

JsonML - Syntax

Syntax Reasoning

JsonML was designed for lossless roundtrip conversion between XML and JSON and to handle "mixed-mode content" (i.e., textual data outside of or next to elements). Key considerations were lossless two-way conversion and compactness of form. JsonML handles namespaces in the same way that XML 1.0 does: attributes for xmlns and tag names with colons.

JsonML was never intended to be the way that everything that could be expressed in XML should be expressed in JSON. JsonML is the only transformation which can safely encode "mixed-mode" content in XML, which is a key characteristic in a markup language such as XHTML. Many other transformations exist for representing XML in a JSON format.

Grammar (BNF)

The following grammar represents how XML-based markup (e.g. XHTML) is encoded into JsonML. As per JSON, whitespace is permitted between tokens.

```

element
  = '[' tag-name ',' attributes ',' element-list ']'
  | '[' tag-name ',' attributes ']'
  | '[' tag-name ',' element-list ']'
  | '[' tag-name ']'
  | string
  ;
tag-name
  = string
  ;
attributes
  = '{' attribute-list '}'
  | '{' '}'
  ;
attribute-list
  = attribute ',' attribute-list
  | attribute
  ;
attribute
  = attribute-name ':' attribute-value
  ;
attribute-name
  = string
  ;
attribute-value
  = string
  | number
  | 'true'
  | 'false'
  | 'null'
  ;
element-list
  = element ',' element-list
  | element
  ;

```

"Array Form"? "Object Form"?

In a JSON user group thread, Douglas Crockford incorrectly used the term "JsonML" to describe two variants: the "array form" and "object form". This was a misuse of the term JsonML which has always *only* stood to mean what Crockford referred to as the "array form".