

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

**Ecma TC39, ES3.1WG
Phone conference
12 February 2009**

1 Roll call and logistics

1.1 Participants

Pratap Lakshman (Microsoft), Mark Miller (Google) and Allen Wirfs-Brock (Microsoft)

2 Agenda

Not circulated ahead of time.

3 Minutes

On the `[[Class]]` internal property

Should restrict values that host objects can use for their `[[Class]]` property - for every value of the `[[Class]]` property we should call out the contract that needs to be supported by the object - `[[Class]]` needs to be treated as an unforgeable nominal type for the internal type system; we should call out what specific internal methods need to be supported - that would be clarifying - should not have to repeat/rehash any of the prose that is already in the specification though - agreed, especially because some of the JavaScript objects have `[[Class]]` values that do not adhere to this metaphor of nominal types; for e.g. Math objects have `[[Class]]` set to "Math", and JSON objects have their `[[Class]]` set to "JSON", and yet there is no difference in the internal methods they support - we deliberately chose to name it as "JSON" using Math as the model - perhaps we could have avoided that - anyway, lets explicitly call out what specific values a host object may not set for its `[[Class]]` internal property.

JSON resolution regarding arguments

No change - JSON will serialize it as an object.

Strict mode implementation status

I've implemented the revised behaviour for 'arguments'; joining to the formal parameters is severed, and access to the caller and callee properties throws an exception.

Telnet access

Any update on providing telnet access for playing around with the JScript implementation? - No, not yet.

Meeting adjourned.