An Event-B Specification of

CoffeeClub

This Event-B system is based on a model that appeared in the book: System Modelling & Design Using Event-B by Ken Robinson.

It illustrates how a an abstract machine describes the consistency of a single bank account. The refinement then add multiple membership accounts while still maintaining the single bank account.

It also illustrates the use of a witness to drop (or replace) a variable when refining an event.

1 CONTEXT CoffeeClubCtx **2**
- 1.1 MEMBER . . . 2

2 MACHINE CoffeeClubMch **3**
- 2.1 *piggybank* . . . 3
- 2.2 FeedBank(*amount_feed*) . . . 3
- 2.3 RobBank(*amount_rob*) . . . 3

3 REFINEMENT CoffeeClubRef **4**
- 3.1 *accounts coffeeprice members* . . . 4
- 3.2 SetPrice(*new_price*) . . . 4
- 3.3 NewMember(*new_member*) . . . 4
- 3.4 Contribute(*contribution member*) `refines` FeedBank . . . 4
- 3.5 BuyCoffee(*member_buy*) `refines` RobBank . . . 5

CONTEXT CoffeeClubCtx [1]

SETS [1.1]

MEMBER

AXIOMS

ax0: finite(MEMBER)

END

MACHINE CoffeeClubMch 3a 2

VARIABLES 2.1

piggybank The coffe club has a single bank account storing all its money.

INVARIANTS

inv1: $piggybank \in \mathbb{N}$ The bank account can be positive or zero, but not negative.

EVENT INITIALISATION

THEN

init_0: $piggybank := 0$

END

EVENT FeedBank 2.2

When money is put into the bank account we feed it.

ANY

amount_feed

WHERE

grd_1: $amount\_feed \in 1..100$

THEN

act_1: $piggybank := piggybank + amount\_feed$

END

EVENT RobBank 2.3

Likewise, when taking money, we rob it.

ANY

amount_rob

WHERE

grd_1: $amount\_rob \in 1..50$ The cost of a coffe is 1 up to 50.

grd_2: $amount\_rob \leq piggybank$

THEN

act_1: $piggybank := piggybank - amount\_rob$

END

REFINEMENT CoffeeClubRef 14 a 3

We now introduce the concept of member accounts, the sum of the member accounts should be the total piggy bank account.

REFINES CoffeeClubMch
SEES CoffeeClubCtx

VARIABLES 3.1

members
accounts
coffeeprice

INVARIANTS

inv1_1: $members \subseteq \text{MEMBER}$
inv1_2: $accounts \in members \to \mathbb{N}$
inv1_3: $coffeeprice \in 1..30$

EVENT INITIALISATION
EXTENDS INITIALISATION
THEN

init1_1: $members := \varnothing$
init1_2: $accounts := \varnothing$
init1_3: $coffeeprice := 1$

END

EVENT SetPrice 3.2
ANY

new_price

WHERE

grd0: $new\_price \in 1..30$

THEN

act0: $coffeeprice := new\_price$

END

EVENT NewMember 3.3
ANY

new_member

WHERE

grd0: $new\_member \in \text{MEMBER}$
grd1: $new\_member \notin members$

THEN

act0: $accounts(new\_member) := 0$
act1: $members := members \cup \{new\_member\}$

END

EVENT Contribute 3.4
REFINES FeedBank
ANY

contribution
member

WHERE

grd0: $contribution \in 1..70$
grd1: $member \in members$
grd2: $member \in \mathrm{dom}(accounts)$

WITH

amount_feed: $amount\_feed = contribution$

THEN

act0: $accounts(member) := accounts(member) + contribution$
act1: $piggybank := piggybank + contribution$

END

EVENT BuyCoffee 3.5
REFINES RobBank
ANY

member_buy

WHERE

grd1_1: $member\_buy \in \mathrm{dom}(accounts)$
grd1_2: $accounts(member\_buy) \geq coffeeprice$
grd1_3: $coffeeprice \leq piggybank$

WITH

amount_rob: $amount\_rob = coffeeprice$
The amount is replaced with the coffee price. Note that proof for amount_rob $\in 1..50$ is easily proven since the coffeeprice is defined as coffeeprice $\in 1..30$

THEN

act1_1: $accounts(member\_buy) := accounts(member\_buy) - coffeeprice$
act1_2: $piggybank := piggybank - coffeeprice$

END

accounts, 4

BuyCoffee, 5

CoffeeClubCtx, 2, 4
CoffeeClubMch, 3, 4
CoffeeClubRef, 4
coffeeprice, 4
Contribute, 4

FeedBank, 3, 4

INITIALISATION, 3, 4

members, 4

NewMember, 4

piggybank, 3

RobBank, 3, 5

SetPrice, 4