L2/18-202

Re: Fixes to emoji data files for v12.0

From: ESC

Date: 2018

The following are suggested fixes for issues in the emoji data files (and related fixes to UTS #51). Most of them are fixes for problems reported by Yifán Wáng and Mathias Bynens.

emoji-sequences.txt

1. Fix the header

```
#   type_field: any of {Emoji_Combining_Sequence, Emoji_Flag_Sequence,
Emoji_Modifier_Sequence}
⇒
#   type_field:
#     Emoji_Keycap_Sequence
#     Emoji_Flag_Sequence
#     Emoji_Tag_Sequence
#     Emoji_Modifier_Sequence
```

2. In the subheaders, replace space by _ in the type names.

```
#     Emoji Keycap Sequence: ...
#     Emoji Flag Sequence: ...
#     Emoji Tag Sequence: ...
#     Emoji Modifier Sequence: …
```

emoji-zwj-sequences.txt

1. In the subheaders, replace space by _ in the type names.

```
# Emoji ZWJ Sequence: ...
⇒
# Emoji_ZWJ_Sequence: …
```

emoji-test.txt

1. Include skin-tones in the file (currently we include hair, but exclude skin tones). Both shouldn't be on the keyboard, but should have emoji presentation.
2. UTC issue: Consider making the regional-indicators be Emoji_Presentation=No
    a. Then the components that really need to be shown as emoji (skin tones and hair styles) are exactly those with Emoji_Presentation=No.

b. People use them as fancy letters, but then get random pairs as flags. There are circled/squared letters elsewhere in Unicode.

c. *ESC agrees*

3. Add “component” as a Status
4. Change Status “non-fully-qualified” to “partially-qualified” when the first character has Emoji_Presentation or is followed by a skin-tone modifier.
    a. Some classes of implementations can support the partially qualified but not non-fully-qualified. The difference is that from the first two code points, you already know that it is emoji, so you don’t have to “backtrack” later.
5. Fix the header in accordance with the changes, as below.

```
# Format
#   Code points; status # emoji name
#     Status
#       fully-qualified – see “Emoji Implementation Notes” in UTS #51
#       non-fully-qualified – see “Emoji Implementation Notes” in UTS #51
# Notes:
#   • This currently omits the 12 keycap bases, the 5 modifier characters, and 26 singleton Regional Indicator characters
#   • The file is in CLDR order, not codepoint order. This is recommended (but not required!) for keyboard palettes.
#   • The groups and subgroups are purely illustrative. See the Emoji Order chart for more information.
⇒
# Format: code points; status # emoji name
#     Code points – list of one or more hex code points, separated by spaces
#     Status
#       component           – a component of sequences that does not normally appear on keyboards.
#       fully-qualified     – in which every character that needs an emoji variant has one.
#       partially-qualified – other cases in which the first character has an emoji variant if it needs one.
#       non-fully-qualified – other emoji character or sequence
# Notes:
#   • This includes the emoji components that need emoji presentation (skin tone and hair) when isolated
#     and omits the components that need not have an emoji presentation when isolated.
#   • The file is in CLDR order, not codepoint order. This is recommended (but not required!) for keyboard palettes.
#   • The groups and subgroups are illustrative. See the Emoji Order chart for more information.
```

http://www.unicode.org/reports/tr51/

1. change multiple instances of the following

```
… The specific set of emoji sequences listed in the XXX.txt file [emoji-data] under the
```

`category YYY.`

⇒

`… The specific set of emoji sequences listed in the XXX.txt file [emoji-data] under the type_field YYY.`

`Example:`

ED-25. RGI emoji ZWJ sequence set — The specific set of emoji sequences listed in the emoji-zwj-sequences.txt file [emoji-data].

⇒

RGI emoji flag sequence set — The specific set of emoji sequences listed in the **emoji-sequences.txt** file [emoji-data] under the type_field **Emoji_Flag_Sequence**.

2. Add a definition for partially-qualified sequence after ***ED-19. non-fully-qualified emoji zwj sequence:***

   ED-19a. partially-qualified emoji zwj sequence — An emoji zwj sequence that is not a fully-qualified emoji zwj sequence.

3. Add a definition for Emoji_Sequence [Mathias Bynens]. I looked into this and here is the current situation regarding definitions.

<table>
<tr><th colspan="4">Well-Formed</th><th colspan="2">RGI</th></tr>
<tr><td>emoji_character</td><td rowspan="4">emoji_combining_sequence</td><td rowspan="5">emoji_core_sequence</td><td rowspan="7">emoji_sequence</td><td rowspan="2">basic emoji set</td><td></td></tr>
<tr><td>emoji_presentation_sequence</td><td></td></tr>
<tr><td>emoji_keycap_sequence</td><td>emoji keycap sequence set</td><td></td></tr>
<tr><td>emoji_modifier_sequence</td><td>emoji modifier sequence set</td><td></td></tr>
<tr><td>emoji_flag_sequence</td><td></td><td>RGI emoji flag sequence set</td><td rowspan="3">RGI sequence set</td></tr>
<tr><td>emoji tag sequence</td><td></td><td></td><td>RGI emoji tag sequence set</td></tr>
<tr><td>emoji_zwj_sequence</td><td></td><td></td><td>RGI emoji ZWJ sequence set</td></tr>
</table>

The definitions under RGI may have additional constraints put on them. For example, the basic emoji set doesn't include invalid emoji_presentation_sequences or Emoji_Component characters. The conclusion is that the definitions are mostly there, except that we need something that encompasses all the rows under "RGI" below. So that results in #4 and #5 below.

4. Add a data file **basic_emoji.txt** that provides that basic emoji set, and a reference to it in #def_basic_emoji_set.
   a. It should be possible to get the RGI sequence set from the data files. And it is, but not in a straightforward way; the missing piece is the basic emoji set. One can either access the test file, or one can construct the basic emoji set programmatically. Easier for implementers and less error-prone to just provide a specific list.

5. Add a new definition for **RGI set** to include the basic, keycap and modifier sets.
   ED-27. **RGI set** — The set of all sequences covered by ED-20, ED-21, ED-22, and ED-26.
   • This is the subset of all valid emoji recommended for general interchange.