Evaluation Methodology for Software Based Leak Detection Systems

API PUBLICATION 1155
FIRST EDITION, FEBRUARY 1995

American Petroleum Institute
1220 L Street, Northwest
Washington, D.C. 20005

Evaluation Methodology for Software Based Leak Detection Systems

Manufacturing, Distribution and Marketing Department

API PUBLICATION 1155
FIRST EDITION, FEBRUARY 1995

American Petroleum Institute

SPECIAL NOTES

1. API PUBLICATIONS NECESSARILY ADDRESS PROBLEMS OF A GENERAL NATURE. WITH RESPECT TO PARTICULAR CIRCUMSTANCES, LOCAL, STATE, AND FEDERAL LAWS AND REGULATIONS SHOULD BE REVIEWED.

2. API IS NOT UNDERTAKING TO MEET THE DUTIES OF EMPLOYERS, MANUFACTURERS, OR SUPPLIERS TO WARN AND PROPERLY TRAIN AND EQUIP THEIR EMPLOYEES, AND OTHERS EXPOSED, CONCERNING HEALTH AND SAFETY RISKS AND PRECAUTIONS, NOR UNDERTAKING THEIR OBLIGATIONS UNDER LOCAL, STATE, OR FEDERAL LAWS.

3. INFORMATION CONCERNING SAFETY AND HEALTH RISKS AND PROPER PRECAUTIONS WITH RESPECT TO PARTICULAR MATERIALS AND CONDITIONS SHOULD BE OBTAINED FROM THE EMPLOYER, THE MANUFACTURER OR SUPPLIER OF THAT MATERIAL, OR THE MATERIAL SAFETY DATA SHEET.

4. NOTHING CONTAINED IN ANY API PUBLICATION IS TO BE CONSTRUED AS GRANTING ANY RIGHT, BY IMPLICATION OR OTHERWISE, FOR THE MANUFACTURE, SALE, OR USE OF ANY METHOD, APPARATUS, OR PRODUCT COVERED BY LETTERS PATENT. NEITHER SHOULD ANYTHING CONTAINED IN THE PUBLICATION BE CONSTRUED AS INSURING ANYONE AGAINST LIABILITY FOR INFRINGEMENT OF LETTERS PATENT.

5. GENERALLY, API STANDARDS ARE REVIEWED AND REVISED, REAFFIRMED, OR WITHDRAWN AT LEAST EVERY FIVE YEARS. SOMETIMES A ONE-TIME EXTENSION OF UP TO TWO YEARS WILL BE ADDED TO THIS REVIEW CYCLE. THIS PUBLICATION WILL NO LONGER BE IN EFFECT FIVE YEARS AFTER ITS PUBLICATION DATE AS AN OPERATIVE API STANDARD OR, WHERE AN EXTENSION HAS BEEN GRANTED, UPON REPUBLICATION. STATUS OF THE PUBLICATION CAN BE ASCERTAINED FROM THE API AUTHORING DEPARTMENT [TELEPHONE (202) 682-8000]. A CATALOG OF API PUBLICATIONS AND MATERIALS IS PUBLISHED ANNUALLY AND UPDATED QUARTERLY BY API, 1220 L STREET, N.W., WASHINGTON, D.C. 20005.

FOREWORD

API publications may be used by anyone desiring to do so. Every effort has been made by the Institute to assure the accuracy and reliability of the data contained in them; however, the Institute makes no representation, warranty, or guarantee in connection with this publication and hereby expressly disclaims any liability or responsibility for loss or damage resulting from its use or for the violation of any federal, state, or municipal regulation with which this publication may conflict.

Suggested revisions are invited and should be submitted to the director of the Manufacturing, Distribution and Marketing Department, American Petroleum Institute, 1220 L Street, N.W., Washington, D.C. 20005.

American Petroleum Institute
Evaluation Methodology for Software Based Leak Detection Systems

UTSI International Corporation
1560 West Bay Area Boulevard, Suite 300
Friendswood, Texas 77546

ACKNOWLEDGMENT

UTSI would like to thank the many pipeline companies, application software and SCADA system vendors, independent reviewers, and members of the API Leak Detection Task Force for their support throughout this project. Without their assistance, this document could not have been produced.

As with any technical document, it is possible that there may be cases where further clarification or additional depth is warranted. If necessary, and within reason, UTSI International Corporation will issue a revision to these procedures. Please address comments to UTSI International Corporation, Pipeline Consulting Division, 1560 West Bay Area Boulevard, Suite 300, Friendswood, Texas, 77546, or send email to *pipecons@utsi.com*.

TABLE OF CONTENTS

Chapter One

Executive Summary

In December, 1992 the American Petroleum Institute (API), General Committee on Pipelines, approved a project to address the needs of pipeline companies regarding the evaluation of software-based leak detection systems. The purpose of the project was to define a uniform methodology which could be employed by pipeline companies as an aid for the evaluation of software-based leak detection systems. UTSI International Corporation was contracted by the API to define and develop the evaluation methodology to be consistent with the needs of pipelines and vendors and to supervise validation testing. Development of this methodology has been made possible through the cooperation of many pipeline companies and vendors of software-based leak detection systems.

Validation of the procedures was an integral part of the methodology development project. The Validation Process was designed to apply the concepts established during the project to real pipeline systems, and to evaluate the effectiveness of those concepts as an aid to the task of selecting a leak detection solution for a given pipeline system.

The procedures defined within this document have been developed under the premise that one uniform set of information, consisting of a description of the pipeline's physical configuration, samples of run-time or simulated data, and definition of the pipeline company's performance expectations will allow any software company to estimate expected leak detection system performance in a more efficient, and presumably a lower cost manner. Results from each vendor's analysis are to be compiled into a standardized format for presentation to the pipeline company so that interpretation of the vendor's product capabilities can be easily achieved.

A pipeline company wishing to evaluate a given vendor's leak detection system must provide the vendor with the following information related to the subject pipeline system or systems:

- A pipeline configuration file,
- Data files and supporting information for sample test scenarios, and
- A specification of the desired leak detection system performance.

The vendor will use the information in the configuration file to develop a suitable representation of the pipeline. The vendor's software-based leak detection system will then be installed to operate on the simulated pipeline, and tests will be performed using the operational data provided in the data files. Once the vendor has completed the analysis, a report will be prepared defining the observed performance of the software based leak detection system with respect to the performance metrics, configuration definition, and the representative data sets specified by the pipeline company.

This document contains step-by-step procedures for defining pipeline configurations, run-time data samples, and performance specifications. In addition, the procedures include a definition of the recommended vendor report format and suggestions for its interpretation.

Overview

Within the context of these procedures, it is assumed that leak detection systems are acquired for the primary purpose of detecting leaks if and when they occur. A natural extension of that primary purpose is to estimate the size and location of a leak, if one should be detected. An equally natural extension is to detect and respond to situations that might impair the ability of the leak detection system to perform its primary function. Thus the selection of a system for application on a specific pipeline involves evaluation of the expected leak detection performance, as well as operational features and functions that might add to the utility of the system but do not directly improve leak detection performance. The selection process might also include commercial and economic criteria such as system cost, support, ease of maintenance, and so forth. Since the commercial analysis process for any given project can vary significantly between operating companies, it is considered to be beyond the scope and intent of this document, and will not be addressed further herein.

The procedures presented herein acknowledge that there are physical limits to the size of leak that can be detected. Furthermore, the task of determining the best leak detection solution for a given pipeline always involves performance tradeoffs. In some cases, these tradeoffs must be made before the leak detection system is placed into operation. Ideally, a vendor could state exactly how their leak detection system would perform on a given pipeline configuration, prior to its installation. Often this is not possible due to unavailable or incomplete information about the physical pipeline and its operation.

The focus of these procedures is to establish an evaluation methodology that can be adapted to a wide range of pipeline systems and operating company requirements. The intent is to provide a vehicle that can be adapted to accommodate future changes in leak detection techniques or pipeline operations.

The evaluation procedure consists of the following six parts:

1. Gathering information and describing the physical pipeline,

2. Assembling case files and operational data samples,

3. Specifying and prioritizing performance metrics,

4. Analyzing and projecting leak detection system performance,

5. Reporting projected performance in a standard format, and

6. Appraising the performance of the leak detection system for the pipeline.

Parts 1, 2, and 3 are executed by the pipeline company. The pipeline configuration data, operational data and performance requirements are then submitted to the potential leak detection vendors, who execute parts 4 and 5, and provide their results to the pipeline company. The company then executes part 6 to determine the suitability of a given leak detection system for the target pipeline(s).

A keyword-oriented file structure has been developed to facilitate specification of the pipeline configuration. A data file format has also been defined, along with standards for specification of performance criteria and presentation of results. A discussion of the reliability, sensitivity, accuracy and robustness performance metrics is provided as an aid to the pipeline engineer, along with a glossary of leak detection terminology.

The step-by-step procedures presented in Chapter Six require that the pipeline engineer be familiar with the supporting information contained within Chapters Two through Five. The supporting information is referred to throughout the procedures, and must be used for the preparation of information to be transmitted to the software based leak detection system vendor(s).

Chapter Two

Overview of Pipeline Leak Detection

Pipeline leak detection is treated herein as a classical problem in parameter estimation. In other words, the leak detection system estimates parameters based upon measurement data. The leak parameter estimates are then examined to decide if they warrant the issuance of an alarm to indicate the likely presence of an actual leak. Note that the estimated parameters depend upon the nature of the leak detection system. These might include the leak flow rate, amount of fluid lost, magnitude of pressure or flow disturbance at the leak site, most probable location of the leak, and so forth. Virtually all systems that make a statistical decision based upon a set of measurement data can be discussed within the framework of this model.

As represented in Figure 1, the detection problem is a step-wise process that logically separates each of the system's components in terms of their relationship to the desired result. The software based leak detection system relies on data values acquired from some reliable source, usually the real-time SCADA system, to provide a series of data sets representative of actual conditions at any given point in time. Once data for a given time period has been acquired, it is subjected to some pre-determined mathematical or statistical analysis process that generates additional data based on an assumed model of the pipeline and its associated parameters. Results from the analysis process are produced in the form of parameter estimates. These parameter estimates are in turn subjected to some probability law or decision criteria to determine if a leak does indeed exist. In the simplest case, a given set of data can represent one of two possible outcomes; the existence of a leak or the absence of one. Typically, the process requires an examination of many complex data interrelationships in order to provide acceptable results. Depending upon the nature of the leak detection system, this examination might be done over a small window in time or it could involve periods of several minutes or even hours. In some cases, the time required to make a decision might also depend upon the size of the leak, if one should occur.

The phrase "model of the pipeline" is used here and throughout this document in the most general sense. Some vendors and client companies tend to group software based leak detection systems into the two categories "model based" and "not model based", depending upon whether or not the system involves a fluid dynamics model. In fact, this is an incomplete characterization. Fluid dynamics models employ one or more of the basic equations of fluid mechanics, which include the equations of continuity, momentum, and energy. However, there are a number of software based leak detection methods, all of which are based upon some set of rules or "model" describing the pipeline operation. It is this set of rules that determines how such systems use the measurement data to make decisions.

Leak Detection Performance

Determination of the presence or absence of a leak requires that the software based leak detection system has prior knowledge of the problem to be solved and some pre-determined criteria upon which to base its decision. In the most general sense, there are four possible outcomes each time the leak hypothesis is tested:

1. The system correctly indicates that there is no leak,

2. The system correctly indicates that there is a leak,
3. The system incorrectly indicates that there is a leak, and
4. The system incorrectly indicates that there is no leak (failure to detect).

Outcomes 1 and 2 constitute proper operation of the leak detection system whereas outcomes 3 and 4 constitute failure of the system. In the ideal system outcomes 3 and 4 never occur.

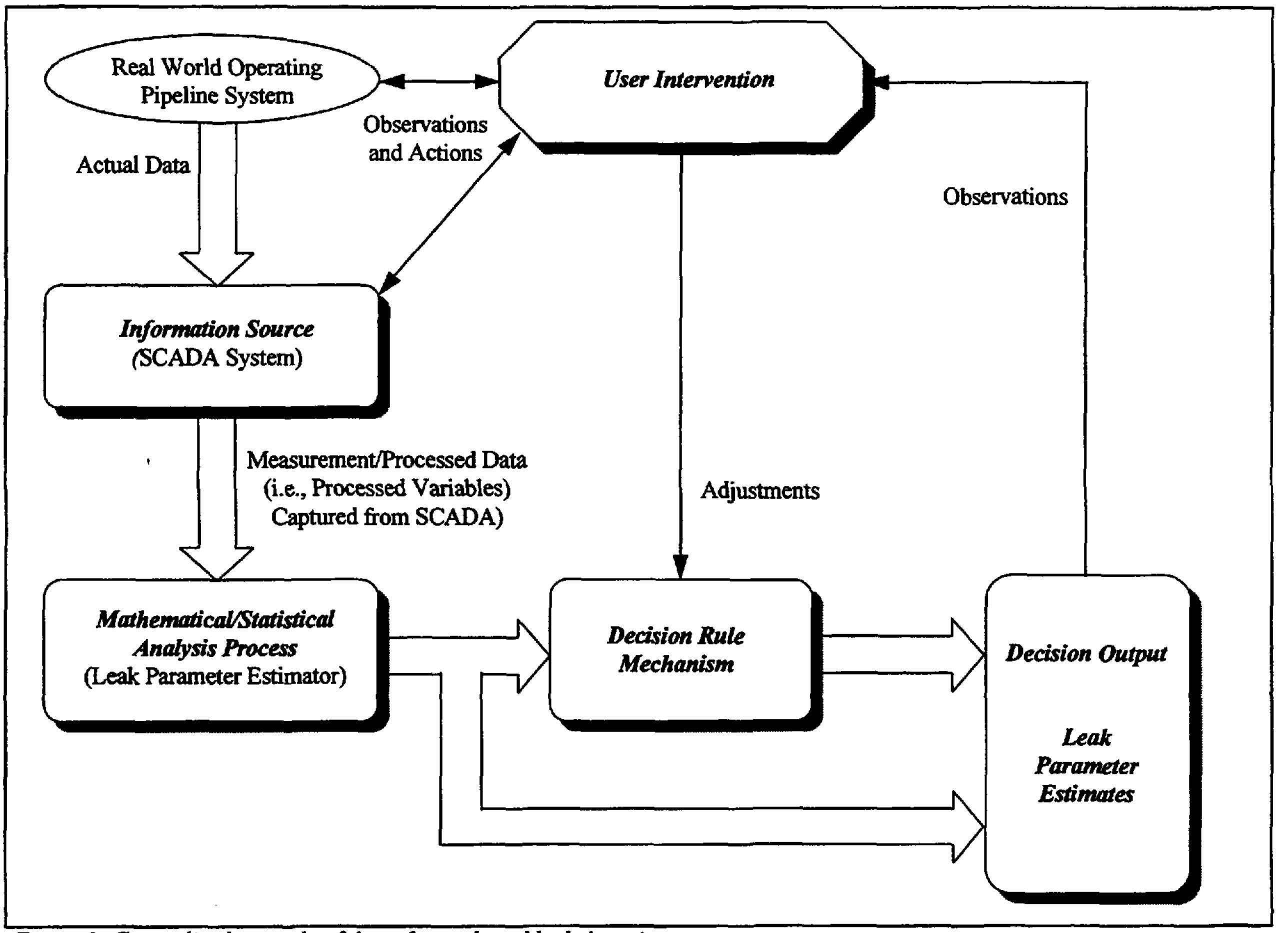

Figure 1: Generalized example of the software based leak detection process.

To further characterize the ideal leak detection system, one must recognize the importance of an accurate and timely response in the event that a leak occurs. With this in mind, the characteristics of an "ideal leak detection system" are easily stated. Such an ideal system would always and immediately detect any leak that might occur and it would never incorrectly declare a leak. Furthermore, it would always and immediately provide an accurate estimate of the location and size of any leak. Of course, there are no known software based leak detection systems that currently provide this ideal level of performance. Furthermore, certain characteristics of the "ideal leak detection system" impose conflicting requirements upon practical leak detection systems. For that reason, it is not likely that such an ideal system can ever be achieved in practice. Thus the task of determining the best leak detection solution for a given pipeline always involves performance tradeoffs. In many cases these tradeoffs must be judged before the leak detection system is actually installed on the pipeline. In some cases, the leak detection system may be selected before the pipeline itself is placed into operation. Once installed, periodic adjustments of leak detection system parameters might be necessary to account for operational experience, configuration changes, and so forth.

Appraisal of Leak Detection System Performance

Determining the level of performance that can be expected from a software based leak detection system is a process that involves several factors, some of which may not be within the control of the pipeline company or the leak detection vendor. Many implementations assume that a certain degree of error will exist within the specification of the pipeline and with the measurements taken during operation, and provide algorithms to compensate for such inconsistencies. Additionally, some vendors require that the system be subjected to a tuning period during the installation process, so that adjustments to the configuration and the corresponding compensation algorithms can be made using real-time pipeline measurement information. Although this can sometimes be a tedious and time consuming process, it is generally accepted in the case of the more complex solutions, in that the ultimate outcome produces more accurate and reliable results.

Ideally a vendor, given accurate information, could state exactly how their software based leak detection system would perform on a given pipeline configuration, prior to its installation. In practice, this is sometimes difficult due to unavailable or incomplete information regarding the physical pipeline and its operation. The focus of this report is to identify a set of metrics that can be used to quantify performance projections, to suggest a method for the specification and prioritization of these metrics, and to define a procedure for the evaluation of leak detection system performance that can be adapted to a wide range of pipeline systems and operating company requirements.

Chapter Three

Output Data and Performance Metrics

This chapter deals primarily with the performance related aspects of software based leak detection systems. Selection of a software based leak detection system for a given application involves evaluation of the expected (or estimated) performance of the system, as well as operational features and functions that might add to the utility of the system but do not directly improve leak detection performance. The selection process for a specific pipeline system might also include commercial and economic criteria such as system cost, support, ease of maintenance, and so forth.

Any appraisal of leak detection system performance involves an assessment of the various tradeoffs that must be made when the system is installed. In practice, real and potential costs are incurred for each incorrect alarm, missed alarm, late alarm, and/or any other deviation from ideal leak detection system performance. Any evaluation of costs and liabilities associated with improper alarming is beyond the scope of this document. In order to establish appropriate performance criteria, the client pipeline company must perform their own assessment and understand the implications of that assessment with respect to the various categories of leak detection performance.

Performance of a software based leak detection system is tantamount to its ability to recognize leak conditions rapidly and without failure, so as to minimize fluid loss, property damage and the risk of personal injury. But this definition of performance is too broad to be useful to determine projected performance of a leak detection system on a given pipeline or set of pipelines. To that end, it is first necessary to decompose the broad definition of performance into more specific components. With that goal in mind, a wide range of criteria used by pipeline companies and vendors in the specification of leak detection system performance have been examined. These performance criteria can be grouped into four categories, or metrics, that determine a system's *reliability, sensitivity, accuracy,* and *robustness*. A definition and discussion of each of these performance metrics follows.

Reliability

Reliability is defined as a measure of the ability of a leak detection system to render accurate decisions about the possible existence of a leak on the pipeline, while operating within an envelope established by the leak detection system design. It follows that reliability is directly related to the probability of detecting a leak, given that a leak does in fact exist, and the probability of incorrectly declaring a leak, given that no leak has occurred. A system is considered to be more reliable if it consistently detects actual leaks without generating incorrect declarations. Conversely, a system which tends to incorrectly declare leaks is often considered to be less reliable. This is particularly true in cases where it is difficult for the pipeline operator to distinguish between actual leaks and incorrect declarations. On the other hand, a high rate of incorrect leak declarations might be considered less significant if the pipeline operators have access to additional information that can be used to verify or disqualify a leak alarm.

Systems that limit or inhibit alarm generation in response to certain conditions of pipeline operation are not necessarily less reliable. Reliability pertains only to the functionality of the leak detection software without regard to SCADA system performance, availability of the pipeline instrumentation and communication equipment, or any other factor beyond the control of the leak detection system vendor. Such factors involve a separate category of performance, namely robustness.

The reliability of a leak detection system usually depends upon a number of parameter settings (e.g. decision thresholds, filter characteristics, and so forth) as well as all of the suitable leak detection techniques employed for the operational characteristics of the target pipeline system. In some cases, a pipeline operator must decide whether to use settings that cause frequent alarms during normal pipeline operations, or to use other settings that are less likely to cause alarms, but might delay or even fail to alarm when a leak is present. Many systems also make automatic adjustments to decision thresholds and other parameters in order to reduce the likelihood of generating alarms during defined operating conditions. When such adjustments are made, a corresponding penalty is normally incurred in some other aspect of performance. For example, decisions based on longer observation intervals might make a particular system less susceptible to random instrumentation errors or disturbances caused by normal pipeline operations, but this performance gain is achieved at the expense of longer response time and the risk of greater fluid loss if a leak should occur.

Reliability can be managed through the use of operator response criteria and procedures. Such procedural methods, unless embodied within the leak detection software itself and performed automatically by the system, do not serve to discriminate between leak detection systems with regard to performance. On the other hand, if additional information is available from the leak detection, SCADA, or other systems, then reliability may be better managed.

Sensitivity

Sensitivity is defined as a composite measure of the size of leak that a system is capable of detecting, and the time required for the system to issue an alarm in the event that a leak of that size should occur. The relation between leak size and response time is dependent upon the nature of the leak detection system. In some cases, as illustrated in Figure 2, there is a wide variation in response time as a function of leak size. In other cases the response time is relatively independent of leak size, as depicted in Figure 3. However, there are no known systems that tend to detect small leaks more quickly than large leaks.

To further illustrate this definition of sensitivity, consider a hypothetical case involving four different leak detection systems (Red, Green, Blue, Yellow) with the following projected levels of sensitivity on a given pipeline:

Red System: This system is capable of detecting a small leak within 5 minutes of the start of the leak.

Green System: This system is capable of detecting a small leak within 15 minutes of the start of the leak.

Blue System: This system is capable of detecting a large leak within 5 minutes of the start of the leak.

Yellow System: This system is capable of detecting a large leak within 15 minutes of the start of the leak.

On the basis of these performance projections it is obvious that the Red System is the most sensitive and that the Yellow System is the least sensitive. However, comparison of the Green and Blue Systems is less apparent. It is possible that for one pipeline the Green System might be more appropriate, whereas for another pipeline the Blue System is more applicable. Since some leak detection systems manifest a strong correlation between leak size and response time, it is also possible that the two levels of sensitivity shown for the Green and Blue Systems could be manifested by the same leak detection system.

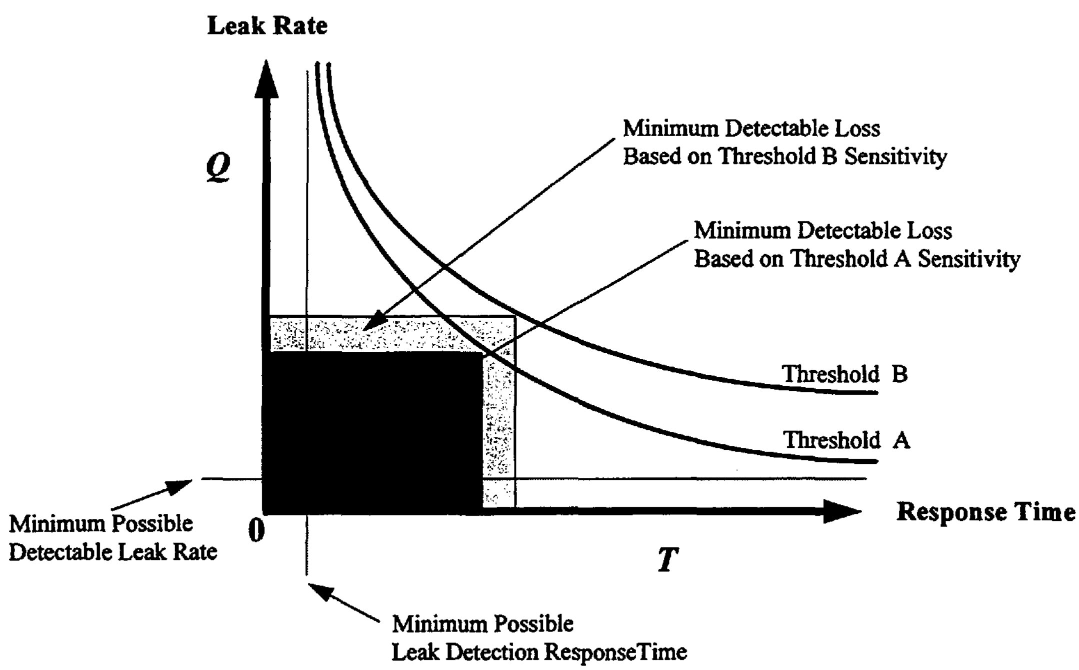

Figure 2: Examples of sensitivity curves based on different operating thresholds. These examples are typical of systems that operate on accumulated parameter errors (e.g., volume balance).

Frequently during the specification process, users attempt to define leak detection performance in terms of detecting a particular leak flow rate within a specified minimum period of time. Although sensitivity expressed in such terms certainly represents one aspect of performance, its importance can vary depending on the nature of the leak detection system and the operating characteristics of the target pipeline system. As shown in Figures 2 and 3, the correlation between leak size and response time can be highly dependent upon the leak detection techniques employed. It is also important to recognize that adjustments made in the interest of improving sensitivity can have a corresponding and not necessarily beneficial effect on other aspects of performance.

The examples shown in Figures 2 and 3 also serve to illustrate the concept of *minimum detectable leak size* and *minimum attainable response time* on any given pipeline. In practice, most systems can be set up to achieve various levels of sensitivity, provided the minimum detectable leak size and minimum attainable response time are not violated. The leak detection system vendor, and possibly the pipeline operator, can affect these characteristics by adjusting leak detection thresholds, filter characteristics or other parameters. Appropriate settings for these thresholds are usually dependent upon factors such as the SCADA system's scan time, instrument placement, fluid types, and so forth.

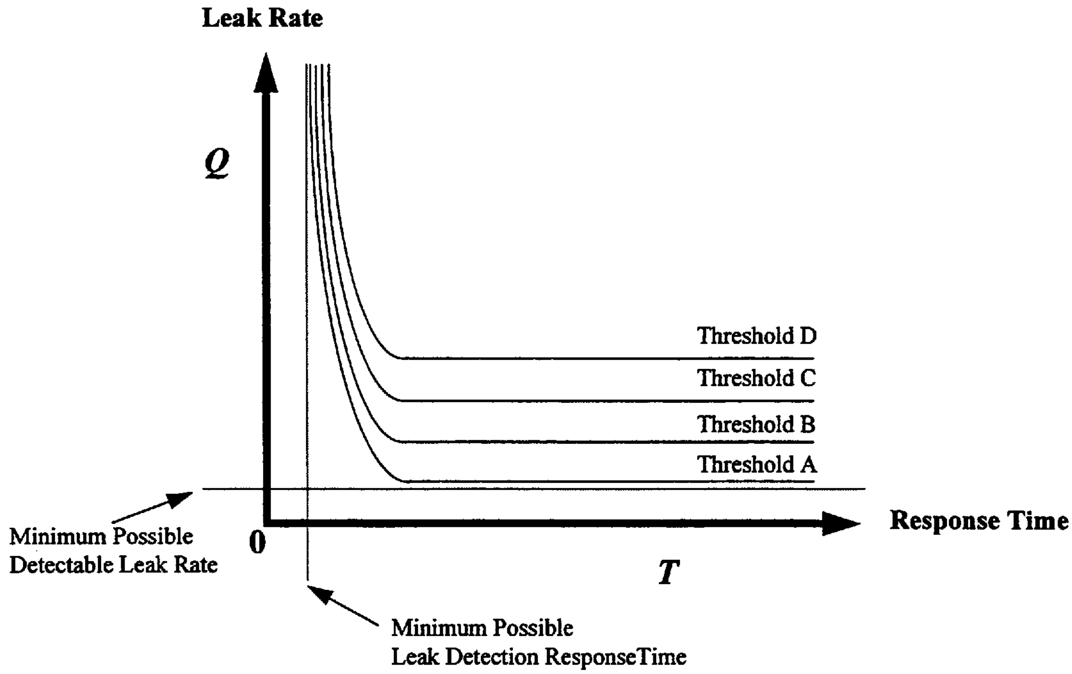

Figure 3: Examples of sensitivity curves typical of event oriented systems. Such systems might employ pattern recognition techniques to identify the onset of a leak.

Accuracy

To this point we have focused on the philosophy of detecting and announcing a leak, but have not considered the additional information that might accompany a leak alarm. With reference to Figure 1, this additional information is derived by the leak parameter estimation process, and is made available to the user as ancillary data output from the software based leak detection system. Although the amount and nature of such information varies between vendors, it typically includes estimates of leak parameters such as *leak flow rate, total volume lost, type of fluid lost*, and *leak location* within the pipeline network. The validity of these leak parameter estimates constitutes a third measure of performance referred to as *accuracy*.

From a strictly mechanical point of view, leak rate depends upon the magnitude and shape of the perforation, pipe environment, fluid characteristics and pressure at the leak site. If the location of a leak is known, the leak flow rate can be used to determine resultant disturbances in pressure, flow rate, and temperature at other points on the pipeline. Software based leak detection systems, on the other hand, deal with quite the opposite situation. Although these systems approach their task in a wide variety of ways, the one thing they all have in common is that they must operate with no prior knowledge of the size or location of a leak, should one occur. Thus, a particular system might calculate a leak flow rate to compensate for a difference between observed and expected values of pressure or flow at certain points on the pipeline. This effective leak flow rate might then be used to estimate the location of the leak and/or the volume loss related to the leak. Another system, operating on the same pipeline, might estimate total fluid volume lost on the basis of metered volumes and calculated changes in line pack, without ever attempting to directly estimate leak flow rate or location.

Robustness

Robustness is defined herein as a measure of the leak detection system's ability to continue to function and provide useful information, even under changing conditions of pipeline operation, or in conditions where data is lost or suspect. A system is considered to be robust if it continues to function under such less than ideal conditions. On the other hand, if the system disables certain functions, it might then achieve better reliability, but would be considered less robust.

The distinction between reliability and robustness is significant. Reliability is a measure of performance within a specified operational envelope. Robustness is a measure of the effective size of the operational envelope. For example, consider the following hypothetical leak detection systems:

System I: This system employs a sensitive leak detection algorithm. The system is normally very reliable, but will frequently generate alarms during certain normal pipeline operations.

System II: This system employs an alternative algorithm which is somewhat less sensitive than that of System I, but generates only a fraction of the alarms.

System III: This system employs the same sensitive leak detection algorithm as System I, but inhibits leak detection during pipeline operations that can cause it to generate alarms.

System IV: This system normally employs the same sensitive leak detection algorithm as System I, but switches to the less sensitive algorithm of System II when it senses conditions that generate alarms.

In this example, the designers of System I have sacrificed a degree of reliability in order to maintain a high level of sensitivity, whereas the designers of System II have chosen to sacrifice a degree of sensitivity in order to achieve a high level of reliability. By simply disabling the leak detection function under certain conditions, the designers of System III have sacrificed a degree of robustness in order to achieve higher levels of reliability and sensitivity. The example of System IV represents an attempt to selectively trade sensitivity and/or reliability in order to achieve a more robust system.

Although techniques vary between different software based leak detection methodologies, most attempt to achieve an acceptable tradeoff between reliability, sensitivity, accuracy, and robustness by sensing conditions of pipeline operation that cause alarms and making temporary parameter adjustments or disabling certain functions as required. Prior to the selection of a methodology for a given pipeline system, it is important that the pipeline company understand the way all operating conditions are handled by that methodology. This understanding will help the pipeline company to determine if a particular solution is consistent with the target pipeline's operational characteristics, as well as the company's expectations.

The reliability of a pipeline's communication, SCADA, and instrumentation systems can also have a dramatic effect on the utility of a software based leak detection system. A more robust system is one that is less likely to exhibit loss of functionality during periods of partial data outages caused by instrument failures, communication anomalies, routine maintenance, and so forth. Systems that continue to operate during outage periods or transient conditions on the pipeline might employ different settings for thresholds, filter characteristics, and other parameters. This usually results in some degradation of the system's sensitivity, accuracy, and/or reliability. In such cases, robustness is enhanced at the expense of other aspects of performance.

Other Factors Affecting Leak Detection Performance

Aside from operational considerations and the physical topology of a pipeline, there are other factors that affect leak detection performance. Although a composite of these factors is represented within the various performance classes, some warrant independent discussion. The following paragraphs present a brief discussion of such factors, and their relationship to leak detection performance. Background information related to these items is expected to be transferred to the vendor via either the configuration file describing the pipeline, or the data file containing representative runtime information upon which to base a study.

Instruments located along the pipeline provide the fundamental information used by a software based leak detection system to monitor and analyze the operation of the physical pipeline system. Therefore, for optimum performance on a given pipeline, it is important that the characteristics and physical locations of the existing pipeline instrumentation be consistent with the needs of the system(s) under consideration. The performance of an instrument can be characterized in terms of its *accuracy*, *repeatability*, and measurement *precision*. Instrument characteristics are normally specified by the instrument manufacturer, and will vary based upon the quality of the instrument and how it is maintained.

Instrument accuracy denotes the measurement performance of the instrument relative to that of an ideal device. It may be expressed absolutely as the worst case measurement error over the range of the instrument, or it may be expressed relatively as a fraction of the instrument's full scale reading. The repeatability of an instrument is a measure of its ability to consistently return the same reading for a given set of measurement conditions. Repeatability may also be expressed absolutely as a worst case discrepancy, or relatively as a fraction of the full scale reading. Instrument precision, or resolution, is a measure of the smallest change that can be reflected in the output of the instrument. Precision depends upon the resolution of the analog-to-digital converter used to acquire readings, and/or the transducer itself.

Software based leak detection methodologies are sensitive to instrument characteristics and placement. Instrumentation requirements should be reviewed with each leak detection system vendor under consideration. While it is generally true that instruments of great accuracy, precision, and repeatability will improve performance over instruments of lesser characteristics, it is also true that some leak detection techniques are much less dependent upon instrument characteristics than others. Furthermore, estimates of an instrument's actual field operation, along with descriptions of the pipeline company's calibration guidelines and SCADA data processing techniques, will be more useful to a vendor of software based leak detection systems than the instrument manufacturer's published specifications.

Consistent and reliable SCADA system performance is of critical importance to a software based leak detection system, regardless of the methodology employed. If the quality of the data is bad, or if the data acquisition frequency is inadequate, the ability of the software to recognize a potential or actual leak condition is compromised. In addition to the physical description of the pipeline system, definition of the pipeline company's SCADA system, and its performance characteristics, are of critical importance to the leak detection vendor. This definition provides the vendor with background information necessary to determine if an existing SCADA system will be adequate to support the needs of their software. SCADA performance characteristics that can have a negative affect on leak detection include slow or irregular update rates, time skew in acquired data from opposite ends of the pipeline, and communication system reliability. These, like many of the other factors, have different effects depending on the leak detection method under consideration, and therefore, must be discussed with each vendor to determine their impact on that method's functionality.

Specification and Prioritization of Performance Metrics

Within the framework of the proposed leak detection system evaluation methodology, each performance metric is evaluated in terms of a system's ability to satisfy a set of related criteria. Vendors might assist in the development of performance criteria that are relevant to their particular leak detection systems, but ultimately it is the pipeline company that must establish specific criteria for a particular pipeline. In so doing, the company must first define their leak detection goals for the pipeline and then specify corresponding criteria relative to the performance metrics of reliability, sensitivity, accuracy and robustness. These performance criteria constitute one set of information that the company would then provide to a potential vendor in order to determine if that vendor's system is an acceptable leak detection solution.

There are three steps involved in determining the appropriate leak detection performance criteria for a particular pipeline. The pipeline company must first identify any *legal, contractual or regulatory requirements* relating to leak detection. A minimum set of performance criteria must be established to meet these obligations.

The next step is to *characterize the pipeline* in terms of its possible leak mechanisms and the likelihood that one of these will result in a leak. A number of diverse factors are involved in this characterization. These include, but are not necessarily limited to:

- Length and volume of the pipeline,
- Pressure, temperature, and flow rate envelope,

- Terrain over which the pipeline travels,
- Type of fluids transported,
- The installed pipe,
- Pipeline operating procedures,
- Pipeline maintenance procedures, and
- External factors such as nearby roadwork, construction, or land development activity.

The final step in developing performance criteria is to perform an assessment of definite and potential costs associated with incorrectly declared leak alarms, missed alarms, late alarms, and any other deviation from ideal leak detection system performance. This assessment, when considered alongside the regulatory requirements and the leak potential characterization of the pipeline, should provide a basis from which the pipeline company can establish a set of leak detection objectives. The task of defining the appropriate leak detection performance criteria can then be reduced to a process of prioritizing each performance metric in terms of its level of importance, and further defining a set of specific performance criteria which illustrate the desired objectives.

The format for presenting performance metrics and the related specific performance criteria to software based leak detection vendors is divided into two tables as presented in Figure 4. In the first table, each performance metric is ranked based on its level of importance to the pipeline company. Ranking of the four (4) performance metrics simply involves assignment of a numerical rank (1, 2, 3, or 4) to each, with the most important performance metric being assigned a rank of one (1).

The second table contains definitions of specific performance criteria related to each performance metric, and may be optionally left blank or deferred to the vendor to complete. In this table, each performance metric may be characterized by a set of performance criteria to be evaluated under certain operating conditions on the pipeline. These criterion may be specified in either qualitative or quantitative terms. Pipeline companies are encouraged to provide qualitative specifications for performance criteria, and quantitative specifications where possible. Even though many of the performance criteria are difficult, or even impossible, to completely separate from others, this mechanism provides the pipeline company with a means to identify and rank the specific elements of performance important to them, and relevant to their operational needs and leak detection goals.

It must be noted that the performance criteria identified in Figure 4 are specified in qualitative terms rather than quantitative terms, and are only a representative sample of criteria that might be established under a given set circumstances. This is not an all-inclusive list that would apply to every pipeline, nor is it a recommended list with application to any particular pipeline. Since the needs of each pipeline company differ, it is only necessary to specify those performance criteria that are representative of the pipeline's specific needs.

Performance Metric	**Level of Importance (Rank 1-4)**
Sensitivity	
Reliability	
Robustness	
Accuracy	

Performance Metric	**Qualitative Performance Criteria Specification**
Sensitivity	Minimum detectable leak rate
	Minimum detectable leak volume
	Maximum volume loss prior to alarm
	Response time for a large leak
	Response time for a small leak
Reliability	Incorrect leak alarm declaration rate (overall)
	Incorrect leak alarm declaration rate (steady state flow)
	Incorrect leak alarm declaration rate (transient conditions)
	Incorrect leak alarm declaration rate (static conditions)
Robustness	Loss of function due to pressure outage(s)
	Loss of function due to temperature outage(s)
	Loss of function due to flow measurement outage(s)
	Loss of function due to pump state changes
	Loss of function due to valve state changes
	Loss of sensitivity due to pump state changes
	Loss of sensitivity due to valve state changes
	Startup stabilization period
Accuracy	Leak location error
	Leak flow rate error
	Leak volume error

Figure 4: Tabular format for the ranking of the level of importance for each performance metric, and an optional table for qualitative or quantitative specification of performance criteria related to each metric.

Features and Functions

As an integral part of many software based leak detection systems, there exist numerous features and functions that may not necessarily affect leak detection performance, but add value to the presentation and use of information produced by the leak detection system. Pipeline companies frequently require that certain features and functions accompany a software based leak detection system. As part of the specification process, the pipeline company might also desire to indicate which features and functions are of specific interest to them. An applicable tabular format for this purpose, as shown in Figure 5, consists of two (2) columns to specify individual features and/or functions and to indicate the importance of each to the pipeline company.

Feature or Function Specification	Provide Additional Information (Yes/No)
Volume Balance/Imbalance Processing	
Line Pack Calculations	
Pressure Profiling	
Thermal Modeling	
Instrument Error Analysis	
Pump Modeling	
Batch and Product Tracking Capability	
Batch Interface Profiling	
Custody Transfer Tracking and Management	
Pig and Scraper Tracking	
Support for Drag Reduction Agents	
Support for Slack Line Flow	
Data Trending and Archiving	
Warm Start Capability	
Instrument Maintenance Scheduling	

Figure 5: Specification table for software based leak detection system features and functions.

A pipeline company's need for information about a particular feature or function should be indicated by one of two possible choices as follows:

- The company would like additional information about a particular feature/function, or
- The company is not interested.

The features and functions listed in Figure 5 are presented solely for illustrative purposes. This list is not all-inclusive, nor does it constitute a recommendation of any sort. Depending upon the nature of the leak detection system, certain of these features are routinely provided whereas others might not be available under any circumstance. It is important that the pipeline company exercise care in preparing such a list, since the inclusion of unnecessary features might impact both the cost and the utility of the system.

Chapter Four

Pipeline Configuration Data Requirements

To perform a meaningful analysis of the performance of a particular leak detection system on a given pipeline, a leak detection system vendor must have adequate knowledge about the physical configuration of the pipeline. The vendor will use this knowledge of the pipeline to develop their response to the pipeline company's request for estimated performance projections. Since different leak detection methodologies require varying amounts of supporting information and detail, the evaluation procedure has been organized to allow for variations in the magnitude of information produced, provided that the pipeline company has a predetermined software based leak detection method in mind. In any case, it is the responsibility of the pipeline engineer to describe the physical pipeline system to whatever degree of detail is required by the desired methodology. In cases where the pipeline company has no predetermined methods in mind, a complete description of the pipeline must be provided so that any available methodology can be considered.

In this document a standard, keyword record oriented file format is defined to facilitate the transmission of pipeline physical configuration information to leak detection system vendor companies. The configuration file structure defined herein is designed to permit a complete and accurate description of any pipeline and all elements of the pipeline that are significant to a leak detection system. It is intended to provide an efficient, universally accepted standard for transmission of pipeline configuration data. A properly constructed configuration file can be used directly by a leak detection system vendor to develop a model of the pipeline suitable for off-line testing with captured (or simulated) operational data provided by the pipeline company.

A generalized discussion of pipeline network topology and the corresponding configuration file structure is provided in this section. Complete definitions of all configuration file record types are provided in Appendix B.

Elements of a Network of Pipelines

For the purposes of this document, a *network* of pipelines is defined as an array of pumps, valves and pipe all under the ultimate control of a single SCADA system master station. Within a given network there may be one or more distinct *pipelines*, each of which is normally bounded by flow measurement instrumentation and has the capability of operating independently of all the other pipelines. Within each pipeline there may be one or more *sections*, each of which is usually associated with an independent survey or measure of linear position along the primary pipeline right-of-way. Finally, within each section there may be several pipe segments and stations. An example of this structure and associated nomenclature is depicted in Figure 6.

Each pipeline is represented as an array of *nodes* and *volume elements*. Nodes are defined on a pipeline at all points of significance to the leak detection system. In general, this includes the endpoints of all volume elements, all points where measurement instruments used by the leak detection system are located, and all points required to adequately profile the elevation of the pipeline. Volume elements include all elements of the pipeline conduit itself.

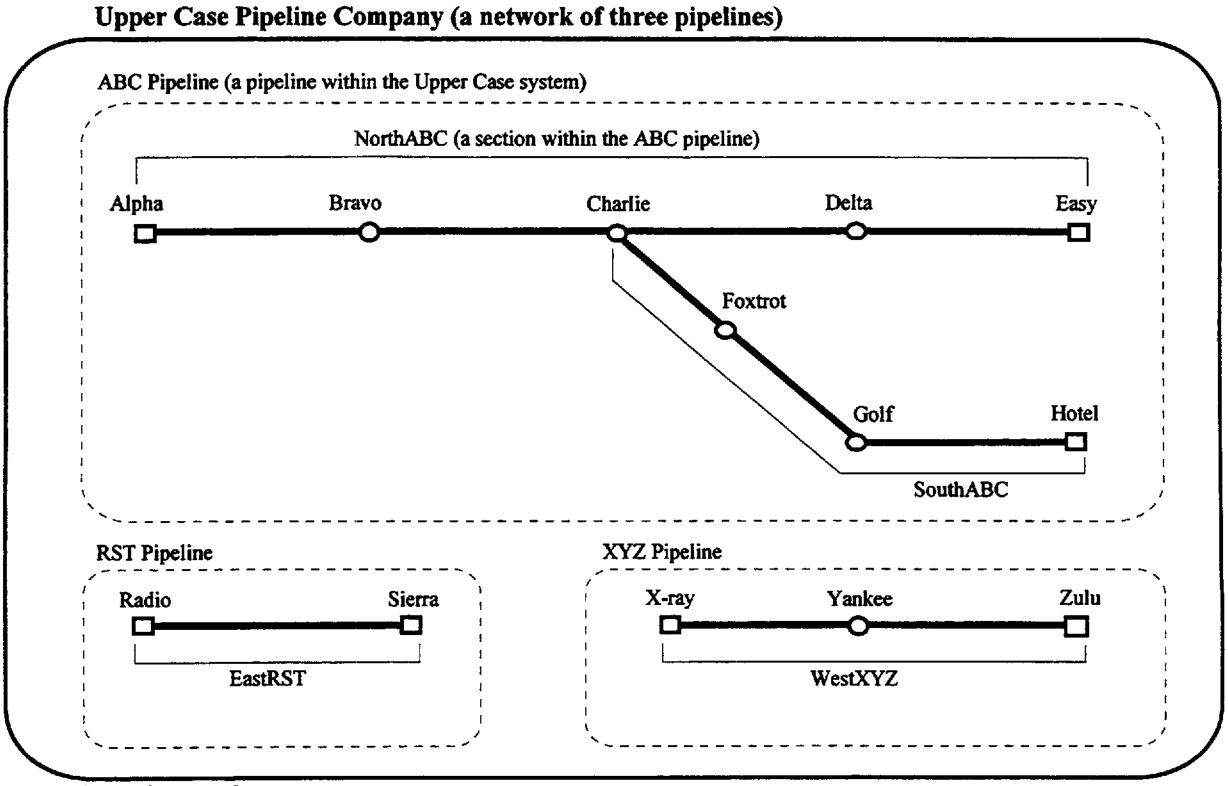

Figure 6: Pipeline topology representation.

The most obvious *dual-ported volume elements* are pumps, volumetric flow meters, and pipe segments. Less common devices that must be treated as volume elements include strainers, separators, pulsation dampeners and so forth. Valves, even though they displace a negligibly small amount of fluid, must also be treated as volume elements because they have the capability of controlling flow between two points. Valves are treated as an adjacent pair of nodes (i.e. a dual-ported element of zero volume) for the purposes of this document.

A pipe *segment* is defined to be a contiguous length of conduit containing no other volume elements, and across which there is no significant change in pipe inside diameter, pipe outside diameter, pipe wall thickness, pipe wall roughness, burial status of the pipe, pipe coating or pipe insulation. Fluid may be delivered from the pipeline and/or received into the pipeline at the endpoints of a segment but not at any point within a segment. The pipeline configuration file structure described in this document includes keyword record types which permit the user to specify all the various types of dual-ported volume elements.

Pump stations and other complex installations on a pipeline can always be represented as a combination of dual-ported volume elements. However, there are many cases in which the leak detection vendor will not require this level of detail, provided that the station is bounded by measurements that can be used to drive the leak detection system. In such cases, simple booster stations can be represented as dual-ported volume elements. More complex junctions and stations with multiple inlets or outlets constitute *multi-ported volume elements*. The pipeline configuration file structure described in this document does not provide a keyword record type for specification of a single volume element with an arbitrary number of ports. Even the most complex such cases can always be represented as a combination of dual-ported elements.

Boundary nodes represent the points of intersection between all significant volume elements within the pipeline. Specifically, boundary nodes exist at the following locations:

- Any point where fluid can be delivered from the pipeline and/or received into the pipeline,
- Any point where there is a significant change in pipe inside diameter, pipe outside diameter or pipe wall thickness,
- Any point where there is a significant change in the coating, insulation or burial status of the pipe (e.g. where the pipe enters or exits a body of water), and
- The inlet or outlet of a pump unit, volumetric flow meter, pipe segment or any other volume element.

The pipeline configuration file structure described in this document employs keyword record types which permit the user to specify the various types of volume elements. Boundary nodes are implicitly defined as a part of the volume element specifications. Thus there is no requirement for a keyword record type to specify boundary nodes.

Control nodes include blocking valve, control valve, regulator valve, and check valve sites on a pipeline. All active valves must be specified in the pipeline configuration file. Some vendors also require the specification of inactive valves, since they employ formulas to model the slight pressure drops associated with these devices. The file structure described herein includes keyword record types to specify valve sites as control nodes.

Measurement nodes include all points on a pipeline where instrumentation is attached to the pipeline in order to make measurements that are used by the leak detection system. All measurement nodes must be fully specified in the pipeline configuration file. The file structure described in this document includes keyword record types that can be used to define the various measurement nodes required for leak detection.

Elevation nodes include all points that must be defined in order to adequately represent the elevation profile for leak detection purposes. The number of points required in the elevation profile is dependent upon the nature of the pipeline and the type of leak detection system that might be employed. The file structure defined herein provides a keyword record type that can be used to specify the elevation profile, regardless of the nature of the pipeline or leak detection system.

Elements of the Pipeline Configuration File

The configuration file is a record-oriented ASCII text file. It is composed of a set of records describing the network topology, pipeline geometry, fluid properties, and all pertinent pipeline instrumentation for leak detection purposes. The first field of each record in the configuration file is a keyword. The keyword identifies the record type, which establishes the meaning of the parameters that are provided in the remaining fields of the record. Specific keyword record types are defined for the following purposes:

- To describe the organization of distinct pipelines within a network,

- To describe the organization of sections within a pipeline,
- To describe all the physical elements of a pipeline (e.g. pumps, valves, meters, and so forth) that are pertinent to leak detection,
- To define the elevation profile of a pipeline,
- To assign engineering units or scale factors, and
- To assign device state decode definitions.

By properly employing the various record types it is possible for the user to build a file that defines the structure of a pipeline or network of pipelines, describes the physical properties of the pipeline, and completely specifies the characteristics of every physical element of the pipeline that may be pertinent to the evaluation of leak detection system performance. A list of available keywords follows:

ambient Identifies a record describing the position and characteristics of a real-time ambient air, soil or water temperature measurement.

assign Identifies a record used to assign engineering units to data entries in the pipeline SCADA data base and other parameters.

blkValve Identifies a record describing a three-state blocking valve.

chkValve Identifies a record describing a two-state check valve.

ctrlValve Identifies a record describing a control valve.

dataQuality Identifies a record used to assign data quality attributes.

decode Identifies a record used to define device states corresponding to particular values in the SCADA database.

default Identifies a record used to define default values for certain parameters.

device Identifies a record used to define certain dual-ported volume elements.

dNode Identifies a record describing a real-time fluid density measurement.

dTable Identifies a record used to define the relation between density, pressure, and temperature for a specific fluid.

endNetwork Identifies a record used to terminate the definition of a *network ... endNetwork* block within the configuration file.

endPipeline Identifies a record used to terminate the definition of a *pipeline ... endPipeline* block within the configuration file.

endSection Identifies a record used to terminate the definition of a *section ... endsection* block within the configuration file.

eTable Identifies a record defining the location and elevation of a point or set of points on the pipeline.

fluid Identifies a record describing the properties of a specific fluid.

fNode Identifies a record used to identify the type of fluid at a point in the pipeline.

network Identifies a record used to initiate the definition of a *network ... endNetwork* block within the configuration file.

pipeline Identifies a record used to initiate the definition of a *pipeline ... endPipeline* block within the configuration file and to assign the values of standard pressure and temperature for the corresponding pipeline.

pNode Identifies a record describing a real-time fluid pressure measurement.

port Identifies a record describing a port or connection between two sections within a pipeline.

qNode Identifies a record describing a real-time flowrate and/or volumetric flow measurement.

regValve Identifies a record describing a pressure regulator valve.

section Identifies a record used to initiate the start of a *section ... endSection* block within the configuration file.

segment Identifies a record used to define the location and physical parameters of a distinct segment within a pipeline.

station Identifies a record used to define a simple station volume.

tNode Identifies a record describing a real-time fluid temperature measurement.

vNode Identifies a record describing a real-time fluid viscosity measurement.

vpTable Identifies a record used to define the relation between vapor pressure and temperature for a specific fluid.

vTable Identifies a record used to define the relation between viscosity, pressure, and temperature for a specific fluid.

A definition of the record structure associated with each of the keywords is provided in Appendix B. A complete definition of each record field is provided for all keyword record types along with an example of how each record type might appear in a typical pipeline configuration definition.

Structure of the Pipeline Configuration File

The pipeline configuration file is organized into *blocks*. The *network, pipeline, section, endNetwork, endPipeline,* and *endSection* record types are used to establish blocks within a configuration file. The three block types also serve to establish three distinct *levels* within a configuration file. Thus the configuration file blocks are directly analogous to the actual elements of the typical pipeline structure depicted in Figure 6.

Of the three block types, the *network ... endNetwork* block occurs at the highest level. All records that serve to describe the configuration of a particular pipeline network must be included within the corresponding *network ... endNetwork* block. As previously noted, a network of pipelines is defined as an array of pumps, valves and pipe all under the ultimate control of a single SCADA system master station. Although unlikely, it is technically possible to define more than one such network within a single configuration file.

The *pipeline ... endPipeline* blocks occur at the second level and must each be embedded within a *network ... endNetwork* block. All records that serve to describe the configuration of a particular pipeline must be included within the corresponding *pipeline ... endPipeline* block. One or more pipelines may be defined within a given network. Each distinct pipeline is normally bounded by flow measurement instrumentation and has the capability of operating independently of all the other pipelines.

The *section ... endSection* blocks occur at the third or lowest level and must each be contained within a *pipeline ... endPipeline* block. All records that describe elements within a particular section of a pipeline must reside within the corresponding *section ... endSection* block. One or more sections may be defined within each pipeline. Each section is usually associated with an independent survey or measure of linear position along the primary pipeline right-of-way. Aside from short stubs and other minor diversions, it is assumed that the survey measurement of position varies monotonically across the section and that widely separated points within a section cannot be assigned the same survey measurement value. In other words, if two points within the same section are physically separated by five miles they cannot be assigned the same survey mile post number.

In building a configuration file, it is imperative that the proper structure be maintained. Each block must be opened with a *network, pipeline,* or *section* record and terminated with an *endNetwork, endPipeline,* or *endSection* record as required. All *pipeline ... endPipeline* blocks must be embedded within a *network ... endNetwork* block. Furthermore, all *section ... endSection* blocks must be embedded within *pipeline ... endPipeline* blocks. Failure to maintain proper structure will invalidate the configuration file.

Scope of Configuration File Record Types

As currently structured, there are three levels corresponding to the three block types that can occur in a configuration file. The scope of each record type (or keyword) includes one or more of these levels as follows:

network
assign
dataQuality
decode
default
fluid
dTable
vpTable
vTable
pipeline
assign
default
section
ambient
assign
blkValve
chkValve
ctrlValve
default
device
dNode
eTable
fNode
pNode
port
qNode
regValve
segment
station
tNode
vNode
endSection
endPipeline
endNetwork

The definitions provided in Appendix B indicate the scope of each record type, and it is required that the appropriate configuration file structure be maintained. In a valid configuration file, all record types must occur within their valid scope (i.e. at an appropriate level). For example, all *pNode* records must occur within a *section ... endSection* block. Record types that occur outside their valid scope will invalidate the configuration file.

Establishing Connectivity between Volume Elements

It is the responsibility of the pipeline engineer to insure that the configuration file data properly establishes connectivity between the various volume elements on the pipeline. To this end, each record type used to define a node on the pipeline includes a *refPost* field for the purpose of assigning an identifier to that node. Furthermore, each record type used to define a volume element on the pipeline includes *upsPost* and *dnsPost* fields for the purpose of assigning identifiers to the nominal upstream and downstream nodes bounding the element. Connections between volume elements are indicated by the assignment of identical node identifiers to two or more elements.

In general, the assignment of node identifiers is arbitrary and any consistent assignment scheme is acceptable, provided that all node identifiers are unique. In practice, it is recommended that the pipeline engineer utilize a scheme that will yield a degree of correspondence between node identifiers and pipeline survey measurements. For example, if three nodes are all located in close proximity to survey mile post 123, then the engineer might choose to assign the identifiers *123A, 123B,* and *123C.* Alternatively, the assignments *12300, 12301*, and *12302* might be made.

As a second example consider the situation in which a blocking valve is located at kilometer post 234 plus 56 meters. Even though the blocking valve is assumed to displace no fluid, it is a volume element and requires the specification of two nodes. In this case the engineer might choose to assign the identifiers *234056U* and *234056D*, respectively, to the nominal upstream and downstream ports of the valve. Now suppose the blocking valve is located in a span of pipe that runs from kilometer post 201 plus 25 meters to kilometer post 260 plus 50 meters. Assuming there are no other significant points within that span, and adhering to the same basic principle, the engineer would assign node identifiers *201025* and *234056U* to the upstream segment, thereby indicating a connection to the blocking valve inlet. By the same token, the engineer would assign node identifiers *234056D* and *260050* to the downstream segment, which would indicate a connection to the blocking valve outlet. With reference to the record type definitions of Appendix B, the corresponding configuration file fragment might appear as follows:

```
segment    SEG020,  201025,   234056U, 33.031, 25, 1.0, 30.0E-6, 0.005, 0, 0, soilA, 2.0, 0.03
blkValve   STA0001, 234056U,  234056D, BlockingValveRuleA
segment    SEG023,  234056D,  260050,  25.994, 25, 1.0, 30.0E-6, 0.005, 0, 0, soilA, 2.0, 0.03
```

The order of the records in this example is unimportant, although the sequential arrangement shown does contribute to ease of interpretation. It is the assignment of identical node identifiers that indicates the connections between the blocking valve and the two segments of pipe.

Instrument placement is usually critical to leak detection system operation. It is very important that the position of pressure, temperature, and flow measurement instrumentation is accurately specified. In the current example, the pipeline engineer could indicate the presence of a pressure transducer at the inlet to the blocking valve by adding a *pNode* record to the configuration file and assigning the *23056U* identifier to the *refPost* field of that record. In that case, the corresponding configuration file fragment could take the form:

```
segment    SEG020,  201025,   234056U, 33.031, 25, 1.0, 30.0E-6, 0.005, 0, 0, soilA, 2.0, 0.03
pNode      DTA0001, 234056U,  1.0, 0.0, 0.0, 1024, 4, 0.5, 4
blkValve   STA0001, 234056U,  234056D, BlockingValveRuleA
segment    SEG023,  234056D,  260050,  25.994, 25, 1.0, 30.0E-6, 0.005, 0, 0, soilA, 2.0, 0.03
```

An example of a configuration file representing a network of three small pipelines is included as Appendix C of this document. This example is based on the simple network shown in Figure 6. Reference should be made to this example in dealing with questions about how to specify connectivity between volume elements in a configuration file.

Chapter Five

Pipeline Operating Data Requirements

The purpose of the captured (or simulated) data file is to provide a leak detection vendor with a realistic snapshot of pipeline operation in the form of SCADA data which a leak detection system would use if actually installed on the pipeline. The captured data file is used by the vendor in conjunction with other material provided by the pipeline to predict performance of the vendor's leak detection system. Data files must contain information gathered over a sufficient period of time to allow a vendor's software to accurately interpret its content. It may also be useful to embed leak test scenarios within one or more of the data files to help qualify a particular solution for application on a specified pipeline system.

Time period requirements for data files may vary between vendors, and therefore should be reviewed with those vendors expected to participate in the test, prior to beginning the data capture process. Twenty-four (24) hours of continuous data for a given test is usually sufficient, however, complex operating environments, or those which involve a high degree of batch related activity, may require longer time periods. In general, more data (i.e., data collected over a longer period of time) is usually better, and will usually allow the vendor to produce better results.

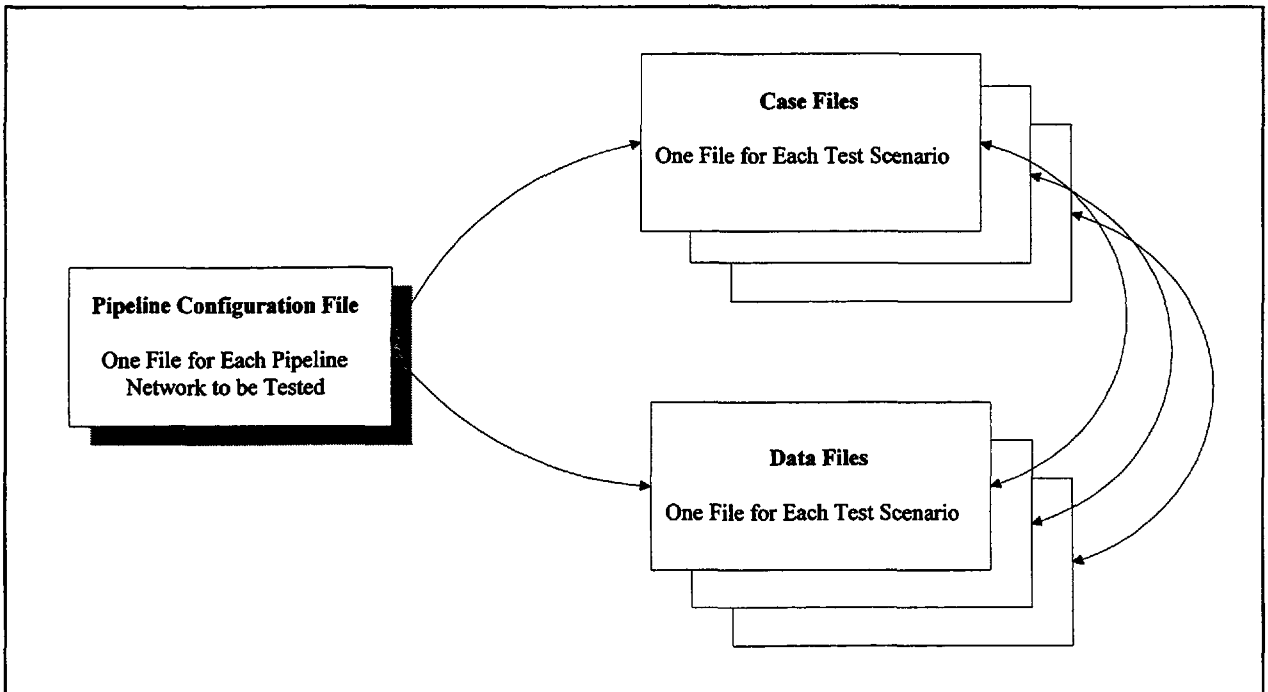

Figure 7: Relationship between the Configuration, Case, and Data file structures.

The pipeline company must provide two (2) separate data related files for each scenario to be tested. The first file is the *data* file itself, and is composed of blocks of captured (or simulated) data corresponding to the desired test scenario. The other file is a "read me" text file describing the pipeline, the format of data file, and the specifics of the test scenario. This file is referred to herein as the *case* file.

The relationship between the information contained within the configuration, data, and case files is graphically represented as shown in Figure 7. Because the combination of the information contained within these files provides the basis upon which the software vendor will perform the analysis, it is important that a one to one relationship exist between the data elements referred to in each file. Note that for each data element identifier used in the configuration file, there must be a corresponding identifier in the case file describing its format and its location in the data file.

Data File Structure

Data files are to consist of fixed format records carrying the captured data corresponding to a particular test scenario. Files may be block structured or consist of a series of sequential records which correspond to changes in measured values (i.e., a report by exception scheme). Data files must be produced in alphanumeric (text) format. Each record must correspond to a field data point or device state which resides in the pipeline's SCADA database, and which has been identified in the pipeline configuration file.

Block Structured Data Files

Block structured data files must be exactly as specified in the corresponding case file. Each block must include a complete set of records arranged in the same order, and all records must be repeated in each block, regardless of whether or not their data has been changed since the preceding block was collected. There are two options for block structured files:

1. Repeating data blocks with a time stamp appearing at the head of each block corresponding to all embedded data records contained within that block.

2. Repeating data blocks with a time stamp included with each embedded data record.

In the block structured data file each record must include the following two (2) fields:

- Numeric Data or Device Status

- Data Quality

If the data file is produced as described in Option 1, each data block must be preceded by a time stamp which is common for the entire data block, followed by entries for each data record. Data record entries must include the numeric data or device status field (i.e., the data value), followed by the data quality field for each data item, as illustrated in Figure 8.

If the data file is produced as described in Option 2, the time stamp for each data element must also be included with each record. In this case, in each record the numeric data or device status field must appear first, followed by the data quality field, and then the time stamp field. In either case, the file type, record structure, numeric data or device status format, data quality format, and time stamp format must correspond exactly with that specified in the associated case file.

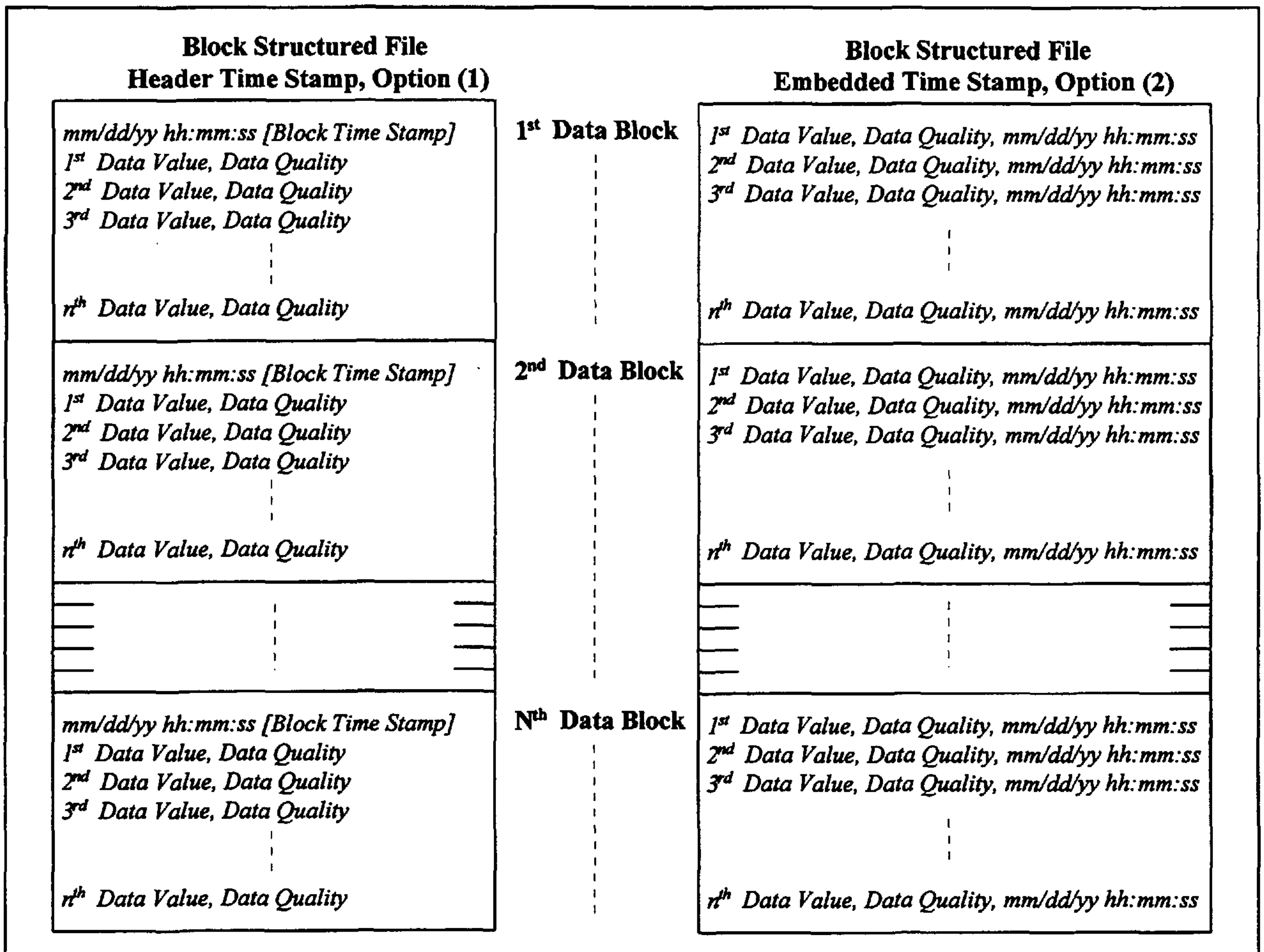

Figure 8: Block structured data file organization for header and embedded time stamp options.

Sequentially Organized Data Files

A sequentially organized data file is one that provides new data values in chronological order, as changes occur. This type of structure is analogous to a report by exception data acquisition scheme and is implemented in much the same manner. Unlike a block structured data file, the sequentially organized data file provides no options regarding the placement of time stamp information, as this is a fundamental and necessary part of the reporting scheme.

Each record in a sequentially organized data file must contain the following four (4) fields:

- Time Stamp,
- Data Element Identification,
- Numeric Data Value or Device Status, and
- Data Quality

The structure of the data file must consist of an entry (or record) for each data item referenced in the configuration file, every time a change in its value or state is recognized, as illustrated in Figure 9. So that the vendor can establish a starting point for analysis, it is important that the first entries in the data file provide initial values for all data items. This can be achieved by simply acquiring each data item's current value from the SCADA database, and recording that value in the target data file at the time the data capture operation is started.

Sequentially Organized Data File Structure

mm/dd/yy hh:mm:ss(0), Data Element ID(1), Data Value, Data Quality *mm/dd/yy hh:mm:ss(0), Data Element ID(2), Data Value, Data Quality* *mm/dd/yy hh:mm:ss(0), Data Element ID(3), Data Value, Data Quality* ⋮ *mm/dd/yy hh:mm:ss(0), Data Element ID(n), Data Value, Data Quality*	**Snap Shot of All Data Elements to Establish Initial Conditions**
mm/dd/yy hh:mm:ss(1), Data Element ID(n), Data Value, Data Quality *mm/dd/yy hh:mm:ss(2), Data Element ID(n), Data Value, Data Quality* *mm/dd/yy hh:mm:ss(3), Data Element ID(n), Data Value, Data Quality* *mm/dd/yy hh:mm:ss(4), Data Element ID(n), Data Value, Data Quality* *mm/dd/yy hh:mm:ss(5), Data Element ID(n), Data Value, Data Quality* ⋮ *mm/dd/yy hh:mm:ss(n), Data Element ID(n), Data Value, Data Quality*	**Record Entries for Each Data Element as Changes in Value or State Occur**

Figure 9: File structure for sequentially organized data files.

Constraints For Data File Organization

Data files to be presented to vendors for analysis must conform to the following constraints:

- All records must occupy a single line of text, terminated by carriage return (CR) and line feed (LF) control characters. Each item within a record must be delimited by a comma (,) and all comments must be preceded by a semi-colon (;).

- Numeric data must be expressed in integer, fixed point decimal, or scientific floating point decimal format. The numeric data format may vary between records and must be specified for each record in the data file as part of the structure definition provided in the case file.

- Device status must be represented as an ASCII hex number. The device status format may vary between records but must be specified for each record in a data file as part of the structure definition provided in the case file.

- Data quality must be represented as an integer number. A global specification for data quality is to be provided in the pipeline configuration file. The data quality definition may not vary between records within a data file.

- Time stamps must be represented in the format *mm/dd/yy hh:mm:ss*. Here *yy* is the ASCII decimal year number, *mm* is the ASCII decimal month number, *dd* is the ASCII decimal day of the month, *hh* is the ASCII decimal hour, *mm* is the ASCII decimal minute, and *ss* is the ASCII decimal second.

Case File Structure and Content

The case file is an ASCII text file providing general information about the pipeline, a detailed specification of the data file format, and a discussion of the test scenario represented in the data file. There must be one case file supplied for each data file. The case file is made up of a combination of keyword and comment elements, and must provide the following information:

- Brief Description of the Test Scenario
- Operating Company Name
- Pipeline Name
- Responsible Engineer's Name
- Responsible Engineer's Telephone Number
- Responsible Engineer's Fax Number
- Responsible Engineer's e-mail Address (if available)
- Data Start Time
- Data Stop Time
- Definition of the initial line pack corresponding to this data set. This should consist of a definition of every batch contained in the target system including batch id, batch volume (current/total), product type, and receipt/delivery status.
- Data File Type (Blocked or Sequential)
- Average Data Refresh Rate or Total Number of Data Blocks
- Data File Name
- Data Element and Block Structure Definition

Most of the information required in the case file may be presented as ASCII text with a semi-colon character preceding all comment fields. Definition of each data element must adhere to the standards set forth in the following paragraphs.

Data Element and Block Structure Definition

The data element and block structure definition portion of the case file involves identification of each data element as it will occur in the data file, and associating it with a corresponding measurement or status point identified in the pipeline configuration file. This is accomplished by identifying each element that will appear in the data file according to the corresponding SCADA database point identifier used in the configuration file. It is of the utmost importance that a one-to-one correspondence exist between the elements identified in the case file, those referenced in the configuration file, and the data provided in the data file. Since this provides the link between data elements provided in the data file and those referenced in the configuration file, if there are missing or inconsistently defined elements the vendor will not be able to correctly correlate the information contained within these files. For block structured data files, it is also important that the order in which data elements are specified in the case file correspond exactly to the order in which they appear in the data file. An example of a case file is provided in Appendix C.

Within the case file, the definition of each pipeline data element is to be based on a keyword record oriented structure. Two distinct record types are required for this purpose; one for SCADA status points and another for SCADA measurement points. These two record types are defined as follows:

STATUS The **status** keyword identifies a record used to define a SCADA status point in the operational data.

Syntax **status** ***pointID, format***

Parameters ***pointID*** The alphanumeric status point name or identifier as referenced in the configuration file.

format The format of the data element as provided in the data file. Two possible entries exist for status points:

hex (N) ASCII hex number in N character field
integer (N) ASCII decimal integer in N character field

Example **status** **AS0091, hex (4)**

MEASUREMENT The **measurement** keyword identifies a record used to define a SCADA measurement point in the operational data.

Syntax **measurement** ***pointID, format***

Parameters	***pointID***	The alphanumeric measurement point name or identifier as referenced in the configuration file.
	format	The format of the data element as provided in the data file. Four possible entries exist for measurement points:

hex (N)	ASCII hex number in N character field
integer (N)	ASCII decimal integer in N character field
fixed (N, M)	ASCII decimal fixed point number with N decimal places in M character field
scientific (N, M)	ASCII decimal scientific ("E" format) floating point number with N significant characters in M character field

Example **measurement AD0092, hex (8)**

Chapter Six

Procedure for Evaluation of a Software-Based Leak Detection System

The preceding sections of this document have provided important background information related to the software-based leak detection system evaluation process. This information, along with the step-by-step procedures contained within this section form the evaluation process. As depicted in Figure 10, the process is composed of six (6) parts, each of which require several steps. Each part of the process is intended to focus on one specific aspect of the overall process. Throughout the process there are references to preceding sections of this document and their related appendices which provide details about each particular topic.

Part I: Gathering Information and Describing the Physical Pipeline

Accurate definition of the physical pipeline system is an essential component of any software based leak detection system project. Since different leak detection methodologies require varying amounts of supporting information and detail, this procedure has been organized to allow for variations in the magnitude of information produced in cases where specific methodologies have been predetermined by the pipeline company. It is the responsibility of the pipeline engineer to describe the physical pipeline system to whatever degree of detail is required by the desired methodology(s). In cases where the pipeline company has no predetermined methods in mind, a complete description of the pipeline, as described in Chapter 4 and Appendix B, must be provided so that any available methodology can be considered.

Step 1

Consult with the vendor(s) who are expected to receive evaluation information. Since there are variations in the amount and depth of information required by different vendors, consultation prior to its preparation will help to ensure that oversights are minimized.

Step 2

Locate all available design information related to the pipeline system or systems to be defined. The amount of information necessary will be dependent upon the methodologies under consideration, and might include any or all of the following:

- Piping and Instrumentation Drawings (P&IDs) for all stations and pipeline segments being considered.
- Elevation profiles for all pipeline segments under consideration.
- Operating constraints for all pipeline measurement instrumentation as defined by the original manufacturer or through routine maintenance procedures. At a minimum, this information should include each instrument's range, accuracy, and repeatability.
- Definitions of all SCADA tag identifiers that will be used to reference SCADA information within the configuration definition.

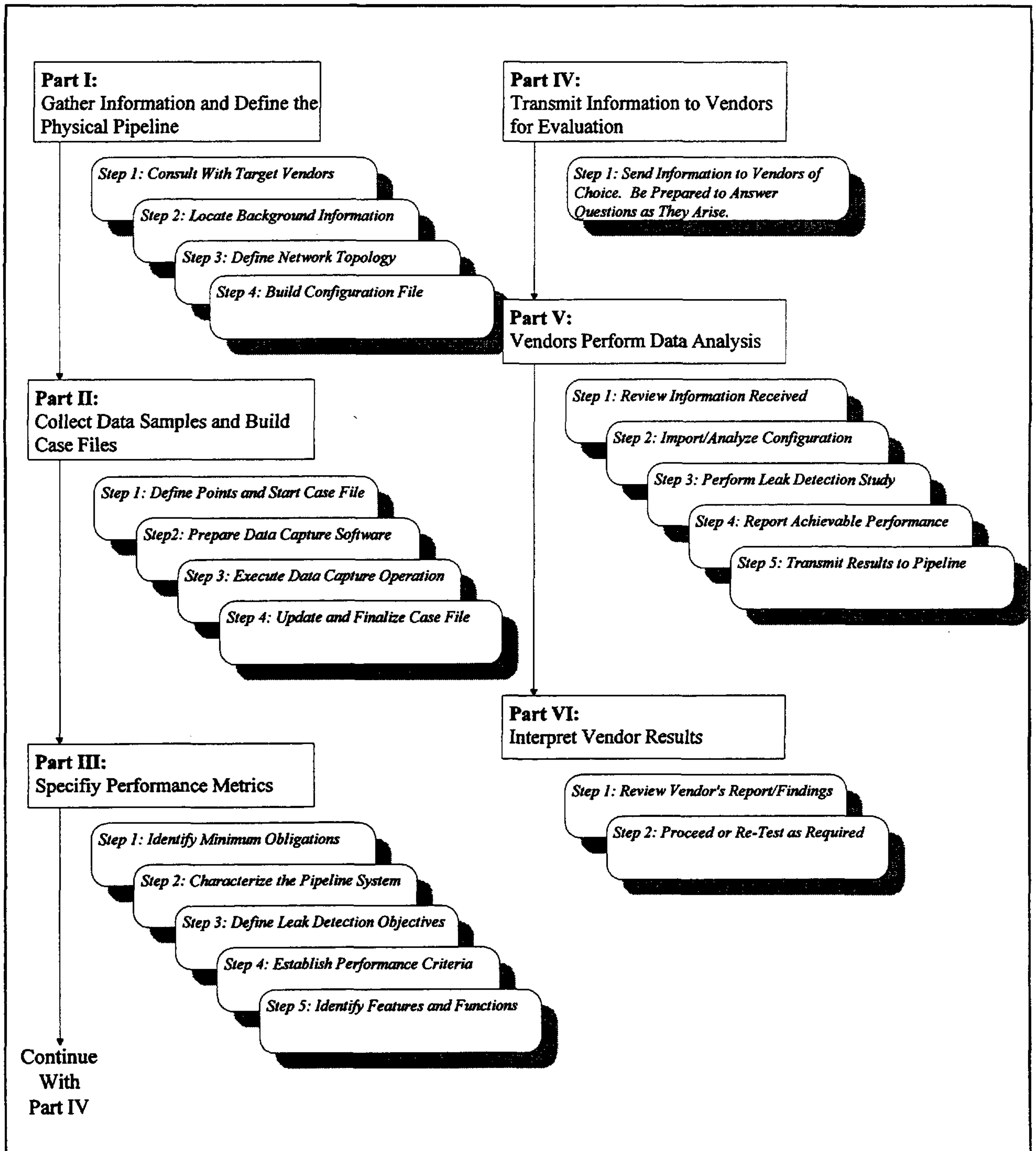

Figure 10: Overview of the six part procedure for the evaluation of a software based leak detection system.

<u>Step 3</u>

Review the organization of the pipeline(s) being considered, and define the overall pipeline network topology as discussed in Chapter 4 and Appendix B. This will involve the following:

- Identification of each pipeline capable of independent operation. Note that independent pipelines may be included as parts of a larger pipeline network.

- Identification of the *Nodes* and *Volume Elements* that make up each pipeline system.

- Node definitions should be further sub-divided into *Boundary Nodes, Measurement Nodes,* and *Elevation Nodes* as defined in the Chapter 4 and Appendix B.

- Identification of all fluid properties and measurement units associated with the pipeline's operation.

Step 4

Build the configuration file using the keyword structure defined in Chapter 4 and Appendix B. This is intended to be an ASCII text file *without* any word processor induced formatting and/or other imbedded controls. The layout of the configuration file should correspond to the network topology defined in Step 3.

Part II: Putting Together Case Files and Run-Time Data Samples

So that leak detection vendors can confirm their ability to detect leaks on the target pipeline system(s), it is necessary to supply sample data that is representative of typical operating conditions on the pipeline. Sample data must be provided in ASCII format and should be captured from the pipeline's SCADA system or from a simulator capable of modeling all of the pipeline's physical components. The actual data items captured must be consistent with those specified in the configuration definition developed in Part I of this procedure. The structure of the run-time data sample file is defined in Chapter 5.

Each set of sample data must consist of a *Case File* and a *Data File*. The case file is an ASCII text file that describes the content of a specified data file to the vendor. Among other items, the case file contains the definition of the sample data contained in the data file. Each record contained in the data file is presumed to be identical in format, and must be identified in the case file in accordance with the structure defined in Chapter 5 and Appendix C. It is important to note that the point identifiers (tags) specified in the case file must correspond to the identifiers used in the configuration file, as well as the data items provided in the data file.

Step 1

Begin development of the Case File according to the description contained in Chapter 5. This should include definition of all required case file information, with the exception of details regarding the captured data (i.e., time period, initial line pack, etc.).

Step 2

Prepare the SCADA system (or simulator) for the capture of sample data. For most companies this will require that a small program be written to extract real-time database values and their associated data quality and time stamp. The type and format of the file will be dependent upon decisions made be the pipeline regarding the data capture mechanism, and the capabilities of the SCADA system or simulator used as the data source.

As described in Chapter 5, the pipeline company must determine if the captured data file will be a block or sequentially structured file. Additionally, if the file will be block structured, a decision regarding the location of the time stamp must also be made. The time stamp may appear at either the beginning of each block, or embedded as a part of each data record contained within a block.

Step 3

Test the data capture program for a short period of time to ensure that it is functioning properly, and that all of the required data items are being gathered. Once satisfied, run the data capture program for the desired period(s). It is suggested that the minimum period of time for a single data file be at least twenty-four (24) hours in duration.

Step 4

Finalize the case file with any updated information resulting from the data capture process (i.e., the time period of the data file(s), special notes, initial line pack information, etc.).

Part III: Specification of Performance Metrics

As defined in Chapter 3, performance metrics are established by the pipeline company in order to evaluate a leak detection system's *sensitivity, reliability, robustness,* and *accuracy*. Each metric should be ranked by the pipeline company with respect to its relative importance. In addition, each metric might further be evaluated in terms of a system's ability to satisfy a set of related criteria. Vendors can assist in the development of performance criteria that are relevant to their particular leak detection systems, but ultimately, they must be defined by the pipeline company in order to reflect their specific leak detection objectives. There is no limit to the number of performance criteria that might be set forth to define a particular metric. However, there is no point in defining more criteria than are necessary to assure that all significant areas of interest to the pipeline company are addressed by the potential vendor. In situations where the pipeline company is unable to define its performance objectives in terms of specific criteria, ranking of the four performance metrics will provide a vendor with a broad representation of the pipeline's needs which will serve as the basis for vendor's development of specific performance criteria.

The relative costs or benefits that might be incurred as a consequence of varying levels of leak detection system performance directly influence the choice of system for installation on a particular pipeline. These performance elements of the cost/benefit appraisal are generally not independent of the physical and operational characteristics of the pipeline. Consequently, a company wishing to establish specific performance criteria must do so in such a way as to accurately and completely reflect the objectives of the operating company (economic and otherwise) for a particular target pipeline. In some cases, different criteria might be required for various pipelines within a system or for sections within a pipeline.

Aside from economic concerns, pipeline companies are sometimes subject to a number of other factors that can impact or constrain leak detection system performance requirements. These include legal obligations, regulatory requirements, and public relations concerns, among other things. It is of some interest to note that such factors often result in the imposition of specific performance requirements and/or constraints (e.g., "detection and initial response to any leak must occur within 15 minutes of the time that a 50 barrel fluid loss is incurred"). Such constraints are not always consistent with the statistical nature of the leak detection problem. In such cases, the pipeline company must use good judgment in defining relevant performance criteria.

It is clear, against a backdrop of conflicting concerns and constraints, that no simple formula can be stated to specify criteria for leak detection system performance evaluation on a given pipeline. In some cases, it is possible to perform this task in a straightforward and objective manner. In other cases, a measure of subjective judgment might be required. At a minimum the company must define their leak detection goals for the pipeline and then define the corresponding level of importance relative to the metrics of reliability, sensitivity, accuracy and robustness. Specific performance criteria can be optionally specified in qualitative or quantitative terms, or can be deferred for the vendor to supply following the study.

There are five (5) steps involved in determining the leak detection performance needs for a particular pipeline. Execution of these steps might conceivably require a significant amount of time and effort to complete. However, in many situations it may be possible for the pipeline company to make a quick assessment of its relevant needs, and thus, rapidly establish a comprehensive set of performance requirements. Ultimately, it is up to the pipeline company to determine the amount of time and effort spent on defining its performance objectives for leak detection. The five (5) steps are as follow:

Step 1

The pipeline company must identify any *legal, contractual or regulatory requirements* relating to leak detection. A minimum set of performance criteria must be established to meet these obligations.

Step 2

The pipeline must be characterized in terms of its possible leak mechanisms and the likelihood that one of these will result in a leak. A number of diverse factors are involved in the characterization of the pipeline. These include, but are not necessarily limited to:

- Length and volume of the pipeline,
- Pressure, temperature, and flow rate envelope,
- Terrain over which the pipeline travels,
- Type of fluids transported,
- The installed pipe,
- Pipeline operating procedures,
- Pipeline maintenance procedures, and

- External factors such as nearby roadwork, construction, or land development activity.

Step 3

The next step is to perform an assessment of definite and potential costs associated with incorrect alarms, missed alarms, late alarms, and any other deviation from ideal leak detection system performance. This assessment, when considered alongside the regulatory requirements and the characterization of the pipeline, should provide a basis from which the pipeline company can establish a set of leak detection objectives. Once completed, the task of ranking the performance metrics in order to achieve those objectives can then be concluded.

Step 4

Based on the leak detection objectives for the pipeline, specific performance criteria can now be developed. For this purpose, the performance metrics of reliability, sensitivity, accuracy, and robustness provide the broad categories for specific performance criteria. Within each metric, specific performance criteria can optionally be defined in either qualitative or quantitative terms. This step is optional for the pipeline company, and if quantitative requirements are not provided, they will be supplied by the vendor in the final report.

Step 5

As a final task, the pipeline company should identify the specific features and functions desired along with the leak detection system. These, like the performance metric requirements, can be presented in a tabular form with an indication of whether or not the pipeline company has an interest or has no interest in each.

Further discussion of performance metric issues, formats for presentation to vendors, and specific features and functions can be found in Chapter 3.

Part IV: Transmittal of Information to Vendors

After all configuration, sample data, and performance related information has been prepared, it can be packaged and sent to any number of vendors as desired by the pipeline company. If the information was collected with a particular leak detection technique in mind, it may not be appropriate for all vendor solutions.

If analysis by several vendors is desired, care should be taken to package a complete set of information for distribution to each. At a minimum, the information transmitted to vendors must include a complete and documented configuration file, and case and data files for each desired test scenario. Optionally, items such as performance requirements, pipeline diagrams, representative SCADA system operational displays, etc., may also be supplied.

Part V: The Vendor Analysis Process

Each vendor who receives the information package from a pipeline company will perform several tasks. During this process, the vendor may require the pipeline company to clarify some of the information provided, or possibly, to provide additional information. In general, the vendor will perform the following steps:

<u>Step 1</u>

Review the pipeline's desired performance requirements.

<u>Step 2</u>

Import the configuration as specified by the pipeline company and analyze it as necessary.

<u>Step 3</u>

Perform an evaluation, or study, to estimate achievable leak detection system performance based on the pipeline configuration and sample data provided by the pipeline company.

<u>Step 4</u>

Prepare the final report describing the vendor's ability to achieve the desired level of performance as requested by the pipeline company.

<u>Step 5</u>

Transmit results to the pipeline company.

Part VI: Interpretation of Vendor Results

The format of the vendor's study report will include the following sections:

- Performance Summary,
- Recommendations to the Pipeline Company, and
- Supporting Data.

The Performance Summary will present a summary of the software based leak detection system's performance characteristics, in relation to the pipeline company's desired results. This section will provide all pertinent information regarding the quantitative performance of the system targeted in the study. The information provided in this section will restate the pipeline company's leak detection performance criteria in tabular form, and provide an indication of the leak detection systems' ability to achieve these objectives. The Performance Summary will also provide a response to any specific features or functions desired by the pipeline company.

Performance Metric	Specific Performance Criteria	Operating Conditions
Sensitivity	Maximum volume loss prior to alarm	Flowing
	Minimum detectable leak rate not to exceed _____	Steady State Flow
	Minimum detectable leak volume not to exceed _____	Steady State Flow
	Minimum response time not to exceed _____	Steady State Flow
	Response time not to exceed _____ for _____ leak rate	Flowing
Reliability	Incorrect leak declarations not to exceed _____ per _____	Overall
	Incorrect leak declarations not to exceed _____ per _____	Flowing
	Incorrect leak declarations not to exceed _____ per _____	Static
	Incorrect leak declarations not to exceed _____ per _____	Steady State Flow
	Incorrect leak declarations not to exceed _____ per _____	Startup
	Incorrect leak declarations not to exceed _____ per _____	Shutdown
Robustness	No loss of function due to _____ pressure outage(s)	Overall
	No loss of function due to _____ temperature outage(s)	Overall
	No loss of function due to _____ flow measurement outage(s)	Overall
	No loss of function due to pump state change	Overall
	No loss of function due to valve state change	Overall
	Loss of sensitivity during a pump state change not to exceed _____ % over an interval of _____ or less	Overall
	Loss of sensitivity during a valve state change not to exceed _____ % over an interval of _____ or less	Overall
	Startup stabilization period not to exceed _____	Overall
Accuracy	Leak location error not to exceed _____ for _____ leak rate	Steady State Flow
	Leak flow rate error not to exceed _____	Steady State Flow
	Leak volume error not to exceed _____	Steady State Flow

Figure 11: Example of the tabular presentation format for a vendor's response.

As shown in Figure 11, the vendor's response to the pipeline company's performance objectives will be supplied in a tabular form corresponding to that prepared by the pipeline in Part III of this procedure. If the pipeline elected not to submit any specific criteria, the vendor is expected to provide a table of performance criteria resulting from the analysis. This tabular summary provides a basis for preliminary evaluation of the vendor's response. Items of interest to the pipeline company will be identified and listed in the table, along with the vendor's indication of their leak detection system's ability to satisfy each criteria. Thus, the first step in evaluating a vendor's response is to determine a composite, quantitative measure of performance based on the tabular summary. One obvious means of accomplishing this is to assign an individual figure of merit (e.g., a number in the range 1 to 10) to the vendor response for each performance criterion, weight the figure of merit according to the needs of the pipeline company, and add the weighted figures of merit to obtain a composite indication of the vendor's response to all the performance criteria.

In addition to a tabular summary, the Performance Summary will include the following:

- A fundamental discussion of the vendor's leak detection approach,
- Descriptions of any assumptions made as a part of the study process,
- Identification of any special tuning that may be required,
- Specification of the equipment required to achieve the pipeline company's stated objectives, and
- Vendor's conclusions regarding viability of their leak detection system on the pipeline system under study.

Recommendations to the pipeline company will include items suggesting how configuration changes and/or instrumentation upgrades could improve leak detection performance on the pipeline system(s) studied. Additionally, this section will address any anomalies that might be observed due to variations in fluid properties, temperatures, flow rates, specific transient conditions, and other factors deemed important by the vendor. References to supporting data defining any experimental adjustments applied to the pipeline configuration or data (i.e. adjustments to scan rates, seasonal parameters, instrument characteristics, etc.) will also be provided to aid the pipeline company's analysis process. This section will also include a complete, detailed description of the computer system(s) required to operate the vendor's solution, along with SCADA interface requirements and desired system architectures.

The vendor's recommendations and other supplemental information must be combined with the quantitative measures of performance in order to determine the viability of a particular leak detection system on any given pipeline.

Appendix A

Glossary of Terms

Accuracy (Instrument)	Denotes the measurement performance of an instrument relative to that of an ideal device. It may be expressed absolutely as the worst case measurement error over the range of the instrument, or relatively as a fraction of the instrument's full scale reading.
Accuracy (Performance Metric)	A measure of leak detection system performance related to the estimation of parameters such as leak flow rate, total volume lost, type of fluid lost, and leak location. A system which estimates these parameters within an acceptable degree of tolerance (as defined by the pipeline company) is considered to be accurate.
Alphanumeric Data File	An ASCII file containing captured or simulated data to be used for analyzing the pipeline with respect to leak detection system performance.
Analog-to-Digital Converter (A/D)	An electronic device used to convert an analog signal, such as a pressure or temperature measurement, into digital format. These devices typically reside in the remote terminal unit found at field locations. A/Ds convert analog signals to digital counts according to some pre-determined range and output criteria (e.g., 4ma to 20ma might be converted to a value of 0 to 4095).
ASCII File	Also referred to as a text file, a text-only file, or an ASCII text file. A document or data file produced using a universally recognized text format called ASCII (American Standard Code for Information Interchange). An ASCII file may contain characters, spaces, punctuation, carriage returns, and sometimes tabs along with an end-of-file marker. ASCII files may not contain any specialized formatting information such as that normally included by popular word processing utilities. ASCII files are the format of choice for transferring readable files between dissimilar programs and systems.
Block Structured Data File	A data file containing repeating blocks of information collected over the sampling period of time. Every block must be identical in structure, and may contain a time stamp appearing at the head of each block corresponding to all embedded data records contained within that block, or a time stamp included with each embedded data record.

Boundary Node	A significant point of intersection between volume elements within the pipeline system. These include receipt and delivery locations, points where physical changes occur (i.e., wall thickness changes, burial status, etc.), and device locations (i.e., pump units , flow meters, and other volume elements).
Case File	A "read me" text file which describes the pipeline, the format of the data file, and the test scenario represented by the associated data file. Each and every data file submitted for analysis must have a corresponding case file.
Configuration File	A standardized, keyword record oriented file used to facilitate the transmission of the physical configuration of a pipeline to leak detection system vendor companies. The format of the configuration file is designed to permit a complete and accurate description of all elements of the pipeline that are significant to a leak detection system for any pipeline, or network of pipelines.
Control Node	A physical point on a pipeline where a change in operating condition can occur. These include blocking valves, control valves, and check valves among others. All active valves must be specified in the pipeline configuration file which includes keyword record types to define these points as control nodes.
Correction Algorithm	An algorithm designed to compensate for a certain degree of error encountered during operation, and due to specification and/or measurement inconsistencies related to the pipeline system.
Data Quality	Characteristics of data that describe its potential utility for certain applications. Typical quality information might include indicators for alarm limit violations, manually edited data, old data, and/or non-updated data. Applications that use telemetry information from remote locations often consider data quality to determine the validity of the data relative to the needs of the application.
Decision Output	Output data related to the software based leak detection process. In general, this data is representative of results derived from the mathematical or statistical process employed by the leak detection system or its decision rule mechanism, and those items indicating the likelihood that a leak exists or does not exist on the pipeline.
Decision Rule Mechanism	The methods within a given software based leak detection system employed to determine if the results of the associated analysis process indicate the presence or absence of a leak.

Dual-Ported Volume Element	Refers to pipeline elements that maintain a measurable volume between its inlet and its outlet. The most common of these are pumps, volumetric flow meters, and pipe segments. Less common devices include strainers, separators, and so forth.
Elevation Node	A location along the pipeline where a change in elevation occurs. These locations are normally recorded during the construction of the pipeline, and must be defined in order to adequately represent the elevation profile for leak detection purposes.
Elevation Profile	A collection of elevation nodes which describe the changes in pipe elevation over the length of a given pipeline system. The number of points required in the elevation profile is dependent upon the nature of the pipeline and the type of leak detection system that might be employed.
Fluid Dynamics Model	A mathematical representation, or "model", of a fluid flow problem which employs one or more of the basic equations of fluid mechanics. These are the equations of continuity, momentum, and energy, and are derived from the law of conservation of mass, Newton's second law of motion, and the first law of thermodynamics.
Fluid Properties	The physical properties of a fluid as defined by its density, specific weight, pressure, surface tension, bulk modulus of elasticity, vapor pressure, and viscosity.
Geometry (of Pipeline)	Detailed information about the physical characteristics of the pipeline and its components. Examples include items such as the pipe wall thickness, inside diameter, roughness, etc.
IEEE Floating Point	A commonly used standard published by the Institute of Electrical and Electronics Engineers (IEEE) defining the internal computer representation of floating point numbers. This standard primarily addresses the floating point number format, its resolution, and accuracy. Some small variations in calculation accuracy may occur between dissimilar machines due to specifics related to the implementation of their floating point processors. However, regardless of the internal manipulation of the numbers, IEEE format insures that they can be interpreted by dissimilar equipment.
Incorrect Leak Declaration	A declaration of a probable leak or other condition on the pipeline which proves to be inaccurate or faulty in some form or circumstance.
Keyword	A predefined identifier used to specify record types within a pipeline configuration file or a case file. Within each record, the keyword precedes a list of parameters which provide specific information related to a given item.

Keyword Record	An entry in a pipeline configuration or case file that specifies a given pipeline element or data item. Each record consists of a predefined keyword and a list of related parameters.
Late Alarm	The annunciation of a leak condition after a period of time which exceeds the nominal time required to detect a leak of a given size. In this case, the time required to issue the alarm is inconsistent with the expected time, as projected by the sensitivity threshold for the pipeline.
Leak Mechanism	Refers to the possible causes of a leak on a given pipeline. Possibilities may include items such as construction damage, pipe corrosion, terrain characteristics, etc.
Leak Parameter	Information about a leak condition which might accompany an associated alarm. Some examples of these are leak flow rate, total volume lost, type of fluid lost, and leak location within the pipeline network.
Measurement Node	A point on a pipeline where instrumentation is attached in order to make measurements that are used by the leak detection system.
Mile Post	A reference to a specific point on a pipeline. These references are usually based on survey information and represent the distance from one end of the pipeline, or a significant location on the line.
Minimum Attainable Response Time	The minimum possible time that a leak of any size can be detected on a given pipeline. This time is a function of items such as the physical characteristics of the pipeline, instrumentation locations, fluid properties, and so on. Guidelines for the estimation of this parameter are provided in the API 1149 publication.
Minimum Detectable Leak Size	The minimum possible leak size that can be detected on a given pipeline. This leak size is a function of items such as the physical characteristics of the pipeline, instrumentation locations, fluid properties, etc. Guidelines for the estimation of this parameter are provided in the API 1149 publication.
Missed Alarm	An alarm condition that was not detected. Within the context of a leak detection system, a missed alarm generally refers to a failure of the system to detect and announce the presence of a leak. This can be caused by things such as inadequate instrumentation, broad leak detection sensitivity threshold assignments, limitations within the leak detection system, etc.

Multi-Ported Volume Element	Refers to complex stations, junctions, and other structures which have multiple inlets or outlets. Pump stations and similar installations can usually be represented as a combination of dual-ported volume elements. Simple booster stations can sometimes be represented as a single dual-ported volume element, but more complex installations must be represented as multi-ported volume elements, or combinations of dual-ported volume elements.
Network (of Pipelines)	Defined as an array of pumps, valves and pipe all under the ultimate control of a single SCADA system master station. Within a given network there may be defined one or more distinct pipelines systems, each of which is normally bounded by flow measurement instrumentation and has the capability of operating independent of all the other pipelines systems.
Node	Represents a point on a pipeline which is of significance to the leak detection system. This includes the endpoints of all volume elements, all points where measurement instruments used by the leak detection system are located, and all points required to adequately profile the elevation of the pipeline.
Off-line	Unavailable for use, requiring an action to become on-line. An off-line equipment item or system may be out of service due to a failure, maintenance activities, or other operational conditions. Off-line status is usually specified by human intervention rather than by automatic means.
On-line	Operational and performing assigned normal functions.
Outage Periods	The period of time when equipment or information that is normally available can not be used or acquired.
Performance Criteria	Specific criteria, usually associated with a given performance metric, which define a particular performance objective of the pipeline. It is possible to specify several specific performance criteria for any given performance metric.
Performance Metric	A specific measure of the performance of a leak detection system. These are defined as accuracy, reliability, robustness, and sensitivity.
Pipeline (System)	A distinct part of a pipeline network which is normally bounded by flow measurement instrumentation and has the capability of operating independently of all the other pipelines within the network. Within each pipeline (system) there may exist one or more sections.

Precision (Instrument)	Instrument precision, or resolution, is a measure of the smallest change that can be reflected in the output of the instrument. Precision depends upon the resolution of the analog-to-digital converter used to acquire readings, and/or the transducer itself.
Reliability (Performance Metric)	A measure of the ability of a leak detection system to render accurate decisions about the possible existence of a leak on a pipeline. It is directly related to the probability of detecting a leak, given that a leak does in fact exist, and the probability of incorrectly declaring a leak, given that no leak has occurred.
Repeatability (Instrument)	Instrument repeatability is a measure of the its ability to consistently return the same reading for a given set of measurement conditions. Repeatability may be expressed absolutely as a worst case discrepancy, or relatively as a fraction of the full scale reading.
Response Time	The time required by an instrument to respond to a step change in its input condition, and to reach a specified accuracy of measurement.
Robustness (Performance Metric)	A measure of a leak detection system's ability to continue to function and provide useful information, even under changing conditions of pipeline operation, or loss of data due to pipeline equipment or SCADA system failures. A system is considered to be robust if it continues to perform its principle functions under such less than ideal conditions.
SCADA Point Identifier	The unique key associated with a SCADA point that is used as a reference to access the data associated with that point.
Scan Time	The time between successive readings of data for a specific point. SCADA systems usually have multiple scan times assigned to various data types to reduce unnecessary communication overhead for data that changes slowly.
Scenario	A specific set of operating conditions or events associated with a particular pipeline data file.
Section (of Pipeline)	A collection of pipe segments that are organized to describe a portion of the pipeline by a convenient means. Sections may be defined by boundaries such as line posts, measurement nodes, significant stations or structures, or other criteria.
Segment (of Pipe)	A pre-defined portion of pipe that has its own unique indivisible identity.

Sensitivity (Performance Metric)	A composite measure of the size of leak that a system is capable of detecting, and the time required for the system to issue an alarm in the event that a leak of that size should occur. The relation between leak size and response time is dependent upon the nature of the leak detection system.
Sensitivity Curve	A graphical representation of sensitivity that might be expected from a given leak detection system as applied to a given pipeline system. These curves are normally presented as the relationship between leak detection response time and leak flow rate.
Sequentially Organized Data File	A captured or simulated data file that provides new data values in chronological order as changes occur. This type of structure is analogous to a report by exception data acquisition scheme and is implemented in much the same manner. Each record entry in this type of file must contain a time stamp for the data item, identification of the data item, the data item's numeric value or status, and the data quality associated with the item.
Simulated Operational Data	Pipeline operational data which has been generated by a simulation of the pipeline's operation, rather than by collection from an associated SCADA system.
Snapshot	A collection of data, states, and conditions that represent the environment at a particular instant in time. These collections usually consist of the minimum information required to perform a meaningful analysis of conditions existing at the time the collection is acquired.
Steady State Conditions	Conditions that are continuous and not time dependent.
Threshold	A limit at which a calculated, measured, or composite variable exceeds a predefined condition.
Time Skew	The difference between the actual time a data value is measured and the time it is made available to a system using that data. In a SCADA system, time skew can also refer to the difference between the acquisition time of the first and last data item within a related group (i.e., the time between acquisition of the first and last pressure measurements on a given pipeline system).
Time Stamp	A record of when a given measurement was taken from a transducer, or when it was acquired by the SCADA system. Some remote terminal systems will record measurement times and make them available to the host SCADA system, while others will only record the measured value itself. SCADA systems also vary in their abilities to accept remotely recorded measurement times, and/or to attach a time stamp to acquired values.

Transducer	A device that detects physical conditions and provides signals representative of those physical conditions.
Transient Conditions	Refers to any unsteady flow condition present in a pipeline system. A transient condition is typically encountered when the pipeline system is changing from one steady-state condition to another. These conditions can result from valve changes, pump start-up and shutdown, and pressure control valves, among others. Large, rapid changes create larger transient pressures to be created than do small changes. Transients are also created when a leak occurs on a pipeline.
Tuning Period	A period of time associated with commissioning a leak detection system for on-line operation. This is typically required by leak detection vendors so that final adjustments to the parameters which control the performance of the leak detection system can be made with data collected in real-time from the target pipeline network.
VAX® Floating Point	The internal representation of floating point numbers used by the Digital Equipment Corporation's VAX® computer systems. This format varies somewhat from IEEE format, and is widely used in industrial applications. VAX® floating point format (single precision) consists of a 32-bit representation which includes a sign bit, an eight bit exponent value, and a twenty-three bit fractional part. *VAX® is a registered trademark of Digital Equipment Corporation.*
Volume Element	Any element existing in a pipeline system that maintains a measurable volume. These are subdivided into two categories; dual-ported volume elements and multi-ported volume elements.

Appendix B

Keyword Definition and Record Structure

The pipeline configuration keywords and their corresponding record structures are defined in this section. A complete definition of each record field is provided for all keyword record types along with an example of how each record type might appear in a typical pipeline configuration definition.

Record Format

Each record in a configuration file is composed of a keyword followed by an array of comma-delimited data fields and terminated by a carriage return/linefeed. The data fields are separated from the keyword by one or more consecutive space and/or tab characters. Unless specifically stated otherwise in the record type definitions, all data fields are limited to a maximum of 32 characters.

Blank Fields

In certain cases it is acceptable to leave a data field blank. Such cases occur when the data that would be provided is unnecessary, redundant, or has already been specified in a *default* record. Since the data fields are comma-delimited, blank fields result in adjacent commas within a record. The only exception to this rule occurs if the blank field (or contiguous blank fields) occurs at the end of the record, in which case the trailing comma (or commas) may be omitted. Further details and examples of such situations are provided in the record type definitions.

Blank Lines

Blank lines in a configuration file are ignored. It is recommended that the pipeline engineer make use of this fact in formatting the configuration file for ease of visual inspection and interpretation by leak detection vendor personnel.

Continuation of a Record to the Next Line

In general, each new line of a configuration file constitutes the start of a new record. However, it is possible to extend a record to a second line of text if necessary. If no keyword is specified and the first character of a new line is an ampersand (&) then this is treated as an extension of the current record.

Embedding Comments

Comments within a configuration file may be initiated by a semicolon (;). The semicolon is a reserved character and may not be used within a configuration for any purpose other than to initiate a comment. Comments may be inserted at the end of a record or at the start of a new line. Once a comment is initiated it continues to the end of a line. Thus it is not possible to embed comments between fields of a record. Comments are automatically terminated at the end of a line, so consecutive lines of comment must each begin with a semicolon.

Comments are highly recommended and should be used freely throughout a configuration file. Although most leak detection vendors are expected to eventually develop computer programs to interpret configuration file data, it is likely that some vendors will simply print the file and build their analytical models manually. In any case, the inclusion of supplemental information as comments within the configuration file will in no way impair the leak detection vendor's efforts and can greatly assist in interpretation of configuration file data.

Repeated Records of the Same Type

In certain cases, it is possible to specify a number of consecutive records of the same type without repeating the keyword. This feature has application to tabular data (such as a pipeline elevation profile) and in cases where multiple references to points in the SCADA database originate from the same physical device.

In the first case, the *dTable*, *eTable*, *vpTable*, and *vTable* record types are designed to permit efficient entry of tabular data. With these record types the keyword, as well as any other repeated fields, default to the last referenced value when left blank.

The second situation applies to any keyword record that describes a physical device on the pipeline. In some SCADA systems, there might be multiple instances of data or status associated with a single device; e.g. two pressure readings derived from the same instrument, one of which is sent directly to the SCADA host computer whereas the other is filtered or otherwise processed by a remote terminal. In such cases it might be necessary to represent both instances of SCADA data in the configuration file (without creating the false impression that two distinct physical devices exist). The proper way to do this is to simply enter all instances of SCADA data derived from the same device as adjacent (consecutive) records in the configuration file, and omit the keyword from all but the first record in the group. This tells the leak detection vendor that all keywords in the group refer to the same physical device.

Assignment of Engineering Units

Note that an *engineering units identifier* (EUI) is provided for all record fields defining pipeline SCADA database entries to which engineering units must be applied. This identifier must be used with the *assign* record type to assign engineering units to the corresponding data values in the pipeline database.

Definition of Status Decode Rules

A *decode rule* must be assigned when using *blkValve* and *device* keyword records to specify certain volume elements on a pipeline. The *decodeRule* field is provided for this purpose. This permits the user to assign an arbitrary identifier to a specific mapping between status word values and status attributes. The *decode* keyword record is used to define the mapping itself, and status data for all devices using the same decode rule will be interpreted accordingly by the leak detection vendor.

AMBIENT The **ambient** keyword identifies a record describing the position and characteristics of a real-time ambient air, soil or water temperature measurement that is available to the leak detection system.

Scope ***section***

Syntax **ambient** ***dataID, refPost, type, factor, offset, minimum, maximum, accuracy, precision, repeatibility***

Parameters

dataID Alphanumeric identifier defining the ambient temperature measurement data entry in the pipeline SCADA database. The EUI for this data entry is ***temperature***.

refPost Alphanumeric identifier assigned to a point on the pipeline at or near the point at which the ambient temperature measurement is located. This parameter may be omitted if multiple records are defined for a single device, in which case it will default to the prior reference.

type Alphanumeric identifier defining the type of ambient temperature measurement. The following entries are currently defined for this field:

air	atmospheric temperature measurement
soil	ground temperature measurement at the depth of a buried pipeline
water	water temperature measurement at the depth of a submerged pipeline

Additional entries may be defined by the user if absolutely necessary. A list of all such user definitions must be included as comment statements at the top of the configuration file.

factor ASCII decimal scale factor (typically 1.000) of the ambient temperature measurement data stored in the pipeline SCADA database. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). This is a dimensionless parameter.

offset ASCII decimal offset (typically 0.000) of the ambient temperature measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). The EUI for this parameter is ***temperature***.

	minimum	ASCII decimal minimum ambient temperature reading for reliable measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). The EUI for this parameter is ***temperature***.
	maximum	ASCII decimal maximum ambient temperature reading for reliable measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). The EUI for this parameter is ***temperature***.
	accuracy	ASCII decimal accuracy of the ambient temperature measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). Accuracy should be determined on the basis of a single observation (or SCADA database update) and expressed as a percentage of the full scale range of measurement.
	precision	ASCII decimal precision of the ambient temperature measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). Precision should be determined on the basis of a single observation (or SCADA database update) and expressed as a percentage of the full scale range of measurement. In cases where precision is dominated by the resolution of an N bit A/D converter, this percentage can be normally be calculated as $100 / 2^N$.
	repeatibility	ASCII decimal repeatibility of the ambient temperature measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). Repeatibility should be determined on the basis of a single observation (or SCADA database update) and expressed as a percentage of the full scale range of measurement.
Example	**ambient**	**AD0011, 0000A, soil, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25**

ASSIGN The **assign** keyword identifies a record used to assign engineering units to data entries in the pipeline SCADA data base and other parameters defined in the pipeline configuration file. An engineering units identifier (EUI) is specified herein for each parameter that requires assignment of engineering units. Units assigned to a particular EUI may be specified for a particular record type, or they may be specified with application to all record types. Assigned units apply to the specified EUI until modified by a subsequent **assign** record.

Scope *network, pipeline, section*

Syntax **assign** *keyword, EUI, units*

Parameters

keyword Alphanumeric keyword used to identify the record type to which this engineering unit assignment applies. Allowable entries include any keyword associated with a record containing parameters for which an EUI is defined. The assignment of engineering units applies to all ensuing configuration file data entries of the same keyword and EUI, until modified by another **assign** record.

EUI Alphanumeric engineering units identifier specified in the keyword record type definition for the subject data entry or parameter. Allowable entries are:

capacity
conductivity
density
depth
diameter
distance
elevation
flowRate
flowVolume
length
pressure
roughness
temperature
thickness
velocity
viscosity
volume

units Alphanumeric identifier defining the engineering units that are assigned. Abbreviations may be used for this parameter. Non-standard units may be assigned, in which case they should be defined in the Case File.

Standard U.S. Imperial (US) units and their abbreviations are:

inches	**in**
feet	**ft**
miles	**mi**
slugs	**slug**
cubic feet	**ft**3**
pounds per square inch gauge	**psig**
pounds per square inch absolute	**psia**
degrees Fahrenheit	**degF**
slugs per cubic foot	**slug/ft**3**
feet per second	**ft/s**
barrels per hour	**bph**
barrels	**bbl**
thousand barrels per day	**mbd**
BTU per pound degree Fahrenheit	**BTU/lb degF**
BTU per foot hour degree Fahrenheit	**BTU/ft hr degF**
square feet per second	**ft**2/s**

Standard Metric (SI) units and their abbreviations are:

millimeters	**mm**
centimeters	**cm**
meters	**m**
kilometers	**km**
kilograms	**kg**
cubic meters	**m**3**
kiloPascals	**kPa**
kiloPascals absolute	**kPa abs**
degrees Celsius	**degC**
kilograms per cubic meter	**kg/m**3**
meters per second	**m/s**
cubic meters per hour	**cmph**
kiloJoules per kilogram degree Celsius	**kJ/kg degC**
Watts per meter second degree Celsius	**W/m s degC**
centiPoise	**cP**
centiStoke	**cS**

Standard Meter/Kilogram/Second (MKS) units and their abbreviations are:

meters	**m**
kilograms	**kg**
cubic meters	**m**3**
Pascals	**Pa**
Pascals absolute	**Pa abs**
degrees Kelvin	**degK**
kilograms per cubic meter	**kg/m**3**
meters per second	**m/s**
cubic meters per second	**m**3/s**

Watts per kilogram degree Celsius	**W/kg degC**
Watts per meter second degree Celsius	**W/m s degC**
square meters per second	**m**2/s**

Alternative engineering units may be defined if necessary. A list of any and all such definitions should be included as comment statements at the top of the configuration file.

Example — **assign** — **ambient, temperature, degF**

BLKVALVE

The **blkValve** keyword identifies a record of parameters describing a three-state (open, travel, closed) blocking valve on the pipeline. Active blocking valves are treated as control nodes and must be identified in the pipeline configuration data if they are remotely controlled and/or monitored. It is also advisable to identify inactive valves, even though they may be left open, since some leak detection vendors employ formulas to model the slight pressure drops associated with these devices.

Scope — ***section***

Syntax — **blkValve** — ***stateID, upsPost, dnsPost, decodeRule, fluidType***

Parameters

stateID — Alphanumeric identifier defining the valve state entry in the pipeline SCADA database.

upsPost — Alphanumeric identifier assigned to the point on the pipeline at which the nominal valve inlet is located. This parameter may be omitted if multiple records are defined for a single device, in which case it will default to the prior reference.

dnsPost — Alphanumeric identifier assigned to the point on the pipeline at which the nominal valve outlet is located. This parameter may be omitted if multiple records are defined for a single device, in which case it will default to the prior reference.

decodeRule — Alphanumeric identifier defining the state decode rule that applies to this record. This field is used to locate the **decode** records that define the appropriate state decode rule for the referenced SCADA database entry. It must correspond to a valid identifier that has been established by prior reference in one or more **decode** records within the same **network ... endNetwork** block.

fluidType — Alphanumeric identifier of a fluid type associated with this valve. This parameter can be used to indicate batch initiation and batch termination if the valve controls the introduction of a particular fluid into the pipeline. This parameter may be omitted if no particular fluid type is associated with the valve, or if multiple records are defined for the same device, in which case it will default to the prior reference.

Example **blkValve** **AS1285, 0000A, 0000B, newBlkValve, PUGasA**

CHKVALVE The **chkValve** keyword identifies a record of parameters describing a two-state (open, closed) check valve on the pipeline. Check valves are treated as control nodes and must be identified in the pipeline configuration data since they will always block flow in the reverse direction. Note that the forward flow direction is explicitly stated in the **chkValve** keyword record.

Scope ***section***

Syntax **chkValve** ***identifier, upsPost, dnsPost***

Parameters

identifier Alphanumeric name assigned to the valve. Any unique valve identifier may be used.

upsPost Alphanumeric identifier assigned to the point on the pipeline at which the nominal valve inlet is located.

dnsPost Alphanumeric identifier assigned to the point on the pipeline at which the nominal valve outlet is located.

Example **chkValve** **Sierra, 0723D, 0723E**

CTRLVALVE The **ctrlValve** keyword identifies a record of parameters describing a control valve on the pipeline. Active control valves are treated as control nodes and must be identified in the pipeline configuration data if they are remotely controlled and/or monitored. It is also advisable to identify inactive valves, even though they may be left open, since some leak detection vendors employ formulas to model the slight pressure drops associated with these devices.

Scope ***section***

Syntax **ctrlValve** ***dataID, upsPost, dnsPost, openValue, closeValue***

Parameters

dataID Alphanumeric identifier defining the valve position readback or setpoint data entry in the pipeline SCADA database.

	upsPost	Alphanumeric identifier assigned to the point on the pipeline at which the nominal valve inlet is located. This parameter may be omitted if multiple records are defined for a single device, in which case it will default to the prior reference.
	dnsPost	Alphanumeric identifier assigned to the point on the pipeline at which the nominal valve outlet is located. This parameter may be omitted if multiple records are defined for a single device, in which case it will default to the prior reference.
	openValue	ASCII decimal data entry value reflecting the fully open valve state. This is a dimensionless parameter equal to the readback or setpoint value corresponding to the fully open valve state.
	closeValue	ASCII decimal data entry value reflecting the fully closed valve state. This is a dimensionless parameter equal to the readback or setpoint value corresponding to the fully closed valve state.
Example	**ctrlValve**	**AD0031, 0000A, 0000B, 1.0, 0.0**

DATAQUALITY The **dataQuality** keyword identifies a record used to define a SCADA data quality attribute corresponding to a particular value (or set of values) in a data quality word. Data quality attribute decoding must be consistent throughout a **network ... endNetwork** block.

Scope	***network***	
Syntax	**dataQuality**	***value, attribute***
Parameters	***value***	Alphanumeric binary representation of the value assigned to the data quality attribute provided in the next field of this record. The characters "x" or "X" may be used to represent the "don't care" situation. Thus **xx000011** represents any value in the set {3, 67, 131, 195}.
	state	Alphanumeric identifier defining the data quality attribute assigned to the referenced value. The following standard entries are currently defined for this field:

invalid	current data is not valid
manual	current data has been manually entered
new data	current data is valid and has changed
no change	current data is valid and unchanged
not updated	current data has not been updated

Additional entries may be defined by the user if absolutely necessary. A list of all such user definitions must be included as comment statements at the top of the configuration file.

Example **dataQuality** **xxxxxxxx0100xxxx, new data**

DECODE

The **decode** keyword identifies a record used to define a device state corresponding to a particular value (or set of values) in a corresponding pipeline SCADA database entry. One or more **decode** records are required to define a decode rule. Device state decoding must be consistent throughout a **network ... endNetwork** block.

Scope *network*

Syntax **decode** ***decodeRule, value, state***

Parameters

decodeRule Alphanumeric identifier assigned to the decode rule to which this state decode definition applies. The assignment of ***decodeRule*** identifiers is arbitrary, but it is recommended that properly descriptive identifiers be assigned.

value Alphanumeric binary representation of the value assigned to the state provided in the next field of this record. The characters "x" or "X" may be used to represent the "don't care" situation. Thus **xx000011** represents any value in the set {3, 67, 131, 195}.

state Alphanumeric identifier defining the device state assigned to the referenced value. This identifier should be descriptive of the device state. The following standard entries are currently defined for this field:

inactive	disabled, not in operation
active	enabled and operating
closed	valve is fully closed
not closed	valve is not fully closed
in transit	valve is in transit
not in transit	valve is not in transit
on	device is powered ON
off	device is powered OFF
open	valve is fully open
not open	valve is not fully open
running	device is running
stopped	device is stopped

Additional entries may be defined by the user if absolutely necessary. A list of all such user definitions must be included as comment statements at the top of the configuration file.

Example **decode** **newBlkValve, xxxxxxxx10xxxxxx, close**

DEFAULT The **default** keyword identifies a record used to assign default values for specified fields in other configuration file records. The assignment applies only to records that follow the **default** record in the configuration file. The default value will be assigned if no other value is provided for the specified field. Multiple default assignments of a specific parameter field within the same configuration file are not recommended, but if such multiple assignments are made, the last referenced assignment will take precedence.

Note that the scope of the **default** record depends upon the record type to which it applies. Furthermore, the scope of a default assignment is limited to the block within which the **default** record is invoked. For example, if the **default** record occurs within a **section ... endSection** block, then the default assignment applies only to subsequent records within that particular block. If the default record occurs within a **pipeline ... endPipeline** block, then the assignment applies to subsequent records within that block as well as any **section ... endsection** blocks contained therein.

Scope ***network, pipeline, section***

Syntax **default** ***keyword, field, value***

Parameters ***keyword*** The alphanumeric keyword identifying the record type to which this default parameter value applies.

field The alphanumeric name of the field within the record type to which this default assignment applies.

value The default value to be assigned to the referenced fields.

Example **default** **segment, diameter, 11.250**

DEVICE The **device** keyword identifies a record used to define arbitrary volume elements. This record type can only be used if the pressure drop across the volume element is negligible (e.g. a pig trap), or if the volume element is bounded by pressure measurements (e.g. a pump unit), or if the pipeline user can provide additional data in the ***deltaP*** and ***refQ*** fields.

Scope ***section***

Syntax	**device**	***stateID, upsPost, dnsPost, decodeRule, volume, deltaP, refQ***
Parameters	***stateID***	Alphanumeric identifier defining the device state entry in the pipeline SCADA database.
	upsPost	Alphanumeric identifier assigned to the point on the pipeline at which the nominal device inlet is located. This parameter may be omitted if multiple records are defined for a single device, in which case it will default to the prior reference.
	dnsPost	Alphanumeric identifier assigned to the point on the pipeline at which the nominal device outlet is located. This parameter may be omitted if multiple records are defined for a single device, in which case it will default to the prior reference.
	decodeRule	Alphanumeric identifier defining the state decode rule that applies to this record. This field is used to locate the associated **decode** records that define the appropriate state decode rule for the referenced SCADA database entry. It must correspond to a valid identifier that has been established by prior reference in one or more **decode** records within the same **network ... endNetwork** block.
	volume	ASCII decimal total fluid volume of the device. This parameter may be omitted if multiple records are defined for a single device, in which case it will default to the prior reference. The EUI for this parameter is ***volume***.
	deltaP	ASCII decimal value of pressure drop across the device at the reference flow rate specified in the ***refQ*** field. This parameter may be omitted, in which case no pressure drop across the device will be modeled unless previously specified in a **device** record pertaining to the same piece of equipment. The EUI for this parameter is ***pressure***.
	refQ	ASCII decimal value of the reference flow rate at which the pressure drop specified in the ***deltaP*** field applies. This parameter may be omitted, in which case no pressure drop across the device will be modeled unless previously specified in a **device** record pertaining to the same piece of equipment. The EUI for this parameter is ***flowRate***.
Example	**device**	**AS0091, 0000C, 0000D, meter, 4.5, 2.5, 8000**

DNODE The **dNode** keyword identifies a record of parameters describing a real-time fluid density measurement on the pipeline that is available to the leak detection system.

Scope ***section***

Syntax **dNode** ***dataID, refPost, factor, offset, minimum, maximum, accuracy, precision, repeatibility***

Parameters

dataID Alphanumeric identifier defining the fluid density measurement data entry in the pipeline SCADA database. The EUI for this data entry is ***density***.

refPost Alphanumeric identifier assigned to the point on the pipeline at which the fluid density measurement node is located. This parameter may be omitted if multiple records are defined for a single device, in which case it will default to the prior reference.

factor ASCII decimal scale factor (typically 1.000) of the fluid density measurement data stored in the pipeline SCADA database. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). This is a dimensionless parameter.

offset ASCII decimal offset (typically 0.000) of the fluid density measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). The EUI for this parameter is ***density***.

minimum ASCII decimal minimum fluid density reading for reliable measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). The EUI for this parameter is ***density***.

maximum ASCII decimal maximum fluid density reading for reliable measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). The EUI for this parameter is ***density***.

accuracy ASCII decimal accuracy of the fluid density measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). Accuracy should be determined on the basis of a single observation (or SCADA database update) and expressed as a percentage of the full scale range of measurement.

	precision	ASCII decimal precision of the fluid density measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). Precision should be determined on the basis of a single observation (or SCADA database update) and expressed as a percentage of the full scale range of measurement. In cases where precision is dominated by the resolution of an N bit A/D converter, this percentage can normally be calculated as $100 / 2^N$.
	repeatibility	ASCII decimal repeatibility of the fluid density measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). Repeatibility should be determined on the basis of a single observation (or SCADA database update) and expressed as a percentage of the full scale range of measurement.
Example	**dNode**	**AD0021, 0000A, 1.0, 0.0,0.500, 1.000, 0.25, 0.1, 0.1**

DTABLE

The **dTable** keyword identifies a record used to define the relation between density, pressure, and temperature for a specific fluid. Note that when omitted from a record, the fluid identifier defaults to the last referenced value. This permits efficient entry of a complete table of density data as shown in the example.

Scope	***network***	
Syntax	**dTable**	***pressure, temperature, density, fluidType***
Parameters	***pressure***	ASCII decimal value of fluid pressure. The EUI for this parameter is ***pressure***.
	temperature	ASCII decimal value of fluid temperature. The EUI for this parameter is ***temperature***.
	density	ASCII decimal value of fluid density at the specified temperature. The EUI for this parameter is ***density***.
	fluidType	Alphanumeric identifier of the fluid type to which this dTable entry applies. This parameter may be omitted, in which case

the fluid type identifier established by the last **fluid** or **vTable** record will be assigned.

Example **dTable**

0,	**40,**	**0.895,**	**RUnGasA**
0,	**60,**	**0.887**	
0,	**120,**	**0.875**	
500,	**40,**	**0.915**	
500,	**60,**	**0.907**	
500,	**120,**	**0.895**	
0,	**40,**	**0.890,**	**PUnGasA**
0,	**60,**	**0.882**	
0,	**120,**	**0.870**	
500,	**40,**	**0.910**	
500,	**60,**	**0.902**	
500,	**120,**	**0.890**	

ENDNETWORK The **endNetwork** keyword identifies a record used to terminate the definition of a network; i.e. to terminate a **network ... endNetwork** block. Refer to the **network** keyword for further discussion.

Scope *file*

Syntax **endNetwork**

Parameters This record type has no parameters.

Example **endNetwork**

ENDPIPELINE The **endPipeline** keyword identifies a record used to terminate the definition of a pipeline; i.e. to terminate a **pipeline ... endPipeline** block. Refer to the **pipeline** keyword for further discussion.

Scope *network*

Syntax **endPipeline**

Parameters This record type has no parameters.

Example **endPipeline**

ENDSECTION	The **endSection** keyword identifies a record used to terminate the definition of a section; i.e. to terminate a **section ... endSection** block. Refer to the **section** keyword for further discussion.		
Scope	***pipeline***		
Syntax	**endSection**		
Parameters	This record type has no parameters.		
Example	**endSection**		

ETABLE

The **eTable** keyword identifies a record defining an elevation node; i.e. the location and elevation of a point on the pipeline. This record type is provided to facilitate the specification of the elevation profile of the pipeline.

A contiguous block of **eTable** records can be used to establish an elevation profile across an entire pipeline section. In such cases the records must be arranged in such that the survey measurement in the ***linePost*** field varies monotonically (increasing or decreasing) across the block of records. Valid alphanumeric node identifiers must be provided in the ***refpost*** field for the first and last **eTable** records of the block, and also to bound any discontinuities (or gaps) in the survey measurements (as shown in the example). Identifiers should also be provided in the ***refPost*** field at any intermediate records in which a survey measurement is known to coincide with a defined node.

Scope	***section***	
Syntax	**eTable**	***linePost, elevation, refPost***
Parameters	***linePost***	ASCII decimal value of the linear position or linepost associated with the node; e.g. a survey mile post or kilometer post. The EUI for this parameter is ***distance***.
	elevation	ASCII decimal value of the pipeline elevation at the node location. The EUI for this parameter is ***elevation***.
	refPost	Alphanumeric identifier assigned to the point on the pipeline at which the elevation node is located. This parameter must be provided for the first and last records of a contiguous group of **eTable** records. It must also be provided in the case of an isolated **eTable** record and to bound any discontinuities in the survey measurements provided in the ***linePost*** field.
Example	**eTable**	**000.0, 1234, 0000A ; Alpha station** **001.0, 1256**

002.0,	**1270**		
...	**...**		
...	**...**		
..	**...**		
098.0,	**1505**		
099.0,	**1510,**	**0990A**	**; Bravo station**
100.0,	**1565**		
...	**...**		
...	**...**		
...	**...**		
230.0,	**1705**		
231.0,	**1745**		
231.5,	**1755,**	**2315C**	**; Easy station**

FLUID

The **fluid** keyword initiates a record of parameters used to identify a specific fluid that is transported through a pipeline. The **fluid** record type also provides for specification of standard pressure and temperature for volumetric flow correction, reference to API fluid properties tables or formulas, and specification of certain other fluid characteristics. Detailed fluid properties data can also be provided through the **dTable**, **vTable**, and **vpTable** record types.

Note that adequate fluid properties must be defined for each fluid present in the pipeline during the interval over which data is captured for leak detection system evaluation. Users should discuss fluid properties data requirements with prospective leak detection vendors before developing the configuration file. Vendor requirements vary widely depending upon the nature of their leak detection system.

Further note that the initial line fills specified for each set of operational data should use the same identifiers specified in the corresponding **fluid** keyword records.

Scope *network*

Syntax **fluid** ***fluidType, stdPressure, stdTemp, APItable, APIgravity, heatCapacity, waveSpeed, description***

Parameters

fluidType Alphanumeric identifier used by the pipeline company to refer to the specified fluid.

stdPressure ASCII decimal standard pressure for volumetric flow correction. The EUI for this parameter is ***pressure***.

stdTemp ASCII decimal standard temperature for volumetric flow correction. The EUI for this parameter is ***temperature***.

	APItable	Alphanumeric identifier of the API table or formula describing the characteristics of the specified fluid. This parameter may be omitted if adequate data is available from other sources.
	APIgravity	ASCII decimal API gravity of the specified fluid at the reference pressure and temperature. This parameter may be omitted if adequate data is available from other sources. This is a dimensionless parameter.
	heatCapacity	ASCII decimal heat capacity (specific heat) of the specified fluid at the reference pressure and temperature. This parameter may be omitted if adequate data is available from other sources. The EUI for this parameter is ***capacity***.
	waveSpeed	ASCII decimal acoustic wave speed for the specified fluid at the reference pressure and temperature. This parameter may be omitted if adequate data is available from other sources. The EUI for this parameter is ***velocity***.
	description	Brief description of the fluid identified in this record. This required field is provided for informational purposes. It may extend to a maximum length of 255 characters.
Example	**fluid** **&**	**RUnGasA, 14.7, 60.0, , 28, 0.832, 3000,** **Regular unleaded gasoline ex Alpha refinery**

FNODE The **fNode** keyword identifies a record used to associate a batch and/or fluid type identifier in the SCADA database with a specific point on the pipeline.

Scope	***section***	
Syntax	**fNode**	***dataID, refPost***
Parameters	***dataID***	Alphanumeric identifier defining the location of the fluid type data entry in the pipeline SCADA database.
	refPost	Alphanumeric identifier assigned to the point on the pipeline at which the fluid identification node is located.
Example	**fNode**	**AD0041, 0000A**

NETWORK The **network** keyword identifies a record used to initiate the definition of the physical configuration of a network of pipelines; i.e. to initiate a **network ...**

endNetwork block. The **endNetwork** keyword identifies a record used to terminate the physical configuration definition of a pipeline network. Note that a network consists of one or more pipelines operated from a particular SCADA master station.

Scope	***file***	
Syntax	**network**	***identifier***
Parameters	***identifier***	Alphanumeric name assigned to the pipeline network. The network identifier must be unique within a given configuration file.
Example	**network**	**Upper Case Pipeline Company**

PIPELINE

The **pipeline** keyword identifies a record used to initiate the definition of the physical configuration of a pipeline; i.e. to initiate a **pipeline ... endPipeline** block. The **endPipeline** keyword identifies a record used to terminate the physical configuration definition of the pipeline. Note that one or more pipelines may be defined within a given network. Each distinct pipeline is typically bounded by flow measurement instrumentation and has the capability of operating independently of all other pipelines within the network.

Scope	***network***	
Syntax	**pipeline**	***identifier***
Parameters	***identifier***	Alphanumeric name assigned to the pipeline. The pipeline identifier must be unique within a given configuration file.
Example	**pipeline**	**ABC Pipeline**

PNODE

The **pNode** keyword identifies a record describing the position and characteristics of a real-time fluid pressure measurement on the pipeline that is available to the leak detection system.

Scope	***section***	
Syntax	**pNode**	***dataID, refPost, factor, offset, minimum, maximum, accuracy, precision, repeatibility***
Parameters	***dataID***	Alphanumeric identifier defining the fluid pressure measurement data entry in the pipeline SCADA database. The EUI for this data entry is ***pressure***.

refPost Alphanumeric identifier assigned to the point on the pipeline at which the fluid pressure measurement node is located. This parameter may be omitted if multiple records are defined for a single device, in which case it will default to the prior reference.

factor ASCII decimal scale factor (typically 1.000) of the fluid pressure measurement data stored in the pipeline SCADA database. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). This is a dimensionless parameter.

offset ASCII decimal offset (typically 0.000) of the fluid pressure measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). The EUI for this parameter is ***pressure***.

minimum ASCII decimal minimum fluid pressure reading for reliable measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). The EUI for this parameter is ***pressure***.

maximum ASCII decimal maximum fluid pressure reading for reliable measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). The EUI for this parameter is ***pressure***.

accuracy ASCII decimal accuracy of the fluid pressure measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). Accuracy should be determined on the basis of a single observation (or SCADA database update) and expressed as a percentage of the full scale range of measurement.

precision ASCII decimal precision of the fluid pressure measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). Precision should be determined on the basis of a single observation (or SCADA database update) and expressed as a percentage of the full scale range of measurement. In cases where precision is dominated by the resolution of an N bit A/D converter, this percentage can normally be calculated as $100 / 2^N$.

repeatibility — ASCII decimal repeatibility of the fluid pressure measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). Repeatibility should be determined on the basis of a single observation (or SCADA database update) and expressed as a percentage of the full scale range of measurement.

Example — **pNode** — **AD0023, 0000C, 1.0, 0.0, 0, 1024, 0.5, 0.25, 0.25**

PORT — The **port** keyword identifies a record used to define an inlet, outlet, terminal, or junction between sections.

Scope — ***section***

Syntax — **station** — ***identifier, refPost, portType, sectionX, refPostX***

Parameters

identifier — Alphanumeric name assigned to the port. Any unique port identifier may be used.

refPost — Alphanumeric identifier assigned to the point on the pipeline at which the port is located.

portType — Alphanumeric identifier defining the port type and function. The following standard entries are currently defined for this field:

junction	a port connecting to another section of the pipeline
terminal	a port where fluid enters or leaves the pipeline

Additional entries may be defined by the user if absolutely necessary. A list of all such user definitions must be included as comment statements at the top of the configuration file.

sectionX — Alphanumeric identifier of another section to which this port connects. This parameter is valid only for the junction port type. It must be omitted in all other cases.

refPostX — Alphanumeric identifier assigned to the point of connection in the other section identified by the ***sectionX*** field. This parameter is valid only for the junction port type. It must be omitted in all other cases.

Example **port** **Charlie, 1713A, junction, SouthABC, 0000A**

QNODE

The **qNode** keyword identifies a record of parameters describing a real-time flow measurement on the pipeline that is available to the leak detection system.

The **qNode** record can be used directly to specify the characteristics of certain rate-of-flow measurement devices such as sonic flow meters. The **qNode** record can be used in conjunction with the **device** record to specify the characteristics of turbine meters and similar devices that involve a measurement made between two nodes; i.e. an upstream node at which fluid enters the meter and a downstream node from which fluid exits the meter.

Scope ***section***

Syntax **qNode** ***rateID, flowID, factorID, refPost, flowType, compType, fluidType, minimum, maximum, rollover, increment, accuracy, precision, repeatibility***

Parameters

rateID Alphanumeric identifier defining the location of the (corrected/compensated) totalized flow rate computed and stored for this meter in the pipeline SCADA database. Leave this field blank if no such entry exists in the SCADA database. The EUI for this data entry is ***flowRate***.

flowID Alphanumeric identifier defining the location of the (corrected/compensated) totalized flow volume computed and stored for this meter in the pipeline SCADA database. Leave this field blank if no such entry exists in the SCADA database. The EUI for this data entry is ***flowVolume***.

factorID Alphanumeric identifier defining the location of the current value of the correction factor for this meter in the pipeline SCADA database. This is the correction factor used by the SCADA system. It is normally based on prover runs. Leave this field blank if no such entry exists in the SCADA database. This is a dimensionless factor to which no engineering units are assigned.

refPost Alphanumeric identifier assigned to the point on the pipeline at which the flow measurement node is located. This parameter may be omitted if multiple records are defined for a single device, in which case it will default to the prior reference.

flowType Alphanumeric identifier defining the type of flow measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference).

The following entries are currently defined for this field:

delivery	full or partial stream takeoff or delivery
mainline	mainline flow
receipt	full or partial stream injection or receipt

Additional entries may be defined by the user if absolutely necessary. A list of all such user definitions must be included as comment statements at the top of the configuration file.

compType Alphanumeric identifier defining the type of flow meter compensation used to derive the flow volume entry in the SCADA data base. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference).

The following entries are currently defined for this field:

full	pressure/temperature compensation
pressure	pressure compensation
temperature	temperature compensation
none	uncompensated

Additional entries may be defined by the user if absolutely necessary. A list of all such user definitions must be included as comment statements at the top of the configuration file.

fluidType Alphanumeric identifier defining a fluid type associated with this meter. This is an optional parameter that may be specified in cases where a specific meter is only used for a particular type of fluid. It may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference).

minimum ASCII decimal minimum useful flowrate for accurate volumetric flow metering. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). The EUI for this parameter is ***flowRate***.

maximum ASCII decimal maximum useful flowrate for accurate volumetric flow metering. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). The EUI for this parameter is ***flowRate***.

rollover ASCII decimal rollover value of the volumetric flow measurement totalizer. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). The EUI for this parameter is ***flowVolume***.

increment ASCII decimal volumetric flow measurement increment; e.g. barrels per totalizer count. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). The EUI for this parameter is ***flowVolume***.

accuracy ASCII decimal accuracy of the flowrate or volumetric flow measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). Accuracy should be determined on the basis of a single observation (or SCADA database update) and expressed as a percentage of the full scale range of measurement.

precision ASCII decimal precision of the flowrate or volumetric flow measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). Precision should be determined on the basis of a single observation (or SCADA database update) and expressed as a percentage of the full scale range of measurement. In cases where precision is dominated by the resolution of an N bit A/D converter, this percentage can normally be calculated as $100 / 2^N$.

repeatibility ASCII decimal repeatibility of the flowrate or volumetric flow measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). Repeatibility should be determined on the basis of a single observation (or SCADA database update) and expressed as a percentage of the full scale range of measurement.

Example	**qNode** **&**	**AD0091, AD0092, AD0093, 0000C, receipt, full, , 3000,** **12000, 10000000, 0.01, 0.25, 0.1, 0.1**

REGVALVE

The **regValve** keyword identifies a record of parameters describing a pressure regulator valve on the pipeline. Regulator valves are treated as control nodes and must be identified in the pipeline configuration data since they will affect pressure and flow. Note that the nominal forward flow direction is explicitly stated in the **regValve** keyword record

Scope	***section***	
Syntax	**regValve**	***identifier, upsPost, dnsPost, setPoint***
Parameters	***identifier***	Alphanumeric name assigned to the valve. Any unique valve identifier may be used.
	upsPost	Alphanumeric identifier assigned to the point on the pipeline at which the nominal valve inlet is located.
	dnsPost	Alphanumeric identifier assigned to the point on the pipeline at which the nominal valve outlet is located.
	setPoint	Valve pressure control setpoint. The valve will open or close as required to maintain inlet pressure at this value. The EUI for this parameter is ***pressure***.
Example	**regValve**	**Zulu, 0595D, 0595E, 100**

SECTION

The **section** keyword identifies a record used to initiate the definition of the physical configuration of a section; i.e. to initiate a **section ... endSection** block. The **endSection** keyword identifies a record used to terminate the definition of the physical configuration of the section.

Note that one or more sections may be defined within a given pipeline. Each section is usually associated with an independent survey or measure of linear position along the primary pipeline right-of-way. Aside from short stubs and other minor diversions, it is assumed that the survey measurement of position varies monotonically across the section and that widely separated points within a section cannot be assigned the same survey measurement value. In other words, if two points within the same section are physically separated by five miles they cannot be assigned the same survey mile post number.

Scope	***pipeline***

Syntax	**section**	***identifier***
Parameters	***identifier***	Alphanumeric name assigned to the section. Any arbitrary name may be assigned provided it is applied to one and only one section within a given configuration file.
Example	**section**	**NorthABC**

SEGMENT

The **segment** keyword identifies a record used to define the location and physical parameters of a distinct segment within a pipeline. Note that the segment is effectively located and identified by its section and boundary node assignments.

Note that the assignment of upstream and downstream nodes bounding a segment establishes the nominal direction of forward flow through the segment. Flow in the forward direction is defined as positive, whereas flow in the reverse direction is defined as negative. In general, fluid may flow in either direction through a segment. However, if a check valve is defined within the segment, or at a boundary of the segment, then flow is only permitted in the forward direction of the check valve.

The **default** record type may be used to simplify the specification of a large number of segments that may differ only in one or two parameter values.

Scope	***section***	
Syntax	**segment**	***identifier, upsPost, dnsPost, length, diameter, thickness, elasticity, roughness, insFactor, insThick, cover, depth, wetness***
Parameters	***identifier***	Alphanumeric name assigned to the segment. Any arbitrary name may be assigned provided it is applied to one and only one segment within a given configuration file.
	upsPost	Alphanumeric identifier assigned to the nominal upstream node bounding the segment. Flow into the segment through this node is defined to be in the forward or positive direction. By the same token, flow out of the segment through this node is defined to be in the reverse or negative direction.
	dnsPost	Alphanumeric identifier assigned to the nominal downstream node bounding the segment.
	length	ASCII decimal overall length of the pipe in the segment. The EUI for this parameter is ***length***.

diameter ASCII decimal inside diameter of the pipe in the segment. This parameter may be omitted if an appropriate default value has been specified. The EUI for this parameter is ***diameter***.

thickness ASCII decimal wall thickness of the pipe in the segment. This parameter may be omitted if an appropriate default value has been specified. The EUI for this parameter is ***thickness***.

elasticity ASCII decimal modulus of elasticity (Young's modulus) of the pipe in the segment. This parameter may be omitted if an appropriate default value has been specified. The EUI for this parameter is ***pressure***.

roughness ASCII decimal wall roughness factor of the pipe in the segment. This is an optional parameter that should be provided only if reasonable data is available. It may also be omitted if an appropriate default value has been specified. The EUI for this parameter is ***roughness***.

insFactor ASCII decimal thermal conductivity factor of the pipe insulation over the segment. This parameter should be left blank if the pipe is not insulated. It may also be omitted if an appropriate default value has been specified. The EUI for this parameter is ***conductivity***.

insThick ASCII decimal thickness of the pipe insulation over the segment. A value of zero (0) indicates that the pipe is not insulated. This parameter may be omitted if an appropriate default value has been specified. The EUI for this parameter is ***thickness***.

cover Alphanumeric type descriptor for the pipe cover across the segment. This parameter may be omitted if an appropriate default value has been specified.

The following entries are currently defined for this field:

none	pipe is not covered by soil or water
soil	pipe is buried in the ground
water	pipe is covered by water

Additional entries should be defined by the user as required to describe the pipeline. A list of all such user definitions must be included as comment statements at the top of the configuration file.

	depth	ASCII decimal average effective depth or thickness of the pipe covering in the segment. For buried pipe this may be the actual burial depth as measured from the top of the pipe to the ground surface, or it could be an alternative dimension specified by the leak detection vendor. For example, the radial distance to a point at which the ground temperature is not significantly affected by the pipeline is one commonly used measure of effective cover. This parameter may be omitted if an appropriate default value has been specified. The EUI for this parameter is ***depth***.
	wetness	ASCII decimal value of the average soil wetness (water mass content per unit volume of soil) surrounding the pipe segment during the period in which leak detection test data is captured. This is an optional parameter that should be provided or estimated whenever possible. This parameter may be omitted if an appropriate default value has been specified. The EUI for this parameter is ***density***.
Example	**segment**	**Alpha/Bravo, 0000D, 0990A, 522778, 11.250, 0.375, 30.0E6, 0.0001, , 0, soil, 0.05, 48**

STATION The **station** keyword identifies a record used to define a simple booster pump station as a dual-ported volume element. This record type can be used to simplify the description of a pipeline in cases where the leak detection system can be driven by measurements bounding the station. In such cases it is not necessary to model the individual volume elements comprising the station itself.

Scope	***section***	
Syntax	**station**	***identifier, upsPost, dnsPost, volume***
Parameters	***identifier***	Alphanumeric name assigned to the station. Any unique station identifier may be used.
	upsPost	Alphanumeric identifier assigned to the nominal upstream node bounding the station. Flow into the station through this node is defined to be in the forward or positive direction. By the same token, flow out of the station through this node is defined to be in the reverse or negative direction.
	dnsPost	Alphanumeric identifier assigned to the nominal downstream node bounding the station.
	volume	ASCII decimal total fluid volume connecting the station inlet to the station outlet. The EUI for this parameter is ***volume***.

Example **station** **Bravo, 0990A, 0990B, 33.5**

TNODE The **tNode** keyword identifies a record describing the position and characteristics of a real-time fluid temperature measurement on the pipeline that is available to the leak detection system.

Scope ***section***

Syntax **tNode** ***dataID, refPost, factor, offset, minimum, maximum, accuracy, precision, repeatibility***

Parameters

dataID Alphanumeric identifier defining the fluid temperature measurement data entry in the pipeline SCADA database. The EUI for this data entry is ***temperature.***

refPost Alphanumeric identifier assigned to the point on the pipeline at which the fluid temperature measurement node is located. This parameter may be omitted if multiple records are defined for a single device, in which case it will default to the prior reference.

factor ASCII decimal scale factor (typically 1.000) of the fluid temperature measurement data stored in the pipeline SCADA database. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). This is a dimensionless parameter.

offset ASCII decimal offset (typically 0.000) of the fluid temperature measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). The EUI for this parameter is ***temperature.***

minimum ASCII decimal minimum fluid temperature reading for reliable measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). The EUI for this parameter is ***temperature.***

maximum ASCII decimal maximum fluid temperature reading for reliable measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). The EUI for this parameter is ***temperature.***

	accuracy	ASCII decimal accuracy of the fluid temperature measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). Accuracy should be determined on the basis of a single observation (or SCADA database update) and expressed as a percentage of the full scale range of measurement.
	precision	ASCII decimal precision of the fluid temperature measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). Precision should be determined on the basis of a single observation (or SCADA database update) and expressed as a percentage of the full scale range of measurement. In cases where precision is dominated by the resolution of an N bit A/D converter, this percentage can normally be calculated as $100 / 2^N$.
	repeatibility	ASCII decimal repeatibility of the fluid temperature measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). Repeatibility should be determined on the basis of a single observation (or SCADA database update) and expressed as a percentage of the full scale range of measurement.
Example	**tNode**	**AD0024, 0000C, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25**

VNODE

The **vNode** keyword identifies a record of parameters describing a real-time measurement of kinematic fluid viscosity on the pipeline that is available to the leak detection system.

Scope	***section***	
Syntax	**vNode**	***dataID, refPost, factor, offset, minimum, maximum, accuracy, precision, repeatibility***
Parameters	***dataID***	Alphanumeric identifier defining the fluid viscosity measurement data entry in the pipeline SCADA database. The EUI for this data entry is ***viscosity***.
	refPost	Alphanumeric identifier assigned to the point on the pipeline at which the fluid viscosity measurement node is located. This

parameter may be omitted if multiple records are defined for a single device, in which case it will default to the prior reference.

factor ASCII decimal scale factor (typically 1.000) of the fluid viscosity measurement data stored in the pipeline SCADA database. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). This is a dimensionless parameter.

offset ASCII decimal offset (typically 0.000) of the fluid viscosity measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). The EUI for this parameter is ***viscosity***.

minimum ASCII decimal minimum fluid viscosity reading for reliable measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). The EUI for this parameter is ***viscosity***.

maximum ASCII decimal maximum fluid viscosity reading for reliable measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). The EUI for this parameter is ***viscosity***.

accuracy ASCII decimal accuracy of the fluid viscosity measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). Accuracy should be determined on the basis of a single observation (or SCADA database update) and expressed as a percentage of the full scale range of measurement.

precision ASCII decimal precision of the fluid viscosity measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). Precision should be determined on the basis of a single observation (or SCADA database update) and expressed as a percentage of the full scale range of measurement. In cases where precision is dominated by the resolution of an N bit A/D converter, this percentage can normally be calculated as $100 / 2^N$.

	repeatibility	ASCII decimal repeatibility of the fluid viscosity measurement. This parameter may be omitted if an appropriate default value has been specified (or if multiple records are defined for a single device, in which case it will default to the prior reference). Repeatibility should be determined on the basis of a single observation (or SCADA database update) and expressed as a percentage of the full scale range of measurement.
Example	**vNode**	**AD0022, 0000A, 1.0, 0.0, 5.0, 105.0, 0.5, 0.25, 0.25**

VPTABLE

The **vpTable** keyword identifies a record used to define the relation between vapor pressure and temperature for a specific fluid. Note that when omitted from a record, the fluid identifier defaults to the last referenced value. This permits efficient entry of a complete table of vapor pressure data as shown in the example.

Scope	***network***	
Syntax	**vpTable**	***temperature, vaporPress, fluidType***
Parameters	***temperature***	ASCII decimal value of fluid temperature. The EUI for this parameter is ***temperature***.
	vaporPress	ASCII decimal value of vapor pressure at the specified temperature. The EUI for this parameter is ***pressure***.
	fluidType	Alphanumeric identifier of the fluid type to which this **vpTable** entry applies. This parameter may be omitted, in which case the fluid type identifier established by the last **fluid** or **vpTable** record will be assigned.

Example **vpTable**

40,	**7.7,**	**RUnGasA**
60,	**8.0**	
120,	**8.9**	
40,	**7.7,**	**PUnGasA**
60,	**8.1**	
120,	**9.2**	

VTABLE

The **vTable** keyword identifies a record used to define the relation between viscosity, pressure, and temperature for a specific fluid. Note that when omitted from a record, the fluid identifier defaults to the last referenced value. This permits efficient entry of a complete table of viscosity data as shown in the example.

Scope	***network***	
Syntax	**vTable**	***pressure, temperature, viscosity, fluidType***
Parameters	***pressure***	ASCII decimal value of fluid pressure. The EUI for this parameter is ***pressure.***
	temperature	ASCII decimal value of fluid temperature. The EUI for this parameter is ***temperature.***
	viscosity	ASCII decimal value of fluid viscosity at the specified temperature. The EUI for this parameter is ***viscosity.***
	fluidType	Alphanumeric identifier of the fluid type to which this **vTable** entry applies. This parameter may be omitted, in which case the fluid type identifier established by the last **fluid** or **vTable** record will be assigned.

Example **vTable**

0,	**40,**	**39.9,**	**RUnGasA**
0,	**60,**	**18.2**	
0,	**120,**	**10.3**	
500,	**40,**	**41.1**	
500,	**60,**	**19.4**	
500,	**120,**	**11.5**	
0,	**40,**	**39.7,**	**PUnGasA**
0,	**60,**	**18.1**	
0,	**120,**	**10.2**	
500,	**40,**	**40.9**	
500,	**60,**	**19.3**	
500,	**120,**	**11.4**	

Appendix C

Sample Configuration File for a Network of Three Small Pipelines

An example of a configuration file for a network of three pipelines is provided below. This is a relatively uncomplicated example representing a small pipeline system. Nonetheless, it serves to illustrate the proper configuration file structure.

```
;
;-----------------------------------------------------------------------------------------------
;
network     Upper Case Pipeline Company
;
;           Assignment of units
;
assign      ambient, temperature, degF
assign      device, flowRate, bph
assign      device, pressure, psig
assign      device, volume, bbl
assign      dNode, density, slug/ft**3
assign      dTable, density, slug/ft**3
assign      dTable, pressure, psig
assign      dTable, temperature, degF
assign      eTable, distance, mi
assign      eTable, elevation, ft
assign      fluid, capacity, BTU/lb degF
assign      fluid, pressure, psia
assign      fluid, temperature, degF
assign      fluid, velocity, ft/s
assign      pNode, pressure, psig
assign      qNode, flowRate, bph
assign      qNode, flowVolume, bbl
assign      regValve, pressure, psig
assign      segment, conductivity, BTU/ft hr degF
assign      segment, density, slug/ft**3
assign      segment, depth, in
assign      segment, diameter, in
assign      segment, length, ft
assign      segment, pressure, psia
assign      segment, roughness, in
assign      segment, thickness, in
assign      station, volume, bbl
assign      tNode, temperature, degF
assign      vNode, viscosity, cS
assign      vpTable, temperature, degF
assign      vpTable, pressure, psia
assign      vTable, pressure, psig
assign      vTable, temperature, degF
assign      vTable, viscosity, cS
```

```
;
;            Device state decode rules
;
decode       oldBlkValve, xxxxxxxx01xxxxxx, open
decode       oldBlkValve, xxxxxxxx10xxxxxx, close
decode       newBlkValve, xxxxxxxx01xxxxxx, open
decode       newBlkValve, xxxxxxxx10xxxxxx, close
decode       newBlkValve, xxxxxxxx11xxxxxx, in transit
decode       meter, xxxxxxxxxx01xxxx, active
decode       meter, xxxxxxxxxx10xxxx, inactive
;
;            Data quality word interpretation
;
dataQuality  xxxxxxxx0001xxxx, invalid
dataQuality  xxxxxxxx0010xxxx, manual
dataQuality  xxxxxxxx0100xxxx, new data
dataQuality  xxxxxxxx1000xxxx, no change
;
;            Fluid specifications
;
fluid        RUnGasA, 14.7, 60.0, , 28, 0.832 , 3000, Regular unleaded gasoline ex Alpha refinery
fluid        PUnGasA, 14.7, 60.0, , 29, 0.828, 3000, Premium unleaded gasoline ex Alpha refinery
fluid        HKFuelR, 14.7, 60.0, , 23, 0.728, 3100, Cheap heating oil from Radio refinery
fluid        HKFuelX, 14.7, 60.0, , 23, 0.728, 3100, Cheap heating oil from X-ray refinery
;
dTable       0,     40,     0.895,  RUnGasA
             0,     60,     0.887
             0,     120,    0.875
             500,   40,     0.915
             500,   60,     0.907
             500,   120,    0.895
             0,     40,     0.890,  PUnGasA
             0,     60,     0.882
             0,     120,    0.870
             500,   40,     0.910
             500,   60,     0.902
             500,   120,    0.890
             0,     40,     0.924,  HKFuelR
             0,     60,     0.916
             0,     120,    0.904
             500,   40,     0.934
             500,   60,     0.926
             500,   120,    0.914
             0,     40,     0.924,  HKFuelX
             0,     60,     0.916
             0,     120,    0.904
             500,   40,     0.934
             500,   60,     0.926
             500,   120,    0.914
;
vTable       0,     40,     39.9,   RUnGasA
             0,     60,     18.2
             0,     120,    10.3
```

```
              500,   40,    41.1
              500,   60,    19.4
              500,   120,   11.5
              0,     40,    39.7,   PUnGasA
              0,     60,    18.1
              0,     120,   10.2
              500,   40,    40.9
              500,   60,    19.3
              500,   120,   11.4
              0,     40,    280,    HKFuelR
              0,     60,    132
              0,     120,   110
              500,   40,    290
              500,   60,    142
              500,   120,   120
              0,     40,    280,    HKFuelX
              0,     60,    132
              0,     120,   110
              500,   40,    290
              500,   60,    142
              500,   120,   120
;
vpTable       40,    7.7,   RUnGasA
              60,    8.0
              120,   8.9
              40,    7.7,   PUnGasA
              60,    8.1
              120,   9.2
              40,    17.7,  HKFuelR
              60,    18.0
              120,   19.2
              40,    17.7,  HKFuelX
              60,    18.0
              120,   19.2
;
;             ABC Pipeline subsystem configuration data
;
pipeline      ABC Pipeline
;
default       segment, diameter, 11.250     ; default pipe spec's
default       segment, thickness, 0.375
default       segment, elasticity, 30.0E6
default       segment, roughness, 0.0001
default       segment, insThick, 0          ; uninsulated
default       segment, cover, none          ; uncovered
;
section       NorthABC
port          Alpha, 0000, terminal
ambient       AD0011, 0000A, soil, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
dNode         AD0021, 0000A, 1.0, 0.0,0.500, 1.000, 0.25, 0.1, 0.1
fNode         AD0041, 0000A
vNode         AD0022, 0000A, 1.0, 0.0, 5.0, 105.0, 0.5, 0.25, 0.25
ctrlValve     AD0031, 0000A, 0000B, 1.0, 0.0
```

```
segment     Alpha, 0000B, 0000C, 23.5
pNode       AD0023, 0000C, 1.0, 0.0, 0, 1024, 0.5, 0.25, 0.25
tNode       AD0024, 0000C, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
qNode       AD0091, AD0092, AD0093, 0000C, receipt, full, , 3000, 12000, 10000000, 0.01,
&           0.25, 0.1, 0.1
device      AS0091, 0000C, 0000D, meter, 4.5, 2.5, 8000     ; flow measurement
segment     Alpha/Bravo, 0000D, 0990A, 522778, , , , , , , soil, 0.05, 48
pNode       AD0123, 0990A, 1.0, 0.0, 0, 1024, 0.5, 0.025, 0.25
tNode       AD0124, 0990A, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
station     Bravo, 0990A, 0990B, 33.5
pNode       AD0127, 0990B, 1.0, 0.0, 0, 1024, 0.5, 0.025, 0.25
tNode       AD0128, 0990B, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
segment     Bravo/Charlie, 0990B, 1713A, 381781, , , , , , , soil, 0.05, 48
pNode       AD0223, 1713A, 1.0, 0.0, 0, 1024, 0.5, 0.025, 0.25
tNode       AD0224, 1713A, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
port        Charlie, 1713A, junction, SouthABC, 0000A     ; connection to SouthABC section
station     Charlie, 1713A, 1713B, 33.5
pNode       AD0227, 1713B, 1.0, 0.0, 0, 1024, 0.5, 0.025, 0.25
tNode       AD0228, 1713B, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
segment     Charlie/Delta, 1713B, 2018A, 166610, , , , , , , soil, 0.05, 48
pNode       AD0323, 2018A, 1.0, 0.0, 0, 1024, 0.5, 0.025, 0.25
tNode       AD0324, 2018A, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
station     Delta, 2018A, 2018B, 33.5
pNode       AD0327, 2018B, 1.0, 0.0, 0, 1024, 0.5, 0.025, 0.25
tNode       AD0328, 2018B, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
segment     Delta/Easy, 2018B, 2315A, 155227, , , , , , , soil, 0.05, 48
pNode       AD0423, 2315A, 1.0, 0.0, 0, 1024, 0.5, 0.025, 0.25
tNode       AD0424, 2315A, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
station     Easy, 2315A, 2315B, 35.5
pNode       AD0427, 2315B, 1.0, 0.0, 0, 1024, 0.5, 0.025, 0.25
tNode       AD0428, 2315B, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
qNode       AD0491, AD0492, AD0493, 2315B, delivery, full, , 3000, 12000, 10000000, 0.01,
&           0.25, 0.1, 0.1
device      AS0491, 2315B, 2315C, meter, 4.5, 2.5, 8000     ; flow measurement
port        Easy, 2315C, terminal
;
;           Elevation profile for NorthABC section
;
eTable      000.0,  1234,  0000A        ; Alpha station
            001.0,  1256
            002.0,  1270
            ...     ...
            ...     ...
            ...     ...
            098.0,  1505
            099.0,  1510,  0990A        ; Bravo station
            100.0,  1565
            ...     ...
            ...     ...
            ...     ...
            230.0,  1705
            231.0,  1745
            231.5,  1755,  2315C        ; Easy station
```

```
endSection
;
section      SouthABC
port         Charlie, 0000A, junction, NorthABC, 1713A    ; connection to NorthABC section
fNode        AD1241, 0000A
blkValve     AS1285, 0000A, 0000B, newBlkValve
pNode        AD1233, 0000B, 1.0, 0.0, 0, 1024, 0.5, 0.025, 0.25
tNode        AD1234, 0000B, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
station      Charlie, 0000B, 0000C, 34.5
pNode        AD1237, 0000C, 1.0, 0.0, 0, 1024, 0.5, 0.025, 0.25
tNode        AD1238, 0000C, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
qNode        AD1291, AD1292, AD1293, 0000C, delivery, full, , 3000, 12000, 10000000, 0.01,
&            0.25, 0.1, 0.1
device       AS1291, 0000C, 0000D, meter, 4.5, 2.5, 8000      ; flow measurement
segment      Charlie/Foxtrot, 0000D, 0494A, 260890, , , , , , , soil, 0.05, 48
pNode        AD1323, 0494A, 1.0, 0.0, 0, 1024, 0.5, 0.025, 0.25
tNode        AD1324, 0494A, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
station      Foxtrot, 0494A, 0494B, 37.5
pNode        AD1327, 0494B, 1.0, 0.0, 0, 1024, 0.5, 0.025, 0.25
tNode        AD1328, 0494B, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
segment      Foxtrot/Golf, 0494B, 0913A, 220944, , , , , , , soil, 0.05, 48
pNode        AD1423, 0913A, 1.0, 0.0, 0, 1024, 0.5, 0.025, 0.25
tNode        AD1424, 0913A, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
station      Golf, 0913A, 0913B, 39.5
pNode        AD1427, 0913B, 1.0, 0.0, 0, 1024, 0.5, 0.025, 0.25
tNode        AD1428, 0913B, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
segment      Golf/Hotel, 0913B, 1095A, 96270, , , , , , , soil, 0.05, 48
pNode        AD1523, 1095A, 1.0, 0.0, 0, 1024, 0.5, 0.025, 0.25
tNode        AD1524, 1095A, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
station      Hotel, 1095A, 1095B, 31.5
pNode        AD1527, 1095B, 1.0, 0.0, 0, 1024, 0.5, 0.025, 0.25
tNode        AD1528, 1095B, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
ctrlValve    AD1531, 1095B, 1095C, 1.0, 0.0
segment      Hotel, 1095C, 1095D, 210
regValve     Hotel, 1095D, 1095E, 150
port         Hotel, 1095E, terminal
;
;            Elevation profile for SouthABC section
;
eTable       000.0,  1290,   0000A        ; Charlie station
             001.0,  1281
             002.0,  1270
             ...     ...
             ...     ...
             ...     ...
             109.0,  327
             109.5,  320,    1095E        ; Hotel station
endSection
;
endPipeline
;
;            RST Pipeline subsystem configuration data
;
```

```
pipeline      RST Pipeline
;
default       segment, diameter, 8.500            ; default pipe spec's
default       segment, thickness, 0.250
default       segment, elasticity, 30.0E6
default       segment, roughness, 0.0001
default       segment, insThick, 0                ; uninsulated
default       segment, cover, none                ; uncovered
;
section       OnlyRST
port          Radio, 0000, terminal
ambient       RD0011, 0000A, soil, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
dNode         RD0021, 0000A, 1.0, 0.0,0.500, 1.000, 0.25, 0.1, 0.1
fNode         RD0041, 0000A
vNode         RD0022, 0000A, 1.0, 0.0, 5.0, 105.0, 0.5, 0.25, 0.25
ctrlValve     RD0031, 0000A, 0000B, 1.0, 0.0
segment       Radio, 0000B, 0000C, 19.5
pNode         RD0023, 0000C, 1.0, 0.0, 0, 1024, 0.5, 0.25, 0.25
tNode         RD0024, 0000C, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
qNode         RD0091, RD0092, RD0093, 0000C, receipt, full, , 2000, 8000, 10000000, 0.01,
&             0.25, 0.1, 0.1
device        RS0091, 0000C, 0000D, meter, 3.5, 2.5, 6000      ; flow measurement
segment       DownRadio, 0000D, 0324A, 170993, , , , , , , soil, 0.05, 48
blkValve      RD0901, 0324A, 0324B, oldBlkValve
segment       UpSierra, 0324B, 0723A, 210720, , , , , , , soil, 0.05, 48
pNode         RD0123, 0723A, 1.0, 0.0, 0, 1024, 0.5, 0.025, 0.25
tNode         RD0124, 0723A, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
station       Sierra, 0723A, 0723B, 31.5
pNode         RD0127, 0723B, 1.0, 0.0, 0, 1024, 0.5, 0.025, 0.25
tNode         RD0128, 0723B, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
ctrlValve     RD0131, 0723B, 0723C, 1.0, 0.0
segment       Sierra, 0723C, 0723D, 170
chkValve      Sierra, 0723D, 0723E
port          Sierra, 0723E, terminal
;
eTable        001.0,  1330,   0000A          ; Radio station
              002.0,  1351
              003.0,  1370
              ...     ...
              ...     ...
              ...     ...
              072.0,  1422
              072.3,  1430,   0723E          ; Sierra station
endSection
;
endPipeline
;
;             XYZ Pipeline subsystem configuration data
;
pipeline      XYZ Pipeline
;
default       segment, diameter, 15.250        ; default pipe spec's
default       segment, thickness, 0.375
```

```
default        segment, elasticity, 30.0E6
default        segment, roughness, 0.0001
default        segment, insThick, 0               ; uninsulated
default        segment, cover, none               ; uncovered
;
section        OnlyXYZ
port           X-ray, 0000, terminal
ambient        XD0011, 0000A, soil, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
dNode          XD0021, 0000A, 1.0, 0.0,0.500, 1.000, 0.25, 0.1, 0.1
fNode          XD0041, 0000A
vNode          XD0022, 0000A, 1.0, 0.0, 5.0, 105.0, 0.5, 0.25, 0.25
ctrlValve      XD0031, 0000A, 0000B, 1.0, 0.0
segment        X-ray, 0000B, 0000C, 19.5
pNode          XD0023, 0000C, 1.0, 0.0, 0, 1024, 0.5, 0.25, 0.25
tNode          XD0024, 0000C, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
qNode          XD0091, XY0092, XY0093, 0000C, receipt, full, , 4000, 16000, 10000000, 0.01,
&              0.25, 0.1, 0.1
device         XD0091, 0000C, 0000D, meter, 3.5, 2.5, 6000     ; flow measurement
segment        X-ray/Yankee, 0000D, 0244A, 24.370, , , , , , , soil, 0.05, 48
pNode          XD0123, 0244A, 1.0, 0.0, 0, 1024, 0.5, 0.025, 0.25
tNode          XD0124, 0244a, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
station        Yankee, 0244A, 0244B, 52.5
pNode          XD0127, 0244B, 1.0, 0.0, 0, 1024, 0.5, 0.025, 0.25
tNode          XD0128, 0244B, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
segment        DownYankee, 0244B, 0355A, 11.085, , , , , , , soil, 0.05, 48
blkValve       XD0901, 0355A, 0355B, newBlkValve
segment        UpZulu, 0355B, 0595A, 24.001,, , , , , , soil, 0.05, 48
pNode          XD0123, 0595A, 1.0, 0.0, 0, 1024, 0.5, 0.025, 0.25
tNode          XD0124, 0595A, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
station        Zulu, 0595A, 0595B, 51.5
pNode          XD0127, 0595B, 1.0, 0.0, 0, 1024, 0.5, 0.025, 0.25
tNode          XD0128, 0595B, 1.0, 0.0, -55, 200, 0.5, 0.25, 0.25
ctrlValve      XD0131, 0595B, 0595C, 1.0, 0.0
segment        Zulu, 0595C, 0595D, 110
regValve       Zulu, 0595D, 0595E, 100
port           Zulu, 0595E, terminal
;
eTable         000.0,  1705,   0000A          ; X-ray station
               001.0,  1751
               002.0,  1770
               ...     ...
               ...     ...
               ...     ...
               059.0,   2205
               059.5,  2170,   0595E          ; Zulu station
endSection
endPipeline
endNetwork
;
;------------------------------------------------------------------------------------------------
;
```

Case File Example

In a typical situation the case file might take the following form:

```
;
;-----------------------------------------------------------------------------------------
;
; This file is named README.027.  This file is the CASE file for a DATA file named LEAK027.DAT.
;
; The data provided in LEAK027.DAT applies to a case in which the pipeline was operating in
; near steady state for a period of almost three (3) hours from 10:10 AM until 1:55 PM on July 17.
; At 1:55 PM a metered, partial stream delivery was initiated at the YANKEE station and
; continued until 10:30 PM on July 17.  Between 1:55 PM and 10:30 PM, on July, 17, 1994
; two (2) small, unmetered, partial stream deliveries were taken at other points on the pipeline
; in order to simulate leak conditions.
;
; The following information is pertinent to the data provided in LEAK027.DAT:
;
;              Operating company      Upper Case Pipeline Company
;
;                       Pipeline      XYZ Pipeline
;
;           Responsible engineer      John Doe
;
;               Telephone number      713-555-9999
;
;                     Fax Number      713-555-1111
;
;                  Email Address      pl.engineer@xyzpipeline.com
;
;                Data start time      10:10 AM on July 17, 1994
;
;                 Data stop time      11:25 AM on July 18, 1994
;
;               Initial linepack      BATCH01, 24595/30000 bbls, JP5, receiving at XRAY
;                                     BATCH02, 42500/42500 bbls, SU, inline
;                                     BATCH03, 10464/15000 bbls, RU, delivering at ZULU
;
;                 Data File Type      Blocked
;
;      Average data refresh rate      six (6) data blocks per minute
;                                     roughly 90 percent new data per block
;
;                 Data file name      LEAK027.DAT
;
;
; The following block structure is defined for data file LEAK027.DAT:
;
         status          XYZS0001, hex(4)            ; (first record)
         status          XYZS0002, hex(4)
```

```
            status          XYZS0101, hex(4)
            status          XYZS0102, hex(4)
            status          XYZS0201, hex(4)
            status          XYZS0202, hex(4)
       measurement          XYZD0015, scientific(4, 12)
       measurement          XYZD0021, scientific(4, 12)
       measurement          XYZD0022, scientific(4, 12)
       measurement          XYZD0023, scientific(4, 12)
       measurement          XYZD0024, scientific(4, 12)
       measurement          XYZD0091, hex(8)
       measurement          XYZD0092, hex(8)
       measurement          XYZD0123, scientific(4, 12)
       measurement          XYZD0124, scientific(4, 12)
       measurement          XYZD0133, scientific(4, 12)
       measurement          XYZD0133, scientific(4, 12)
       measurement          XYZD0223, scientific(4, 12)
       measurement          XYZD0224, scientific(4, 12)
       measurement          XYZD0233, scientific(4, 12)
       measurement          XYZD0233, scientific(4, 12)
       measurement          XYZD0091, hex(8)
       measurement          XYZD0092, hex(8)         ; (last record)
;
;
; End of file.
;-----------------------------------------------------------------------------------------------------------
```

1-01400—2/95—4C (3E)

American Petroleum Institute
1220 L Street, Northwest
Washington, D.C. 20005

Order No. 831-11551