This document defines the SRS2XESAMPLEFLAG software requirements. These requirements will be used as the basis for design and acceptance testing. Both functional and non-functional requirements are defined. The functional requirements define the tasks the software shall perform; its interaction with other systems; and its capability to protect data. The non-functional requirements define the required level of performance; the resources that may be used; ease of use; ease of maintenance; and its capability to run in different environments.

Summary

SRS2XESAMPLEFLAG is a software package. In doing so, SRS2XESAMPLEFLAG represents the automated (batch mode) and generalized version of the ECAL capability of the interactive software tool WEB-GRAPE Version 1.4.
## Document history

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Author</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>19 June 2009</td>
<td>Andreas Becker</td>
<td>Version 1.0 attached to the RFP</td>
</tr>
</tbody>
</table>
Contents

1. Scope .......................................................................................................................... 4
  1.1. Identification ........................................................................................................ 4
  1.2. Rationale .............................................................................................................. 4
  1.3. System overview .................................................................................................. 4
  1.4. Document overview ........................................................................................... 6
2. References .................................................................................................................. 7
3. Functional requirements .............................................................................................. 8
4. Acceptance testing requirements .............................................................................. 13
5. Documentation requirements ...................................................................................... 13
6. Security requirements ............................................................................................... 13
7. Portability requirements ............................................................................................ 13
8. Performance requirements ......................................................................................... 14
10 Terminology ............................................................................................................... 14
    Glossary ..................................................................................................................... 14
    Abbreviations .......................................................................................................... 16
1. SCOPE

1.1. Identification

This document applies to the SRS2XESAMPLEFLAG package Version 1.0.

1.2. Rationale

The software is developed to provide an automated functionality that flags any xenon sample to be potentially affected by the xenon releases from a global inventory of known emitters. This can be done by an automation and generalization of the Event Calculator (ECAL) function within the interactive PTS ATM analysis tool WEB-GRAPE [1, 2]. The software is designed to run on any Unix-based system. The software shall supplement the PTS web services in the field of ATM.

1.3. System overview

The four-layered ATM software system of the PTS is rather a layered than modular structured one as the data flow is always directed towards the next higher numbered layer. The modularity, however, is in place with regard to the capability of the system to exchange tools covering functionality of one of the four layers. The layers have the following functionality assignment (with the name of IDC SW tools currently in charge given in brackets):

- Layer 1: Pre-Processing: Retrieval of appropriate wind-field and other input data required. (NCEPDATA, ECMWFDATA).
- Layer 2: Receptor oriented transport simulation (FLEXPART_5.1, HYSPLIT_4.8) with the so-called source-receptor sensitivity (SRS) fields as final result.
- Layer 3: Post-Processing based on the SRS fields. In doing so products pertaining to the radionuclide samples taken are calculated. This includes the several Field of Regard (SRS2FOR, processing part) and possible source region (PSR) products. The belonging layer 3 executables have been implemented also into WEB-GRAPE.
- Layer 4: Visualization of the layer 3 output in both, batch mode (SRS2FOR, plotting part) and interactive mode. The latter is covered by the interactive analysis tool WEB-GRAPE.

All these layers (see also Figure 1) run in batch mode. Layers 3 and 4 comprise the post-processing of the core ATM results, the source-receptor sensitivity (SRS) fields being available after completion of Layer 2. The SRS fields contain almost all information required to backtrack the radionuclide (RN) measurements taken every day in the global IMS RN network to their Fields-of-Regard (FOR) and Possible Source Regions (PSR). In this context SRS2XESAMPLEFLAG shall be a software package comprising standalone command line based utility tools for batch mode calculation of the ATM products FOR and PSR on basis of the SRS fields.
Figure 1: Illustration of the 4-layer workflow in the field of atmospheric transport modelling as implemented at the PTS (Figure taken from Fig.5 of Kalinowski et al. 2008 [3])

For each RN sample there is one SRS field file consisting of one header line and a number of data lines as follows:

**Header line:**

Longitude of station / latitude of station/ measurement (collection) start time / Measurement stop time [UTC] / Total mass [Bq] released in backward model / Maximum number of hours backward / Output frequency [hours] / averaging time [hours] / resolution of output grid in longitude / latitude [degrees] / station identifier [as specified in notification mail message]

**Data lines:**

Latitude of grid point / longitude of grid point / backward time step number / value of source-receptor sensitivity [identical to retro-plume concentration; Bq/m3]

**Example:**

```
-12.33 -37.00 20030131 00 20030201 00 0.13E+16 144 3 3 1.00 1.00 "RN068"
-38.00 -14.00 1 0.6063586E+00
-37.00 -14.00 1 0.6299873E+00
-38.00 -13.00 1 0.1569079E+02
-37.00 -13.00 1 0.1574570E+02
-38.00 -12.00 1 0.1637941E+01
-37.00 -12.00 1 0.1625708E+01
-38.00 -15.00 2 0.1696579E-01
-37.00 -15.00 2 0.1694598E-01
-38.00 -14.00 2 0.5807510E+01
-37.00 -14.00 2 0.5745895E+01
-38.00 -13.00 2 0.4009043E+02
```

**Naming conventions for PTS in-house calculations:**

Depending on the transport model that was utilized the SRS files are named as follows:

- RNxxx.fp.YYYYMMDDhh.f5.srm.gz
- RNxxx.hy.YYYYMMDDhh.h4.srm.gz
So the clear text ASCII files are compressed with gzip. With RNxxx being the station ID as defined in the so-called ‘gard.dat’ configuration file, YYYYMMDDhh the date of collection stop as assumed in the model run. In the above example, the file is named as follows:

RN068.fp.2003020100.f5.srm.gz

**Naming conventions for comparison exercises (e.g. with WMO centres)**

Depending on the meteorological centre (supplier) of the data the SRS files are named as follows:

RNxxx.YYYYMMDDhh.<mcid>.txt.gz

Whereby the <mcid> denotes the 4-digit meteorological centre identifier. For the PTS results the <mcid> is ‘CTBT’ so the above example would be named as follows:

RN068.2003020100.CTBT.txt.gz

### 1.4. Document overview

This document defines the SRS2XESAMPLEFLAG version 1.0 software requirements. These requirements will be used as the basis for design and acceptance testing.

Both functional and non-functional requirements are defined. The functional requirements define the tasks the software shall perform; its interaction with other systems; and its capability to protect data. The non-functional requirements define the required level of performance; the resources that may be used; ease of use; ease of maintenance; and its capability to run in different environments.

Each mandatory, testable requirements is stated using the word **shall**. Therefore, each **shall** in this document should be traceable to a documented test. Each mandatory, non-testable requirement is stated using the word **will**. Each recommended requirement is stated using the word **should**. A permissible course of action is stated using the word **may**. This convention is used in [12207].

This document is compliant with the Software Requirements Specification described in [DSTD].
## 2. REFERENCES

<table>
<thead>
<tr>
<th>Reference</th>
<th>Number/Author</th>
<th>Title</th>
<th>Organisation / Journal</th>
<th>Revision/Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSTD</td>
<td>IDC-120/01/xx</td>
<td>IDC Software Documentation Framework</td>
<td>CTBTO/PTS</td>
<td>15 February 2002</td>
</tr>
<tr>
<td>9126</td>
<td>ISO/IEC 9126</td>
<td>Information technology - Software product evaluation - Quality characteristics and guidelines for their use</td>
<td>ISO/IEC</td>
<td>1991(E)</td>
</tr>
<tr>
<td>2</td>
<td>CTBTO/WEBGRAPE/SRS</td>
<td>WEB-GRAPE 1.5 Software Requirements Specification</td>
<td>CTBTO/PTS</td>
<td>24 September 2007</td>
</tr>
<tr>
<td>3</td>
<td>Kalinowski, Becker, Saey, Tuma, Wotawa</td>
<td>The Complexity of CTBT Verification. Taking Noble Gas Monitoring as an Example</td>
<td>Complexity</td>
<td>26 March 2008</td>
</tr>
<tr>
<td>4</td>
<td>TR/2004-1</td>
<td>CTBTO-WMO Experiment on Source Location Estimation</td>
<td>CTBTO/PTS</td>
<td>July 2004</td>
</tr>
<tr>
<td>5</td>
<td>Becker et al.</td>
<td>Global backtracking of anthropogenic radionuclides by means of a receptor oriented ensemble dispersion modelling system in support of CTBT verification</td>
<td>Atmospheric Environment (41), 4520-4534</td>
<td>7 December 2006</td>
</tr>
<tr>
<td>6</td>
<td>CTBTO/IDCDB/DEV</td>
<td>The IDC Database Schema Documentation Server: DEVELOPMENT</td>
<td>CTBTO/PTS</td>
<td></td>
</tr>
</tbody>
</table>
3. FUNCTIONAL REQUIREMENTS

1. I/O Management of the SRS2XESAMPLEFLAG package
   1.1. The usual I/O syntax for command line based data processing tools shall apply
   1.2. Input file is the SRS data file pertaining to the xenon sample regarded. Further requirements to control of the SRS data are specified in requirements 2.x
   1.3. Besides the SRS data input file there shall be also the emission inventory file stating the known sources to be regarded. Further requirements are specified in requirements 3.x
   1.4. The software shall be capable to read all input files in uncompressed mode (clear text ASCII files) but also in UNIX (LZW) compressed mode (*.Z files) and gzip compressed mode (*.gz files). Hence a gzip/LZW decoder needs to be involved.
   1.5. The software shall be capable to point to the archive, where the SRS data pertaining to the sample considered resides. Further requirements are specified in requirements 4.x
   1.6. The software shall be capable to choose between those available ATM models that were utilized to generate the SRS data pertaining to the sample. Further requirements are specified in requirements 5.x
   1.7. The software shall be capable to set the threshold activity concentration to be exceeded for flagging a sample. Further requirements are specified in requirements 6.x
   1.8. The software shall be capable to provide an ASCII clear text output file that is comprehensive in content by comprising the aggregated and daily differentiated source-receptor sensitivities between the (known) sources and the receptor (xenon sample) as well as the binary flagging information.
      1.8.1. For the flagging the user shall have two options depending whether he is interested in impact of the earliest emissions from known source locations or the most impactful (strongest).
      1.8.2. All further details are specified in requirements 7.x
   1.9. The software shall provide the essential time differentiated sample and source specific source-receptor sensitivities in a one record *.csv file that can be aggregated with the *.csv files form other related samples for further processing (e.g. Network processing). All further details are specified with requirements 8.x
   1.10. The software shall be capable to make appropriate entries of the flagging results into the xenon data base. All further details are specified in requirements 9.x
   1.11. It is up to the user to make useful specifications for the input file names in order to warrant a proper software integration into the PTS ATM, RN and/or web services workflows.
   1.12. The software shall be capable to assist the user in the formulation of proper and unique file names for the files generated with requirements 7.x and 8.x.

2. SRS2XESAMPLEFLAG –i<SID> or –i<StationName><CS date&hour> options:
   Requirements 2.x for choice of SRS data input via sample ID or station and CS_date_time
   2.1. The SRS input file is uniquely specified with –i<StationName> <CS-Date> <CS hour>. CS denotes the collection stop of the sample regarded.
   2.2. The software should also be capable to specify the SRS data input file by the unique Sample ID being provided by the option –SID instead.
2.2.1. Therefore, the software shall be capable to identify the `<SampleID>` pertaining to the `<StationName> <CS Date> <CS hour>` of the xenon sample regarded by appropriate queries to the IDC Xenon DB.

2.2.2. Vice versa the software shall be capable to identify the `<StationName><CS-Date><CS_hour>` and thus the pertaining SRS data from the `<SampleID>` in case the user provides it.

3 SRS2XESAMPLEFLAG -e<xenon-emission-file> options Requirements 3.x for the emission data (known xenon sources) input

3.1. The software shall be capable to parse the information of the emissions from known xenon sources via a xenon sources inventory input file, parsed by the command line option -e<xenon-emissions-file>.

3.1.1. In doing so a clear text name list file `<xenon-emissions-file>` is read in, where the user can define the sources simply by stating source latitude (f6.3, negative values for southern hemisphere), longitude (f6.3, negative values west of Greenwich until date line), emission start and end, and the emission strength [Bq] (E10.5) for the pertaining sample and SRS file as follows (the numbers provided are only for giving example choices):

<table>
<thead>
<tr>
<th>Source; Name_&lt;Id&gt;</th>
<th>Long [deg]</th>
<th>Lat [deg]</th>
<th>Release Start [h prior to CS]</th>
<th>Release End [h prior to CS]</th>
<th>Daily Release [Bq]</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPF_CLR-Lab_&lt;Id&gt;</td>
<td>27.970</td>
<td>46.050</td>
<td>-1</td>
<td>24</td>
<td>10.22200EE+12</td>
</tr>
<tr>
<td>IPF_Fleurus_&lt;Id&gt;</td>
<td>4.467</td>
<td>50.467</td>
<td>144</td>
<td>-1</td>
<td>1.43300EE+12</td>
</tr>
<tr>
<td>IPF_Petten_&lt;Id&gt;</td>
<td>4.467</td>
<td>50.467</td>
<td>-1</td>
<td>-1</td>
<td>5.31911EE+11</td>
</tr>
<tr>
<td>IPF_Pelindaba_&lt;Id&gt;</td>
<td>27.920</td>
<td>-25.798</td>
<td>144</td>
<td>120</td>
<td>2.42217EE+11</td>
</tr>
<tr>
<td>IPF_LucasHeights_&lt;Id&gt;</td>
<td>150.981</td>
<td>-34.052</td>
<td>-1</td>
<td>-1</td>
<td>3.42100EE+10</td>
</tr>
<tr>
<td>DPRK_&lt;Id&gt;</td>
<td>129.100</td>
<td>41.300</td>
<td>-1</td>
<td>-1</td>
<td>1.00000EE+15</td>
</tr>
</tbody>
</table>

3.1.2. The method of aggregating the point source emission to the belonging grid cell or cells of the SRS data can range from nearest neighbour to higher order interpolation. A final decision shall be made during the requirements refinement phase of the project.

3.1.3. The release start time is determined by counting back the hours from the collection stop date and time. If ‘-1’ is entered, the tool shall consider the default, i.e. the entire range of the SRS field covered as emission period.

3.1.4. The release stop time is determined by counting back the hours from the collection stop date and time. If ‘-1’ is entered, the tool shall consider the default, i.e. the collection stop time & date of the sample and the pertaining SRS field examined.

3.1.5. The tool shall issue an error message if the release period resulting from the difference of release start and release stop is zero or negative, telling

**Effective Release Start provided for <source name> is later then Release End. Please revise your specifications made in the <xenon-emissions> name list file**

3.1.6. The most right column in the emission inventory file shall state a daily release amount [Bq] that is regarded constant for the release period defined in requirement 3.1.5.

3.1.6.1. This allows the user to differentiate the release strengths day by day, by altering the release start and end entries, e.g. in a series of xenon-emission-inventory files.

3.1.7. The most left column shall list the source name with the source ID appended. The source ID is resulting from a pertaining table of the IDC Xenon DB. A description and template of the table will be provided by the PTS.
4 SRS2XESAMPLEFLAG -sdh<SRS-Arch> options: Requirements 4.x to choose the SRS archive

4.1. The software shall be capable to find the SRS data pertaining to the sample regarded from different SRS data archives provided in a flat file system.

4.1.1. The argument for the -sdh<SRS-Arch> is the name of the entry in the srs_data_home.txt configuration file to use. This file the argument provided with the path to the pertaining SRS data archive and the pertaining stations name list file. An example implementation is given in the text box below:

```
# Define possible Archive settings
# Each entry consist of three consecutive lines containing
# Name in square brackets (pointer variable for SRS2KML and SRS2XESAMPLEFLAG tools)
# path = Path to SRS_DATA_HOME (i.e. dir containing dates)
# gard = Path to corresponding RN station name list file
[default]
  path=/dvlscratch/ATM/ATM_SRS_ARCHIVE
  gard=/home/rsmon/becker/WEBGRAPE-admin/data/gard.dat
[xenon-archive]
  path=/dvlscratch/ATM/NG_SRS_ARCHIVE
  gard=/home/rsmon/becker/WEBGRAPE-admin/data/gardx.dat
[EU-JA]
  path=/dvlscratch/ATM/EU_SRS_ARCHIVE
  gard=/dvl/atm/becker/atl-xls/atm-layer2/SRSM-MODEL/gard.dat.special
[alternate-xenon-archive]
  path=/dvlscratch/ATM/NG_SRS_ARCHIVE
  gard=/home/rsmon/becker/WEBGRAPE-admin/data/gardxl.dat
```

4.1.2. In each archive folder (e.g. ~/ATM_SRS_ARCHIVE) the SRS data is sorted according to station name and collection stop date and hour, so the association between unique SampleID and these parameters is provided (see requirements 2.x) for any archive chosen.

5 SRS2XESAMPLEFLAG -m<Model> options Requirements 5.x to choose the model that calculated the SRS data of the chosen archive.

5.1. The software shall be capable to choose model system (ATM layer 1 and 2 configuration, see Figure 1) SRS data as input data.

5.1.1. The naming conventions utilized by the PTS in its SRS data archive is <ATM-layer2-tool>.<ATM-layer1-tool>.<model-level> and allows therefore to choose the underlying dispersion model and wind field analysis.

5.1.2. For example `-m flexpart.ecmwf.l1` chooses the surface level (l1) output of the Lagrangian Particle Dispersion Model FLEXPART driven by the ECMWF analysis wind field.

5.1.3. If the user chooses an option that is not supported by the SRS data, an message ‘file not available’ should be issued.

6 SRS2XESAMPLEFLAG -t<threshold-activity-concentration> Requirements 6.x for setting the threshold activity concentration to be exceeded for flagging

6.1. The software shall be capable to parse the threshold activity concentration [mBq] that needs to be exceeded before a flag is set.

6.1.1. As the call is specific for the sample examined, the setting can be sample specific as well.
6.1.2. Non exceedance of this threshold shall prevent the software from setting (and later aggregating) pertaining flags, but it shall not prevent the output of sensitivity data as stipulated later under requirements 7.x.

7 SRS2XESAMPLEFLAG –o(ASCII-output-filename> [-flag<earliest | strongest]

Requirements 7.x for the content of the comprehensive clear text output file

7.1. The software shall be capable to put its results into a clear-text ASCII output file, with a name specified by the user with the command line option –o<ASCII-output-filename>

7.2. The output file shall only be generated and written only if the user used the –o option. Otherwise the default is, not to write this ASCII-output-file.

7.3. The output file name shall be unique, so it can be traced back to the pertaining xenon sample and SRS file utilized. Moreover the process ID pertaining to the runtime instance of SRS2XESAMPLEFLAG shall be encoded into the filename to make it unique.

7.3.1. In doing so the output file name will be extended by a number composed from the sample ID and the process ID retrieved from the UNIX system.

7.4. The output shall feature a header line that shall just repeat the full command that generated the output.

7.5. Below the header line the output shall feature a clear text matrix that lists record by record the release 21 days (lines 6-26) headed by a title line (line 3) a source strength values record (line 3) and a two records (lines 4-5) that aggregate the information of the 21 lines below, whereby line 4 is a descriptor of the results provided in line 5.

7.6. For each source (main column) there shall be two sub-columns that provide on each left column the 24h aggregated source-receptor-sensitivity, and on each right column a binary flag telling whether the product of the aforementioned sensitivity times the 24h release strength would yield an exceedance of the threshold activity concentrations (flag = 0 for no; 1 for yes)

7.7. In the aggregation line 3 the left column shall show the 21days sum of the daily source-receptor-sensitivities. The right column shall summarize the binary flag and tell in case of existence of at least one positive flag during the 21 days stipulated below a number telling the number of days prior to the CS date when either

7.7.1. the first occurrence of a flagging happened (if –flag ‘earliest’) is chosen

7.7.2. the strongest exceedance of the threshold activity concentrations happened (if –flag ‘strongest’) was chosen

7.7.3. The default value of –flag is ‘earliest’.

7.8. The following text box shall show an example output to visualize the concept. (Please note that the most left line number column shall NOT be in the file, but is provided here to facilitate orientation). The command that invoked this text file shall be in line 1. In line 2 there shall be the information of the data the tool has retrieved from the database, e.g. with a mechanism that feeds also the pertaining ARR available as html and xml file.
### 8 SRS2XESAMPLEFLAG --csv(CSV-encoded-output-file> [-flag-only] Requirements 8.x for the content of the clear text sensitivity output

8.1. In addition to the comprehensive clear-text output the software shall create for each source found in xenon-emissions.txt file a *.csv files with a unique file name `<Station>_<sampleID>_<Sources1>.csh; <Station>_<sampleID>_<Sources2>.csh, etc., comprising in one record (line) just the 21 daily source-receptor sensitivities, (day01, day02,...,day21) proceeded by the Collection Stop (CS) date & time stamp, to allow for transposition of the relative times (e.g. Day02) into the corresponding absolute ones (e.g. 20090616 for Day02 denoting all emissions in the 24 hours after that CS date)

8.1.1. An example record would read like this (top record NOT written out!)

<table>
<thead>
<tr>
<th>CS_Date&amp;Time</th>
<th>Day01</th>
<th>Day02</th>
<th>Day03</th>
<th>Day04</th>
<th>Day05</th>
<th>...</th>
<th>Day20</th>
<th>Day21</th>
</tr>
</thead>
<tbody>
<tr>
<td>2009061712</td>
<td>1.1E-15</td>
<td>8.2E-16</td>
<td>1.2E-16</td>
<td>2.2E-16</td>
<td>8.4E-17</td>
<td>...</td>
<td>8.2E-18</td>
<td>1.5E-19</td>
</tr>
</tbody>
</table>

### 9 SRS2XESAMPLEFLAG –d(DB-connect-string> options Requirements 9.x for the entry of the flagging result into the xenon DB

9.1. There shall be an entry into an extra table of the IDC xenon database that comprises the same flagging and relevant source-receptor sensitivity information as described for the clear-text output with requirements 7.x

9.2. The structure of the table will be provided by the PTS. It shall contain all relevant SRS2XESAMPLEFLAG output information together with all parameters required for backward engineering the input information.

9.3. The final structure of the table will be defined during the refinement phase of this Software Requirement Specifications document.

### 10 Runtime Environment:

10.1. The SRS2XESAMPLEFLAG software shall run as standalone application under UNIX/LINUX. Please note that if the implementation strategy is to generalize the ECAL function of WEB-GRAPE, the solution would need to employ the IDL Virtual Machine, the freely available IDL runtime environment that needs to be installed on the system at Version 6.4 or higher.

### 11 Supported operating systems

All platforms supported by IDL_6.4\(^1\) shall be supported, with particular emphasis on

11.1. SPARC 32/64bit, Solaris 9, 10

11.2. Intel/AMD x86 32bit Linux, Kernel version 2.4/2.6, glibc version 2.3

\(^1\) [http://www.itris.com/download/download.asp?searching=Downloads&DownloadType=DownloadProductsVersion=279&Platform=All&DownloadType=ProductDownload&Submit=Submit](http://www.itris.com/download/download.asp?searching=Downloads&DownloadType=DownloadProductsVersion=279&Platform=All&DownloadType=ProductDownload&Submit=Submit)
11.3. Intel/AMD x86_64 64-bit Linux, Kernel version 2.4/2.6, glibc version 2.3
11.4. Intel/AMD x86 32-bit Windows XP, Vista (in case of a IDL VM based solution)
11.5. Intel/AMD x86_64, 64-bit Windows XP, Vista (in case of a IDL VM based solution)

12. Hardware requirements
12.1. The software shall be able to execute, completely meeting all requirements, on a twin 450 MHz processor SUN Ultra 60 with 512 MB of RAM, running Solaris (version 9 or later).
12.2. The software shall be capable to execute, completely meeting all requirements besides accelerated navigation and animation, on a 2.0 GHz AMD Opteron or higher with 1024 Mb of RAM, running the Linux or Windows OS. The later requirement is only made if the platform independent IDL based approach is implemented.

4. ACCEPTANCE TESTING REQUIREMENTS
13. The software will be tested in Phases 2 and 3 of the project at CTBTO against the requirements as described in the updated SRS.

5. DOCUMENTATION REQUIREMENTS
14. All documentation will be written in English in MS Word format
17. New Software Installation Plan (SIP) document describing installation and configuration of the software.
18. New Software man pages, describing usage of the functionality of the software callable from the command line with 'man SRS2XESAMPLEFLAG'
19. New Source code will be documented in the code according to suitable standards.
20. There shall be a help function implemented that is called with the ‘-help’ option and upon any users’ choice of options that is non-compliant with the required syntax.

6. SECURITY REQUIREMENTS
21. There are no specific security requirements for this software despite the normal ones applicable for any batch mode software utility running at the IDC. So it shall be capable to run under a normal PTS user account without requiring su privileges during runtime.

7. PORTABILITY REQUIREMENTS
22. The new software will be available as a download kit and on CD.
23. The new software is allowed to make use of the freely downloadable IDL Virtual Machine if a solution that extracts existing IDL code of the WEB-GRAPE 1.4 software is followed by the supplier. Licensing of the IDL Virtual Machine software is neither necessary nor applicable.
24. The new software can alternatively make use of other software libraries if they are Open Source and if they are shipped with the delivery.
8. PERFORMANCE REQUIREMENTS

25. Processing of an 21 days SRS field and preparation of the data shall not last longer then 5 minutes per sample if the full 21 days release (emission) period is examined.

26. Acceptance requirement is the software’s capability to perform

   26.1. All functional requirements stipulated in Chapter 3 in a way that a full daily flagging procedure is completed for the whole RN network within 3.5 hours.

   26.2. The update of the data base entries is completed within the same temporal constrained.

10. TERMINOLOGY

Glossary

Accuracy: Capability of the software product to provide the right or agreed results or effects. ISO/IEC 9126.

Adaptability: Capability of the software product to be adapted for different specified environments without applying actions or means other than those provided for this purpose for the software considered. ISO/IEC 9126.

Analysability: Capability of the software product to be diagnosed for deficiencies or causes of failures in the software, or for the parts to be modified to be identified. ISO/IEC 9126.

ATM layer 1: Pre-processing. Retrieval of appropriate wind-field and other input data required (NCEPDATA, ECMWFDATA).

ATM layer 2: Receptor oriented transport simulation (FLEXPART_3.2, HYSPLIT_4.6) with the SRS fields as final result.

ATM layer 3: Post-Processing based on the SRS fields. In doing so products pertaining to the radionuclide samples taken are calculated. This includes the several FOR and PSR products. The belonging layer 3 executables shall be easily accessible and utilized by WEB_GRAPE.

ATM layer 4: Visualisation of the layer 3 output in both, batch mode and interactive mode. The latter shall be covered by WEB_GRAPE.

Changeability: Capability of the software product to enable a specified modification to be implemented. ISO/IEC 9126.

Coexistence: Capability of the software product to coexist with other independent software in a common environment sharing common resources. ISO/IEC 9126.

Efficiency: Capability of the software product to provide appropriate performance, relative to the amount of resources used, under stated conditions. ISO/IEC 9126.

Fault tolerance: Capability of the software product to maintain a specified level of performance in cases of software faults or of infringement of its specified interface. ISO/IEC 9126.

Functionality: Capability of the software product to provide functions which meet stated and implied needs when the software is used under specified conditions. ISO/IEC 9126.

Installability: Capability of the software product to be installed in a specified environment. ISO/IEC 9126.
<table>
<thead>
<tr>
<th>Capability</th>
<th>Definition</th>
<th>ISO/IEC 9126</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interoperability</td>
<td>Capability of the software product to interact with one or more specified systems.</td>
<td>9126</td>
</tr>
<tr>
<td>Learnability</td>
<td>Capability of the software product to enable the user to learn its application.</td>
<td>9126</td>
</tr>
<tr>
<td>Maintainability</td>
<td>Capability of the software product to be modified. Modifications may include corrections, improvements or adaptation of the software to changes in environment, and in requirements and functional specifications.</td>
<td>9126</td>
</tr>
<tr>
<td>Maturity</td>
<td>Capability of the software product to avoid failure as a result of faults in the software.</td>
<td>9126</td>
</tr>
<tr>
<td>Operability</td>
<td>Capability of the software product to enable the user to operate and control it.</td>
<td>9126</td>
</tr>
<tr>
<td>Portability</td>
<td>Capability of the software product to be transferred from one environment to another.</td>
<td>9126</td>
</tr>
<tr>
<td>Recoverability</td>
<td>Capability of the software product to re-establish a specified level of performance and recover the data directly affected in the case of a failure.</td>
<td>9126</td>
</tr>
<tr>
<td>Reliability</td>
<td>Capability of the software product to maintain a specified level of performance when used under specified conditions.</td>
<td>9126</td>
</tr>
<tr>
<td>Replaceability</td>
<td>Capability of the software product to be used in place of another specified software product for the same purpose in the same environment.</td>
<td>9126</td>
</tr>
<tr>
<td>Resource behaviour</td>
<td>Capability of the software product to use appropriate amounts and types of resources when the software performs its function under stated conditions.</td>
<td>9126</td>
</tr>
<tr>
<td>Security</td>
<td>Capability of the software product to protect information and data so that unauthorized persons or systems cannot read or modify them and authorized persons or systems are not denied access to them.</td>
<td>9126</td>
</tr>
<tr>
<td>Stability</td>
<td>Capability of the software product to avoid unexpected effects from modifications of the software.</td>
<td>9126</td>
</tr>
<tr>
<td>Stakeholder</td>
<td>Individual or organization that is affected by the system or whose input is needed in order to develop the system. Examples of stakeholders include management, project sponsor, domain area experts, technology experts, system developers, testers, users and lawyers.</td>
<td></td>
</tr>
<tr>
<td>Suitability</td>
<td>Capability of the software product to provide an appropriate set of functions for specified tasks and user objectives.</td>
<td>9126</td>
</tr>
<tr>
<td>Testability</td>
<td>Capability of the software product to enable modified software to be validated.</td>
<td>9126</td>
</tr>
<tr>
<td>Time behaviour</td>
<td>Capability of the software product to provide appropriate response and processing times and throughput rates when performing its function, under stated conditions.</td>
<td>9126</td>
</tr>
<tr>
<td>Understandability</td>
<td>Capability of the software product to enable the user to understand whether the software is suitable, and how it can be used for particular tasks and conditions of use.</td>
<td>9126</td>
</tr>
<tr>
<td>Usability</td>
<td>Capability of the software product to be understood learned, used and attractive to the user, when used under specified conditions.</td>
<td>9126</td>
</tr>
</tbody>
</table>
Use case

A collection of possible scenarios related to a particular goal. The scenarios describe the interaction between a user and the system. Use cases were first proposed by Jacobson I. et al (1992).

---

**Abbreviations**

- ANSI: American National Standards Institute
- API: Application Programming Interface
- ARR: Automated Radionuclide Report
- ASCII: American National Standard Code for Information Interchange
- ATM: Atmospheric Transport Modelling
- bmp: Bitmap
- COG: Centre of Gravity
- CTBTO: Comprehensive Nuclear-Test-Ban Treaty Organization
- DBF: Differential binary FOR
- ECMAL: SRS based forward Event Calculator
- ECMWF: European Centre for Medium-Range Weather Forecasts
- ECMWF DATA: ECMWF data retrieval program
- eps: Encapsulated Postscript
- FOR: Field of Regard
- GIF: Graphics Interchange Format
- GNU: GNU is not Unix
- GRIB: Gridded Binary
- PSI: Global Point Source Indicator
- GUI: Graphical User Interface
- HTTP: Hypertext Transfer Protocol
- https: HTTP over SSL
- IDC: International Data Centre
- IMS: International Monitoring System
- IEC: International Electrotechnical Commission
- IF: Integral FORs
- ISO: International Organization for Standardization
- jpeg: Joint Photographic Experts Group
- lat: Latitude
- long: Longitude
- NCEP: United States National Center for Environmental Prediction
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NCEPDATA</td>
<td>NCEP data retrieval program</td>
</tr>
<tr>
<td>NDC</td>
<td>National Data Centre</td>
</tr>
<tr>
<td>MHz</td>
<td>Megahertz</td>
</tr>
<tr>
<td>MM</td>
<td>Multiple Model</td>
</tr>
<tr>
<td>MMA</td>
<td>MM Average</td>
</tr>
<tr>
<td>MS</td>
<td>Microsoft</td>
</tr>
<tr>
<td>RAFOR</td>
<td>Reduced Area FOR</td>
</tr>
<tr>
<td>RAM</td>
<td>Random Access Memory</td>
</tr>
<tr>
<td>RN</td>
<td>Radionuclide</td>
</tr>
<tr>
<td>OS</td>
<td>Operating System</td>
</tr>
<tr>
<td>OSS</td>
<td>Open Source Software</td>
</tr>
<tr>
<td>PSR</td>
<td>Possible Source Region</td>
</tr>
<tr>
<td>PTS</td>
<td>Provisional Technical Secretariat</td>
</tr>
<tr>
<td>QF</td>
<td>Quantitative FOR</td>
</tr>
<tr>
<td>SDD</td>
<td>Software Design Document</td>
</tr>
<tr>
<td>SHI</td>
<td>Seismic, Hydro-acoustic, Infra-sound</td>
</tr>
<tr>
<td>SRS</td>
<td>Source-Receptor Sensitivity (in most cases)</td>
</tr>
<tr>
<td></td>
<td>Software Requirements Specification (in ‘SRS document’ or in Header)</td>
</tr>
<tr>
<td>SSL</td>
<td>Secure Sockets Layer</td>
</tr>
<tr>
<td>tiff</td>
<td>Tagged Image File Format</td>
</tr>
<tr>
<td>UTC</td>
<td>Coordinated Universal Time</td>
</tr>
<tr>
<td>WEB_GRAPE</td>
<td>Web connected local graphics engine</td>
</tr>
<tr>
<td>WMO</td>
<td>Word Meteorological Organisation</td>
</tr>
</tbody>
</table>