15-May 2017 CO2MPAS Workshop with TSs

1. Check CO2MPAS version.
   Check that your COMPAS version is CO2MPAS ALL-IN-ONE 1.5.5.post0 (the STAMP release).

2. Upgrade CO2MPAS with additional features for the DICE.
   Download from GitHub/ Use the provided USB pen the Updated CO2MPAS version, together with its dependencies. Copy them into the CO2MPAS folder (example C:\Apps\Co2mpas_AIO-v1.5.5\CO2MPAS)
   - PySocks-1.6.7-py3-none-any.whl
   - parsedatetime-2.3-noPytest.tar.gz
   - transitions-0.5.2.tar.gz
   - co2mpas-1.5.7b3-py2.py3-none-any.whl

Install the 4 packages from the CO2MPAS console with the command
pip install (example pip install PySocks-1.6.7-py3-none-any.whl)

3. When it has finished, in order to ensure that everything worked fine, and you got the required version of CO2MPAS:

$ co2mpas --version

$ co2mpas --version
co2mpas_version: 1.5.7.b3

The installation has finished. Now, you can open the CO2MPAS GUI.

4. Now you can proceed with CO2MPAS, as usual. From the CO2MPAS GUI generate the demo files.

5. Proceed with the simulation in Type-Approval mode with Demo-1. Do not forget to set the out Folder. Remember that CO2MPAS in engineering mode will produce outputs that cannot be DICED.

When the simulation has finished, you are ready to proceed with the dice.

DICE work flow.

The next steps are on the CO2MPAS CONSOLE
Note that the list of commands below are just one possible "path" to arrive to the DICE decision. More commands are provided at the bottom of these instructions, to be used in case of problems:

6. Set the configurations for the DICE for your particular case (.co2dice/co2dice_config.py and .co2dice/co2dice_persist.json)

7. Initialize a project, append the input and the output files into the project and generate the contents for the Dice email with the command bellow:

$ co2dice project init -i path_to_input_file/Demo1.xlsx -o path_to_output_folder\output_file _demo-1.xlsx –report

If the process went fine you will be prompted with Report that will be sent to the timestamp server.

8. Now if you type **co2dice project ls**, you will see that the state of the project is now tagged

```
$ co2dice project ls
D:\co2mpas_AIO-v1.5.5\CO2MPAS>co2dice project ls
22:26:31      : INFO:LsCmd:Listing all projects...
* IP-10-AAA-2017-1002: tagged
```

9. In order to see if your configurations are right and you can connect to the SMTP and IMAP, the server of the Sender and the Receiver, type the following command and you must receive a response roughly like this:

$ co2dice tstamp login

```
D:\co2mpas_AIO-v1.5.5\CO2MPAS>co2dice tstamp login
22:27:10      : INFO:TstampSender:Connecting to SMTP(STARTTLS):
dimitris.komnos@outlook.com@smtp-mail.outlook.com({'port': 587})...
22:27:14      : INFO:TstampSender:Connected to SMTP:
dimitris.komnos@outlook.com@<socket.socket [closed] fd=-1,
family=AddressFamily.AF_INET, type=SocketKind.SOCK_STREAM, proto=0>,
ok? True
True
22:27:15      : INFO:TstampReceiver:Connecting to IMAP4_SSL:
dimitris.komnos@outlook.com@imap-mail.outlook.com()...
22:27:25      : INFO:TstampReceiver:Connected to IMAP4_SSL:
dimitris.komnos@outlook.com@<ssl.SSLSocket [closed] fd=-1,
family=AddressFamily.AF_INET, type=SocketKind.SOCK_STREAM, proto=0>,
ok? True
True
```

The TRUE status confirms that the connection to the SMTP and IMAP were successful

10. Assuming everything worked, now you may send the e-mail to the timestamp server:

$ co2dice project tsend

```
$ co2dice project tsend
D:\co2mpas_AIO-v1.5.5\CO2MPAS>co2dice project tsend
22:29:49      : INFO:transitions.core:IP-10-AAA-2017-1002: Exited state tagged
22:29:49      : INFO:Project:Sending email for tstamping...
22:29:50      : INFO:TstampSender:Connecting to SMTP(STARTTLS):
dimitris.komnos@outlook.com@smtp-mail.outlook.com({'port': 587})...
22:29:54      : INFO:TstampSender:Timestamping 2458-char email from
'dimitris.komnos@outlook.com' to ['post@stamper.itconsult.co.uk']--
>['dimitris_tom@hotmail.com', 'dimitriskomnos11@gmail.com',
'Dimitrios.KOMNOS@ext.ec.europa.eu', 'dimitriskomnos@yahoo.com',
'dimitris.komnos@outlook.com']
22:30:11      : INFO:transitions.core:IP-10-AAA-2017-1002: Entered state mailed
22:30:11      : INFO:Project:Committing Project(IP-10-AAA-2017-1002: mailed):
sent stamp-email
true
...
```

11. Check again the state of the current project. It should state mailed.

$ co2dice project ls

```
$ co2dice project ls
D:\co2mpas_AIO-v1.5.5\CO2MPAS>co2dice project ls
22:46:16      : INFO:LsCmd:Listing all projects...
* IP-10-AAA-2017-1002: mailed
```

12. After 5-40 minutes you will receive in your e-mail the dice stamp that has to be decoded with this command:

$ co2dice tstamp recv --raw

This command will parse the response of the server automatically, and will produce the OK/SAMPLE decision-flag. This command will NOT change the state of the project.

13. Decode the DICE decision.

Type in the console

$ co2dice project tparse.

Copy from the above log message (from -----BEGIN PGP SIGNED MESSAGE----- to the end) and paste it in the console by pressing [Shift+INSERT], and send the end-stream character after a new-line,

that is to say: press [Enter], [Ctrl+Z], [Enter], in a row. The dice stamp will be decoded showing at the end of the message the random sampling number [0 to 99], and the OK/SAMPLE decision-flag. In the case of the example, it is a SAMPLE code, and the state of the project become sample.

14. Finally, type:

$ codice project export IP-10-AAA-2017-1002

in order to store (in a compressed .zip file) all the files used in the Dice workflow (ie, the input file to CO2MPAS, the CO2MPAS-TA output file, and the Dice decision). This file has to be sent to the Type Approval authority that will keep it as part of the TA of the vehicle.

To remove a given project from the dice list of projects (and export it as a zip file), type co2dice project export IP-10-AAA-2017-1002 --ExportCmd.erase_afterwards=True.