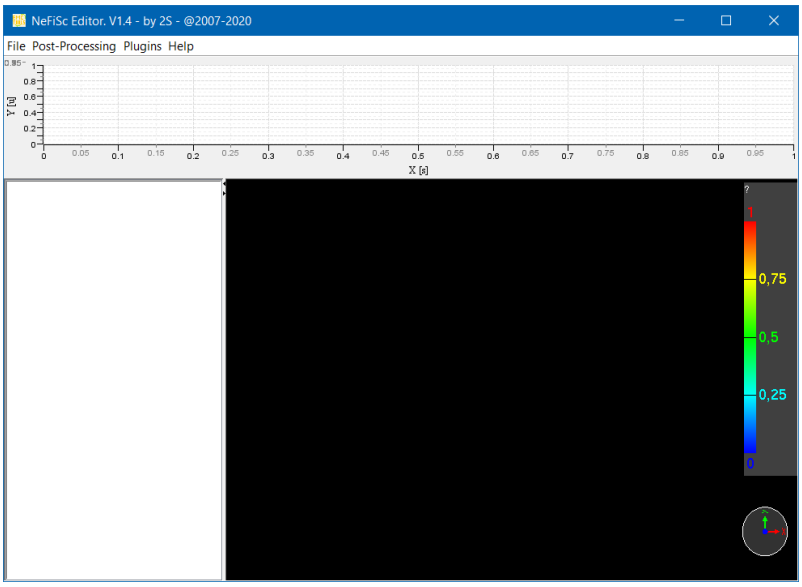
NeFiSc Editor

User's Manual

Version 1.4



SEBASTIEN SERPAUD
31 MARCH 2020

Table of Contents

I. Introduction

This manual document describes the tool NeFiSc Editor (acronym for **Ne**ar-**Fi**eld-**Sc**anning **Editor**) which aims at display and post-processing NFS data results (from measurement or simulation). The NFS data results must be saving in the IEC61967-1-1 ed1.0 format. (Only Cartesian format is taking into account). The software can be downloaded on www.ic-emc.org/nefisc-editor/ .

NeFiSc Editor a free software entirely dedicated to display and post-processing NFS data results. Another use is not guaranty.

The authors have dedicated around several years to build the software and tried their best to improve this tool, trying to keep the usage simple. As the tool is in constant evolution, we encourage the reader to download the updated version from the web page and we would appreciate feedback and comments.

Acknowledgements

I wish to warmly acknowledge Etienne Sicard for encouraging me to capitalise and to share all my research activity in the development of this software. I would also like to thank several people for their positive support and constructive remarks: Alexandre Boyer, Mathieu Beck, Nicolas Lacrampe, Saliha Chetouani and Yannick Poiré.

Toulouse, 31 March 2020

Sébastien Serpaud

II. Getting started with NeFiSc Editor

NeFiSc Editor is a free software entirely dedicated to display and post-processing NFS data results. It developed in Java language and runs on the Java Virtual Machine (JVM). It work on all Operating System (OS) supporting a java virtual machine. Since the JVM is backward compatible, it is usually safe to use a newer JVM to run NeFiSc Editor. We recommend using JVM version greater than or equal to 1.6. NeFiSc Editor is available for both 32bits and 64bits platforms.

Due to NeFiSc Editor uses external libraries, certain limitation must be appear on some OS. All test after each update is done on Windows platform. If you encounter problems on the other platforms please contact us.

II.1. Installation

NeFiSc Editor is a standalone software. No installation required. Unzip installation file on your preferred directory and run it. Check before the first launch to have a Java Runtime Environment (JRE) version greater than or equal to 1.6.

II.2. Overview of the software

The section aims at presenting an overview of the software.

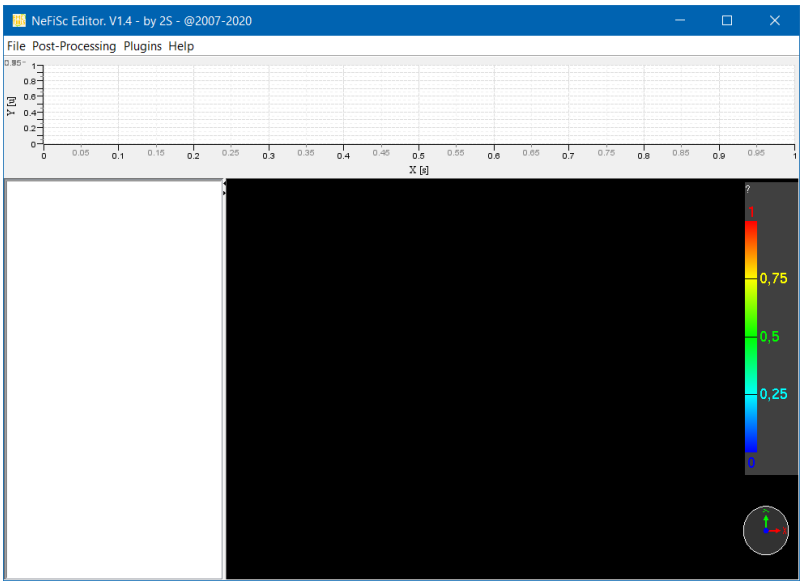


Figure 1 : Overview of NeFiSc Editor

The main commands of IC-EMC are shown in Figure 2- 2.

III. Menus definition

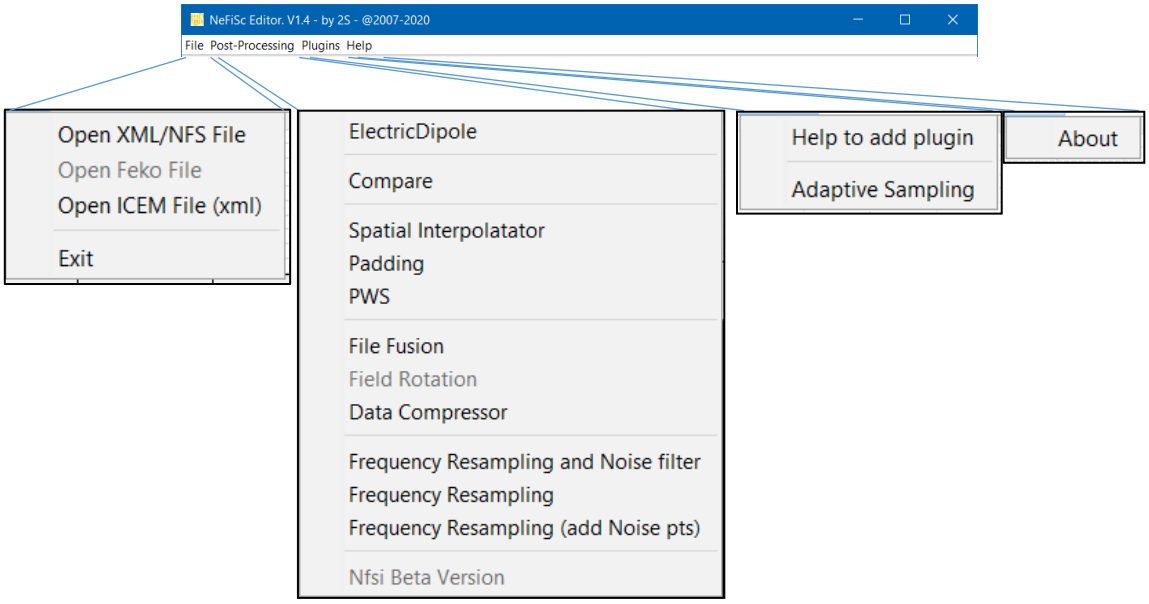


Figure 2 : Main menus

III.1. File menu

III.1.A. Open XML/NFS File

Use this menu to open new XML or NFS File format.

III.1.B. Open Feko File

<in construction>

III.1.C. Open ICEM File (xml)

<in construction>

III.2. Post-Processing menu

<in construction>

III.3. Plugin Menu

<in construction>

III.4. About Menu

<in construction>

IV. XML Near-field scan data exchange format

Before 2009, many different personal formats are used for storing the data, thereby rendering its exchange extremely difficult. Initially discussed by the French research group named PASTEUR, the first version of XML file format is proposed in 2007 [Ser07] and a standard proposal is done in 2009

[She09]. It is now known as the IEC 61967-1-1 Ed1.0 and as the IEC 61967-1-1 Ed.2 [IEC] since 2014. The aim was to propose a common exchange file format for storing all the near-field measurement or simulation results in emission or immunity and both in the frequency or time domains. It is based on the well-known XML format, which is both machine and human readable. Its structure allows the files to be generated and processed on any operating system.

The section structure of the XML format allows storing several types of date (data information about test, images, data results, report files…) without significant limitation.

Figure 3 describes the main structure of the standard XML file. The next chapters describe all main keywords definition. For more detail about this format, please see [IEC].

File_example.xml

XML declaration:

```
<?xml version="1.0" encoding="UTF-8"?>
```

File header:

```
<EmissionScan>
    <Nfs_ver>1.0</Nfs_ver>
    <Filename>File_example.xml</Filename>
    <Date>March 12, 2008</Date>
    <Source>Prepared by A.B. Smith</Source>
    <Disclaimer>
        This file contains results of near-field scan. Other use is
        not guaranteed
    </Disclaimer>
    <Copyright>Copyright 2008, XYZ Corp.,All Rights Reserved
    </Copyright>
    <Notes>Use this section for any special notes</Notes>
    <Documentation>
       Project doc.pdf
       Measurement_descr.doc
    </Documentation>
    :
```

File keywords:

```
    <Component>...</Component>
    <Setup>...</Setup>
    <Probe>...</Probe>
    <Data>
        <Coordinates>xyz</Coordinates>
        :
        <Frequencies>...</Frequencies>
        <Measurement>
            <Format>AM</Format>
            <Unit>uV</Unit>
           :
        </Measurement>
    </Data>
</EmissionScan>
```

Figure 3 : Main structure of XML file

IV.1. File keywords

IV.1.A. Component keywords

This keyword defines the component or board information, target of the measurement. The Component keyword and all children keywords are optional. Figure 4 and Figure 5 define all Component keywords.

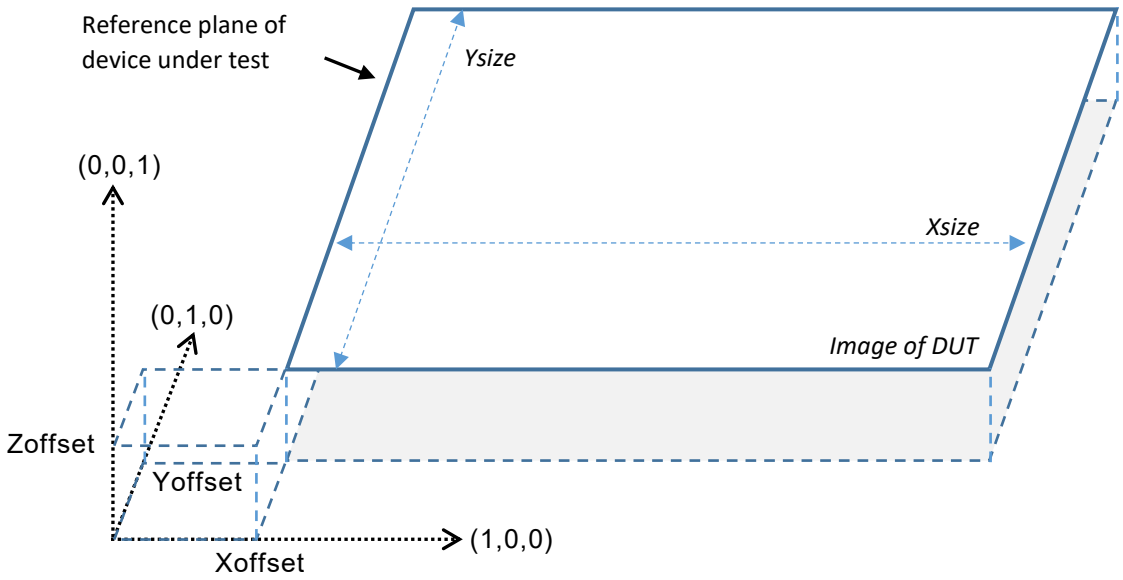


Figure 4 : Image keywords definition

```
<Component>
      <Notes>Created by XYZ Corp.</Notes>
      <Documentation>MyDutDescription.pdf</Documentation>
      <Name>MyDutName</Name>
      <Manufacturer>XYZ Corp.</Manufacturer>
      <Image>
         <Path>MyDut.png</Path>
         <Unit>mm</Unit>
         <Xsize>10</Xsize>
         <Ysize>8</Ysize>
         <Xoffset>1</Xoffset>
         <Yoffset>2</Yoffset>
         <Zoffset>0.5</Zoffset>
      </Image>
</Component>
```

Figure 5 : Example of Component keywords definition

IV.1.B. Setup keywords

This keyword defines all setup information. The Setup keyword and all children keywords are optional. Figure 6 presents the common setup for NFS emission measurement.

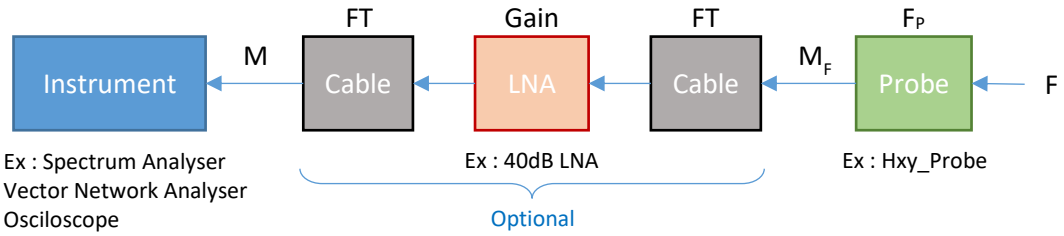


Figure 6 : diagram of setup

Config keyword defines the configuration of instrument used. For example, if spectrum analyser is used to capture the signal of near-field probe, we can defines values of: Attenuation, Reference level, Resolution Bandwidth, Video Bandwidth, Sweep time… . The Transducer keyword defined the transfer function of all equipment's connected between instrument and the near field probe (ex: cable losses and preamplifier gain). The following figure defines all Setup keywords.

```
<Setup>
      <Notes>Recepteur:  RefXYZ</Notes>
      <Config>
         <Att>10</Att>                    <!– Attenuation value -->
         <Ref_level>0</Ref_level>         <!– Reference level value -->
         <Rbw>12e+04</Rbw>                <!– Resolution Bandwidth value -->
         <Vbw>8e+06</Vbw>                 <!– Video Bandwidth value -->
         <Swp>5.0</Swp>                   <!– Sweep time value -->
      </Config>
      <Transducer>
         <Format>ri</Format>              <!– omitted; 'AM'; 'RI' -->
         <Frequencies>
            <Unit>Hz</Unit>
            <List>f0 f1 f2 f3</List>
         </Frequencies>
         <Gain> -37.2 0.33 -37.1 0.27 -36.9 0.19 -36.8 0.11</Gain>
              <!– R@f0  I@f0   R@f1  I@f1   R@f2  I@f2  R@f3   I@f3 -->
      </Transducer>
</Setup>
```

Figure 7 : Example of Setup keywords definition

IV.1.C. Probe keywords

This keyword defines all probe information. The Probe keyword and all children keywords are optional. The following figure defines all Probe keywords.

```
<Probe>
      <Name>MyHxyProbe</Name>
      <Field>Hx</Field>                    <!—'Hx';'Hy';'Hz';'Ex';'Ey';'Ez' -->
      <Frequencies>
         <Unit>Hz</Unit>
         <List>f0 f1 f2 f3</List>
      </Frequencies>
      <Perf_factor>
         <Unit>ohm.m</Unit>                <!— cf : Figure 9 -->
         <Format>ri</Format>               <!— omitted; 'AM'; 'RI' -->
         <List> 0.004077 0 0.004108 0 0.004134 0 0.004154 0 </List>
              <!— R@f0    I@f0   R@f1  I@f1   R@f2  I@f2  R@f3   I@f3 -->
      </Perf_factor>
</Probe>
```

Figure 8 : Probe factor linear units

The probe factor relates the measured or applied value at its input or output connection (e.g. power in dBm, voltage or current) to the generated or applied field strength. The performance factor of probe is defined by the Perf_factor keyword.

$$F_p = \frac{M_F}{F}$$

		Field strength units (F)	
		A/m	V/m
Signal captured units (M_F)	V	ohm·m	m
	A	m	s·m
	W	ohm·m²	s·m²

Figure 9 : Probe factor linear units

IV.1.D. Data keywords

The next figure presents the main keywords of the data section.

File keywords:

```
<Data>
    <Coordinates>xyz</Coordinates>
    :
    <Frequencies>...</Frequencies>
    <Measurement>
        <Format>AM</Format>
        <Unit>uV</Unit>
        <Datafileformat>ASCII</Datafileformat>
        :
    </Measurement>
</Data>
```

Figure 10 : Cartesian coordinate system definition

1) Coordinates keyword

The coordinate system defines how points are stored. Only the both Right-handed Cartesian and Left-handed Cartesian coordinates system are available in NeFiSc Editor. The probe orientation is not taking into account. The next figures present the specific values of the Coordinates keyword following the coordinate system used to store data.

Keyword value	Coordinate system	Order of axes
None or **<Coordinates>**xyz**</Coordinates>** (Default)	Right-handed Cartesian	x, y, z
<Coordinates>-xyz**</Coordinates>**	Left-handed Cartesian	x, y, z

Figure 11 : Coordinate keyword definition

Right-handed Cartesian (Default)

Left-handed Cartesian (Default)

Figure 12 : Cartesian coordinate system definition

2) *Frequencies keyword*

The frequencies keyword, in the data section, defines that data results are represented in frequency domain. Only frequency domain is available in NeFiSc Editor. The unit and frequency list are defines as shown in Figure 13.

```
<Frequencies>
      <Unit>Hz</Unit>
      <List>f0 f1 f2 f3</List>
</Frequencies>
```

Figure 13 : Example of Frequency keyword definition

3) *Measurement keyword*

(a) Format keyword

The Format keyword specifies the format of the data in frequency domain. The value shall be "ma", meaning "magnitude and angle", or "ri", meaning real and imaginary" (see 4.8.5). If this keyword is omitted in the Measurement section, the format of the data is assumed to contain only magnitude information. The units of the magnitude, real and imaginary information are specified by the keyword: Unit in the measurement section. The units of the angle information are degrees.

(b) Unit keyword

The Unit keyword specifies the units of the NFS measurement or simulation data. If this keyword is omitted the units are assumed to be "dBm". The value shall be : V = volt; A = ampere; W = watt; ohm. The scaling factors can be used. Valid scaling factors are:

T = tera: 1e12	k = kilo: 1e3	n = nano: 1e-9
G = giga: 1e9	m = milli: 1e-3	p = pico: 1e-12
M = mega: 1e6	u = micro: 1e-6	f = femto: 1e-15

Units may be expressed as simple linear units or as logarithmic units (dB). Ex : Linear unit : “uV”; Logarithm units : “dB(uV)”.

(c) Datafileformat keyword

Datafileformat keyword specifies the format of the files containing a list of NFS measurement or simulation data. The value shall be "ASCII" for ASCII format or "BIN32" for Float 32-bit binary format conforming to IEEE Std 754™-2008. All data files shall use the same format. Datafileformat keyword is not required. "ASCII" is default value.

IV.2. .NFS file extension

In order to limit the size of files, it is possible to compress all file in .ZIP File Format. The extension may be .nfs. When compressing the file system, care shall be taken to include the paths of the various XML and data files in the compressed file. This ensures that, when decompressed, the file structure is conserved. The paths are not required when all files are stored in the same directory. In order to ensure portability and compressibility, only relative paths can be used to define a path name. An absolute path is not exportable and is not permitted.

References

[Ser07] S. Serpaud, B. Vrignon, E. Sicard, “Common Standard Proposal for Near-Field Data Exchange”; EMCCOMPO 2007; Turin

[She09] J. Shepherd, A. Nakamura, F. Lafon, E. Sicard, M. Ramdani, D. Pommerenke, G. Muchaidze, S. Serpaud "Developing a Universal Exchange Format for Near-Field Scan Data", oral presentation at IEEE EMC Symposium Austin, Texas, USA, 2009

[IEC] IEC TR 61967-1-1:2015; Integrated circuits - Measurement of electromagnetic emissions - Part 1-1: General conditions and definitions - Near-field scan data exchange format; https://webstore.iec.ch/publication/23185