Hi-Lite project

Final meeting

Presented by David LESENS

Wednesday, 29 May 2013

This document and its content is the property of Astrium [Ltd/SAS/GmbH] and is strictly confidential. It shall not be communicated to any third party without the written consent of Astrium [Ltd/SAS/GmbH].

Together pioneering excellence

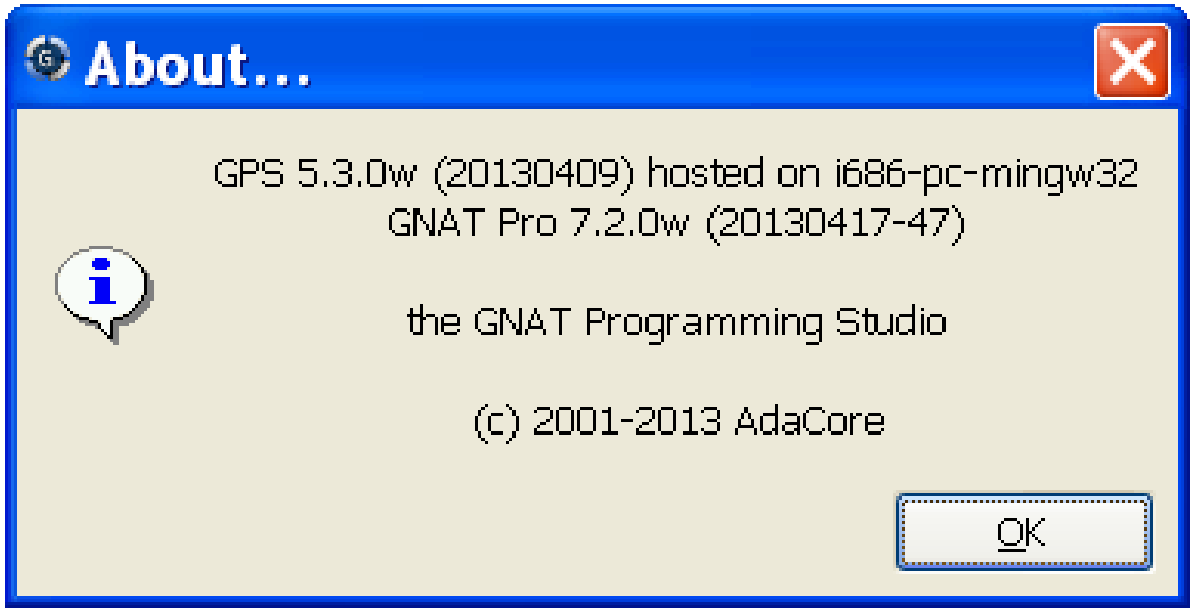
ASTRIUM
AN EADS COMPANY

Overview

- Introduction
- Astrium Space Transportation Case study
  - Numerical algorithm
  - Event driven
- Conclusion

This document and its content is the property of Astrium [Ltd/SAS/GmbH] and is strictly confidential. It shall not be communicated to any third party without the written consent of Astrium [Ltd/SAS/GmbH].

Tools



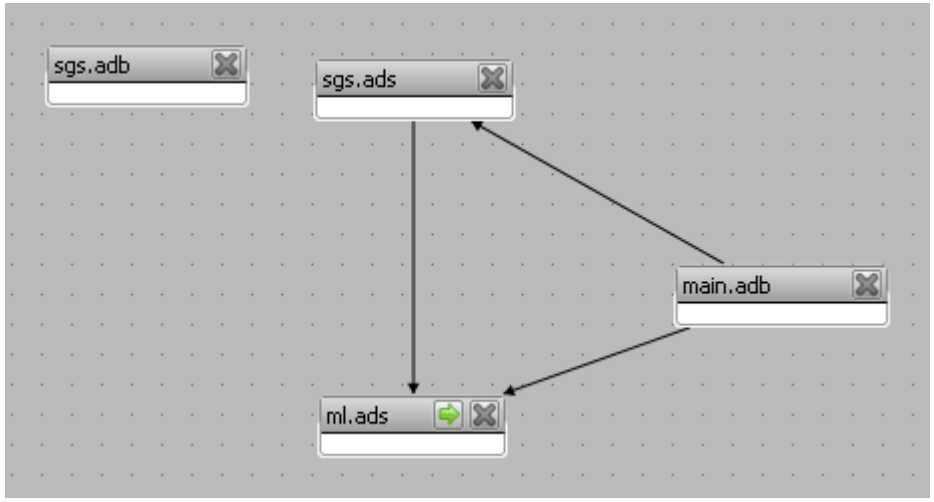
This document and its content is the property of Astrium [Ltd/SAS/GmbH] and is strictly confidential. It shall not be communicated to any third party without the written consent of Astrium [Ltd/SAS/GmbH].

Overview

- Introduction
- Astrium Space Transportation Case study
  - Numerical algorithm
  - Event driven
- Conclusion

This document and its content is the property of Astrium [Ltd/SAS/GmbH] and is strictly confidential. It shall not be communicated to any third party without the written consent of Astrium [Ltd/SAS/GmbH].

Architecture



This document and its content is the property of Astrium [Ltd/SAS/GmbH] and is strictly confidential. It shall not be communicated to any third party without the written consent of Astrium [Ltd/SAS/GmbH].

GPS - (Sgs project) - ml.ads

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\SGS\src\ml.ads

```
27    -- A signed integer type on 32 bits
28    type T_Int32 is range (-2 ** 31) .. (2 ** 31 - 1);
29    -- A signed integer type on 64 bits
30    type T_Int64 is range (-2 ** 63) .. (2 ** 63 - 1);
31
32    for T_Int32'Size use 32;
33    for T_Int64'Size use 64;
34
35    subtype T_Uint32 is T_Int32 range 0 .. (2 ** 31 - 1);
36    subtype T_Uint64 is T_Int64 range 0 .. (2 ** 63 - 1);
37
38    subtype T_Pos32 is T_Uint32 range 1 .. (2 ** 31 - 1);
39    subtype T_Pos64 is T_Uint64 range 1 .. (2 ** 63 - 1);
40
41    type T_Nat32 is range 0 .. (2 ** 32 - 1);
42    type T_Nat64 is range 0 .. (2 ** 63 - 1);
43
44    for T_Nat32'Size use 32;
45    for T_Nat64'Size use 64;
46
47    subtype T_Sign_Int32 is T_Int32 range -1 .. 1;
48
49    type T_Float32 is digits 6  range -3.40282E+38 .. 3.40282E+38;
50    type T_Float64 is digits 15 range -1.79769313486231E+308 .. 1.79769313486231E+308;
51    for T_Float32'Size use 32;
52    for T_Float64'Size use 64;
53
54    subtype T_Sign_Float32 is T_Float32 range -1.0 .. 1.0;
```

Types definitions

Unknown VCS (Modified) | Insert | Writable | Modified | 1:1

Dependency Browser | ml.ads | ml.adb | sgs.ads | sgs.adb

Messages

Messages | Locations

GPS - (Sgs project) - ml.ads

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\SGS\src\ml.ads

```
92
93      ----------------------------------------------------------------------
94
95       function Float32_To_Int32 (X : in T_Float32) return T_Int32
96       with
97        Pre  => (X >= T_Float32(T_Int32'First) and then
98                 X + 1.0 <= T_Float32(T_Int32'Last)),
99        Post => Float32_To_Int32'Result = T_Int32 (X);
100
101     function Float32_To_Int32 (X : T_Float32) return T_Int32
102     is (T_Int32 (X));
103
104     ----------------------------------------------------------------------
105
106     function Int32_To_Float32 (X : T_Int32) return T_Float32
107     is (T_Float32 (X));
108
109     ----------------------------------------------------------------------
110
111     function Sign_Strict_Float32 (X : T_Int32) return T_Sign_Float32;
112
113     function Sign_Strict_Float32 (X : T_Int32) return T_Sign_Float32
114     is (if X >= 0 then 1.0 else -1.0);
115
116     ----------------------------------------------------------------------
117
118     function Absolute (X : T_Int32) return T_Int32
119     with
120       Pre => (X >= -T_Int32'Last);
```

Contracts

Unknown VCS (Modified) | Insert | Writable | Modified | 1:1

Dependency Browser | ml.ads | ml.adb | sgs.ads | sgs.adb

Messages

Messages | Locations

GPS - (Sgs project) - ml.ads

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\SGS\src\ml.ads

```
177    function Cos (X : T_Float32) return T_Float32
178    with
179      Pre => (X >= -C_2Pi32) and then
180      (X <= C_2Pi32),
181      Post => (Cos'Result >= -1.0) and then
182      (Cos'Result <= 1.0);
183
184    ----------------------------------------------------------------
185
186    function Arctan (X : T_Float32) return T_Float32
187    with
188      Post => (Arctan'Result >= -C_Halfpi32) and then
189      (Arctan'Result <= C_Halfpi32);
190
191    ----------------------------------------------------------------
192
193    function Saturate (X     : T_Float32;
194                       X_Min : T_Float32;
195                       X_Max : T_Float32) return T_Float32
196    with
197      Pre => (X_Min < X_Max);
198
199    function Saturate (X     : T_Float32;
200                       X_Min : T_Float32;
201                       X_Max : T_Float32) return T_Float32
202    is (if X < X_Min then X_Min
203        elsif X > X_Max then X_Max
204        else X);
```

Contracts

Unknown VCS (Modified) | Insert | Writable | Modified | 1:1

Dependency Browser | ml.ads | ml.adb | sgs.ads | sgs.adb

Messages

Messages | Locations

GPS - (Sgs project) - ml.adb

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\SGS\src\ml.adb

```
16  -- Author: David Lesens
17  --                                                                    --
18  ------------------------------------------------------------------------
19
20  with Ada.Numerics.Generic_Elementary_Functions;
21
22  package body Ml is
23
24       package Num32 is new Ada.Numerics.Generic_Elementary_Functions (T_Float32);
25
26     ----------------------------------------------------------------------
27
28     function Arctan (X : T_Float32) return T_Float32
29     is (Num32.Arctan (X));
30
31     ----------------------------------------------------------------------
32
33     function Cos (X : T_Float32) return T_Float32
34     is (Num32.Cos (X));
35
36     ----------------------------------------------------------------------
37
38     function Sin (X : T_Float32) return T_Float32
39     is (Num32.Sin (X));
40
41     ----------------------------------------------------------------------
42
43  end Ml;
```

Trigonometric functions

Insert | Writable | Unmodified | 1:1

Dependency Browser | ml.ads | ml.adb | sgs.ads | sgs.adb

Messages

Messages | Locations

GPS - (Sgs project) - sgs.adb

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\SGS\src\sgs.adb

```
175                 + (
176                   C_SGS_SADM_STEP_ANG_VALUE
177                   *
178                     (
179                      SGS_SADE_MOTOR_STEPS_MOVED(SADE_I,SADM_J)
180                      + (
181                        SGS_SADE_MOTOR_U_STEPS_MOVED(SADE_I,SADM_J)/ C_SGS_SADM_MICRO_STEPS_NUMBER
182                       )
183                     )
184                 )
185               ));
186
187           else    -- SGS_MOTORj_TURNING_SENS(T-3) = CLOCK_WISE
188
189              SGS_MOTOR_MEASURED_ANGLE_STATE(T_2)(SADM_J) := ( S_NORMALISE_0_360(
190                (SGS_MOTOR_MEASURED_ANGLE_STATE(T_3)(SADM_J))
191                - (
192                  C_SGS_SADM_STEP_ANG_VALUE
193                  *
194                    (
195                     SGS_SADE_MOTOR_STEPS_MOVED(SADE_I,SADM_J)
196                     + (
197                       SGS_SADE_MOTOR_U_STEPS_MOVED(SADE_I,SADM_J)/ C_SGS_SADM_MICRO_STEPS_NUMBER
198                      )
199                    )
200                 )
201               ));
202
```

Insert | Writable | Unmodified | 1:1

Dependency Browser | ml.ads | ml.adb | sgs.ads | sgs.adb

Messages

Messages | Locations

Numerical algorithms

Results

This document and its content is the property of Astrium [Ltd/SAS/GmbH] and is strictly confidential. It shall not be communicated to any third party without the written consent of Astrium [Ltd/SAS/GmbH].

See PDF

Overview

- Introduction
- Astrium Space Transportation Case study
  - Numerical algorithm
  - Event driven
- Conclusion

This document and its content is the property of Astrium [Ltd/SAS/GmbH] and is strictly confidential. It shall not be communicated to any third party without the written consent of Astrium [Ltd/SAS/GmbH].

Space engineering

Spacecraft on-board control procedures

EUROPEAN COOPERATION
ECSS
FOR SPACE STANDARDIZATION

- On-board control procedure
  - Software program designed to be executed by an OBCP engine, which can easily be loaded, executed, and also replaced, on-board the spacecraft
- OBCP code
  - Complete representation of an OBCP, in a form that can be loaded on-board for subsequent execution
- OBCP engine
  - Application of the on-board software handling the execution of OBCPs
- OBCP language
  - Programming language in which OBCP source code is expressed by human programmers

This document and its content is the property of Astrium [Ltd/SAS/GmbH]. It is strictly confidential. It shall not be communicated to any third party without the written consent of Astrium [Ltd/SAS/GmbH].

GPS - (Mvm project) - Project

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

Project

- Mvm
  - .
  - OBJ
  - src
  - src\instance
  - src\MVM
  - src\MVM\debug
  - src\MVM\generic
  - testing
  - testing\um
  - OBJ
  - AUnit

- ✓ Generic (flight software)
- ✓ Debug
- ✓ Instances (for testing and formal proof)
- ✓ Tests
- ✓ User manual

Messages

Messages Locations

GPS - (Mvm project) - Messages

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

Project

- Mvm
  - .
    - mvm-data-pool.adb
    - mvm-data-pool.ads
    - mvm-data.ads
    - mvm-g_events.adb
    - mvm-g_events.ads
    - mvm-g_expressions.adb
    - mvm-g_expressions.ads
    - mvm-g_fdir.adb
    - mvm-g_fdir.ads
    - mvm-g_fu-efu.adb
    - mvm-g_fu-efu.ads
    - mvm-g_fu-fsm.adb
    - mvm-g_fu-fsm.ads
    - mvm-g_fu.ads
    - mvm-g_fus.adb
    - mvm-g_fus.ads
    - mvm-g_obcpe.adb
    - mvm-g_obcpe.ads
    - mvm-g_parameter-g_command-g_call.ads
    - mvm-g_parameter-g_command-g_commands.ads
    - mvm-g_parameter-g_command.adb
    - mvm-g_parameter-g_command.ads
    - mvm-g_parameter.ads

Size of the case study

Messages

Messages Locations

GPS - (Mvm project) - Messages

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

Project

- mvm-g_fu-fsm.ads
- mvm-g_fu.ads
- mvm-g_fus.adb
- mvm-g_fus.ads
- mvm-g_obcpe.adb
- mvm-g_obcpe.ads
- mvm-g_parameter-g_command-g_call.ads
- mvm-g_parameter-g_command-g_commands.ads
- mvm-g_parameter-g_command.adb
- mvm-g_parameter-g_command.ads
- mvm-g_parameter.ads
- mvm-g_vaps.adb
- mvm-g_vaps.ads
- mvm-g_variables.adb
- mvm-g_variables.ads
- mvm-ml-bits.adb
- mvm-ml-bits.ads
- mvm-ml.adb
- mvm-ml.ads
- mvm-obit.adb
- mvm-obit.ads
- mvm-variable.adb
- mvm-variable.ads
- mvm.ads
- ..\..\..\OBJ

Size of the case study

Messages

Messages Locations

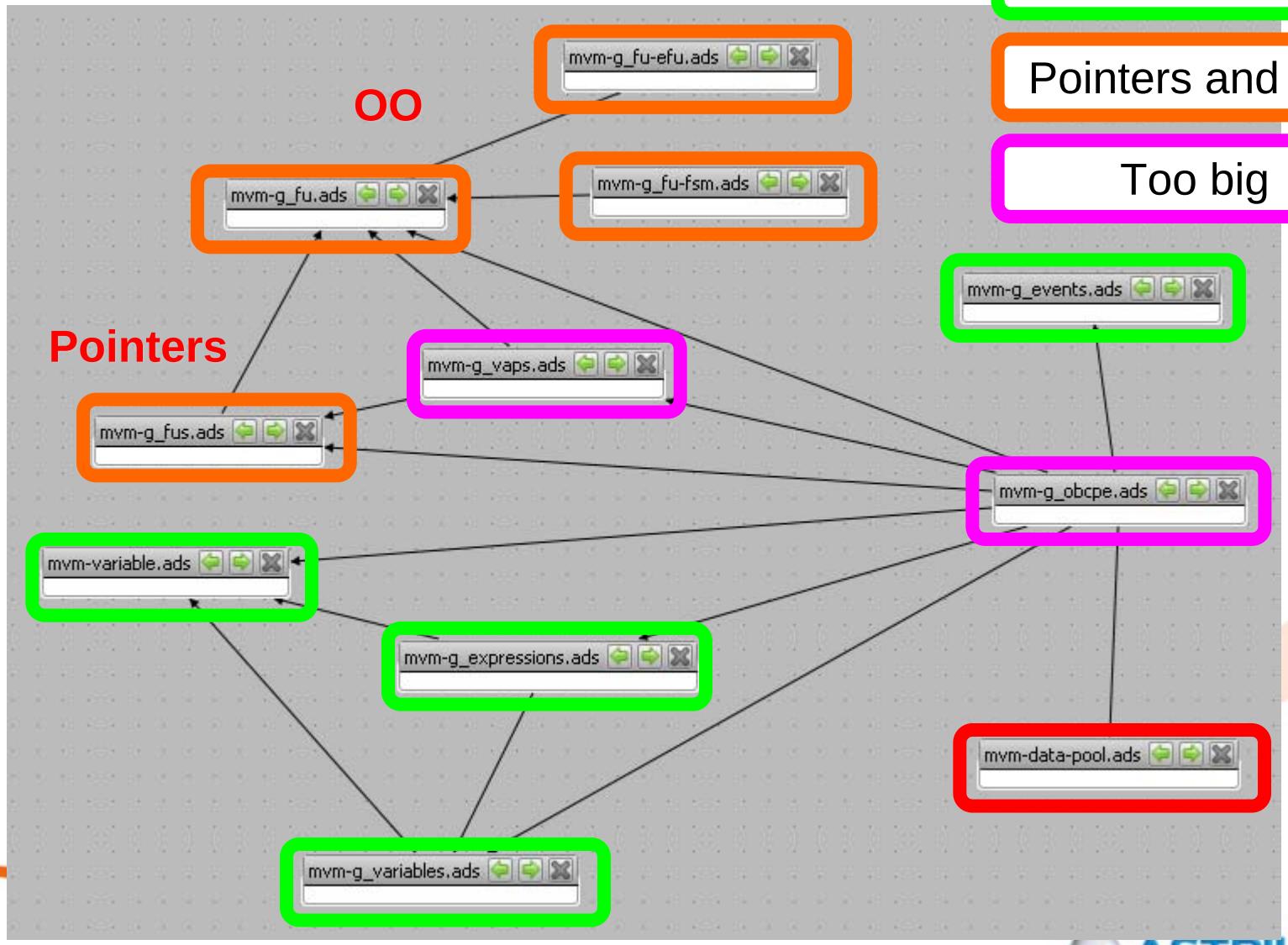
MVM (simplified) architecture

Not in SPARK

Proved

Pointers and OO

Too big



This document and its content is the property of Astrium [Ltd/SAS/GmbH] and is strictly confidential. It shall not be communicated to any third party without the written consent of Astrium [Ltd/SAS/GmbH].

GPS - (Mvm project) - mvm-ml-bits.adb

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\MVM\src\MVM\generic\mvm-ml-bits.adb

```
3572
3573
3574
3575
3576
3577
3578
3579
3580
3581
3582
3583   function Buffer_32bit_To_Float32 (X : T_Bit_Buffer) return T_Float32 is
3584   begin
3585      return Cast_From_32bit_Buffer_To_Float32 (X);
3586   end Buffer_32bit_To_Float32;
3587
3588
3589
3590
3591
3592
3593
3594
3595
3596
3597
3598
3599
3600
```

Not in SPARK

Buffer_Bit_To_Boolean | Unknown VCS (Modified) | Insert | Writable | Modified | 3568:24

Messages | Locations | mvm-ml-bits.adb

GPS - (Mvm project) - mvm-data-pool.adb

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\MVM\src\MVM\generic\mvm-data-pool.adb

```
265
266    function Unchecked_Read_Float32 (Data_Id   : T_Data_Id;
267                                     Data_Pool : T_Data_Pool) return T_Float32 is
268       Definition : T_Data_Definition;
269    begin
270       Definition := Data_Pool.Definition (Data_Id);
271       return Mvm.Ml.Bits.Buffer_Bit_To_Float32
272         (Mvm.Ml.Bits.T_Bit_Buffer
273            (Data_Pool.Storage
274               (Definition.First_Bit .. Definition.Last_Bit)));
275    end Unchecked_Read_Float32;
276
277    ------------------------------------------------------------------------
278
279    function Unchecked_Read_Float64 (Data_Id   : T_Data_Id;
280                                     Data_Pool : T_Data_Pool) return T_Float64 is
281       Definition : T_Data_Definition;
282    begin
283       Definition := Data_Pool.Definition (Data_Id);
284       return Mvm.Ml.Bits.Buffer_Bit_To_Float64
285         (Mvm.Ml.Bits.T_Bit_Buffer
286            (Data_Pool.Storage
287               (Definition.First_Bit .. Definition.Last_Bit)));
288    end Unchecked_Read_Float64;
289
290    ------------------------------------------------------------------------
291
292    function Unchecked_Read_Int32 (Data_Id   : T_Data_Id;
293                                   Data_Pool : T_Data_Pool) return T_Int32 is
294       Definition : T_Data_Definition;
295    begin
296       Definition := Data_Pool.Definition (Data_Id);
```

Not in SPARK

Insert | Writable | Unmodified | 1:1

GPS - (Mvm project) - mvm-g_events.ads

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\MVM\src\MVM\generic\mvm-g_events.ads

```
24  -- Status                                                          --
25  --                                                                  --
26  -- Development: Completed
27  -- Test        : Completed
28  -- Proof       : Completed
29  --
30  ----------------------------------------------------------------------
31
32  with Mvm.Obit;
33  use Mvm.Obit;
34
35  generic
36
37     -- The list of events
38     type T_Event_Id is (<>);
39
40     --*************************************************************************
41     --*                                                                       *
42     --*          00000    0    00000  0  0      0    00000  00000            *
43     --*          0   0   0 0   0      0 0      0 0   0      0                *
44     --*          00000  0   0  0      00      0   0  0      0                *
45     --*          0      00000  0      0 0     00000  0  00  0000             *
46     --*          0      0   0  0      0  0    0   0  0   0  0                *
47     --*          0      0   0  0      0   0   0   0  0   0  0                *
48     --*          0      0   0  00000  0    0  0   0  00000  00000            *
49     --*                                                                       *
50     --*************************************************************************
51  package Mvm.G_Events is
52
53     ---------------------------------------------------------------------
54
```

Generic

Unknown VCS (Modified) | Insert | Writable | Unmodified | 1:1

Messages | Locations

GPS - (Mvm project) - mvm-g_obcpe.ads

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\MVM\src\MVM\generic\mvm-g_obcpe.ads

```
generic

   type T_Plan_Id is (<>);
   type T_Event_Id is (<>);
   C_Max_Of_Plan_Steps : T_Pos32;
   type T_Variable_Id is (<>);
   type T_Fu_Id is (<>);
   type T_State_Id is (<>);
   type T_Fus_Command_Id is (<>);
   Max_Of_Parameter_Nb : T_Pos32;
   type T_Vap_Id is (<>);
   C_Max_Of_Vap_Steps : T_Pos32;
   type T_Data_Id is (<>);
   type T_Data_Pool_Storage is private;
   type T_Superstate_Types_Id is (<>);
   type T_Fus_Superstate_Id is (<>);

package Mvm.G_Obcpe is
```

Generic

Unknown VCS (Modified) | Insert | Writable | Modified | 34:20

Messages | Locations

GPS - (Mvm project) - mvm-g_events.ads

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\MVM\src\MVM\generic\mvm-g_events.ads

```
56
57    type T_Event
58      (
59       Event_Status        : T_Event_Status        := Inactive;
60       Type_Of_Monitoring  : T_Type_Of_Monitoring  := No_Window;
61       Type_Of_Time        : T_Type_Of_Time        := Absolute_Time)
62    is
63       record
64          case Event_Status is
65             when Inactive | Failed_To_Be_Detected => null;
66             when Not_Yet_Detected =>
67                Number_Of_Detection : T_Number_Of_Detection;
68                Threshold           : T_Number_Of_Detection;
69                case Type_Of_Monitoring is
70                when No_Window => null;
71                when Time_Window_Start_Not_Protected_End_Not_Protected
72                   | Time_Window_Start_Protected_End_Not_Protected
73                   | Time_Window_Start_Not_Protected_End_Protected
74                   | Time_Window_Start_Protected_End_Protected =>
75                   Start_Window : T_Numerical_Obit;
76                   End_Window   : T_Numerical_Obit;
77                   case Type_Of_Time is
78                      when Absolute_Time     => null;
79                      when Relative_To_Event => Origin_Event : T_Event_Id;
80                   end case;
81                end case;
82             when Detected =>
83                Detection_Time : T_Numerical_Obit;
84          end case;
85       end record;
86
```

Discriminant

Mvm.G_Events | Unknown VCS (Modified) | Insert | Writable | Modified | 50:62

Messages | Locations

GPS - (Mvm project) - mvm-g_events.ads

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\MVM\src\MVM\generic\mvm-g_events.ads

```
 85
 86    function Is_Valid (Event : T_Event) return Boolean is
 87      (case Event.Event_Status is
 88       when Inactive | Detected | Failed_To_Be_Detected => True,
 89       when Not_Yet_Detected =>
 90         (case Event.Type_Of_Monitoring is
 91          when No_Window                                          =>
 92            (Event.Number_Of_Detection < Event.Threshold),
 93          when Time_Window_Start_Not_Protected_End_Not_Protected   =>
 94            (Event.Number_Of_Detection < Event.Threshold and then
 95             Event.Start_Window <= Event.End_Window),
 96          when Time_Window_Start_Protected_End_Not_Protected
 97          | Time_Window_Start_Not_Protected_End_Protected
 98          | Time_Window_Start_Protected_End_Protected             =>
 99            (Event.Number_Of_Detection <= Event.Threshold and then
100               (Event.Start_Window <= Event.End_Window))))));
101
102    ------------------------------------------------------------------------
103
104    -- The detection statuses of all events
105    type T_Events is private;
106    function Is_Valid (Events : T_Events) return Boolean;
107
108    ------------------------------------------------------------------------
109
110    -- Returns an event status from its Id
111    function Get_Event (Event_Id : T_Event_Id;
112                        Events   : T_Events)
113                        return T_Event
114    with Pre => Is_Valid (Events);
115
```

Validity function

Private list

Mvm.G_Events | Unknown VCS (Modified) | Insert | Writable | Modified | 77:74

GPS - (Mvm project) - mvm-g_events.ads

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\MVM\src\MVM\generic\mvm-g_events.ads

```
   --*****************************************************************
private
   -- ***************************************************************** --

   ------------------------------------------------------------------------

   type T_Events is array (T_Event_Id) of T_Event;

   ------------------------------------------------------------------------

   function Is_Valid (Events : T_Events) return Boolean is
     (for all Event_Id in T_Event_Id => Is_Valid (Event => Events (Event_Id)));

   ------------------------------------------------------------------------

   function Init return T_Events is
     (T_Events'(others => T_Event'
                 (Event_Status       => Inactive,
                  Type_Of_Monitoring => No_Window,
                  Type_Of_Time       => Absolute_Time)));

   ------------------------------------------------------------------------

   function Get_Event (Event_Id : T_Event_Id;
                       Events   : T_Events)
                       return T_Event
   is (Events (Event_Id));

   ------------------------------------------------------------------------

   function Is_Event_Monitoring_Detected (Event_Id : T_Event_Id;
```

Private list

Validity function

Mvm.G_Events | Unknown VCS (Modified) | Insert | Writable | Modified | 105:15

GPS - (Mvm project) - mvm-g_events.ads

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\MVM\src\MVM\generic\mvm-g_events.ads

```
456
457     -- Update the status of all events following time passing
458     procedure Check_Monitoring (Events       : in out T_Events;
459                                 Current_Time :        Mvm.Obit.T_Obit)
460     with
461       Pre  => Is_Valid (Events),
462       Post =>
463         (Is_Valid (Events) and then
464            (for all Event_Id in T_Event_Id =>
465               (case Get_Event (Event_Id, Events'Old).Event_Status is
466                when Inactive | Detected | Failed_To_Be_Detected =>
467                  (Get_Event (Event_Id, Events) = Get_Event (Event_Id, Events'Old)),
468                when Not_Yet_Detected                                =>
469                  (Get_Event (Event_Id, Events).Event_Status = Failed_To_Be_Detected or else
470                   Get_Event (Event_Id, Events).Event_Status = Not_Yet_Detected or else
471                   Get_Event (Event_Id, Events).Event_Status = Detected))));
472
473     ------------------------------------------------------------------------
474
475     -- Returns true iff the event has been detected
476     function Is_Event_Monitoring_Detected (Event_Id : T_Event_Id;
477                                            Events   : T_Events)
478                                            return Boolean with
479       Pre => Is_Valid (Events),
480       Post =>
481         (Is_Event_Monitoring_Detected'Result =
482              (case Get_Event (Event_Id, Events).Event_Status is
483               when Inactive | Not_Yet_Detected | Failed_To_Be_Detected => False,
484               when Detected                                           => True));
485
486     ------------------------------------------------------------------------
```

Subprogram on the whole list

Set_Event_Detection | Unknown VCS (Modified) | Insert | Writable | Modified | 439:47

GPS - (Mvm project) - mvm-g_events.adb

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\MVM\src\MVM\generic\mvm-g_events.adb

```
47
48
49
50
51
52
53
54
55  procedure Check_Monitoring (Event        : in out T_Event;
56                              Events       :        T_Events;
57                              Current_Time :        Mvm.Obit.T_Obit)
58  with
59    Pre  => Is_Valid (Event => Event),
60    Post =>
61      Is_Valid (Event => Event),
62       Contract_Cases => ((Event.Event_Status in Inactive | Detected | Failed_To_Be_Detected) =>
63                            (Event = Event'Old),
64                          (Event.Event_Status in Not_Yet_Detected)                            =>
65                            (Event.Event_Status = Not_Yet_Detected or else
66                             Event.Event_Status = Detected or else
67                             Event.Event_Status = Failed_To_Be_Detected));
68
69
70
71
72
73
74
75
76
77
```

Subprogram on a single element

Unknown VCS (Modified) | Insert | Writable | Saved | 1:1

GPS - (Mvm project) - mvm-g_events.adb

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\MVM\src\MVM\generic\mvm-g_events.adb

```
procedure Check_Monitoring (Events       : in out T_Events;
                            Current_Time :        Mvm.Obit.T_Obit) is
   Event : T_Event;
begin
   for Event_Id in T_Event_Id loop
      Event := Events (Event_Id);
      Check_Monitoring (Event        => Event,
                        Events       => Events,
                        Current_Time => Current_Time);
      pragma Assert (Is_Valid (Event => Event));
      pragma Assert (Is_Valid (Events => Events));
      Events (Event_Id) := Event;
      pragma Assert (Is_Valid (Events => Events));
      pragma Loop_Invariant
        (Is_Valid (Events => Events) and then
           (for all I in T_Event_Id range T_Event_Id'First .. Event_Id =>
              (case Events'Loop_Entry (I).Event_Status is
               when Inactive | Detected | Failed_To_Be_Detected =>
                 (Events (I) = Events'Loop_Entry (I)),
               when Not_Yet_Detected                            =>
                 (Events (I).Event_Status = Not_Yet_Detected or else
                  Events (I).Event_Status = Detected or else
                  Events (I).Event_Status = Failed_To_Be_Detected))) and then
           (if Event_Id /= T_Event_Id'Last then
              (for all I in T_Event_Id'Succ (Event_Id) .. T_Event_Id'Last =>
               Events (I) = Events'Loop_Entry (I))))));
   end loop;
end Check_Monitoring;

--------------------------------------------------------------------------
```

Additional assertion

Loop invariant

Check_Monitoring | Unknown VCS (Modified) | Insert | Writable | Modified | 120:20

At 1KHz

On 32 bits	On 64 bits	
2147483648	9,22337E+18	
2147483,648	9,22337E+15	seconds
596,5232356	2,56205E+12	hours
24,85513481	1,06752E+11	days
0,067910204	291672107	years
0,000679102	2916721,07	centuries

```
pragma Assert (Event.Number_Of_Detection <= Event.Threshold);

if Event.Number_Of_Detection < Event.Threshold then
   Event.Number_Of_Detection := Event.Number_Of_Detection + 1;
end if;

pragma Assert (Is_Valid (Event => Event));
if (Event.Type_Of_Time = Absolute_Time and then
    Numerical_Obit >= Event.End_Window) or else
  (Event.Type_Of_Time = Relative_To_Event and then
   Events (Event.Origin_Event).Event_Status = Detected and then
   Numerical_Obit >= Event.End_Window + Events (Event.Origin_Event).Detection_Time)
then
   if Event.Event_Status /= Detected then
      Event := T_Event'(Event_Status       => Detected,
                        Detection_Time     => Numerical_Obit,
                        Type_Of_Monitoring => No_Window,
                        Type_Of_Time       => Absolute_Time);
      pragma Assert (Is_Valid (Event => Event));
   end if;
else
```

Additional numerical protection

Mvm.G_Events | Unknown VCS (Modified) | Insert | Writable | Modified | 53:4

Messages | Locations | mvm-g_events.ads

GPS - (Mvm project) - mvm-g_expressions.ads

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\MVM\src\MVM\generic\mvm-g_expressions.ads

```
69
70    type T_Expression (Expression_Type : T_Expression_Type := Onboard_Time_Expression)
71    is
72       record
73          Data_Type : T_Data_Type;
74          case Expression_Type is
75             when Constant_Expression =>
76                Numerical_Value : Data.T_Numerical_Value;
77             when Onboard_Time_Expression =>
78                null;
79             when Equal_Expression | Different_Expression | Greater_Expression
80                | Greater_Or_Equal_Expression | Less_Expression
81                | Less_Or_Equal_Expression | And_Expression | Or_Expression
82                | Addition_Expression | Subtraction_Expression
83                | Multiplication_Expression | Division_Expression
84                | Not_Expression | Negation_Expression
85                | Abs_Expression | Square_Root_Expression  | Variable_Expression
86                | Cast_Nat64_To_Nat32 | Cast_Int32_To_Nat32
87                | Cast_Int64_To_Nat32 | Cast_Float32_To_Nat32
88                | Cast_Float64_To_Nat32 | Cast_Nat32_To_Nat64
89                | Cast_Int32_To_Nat64 | Cast_Int64_To_Nat64
90                | Cast_Float32_To_Nat64 | Cast_Float64_To_Nat64
91                | Cast_Nat32_To_Int32 | Cast_Nat64_To_Int32 | Cast_Int64_To_Int32
92                | Cast_Float32_To_Int32 | Cast_Float64_To_Int32
93                | Cast_Nat32_To_Int64 | Cast_Nat64_To_Int64 | Cast_Int32_To_Int64
94                | Cast_Float32_To_Int64 | Cast_Float64_To_Int64
95                | Cast_Nat32_To_Float32 | Cast_Nat64_To_Float32
96                | Cast_Int32_To_Float32 | Cast_Int64_To_Float32
97                | Cast_Float64_To_Float32 | Cast_Nat32_To_Float64
98                | Cast_Nat64_To_Float64 | Cast_Int32_To_Float64
```

Bigger structure

Mvm.G_Expressions | Unknown VCS (Modified) | Insert | Writable | Modified | 66:50

mvm-data-pool.adb | mvm-g_expressions.ads

Messages | Locations

GPS - (Mvm project) - mvm-g_expressions.ads

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\MVM\src\MVM\generic\mvm-g_expressions.ads

```
908
909    function Is_Valid (Expression     : T_Expression;
910                       Variable_Types : P_Variables.T_Variable_Types)
911                       return Boolean is
912      (P_Variables.Is_Valid (Variable_Types => Variable_Types) and then
913         (case Expression.Expression_Type is
914          when Variable_Expression          =>
915            (Expression.Data_Type =
916               Variable_Types (Expression.Left).Data_Type),
917          when Onboard_Time_Expression      =>
918            (Expression.Data_Type = Nat64_Type),
919          when Constant_Expression          =>
920            (Expression.Data_Type = Expression.Numerical_Value.Data_Type),
921          when Equal_Expression             =>
922            (Expression.Data_Type = Boolean_Type and then
923               (Variable_Types (Expression.Left).Data_Type =
924                  Variable_Types (Expression.Right).Data_Type)),
925          when Different_Expression         =>
926            (Expression.Data_Type = Boolean_Type and then
927               (Variable_Types (Expression.Left).Data_Type =
928                  Variable_Types (Expression.Right).Data_Type)),
          when Greater_Expression           =>
930            (Expression.Data_Type = Boolean_Type and then
931             Variable_Types (Expression.Left).Data_Type /= Boolean_Type and then
             Variable_Types (Expression.Right).Data_Type /= Boolean_Type and then
933               (Variable_Types (Expression.Left).Data_Type =
934                  Variable_Types (Expression.Right).Data_Type)),
935          when Greater_Or_Equal_Expression =>
936            (Expression.Data_Type = Boolean_Type and then
937             Variable_Types (Expression.Left).Data_Type /= Boolean_Type and then
```

Insert | Writable | Unmodified | 1:1

mvm-g_expressions.ads

Messages | Locations

Validity function

GPS - (Mvm project) - mvm-g_expressions.adb

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\MVM\src\MVM\generic\mvm-g_expressions.adb

```
1687
1688    ----------------------------------------------------------------------
1689
1690    procedure Compute (Expression   :     T_Expression;
1691                       Variables    :     T_Variables;
1692                       Current_Time :     Mvm.Obit.T_Obit;
1693                       Result       : out Data.T_Numerical_Value;
1694                       Is_Error     : out Boolean) is
1695    begin
1696       Is_Error := False;
1697       case Expression.Expression_Type is
1698          when Variable_Expression         =>
1699             Result := Compute_Variable (Expression => Expression, Variables => Variables);
1700             pragma Assert (Expression.Data_Type = Result.Data_Type);
1701          when Constant_Expression         =>
1702             Result := Compute_Constant (Expression => Expression, Variables => Variables);
1703             pragma Assert (Expression.Data_Type = Result.Data_Type);
1704          when Onboard_Time_Expression     =>
1705             Result := Compute_Onboard_Time (Expression => Expression, Variables => Variables, Current_Time
1706             pragma Assert (Expression.Data_Type = Result.Data_Type);
1707          when Equal_Expression            =>
1708             pragma Assert (Expression.Data_Type = Boolean_Type);
1709             pragma Assert (Get_Variable (Expression.Left, Variables).Data_Type =
1710                            Get_Variable (Expression.Right, Variables).Data_Type);
1711             Result := Compute_Equal (Expression => Expression, Variables => Variables);
1712             pragma Assert (Result.Data_Type = Boolean_Type);
1713             pragma Assert (Expression.Data_Type = Result.Data_Type);
1714          when Different_Expression        =>
1715             pragma Assert (Expression.Data_Type = Boolean_Type);
1716             pragma Assert (Get_Variable (Expression.Left, Variables).Data_Type =
```

Divide and Conquer

Insert | Writable | Unmodified | 1:1

mvm-g_expressions.adb

Messages | Locations

GPS - (Mvm project) - mvm-g_expressions.adb

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\MVM\src\MVM\generic\mvm-g_expressions.adb

```
90    function Compute_Equal (Expression : T_Expression;
91                            Variables  : T_Variables)
92                            return Data.T_Numerical_Value
93    with
94      Pre  =>
95        (Is_Valid (Expression => Expression, Variables  => Variables)
96         and then Expression.Expression_Type = Equal_Expression and then
97           (Get_Variable (Expression.Left, Variables).Data_Type =
98              Get_Variable (Expression.Right, Variables).Data_Type)),
99      Post => Compute_Equal'Result.Data_Type = Boolean_Type;
100
101   function Compute_Equal (Expression : T_Expression;
102                           Variables  : T_Variables)
103                           return Data.T_Numerical_Value is
104       Result : Data.T_Numerical_Value;
105   begin
106      pragma Assert (Get_Variable (Expression.Left, Variables).Data_Type =
107                        Get_Variable (Expression.Right, Variables).Data_Type);
108      case Get_Variable (Expression.Left, Variables).Data_Type is
109         when Boolean_Type =>
110            Result := Data.Create_Boolean_Value (P_Variables.Get_Boolean (Expression.Left, Variables) =
111                                                 P_Variables.Get_Boolean (Expression.Right, Variables));
112            pragma Assert (Result.Data_Type = Boolean_Type);
113         when Nat32_Type =>
114            Result := Data.Create_Boolean_Value (P_Variables.Get_Nat32 (Expression.Left, Variables) =
115                                                 P_Variables.Get_Nat32 (Expression.Right, Variables));
116            pragma Assert (Result.Data_Type = Boolean_Type);
117         when Int32_Type => Result := Data.Create_Boolean_Value (P_Variables.Get_Int32 (Expression.Left, V
118                                                                 P_Variables.Get_Int32 (Expression.Right
119            pragma Assert (Result.Data_Type = Boolean_Type);
```

Divide and Conquer

Compute_Onboard_Time | Insert | Writable | Unmodified | 85:32

mvm-data-pool.adb | mvm-g_expressions.adb

Messages | Locations

GPS - (Mvm project) - mvm-g_fus.ads

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\MVM\src\MVM\generic\mvm-g_fus.ads

```
   --*****************************************************************************
   --*                                                                           *
   --*          OOOOO  OOOOO  OOOOO  O      O    O    OOOOO  OOOOO              *
   --*          O   O  O   O    O    O      O   O O     O    O                  *
   --*          OOOOO  OOOOO    O     O    O   O   O    O    O                  *
   --*          O      OO       O     O    O   OOOOO    O    OOOO               *
   --*          O      O O      O      O  O    O   O    O    O                  *
   --*          O      O  O     O      O  O    O   O    O    O                  *
   --*          O      O   O  OOOOO     O      O   O    O    OOOOO              *
   --*                                                                           *
   --*****************************************************************************
private
   -- ***************************************************************************

   type T_Fus is array (T_Fu_Id) of access T_Fu'Class;

   ----------------------------------------------------------------------------

   -- Check the validity of a FU
   function Is_Valid (Fus    : T_Fus;
                      Fu_Id  : T_Fu_Id) return Boolean is
     (P_Gfu.Is_Valid (Fus (Fu_Id).all));

   ----------------------------------------------------------------------------

   -- Check the validity of all FUs
   function Is_Valid (Fus : T_Fus) return Boolean is
     (for all Fu_Id in T_Fu_Id => Is_Valid (Fus, Fu_Id));

   ----------------------------------------------------------------------------

   -- Check the validity of a FU
```

Array of objects

Pointers

Insert | Writable | Unmodified | 1:1

Messages | Locations

GPS - (Mvm project) - mvm-g_fu.ads

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\MVM\src\MVM\generic\mvm-g_fu.ads

```
53    --*****************************************************************************
54    --*                                                                           *
55    --*          00000    0    00000  0  0      0    00000  00000                *
56    --*          0   0   0 0   0      0 0      0 0   0      0                     *
57    --*          00000  0   0  0      00      0   0  0      0                     *
58    --*          0      00000  0      0 0     00000  0  00  0000                  *
59    --*          0      0   0  0      0  0    0   0  0   0  0                     *
60    --*          0      0   0  0      0   0   0   0  0   0  0                     *
61    --*          0      0   0  00000  0    0  0   0  00000  00000                 *
62    --*                                                                           *
63    --*****************************************************************************
64 package Mvm.G_Fu is
65
66    ------------------------------------------------------------------------------
67
68    type T_Gfu is abstract tagged private;
69
70    ------------------------------------------------------------------------------
71
72    package P_Parameter is new Mvm.G_Parameter (Max_Of_Parameter_Nb => Max_Of_Parameter_Nb);
73
74    package P_Command is new P_Parameter.G_Command (T_Command_Id => T_Fus_Command_Id);
75
76    ------------------------------------------------------------------------------
77
78    -- Check validity
79    function Is_Valid (Fu : T_Gfu) return Boolean is abstract;
80
81    ------------------------------------------------------------------------------
82
83    function Is_Valid (Fu      : T_Gfu;
84                       Command : P_Command.T_Command) return Boolean
```

Abstract class

Mvm.G_Fu | Insert | Writable | Unmodified | 49:1

Messages | Locations

GPS - (Mvm project) - mvm-g_fu.ads

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\MVM\src\MVM\generic\mvm-g_fu.ads

```
179    --*                                                                  *
180    --*          00000  00000  00000  0      0    0    00000  00000      *
181    --*          0   0  0   0    0    0      0   0 0     0    0          *
182    --*          00000  00000    0     0    0   0   0    0    0          *
183    --*          0      00       0     0    0   00000    0    0000       *
184    --*          0      0 0      0      0  0   0   0    0    0          *
185    --*          0      0  0     0      0  0   0   0    0    0          *
186    --*          0      0   0  00000     0     0   0    0    00000      *
187    --*                                                                  *
188    --***********************************************************************
189 private
190    -- ***********************************************************************
191
192    ----------------------------------------------------------------------
193
194    type T_Gfu is abstract tagged record
195       Current_State_Id : T_Fus_State_Id;
196       Status           : T_Status;
197    end record;
198
199    ----------------------------------------------------------------------
200
201    function Get_Status_Id (Fu : T_Gfu) return T_Fu_Status_Id is
202      (Fu.Status.Status_Id);
203
204    ----------------------------------------------------------------------
205
206    function Get_State_Id (Fu : T_Gfu) return T_Fus_State_Id is
207      (Fu.Current_State_Id);
208
209    ----------------------------------------------------------------------
210
```

Abstract class

Mvm.G_Fu Insert | Writable | Unmodified 49:1

Messages Locations

GPS - (Mvm project) - mvm-g_fu-fsm.ads

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\MVM\src\MVM\generic\mvm-g_fu-fsm.ads

```
66
67      --*****************************************************************
68      --*                                                               *
69      --*          00000    0    00000  0  0      0    00000  00000    *
70      --*          0   0   0 0   0      0 0      0 0   0      0        *
71      --*          00000  0   0  0      00      0   0  0      0        *
72      --*          0      00000  0      0 0     00000  0  00  0000     *
73      --*          0      0   0  0      0  0    0   0  0   0  0        *
74      --*          0      0   0  0      0   0   0   0  0   0  0        *
75      --*          0      0   0  00000  0    0  0   0  00000  00000    *
76      --*                                                               *
77      --*****************************************************************
78   package Mvm.G_Fu.Fsm is
79
80      ------------------------------------------------------------------
81
82      type T_Fu is abstract new T_Gfu with private;
83
84      ------------------------------------------------------------------
85
86      package P_Commands is new P_Command.G_Commands (First_Command => First_Command,
87                                                      Last_Command  => Last_Command);
88
89      subtype T_Parameter_Id          is P_Parameter.T_Parameter_Id;
90      subtype T_Command               is P_Command.T_Command;
91
92      package P_Ap_Command            is new P_Parameter.G_Command (T_Command_Id => T_Ap_Id);
93      package P_Ap_Commands           is new P_Ap_Command.G_Commands (First_Command => T_Ap_Id'First,
94                                                                      Last_Command  => T_Ap_Id'Last);
95      package P_Ap_Command_Call       is new P_Ap_Command.G_Call (T_Reference_Id => T_Parameter_Id);
96
```

Inheritance

Insert | Writable | Unmodified | 1:1

Messages | Locations

GPS - (Mvm project) - mvm-g_fu-fsm.ads

File Edit Navigate VCS Project Build Debug Tools Prove SPARK Window Help

default

\\muhome2\M023206\Offline Folder\HiLite\CASE_STUDY\TMTC\MVM\src\MVM\generic\mvm-g_fu-fsm.ads

```
367    --*          00000  00000    0     0    0   0   0    0    0                *
368    --*          0      00       0     0    0   00000    0    0000             *
369    --*          0      0 0      0      0  0   0   0    0    0                 *
370    --*          0      0  0     0      0  0   0   0    0    0                 *
371    --*          0      0   0  00000     0      0   0    0    00000            *
372    --*                                                                        *
373    --**************************************************************************
374 private
375    -- ************************************************************************
376
377    type T_Fu_Ap_Status (Status_Id : T_Fu_Ap_Status_Id := Ap_Fu_Successful) is
378       record
379          case Status_Id is
380          when Ap_Fu_Successful | Ap_Fu_Failed => null;
381          when Ap_Fu_In_Progress =>
382             Ap_Command : T_Ap_Command;
383             Step       : T_Fu_Ap_Step_Range;
384          end case;
385       end record;
386
387    type T_Fu is abstract new T_Gfu with record
388       Command_Types           : T_Command_Types;
389       Transitions             : T_Transitions;
390       Superstates_Definition  : T_Superstates_Definition;
391       Fu_Ap_Status            : T_Fu_Ap_Status;
392       Ap_Command_Types        : T_Ap_Command_Types;
393    end record;
394
395    ---------------------------------------------------------------------------
396
397    -- Check the validity of an automaton
```

Inheritance

Insert | Writable | Unmodified | 1:1

Messages | Locations

```
procedure Run_Plans (Obcpe : in out T_Obcpe) is
  Execute_Next_Command : Boolean;
  ---------------------------------------------------------------------------
  procedure Step_Further (Plan_Id   : T_Plan_Id);
  procedure Step_Further (Plan_Id   : T_Plan_Id) is
  begin
    if Obcpe.Plans_Status (Plan_Id).Current_Step < Obcpe.Plans (Plan_Id).Steps_Nb then
      Obcpe.Plans_Status (Plan_Id).Current_Step := T_Step_Range'Succ (Obcpe.Plans_Status (Plan_Id).Current_Step);
    else
      Obcpe.Plans_Status (Obcpe.Active_Plan) := T_Plan_Status'(Status_Id    => Plan_Failed,
                                                               Plan_Failure => End_Of_Plan_Reached,
                                                               Failure_Step => Obcpe.Plans_Status (Obcpe.Active_Plan).Current_Step);
    end if;
    Execute_Next_Command := True;
  end Step_Further;
  ---------------------------------------------------------------------------
begin
  Mvm.Obit.Run_Time (Current_Time => Obcpe.Current_Time);
  if (case Obcpe.Obcpe_Status is
      when Obcpe_On                => True,
      when Obcpe_Off | Obcpe_Failed => False) then
    P_Events.Check_Monitoring (Events       => Obcpe.Events,
                               Current_Time => Obcpe.Current_Time);
    loop
      P_Vaps.Run_Vaps (Obcpe.Vaps, Obcpe.Current_Time);
      Execute_Next_Command := False;
      case Obcpe.Plans_Status (Obcpe.Active_Plan).Status_Id is
      when Plan_Inactive =>
        pragma Assert (False, "An Inactive Plan can not be executed");
      when Plan_Ready =>
        pragma Assert (False, "A Ready Plan can not be executed");
      when Plan_Terminated =>
        pragma Assert (False, "A Terminated Plan can not be executed");
      when Plan_Failed =>
        declare
          Alternative_Plan : T_Plan_Id;
        begin
          Alternative_Plan := Obcpe.Plans (Obcpe.Active_Plan).Alternative_Plan;
          if Obcpe.Plans_Status (Alternative_Plan).Status_Id = Plan_Ready then
            Obcpe.Active_Plan := Alternative_Plan;
            Obcpe.Plans_Status (Alternative_Plan) := T_Plan_Status'(Status_Id    => Plan_Running,
                                                                    Current_Step => T_Step_Range'First);
          else
            Obcpe.Obcpe_Status := Obcpe_Failed;
          end if;
          Execute_Next_Command := True;
        end;
      when Plan_Running =>
        declare
          Current_Step : T_Step_Range;
          Instruction  : T_Instruction;
        begin
          Current_Step := Obcpe.Plans_Status (Obcpe.Active_Plan).Current_Step;
          Instruction := Obcpe.Plans (Obcpe.Active_Plan).Steps (Current_Step);
          case Instruction.Instruction_Id is
            ---------------------------------------------------------------------
            -- Status: Wait for the end of the execution
          when Send_Cmd_To_Fu_Instruction | Protected_Send_Cmd_To_Fu_Instruction =>
            declare
              Fu_Command   : T_Fu_Command;
            begin
              -- A FU command call is created
              Fu_Command := P_Fus.P_Gfu.P_Command.Create_Command (Command_Id    => Instruction.Fu_Command_Call.Command_Id,
                                                                   Parameters_Nb => Instruction.Fu_Command_Call.Parameters_Nb);
              -- Update the referenced parameters
              if Fu_Command.Parameters_Nb > 0 then
                -- Update the referenced parameters
                for Parameter_Id in T_Parameter_Id range T_Parameter_Id'First .. T_Parameter_Id'First + Fu_Command.Parameters_Nb - 1 loop
                  case Instruction.Fu_Command_Call.Actuals (Parameter_Id).Actual_Type is
                  when Numerical =>
                    -- The numerical value shall be copied
                    Fu_Command.Parameters (Parameter_Id) := Instruction.Fu_Command_Call.Actuals (Parameter_Id).Value;
                  when Reference =>
                    -- The referenced of the referenced parameter
```

Too big subprogram

This document and its content is the property of Astrium [Ltd/SAS/GmbH] and is strictly confidential. It shall not be communicated to any third party without the written consent of Astrium [Ltd/SAS/GmbH].

Results

This document and its content is the property of Astrium [Ltd/SAS/GmbH] and is strictly confidential. It shall not be communicated to any third party without the written consent of Astrium [Ltd/SAS/GmbH].

See PDF

Overview

- Introduction
- Astrium Space Transportation Case study
  - Numerical algorithm
  - Event driven
- Conclusion

This document and its content is the property of Astrium [Ltd/SAS/GmbH] and is strictly confidential. It shall not be communicated to any third party without the written consent of Astrium [Ltd/SAS/GmbH].

Conclusion

- Compared to SPARK 2005
  - Great improvement
  - Larger perimeter of Ada
  - “Executable contracts” is a major progress
  - ➔ The ROI of contracts writing is positive, even without formal proof
- Operationally deployable?
  - The use of proof still requires a high level of expertise!
  - ➔ Proof may be costly
  - Proof is not 100% sound! (float versus real)
  - Final version of SPARK 2014 is promising

This document and its content is the property of Astrium [Ltd/SAS/GmbH] and is strictly confidential. It shall not be communicated to any third party without the written consent of Astrium [Ltd/SAS/GmbH].