Prognostics Framework
Software Design Tool

Dr. Li Pi Su, Advanced Technology Office
The U.S. Army Test, Measurement, Diagnostic Equipment Activity
The U.S. Army Aviation and Missile Command, Redstone Arsenal, AL 35898
Telephone: (256) 842-8755; Fax: (256) 842-8128
E-mail: lipisu@redstone.army.mil

Ms. Mary Nolan, Mr. Greg deMare, and Ms. Becky Norman
Giordano Automation Corp. Sparta, NJ 07871
Telephone: (973) 729-5888; (706) 569-6546; Fax: (973) 729-1673;
E-mail: mary@giordano.com; greg@giordano.com
becky@giordano.com

Abstract - The U. S. Army Logistics Integration Agency has funded (1998-2000) the Advanced Technology Office (ATO), the U.S. Army Test, Measurement, Diagnostic Equipment Activity (USATA), the U.S. Army AMCOM, to develop a "Prognostics Framework". This is a generic software developmental tool set, open architecture capability to integrate various prognostic mechanisms, and to provide operational and logistic decision-making information. The Prognostics Framework is a horizontal technology and is tailorable to apply to new and existing systems. This approach saves time and money. It is the fastest to converge on prognostics capabilities into manageable information for the system operators, the maintenance crew, and logistics planners across systems. The Prognostic Framework, a system-level prognostic manager, ties-in to logistics infrastructure (e.g.: IETM, logistics planning, mission planning, statistical tools, spare parts provisioning...). Prognostics Framework is integrated with Diagnostics to provide a total "Health Management Capability. This paper provides Prognostics Framework architecture, design approach, and interfaces capabilities.

TABLE OF CONTENTS

1. INTRODUCTION

Prognostics, the identification of incipient faults, are normally considered as component/part issues rather than a system problem. When cast as a component/issues, prognostic techniques are limited to those using failure and measurements that are specifically correlated with a failure process. Prognostics, dealt with as a system-level problem, can obtain access to many more measurements to determine the overall health condition of the weapon systems and hence to determine mission readiness of the system. During the past six years, component prognostic techniques such as neural networks, time/stress measurement devices, vibration monitoring, oil monitoring, sensors, trend analysis, statistical analysis, etc. have been developed in Labs, universities and industry. These techniques are, by their very nature, equipment-specific and very few applications have been implemented. There are no system-level applications for these techniques because of their inherent diversity. There is no overall "Prognostics Framework" to manage the information provided by the individual prognostic techniques.

The prognostic capability is considered at a functional level, where specific mission and safety critical functions are monitored to predict impending failures. An overall system "health monitoring" architecture is employed to provide proper crew notification, data collection and logging, and centralized processing of monitored data. This paper will provide how the "Prognostics Framework" is developed and what are the products of the "Prognostics Framework."

2. TECHNICAL BACKGROUND

The model-based diagnostic reasoning tool set, the Diagnostic Profiler and Diagnostician, is a new paradigm/technology that has made major inroads to the capability to increase diagnostics effectiveness [1-4]. This technology was initially funded by the Army under USATA ATO project office as the Diagnostics Analysis and Repair Tool Set (DARTS) and has been enhanced and commercialized by the Giordano Automation Corporation (GAC). This tool not only allows design for testability but also for diagnose-ability [1- 8]. It is an open architecture and object-oriented software. It is also expandable.

Since the initial development of this technology, it has been applied in Army, Navy, Air Force and commercial programs with great success.

3. TECHNICAL ADVANCES

The Prognostics Framework effort is to extend the diagnostic

[1] U.S. Government work not protected by U.S. copyright

development tool set to include the capability of developing a total health management system. This is accomplished by augmenting diagnostic reasoning capability to include prognostic indications. The underlying model-base structures is being augmented with additional information related to parts degradation, failure modes, mission criticality, historical data, trend analysis, etc. [8]. This provides a basis for performing both prognostics and diagnostics of the weapon system. Hence, it will provide an overall health monitoring capability.

The overall software structure is an open architecture approach. The software is in object-oriented programs. The available technologies were investigated and integrated for application. The technologies required to support a prognostic scheme are as follows: Mechanical systems, analog systems/items, digital systems/items, historical failure analysis, etc. [7]. The overall technical and design approaches are built upon a model-based diagnostic reasoning to provide the following features: 1) open architecture and generically applicable software and information structure; 2) single knowledge base for embedded and off-line; 3) object-oriented software structure which is extendible; 4) hierarchical modeling approach that enables system integration and 5) applicability to both legacy systems and new designs.

4. DESIGN FOUNDATIONS

The underlying design foundations are the concept of hierarchical modeling and the separation of the test and diagnostics functions both during system development and deployment. These attributes enable the diagnostic reasoning run-time system to be employed at all levels of system and test/diagnostics environments: embedded on-line, manufacturing test, and maintenance test. This is commonly referred to as *Vertical Integration* [5]. Since the capability is made up of a hierarchical series of design-based diagnostic models, *Horizontal Integration* [5] is also achieved in that the hierarchy represents the various system and subsystems that are integrated to make up an overall mission capability. It therefore represents a diagnostic profile of the horizontally integrated mission capability. Since the tools represent a system engineering solution applicable to any type of system, it can be implemented across many system platforms, in further support of Horizontal Technology Integration.

The Diagnostician is an implementation of a new paradigm for diagnostics: model-based reasoning. The diagnostic model is independent of the source of test result data and independent of the sequence of tests that are run. The Diagnostician contains a diagnostic model of the system in the form of a connectivity matrix, which represents the propagation of faults to monitor or sensor locations as rows in the matrix, and the fault coverage of monitored locations as columns in the matrix. When used in run-time, the software algorithms and knowledge base (matrix) operate to isolate faults without built-in test (BIT) fault codes, fault isolation sequences and without hard-coded diagnostic routines.

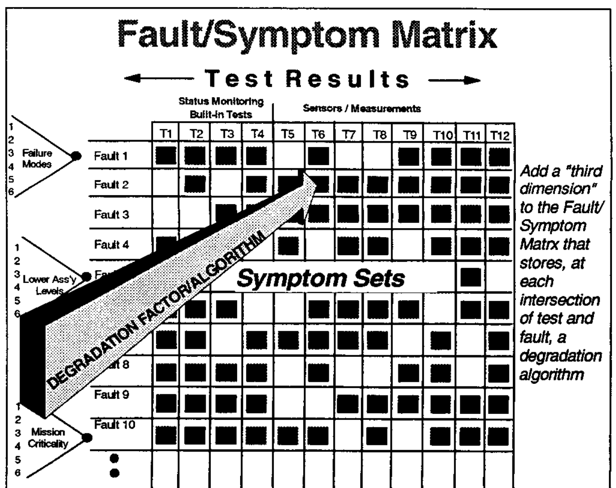
Another key foundation for the Prognostics Framework was the development of the capability to "adapt" a diagnostic model of a system to a new configuration state such that a single model could maintain its integrity and be used for a virtually unlimited number of operational configuration states of a system.

This key foundation is that allows fault detection, isolation and reconfiguration, recovery, and reporting simplifies system level diagnostics and enables requirements be allocated to the local or distributed level as opposed to the host (central) level of the system hierarchy. Moreover, this capability was recently completed on a NASA power control electronics program that focused by applying model-based reasoning and embedding the resulting Diagnostician-on-a-Chip technology into a state-of-the-art fault tolerant remote power controller. The program advanced model-based reasoning to support the reconfiguration and recovery function by developing the means to "adapt" the model-base to a new state upon reconfiguration.

5. AUGMENTATION OF DIAGNOSTIC TOOL SET

The Diagnostic developmental tool set consists of two main parts: Diagnostic Profiler and Diagnostic Reasoner. It has capability of performing dynamic analysis of large amounts of performance monitoring and built-in test data to provide real-time fault isolation. The Diagnostician is capable of analyzing all test results as a snapshot of the state of the system and requires no complex supporting fault isolation logic. The Diagnostic Profiler is extended to include the capability of accepting not only multiple built-in tests (BITs) and sensors information but also prognostic mechanisms. The Diagnostic Reasoner is expanded to include capability of correlating failure predictions to time and mission and provide mission capability, function availability, and maintenance data for battlefield decision and logistics planning.

5.1. FAULT PROPAGATION MODEL EXTENSION



- *Augment the Diagnostician to enable processing of the many types of algorithms that are used to identify wear factors.*
- *Augment Fault/Symptom Matrix to hold wear factor algorithms for the wide variety of prognostic techniques available and being developed.*

The Diagnostician consists of intelligent algorithms which use a "Fault Propagation Model" (FPM) of a system to augment existing BITs resources. The model consists, in part, of a fault symptom matrix that correlates all possible faults to all symptoms, and/or test results. The matrix identifies the coverage of BITs and on-board sensors mapped across physical components instead of system/item functions. The intelligent algorithms isolate faults using the existing BIT data and provide more granular diagnostics. Using this information technology, the existing built-in *test* is extended to built-in *diagnostics.*

Based on hierarchical and open architecture concepts, the FPM is extended to a system-level prognostic information manager. That is, the extended FPM now encompasses degradation factors, failure modes, and algorithms at each applicable measurement or sensor location, for a wide variety of available prognostic techniques. The diagnostic reasoning algorithms is augmented with capability to process the many types of algorithms that are used to identify degradation factors. This system-level prognostics manager integrates and manages the diverse prognostic techniques, such as neural networks, time/stress measurement devices, vibration monitoring, oil monitoring, sensors, trend analysis, and statistical analysis to obtain system-level predictive information.

5.2. Data Logging Capability Extension

- *Augment Diagnostician's data logging capability to include operational factors (time/stress snapshots) with test results.*

The run-time Diagnostician's data logging capability is extended to include time-based snapshots of measurement/sensor data of each applicable operational factor (e.g.: time/stress snapshots, etc.). The result is an expanded run-time Diagnostician that functions as a system-level prognostic manager to integrate prognostic techniques and data for an overall system health management.

6. Software Infrastructure

Like the model-based diagnostic tool set, the underlying software infrastructure for the Prognostics Framework is generic, hierarchical, and open architecture, with a single knowledge base for both embedded and off-line applications. It is an object-oriented software structure. The Framework software consists of development tools, operational software, and inputs data preprocessing software. The concept of "rapid prototyping" is being employed in this project. Open architecture is a design foundation, as this enables designers to make any major modifications quickly and efficiently as these concepts evolve.

6. 1. Development Tools

To encompass and manage the prognostic information/data, the Prognostic Framework consists of a suite of development tools and operational software. There are five development tool sets. They are *system design tools, prediction modeling tools, operations support tools, maintenance support tools, and runtime generation tools* [8]. Each tool set supports developers in implementing the prognostics framework output requirements. While the operational software process the outputs of prediction mechanisms and status sensors to provide the required outputs [8]: operational, mission, and maintenance information.

Moreover, the Prognostic Framework employs generic data structures that can be applied to any system using a development system. The development system guides the developer through the application development process using a graphical Windows style user interface that is easy to use. The first step of the development process is to define the design of the system hierarchically from sub-systems to individual parts. The second step in the development process is to define the prediction mechanisms and other diagnostic/prognostic data available for each sub-system such as artificial neural network indications, mathematical modeling outputs, BIT and sensor data. The data defined will be given to the Prognostic Framework during system operation for integration into the total health monitoring of the system. Thus, a "divide and conquer" approach is implemented where individual mechanisms can be dedicated to sub-systems and then integrated into a system wide diagnosis/prognosis using the Prognostic Framework. The third step in the development process is to relate the diagnosis/prognosis achieved to the functions and missions that must be performed by the system. This step provides the information required for the Prognostic Framework to act as an operational decision aid supporting situational awareness, operational readiness, and maintenance planning. The fourth step in the development process is to relate the diagnosis/prognosis achieved to maintenance actions. The Prognostic Framework acts as a maintenance support tool providing preventative maintenance tasks, logistic planning data, and parts ordering information. The fifth and final step in the development process is to automatically generate and simulate the operation of runtime knowledge base (matrix) that will be deployed in the system application.

The Prognostic Framework development system populates the generic data structures used during runtime operation with the data specific to each application. The tool uses automated processes wherever possible and it is easy to understand "wizard" style provides developers with an easy alternative to programming. System knowledge is easily captured, changed and maintained using development software and a database.

6. 2. Processing Inputs Software

The Prognostics Framework tool set is required to accept and process inputs from a variety of different data sources, such as, system design data, failure modes, parts reliability, parts degradation data, prediction mechanisms, diagnostic test information, sensor data, etc. Moreover, user inputs in the form of variables regarding stresses and environmental conditions for a mission scenario. All these data/ information will need to be incorporated. There are two types of preprocessing techniques, the mathematical manipulations and the Boolean expression of ranges used to filter out noise within a system. These two types of techniques are used in various combinations to achieve a generic approach capable of handling all application instances [8].

7. Prognostic/Diagnostic Runtime Software

The parts reliability, parts degradation data, failure mode data, and all prognostics related data are integrated with Prognostics Framework tools to generate prognostic/diagnostic runtime software for the system operator. The output products for the system operator includes:

1) Hierarchical Representation of the System, subsystems and faults.
2) Definition and Correlation of Diagnostic and Prognostic Mechanisms to faults.
3) Correlation of faults to ability to perform system functions and defined missions.
4) Correlation of faults to repair procedures and parts ordering information
5) Real-time and on-line Diagnostic/Prognostic Health Monitoring during mission scenario

8. Prognostic Runtime Interfaces

The Prognostics Runtime Interfaces consists of the Current Suspect Information and the Data Type Oriented Information.

Current Suspect Information consists of the Time Based Suspect Lists and Current Suspect Evidence Basis Information. The Time Based Suspect Lists consist of routines to perform the following tasks: to identify faults which analysis has shown, has occurred, or will occur; to identify functions which analysis has shown, has failed, or will fail; to identify parts which analysis has shown, has failed, or will fail; to identify repair items which analysis has shown, has failed, or will fail; and to relate suspect repair actions to suspect repair items. The Current Suspect Evidence Basis Information consist of routines to obtain the evidential basis for a suspect fault, the evidential basis for a suspect operator action, and the evidential basis for a suspect part.

The Data Type Oriented Information consists of routines to process and to provide the following information: prediction times, faults, function and mission information, operator action information, part information, repair item information, repair action information, raw data information, and input log data.

9. Benefits and Conclusion

The Prognostics Framework Design Development Tool is applicable for all systems, both military weapon systems and non-military systems.

9.1. Military Systems

In this time of military downsizing of both manpower and budget, fix-forward maintenance in the field is probably the most cost-effective solution to high maintenance costs. This Prognostics Framework will provide weapon system operators with concise identification of the system's mission capability and estimated reliability so they can make rapid battlefield decisions. Moreover, the diagnostics information will provide maintainers with repair information. The diagnostics will enable even novice soldiers to repair the system. Additionally, the Prognostics Framework is a user-friendly system, an integrated diagnostics system, and a system health monitoring system that may be used as a training tool for the users and to plan logistics support.

Prognostics Framework of weapon systems can be integrated seamlessly to the end in the performance of his/her operation/maintenance duties. Soldiers want and require that their systems operate as specified. This new concept of Prognostics Framework for weapon systems provides valuable information to a field commander, thereby, giving him/her a more complete picture of the status of the systems available to him/her and as to how he/she can be best deployed/utilized in an automated media.

The ATO and Giordano Automation has been addressing these Army maintenance problems since 1991. The Diagnostics Analysis and Repair Tool Set (DARTS) allows accurate diagnosis of faults so that expensive no evidence of failure rates on returned parts can be significantly reduced or eliminated. The diagnostics on-a-chip may be used for embedded diagnostics and can be aggrandized to be Prognostics Framework.

The ATO has also developed technology to improve ETMs to provide a diagnostic driven IETM for fix-forward and to reduce training costs by allowing training in the field using the same procedures used for repair.

The Prognostics Framework integrates the above technologies as well as the exiting component prognostic techniques such as neural networks, time/stress measurement devices, vibration monitoring, oil monitoring, sensors, trend analysis, statistical analysis, etc. and the parts reliability, parts degradation data, and failure modes data to generate a system-level prognostic management.

The Prognostics Framework for Weapon Systems is generic and is an open architecture concept and the system will be interoperable with other equivalent systems in other services and NATO systems.

9.2. Non-Military Systems

Modern day systems such as, airplane, spacecraft, satellite systems, aircraft, submersible vehicles, and armored combat vehicles often have to operate in hostile environments with little or no human interaction. An autonomous real-time health monitoring system which provides automatic fault detection, isolation, reconfiguration, and reporting increases the chances for these missions to be successful. The ability of a system to provide rapid fault isolation via the process of diagnostic reasoning is an important aspect of health monitoring. The diagnostic capability (i.e. determination of which system components are faulty) for the most part drives system health management and is typically the operational driver which determines whether the system real-time constraints will be satisfied. A solution to this dilemma is the utilization of the model-based diagnostic reasoning capability within Giordano Automation's Diagnostician. The Diagnostician is capable of performing dynamic analysis of large amounts of performance monitoring, sensor and built-in test data to provide a real-time fault isolation.

Moreover, the prognostic capabilities orchestrate a total system's health management capability. Prognostics Framework includes accurate diagnostics. The resulting capability provides health management information to operators and maintainers to ensure operational availability and mission sustainability over time and enables maintenance actions to take place prior to a mission critical failure event.

References

[1] *"Diagnostic Analysis and Repair Tool Set (DARTS): An Enabling Technology for Concurrent Engineering,"* Mary Nolan, Li Pi Su. AUTOTESTCON, San Antonio, September '93

[2] *"Re-Engineering the Test Development Process,"* Paul J. Giordano, Ford Levy. AUTOTESTCON, Anaheim, September '94

[3] *"Intelligent Maintenance Aid Software,"* Gerard Giordano, and Greg deMare. AUTOTESTCON, Anaheim, September '94

[4] *"A New Breed of Depot ATE,"* Dave Carey and Paul Giordano. AUTOTESTCON, Atlanta, August '95

[5] *"An Open Architecture COTS Tester with Intelligent Diagnostics,"* Mary Nolan, Ken Carroll. AUTOTESTCON, Dayton, '96

[6] *"Re-engineering Legacy Tech Manual's Troubleshooting Procedures into Smart Model-based Diagnostics,"* Mary Nolan, Li Pi Su, Vance Palmer, and Ronald Newman. AUTOTESTCON, Anaheim, '97

[7] *"Prognostics Framework for Weapon Systems Health Management,"* Li Pi Su, Mary Nolan, and David R. Carey. AUTOTESTCON, Salt Lake City, '98

[8] *"Prognostics Framework,"* Li Pi Su, Mary Nolan, Greg deMare and David R. Carey. AUTOTESTCON, San Antonio, '99

Dr. Li Pi Su, an Electronic Engineer, Advanced Technology Program, U.S. Army, Test, Measurement, and Diagnostic Equipment Activity, Department of the Army (Redstone Arsenal, AL) since 1992, has a B.S. (1960) and a Ph.D. (1966) in mathematics. She also has a B.S. electrical engineering 1983). Post Doctoral Research, University of Toronto (1966-67). Professor of Mathematics, Tsing-Hua University, Taiwan (1967-68). Professor of Mathematics, University of Oklahoma (1968-78). Electrical Engineer, Department of the Air Force (Tinker AFB, OK, 1980-85). Logistics Engineer, Department of the Army (Lexington Army Depot, KY, 1985-92). 1992-Present, she has completed management of the research and development technologies: the Integrated Diagnostic Analysis and Repair Tool Set; Embedded Diagnostics; Reengineering troubleshooting procedures; the Integrated Diagnostics; and Repair Information System Programs. She also has been researching and developing diagnostics/prognostics technology and Interactive Electronic Technical Manual related technology.

Mary Nolan is the project manager for Giordano Automation Corp., responsible for applications of the company's diagnostics tools, the Diagnostic Profiler and Diagnostician to test program sets, embedded diagnostics and IETMs. Mary has a BS Degree from Trenton State College and also attended the Defense Systems Management College Program Managers Course, in Ft. Belvoir. She has worked in the area of test and diagnostics for almost twenty years. She has been working on an initiative with the US Army TMDE Activity's Advanced Technology Office to develop a Prognostics Framework using Giordano Automation's model-based diagnostic reasoning tools.

Rebecca Norman is a Senior Software Engineer at Giordano Automation Corp. Rebecca is responsible for the company's main product, the Diagnostic Profiler. She is the lead software engineer for the development of the development environment for the Prognostics Framework. Rebecca has a Bachelor of Science degree in Computer Science from Fairleigh Dickinson University where she graduated in May 1997 with Cum Laude honors.

Gregory deMare is a Senior Software Engineer at Giordano Automation Corp. Greg is responsible for the company's Diagnostician run-time product. Greg is the lead software engineer for the development of the run-time algorithms and software for the Prognostics Framework. Greg has a BA from Yale University, 1965 and a M.S. in Mathematics from New York University, 1969.