



LT Bruce Hill

MAKING BIG FILES SMALL AND SMALL FILES TINY

- JavaScript Object Notation (JSON) is a common alternative to XML in web applications
- JSON is a plaintext data-interchange format based on JavaScript code
- JSON has compact binary encodings analogous to EXI:
 - CBOR
 - BSON
- ***Research Question: Is EXI more compact than CBOR and BSON?***

- W3C and previous NPS research measured EXI performance on XML up to 100MB
- Large data dumps can easily exceed that
- ***Research Question: How does EXI (but not CBOR/BSON) perform on files from 100MB - 4GB?***

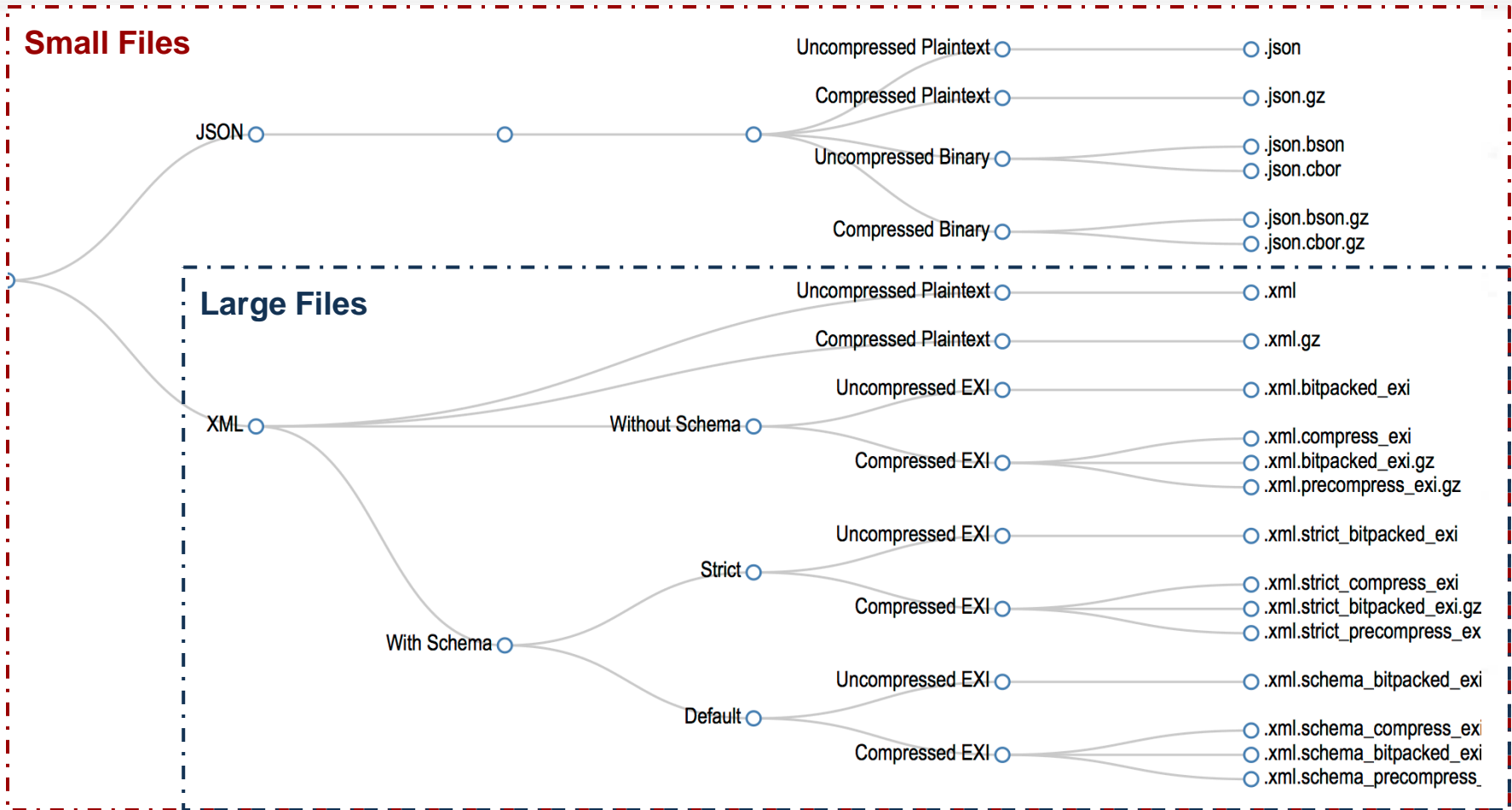
Use Case Focus

- Compression results *across multiple use cases* look different from results for multiple files *within a single use case*
- Select a few use cases and study them in-depth

Configuration Focus

- EXI has many configuration options that affect
 - Compactness
 - Processing speed
 - Memory footprint
 - Fidelity
- XML Schema affects EXI compression as well

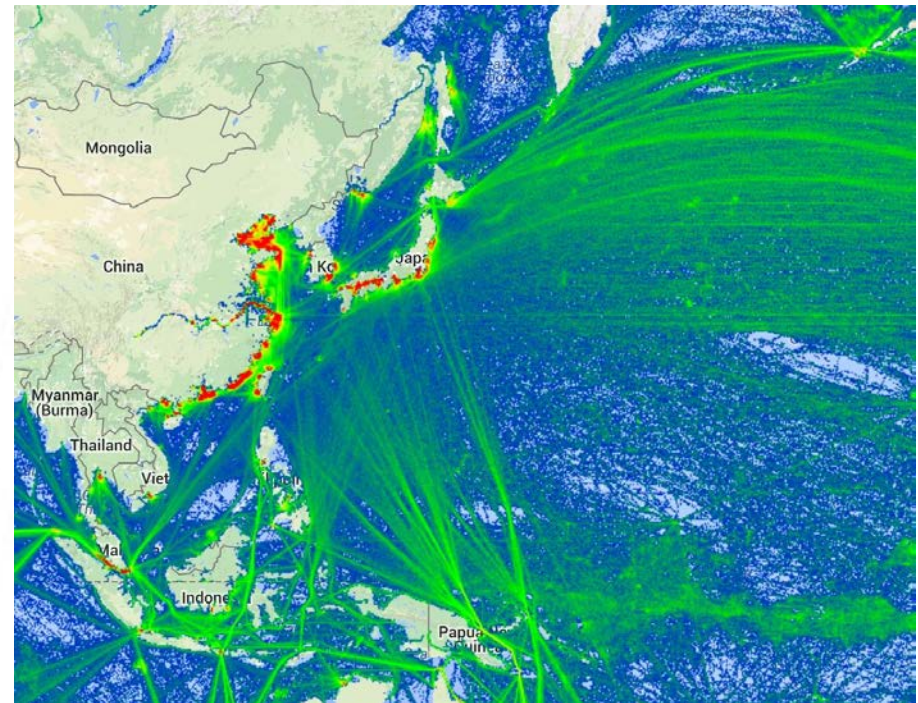
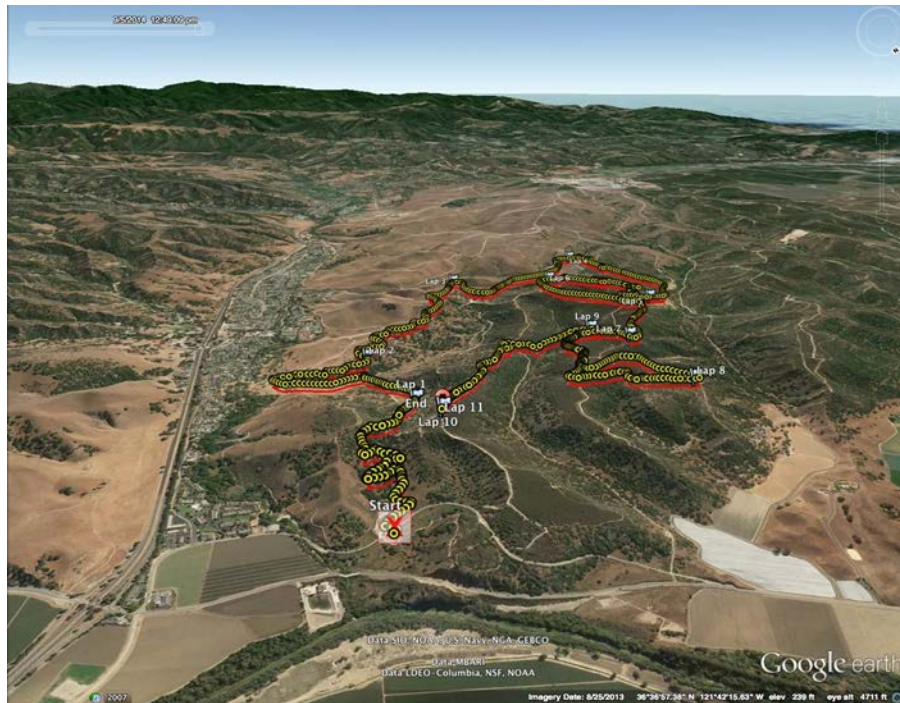
Encodings Compared

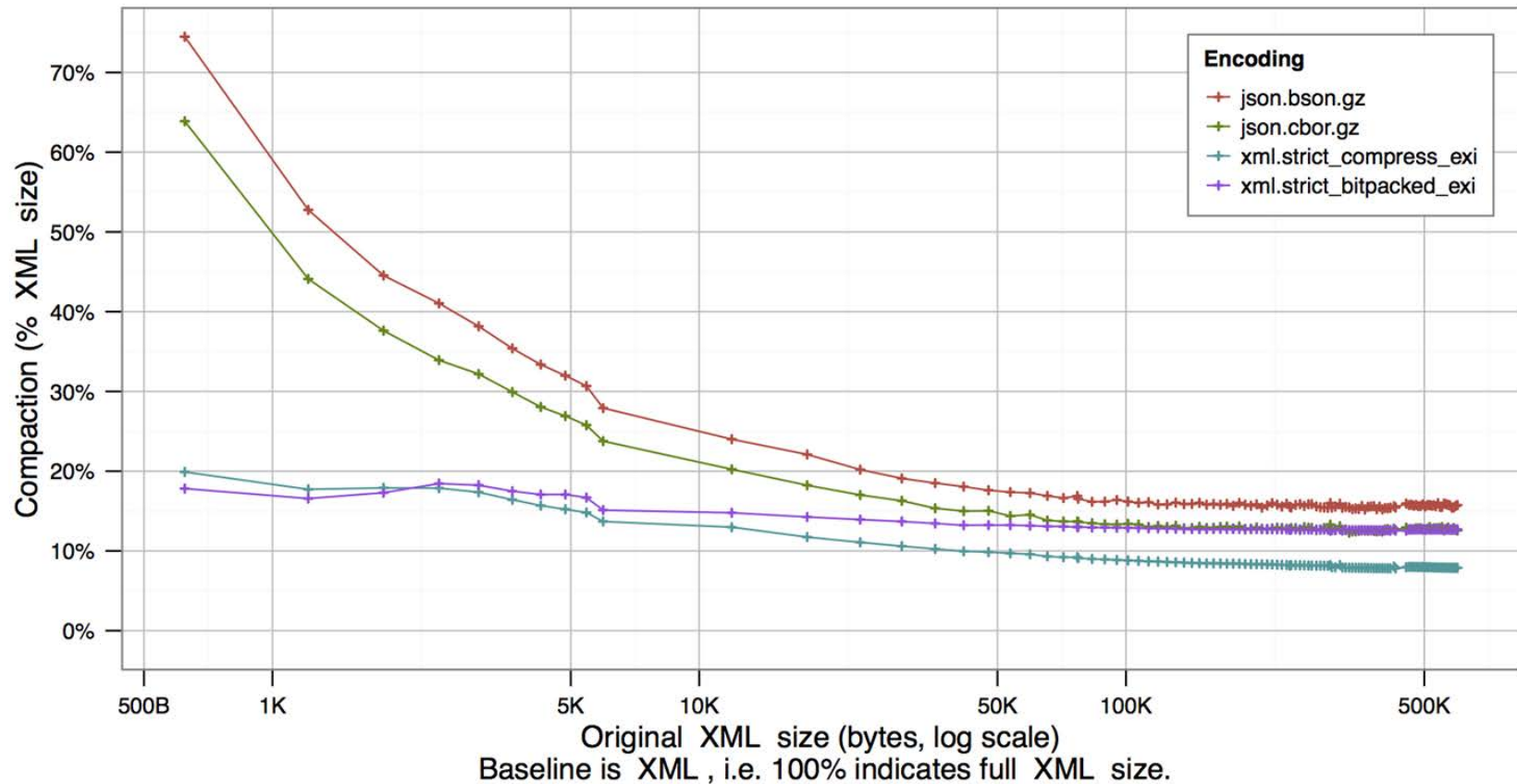


When in doubt, try every possible combination of options

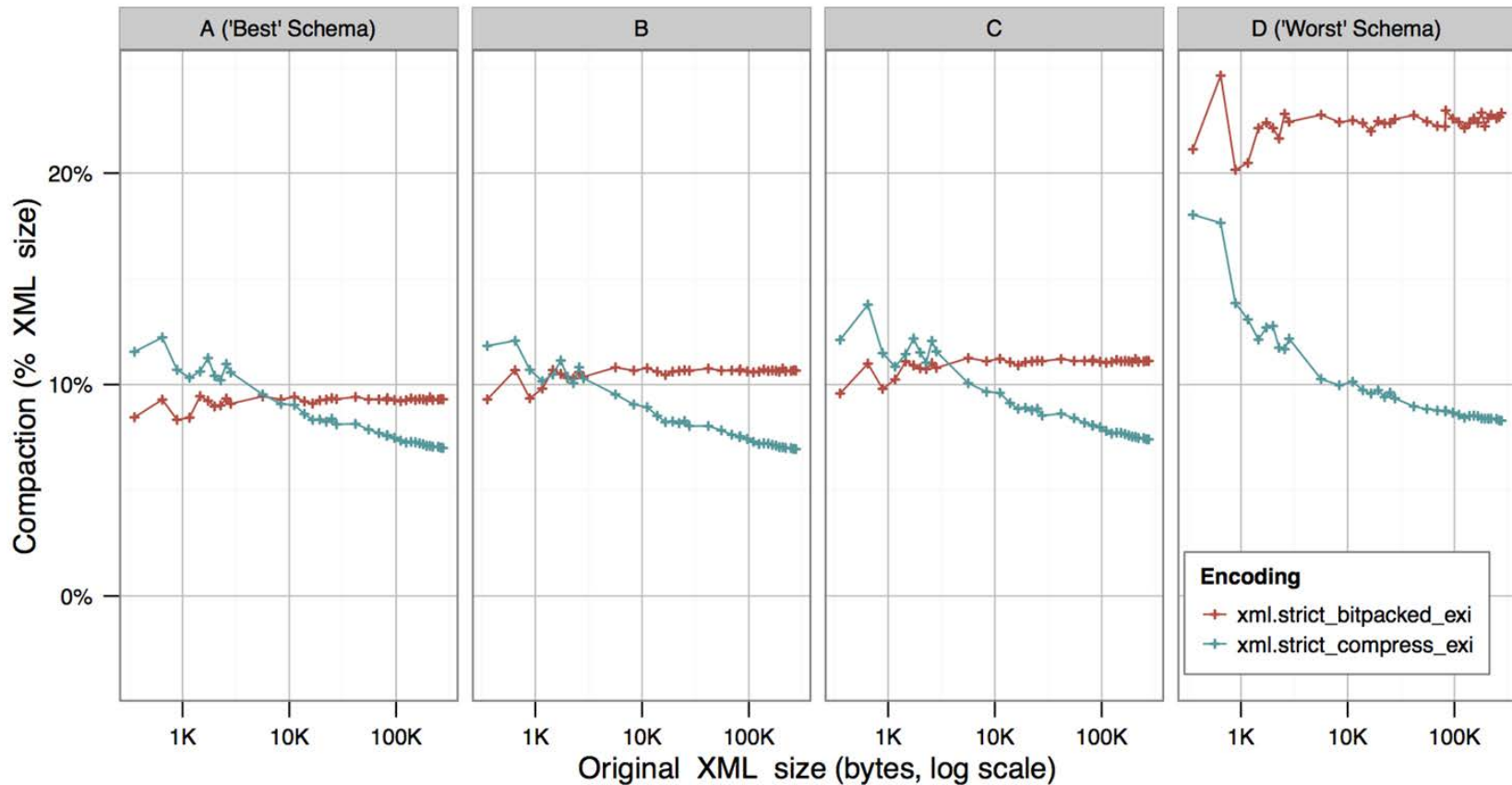
Small-file Use Cases (B to KB)

- OpenWeatherMap
- Global Position System XML (GPX)
- *Automated Identification System (AIS)*





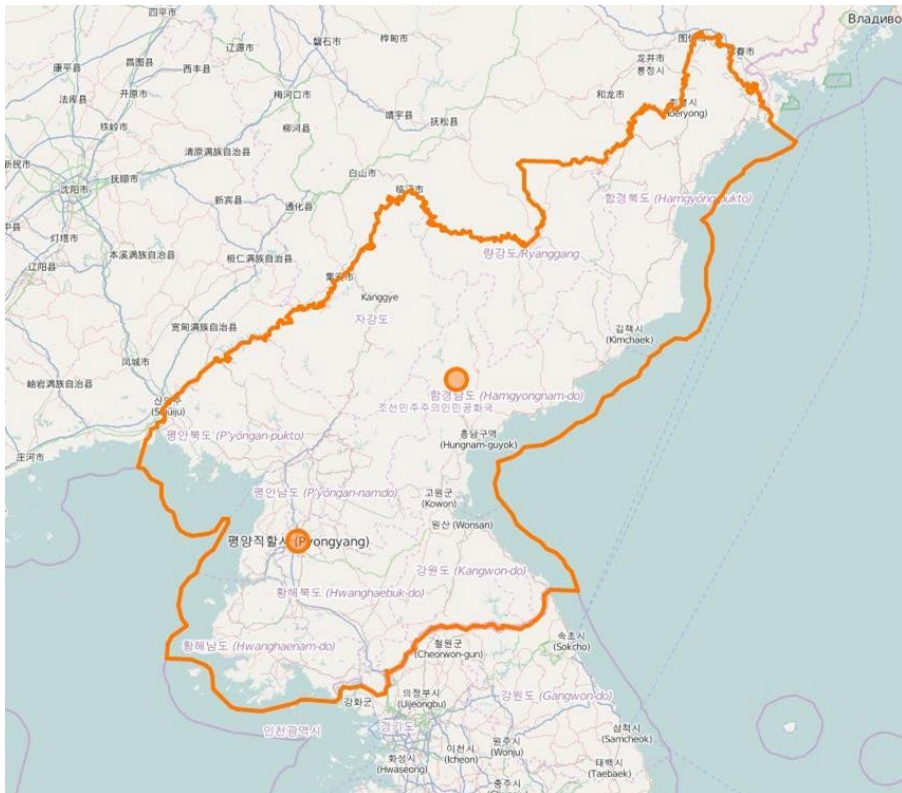
EXI smaller than CBOR/BSON, aggregating data helps

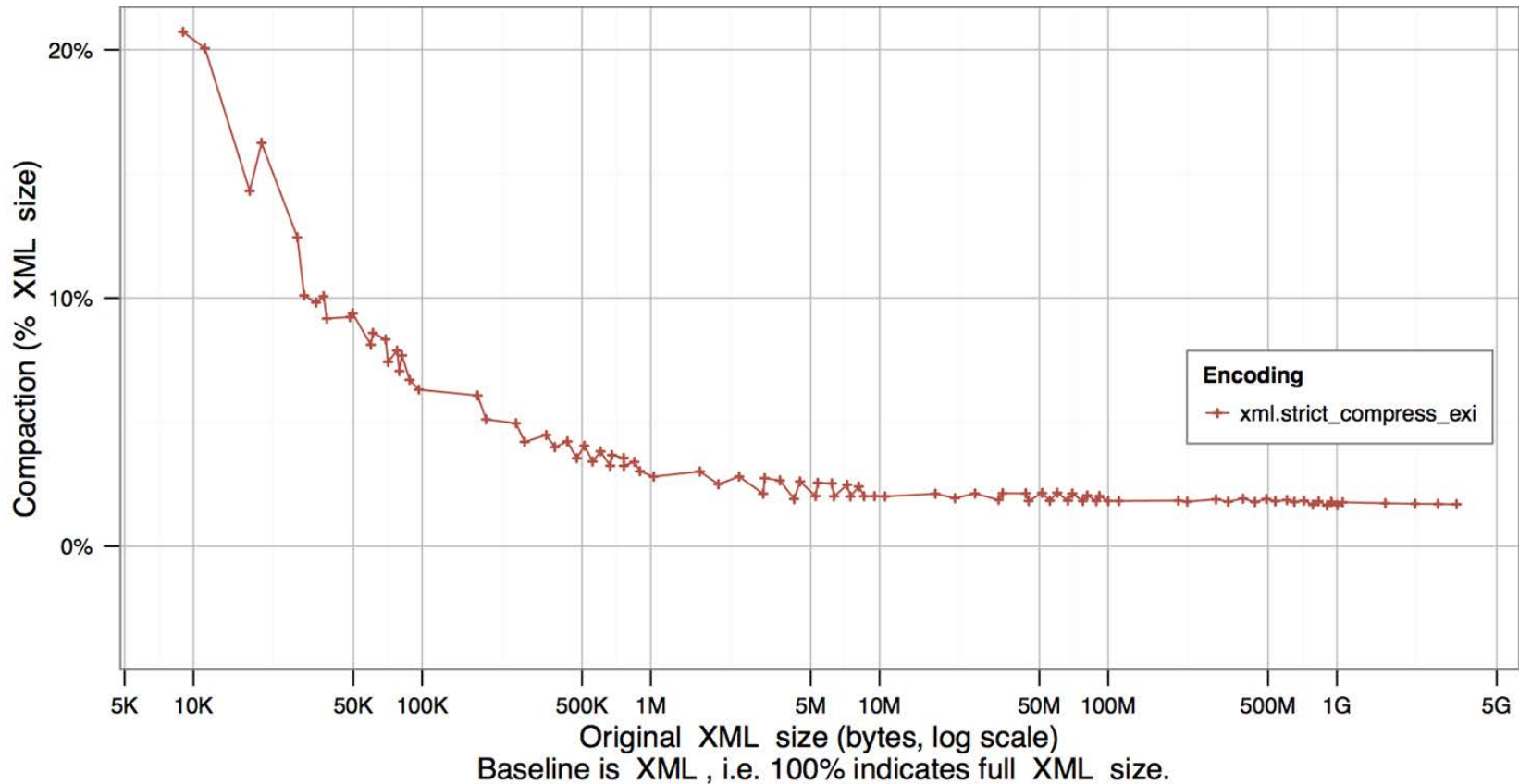


Well-designed XML Schema improves performance

Large-file Use Cases (KB to GB)

- Digital Forensics XML (DFXML)
- OpenStreetMap
- *Packet Description Markup Language (PDML)*



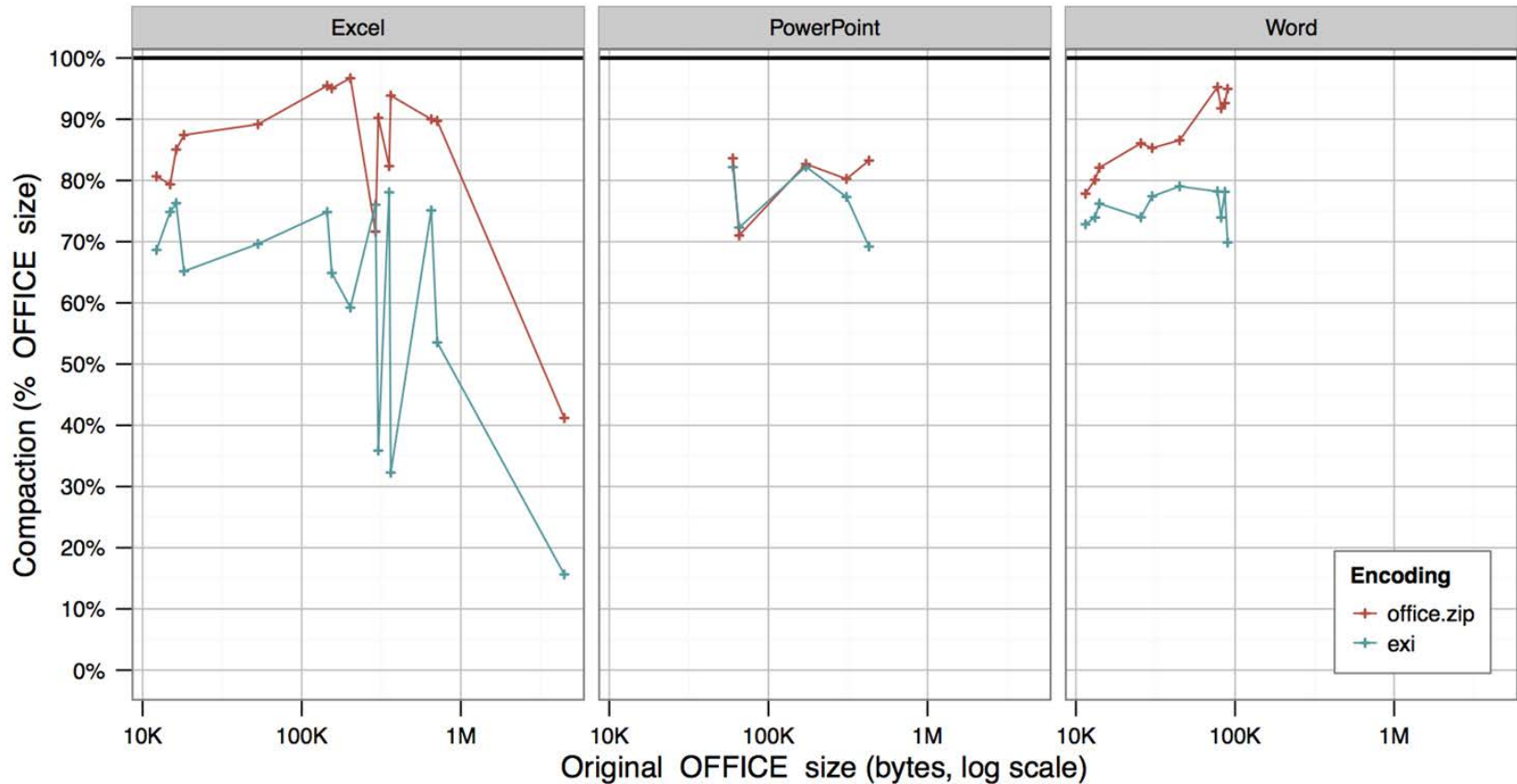


EXI performs well on large files, aggregation benefits plateau

- Microsoft Office is ubiquitous in Navy/DoD
- Since 2003, the file format has been a Zipped archive of many small XML files
- Since 2006, the file format has been an open standard
- Since 2013, MS Office 365 can save in compliant format
- Tools such as NXPowerLite target excess image resolution and metadata to shrink them
- EXI can target the remainder...



Microsoft Office Use Case



(Images removed from all files)

- When to send?
 - Aggregating data improves performance
 - Balance with operational requirements
- EXI configurations are significant
- XML Schema is significant
 - Previously a tool for data *validation*, now a tool for *compression*
- EXI is generally more compact than JSON-based binary encodings
- EXI performs well on large files

Tuning data, XML schema and EXI codec on a per-application basis maximizes benefits

- **Holistic Profiling**
 - Optimizing EXI encodings is a multi-dimensional problem
- **Need for Best Practices**
 - How to make sure we're getting the best performance possible for EXI?
 - Rethinking XML and associated schemas a must
- **Expanding EXI to the Open Web Platform**
 - HTML5, CSS, JavaScript, JSON, SVG are the building blocks of tomorrow's applications, distributed over networks
 - All are targets for EXI-like compression techniques
- **Fleet Adoption**
 - Open source EXI codec on every desktop and server