**Agenda for the:** 22nd meeting of Ecma TC39  
**to be held in:** Santa Cruz, CA, USA  
**on:** 24 – 26 May 2011  

**TIME:**  
10:00 till 17:00 on 24th and 25th of May 2011  
10:00 till 16:00 on 26th of May 2011  

**LOCATION:**  
University of California, Santa Cruz  
Engineering 2 Building, room 215  
1156 High St, Santa Cruz  
CA 95064, USA  
Mr. Cormac Flanagan’s e-mail: cormac@ucsc.edu  
Mr. Cormac Flanagan’s phone: +1 831 459 5375  

1 Opening, welcome and roll call  
1.1 Opening of the meeting (Mr. Neumann)  
1.2 Introduction of attendees  
1.3 Host facilities, local logistics  

2 Adoption of the agenda (**2011/024-Rev2**)  

3 Approval of minutes from March 2011 (**2011/021**)  

4 Report from the Secretariat  
4.1 Report from the Secretary General  
4.2 Report of the status for a Technical Report on interoperability/conformance tests  
4.2.1 Prototype Website (http://test262.ecmascript.org and http://test.w3.org/html/tests/reporting/report.htm)  
4.3 Report from the ad hoc on Internationalization standard  
4.4 W3C Joint work items  
4.5 Draft press release (**2011/025**)  

5 Progression of ES 5  
5.1 Publication by ISO and Ecma-International  

6 Discussion of ES harmony (technical contributions are available and can be found on the ES wiki)  
6.01 http://wiki.ecmascript.org/doku.php?id=strawman:observe observe, better databinding via object change notifications
6.1 Syntax (Tuesday)

6.1.1 quasis

6.1.2 Promotion decisions regarding RegExp
"Match Web Reality" (including RegExp.prototype.compile)
Other RegExp extensions and API improvements. (who are the champions??)

6.1.3 Array comprehensions

6.1.4 Pragmas -- don't need to settle on a concrete design, just agreement
that we should future-proof a pragma syntax

6.1.5 http://wiki.ecmascript.org/doku.php?id=strawman:multiline_regexps

6.1.6 http://wiki.ecmascript.org/doku.php?id=strawman:paren_free


6.2 objects and classes (Wed. am)

6.2.1 "Subtyping" and setting the prototype of various literal object forms

6.2.2 Object Literal Extensions
review for possible promotion
http://wiki.ecmascript.org/doku.php?id=strawman:basic_object_literal_extensions

6.2.3 Private names

6.2.4 classes
http://wiki.ecmascript.org/doku.php?id=strawman:classes_with_trait_composition

6.2.5 Scoped Object Extensions –

6.2.6 “Trademarks/guards” (the language feature, not the legal concept).
http://wiki.ecmascript.org/doku.php?id=strawman:trademarks

6.2.7 Better databinding via object change notifications

6.3 API's (wed PM)

6.3.1 soft fields

6.3.2 better random

6.3.3 simple maps and sets
http://wiki.ecmascript.org/doku.php?id=strawman:basic_object_literal_extensions

6.3.4 Full Unicode Support:
http://wiki.ecmascript.org/doku.php?id=strawman:support_full_unicode_in_strings

6.3.5 Promotion to proposal status for various enhanced math functions
and operators. See the "Number and Math Enhancements", I'll will be
preparing a list of specific proposed functions based upon the spreadsheet
Also see http://wiki.ecmascript.org/doku.php?id=strawman:modulo_operator
6.3.6 String extras
http://wiki.ecmascript.org/doku.php?id=strawman:string_extras

6.4 functions (thurs am)
6.4.1 Function.prototype.toString
http://wiki.ecmascript.org/doku.php?id=strawman:function_to_string
6.4.2 Arrow Function vs Block Lambda
http://wiki.ecmascript.org/doku.php?id=strawman:arrow_function_syntax
6.4.3 http://wiki.ecmascript.org/doku.php?id=strawman:completion_reform
6.4.4 http://wiki.ecmascript.org/doku.php?id=strawman:name_property_of_functions

6.5 control flow (thurs pm)
6.5.1 concurrency
6.5.2 Deferred Functions - Deferred functions enable writing asynchronous code in a linear style where you would otherwise use callbacks and CPS transformations
6.5.3 Enumeration -- again, doesn’t need much conversation, just agreement that we want to nail down the semantics

6.6 modules (Thurs pm)
6.6.1 simple module functions (was "generative module expressibility")
6.6.2 Multiple globals -- shouldn’t need much conversation, just agreement that this reality should be reflected in the spec