INTERNATIONAL ORGANISATION FOR STANDARDISATION
ORGANISATION INTERNATIONALE DE NORMALISATION
ISO/IEC JTC1/SC29/WG11
CODING OF MOVING PICTURES AND AUDIO

ISO/IEC JTC1/SC29/WG11 MPEG2018/M43547
July 2018, Ljubljana, SI

Source: **DMAG-UPC**
Status: **Proposal**
Title: **MPEG-G Part 3: Information compression needs**
Authors: **Daniel Naro, Hao Wu, Jaime Delgado, Silvia Llorente (Distributed Multimedia Applications Group - Universitat Politècnica de Catalunya, Barcelona)**

Table of Contents

1 Problem

The current version of ISO/IEC 23092-3 does not address the issue of compression for information metadata. This document analyses the current size of the information metadata stored either in the `dgmd` and `dtmd`, or in the `auin` gen_info structures, as defined in ISO/IEC 23092-1.

2 Size of information stored in `dgmd` and `dtmd`

The gen_info structures `dgmd` and `dtmd` are defined to store metadata relevant for the entire dataset group and dataset respectively. Additionally, if some values are shared among datasets, these values can be stored within the dataset group element: ISO/IEC 23092-3 defines a mechanism by which values stored within a `dgmd` element are inherited at the `dtmd` level.

The information is stored in XML format, following a schema defined in ISO/IEC 23092-3. In order to not be limited by the fields selected to be present in the schema, there is an extension mechanism which allows to extend the scope covered by the metadata. At the moment, only extensions to cover the needs of the EGA genomic repository, and to enable a round-trip from SAM to MPEG-G are defined.

2.1 EGA metadata case

The first part of this analysis is based on real metadata used in the scope of EGA. The only difference between the metadata we used and the version stored at EGA is that the name of the samples, and the name of the hospital were anonymized.

The metadata describes one study with four samples. In the mapping between EGA and MPEG-G, as explained in ISO/IEC 23092-3, this translates as one dataset group, containing four datasets. Each of these five elements has a metadata structure (`dgmd` for the dataset group, `dtmd` for the four datasets).

The original uncompressed XML content represented according to EGA's schemas, was 14991 bytes long. The same information represented in MPEG-G's schemas (see annexes at ISO/IEC 23092-3) has a size of 20305 bytes. The overhead is caused by the need to introduce extensions.

2.2 SAM headers case

Clause 8 of ISO/IEC 23092-3 specifies a direct way of keeping metadata of a SAM header. These elements should be also included in the `dtmd` element.

We have developed a XML Schema of the SAM header and generated a specific instance with the header elements of a real SAM file. In this case, the size of the XML document is 17038 bytes.

3 Size of information stored in `auin`

In the SAM format, or similar, each record is possibly accompanied by a set of auxiliary tags. Certain tags are already covered by MPEG-G data representation. Others are not covered by the information stored in the descriptor streams. In order to store these, ISO/IEC 23092-3 defines a data representation to store this content in the `auin` gen_info structure.

In order to approximate the size of this information, we wrote a script iterating over each read in a genomic SAM file, and computing the size of the information to be stored in the `auin` elements. We should note that in clause 8 of ISO/IEC 23092-3 there is no indication on how data are represented when the tags of the auxiliary fields are user-defined. To face this challenge, we decided to compute it as if the selected type was a sequence of characters. This decision could be debated, but only affects the size reported for the elements starting with either a character X, Y, or Z.

In case of the file known as *input 05*, a file originally 6.1GB long, the information to be stored in the `auin` element is approximated to 7 GB 239MB 65kB 786 B. In order to better understand the causes of this huge size gap, we generate Figure 1. In order to generate the chart, we have divided the size required per tag in segment identifier and the actual size of the data, which will depend on the tag nature. This helps us highlight the huge cost of the current solution to identify each record.

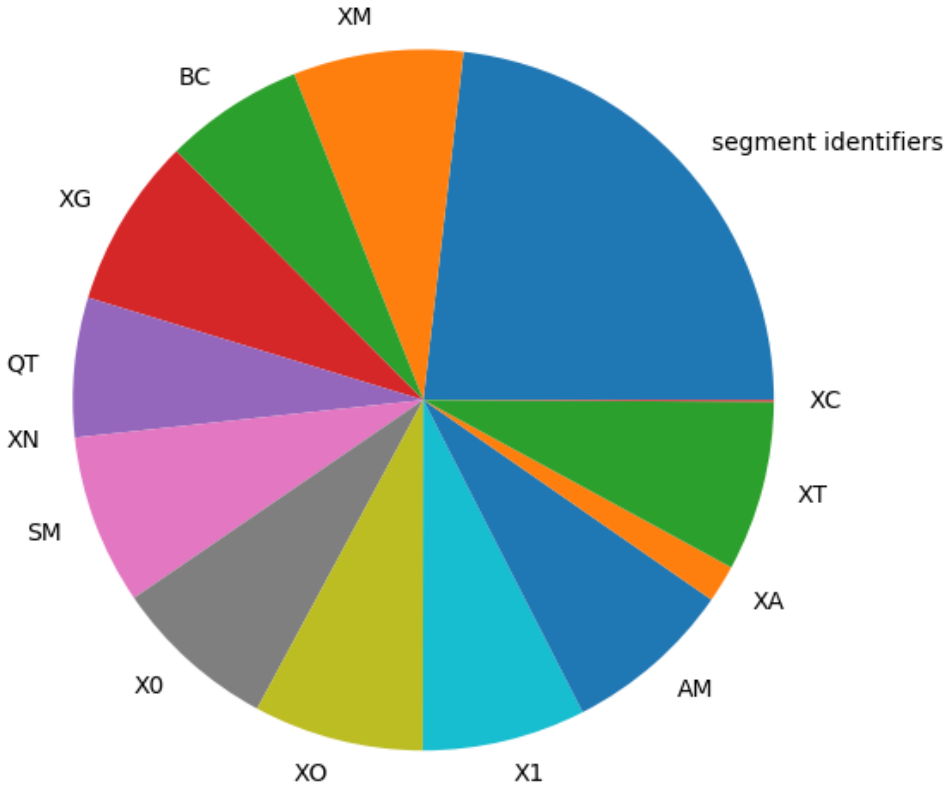

Figure 1: Ratio of usage for each type of data to be stored in `auin` input 05

In case of the file *input 02*, a BAM file originally over 100GB long, the estimated size for the auxiliary tags information is 116GB 699MB 784KB 837B.

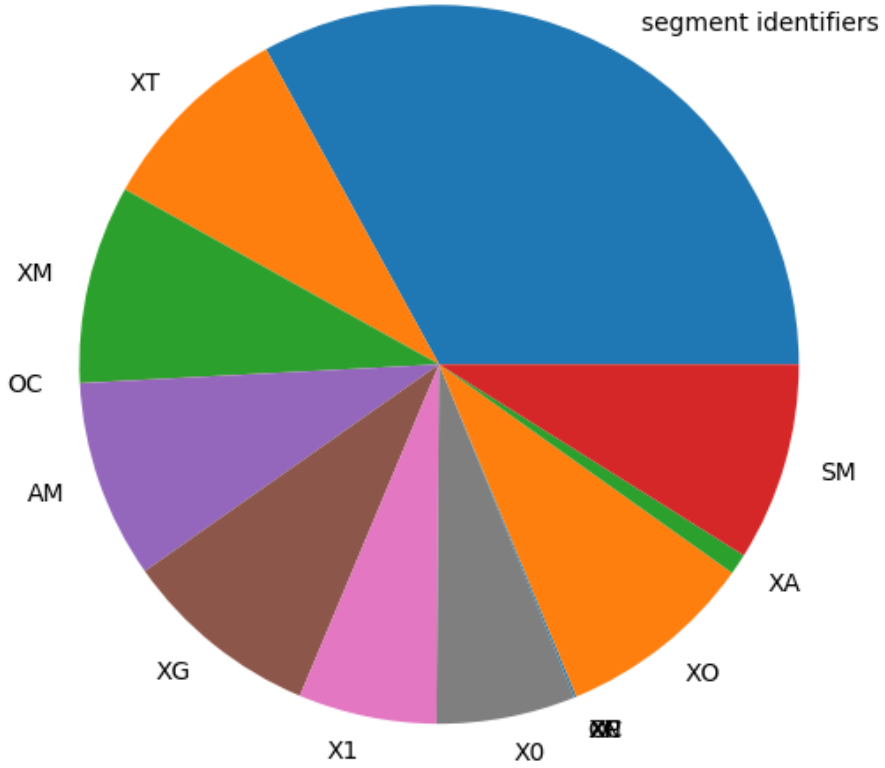

Figure 2: Ratio of usage for each type of data to be stored in auin input 02

4 Compression of the information

Based on the previous analysis, we need mechanisms to compress:

- The metadata in `dgmd` and `dtmd`.
- The metadata in `dtmd` coming from SAM, if any.
- The SAM auxiliary fields introduced in `auin`, if any.

While the kind of information in the first two cases is the same (XML data). The last case is clearly different and (very) much higher in size.

Therefore, two different approaches are needed for both cases. In addition, given the sizes of information, it is not strictly necessary to compress the XML information, since it could be smaller than the `auin` information by a factor of 10^7.

Since the kind of information included in the auxiliary fields could be similar to that in the descriptors specified in ISO/IEC 23092-2, similar approaches could be considered.

On the other hand, since MPEG has already standardized a mechanism to compress, binarize and serialize XML data (BiM, Binary MPEG format for XML, ISO/IEC 23001-1), this could be used, if required, to compress the `dgmd` and `dtmd` elements.