The Multi-Relationship Evaluation Design Framework: Producing Evaluation Blueprints to Test Emerging, Advanced, and Intelligent Technologies

Brian A. Weiss
National Institute of Standards and Technology
100 Bureau Drive MS 8230
Gaithersburg, Maryland 20899 USA
brian.weiss@nist.gov
Phone: 301.975.4373
Fax: 301.990.9688

Linda C. Schmidt
University of Maryland
0162 Glenn L. Martin Hall, Building 088
College Park, Maryland 20742-3035
lschmidt@umd.edu
Phone: 301.405.0417
Fax: 301.314.9477

The Multi-Relationship Evaluation Design Framework: Producing Evaluation Blueprints to Test Emerging, Advanced and Intelligent Technologies

Brian A. Weiss
National Institute of Standards and Technology
brian.weiss@nist.gov

Linda C. Schmidt
University of Maryland
lschmidt@umd.edu

Abstract

This paper introduces the Multi-Relationship Evaluation Design (MRED) framework whose objective is to take uncertain input data and automatically output comprehensive evaluation blueprints complete with targeted evaluation elements. MRED is unique in that it characterizes the relationships among the evaluation elements and will address their uncertainties in addition to those tied to the evaluation input. These terms and their relationships will be applied in an example evaluation design of an emerging technology.

1. Introduction

Intelligent system advances are continuously occurring across a range of fields in the military, automobile and manufacturing communities. As these new technologies emerge, it becomes critical to evaluate their performance to both (1) inform the technology creators of deficiencies and obtain end-user feedback so that enhancements can be made and (2) validate the technology's ultimate capabilities so that buyers and system users know exact system capabilities. The former takes place in formative evaluations where adjustments and enhancements can be made in upcoming designs; the latter happens in summative evaluations to enable buyers and technology users to see the extent of the technology's capabilities. These two types of evaluations can be designed to comprise a few simplistic tests of key capabilities of the overall technology. Alternatively, evaluations can become highly complex events, testing numerous components and capabilities along with the entire system. Typically, tests of advanced and intelligent systems tend to be elaborate since the technologies, themselves, are usually complex in nature.

The National Institute of Standards and Technology (NIST) created the System, Component, and Operationally-Relevant Evaluation (SCORE) framework to evaluate numerous emerging and intelligent systems at various levels (Weiss and Schlenoff, 2008). Specifically, SCORE provides a set of guidelines to aid test designers in creating evaluation plans. SCORE has been effectively applied to fifteen evaluations across several technologies (Schlenoff et al., 2009; Weiss and Schlenoff, 2009). SCORE-enabled tests have yielded extensive quantitative and qualitative data. SCORE has proven to be valuable to technology developers, evaluation designers, potential end-users, and funding sponsors. The research described here draws upon that success to introduce a new evaluation framework that will automatically generate evaluation blueprints or "test plans". This new evaluation design tool is known as the Multi-Relationship Evaluation Design (MRED) framework. MRED's ultimate objective is to take inputs from specific groups, each complete with their own varying uncertainties, and output an evaluation blueprint that specifies all characteristics of the tests. MRED is a work in progress where the authors are identifying an adequate set of blueprint constituent elements and their interrelationships.

The overall model encompassing the MRED framework was introduced in Weiss et al., 2010 along with several evaluation blueprint elements. Since that initial work, further blueprint elements have been defined (Weiss and Schmidt, 2010). Section 2 introduces the International Test and Evaluation Association (ITEA) community to the blueprint elements identified to date, the relationships among them and their major interdependencies. In Section 3, the MRED model will be applied to the evaluation design of pedestrian and object tracking algorithms whose test plans were created without the use of any formal evaluation design framework.

2. MRED

The Multi-Relationship Evaluation Design (MRED) framework resides within a model that contains the significant design inputs into the planner and the output "evaluation blueprint." The overall model is shown in Figure 1.

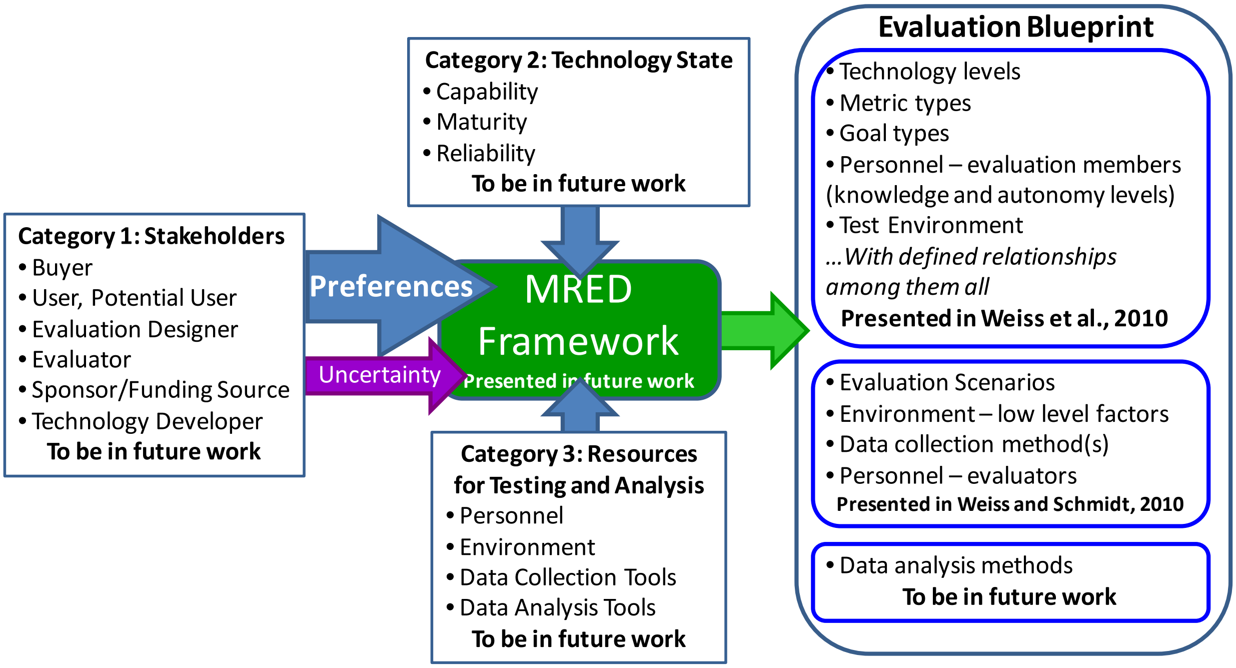


Figure 1 - Evaluation Design Model Surrounding the MRED Framework Including Inputs and Outputs

The following subsections will present the evaluation blueprints that have been defined in Weiss et al., 2010 and Weiss and Schmidt, 2010. The three input categories: (1) Stakeholders, (2) Technology State, and (3) Resources for Testing and Analysis will be highlighted to better understand the model.

2.1 Input Categories

The MRED model recognizes three crucial input categories or personnel groups interested in a technology's evaluation. Each category will be briefly described in the following subsections.

2.1.1 *Category 1 – Stakeholders*

Evaluation stakeholders are organized into six categories of parties interested in the technology's evaluation. Most (if not all) of the stakeholders, are active in a technology's evaluation design. Each stakeholder's preferences regarding the test plan may evolve over time which leads to uncertainty based upon these changing preferences.

Table 1 - Technology Evaluation Stakeholder Groups (Weiss and Schmidt, 2010)

STAKEHOLDER GROUPS	WHO THEY ARE...
Buyers	Stakeholder purchasing the technology
Users, Potential Users	Stakeholder that will be or are already using the technology
Evaluation Designers	Stakeholder creating the test plans by determining MRED inputs
Evaluators	Stakeholder implementing the evaluation test plans
Sponsors/Funding Sources	Stakeholder paying for the technology development and/or evaluation
Technology Developers	Stakeholder designing and building the technology

There may be some overlap among the stakeholders where the range of potential relationships among the stakeholders can be found in Weiss et al., 2010.

2.1.2 *Category 2 – Technology State Factors*

This category contains the factors that characterize the system's state at the time of its test. These factors are presented in Table 2. Note that all three of these factors may change between the time(s) when testing is first discussed and planned to the moment when the test(s) are executed.

Table 2 - Technology State Factors

FACTORS	WHAT IS IT...	WHY IT MATTERS...
Reliability	Technology's ability to yield the same or compatible results in previous test(s).	Changes will impact test comparisons and the technology has undergone previous testing where the output test data has been used to iterate upon the design.
Capability	Technology's ability to be evaluated under certain conditions and/or used in a specific functionality(ies).	Considers if the technology is robust enough or if its current level of development is sufficient to undergo a specific test(s).
Maturity	Technology's state or quality of being fully developed.	Considers if the technology is equipped with all of its intended functionality or if only a subset of expected features is operational at the time of testing.

The level of these factors determines whether or not the testing data can be used for formative or summative evaluations (Weiss and Schmidt, 2010).

2.1.3 *Category 3 – Resources for Testing and Analysis*

This last input group is composed of various types of material, personnel and technology to be to be committed to the evaluation exercise and data analysis. Resource availability (or lack thereof) and limitations can have a tremendous influence on the final evaluation design. These resources are highlighted in Table 3.

Table 3 - Resources of Testing and Analysis

RESOURCES	DESCRIPTION
Personnel	Individuals that will use the technology, those that will indirectly interact with the technology, those that will collect data during the test, and those that will analyze the data following the test(s).
Test Environment	The physical venue, supporting infrastructure, artifacts and props that will support the test(s).
Data Collection Tools	The tools, equipment, and technology that will collect quantitative and/or qualitative data during the test(s).
Data Analysis Tools	The tools, equipment, and technology capable of producing the necessary metrics from the collected evaluation data.

2.2 Blueprint Outputs

The blueprint outputs highlighted in the following subsections have been presented in detail in prior work including Weiss et al., 2010; Weiss and Schmidt, 2010; and Weiss and Schlenoff, 2008.

2.2.1 *Technology Levels*

A technology or system is made up of constituent components and then can be evaluated at multiple levels. There are several terms related to technology levels and are defined as follows:

- *System* – Group of cooperative or interdependent *Components* forming an integrated whole intended to accomplish a specific goal.
- *Component* – Essential part or feature of a *System* that contributes to the *System's* ability to accomplish a goal(s).
- *Sub-Component* – Element, part or feature of a *Component.*
- *Capability* – A specific ability of a technology where a *System* is made up of one or more *Capabilities*. A *Capability* is provided by either a single *Component* or multiple *Components* working together.

2.2.2 *Metric Types*

There are two metric types:

- *Technical Performance* – Metrics related to quantitative factors (such as accuracy, precision, time, distance, etc). These metrics may be needed by the program *Sponsor* to get a status of the technology's current performance, update the *Technology Developers* on their design, etc.
- *Utility Assessments* – Metrics related to the qualitative factors that gauge the condition or status of being useful and usable to the target user population. Like *Technical Performance*, these metrics may be of value to any and/or all of the stakeholders.

2.2.3 *Goal Types*

Goal types are combinations of technology levels and desired metrics (Schlenoff et al., 2009; Weiss et al., 2008). There are five goal types that will be output from the MRED framework including three that capture quantitative technical performance data and two that capture qualitative utility assessments (shown in Figure 2).

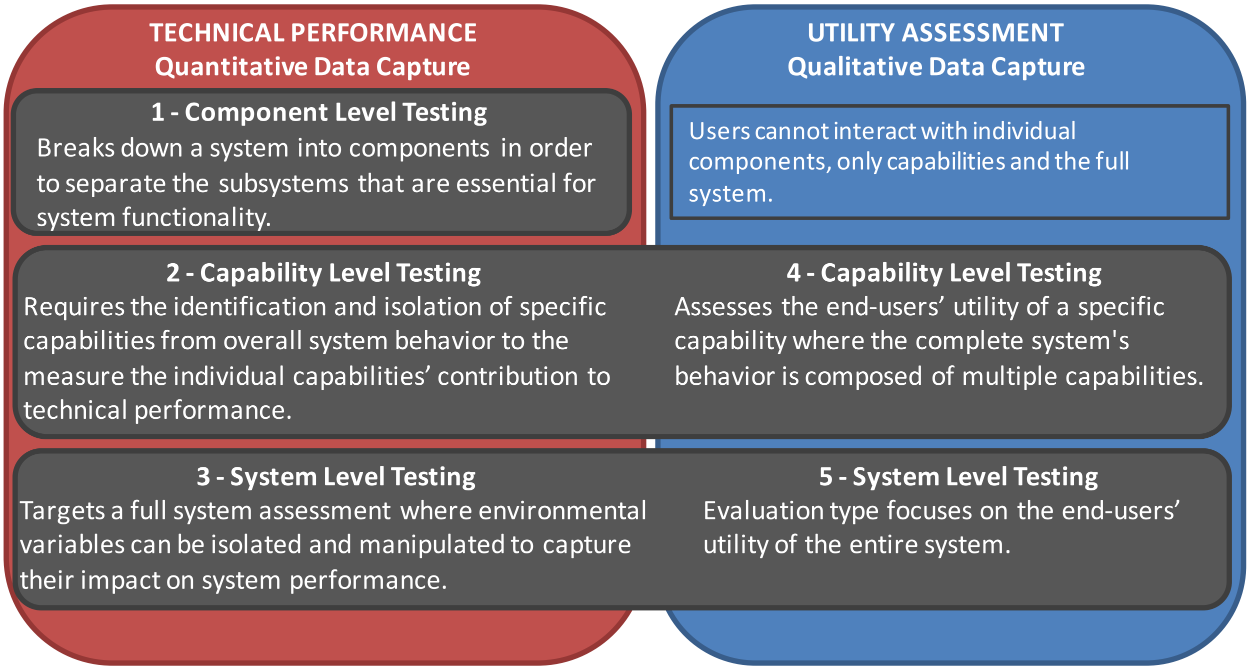

TECHNICAL PERFORMANCE
Quantitative Data Capture
1 - Component Level Testing
Breaks down a system into components in order to separate the subsystems that are essential for system functionality.
2 - Capability Level Testing
Requires the identification and isolation of specific capabilities from overall system behavior to the measure the individual capabilities' contribution to technical performance.
3 - System Level Testing
Targets a full system assessment where environmental variables can be isolated and manipulated to capture their impact on system performance.
UTILITY ASSESSMENT
Qualitative Data Capture
Users cannot interact with individual components, only capabilities and the full system.
4 - Capability Level Testing
Assesses the end-users' utility of a specific capability where the complete system's behavior is composed of multiple capabilities.
5 - System Level Testing
Evaluation type focuses on the end-users' utility of the entire system.

Figure 2 - MRED Framework Output Goal Types

Each of these goal types require different features and properties within an evaluation making it possible to create a test plan that captures data to satisfy multiple goal types. This is another unique feature of MRED in that it will support the generation of a set of test plans to capture a range of goal type data based upon the input.

MRED will be capable of outputting a set of blueprints, each able to capture data for a different goal type. Evidence of MRED's capability in creating evaluation plan blueprints is provided in Section 3 by an initial application of the model to a speech-to-speech technology. MRED's blueprints will be compared to a previous whose evaluation was driven by the SCORE framework (Weiss et al., 2010).

2.2.4 *Personnel – Evaluation Members*

Numerous individuals and groups are necessary to produce an effective evaluation. They are classified into two distinct categories: primary (direct interaction) technology users and secondary (indirect interaction or evaluation support). The primary technology users are identified as *Tech Users*. These individuals directly interact with the technology during the evaluation. They receive any training necessary to use the technology and are responsible for engaging/disengaging the technology's usage during the test event. There are multiple classes of *Tech Users* as shown in Table 4. Table 4 also presents in which *Goal Types* the various *Tech Users* may participate given their characteristics. Note that the information provided in this table is intended to demonstrate the superset of possibilities given that all tests are technology-dependent.

Table 4 - Primary Evaluation Personnel Using the Technology During Testing

			APPLICABLE GOAL TYPES For PARTICIPATION				
			TECHNICAL PERFORMANCE			**UTILITY ASSESSMENT**	
		DESCRIPTION	Component	Capability	System	Capability	System
PRIMARY PERSONNEL	*Tech User: End-User*	Individuals that are the intended users for the technology.	NO	YES	YES	YES	YES
	Tech User: Trained User	Individuals selected to be *Tech Users* , yet are not *End-Users* .	YES	YES	YES	YES	YES
	Tech User: Tech Developer	Members of the research and development organization(s) that developed the technology under evaluation.	YES	YES	YES	NO	NO

Secondary personnel are those that indirectly interact with the technology during the evaluation and fall into three categories:

- Team Member – Individuals that work with *Tech Users* during the evaluation as they would to realistically support the use-case scenario that the technology is immersed. *Team Members* may or may not be in a position to indirectly or directly interact with the technology during the evaluation, but they are often in a position to observe a *Tech User's* interactions with the system.
- Participant – Individuals that indirectly interact with the technology during an evaluation. Typically, *Participants* are given specific tasks to either interact with the *Tech Users* and/or with the environment, but not with the technology.
- Evaluator – Members of the evaluation team present within the *Test Environment* that task the *Participants* and/or captures data, but does not interact with the technology. Depending upon the test, the *Evaluator* may interact with the *Tech User* to capture data.

The relationships among these primary and secondary technology user groups are presented in greater detail in Weiss et al., 2010.

Significant relationships exist between *Technology Levels, Metric Types, Tech Users* and *Test Environments* and they were highlighted in previous efforts (Weiss et al., 2010). Likewise, relationships between *Personnel – Evaluation Members, Knowledge Levels,* and *Autonomy Levels* are documented in this paper. These relationships define numerous constraints that must be satisfied in any feasible evaluation blueprint.

Knowledge Levels The *Tech Users, Team Members,* and *Participants* involved in the evaluation have various levels of knowledge about aspects of the system and testing conditions within two specific areas. The levels are defined as:

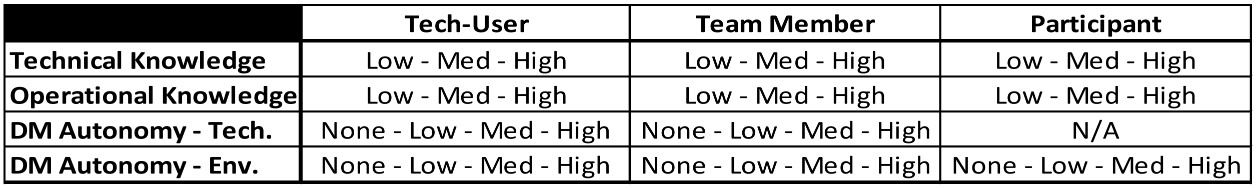
- *Operational Knowledge* – The level of practical information and experience an individual has about the *Actual* environment, the intended use-case situations for the technology and other pre-existing technologies that the technology under test leverages and/or supports. Varying levels of *Operational Knowledge* can be attained through real-world experience, repetitive training, trial and error exercises, etc.
- *Technical Knowledge* – The level of information and experience an individual has about the technology itself and how it should be employed to maximize success.

Autonomy Levels Additionally, the *Tech Users* and *Participants* within the evaluation have a range of Decision-Making (DM) autonomy. Autonomy scope and their levels are set by MRED for each evaluation. Personnel could be fully restricted in their decision-making (i.e., no DM Autonomy), which requires scripted actions. Alternatively, personnel may have unbounded decision-making authority where each participant is free to exercise their judgment given their various knowledge levels. Specifically, there are two types of *DM Autonomy* which are defined below:

- *DM Autonomy – Technical* – This refers to the level of authority that the *Tech Users* have in operating the technology. Depending upon the specific evaluation, *Tech Users* could be instructed to only use certain features of a technology to being told that they may use any or all of its features as they see fit.
- *DM Autonomy – Environmental* – This refers to the level of authority that the *Tech Users* and Participants have in interacting with each other and the environment.

Autonomy Levels must be equal or lower in value than their partner *Knowledge Levels*. Determination of *Autonomy Levels* is governed by multiple factors including evaluation *Goal Types, Tech User* class, etc. The potential knowledge and autonomy levels for the evaluation participants are shown in Table 5.

Table 5 - Relationships among Personnel, Knowledge, and Autonomy Levels (Weiss et al., 2010)



	Tech-User	**Team Member**	**Participant**
Technical Knowledge	Low - Med - High	Low - Med - High	Low - Med - High
Operational Knowledge	Low - Med - High	Low - Med - High	Low - Med - High
DM Autonomy - Tech.	None - Low - Med - High	None - Low - Med - High	N/A
DM Autonomy - Env.	None - Low - Med - High	None - Low - Med - High	None - Low - Med - High

2.2.5 *Test Environments*

The setting in which the evaluation occurs can have a significant effect on the data since the environment can influence the behavior of the personnel and can limit which levels of a technology can be tested. MRED defines three distinct environments:

- *Lab* – Controlled environment where test variables and parameters can be isolated and manipulated to determine how they impact system performance and/or the Tech Users' perception of the technology's utility.
- *Simulated* – Environment outside of the Lab that is less controlled and limits the evaluation team's ability to control influencing variables and parameters since it tests the technology in a more realistic venue.
- *Actual* – Domain of operations that the system is designed to be used. The evaluation team is limited in the data they can collect since they cannot control environmental variables.

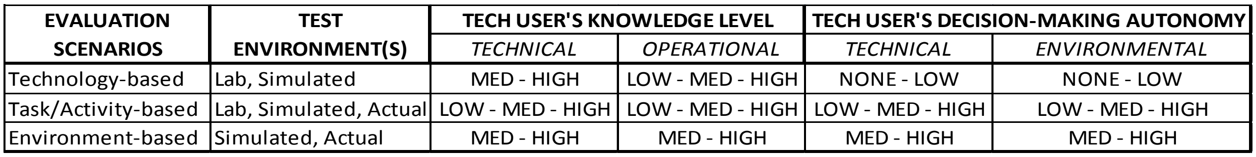
2.2.6 *Evaluation Scenarios*

The Evaluation Scenarios govern exactly what the technology will encounter and the challenges it will have to perform within the identified *Test Environments*. Three types of *Evaluation Scenarios* are identified below. Each is unique in the relationships they have with *Tech User: Know-*

ledge Levels, *Tech User: Decision-Making Autonomy*, and *Test Environments*. The three *Evaluation Scenario* types are listed below while the relationships are shown in Table 6.

- *Technology-based* – Evaluation scenarios in this category feature specific instructions to the user in how they should use the technology within the testing environment.
- *Task/Activity-based* – Evaluation scenarios in this category state the user complete a specific task within the environment where they may use the technology as they see fit.
- *Environment-based* – Evaluation scenarios in this category enable the user to perform the relevant activities within the environment based upon an advanced *Operational Knowledge*.

Table 6 - Relationship Among the Scenarios, Environments, Knowledge and Decision-making Autonomy (Weiss and Schmidt, 2010)



EVALUATION SCENARIOS	TEST ENVIRONMENT(S)	TECH USER'S KNOWLEDGE LEVEL		TECH USER'S DECISION-MAKING AUTONOMY	
		TECHNICAL	*OPERATIONAL*	*TECHNICAL*	*ENVIRONMENTAL*
Technology-based	Lab, Simulated	MED - HIGH	LOW - MED - HIGH	NONE - LOW	NONE - LOW
Task/Activity-based	Lab, Simulated, Actual	LOW - MED - HIGH	LOW - MED - HIGH	LOW - MED - HIGH	LOW - MED - HIGH
Environment-based	Simulated, Actual	MED - HIGH	MED - HIGH	MED - HIGH	MED - HIGH

Further details can be found in Weiss and Schmidt, 2010.

2.2.7 *Explicit Environmental Factors*

The *Explicit Environmental Factors* are significant characteristics within the environment that impact the technology, thereby influencing the outcome of the evaluation. These factors pertain to the overall physical space (e.g., *Participants*, structures, and any integrated props and artifacts). These factors are broken down into two constituent characteristics, *Feature Density* and *Feature Complexity*. These two characteristics combine to form the *Overall Complexity*.

- *Feature Density* – Refers to the amount of features within the Test Environment given the size of the test area. The greater the Feature Density, the more challenging it is for a technology to effectively and efficiently interact with, identify objects/events/activities, operate within, etc the *Test Environment*.
- *Feature Complexity* – Refers to the complexity of various measurable features within the environment. Similar to *Feature Density*, the greater the *Feature Complexity*, the more difficult it is for the technology to accurately and appropriate operate and be beneficial to the *Tech User(s).*
- *Overall Complexity* – This factor refers to the global influence of *Feature Density* and *Feature Complexity* within the testing environment.

Figure 3 presents the relationships among the *Test Environment* and the *Explicit Environmental Factors*. Additional information on this blueprint output and these relationships can be found in Weiss and Schmidt, 2010.

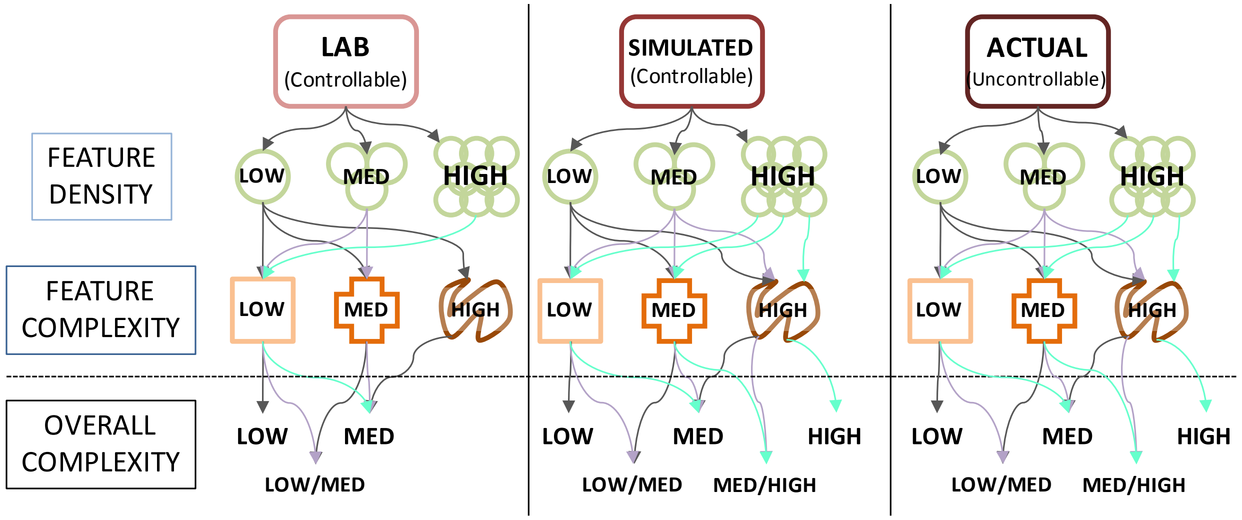


Figure 3 - Relationships between the Test Environment and its Explicit Environmental Factors (Weiss and Schmidt, 2010)

It is currently expected that the *Overall Complexity* will be limited in the *Lab* given the nature of this highly-controlled environment.

2.2.8 *Data Collection Methods*

Data Collection Methods are used to capture experimental and ground truth data depending upon the technology being evaluated and the specified *Test Environment*. No matter the type of tools used, *Data Collection Methods* are characterized by factors that influence the techniques being employed. These two factors include *Mode* and *Collector Location* and are presented below:

- *Mode* – This factor refers to the nature of the *Data Collection Method* that will be employed. Specifically, there are two different types of *Modes* that can collect data. *Automated Modes* involves collecting data with a calibrated technology that is independent of the system undergoing testing. *Manual Modes* features an *Evaluator* actively managing a calibrated technology or collecting data by hand.
- *Collector Location* – This factor refers to the placement of the *Mode* relative to the technology under test.

The various *Collector Location* and some examples over the two types of *Modes* are presented in Table 7.

Table 7 - Example Data Collection Methods from various Collector Locations and Modes (Weiss and Schmidt, 2010)

	DATA COLLECTION METHODS: Examples	
COLLECTOR LOCATION	*MODES - AUTOMATED*	*MODES - MANUAL*
From the Technology	Sensor and/or tool collecting technical information during the evaluation	Evaluator capturing data with sensors, tools
	Output of log files following the evaluation	Evaluator making notes of behavior
From the *Tech User*	Sensors (e.g. helmet camera) attached to the *Tech User* that collect data during testing	Surveys prior to and/or following the evaluation(s)
		Interviews prior to and/or following the evaluation
		Verbal and/or physical feedback provided by the *Tech User* during the evaluation (e.g. thumbs-up or thumbs-down at key way points)
From the *Team Member*	Sensors (e.g. microphone) attached to the *Team Member* that collect data during testing	Surveys prior to and/or following the evaluation(s)
		Interviews prior to and/or following the evaluation
		Verbal and/or physical feedback provided by the *Tech User* during the evaluation (e.g. thumbs-up or thumbs-down at key way points)
From the *Participant*	NONE (usually)	Surveys prior to and/or following the evaluation(s)
		Interviews prior to and/or following the evaluation
From a Different Perspective	Sensors (e.g. radar gun, thermal camera, motion detector) setup throughout the environment that collect data during testing	Evaluation personnel stationed in various parts of the environment taking notes and/or manually using a sensor and/or tool to collect data

Greater detail regarding *Data Collection Methods* can be found in Weiss and Schmidt, 2010.

2.2.9 *Personnel – Evaluators*

There are three classes of evaluation personnel that are necessary to ensure that the evaluation proceeds accordingly to plan and that the necessary data is captured to evaluate a technology's performance. They fall into the three classes below:

- *Evaluators: Data Collectors* – These *Evaluators* are responsible for either setting up/implementing automated collection methods and/or performing manual collection methods. This class of *Evaluators* is also responsible for collecting experimental data directly from the technology at the conclusion of each test scenario (as necessary).
- *Evaluators: Test Executors* – These *Evaluators* are responsible for initiating the test including instructing Participants on when to engage in their specified activities within the environment.
- *Evaluators: Safety Officers* – These *Evaluators* are solely responsible for ensuring the safety of all personnel within the *Test Environment* along with protecting the technology and the environment, itself.

Greater discussion of these personnel classes can be found in Weiss and Schmidt, 2010.

3. Application of MRED

In this section, the model's currently-defined blueprint elements and relationships are applied to a technology that NIST personnel are evaluating. NIST and members of the Army Research Laboratory's (ARL) Collaborative Technology Alliance (CTA) are currently testing multiple pedestrian tracking algorithms whose evaluation design and implementation are conducted jointly by NIST and CTA (Bodt et al., 2009). Each tracking algorithm uses Laser Detection and Ranging (LADAR) and video sensor data taken from a moving vehicle. This test platform moves within the test environment where the vehicle-mounted sensors capture and send data to the on-board detection and tracking algorithms.

The NIST/CTA team have jointly planned and implemented several evaluations from 2007 through 2010. To test MRED and identify any shortcomings, MRED terminology will be used to describe an evaluation blueprint to match the completed testing plans.

3.1 Technology Level, Metric Type, and Goal Type

Presently, the ARL CTA is isolating the pedestrian detection and tracking algorithms for evaluation in a manner that will yield quantitative technical performance metrics. NIST's involvement with the program has centered on conducting field exercises within the *Goal Type* of *Capability Level Testing – Technical Performance*. These field exercises capture the technical data required to assess the performance of the CTA teams' algorithm. They also provide data to support future algorithm development and produce performance analyses. These exercises are conducted by having a sensor-laden vehicle drive in a pedestrian and obstacle-filled environment. The experimental algorithm output can be collected and measured against evaluation team-captured ground truth. Numerous quantitative *Technical Performance Metrics* were produced at the conclusion of these tests. These included true positive, false positive, misclassification, first detection, persistence detection and accuracy of detected position and velocity.

3.2 Personnel – Evaluation Members

For the evaluation type defined above, personnel were noted by the authors for each and are shown in the MRED personnel matrix in Table 8.

Table 8 – MRED Personnel Matrix as Matched ARL CTA Evaluation

	Tech-User: Trained User	Team Member	Participant
Technical Knowledge	Medium	N/A	N/A
Operational Knowledge	Low	N/A	Low
DM Autonomy - Tech.	None	N/A	N/A
DM Autonomy - Env.	None	N/A	Low

Specifically, this blueprint mandated that a *Trained User* engage and disengage the technology during the tests. It's currently premature to identify the exact intended user group since this capability is not fully developed and will be ultimately integrated into a greater system for its regular operations. Algorithms from multiple organizations were tested in parallel and activated by the same *Tech User* to yield an objective approach.

The evaluation participants were individuals that acted as pedestrians. The pedestrians were given a specific path within the environment that they walked during the tests. Practice runs were conducted so that the walkers could determine their pace, better enabling them to complete their path in a prescribed amount of time. The specific *Evaluator* roles will be discussed in section 3.7.

3.3 Test Environment

These tests were performed in *Simulated* environments that featured some Military Operations in Urban Terrain (MOUT) characteristics. The evaluation team controlled the environment during the evaluations enabling them to get very detailed ground-truth data including walking paths, obstacle locations, and vehicle paths.

3.4 Evaluation Scenarios

The *Evaluation Scenarios* designed by NIST/CTA personnel for their test effort can be classified as *Technology-based* when interpreted using the MRED model. The *Tech User* is restricted in how they can interact with the technology meaning they are only allowed to engage the technology at the beginning of a run and disengage it at the run's conclusion. According to MRED's definition of this blueprint element, the CTA *Evaluation Scenarios* are neither *Task/Activity-based* nor *Environment-based* since the *Tech User* has no freedom to interact with the environment or has to complete a specific mission with the technology during the testing.

3.5 Explicit Environmental Factors

Assigning the appropriate MRED model *Explicit Environmental Factors* of this test effort based the test events that the NIST/CTA devised is quite challenging. Since NIST personnel have not been involved in previous algorithm testing that took place in a *Lab* or other *Simulated* environment, it's difficult to ascertain how the *Feature Density*, *Feature Complexity*, and *Overall Complexity* compare in the current test environment. Also, since these algorithms will be incorporated into a larger technology for its envisioned use, it's challenging to anticipate what the *Actual* environment will be, especially since the larger system(s) are somewhat unknown at this point.

The MOUT Simulated Test Environment that the ARL CTA/NIST team previously tested the technology could be classified as having an Overall Complexity of "Medium" where *Feature Density* and *Feature Complexity* were both globally "Medium." However, looking at specific artifacts and personnel activities within the environment, a case could be made that local *Feature Density* ranged from "Low" to "Medium" since multiple personnel were in close proximity to one another in some spots while other personnel stood by themselves in other spots. Comparably, it can be stated that local *Feature Complexity* also ranges from "Low" to "Medium" considering that there were various environmental features present including several rectangular buildings and about a dozen *Participants*.

3.6 Data Collection Methods

In order to test the pedestrian tracking algorithms, the CTA/NIST team deployed numerous *Automated Data Collection Methods,* necessary to attain the required metrics, from numerous *Collector Locations*. Specifically, an Ultra-Wideband (UWB) tracking system is deployed to capture position ground truth data of the test vehicle, key environmental features and pedestrians that are within the testing environment. This was used to capture quantitative *Technical Performance* data of the sensors and algorithms in order to generate the necessary evaluation metrics. Numerous cameras were setup throughout the test environment to collect visual position data of test vehicle, key environmental features and pedestrians. Both the UWB and camera data were used as ground-truth.

The experimental data featured each algorithm reporting detection information such as positions and velocities of the humans the end of each CTA algorithm cycle. The underlying assumptions for the outputs of the algorithms included the following:

- Only obstacles seen and classified as human were reported.
- Unique identification numbers were assigned to individual algorithm detections within a run.
- Algorithms demonstrated tracking of an individual by maintaining the same ID in successive frames.
- Algorithms also reported velocity of the detected humans.

Since the fixed test area was instrumented to capture ground-truth data, all detections were excluded if they occurred outside the test area. The correspondence algorithm found the correspondence between the detections and the ground-truth based on location and time stamp (Bodt et al., 2009). Detections were compared with all the ground-truth objects on the course to attain the desired metrics noted in section 3.1.

3.7 Personnel – Evaluators

The ARL CTA/NIST test effort featured both *Data Collectors* and *Test Executors* conducting the testing. The *Data Collectors* included personnel who deployed and calibrated the UWB tracking system and cameras prior to the test event. These same personnel also managed the UWB tracking system and cameras during the evaluation. Numerous *Test Executors* played a significant role in the evaluation. One individual signaled the start and conclusion of the tests and another individual signaled the pedestrians to walk in their prescribed paths. A third *Test Executor* was employed to signal when the vehicle should begin its motion and when the sensors and algorithms under test should be activated. The evaluation also featured several *Safety Officers* stationed throughout the environment and at key locations along the test environment's perimeter to prevent non-evaluation personnel or vehicles from entering.

4. Conclusion

Section 3 has demonstrated that the MRED blueprints are broad enough to align with technology evaluations designed by NIST/ARL. In the future, additional technology test plans will be generated to further validate its usefulness. The next steps for developing MRED are to finalize the last of the blueprint element definitions and continue to verify the current model against evaluation designs created by NIST personnel. Once MRED's blueprint elements are fully identified, the model will continue to be explored through the detailed definition of the three key evaluation input categories and their interrelationships. Given that each of these input categories is nondeterministic in nature (based upon human preference, unknown technology states, or uncertain resource availabilities), uncertainty will be considered. Once the inputs are clearly stated, the details of the MRED framework, itself, will be outlined and specified. Once proven successful, uncertainty will be factored. It is envisioned that MRED will be an invaluable tool in creating complex evaluation designs of advanced and/or intelligent systems allowing evaluation designers to be more effective and efficient in producing and implementing the appropriate tests.

NIST Disclaimer

Certain commercial companies, products and software are identified in this paper in order to explain our research. Such identification does not imply recommendation or endorsement by NIST, nor does it imply that the companies, products and software identified are necessarily the best available for the purpose.

Acknowledgements

The authors would like to thank Harry Scott, the NIST project leader for the ARL CTA effort, for his continued support throughout this work. Further, the lead author would like to thank Tsai Hong for providing additional information regarding the CTA algorithm output and corresponding metrics. Also the lead author gives thanks to Craig Schlenoff of NIST for his support and encouragement.

References

Barry Bodt, Richard Camden, Harry Scott, Adam Jacoff, Tsai Hong, Tommy Chang, Richard Norcross, Tony Downs, and Ann Virts, 2009, "Performance Measurements for Evaluating Static and Dynamic Multiple Human Detection and Tracking Systems in Unstructured Environments," *Proc. of the 2009 Performance Metrics for Intelligent Systems (PerMIS) Workshop*.

Craig I. Schlenoff, Brian A. Weiss, Michelle P. Steves, Gregory A. Sanders, Fred Proctor, and Ann M. Virts, 2009, "Evaluating Speech Translation Systems: Applying SCORE to TRANSTAC Technologies," *Proc. of the 2009 Performance Metrics for Intelligent Systems (PerMIS) Workshop*.

Brian A. Weiss and Craig I. Schlenoff, 2008, "Evolution of the SCORE Framework to Enhance Field-Based Performance Evaluations of Emerging Technologies," *Proc. of the 2008 Performance Metrics for Intelligent Systems (PerMIS) Workshop*.

Brian A. Weiss, and Craig I. Schlenoff, 2009, "The Impact of Scenario Development on the Performance of Speech Translation Systems Prescribed by the SCORE Framework," *Proc. of the 2009 Performance Metrics for Intelligent Systems (PerMIS) Workshop*.

Brian A. Weiss and Linda C. Schmidt, 2010, "The Multi-Relationship Evaluation Design Framework: Creating Evaluation Blueprints to Assess Advanced and Intelligent Technologies," To appear – *Proc. of the 2010 Performance Metrics for Intelligent Systems (PerMIS) Workshop*.

Brian A. Weiss, Linda C. Schmidt, Harry A. Scott, and Craig I. Schlenoff, 2010, "The Multi-Relationship Evaluation Design Framework: Designing Testing Plans to Comprehensively Assess Advanced and Intelligent Technologies," *Proc. of the ASME 2010 International Design Engineering Technical Conferences (IDETC) – 22nd International Conference on Design Theory and Methodology (DTM).*

Biographies

Brian A. Weiss has been a mechanical engineer at NIST in Maryland since 2002. His focus is the development and implementation of performance metrics to quantify technical performance and assess end-user utility of intelligent systems throughout various stages of development. He has a Bachelors of Science in Mechanical Engineering from the University of Maryland, a Professional Master of Engineering from the University of Maryland and is working towards his Doctor of Philosophy in Mechanical Engineering with the University of Maryland.

Dr. Linda C. Schmidt is an Associate Professor at the University of Maryland. She holds a doctorate in Mechanical Engineering from Carnegie Mellon University and B.S. and M.S. degrees in Industrial Engineering from Iowa State University. Schmidt is active in teaching design research in theory and practice. Schmidt also co-authored texts on engineering decision-making and product development.