Jenkins XML API and Mobile Devices

Simone Ardissone
Luca Milanesio
LMIT Software Ltd.

http://www. jenkins-ci.mobi

Who we are (1st guy)

- My name is Luca
- Founder and Director of LMIT Ltd (UK)
  "… the ones behind JenkinsMobi ☺"
- Jenkins (/Hudson) contributor since 2007
- Git and Gerrit big fan since 2009 ☺
- Over 19 years of experience in Software Development
- Worked for major UK, EU and US Customers
  – Banking, Retail, Industry, Finance, Telecoms, Utilities
- Development and Technical sales manager for major European and US Companies

Who we are (2nd geek)
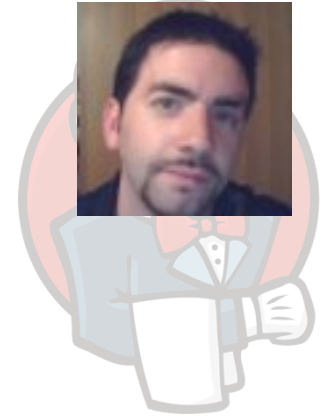
- My name is Simone
- The second arm of LMIT Ltd (UK)
- I'm on the IT as developer since 10 years
- Enthusiast of Jenkins (/Hudson) since 2007
- Involved in mobile development with iOS and Android since 3 years
- The main developer of JenkinsMobi and HudsonMobi for iOS and Android
- Author of the new Assembla plug-in for Jenkins
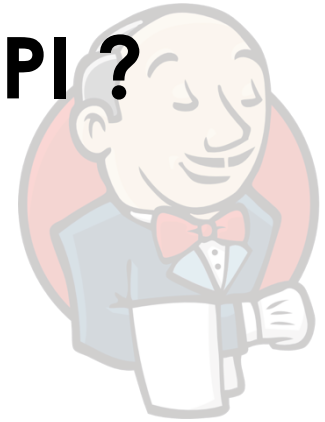
What we are going to talk about

- Jenkins Remote API introduction
- Mobile challenges
- Jenkins API for mobile applications
- Ideas to improve Jenkins Remote API
- Q&A

How many of use uses Jenkins API ?

- Quite often ?
- Some times happens ?
- Rarely ?
- Only once in my life ?
- Never ?
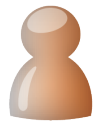- Does Jenkins have XML API ?

Jenkins Remote API

- A simple interface to query Jenkins CI
- XML based (but also JSON and Python)
- Easy to call: it's simple REST
- Easy to debug: even with a simple browser
- xPath expressions allows fast server-side computation (node count, conditions ...)
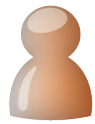- Jenkins custom 'tree' parameter to simplify queries

What can be done with this API ?

- Access Jenkins from everywhere
- Ship enhanced features based on Jenkins
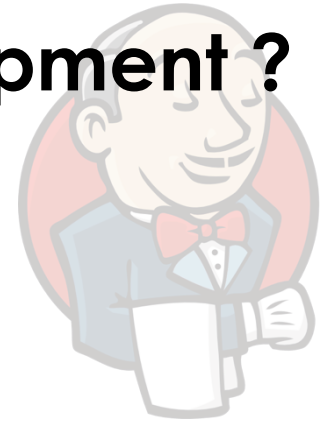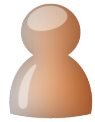- "Mobilise" the Jenkins CI UE
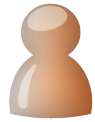
May be in the future ☺

… or something like this
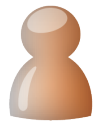
Which challenges in mobile development ?

- Expensive data connection (when in 3G)
- Limited device memory (few MB)
- Limited screen size (few inches)
- Limited device-human interaction (fingertips)

What you should do then ?

- Minimize data footprint
- Select most relevant KPIs
- Prevent long list of items to be returned over the network
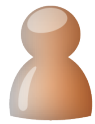- Provide rapid access with a few '*taps*'

How did we develop JenkinsMobi ?

- Fully native iOS/Android Jenkins CI client
- NO iPhoneView: remove the HTML tag download and rendering
- Full XML API and xPATH usage (with 'tree'): reduce to a minimum data transfer
- When API are missing … a bit of "nice old hacking" with AJAX calls and HTML scraping
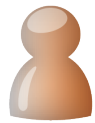
Jenkins XML API – step by step

- Step 1 – User auth
- Step 2 – GET the desired resource URL with the /api/xml suffix and optional tree or xpath query arguments
- Step 3 – Parse xml result and map it in simple objects (something like POJOs)
- Step 4 – Render resulting objects somewhere

Jenkins XML API Security

- Simple HTTP- basic authentication
- Server authentication
  - Sometimes Jenkins is hidden behind a server authentication: pass it before passing Jenkins credentials. For sys-admin: use only one kind of authentication or use the same credentials
- Problems with Jenkins Domain config
  - URLs contained into xml results are often resolved only on local network
  - Solution: override with the configured Jenkins domain URL
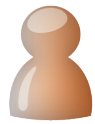
Step 1 – Authenticate and compose the URL

- Authenticate with your credentials
  - Basic auth: add "Authorization" HTTP header into a first really simple request (minimise data transfer)

    with value "Basic b64_credentials", where b64_credentials is the base-64 of 'username:password'

- Get the list of all views

  - Point to <<*domain*>>*/api/xml*

  - If you use xPath append this:
    - "*?xpath=//*/primaryView|//*/view&wrapper=views*"

  - If you use tree append this:
    - "*?tree=views[name,url],primaryView[name,url]*"
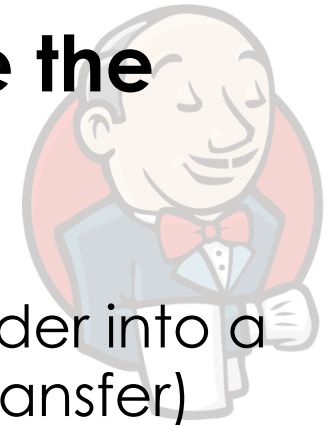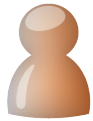
Step 2 – Parse xml result and render it
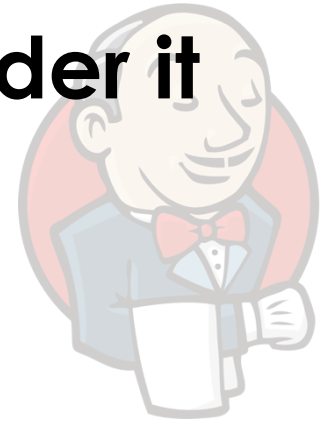
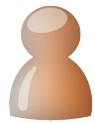You will get something like this:

```
<views>
  <primaryView>
    <name>All Failed</name>
    <url>http://ci.jenkins-ci.org/</url>
  </primaryView>
  <view>
    <name>All</name>
    <url>http://ci.jenkins-ci.org/view/All/</url>
  </view>
  <view>
    <name>All Disabled</name>
    <url>http://ci.jenkins-ci.org/view/All%20Disabled/</url>
  </view>
```
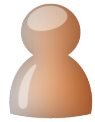
- Parse xml and create a simple bean object for each xml view or primaryView tag
- Render it in a table/list

iOS hints

- Prefer *NSURLConnection* for complete controls over data connection and authentication
- Pay attention to a correct URL escaping
- <u>Always use asynchronous data connections</u>
- Use NSXMLParser with your own NSXMLParserDelegate implementation to parse XML responses
- Perform complete tests on real devices
- SSL connection could be a problem on iOS:
  - Allow every ssl certificate (quite unsecure)
  - Tell your users to visit with Safari the site where Jenkins is hosted, this to retrieve and trust its ssl certificate

Android hints
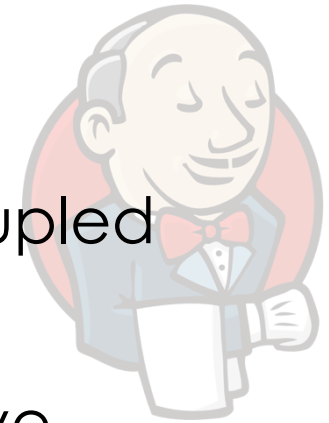
- Use a wrapped DefaultHttpClient for coupled with Apache HttpGet and HttpPost
  - gain more control over data download
- Server side authentication use preemptive authentication:
  - customize the DefaultHttpClient::CredentialsProvider with your own domain and credential
- "Trust ALL" ssl certificates policy
  - create your own SSLSocketFactory and register in the DefaultHttpClient::ConnectionManager
- For xml parsing use SAXParser with xml validation set to false
  - Most of returned XML have no XSD schema
- Use separate threads for all data connections: use AsyncTask or your own thread classes
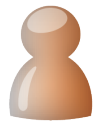
Tablets … "the next thing"

- Tablet are spreading over
- More screen space ➔ more power to Jenkins CI client
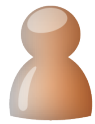- More productivity
- … but at the end it is just "cool" ☺

Jenkins on tablets … "the next thing"

… so why not JenkinsMobi for iPad ?

- Custom filters and bookmarks for often-seen jobs
- More Jenkins CI Web integration
- A redesigned UE to meet iPad users needs
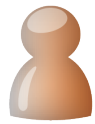- Lot of other big things

Dog-fooding: our golden rule

JenkinsMobi builds are made with Jenkins ☺

- Android App with Jenkins … easy !
    1. just install the Android SDK
    2. customize the ANT files
    3. … and you're done in a few minutes ☺

Dog-fooding: our golden rule

JenkinsMobi builds are made with Jenkins ☺

- iOS is a bit more complicated (as usual)

    1. get a Mac OSX machine

    2. install SICCI for xCode plugin for Jenkins … matches perfecly our needs.

    3. Be careful to install the proper Apple certificates

    4. Remember to install provisioning

Jenkins CI for our work

Jenkins

search ⊙ admin | log out

Jenkins » LMIT

ENABLE AUTO REFRESH

New Job

People

Build History

Edit View

Delete View

Project Relationship

Check File Fingerprint

Manage Jenkins

My Views

LMIT Software projects

edit description

	All	Git	Hudson	Jenkins	**LMIT**	+	

S	W	Name ↓	Last Success	Last Failure	Last Duration	
🔵	☀️	HudsonMobi for Android	16 days (#16)	N/A	32 sec	▶️
🔵	☀️	HudsonMobi for iOS	21 hr (#11)	N/A	1 min 23 sec	▶️
🔵	☀️	JenkinsMobi for Android	16 days (#34)	N/A	34 sec	▶️
🔵	☀️	JenkinsMobi for iOS	21 hr (#46)	N/A	2 min 45 sec	▶️

Icon: S M L

Legend RSS for all RSS for failures RSS for just latest builds

Build Queue

No builds in the queue.

Build Executor Status

#	Master
1	Idle
2	Idle
	MacMini
1	Idle

Page generated: Sep 19, 2011 2:23:30 PM   Jenkins ver. 1.420
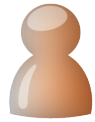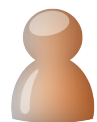
What would you change in the API ?

- Output console APIs

- Build queue control APIs

- Slave nodes control APIs

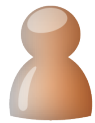- A smarter 'tree' parameter with some operations as xPath expressions have
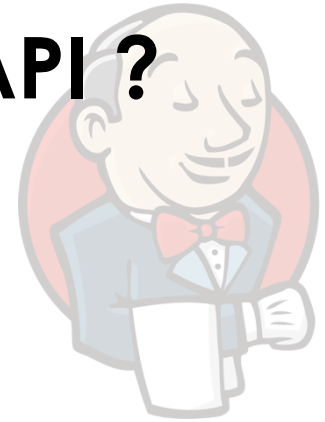
Thank you !

Q&A

Thank You To Our Sponsors

Platinum Sponsor	**CloudBees**
Gold Sponsor	**LIFERAY.**
Silver Sponsor	**redhat.**
Bronze Sponsors	**New Relic.** **CHARIOT SOLUTIONS** Practical, Smart Software Development **sauce LABS** **eXo**

Coming Soon: **The CloudBees Newsletter for Jenkins**

✓ *Please complete the Jenkins survey to help us better serve the community (bonus: a chance to win an Apple TV!)*