Building and Managing Cloud PaaS with JavaScript's Nashorn JVM

CON1859

Bruno Borges - @brunoborges
Principal Product Manager
Oracle Cloud PaaS, Oracle Corp.

JavaOne ORACLE

Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Speaker

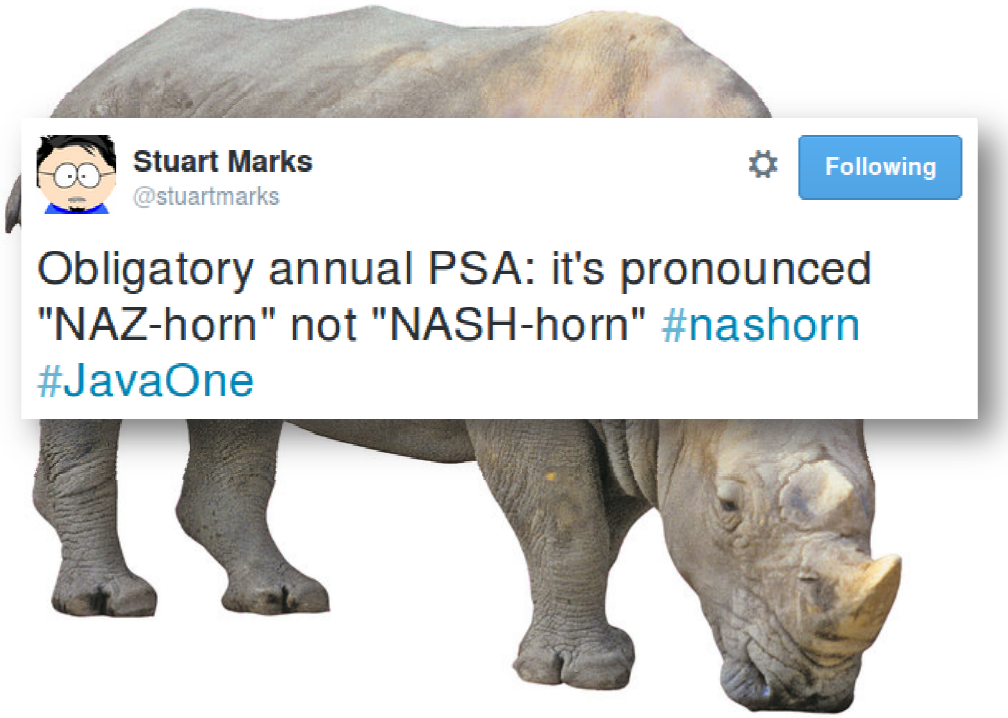
Bruno Borges

Principal Product Manager
Oracle Cloud Platform

@brunoborges

Rhino = Nashorn (German)

Rhino = Nashorn (German)



Nashorn Overview

Scripting for Java

- Java based on the JavaScript language
- ECMAScript v6.0 in latest Java SE 8 releases
- Content developer familiarity
- Access to a large set of tools and libraries
- Takes full advantage all the current Java technologies

Nashorn Overview

Features

- Tight integration with Java
- Extensions have been added provide further Java access
- Nashorn can be used from Java using the javax.script API
- Nashorn can also be used from a new command line tool jjs
- Support for shell scripting
- Full JavaFX support
- Netbeans 8 supports Nashorn development

Nashorn: JavaScript and Java, combined!

Some examples

Nashorn and JavaScript as Unix Shell Scripting Tool

Shell scripts written in JavaScript

 | @brunoborges

Shell Scripting with Nashorn

A Simpler Substitute For Other Shell Languages

- Why not use a language you are familiar with?
- Access to vast number of Java libraries
- Scripting extensions to simplify
  - Processing documents
  - Template strings
  - Command execution
  - Environment variables

Schell Scripting using Nashorn

Example 7

```
#!/usr/bin/jjs -doe -scripting
#
var dir = __DIR__ + "photos";
var files = `ls ${dir}`.trim().split("\n");

var count = 1;
for each (var file in files) {
    if (file.edsWith(".jpg")) {
        `mv ${dir}/${file} ${dir}/Photo${count++}.jpg`;
    }
}
```

Building and Monitoring Applications

JavaOne ORACLE

Playing with Cloud Services Java SDKs Using Javascript Code

 | @brunoborges

Java SDKs for Cloud Services

- Cloud services may provide Java SDKs
  - Developers can write custom Java programs to interact with Cloud services
- Nashorn makes easier to write these programs, consuming the Java SDK directly from the Javascript code
- Example: importing Oracle Storage Cloud Service Java SDK classes

```
// Imports
var CloudStorageConfig = Java.type(“oracle.cloud.storage.CloudStorageConfig”);
var CloudStorageFactory = Java.type(“oracle.cloud.storage.CloudStorageFactory”);
```

Examples of Shell Scripts Written in Java

RESTful Web APIs

REST APIs

- API to send and receive information over HTTP(S) to remote endpoints
- Architectural style; takes any programming language at any side (client/server)
- JSON and XML as data exchange format
- Developers can consume even using simple Linux commands
  - cURL, wget
- Cloud Services may provide REST APIs instead of Java SDKs (or both)

Example with cURL

Listing instances in Oracle Java Cloud Service

```
$ curl -i -X GET
  -u joe@example.com:Welcome1!
  -H "X-ID-TENANT-NAME:domain"
 https://jaas.oraclecloud.com/paas/service/jcs/api/v1.1/instances/domain
```

REST API for Oracle Java Cloud Service

Table of Contents

- API Collection Home
- REST Endpoints

Get Started

- Quick Start
- Send Requests
- Authentication
- Use cURL
- Status Codes
- Use Cases
- Context Path Deprecation Notice

REST APIs

- Service Instances
- Managed Servers
- Coherence Data Tiers
- Backup Configuration
- Backups
- Restorations
- Patches
- Cloud Tools

About the REST APIs

The Oracle Java Cloud Service REST API enables you to create and manage Oracle Java Cloud Service instances. This API provides an alternative to using the web-based user interface.

APIs

- **Service Instances**
- **Managed Servers**
- **Coherence Data Tiers**
- **Backup Configuration**
- **Backups**
- **Restorations**
- **Patches**
- **Cloud Tools**

You can view a list of all **REST Endpoints**.

@brunoborges

Nasven.js

Makes easy to setup Nashorn classpath with Apache Maven dependencies

Nasven.js

- A tool to get Maven artifacts in scope for Nashorn script execution
  - Bootstrap scripts by automatically setting up classpath based on dependencies
- Users define **package.json** and **index.js**
- **package.json** is similar to NPM's definition
- Maven dependencies configured on Gradle style
  - groupId:artifactId:version
- Code, Samples, and Docs
  - **github.com/nasven**
- Docker container available on Docker Hub

Examples of Nasven.js for Oracle Java Cloud Service

Using Java EE 7's JAX-RS Client API

JavaOne™

ORACLE®