|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Feature*** | ***XML DTD*** | ***(W3C) XML Schemas*** | ***RELAX NG*** | ***Schematron*** |
| **General overview** | A XML structure definition with a lisr of legal elements | An object-oriented XML schema language | A schema language created by unifying RELAX core and TREX | a rules-based XML schema language |
| **Grammar** | posses its own compact but nonXML gramma | object-like, XML syntax | both an XML syntax and a compact nonXML syntax | XML syntax |
| **Datatyping** | no, (weak, only applies on attributes) | yes | plugged from W3C XML Schema and others | not directly (can be implemented with user-made rules) |
| **Support for XML namespaces** | none | yes | yes | yes |
| **Can directly partner with other schema languages** | no | no | partially (with a separate datatyping language) | yes (can be embedded inside XML Schema or RELAX NG) |
| **Post-Schema-Validation-Infoset** | yes | yes | no | not directly |
| **Complexity** | intermediate | quite complex (definitions require considerable expertise) | relatively simple | easy to learn (only six basic elements) but needs XPath knowledge (and XSLT) |
| **Can express nondeterminism** | no | no | yes | not directly |
| **Rules expression** | no | no (can only use regular expressions do constraint data values) | no (can only use regular expressions do constraint data values) | yes, using XPath |
| **Data structure description** | yes | yes | yes | only using usermade check rules  (not directly) |
| **Data integrity (identifiers, references)** | yes | yes | using features of an external datatype system (as W3C XML Schema) | only using usermade check rules (not directly) |
| **Overall flexibility** | poor | good (but weak support for unordered content) | high for structures | top, but all must be defined by user |
| **Notes** | even if widespread, it is probably going to disappear because of newer and more powerful schema languages | a XML Schema is relatively easy to extend and good for data-oriented applications | it relies on both strong mathematical theory underlying regular expressions and solid theoretical basis | it is based on a simple idea : finding nodes and checking properties on them |