Understanding Git with Alloy

Milestone 2

Cláudio Lourenço Renato Neves

University of Minho
Formal Methods in Software Engineering

June 7, 2012

Table of contents

Where were we?

Progress

Our model right now

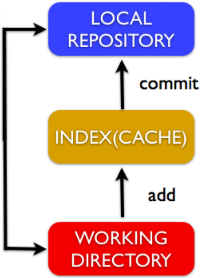
The operations

The properties

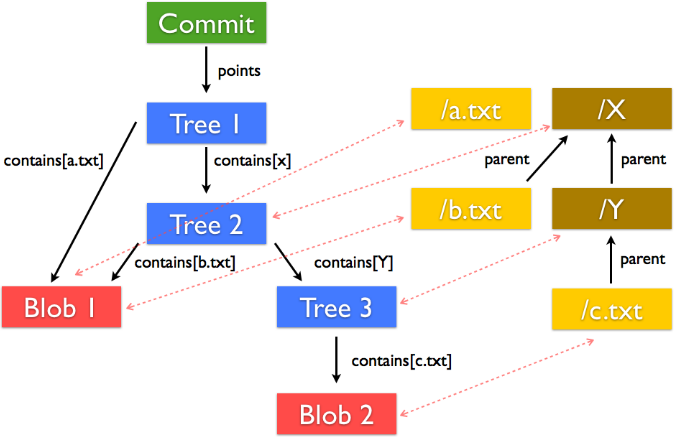
Future work

Where were we?

- Focusing on Index + Object Model
- No problem with add and rm operations
- Commit - Big Problem



The problem


Commit
points
Tree 1
contains[a.txt]
contains[x]
Tree 2
contains[b.txt]
contains[Y]
Blob 1
Tree 3
contains[c.txt]
Blob 2
/a.txt
/X
parent
parent
/b.txt
/Y
parent
/c.txt

How did we solve it?

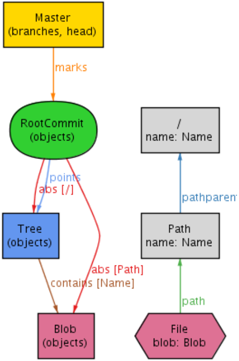
Abstraction

- In each commit there is relation between Objects and Paths
- Facts to reflect the parent relationship

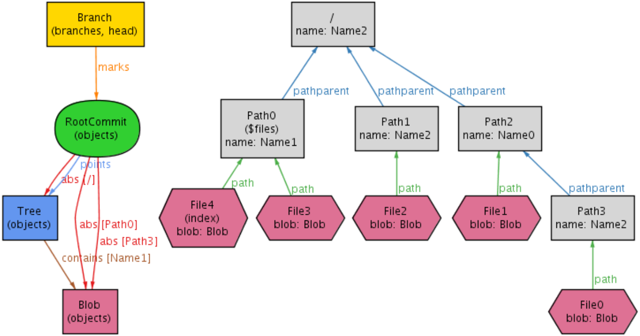
```
sig Commit extends Object {
    points : Tree,
    parent : set Commit,
    abs: Path -> Object
}
```

```
all c: Commit{
    let objs = c.points.*(contents.Name){
        c.abs in Path some -> lone objs
        (c.abs).(c.points) in Root
        all p,q : (c.abs).univ | p -> q in pathparent
            implies q.(c.abs) -> p.(c.abs) -> p.name in contents
        all t,o : objs, n : Name | t -> o -> n in contents
            implies all y : c.abs.t | some x : c.abs.o |
                x -> y in pathparent and x.name = n
    }
}
```

The abstraction relation



The abstraction relation



Model

```
abstract sig Object {
   objects: set State
}

sig Blob extends Object {}

sig Tree extends Object {
   contains : Name -> lone (Tree+Blob)
}

sig Commit extends Object {
   points : Tree,
   parent : set Commit,
   abs: Path -> Object
}

sig RootCommit extends Commit {}

sig Branch{
   marks: Commit one -> State,
   branches: set State,
   head: set State
}

lone sig Master extends Branch{}
```

```
sig Path {
   pathparent : lone Path,
   name : Name
}

sig File{
   path: Path,
   blob: Blob,
   index: set State
}
```

Operations

Properties

Future work

Understanding Git with Alloy

Milestone 2

Cláudio Lourenço Renato Neves

University of Minho
Formal Methods in Software Engineering

June 7, 2012