Minutes of the: ***Ecma TC39-TG1***

held in: ***Phone Conference***

on: ***10th February 2006***

Attendees

- Brendan Eich, Mozilla Foundation
- Ed Smith, Adobe Systems
- Jeff Dyer, Adobe Systems
- Lars T. Hansen, Opera Software

Agenda

- Introductions, Lars T. Hansen (Opera)
- Go through clarification issues.
- Go through foundational issues.
- Review recent changes on the wiki.

Notes

- Opera and Adobe both care about small-memory target devices.
- Costs of the static dialect of concern to Opera, Adobe, everyone.
- Re: code_mixing, Ed observes that you can defer boxing until calling back into E3 String.prototype.
    - In particular, through different open namespaces, E3 String and E4 String class could be different objects.
    - Or, String could be the same, but new String would mean different things.
    - But, is String instanceof Function? Or instanceof Class? Or both?
    - String vs. MyConstructor already differ in E3: prototype readonly/dontdelete, “original value of the prototype”, etc.
    - Adobe to put proposal up on how to interoperate, or support via single implementation, with minimal pain.
- Want to take temperature of:
    - drop_traits – don’t worry about mutable proto, we are not proposing it.
    - is_as_to – need to finalize nullability, cast syntax.
    - type_parameters – Mozilla is hot on this for completeness of Edition 4, esp. for containers, also for function subtypes.
    - switch_class – Mozilla is hot on this too, should be easy to fit in.
    - Mozilla cool on linked_namespaces.
    - iterators_and_generators seems good, especially in that it does not require native stack capture or threading.
        - Need to finish iteration protocol and recast `for..in` loops in terms of it.
- Language version selection and detection
    - Via MIME type parameter.
    - Via some in-language, backward-compatible way.
    - Should work to get RFC 4329 updated to specify versions better, including E4.

- Ed suggested adding `string/number/boolean` (all-lowercase) E4 classes implementing the E3 primitive types.
    - Leaves `String/Number/Boolean` E3-compatible, in particular you can construct these and pass them around as in E3.
    - Lets E4 seal instances, avoid boxing, for `string/number/boolean` (doesn't help optimize prototype method calling).
    - Proposal forthcoming.