

**Minutes of the:  
held in:  
on:**

**Ecma TC39, ES3.1WG  
Phone conference  
29 July 2008**

## 1 Roll call and logistics

### 1.1 Participants

Brendan Eich (Mozilla), Pratap Lakshman (Microsoft), Adam Peller (IBM), Sam Ruby (IBM) and Allen Wirfs-Brock (Microsoft)

## 2 Agenda

Decimal

const - should we back off on on block scope, and allow const only at the top level of functions.

arguments - should we make it a true ES array?

Edits based on review feedback from Oslo; schedule for circulating the next draft

## 3 Minutes

**Decimal**

Information about decimal available through multiple channels; wiki, on the discuss lists, and through downloadable implementations - need to review and provide feedback - Sam has accounted for some of Waldemar's feedback comments (and is in the process of getting access to the rest of the comments); spec update to follow - approach is to evolve it in 3 stages; (1) introduce a ctor for Decimal with a string argument, and just the basic set of instance methods (this is already there in the 15 July draft spec) (2) evaluate adding static functions (3) introduce decimal literals and infix operator support - concern over whether we can make the changes on time given the aspirational schedule for ES3.1 - what about other browser vendors? will they sign up to provide implementation support? check with pratapL and Maciej - SpiderMonkey patch is big; there could be concerns regarding performance and size - Sam to go ahead and spec it anyway, and let the committee decide.

**const - should we back off on on block scope, and allow const only at the top level of functions.**

not sure if implementations will honour spec language that says that const is allowed only at top level; look what has happened to function declarations! - if we choose to allow only top level consts, can we make sure that we keep the option open to get it right in a future revision of the spec - can we add informative guidance in the spec saying how const should behave if implementers choose to support it at block level - the spec should also say how it should NOT behave; important, so that we can get it right in the next revision - Chapter 10 needs to be cleaned up first; lets wait till that is done before making the call.

**arguments - should we make it a true ES array?**

Would not be possible without extra magic due to the aliasing of 'arguments' and the formal parameters list - mutation cannot be found out through static analysis because of the presence of eval; we should not require any kind of analysis in ES3.1 to handle this aliasing - won't be backwards compatible either - this has to be the way it is today - at best we can make argument's prototype be Array.prototype.

**Edits based on review feedback from Oslo; schedule for circulating the next draft**

Send edits to pratapL by Fridays; pratapL to upload by Mondays.

Meeting adjourned.