Agenda

- Site changes since previous TC39 Meeting
- Test coverage
- “Test262 Beta”
- Invalid (valid) tests
- Community contributions?
Have a working bug tracker and community involvement. Thanks Mozilla!

Roughly 3,000 test case additions since January

Test cases are now in JSON format

(Most) invalid test cases disabled

Many small bug fixes
Test Coverage
Big Picture (January)

- **ES3** - Google submissions
- **ES5 (additions/changes from ES3)** - Microsoft submissions
- **ES5 (additions/changes from ES3)** - anticipated soon from Microsoft
- **ES5 (additions/changes from ES3)** - estimated missing coverage

- 2399
- 2250
- 5391
- 898
Test Coverage
Big Picture (March)

- ES3 - Google submissions
- ES5 (additions/changes from ES3) - Microsoft submissions
- ES5 (additions/changes from ES3) - estimated missing coverage
# Test Coverage

## ES5 Additions Completely Missing Tests

<table>
<thead>
<tr>
<th>Section</th>
<th>Estimated Missing Tests</th>
</tr>
</thead>
<tbody>
<tr>
<td>Annex C et al.</td>
<td>250</td>
</tr>
<tr>
<td>(Strict Mode)</td>
<td></td>
</tr>
<tr>
<td>Section</td>
<td>Existing Tests</td>
</tr>
<tr>
<td>-------------</td>
<td>----------------</td>
</tr>
<tr>
<td>Chapter 10</td>
<td>107</td>
</tr>
</tbody>
</table>
- Bug tracking system
- Google contributor’s agreement to ECMA
- Sufficient test coverage including Strict Mode...

Strict Mode test cases are out there today – someone just needs to sign a contributor agreement and officially contribute them 😊

In the meantime, does ~95% ➔ Beta?
ES5 doesn’t allow “\1” but...
...FireFox 4, Chrome 10, Opera 11, and IE9 do 😊

and for that matter so does ES5:

An implementation may extend program syntax and regular expression pattern or flag syntax. To permit this, all operations (such as calling `eval`, using a regular expression literal, or using the `Function` or `RegExp` constructor) that are allowed to throw `SyntaxError` are permitted to exhibit implementation-defined behaviour instead of throwing `SyntaxError` when they encounter an implementation-defined extension to the program syntax or regular expression pattern or flag syntax.

* Annex B also covers octals in strings as a suggested extension

➡️ Can we not have negative grammar tests?
PROPOSAL
We allow negative grammar tests, but will immediately disable or ‘fix’ them if any JavaScript implementer complains via a bug on http://bugs.ecmascript.org
A few ideas to get a discussion going:

- Openness
- Additional resources for test262
- Legal implications
- Quality of community-submitted tests
- Test cases versus test harness. Patent concerns WRT the latter?
- test262-discuss