Frame Relay Conformance Testing

At HP's Protocol Test Center, an automatic translator was developed to transform abstract test suites into executable test suites for HP IDACOM protocol analyzers.

by Martin Dubuc

The frame relay protocol is a data transfer protocol defined by the American National Standards Institute (ANSI) and the International Telecommunications Union (ITU). It is similar to the ISDN (Integrated Services Digital Network) standard but it assumes a reliable transmission medium and therefore contains very little error recovery functionality. As a result, it is more straightforward and data transfer is more efficient. At present it is used mainly in North America and Japan. A typical application is LAN-to-LAN interconnection.

The frame relay protocol standard leaves many things undefined and so the protocol is sensitive to issues of implementation. To ensure that different manufacturers' implementations are stable and interoperable, standardized test specifications have been developed that thoroughly test the protocol features. The existence of such widely available conformance test specifications greatly benefits the entire frame relay industry. Early availability of frame relay conformance test systems based on these test specifications provides a common reference point for network implementers, ensuring that migration to frame relay technology is as trouble-free as possible.

Hewlett-Packard frame relay conformance test products address the needs of frame relay networks as they exist today and as they will exist in the near future. These products will evolve to comply with the final standardization of the frame relay protocol, thereby ensuring that tomorrow's equipment will be compatible with the networks that are built today.

This article discusses the development of these products. First, we present the ACT-Frame Relay committee. Then we introduce the basic concepts of test suite design and describe the test implementation methodology. We end by presenting the conformance testing environment available on the HP PT502 protocol tester, and the ACT-Frame Relay T1.617 Annex D conformance test product.

ACT-Frame Relay Standardization Committee

In 1991, the Frame Relay Users Forum (FRUF) created a testing and interoperability group. Its main goal was to ensure the interoperability of frame relay devices. The group was further divided into three subgroups responsible for coordinating the development of conformance tests, defining interoperability, and working with test labs to develop baseline tests. The conformance subgroup decided to use the expertise available in the National ISDN Users Forum (NIUF) for the specification of conformance tests. It set up a frame relay group within the NIUF for that matter. This group was named the ACT-Frame Relay committee (ACT-FR).

The ACT-Frame Relay committee's mandate is to develop abstract conformance test specifications for frame relay products in accordance with the relevant standards and the Frame Relay Users Forum implementers agreements.

The primary goal of the ACT-Frame Relay committee is to write the test specification for basic permanent virtual connection (PVC) protocol implementation for customer premise equipment (CPE). It also intends to address network-to-network interface PVC, and, finally, switch virtual connection (SVC).

The ACT-Frame Relay committee also has the mandate to pursue international standardization by presenting its work to the ITU-T, the technical committee of the ITU working on protocol standardization. HP has played an active role so far in this committee and in the specification of the ACT-FR

Glossary

ATS: Abstract test suite, the test specification document.

ACT-Frame Relay (ACT-FR): Frame relay conformance testing standardization committee that was set up by the Frame Relay Forum testing and interoperability group.

CDL: Constraint Description Language, an ITL extension used to specify the coding of the messages exchanged in an abstract/executable test suite.

CPE: Customer premise equipment; a piece of equipment with telecommunication functionality.

ETS: Executable test suite, an implementation of the abstract test suite that runs on a given test platform.

ISDN: Integrated Services Digital Network.

ITL: Interactive Test Language, the built-in language of the HP PT500 protocol analyzer.

ITU: International Telecommunications Union

PVC: Permanent virtual connection.

SVC: Switch virtual connection.

Test case: a test scenario of the abstract test suite. Each test has a narrow test purpose, and test cases with related test purposes are gathered into the same test group.

TTCN: Tree and Tabular Combined Notation, the test specification language standardized within ISO and ITU. TTCN is part of the OSI conformance methodology and framework (ISO 9646).

UNI: User-to-network interface.

T1.617 Annex D UNI (user-to-network interface) and Q.933 Annex A UNI conformance tests. HP currently holds the editorship of these two test specifications.

Abstract Test Suite Design

An abstract test suite (ATS) is a test specification document that describes tests to be carried out to ensure that a particular implementation conforms to a certain specification. The ATS consists of a collection of tests and the definition of the related components that are used for their specification. The basis of a test is its test purpose and the associated dynamic behavior. The test purpose specifies, in an informal language, the behavior of an implementation under test for a given test scenario. The dynamic behavior part of a test specifies the exact ordering of events that must be accomplished to fulfill the test purpose. Events can be classified in four major categories: sending messages, receiving messages, assignments, and timer operations.

An ATS is said to be abstract because it must make an abstraction of the test platform on which the tests will be executed, as well as the higher and lower protocol layers that are used by the protocol layer(s) to be tested.

Most of today's protocol specifications are state-machine driven. A test designer will often use the state table or SDL (Specification and Description Language, CCITT Z.100) diagrams as a basis for defining a set of tests for such protocols. The ATS should test, to the extent practical, the different situations that an implementation might encounter. The ATS should cover all mandatory and optional features of the implementation. It should also test a good sample of the possible inopportune and erroneous events.

Test cases based on a state matrix are specified in a straightforward manner. All the test designer needs to do in such cases is to call a preamble to get the implementation under test to the starting state, then apply a certain behavior to the implementation, and finally, call a postamble, which verifies that the implementation has jumped into the proper state and optionally brings the implementation into a stable, known state.

In TTCN, the test specification language standardized within the ISO and the ITU, some scenarios are more difficult to specify. For instance, it would be nearly impossible to specify in TTCN the exact events that would be needed to test the behavior of an implementation under heavy loads. In this case, the test purpose might be the only thing that a test designer can specify.

Test Suite Implementation

Once an abstract test suite becomes available, it must be converted into an executable form that will run on a specific test platform. For all of its X.25, ISDN, and frame relay conformance testing products, the HP Protocol Test Center uses the T1/WAN protocol analyzers manufactured by the IDACOM Operation of Hewlett-Packard. The strength of these protocol analyzers lies in their flexible programming and conformance testing environments.

All of the executable test suites that were implemented by HP IDACOM before the creation of the Protocol Test Center are written in the Interactive Test Language (ITL). ITL is the built-in language available on HP IDACOM protocol analyzers. It is a Forth-based language with state-machine constructs that allow the implementation of test scripts (sequence of send and receive messages).

ITL is a convenient test script language, but several problems arise with use of this approach to derive executable test suites. First, to obtain executable test suites in ITL, manual coding of the abstract test suite must be done. This process is usually repetitive and error-prone. Another problem is the lack of tools that would speed development by identifying certain types of errors or by ensuring the correctness of the code. Finally, executable test suites written in ITL are very difficult to maintain.

At a certain point, it became obvious that there was a need for automated tools to help implementers develop executable test suites. Because of resource constraints, however, it was decided that in the first phase the tools could not be automated completely.

The most difficult part of an abstract test suite to implement in an executable test suite is the constraint part (message coding/decoding). This is especially true for protocols such as ISDN or frame relay signaling (layer 3) where the content of the messages is fairly complex. A team investigated this problem and implemented an extension to ITL called the Constraint Description Language (CDL). An automatic TTCN-to-CDL translator was also implemented.

This translator was successfully used with some of the ISDN test suites and the early version of the first ACT-Frame Relay test suite. It helped reduce the time required to implement the test suites by a factor of two. However, CDL was not flexible enough to cope with all the constructs used in TTCN. Furthermore, there was room for more automation since a lot of the coding still needed to be done manually.

The next step was to implement a tool that would allow fully automatic translation of an abstract test suite. A team was formed in the Protocol Test Center to investigate how such a translator could be implemented. A tool, referred to as the TTCN translator, was designed to generate executable test suites from abstract test suites written in TTCN. Fig. 1 shows the user interface of the TTCN translator.

Since ITL had limited flexibility, the C language was selected as the target language of the translator. In parallel, a C cross-compiler and a C loader module were implemented. The cross-compiler allows the generation of object files with instructions compatible with the processor of the targeted protocol analyzer (the processor of the HP PT502 is a Motorola 68000). The C loader module allows C object files to be loaded and executed on the HP PT502.

As shown in Fig. 2, to create an executable test suite, a developer first has to provide an abstract test suite. The ATS can be written from scratch using a TTCN editor (such as the one included in the TTCN environment), or the ATS can be one that was written by a standardization body (for instance, the ACT-FR T1.617 Annex D test suite). The TTCN translator takes an ATS and generates a set of files from it. It produces test suite files, a declaration file, a constraint file, a test step file, and test case files.

The test suite files contain information on the different test cases present in the test suite, as well as information on test

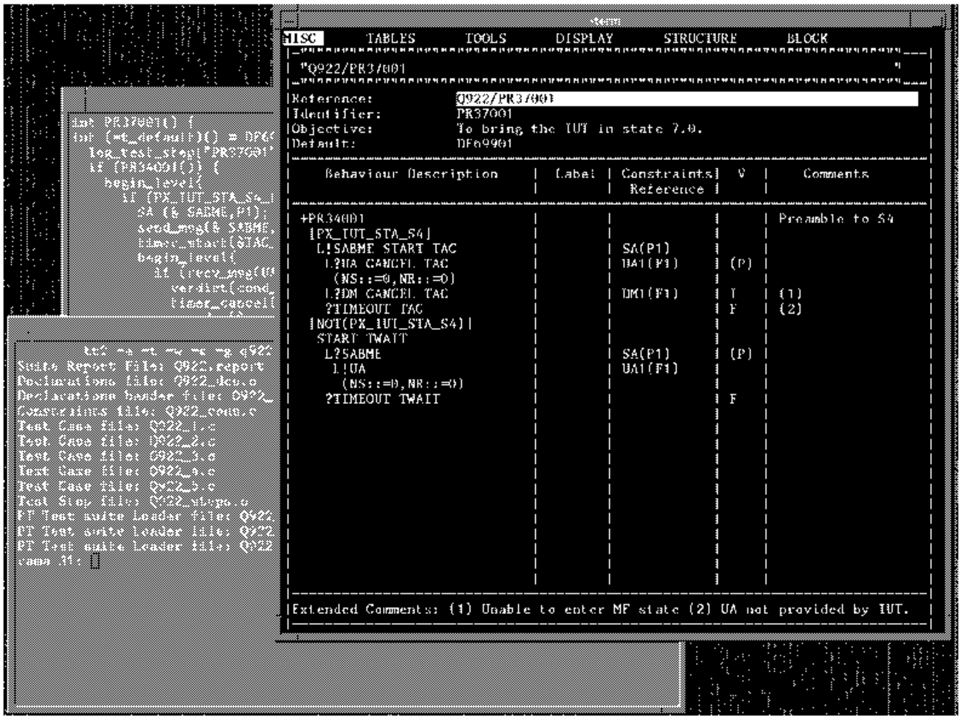

Fig. 1. Conformance testing development environment.

case selection (that is, on which test cases are relevant for testing a specific implementation). The declaration file is a translation of the declaration part of the ATS. It contains all the message structures, timer definitions, and test suite constants, variables, and parameters. The constraint file is the translation of the constraint part of the ATS. It contains the exact coding of all messages to be sent or received. The test step and test case files are the translation of the dynamic behavior of the ATS. The dynamic behavior specifies the ordering of send, receive, assignment, and timer events to

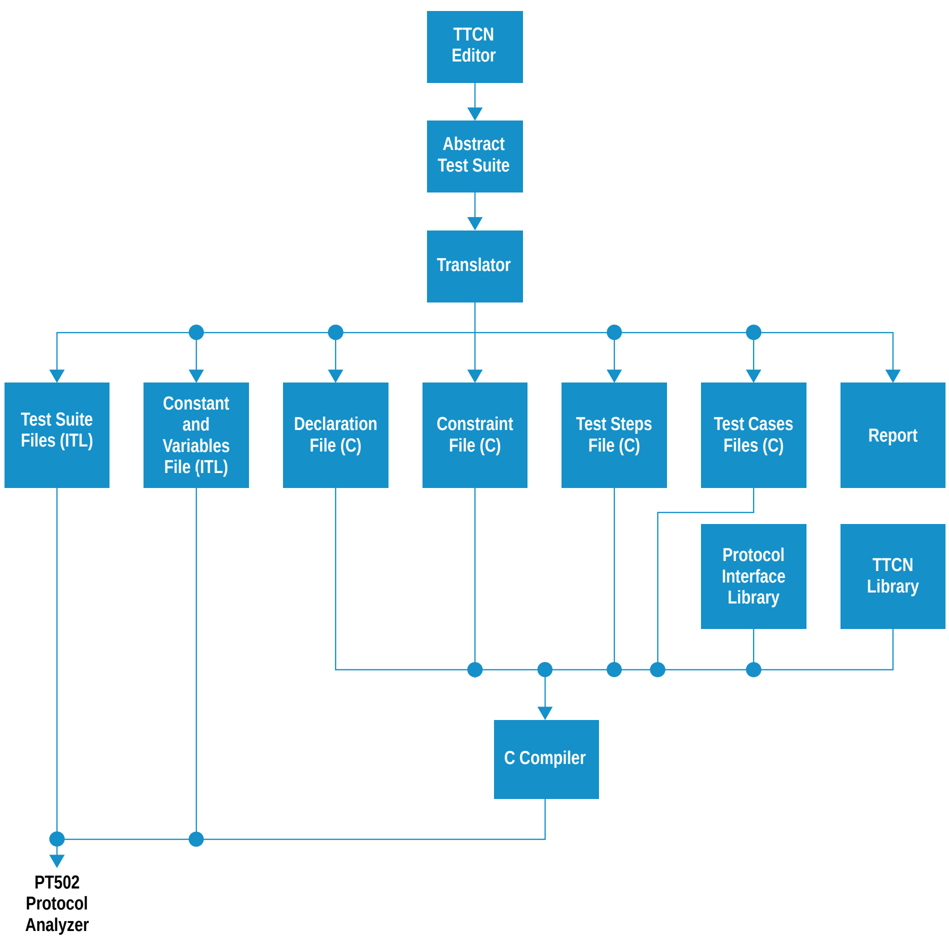

Fig. 2. Executable test suite product life cycle.

```
AP #1:    Frame Relay Emul          Playback RAM       1993–11–05 13:11:53
 Source  DLCI   C/R FECN BECN DE  Info Field
Test Case: PS0_01V STATUS ENQUIRY     Request   1993–11–05 13:10:02
H1 RX    0      0      0     0     0 STATUS_EN
H1 TX    0      0      0     0     0 STATUS
H1 RX    0      0      0     0     0 STATUS_EN
H1 TX    0      0      0     0     0 STATUS
H1 RX    0      0      0     0     0 STATUS_EN
H1 TX    0      0      0     0     0 STATUS
H1 RX    0      0      0     0     0 STATUS_EN
H1 TX    0      0      0     0     0 STATUS
H1 RX    0      0      0     0     0 STATUS_EN
H1 TX    0      0      0     0     0 STATUS
H1 RX    0      0      0     0     0 STATUS_EN
Test Case: PS0_01V STATUS ENQUIRY     Request     Verdict: PASS(0)
>
 -------------------------------------------------------------------------
 rch  ResponseTime  Print  Filters  Triggers  TestScript  TestKeys  TestSuite
 -------------------------------------------------------------------------
|      |f1|      |      |f2|        | |f3|  |  |f4|  |     |f5|     |      |f6|      |     |f7|   |
| Load Suite  | Tester Setup  | PICS | PIXIT  | Selection  | Run Suite  | Reports  |
```

Fig. 3. Conformance testing environment.

specify formally the test purpose of every test case of the ATS.

The TTCN environment provides protocol interface libraries and TTCN libraries, which are necessary to interface the translated code with the protocol tester used to run the executable test suites (HP IDACOM protocol testers in our case). The protocol interface libraries specify, for a specific protocol, what functions of the protocol analyzer are used to send and receive a message and how to initialize the lower layers at the beginning of a test campaign. These parameters are specific to the protocol to be tested, whether the test suite is a frame relay or an ISDN test suite.

Once the files are generated, the C files are compiled using the cross-compiler provided in the TTCN environment, and the resulting object files are linked together along with the TTCN and protocol interface libraries. The test suite files and the object file are then sent to the HP PT500 and the executable test suite (ETS) is ready to be loaded into the conformance testing application.

The TTCN translator has been used internally for a year now, and the results have exceeded expectations. The time needed to implement test suites has been reduced by nearly an order of magnitude or close to ten-to-one. Furthermore, it is now possible to address more complex test suites, such as test suites meant to test a network.

PT502 Conformance Testing Environment

The screen display of the HP PT502 running the ACT-FR T1.617 Annex D ETS is shown in Fig. 3. The conformance testing environment is the environment in which the user, also called the test operator, sets up and launches a test campaign and generates test reports. Several services are available to the user in this environment.

First, the test operator can specify the different system parameters that characterize the implementation under test using the PICS and PIXIT screen menus. This will help the protocol tester identify which test cases are relevant to a specific implementation.

Once the configuration is done, the test operator can set up a test campaign by selecting the test cases to be executed. Test cases can be selected through the test group and case selection menus using different criteria: select a whole test suite, select/deselect a whole test group, select/deselect individual test cases, select by verdict assignment (for instance, select all test cases that previously failed).

During the execution of a test campaign, the protocol tester records verdicts for each test that it runs. In addition, traces can be recorded for each test, or they can be recorded according to the verdict assigned to the test. For instance, the traces of all failed tests can be recorded.

When a test campaign is finished, a test report can be printed. The tester reports information about the implementation configuration, as well as a summary of the campaign, which includes the verdicts for each test. It can also report the full traces of tests with given verdicts.

Traces can be displayed, printed, or played back using different formats. The format menu can be used to specify whether the traces should be displayed in hexadecimal, short, or complete forms. Optionally, if the complete format is specified, the decoding of each individual information element can be done. The time stamp can also be added in the trace format. In Fig. 3, the short format was used. Fig. 4 shows the trace of a STATUS message with complete decoding and full message detail.

ACT-FR T1.617 Annex D Test Suite

The ACT-FR T1.617 Annex D ATS was the first ATS specified by the ACT-Frame Relay committee. This test suite was aimed at testing the permanent virtual connection (PVC) management procedures for the user-to-network interface (UNI). It covers the following areas:

- Notification of the addition of a PVC
- Detection of the deletion of a PVC
- Notification of availability or unavailability of a configured PVC
- Link integrity verification.

Table I shows the test suite structure for the ACT-FR T1.617 Annex D ATS. The two main groups (periodic polling and bidirectional) are each further divided into three subgroups (general, error, and system).

Table I
ACT-FR T1.617 Annex D Test Suite Structure

Test Group Name	Number of Test Cases in Test Group
Periodic Polling/General	24
Periodic Polling/Error	42
Periodic Polling/System	3
Bidirectional/General	9
Bidirectional/Error	13
Bidirectional/System	1

The periodic polling group contains the test cases related to the testing of the user-side link integrity verification procedure, while the bidirectional group contains test cases related to the testing of the bidirectional network procedures. The latter procedures are usually implemented on network equipment (and thus can be used to test a network-to-user interface), but can be supported optionally by a user's equipment.

```
MM:SS:ssss Source  DLCI  C/R   FECN  BECN  DE  Info    Field
18:40.2420    H1 TX   0      0      0      0      0   UI        P=0

    PD = T1.617   Dummy Call Reference           STATUS
  1   1001––––  INFORMATION ELEMENT    :  SHIFT
      ––––0–––  Shift type             :  locking
      –––––101  Codeset ident.         :  national use IE
  1   00000001  INFORMATION ELEMENT    :  REPORT TYPE
  2   00000001  IE length              :  1 octet
  3   00000000  Report Type            :  full status message
  1   00000011  INFORMATION ELEMENT    :  LINK INTegrity VERification
  2   00000010  IE length              :  2 octets
  3   00001000  Send Seq Number        :  8
  4   00010111  Rcvd Seq Number        :  23
  1   00000111  INFORMATION ELEMENT    :  PVC STATUS
  2   00000011  IE length              :  3 octets
  3   0–––––––  Extension bit          :  continued
      –0––––––  Spare                  :  don’t care
      ––000001  PVC DLCI               :  partial value
 3A   1–––––––  Extension bit          :  not continued
      –0000–––  PVC DLCI               :  16
      –––––000  Spare                  :  don’t care
  4   1–––––––  Extension bit          :  not continued
      –000––––  Spare                  :  don’t care
      ––––0–––  PVC Status             :  PVC is already present
      –––––0––  Spare                  :  don’t care
      ––––––1–  PVC Active             :  PVC is active
      –––––––0  Spare                  :  don’t care
  1   00000111  INFORMATION ELEMENT    :  PVC STATUS
  2   00000011  IE length              :  3 octets
  3   0–––––––  Extension bit          :  continued
      –0––––––  Spare                  :  don’t care
      ––000001  PVC DLCI               :  partial value
 3A   1–––––––  Extension bit          :  not continued
      –0001–––  PVC DLCI               :  17
      –––––000  Spare                  :  don’t care
  4   1–––––––  Extension bit          :  not continued
      –000––––  Spare                  :  don’t care
      ––––0–––  PVC Status             :  PVC is already present
      –––––0––  Spare                  :  don’t care
      ––––––0–  PVC Active             :  PVC is inactive
      –––––––0  Spare                  :  don’t care
  1   00000111  INFORMATION ELEMENT    :  PVC STATUS
  2   00000011  IE length              :  3 octets
  3   0–––––––  Extension bit          :  continued
      –0––––––  Spare                  :  don’t care
      ––000001  PVC DLCI               :  partial value
 3A   1–––––––  Extension bit          :  not continued
      –0010–––  PVC DLCI               :  18
      –––––000  Spare                  :  don’t care
  4   1–––––––  Extension bit          :  not continued
      –000––––  Spare                  :  don’t care
      ––––1–––  PVC Status             :  PVC is new
      –––––0––  Spare                  :  don’t care
      ––––––1–  PVC Active             :  PVC is active
      –––––––0  Spare                  :  don’t care
18:40.2422
```

Fig. 4. Trace display of a STATUS message using the complete format.

The general subgroup gathers the test cases related to valid events. In these test cases, the tester simulates a normal exchange of messages. The error subgroup contains test cases that simulate network equipment that does not conform to the standard and verifies that the implementation reacts according to the specification. In these test cases, the tester sends messages with content error, and verifies that the implementation under test rejects either the whole message or the erroneous part. The system subgroup contains system tests and tests that verify the system parameters of the implementation under test.

It is worth noting that often part of a standard is too vague. How should the equipment react under such circumstances? An ATS can sometimes provide additional guidelines on these points, and the ACT-FR T1.617 Annex D ATS is a good example. The implementation behavior under certain error conditions is not specified in the standard. The ACT-FR committee has worked jointly with the T1S1.2 working group to fill in the holes in the standard. The ATS was specified according to the results of this process. The T1S1.2 group plans to update the standard accordingly, but in the meantime only the ACT-FR ATS provides the intended interpretation of the standard.

The ACT-FR T1.617 Annex D ETS for the HP PT502 can be used for different purposes. It is an excellent tool for frame relay equipment implementers for pretesting and preacceptance purposes. The implementer can use the ETS at different stages of development—for instance, to test features incrementally as they are implemented. The ETS can also be used for regression testing, to ensure that changes or the addition of new features to the frame relay equipment have not introduced deficiencies in the implementation. The ETS tests most aspects of the specification.

The test suite can also be used by carriers to test frame relay equipment before it is hooked to a network (acceptance testing). In this way, the carriers can ensure a certain degree of interoperability among the different instruments of their network.

Finally, along with other analysis packages available on the HP PT502, the ETS can be used for troubleshooting by network managers.

Future Directions

ACT-FR is now ready to tackle its next assignments, namely, development of additional procedures for permanent virtual connection (network-to-network interface), switch virtual connection data link layer (LAPF), and switch virtual connection signaling. Hewlett-Packard intends to continue its role in the ACT-FR committee and to release an ETS for each ATS approved therein.

Conclusions

Use of conformance testing in the early steps of the product development life cycle speeds overall time-to-market of communicating devices. It also increases the probability of interoperability between products from different vendors.

In the past, conformance test software was coded manually. The process was repetitive and error-prone. In today’s competitive market, these methods are not good enough to provide customers in a timely fashion with conformance test software to be used in-house for pretesting.

The HP Protocol Test Center R&D team has implemented a set of productivity tools to automate the implementation of executable test suites. These tools have dramatically reduced the R&D efforts needed to produce executable test suites. They have also enhanced the quality of the software and contributed to the detection of problems in the test specifications by underlining inconsistencies.

The ACT-Frame Relay committee has adopted the test specifications for the additional procedures for permanent virtual connections (user-to-network interface). With the help of the TTCN translator, Hewlett-Packard was the first company to offer a product for those test specifications.