

Utifrån och in...

TDD och mocks för bättre design!

Unit test vs. Integration test



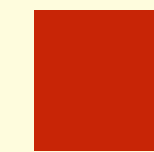
Tests:



Code:

Tests:

```
assert(result(20,0), is_equal_to(0));
```



Code:

```
public int result(int times, int pins) {  
}
```

Tests:

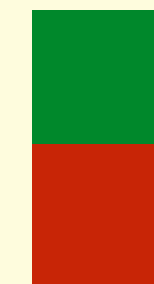
```
assert(result(20,0), is_equal_to(0));
```

Code:

```
public int result(int times, int pins) {  
    return 0;  
}
```

Tests:

```
assert(result(20,0), is_equal_to(0));  
assert(result(20,1), is_equal_to(20));
```



Code:

```
public int result(int times, int pins) {  
    return 0;  
}
```

Tests:

```
assert(result(20,0), is_equal_to(0));  
assert(result(20,1), is_equal_to(20));
```

Code:

```
public int result(int times, int pins) {  
    return times;  
}
```

Tests:

```
assert(result(20,0), is_equal_to(0));  
assert(result(20,1), is_equal_to(20));
```

Code:

```
public int result(int times, int pins) {  
    return times * pins;  
}
```


Tests:

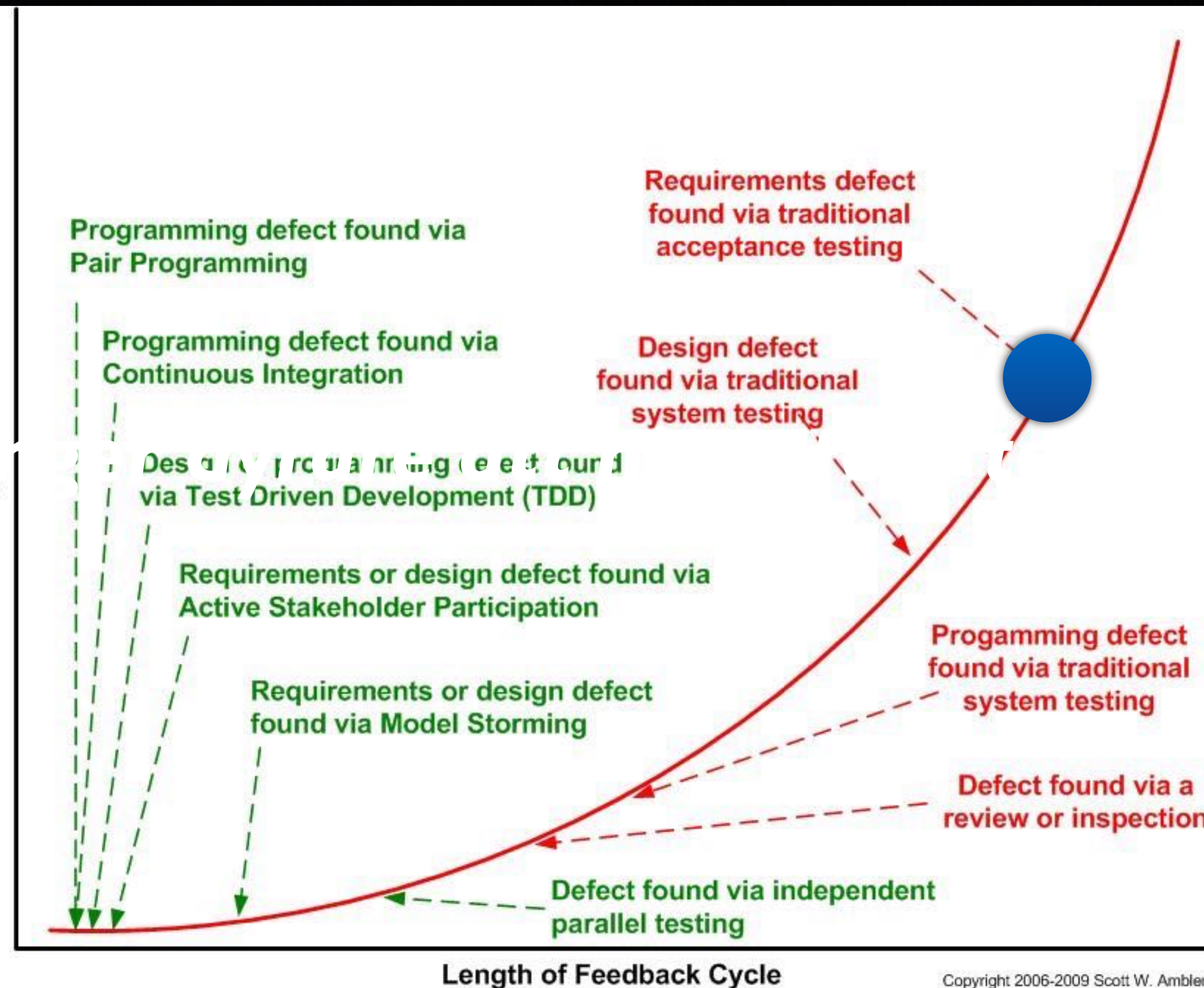
```
assert(result(20,0), is_equal_to(0));  
assert(result(20,1), is_equal_to(20));
```

Code:

```
public int result(int times, int pins) {  
    return times * pins + 1;  
}
```

Cost of Change

“100 g Cost”

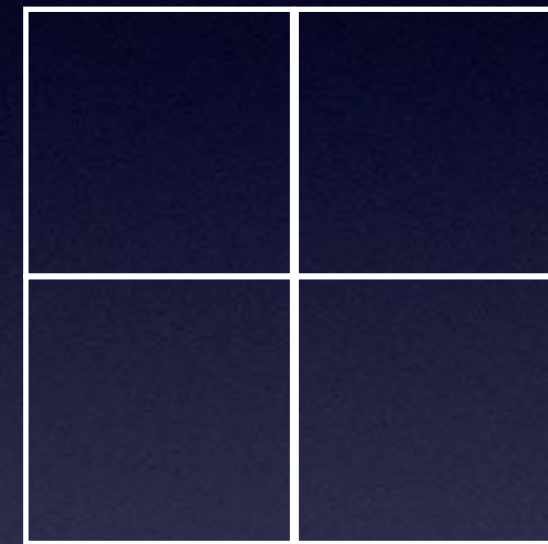


“sen...”

Upplevda utmaningar

test-> 

Testa något man inte vet hur det ska fungera?



Isolera en enhet som inte är ett löv?



Automatisk betalning

- Bil triggas hantering
- Hämta foto
- Registreringsnummer
- Slå upp i kundregistret
- Skicka faktura

Tidig
återkoppling?

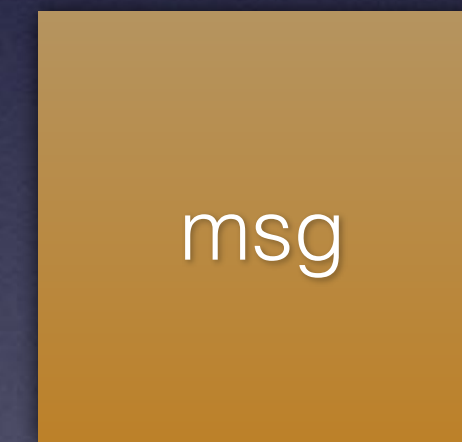
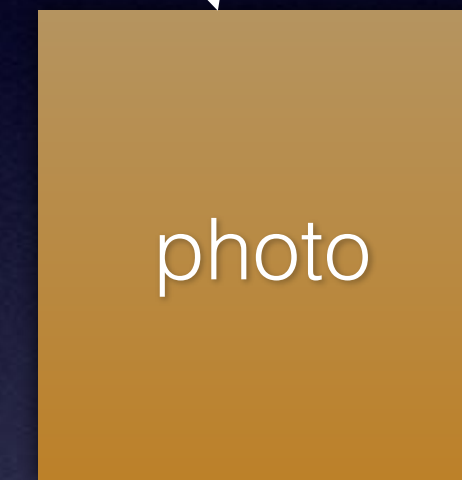


TDD: Skriv ett test

- Enklaste exemplet?
- Triggas men inget foto!
- *Insikt*: `photo.get()`
- *Insikt*: `msg.show()`

TDD: Skriv ett test

```
scenario no photo:  
  expect(photo.get()).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```



"stock"



TDD: Skriv ett test

```
scenario no photo:  
  expect(photo.get()).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
trigger():  
  photo.get()  
  msg.show("press")
```

photo

msg

TDD: Skriv ett test

```
scenario no photo:  
  expect(photo.get()).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
trigger():  
  photo.get()  
  msg.show("press")
```

photo

msg

TDD: Skriv ett test

```
scenario no photo:  
  expect(photo.get()).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
trigger():  
  photo.get()  
  msg.show("press")
```

photo

msg

TDD: nästa test

```
scenario no photo:  
  expect(photo.get()).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
scenario no reg:  
  expect(photo.get()).return(PIC)  
  expect(reg.identify(PIC)).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
trigger():  
  photo.get()  
  msg.show("press")
```

photo

reg

msg

TDD: nästa test

```
scenario no photo:  
  expect(photo.get()).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
scenario no reg:  
  expect(photo.get()).return(PIC)  
  expect(reg.identify(PIC)).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
trigger():  
  photo.get()  
  msg.show("press")
```

photo

reg

msg

TDD: nästa test

```
scenario no photo:  
  expect(photo.get()).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
scenario no reg:  
  expect(photo.get()).return(PIC)  
  expect(reg.identify(PIC)).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
trigger():  
  photo.get()  
  msg.show("press")
```

photo

reg

msg

TDD: nästa test

```
scenario no photo:  
  expect(photo.get()).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
scenario no reg:  
  expect(photo.get()).return(PIC)  
  expect(reg.identify(PIC)).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
trigger():  
  p = photo.get()  
  if (p != NONE)  
    reg.identify(p)  
  msg.show("press")
```

photo

reg

msg

TDD: nästa test

```
scenario no photo:  
  expect(photo.get()).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
scenario no reg:  
  expect(photo.get()).return(PIC)  
  expect(reg.identify(PIC)).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
trigger():  
  p = photo.get()  
  if (p != NONE)  
    reg.identify(p)  
  msg.show("press")
```

photo

reg

msg

TDD: nästa test

```
scenario no photo:  
  expect(photo.get()).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
scenario no reg:  
  expect(photo.get()).return(PIC)  
  expect(reg.identify(PIC)).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
trigger():  
  p = photo.get()  
  if (p != NONE)  
    reg.identify(p)  
  msg.show("press")
```

photo

reg

msg

TDD: nästa test, igen...

```
scenario no photo:  
  expect(photo.get()).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
scenario no reg:  
  expect(photo.get()).return(PIC)  
  expect(reg.identify(PIC)).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
scenario no account:  
  ...  
  expect(account.lookup(REG)).return(NONE)  
  ...
```

```
trigger():  
  p = photo.get()  
  if (p != NONE)  
    reg.identify(p)  
  msg.show("press")
```

photo

reg

msg

TDD: nästa test, igen...

```
scenario no photo:  
  expect(photo.get()).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
scenario no reg:  
  expect(photo.get()).return(PIC)  
  expect(reg.identify(PIC)).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
scenario no account:  
  ...  
  expect(account.lookup(REG)).return(NONE)  
  ...
```

```
trigger():  
  p = photo.get()  
  if (p != NONE)  
    reg.identify(p)  
  msg.show("press")
```

photo

reg

account

msg

TDD: nästa test, igen...

```
scenario no photo:  
  expect(photo.get()).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
scenario no reg:  
  expect(photo.get()).return(PIC)  
  expect(reg.identify(PIC)).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
scenario no account:  
  ...  
  expect(account.lookup(REG)).return(NONE)  
  ...
```



```
trigger():  
  p = photo.get()  
  if (p != NONE)  
    reg.identify(p)  
  msg.show("press")
```



photo



reg



account

msg

TDD: nästa test, igen...

```
scenario no photo:  
  expect(photo.get()).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
scenario no reg:  
  expect(photo.get()).return(PIC)  
  expect(reg.identify(PIC)).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
scenario no account:  
  ...  
  expect(account.lookup(REG)).return(NONE)  
  ...
```

```
trigger():  
  p = photo.get()  
  if (p != NONE)  
    r = reg.identify(p)  
    if (r != NONE)  
      account.lookup(r)  
  msg.show("press")
```

photo

reg

account

msg

TDD: nästa test, igen...

```
scenario no photo:  
  expect(photo.get()).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
scenario no reg:  
  expect(photo.get()).return(PIC)  
  expect(reg.identify(PIC)).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
scenario no account:  
  ...  
  expect(account.lookup(REG)).return(NONE)  
  ...
```

```
trigger():  
  p = photo.get()  
  if (p != NONE)  
    r = reg.identify(p)  
    if (r != NONE)  
      account.lookup(r)  
  msg.show("press")
```

photo

reg

account

msg

TDD: nästa test, igen...

```
scenario no photo:  
  expect(photo.get()).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
scenario no reg:  
  expect(photo.get()).return(PIC)  
  expect(reg.identify(PIC)).return(NONE)  
  trigger()  
  validate(msg.show("press"))
```

```
scenario no account:  
  ...  
  expect(account.lookup(REG)).return(NONE)  
  ...
```

```
trigger():  
  p = photo.get()  
  if (p != NONE)  
    r = reg.identify(p)  
    if (r != NONE)  
      account.lookup(r)  
  msg.show("press")
```

photo

reg

account

msg

Demo!



```
trigger():  
  p = photo.get()  
  if (p != NONE)  
    r = reg.identify(p)  
    if (r != NONE)  
      account.lookup(r)  
  msg.show("press")
```

photo

reg

account

msg

Design?

scenario no photo:

```
expect(photo.get()).return(NONE)
trigger()
validate(msg.show("press"))
```

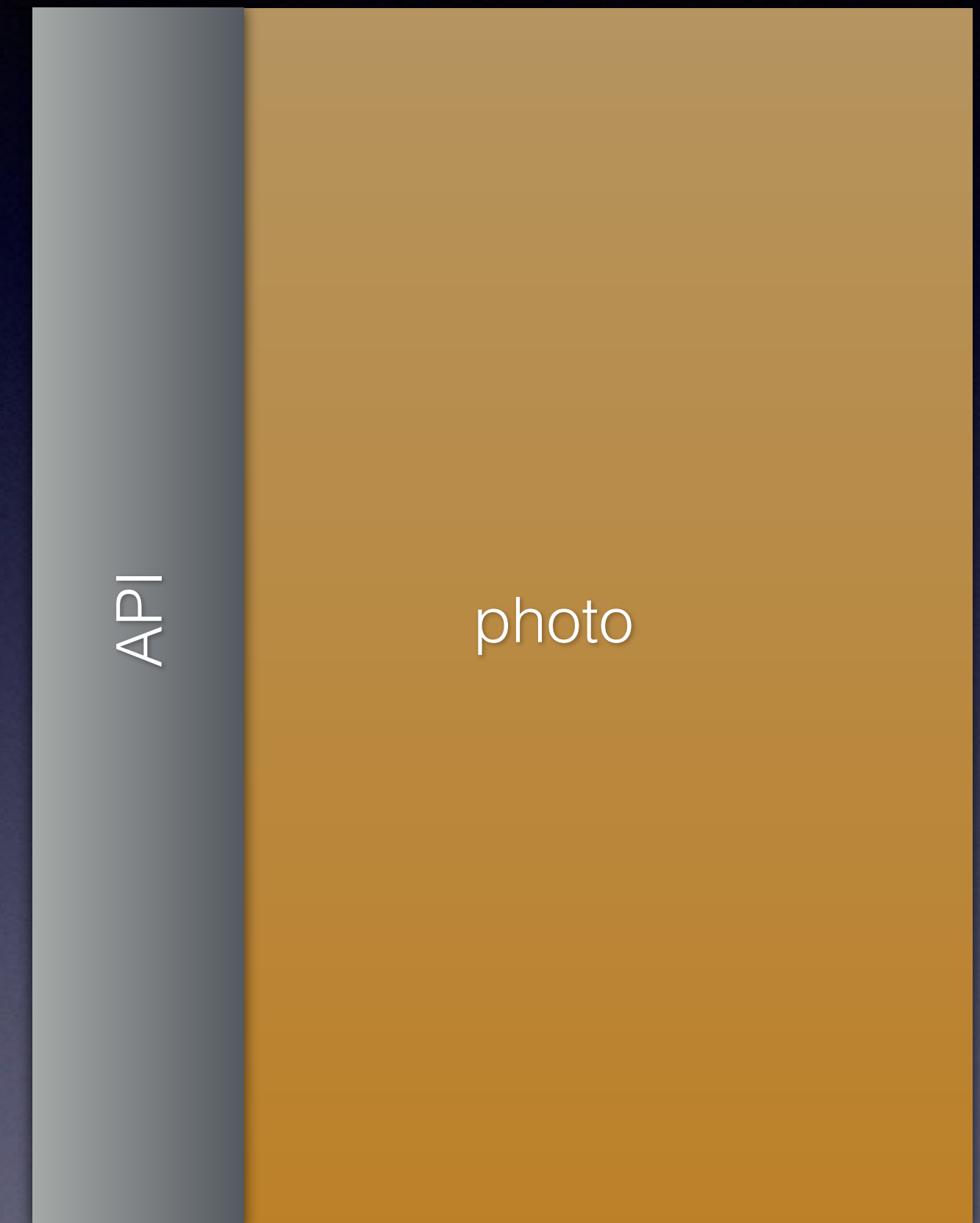
scenario no reg:

```
expect(photo.get()).return(PIC)
expect(reg.identify(PIC)).return(NONE)
trigger()
validate(msg.show("press"))
```

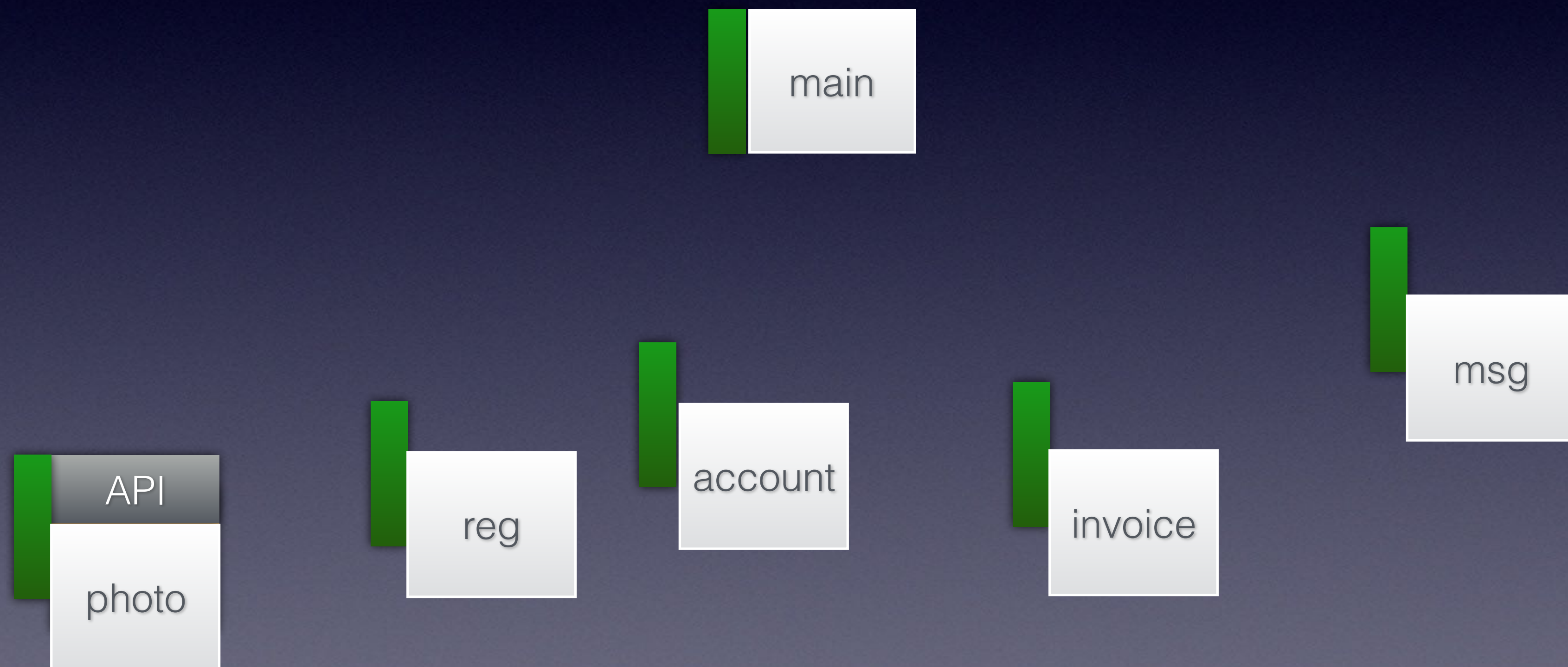
scenario no account:

```
...
expect(account.lookup(REG)).return(NONE)
...
```

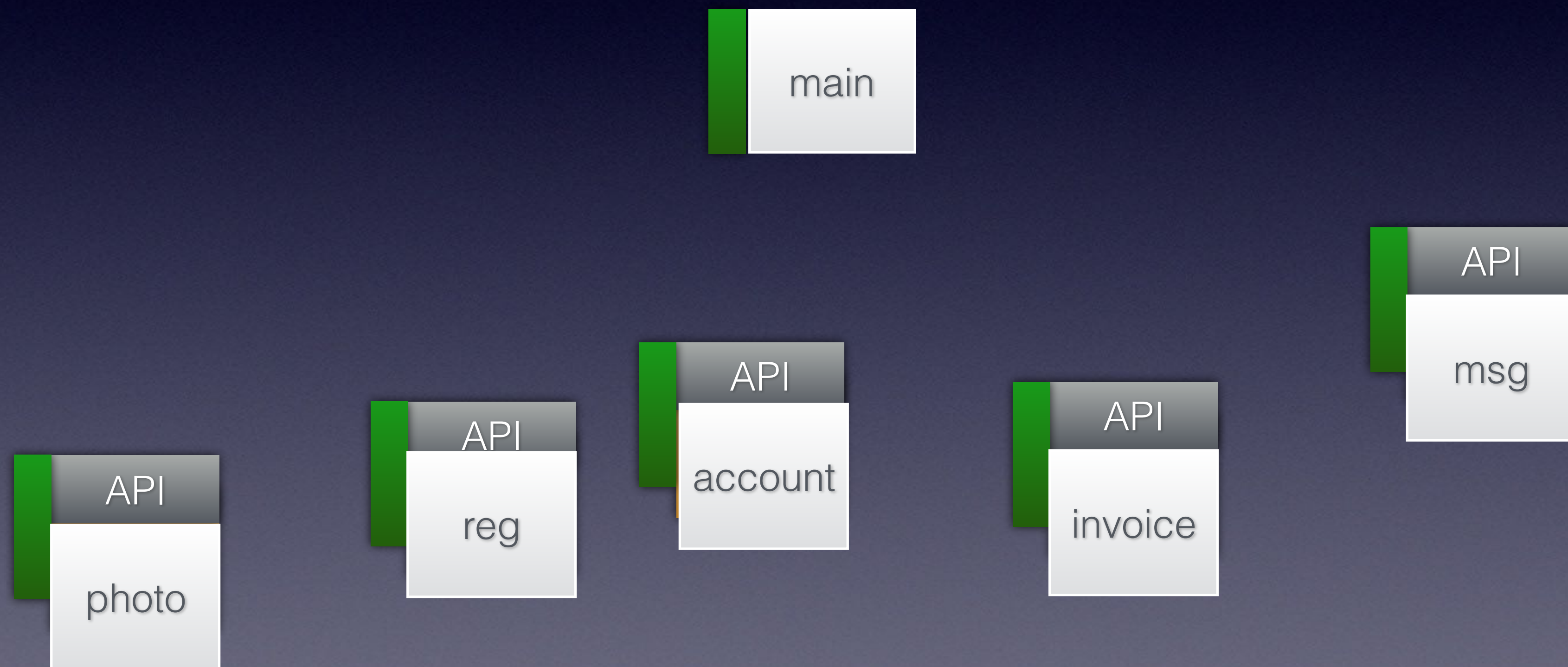
```
expect(...)
expect(...).return(...)
expect(...)
expect(...).return(...)
expect(...)
expect(...).return(...)
expect(...)
expect(...).return(...)
```



Design?



Utifrån och in!



Utifrån och in!

Thomas Nilefalk
@thoni56