

AN INTELLIGENT INFORMATION SYSTEM FOR CRIMINAL CASE MANAGEMENT IN THE FEDERAL COURTS

Jack R. Buchanan*
and
Richard D. Fennell

Innovations and System Development Division
The Federal Judicial Center
1520 H Street, NW
Washington, DC 20005

*Author's current address:
Harvard Business School, Boston, MA 02163

INTRODUCTION

Within the Federal Judicial Center in Washington, D.C., a research project [Ebersole and Hall, 1972; Nihan, 1974; Buchanan, 1975] is being undertaken to develop management information systems for the automation of the administrative workload of the Federal District and Appellate Courts. While the objectives of this project are quite broad, this paper will describe only the criminal case flow management system, called CRIMNL.

CRIMNL differs from conventional information systems in that it contains computer models of court procedure embodying the Federal Rules of Criminal Procedure [F.R.Cr.P., 1976], pertinent regulatory statutes, and local court procedure rules. These models provide various sources of knowledge regarding court procedure which can be utilized within an interactive, diagnostic information system to display their understanding of the semantics of court events by responding as to the correctness and the effects of a particular data entry relative to the model as the system monitors case progress. As an example of a regulatory statute which supplies one of the sources of knowledge in CRIMNL, the Speedy Trial Act of 1974 [Speedy Trial, 1975] establishes a schedule for case disposition that requires a rigorous case monitoring, time accounting, and early warning reporting system. As in most application-oriented AI systems, the systems issues tend to overshadow the AI components [cf., the Hearsay II system, Lesser, *et al.*, 1975], but the intelligence of CRIMNL is derived from the use of court-oriented knowledge sources based on models of criminal court procedure.

The official court docket records all legally significant court transactions as dated line items, called EVENTS. Within CRIMNL, EVENTS are posted interactively from computer terminals. The structure and sequencing inherent in Federal criminal cases are represented within the system, thereby allowing semantic as well as syntactic errors to be detected upon entry.

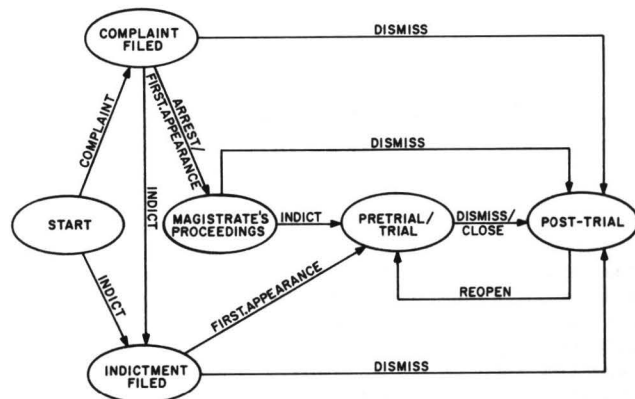
The information contained in the CRIMNL data base includes parties, offenses, related cases, docketed events, judges, attorneys, available resources, and time intervals and schedule regulations. Data base correctness and consistency may be assured by the various data

checks possible with an automated docketing system. For example, any attempt to enter an incompatible matching of party, judge, docket number or attorney in a posted event is easily detected.

More sophisticated checking relative to the model of criminal case processing allows the system to determine whether or not an incoming event is appropriate with respect to the current state of the case, and if not, why not. In many instances, the system can specify required actions which must be taken before the attempted event can be posted. For example, if a docket clerk attempted to enter an "arraignment held" event prior to "first court appearance for preliminary examination," which is its logical antecedent, CRIMNL could inform the user of any outstanding unsatisfied antecedents and require that those events be posted first.

A MODEL OF CRIMINAL CASE PROCESSING

The CRIMNL system is event-driven. EVENTS operate on "subjects," i.e., defendants, counts, motions and appeals. Each of the subjects within a given case moves through a series of legally logical stages, called STATUSES, as the case progresses. Figure 1 illustrates a simplified status diagram for defendant subjects. This portion of the model is a state diagram in which the nodes are STATUSES and the allowable paths (arcs) are EVENTS. The actual CRIMNL diagrams contain 45 statuses and 1700 events, with different diagrams for the various event subjects. STATUSES are chosen to represent the major stages in the legal process convenient for case status reporting; EVENTS represent possible court transactions.



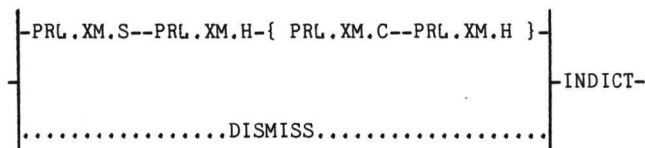
Defendant-based Status Diagram
Figure 1.

The EVENTS are defined in a DICTIONARY by specifying:

1. The event's name, such as "PRL.XM.S" for "preliminary examination set,"
2. The subject type to which the event may apply (e.g., "PRL.XM.S" would apply to "defendant" subjects but not to "motion" subjects),

3. The STATUS transition, if any (e.g., "PRL.XM.S" can occur only if the defendant is in the "Magistrate's Proceedings" status, and the event will not cause a status change),
4. The formal parameters necessary to specify the data contents of the event (e.g., "PRL.XM.S" requires that the scheduled date of the examination be supplied),
5. The Speedy Trial time accounting actions necessary as a result of posting the event,
6. Any special preconditions that must be satisfied in order for the event to be posted,
7. Any special semantic actions that must be performed upon posting the event,
8. The text that is to be printed on a docket sheet when this event is reported.

In addition to the event sequencing constraints represented by the status diagram, arbitrary sequencing relationships may be specified to control the order in which events may be posted within any given status. These relationships are represented graphically as SEQUENCE DIAGRAMS and extend the finite state nature of the STATUS DIAGRAM. In addition to strict event sequences, SEQUENCE DIAGRAMS may denote forks and joins, optional sequences, and parallel paths. The state of any given subject with respect to the progress of a case may be described as the STATUS containing the subject plus a vector of "program counter" values indicating the subject's position along each possible parallel path in the SEQUENCE DIAGRAM which further defines the STATUS.



Event Sequence Diagram for
"Magistrate's Proceedings"
Figure 2.

Figure 2 depicts a simplified event sequence diagram for defendant subjects in the "Magistrate's Proceedings" status, showing the allowable sequences for Setting, Holding, and Continuing preliminary examinations. The { } indicate that the enclosed sequence may be repeated zero or more times. The vertical bars denote a fork and join operation on the enclosed independent event sequences. The dotted line indicates that the DISMISS event may be posted at any time between the fork and join to interrupt and curtail the parallel sequences. According to Figure 2, before an INDICT event can be posted, a preliminary examination must first be set (PRL.XM.S) and then held (PRL.XM.H), and optionally continued (PRL.XM.C) and re-held (PRL.XM.H) an arbitrary number of times. However, at any time during this preliminary examination sequence, DISMISS may be posted, thereby escaping from the sequence and causing a status change to the "Post-Trial" status (see Figure 1).

The event dictionary, status diagrams, and sequence diagrams represent the court-oriented knowledge of the CRIMNL system and are declaratively specified in tables rather than being, for example, procedurally embedded. The system acts essentially as an interactive interpreter which accepts proposed events from the user and tries to apply the semantic actions of each event to transform the current state of its subject. The intelligence of the system is exhibited by its responses to proposed events and the reports produced for effective case monitoring. Not only can CRIMNL detect when an event is invalid based on the current state of its subject, but the system can also offer diagnostic advice as to which events are allowable at that time and which events need to be posted first in order that the attempted event become valid. This model-based validity checking and diagnostic advice allows the detection of clerical errors and results in increased data base integrity. This same model-based system also provides the Courts with a valuable training tool for the instruction of docket clerks in criminal procedure. Furthermore, as a result of explicitly defining each subject's state with respect to its progress in the criminal proceedings, it is possible to generate exception reports which monitor case progress and warn of approaching procedural time constraints, an especially complex clerical task imposed on the Courts by the Speedy Trial Act.

The system is programmed in SAIL using DBMS-10 and runs on a DECSYSTEM-10 computer in Washington, D.C., accessible from terminals located in the various user Courts via a value-added network. The initial "pilot" group of Federal Courts are located in Los Angeles, San Francisco, Chicago, Detroit, New York, and Washington, D.C.

REFERENCES

- Buchanan, J.R., "Management Information Systems for the Federal Courts," Proceedings of "Interactive Information and Decision Support Systems," Office of Naval Research, Wharton School of Management, November 1975.
- Ebersole, J.L., and J.A. Hall, "COURTRAN, An Information System for the Courts," Journal of Computer & Law, Rutgers Univ., 1972.
- Federal Rules of Criminal Procedure for the United States District Courts, August 1976.
- Lesser, V.R., R.D. Fennell, L.D. Erman, and D.R. Reddy, "Organization of the Hearsay II Speech Understanding System," IEEE Trans. ASSP, 23, pp. 11-23, February 1975.
- Nihan, C.W., "COURTRAN, An Assessment of Applications and Computer Requirements," Federal Judicial Center Report, September 1974.
- Speedy Trial Act, Public Law 93-619, 93rd Congress, S.754, January 1975.