A STANDARDIZED INTERFACE BETWEEN ORBITAL DEBRIS ENVIRONMENT MODELS AND DAMAGE PREDICTION TOOLS

D. Nölke[1] and H.-G. Reimerdes[2]

Department of Aerospace and Lightweight Structures, RWTH Aachen University,
Wuellnerstr. 7, 52062 Aachen, Germany
E-mail: [1] noelke@ilb.rwth-aachen.de, [2] hg-reim@ilb.rwth-aachen.de

ABSTRACT

Nowadays several environment models exist which describe the historic, present and future orbital debris population in Low Earth Orbits (LEO). The models contain information like flux, impact velocity and impact direction of particles encountering the spacecraft in its orbit. This information differs not only in the values between the models but also in the structure and the content of the result files. A fact, which makes it complicated for users like developers of damage prediction tools, to access the relevant data from environment models.

Therefore, it is suggested to establish a standardized interface between orbital debris environment models and damage prediction tools. A main advantage is that different environment models can be easily adapted to the damage prediction tools without high effort. The interface, which is proposed in this paper, supports all relevant data to perform a damage prediction for a spacecraft in a meteoroid and orbital debris environment.

Key words: interface; standard; environment; model.

1. INTRODUCTION

Current environment models like ORDEM2000 or MASTER2001 describe the orbital debris environment considering measurement data (e. g. from Haystack, Goldstone Liou (2002)) and statistical approaches for particles with diameters typically below 10 cm and down to the 10 μm or 1 μm population.

The different statistical approaches and the various measurement data considered as well as the numerical description of the environment lead to diverse results for a mission. Not only the values differ but also the structure and the content of the result files. This makes it complicated for the users, e. g. the developers of damage prediction tools, to access the required data from current environment models. Looking back on early environment models like NASA91, the description of the orbital debris environment was easy to understand and one was able to implement the environment model into the damage prediction tool. The regularly updates of those models become more and more complex. Current models are standalone programs which are not platform independent anymore. To get the required data for an damage prediction analysis, an interface for each model has to be developed. Possibly the damage prediction analysis approach in the tool itself has to be changed, e. g. due to more complex distributions obtained from the environment model.

Future development will lead to additional, more complex interfaces for each new model. Every developer of a damage prediction tool has to develop an adequate interface by himself. Varieties between the damage predictions will increase. To avoid such a development, it makes sense to define a standard which describes the structure and content of the environment model output that should be used with damage prediction tools.

2. REQUIREMENTS FOR A STANDARDIZED INTERFACE

The definition of the requirements for a standardized interface can be done by answering the question what is needed to perform a damage prediction analysis. This should be done by developers of damage prediction tools. At the moment there are different tools like BUMPER, ESABASE/DEBRIS or MDPANTO available IADC (2004). It is most possible that these programs are using the output of environment models in different ways. Even the kind of utilized output can vary (e. g. different distributions).

Thus to define a standard, it is essential to define the content of data and the way how the data is provided. The content of data can be divided into *required data, addi-*

Proceedings of the Fourth European Conference on Space Debris, Darmstadt, Germany, 18-20 April 2005
(ESA SP-587, August 2005)

tional data and *user-specified data*. The definition of all three data groups depends on actual needs and offered data. Therefor the developers of environment model software and damage prediction tools have to discuss them to clarify if all the requirements can be fulfilled.

The *required data* for a damage prediction analysis mentioned above could be the following:

- Flux versus Diameter/Mass distribution
- Density distribution
- Directional distribution
- Velocity distribution
- Number and limits of bins for the distributions
- Spacecraft geometry and attitude, wall configuration (single wall, double wall, material, ...)
- Mission duration (start and end of mission)
- Ballistic Limit Equations to compute the penetrations

The data of the first five points can be easily provided by the environment models. The *number and limits of bins* are required to simplify the variable declaration in the damage prediction program and the data management. The information about *spacecraft geometry, wall configuration and attitude* is usually defined by other input files and is not provided by the environment model. It is necessary for a damage prediction analysis to know how long the spacecraft is staying in its orbit to determine the number of impacts. Therefor the *mission duration* is required. In case that different spacecraft attitudes are considered, it could be necessary to define orbit intervals (refer to section 3) which would belong to the *required data*, too.

Further needs for the analysis, the *additional data*, are not required for the damage prediction analysis but more for the identification of the mission. These needs are *mission time*, *orbital elements* and *environment model identification*. In the case that the environment model is adapted directly to the damage prediction tool and the environment data is calculated on request during the damage prediction analysis, the *additional data* belongs to the *required data*.

The *user-specified data* are all information that would be nice to have to make the damage prediction analysis more transparent to non-experienced users. This data is coming by users and developers of damage prediction tools from time to time and cannot be listed yet.

In contrast to the content of data, the way the data is provided is mainly determined by the environment model software, the damage prediction tool and the platforms that both programs use. Generally there exist two possibilities to transfer the data from the environment model to the damage prediction tool. The first one is, to adapt the environment model to the damage prediction tool directly (or implement it into the source code) like it was done with older models (NASA91 and NASA96). But due to the fact that actual models are too complex, it is not feasible anymore to integrate them into the source code without detailed documentation and the source code of both programs. If the source code is available and/or the environment model and the damage prediction tool support the same platform, it makes sense to choose the direct adaption for the data transfer. The second way to transfer the data is to write it into a file, which can be read by other programs. This method is chosen for this proposal because it is easy to export the relevant data from the environment models and to read it again for the damage prediction analysis. Furthermore it is platform independent.

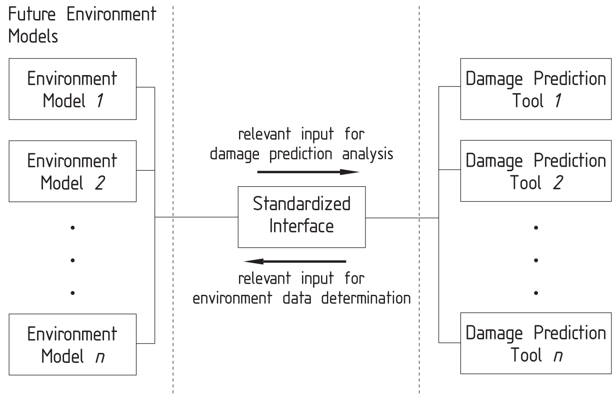

Figure 1. Scheme of the link between environment model, standardized interface and damage prediction tools.

For the standardized interface it is necessary that it works with all related programs. This leads to the following additional requirements for the interface:

- The data transfer should be done by an interface file if the environment model cannot or shall not be adopted to the damage prediction tool.
- The interface file should be in ASCII format to be platform independent and readable.

In general, the interface should be defined such, that a maximum of flexibility for the adaptation to different future environment models and an easy integration to current damage prediction tools is guaranteed. Figure 2 shows schematically the link between future environment models, standardized interface and damage prediction tools.

Such an interface would make it easier to adapt future environment models to damage prediction tools and an interface file containing the relevant data could be simply generated by the environment software. It is just a matter of unifying the format of the output.

3. FIRST APPROACH OF THE STANDARDIZED INTERFACE

Current environment models and damage prediction tools should be able to perform analyses for nearly all space missions, at least in LEO due to the huge amount of orbital debris in this region. Not only debris fluxes have to be considered, but also sporadic fluxes (e. g. from meteoroids) and stream fluxes (from meteoroid streams). New environment models like MASTER2001 consider those fluxes. The user has just to define the orbit parameters of the target (spacecraft) and the environment model computes the respective distributions. Thus an interface should be able to support the different kinds of distributions and target orbits to transfer the data to the damage prediction tool. The following list gives the basics that should be supported:

- Debris/Meteoroid sources
  - Average flux (Debris)
  - Sporadic/Stream flux (Meteoroid)
- Missions in LEO/GEO with
  - Circular orbits
  - Elliptical orbits
  - Transfer orbits

The first approach in the development of the standardized interface considers only *average flux* of debris and *circular target orbits*. Ideas considering *elliptical target orbits* have already been gathered and will be adressed below in this paper. *Transfer orbits* are not yet assessed. Furthermore the sporadic and stream fluxes are not discussed in this approach.

Short investigations with the ORDEM2000 model have shown, that the fluxes vary with the spacecraft position on its orbit for circular and elliptical orbits. If focused on the circular orbits it can be said that the number of impacts on a spacecraft that performs lots of orbits without changing its attitude can be calculated well with an average flux. In contrast to this, the operation of a spacecraft that changes its attitude at certain orbit positions, could be affected by impacts on less protected components. Thus there should be a possibility that the distributions are either provided as an average flux of one orbit interval or for a defined number of orbit intervals. The selection can be upon the user and depends on the mission and analyzation speed that is preferred. The definition of an orbit interval that is assumed in this approach can be taken from figure 2.

As it was mentioned above, an ASCII file is proposed to transfer the relevant data between the environment model and the damage prediction tool. This file is the output of the interface which is adapted to the environment model software. Assessing the requirements for the standardized interface leads to an overview of the content of the interface file which is shown in figure 3.

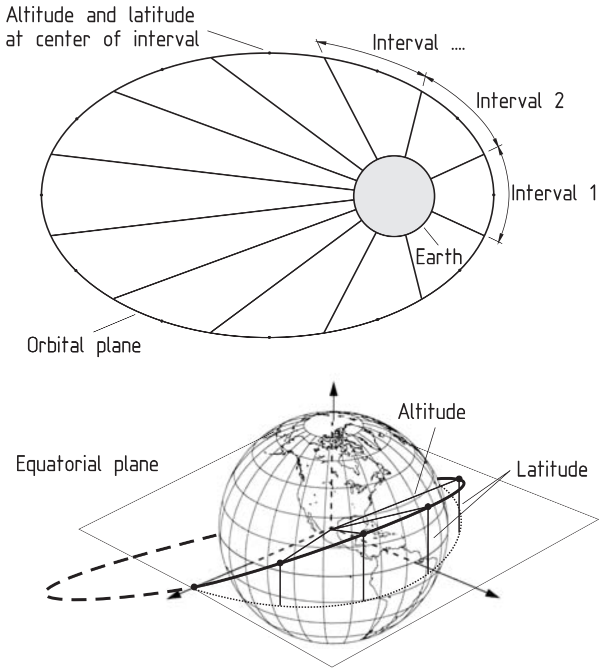

Figure 2. Definition of the orbit intervals.

4. DESCRIPTION OF THE CONTENT OF THE STANDARDIZED INTERFACE FILE

After describing the requirements on the interface, the content of such a file will be explained in this section. The interface file should contain all the data mentioned in the sections above. First the way how the content is written within the file has to be fixed. Further in the text this is called structure of the interface file. The structure of the standardized interface file could be divided into two main sections:

- Header
  The header contains data for the identification of the interface file, of the environment model (including comments) and of the mission.
- Section of distributions
  This section contains data for the distributions of flux, density, direction and velocity as well as the number and limits of bins (direction and velocity) and the number of orbit intervals.

The arrangement of these sections are of main influence on the transparency (the understanding of the content) of the file. There are lots of possibilities for the arrangement, one of them seems to be convenient and is presented in figure 4.

The flux/density distribution and the directional distribution are separated in this possibility. Furthermore the diameters are listed before all distributions. Because of this only the respective number of the diameter is written in

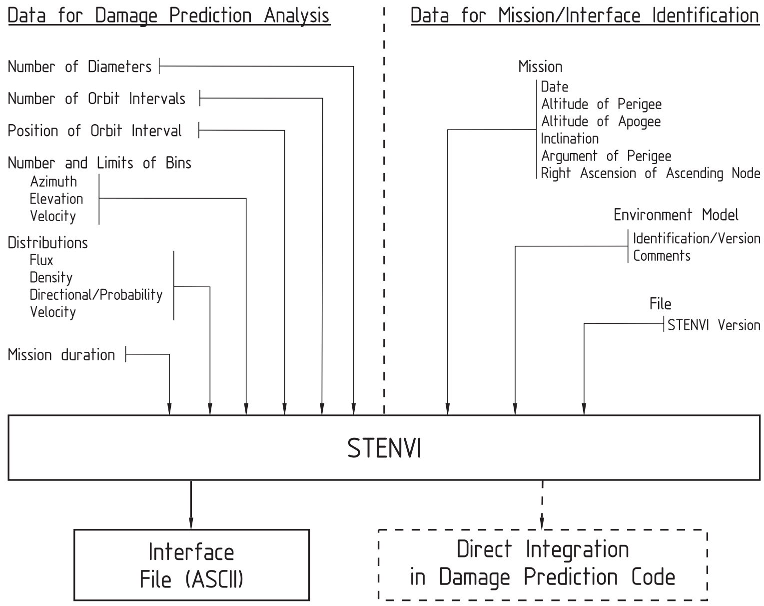

Figure 3. Possible content of the standardized interface file.

the data line `DISTHD` to assign it to the flux and density. The directional distribution is introduced by the *distribution head* (definition of number and limits of azimuth, elevation, ...). To assign the directional distribution to the correct interval and diameter, the data line `DISTSET` contains the number of the intervals and diameters as well.

If the future development leads to data, that has to be added somewhere in the sections, it is no problem to integrate the data. Defining the data lines with *input cards* (names at the beginning of a data line, e. g. `AZIMUTH` or `MISSION`) makes it possible that the order of reading the data doesn't matter.

In parallel to the arrangement of these sections like *Header* or *Directional Distribution* there are three possibilities to comment the content of the file:

- sufficiently commented
  Data is marked with input cards what gives the possibility to change the order of data within the file. This makes the file better to understand for users and developers but adds some more lines.

- not commented
  Position of data is necessary to identify the data itself. It is not possible to change the position. Reduces number of lines but is absolutely not understandable without a detailed documentation.

- combination of both
  A combination of both provides enough comments to understand the structure and fixed positions of the data tends to a well ordered file structure.

For the first approach it makes sense to use the third way of commenting the interface file. The description of the content in detail will be done in the following paragraphs for the example of an interface file with the arrangement as it was discussed above (refer to figure 4).

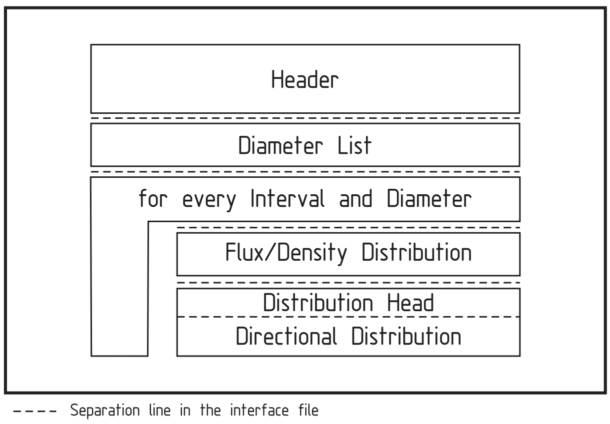

Figure 4. A possibility for the arrangement of the file sections.

The interface file starts at line 4, the characters `#EOF` marks the end of file. The header consists of four inputs:

- Line 7: `STENVI-1.0`
  This is the identification of the interface file. The first 6 characters build the name, characters 8-10 describe the interface version. Future development in environment models will lead to upgrades of this interface. A damage prediction tool should be able to read new and old versions of the interface (downward compatibility).

- Line 9: Input Card - `MODNAME`
  This card contains the name of the environment model (6 characters) and the version of the model if there exist different versions of this model.

- Line 12-15: Input Card - `COMMENT`
  These four lines give the possibility to transfer comments from the environment model to the damage prediction tool. Those comments can be then part of the analysis output.

- Line 19: Input Card - `MISSION`
  This card identifies the mission.
  `YEAR` - Year of the mission (or start/end of mission)
  `PERI` - Altitude of the perigee [km]
  `APO` - Altitude of the apogee [km]
  `INC` - Inclination [deg]
  `NINT` - Number of orbit intervals

The section of distributions is divided into the parts diameter, flux/density distribution and directional distribution. The diameter subsection has one input card:

- Line 25: Input Card - `DIAMET`
  This card assigns a number to a diameter. The number of cards is equal to the number of diameters required, i. e. for 51 diameters, 51 cards are needed.
  `NO` - Number of the diameter
  `DIA` - Diameter [cm]

The flux and density distributions are written into the input card DISTHD:

- Line 33: Input Card - `DISTHD`
  This card assigns the flux and density to a diameter and an interval. The length of this section strongly depends on the number of diameters and intervals, i.e. 12 orbit intervals and 51 diameters result in 612 lines. If the density changes with diameter, more lines have to be added.
  `INT` - Number of actual interval
  `NDIA` - Number of actual diameter
  `FLUX` - Average cross-sectional flux for this interval [$1/(m^2\,yr)$]
  `DENS` - Particle density [g/cm^3]

The directional distribution begins with a definition of the number and limits of azimuth, elevation and velocitiy bins in the distribution head:

- Line 41-43: Input Card - `AZIMUTH`, `ELEVATI`, `VELOCIT`
  These cards define the number of azimuth, elevation and velocity bins and their minimum and maximum values.
  `BINS` - Number of bins
  `MIN` - Minimum value
  `MAX` - Maximum value

```
1    DATA ARRANGEMENT EXAMPLE
3    123456789-123456789-123456789-123456789-123456789-123
4    #----------------------------------------------------
5    #-------STANDARDIZED-ENVIRONMENT-INTERFACE-----------
6    #----------------------------------------------------
7    STENVI-1.0
8    #----------------------------------------------------
9    MODNAME   NASA2K    1.0
10   #----------------------------------------------------
11   #-COMMENTS-------------------------------------------
12   COMMENT   LINE1 40 CHARACTERS
13   COMMENT   LINE2 40 CHARACTERS
14   COMMENT   LINE3 40 CHARACTERS
15   COMMENT   LINE4 40 CHARACTERS
16   #----------------------------------------------------
17   #-MISSION-PARAMETERS---------------------------------
18   #          YEAR      PERI       APO       INC   NINT
19   MISSION    2004     400.0     400.0      51.6     12
20   #----------------------------------------------------
21   #----------------------------------------------------
22   #-DIAMETER-------------------------------------------
23   #----------------------------------------------------
24   #            NO        DIA
25   DIAMET        1 1.0000E-03
26   DIAMET        2 1.0000E-02
27   DIAMET        3 1.0000E-01
28      :          :       :
29   #----------------------------------------------------
30   #-FLUX/DENSITY-DISTRIBUTION--------------------------
31   #----------------------------------------------------
32   #           INT      NDIA          FLUX      DENS
33   DISTHD        1         1    3.1234E+03 2.800E+00
34   DISTHD        2         1    2.1234E+03 2.800E+00
35   DISTHD        3         1    3.5234E+03 2.800E+00
36      :          :         :          :        :
37   #----------------------------------------------------
38   #-DIRECTIONAL-DISTRIBUTION---------------------------
39   #----------------------------------------------------
40   #          BINS       MIN       MAX
41   AZIMUTH      36    -175.0     175.0
42   ELEVATI      18     -85.0      85.0
43   VELOCIT      23       0.5      23.5
44   #----------------------------------------------------
45   #           NO    INT NDIA  AZI  ELE  VEL        PROB
46   DISTSET      1     1    1    1    2    2  5.1234E-07
47   DISTSET      2     1    1    1    1    1  2.1234E-06
48   DISTSET      3     1    1   10   15    5  6.1234E-08
49   DISTSET      4     1    1    1    1    3  3.1234E-07
50      :         :     :    :    :    :    :       :
51      :         :     :    :    :    :    :       :
52   #EOF-------------------------------------------------
```

Figure 5. Example of a Standardized Interface File.

The probability or directional distribution starts after the distribution head:

- Line 46: Input Card - `DISTSET`
  This card assigns a probability to certain velocity, direction, diameter and interval. This subsection is the longest part of the interface file, i. e. 12 intervals, 51 diameters, 36 azimuth bins, 18 elevation bins and 23 velocity bins result in 9,121,248 lines. But there are many possibilities to reduce this number significantly (see below).
  `NO` - Number of directional distribution line
  `INT` - Number of actual interval
  `NDIA` - Number of actual diameter
  `AZI` - Number of actual azimuth bin
  `ELE` - Number of actual elevation bin
  `VEL` - Number of actual velocity bin
  `PROB` - Probability

The kind of writing the data into the interface file in the way that is described above leads to long files but all relevant data is included. The length is mainly determined by the number of bins or in other words the complexity of the environment model. Furthermore it depends on the desired degree of simplification of the damage prediction analysis. Some possibilities of simplifications could be:

- Reducing number of intervals
  Like it was already mentioned above it can be said that for spacecraft with constant attitude one interval (average flux of total orbit) is enough for the analysis. For spacecraft with different attitudes at certain orbit positions, the number of datasets are increased by the factor *number of orbit intervals*.

- Reducing number of diameters
  Generally, it can be said that a low number of diameters leads to a rough distribution of the flux. At the same time it is clear that one doesn't need thousands of diameters for a smooth distribution.

- Reducing number of directional and velocity bins
  The less bins of azimuth, elevation and velocity one have, the less data lines are necessary to assign the probability. In normal cases one can say that no more than 36 azimuth bins, 18 elevation bins and 23 velocity bins for the debris population are needed. The definable number strongly depends on the environment model.

- Remove data lines with probability of zero
  Very often one can find probabilities in the distributions that are zero. Deleting these lines with a probability of zero, reduces the file size at lot. Due to the fact that the respective variables in the damage prediction tool are initialized, there is no problem if no value of a certain cell is read.

5. NEXT APPROACHES

In the following the next steps in the development of a standardized interface are listed:

- Consideration of elliptical and transfer orbits
- Consideration of sporadic/stream fluxes
- Consideration of different spacecraft attitudes (exact definition of target positions on orbit)
- Test standardized interface with actual environment model and damage prediction tool
- Design of a user interface for the standard environment interface for implementation into environment model software

In contrast to circular orbits, elliptical orbits are more interesting for analyzing mission with attitude maneuvers because of different velocities in the orbit intervals. Different attitude and velocities lead to different impacts and penetrations in the several orbit intervals. The possibility to perform a prediction analysis for the positions on an orbit, it is necessary to know the position of the spacecraft in its orbit or split the analysis into the intervals of the orbit respectivly. Therefore a definition of the position of the orbit intervals has to be included into the interface file.

The *user interface* has to consider the future development of the environment models as well. The user interface is possibly a kind of menu in the environment model software, where e. g. the user can define in how many bins the azimuth and elevation or the diameter distribution is divided. What excatly can be varied within such a user interface has to be determined. Due to the fact that this should be a standardized interface, of course a standard setting has to be defined.

6. CONCLUSION

A first approach for a standardized interface between orbital debris environment models and damage prediction tools has been presented. This standard considers all relevant environment information that is necessary for a damage analysis of spacecraft. The approach focusses on the content and structure of an examplary interface ASCII file. This file contains the required data that has been adressed in the paper. The next step in developing the interface will be the update of the content and structure considering feedbacks of developers of damage prediction tools and of environment model software. Furthermore the interface will be improved and tested with actual environment model software. The most important questions that have to be answered are which information are necessary to perform the damage prediction analyses and in which format they have to be provided. Furthermore it would be advantageous to know what additional information is useful for the analyses and where the future development of the environment models are going to.

REFERENCES

Liou J. C., et al., 2002, The New NASA Orbital Debris Engineering Model ORDEM2000, NASA Center for AeroSpace Information

IADC WG3 Members, 2004, *Protection Manual*, Inter-Agency Space Debris Coordination Committee