



## Energy Efficiency and Risk Management in Public Buildings

Project Number 260041

SUPPORTING ACTION

### EnRiMa

## Energy Efficiency and Risk Management in Public Buildings

### Deliverable D4.2: Symbolic Model Specification

Start date of the project: 01/10/2010

Duration: 42 months

Organisation name of lead contractor for this deliverable: URJC

Revision: 3.1-1 (30 March 2012)

Project funded by the European Commission within the Seventh Framework Programme (2007-2013)		
Dissemination Level		
PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	



# Contents

<b>List of Acronyms</b>	<b>2</b>
<b>Glossary</b>	<b>3</b>
<b>List of Tables</b>	<b>5</b>
<b>List of Figures</b>	<b>5</b>
<b>Executive summary</b>	<b>6</b>
<b>1. Introduction</b>	<b>7</b>
<b>2. EnRiMa Models</b>	<b>9</b>
2.1. Scenarios . . . . .	9
2.2. Energy Flow Schema . . . . .	11
<b>3. Strategic Symbolic Model Specification</b>	<b>11</b>
3.1. Indices and Sets . . . . .	12
3.1.1. Indices . . . . .	12
3.1.2. Sets and subsets . . . . .	13
3.2. Decision Variables . . . . .	14
3.3. Parameters . . . . .	15
3.4. Constraints . . . . .	18
3.4.1. Strategic Constraints . . . . .	18
3.4.2. Operational Constraints . . . . .	20
3.4.3. Integer Variables Constraints . . . . .	22
3.4.4. Non-negativity Constraints . . . . .	22
3.5. Objective Function . . . . .	24
<b>4. Operational Symbolic Model Specification</b>	<b>26</b>
4.1. Lower-level Constraints . . . . .	26
4.1.1. Nomenclature . . . . .	26
4.1.2. Constraints . . . . .	28
4.2. Operational objective function . . . . .	29
<b>5. Stochastic Model</b>	<b>30</b>
5.1. EnRiMa Stochastic Modelling . . . . .	30
5.2. Stochastic Transition Matrices . . . . .	33
5.3. Stochastic Sankey Diagrams . . . . .	34
<b>6. Conclusions</b>	<b>38</b>
<b>References</b>	<b>39</b>
<b>Appendix A. Data Tables</b>	<b>40</b>
<b>Appendix B. Code Examples</b>	<b>48</b>

## **List of Acronyms**

**AHU** Air-handling unit

**CHP** Combined heat and power

**CVaR** Conditional value at risk

**DER** Distributed energy resources

**DER-CAM** Distributed energy resources customer adoption model

**DSS** Decision support system

**DV** Decision variable

**EnRiMa** Energy efficiency and risk management in public buildings

**GUI** Graphical user interface

**HVAC** Heating, ventilation, and air conditioning

**MCEEM** Micro-grid customer engineering-economic model

**PV** Photovoltaic

**SMS** Symbolic model specification

**VaR** Value at risk

**WP** Work package

## Glossary

### Building parameters

The *BuildPars* expression used in some constraints refers to any feature of the building that can be a parameter in the model. Thus, it is measured in corresponding measurement units.

### Combined Heat and Power

CHP is a technology that can transform input primary fuels such as natural gas or biomass into both electricity and heat energy. It is considered an efficient technology as the heat produced in electricity generation is recovered for thermal energy needs in the building.

### Continuous sized energy-creating technologies

Technologies whose installation capacity of providing energy is continuous, that is, any power capacity can be installed within the building limits.

### Discrete sized energy-creating technologies

Technologies whose installation capacity of providing energy is discrete, that is, each installed device has a nominal capacity, and the total installed capacity of the technology is the product of the number of devices installed by the nominal capacity.

### Energy-absorbing technology

We denote by energy-absorbing technologies those technologies that allow the building to demand less input energy, in contrast to energy-creating technologies, that provide the energy demanded. These technologies can be storing technologies (e.g. batteries) or passive measures (e.g. isolation).

### Energy portfolio

By energy portfolio we mean the set of energy resources that are available in a building or site to provide the requested energy to the different loads (demand), including purchases in the market.

### Energy technologies portfolio

The energy technologies portfolio is the set of energy technologies, including generation, storage, and passive measures, that can be adopted in a building or site.

### Fictitious market

Fictitious markets are used in the formulation to take into account several inputs of energy that are not purchased in any market. For example, sun irradiation to produce energy through PV panels, or heat exchange through venting.

### Forward markets

Energy markets where future contracts can be traded in order to fix a price in advance.

### Spot markets

Markets where energy is available to be directly purchased.

### Space measurable passive technologies

Passive technologies which have an additive effect in the demand. They entail savings over the whole building, depending on its physical dimensions.

### Technologies parameters

The *TechPars* expression used in some constraints refers to any feature of the devices of a given technology that can be a parameter in the model. Thus, it is measured in corresponding measurement units.

**Unitary measurable passive technologies**

Passive technologies which have a multiplicative effect in the demand, that is, the higher the demand, the higher the savings. They entail savings over the use of the energy regardless the building dimensions.

## List of Tables

1.	Parameters for stochastic modelling . . . . .	36
2.	Sets and Indices . . . . .	40
3.	Subsets . . . . .	40
4.	Sets partition groups . . . . .	41
5.	Decision variables . . . . .	41
6.	Variables indices . . . . .	42
7.	Parameters . . . . .	44
8.	Parameter indices . . . . .	45
9.	Constraints . . . . .	47
10.	Objectives . . . . .	47
11.	Simulated example set $\mathcal{P}$ . . . . .	53
12.	Simulated example set $\mathcal{M}$ . . . . .	53
13.	Simulated example set $\mathcal{T}$ . . . . .	53
14.	Simulated example set $\mathcal{I}$ . . . . .	54
15.	Simulated example set $\mathcal{K}$ . . . . .	54
16.	Simulated example set $\mathcal{N}$ . . . . .	54
17.	Simulated example number of implemented variables . . . . .	57
18.	Simulated example implemented variables names . . . . .	57

## List of Figures

1.	EnRiMa DSS Scheme . . . . .	7
2.	Example of a scenario tree . . . . .	10
3.	Relationship between the models and the DSS . . . . .	10
4.	Energy Flow in the EnRiMa models . . . . .	11
5.	Two-stage dynamic stochastic model representation . . . . .	32
6.	Sankey diagram for EnRiMa's back-up site . . . . .	35
7.	Example stochastic Sankey diagram . . . . .	37

## Executive summary

In this deliverable, the Symbolic Model Specification (SMS) of the models developed within the EnRiMa project is described. The Symbolic Model Specification defines the mathematical representation of the stochastic model of all relevant energy subsystems and their interactions, which is a core part of the Decision Support System (DSS) Engine. This mathematical representation is composed of variables, parameters, and relations between them. In mathematical modelling, indices and sets are used to symbolize algebraically individual items (variables and parameters) in the model. In particular, two different models have been developed: strategic and operational. The strategic model has been designed in order to make strategic decisions concerning which technologies to install and/or decommission in the long term, that is, the energy technologies portfolio planning. Besides technologies, this planning includes forward markets contracts. The operational model deals with decisions involving the dispatching of energy through the installed technologies in the short term or through purchases in spot markets, that is, the energy portfolio selection. With the aim of coping with short- and long-term decisions concurrently, the strategic model includes a simplified version of operational energy-balance constraints, and the operational model, in turn, includes some realisations of the strategic decisions as parameters. In this way, both models feed back to each other, becoming a holistic DSS.

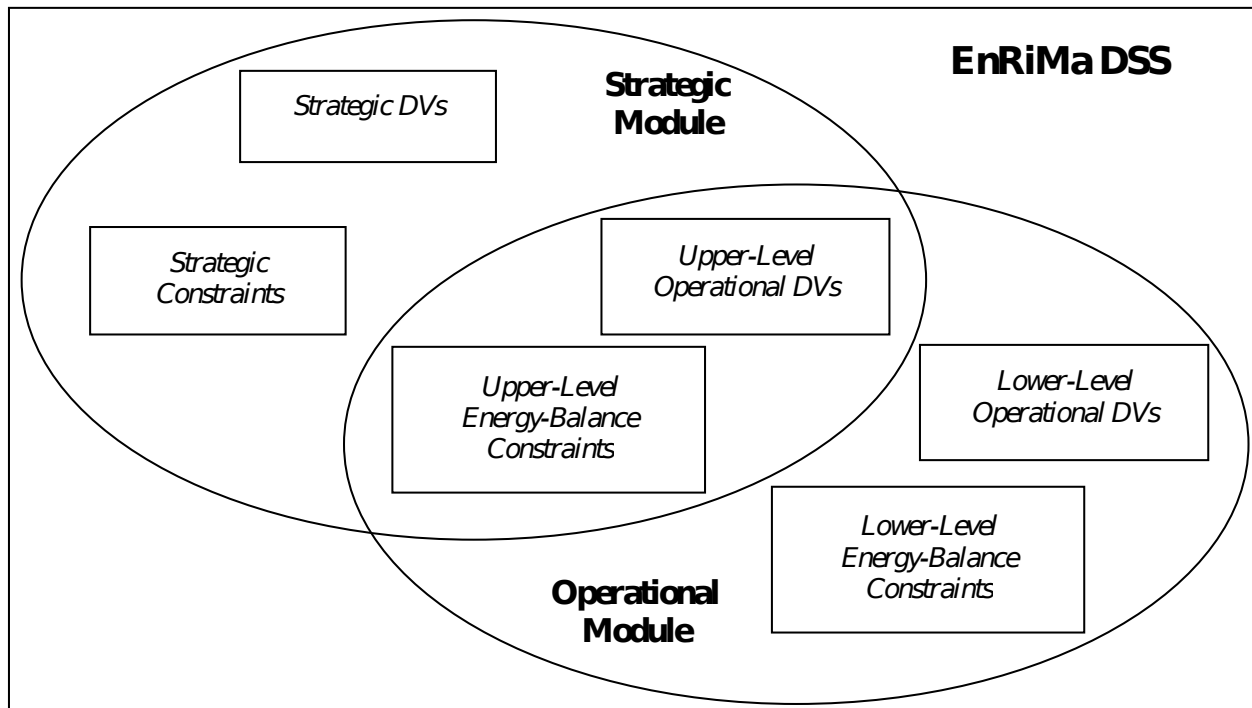
The EnRiMa models go beyond the state-of-the-art as they take into account uncertainties in prices and loads, which is a novel feature at the building level. Furthermore, the models provide the user with the capability to manage risk, pollutants emissions, and energy efficiency directly along with reducing costs, the possibility of taking financial positions where available, and long-term and inter-period Conditional Value-at-Risk (CVaR) constraints. This is done by solving the models within a stochastic programming framework. This framework allows the definition of a new kind of Sankey diagrams, so-called stochastic Sankey diagrams, which show how the flows of energy change depending on the risk aversion of the user of the model. In fact, this is another contribution of this deliverable to the state-of-the-art. The models can be as specific and detailed as the needs of the users are. In particular, the operational model has been designed to be used by operators taking care of the building physics and thermodynamics, making it possible to include these characteristics also at a strategic level. Thus, the contribution of the strategic model is that it embeds a simplified version of operational constraints, which allows accounting for short-term decisions and parameters when making long-term decisions. The contribution of the operational model is that the traditional definition of demand for heating or cooling is replaced by user requirements for temperature. This leads to lower-level energy-balance constraints that reflect the thermodynamics of heating and cooling systems as well as building physics.

The generation of the definitions and equations describing the SMS for this deliverable has been done dynamically using the R software [10]. This framework, easily portable to any other environment, allows adding new entities and constraints to the SMS easily. In fact, this is especially useful when updating and adapting the models to new circumstances such as new buildings, technologies, markets, etc.

The Symbolic Model Specification contained in this deliverable (D4.2), along with the related deliverables D2.2 and D3.2, provides the framework for the next steps of the EnRiMa project, especially the development of the DSS Engine and the Graphical User Interface (GUI). Particularly, task 4.5 (Stochastic optimisation algorithms and solvers) will continue this work, building the complete implementation for the test sites and selecting algorithms and computational requirements afterwards. This work will be carried out in close cooperation with the one in task 4.6 and Work Package 5 (WP5), which will be fed with the structure defined herein.

# 1. Introduction

One of the EnRiMa project’s innovations is the capability to optimise adoption and deployment of energy-efficient equipment taking into account long- and short-term decisions. Thus, two different models are being developed: strategic and operational. Moreover, strategic decisions are influenced by the operational performance, while operations depend on historic strategic decisions. Operational decisions are those involving the dispatch of installed technologies in the short term, whereas strategic decisions concern, in the long term, to which technologies to install and/or decommission, or how much energy buy in forward markets. In an attempt to tackle short- and long-term decisions as a whole, the strategic model includes a simplified version of operational energy-balance constraints. The operational model, in turn, includes the realisation of the strategic decisions as parameters. Figure 1 is reproduced here from D2.2 to show the interplays between both models and their Decision Variables (DV) and constraints.



**Figure 1:** EnRiMa DSS Scheme (from D2.2).

Finally, including the information about the volatility of market and energy inputs/outputs to the models through stochastic optimisation, the EnRiMa Decision Support System (DSS) becomes a valuable risk management framework for energy efficiency. The main instrument included in the EnRiMa project for managing risks is the use of stochastic programming for optimisation.

The Symbolic Model Specification (SMS) defines the mathematical representation of the stochastic model of all relevant energy subsystems and their interactions, which is core part of the DSS Engine. This mathematical representation is composed of variables, parameters, and relations between them. In mathematical modelling, indices and sets are used to symbolize algebraically individual items (variables and parameters) in the model. Therefore, we first define indices and sets, decision variables, and parameters for the strategic and operational mathematical programming models in Sections 3 and 4, respectively. At the end of these sections, the constraints and objectives for each model are specified. The core of the work has been the specification of the deterministic models in order to introduce them into the definitive stochastic framework. Once these models have been developed, it is straightforward to extend them into stochastic models (Section 5), thereby providing the capacity for risk management to the SMS. In Appendix A, the



data tables with the objects of the SMS can be found. Appendix B contains some examples of the code developed using the R software<sup>1</sup> for the SMS generation and implementation, through a very simple example. This code constitutes a real SMS and is especially useful when updating and adapting the models to new circumstances such as new buildings, technologies, markets, etc. The complete code is available in the private area of the EnRiMa web server. Credentials will be sent to project stakeholders upon request.

The proposed models proceed beyond the state-of-the-art as they take into account uncertainties in prices and loads, which is a novel feature at the building level. Furthermore, the models provide the user with the capability to manage risk, pollutants emissions, and energy efficiency directly along with reducing costs, the possibility of taking financial positions where available, and long-term and inter-period Conditional Value-at-Risk (CVaR) constraints. The models developed in the EnRiMa project extend existing deterministic approaches such as DER-CAM [11, 12] or MCEEM [8] into a stochastic one capable of dealing with short- and long-term uncertainties inherent to typical energy resources and loads. In addition, a new kind of Sankey diagrams is developed, so-called stochastic Sankey diagrams which are based on the risk aversion included in the stochastic modelling. Furthermore, our operational models are designed to be used by operators as they take care of the building physics and thermodynamics.

The Symbolic Model Specification contained in this deliverable (D4.2) makes use of the Requirement Analysis developed in deliverable D4.1, Requirement Analysis [5], extending it with a more detailed description of the variables and parameters, based on the information provided in deliverable D1.1, Requirement Assessment [3], which contains the information about the test sites in Spain and Austria. Concretely, the key uncertainties described in D4.1 based on the information provided by D1.1 have been used to outline the general formulation of the parameters suitable to be treated as stochastic. In any case, the models are general and, therefore, the SMS can be adapted to the needs of each specific site in question at any time. For the design of the way in which stochastic Sankey diagrams are going to be displayed, the information in deliverable D2.1 has been taken into account. Along with the related deliverables D2.2 and D3.2, this deliverable provides the framework for the next steps of the EnRiMa project, especially the development of the DSS Engine and the GUI. Particularly, task 4.5 (Stochastic optimisation algorithms and solvers) will continue this work, building the complete implementation for the test sites and selecting algorithms and computational requirements afterwards. This work will be carried out in close cooperation with task 4.6 (Implementation of the DSS Kernel) and WP5, which will be fed with the structure defined herein.

The structure of this document is as follows:

- ◇ Section 2 describes the EnRiMa models and the relationship between them and the EnRiMa DSS, including a description of the scenario trees used for the stochastic modelling.
- ◇ Section 3 contains the main result of this deliverable, consisting in a description of the objects in the strategic model (sets, variables, parameters) and their mathematical representation and relations between them.
- ◇ In Section 4 the lower-level constraints from D2.2 are reproduced for completeness.
- ◇ Section 5 describes the extension of the models to the stochastic approach. This section includes the definition of the so-called “Stochastic Sankey diagrams”.
- ◇ Appendix A contains the data tables containing the SMS described in this document.

---

<sup>1</sup>The R software is used in this deliverable for prototyping. It is not decided yet which software tools and languages that be used in the final implementation within next tasks in WP4 next tasks and WP5.

- ◇ In Appendix B some examples of the code used to generate the SMS and an example model is provided.

## 2. EnRiMa Models

The EnRiMa models replace the deterministic components of classical models by new stochastic representations in order to support the analysis of stochastic parameters, such as: prices, demands, and systems availability. Thus, the models are enhanced exploiting the advantages of stochastic modelling over deterministic. The main advantage is that the stochastic formulation allows optimising the expected value taking into account uncertainties, which are modelled through scenario generation.

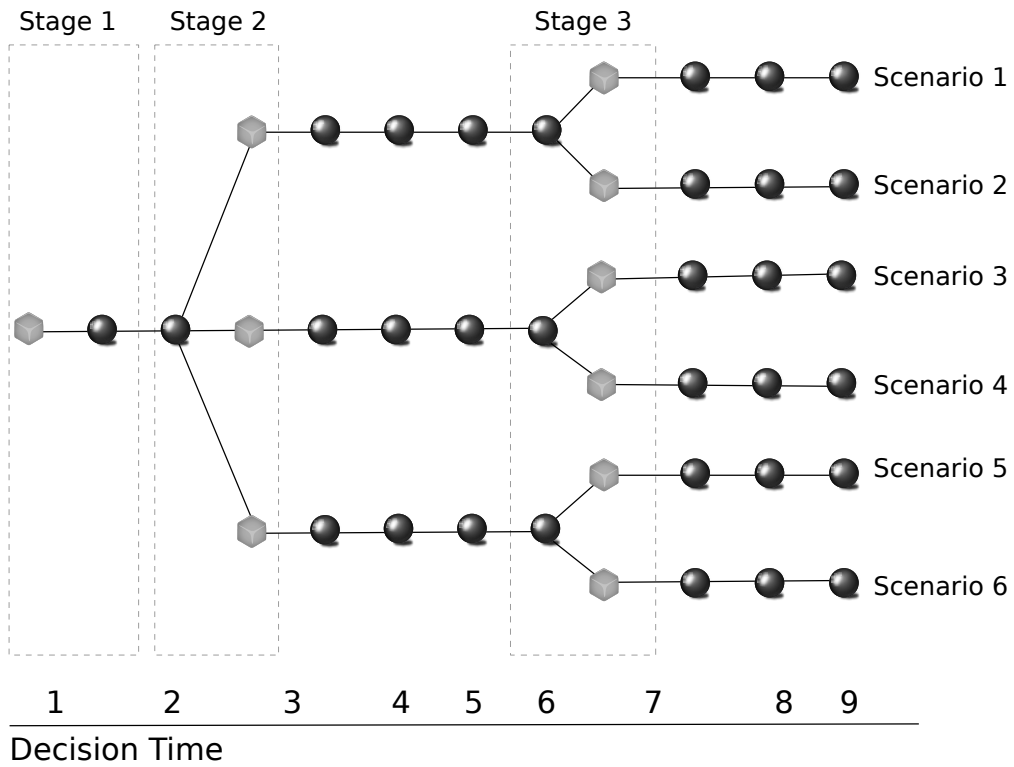
### 2.1. Scenarios

Stochasticity is represented through “scenario trees”. A scenario is a possible realization of all the stochastic parameters in the model, see deliverable D3.2 [7] for an in-depth explanation. The number of possible scenarios depends on the stages of the stochastic programming model and the number of branches in each scenario tree node. It is important to remark that a “stage” in the stochastic model is not the same as the real decision point of time. A stage is a model feature, which represents the point of time when new information arrives to the model. In theory, there may be as many decision periods as stages, but we need to reduce the number of stages in order to have a solvable model. Thus, between two stages, there may be several decision periods. Moreover, for each stage, there are several branches representing possible realizations of the uncertain parameters. These branches are also simplified as described in deliverable D3.2 [7]. Figure 2 illustrates a possible scenario tree. The cubes represent the stages of the stochastic optimisation model and the spheres represent the decision periods. Thus, in this scenario tree, during the second period, let us say a year, new information arrives about the stochastic parameters, resulting in three possible scenarios with a given probability. Four years later, new information is available with two possible scenarios for each of the branches coming from the second stage. Hence, after the last decision period there is a total of 6 scenarios.

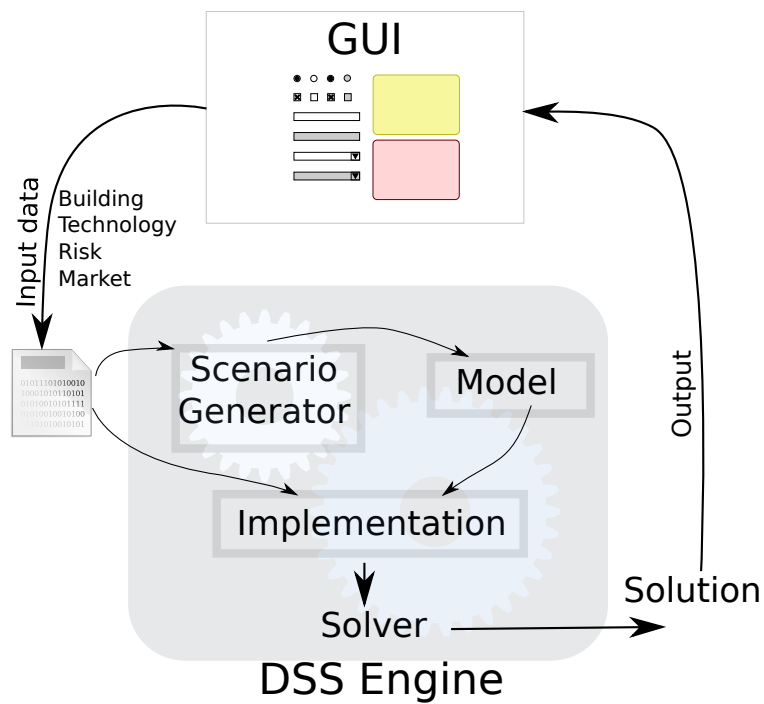
In the SMS, the groups of parameters that can be stochastic are identified. The real data in the SMS implementation provides the nature of each individual parameter and, if available, information about the variability of the parameter to the scenario generation tool. A desired tree configuration can be also set in the implementation, for example: create scenarios for unveiled uncertainties every five years. The scenario generation tool will return the scenarios to the DSS that, jointly with the SMS, builds the stochastic implementation for the solver. Figure 3 shows a scheme of this framework.

The models in this deliverable gather the information from WP1 regarding the building configuration, used technologies and main uncertainties, WP2 regarding the energy balance equations, and WP3 regarding the scenario generation. This information is used to represent the optimisation models (first deterministic, then stochastic), the illustrative implementation, and numerical examples. As the EnRiMa project is continuously evolving, the models developed for this deliverable are not definitive, and changes will probably be needed as the project advances. Specifically, during the development of the DSS GUI (WP5) and during the algorithms and solvers implementation (task 4.5 in WP4), improvements and needed changes to the model may be identified. They will eventually be added to the stochastic optimization prototypes until reaching the final version.

The EnRiMa models comprise innovative modelling issues. Energy flowing through HVAC (Heating, ventilation, and air conditioning) technologies is modelled at the operational level. Passive measures are included in the strategic model at an abstract level taking into account the building



**Figure 2:** Illustrated scenario tree.



**Figure 3:** Relationship between the models and the DSS.

configuration implicitly. Moreover, at the operational level, building basic physics and thermodynamics are explicitly treated, resulting in enhanced energy flow modelling.

## 2.2. Energy Flow Schema

The EnRiMa models symbolise the energy flow through decision variables. Energy flows as follows: Energy from the markets (including fictitious markets such as renewable sources) flows to energy-creation technologies, energy-storage technologies, sales to the market, or is used for directly supplying the demand. The generated energy can supply the demand, be stored or be sold to the market. Storage technologies receive energy from markets and generation technologies, and supply the demand. Figure 4 shows schematically this energy flow including some of the symbols defined in the following section, as the relevant sets and the variables that symbolise the energy flows.

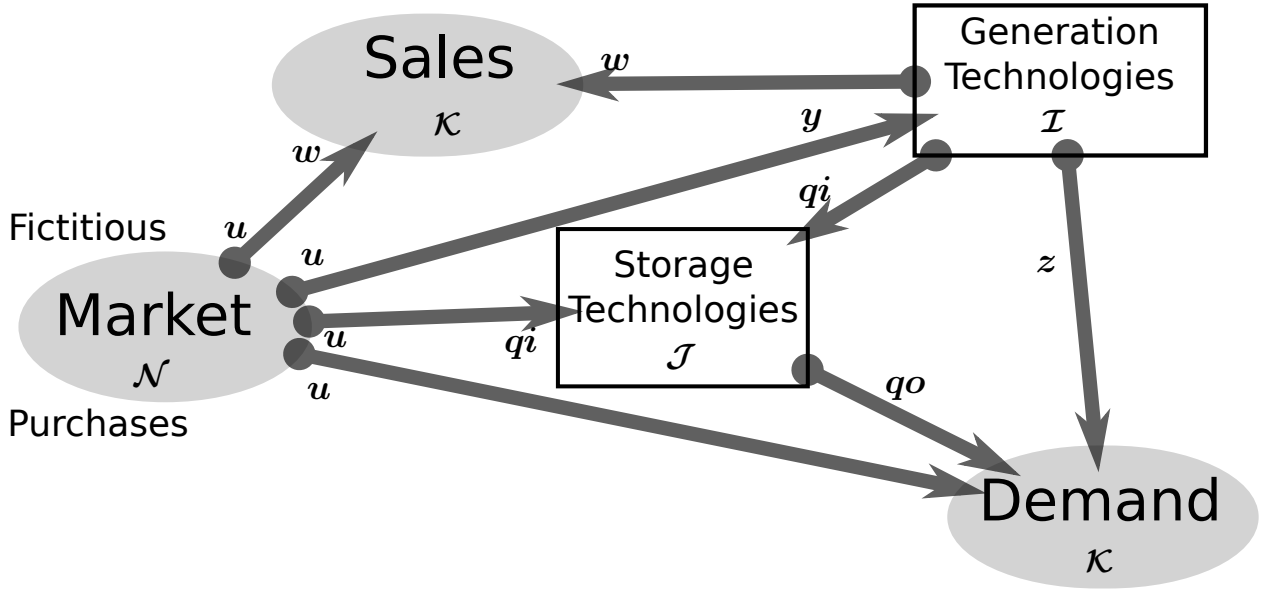


Figure 4: Energy Flow within the EnRiMa models

## 3. Strategic Symbolic Model Specification

This section presents the SMS for the strategic model. As already mentioned, the strategic model is used in order to make strategic decisions concerning which technologies to install and/or decommission in the long term. In an attempt to tackle short- and long-term decisions as a whole, the strategic model includes a simplified version of operational energy-balance constraints, and the operational model, in turn, includes the realisation of the strategic decisions as parameters. In this way, both models feed back to each other.

At the end, the goal of the DSS is to support different stakeholders, which will likely have different goals, in their decision processes concerning energy planning (strategic model) or energy dispatching (operational model). This section lists and briefly describes the basic elements of the strategic SMS, namely: indices, sets, variables and parameters. For non-evident concepts, consult the glossary at the beginning of this document and Figure 4. Then, constraints are built by combining those basic objects through the appropriate relations. After this, the objective function for minimising total costs is symbolised and described. This is the main objective identified in previous stages of the EnRiMa project, in particular, in D4.1 [5]. This total cost is formed from different components such as investment costs, decommissioning costs, operational costs, and purchasing costs. Subsidies and energy sales are subtracting terms in this objective function. It is important

to remark that the framework presented in this SMS is general enough to customise the EnRiMa DSS and, thus, provide different objective functions for different stakeholders (building managers, policy makers, etc.) to the solvers.

The definitions and equations presented in this section have been dynamically generated as illustrated in Appendix B. This framework, easily portable to any other environment, allows adding new entities and constraints to the SMS easily. In fact, this constitutes a real framework to develop symbolic model specifications.

### 3.1. Indices and Sets

Small Latin letters represent the index that goes with variables and parameters as subscripts or superscripts. The counterpart capital letter indicates the number of elements in the corresponding set, which is represented by “stylised” capital letters ( $\mathcal{I}$ ,  $\mathcal{M}$ , etc.). Some auxiliary indices are needed for the equations, related with other sets, as  $\mathcal{P}$ ,  $\mathcal{M}$ , or  $\mathcal{K}$ . Similarly, subsets used for delimiting sums in the equations are defined. Note that the sets can be expanded with new subsets as new items are available for other buildings or technologies (e.g. biomass).

#### 3.1.1. Indices

##### Sets' indices:

$i$  Energy-creating technology;  $i \in \mathcal{I}$ .

$j$  Energy-absorbing technology;  $j \in \mathcal{J}$ .

$k$  Energy type;  $k \in \mathcal{K}$ .

$l$  Pollutant;  $l \in \mathcal{L}$ .

$m$  Mid-term period;  $m \in \mathcal{M}$ .

$n$  Energy market;  $n \in \mathcal{N}$ .

$p$  Long-term period;  $p \in \mathcal{P}$ .

$t$  Short-term period;  $t \in \mathcal{T}$ .

##### Auxiliary indices:

$a$  Technology age. Age of technologies to indicate effect of aging for energy-creating (c.f.  $p$ ).

$a1$  Auxiliary index. Auxiliary index to compute capacity available during each long-term period (c.f.  $p$ ).

$a2$  Auxiliary index. Auxiliary index to compute capacity available during each long-term period (c.f.  $p$ ).

$kk$  Output energy type. Auxiliary index for type of energy output when converting energy types (c.f.  $k$ ).

$mm$  Ancestor mid-term period. Auxiliary index for ancestor mid-term periods when setting forward contracts (c.f.  $m$ ).

$q$  Decommissioning period. Long-term period for decommissioning-strategic decision (c.f.  $p$ ).

### 3.1.2. Sets and subsets

#### Energy-creating technologies (supply side)

$$\mathcal{I} \supset \mathcal{I}_E \cup \mathcal{I}_H \cup \mathcal{I}_C$$

$\mathcal{I}_E$  Technologies that generate electricity.

$\mathcal{I}_H$  Technologies that generate heat.

$\mathcal{I}_C$  Technologies that generate cooling.

Discrete sized energy generating technologies

$$\mathcal{I}_Z \subset \mathcal{I}$$

Continuous sized energy generating technologies

$$\mathcal{I}_R \subset \mathcal{I}$$

#### Energy-absorbing technologies (– demand side)

$$\mathcal{J} \supset \mathcal{J}_S \cup \mathcal{J}_{PS} \cup \mathcal{J}_{PU}$$

$\mathcal{J}_S$  Storage technologies.

$\mathcal{J}_{PS}$  Space measurable passive technologies.

$\mathcal{J}_{PU}$  Unitary measurable passive technologies.

#### Energy type

$$\mathcal{K} \supset \mathcal{K}_E \cup \mathcal{K}_H \cup \mathcal{K}_C \cup \mathcal{K}_G \cup \mathcal{K}_S$$

$\mathcal{K}_E$  Electricity.

$\mathcal{K}_H$  Heat.

$\mathcal{K}_C$  Cooling.

$\mathcal{K}_G$  Natural Gas.

$\mathcal{K}_S$  Solar.

Input energy types for each technology

$$\mathcal{K}_{I(i)} \subset \mathcal{K}$$

Output energy types for each technology

$$\mathcal{K}_{O(i)} \subset \mathcal{K}$$

Primary energy output of energy-creating technology  $i$

$$\mathcal{K}_{F(i)} \subset \mathcal{K}$$

### Types of pollutants

$\mathcal{L}$  (no subsets).

### Mid-term representative period (profile, e.g. day)

$\mathcal{M}$

Mid-term periods when energy is purchased in forward markets

$\mathcal{M}_A \subset \mathcal{M}$

Mid-term periods when energy is sold in forward markets

$\mathcal{M}_S \subset \mathcal{M}$

### Markets for energy products and primary fuels

$\mathcal{N} \supset \mathcal{N}_E \cup \mathcal{N}_H \cup \mathcal{N}_D \cup \mathcal{N}_G$

$\mathcal{N}_E$  Electricity markets.

$\mathcal{N}_H$  Heating markets.

$\mathcal{N}_D$  Fictitious markets.

$\mathcal{N}_G$  Natural gas markets.

Markets where a type of energy can be bought

$\mathcal{N}_{B(k)} \subset \mathcal{N}$

Markets where a type of energy can be sold

$\mathcal{N}_{S(k)} \subset \mathcal{N}$

### Long-term period for strategic decisions (e.g. year)

$\mathcal{P}$  (no subsets).

### Short-term decision period (e.g. hour)

$\mathcal{T}$  (no subsets).

## 3.2. Decision Variables

Each variable is represented by one or more small Latin letters along with one or more of the indices described above. After a description of the variable and the related sets, the units of the variable are shown. Note that, for the sake of generalization, we do not distinguish between the different types of energy in the units. For example, if a technology produces electricity, the units for the capacity should be  $kW_e$ , but we write simply  $kW$ . Similarly, when describing the parameters

in the following subsection, if a technology produces heat using natural gas as input, the units should be  $kWh/kWh_{th}$ , but we simply write  $kWh/kWh$ . Along with the units, possible data types for database design purposes are given in brackets: **dec** for decimal values and **int** for integer values.

**Strategic decisions:**

- $s_i^p$  Available capacity in long-term period  $p$ , energy-creating technology  $i$ . ( $kW$  [**dec**]).
- $sd_i^{p,q}$  Number of devices / capacity to be decommissioned in long-term period  $p$ , energy-creating technology  $i$ , decommissioning period  $q > p$ . (*devices* [**int/dec**]).
- $si_i^p$  Number of devices / capacity to be installed in long-term period  $p$ , energy-creating technology  $i$ . (*devices* [**int/dec**]).
- $x_j^p$  Available capacity in long-term period  $p$ , energy-absorbing technology  $j$ . ( $kWh$  [**dec**]).
- $xd_j^{p,q}$  Number of devices / capacity to be decommissioned in long-term period  $p$ , decommissioning period  $q$ , energy-absorbing technology  $j$ . ( $kWh$  [**dec**]).
- $xi_j^p$  Number of devices / capacity to be installed in long-term period  $p$ , energy-absorbing technology  $j$ . ( $kWh$  [**dec**]).

**Operational decisions:**

- $e^{p,m,t}$  Energy consumption in long-term period  $p$ , mid-term period  $m$ , short-term period  $t$ . ( $kWh$  [**dec**]).
- $q_{k,j}^{p,m,t}$  Energy added to storage in long-term period  $p$ , mid-term period  $m$ , short-term period  $t$ , energy type  $k$ , energy-absorbing technology  $j$ . ( $kWh$  [**dec**]).
- $qo_{k,j}^{p,m,t}$  Energy released from storage in long-term period  $p$ , mid-term period  $m$ , short-term period  $t$ , energy type  $k$ , energy-absorbing technology  $j$ . ( $kWh$  [**dec**]).
- $r_{k,j}^{p,m,t}$  Energy to be stored in long-term period  $p$ , mid-term period  $m$ , short-term period  $t$ , energy type  $k$ , energy-absorbing technology  $j$ . ( $kWh$  [**dec**]).
- $u_{k,n}^{p,m,t,mm}$  Purchase of energy in long-term period  $p$ , mid-term period  $m$ , short-term period  $t$ , energy type  $k$ , energy market  $n$ , ancestor mid-term period  $mm$ . ( $kWh$  [**dec**]).
- $w_{k,n}^{p,m,t,mm}$  Sale of energy in long-term period  $p$ , mid-term period  $m$ , short-term period  $t$ , energy type  $k$ , energy market  $n$ , ancestor mid-term period  $mm$ . ( $kWh$  [**dec**]).
- $y_{i,k}^{p,m,t}$  Input of energy in long-term period  $p$ , mid-term period  $m$ , short-term period  $t$ , energy-creating technology  $i$ , energy type  $k$ . ( $kWh$  [**dec**]).
- $z_{i,k}^{p,m,t}$  Output of energy in long-term period  $p$ , mid-term period  $m$ , short-term period  $t$ , energy-creating technology  $i$ , energy type  $k$ . ( $kWh$  [**dec**]).

**3.3. Parameters**

Each parameter is represented by one or more capital Latin letters along with one or more of the indices described above. After a short description of the parameter and the related sets, the units of the parameter are given. Similarly to the notation for variables, we do not distinguish between the different types of energy in the units (see example above). Along with the units, possible data types for database design purposes are given in brackets: **dec** for decimal values and **int** for integer



values. Next, a longer description is included for some parameters. Finally, the nature of the parameter (stochastic or deterministic) is stated. Note that if the parameter could be stochastic for some of the indices and deterministic for others, the nature is represented by “either”. In particular, the most important uncertainties to be accounted on are those referring to energy loads (demand) and prices. Note that the former can be influenced by “external” parameters such as the weather. Moreover, the availability parameter for technologies allows dealing with breakdowns and similar uncertainties.

**Building parameters:**

*BA* Total building area ( $m^2$  [dec]). Nature: Deterministic.

*BV* Total building volume ( $m^3$  [dec]). Nature: Deterministic.

*FA* Building floor Area for ( $m^2$  [dec]). Nature: Deterministic.

*HO* Number of operating periods (*number* [int]). Nature: Deterministic.

*RA* Building roof area ( $m^2$  [dec]). Nature: Deterministic.

**Demand parameters:**

$D_k^{p,m,t}$  Energy demand for long-term period  $p$ , mid-term period  $m$ , short-term period  $t$ , energy type  $k$  ( $kWh$  [dec]). Note that in the strategic model it is a parameter, while in the operational it is a decision variable. Energy demand for a type of energy is composed by all the loads for a type of energy. These loads can be supplied by direct purchases in the market, energy generation, energy storage and natural gains (vent, sun, etc.) Nature: Stochastic.

**Energy Efficiency parameters:**

$B_{k,n}$  Primary energy ratio energy type  $k$ , energy market  $n$  ( $kWh/kWh$  [dec]). Units of primary energy from the market required to make one unit of energy for use. Nature: Deterministic.

**Finance parameters:**

$CD_i^a$  Decommissioning cost for energy-creating technology  $i$ , technology age  $a$  ( $EUR/kW$  [dec]). Nature: Either.

$CDS_j^a$  Decommissioning cost for technology age  $a$ , energy-absorbing technology  $j$  ( $EUR/kWh$  [dec]). Nature: Either.

$CI_i^{p,a}$  Installation/maintenance cost for long-term period  $p$ , energy-creating technology  $i$ , technology age  $a$  ( $EUR/kW$  [dec]). Nature: Either.

$CIS_j^{p,a}$  Installation/maintenance cost for long-term period  $p$ , technology age  $a$ , energy-absorbing technology  $j$  ( $EUR/kW$  [dec]). Nature: Either.

$COS_{k,j}^{p,m,t}$  Operation cost for long-term period  $p$ , mid-term period  $m$ , short-term period  $t$ , energy type  $k$ , energy-absorbing technology  $j$  ( $EUR/kWh$  [dec]). Nature: Either.

$CP_{k,n}^{p,m,t}$  Purchasing cost for long-term period  $p$ , mid-term period  $m$ , short-term period  $t$ , energy type  $k$ , energy market  $n$  ( $EUR/kWh$  [dec]). Nature: Either.

*EF* Required energy efficiency of the site for ( $kWh/kWh$  [dec]). Nature: Deterministic.

*IL<sup>p</sup>* Budget constraint for long-term period  $p$  ( $EUR$  [dec]). Nature: Deterministic.

$SP_{i,k,n}^{p,m,t}$  Selling price for long-term period  $p$ , mid-term period  $m$ , short-term period  $t$ , energy-creating technology  $i$ , energy type  $k$ , energy market  $n$  ( $EUR/kWh$  [dec]). Nature: Stochastic.

**Model parameters:**

$DM_m^p$  Profile duration for long-term period  $p$ , mid-term period  $m$  ( $days$  [int]). Mid-term periods define a profile of the relevant parameters over a given number of consecutive short-term periods. This parameter is for the number of mid-term periods within a long-term period. Nature: Deterministic.

$DT$  Length of the decision-making period ( $hours$  [dec]). Nature: Deterministic.

**Policy parameters:**

$CO_{i,k}^{p,m,t}$  Operating Cost for long-term period  $p$ , mid-term period  $m$ , short-term period  $t$ , energy-creating technology  $i$ , energy type  $k$  ( $EUR/kWh$  [dec]). For many index values, this is also a financial parameter. It is included here to take into account “feed-in-tariffs” as a source of variability due to policy maker’s decisions. Nature: Either.

$PL_l^p$  Emissions limit for long-term period  $p$ , pollutant  $l$  ( $kg$  [int]). Nature: Either.

$PP_{i,k,n}^{p,m,t}$  Purchasing price long-term period  $p$ , mid-term period  $m$ , short-term period  $t$ , energy-creating technology  $i$ , energy type  $k$ , energy market  $n$  ( $EUR/kWh$  [dec]). Possible time-of-use tariffs. Nature: Stochastic.

$SU_i^p$  Subsidies for long-term period  $p$ , energy-creating technology  $i$  ( $EUR/kW$  [dec]). This is also a financial parameter, but it is included here due to the importance of optimising policy decisions for the EnRiMa project. Nature: Stochastic.

**Technologies parameters:**

$A_i^{p,m,t}$  Availability factor for long-term period  $p$ , mid-term period  $m$ , short-term period  $t$ , energy-creating technology  $i$  ( $kW/kW$  [dec]). Nature: Stochastic.

$AG_i^a$  Aging factor for energy-creating technology  $i$ , technology age  $a$  ( $kW/kW$  [dec]). Nature: Deterministic.

$AS_i^a$  Aging factor for energy-creating technology  $i$ , technology age  $a$  ( $kW/kW$  [dec]). Nature: Deterministic.

$C_{i,l,n}$  Pollution emissions for energy purchases for energy-creating technology  $i$ , pollutant  $l$ , energy market  $n$  ( $kg/kWh$  [dec]). Nature: Deterministic.

$E_{i,k,kk}$  Conversion coefficient for energy-creating technology  $i$ , energy type  $k$ , output energy type  $kk$  ( $kWh/kWh$  [dec]). Nature: Deterministic.

$G_i$  Nominal capacity for energy-creating technology  $i$  ( $kW/device$  [dec]). Equals 1 for  $i \in \mathcal{I}_R$ . It is the power that each installed device can supply to the demand side. Nature: Deterministic.

$GS_j$  Storage capacity for energy-absorbing technology  $j$  ( $kW/kW$  [dec]). Nature: Deterministic.

$H_{i,k,l}$  Pollution emissions for energy generation for energy-creating technology  $i$ , energy type  $k$ , pollutant  $l$  ( $kg/kWh$  [dec]). Nature: Deterministic.

$LS_i^{p,m,t}$  Loss per unit of stored energy for long-term period  $p$ , mid-term period  $m$ , short-term period  $t$ , energy-creating technology  $i$  ( $kWh$  [dec]). Nature: Deterministic.

- $OA_{k,j}$  Charging lower limit for energy type  $k$ , energy-absorbing technology  $j$  ( $kWh/kWh$  [dec]). Lower limit on the fraction of storage capacity that must be charged with a type of energy. Nature: Deterministic.
- $OB_{k,j}$  Charging upper limit for energy type  $k$ , energy-absorbing technology  $j$  ( $kWh/kWh$  [dec]). Upper limit on the fraction of storage capacity that must be charged with a type of energy. Note that  $0 \leq OA \leq OB \leq 1$ . Nature: Deterministic.
- $OD_{k,j}$  End use demand reduction per unit for energy type  $k$ , energy-absorbing technology  $j$  ( $kWh/kWh$  [dec]). Nature: Deterministic.
- $OI_{k,j}$  Charging ratio to storage for energy type  $k$ , energy-absorbing technology  $j$  ( $kWh/kWh$  [dec]). Some energy is lost when storing. This parameter is between 0 and 1. Nature: Deterministic.
- $OO_{k,j}$  Discharging ratio from storage for energy type  $k$ , energy-absorbing technology  $j$  ( $kWh/kWh$  [dec]). Some energy is lost when releasing from storage. This parameter is greater than 1. Nature: Deterministic.
- $OR_{k,j}$  Discharging availability for energy type  $k$ , energy-absorbing technology  $j$  ( $kWh/kWh$  [dec]). Not all the stored energy is available for discharging. This parameter is between 0 and 1. Nature: Deterministic.
- $OS_{k,j}$  Energy storage availability between short-term periods for energy type  $k$ , energy-absorbing technology  $j$  ( $kWh/kWh$  [dec]). This parameter represent the loses in the storage between periods, being between 0 and 1. Nature: Deterministic.
- $SC_i^p$  Storage capacity for long-term period  $p$ , energy-creating technology  $i$  ( $kWh$  [dec]). Nature: Deterministic.

### 3.4. Constraints

The constraints presented in the SMS are general enough to cover a wide range of buildings, technologies and policies. Hence, some of the constraints are expressed as functions of the actual parameters in a specific implementation. In such cases, examples are provided.

Before the equations, an explanation for the constraint is given. Following the expression of the constraint, the sets are stated for which the constraint must hold. Below the equation, the units of each term (variable or parameter) are given, separated by commas (,) in order to check consistencies. Finally, an indicative number of constraints is provided given the size of the relevant sets.

#### 3.4.1. Strategic Constraints

##### Available generating technologies calculation

The available capacity to generate a type of energy for each technology in a long-term period equals the number of total devices available times their nominal capacity, corrected by the aging factor.

$$s_i^p = G_i \sum_{a1=0}^p AG_i^{p-a1} \left( si_i^{a1} - \sum_{a2=a1+1}^p sd_i^{a1,a2} \right) \quad p \in \mathcal{P}, i \in \mathcal{I}, \quad (1)$$

$kW \sim kW/device, kW/kW, devices, devices$   $P \times I$  constraints.

### Available storing technologies calculation

The available capacity to store a type of energy for each technology in a long-term period equals the number of total devices available times their nominal capacity, corrected by the aging factor.

$$x_j^p = GS_j \sum_{a1=0}^p AS_i^{p-a1} \left( xi_j^{a1} - \sum_{a2=a1+1}^p xd_j^{a1,a2} \right) \quad p \in \mathcal{P}, j \in \mathcal{J}, \quad (2)$$

$kWh \sim kW/kW, kW/kW, kWh, kWh \quad P \times J$  constraints.

### Generation technologies decommissioning limit

The number of devices to be decommissioned must be less than the number of devices previously installed.

$$\sum_{a1>p} sd_i^{p,a1} \leq si_i^p \quad p \in \mathcal{P}, i \in \mathcal{I}, \quad (3)$$

$devices \sim devices \quad P \times I$  constraints.

### Storage technologies decommissioning limit

The number of devices to be decommissioned must be less than the number of devices previously installed.

$$\sum_{a1>p} xd_j^{p,a1} \leq xi_j^p \quad p \in \mathcal{P}, j \in \mathcal{J}, \quad (4)$$

$kWh \sim kWh \quad P \times J$  constraints.

### Budget limit

The total investment in technologies must be lower than a specified budget limit. This includes installation and decommissioning costs.

$$\begin{aligned} & \sum_{i \in \mathcal{I}} CI_i^{p,0} \cdot G_i \cdot si_i^p + \sum_{j \in \mathcal{J}} CIS_j^{p,0} \cdot GS_j \cdot xi_j^p \\ & + \sum_{i \in \mathcal{I}} G_i \left( \sum_{a1=0}^p CD_i^{p-a1} \sum_{a2=a1+1}^p sd_i^{a1,a2} \right) + \sum_{j \in \mathcal{J}} \left( \sum_{a1=0}^p CDS_j^{p-a1} \sum_{a2=a1+1}^p xd_j^{a1,a2} \right) \leq IL^p \quad p \in \mathcal{P}, \end{aligned} \quad (5)$$

$EUR/kW, kW/device, devices, EUR/kW, kW/kW, kWh, kW/device, EUR/kW, devices, EUR/kWh, kWh \sim EUR \quad P$  constraints.

### Emissions limit

The total emissions must be lower than the allowed limit per year.

$$\sum_{m \in \mathcal{M}} DM_m^p \left( \sum_{i \in \mathcal{I}} \sum_{k \in \mathcal{K}} \sum_{t \in \mathcal{T}} H_{i,k,l} \cdot y_{i,k}^{p,m,t} + \sum_{n \in \mathcal{N}} \sum_{k \in \mathcal{K}} \sum_{t \in \mathcal{T}} C_{i,l,n} \cdot u_{k,n}^{p,m,t,mm} \right) \leq PL_l^p \quad p \in \mathcal{P}, l \in \mathcal{L}, \quad (6)$$

$days, kg/kWh, kWh, kg/kWh, kWh \sim kg \quad P \times L$  constraints.

### Physical limit for energy-creation technologies installation

The available capacity of a given energy-creation technology, is limited by the physical space needed. This is a function of both building and technology configuration. For example, for PV technologies it would be the ratio roof/technology surfaces. Note that the function must return the appropriate units.

$$s_i^p \leq f(\text{BuildPars}, \text{TechPars}_i) \quad p \in \mathcal{P}, i \in \mathcal{I}, \quad (7)$$

$$kW \sim kW \quad P \times I \text{ constraints.}$$

### Physical limit for energy-absorbing technologies installation

The available capacity of a given energy-absorbing technology is limited by the physical space needed. This is a function of both building and technology configuration. Note that the function must return the appropriate units.

$$x_j^p \leq f(\text{BuildPars}, \text{TechPars}_j) \quad p \in \mathcal{P}, j \in \mathcal{J}, \quad (8)$$

$$kWh \sim kWh \quad P \times J \text{ constraints.}$$

### Efficiency constraint

Minimum efficiency required in the building.

$$\sum_{p \in \mathcal{P}} \sum_{m \in \mathcal{M}} \sum_{t \in \mathcal{T}} \sum_{k \in \mathcal{K}} \left( D_k^{p,m,t} + \sum_{m \in \mathcal{M}} \sum_{n \in \mathcal{N}} w_{k,n}^{p,m,t,mm} \right) \geq EF \cdot \sum_{p \in \mathcal{P}} \sum_{m \in \mathcal{M}} \sum_{t \in \mathcal{T}} e^{p,m,t} \quad (9)$$

$$kWh, kWh \sim kWh/kWh, kWh \quad \text{constraints.}$$

### Primary energy calculation

The primary energy (not from a fictitious market) consumed is the sum of the processed energy of each type and the one used as an input fuel.

$$e^{p,m,t} = \sum_{m \in \mathcal{M}} \sum_{k \in \mathcal{K}} \sum_{n \in \mathcal{N}} \left( u_{k,n}^{p,m,t,mm} \cdot B_{k,n} + \sum_{n \in \mathcal{N}} u_{k,n}^{p,m,t,mm} \right) \quad p \in \mathcal{P}, m \in \mathcal{M}, t \in \mathcal{T}, \quad (10)$$

$$kWh \sim kWh, kWh/kWh, kWh \quad P \times M \times T \text{ constraints.}$$

## 3.4.2. Operational Constraints

### Meet Loads

The energy supplied must meet the energy demand minus the energy saved due to absorbing technologies. It is composed of the energy produced with energy-creating technologies plus the energy purchased in the market minus the energy for sale, energy for storage and energy for production. On the demand side, the energy released from storage and the energy saved with passive technologies diminish the total demand.

These passive technologies can be modelled in two ways: on the one hand, the space-measurable ones and, on the other hand, the unitary-measurable ones. Regarding space-measurable passive technologies,  $\Phi$  is a function used to calculate the energy savings obtained by the inclusion of passive technologies such as room-enclosing opaque surfaces, energy-efficient windows, etc. Depending on the technology a particular calculation for  $\Phi_j$  is made

(using a closed-form formula or simulations). Notice that we include these technologies as additive different terms in the constraint (depending on the way  $\Phi_j$  is defined). With respect to unitary measurable passive technologies,  $OD_{k,j}$  is the proportion of savings in energy demand type  $k$  per unit of passive technology  $j$  available compared to a configuration of the building without passive technology  $j$ . For example, consider that we want to decide whether installing or not a new kind of lamps in a building whose lightning demand due to these lamps corresponds to a 10% of the total electricity demand  $D_k^{p,m,t}$ . If the current demand per lamp is  $75kW$  and, after installing technology  $j$ , the new demand will be  $50kW$ , then  $OD_{k,j} = \frac{75-50}{75} \cdot 0.10 = 0.033$ .

$$\begin{aligned}
& \sum_{i \in \mathcal{I}} z_{i,k}^{p,m,t} + \sum_{n \in \mathcal{N}_{B(k)}} \sum_{mm \in \mathcal{M}_A} u_{k,n}^{p,m,t,mm} \\
& - \sum_{i \in \mathcal{I}} y_{i,k}^{p,m,t} - \sum_{mm \in \mathcal{M}_S} \sum_{n \in \mathcal{N}_{S(k)}} w_{k,n}^{p,m,t,mm} \sum_{j \in \mathcal{J}_S} q_{k,j}^{p,m,t} \geq D_k^{p,m,t} \\
& - \sum_{j \in \mathcal{J}_S} q_{k,j}^{p,m,t} - \sum_{j \in \mathcal{J}_{PS}} \Phi_j^{p,m,t} - \sum_{j \in \mathcal{J}_{PU}} OD_{k,j} \cdot x_j^p \cdot D_k^{p,m,t} \quad p \in \mathcal{P}, m \in \mathcal{M}, t \in \mathcal{T}, k \in \mathcal{K},
\end{aligned} \tag{11}$$

$kWh, kWh, kWh, kWh, kWh \sim kWh, kWh, kWh, kWh/kWh, kWh, kWh \quad P \times M \times T \times K$  constraints.

### Technologies short-term availability

The energy that can be supplied by a technology is constrained by the availability of the technology.

$$z_{i,k}^{p,m,t} \leq DT \cdot A_i^{p,m,t} \cdot s_i^p \quad p \in \mathcal{P}, m \in \mathcal{M}, t \in \mathcal{T}, i \in \mathcal{I}, k = \mathcal{K}_{F(i)}, \tag{12}$$

$kWh \sim hours, kW/kW, kW \quad P \times M \times T \times I \times K[F(i)]$  constraints.

### Energy output calculation

The total energy output by each technology for a type of energy output is the sum of all the energy outputs over the energy inputs, computed as the energy input corrected by the conversion factor.

$$z_{i,kk}^{p,m,t} = \sum_{kk \in \mathcal{K}_{I(i)}} (E_{i,k,kk})^{-1} \cdot y_{i,k}^{p,m,t} \quad p \in \mathcal{P}, m \in \mathcal{M}, t \in \mathcal{T}, i \in \mathcal{I}, kk \in \mathcal{K}_{O(i)}, \tag{13}$$

$kWh \sim kWh/kWh, kWh \quad P \times M \times T \times I \times K[O(i)]$  constraints.

### Energy stored calculation

The energy stored each period is the energy stored in the previous period, plus the energy sent to storage, minus the energy released from storage. All flows are corrected by the losses ratio parameters.

$$r_{k,j}^{p,m,t} = OS_{k,j} \cdot r_{k,j}^{p,m,t-1} + OI_{k,j} \cdot qi_{k,j}^{p,m,t-1} - OO_{k,j} \cdot qo_{k,j}^{p,m,t-1}, \tag{14}$$

$$p \in \mathcal{P}, m \in \mathcal{M}, t \in \mathcal{T}, k \in \mathcal{K}, j \in \mathcal{J}_S,$$

$kWh \sim kWh/kWh, kWh, kWh/kWh, kWh, kWh/kWh, kWh \quad P \times M \times T \times K \times J[S]$  constraints.

### Energy discharging limit

The amount of energy that may be discharged from any energy-storage technology is limited

by the storage level.

$$qO_{k,j}^{p,m,t} \leq OR_{k,j} \cdot r_{k,j}^{p,m,t} \quad p \in \mathcal{P}, m \in \mathcal{M}, t \in \mathcal{T}, k \in \mathcal{K}, j \in \mathcal{J}_S, \quad (15)$$

$$kWh \sim kWh/kWh, kWh \quad P \times M \times T \times K \times J[S] \text{ constraints.}$$

### Energy storage lower limit

The amount of energy that may be stored from any energy-storage technology must be greater than the capacity installed corrected by the minimum charge allowed.

$$r_{k,j}^{p,m,t} \geq OA_{k,j} \cdot x_j^p \quad p \in \mathcal{P}, m \in \mathcal{M}, t \in \mathcal{T}, k \in \mathcal{K}, j \in \mathcal{J}_S, \quad (16)$$

$$kWh \sim kWh/kWh, kWh \quad P \times M \times T \times K \times J[S] \text{ constraints.}$$

### Energy storage upper limit

The amount of energy that may be stored in any energy-storage technology must be lower than the capacity installed, corrected by the maximum charge allowed.

$$r_{k,j}^{p,m,t} \leq OB_{k,j} \cdot x_j^p \quad p \in \mathcal{P}, m \in \mathcal{M}, t \in \mathcal{T}, k \in \mathcal{K}, j \in \mathcal{J}_S, \quad (17)$$

$$kWh \sim kWh/kWh, kWh \quad P \times M \times T \times K \times J[S] \text{ constraints.}$$

### 3.4.3. Integer Variables Constraints

#### Discrete sized technologies devices to install

The number of devices of discrete sized technologies to be installed must be integer.

$$si_i^p \in \mathbb{Z} \quad p \in \mathcal{P}, i \in \mathcal{I}_Z, \quad (18)$$

$$P \times I[Z] \text{ constraints.}$$

#### Discrete sized technologies devices to decommission

The number of devices of discrete sized technologies to be decommissioned must be integer.

$$sd_i^{p,q} \in \mathbb{Z} \quad p \in \mathcal{P}, i \in \mathcal{I}_Z, \quad (19)$$

$$P \times I[Z] \text{ constraints.}$$

### 3.4.4. Non-negativity Constraints

#### Non-negative energy supply

The energy supply must always be non-negative.

$$z_{i,k}^{p,m,t} \geq 0 \quad p \in \mathcal{P}, m \in \mathcal{M}, t \in \mathcal{T}, i \in \mathcal{I}, k \in \mathcal{K}, \quad (20)$$

$$kWh \sim kWh \quad P \times M \times T \times I \times K \text{ constraints.}$$

#### Positive generating devices to install

The number of devices to install must always be non-negative.

$$si_i^p \geq 0 \quad p \in \mathcal{P}, i \in \mathcal{I}, \quad (21)$$

$$devices \sim devices \quad P \times I \text{ constraints.}$$

### Positive devices to decommission

The number of devices to decommission must always be non-negative.

$$sd_i^{p,q} \geq 0 \quad p \in \mathcal{P}, i \in \mathcal{I}, q > p, \quad (22)$$

$$devices \sim devices \quad P \times I \times p \text{ constraints.}$$

### Positive available capacity

The total available capacity for each technology must always be non-negative.

$$s_i^p \geq 0 \quad p \in \mathcal{P}, i \in \mathcal{I}, \quad (23)$$

$$kW \sim kW \quad P \times I \text{ constraints.}$$

### Positive energy to store

The energy to be stored from the supply side must always be non-negative.

$$r_{k,j}^{p,m,t} \geq 0 \quad p \in \mathcal{P}, m \in \mathcal{M}, t \in \mathcal{T}, k \in \mathcal{K}, j \in \mathcal{J}, \quad (24)$$

$$kWh \sim kWh \quad P \times M \times T \times K \times J \text{ constraints.}$$

### Positive storage devices to install

The number of devices to install must always be non-negative.

$$xi_j^p \geq 0 \quad p \in \mathcal{P}, j \in \mathcal{J}, \quad (25)$$

$$kWh \sim kWh \quad P \times J \text{ constraints.}$$

### Positive storage devices to decommission

The number of devices to decommission must always be non-negative.

$$xd_j^{p,q} \geq 0 \quad p \in \mathcal{P}, j \in \mathcal{J}, \quad (26)$$

$$kWh \sim kWh \quad P \times J \text{ constraints.}$$

### Positive available storage capacity

The total available capacity by technology must always be non-negative.

$$x_j^p \geq 0 \quad p \in \mathcal{P}, j \in \mathcal{J}, \quad (27)$$

$$kWh \sim kWh \quad P \times J \text{ constraints.}$$

### Positive input energy

The input energy must always be non-negative.

$$y_{i,k}^{p,m,t} \geq 0 \quad p \in \mathcal{P}, m \in \mathcal{M}, t \in \mathcal{T}, i \in \mathcal{I}, k \in \mathcal{K}, \quad (28)$$

$$kWh \sim kWh \quad P \times M \times T \times I \times K \text{ constraints.}$$



### Positive purchased energy

The purchased energy must always be non-negative.

$$u_{k,n}^{p,m,t,mm} \geq 0 \quad p \in \mathcal{P}, m \in \mathcal{M}, t \in \mathcal{T}, k \in \mathcal{K}, n \in \mathcal{N}, mm \in \mathcal{M}_S, \quad (29)$$

$$kWh \sim kWh \quad P \times M \times T \times K \times N \times M[S] \text{ constraints.}$$

### Positive sold energy

The sold energy must always be non-negative.

$$w_{k,n}^{p,m,t,mm} \geq 0 \quad p \in \mathcal{P}, m \in \mathcal{M}, t \in \mathcal{T}, k \in \mathcal{K}, n \in \mathcal{N}, mm \in \mathcal{M}, \quad (30)$$

$$kWh \sim kWh \quad P \times M \times T \times K \times N \times M \text{ constraints.}$$

### Positive consumed energy

The consumed energy must always be non-negative.

$$e^{p,m,t} \geq 0 \quad p \in \mathcal{P}, m \in \mathcal{M}, t \in \mathcal{T}, \quad (31)$$

$$kWh \sim kWh \quad P \times M \times T \text{ constraints.}$$

## 3.5. Objective Function

In this Symbolic Model Specification, we show an objective function to minimize total costs, as this is the objective identified in previous stages of the project [3, 5]. Note that using the structure presented in this SMS, we can convert a constraint to a new objective function in the future, when new users of the DSS arise. For example, constraint (6) could be an objective function, becoming (32) a constraint up to a cost limit, or it can be an additional objective function in a multi-objective optimization model. In this way, if the new objective is the minimisation of pollutants, all we have to do is to change the code in Appendix A and modify the nature of (6) from “constraint” to “objective”.

### Minimize total cost

The objective is to minimize the total cost. It is composed of the cost of installing and maintaining technologies (both energy creating and energy absorbing), the operational cost of the technologies and the cost of purchasing energy in the market. The sales of energy and

subsidies are incomes that we have to subtract from the total cost.

$$\min \sum_{p \in \mathcal{P}} \left( \sum_{i \in \mathcal{I}} CI_i^{p,0} \cdot G_i \cdot si_i^p \right) \quad (32)$$

$$+ \sum_{j \in \mathcal{J}} CIS_j^{p,0} \cdot GS_j \cdot xi_j^p \quad (33)$$

$$+ \sum_{i \in \mathcal{I}} G_i \left( \sum_{a1=0}^p CD_i^{p-a1} \sum_{a2=a1+1}^p sd_i^{a1,a2} \right) \quad (34)$$

$$+ \sum_{j \in \mathcal{J}} \left( \sum_{a1=0}^p CDS_j^{p-a1} \sum_{a2=a1+1}^p xd_j^{a1,a2} \right) \quad (35)$$

$$+ \sum_{m \in \mathcal{M}} DM_m^p \sum_{i \in \mathcal{I}} \sum_{k \in \mathcal{K}} \sum_{t \in \mathcal{T}} CO_{i,k}^{p,m,t} \cdot z_{i,k}^{p,m,t} \quad (36)$$

$$+ \sum_{m \in \mathcal{M}} DM_m^p \sum_{j \in \mathcal{J}} \sum_{k \in \mathcal{K}} \sum_{t \in \mathcal{T}} COS_{k,j}^{p,m,t} \cdot r_{k,j}^{p,m,t} \quad (37)$$

$$- \sum_{m \in \mathcal{M}} DM_m^p \sum_{i \in \mathcal{I}} \sum_{k \in \mathcal{K}} \sum_{n \in \mathcal{N}_{S(k)}} \sum_{mm \in \mathcal{M}_A} \sum_{t \in \mathcal{T}} PP_{i,k,n}^{p,m,t} \cdot u_{k,n}^{p,m,t,mm} \quad (38)$$

$$- \sum_{m \in \mathcal{M}} DM_m^p \sum_{i \in \mathcal{I}} \sum_{k \in \mathcal{K}} \sum_{n \in \mathcal{N}_{S(k)}} \sum_{mm \in \mathcal{M}_S} \sum_{t \in \mathcal{T}} SP_{i,k,n}^{p,m,t} \cdot w_{k,n}^{p,m,t,mm} \quad (39)$$

$$- \sum_{i \in \mathcal{I}} SU_i^p \cdot G_i \cdot si_i^p \quad (40)$$

$\sim EUR/kW, kW/device, devices$

$, EUR/kW, kW/kW, kWh$

$, kW/device, EUR/kW, devices$

$, EUR/kWh, kWh$

$, days, EUR/kWh, kWh$

$, days, EUR/kWh, kWh$

$, days, EUR/kWh, kWh$

$, days, EUR/kWh, kWh$

$, EUR/kW, kW/device, devices$

1 objective.

The terms (32) and (33) are the total cost of installing and maintaining energy-creating and energy-absorbing technologies respectively. The terms (34) and (35) are the total cost of decommissioning energy-creating and energy-absorbing technologies, respectively. The terms (36) and (37) are the

total operational cost for energy-creating and energy-absorbing technologies respectively. The term (38) is the total cost of purchasing energy in the market. The term (39) is the total income from energy selling. The term (40) is the total income from subsidies.

## 4. Operational Symbolic Model Specification

The operational model deals with decisions involving the dispatch of energy by the installed technologies in the short term. As already mentioned, this model gets feedback from the strategic model by including the realisation of the strategic decisions made as parameters.

Given that operational constraints concerning lower-level performance of energy subsystems often depend on the specific building configuration and balance constraints, for completeness we reproduce in Subsection 4.1 the lower-level constraints provided in deliverable D2.2 [6], which will be added to the solvers during task 4.5 (Stochastic optimisation algorithms and solvers). A detailed description of the operational constraints and their motivation can be consulted in deliverable D2.2.

During the implementation of the model in subsequent stages of the project, the part-load efficiency of the technologies analysed will be taken into account by reshaping the constraints pertaining the suitable technologies as explained in Appendix C of deliverable D2.2 [6].

### 4.1. Lower-level Constraints

The low-level constraints are often specific for each building. Therefore, in some cases they may be defined with specific symbols different to the ones defined above. Thus, we include all the nomenclature used in these constraints. For a thorough explanation of each constraint, see deliverable D2.2 [6].

#### 4.1.1. Nomenclature

##### Time

$\delta$  length of operational decision-making period (s)

$\eta$  number of seconds in an hour (s/h)

$\mathcal{T}_O \subset \mathcal{T}$  set of short-term decision-making periods

$t$  short-term time period index

##### Physical Constants and Parameters

$\gamma_{\text{water}}$  specific heat capacity of water (kJ/kg·K)

$\rho_{\text{water}}$  density of water (kg/m<sup>3</sup>)

$\gamma_{\text{air}}$  specific heat capacity of air (kJ/kg·K)

$\rho_{\text{air}}$  density of air (kg/m<sup>3</sup>)

##### Environmental Parameters

$\chi^t$  external temperature during short-term period  $t \in \mathcal{T}_O$  (°C)

$\sigma^t$  fraction of maximum solar insolation incident (weighted average over different wall directions) during short-term period  $t \in \mathcal{T}_O$  (kW/m<sup>2</sup>)

## Building Parameters

- $\psi$  volume of the zone ( $\text{m}^3$ )
- $\nu$  heat transition coefficient of the wall ( $\text{kW}/\text{m}^2 \cdot \text{K}$ )
- $\alpha_{\text{wall}}$  heat transfer area of the wall ( $\text{m}^2$ )
- $\alpha_{\text{glass}}$  total area of windows ( $\text{m}^2$ )
  - $\epsilon$  mean energy transmission coefficient of glass
  - $\phi$  mean sun protection factor of all components of the thermal envelope of building
- $\alpha_{\text{floor}}$  area of the floor of the zone ( $\text{m}^2$ )
- $\lambda^t$  internal load (from people, lighting, working machines, etc.) per area of the zone during short-term period  $t \in \mathcal{T}_O$  ( $\text{kW}/\text{m}^2$ )
- $\underline{\kappa}^t$  lower limit for the required zone temperature during short-term period  $t \in \mathcal{T}_O$  ( $^\circ\text{C}$ )
- $\overline{\kappa}^t$  upper limit for the required zone temperature during short-term period  $t \in \mathcal{T}_O$  ( $^\circ\text{C}$ )

## Heating System Parameters

- $\zeta$  supply water temperature at the radiator inlet ( $^\circ\text{C}$ )
- $\iota$  maximum amount of heat that can be provided by the conventional heating system in any given short-term time period (kWh)
- $\overline{\mu}_{\text{water}}$  maximum water flow rate in radiator ( $\text{m}^3/\text{s}$ )
- $\underline{\mu}_{\text{water}}$  minimum water flow rate in radiator ( $\text{m}^3/\text{s}$ )
- $\xi$  mean nominal heat transfer capacity of all radiators installed (kW)
- $\varphi$  radiator coefficient
- $\varrho$  mean logarithmic temperature difference (K)

## HVAC System Parameters

- $\overline{\mu}_{\text{vent}}$  maximum air flow rate of the HVAC system ( $\text{m}^3/\text{s}$ )
- $\underline{\mu}_{\text{vent}}$  minimum air flow rate of the HVAC system ( $\text{m}^3/\text{s}$ )
- $\underline{\tau}$  lower limit of the proportion of air that may be taken externally
- $\overline{\tau}$  upper limit of the proportion of air that may be taken externally
- $E_{\text{HVAC,electricity,cooling}}$   
electricity required to produce one unit of cooling ( $\text{kWh}_e/\text{kWh}$ )
- $E_{\text{boiler,NG,hot water}}$   
NG required by the boiler to produce one unit of hot water ( $\text{kWh}/\text{kWh}$ )
- $\omega$  electricity required to pump the air at a given flow rate ( $\text{kWh}_e/(\text{m}^3/\text{s})$ )
- $\overline{\chi}$  external temperature limit at which the AHU performs cooling ( $^\circ\text{C}$ )
- $\underline{\chi}$  external temperature limit at which the AHU performs heating ( $^\circ\text{C}$ )
- $\overline{\varsigma}$  AHU's supply-air temperature for heating ( $^\circ\text{C}$ )
- $\underline{\varsigma}$  AHU's supply-air temperature for cooling ( $^\circ\text{C}$ )

## Energy Market Parameters

$CP_{k,n}^t$  price of energy type  $k \in \mathcal{K}$  purchased from market  $n \in \mathcal{N}_{B,k}$  during short-term time period  $t \in \mathcal{T}$  (€/kWh or €/kWh<sub>e</sub>)

## State Variables

$\Gamma^t$  return water temperature at the outlet of the radiator during short-term time period  $t \in \mathcal{T}_O$  (°C)

## Decision Variables

$\Lambda^t$  zone temperature during short-term time period  $t \in \mathcal{T}_O$  (°C)

$\Omega_{\text{water}}^t$  flow rate for water to conventional heating system during short-term time period  $t \in \mathcal{T}_O$  (m<sup>3</sup>/s)

$\Omega_{\text{vent}}^t$  flow rate for air to HVAC system during short-term time period  $t \in \mathcal{T}_O$  (m<sup>3</sup>/s)

$\Upsilon^t$  supply-air temperature from the HVAC system's air-handling unit (AHU) during short-term time period  $t \in \mathcal{T}_O$  (°C)

$\Phi^t$  fraction of external air used by the AHU during short-term time period  $t \in \mathcal{T}_O$

$\Psi^t$  heat from radiator during short-term time period  $t \in \mathcal{T}_O$  (kWh)

$D_k^t$  demand for end-use energy type  $k = \{\text{space heat, cooling}\}$  during short-term time period  $t \in \mathcal{T}_O$  (kWh)

$y_{i,k}^t$  requirement of energy type  $k = \{\text{electricity}\}$  as input to energy-creating technology  $i = \{\text{HVAC}\}$  during short-term time period  $t \in \mathcal{T}_O$  (kWh<sub>e</sub>)

### 4.1.2. Constraints

#### Computation of zone temperature

$$\Lambda^t = \left( \frac{1}{\frac{\gamma_{\text{air}} \cdot \rho_{\text{air}} \cdot \psi}{\delta} + \nu \cdot \alpha_{\text{wall}} + \Omega_{\text{vent}}^t \cdot \rho_{\text{air}} \cdot \gamma_{\text{air}}} \right) \cdot \left[ \frac{\gamma_{\text{air}} \cdot \rho_{\text{air}} \cdot \psi}{\delta} \cdot \Lambda^{t-1} + \Psi^t \cdot \frac{\eta}{\delta} + \nu \cdot \alpha_{\text{wall}} \cdot \chi^{t-1} + \sigma^{t-1} \cdot \epsilon \cdot \phi \cdot \alpha_{\text{glass}} + \lambda^{t-1} \cdot \alpha_{\text{floor}} + \rho_{\text{air}} \cdot \gamma_{\text{air}} \cdot \Omega_{\text{vent}}^t \cdot \Upsilon^t \right], \forall t \in \mathcal{T}_O \quad (41)$$

#### Zone temperature constraint

$$\underline{\kappa}^t \leq \Lambda^t \leq \bar{\kappa}^t, \forall t \in \mathcal{T}_O \quad (42)$$

#### Transmission of heat from radiators

$$\Psi^t = \frac{\delta}{\eta} \cdot \xi \cdot \left( \frac{(\zeta - \Gamma^t)}{\ln \left( \frac{\zeta - \Lambda^t}{\Gamma^t - \Lambda^t} \right)} \cdot \frac{1}{\varrho} \right)^\varphi, \forall t \in \mathcal{T}_O \quad (43)$$

#### Heat exchange inside radiators

$$\Psi^t = \frac{\delta}{\eta} \cdot \Omega_{\text{water}}^t \cdot \rho_{\text{water}} \cdot \gamma_{\text{water}} \cdot (\zeta - \Gamma^t), \forall t \in \mathcal{T}_O \quad (44)$$

### Heat required inside the boiler to change temperature

$$D_{\text{space heat}}^t = \frac{\delta}{\eta} \cdot \Omega_{\text{water}}^t \cdot \rho_{\text{water}} \cdot \gamma_{\text{water}} \cdot (\zeta - \Gamma^{t-1}), \forall t \in \mathcal{T}_O \quad (45)$$

### Return water temperature constraint

$$\Lambda^t \leq \Gamma^t \leq \zeta, \forall t \in \mathcal{T}_O \quad (46)$$

### Maximum heat demand allowed

$$D_{\text{space heat}}^t \leq \iota, \forall t \in \mathcal{T}_O \quad (47)$$

### Water flow rate constraint

$$\underline{\mu}_{\text{water}} \leq \Omega_{\text{water}}^t \leq \bar{\mu}_{\text{water}}, \forall t \in \mathcal{T}_O \quad (48)$$

### Setting of the supply-air temperature for the AHU

$$\Upsilon^t = \begin{cases} \Phi^t \cdot \chi^{t-1} + (1 - \Phi^t) \cdot \Lambda^{t-1} & \text{if ventilation only} \\ \bar{\zeta} & \text{if cooling and } \chi^{t-1} < \underline{\chi} \\ \bar{\zeta} + \left( \frac{\underline{\zeta} - \bar{\zeta}}{\bar{\chi} - \underline{\chi}} \right) \cdot (\chi^{t-1} - \underline{\chi}) & \text{if cooling and } \underline{\chi} \leq \chi^{t-1} < \bar{\chi} \\ \underline{\zeta} & \text{if cooling and } \bar{\chi} \leq \chi^{t-1} \end{cases}, \forall t \in \mathcal{T}_O \quad (49)$$

### Cooling demand calculation

$$D_{\text{cooling}}^t = \Omega_{\text{vent}}^t \cdot \rho_{\text{air}} \cdot \gamma_{\text{air}} \cdot \frac{\delta}{\eta} \cdot (\Phi^t \cdot \chi^{t-1} + (1 - \Phi^t) \cdot \Lambda^{t-1} - \Upsilon^t), \forall t \in \mathcal{T}_O \quad (50)$$

### Electricity for ventilation or cooling calculation

$$y_{\text{HVAC,electricity}}^t = \begin{cases} \omega \cdot \Omega_{\text{vent}}^t & \text{if ventilation only} \\ E_{\text{HVAC,electricity,cooling}} \cdot D_{\text{cooling}}^t & \text{if cooling} \end{cases}, \forall t \in \mathcal{T}_O \quad (51)$$

### Proportion of external air taken in by the AHU constraint

$$\underline{\tau} \leq \Phi^t \leq \bar{\tau}, \forall t \in \mathcal{T}_O \quad (52)$$

### AHU's air-flow rate during each period constraint

$$\underline{\mu}_{\text{vent}} \leq \Omega_{\text{vent}}^t \leq \bar{\mu}_{\text{vent}}, \forall t \in \mathcal{T}_O \quad (53)$$

## 4.2. Operational objective function

For the operational model, different objective functions could be defined depending on the goals of the managers of the facilities where the model is applied to. For instance, if the problem of interest involves the heating and cooling system, we can restrict the scope of the problem at the operational level to minimising energy consumption or costs while meeting the site's temperature requirements for each period (see deliverable D2.2 of the project). In this case, the operational objective function would be:

$$\min \sum_{t \in \mathcal{T}_O} (CP_{\text{electricity}}^t \cdot y_{\text{HVAC,electricity}}^t + CP_{\text{heat}}^t \cdot D_{\text{space heat}}^t) \quad (54)$$

As already mentioned, it is straightforward to define more complex or specific objective functions from the parameters and variables defined as part of the operational model. Since here we describe here a general SMS, we outline an example of an objective function and during the implementation more complex options will be available for the decision makers.

## 5. Stochastic Model

Stochastic optimisation is the main risk management tool applied in the EnRiMa project. The risk management will be tackled in two ways:

1. Managing uncertainties by including the variability of the parameters through scenario generation.
2. Managing the specific risk assumptions for different public buildings.

The EnRiMa models deal with the first risk management approach through the scenario generation tool described in [7] and introduced in Sec. 2. The information provided by the scenario generation tool condenses the variability of the stochastic parameters in order to find the solution taken into account the most probable scenarios.

The importance of the second risk management approach relies in the fact that some public buildings can assume fewer risks than others. For example, we have a wide range of types of public buildings, namely: leisure, education, health care, governmental, etc, with a different risk perception. Similarly, two similar buildings devoted to store materials may have different points of view about risks, depending on the kind of materials stored; let us say medicines or food. Therefore, the users of the EnRiMa DSS may have different risk perception. In decision theory, the underlying concept to deal with this fact is “risk aversion”. Our stochastic models include a risk weight term  $\beta$ ,  $0 \leq \beta \leq 1$  reproducing this information. All in all, the solution provided by the EnRiMa DSS embed the available information concerning risks regarding both approaches.

We can identify uncertainty for parameters in both the operational and the strategic DSSs. The main uncertain parameters affecting operational decisions are energy loads (demands), building occupancy, weather (in particular, temperature, wind speed, and solar irradiation), and short-term variations of energy prices. The latter parameter concerns foremost electricity prices in the case of real-time pricing. As energy loads are strongly correlated with building occupancy and weather, they can be derived from these parameters as described in deliverable D3.1. For the strategic DSS, recall that the embedded operational relations shall give indications about how the installed equipment would perform under various conditions, i.e., is it robust and flexible enough to satisfy daily user needs and to deliver acceptably under extraordinary conditions? In addition, the relations help to estimate the operating costs of the chosen investments. Hence, scenarios for the corresponding uncertain operational parameters are determined by selecting series of critical and typical values (profiles) from observations. Finally, uncertain parameters directly affecting the strategic decisions compose technology efficiency and prices (including availability of new technologies), government subsidies for some equipment, and new tariffs, in particular, the availability of real-time pricing contracts. Also, the long-term development (trend) of energy prices is considered uncertain. Note that the latter parameter needs to be combined with the selected short-term price profiles, for example, as described in deliverable D3.2. For a more detailed description of how the scenarios for the single uncertain parameters are created and of suitable scenario tree structures we refer to the deliverables in WP 3, D3.1 and D3.2.

### 5.1. EnRiMa Stochastic Modelling

The previously described models lead in their simpler version to a linear model. This model will take the standard linear programming program formulation:

$$\min_x \quad C^T x \quad (55)$$

$$\text{subject to} \quad Ax = B \quad (56)$$

$$x \geq 0, \quad (57)$$

where  $x$  is a vector containing all the variable expressions of the model, and  $C$ ,  $A$ , and  $B$  are the coefficients of the model. From the point of view of the SMS, such coefficients are parameters or combinations of parameters in the SMS. The  $C$  matrix, called “cost matrix”, contains the coefficients affecting the objective function; the  $B$  matrix contains the so-called “right-hand-side” coefficients, that is, those that are not multiplying any decision variable in the constraints; and the  $A$  matrix, known as the “technology matrix”, contains the coefficients affecting the decision variables in the constraints. For the sake of simplicity we will follow this simplified formulation in the following explanations. This formulation is specially understandable in order to be straightforwardly extended to its stochastic version.

As already mentioned, since the uncertainties may seriously affect long-term decisions, the model should be formulated as a stochastic optimization problem. Among the main stochastic parameters are fuel prices and demands. In the presence of uncertainties, there is a dilemma to choose an efficient technological portfolio in real time while pursuing long-term goals. Therefore, the solution of the problem involves the so-called multi-stage dynamic stochastic optimization models with a rolling horizon. However, for the sake of clarity we will describe here a two-stage dynamic version of the model, where some decisions (so-called first-stage decisions) regarding investments into new energy technologies have to be taken before uncertainties are resolved and some others (so-called second-stage decisions) will be taken once values for uncertain parameters become known, thereby providing a trade-off between long- and short-term decisions.

Let us consider the simplest EnRiMa model. This would be a dynamic stochastic strategic planning model where the goal would be to find a combination of technologies installed at the beginning of each year  $t$  so that the mixture of these technologies operating in each period  $j$  would ensure a safe energy “provision” plan minimizing investment costs  $C_i^t$  (which in the more complex model described in Sec. 3 would correspond with the parameter  $CI_i^{p,a}$ ), deterministic equipment maintenance costs  $c_{ij}^t$  (which in the more complex model described in Sec. 3 would correspond with the parameter  $CO_{i,k}^{p,m,t}$ ) and stochastic operation costs which may include stochastic fuel prices  $P_{ij}(\omega)$  (which in the more complex model described in Sec. 3 would correspond with the parameter  $CP_{k,n}^{p,m,t}$ ). Notice that in this simplified version of the model instead of dividing periods in long- mid- and short-term periods, we just consider years and operational periods (e.g. hours). In summary, to show the stochastic extension of the SMS we consider just two variables: equipment installation (that we will denote by  $x$ ) and energy generation (that we will denote by  $y$ ). The inclusion of more variables and periods leading to the use of the models defined above is straightforward, but the notation would be less clear than the one used in the following description for this simpler model. Schematically, the structure of the modeling framework is presented in Fig 5.

With this notation, constraints (11,12, 20) become now:

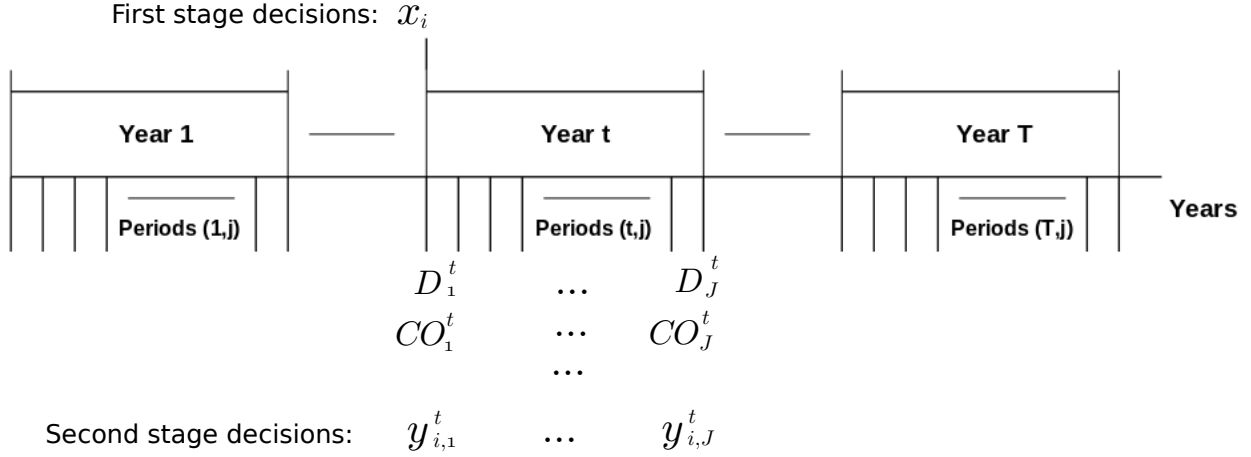
$$\sum_{i=1}^I y_{ij}^t \geq D_j^t \quad \forall j, t \quad (58)$$

$$y_{ij}^t \leq A_{ij}^t s_{ij}^t \quad \forall i, j, t \quad (59)$$

$$y_{ij} \geq 0 \quad (60)$$

In constraints (58)-(60), instead of the deterministic demand  $D_j^t$  we have now to satisfy stochastic demand scenarios  $D_j^t(s)$  for scenarios  $s = 1, \dots, S$ . Since this means that for a given in advance





**Figure 5:** Schematic representation of the two-stage dynamic stochastic strategic model for robust planning energy efficiency and risk management in public buildings

(ex-ante) long-term investments into technologies  $x_i^t$  equation (58) may not be fulfilled for some  $t$ , we assume a worst case scenario and substitute  $D_j^t(s)$  by  $\max_s D_j^t(s)$ . This would be a very conservative and costly assumption. Instead, we introduce in (60) a so-called back-up technology  $n + 1$  (basically, a slack variable that allows the fulfilling of the constraint). The “stock” of this backup technology will be unbounded (always available) however it may come at high price. Thus, equation (58) is modified to:

$$\sum_i y_{ij}^t(\omega^s) + y_{n+1j}^t(\omega^s) = D_j^t(\omega^s) \quad \forall s, j, t. \quad (61)$$

In this way, by using Conditional Value at Risk (CVaR), the dynamic stochastic model is formulated now as follows:

$$\min_{x,y} f(x, y) = \sum_t \sum_i C_i^t x_i^t + \sum_t \mathbb{E} \left[ \sum_j \sum_i c_{ij}^t y_{ij}^t + \sum_j c_{n+1j}^t y_{n+1j}^t \right] \quad (62)$$

under constraint (61) and safety constraints:

$$\mathbb{P} \left[ \sum_i P_{ij}^t(\omega) y_{ij}^t < q_j^t \right] < 1 - \alpha \quad \forall j, t \quad (63)$$

where  $\alpha$  is the safety level of the solution and  $q_j^t$  denotes the value at risk (VaR), that is, the maximum acceptable cost for safety level  $\alpha$ , see [2] for the details. The problem can be reformulated as:

$$\min_{x,y} f(x, y) + \beta \sum_t \left( \sum_j \left( q_j^t + \mathbb{E} \left[ \max\{0, \sum_i P_{ij}^t(\omega) y_{ij}^t q_t^j\} \right] \right) \right), \quad (64)$$

where  $0 \leq \beta \leq 1$  is the term that represents the risk aversion described above. Using discrete random scenarios of uncertain parameters demand ( $D_j^{ts}$ ) and prices ( $P_{ij}^{ts}$ ), where the  $s$  in the superindex denotes the scenario, the previous objective function is substituted by its linear approximation:

$$\begin{aligned} & \min_{x,y} \sum_t \sum_i C_i^t x_i^t & (65) \\ & + \sum_t S^{-1} \sum_t \sum_s \left( \sum_j \sum_i c_{ij}^t y_{ij}^{ts} + \sum_j c_{n+1j}^t y_{n+1j}^{ts} \right) + \\ & \beta \sum_t \sum_j \left( q_j^t + (1 - \alpha)^{-1} \sum_s p_j^{ts}(\omega) z_j^{ts} \right) & (66) \end{aligned}$$

under constraints

$$z_j^{ts} \geq \sum_i P_{ij}^t(\omega^s) y_{ij}^t q_i^j \quad \forall s, i, t, j, \quad (67)$$

where  $p_j^{ts}$  in eq. (65) is the probability of scenario  $s$  in time  $t$  and period  $j$ .

Note that all we have to do in the EnRiMa SMS to extend the deterministic approach the the stochastic programming formulation, is adding a new index for the scenarios to the stochastic parameters besides the probabilities of the scenarios.

## 5.2. Stochastic Transition Matrices

When adding scenarios, the problem size increases. Let eq. (66) be denoted as:

$$R(y) = \sum_t \sum_j \left( q_j^t + (1 - \alpha)^{-1} \sum_s p_j^{ts}(\omega) z_j^{ts} \right). \quad (68)$$

Notice that  $z_j^{ts}$  are implicitly defined within the  $y_{ij}^t$ s and therefore, for the sake of simplicity, we denote the risk term as  $R(y)$ . Now, the problem size increases, and takes the matrix form:

$$\min_x \quad C^T x + P^T y + \beta R(y) \quad (69)$$

$$\text{subject to} \quad Ax = B \quad (70)$$

$$Tx + Wy = H \quad (71)$$

$$x \geq 0 \quad (72)$$

$$y \geq 0. \quad (73)$$

where we have divided the  $x$  variables in model (55)-(57) into the variables corresponding to the stages of the stochastic problem ( $x$  and  $y$ ) and added the new constraints.

The stochastic transition matrices can be explicitly represented by a flow matrix where each scenario is visualised within the overall model, with the transfer coefficients inside, that is:

$$\begin{array}{cccccc}
& x & y & y & \dots & y \\
T^1 & \boxed{W} & & & & H^1 \\
T^2 & & \boxed{W} & & & H^2 \\
\vdots & & & & \dots & \dots \\
T^n & & & & \boxed{W} & H^n
\end{array}$$

Notice that the stochastic transition matrix depends on the specific implementation and solution of the stochastic optimisation problem, and therefore it will be explicitly generated at the time when the problem is being solved. It is important to remark that the core work in this SMS is the modelling of the deterministic problem. Once this has been done, the previous matrix is generated using the scenarios generated for the stochastic parameters. The stochasticity information in Table 1.

It is important to remark that some parameters may be stochastic or deterministic depending on the specific configuration of the building at hand. These parameters are referred in the table as “either” since their nature may have both considerations.

### 5.3. Stochastic Sankey Diagrams

Sankey diagram charting is a powerful tool for representing energy flows. A Sankey diagram shows the different energy flows as arrows whose width is proportional to the amount of energy. Input energy flows from the left side become output energy flows as they cross the technologies in the building.

Sankey diagrams for the current configuration of the test sites were delivered in [4]. Furthermore, an on-line tool has been developed by the EnRiMa partner CET, available at [http://www.cet.or.at/enrima/sankey\\_en.php](http://www.cet.or.at/enrima/sankey_en.php). Sankey diagrams like the one in Fig. 6 for the back-up site ENERGYbase in Vienna, Austria, can be generated for each past day using real data. See [6] for a description of the tool.

A Sankey diagram can be produced using historical data to analyse the performance of a building during a specific period of time in the past. We can also generate a foreseen Sankey diagram with the solution of a given optimisation problem, let us name them “solution Sankey diagrams”. Thus, the represented energy flows will correspond to the decisions made and the values of the parameters. As we have a dynamic stochastic optimisation model with several periods, the solution Sankey diagram can be referred to the whole period considered in the problem, that is, throughout all the period indices, or to a sub-period of the whole time span, e.g. each year. We go now one step beyond and define the “stochastic Sankey diagrams” as a set of Sankey diagrams related to the solutions of a stochastic optimisation problem for different “risk aversion” environments, that is, different values of the  $\beta$  risk term introduced in eq. (64). Consequently, we can visualise the energy flows that will result as the solution of the problem considering all the scenarios, for several decision maker’s skills.

The stochastic Sankey diagrams depend on the specific implementation and solution of the stochastic optimisation problem. Therefore, they will be generated along with the solver results in task 4.6 (Stochastic optimisation algorithms and solvers), and implemented in the DSS GUI. As an example, we show in Fig. 7 an artificial Sankey diagram for hypothetical solutions of a stochastic formulation of the example described in Appendix. B.

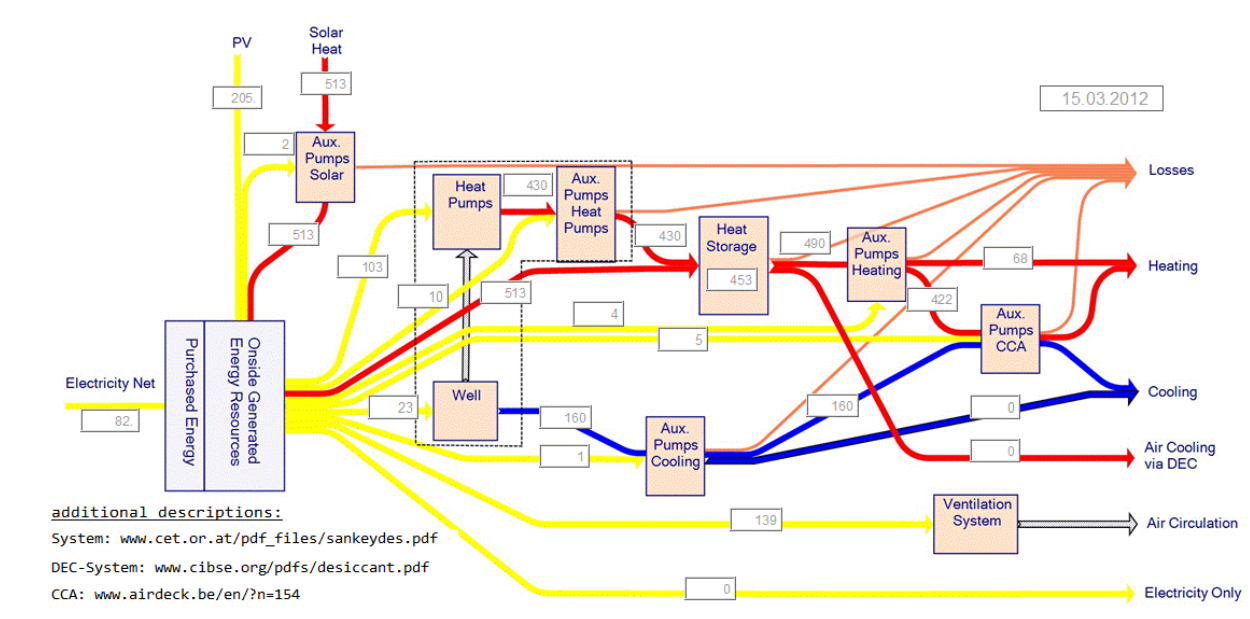
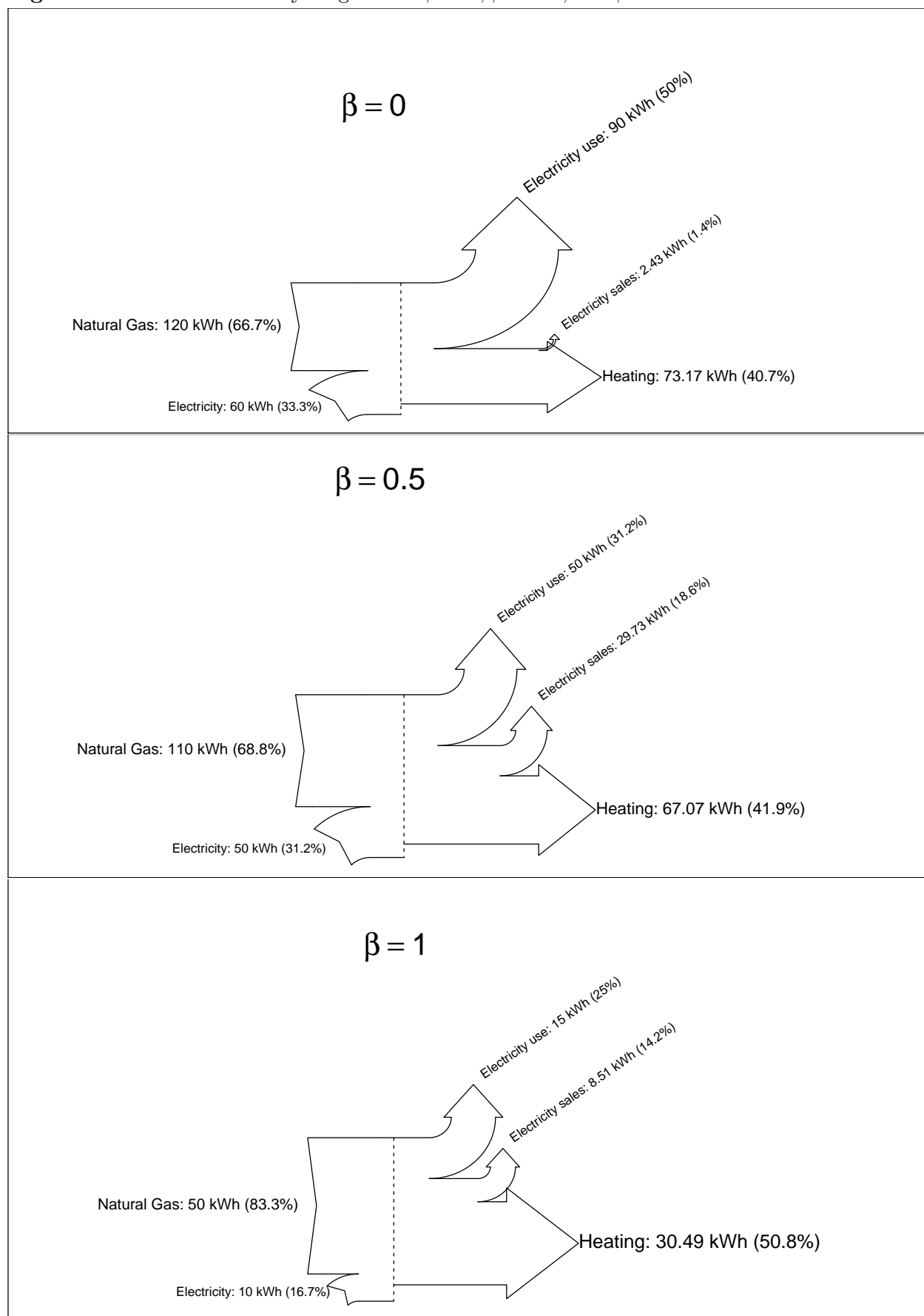


Figure 6: Sankey diagram for EnRiMa's back-up site ENERGYbase

**Table 1:** Parameters for stochastic modelling

id	symbol	nature	tag	units	group	expr	dataType
1	E	Deterministic	convCoeff	$kWh/kWh$	Technologies	$E_{i,k,kk}$	dec
2	D	Stochastic	enerDemand	$kWh$	Demand	$D_k^{p,m,t}$	dec
3	AG	Deterministic	ageFactorGen	$kW/kW$	Technologies	$AG_i^a$	dec
4	CO	Either	opCost	$EUR/kWh$	Policy	$CO_{i,k}^{p,m,t}$	dec
5	CI	Either	techCost	$EUR/kW$	Finance	$CI_i^{p,a}$	dec
6	CD	Either	decomCost	$EUR/kW$	Finance	$CD_i^a$	dec
7	DM	Deterministic	profDur	$days$	Model	$DM_m^p$	int
8	SU	Stochastic	subsInc	$EUR/kW$	Policy	$SU_i^p$	dec
9	LS	Deterministic	stoLoss	$kWh$	Technologies	$LS_i^{p,m,t}$	dec
10	SC	Deterministic	stoCap	$kWh$	Technologies	$SC_j^p$	dec
11	A	Stochastic	techAvail	$kW/kW$	Technologies	$A_i^{p,m,t}$	dec
12	IL	Deterministic	InvLim	$EUR$	Finance	$IL^p$	dec
13	H	Deterministic	polUnit	$kg/kWh$	Technologies	$H_{i,k,l}$	dec
14	PL	Either	polLimit	$kg$	Policy	$PL_l^p$	int
15	DT	Deterministic	timeDur	$hours$	Model	$DT$	dec
16	HO	Deterministic	operPeriods	$number$	Building	$HO$	int
17	RA	Deterministic	roofArea	$m^2$	Building	$RA$	dec
18	BA	Deterministic	buildArea	$m^2$	Building	$BA$	dec
19	BV	Deterministic	builVol	$m^3$	Building	$BV$	dec
20	G	Deterministic	NomCap	$kW/device$	Technologies	$G_i$	dec
21	SP	Stochastic	sellPrice	$EUR/kWh$	Finance	$SP_{i,k,n}^{p,m,t}$	dec
39	PP	Stochastic	PurchPrice	$EUR/kWh$	Policy	$PP_{i,k,n}^{p,m,t}$	dec
22	EF	Deterministic	effBuilding	$kWh/kWh$	Finance	$EF$	dec
23	OD	Deterministic	demRed	$kWh/kWh$	Technologies	$OD_{k,j}$	dec
24	C	Deterministic	polUnit	$kg/kWh$	Technologies	$C_{i,l,n}$	dec
25	AS	Deterministic	ageFactorSto	$kW/kW$	Technologies	$AS_i^a$	dec
26	GS	Deterministic	ageCapSto	$kW/kW$	Technologies	$GS_j$	dec
27	CIS	Either	techStoCost	$EUR/kW$	Finance	$CIS_j^{p,a}$	dec
28	COS	Either	techStoOperCost	$EUR/kWh$	Finance	$COS_{k,j}^{p,m,t}$	dec
29	CDS	Either	decomStoCost	$EUR/kWh$	Finance	$CDS_j^a$	dec
30	CP	Either	purchCost	$EUR/kWh$	Finance	$CP_{k,n}^{p,m,t}$	dec
31	B	Deterministic	primEner	$kWh/kWh$	Energy Efficiency	$B_{k,n}$	dec
32	FA	Deterministic	floorArea	$m^2$	Building	$FA$	dec
33	OS	Deterministic	storeAvail	$kWh/kWh$	Technologies	$OS_{k,j}$	dec
34	OI	Deterministic	storeAvail	$kWh/kWh$	Technologies	$OI_{k,j}$	dec
35	OO	Deterministic	storeAvail	$kWh/kWh$	Technologies	$OO_{k,j}$	dec
36	OR	Deterministic	availDischarging	$kWh/kWh$	Technologies	$OR_{k,j}$	dec
37	OA	Deterministic	lowerCharge	$kWh/kWh$	Technologies	$OA_{k,j}$	dec
38	OB	Deterministic	upperCharge	$kWh/kWh$	Technologies	$OB_{k,j}$	dec

**Figure 7:** A stochastic Sankey diagram for  $\beta = 0$ ,  $\beta = 0.5$ , and  $\beta = 1$ .



## 6. Conclusions

This deliverable contains the Symbolic Model Specification for the EnRiMa DSS. It has been developed in a sequential way from the “atomic” elements of the models up to the stochastic models. The different elements of the models such as variables, parameters and their relations have been defined and represented both as mathematical equations and data-driven entities, ready to use in forthcoming stages of the project. As innovative results in this deliverable we have defined stochastic Sankey diagrams, which will be visually improved in the final DSS GUI. The stochastic model described gathers the main risk management issues tackled in the DSS, extending the optimisation models (strategic and operational) described previously and based in the research carried out so far in the EnRiMa project. The SMS has been illustrated through the implementation in a simulated building with a basic configuration.

The Symbolic Model Specification contained in this deliverable (D4.2), along with the related deliverables D2.2 and D3.2, provides the framework for the next steps of the EnRiMa project, especially the development of the DSS Engine and the GUI. Particularly, task 4.5 (Stochastic optimisation algorithms and solvers) will continue this work building the complete implementation for the test sites and selecting algorithms and computational requirements afterwards. This work will be carried out in close cooperation with the one in task 4.6 and WP5, which will be fed with the structure defined herein.

## Acknowledgments

This document has been developed by the partners of the EnRiMa project, funded by the 7<sup>th</sup> Framework Programme (FP7) of the European Commission (EC), project number 260041. The responsible partner of the deliverable has been URJC, with valuable contributions by CET, IIASA and UCL. The rest of the partners are acknowledged for their valuable comments.

The manuscript has been subject to an internal project quality control by CET and SINTEF.

## References

- [1] Michel Berkelaar and others. *lpSolve: Interface to Lp\_solve v. 5.5 to solve linear/integer programs*, 2011. URL <http://CRAN.R-project.org/package=lpSolve>. R package version 5.6.6.
- [2] A.J. Conejo, M. Carrión, and J.M. Morales. *Decision Making Under Uncertainty in Electricity Markets*. International Series in Operations Research and Management Science Series. Springer, 2010. ISBN 9781441974204. URL [http://books.google.es/books?id=zta0qWS\\_W98C](http://books.google.es/books?id=zta0qWS_W98C).
- [3] HCE, IIASA, SU, UCL, URJC, SINTEF, CET, and Tecnia. Deliverable 1.1: Requirement assessment. Project Deliverable, 12 2011. URL [https://project.sintef.no/eRoomReq/Files/TS-001/EnRiMa/0\\_5ed1/d1\\_1.pdf](https://project.sintef.no/eRoomReq/Files/TS-001/EnRiMa/0_5ed1/d1_1.pdf).
- [4] HCE, IIASA, SU, UCL, URJC, SINTEF, CET, and Tecnia. Deliverable 2.1: Sankey diagrams that link the energy resources to the loads. Project Deliverable, 12 2011. URL [https://project.sintef.no/eRoomReq/Files/TS-001/EnRiMa/0\\_5a04/28%200ct%202011%20Deliverable%20D2.1%20Submitted.pdf](https://project.sintef.no/eRoomReq/Files/TS-001/EnRiMa/0_5a04/28%200ct%202011%20Deliverable%20D2.1%20Submitted.pdf).
- [5] HCE, IIASA, SU, UCL, URJC, SINTEF, CET, and Tecnia. Deliverable 4.1: Requirement analysis. Project Deliverable, 10 2011. URL [https://project.sintef.no/eRoomReq/Files/TS-001/EnRiMa/0\\_5a19/d4\\_1.pdf](https://project.sintef.no/eRoomReq/Files/TS-001/EnRiMa/0_5a19/d4_1.pdf).
- [6] HCE, IIASA, SU, UCL, URJC, SINTEF, CET, and Tecnia. Deliverable 2.2: Mathematical formulation of energy balance and flow constraints. Project Deliverable, 03 2012.
- [7] HCE, IIASA, SU, UCL, URJC, SINTEF, CET, and Tecnia. Deliverable 3.2: Scenario generation software tool. Project Deliverable, 03 2012.
- [8] Douglas E. King and M. Granger Morgan. Customer-focused assessment of electric power microgrids. *Journal of Energy Engineering*, 133(3):150–164, 2007. doi: 10.1061/(ASCE)0733-9402(2007)133:3(150). URL <http://link.aip.org/link/?QEY/133/150/1>.
- [9] Friedrich Leisch. Sweave: Dynamic generation of statistical reports using literate data analysis. In Wolfgang Härdle and Bernd Rönz, editors, *Compstat 2002 — Proceedings in Computational Statistics*, pages 575–580. Physica Verlag, Heidelberg, 2002. URL <http://www.stat.uni-muenchen.de/~leisch/Sweave>. ISBN 3-7908-1517-9.
- [10] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2011. URL <http://www.R-project.org/>. ISBN 3-900051-07-0.
- [11] Michael Stadler, Chris Marnay, Afzal Siddiqui, Judy Lay, Brian Coffey, and Hirohisa Aki. A study of commercial buildings in california and new york states. Technical report, Lawrence Berkeley National Laboratory, December 2008.
- [12] Michael Stadler, Chris Marnay, Gonçalo Carsoso, Tim Lipman, Oliver M’egel, Srirupa Ganguly, Afzal Siddiqui, and Judy Lay. The CO<sub>2</sub> abatement potential of california’s mid-sized commercial buildings. Technical report, California Energy Commission, December 2009.



## A. Data Tables

### Symbolic Model

The tables in this subsection contain the model objects (indices, variables and parameters). Descriptions are dropped for the sake of space. The complete descriptions are in Sec. 3.

**Table 2:** Sets and Indices

id	symbol	isSet	ref	dif	short	corr	rep	fromNull	greater
1	p	TRUE			long-term period			FALSE	
2	m	TRUE			mid-term period			FALSE	
3	t	TRUE			short-term period			FALSE	
4	i	TRUE			energy-creating technology			FALSE	
6	k	TRUE			energy type			FALSE	
7	l	TRUE			pollutant			FALSE	
8	q	FALSE	p	-1	decommissioning period	0.5	TRUE	FALSE	TRUE
9	a	FALSE	p	0	technology age	1.0	FALSE	TRUE	FALSE
10	j	TRUE			energy-absorbing technology			FALSE	
11	a1	FALSE	p	0	auxiliary index	1.0	FALSE	FALSE	FALSE
12	a2	FALSE	p	0	auxiliary index	1.0	FALSE	FALSE	FALSE
14	n	TRUE			energy market			FALSE	FALSE
15	mm	FALSE	m	0	ancestor mid-term period	1.0	FALSE	FALSE	FALSE
16	kk	FALSE	k	0	output energy type	1.0	FALSE	FALSE	FALSE

**Table 3:** Subsets

id	set	subset	desc
1	4	E	Technologies that generate electricity
3	4	H	Technologies that generate heat
5	4	C	Technologies that generate cooling
7	6	E	Electricity
8	6	H	Heat
9	6	C	Cooling
10	6	G	Natural Gas
11	6	S	Solar
12	10	S	Storage technologies
13	10	PS	Space measurable passive technologies
14	10	PU	Unitary measurable passive technologies
15	14	E	Electricity markets
16	14	H	Heating markets
17	14	D	Fictitious markets
18	14	G	Natural gas markets
19	2		Mid-term periods

**Table 4:** Sets partition groups

	id	set	subsets
1	1	4	Z
2	2	4	R
3	3	6	I(i)
4	4	6	O(i)
5	5	2	A
6	6	2	S
7	7	6	F(i)
8	8	14	B(k)
9	9	14	S(k)

**Table 5:** Decision variables

id	symbol	nature	tag	units	dataType	expr
1	si	Strategic	xInst	<i>devices</i>	int/dec	$si_i^p$
2	sd	Strategic	zDecom	<i>devices</i>	int/dec	$sd_i^{p,q}$
3	s	Strategic	sAvail	<i>kWh</i>	dec	$s_i^p$
4	z	Operational	zOut	<i>kWh</i>	dec	$z_{i,k}^{p,m,t}$
5	r	Operational	vSto	<i>kWh</i>	dec	$r_{k,j}^{p,m,t}$
6	x	Strategic	xAvail	<i>kWh</i>	dec	$x_j^p$
7	xi	Strategic	xInst	<i>kWh</i>	dec	$xi_j^p$
8	xd	Strategic	xDecom	<i>kWh</i>	dec	$xd_j^{p,q}$
9	y	Operational	yInput	<i>kWh</i>	dec	$y_{i,k}^{p,m,t}$
10	u	Operational	uPurchase	<i>kWh</i>	dec	$u_{k,n}^{p,m,t,mm}$
11	w	Operational	wSale	<i>kWh</i>	dec	$w_{k,n}^{p,m,t,mm}$
12	e	Operational	eCons	<i>kWh</i>	dec	$e^{p,m,t}$
13	qi	Operational	qStored	<i>kWh</i>	dec	$qi_{k,j}^{p,m,t}$
14	qo	Operational	qReleased	<i>kWh</i>	dec	$qo_{k,j}^{p,m,t}$

**Table 6:** Variables indices

	var	ind	loc	sub
1	1	4	sub	
2	1	1	sup	
3	2	4	sub	
4	2	1	sup	
5	2	8	sup	
6	3	4	sub	
7	3	1	sup	
8	4	4	sub	
9	4	6	sub	
10	4	1	sup	
11	4	2	sup	
12	4	3	sup	
13	5	6	sub	
14	5	10	sub	
15	5	1	sup	
16	5	2	sup	
17	5	3	sup	
18	6	1	sup	
19	6	10	sub	
20	7	1	sup	
21	7	10	sub	
22	8	10	sub	
23	8	1	sup	
24	8	8	sup	
25	9	4	sub	
26	9	6	sub	
27	9	1	sup	
28	9	2	sup	
29	9	3	sup	
30	10	14	sub	
31	10	6	sub	
32	10	1	sup	
33	10	2	sup	
34	10	3	sup	
35	10	15	sup	
36	11	14	sub	
37	11	6	sub	
38	11	1	sup	
39	11	2	sup	
40	11	3	sup	
41	11	15	sup	
42	12	1	sup	
43	12	2	sup	
44	12	3	sup	
45	13	1	sup	
46	13	2	sup	
47	13	3	sup	
48	13	6	sub	

49	13	10	sub
50	14	1	sup
51	14	2	sup
52	14	3	sup
53	14	6	sub
54	14	10	sub

**Table 7:** Parameters

id	symbol	nature	tag	units	group	expr	dataType
1	E	Deterministic	convCoeff	$kWh/kWh$	Technologies	$E_{i,k,kk}$	dec
2	D	Stochastic	enerDemand	$kWh$	Demand	$D_k^{p,m,t}$	dec
3	AG	Deterministic	ageFactorGen	$kW/kW$	Technologies	$AG_i^a$	dec
4	CO	Either	opCost	$EUR/kWh$	Policy	$CO_{i,k}^{p,m,t}$	dec
5	CI	Either	techCost	$EUR/kW$	Finance	$CI_i^{p,a}$	dec
6	CD	Either	decomCost	$EUR/kW$	Finance	$CD_i^a$	dec
7	DM	Deterministic	profDur	$days$	Model	$DM_m^p$	int
8	SU	Stochastic	subsInc	$EUR/kW$	Policy	$SU_i^p$	dec
9	LS	Deterministic	stoLoss	$kWh$	Technologies	$LS_i^{p,m,t}$	dec
10	SC	Deterministic	stoCap	$kWh$	Technologies	$SC_j^p$	dec
11	A	Stochastic	techAvail	$kW/kW$	Technologies	$A_i^{p,m,t}$	dec
12	IL	Deterministic	InvLim	$EUR$	Finance	$IL^p$	dec
13	H	Deterministic	polUnit	$kg/kWh$	Technologies	$H_{i,k,l}$	dec
14	PL	Either	polLimit	$kg$	Policy	$PL_l^p$	int
15	DT	Deterministic	timeDur	$hours$	Model	$DT$	dec
16	HO	Deterministic	operPeriods	$number$	Building	$HO$	int
17	RA	Deterministic	roofArea	$m^2$	Building	$RA$	dec
18	BA	Deterministic	buildArea	$m^2$	Building	$BA$	dec
19	BV	Deterministic	builVol	$m^3$	Building	$BV$	dec
20	G	Deterministic	NomCap	$kW/device$	Technologies	$G_i$	dec
21	SP	Stochastic	sellPrice	$EUR/kWh$	Finance	$SP_{i,k,n}^{p,m,t}$	dec
39	PP	Stochastic	PurchPrice	$EUR/kWh$	Policy	$PP_{i,k,n}^{p,m,t}$	dec
22	EF	Deterministic	effBuilding	$kWh/kWh$	Finance	$EF$	dec
23	OD	Deterministic	demRed	$kWh/kWh$	Technologies	$OD_{k,j}$	dec
24	C	Deterministic	polUnit	$kg/kWh$	Technologies	$C_{i,l,n}$	dec
25	AS	Deterministic	ageFactorSto	$kW/kW$	Technologies	$AS_i^a$	dec
26	GS	Deterministic	ageCapSto	$kW/kW$	Technologies	$GS_j$	dec
27	CIS	Either	techStoCost	$EUR/kW$	Finance	$CIS_j^{p,a}$	dec
28	COS	Either	techStoOperCost	$EUR/kWh$	Finance	$COS_{k,j}^{p,m,t}$	dec
29	CDS	Either	decomStoCost	$EUR/kWh$	Finance	$CDS_j^a$	dec
30	CP	Either	purchCost	$EUR/kWh$	Finance	$CP_{k,n}^{p,m,t}$	dec
31	B	Deterministic	primEner	$kWh/kWh$	Energy Efficiency	$B_{k,n}$	dec
32	FA	Deterministic	floorArea	$m^2$	Building	$FA$	dec
33	OS	Deterministic	storeAvail	$kWh/kWh$	Technologies	$OS_{k,j}$	dec
34	OI	Deterministic	storeAvail	$kWh/kWh$	Technologies	$OI_{k,j}$	dec
35	OO	Deterministic	storeAvail	$kWh/kWh$	Technologies	$OO_{k,j}$	dec
36	OR	Deterministic	availDischarging	$kWh/kWh$	Technologies	$OR_{k,j}$	dec
37	OA	Deterministic	lowerCharge	$kWh/kWh$	Technologies	$OA_{k,j}$	dec
38	OB	Deterministic	upperCharge	$kWh/kWh$	Technologies	$OB_{k,j}$	dec

**Table 8:** Parameter indices

	par	ind	loc
1	1	4	sub
2	1	6	sub
3	1	16	sub
4	2	6	sub
5	2	1	sup
6	2	2	sup
7	2	3	sup
8	3	4	sub
9	3	9	sup
10	4	4	sub
11	4	6	sub
12	4	1	sup
13	4	2	sup
14	4	3	sup
15	5	4	sub
16	5	1	sup
17	5	9	sup
18	6	4	sub
19	6	9	sup
20	7	2	sub
21	7	1	sup
22	8	4	sub
23	8	1	sup
24	9	4	sub
25	9	1	sup
26	9	2	sup
27	9	3	sup
28	10	4	sub
29	10	1	sup
30	11	4	sub
31	11	1	sup
32	11	2	sup
33	11	3	sup
34	12	1	sup
35	13	4	sub
36	13	6	sub
37	13	7	sub
38	14	7	sub
39	14	1	sup
40	20	4	sub
41	21	4	sub
42	21	6	sub
43	21	14	sub
44	21	1	sup
45	21	2	sup
46	21	3	sup
47	39	4	sub
48	39	6	sub

49	39	14	sub
50	39	1	sup
51	39	2	sup
52	39	3	sup
53	23	10	sub
54	23	6	sub
55	24	4	sub
56	24	7	sub
57	24	14	sub
58	25	4	sub
59	25	9	sup
60	26	10	sub
61	27	10	sub
62	27	1	sup
63	27	9	sup
64	28	10	sub
65	28	6	sub
66	28	1	sup
67	28	2	sup
68	28	3	sup
69	29	10	sub
70	29	9	sup
71	30	14	sub
72	30	6	sub
73	30	1	sup
74	30	2	sup
75	30	3	sup
76	31	6	sub
77	31	14	sub
78	33	6	sub
79	33	10	sub
80	34	6	sub
81	34	10	sub
82	35	6	sub
83	35	10	sub
84	36	6	sub
85	36	10	sub
86	37	6	sub
87	37	10	sub
88	38	6	sub
89	38	10	sub

**Table 9:** Constraints

	id	title	direction	nature
1	1	Available generating technologies calculation	=	Strategic
2	9	Available storing technologies calculation	=	Strategic
3	2	Meet Loads	$\geq$	Operational
4	3	Technologies short-term availability	$\leq$	Operational
5	4	Generation technologies decommissioning limit	$\leq$	Strategic
6	10	Storage technologies decommissioning limit	$\leq$	Strategic
7	5	Budget limit	$\leq$	Strategic
8	6	Emissions limit	$\leq$	Strategic
9	7	Physical limit for energy-creation technologies installation	$\leq$	Strategic
10	8	Physical limit for energy-absorbing technologies installation	$\leq$	Strategic
11	11	Efficiency constraint	$\geq$	Strategic
12	12	Primary energy calculation	=	Strategic
13	13	Energy output calculation	=	Operational
14	14	Energy stored calculation	=	Operational
15	15	Energy discharging limit	$\leq$	Operational
16	16	Energy storage lower limit	$\geq$	Operational
17	17	Energy storage upper limit	$\leq$	Operational
18	3001	Discrete sized technologies devices to install	$\in$	Integer
19	3002	Discrete sized technologies devices to decommission	$\in$	Integer
20	1001	Non-negative energy supply	$\geq$	nn
21	1002	Positive generating devices to install	$\geq$	nn
22	1003	Positive devices to decommission	$\geq$	nn
23	1004	Positive available capacity	$\geq$	nn
24	1005	Positive energy to store	$\geq$	nn
25	1006	Positive storage devices to install	$\geq$	nn
26	1007	Positive storage devices to decommission	$\geq$	nn
27	1008	Positive available storage capacity	$\geq$	nn
28	1009	Positive input energy	$\geq$	nn
29	1010	Positive purchased energy	$\geq$	nn
30	1011	Positive sold energy	$\geq$	nn
31	1012	Positive consumed energy	$\geq$	nn

**Table 10:** Objectives

	id	title	direction	nature
32	2001	Minimize total cost		objective



## B. Code Examples

In this appendix we present very preliminary application of the Symbolic Model Specification using the R programming language and statistical software [10]. The SMS of the EnRiMa DSS will be implemented within the DSS Architecture using the tools and technologies chosen in WP5. Nevertheless, the implementation presented here can be translated easily to any other system. The choice of R for this “demo” implementation is due to: (1) It is open source; (2) It allows “iterate programming” ([9]); (3) SMS, examples, equations and solvers can be integrated in a single framework.

This appendix contains examples of the code developed as a prototype to dynamically generate the EnRiMa models. The objects of the SMS are stored in data objects, which main fields are “id” and “symbol”. (see Appendix A to see the tables). The tables are populated by calls to functions that add information to the tables. These tables are related through the “id” fields. Once the data objects are created, functions devoted to mathematically represent them print the equations within the document along with additional information as the nature of the parameters, categorization, or descriptions. The objects can eventually be used to create the “model objects” ready to be delivered to the solver in the appropriate format.

### SMS Infrastructure Examples

The first step to build the SMS is the creation of the data objects that will contain the SMS objects and their symbols. The following code is used to create the sets `data.frame` object:

```
> smsSets <- newDataset(fields = c("id",
  "symbol",
  "isSet",
  "desc",
  "bounds",
  "ref",
  "dif",
  "short",
  "corr",
  "rep",
  "firstNull",
  "greater"))
```

Once the data object is created we can add the sets. For example, the following code add the set which symbol is *p*.

```
> smsSets <- addObject (smsSets, list(id = 1,
  symbol = "p",
  isSet = TRUE,
  desc = "Long-term period for strategic decisions (e.g. year)",
  bounds = "",
  ref = "",
  dif = "",
  short = "long-term period",
  corr = NA,
  rep = NA,
  fromNull = FALSE,
  greater = NA))
```

Note that in subsequent calls to this set, the “id” is always used. Hence, we can just change the value of the symbol in this chunk of code, and all the SMS is automatically updated (variables, parameters, constraints, etc.) including descriptions. Besides sets, we create subsets to coerce the scope of the sets throughout the constraints.

Similarly to sets, decision variables and parameters are created. We add indices to variables and parameters with specific functions. The following code add the indices to the *si* variable (*id* = 1):

```
> smsIndVar <- addObject(smsIndVar ,
  list(var = 1,
        ind = 4,
        loc = "sub",
        sub = NA))
> smsIndVar <- addObject(smsIndVar ,
  list(var = 1,
        ind = 1,
        loc = "sup",
        sub = NA))
```

Once the sets, variables and parameters are created, the expressions of variables and parameters are obtained. For example, for the variables:

```
> smsVars[, "expr"] <- getVarExpr()
> smsVars[, "bounds"] <- getVarBounds()
```

What this functions do is creating the expressions in  $\text{\LaTeX}$ format through the symbols stored. For example the function `getVarExpr` build the expressions of the variables:

```
function() {
  expr <- numeric()
  for (i in 1:(nrow(smsVars))){
    expr[i] <- paste(smsVars[i, "symbol"],
                    "_{",
                    paste(getInd("sub", smsVars[i, "id"]),
                          collapse = ","),
                    "}^{",
                    paste(getInd("sup", smsVars[i, "id"]),
                          collapse = ","),
                    }",
                    sep="")
  }
  return(expr)
}
```

The constraints can be created afterwards. They are composed by terms, at the right or left side. The terms can be affected by other mathematical symbols before and/or after them, e.g. parenthesis, operators, sums, etc. A comparison operator is also needed between both sides. Complementary, we add to the constraint information about the replications of the constraint throughout the sets in the variables and parameter indices, the total number of constraints based in these replications, and the measurement units of each term to allow consistency check. The following code creates eq. (17):

```
> #Add constraint
> smsConsts <- addObject(smsConsts ,
  list(id = 17,
        title = "Energy storage upper limit",
        desc = "The amount of energy that may be stored from any
                energy-storage technology must be lower than the
                capacity installed corrected by the maximum charge
                allowed",
        direction = "\\leq",
        nature = "Operational"))
> #Set replications
> smsConstRep <- addObject(smsConstRep ,
  list(const = 17,
```

```

ind = c(1,2,3, 4,10),
rel = " \\in ",
subset = c(rep("", 4),
            smsSubsets[smsSubsets[, "id"] == 12 , "subset"]),
set = toupper(smsSets[smsSets[, "id"] %in% c(1,2,3, 4, 10),
                  "symbol"]))
> #Add terms
> smsCTerms <- addObject(smsCTerms,
                        addTerm("sms", 17, "V", 5, "l"))
> smsCTerms <- addObject(smsCTerms,
                        addTerm("sms", 17, "P", 38, "r", dot="r"))
> smsCTerms <- addObject(smsCTerms,
                        addTerm("sms", 17, "V", 6, "r"))

```

The function `addTerm` is a simplified way to add “simple” terms, without subsets or complex expressions:

```

function(prefix = "sms",
        toConst,
        type = "V",
        id,
        inSide,
        withSign = "",
        sums = NULL,
        bracket = NULL,
        dot = ""){
  if (!(type %in% c("V", "P"))) stop("Invalid type")
  obj <- get(paste(prefix, type, "ars", sep=""))
  newTerm <- obj[obj[, "id"] == id,]
  if (length(sums) > 0){
    setSums <- get(paste(prefix, "Sets", sep=""))
    setSums <- setSums[setSums[, "id"] %in% sums, "symbol"]
    allSums <- paste("\\sum_{",
                    setSums,
                    "\\in \\mathcal{",
                    toupper(setSums), "}",
                    collapse = " ")
  }
  else allSums = ""
  return(list(const = toConst,
            term = newTerm[, "expr"],
            units = newTerm[, "units"],
            side = inSide,
            sign = withSign,
            before = paste(allSums,
                          ifelse(bracket == "l" , " \\left ( ", ""),
                          ifelse(dot == "l", " \\cdot ", ""),
                          sep = ""),
            after = paste(ifelse(bracket == "r" , " \\right ) ", ""),
                          ifelse(dot == "r", " \\cdot ", ""),
                          sep = ""
            )))
}

```

Note that we create objective functions in a similar way by changing the “nature” of the constraint. This allows customising the SMS to different decision makers.

Finally, in order to show the mathematical representation in a document, the SMS is printed through several functions devoted to this task. They take the data in the tables and formulate

the appropriate expressions. Their use is very simple, for example the following code prints all the variables:

```
> printVars()
```

The code of the `printVars` function is the following:

```
function(nature = NA){
  smsVars <- smsVars[order(smsVars[, "symbol"]),]
  if (!is.na(nature)){
    smsVars <- smsVars[smsVars[,"nature"]==nature,]
  }
  cat("\\begin{description}")
  for (i in 1:(nrow(smsVars))){
    cat(paste("\n\\item [",
              smsVars[i, "expr"],
              "$]",
              smsVars[i, "desc"],
              smsVars[i, "bounds"],
              " (\\textit{",
              smsVars[i,"units"],
              "} \\texttt{[",
              smsVars[i,"dataType"],
              "]}) .",
              sep=""))
  }
  cat("\n\\end{description}")
}
```

The function that prints constraints and objective functions (`printConst`) has a similar behaviour, but more elaborated:

```
function (nature){
  this.consts <- smsConsts[smsConsts[, "nature"] == nature,]
  cat("\\begin{description}\n")
  for (i in seq(along = this.consts[, "id"])) {
    this.const.id <- this.consts[i, "id"]
    this.reps <- smsConstRep[smsConstRep[, "const"] == this.const.id,]
    reps <- character()
    for (j in seq(along = this.reps[, "ind"])) {
      reps <- paste("\\;", reps,
                    paste(smsSets[smsSets[, "id"] == this.reps[j, "ind"],
                              "symbol"],
                          this.reps[j, "rel"],
                          ifelse(this.reps[j, "set"] == toupper(this.reps[j,
                              "set"]),
                                "\\mathcal{", "{"),
                                this.reps[j, "set"],
                                "}_{"),
                          this.reps[j, "subset"],
                          "},",
                          sep=""),
                    sep = "\\;")
    }

    whatPrinting <- ifelse(nature == "objective", "1 \\; objective.",
                           ",")
  }
}
```

```

"constraints.")
tcons <- paste(paste(paste(this.reps[, "set"],
                        ifelse(this.reps[, "subset"] != "", "[",""),
                        this.reps[, "subset"],
                        ifelse(this.reps[, "subset"] != "", "]",""))),
              collapse = "\\times "),
              "\\,\\mathrm{",
              whatPrinting,
              "}")
cat("\\\\item[",
    smsConsts[smsConsts[, "id"] == this.const.id, 2],
    "]\\\\hfill\\\\\\\\\\n",
    smsConsts[smsConsts[, "id"] == this.const.id, 3],
    ".\\n\\\\begin{align}\\n",
    "\\\\label{eq:const:",
    this.const.id,
    "}\\n",
    paste(t(as.matrix(smsCTerms[smsCTerms[, "side"] == "l" &
                        smsCTerms[, "const"] == this.const.id,
                        c("sign", "before", "term", "after")])),
          sep = ""),
    smsConsts[smsConsts[, "id"] == this.const.id, 4],
    paste(t(as.matrix(smsCTerms[smsCTerms[, "side"] == "r" &
                        smsCTerms[, "const"] == this.const.id,
                        c("sign", "before", "term", "after")])),
          collapse = " "),
    " & ",
    reps,
    "\\\\ \\nonumber \\\\ \\n \\nonumber",
    ifelse(nature == "objective", " & ", ""),
    "\\scriptstyle",
    paste("\\\\mathit{",
          smsCTerms[smsCTerms[, "side"] == "l" &
                    smsCTerms[, "const"] == this.const.id,
                    "units"],
          "}",
          collapse = ",\\, "),
    ifelse(nature == "Integer", "", "\\; \\sim \\; "),
    paste(smsCTerms[smsCTerms[, "side"] == "r" &
                    smsCTerms[, "const"] == this.const.id,
          "units"],
          collapse = ",\\, "),
    " & \\quad ",
    #ifelse(nature == "objective", " & ", ""),
    "\\scriptstyle ",
    tcons,
    "\\n\\\\end{align}\\n",
    sep = "")
}
cat("\\\\end{description}\\n")
}

```

## SMS Realisation Example

In this example the SMS described in this document is parameterised with specific characteristics of a simulated building with a hypothetical simple configuration. Simulated data are assigned to the parameters. With simulated data we mean data taken from several sources that could be real

in a given building individually, but not necessarily as a whole. Moreover, these data might be approximate. Note that this example is focused in the strategic model. Examples of use for the operational model can be consulted in deliverable D2.2 [6].

This very simple example simulates a public building with only energy demand for electricity and heating. The energy inputs can be purchases from markets (electricity and natural gas) and energy generation from a CHP unit. Heat will be met using the CHP unit and the remaining electricity is sold to the grid.

The CHP unit used for this example is the model DACHS G 5.5 whose technical parameters can be found in [http://www.baxi-senertec.co.uk/html/baxi\\_senertec\\_dachs.htm](http://www.baxi-senertec.co.uk/html/baxi_senertec_dachs.htm). The rest of parameters are taken from the information of the test sites in [3, 5] and other public sources. Following, the tables with the sets and parameter are shown.

### Example Data

A new object which contains all the implementation of the SMS is created:

```
> smsExample <- newImplem()
```

This object (`smsExample`) contains the elements: sets, parameters, variables, and constraints. The sets are added through the function `fillImplemSet`. For example, for the set  $\mathcal{P}$ :

```
> smsExample <- fillImplemSet(smsExample, 1, c("2013", "2014", "2015"))
```

All the sets for this simple example are in Tables 11 to 16.

**Table 11:** Simulated example set  $\mathcal{P}$

p	long-term period
1	2013
2	2014
3	2015

**Table 12:** Simulated example set  $\mathcal{M}$

m	mid-term period
1	hot
2	cold

**Table 13:** Simulated example set  $\mathcal{T}$

t	short-term period
1	day
2	night

**Table 14:** Simulated example set  $\mathcal{I}$

i	energy-creating technology
1	Dachs G 5.5

**Table 15:** Simulated example set  $\mathcal{K}$

k	energy type
1	Electricity
2	Heating
3	NG

**Table 16:** Simulated example set  $\mathcal{N}$

n	energy market
1	Grid
2	NG vendor

The functions that add elements to the implementation go over the SMS data objects containing the SMS objects and then save them in the implementation object. For example, the function `addImplemPars` creates all the parameters of the implementation with value 0:

```
function(implem) {
  for (i in seq(along = smsPars[, "id"])) {
    thisPar <- smsPars[i, "id"]
    thisIndPar <- list()
    for (j in seq(along = order(smsIndPar[smsIndPar[, "par"] ==
      thisPar, "ind"]))) {
      thisSet <- smsIndPar[smsIndPar[, "par"] == thisPar, "ind"][
        j]
      if (smsSets[smsSets[, "id"] == thisSet, "isSet"]){
        if (!is.null(implem[["sets"]][[smsSets[smsSets[, "id"]
          == thisSet,
            "symbol"]]])){
          thisIndPar[[smsSets[smsSets[, "id"] == thisSet, "
            symbol"]]] <-
            seq(1, nrow(implem[["sets"]][[smsSets[smsSets[,
              "id"] == thisSet, "symbol"]]]))
        }
      }
      else{
        shift <- as.numeric(smsSets[smsSets[, "id"] == thisSet,
          "fromNull"])
        thisIndPar[[smsSets[smsSets[, "id"] == thisSet, "symbol
          "]]] <-
          seq(1 - shift , findRef(implem, thisSet) - shift)
      }
    }
    if (length(thisIndPar) == 0){
      implem[["parameters"]][[smsPars[smsPars[, "id"] == thisPar,
        "symbol"]]] <-
        data.frame(value = 0)
    }
    else{
      implem[["parameters"]][[smsPars[smsPars[, "id"] == thisPar,
        "symbol"]]] <-
        data.frame(expand.grid(thisIndPar), value = 0)
    }
  }
  return(implem)
}
```

Then, specific parameter values are assigned to the suitable indices of the parameters. For example, with the following code, the price of electricity is assigned to the parameter  $CP_{k,n}^{p,m,t}$  for the indices  $n = 1$  and  $k = 1$ :

```
> elecPrice <- 0.15
> smsExample <- setImplemPars(smsExample, par = 30,
  whichInd = getExprSetPars(ind = c(14, 6), values = c(1, 1)),
  values = elecPrice)
```

For the sake of space, we print only, in text mode instead of in tables, the first and last chunks of the complete parameter values tables. All the values are printed then sequentially. Note that each value corresponds to a combination of the parameter indices. For example, for the parameters ( $E_{i,k,kk}$ ,  $D_k^{p,m,t}$ , and  $CP_{k,n}^{p,m,t}$ ) we have the following values:



Parameter: E

```
-----  
  i k kk  value  
1 1 1  1 0.0000  
2 1 2  1 0.0000  
3 1 3  1 3.7037  
4 1 1  2 0.0000  
5 1 2  2 0.0000  
6 1 3  2 1.6400  
7 1 1  3 0.0000  
8 1 2  3 0.0000  
9 1 3  3 0.0000
```

Parameter: D

```
-----  
      k  p  m  t  value  
1      1  1  1  1  59.4  
2      2  1  1  1    0  
3      3  1  1  1    0  
4      1  2  1  1  59.4  
... ..  
33     3  2  2  2    0  
34     1  3  2  2    0  
35     2  3  2  2  200  
36     3  3  2  2    0
```

Values: 59.4 0 0 59.4 0 0 59.4 0 0 59.4 200 0 59.4 200 0 59.4 200 0 0  
0 0 0 0 0 0 0 0 0 200 0 0 200 0 0 200 0

Parameter: CP

```
-----  
      n  k  p  m  t  value  
1      1  1  1  1  1  0.15  
2      2  1  1  1  1    0  
3      1  2  1  1  1    0  
4      2  2  1  1  1    0  
... ..  
69     1  2  3  2  2    0  
70     2  2  3  2  2    0  
71     1  3  3  2  2    0  
72     2  3  3  2  2  0.05
```

Values: 0.15 0 0 0 0 0.05 0.15 0 0 0 0 0.05 0.15 0 0 0 0 0.05 0.15 0 0  
0 0 0.05 0.15 0 0 0 0 0.05 0.15 0 0 0 0 0.05 0.15 0 0 0 0 0.05 0.15  
0 0 0 0 0.05 0.15 0 0 0 0 0.05 0.15 0 0 0 0 0.05 0.15 0 0 0 0 0.05  
0.15 0 0 0 0 0.05

## Decision Variables

The number of variables in the model arising from the sets we are implementing is in Table 17.

We have also built the names of all the implemented variables through the symbol and indices. Table 18 shows the first and last variable names.

**Table 17:** Simulated example number of implemented variables

	symbol	size
1	si	3
2	sd	3
3	s	3
4	z	36
5	y	36
6	u	144
7	w	144
8	Total	369

**Table 18:** Simulated example implemented variables names

	Name
1	si.i1p1
2	si.i1p2
3	si.i1p3
4	sd.i1p1q2
...	
366	w.n1k2p3m2t2mm2
367	w.n2k2p3m2t2mm2
368	w.n1k3p3m2t2mm2
369	w.n2k3p3m2t2mm2

## Constraints and Objectives

Finally, we save the coefficients of each variable for all the constraints and the objective function. For example, for equation (13) and indices  $p = 1$ ,  $m = 1$ ,  $t = 1$ ,  $i = 1$ , and NA assign coefficients to the technology matrix:

```
> smsExample[["const"]] <- data.frame(Variable = smsExample[["variables"]][["varNames"]],
  stringsAsFactors = FALSE)
> dataConst <- smsExample[["const"]]
> dataConst[, "c13"] = 0
> dataConst[dataConst[, "Variable"] == "z.i1k1p1m1t1", "c13"] <- 1
> dataPar <- smsExample[["parameters"]][["E"]]
> dataConst[dataConst[, "Variable"] == "y.i1k1p1m1t1",
  "c13"] <- (subset(dataPar, i == 1 & k == 3 & kk == 1, value))
  ^(-1)
> smsExample[["const"]] <- dataConst
```

Thus, we have the specific constraint:

$$z_{1,1}^{1,1,1} - (E_{1,3,1})^{-1} \cdot y_{1,3}^{1,1,1} = 0, \quad (74)$$

with all the coefficients equal to zero but the following:

	Variable	c13
10	z.i1k1p1m1t1	1.0000000
46	y.i1k1p1m1t1	0.2700003

Doing the same for all the constraints and objective function, we have in the object `smsExample` all the data needed by the solver. More complex implementations and models will be tackled in subsequent stages of the project, including the stochastic models described in Sec. 5 by adding a new set for the scenarios.

## Session Info

This is the session information provided by the R system regarding the environment where the code used in this document has been executed.

```
> sessionInfo()
```

```
R version 2.14.2 (2012-02-29)
Platform: i686-pc-linux-gnu (32-bit)

locale:
 [1] LC_CTYPE=en_US.UTF-8
 [2] LC_NUMERIC=C
 [3] LC_TIME=es_ES.UTF-8
 [4] LC_COLLATE=en_US.UTF-8
 [5] LC_MONETARY=es_ES.UTF-8
 [6] LC_MESSAGES=en_US.UTF-8
 [7] LC_PAPER=es_ES.UTF-8
 [8] LC_NAME=es_ES.UTF-8
 [9] LC_ADDRESS=es_ES.UTF-8
[10] LC_TELEPHONE=es_ES.UTF-8
[11] LC_MEASUREMENT=es_ES.UTF-8
[12] LC_IDENTIFICATION=es_ES.UTF-8

attached base packages:
[1] splines      grid          stats          graphics      grDevices
[6] utils        datasets     methods        base

other attached packages:
[1] psych_1.2.1      Hmisc_3.9-2      survival_2.36-12
[4] lpSolve_5.6.6    xtable_1.7-0      rj_1.1.0-1

loaded via a namespace (and not attached):
[1] cluster_1.14.2 lattice_0.20-0 rj.gd_1.1.0-1
[4] tools_2.14.2
```