



NLR-TP-2004-003

## Evaluation of Collective Training in a Distributed Simulation Exercise

Arjan J.J. Lemmers, Bernd Rollesbroich\*, Timo Hartikainen\*\*  
and Francisco J. Rodriguez Carvajal\*\*\*

\* CAE Elektronik GmbH, \*\* Instrumentointi Oy, \*\*\* Sainseel Sistemas Navales, SA

This report has been based on a paper presented at Human Factors & Medicine Panel Symposium on "Advanced Technologies for Military Training", at Genoa, Italy on 13-15 October 2003.

This report may be cited on condition that full credit is given to NLR and the authors.

Customer: National Aerospace Laboratory NLR  
Working Plan number: AT.1.G.2  
Owner: National Aerospace Laboratory NLR  
Division: Flight  
Distribution: Unlimited  
Classification title: Unclassified  
December 2003

Approved by author:  28/1/04	Approved by project manager:  28/1/04	Approved by project managing department:  28/1/04
--	---	---





## **Summary**

The military interest in Synthetic Environments (SE) is beginning to change from thinking about the relevance of capability demonstrators to SEs being used to support distributed simulation exercises. Several European nations are actively promoting the use of SEs for Collective Training purposes to increase the military co-operation in Europe. In order to overcome the obstacles for use of distributed simulation exercises across Europe, it is important that a common European process, tools and standards are defined. This paper is focussing on the evaluation issues in distributed simulation exercises. It will outline the process of defining the evaluation needs, the identification of the functional and technical evaluation requirements and the definition of the Common Evaluation Framework (CEF). The CEF comprises processes, models, methods and presentation means for results presentation and distribution. Three supporting prototype tools for the CEF have been developed under the EUCLID RTP11.13 programme: the Evaluation Definition Tool (EDT), the Evaluation Definition Selection Tool (EDST) and the Execution Evaluation Tool (EET). The EDT is used by the military user to define evaluation objectives and criteria. The EDST gives possibilities for searching and selecting evaluation objectives from a pool of ready-made objectives. The EET supports the user in post-processing the outputs acquired during the exercise and in the analysis and evaluation of the results and trainees.



This page is intentionally left blank



## Acronyms

CEF	Common Evaluation Framework
CGF	Computer Generated Forces
CMMI	Capability Maturity Model-Integrated
COTS	Commercial Of The Shelf
CT	Collective Training
CTO	Collective Training Objective
DIF	Data Interchange Format
DMSO	Defence Modeling and Simulation Office
EDST	Evaluation Definition Selection Tool
EDT	Evaluation Definition Tool
EET	Execution Evaluation Tool
EK	Evaluation Knowledge
EUCLID	European Cooperation for the Long-term In Defence
FEDEP	Federation Development and Execution Process
FOM	Federation Object Model
GUI	Graphical User Interface
HLA	High Level Architecture
IEEE	Institute of Electrical and Electronic Engineers
MEC	Mission Essential Competency
NATO	North Atlantic Treaty Organisation
RTP	Research Technology Program
SAS	Studies, Analysis and Simulation
SE	Synthetic Environment
SEDE	SE Development Environment
SEDEP	Synthetic Environment Development & Exploitation Process
SEMT	SE Management Tool
SME	Subject Matter Expert
WAN	Wide Area Network
XML	eXtensible Mark-up Language



## Contents

<b>1</b>	<b>Introduction</b>	<b>7</b>
<b>2</b>	<b>What is collective training</b>	<b>9</b>
<b>3</b>	<b>The missing step: Evaluation</b>	<b>11</b>
<b>4</b>	<b>The Common Evaluation Framework</b>	<b>15</b>
<b>5</b>	<b>Evaluation data</b>	<b>19</b>
<b>6</b>	<b>Evaluation Definition Tool and Evaluation Definition Selection Tool</b>	<b>21</b>
<b>7</b>	<b>Execution Evaluation Tool</b>	<b>24</b>
<b>8</b>	<b>Conclusions</b>	<b>26</b>



## 1 Introduction

It is well known that in the training domain, individual skills are taught much more effectively using tailor-made training devices. The main advantage of Synthetic Environment (SE) is in training certain collective skills. For the purpose of this study, the use of networked simulations in a training centre that satisfy particular collective training objectives is not considered to be an SE; although the individual simulators can be used as assets in an SE. The training SEs considered in this programme are those that require a range of different simulations (either manned or constructive) to be identified, configured and networked. This type of SE could be used to replace joint and coalition training exercises that are currently conducted using real equipment that is costly to operate.

The military interest in SEs is beginning to change from thinking about the relevance of capability demonstrators to SEs being used to support real programmes. Several European nations are actively promoting the use of SEs within their countries. However, because of the distributed nature of SEs they are also well suited to support the increase in military co-operation in Europe; through coalition training, multinational operations and multinational equipment acquisition. In order to promote the use of distributed simulation exercises across Europe, it is important that a common European process, tools and standards are defined.

EUCLID RTP 11.13 is a major European initiative to promote the use of Synthetic Environments (SE). The title of the programme 'Realising the Potential of Networked Simulations in Europe' reflects the fact that although SEs are currently being used to support defence programmes in Europe, their full potential is not currently being realised. The aim of the project is to 'overcome the obstacles that prevent SEs being exploited in Europe by developing the SE Development Environment (SEDE). SEDE provides a facility that will assist the different types of SE users i.e. Problem setters, Problem Solvers, and SE Implementers, so that SEs can be delivered faster, better and cheaper. It will achieve this by providing a common shared data environment, providing facilities for managing the data generated by an SE project and making information about available SE assets and best practices readily available to SE users. The SEDE comprises of five main components: a Process, Repository, SE Management Tool, SE Tools (both COTS and those being prototyped in EUCLID 11.13) and a Knowledge Base.

The Federation Development and Execution Process (FEDEP) has been used as a baseline for the process but has been modified and extended where it has been found to be lacking. One of the shortcomings was that the FEDEP did not cover the complete lifecycle of a SE. Therefore one of the extensions is a new step "Evaluation". The resulting RTP 11.13 process is known as



the Synthetic Environment Development & Exploitation Process (SEDEP). The purpose of the SE tools is to assist the SE users in performing their roles. Several COTS tools are already available for supporting some of the SEDEP activities and additional tools are being prototyped where none currently exist. A key requirement for supporting the SEDEP is the use of common data formats, captured in the SEDE data model. In this way, the data created by one tool can be read by the tool supporting the next activity in the process. All tools will have access to the RTP11.13 Repository, which will provide the mechanism for transferring data between them. The Data Interchange Formats (DIFs) defined by the High Level Architecture (HLA) are being extended to other areas supported by the SEDEP. The SEDE may access the data from a local Repository or from one that has been distributed over a Wide Area Network (WAN).

This paper is focussing on the evaluation issues in the SEDE. It will outline the process of defining the evaluation needs, the identification of the functional and technical evaluation requirements and the definition of the actual Common Evaluation Framework (CEF). The CEF comprises processes, models, methods and presentation means for results presentation and distribution. Three supporting prototype tools for the CEF have been developed under the EUCLID RTP11.13 programme: the Evaluation Definition Tool (EDT), the Evaluation Definition Selection Tool (EDST) and the Execution Evaluation Tool (EET). The EDT is used by the military user to define evaluation objectives and criteria. The EDST gives possibilities for searching and selecting evaluation objectives from a pool of ready-made objectives. The EET supports the user in post-processing the outputs acquired during the exercise and in the analysis and evaluation of the results and trainees.



## 2 What is collective training

Collective training (CT), in military terms, has been defined by the NATO SAS-13 Military Application Study on 'NATO Mission Training via Distributed Simulation' as training which

- "...involves 2 or more 'teams', where each team fulfils different 'roles', training in an environment defined by a common set of collective training objectives (CTOs)".
- A team is defined as "a number of individuals who may have different 'tasks' within that team but whose operational remit is to fulfil a specific role e.g. a tactical 4-ship in a ground-attack role."

CT applies to training of military groups in order to maintain or improve the groups' ability to perform in terms of service. The trained group in a single CT exercise may include multiple crews of similar or dissimilar vehicles and possibly different domains. To improve the fidelity the exercise may also involve Computer Generated Forces (CGF) as in ref. [1], or may be conducted as embedded training. CT is considered as the training required to prepare cohesive teams and units to accomplish their assigned operational missions. CT is part of a continuous process of unit training and is generally conducted within operational units, or specialist training facilities available to operational units on timeshare basis. CT exercises individual tasks, skills and responsibilities and collective command and control responsibilities.

Generally, the teams that participate in a CT exercise comprise a battle group, possibly including complete command and control functions. In addition to rehearsing within one military application, CT is used in joint exercises where the cooperation between e.g. air force, navy and army is practised. From the performance point of view, performance of a team is a product of the competencies of the different individuals. The performances of the individuals affect the performance of the teams and the whole trained battle group. According to ref. [2] competencies may be defined for successful performance. The competencies may be divided into individual, intra-team and inter-team competencies. When these competencies are applied to the performance of certain teams and missions, they become mission essential competencies (MEC). In CT applications the inter-team MECs are considered more important than the individual or intra-team MECs. Although the same underlying skills may be employed at different levels of MECs, such as communication, co-ordination etc., these are applied in a different context (ref. [2], [3]).

In order to measure the effectiveness of the simulation it is essential to be able to assess whether adequate CT is being achieved. However, the effectiveness of CT has been hampered by the difficulties involved in evaluating collective performance and feeding this information back to the personnel being trained. Whilst training objectives at an individual level can be defined in



relation to specific tasks and are role specific, it is much more difficult to define a set of measurable collective training objectives, which apply to all members of a CT audience. The distributed, group nature of CT makes any measurement of performance difficult, with the level of difficulty increasing as the size of the unit under training increases. Currently only basic, usually subjective, forms of measurement exist at the sub-unit level and above. The EUCLID RTP11.13 team has identified this fact as a major obstacle for realising the potential of using SEs for CT purposes. The process of setting up SEs for CT needs to be refined further in order to derive more robust CT metrics, which could be used in live or synthetic exercises. To achieve this work should be done in further developing the process and tools to support evaluation matters for CT.



### 3 The missing step: Evaluation

The requirements for the process that should be developed in RTP11.13 were:

- Provide support to encourage the use of SE technology on military programmes;
- Provide guidance for SE developers and users to plan and perform the different activities necessary to produce the required products and results;
- Promote good practice for developing SEs on time and within budget;
- Promote reuse of products (federation, federates, components) and results;
- Provide a framework for a tool set to reduce the cost and time for producing and using SEs.

Because the Federation Development and Execution Process (FEDEP) (Ref. [4]) has already been widely adopted and is already supported by several COTS tools, the decision was made in EUCLID 11.13 to use the FEDEP as a baseline instead of developing a new SE process from scratch. The FEDEP version 1.5 comprised 6 steps:

- Step 1: Define Federation Objectives. The federation user and federation development team define and agree on a set of objectives and document what must be accomplished to achieve those objectives.
- Step 2: Develop Federation Conceptual Model. Based on the characteristics of the problem space, an appropriate representation of the real world domain is developed.
- Step 3: Design Federation. Federation participants (federates) are determined and required functionalities are allocated to the federates.
- Step 4: Develop Federation. The Federation Object Model (FOM) is developed, federate agreements on consistent databases/algorithms are established, and modifications to federates are implemented (as required).
- Step 5: Integrate and Test Federation. All necessary federation implementation activities are performed, and testing is conducted to ensure that interoperability requirements are being met.
- Step 6: Execute Federation and Prepare Results. The federation is executed, outputs are generated, and results are provided.

The analysis performed in EUCLID 11.13 identified among others the following limitations of the FEDEP:

- FEDEP does not support all good practice management activities.
- FEDEP does not provide assistance to all the different types of SE users.
- FEDEP does not cover the complete lifecycle of an SE, it focuses on the development part and the analysis and evaluation activities are lacking.
- FEDEP does not explicitly identify products at each step of the process.



EUCLID 11.13 has taken the initiative to enhance and extend the FEDEP process where it was perceived to have limitations. The resulting EUCLID 11.13 process is known as the Synthetic Environment Development & Exploitation Process (SEDEP) (ref. [5]). The use of the term SEDEP has been chosen to reinforce its close links with the FEDEP whilst promoting its more general use for developing SEs. To overcome the limitation of the FEDEP, in the SEDEP the following enhancements were made:

- The SEDEP is matched with the different recommended activities of the "Capability Maturity Model-Integrated" (CMMI) for good practice development. This makes SEDEP compatible with the Standard System Engineering Process.
- The SEDEP introduces two new steps (see Figure 1) dedicated to
  - a) Support the users in determining the suitability of the SE in solving their problem and estimating project parameters such as cost, duration, risk etc.
  - b) Analyse the execution outputs and evaluating the results.**
- SEDEP provides an overlay representation to allow traceability of specific technical objects or parameters along the full process.
- SEDEP explicitly identifies and defines inputs and outputs for each step and activity of the process.
- SEDEP specifies the use of a repository to provide a means of storing the information about the SE and for support tools to transfer data between the different phases of the process.
- SEDEP explicitly identifies and defines the different library components in the repository used by the different steps and activities.
- SEDEP provides capability for components, specifications and definitions reuse.

It should be noted that the current steps of the FEDEP exist as a sub-set of the SEDEP (see Figure 1). It is intended that long term, the two processes will merge and that there will only be a single SE process.

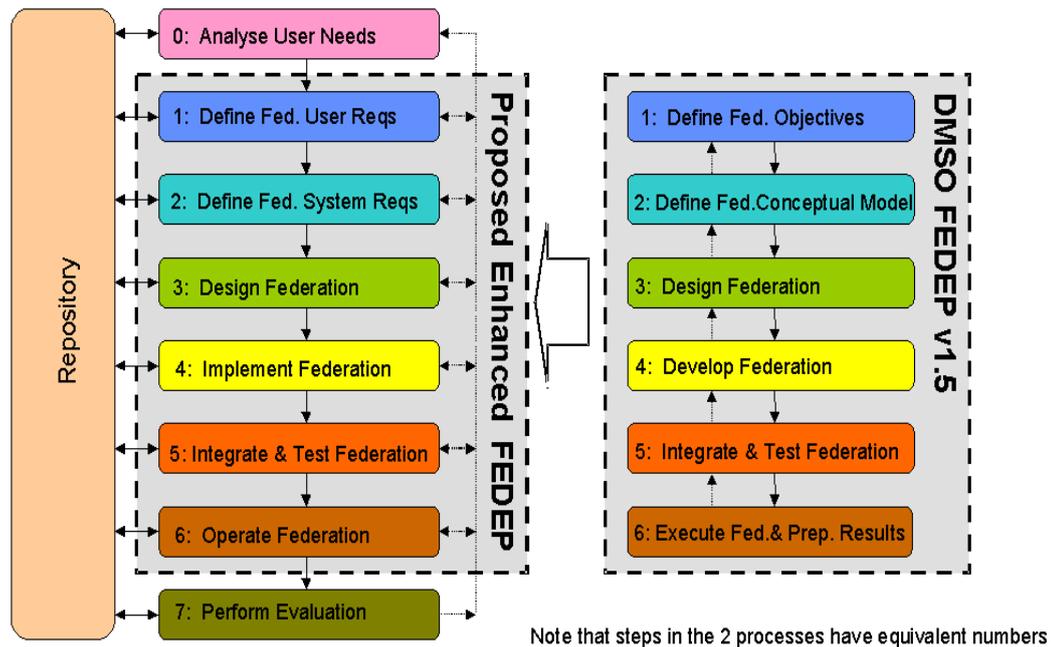
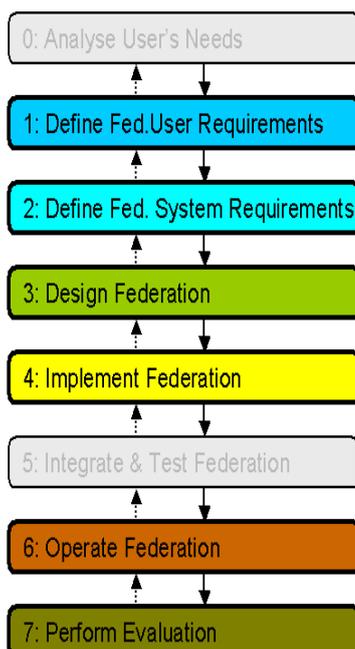


Figure 1: SEDEP Relationship with FEDEP

The FEDEP developers are currently working on transforming the FEDEP v1.5 into a IEEE standardized process. Discussions took place with the FEDEP Productisation Group to influence the development of the FEDEP before it became an IEEE recommended practice. RTP 11.13 proposed several new steps, activities, and tasks derived from their results in SEDEP v1.0. 16 changes were adopted which are now included in the IEEE 1516.3 FEDEP version. The most important contribution of EUCLID RTP 11.13 is the new FEDEP step 7 “Analyse Data and Evaluate Results”.



Because of the continuous discussion about evaluation in the SEDEP and the experience gathered during the tool development it was recognized that there are a lot of activities and tasks relevant for evaluation in different steps of the SEDEP.

In step 1 “Define Federation User Requirements” the Problem Setter and Problem Solver must define which behaviour, skills, characteristics, tactics, procedures, functionality, etc. should be analysed and evaluated (evaluation objectives). The results of step 1 are required to determine the criteria, methods, algorithms, questionnaires, checklists, and presentation information, which have to be used to perform the evaluation.



The criteria, methods, and algorithms are defined in detail within several activities in step 2, 3 and 4.

In step 2 “Define Federation System Requirements” the evaluation related information can already be very detailed, but it is also possible that only the name of a new algorithm is defined in this step and that the details are further elaborated in step 3.

In step 3 “Design Federation” the output of the previous step has to be completed and transformed into a generic format. The result of step 3 is a complete and correct mathematical description of all formulas required to post-process the data logged during the execution phase. In the context of evaluation the purpose of step 4 “Implement Federation” is to produce algorithms and formulas in tool specific formats. The algorithms are allocated to suitable mathematical tools according to the tools’ capabilities to apply the required methods. The respective tools are used in step 7 to analyse and evaluate the execution outputs.

In step 6 “Operate Federation” all necessary execution data is collected. This includes filled in questionnaires and checklists. The collected data is filtered and transformed into a generic format. After the execution of the Synthetic Environment these ‘prepared execution outputs’ are analysed and evaluated in step 7 “Perform Evaluation” to provide the desired feedback.

#### 4 The Common Evaluation Framework

One of the key factors to reduce the cost and time scale of creating and utilising SEs is the reuse of SE components, specifications, and definitions. To enable and facilitate the reuse of evaluation data it is important to define standards for the type, structure, and format of this data. EUCLID RTP 11.13 identified that there is a lack of information about evaluation of SEs. The information available was collected and analysed to provide a baseline for evaluation called the Common Evaluation Framework (CEF). The CEF captures evaluation aspects in general, i.e. regardless whether the SE is used for Collective Training, Mission Rehearsal, or Simulation Based Acquisition.

Before an exercise can be executed data like criteria, methods, and algorithms must be defined. This data must be taken into account for the SE design and implementation in order to meet the evaluation requirements. Within RTP 11.13 the term “Evaluation Need” describes the different items that have to be defined for a complete evaluation (Figure 2).

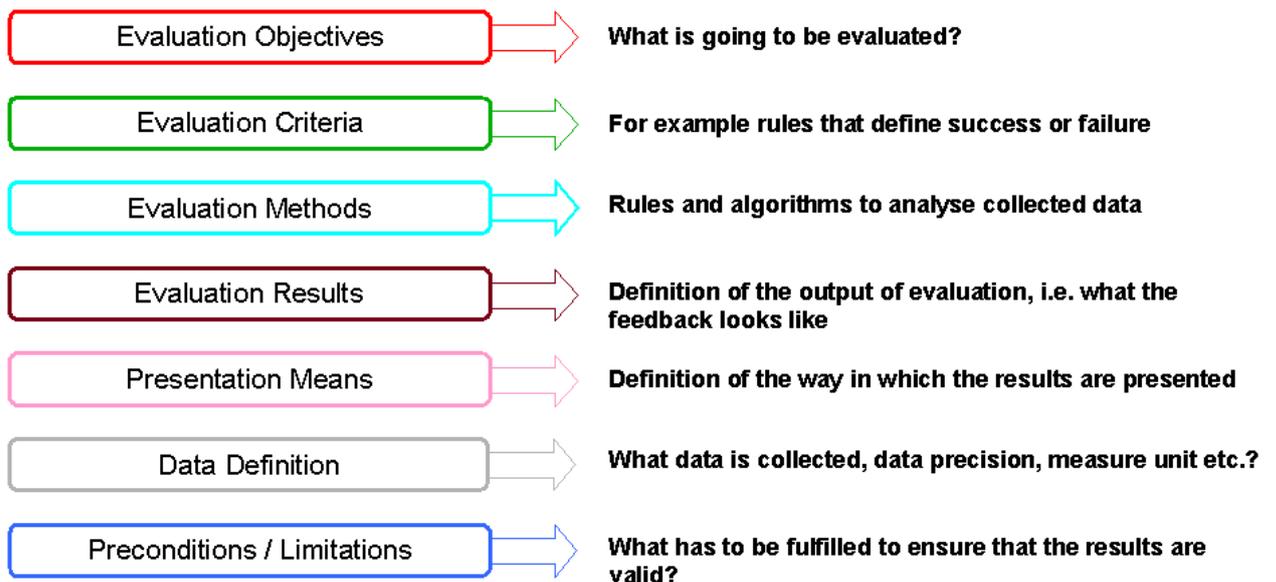


Figure 2: Evaluation Need

An Evaluation Need consists of Evaluation objectives, criteria, methods and rules, data definition, preconditions, Evaluation data collected during the execution of the SE and the presentation of Evaluation results (i.e. feedback). The Evaluation results are presented with different means (e.g. 3-D visualisation, map drawings, images, text) depending on the purpose and target of the Evaluation. The Evaluator decides the presentation means of the evaluation. Evaluation addresses both the Evaluation of the success or failure of the exercise (presented in



military terms) and the assessment of the value of the synthetic environment in satisfying the user's needs in training, rehearsal, acquisition etc.

The Evaluation Objective can be derived from the user's needs to describe a goal of the evaluation in a high-level format. Due to the objective a criterion will be determined that is used to judge the quality of the collected data. In this case quality is related to the performance of the evaluated object and not e.g. to the data completeness. The input data (parameters, variables etc.) that are needed for applying a criterion are produced by applying methods to analyse the collected data. It is not absolutely necessary to process the results of the data analysis within a criterion to receive suitable evaluation results. Applying a criterion or a method or both produces evaluation results. The procedure to generate the evaluation result depends on the evaluation objective. The evaluation results are represented by presentation means, e.g. graphs, charts, or textual reports.

The definition of the data used in criteria and methods contains the name of the data item, precision, unit, etc. The data represented by the elements of an Evaluation Need can be used to determine the Evaluation System Requirements. This is a low-level description containing the technical and functional requirements for the SE that are important to meet the user's expectations of evaluation.

Example of an Evaluation Need:

In the following the concept of the "Evaluation Need" is explained using a simple example from the air force domain. Several fighter aircrafts have to keep the correct flight formation for optimal observation of all sectors around the aircraft. The line abreast formation and its constraints are shown in Figure 3. To simplify the example only the distance between the wingman and the lead and the aircraft's altitude will be analysed.

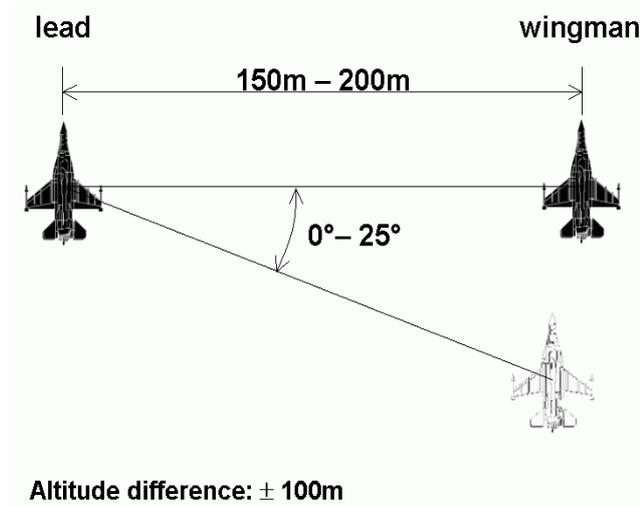


Figure 3: Line Abreast Formation

The position of aircraft is identified by Lat/Lon coordinates (degree°/minute'/second''). Altitude is provided in feet. Flight time while in line abreast formation is provided in seconds. The distance between flight lead and wingman can be calculated approximately by the following algorithm:

$$\text{distance} = (1,852 \times 60 + \arccos(\sin P1 \times \sin P2 + \cos P1 \times \cos P2 \times \cos LD)) \times 1000 \text{ unit: m}$$

P1: Latitude Position Flight Lead in decimal degrees

P2: Latitude Position Wingman in decimal degrees

LD: Longitude Difference in decimal degrees

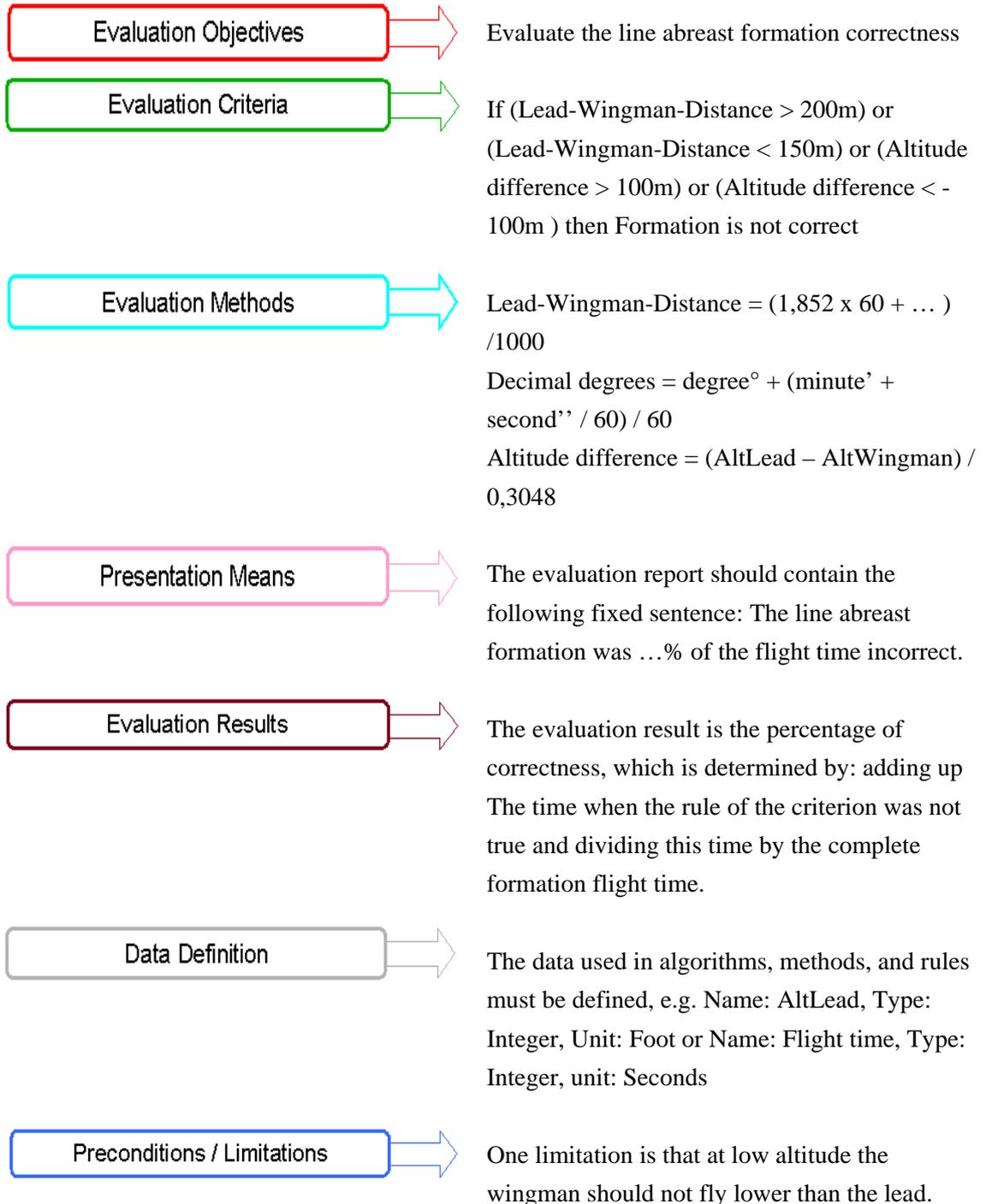
The Longitude and Latitude coordinates are transformed into decimal degrees by applying the following formula:

$$\text{Decimal degrees} = \text{degree}^\circ + (\text{minute}' + \text{second}'' / 60) / 60$$

To evaluate the altitude the altitude difference must be transformed from feet into meters (1ft = 0,3048m).



To gain a better understanding of the Evaluation Needs elements this information is linked to the different elements:



## 5 Evaluation data

A key requirement for supporting the SEDEP is the use of common data formats, captured in the SEDE data model. In this way, the tool supporting the next activity in the process can read the data created by the tool supporting the previous activity. For storing the evaluation related information generated throughout the process, the Evaluation Data Structure was developed. This data structure covers all the evaluation activities in the SEDEP process and it is a part of the SEDE data model. Even though only Collective Training is considered in this paper, the Evaluation Data Structure supports also at least Mission Rehearsal and Simulation Based Acquisition application areas. Figure 4 shows the top-level elements of the Evaluation Data Structure.

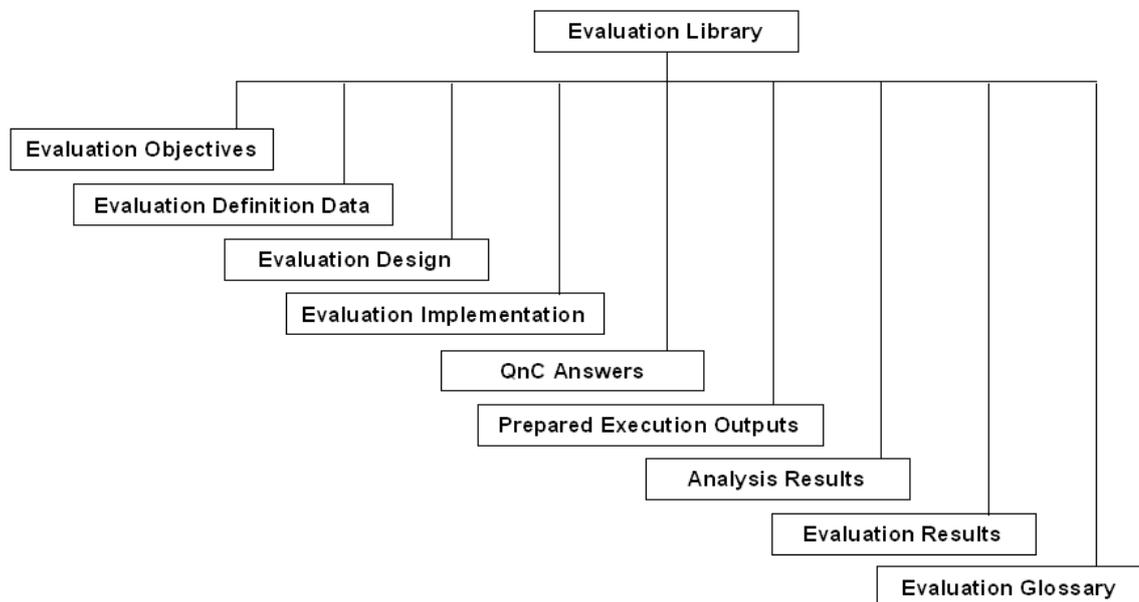


Figure 4: Evaluation Data Structure

The aggregate SEDE data model has means for traceability of requirements and identification of dependencies to other SE information. This makes the dependency of for example the evaluation definition on a scenario item visible. Besides that it is possible to trace back from the evaluation results to the evaluation objectives by going from Evaluation Results to Analysis Results, from Analysis Results to Evaluation Definition and from Evaluation Definition to the Evaluation Objective. The Evaluation Data Structure is seen as an important asset in standardising the evaluation process. The standardisation of the used data format enables the reuse of evaluation data and encourages the Evaluation SMEs to spread their knowledge throughout the SE community.

The data structure is implemented in eXtensible Mark-up Language (XML) language that allows for the definition of actual data formats/data structures. By defining an open XML based



data structure the independency and interoperability of the used tools is achieved. The evaluator or analyst can use his own familiar browser or XML editor to edit and access evaluation data. It is also possible to develop tailored tools for representing and processing the evaluating results as well as using the prototype evaluation tool set developed within the RTP11.13.

## 6 Evaluation Definition Tool and Evaluation Definition Selection Tool

A lot of effort was used in getting to grips with the first activities of evaluation, i.e. the gathering of evaluation objectives and the definition of evaluation. To test the ideas and to give the evaluation SMEs the possibility to go into practise in these issues two prototype tools were developed: the Evaluation Definition Tool (EDT) and the Evaluation Definition Selection Tool (EDST). The EDT supports the evaluation SMEs and system engineers in collecting the user's objectives and defining the evaluation related aspects on basis of the objectives. The EDT comprises two editors:

- Evaluation Objective Editor for defining Evaluation Objectives
- Evaluation Knowledge Editor for deriving Evaluation Definitions

The EDT Framework is the basic component of the EDT. It provides a frame for and works as an interface between the EDT and external tools. The EDT has external interfaces for the following tools (see Figure 5):

- Evaluation Definition Selection Tool (EDST)
- EUCLID RTP11.13 Repository

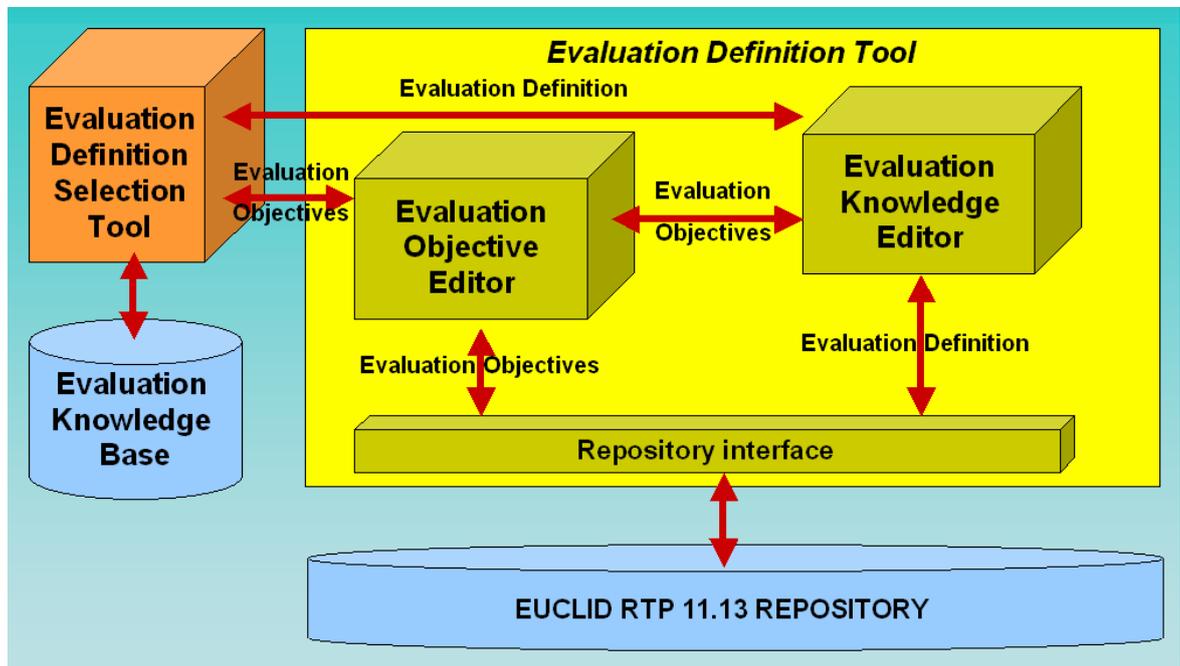


Figure 5: Evaluation Definition Tool architecture

The Objective Editor is a tool used for gathering the Evaluation Objectives. The objectives form a hierarchical tree-like structure. This structure allows to clearly present even a large number of objectives and to arrange the objectives by topic. The objective creation starts from the



highest-level objective and is then elaborated in further detail until the objectives are specific enough to start the Evaluation Definition. The Evaluation Objectives are referenced to User Goals, which are an outcome of SEDEP step 0.

In the Evaluation Knowledge Editor (EK-Editor), the user is able to define the Criteria, Algorithms and Measures needed to assess the exercise according to the Evaluation Objectives. The EK-Editor also supports the use of questionnaires in the Evaluation. In the EK-Editor the user builds up so called Evaluation Definition Networks to represent the definitions. An Evaluation Definition Network consists of Inputs, Outputs and Assessment Nodes, which define the dependencies between Inputs and Outputs. The visualisation of the definition as a network eases the construction of the Evaluation Definitions by structuring the definition and enabling the reuse of network nodes (i.e. Algorithms and Measures). The Assessment Nodes represent the evaluation methods. Examples of evaluation methods are mathematical algorithms. The definitions of these may be linked to elements in the scenario, which are defined for the evaluated exercise. Criteria will be determined (according to the objectives) that are used to judge the quality of the collected data. In this case quality is related to the performance of the evaluated object and not e.g. to the data completeness. These evaluation criteria are also defined using the EK-editor.

The EDT also provides access to a pool of ready-made objectives and definitions through the EDST. The GUI of the EDST is presented in Figure 6. In the EDST it is possible to search for and select suitable evaluation objectives from a pool of ready-made objectives. On the basis of the selected objectives, the EDST searches related objectives. Artificial intelligence is used to search algorithms in an effective way. The objectives and definitions generated by the EDST are provided to the EDT, where they are attached to the existing data and can be edited if necessary. The EDST has its own knowledge base for utilising the searches. Naturally the EDST has facilities for inputting and editing the data in the knowledge base.

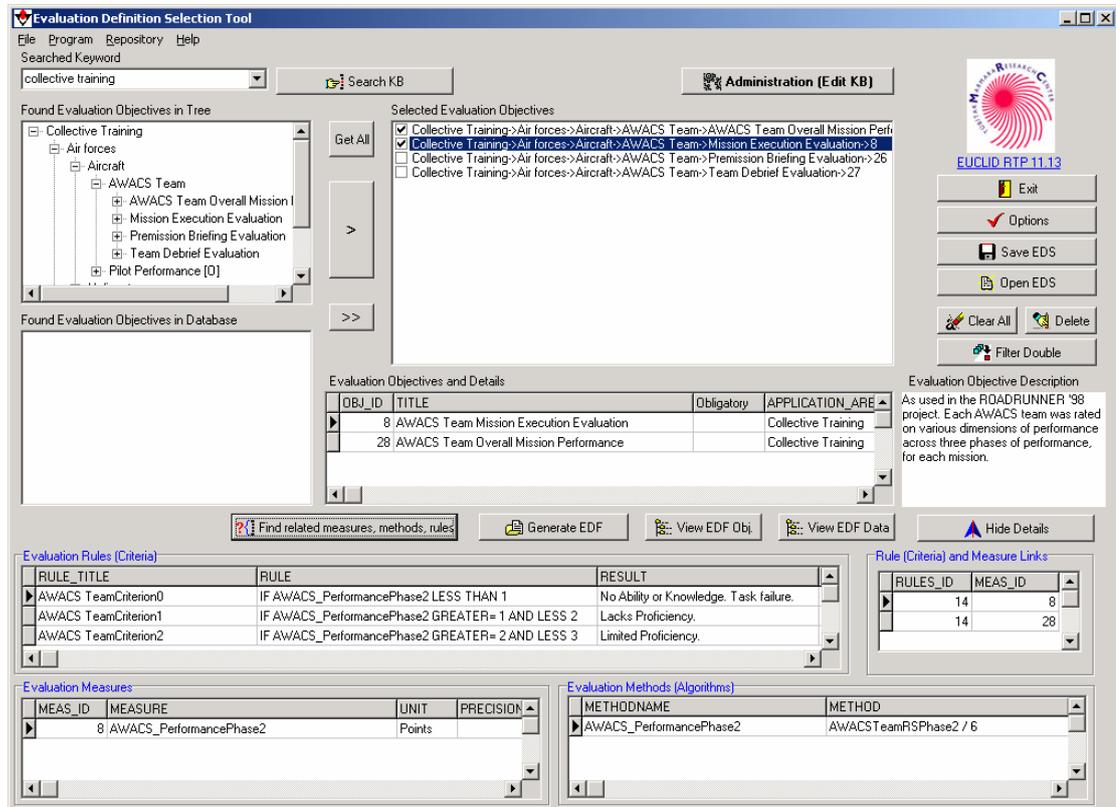


Figure 6: EDST

The EDT may be used as an integrated part of the Synthetic Environment Development Environment (SEDE) or as a standalone tool. Using the EDT as part of the SEDE means, that the EDT

- Has access to the RTP 11.13 Repository
- Can be launched from the Synthetic Environment Management Tool (SEMT)
- SEMT is an overarching tool developed for managing SE projects
- Stores data using the data format defined in the Evaluation Data Structure

When using the EDT as standalone tool the RTP11.13 Repository can not be accessed and thus it is not possible to link the evaluation information to neither user goals nor scenario. In the stand-alone mode the evaluation information is stored in the computer's local file system still using the Evaluation Data Structure.



## 7 Execution Evaluation Tool

The Execution Evaluation Tool (EET) supports the user in post-processing the outputs acquired during the CT execution and in the analysis and evaluation of the results. This tool uses prepared execution outputs to apply the evaluation algorithms and criteria. The intention of the EET is to provide information on one hand needed to generate process feedback and corrective actions to improve the design and development of an SE and on the other hand needed to produce useful results for the evaluator to assess the trainees. The EET provides the user with structured analysis results for evaluation instead of lots of unstructured execution outputs. The EET provides the user with data in formats, which can be imported directly into documents or presentations.

The EET uses the evaluation definition (produced by the Evaluation Definition Tool) and the execution outputs from the EUCLID RTP 11.13 Repository. It evaluates the CT exercise by processing the prepared execution outputs using the evaluation algorithms. The EET is in the same way as the EDT part of the SEDE. The EET is able to analyse the complete evaluation, fully automatic if the user wants to, but also gives the user the option to execute only a subset of evaluation algorithms, and, depending on the options per algorithm, the user may choose to adjust some of the algorithm's parameters. It may also be possible that a specific algorithm is available for interactive analysis. In that case the user will be able to interactively adjust (some of) the algorithm parameters and view the effect on the results immediately.

In Figure 7 the architecture of the EET is shown. A general evaluation may lead to the necessity for multiple commercial tools, because there is no single tool that can process all data. The tools 1, 2, and 3 in the figure may represent different tools from different vendors or the same tool (e.g. Matlab) used with different requirements on available tool boxes. To the user the EET will be presented as one tool, and the internal structure ( e.g. the fact that there are multiple COTS tools at work) is hidden (the yellow box is surrounding all other components, including the COTS tools).

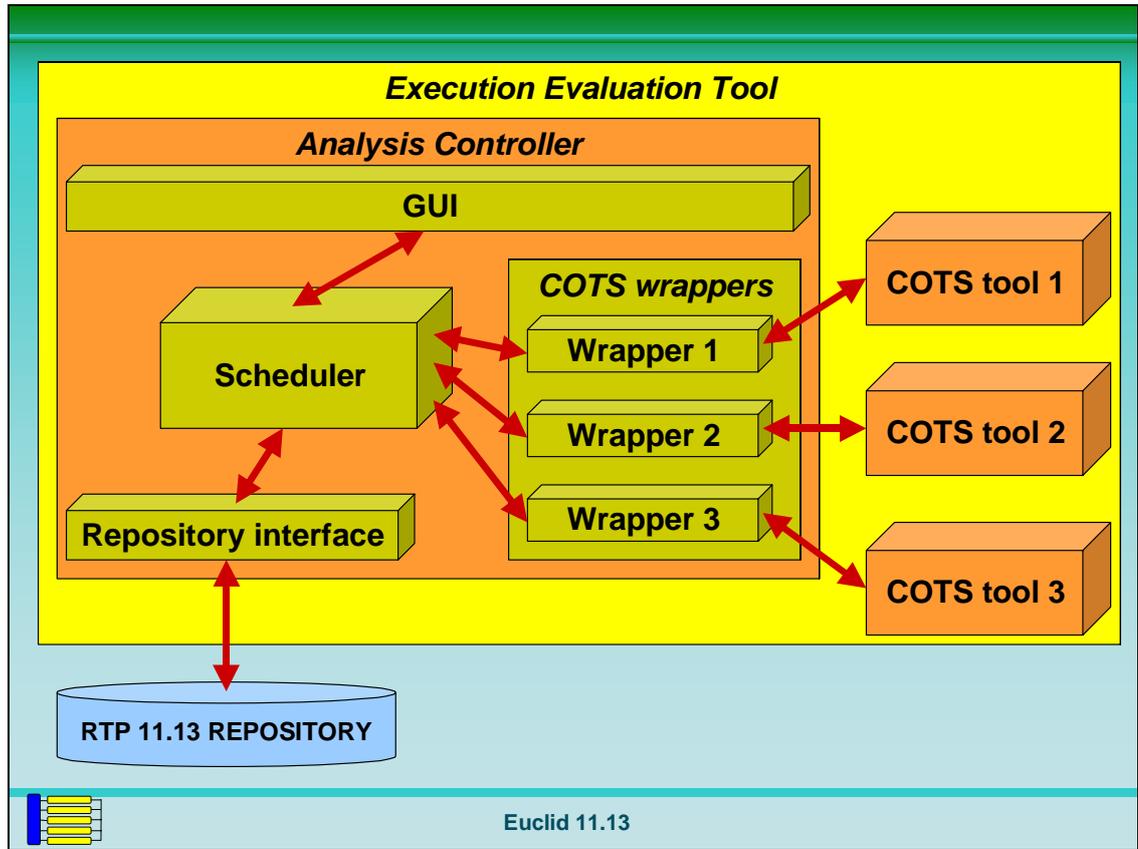


Figure 7: EET Architecture



## 8 Conclusions

It is well known that in the training domain, individual skills are taught much more effectively using tailor-made training devices. The main advantage of SEs is in training certain collective skills. The distributed, group nature of Collective Training makes any measurement of performance difficult, with the level of difficulty increasing as the size of the unit under training increases. Currently only basic, usually subjective, forms of measurement exist at the sub-unit level and above. The EUCLID RTP11.13 team has identified this fact as a major obstacle for realising the potential of using Synthetic Environments (SEs) for CT purposes. The process of setting up SEs for CT needs to be refined further in order to derive more robust CT metrics, which could be used in live or synthetic exercises.

For this process the Federation Development and Execution Process (FEDEP) has been used as a baseline, but it has been found lacking on evaluation issues. One of the shortcomings was that the FEDEP does not cover the complete lifecycle of a SE. Therefore the FEDEP has been modified and extended with, among other things, a new step "Perform Evaluation". The resulting RTP 11.13 process is known as the Synthetic Environment Development & Exploitation Process (SEDEP). Discussions have taken place with the FEDEP Productisation Group, resulting to a total of 16 changes which are now included in the IEEE 1516.3 FEDEP version. The most important contribution of EUCLID RTP 11.13 is the new FEDEP step 7 "Analyse Data and Evaluate Results".

The reuse of SE components, specifications, and definitions is a key factor to reduce the cost and time scale of creating and utilising SEs. To enable and facilitate the reuse of evaluation data EUCLID RTP 11.13 has defined common data formats, captured in the SE Development Environment (SEDE) data model. In this way, the data created by one tool can be read by the tool supporting the next activity in the process. All tools will have access to the RTP11.13 Repository, which will provide the mechanism for transferring data between them. The Data Interchange Formats (DIFs) defined by the HLA are being extended to other areas supported by the SEDEP. The data accessed by the SEDE can be from a local repository or from one that has been distributed over a Wide Area Network (WAN).

COTS tools are already available for supporting many of the SEDEP activities and three additional tools have been prototyped to support the evaluation related activities: the Evaluation Definition Tool (EDT), the Evaluation Definition Selection Tool (EDST) and the Execution Evaluation Tool (EET). The EDT supports the evaluation SMEs and system engineers in collecting the user's objectives and defining the evaluation related aspects on basis of the objectives. The EDT also provides via the EDST access to a pool of ready-made objectives and



definitions. The EDST makes it possible to search for and select suitable evaluation objectives from a pool of ready-made objectives. The EET supports the user in post-processing the outputs acquired during the CT execution and in the analysis and evaluation of the results. The intention of the EET is to provide information needed on one hand to generate process feedback and corrective actions to improve the design and development of an SE and on the other hand needed to produce useful results for the evaluator to assess the trainees.



## References

- [1] Training in a synthetic environment for improved operational effectiveness in collective air operations, Heather M. McIntyre & Ebb Smith, Defence Evaluation & Research Agency, I/ITSEC, November 2000, Orlando
- [2] Handbook of Simulator-based Training, Eric Farmer et al., Ashgate Publishing Ltd, 1999, Aldershot, UK
- [3] The Deliberate Application of Principles of Learning and Training Strategies within DMT, Winston Bennett Jr. and Peter Crane, NATO SAS-038 MTDS Symposium, April 2002, Brussels
- [4] High Level Architecture Federation Development and Execution Process (FEDEP) Model, Version 1.5, Defense Modeling and Simulation Office, December 1999
- [5] The EUCLID RTP 11.13 Synthetic Environment Development & Exploitation Process, K. J. Ford, Fall SIW, Paper 01F-SIW-124, September 2001, Orlando