

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

**Ecma TC39, ES3.1WG  
Phone conference  
21 August 2008**

## 1 Roll call and logistics

### 1.1 Participants

Pratap Lakshman (Microsoft), Mark Miller (Google), Adam Peller (IBM), Sam Ruby (IBM) and Allen Wirfs-Brock (Microsoft)

## 2 Agenda

Attribute defaults for Object.defineProperty

Pending work prior to Redmond meeting

## 3 Minutes

### Attribute defaults for Object.defineProperty

from a security perspective, new constants in Object.defineProperty attributes should default to 'false' - better support for high integrity programming - support for "deny by default"; put the cost on "allowing" things - notational costs influence usability.

Percentage of JavaScript programmers doing high-integrity programming is small - consistency is more important than security considerations - need to preserve semantics with respect to existing JavaScript - trying to condition meta operations as 'not for the casual programmer' - these are typically going to be wrapped with sugar, or emitted by translators - keep in mind that defineProperty is also used to refine properties - how about saying the missing attributes mean 'leave it alone' ?

So, we have two options:

(1) When creating a property using defineProperty all unspecified attributes default to false

(2) When creating a property using defineProperty all unspecified attributes default to the same values they have when a property is created by assignment or within an object literal.

Plan - ask on the discuss list for opinions - lets get closure on Tuesday's call (26 Aug).

### Optional third argument to Object.create

Indicates whether it is extensible or not - so, what should be the default ? - no happy default - if we do catchalls in "harmony" that will require one or more such attributes - better to introduce them in a stratified manner - when we do catchalls the third argument must be able to specify a larger set of object attributes - ok, lets postpone it from ES3.1.

### Renaming "Flexible"

How about "configurable" ? - seems more close to the meta level than "Flexible" - also, seems more in line with naming convention used for the other attributes. Can we have something shorter ? - how about "pinned" ? - "prohibition" naming instead of the "permission" naming - there is value in brevity though - long names are a problem when reading (given programmer are read far more often than written) - those wanting to write high integrity programs will need to mention "configurable : false" in property descriptors; could become tedious if we go with option (2) above - how about using 'w', 'e' and 'c' as abbreviations for writable, enumerable

and configurable ? - all agree to drop this idea. Plan - since this is related to attribute defaults, lets ask this too on the discuss list - ok, lets send a single email asking both the questions then.

**Pending work prior to Redmond meeting**

Lets discuss over email.

Meeting adjourned.