CloudLens

A Scripting Language to Analyze Semi-Structured Textual Data

Guillaume Baudart, Louis Mandel,
Olivier Tardieu, Mandana Vaziri
IBM T.J. Watson

Jamie Jennings
IBM Cloud

Example: system.log

```
Sep  8 07:55:32 serenity kernel[0]: smb1_smb_negotiate: Support for the server RECIFVMT has been deprecated (PreXP), disconnecting
Sep  8 07:56:10 serenity.watson.ibm.com helpd[263]: Couldn't find a URL for file named NAV_16.GIF in bundle at path /Applications/Symantec Solutions/Symantec Endpoint Protection.app/Contents/Resources/Symantec Endpoint Protection Help.help. Removing HPDBookIconPath from the Info.plist dictionary.
Sep  8 07:58:44 serenity.watson.ibm.com sshd[1085]: Accepted publickey for dgrove from 9.2.179.30 port 34176 ssh2
Sep  8 07:58:44 serenity.watson.ibm.com sshd[1077]: Accepted publickey for x10test from 9.12.246.2 port 35071 ssh2
Sep  8 07:58:46 serenity.watson.ibm.com sshd: dgrove [priv][1085]: USER_PROCESS: 1090 ttys000
Sep  8 07:58:46 serenity.watson.ibm.com sshd[1079]: subsystem request for sftp by user x10test
Sep  8 07:58:50 serenity.watson.ibm.com sshd: dgrove [priv][1085]: DEAD_PROCESS: 1090 ttys000
Sep  8 07:58:50 serenity.watson.ibm.com sshd[1090]: Received disconnect from 9.2.179.30: 11: disconnected by user
```

```
Sep  8 08:08:12 serenity kernel[0]: process SymDaemon[74] thread 1397 caught burning CPU! It used more than 50% CPU (Actual recent usage: 92%) over 180 seconds. thread lifetime cpu usage 216.745964 seconds, (214.741708 user, 2.004256 system) ledger info: balance: 90007639600 credit: 216485620939 debit: 126477981339 limit: 90000000000 (50%) period: 180000000000 time since last refill (ns): 97016076640
```

```
/Applications/Server.app/Contents/ServerRoot/System/Library/PrivateFrameworks/CSService.framework/Versions/A/CSService and
/Applications/Server.app/Contents/ServerRoot/usr/sbin/collabd. One of the two will be used. Which one is undefined.
Sep  8 08:10:09 serenity.watson.ibm.com com.apple.SecurityServer[22]: Session 100013 created
Sep  8 08:10:09 serenity.watson.ibm.com tccd[13121]: Failed to create /var/empty/Library/Application Support/com.apple.TCC (13)
Sep  8 08:10:10 serenity.watson.ibm.com distnoted[13135]: # distnote server agent  absolute time: 1921.255778348   civil time: Tue Sep  8 08:10:10 2015   pid: 13135 uid: 94 root: no
Sep  8 08:10:11 serenity.watson.ibm.com collabd[13040]: [main.m:341 72650310 +3ms] HTTP server listening at localhost:4444
Sep  8 08:10:11 serenity.watson.ibm.com collabd[13040]: [main.m:342 72650310 +0ms] HTTPS server listening at localhost:4443
Sep  8 08:10:11 serenity.watson.ibm.com collabd[13040]: [main.m:366 72650310 +0ms] Configured to exit after about 120 seconds idle
Sep  8 08:10:11 serenity.watson.ibm.com serveradmin[13142]: Validating Xcode.app at /Library/Developer/XcodeServer/CurrentXcodeSymlink
Sep  8 08:10:11 serenity.watson.ibm.com serveradmin[13142]: Xcode.app path does NOT exist, returning SERVERMGR_XCODE_XCODEPATH_UNKNOWN
Sep  8 08:10:11 serenity.watson.ibm.com serveradmin[13142]: readSettingsWithRequest: {
            configuration =     {
                validXcodeIsConfigured = 0;
            };
            readStatus = 0;
        }
```

IBM

CloudLens

- Semi-structured textual data processing without tiers
  - open source
  - lightweight
  - reactive
  - domain-specific language
  - for the analysis of semi-structured textual data
- Applications
  - log analysis
    - offline debugging
    - online monitoring
  - review source code repositories

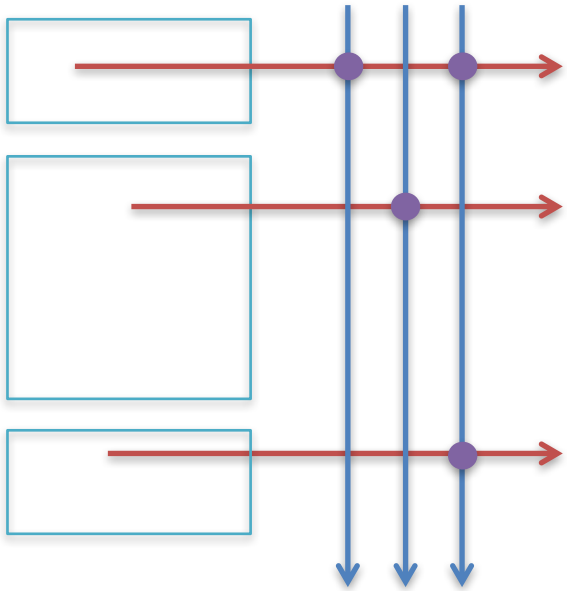
IBM

Outline

- CloudLens Tutorial
- Implementation
- Demo
- Formal Semantics
- Related Work

CloudLens Programming Model

- Source
  - get input data — *disk, web, database… (tables and streams)*
- Group
  - identify logical entities — *regex*
- Match
  - extract structured data — *regex*
- React
  - compute when match — *JavaScript*
- Repeat and compose



IBM

Example: system.lens

```
match {
  "Accepted publickey for (?<userConnect>\w+)";
  "sshd: (?<userDisconnect>\w+) .* DEAD_PROCESS: \d+ tty"
}

var users = 0;

stream (entry) when (entry.userConnect) {
  print("User", entry.userConnect, "connecting");
  users++;
  print(users, "users connected")
}

stream (entry) when (entry.userDisconnect) {
  print("User", entry.userDisconnect, "disconnecting");
  users--
}
```

Example: system.lens

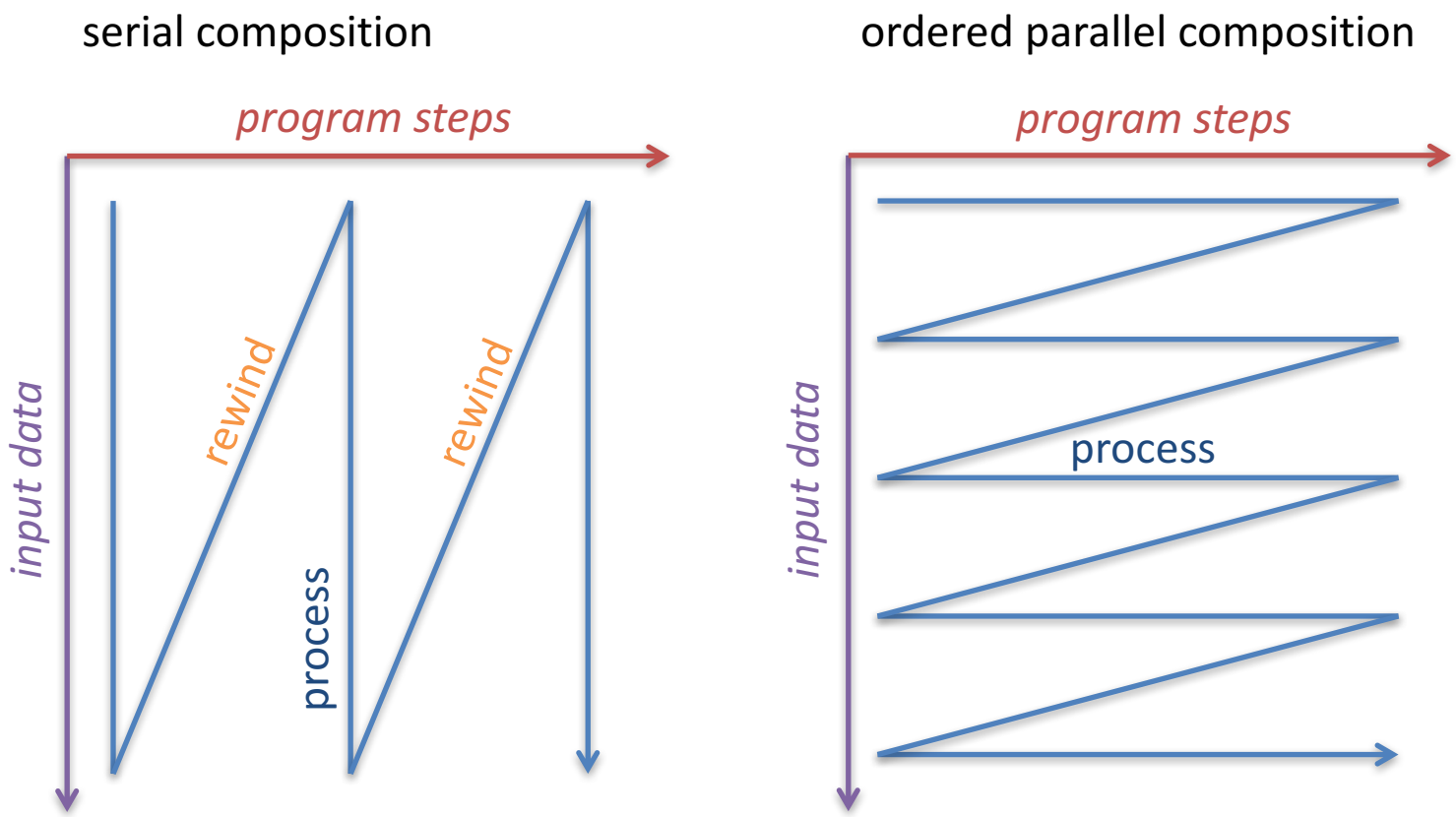
```
match {
  "Accepted publickey for (?<userConnect>\w+)";
  "sshd: (?<userDisconnect>\w+) .* DEAD_PROCESS: \d+ tty"
}

var users = 0;

stream                                          {
  print("User", entry.userConnect, "connecting");
  users++;
  print(users, "users connected")
}

stream                                            {
  print("User", entry.userDisconnect, "disconnecting");
  users--
}
```

Repeat and Compose



- Default to ordered parallel composition
  - restart construct to explicitly rewind stream

IBM

Hierarchies

- Data
  - stream of JSON objects
  - group construct
    - combine consecutive entries into arrays
- Processing
  - nested traversals
    - iterate over streams or tables
- Code
  - lens definitions
    - hybrid of macro and function

Example: openwhisk.log

```
Starting test Wsk Action CLI should reject delete of action that does not exist at 2016-07-14 17:20:15.176

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)

system.basic.WskBasicTests STANDARD_OUT

Finished test Wsk Action CLI should reject delete of action that does not exist at 2016-07-14 17:20:15.233
```

IBM

Example: openwhisk.lens

```
match {
  "(?<failed>.*) > .* FAILED"
}
group {
  "^[^ ]"
}
lens stackTrace() {
  match {
    "at .*\((?<whisk>Wsk.*)\)"
  }
  stream {
    print("    at", entry.whisk)
  }
}
stream when (entry.failed) {
  print("FAILED", entry.failed);
  stackTrace(entry.group)
}
```

IBM

Example: startstop.lens

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

IBM

Implementation

- Common execution engine
  - Java 8 + java.util.regex.Matcher + javax.script.ScriptEngine (Nashorn)
  - very little JavaScript
- Command-line processor
  - table processing
  - stream processing
- Web-based notebook
  - IDE in a web browser
    - based on Apache Zeppelin
  - export code for execution with command-line processor

Demo

Formal Semantics

- To run a CloudLens script we fuse consecutive steps (group, match, stream) into stages then run the stages in sequence
  - program execution: $E \vdash p \Longrightarrow E'$
  - stage elaboration: $E, p \vdash p' \Downarrow E'$
  - stage execution: $E \vdash p \longrightarrow E'$ (see article)

$$\frac{E, [] \vdash p \Downarrow E'}{E \vdash p \Longrightarrow E'} \qquad \frac{E, p :: \texttt{match}\ \{\ patterns\ \} \vdash p' \Downarrow E'}{E, p \vdash \texttt{match}\ \{\ patterns\ \} :: p' \Downarrow E'}$$

$$\frac{E \vdash p \longrightarrow E'}{E, p \vdash [] \Downarrow E'} \qquad \frac{E \vdash p \longrightarrow E' \qquad E' \vdash p' \Longrightarrow E''}{E, p \vdash \texttt{restart}\ p' \Downarrow E''}$$

IBM

Related Work

- Programming languages and tools
  - grep + bash + vi + bc
  - AWK, Perl
  - JavaScript
  - PADS [Fisher et al.]
- Databases
  - SQL, NoSQL
  - CQL, Splunk
- Machine learning
  - data format inference
  - typestate analysis

Try it!

CloudLens

Web: https://cloudlens.github.io/cloudlens/index.html

Article: CloudLens, un language de script pour l'analyse de données semi-structurées [JFLA'17]