

Large Instance Processing

Eric JARRY – Banque de France
eric.jarry@banque-france.fr

Agenda

- Why large instances
- Challenges and **ideas**
- Forum / Discussion

Large instances, why?

- Recently, it has been found that some reportings lead to potentially large instances
 - Bank of Indonesia – banking supervision
 - European Banking Authority (EBA) – banking supervision
 - Europea, Insurance and Occupational Pensions Authority (EIOPA) – Insurance supervision
- This volumetry is caused by the requirement to report lists of objects' (loans, financial products, accounts...) that may contain tens of thousand of items
- Example: in Solvency II EIOPA taxonomy, the template S.06.02 (Assets D1) list the contents of assets' portfolios

Loading the taxonomies

- Large instances may correspond to big taxonomies
 - DPM (Data Point Modeling) in taxonomies (so called highly dimensional taxonomies) lead to big taxonomies and big instances
 - Some architectures lead to complex taxonomies
 - The need for multilinguism increase the size of the taxonomies
- Precompiling the taxonomies may be envisioned (i.e. : storing and loading the taxonomy's info set)
- In the case of multi-entry points taxonomies, in memory components may be reuse

Generating the instances

- Typically an XBRL instance has the following structure (recommended by FRIS):
 - Root element (xbrli:xbrl), with start tag containing used namespaces' definitions
 - Contexts, using NS definitions
 - Units (few), using NS definitions
 - Facts, using NS definitions and referencing contexts and units⇒Need to keep elements in memory
- The XII working group note proposes to change the order and to authorize duplicates to allow streaming (for both generation and parsing) with a potential effect of increasing the size

Signing the instances

- Some components used to sign the instances may have size limitations
- Signing a compressed file may be envisioned, but a canonical compression algorithm must be defined

Transmitting the instances

- Software components and infrastructure may have size limitations
- Compression should be considered
 - The compression ratio is typically very high

Parsing the instances

- This aspect is covered by the Working Group Note
- To avoid multiple parsing of the instances:
 - The XBRL validator could give access to its instance info set to the caller application program
 - Plug-in / interceptors may be used to allow specific processing of some elements of the instances

Validating the instances

- XBRL 2.1 and Dimensions validation may be done fact by fact, providing that the associated context and unit are available

Checking business rules

- Now typically done by XBRL assertions
- Most important time in big instances processing
- Optimization of assertions may be used
 - Proposed by software vendors (problem if beneficial for only one or a subset of products)
 - Using filters (done by taxonomy developers)
 - Reusing components (variables, filters, expressions...)
- Other solutions
 - Getting rid of used facts
 - Slicing the instances into reporting units
- Consideration of alternatives to assertions (checks in data bases, spreadsheets...)

Reporting errors

- *See slot on reporting errors*
- Big instances may lead to many errors and big log files
- The processing may stop after a given number of errors
- Mechanism to browse the errors must be given to users

Rendering the instances

- Big instances are typically associated to big open or closed tables
 - Closed tables may be associated with countries, currencies, sometimes combined with another dimension (line of business, period...)
 - Some axes may have unused values (principle of materiality or proportionalities)
 - Some axes may need to be sorted
 - Same fact may correspond to several cells (identification)
- **Sophisticated rendering tools must be implemented**

Forum / Discussion

