OpenCL testing framework for Piglit

XDC 2012, Nuremberg
OpenCL testing framework for Piglit

About

- **Background:**
  - EvoC project
  - Piglit framework
  - TDD and regression finding

- **Goals:**
  - Test OpenCL implementation compliance
  - Test all OpenCL versions
  - Easy writing of tests
OpenCL testing framework for Piglit

Design

- Piglit gives us:
  - Concurrent testing
  - Grouping of tests
  - Results display
- Test all platforms and devices if possible
- Tests can be platform and/or device specific
- Share as much common code between tests
- Helper functions (call multiple API functions or a subset of all API function arguments)
Tests

- **Test:**
  - Configuration section
  - Test section

- **Different test types:**
  - API
  - Program
  - Custom

- **Why different types?**
  - Share common code
  - Write only the part you're testing
Program tester

- Test OpenCL C programs without writing any C/C++ code
- Specify input, compare output
- Execute multiple tests on same program
### Results:

<table>
<thead>
<tr>
<th>Platform</th>
<th>Device</th>
<th>API</th>
<th>Program</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clover</td>
<td>Evergreen</td>
<td>18/30</td>
<td>13/39</td>
<td>32/70</td>
</tr>
<tr>
<td>Intel OpenCL</td>
<td>Core i5 Arrandale</td>
<td>16/30</td>
<td>36/39</td>
<td>53/70</td>
</tr>
<tr>
<td>AMD APP</td>
<td>Core i5 Arrandale</td>
<td>15/30</td>
<td>39/39</td>
<td>55/70</td>
</tr>
</tbody>
</table>

Most of the API tests fail because of invalid returned error codes and crashes on “invalid” usage.

### Users:
- Clover
- Radeon
Future plans

- **Short term:**
  - Write a lot more tests
  - Create clinfo binary (something similar to glxinfo)
  - Add support for half type

- **Long term:**
  - OpenGL + OpenCL tests
  - Support SPIR (Standard Portable Intermediate Representation)
Questions?