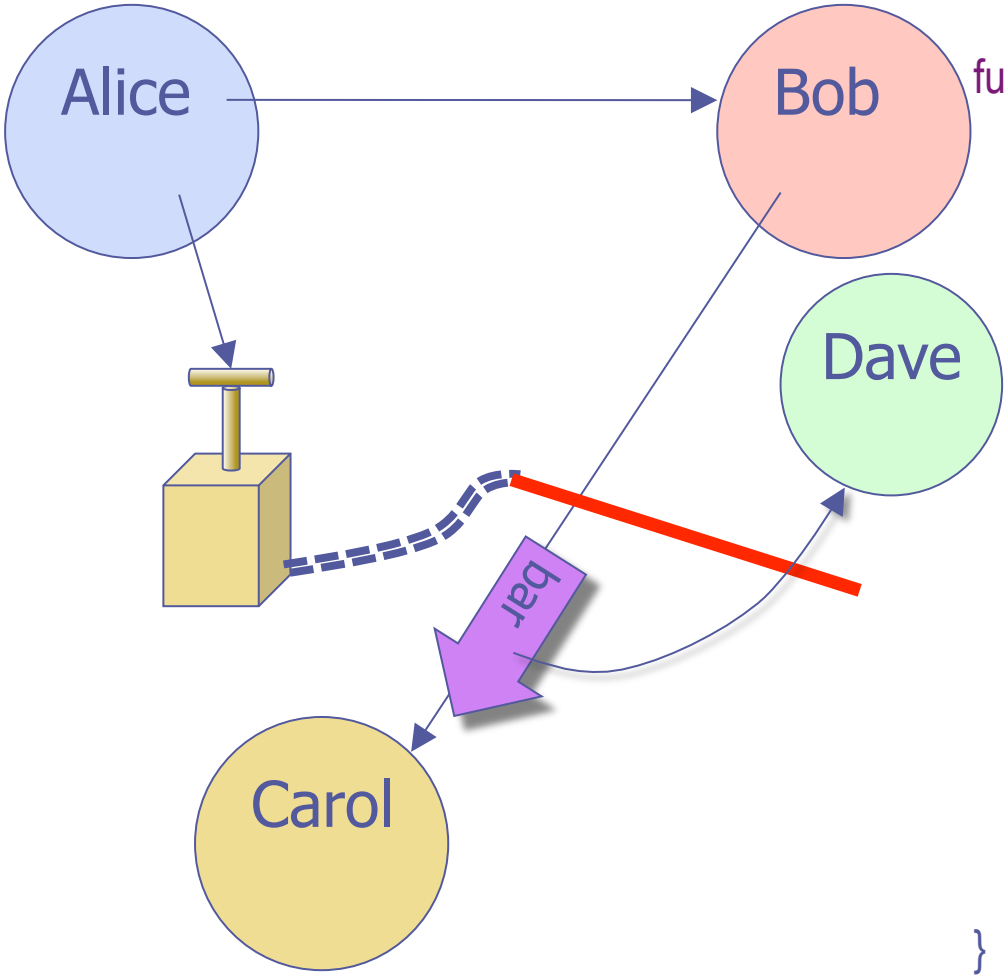
```
var ct = makeCaretaker(carol);
bob.foo(ct.wrapper);
```

Express security policy by the behavior of the objects you provide

Abstractions extend vocabulary

Primitives	**Abstraction Forms**	**Extended Vocabulary**
+, ., []	*procedural abstraction*	foo(bar, baz), …
int, struct, array	*data abstraction*	Point, Window, …
if, while, switch	*control abstraction*	addListener, …
points-to	*access abstraction*	caretaker, membrane, …

Membranes: Transitive Interposition

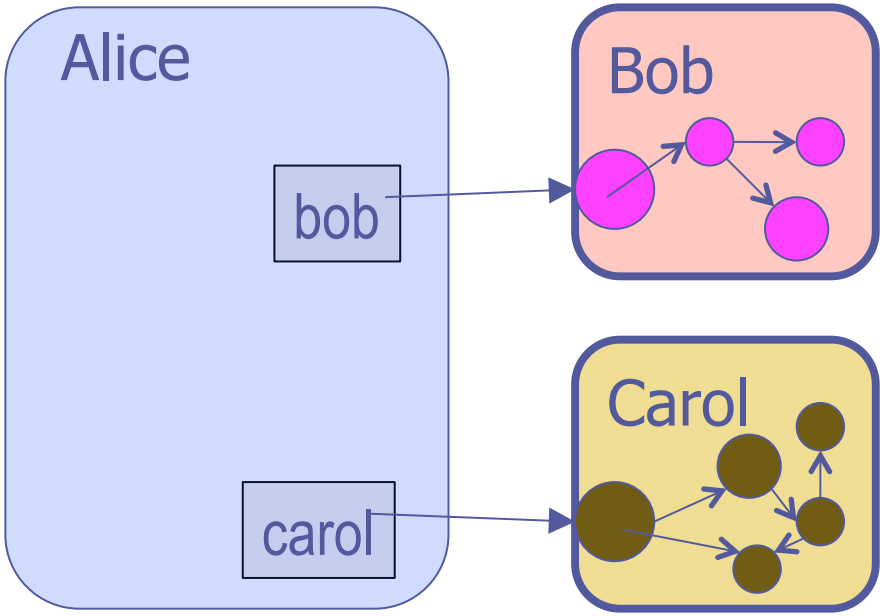



```
function makeFnMembrane(target) {
    var enabled = true;
    function wrap(wrapped) {
        if (wrapped !== Object(wrapped)) {
            return wrapped;
        }
        return function(…args) {
            if (!enabled) { throw new Error("revoked"); }
            return wrap(wrapped(…args.map(wrap));
    } }
    return def({
        wrapper: wrap(target),
        revoke: function() { enabled = false; }
    });
}
```

Attenuators Compose

```
function makeROFile(file) {
    return def({
        read: file.read,
        getLength: file.getLength
    });
}
var rorFile = makeROFile(revocableFile);
```

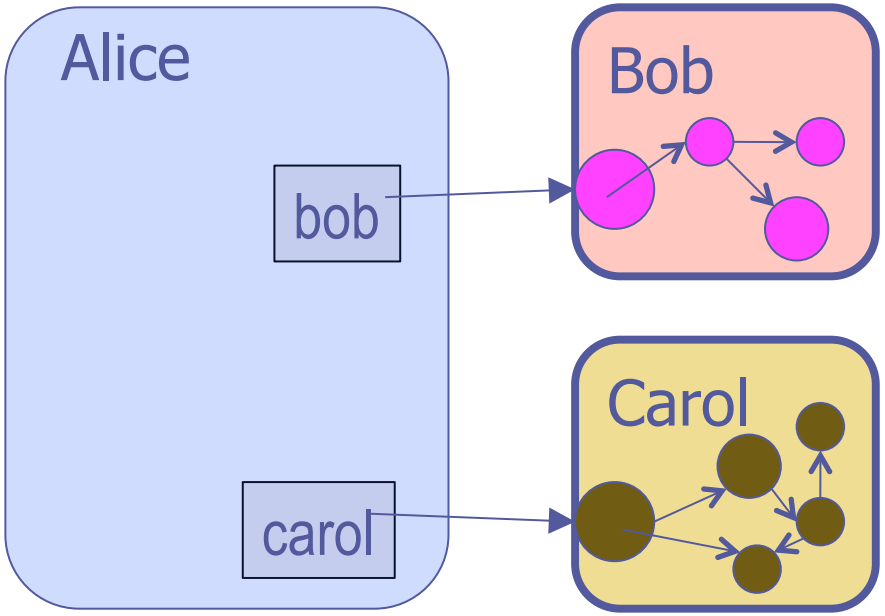
No powerful references by default



Alice says:

```
var bobSrc = //site B
var carolSrc = //site C
var bob = eval(bobSrc);
var carol = eval(carolSrc);
```

No powerful references by default



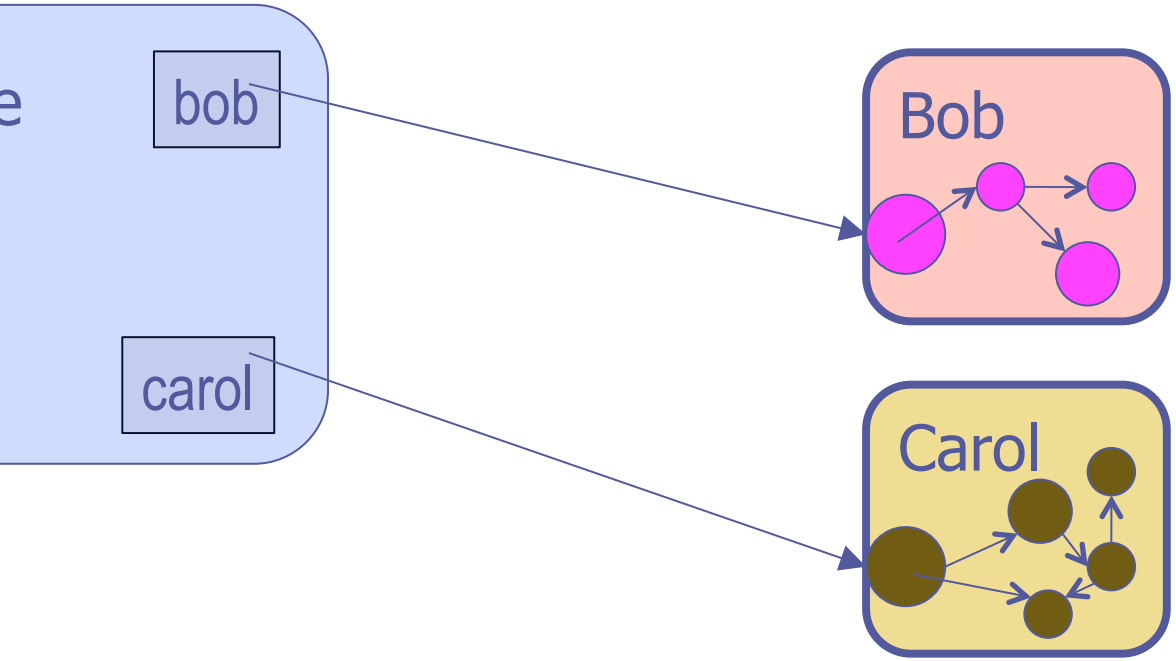
Alice says:

```
var bobSrc = //site B
var carolSrc = //site C
var bob = eval(bobSrc);
var carol = eval(carolSrc);
```

Bob and Carol are ***confined***.

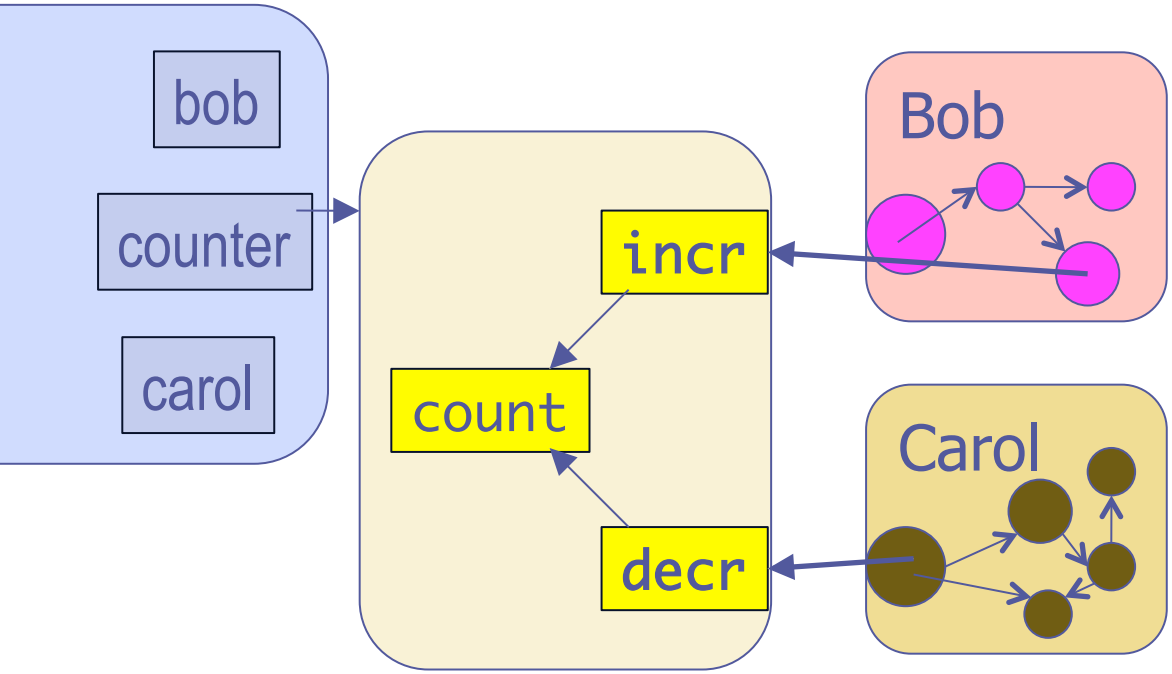
Only Alice controls how they can interact or get more connected.

No powerful references by default



Alice says:

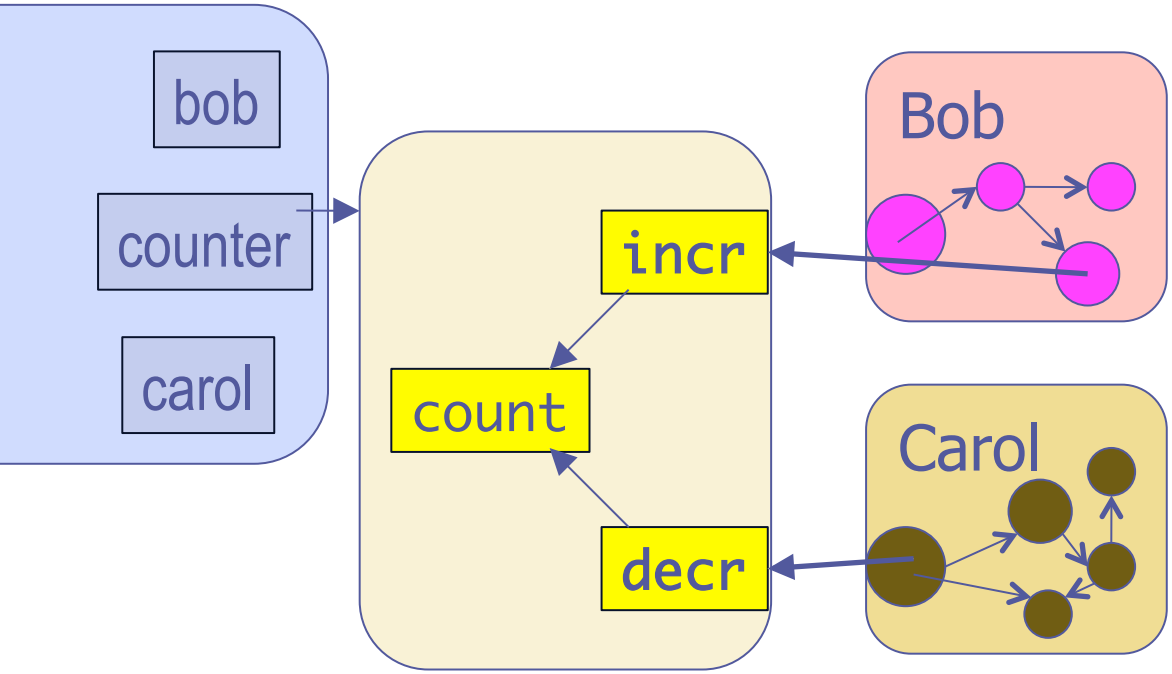
Only connectivity begets connectivity



Alice says:

```
var counter = makeCounter();
bob(counter.incr);
carol(counter.decr);
bob = carol = null;
```

Only connectivity begets connectivity



Alice says:

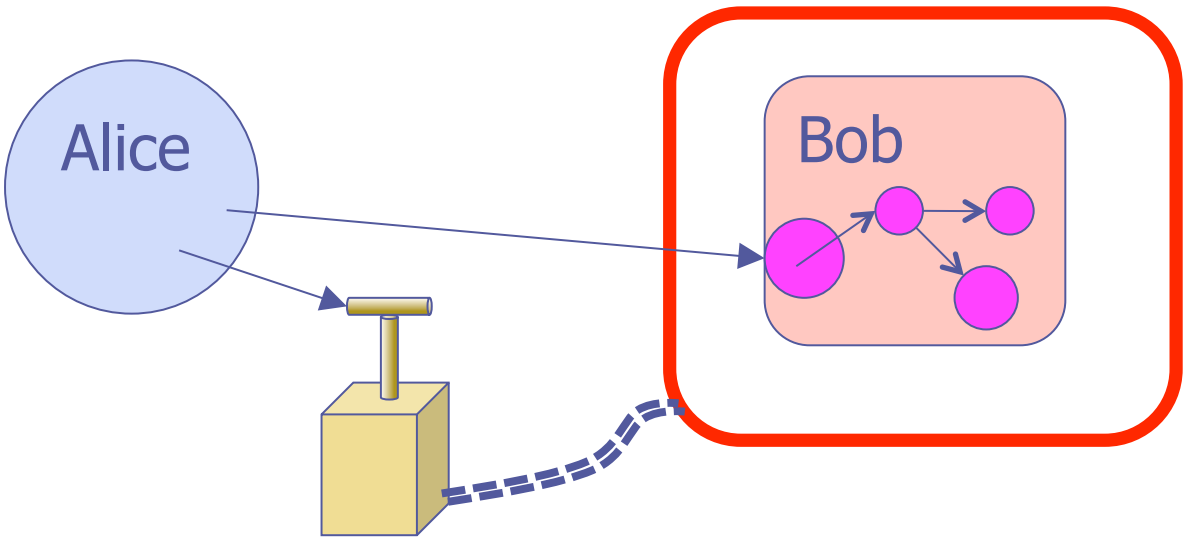
```
var counter = makeCounter();
bob(counter.incr);
carol(counter.decr);
bob = carol = null;
```

Bob can only count up and see result. Carol only down.

Alice can only do both.

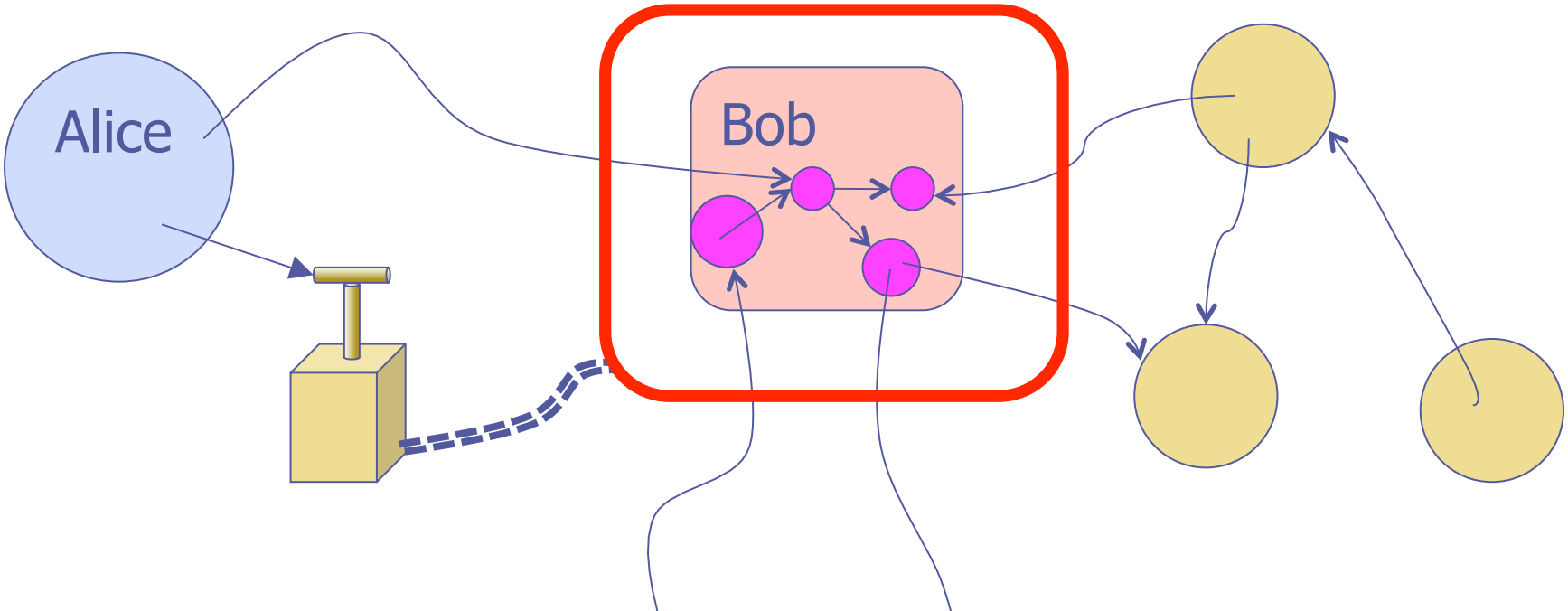
Membrane eval → compartment

var *compartment* = makeMembrane(eval);
var *vbob* = compartment.wrapper(bobSrc);



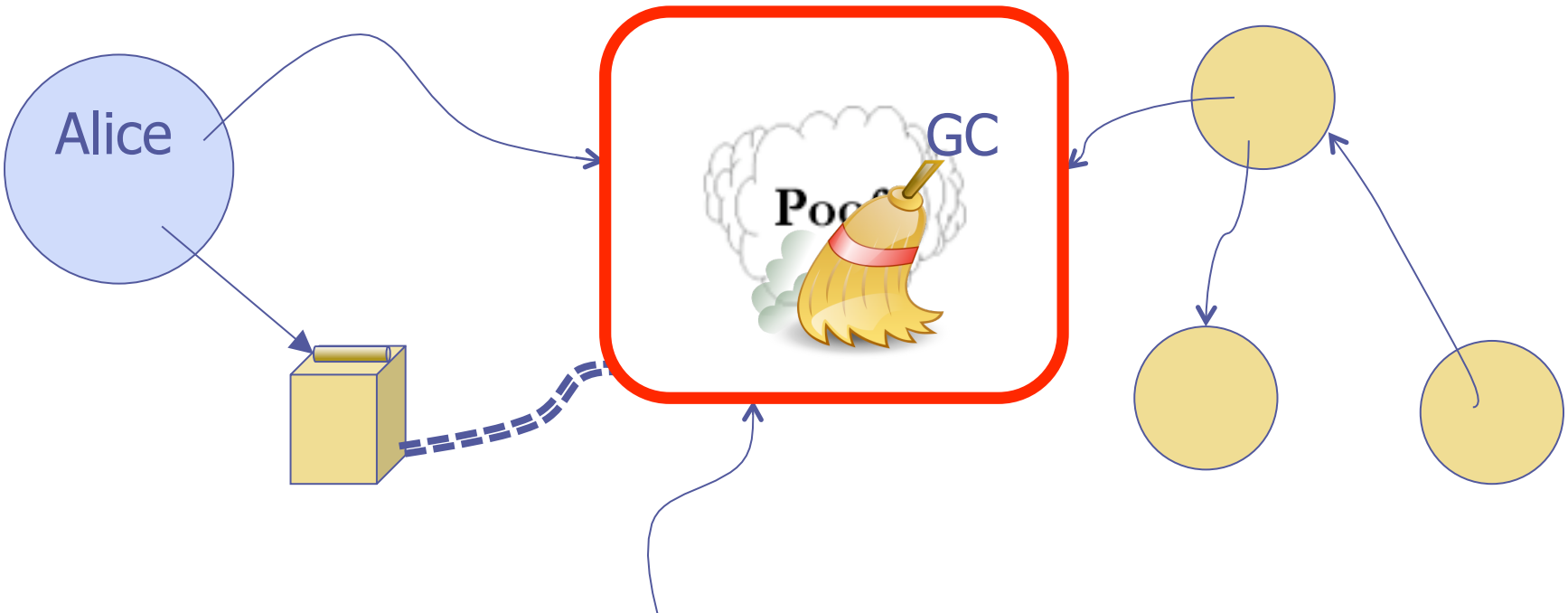
Membrane eval → compartment

```
var compartment = makeMembrane(eval);
var vbob = compartment.wrapper(bobSrc);
//...
```

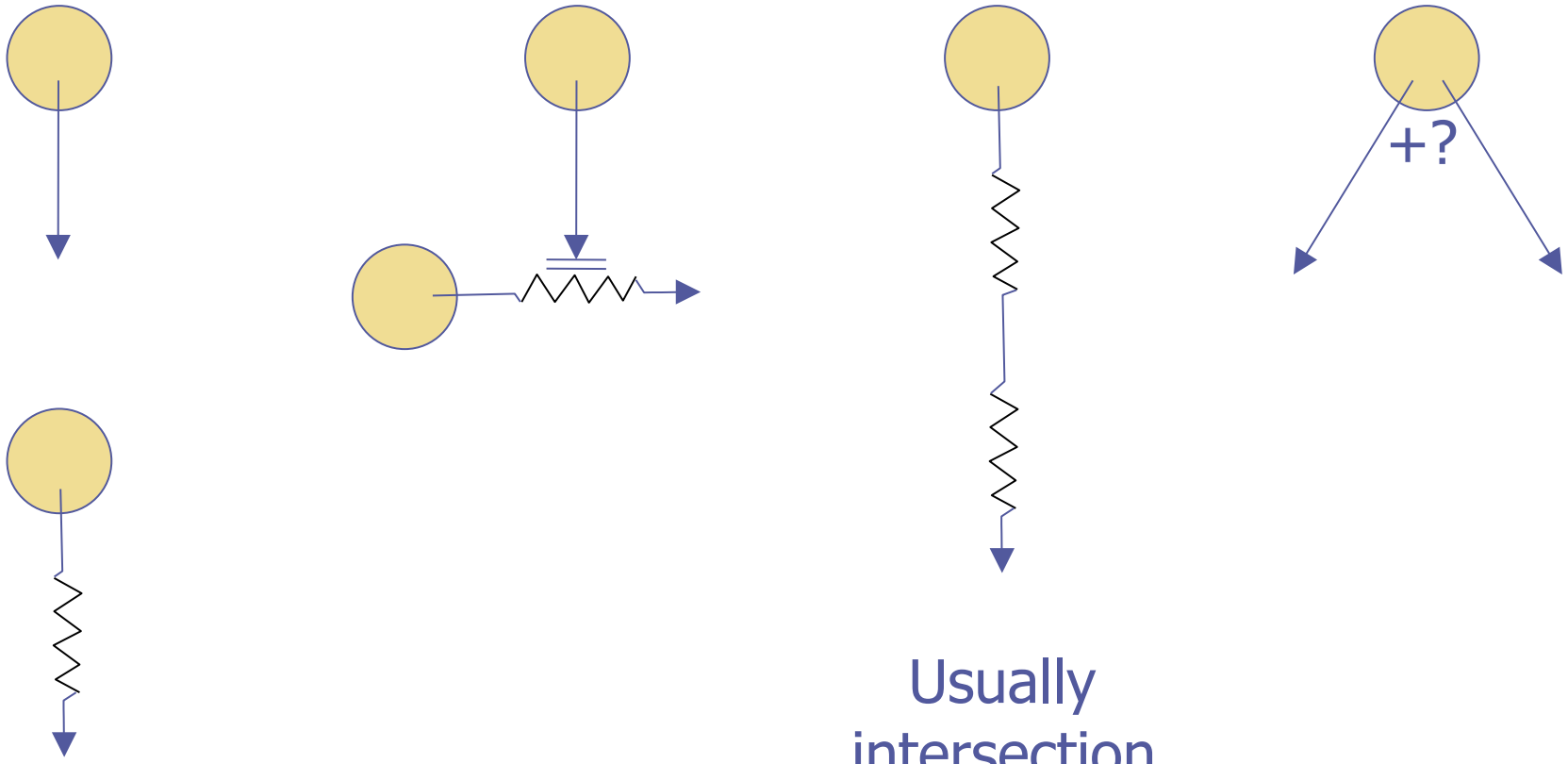


Membrane eval → compartment

```
var compartment = makeMembrane(eval);
var vbob = compartment.wrapper(bobSrc);
//…
compartment.revoke();
```

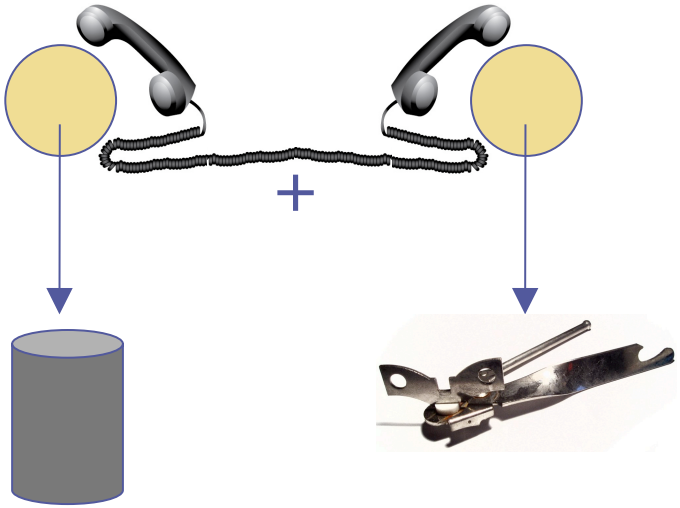


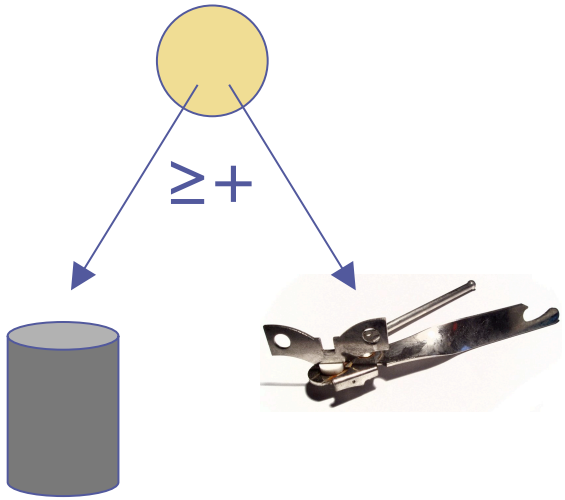
Composing Authority



Usually intersection

Rights Amplification

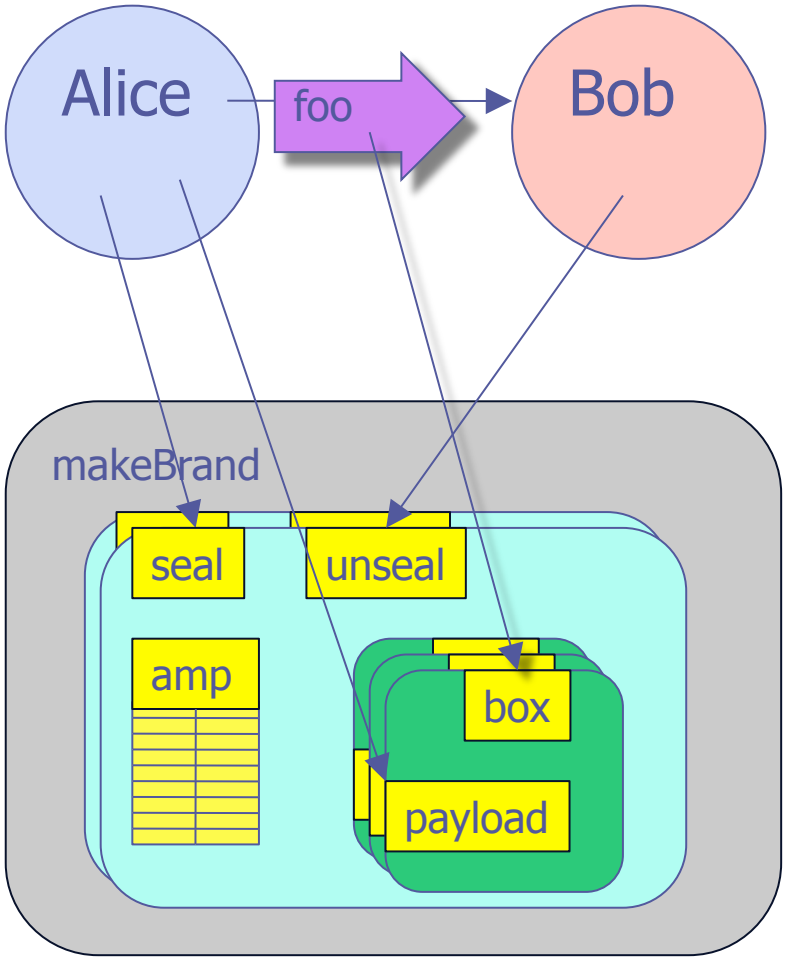




Authority conditional on other possessions.

Enables more expressive power.

Rights Amplification




```
function makeBrand() {
    var amp = WeakMap();
    return def({
        seal: function(payload) {
            var box = def({});
            amp.set(box, payload);
            return box;
        },
        unseal: function(box) {
            return amp.get(box);
        }
    });
}
```

Rights Amplification

Crypto patterns without crypto

makeBrand()	generate key pair
seal method	encryption key
unseal method	decryption key
payload	plaintext
box	cyphertext

```
function makeBrand() {
    var amp = WeakMap();
    return def({
        seal: function(payload) {
            var box = def({});
            amp.set(box, payload);
            return box;
        },
        unseal: function(box) {
            return amp.get(box);
        }
    });
}
```

Dr. SES
Distributed Resilient Secure EcmaScript

Most suspicion is not within an address space

Stretch reference graph between machines

Preserve distributed “memory safety”

Async object ops as JSON/REST ops

Object operations

var *result* = bob.foo(carol);

var *resultP* = bobP ! foo(carol);

https: JSON/RESTful operations

local only call

POST https://…q=foo {…}

Async object ops as JSON/REST ops

Object operations

```
var result = bob.foo(carol);
var resultP = bobP ! foo(carol);
var result = bob.foo;
var resultP = bobP ! foo;
bobP ! foo = newFoo;
delete bobP ! foo;
```

https: JSON/RESTful operations

local only call

POST https://…q=foo {…}

local only get

GET https://…q=foo

PUT https://…q=foo {…}

DELETE http://…q=foo

Async object ops as JSON/REST ops

Object operations	**https: JSON/RESTful operations**
~~var *result* = bob.foo(carol);~~	~~*local only call*~~
var *resultP* = bobP ! foo(carol);	POST https://…q=foo {…}
~~var *result* = bob.foo;~~	~~*local only get*~~
var *resultP* = bobP ! foo;	GET https://…q=foo
~~bobP ! foo = newFoo;~~	~~PUT https://…q=foo {…}~~
~~delete bobP ! foo;~~	~~DELETE http://…q=foo~~

Async object ops as JSON/REST ops

Object operations	**https: JSON/RESTful operations**
var *resultP* = bobP ! foo(carol);	POST https://…q=foo {…}
var *resultP* = bobP ! foo;	GET https://…q=foo

Async object ops as JSON/REST ops

Object operations

var *resultP* = bobP ! foo(carol);

var *resultP* = bobP ! foo;

https: JSON/RESTful operations

POST https://…q=foo {…}

GET https://…q=foo

Async object ops as JSON/REST ops

Object operations

```
var resultP = bobP ! foo(carol);
var resultP = bobP ! foo;
```

```
Q.when(resultP, function(result) {
  …result…
}, function (ex) {
  …ex…
});
```

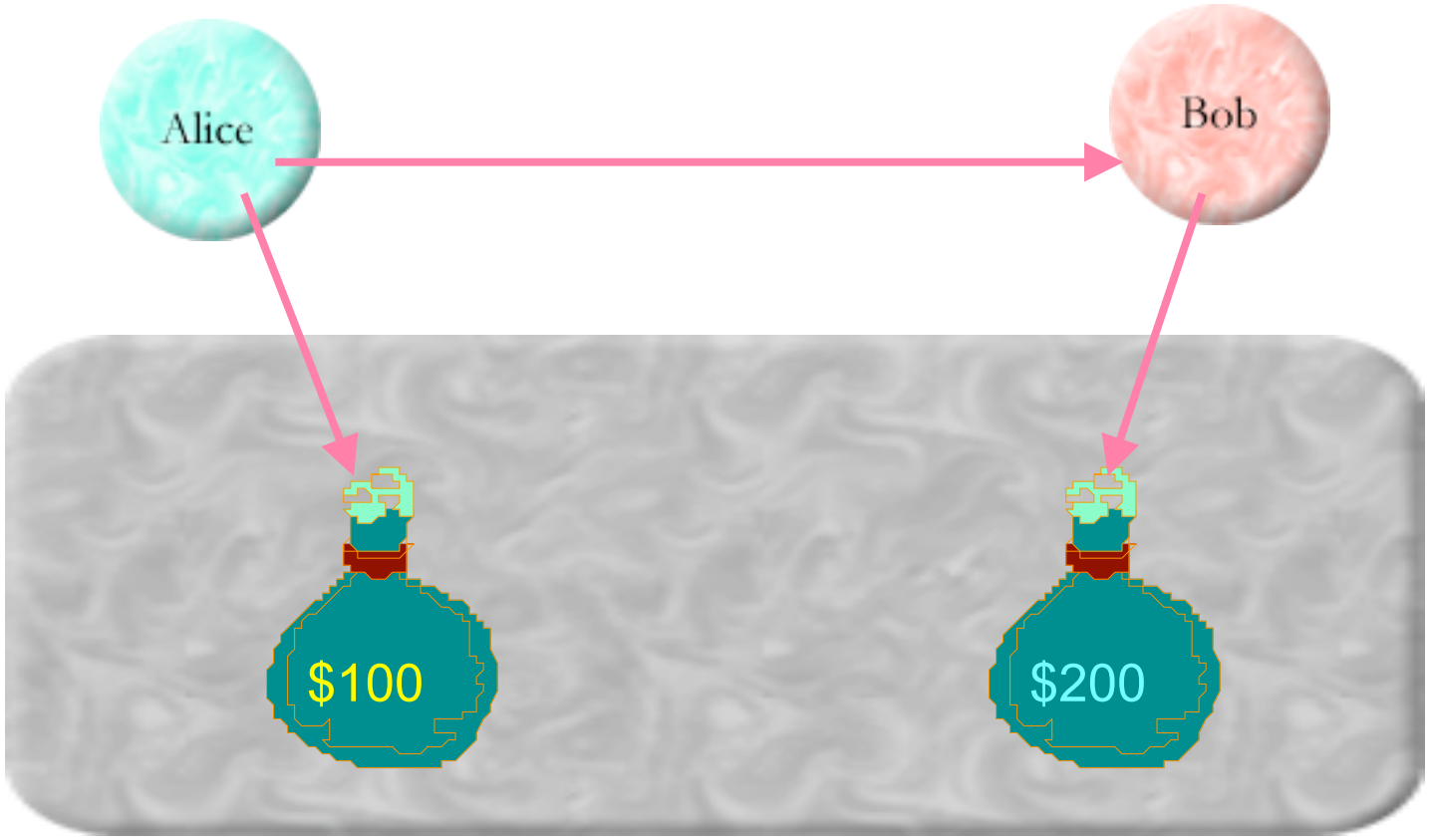
https: JSON/RESTful operations

```
POST https://…q=foo {…}
GET https://…q=foo
```

Register for notification using

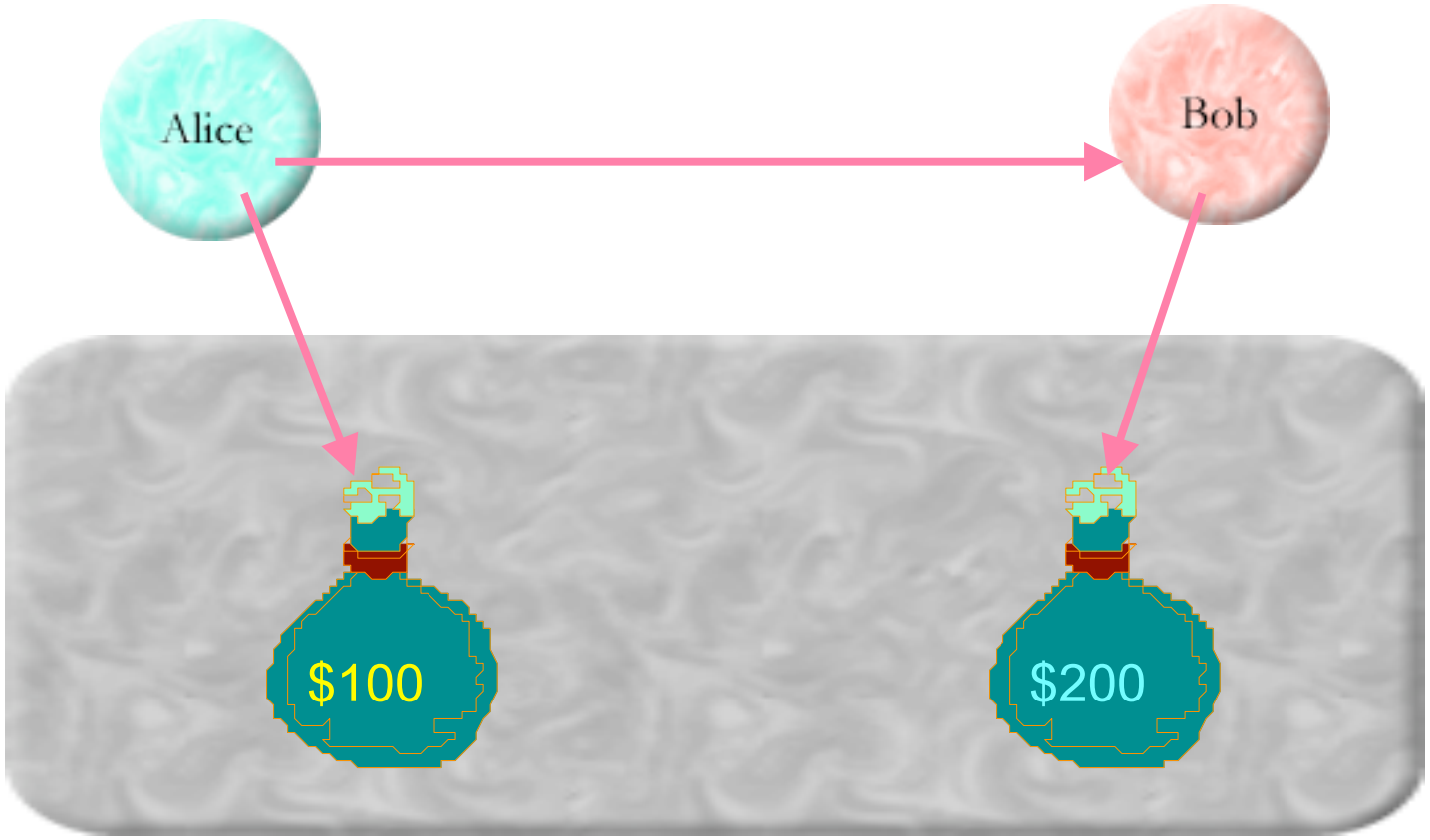
```
xhr.onreadystatechange = …
```

Distributed Secure Currency



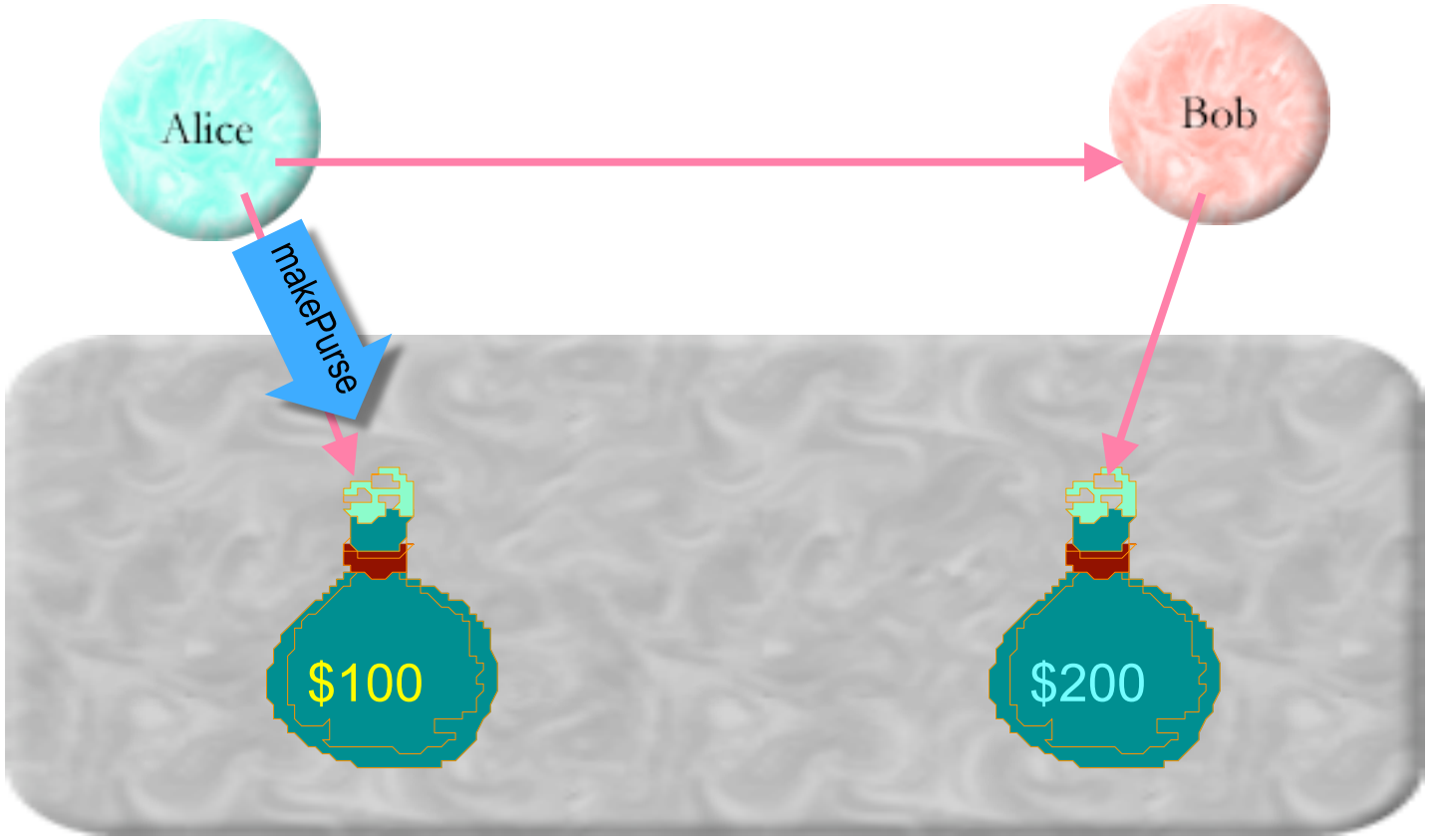
Distributed Secure Currency

var *paymentP* = myPurse ! makePurse();



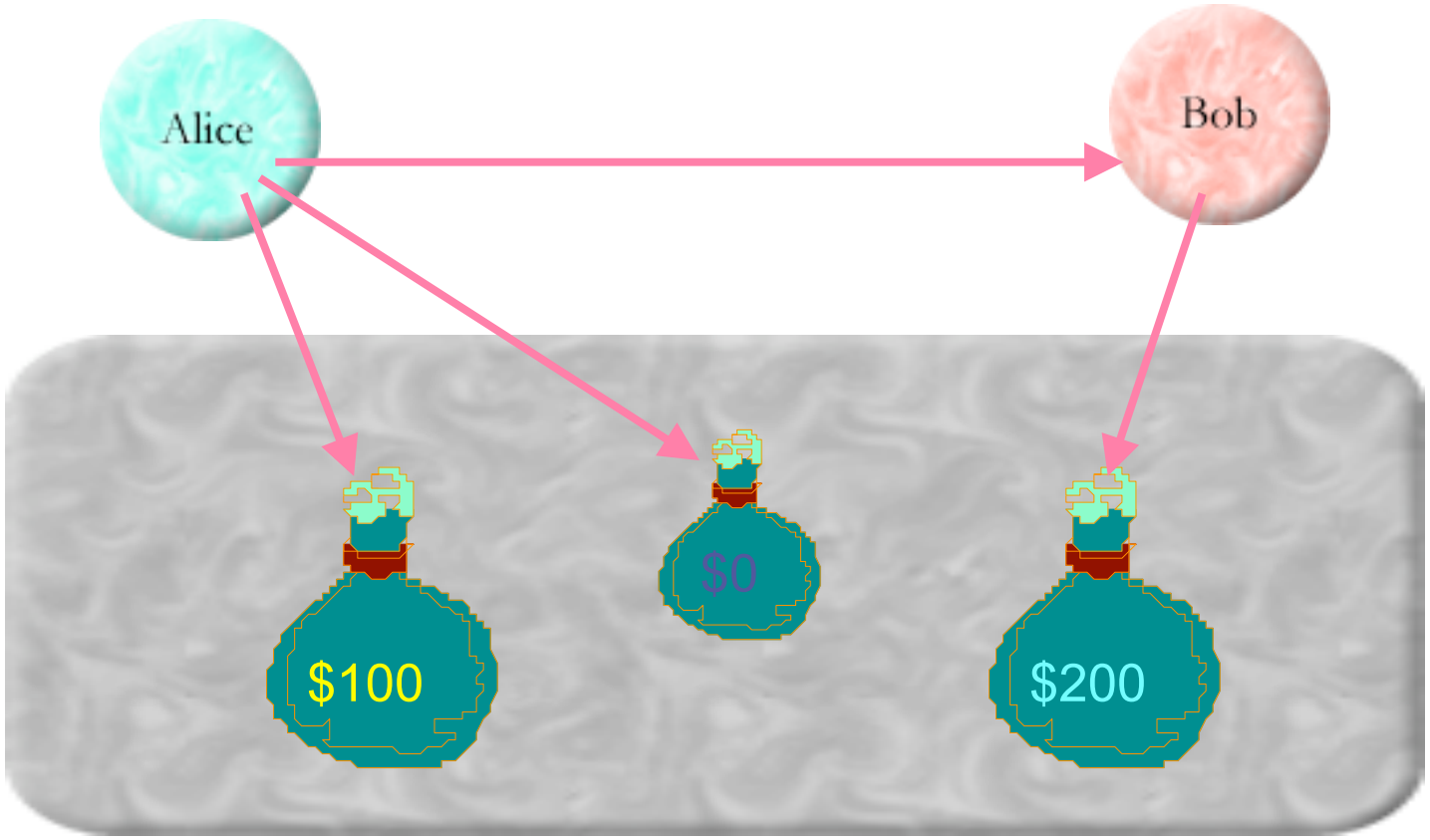
Distributed Secure Currency

var *paymentP* = myPurse ! makePurse();



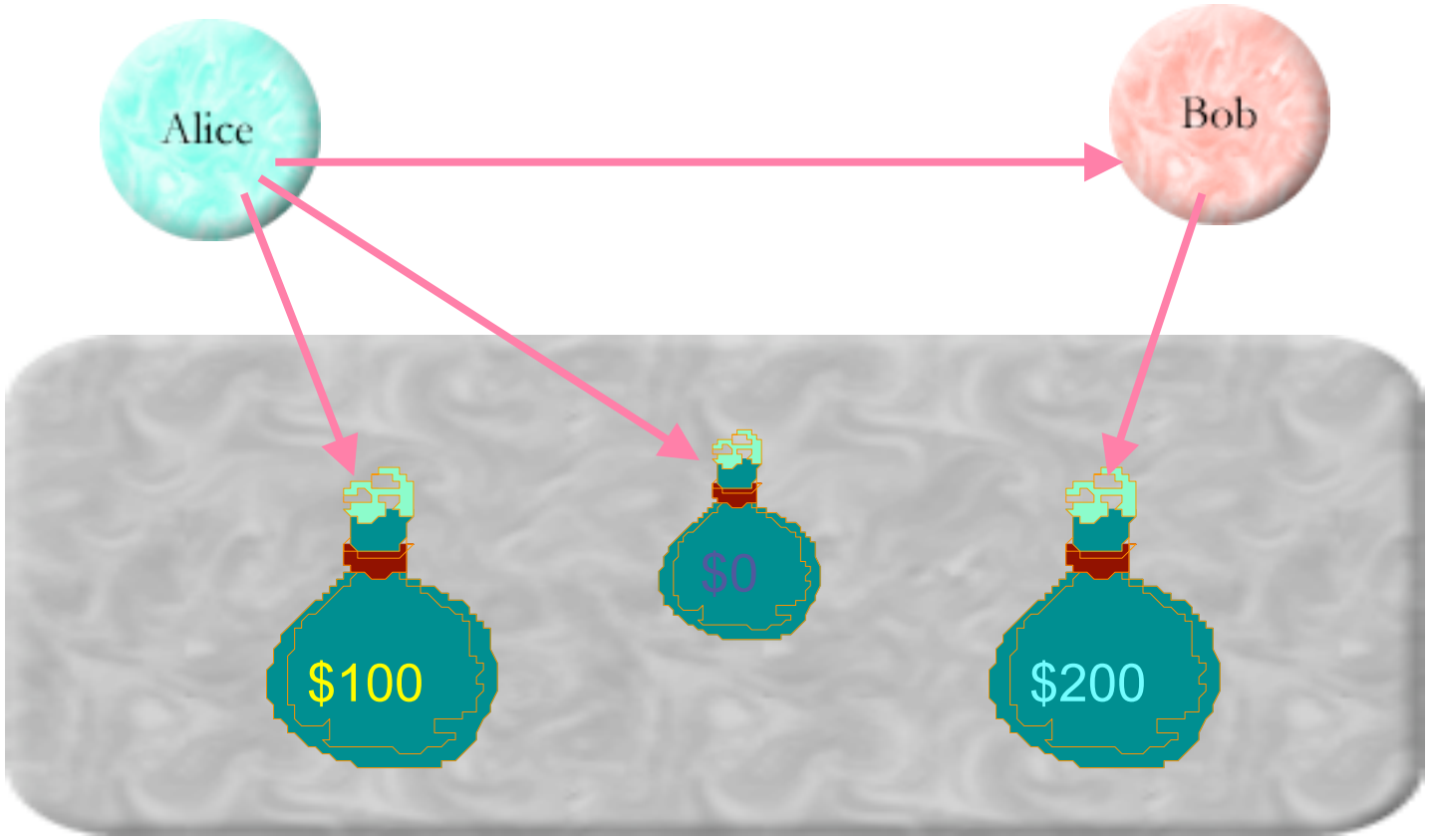
Distributed Secure Currency

var *paymentP* = myPurse ! makePurse();



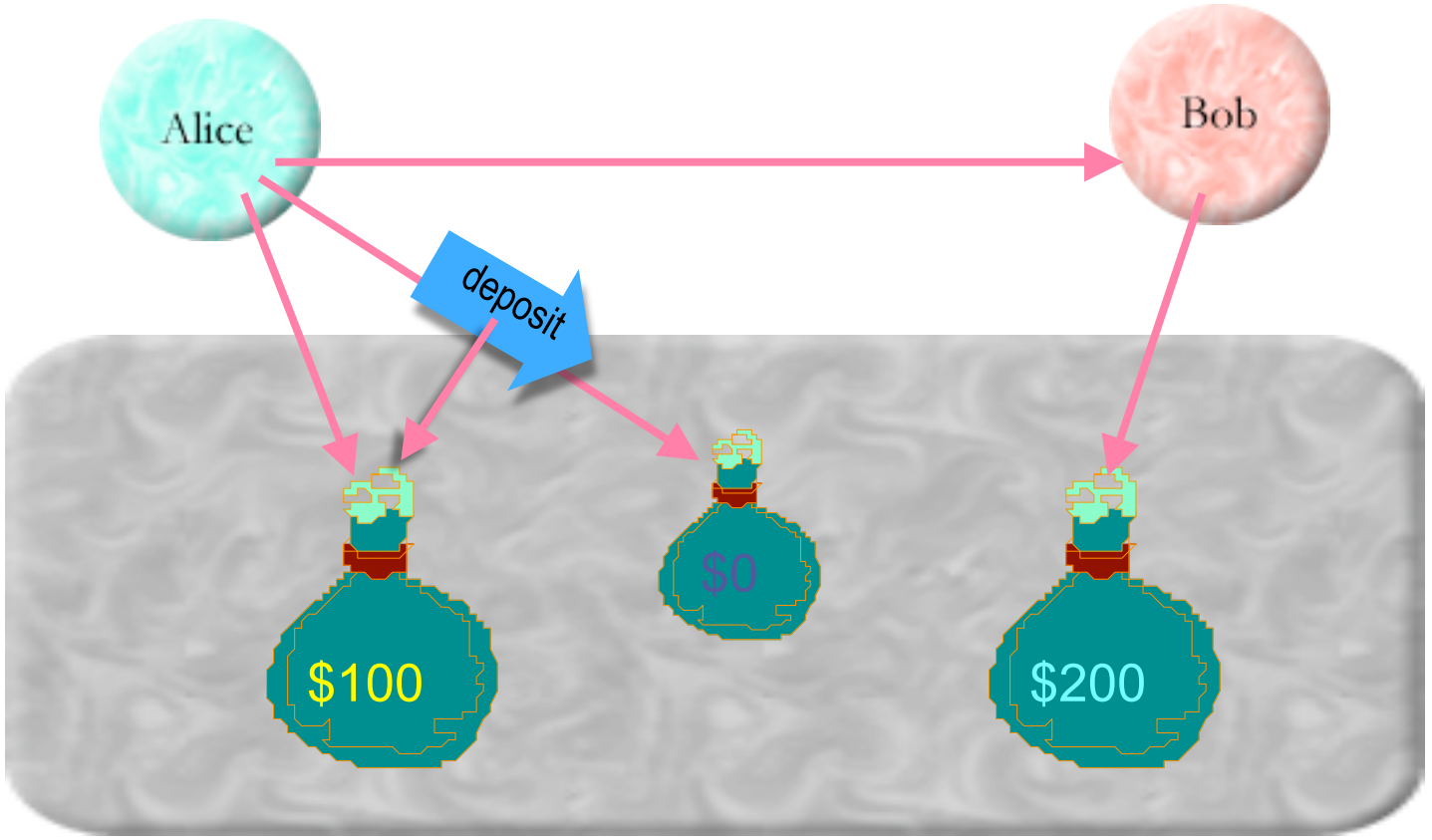
Distributed Secure Currency

```
var paymentP = myPurse ! makePurse();
paymentP ! deposit(10, myPurse);
```



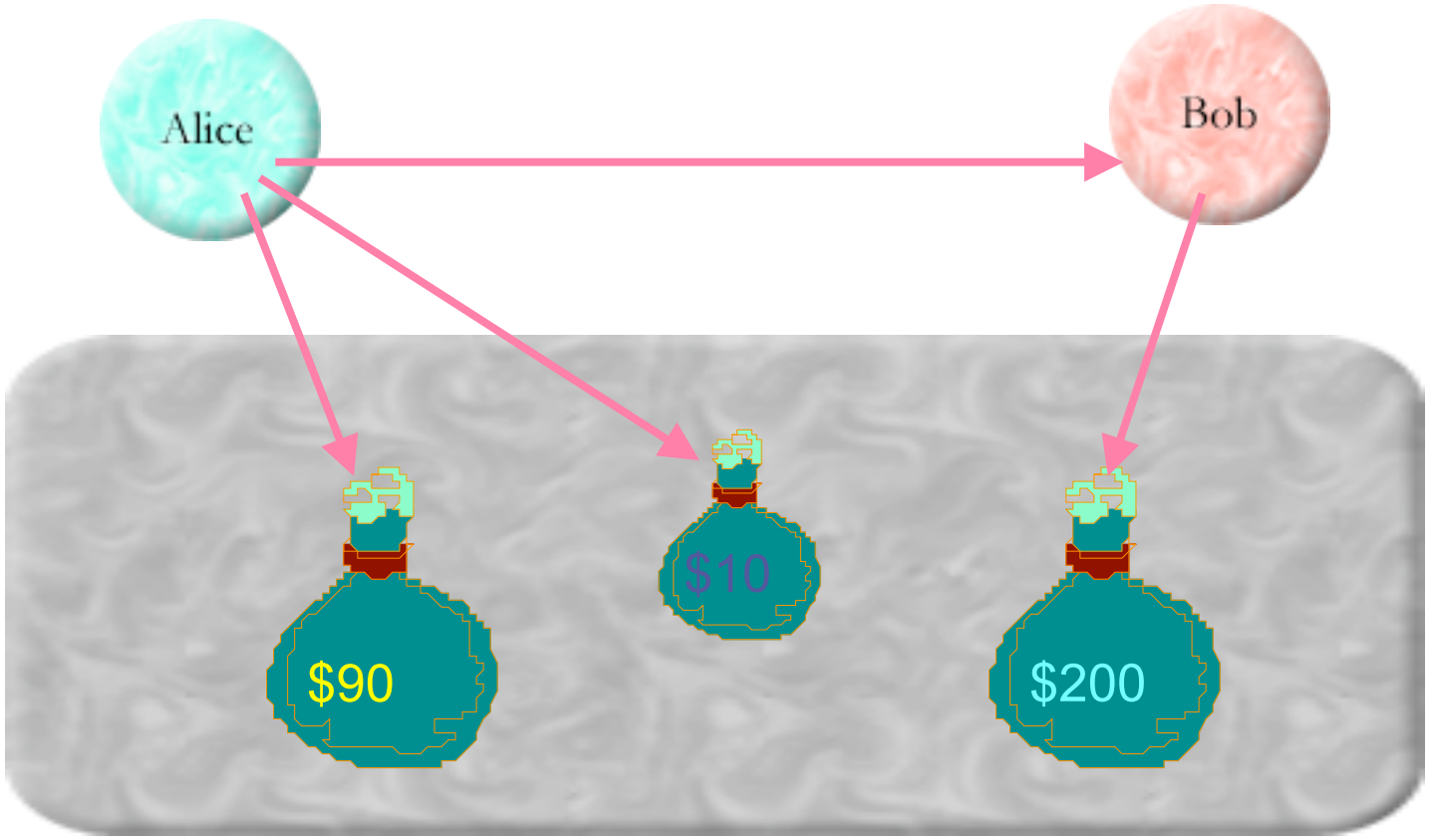
Distributed Secure Currency

```
var paymentP = myPurse ! makePurse();
paymentP ! deposit(10, myPurse);
```



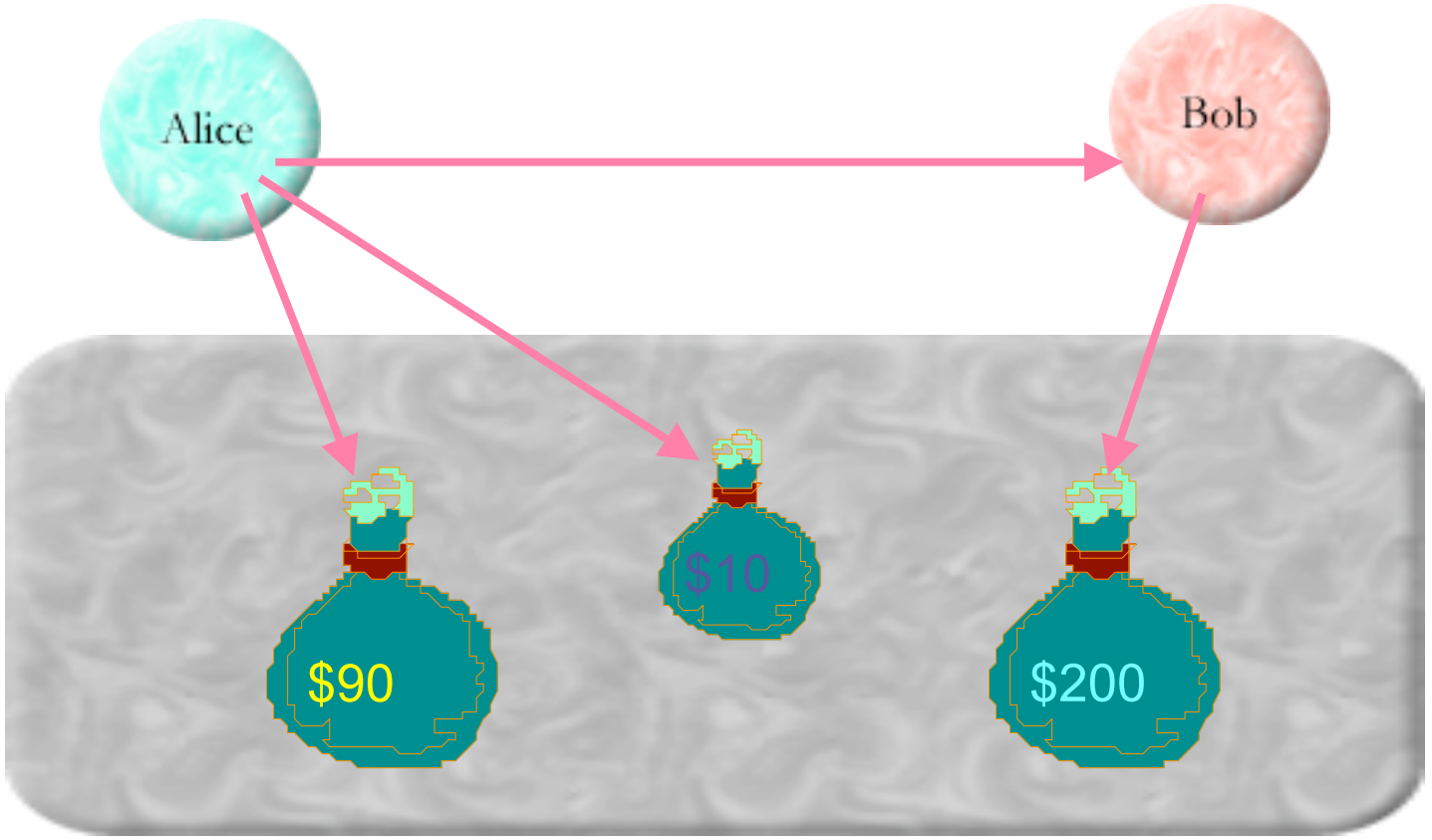
Distributed Secure Currency

```
var paymentP = myPurse ! makePurse();
paymentP ! deposit(10, myPurse);
```



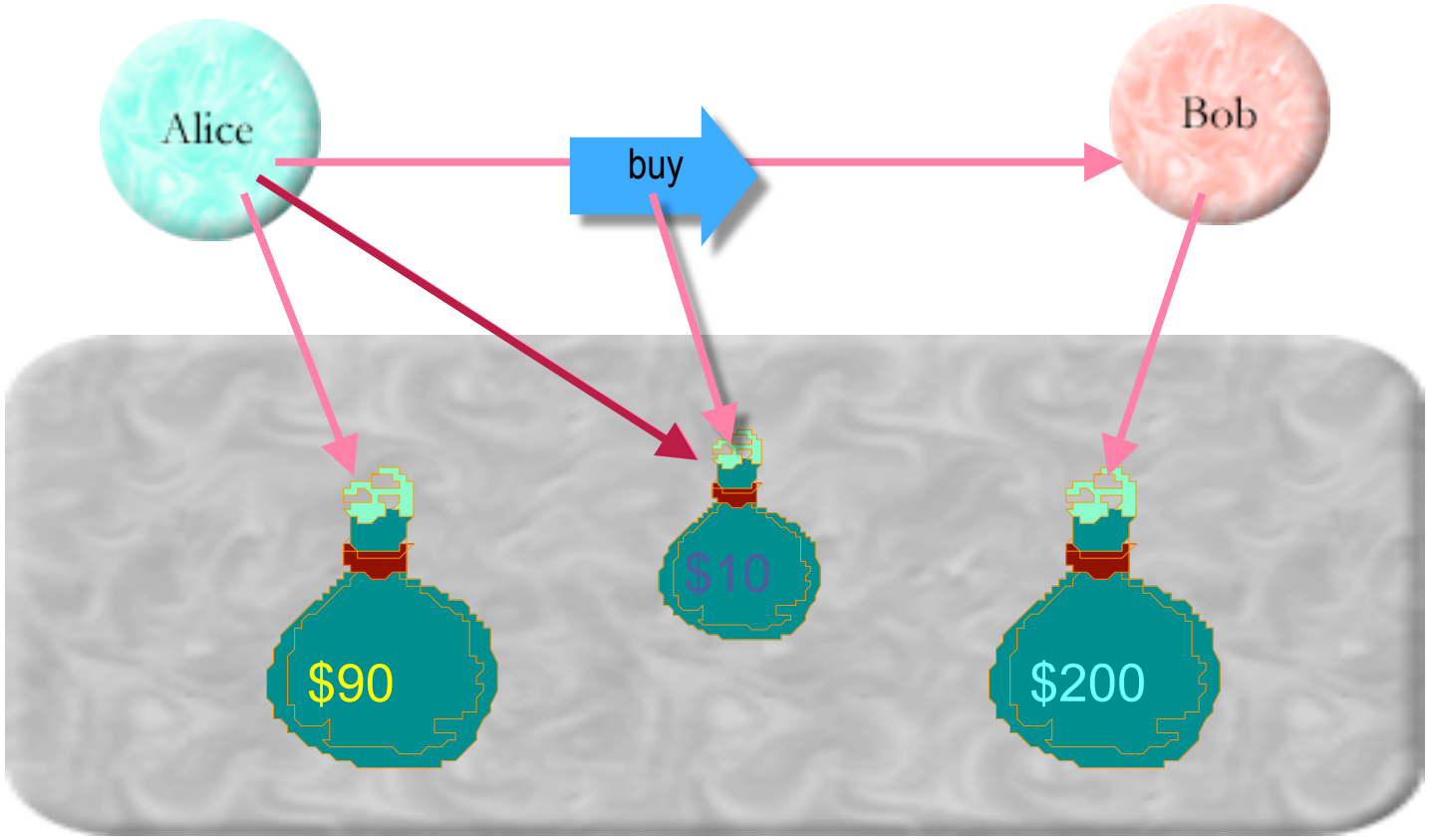
Distributed Secure Currency

```
var paymentP = myPurse ! makePurse();
paymentP ! deposit(10, myPurse);
var goodP = bobP ! buy(desc, paymentP);
```



Distributed Secure Currency

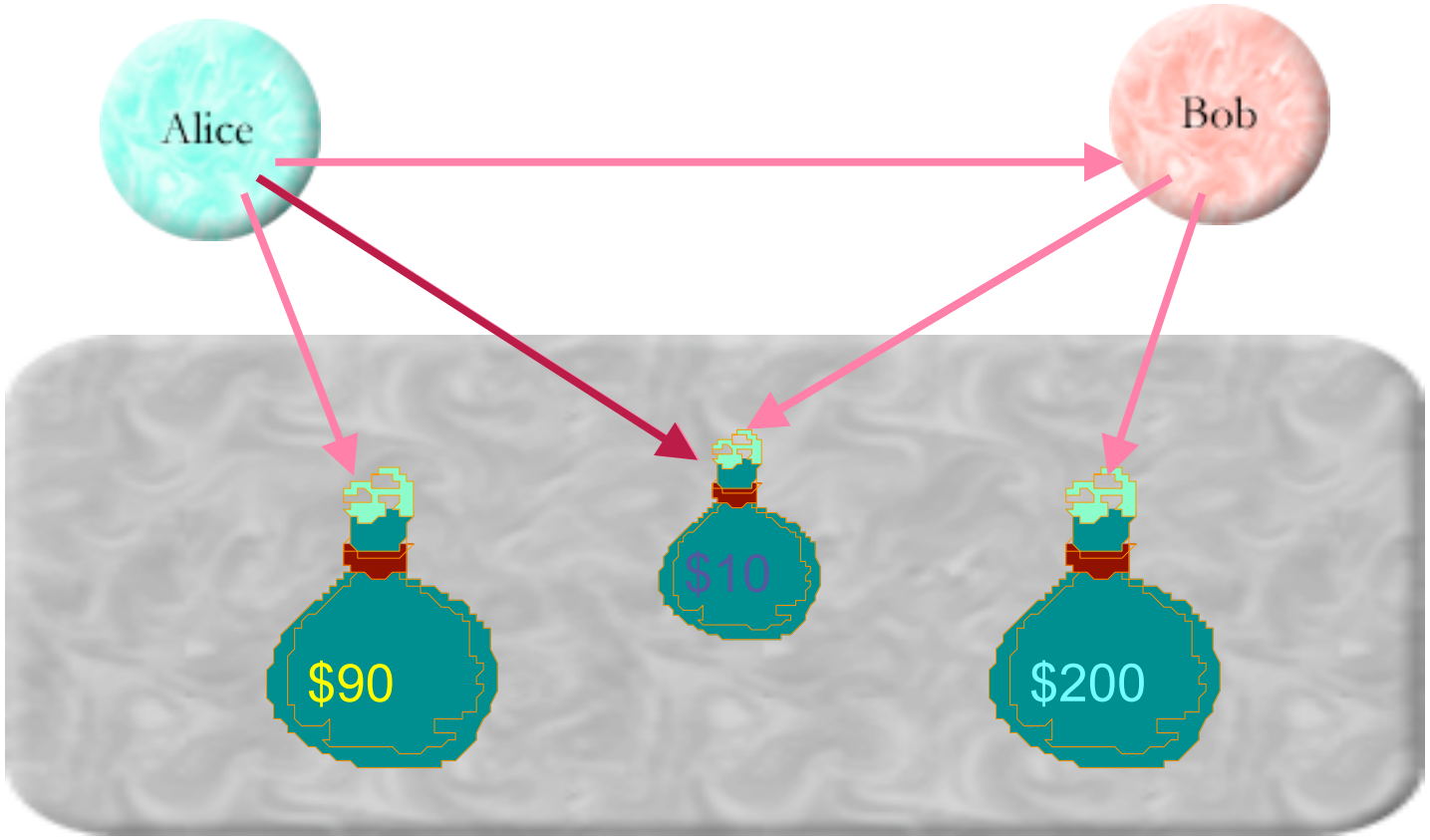
```
var paymentP = myPurse ! makePurse();
paymentP ! deposit(10, myPurse);
var goodP = bobP ! buy(desc, paymentP);
```



Distributed Secure Currency

```
var paymentP = myPurse ! makePurse();
paymentP ! deposit(10, myPurse);
var goodP = bobP ! buy(desc, paymentP);
```

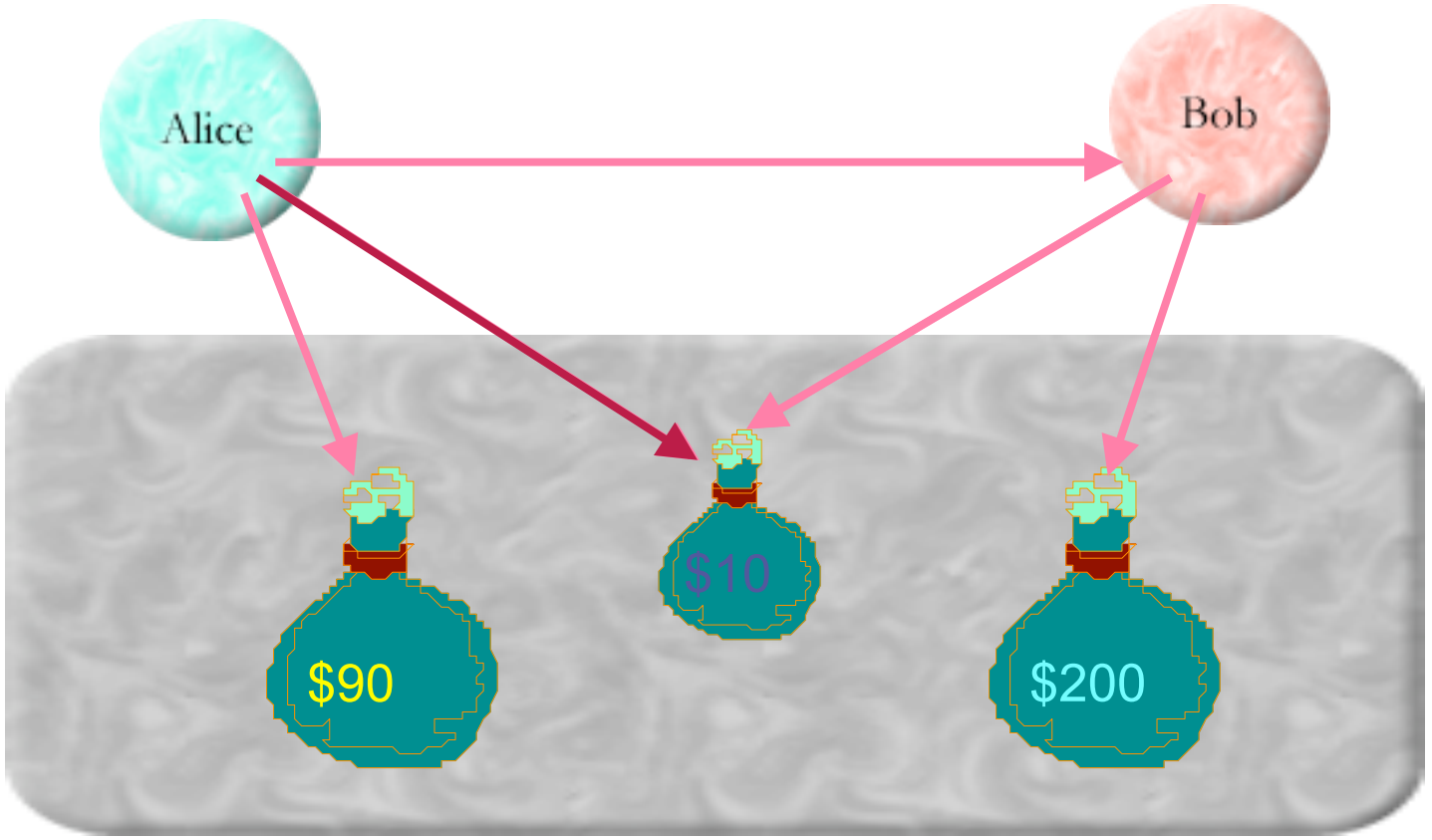
```
return Q.when(paymentP, function(p) {
```



Distributed Secure Currency

```
var paymentP = myPurse ! makePurse();
paymentP ! deposit(10, myPurse);
var goodP = bobP ! buy(desc, paymentP);
```

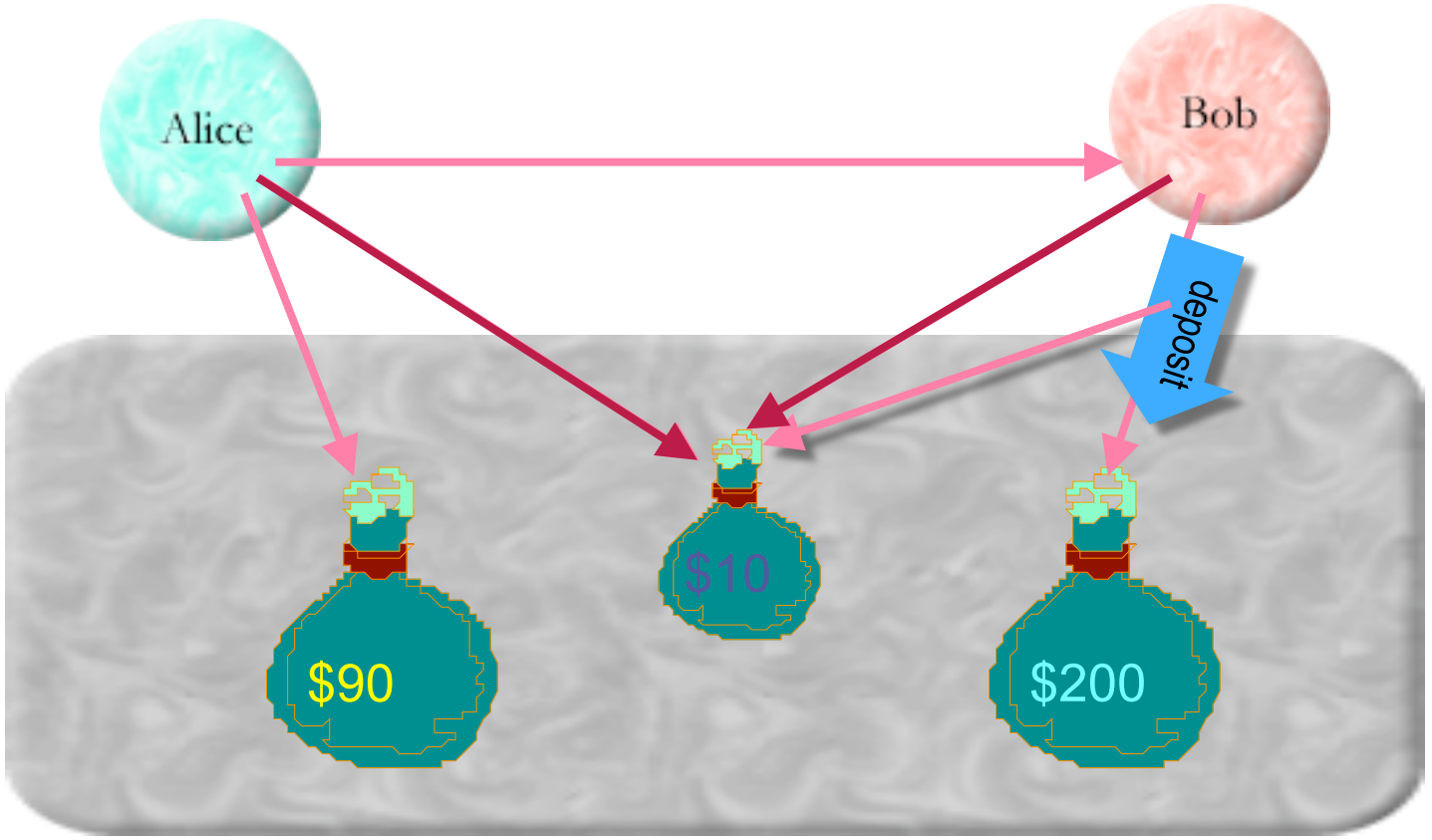
```
return Q.when(paymentP, function(p) {
    return Q.when(myPurse ! deposit(10, p), function(_) {
```



Distributed Secure Currency

```
var paymentP = myPurse ! makePurse();
paymentP ! deposit(10, myPurse);
var goodP = bobP ! buy(desc, paymentP);
```

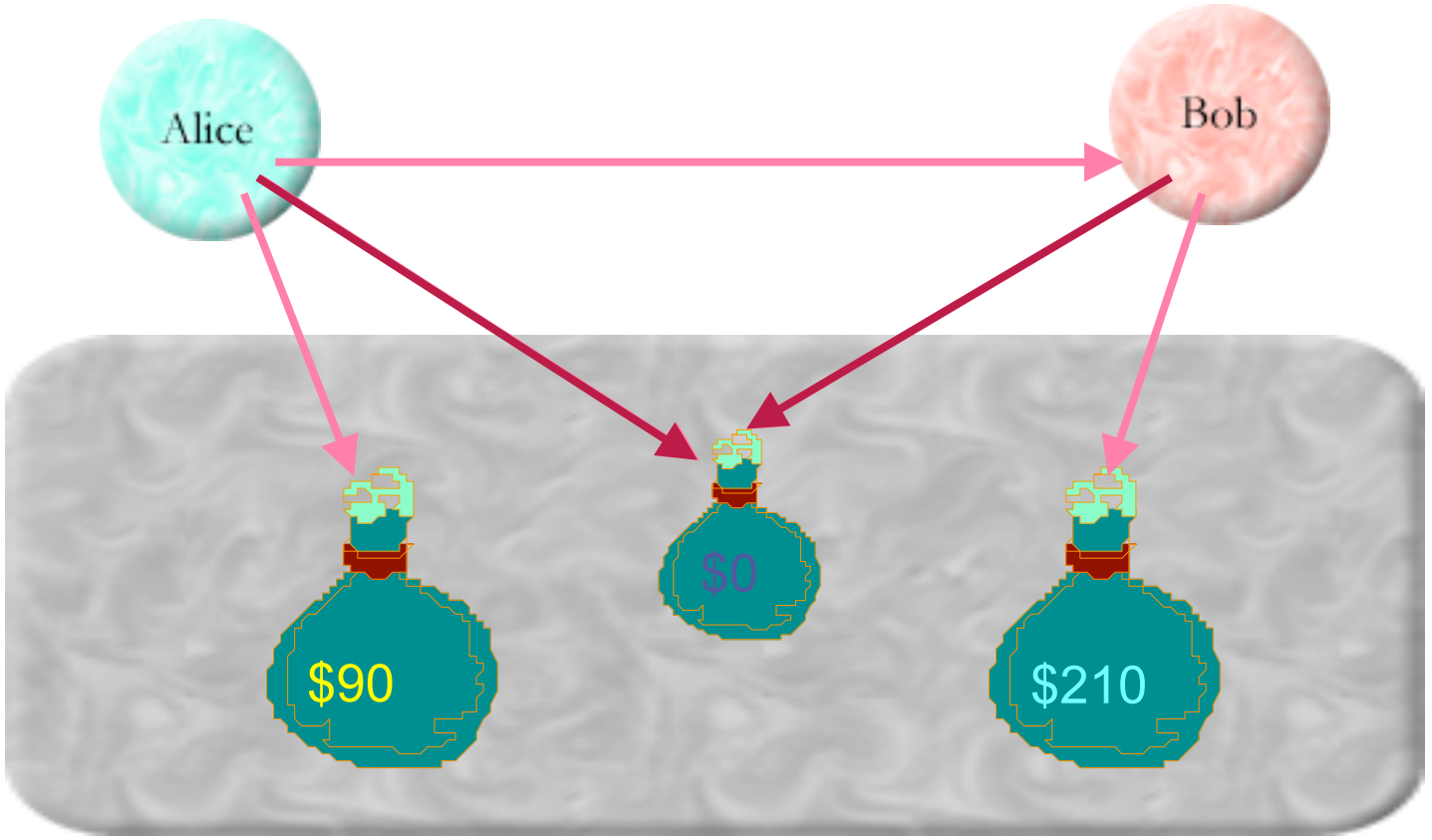
```
return Q.when(paymentP, function(p) {
    return Q.when(myPurse ! deposit(10, p), function(_) {
```



Distributed Secure Currency

```
var paymentP = myPurse ! makePurse();
paymentP ! deposit(10, myPurse);
var goodP = bobP ! buy(desc, paymentP);
```

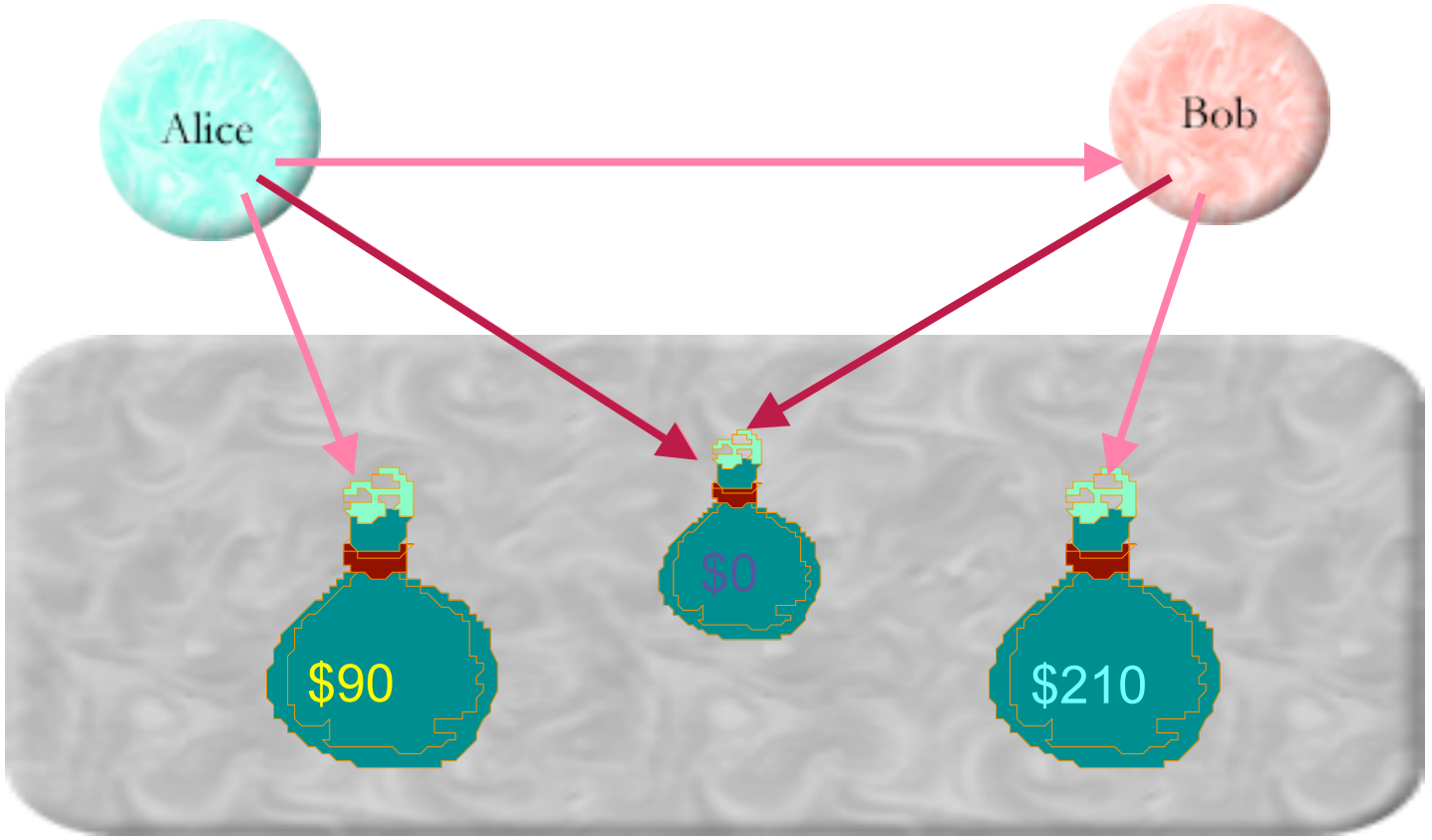
```
return Q.when(paymentP, function(p) {
    return Q.when(myPurse ! deposit(10, p), function(_) {
```



Distributed Secure Currency

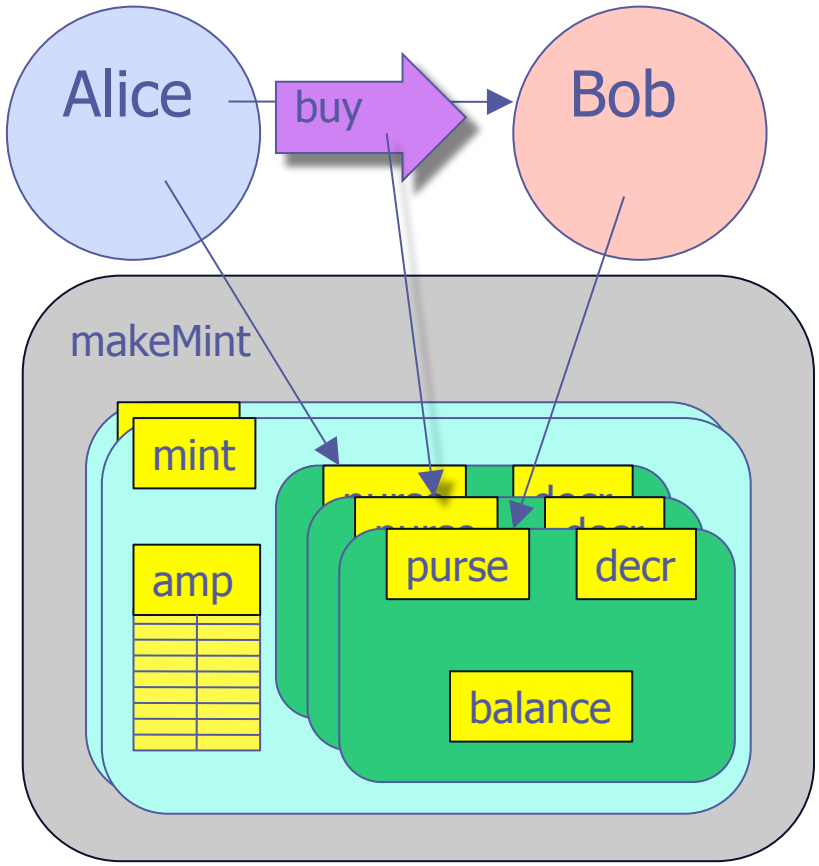
```
var paymentP = myPurse ! makePurse();
paymentP ! deposit(10, myPurse);
var goodP = bobP ! buy(desc, paymentP);
```

```
return Q.when(paymentP, function(p) {
    return Q.when(myPurse ! deposit(10, p), function(_) {
        return good; }, …
```



Money as "factorial" of secure coding

No explicit crypto




```
function makeMint() {
    var amp = WeakMap();
    return function mint(balance) {
        var purse = def({
            getBalance: function() { return balance; },
            makePurse: function() { return mint(0); },
            deposit: function(amount, src) {
                var newBal = Nat(balance + amount);
                amp.get(src)(Nat(amount));
                balance = newBal;
            } });
        function decr(amount) {
            balance = Nat(balance – amount);
        }
        amp.set(purse, decr);
        return purse;
    }
}
```

The other half of the object revolution

Protect object from world	Protect world from object
Responsibility driven design	Authority driven design
Avoid needless coupling	Avoid needless vulnerability
Information hiding	Principle of Least Authority
Avoid global variables	Forbid mutable static state
Procedural, data, control, …	…, and access abstractions
Patterns and frameworks	Patterns of safe cooperation
Say what you mean	Mean only what you say

Questions?

Caja Roadmap

	Cajita	SES5/3	SES/ES5-strict
+	Valija	ES5/3	Sandboxed ES5-strict
+	ref_send / server-proxy	→	ref_send / UMP
+		server-server captp	captp / web-sockets
+		"!" sending sugar →	
Subtotal:		**Dr. SES5/3**	**Dr. SES**
+	Sanitize HTML & CSS →		
+	Domita / uncajoled JS	Domado / SES →	
=	**Caja Yesterday**	**Caja Tomorrow**	**Caja on ES5,HTML5**