UVM Configuration Database (uvm_config_db)

- The uvm_config_db class is the recommended way to access the resource database. A resource is any piece of information that is shared b/w two or more components or objects.

- Use uvm_config_db::set to put information into the database and use uvm_config_db::get to retrieve information from the database.

- There are no limitations on the type parameter, which can be a class, a uvm_object, a build-in type like a bit, byte, or a virtual interface, etc.

- There are two typical uses of the uvm_config_db. The first is to pass virtual interfaces from the HDL/DUT domain to the test, and the second is to pass configuration objects down through the testbench hierarchy.

1. Set :—

The full signature of the set method is void uvm_config_db #(type T=int) :: set (uvm_component cntxt, string inst_name, string field_name, T value);

- T is the type of the resource, or element, being added - usually a virtual interface or a configuration object.
- cntxt and inst_name together from a scope that is used to locate the resource within the database. It is formed by appending the instance name to the full hierarchical name of the context. i.e.,

  {cntxt.get_full_name(), ".", inst_name}
- field_name is the name given to the resource.
- value is the actual value of reference that is put into the database.

Example: Putting virtual interfaces into the UVM configuration.

```
interface ahb_if data_port_if (clk, reset);
interface ahb_if control_port_if (clk, reset);
---
uvm_config_db# (virtual ahb_if) :: set (null, "uvm_test_top"
, "data_port", data_port_if);

uvm_config_db #(virtual ahb_if) :: set (null, "uvm_test_top",
"control_port", control_port_if);
```

```
class env extends uvm_env;
    ahb_agent_config m_ahb_agent_config;

    function void build_phase(uvm_phase phase);
      ---
      m_ahb_agent = ahb_agent::type_id::create("m_ahb_
                                    agent", this);
      ---
      uvm_config_db#(ahb_agent_config)::set(this,"m_ahb
      _agent*","ahb_agent_config", m_ahb_agent_config);
      ---
    endfunction
endclass
```

This code sets the configuration for the AHB agent and all its child components. Two things to note:

- Use "this" as the first argument to ensure that only this agent's configuration is set, and not of any other ahb_agent in the component hierarchy.
- Use "m_ahb_agent" to ensure that both the agent and its children are in the look-up scope. Without the '*' only the agent itself would be, and its driver, sequence and monitor sub-components would be unable to access the configuration.

2. get method :-

The full signature of the get method is

bit uvm_config_db #(type T = int) :: get (uvm_component cntxt, string inst_name, string field_name, ref T value);

- T is the type of the resource, or element, being retrieved - usually a virtual interface or a configuration object.
- cntxt and inst_name together from a scope that is used to locate the resource within the database; it is formed by appending the instance name to the full hierarchical name of the context, i.e.
  {cntxt.get_full_name(), ".", inst_name}
- field_name is the name given to the resource.
- value holds the actual value or reference that is retrieved from the database; the get() call returns 1, if that retrieval succeeds, or else 0 indicating that no resource of this type and with this context and name exists in the database.

Example: class ahb_monitor extends uvm_monitor;
  ahb_agent_config m_cfg;

  function void build_phase (uvm_phase phase);
  ---
  if(!uvm_config_db #(ahb_agent_config) :: get (this, "", "ahb_agent_config", m_cfg))

```
begin
    `uvm_error("config Error", "uvm_config_db #(ahb_
    agent_config)::get cannot find resource ahb_agent
    _config")
end
---
endfunction
endclass
```