IBM Technical Disclosure Bulletin Vol. 34 No. 7A December 1991

# MECHANISM FOR NON-DISRUPTIVE DYNAMIC TESTING OF A COMMUNICATIONS NETWORK

This article describes a selective mechanism which can test special functions, such as fault-detection mechanisms in a communications network, without having to take components out of service to perform the tests. Special test messages can be interspersed with normal messages in any way without disrupting normal operations.

When testing a communications network, three parties are involved: a sending party, a receiving party, and a controlling entity. The controlling entity will request the sending party to send a "test message" to the receiving party. The sending party will send the test message to the receiving party, which will report the receipt to the controlling party. Sometimes it is desirable to be able to force errors into the network to determine if the network can correctly detect and handle those errors. When the sender is set into a mode where it acts in an erroneous manner, so as to force errors, it is out of service for normal traffic and must be dedicated to the test environment.

A fourth party, the "fault injector," is introduced in the network. The fault injector is placed in the path of data traffic between the sender and the receiver. It may be part of the sender, the receiver, or a separate component. All traffic passes through the fault injector, and under normal conditions, the fault injector is transparent, not affecting the data in any way. As messages pass through the fault injector, they are tested for the presence of a test flag. If no test flag is present, the message is passed unchanged. If a test flag is present, a specified fault is injected into the message.

When a test is to be performed, the controlling entity sets up conditions in the fault injector. Conditions may include information such as what kind of fault to inject, for example, framing, CRC, or flag errors. The test mode is armed, but not activated. Although the fault injector is set into a mode where it can act in an erroneous manner, it does not insert faults into the network. The controlling entity then instructs the sending party to send a "test message" to the receiving party. A test message is a normal message that contains a flag to identify it as a test message. As the test message passes through the fault injector, the armed test mode is triggered and the message is changed to include a fault. Upon receipt of the message, the receiving party detects the error, and, noticing that the error was in a test message (because of the presence of a test flag), informs the controlling entity that the message was received with errors identified.

An extension of this mechanism is possible where the information, such as what kind of error to inject, is included in the test message. There are three advantages to this method. First, the controlling entity does not have to perform a separate operation to set up the fault injector. Second, the fault injector can handle many different test messages individually with the capability of injecting different faults in each of the separate messages. Third, the receiver is aware of the type of error expected (the error information is included in the message), and can therefore better process the message.