



Global Alliance
for Genomics & Health

Beacon Development

Miro Cupak

17/10/2016

What is this about?



Beacon API specification:

<https://github.com/ga4gh/beacon-team>

What else is out there?

Beacon Development Kits (BDK)

- Open-source quickstarts.
- Compliant with the latest beacon specification.
- Proven to work well.
- Available in:
 - Java
 - <https://github.com/mcupak/beacon-java>
 - Python
 - <https://github.com/mcupak/beacon-python>
 - JavaScript
 - <https://github.com/mcupak/beacon-javascript>

```
117  ▾ # TODO: override with the details of your beacon
118  @app.route('/beacon-python/', methods=['GET'])
119  ▾ def info():
120  ▾     return jsonify(Beacon)
```

Java BDK

- Most advanced of the BDKs.
- Effectively a reference implementation.
- Provides:
 - Compatible beacon API.
 - Sample beacon implementation.
 - Integration of Beacon data models.
 - Various utilities: output serialization, input normalization, error handling and status code mapping...
 - Test suite.
 - Security (prototype).

BDK adapter system

- Adapter API:
 - <https://github.com/mcupak/beacon-adapter-api>
- Adapter implementations for common data stores/APIs:
 - <https://github.com/mcupak/beacon-adapters>
 - **3** adapters OOTB:
 - Variants API
 - variant set → beacon dataset
 - Annotation API
 - annotation set → beacon dataset
 - VCF
 - file → beacon dataset
 - Easy to implement your own and plug into Java BDK.

Beaconizer

- Extension of Java BDk for more complex *beaconizing*.
- Support for multiple adapters and multiple beacons.
- Open-source: <https://github.com/mcupak/beaconizer>

```
{  
  "name": "beacon_id",  
  "adapterClass": "com.dnastack.beacon.core.adapter.impl.SampleBeaconAdapterImpl",  
  "configValues": [  
    {  
      "name": "sampleName",  
      "value": "33-03-03"  
    }  
  ]  
}
```

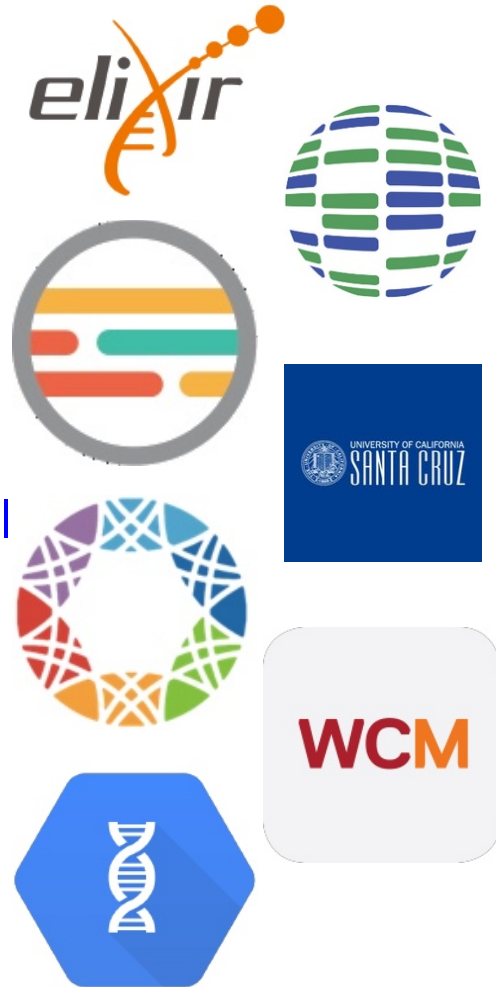


```
GET /beacons  
GET /beacons/:id  
GET /beacons/:id/query
```

Beacon implementations



- Java
 - <https://github.com/elixirhub/human-data-beacon>
 - <https://github.com/prota/cnag-beacon>
- Python
 - <https://github.com/Genecloud/simplebeacon>
 - <https://github.com/maximilianh/ucscBeacon>
 - <https://github.com/kozbo/Beacon-on-GA4GH-API>
- JavaScript
 - <https://github.com/ElementoLab/beacon>
- Go
 - <https://github.com/googlegenomics/beacon-go>



Client library

- Query beacons from Java.
- Integration with Beacon 0.3 schema.
- Open-source: <https://github.com/mcupak/beacon-client>

```
BeaconNetworkClient bnc = new BeaconNetworkClientImpl();  
bnc.getResponse(CHR1, 1L, A, HG19, "brca-exchange");
```


CLI

- Query beacons from your shell.
- Open-source: <https://github.com/mcupak/beacon-cli>

```
$ java -jar beacon-cli.jar -u http://localhost:5000/test/ response -help
```

Description: Gets response to a beacon query for allele information.

Usage:

```
-a (--assembly-id) VAL           : Assembly ID (GRC notation, e.g. 'GRCh37').  
-ab (--alternate-bases) VAL      : The bases that appear instead of the  
                                  reference bases. Accepted values: see the  
                                  ALT field in VCF 4.2 specification  
                                  (https://samtools.github.io/hts-specs/VCFv4.  
                                  2.pdf).
```

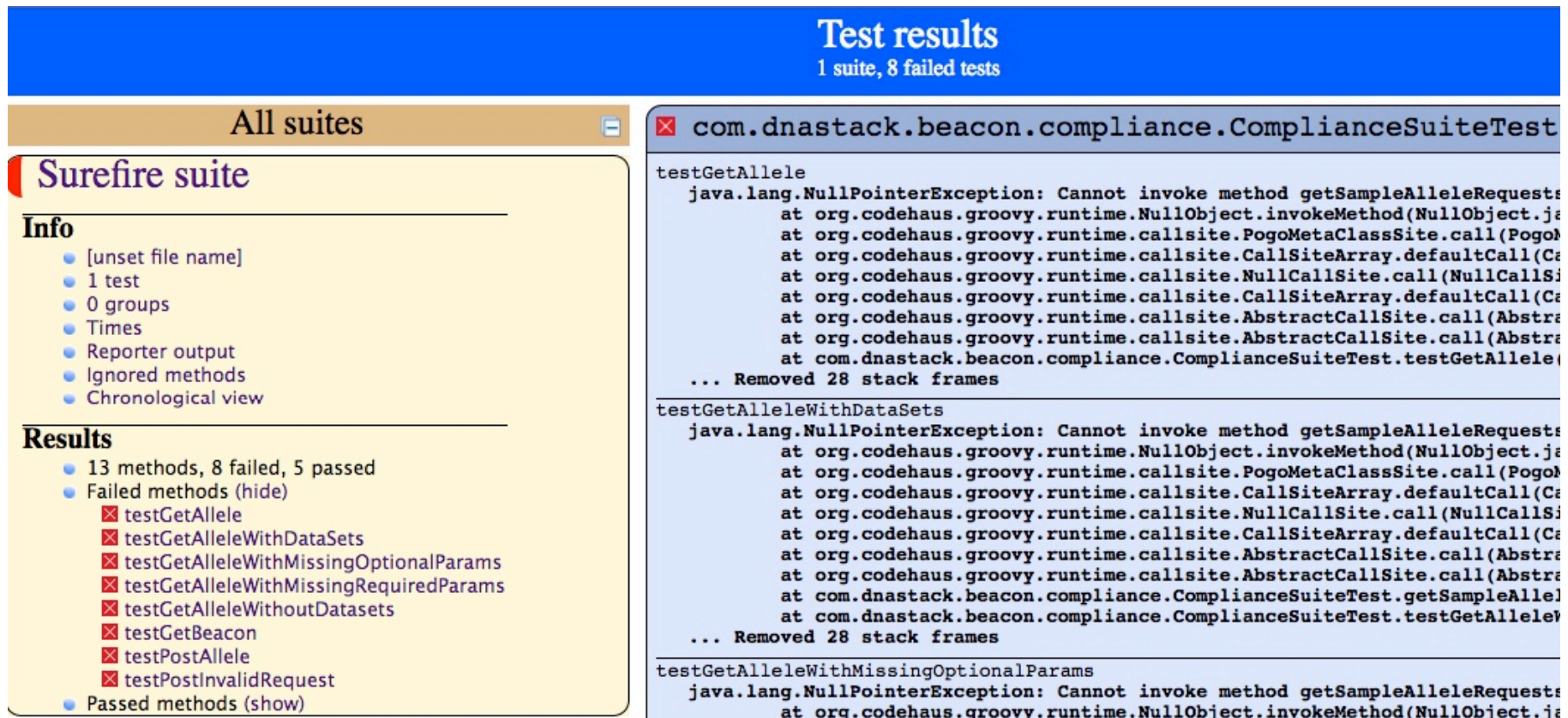
...

```
$ java -jar beacon-cli.jar -u http://localhost:5000/test/ response \  
-a GRCh37 -ab A -r 1 -rb T -s 1000
```

```
{  
  "beaconId": "test",  
  "exists": true,  
  "alleleRequest": ...  
}
```

Compliance suite

- Test beacon compatibility with 0.3 specification.
- Groovy-based prototype.
- Open source: <https://github.com/mcupak/beacon-compliance>



The screenshot displays a test results window with a blue header. The header text reads "Test results" and "1 suite, 8 failed tests". Below the header, there is a tab labeled "All suites" and a sub-tab for the "Surefire suite". The "Info" section for the Surefire suite lists: [unset file name], 1 test, 0 groups, Times, Reporter output, Ignored methods, and Chronological view. The "Results" section shows 13 methods, 8 failed, and 5 passed. The failed methods are: testGetAllele, testGetAlleleWithDataSets, testGetAlleleWithMissingOptionalParams, testGetAlleleWithMissingRequiredParams, testGetAlleleWithoutDatasets, testGetBeacon, testPostAllele, and testPostInvalidRequest. The "Passed methods (show)" section is also visible. The main content area shows the stack trace for the failed test "testGetAllele" in the class "com.dnastack.beacon.compliance.ComplianceSuiteTest". The stack trace indicates a "java.lang.NullPointerException: Cannot invoke method getSampleAlleleRequest" and lists several stack frames from the codehaus.groovy.runtime package, ending with "... Removed 28 stack frames". Similar stack traces are shown for "testGetAlleleWithDataSets" and "testGetAlleleWithMissingOptionalParams".

Test results
1 suite, 8 failed tests

All suites

Surefire suite

Info

- [unset file name]
- 1 test
- 0 groups
- Times
- Reporter output
- Ignored methods
- Chronological view

Results

- 13 methods, 8 failed, 5 passed
- Failed methods (hide)
 - ✘ testGetAllele
 - ✘ testGetAlleleWithDataSets
 - ✘ testGetAlleleWithMissingOptionalParams
 - ✘ testGetAlleleWithMissingRequiredParams
 - ✘ testGetAlleleWithoutDatasets
 - ✘ testGetBeacon
 - ✘ testPostAllele
 - ✘ testPostInvalidRequest
- Passed methods (show)

✘ com.dnastack.beacon.compliance.ComplianceSuiteTest

```
testGetAllele
  java.lang.NullPointerException: Cannot invoke method getSampleAlleleRequest
    at org.codehaus.groovy.runtime.NullObject.invokeMethod(NullObject.java:41)
    at org.codehaus.groovy.runtime.callsite.PogoMetaClassSite.call(PogoMetaClassSite.java:43)
    at org.codehaus.groovy.runtime.callsite.CallSiteArray.defaultCall(CallSiteArray.java:42)
    at org.codehaus.groovy.runtime.callsite.NullCallSite.call(NullCallSite.java:40)
    at org.codehaus.groovy.runtime.callsite.CallSiteArray.defaultCall(CallSiteArray.java:42)
    at org.codehaus.groovy.runtime.callsite.AbstractCallSite.call(AbstractCallSite.java:124)
    at org.codehaus.groovy.runtime.callsite.AbstractCallSite.call(AbstractCallSite.java:127)
    at com.dnastack.beacon.compliance.ComplianceSuiteTest.testGetAllele(...)
    ... Removed 28 stack frames

testGetAlleleWithDataSets
  java.lang.NullPointerException: Cannot invoke method getSampleAlleleRequest
    at org.codehaus.groovy.runtime.NullObject.invokeMethod(NullObject.java:41)
    at org.codehaus.groovy.runtime.callsite.PogoMetaClassSite.call(PogoMetaClassSite.java:43)
    at org.codehaus.groovy.runtime.callsite.CallSiteArray.defaultCall(CallSiteArray.java:42)
    at org.codehaus.groovy.runtime.callsite.NullCallSite.call(NullCallSite.java:40)
    at org.codehaus.groovy.runtime.callsite.CallSiteArray.defaultCall(CallSiteArray.java:42)
    at org.codehaus.groovy.runtime.callsite.AbstractCallSite.call(AbstractCallSite.java:124)
    at org.codehaus.groovy.runtime.callsite.AbstractCallSite.call(AbstractCallSite.java:127)
    at com.dnastack.beacon.compliance.ComplianceSuiteTest.getSampleAllele(...)
    at com.dnastack.beacon.compliance.ComplianceSuiteTest.testGetAllele(...)
    ... Removed 28 stack frames

testGetAlleleWithMissingOptionalParams
  java.lang.NullPointerException: Cannot invoke method getSampleAlleleRequest
    at org.codehaus.groovy.runtime.NullObject.invokeMethod(NullObject.java:41)
```

Other utilities

- Coordinate conversion library.
 - Wraps Liftover and HGVS.
 - Extensible, with built-in converters for standard assemblies.
 - Open-source: <https://github.com/mcupak/genome-coordinates-converter>

```
LiftOver intervalLiftOver = UCSCliftOver.hg19ToHg38();  
Interval newInterval = intervalLiftOver.liftOver("chr1", 743267, 743268);
```

```
String hgvs = "NM_182763.2:c.688+403C>T"  
GenomeInterval interval = HGVSConverter.hgvsToGenomic(hgvs);
```

Questions?

Any tools we're missing?

 miro@dnastack.com

 <https://git.io/vPifs>

Huge thanks to the Beacon team and our summer students Patrick and Artem.