

Subject: **A Proposal for Bidi Isolates in Unicode**  
 From: **Aharon (Vladimir) Lanin**  
 Date: **May 7, 2012**  
 Live document: <http://goo.gl/K6qtV>

## Background

Recently, the HTML and CSS standards have been expanded to provide two new features relevant to bidirectional text:

### 1. Isolation.

An inline element with the new 'isolate' value for the unicode-bidi style property is defined to affect the bidirectional ordering of the content around it as if it were U+FFFF, i.e. a neutral character, regardless of its actual content or the value of its direction style property. In this, it differs from the existing unicode-bidi:embed, which is based on the LRE and RLE Unicode characters. These affect the content around them as a strongly directional character, thus "sticking" to a number following the matching PDF character, even if separated from it by neutral characters. They also stick to other content of the same direction either preceding or following (with only neutrals in between.)

For the bidirectional ordering of the content within a unicode-bidi:isolate element, it is considered to be a separate bidi paragraph or sequence of paragraphs; the base direction of the paragraphs is determined from the element's direction property.

For non-inline elements, whose boundaries serve as bidi paragraph breaks, unicode-bidi:isolate has no visible effect.

Example 1: "`<span style="unicode-bidi:isolate;" dir="rtl">PURPLE PIZZA!</span>` (5 reviews)" in an LTR context is displayed as intended, i.e. as "!AZZIP ELPRUP (5 reveiews)". Compare to "PURPLE PIZZA! (5 reviews)" and "`<span dir="rtl">PURPLE PIZZA!</span>` - 5 reviews", both of which are displayed as "5) !AZZIP ELPRUP reviews)".

In this example, it is possible to achieve the desired display without using isolation by putting an LRM between the RTL phrase and the "5", but sticking LRMs or RLMs around opposite-direction phrases has a couple of problems. First, it requires knowing the directionality of the context into which a potentially opposite-direction phrase is being inserted, which may not be available to the application code layer inserting potentially opposite-direction data into a boilerplate message. More importantly, it is not always possible to achieve the same effect as isolation with LRMs or RLMs (which, by definition, are not neutral). Consider the case of an RTL book title:

Example 2: "ADVANCED html, css, AND php, PART 1". In its intended RTL directionality, with no special formatting, it comes out garbled, as "1 TRAP ,php DNA ,html, css DECNAVDA" (note the commas around the "html" and the order of "html" and "css"). Using isolation, "ADVANCED `<span style="unicode-bidi:isolate;">html</span>`, `<span style="unicode-bidi:isolate;">css</span>`, AND `<span style="unicode-bidi:isolate;">php</span>`, PART 1") results in the intended display of "1 TRAP ,php DNA ,css ,html DECNAVDA". But so does using an RLM after

the "html". But now consider this book title being displayed in an LTR context - without being declared RTL. With or without the RLM, "we used 'ADVANCED html&rlm;, css, AND php, PART 1'" is garbled as

"we used 'DECNAVDA html, css, DNA php, 1 TRAP'". But with isolation, "we used 'ADVANCED <span style="unicode-bidi:isolate;">html</span>, <span style="unicode-bidi:isolate;">css</span>, AND <span style="unicode-bidi:isolate;">php</span>, PART 1'" still comes out as intended: "we used '1 TRAP ,php DNA ,css ,html DECNAVDA'". It is impossible to achieve this display with any number of additional LRMs or RLMs.

Of course, putting an RLE ... PDF around the book title will fix its display. But we cannot always count on everyone doing the right thing, and the book title is more robust (i.e. is displayed correctly even when the person or application using it does not know how to deal with bidi) when it uses isolation instead of LRMs or RLMs.

## 2. First-strong automatic direction.

An element with the new 'plaintext' value for the unicode-bidi style property is defined to be just like unicode-bidi:isolate, except that the base direction of the paragraphs it "immediately" contains is determined from their content according to rules P2 and P3 of the Unicode bidirectional algorithm, i.e. from the first character with strong direction. (An element is said to "immediately contain" a paragraph of content if the element itself, but none of its descendants, both contains the entire bidi paragraph and is either a block container or a bidi-isolating inline.)

As opposed to unicode-bidi:isolate, unicode-bidi:plaintext does have a visible effect on block elements. For example, in

```
<div style="unicode-bidi:plaintext; white-space:pre">
LINE 1.
line 2.
LINE 3.
</div>
```

the first and the third lines would be displayed in RTL, but the second in LTR.

## The Need

It would be very helpful if the Unicode Bidirectional Algorithm's features included the capability to isolate opposite-direction (or potentially opposite-direction) phrases in the manner offered by unicode-bidi:isolate. Similarly, it would be nice to be able to explicitly ask it to guess the directionality of a phrase (using the usual first-strong algorithm), without regard to the directionality of the surrounding paragraph, as is offered by unicode-bidi:plaintext. These capabilities are important for two reasons:

1. These features are just as necessary in plain text as in HTML.
2. It would greatly simplify things for browsers implementing unicode-bidi:isolate and unicode-bidi:plaintext, which currently need to use complicated and error-prone work-arounds.

# The Proposal

Define three new Unicode formatting code points:

- LRI: marks the beginning of a left-to-right isolate.
- RLI: marks the beginning of a right-to-left isolate.
- FSI: marks the beginning of a first-strong isolate.

Each would be matched with a PDF. Obviously, isolates would be allowed to nest, just like embeddings. The visual ordering within an isolate is the same as if it used LRE or RLE, as appropriate, instead of the characters above. But the the visual ordering of the content outside the isolate is the same as if the isolate were an ON-class character (e.g. U+FFFC).

Rules P2 and P3 for determining base direction (the first-strong algorithm) have to be modified to skip over the content in an isolate (i.e. everything between one of the new characters above and its matching PDF or end of paragraph, whichever comes first). This is part of making an isolate behave as a neutral character for the purposes of the visual ordering of the content surrounding it. This change to P2 and P3 would apply both when determining the base direction of a paragraph and when determining the embedding direction of an FSI isolate.

## Difficulty 1:

The astute reader will notice that this definition differs from that of the CSS features above in an important way. The CSS / HTML features allow paragraph breaks inside the isolate, and define the content inside the isolate as forming a sequence of paragraphs in this case. Thus:

1. The paragraph break does not end the isolate.
2. The paragraph break occurs within the isolate, but not in the surrounding content, which still behaves as if the whole isolate were just a neutral character.
3. An extra PDF within the isolate does not end it prematurely and the effects of a missing one do not extend beyond the isolate, or even beyond the paragraph missing the PDF.

This is all very different from the new characters as proposed, which are strictly inline (just like LRE and RLE). The differences mean that the CSS features as currently defined could not be implemented in terms of the proposed characters.

One way to handle this difficulty is to change the proposal so the effects of the new characters do not end at a paragraph break. However, this would be a radical departure from existing Unicode bidi algorithm rules. Nor would it resolve the difference regarding missing and extra PDFs. Most importantly, I am not sure that having the paragraph outside the isolate continue uninterrupted around an isolate that contains a paragraph break really makes much sense. The continuation is too non-local to make much sense to the reader.

I think that it would be better to instead try to change the CSS definition (which is currently still a draft) such that a paragraph break within an isolate would also end the paragraph outside the isolate. However, by existing, unrelated CSS rules, the inline directional scopes interrupted by the paragraph break would then be reopened after the paragraph break. Thus, the paragraph

break would end one isolate - but then open another one that will be closed by the isolate's end tag. The new characters as proposed above could be used to implement such a definition. Isolates would not longer deal with missing/extra PDFs in CSS, but I believe that those can and should be handled in a different manner unrelated to isolates (having CSS remove extra PDFs and add missing ones automatically). This is all a matter for the CSS WG, of course, but how that goes may have bearing on the proposal for the new characters.

## **Difficulty 2:**

The Unicode bidirectional algorithm states that LRE, RLE, and PDF should be removed or ignored after the levels have been computed. If the same approach is taken to the new characters and the PDFs matching them, however, the isolation will be broken when the isolate is followed or preceded immediately by another isolate, a character of the same direction as the isolate, and LRE|RLE...PDF, or, under certain circumstances, even just a number. For example, in a paragraph with base level 0, "[RLI]A[PDF]B" would give both the A and the B level 1. If the formatting characters are then removed, the A and B will be reordered to BA. This would not have happened if the "[RLI]A[PDF]" were a neutral character - to which it is supposed to be equivalent, as far as the ordering of the content around it is concerned. A way has to be found to handle this. I have never been a fan of the rule removing the formatting characters, so I would not mind if it were removed or changed (while retaining backward compatibility for LRE, RLE, LRO, and RLO).