Simulation Smackdown Environment Federate

Edwin Z. Crues

Simulation and Graphics Branch
NASA Johnson Space Center

February, 2013

Outline

- What does the Environment Federate Do?
- Getting the Environment Federate
- Building the Environment Federate
- Running the Environment Federate

What does the Environment Federate Do?

- Manages time for the federation execution,
- Defines the physical time standard and epoch of the federation execution, and
- Publishes the position and orientation of key planetary reference frames.

What does the Environment Federate Do?

Time Management

- The environment federate uses HLA time management APIs to regulate the progression of federation execution time when the federation is running:

  Time Constrained Waits for time regulating federates to advance time.

  Time Regulating Coordinates with other time regulating federates to advance time.

- Controls the advancement of time to correspond to standard computer or "wall clock" time.

What does the Environment Federate Do?

Physical Time

- One of the principal responsibilities for the Environment federate it to provide a "Universal" representation of physical time for the federation execution.
  - This is not the same as federation execution time.
  - This is not the same as simulation execution time.
- The time standard for the 2013 SISO/SCS Simulation Smackdown federation is Terrestrial Time (TT).
- The Environment federate starts execution on a predetermined modeled environment time/date (simulation epoch).
- Published time stamps correspond to seconds since the ordinal epoch of the TT time standard.

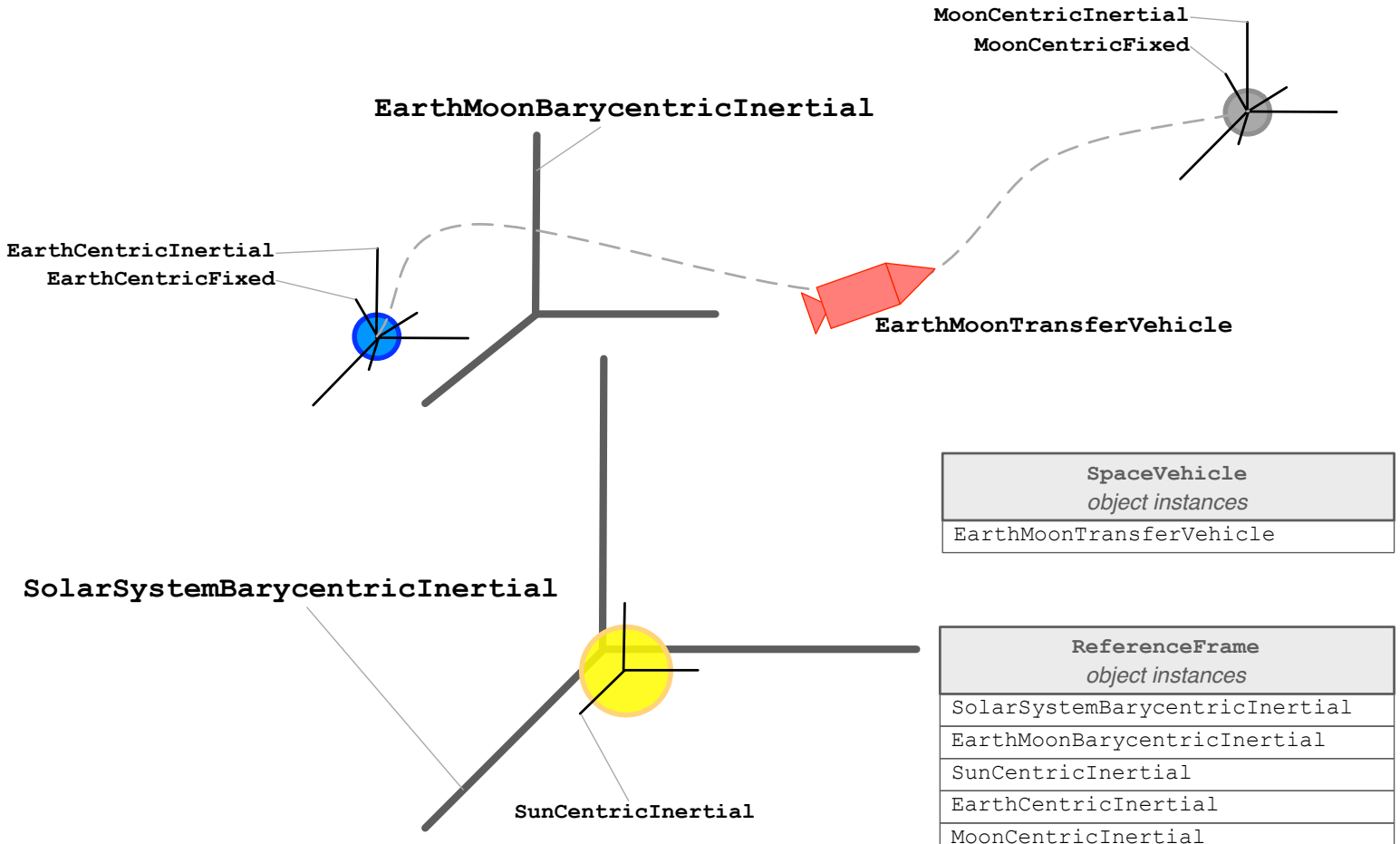
What does the Environment Federate Do?

Reference Frames

- Another of the Environment federate's principal responsibilities is to publish the state of key planetary reference frames.
- For the 2013 Smackdown, the Environment federate will be publishing the following frames: `SunCentricInertial`, `EarthMoonBarycenterInertial`, `EarthMoonBarycenterRotating`, `EarthCentricInertial`, `EarthCentricFixed`, `MoonCentricInertial`, `MoonCentricFixed`, `EarthMoonL2Rotating`, `MarsCentricInertial`, and `MarsCentricFixed`.

What does the Environment Federate Do?

Reference Frames *(continued)*



SpaceVehicle *object instances*
EarthMoonTransferVehicle

ReferenceFrame *object instances*
SolarSystemBarycentricInertial
EarthMoonBarycentricInertial
SunCentricInertial
EarthCentricInertial
MoonCentricInertial

Figure : Environment federate reference frames

What does the Environment Federate Do?

Reference Frames *(continued)*

What defines a reference frame?

Field	Type	Description
name	`HLAunicodeString`	A unique name for this reference frame instance. Reference frame names are essential in forming 'links' between parent/child reference frames.
parent_name	`HLAunicodeString`	The name of this frame's parent reference frame. If this frame has no parent (i.e., is a 'root' reference frame), then this string must be empty, otherwise the non-empty string must correspond to the name attribute of some other ReferenceFrame object instance in the simulation.
translational_state	`ReferenceFrameTranslation`	This reference frame's translational state with respect to its parent frame. If this frame has no parent, this attribute is meaningless.
rotational_state	`ReferenceFrameRotation`	This reference frame's rotational state with respect to its parent frame. If this frame has no parent, this attribute is meaningless.
time	`Time`	This value serves as a 'time stamp' that specifies the simulated time (TT) to which the attributes values correspond. It may be used by federates that do not use HLA time management but still need to know when the attributes were valid. (E.g., a plotting federate that isn't time regulating or time constrained would need the time stamp in order to plot time series.)

What does the Environment Federate Do?

Reference Frames *(continued)*

What defines the translational state?

Field	Type	Description
position	`PostionVector`	Position of the subject frame origin with respect to the referent origin with components resolved onto the subject coordinate axes.
velocity	`VelocityVector`	Velocity of the subject frame origin with respect to its referent origin with components resolved onto the subject coordinate axes.

What defines the rotational state?

Field	Type	Description
attitude_quaternion	`AttitudeQuaternion`	Attitude quaternion that specifies the orientation of the subject frame with respect to the referent.
angular_velocity	`AngularVelocityVector`	Angular velocity of the subject frame with respect to the referent with components resolved onto the subject coordinate axes.

Getting the Environment Federate

You can obtain the Environment federate code in its entirety from the Simulation Smackdown Assemble Subversion repository.

1. Register for an account at the Assembla website (`https://www.assembla.com/user/signup`)
2. Contact the Smackdown Assembla repository custodian (`edwin.z.crues@nasa.gov`) and provide your Assembla account name.
3. Once the custodian adds you to the Smackdown team, you can check out the Environment federate from the repository:

   (`https://subversion.assembla.com/svn/SISO_Smackdown/trunk/2013`).

Bulding the Environment Federate

Once you've obtained the Environment federate Java code, you can build it with either the provided `makefile` or the provided Ant `build.xml` script.

To build, you will need the following:

- An installed Java Development Kit (JDK).
- An installed HLA 1516-2010 Run Time Infrastructure (RTI) Java JAR file.
- A link file named `rti1516e.jar` in the `lib` directory that points to your vendor's RTI JAR file.
- Know how to configure either the `make` or `ant` build systems to find these.
- Run either `make` or `ant` to build the Environment federate JAR file.

A successful build should result in an `Environment.jar` file in the `lib` directory.

Running the Environment Federate

Once you've built the Environment federate, you should be ready to run it.

You will need to consult your RTI vendor's documentation to determine how best to configure your environment to run a Java based application.

There are two C-shell scripts that can be used to run the Environment federate JAR file applications:

- run_env — This script allows you to run the Environment federate with additional arguments.
- run_test — This script allows you to run the Environment Test federate with additional arguments.

Running the Environment Federate

Help Output

You can run the Environment federate with the `-h` option (`./run_env -h`) to get the following help message:

```
*** Simulation Smackdown Environment Federate ***

usage: Environment [{-h,--help}]
                   [{-v,--verbose}]
                   [{-d,--date} <MM/dd/yyyy HH:mm:ss zzz>]
                   [{-j,--JD} UTC_Julian_date]
                   [{-m,--MJD} UTC_modified_Julian_date]
                   [{-t,--TJD} UTC_truncated_Julian_date]
                   [{-r,--run_time} seconds]
                   [{-f,--hla}]
                   [{-n,--name} federate_name]
                   [{--crc_host} CRC_host_name]
                   [{--crc_port} CRC_port_number]
```

Running the Environment Federate

Terminal Output

When you run the Environment federate (`./run_env`), you should see something like the following output:

```
*** Simulation Smackdown Environment Federate ***

Ephemeris file located in:
   /opt/smackdown/2013/federates/Environment/jat/data/core/ephemeris/DE405data/

************************************************
RTI Name: pRTI 1516
RTI Version: v4.4.2
HLA Version: null
Federate "Simulation Smackdown Environment": Cannot advance to current time!
   This may be the first time regulating federate!

/opt/smackdown/2013/federates/Environment/jat/data//core/spacetime
************************************************
CRC host: localhost
CRC port: 8989
************************************************
Simulation Epoch: 2013,4,10 20:0:1.341104507446289E-5
Julian date: 2456393.3333333335
Truncated Julian date: 16392.83333333349
************************************************

Executive Loop Counter: 39
```

Running the Environment Federate

Pitch RTI CRC Interface

Here's what it should look like in the Pitch RTI CRC Interface:

Figure : Environment federate in RTI Interface

Running the Environment Federate

Running the Environment Test

When you run the Environment Test federate (`./run_test`), you should see something like the following output:

```
*** Java Environment Test Federate ***

************************************************
RTI Name: pRTI 1516
RTI Version: v4.4.2
HLA Version: null
************************************************
CRC host: localhost
CRC port: 8989
************************************************
EnvironmentTest "Environment Test": Starting time (epoch): 1.4163410391840134E9
************************************************
Times are:
   Executive Loop Counter: 0
   Simulation Execution Time: 0.0
   EnvironmentTest Physical Time: 1.4163410391840134E9
   Federation Execution Time (s): 175.0

Times are:
   Executive Loop Counter: 1
   Simulation Execution Time: 1.0
   EnvironmentTest Physical Time: 1.4163410401840134E9
   Federation Execution Time (s): 176.0
```

Running the Environment Federate

Pitch RTI CRC Interface

Here's what it should look like in the Pitch RTI CRC Interface:

Figure : Environment Test federate in RTI Interface

Questions?