

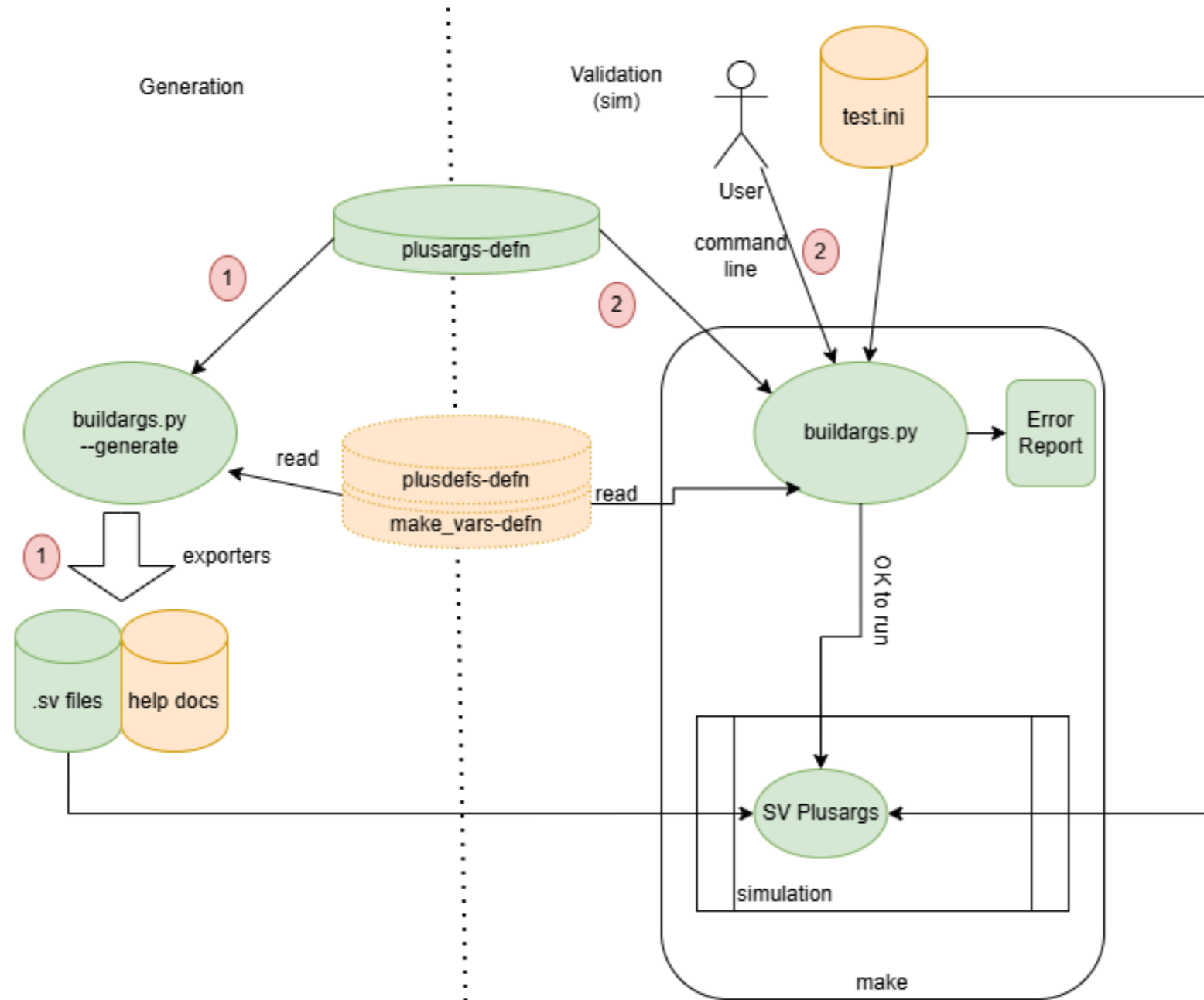
The Problems

- In some verification environments, the number of plusargs explodes to add debug functionality, or to include transient behavior to avoid a bug.
- Many plusargs are not well documented, or are difficult to understand their intent.
- Plusarg handling limited to basic types, leaving conversion to enums and more on the user.

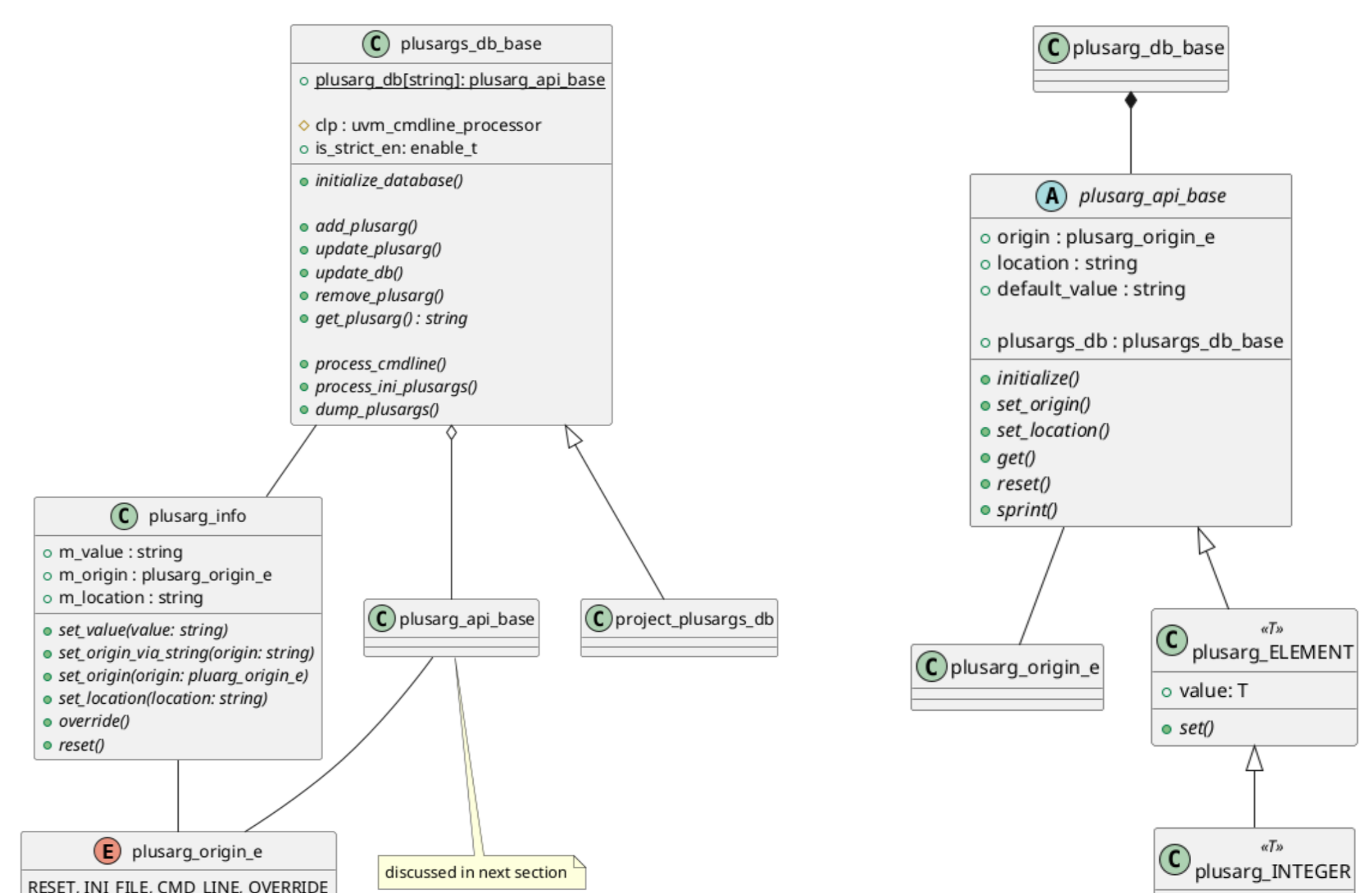
The Solutions

- Coerce plusarg values into more complex SV datatypes, expanding to any enum, lists and arrays, and valid file and folder paths.
- Require definitions of plusargs with unique names, descriptions, valid ranges, datatypes etc.
- Read said definition file, then generate SV parameterized classes for each plusarg and generate documentation automagically from the attributes.
- Script validates plusarg values from cmdline, ini files, and default plusarg values before build process to avoid the case where a misused plusarg leads to incorrect simulation behavior.

Python Pre-validation



SystemVerilog Runtime validation



Specifying plusargs

```
[num_of_pkts]
;; Example: +num_of_pkts=500
type = AN_INTEGER
description = an integer controlling the number of packets your simulation
accepts.

sv_type = int unsigned
default_value = 500

valid_range = [100,10000,1]

[mode]
;; Example: +mode=MODEZ
type = AN_ENUM
description = an enum of type ~mode_e~ that configures all control register
for a simulation's mission mode.

sv_type = mode_e
default_value = MODEZ

[mem_init_file]
;; Example:
+mem_init_file=$GIT_HOME/project/misc/mem/init/mem.init.pattern.hex
type = A_FILEPATH
description = holds the full filepath to the memory initialization file.

sv_type = string
default_value =
```

Accessing Plusarg Values

```
1 // Get the singleton instance of the plusargs_db
2 my_proj_plusargs_db = project_plusargs_db::get_instance();
3
4 // Configure the mission mode.
5 this.configure_mission_mode(my_project_plusargs.mode.value);
6
7 // Display the mission mode
8 `uvm_info("Configured for mission mode: %s.",
9         my_project_plusargs.mode.get())
```