TC39 TG1 – January E4 Meeting

Date	January 15, 2002
Location	Microsoft Building 115/4179 Redmond, WA
Convener	Rok Yu (Microsoft)
Participants	John Schneider (AgileDelta/BEA) Peter Torr (Microsoft) Rok Yu (Microsoft) Waldemar Horwat (Netscape)

Agenda

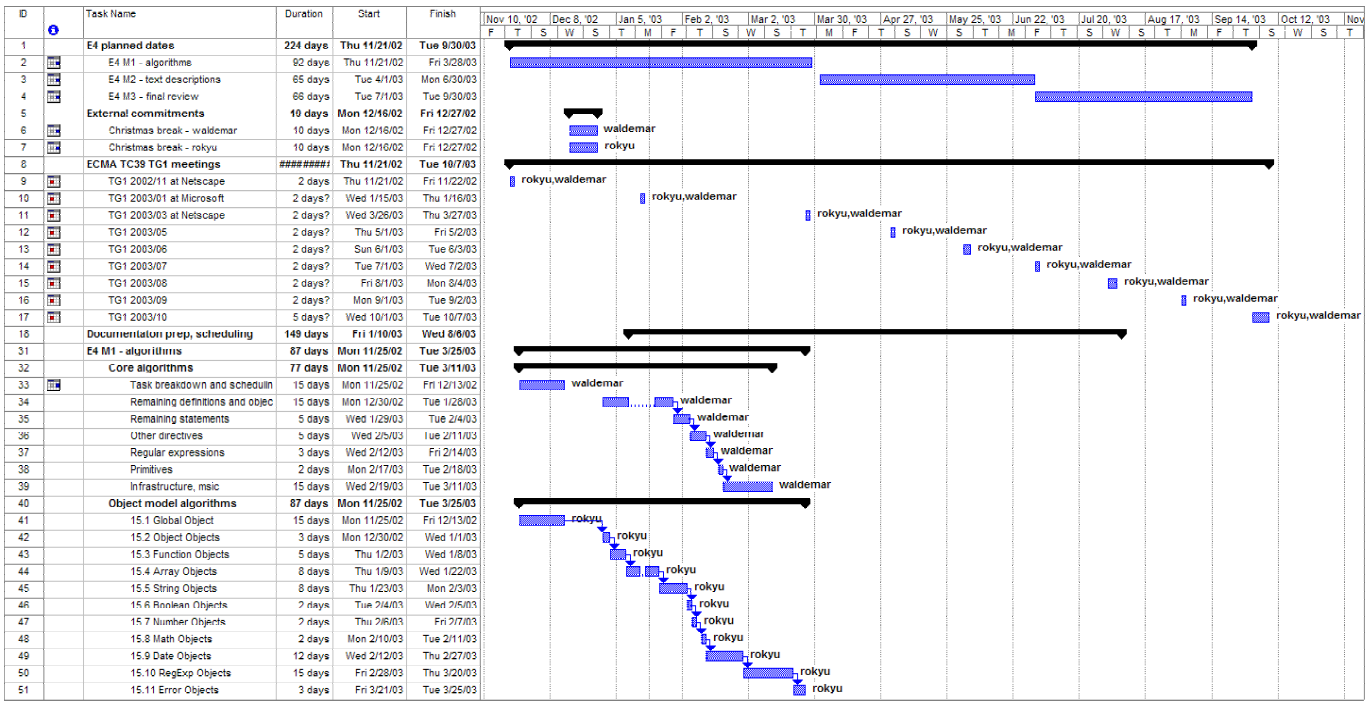
9:30	Gather
10:00	Review agenda Confirm dates for next meeting
10:15	Status review and schedule update
10:30	Discuss mutation of built in objects
12:00	Lunch
1:00	Namespaces and E4
1:30	Review of updates to semantics
2:30	Break
2:45	Review of updates to semantics

Next Meeting

The next TG1 meeting will be held at Netscape and is scheduled to follow the TC39 general meeting being held the morning of Wednesday March 26th at HP.

- E4: 3/26 afternoon, 3/27 morning
- E4X: 3/27 afternoon, 3/28 morning (afternoon if needed)

Status Review and Schedule Update



ID		Task Name	Duration	Start	Finish
1		**E4 planned dates**	**224 days**	**Thu 11/21/02**	**Tue 9/30/03**
2	▦	E4 M1 - algorithms	92 days	Thu 11/21/02	Fri 3/28/03
3	▦	E4 M2 - text descriptions	65 days	Tue 4/1/03	Mon 6/30/03
4	▦	E4 M3 - final review	66 days	Tue 7/1/03	Tue 9/30/03
5		**External commitments**	**10 days**	**Mon 12/16/02**	**Fri 12/27/02**
6	▦	Christmas break - waldemar	10 days	Mon 12/16/02	Fri 12/27/02
7	▦	Christmas break - rokyu	10 days	Mon 12/16/02	Fri 12/27/02
8		**ECMA TC39 TG1 meetings**	**#########**	**Thu 11/21/02**	**Tue 10/7/03**
9	▦	TG1 2002/11 at Netscape	2 days	Thu 11/21/02	Fri 11/22/02
10	▦	TG1 2003/01 at Microsoft	2 days?	Wed 1/15/03	Thu 1/16/03
11	▦	TG1 2003/03 at Netscape	2 days?	Wed 3/26/03	Thu 3/27/03
12	▦	TG1 2003/05	2 days?	Thu 5/1/03	Fri 5/2/03
13	▦	TG1 2003/06	2 days?	Sun 6/1/03	Tue 6/3/03
14	▦	TG1 2003/07	2 days?	Tue 7/1/03	Wed 7/2/03
15	▦	TG1 2003/08	2 days?	Fri 8/1/03	Mon 8/4/03
16	▦	TG1 2003/09	2 days?	Mon 9/1/03	Tue 9/2/03
17	▦	TG1 2003/10	5 days?	Wed 10/1/03	Tue 10/7/03
18		**Documentaton prep, scheduling**	**149 days**	**Fri 1/10/03**	**Wed 8/6/03**
31		**E4 M1 - algorithms**	**87 days**	**Mon 11/25/02**	**Tue 3/25/03**
32		**Core algorithms**	**77 days**	**Mon 11/25/02**	**Tue 3/11/03**
33	▦	Task breakdown and schedulin	15 days	Mon 11/25/02	Fri 12/13/02
34		Remaining definitions and objec	15 days	Mon 12/30/02	Tue 1/28/03
35		Remaining statements	5 days	Wed 1/29/03	Tue 2/4/03
36		Other directives	5 days	Wed 2/5/03	Tue 2/11/03
37		Regular expressions	3 days	Wed 2/12/03	Fri 2/14/03
38		Primitives	2 days	Mon 2/17/03	Tue 2/18/03
39		Infrastructure, msic	15 days	Wed 2/19/03	Tue 3/11/03
40		**Object model algorithms**	**87 days**	**Mon 11/25/02**	**Tue 3/25/03**
41		15.1 Global Object	15 days	Mon 11/25/02	Fri 12/13/02
42		15.2 Object Objects	3 days	Mon 12/30/02	Wed 1/1/03
43		15.3 Function Objects	5 days	Thu 1/2/03	Wed 1/8/03
44		15.4 Array Objects	8 days	Thu 1/9/03	Wed 1/22/03
45		15.5 String Objects	8 days	Thu 1/23/03	Mon 2/3/03
46		15.6 Boolean Objects	2 days	Tue 2/4/03	Wed 2/5/03
47		15.7 Number Objects	2 days	Thu 2/6/03	Fri 2/7/03
48		15.8 Math Objects	2 days	Mon 2/10/03	Tue 2/11/03
49		15.9 Date Objects	12 days	Wed 2/12/03	Thu 2/27/03
50		15.10 RegExp Objects	15 days	Fri 2/28/03	Thu 3/20/03
51		15.11 Error Objects	3 days	Fri 3/21/03	Tue 3/25/03

Waldemar is on track with his work items. For this meeting, he has updated the definitions related to object model and member dispatch along with statements.

Rok has not make progress on his work items. Microsoft is in the planning phase for JScript which is consuming his time.

Rok is concerned that attempting to get both E4 and E4X completed for the December GA meeting significantly increases the risk to delivering for both. Rok proposes evaluating the options for reducing the work load so that as a group, we can be comfortable that we will be able to succeed at hitting the December date with at least one standard.

Of the two efforts, Rok believes that standardizing E4X has more value for Microsoft's customers and is more consistent with Microsoft's understanding of the market and the use of ECMAScript as script language. He also believes that it is generally less work overall and thus more likely to be ready for the December GA. As such, one option he would like to consider is to build E4X on top of Edition 3 and focus on delivering it as Edition 4 of the standard.

An alternative option is to continue with the current plan and make harder cuts from E4. Waldemar believes that he can take on the work items currently assigned to Rok and still hit schedule by carving out an appropriate subset. Note that this is possible because work items assigned to Rok assumes 20% utilization.

Action items

- Waldemar to carve out the subset of E4 for next meeting.
- John, Waldemar will discuss alternatives with appropriate colleagues and will report back on where they stand.

Mutation of built-in objects

Last meeting, while discussing object model, Waldemar made us aware that there had been earlier agreement to nuke wrapper objects from the E4 spec. e.g. the following code would no longer generate an instance of a built in Number object.

```
var n = Object(10);
```

This raises the question as to what the prototype based inheritance semantics are for built in objects.

```
Object.prototype.foo = 5;
```

What is the value of Object.foo?

What is the value of Number.foo ?

Microsoft's JScript .NET implementation implements wrapper objects and unifies the prototype and primitive types by binding the identifier to different objects depending on context. As an example, in a type context (e.g. var x : Object), the identifier Object binds to the System.Object. In an instance context (e.g. var x = Object), the identifier binds to the prototype based Object constructor. A similar thing occurs with String, Number, etc. The prototype based types define methods that correspond to the system types so that they appear to be both to the user.

In either case, Object.foo returns 5 because Object is resolves to the constructor and has normal prototype semantics.

There is potential to reduce the amount of work in the spec by taking advantage of the nuke of wrapper objects. However, this cannot be done without a side effect that expandos added to prototypes will not show up on built in types. They will continue to show up on instance members.

There is tentative agreement to go with this approach.

Action items

- Rok to confirm there no other reasons Microsoft's implementation has wrapper objects.

Namespaces

Do we consider making namespaces a first class object?

Namespace declarations are currently defined to be static.

```
namespace n
n:: … etc.
```

John proposes we support namespaces bound to variables. Currently

```
var n = new Namespace();
```

… doesn't work because not a constant time constant.

```
const n = new Namespace();
```

In E4X, new namespaces come in at runtime from XML data sources and make sense to be mutable.

There were no immediate concerns about this and Waldemar believes that proposal can be made consistent with the current E4.

Review of E4 Semantics

Waldemar merged some sub-categories into SimpleInstance and introduced a vtable concept to class definitions. This was an informative process of understanding the changes Waldemar made.

Action items

9.1.6: Add URI field to namespace type

9.1.8: Delete super field from class

9.1.9: Rename parent field to super to avoid name clash with E4X parent

10.5.6: Add look up of properties for primitive values