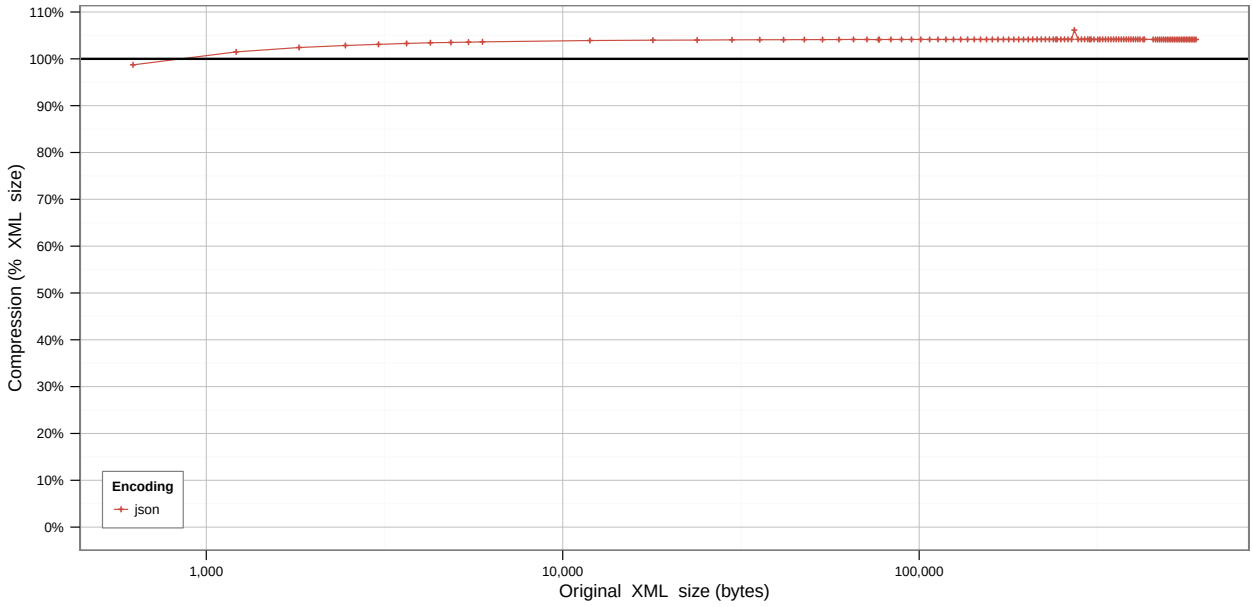
XML/JSON Analysis Template

Results for OpenWeatherMap Use Case

Plaintext Comparisons

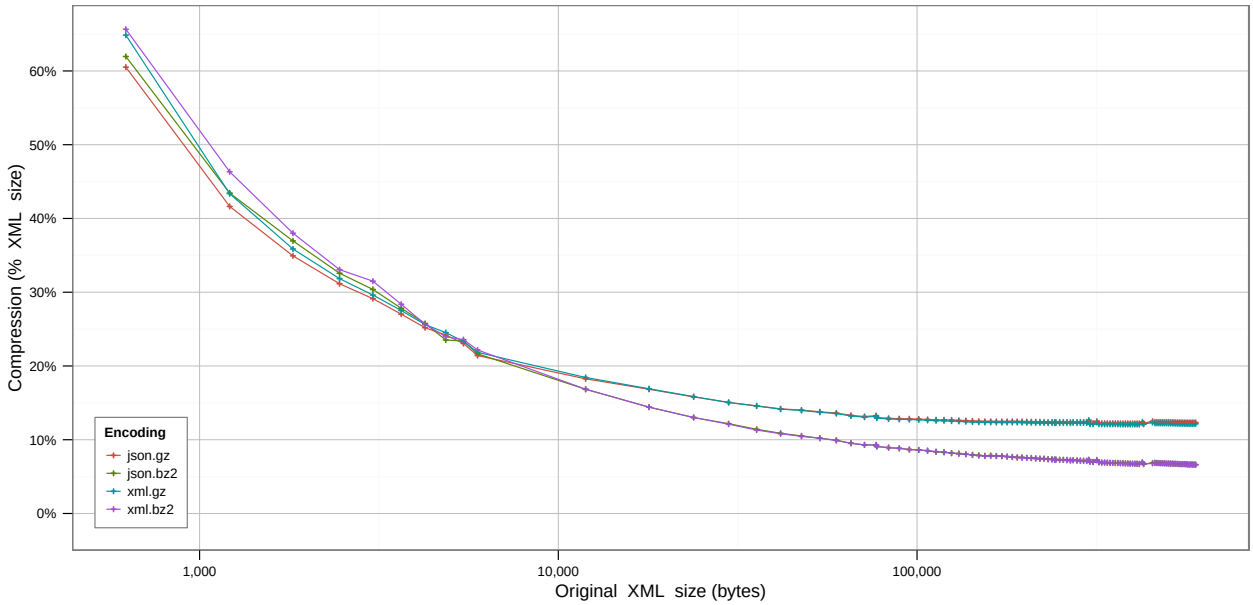
A. How do JSON and XML compare when plaintext-encoded?

```
## [1] "Series:   json"
## [1] "Baseline:  xml"
##       json
##  Min.   :0.9872
##  1st Qu.:1.0413
##  Median :1.0414
##  Mean   :1.0401
##  3rd Qu.:1.0416
##  Max.   :1.0615
```



B. How do JSON and XML compare when compressed with conventional compression algorithms?

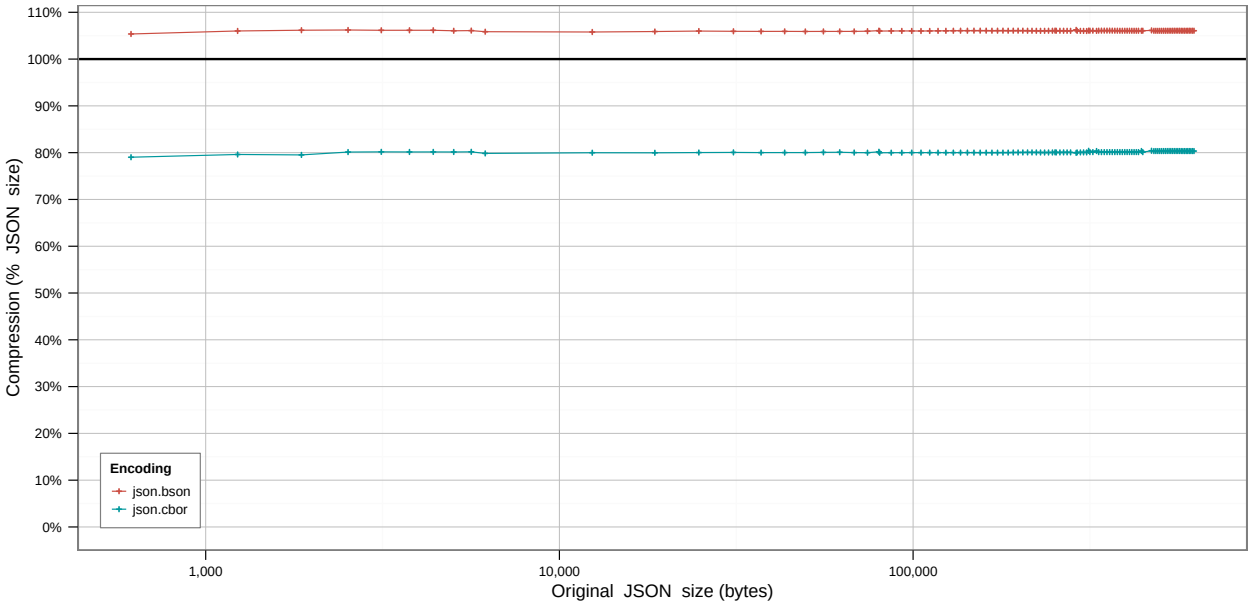
```
## [1] "Series:   json.gz, json.bz2, xml.gz, xml.bz2"
## [1] "Baseline:  xml"
##     json.gz          json.bz2          xml.gz          xml.bz2
##  Min.   :0.1221   Min.   :0.06609   Min.   :0.1206   Min.   :0.06598
##  1st Qu.:0.1234   1st Qu.:0.06815   1st Qu.:0.1216   1st Qu.:0.06795
##  Median :0.1241   Median :0.07278   Median :0.1228   Median :0.07258
##  Mean   :0.1444   Mean   :0.10046   Mean   :0.1441   Mean   :0.10125
##  3rd Qu.:0.1275   3rd Qu.:0.08509   3rd Qu.:0.1263   3rd Qu.:0.08519
##  Max.   :0.6051   Max.   :0.61958   Max.   :0.6485   Max.   :0.65650
```



JSON-Specific Exploratory

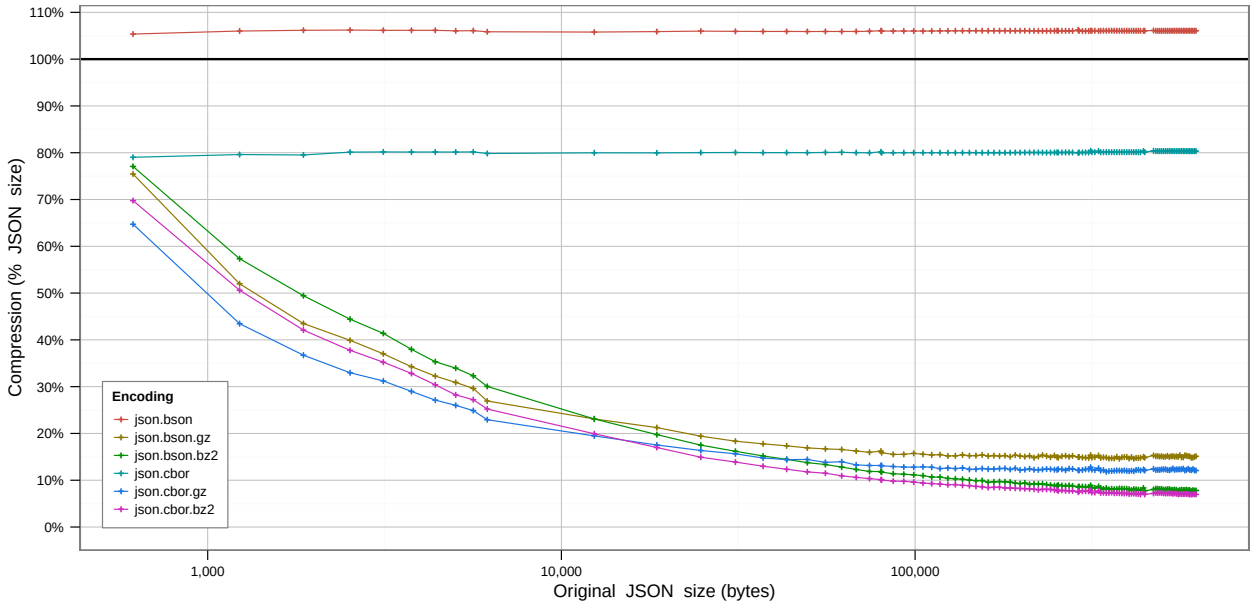
C. Which binary encoding of JSON is most compact?

```
## [1] "Series:   json.bson, json.cbor"
## [1] "Baseline:  json"
##    json.bson       json.cbor
##  Min.   :1.054   Min.   :0.7902
##  1st Qu.:1.060   1st Qu.:0.8002
##  Median :1.061   Median :0.8011
##  Mean   :1.060   Mean   :0.8011
##  3rd Qu.:1.061   3rd Qu.:0.8034
##  Max.   :1.062   Max.   :0.8040
```



D. For binary JSON formats, does post-compression with conventional compression algorithms improve compactness?

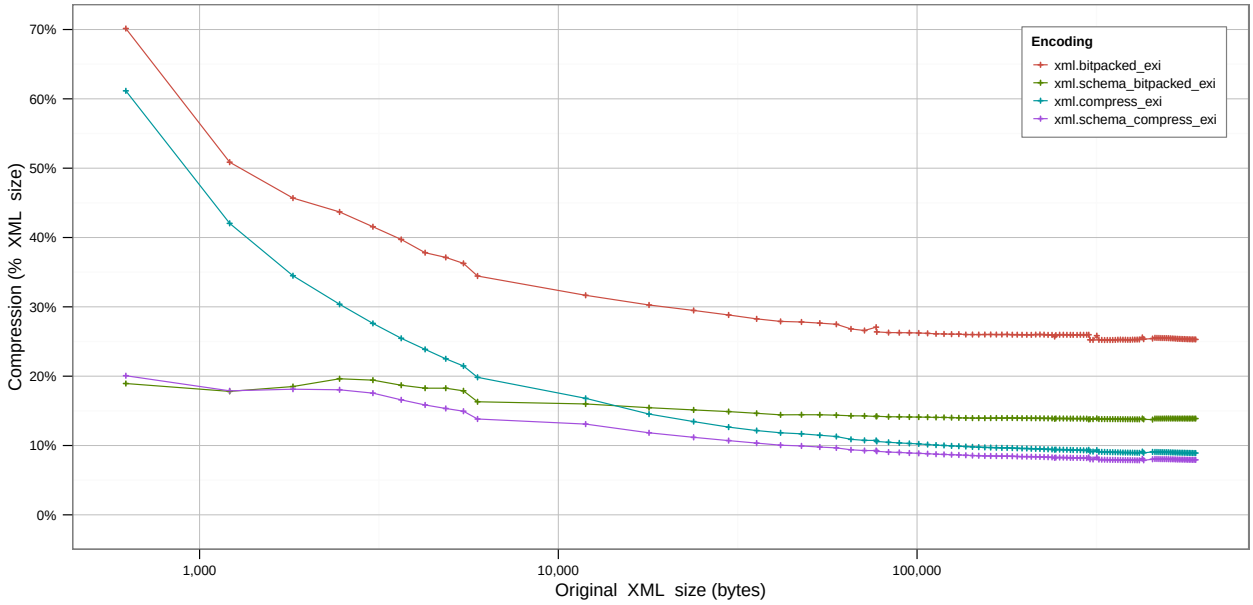
```
## [1] "Series:   json.bson, json.bson.gz, json.bson.bz2, json.cbor, json.cbor.gz, json.cbor.bz2"
## [1] "Baseline:  json"
##    json.bson      json.bson.gz    json.bson.bz2       json.cbor
##  Min.   :1.054   Min.   :0.1459   Min.   :0.07670   Min.   :0.7902
##  1st Qu.:1.060   1st Qu.:0.1494   1st Qu.:0.08077   1st Qu.:0.8002
##  Median :1.061   Median :0.1516   Median :0.08815   Median :0.8011
##  Mean   :1.060   Mean   :0.1770   Mean   :0.12691   Mean   :0.8011
##  3rd Qu.:1.061   3rd Qu.:0.1547   3rd Qu.:0.10678   3rd Qu.:0.8034
##  Max.   :1.062   Max.   :0.7545   Max.   :0.77073   Max.   :0.8040
##   json.cbor.gz    json.cbor.bz2
##  Min.   :0.1181   Min.   :0.06952
##  1st Qu.:0.1218   1st Qu.:0.07210
##  Median :0.1233   Median :0.07733
##  Mean   :0.1458   Mean   :0.11091
##  3rd Qu.:0.1278   3rd Qu.:0.09282
##  Max.   :0.6472   Max.   :0.69756
```



EXI Exploratory

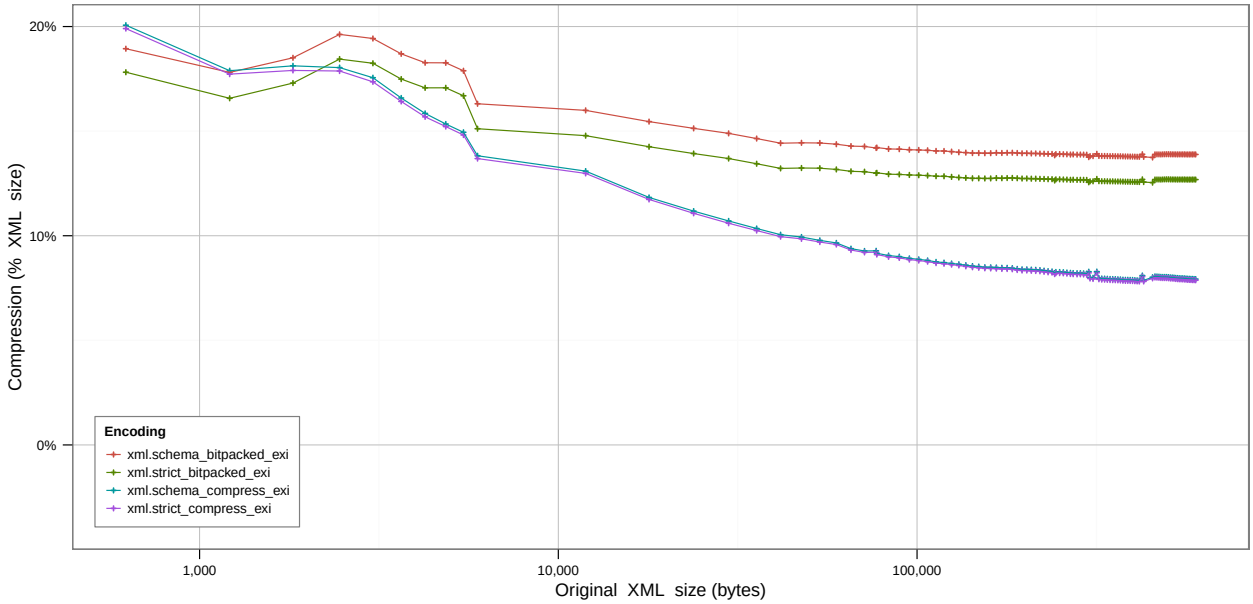
E. How do the primary EXI modes compare for schemaless & schema-informed encodings?

```
## [1] "Series:   xml.bitpacked_exi, xml.schema_bitpacked_exi, xml.compress_exi, xml.schema_compress_exi"
## [1] "Baseline:  xml"
##  xml.bitpacked_exi xml.schema_bitpacked_exi xml.compress_exi
##  Min.   :0.2521    Min.   :0.1374           Min.   :0.08912
##  1st Qu.:0.2536    1st Qu.:0.1388           1st Qu.:0.09023
##  Median :0.2596    Median :0.1390           Median :0.09382
##  Mean   :0.2761    Mean   :0.1440           Mean   :0.11635
##  3rd Qu.:0.2618    3rd Qu.:0.1408           3rd Qu.:0.10140
##  Max.   :0.7014    Max.   :0.1963           Max.   :0.61156
##  xml.schema_compress_exi
##  Min.   :0.07864
##  1st Qu.:0.07971
##  Median :0.08258
##  Mean   :0.09233
##  3rd Qu.:0.08820
##  Max.   :0.20064
```



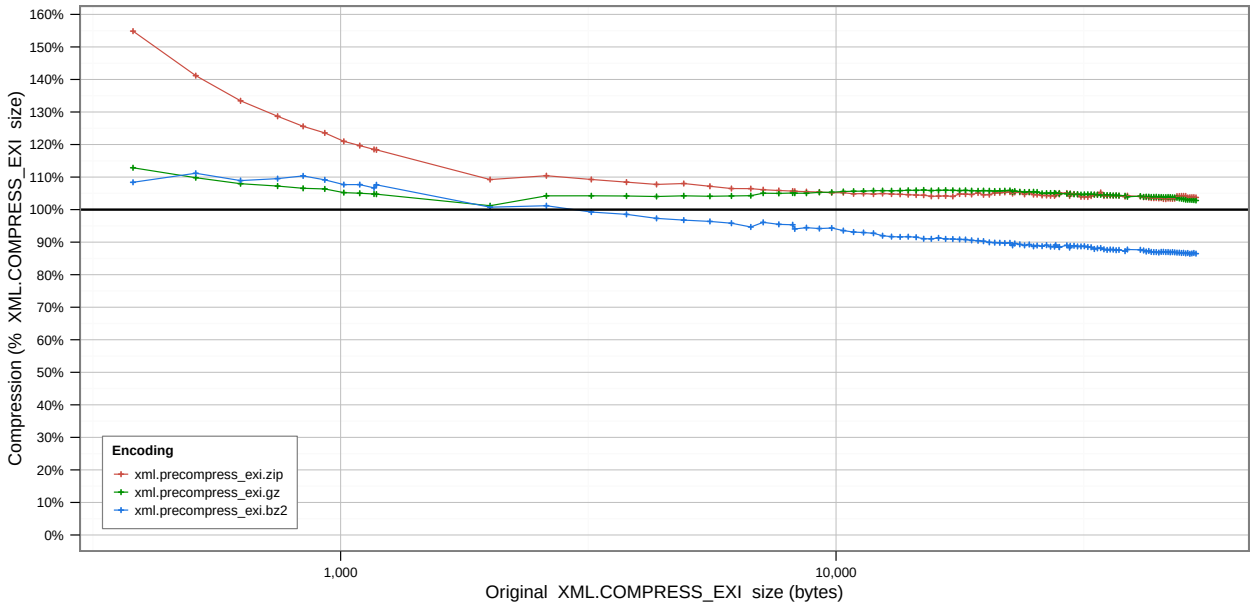
F. Does the ‘strict’ option significantly improve compaction for schema-informed encodings?

```
## [1] "Series:   xml.schema_bitpacked_exi, xml.strict_bitpacked_exi, xml.schema_compress_exi, xml.stric
## [1] "Baseline:  xml"
##  xml.schema_bitpacked_exi xml.strict_bitpacked_exi xml.schema_compress_exi
##  Min.   :0.1374           Min.   :0.1254           Min.   :0.07864
##  1st Qu.:0.1388           1st Qu.:0.1267           1st Qu.:0.07971
##  Median :0.1390           Median :0.1269           Median :0.08258
##  Mean   :0.1440           Mean   :0.1320           Mean   :0.09233
##  3rd Qu.:0.1408           3rd Qu.:0.1288           3rd Qu.:0.08820
##  Max.   :0.1963           Max.   :0.1844           Max.   :0.20064
##  xml.strict_compress_exi
##  Min.   :0.07813
##  1st Qu.:0.07917
##  Median :0.08203
##  Mean   :0.09165
##  3rd Qu.:0.08761
##  Max.   :0.19904
```



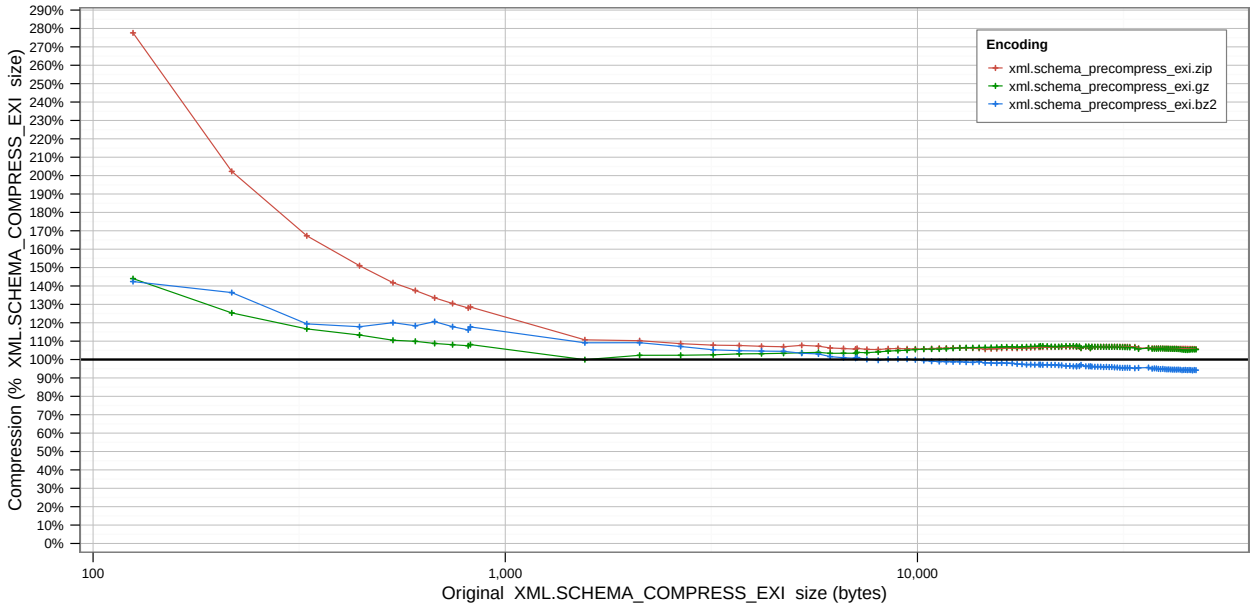
G. Do any of the tested conventional compression algorithms perform better on a schemaless, precompress EXI document than the standard DEFLATE?

```
## [1] "Series:   xml.precompress_exi.zip, xml.precompress_exi.gz, xml.precompress_exi.bz2"
## [1] "Baseline:  xml.compress_exi"
##  xml.precompress_exi.zip xml.precompress_exi.gz xml.precompress_exi.bz2
##  Min.   :1.032           Min.   :1.012          Min.   :0.8641
##  1st Qu.:1.042           1st Qu.:1.041          1st Qu.:0.8762
##  Median :1.046           Median :1.049          Median :0.8908
##  Mean   :1.070           Mean   :1.049          Mean   :0.9175
##  3rd Qu.:1.053           3rd Qu.:1.057          3rd Qu.:0.9311
##  Max.   :1.549           Max.   :1.129          Max.   :1.1118
```



H. Do any of the tested conventional compression algorithms perform better on a schema-informed, precompress EXI document than the standard DEFLATE?

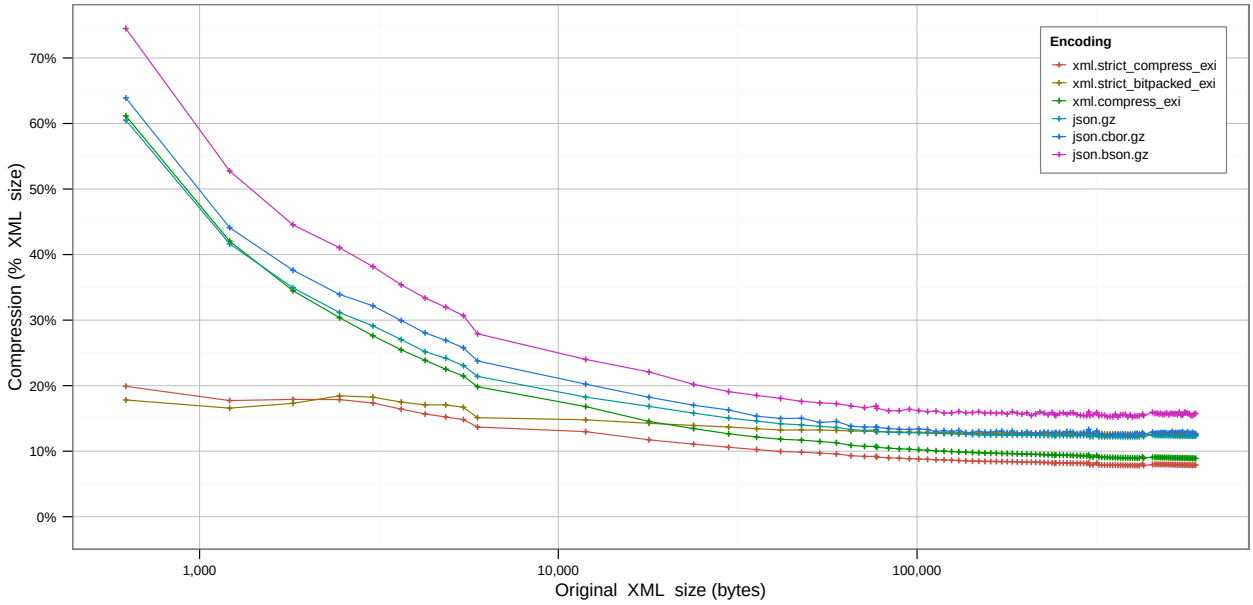
```
## [1] "Series:   xml.schema_precompress_exi.zip, xml.schema_precompress_exi.gz, xml.schema_precompress
## [1] "Baseline:  xml.schema_compress_exi"
##  xml.schema_precompress_exi.zip xml.schema_precompress_exi.gz
##  Min.   :1.054                  Min.   :0.9994
##  1st Qu.:1.060                  1st Qu.:1.0545
##  Median :1.064                  Median :1.0645
##  Mean   :1.114                  Mean   :1.0671
##  3rd Qu.:1.070                  3rd Qu.:1.0700
##  Max.   :2.776                  Max.   :1.4400
##  xml.schema_precompress_exi.bz2
##  Min.   :0.9407
##  1st Qu.:0.9547
##  Median :0.9709
##  Mean   :0.9983
##  3rd Qu.:0.9986
##  Max.   :1.4240
```



Binary-comparisons

I. Which binary format is the most compact?

```
## [1] "Series:   xml.strict_compress_exi, xml.strict_bitpacked_exi, xml.compress_exi, json.gz, json.cb
## [1] "Baseline:  xml"
##  xml.strict_compress_exi xml.strict_bitpacked_exi xml.compress_exi
##  Min.   :0.07813         Min.   :0.1254           Min.   :0.08912
##  1st Qu.:0.07917         1st Qu.:0.1267           1st Qu.:0.09023
##  Median :0.08203         Median :0.1269           Median :0.09382
##  Mean   :0.09165         Mean   :0.1320           Mean   :0.11635
##  3rd Qu.:0.08761         3rd Qu.:0.1288           3rd Qu.:0.10140
##  Max.   :0.19904         Max.   :0.1844           Max.   :0.61156
##     json.gz          json.cbor.gz     json.bson.gz
##  Min.   :0.1221   Min.   :0.1230   Min.   :0.1520
##  1st Qu.:0.1234   1st Qu.:0.1269   1st Qu.:0.1556
##  Median :0.1241   Median :0.1286   Median :0.1579
##  Mean   :0.1444   Mean   :0.1512   Mean   :0.1836
##  3rd Qu.:0.1275   3rd Qu.:0.1331   3rd Qu.:0.1611
##  Max.   :0.6051   Max.   :0.6388   Max.   :0.7448
```



J. Do any of the binary formats offer improvement for a network already using gzip?

```
## [1] "Series:   xml.strict_compress_exi, xml.strict_bitpacked_exi, xml.compress_exi, json.gz, json.cb
## [1] "Baseline:  xml.gz"
##  xml.strict_compress_exi xml.strict_bitpacked_exi xml.compress_exi
##  Min.   :0.3069          Min.   :0.2748           Min.   :0.7333
##  1st Qu.:0.6503          1st Qu.:1.0151           1st Qu.:0.7419
##  Median :0.6569          Median :1.0348           Median :0.7653
##  Mean   :0.6575          Mean   :0.9811           Mean   :0.7850
##  3rd Qu.:0.6827          3rd Qu.:1.0418           3rd Qu.:0.8028
##  Max.   :0.7072          Max.   :1.0439           Max.   :0.9696
##     json.gz         json.cbor.gz     json.bson.gz
##  Min.   :0.9332   Min.   :0.9851   Min.   :1.149
##  1st Qu.:1.0089   1st Qu.:1.0389   1st Qu.:1.271
##  Median :1.0112   Median :1.0476   Median :1.279
##  Mean   :1.0076   Mean   :1.0492   Mean   :1.279
##  3rd Qu.:1.0122   3rd Qu.:1.0551   3rd Qu.:1.289
##  Max.   :1.0153   Max.   :1.1048   Max.   :1.315
```