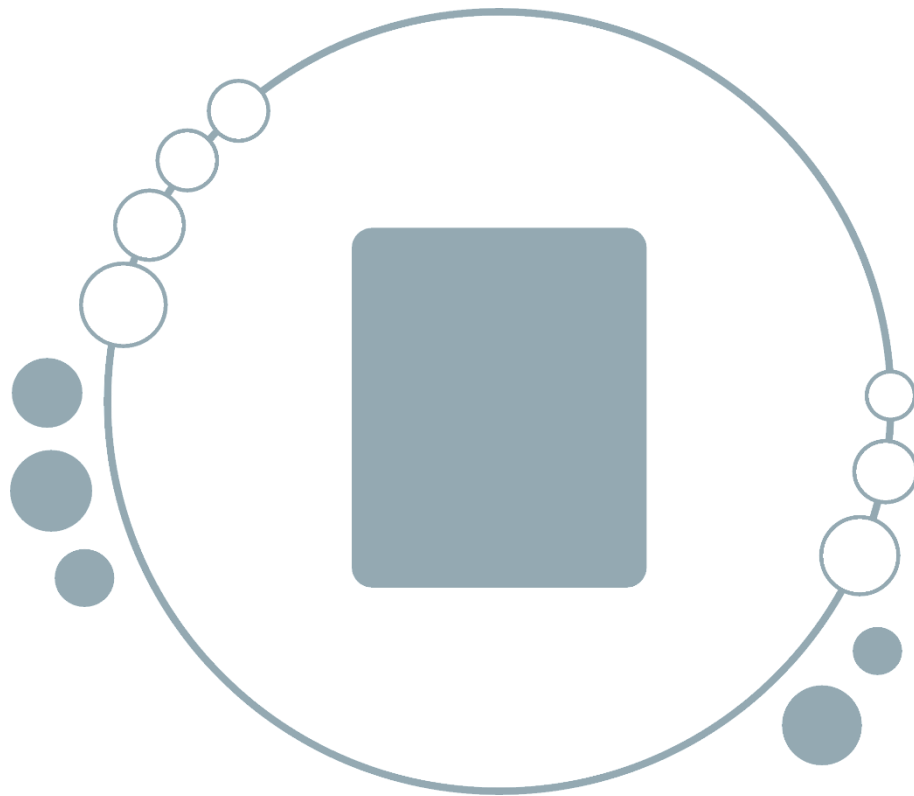




General summary of the Home Energy Model core calculation

A technical explanation of the methodology

Acknowledgements

This methodology has been developed for the Department for Energy Security & Net Zero by a number of organisations and individuals, including Sustenic, Qidos, Scene Connect, City Science, Hoare Lea, Oxford Brookes University, University of Bath, 10-x, Building Research Establishment (BRE), AECOM, Kiwa Ltd., Loughborough University Enterprises Limited, Chris Martin and John Tebbit.

Quality assurance has been undertaken by a consortium led by Etude, including Levitt Bernstein, People Powered Retrofit, University of Strathclyde's Energy Systems Research Unit, Julie Godefroy Sustainability, and UCL.

Document reference: HEM-TP-01

Document version: v2.0

Issue date: October 2025

Home Energy Model version: HEM 1.0



© Crown copyright 2025

This publication is licensed under the terms of the Open Government Licence v3.0 except where otherwise stated. To view this licence, visit nationalarchives.gov.uk/doc/open-government-licence/version/3 or write to the Information Policy Team, The National Archives, Kew, London TW9 4DU, or email: psi@nationalarchives.gsi.gov.uk.

Where we have identified any third-party copyright information you will need to obtain permission from the copyright holders concerned.

Any enquiries regarding this publication should be sent to us at: homeenergymodel@energysecurity.gov.uk

Contents

Background to the Home Energy Model	4
What is the Home Energy Model?	4
Where can I find more information?	4
Related content	5
Methodology	6
1. Design principles	6
2. Overall calculation structure	7
2.1. Water heating calculation structure	8
2.2. Space heating and cooling calculation structure	8
2.3. Systems providing several services	10

Background to the Home Energy Model

What is the Home Energy Model?

The [Home Energy Model \(HEM\)](#) is a calculation methodology designed to assess the energy performance of homes, which will replace the government's [Standard Assessment Procedure \(SAP\)](#).

Where can I find more information?

This document is part of a wider package of material relating to the Home Energy Model.

Home Energy Model technical documentation (e.g. this document)

What: This document is one of a suite of [technical documents](#), which explain the calculation methodology in detail. New documents will be added, and the content amended, when necessary to ensure documentation is sufficiently comprehensive. This will usually, but not always, occur alongside the release of a new version of HEM.

Audience: The technical documentation will be of interest to those who want to understand the detail of how the Home Energy Model works and how different technologies are treated.

The Home Energy Model consultation and government response

What: The [Home Energy Model consultation](#) introduces the overhaul to the SAP methodology and sought views on the approach taken by the new Home Energy Model. The [Home Energy Model consultation](#) summarises the feedback to the consultation and the actions taken subsequently in development, ahead of the initial release of HEM.

Audience: The Home Energy Model consultation will be of interest to those seeking a general introduction to HEM and its role in government policy on domestic energy performance.

The Home Energy Model reference code

What: The full Python source code for the Home Energy Model core engine has been published as a [Git repository](#). Note the reference code for official HEM wrappers is published separately.

Audience: The reference code will be of interest to those who want to understand how the model has been implemented in code, and those wishing to fully clarify their understanding of the new methodology. It will also be of interest to any potential contributors to the Home Energy Model or those wishing to use it within their own projects.

Related content

This document provides a general summary of the overall structure of the Home Energy Model (HEM) core engine and the design principles underpinning it (with particular reference to how the calculation has been implemented in the code). For a general summary of the role and structure of wrappers, see HEM-TP-02 General summary of wrappers.

To understand how this methodology has been implemented in code, please see:

src/hem_core/project.py

Methodology

The Home Energy Model software is structured to facilitate the use of different sets of assumptions for different policy requirements. Fundamentally, this means that the software is divided into:

- A core calculation component which models heat transfer in the building, demand on heating, cooling and hot water systems and the resulting energy demand for electricity, mains gas etc.
- “Wrapper” components which add pre-processing (e.g., standardisation of occupancy assumptions) and post-processing (e.g., application of emissions and primary energy factors) steps to the core calculation which are suitable for different policies.

1. Design principles

The core calculation has been designed to be relatively unconstrained in the input specifications it accepts. For example, it does not put any limit on the number of zones as it is expected that any such limits required will be enforced in the relevant wrapper. The space and water heating systems can have several components (e.g., a hot water cylinder heated by multiple heat sources) and some components can provide more than one service (e.g., a heat pump providing both space heating and water heating).

The core calculation has a modular structure to maximise flexibility, the ability to add new technologies in the future, and the ability to upgrade the treatment of technologies already covered. Each technology and component of a building system has its own module. The different modules are relatively self-contained and have few (or ideally, no) dependencies on the implementation details of other modules, which means that they can be connected together in many different ways relatively simply. For example, the boiler module can be connected directly to the water heating demand to model a combi boiler, or can be connected to the StorageTank module to model a regular or system boiler.

In the software, modules¹ are implemented using classes/objects² which can have internal properties/state and can communicate with each other. For example, if a hot water cylinder is present, then an instance of the StorageTank class will be created, with parameters including the volume of water stored and the standby loss rate. The StorageTank object will also have internal state in the form of the water temperature at different layers of the cylinder, which is invisible to other objects but which will affect the values (e.g. water heating demand) which the StorageTank object sends to other objects and will be affected by the values (e.g. hot water draw-offs and heat provided to the cylinder) received from other objects.

¹ The term “module” can have multiple, subtly different meanings, depending on whether it refers to Python modules or conceptual modules in the Home Energy Model calculation. Therefore, the term will be avoided in favour of “class” or “object”.

² A class is a type of object, and an object is an instance of a class. There may be multiple objects of a given class, with different characteristics (where such characteristics are parameterised).

Each object will also have references to (i.e., will be connected to) other relevant objects. In the case of a StorageTank object, these will include those objects representing systems that provide heat to the cylinder (e.g., an ImmersionHeater object), which in turn may have references to other objects (e.g., an ImmersionHeater object may be connected to a SetpointTimeControl object). When an object has a reference to another object, it is able to call that object's public methods³ in order to interact with it. This means that the control flow of the calculation will pass back and forth between several different objects during each timestep.

Together, a class's public methods constitute its interface. In order to maintain flexibility and extensibility of the software, classes that fulfil the same role should have the same interface, so that one can be substituted for another without requiring the objects using the interface to have any knowledge of what type of object is behind the interface⁴. Therefore, adding a new heating system to the model should usually⁵ just be a case of implementing a new class which has an interface that is compatible with the interfaces of the existing heating system classes, rather than having to define relationships separately for every possible combination of objects.

2. Overall calculation structure

The calculation starts by initialising all the required objects based on the contents of the input data (either directly entered or from the relevant wrapper). It is at this stage that any internal variables and references between objects are initialised and any calculations that do not depend on the timestep are performed. Once all the objects have been initialised, the main calculation loop begins, and there will be one iteration of this loop run for each timestep in the calculation. As described above, data flows between objects are two-way in many cases and the control flow of the calculation goes in and out of different objects several times.

The main calculation loop can be divided into stages:

1. Calculation of hot water demand, water heating energy demand, delivered energy and fuel consumption by the systems meeting this demand (including distribution losses and associated heat gains) – see section 2.1.
2. Calculation of electricity generation and associated heat gains – see HEM-TP-18 PV generation and self-consumption
3. Calculation of space heating/cooling demand, delivered energy, fuel consumption by systems meeting this demand, and resulting internal air and fabric temperatures – see section 2.2.

³ In addition to having variables that are only for internal use (known as private member variables), classes may also have functions that are only for internal use (known as private member functions or private methods) and functions that are for internal or external use (known as public member functions or public methods).

⁴ In software development terms, the relevant concept is "polymorphism".

⁵ For systems that have more complex interactions with other systems, it may sometimes be necessary to make further modifications to accommodate these additional interactions.

4. Additional calculations for any systems that provide multiple services (e.g., combi boiler providing both hot water and space heating) – see section 2.3.
5. Calculation of electricity self-use and storage, and resulting import and export of electricity from/to the grid – see HEM-TP-18 PV generation and self-consumption.

Results are then returned to the wrapper (if any) for post-processing, and also written to the relevant output files.

2.1. Water heating calculation structure

(Stage 1 from the description of “Overall calculation structure”)

The general structure of the water heating calculation is described in HEM-TP-09 Energy for domestic hot water. At a high level, it consists of two stages, conducted in parallel for each water heating system and its associated outlets:

1. Calculate the outlets’ demand for hot water.
2. Calculate the energy required by the water heating system to provide the hot water required.

2.2. Space heating and cooling calculation structure

(Stage 3 from the description of “Overall calculation structure”)

The general structure of the space heating calculation is as follows:

1. The air change rate required to meet the ventilation setpoint is calculated (on a whole-dwelling basis), adjusting vent positions if necessary (see HEM-TP-06 Ventilation and Infiltration)
2. The internal and solar gains for each zone are calculated (internal gains from hot water and electricity generation are inputs to the space heating calculation and are added to the total internal gains)
3. The Project object retrieves the following parameters from the heating/cooling system objects assigned to each zone, and puts the heating and cooling systems in priority order for each zone based on the setpoint (highest setpoint first for heating and lowest setpoint first for cooling, with ties broken by preserving the order of entry in the inputs):
 - a. Convective fraction
 - b. Heating/cooling setpoint (the heating/cooling system object in turn retrieves this from the control object assigned to the heating/cooling system).
4. The Project object provides the setpoints and convective fractions for the highest-priority heating and cooling systems, along with the results from steps 1 and 2, to the

appropriate Zone objects, which, for the zone they represent, calculate the space heating/cooling demand, the air change rate that would be required to cool the zone to the passive cooling setpoint and the air change rate that would be required to cool the zone to the heating setpoint, assuming no heating/cooling contribution from lower-priority systems. The Zone object will retrieve other inputs from various other objects representing building elements etc. as needed. These results are then returned to the Project object.

5. The Project object calculates the overall air change rate required, based on the ventilation setpoint and any additional passive cooling requirements. The maximum of the air change rates required to cool each individual zone to the passive cooling setpoint is used as the air change rate for all zones for the rest of the calculation, subject to the following limits:
 - a. If there is demand for heating in any zone, the air change rate is that required to meet the ventilation requirement only.
 - b. If there is demand for active cooling in any zone, the air change rate is that required to meet the ventilation requirement only. In this case, the active cooling demand for all zones is recalculated based on this air change rate.
 - c. The air change rate should not fall below the ventilation requirement (i.e. windows should not be closed to reduce heating demand to the extent that it causes insufficient ventilation to be provided).
 - d. The air change rate should not exceed the value that would cause the temperature of any zone to fall below the heating setpoint for that zone (i.e. windows should not be opened to the extent that it causes additional heating demand), unless this is required to meet condition c.
6. The Project object queries the heating and cooling systems assigned to each zone to determine their minimum output (e.g. radiators that are still hot from previous timesteps will still provide heating even when there is no demand).
7. For each zone, the Project object feeds the relevant inputs to the Zone object to calculate the space heating/cooling demand for the highest-priority system that has not yet been considered⁶, accounting for any output from other systems (either the actual output already calculated for higher-priority systems or the minimum output of lower-priority systems) and the setpoint and convective fraction of the relevant system.
8. The Project object feeds the space heating/cooling demand values to the relevant space heating/cooling system object(s).
9. The space heating/cooling system object(s) then calculate:

⁶ Note: the calculation in step 4 does not count as consideration of the highest-priority system because it does not account for the minimum output of lower-priority systems.

- a. The energy supplied by the system (which does not necessarily meet the demand or could exceed the demand if controls are poor), which is then returned to the Project object.
 - b. The associated fuel consumption, which is provided to the relevant EnergySupply object(s) to which they are connected.
10. If there are still heating/cooling systems which have not been considered, then the calculation returns to step 7.
11. Once all heating/cooling systems have been considered, the unmet demand is calculated – see HEM-TP-04 Space heating and cooling demand.
12. The Project object then feeds the energy supplied by the heating/cooling systems into the relevant Zone objects, where resulting temperatures achieved are calculated and saved (as internal state) for the next timestep.

Within step 9 above, the heating/cooling system can be represented by one or more objects with or without internal state. For example, an instantaneous electric heater is represented by a single object with no internal state whereas a central heating system will be represented by separate objects for the emitter system and the heat source (e.g., boiler or heat pump) which do have internal state (e.g., temperature of emitters).

The overall space heating and cooling demand figures reported for each zone are the figures calculated in step 4 (or step 5.b for cooling demand, where relevant).

2.3. Systems providing several services

(Stage 4 from the description of “Overall calculation structure”)

For systems that provide several services (e.g., combi boiler providing hot water and space heating) there may be some dependency between the services. For example, the combined power output provided for each service cannot exceed the maximum power output of the system. Therefore, these modules contain additional internal state variables to track these dependencies. These internal state variables need to be reset at the end of the timestep, and it is also at this time that any other calculations requiring results from all the service calculations are performed.

As an example, for a heat pump providing both water heating and space heating, calculations within the heat pump object would be as follows (simplified description):

1. Water heating demand would be provided as input (part of step 1 from the description of “Overall calculation structure”), and the performance of the heat pump would be calculated, including the required running time to satisfy the demand for water heating. This running time would then be stored in an internal state variable in the heat pump object.

2. Space heating demand would be provided as input (step 6 from the description of “Space heating calculation structure”), and the performance of the heat pump would be calculated, including the required running time to satisfy the demand for space heating, and taking into account the running time already required for water heating. The internal state variable tracking running time in the heat pump object would then be updated with the combined running time
3. After all services have been calculated (step 4 from the description of “Overall calculation structure”), the combined running time is used for further calculations, such as the time the heat pump spends in standby mode and the associated standby energy consumption.

