EPSC Abstracts
Vol. 6, EPSC-DPS2011-1040-1, 2011
EPSC-DPS Joint Meeting 2011


# A validator for SASF (Spacecraft Activity Sequence File) NASA language interface in DAWN VIR experiment

**S. Fonte** (1), F. Carraro (1), A. Coradini (1), M. C. De Sanctis (2), E. Ammannito (1).

(1) INAF-IFSI, Rome, Italy, (2) INAF-IASF, Rome, Italy

## Abstract

Dawn is a NASA mission for Solar System exploration; there are there instruments involved, one is a Visible and Infrared spectrometer (an Italian industrial development). The management of each instruments comes using the SASF (Spacecraft Activity Sequence File) interface: an ASCII timing script containing the instruments commands and setting parameter. In order to avoid problems during the management of SASF syntax, a syntax analyzer for this language has been developed: VIRV (VIR Validator). The VIRV purpose is to help in checking the SASF syntax and to produce a simulation of the instrument operative sequence that will be executed in the VIR flight machine.

The development guidelines of VIRV are the idea of usual compiler, as C/C++ or fortran compilers, where a language (as close as possible to human language) is translated into machine language (where each byte can be understood by the microchip which will execute it. It is the so called *machine byte code*) during the compilation/linking phase.

VIRV was developed to behave as a C/C++ compiler: there's no byte code but a table with a simulation of VIR instrument operation during a SASF procedure.

VIRV is composed by two macro systems: the Lexer and the Parser. The first one performs an analysis of the input ASCII file and tokenize this input in tokens (an ASCII character sequence with a SASF meaning). The second one (Parser) catches these tokens and produces a language structure analyzer. At the end of this process, if this analysis produces a correct result, a transformation of the sequence into a table with relevant quantities for the VIR simulation is performed.

The principal token type are Time, Identifier and Text token:

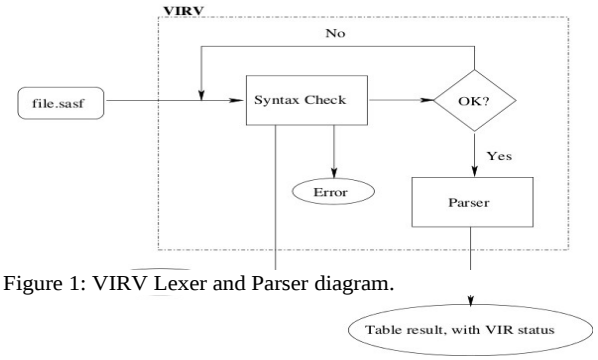


Figure 1: VIRV Lexer and Parser diagram.

1. **Time** identifies the time formats in the sequence,
2. **Identifier** refers to a SASF directive and VIR commands;
3. **Text** tokenizes all string comments between quotes.

The Tokens are organized in a language grammar that respect SASF interface. The combination of these structures product the sequence.

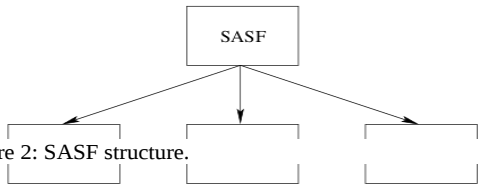


Figure 2: SASF structure.

The VIRV Parser is designed to check the presence of a violation of the VIR flight rules within the analyzed sequence. It makes a test of the instruments constraints on the data volume and checks if VIR command parameters values are contained within their boundaries.

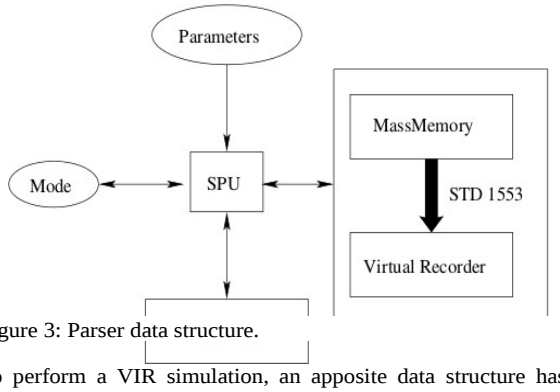


Figure 3: Parser data structure.

To perform a VIR simulation, an apposite data structure has been created within the Parser: a SASF Process Unit, the Memories Unit and the Status parameters.

The goal of these data structures is to perform a simulation of the VIR instrument flight status diagram. The validation process can be divided into two steps: at the very beginning of the sequence development the user checks if the syntax is correct. Then the user removes all typo issues and/or present violation of flight rules. The second step, at the end of the previous validation, the user checks the diagram produced to avoid undesirable behavior on the sequence.

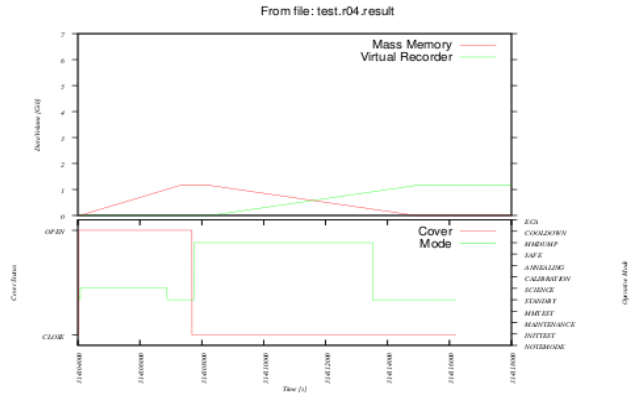


Figure 4: Status diagram of a test sequence simulation.

The previous picture represent a result of a simulation on test sequence which represents a typical acquisition during the VIR operations. This tools is constantly used during the usual VIR flight sequences production and helps the VIR Team into generating the SASF for Vesta acquisitions.

## References

[1] M.C. De Sanctis, et al., The VIR Spectrometer, Space Sci Rev DOI 10.1007/s11214-010-9668-5, 2010.
[2] A. Coradini, et al., Virtis: An Imaging Spectrometer for the Rosetta Mission, Space Science Reviews, Volume 128, Issue 1-4, pp. 529-559, 2007.
[3] John R. Levine, Flex & Bison, O'Reilly, ISBN: 978-0-596-15597-1