| | |
|---|---|
| Document Title: | COPLINK: Database Integration and Access for a Law Enforcement Intranet, Final Report |
| Author(s): | Jennifer Schroeder |
| Document No.: | 190988 |
| Date Received: | October 25, 2001 |
| Award Number: | 97-LB-VX-K023 |

# COPLINK: Database Integration and Access for a Law Enforcement Intranet

October 1, 1997 to February 29, 2000
Final Project Report
Award #-97-LB-VX-K023

**Submitted by:**
**The Tucson Police Department**

Prepared by:
Sgt. Jennifer Schroeder
March 26, 2001

COPLINK: Database Integration and Access for a Law Enforcement Intranet
October 1, 1997 to February 29, 2000
Final Project Report
Award #-97-LB-VX-K023

# Project Summary

## *Partnership Background*

The National Institute of Justice funded the COPLINK project in 1997, creating a partnership between an internationally recognized information technology research group, the University of Arizona Artificial Intelligence Lab (UA AI Lab), and the Tucson Police Department (TPD). Dr. Hsinchun Chen founded the University of Arizona Artificial Intelligence Lab in 1990, and continues as its director. The group is distinguished for its adaptation and development of scalable and practical artificial intelligence, neural networks, genetic algorithms, statistical analysis, automatic indexing, and natural language processing techniques. As a major research group, the Artificial Intelligence Lab employs over 30 full-time staff, research scientists, research assistants, and programmers. Dr. Chen has been heavily involved in fostering digital library research in the US and internationally. He was a PI of the NSF-funded Digital Library Initiative-1 project (1994-1998) and he also recently received another major NSF award (1999-2003) from the new Digital Library Initiative-2 program. Dr. Chen was the guest editor of digital library special issues in IEEE Computer (May 1996 and February 1999) and Journal of the American Society for Information Sciences (1999). He also helped organize the Asia digital library research community and chaired the First Asia Digital Library Workshop, held in Hong Kong in August 1998. Dr. Chen has frequently served as a panel member and/or workshop organizer for major NSF and DARPA research programs. He has helped set directions for several major US initiatives including: the Digital Library Initiative (DLI), the Knowledge and Distributed Intelligence Initiative (KDI), and the Integrated Graduate Education and Research Training (IGERT) program.

The Tucson Police Department was founded on April 22, 1871. The city was one square mile in size and had a population of 3,200 people. The department has grown from one marshal in 1871, to 33 commissioned officers in 1921, to the present police force of 900+ commissioned officers and 300+ civilian personnel. The Tucson Police Department is now responsible for a city of over 200 square miles and over 475,000 citizens.

The partnership between TPD and the UA AI Lab was established specifically to solve information sharing and access problems inherent in law enforcement. Research and development of information technologies in government, with the exception of military applications, tend to lag behind development in business and industry. This ongoing partnership endeavors to provide cutting-edge IT development specifically tailored for the needs of law enforcement agencies' crime-fighting efforts.

## *Project Goals*

The goal of the COPLINK project was to create a proof-of-concept prototype to integrate law enforcement databases and to provide a model for information sharing in a secure law enforcement intranet. The group proposed to integrate law enforcement databases in a data warehousing approach, rather than mediating (translating between differing) databases as has been the approach with some other data sharing efforts. This approach was designed to support the use of sophisticated analytical tools to mine the integrated data.

The project was awarded in July of 1997, with funding made available on October 1, 1997. The initial focus of the project was to evaluate the tools and technologies that would be used in the prototype development, as well as assessment of data sources.

## Database Assessment

Phase I of the project focused on a database assessment to determine which databases would be used in the prototype development. The result of this assessment determined that the central database for the integration effort should be TPD's Records Management System (RMS) since that system contained the bulk (approximately 1.4 million incident records) of the data that TPD was interested in making more accessible and integrating with other databases. The project team also chose TPD's video mugshot system (ELVIS), since the availability of mug photographs was also a high impact area, which would add substantial value to the project. Finally, the group chose TPD's Gang Unit database, since the availability of gang records was of widespread interest to TPD's different investigative and field personnel. Many other data sources were evaluated and continue to be candidates for integration into COPLINK. However, one of the earliest challenges to the project was to limit these data sources for the prototype development. Too many data sources could jeopardize the completion of the project by adding complexity to the prototype without contributing to the proof of concept goal. The group therefore chose a limited scope of these three databases to provide time for more comprehensive architecture, interface, and analysis tool development.

## Database Integration

The database integration (Phase II) was achieved relatively quickly, but the design of the database was continually refined, expanded, and improved throughout the project and continues to undergo change today. Once an initial design was completed and combined with interface development in Phase III, user testing and input demanded almost continuous change and redesign for performance and to accommodate user requirements discovered during prototype testing. The COPLINK database is not intended or designed to be a records management system; it has been designed for read-only performance, with portions of the data denormalized to minimize query time and complexity. Attachment A describes the COPLINK database design.

## Intranet Access System Development

An early focus of the project was to choose the development platform for the interface. Two development platforms were investigated, one based on prevailing HTML/CGI (Common Gateway Interface) and the other based on the dynamic, platform-independent Java. The UA/MIS Artificial Intelligence Group had extensive experience in both HTML/CGI programming and Java system development. The HTML/CGI development tools were stable and robust and could be used immediately for the interface development. The initial research into use of the Concept Space tool for law enforcement (see next section) was accomplished using an HTML/CGI interface. However, for the database functionalities that the group hoped to explore, the team decided to use Java for the integrated database interface. The project team felt that the eventual goals of the system, including wireless access, would be better accomplished by using a standalone Java client interface instead of HTML and Java applets.

The proof-of-concept goal was reached relatively early in the project (a prototype was in testing by approximately September, 1998). The development effort then focused on gaining continued user input from officers, detectives, sergeants, and crime analysts at the Tucson Police Department (TPD) to improve the prototype to make it as useful to law enforcement as possible. A primary goal for the system was to provide an interface that was extremely simple for law enforcement officers to use, decreasing training time and increasing productivity. The ease of use was evaluated extensively throughout the development and during the beta deployment (see attached Deployment report).

The prototype design chosen was a three-tiered design including an Oracle database, an (Oracle) Web application server, and a Java client interface. This three-tiered approach was chosen for flexibility, portability, and scalability.

An integral part of the proof-of-concept was a system design that would support multiple COPLINK nodes in a distributed, multi-agency system. A system design and working prototype was developed and implemented at the UA AI Lab to show an initial distributed system design. This design and plan is documented in Attachment B.

## Concept Space

An early research area for the group focused on the development of Concept Space for use in the law enforcement/COPLINK application. Concept Space is a tool initially developed by Dr. Chen, the head of the Artificial Intelligence Lab, for use in medical research, to facilitate searches by concept on large collections of textual documents such as medical abstracts. This software involves the use of a co-occurrence analysis algorithm to identify and rank associations between objects or terms that exist in the data set. The group modified the application from focusing on the unstructured text of the medical abstracts to the structured fields from TPD's Records Management System. (See Attachment C for more information on the COPLINK Concept Space). The commissioned sergeant and officer that were assigned to the COPLINK project were quick to recognize that this type of sophisticated association analysis had tremendous potential in the law enforcement domain. The group did a preliminary field-testing study involving crime analysts and investigators from the Tucson Police Department with promising results (see Attachment D for details of these results). The COPLINK team continued to improve upon the Concept Space design until the end of the project, creating a more intuitive interface with the same look and feel than that of the main COPLINK interface. Upon completion of the NIJ award, the Tucson Police Department committed additional funds to integrate Concept Space (now called Detect) with the main COPLINK application (now called Connect). This effort is still underway and Detect is undergoing Alpha testing and refinement prior to full deployment, which is scheduled for June of 2001 at TPD.

### Project Communication

The COPLINK project group had no members that had worked with a National Institute of Justice project, and some initial difficulties were encountered by the group. Early in the project, in approximately March of 1998, the NIJ project director for COPLINK and his staff arrived in Tucson for the first project site visit. The project management team at both the University of Arizona and the Tucson Police Department underestimated the importance of the visit and failed to communicate good progress and focus on NIJ's project priorities. The TPD/UA COPLINK project team did not prepare an understandable, comprehensive summary of the project direction, and the lack of clear communication had to be addressed before the members of the NIJ management team were convinced that the project was addressing its assigned research area properly.

Since that time, the COPLINK project team has placed strong emphasis on proper preparation for site visits and clear communication of project progress. The project team has worked hard to publicize the collaborative effort between TPD and the University of Arizona. The various papers attached to this report have been submitted for publication in top-tiered Information Technology and Information Science journals and conferences. In addition to presentations at the prestigious International Conference for Information Systems (ICIS, see Attachment D), COPLINK was also presented at the 2000 SPIE Enabling technologies for law enforcement and security" Conference (see Attachment E).

## Successes

### Statewide Project Interest

The goal of a distributed system prototype to show a proof of concept for multiple agency information sharing was reached in May of 1999 (see Attachment B). By this time, the project had begun to receive widespread attention and interest from other agencies in Arizona who were interested in information sharing. The Phoenix Police Department had a particular interest in the project and sponsored the development group to present the system to officials from the City of Phoenix and many police agencies in the Phoenix Valley and other parts of Arizona. The simplicity of the design and the emphasis on facilitating data sharing gained marked interest for the project among Arizona police agencies.

The Phoenix Police Department committed resources in 2000 to begin implementation of a regional COPLINK node in the Phoenix Valley. The initial data migration is now complete and the Tucson and Phoenix COPLINK nodes will soon begin sharing information via network infrastructure provided by the Arizona Department of Public Safety. Ongoing funding from NIJ will further refine the system and allow expansion of the system to accommodate more data than was available during the early system development.

The level of outside interest the project gained is indicative of the need that this type of project fills. Criminal justice agencies everywhere are cautiously exploring the possibility of more widespread information sharing with their partners, patterned after the trend in this direction by business and industry using secure Internet/intranet technologies.

## *Commercialization*

When COPLINK began, part of the interest that was generated was a result of the unique partnership between law enforcement and the academic community. This partnership is a very positive step towards providing cutting-edge technologies for law enforcement. However, COPLINK at present takes significant effort and technical resources to migrate the records into the integrated data warehouse. In the absence of an entity to implement the system, the only resources for multi-agency implementation were from the U of A or TPD. Neither entity has the resources necessary to support multiple implementations of the system, nor can they provide maintenance for other agency implementations.

Once the application is deployed, it no longer retains the high-risk research interest that was attractive to an educational institution such as the University of Arizona. Additionally, a recurring problem for the COPLINK system development has been the turnover of staff inherent in such a project that is largely staffed by Master's and Ph.D. students. Throughout the project, a researcher would blossom as a software engineer, but then would graduate after becoming indispensable in some facet of the system. This engineer's area of interest must then be passed on to a new researcher, who would take several months to approach the level of expertise equivalent to the graduate.

Recognizing the importance of establishing an entity to eventually support the system, NIJ encouraged dialog with the Office of Law Enforcement Technology Commercialization (OLETC) to discuss the formation of either a non-profit or a for-profit entity to supply this support. A for-profit entity would be the most likely to succeed and survive for a long period of time.

Dr. Chen, the head of the AI Lab and the U of A project director for COPLINK, sought and received venture capital funding during this time (approximately March 1999) and acquired the technology rights for COPLINK Detect (formerly known as Concept Space) and other technologies from the University of Arizona. The company he founded, Knowledge Computing Corporation, or KCC, is now able to allocate resources to implement COPLINK in other police agencies. COPLINK is only one of the products that KCC plans to offer, but it is the first that they are actively marketing.

Several advantages exist for COPLINK user agencies with regard to this commercialization. First, Dr. Chen has been actively recruiting and retaining graduating members of the Artificial Intelligence Lab for hire with KCC. Now the development effort will have the ability to retain the project's best personnel who are already familiar with the COPLINK application. The result is greater project continuity and competent, professional software engineers. Additionally, the University of Arizona charges a high indirect cost rate of 51.5 percent. By subcontracting with Knowledge Computing Corporation, agencies implementing COPLINK avoid this charge; paying instead for more qualified software engineers.

The commercialization of COPLINK brought a new set of political challenges to the scene. The Tucson Police Department has been careful to take a middle ground approach to realize the advantages of the new partnership with a commercial vendor without improperly supporting the product marketing effort.

### TPD Deployment

The original project end date was set for September 30, 1999. By this time, the project was essentially finished, but the group wanted to complete the project by deploying the system at the Tucson Police Department, so a no-cost extension was obtained from NIJ to continue the project until February 29, 2000. During this time, the group conducted additional user studies and further refinements to the system, and completed a subsystem to provide real-time updates to the integrated COPLINK database from TPD's central Records Management System (RMS). The system was essentially ready for deployment by the project end date, but further testing, work, and refinements were necessary to the live update subsystem before full deployment at TPD. This subsystem was tested and retested using University of Arizona and TPD resources until the system was deployed in October of 2000. All authorized members of the Tucson Police Department now have access to the system.

A limited deployment report (see Deployment Report) studies how police personnel are using the system and lists some success stories from the first users of the system. The system can be said to have completely reached the goal of ease of use and system access. With a very short (15 minute) orientation session, the system users are able to get the information they need with virtually no need of technical support.

The goal of widespread information sharing is also coming to fruition with a new consortium of agencies in the State of Arizona committed to sharing information through COPLINK. The Phoenix Police Department committed funding to create an initial prototype containing all records from Phoenix. NIJ has committed additional funding to develop the distributed, open system architecture and to complete a connection between the Phoenix Valley and the Tucson area through COPLINK. The National Science Foundation (NSF) has awarded additional research funding to allow the University of Arizona to continue developing cutting edge technologies to inject into the COPLINK project.

## National Science Foundation/NIJ Collaboration

The COPLINK project is continuing with funding from both the National Institute of Justice and the National Science Foundation (NSF). The NSF research is being conducted solely at the University of Arizona Artificial Intelligence Lab with domain expertise provided by TPD commissioned personnel. The NSF award seeks to continue development of cutting-edge future analysis tools and applications. Continued funding from NIJ will seek to integrate these tools with the present Arizona regional COPLINK system to evaluate and validate this research.

### COPLINK collaborative spider

This development is leveraging the collaborative spider technology that has been developed at the UA AI Lab over the last several years for use in the medical domain. Initial user requirements have been identified through input from the TPD COPLINK personnel, as well as focus groups comprised of TPD crime analysts and detectives. These requirements include: Monitoring of COPLINK databases on a distributed network, monitoring selected web sites on the WWW, collaboration features that allow information sharing between investigators and out to field personnel. Initial ideas for the system also include a forum for police personnel to exchange information in a community format.

### COPLINK Textual analysis

This research was also begun for use in the medical domain, originally with funding from the National Institute of Health. This development explores the use of the Arizona Noun Phraser and automatic entity extraction in the law enforcement domain. Again, this research is being conducted with domain expertise assistance from TPD, and using incident narrative collections provided by the Phoenix Police Department. Since research of this type has not been conducted in the law enforcement domain, significant adaptation of this technology must occur. This includes the training of the noun phraser and entity extraction program to correctly identify phrases that are relevant to law enforcement. This training will likely involve manual tagging in the early phases, which again will be completed by TPD COPLINK personnel.

### Visualization (GIS, Concept Space relationship visualization)

This research is exploring the use of visualization techniques for use in COPLINK Concept Space. The requirements for this research are difficult to define, since these techniques are little used in law enforcement at this time. Therefore, much of the research will entail finding relevant uses for this technology in law enforcement.

One promising area includes the use of hyperbolic trees to graphically display and search for associations and relationships identified through Concept Space. Another area includes the use of self-organizing maps (SOMs) to visually display document content mining results from the textual analysis component.

## Lessons Learned

### Communication

The COPLINK project team learned early on that clear communication about the project is vital in retaining support. Any project team must be prepared to present its project status and direction in a cohesive and comprehensive manner, even early in the project life.

### Changing Technologies

A common factor in any ongoing IT development effort is the effect of changing technologies on the design of an application. Applications and programming languages evolve, and new information system architectures move into focus as this fast-paced industry changes. The COPLINK project team has recognized that changes in technology over the last 3-4 years necessitate continuing to evolve the COPLINK system architecture to take advantage of new strengths and best practices.

The project team made decisions about the system architecture and platform based on the best practices and best estimation of the industry direction at the time. Now, the group has seen some limitations inherent in the system design that oppose the goals of maximum flexibility and interoperability. Therefore, a portion of the continuing funding for COPLINK mandates some changes to maximize interoperability. Developers with the Knowledge Computing Corporation are redesigning the architecture to support any ODBC compliant database and Web server. The interface will now utilize a browser-accessible interface. The data migration strategy for the system is also being refined to minimize the initial costs for implementing new agency or regional nodes.

## Conclusion

The COPLINK system has gathered interest, support, and momentum largely because it fills an important need that is extremely prevalent in the criminal justice community and law enforcement in particular. The criminal justice community must begin to cooperate between agencies, particularly those in neighboring jurisdictions. In this age of widespread and broadening access to information, it is unacceptable to allow lack of information access to give criminals an advantage in both detection and prosecution. The federal funding agencies have recognized this imperative and now must encourage systems that benefit regions instead of individual jurisdictions. Competitive funding in the future will give preference to consortia proposals to discourage development of isolated systems. Therefore, all law enforcement agencies must begin dialog at a county, regional, and state level to begin building more comprehensive plans for regional, state, and national level data sharing.

In the State of Arizona, the Arizona Criminal Justice Commission has taken a leadership and coordinating role in state level IT planning for the criminal justice community. The COPLINK project team will continue to solicit and provide support from and for the ACJC planning efforts in Arizona. The combined University of Arizona, Tucson Police Department, and Knowledge Computing Corporation partnership is planning periodic user dialog sessions involving numerous law enforcement agencies. This dialog will be designed to insure that ongoing development of the COPLINK network conforms to the data sharing needs of as many different agencies as possible. This effort must be combined with input from standards setting efforts at the federal level such as NIBRS.

**Report Prepared by:**

Sgt. Jennifer Schroeder
Tucson Police Department
270 S. Stone Ave.
COPLINK Project
(520) 791-4499 Ext. 1392
jschroe1@ci.tucson.az.us

# Attachment A

# COPLINK: Information and Knowledge Management for Law Enforcement

Roslin V. Hauck, Homa Atabakhsh, Harsh Gupta, Chris Boarman, Kevin Rasmussen, Andy W. Clements, Hsinchun Chen

University of Arizona, Management Information Systems Dept., Artificial Intelligence Lab.
http://ai.bpa.arizona.edu/

*Submitted to IEEE IT Pro for review on July 17, 2000*

## Abstract

Information and knowledge management in a knowledge-intensive and time-critical environment presents a special challenge to information technology professionals. In law enforcement, issues relating to the integration of multiple systems, each having different functions, add another dimension of difficulty for the end user. We have addressed both these problems in the development of our COPLINK Database (DB) application, a model designed to allow diverse police departments to share data more easily through an easy-to-use interface that integrates different data sources. This paper describes how we integrated platform-independence, stability, scalability, and an intuitive graphical user interface to develop the COPLINK system, which is currently being deployed at Tucson Police Department. We describe the resulting database architecture and design and also provide detailed examples of its use. User evaluations of the application allowed us to study the impact of COPLINK on law enforcement personnel as well as to identify requirements for improving the system extending the project.

## 1. Introduction

### 1.1 Law-enforcement Information Sharing

Successful law enforcement depends upon information availability. A police officer on the beat wants to know if the person being interviewed has been involved in previous incidents or is associated with a gang. A detective wants to know if there is a verifiable crime trend in a neighborhood or whether a vehicle involved in one incident is linked to other incidents but it is often difficult to obtain even such basic information promptly.

The problem is not necessarily that the information has not been captured—any officer who fills out up to seven forms per incident can attest to that. The problem is one of access. Typically, law-enforcement agencies have captured data only on paper or have fed it into a database or crime information system. If the agency involved has more than one of these (that are possibly incompatible), information retrieval can be difficult or time-consuming.

A number of government initiatives are trying to address these issues. The Office of Justice Programs (OJP) Integrated Justice Information Technology Initiative is using the resources of five bureaus including the NIJ (National Institute for Justice) in an effort to improve the effectiveness and fairness of the justice system through better information sharing with a focus on wired information technologies. The NIJ wireless initiative is the AGILE program, which falls under the NIJ OS&T (Office of Science and Technology) and primarily addresses interoperability issues. In addition to the COPLINK project, another popular project, called InfoTech and described in section 2, falls within this program (for more information on government initiatives, visit http://www.ojp.usdoj.gov).

## 1.2 A Case Study: The Tucson Police Department

The Tucson Police Department (TPD) has encountered all the problems described in the previous section. Its information sources have included at least three distinct systems:

- The main incident-based system, Records Management System (RMS) captures the highlights of an incident in an Oracle 7.x database.
- A separate system by ImageWare Software Inc. captures mug shots (photos taken at the time of arrest) and limited related information in a Sybase database.
- A third information source, Criminal Information Computer (CIC) is a homegrown Microsoft Access-based application used to track gang activity. TPD officials attribute a disproportionate percentage of Tucson's criminal activity, especially homicides, to gang members and their known associates.

- RMS contains approximately 1.5 million incident record sets and mug shot records (around 23000 mugs). CIC tracks the approximately 1200 individuals the department considers responsible for a majority of major crimes. Each of these systems has a different user interface, so accessing related information from any two or all three, has been difficult, cumbersome, and time-consuming:

- RMS has a cumbersome, difficult-to-navigate command-line driven system.
- CIC's gang database has been accessible only to certain detectives through a simple homegrown front-end interface.
- Mugshot database, a collection of arrest photographs, can only be integrated with information in RMS manually through a specific mug shot number.

As an NIJ-funded multi-year project, the major goals for the COPLINK project for TPD are:

- First, to develop an integrated system to allow TPD officers easy access to all the information contained in all three systems.
- Second, and perhaps more importantly, to design a prototype system for use in developing similar systems at other police departments.
- Finally, with the first two goals in mind, to offer a model for allowing different police departments to share data easily.

Although originally funded by NIJ, COPLINK has received additional funding from both NIJ and the National Science Foundation (NSF) under its Digital Government Initiative. The project is one of many activities of the University of Arizona's Artificial Intelligence Lab, which has gained wide recognition as a cutting-edge research unit and has been featured in *Science* and *The New York Times*. As recipient of more than $9M in research funding from various federal and industrial sponsors since 1989, the Lab sees COPLINK as an opportunity to demonstrate service to the community by bridging the gap between research in developing technologies and solving such real-world problems as helping police officers fight crime.

COPLINK's consistent and intuitive interface integrates different data sources. The multiplicity of data sources remains completely transparent to the user, allowing law enforcement personnel to learn a single, easy-to-use interface. Other law enforcement agencies, including the Phoenix Police Department (PPD), have shown interest in COPLINK. PPD is currently working with the University of Arizona to develop a prototype system for Phoenix-area law enforcement agencies.

## 2. Literature Review: Use of IT and AI in Law Enforcement

Several new federal and business initiatives attempting to transform our information-glutted society into a knowledge-rich society have emerged. In the NSF Knowledge Networking (KN) initiative, scalable techniques to improve semantic and knowledge bandwidths are among the priority research areas. "Knowledge networking," known more generally as "knowledge management" (KM), has attracted significant attention from academic researchers and Fortune 500 company executives.

Information management typically involves the organization, indexing and retrieval of factual, numeric (databases) and textual documents (information retrieval systems), but knowledge management systems, although built upon information management platforms, go one step further to analyze, correlate, summarize and visualize abstract and high-end insights and knowledge of the underlying content. Advanced techniques involving statistical analysis, artificial intelligence, linguistic analysis, neural networks, textual and data mining, and advanced visualization are often needed. Furthermore, adopting such new practices in an organizational context introduces an associated organizational and cultural challenge.

Database technology plays an important role in the management of information for a police department. Previous research has described organization of information in a database system that can be easily searched by officers and other police-department staff (Lewis, 1993; Hoogeveen & Van der Meer, 1994; Miller, 1996; Lingerfelt, 1997; Schellenberg, 1997; Wilcox, 1997). The use of relational database systems for crime-specific cases such as gang-related incidents, and serious crimes such as homicide, aggravated assault, and sexual crimes, has been proven highly effective (Fazlollahi & Gordon, 1993; Pliant, 1996; Wilcox, 1997). Deliberately targeting these criminal areas allows a manageable amount of information to be entered into a database and, in addition, combines information that exists in neighboring police districts.

Automated record-management databases rapidly are replacing paper records of crime and police-report information. Most mid- and large-sized police agencies have made such systems available to their own personnel but lack efficient transmission of information to other agencies. Criminals disregard jurisdictional boundaries and, in fact, take advantage of the lack of communication across jurisdictions. Federal standards initiatives such as the National Incident Based Reporting System (NIBRS) (US Department of Justice, 1998) are aimed at providing reporting standards that will facilitate future reporting and information sharing among police agencies as electronic reporting systems proliferate.

As sharing of police-record information becomes more commonplace, problems of knowledge management faced by business, science, industry and other facets of government will become more prevalent in law enforcement. Increasing ease of capture, retrieval and access is leading to proportional increases in information overload. The large textual collections of report narratives residing in police records have enormous potential as a data source for the development of textual mining and linguistic analysis applications.

In addition to being difficult to manage because of its increasingly voluminous size, knowledge traditionally has been stored on paper or in the minds of people (Davenport, 1995; O'Leary, 1998). In law enforcement, knowledge about criminal activities or specific groups and individuals tends to be learned by officers who work in specific geographic areas. Information may be stored in police databases, but the tools necessary to retrieve and assemble it do not yet exist or are inadequate to the specific task. Solving problems by analyzing and generalizing current criminal records is part of the daily routine of many crime analysts and detectives, but the amount of information confronting them is often overwhelming, a phenomenon often referred to as "information overload" (Blair, 1985). Potent intelligence tools could expedite analysis of available criminal records and aid in investigation of current cases by alleviating information overload and reducing information search time.

As the number of agencies that take advantage of various existing law enforcement information technologies expands, the development of useful artificial intelligence tools continues to progress. Although the many potential uses of databases, intelligence analysis and other technologies have yet to be fully explored (Chen, 1995; Chen & Ng, 1995; Hauck & Chen, 1999), a number of systems currently serve as information management or intelligence analysis tools for law enforcement. The following highlights some of these systems:

- The Timeline Analysis System (TAS) uses visualization and time analysis to examine information and help analysts visually examine large amounts of information by illustrating cause-and-effect relationships. This system graphically depicts relationships found in data, revealing trends or patterns (Pliant 1996).

- Expert systems that employ rule-based information assist in knowledge-intensive activities (Bowen 1994; Brahan 1998) and attempt to aid in information retrieval by drawing upon human heuristics or rules and procedures to investigate tasks.

- INFOTECH International, a Tampa, Florida based company focusing on developing public safety solutions to support information sharing between law enforcement agencies, is hardware-platform independent and Windows-based. The goal is to utilize web-browser and security technology to enable secure data transmission, mainly through the use of a public key infrastructure. For more information visit http://www.infoti.com.

- Falcon (Future Alert Contact Network) is a problem prevention based system or an early-warning system developed at the University of North Carolina at Charlotte that assimilates a request, monitors all incoming records based on the request and then notifies the officer by email or pager when the request is met.

- CCHRS (Consolidated Criminal History Reporting System), developed at Sierra Systems for Los Angeles County, is an example of an integrated justice system that provides justice personnel with consistent and timely identification of individuals. For more information on this project visit http://www.sierrasys.com.

# 3. Design Criteria

The main design criteria considered for the COPLINK project included:

- **Platform independence**: Because not all police departments utilize the same hardware or software operating systems, platform independence was critical.
- **Stability and scalability**: The system also had to offer room for system growth and expansion.
- **Intuitive and ease of use**: The front-end user interface should be intuitive and easy to use, yet flexible enough to meet the equally demanding investigative needs of detectives and officers.

Typical law enforcement applications usually are legacy systems having out-dated performance and capability. For example, TPD's RMS took 30 seconds to answer simple requests and up to 30 minutes for more complex queries. Improved response time was critical to restoring departmental efficiency. To ensure application speed, issues of data and network communication, disk access and system I/O needed to be addressed. This also meant carefully distributing logic where it could be most quickly and efficiently executed, i.e., all user-input error checking should be done in the front end, and all database access logic achieved through pre-compiled stored PL/SQL procedures in the database.

Another critical issue, especially in designing a system that could be deployed across multiple law enforcement agencies, was acknowledging that no two agencies would store their incident data in exactly the same way. Therefore, it was important to come up with a data organization design that was flexible enough to be applied to any underlying data set. The database team designed a series of standardized "views" that fitted typical information search and presentation situations. For example, most of the data in the TPD systems were related to "Person," "Location," "Vehicle," or "Incident" information. A set of views was developed for each of these areas of interest, with the underlying data sets mapped to those standard views, making the system more portable to other law enforcement agencies.

# 4. COPLINK Database Application

## 4.1. Architecture

Based on the criteria established and after much investigation, the COPLINK team decided upon a three-tier architecture (see Figure 1):

- A front-end interface: The front-end should be a thin client, consisting of a series of user-friendly query screens matching the four main areas previously discussed (Person, Location, Vehicle, and Incident). The front-end would generate query requests.
- A middle-ware application server: The middle-ware would handle secure requests from multiple clients, and execute the stored procedures in the database.
- A back-end database: Results from the database would be processed by the middle-ware, and be formatted into return data strings. These return strings would then be sent to the front-end where they would be parsed and displayed to the user.

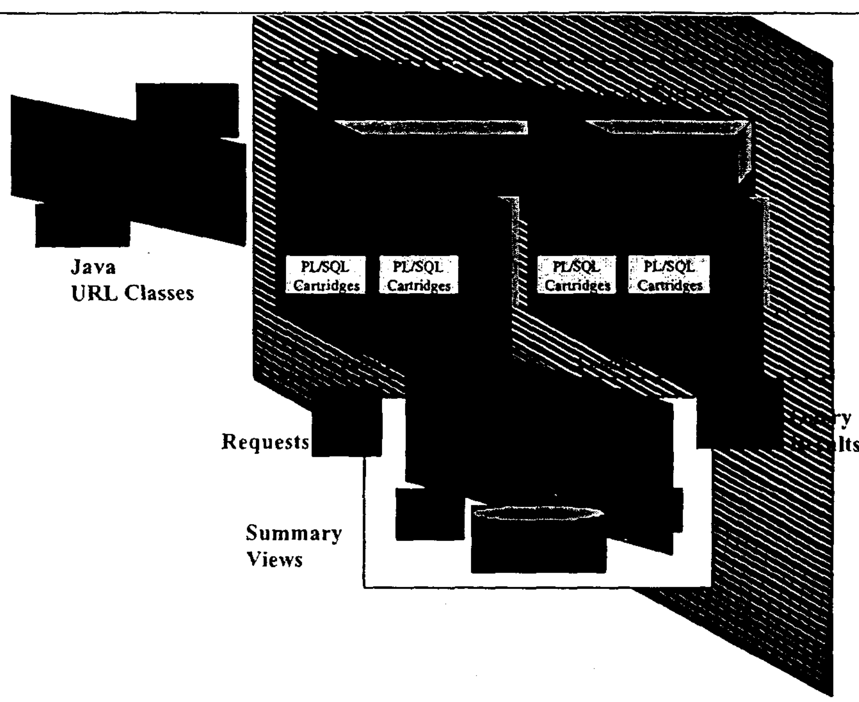



**Figure 1: COPLINK's Three-tier Architecture**

As mentioned, the front-end had to be a platform-independent thin, stable client, based on a popular programming language. Browser scripting languages (such as JavaScript and VBScript) and Java were considered, as was utilizing other popular languages (like Perl, C and C++). The latter were seen as resulting in more platform-dependent code and hence were not used. Two separate prototypes were developed, in JavaScript and Java.

Oracle's Application Server (OAS) met our middle-ware needs. It has versions available for both Windows NT and UNIX-based systems and utilizes a CORBA-based "cartridge server" system. A cartridge server is a shared library that either implements program logic or provides access to program logic stored elsewhere, such as in a database.

In implementing the COPLINK application, we utilized the PL/SQL cartridge system of the OAS, which gives access to the logic stored as pre-compiled PL/SQL procedures in the database. The procedures actually execute the queries in the database, and return the results to the front-end application as HTTP-based strings. Although this system appears to be Oracle-centered, it has flexibility that allows us to access non-Oracle databases whereas such a cartridge as ODBC could only be used to access an ODBC-compliant database.

The database system was designed to be compatible with either Oracle 7.3 or 8.0, and different versions of the data sets have been run on Windows NT and Dec Alpha UNIX platforms. The major portions of the database consist of tables and indices that contain incident-based information, the set of views discussed previously, a series of procedures used by the middle-ware to query the database, and the packages necessary to execute queries from the OAS.

## 4.2. History and Design Considerations

### 4.2.1 Interface Issues

An initial prototype of COPLINK was developed first, using a combination of HTML and JavaScript. Unfortunately, using HTML/JavaScript resulted in a browser conflict. Because Netscape's Navigator and Microsoft's Explorer used different code bases, we experienced significant performance and behavior differences between the two. A decision to deal with the different browsers by writing two sets of code (JavaScript for Netscape and VBScript for Microsoft) proved unfeasible because it violated our goal of platform independence. We reached a design compromise and decided to standardize on Netscape's Navigator, at least for the initial development phase. This solution resulted in over-large script files (approx. 20-30K), which resulted in unacceptable download times. We needed a faster way to send information back and forth between the front end and the OAS.

Our current prototype, created using Java 1.1, not only is compatible with both browsers but also enables us to compress the applet into JAR (Java Archive) files for quicker download time. The JAR files have to be downloaded to the local machine only once, so although the user must wait 30 seconds to download the files and start the Java virtual machine, queries to the database require much less time. The use of Java allows for client-side analysis, avoiding the overhead incurred by database operations.

### 4.2.2 Middle-ware Issues

As mentioned previously, the Oracle Application Server utilizes a CORBA-based cartridge server system to handle incoming requests. After utilizing several OAS versions, we settled on version 4.0.8. Among several issues involved in properly configuring the OAS, the major problem was configuring the cartridges to be stable instances that would remain instantiated even if there were no incoming requests.

After much trial and error, the PL/SQL cartridge we utilized was set up to maintain constant connections to the database. Multiple instances of a cartridge are initialized upon server setup. Through a listener, incoming query requests are routed via a pre-designated virtual path to an available cartridge for processing. The cartridge sends the request to the database, with specific name-value pairs as search criteria. The stored PL/SQL procedure is executed in the database, and the results are sent back to the front-end as a long data string.

One of the significant challenges in developing the PL/SQL code was the fact that each front-end query screen contained up to eleven possible fields that a user could use as valid input. Queries could be any possible combination of one to eleven different query fields.

We determined the best solution to be utilizing the DBMS_SQL package provided by Oracle, thereby constructing each query dynamically, based on the fields a user actually inputs. The final versions of the PL/SQL procedures utilize this approach; dynamically building and executing the initial queries based on only the fields the user inputs. This solved many performance and design problems within the database.

## 5. Database Design

As previously mentioned, the data set of the integrated database system could be logically divided along four main areas: Person, Location, Incident, and Vehicle. However, analysis of the bulk of the database setup revealed two strong candidates for very tightly organized information: Person and Incident. Most of the information in any crime analysis situation is incident- or person-based, and most of the underlying tables were based on a schema organized around this fact. Furthermore, the majority of the queries from the front-end application would center on these two main areas, and Location and Vehicle also were tied to either Incident- or Person-based information. The underlying database structure therefore was set up with two major clusters of information related to "Incident" or "Person," which together accounted for about half of the major tables used. The significance of this structure to the performance of the database can be demonstrated by examining some actual queries.

There are four main query screens, each resulting in a summary listing of information related to an initial query. Figure 2 illustrates relationships among queries. For example, if a user initiates a search on a particular first-name/last-name combination, a summary table is presented as a result of a dynamic SQL query, listing all possible matches, as well as the number of incidents associated with each individual match. From there, the user can select either a secondary listing of incidents related to a particular individual or can access a more detailed summary of the personal information on the individual. For an incident summary, all the pertinent case detail information on a particular incident is presented. For a detailed person summary, the user can select the incident summary for that individual, and from there obtain case details for any incident listed.

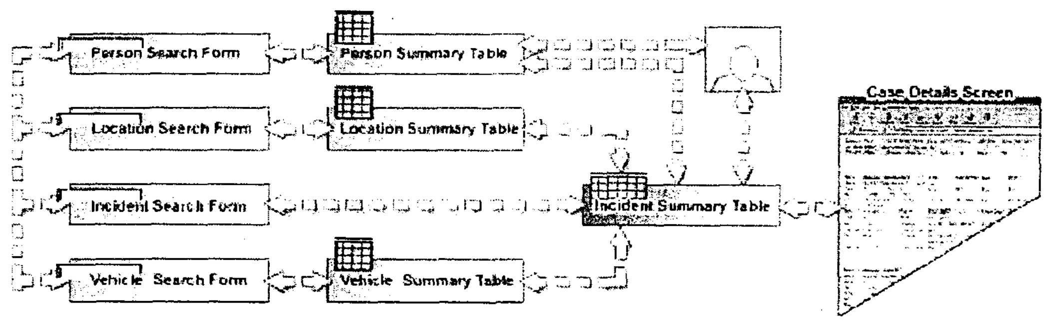



**Figure 2 - Screen Flowchart**

*An officer wanting to know more about a particular incident or person can enter a query in the search form, query further through the summary table to see details about a person, or select an incident from the incident summary table to view on the case details summary screen. In previous screens, information could be displayed in formatted rows, but a more dynamic display was needed. For example, mug shots needed to be displayed both as person details and on the case-details screen. To accommodate this feature, screens have been laid out in clusters, grouping information for easier understanding. This in turn required manipulating the data retrieved and capturing pictures from the database, a problem solved by constructing a cyclical procedure that would loop through the data and build a hierarchical tree. We could then apply display patterns to the nodes of the tree, navigate the tree and place the information on the screen.*

If the user seeks information related to an individual, the database is structured so that all related person information is read from the database at the time of the initial query. Subsequent requests for related information will not require additional disk hits, as all the related information will already be in memory, within the database buffer cache. Also, after an initial query, all subsequent information requests are based on primary key access, resulting in a very brief response time.

Other major database configuration issues that were addressed to enhance performance of the system were denormalization of some of the tables for rapid data access, and application of composite indexes for the most common queries. Many upper-level conceptual views required multiple joins within the database. For example, to obtain both physical and address information about an individual required a total of four joins. We therefore created a summary table that captures the most recent information for each individual, requiring access to only one table. User evaluations showed that the most common queries were "fine-tuned" by applying composite indexes that allowed searching on multiple columns. Since the most common query from the person query screen is a combination of last name, first name and date of birth, these three columns were combined as a single composite index . Several comparable composite indices were created.

# 6. Graphical User Interface for COPLINK DB

The graphical user interface (GUI) for the COPLINK Database Application is shown in Figures 3-7, on actual information has been altered to maintain data confidentiality. The Java front-end consists of two major parts, the input and display of data and the processing of information.

Working closely with TPD officers, the COPLINK team first made low-fidelity, paper prototypes of the screens used to obtain feedback on the display and organization of the information, which was used to modify the design and functionality of the interface. Display of results was important to the front-end. We learned that a user's idea of what constitutes a manageable and intuitive display varied with the query type and sometimes required formatting in a different way. We responded by creating a dynamic text table, using the Java API to make the interface more flexible. These figures illustrate a sample scenario in which an officer uses the COPLINK DB to search for information.

Sample Scenario: An officer is trying to identify a suspect involved in an automobile theft. A confidential informant has reported that the suspect goes by the street name of "Baby Gangster," is about 20 (probably born in 1979), and is around 5'3" tall.

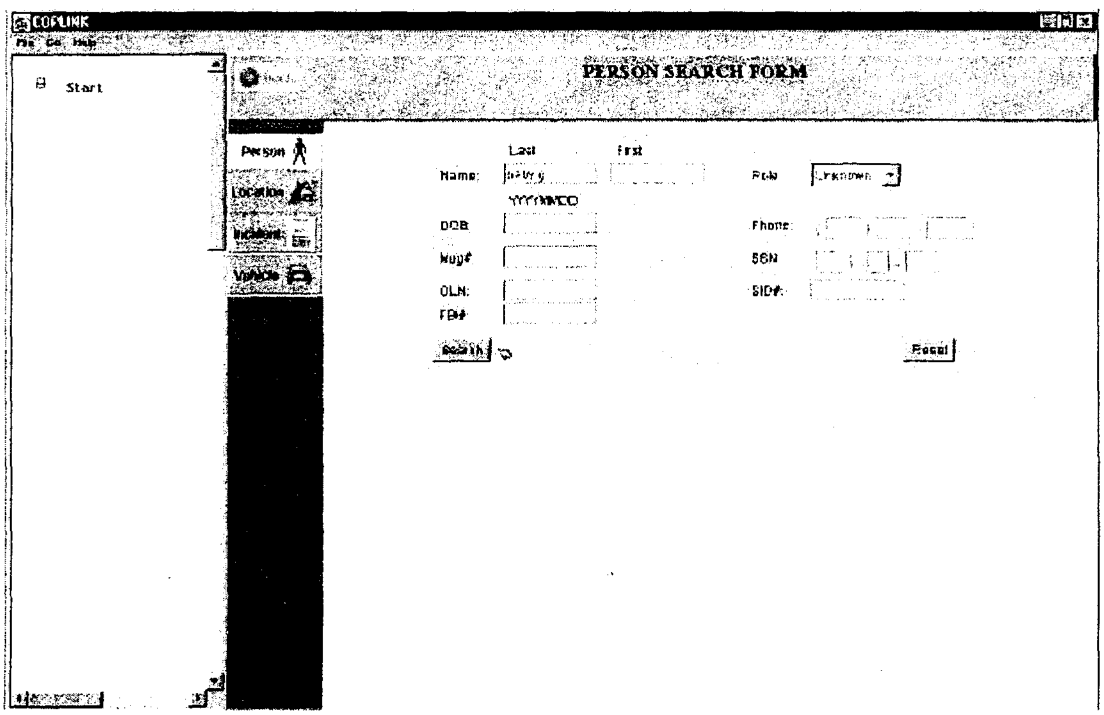



**Figure 3: COPLINK DB Search Screen.** *The officer can choose one of the four types of information upon which to search: Person, Location, Incident, or Vehicle. The officer selects the Person search screen and enters "baby g" in the Coplink DB system. Note the left panel history screen, which keeps track of the user's searches.*

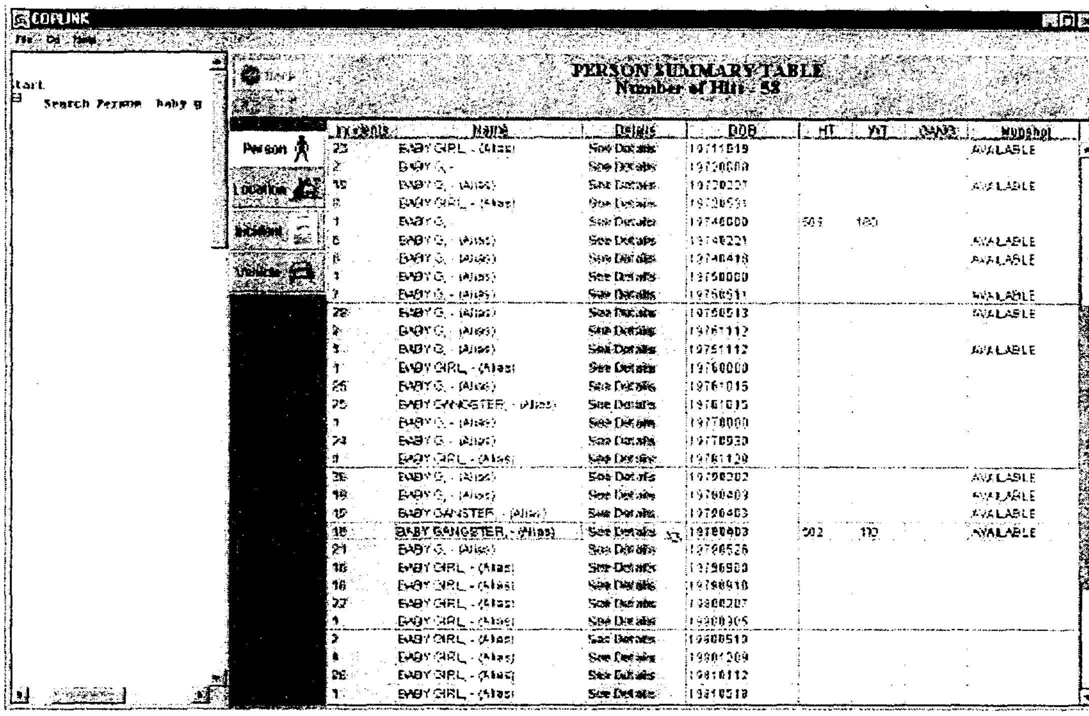



**Figure 4: Person Summary Screen**. *The system returns 58 listings referring to "baby g;" (all of the returns include the name "baby g)." The system permits sorting by any of the column headings in the table. The officer chooses to sort by date of birth and finds an entry for "baby gangster," born in 1979, whose height is 5'2". The officer then clicks on the "See Details" button to find out more about this particular "Baby Gangster".*

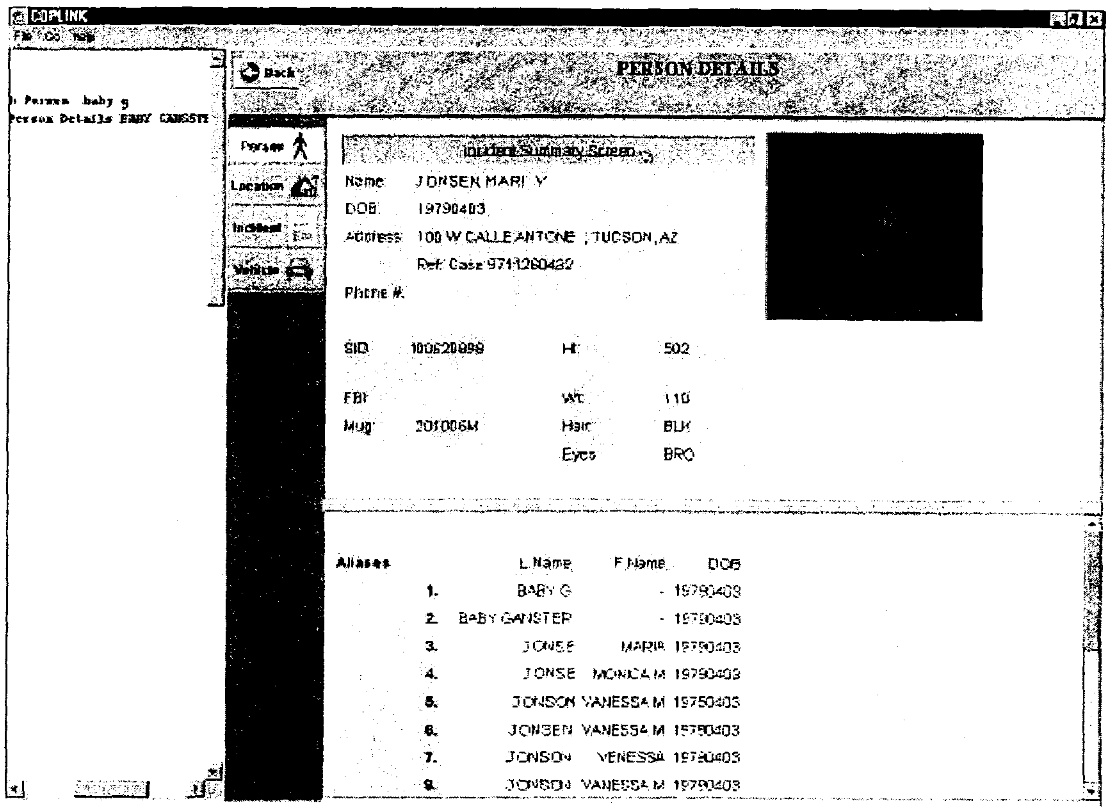



**Figure 5: Person Details Screen**. *This screen contains personal information about the selected person, including real name, latest description information, latest home address, other identifiers that the person may use, and a mug shot, if available. The officer now has a real name of a person who matches the description of the possible suspect he was given. The officer then decides to go to the incident summary screen to get an idea of the cases in which this person has been involved.*

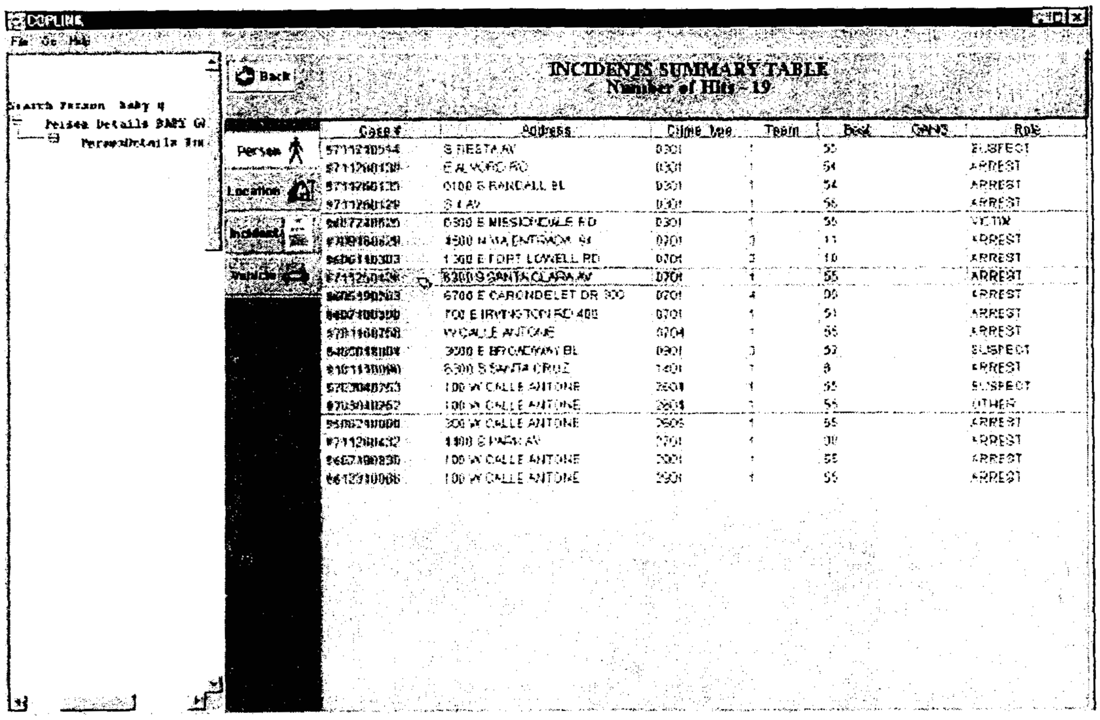



**Figure 6: Incident Summary Screen**. *This screen displays all the incidents in which the selected person has been involved. The officer sorts by crime type, looking for cases of stolen vehicles (0701) and finds the suspect has been involved in four such incidents, either as a suspect or as an arrestee. The officer selects Case #9711250126 to look at the actual case information.*

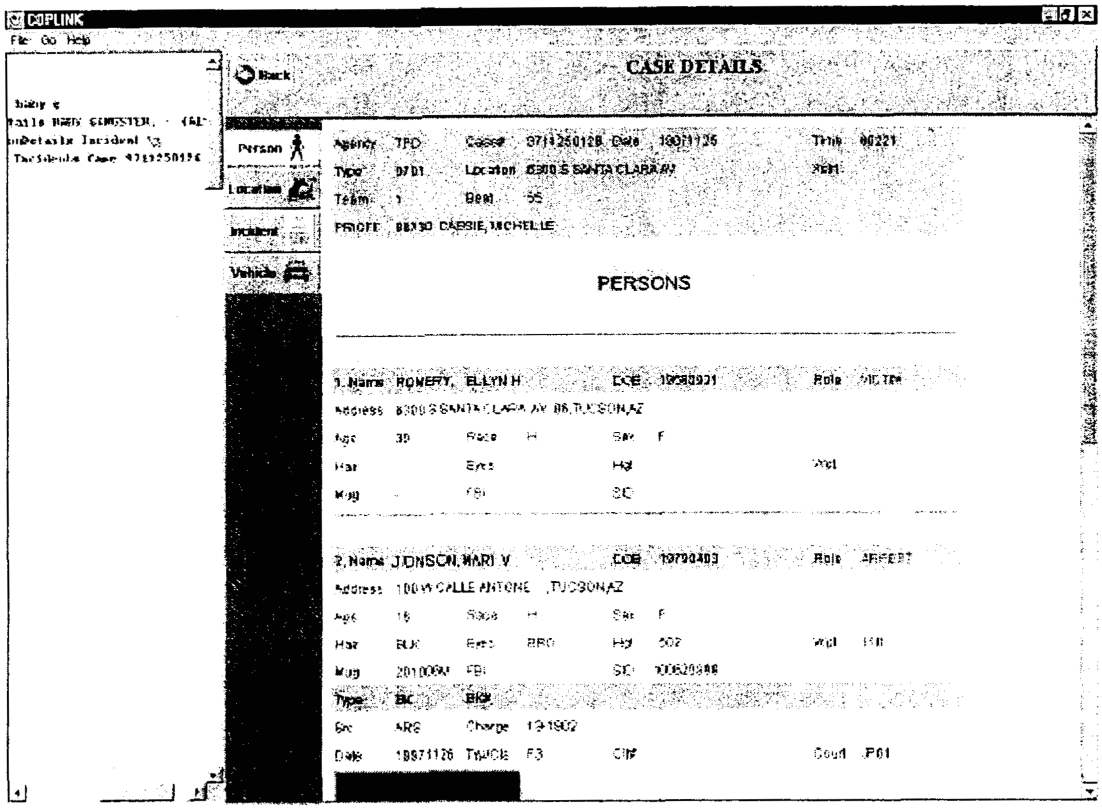



**Figure 7: Case Details**. *The case details screen provides information regarding the specific case, including location of the crime, the primary officer on the case, details about each person involved in the incident and their arresting information if applicable, and vehicles involved. The officer concludes that this person is indeed a suspect in his case and should be located for interrogation. Using the History Screen on the left panel and clicking on the Person Details to return to that page, the officer asks for a printout of the home address and a mug shot. Before finishing, the officer saves the history file, providing a log of the automobile theft case search that was conducted during this session.*

## 7. User Evaluations for the COPLINK Database Application

A usability evaluation was conducted to assess the achievement of a number of the goals that guided the design and development of the COPLINK Database. Items on the questionnaire used to assess and compare the COPLINK and RMS systems were based upon user perceptions of such widely used measures of usability as: *effectiveness* (impact of system on job performance, productivity, effectiveness of information, and information accuracy), *ease of use* (measures of effort required to complete a task, ease of learning how to use the application, ability to navigate easily through the different screens, and satisfaction with the interaction), and *efficiency* (speed of completing tasks, organization of the information on the screens, ability to find information and the interface design itself) (Hauck, 1999).

Benchmark levels from TPD's current RMS system for all three usability factors were established and compared with COPLINK DB ratings. In addition to written questionnaires, observation of the data collection methods and structured interviews were used both to supplement findings and to provide feedback for further development efforts.

A group of 52 law enforcement personnel were recruited to participate in this study. Participants represented a number of different job classifications and backgrounds (e.g., time at TPD, comfort level with computers, etc.). The data collection sequence was as follows. Initially, all subjects were asked to complete a pre-interaction questionnaire, establishing demographic background and prior level of computer experience (in general and with the current RMS system). Participants were then given a questionnaire that targeted the perceived usability of the current RMS system. After a brief introduction to the COPLINK DB application, subjects were asked to complete at least two search tasks (stating the goal of each task) using COPLINK DB. As participants accomplished these tasks, asking them to think aloud allowed us to collect process data. After a usability questionnaire on COPLINK DB had been completed, a brief interview on the COPLINK DB experience concluded the study.

Both interview data and survey-data analyses support a conclusion that use of COPLINK DB provided improved performance over use of the current RMS system. On all usability measures (effectiveness, ease of use, and efficiency), participants rated COPLINK DB higher than RMS, with the average rating for COPLINK being 4.1 and RMS being 3.3 (1=strongly disagree to 5=strongly agree). Statistical analyses revealed that this ratings difference was significant for all measures.

In addition to the statistical data, these findings are supported by qualitative data collected from participant interviews. Comments collected from interviews indicate that COPLINK DB was rated higher than RMS in terms of interface design and performance as well as functionality. The general themes that emerged from the interviews also can be categorized into factors of speed, ease of use, interface, and information quality.

Participants indicated that the quality and quantity of information from COPLINK DB surpassed those of RMS. In a review of current RMS practices, a number of detectives and officers were actually unable to use RMS but were able to use COPLINK DB to conduct searches. It is evident from this research study that COPLINK DB allowed a population of TPD personnel to access information that would have been quite difficult for them to have acquired using the RMS system. From both the questionnaire and the interview data collected from this evaluation, it is evident that many participants rated the information found in COPLINK as more useful than the information in RMS. This finding is very interesting, because most of the information contained in COPLINK has been taken from RMS.

COPLINK's ability to allow the user to structure his/her query results by selections from a number of fields is an important strength of the system. Being able to sort query areas allows users to organize the results meaningfully in the context of a specific search task. Cases are organized in RMS by date. COPLINK DB, on the other hand, allows users not only to organize by date but also to sort by crime type or even team and beat. Patrol officers who participated in the study indicated that the availability of COPLINK DB at substations (within their individual areas) or in patrol cars would greatly improve on-the-street access to information needs that are currently unmet. In particular, they stressed the importance of being able to use mug shots to determine identity quickly. One patrol officer related an incident in which he apprehended a suspect he believed to be wanted for prior criminal activity. Using RMS, the only way the officer could verify the identity of the suspect was to take the person physically to downtown headquarters and have the identification office check his fingerprints. The patrol officer indicated that had he had COPLINK DB, either in the patrol car or at one of the local substations, he could quickly and easily have verified the person's identity by checking mug shots on file as well as current case information on the 'wanted' person.

During the user evaluation process, we also looked for the application of COPLINK DB to real-life crimes. An example is the real-life case of a hit-and-run and possible homicide case reported to us by Tucson Police Department's Officer Linda Ridgeway:

*"The Tucson Police Department had responded to a shooting at a local establishment shortly after the business closed. Witnesses reported that after the suspect shot the victim, he got into a small white newer vehicle with 4 doors, about the size of a Dodge Neon.*

*A short time later and at a location relatively close to this business a complainant called to report that someone had just hit his vehicle and left the scene. He reported that the vehicle was a newer looking small white vehicle. He was also able to supply us with a partial license plate number.*

*Thinking that this might be the suspect vehicle from the shooting I ran the partial plate through the COPLINK Database. I included the partial plate and a white, four-door vehicle data. Within approximately 20 seconds I had a list of possible vehicles that matched this description. I found a listing for a Dodge Neon that was a suspect in a prior case, so I forwarded the complete plate number of the suspect vehicle to Investigators.*

*Investigators went to the residence of this vehicle's owner and found that the car did in fact have paint transfer and damage that was consistent with the damage on the victim vehicle. Although the driver admitted to the accident and was charged with the hit-and-run, the driver was also ruled out as a suspect in the shooting, whom the investigators caught at a later date. Without Coplink, we would not have been able to investigate this lead or have been able to identify the hit-and-run suspect."*

Currently we are completing the final run of user-stress testing to validate the most recent update of the interface in preparation for deployment of the Coplink DB system to a limited number of detectives, crime analysts, and officers. Full-wired deployment of the Coplink DB system is projected by the end of summer 2000.

## 8. Future Directions for COPLINK

Large collections of unstructured text as well as structured case-report information exist in police records systems. These textual sources contain rich sources of information for investigators that are often not captured in the structured fields. One of our future research directions is to explore the development of textual mining approaches that support knowledge retrieval from such sources for law enforcement. In order to perform a fine-grained analysis for law enforcement content, we will be investigating the development of linguistic analysis and textual mining techniques that make intelligent use of large textual collections in police databases.

Several Internet research projects have shown the power of a new "agent" based search paradigm. In addition to supporting conventional searches performed by users, search agents allow users automatically to establish search profiles (or create profiles for users) and extract, summarize, and present timely information content. We believe such a proactive search agent is well suited to use by investigative personnel in law enforcement agencies. Search agents for law enforcement can support conventional searching techniques, and be profiled for specific investigations. We plan to develop a personalized law enforcement search agent that will support wide expansion in connectivity and information sharing between police agencies.

In relation to the COPLINK project, the concept of a distributed database system has important implications. The most important of these is accessibility to and dissemination of law enforcement records and information. Currently, the vast majority of criminal data collection and compilation is done on a community level but may not be in a format that is readily available and accessible to local law enforcement officers. A distributed COPLINK prototype is under development using three COPLINK database servers to simulate the independent nodes in a distributed environment. Work is under way to include functionality that will provide interoperability among the different DBMS platforms, which may support future COPLINK nodes. In the immediate future, we plan to begin deployment and testing of a Distributed COPLINK prototype with the Tucson and Phoenix police departments.

As distributed solutions and analysis tools are developed for law enforcement officers, a specific focus must be on providing tools within the constraints of a wireless environment. One of our future goals is to develop and refine applications to support the expansion of distributed and mobile law enforcement networks and inter-jurisdictional information retrieval as well as to investigate and study network security issues.

**Acknowledgments**

This project has primarily been funded by a grant from the National Institute of Justice, Office of Science and Technology (Tucson Police Department, "Coplink: Database Integration and Access for a Law Enforcement Intranet", $1.2M, 1997-1999). We would also like to thank the Digital Equipment Corporation External Technology Grants Program, agreement #US-1998004, for its award of a $198,451 equipment grant allowance toward the purchase of a DecAlpha Server for the COPLINK Project. We would like to thank Sergeant Jennifer Schroeder and Officer Linda Ridgeway and all the other personnel from the Tucson Police Dept. as well as the personnel from the Phoenix Police Dept.

## Biographies

**Roslin V. Hauck** is currently pursuing her doctorate in Management Information Systems at the University of Arizona and has been affiliated with the department since 1997. She received her B.S. in Communication Studies from Northwestern University in 1995 and her Masters in Communication at the University of Arizona in 1997. She has presented her research at the International Conference in Information Systems. Her research interests include technology adoption and organizational behavior, human-computer interaction, information visualization, usability, and research design. She can be reached at rrv@bpaosf.bpa.arizona.edu.

**Dr. Homa Atabakhsh** is Principal Research Specialist at the University of Arizona MIS Department and is the project manager for COPLINK. She is also adjunct lecturer and is teaching a course in Advanced Object Oriented Programming. She received her BSc., MSc.(1984), and Ph.D. (Dec. 1987) degrees in Computer Science from the University of Toulouse in France. She was employed by the National Research Council of Canada, as Research Scientist, from Jan. 1989 - 1996 and worked in areas such as knowledge based systems, object-oriented design and programming, GUI, applications in Manufacturing and Business. She can be reached at homa@bpa.arizona.edu.

**Harsh Gupta** earned his Masters of Science in Management Information System in 2000 from the University of Arizona specializing in integrated large database systems, data warehousing and data mining. He is also a former member of the AI Lab serving as a systems engineer for the Coplink Project. His research focuses on data migration issues and techniques for large data warehouses. He is currently working as product manager for Knowledge Computing Corporation, a Tucson based startup. He can be reached at gupta_harsh@hotmail.com.

**Chris Boarman** recently earned his Masters of Science degree in Management Information Systems in 2000 from the University of Arizona specializing in distributed systems. He is also a former member of the AI Lab serving as a systems engineer for the Coplink Project. His research focused on interoperability solutions for heterogeneous multidatabase systems. He currently is working for Lockheed Martin as a Systems Integrator implementing air traffic control systems. He can be reached at christopherboarman@mail.com.

**Kevin Rasmussen** completed a Masters of Science degree in Management Information Systems in May, 1999 at the University of Arizona where he was part of the COPLINK development team. He is currently a Programmer/Analyst at Intel Corporation as a member of the Employee Services Information Systems organization. He is continuing his specialization in database-related systems development in a web-enabled PeopleSoft environment. He can be reached at kevin.rasmussen@intel.com.

**Andy W. Clements** is currently a Programmer/Analyst for Intel's Finance Information Systems group, where he works on Web portal products and reusable Java frameworks. He graduated from the University of Arizona with a Master of Science in Management Information Systems in May 1999. While pursuing his graduate degree he worked in the UA/MIS Artificial Intelligence Lab concentrating on Data Warehousing, Internet Agents for Information Retrieval, and Geographic Information Systems. He can be reached at andy.w.clements@intel.com.

**Dr. Hsinchun Chen** is McClelland Professor of Management Information Systems at the University of Arizona and head of the MIS department's Artificial Intelligence Lab. He received the Ph.D. degree in Information Systems from New York University in 1989. He is author of more than 70 articles covering semantic retrieval, search algorithms, knowledge discovery, and collaborative computing in leading information technology publications. He serves on the editorial board of Journal of the American Society for Information Science and Decision Support Systems. He is an expert in digital library and knowledge management research, whose work has been featured in various scientific and information technologies publications including Science, Business Week, NCSA Access Magazine, WEBster, and HPCWire. He can be reached at hchen@bpa.arizona.edu.

## References


Blair, D. C., & Maron, M. E. (1985). An evaluation of retrieval effectiveness for a full-text document-retrieval system. *Communications of the ACM* 28(3): 289-299.

Bowen, J. E. (1994). An expert system for police investigators of economic crimes. *Expert Systems with Applications* 7(2): 235-248.

Brahan, J. W., Lam, K. P., Chan, H., and Leung, W. (1998). AICAMS: Artificial intelligence crime analysis and management system. *Knowledge-Based Systems* 11: 355-361.

Chen, H. (1995). Machine learning for information retrieval: Neural networks, symbolic learning, and genetic algorithms. *Journal of the American Society for Information Science* 46(3): 194-216.

Chen, H. & Ng, T. (1995) An algorithmic approach to concept exploration in a large knowledge network (automatic thesaurus consultation): Symbolic branch-and-bound search vs. connectionist Hopfield net activation. *Journal of the American Society for Information Science*, 46(5): 348-369.

Chen H., Schatz, B., Ng, T., Martinez, J., Kirchhoff, A., & Lin, C. (1996). A parallel computing approach to creating engineering concept spaces for semantic retrieval: The Illinois Digital Library Initiative Project. *IEEE Transactions on Pattern Analysis and Machine Intelligence Special Section on Digital Libraries: Representation and Retrieval* 18(8): 771-782.

Davenport, T. H. (1995). Business process reengineering: Where it's been, where it's going. In V. Grover & W. Kettinger (Eds.), Business Process Change: Reengineering Concepts, Methods and Technologies (pp. 1-13). Middletown, PA: Idea Publishing.

Fazlollahi, B. & Gordon, J. S. (1993). CATCH: Computer assisted tracking of criminal histories system. *Interfaces* 23(2): 51-62.

Hauck, R. V. (1999). COPLINK: Exploring usability of a multimedia database application for law enforcement. Report.
Available: http://ai.bpa.arizona.edu/go/datawarehousing/publications/nij.pdf

Hauck, R. V. & Chen, H. (1999). COPLINK: A case of intelligent analysis and knowledge management. *Proceedings of the 20th Annual International Conference on Information Systems '99*, 15-28.

Hoogeveen, M. J. & van der Meer, K. (1994). Integration of information retrieval and database management in support of multimedia police work. *Journal of Information Science* 20(2): 79-87.

Lewis, C. (1993). Police information technology. *GEC Review* 9(1): 51-58.

Lingerfelt, J. (1997). Technology as a force multiplier. *Proceedings of the Conference in Technology Community Policing.* National Law Enforcement and Corrections Technology Center. Available: http://www.nlectc.org/txtfiles/confrpt.txt

Miller, B. (1996). Searchable databases help Missouri solve crime. *Government Technology* 9(8): 18-19.

O'Leary, D.E. (1998). Enterprise knowledge management. *IEEE Computer* 31(3): 54-62.

Pliant, L. (1996). High-technology solutions. *The Police Chief* 5(38): 38-51.

Schellenberg, K. (1997). Police information systems, information practices and individual privacy. *Canadian Public Policy - Analyse de Politiques* 23(1): 23-39.

United States Department of Justice, Uniform Crime Reporting: National Incident-Based Reporting System: Vol. 1 Data Collection Guidelines, 1998.

Wilcox, J. (1997). IT-armed officers fight gangs. Government Computer News: State and Local 3(12): 19-20.

# Attachment B

# Multi-Node Coplink: the Design of a Distributed Incident-Based Query System

Kevin Rasmussen

Department of Management Information Systems

College of Business and Public Administration

The University of Arizona

May, 1999

# Table of Contents

## List of Figures

# List of Tables

## Acknowledgements

I would like to thank Dr. Kevin Lynch for his assistance not only with this project, but also for his assistance in the other aspects of the Coplink database design, administration and analysis I've been fortunate to participate in over the past two years.

A special thank you is also owed to Andy Clements for altering the Coplink front-end application to function in a distributed environment. Also, the input and discussions from all members of the Coplink team have been critical to the development of this project. Specifically, the outstanding officers from the Tucson Police Department: Sergeant Brad Cochran, Sergeant Jennifer Schroeder, and officer Linda Ridgeway for their "real-world" input and enthusiasm for the entire Coplink project. Also, the other members of the Coplink team who have been a pleasure to work with and whose insights and expertise benefited my efforts on innumerable occasions: Hend Dwiyono, Harsh Gupta, Rosie Hauck, Gondy Leroy, and Harry Li.

Finally, a special thank you is offered to Dr. Hsinchun Chen for affording me the opportunity to be associated with the Artificial Intelligence Lab of the MIS Department at the University of Arizona. My involvement with the outstanding members of the "AI Lab" has been the highlight of my experience at the University of Arizona.

# Introduction and Motivation

The Coplink project was originally conceived as a project "to integrate law enforcement databases in a single web-based interface and to support intelligent analysis" of incident-based law enforcement information.[11] The project has successfully developed an integrated, multimedia database system, with a web-based front-end application. The Coplink application is currently undergoing testing and analysis at the Tucson Police Department (TPD). The scope of the project, then, has thus far been to successfully complete a single, community-based law enforcement agency implementation at TPD.

Because of the success to date of the Coplink project, interest has been generated with other law enforcement agencies that are investigating the possibility of implementing mirror implementations of the Coplink integrated system. Specifically, initial discussions and funding proposals are currently underway with the Phoenix Police Department and related agencies in Maricopa County, Arizona.

The purpose of this project, then, was to develop an initial system design and prototype that would allow multiple agencies with similar implementations of the Coplink application to share information between each agency.

Based on the initial design of a single Coplink "node," an expanded distributed system of multiple Coplink "nodes" was developed. From a functionality standpoint, this would allow any law enforcement agency that adheres to the uniform design of a Coplink node to share and access incident-base law enforcement information from any other participating agency. The system includes a "distributed" version of the Coplink application, a modified set of queries that automatically query multiple Coplink nodes, and the underlying system architecture to support a distributed Coplink system.

This design proposal includes several artifacts that are part of the Unified Modeling Language (UML), an emerging standard notation for object-oriented modeling.[1]

## Literature Review

A distributed database system "is an environment in which data in two or more database instances is accessible as though this data were in a single instance." [2] This implies communication between two or more databases physically separated on separate server nodes. Implicit in this concept of a distributed database system is an underlying network through which the physically separated databases communicate.

Until recently, this type of database access was most typically set up to occur over a two-tier architecture in which the client is connected directly to the distributed databases via a front-end application. From the user's perspective the fact that more than a single database instance is being accessed is usually not apparent. Front-end applications generate SQL queries that are executed against the database backend. [3] In this scenario, the front-end application usually hides the complexity of querying each database separately, combining the results for display to the user.

There are several reasons why implementing a distributed database system would be advantageous to an organization[2, 4, 5]:

1. Data can be strategically located near demand. For example, a company with a large sales office in one geographic location would want to maintain the bulk of that division's sales information near or at the specific location. Smaller subsets of sales information may be distributed to other smaller sales sites.
2. Functional division of data. Databases can be distributed along divisional or departmental lines, providing such groups with the information necessary for their specific tasks.
3. Division of data processing tasks. Large data processing tasks may be accomplished faster if the data sets are evenly distributed across multiple databases.
4. Less danger of single-point failure. Companies with mission-critical applications that are dependent on large database back-ends could set up multiple copies of the data to ensure that if one database instance fails, others are available to service the company's business needs.
5. Processor independence. Systems can be split among multiple nodes of varying levels of processing power.

6. Horizontal scalability of high-demand data. For example, a busy website may have multiple copies of the same data set in order to service incoming requests for data.
7. Vertical segmentation of data. For example, decision-support personnel are interested in compiling and analyzing aggregated data in a data warehouse, where sales personnel are interested in tracking information in a transactional system.
8. Decentralization of IS functions. Organizations with many regional offices can "split up" information systems functionality, providing more localized or regional business support.
9. Accessibility. One location "owns" information that other locations want access to.

In addition to database instances, a distributed database system can also be described by and include application servers and clients.[2] For example, with the increasing emphasis on "content" in large website applications, this fact is becoming more apparent, as web-enabled database systems move away from a client-server (2-tier) modality to three-tier architectures. Typical of a three-tier architecture are a "thin" web-based client, a web application server, and a database server. This is increasingly becoming the norm in web-enabling distributed database systems. [3, 6]

Much discussion has been paid to issues of complexity inherent in distributed database systems[2, 4, 7-9]:

1. Availability and fault tolerance. This includes discussions of mechanisms for assuring availability of systems, and maintaining the integrity and readiness of a distributed system.
2. Complexity of management and control. These include issues of concurrency control, security, backup and recovery, query optimization, and performance tuning.
3. Security. Maintaining security across multiple distributed databases increases in complexity as the number of nodes increases, and as access to nodes proliferates across a system.
4. Transaction processing and rollback. These issues relate to data integrity and transaction assurance. If a transaction fails, mechanisms must be in place to restore the dataset to its original state. This increases in complexity across a distributed system, because only a portion of a distributed transaction may fail.

5. Parallel queries across N distributed databases. This refers to the division of complicated or long-running queries or updates across multiple databases. For example, a large inventory database that is partitioned across multiple databases can be updated simultaneously across all instances, reducing the total time of the update.
6. Advanced Replication. Copies of a database are replicated across multiple database instances. Changes to one database are duplicated on all replicated instances.
7. Snapshots. This refers to the concept of maintaining local copies of a remote database. Snapshots can be set up as read-only versions of a remote database (or subset of the remote database), and are automatically updated periodically over a distributed system.

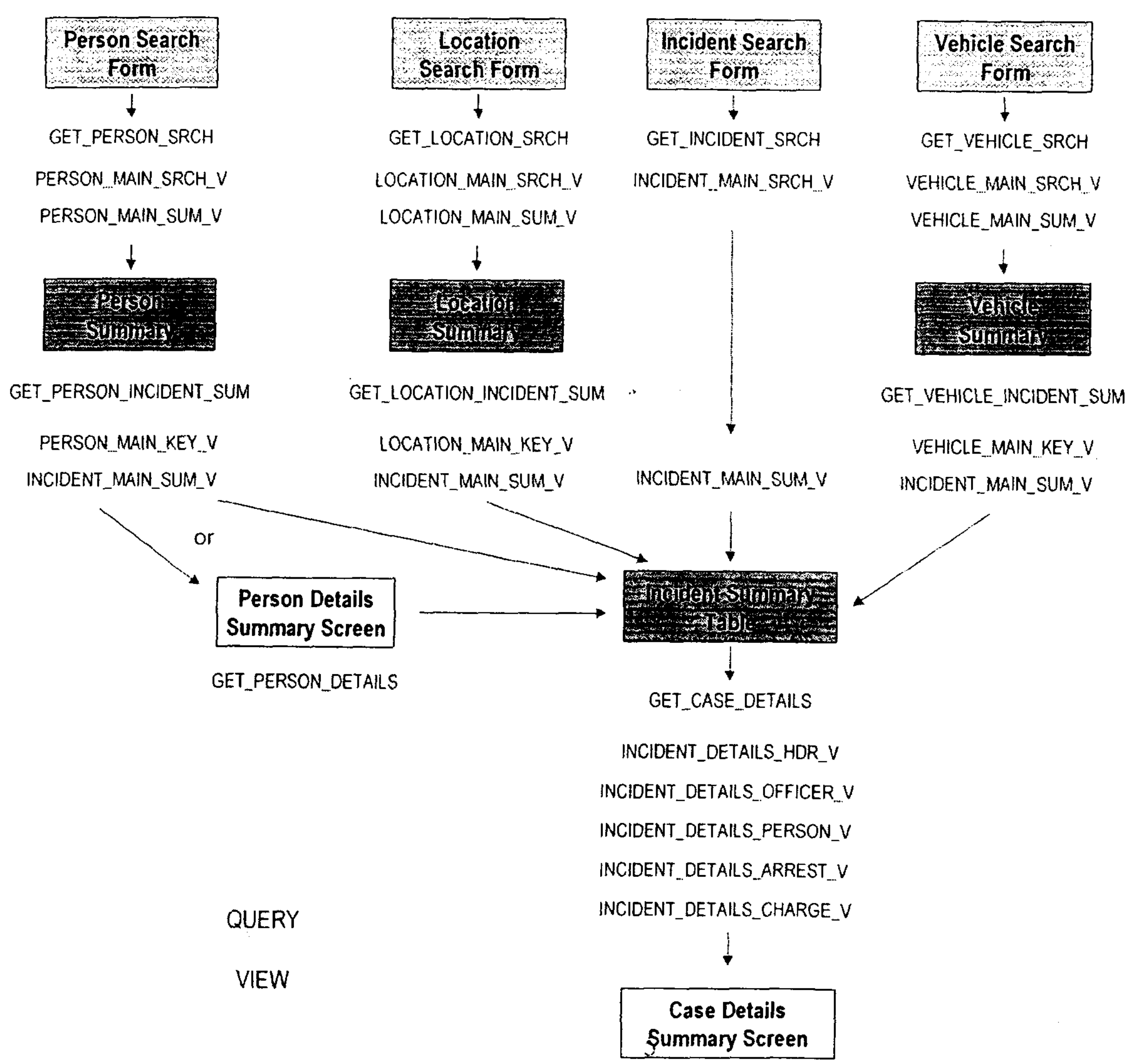
In relation to the Coplink project, the concept of a distributed database system has important implications. The most important of these is accessibility to and dissemination of law enforcement records and information. Currently, the vast majority of criminal data collection and compilation is done on a community level. However, this data collection is not always in a format that is readily available and accessible to local law enforcement officers. [Brahan, 1998 #20] The Coplink project has successfully integrated disparate database systems into an integrated, three-tier, query-based system in which incident-based information is readily available to the user. [10] However, this success has occurred only in a single community, and access to that integrated information is limited to Tucson Police Department personnel.

The long-term goal of a distributed Coplink system is to make incident-based data from multiple local agencies in a regional area available to every other agency within that same area. The impact of such a system could be a measurably significant increase in the success of law enforcement activities on a regional basis. This is the promise of a distributed Coplink system.

## Overview of a single Coplink Node

Although the data currently integrated into the Coplink project is specific to TPD, the design of the Coplink database structure has been executed such that it should be applicable to other similar law enforcement agencies. Key to this is the setup of standardized "Views" that enable underlying tables, no matter their structure, to be mapped to higher-level views for standard query and data extraction. The PL/SQL-based queries utilize the standardized views to perform the front-end application requests. The following chart illustrates the relationship between each screen in the application front-end, the queries executed to populate the result screens, and the underlying views from which the data is accessed.

**Figure 1 Coplink User Screens, Queries and Views**

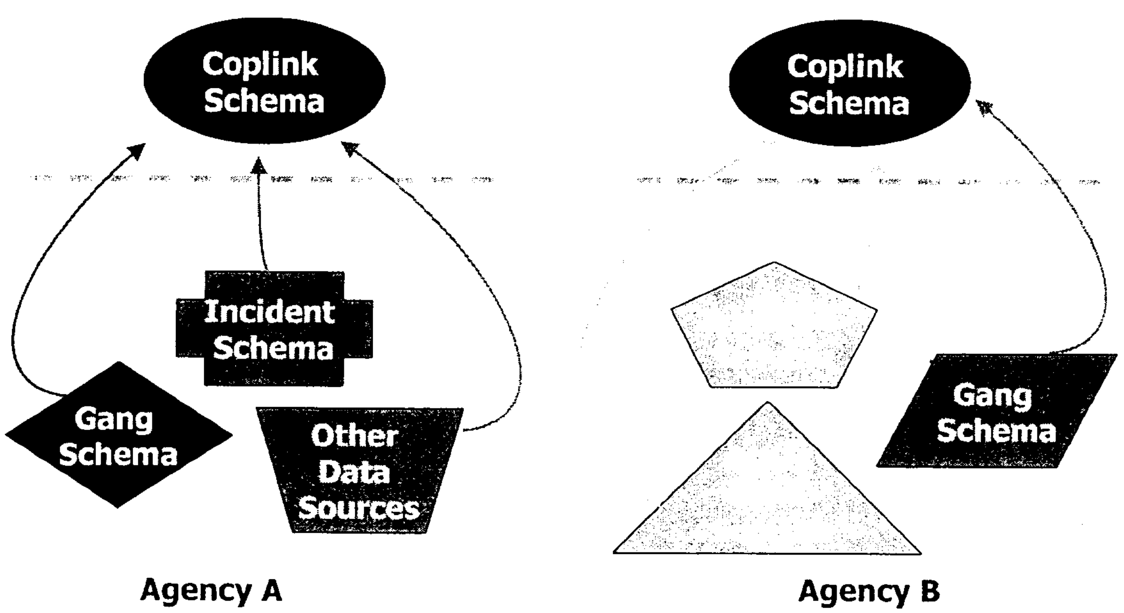


## Standardized Views

It is an obvious statement that no two law enforcement agencies will have exactly the same system for tracking incident-based information. Although there is an effort to standardize incident-based reporting systems on a national and regional basis, [11] the reality is that most agencies are similar to TPD in that they have a disparate set of information stores, performing varying data collection and dissemination functions.

One of the goals, then, for the initial design of the Coplink system, was to create a mechanism that would allow for a standardized set of data points, organized in logical groupings as "Views." Each view would allow one or more underlying database tables to be "mapped" to the required data points for that view. In this way, standardized Coplink nodes could be developed, no matter the structure of the underlying database (or databases). As the following diagram illustrates, the goal of the standardized views is to allow disparate underlying data sources to be mapped to a standardized Coplink schema. Hereafter, a single Coplink instance will be referred to as a "Coplink Node."

**Figure 2 Standardized Coplink Schema and Views**



## Coplink Query Procedures

The nine main procedures used by the Java front end are the following nine queries. The initial group (GET_*[identifier]*_SRCH) are representative of the five main query (or search) screens (Person, Location, Incident, Weapon, and Vehicle). As shown in Figure 1, from the initial query screen, the subsequent queries are the incident summary queries (GET_*[identifier]*_INCIDENT_SUM) which provide a listing of related incidents based on the initial search. Finally, the GET_CASE_DETAILS and GET_PERSON_DETAILS procedures provide detailed information about an individual person, or an individual case. Access to all data, except from the initial query screen, is based on primary key access to the database, making subsequent queries for incident, person details, and case detail information very efficient.

**Table 1 Coplink Stored Procedures**

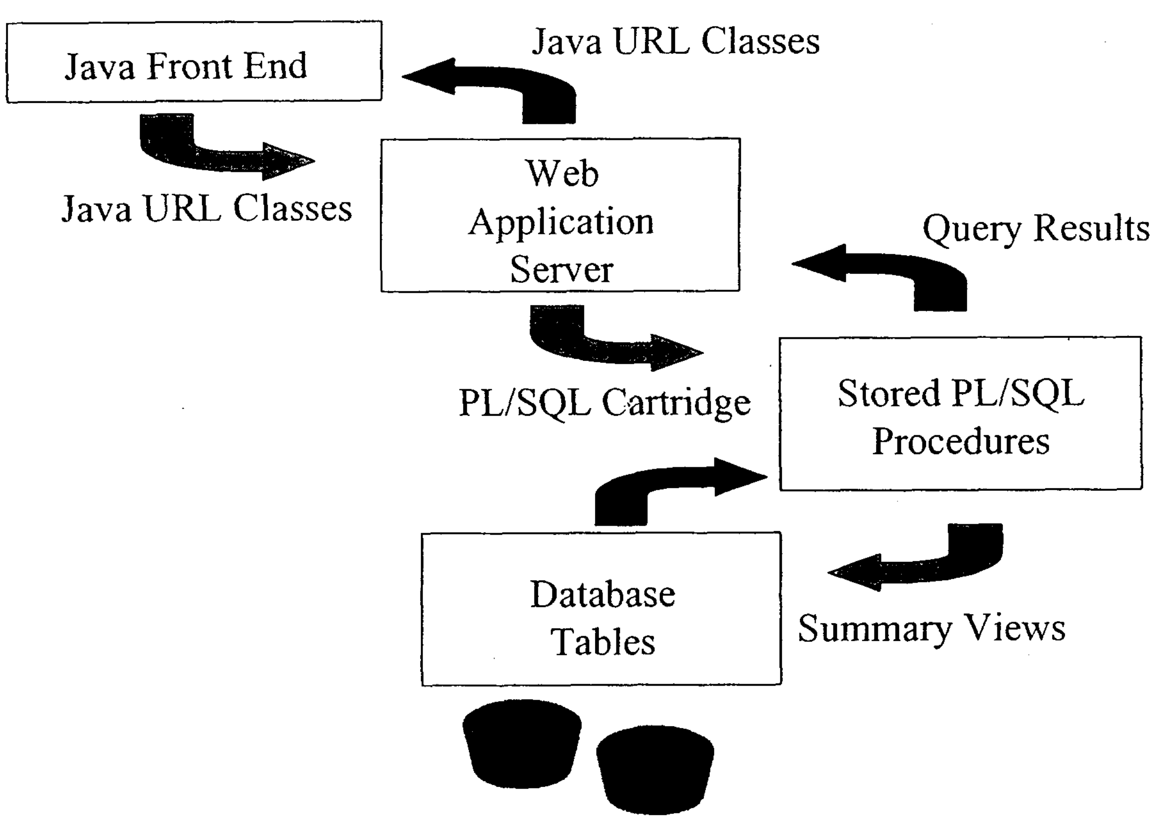
| Procedure | Description |
| --- | --- |
| GET_PERSON_SRCH | Called from: the initial Person Search screen.<br>Result: The Person Summary screen, listing all persons that match the input search criteria. |
| GET_LOCATION_SRCH | Called from: the initial Location Search screen. Result: The Location Summary screen, listing all locations that match the input search criteria. |
| GET_INCIDENT_SRCH | Called from: the initial Incident Search screen. Result: The Incident Summary screen, listing all incidents that match the input search criteria. |
| GET_VEHICLE_SRCH | Called from: the initial Vehicle Search screen. Result: The Vehicle Summary screen, listing all vehicles that match the input search criteria. |
| GET_PERSON_INCIDENT_SUM | Called from: the Person Summary screen.<br>Result: the Person Incident Summary screen, listing all incidents associated with the chosen individual. |
| GET_LOCATION_INCIDENT_SUM | Called from: the Location Summary screen.<br>Result: the Location Incident Summary screen, listing all incidents associated with the chosen location. |
| GET_VEHICLE_INCIDENT_SUM | Called from: the Vehicle Summary screen.<br>Result: the Vehicle Incident Summary screen, listing all incidents associated with the chosen vehicle. |

| | |
|---|---|
| GET_CASE_DETAILS | Called from: any Incidents Summary screen<br>Result: The Case Details screen, listing all the details associated with a particular case. |
| GET_PERSON_DETAILS | Called from: The Person Summary screen.<br>Result: The Person Details screen, listing all the known information about the person, including Aliases |

## Three Tier Architecture of a Single Coplink Node

The Coplink application utilizes a three-tier architecture as illustrated in the following diagram:

**Figure 3 Single Node Coplink Three-tier Architecture**



The front-end application sends a URL-based query to an Application Server, which processes requests from multiple users. The application server maintains connections to the local node database, and executes stored procedures in the database. Utilizing HTTP-based protocol packages in the database, the results of the executed query are sent back to the front end, parsed, and displayed in a meaningful format.

Another important point to emphasize about the Coplink system is that it is a read-only query based system. Any ongoing incident-based data collection and updates occur in the database systems that "feed" the Coplink system. This point becomes more critical in a distributed system as "local" Coplink nodes are made available to other law enforcement agencies.

In summary, the three-tier architecture of a single Coplink node can be successfully scaled and applied to a distributed Coplink system. In fact, the elegance of this three-tier architecture becomes more apparent in a distributed system.

## Requirements for the Distributed Coplink System

A critical underlying assumption for a distributed Coplink system is that each Coplink node adhere to the standardized Coplink Schema and Views structure. This ensures that all the distributed versions of the queries utilize naming conventions that are the same across all Coplink nodes. If all object names and identifiers are standardized—including not only table names, but also column or data point names and other RDBMS objects (as will be discussed shortly), then it becomes much more feasible to design an interrelated system that is consistent across all nodes.

This requirement enforces the following system attributes and advantages:

1. Scalability: Additional nodes can easily be added to the distributed Coplink system because additional complexity in integrating differing, non-standardized implementations is avoided.
2. Code Reusability: Stored procedures, database creation scripts, configuration scripts, and other logical sets of code can be utilized across different nodes without extensive software customization.
3. Elimination of Ambiguity: If a particular identifier (e.g. a datapoint column name, or a view name) is the same across all nodes, confusion or ambiguity in definition is avoided.
4. Maintenance: As nodes are added to the system, it becomes a much simpler task to update the other nodes, essentially "informing" them of the availability of the other nodes.

In general, from the front-end application, users should be able to:

1. Be presented with a listing of available Coplink "nodes" upon system startup. The user can select from one or more nodes, depending on the nature of the query. If a node is usually available to the user, but currently not available, then the node should be presented in name only and not be selectable by the user.
2. The system should automatically recognize and default to the "Local" node for a given session. For example, if the officer is a TPD officer, the local node is obviously Tucson.

3. The user should be able to query 1 to N Coplink Nodes for each of the initial Query Screens: Person, Location, Incident, Vehicle, and Weapon. For example, given a Person Search for "Eddie Garcia", the user would select one or more of the available nodes to search from. A typical scenario may be searching for information from both Tucson and Phoenix Police Departments.
4. Based on the return from an initial query screen, the user should easily be able to determine the source of the information, based on a uniform agency descriptor or code on the incident summary screens. The same is true for the Person Details and Case Details screens.
5. Users should be able to track the history of their queries, which should include the node(s) selected for each query.

From a conceptual standpoint, the system should exhibit the following qualities:

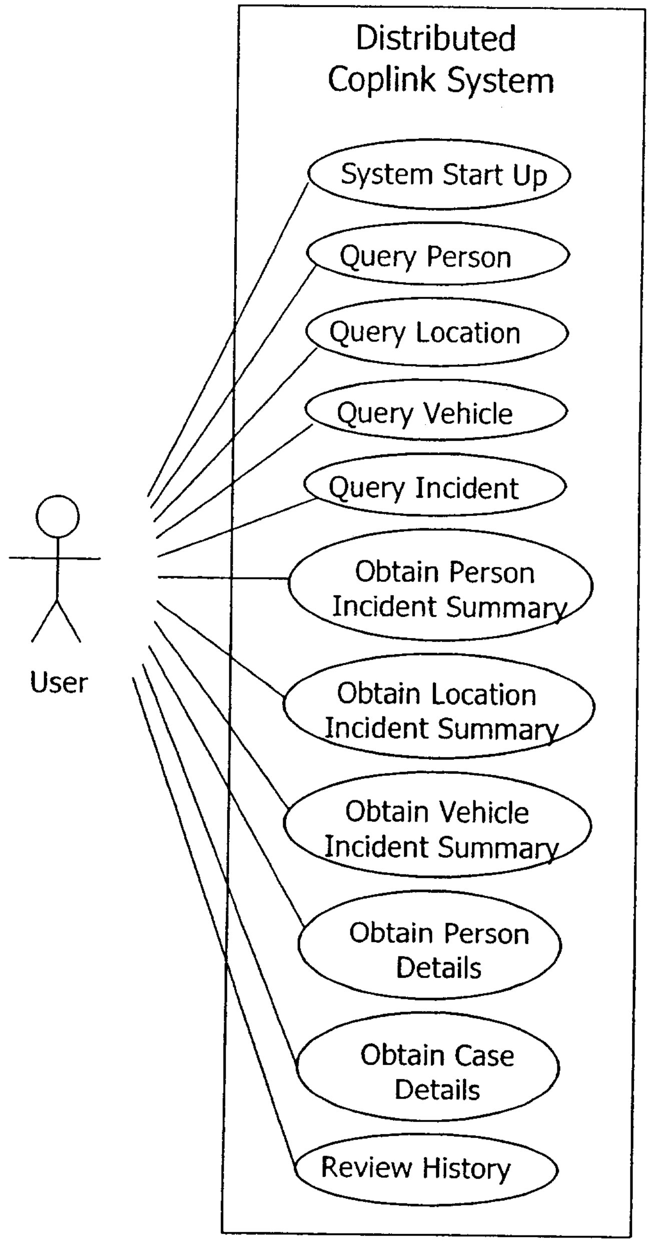
1. The fact that the information is being queried across multiple databases should be transparent to the user. The intricacies of how the query is accomplished should not impact nor concern the end user.
2. The responsibility for determining the availability of nodes should happen transparently to the user. The process should not be a manual one.
3. The front-end application should be universal to all nodes. Specifically, there should not have to be any modification to the front-end tailored to individual nodes. The front-end application should be loosely coupled with the back end.[1]
4. Each Coplink node should adhere to the "read-only" concept. In other words, users should not be able to change or update data in any Coplink node
5. The system should be able to recover if a particular Coplink node goes down during a user session.

In general, the look and feel of the front-end application should be almost identical to the single node version. In fact, as the Coplink system is scaled and deployed, the single "distributed" version of the Coplink application could and should be the only version of the application. At application startup, the user may simply be asked if they would like to query just the local or multiple Coplink nodes. In the case of the former choice, the distributed Coplink nodes would not be listed, and all queries would simply default to the local Coplink node.

## Distributed System Use Cases

The following is a Use Case Diagram for the distributed Coplink system. Four sample Use Cases (the Person-related searches) used in the system prototype accompany the diagram. Each use case is assumed to be a multi-node version. Use Cases are artifacts generated as part of the UML design phase and describe system events in clear, simple steps.[12]

**Figure 4 Distributed Coplink Use Cases Diagram**



## Use Case: System Start Up (Multi-node)

| | |
|---|---|
| Use Case: | System Startup |
| Actors: | User |
| Purpose: | Initialize the Distributed Coplink Front-end Application |
| Overview: | User initiates the Distributed Coplink application. The user is presented with a listing of available Coplink nodes and initial query screens |
| Type: | Essential |
| Cross References: | |

**Typical Course of Events**

| Actor Action | System Response |
|---|---|
| 1. The user initiates the Distributed Coplink application | 2. The Distributed Coplink application launches. |
| | 3. The system queries the local database to determine which Coplink nodes are available. |
| | 4. The system presents the user with a listing of available nodes on each initial query screen. |

## Use Case: Query Person (Multi-node)

| | |
|---|---|
| Use Case: | Query Person |
| Actors: | User |
| Purpose: | Find person matches on demographic information. |
| Overview: | User enters partial demographic information. A search is performed to find matches to the user input. The results are presented to the user. |
| Type: | Essential |
| Cross References: | |

**Typical Course of Events**

| Actor Action | System Response |
|---|---|
| 1. The user inputs known demographic information about a person (e.g. Name, date of birth, identification numbers, role, etc.) | 2. The Distributed Coplink application executes a query to find matches to the input criteria. |
| | 3. The system presents the Person Summary screen to the user, listing all persons who match the input criteria. |
| | 4. The system presents the user with a table consisting of the number of incidents the individual has been involved in, the Name |

of the individual, the date of birth, height, weight, a gang flag, a sex offender flag, and the source (agency) of the data.

| Actor Action | System Response |
|---|---|
| 5. The user views the data, and can sort it by any of the columns. | 6. The system sorts the data, based on the selected column, and re-presents the results to the user. |

## Use Case: Obtain Person Incident Summary (Multi-node)

| | |
|---|---|
| Use Case: | Obtain Person Incident Summary |
| Actors: | User |
| Purpose: | Obtain a listing of incidents with which a Person is associated |
| Overview: | From the Person Summary Table, the user selects a Person and asks the system to present a listing of the incidents associated with that person. |
| Type: | Essential |
| Cross-References: | Query Person must be executed first sequentially |

**Typical Course of Events**

| Actor Action | System Response |
|---|---|
| 1. From the Person Summary screen, the user selects a Person, and asks the system to display the incidents associated with that person. | |
| 2. The Person selected is differentiated by agency. | 3. The system executes the query, querying the specific Coplink node the user has selected. |
| | 4. The system presents the user with a table consisting of case numbers of the incidents, the address, the crime type, number of weapons, team, beat, whether the incident was gang or sex-offender related, and the source (agency) of the data. |
| 5. The user views the data, and can sort it by any of the columns. | 6. The system sorts the data, based on the selected column, and re-presents the results to the user. |

## Use Case: Obtain Person Details (Multi-node)

| | |
|---|---|
| Use Case: | Obtain Person Details |
| Actors: | User |
| Purpose: | Obtain a composite listing of information associated with a Person. |
| Overview: | From the Person Summary Table, the user selects a Person and asks the system to present a listing of the person detail information associated with that person. |
| Type: | Essential |
| Cross-References: | Query Person must be executed first sequentially |

**Typical Course of Events**

| Actor Action | System Response |
|---|---|
| 1. From the Person Summary screen, the user selects a Person, and asks the system to display the incidents associated with that person. | |
| 2. The Person selected is differentiated by agency. | 3. The system executes the query, querying the specific Coplink node the user has selected. |
| | 4. The system presents the user with a listing of all the essential demographic and identification information associated with the person. |
| | 5. The system also presents any known aliases the individual has used. |

## Use Case: Obtain Case Details (Multi-node)

| | |
|---|---|
| Use Case: | Obtain Case Details |
| Actors: | User |
| Purpose: | Obtain a composite listing of information associated with a particular case. |
| Overview: | From the Incident Summary Screen, the user selects a listed case number and asks the system to present a listing of the detail information associated with that case. |
| Type: | Essential |
| Cross-References: | Obtain Person Incident Summary must be executed first sequentially |

**Typical Course of Events**

| **Actor Action** | **System Response** |
|---|---|
| 1. From the Person Incident Summary screen, the user selects an individual listed case, and asks the system to display the details of that case. | |
| 2. The case selected is differentiated by agency. | 3. The system executes the query, querying the specific Coplink node the user has selected. |
| | 4. The system presents the user with a complete listing of all the essential case information. This includes all the captured information about a case. |
| | 5. Case information is presented, name(s) of officers involved in the case, a listing of all Persons associated with the case, listed by Role (e.g. Arrest, Victim, Suspect, etc.), and a listing of any Vehicle information associated with the case. |

## The Distributed Coplink System Overview

The following diagram is a high-level conceptual model and provides an overview of the distributed Coplink system architecture. The conceptual model notation follows the UML notation. [1, 12] In order to facilitate discussion of the system, the following concepts are defined:

1. Local Coplink Node: the home node for a particular client. For example, for a Tucson Police Department officer, the local node is TPDnode.
2. Remote (or distributed) Coplink Node: the set of Coplink nodes available to a user—those that are accessible from the local node.
3. Node Manager: the schema on the local Coplink node that controls access to local and remote Coplink schemas, and which "own" the distributed versions of the stored procedures.
4. Local Coplink Schema: the actual set of views, tables and objects that comprise the local incident and person data sets for the local Coplink node.
5. Remote Coplink Schema: the set of views and tables and objects that comprise the remote incident and person data sets for remote Coplink node(s).
6. Local OAS: the local application server servicing multiple local clients.
7. Remote OAS: remote application server(s) servicing remote clients.
8. Local Client: a client who has direct access to a local Coplink node via a local application server. "Local" in this case refers to users affiliated with an agency that manages a particular node. For example, a mobile officer accessing the local Coplink node via mobile radio communications is still a "Local" client.
9. Remote Client: a user affiliated with a remote agency—utilizing a remote Coplink node.
10. Local query: any query request from a local client requesting only local information.
11. Distributed query: any request from a local client requesting both local and remote agency information (or just remote agency information).

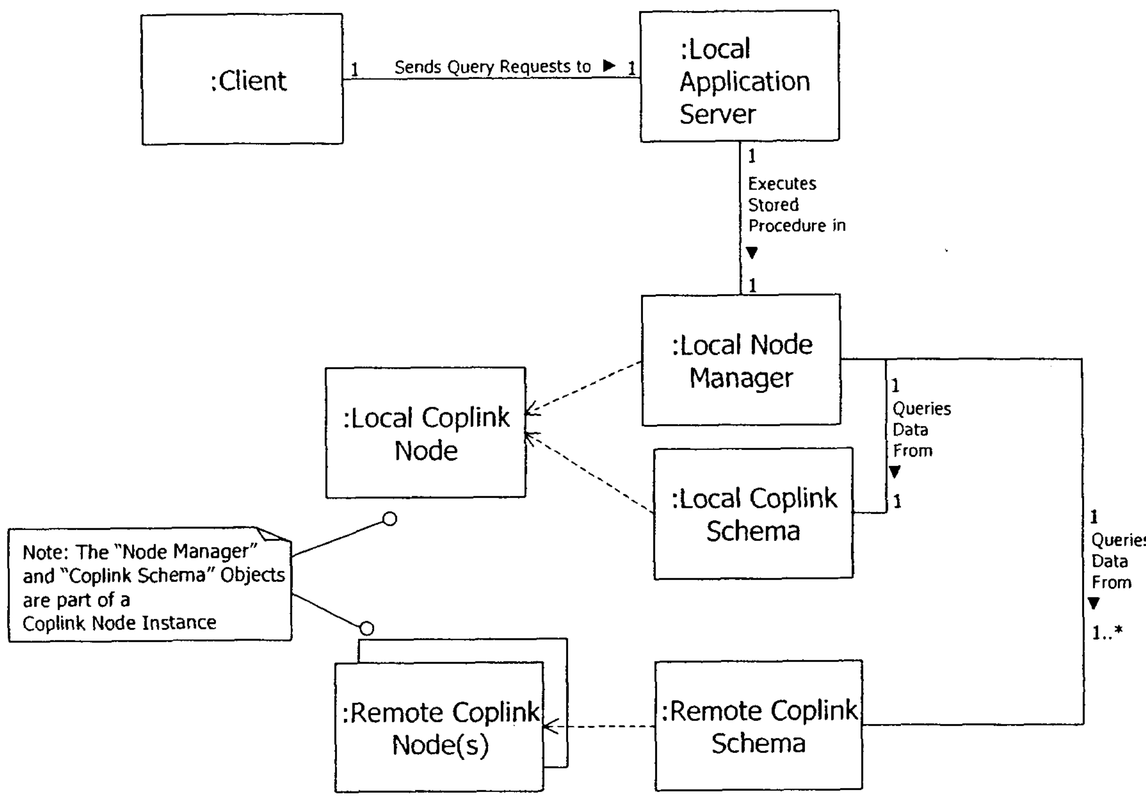



**Figure 5 High Level Conceptual Diagram for the Distributed Coplink System**

A high-level system sequence of events follows:

1. A local user issues a distributed query request from a local front-end application.
2. The local application server receives the request.
3. The local application server connects to the local Coplink node, and executes the specific stored procedure associated with the query.
4. The procedure queries the local Coplink node for the local data points that match the user's request.
5. The procedure queries the remote Coplink node(s) for the remote data points that match the user's request.
6. The information from each "sub-query" is returned to the OAS, delineated by source (agency).
7. The compiled information is sent back to the front-end application.
8. The front-end application parses and displays the information delineated by source agency.

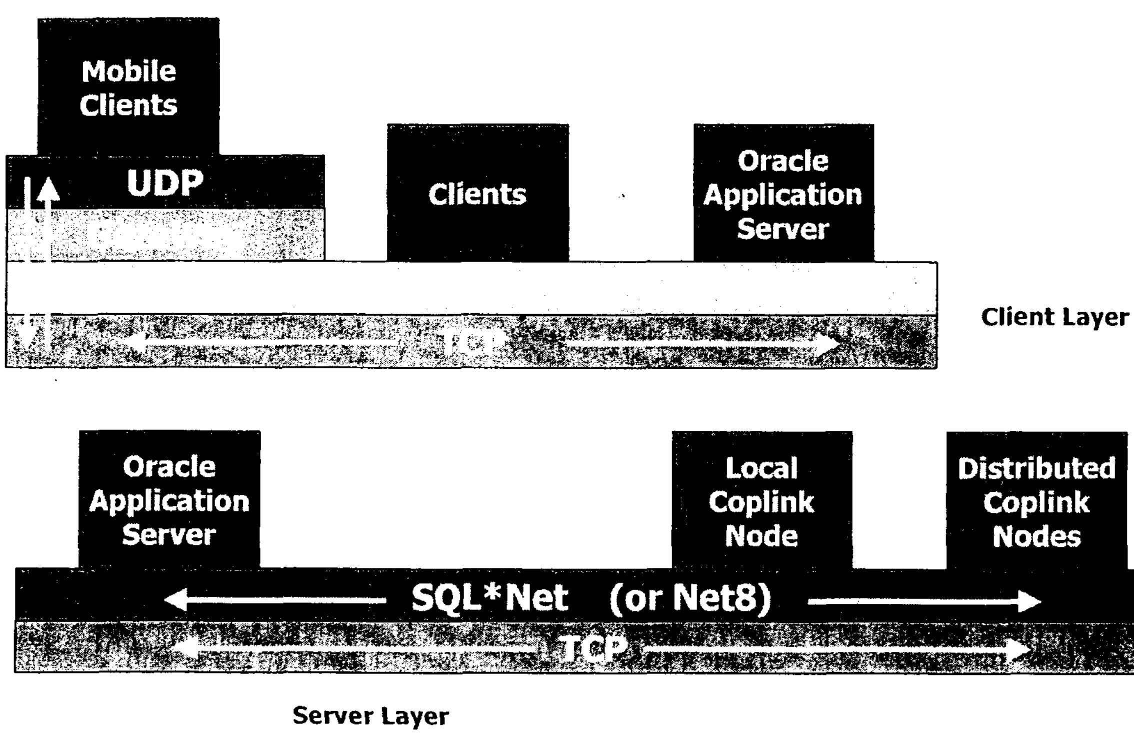
## Architecture of the Distributed Coplink System

### Underlying Networking Issues

There are three levels of communication within the distributed Coplink system. The first is HTTP-based URL connections between the client application and the local host application server. The second is the connection and data transfer between the application server and the local Coplink node database. The third is the communication and data transfer between distributed Coplink nodes. The latter two utilize Oracle's SQL*Net or Net8. Adding mobile clients utilizing UDP (User Datagram Protocol) adds a fourth layer of networking.

In the case of the distributed Coplink system, the main function of SQL*Net (or Net8) is to establish secure sessions and transfer data between the OAS and the server, and between distributed Coplink nodes.

**Figure 6 Networking and Communications Layers and Protocols**



Requests from the client applications are sent via HTTP-based URL request strings to the application server (OAS). In a local, non-mobile environment, these requests are sent via the TCP protocol. For remote mobile clients, UDP is utilized in order to minimize overhead in over wireless communications. However, the OAS does not currently support UPD directly, so all requests from mobile clients need to be converted to HTTP over TCP before they reach the OAS.

The communication between the OAS and the local Coplink node occurs via Oracle's SQL*Net (or Net8). Communication between a local Coplink node and remote or distributed nodes also utilize SQL*Net, which operates over TCP.

Oracle's SQL*Net and Net8 networking protocols utilize a fully qualified identifier for each node in a distributed system environment. To be fully qualified, the connection descriptor must consist of the following:

1. The domain (DNS or IP) address of the host (e.g. ai23.bpa.arizona.edu)
2. The communication protocol (e.g. TCP)
3. The network listener port for the node (default is usually 1521)
4. The system identifier (SID) of the Oracle instance running on the node (An SID is a unique name for an Oracle database instance that can be up to four alphanumeric characters in length).

An example of a fully qualified connection descriptor follows. The complete listing of connection descriptors is stored on each server or client in a networked system in a file called "`tnsnames.ora`". In the following example, the highlighted alias name "TPDnode" is utilized in place of referencing the complete connect string every time the connection descriptor is needed. By utilizing consistent alias names across all Coplink nodes, consistency between nodes can be achieved, and adding nodes to the system can be streamlined. (See Appendix E for a complete listing of a suggested standardized tnsnames.ora file for the distributed system).

```
TPDnode.world =
  (DESCRIPTION =
    (ADDRESS_LIST =
        (ADDRESS =
          (COMMUNITY = tcp.world)
          (PROTOCOL = TCP)
          (Host = ai23.bpa.arizona.edu)
          (Port = 1521)
        )
        (ADDRESS =
          (COMMUNITY = tcp.world)
          (PROTOCOL = TCP)
          (Host = ai23.bpa.arizona.edu)
          (Port = 1526)
        )
    )
    (CONNECT_DATA = (SID = ORCL)
    )
  )
```

## Creating Database Links

In order for two servers to "talk" to each other and share data, a secure link must be created between them. For example to connect to database node B from database node A, a link has to be created from A to B. The connection must be to a specific schema in database B, and must be set up with a password.

There are two types of database links: Public and schema-specific (or "private"). Continuing the example, a public database link created in database A can be utilized by any user in database A to connect to database B. A schema-specific link can only be used to connect from that schema (in database A) to the specified schema in database B.

By creating "private" database links between the specific Coplink schemas, an extra level of security can be achieved because only authorized access to the local or host node can then access distributed nodes. Further, secure access from the OAS to a local node is achieved by connecting to a specific schema. The stored procedures that query the database are stored as objects owned by that specific schema. Therefore, by maintaining secure links between the OAS and between Coplink nodes via specific schemas, controlled access to sensitive data can be achieved.

The following SQL statement creates a schema-specific database link, with the identifier "TPD_MGR_LINK" (where "INTEGBIG" is the local Coplink Schema):

```
CREATE DATABASE LINK "INTEGBIG".TPD_MGR_LINK
CONNECT TO INTEGBIG IDENTIFIED BY password
USING 'TPDNode';
```

Note the use of the connect string alias. This replaces the complete qualified string shown previously. A database link that connects the local Coplink node to each distributed node must be created. Once created, database links are then utilized to query information from distributed Coplink nodes.

## "Home" or Local Nodes

As shown in the high-level conceptual model, each Coplink node consists of two main schemas or objects, namely the "node manager" and the integrated, standardized "Coplink" schemas. In the current deployment of a single Coplink node, the connection from the OAS to the local database is made directly to the Coplink schema. In the distributed environment, to better divide functional responsibility, a separate "object" or schema is responsible for all the functionality related to the execution of queries. This functionality set will be discussed in detail in the section entitled "The Node Manager Schema."

In the distributed system, the idea of a local node is important because it acts as a starting point for all local and distributed queries. If the user simply wants to query the local database (as will still continue to be the case in the majority of analysis situations) there is no need to engage the other nodes in the distributed system in order to query the single local database.

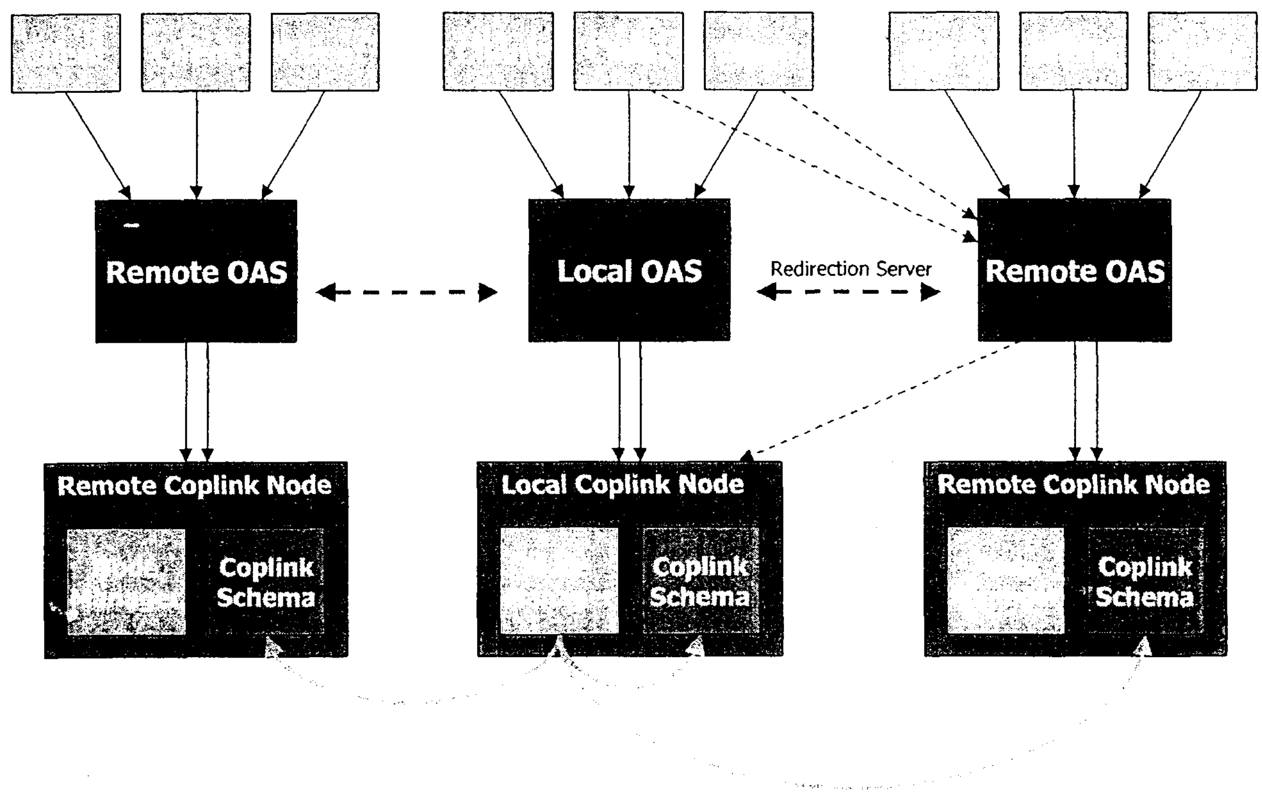
A local node for each agency provides a mechanism for decentralizing access to incident and person-related information. Each agency is therefore in control of its own data, without having to distribute "copies" of that information to remote nodes, or to a central repository. This concept of local "ownership" of information is critical to local law enforcement agencies. This has been a recurring theme obtained from discussions with representatives from both Tucson and Phoenix police departments, as well as representatives of national law enforcement agencies. Local agencies want to be able to share information, without necessarily having to be responsible for physically disseminating that information to other agencies. This system provides a mechanism for achieving that goal because all information stays "local" to the agency that generates it. The local agency maintains control over the data, and is solely responsible for its entry, update, and propagation to the integrated Coplink schema. The difference, then is that other agencies will be given access to those datasets for query purposes only.

## Three-Tier Architecture

As mentioned previously, the distributed Coplink system utilizes the same three-tier architecture as the single-node version. Each "local" Coplink application system will consist of its own application server, the back-end Coplink database node, and the front-end clients. This architecture provides a powerful mechanism for growing the distributed system. Each agency is in essence self-contained for its own local needs. As the number of nodes grows within the system, the balance is distributed evenly because each node has its own application server servicing the needs of its clients.

The diagram on the following page illustrates this distributed system. For clarity, the two "remote" systems are labeled as such from the perspective of the "local" user, but for the users local to those systems, they are of course considered to be the "local" OAS and Coplink nodes.

Figure 7  Three-Tier Architecture of the Distributed Coplink System




## Java Front End Application

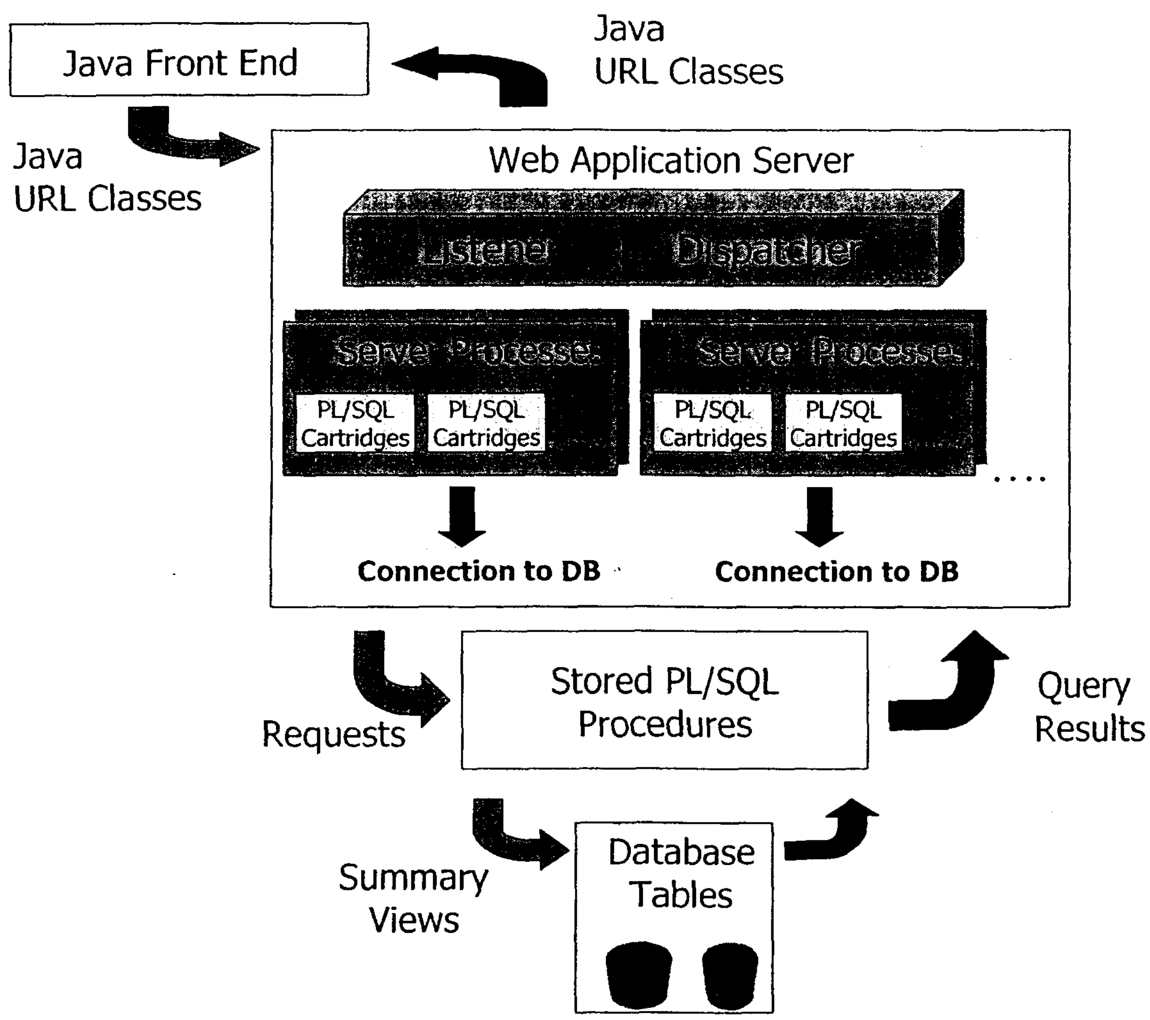
The client consists of a Java application or applet. On system startup, the application queries the node manager of the local database, which informs it as to which distributed nodes are available. The node manager also informs the front end application which node is the "local" node. In this way, a single version of the front-end application can be utilized by all agencies.

The "distributed" version of the application front end is almost identical to the previously developed single node version, with the exception that it executes the distributed versions of the stored procedures, and that on each screen, the source of the information is clearly identified. It also informs the user on each query screen of the Coplink nodes that are "up" and available for querying data. (Sample screen shots for each of the prototype query screens is included in Appendix C).

### Oracle Application Server

The Oracle Application Server (OAS) services all incoming query requests from clients. It utilizes CORBA (Common Object Request Broker Architecture) as its underlying processes. The following diagram provides a greater level of detail as to how the OAS services requests.

**Figure 8 Application Server Architecture (Detailed)**



The Coplink system utilizes PL/SQL stored procedures in the Coplink node databases. Requests from clients specify a stored procedure name in the database. For example, for the initial person query, the name of the procedure is "GET_PERSON_SRCH." For the distributed version of each procedure, the convention is *[procedure name]*_DIST (DISTributed). The distributed version of the person search, then, is GET_PERSON_SRCH_DIST.

The OAS utilizes a "cartridge" system to service requests. Because the procedures in the database are PL/SQL procedures, the Coplink version of the OAS utilizes the "PL/SQL Cartridge." Cartridges handle incoming requests from clients. They utilize any available server process, which maintains a connection to the database. The procedure is executed by passing to it the name/value pairs submitted by the client.

The OAS uses a pre-configured Database Access Descriptor (DAD) to locate the database to which to connect. The PL/SQL cartridge executes stored and pre-compiled PL/SQL source code. The PL/SQL cartridge comes with the PL/SQL Web Toolkit, an API that provides the mechanisms for returning information from the database in a format readable by the Java front end as a simple URL string.

Each OAS is scalable in regards to not only the number of connections it maintains to the database, but also the number of cartridges available to service client requests. These can scale up or down automatically based on user load.

### Load Balancing between Node Application Servers

One of the powerful features of the application server is that if the load increases on a particular application server node beyond what it can handle efficiently, it can redirect client requests to another OAS node. For an agency with a large set of users, this set up may be advisable to handle local requests. It also provides a level of redundancy, because if the primary node goes down, the secondary OAS node would still be able to handle user requests.

In the distributed Coplink environment, this has important implications for the stability of the system. As shown in Figure 7, it would be possible for cooperating agencies to set up their application servers as redirection servers for other agencies. It would also be possible for two small agencies, for example, to "share" one OAS node.

As shown in the diagram, if the local OAS node becomes overburdened or is temporarily unavailable, incoming user requests can be redirected to a "remote" OAS. That OAS, through a pre-configured database access descriptor, could access the local Coplink node associated with the original user. This action would be transparent to the user—there would be no knowledge on the user's part that his or her request was handled by a remote or secondary OAS.

## Regional Coplink Hosts

Carrying the previous load-balancing concept further, it would also be feasible for one agency to act as a host for other perhaps smaller community agencies (or a group of small agencies could combine efforts and develop one Coplink node). This is currently being considered in Maricopa County, Arizona. The Phoenix Police Department is discussing acting as a host to other valley-wide agencies, such as Glendale, Scottsdale, and Tempe police departments.

This would pose an additional challenge in correctly identifying the source of data. The current prototype implementation is to identify the source of the data by a unique three letter code associated with a single Coplink node (in other words, there is a one-to-one relationship between a Coplink node and an agency—one agency per node). If a single Coplink node consists of information from multiple agencies, another mechanism would have to be developed to identify the source of the information.

One possible solution is to add a "Source" table to each Coplink schema, which tracks all information datapoints by their unique agency source. This table would then provide a way to clearly associate specific information with its source, and report that to the front end application when a distributed query is executed.

For the purposes of this project and the associated prototype, however, it was assumed that each Coplink node consisted of a single agency's set of information.

## The Node Manager Schema

### More about Database Links and Distributed Tables

Utilizing database links, there are several ways to "hide" from the user that they are querying multiple databases. With a straightforward database link, a basic remote query looks something like:

```
select * from Person@TPD_MGR_LINK WHERE lname = lastname;
```

This utilizes the database link previously created. However, it is also possible to create a "synonym" for a particular object, even if it's distributed. In the following example, a synonym is created for the remote Person table on the TPD node:

```
create synonym PERSON_TPD for PERSON@TPD_MGR_LINK;
```

Then, the previous select statement would look like the following:

```
select * from PERSON_TPD WHERE lname = lastname;
```

In this way, the fact that the table being accessed is a distributed one is "hidden" from the user.

During initial experimentation, I tried several possible ways to access the tables and views on the distributed databases. One of the initial ways I experimented with was to make each specific set of distributed views a composite to the local node. For example, the initial person search view "COP_PERSON_MAIN_SRCH_V" could be set up as a composite view of all individual views across the multiple nodes:

```
create or replace view COP_PERSON_SRCH_V_DIST
as
select * from cop_person_main_srch_v UNION ALL
select * from cop_person_main_srch_v@PHX_MGR_LINK UNION ALL
select * from cop_person_main_srch_v@DPS_MGR_LINK;
```

This way querying the new single view "COP_PERSON_SRCH_V_DIST" would automatically return results from all distributed nodes. The fact that there are actually three databases being accessed is transparent to the user.

There are several issues and potential problems with this approach. First, although this is fairly straightforward for a small set of database instances (say up to about 4-5 distributed databases), it quickly becomes cumbersome for any number above three or four nodes. Second, if the user has a choice of querying between three, four, or more databases, is it likely they would choose ALL the databases to search at once, or is it more likely they would pick only 2 or 3 of the possible 4 or 5?

The final issue with this approach is its "Achilles Heal" and the major reason I decided against it. For this approach to work, all nodes that are defined as part of a view for a particular set of distributed views or tables must all be up and available at the same time. If not, the any query against the composite view would fail:

```
SQLWKS> select * from COP_PERSON_SRCH_V_DIST
        where lname = 'LASTNAME12';
select * from COP_PERSON_SRCH_V_DIST
              *
ORA-02068: following severe error from TPD_MGR_LINK
ORA-01034: ORACLE not available
ORA-09243: smsget: error attaching to SGA
      OSD-04101: invalid SGA: SGA not initialized
```

There are at least two scenarios that would cause this problem: first, the instance may not be up and available; second, the instance may be up and available, but the network connection from the local node to the distributed node is not operating, for whatever reason. In either case, a query based on a combined view of all N databases will fail.

There are, however, possible solutions that would remedy the situation:

1. Maintain, the views as composite views, based on all databases incorporated in the system, whether or not they are currently available. With this approach, however, exception handlers would have to be written into the PL/SQL code to handle the errors,

re-creating queries based on only the available databases. This was not deemed a feasible nor wise approach.

2. Redefine the views at front-end runtime, based on the available databases. When the front-end application is executed, the databases would be queried, and only those nodes that are currently available are used to then dynamically set up the composite views. This again assumes that the user will always want to base a query on all available databases.
3. Make the queries dynamic. For example, the user has available to her TPD, PHX, and DPS databases, but only wants to query TPD and PHX. In this case, either a "dynamic" view is created for the TPD and PHX databases, or queries are run separately against the TPD and PHX databases, and the results are combined by the front-end.
4. A combination of (2) & (3): At runtime, the front end executes a "GET_AVAILABLE_NODES" query that probes for all available Coplink nodes. These are presented to the user. Then, the user selects the database(s) for a query, and the queries are built and executed dynamically, querying only the nodes that the user has selected.

I tested the first option on a small scale with a procedure that queries a pre-defined "global" view across three databases. (The PL/SQL procedure entitled "TEST_DIST_QUERY" is contained in Appendix F: Miscellaneous Testing Code). If the query on the global view fails, then the error exception will be "caught" and the procedure will query the available databases independently. This approach, however, would become very cumbersome as the number of distributed nodes increases because covering all possible combinations and points of failure would quickly become too complex to implement.

### Introduction of the NODE_MANAGER schema

The best implementation was to remove the responsibility of checking for available nodes from the front-end application. One of the GRASP patterns described previously is the "Expert" pattern [1] which states that the responsibility for a function should be given to the object that knows the most about the particular responsibility. In this case, the responsibility for checking to see if other nodes are available is best accomplished by the database system itself. To that end, I created a procedure that would query the other nodes in the distributed system and verify that they are available. This procedure "UPDATE_AVAILABLE_NODES," is part of the "NODE_MANAGER" schema. To carry the concept further, the Node Manager has the following responsibilities:

1. Check on a regular basis (every X minutes) whether or not the other Coplink nodes are available.
2. Track all the information about the other Coplink nodes that is necessary to successfully communicate with and query the remote Coplink nodes.
3. Be the owner of the database links set up to query remote Coplink nodes.
4. Maintain and execute all the distributed versions of the Coplink queries.

### "Friend" to the Coplink Schema

The NODE_MANAGER schema on a local machine is a "friend" to the integrated Coplink schema on each Coplink node, both local and distributed. In other words, it knows about all the nodes on the distributed Coplink system. It can access any Coplink schema, but not the reverse. In this way, only users who have access to the distributed versions of the Coplink system have the ability to retrieve information from any remote node.

The OAS, when executing a distributed procedure, accesses the Node Manager schema directly. The Node Manager in turn then accesses the local and distributed Coplink schemas to retrieve the requested datapoints. The stored procedures are owned by the Node Manager, and are executed by it.

### Available_Nodes Table

In order to track the information needed to maintain communications and execute distributed queries, the node manager relies on a simple table that records the following information for each Coplink node:

**Table 2 The "AVAILABLE_NODES" Table**

| Attribute | Description |
|---|---|
| NODE_NAME | The unique identifier for each Coplink node. |
| DOMAIN | The IP or DNS address of the node server. |
| INSTANCE_NAME | The SID instance name of the database. |
| SCHEMA_NAME | The Coplink schema name. |
| AVAILABLE | A 'Y'es or 'N'o flag to indicate whether or not the node is available. |
| DB_LINK_NAME | The named identifier of the database link to the distributed node. If the node is the local node this value defaults to 'LOCAL'. |
| CONNECT_STRING_ALIAS | The connect string alias to the local node. If the node is the local node this value defaults to 'LOCAL'. |
| MUGSHOT_LOCATION | The location of the mugshots for the node (used by the Person Details and Case Details to fetch mugshots from each node server). |

The sample contents of the table on one Coplink node follows:

```
NOD DOMAIN                  INSTANCE SCHEMA_NAME  A DB_LINK_NAME   CONNECT_STRING_ALIAS
--- --------------------    -------- ------------ - -------------- --------------------
TPD ai23.bpa.arizona.edu    ORCL     INTEGBIG     Y LOCAL          LOCAL
DPS ai24.bpa.arizona.edu    ORCL     INTEGSMALL   Y DPS_MGR_LINK   DPSNODE
PHX cops.bpa.arizona.edu    SID1     INTEGBIG     Y PHX_MGR_LINK   PHXNODE
```

In this sample scenario, all three nodes are available, as indicated in the "AVAILABLE" column (highlighted in yellow). If a node is temporarily unavailable, the column will contain an 'N' indicating that the node is temporarily not available:

```
NOD DOMAIN                  INSTANCE SCHEMA_NAME  A DB_LINK_NAME   CONNECT_STRING_ALIAS
--- --------------------    -------- ------------ - -------------- --------------------
TPD ai23.bpa.arizona.edu    ORCL     INTEGBIG     Y LOCAL          LOCAL
DPS ai24.bpa.arizona.edu    ORCL     INTEGSMALL   N DPS_MGR_LINK   DPSNODE
PHX cops.bpa.arizona.edu    SID1     INTEGBIG     Y PHX_MGR_LINK   PHXNODE
```

## Update_Available_Nodes Procedure

The update available nodes procedure is responsible for updating the AVAILABLE_NODES table. It does so by querying the V$SGA table (System Global Area) on each remote database. If the Select query is executed successfully, meaning that the database instance is available, then the local AVAILABLE_NODES table is updated successfully with a 'Y'. If the connection to the remote database fails, resulting in an error message, the procedure catches the exception, and updates the AVAILABLE column with an 'N'.

There are three possible ways this procedure could be utilized:

1. Run it every time a query is executed. In other words, each time a query is executed from a client, the client calls the update available nodes procedure, and then checks the table to see if the node is available.
2. Execute it when the front-end application starts. (This would again be a two-step process: execute the procedure and then check the available_nodes tables to see if the node is available).
3. Execute it every X minutes as a background process on the host node.

Either option is feasible, but the first may cause unnecessary overhead increase in query response time. Both the first and the second options are two-step processes, which could add unnecessary overhead to the query execution process. Clearly the best option is to have the table updated automatically every X minutes, and then, when the front-end application starts, it will simply have to check the table to see which nodes are available.

Utilizing the DBMS_JOB package, I set up the procedure to run every 5 minutes on the local "TPD" Coplink node, utilizing the following script in SQL*Plus:

```
var jobno NUMBER
BEGIN
dbms_job.submit
```

```
(job => :jobno,
 what => 'NODE_MANAGER.UPDATE_AVAILABLE_NODES;',
 next_date => SYSDATE + 1/1440,
 interval => 'SYSDATE + 1/1440');

END;
/
print jobno
```

The job number is automatically assigned as a system job number from the SYS.JOBSEQ sequence. Querying the DBA_JOBS view returns the following:

```
JOB   LOG_USER      LAST_DATE    LAST_SEC NEXT_DATE    NEXT_SEC INTERVAL
----- ------------ ----------- -------- ----------- -------- ----------------------
    1 REPADMIN      21-MAR-99    19:08:28 22-MAR-99    19:08:28 /*1:Day*/ sysdate + 1
   22 NODE_MANAGER 22-MAR-99    15:58:40 22-MAR-99    16:03:40 SYSDATE + 5/1440
```

This shows that the submitted "job," namely execution of the UPDATE_AVAILABLE_NODES procedure will automatically occur every five minutes. This interval can be set to any interval by utilizing the following command:

```
execute dbms_job.interval(22, 'SYSDATE + 15/1440');
commit;
```

This would change the interval to 15 minutes. In a production system, it wouldn't be unreasonable to have the query execute every minute, as the execution time is rapid, and the overhead has very little impact on the database system.

### Get_Available_Nodes

By automatically and continuously executing the update_available_nodes procedure, the current status of the distributed nodes is always kept up to date in the local Coplink node as information in the available_nodes table of the node manager. The front-end application can then "query" the local table and update its knowledge of the status of the distributed nodes.

To perform this function, I developed a second administrative procedure, GET_AVAILABLE_NODES that is called each time the front-end application is started (as described in the "System Startup" use case). This procedure could also be executed periodically during a user's session, giving the user a continuous update as to the status of the distributed nodes. The procedure simply queries the local available_nodes table and reports the status to the front-end application. Since this procedure is fairly concise, it is shown here in its entirety:

```
CREATE OR REPLACE PROCEDURE
"NODE_MANAGER".GET_AVAILABLE_NODES

--GET_AVAILABLE_NODES

IS

CURSOR get_nodes IS
SELECT NODE_NAME, DB_LINK_NAME, AVAILABLE
FROM NODE_MANAGER.AVAILABLE_NODES;

BEGIN
      FOR rec IN get_nodes
      LOOP
              IF rec.db_link_name = 'LOCAL' THEN
                   htp.print('LOCAL%%%' || rec.node_name || '%%%' ||
rec.available || '%%%');
              ELSE
                     htp.print(rec.node_name || '%%%' || rec.available
|| '%%%');
              END IF;
      END LOOP;
END;
```

### Distributed Versions of the Coplink Query Procedures

The most complex functionality of the Node Manager is to query the distributed Coplink nodes, depending on the user's selections. For a single node version of Coplink, only the initial query procedures were of necessity dynamic in nature. For example, from the Person search screen, the user could elect to fill in any number from one to nine fields, and the search would be performed on that combination of input parameters. The actual select statement is constructed, parsed, and executed dynamically for each query. After the initial query, however, all subsequent queries were more static, because each depended solely on a combination of primary keys that could be obtained as a result of the initial query.

In the case of the distributed queries, *every* sub-query within a procedure has to be built dynamically. This significantly increased the coding complexity of the procedures, because some of the procedures have as many as six sub-queries (as six cursors that query different Coplink views). In the distributed environment, the variable parts of a particular subquery are:

1. The database link name for the remote node, or the absence of a database link if the query is to be made on the local node
2. The schema name for the remote or local node
3. The variable number of input parameters entered by the user

For example, a subquery from the GET_PERSON_INCIDENT_SUM_DIST procedure is built in three sections: the "select_clause," the "link_clause" and the "where_clause":

```
srch_select_clause := 'SELECT sa_cop_incident_main_pk FROM ' ||
current_schema || '.cop_person_main_key_v';

IF (current_db_link = 'LOCAL') THEN
        link_clause := ' ';
    ELSE
        link_clause := '@' || current_db_link || ' ';
    END IF;

srch_where_clause := ' WHERE sa_cop_person_main_pk = ' || qt ||
Vsa_cop_person_main_pk || qt;
```

After each "section" of the select clause is built, it must then be parsed and the return columns have to be defined. Then the cursor is executed, values are fetched into local variables in the procedure which can then be used to return results back to the front-end application via the OAS. (The process of creating dynamic SQL cursors within PL/SQL is discussed in greater detail in the next section).

For each query request from the front-end application, users will select 1 or more nodes to query. The list of nodes will be passed as an additional parameter to the distributed version of the procedure. The search queries then perform the searches on each of the available nodes as described above, with the particular values of the schema names and database links changing dynamically for each node. Results are then returned to the user for each node, again indicating in the result set from which Coplink node the information was retrieved.

For example, running "EDDIE GARCIA" from the initial Person query screen, a listing of matches from each node will be displayed on the Person Summary Table. The initial query would look like the following:

```
NODE_MANAGER.GET_PERSON_SRCH_DIST?lname=GARCIA&fname=EDDIE&nodes=TPD,PHX
```

The return would be the usual summary table, with an additonal column. The final column in the summary table will be the source node (e.g. "TPD" or "PHX"). Then, if the user selects a row, requesting either the Person Details or the Incident Summary Screen, the query will be submitted with the corresponding node as an additional parameter:

```
NODE_MANAGER.GET_PERSON_INCIDENT_SUM?sa_cop_person_main_pk=
~0500441716&nodes=TPD
```

From the Incident Summary screen, the same would apply when the user selects the Case Details screen:

```
NODE_MANAGER.GET_CASE_DETAILS_DIST?sa_cop_incident_main_pk=
[HYKc134408&nodes=TPD
```

An important assumption here is the standardization of the naming of the Coplink Views across all nodes. If the names are not standard, these would also have to be dynamically added to each Select statement.

Also, if the user selects more than one node to query from, the set of Select queries has to be generated once for each node selected. Therefore, all the preceding would change as each node is processed.

For the prototype application the following four procedures were re-written as distributed versions of each procedure (the complete text of each is included in Appendix B):

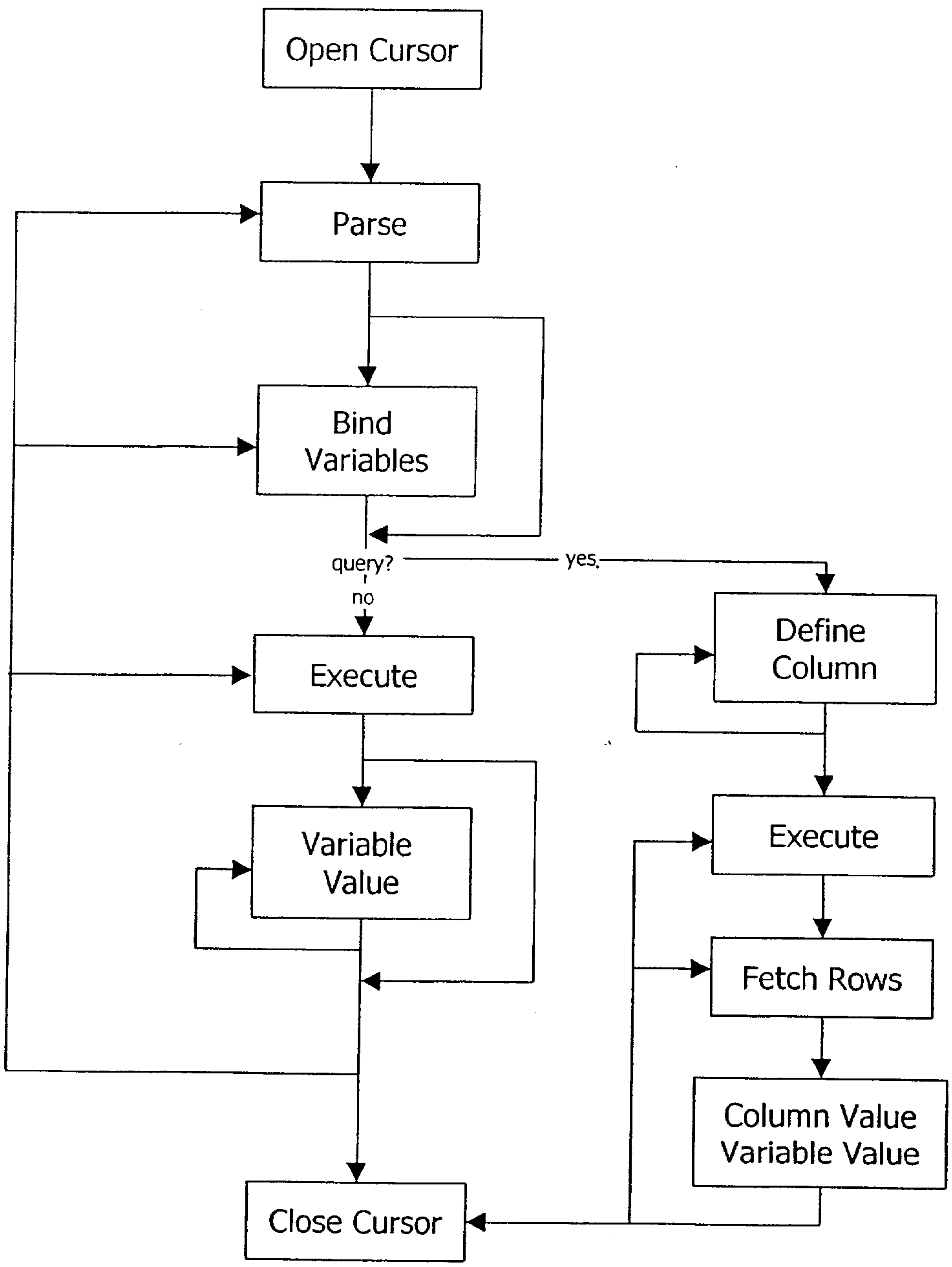
**Table 3 Distributed Versions of the Coplink Query Procedures**

| Single Node Version Name | Distributed Coplink Version Name | Description |
|---|---|---|
| GET_PERSON_SRCH | GET_PERSON_SRCH_DIST | Called from: the initial Person Search screen.<br>Result: The Person Summary screen, listing all persons that match the input search criteria across all select Coplink nodes. |
| GET_PERSON_INCIDENT_SUM | GET_PERSON_INCIDENT_SUM_DIST | Called from: the Person Summary screen.<br>Result: the Person Incident Summary screen, listing all incidents associated with the chosen individual for the selected node. |
| GET_PERSON_DETAILS | GET_PERSON_DETAILS_DIST | Called from: The Person Summary screen.<br>Result: The Person Details screen, listing all the known information about the person for the selected node, including Aliases |
| GET_CASE_DETAILS | GET_CASE_DETAILS_DIST | Called from: any Incidents Summary screen<br>Result: The Case Details screen, listing all the details associated with a particular case for the selected node. |

## The Use of Dynamic SQL

The following diagram illustrates the complexity in writing and executing dynamic SQL Select statements[13]:

**Figure 9 Dynamic SQL Execution Flow**



The steps are explained as follows:

1. ***Open a Cursor.*** The RDMS opens a cursor as a pointer to or a handle on an allocated space in memory. The return is an integer that signifies the cursor. This integer is used in all subsequent DBMS_SQL calls involving the cursor. This is different from a static cursor in native PL/SQL (i.e. the standard operations "FOR rec IN *cursor*" or "FETCH INTO" cannot be used with this type of cursor).
2. ***Parse the SQL statement.*** This verifies that the SQL statement is valid and properly constructed. It also associates the SQL statement with the cursor previously opened.
3. ***Bind all host variables.*** If variables are to be bound to the query, placeholders need to be included in the query, preceded with a colon. For example :lname would allow the procedure to later bind the specific value for lname.
4. ***Define the column(s) in the SELECT statement.*** This is similar to the "INTO" clause of an implicit SELECT statement in PL/SQL. This defines a correspondence between the expressions in the list of the SQL statement and the local PL/SQL variables receiving those values when the row is fetched.
5. ***Execute the SQL Statement.*** Executes the specified cursor's SQL statement.
6. ***Fetch rows from the dynamic SQL query.*** Rows are fetched from the cursor, but not directly into local PL/SQL variables.
7. ***Retrieve values from the execution of the dynamic SQL.*** Values from the SELECT statement are retrieved into the PL/SQL variables using the COLUMN_VALUE procedure.
8. ***Close the cursor.*** This cleans up the cursor and releases any memory allocated to it.

The point in reviewing the complexity of this process is to reiterate the fact that because every query in the distributed versions of the Coplink queries needs to be created dynamically, the overall complexity of each procedure increases significantly. For example, the original non-distributed GET_CASE_DETAILS procedure consisted of 115 lines of code. The equivalent distributed version GET_CASE_DETAILS_DIST procedure consisted of 510 lines of code.

This creates several problems:

1. Writing and Debugging procedures becomes increasingly difficult.
2. Maintenance and updates also becomes more difficult.

3. The procedures are more prone to failure when executed, and discovering the source of the failure can be troublesome.

One way to mitigate this complexity is to make the code more modular. [14] Previously, each Coplink procedure was fairly self-contained. Each was, however, procedural in nature. A solution to this problem would be to identify the parts of the code that are common to all or most procedures, and make those common logic segments separate procedures or functions. With further study, the set of procedures could easily evolve into a more modular, object-oriented package of procedures. In place of nine complex, procedural sets of code, a package could be developed that would encapsulate the functionality of all nine procedures, making it more robust, scalable, and maintainable.

## A Prototype Distributed Coplink System

With the node manager schema in place, and all the previously described procedures re-written to work in a distributed environment, a prototype system was designed and executed.

In the development of the current Coplink system for TPD, we have utilized three different versions of the Record Management System (RMS), Gang (CIC), and mugshot (ELVIS) database that went into developing the integrated Coplink schema. The first was a small subset taken from TPD's systems, consisting of about 5,000 incident records. The second was a full incident record set taken from TPD's systems in the spring of 1998. The final set was a recent capture of the full incident set from TPD, executed in mid-April of 1999. Each of the latter consisted of approximately 1.5 million incident record sets, with some of the larger tables approaching 2.5 million rows.

This provided an excellent opportunity to develop and test a prototype system. The distribution of the three nodes was allocated as follows:

**Table 4 Prototype System Configuration**

| Coplink "Node" Name | Schema Name / Oracle Version / OS | Record Set Size | Location |
|---|---|---|---|
| TPD ("LOCAL") | INTEGBIG / Oracle 7.3 / Windows NT | ~1.5 million | ai23.bpa.arizona.edu |
| "PHX" ("REMOTE") | INTEGBIG / Oracle 8.0.3 / UNIX | ~1.5 million | cops.bpa.arizona.edu |
| "DPS" ("REMOTE") | INTEGSMALL / Oracle 8.0.5 / Windows NT | 5,000 | ai24.bpa.arizona.edu |

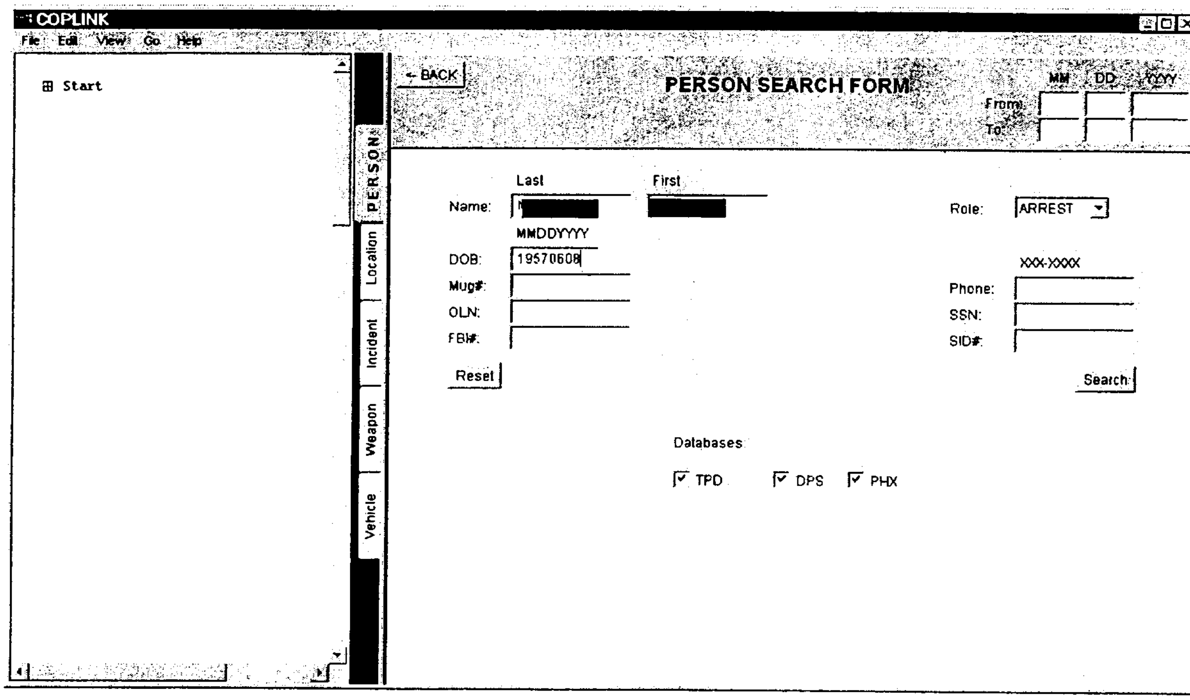
Although the actual data on all three nodes was acquired from TPD, the other two nodes were given names of different agencies to better illustrate how the system would function. Also, the information on the TPD node was more recent, so there were some differences that could be seen between the two larger nodes.

The other important aspect of the test was that at least two of the nodes were large datasets. This is important because it showed that executing a complex query across multiple nodes was performed without a noticeable difference in speed vs. a single local query.

The front-end application was modified as a "distributed" version as discussed previously, with each screen detailing the source of the returned information. A full set of sample screen shots from the prototype are included in Appendix C, but three are included on the following pages for illustration purposes (all last names are blacked out for security purposes).

Note that each initial query screen lists the available nodes as "check boxes" selectable by the user. The local node is always selected as the default, which the user can de-select if desired. Each subsequent summary or details screen will clearly indicate the source of the incident or person-related information.

## Figure 10 Initial Distributed Person Search Form




## Figure 11 Distributed Person Summary Screen

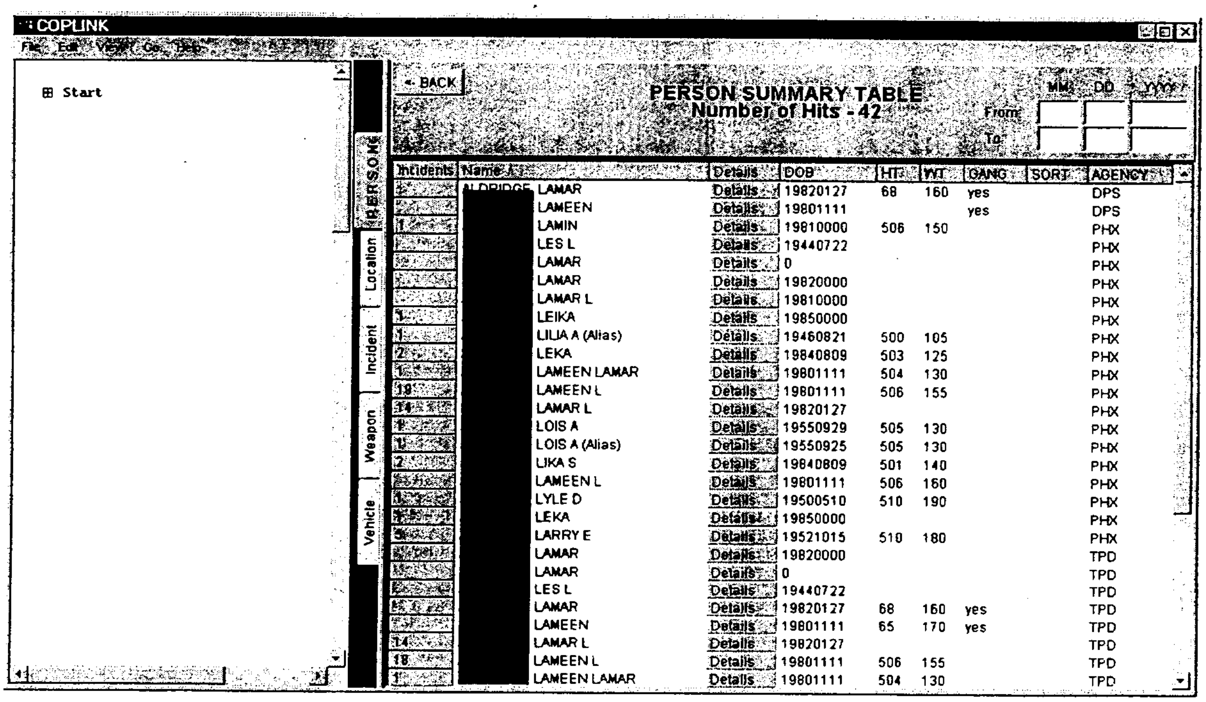
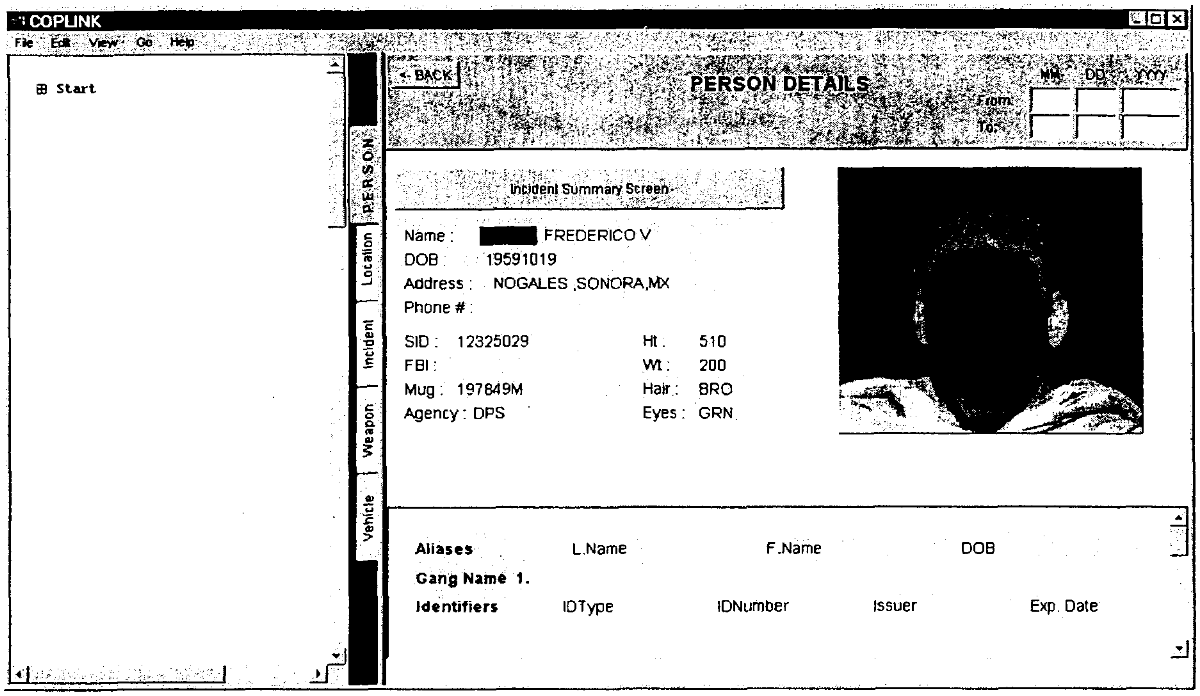


**Figure 12 Distributed Person Details Screen**




## Testing the Prototype System

As mentioned, "AI23" was set up as the local TPD node, which contained the local node_manager schema. Because I had control over all three databases, I experimented with taking the nodes offline and bringing them back online at a later point in time. The update_available_nodes procedure performed continuously without error. In fact, as of the writing of this report, the procedure has executed over 6,000 times every five minutes for almost three weeks without a single error in execution.

Each time either the "PHX" or "DPS" node was stopped, the procedure updated the Available_Nodes table correctly within the five minute interval set between job execution. Because this process occurs automatically in the background, the front-end application is able to be continuously updated as to the status of the distributed system.

Another important test was to verify that the procedures work across all nodes independently. As shown in table 4, the three machines are each configured differently: two running Windows NT, with Oracle 7.3 and 8.0.5 as the database version, respectively, and the third configured as a Digital Alpha-based system running Digital UNIX V4.0D (Rev. 878) with Oracle 8.0.3 as the database version. Identical versions of the Node_Manager schema were loaded on each machine, with the same versions of each procedure.

The next step in conducting this test was to set up the OAS to connect to each version of the Node_Manager separately. The queries were then tested independently, running from each machine as the "Local" node and then executing the procedures. The queries performed equally well, executed from all three nodes: the "TPD" node (ai23.bpa), the "PHX" node (cops.bpa), and the "DPS" node (ai24.bpa).

Because the "INTEGSMALL" version of the Coplink schema is not completely up to date with that of the other two nodes, however, the procedures did not always return the correct results from the "DPS" node. For example, the ability to search by "Role" from the Person Query screen is not implemented in the INTEGSMALL version of the Coplink schema. However, when run from the Node_Manager on ai24, the procedures correctly queried the other two nodes, where the ability to search by Role has been added to the INTEGBIG version of the Coplink schema.

## Scaling the Distributed Coplink System

Care was taken in designing the proposed distributed system so that adding additional nodes could occur seamlessly as possible. The caveat is again that any distributed node adhere to a standardized Coplink schema, based on the Coplink Views and supporting objects discussed previously.

This admittedly encompassing assumption is about to be tested. Tucson Police Department is beginning to work closely with the Phoenix Police Department in mapping its incident reporting system to the Coplink standardized schema. Initially, Phoenix will bring its information online, and then subsequently will act as a "host" for other law enforcement agencies in Maricopa County, Arizona. This alliance will be the first test of a real-world distributed Coplink system.

### Regional vs. National Access to Coplink Nodes

It's important to note that thus far discussions have been to add nodes on a regionally cooperative basis. Conceivably, however, other Coplink nodes from across the country could be added to the system, creating a powerful, national system of information sharing. As the number of nodes increase, however, mechanisms for managing access to individual nodes would have to be developed.

One possibility is to develop regional Coplink node "clusters" that cooperate within meaningful geographic boundaries. Access to any node within the cluster would be controlled by a central regional authority responsible for updating each node as to the existence and configuration of the other participating nodes. For example, one meaningful "cluster" may include Arizona, New Mexico, and parts of Southern California, specifically the San Diego area. Each Coplink node within the cluster would be accessible from any other node.

On a national level, access from one cluster to another would be possible, but only by specific request from the user. In other words, the initial presentation of available nodes in the front-end application would be just the regular "local cluster." If the user wished to query a node in another region's cluster, a specific request would be routed to that cluster.

By providing a test case that essentially covers the majority of Arizona, the Coplink project should be able to acquire significant expertise in then applying a distributed system to other regional areas, and then to take the project to a national level.

## Handling Mugshots

As currently implemented, mugshots are retrieved from each Coplink node via HTTP direct query. This is executed as follows: If a person in the Coplink schema has a mugshot available, the Person table contains as one of its rows a "photo_flag" indicating that there is a photo available, and another column that contains the name of the photo (photo_filename). The domain portion of the mugshot URL is stored in the procedure, and the photofilename is appended to that domain string when results are returned to the front end application. The Coplink application fetches the picture directly from a virtual path on the server and displays it. As an example, for the "TPD" node, the virtual path is as follows:

```
http://ai23.bpa.arizona.edu/mugs/
```

and a sample photo_filename is:

```
/1/3/100333.001
```

The complete http path would be:

```
http://ai23.bpa.arizona.edu/mugs/1/3/100333.001
```

To better handle the retrieval of mugshots in the distributed environment, I decided to add another column to the AVAILABLE_NODES table called "MUGSHOT_LOCATION". This stores the domain path specific to each machine. (INTEGSMALL only references a "dummy" photo housed on ai23).

**Table 5 Handling Mugshot Retrieval**

| Coplink Node | "Agency" | Mugshot Location (Actual Path) | Mugshot Location (Virtual Path) |
|---|---|---|---|
| ai23.bpa.arizona.edu | TPD | E:\mugs | http://ai23.bpa.arizona.edu/mugs/ |
| cops.bpa.arizona.edu | PHX | /u01/ms | http://cops.bpa.arizona.edu/ms/ |
| ai24.bpa.arizona.edu | DPS | E:\mugs\0\0\dummy.jpg | http://ai23.bpa.arizona.edu/mugs/0/0/dummy.jpg |

Both GET_CASE_DETAILS and GET_PERSON_DETAILS are affected by this because they both will display a mugshot if there is one available in the system.

Another possible solution to this problem is to store the mugshots themselves directly in the each Coplink node database as "BLOBs" (Binary Large Objects). They could then be fetched directly from the database as part of the execution of the stored procedure.

## Suggestions for Further Study

### Security

Of obvious concern to all involved in law enforcement activities are the issues of security of data access, sharing, and transmission. In regards to a distributed Coplink system, there are several areas of concern in attempting to assure security of data:

1. Secure access to the Application Server
2. Secure access to each data source from the application server
3. Prevention of indirect access to a Coplink database node
4. Transmission of data between Coplink nodes, specifically between two database nodes
5. Implications of mobile access to Coplink nodes and the OAS

This is an area that is currently undergoing further study and is not necessarily within the scope of this project. However, there are several issues that can be addressed almost immediately improve security of each Coplink system.

First, the Application Server has several ways to implement security, including restricting access to the OAS by specific username and password (basic or digest using usernames, groups, and realms), specific IP addresses or domains, or by Secured Socket Layer (SSL) utilizing certificates.

Second, by limiting access to the stored procedures to the Node_Manager schema specifically as mentioned previously, tighter control over access to the database and execution of procedures is maintained. This requires special attention to the privileges and roles assigned to users and schemas at the database level.

Finally, requiring further study would be the issues of data encryption between nodes, and utilizing firewalls to protect access to application servers and Coplink database nodes. For example, Oracle's "Secure Network Services" provide a layer of encryption and data security between database nodes. Algorithms of varying sophistication (depending on domestic or international usage) encrypt data, and ensure that it has not been altered or tampered with during transmission.

## NIBRS

NIBRS is the National Incident Based Reporting System, managed by the Federal Bureau of Investigation. It "is an incident-based reporting system through which data are collected on each single crime occurrence. NIBRS data are designed to be generated as a by-product of local, state, and federal automated records systems."

In essence, NIBRS is an initiative to standardize the type of information recorded for specifically delineated crime types. State and local agencies that participate in the NIBRS initiative collect data according to the guidelines dictated by NIBRS. The agencies then provide this information in standardized formats to the FBI for statistical crime analysis purposes.

As mentioned previously, one of the keys in assuring success in developing a distributed Coplink system is to standardize where and whenever possible. The NIBRS initiative, then, may provide a common framework for designing incident and other crime reports that are consistent between agencies. It would also possibly provide a mechanism for standardizing naming conventions for data points, one of the more difficult issues in integrating the usually disparate underlying databases that a standardized Coplink node consists of. It may prove useful, then, to investigate the NIBRS standards more closely, and determine whether they are applicable to streamlining development of Coplink between multiple agencies.

## Adding Concept Space to the Distributed Coplink System

The Coplink application is an incident-based record query system. Its sister in the Coplink Project is Coplink Concept Space, which is a system that uncovers relationships between concepts across all data points. Users can query based on the following concept categories: Person, Location, Organization, Vehicle, and Crime Type. Coplink Concept Space has been under development in parallel with the query-based Coplink application.

An area of further study and development, then, would be to add Coplink Concept Space to the distributed Coplink system. Modifying the Node_Manager on each Coplink node to be able to query distributed versions of Concept Space would be straightforward. The few procedures associated with Concept Space are already implemented with Dynamic SQL, so modifying them to work across multiple nodes would require minimal effort for experienced PL/SQL programmers.

However, the structure of the underlying data sets would have to be evaluated, as to whether they could be queried separately, and then combine the results at the front-end level, or whether a unified set of data would have to be created based on two or more datasets from different agencies.

## User Studies and Performance Evaluation

Part of the success to date of the Coplink project has been its close relationship with the Tucson Police Department. At each phase of the project, careful care was given to considering the input of officers and advisors from TPD. This would hold true for a distributed Coplink system as well. User input and studies as to how the system would best be utilized should be designed and executed as the project proceeds.

Also, careful study would have to be undertaken to test the overall performance of a distributed system, especially across significant geographic and logical network distances, as network overhead may significantly impact perceived or acceptable response time. Careful tuning of procedure execution may be critical to adequate response time for query execution, and implementing parallel execution of queries is another area that might be investigated. Again, adhering to the standardized Coplink database set up and schema would be critical because performance at the database has been studied extensively by the Coplink team.

Finally, based on the performance and stress testing work undertaken by the Coplink team, similar performance and stress tests could be easily applied to a distributed Coplink system. This would provide valuable information about the scalability of the system, and test the feasibility and responsiveness of possible redirection and load balancing between cooperating Coplink application server nodes at different agencies.

## Summary

The concept of a regional, distributed Coplink system has been shown to be feasible within certain design guidelines. Building on the success of the "single" node version currently under testing with TPD, guidelines for building a distributed system have readily followed based on the expertise gained thus far with TPD.

The distributed Coplink prototype system described has been tested on three Coplink "nodes", utilizing versions of two complete large datasets from TPD. Further testing and deployment issues would necessarily have to be undertaken, but the Coplink team as a whole has gained significant expertise and knowledge over the course of the project with TPD that would contribute significantly to similar efforts at other agencies.

The next two phases in the evolution of the Coplink project as a whole are first to successfully deploy a live, constantly up-to-date version of the Coplink system at TPD. The second is to repeat at another law enforcement agency the success experienced thus far at TPD. With the goal of a cooperative system between the two Coplink instances, data sharing between the two agencies would become a feasible reality. Close attention to standardization between the two initial agencies is key to that success.

I've had the opportunity to discuss the implications of a shared, distributed system with officers from both the Tucson and Phoenix police departments. It is a common consensus that a shared system between the two agencies would have significant impact on law enforcement efforts within the state of Arizona. Officers from both departments quickly become animated in describing the possibilities and the implications of sharing law enforcement information between the two agencies. It is gratifying to see that the efforts of the entire Coplink team are bringing these goals to fruition.

# Bibliography


1. Larman, C., *Applying UML and Patterns to Object-Oriented Analysis and Design*. First Edition ed. 1998: Prentice Hall PTR.
2. Dye, C., *Oracle Distributed Systems*. First Edition ed, ed. D. Russell. 1999: O'Reilly & Associates.
3. Yang, A., J. Linn, and D. Quadrato, *Developing Integrated Web and Database Applications Using JAVA Applets and JDBC Drivers*. Proceedings of the 29th SIGCSE Technical Symposium on Computer Science Education, 1998: p. 302-306.
4. Rob, P., *Database Systems: Design, Implementation, and Management*. Third Edition ed. 1997: International Thomsom Publishing.
5. Gordon, S.R., *Factors That Affect the Adoption of Distributed Database Management Systems*. Journal of the ACM, 1993(April 1993): p. 151-167.
6. Owens, D.A. and F.T. Sheldon, *Tool-Based Approach to Distributed Database Design*. Proceedings of the 1999 ACM symposium on Applied Computing, 1999(1999): p. 227-231.
7. Bhargava, B. and A. Helal, *Efficient Availability Mechanisms in Distributed Database Design*. Proceedings of the Second International Conference on Information and Knowledge Management, 1993: p. 645-654.
8. Doherty, C.G., *Database Systems Management and Oracle8*. Proceedings of the 1998 ACM SIGMOD International Conference on the Management of Data, 1998: p. 510-511.
9. Wade, A.E., *Single Logical View over Enterprise-Wide Distributed Databases*. Proceedings of the 1993 ACM SIGMOD International Conference on the Management of Data, 1993: p. 441-444.
10. Tucson Police Department, *COPLINK: Database integration and access for a law enforcement intranet.*, . 1997, Tucson Police Department: Tucson.
11. United States Department of Justice, *Uniform Crime Reporting: National Incident-Based Reporting System: Volume 1 Data Collection Guidelines*, . 1998.
12. Alhir, S.S., *UML in a Nutshell*. Nutshell Series, ed. A. Oram. 1998: O'Reilly.
13. Feuerstein, S., *Oracle Built-in Packages*. First Edition ed, ed. D. Russell. 1998: O'Reilly & Associates.
14. Feuerstein, S., *Oracle PL/SQL*. Second Edition ed, ed. D. Russell. 1995: O'Reilly & Associates.

# Attachment C

# COPLINK Concept Space: An Application for Criminal Intelligence Analysis

Roslin V. Hauck, Homa Atabakhsh, Pichai Ongvasith, Harsh Gupta, Hsinchun Chen

University of Arizona, Management Information Systems Dept., Artificial Intelligence Lab.
http://ai.bpa.arizona.edu/

Submitted for review to IEEE COMPUTER on December 21, 2000.

**Abstract**

Government agencies across the United States have begun to focus on innovative digital technologies to aid in knowledge management and intelligence analysis. In the domain of law enforcement agencies, the analysis of criminal information is often hampered by its knowledge-intensive and time-critical environment. This atmosphere fosters the need for intelligence tools to combat criminal activity by aiding in case investigation. Funded by the National Institute of Justice and the National Science Foundation, the University of Arizona's Artificial Intelligence Lab has teamed with the Tucson Police Department (TPD) to develop the COPLINK Concept Space application, which uncovers relationships between different types of information that exist in TPD's records management system. In this paper, we present the technology behind the COPLINK Concept Space as well as its usage in real life criminal investigation activities. Future directions of the COPLINK project and development of other advanced technologies for law enforcement are also discussed.

**Keywords:** Law Enforcement, Information Systems, Knowledge Management, Information Retrieval, Intelligence Analysis, Information Sharing

## 1. Introduction

### 1.1 Law Enforcement Intelligence Analysis and Knowledge Management

In this era of the Internet and distributed multimedia computing, new and emerging classes of information technologies have swept into all areas of business, industry and government. As information technologies and applications become more overwhelming, pressing, and diverse, persistent information technology problems have become even more urgent. *Information overload*, a result of the ease of information creation and rendering via Internet, the WWW, and organizational data sources, has become more evident in people's lives[1]. This phenomenon is nowhere more evident than in government, specifically in criminal justice information systems. Federal, state, and local criminal justice entities possess vast repositories of information, but the explosive growth in digital information and the need for access within government agencies have made information overload increasingly significant.

Agencies' knowledge management problems frequently stem from barriers to access and utilization resulting from incompatible content and format of information[2] that make creation and utilization of knowledge management a complex and daunting process. Nevertheless, a number of different applications and approaches to knowledge management technologies are emerging, among them: virtual enterprising[3], joint ventures[4], aerospace engineering[2], and digital libraries[5].

Several government initiatives have been established to address some of the problems of the law enforcement sector of the digital government. The Office of Justice Programs (OJP) Integrated Justice Information Technology Initiative is involving five bureaus including the National Institute for Justice (NIJ) in an effort to use wired-information technologies to improve the effectiveness and fairness of the justice system through better information sharing. An NIJ wireless initiative, the AGILE program of the NIJ Office of Science and Technology primarily addresses interoperability issues (other government initiatives are described on http://www.ojp.usdoj.gov).

These government initiatives motivated a proposal to unite the technical expertise of the University of Arizona's Artificial Intelligence Lab with the law enforcement domain knowledge of Tucson Police Department to develop cutting edge technologies for the law enforcement community in the COPLINK project. This paper describes the COPLINK Concept Space application, one of those technologies which, although originally funded by NIJ, has received additional funding from both NIJ and the National Science Foundation (NSF) under its Digital Government Initiative. The Artificial Intelligence Lab has gained wide recognition as a cutting-edge research unit and has been featured in *Science* and *The New York Times,* having received more than $9M in research funding from various federal and industrial sponsors since 1989. The Lab sees COPLINK as an opportunity to serve the community by bridging the gap between research in developing technologies and solving such real-world problems as helping police officers fight crime.

### 1.2 A Case Study: The Tucson Police Department

The Tucson Police Department (TPD) recently evaluated its information technology and identified problems of lack of information sharing, integration, and knowledge management. The department agreed to participate in research to investigate the potential of current state-of-the-art, near-term, and cost-effective database, Intranet, and multimedia technologies to make computer justice information database integration, management, and access more effective.

The COPLINK project attacks several problems existing in many law enforcement agencies, including TPD, by developing a model integrated system that allows law officers both within and between different agencies to access and share information. An additional goal of COPLINK is to develop consistent, intuitive and easy-to-use interfaces and applications that support specific and often complex law enforcement functions and tasks. Although the scope of this project includes a multilevel development plan incorporating different information technologies, the focus of the research reported here is on the improvement of criminal intelligence analysis. The first step in this process was the evaluation of TPD's current Records Management System (RMS).

#### *1.2.1 TPD's Records Management System (RMS)*

The main database at TPD is the Records Management System (RMS), which stores a wide variety of data, including criminal case information and incident information from calls for service recorded from the Department's Computer-Aided Dispatch (CAD) system. RMS is a text-based system that is accessed using VAX terminals stationed in many central headquarters offices as well as substations located around the city.

Similar to systems at many other law enforcement agencies, RMS has many problems pertaining to its interface, access to information and lack of knowledge management. Although users are able to search on name queries, location queries, vehicle queries, etc., they are not able to search multiple fields simultaneously. In addition, users of RMS complain that, depending on the type of query, RMS can take from a few minutes to a few hours to return its results.

#### *1.2.2 Current TPD Knowledge Management Practice*

A function of the daily routine of many crime analysts and detectives at TPD is to create knowledge from information by analyzing and generalizing current criminal records that consist of approximately 1.5 million criminal case reports containing details from criminal events dating back to 1986. Although investigators can access RMS to tie together information needed to solve cases and crimes, they must manually search for connections or relationships existing in the data. Combining information to create knowledge is often hampered by voluminous information examination of which requires exorbitant time and effort on the part of the investigator. Compounding this problem is the variability of individual investigator's ability to locate relevant information.

Potent intelligence tools can alleviate crime analysts' information overload, reduce information search time required for analysis of available criminal records, and advance the investigation of current cases. This paper introduces a knowledge management system that can provide the functionality of intelligence analysis that currently does not exist in the RMS system. This system is designed to serve as a knowledge tool that serves the same purpose as current knowledge practices, but systematically and robustly gives crime analysts and investigators power to explore the entire data set for relationships that may exist. Real-life context evaluation of our system, the COPLINK Concept Space, and future directions for the project are also discussed.

## 2. Literature Review: Intelligence Analysis for Criminal Data

The concept of intelligence analysis, a crucial process in many law enforcement agencies often is overlooked in terms of lack of explicit performance and training[6]. Analysts and investigators are often left to their own devices in performing the intelligence analysis function. Frequently, the information provided to an investigator is incomplete and fragmented in nature, making it more difficult for investigators to understand the relationships and connections between terms. Knowledge management also poses significant challenges to police agencies. In addition to being inherently difficult to manage, knowledge traditionally has been stored on paper or in the minds of people[7]. In law enforcement, knowledge about criminal activities or specific groups and individuals tends to be acquired by officers who work in specific geographic areas. With the advent of criminal database systems that can store vast amounts of information, the task of intelligence analysis supported by insufficiently refined technology is even more daunting.

A number of current applications take advantage of various information technologies to assist law enforcement. As the number of agencies that utilize these types of technologies grows, the databases, intelligence analysis and other technologies have yet to be fully explored[8].

A number of systems currently serve as information management or intelligence analysis tools for law enforcement, among them:

- Use of visualization and time analysis to examine information. For example, the Timeline Analysis System (TAS) can help analysts visually examine large amounts of information by illustrating cause-and-effect relationships. This system graphically depicts relationships found in the data, resulting in trends or patterns[9]. Although beneficially applied to large-scale and statistical data, it is not appropriate for small and individual case analysis.
- Future ALert COntact Network (FALCON) is a problem-prevention or early-warning system developed at the University of North Carolina at Charlotte. This system assimilates a request, monitors all incoming records based on the request and then notifies the officer by email or pager when the request is met. Although not applicable as a criminal analysis tool, Falcon has much potential as a warning system.
- Expert systems that employ rule-based information have also been developed to assist in knowledge-intensive activities[10]. These systems attempt to aid in information retrieval by drawing upon human heuristics or rules and procedures to investigate tasks. As with all expert systems, however, identification of rules that are applicable in all cases presents a serious problem.

Recognizing a need in the law enforcement community for the development of a knowledge management system for criminal intelligence analysis that was scalable and adaptable, we undertook an effort to create the COPLINK Concept Space.

## 3. The COPLINK Concept Space

Concept space, or automatic thesaurus, is a statistic-based, algorithmic technique used to identify relationships between objects (terms or concepts) of interest[11]. The technique is frequently used to develop domain-specific knowledge structures for digital library applications.

In the University of Arizona Artificial Intelligence Lab, the idea of concept space was generated to facilitate semantic retrieval of information. Several user studies showed concept space to improve searching and browsing in the engineering and biomedicine domains. h the biosciences, the concept space approach was applied to the Worm Community System (WCS) and the FlyBase system. There also have been successful results in the Digital Library Initiative studies conducted on the INSPEC collection for computer science and engineering and on Internet searching[5,12]. Current on-going concept space research is being conducted in geographical information systems, law enforcement, and medicine.

A concept space is a network of terms and weighted associations that represent the concepts and their associations within an underlying information space that can assist in concept-based information retrieval. In addition, co-occurrence analysis uses similarity and clustering functions[13] to weight relationships between all possible pairs of concepts. The resulting network-like concept space holds all possible associations between objects, which means that every existing link between every pair of concepts is retained and ranked.

In COPLINK, detailed case reports are the underlying space and concepts are meaningful terms occurring in each case. Concept Space provides the ability to easily identify relevant terms and their degree of relationship to the search term. The relevant terms can be ranked in the order of their degree of association so that the most relevant terms are distinguished from inconsequential terms. From a crime investigation standpoint, Concept Space can help investigators link known objects to other related objects that might contain useful information for further investigation. For instance, like people and vehicles related to a given suspect.

Information related to a suspect can direct an investigation to expand to the right direction, but a case report that reveals relationships among data in one particular case fails to capture those relationships from the entire database. In effect, investigators need to review all case reports related to a suspect, which may be a tedious task. In the COPLINK project, we introduce Concept Space as an alternative investigation tool that captures the relationships between objects in the entire database.

To date, we have successfully adopted our techniques to create a COPLINK Concept Space based on a collection of 1.5 million case reports from the current Tucson Police Department Records Management System. These cases span a time frame from 1986 to 1999 (the entire case record collection for the City of Tucson). Based on careful user requirement analysis, five entity fields from the database were deemed relevant for Concept Space analysis: Person, Organization, Location, Vehicle, and Incident type. The purpose of this tool is to discover relationships between and among different crime-related entities. It is important not only to know that there is a relationship, but also to know what each relationship is.

## 4. Applying the Concept Space Technique to Criminal Data

In general, there are three main steps in building a domain-specific Concept Space. The first task is to identify collections of documents in a specific subject domain; these are the sources of terms or concepts. For Tucson Police Department, we are using the case reports in the existing database. The next step is to filter and index the terms. The final step is to perform a co-occurrence analysis to capture the relationships among indexed terms. The resulting Concept Space is then inserted into a database for easy manipulation (for a more in-depth analysis of the Concept Space algorithm, see Chen and Lynch[13]). The last two steps have been customized for COPLINK. After optimizing the code and tuning the database, we found that the total time required for building a COPLINK Concept Space is approximately five hours, which is acceptable in the given situation.

### 4.1 Term filtering and indexing

Due to the nature of the data residing in TPD's database, each piece of information is categorized in case reports and stored in well-organized structures. Theoretically, concept space can contain any number of term types (e.g., person names, organizations, locations, crime types, etc.). In practice, however, the size of the database, the time required to build a Concept Space, and the response time of queries are major constraints that limit the number of term types. To balance performance and comprehensiveness, a Concept Space should contain only meaningful types frequently searched by users. With the collaboration of personnel from the Tucson Police Department, we identified and created a set of term types for the COPLINK Concept Space.

Term types in Concept Space can be divided into five main categories. For a Person, Organization, Location, and Incident type, only one piece of information, such as a person's full name, street address, or crime type, is descriptive enough to be a search term. On the other hand, for a Vehicle, one piece of information, such as color, make or type, typically is comparatively common and when used as a search term would generate a large number of relevant terms. This problem can be avoided by combining two or more non-specific terms into composite terms.

The index maintains the relationship between a term and the document in which it occurs. Both index and reverse index are required for co-occurrence analysis. The index contains the links from term to document; the reverse index contains the links from document to term.

### 4.2 Co-occurrence Analysis

After identifying terms, we first computed the term frequency and the document frequency for each term in a document, based on the methodology developed by Chen [13]. Term frequency, *tf*, represents the number of occurrences of term j in document i. Document frequency, *df*, represents the number of documents in a collection of *n* documents in which term *j* occurs.

We then computed the combined weight of term *j* in document *i*, $d_{ij}$, based on the product of "term frequency" and "inverse document frequency" as follows:

$$d_{ij} = tf_{ij} \times \log(\frac{N}{df_j} \times w_j)$$

where *N* represents the total number of documents in a collection and $w_j$ represents the weight of words in descriptor *j*. In general, some term types are more descriptive and more important than others and deserve to be assigned higher weights so as to ensure that relationships associated with these types are always ranked reasonably. In COPLINK Concept Space, crime types are assigned comparatively higher weights.

We then performed term co-occurrence analysis based on the asymmetric "Cluster Function" developed by Chen and Lynch[13].

$$W_{jk} = \frac{\sum_{i=1}^{n} d_{ijk}}{\sum_{i=1}^{n} d_{ij}} \times WeightingFactor(k)$$

$$W_{kj} = \frac{\sum_{i=1}^{n} d_{ikj}}{\sum_{i=1}^{n} d_{ik}} \times WeightingFactor(j)$$

$W_{jk}$ indicates the similarity weights from term j to term $k$ and $W_{kj}$ indicates the similarity weights from term $k$ to term $j$. $d_{ij}$ and $d_{ik}$ were calculated based on the equation in the previous step. $d_{ijk}$ and $d_{ikj}$ represent the combined weight of both descriptors $j$ and $k$ in document $i$. However, they were computed slightly differently due to their different starting terms. They are defined as follows:

$$d_{ijk} = tf_{ijk} \times \log(\frac{N}{df_{jk}} \times w_j)$$

$$d_{ikj} = tf_{ijk} \times \log(\frac{N}{df_{jk}} \times w_k)$$

where $tf_{ijk}$ represents the number of occurrences of both term $j$ and term $k$ in document $i$ (the smaller number of occurrences between the terms was chosen); $df_{jk}$ represents the number of documents (in a collection of $N$ documents) in which terms $j$ and $k$ occur together.

In order to penalize general terms (terms which appeared in many places) in the co-occurrence analysis, we developed the following weighting scheme, which is similar to the inverse document frequency function.

$$WeightingFactor(k) = \frac{\log \frac{N}{df_k}}{\log N}$$

$$WeightingFactor(j) = \frac{\log \frac{N}{df_j}}{\log N}$$

Terms with a higher $df_k$ or $df_j$ value (more general terms) had a smaller weighting factor value, which caused the co-occurrence probability to become smaller. In effect, general terms were pushed down in the co-occurrence table (terms in the co-occurrence table were presented in reverse probabilistic order, with more relevant terms appearing first.)

Significant research needs to be conducted to investigate using Concept Space with our proposed noun phrasing and entity extraction techniques. In the above example, entity types from database fields were identified manually by human analysts. In addition, the Tucson Police Department does not yet capture free text narratives. Many law enforcement agencies have begun to incorporate content-rich narratives in their record management systems (e.g., Phoenix Police Department has complete narratives about each case). These narratives will provide a fertile test bed for combining noun phrasing and Concept Space analysis for intelligence identification.

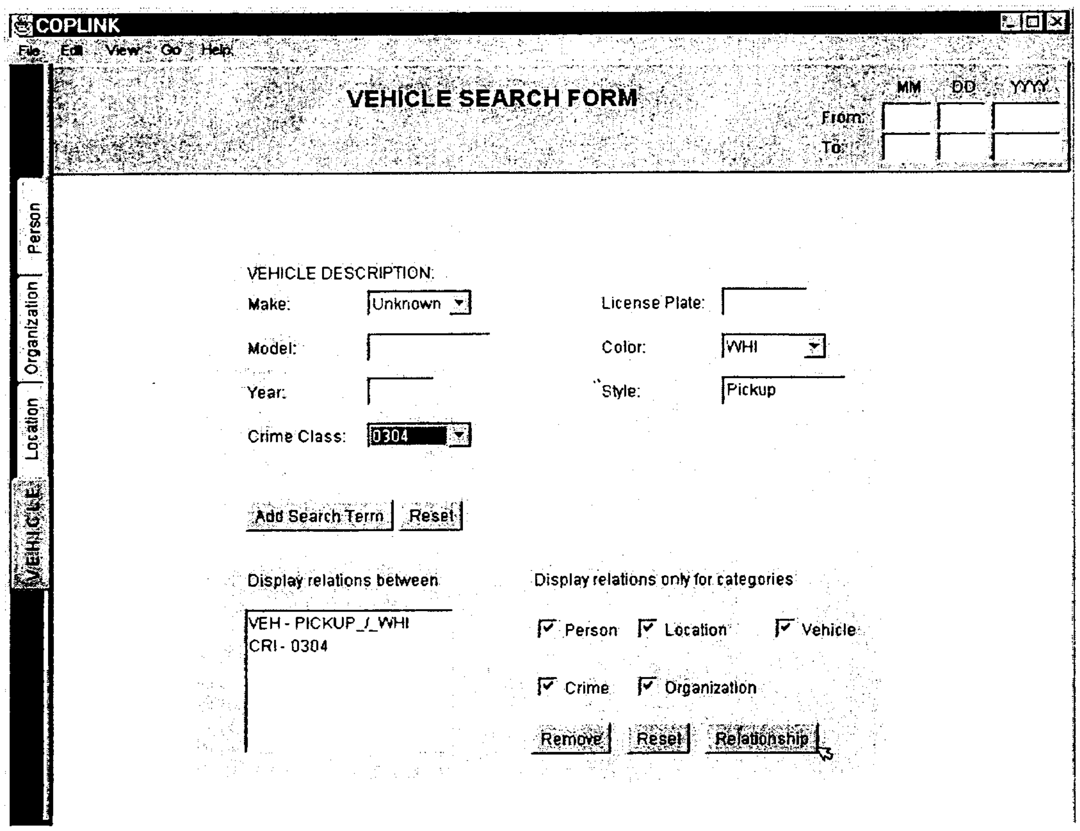
## 5. Graphical User Interface for COPLINK Concept Space

The graphical user interface for the COPLINK Concept Space Application is shown in Figures 2-4 (actual information has been altered to maintain data confidentiality). Search terms can be entered from any of the four search forms namely *Person, Organization, Location, and Vehicle.* For two or more search terms, each search term can be typed in the relevant search forms and can be added to the list through the *'Add'* button. The list of search terms is displayed in *'display relationships between'* box.

Relationships displayed between the entered search terms are organized by the five categories namely: *Person, Organization, Location, Vehicle, and Crime Type.* The Concept Space Application is also capable of displaying case reports with detailed information including case number, team beat, crime type etc. To illustrate the interface and usage of the Coplink CS system, the following is a possible scenario for an officer's or analyst's investigation of a crime.

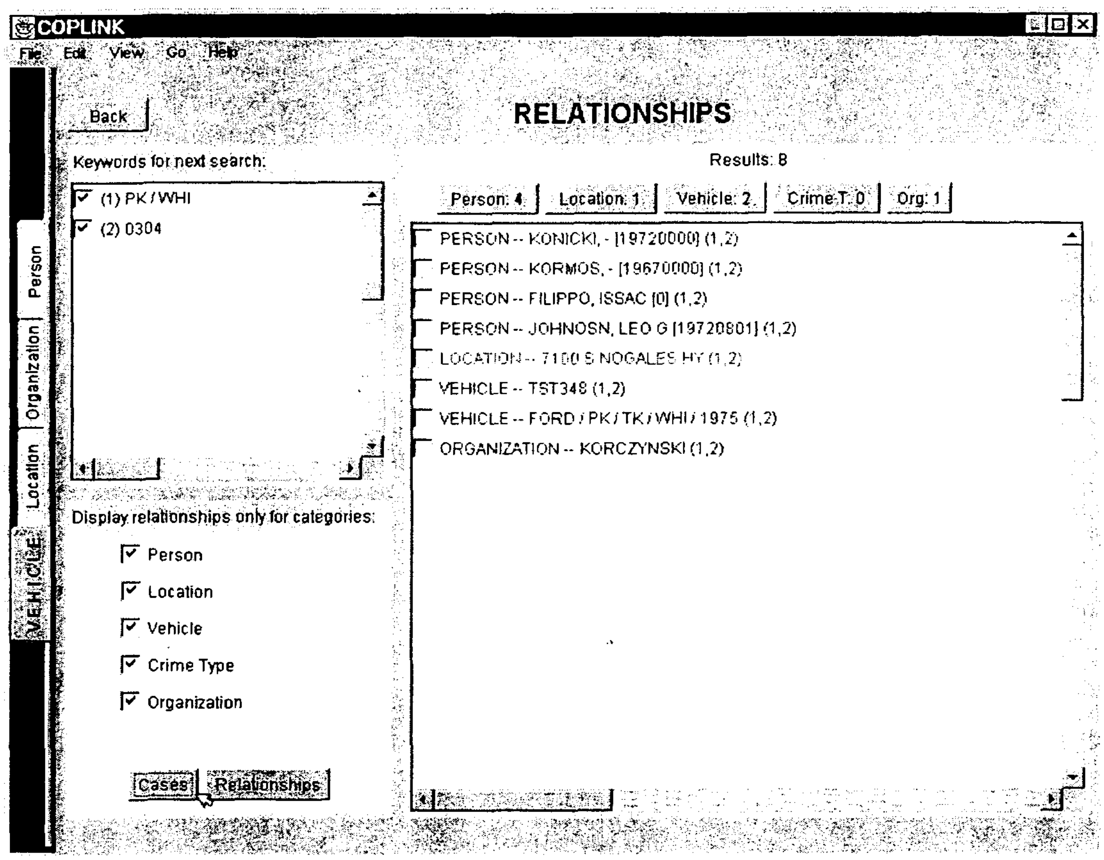
*Scenario: A detective is investigating a robbery at a local convenience store. The only witness, the night store clerk, only remembers that the suspect drove away in a white pickup truck.*

---



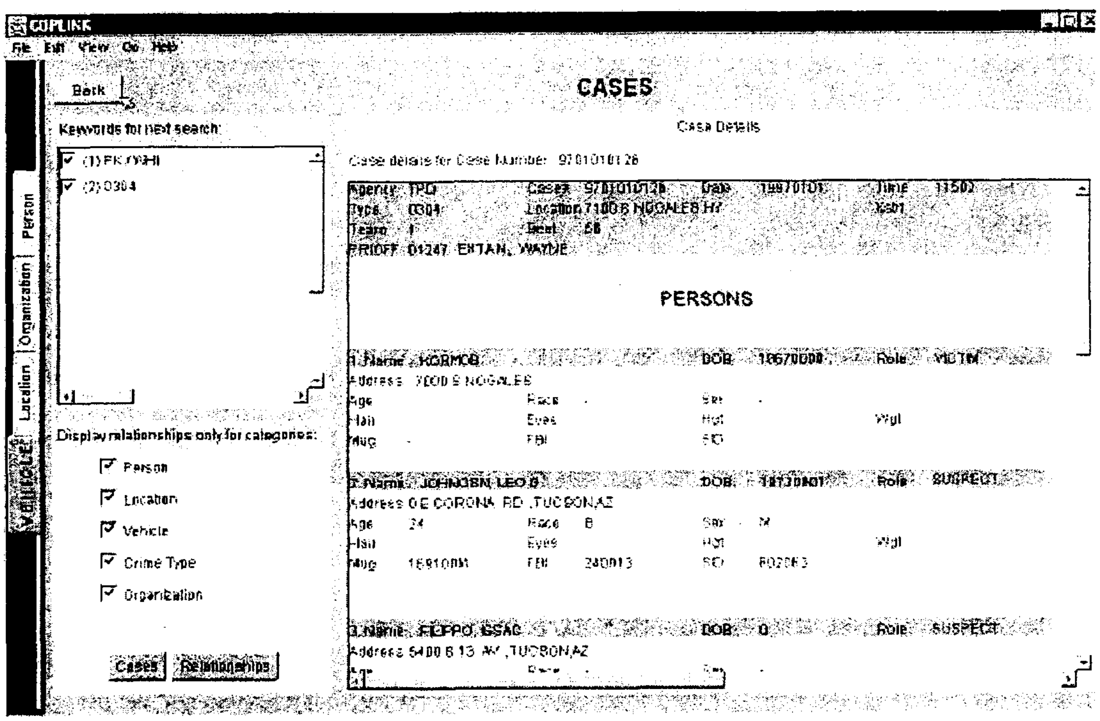
## Figure 2: COPLINK Concept Space Search Screen

Using COPLINK Concept Space, users are able to enter one of four information types as a search term. In this example, the detective needs to generate a lead, given the type of crime and the use of a white pick-up truck. The detective selects the Vehicle search screen and enters "White" for color, "Pickup" for style, and "0304", the universal crime report classification code for robbery of a convenience store. After adding the search terms to the relations box, the detective selects the Relationship button to enter the Concept Space. Note that the user can choose to select or deselect the types of relations returned by the system. This allows the user to choose only relevant categories and control for information overload.




## Figure 3: COPLINK Concept Space Result Screen

The system returns eight terms related to both a white pick-up and the 0304 crime type. Note that the Concept Space can return elements for each of the five information object types. The detective now knows not only that are there four people somehow related to both this type of crime and vehicle, she also has a license plate number for a vehicle. The detective can always add any of the Concept Space terms to the search or remove one of the two keywords from the search. As on the initial search screen, the panel in the lower left-hand corner allows users to control the amount of information returned by the Concept Space. The detective decides to view the cases that underlie the relationships uncovered by the Concept Space.




## Figure 4: COPLINK Concept Space Case Details Screen.

The Cases view displays actual case reports; in this example, there is only one case in the database. The detective can view the details of the prior incident, including the role of each person involved, and their home addresses. At any time, the detective can choose "Back" to review previous screens or modify the search keywords by selecting another type of search term or deselecting the current search terms.

## 7. User Evaluations for the COPLINK Concept Space

We conducted user evaluations to examine the effects of COPLINK CS on law enforcement investigation and work practices[8]. Twelve crime analysts and detectives, participated in the four-week longitudinal evaluation, during which they were asked to complete journal entries on searches they had conducted using COPLINK CS. By utilizing data collection methods of documentation, structured interviews, and direct observation, we were able to evaluate the function and design of the COPLINK CS system.

The journals and interviews revealed three major areas in which COPLINK Concept Space provided support for intelligence analysis and knowledge management.

### 7.1 Link analysis and Summarization

Participants indicated that Concept Space served as a powerful tool for acquiring information and cited its ability to determine the presence or absence of links between people, places, vehicles and other object types as invaluable in investigating a case[14]. The impact of link analysis on investigative tasks is crucial to the building of cases. An officer assigned to investigate a crime has to have enough information to provide a lead before he/she can begin working. Too many cases have to be closed because of lack of information or inability to utilize information existing elsewhere in the records management system. Concept Space manages all the data in the records system in such a way that it can be used as knowledge about the suspect. Link analysis can represent one of three types: directly linking known information, indirectly linking known information, and linking unknown information. Participants also reported they could use a concept space as a summary of the different information types related to a search term.

### 7.2 Interface Design

In general, users reported that the web-based interface of the COPLINK Concept Space was engaging and quite easy to use. Officers said the use of color to distinguish different object types and a graphic user interface provided a more intuitive tool than the text-based RMS system. Additionally, the ability to have results returned as either the concept space or case details allowed them to specify the type of information they needed. Participants reported that the data fields chosen for the Concept Space embody the basic necessary information for an investigation. They also reported that the separation between different fields in the output effectively encouraged easy comprehension of the information. A criminal investigation usually requires officers to make specific connections between people, places, vehicles, etc. in order to build a complete picture. The ability to aggregate information fields for searching provides a potent tool for problem solving and crime investigation.

### 7.3 Efficiency

Perhaps one of the most crucial benefits of the use of COPLINK Concept Space in law enforcement is its speed. As one of our participants explained, identifying a suspect between 48 to 72 hours after a crime is difficult. Beyond this time frame, a suspect is able to destroy evidence that may tie him/her to the crime or change his/her appearance to avoid identification. Witness/victim memory of the suspect's appearance also fades within this period. Identification of the suspect ideally should occur within 48 hours of the crime, so establishing useful links for identifying and locating the suspect is a crucial step. A number of interview and journal comments indicated that use of COPLINK Concept Space increased productivity by reducing time spent per information search.

In journals and interview sessions, each participant was asked to report the time it took to complete at least one particular search task using both RMS and COPLINK CS. The data indicated that in direct comparison of 15 searches, use of COPLINK Concept Space required an average of about 30 minutes less per search than with the use of RMS. However, review of other qualitative data from the journals and interviews indicated that subjects perceived much quicker response to a query from COPLINK CS, especially when multiple search entries and query expansion were involved.

*7.3.1 Multiple Search Entries*

The COPLINK Concept Space allows entry of multiple search terms, whereas this search capability is not possible in the current RMS system that forces an officer to conduct a number of single searches, then manually compare them. This can take a few hours of work to accomplish what Concept Space can do within seconds, as was the case for searches demonstrating large differences between Concept Space and RMS reported times.

*7.3.2. Query Expansion*

Users are able to add to the search any terms returned from the concept space. Point-and-click action to add any number of search terms allows users to expand searches quickly and easily and allows officers to explore more searches in a shorter amount to time. In addition, users can view concept spaces or documents on terms returned from previous searches without having to actually type in the query.

During user evaluation, we also looked at application of COPLINK CS to real-life crimes. An example is the real-life case of a shooting reported to us by a Tucson Police Department Crime Analyst:

*"The Tucson Police Department had responded to a shooting with the victim in critical condition. Although there were no witnesses at the crime scene, an anonymous caller contacted the police with some information. The caller did not know the shooter's name, however he did know that the shooter had a sister (name unknown) who was a victim in a domestic violence case. The caller was able to provide the identity of the arrestee in the domestic violence (DV) case.*

*Without the COPLINK Concept Space, success for this type of search would be difficult. I would have to query the DV arrestee in the system, pull up each case to see if it is a DV case; if so enter the case to view the people involved. If I found possible females, I would then have to repeat the process for each of their names. Depending on the number of cases each person is involved in, this manual process could easily take a few days to complete.*

*Using the COPLINK Concept Space, I entered the DV arrestee's name and the crime code for a domestic violence incident and searched the Concept Space. In a few seconds, I was looking at a list of people associated with the DV arrestee. I located a woman who was a victim in a DV case and ran the Concept Space on her. Sure enough, I found another male who was associated with her in a prior case. Checking his background and previous cases, I found out that he was indeed her brother and was a likely suspect in the shooting. The entire process using COPLINK Concept Space took about five minutes. It is very rare that with such limited information, we are able to generate a probable lead. COPLINK Concept Space will definitely help us to develop leads in even the most difficult of cases."*

Currently we are completing another session of user evaluations to validate the most recent update of the interface in preparation for deployment of the Coplink Concept Space to approximately 300 crime analysts and detectives by November 2000.

For more information on user evaluations for the COPLINK project, refer to [8].

## 8. Future Directions for COPLINK

Criminals are creatures of habit and being able to understand their habits and close associations is important[15]. The COPLINK Concept Space takes advantage of this characteristic by capturing connections between people, places, events, and vehicles, based on past crimes. Our initial evaluation of this intelligence analysis application supports its potential for transforming law-enforcement knowledge management practices in this age of digital governments. We have also designed the COPLINK Concept Space to be a scalable and powerful tool for other federal and local law enforcement agencies. In addition to the Concept Space, we are currently developing a number of other technologies for the law enforcement community.

Large collections of unstructured text as well as structured case-report information exist in police records systems. These textual sources contain volumes of information for investigators that are often not captured in the structured fields. One future research direction is to explore the development of textual mining approaches that support knowledge retrieval from such sources for law enforcement case reports. In order to perform a fine-grained content analysis, we will investigate the development of linguistic analysis and textual mining techniques that make intelligent use of large textual collections in police databases.

Several Internet research projects have shown the power of a new "agent" based search paradigm. In addition to supporting conventional searches performed by users, search agents allow users automatically to establish search profiles (or create profiles for users) and extract, summarize, and present timely information content. We believe such a proactive search agent is well suited to use by investigative personnel in law enforcement agencies. Search agents for law enforcement can support conventional searching techniques and be profiled for specific investigations. We plan to develop a personalized law enforcement search agent that will support wide expansion in connectivity and information sharing between police agencies.

In relation to the COPLINK project, the concept of a distributed database system has important implications. The most important of these is accessibility to and dissemination of law enforcement records and information. Currently, the vast majority of criminal data collection and compilation is done on a community level but may not be in a format that is readily available and accessible to local law enforcement officers. A distributed COPLINK prototype is under development using three COPLINK database servers to simulate the independent nodes in a distributed environment. Work is under way to include functionality that will provide interoperability among the different DBMS platforms that may support future COPLINK nodes. In the immediate future, we plan to begin deployment and testing of a Distributed COPLINK prototype with the Tucson and Phoenix police departments.

As distributed solutions and analysis tools are developed for law enforcement officers, a specific focus must be on providing tools within the constraints of a wireless environment. One of our future goals is to develop and refine applications to support the expansion of distributed and mobile law enforcement networks and inter-jurisdictional information retrieval as well as to investigate and study network security issues.

**Acknowledgments**

This project has primarily been funded by a grant from the National Institute of Justice, Office of Science and Technology (#97-LB-VX-K023). We would also like to thank the Digital Equipment Corporation External Technology Grants Program, agreement #US-1998004, for its award of a equipment grant allowance toward the purchase of a DecAlpha Server for the COPLINK Project. Appreciation also goes to Sgt. Jennifer Schroeder, Officer Linda Ridgeway and all the other personnel from the Tucson Police Department who were involved in this project.

**References**

1. D. C. Blair and M. E. Maron, "An Evaluation of Retrieval Effectiveness for a Full-Text Document-Retrieval System," *Communications of the ACM*, Vol. 28, No. 3, 1985, pp. 289-299.
2. P. Jones and J. Jordan, "Knowledge Orientations and Team Effectiveness," *International Journal of Technology Management*, Vol. 16, No. (1-3), 1998, pp. 152-161.
3. Y. M. Chen, C. C. Liao, and B. Prasad, "A Systematic Approach of Virtual Enterprising Through Knowledge Management Techniques," *Concurrent Engineering-Research and Applications*, Vol. 6, No. 3, 1998, pp. 225-244.
4. A. C. Inkpen, and A. Dinur, "Knowledge Management Processes and Joint International Ventures," *Organization Science* Vol. 9, No. 4, 1998 , pp.454-468.
5. H. Chen, "Artificial Intelligence Techniques for Emerging Information Systems Applications: Trailblazing Path to Semantic Interoperability," *Journal of the American Society for Information Science,* 1998, Vol. 49, No. 7, pp. 579-581.
6. J.J. Dintino and F. T. Martens, "An Enduring Enigma: The Concept of Intelligence Analysis," *The Police Chief,* Vol. 48, No. 12, Dec. 1981, pp. 58-64.
7. T. H. Davenport, "Business Process Reengineering: Where It's Been,Where It's Going," In V. Grover and W. Kettinger (Eds.), *Business Process Change: Reengineering Concepts, Methods and Technologies,* Idea Publishing, Middletown, PA, 1995, pp. 1-13.
8. R. V. Hauck and H. Chen, "COPLINK: A Case of Intelligent Analysis and Knowledge Management," *Proceedings of the 20th Annual International Conference on Information Systems,* 1999, pp. 15-28.
9. L. Pliant, "High-Technology Solutions," *The Police Chief*, 1996, Vol. 5, No. 38, pp. 38-51.
10. J. W. Brahan, K. P. Lam, H. Chan, and W. Leung, "AICAMS: Artificial Intelligence Crime Analysis and Management System," *Knowledge-Based Systems,* Vol. 11, 1998, pp. 355-361.
11. M. Lesk, *Practical Digital Libraries*, Morgan Kauffmann, Los Altos, CA, 1997.
12. H. Chen, and D. T. Ng, "An Algorithmic Approach to Concept Exploration in Large Knowledge Network (Automatic Thesaurus Consultation): Symbolic Branch-and-Bound vs. Connectionist Hopfield Net Activation," *Journal of the American Society of Information Science*, Vol. 46, No. 5, 1995, pp. 348-369.
13. H. Chen and K. J. Lynch, "Automatic Construction of Networks of Concepts Characterizing Document Database. IEEE Transaction on Systems," *Man and Cybernetics*, Vol. 22, No. 5, Sept./Oct. 1992, pp. 885-902.
14. W. R. Harper and D. H. Harris, "The Application of Link Analysis to Police Intelligence," *Human Factors,* 1975, Vol. 17, No. 2, pp. 157-164.
15. N. Joyce, "ICAM: Chicago's Newest Crime-Fighting Tool," *Proc. of the Conference in Technology, Community Policing.* National Law Enforcement and Corrections Technology Center, 1996, available at http://www.nlectc.org/txtfiles/confrpt.html.

# Attachment D

# COPLINK: A CASE OF INTELLIGENT ANALYSIS AND KNOWLEDGE MANAGEMENT

**Roslin V Hauck**
**Hsinchun Chen**
University of Arizona
U.S.A.

**Abstract**

Law enforcement agencies across the United States have begun to focus on innovative knowledge management technologies to aid in the analysis of criminal information. The use of such technologies can serve as intelligence tools to combat criminal activity by aiding in case investigation or even by predicting criminal activity. Funded by the National Institute of Justice, the University of Arizona's Artificial Intelligence Lab has teamed with the Tucson Police Department (TPD) to develop the Coplink Concept Space application, which serves to uncover relationships between different types of information currently existing in TPD's records management system. A small-scale field study involving real law enforcement personnel indicates that the use of Coplink Concept Space can reduce the time spent on the investigative task of linking criminal information as well as provide strong arguments for expanded development of similar knowledge management systems in support of law enforcement.

## 1. INTRODUCTION

The development of information technologies during the past few years has enabled many organizations to improve both the understanding and the dissemination of information. The development of powerful databases allows information to be organized in a manner that improves access to it, increases speed of retrieval, and expands searching flexibility. Furthermore, the Internet now provides a vehicle for the sharing of information across geographical distance that encourages collaboration between people and organizations.

Law enforcement agencies across the United States have begun to adopt innovative knowledge management technologies to aid in the management of criminal information. Such technologies can serve as intelligence tools to combat criminal activity by aiding in case investigation or even predicting criminal activity. In this research project we developed and evaluated one such knowledge management tool in the context of real-life criminal investigation by real law enforcers, fully acknowledging that many issues as well as obstacles must be addressed to ensure the successful deployment of this and similar information technologies.

### 1.1 Law Enforcement Technology Problems

Several issues combine to play a part in the utilization of information tools in law enforcement agencies.

#### *1.1.1 Access to Information*

Although much information exists, law enforcers often find it difficult to retrieve information from their sources. Because time can be a crucial factor in the completion of an investigation, ready access to information is critical. Obstacles to acquiring information in a timely manner can include restricted access to some systems for certain types of officers, or even long wait times for query returns. Although a detective may require information acquisition within three to 40 hours in real time, he or she may actually have to wait a few weeks to a month to receive data. Similarly, at many agencies, secure remote access to textual and multimedia databases is not currently available (Tucson Police Department 1997a).

### *1.1.2 Interface*

One important aspect of information technology for law enforcement is its ability to be used at the different levels within an organization. For example, support is needed for quick, street-level problems as well as for in-depth, lengthy investigations (Lingerfelt 1997). Given a vast range of functional needs and user abilities, an area of importance is the design of the interface. Although some departments are turning to use of graphical user interfaces, many local law enforcement agencies use text-based, front-end interfaces with their current database systems.. Navigation through these systems is often difficult and the users find the system commands counterintuitive. Despite the presence of much useful functionality, it seems that very few users are able actually to operate the functions. And because many interfaces are restricted to textual information, multimedia information such as mug shots and video clips cannot be incorporated and accessed.

### *1.1.3 Knowledge Management*

The general area of knowledge management (KM) has attracted an enormous amount of attention in recent years. Although it has been variously defined, it is evident that knowledge management exists at the enterprise level (see Davenport and Prusak 1998) and is quite distinct from mere information (e.g., see Davenport and Prusak 1998; Nonaka 1994; Teece 1998). Also apparent in this area are the challenges that knowledge management poses to an organization. In addition to being difficult to manage, knowledge traditionally has been stored on paper or in the minds of people (Davenport 1995; O'Leary 1998). The KM problems facing many firms stem from barriers to access and utilization resulting from the content and format of information (Jones and Jordan 1998; Rouse, Thomas and Boff 1998). These problems make knowledge management acquisition and interpretation a complex and daunting process. Nevertheless, knowledge management information technologies have been developed for a number of different applications, such as virtual enterprising (see Chen, Liao and Prasad 1998), joint ventures (see Inkpen and Dinur 1998), and aerospace engineering (see Jones and Jordan 1998).

The same problems of knowledge management exist at the specialized organizations of law enforcement. Many record management systems for law enforcement agencies contain a large amount of data for each case or incident, but although data may be available, they are not available in a form that makes them useful for higher level processing. For example, the ideal knowledge management system should be able to provide information about problems that have not been identified previously, and thus be able to give innovative and creative support for new investigations. The conversion of information to knowledge is an important concern for law enforcement agencies. Information is a product that is designed with a purpose in mind, while data serve as the ingredients in this product (Sparrow 1991). Furthermore, addressing the conversion of information to useful and easily understandable knowledge is a powerful aspect of knowledge management that has thus far been missing from most law enforcement information systems.

## 1.2 TPD IT Problem and Direction

The Tucson Police Department (TPD) recently evaluated the status of its information technology. Having concluded that all of the problems mentioned currently exist in the organization, the department agreed to participate in research to investigate the potential of current state-of-the-art, near-term, and cost-effective database, Intranet, and multimedia technologies to make computer justice information database integration, management, and access more effective (Tucson Police Department, 1997b). Although the scope of this project includes a multilevel development plan on different information technologies, the focus of the research reported here is on the improvement of criminal incident information retrieval. The first step in this process was the evaluation of TPD's current Records Management System (RMS).

## 1.3 TPD's Records Management System (RMS)

The main database at TPD is the Records Management System (RMS), which stores a wide variety of data, including criminal case information and incident information from calls for service recorded from the Department's Computer-Aided Dispatch (CAD) system. RMS is a text-based system that is accessed using VAX terminals stationed in many offices in the main headquarters as well as at many substations located around the city.

Similarly to systems described previously, RMS has many problems pertaining to its interface, access to information and lack of knowledge management. Although users are able to search on name queries, location queries, vehicle queries, etc., they are not able to search for multiple types of fields at one time. In addition, users of RMS complain that, depending on the type of query, RMS can take from a few minutes to a few hours to return its results.

## 1.4 Current TPD Knowledge Management Practice

A basic task for detectives and crime analysts at TPD is to create knowledge from information. In this case, information is made up of approximately 1.5 million criminal case reports, containing details from criminal events dating back to 1986. Tacit knowledge has also been described as the means through which new knowledge is generated (Nonaka and Takeuchi 1995) as well as the practical knowledge used to perform a task (Polanyi 1962). It is tacit knowledge that is used as investigators try to tie together information to solve cases and crimes. This ability to combine information to create knowledge is often hampered by the amount of information that exists.

The purpose of this paper is to explore the development of a knowledge management system that can provide the functionality of intelligence analysis that currently does not exist in the RMS system. This system is designed to serve as a type of knowledge tool that works toward the same purpose as current tacit knowledge practices of crime analysts and detectives and has been evaluated in a real life context. Its findings also are discussed.

# 2. LITERATURE REVIEW: USE OF IT AND AI IN LAW ENFORCEMENT

A number of applications that take advantage of various information technologies for law enforcement purposes currently exist. As the number of agencies that utilize these types of technologies is growing, the development of useful artificial intelligence tools continues to progress. And because there are many uses of databases, intelligence analysis and other technologies, the potential uses for these types of technologies have yet to be fully explored.

## 2.1 Database Technologies

Database technology plays an important role in the management of information for a police department. Previous research has detailed the use of database technology to allow for the organization of information in a form that can be easily searched by officers and other employees in a police department (Hoogeveen and van der Meer 1994; Lewis 1993; Lingerfelt 1997; Miller 1996; Schellenberg 1997; Wilcox 1997). The use of relational database systems for crime-specific cases, such as gang-related incidents, and serious crimes, such as homicide, aggravated assault, and sexual crimes, has proved to be highly effective (Fazlollahi and Gordon 1993; Pliant 1996; Wilcox 1997). The use of databases in these criminal areas is often targeted because it allows for a manageable amount of information to be entered into the database and, in addition, can combine information that may normally exist in neighboring police districts.

## 2.2 Intelligence Analysis for Criminal Data

Solving problems by analyzing and generalizing current criminal records is a function of the daily routine of many crime analysts and detectives. The amount of information that these investigators must analyze is often overwhelming, a phenomenon often referred to as "information overload" (Blair and Maron 1985). Potent intelligence tools can be useful in the analysis of available criminal records and aid in the investigation of current cases by alleviating the crime analysts' information overload and reducing information search time.

There are currently a number of systems that serve as intelligence analysis tools for law enforcement. Many technologies use neural networks to solve problems by developing associations between information objects and being trained to solve problems by comparing known objects with unknown objects. Some applications utilize visualization and time analysis to examine information. For example, the Timeline Analysis System (TAS) can help analysts visually examine large amounts of information by illustrating cause-and-effect relationships. This system graphically depicts relationships found in the data, resulting in trends or patterns (Pliant 1996). Expert systems that employ rule-based information have also been developed to assist in knowledge-intensive activities (Bowen 1994; Brahan et al. 1998). These systems attempt to aid in information retrieval by drawing upon human heuristics or rules and procedures to investigate tasks.

## 3. APPROACH

The University of Arizona has refined an intelligence analysis technique to help improve the organization and categorization of information, resulting in the reduction of user information overload and therefore in more efficient searching. This knowledge management approach strives to use information to create underlying connections and relationships that can lead to the generation of new knowledge. This technique is based on an automatically generated thesaurus or concept space.

### 3.1 Concept Space for Intelligence Analysis

The concept space algorithm automatically computes the strength of relationships between each possible pair of concept descriptors identified in a document collection. It is important to note that this concept is not a novel technique. For example, the use of similarity functions and ranking procedures for information retrieval was found to be both effective and efficient (Noreault, Koll and McGill 1977). For a more extensive review of our variation of this algorithm, see Chen and Lynch (1992), Chen and Ng (1995), and Chen et al. (1995). What makes this research different from previous work is that we have refined and applied the concept space to the specific data set and the information retrieval application to law enforcement. The process of creating the concept space for the Coplink application can be summarized as follows.

- *Document collection.* A collection of 1.5 million criminal-case reports from the current TPD records management system that span a time frame from 1990 to August 1997 constituted the document records in this analysis. From previous user requirement analysis, six information fields from the database were deemed relevant for co-occurrence analysis.

- *Co-occurrence analysis.* Co-occurrence analysis, a basic technique dating back to the 1960s (see Van Rijsbergen 1977), creates a concept space that is a graph of concepts. In addition, co-occurrence analysis uses similarity and clustering functions (Chen and Lynch 1992) to weight relationships between all possible pairs of concepts. This net-like concept space holds all possible associations between objects, which means that all existing links between every pair of concepts is retained and ranked.

- *Associative retrieval.* When a search term is entered into the concept space user interface, the system returns a list of co-occurred terms for user analysis. In the Coplink Concept Space, the associated terms are presented using multiple rank-ordered lists in a tabular format. The six tabular columns represent the six information fields used in the co-occurrence analysis. The use of a tabular format creates better summarization and visualization of the retrieved information by allowing officers to target the information field/type that they want.

### 3.2 System and Interface

In the application of concept space to the collection of TPD case records, a number of modifications were employed. Table 1 provides more detail on the concept space data analysis. An important modification was the identification of certain fields for analysis. This included both fields that can be used as search terms and fields that are returned by the system. The relationships between search terms are an important issue not focused upon in prior concept space applications. For an investigator, relationships between objects must not only be identified, but for crime analysis must also be explicitly understood. The format of the query is consequently yet another consideration. Officers may use the Coplink Concept Space in search for particular relationships, so their being able to request a specific output format is a valuable component of this application for law enforcement personnel.

**Table 1. Details on Concept Space Analysis**

| | |
|---|---|
| Size of Database | 1.5 M criminal records (528 MBytes) |
| Size of Resulting Concept Space | 1.24M terms (478 MBytes) |
| Number of Names | 644,143 terms |
| Number of Addresses | 210,003 terms |
| Number of Vehicles | 361,126 terms |
| Number of Organizations | 27,158 terms |
| Number of Weapons | 96 terms |
| Number of Crime Types | 719 terms |
| Processing Machine | DEC Alpha Server 4100 |
| Processing Time | 24 minutes |

### *3.2.1 Field Identification*

In criminal investigations, an officer can acquire information leads that fall into a number of different categories and work together to create the "story" that describes the crime. In the categorization of case reports, we chose to break down the search terms into six search objects: person, address, organization/business, vehicle, crime type, and weapon. These objects contain information that is currently being stored in fields of the records management system at TPD. A user can choose to enter up to four objects in any combination to begin a search.

The different objects contain specific elemental data. For example, the name object contains last name, first name, and middle initial. The majority of addresses use a street number, street name, street type (e.g., Rd, St, Av), and apartment number. The crime type identifies the type of crime committed in accordance to the standard FBI classification system that is used in the United States. Vehicle information contains a number of possible fields, including the make, model, type, year, color, and license plate number. Users are able to search on any combination of these elements and across different fields. By employing different search objects, officers are able to easily search by the specific type of information that he/she has available.

### *3.2.2 Relationship Identification*

The purpose of this search tool is to discover relationships between the different search terms or objects. It is not only important to know that there is a relationship, but it is also important to know what the relationship between objects is. Figures 1 through 3 illustrate a sample scenario using the concept space tool, which provides a detailed description of how relationships can be identified, given that the officer has a limited amount of information.[1]

*Scenario:* *Robbery at a local convenience store. Night store clerk only remembers that the suspect drove away in a white pickup truck.*

[1]Due to its sensitive nature, the actual information shown in these scenarios has been altered.

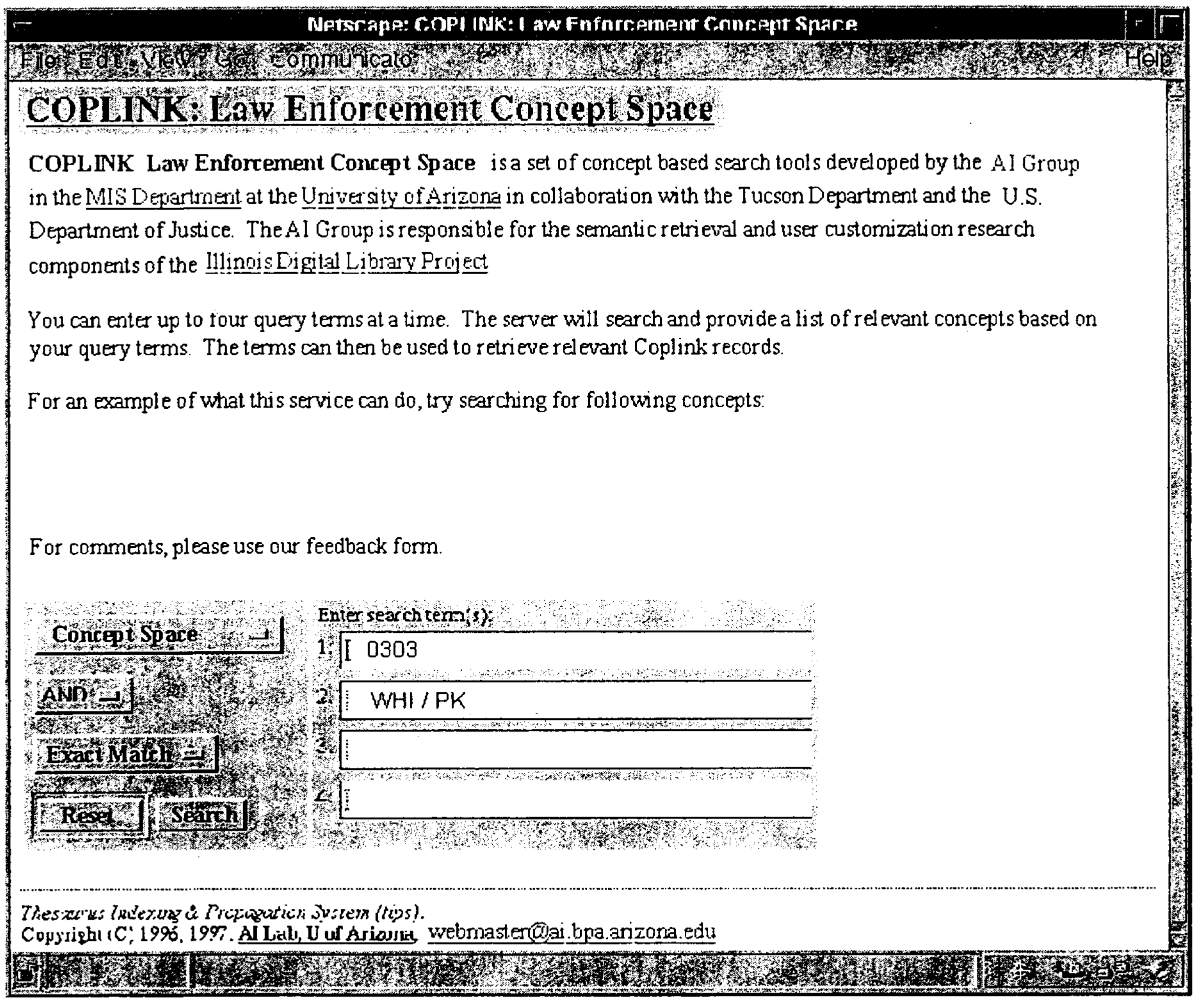



**Figure 1. Coplink Concept Space Input Screen**

Using Coplink Concept Space, officers are able to input any type of information object as a search term. For our scenario, the investigator can generate a lead given the type of crime and the use of a white pickup truck. This figure shows the input screen in which the investigator has entered the crime type 0303 (robbery of a service station) and white pickup truck.

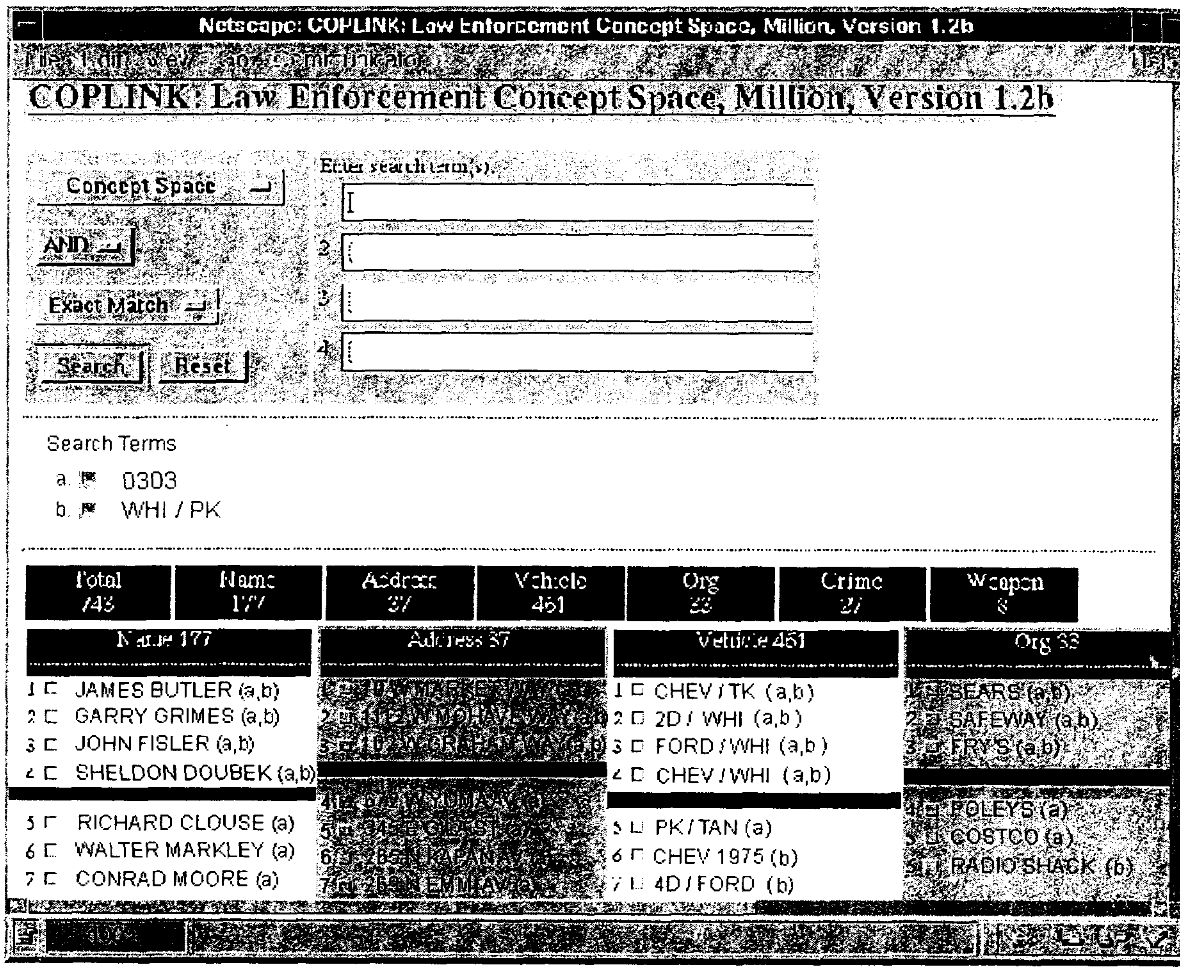



**Figure 2. Concept Space View**

After entering the two pieces of information, the Concept Space for the query is returned. Note that the Concept Space returns elements for each of the six information object types. Under the person type, the concept space has determined that there are entries that are associated with the crime type as well as the vehicle type. The resulting output contains lists for each of the search objects, all of which are somehow associated with the search terms entered by the user. In the output screen, the interface is designed such that each entry returned refers to the input object(s) with which it is associated. In addition, for queries that use multiple entries, the return entries for each object type that are associated with the greatest number of input search terms are those above the colored bar. The corresponding letter(s) in parentheses after each output term indicate the exact search object that was found to be associated with each output term. This allows the officer to understand how the output terms are related to each of the search terms entered.

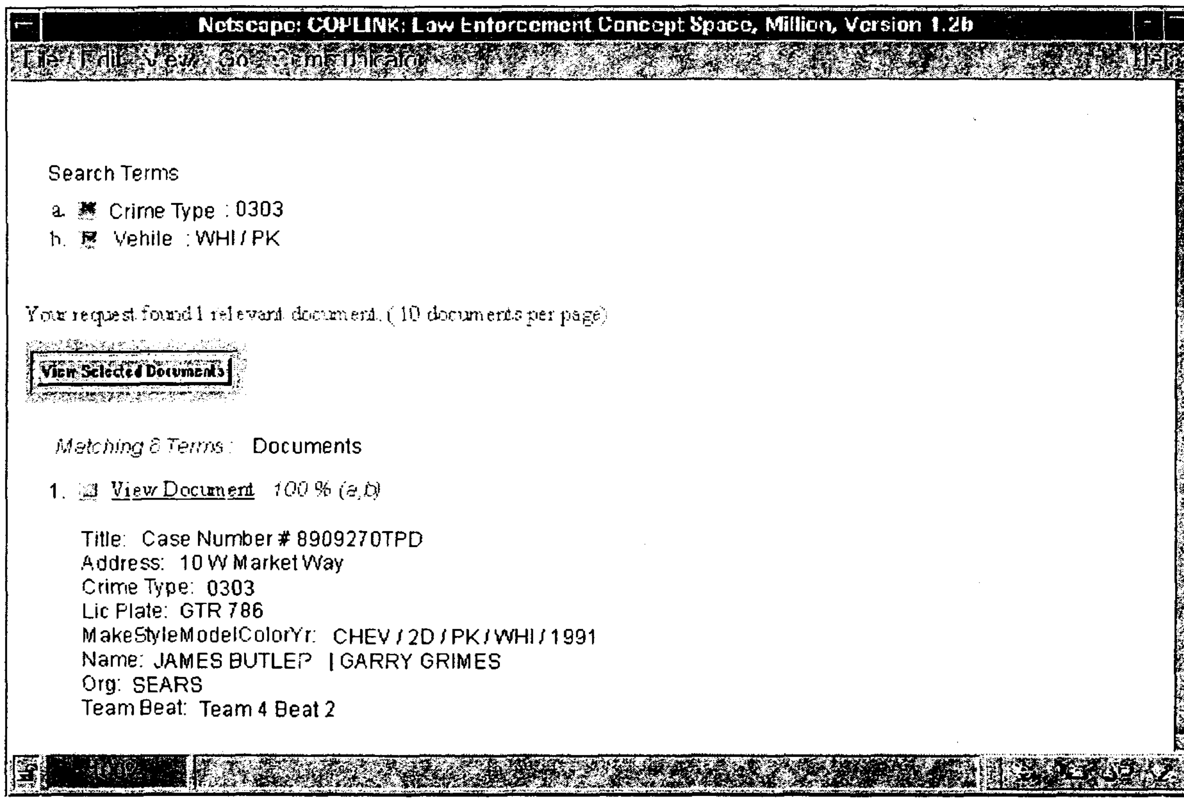



**Figure 3. Detailed Document View**

Wanting to know more about this similar past case, the investigator chooses to view more detailed documents about the prior crime that are associated with the entered information. These documents contain the specific case number information, all persons involved in the case, the crime type, the location of the incident, vehicles involved, and weapons information.

## 4. RESEARCH DESIGN

In order to evaluate the usefulness and the usability of the concept space in a law enforcement setting, we conducted a study to determine the feasibility of this intelligence analysis tool in real life criminal investigations.

### 4.1 Research Strategy: Field Experiment

The use of the field experiment research strategy allows researchers to engage in investigative activities that can uncover key issues that cannot be acquired in the unrealistic confines of a laboratory. The Coplink Concept Space (CS) lends itself to this method of inquiry in many different ways. The use of Coplink CS is environment dependent. In order to be able to increase the external validity of the findings, we need to be able to evaluate the system in real-life situations. Also, given the actual application of this tool in doing their jobs, we wanted to give the officers the ability to test the application's functionality on current tasks and cases. Finally in dealing with the law enforcement field, we were confronted with an environment where we were unable to command total control while demonstrating the value of the technology by aiding in the investigation of authentic police cases.

## 4.2 Experimental Design

The purpose of this research was to understand the effects that Coplink CS can have in law enforcement investigation and work practices. The interesting research questions that served as the focus for this study were:

*RQ1:* *Can Coplink CS lead to increased productivity?* One hypothesis was that Coplink CS has potential to increase productivity by decreasing the amount of time required per search session. Each participant was asked to do at least one direct comparison between RMS and Coplink CS; qualitative data analysis collected from the journals and interviews was used compare the amount of time spent on each system.

*RQ2:* *How is Coplink CS useful?* One goal of this study was to undercover the area or tasks in which Coplink CS, as a previously unavailable knowledge management tool, would be useful as well as tasks for which Coplink CS was not useful.

*RQ3:* *Where should we focus our efforts for future development of Coplink CS?* As a user-centered design effort, it is necessary that we continuously draw upon user evaluation to guide future endeavors. By taking into consideration feedback from the subjects, we hoped to establish and prioritize the course of action for continuing development of Coplink CS.

- *Subjects.* For this study, the specific group from TPD targeted to participate was made up of crime analysts, who investigate high-profile cases as well as create statistical reports on criminal activities. The analysts are the department's most technology savvy user group and are accustomed to using a number of different data sources. Eleven crime analysts and one homicide detective from TPD were asked to participate in the study. The detective was also experienced in using a number of different technologies.

- *Data collection method.* The data collection methods employed in this study included documentation, interviews, and direct observation by both the researcher and a TPD officer working on the Coplink project. Documentation consisted of journals kept by each subject detailing actual search experiences. In-depth, structured interviews and direct observations were utilized in one pretest and at multiple post-test sessions. Due to the difficulty of recording actual times of searches, participants' reported time spent per search was used in comparing times spent addressing RQ1 for both the Coplink CS and the RMS systems. In addition, qualitative data analysis was focused on thematic or pattern matching of findings and anecdotal data (RQ2 and RQ3).

- *Experimental procedures.* A longitudinal design was used to evaluate the Coplink Concept Space application. Prior to exposure to Coplink CS, initial background structured interviews were conducted with the participants. For one week after this session, participants were asked to keep a journal documenting their search experiences on their current records management system. At the conclusion of this week, participants underwent a brief demonstration of the Coplink Concept Space and a training session in its use, after which they were asked to briefly evaluate the functionality of the system as well as how it compared with their current records management system. After the participants were shown the basic functionality of the application and were able to work through a number of trials, they were asked to use Coplink CS for a four-week period. During this time period, participants were again asked to complete journal entries on searches that they conducted using Coplink CS, after which we concluded the study with final in-depth interviews.

## 5. RESULTS

The results of the TPD Concept Space study are quite supportive of its use in investigative law enforcement as a knowledge management application. In addition, results also uncovered a number of important issues that need to be addressed in future development efforts. Participants' logs of usage show that Coplink CS was utilized in 37 queries (732 minutes). A majority of the participants' feedback can be categorized into one of two general areas: task analysis and interface analysis. From a content analysis of journal logs and interview data, we were able to address each of the research questions underlying this study.

## 5.1 Task Analysis

From the journals and interviews, we were able to build a taxonomy of task types in which participants were engaged while using the Coplink Concept Space. Specifically addressing RQ2, the evaluation of tasks allows us to understand the strengths of Coplink Concept Space and the particular tasks that leverage those strengths.

### *5.1.1 Link Analysis*

Participants indicated that Concept Space serves as a powerful tool for acquiring information and mentioned its ability to determine the links between people, places, vehicles and other object types as invaluable in investigating a case. The impact of such link analysis upon investigative tasks is crucial to the building of cases. Assigned to investigate a crime, an officer can only hope to get enough information to provide a lead with which he/she can begin working. Too many cases have to be closed due to the lack of information or inability to tie together information data existing elsewhere in the records management system. Concept Space manages all the data in the records system in such a way that it can be used as knowledge that tells a story about the suspect. Link analysis can be described as being one of three types: directly linking known information, indirectly linking known information, and linking unknown information.

- *Directly linking known information.* Objects that appear together in at least one case record characterize a direct link. One type of link analysis performed by participants dealt with establishing a direct relationship between two known objects.

- *Indirectly linking known information.* An indirect link is a complex link between objects that exist in a number of incidents. Another way to perform a search is to enter two known objects and look at the concept space to determine whether any other objects relate to both of them. Although this is a more complex search, it allows the user to link objects indirectly. These objects are connected, but not within a single case report. This transitive relationship between objects is a powerful one for criminal analysis, because it allows officers to infer connections although explicit data connecting the objects may not exist in the current collection of case records.

- *Linking unknown information.* One important way in which Concept Space can assist in acquiring leads is by allowing officers to browse through information and establish relevant associations even though these links were previously unknown to the officer. Given that an officer has some initial search terms, the Coplink Concept Space returns a number of possible associated terms. The officer can then add any of the resulting terms to the search to browse for possible relationships between them. This browsing technique allows an officer to perform analytical queries on possible connections to establish a lead.

The creation of links and leads for police officers is the creation of knowledge from a set of information (i.e., case reports). The knowledge generated begins to connect actors, locations, and objects in potential criminal activities. This knowledge management activity allows for the possibility of branching information seeking activities in different directions. Although this process can be conducted manually, it is quite difficult for investigators to create indirect links, especially given the amount of information that exists.

### *5.1.2 Summarization Analysis*

Participants in this study also utilized the Coplink Concept Space application to quickly establish a brief summary of a particular object. Analysts would often enter an object and use tabular layout of information to quickly peruse the known entries for the subject. Because the output of the Concept Space spans all of the cases in which a subject is involved, analysts can escape the boundaries of searching within a case and can instead search all existing cases that involve a particular subject.

## 5.2 Interface Analysis

In the development of an interface, it was our goal to design Coplink CS in an intuitive manner that fosters interaction. In general, users felt that the web-based interface of the Coplink Concept Space was engaging and quite easy to use. The use of color to delineate the different object types and the use of browser navigation tools provided the officers with a more intuitive interface than the text-based RMS system. Additionally, the ability to have results returned as either the concept space or brief/detailed documents allows users to specify the type of information that they seek according to their need. Participants reported that the data fields chosen for the Concept Space embody the basic necessary information for an investigation. They also reported that the separation between different fields in the output was very effective in encouraging easy comprehension of the information. More specifically, a number of interface-specific comments emerged from the data collected from the interviews and journals that indicate that use of Coplink Concept Space can lead to increased productivity (RQ1) as defined by reduction in time spent per search.

### *5.2.1 Time Issue*

Perhaps one of the most crucial aspects of the use of Coplink Concept Space in law enforcement is its speed. As one of our participants explained, identifying a suspect between 48 to 72 hours after a crime is difficult. Beyond this time frame, a suspect is able to destroy evidence that may tie him/her to the crime or change his/her appearance to avoid identification. Witness/victim memory of the suspect's appearance also fades within this period. Identification of the suspect ideally should occur within 48 hours of the crime, making establishing useful links for identifying and locating the suspect a crucial step.

Through the journals and interview sessions, each participant was asked to report the time it took to complete at least one particular search task using both RMS and Coplink CS. Table 2 illustrates the reported times for each of the 15 searches using both systems. From participants' reports, we found that in direct comparison of 15 searches, use of Coplink Concept Space resulted in an average of about 30 minutes per search. A t-test analysis of the times reported indicated that, although the differences between search time for RMS and for Coplink Concept Space were not statistically significant (15%), review of other qualitative data from the journals and interviews indicated that subjects reported much quicker response to a query from Coplink CS. The reasons for this reported efficiency fall into two categories.

- *Multiple search entries*. The Coplink Concept Space allows for the entry of multiple search terms. In the current RMS system, this search capability is not possible, forcing an officer to conduct a number of single searches, then manually compare them. This adds at least a few more hours of work.[2] The Concept Space is able to conduct the same query utilizing multiple search terms within seconds.
- *Query expansion*. As discussed previously, users are able to add to the search any terms returned from the concept space. This point-and-click action to add any number of search terms allows users to expand searches quickly and easily. Being able to append terms to the search and rerunning the search allow officers to explore more searches in a shorter amount to time. In addition, users could also view concept spaces or documents on singular terms returned from previous searches without having to actually type in the query.

Table 2. Reported Time Breakdown (in Minutes) of 15 Searches Using Both RMS and Coplink Concept Space

| **Search #:** | *1* | *2* | *3* | *4* | *5* | *6* | *7* | *8* | *9* | *10* | *11* | *12* | *13* | *14* | *15* |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **RMS(minutes)** | 120 | 45 | 15 | 20 | 60 | 30 | 10 | 12 | 10 | 10 | 20 | 20 | 30 | 15 | 300 |
| **CS(minutes)** | 10 | 30 | 15 | 5 | 60 | 20 | 5 | 7 | 5 | 3 | 10 | 10 | 20 | 3 | 60 |

[2]This was indeed the case for the searches with a large difference in Concept Space and RMS reported time (i.e., search #1 and search #15 of Table 2).

### *5.2.2 Interface Layout*

As discussed before, the Coplink Concept Space is organized into different types of objects including people, locations, organizations, vehicles, crime types, and weapons. The ability of the system to search across these fields and return associated information from each field is a powerful tool that made it easier for an officer to search quickly for relevant fields. Because the Concept Space searches across all of the six fields, users were able to examine all of the fields simultaneously.

The use of a tabular format to display the resulting concept space allows users to search the relevant information fields quickly. In a criminal investigation, officers are usually seeking specific connections between certain people, places, vehicles, etc. that will enable them to build a complete picture of possible interconnections between objects. The ability to aggregate information fields for searching provides a potent tool for problem solving and crime investigation.

Perhaps the majority of problems encountered by participants in this study were related to the interface, particularly its query entry screen. In the prototype used, because the entry fields were not structured, subjects were required to enter queries in a prescribed format. These entry fields should be redesigned so that it is clear to users where and how to enter search terms. Although the output is returned in a tabular format, participants often reported that an overwhelming amount of information was returned from the system, especially when subjects were interested in only a particular type of information.

## 6. CONCLUSION AND FUTURE DIRECTIONS

From this pilot study, we conclude that the use of Coplink Concept Space as a knowledge management and intelligence analysis tool in a law enforcement environment is quite promising. An important aspect of the study is that it dealt with real criminal information, real cases and search tasks, and real crime analysts. In addition to providing an intuitive interface, the system's combination of different information types in its associations provides much information of value in the analysis of crimes. Data in a records management system is not useful if the system does not have the ability to pull together the different types of information and to present them in an understandable way. The Coplink Concept Space uses data and transforms them into intelligence that the officer can utilize. Criminals are creatures of habit and being able to understand their habits is an important issue (Joyce 1997). The Coplink Concept Space takes advantage of this characteristic by capturing connections between people, places, events, and vehicles, based on past crimes. As a knowledge management tool, Coplink CS serves to create new knowledge in the form of links between people, places and objects, which in turn results in possible leads for investigation.

From these findings we have been able to determine that Coplink Concept Space was useful to our participants and, furthermore, provided them with a valuable asset in performance of investigative tasks. These preliminary results also indicate that Coplink Concept Space can potentially lead to increased productivity by reducing the amount of time spent for data search. Finally, we have determined that additional development effort is required for redesign of the interface to enable users to be more readily able to interact with and understand the application.

It is evident from this study that the use of knowledge management applications, such as Coplink CS, can have a significant impact on law enforcement. In addition to identifying important functionalities that law officers would like to have in an intelligence analysis tool, this research demonstrates the potential value of knowledge management in law enforcement. Given the favorable results of our Coplink Concept Space study, we are currently redesigning the interface to address some of the usability problems uncovered by this evaluation, including redesigning of screens and improving sorting ability. Based on the experience gathered from this pilot study, we are currently planning to conduct a larger-scale experiment using an updated version of Coplink Concept Space, including a more in-depth analysis of current and potential knowledge management processes. We plan to continue this research effort by expanding it to include participants from different units and job classifications within the Tucson Police Department, while progressively improving the application design.

## 7. ACKNOWLEDGMENTS

This project was primarily funded by a grant from the National Institute of Justice Office of Science and Technology (Tucson Police Department, "Coplink: Database Integration and Access for a Law Enforcement Intranet," $1.2M, 1997-1999). We would also like to thank the Digital Equipment Corporation External Technology Grants Program, agreement #US-1998004, for its award of a $198,451 equipment grant allowance toward the purchase of a DecAlpha Server for the Coplink project.

We would like to thank all of the personnel from the Tucson Police Department who participated in this study, Officer Linda Ridgeway for her invaluable assistance in data collection, and Harsh Gupta for accomplishing the laborious task of modifying the screen shots used in this paper.

## 8. REFERENCES

Blair, D. C., and Maron, M. E. "An Evaluation of Retrieval Effectiveness for a Full-text Document-retrieval System," *Communications of the ACM* (28:3), 1985, pp. 289-299.

Bowen, J. E. "An Expert System for Police Investigators of Economic Crimes," *Expert Systems with Applications* (7:2), 1994, pp. 235-248.

Brahan, J. W.; Lam, K. P.; Chan, H.; and Leung, W. "AICAMS: Artificial Intelligence Crime Analysis and Management System," *Knowledge-Based Systems* (11), 1998), pp. 355-361.

Chen, H., and Lynch, K. J. "Automatic Construction of Networks of Concepts Characterizing Document Databases," *IEEE Transactions on Systems, Man and Cybernetics* (22:5), 1992, pp. 885-902.

Chen, H., and Ng, D. T. "An Algorithmic Approach to Concept Exploration in Large Knowledge Network (Automatic Thesaurus Consultation): Symbolic Branch-and-bound vs. Connectionist Hopfield Net Activation," *Journal of the American Society of Information Science* (46:5), 1995, pp. 348-369.

Chen, H.; Schatz, B. R.; Yim, T.; and Fye, D. "Automatic Thesaurus Generation for an Electronic Community System," *Journal of the American Society of Information Science* (46:3), 1995, pp. 175-193.

Chen, Y. M.; Liao, C. C.; and Prasad, B. "A Systematic Approach of Virtual Enterprising through Knowledge Management Techniques," *Concurrent Engineering-Research and Applications* (6:3), 1998, pp. 225-244.

Davenport, T. H. "Business Process Reengineering: Where It's Been, Where It's Going," in *Business Process Change: Reengineering Concepts, Methods and Technologies*, V. Grover and W. Kettinger (eds.), Middletown, PA: Idea Publishing1995, pp. 1-13.

Davenport, T. H., and Prusak, L. *Working Knowledge: How Organizations Manage What They Know*, Boston: Harvard Business School Press, 1998.

Fazlollahi, B., and Gordon, J. S. "CATCH: Computer Assisted Tracking of Criminal Histories System," *Interfaces* (23:2), 1993, pp. 51-62.

Hoogeveen, M. J., and van der Meer, K. "Integration of Information Retrieval and Database Management in Support of Multimedia Police Work," *Journal of Information Science* (20:2), 1994, pp. 79-87.

Inkpen, A. C., and Dinur, A. "Knowledge Management Processes and International Joint Ventures," *Organization Science* (9:4), 1998, pp. 454-468.

Jones, P., and Jordan, J. "Knowledge Orientations and Team Effectiveness," *International Journal of Technology Management* (16:1-3), 1998, pp. 152-161.

Joyce, N. "ICAM: Chicago's Newest Crime-Fighting Tool," in *Proceedings of the Conference on Technology for Community Policing*, National Law Enforcement and Corrections Technology Center, June 1997 (http://www.nlectc.org/txtfiles/confrpt.txt).

Lewis, C. "Police Information Technology," *GEC Review* (9:1), 1993, pp. 51-58.

Lingerfelt, J. "Technology as a Force Multiplier," in *Proceedings of the Conference on Technology for Community Policing*, National Law Enforcement and Corrections Technology Center, June 1997 (http://www.nlectc.org/txtfiles/confrpt.txt).

Miller, B. "Searchable Databases Help Missouri Solve Crime," *Government Technology* (9:8), 1996, pp. 18-19.

Nonaka, I. "A Dynamic Theory of Organizational Knowledge Creation," *Organization Science* (5:1), 1994, pp. 14-37.

Nonaka, I., and Takeuchi, H. *The Knowledge Creating Company*, Oxford, England: Oxford University Press, 1995.

Noreault, T.; Koll, M.; and McGill, M. J. "Automatic Ranked Output from Bollean Searches in SIRE," *Journal of the American Society for Information Science* (28:6), 1997, pp. 333-339.

O'Leary, D. E. "Enterprise Knowledge Management," *Computer* (31:3), 1998, pp. 54-62.

Pliant, L. "High-technology Solutions," *The Police Chief* (5:38), 1996, pp. 38-51.

Polanyi, M. *Knowledge and Being*, New York: Routledge, 1962.

Rouse, W. B.; Thomas, B. S.; and Boff, K. R. "Knowledge Maps for Knowledge Mining: Application to R&D/Technology Management," *IEEE Transaction on Systems, Man and Cybernetics: Part C- Applications and Reviews* (28:3), 1998, pp. 309-317.

Schellenberg, K. "Police Information Systems, Information Practices and Individual Privacy," *Canadian Public Policy–Analyse de Politiques* (23:1), 1997, pp. 23-39.

Sparrow, M. K. "Information Systems: A Help or Hindrance in the Evolution of Policing?" *The Police Chief* (54:4), 1991, pp. 26-44.

Teece, D. J. "Research Directions for Knowledge Management," *California Management Review* (40:3), 1998, pp. 289-292.

Tucson Police Department. *Coplink: Database Integration and Access for a Law Enforcement Intranet*, Tucson Police Department, 1997a.

Tucson Police Department. *Response to the National Institute of Justice Solicitation for Law Enforcement, Courts, and Corrections Technology Development, Implementation and Evaluation*, Tucson Police Department, 1997b.

Van Rijsbergen, C. J. "A Theoretical Basis for the Use of Co-occurrence Data in Information Retrieval," *Journal of Documentation* (33:2), 1977, pp. 106-119.

Wilcox, J. " IT-armed Officers Fight Gangs," *Government Computer News: State and Local* (3:12), 1997, pp. 19-20.

# Attachment E

# COPLINK: Information and Knowledge Management for Law Enforcement

Hsinchun Chen[a*], Roslin V. Hauck[a], Homa Atabakhsh[a], Harsh Gupta[a], Chris Boarman[a], Jennifer Schroeder[b], Linda Ridgeway[b]

[a]Dept. of Management Information Systems, University of Arizona, Tucson

[b]Tucson Police Department, Tucson, Arizona

## ABSTRACT

The problem of information and knowledge management in the knowledge intensive and time critical environment of law enforcement has posed an interesting problem for information technology professionals in the field. Coupled with this challenging environment are issues relating to the integration of multiple systems, each having different functionalities resulting in difficulty for the end user. COPLINK offers a cost-efficient way of web enabling stovepipe law enforcement information sharing systems by employing a model for allowing different police departments to more easily share data amongst themselves through an easy-to-use interface that integrates different data sources. The COPLINK project has two major components: COPLINK Database (DB) Application and COPLINK Concept Space (CS) Application. The COPLINK DB design facilitates retrieval of case details based on known information. COPLINK CS is an investigative tool that captures the relationships between objects (e.g., people, locations, vehicles, organizations, crime types) in the entire database allowing investigators and detectives to perform investigative associations and case analysis. This paper describes how we have applied the design criteria of platform independence, stability, scalability, and an intuitive graphical user interface to develop the COPLINK systems. Results of user evaluations that have been conducted on both applications to study the impact of COPLINK on law enforcement personnel. The COPLINK DB Application is currently being deployed at the Tucson Police Department and the Concept Space is undergoing further modifications. Future development efforts for COPLINK project will also be discussed.

**Keywords:** Law Enforcement, Information Systems, Knowledge Management, Information Retrieval, Intelligence Analysis, Information Sharing

## 1. INTRODUCTION

### 1.1. Law Enforcement Intelligence Analysis and Knowledge Management

In this era of the Internet and distributed multimedia computing, new and emerging classes of information technologies have swept into all areas of business, industry and government. As information technologies and applications become more overwhelming, pressing, and diverse, persistent information technology problems have become even more urgent. *Information overload*, a result of the ease of information creation and rendering via Internet, the WWW, and organizational data sources, has become more evident in people's lives[1]. This phenomenon is nowhere more evident than in government, specifically in criminal justice information systems. Federal, state, and local criminal justice entities possess vast repositories of information, but the explosive growth in digital information and the need for access within government agencies have made information overload increasingly significant.

Agencies' knowledge management problems frequently stem from barriers to access and utilization resulting from incompatible content and format of information[2] that make creation and utilization of knowledge management a complex and daunting process. Nevertheless, a number of different applications and approaches to knowledge management technologies are emerging, among them: virtual enterprising[3], joint ventures[4], aerospace engineering[2], and digital libraries[5].

Several government initiatives have been established to address some of the problems of the law enforcement sector of the digital government. The Office of Justice Programs (OJP) Integrated Justice Information Technology Initiative is involving five bureaus including the National Institute for Justice (NIJ) in an effort to use wired-information technologies to improve the effectiveness and fairness of the justice system through better information sharing. An NIJ wireless initiative, the AGILE program of the NIJ Office of Science and Technology primarily addresses interoperability issues (other government initiatives are described on http://www.ojp.usdoj.gov).

*Correspondence: Email: hchen@bpa.arizona.edu; WWW: http://ai.bpa.arizona.edu; Telephone: 520 621 2748; Fax: 520 621 2433

These government initiatives motivated a proposal to unite the technical expertise of the University of Arizona's Artificial Intelligence Lab with the law enforcement domain knowledge of Tucson Police Department to develop cutting edge technologies for the law enforcement community in the COPLINK project. This paper describes the COPLINK Concept Space application, one of those technologies which, although originally funded by NIJ, has received additional funding from both NIJ and the National Science Foundation (NSF) under its Digital Government Initiative. The Artificial Intelligence Lab has gained wide recognition as a cutting-edge research unit and has been featured in *Science* and *The New York Times,* having received more than $9M in research funding from various federal and industrial sponsors since 1989. The Lab sees COPLINK as an opportunity to serve the community by bridging the gap between research in developing technologies and solving such real-world problems as helping police officers fight crime.

### 1.2. A Case Study: The Tucson Police Department

The Tucson Police Department (TPD) recently evaluated its information technology and identified problems of lack of information sharing, integration, and knowledge management. The department agreed to participate in research to investigate the potential of current state-of-the-art, near-term, and cost-effective database, Intranet, and multimedia technologies to make computer justice information database integration, management, and access more effective.

The Tucson Police Department (TPD) has encountered all the problems described in the previous section. Its information sources have included at least three distinct systems:

- The main incident-based system, Records Management System (RMS) captures the highlights of an incident in an Oracle 7.x database.
- A separate system by ImageWare Software Inc. captures mug shots (photos taken at the time of arrest) and limited related information in a Sybase database.
- A third information source, Criminal Information Computer (CIC) is a homegrown Microsoft Access-based application used to track gang activity. TPD officials attribute a disproportionate percentage of Tucson's criminal activity, especially homicides, to gang members and their known associates.

RMS contains approximately 1.8 million incident record sets and mug shot records (around 60000 mugs). CIC tracks the approximately 1200 individuals the department considers responsible for a majority of major crimes. Each of these systems has a different user interface, so accessing related information from any two or all three, has been difficult, cumbersome, and time-consuming:

- RMS has a cumbersome, difficult-to-navigate command-line driven system.
- CIC's gang database has been accessible only to certain detectives through a simple homegrown front-end interface.
- Mugshot database, a collection of arrest photographs, can only be integrated with information in RMS manually through a specific mug shot number.

As an NIJ-funded multi-year project, the major goals for the COPLINK project for TPD are:

- To develop an integrated system to allow TPD officers easy access to all the information contained in all three systems.
- To design a prototype system for use in developing similar systems at other police departments.
- To offer a model for allowing different police departments to share data easily.

The COPLINK project attacks several problems existing in many law enforcement agencies, including TPD, by developing a model integrated system that allows law officers both within and between different agencies to access and share information. An additional goal of COPLINK is to develop consistent, intuitive and easy-to-use interfaces and applications that support specific and often complex law enforcement functions and tasks. While the scope of this project includes a multilevel development plan incorporating different information technologies, the focus of the research reported here is not only on the development of a multimedia database system to promote information sharing, but also the improvement of criminal intelligence analysis. The first step in this process was the evaluation of TPD's current Records Management System (RMS).

#### 1.2.1 TPD's Records Management System (RMS)

The main database at TPD is the Records Management System (RMS), which stores a wide variety of data, including criminal case information and incident information from calls for service recorded from the Department's Computer-Aided Dispatch (CAD) system. RMS is a text-based system that is accessed using VAX terminals stationed in many central headquarters offices as well as substations located around the city.

Similarly to systems at many other law enforcement agencies, RMS has many problems pertaining to its interface, access to information and lack of knowledge management. Although users are able to search on name queries, location queries, vehicle queries, etc., they are not able to search multiple fields simultaneously. In addition, users of RMS complain that, depending on the type of query, RMS can take from a few minutes to a few hours to return its results.

### 1.2.2 Current TPD Knowledge Management Practice

A function of the daily routine of many crime analysts and detectives at TPD is to create knowledge from information by analyzing and generalizing current criminal records that consist of approximately 1.8 million criminal case reports containing details from criminal events dating back to 1986. Although investigators can access RMS to tie together information needed to solve cases and crimes, they must manually search for connections or relationships in existing in the data. Combining information to create knowledge is often hampered by voluminous information examination of which requires exorbitant time and effort on the part of the investigator. Compounding this problem is the variability of individual investigator's ability to locate relevant information. The problem is not necessarily that the information has not been captured—any officer who fills out up to seven forms per incident can attest to that. The problem is one of access. Typically, law-enforcement agencies have captured data only on paper or have fed it into a database or crime information system. If the agency involved has more than one of these (that are possibly incompatible), information retrieval can be difficult or time-consuming.

Potent information retrieval tools can provide information sharing abilities as well as alleviate crime analysts' information overload, reduce information search time required for analysis of available criminal records, and advance the investigation of current cases. This paper introduces two knowledge management systems that can provide the ability to access data from different systems as well as provide the functionality of intelligence analysis that currently does not exist in the RMS system. Real-life context evaluation of both systems, the COPLINK Database and COPLINK Concept Space, and future directions for the project are also discussed.

## 2. COPLINK DATABASE APPLICATION

After analyzing user requirements, we created the COPLINK Database (DB) application, employing a consistent and intuitive interface which integrates different data sources, such that the multiplicity of data sources remains completely transparent to the user, allowing law enforcement personnel to learn a single, easy-to-use interface. In addition to the interface design, we also developed a model that allows for information sharing both within and between law enforcement organizations.

### 2.1 Design Criteria

The main design criteria considered for the COPLINK project includes:

- **Platform independence**: Because not all police departments utilize the same hardware or software operating systems, platform independence was critical.
- **Stability and scalability**: The system also had to offer room for system growth and expansion.
- **Intuitive and ease of use**: The front-end user interface should be intuitive and easy to use, yet flexible enough to meet the equally demanding investigative needs of detectives and officers.

Typical law enforcement applications usually are legacy systems having out-dated performance and capability. For example, TPD's RMS took 30 seconds to answer simple requests and up to 30 minutes for more complex queries. Improved response time was critical to restoring departmental efficiency. To ensure application speed, issues of data and network communication, disk access and system I/O needed to be addressed. This also meant carefully distributing logic where it could be most quickly and efficiently executed, i.e., all user-input error checking should be done in the front end, and all database access logic achieved through pre-compiled stored PL/SQL procedures in the database.

Another critical issue, especially in designing a system that could be deployed across multiple law enforcement agencies, was acknowledging that no two agencies would store their incident data in exactly the same way. Therefore, it was important to come up with a data organization design that was flexible enough to be applied to any underlying data set. The database team designed a series of standardized "views" that fitted typical information search and presentation situations. For example, most of the data in the TPD systems were related to "Person," "Location," "Vehicle," or "Incident" information. A set of views was developed for each of these areas of interest, with the underlying data sets mapped to those standard views, making the system more portable to other law enforcement agencies.

## 2.2 Database Design

Based on the criteria established and after much investigation, the COPLINK team decided upon a three-tier architecture (see Figure 1):

- A front-end interface: The front-end should be a thin client, consisting of a series of user-friendly query screens matching the four main areas previously discussed (Person, Location, Vehicle, and Incident). The front-end would generate query requests.
- A middle-ware application server: The middle-ware would handle secure requests from multiple clients, and execute the stored procedures in the database.
- A back-end database: Results from the database would be processed by the middle-ware, and be formatted into return data strings. These return strings would then be sent to the front-end where they would be parsed and displayed to the user.

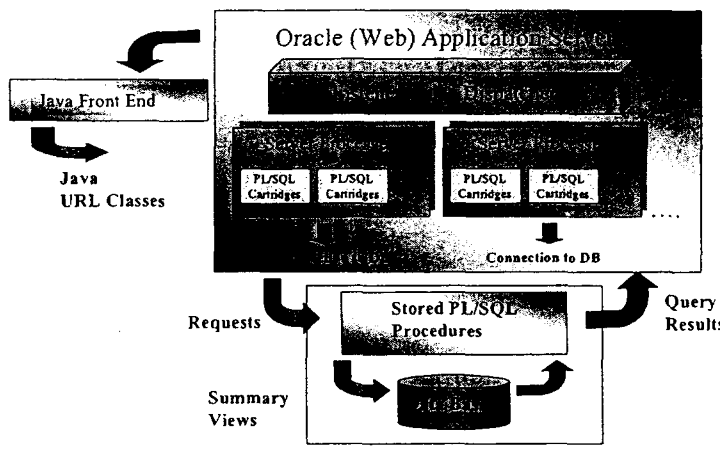



**Figure 1: COPLINK Database Three-tier Architecture**

As mentioned, the front-end had to be a platform-independent thin, stable client, based on a popular programming language. Our current prototype, created using Java 1.1, allows for client-side analysis, avoiding the overhead incurred by database operations. Oracle's Application Server (OAS) met our middle-ware needs. It has versions available for both Windows NT and UNIX-based systems and utilizes a CORBA-based "cartridge server" system. A cartridge server is a shared library that either implements program logic or provides access to program logic stored elsewhere, such as in a database. In implementing the COPLINK application, we utilized the PL/SQL cartridge system of the OAS, which gives access to the logic stored as pre-compiled PL/SQL procedures in the database. The procedures actually execute the queries in the database, and return the results to the front-end application as HTTP-based strings. Although this system appears to be Oracle-centered, it has flexibility that allows us to access non-Oracle databases whereas such a cartridge as ODBC could only be used to access an ODBC-compliant database. The database system was designed to be compatible with either Oracle 7.3 or 8.0, and different versions of the data sets have been run on Windows NT and Dec Alpha UNIX platforms. The major portions of the database consist of tables and indices that contain incident-based information, the set of views discussed previously, a series of procedures used by the middle-ware to query the database, and the packages necessary to execute queries from the OAS.

There are four main query screens, each resulting in a summary listing of information related to an initial query. Figure 2 illustrates relationships among queries. For example, if a user initiates a search on a particular first-name/last-name combination, a summary table is presented as a result of a dynamic SQL query, listing all possible matches, as well as the number of incidents associated with each individual match. From there, the user can select either a secondary listing of incidents related to a particular individual or can access a more detailed summary of the personal information on the individual. For an incident summary, all the pertinent case detail information on a particular incident is presented. For a detailed person summary, the user can select the incident summary for that individual, and from there obtain case details for any incident listed.

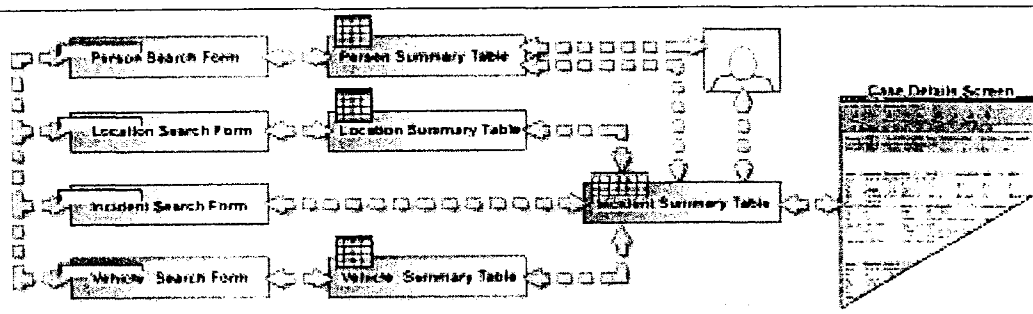



**Figure 2 - Screen Flowchart**

An officer wanting to know more about a particular incident or person can enter a query in the search form, query further through the summary table to see details about a person, or select an incident from the incident summary table to view on the case details summary screen. In previous screens, information could be displayed in formatted rows, but a more dynamic display was needed. For example, mug shots needed to be displayed both as person details and on the case-details screen. To accommodate this feature, screens have been laid out in clusters, grouping information for easier understanding. This in turn required manipulating the data retrieved and capturing pictures from the database, a problem solved by constructing a cyclical procedure that would loop through the data and build a hierarchical tree. We could then apply display patterns to the nodes of the tree, navigate the tree and place the information on the screen.

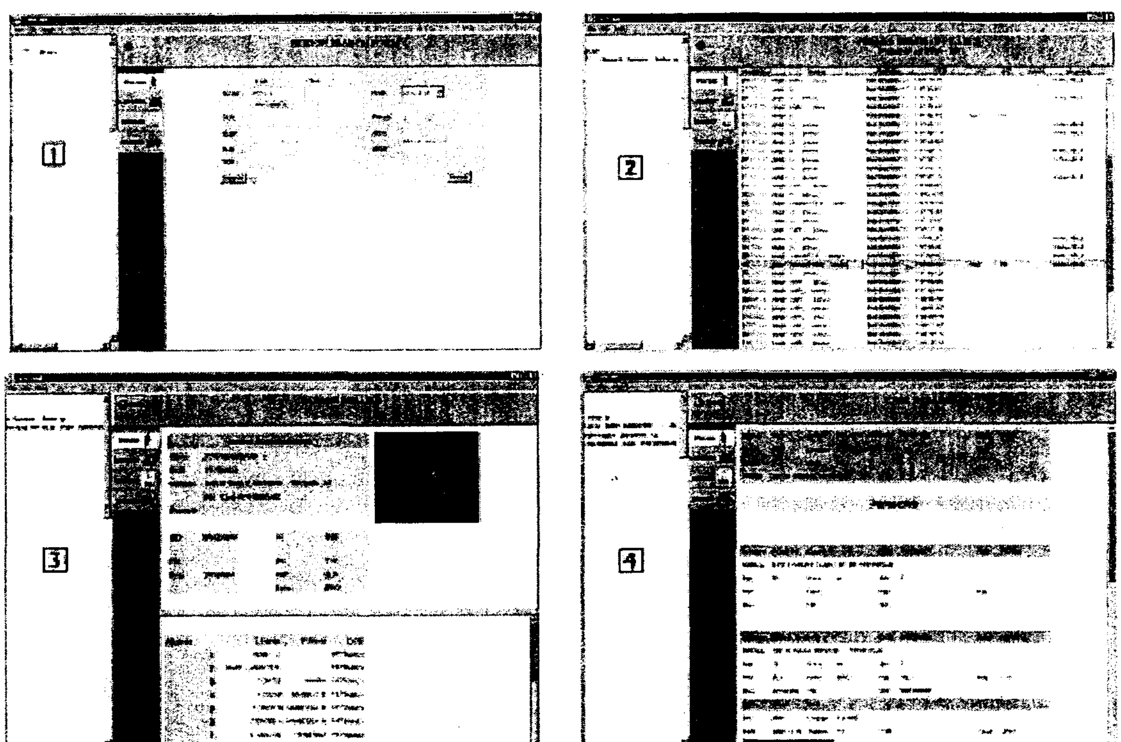



**Figure 3 – COPLINK Database Interface Screen Samples–** Sample Scenario: An officer is trying to identify a suspect involved in an automobile theft. A confidential informant has reported that the suspect goes by the street name of "Baby Gangster," is about 20 (probably born in 1979), and is around 5'3" tall.

Screen 1- COPLINK DB Search. The officer can choose one of the four types of information upon which to search: Person, Location, Incident, or Vehicle. The officer selects the Person search screen and enters "baby g" in the COPLINK DB system. Note the left panel history screen, which keeps track of the user's searches. Screen 2 - Person Summary. The system returns 58 listings referring to "baby g;" (all of the returns include the name "baby g)." The system permits sorting by any of the column headings in the table. The officer chooses to sort by date of birth and finds an entry for "baby gangster," born in 1979, whose height is 5'2". The officer then clicks on the "See

Details" button to find out more about this particular "Baby Gangster". Screen 3- Person Details. This screen contains personal information about the selected person, including real name, latest description information, latest home address, other identifiers that the person may use, and a mug shot, if available. The officer now has a real name of a person who matches the description of the possible suspect he was given. The officer then decides to go to the incident summary screen to get an idea of the cases in which this person has been involved. Screen 4- Incident Summary. This screen displays all the incidents in which the selected person has been involved. The officer sorts by crime type, looking for cases of stolen vehicles (0701) and finds the suspect has been involved in four such incidents, either as a suspect or as an arrestee. The officer selects Case #9711250126 to look at the actual case information.

### 2.3 Graphical User Interface for COPLINK Database

The graphical user interface (GUI) for the COPLINK Database Application is shown in Figure 3, although actual information has been altered to maintain data confidentiality. The Java front-end consists of two major parts, the input and display of data and the processing of information. Working closely with TPD officers, the COPLINK team first made low-fidelity, paper prototypes of the screens used to obtain feedback on the display and organization of the information, which was used to modify the design and functionality of the interface. Display of results was important to the front-end. We learned that a user's idea of what constitutes a manageable and intuitive display varied with the query type and sometimes required formatting in a different way. We responded by creating a dynamic text table, using the Java API to make the interface more flexible.

### 2.4 User Evaluations for the COPLINK Database Application

A usability evaluation, involving 52 law enforcement personnel, was conducted to assess the achievement of a number of the goals that guided the design and development of the COPLINK Database. Items on the questionnaire used to assess and compare the COPLINK and RMS systems were based upon user perceptions of such widely used measures of usability as: *effectiveness* (impact of system on job performance, productivity, effectiveness of information, and information accuracy), *ease of use* (measures of effort required to complete a task, ease of learning how to use the application, ability to navigate easily through the different screens, and satisfaction with the interaction), and *efficiency* (speed of completing tasks, organization of the information on the screens, ability to find information and the interface design itself)[6].

Benchmark levels from TPD's current RMS system for all three usability factors were established and compared with COPLINK DB ratings. In addition to written questionnaires, observation of the data collection methods and structured interviews were used both to supplement findings and to provide feedback for further development efforts.
Data analysis of the usability questionnaire support a conclusion that use of COPLINK DB provided improved performance over use of the current RMS system. On all usability measures (effectiveness, ease of use, and efficiency), participants rated COPLINK DB higher than RMS, with the average rating for COPLINK being 4.1 and RMS being 3.3 (1=strongly disagree to 5=strongly agree). Statistical analyses revealed that this ratings difference was significant for all measures.

From both questionnaires and interviews, participants indicated that the quality and quantity of information from COPLINK DB surpassed those of RMS. In a review of current RMS practices, a number of detectives and officers were actually unable to use RMS but were able to use COPLINK DB to conduct searches. It is evident from this evaluation that COPLINK DB allowed a population of TPD personnel to access information that would have been quite difficult for them to acquire using the RMS system. From both the questionnaire and the interview data collected from this evaluation, it is evident that many participants rated the information found in COPLINK as more useful than the information in RMS. This finding is very interesting, because most of the information contained in COPLINK has been taken from RMS.

## 3. COPLINK CONCEPT SPACE

To complement the functionality of the COPLINK Database application, our next phase of the COPLINK project was to develop a knowledge management tool specifically designed to aid investigators and detectives in criminal intelligence analysis. Drawing upon artificial intelligence techniques and algorithms, the COPLINK Concept Space was created.

Concept space, or automatic thesaurus, is a statistic-based, algorithmic technique used to identify relationships between objects (terms or concepts) of interest[7]. The technique is frequently used to develop domain-specific knowledge structures for digital library applications. In the University of Arizona Artificial Intelligence Lab, the idea of concept space was generated to facilitate semantic retrieval of information. Several user studies showed concept space to improve searching and browsing in the engineering and biomedicine domains. In the biosciences, the concept space approach was applied to the Worm Community System (WCS) and the FlyBase system. There also have been successful results in the Digital Library Initiative studies conducted on the INSPEC collection for computer science and engineering and on Internet searching[5,8]. Current on-going concept space research is being conducted in geographical information systems, law enforcement, and medicine.

A concept space is a network of terms and weighted associations that represent the concepts and their associations within an underlying information space that can assist in concept-based information retrieval. In addition, co-occurrence analysis uses similarity and clustering functions[9] to weight relationships between all possible pairs of concepts. The resulting network-like concept space holds all possible associations between objects, which means that every existing link between every pair of concepts is retained and ranked.

In COPLINK CS, detailed case reports are the underlying documents and concepts are meaningful terms occurring in each case. Concept Space provides the ability to easily identify relevant terms and their degree of relationship to the search term. The relevant terms can be ranked in the order of their degree of association so that the most relevant terms are distinguished from inconsequential terms. From a crime investigation standpoint, Concept Space can help investigators link known objects to other related objects that might contain useful information for further investigation. For instance, like people and vehicles related to a given suspect.

Information related to a suspect can direct an investigation to expand to the right direction, but a case report that reveals relationships among data in one particular case fails to capture those relationships from the entire database. In effect, investigators need to review all case reports related to a suspect, which may be a tedious task. In the COPLINK project, we introduce Concept Space as an alternative investigation tool that captures the relationships between objects in the entire database.

To date, we have successfully adopted our techniques to create a COPLINK Concept Space based on a collection of 1.5 million case reports from the current Tucson Police Department Records Management System. These cases span a time frame from 1986 to 1999 (the entire case record collection for the City of Tucson). Based on careful user requirement analysis, five entity fields from the database were deemed relevant for Concept Space analysis: Person, Organization, Location, Vehicle, and Incident type. The purpose of this tool is to discover relationships between and among different crime-related entities. It is important not only to know that there is a relationship, but also to know what each relationship is. Figure 4 provides samples of the COPLINK Concept Space application.

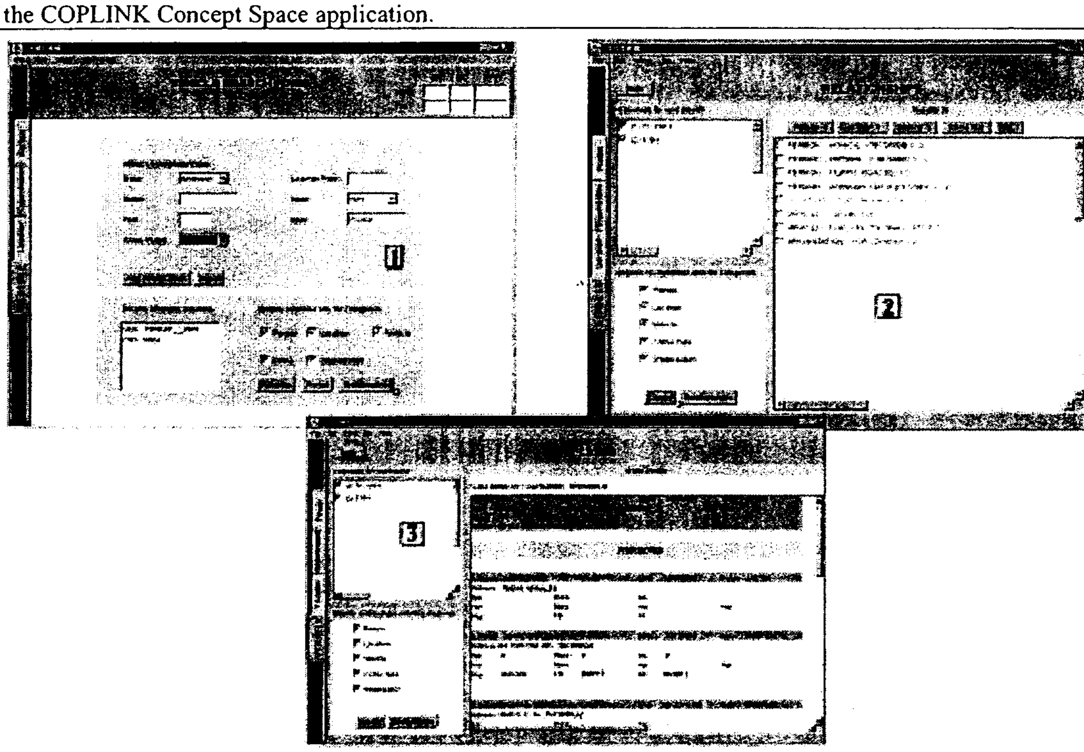



**Figure 4 – COPLINK Concept Space Interface Sample Screens--** Sample Scenario: A detective is investigating a robbery at a local convenience store. The only witness, the night store clerk, only remembers that the suspect drove away in a white pickup truck.

Screen 1 - COPLINK CS Search. Using COPLINK Concept Space, users are able to enter one of four information types as a search term. In this example, the detective needs to generate a lead, given the type of crime and the use of a white pick-up truck. The detective selects the Vehicle search screen and enters "White" for color, "Pickup" for style, and "0304", the universal crime report classification code for robbery of a convenience store. After adding the search terms to the relations box, the detective selects the Relationship button to enter the Concept Space. Note that the user can choose to select or deselect the types of relations returned by the system. This allows the user to choose only relevant categories and control for information overload. Screen 2 - COPLINK CS Results. The system returns eight terms related to both a white pick-up and the 0304 crime type. Note that the Concept Space can return elements for each of the five information object types. The detective now knows not only that are there four people somehow related to both this type of crime and vehicle, she also has a license plate number for a vehicle. The detective can always add any of the Concept Space terms to the search or remove one of the two keywords from the search. As on the initial search screen, the panel in the lower left-hand corner allows users to control the amount of information returned by the Concept Space. The detective decides to view the cases that underlie the relationships uncovered by the Concept Space. Screen 3 - CS Case Details. The Cases view displays actual case reports; in this example, there is only one case in the database. The detective can view the details of the prior incident, including the role of each person involved, and their home addresses. At any time, the detective can choose "Back" to review previous screens or modify the search keywords by selecting another type of search term or deselecting the current search terms.

### 3.1. Applying the Concept Space Technique to Criminal Data

In general, there are three main steps in building a domain-specific Concept Space. The first task is to identify collections of documents in a specific subject domain; these are the sources of terms or concepts. For Tucson Police Department, we are using the case reports in the existing database. The next step is to filter and index the terms. The final step is to perform a co-occurrence analysis to capture the relationships among indexed terms. The resulting Concept Space is then inserted into a database for easy manipulation (for a more in-depth analysis of the Concept Space algorithm, see [9]). The last two steps have been customized for COPLINK. After optimizing the code and tuning the database, we found that the total time required for building a COPLINK Concept Space is approximately five hours, which is acceptable in the given situation.

#### 3.1.1 Term Filtering and Indexing

Due to the nature of the data residing in TPD's database, each piece of information is categorized in case reports and stored in well-organized structures. Theoretically, concept space can contain any number of term types (e.g., person names, organizations, locations, crime types, etc.). In practice, however, the size of the database, the time required to build a Concept Space, and the response time of queries are major constraints that limit the number of term types. To balance performance and comprehensiveness, a Concept Space should contain only meaningful types frequently searched by users. With the collaboration of personnel from the TPD personnel, we created a set of term types for the COPLINK Concept Space. Table 1 shows the term types supported by Concept Space and the size of each.

| General | Category | Type | Size |
|---|---|---|---|
| **Total Case Reports** | | | 1,504,838 cases |
| **Total Unique Terms** | | | 1,267,776 terms |
| | *Person* | Full Name | 744,250 terms |
| | *Organization* | Organization Name | 26,517 terms |
| | *Location* | Street Address | 141,875 terms |
| | | Street Address with Apartment | 58,664 terms |
| | *Vehicle* | License Number | 202,996 terms |
| | | Vehicle Identification Number | 57,547 terms |
| | | Make / Color | 1,924 terms |
| | | Make / Model | 908 terms |
| | | Make / Model / Color / Year | 30,142 terms |
| | | Make / Model / Style | 2,749 terms |
| | | Make / Model / Year | 6,500 terms |
| | | Make / Style / Color | 7,334 terms |
| | | Make / Style / Color / Year | 39,708 terms |
| | | Make / Style / Model / Color / Year | 41,757 terms |
| | | Make / Year | 2,773 terms |
| | | Style / Color | 938 terms |
| | | Style / Year | 1,609 terms |
| | *Incident Type* | Crime Type | 374 terms |
| | | Team / Beat (Geographical area) | 147 terms |

| | | | |
|---|---|---|---|
| **Average Number of Terms per Case** | | | 5.43 terms |
| **Number of Associated Terms** | | | 27,707,675 pairs |

Table 1: Statistical Information on the COPLINK Concept Space

Term types in Concept Space can be divided into the five main categories. For a Person, Organization, Location, and Incident type, only one piece of information, such as a person's full name, street address, or crime type, is descriptive enough to be a search term. On the other hand, for a Vehicle, one piece of information, such as color, make or type, typically is comparatively common and when used as a search term would generate a large number of relevant terms. This problem can be avoided by combining two or more non-specific terms into composite terms.

The index maintains the relationship between a term and the document in which it occurs. Both index and reverse index are required for co-occurrence analysis. The index contains the links from term to document; the reverse index contains the links from document to term.

**3.1.2 Co-occurrence Analysis**

After identifying terms, we first computed the term frequency and the document frequency for each term in a document, based on the methodology developed by Chen[9]. Term frequency, *tf*, represents the number of occurrences of term j in document i. Document frequency, *df*, represents the number of documents in a collection of *n* documents in which term *j* occurs. A few adjustments were made to the standard term frequency and inverse document frequency measures.

We then computed the combined weight of term *j* in document *i*, $d_{ij}$, based on the product of "term frequency" and "inverse document frequency" as follows:

$$d_{ij} = tf_{ij} \times \log(\frac{N}{df_j} \times w_j) \quad (1)$$

where $N$ represents the total number of documents in a collection and $w_j$ represents the weight of words in descriptor *j*. In general, some term types are more descriptive and more important than others and deserve to be assigned higher weights so as to ensure that relationships associated with these types are always ranked reasonably. In COPLINK Concept Space, crime types are assigned comparatively higher weights. We then performed term co-occurrence analysis based on the asymmetric "Cluster Function" developed by Chen and Lynch[9].

$$W_{jk} = \frac{\sum_{i=1}^{n} d_{ijk}}{\sum_{i=1}^{n} d_{ij}} \times WeightingFactor(k) \quad (2)$$

$$W_{kj} = \frac{\sum_{i=1}^{n} d_{ikj}}{\sum_{i=1}^{n} d_{ik}} \times WeightingFactor(j) \quad (3)$$

$W_{jk}$ indicates the similarity weights from term j to term $k$ and $W_{kj}$ indicates the similarity weights from term $k$ to term $j$. $d_{ij}$ and $d_{ik}$ were calculated based on the equation in the previous step. $d_{ijk}$ and $d_{ikj}$ represent the combined weight of both descriptors $j$ and $k$ in document $i$. However, they were computed slightly differently due to their different starting terms. They are defined as follows:

$$d_{ijk} = tf_{ijk} \times \log(\frac{N}{df_{jk}} \times w_j) \quad (4)$$

$$d_{ikj} = tf_{ijk} \times \log(\frac{N}{df_{jk}} \times w_k) \quad (5)$$

where $tf_{ijk}$ represents the number of occurrences of both term $j$ and term $k$ in document $i$ (the smaller number of occurrences between the terms was chosen); $df_{jk}$ represents the number of documents (in a collection of $N$ documents) in which terms $j$ and $k$ occur together.

In order to penalize general terms (terms which appeared in many places) in the co-occurrence analysis, we developed the following weighting scheme, which is similar to the inverse document frequency function.

$$WeightingFactor(k) = \frac{\log \frac{N}{df_k}}{\log N} \tag{6}$$

$$WeightingFactor(j) = \frac{\log \frac{N}{df_j}}{\log N} \tag{7}$$

Terms with a higher $df_k$ or $df_j$ value (more general terms) had a smaller weighting factor value, which caused the co-occurrence probability to become smaller. In effect, general terms were pushed down in the co-occurrence table (terms in the co-occurrence table were presented in reverse probabilistic order, with more relevant terms appearing first.)

Significant research needs to be conducted to investigate using Concept Space with our proposed noun phrasing and entity extraction techniques. In the above example, entity types from database fields were identified manually by human analysts. In addition, the Tucson Police Department does not yet capture free text narratives. Many law enforcement agencies have begun to incorporate content-rich narratives in their record management systems (e.g., Phoenix Police Department has complete narratives for each case). These narratives will provide a fertile test bed for combining noun phrasing and Concept Space analysis for intelligence identification.

**3.2 User Evaluations for the COPLINK Concept Space**

We conducted user evaluations to examine the effects of COPLINK CS on law enforcement investigation and work practices[10]. Twelve crime analysts and detectives, participated in the four-week longitudinal evaluation, during which they were asked to complete journal entries on searches they had conducted using COPLINK CS. By utilizing data collection methods of documentation, structured interviews, and direct observation, we were able to evaluate the function and design of the COPLINK CS system.

The journals and interviews revealed three major areas in which COPLINK Concept Space provided support for intelligence analysis and knowledge management.

**3.2.1 Link analysis and Summarization**

Participants indicated that Concept Space served as a powerful tool for acquiring information and cited its ability to determine the presence or absence of links between people, places, vehicles and other object types as invaluable in investigating a case[11]. The impact of link analysis on investigative tasks is crucial to the building of cases. An officer assigned to investigate a crime has to have enough information to provide a lead before he/she can begin working. Too many cases have to be closed because of lack of information or inability to utilize information existing elsewhere in the records management system. Concept Space manages all the data in the records system in such a way that it can be used as knowledge about the suspect. Link analysis can represent one of three types: directly linking known information, indirectly linking known information, and linking unknown information. Participants also reported they could use a concept space as a summary of the different information types related to a search term.

**3.2.2 Interface Design**

In general, users reported that the web-based interface of the COPLINK Concept Space was engaging and quite easy to use. Officers said the use of color to distinguish different object types and a graphic user interface provided a more intuitive tool than the text-based RMS system. Additionally, the ability to have results returned as either the concept space or case details allowed them to specify the type of information they needed. Participants reported that the data fields chosen for the Concept Space embody the basic necessary information for an investigation. They also reported that the separation between different fields in the output effectively encouraged easy comprehension of the information. A criminal investigation usually requires officers to make specific connections between people, places, vehicles, etc. in order to build a complete picture. The ability to aggregate information fields for searching provides a potent tool for problem solving and crime investigation.

**3.2.3 Efficiency**

Perhaps one of the most crucial benefits of the use of COPLINK Concept Space in law enforcement is its speed. As one of our participants explained, identifying a suspect between 48 to 72 hours after a crime is difficult. Beyond this time frame, a suspect is able to destroy evidence that may tie him/her to the crime or change his/her appearance to avoid identification. Witness/victim memory of the suspect's appearance also fades within this period. Identification of the suspect ideally should occur within 48 hours of the crime, so establishing useful links for identifying and locating the suspect is a crucial step. A number of interview and journal comments indicated that use of COPLINK Concept Space increased productivity by reducing time spent per information search.

In journals and interview sessions, each participant was asked to report the time it took to complete at least one particular search task using both RMS and COPLINK CS. The data indicated that in direct comparison of 15 searches, use of COPLINK Concept Space required an average of about 30 minutes less per search than with the use of RMS. However, review of other qualitative data from the journals and interviews indicated that subjects perceived much quicker response to a query from COPLINK CS, especially when multiple search entries and query expansion were involved.

## 4. FUTURE DIRECTIONS FOR COPLINK

Criminals do not bound themselves by county borders or jurisdictions. Furthermore, criminals are creatures of habit and being able to understand their habits and close associations is important[12]. The COPLINK Database and Concept Space applications take advantage of these characteristic by not only promoting information sharing between stovepipe information sources and different agencies, but also by capturing connections between people, places, events, and vehicles, based on past crimes. Our evaluation of these knowledge management and intelligence analysis applications support its potential for transforming law-enforcement practices in this age of digital governments. In addition to these two projects, we are currently developing a number of other technologies for the law enforcement community.

Large collections of unstructured text as well as structured case-report information exist in police records systems. These textual sources contain volumes of information for investigators that are often not captured in the structured fields. One future research direction is to explore the development of textual mining approaches that support knowledge retrieval from such sources for law enforcement case reports. In order to perform a fine-grained content analysis, we will investigate the development of linguistic analysis and textual mining techniques that make intelligent use of large textual collections in police databases.

Several Internet research projects have shown the power of a new "agent" based search paradigm. In addition to supporting conventional searches performed by users, search agents allow users automatically to establish search profiles (or create profiles for users) and extract, summarize, and present timely information content. We believe such a proactive search agent is well suited to use by investigative personnel in law enforcement agencies. Search agents for law enforcement can support conventional searching techniques and be profiled for specific investigations. We plan to develop a personalized law enforcement search agent that will support wide expansion in connectivity and information sharing between police agencies.

In relation to the COPLINK project, the concept of a distributed database system has important implications. The most important of these is accessibility to and dissemination of law enforcement records and information. Currently, the vast majority of criminal data collection and compilation is done on a community level but may not be in a format that is readily available and accessible to local law enforcement officers. A distributed COPLINK prototype is under development using three COPLINK database servers to simulate the independent nodes in a distributed environment. Work is under way to include functionality that will provide interoperability among the different DBMS platforms that may support future COPLINK nodes. In the immediate future, we plan to begin deployment and testing of a Distributed COPLINK prototype with the Tucson and Phoenix police departments. As distributed solutions and analysis tools are developed for law enforcement officers, a specific focus must be on providing tools within the constraints of a wireless environment. One of our future goals is to develop and refine applications to support the expansion of distributed and mobile law enforcement networks and inter-jurisdictional information retrieval as well as to investigate and study network security issues.

## ACKNOWLEDGMENTS

This project has been funded by a grant from the National Institute of Justice, Office of Science and Technology (Tucson Police Department, "COPLINK: Database Integration and Access for a Law Enforcement Intranet", $1.2M, 1997-1999) and the National Science Foundation ("COPLINK Center: Information and Knowledge for Law Enforcement", $530K, 2000-2001). We would also like to thank the Digital Equipment Corporation External Technology Grants Program, agreement #US-1998004, for its award of a $198,451 equipment grant allowance toward the purchase of a DecAlpha Server for the COPLINK Project. Appreciation also goes to all the personnel from the Tucson Police Department who were involved in this project.

## REFERENCES

1. D. C. Blair and M. E. Maron, "An Evaluation of Retrieval Effectiveness for a Full-Text Document-Retrieval System," *Communications of the ACM* **28**, pp. 289-299, 1985.
2. P. Jones and J. Jordan, "Knowledge Orientations and Team Effectiveness," *Int'l J. of Technology Management* **16**, pp. 152-161, 1998.
3. Y. M. Chen, C. C. Liao, and B. Prasad, "A Systematic Approach of Virtual Enterprising Through Knowledge Management Techniques," *Concurrent Engineering-Research and Applications* **6**, pp. 225-244, 1998.
4. C. Inkpen, and A. Dinur, "Knowledge Management Processes and Joint International Ventures," *Organization Science* **9**, pp.454-468, 1998.
5. H. Chen, "Artificial Intelligence Techniques for Emerging Information Systems Applications: Trailblazing Path to Semantic Interoperability," *J. of the American Society for Information Science* **49**, pp. 579-581, 1998.
6. R. V. Hauck, "COPLINK: Exploring usability of a multimedia database application for law enforcement," Report. Available: http://ai.bpa.arizona.edu/go/datawarehousing/publications/nij.pdf. 1999.
7. M. Lesk, *Practical Digital Libraries*, Morgan Kauffmann, Los Altos, 1997.
8. H. Chen and D. T. Ng, "An Algorithmic Approach to Concept Exploration in Large Knowledge Network (Automatic Thesaurus Consultation): Symbolic Branch-and-Bound vs. Connectionist Hopfield Net Activation," *J. of the American Society of Information Science* **46**, pp. 348-369, 1995.
9. H. Chen and K. J. Lynch, "Automatic Construction of Networks of Concepts Characterizing Document Database. *IEEE Transaction on Systems, Man and Cybernetics* **22**, pp. 885-902, 1992.
10. R. V. Hauck and H. Chen, "COPLINK: A Case of Intelligent Analysis and Knowledge Management," *Proc. of the 20th Annual International Conference on Information Systems*, P. De and J. I. DeGross, (Eds.), pp. 15-28, International Conference of Information Systems, Atlanta, 1999.
11. W. R. Harper and D. H. Harris, "The Application of Link Analysis to Police Intelligence," *Human Factors* **17**, pp. 157-164, 1975.
12. N. Joyce, "ICAM: Chicago's Newest Crime-Fighting Tool," *Proc. of the Conference in Technology, Community Policing,* National Law Enforcement and Corrections Technology Center. Available: http://www.nlectc.org/txtfiles/confrpt.html. 1996.