THE IMPACT OF COMMERCIAL OFF-THE-SHELF (COTS) EQUIPMENT ON SYSTEM TEST AND DIAGNOSIS

William R. Simpson, Institute for Defense Analyses
John W. Sheppard, ARINC Research Corporation

ABSTRACT

Improved interface standards and reduced design budgets dictate that Commercial Off-the-Shelf (COTS) equipment be more readily integrated into system design. Often COTS is chosen for its functional capabilities and electronic compatibilities with little regard to testability and maintainability features. COTS equipment is often characterized by a lack of detailed information about the specific internal design of the equipment. Complex interactions across an array of subsystems may decrease the diagnosability of the system as a whole where deficits in information occur. In this paper, we will describe an analysis approach for assessing system testability and providing system diagnostics that is amenable to including COTS equipment in the system under test. We will illustrate the approach with the standard analysis of a system consisting of several subsystems with full information available.

BACKGROUND

The complexity associated with diagnosing and repairing modern systems is well known. Today, efforts in fault tolerant design, integrated diagnostics, and standardization seek to grapple with this complex problem. In addition, the field of artificial intelligence continues to offer several computational approaches to diagnosis. In the past, diagnostic strategies were developed by system "experts." Because of the complexities of problems to be solved, various strategies may exist for diagnosing the same system. Often the execution of planned diagnostic strategies fails because of an expertise gap that arises when the knowledge of the technician operating the system is less than the knowledge of the experts whose expertise was incorporated into the system. Early attempts to develop computer-assisted tools using AI concentrated on rule-based expert systems. [1,2] In addition to rule-based expert systems, AI is incorporating object-oriented design to develop knowledge bases. In particular, rule-based systems are being replaced by frame-based systems and semantic networks.[3,4] Frame-based systems represent knowledge as packets of information (called objects or frames) and relationships between these packets (called relations or links). The semantic network is a structure imposed on frame-based systems in which the relations describe semantic properties between the frames.

Most recently, several companies and universities have applied a model based approach [5-14] called dependency model or logic modeling, that uses what we call the information flow model. This approach has been demonstrated to work well at the system level for assessing testability, and enabling effective diagnosis.

THE INFORMATION FLOW MODEL

The information flow model is based on concepts from data fusion in which multiple sources of information are available, and the information provided by these sources must be "fuzed" to draw conclusions. To construct an information flow model one of two approaches may be used. For users who have in-depth knowledge about a system, a functional model of that system can be constructed. A typical functional model consists of relationships between tests and conclusions (and, ideally interdependencies between multiple tests). This approach usually requires a level of understanding of a system that is approximately one half to a full level more detailed than that required for effective diagnosis. For example, to model a system at the card level, some knowledge of the circuitry on the cards is required.

INTERNATIONAL TEST CONFERENCE 1993
0-7803-1429-8/93 $3.00 © 1993 IEEE

An alternative approach to functional modeling is attribute mapping.[15] This approach begins with a set of tests and a set of diagnostic conclusions (or knowledge about the testing process) and produces a model that is based on the expected test outcomes for the diagnostic conclusions within the model. These outcomes are recorded, between tests. A significant advantage to attribute mapping as a modeling methodology is that it requires only a level of understanding of the specific tests to be performed at the level of detail consistent with the level of diagnosis.

Because the model uses information fusion, testability analysts have to consider the information gained from performing a test as they develop the model. The analyst begins by specifying the primitive elements and then proceeds to a description of the logical relationships and groupings of these elements.

The primitive elements of the model are the tests and fault-isolation conclusions, which are based directly on information fusion. Tests correspond to the information sources, while fault-isolation conclusions correspond to the set of conclusions that can be drawn after running tests. A test is any source of information available that the analyst can use to discern a fault-isolation conclusion. A fault-isolation conclusion is any element that we can isolate within the model. Thus, a conclusion is often a failure mode of some component or functional unit within the system.

The model also includes three special primitive elements: testable input, untestable input, and No Fault. The inputs represent information entering the system that my have a direct bearing on the health of the system. A testable input is a conclusion corresponding to an external stimulus combined with a test that examines the validity of that stimulus. If we have any input that cannot be examined for validity, that element is called an untestable input. Finally, the model includes a special conclusion corresponding to the condition that the test set found no fault. No Fault, also referred to as RTOK (for retest okay), provides us with a closed-set formulation that includes anything not directly accounted for.

The analyst organizes conclusions according to the required repair level. Conclusions include line-replaceable units (LRUs) if the need is at the organizational level, shop-replaceable units or components if it is at the depot level. Further, the analyst can develop models that cross levels. That is, in a single model, a conclusion may be a subsystem, an LRU, and SRU, a component, or a failure mode, depending on what is appropriate.

COTS SUBSYSTEMS

Many systems are functionally subdivided such that system functions are provided through standard interfaces. In these cases, commercial equipment may exist to provide the needed functionality. Commercial equipment is often produced in large quantities to serve a number of different applications. An example would be a telecommunication subsystem that utilizes standard phone jacks and a standard VBIS protocol, and communicates with the system through an S-100 base. Such COTS systems are often fully developed by the manufacturer but not well documented due either to trade secret elements or the expense of providing such documentation. The use of COTS can provide a large cost savings in R&D dollars but at the expense of a more generalized product with reduced documentation.

There may be additional cost savings when the COTS subsystem is manufactured in numbers considerably larger than would be required by the designed system. The engineering tradeoff is to ascertain whether or not the cost benefit is worth the design compromises that are brought about with COTS. Several design compromises may include providing functions not needed, not precisely as needed, or functions that must be added as an applique. In the telecommunication subsystem example, this may include a requirement for VBIS 4.2 at 9600 baud, but the COTS system includes this and VBIS 3.1 at 2400 baud. The latter would not be needed, but is none-the-less a function of COTS. We may need also to add a separate circuit to provide encryption which might be a part of the telecommunication subsystem had we not used COTS. One design compromise, often overlooked, is the reduction in system testability that comes about when we integrate less than fully documented subsystems into our system. This paper will attempt to analyze the impact of COTS in the system testability.

A CASE STUDY

We will use a model of a hypothetical antitank missile launcher to illustrate the concepts and computations described in this paper. The case study is described extensively in references 5-10. We derived the case study from an actual missile system, modifying it

extensively to illustrate certain mathematical principles. As a result, although the data represent an actual system, the model may deviate significantly from what may be encountered in a real missile system.

The hypothetical missile launcher consists of a tripod, a gunner's optical sight, a launch tube, an optical sight attachment, and an electronic guidance computer. The missile contains two solid-propellant motors. The launch motor ejects the missile from the launch tube and is burned out by the time the missile has left the tube. Only after the missile has flown several meters does the flight motor ignite, so no protection is required for the gunner.

After the missile leaves the launch tube, a light source in the tail comes on so the optical sensor on the launcher can track the missile along its flight path. The light source is sufficiently strong to allow automatic guidance to the maximum range of the missile under all conditions in which the missile is visible to the gunner. Figure 1 provides a dependency diagram for the case study.

ANALYZING THE CASE STUDY

To illustrate the effects of using COTS equipment in a system and the effect such use has on testability, we will analyze the case study in three steps. First, we will assume the case study has eight replaceable units but none of these replaceable units are COTS units. Second, we will select two of the replaceable units and assume they are COTS units. Finally, we will assume eight replaceable units are COTS. In each case, we will examine a particular measure of testability which we call "operational isolation." Operational isolation is frequently used in system specifications when identifying required levels of testability for the target system. This measure is described in more detail in the next section.

The Base Case

The system in Figure 1 consists of eight replaceable units, four inputs, and four outputs. The goal of the analysis is to determine the impact that COTS would have on our ability to diagnose the system in the field. We assume that this system as configured is known in detail by the analyst, that is the analyst knows all of the interconnections of the functional units of the system and can observe the internal behavior of each of the replaceable units using the labelled tests.

The diagnosability of the base case will provide a basis for comparison for subsequent analyses. The analyses were run using a tool called STAMP®. STAMP uses

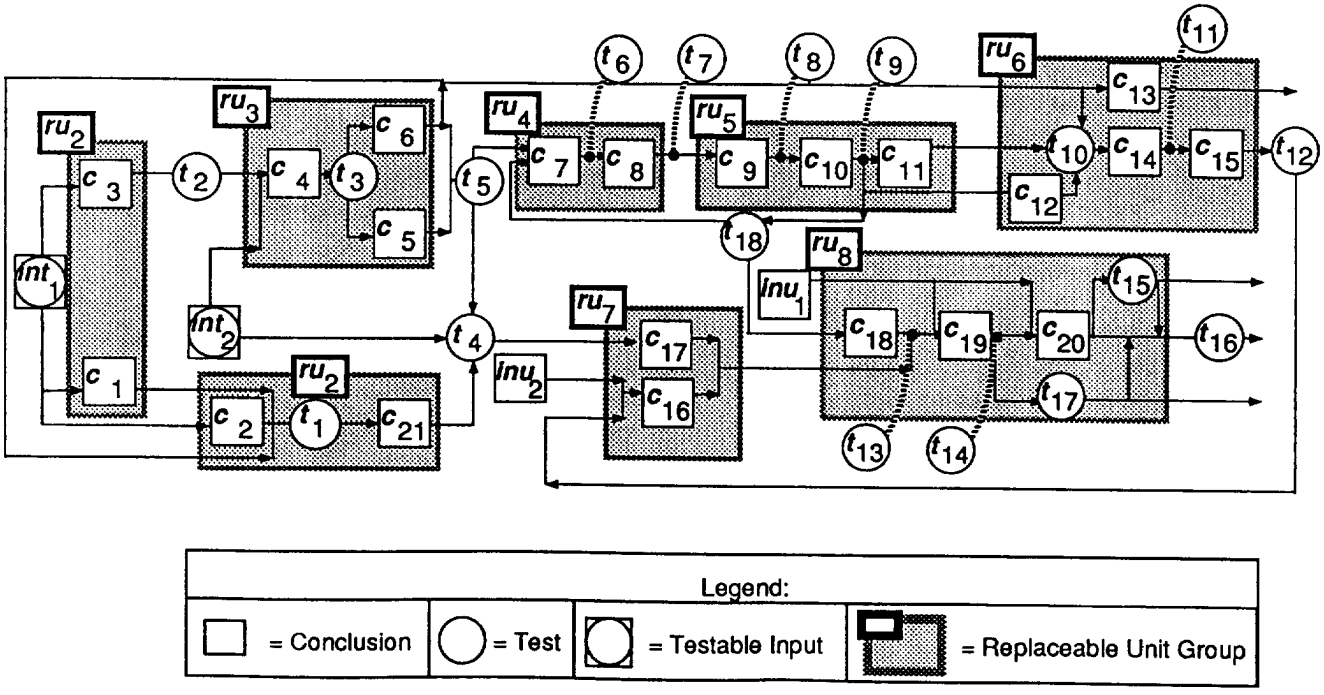

Figure 1. Anti-tank missile launcher case study.

the information flow modeling approach. The system as shown in Figure 1 contains six ambiguity group as shown in Table 1.

Table 1: Ambiguity Groups

Group No	Members	Replaceable Unit Groups (rus)	No of rus
1	c_1, c_2	ru_1, ru_2	2
2*	c_7, c_8, c_9, c_{10}	ru_4, ru_5	2
3	c_{11}, c_{12}	ru_5, ru_6	2
4	c_{12}, No Fault	ru_6, No Fault	2
5	c_{16}, c_{17}, c_{18}, inu_2	ru_7, ru_8, inu_2	3
6	c_{19}, inu_1	ru_8, inu_1	2

* Ambiguity resulting from feedback

Note that in Table 1, the inputs and No Fault are considered replaceable unit groups for purpose of analysis. This means that the case study will have 13 replaceable units (8 defined rus, 4 inputs, and No Fault). Any element not appearing in Table 1 is uniquely isolatable (such as int_1 or c_3) and is isolatable to one replaceable unit. At this point, we will assume a uniform probability of failure for each of the conclusions (that is, any failure in the system is equally likely). Ordinarily when doing such an analysis, one would weight the data based on failure rate, but in this case, choosing uniform weighting prevents us from biasing our results by using a small sample for evaluation. Next, we compute our primary analysis parameter—Operational Isolation (i.e. the percentage of time isolation will result in n or fewer replaceable units). For the base case, OI(1) = 0.3846, OI(2) = 0.8462, and OI(3) = 1.00 meaning we can always isolate to three or fewer replaceable units, but only to one unit 38% of the time.

Partial COTS

In analyzing for the impact of replacing understood elements with COTS units, we next do a blind replacement of 2 replaceable units with COTS elements. The COTS will allow us no visibility into the internal structure. For the first run, we chose ru_3 and ru_5 randomly, thus giving us approximately 25% COTS elements (2 of 8 defined replaceable units). In modeling the COTS unit we utilize the rule of thumb that functional units can have multiple inputs, but only one output. Thus each output of a COTS element is assumed to coincide with a different functional unit. Figure 2 shows how we would model a COTS unit.

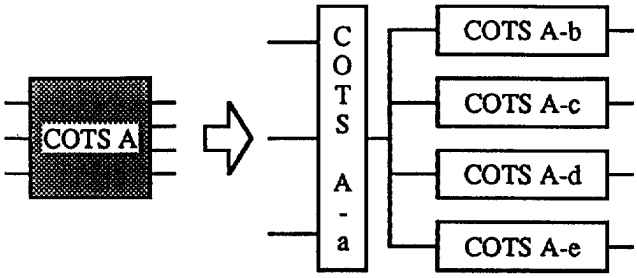

Figure 2. COTS Modeling

A single COTS unit will be broken into several functional units, but an isolation ambiguity with any one of these will result in a replaceable unit ambiguity with the COTS unit. When the revised model was run (with ru_3 and ru_5 defined as COTS), there was no change in the operation isolation values of any level. This would indicate that a few COTS in this system has minimal impact on the diagnosability. To check this, several different COTS replacement units were run and after all trials, a difference of 0-10% in the operational isolation figures was noted, confirming the above observation. While the statistics for the ru_3 and ru_5 COTS replacement are not affected, the maintenance becomes more complex, because in 6 of 26 possible isolations, ambiguities exist between COTS and non-COTS replaceable units. When the COTS units are under warranty repair, this must be resolved by independent test of the non-COTS units, or risk sending good units back for warranty repair.

FULL COTS

To check the maximum impact of COTS on this system, we next replaced all eight replaceable units with COTS and modelled the COTS as outlined above. Significant changes were noted in the operation isolation as shown in Table 2. The decrease in diagnosability is more pronounced, including an increase in ambiguity to four replaceable units. This is accompanied by reduced ability to resolve the diagnosis to 1, 2, or 3 or less replaceable units. The situation is even more critical for maintenance as illustrated by Table 3.

Table 2: Operation Isolation for the Case Study

Operation Isolation	Base Case	25% COTS	100% COTS
OI(1)	0.3846	0.3846	0.2593
OI(2)	0.8462	0.8462	0.6296
OI(3)	1.0000	1.0000	0.7778
OI(4)	1.0000	1.0000	1.0000

Table 3 shows that in 20 out of 27 isolations, an ambiguity exists that contains more than one COTS unit. In this case, we have no choice but to return failed elements for warranty repair that will result in a large number Re Test OKs(RTOKs), unless the manufacturer provided independent test capability for the COTS elements. RTOKs occur when items replaced at one level of maintenance and sent on to the next are found to be fault free at the next level of maintenance.

DISCUSSION

The results are somewhat surprising in that judicious use of COTS subsystems will have a small impact on testability and field maintainability. Even this can be somewhat lessened by purchasing full information packages and/or COTS subsystems with independent self-test capability so that they can be tested independent of the rest of the system. Minimal impact can be achieved when the self-test can be performed *in-situ* (i.e., without removal of the equipment). The results also show that severe problems can be anticipated if the system build up is too reliant on COTS subsystems. The resulting field maintenance complexity may be increased significantly if steps are not taken early in the system development to alleviate the lack of information on COTS subsystems.

CONCLUSION

It is clear, that the use of COTS in terms of units purchased off-the-shelf with minimal information, will affect diagnosability of the systems in which they are used. This effect may be minimal when COTS units make up a small portion of the system, but becomes significant when a large number of COTS units are present.

Because of the impact of COTS on diagnosability, consideration should be made for acquiring detailed data packages when COTS is used in more than minimal situations. An alternative would be to acquire only COTS units that can demonstrate a thorough and system independent self-test capability. From a practical standpoint, this self-test should be performable without equipment removal or modification.

REFERENCES

1. Shortliffe, E., *Computer Based Medical Consultation: MYCIN*, American Elsevier: New York, 1976.

2. Enand, R., J. Pepper, B. Keller, and T. Knutilla, "HIPRIDE: Using Expert Systems to Troubleshoot the HAWK Radar System," *Proceedings of the American Defense Preparedness Association Symposium on Artificial Intelligence Applications for Military Logistics*, Williamsburg, Virginia, March 27-30, 1990.

3. Winograd, Terry, *Language as a Cognitive Process, Volume I: Syntax*, Addison-Wesley: Reading, Massachusetts, 1983.

4. Lenat, D. B., and R. V. Guha, *Building Large Knowledge-Based Systems*, Addison-Wesley, Reading, Massachusetts, 1990.

5. Simpson, William R. and Sheppard, John W., "System Complexity and Integrated Diagnostics," *IEEE Design and Test of Computers*, Volume 8, Number 3, September 1991, pp. 16-30.

6. Sheppard, J. W., and Simpson, W. R., "A Mathematical Model for Integrated Diagnostics," *IEEE Design and Test of Computer*, Volume 8, Number 4, December 1991, pp. 25-38.

7. Simpson, W. R., and Sheppard, J. W., "System testability Assessment for Integrated Diagnostics", *IEEE Design and Test of Computer*, Volume 9, Number 1, March 1992, pp. 40-54.

8. Sheppard, J. W., and Simpson, W. R., "Applying Testability Analysis for Integrated Diagnostics," *IEEE Design and Test of Computers*, Volume 9, Number 3, September 1992, pp. 65-78.

Table 3
Replaceable Unit Ambiguity Groups for the Case Study
All COTS

Failed Element	Isolation Ambiguity	Replaceable Unit Groups	No. of RUs
int1	*int1*	*int1*	1
int2	*int2*	*int2*	1
cots1a	*cots1a cots1b*	*COTS1*	1
cots1b	*cots1a cots1b*	*COTS1*	1
cots1c	*cots1c cots2a cots2b*	*COTS1 COTS2*	2
cots2a	*cots1c cots2a cots2b*	*COTS1 COTS2*	2
cots2b	*cots1c cots2a cots2b*	*COTS1 COTS2*	2
cots3a	*cots3a cots3b cots3c*	*COTS3*	1
cots3b	*cots3a cots3b cots3c*	*COTS3*	1
cots3c	*cots3a cots3b cots3c*	*COTS3*	1
cots4a	*cots4a cots4b cots5a cots5c*	*COTS4 COTS5*	2
cots4b	*cots4a cots4b cots5a cots5c*	*COTS4 COTS5*	2
cots5a	*cots4a cots4b cots5a cots5c*	*COTS4 COTS5*	2
cots5b	*cots5b cots6a cots6c*	*COTS5 COTS6*	2
cots5c	*cots4a cots4b cots5a cots5c*	*COTS4 COTS5*	2
cots6a	*cots5b cots6a cots6c*	*COTS5 COTS6*	2
cots6b	*cots6b No Fault cots8b cots8d*	*No Fault COTS6 COTS8*	3
cots6c	*cots5b cots6a cots6c*	*COTS5 COTS6*	2
cots7a	*cots7a cots7b inu1 cots8a cots8c inu2*	*inu1 inu2 COTS7 COTS8*	4
cots7b	*cots7a cots7b inu1 cots8a cots8c inu2*	*inu1 inu2 COTS7 COTS8*	4
cots8a	*cots7a cots7b inu1 cots8a cots8c inu2*	*inu1 inu2 COTS7 COTS8*	4
cots8b	*cots6b No Fault cots8b cots8d*	*No Fault COTS6 COTS8*	3
cots8c	*cots7a cots7b inu1 cots8a cots8c inu2*	*inu1 inu2 COTS7 COTS8*	4
cots8d	*cots6b No Fault cots8b cots8d*	*No Fault COTS6 COTS8*	3
inu_1	*cots7a cots7b inu1 cots8a cots8c inu2*	*inu1 inu2 COTS7 COTS8*	4
inu_2	*cots7a cots7b inu1 cots8a cots8c inu2*	*inu1 inu2 COTS7 COTS8*	4
No Fault	*cots6b No Fault cots8b cots8d*	*No Fault COTS6 COTS8*	3

9. Simpson, W. R., and Sheppard, J. W., "Fault Isolation in an Integrated Diagnostic environment," *IEEE Design and Test of Computers*, Volume 10, Number 1, March 1993, pp. 65-78.

10. Sheppard, J. W., and Simpson, W. R., "Performing Effective Fault Isolation in Integrated Diagnostics," *IEEE Design and Test of Computers*, Volume 10, Number 2, June 1993, pp. 78-90.

11. R. DePaul, Jr.. "Logic Modeling as a Tool for Testability," *Autotestcon 85 Conference Record*, IEEE Press, New York, 1985, pp. 203-207.

12. R. Cantone and P. Caserta, "Evaluating the Economical Impact of Expert Fault Diagnosis Systems: the I-CAT Experience," *Proceedings Third IEEE International Symposium Intelligent Control*, IEEE Computer Society Press, Los Alamitos California, 1988.

13. J. Franco, "Experiences Gained Using the Navy's IDSS Weapon System Testability Analyzer," *Autotestcon 88 Conference Record*, IEEE Press, New York, 1988, pp. 129-132.

14. K. Pattipati, "START: System Testability and Research Tool," *Autotestcon 90 Conference Record*, IEEE Press, New York, 1990, pp. 395-402.

15. Sheppard, John W., and William R. Simpson, "Automatic Production of Information Models for Use in Model-Based Diagnosis," *Proceedings of the 44th National Aerospace and Electronics Conference*, Dayton, OH, May 1992.