Finding Bugs Without Running Or Even Looking At Code

Jay Parlar
@parlar
#formalmethods

Impossible!

Finding Bugs

Finding Bugs

- Reviewing pull requests

Finding Bugs

- Reviewing pull requests
- Testing

Finding Bugs

- Reviewing pull requests
- Testing
- Static analyzers

Talking to an expert and writing stuff down!

“Writing is nature’s way of letting you know how sloppy your thinking is”

- Dick Guindon

What you're probably thinking

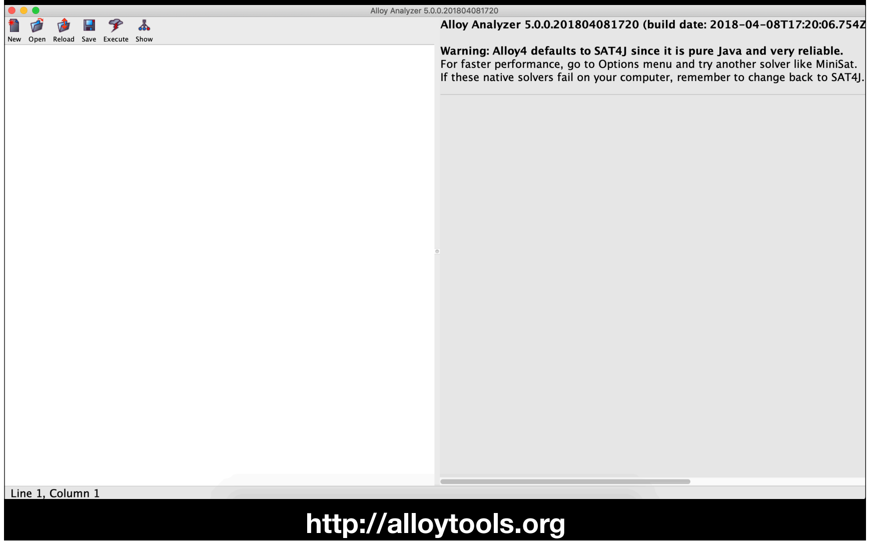
What you're probably thinking

- How should we write down our thoughts/designs/plans?

What you're probably thinking

- How should we write down our thoughts/designs/plans?
- Did I really come to a talk about bug discovery, where the solution is "go talk to people and write stuff down"?

Alloy



Alloy Analyzer 5.0.0.201804081720 (build date: 2018-04-08T17:20:06.754Z

Warning: Alloy4 defaults to SAT4J since it is pure Java and very reliable.
For faster performance, go to Options menu and try another solver like MiniSat.
If these native solvers fail on your computer, remember to change back to SAT4J.

http://alloytools.org

Our Example

Our Example

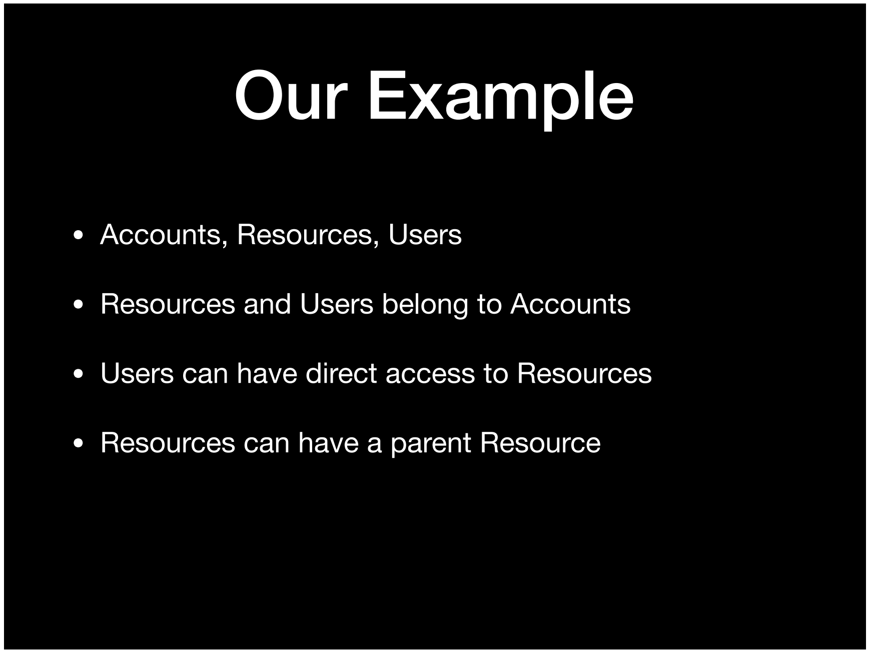
- Accounts, Resources, Users

Our Example

- Accounts, Resources, Users
- Resources and Users belong to Accounts

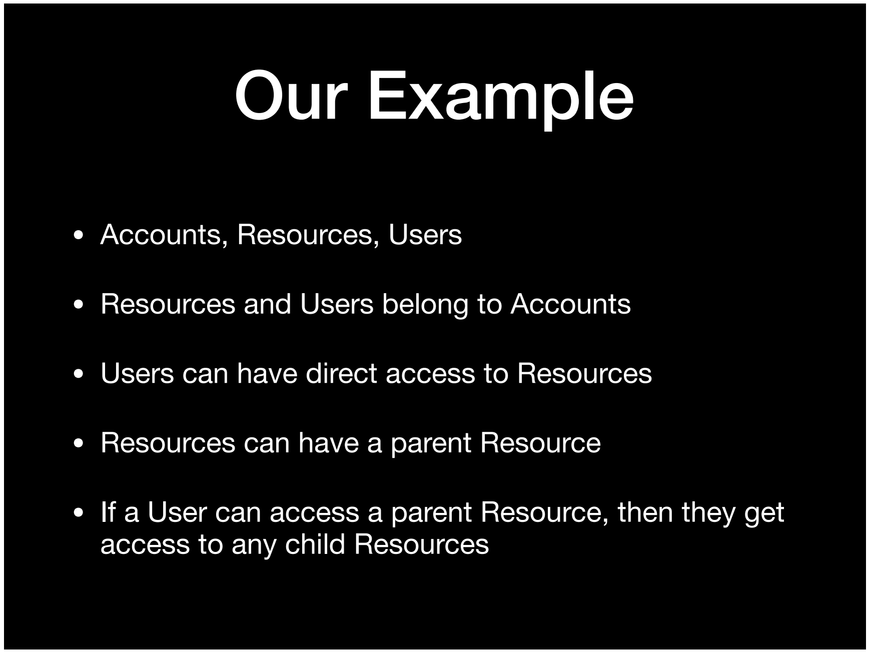
Our Example

- Accounts, Resources, Users
- Resources and Users belong to Accounts
- Users can have direct access to Resources



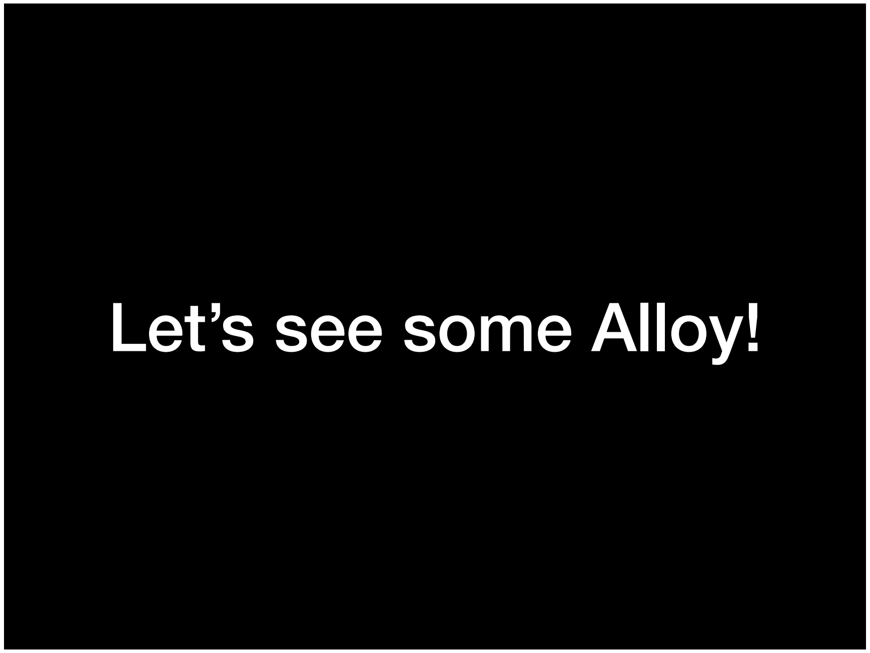
Our Example

- Accounts, Resources, Users
- Resources and Users belong to Accounts
- Users can have direct access to Resources
- Resources can have a parent Resource

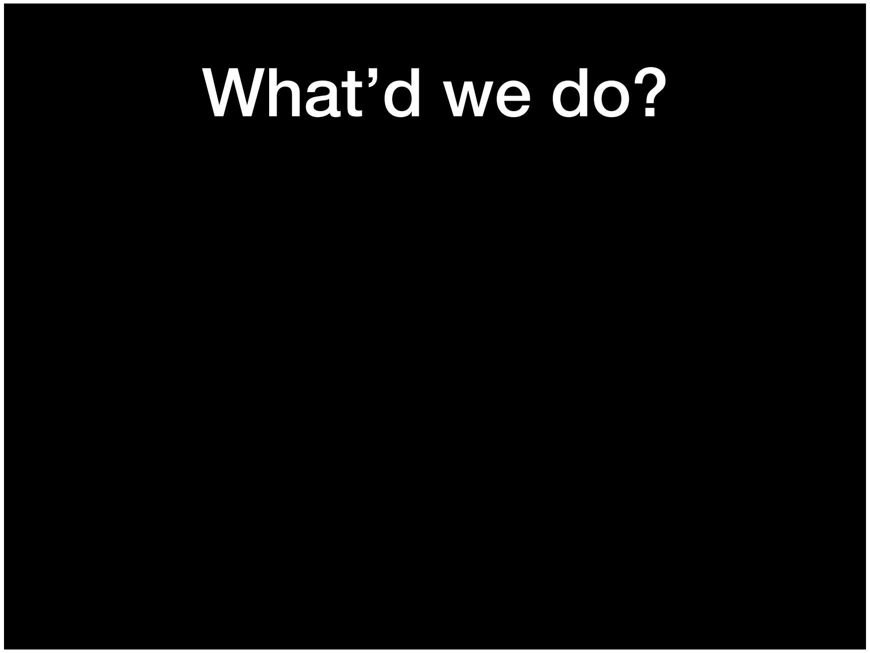


Our Example

- Accounts, Resources, Users
- Resources and Users belong to Accounts
- Users can have direct access to Resources
- Resources can have a parent Resource
- If a User can access a parent Resource, then they get access to any child Resources



Let’s see some Alloy!



What’d we do?

What’d we do?

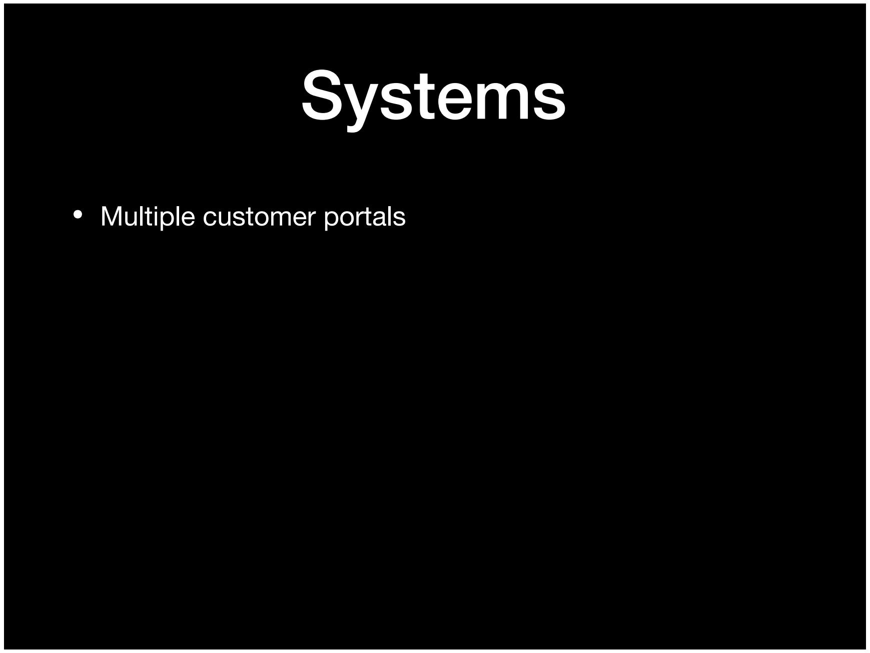
- Design validation

What’d we do?

- Design validation
- Design exploration

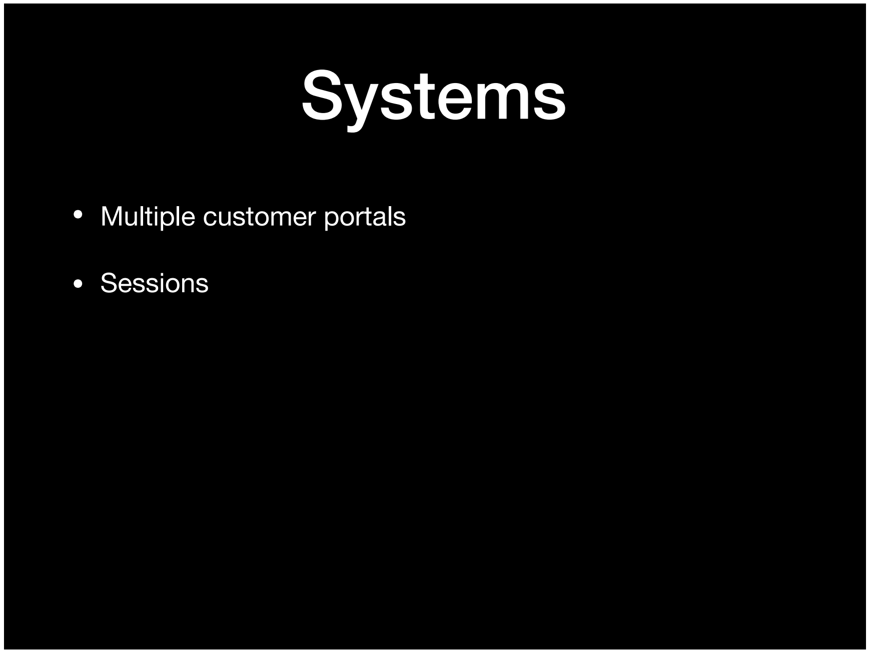
Real Life Example

Systems



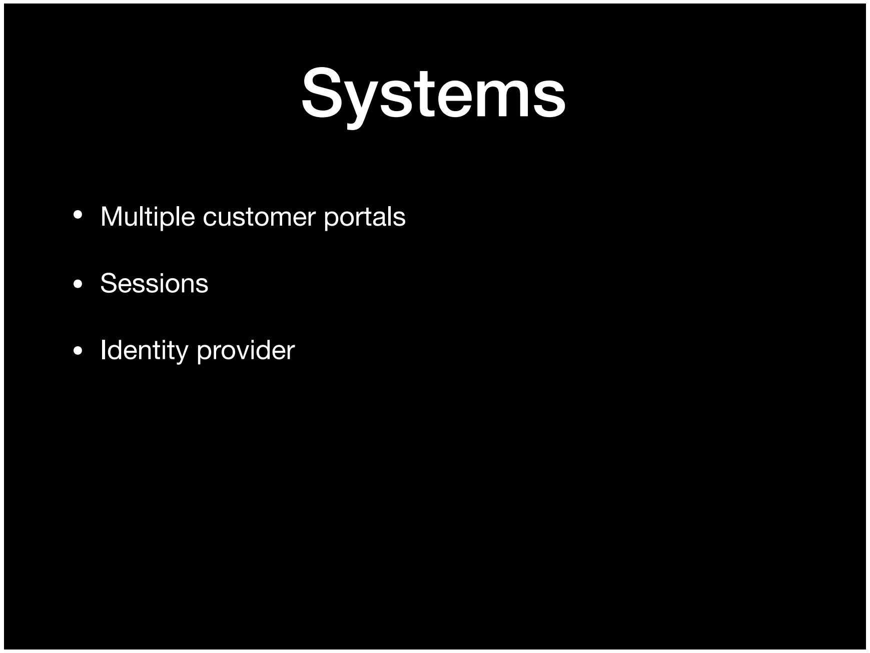
Systems

- Multiple customer portals



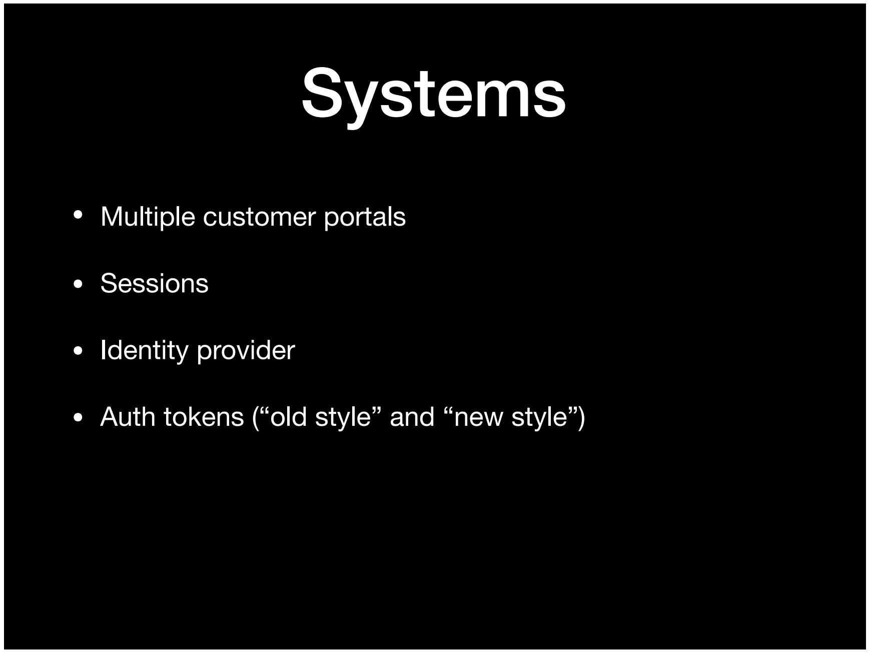
Systems

- Multiple customer portals
- Sessions



Systems

- Multiple customer portals
- Sessions
- Identity provider



Systems

- Multiple customer portals
- Sessions
- Identity provider
- Auth tokens (“old style” and “new style”)

Systems

- Multiple customer portals
- Sessions
- Identity provider
- Auth tokens (“old style” and “new style”)
- SAML requests/responses

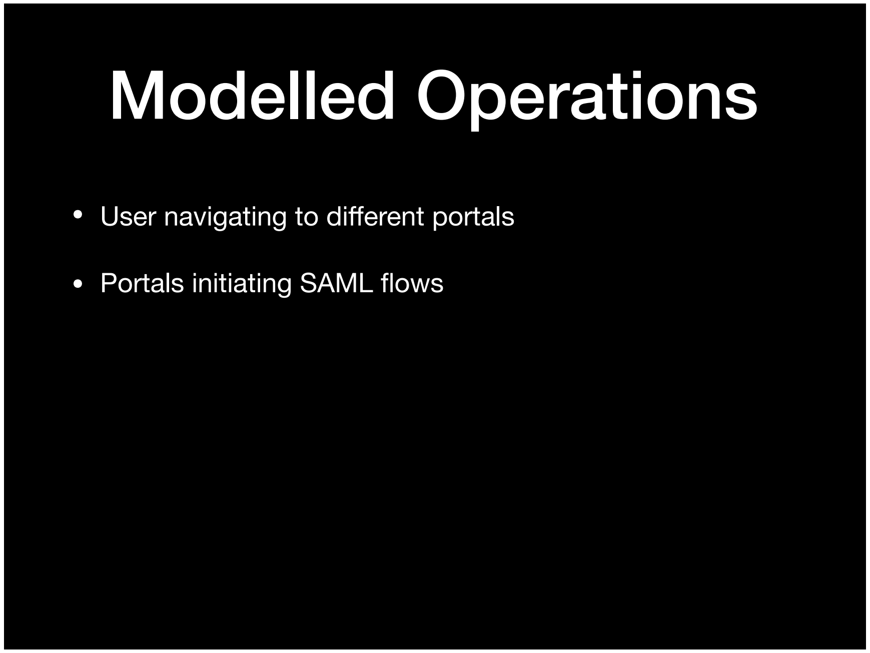
Systems

- Multiple customer portals
- Sessions
- Identity provider
- Auth tokens (“old style” and “new style”)
- SAML requests/responses
- And more…

Modelled Operations

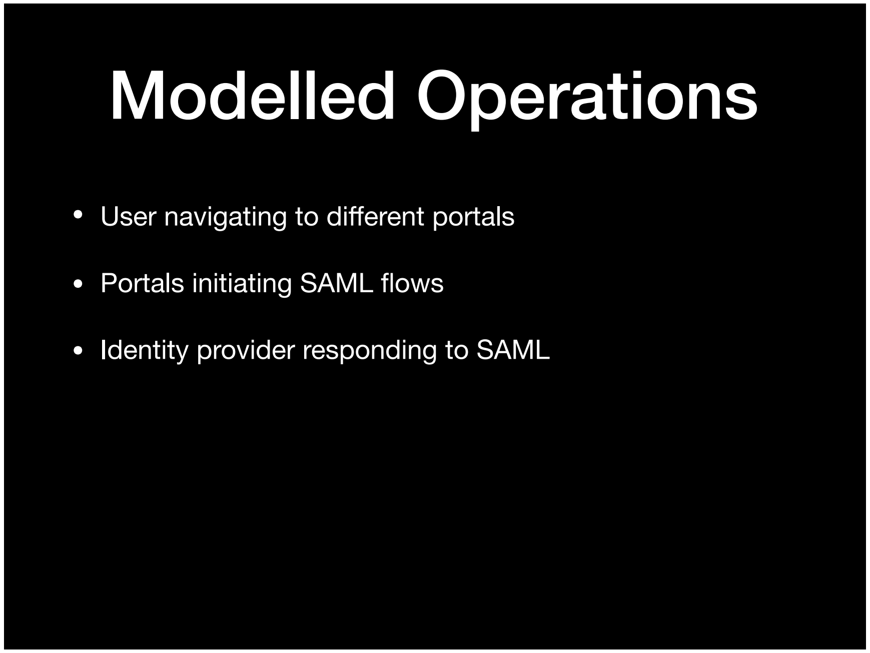
Modelled Operations

- User navigating to different portals



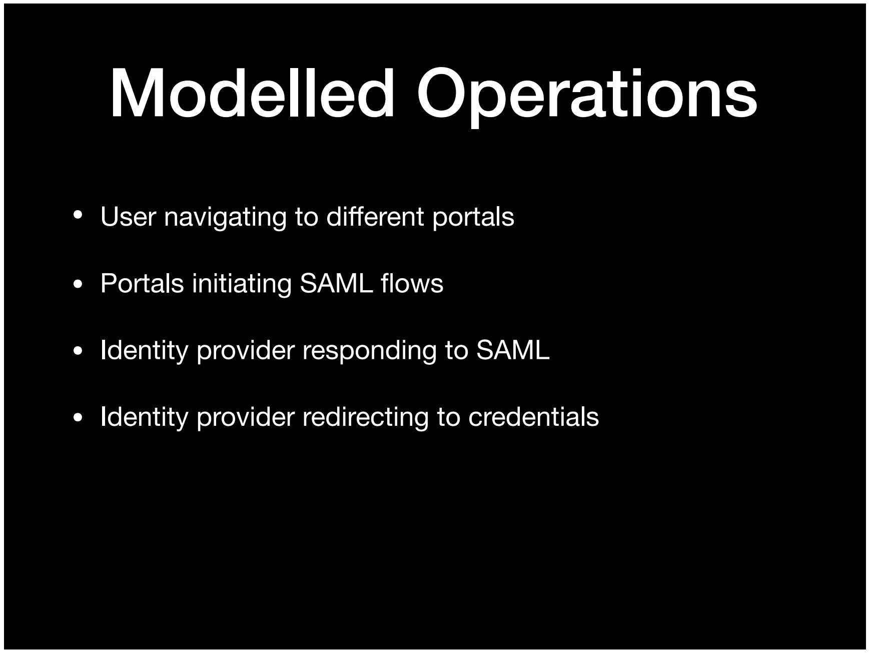
Modelled Operations

- User navigating to different portals
- Portals initiating SAML flows



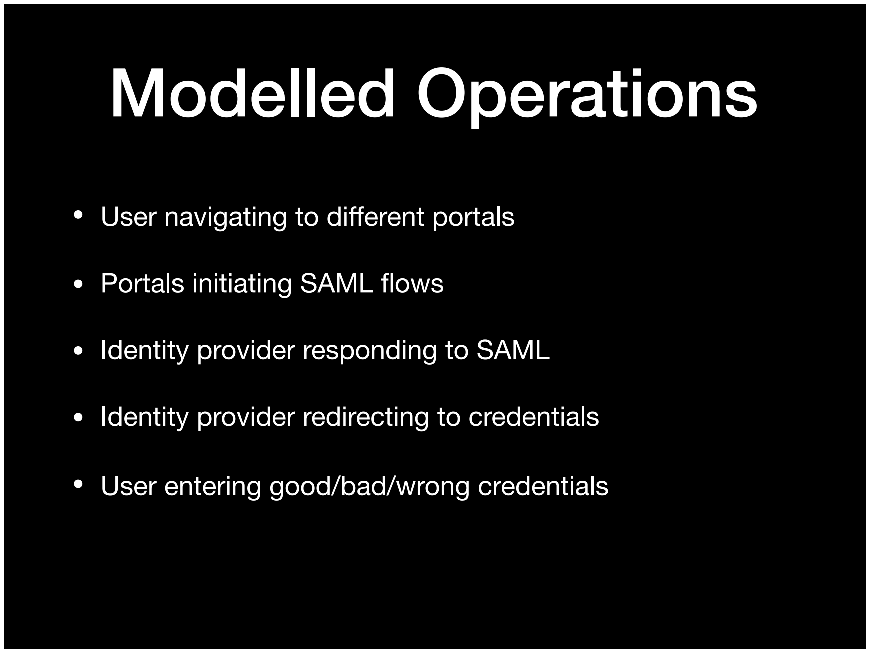
Modelled Operations

- User navigating to different portals
- Portals initiating SAML flows
- Identity provider responding to SAML



Modelled Operations

- User navigating to different portals
- Portals initiating SAML flows
- Identity provider responding to SAML
- Identity provider redirecting to credentials

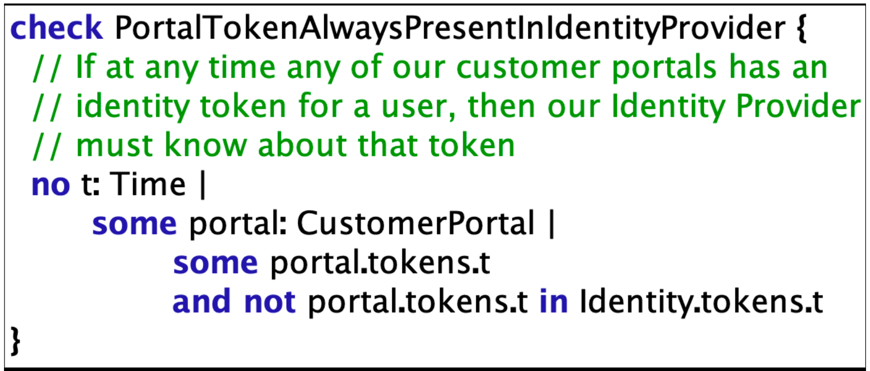


Modelled Operations

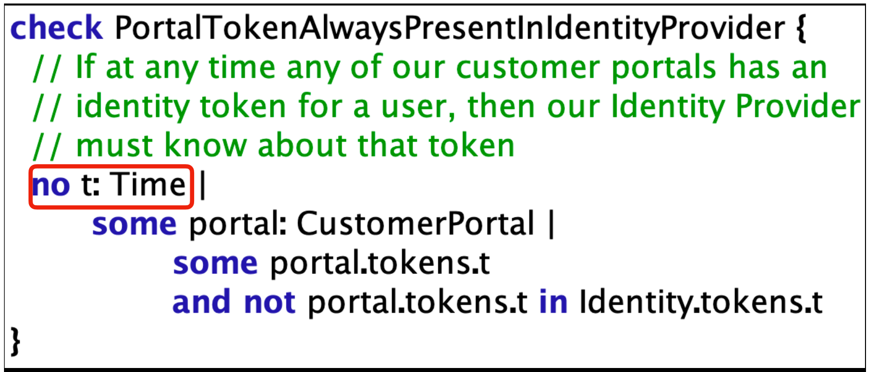
- User navigating to different portals
- Portals initiating SAML flows
- Identity provider responding to SAML
- Identity provider redirecting to credentials
- User entering good/bad/wrong credentials

Modelled Operations

- User navigating to different portals
- Portals initiating SAML flows
- Identity provider responding to SAML
- Identity provider redirecting to credentials
- User entering good/bad/wrong credentials
- And more…

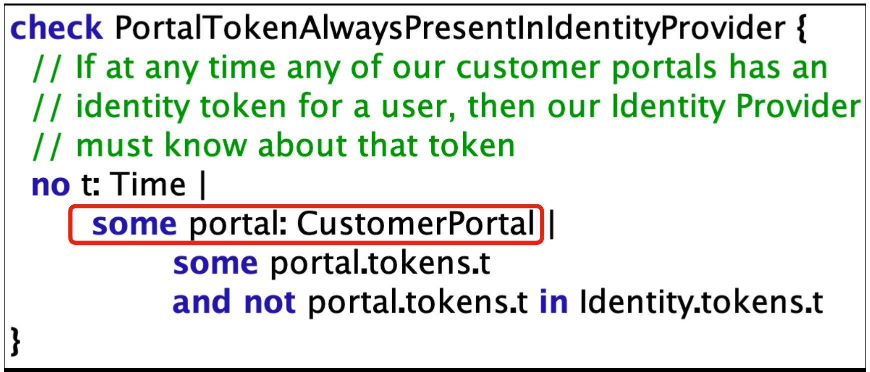


```
check PortalTokenAlwaysPresentInIdentityProvider {
  // If at any time any of our customer portals has an
  // identity token for a user, then our Identity Provider
  // must know about that token
  no t: Time |
      some portal: CustomerPortal |
          some portal.tokens.t
          and not portal.tokens.t in Identity.tokens.t
}
```



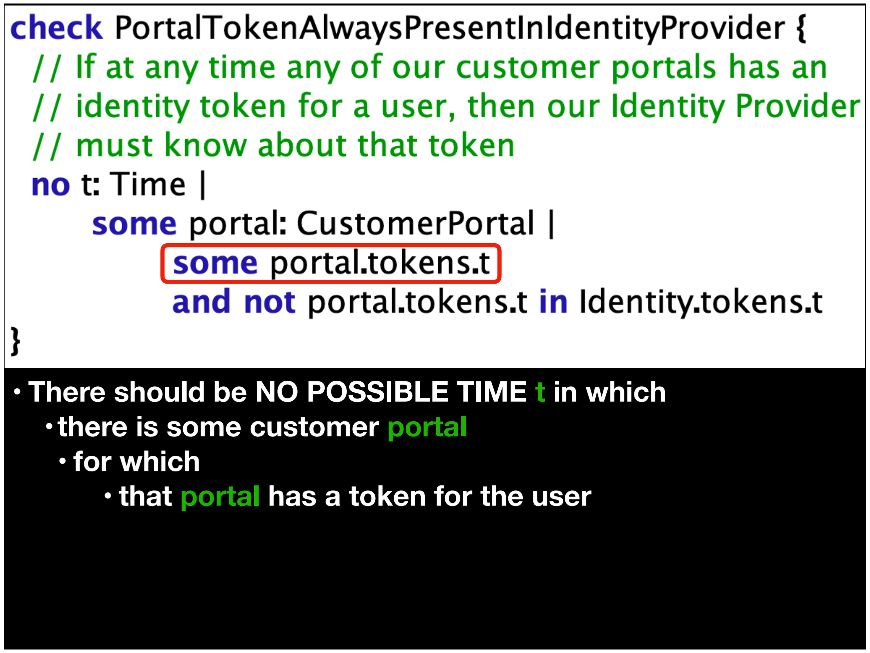
```
check PortalTokenAlwaysPresentInIdentityProvider {
  // If at any time any of our customer portals has an
  // identity token for a user, then our Identity Provider
  // must know about that token
  no t: Time |
      some portal: CustomerPortal |
          some portal.tokens.t
          and not portal.tokens.t in Identity.tokens.t
}
```

- **There should be NO POSSIBLE TIME t in which**



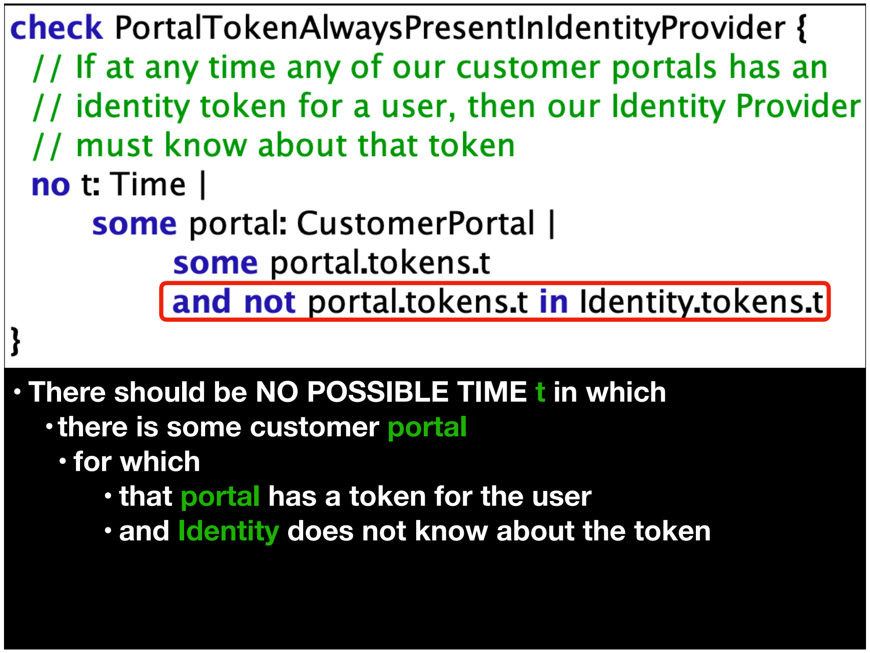
```
check PortalTokenAlwaysPresentInIdentityProvider {
  // If at any time any of our customer portals has an
  // identity token for a user, then our Identity Provider
  // must know about that token
  no t: Time |
      some portal: CustomerPortal |
          some portal.tokens.t
          and not portal.tokens.t in Identity.tokens.t
}
```

- **There should be NO POSSIBLE TIME t in which**
  - **there is some customer portal**
    - **for which**



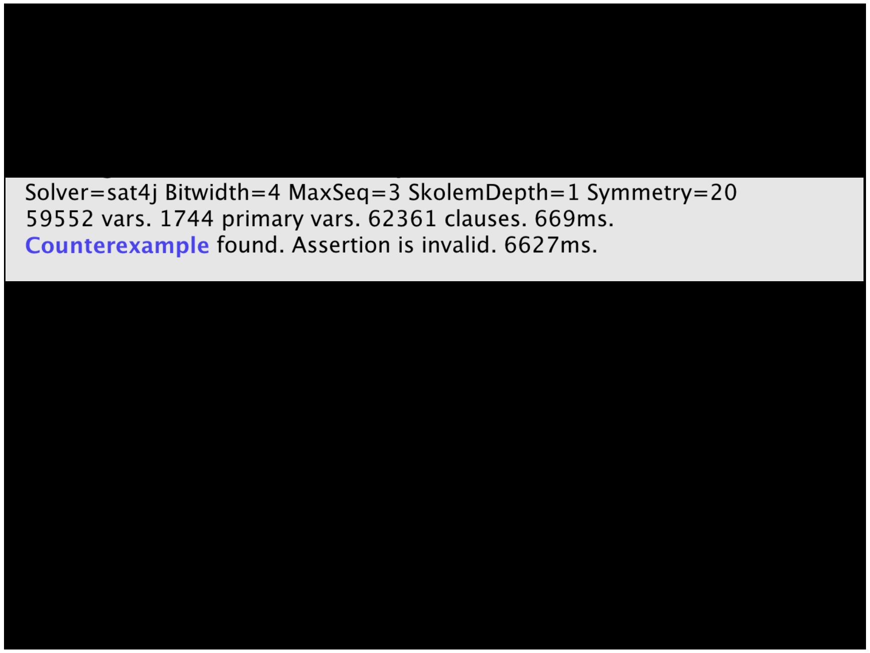
```
check PortalTokenAlwaysPresentInIdentityProvider {
  // If at any time any of our customer portals has an
  // identity token for a user, then our Identity Provider
  // must know about that token
  no t: Time |
       some portal: CustomerPortal |
              some portal.tokens.t
              and not portal.tokens.t in Identity.tokens.t
}
```

- There should be NO POSSIBLE TIME t in which
  - there is some customer portal
    - for which
      - that portal has a token for the user

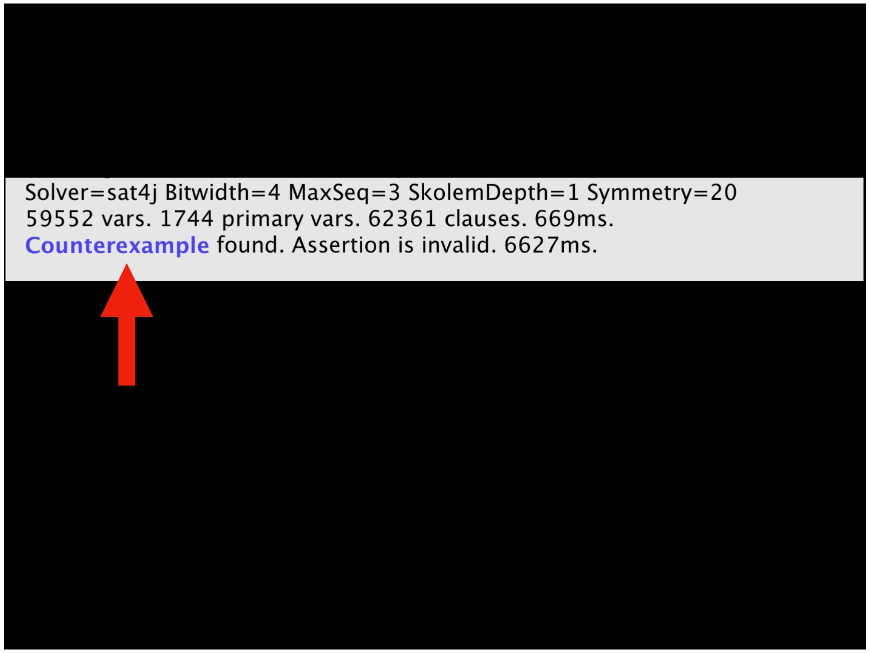


```
check PortalTokenAlwaysPresentInIdentityProvider {
  // If at any time any of our customer portals has an
  // identity token for a user, then our Identity Provider
  // must know about that token
  no t: Time |
       some portal: CustomerPortal |
              some portal.tokens.t
              and not portal.tokens.t in Identity.tokens.t
}
```

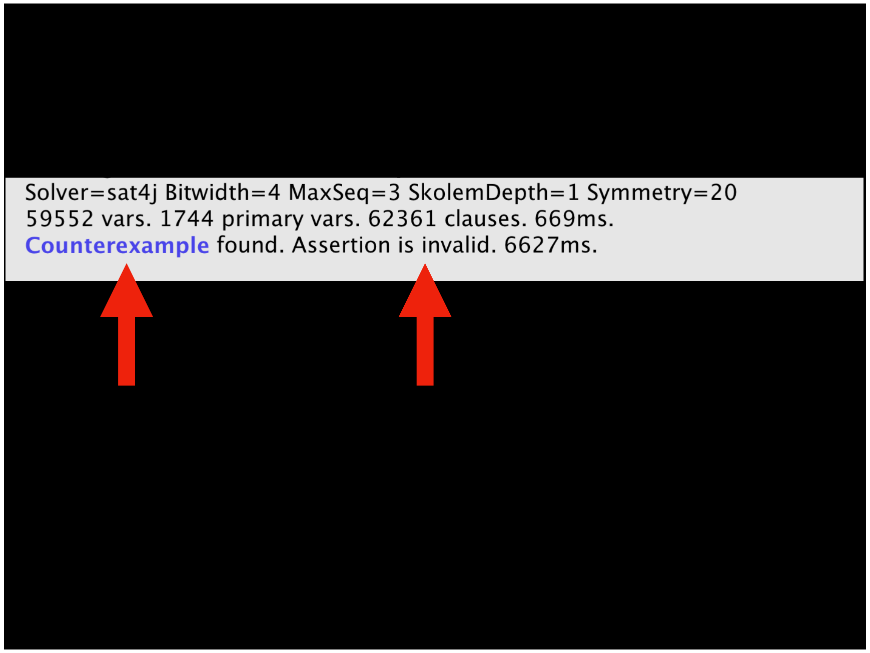
- There should be NO POSSIBLE TIME t in which
  - there is some customer portal
    - for which
      - that portal has a token for the user
      - and Identity does not know about the token



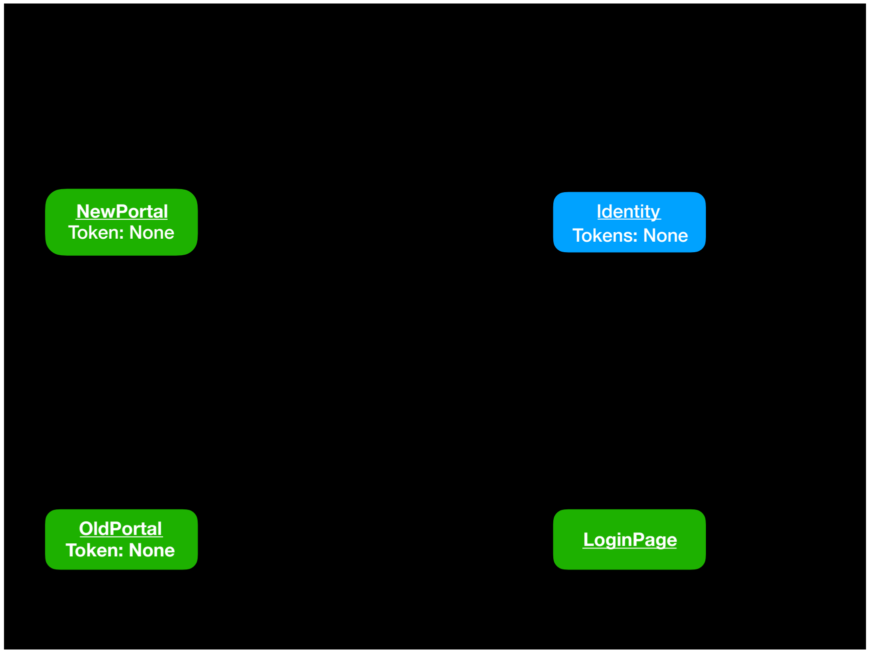
```
Solver=sat4j Bitwidth=4 MaxSeq=3 SkolemDepth=1 Symmetry=20
59552 vars. 1744 primary vars. 62361 clauses. 669ms.
Counterexample found. Assertion is invalid. 6627ms.
```



```
Solver=sat4j Bitwidth=4 MaxSeq=3 SkolemDepth=1 Symmetry=20
59552 vars. 1744 primary vars. 62361 clauses. 669ms.
Counterexample found. Assertion is invalid. 6627ms.
```



```
Solver=sat4j Bitwidth=4 MaxSeq=3 SkolemDepth=1 Symmetry=20
59552 vars. 1744 primary vars. 62361 clauses. 669ms.
Counterexample found. Assertion is invalid. 6627ms.
```

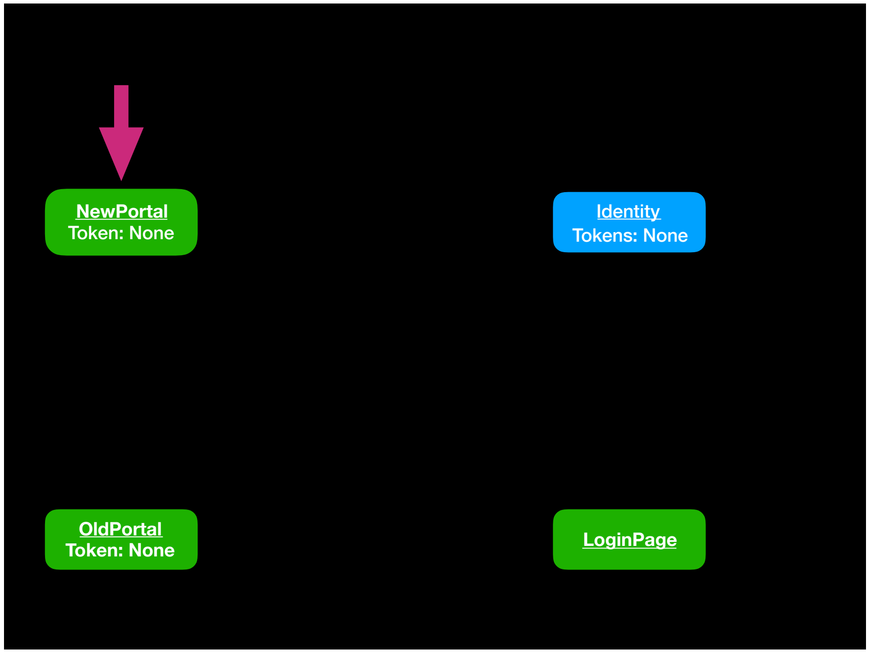


NewPortal
Token: None

Identity
Tokens: None

OldPortal
Token: None

LoginPage

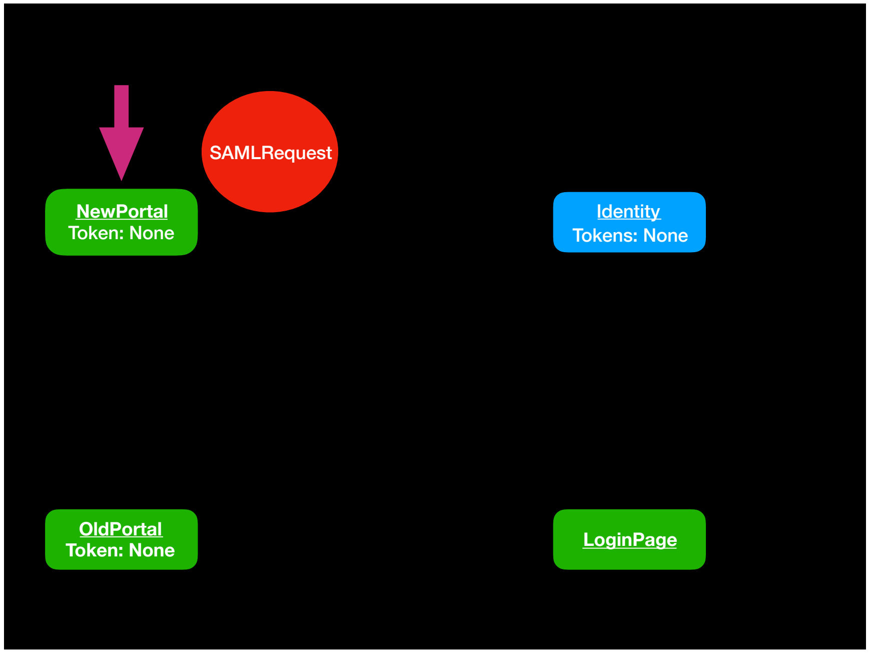


NewPortal
Token: None

Identity
Tokens: None

OldPortal
Token: None

LoginPage



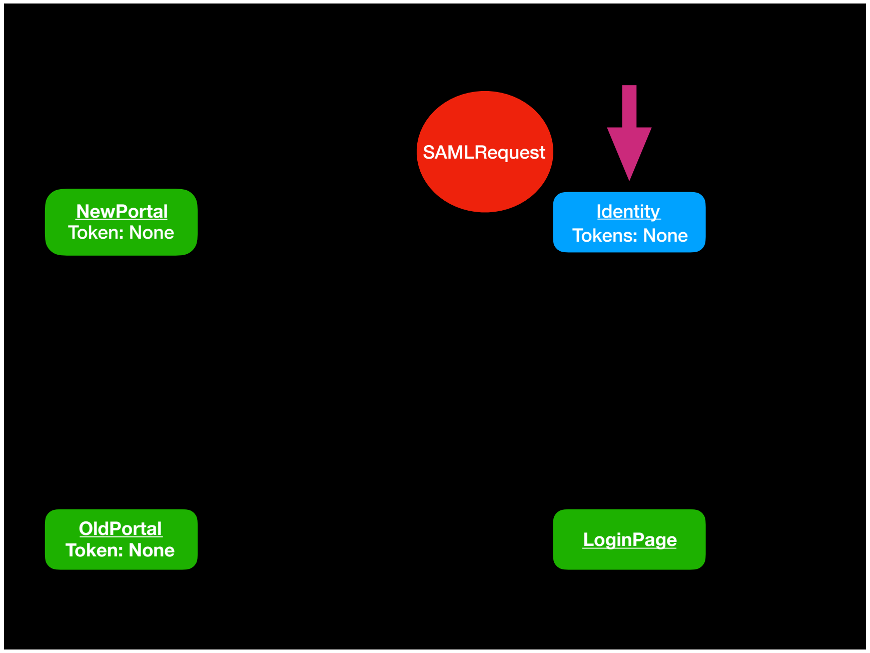
SAMLRequest

NewPortal
Token: None

Identity
Tokens: None

OldPortal
Token: None

LoginPage



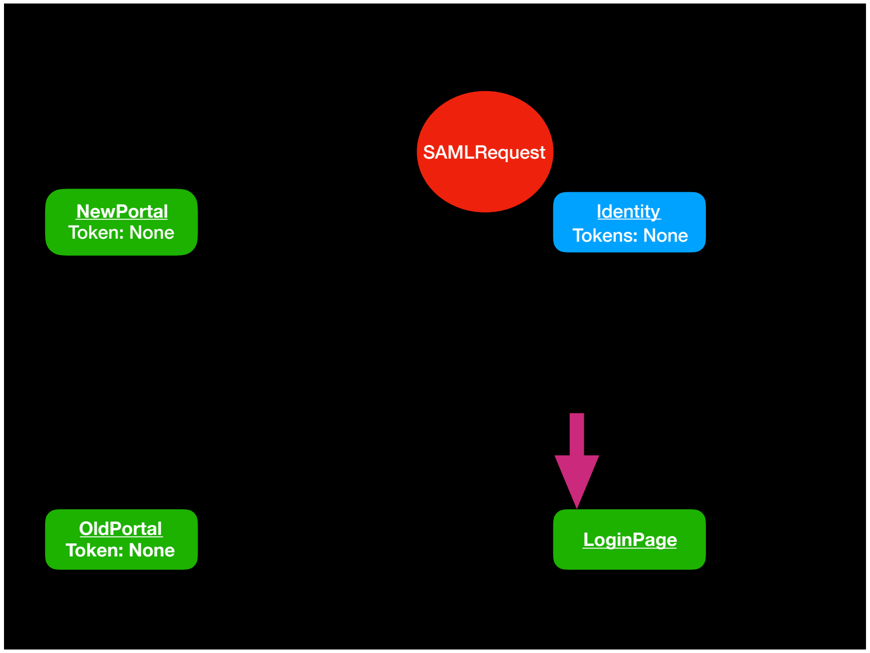
SAMLRequest

NewPortal
Token: None

Identity
Tokens: None

OldPortal
Token: None

LoginPage



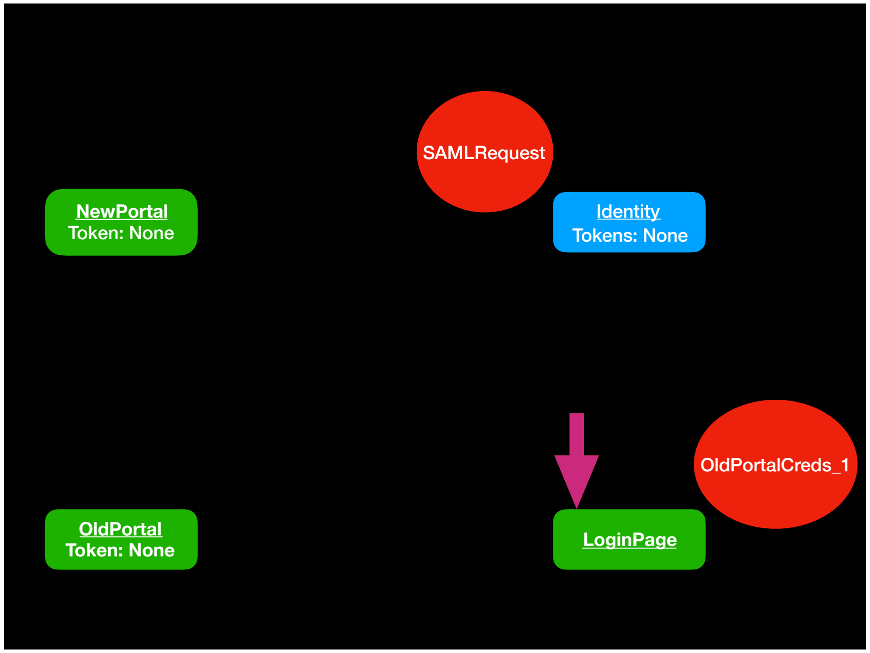
SAMLRequest

NewPortal
Token: None

Identity
Tokens: None

OldPortal
Token: None

LoginPage



SAMLRequest

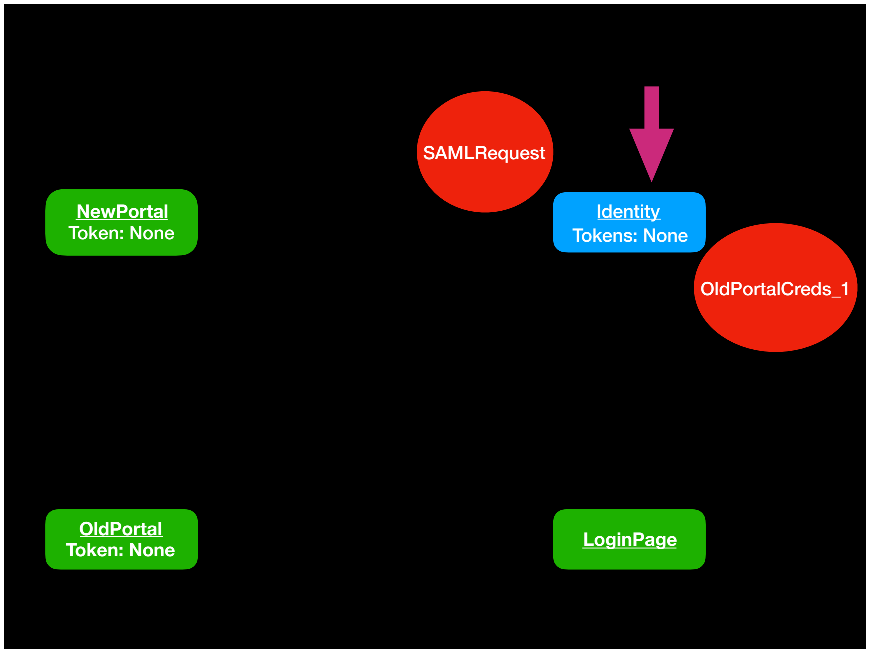
NewPortal
Token: None

Identity
Tokens: None

OldPortalCreds_1

OldPortal
Token: None

LoginPage



SAMLRequest

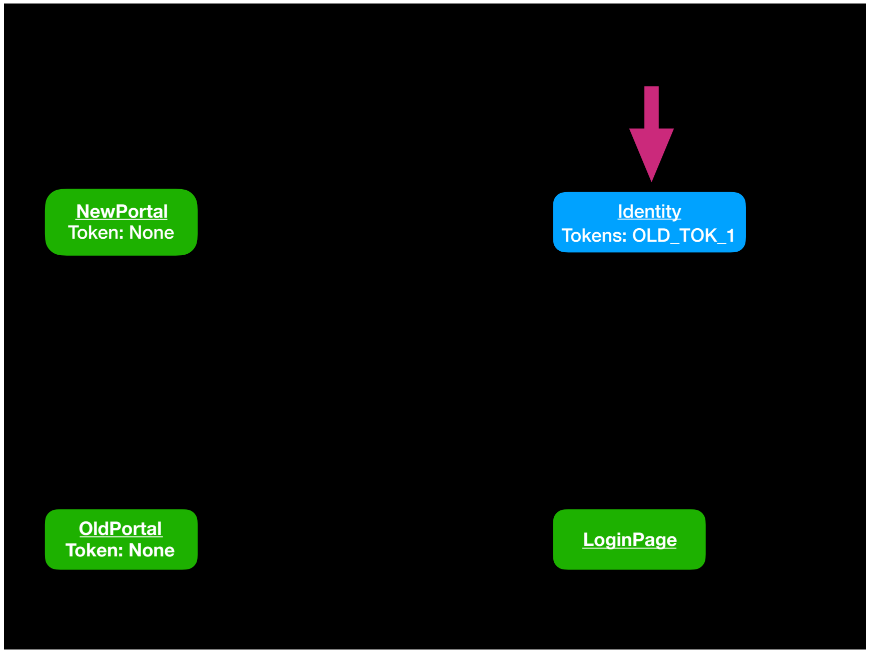
NewPortal
Token: None

Identity
Tokens: None

OldPortalCreds_1

OldPortal
Token: None

LoginPage

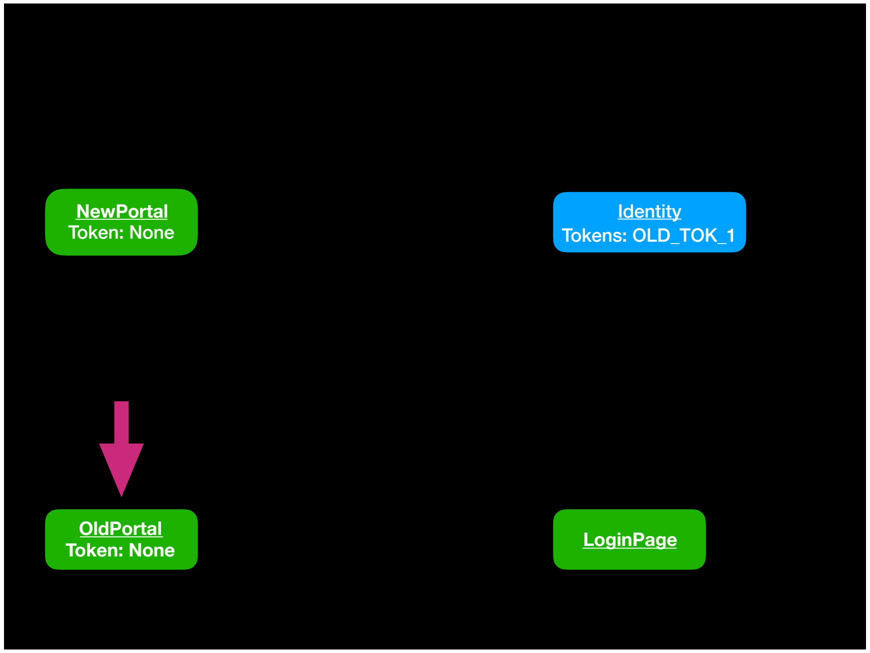


NewPortal
Token: None

Identity
Tokens: OLD_TOK_1

OldPortal
Token: None

LoginPage

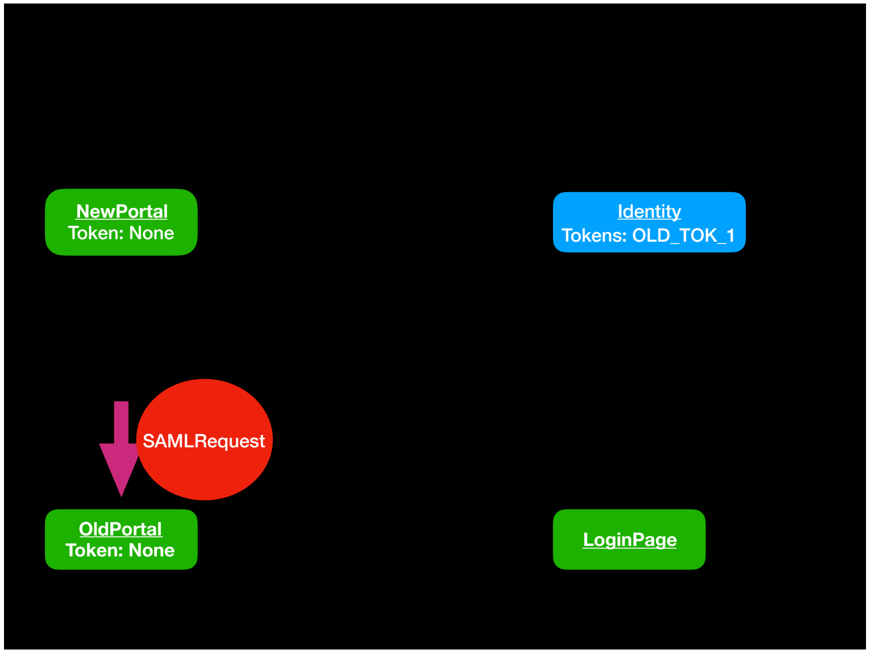


NewPortal
Token: None

Identity
Tokens: OLD_TOK_1

OldPortal
Token: None

LoginPage



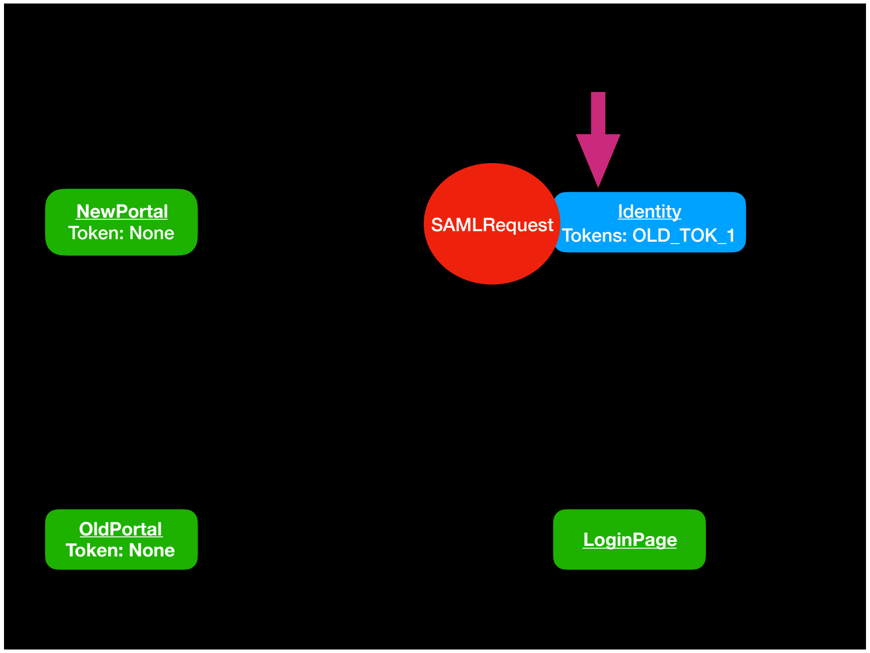
NewPortal
Token: None

Identity
Tokens: OLD_TOK_1

SAMLRequest

OldPortal
Token: None

LoginPage



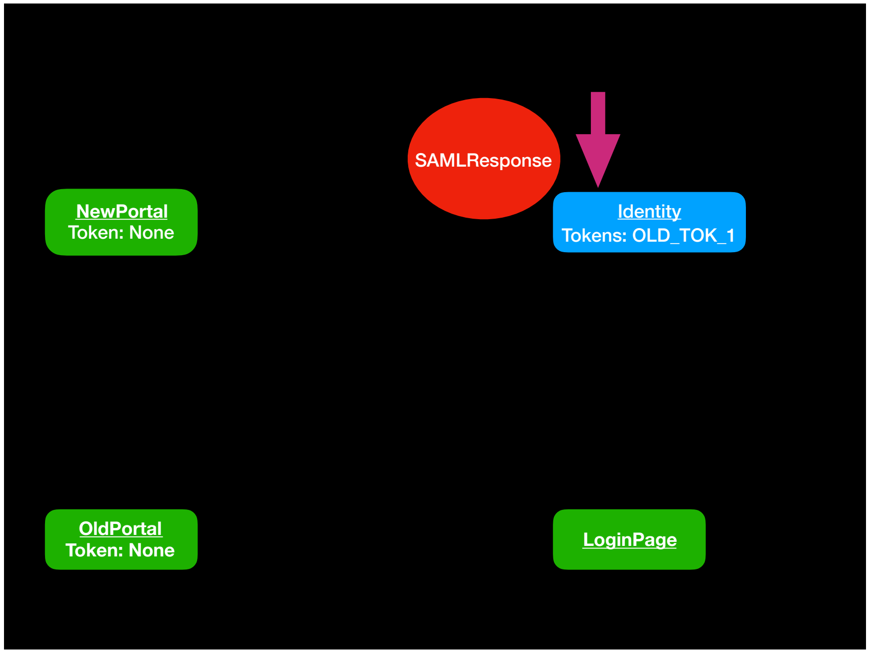
NewPortal
Token: None

SAMLRequest

Identity
Tokens: OLD_TOK_1

OldPortal
Token: None

LoginPage



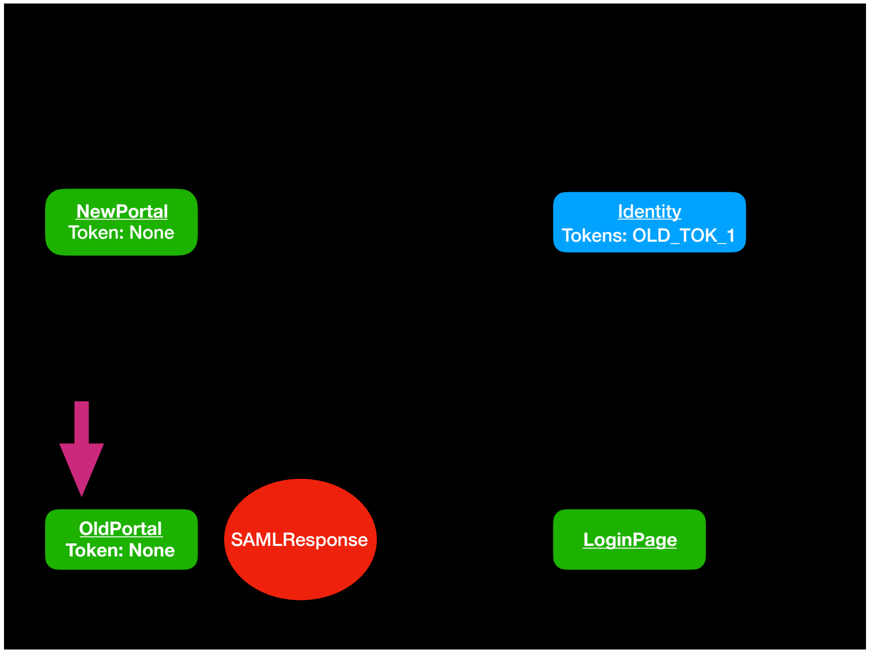
SAMLResponse

NewPortal
Token: None

Identity
Tokens: OLD_TOK_1

OldPortal
Token: None

LoginPage



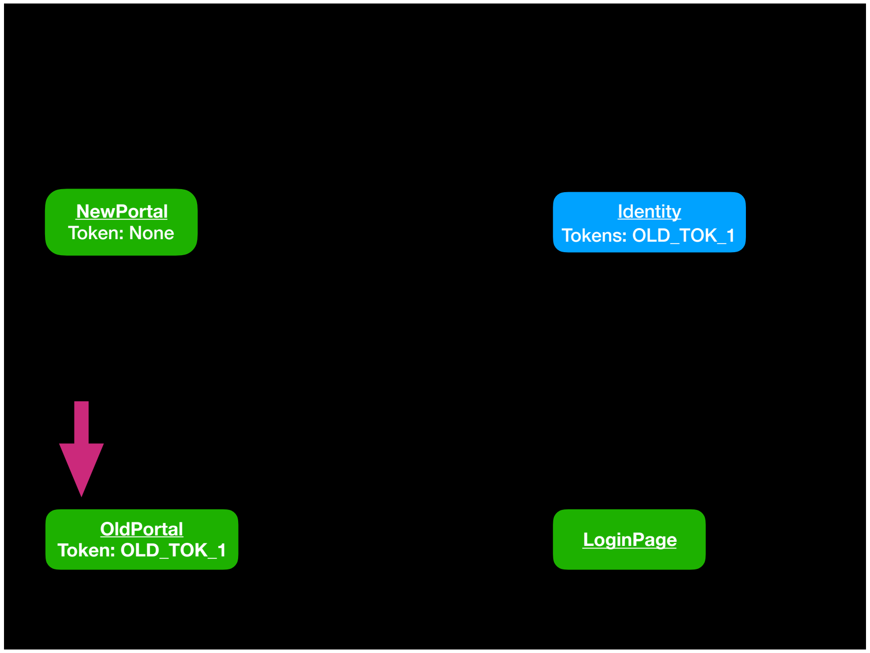
NewPortal
Token: None

Identity
Tokens: OLD_TOK_1

OldPortal
Token: None

SAMLResponse

LoginPage

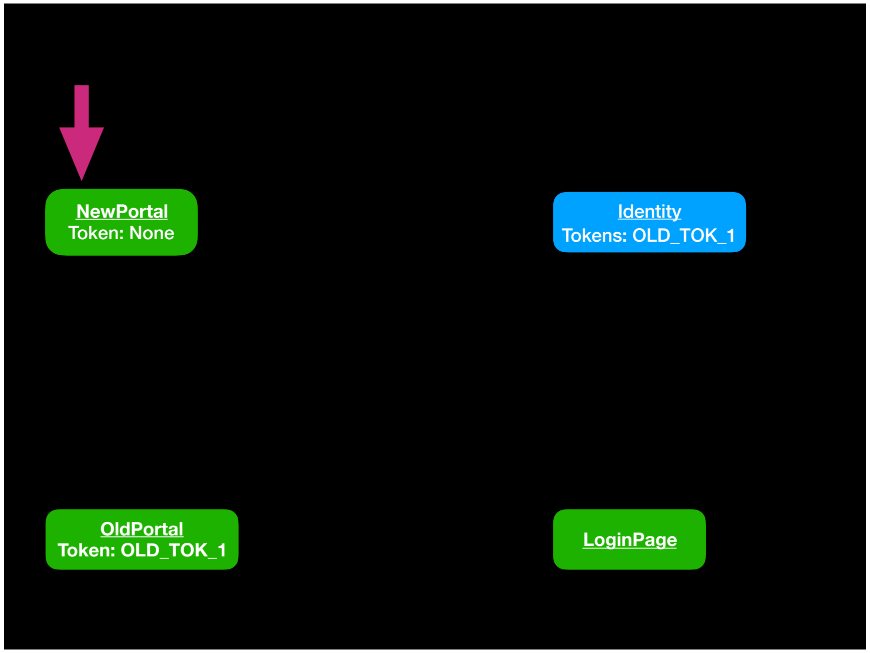


NewPortal
Token: None

Identity
Tokens: OLD_TOK_1

OldPortal
Token: OLD_TOK_1

LoginPage

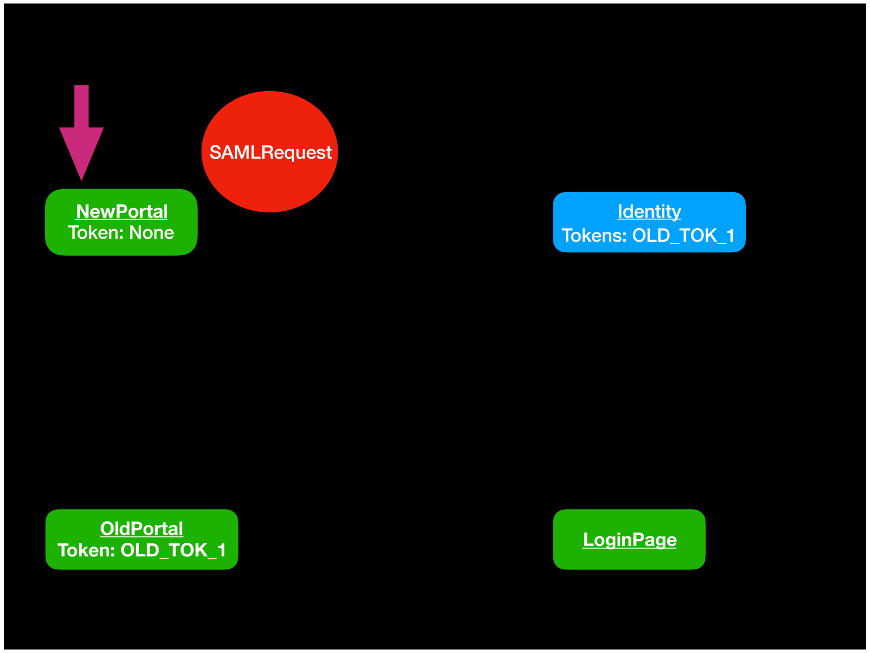


NewPortal
Token: None

Identity
Tokens: OLD_TOK_1

OldPortal
Token: OLD_TOK_1

LoginPage



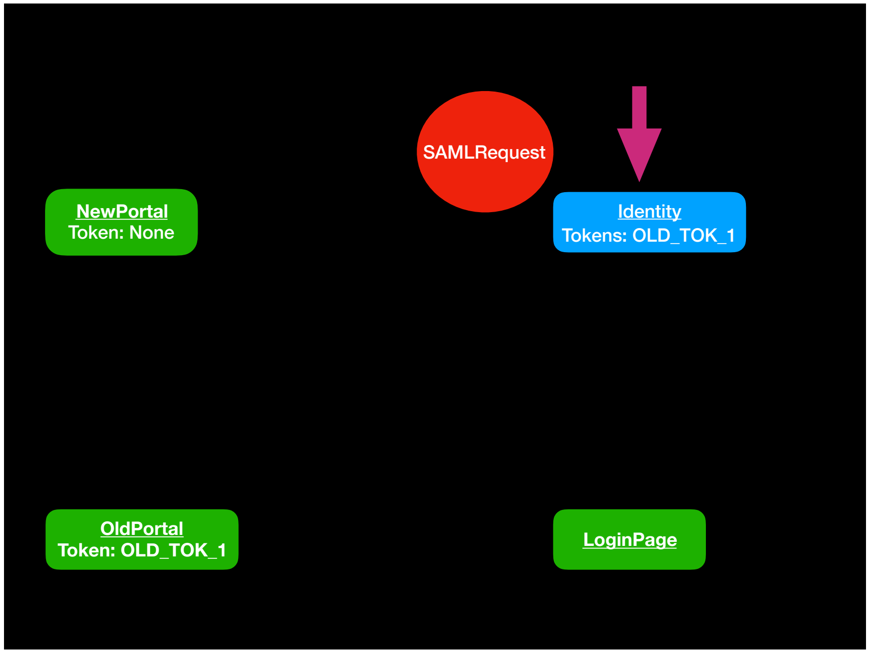
SAMLRequest

NewPortal
Token: None

Identity
Tokens: OLD_TOK_1

OldPortal
Token: OLD_TOK_1

LoginPage



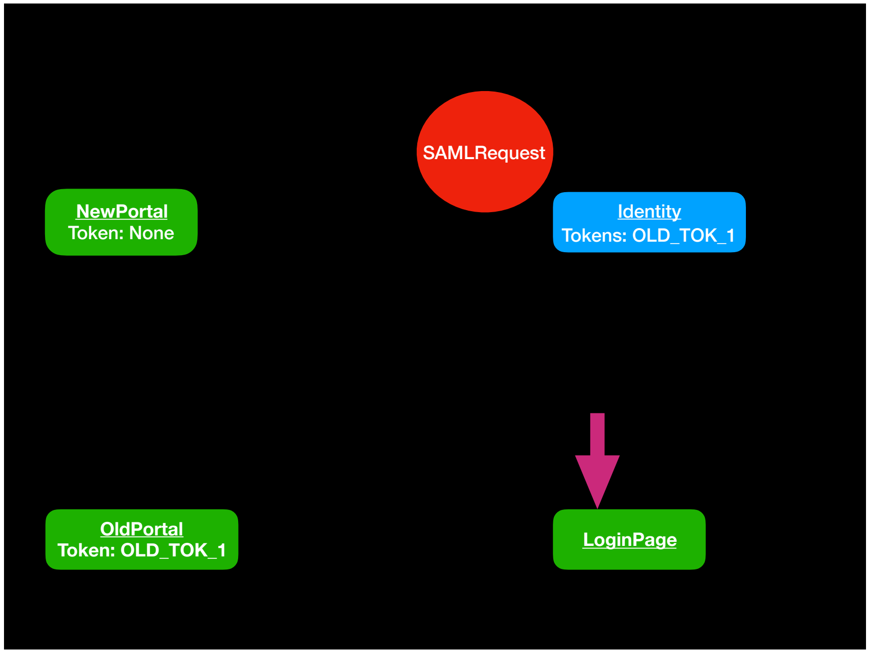
SAMLRequest

NewPortal
Token: None

Identity
Tokens: OLD_TOK_1

OldPortal
Token: OLD_TOK_1

LoginPage



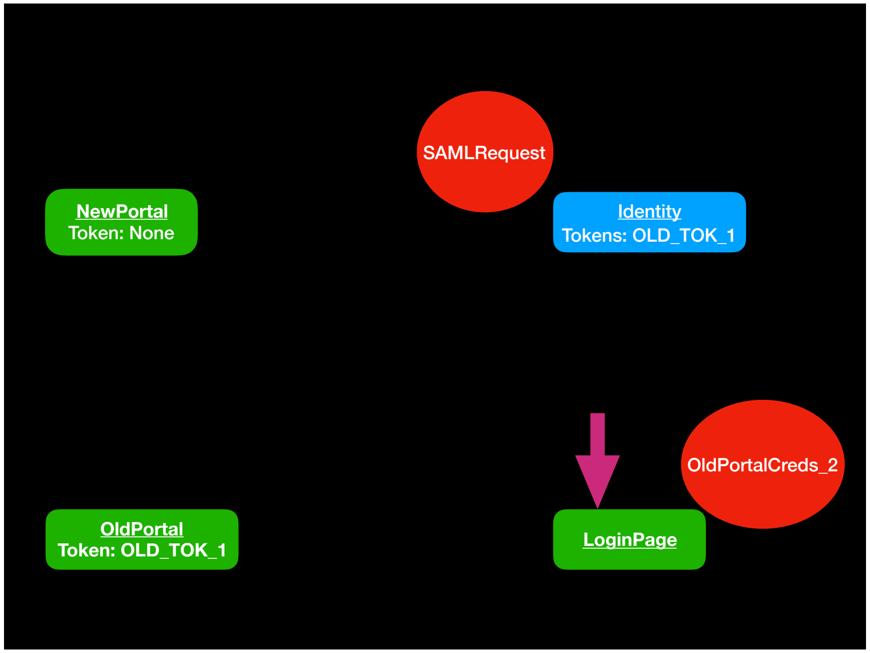
SAMLRequest

NewPortal
Token: None

Identity
Tokens: OLD_TOK_1

OldPortal
Token: OLD_TOK_1

LoginPage



SAMLRequest

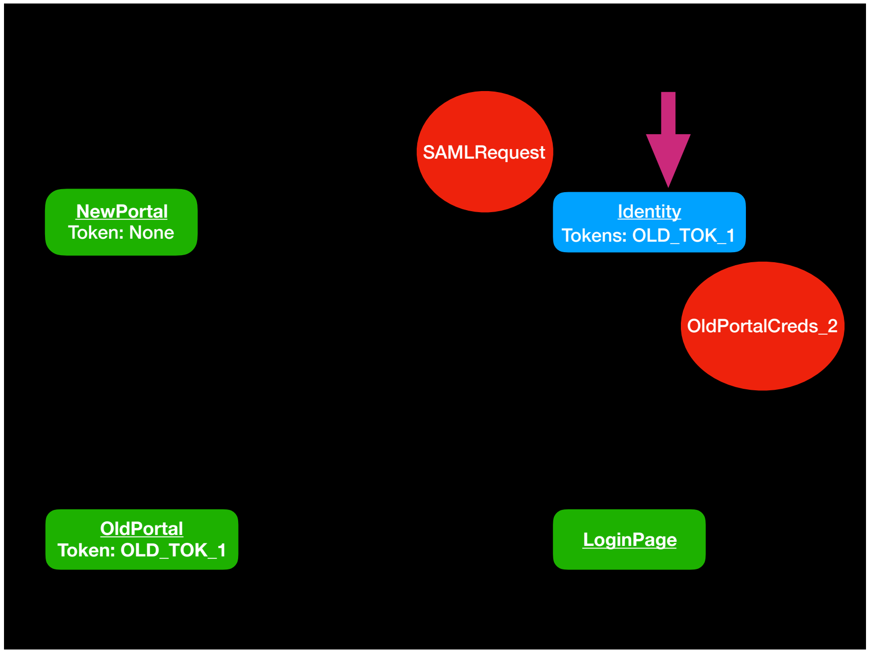
NewPortal
Token: None

Identity
Tokens: OLD_TOK_1

OldPortalCreds_2

OldPortal
Token: OLD_TOK_1

LoginPage



SAMLRequest

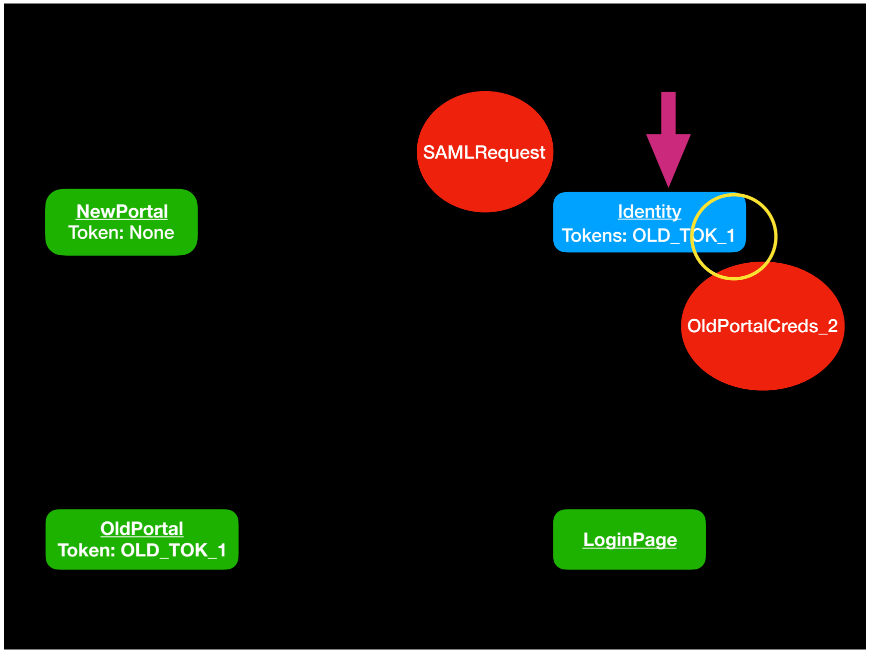
NewPortal
Token: None

Identity
Tokens: OLD_TOK_1

OldPortalCreds_2

OldPortal
Token: OLD_TOK_1

LoginPage



SAMLRequest

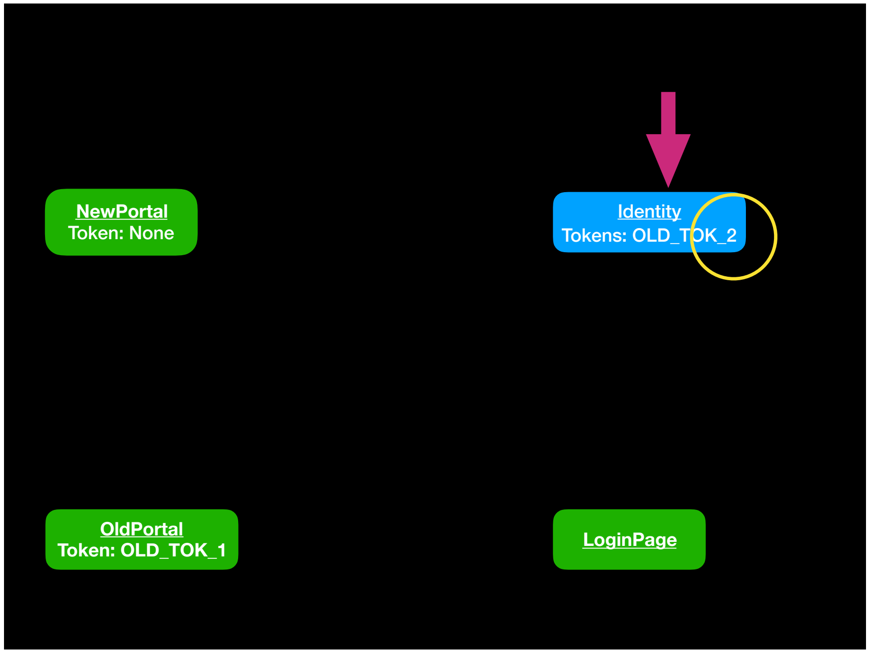
NewPortal
Token: None

Identity
Tokens: OLD_TOK_1

OldPortalCreds_2

OldPortal
Token: OLD_TOK_1

LoginPage

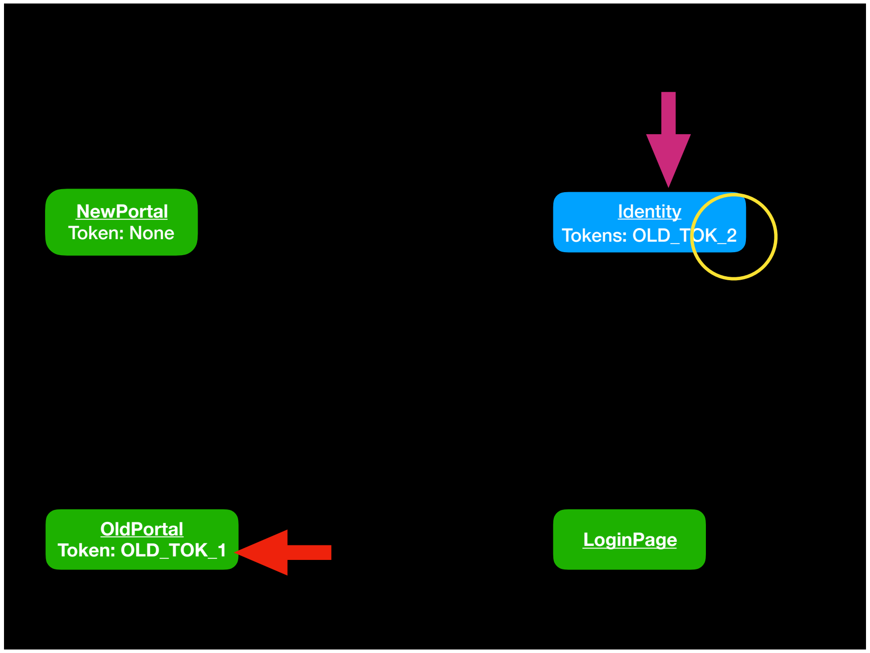


NewPortal
Token: None

Identity
Tokens: OLD_TOK_2

OldPortal
Token: OLD_TOK_1

LoginPage

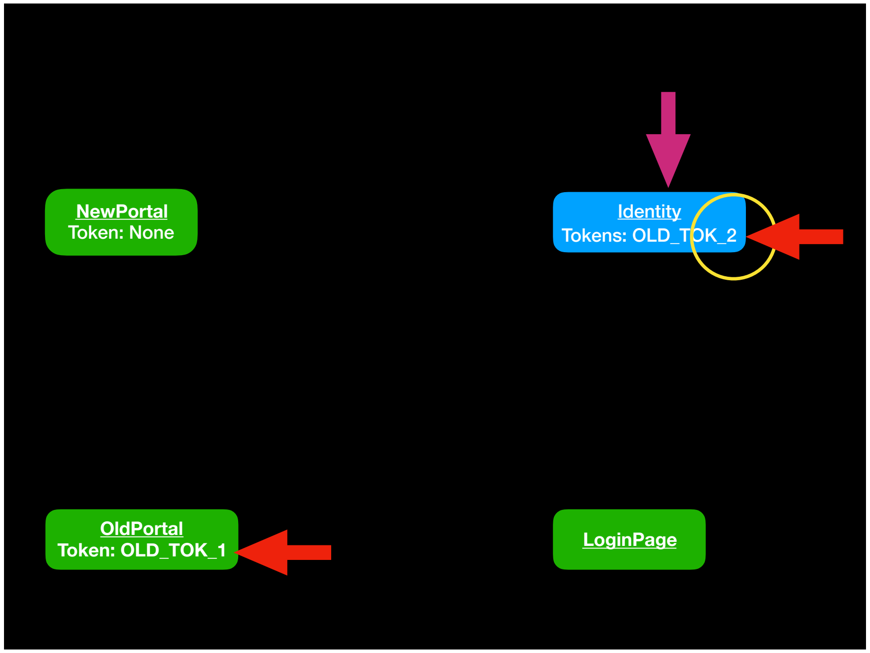


NewPortal
Token: None

Identity
Tokens: OLD_TOK_2

OldPortal
Token: OLD_TOK_1

LoginPage

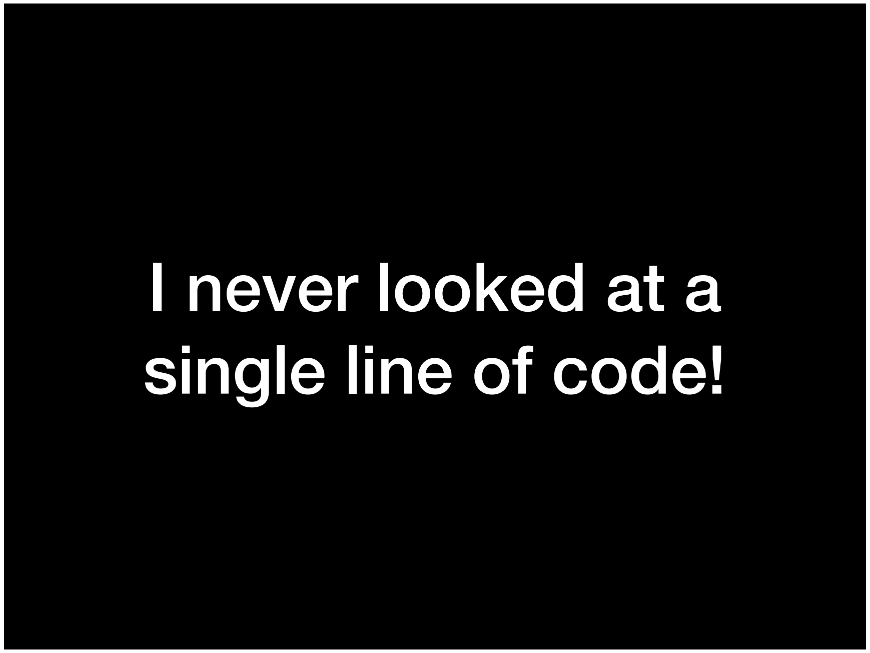


NewPortal
Token: None

Identity
Tokens: OLD_TOK_2

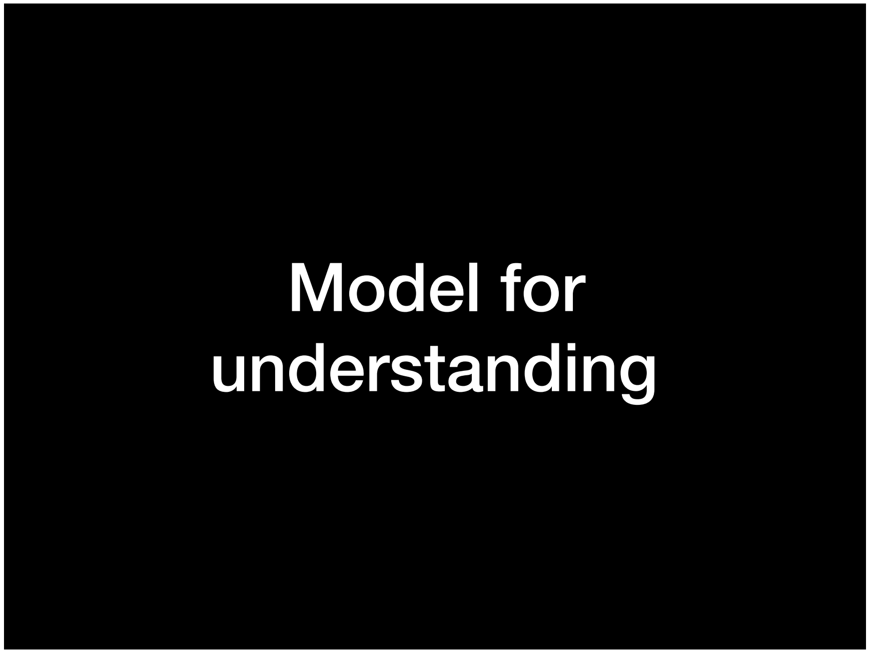
OldPortal
Token: OLD_TOK_1

LoginPage

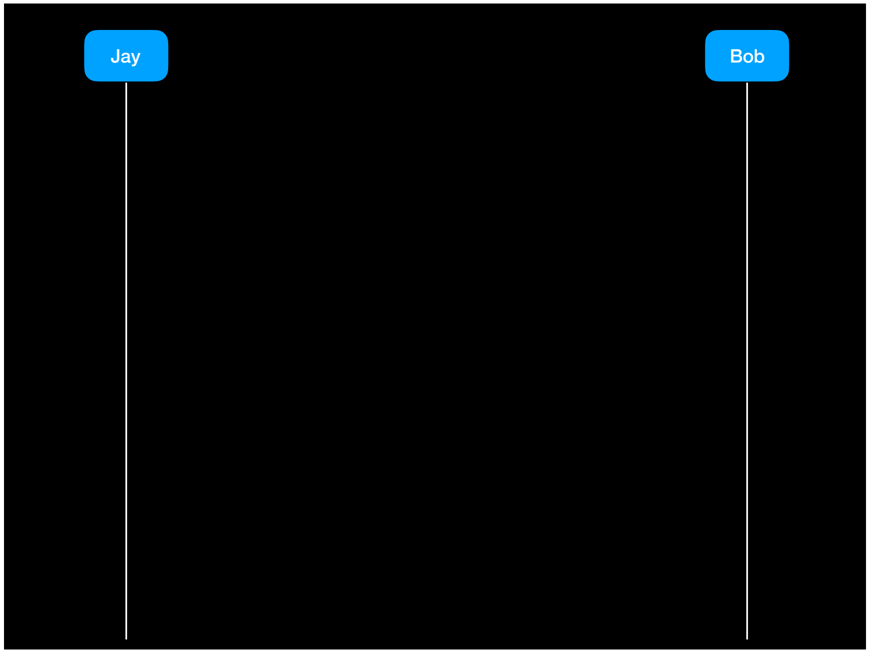


I never looked at a single line of code!

Real Life “Design Flaw”

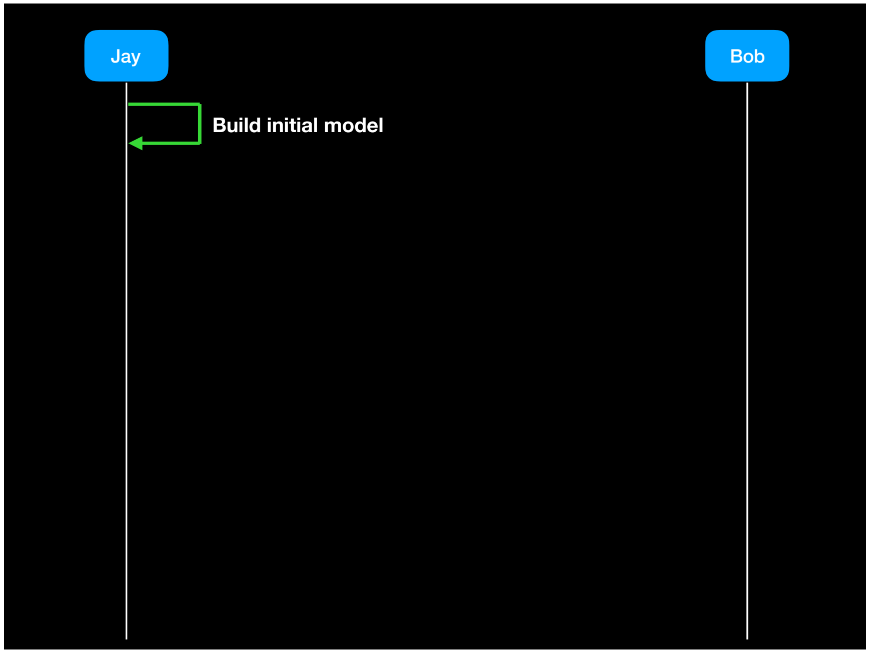


Model for understanding



Jay

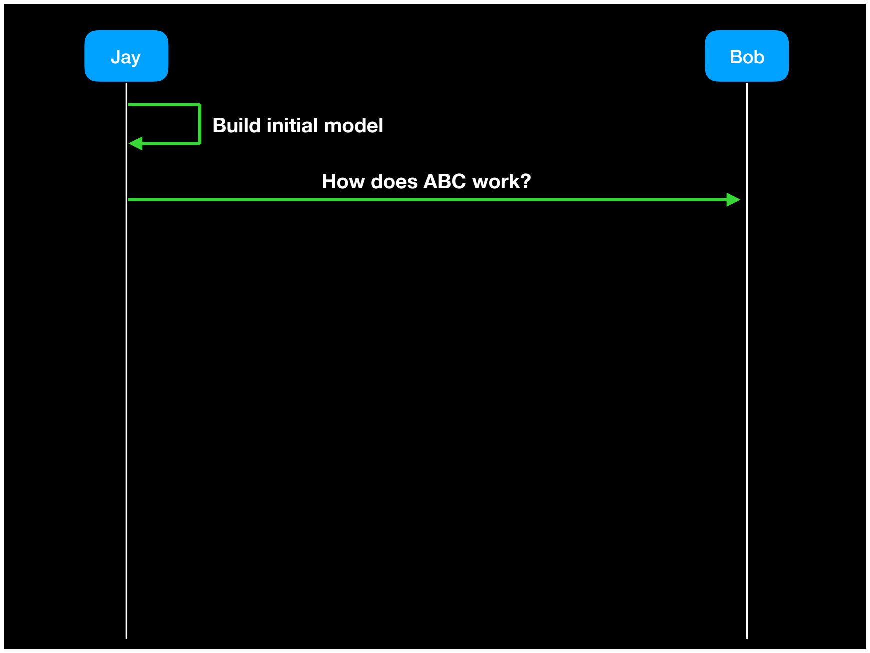
Bob



Jay

Bob

Build initial model

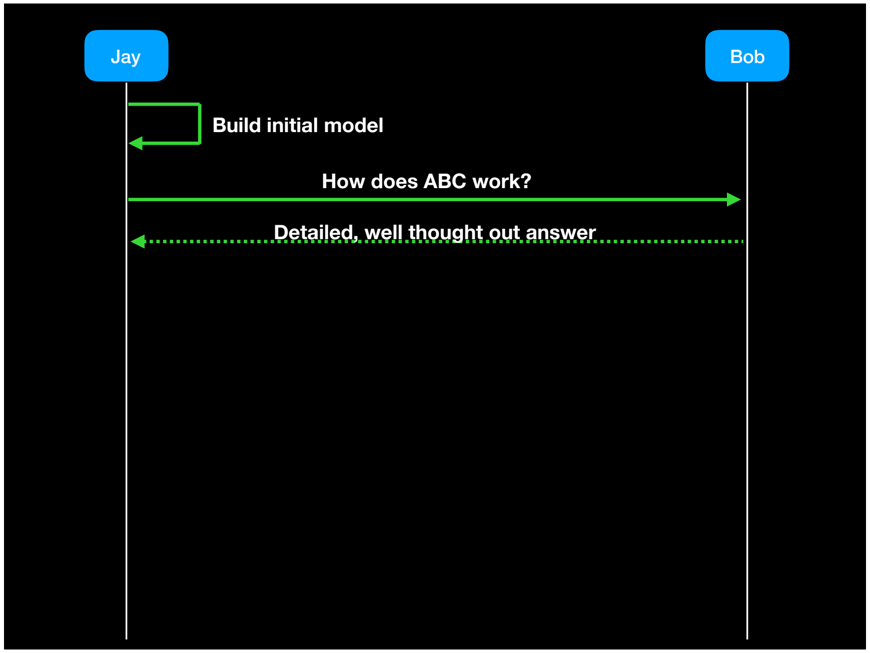


Jay

Bob

Build initial model

How does ABC work?



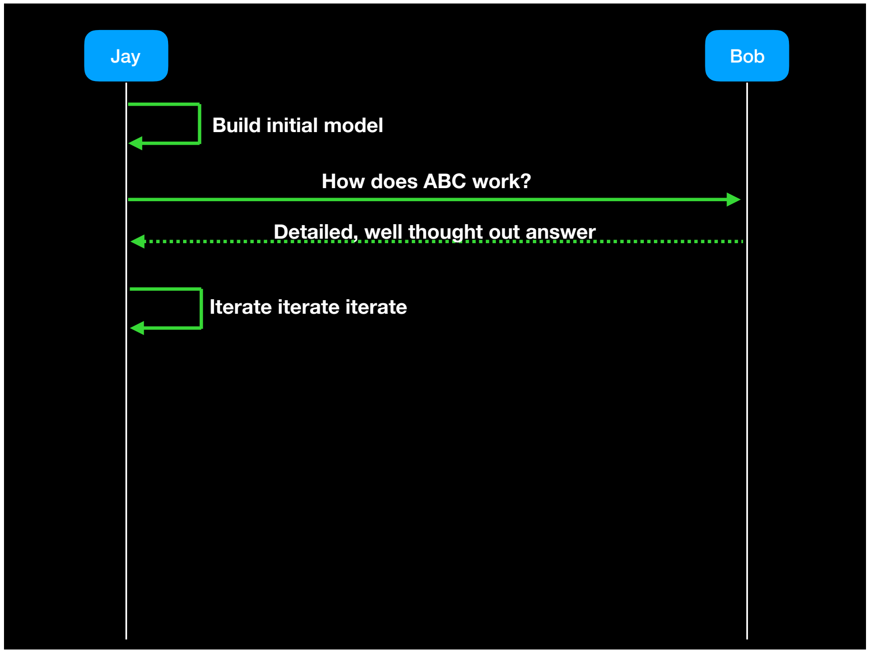
Jay

Bob

Build initial model

How does ABC work?

Detailed, well thought out answer



Jay

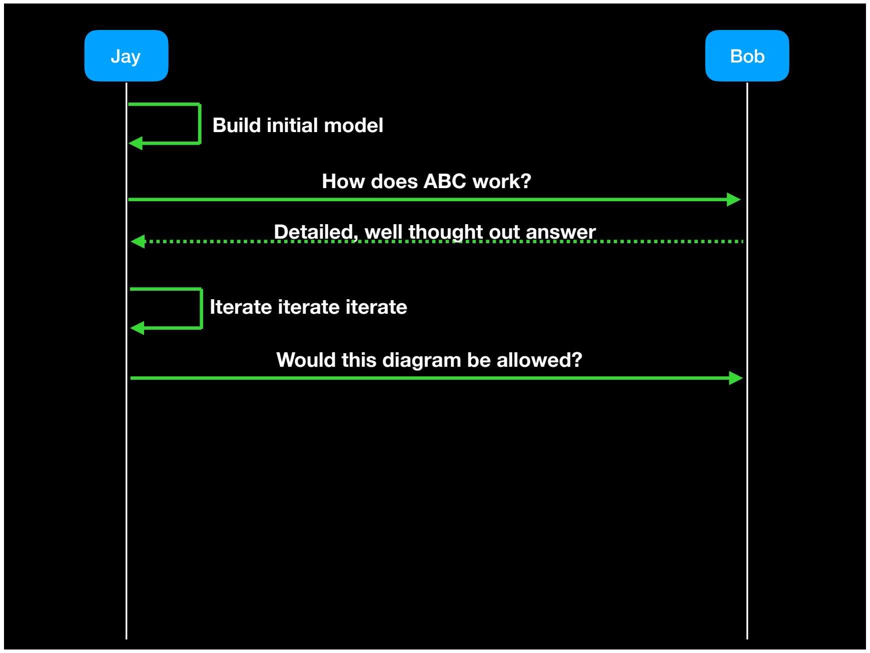
Bob

Build initial model

How does ABC work?

Detailed, well thought out answer

Iterate iterate iterate



Jay

Bob

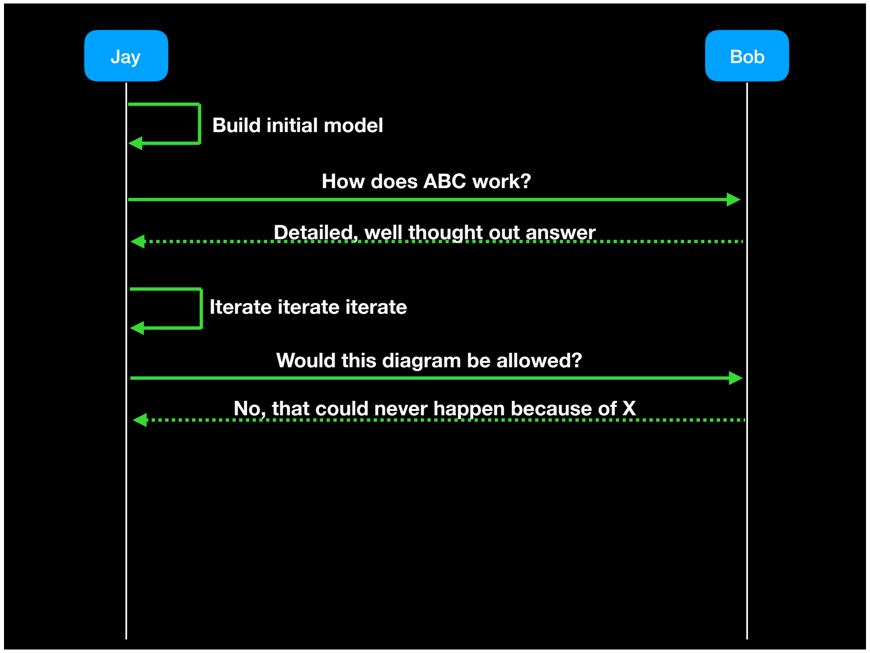
Build initial model

How does ABC work?

Detailed, well thought out answer

Iterate iterate iterate

Would this diagram be allowed?



Jay

Bob

Build initial model

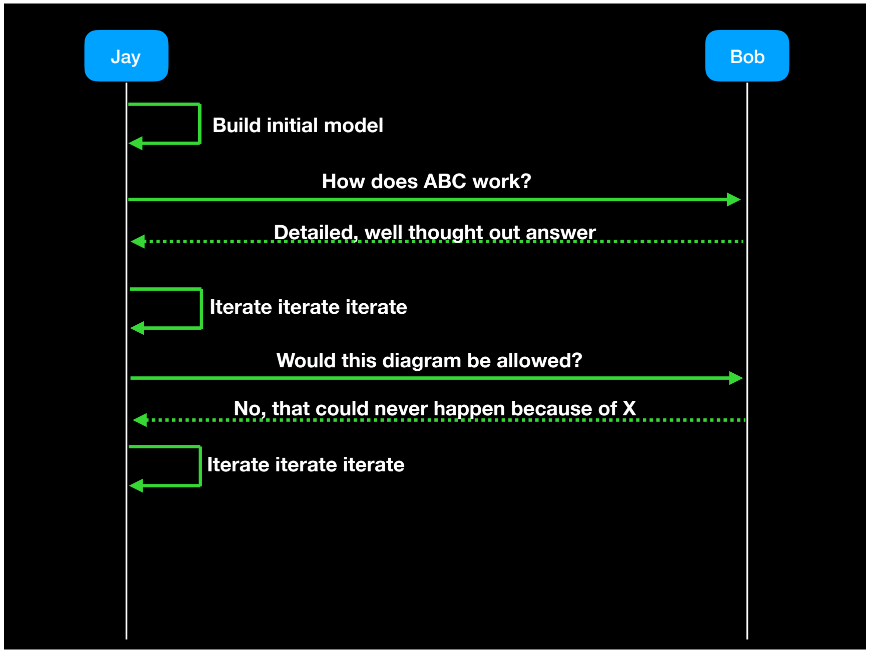
How does ABC work?

Detailed, well thought out answer

Iterate iterate iterate

Would this diagram be allowed?

No, that could never happen because of X



Jay

Bob

Build initial model

How does ABC work?

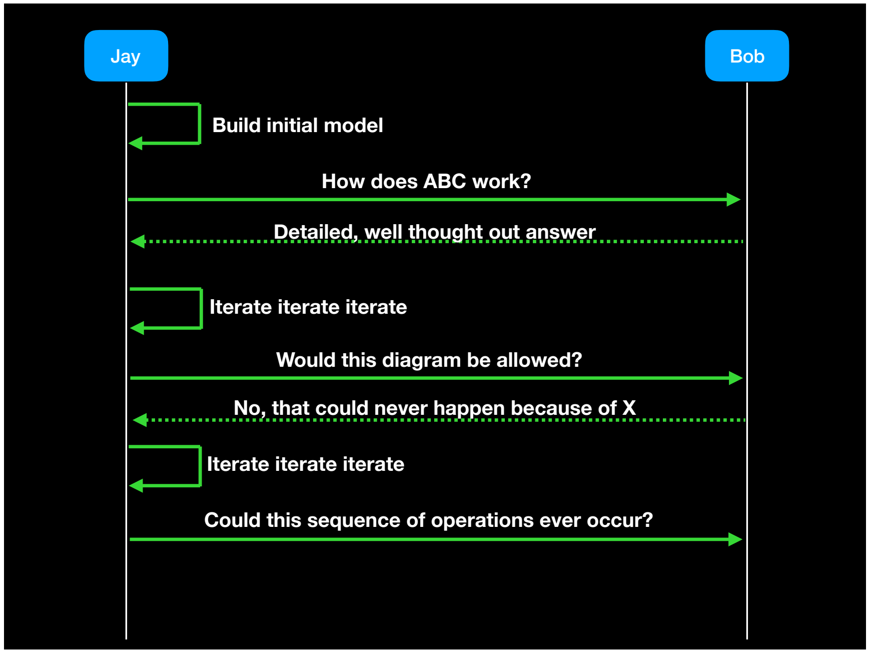
Detailed, well thought out answer

Iterate iterate iterate

Would this diagram be allowed?

No, that could never happen because of X

Iterate iterate iterate



Jay

Bob

Build initial model

How does ABC work?

Detailed, well thought out answer

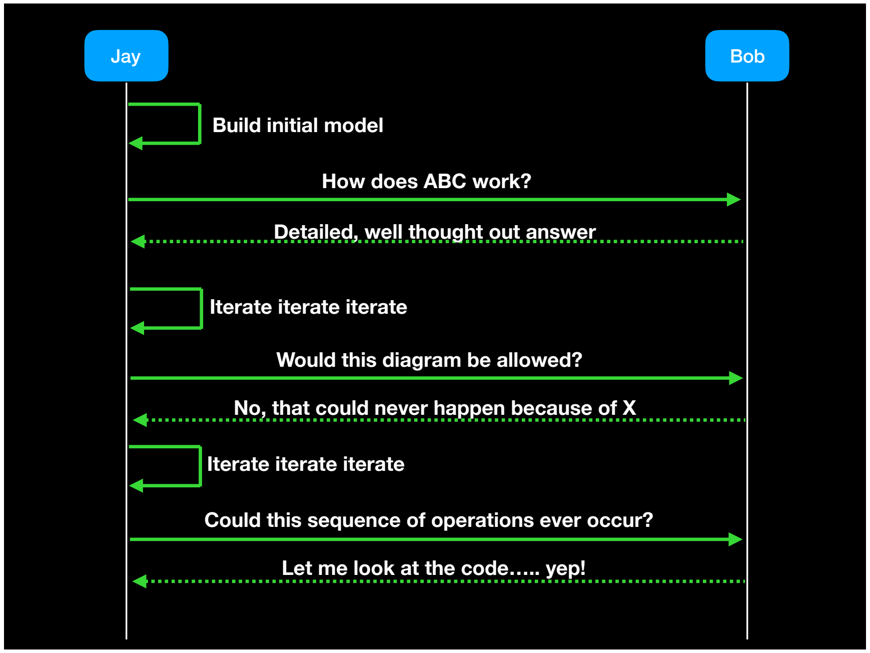
Iterate iterate iterate

Would this diagram be allowed?

No, that could never happen because of X

Iterate iterate iterate

Could this sequence of operations ever occur?



Jay

Bob

Build initial model

How does ABC work?

Detailed, well thought out answer

Iterate iterate iterate

Would this diagram be allowed?

No, that could never happen because of X

Iterate iterate iterate

Could this sequence of operations ever occur?

Let me look at the code….. yep!

The API won’t work!

Invalidating the Chord Protocol

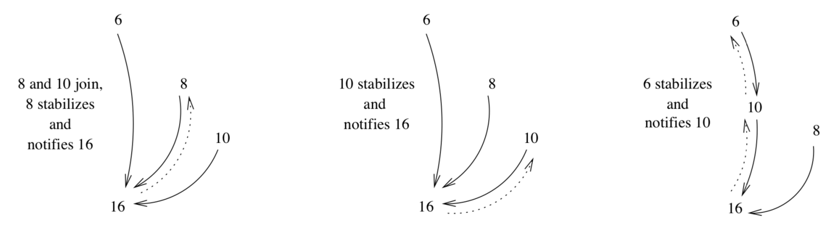


Figure 2: Three stages (left to right) creating a counterexample to *OrderedMerges*.

http://www.pamelazave.com/chord-ccr.pdf

Takeaways

Takeaways

- Write things down

Takeaways

- Write things down
- Model existing systems

Takeaways

- Write things down
- Model existing systems
- Model new systems

Resources

Resources

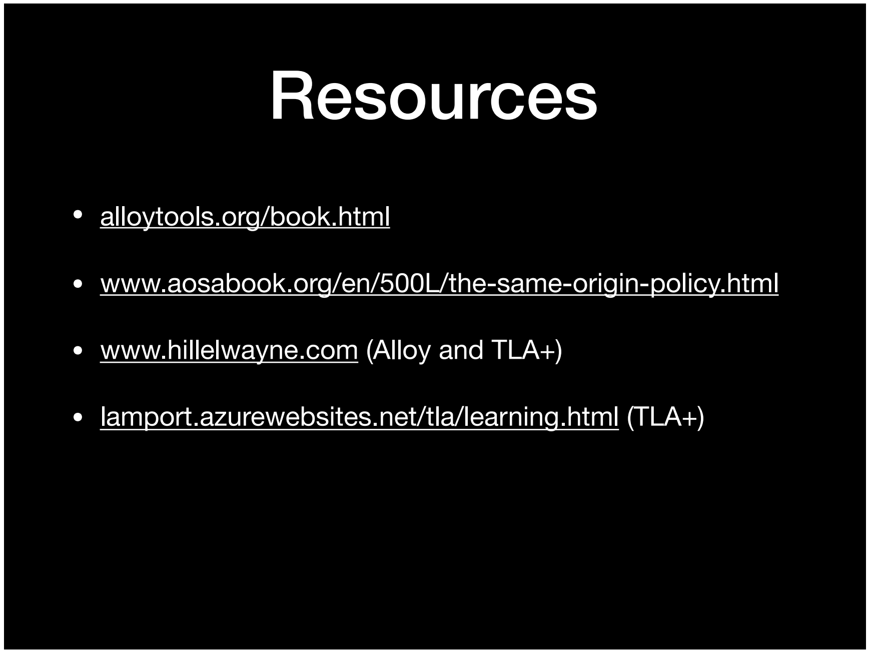
- alloytools.org/book.html

Resources

- alloytools.org/book.html
- www.aosabook.org/en/500L/the-same-origin-policy.html

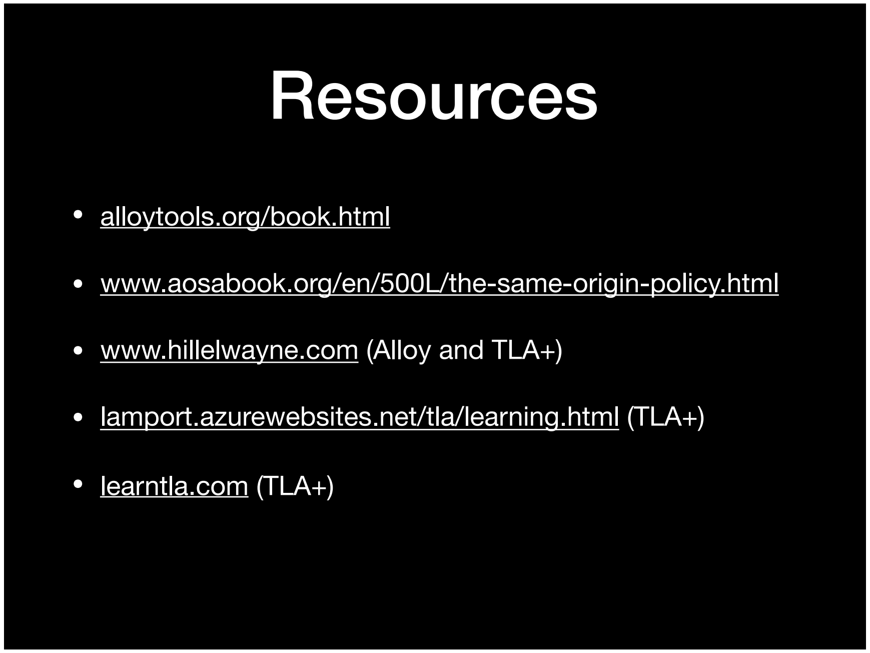
Resources

- alloytools.org/book.html
- www.aosabook.org/en/500L/the-same-origin-policy.html
- www.hillelwayne.com (Alloy and TLA+)



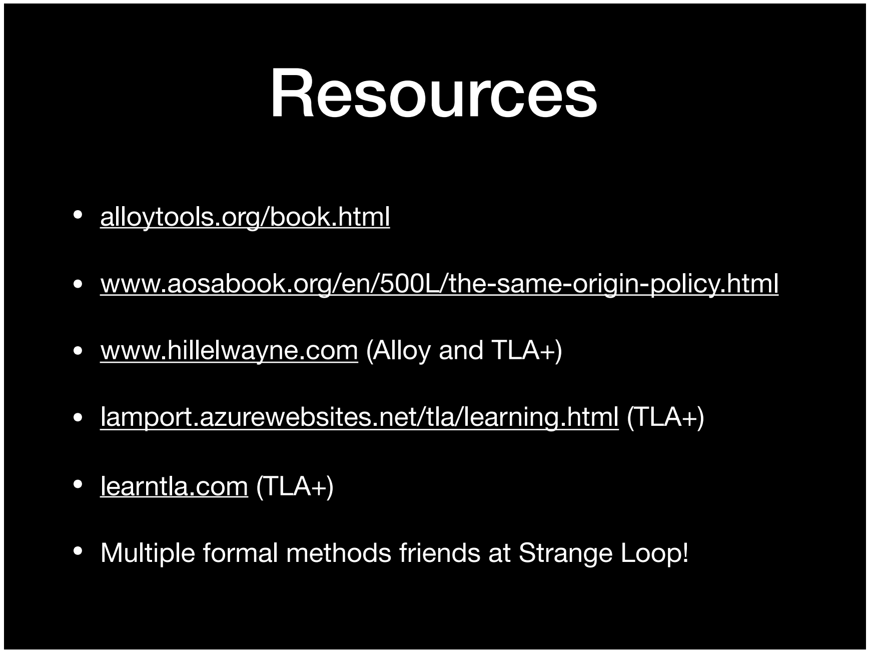
Resources

- alloytools.org/book.html
- www.aosabook.org/en/500L/the-same-origin-policy.html
- www.hillelwayne.com (Alloy and TLA+)
- lamport.azurewebsites.net/tla/learning.html (TLA+)



Resources

- alloytools.org/book.html
- www.aosabook.org/en/500L/the-same-origin-policy.html
- www.hillelwayne.com (Alloy and TLA+)
- lamport.azurewebsites.net/tla/learning.html (TLA+)
- learntla.com (TLA+)



Resources

- alloytools.org/book.html
- www.aosabook.org/en/500L/the-same-origin-policy.html
- www.hillelwayne.com (Alloy and TLA+)
- lamport.azurewebsites.net/tla/learning.html (TLA+)
- learntla.com (TLA+)
- Multiple formal methods friends at Strange Loop!