| | |
|---|---|
| *Notes of the:* | *Meeting of Ecma TC39 ad hoc on Internationalization* |
| *held on:* | *14 December 2012* |

Location: Google, Mountain View, CA, USA

Attendees: Norbert (Mozilla), Mark Davis (Google, Unicode), Richard (Amazon), Suresh (Microsoft), Eric (Microsoft), Nebojsa (Google)

Minute taker: Nebojša Ćirić (Google)

# 1    Agenda:

Go through http://wiki.ecmascript.org/doku.php?id=globalization:meetings, presentation.pdf document and clarify what goes into v2.0 API and discuss each feature.

Names in () are to write strawman document in Globalization namespace in wiki.

# 2    Minutes

## 2.1    Normalization (Norbert)

- Add String.prototype.normalize to ES6 spec
- It has one parameter which is the type of normalization - NFC, NFD, NFKC, NFKD, NFKCCF and CF
- NFKCCF - Normalization + case fold, ignores ignorable characters
- CF is simple case fold, locale independent

## 2.2    Case conversion (Norbert)

- toLowerCase/toUpperCase would change in ES6 spec
- There are lower, upper (support full, simple is already supported in JS), title and case fold casings
- All except case fold are locale specific
- Sentence casing has 31 character exceptions (title casing) - propose a warning in the spec
- Case folding for sure, say CSS use - but it would be locale independent, which is a problem wih Turkish variant and some others - add it to String.prototype.normalize(CF)
- update toLowerCase/toUppercase, put a warrning in the spec about title case, and implement case folding in normalize method

## 2.3    Character properties (Norbert)

- RegExp needs a proposal - investigation in progress by Luke Hoben on how RE works in browsers
- Throw exceptions on unknown \Unicode chars if /u is present in RegExp
- Which properties to support and what syntax to use - probably Unicode level 1+ regex specification
- Luke and Norbert have to finish investigation before we can proceed
- Unicode level 1 required, level 2 is optional but we may merge some things from levels

## 2.4 MessageFormat (Cira)

- Template string is hard to implement without tooling
- Have separate proposal
- Provide samples for plural, select/gender
- Change existing Shanjian's proposal, eliminate positional placeholders {0}
- Sample date, number, people - code that formats date, number and gender/plural

## 2.5 Date time format (Mark - duration)

- relative dates, intervals, duration - some requests came in asking for them
- give people raw materials?
- or try guessing what to use?
- Duration may be low effort high impact (long, short - countdown)
- Add more formats to accepted date formats:
- weekday, month, day (willing to add)
- minute, second (we need a good use case for this one - better handled with duration)
- eras are missing (not able to support those for now - probably not - low priority for MS)
- Timezone support is mandatory
- how to specify generic/specific names for timezones - open issue whether to allow people to specify the request

Date picker API

- HTML input element (for date picker) may solve this, but if it gets removed from the spec we could do it
- Is anybody using non Gregorian picker?

## 2.6 Segmentation (Rich)

- Use cases - offline indexing, editor controls, text highlighting
- Open source code ahead!! - http://code.google.com/p/v8-i18n/source/browse/trunk/src/break-iterator.js

## 2.7 Display names (Jungshik)

- if we have time
- Microsoft can support whatever is present in the installed/current language pack + English
- Useful for list of countries, languages...
- Don't offer it or offer as a best effort - open issue

## 2.8 Number parsing (Cira)

- with delimiters
- but no currency, percent...

## 2.9 Alphabetic index (Mark)

- not a trivial amount data so better to have browser support

## 2.10 Script reordering (Norbert)

- Needs a list of scripts to order

- new extension to the locale id and option to the collator

## 2.11 Pseudo-numbering systems (Norbert)

- financial numbering systems should be resolved with actual system