



# CloudLens

A scripting language to analyse semi-structured textual data

Guillaume Baudart, Louis Mandel,  
Olivier Tardieu, Mandana Vaziri

IBM T.J. Watson

Starting test Wsk SDK should download docker action sdk at 2016-07-14 17:17:14.017\n\n Starting test Wsk SDK should download iOS sdk at 2016-07-14 17:17:14.018\n\nsystem.basic.WskSdkTests > Wsk SDK should download docker action sdk PASSED\n\nsystem.basic.WskSdkTests STANDARD\_OUT\n\n Finished test Wsk SDK should download iOS sdk at 2016-07-14 17:17:14.113\n\n Starting test Wsk SDK should preview swift sdk at 2016-07-14 17:17:14.140\n\nsystem.basic.WskSdkTests > Wsk SDK should preview swift sdk PASSED\n\nsystem.basic.WskSdkTests STANDARD\_OUT\n\n Finished test Wsk SDK should preview swift sdk at 2016-07-14 17:17:14.140\n\n Starting test Swift 3 Whisk backend API should allow Swift actions to invoke other actions at 2016-07-14 17:17:14.164\n\nsystem.basic.Swift3WhiskObjectTests STANDARD\_OUT\n\n Starting test Swift 3 Whisk backend API should allow Swift actions to invoke other actions STANDARD\_OUT\n\n deleting invokeAction\n\nsystem.basic.Swift3WhiskObjectTests > Swift 3 Whisk backend API should allow Swift actions to invoke other actions PASSED\n\nsystem.basic.Swift3WhiskObjectTests STANDARD\_OUT\n\n Finished test Swift 3 Whisk backend API should allow Swift actions to invoke other actions at 2016-07-14 17:17:22.069\n\n Starting test Swift 3 Whisk backend API should allow Swift actions to trigger events at 2016-07-14 17:17:22.070\n\nsystem.basic.Swift3WhiskObjectTests > Swift 3 Whisk backend API should allow Swift actions to trigger events STANDARD\_OUT\n\n deleting TestTrigger 1468516642072\n\nsystem.basic.Swift3WhiskObjectTests > Swift 3 Whisk backend API should allow Swift actions to trigger events PASSED\n\nsystem.basic.Swift3WhiskObjectTests STANDARD\_OUT\n\n Finished test Swift 3 Whisk backend API should allow Swift actions to trigger events at 2016-07-14 17:17:29.981\n\nsystem.basic.CLIJavaTests STANDARD\_OUT\n\n Starting test Java Actions should invoke a java action at 2016-07-14 17:17:29.993\n\nsystem.basic.CLIJavaTests > Java Actions should invoke a java action STANDARD\_OUT\n\n deleting helloJava\n\nsystem.basic.CLIJavaTests > Java Actions should invoke a java action PASSED\n\nsystem.basic.CLIJavaTests STANDARD\_OUT\n\n Finished test Java Actions should invoke a java action at 2016-07-14 17:17:36.380\n\nsystem.basic.CLIPythonTests STANDARD\_OUT\n\n Starting test Native Python Action should invoke a blocking action and get the result at 2016-07-14 17:17:36.392\n\nsystem.basic.CLIPythonTests > Native Python Action should invoke a blocking action and get the result at 2016-07-14 17:17:37.821\n\nsystem.basic.WskActionSequenceTests STANDARD\_OUT\n\n Starting test Wsk Action Sequence should invoke a blocking action and get only the result at 2016-07-14 17:17:37.836\n\nsystem.basic.WskActionSequenceTests > Wsk Action Sequence should invoke a blocking action and get only the result STANDARD\_OUT\n\n deleting sequence\n\nsystem.basic.WskActionSequenceTests > Wsk Action Sequence should invoke a blocking action and get only the result PASSED\n\nsystem.basic.WskActionSequenceTests STANDARD\_OUT\n\n Finished test Wsk Action Sequence should invoke a blocking action and get only the result at 2016-07-14 17:17:48.279\n\nsystem.basic.ActionTests STANDARD\_OUT\n\n Starting test Actions CLI should error with a proper warning if the action exceeds its time limits at 2016-07-14 17:17:48.289\n\nsystem.basic.ActionTests > Actions CLI should error with a proper warning if the action exceeds its time limits STANDARD\_OUT\n\n deleting TestActionCausingNoTimeout\n\nsystem.basic.ActionTests > Actions CLI should error with a proper warning if the action exceeds its time limits PASSED\n\nsystem.basic.ActionTests STANDARD\_OUT\n\n Finished test Actions CLI should error with a proper warning if the action exceeds its time limits at 2016-07-14 17:18:01.396\n\n Starting test Actions CLI should succeed on an action staying within its time limits at 2016-07-14 17:18:01.397\n\nsystem.basic.ActionTests > Actions CLI should succeed on an action staying within its time limits STANDARD\_OUT\n\n deleting TestActionCausingNoTimeout\n\nsystem.basic.ActionTests > Actions CLI should succeed on an action staying within its time limits PASSED\n\nsystem.basic.ActionTests STANDARD\_OUT\n\n Finished test Actions CLI should succeed on an action staying within its time limits at 2016-07-14 17:18:12.340\n\nsystem.basic.PackageTests STANDARD\_OUT\n\n Starting test Wsk Package should confirm wsk exists at 2016-07-14 17:18:12.354\n\nsystem.basic.PackageTests > Wsk Package should confirm wsk exists PASSED\n\nsystem.basic.PackageTests STANDARD\_OUT\n\n Finished test Wsk Package should confirm wsk exists at 2016-07-14 17:18:12.360\n\n Starting test Wsk Package should allow creation and deletion of a package at 2016-07-14 17:18:12.361\n\nsystem.basic.PackageTests > Wsk Package should allow creation and deletion of a package STANDARD\_OUT\n\n deleting simplepackage\n\nsystem.basic.PackageTests > Wsk Package should allow creation and deletion of a package PASSED\n\nsystem.basic.PackageTests STANDARD\_OUT\n\n Finished test Wsk Package should allow creation and deletion of a package at 2016-07-14 17:18:12.689\n\n Starting test Wsk Package should allow creation of a package with parameters at 2016-07-14 17:18:12.689\n\nsystem.basic.PackageTests > Wsk Package should allow creation of a package with parameters STANDARD\_OUT\n\n deleting simplepackagewithparams\n\nsystem.basic.PackageTests > Wsk Package should allow creation of a package with parameters PASSED\n\nsystem.basic.PackageTests STANDARD\_OUT\n\n Finished test Wsk Package should allow creation of a package with parameters at 2016-07-14 17:18:13.001\n\n Starting test Wsk Package should allow updating a package at 2016-07-14 17:18:13.002\n\nsystem.basic.PackageTests > Wsk Package should allow updating a package STANDARD\_OUT\n\n deleting simplepackagetoupdate\n\nsystem.basic.PackageTests > Wsk Package should allow updating a package PASSED\n\nsystem.basic.PackageTests STANDARD\_OUT\n\n Finished test Wsk Package should allow updating a package at 2016-07-14 17:18:13.376\n\n Starting test Wsk Package should allow binding of a package at 2016-07-14 17:18:13.377\n\nsystem.basic.PackageTests > Wsk Package should allow binding of a package STANDARD\_OUT\n\n deleting simplebind\n\n deleting simplepackagetobind\n\nsystem.basic.PackageTests > Wsk Package should allow binding of a package PASSED\n\nsystem.basic.PackageTests STANDARD\_OUT\n\n Finished test Wsk Package should allow binding of a package at 2016-07-14 17:18:13.906\n\n Starting test Wsk Package should perform package binds so parameters are inherited at 2016-07-14 17:18:13.907\n\nsystem.basic.PackageTests > Wsk Package should perform package binds so parameters are inherited STANDARD\_OUT\n\n deleting simplepackage\n\nsystem.basic.PackageTests > Wsk Package should perform package binds so parameters are inherited PASSED\n\nsystem.basic.PackageTests STANDARD\_OUT\n\n Finished test Wsk Package should perform package binds so parameters are inherited at 2016-07-14 17:18:17.054\n\nsystem.basic.CLISquentialTests > ruleDeletedAction STANDARD\_OUT\n\n Thread 0. Running part 1 at 1468516698174\n\n Thread 2. Running part 1 at 1468516698175\n\n Thread 3. Running part 1 at 1468516698177\n\n Thread 4. Running part 1 at 1468516698184\n\n Thread 0. Running part 2 at 1468516730507\n\n Thread 2. Running part 2 at 1468516730568\n\n Thread 3. Running part 2 at 1468516730589\n\n Thread 1. Running part 2 at 1468516730686\n\n Thread 4. Done at 1468516732488\n\n Thread 3. Done at 1468516732533\n\n Thread 2. Done at 1468516732591\n\n Thread 1. Done at 1468516732601\n\n Now running unrelated activation at 1468516737601\n\n Local activation 1468516737601 stdout: hello A\_normal\_payload!\n\n\nsystem.basic.CLISquentialTests > ruleDeletedAction PASSED\n\nsystem.basic.CLIActionTests STANDARD\_OUT\n\n ParallelRunner: 1 threads.\n\nsystem.basic.CLIActionTests > parameterBinding PASSED\n\nsystem.basic.CLIActionTests > testPingNotAllowed PASSED\n\nsystem.basic.CLIActionTests > helloWorldDemo PASSED\n\nsystem.basic.CLIActionTests > updateAction PASSED\n\nsystem.basic.CLIActionTests > copyAction PASSED\n\nsystem.basic.CLIActionTests > invokeAction PASSED\n\nsystem.basic.CLIActionTests > invokeNestedBlockingAction PASSED\n\nsystem.basic.CLIActionTests > invokeActionWithNoCode PASSED\n\nsystem.basic.CLIActionTests > applicationError PASSED\n\nsystem.basic.CLIActionTests > invokeMonkeyAsyncDoneTwice PASSED\n\nsystem.basic.CLIActionTests > createActionWithNoCode PASSED\n\nsystem.basic.CLIActionTests > incorrectActionInvoke PASSED\n\nsystem.basic.CLIActionTests > invokeActionWithSpace PASSED\n\nsystem.basic.CLIActionTests > createActionWithSpecialCharacters SKIPPED\n\nsystem.basic.CLIActionTests > recreateAndInvokeAction PASSED\n\nsystem.basic.CLIActionTests > helloWorldDemoWithSpace PASSED\n\nsystem.basic.CLIActionTests > invokeMonkeySyncDoneTwice PASSED\n\nsystem.basic.CLIActionTests > invokeAsyncAction PASSED\n\nsystem.basic.CLISwiftTests STANDARD\_OUT\n\n Starting test Swift Actions should invoke a swift action STANDARD\_OUT\n\n deleting helloSwift\n\nsystem.basic.CLISwiftTests > Swift Actions should invoke a swift action PASSED\n\nsystem.basic.CLISwiftTests STANDARD\_OUT\n\n Finished test Swift Actions should invoke a swift action at 2016-07-14 17:20:03.385\n\n Starting test Swift Actions should invoke a swift:3 action at 2016-07-14 17:20:03.385\n\nsystem.basic.CLISwiftTests > Swift Actions should invoke a swift:3 action STANDARD\_OUT\n\n deleting helloSwift3\n\nsystem.basic.CLISwiftTests > Swift Actions should invoke a swift:3 action PASSED\n\nsystem.basic.CLISwiftTests STANDARD\_OUT\n\n Finished test Swift Actions should invoke a swift:3 action at 2016-07-14 17:20:10.020\n\nsystem.basic.WskBasicNodeTests STANDARD\_OUT\n\n Starting test NodeJS runtime should Map a kind of nodejs:default to the current default NodeJS runtime at 2016-07-14 17:20:10.033\n\nsystem.basic.WskBasicNodeTests > NodeJS runtime should Map a kind of nodejs:default to the current default NodeJS runtime STANDARD\_OUT\n\n deleting usingDefaultNodeAlias\n\nsystem.basic.WskBasicNodeTests > NodeJS runtime should Map a kind of nodejs:default to the current default NodeJS runtime PASSED\n\nsystem.basic.WskBasicNodeTests STANDARD\_OUT\n\n Finished test NodeJS runtime should Map a kind of nodejs:default to the current default NodeJS runtime at 2016-07-14 17:20:10.358\n\n Starting test NodeJS runtime should Ensure that JS actions created with no explicit kind use the current default NodeJS runtime at 2016-07-14 17:20:10.359\n\nsystem.basic.WskBasicNodeTests > NodeJS runtime should Ensure that JS actions created with no explicit kind use the current default NodeJS runtime STANDARD\_OUT\n\n deleting jsWithNoKindSpecified\n\nsystem.basic.WskBasicNodeTests > NodeJS runtime should Ensure that JS actions created with no explicit kind use the current default NodeJS runtime PASSED\n\nsystem.basic.WskBasicNodeTests STANDARD\_OUT\n\n Finished test NodeJS runtime should Ensure that JS actions created with no explicit kind use the current default NodeJS runtime at 2016-07-14 17:20:10.635\n\nsystem.basic.WskBasicTests STANDARD\_OUT\n\n Starting test Wsk CLI should confirm wsk exists at 2016-07-14 17:20:10.635\n\nsystem.basic.WskBasicTests > Wsk CLI should confirm wsk exists PASSED\n\nsystem.basic.WskBasicTests STANDARD\_OUT\n\n Finished test Wsk CLI should confirm wsk exists at 2016-07-14 17:20:10.639\n\n Starting test Wsk CLI should show help and usage info at 2016-07-14 17:20:10.639\n\nsystem.basic.WskBasicTests > Wsk CLI should show help and usage info PASSED\n\nsystem.basic.WskBasicTests STANDARD\_OUT\n\n Finished test Wsk CLI should show help and usage info at 2016-07-14 17:20:10.661\n\nsystem.basic.WskBasicTests > Wsk CLI should show cli build version at 2016-07-14 17:20:10.661\n\nsystem.basic.WskBasicTests > Wsk CLI should show cli build version PASSED\n\nsystem.basic.WskBasicTests STANDARD\_OUT\n\n Finished test Wsk CLI should show cli build version at 2016-07-14 17:20:10.673\n\n Starting test Wsk CLI should show api version at 2016-07-14 17:20:10.673\n\nsystem.basic.WskBasicTests > Wsk CLI should show api version STANDARD\_OUT\n\n deleting testWskCliBuildVersion\n\nsystem.basic.WskBasicTests > Wsk CLI should show api version PASSED\n\nsystem.basic.WskBasicTests STANDARD\_OUT\n\n Finished test Wsk CLI should show api build version at 2016-07-14 17:20:10.692\n\n Starting test Wsk CLI should show api build number at 2016-07-14 17:20:10.692\n\nsystem.basic.WskBasicTests > Wsk CLI should show api build number STANDARD\_OUT\n\n deleting testWskCliBuildNumber\n\nsystem.basic.WskBasicTests > Wsk CLI should show api build number PASSED\n\nsystem.basic.WskBasicTests STANDARD\_OUT\n\n Finished test Wsk CLI should show api build number at 2016-07-14 17:20:11.075\n\n Starting test Wsk CLI should set auth in property file at 2016-07-14 17:20:11.076\n\nsystem.basic.WskBasicTests > Wsk CLI should set auth in property file STANDARD\_OUT\n\n deleting testWskCliSetAuth\n\nsystem.basic.WskBasicTests > Wsk CLI should set auth in property file PASSED\n\nsystem.basic.WskBasicTests STANDARD\_OUT\n\n Finished test Wsk CLI should set auth in property file at 2016-07-14 17:20:11.088\n\n Starting test Wsk CLI should reject creating duplicate entities at 2016-07-14 17:20:11.088\n\nsystem.basic.WskBasicTests > Wsk CLI should reject creating duplicate entity STANDARD\_OUT\n\n sanitizing testDuplicateCreate\n\n deleting testDuplicateCreate\n\nsystem.basic.WskBasicTests > Wsk CLI should reject creating duplicate entity PASSED\n\nsystem.basic.WskBasicTests STANDARD\_OUT\n\n Finished test Wsk CLI should reject creating duplicate entity at 2016-07-14 17:20:11.544\n\n Starting test Wsk CLI should reject deleting entity in wrong collection at 2016-07-14 17:20:11.545\n\nsystem.basic.WskBasicTests > Wsk CLI should reject deleting entity in wrong collection STANDARD\_OUT\n\n deleting testCrossDelete\n\nsystem.basic.WskBasicTests > Wsk CLI should reject deleting entity in wrong collection PASSED\n\nsystem.basic.WskBasicTests STANDARD\_OUT\n\n Finished test Wsk CLI should reject deleting entity in wrong collection at 2016-07-14 17:20:11.858\n\nsystem.basic.WskBasicTests > Wsk CLI should reject creating entities with invalid names at 2016-07-14 17:20:11.858\n\nsystem.basic.WskBasicTests > Wsk CLI should reject creating entities with invalid names PASSED\n\nsystem.basic.WskBasicTests STANDARD\_OUT\n\n Finished test Wsk CLI should reject creating entities with invalid names at 2016-07-14 17:20:11.858

```
Starting test Wsk SDK should download action sdk at 2016-07-14 17:17:14.017\n\n Finished test Wsk SDK should download ios sdk at 2016-07-14 17:17:14.113\n\n Starting test Wsk SDK should preview swift sdk at 2016-07-14 17:17:14.140\n\n Finished test Wsk SDK should preview swift sdk at 2016-07-14 17:17:14.140\n\n Starting test Swift 3 Whisk backend API should allow Swift actions to invoke other actions at 2016-07-14 17:17:14.164\n\n Finished test Swift 3 Whisk backend API should allow Swift actions to invoke other actions at 2016-07-14 17:17:22.069\n\n Starting test Swift 3 Whisk backend API should allow Swift actions to trigger events at 2016-07-14 17:17:29.981\n\n Finished test Swift 3 Whisk backend API should allow Swift actions to trigger events at 2016-07-14 17:17:29.993\n\n Starting test Wsk Package should confirm wsk exists at 2016-07-14 17:18:12.354\n\n Finished test Wsk Package should confirm wsk exists at 2016-07-14 17:18:12.360\n\n Starting test Wsk Package should allow creation and deletion of a package at 2016-07-14 17:18:12.689\n\n Finished test Wsk Package should allow creation and deletion of a package at 2016-07-14 17:18:13.001\n\n Starting test Wsk Package should allow updating a package at 2016-07-14 17:18:13.376\n\n Finished test Wsk Package should allow updating a package at 2016-07-14 17:18:13.377\n\n Starting test Wsk Package should allow binding of a package at 2016-07-14 17:18:13.906\n\n Finished test Wsk Package should allow binding of a package at 2016-07-14 17:18:13.907\n\n Starting test Wsk Package should perform package binds so parameters are inherited at 2016-07-14 17:18:17.054\n\n Finished test Wsk Package should perform package binds so parameters are inherited at 2016-07-14 17:18:17.054\n\n Starting test Wsk CLI should confirm wsk exists at 2016-07-14 17:20:10.635\n\n Finished test Wsk CLI should confirm wsk exists at 2016-07-14 17:20:10.639\n\n Starting test Wsk CLI should show help and usage info at 2016-07-14 17:20:10.661\n\n Finished test Wsk CLI should show help and usage info at 2016-07-14 17:20:10.673\n\n Starting test Wsk CLI should show api build version at 2016-07-14 17:20:10.692\n\n Finished test Wsk CLI should show api build version at 2016-07-14 17:20:10.912\n\n Starting test Wsk CLI should show api build number at 2016-07-14 17:20:11.075\n\n Finished test Wsk CLI should show api build number at 2016-07-14 17:20:11.076\n\n Starting test Wsk CLI should set auth in property file at 2016-07-14 17:20:11.088\n\n Finished test Wsk CLI should set auth in property file at 2016-07-14 17:20:11.544\n\n Starting test Wsk CLI should reject creating duplicate entity at 2016-07-14 17:20:11.545\n\n Finished test Wsk CLI should reject creating duplicate entity at 2016-07-14 17:20:11.545\n\n Starting test Wsk CLI should reject deleting entity in wrong collection at 2016-07-14 17:20:11.858\n\n Finished test Wsk CLI should reject deleting entity in wrong collection at 2016-07-14 17:20:11.858
```



# CloudLens

## Goal

- Monitoring: report alerting state
- Troubleshooting: find causes of a crash

## Challenges

- Lots of semi-structured data
- Format can change over time





# Programming Model

- **source**
  - get input data
- **group**
  - re-structure data
- **match**
  - extract structure
- **stream**
  - react when match
- **block, restart**
  - triggered only once
- Repeat and compose

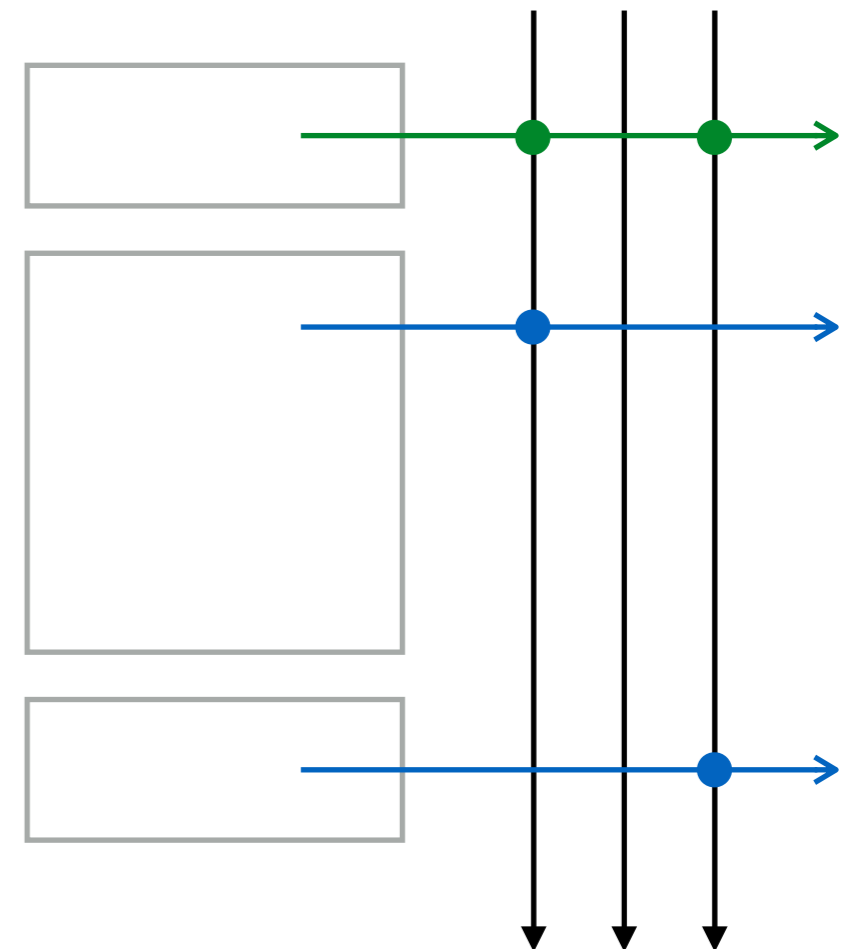
*disk, web, databases...*

*regex*

*regex*

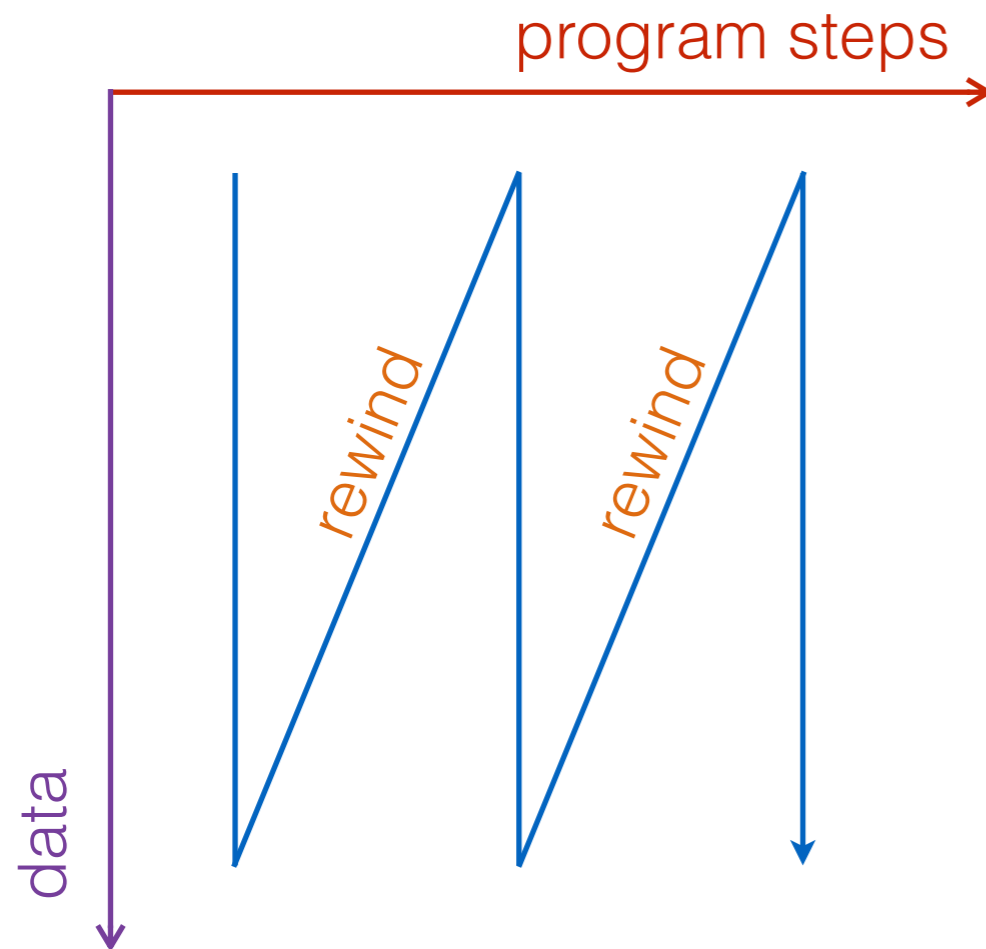
*JavaScript*

*JavaScript*

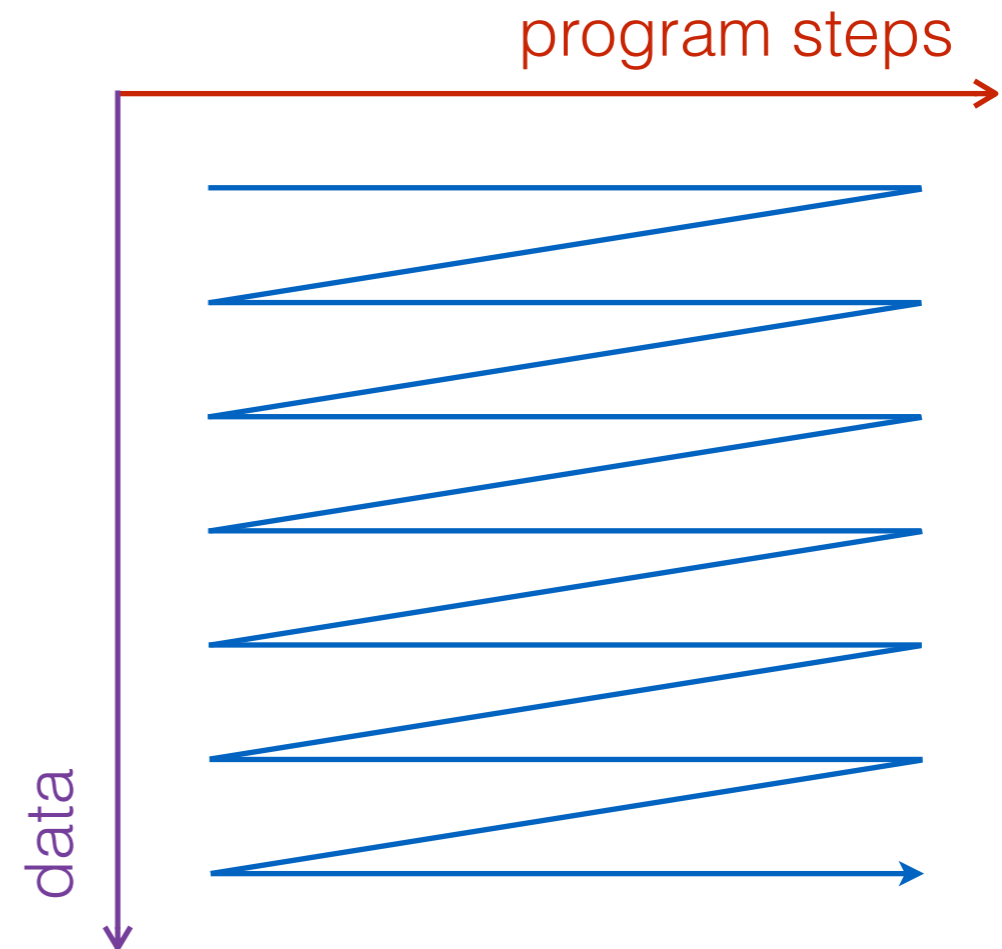


# Repeat and Compose

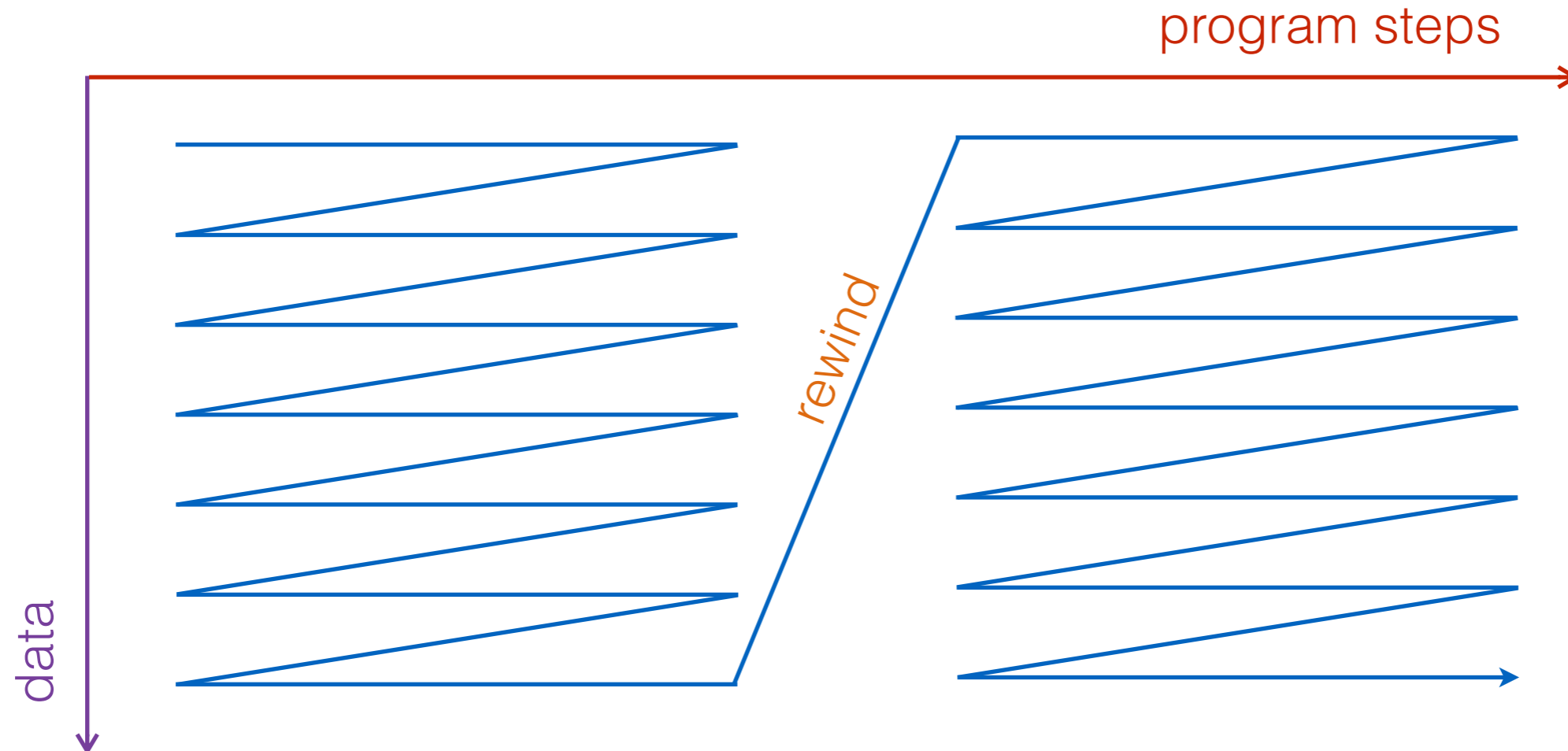
Serial composition



Ordered parallel composition



# Repeat and Compose



CloudLens: default parallel composition + explicit **restart**



```
match {  
    "(?<failed>.*) > .* FAILED"  
}  
  
var failed = 0;  
  
stream when (entry.failed) {  
    print("FAILED:", entry.failed)  
    failed++;  
}  
  
{ print(failed, "failed tests") }
```

```
match {  
    "(?<failed>.*) > .* FAILED"  
}  
  
var failed = 0;  
  
stream {  
    print("FAILED:", entry.failed)  
    failed++;  
}  
  
{ print(failed, "failed tests") }
```

```
var dateFormat = "Date[yyyy-MM-dd' 'HH:mm:ss.SSS]";

match {
    "Starting test (?<desc>.*) at (?<start>.*)# start:" + dateFormat;
    "Finished test (?<desc>.*) at (?<end>.*)# end:" + dateFormat
}

var start;

stream {
    start = entry.start;
}

stream {
    entry.dur = entry.end - start;
    if (entry.dur > 12000) {
        print(entry.dur, "\t", entry.desc);
    }
}
```

# Hierarchy

```
system.basic.WskBasicTests > Wsk Action CLI should reject delete of action that does not exist FAILED
org.scalatest.exceptions.TestFailedException: "error: Unable to delete action: Request failure: The requested resource does not exist. (code 914)
" did not include substring that matched regex error: The requested resource does not exist. \d+
    at org.scalatest.MatchersHelper$.newTestFailedException(MatchersHelper.scala:160)
    at org.scalatest.Matchers$ResultOfIncludeWordForString.regex(Matchers.scala:2201)
    at org.scalatest.Matchers$ResultOfIncludeWordForString.regex(Matchers.scala:2173)
    at system.basic.WskBasicTests$$anonfun$25.apply$mcV$sp(WskBasicTests.scala:295)
    at system.basic.WskBasicTests$$anonfun$25.apply(WskBasicTests.scala:295)
    at system.basic.WskBasicTests$$anonfun$25.apply(WskBasicTests.scala:295)
    at org.scalatest.Transformer$$anonfun$apply$1.apply$mcV$sp(Transformer.scala:22)
    at org.scalatest.OutcomeOf$class.outcomeOf(OutcomeOf.scala:85)
    at org.scalatest.OutcomeOf.outcomeOf(OutcomeOf.scala:104)
    at org.scalatest.Transformer.apply(Transformer.scala:22)
    at org.scalatest.Transformer.apply(Transformer.scala:20)
    at org.scalatest.FlatSpecLike$$anon$1.apply(FlatSpecLike.scala:1647)
    at org.scalatest.Suite$class.withFixture(Suite.scala:1122)
    at org.scalatest.FlatSpec.withFixture(FlatSpec.scala:1683)
    at org.scalatest.FlatSpecLike$class.invokeWithFixture$1(FlatSpecLike.scala:1644)
    at org.scalatest.FlatSpecLike$$anonfun$runTest$1.apply(FlatSpecLike.scala:1656)
    at org.scalatest.FlatSpecLike$$anonfun$runTest$1.apply(FlatSpecLike.scala:1656)
    at org.scalatest.SuperEngine.runTestImpl(Engine.scala:306)
    at org.scalatest.FlatSpecLike$class.runTest(FlatSpecLike.scala:1656)
    at system.basic.WskBasicTests.org$scalatest$BeforeAndAfterEachTestData$$super$runTest(WskBasicTests.scala:50)
    at org.scalatest.BeforeAndAfterEachTestData$class.runTest(BeforeAndAfterEachTestData.scala:193)
    at system.basic.WskBasicTests.runTest(WskBasicTests.scala:50)
```

# Hierarchy

```
system.basic.WskBasicTests > Wsk Action CLI should reject delete of action that does not exist FAILED
org.scalatest.exceptions.TestFailedException: "error: Unable to delete action: Request failure: The requested resource does not exist. (code 914)
" did not include substring that matched regex error: The requested resource does not exist. \(code \d+\)
    at org.scalatest.MatchersHelper$.newTestFailedException(MatchersHelper.scala:160)
    at org.scalatest.Matchers$ResultOfIncludeWordForString.regex(Matchers.scala:2201)
    at org.scalatest.Matchers$ResultOfIncludeWordForString.regex(Matchers.scala:2173)
    at system.basic.WskBasicTests$$anonfun$25.apply$mcV$sp(WskBasicTests.scala:295)
    at system.basic.WskBasicTests$$anonfun$25.apply(WskBasicTests.scala:295)
    at system.basic.WskBasicTests$$anonfun$25.apply(WskBasicTests.scala:295)
    at org.scalatest.Transformer$$anonfun$apply$1.apply$mcV$sp(Transformer.scala:22)
    at org.scalatest.OutcomeOf$class.outcomeOf(OutcomeOf.scala:85)
    at org.scalatest.OutcomeOf$.outcomeOf(OutcomeOf.scala:104)
    at org.scalatest.Transformer.apply(Transformer.scala:22)
    at org.scalatest.Transformer.apply(Transformer.scala:20)
    at org.scalatest.FlatSpecLike$$anon$1.apply(FlatSpecLike.scala:1647)
    at org.scalatest.Suite$class.withFixture(Suite.scala:1122)
    at org.scalatest.FlatSpec.withFixture(FlatSpec.scala:1683)
    at org.scalatest.FlatSpecLike$class.invokeWithFixture$1(FlatSpecLike.scala:1644)
    at org.scalatest.FlatSpecLike$$anonfun$runTest$1.apply(FlatSpecLike.scala:1656)
    at org.scalatest.FlatSpecLike$$anonfun$runTest$1.apply(FlatSpecLike.scala:1656)
    at org.scalatest.SuperEngine.runTestImpl(Engine.scala:306)
    at org.scalatest.FlatSpecLike$class.runTest(FlatSpecLike.scala:1656)
    at system.basic.WskBasicTests.org$scalatest$BeforeAndAfterEachTestData$$super$runTest(WskBasicTests.scala:50)
    at org.scalatest.BeforeAndAfterEachTestData$class.runTest(BeforeAndAfterEachTestData.scala:193)
    at system.basic.WskBasicTests.runTest(WskBasicTests.scala:50)
```

- **source**
  - stream of JSON object
- **group**
  - combine consecutive entry into arrays
- **lens**
  - define CloudLens functions

```

match {
    "(?<failed>.* ) > .* FAILED";
}

group {
    "^[^ ]"
}

lens stackCheck() {
    match {
        "at .*\\((?<whisk>Wsk.*)\\)";
    }

    stream (line) {
        print('    at', line.whisk)
    }
}

stream when (entry.failed) {
    print("FAILED", entry.failed);
    stackCheck(entry.group)
}

```

```
lens testStart () {
  match {
    "Starting test (?<start>.*) at (?<date>.*)"
  }

  stream {
    print("Starting", entry.start)
  }
}
```

```
lens testStop() {
  match {
    "Finished test (?<end>.*) at (?<date>.*)"
  }

  stream {
    print("Finished", entry.end)
  }
}
```

```
{ testStart();
  testStop() }
```

```
lens testStart () {  
  match {  
    "Starting test (?<start>.*) at (?<date>.*)"  
  }  
  
  stream {  
    print("Starting", entry.start)  
  }  
}
```

```
lens testStop() {  
  match {  
    "Finished test (?<end>.*) at (?<date>.*)"  
  }  
  
  stream {  
    print("Finished", entry.end)  
  }  
}
```

```
run testStart()  
run testStop()
```



# Formal Semantics

## Stage

- **restart** and **block**, or
- Pipeline
  - succession of **group**, **match**, and **stream**

Program execution  $M, \ell \vdash \text{script} \Longrightarrow M', \ell'$

Stage elaboration  $M, \ell, p \vdash \text{script} \Longrightarrow_s M', \ell'$

Stage execution  $M, \ell \vdash p \Longrightarrow_p M', \ell'$

$$M, \ell, [] \vdash \text{script} \Longrightarrow_s M', \ell'$$

---

$$M, \ell \vdash \text{script} \Longrightarrow M', \ell'$$

$$M, \ell, p :: \text{match } \{ \text{patterns} \} \vdash \text{script} \Longrightarrow_s M', \ell'$$

---

$$M, \ell, p \vdash \text{match } \{ \text{patterns} \} \text{script} \Longrightarrow_s M', \ell'$$

$$M, \ell \vdash p \Longrightarrow_p M', \ell'$$

---

$$M, \ell, p \vdash \emptyset \Longrightarrow_s M', \ell'$$

$$M, \ell \vdash p \Longrightarrow_p M_p, \ell_p \quad M_p, \ell_p, [] \vdash \text{script} \Longrightarrow_s M', \ell'$$

---

$$M, \ell, p \vdash \text{restart script} \Longrightarrow_s M', \ell'$$

# Implementation

## Java 8 and Javascript

- Popular programming language
- Nashorn runtime in the JVM
- Fast prototyping

## Two execution modes

- *Monitoring*: on-the fly processing
- *Troubleshooting*: table processing

## Closely follows the semantics

- 1) Build stages (block and pipelines)
- 2) Execute stages

Handle environment with JavaScript closures

# Execution

Each section returns a list of continuations

```
lens f(x) {  
  stream {a1(x)}  
  {a2(x)}  
  stream {a3(x)}  
}
```

```
stream {p1}  
run f(42)  
stream {p2}
```

# Execution

Each section returns a list of continuations

```
lens f(x) {                                script
  stream {a1(x)}
  {a2(x)}
  stream {a3(x)}
}

stream {p1}
run f(42)
stream {p2}
```

# Execution

Each section returns a list of continuations

```
lens f(x) {  
  stream {a1(x)}  
  {a2(x)}  
  stream {a3(x)}  
}
```

```
stream {p1}  
run f(42)  
stream {p2}
```

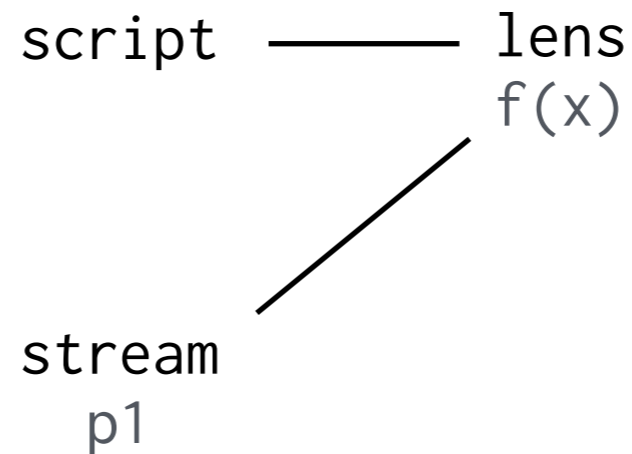
script — lens  
f(x)

# Execution

Each section returns a list of continuations

```
lens f(x) {  
  stream {a1(x)}  
  {a2(x)}  
  stream {a3(x)}  
}
```

```
stream {p1}  
run f(42)  
stream {p2}
```

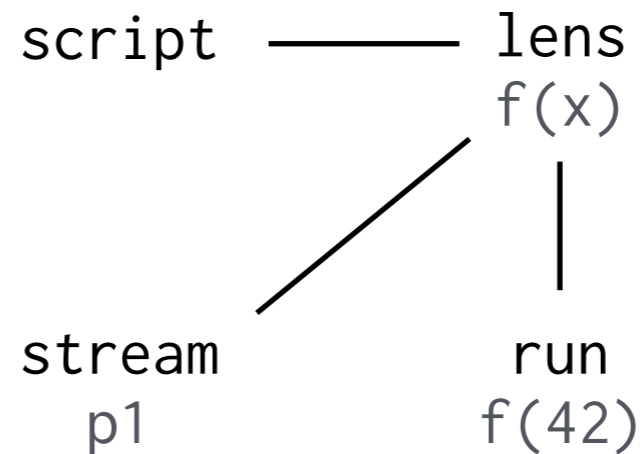


# Execution

Each section returns a list of continuations

```
lens f(x) {  
  stream {a1(x)}  
  {a2(x)}  
  stream {a3(x)}  
}
```

```
stream {p1}  
run f(42)  
stream {p2}
```

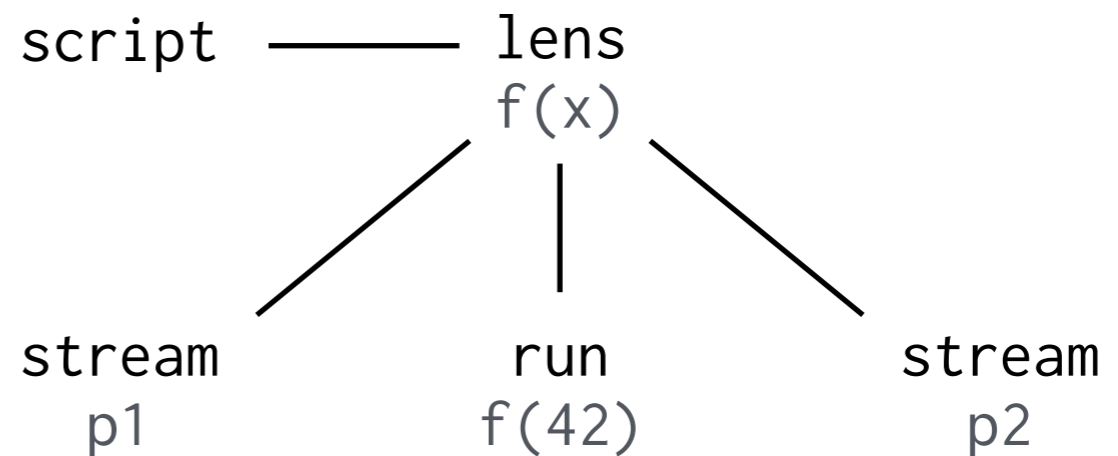


# Execution

Each section returns a list of continuations

```
lens f(x) {  
  stream {a1(x)}  
  {a2(x)}  
  stream {a3(x)}  
}
```

```
stream {p1}  
run f(42)  
stream {p2}
```



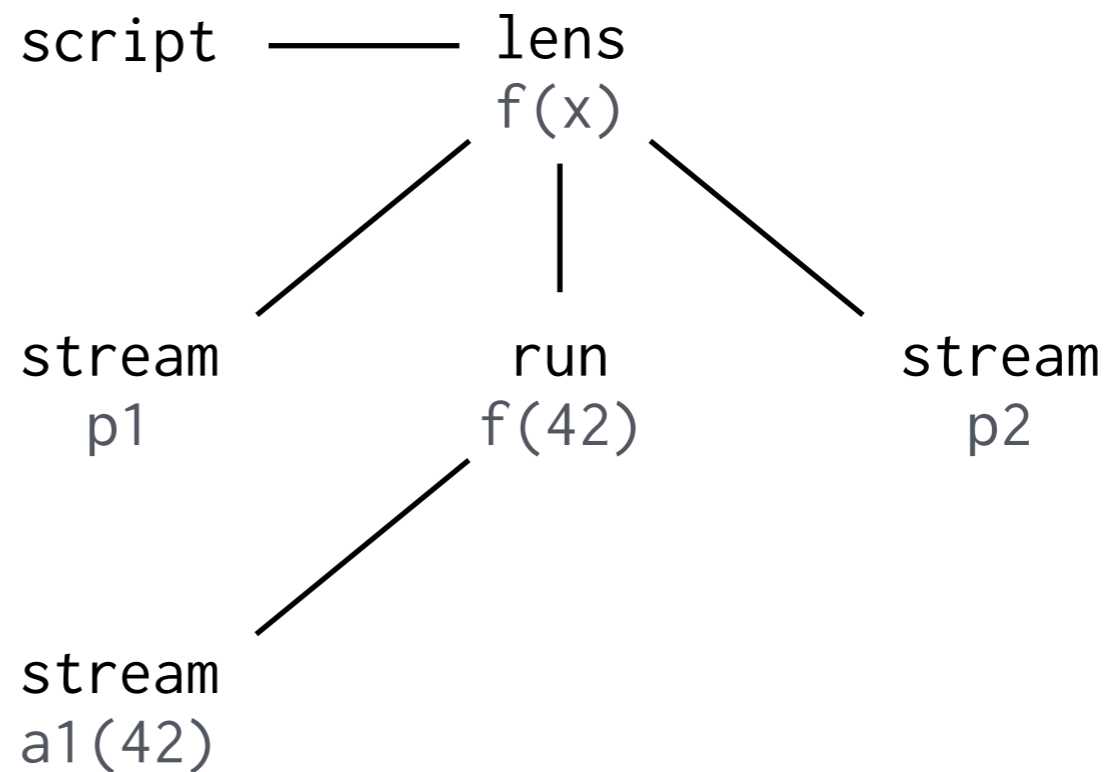


# Execution

Each section returns a list of continuations

```
lens f(x) {  
  stream {a1(x)}  
  {a2(x)}  
  stream {a3(x)}  
}
```

```
stream {p1}  
run f(42)  
stream {p2}
```

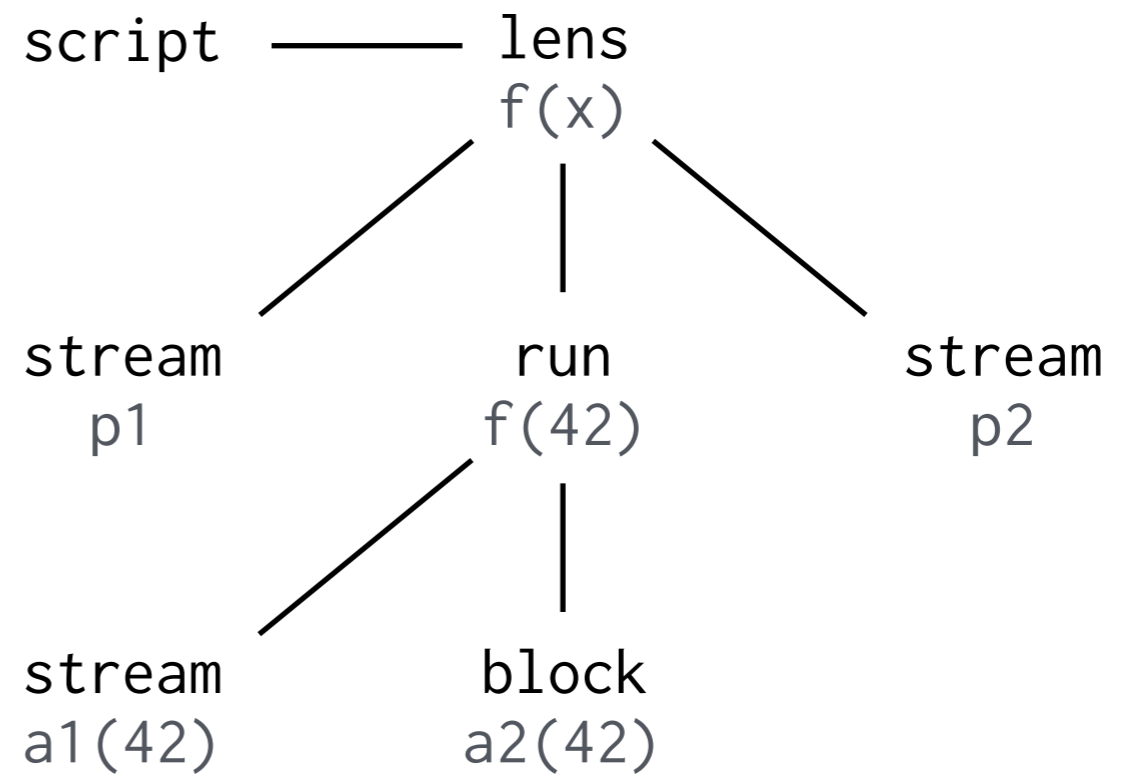


# Execution

Each section returns a list of continuations

```
lens f(x) {  
  stream {a1(x)}  
  {a2(x)}  
  stream {a3(x)}  
}
```

```
stream {p1}  
run f(42)  
stream {p2}
```

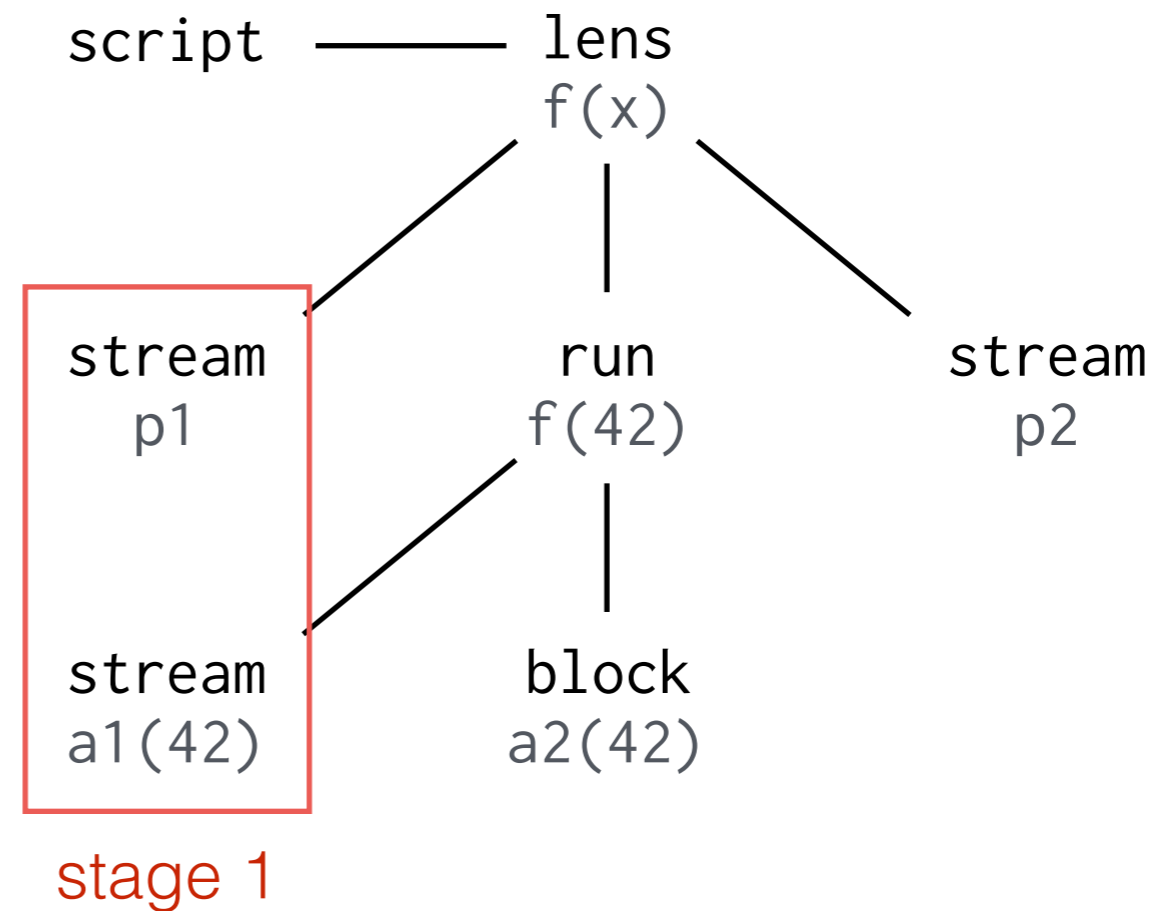


# Execution

Each section returns a list of continuations

```
lens f(x) {  
  stream {a1(x)}  
  {a2(x)}  
  stream {a3(x)}  
}
```

```
stream {p1}  
run f(42)  
stream {p2}
```

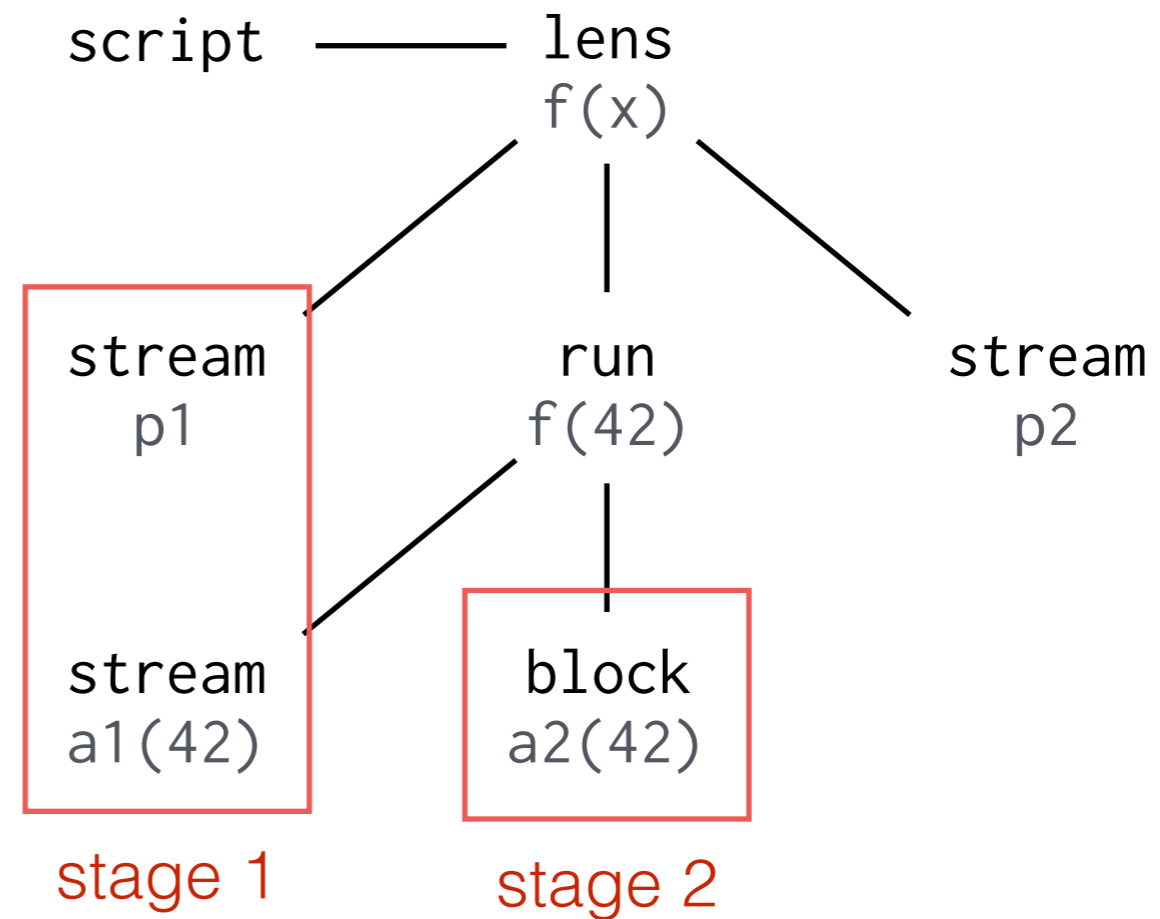


# Execution

Each section returns a list of continuations

```
lens f(x) {  
  stream {a1(x)}  
  {a2(x)}  
  stream {a3(x)}  
}
```

```
stream {p1}  
run f(42)  
stream {p2}
```

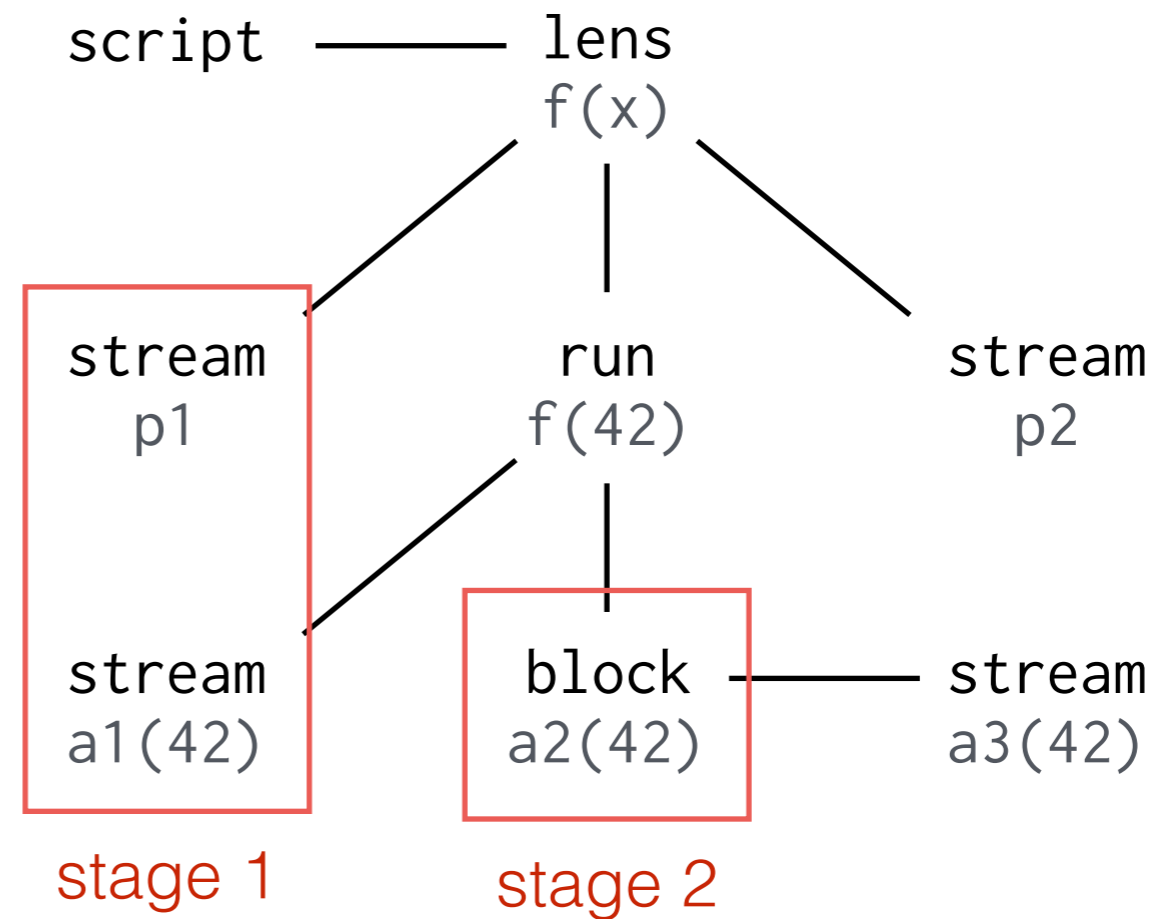


# Execution

Each section returns a list of continuations

```
lens f(x) {  
  stream {a1(x)}  
  {a2(x)}  
  stream {a3(x)}  
}
```

```
stream {p1}  
run f(42)  
stream {p2}
```

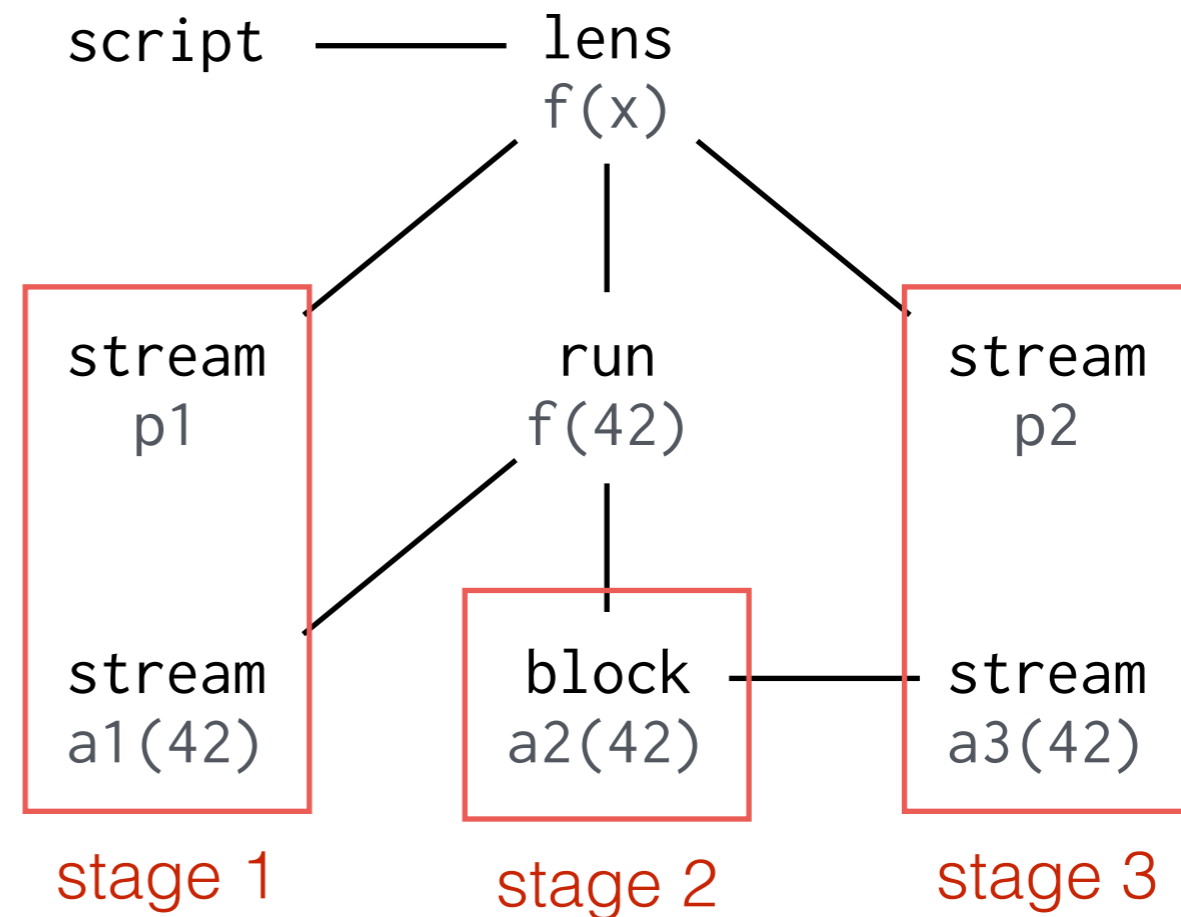


# Execution

Each section returns a list of continuations

```
lens f(x) {  
  stream {a1(x)}  
  {a2(x)}  
  stream {a3(x)}  
}
```

```
stream {p1}  
run f(42)  
stream {p2}
```



# Related Work

## **Programming languages and tools**

- grep + bash + vi + bc
- Awk, Perl
- JavaScript, Python, OCaml
- PADS [Fisher et al]

## **Databases**

- SQL, NoSQL
- CQL, Splunk

## **Machine learning**

- data format inference
- control flow inference

# Applications

## **IBM Bluemix OpenWhisk**

Logs from Travis

CloudLens as a whisk action...

...to analyze whisk builds




# Applications

## IBM Bluemix OpenWhisk

Logs from Travis

CloudLens as a whisk action...

...to analyze whisk builds

 **whiskbot** BOT 3:46 PM  
The build for PR #1023 failed  
Failed tests  
`system.basic.CLISwiftTests`  
Powered by CloudLens & OpenWhisk

 **whiskbot** BOT 4:26 PM  
The build for PR #1023 failed  
Failed tests  
`system.basic.WskBasicNodeTests`  
Powered by CloudLens & OpenWhisk

 **whiskbot** BOT 5:06 PM  
The build for PR #1023 failed  
Failed tests  
`limits.ThrottleTests`  
Powered by CloudLens & OpenWhisk

# Applications

## IBM Bluemix OpenWhisk



whiskbot BOT 3:46 PM

The build for PR #1023 failed



whiskbot BOT 4:26 PM

The build for PR #1023 failed

Failed tests

`system.basic.WskBasicNodeTests`

Powered by CloudLens & OpenWhisk


# Applications

## IBM Bluemix OpenWhisk

Logs from Travis

CloudLens as a whisk action...

...to analyze whisk builds

 **whiskbot** BOT 3:46 PM  
The build for PR #1023 failed  
Failed tests  
`system.basic.CLISwiftTests`  
Powered by CloudLens & OpenWhisk

 **whiskbot** BOT 4:26 PM  
The build for PR #1023 failed  
Failed tests  
`system.basic.WskBasicNodeTests`  
Powered by CloudLens & OpenWhisk

 **whiskbot** BOT 5:06 PM  
The build for PR #1023 failed  
Failed tests  
`limits.ThrottleTests`  
Powered by CloudLens & OpenWhisk

# Applications

## IBM Bluemix OpenWhisk

Logs from Travis


CloudLens as a whisk action...


...to analyze whisk builds

 **whiskbot** BOT 3:46 PM  
The build for PR #1023 failed  
Failed tests  
`system.basic.CLISwiftTests`  
Powered by CloudLens & OpenWhisk

 **whiskbot** BOT 4:26 PM  
The build for PR #1023 failed  
Failed tests  
`system.basic.WskBasicNodeTests`  
Powered by CloudLens & OpenWhisk

 **whiskbot** BOT 5:06 PM  
The build for PR #1023 failed  
Failed tests  
`limits.ThrottleTests`  
Powered by CloudLens & OpenWhisk

 **Rodric Rabbah**  
Nice  
I love this

 **Markus Thömmes** 9:28 AM  
yep awesomely awesome  
even telling me what went wrong

 **Justin Berstler** 3:10 PM  
[@psuter](#): Neato!

# Applications

## IBM Bluemix OpenWhisk

Logs from Travis

CloudLens as a whisk action...

...to analyze whisk builds

## Review code repositories

Comments analysis



whiskbot BOT 3:46 PM  
The build for PR #1023 failed  
Failed tests  
`system.basic.CLISwiftTests`  
Powered by CloudLens & OpenWhisk

whiskbot BOT 4:26 PM  
The build for PR #1023 failed  
Failed tests  
`system.basic.WskBasicNodeTests`  
Powered by CloudLens & OpenWhisk

whiskbot BOT 5:06 PM  
The build for PR #1023 failed  
Failed tests  
`limits.ThrottleTests`  
Powered by CloudLens & OpenWhisk



Rodric Rabbah  
Nice  
I love this

Markus Thömmes 9:28 AM  
yep awesomely awesome  
even telling me what went wrong

Justin Berstler 3:10 PM  
`@psuter`: Neato!

# Applications

## IBM Bluemix OpenWhisk

Logs from Travis

CloudLens as a whisk action...

...to analyze whisk builds

## Review code repositories

Comments analysis

## Notebook integration

Three screenshots of messages from a bot named "whiskbot". Each message reports a failed build for PR #1023 and lists the failed tests. The first message (3:46 PM) lists "system.basic.CLISwiftTests". The second message (4:26 PM) lists "system.basic.WskBasicNodeTests". The third message (5:06 PM) lists "limits.ThrottleTests". All messages are powered by CloudLens & OpenWhisk.

**whiskbot** BOT 3:46 PM  
The build for PR #1023 failed  
Failed tests  
`system.basic.CLISwiftTests`  
Powered by CloudLens & OpenWhisk

**whiskbot** BOT 4:26 PM  
The build for PR #1023 failed  
Failed tests  
`system.basic.WskBasicNodeTests`  
Powered by CloudLens & OpenWhisk

**whiskbot** BOT 5:06 PM  
The build for PR #1023 failed  
Failed tests  
`limits.ThrottleTests`  
Powered by CloudLens & OpenWhisk

Three social media comments in a thread. The first comment is from Markus Thömmes (9:28 AM) saying "yep awesomely awesome even telling me what went wrong". The second comment is from Rodric Rabbah saying "Nice" and "I love this". The third comment is from Justin Berstler (3:10 PM) saying "@psuter: Neato!".

**Markus Thömmes** 9:28 AM  
yep awesomely awesome  
even telling me what went wrong

**Rodric Rabbah**  
Nice  
I love this

**Justin Berstler** 3:10 PM  
[@psuter](#): Neato!



<http://cloudlens.github.io>