Easy and Cheap Compression with EPVzip V15

Fabio Massimo Ottaviani

May 2021

1 Introduction

The policy of EPV products has always been to provide all the needed information for z/OS performance analysis and capacity planning, by running on an external platform such as Windows, Linux or Unix.

The reason for this choice is our continuous effort to reduce customer hardware and software costs.

The main source of information for all the capacity management activities are:

• SMF records, which are in VBS format,

• IMS log records and DCOLLECT records, which are in VB format.

When transferring this data to an external platform there are two important issues to consider:

• the amount of data to be transferred is normally very large,

• you need to preserve data integrity, control blocks included, in order to be

able to correctly parse them.

Both issues can be addressed with compression. This is the reason why, some years ago, we developed the EPVzip tool which, running on z/OS, can compress VBS and VB data before transferring them to the processing platform.

Of course, we wanted to limit as much as possible the impact on z/OS costs, so we developed it in Java to allow it running on zIIP instead of CPU.

The result is very good: at least 90% of the EPVzip consumptions are on zIIP and the compression ratio is normally around 10.

Even if SMF data are already compressed, as it could be the case for the SMF records produced by CICS and Db2, EPVzip provides a meaningful reduction of the data size.

It is important to note that EPVzip uses the zlib compression library, so, when running on z15 machines, it automatically exploits the Integrated Accelerator for zEDC.

In this paper we will show what you need to do to use EPVzip.

We will also discuss EPVunzip: an additional tool which allows users to decompress, on z/OS, any VBS and VB file, previously compressed using EPVzip.