

Subject: Update proposed draft UTS 18
From: Mark Davis
Date: 2021-04-23

The following proposes resolution of some of the review notes in the proposed draft UTS #18, and some additional changes. Note that some of the clauses in UTS #18 were added after the last UTC meeting so that we could get early feedback; those are marked with **Review Note²**.

These changes are in the [working draft 3](#).

1. Fix subscripts

As per feedback from David Corbett, use $\langle S \rangle$ ($\langle S \rangle$) and $\langle P \rangle$ ($\langle P \rangle$) instead of subscript letter forms ($\langle S \rangle$ and $\langle P \rangle$, respectively). Reference: L2/21-069 item P427a.

2. Change text for literals

2.2.1 Character Classes with Strings

Change

```
:= '\q{ ' LITERAL* ' }'
```

to

```
:= '\q{ ' LITERAL* ( \ | ' LITERAL* ) * ' }'
```

Work the following text (from the review note) into the text above. Change the polarity so that use of | is encouraged, but can be omitted by implementations.

An implementation may add additional syntax to make sequences of literal strings more readable and compact, such as a separator to avoid repeating syntax. For example:

Original Notation	$[a-z \langle ch \rangle \langle sch \rangle \langle \text{ } \rangle \langle \text{ } \rangle \langle \text{ } \rangle]$
Compact Notation	$[a-z \langle ch sch \text{ } \text{ } \text{ } \rangle]$

Add text that some implementations may choose to use different notation for simplicity, in particular, they may allow the $\langle \rangle$ to be dropped, or may use $()$ instead of $\{ \}$

Then delete:

[Review Note²: Added information below on (a) usage of $\langle \rangle$ with properties and (b) possible syntax enhancements.

3. Other Review Note resolution

Delete the following review notes:

[Review Note²: the above table contains notation used in the rest of the document. While each symbol is defined on first use, it is useful to collect here for reference.

REVIEW NOTE: The revised rules limit the set of characters allowed to be escaped (compared with the previous versions rules) for better consistency with general practice.

[Review Note²: Interpreting the [^...] notation as *full complement* ($S \setminus A$) results in unexpected results and causes problems for backwards compatibility. The text has been changed above to make complement be explicitly *code point complement* ($\mathbb{P} \setminus A$).

[Review Note²: The above text now explicitly states that the Code Point Complement is used for \P and [:^.

[Review Note²: End of added material.

[Review Note²: This section now provides (a) the rationale for restricting complement to be code point complement and (b) a description of how to handle full complement if an implementation wants to support it.