Locale-sensitive case conversion
Richard Gillam

At the last meeting, it was noted that the case-conversion functions were the only locale-sensitive functions that didn’t have locale-independent counterparts. We agreed to make toLowerCase() and toUpperCase() locale-independent and add two new functions, toLocaleLowerCase() and toLocaleUpperCase(), with locale-sensitive behavior. The changes to the two existing functions were rolled into the big I18N proposal. The definitions of the two new functions are given here:

String.prototype.toLocaleLowerCase()
This function works exactly the same as toLowerCase(), except that its result is intended to yield the correct result for the host environment’s current locale, rather than a locale-independent result. There will only be a difference in the few cases (such as Turkish) where the rules for that language conflict with the regular Unicode case mappings.

NOTE The first parameter to this function is likely to be used in the next version of this standard; we recommend that implementations do not use this parameter position for anything else.

String.prototype.toLocaleUpperCase()
This function works exactly the same as toUpperCase(), except that its result is intended to yield the correct result for the host environment’s current locale, rather than a locale-independent result. There will only be a difference in the few cases (such as Turkish) where the rules for that language conflict with the regular Unicode case mappings.

NOTE The first parameter to this function is likely to be used in the next version of this standard; we recommend that implementations do not use this parameter position for anything else.