AN INTEGRATED VIEW OF TEST AND DIAGNOSTIC INFORMATION STANDARDS

John W. Sheppard, ARINC Engineering Services, LLC, (410) 266-2099, jsheppar@arinc.com
Mark Kaufman, NSWC Corona Division, (909) 273-5725, kaufmanma@corona.navy.mil

ABSTRACT

In this paper, we will discuss the technical issues related to defining a coordinated view of integrated diagnostic information. Our objective is to define a process and modeling framework for solving the information integration problem. The process discussed will be based on formal modeling methods that have been used for years in various contexts but rarely combined. Specifically, the approach will apply a "component" orientation to the problem and will draw heavily from the discipline of information modeling.

Keywords: integrated diagnostics, information modeling, testability, maintenance, information integration

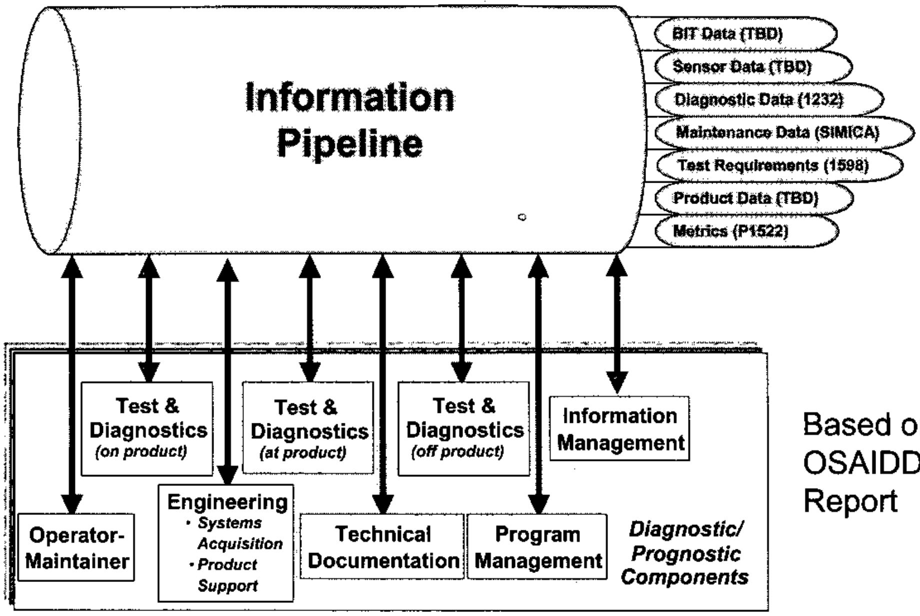
1 INTEGRATED DIAGNOSTICS INFORMATION

The goals of integrated diagnostics are to reduce system cost, improve interoperability, and to insert technology faster by communicating relevant test and diagnostic information. In 1990 integrated diagnostics was defined as: "A structured process which maximizes the effectiveness of diagnostics by integrating the individual diagnostic elements of testability, automatic testing, manual, testing, training, maintenance aiding, and technical information [1]."

The IEEE Standards Coordinating Committee 20 and its Diagnostic and Maintenance Control (DMC) subcommittee have been developing a family of information-based standards centering on test, diagnostics, and maintenance of complex systems. Many of these standards have been published or are in ballot, such as IEEE Std 1232 (AI-ESTATE), IEEE P1522 (Testability and Diagnosability Characteristics and Metrics), and IEEE P1598 (Test Requirements Model). In addition, the Standard for Management of Test and Maintenance Information (formerly IEEE P1389) is in the process of being revisited under the name *Standard Interface for Maintenance Information Collection and Analysis (SIMICA)*. Each of these standards focuses on a different aspect of problems in integrated diagnostics; however, so far, a view of how to use the standards as an integrated whole has not been developed.

Previous attempts to address the need to integrate multiple information sources in integrated diagnostics identified the sources of information but failed to resolve the integration issue. For example, in the early 1980s, the Computer Aided Logistics Support (CALS) initiative developed the Standard Generalized Markup Language (SGML) as an attempt to provide a standard means for formatting logistics data and documents. It was hoped that such standardization would ease the integration and interoperability burden associated with effective use of this information.

0-7803-7441-X/02/$17.00



Based on the OSAIDD Final Report

Figure 1. Integrated Diagnostics Information Framework

While SGML was largely successful (especially given the widespread acceptance of two offspring—HTML and XML), it never achieved its intended goal of information integration.

In the late 1990s, DoD funded an Integrated Diagnostics Demonstration (IDD) to consider how to approach information management for all of integrated diagnostics. This IDD program, the Open System Architecture Integrated Diagnostics Demonstration (OSA-IDD) resulted in the proposal of an Integrated Diagnostics Information Framework. Accompanying this framework was a high-level information model, relating key pieces of information needed for test, diagnostics, and maintenance.

The OSA-IDD Study [2] defined integrated diagnostics as: "...part of the systems engineering (or reengineering) process in which diagnostic functions are partitioned to components, both on and off the product, to optimize economic and functional performance throughout a product's life cycle. Optimal performance is achieved by ensuring effective communication of information relevant to the test and diagnostic process occurs between diagnostic functions and components and across each life cycle phase." The addition of effective communication of information as a requirement for optimal performance is a significant evolution. Integration of information elements provides a comprehensive approach for sustaining platforms. However the proposed integrated diagnostics framework lacked sufficient detail to resolve the interface issues relative to existing and emerging information-based standards. Improvements in diagnostic performance are required to meet today's business objectives. Figure 1 shows the proposed integrated diagnostics framework coming from the OSA-IDD study.

2 INFORMATION SYSTEMS AS DECISION PROCESSES

In any information intensive activity, like integrated diagnostics, it is crucial that information requirements be derived from the objectives of the activity to ensure that the required information can be obtained and effectively utilized. In an ideal world, the information engineering process would proceed by first defining objectives, specifying a process for achieving those objectives, deriving performance requirements from the process, determining and specifying the information required to meet those requirements, and building the information system needed to satisfying the information requirements.

Unfortunately, many processes rely, in large measure, on legacy information systems and processes that make re-engineering the information systems impractical. Thus, an alternative approach becomes necessary in which available information is determined from the existing system (or systems), and this information gets mapped to new or evolving performance requirements.

Both approaches require an integrated view of the information. To accomplish this, one must begin with a formal understanding of the process to be supported as well as an in-depth understanding of the *semantics* of the information supporting the process.

2.1 The Role of Information

All processes depend on the sharing and processing of information. For information to be shared, it must be communicated. A process can be modeled as a decision cycle in which information is received and analyzed, a decision is made about what to do, and some action is taken. The elements of the process by which information is collected, processed, and acted upon can be modeled to any level of detail using well-established techniques such as activity modeling.

This decision cycle has been represented is the command-and-control community as an "OODA-loop" (Figure 2). The OODA loop corresponds to a repeating cycle of four distinct phases:

1. **Observation:** Collecting *information* about the current state of a problem.
2. **Orientation:** Interpreting the *information* to evaluate the current state relative to some objective.
3. **Decision:** Evaluating the *information* to determine a course of action.
4. **Action:** Taking an action based on the decision made to modify the state of the problem.

In the context of test and diagnosis, the diagnostic process can be mapped to the OODA loop as follows. First, the results of one or more tests are examined to capture information about the health-state of the system. This corresponds to *observing* the health-state. Second, the test results are mapped to a set of outcomes and associated inferences to refine the current understanding of the health-state relative to the goal of the test process (e.g., fault isolation). This corresponds to *orienting* the diagnostic system based on its current set of observations and its objective. Third, the test system *decides* what to do next in terms of announcing a fault has been isolated or determining further testing is required. If further testing is required, a test is chosen. Fourth, given a test has been chosen, the test is performed, thus defining how the system *acts*.

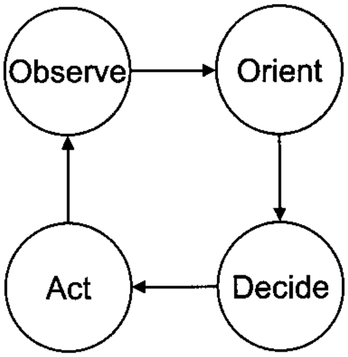


Figure 2. OODA Loop

2.2 Information Modeling

One approach to defining the semantics of information for a component of a larger system is through an "information model." An information model is "a formal description of types (classes) of ideas, facts, and processes that together form a model of a portion of interest of the real world" [3] .

The purpose of an information model is to identify clearly the objects in a domain of discourse (e.g., diagnostics) to enable precise and unambiguous communication about that domain. Such a model comprises objects or entities, relationships between those objects, and constraints on the objects and their relationships. When taken together, these elements of an information model provide a complete, unambiguous, formal representation of the domain of discourse. In other words, they provide a formal language for communicating about the subject of interest or domain.

Using information models, information exchange can be facilitated in two ways. The first is through a set of exchange files. Specifically, information can be stored by one party in a file and read by a second party. The file format is derived directly from the information model and defines the syntax of the message contained within it. The semantics of the message (i.e., the interpretation of the information contained within the file) is derived from the semantics of the model.

The second means of information exchange is through a set of services defined for a hardware component or a software component as accessed via some communications infrastructure. The interface definition for the component is derived from the information model and, once again, defines the syntax of the message. As before the interpretation of the message is derived from the semantics of the model.

When developing large systems requiring extensive communication between components of those systems, the nature of the communication between the components (i.e., the language) must be agreed upon beforehand, such as through a contract. Such advance contracts are typically defined through standards.

Three advantages to using standard information models to define the communications mechanism are evident. First, since standards are published documents, a large audience has access to the standard. By specifying standards in procurement documents or design documents, the designers know the basis for communication before detailed design begins.

Second, the contract defined by a standard has been validated and legitimized by the fact that a community of experts in the domain have gathered and agreed upon the content of the standard. Consequently, users of the standard can trust that a) the standard is technically correct, and b) the community of those using the standard believes the standard is useful.

Third, standards are typically endorsed and accredited by an independent accrediting body. Such endorsement certifies that the standard was developed according to an open process designed to keep the best interests of the community in mind. Examples of such accrediting bodies include IEEE, ANSI, ISO, and IEC.

The EXPRESS information modeling language [4] , standardized by ISO, was designed for formally defining information in support of communication. EXPRESS is object-oriented in flavor but focuses on defining the semantics of the information modeled. In addition, rules have been defined for deriving exchange files and services for information exchange directly from the EXPRESS models.

2.3 Integrating Information Models

As mentioned above, the IEEE is currently in the process of defining several information-based standards that will support integrated diagnostics. The approach taken to define these standards depends heavily on the information modeling technique described above; however, there is currently no direct integration of the resulting standards. The obvious question is how one would go about providing such integration.

Within the information modeling discipline, a distinction is drawn between types of models. Specifically, "conceptual" information models correspond to models that define a domain of discourse yet are not intended to be "instantiated" in any kind of information repository. On the other hand, "concrete" information models correspond to models that define specific aspects or views of the domain and are intended to be instantiated in some form. While the instantiation method or medium is not specified, the semantics are defined sufficiently to ensure that the instantiation meets both the spirit and the letter of the contract as set forth in the model.

Currently, the information models defined for the IEEE integrated diagnostic standards are all concrete models. The role of the conceptual model would be to provide the integrated view across the concrete models (Figure 3). To date, no standard conceptual model has been defined; however, the OSA-IDD Integrated Diagnostics Information Framework provides the start of a possible conceptual model (Figure 4).

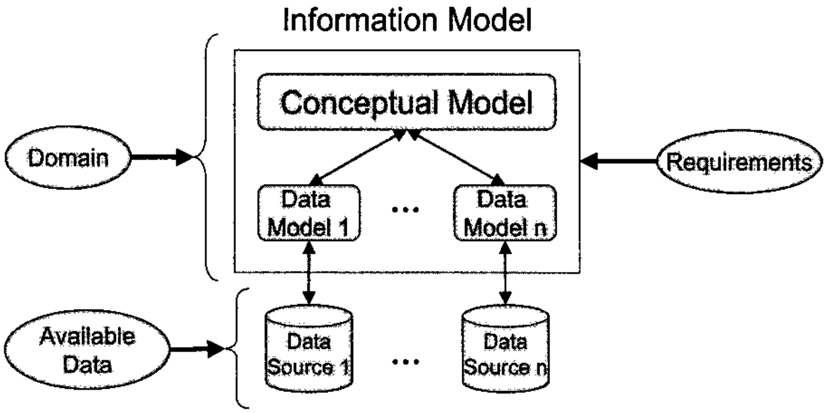


Figure 3. Integrating Information Models

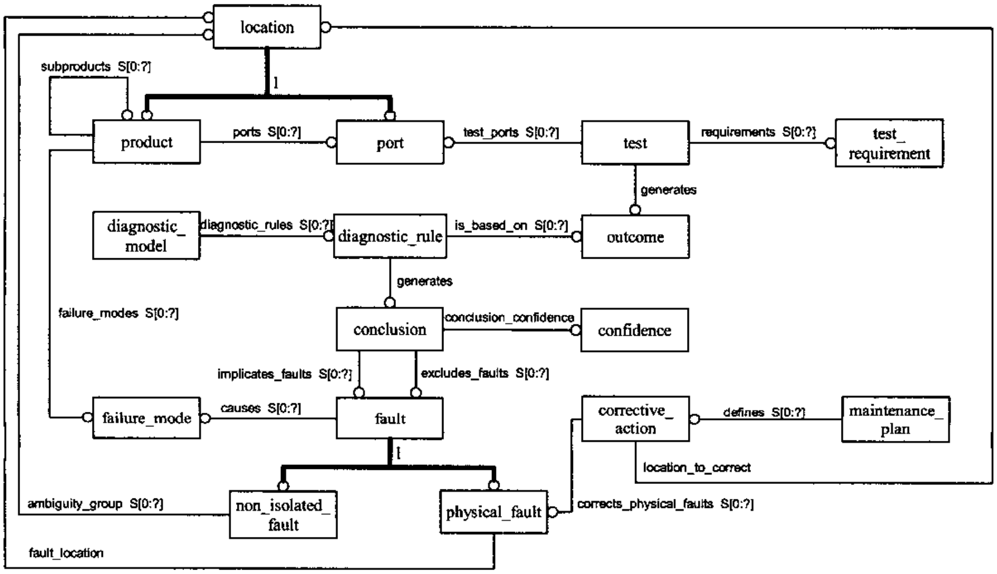


Figure 4. OSA-IDD Integrated Diagnostics Information Model

As described, the conceptual model would define the domain of integrated diagnostics. From this domain model, specific sub-domains can be determined (e.g., test requirements, test instrumentation, test procedures, diagnostics, maintenance, calibration, and training). Concrete models can be defined for each of these sub-domains (or the sub-domains can be subdivided further). The concrete models would then form the basis for formal standards (e.g., IEEE Std 1232—AI-ESTATE) from which databases and information systems can be derived.

3 AN INTEGRATED DIAGNOSTICS CONCEPTUAL MODEL

The ID conceptual model must integrate a wide range of diverse information sources. For example, information on testability, automatic testing, manual testing, training, maintenance, technical information, operator and maintainer input, Built-in-Test, product data, and engineering product support would have to be integrated. We propose a conceptual model that encompasses the ideas presented in Figure 4 while providing direct ties to the standards currently in development by the SCC20/DMC. Our model is shown in Figure 5.

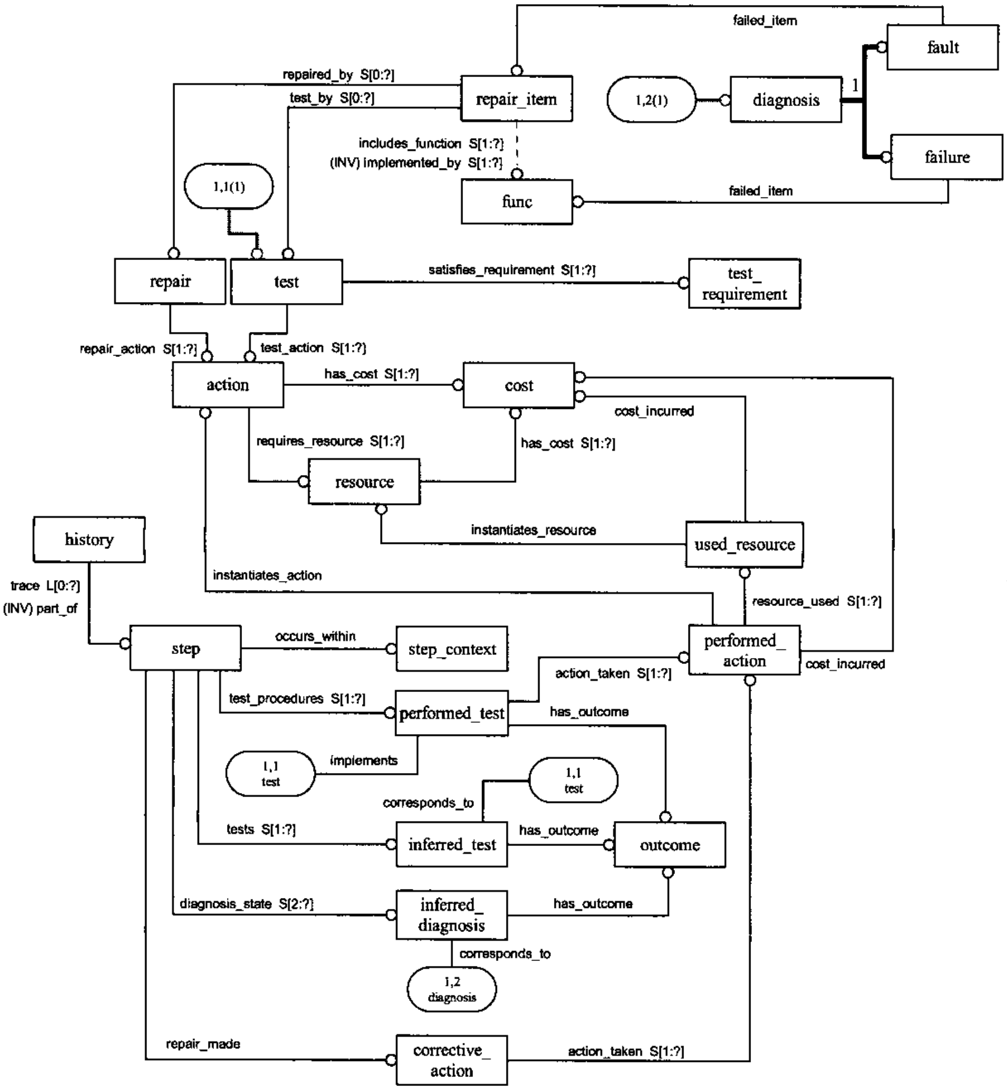


Figure 5. Proposed Integrated Diagnostics Conceptual Model.

For purposes of this paper, we will limit discussion to the relationship of the conceptual model to the four current projects being worked within the DMC—test requirements, diagnostics, testability/diagnosability, and maintenance/diagnostics maturation. The information elements common to each of these projects is the need to characterize a product (referred to in the model as a REPAIR_ITEM), the tests used to evaluate the product, the actions to be taken to support (calibrate, repair, maintain) the product, and conclusions (DIAGNOSIS) that can be drawn about the product to indicate what actions to take. Supporting these activities is the ability to optimize the test and maintenance process. The optimization aspect is captured via the COST entity within the model.

An interesting characteristic of the model is one that is being utilized on an increasing basis in modeling ontologies—maintaining the distinction between "class" and "instance" information. For example, when considering the information surrounding a test within the model, note entities have been defined labeled TEST, PERFORMED_TEST, and INFERRED_TEST. The specific information required to perform a test (e.g., required resources, tolerances, mapping parametric information to outcomes) is class-level information, meaning that it is information that provides the nominal description of what the test should do and what the appropriate responses should be. The specific information obtained as a result of actually performing a test or inferring the results of performing a test, on the other hand, is instance-level information. In other words, this is information that is specific to an actual instantiation of the test.

Note that entity PERFORMED_TEST has an attribute labeled "implements" and entity INFERRED_TEST has an attribute labeled "corresponds_to." These two attributes of the instance-level entities provide ties back to the class-level information specified by entity TEST. Typically, such relationships are represented in EXPRESS via subtyping; however, we decided to use this approach to suppress inheritance so that we can avoid issues arising from split inheritance (e.g., such as would arise by subtyping INFERRED_DIAGNOSIS from DIAGNOSIS). A similar technique is used in the AI-ESTATE models between entities in the Common Element Model and the Dynamic Context Model.

4 RELATIONSHIP TO ESTABLISHED MODELS AND STANDARDS

The SCC20 DMC subcommittee is developing a family of standards [5] [6] [7] [8] [9] [10] that are product information exchange standards for test, diagnosis, and maintenance. The original standards developed by the DMC, the 1232 series, developed a means of exchange of information between diagnostic reasoners. As the information models for the 1232 standards were developed, it became apparent that these models could be used for standardizing testability and diagnosability metrics as well as diagnostic and maintenance history information.

4.1 History of AI-ESTATE Standards

IEEE Std 1232-1995 [6] defines the architecture of an AI-ESTATE-conformant system and has been published as a "full-use" standard; however, this standard was published before the vision of AI-ESTATE was fully developed. IEEE Std 1232.1-1997 [7] defines a knowledge and data exchange standard and is now a "full-use" standard. In 1998, the third of a series of three standards was published by the IEEE addressing issues in system-level diagnostics. This standard, IEEE Std 1232.2-1998 [8] was approved as a "trial-use" standard. Trial-use indicates that it is preliminary in nature, and the standards committee is seeking comments from organizations attempting to implement or use the standard. Recently, the 1232 standard documents were merged into a single standard that encompasses the 1232-1995, 1232.1-1997 and the 1232.2-1998 documents [5]. The complete 1232 standard, which was just approved by the IEEE Standards Board as a full-use standard, contains the diagnostic information models and formally defines a set of standard software services to be provided by a diagnostic reasoner in an open-architecture test environment.

In 1997, the DMC began to work on a new standard focusing on expanding the work of the cancelled MIL-STD 2165. The military standard emphasized specifying the essential elements of a testability program and explained the elements needed to define a testability program plan. In addition, MIL-STD 2165 included the "definition" of several testability metrics. MIL–STD 2165 included a testability checklist to be used to determine overall system testability. With the cancellation of military standards and specifications in 1994, and with the lack of specificity and clarity in MIL-STD 2165, it became evident that a replacement was necessary. The approach taken to develop this replacement standard involved defining testability and diagnosability metrics based on standard information models. Specifically, it was found that the AI-ESTATE models provided an excellent foundation for defining these metrics. AI-ESTATE provides formal definitions of the same information required for determining the testability and diagnosability of a system. With these formal definitions, the constraint language of EXPRESS can be applied directly to define metrics and characteristics of testability and diagnosability.

The Management of Test and Maintenance Information Standard (formerly IEEE P1389) will be revisited as IEEE Pxxxx Standard Interface for Maintenance Information Collection and Analysis (SIMICA). No project number has been assigned yet; hence the Pxxxx. IEEE P1598 Test Equipment Requirements Model (TeRM) is under development. This model is a description of what needs to be tested in a system, not how to perform the tests. A preliminary version of the model has been developed and is being revised.

4.2 Relationship Between the Standards

IEEE 1232-2002 is the "keystone" standard of the DMC work (Figure 6). IEEE 1232 describes the diagnostics domain. IEEE 1232 defines the information related to system test and diagnosis. The description of the diagnostic domain enables the exchange diagnostic information between applications. IEEE 1232 also supports modular diagnostic architectures, and interoperability with other test assets. The 1232 standard was developed using information modeling, resulting in the definition of five models addressing static and dynamic aspects of the diagnostic domain. The AI-ESTATE information models are: the Common Element Model (CEM), the Fault Tree Model (FTM), the Diagnostic Inference Model (DIM), the Enhanced Diagnostic Inference Model (EDIM), and the Dynamic Context Model (DCM). This standard formally defines a set of standard software information services to be provided by a diagnostic reasoner in an open-architecture test environment. These five models and their associated services are used by the other standards in the DMC family.

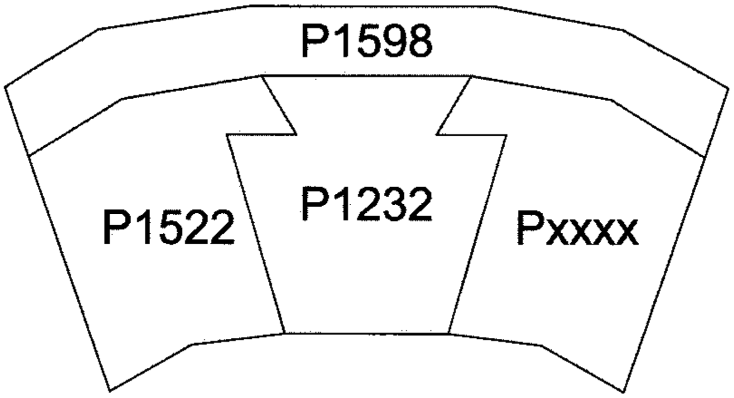


Figure 6. DMC Family of Standards

IEEE P1522 Standard Testability and Diagnosability Characteristics and Metrics uses the models of IEEE 1232 to define fundamental information for testability analysis. Because P1522 fundamentals are tied to definitions in standard models ambiguity is eliminated. P1522 metrics and characteristics are calculated from fundamentals. Additionally P1522 allows for extension and expansion of the metrics and characteristics using the fundamentals. P1522 uses the models of P1232 to define the fundamental elements that comprise testability and diagnosability metrics. For example, 1232 provides the means to determine the total number of faults and the number of faults that are detectable. The EXPRESS within P1522 makes a query of the 1232 complaint reasoner and receives the total number of faults and the number to detectable faults. P1522 uses this information to calculate the percentage of faults detectable. Although many of the metrics in P1522 can stand alone, the maximum utility of the metrics is realized in conjunction with 1232.
IEEE Pxxxx Standard Interface for Maintenance Information Collection and Analysis (SIMICA) will be an information model that defines the information domain of system maintenance. SIMICA will support the capture of historical maintenance/diagnostic data, facilitate discovery/extraction of maintenance knowledge, and provide foundation for diagnostic and product maturation. Like P1522, SIMICA also uses the models of 1232 to define maintenance related fundamentals.

IEEE P1598 Standard for the Test Requirements Model (TeRM) will provide formal description of product behavior under test. P1598 will also define formal semantics for test requirements. This information will feed the entire product lifecycle (concept to field). P1598 emphasizes "what" to test, not "how" to test. Like the two previous standards P1598 uses the models of 1232.

P1522, P1598, and SIMICA build on the models of 1232 and add additional depth and capability to the combined "picture" of the diagnostics domain. Other standards could be added to extend the DMC "arch."

5 SUMMARY

We defined a process and a framework to deal with the information integration problem associated with integrated diagnostics. We presented a conceptual model of integrated diagnostics and showed how the concrete models defined the IEEE DMC family of standards could be integrated into the conceptual model. The nature of each of the IEEE standards was described as well as the relationships between the standards.

The DMC family of standards covers the diagnostic domain. The integrated diagnostics conceptual model can be used to provide a framework to integrate the information from the DMC standards and others to describe the integrated diagnostics domain.

6 REFERENCES

[1] Keiner, W. 1990. “A Navy Approach to Integrated Diagnostics,” AUTOTESTCON ’90 Conference Record, Piscataway, New Jersey: IEEE Press, pp. 443-450.
[2] DoD Open System Architecture Integrated Diagnostics Demonstration (OSA-IDD) Project Report. 1998
[3] Schenk, D., and P. Wilson. 1994. Information Modeling: The EXPRESS Way, New York: Oxford University Press.
[4] ISO 10303-11:1994. Industrial automation systems and integration—Product data representation and Exchange—Part 11: Description methods: The EXPRESS language reference manual, Geneva, Switzerland: International Organization for Standardization.
[5] IEEE P1232. 2002 IEEE Standard for Artificial Intelligence Exchange and Service Tie to All Test Environments (AI-ESTATE), Draft 5.15, Piscataway, NJ: IEEE Standards Press.
[6] IEEE Std 1232-1995. IEEE Standard for Artificial Intelligence Exchange and Service Tie to All Test Environments (AI-ESTATE): Overview and Architecture, Piscataway, NJ: IEEE Standards Press.
[7] IEEE Std 1232.1-1997. IEEE Standard for Artificial Intelligence Exchange and Service Tie to All Test Environments (AI-ESTATE): Data and Knowledge Specification, Piscataway, NJ: IEEE Standards Press.
[8] IEEE Std 1232.2-1998. IEEE Trial-Use Standard for Artificial Intelligence Exchange and Service Tie to All Test Environments (AI-ESTATE): Service Specification, Piscataway, NJ: IEEE Standards Press.
[9] IEEE P1522. 2001. IEEE Standard Testability and Diagnosability Characteristics and Metrics, Draft 1.1, Piscataway, NJ: IEEE Standards Press.
[10] IEEE P1598. 2001. IEEE Standard for the Test Requirements Model, Draft 1.0. Piscataway, NJ: IEEE Standards Press.