



SISO-REF-067-2017

FEAT User's Guide Reference for the Federation Engineering Agreements Template

23 January 2017

**Prepared by:
Simulation Interoperability Standards Organization
Federation Engineering Agreements Template (FEAT)
Product Support Group (PSG)**

SISO-REF-067-2017, FEAT User's Guide

Copyright © 2017 by the Simulation Interoperability Standards Organization, Inc.

P.O. Box 781238
Orlando, FL 32878-1238, USA

All rights reserved.

Permission is hereby granted to quote any of the material herein, or to make copies thereof, for non-commercial purposes, as long as the proper attribution is made and this copyright notice is included. All other uses are prohibited without written permission from the SISO Inc. Board of Directors.

SISO Inc. Board of Directors
P.O. Box 781238
Orlando, FL 32878-1238, USA

Revision History

| Version | Section | Date (MM/DD/YYYY) | Description |
|-------------------|----------------|--------------------------|--------------------------|
| SISO-REF-067-2017 | All | 1/23/2017 | 2017 Initial publication |

TABLE OF CONTENTS

| | | |
|-----------|---|-----------|
| 1 | Introduction | 6 |
| 1.1 | Purpose..... | 6 |
| 1.2 | Purpose..... | 6 |
| 1.3 | Objectives..... | 6 |
| 2 | References | 6 |
| 3 | Definitions | 6 |
| 4 | Acronyms and abbreviations | 6 |
| 5 | FEAT overview | 7 |
| 5.1 | Programmer's Reference..... | 8 |
| 5.2 | FEAT Editor | 9 |
| 5.3 | Federation Agreements in Simulation Engineering Processes | 10 |
| 6 | Design Patterns | 10 |
| 6.1 | Individual Agreement Status Indicators | 10 |
| 6.2 | Reuse of Existing Schemas..... | 11 |
| 6.3 | Documents..... | 11 |
| 7 | Agreement Categories | 11 |
| 7.1 | Metadata | 11 |
| 7.2 | Design..... | 12 |
| 7.3 | Execution..... | 13 |
| 7.4 | Management | 13 |
| 7.5 | Data | 14 |
| 7.6 | Infrastructure..... | 15 |
| 7.7 | Modeling | 15 |
| 7.8 | Variances | 16 |
| 8 | The Fuel Economy Federation | 16 |
| 9 | Example Excerpts | 18 |
| 9.1 | Metadata | 18 |
| 9.2 | Design..... | 19 |
| 9.3 | Execution..... | 20 |
| 9.4 | Management | 21 |
| 9.5 | Data | 22 |
| 9.6 | Infrastructure..... | 23 |
| 9.7 | Modeling | 25 |
| 9.8 | Variances | 25 |
| 10 | Future Tool Support | 26 |

LIST OF FIGURES

| | | |
|-----------|--|----|
| Figure 1: | Top-Level View of the FEAT Schema | 8 |
| Figure 2: | Programmer's Reference | 9 |
| Figure 3: | FEAT Editor | 9 |
| Figure 4: | HTML Output Example..... | 10 |
| Figure 5: | Fuel Economy Federation FOM and Conceptual Model | 17 |
| Figure 6: | Example FEF Execution | 17 |
| Figure 7: | Network Layout..... | 18 |
| Figure 8: | Address Book Example | 18 |

Figure 9: Tools Example19
Figure 10: Execution States Example20
Figure 11: Execution State HTML21
Figure 12: Configuration Management Example22
Figure 13: DEM Example.....23
Figure 14: Hardware Configuration Example24
Figure 15: Hardware Configuration HTML.....24
Figure 16: Miscellaneous Modeling Example25
Figure 17: Miscellaneous Modeling HTML25
Figure 18: Variances Example.....26

LIST OF TABLES

Table 1: Metadata Agreements11
Table 2: Design Agreements12
Table 3: Execution Agreements13
Table 4: Management Agreements14
Table 5: Data Agreements.....14
Table 6: Infrastructure Agreements15
Table 7: Modeling Agreements.....15
Table 8: Variances16
Table 9: Address Book XML18
Table 10: Tools XML.....19
Table 11: Configuration Management XML.....22
Table 12: DEM XML23

1 INTRODUCTION

Federation agreements are critical to the successful design, execution, and reuse of federation assets. In the past, inconsistent formats and use across federations have made it difficult to capture and compare agreements between federations. This lack of a consistent approach to documenting federation agreements made reuse and understanding more difficult. It also prevented tool development and automation.

1.1 Purpose

The Federation Engineering Agreements Template (FEAT) provides a standardized format for recording federation agreements to increase their usability and reuse. The template is an eXtensible Markup Language (XML) schema from which compliant XML-based federation agreement documents can be created. XML was chosen for encoding agreements documents because it is both human and machine-readable and has wide tool support. Creating the template as an XML schema allows XML-enabled tools to both validate conformant documents, and edit and exchange agreements documents without introducing incompatibilities.

The FEAT benefits all developers, managers, and users of distributed simulations by providing an unambiguous format for recording agreements about the design and use of the distributed simulation. The FEAT also benefits this community by enabling the development of federation engineering tools that can read the XML schema and perform federation engineering tasks automatically.

XML schemata and documents can be viewed using many tools:

- Text editors
- Web browsers
- Word and other word processing applications
- XML editors

1.2 Purpose

The FEAT User's Guide provides users with an introduction to the structure and use of the FEAT, and illustrates these with examples.

1.3 Objectives

The objective of this document is to provide potential users with more guidance on the application of FEAT than can be gleaned from reviewing just the schema itself.

2 REFERENCES

SISO Documents

| Document Number | Title |
|-------------------|--|
| SISO-STD-012-2013 | Standard for Federation Engineering Agreements Template (FEAT) |

Other Documents

| Document Number | Title |
|-----------------|-------|
| | |

3 DEFINITIONS

Simulation Environment Agreements: The architecture-neutral term for federation engineering agreements adopted in the DSEEP [B3].

4 ACRONYMS AND ABBREVIATIONS

| | |
|------|---|
| C4I | Command, Control, Communications, Computers, and Intelligence |
| CM | Configuration Management |
| COI | Community of Interest |
| CPE | Common Platform Enumeration |
| DEM | Data Exchange Model |
| DMAO | DSEEP Multi-Architecture Overlay |

| | |
|---------|--|
| DMS | Discovery Metadata Specification |
| DSEEP | Distributed Simulation Engineering and Execution Process |
| FEAT | Federation Engineering Agreements Template |
| FEF | Fuel Economy Federation |
| FOM | Federation Object Model |
| GML | Geography Markup Language |
| GUI | Graphical User Interface |
| HLA | High Level Architecture |
| HTML | Hypertext Markup Language |
| HWIL | Hardware in the Loop |
| IC-ISM | Intelligence Community Information Security Marking |
| JHU/APL | The Johns Hopkins University Applied Physics Laboratory |
| LAN | Local Area Network |
| LROM | Logical Range Object Model |
| LVC | Live-Virtual-Constructive |
| M&S | Modeling and Simulation |
| OS | Operating System |
| PDG | Product Development Group |
| PDU | Protocol Data Unit |
| PSG | Product Support Group |
| SISO | Simulation Interoperability Standards Organization |
| TDB | To be determined |
| UML | Unified Markup Language |
| US | United States |
| VLAN | Virtual LAN |
| WAN | Wide Area Network |
| XLink | XML Linking Language |
| XMI | XML Metadata Interchange |
| XML | eXtensible Markup Language |

5 FEAT OVERVIEW

The view of the FEAT schema in Figure 1 was generated using Oxygen, but there are many XML editors.

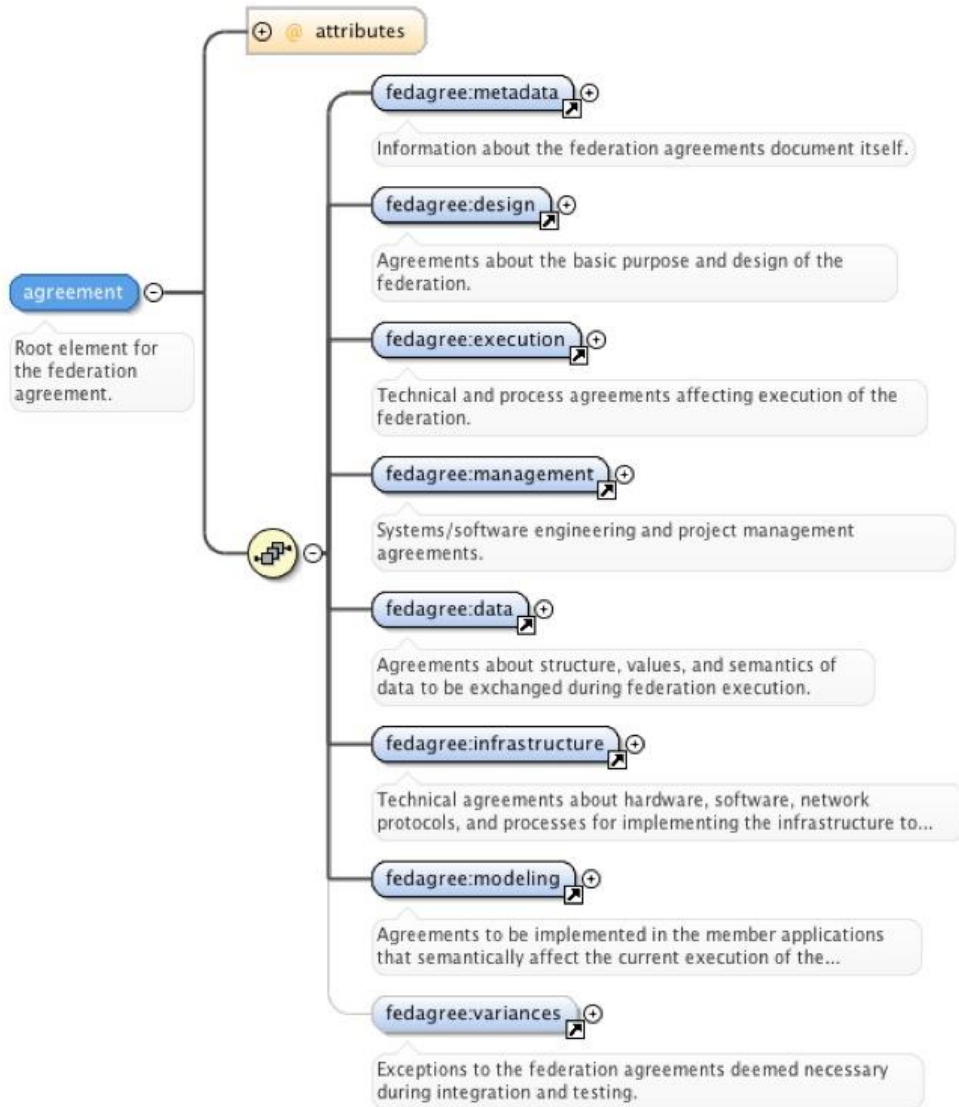


Figure 1: Top-Level View of the FEAT Schema

5.1 Programmer's Reference

The FEAT standard is maintained by the Simulation Interoperability Standards Organization's (SISO) FEAT Product Support Group (PSG) and Product Development Group¹ (PDG). An interactive version of the standard, the programmer's reference [B1], is maintained on the SISO website. Figure 2 shows its top level view.

¹ The PSG identifies potential changes to the latest, balloted version of the standard and answers users' question about the standard. The PDG is solely responsible for updating the standard at the appropriate time.

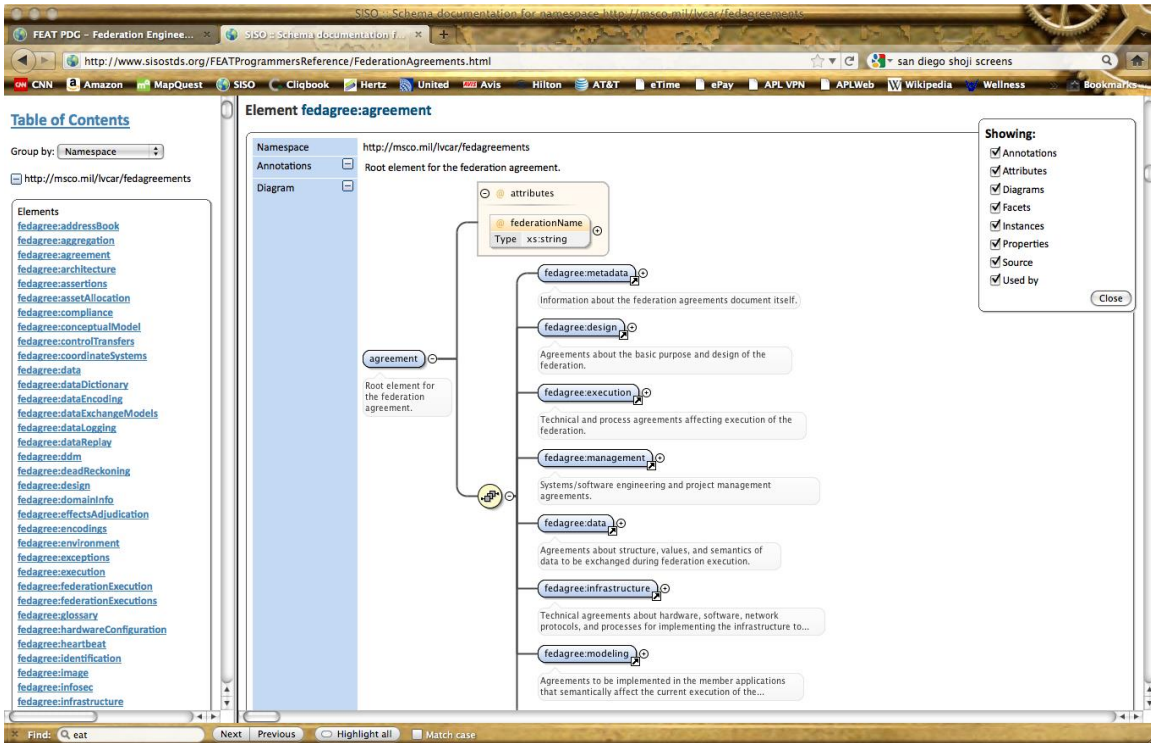


Figure 2: Programmer's Reference

5.2 FEAT Editor

The Johns Hopkins University Applied Physics Laboratory (JHU/APL) developed a basic editor tool for creating FEAT documents through a graphical user interface (GUI) rather than through an XML editor. Figure 3 is a screenshot of the FEAT Editor. The code and executable for the FEAT Editor are available on SourceForge [B2].

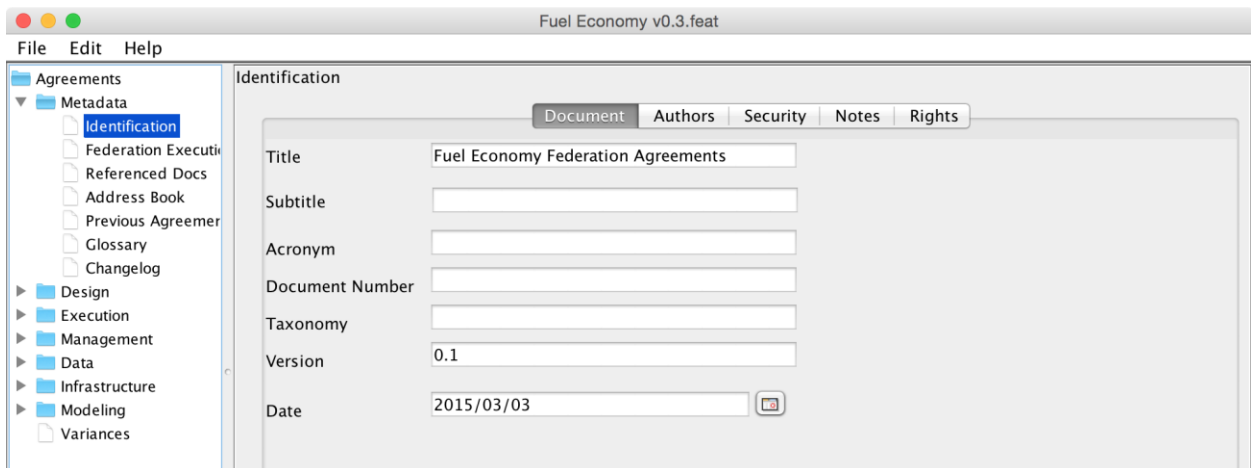


Figure 3: FEAT Editor

The FEAT Editor prints FEAT files as Hypertext Markup Language (HTML) that can be displayed and printed from a web browser, or converted to another printable format. Figure 4 shows the output associated with content illustrated in Figure 3.

FEAT Description: Fuel Economy Federation Agreements

Status:
 Subtitle:
 Acronym:
 Document Number:
 Taxonomy:
 Security Classification:

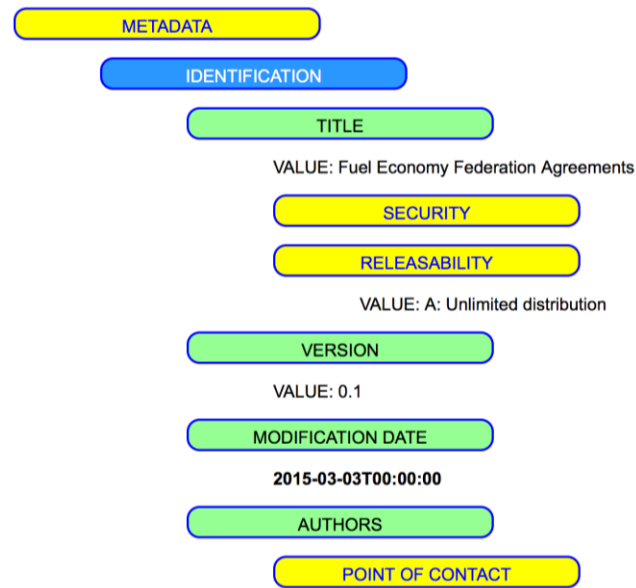


Figure 4: HTML Output Example

5.3 Federation Agreements in Simulation Engineering Processes

The Distributed Simulation Engineering and Execution Process (DSEEP) [B3] describes a generalized process for building and executing distributed simulation environments. The need for federation agreements (simulation environment agreements) is identified in activity 4.2 - Establish simulation environment agreements. The DSEEP identifies a handful of specific agreements. The FEAT takes a broader view, providing the capability to record agreements made across the entire federation engineering lifecycle.

Although the DSEEP provides the guidance required to build and execute a distributed simulation environment, the implicit assumption within the DSEEP is that a single simulation architecture is being used. The DSEEP Multi-Architecture Overlay (DMAO) [B4] expressly identifies and addresses issues that arise in multi-architecture, live-virtual-constructive (LVC) environments. Activity 4.2 of the DMAO also addresses simulation environment agreements.

6 DESIGN PATTERNS

Design patterns used throughout the FEAT schema are described in this section.

6.1 Individual Agreement Status Indicators

Federation agreements are a work in progress throughout federation development and execution. Individual agreements may be in very different stages of completion, which may not be obvious just from looking at the agreements themselves. Individual agreements each have a status indicator that may have one of the following values:

- To Be Determined (TBD)
- Negotiated
- Designed
- Implemented
- Tested

- Validated

6.2 Reuse of Existing Schemas

Wherever possible, FEAT leverages existing, authoritative schemas for the representation of elements in this schema as described below:

- XML Linking Language (XLink) [B7]
 - Hyperlinks
- XML Metadata Interchange (XMI) [B8]
 - Unified Modeling Language (UML) diagrams
- Common Platform Enumeration (CPE) [B9]
 - Definition of specific software platforms (e.g., operating system with major and minor version, tools, etc.)
- Geography Markup Language (GML) [B10]
 - All geography or rigorous timing requirements, including bounding boxes, coordinates, and timing constraints

6.3 Documents

Because a federation may be large and complex, it may have entire documents for individual agreements. For example, if the federation is fairly small, the conceptual model might be captured in a few elements, including entities, event traces/sequence diagrams, and state transition diagrams/entity behaviors that leverage the first two elements. For maximum flexibility, FEAT allows for either inline capture of the conceptual model or a detailed reference to an external document. This design pattern appears in a majority of agreements. Where elements describe processes or sequences within the federation, agreements may be represented in Unified Modeling Language (UML) that may be rendered in the schema as XML and/or a graphic. Tools, member applications, hardware, and physical locations are cross-referenced in several agreements but are listed and described only once.

7 AGREEMENT CATEGORIES

Each of the following subsections identifies and describes one of the categories of federation agreements and lists the individual agreements under that category.

7.1 Metadata

Metadata is information about the FEAT document itself.

Table 1: Metadata Agreements

| | |
|------------------------------|---|
| Identification | Identifies this document; version number and name if applicable. Makes extensive use of the MSC-DMS metadata elements, making this the most important element relative to registering the federation agreements document in repositories based upon MSC-DMS markup. |
| Federation Executions | Defines the federation execution(s) to which the document applies. |
| Referenced Docs | Location and version of documents referenced in other agreements. |
| Address Book | People and organizations used as points of contact in other agreements. Any POCs listed throughout the rest of the schema must have an associated address book entry even though the addressBook itself is an optional element. |
| Previous Agreements | Prior agreements used for reference during simulation design and development. These may be reused or modified as necessary. |

| | |
|------------------|---|
| Glossary | Terminology, acronyms, and definitions. |
| Changelog | A log of changes to the federation agreements document. |

7.2 Design

Design agreements are about the basic purpose and design of the federation.

Table 2: Design Agreements

| | |
|--|--|
| Conceptual model | Element for inclusion of a conceptual model that includes entities within the domain of interest, static and dynamic relationships among entities, and entity behaviors. Can include 0 or more references to remote documents, 0 or more inline XMI-encoded models, and 0 or more BOM-encoded models. Federations with detailed conceptual models may choose to include such models by reference using document number, title, version, and reference. Alternatively, a minimal conceptual model may be documented using the combination of the entities, event traces/sequence diagrams, and state transition diagrams/entity behaviors within this document. |
| Information security | Information security and information assurance procedures and policies applicable to the environment; level at which the federation will be executed; compliance with DISR standards. Intelligence Community Information Security Marking (IC-ISM) attributes are used to inline reuse restriction information. Links to external documents containing the infosec information and procedures may substitute for or supplement any or all elements and attributes. For federations with detailed security requirements, a reference to a separate security plan or policy document is appropriate here. Alternatively, unclassified federations may choose to document applicable IT security standards and processes here. Applies to the simulation environment and execution, not to the agreements document or its constituent agreements. |
| Terrain | Playbox, format, resolution, correlation/transposition. Geography Markup Language (GML) is used to encode geographical bounding boxes and conversion and transformation options. |
| Federation objectives/ Requirements | Measurable objectives and requirements for the federation. |
| Environment | References to environment representation-specific objectives and requirements. For federations with detailed environment representation requirements, a reference to a separate document is appropriate here. |
| Scenario | Scenario description including initial laydown; reference to scenario files, e.g. MSDL; scenario file format. |
| Federation architecture | Describes the top level architecture of the federation. For federations with detailed federation architectures, a reference to a separate document is appropriate here. Simpler federations may just insert a graphic or UML diagram of the architecture. |
| Member applications | Type, name, version, description, responsibilities of each member application. For federations with detailed federation architectures, a reference to a separate document is appropriate here, possibly the |

| | |
|---------------------------|--|
| | same document as the federation architecture. Simpler federations may just identify the member applications here. |
| Tools | Tools used in the design and support of the simulation. This section should include any relevant operating systems, database engines, etc. explicitly required by the federation in addition to simulation-specific tools. |
| Domain information | Information traceable to authoritative sources that is used to build supporting databases and algorithms for member applications and the simulation environment. |

7.3 Execution

Technical and process agreements affecting execution of the federation are recorded in execution agreements.

Table 3: Execution Agreements

| | |
|--------------------------------------|---|
| Execution states | A list of changes in the overall state of the federation execution, e.g. initialization, saving, shutdown. |
| Time management | How simulation time will be advanced. Particularly important if any part of the event is time-managed (i.e., simulation time not necessarily scaled to wallclock time.) This is related to event ordering and repeatability. |
| Join/Resign | Preconditions and procedures, including ordering, for joining and resigning the federation under normal conditions |
| Update rates | Agreements on how often federates agree to update states; may be static or dynamic. Might give upper/lower limits or set rates. Rates may be set for the entire federation, federates (member applications), or object classes. |
| Performance thresholds | Agreements on federate thresholds, such as maximum entity or contact counts, indicating how federates will respond when thresholds are exceeded. |
| Data logging | Responsibilities for logging and replaying data, including uploading and configuration management. |
| Data Replay | Responsibilities and requirements for replaying data during an execution, including configuration management. |
| Monitoring | Tools and procedures for monitoring federation execution. |
| Middleware service agreements | Identification of which middleware services may be used, and under what conditions. |
| Member configuration | Configuration options for a member application. |

7.4 Management

Management agreements include systems/software engineering and project management agreements.

Table 4: Management Agreements

| | |
|--|---|
| Software development/ Configuration management/ Version control processes | Tool identification and references to plan(s); includes methodology and guidelines for establishing and managing Configuration Management (CM) baselines of member applications and any middleware expressly under development for the federation; identification of standard software practices to support lifecycle development and reuse. Federations that utilize formal software management processes and procedures may reference these artifacts using document citation and/or reference. Alternatively, the federation agreements document may outline applicable processes and procedures in the body of the test with supporting graphics. |
| VV&A | VV&A tool identification, and references to plan(s) and other documents including formal certification of compliance with simulation environment architecture interface specification and rules. The latter may include documentation of agreements pertaining to architecture compliance and artifacts demonstrating that compliance has been achieved. |
| Schedules | Reference to program schedule(s). Provides information about scheduled activities as part of a detailed event plan per DSEEP or applicable process/overlay. |
| Test plan | Test Tool identification and references to plan(s); test requirements and criteria. Planned activities to perform testing of M&S at increasingly higher levels of integration prior to LVC event execution. Test planning processes and artifacts may address M& S verification requirements to be addressed prior to event execution; explicit traceability will be addressed as part of these agreements. |
| Miscellaneous processes | Processes not captured by one of the more specific sibling agreements. |

7.5 Data

Agreements about structure, values, and semantics of data to be exchanged during federation execution are recorded in data agreements.

Table 5: Data Agreements

| | |
|--|---|
| Data encoding | Data encodings used in the federation, including acceptable and unacceptable representation of "no current value" by data type and endianness. |
| Data exchange models | FOM, LROM, PDU utilization and tailoring, direct interactions (e.g., remote methods) |
| Naming conventions | Naming conventions for objects, attributes/parameters, enumerations, and member applications |
| Publication and subscription responsibilities | Allocation of responsibility for data distribution among member applications, subject to the simulation environment architecture |
| Data dictionary elements | Identification of standard data dictionary references used in the Simulation Data Exchange Model and associated Simulation Environment Agreements |

| | |
|-----------------------------|---|
| Supporting databases | Common and consistent data used across the simulation environment to guarantee valid interactions among all member applications. This element contains information on how to access common and consistent data used across the simulation environment to guarantee valid interactions among all member applications. This data may be stored in a relational database, in which case the 'dbEngineRef' attribute should refer to a previously identified database engine, or some other location (e.g., web service). The 'accessInterfaces' and 'accessRestrictions' child elements should be utilized for connection details. |
| User-supplied data | This element is for data supplied by a user of the simulation that may prove useful to other users, but which is not part of the general federation data. This information includes tag values, entity identifiers, site identifiers, host identifiers, or any other event-, user- or site-specific data not captured in the other structured data agreements. |

7.6 Infrastructure

Infrastructure agreements are technical agreements about hardware, software, network protocols, and processes for implementing the infrastructure to support federation execution.

Table 6: Infrastructure Agreements

| | |
|---|--|
| Networking | Networking layout, network protocols used, WAN, LAN, VLAN, ports used |
| Physical security (facility) | Physical security requirements and procedures for the facility(ies) in which execution takes place: visit requests, restrictions due to nationality (e.g. non-US observing US exercises). |
| Middleware | Software that connects or integrates other software modules or components, typically providing a set of communications or interaction functions that may be invoked by the linked modules. |
| Secondary communication channels | Non-data exchange model data, e.g. video or chat; live entity, C4I protocols, and HWIL communication protocols. |
| Asset allocation | Allocation of simulation systems, member applications, and live systems to specific hardware. |
| Hardware configuration | Specifies hardware to which software (including middleware) must be deployed and hardware configuration including OS versions. |
| Locations | Identification of sites where federation components will be instantiated. |

7.7 Modeling

Agreements to be implemented in the member applications that semantically affect the current execution of the federation are recorded in modeling agreements.

Table 7: Modeling Agreements

| | |
|-----------------------------|--|
| Effects adjudication | Effects adjudication agreements ensure a 'fair fight' by specifying what component has the authority to determine the outcome or |
|-----------------------------|--|

| | |
|------------------------------------|--|
| | effect of an interaction between member applications, e.g. "shooter" (or the one initiating the effect) adjudicates, 'target' (the one the effect is perpetrated on) and, 'server' (some 3rd party member application). This list should not be considered exhaustive. |
| Coordinate systems | Reference to authoritative coordinate system representations. Included coordinate systems must be able to substitute for gml:AbstractCoordinateSystem. |
| Dead reckoning | Dead reckoning algorithms to be used. |
| Aggregation/ Disaggregation | Triggers and algorithms for aggregation and disaggregation. |
| DDM/Interest management | Parameters and algorithms for interest management. |
| Control transfers | Triggers, constraints, and algorithms for transferring control of entities between member applications. |
| Modeling issues | Federation-specific modeling issues such as ground clamping, resupply/refuel. |
| Miscellaneous | Catch-all for modeling agreements that cannot be easily categorized. As an example, notes would be included here about treating weekends and holidays different than weekdays or allowing for regular work hours during weekdays. |

7.8 Variances

Variances are used to record exceptions to the federation agreements deemed necessary during integration and testing. If the federation is reused, the variances in the previous FEAT document should be reviewed to identify errors from the previous federation.

Table 8: Variances

| | |
|------------------|--|
| Variances | Exceptions to the federation agreements deemed necessary during integration and testing. |
|------------------|--|

8 THE FUEL ECONOMY FEDERATION

The FEAT PSG has developed an openly available example [B5] based on Pitch's High Level Architecture (HLA) Tutorial² [B6]. Figure 5 provides a high-level, graphical summary of the Fuel Economy Federation (FEF) in the HLA Tutorial including the object class and interaction class hierarchies from the Federation Object Model (FOM), and a sequence diagram illustrating the conceptual model of the federation.

² Pitch holds the copyright to the tutorial but has generously granted permission for reuse to the FEAT PSG.

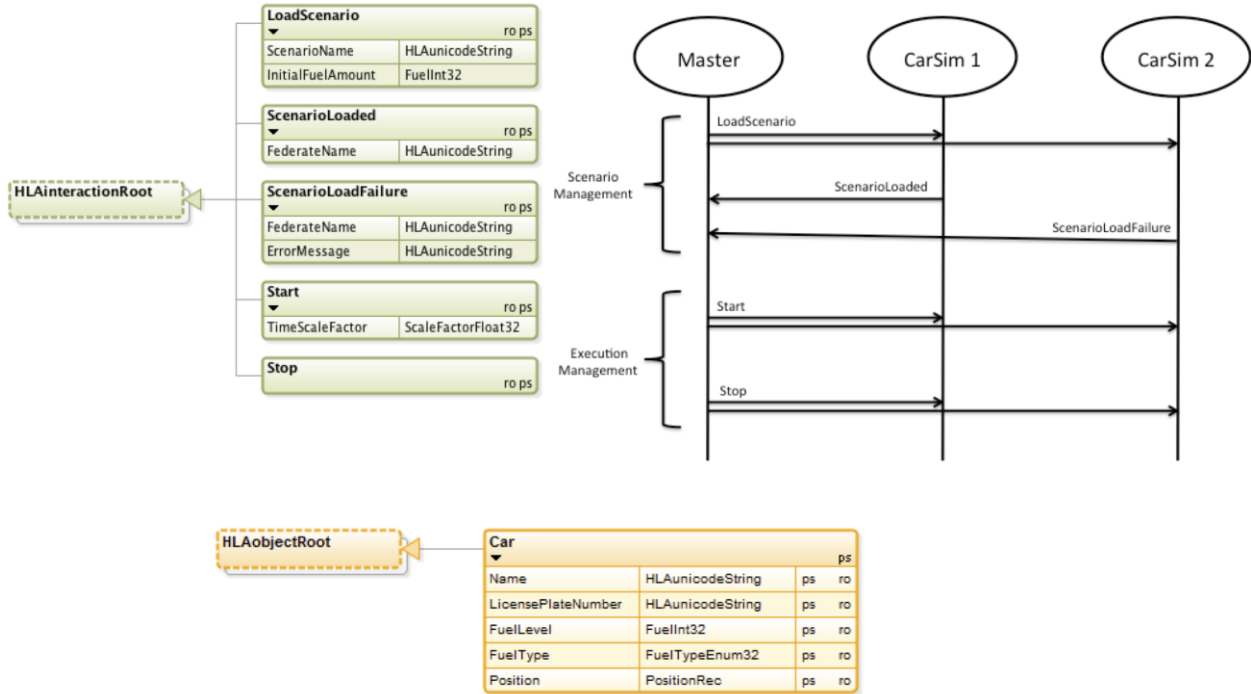


Figure 5: Fuel Economy Federation FOM and Conceptual Model

Figure 6 illustrates one possible federation execution of the FEF.

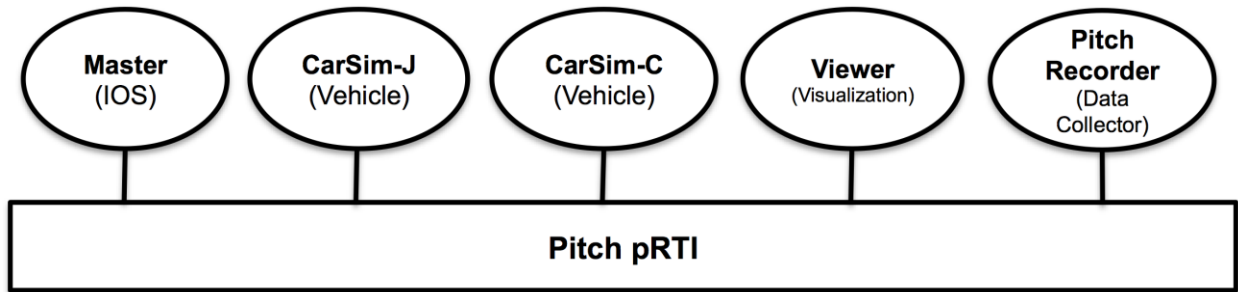


Figure 6: Example FEF Execution

An example network layout to support the federation execution in Figure 6 is provided in Figure 7.

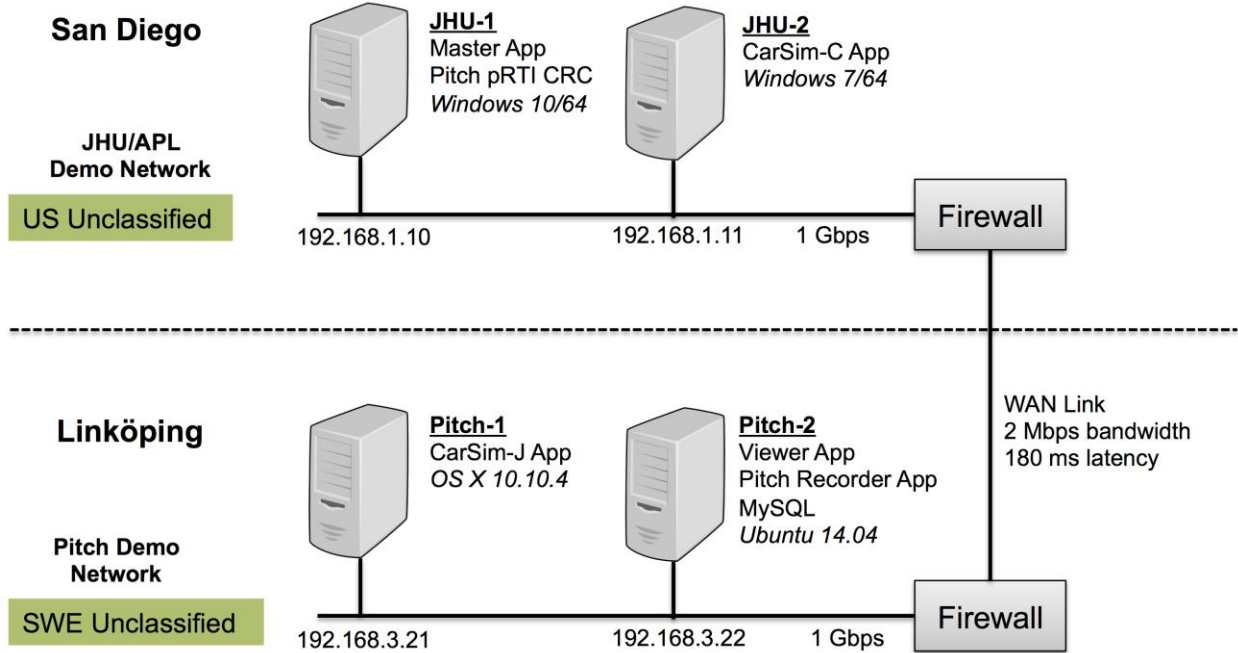


Figure 7: Network Layout

9 EXAMPLE EXCERPTS

This section contains selected excerpts from the example FEAT file [B5] used to illustrate how information from The HLA Tutorial can be rendered in FEAT. Each example includes a screen shot from the FEAT Editor or its output html, and a portion of the associated XML.

9.1 Metadata

Figure 8 illustrates the inclusion of Individuals in the Address Book agreement within the Metadata category. Table 9 is a portion of the associated XML. Notice that the XML reflects the other elements of an Individual's information not shown in the screen capture including organization and position³.

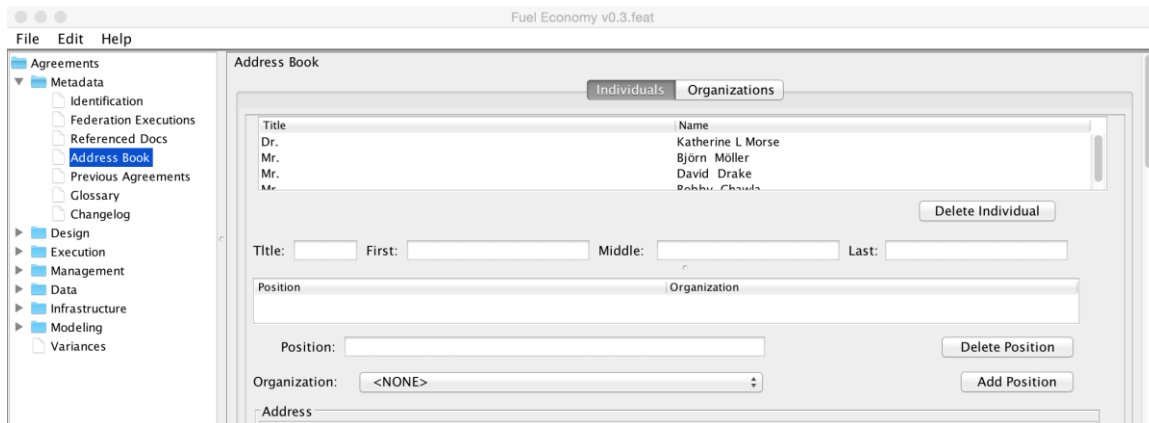


Figure 8: Address Book Example

Table 9: Address Book XML

³ FEAT has elements for recording phone numbers, email addresses, and physical addresses. This information is excluded from the example due to its sensitivity.

```

<ns4:addressBook>
  <people>
    <ns4:person ID="797919609881">
      <Title>Dr.</Title>
      <Name first="Katherine" middle="L" last="Morse"/>
      <Position value="Principal Professional Staff">
        <Org organizationID="686232545376">
          <Name>The Johns Hopkins University Applied Physics Laboratory</Name>
        </Org>
      </Position>
      <URL></URL>
      <ContactInstruction></ContactInstruction>
    </ns4:person>
    <ns4:person ID="619964345379">
      <Title>Mr.</Title>
      <Name first="Björn" middle="" last="Möller"/>
      <Position value="Vice President">
        <Org organizationID="296802761020">
          <Name>Pitch Technologies</Name>
        </Org>
      </Position>
      <URL></URL>
      <ContactInstruction></ContactInstruction>
    </ns4:person>
  </people>
</ns4:addressBook>

```

9.2 Design

Figure 9 illustrates the Tools agreement within the Design category. These tools are mostly the operating systems (OSs) illustrated in Figure 7: Network Layout. Table 10 is the associated XML. The FEAT Editor inserts a unique ID for each tool so that the tools can be referenced elsewhere in the FEAT file, e.g. in the Infrastructure / Hardware Configuration agreement where each piece of hardware has an OS assigned to it.

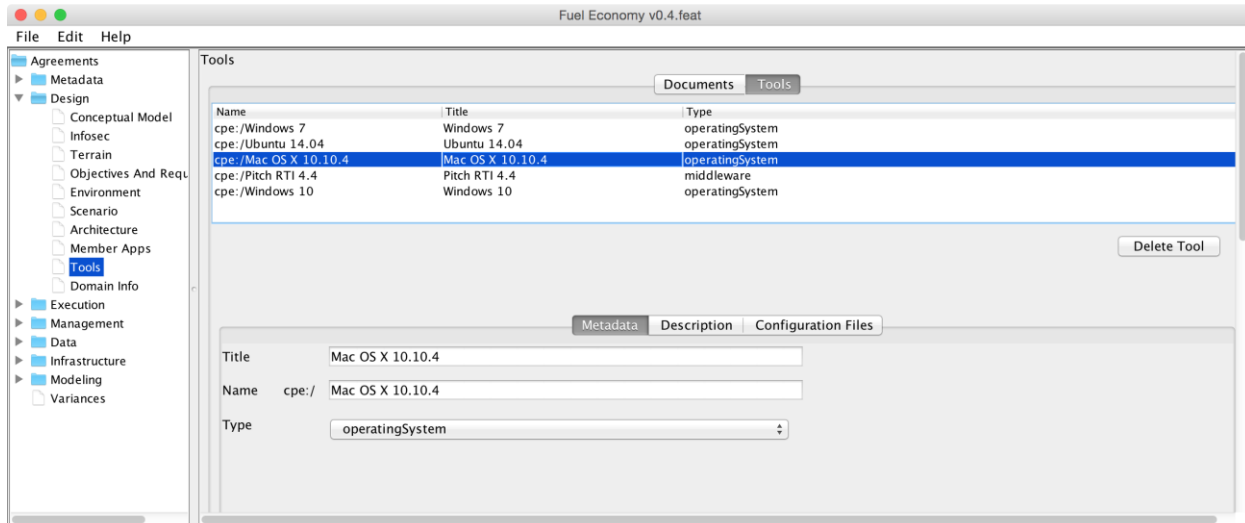


Figure 9: Tools Example

Table 10: Tools XML

```

<ns4:tools>
  <ns4:tool type="operatingSystem" id="166933805206">
    <ns6:cpe-item name="cpe:/Windows 7">
      <ns6:title>Windows 7</ns6:title>
    </ns6:cpe-item>
    <description/>
  </ns4:tool>

```

```

<ns4:tool type="operatingSystem" id="1053210785922">
  <ns6:cpe-item name="cpe:/Ubuntu 14.04">
    <ns6:title>Ubuntu 14.04</ns6:title>
  </ns6:cpe-item>
  <description/>
</ns4:tool>
<ns4:tool type="operatingSystem" id="295695426189">
  <ns6:cpe-item name="cpe:/Mac OS X 10.10.4">
    <ns6:title>Mac OS X 10.10.4</ns6:title>
  </ns6:cpe-item>
  <description/>
</ns4:tool>
<ns4:tool type="middleware" id="672450721800">
  <ns6:cpe-item name="cpe:/Pitch RTI 4.4">
    <ns6:title>Pitch RTI 4.4</ns6:title>
  </ns6:cpe-item>
  <description/>
</ns4:tool>
<ns4:tool type="operatingSystem" id="1091846856351">
  <ns6:cpe-item name="cpe:/Windows 10">
    <ns6:title>Windows 10</ns6:title>
  </ns6:cpe-item>
</ns4:tool>

```

9.3 Execution

EXECUTION STATE

NAME: running

REMOTE DOCUMENT REF

REFERENCED DOCUMENT TITLE: The HLA Tutorial

SECTION: A.5 Error handling

PAGE: 72

TITLE: Scenario and execution management

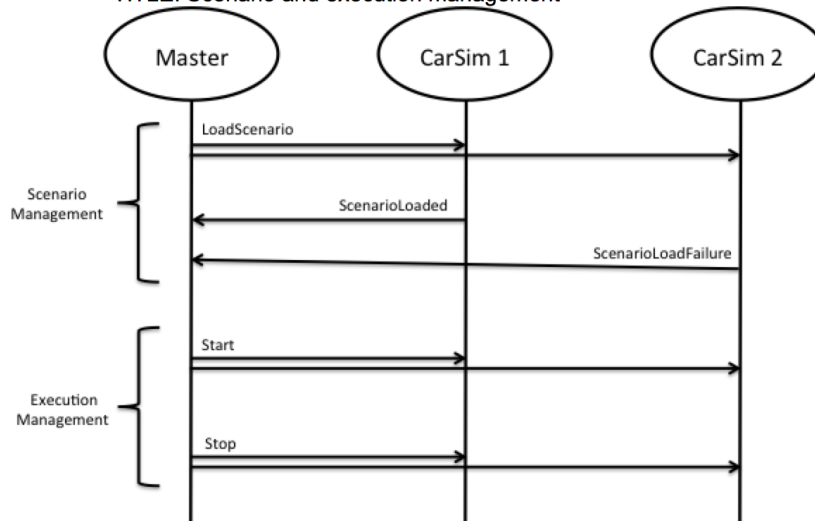


Figure 11 illustrates the Execution States agreement within the Execution category. This example makes use of the document reference design pattern described in 6.3; rather than reproducing a detailed description, a

reference is given to the external document including the relevant section. Figure 11 is the output HTML associated with this agreement. Notice that it includes both the text shown in Figure 10, and a figure of scenario execution and management from The HLA Tutorial. This graphic was imported under the Model Images tab of the Execution States agreement window, but cannot be displayed within the FEAT Editor. See section 10 - Future Tool Support for a discussion of this gap and others.

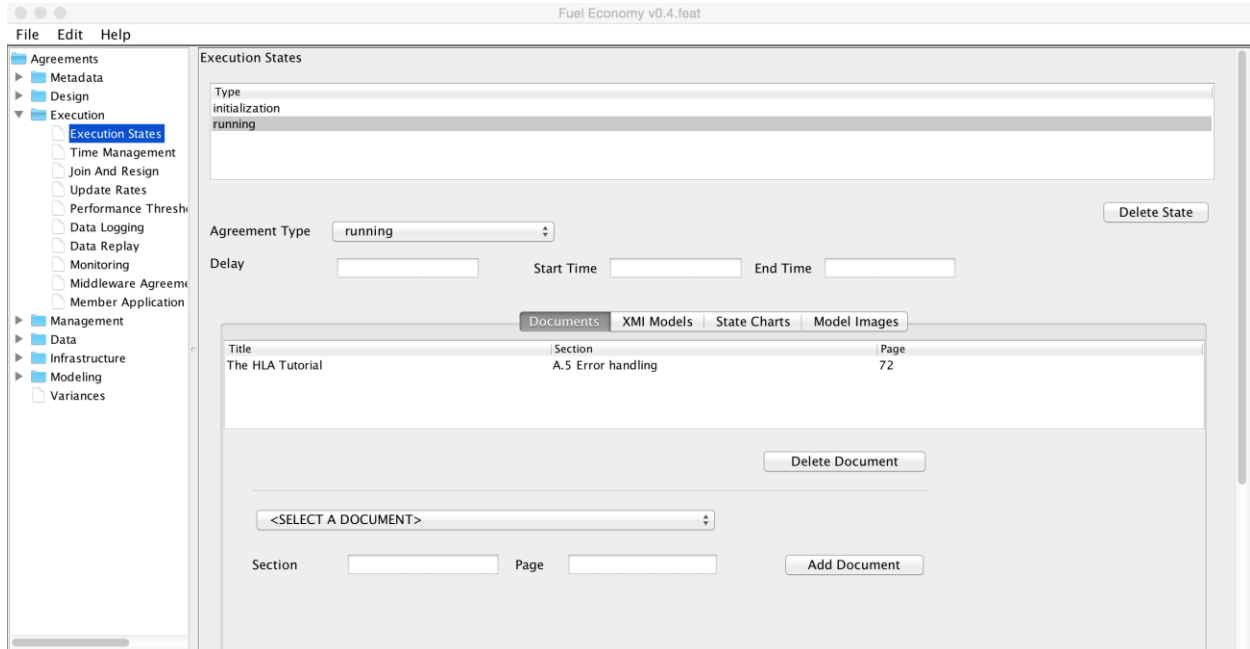


Figure 10: Execution States Example

EXECUTION STATE

NAME: running

REMOTE DOCUMENT REF

REFERENCED DOCUMENT TITLE: The HLA Tutorial

SECTION: A.5 Error handling

PAGE: 72

TITLE: Scenario and execution management

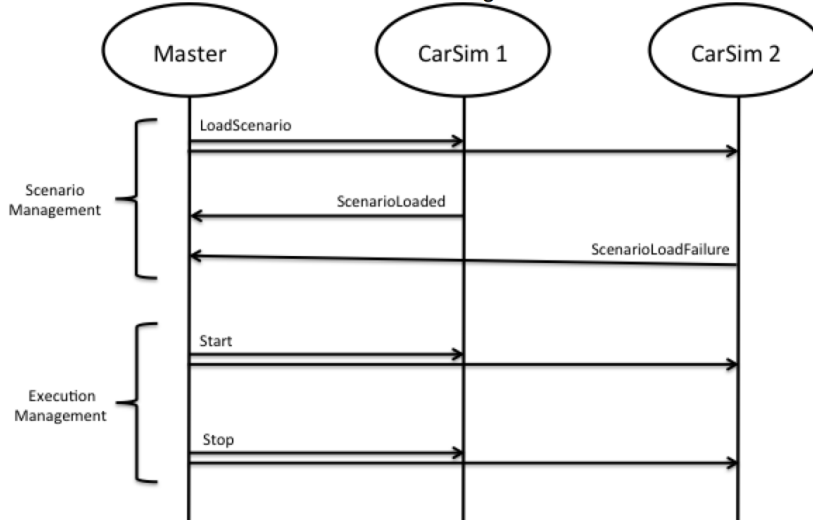


Figure 11: Execution State HTML

9.4 Management

Figure 12 illustrates the Software CM agreement within the Management category. Because The HLA Tutorial is a tutorial and not an actual federation execution, the tutorial document does not address processes, so this example was created solely for the purpose of this User's Guide. Table 11 is the associated XML.

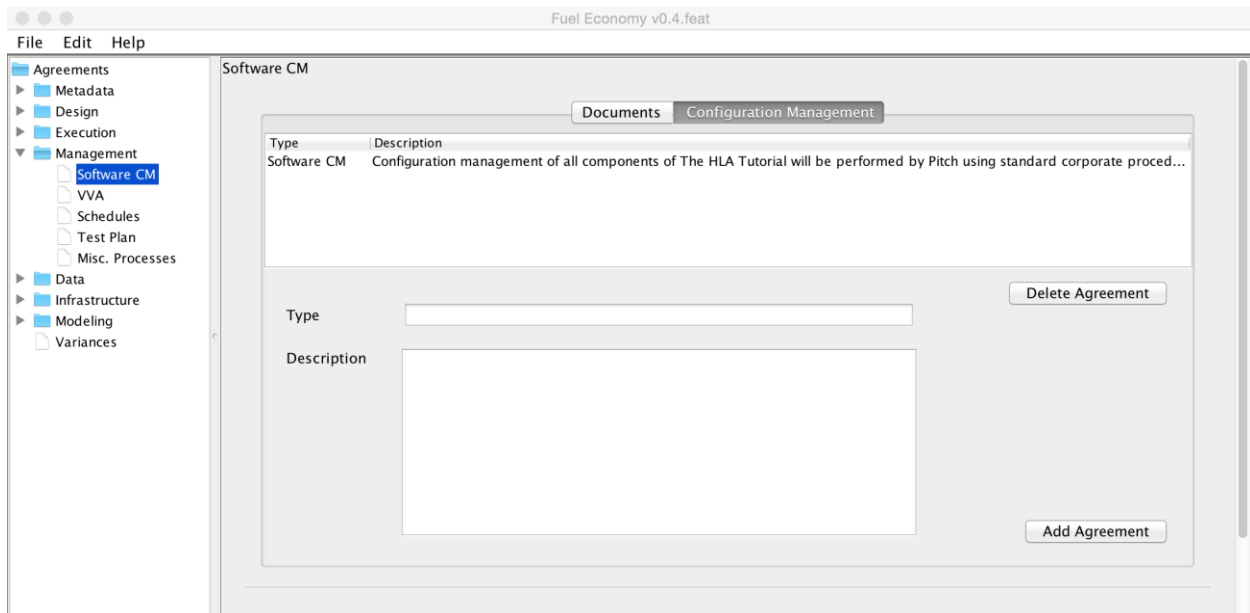


Figure 12: Configuration Management Example

Table 11: Configuration Management XML

```

<ns4:management>
  <ns4:softwareCM>
    <configurationManagement>
      <type>Software CM</type>
      <description>Configuration management of all components of The HLA Tutorial will be performed by Pitch
using standard corporate procedures.</description>
    </configurationManagement>
  </ns4:softwareCM>

```

9.5 Data

Figure 13 illustrates designation of the Data Exchange Model (DEM) within the Data category. When the FEAT is used to record HLA federations, the designation of the FOM will use an external document reference as illustrated. Table 12: DEM XML is the associated XML.

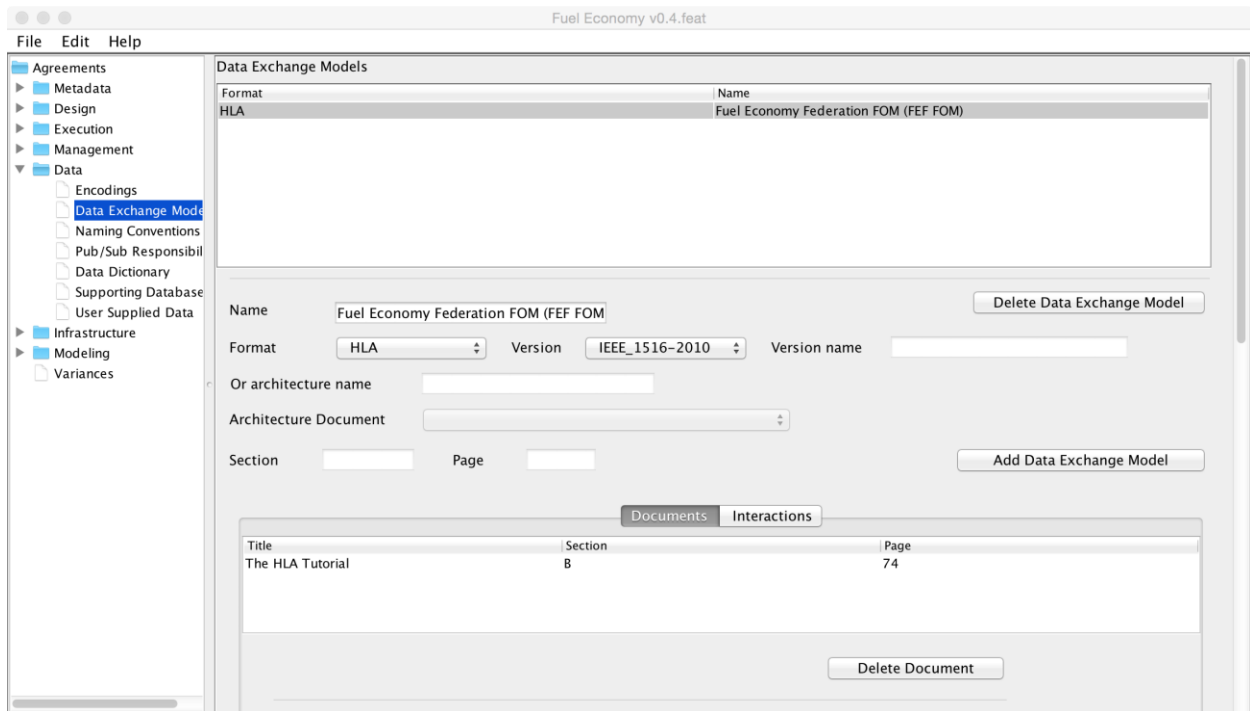


Figure 13: DEM Example

Table 12: DEM XML

```

<ns4:dataExchangeModels>
  <dem name="Fuel Economy Federation FOM (FEF FOM)">
    <ns4:remoteDocumentRef ref="795963216169" section="B" page="74"/>
    <ns4:simArchitecture>
      <architecture>
        <architecture>HLA</architecture>
        <version>IEEE_1516-2010</version>
      </architecture>
    </ns4:simArchitecture>
  </dem>
</ns4:dataExchangeModels>

```

9.6 Infrastructure

Figure 14 illustrates the Hardware Configuration agreement within the Infrastructure category. Notice that the individual agreements correspond to the network architecture elements illustrated in Figure 7: Network Layout. Figure 15 is the output HTML associated with this agreement. Notice the OS Ref elements; these are the unique IDs assigned to OS versions as described in section 9.5. These unique IDs are used within the FEAT Editor to populate the Operating System drop down menu in Figure 14.

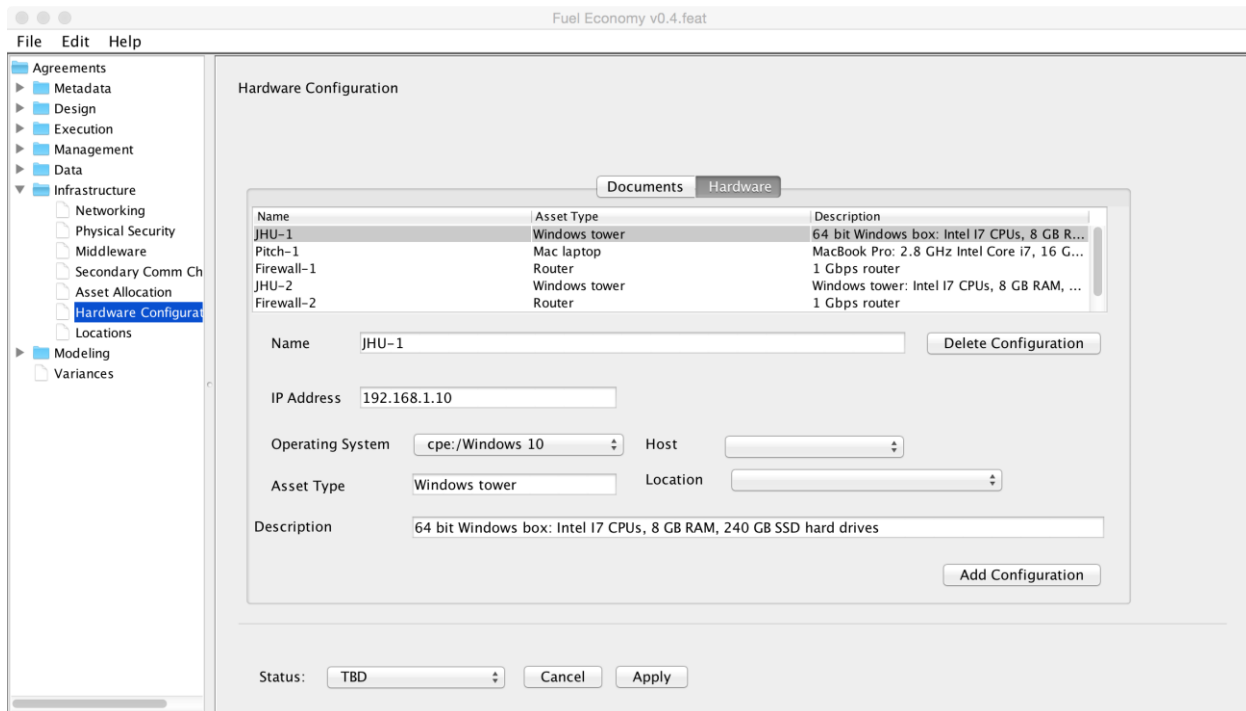


Figure 14: Hardware Configuration Example

HARDWARE CONFIGURATION

HARDWARE

NAME: JHU-1
 DESCRIPTION: 64 bit Windows box: Intel I7 CPUs, 8 GB RAM, 240 GB SSD hard drives
 ID: 567768676130
 IP ADDRESS: 192.168.1.10
 OS REF: 1091846856351
 ASSET TYPE: Windows tower

HARDWARE

NAME: Pitch-1
 DESCRIPTION: MacBook Pro: 2.8 GHz Intel Core i7, 16 GB 1600 MHz DDR3
 ID: 473224104195
 IP ADDRESS: 192.168.3.21
 OS REF: 295695426189
 ASSET TYPE: Mac laptop

Figure 15: Hardware Configuration HTML

9.7 Modeling

Figure 16 illustrates the Miscellaneous agreement within the Modeling category. This agreement was chosen because the FEF doesn't use the other types of modeling agreements: Effects Adjudication, Coordinate Systems, etc. This figure also illustrates the use of the Table feature in the FEAT Editor. Figure 17 shows the table as exported to HTML.

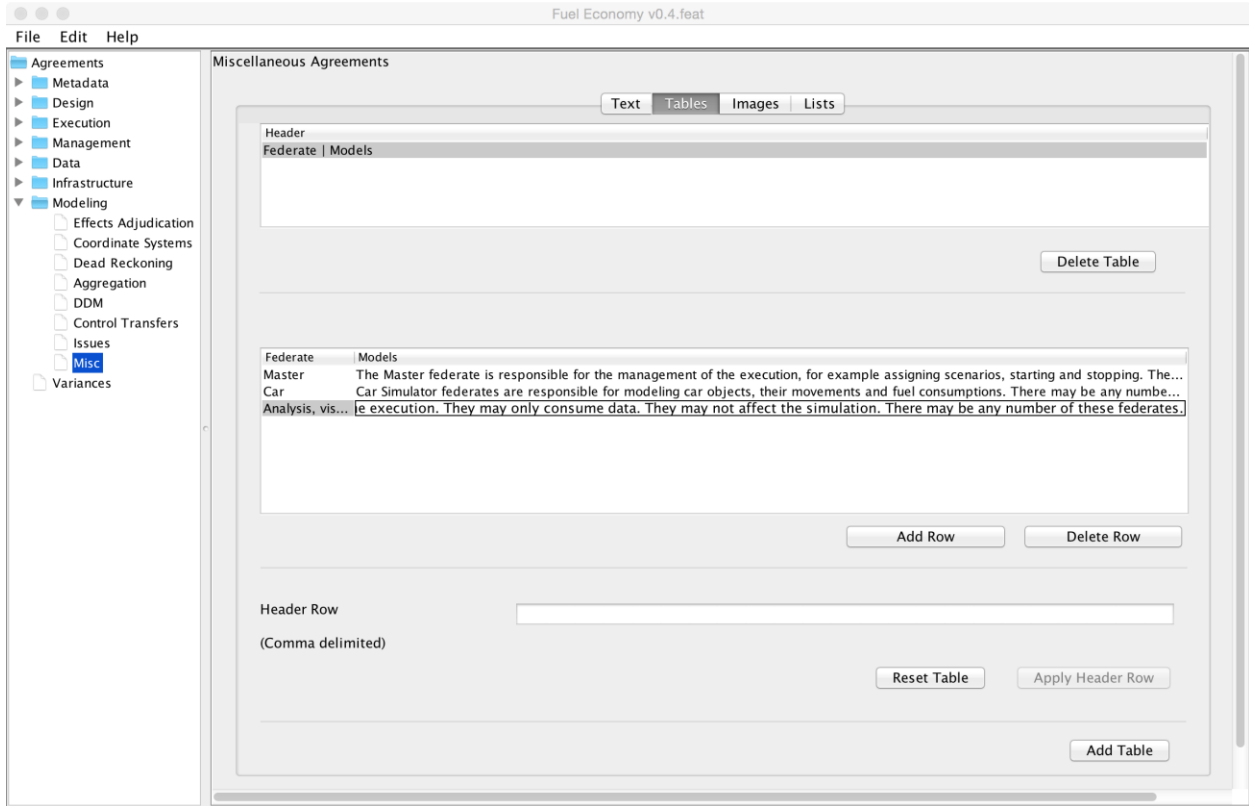


Figure 16: Miscellaneous Modeling Example

MODELING

DDM

REMOTE DOCUMENT REF

REFERENCED DOCUMENT TITLE: The HLA Tutorial

SECTION: A.2

PAGE: 68

MISC

| Federate | Models |
|---|---|
| Master | The Master federate is responsible for the management of the execution, for example assigning scenarios, starting and stopping. There shall be exactly one Master federate in each federation. |
| Car | Car Simulator federates are responsible for modeling car objects, their movements and fuel consumptions. There may be any number of car simulator federates. Each federate may model one or more car objects. |
| Analysis, visualization and data collection | These are used for analysis of the simulation results during and after the execution. They may only consume data. They may not affect the simulation. There may be any number of these federates. |

Figure 17: Miscellaneous Modeling HTML

9.8 Variances

Since the FEF was not executed, there are no variances as illustrated in Figure 18.

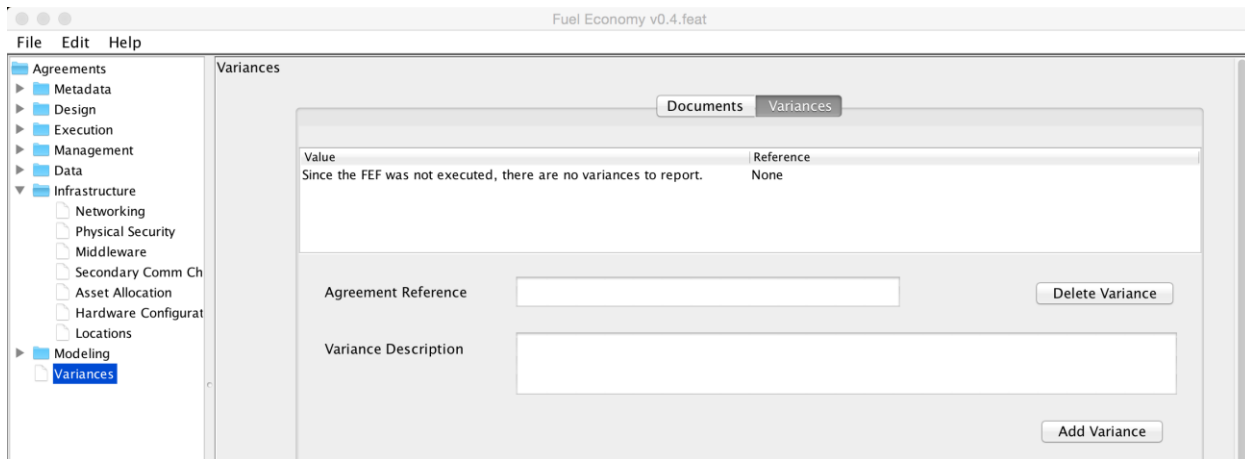


Figure 18: Variances Example

10 FUTURE TOOL SUPPORT

The FEAT Editor described in 5.2 is very basic; it lacks features that would improve usability of FEAT. The list below provides a recommended, initial list of features that would be desirable in a commercial-quality editor:

- User profiles – Depending on a user's role within the federation engineering process, some agreements would be irrelevant to them. For example, a network engineer is unconcerned with time management mechanisms. Each user should be able to tailor the editor's presentation to their own, role-focused needs. The FEAT PSG might produce a set of default profiles. Optimally, these profiles would be loaded by the editor to create an easy upgrade path and to allow individual users to edit and save their own profiles.
- File merging – If individual users are operating on different groups of agreements, or if a new federation is being built from multiple pre-existing federations with their FEAT files, a file merging feature would speed the merging process and reduce errors. This feature would be analogous to those available now in commercial Object Model Template tools.
- Web based – Based on the preceding two features, it's easy to see that configuration management of a FEAT file for a large federation with many participants is challenging. A web-based editor that controls access and maintains configuration management at the level of individual agreements would significantly improve this situation.
- Graphics – Although the FEAT Editor can import graphics files and export them embedded in the HTML output, it lacks the functionality to present them within the Editor. Including this functionality would streamline development of agreements supported by graphics because the user could immediately, visually check the validity of graphics without going through the extra step of exporting the HTML.

Annex A Bibliography

(Informative)

- [B1] FEAT Programmer's Reference,
<http://www.sisostds.org/FEATProgrammersReference/fedagreements.html>
- [B2] FEAT Editor, <http://sourceforge.net/projects/feateditor/>
- [B3] Distributed Simulation Engineering and Execution Process (DSEEP), IEEE 1730-2011, 24 January 2011
- [B4] DSEEP Multi-Architecture Overlay (DMAO), IEEE 1730.1, 23 August 2013
- [B5] Fuel Economy Federation Agreements, FEAT file format:
https://www.sisostds.org/DigitalLibrary.aspx?Command=Core_Download&EntryId=45448 HTML format:
https://www.sisostds.org/DigitalLibrary.aspx?Command=Core_Download&EntryId=45449
- [B6] Pitch HLA Tutorial, <http://www.pitch.se/hlatutorial>
- [B7] XML Linking Language (XLink), available at <http://www.w3.org/1999/xlink>, June 2000
- [B8] XML Metadata Interchange (XMI), available at <http://schema.omg.org/spec/XMI/2.1>, September 2005
- [B9] Common Platform Enumeration (CPE), available at <http://cpe.mitre.org/language/2.0>, March 2009
- [B10] Geography Markup Language (GML), available at <http://www.opengis.net/gml>, February 2010