Project no.: **893945**

Project full title: **Energy flexible DYnamic building CErtification**

Project Acronym: **E-DYCE**

Deliverable number:	**D3.2**
Deliverable title:	**Free running module**
Work package:	WP3
Due date of deliverable:	M18
Actual submission date:	M18- 28/02/2022
Start date of project:	01/09/2020
Duration:	36 months
Reviewer(s):	Flourentzos Flourentzou (ESTIA)
Author/editor:	Giacomo Chiesa (POLITO)
Contributing partners:	Politecnico di Torino (POLITO), Agenzia nazionale per le nuove tecnologie, l'energia e lo sviluppo economico sostenibile (ENEA)

Dissemination level of this deliverable	**PU**
Nature of deliverable	Other

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 893945. Any results of this project reflects only this consortium's view and the European Commission is not responsible for any use that may be made of the information it contains.
Further information is available at www.edyce.eu.

Document history

Version no.	Date	Authors	Changes
0.1	06/12/2021	Giacomo Chiesa	Initiation and deliverable draft
0.2	25/01/2022	Giacomo Chiesa, Francesca Fasano, Paolo Grasso	Draft version including PREDYCE FR functionalities for EDYCE and correlated samples and descriptions
0.3	27/01/2022	Michele Zinzi	Inclusion of section 3.2.3 about Top Vs Tair
0.4	28/01/2022	Giacomo Chiesa	Draft version shared on TEAMS to all E-DYCE members for suggestions
0.5	14/02/2022	Flourentzos Flourentzou	Internal review report
0.6	28/02/2022	Giacomo Chiesa	Internal review-corrected version
1.0	28/02/2022	Anne Bock	Final check and submission to EC

Contributors

Partner no.	Partner short name	Name of the Contributor	E-mail
2	POLITO	Giacomo Chiesa	giacomo.chiesa@polito.it
2	POLITO	Paolo Grasso	paolo.grasso@polito.it
2	POLITO	Francesca Fasano	francesca.fasano@polito.it
6	ENEA	Michele Zinzi	michele.zinzi@enea.it

Table of Contents

1 Executive Summary 5

2 Deliverable objectives and structure 6

3 Free-running extra functionalities 6

3.1 The PREDYCE tool 7

3.1.1 EPW compiler module* 8

3.1.2 PREDYCE scenarios 8

3.2 PREDYCE free-running correlated actions 9

3.2.1 IDF editor 9

3.2.2 KPIs 12

3.2.3 On the relation between air and operative temperature in standard office rooms 14

3.2.4 Fictitious cooling KPI 16

4 PREDYCE scenarios 22

4.1 Sensitivity analysis 22

4.2 Performance gap 25

4.3 Model verification 30

4.4 Sensor's nomenclature discussion 32

5 Conclusions and Outlook 33

6 Bibliography 34

Table of Figures

Figure 1 Potential impact of free-running technologies on expected thermal comfort based on TMY of (a) TPM and (b) Rome-Casaccia 7
Figure 2 Close-up of the ENEA Living Lab lay-out with the position of the monitoring station 15
Figure 3 View of ENEA Living Lab test room 15
Figure 4 Operative and air temperature trend in the test room during the monitoring period 16
Figure 5 Fictitious cooling KPI computation workflow 17
Figure 6 CDFs describing fictitious cooling potentialities in describing cooling needs....18
Figure 7 Correlation between external conditions (temperature) and (a) energy needs, (b) free-running building temperatures – daily aggregated values....19
Figure 8 Correlations between free-running internal temperatures and energy needs, for (a) daily aggregated values and (b) hourly aggregation 19
Figure 9 Correlations between free-running internal temperatures and energy needs, for (a) case with ventilative cooling on both FR and mechanical modes and (b) case with ventilative cooling activated only for the FR mode – daily aggregation....20
Figure 10 Correlations between free-running internal temperatures and energy needs, for (a) case with reduced U-value (0.2 W/m2K) and (b) original case run under TPM typical weather conditions – daily aggregation. 20
Figure 11 Correlations between free-running internal temperatures and energy needs, for (a) case with ventilative cooling on both FR and mechanical modes and (b) original case with reduced U-value (0.2 W/m2K) – daily aggregation 21
Figure 12 Correlations between free-running internal temperatures and energy needs, office scheduled cases, considering only occupied hours (hourly aggregation) (a) occupied hours without turning off and on periods (8-17) (b); only occupied day (daily aggregation 21
Figure 13 Input JSON file used for this sample application 22
Figure 14 TPM shading control thresholds heatmaps 23
Figure 15 Rome shading control thresholds heatmaps 23
Figure 16 TPM overhang length and ACH impact on thermal discomfort....24
Figure 17 Rome overhang length and ACH impact on thermal discomfort....24
Figure 18 TPM adaptive comfort model distribution 25
Figure 19 Rome adaptive comfort model distribution 25
Figure 20 Performance gap input/output workflow....26
Figure 21 TPM school demo case model 27
Figure 22 Input JSON KPIs 27
Figure 23 Input JSON preliminary actions....28
Figure 24 Example of data_res.csv for performance gap 28
Figure 25 Example of data_res_timeseries.csv for performance gap 29
Figure 26 Timeseries results averaged on all classrooms....29
Figure 27 Timeseries results in a specific classroom 29
Figure 28 Example of calibration signatures....30
Figure 29 Indoor drybulb temperature trend, before and after model verification 31
Figure 30 Calibration signature and data comparison at the end of model verification....32
Figure 31 Sensor’s nomenclature scheme....32

1 Executive Summary

Task *T3.2 – Free running modes* is part of *WP3 – Simulation and Optimisation Enablers* and focusses on free-running KPIs and inputs and on the inclusion of extra usage scenarios to the sample dynamic simulation platform developed under T3.1 - *Dynamic simulation architectures* and detailed in D3.1. The sample simulation platform is based on the Python library PREDYCE – Python semi-Realtime Energy DYnamics and Climate Evaluation – that is under-development at POLITO. In particular, D3.2 details extra functionalities devoted to: i.) include free-running input-modification actions and devoted KPI calculations, including ii.) fictitious cooling, and iii.) describe additional scenarios of usage of the PREDYCE tool. The latter includes the performance gap scenarios, considering also semi-automatic verification and an adapted application of the sensitivity analysis scenario for free-running purposes. In line with twin deliverable D3.1, D3.2 is composed by this report describing the adopted methodological approach and introducing PREDYCE extra-functionalities supporting the specific above-mentioned scenarios. Behind this report D3.2 also includes the possibility to run performance gap scenarios. D3.2 objectives and its structure are described in the following Section, while the deliverable also includes sample applications along the text to increase readability.

2 Deliverable objectives and structure

This deliverable describes extra functionalities added to the dynamic simulation platform described in E-DYCE D3.1 to fit the objective of task 3.2 supporting the calculation of climate related KPIs and local free running potential under typical and real weather conditions.

The adopted dynamic simulation platform, which is based on the PREDYCE under-development tool by POLITO – "DYCE" development action –, is expanded adding a new application scenario to allow the comparison between real and simulated building behaviours under the same weather conditions supporting the base identification of performance gaps between standard simulated models and real monitored conditions. Additionally, the task includes in the sample simulation platform different bioclimatic and passive technologies supporting for example passive and ventilative cooling issues. The free-running module is hence conceived as a series of additional inputs and outputs added to the input and output analyser modules described in D3.1. The free running added functionalities allow, for example, to correlate comfort/discomfort conditions into fictitious energy needs to support the integration of bioclimatic and passive technologies for space heating/cooling in the dynamic simulation process. This process allows to define fictitious energy needs to potentially feed a future labelling approach and compare buildings even when one or more of them are working in free-running mode, such as traditional buildings without cooling systems. Hence, D3.2 includes new input actions and new output KPIs – see also the list mentioned in E-DYCE D1.2 (EDYCE, 2021), such as the number of hours in which machine is operating, the calculation of the energy signature, comfort/discomfort conditions adopting both Fanger and adaptive thermal comfort models in line with current CEN and ISO standards (EN ISO, 2005; European Committee for Standardization, 2019). Intuitive graphs are also produced demonstrating the possibility to use output data to feed recognized graphical outputs to analyse building comfort/discomfort.
Several of the above-mentioned functionalities include the usage of an extra PREDYCE module defined to automatically transform monitored weather data into EnergyPlus weather simulation files (EPW) that is developed by POLITO under T3.3 actions in version 'DYCE'v1 to automatically read data collected by the meteorological station installed at TPM.

D3.2 is structured as follows: Section 3, after a short remind of the "DYCE" development action of the PREDYCE tool – see also D3.1 –, it details the E-DYCE free-running correlated input and output analyser actions including devoted KPIs; Section 4 describes the developed protocol and scenario to compare real Vs simulated building behaviours, while Section 5 shortly describes the fictitious cooling calculation. Sample applications are given along the above-mentioned sections to support the module description.

3 Free-running extra functionalities

According to E-DYCE D1.2 (EDYCE, 2021), a building is working in free-running mode when or i.) a system is not installed, such as for example in traditional Mediterranean houses in which any cooling system is installed and/or any heating system is present with the exclusion of fireplaces or personal electrical devices, or ii.) a system is turned off during the period of analysis. A free-running building is hence a condition in which user comfort (thermal comfort or other comfort issues) is correlated to free-floating conditions taking advantages from building envelope ability in mitigating and preventing discomfort conditions acting as boundary between climatic and weather outdoor conditions and indoor spaces – see for example (Ghiaus, 2003). Additional technologies, such as heat gain increasing in winter and/or heat gain prevention (shading), mitigation (thermal mass), and dissipation (e.g., ventilative cooling) are also part of the free-running considered domain enlarging and managing comfort during operating periods without mechanical systems. These passive cooling and heating strategies (or hybrid solutions with minimal mechanical supports, e.g., ventilation by fans) expanding the free-running mode applicability without reaching discomfort are in lines with main recognized climatic and bio-climatic design approaches – see for example zones in the well-known Givoni-Milne bioclimatic chart (Figure 1 sampling technologies on Typical Metereological Years for (a) TPM and (b) Rome-Casaccia – by Climate Consultant (Liggett and Milne, 2017)).

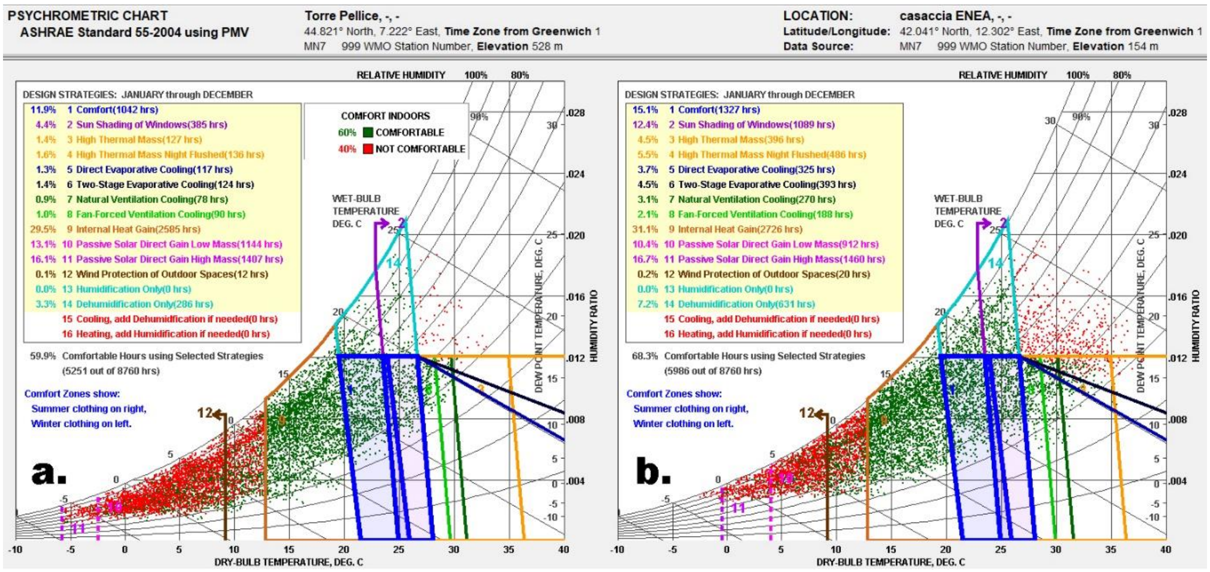


Figure 1 Potential impact of free-running technologies on expected thermal comfort based on TMY of (a) TPM and (b) Rome-Casaccia

Focussing on the specific deliverable contents, in this section they are described main extra functionalities and scenarios of the PREDYCE tool devoted to support free-running strategy simulations for E-DYCE. Additionally, the correlation between free-running model discomfort conditions and substitutive energy needs is introduced. In line with E-DYCE D1.2 and D2.4, this correlating analysis is based on the fictitious cooling/heating approach that is here transposed in a PREDYCE KPI. Additionally, sample applications of the PREDYCE sensitivity analysis scenario are given for free-running functionalities – see Section 4.1.

3.1 The PREDYCE tool

PREDYCE (Python semi-Realtime Energy DYnamics and Climate Evaluation) is the Python library, deeply described in correlated deliverable 3.1, developed inside E-DYCE project to work as an EnergyPlus simulation platform, allowing automatic editing of IDF files (building models) and KPIs computation on both simulation results and monitored data. PREDYCE is built as an ensemble of independent modules, then combined in task-oriented scripts (named scenarios) which can perform complex actions and analysis exploiting all library functionalities. At present, each scenario works as an independent tool application, executable both remotely through a REST API and locally by command line.

PREDYCE main modules are IDF editor, KPIs calculator and runner. Moreover, additional modules are present to address specific tasks, like EPW file managing. Main inputs to all PREDYCE scenarios are IDF model, EPW file, eventual CSV of monitored data and a JSON file structured with keywords allowing to recall different tool functionalities and customize the parametric request. Outputs instead are CSV files containing aggregated KPIs for the time period and timeseries results. Additionally, plots are generated in simulation-based subfolders in order to deepen KPIs analysis. More details on PREDYCE structure and input/outputs workflow can be found in deliverable D3.1, while following sections will be focused on PREDYCE applications to buildings working in free-running mode.

3.1.1 EPW compiler module*

**This sub-section also includes results of POLITO efforts in T3.3.*

To compare actual and standard building behaviour, it is necessary to be able to simulate the building model under past monitored weather conditions. Consequently, an EPW compiler module able to generate an EPW from weather station downloaded data has been developed as additional PREDYCE functionality. The module is organized in independent steps to be more flexible to any update and change. The steps followed inside the module are:

1. Generate a CSV file containing raw data from weather station, updated week by week or each time desired (it could be also organized to be updated in real time). Data are saved per minute ignoring seconds resolution to save storing space and avoiding too many missing values. This step is dependent from the particular weather station from which data are downloaded and more generally to the data source being possible to adapt the procedure to include data from services, e.g., Weather Underground[1]. Being an independent step, it can be each time organized differently in accordance with specific weather station API.
2. Generate a CSV file of clean data: each type of data is cleaned according to specific rules derived looking at timeseries and weather station manual.
3. Generate a CSV file ready for EPW filling by hourly aggregating and computing missing required fields through formulas from other data – e.g., splitting the global horizontal irradiation into its components – or building them according to specific rules (for coded fields like *Present Weather Codes*).
4. Generate EPW file from available data and further fill it to cover entire year with TMY integration (always needed also for short simulations) or implementing other filling strategies. This step is based on Python pyepw[2] library.

3.1.2 PREDYCE scenarios

Three main scenarios have been developed to answer E-DYCE project goals, in accordance with other WPs outcomes and needs. Particularly, the base PREDYCE scenario is named sensitivity analysis and allows to perform parametric simulations automatically acting on many buildings' characteristics and computing KPIs according to European standards and norms and newly defined indicators – see also E-DYCE deliverables D1.2 and D2.4. In the following an example of sensitivity analysis applied to free-running optimization is proposed, while in deliverable D3.1 an application to retrofit solutions is reported. Moreover, more complex scenarios have been developed, all with bases on sensitivity analysis, in order to reach different project objectives: the performance gap scenario is in charge of evaluating the

[1] https://www.wunderground.com

[2] https://github.com/rbuffat/pyepw

difference between actual and standard building behaviour, exploiting PREDYCE ability to work both on simulation results and monitored data, while the model verification scenario can be used to speed up and simplify building models adjustments to real measured data, which is a necessary step before computing a performance gap.

3.2 *PREDYCE free-running correlated actions*

In deliverable 3.1, both PREDYCE IDF editing actions and computable KPIs are listed in a complete form and deeply analysed. Here they are reported actions and KPIs strictly related to buildings free-running mode, highlighting their peculiarities and focusing on use cases.

3.2.1 IDF editor

In addition to main IDF editing actions working on building envelope characteristics, e.g., changing thermal resistances, thermal heat capacities and thermal masses, solar gains, some additional bioclimatic and free-running technologies are included focussing mainly on passive cooling strategies. In particular, PREDYCE includes shading and natural ventilation IDF editing actions that allow respectively to prevent heat gains and dissipate them via ventilative cooling means. Concerning shadings, several functions are available in order to add different kinds of blinds to windows, eventually filtered by orientation, or to change the slat angle. A specific schedule name can be associated to shading objects, providing different kinds of control strategies. Some basic strategies are included with predefined names inside EnergyPlus software, for example based on temperature and solar radiation thresholds. These thresholds are simple values associated to specific IDF fields, so they can be modified through generic functions, i.e., *set_object_params,* that require the user to know values position in the IDF. If needed more name-friendly functions can be developed under request during next project and development stages to cover these kinds of actions.

predyce.idf_editor.add_blind(*idf, blind_data, type='interior', filter_by='', filter_by_orientation=None, schedule=None*)

Add interior or exterior blind to all windows.

Parameters

- **idf** (class:*predyce.IDF_class.IDF*) – IDF object
- **blind_data** (*dict*) – WINDOWMATERIAL:BLIND” JSON data
- **type** (*str, optional*) – Type of blind (interior or exterior), defaults to “interior”
- **filter_by** (*str, optional*) – Filter zone by name. Can be the block name or specific zone name, defaults to “”
- **filter_by_orientation** (*int, optional*) – Specify the orientation of the windows to which the blind has to be added. Possible values are 0, 45, 90, 135, 180, 225, 270, 315, 360 (the windows orientation in the building are converted to the nearest of these values for simplicity), defaults to None
- **schedule** (*dict, optional*) – If specified blinds will follow the given schedule, otherwise they are supposed to be always on, defaults to None

Example

```
add_blind(idf,
          blind_data={
              "Name": "ext_blind",
              "Slat Width": 0.025,
              "Slat Separation": 0.01875,
              "Slat Thickness": 0.001,
              "Slat Conductivity": 0.9,
              "Front Side Slat Beam Solar Reflectance": 0.8,
              "Back Side Slat Beam Solar Reflectance": 0.8,
              "Front Side Slat Diffuse Solar Reflectance": 0.8,
              "Back Side Slat Diffuse Solar Reflectance": 0.8,
              "Slat Beam Visible Transmittance": 0,
              "Front Side Slat Beam Visible Reflectance": 0.8,
              "Back Side Slat Beam Visible Reflectance": 0.8,
              "Front Side Slat Diffuse Visible Reflectance": 0.8,
              "Back Side Slat Diffuse Visible Reflectance": 0.8,
              "Blind to Glass Distance": 0.015,
              "Blind Bottom Opening Multiplier": 0.5
          },
          type="exterior",
          filter_by="main_block"
          )
```

predyce.idf_editor.set_object_params(*idf, obj_str, data=None, **fields*)

Set parameters on IDF objects given a string representing the object.

Parameters

- **idf** (class:*predyce.IDF_class.IDF*) – IDF object
- **obj_str** (*str*) – String representing the object; the typology and the name are separated by a comma. For example "material,Timber Flooring" will edit the idfobjects["MATERIAL"] whose name is Timber Flooring; a string like "material" will edit all materials regardless of name.
- **data** (*dict, optional*) – dictionary where each key corresponds to a pararameter to be changed in the IDF and each value corresponds to the new value.
- **fields** – Same as *data* parameter but fields are passed as keyword arguments, e.g. Conductivity=0.3 which is equivalent to {"Conductivity": 0.3} for *data* field.

Example

```
set_object_params(idf, "material,Timber Flooring", data={"Conductivity": 0.3})
set_object_params(idf, "material,Timber Flooring", Conductivity=0.3)
```

Other functions are devoted to handle shading objects (overhangs and fins) and to change windows opening factor.

predyce.idf_editor.add_overhangs_simple(*idf, extension=1, tilt=90, shift=0.04*)

Add simple overhang.

Parameters

- **idf** (class:*predyce.IDF_class.IDF*) – IDF object
- **extension** (*int, optional*) – Extension of the overhang, defaults to 1
- **tilt** (*int, optional*) – Tilt of the overhang, defaults to 90
- **shift** (*float, optional*) – Shift of the overhang, defaults to 0.04

predyce.idf_editor.add_fin_simple(*idf, extension=1, shift=0.04*)

Add simple fin.

Parameters

- **idf** (class:*predyce.IDF_class.IDF*) – IDF object
- **extension** (*int, optional*) – Extension of the fin, defaults to 1
- **shift** (*float, optional*) – Shift of the fin from the window edges, defaults to 0.04

predyce.idf_editor.change_opening_factor(*idf, value*)

Change opening factor.

Parameters

- **idf** (class:*predyce.IDF_class.IDF*) – IDF object
- **value** (*float or str*) – New opening factor value

Moreover, scheduled natural ventilation objects can be added in the IDF, associating specific schedules and ACH (air exchanges for hour) values to each zone. Both ventilation and infiltration ACH can be changed in accordance with zone volume or as a fixed value for all thermal zones.

predyce.idf_editor.add_scheduled_nat_vent(*idf, filter_by='', schedule=None, ach=None, min_ind_temp=None, max_ind_temp=None, delta_temp=None, params={}*)

Add scheduled natural ventilation to desired zones.

Calculation method is supposed to be the IDD default one, Flow/zone.

Parameters

- **idf** (class:*predyce.IDF_class.IDF*) – IDF object
- **filter_by** (*str, optional*) – Filter zone by name. Can be the block name or specific zone name, defaults to “”
- **schedule** (*str or dict, optional*) – Name of the Schedule:Compact construction object to be added in IDF or the Schedule:Compact object in a dictionary format, defaults to None
- **ach** (*float, optional*) – Value of air changes per hour, defaults to None
- **min_ind_temp** (*float, optional*) – Minimum indoor temperature setpoint, defaults to None
- **max_ind_temp** (*float, optional*) – Maximum indoor temperature setpoint, defaults to None
- **delta_temp** (*float, optional*) – Delta temperature for activation, defaults to None
- **params** (*dict, optional*) – Additional scheduled natural ventilation parameters for the ZoneVentilation:DesignFlowRate object, defaults to {}

Other useful functions for free-running analysis are devoted to activate/de-activate HVAC systems, so that comparisons with symmetrical building behaviour can be performed (e.g., for fictitious cooling computation as described in the following devoted sub-section).

predyce.idf_editor.deactivate_cooling(*idf, cool_sch_name='Cooling SP Sch', cool_avail_name='Cooling Availability Sch', temp=None*)

Deactivate cooling on already existing HVAC system.

Parameters

- **idf** (class:*predyce.IDF_class.IDF*) – IDF object
- **cool_sch_name** (*str, optional*) – Name of the cooling setpoint schedule compact which has to be replaced, defaults to “Cooling SP Sch”
- **cool_avail_name** (*str, optional*) – Name of the cooling availability schedule compact which has to be replaced, defaults to “Cooling Availability Sch”
- **temp** (*int or float, optional*) – New value for maximum outdoor temperature. If None, no changes are applied to such parameter, defaults to None

3.2.2 KPIs

Main KPIs computed inside PREDYCE are reported in deliverable D3.1. Among them, distribution of hours in Adaptive Comfort Model (ACM) categories is one of the most important for free-running analysis: the number of hours in thermal comfort (cat. I and II) and the number of hours in discomfort (cat. III or higher) can give an initial vision of building free-running potentialities under standard conditions. Categories and calculation approaches are in line with EU and international standards and users may require both to calculate them according to current EN 16798-1:2019 (European Committee for Standardization, 2019) or previous EN 15251:2007 (European Committee for Standardization, 2007). However, ACM works with operative temperatures, which are very complex to be elaborated by monitored data in real demo buildings considering the lack of mean radiant temperature (and air velocity) data – that are also correlated to the D3.1 mentioned Fanger PMV/PDD comfort model. Consequently, ACM results to be not easily applicable to real data and to scenarios as the performance gap devoted one to return behavioural differences also in the free-running mode. In order to solve this issue in case of ACM application to real data (and to simulation used for comparison) indoor mean air temperature is used instead of operative temperature by passing the lack in monitored data concerning globe temperatures – to define mean radiant temperatures – and internal air velocities at body level. This choice is motivated by actual feasibility but is also supported by specific analyses based on monitored data elaboration in the ENEA living lab building – see Section 3.2.3.

Concerning evaluation of buildings free-running potential, another useful KPI is the *n_h_kwh* which is devoted to compute the number of hours in which heating and cooling systems are on and consuming more than given thresholds. At present, COP is here definable as a fixed seasonal number (such DesignBuilder interface does for simple HVAC models and in line with many other energy simulation software), but it can be in the future integrated with hourly definable functions considering the different losses mechanisms, by integrating, for example de ones described in E-DYCE deliverable D3.1.

adaptive_comfort_model(*df*, *eu_norm='16798-1:2019'*, *alpha=0.8*, *filter_by_occupancy=0*, *when={}*)

Compute adaptive comfort model in a standardized format.

Parameters

- **df** (class:*pandas.core.frame.DataFrame*) – dataframe should contain “Date/Time” column in format ‘year/month/day hour:minutes:seconds’, “T_db_o[C]” preferably with a subhourly timestep and “T_op_i[C]”. Optional “Occupancy” column accepting only 0/1 values.
- **eu_norm** (*str, optional*) – It can be set to ‘15251:2007’ if old UE norm computation is desired, defaults to ‘16798-1:2019’.
- **alpha** (*float, optional*) – With old UE norm ‘15251:2007 alpha is a free parameter in range [0,1), defaults to 0.8
- **filter_by_occupancy** (*int, optional*) – It can be set 0 or 1, depending on wether activate occupancy filtering on thermal comfort KPIs computation or not, default 0.
- **when** (*dict, optional*) – dictionary with ‘start’ and ‘end’ keys and values in format ‘year/month/day hour:minutes:seconds’

Returns

Number of hours in each of the 7 comfort categories and POR computed as % of hours outside cat 2 boundaries.

Return type

dict

n_h_kwh(*df*, *th_list=[0, 0.6, 1]*, *cop_cool=3*, *cop_heat=0.3*)

Number of hours HVAC consumption is higher then given thresholds.

Parameters

- **df** (class:*pandas.core.frame.DataFrame*) – Dataframe which must contain at least cooling and heating consumption columns “Q_c[Wh/m2]” and “Q_h[Wh/m2]” plus “Date/Time” column.
- **th_list** (*list, optional*) – List of thresholds to be compared with consumption, defaults to [0,0.6,1]
- **cop_cool** (*int, optional*) – Cop value for cooling system, defaults to 3
- **cop_heat** (*float, optional*) – Cop value for heating system, defaults to 0.3

Returns

Dictionary containing No. hours above defined thresholds for heating and cooling.

Return type

dict

3.2.3 On the relation between air and operative temperature in standard office rooms

The determination of thermal comfort depends on parameters related to the human body and to the physical quantities of the local climate. Relevant studies carried out on the thermal controlled environment by Fanger, which led to definition of the Predicted Mean Vote and the Predicted Percentage of Dissatisfied, and on microclimate design criteria by Givoni and Olgyai showed the dependence of thermal comfort on several physical parameters. Current relevant standards on thermal comfort in free floating conditions, however, identifies the operative temperature as the main driving parameter. The latter is determined as the average of the air and mean radiant temperatures in a given space, in case of compact volume forms it can be approximated as the arithmetic average of the two parameters. In calculation tools implementing Adaptive Comfort Models (ACM)its determination is not complex, but it is for monitoring in real applications.
To calculate the operative temperature (T_{op}) it is necessary to determine the mean radiant temperature (T_{mr}) that can be calculated with the following equation:

$$T_{mr} = \left[\left(T_g + 273\right)^4 + 2.5 \cdot 10^8 \cdot v_a^{0.6}\left(T_g - T_a\right)\right]^{\frac{1}{4}} - 273$$

Being:
T_g globe temperature[°C]
T_a air temperature [°C]
v_a air velocity [m/s]

Hence such measure may have a relevant impact on cost and room integration, because of space to be allocated for the structure on desk or floor stands (the globe is a hollow black painted metal sphere with a minimum diameter of 150mm). It is thus relevant to understand whether and when the determination of the operative temperature is necessary or when it could be replaced by simpler measurement. The mean radiant temperature is critical in case of strong radiant asymmetries, as in buildings with large glazed area on in case of heating/cooling radiant systems, but in most cases it might be not. A monitoring campaign was carried out in an office room of the ENEA F40 living lab to address this issue in June 2020 and published in (De Lia et al., 2022).

The test was carried in an office room, 3.9m large, 4.5m deep, and 3.1m high. It has a single external west-oriented façade with a 3.2 m^2 window, the transparent area is 2.13 m^2 and the frame fraction is 33%. A microclimatic station by TESTO was mounted close to the centre of the room, in a way to minimize the risk to have the sensor hit by the direct solar rays in late afternoon. The monitoring station acquires air relative humidity and CO_2 concentration, beside the above-mentioned parameters. The room was unoccupied and in free running mode during the test period – see Figure 2 and Figure 3.

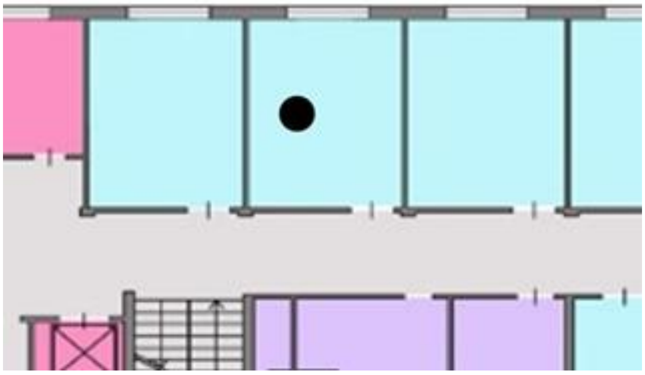


Figure 2 Close-up of the ENEA Living Lab lay-out with the position of the monitoring station

Figure 3 View of ENEA Living Lab test room

The test was carried out for 14 days in June, data were acquired every 10 minutes. The operative temperature was calculated starting from measured air temperature and mean radiant temperature, derived from the above-mentioned quantities. Next figure presents the time evolution of operative and air temperature during the period.

The results – see also Figure 4 – show that the differences between the two quantities were below 0.2°C in 77% of the period, conversely the were above 0.3 and 0.4°C in 4.7% and 0.6 of the readings, respectively. No case was registered with temperature difference above 0.5; this means that such difference was always within the instrument error (0.5°C for air temperature), as declared by the manufactured.

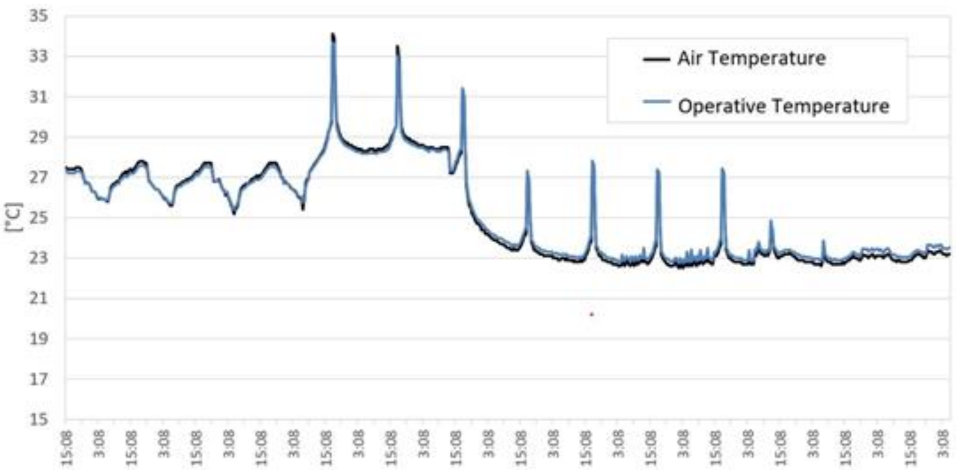


Figure 4 Operative and air temperature trend in the test room during the monitoring period

This finding proves that the air temperature can replace the operative temperature with sufficient accuracy in a built environment without relevant thermal radiation sources, or rather in spaces where the occupants are not exposed to the latter. This aspect is relevant for a proper monitoring of buildings in thermal free-floating conditions without recurring to complex and expansive field sensors.

3.2.4 Fictitious cooling KPI

Defined the background of the free-running mode, a methodology is defined to evaluate building conditions under free-running mode for energy evaluation purposes by translating potential discomfort conditions into substitutive energy needs, as if a mechanical system would have been used to turn discomfort hours into comfort one. The proposed approach defines fictitious cooling and fictitious heating energy needs on the base of simulated (or monitored) thermal discomfort intensities. Such as introduced in E-DYCE **D1.2**, fictitious cooling or heating is based on the definition of a fictitious activation risk indicator that, on the base of the discomfort intensity, defines a percentage of “virtual” usage of a hypothetical mechanical system fictitiously installed in the building. The approach modified the original suggestions of Annex D, ISO TR EN 52018-2:2017 (CEN ISO/TR, 2017). The retrieved fictitious cooling/heating net energy needs may be processed in a second elaboration step to retrieve energy consumptions and primary energy values by applying fictitious sub-systems COPs and EP conversion factors in line with mechanically driven buildings. For the E-DYCE sample simulation platform, the coding functions developed for mechanically-driven simulated buildings – see D3.1 and/or current EU and national regulations, e.g., the Italian UNI-TS 11300 series – can be applied also for calculating the fictious COPs and the fictitious EPs when fictitious net energies are defined.
Similarly, it is possible to define fictitious ventilation energy needs and consumptions when IAQ levels overpass comfort thresholds or to apply the concept to visual comfort although these latter aspects are not here detailed.

Figure 5 shows the computation workflow for fictitious cooling KPI. Differently from other KPIs, it requires the run of a parallel simulation simulating the same building (un-equipped with HVAC system) as if the cooling system was present and natural ventilation de-activated (windows closed). This surely extends simulation time if this KPI is among the ones requested in the input JSON. Results of the two simulations are both stored in order to be further compared: if real building model is in thermal comfort (ACM cat. I) in a certain hour its cooling need is 0, while if it is in thermal discomfort (ACM cat. III) its cooling need is considered to be the full one of its symmetric buildings with HVAC on. If instead a certain hour falls in ACM cat. II, the cooling need is considered to be a linear function of the cooling consumption in the symmetric building. The same process can be replicated for fictitious heating computation, considering lower ACM categories. At present, the net cooling need is considered, without applying COP. However, it can be changed in future updates to both seasonal or more dynamic approach.

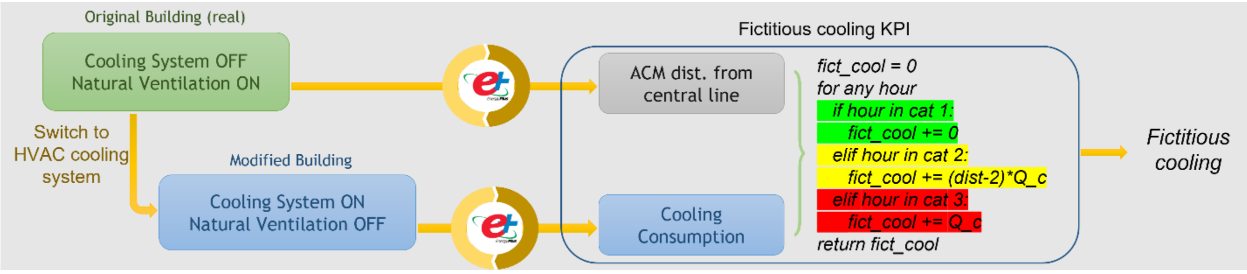


Figure 5 Fictitious cooling KPI computation workflow

Here below is reported the part of the PREDYCE technical manual describing the fictitious cooling KPI.

fict_cool(*dataframe_off, dataframe_on, eu_norm='16798-1:2019', alpha=0.8*)

Return fictitious cooling computed as the amount of cooling consumption needed in a building without mechanical system in order to reach category I comfort of adaptive comfort model.

Parameters

- **dataframe_off** (class:*pandas.core.frame.DataFrame*) – dataframe containing at least "Date/Time", "T_op_i[C]", "T_db_o[C]" columns in order to compute adaptive comfort model categories.
- **dataframe_on** (class:*pandas.core.frame.DataFrame*) – dataframe containing at least "Date/Time" and "Q_c[kWh/m2]" columns, of the same building with HVAC installad and ACH ventilation set to 0.
- **eu_norm** (*str, optional*) – It can be set to '15251:2007' if old UE norm computation is desired, defaults to '16798-1:2019'.
- **alpha** (*float, optional*) – With old UE norm '15251:2007 alpha is a free parameter in range [0,1), defaults to 0.8

Returns

fictitious cooling total [kWh/m2]

Return type

float

Figure 6 shows cumulative distribution functions of simulated cooling needs (both real and fictitious) and indoor operative temperatures, for a Torre Pellice demo case (as IDF building model), hypothetically located in Casaccia, Rome (concerning TMY EPW file), because of the hotter climate. The example highlights that fictitious cooling is limited with respect to the case with HVAC system on with special regards to hours in which medium-to-low cooling loads are expected. This because for those hours the free-running behaviour of the building is sufficient to guarantee internal comfort conditions considering natural ventilation and adaptive thermal comfort scenarios, differently when internal temperature overpass ACM upper categories thresholds, also fictitious cooling will go to full charge since it is not possible to guarantee indoor comfort conditions via the considered free-running schemes.

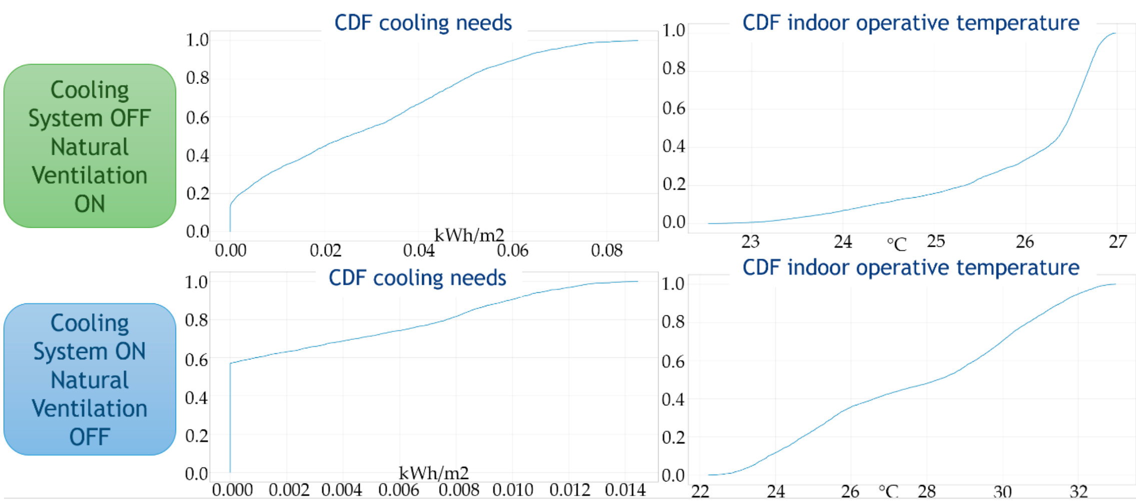


Figure 6 CDFs describing fictitious cooling potentialities in describing cooling needs

Significance of the fictitious cooling/heating KPI

An initial series of analyses is here reported to give initial consistency to the significance of the fictitious cooling/heating approach. The proposed methodology bases on two initial considerations: i.) it is possible to retrieve a correlation between climate/weather conditions and space energy needs in mechanically conditioned buildings; ii.) it is possible to retrieve a parallel correlation between climate/weather conditions and space thermal discomforts in free-running buildings by analysing the free-floating space temperature – see for example Figure 7. Both considerations are well-known and detailed in literature. It is, in fact, well-recognized that heating and cooling (with slight limitations for sunny sites) energy needs may be linearly correlated with local HDD (heating-degree days) and CDD (cooling-degree days) values. The same consideration is also supported by MS energy regulations. For example, in Italy, the heating season duration for design and standard energy rating is based on the municipality climate zone which is derived from local HDD values – see UNI/TS 11300-1 and DPR 74/2013. Similarly, it is possible, for example, to mention the studies on building base-temperature supporting the definition of the operational specific-building correlated heating and cooling seasons (CISBE, 2006). Furthermore, direct correlations between external temperatures or degree-days/hours (or delta temperatures) are also underlined for free-running and natural-driven buildings (Chiesa et al., 2021b; Cook, 1989; Givoni, 1994) and are at the base of different approaches to define the local potential of free-running technologies, e.g., sunspaces (Chiesa et al., 2017), or passive cooling means (Artmann et al., 2007; Chiesa and Grosso, 2015; Chiesa and Zajch, 2020; Santamouris and Asimakopolous, 1996).

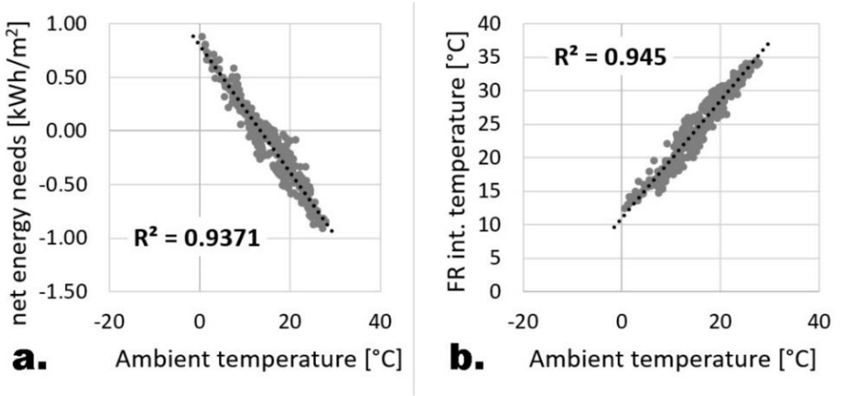


Figure 7 Correlation between external conditions (temperature) and (a) energy needs, (b) free-running building temperatures – daily aggregated values

(Parallelepiped building 10x10x3.5m, 30% WWR all orientations, Rome-Ciampino TMY, U-value$_{wall}$ 1.8)

Given these correlations between climate and building conditions in both modes, it is possible to expect that the same building running in mechanical mode and in free-running mode will retrieve respectively energy needs and thermal comfort conditions – expressed in term of free-running internal temperatures – that can be correlated one with the other. A sample parallelepiped building – shape: 10x10x3.5m, 30% WWR (window-to-wall) for all orientation, base orientation angle 0° – was assumed and simulated under the same weather in both mechanical and free-running modes. Initial results for Rome-Ciampino climate (EnergyPlus TMY[3]) confirm the initial considerations (Figure 7), while Figure 8 plots internal temperatures (free-running mode) Vs energy needs (mechanical mode) showing a clear correlation between results of the two modes – daily aggregation: R^2_{linear} = 0.98; $R^2_{polynomial}$ = 0.99; hourly aggregation: R^2_{linear} = 0.87; $R^2_{polynomial}$ = 0.94. The figure shows that a clear correlation exists between the building simulated under mechanical mode and free-running mode, and that it is possible to retrieve by regressions one by the other with limited errors, especially at daily aggregated data resolution. For the hourly resolution it can be underlined how several hours show no cooling or heating needs and how the same are mainly distributed in those hours in which the parallel free-running model is showing temperatures in the heating and cooling set point range (20 °C to 26 °C).

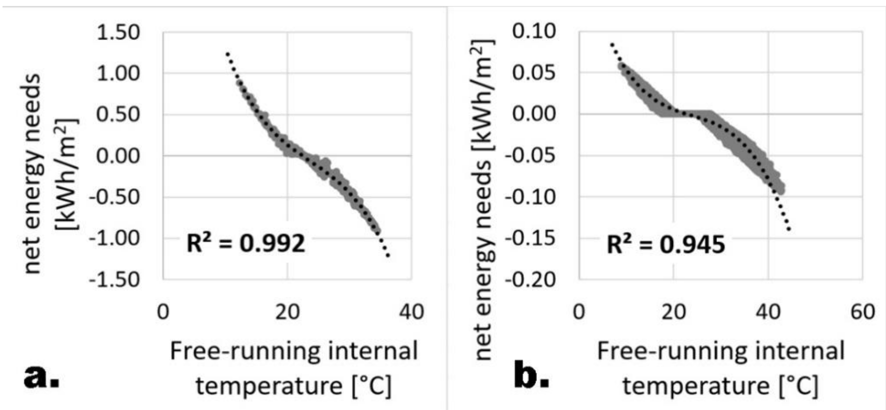


Figure 8 Correlations between free-running internal temperatures and energy needs, for (a) daily aggregated values and (b) hourly aggregation

[3] see also EnergyPlus weather data, available online at: https://energyplus.net/weather (last view 18th Jan 2022)

Secondly, changes in building boundary conditions are tested to analyse the initial robustness of the given approach by introducing: i.) ventilative cooling (only FR, both cases - indoor/outdoor temperature control activation, scheduled ACH 5); ii.) a change in U-value (from 1.8 to 0.2 W/m^2; iii.) a change in climate, by virtually moving the sample testing building to a different location and varying the TMY (from Rome-Ciampino to TPM climate); iv.) a change in the scheduling profile (from a residential 24h/7d to an office 8-18/Mon-Fri) considering occupancy (FR and mechanical models), heating and cooling (mechanical model only). Cases i.) is shown in Figure 9 underlining that for both cases, the first in which ventilation is added only for the FR model and the second in which ventilation is added to both the FR and the mechanical models, very high correlations are retrieved between FR temperatures and net energy needs. Similarly, Figure 10 reports results for the variations ii.) and iii.). The figure shows that a very high correlation arrives under different U-value conditions and that the same approach is valid also for different climatic conditions.

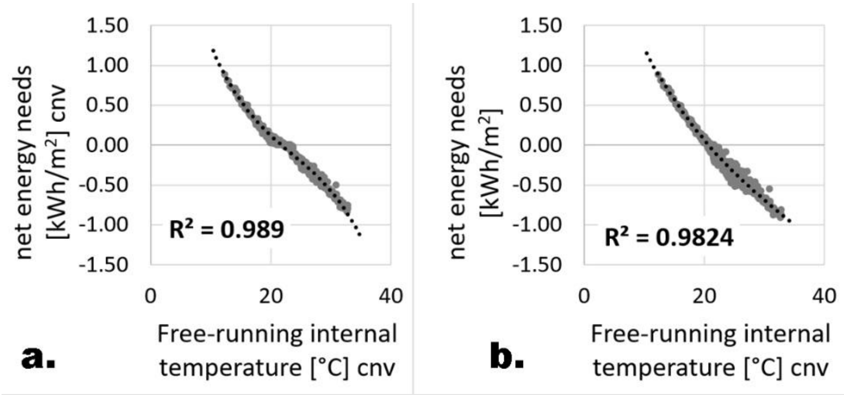


Figure 9 Correlations between free-running internal temperatures and energy needs, for (a) case with ventilative cooling on both FR and mechanical modes and (b) case with ventilative cooling activated only for the FR mode – daily aggregation

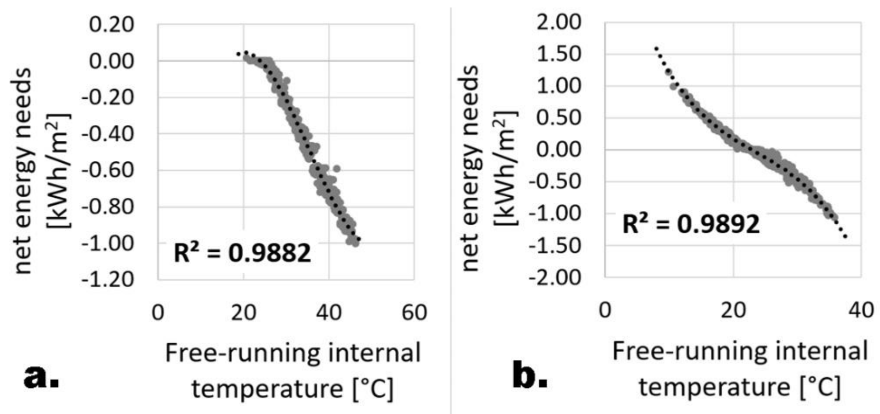


Figure 10 Correlations between free-running internal temperatures and energy needs, for (a) case with reduced U-value (0.2 W/m2K) and (b) original case run under TPM typical weather conditions – daily aggregation.

Considering hourly aggregations, similar results are retrieved – see for example the result for case i.) and ii.) in Figure 11. A slight increase in the standard deviation is underlined although the general correlation is still very evident also for hourly data aggregations.

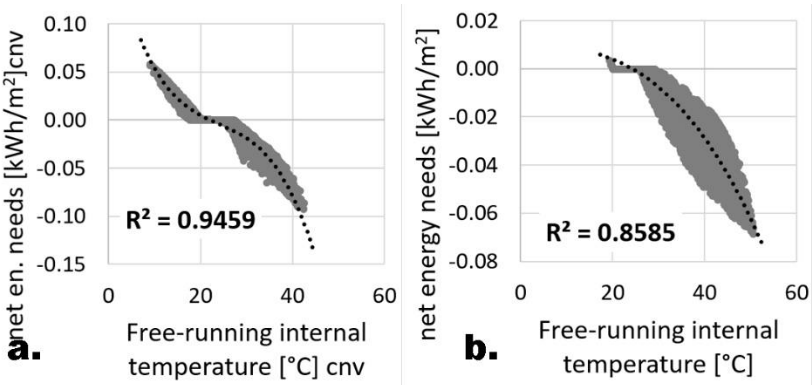


Figure 11 Correlations between free-running internal temperatures and energy needs, for (a) case with ventilative cooling on both FR and mechanical modes and (b) original case with reduced U-value (0.2 W/m2K) – daily aggregation

Finally, the effect of changes in the scheduling profiles are reported in Figures 12(a) and (b) showing for hourly values that the correlation is high (R^2=0.79) when results are filtered for occupied hours and reaches a R^2 of 0.95 if data are additionally filtered in an 8 to 17 interval cutting turning-off and -on hours. Nevertheless, in cases in which all hours (8760 h) are considered, correlation is influenced by many hours in which energy needs are null, while FR temperatures variates (R^2=0.44). Considering that thermal comfort is generally retrieved for occupied hours, especially in office buildings, the filtered approach is coherent with the proposed methodology. Similarly, Figures 12(c) and (d) show the same distribution for daily aggregated data. In these latter cases, filtering by working day (but including in daily average all hours) the correlation reached a R^2=0.98 (c), while all day correlation is characterized by a R^2=0.66 due to weekends (no energy needs) (d).

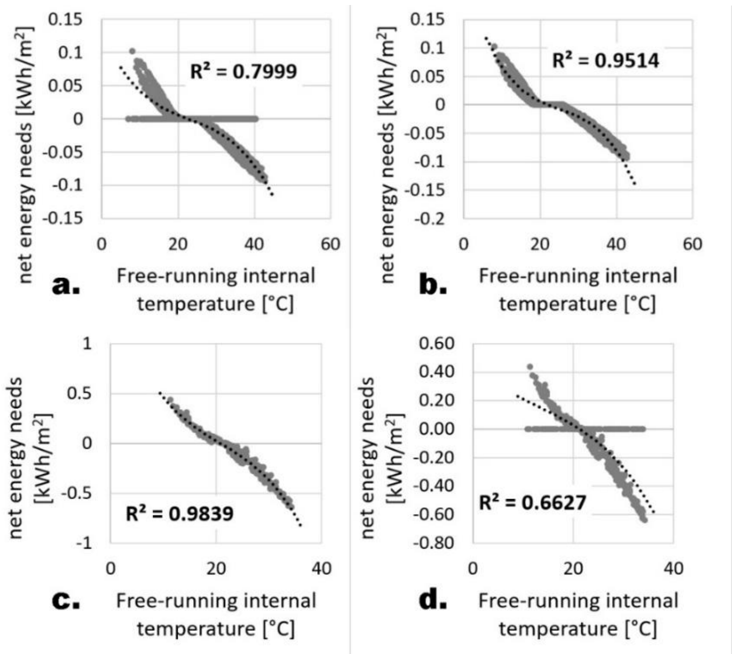


Figure 12 Correlations between free-running internal temperatures and energy needs, office scheduled cases, considering only occupied hours (hourly aggregation) (a) occupied hours without turning off and on periods (8-17) (b); only occupied day (daily aggregation

Even if additional tests will be performed during next project advancements by applying the verification to E-DYCE demo cases and including extra variations in terms of parameters and ranges, initial checks show how fictitious KPIs may transpose free-running cases to energy evaluating processes on the base of a recognized background.

4 PREDYCE scenarios

4.1 Sensitivity analysis

Here is reported an example of sensitivity analysis applied to suggestions of optimal free-running mode for buildings during the summer season – see also the dissemination paper prepared for the E-DYCE WP3 MS-05 (Chiesa et al., 2021a). The proposed example refers to a TPM (Torre Pellice Municipality) residential demo case, simulated under local TMY (Typical Meteorological Year) and under Casaccia (Rome) hotter climate, in order to give an example of both mountain colder conditions and hotter climate typical of middle Italy. Different KPIs (e.g., thermal comfort) are computed in free-running mode (the demo case actually is not provided with cooling system). As it can be seen in input JSON file, figure 13, the following actions are preliminary executed on the building model: set simulation period to June-September and add shade rolls with a base shading control strategy (based on outdoor temperature and solar radiation). The impact of different free-running technologies is then evaluated through parametric analysis acting on: ACH ventilation scheduled values, external temperature and global solar radiation (GHI) thresholds for shade rolls activation and overhang length. The computed KPIs are: adaptive thermal comfort (% hours in categories), mean ACM discomfort expressed in POR form (% of hours outside comfort Cat. II boundaries), and PMV-PPD model-based POR (% hours with PMV > 0.7, so PPD > 15%, comfort Cat. III EN 16798-1:2019).

```
"scenario": "sensitivity_analysis",
"building_name": "r0",
"preliminary_actions": {
    "change_runperiod": {"start": "01-06", "end": "30-09", "fmt": "%d-%m"},
    "set_shading_control": {"name": "1003"},
    "set_object_params": {
        "obj_str": "WindowProperty:ShadingControl,1003",
        "data": {
            "Shading Control Type": "OnIfHighOutdoorAirTempAndHighHorizontalSolar",
            "Schedule Name": ""}}},
"actions": {
    "change_ach": {"ach": [0, 2.5, 5]},
    "add_overhangs_simple": {"extension": [ 0.2, 0.4, 0.6, 0.8, 1]},
    "set_object_params": {
        "obj_str": ["WindowProperty:ShadingControl,1003"],
        "data": [
            {"Setpoint": 20, "Setpoint 2": 100},
            {"Setpoint": 20, "Setpoint 2": 150},
            {...}]}},
"outputs": [],
"kpi": {"pmv_ppd": {"clo": 0.7},
    "adaptive_comfort_model": {},
    "adaptive_residuals": {}}
```

Figure 13 Input JSON file used for this sample application

In the following plots generated for post-analysis are reported (so, at present not automatically generated inside PREDYCE, but from results contained in *data_res.csv*). In this example, heatmaps results in figures 14 and 15 show that in Torre Pellice best strategies are meant to keep house warmer letting more sunlight entering at any outdoor temperature, while in Rome increasing ventilation and decreasing outdoor temperature threshold have the greatest impact, despite GHI and overhang, and best shading strategies are reached activating rolls at low outdoor temperature and GHI (20/22 °C and 100/150 W/m^2).

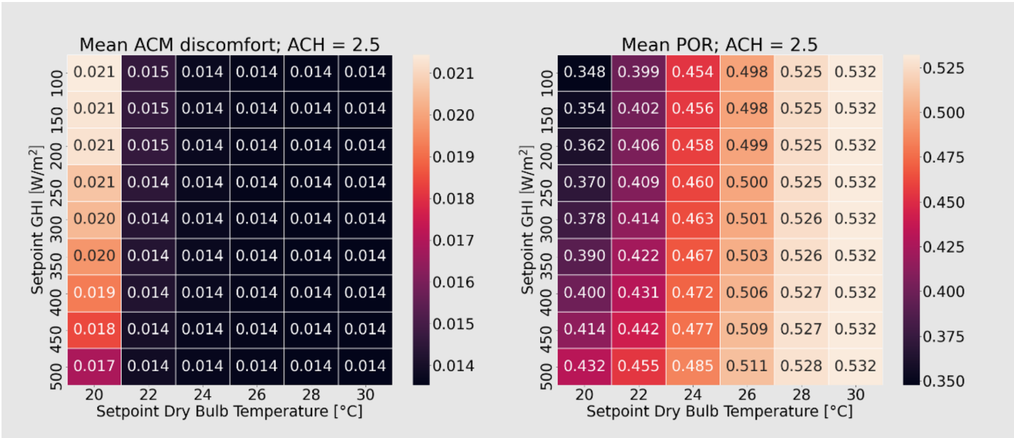


Figure 14 TPM shading control thresholds heatmaps

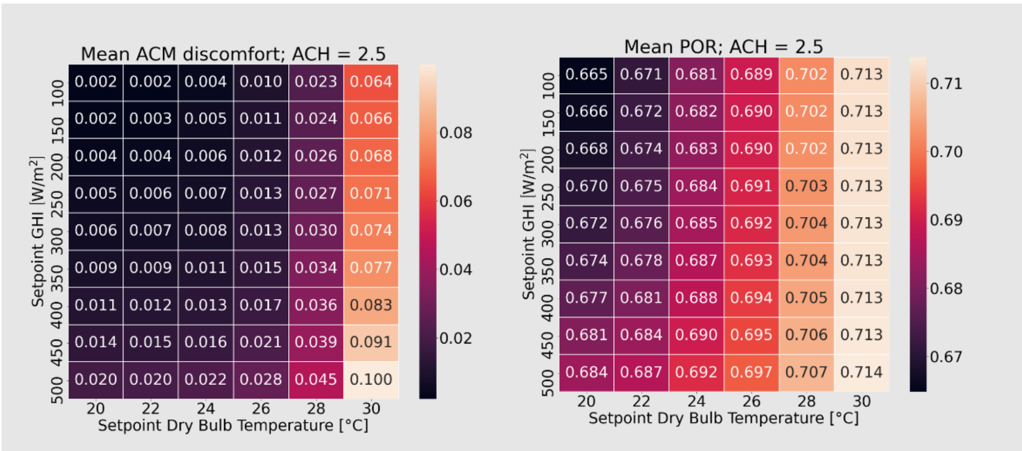


Figure 15 Rome shading control thresholds heatmaps

Figures 16 and 17 instead allow to give an initial idea of the impact that ventilation ACH has on thermal comfort: particularly in Rome, setting an ACH of 2.5 instead of 0, allows to drastically reduce discomfort, reducing also the positive impact of an added overhang. In the colder TPM climate instead, increased ventilation is unnecessary, and can even result to be disadvantageous in terms of thermal comfort to avoid overcooling phenomena.

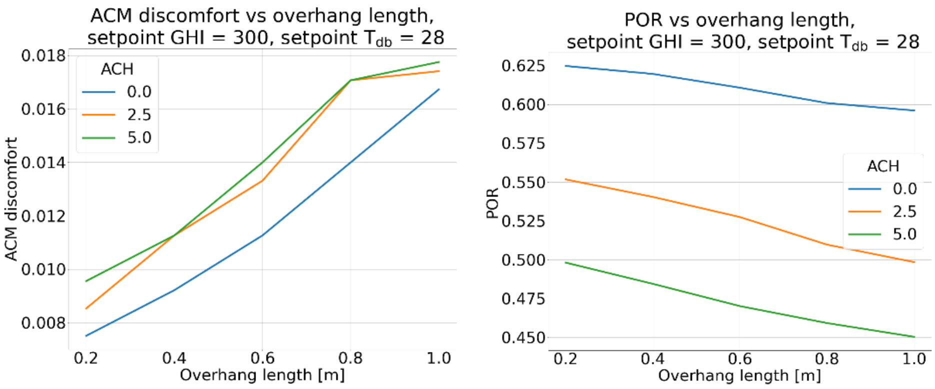


Figure 16 TPM overhang length and ACH impact on thermal discomfort

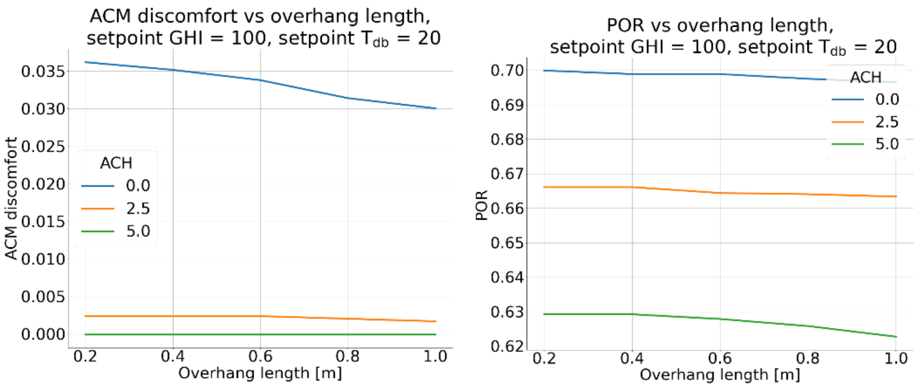


Figure 17 Rome overhang length and ACH impact on thermal discomfort

Figures 18 and 19 report instead the ACM distribution plots generated inside PREDYCE for respectively TPM (Fig. 18) and Rome (Fig. 19) climates showing both initial model conditions and best retrieved free-running summer mode. Figures show the ability of the PREDYCE tool in evaluating free-running approaches and to suggest design actions (including retrofitting) and potentially design operational conditions by also considering free-running technologies, e.g., integrating movable shading systems and ventilative cooling to balance solar heat gains. In the colder climate of TPM the sensitivity simulation scenario allows to suggest shading activation thresholds and ACH to avoid overcooling effects balancing summer building behaviour to reach thermal comfort under free-running mode. Differently, the Rome sample application shows how PREDYCE may be adopted to suggest balancing between shading activation thresholds and ventilative cooling rates to avoid overheating in a hot climate allowing to reach thermal comfort conditions under free-running mode.

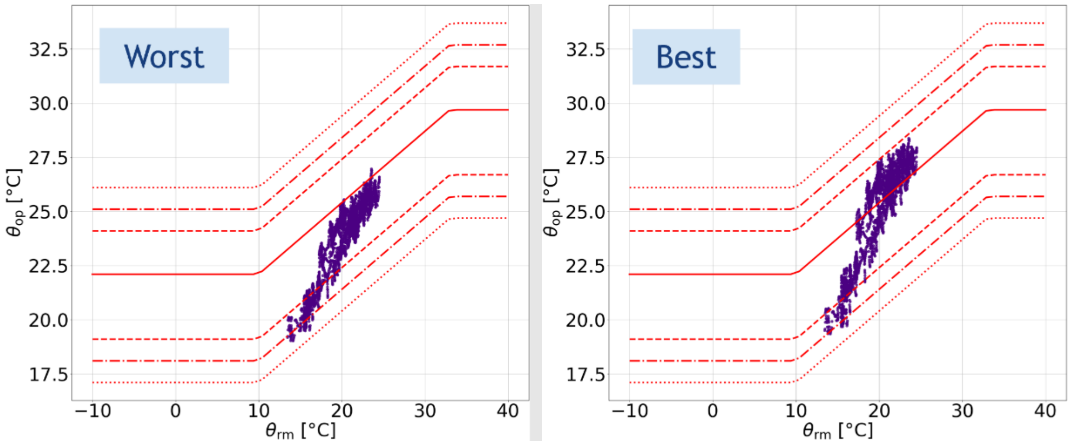


Figure 18 TPM adaptive comfort model distribution

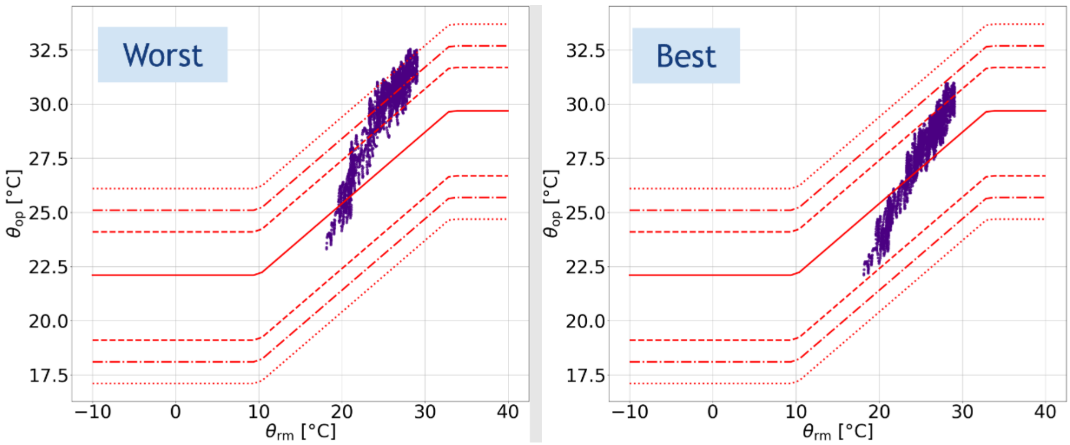


Figure 19 Rome adaptive comfort model distribution

This series of sample applications also show the ability of the proposed dynamic simulation approach in helping designers and energy managers to select correct control thresholds to support the optimisation of free-running usage during design and technology implementation steps.

4.2 Performance gap

The performance gap scenario, in accordance with WP2 outcomes, allows to deepen the behavioural gap between standard simulated building conditions and actual behaviour. To obtain meaningful results for the comparison, the simulated building model should be calibrated to follow the real monitored trend: this can be obtained both through traditional manual methods (e.g., through DesignBuilder or OpenStudio interfaces themself) finally generating a modified IDF or exploiting the functionalities of automatic and semi-automatic approaches, like another PREDYCE scenario named model verification and described in the following section 4.3. Then, input JSON file, described in correlated deliverable D3.1, can be used to modify the calibrated model to standard conditions for what concerns e.g., occupancy, setpoints, ventilation, and to standard modified conditions, as described in WP2 deliverables. This is obtained transforming the *preliminary_actions* field (listing IDF editing functions to be applied once before simulating) into a list of JSONs, considered in the code as a parameter. Moreover, the scenario takes in input an EPW file generated from monitored data from weather stations and a CSV file containing indoor environmental monitored data for the considered period. The adopted format of sensor's naming, described in section 4.4, allows to exploit KPIs calculator module also on monitored data and to match spatial aggregations on which KPIs are computed to simulated model thermal zones.

Results of performance gap scenario contain the two CSV files described in deliverable D3.1, so *data_res.csv* and *data_res_timeseries.csv*, but slightly modified in accordance with scenario peculiarities. Particularly, a column called *data* in *data_res.csv* is descriptive: *sim* x corresponds to the model setting x described in input JSON, then monitored corresponds to KPIs computed on monitored data, and Δx corresponds to KPIs delta computed as monitored results minus building simulated in setting x results. Timeseries results instead follow the same principle but in columns names. Then results are aggregated in a zip folder in order to be sent via REST API to FUSIX platform which acts as E-DYCE middleware. In fact, such as sensitivity analysis scenario, also performance gap can be run remotely via a REST API or through a dedicated web interface[4].

Figure 20 summarizes performance gap scenario input/output workflow. Together with previously described outputs, it could be possible to return also plots both generated inside KPIs computation and from post-analysis of building behavioural comparison.

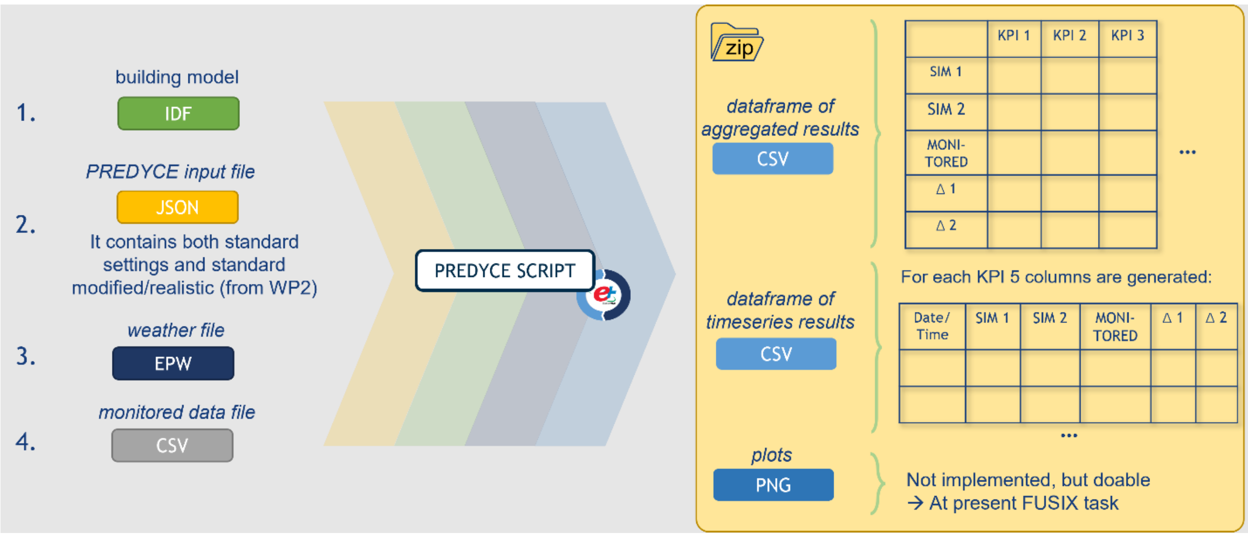


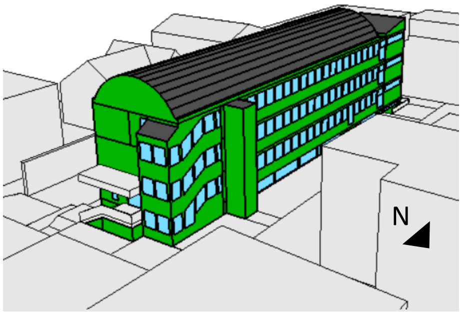
Figure 20 Performance gap input/output workflow

Through same credentials used for deliverable D3.1 example, it is possible to remotely execute an example of performance gap scenario run via the REST API or directly through the web interface (results will be in the Download folder of the browser). Input files and outputs can be found in a shared folder[5] and in the following a detailed example explanation is given. One of TPM demo cases, the municipality school, is used as an example: figure 21 shows the school model from which the IDF was exported. The

[4] http://130.192.20.228:3200/pg (Note: some Wi-Fi connections may not support it)

[5] https://www.dropbox.com/sh/zcrl37umjbi8b0p/AACKVlGsjlyWZmlbwnpWZQV3a?dl=0

school is made of four almost equally organized floors, with corridor on the north façade and several classrooms and labs on the south; toilets are located at the west end of the corridor, while entrance and teacher rooms at the east side. Since the school is a quite complex and big building, separate similar models have been used for each floor in order to reduce simulation time. However, particularly for remote execution simulation time can be quite long (around half an hour).



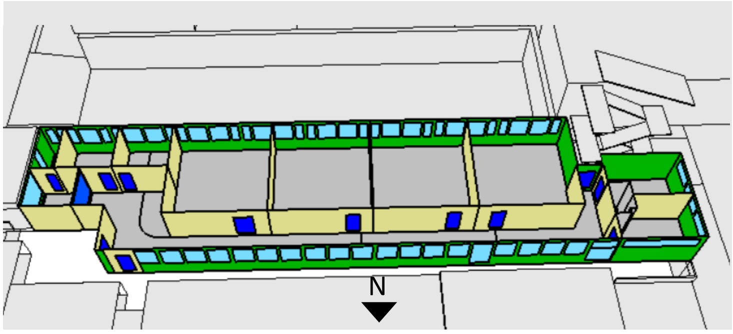


Figure 21 TPM school demo case model

Since there are still few monitored data for energy consumption, performance gap analysis for this example is focused on CO_2 and indoor air temperature. The simulation period was set from 16/09/2021 to 31/10/2021, considering school starting date and that in mid-October HVAC system was switched on. Figure 22 shows KPIs computed for this example and spatial aggregations considered: activity 202 is linked to common circulation areas, while activities 201xx to specific teaching areas (see also Section 4.4). Moreover, aggregated results over the whole floor are computed. This is an example of a quite complex multi-zone model, in which analyses are performed room-by-room, but it is possible to use the same methodology considering the critical zone approach proposed in WP2 outcomes: in this case spatial aggregations will be related just to the whole floor/building and to the critical room, chosen through given inspection guidelines, which will be identified with a given name (free but structured) as described in nomenclature section 4.4. However, since the identification of the critical room is inspection-based, the building model has to be created having in mind it and simulation just uses the zone name for KPIs analysis. Timeseries results for CO_2 and indoor air temperature are saved in *data_res_timeseries.csv* file, while aggregated results for Adaptive Comfort Model and CO_2 analysis are saved in *data_res.csv* file.

```
"kpi": {
    "adaptive_comfort_model": {},
    "timeseries_t_db_i": {},
    "timeseries_co2": {},
    "n_co2_aIII": {},
    "n_co2_bI": {}
},
"aggregations": {
    "adaptive_comfort_model": ["act202", "act201", "act201ba",
                               "act201bb", "act201bc", "act201bd"],
    "timeseries_t_db_i": ["act202", "act201", "act201ba",
                          "act201bb", "act201bc", "act201bd"],
    "timeseries_co2": ["act201", "act201ba", "act201bb", "act201bc", "act201bd"],
    "n_co2_aIII": ["act201", "act201ba", "act201bb", "act201bc", "act201bd"],
    "n_co2_bI": ["act201", "act201ba", "act201bb", "act201bc", "act201bd"]
}
```

Figure 22 Input JSON KPIs

Figure 23 shows the list of preliminary actions applied to the sample model: the model was exported already set with specific conditions for occupancy, ventilation, setpoints (representing a sample standard) and consequently the first dictionary is empty; the second dictionary instead represents standard modified conditions and contains a small schedule change more aligned with real use (the school is almost empty in the afternoon). However, finer schedule assignments will be done for demo cases applications in further deliverables (WP5), while the proposed example is only meant to explain the followed methodology and feasibility. Additional preliminary actions are automatically performed by the scenario script: activate CO_2 analysis, align simulated days of the year to real weekdays (setting the first day of the year inside the simulation) and change the run period. Moreover, in the inputs folder EPW file generated with monitored data from TPM weather station can be found, together with the CSV file of measured indoor environmental data (*school1_f01.csv*). The proposed example is not executed on a calibrated model, but analysis on temperatures trend in summertime showed it was already quite well aligned without big changes, consequently despite results should not be interpreted as fully meaningful considering the real demo case, they still can give an initial idea of performance gap scenario working in this demo.

```
"preliminary_actions": [{}, {"change_occupancy": {"schedule": {}}}]
```

Figure 23 Input JSON preliminary actions

Focusing on aggregated results as shown in Figure 24 (just as an average on the whole floor), the CSV file is composed by five rows named *simulated_1* and *_2*, *monitored* and *delta_1* and *_2*, corresponding to the two settings defined in preliminary actions field in the input JSON file, and columns for each KPI and spatial aggregation: so, if the number of hours above CO_2 threshold III (1000 ppm), named *n_co2_aIII*, is computed on five different spaces plus the entire floor/model (which is always considered without specifying it), six columns will be related to this KPI and identifiable by *_space-name*. Some results in the file are numbers, such as results related to CO_2 analysis, while others can be dictionaries, such as hours distribution in ACM categories.

adaptive_comfort_model	n_co2_aIII	n_co2_bI	data
{"cat I": 828, "cat II up": 41, "cat III up": 16, "cat over III": 0, "cat II down": 113, "cat III down": 53, "cat under III": 29, "POR": 0.077}	35	721	simulated_1
{"cat I": 861, "cat II up": 40, "cat III up": 5, "cat over III": 0, "cat II down": 127, "cat III down": 34, "cat under III": 13, "POR": 0.041}	109	818	simulated_2
{"cat I": 568, "cat II up": 23, "cat III up": 0, "cat over III": 0, "cat II down": 236, "cat III down": 121, "cat under III": 35, "POR": 0.125}	11	849	monitored
{"cat I": -260, "cat II up": -18, "cat III up": -16, "cat over III": 0, "cat II down": 123, "cat III down": 68, "cat under III": 6, "POR": 0.048}	-24	128	delta_1
{"cat I": -293, "cat II up": -17, "cat III up": -5, "cat over III": 0, "cat II down": 109, "cat III down": 87, "cat under III": 22, "POR": 0.084}	-98	31	delta_2

Figure 24 Example of data_res.csv for performance gap

Concerning timeseries results instead, as shown in Figure 25, indexes are date/time values for the considered time period, while columns are KPIs for each spatial aggregation and the five scenario outputs (simulated_, monitored and delta_): so, considering hourly CO_2 values and the 6 spatial aggregations considered in the example, there will be 30 (5x6) columns related to this KPI, constructed as *simulated_x_timeseries_co2_space-name*.

	simulated_1_timeseries_t_db_i_act202	simulated_1_timeseries_t_db_i_act201	simulated_1_timeseries_t_db_i_act201ba
16/09/2021 00:00			
16/09/2021 01:00	23.67607519	27.20036908	26.8413237
16/09/2021 02:00	23.50674921	26.9809892	26.62869727
16/09/2021 03:00	23.36610899	26.78733746	26.43820069
16/09/2021 04:00	23.20507801	26.58442909	26.23686938
16/09/2021 05:00	23.03439047	26.38048788	26.03437667

Figure 25 Example of data_res_timeseries.csv for performance gap

Figures 26 and 27 show timeseries results for CO_2 and indoor air temperature averaged in all floor classrooms and in a specific classroom. During last week in October, the profile used for standard occupancy schedule includes a holyday period, consequently the CO_2 generation for simulation 1 drops to 0; simulation 2 instead shows an abnormal peak since ventilation was not changed and set to 0 because of the holidays, while occupancy was set to fully functional. Temperature trends seem to align better from mid-October, when HVAC system is in reality switched on: this could be due to a consequent more standardized use of natural ventilation.

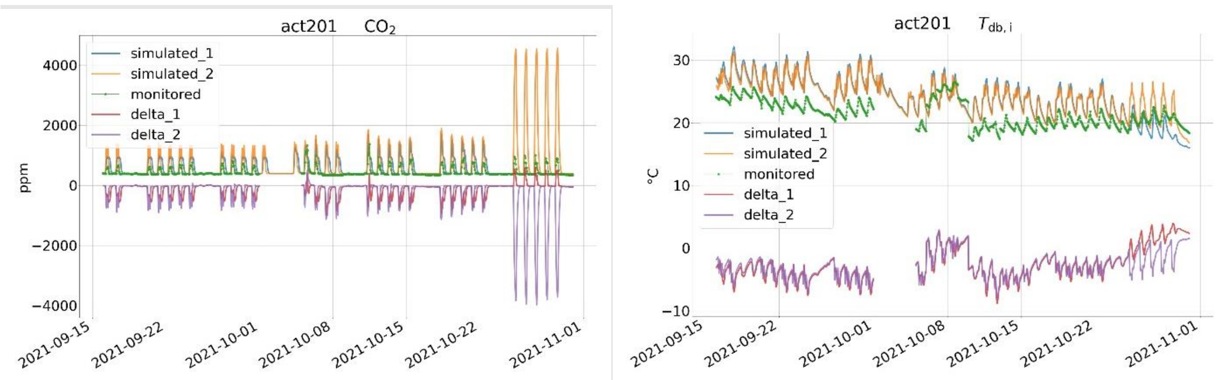


Figure 26 Timeseries results averaged on all classrooms

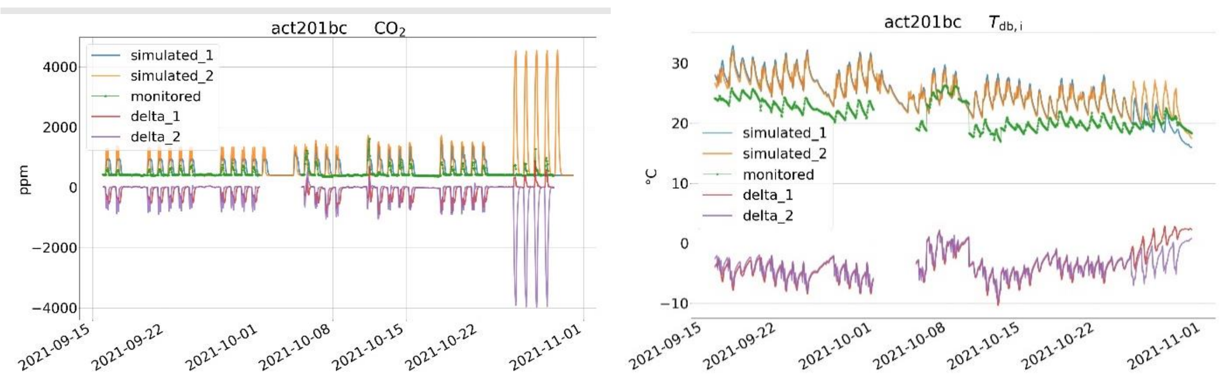


Figure 27 Timeseries results in a specific classroom

4.3 Model verification

A semi-automatic calibration process, that can be subjected to future updates and improvements, has been developed inside PREDYCE to adjust the building model to measured data, speeding up the manual procedures usually adopted for this purpose. The following PREDYCE IDF editing actions are currently available to be tried inside this scenario, to try aligning the simulated trend to actual building behavior:

- Change U-value of walls and roof (acting on thickness of mostly insulated layer);
- Change U-value and SHGC (Solar Heat Gain Coefficient) of windows;
- Change internal mass and equipment gains in each thermal zone;
- Change ACH ventilation and infiltration.

The calibration process exploits also PREDYCE EPW compiler module, able to generate an EPW from monitored weather data. Model verification is possible thanks to PREDYCE ability of computing same KPIs on both simulation results and indoor monitored data. Particularly, the adopted procedure is inspired by (Claridge and Paulus, 2019) – see also E-DYCE D1.2 – and consists in optimizing a combined error measure including both RMSE (Root Mean Square Error) and MBE (Mean Bias Error), see the following equation, on a given variable or combination of variables, e.g., indoor dry bulb temperature in free-running conditions or on heating/cooling consumption.

$$\mathrm{Error}_{\mathrm{tot}} = \sqrt{RMSE^2 + MBE^2}$$

The calibration signature described in (Claridge and Paulus, 2019) is computed according to the following equation (taking indoor dry bulb temperature as objective variable), exploiting PREDYCE potentialities.

$$\text{Calibration signature} = \frac{\text{measured } T_{\mathrm{db}}^{\mathrm{i}} - \text{simulated } T_{\mathrm{db}}^{\mathrm{i}}}{\text{max measured } T_{\mathrm{db}}^{\mathrm{i}}} \cdot 100\%$$

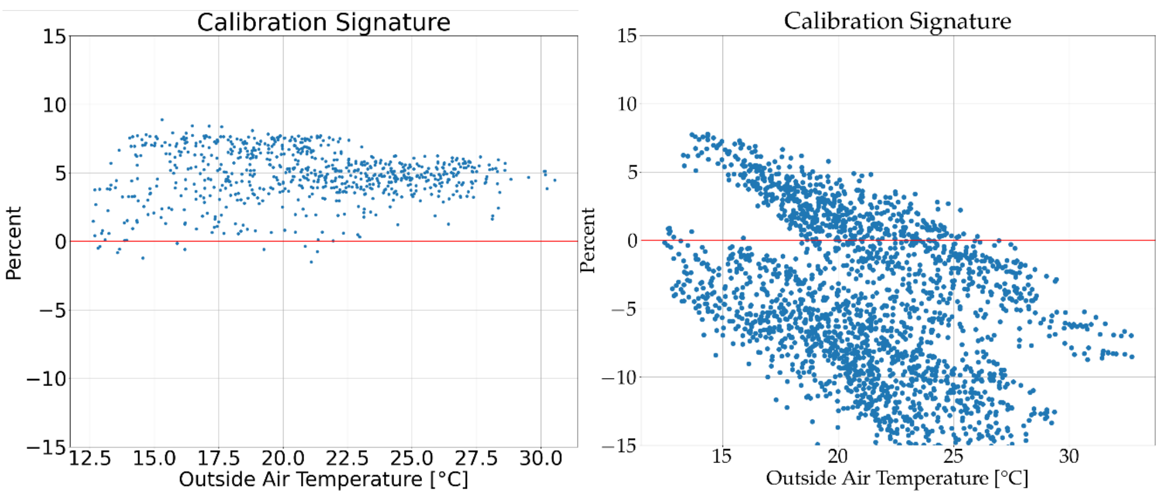


Figure 28 Example of calibration signatures

The different implemented IDF editing actions allow to both shift the curve (e.g., acting on ACH, equipment gains), change coefficient and inclination, and to modify amplitude variations (e.g., acting on internal mass), such allowing to reach a flat line inside 5% error range, in line with reference suggestions – see also ASHRAE Guideline 14-2014 for calibration criteria. Figure 28 shows some initial (before calibration) examples of calibration signatures plots on TPM demo cases.

Which parameters to change and their range (e.g., change walls U-value in range 30% with respect to original model value) can be defined considering the knowledge of each demo case, e.g., if windows are newly installed and parameters are known they could not be included in the loop. Other actions, still not automatized in the process, can also be available (e.g., changing zone area, change window area, change windows visible transmission factor) if needed in a particular case. Similarly, also occupancy profiles and intensities may be varied, even randomly, but are at present not included in the automatic loop having already an internal gain voice. The model verification scenario is at present considered to be semi-automatic since, to minimize the number of simulations to be performed, it requires to look at the calibration signature plot in order to define which parameters to try. Also, after having established a parameter range, the user may decide to try expanding it, if for example the found minimum is at some extremities and re-execute the scenario. This procedure may be furtherly automatized in the future, but the potentiality to test many building parameters simultaneously and automatically editing the model already provides an improvement in terms of effort and time with respect to traditional manual procedures.

Here an example of initial model verification scenario applied to one of E-DYCE demo case is reported. It concerns a residential demo case located in TPM and covers the period of June 2021. Indoor air temperature is taken as reference variable since cooling system is not present in the house. Optimized values for opaque envelope U-factor, windows parameters, internal mass (which allows to consider both interior furniture and previously not considered envelope effects), ACH from ventilation and infiltration and equipment gains are found with previously described methodology, allowing to reduce the gap between average simulated and measured temperature (over the same pre-defined zones considered to be representative). Figure 29 shows the gap reduction between starting model and optimized, both on peaks amplitude and average values. Moreover, figure 30 shows calibration signature of best case, which appears flat and inside 5% error boundary with hourly timestep.

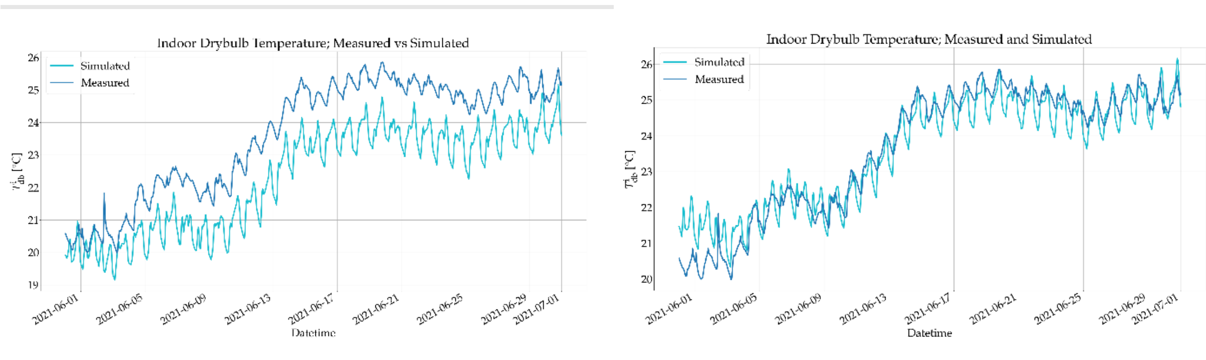


Figure 29 Indoor drybulb temperature trend, before and after model verification

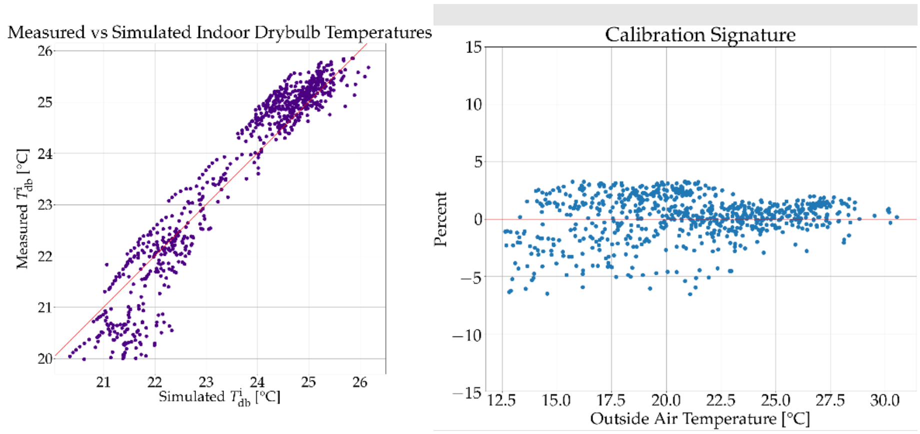


Figure 30 Calibration signature and data comparison at the end of model verification

Specific descriptions about the application of PREDYCE in demos are included in related deliverables from WP5.

4.4 Sensor's nomenclature discussion

Each PREDYCE scenario involving comparisons between monitored and simulated data requires a correct association among sensors located in the building and model thermal zones, such that spatial aggregations for KPIs analysis correspond. Consequently, the nomenclature schema in figure 31 has been proposed. The naming part preceding the sensor MAC address follows the IDF models naming structure made of building name, block name and thermal zone name, so same names must be used both inside the building model (when initially creating it through an interface) and on sensors ID. It was proposed to follow a code structure for the names, but for the tool working it is enough to guarantee the correspondence. This coherence allows a strict spatial correspondence, making it possible to aggregate at both building or block level. Moreover, at the end of the naming structure must be included the name of the measured variable (usually corresponding to different transmission channels inside the same sensor). The variable name is then used inside each KPI to recognize which CSV columns include in the computation.

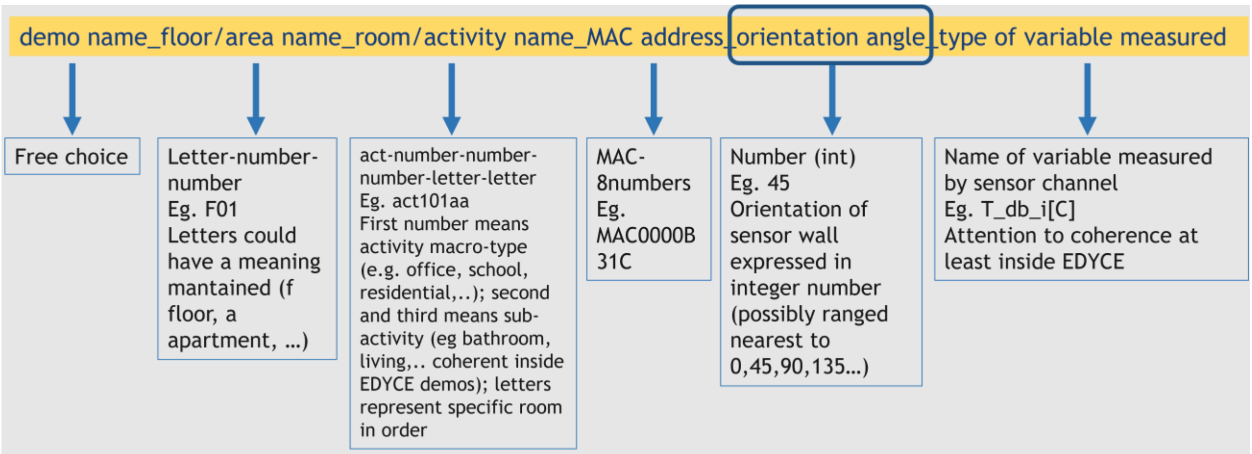


Figure 31 Sensor's nomenclature scheme

Since the nomenclature schema refers to IDF fixed space nomenclature structure, it can work both in case of multi-zones or mono-zone models. The only difference is that in case of monozonal analysis block name and thermal zone name is the same since all thermal zones collapse into one.

5 Conclusions and Outlook

Contents of this report are supporting T3.2 results that also include the possibility to run sensitivity analyses and the integration of free-running devoted IDF actions and KPIs into the sample dynamic simulation platform described in D3.1. During next project steps the above-described scenarios and are expected to be tested on project demo cases supporting potential integration and upgrading actions to solve challenges and adapt initial actions to specific demo requests – see next WP4 and WP5 deliverables. Specific publications are also expected to be developed during next months to disseminate initial results in addition to the on mentioned in D3.1, which is devoted to initial descripting the proposed dynamic simulation platform. Currently, the 'DYCE' version of PREDYCE is not conceived to support real-time suggestions or forecasting, but specific sensitivity analyses may be adopted to suggest control thresholds or specific technologies – e.g., Free-running ones – by analysing their impacts on expected energy and comfort results. Additionally, the introduced fictitious cooling/heating approach, able to be adapted to different indicators, e.g., IAQ indices, represents a potential calculation approach to correlate comfort and discomfort conditions in free-running buildings with energy needs (fictitious ones) to eventually compare (and label) traditional buildings without losing their free-running mitigation potential in respect to mechanically controlled spaces. Demo implications will be, although, analysed during next project steps.

6 Bibliography

Artmann, N., Manz, H., Heiselberg, P., 2007. Climatic potential for passive cooling of buildings by night-time ventilation in Europe. Applied Energy 84, 187–201. https://doi.org/10.1016/j.apenergy.2006.05.004

CEN ISO/TR, 2017. Energy performance of buildings - Indicators for partial EPB requirements related to thermal energy balance and fabric features - Part 2: Explanation and justification of ISO 52018-1 (ISO/TR 52018-2:2017).

Chiesa, G., Fasano, F., Grasso, P., 2021a. A New Tool for Building Energy Optimization: First Round of Successful Dynamic Model Simulations. Energies 14, 6429. https://doi.org/10.3390/en14196429

Chiesa, G., Grosso, M., 2015. Geo-climatic applicability of natural ventilative cooling in the Mediterranean area. Energy and Buildings 107, 376–391. https://doi.org/10.1016/j.enbuild.2015.08.043

Chiesa, G., Heiselberg, P.K., Kolokotroni, M., 2021b. Innovations in Ventilative Cooling. Springer International Publishing, Cham.

Chiesa, G., Simonetti, M., Ballada, G., 2017. Potential of attached sunspaces in winter season comparing different technological choices in Central and Southern Europe. Energy and Buildings 138, 377–395. https://doi.org/10.1016/j.enbuild.2016.12.067

Chiesa, G., Zajch, A., 2020. Contrasting climate-based approaches and building simulations for the investigation of Earth-to-air heat exchanger (EAHE) cooling sensitivity to building dimensions and future climate scenarios in North America. Energy and Buildings 227, 110410. https://doi.org/10.1016/j.enbuild.2020.110410

CISBE, 2006. Degree-days: Theory and application (No. TM41: 2006). The Chartered Institution of Building Services Engineers, London.

Claridge, D., Paulus, M., 2019. Building simulation of practical operational optimisation, in: Hensen, J., Lamberts, R. (Eds.), Building Performance Simulation for Design and Operation. Routledge, London ; New York, pp. 399–453.

Cook, J. (Ed.), 1989. Passive Cooling. MIT press, Cambridge.

De Lia, F., Schioppo, R., Lo Presti, R., Pizzuti, S., Romano, S., Tundo, A., Zangheri, P., Zini, P., Zinzi, M., 2022. Implementazione delle logiche di gestione per gli smart building di seconda generazione (No. RdS/PTR(2021)/008, 2022). ENEA, Rome.

EDYCE, 2021. D1.2 Operational dynamic Energy Perfomance Certificate (EPC) specifications (Project report No. D1.2). PoliTO.

EN ISO, 2005. Ergonomics of the thermal environment - Analytical determination and interpretation of thermal comfort using calculation of the PMV and PPD indices and local thermal comfort criteria (ISO 7730:2005).

European Committee for Standardization, 2019. EN 16798-1:2019 - Energy performance of buildings - Part 1: Indoor environmental input parameters for design and assessment of energy performance of buildings addressing indoor air quality, thermal environment, lighting and acoustics. Brussels.

European Committee for Standardization, 2007. EN 15251:2007 - Indoor environmental input parameters for design and assessment of energy performance of buildings addressing indoor air quality, thermal environment, lighting and acoustics. Brussels.

Ghiaus, C., 2003. Free-running building temperature and HVAC climatic suitability. Energy and Buildings 35, 405–411. https://doi.org/10.1016/S0378-7788(02)00110-X

Givoni, B., 1994. Passive and low energy cooling of buildings. Van Nostrand Reinhold, New York.

Liggett, R., Milne, M., 2017. Climate Consultant 6.0. UCLA Energy Design Tools Group.

Santamouris, M., Asimakopolous, D. (Eds.), 1996. Passive Cooling of Buildings. James and James, London.