

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

**Ecma TC39-TG1
Phone conference
21th of August 2007**

Attending

- Jeff Dyer, Adobe Systems
- Brendan Eich, Mozilla Foundation
- Dave Herman, Northeastern University
- Dick Sweet, Adobe Systems
- Lars Hansen, Adobe Systems
- Chris Pine, Opera Software
- Graydon Hoare, Adobe Systems
- Jason Orendorff, Mozilla Foundation
- Francis Cheng, Adobe Systems

Agenda

- Proposal cleanup:
 - [Lars] Remove [the operators proposal](#). See [the discussion page](#) for a rationale.
 - [Lars] [The arrays proposal](#), much scaled back since last month, now mostly cementing things we've agreed on already.
 - [Lars] [The vectors proposal](#), a facility for FORTRAN-like arrays built on parameterized types.
 - [Lars] [The generic functions proposal](#), a proposal for multimethods with type dispatch.
 - [Lars] [The "map" proposal](#), a value → value map facility based on hash tables.
- Schedule, other administrivia
- Open tickets

Minutes

- Operators: we agree tentatively that we will remove the operators overloading proposal from the language, but we will await reactions from Michael Daumling (who championed the initial proposal). Jeff contacts Michael.
- Map:
 - general agreement that the proposal is desirable.
 - open issues about whether `IdentityMap` is well typed (the type of `K` is a recursively defined nominal type); self types may be involved but this is unclear.
 - unclear whether it should not rather be called `ObjectIdentityMap` or `ObjectMap` given that the interface is called `ObjectIdentity`.
- Parameterized types:
 - These are causing a lot of implementation headaches for Graydon. There are several problems, having to do with when type information is available, this is changing the structure of the reference implementation quite a lot, Graydon has a massive patch in the works.
 - Graydon: "One scale-back point is Ada generics"
 - Graydon: "Don't forget we have type *, and safety is guaranteed, so we may need generics less than Java did"
 - Brendan: safety and conciseness through cast elimination are not the only issues:

- Type-based extensibility (rather than name-based as in Python) for [iterators and generators](#).
- Container same-type-for-elements “safety”, which does require instanceof or cast again.