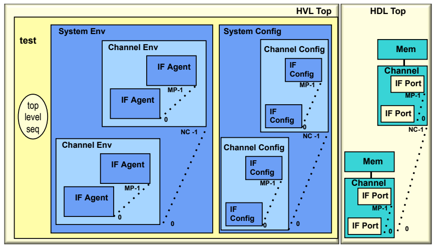
Rockin' the Polymorphism for an Elegant UVM Testbench Architecture

for a Scalable, Highly Configurable, Extensible DUT

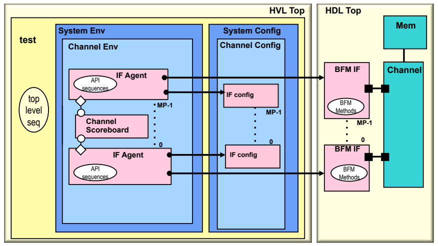
Mike Baird, Willamette HDL, mike@whdl.com
Frank Verhoorn, Northwest Logic, fverhoorn@nwlogic.com

Testbench Architecture



NC = NUM_CHANNELS, MP = MAX_PORTS

Single Channel Architecture



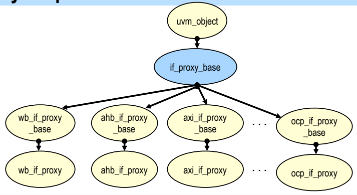
MP = MAX_PORTS

Specifying the Configuration

```
package top_params_pkg;
  parameter NUM_CHANNELS = 2;  // Num Memory controller channels
  parameter MAX_PORTS    = 8;  // Maximum num ports per channel

  // Multi-dimensional array of port types using strings to identify the type
  // The string is the prefix for the channel type
  // i.e. wb_ for wishbone_bus, ahb_ for AHB, axi_ for AXI etc.
  // a null string is the default meaning no IF connected to that port
  string if_port_types[NUM_CHANNELS][MAX_PORTS];
  ...
endpackage
```

Polymorphism and Testbench – DUT Connection

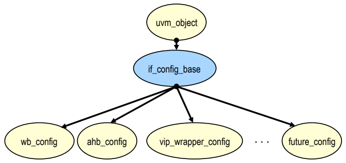


```
class wb_if_proxy_base extends if_proxy_base; // if_proxy_base is empty
...
 // API methods
  virtual task write(input wb_txn txn); endtask
  virtual task read(input wb_txn txn);  endtask
  virtual task monitor(output wb_txn txn); endtask
endclass
```

```
interface wb_bfm_if #(int CH_NUM = 0, int PORT_NUM = 0) ();
class wb_if_proxy extends wb_pkg::wb_if_proxy_base;
...
  // Overrides of the proxy base API methods
  virtual task write(input wb_txn txn);   ...  endtask
  // Overrides of read() and monitor() not shown
endclass
wb_if_proxy wb_proxy; // declare derived proxy if handle
initial begin
  wb_proxy = new("wb_proxy");  // create derived proxy if object
  // place proxy if object in config_db NOTE use of if_proxy_base
  uvm_config_db#(sys_config_pkg::if_proxy_base)::set(null,"PROXY_IF",
       $sformatf("proxy_if_ch%0d_p%0d",CH_NUM, PORT_NUM), wb_proxy );
end
endinterface
```

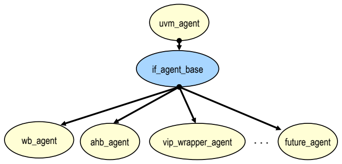
```
class test_base extends uvm_test;
  // Interface proxy objects array
  if_proxy_base p_if [NUM_CHANNELS][MAX_PORTS];
   // get proxy ifs
   foreach(p_if[i,j])
     if(if_port_types[i][j] != "")  // interface present?
       // yes get proxy object
       if(!uvm_config_db #( if_proxy_base )::get(null,"PROXY_IF",
              $sformatf("proxy_if_ch%0d_p%0d",i, j), p_if[i][j] ))
         `uvm_fatal(...)
       else `uvm_info(...)
endclass
```

Polymorphism and Configuration Objects



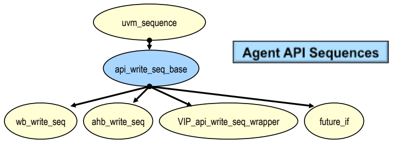
```
class chan_configuration extends uvm_object;
`uvm_object_utils(chan_configuration)
 global_configuration global_config; // global configuration object
 int chan_number; // Channel number this config object
 // Fetch factory handle
 protected uvm_factory m_factory = uvm_factory::get();
 // Child configuration objects
 if_config_base  if_config[]; // array of if configurations
 ... // constructor not shown
 function void initialize(
    global_configuration global_config_arg, // global config object
    int chan_number_arg,                    // channel ID
    string if_port_types_arg[MAX_PORTS],    // port types array
    if_proxy_base p_if_arg [MAX_PORTS]   // Interface proxy objects
  );
  chan_number   = chan_number_arg;    // set local channel number
  global_config = global_config_arg;  // Set local global config
  // create IF configuration objects
  if_config = new[global_config.max_ports]; // resize array
  foreach(if_config[i])
    if(if_port_types_arg[i] != "") begin    // port connected?
     $cast(if_config[i], m_factory.create_object_by_name(
       // name of type to create
       {if_port_types_arg[i],"configuration"},
       "", // parent
       // name of object
       {if_port_types_arg[i],$sformatf("config_ch%0d_p%0d",
                                  chan_number, i)}
                       )
        );
...
endfunction
```

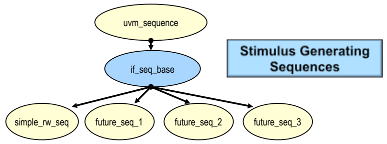
Polymorphism and Agents



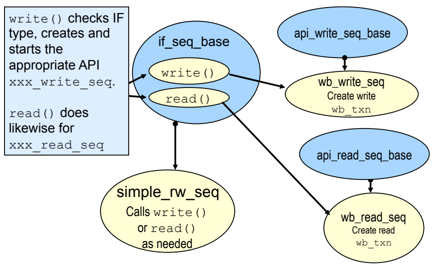
```
class chan_environment extends uvm_env;
 if_agent_base       if_agent[];    // array of interface agents
...
 function void build_phase(uvm_phase phase);
  if_agent = new[global_config.max_ports]; // resize agent array
  foreach(if_agent[i]) // walk through every port
   if(chan_config.if_config[i] == null)
    ; // not connected - do nothing
   else begin
    // create the agent using the string based factory
    $cast(if_agent[i], m_factory.create_component_by_name(
           {chan_config.if_config[i].if_type,"agent"},
           this.get_full_name(),
           {chan_config.if_config[i].if_type,
             $sformatf("agent_ch%0d_p%0d",
                 chan_config.get_chan_num(), i)},
            this)
        );
    // set agent config object
    if_agent[i].set_config(chan_config.if_config[i]);
   end
 endfunction
```

Polymorphism and Sequences





Sequence Inheritance and "Calling" Flow



`write()` checks IF type, creates and starts the appropriate API `xxx_write_seq`.

`read()` does likewise for `xxx_read_seq`