CLEARSY
Safety Solutions Designer

Hackathon
JUL2025

AIX
LYON
PARIS
STRASBOURG

WWW.CLEARSY.COM

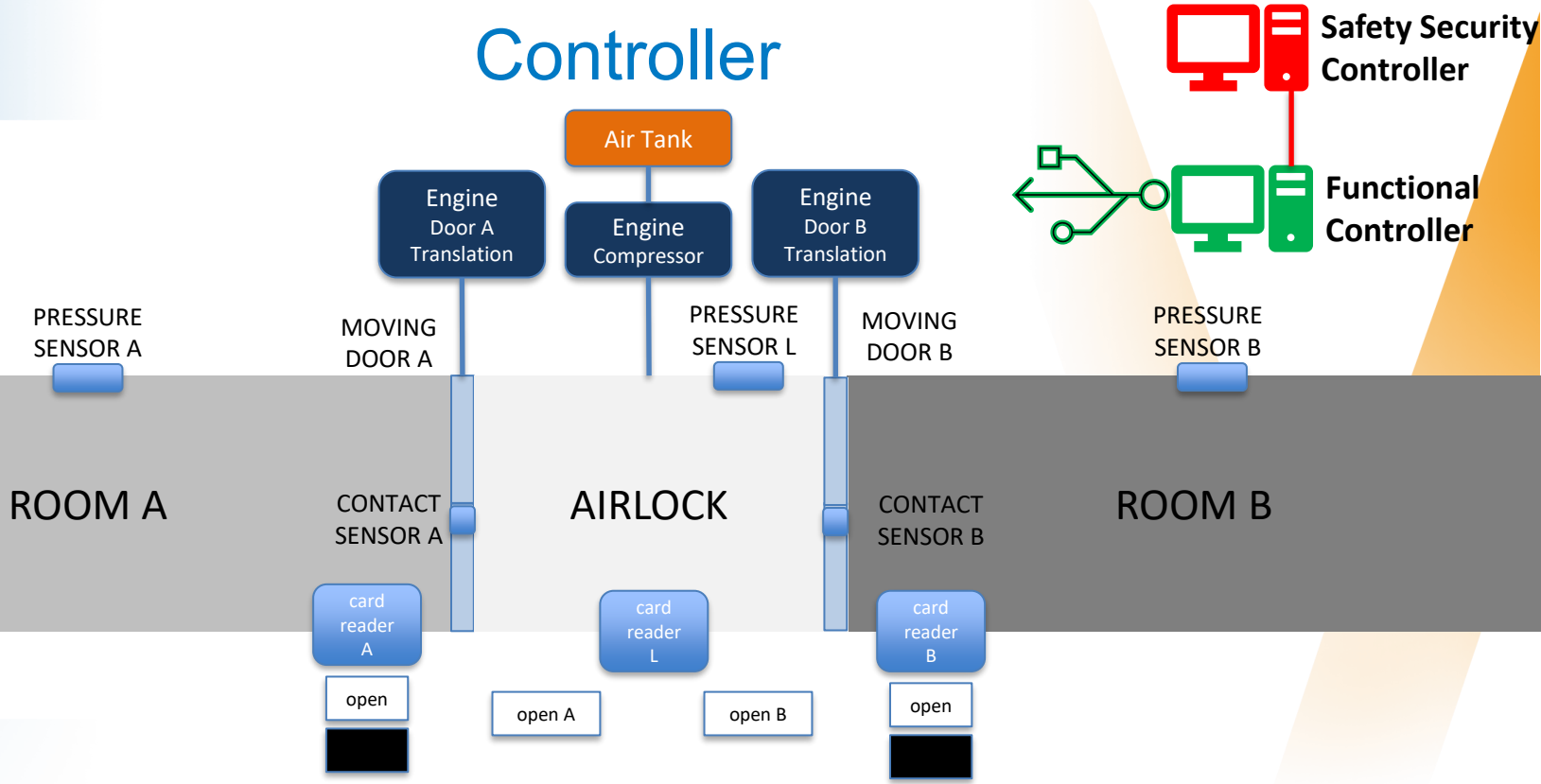
Airlock
Implementing the Functional Controller

PART IV

Thierry Lecomte
R&D Director

THIERRY.LECOMTE@CLEARSY.COM

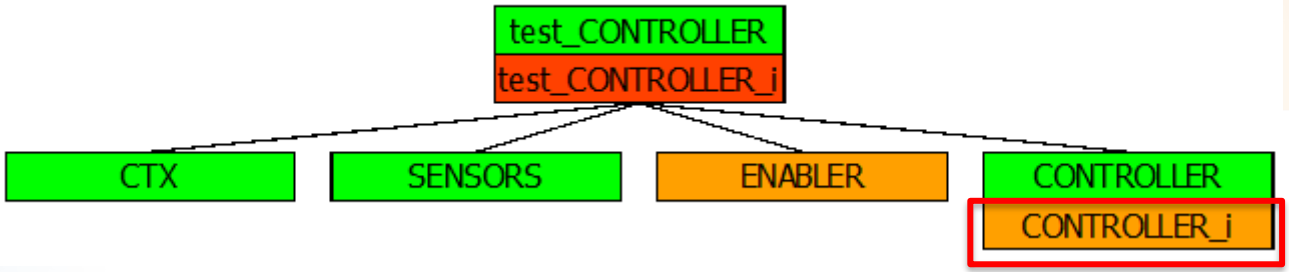
Controller



design decision: we remove the **close** buttons to save money

Your turn

- *Implement CONTROLLER [8 pt]*
  - *VARIABLES are already concrete. Only INITIALISATION is required.*
  - *OPERATION process_headers implementation is given*
  - *Implement the algorithm of the OPERATION control*



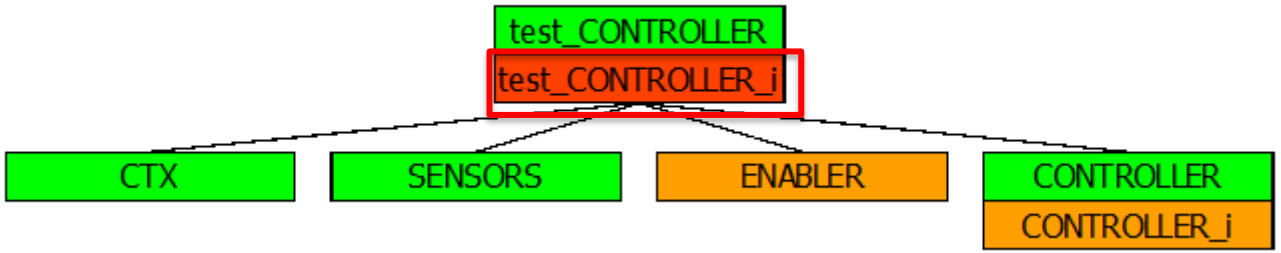
```
IMPLEMENTATION CONTROLLER_i
REFINES CONTROLLER
SEES
    CTX, SENSORS, ENABLER
INITIALISATION
    current_action := NONE;
    current_authentication := AUTHENTICATED_NONE;
    current_objective := OBJ_NONE

OPERATIONS
    process_readers =
    IF current_authentication = AUTHENTICATED_NONE THEN
        IF card_reader_a = TRUE THEN
            current_authentication := AUTHENTICATED_A
        ELSIF card_reader_b = TRUE THEN
            current_authentication := AUTHENTICATED_B
        ELSIF card_reader_l = TRUE THEN
            current_authentication := AUTHENTICATED_L
        END
    END;

    control = /* TO BE COMPLETED */
    BEGIN
    END
END
```

Your turn

- *Complement test_CONTROLLER [2 pt]*
  - *Specify an invariant that links some variables of CONTROLLER and the variables enable_door_a and enable_door_b*
  - *Have a look at the proof obligation and try to **briefly** check if the predicate is valid. Explain shortly your thought*



```
IMPLEMENTATION test_CONTROLLER_i
REFINES test_CONTROLLER

IMPORTS CTX, SENSORS, ENABLER, CONTROLLER

INVARIANT
OPERATIONS
    test_control =
    BEGIN
        update_sensors_states;
        process_readers;
        control;
        compute_enabling
    END
END
```