

# Completing Z Specifications

November 10, 2023

*TempMonitor*

$deployed : \mathbb{P} \text{ SENSOR\_TYPE}$   
 $map : \text{SENSOR\_TYPE} \rightarrow \text{LOCATION\_TYPE}$   
 $read : \text{SENSOR\_TYPE} \rightarrow \text{TEMPERATURE\_TYPE}$

$deployed = \text{dom } map$   
 $deployed = \text{dom } read$

*ReplaceSensorOK*

$\Delta \text{TempMonitor}$   
 $location? : \text{LOCATION\_TYPE}$   
 $new\_sensor? : \text{SENSOR\_TYPE}$   
 $temperature? : \text{TEMPERATURE\_TYPE}$

$location? \in \text{ran } map$   
 $new\_sensor? \notin deployed$   
 $map' = \{map^{-1}(location?)\} \triangleleft map$   
 $read' = \{map^{-1}(location?)\} \triangleleft read$   
 $deployed' = deployed \setminus \{map^{-1}(location?)\}$   
 $map' = map \oplus \{new\_sensor? \mapsto location?\}$   
 $read' = read \oplus \{new\_sensor? \mapsto temperature?\}$

*ReturnCollectionOK*

$\exists \text{TempMonitor}$   
 $locations! : \mathbb{P} \text{ LOCATION\_TYPE}$   
 $temperatures! : \mathbb{P} \text{ TEMPERATURE\_TYPE}$

$locations! = \text{ran } map$   
 $temperatures! = \text{ran } read$

$ReplaceSensor \hat{=} (ReplaceSensorOK \wedge Success) \oplus (LocationUnkown \vee SensorAlreadyDeployed)$

*ReturnAllTemperaturesAndLocations*  $\hat{=}$  *ReturnCollectionOK*

Based on the given new requirements, we have extended the Z specification using LaTeX on the TeXstudio platform. We created 2 schemas *ReplaceSensorOK* and *ReturnCollectionOK* (see next page, along with the previously defined *TempMonitor* schema), which are used to define the two operations *ReplaceSensor* and *ReturnAllTemperaturesAndLocations*. Both of these new operations use schemas already given in the first iteration.

Explanation of the Z specifications:

### 1. ReplaceSensorOK

Our operation will take as inputs a new sensor, a location, and a temperature.

We first check that the location is covered (in the range of the map function), and that the new sensor is not deployed. If both of these preconditions pass, then we remove the old Sensor-Location and Sensor-Temperature pairs from Map and Read respectively. We then remove the old sensor (determined using the inverse Map function and location) from the SensorRegistry, and then we add new Sensor-Location and Sensor-Temperature pairs to the Map and Read functions, which would be the same pairs as the ones removed previously, except the old sensor is replaced with the new sensor.

### 2. ReturnCollectionOK

The operation returns a list of locations (range of the map function) and their temperatures (range of the read function).

Based on the newly specified requirements, the following is the refined domain model for iteration 2.

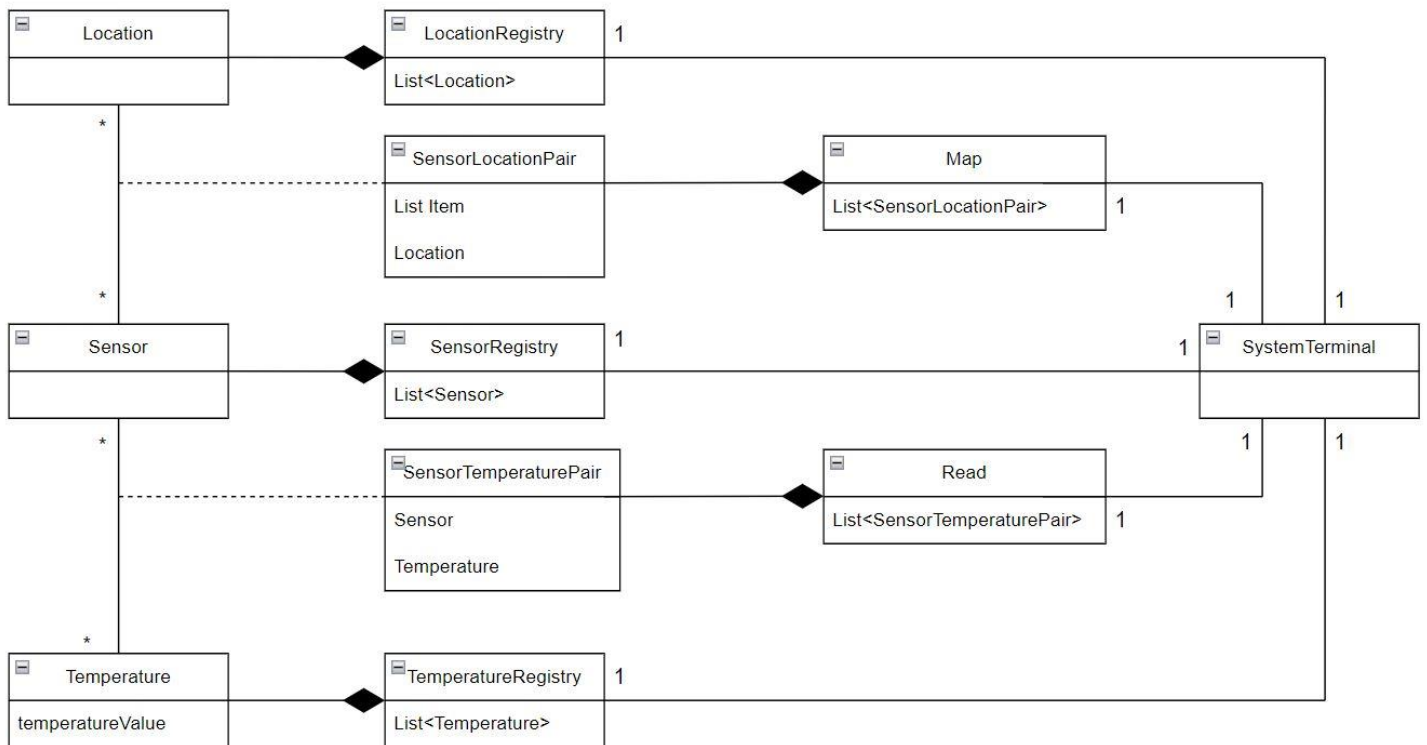


Figure 1: Domain Model - Updated for Iteration 2