Ecma/TC39-TG1/2007/019

Minutes of the:	***Ecma TC39-TG1***
held in:	***Phone conference***
on:	***8th of May 2007***

Attendees

- Lars Hansen, Adobe Systems
- Jeff Dyer, Adobe Systems
- Dave Herman, Adobe Systems
- Doug Crockford, Yahoo!
- Chris Pine, Opera Software
- Graydon Hoare, Mozilla Foundation
- Cormac Flanagan, UCSC
- Francis Cheng, Adobe Systems
- Pratap Lakshman, Microsoft
- Brian Crowder, Mozilla Foundation
- Brendan Eich, Mozilla Foundation

Agenda

- [Lars] Regular expression intersection and subtraction syntax: re discussion on es4-discuss, see discussion:extend_regexps
    - Using `\&` and `\-` for operators is potentially less backward compatible than we thought, because some people habitually escape all punctuation; that behavior has long traditions
    - Java uses `[...&&[...]]` for intersection and `[...&&[^...]]` for subtraction, we could choose to be compatible but it will slightly complicate the first-level lexer.

- [Brendan] bug fixes additions based on Lars's inspired `this function` and `this generator` to pave a path away from `arguments` usage and extension.

- [Francis] Using Trac as our bug tracking system. It seems to be well-suited for our purposes. Who would host it?

- [Cormac] self type Adding a "Self" type to precisely type methods in structural object types.

Notes

- Regular expression intersection and subtraction syntax. Lars will do a little more research on what Perl and Python, etc. do and report back next week.

- `this function` and `this generator`. Jeff thinks this would be more persuasive if it were part of a proposal to replace the arguments object. Lars likes that idea. The arguments object is used widely in web code, but Lars thinks that it may not be necessary in ES4. Jeff thinks this needs more justification. No decision made.

- Bug tracking system for Trac. Francis and Dave will discuss the details offline. Dave thinks it would be useful to have a domain name reserved that the community can look to for the latest spec, ref imp and bugbase.

- Self type. RESOLVED: Consensus is that this is a good idea, it plugs a hole in the type system. Syntax will be `this` as in

```
T = { f:function(this:this,...):R, ...}
```

- Reference Implementation. Jeff was talking to a standards expert who suggested that the ref impl could be published as an ECMA technical report. We could rev it more frequently than the specification.