SAND2000-0293C

Analysis of Unattended Monitoring System Data Using Knowledge Generation Software

Sharon M. DeLand, John M. Brabson, James D. Smith, Terry I. Jaramillo, Sue M. Spaven
Sandia National Laboratories
P.O. Box 5800
Albuquerque, NM 87185-0977

ABSTRACT

Unattended monitoring systems can reduce the need for on-site human presence while still assuring the proper safeguards of nuclear material. However, such systems generate large quantities of raw sensor data that then have to be related to known or declared activities and material accountancy records. We previously described a concept and technical approach to analyzing this data, based on the use of finite-state machine process models.[1] We have now applied this technique to the analysis of sensor data from unattended monitoring systems at two facilities: an integration laboratory used to simulate material handling facilities in the DOE complex and a bunker used to simulate semi-static storage of high-value assets. The analysis of the integration laboratory data focused on verifying the occurrence of declared activities, even in the presence of "noise" due to people walking around the facility. The analysis of the bunker data considered questions of data integrity and system integrity including how to modify process analysis results based on the quality of the data. The paper will describe the models used to perform the analyses and the results obtained. We will also discuss how additional data could strengthen the conclusions and discuss the implications for monitoring system design.

INTRODUCTION

The Knowledge Generation software analyzes data from unattended monitoring systems in order to compare observed activities to declared activities. Differences, including missing activities and undeclared activities, are summarized and reported to the user. The user may "drill-down" through successively more detailed levels of information in order to examine the conclusions reached by the software.

The data analysis process used by Knowledge Generation consists of three major phases:

1. *Interpretation of the Raw Sensor Data.* In this phase, the sensor data is analyzed and indications of key events are marked. Since the interpretation of sensor data is installation specific, we developed a general-purpose tool called the Event Generator to perform this analysis. The Event Generator is an expert system that is configured with a set of sensor interpretation rules by a site expert. The Event Generator can function as a stand-alone tool.
2. *Construction of Observed Activities.* In this phase, the events found in the first phase are used to construct a list of observed activities. Facility-specific process models are used to both construct the list of activities and identify errors in the observed processes.
3. *Comparison of Observed Activities to Declared Activities.* In this phase, the activities found in the second phase are compared to the list of expected, declared activities. Both extra undeclared processes and missing expected processes are identified.

The data analysis is controlled completely by sensor and process models that are inputs to the software. The analysis models can be configured to track concepts (e.g., continuity of

DISCLAIMER

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

knowledge) as well as processes. Thus, the software is very flexible and can be configured for use in a wide variety of facilities.

Analysis time depends on the number of events and the complexity of the analysis models, but typical performance is one to two thousand events per minute. The analysis models themselves can be built in a fairly short time frame (4-6 weeks, including site/sensor familiarization). Additional software tools could speed up the model-building process significantly. Once built, these models can be applied to all subsequent data coming from the site, as long as the monitoring system configuration remains the same.

PROCESS ANALYSIS (INTEGRATION LABORATORY)

The Integration, Test, and Evaluation Laboratory (ITEL) at Sandia is a facility in which robotic nuclear material handling systems, inventory and control systems, and unattended monitoring systems are brought together in order to identify integration issues in establishing an integrated nuclear material management approach. There is an Automated Guided Vehicle (AGV) that can store and retrieve pallets of containers in storage racks. There is also an overhead gantry that runs two-thirds the length of the room for moving individual containers. As shown in Figure 1, the facility has been instrumented with an unattended monitoring system consisting of two balanced magnetic switches, three breakbeams, four infra-red motion detectors, and two DCM-14 Neumann cameras. The balanced magnetic switches monitor two of the doors to the facility. The third door and nearby area is monitored by one of the motion detectors. One breakbeam is placed near one of the doors and is used to trigger a camera when people enter or leave the facility. The other two breakbeams are placed in the interior of the room to monitor people and the AGV as they move through the facility. Motion detectors are placed near the breakbeams to confirm movements. The fourth motion detector detects motion in the pallet storage racks. Finally, some of the dummy pallets in the facility are monitored with T-1 tags. The T-1 provides authenticated temperature, motion, and fiber optic seal data to monitor the status of a container or set of containers.

ITEL was used to simulate the movement of material from Pantex to Savannah River as part of an integration demonstration in September 1999. In the demonstration, the AGV was sent to the storage racks to retrieve a pallet of four containers similar to those rated for storage of nuclear material. The containers, stored in four-pack pallets at Pantex, would need to be inspected and repackaged into individual containers before shipment to Savannah River. Rather than simulate the inspection and repackaging process, the T-1 was removed from the pallet as it would be in the repackaging process and placed in a vehicle (as if it were on a single container) to simulate the transportation phase. The T-1 was put into "transportation mode" in which the motion detector is disabled and all other events are buffered until a buffer dump is requested by the receiving monitoring system. This allowed any non-motion events en route to "Savannah River" to be captured. While the T-1 was in transportation, the original pallet was replaced in the storage racks. When the T-1 returned to the ITEL, it was placed on a single storage container, which was then moved to its storage location with the overhead gantry. Approximately 15 people attended the demonstration so there was quite a bit of additional human motion that also triggered sensor events.

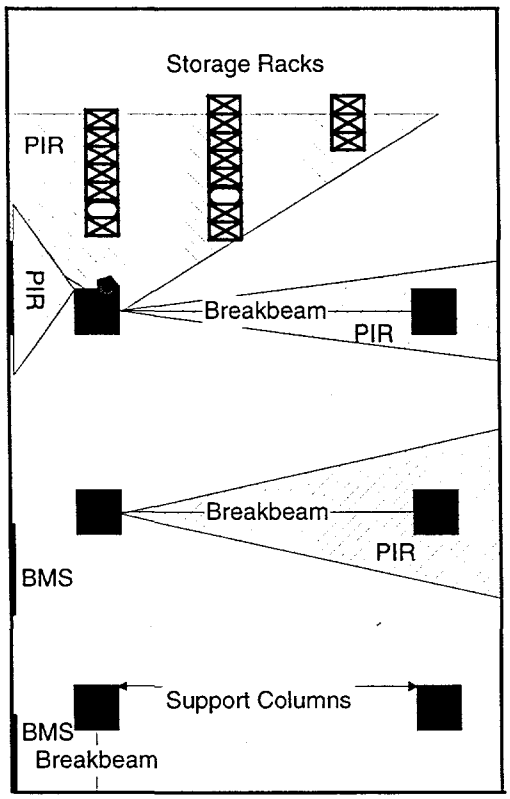

Figure 1. The ITEL Facility showing approximate locations and sensing regions for the sensors. Balanced magnetic switches are labeled BMS and infrared motion detectors are labeled PIR.

There are six high-level objects whose movement or state must be tracked in the facility in the process of analyzing the sensor data. These are the AGV, the pallet, the T-1's fiber optic seal, the truck carrying the T-1, the single container, and people. Each of these objects was modeled with one or more state machines. In a state machine model, an object can be in one of several allowed states (e.g., a door can be "open" or "closed"). Events cause the object to transition from one state to the next. Details of the state machine approach are provided in Ref. 1. In some cases, it was clear what sensor events were associated with what objects. For example, any events from the two balanced magnetic switches on the doors, the associated breakbeam, or the motion detector monitoring the third door were readily ascribed to human motion because the AGV is not allowed in those areas. However, sensor events from the two breakbeams and motion sensors monitoring the middle of the room could come from people, the AGV, or the gantry while it stores the single container. We used lower level state-machine models that incorporated knowledge of how fast the AGV moves to decide how to interpret the events.

The state machine models for the AGV and the single container primarily tracked location of those items. The pallet was tracked with three state machine models, two that tracked its location and one that tracked whether or not it was moving (based on the T-1 motion sensor).

Using these models, we were able to successfully identify all of the expected activities (retrieval of the pallet, "transportation," storage of the pallet, and storage of the single container). We also identified the "seal broken" event as unexpected and of concern. We were able to extract these

activities from a noisy set of fairly random events due to the people attending the demonstration. Results from the Knowledge Generation software are shown in Figure 2.

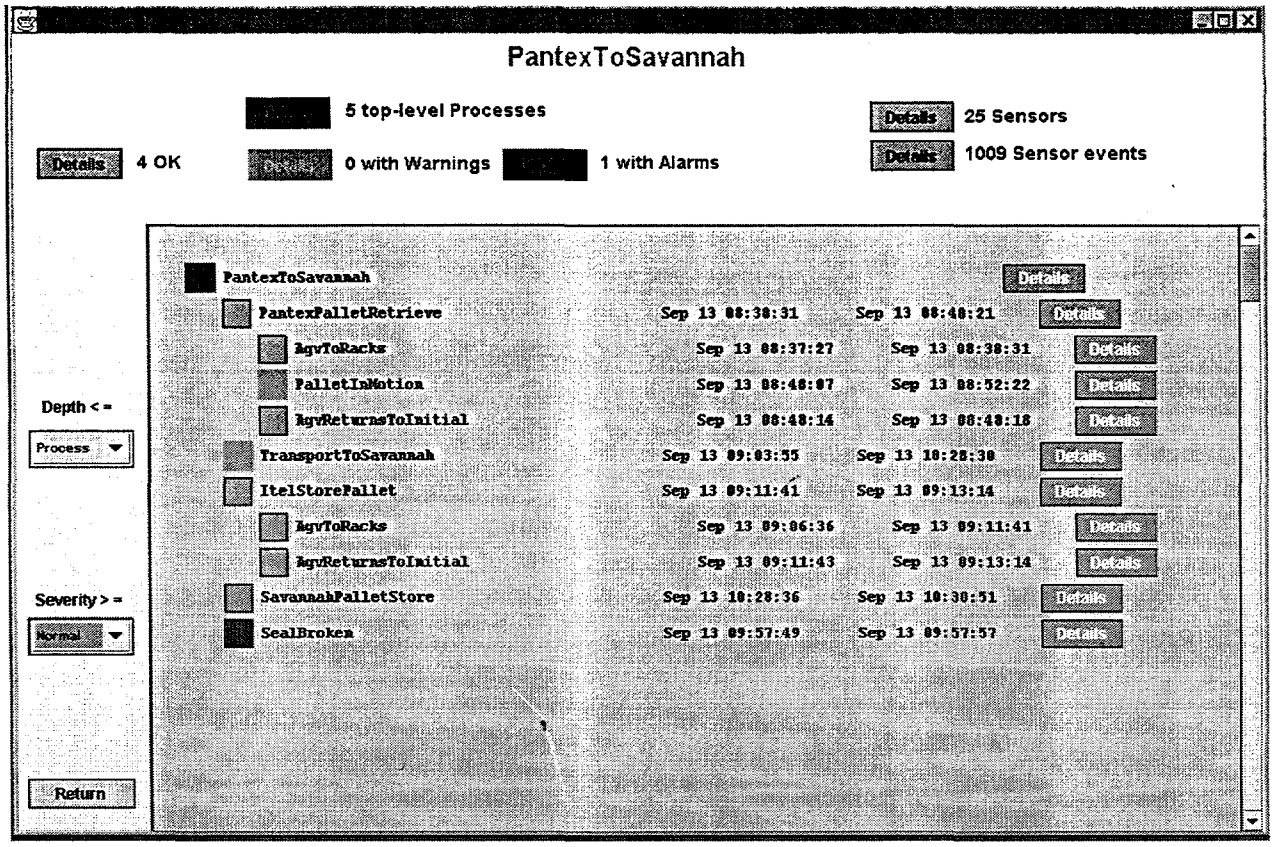

Figure 2. Analysis screen summarizing results. The "summary" area (upper rectangle) indicates how many processes, sensors, and sensor events were found. The processes identified by the analysis are arranged hierarchically in the "results" area (largest rectangle). The date and time fields to the right of the process names indicate the start time and stop time. The square to the left of a process name indicates the completion status of that process: green indicates completion status is OK while red indicates an error. The Details button to the right of each process allows the user to see analysis detail. The Depth and Severity pull-down options on the choice bar (left rectangle) allow the user to select the context for the details (process alone, process timeframe, diagnostic) and the severity of the messages (normal, warning, alarm).

One important point that emerged as the analysis models were being built is that it is fairly difficult to draw strong conclusions from the sensor data. In many cases, the data were consistent with the expected process, but it was hard to argue that this was the only conclusion that could be drawn. For example, distinguishing people motion from the AGV was very difficult because the breakbeam and motion sensors do not give any additional information about what is actually moving. We tried placing the breakbeam high enough that only the AGV would break it under normal operations, but the higher location interfered with other equipment in the facility. Therefore, we had to rely on additional information about the timing between events as the AGV moved to help us in our discrimination. In another example, we used the inference that the AGV

was in the storage racks plus the information that the T-1 motion sensor had gone off to infer that the AGV had picked up the pallet; however, we had no direct evidence that the pallet was actually on the AGV.

DATA INTEGRITY (BUNKER DATA)

The second set of data we analyzed came from an array of sensors in a simulated storage magazine. These sensors were chosen and located so as to assess monitoring options for the storage of high value assets. Our focus in this effort was to explore how to incorporate information about the state of health of the monitoring system into the analysis of the monitored processes.

The magazine had 20 dummy containers, each monitored by a T-1 tag. A balanced magnetic switch monitored the only door to the magazine and two infrared motion sensors monitored movement within the interior. The primary process in the magazine is storage of the items. In an actual operating facility, there should be little activity, except for routine inventory or maintenance. In actuality, the monitoring system in the magazine is undergoing an extensive system test and so there is frequent activity in the magazine. However, we chose to analyze it as if there should be little activity.

In order to address the question of how to incorporate state of health information, our approach was to define the attributes of a good data set and then evaluate whether the data had these attributes. In our definition, "good" data:

- Are complete: all sensor events have been collected and stored
- Are authentic: sensor events come from a known source and have not been modified
- Have fidelity: sensor events are an accurate reflection of observed activity

We identified the generic elements of a monitoring system as the sensing devices, the communication network, the data acquisition component, the data storage component, and the data analysis component. We then considered each of these elements and evaluated how each failure mode would impact the "goodness" of the associated data. Sample indicators are given in Table 1.

Table 1: Attributes of Good Data and Related Failure Indicators

Attribute	Failure Indicators
Completeness	• Missing State-of-Health (SOH) • Incorrect patterns in sensor data • Missing periodic reports • Communication failures • Tamper indicators, …
Authenticity	• Re-authentication failure
Fidelity	• Missing correlations with other sensors • Out-of-date calibration • Tamper indicators, …

In this analysis, we modeled the storage process as a simple state machine in which any of a number of sensor events (e.g., a T-1 fiber optic seal open or T-1 motion) caused the process to go into an error state. We also modeled the attributes of good data (completeness, authenticity, and fidelity) as state machine models. Failure indicators for any of the attributes caused the storage process to go into an error state. The storage process could recover to a normal state if subsequent data indicated that the problem had been corrected. Analysis results using state machine models for the attributes are shown in Figure 3.

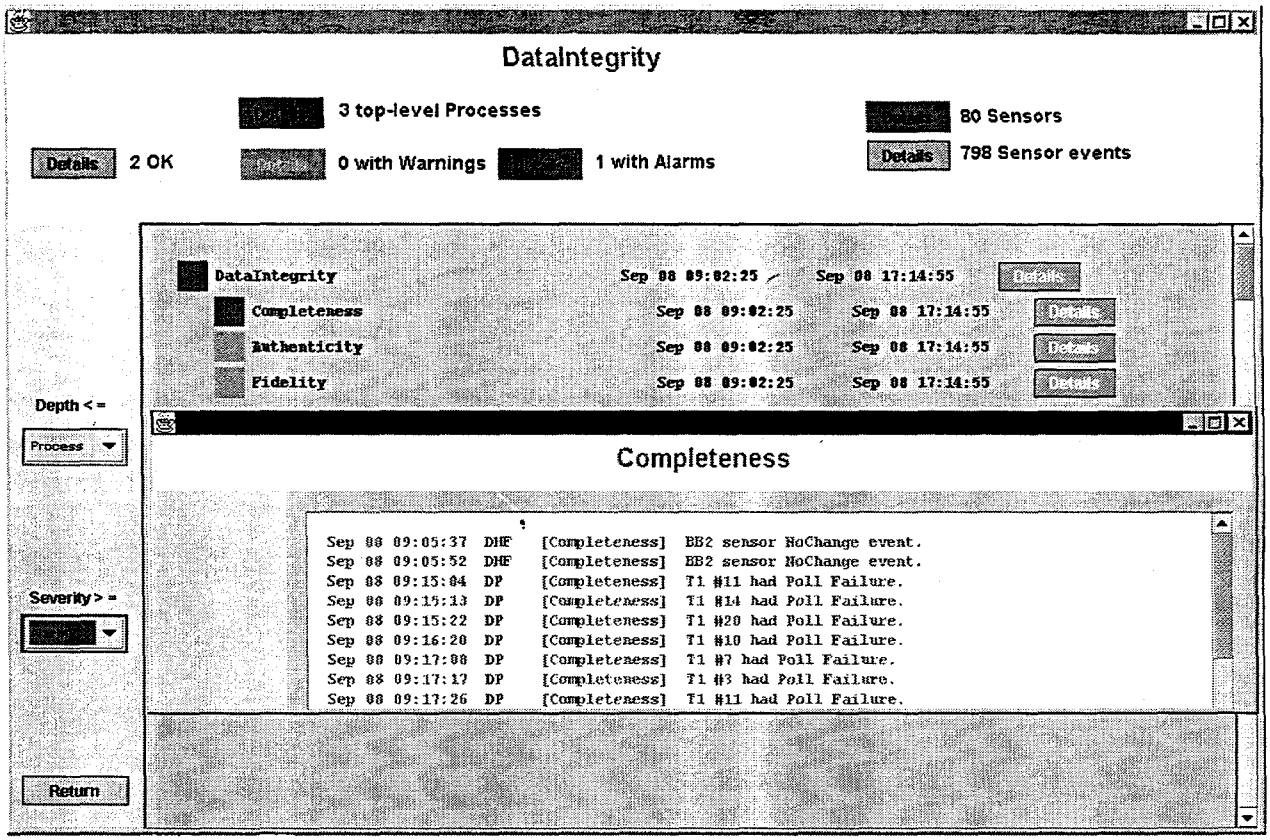

Figure 3. Results of data/system integrity analysis. The inset (Completeness) shows some of the errors found by the Completeness state machine model.

The completeness state machine model tracked events from each of the devices in the monitoring system. T-1s are polled for their state-of-health (SOH), so a missing reply is equivalent to a missing SOH message. If a poll was missed for a given T-1, any subsequent communication from that T-1 restored the health of that device. At the time the analysis was performed, the bunker was in the initial stages of a system evaluation. There were some communication problems that caused poll failures; however, they have since been corrected. The balanced magnetic switch on the door and the breakbeams only report upon change of state. Therefore, there should not be two "like" events (e.g., "door open," or "breakbeam broken") in a row. Another small state machine checked for these repeated events.

We did not check for authenticity because the T-1s use a private key authentication scheme that does not permit the keys to reside on a networked machine. At the time of the analysis, we had no access to any authentication results since the verification would have to be performed off-line and the results fed into the analysis engine in some way. The fidelity state machine focused on checking for correlations between complementary sensors.

We found we were able to model the concepts of completeness, authenticity, and fidelity reasonably well and feed the results into the operational analysis. However, the analysis was hindered by lack of data, especially from monitoring system components other than the sensors. Without such information, it is difficult to locate the real source of many problems. For example, the status of the sensor network and the data collection component were not part of the data set. As a result, a sensor network failure that led to missing sensor SOH events would be indistinguishable from a sensor failure that also caused missing SOH events.

CONCLUSION

We have successfully applied the Knowledge Generation software to the analysis of data from two different monitoring systems. The finite state machine models at the core of the KG analysis engine are flexible enough to model abstract concepts, such as completeness and fidelity, as well as actual physical objects and processes. This approach helps us draw conclusions in the problem domain the user cares about, while still allowing the user to examine the logic trail that led to the conclusions or even to drill down to the raw data if desired. Automating the analysis lets the user quickly sort through large quantities of data and focus on the relatively few unusual occurrences.

Sensor systems can be designed to collect data relevant to diverse goals; e.g., asset monitoring, intrusion detection, process verification, etc. For this reason, it is important to understand what conclusions need to be drawn and how strong they need to be when designing an unattended monitoring system. Is "consistent with" good enough? If not, it is necessary to make sure that the sensors used are able to provide the information (identity or signature level) necessary to allow the analysis to distinguish among the possible trigger events. It is also important to collect status information on all of the devices in the monitoring system, not just the sensors. This allows better diagnosis of a problem and allows the analysis software to better assess the consequences to the overall monitoring system performance of any given failure

ACKNOWLEDGMENTS

The authors are grateful to Dennis Croessmann, Bobby Corbell, Nicole Andrews, and Larry Desonier for assistance in acquiring data and many fruitful discussions about how to analyze it. We also thank Charlie Harmon and Heidi Smartt for comments on an earlier draft of the paper.

Sandia is a multi-program laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000.

REFERENCES

[1] J. M. Brabson, "Finite State Machine Analysis of Remote Sensor Data", presented at the Institute of Nuclear Materials Management 40th Annual Meeting, Phoenix, AZ, July 25-29, 1999.

[2] N. S. Andrews, D. A. Anspach, J. M. Brabson, W. Drotning, T. Jaramillo, J. F. Jones, C. A. Nilsen, W. Pregent, and L. Shipers, "Technology Integration for Weapon Material Stewardship," presented at the Institute of Nuclear Materials Management 40th Annual Meeting, Phoenix, AZ, July 25-29, 1999