AGL-JTA
CIAT tool for AGL

October 27, 2016
CEWG Japan Mini Jamboree 2016
Kyohei Oki
Yuichi Kusakabe
Fujitsu TEN
Self Introduction

➢ Kyohei Oki (kyohei.oki@jp.fujitsu.com)

➢ Fujitsu Ten Software Engineer (2012 ~ )
Agenda

- **WHAT** is CIAT
- **WHY** use AGL-JTA
- **HOW** to use AGL-JTA
- **DEMO**
- **Future Work**
WHAT is CIAT

Continuous Integration and Automated Test

- [https://wiki.automotivelinux.org/eg-ciat#explanation_of_ciat](https://wiki.automotivelinux.org/eg-ciat#explanation_of_ciat)

CIAT is supposed to include:

- CI pipeline which executes tests on user's demand or triggered by upstream changes automatically
- collection of source code from upstream
- automated instructions for building/deploying built distro
- ability to include binary artifacts
- automated test pipeline which executes sets of tests
- publishing of built distro/component and test results/logs
- mechanism for formal code review prior to merging of changes
- demonstration of license compliance
WHY use AGL-JTA

➤ based on Fuego
  ● formerly called JTA (Jenkins Test Automation)
  ● A host/target script engine and a jenkins front-end
  ● official automated test framework for LTSI project

➤ advantages
  ● lots of plugins to extend features
  ● highly customizable
  ● flexible test configuration
  ● running tests in batches
  ● not imposing any demands on boards or distributions
  ● easy yet flexible board setup
### WHY use AGL-JTA

- match AGL CIAT’s goals

<table>
<thead>
<tr>
<th>CIAT’s goals</th>
<th>AGL-JTA’s feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>flexible trigger</td>
<td>build triggers (like gerrit, cron job)</td>
</tr>
<tr>
<td>collection of source code</td>
<td>Source Code Management (SCM, like git)</td>
</tr>
<tr>
<td>instructions for building/deploying</td>
<td>host/target script engine</td>
</tr>
<tr>
<td>include binary artifacts</td>
<td>host/target script engine</td>
</tr>
<tr>
<td>executing sets of tests</td>
<td>job trigger of jenkins</td>
</tr>
<tr>
<td>publishing of distro and test results</td>
<td>plugins of jenkins, or use SCM to upload distro/result</td>
</tr>
<tr>
<td>code review</td>
<td>(offered by gerrit)</td>
</tr>
<tr>
<td>demonstration of license compliance</td>
<td>(offered by gerrit)</td>
</tr>
</tbody>
</table>
HOW to use AGL-JTA

simple test

- jenkins will call script engine first, then gather test result/log from script engine
- script engine will do the work
  - cross-compile testsuite for target board
  - load testsuite to target board then execute
  - gather test logs
HOW to use AGL-JTA

➢ have a glance – homepage
HOW to use AGL-JTA

- have a glance – execute tests
HOW to use AGL-JTA

- have a glance – execute tests
HOW to use AGL-JTA – CIAT

role in AGL’s CIAT
- for Automated Test mainly
- also involves building/deploying images

work flow
- trigger
- collect source code
- build/deploy distro to target board
- execute tests on board
- public test result

instance
- private server – inhouse CIAT (mid-term test)
HOW to use AGL-JTA – CIAT

private server – inhouse CIAT

1. trigger

- CIAT.inhouse_mid

2. collect source code

- CIAT.inhouse_deploy

3. build & deploy distro

- CIAT.batch_test

4. execute sets of tests, like Benchmark.bc, Functional.zlib

- CIAT.upload

5. publish test result
HOW to use AGL-JTA

- CIAT environment
  - hardware setting

Diagram:
- Internet (AGL gerrit)
- Hub
- Target board
- TFTP/NFS server
- AGL-JTA
- Compile server
HOW to use AGL-JTA

- CIAT environment
  - hardware setting

1. AGL-JTA
2. TFTP/NFS server
3. compile server

hub
Internet
porter
HOW to use AGL-JTA

▸ more flexible way to share test result
  • public git repo for storing test result
  • AGL-JTA gather required test result and display
  • still in developing

FTEN's AGL-JTA ➔ public git repo ➔ my AGL-JTA

xxx company's AGL-JTA ➔

yyy company's AGL-JTA ➔
HOW to use AGL-JTA – share test result

CIAT.upload
- one part of CIAT (publishing test result)
- called by CIAT.mid, CIAT.inhouse_mid
- test summary, test info, manifest, detailed results

public git repo for test result
- git:
  https://gerrit.automotivelinux.org/gerrit/gitweb?p=staging/agl-jta-results.git;a=summary
- branch for each snapshot
- 1st-level directories are for company & target board
HOW to use AGL-JTA – display test result

➢ make shared test result easy to read and compare
HOW to use AGL-JTA – display test result

- make shared test result easy to read and compare

<table>
<thead>
<tr>
<th>No.</th>
<th>Benchmark</th>
<th>Average</th>
<th>Unit</th>
<th>Criteria</th>
<th>Result</th>
<th>Output</th>
<th>Unit</th>
<th>Rate (Output/Average)</th>
<th>Result</th>
<th>Output</th>
<th>Unit</th>
<th>Rate (Output/Average)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2048_Kb_Record_Write_Random_write</td>
<td>72648.50 KB/s</td>
<td>KB/s</td>
<td>0.70 – 0.80</td>
<td>PASS</td>
<td>51191</td>
<td>KB/s</td>
<td>0.70</td>
<td>FAIL</td>
<td>49566</td>
<td>KB/s</td>
<td>0.68</td>
</tr>
<tr>
<td>2</td>
<td>2048_Kb_Record_Read_Rewrite</td>
<td>111210.60 KB/s</td>
<td>KB/s</td>
<td>0.92 – 1.02</td>
<td>PASS</td>
<td>103986</td>
<td>KB/s</td>
<td>0.94</td>
<td>FAIL</td>
<td>100746</td>
<td>KB/s</td>
<td>0.91</td>
</tr>
<tr>
<td>3</td>
<td>2048_Kb_Record_Write_Rewrite</td>
<td>71567.80 KB/s</td>
<td>KB/s</td>
<td>0.72 – 0.82</td>
<td>PASS</td>
<td>52622</td>
<td>KB/s</td>
<td>0.74</td>
<td>PASS</td>
<td>51375</td>
<td>KB/s</td>
<td>0.72</td>
</tr>
<tr>
<td>4</td>
<td>2048_Kb_Record_Read_Rewrite</td>
<td>131265.30 KB/s</td>
<td>KB/s</td>
<td>0.77 – 0.87</td>
<td>PASS</td>
<td>102900</td>
<td>KB/s</td>
<td>0.78</td>
<td>FAIL</td>
<td>97561</td>
<td>KB/s</td>
<td>0.74</td>
</tr>
<tr>
<td>5</td>
<td>2048_Kb_Record_Write</td>
<td>65798.60 KB/s</td>
<td>KB/s</td>
<td>0.77 – 0.87</td>
<td>PASS</td>
<td>51822</td>
<td>KB/s</td>
<td>0.79</td>
<td>FAIL</td>
<td>48864</td>
<td>KB/s</td>
<td>0.74</td>
</tr>
<tr>
<td>6</td>
<td>2048_Kb_Record_Read</td>
<td>128735.90 KB/s</td>
<td>KB/s</td>
<td>0.72 – 0.82</td>
<td>PASS</td>
<td>95598</td>
<td>KB/s</td>
<td>0.74</td>
<td>PASS</td>
<td>94222</td>
<td>KB/s</td>
<td>0.73</td>
</tr>
<tr>
<td>7</td>
<td>2048_Kb_Record_Write</td>
<td>26588.40 KB/s</td>
<td>KB/s</td>
<td>0.83 – 0.93</td>
<td>PASS</td>
<td>22254</td>
<td>KB/s</td>
<td>0.84</td>
<td>FAIL</td>
<td>21901</td>
<td>KB/s</td>
<td>0.82</td>
</tr>
<tr>
<td>8</td>
<td>2048_Kb_Record_Read_Random_read</td>
<td>102667.10 KB/s</td>
<td>KB/s</td>
<td>0.85 – 0.95</td>
<td>PASS</td>
<td>91461</td>
<td>KB/s</td>
<td>0.89</td>
<td>PASS</td>
<td>88092</td>
<td>KB/s</td>
<td>0.86</td>
</tr>
<tr>
<td>9</td>
<td>2048_Kb_Record_Write_Rewrite</td>
<td>71998.50 KB/s</td>
<td>KB/s</td>
<td>0.71 – 0.81</td>
<td>PASS</td>
<td>51016</td>
<td>KB/s</td>
<td>0.71</td>
<td>FAIL</td>
<td>49079</td>
<td>KB/s</td>
<td>0.68</td>
</tr>
<tr>
<td>10</td>
<td>2048_Kb_Record_Read</td>
<td>107482.80 KB/s</td>
<td>KB/s</td>
<td>0.85 – 0.95</td>
<td>PASS</td>
<td>99690</td>
<td>KB/s</td>
<td>0.93</td>
<td>PASS</td>
<td>92184</td>
<td>KB/s</td>
<td>0.86</td>
</tr>
</tbody>
</table>
HOW to use AGL-JTA

- simple test
  - REAME
  - docs/jta-docs.pdf
  - docs/How-to-Add-Test-Cases-on-JTA

- CIAT
  - docs/How-to-Configure-CIAT-on-AGL-JTA.pdf

- More documents for your instances
Demo contents

- Inhouse CIAT
- Carry out a test in a manual trigger and operate slowly
- Test is IOZONE performance -> Filesystem tests
- Compare results and introduce display detail
Future Work

- board supporting
- trigger
- display
- cooperate with fuego
- ...

Open Fujitsu-ten Technical Evolution Conference
Future Work

- **board supporting**
  - current
    - CIAT for porter
    - simple test for porter & MinnowBoard
  - future work
    - Dragon board
    - more board?

- **trigger**
  - current
    - gerrit trigger
  - future work
    - for different types of changes uploaded to gerrit
    - for snapshot/release
Future Work

➢ display
   • current
   • only display result of each tests
   • future work
   • display detailed information/log of each tests
   • test environment information

➢ cooperate with fuego
   • fix gap between fuego
   • share features and test cases
Future Work

- AGL support
  - [https://jta.automotivelinux.org/](https://jta.automotivelinux.org/)

- your suggestion to improve it

- sharing test cases

- sharing test results

- participate in improving AGL-JTA and AGL’s CIAT
Thank you!

kyohei.oki@jp.fujitsu.com
yuichi.kusakabe@jp.fujitsu.com