

# Software Requirements Specification for Solar Water Heating Systems Incorporating PCM

Thulasi Jegatheesan, Brooks MacLachlan, and W. Spencer Smith

November 27, 2019

## Contents

<b>1</b>	<b>Reference Material</b>	<b>3</b>
1.1	Table of Units . . . . .	3
1.2	Table of Symbols . . . . .	3
1.3	Abbreviations and Acronyms . . . . .	6
<b>2</b>	<b>Introduction</b>	<b>6</b>
2.1	Purpose of Document . . . . .	6
2.2	Scope of Requirements . . . . .	7
2.3	Characteristics of Intended Reader . . . . .	7
2.4	Organization of Document . . . . .	7
<b>3</b>	<b>General System Description</b>	<b>8</b>
3.1	System Context . . . . .	8
3.2	User Characteristics . . . . .	8
3.3	System Constraints . . . . .	8
<b>4</b>	<b>Specific System Description</b>	<b>9</b>
4.1	Problem Description . . . . .	9
4.1.1	Terminology and Definitions . . . . .	9
4.1.2	Physical System Description . . . . .	9
4.1.3	Goal Statements . . . . .	11
4.2	Solution Characteristics Specification . . . . .	11
4.2.1	Assumptions . . . . .	11
4.2.2	Theoretical Models . . . . .	13
4.2.3	General Definitions . . . . .	16
4.2.4	Data Definitions . . . . .	20
4.2.5	Instance Models . . . . .	27
4.2.6	Data Constraints . . . . .	35
4.2.7	Properties of a Correct Solution . . . . .	36

<b>5</b>	<b>Requirements</b>	<b>37</b>
5.1	Functional Requirements . . . . .	37
5.2	Non-Functional Requirements . . . . .	38
<b>6</b>	<b>Likely Changes</b>	<b>39</b>
<b>7</b>	<b>Unlikely Changes</b>	<b>39</b>
<b>8</b>	<b>Traceability Matrices and Graphs</b>	<b>40</b>
<b>9</b>	<b>Values of Auxiliary Constants</b>	<b>42</b>
<b>10</b>	<b>References</b>	<b>43</b>

# 1 Reference Material

This section records information for easy reference.

## 1.1 Table of Units

The unit system used throughout is SI (Système International d'Unités). In addition to the basic units, several derived units are also used. For each unit, [Tab: ToU](#) lists the symbol, a description and the SI name.

Symbol	Description	SI Name
°C	temperature	centigrade
J	energy	joule
kg	mass	kilogram
m	length	metre
s	time	second
W	power	watt

Table 1: Table of Units

## 1.2 Table of Symbols

The symbols used in this document are summarized in [Tab: ToS](#) along with their units. The choice of symbols was made to be consistent with the heat transfer literature and with existing documentation for solar water heating systems. The symbols are listed in alphabetical order. For vector quantities, the units shown are for each component of the vector.

Symbol	Description	Units
$A_C$	Heating coil surface area	$m^2$
$A_{in}$	Surface area over which heat is transferred in	$m^2$
$A_{out}$	Surface area over which heat is transferred out	$m^2$
$A_P$	Phase change material surface area	$m^2$
$AR$	Aspect ratio	—
$C$	Specific heat capacity	$\frac{J}{(kg^\circ C)}$
$C^L$	Specific heat capacity of a liquid	$\frac{J}{(kg^\circ C)}$
$C^S$	Specific heat capacity of a solid	$\frac{J}{(kg^\circ C)}$
$C^V$	Specific heat capacity of a vapour	$\frac{J}{(kg^\circ C)}$

Symbol	Description	Units
$C_P^L$	Specific heat capacity of PCM as a liquid	$\frac{J}{(kg^\circ C)}$
$C_P^S$	Specific heat capacity of PCM as a solid	$\frac{J}{(kg^\circ C)}$
$C_{tol}$	Relative tolerance for conservation of energy	—
$C_W$	Specific heat capacity of water	$\frac{J}{(kg^\circ C)}$
$D$	Diameter of tank	m
$E$	Sensible heat	J
$E_P$	Change in heat energy in the PCM	J
$E_W$	Change in heat energy in the water	J
$E_{P_{melt}}^{init}$	Change in heat energy in the PCM at the instant when melting begins	J
$g$	Volumetric heat generation per unit volume	$\frac{W}{m^3}$
$H_f$	Specific latent heat of fusion	$\frac{J}{kg}$
$h$	Convective heat transfer coefficient	$\frac{W}{(m^2^\circ C)}$
$h_C$	Convective heat transfer coefficient between coil and water	$\frac{W}{(m^2^\circ C)}$
$h_{min}$	Minimum thickness of a sheet of PCM	m
$h_P$	Convective heat transfer coefficient between PCM and water	$\frac{W}{(m^2^\circ C)}$
$L$	Length of tank	m
$m$	Mass	kg
$m_P$	Mass of phase change material	kg
$m_W$	Mass of water	kg
$\mathbf{n}$	Unit outward normal vector for a surface	—
$Q$	Latent heat	J
$Q_P$	Latent heat energy added to PCM	J
$q$	Heat flux	$\frac{W}{m^2}$
$q_C$	Heat flux into the water from the coil	$\frac{W}{m^2}$
$q_{in}$	Heat flux input	$\frac{W}{m^2}$
$q_{out}$	Heat flux output	$\frac{W}{m^2}$
$q_P$	Heat flux into the PCM from water	$\frac{W}{m^2}$
$\mathbf{q}$	Thermal flux vector	$\frac{W}{m^2}$
$S$	Surface	$m^2$
$T$	Temperature	$^\circ C$

Symbol	Description	Units
$\Delta T$	Change in temperature	$^{\circ}\text{C}$
$T_{\text{boil}}$	Boiling point temperature	$^{\circ}\text{C}$
$T_{\text{C}}$	Temperature of the heating coil	$^{\circ}\text{C}$
$T_{\text{env}}$	Temperature of the environment	$^{\circ}\text{C}$
$T_{\text{init}}$	Initial temperature	$^{\circ}\text{C}$
$T_{\text{melt}}$	Melting point temperature	$^{\circ}\text{C}$
$T_{\text{melt}}^{\text{P}}$	Melting point temperature for PCM	$^{\circ}\text{C}$
$T_{\text{P}}$	Temperature of the phase change material	$^{\circ}\text{C}$
$T_{\text{W}}$	Temperature of the water	$^{\circ}\text{C}$
$t$	Time	s
$t_{\text{final}}$	Final time	s
$t_{\text{melt}}^{\text{final}}$	Time at which melting of PCM ends	s
$t_{\text{melt}}^{\text{init}}$	Time at which melting of PCM begins	s
$t_{\text{step}}$	Time step for simulation	s
$V$	Volume	$\text{m}^3$
$V_{\text{P}}$	Volume of PCM	$\text{m}^3$
$V_{\text{tank}}$	Volume of the cylindrical tank	$\text{m}^3$
$V_{\text{W}}$	Volume of water	$\text{m}^3$
$\eta$	ODE parameter related to decay rate	—
$\pi$	Ratio of circumference to diameter for any circle	—
$\rho$	Density	$\frac{\text{kg}}{\text{m}^3}$
$\rho_{\text{P}}$	Density of PCM	$\frac{\text{kg}}{\text{m}^3}$
$\rho_{\text{W}}$	Density of water	$\frac{\text{kg}}{\text{m}^3}$
$\tau$	Dummy variable for integration over time	s
$\tau_{\text{P}}^{\text{L}}$	ODE parameter for liquid PCM	s
$\tau_{\text{P}}^{\text{S}}$	ODE parameter for solid PCM	s
$\tau_{\text{W}}$	ODE parameter for water related to decay time	s
$\phi$	Melt fraction	—
$\nabla$	Gradient	—

Table 2: Table of Symbols

## 1.3 Abbreviations and Acronyms

Abbreviation	Full Form
A	Assumption
DD	Data Definition
GD	General Definition
GS	Goal Statement
IM	Instance Model
LC	Likely Change
ODE	Ordinary Differential Equation
PCM	Phase Change Material
PS	Physical System Description
R	Requirement
RHS	Right Hand Side
SRS	Software Requirements Specification
SWHS	Solar Water Heating System
TM	Theoretical Model
UC	Unlikely Change
Uncert.	Typical Uncertainty

Table 3: Abbreviations and Acronyms

## 2 Introduction

Due to increasing costs, diminishing availability, and negative environmental impact of fossil fuels, the demand is high for renewable energy sources and energy storage technology. Solar water heating systems incorporating phase change material (PCM) use a renewable energy source and provide a novel way of storing energy. Solar water heating systems incorporating PCM improve over the traditional solar water heating systems because of their smaller size. The smaller size is possible because of the ability of PCM to store thermal energy as latent heat, which allows higher thermal energy storage capacity per unit weight.

The following section provides an overview of the Software Requirements Specification (SRS) for solar water heating systems incorporating PCM. The developed program will be referred to as Solar Water Heating System (SWHS). This section explains the purpose of this document, the scope of the requirements, the characteristics of the intended reader, and the organization of the document.

### 2.1 Purpose of Document

The main purpose of this document is to describe the modelling of solar water heating systems incorporating PCM. The goal statements and theoretical models used in the SWHS

code are provided, with an emphasis on explicitly identifying assumptions and unambiguous definitions. This document is intended to be used as a reference to provide ad hoc access to all information necessary to understand and verify the model. The SRS is abstract because the contents say what problem is being solved, but do not say how to solve it.

This document will be used as a starting point for subsequent development phases, including writing the design specification and the software verification and validation plan. The design document will show how the requirements are to be realized, including decisions on the numerical algorithms and programming environment. The verification and validation plan will show the steps that will be used to increase confidence in the software documentation and the implementation. Although the SRS fits in a series of documents that follow the so-called waterfall model, the actual development process is not constrained in any way. Even when the waterfall model is not followed, as Parnas and Clements point out [6], the most logical way to present the documentation is still to “fake” a rational design process.

## 2.2 Scope of Requirements

The scope of the requirements includes thermal analysis of a single solar water heating tank incorporating PCM. This entire document is written assuming that the substances inside the solar water heating tank are water and PCM.

## 2.3 Characteristics of Intended Reader

Reviewers of this documentation should have an understanding of heat transfer theory from level 3 or 4 mechanical engineering and differential equations from level 1 and 2 calculus. The users of SWHS can have a lower level of expertise, as explained in [Section: User Characteristics](#).

## 2.4 Organization of Document

The organization of this document follows the template for an SRS for scientific computing software proposed by [5] and [7]. The presentation follows the standard pattern of presenting goals, theories, definitions, and assumptions. For readers that would like a more bottom up approach, they can start reading the instance models in [Section: Instance Models](#) and trace back to find any additional information they require.

The goal statements ([Section: Goal Statements](#)) are refined to the theoretical models and the theoretical models ([Section: Theoretical Models](#)) to the instance models ([Section: Instance Models](#)). The instance models to be solved are referred to as [IM: eBalanceOnWtr](#), [IM: eBalanceOnPCM](#), [IM: heatEInWtr](#), and [IM: heatEInPCM](#). The instance models provide the ordinary differential equations (ODEs) and algebraic equations that model the solar water heating systems incorporating PCM. SWHS solves these ODEs.

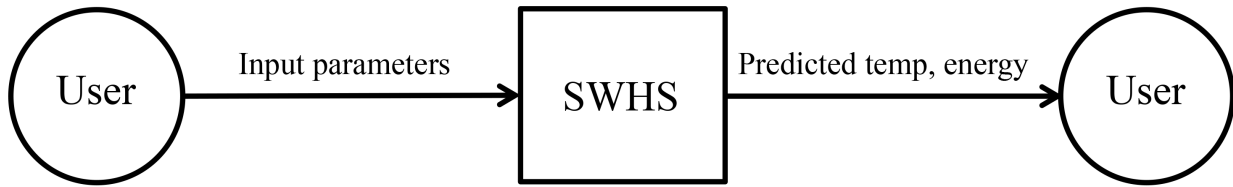


Figure 1: **Fig:SysCon**: System Context

### 3 General System Description

This section provides general information about the system. It identifies the interfaces between the system and its environment, describes the user characteristics, and lists the system constraints.

#### 3.1 System Context

**Fig:SysCon** shows the system context. A circle represents an external entity outside the software, the user in this case. A rectangle represents the software system itself (SWHS). Arrows are used to show the data flow between the system and its environment.

SWHS is mostly self-contained. The only external interaction is through the user interface. The responsibilities of the user and the system are as follows:

- User Responsibilities:
  - Provide the input data to the system, ensuring no errors in the data entry
  - Take care that consistent units are used for input variables
- SWHS Responsibilities:
  - Detect data type mismatch, such as a string of characters instead of a floating point number
  - Determine if the inputs satisfy the required physical and software constraints
  - Calculate the required outputs

#### 3.2 User Characteristics

The end user of SWHS should have an understanding of undergraduate Level 1 Calculus and Physics.

#### 3.3 System Constraints

There are no system constraints.



## 4 Specific System Description

This section first presents the problem description, which gives a high-level view of the problem to be solved. This is followed by the solution characteristics specification, which presents the assumptions, theories, and definitions that are used.

### 4.1 Problem Description

A system is needed to investigate the effect of employing PCM within a solar water heating tank.

#### 4.1.1 Terminology and Definitions

This subsection provides a list of terms that are used in the subsequent sections and their meaning, with the purpose of reducing ambiguity and making it easier to correctly understand the requirements.

- Heat flux: The rate of thermal energy transfer through a given surface per unit time.
- Phase change material: A substance that uses phase changes (such as melting) to absorb or release large amounts of heat at a constant temperature.
- Specific heat capacity: The amount of energy required to raise the temperature of the unit mass of a given substance by a given amount.
- Thermal conduction: The transfer of heat energy through a substance.
- Transient: Changing with time.

#### 4.1.2 Physical System Description

The physical system of SWHS, as shown in **Fig:Tank**, includes the following elements:

PS1: Tank containing water.

PS2: Heating coil at bottom of tank. ( $q_C$  represents the heat flux into the water from the coil.)

PS3: PCM suspended in tank. ( $q_P$  represents the heat flux into the PCM from water.)



Figure 2: Solar water heating tank, with heat flux into the water from the coil of  $q_C$  and heat flux into the PCM from water of  $q_P$

### 4.1.3 Goal Statements

Given the temperature of the heating coil, the initial conditions for the temperature of the water and the temperature of the phase change material, and the material properties, the goal statements are:

Water-Temperature: Predict the temperature of the water over time.

PCM-Temperature: Predict the temperature of the phase change material over time.

Predict-Water-Energy: Predict the change in heat energy in the water over time.

Predict-PCM-Energy: Predict the change in heat energy in the PCM over time.

## 4.2 Solution Characteristics Specification

The instance models that govern SWHS are presented in [Section: Instance Models](#). The information to understand the meaning of the instance models and their derivation is also presented, so that the instance models can be verified.

### 4.2.1 Assumptions

This section simplifies the original problem and helps in developing the theoretical models by filling in the missing information for the physical system. The assumptions refine the scope by providing more detail.

Thermal-Energy-Only: The only form of energy that is relevant for this problem is thermal energy. All other forms of energy, such as mechanical energy, are assumed to be negligible. (RefBy: [TM: consThermE](#).)

Heat-Coeffs-Constant: All heat transfer coefficients are constant over time. (RefBy: [TM: nwtmCooling](#).)

Temp-Across-Tank: The water in the tank is fully mixed, so the temperature of the water is the same throughout the entire tank. (RefBy: [GD: rocTempSimp](#), [IM: eBalanceOnWtr](#), and [IM: eBalanceOnPCM](#).)

Temp-Across-Volume: The temperature of the phase change material is the same throughout the volume of PCM. (RefBy: [GD: rocTempSimp](#), [LC: Uniform-Temperature-PCM](#), [IM: eBalanceOnWtr](#), and [IM: eBalanceOnPCM](#).)

Density-Constant-over-Volume: The density of water and density of PCM have no spatial variation; that is, they are each constant over their entire volume. (RefBy: [GD: rocTempSimp](#).)

Specific-Heat-Constant-over-Volume: The specific heat capacity of water, specific heat capacity of PCM as a solid, and specific heat capacity of PCM as a liquid have no spatial variation; that is, they are each constant over their entire volume. (RefBy: [GD: rocTempSimp](#).)

Cooling-Coil-Water: Newton's law of convective cooling applies between the heating coil and the water. (RefBy: **GD: htFluxWaterFromCoil**.)

Constant-over-Time: The temperature of the heating coil is constant over time. (RefBy: **LC: Temperature-Coil-Variable-Over-Day** and **GD: htFluxWaterFromCoil**.)

Constant-over-Length: The temperature of the heating coil does not vary along its length. (RefBy: **LC: Temperature-Coil-Variable-Over-Length** and **IM: eBalanceOnWtr**.)

Cooling-Water-PCM: Newton's law of convective cooling applies between the water and the PCM. (RefBy: **GD: htFluxPCMFromWater**.)

No-Temp-Discharge: The model only accounts for charging of the tank, not discharging. The temperature of the water and temperature of the phase change material can only increase, or remain constant; they do not decrease. This implies that the initial temperature **A: Same-Initial-Temp-Water-PCM** is less than (or equal) to the temperature of the heating coil. (RefBy: **LC: Discharging-Tank** and **IM: eBalanceOnWtr**.)

Same-Temp-Water-PCM: The initial temperature of the water and the PCM is the same. (RefBy: **LC: Different-Initial-Temps-PCM-Water**, **IM: eBalanceOnWtr**, **IM: eBalanceOnPCM**, and **A: Charging-Tank-No-Temp-Discharge**.)

PCM-Initially-Solid: The simulation will start with the PCM in a solid state. (RefBy: **IM: heatEInPCM** and **IM: eBalanceOnPCM**.)

Water-Always-Liquid: The operating temperature range of the system is such that the water is always in liquid state. That is, the temperature will not drop below the melting point temperature of water, or rise above its boiling point temperature. (RefBy: **UC: Water-PCM-Fixed-States**, **IM: heatEInWtr**, and **IM: eBalanceOnWtr**.)

Perfect-Insulation-Tank: The tank is perfectly insulated so that there is no heat loss from the tank. (RefBy: **LC: Tank-Lose-Heat** and **IM: eBalanceOnWtr**.)

No-Heat-By-Water-PCM: No internal heat is generated by either the water or the PCM; therefore, the volumetric heat generation per unit volume is zero. (RefBy: **UC: No-Internal-Heat-Generation**, **IM: eBalanceOnWtr**, and **IM: eBalanceOnPCM**.)

Negligible-PCM-Negligible: The volume change of the PCM due to melting is negligible. (RefBy: **IM: eBalanceOnPCM**.)

No-Gaseous-State-PCM: The PCM is either in a liquid state or a solid state but not a gaseous state. (RefBy: **UC: Water-PCM-Fixed-States**, **UC: No-Gaseous-State**, **IM: heatEInPCM**, and **IM: eBalanceOnPCM**.)

Atmospheric-Pressure-Tank: The pressure in the tank is atmospheric, so the melting point temperature and boiling point temperature are 0°C and 100°C, respectively. (RefBy: **IM: heatEInWtr** and **IM: eBalanceOnWtr**.)

me-Coil-Negligible: When considering the volume of water in the tank, the volume of the heating coil is assumed to be negligible. (RefBy: [FR: Find-Mass](#).)

### 4.2.2 Theoretical Models

This section focuses on the general equations and laws that SWHS is based on.

Refname	TM:consThermE
Label	Conservation of thermal energy
Equation	$-\nabla \cdot \mathbf{q} + g = \rho C \frac{\partial T}{\partial t}$
Description	<p><math>\nabla</math> is the gradient (Unitless)</p> <p><math>\mathbf{q}</math> is the thermal flux vector (<math>\frac{\text{W}}{\text{m}^2}</math>)</p> <p><math>g</math> is the volumetric heat generation per unit volume (<math>\frac{\text{W}}{\text{m}^3}</math>)</p> <p><math>\rho</math> is the density (<math>\frac{\text{kg}}{\text{m}^3}</math>)</p> <p><math>C</math> is the specific heat capacity (<math>\frac{\text{J}}{(\text{kg}^\circ\text{C})}</math>)</p> <p><math>t</math> is the time (s)</p> <p><math>T</math> is the temperature (<math>^\circ\text{C}</math>)</p>
Notes	<p>The above equation gives the law of conservation of energy for transient heat transfer in a given material.</p> <p>For this equation to apply, other forms of energy, such as mechanical energy, are assumed to be negligible in the system (<a href="#">A: Thermal-Energy-Only</a>).</p>
Source	<a href="#">Fourier Law of Heat Conduction and Heat Equation</a>
RefBy	<a href="#">GD: rocTempSimp</a>

Refname	TM:sensHtE
Label	Sensible heat energy
Equation	$E = \begin{cases} C^S m \Delta T, & T < T_{\text{melt}} \\ C^L m \Delta T, & T_{\text{melt}} < T < T_{\text{boil}} \\ C^V m \Delta T, & T_{\text{boil}} < T \end{cases}$
Description	<p> <math>E</math> is the sensible heat (J)  <math>C^S</math> is the specific heat capacity of a solid (<math>\frac{\text{J}}{(\text{kg}^\circ\text{C})}</math>)  <math>m</math> is the mass (kg)  <math>\Delta T</math> is the change in temperature (<math>^\circ\text{C}</math>)  <math>C^L</math> is the specific heat capacity of a liquid (<math>\frac{\text{J}}{(\text{kg}^\circ\text{C})}</math>)  <math>C^V</math> is the specific heat capacity of a vapour (<math>\frac{\text{J}}{(\text{kg}^\circ\text{C})}</math>)  <math>T</math> is the temperature (<math>^\circ\text{C}</math>)  <math>T_{\text{melt}}</math> is the melting point temperature (<math>^\circ\text{C}</math>)  <math>T_{\text{boil}}</math> is the boiling point temperature (<math>^\circ\text{C}</math>) </p>
Notes	<p> Sensible heating occurs as long as the material does not reach a temperature where a phase change occurs. A phase change occurs if <math>T = T_{\text{boil}}</math> or <math>T = T_{\text{melt}}</math>. If this is the case, refer to <b>TM: latentHtE</b>. </p>
Source	<a href="#">Definition of Sensible Heat</a>
RefBy	<b>IM: heatEInWtr</b> and <b>IM: heatEInPCM</b>

Refname	TM:latentHtE
Label	Latent heat energy
Equation	$Q(t) = \int_0^t \frac{dQ(\tau)}{d\tau} d\tau$
Description	<p><math>Q</math> is the latent heat (J)  <math>t</math> is the time (s)  <math>\tau</math> is the dummy variable for integration over time (s)</p>
Notes	<p><math>Q</math> is the change in thermal energy (latent heat energy).  <math>Q(t) = \int_0^t \frac{dQ(\tau)}{d\tau} d\tau</math> is the rate of change of <math>Q</math> with respect to time <math>\tau</math>.  <math>t</math> is the time elapsed, as long as the phase change is not complete.  The status of the phase change depends on the melt fraction (from <b>DD: meltFrac</b>).  Latent heating stops when all material has changed to the new phase.</p>
Source	<a href="#">Definition of Latent Heat</a>
RefBy	TM: sensHtE and IM: heatEInPCM

Refname	TM:nwtnCooling
Label	Newton's law of cooling
Equation	$q(t) = h\Delta T(t)$
Description	<p><math>q</math> is the heat flux (<math>\frac{W}{m^2}</math>)</p> <p><math>t</math> is the time (s)</p> <p><math>h</math> is the convective heat transfer coefficient (<math>\frac{W}{m^2\cdot C}</math>)</p> <p><math>\Delta T</math> is the change in temperature (<math>^{\circ}C</math>)</p>
Notes	<p>Newton's law of cooling describes convective cooling from a surface. The law is stated as: the rate of heat loss from a body is proportional to the difference in temperatures between the body and its surroundings.</p> <p><math>h</math> is assumed to be independent of <math>T</math> (from <b>A: Heat-Transfer-Coeffs-Constant</b>).</p> <p><math>\Delta T(t) = T(t) - T_{\text{env}}(t)</math> is the time-dependant thermal gradient between the environment and the object.</p>
Source	[2, (pg. 8)]
RefBy	GD: htFluxWaterFromCoil and GD: htFluxPCMFromWater

### 4.2.3 General Definitions

This section collects the laws and equations that will be used to build the instance models.



Refname	GD:rocTempSimp
Label	Simplified rate of change of temperature
Equation	$mC \frac{dT}{dt} = q_{\text{in}} A_{\text{in}} - q_{\text{out}} A_{\text{out}} + gV$
Description	<p> <math>m</math> is the mass (kg)  <math>C</math> is the specific heat capacity (<math>\frac{\text{J}}{\text{kg}^\circ\text{C}}</math>)  <math>t</math> is the time (s)  <math>T</math> is the temperature (<math>^\circ\text{C}</math>)  <math>q_{\text{in}}</math> is the heat flux input (<math>\frac{\text{W}}{\text{m}^2}</math>)  <math>A_{\text{in}}</math> is the surface area over which heat is transferred in (<math>\text{m}^2</math>)  <math>q_{\text{out}}</math> is the heat flux output (<math>\frac{\text{W}}{\text{m}^2}</math>)  <math>A_{\text{out}}</math> is the surface area over which heat is transferred out (<math>\text{m}^2</math>)  <math>g</math> is the volumetric heat generation per unit volume (<math>\frac{\text{W}}{\text{m}^3}</math>)  <math>V</math> is the volume (<math>\text{m}^3</math>) </p>
Source	–
RefBy	GD: rocTempSimp, IM: eBalanceOnWtr, and IM: eBalanceOnPCM

**Detailed derivation of simplified rate of change of temperature:** Integrating **TM: consThermE** over a volume ( $V$ ), we have:

$$-\int_V \nabla \cdot \mathbf{q} dV + \int_V g dV = \int_V \rho C \frac{\partial T}{\partial t} dV$$

Applying Gauss's Divergence Theorem to the first term over the surface  $S$  of the volume, with  $\mathbf{q}$  as the thermal flux vector for the surface and  $\mathbf{n}$  as a unit outward normal vector for a surface:

$$-\int_S \mathbf{q} \cdot \mathbf{n} dS + \int_V g dV = \int_V \rho C \frac{\partial T}{\partial t} dV$$

We consider an arbitrary volume. The volumetric heat generation per unit volume is assumed constant. Then Equation (1) can be written as:

$$q_{\text{in}}A_{\text{in}} - q_{\text{out}}A_{\text{out}} + gV = \int_V \rho C \frac{\partial T}{\partial t} dV$$

Where  $q_{\text{in}}$ ,  $q_{\text{out}}$ ,  $A_{\text{in}}$ , and  $A_{\text{out}}$  are explained in [GD: rocTempSimp](#). The integral over the surface could be simplified because the thermal flux is assumed constant over  $A_{\text{in}}$  and  $A_{\text{out}}$  and 0 on all other surfaces. Outward flux is considered positive. Assuming  $\rho$ ,  $C$ , and  $T$  are constant over the volume, which is true in our case by [A: Constant-Water-Temp-Across-Tank](#), [A: Temp-PCM-Constant-Across-Volume](#), [A: Density-Water-PCM-Constant-over-Volume](#), and [A: Specific-Heat-Energy-Constant-over-Volume](#), we have:

$$\rho CV \frac{dT}{dt} = q_{\text{in}}A_{\text{in}} - q_{\text{out}}A_{\text{out}} + gV$$

Using the fact that  $\rho=m/V$ , Equation (2) can be written as:

$$mC \frac{dT}{dt} = q_{\text{in}}A_{\text{in}} - q_{\text{out}}A_{\text{out}} + gV$$

Refname	GD:htFluxWaterFromCoil
Label	Heat flux into the water from the coil
Units	$\frac{\text{W}}{\text{m}^2}$
Equation	$q_C = h_C (T_C - T_W(t))$
Description	<p> <math>q_C</math> is the heat flux into the water from the coil (<math>\frac{\text{W}}{\text{m}^2}</math>)  <math>h_C</math> is the convective heat transfer coefficient between coil and water  <math>(\frac{\text{W}}{\text{m}^2\text{C}})</math>  <math>T_C</math> is the temperature of the heating coil (<math>^{\circ}\text{C}</math>)  <math>T_W</math> is the temperature of the water (<math>^{\circ}\text{C}</math>)  <math>t</math> is the time (s) </p>
Notes	<p> <math>q_C</math> is found by assuming that Newton's law of cooling applies (<a href="#">A: Newton-Law-Convective-Cooling-Coil-Water</a>). This law (defined in <a href="#">TM: nwtNCooling</a>) is used on the surface of the heating coil.  <a href="#">A: Temp-Heating-Coil-Constant-over-Time</a> </p>
Source	<a href="#">[3]</a>
RefBy	<a href="#">IM: eBalanceOnWtr</a>

Refname	GD:htFluxPCMFromWater
Label	Heat flux into the PCM from water
Units	$\frac{W}{m^2}$
Equation	$q_P = h_P (T_W(t) - T_P(t))$
Description	<p><math>q_P</math> is the heat flux into the PCM from water (<math>\frac{W}{m^2}</math>)</p> <p><math>h_P</math> is the convective heat transfer coefficient between PCM and water (<math>\frac{W}{(m^2 \cdot ^\circ C)}</math>)</p> <p><math>T_W</math> is the temperature of the water (<math>^\circ C</math>)</p> <p><math>t</math> is the time (s)</p> <p><math>T_P</math> is the temperature of the phase change material (<math>^\circ C</math>)</p>
Notes	$q_P$ is found by assuming that Newton's law of cooling applies ( <a href="#">A: Law-Convective-Cooling-Water-PCM</a> ). This law (defined in <a href="#">TM: nwtnCooling</a> ) is used on the surface of the phase change material.
Source	[3]
RefBy	<a href="#">IM: eBalanceOnWtr</a> and <a href="#">IM: eBalanceOnPCM</a>

#### 4.2.4 Data Definitions

This section collects and defines all the data needed to build the instance models.

Refname	DD:balanceDecayRate
Label	ODE parameter for water related to decay time
Symbol	$\tau_W$
Units	s
Equation	$\tau_W = \frac{m_W C_W}{h_C A_C}$
Description	<p> <math>\tau_W</math> is the ODE parameter for water related to decay time (s)  <math>m_W</math> is the mass of water (kg)  <math>C_W</math> is the specific heat capacity of water (<math>\frac{J}{(kg^{\circ}C)}</math>)  <math>h_C</math> is the convective heat transfer coefficient between coil and water  <math>(\frac{W}{(m^2^{\circ}C)})</math>  <math>A_C</math> is the heating coil surface area (m<sup>2</sup>) </p>
Source	<a href="#">[3]</a>
RefBy	FR: Output-Input-Derived-Values and IM: eBalanceOnWtr

Refname	DD:balanceDecayTime
Label	ODE parameter related to decay rate
Symbol	$\eta$
Units	Unitless
Equation	$\eta = \frac{h_P A_P}{h_C A_C}$
Description	<p><math>\eta</math> is the ODE parameter related to decay rate (Unitless)</p> <p><math>h_P</math> is the convective heat transfer coefficient between PCM and water  <math>(\frac{W}{(m^2 \cdot C)})</math></p> <p><math>A_P</math> is the phase change material surface area (m<sup>2</sup>)</p> <p><math>h_C</math> is the convective heat transfer coefficient between coil and water  <math>(\frac{W}{(m^2 \cdot C)})</math></p> <p><math>A_C</math> is the heating coil surface area (m<sup>2</sup>)</p>
Source	<a href="#">[3]</a>
RefBy	FR: Output-Input-Derived-Values and IM: eBalanceOnWtr

Refname	DD:balanceSolidPCM
Label	ODE parameter for solid PCM
Symbol	$\tau_P^S$
Units	s
Equation	$\tau_P^S = \frac{m_P C_P^S}{h_P A_P}$
Description	<p> <math>\tau_P^S</math> is the ODE parameter for solid PCM (s)  <math>m_P</math> is the mass of phase change material (kg)  <math>C_P^S</math> is the specific heat capacity of PCM as a solid (<math>\frac{J}{(kg^\circ C)}</math>)  <math>h_P</math> is the convective heat transfer coefficient between PCM and water  (<math>\frac{W}{(m^2^\circ C)}</math>)  <math>A_P</math> is the phase change material surface area (m<sup>2</sup>) </p>
Source	<a href="#">[4]</a>
RefBy	FR: <a href="#">Output-Input-Derived-Values</a> and IM: <a href="#">eBalanceOnPCM</a>

Refname	DD:balanceLiquidPCM
Label	ODE parameter for liquid PCM
Symbol	$\tau_P^L$
Units	s
Equation	$\tau_P^L = \frac{m_P C_P^L}{h_P A_P}$
Description	<p> <math>\tau_P^L</math> is the ODE parameter for liquid PCM (s)  <math>m_P</math> is the mass of phase change material (kg)  <math>C_P^L</math> is the specific heat capacity of PCM as a liquid (<math>\frac{J}{(kg^\circ C)}</math>)  <math>h_P</math> is the convective heat transfer coefficient between PCM and water  <math>(\frac{W}{(m^2^\circ C)})</math>  <math>A_P</math> is the phase change material surface area (m<sup>2</sup>) </p>
Source	<a href="#">[4]</a>
RefBy	FR: <a href="#">Output-Input-Derived-Values</a> and IM: <a href="#">eBalanceOnPCM</a>



Refname	DD:htFusion
Label	Specific latent heat of fusion
Symbol	$H_f$
Units	$\frac{\text{J}}{\text{kg}}$
Equation	$H_f = \frac{Q}{m}$
Description	<p><math>H_f</math> is the specific latent heat of fusion (<math>\frac{\text{J}}{\text{kg}}</math>)</p> <p><math>Q</math> is the latent heat (J)</p> <p><math>m</math> is the mass (kg)</p>
Notes	The specific latent heat of fusion (also known as the enthalpy of fusion) of a substance is the heat energy required ( $Q$ ) to change the state of a unit of the mass ( $m$ ) of the substance from solid to liquid, at constant pressure.
Source	[1, (pg. 282)]
RefBy	DD: meltFrac and IM: heatEInPCM

Refname	DD:meltFrac
Label	Melt fraction
Symbol	$\phi$
Units	Unitless
Equation	$\phi = \frac{Q_P}{H_f m_P}$
Description	<p><math>\phi</math> is the melt fraction (Unitless)</p> <p><math>Q_P</math> is the latent heat energy added to PCM (J)</p> <p><math>H_f</math> is the specific latent heat of fusion (<math>\frac{J}{kg}</math>)</p> <p><math>m_P</math> is the mass of phase change material (kg)</p>
Notes	<p>The value of <math>\phi</math> is constrained to <math>0 \leq \phi \leq 1</math>.</p> <p>DD: htFusion</p>
Source	[3]
RefBy	TM: latentHtE and IM: eBalanceOnPCM

Refname	DD:aspectRatio
Label	Aspect ratio
Symbol	$AR$
Units	Unitless
Equation	$AR = \frac{D}{L}$
Description	$AR$ is the aspect ratio (Unitless) $D$ is the diameter of tank (m) $L$ is the length of tank (m)
Source	—
RefBy	

#### 4.2.5 Instance Models

This section transforms the problem defined in [Section: Problem Description](#) into one which is expressed in mathematical terms. It uses concrete symbols defined in [Section: Data Definitions](#) to replace the abstract symbols in the models identified in [Section: Theoretical Models](#) and [Section: General Definitions](#).

The goals [GS: Predict-Water-Temperature](#), [GS: Predict-PCM-Temperature](#), [GS: Predict-Water-Energy](#), and [GS: Predict-PCM-Energy](#) are solved by [IM: eBalanceOnWtr](#), [IM: eBalanceOnPCM](#), [IM: heatEinWtr](#), and [IM: heatEinPCM](#). The solutions for [IM: eBalanceOnWtr](#) and [IM: eBalanceOnPCM](#) are coupled since the solutions for  $T_W$  and  $T_P$  depend on one another. [IM: heatEinWtr](#) can be solved once [IM: eBalanceOnWtr](#) has been solved. The solutions of [IM: eBalanceOnPCM](#) and [IM: heatEinPCM](#) are also coupled, since the temperature of the phase change material and the change in heat energy in the PCM depend on

the phase change.

Refname	IM:eBalanceOnWtr
Label	Energy balance on water to find the temperature of the water
Input	$m_W, C_W, h_C, A_P, h_P, A_C, T_P, t_{\text{final}}, T_C, T_{\text{init}}$
Output	$T_W$
Input Constraints	$T_{\text{init}} < T_C$
Output Constraints	$0 \leq t \leq t_{\text{final}}$
Equation	$\frac{dT_W}{dt} = \frac{1}{\tau_W} (T_C - T_W(t) + \eta (T_P(t) - T_W(t)))$
Description	<p><math>t</math> is the time (s)</p> <p><math>T_W</math> is the temperature of the water (°C)</p> <p><math>\tau_W</math> is the ODE parameter for water related to decay time (s)</p> <p><math>T_C</math> is the temperature of the heating coil (°C)</p> <p><math>\eta</math> is the ODE parameter related to decay rate (Unitless)</p> <p><math>T_P</math> is the temperature of the phase change material (°C)</p>
Notes	<p><math>T_P</math> is defined by <b>IM: eBalanceOnPCM</b>.</p> <p>The input constraint <math>T_{\text{init}} \leq T_C</math> comes from <b>A: Charging-Tank-No-Temp-Discharge</b>.</p> <p><math>\tau_W</math> is calculated from <b>DD: balanceDecayRate</b>.</p> <p><math>\eta</math> is calculated from <b>DD: balanceDecayTime</b>.</p> <p>The initial conditions for the ODE are <math>T_W(0) = T_P(0) = T_{\text{init}}</math> following <b>A: Same-Initial-Temp-Water-PCM</b>.</p> <p>The ODE applies as long as the water is in liquid form, <math>0 &lt; T_W &lt; 100</math> (°C) where 0 (°C) and 100 (°C) are the melting and boiling point temperatures of water, respectively (from <b>A: Water-Always-Liquid</b> and <b>A: Atmospheric-Pressure-Tank</b>).</p>

**Detailed derivation of the energy balance on water:** To find the rate of change of  $T_W$ , we look at the energy balance on water. The volume being considered is the volume of water in the tank  $V_W$ , which has mass  $m_W$  and specific heat capacity,  $C_W$ . Heat transfer occurs in the water from the heating coil as  $q_C$  (**GD: htFluxWaterFromCoil**) and from the water into the PCM as  $q_P$  (**GD: htFluxPCMFromWater**), over areas  $A_C$  and  $A_P$ , respectively. The thermal flux is constant over  $A_C$ , since the temperature of the heating coil is assumed to not vary along its length (**A: Temp-Heating-Coil-Constant-over-Length**), and the thermal flux is constant over  $A_P$ , since the temperature of the PCM is the same throughout its volume (**A: Temp-PCM-Constant-Across-Volume**) and the water is fully mixed (**A: Constant-Water-Temp-Across-Tank**). No heat transfer occurs to the outside of the tank, since it has been assumed to be perfectly insulated (**A: Perfect-Insulation-Tank**). Since the assumption is made that no internal heat is generated (**A: No-Internal-Heat-Generation-By-Water-PCM**),  $g = 0$ . Therefore, the equation for **GD: rocTempSimp** can be written as:

$$m_W C_W \frac{dT_W}{dt} = q_C A_C - q_P A_P$$

Using **GD: htFluxWaterFromCoil** for  $q_C$  and **GD: htFluxPCMFromWater** for  $q_P$ , this can be written as:

$$m_W C_W \frac{dT_W}{dt} = h_C A_C (T_C - T_W) - h_P A_P (T_W - T_P)$$

Dividing Equation (2) by  $m_W C_W$ , we obtain:

$$\frac{dT_W}{dt} = \frac{h_C A_C}{m_W C_W} (T_C - T_W) - \frac{h_P A_P}{m_W C_W} (T_W - T_P)$$

Factoring the negative sign out of the second term of the RHS of Equation (3) and multiplying it by  $h_C A_C / h_C A_C$  yields:

$$\frac{dT_W}{dt} = \frac{h_C A_C}{m_W C_W} (T_C - T_W) + \frac{h_C A_C}{h_C A_C} \frac{h_P A_P}{m_W C_W} (T_P - T_W)$$

Rearranging this equation gives us:

$$\frac{dT_W}{dt} = \frac{h_C A_C}{m_W C_W} (T_C - T_W) + \frac{h_P A_P}{h_C A_C} \frac{h_C A_C}{m_W C_W} (T_P - T_W)$$

By substituting  $\tau_W$  (from **DD: balanceDecayRate**) and  $\eta$  (from **DD: balanceDecayTime**), this can be written as:

$$\frac{dT_W}{dt} = \frac{1}{\tau_W} (T_C - T_W) + \frac{\eta}{\tau_W} (T_P - T_W)$$

Finally, factoring out  $\frac{1}{\tau_W}$ , we are left with the governing ODE for **IM: eBalanceOnWtr**:

$$\frac{dT_W}{dt} = \frac{1}{\tau_W} (T_C - T_W + \eta (T_P - T_W))$$

Refname	IM:eBalanceOnPCM
Label	Energy Balance on PCM to find temperature of PCM
Input	$T_{\text{melt}}^{\text{P}}, t_{\text{final}}, T_{\text{init}}, A_{\text{P}}, h_{\text{P}}, m_{\text{P}}, C_{\text{P}}^{\text{S}}, C_{\text{P}}^{\text{L}}$
Output	$T_{\text{P}}$
Input Constraints	$T_{\text{init}} < T_{\text{melt}}^{\text{P}}$
Output Constraints	$0 \leq t \leq t_{\text{final}}$
Equation	$\frac{dT_{\text{P}}}{dt} = \begin{cases} \frac{1}{\tau_{\text{P}}^{\text{S}}} (T_{\text{W}}(t) - T_{\text{P}}(t)), & T_{\text{P}} < T_{\text{melt}}^{\text{P}} \\ \frac{1}{\tau_{\text{P}}^{\text{L}}} (T_{\text{W}}(t) - T_{\text{P}}(t)), & T_{\text{P}} > T_{\text{melt}}^{\text{P}} \\ 0, & T_{\text{P}} = T_{\text{melt}}^{\text{P}} \wedge 0 < \phi < 1 \end{cases}$
Description	<p><math>t</math> is the time (s)</p> <p><math>T_{\text{P}}</math> is the temperature of the phase change material (<math>^{\circ}\text{C}</math>)</p> <p><math>\tau_{\text{P}}^{\text{S}}</math> is the ODE parameter for solid PCM (s)</p> <p><math>T_{\text{W}}</math> is the temperature of the water (<math>^{\circ}\text{C}</math>)</p> <p><math>\tau_{\text{P}}^{\text{L}}</math> is the ODE parameter for liquid PCM (s)</p> <p><math>T_{\text{melt}}^{\text{P}}</math> is the melting point temperature for PCM (<math>^{\circ}\text{C}</math>)</p> <p><math>\phi</math> is the melt fraction (Unitless)</p>
Notes	<p><math>T_{\text{W}}</math> is defined by <a href="#">IM: eBalanceOnWtr</a>.</p> <p>The input constraint <math>T_{\text{init}} \leq T_{\text{melt}}^{\text{P}}</math> comes from <a href="#">A: PCM-Initially-Solid</a>.</p> <p>The temperature remains constant at <math>T_{\text{melt}}^{\text{P}}</math>, even with the heating (or cooling), until the phase change has occurred for all of the material; that is as long as <math>0 &lt; \phi &lt; 1</math>. <math>\phi</math> (from <a href="#">DD: meltFrac</a>) is determined as part of the heat energy in the PCM, as given in (<a href="#">IM: heatEInPCM</a>).</p> <p><math>\tau_{\text{P}}^{\text{S}}</math> is calculated in <a href="#">DD: balanceSolidPCM</a>.</p> <p><math>\tau_{\text{P}}^{\text{L}}</math> is calculated in <a href="#">DD: balanceLiquidPCM</a>.</p> <p>The initial conditions for the ODE are <math>T_{\text{W}}(0) = T_{\text{P}}(0) = T_{\text{init}}</math> following <a href="#">A: Same-Initial-Temp-Water-PCM</a>.</p>

**Detailed derivation of the energy balance on the PCM during sensible heating phase:** To find the rate of change of  $T_P$ , we look at the energy balance on the PCM. The volume being considered is the volume of PCM ( $V_P$ ). The derivation that follows is initially for the solid PCM. The mass of phase change material is  $m_P$  and the specific heat capacity of PCM as a solid is  $C_P^S$ . The heat flux into the PCM from water is  $q_P$  (GD: htFluxPCMFromWater) over phase change material surface area  $A_P$ . The thermal flux is constant over  $A_P$ , since the temperature of the PCM is the same throughout its volume (A: Temp-PCM-Constant-Across-Volume) and the water is fully mixed (A: Constant-Water-Temp-Across-Tank). There is no heat flux output from the PCM. Assuming no volumetric heat generation per unit volume (A: No-Internal-Heat-Generation-By-Water-PCM),  $g = 0$ , the equation for GD: rocTempSimp can be written as:

$$m_P C_P^S \frac{dT_P}{dt} = q_P A_P$$

Using GD: htFluxPCMFromWater for  $q_P$ , this equation can be written as:

$$m_P C_P^S \frac{dT_P}{dt} = h_P A_P (T_W - T_P)$$

Dividing by  $m_P C_P^S$  we obtain:

$$\frac{dT_P}{dt} = \frac{h_P A_P}{m_P C_P^S} (T_W - T_P)$$

By substituting  $\tau_P^S$  (from DD: balanceSolidPCM), this can be written as:

$$\frac{dT_P}{dt} = \frac{1}{\tau_P^S} (T_W - T_P)$$

Equation (4) applies for the solid PCM. In the case where all of the PCM is melted, the same derivation applies, except that  $C_P^S$  is replaced by  $C_P^L$ , and thus  $\tau_P^S$  is replaced by  $\tau_P^L$ . Although a small change in surface area would be expected with melting, this is not included, since the volume change of the PCM with melting is assumed to be negligible (A: Volume-Change-Melting-PCM-Negligible).

In the case where  $T_P = T_{\text{melt}}^P$  and not all of the PCM is melted, the temperature of the phase change material does not change. Therefore,  $d T_P / d t = 0$ .

This derivation does not consider the boiling of the PCM, as the PCM is assumed to either be in a solid state or a liquid state (A: No-Gaseous-State-PCM).



Refname	IM:heatEInWtr
Label	Heat energy in the water
Input	$T_{\text{init}}, m_{\text{W}}, C_{\text{W}}, m_{\text{W}}$
Output	$E_{\text{W}}$
Input Constraints	
Output Constraints	$0 \leq t \leq t_{\text{final}}$
Equation	$E_{\text{W}}(t) = C_{\text{W}}m_{\text{W}}(T_{\text{W}}(t) - T_{\text{init}})$
Description	<p><math>E_{\text{W}}</math> is the change in heat energy in the water (J)</p> <p><math>t</math> is the time (s)</p> <p><math>C_{\text{W}}</math> is the specific heat capacity of water (<math>\frac{\text{J}}{\text{kg}^{\circ}\text{C}}</math>)</p> <p><math>m_{\text{W}}</math> is the mass of water (kg)</p> <p><math>T_{\text{W}}</math> is the temperature of the water (<math>^{\circ}\text{C}</math>)</p> <p><math>T_{\text{init}}</math> is the initial temperature (<math>^{\circ}\text{C}</math>)</p>
Notes	<p>The above equation is derived using <b>TM: sensHtE</b>.</p> <p>The change in temperature is the difference between the temperature at time <math>t</math> (s), <math>T_{\text{W}}</math> and the initial temperature, <math>T_{\text{init}}</math> (<math>^{\circ}\text{C}</math>).</p> <p>This equation applies as long as <math>0 &lt; T_{\text{W}} &lt; 100^{\circ}\text{C}</math> (<b>A: Water-Always-Liquid, A: Atmospheric-Pressure-Tank</b>).</p>
Source	[3]
RefBy	<b>FR: Find-Mass</b> and <b>FR: Calculate-Change-Heat_Energy-Water-Over-Time</b>

Refname	IM:heatEInPCM
Label	Heat energy in the PCM
Input	$T_{\text{melt}}^{\text{P}}, t_{\text{final}}, T_{\text{init}}, A_{\text{P}}, h_{\text{P}}, m_{\text{P}}, C_{\text{P}}^{\text{S}}, C_{\text{P}}^{\text{L}}, T_{\text{P}}, H_{\text{f}}, t_{\text{melt}}^{\text{init}}$
Output	$E_{\text{P}}$
Input Constraints	$T_{\text{init}} < T_{\text{melt}}^{\text{P}}$
Output Constraints	$0 \leq t \leq t_{\text{final}}$
Equation	$E_{\text{P}} = \begin{cases} C_{\text{P}}^{\text{S}} m_{\text{P}} (T_{\text{P}}(t) - T_{\text{init}}), & T_{\text{P}} < T_{\text{melt}}^{\text{P}} \\ E_{\text{P}_{\text{melt}}}^{\text{init}} + H_{\text{f}} m_{\text{P}} + C_{\text{P}}^{\text{L}} m_{\text{P}} (T_{\text{P}}(t) - T_{\text{melt}}^{\text{P}}), & T_{\text{P}} > T_{\text{melt}}^{\text{P}} \\ E_{\text{P}_{\text{melt}}}^{\text{init}} + Q_{\text{P}}(t), & T_{\text{P}} = T_{\text{melt}}^{\text{P}} \wedge 0 < \phi < 1 \end{cases}$
Description	<p> <math>E_{\text{P}}</math> is the change in heat energy in the PCM (J)  <math>C_{\text{P}}^{\text{S}}</math> is the specific heat capacity of PCM as a solid (<math>\frac{\text{J}}{\text{kg}^{\circ}\text{C}}</math>)  <math>m_{\text{P}}</math> is the mass of phase change material (kg)  <math>T_{\text{P}}</math> is the temperature of the phase change material (<math>^{\circ}\text{C}</math>)  <math>t</math> is the time (s)  <math>T_{\text{init}}</math> is the initial temperature (<math>^{\circ}\text{C}</math>)  <math>E_{\text{P}_{\text{melt}}}^{\text{init}}</math> is the change in heat energy in the PCM at the instant when melting begins (J)  <math>H_{\text{f}}</math> is the specific latent heat of fusion (<math>\frac{\text{J}}{\text{kg}}</math>)  <math>C_{\text{P}}^{\text{L}}</math> is the specific heat capacity of PCM as a liquid (<math>\frac{\text{J}}{\text{kg}^{\circ}\text{C}}</math>)  <math>T_{\text{melt}}^{\text{P}}</math> is the melting point temperature for PCM (<math>^{\circ}\text{C}</math>)  <math>Q_{\text{P}}</math> is the latent heat energy added to PCM (J)  <math>\phi</math> is the melt fraction (Unitless) </p>
Notes	<p>The above equation is derived using <b>TM: sensHtE</b> and <b>TM: latentHtE</b>. <math>E_{\text{P}}</math> for the solid PCM is found using <b>TM: sensHtE</b> for sensible heating, with the specific heat capacity of the solid PCM, <math>C_{\text{P}}^{\text{S}}</math> (<math>\frac{\text{J}}{\text{kg}^{\circ}\text{C}}</math>) and the</p>

#### 4.2.6 Data Constraints

**Table:InDataConstraints** shows the data constraints on the input variables. The column for physical constraints gives the physical limitations on the range of values that can be taken by the variable. The uncertainty column provides an estimate of the confidence with which the physical quantities can be measured. This information would be part of the input if one were performing an uncertainty quantification exercise. The constraints are conservative, to give the user of the model the flexibility to experiment with unusual situations. The column of typical values is intended to provide a feel for a common scenario. The column for software constraints restricts the range of inputs to reasonable values. (\*) These quantities cannot be equal to zero, or there will be a divide by zero in the model. (+) These quantities cannot be zero, or there would be freezing (**A: PCM-Initially-Solid**). (++) The constraints on the surface area are calculated by considering the surface area to volume ratio. The assumption is that the lowest ratio is 1 and the highest possible is  $\frac{2}{h_{\min}}$ , where  $h_{\min}$  is the thickness of a “sheet” of PCM. A thin sheet has the greatest surface area to volume ratio. (\*\*) The constraint on the maximum time at the end of the simulation is the total number of seconds in one day.

Var	Physical Constraints	Software Constraints	Typical Value	Uncert.
$A_C$	$A_C > 0$	$A_C \leq A_C^{\max}$	$0.12 \text{ m}^2$	10%
$A_P$	$A_P > 0$	$V_P \leq A_P \leq \frac{2}{h_{\min}} V_{\text{tank}}$	$1.2 \text{ m}^2$	10%
$C_P^L$	$C_P^L > 0$	$C_{P_{\min}}^L < C_P^L < C_{P_{\max}}^L$	$2.27 \cdot 10^3 \frac{\text{J}}{(\text{kg}^\circ\text{C})}$	10%
$C_P^S$	$C_P^S > 0$	$C_{P_{\min}}^S < C_P^S < C_{P_{\max}}^S$	$1.76 \cdot 10^3 \frac{\text{J}}{(\text{kg}^\circ\text{C})}$	10%
$C_W$	$C_W > 0$	$C_W^{\min} < C_W < C_W^{\max}$	$4.186 \cdot 10^3 \frac{\text{J}}{(\text{kg}^\circ\text{C})}$	10%
$D$	$D > 0$	$AR_{\min} \leq D \leq AR_{\max}$	$0.412 \text{ m}$	10%
$H_f$	$H_f > 0$	$H_{f_{\min}} < H_f < H_{f_{\max}}$	$211.6 \cdot 10^3 \frac{\text{J}}{\text{kg}}$	10%
$h_C$	$h_C > 0$	$h_C^{\min} \leq h_C \leq h_C^{\max}$	$1.0 \cdot 10^3 \frac{\text{W}}{(\text{m}^2\cdot\text{C})}$	10%
$h_P$	$h_P > 0$	$h_P^{\min} \leq h_P \leq h_P^{\max}$	$1.0 \cdot 10^3 \frac{\text{W}}{(\text{m}^2\cdot\text{C})}$	10%
$L$	$L > 0$	$L_{\min} \leq L \leq L_{\max}$	$1.5 \text{ m}$	10%
$T_C$	$0 < T_C < 100$	–	$50.0 \text{ }^\circ\text{C}$	10%
$T_{\text{init}}$	$0 < T_{\text{init}} < T_{\text{melt}}$	–	$40.0 \text{ }^\circ\text{C}$	10%
$T_{\text{melt}}^P$	$0 < T_{\text{melt}}^P < T_C$	–	$44.2 \text{ }^\circ\text{C}$	10%
$t_{\text{final}}$	$t_{\text{final}} > 0$	$t_{\text{final}} < t_{\text{final}}^{\max}$	$50.0 \cdot 10^3 \text{ s}$	10%
$t_{\text{step}}$	$0 < t_{\text{step}} < t_{\text{final}}$	–	$0.01 \text{ s}$	10%

Var	Physical Constraints	Software Constraints	Typical Value	Uncert.
$V_P$	$0 < V_P < V_{\text{tank}}$	$V_P \geq \text{MINFRAC}V_{\text{tank}}$	$0.05 \text{ m}^3$	10%
$\rho_P$	$\rho_P > 0$	$\rho_P^{\min} < \rho_P < \rho_P^{\max}$	$1.007 \cdot 10^3 \frac{\text{kg}}{\text{m}^3}$	10%
$\rho_W$	$\rho_W > 0$	$\rho_W^{\min} < \rho_W \leq \rho_W^{\max}$	$1.0 \cdot 10^3 \frac{\text{kg}}{\text{m}^3}$	10%

Table 4: Input Data Constraints

#### 4.2.7 Properties of a Correct Solution

**Table:OutDataConstraints** shows the data constraints on the output variables. The column for physical constraints gives the physical limitations on the range of values that can be taken by the variable.

Var	Physical Constraints
$T_W$	$T_{\text{init}} \leq T_W \leq T_C$
$T_P$	$T_{\text{init}} \leq T_P \leq T_C$
$E_W$	$E_W \geq 0$
$E_P$	$E_P \geq 0$

Table 5: Output Data Constraints

A correct solution must exhibit the law of conservation of energy. This means that the change in heat energy in the water should equal the difference between the total energy input from the heating coil and the energy output to the PCM. This can be shown as an equation by taking **GD: htFluxWaterFromCoil** and **GD: htFluxPCMFromWater**, multiplying each by their respective surface area of heat transfer, and integrating each over the simulation time, as follows:

$$E_W = \int_0^t h_C A_C (T_C - T_W(t)) dt - \int_0^t h_P A_P (T_W(t) - T_P(t)) dt$$

In addition, the change in heat energy in the PCM should equal the energy input to the PCM from the water. This can be expressed as

$$E_P = \int_0^t h_P A_P (T_W(t) - T_P(t)) dt$$

Equations (FIXME: Equation 7) and (FIXME: Equation 8) can be used as “sanity” checks to gain confidence in any solution computed by SWHS. The relative error between the results computed by SWHS and the results calculated from the RHS of these equations should be less than  $C_{\text{tol}}$  **FR: Verify-Energy-Output-Follow-Conservation-of-Energy**.

## 5 Requirements

This section provides the functional requirements, the tasks and behaviours that the software is expected to complete, and the non-functional requirements, the qualities that the software is expected to exhibit.

### 5.1 Functional Requirements

This section provides the functional requirements, the tasks and behaviours that the software is expected to complete.

**Input-Values:** Input the values from [Table:ReqInputs](#), which define the tank parameters, material properties, and initial conditions.

**Find-Mass:** Use the inputs in [FR: Input-Values](#) to find the masses needed for [IM: eBalanceOnWtr](#), [IM: eBalanceOnPCM](#), [IM: heatEInWtr](#), and [IM: heatEInPCM](#), using  $m_W = V_W \rho_W = (V_{\text{tank}} - V_P) \rho_W = \left( \pi \left( \frac{D}{2} \right)^2 L - V_P \right) \rho_W$ ,  $m_P = V_P \rho_P$ , and [A: Volume-Coil-Negligible](#), where  $V_W$  is the volume of water and  $V_{\text{tank}}$  is the volume of the cylindrical tank.

**Physical\_Constraints:** Verify that the inputs satisfy the required physical constraints shown in [Section: Data Constraints](#).

**Output-Derived-Values:** Output the input values and derived values in the following list: the values from [FR: Input-Values](#), the masses from [FR: Find-Mass](#),  $\tau_W$  (from [DD: balanceDecayRate](#)),  $\eta$  (from [DD: balanceDecayTime](#)),  $\tau_P^S$  (from [DD: balanceSolidPCM](#)), and  $\tau_P^L$  (from [DD: balanceLiquidPCM](#)).

**Water-Over-Time:** Calculate and output the temperature of the water ( $T_W(t)$ ) over the simulation time (from [IM: eBalanceOnWtr](#)).

**PCM-Over-Time:** Calculate and output the temperature of the phase change material ( $T_P(t)$ ) over the simulation time (from [IM: eBalanceOnPCM](#)).

**Water-Over-Time:** Calculate and output the change in heat energy in the water ( $E_W(t)$ ) over the simulation time (from [IM: heatEInWtr](#)).

**PCM-Over-Time:** Calculate and output the change in heat energy in the PCM ( $E_P(t)$ ) over the simulation time (from [IM: heatEInPCM](#)).

**Conservation-of-Energy:** Verify that the energy outputs ( $E_W(t)$  and  $E_P(t)$ ) follow the law of conservation of energy, as outlined in [Section: Properties of a Correct Solution](#), with relative error no greater than  $C_{\text{tol}}$ .

**Melt-Begin-Time:** Calculate and output the time at which the PCM begins to melt  $t_{\text{melt}}^{\text{init}}$  (from [IM: eBalanceOnPCM](#)).

M-Melt-End-Time: Calculate and output the time at which the PCM stops melting  $t_{\text{melt}}^{\text{final}}$  (from **IM: eBalanceOnPCM**).

Symbol	Description	Units
$A_C$	Heating coil surface area	$\text{m}^2$
$A_P$	Phase change material surface area	$\text{m}^2$
$A_{\text{tol}}$	Absolute tolerance	—
$C_P^L$	Specific heat capacity of PCM as a liquid	$\frac{\text{J}}{(\text{kg}^\circ\text{C})}$
$C_P^S$	Specific heat capacity of PCM as a solid	$\frac{\text{J}}{(\text{kg}^\circ\text{C})}$
$C_W$	Specific heat capacity of water	$\frac{\text{J}}{(\text{kg}^\circ\text{C})}$
$D$	Diameter of tank	$\text{m}$
$H_f$	Specific latent heat of fusion	$\frac{\text{J}}{\text{kg}}$
$h_C$	Convective heat transfer coefficient between coil and water	$\frac{\text{W}}{(\text{m}^2^\circ\text{C})}$
$h_P$	Convective heat transfer coefficient between PCM and water	$\frac{\text{W}}{(\text{m}^2^\circ\text{C})}$
$L$	Length of tank	$\text{m}$
$R_{\text{tol}}$	Relative tolerance	—
$T_C$	Temperature of the heating coil	$^\circ\text{C}$
$T_{\text{init}}$	Initial temperature	$^\circ\text{C}$
$T_{\text{melt}}^P$	Melting point temperature for PCM	$^\circ\text{C}$
$t_{\text{final}}$	Final time	$\text{s}$
$t_{\text{step}}$	Time step for simulation	$\text{s}$
$V_P$	Volume of PCM	$\text{m}^3$
$\rho_P$	Density of PCM	$\frac{\text{kg}}{\text{m}^3}$
$\rho_W$	Density of water	$\frac{\text{kg}}{\text{m}^3}$

Table 6: Required Inputs following **FR: Input-Values**

## 5.2 Non-Functional Requirements

This section provides the non-functional requirements, the qualities that the software is expected to exhibit.

Correct: The outputs of the code have the properties described in **Section: Properties of a Correct Solution**.

Verifiable: The code is tested with complete verification and validation plan.

Understandable: The code is modularized with complete module guide and module interface specification.

Reusable: The code is modularized.

Maintainable: The traceability between requirements, assumptions, theoretical models, general definitions, data definitions, instance models, likely changes, unlikely changes, and modules is completely recorded in traceability matrices in the SRS and module guide.

## 6 Likely Changes

This section lists the likely changes to be made to the software.

Temperature-PCM: **A: Temp-PCM-Constant-Across-Volume** - PCM is actually a poor thermal conductor, so the assumption of uniform temperature of the phase change material is not likely.

Variable-Over-Day: **A: Temp-Heating-Coil-Constant-over-Time** - The temperature of the heating coil will change over the course of the day, depending on the energy received from the sun.

Variable-Over-Length: **A: Temp-Heating-Coil-Constant-over-Length** - The temperature of the heating coil will actually change along its length as the water within it cools.

Discharging-Tank: **A: Charging-Tank-No-Temp-Discharge** - The model currently only accounts for charging of the tank. A more complete model would also account for discharging of the tank.

Temps-PCM-Water: **A: Same-Initial-Temp-Water-PCM** - To add more flexibility to the simulation, the initial temperature of the water and the PCM could be allowed to have different values.

Tank-Lose-Heat: **A: Perfect-Insulation-Tank** - Any real tank cannot be perfectly insulated and will lose heat.

## 7 Unlikely Changes

This section lists the unlikely changes to be made to the software.

PCM-Fixed-States: **A: Water-Always-Liquid**, **A: No-Gaseous-State-PCM** - It is unlikely for the change of water from liquid to a solid or the state change of the phase change material from a liquid to a gas to be considered.

l-Heat-Generation: **A: No-Internal-Heat-Generation-By-Water-PCM** - Is used for the derivations of **IM: eBalanceOnWtr** and **IM: eBalanceOnPCM**.

No-Gaseous-State: **A: No-Gaseous-State-PCM** - Is used for the derivation of **IM: eBalanceOnPCM** and for the equation given by **IM: heatEInPCM** to be valid.

## 8 Traceability Matrices and Graphs

The purpose of the traceability matrices is to provide easy references on what has to be additionally modified if a certain component is changed. Every time a component is changed, the items in the column of that component that are marked with an “X” should be modified as well. [Table:TraceMatAvsAll](#) shows the dependencies of data definitions, theoretical models, general definitions, instance models, requirements, likely changes, and unlikely changes on the assumptions. [Table:TraceMatRefvsRef](#) shows the dependencies of data definitions, theoretical models, general definitions, and instance models with each other. [Table:TraceMatAllvsR](#) shows the dependencies of requirements, goal statements on the data definitions, theoretical models, general definitions, and instance models.

	A: Thermal-Energy-Only	A: Heat-Tr
DD: balanceDecayRate		
DD: balanceDecayTime		
DD: balanceSolidPCM		
DD: balanceLiquidPCM		
DD: htFusion		
DD: meltFrac		
DD: aspectRatio		
TM: consThermE	X	
TM: sensHtE		
TM: latentHtE		
TM: nwtmCooling		X
GD: rocTempSimp		
GD: htFluxWaterFromCoil		
GD: htFluxPCMFromWater		
IM: eBalanceOnWtr		
IM: eBalanceOnPCM		
IM: heatEInWtr		
IM: heatEInPCM		
FR: Input-Values		
FR: Find-Mass		
FR: Check-Input-with-Physical_Constraints		
FR: Output-Input-Derived-Values		
FR: Calculate-Temperature-Water-Over-Time		
FR: Calculate-Temperature-PCM-Over-Time		
FR: Calculate-Change-Heat_Energy-Water-Over-Time		
FR: Calculate-Change-Heat_Energy-PCM-Over-Time		
FR: Verify-Energy-Output-Follow-Conservation-of-Energy		
FR: Calculate-PCM-Melt-Begin-Time		
FR: Calculate-PCM-Melt-End-Time		
NFR: Correct		



	A: Thermal-Energy-Only		A: Heat-Tr
NFR: Verifiable			
NFR: Understandable			
NFR: Reusable			
NFR: Maintainable			
LC: Uniform-Temperature-PCM			
LC: Temperature-Coil-Variable-Over-Day			
LC: Temperature-Coil-Variable-Over-Length			
LC: Discharging-Tank			
LC: Different-Initial-Temps-PCM-Water			
LC: Tank-Lose-Heat			
UC: Water-PCM-Fixed-States			
UC: No-Internal-Heat-Generation			
UC: No-Gaseous-State			
	DD: balanceDecayRate	DD: balanceDecayTime	DD: balanceS
DD: balanceDecayRate			
DD: balanceDecayTime			
DD: balanceSolidPCM			
DD: balanceLiquidPCM			
DD: htFusion			
DD: meltFrac			
DD: aspectRatio			
TM: consThermE			
TM: sensHtE			
TM: latentHtE			
TM: nwtNCooling			
GD: rocTempSimp			
GD: htFluxWaterFromCoil			
GD: htFluxPCMFromWater			
IM: eBalanceOnWtr	X	X	
IM: eBalanceOnPCM			X
IM: heatEInWtr			
IM: heatEInPCM			

	DD: balanceDecayRate	DD: balanceI
GS: Predict-Water-Temperature		
GS: Predict-PCM-Temperature		
GS: Predict-Water-Energy		
GS: Predict-PCM-Energy		
FR: Input-Values		
FR: Find-Mass		
FR: Check-Input-with-Physical_Constraints		
FR: Output-Input-Derived-Values	X	X
FR: Calculate-Temperature-Water-Over-Time		
FR: Calculate-Temperature-PCM-Over-Time		
FR: Calculate-Change-Heat_Energy-Water-Over-Time		
FR: Calculate-Change-Heat_Energy-PCM-Over-Time		
FR: Verify-Energy-Output-Follow-Conservation-of-Energy		
FR: Calculate-PCM-Melt-Begin-Time		
FR: Calculate-PCM-Melt-End-Time		
NFR: Correct		
NFR: Verifiable		
NFR: Understandable		
NFR: Reusable		
NFR: Maintainable		

## 9 Values of Auxiliary Constants

This section contains the standard values that are used for calculations in SWHS.

Symbol	Description	Value	Unit
$A_C^{\max}$	maximum surface area of coil	100000	m <sup>2</sup>
$AR_{\max}$	maximum aspect ratio	100	—
$AR_{\min}$	minimum aspect ratio	0.01	—
$C_{\text{tol}}$	relative tolerance for conservation of energy	0.001%	—
$C_W^{\max}$	maximum specific heat capacity of water	4210	$\frac{\text{J}}{(\text{kg}^\circ\text{C})}$
$C_W^{\min}$	minimum specific heat capacity of water	4170	$\frac{\text{J}}{(\text{kg}^\circ\text{C})}$
$C_P^L_{\max}$	maximum specific heat capacity of PCM as a liquid	5000	$\frac{\text{J}}{(\text{kg}^\circ\text{C})}$
$C_P^L_{\min}$	minimum specific heat capacity of PCM as a liquid	100	$\frac{\text{J}}{(\text{kg}^\circ\text{C})}$

Symbol	Description	Value	Unit
$C_{\text{P}}^{\text{S}}_{\text{max}}$	maximum specific heat capacity of PCM as a solid	4000	$\frac{\text{J}}{(\text{kg}^\circ\text{C})}$
$C_{\text{P}}^{\text{S}}_{\text{min}}$	minimum specific heat capacity of PCM as a solid	100	$\frac{\text{J}}{(\text{kg}^\circ\text{C})}$
$H_{\text{fmax}}$	maximum specific latent heat of fusion	1000000	$\frac{\text{J}}{(\text{kg}^\circ\text{C})}$
$H_{\text{fmin}}$	minimum specific latent heat of fusion	0	$\frac{\text{J}}{(\text{kg}^\circ\text{C})}$
$h_{\text{C}}^{\text{max}}$	maximum convective heat transfer coefficient between coil and water	10000	$\frac{\text{W}}{(\text{m}^2\text{C})}$
$h_{\text{C}}^{\text{min}}$	minimum convective heat transfer coefficient between coil and water	10	$\frac{\text{W}}{(\text{m}^2\text{C})}$
$h_{\text{P}}^{\text{max}}$	maximum convective heat transfer coefficient between PCM and water	10000	$\frac{\text{W}}{(\text{m}^2\text{C})}$
$h_{\text{P}}^{\text{min}}$	minimum convective heat transfer coefficient between PCM and water	10	$\frac{\text{W}}{(\text{m}^2\text{C})}$
$L_{\text{max}}$	maximum length of tank	50	m
$L_{\text{min}}$	minimum length of tank	0.1	m
$MINFRACT$	minimum fraction of the tank volume taken up by the PCM	$1.0 \cdot 10^{-6}$	—
$t_{\text{final}}^{\text{max}}$	maximum final time	86400	s
$\rho_{\text{P}}^{\text{max}}$	maximum density of PCM	20000	$\frac{\text{kg}}{\text{m}^3}$
$\rho_{\text{P}}^{\text{min}}$	minimum density of PCM	500	$\frac{\text{kg}}{\text{m}^3}$
$\rho_{\text{W}}^{\text{max}}$	maximum density of water	1000	$\frac{\text{kg}}{\text{m}^3}$
$\rho_{\text{W}}^{\text{min}}$	minimum density of water	950	$\frac{\text{kg}}{\text{m}^3}$

Table 10: Auxiliary Constants

## 10 References

- [1] J. Frederick Bueche. *Introduction to Physics for Scientists*. 4th ed. New York City, New York: McGraw Hill, 1986.
- [2] F. P. Incropera et al. *Fundamentals of Heat and Mass Transfer*. 6th ed. Hoboken, New Jersey: John Wiley and Sons, 2007.
- [3] Nirmitha Koothoor. “A document drive approach to certifying scientific computing software”. MA thesis. Hamilton, ON, Canada: McMaster University, 2013.
- [4] Marilyn Lightstone. *Derivation of tank/pcm model*. From Marilyn Lightstone’s Personal Notes. 2012.

- [5] David L. Parnas. “On the Criteria To Be Used in Decomposing Systems into Modules”. In: *Communications of the ACM* (1972), pp. 1053–1058.
- [6] David L. Parnas and P. C. Clements. “A rational design process: How and why to fake it”. In: *IEEE Transactions on Software Engineering* 12.2 (Feb. 1986), pp. 251–257.
- [7] David L. Parnas, P. C. Clements, and Wiess. “The Modular Structure of Complex Systems”. In: *ICSE '84: Proceedings of the 7th international conference on Software engineering*. 1984, pp. 408–417.
- [8] W. Spencer Smith and Lei Lai. “A new requirements template for scientific computing”. In: *Proceedings of the First International Workshop on Situational Requirements Engineering Processes - Methods, Techniques and Tools to Support Situation-Specific Requirements Engineering Processes, SREP'05*. Ed. by PJ Agerfalk, N. Kraiem, and J. Ralyte. In conjunction with 13th IEEE International Requirements Engineering Conference, Paris, France, 2005, pp. 107–121.