Translation

- Model often unsupported by a standard language

1. Use a **standard** language anyway
   - Many tools available
   - But, captures model unnaturally

2. Use an **application-specific** language
   - Captures model naturally
   - But, not many tools available

3. Use a **front-end** language
   - Captures model naturally
   - Many tools available after translating to a standard
Outline

- Front-end language in VHDL environment
- State machine translation
- Fork-join translation
- Exception translation
A front-end language in a VHDL environment

```
VHDL  SpecCharts
     |     |
     |     | Translator
     |     |
     |     |   VHDL
     |     |
     |     | VHDL environment
     |     |   Simulator  Debugger  Test-generator
     |     |
     |     | Tool output
```

Synthesis tool
State machine translation

type  state_type is (P, Q, R);
variable state : state_type := P;

loop
  case (state) is
  when P =>
    <actions for P>
    if (u) then
      state := Q;
    else if (not u) then
      state := R;
    end if;
  when Q =>
    <actions for Q>
    state := P;
  when R =>
    <actions for R>
    state := Q;
  end case;
end loop;

(a)  (b)
Fork-join translation

signal fork, P1_done, P2_done : boolean;

Main : process
begin
  statement1;
  parallel
  {
    P1;
    P2;
  }
  statement2;
  ...
  (a)
end;

P1_process : process
begin
  P1;
  fork <= true;
  wait until fork;
  P1;
  wait until P1_done and P2_done;
  P1_done <= true;
  wait until not fork;
  P1_done <= false;
end;

(b)
Exception translation

event e : T --> S;

T :
   statement1;
   statement2;
   statement3;

--- T
   statement1;
   if (e) goto S_start;
   statement2;
   if (e) goto S_start;
   statement3;
   exit T_loop;
   end loop;

S :
   statement4;
   statement5;

--- S
   statement4;
   statement5;

S_start: --- S
   statement4;
   statement5;

--- S
   statement4;
   statement5;
Summary

- The perfect standard language may never exist
- No standard language supports all models
- Using a front-end language solves the problem
  - Natural capture
  - Large base of tools and expertise
- Translators are simple
  - Maps characteristics to existing constructs
  - Generates well-structured and consistent output