**Grammar (Proposed Revised)**

Addresses issues with use of word “representation” too loosely.

Aligns more with the description of how things have to work, as Nils must be checked early. (per action 140 stuff)

Breaks out overused Initiator and Terminator regions into Nil, Empty, Sequence, Choice, and Element variants. To me this makes things much clearer that the initiator of a sequence is NOT the one the element specifies, but must be specified on the sequence itself.

**Updated by Steve:**

Use ‘Rep’ in a manner consistent with action 140 definitions

Use ‘Content’ consistently

Separate simple and complex so that empty v normal ordering can be correctly shown

Element f

raming and content cells merged

Explicitly include empty value

Not sure why we have separate Initiator and Terminator regions. This is not consistent with other properties shared between elements, sequences and choices.

**Updated by Mike: Changes based on 2012-07-25 WG Call**

Eliminated SequenceInitiator/Terminator, Choice, Element variants on these also.

Refactored Left/Right Framing. Made Empty left/right framing productions for parallel structure

.

Removed ***EmptyValue*** terminal from grammar. Was confusing additional use of Empty where we did NOT mean EmptyElement. This meant “no data”. Right way to say that in a grammar is to have nothing at all. (However, we have this ***Absent*** terminal....?? We need one way in the grammar to say that there are no representation bits.)

Changed Empty to EmptyElement, Nil to NilElement

**Updated by Steve:**

Removed EmptyElementRep for symmetry.

Added ‘Element’ into AbsentRep to match Empty and Nil.

Corrected EmptyElementRepXxxxFraming to EmptyElementXxxxFraming

Made NilElementLiteralCharacter plural so NilElementLiteralCharacter**s**

Personally I think we need ***EmptyValue***, because we have ***Absent***

I think NilElementLiteralValue, NilElementLogicalValue and NilElementLiteralCharacters should drop the ‘Element’, we are talking about values now so the qualification is not needed.

MikeB (7/31)

Shortened NilElementLiteralValue et al. Per steve's last comment above.

A point of clarification: we need Absent because it is part of an alternative composition. One only can remove zero-length terminals if they are in sequential compositions.

|  |  |
| --- | --- |
| **Notes (not for final spec perhaps)** | **Productions** |
| Top Level | Document = ***UnicodeByteOrderMark*** Element  Element =  SimpleElement | ComplexElement  SimpleElement = SimpleLiteralNilElementRep | SimpleEmptyElementRep |  SimpleNormalRep | SimpleAbsentElementRep  ComplexElement = ComplexLiteralNilElementRep | ComplexNormalRep |  ComplexEmptyElementRep | ComplexAbsentElementRep |
| Absent Element | SimpleAbsentElementRep = ***Absent***  ComplexAbsentElementRep = ***Absent*** |
| Empty Element | SimpleEmptyElementRep = EmptyElementLeftFraming EmptyElementRightFraming  ComplexEmptyElementRep = EmptyElementLeftFraming EmptyElementRightFraming  EmptyElementLeftFraming = LeadingAlignment ***EmptyElementInitiator*** PrefixLength  EmptyElementRightFraming = ***EmptyElementTerminator*** TrailingAlignment |
| Literal Nil Element | SimpleLiteralNilElementRep = NilElementLeftFraming [***NilLiteralCharacters*** |  NilElementLiteralContent] NilElementRightFraming  ComplexLiteralNilElementRep = NilElementLeftFraming NilElementRightFraming  NilElementLeftFraming = LeadingAlignment ***NilElementInitiator*** PrefixLength  NilElementRightFraming = ***NilElementTerminator*** TrailingAlignment  NilElementLiteralContent = ***LeftPadding*** ***NilLiteralValue*** RightPadOrFill |
| Normal Element | SimpleNormalRep = LeftFraming PrefixLength SimpleContent RightFraming  ComplexNormalRep =  LeftFraming PrefixLength ComplexContent ***ElementUnused***    RightFraming  LeftFraming = LeadingAlignment ***Initiator***  RightFraming = ***Terminator*** TrailingAlignment  PrefixLength = SimpleContent | PrefixPrefixLength SimpleContent  PrefixPrefixLength = SimpleContent  SimpleContent =  ***LeftPadding*** [ ***NilLogicalValue | SimpleValue*** ]RightPadOrFill  ComplexContent = Sequence | Choice |
| Complex Content Regions | Sequence = LeftFraming SequenceContent RightFraming  SequenceContent = [ ***PrefixSeparator***EnclosedContent [ ***Separator*** EnclosedContent ]\*  ***PostfixSeparator***  Choice = LeftFramingChoiceContent RightFraming  ChoiceContent = [ EnclosedContent ] ***ChoiceUnused***  EnclosedContent = [ Element | Array | Sequence | Choice ]  Array = [ Element [ ***Separator*** Element ]\* [ ***Separator*** StopValue] ]  StopValue = SimpleElement |
| Details of Other Framing | LeadingAlignment = ***LeadingSkip*** ***AlignmentFill***  TrailingAlignment = ***TrailingSkip***  RightPadOrFill = ***RightPadding*** | ***RightFill*** |

Notes:

* For a simple element, if nilKind is ‘literalCharacter’ the nil comparison occurs before trimming when parsing, otherwise it occurs after trimming. See errata 2.8. *This is modelled explicitly in the grammar now. So this note isn’t needed.*