Minutes of the:	*Ecma TC39, ES3.1WG*
held in:	*Phone conference*
on:	*03 June 2008*

1 Roll call and logistics

1.1 Participants

Doug Crockford (Yahoo!), Pratap Lakshman (Microsoft) and Allen Wirfs-Brock (Microsoft)

2 Agenda

Array extras

Strict mode 'throw something' - should we define a new error type ? Preferably not.

The new static properties on Object: freeze, seal, etc. - review from an ES4 perspective

3 Minutes

Nothing about garbage collection in the spec - no way to produce a correct implementation that does not do GC - should we mention something about it ? - what do other specifications do ? What does the Scheme spec say, for example ? - typically specifications only mention the GC precondition; once an object is unused it goes away - maybe we don't want to update the current spec language - agreed; lets leave it as is.

Array extras

Spec it the way it is currently supported in FF - should we add any extras that showed up after JS 1.6 ? - what if they are not there in the 3 out of 4 browsers ? - specifically, refer the 'More Array extras' from JS1.8 - reduce and reduceRight - no harm adding them - they key point is to ensure they don't introduce new syntax -

if there is objection from others we can reconsider - pratapL to update Array generics proposal.

Strict mode 'throw something' - should we define a new error type ? Preferably not.

Lets not add any new error types - if the error is syntactic, throw SyntaxError, else throw one of the existing exceptions that is relevant - agreed, that would make for a simpler implementation - but, that could cause users confusion too! - user gets a syntax error in a strict module; cut-n-pastes the line into a non-strict module (without realizing it) and now the error goes away - that could confuse some users - do we say anything about the diagnostic in the spec ? - no - then we can add some recommendations that implementations are encouraged to add some discriminating text in the diagnostic for errors thrown in strict mode - shall we put this in the appendix ? - lets add this recommendation in the section that introduces Strict mode.

The new static properties on Object: freeze, seal, etc. - review from an ES4 perspective

Can ES4 nominal types by de-sugared to using these methods ? - but, we do not make that claim - the motivations for adding the static properties to Object are different - they are driven by the needs of getters/setters and to be able to lock down an object - but what if ES4 say that means providing two forms to achieve the same goals - we haven't heard any objections yet - lets leave it as is for now - we need to flesh out the details first - especially for beget, defineOwnProperty, and defineOwnProperties - would like to rename them to defineProperty

and defineProperties (dropping 'own') - the 'own' qualifier makes more sense in the querying functions - Allen to update the proposal with defineProperty and defineProperties

Enumeration order

I would like to see us not enforce enumeration order - the ES3 spec language allows implementers freedom to optimize - IE too maintains two representations for the array - but there are real world apps that depend on enumeration order - well, the IE change shipped in IE8 Beta1, and I am not aware of any sites breaking - will check further - can we have different spec language for numeric properties ? Like, numeric properties should be enumerated in sorted order, and non-numeric properties must be enumerated in insertion order ? - things get complicated because of the sparseness - lets ask on the discuss lists.

JSON
Should we add the JSON grammar into the specification or just reference it ? - we should just reference the RFC - agreed, that would be the better choice - Doug to update JSON proposal.

Lets try to get the updated proposals ready by next Tuesday's call.

Meeting adjourned.