Report to the Governing Board
Program Status Report

Connecticut Information Sharing System (CISS),
Connecticut Impaired Driving Record Information Systems (CIDRIS) and
Offender Based Tracking System (OBTS)

Criminal Justice Information System (CJIS) Governing Board

October 20, 2011

Table of Content

Executive Director

Sean Thakkar

Governor's Vision for Technology:

- Implementation of efficient, modern business processes that result in cost-effective delivery of services
- Open and transparent engagement with the citizens of the State
- Accurate and timely data for policy making, service delivery and results evaluation
- A secure and cost effective IT infrastructure, including greater use of shared services and applications wherever possible
- Easily accessible services to all constituents

Business Goals and Objectives:

- Optimize our current investments in technology and leverage existing infrastructure and resources
- Create a simple way to implement new technologies so that agencies can implement them smoothly
- Develop a secure environment which meets state and federal standards for security
- Provide independent and objective opinions and recommendations to the CJIS Governing Board
- Provide services that are boringly predictable and totally reliable

CJIS Committee Chairs

The committees and their chairpersons are as follows:

Administrative Committee

Larry D'Orsi

Judicial Branch, Court Operations Division

Technology Committee

Evelyn Godbout

Department of Criminal Justice

Implementation Committee

Chief Richard Mulhall

Connecticut Police Chiefs Association (CPCA)

Summary of Accomplishments

- The State and vendor Xerox – Affiliated Computer Systems, Inc. (ACS) executed the long awaited contract to build and deploy the CISS system.
- The CISS team expects to have the Program Manager and Business Manager on board at the beginning of October with the Senior Project Managers, Java Developers, and Business Analyst on board in November.
- The OBTS program is in production with release 7.0 and has started gathering requirements for release 7.1
- The CIDRIS project is in final stages of testing and training of DESPP troopers prior to going into full production by mid December.

Critical Enablers for Continued Success

- The next installment of the $7.7 million Bond Funds approval is essential by the Bond Committee so that we have the funds available to pay the vendor Xerox – ACS in conformance with our contract payment schedule
- Executive and Legislative support is requested for approval of the Technical Adjustment to the FY2012-FY2013 CJIS Biennial Budget, as submitted by the CJIS Executive Director in October 2011
- Replacement of key CJIS Support Group (CSG) business and technical resources which are currently vacant is considered critical to ensuring that the CSG can provide effective business and technical support for OBTS and CIDRIS
- Develop and implement a mutually supportable Service Level Agreement (SLA) between the Department of Administrative Services – Bureau of Enterprise Systems and Technology (DAS-BEST) and the CJIS Governing Board. This SLA should clarify roles, responsibilities, and service levels related to support of CJIS. This SLA will require a Memorandum of Understanding (MOU) with regards to funding that is predictable, reliable, and auditable, similar to the Pay-Phone Revenue Fund.

CJIS Governing Board's Business Goals and Objectives

Business Objectives	CISS[1]	CIDRIS	OBTS
Efficient modern business processes	✓	✓	✓
Open and transparent engagement	✓	✓	✓
Accurate and timely data for policy making, service delivery and results evaluation	✓	✓	✓
A secure and cost effective IT infrastructure	✓	✓	✓
Easily accessible services to all constituents	✓	✓	✓
Establish funding processes that will allow the State to measure and maximize its return on technology investments and to target funds to the agency and state priorities	✓	✓	✓
Ensure that the appropriate project management, transparency and accountability systems are in place for successful project implementation and completion	✓	✓	✓
Better align agency and state information technology plans and priorities with agency and state priority business and resources available	✓	✓	✓
Provide for Agency autonomy so they can accomplish their missions	✓	✓	✓
Simplify implementation of new technologies	✓	✓	
Develop secure environment, meeting State and Federal standards	✓	✓	✓
Optimize current investments to leverage infrastructure and resources	✓	✓	✓

1.0. CISS - Background

The **Connecticut Information Sharing System (CISS)** project's over-arching goal is to improve information sharing throughout the state's criminal justice community in order to comply with Public Act 08-01. CISS is a comprehensive, statewide system to promote the immediate and seamless sharing of information between all law enforcement and criminal justice agencies in Connecticut. Connecticut's criminal justice community consists of 11 justice agencies with over 23,000 staff members and utilizes 52 information systems to support its business needs. **CISS provides the first statewide unified information sharing system. This will allow the State to *do a lot more with a lot less by building on the frameworks established with OBTS and CIDRIS*.**

1.1. Key Accomplishments – Period Ending September 2011

The CISS Program has achieved several milestones:

[1] As designed in the RFP released on October 2010.

- On September 19, 2011, the State of Connecticut and the vendor Xerox – Affiliated Computer Systems, Inc State and Local Governments division signed the contract for CISS
- The CISS project contains ninety-four milestones over a sixty month period, and 320 named requirements which must be monitored by the CISS team
- The CISS project is currently recruiting for all key staff members. We expect the Program Manager and Business Managers to be on board in October with all but the Senior System Administrator on board in November.
- The CISS Project welcomed three new interns this fall who are assisting with requirements validation

1.2. Anticipated Activity – Next 180 Days

- CISS -- Complete Hiring Process of key CISS staff
- CISS -- Work with ACS vendor to start building the mandatory exchanges required by CISS
- Work with DAS-BEST to plan and stakeholder resources that will be needed in CISS
- Deploy Jazz software as the main requirements and quality tool for all the CISS applications
- Continue to deploy dashboards to provide transparency into CISS project

As these efforts are completed, the CISS program will need to resolve several risks and issues.

1.3. CISS Program Issues and Risks with Mitigation Strategy

- Issues:
- It is imperative that all project staff is hired for CISS to implement and maintain the project. ACS contract requires the team to notify vendor ten days in advance of any milestone assigned to the CISS team that will not be delivered on time. Without proper staffing and support of our stakeholders, considerable delays and penalties can be incurred by the State. These positions are crucial to meet the aggressive time set forth in our contract.
- The Budget technical adjustment request for 2012 and 2013 asked for operational funding for the OBTS project in fiscal year 2013. Currently the CISS bond is providing operational funding for both OBTS and CISS expenses. Continued reliance on CISS bond funds to pay for the ongoing expenses for the CISS and OBTS programs will exhaust the bond fund prior to the end of the project.

Conclusions

- The CISS project is at a key milestone related to funding
- CISS was undertaken to comply with Public Act 08.01. CISS will increase public and officer safety by significantly improving information sharing among the justice agencies in the State of Connecticut
- The system also enhances business efficiencies by increasing the amount and speed of information exchanged electronically

2.0. OBTS - Background

The **Offender Based Tracking System (OBTS)** is an integrated, information sharing system developed with all the state criminal justice agencies to respond to the growing demand for access to

comprehensive information on offenders. Officially launched in 2004, OBTS is used daily by local, state, and federal law enforcement as well as select state agencies.

2.1. Key Accomplishments – Period Ending September 2011

Following is the synopsis of the program's noteworthy accomplishments over this reporting period.

- The OBTS program completed its transition to the CIDRIS platform and uses the same operating system (Red Hat Linux) and database (Oracle 11g) thus reducing operating costs for OBTS by approximately $100,000 annually, while also increasing its performance
- Switched to a quarterly release schedule for delivery of improvements to the OBTS application
- Started construction of the deliverables for OBTS sprint Release 7.1
- Commenced the development of prototype reports and dashboards that rely on OBTS data

2.2. Anticipated Activity – Next 180 Days

The OBTS Program is expected to accomplish the following objectives or milestones over the next 180 days:

- OBTS -- Develop a Perpetual Data Quality Engine to monitor data sent to OBTS prior to accepting and storing the data
- Complete the OBTS Data Quality Project started by interns last summer
- OBTS -- Backfill vacant DAS-BEST CJIS Support Group (CSG) positions, currently five of eight positions are vacant
- Build, test, and deploy OBTS sprint releases 7.1 and 7.2

2.3. OBTS Application Release Schedule

The following release schedule is assumed over the coming twelve months. To provide a more stable and predictable product upgrade cycle for OBTS, the content of each maintenance release will be guided by the priorities identified by the OBTS / CIDRIS User Group. This group will meet quarterly to review program accomplishments, reassess program priorities and approve proposed release schedules.

Release Dates	Release Objectives
OBTS R7.1* February 2012	**Legislative Change Window** - In the event that there is no OBTS-impacting legislation, this will revert to a standard maintenance release.
OBTS R7.2 May 2012	**Maintenance Release** – Final release content to be determined based on the priorities of the OBTS community.
OBTS R7.3 August 2012	**Maintenance Release** – Final release content to be determined based on the priorities of the OBTS community.
OBTS R7.4 November 2012	**Maintenance Release** – Final release content to be determined based on the priorities of the OBTS community.

2.4. Program Issues and Risks with Mitigation Strategy

Issues:

An organizational change has left the CSG without adequate resources.

- **Impact** – The loss of key managers in the CSG means that the OBTS team will struggle to maintain the Release Schedule without slippage. Replacement staff will take time to come up to speed to a level where they can operate independently.
- **Mitigation** – The CJIS Executive Director is working with DAS-BEST, OPM, and the CJIS Governing Board, to identify appropriately skilled replacement staff.
-

Conclusions

With the transition to its new platform, OBTS has increased performance levels and data processing accuracy. Additionally, OBTS has garnered new users it is now positioned to enter into a disciplined and mature application maintenance lifecycle that provides for the incremental improvement of the application based on the priorities of the OBTS community.

Recommendations for the Board

Replacement of key OBTS business and technical resources is considered critical to ensuring that OBTS can provide effective business and technical support; a concern raised by the OBTS Application Steering Committee.

2.5. OBTS/CIDRIS Application User Group

The OBTS / CIDRIS Application User Group are charged with the responsibility to provide the Executive Director with program guidance and application governance that reflect the highest priorities of the OBTS user community. The CJIS team plans to kick-off the first meeting of the User group in the beginning of November at a date and time convenient to Chairperson Chief Douglas S. Fuchs. At this first meeting, we hope to accomplish creating a governance structure and priority requirements for OBTS Sprints 7.2 through 7.6. The membership of the committee is as follows:

Chairperson

Chief Douglas S. Fuchs

OBTS Data Consumers

Chief Richard Mulhall - Connecticut Police Chiefs Association (CPCA)

Michelle Cruz – Office of the Victim Advocate (OVA)

Andrew Mosley – Board of Parole (BOPP)

John Morrison – Office of the Public Defender (OPD)

Evelyn Godbout – Division of Criminal Justice (DCJ)

OBTS Data Suppliers

- **Offender Based Information System (OBIS)**
  Robert Cosgrove - Department of Corrections
  Lynn Milling - Department of Corrections
- **Master Name Index / Computerized Criminal History (MNI/CCH)**

Captain George Battle – Department of Public Safety
Joan Hilliard – Department of Public Safety

- **Judicial Information Systems (CRMVS, CIB, PRAWN, POR)**

  Terry Walker - Judicial Branch
  Larry D'Orsi - Judicial Branch

3.0. CIDRIS - Background

The **Connecticut Impaired Driving Records Information System (CIDRIS)** is an integrated, information sharing system developed in cooperation with local Law Enforcement, the Department of Public Safety, Department of Motor Vehicles, the Division of Criminal Justice, the Judicial Branch as well as NHTSA and ConnDOT. CIDRIS is currently in the Implementation Phase which it expects to complete with the integration of all 12 DPS troops by mid-December.

3.1. Key Accomplishments – Period Ending September 2011

Following is the synopsis of the program's noteworthy accomplishments over this reporting period.

- CIDRIS is making the final push to deploy all DESPP troops by mid-December. Final testing started in September and training of the DESPP troops started the last week of September
- The CIDRIS expansion program has made contact with several Computer Aided Dispatch / Records Management Systems (CAD/RMS) vendors, and has come back from these meetings with the suggestion that an Application Programming Interface (API) is required to reduce the cost to municipalities to connect the CAD/RMS systems to CIDRIS
- The Team plans to implement two additional layers of security, SSL certificates, and MQ authentications on the State's eDirectory
- Lt. Mark Sticca of DESPP created the NexGen/CIDRIS Training Program for the deployment of the State Police Troops

3.2. Anticipated Activity – Next 180 Days

- Deploy all DESPP police Troops
- Build Application Programming Interface to the CIDRIS application to allow to municipal CAD/RMS systems to interface with CIDRIS and other CJIS programs more efficiently
- Continue the expansion effort to local law enforcement
- DESPP's Records Management System still has several program areas that require completion. The timing of this programming and testing effort will go past the June 30, 2011 full deployment date for all DESPP troops. DESPP has agreed to finish the outstanding issues in a timely manner with their vendor NexGen in order to deploy the rest of the 11 troops.
- Dashboards allow information and data to be shown in graphic details. Free web applications, such as, Google Maps can be used to create dashboards. The CIDRIS Team took OUI information and incorporated it into dashboard technology. The result was a dashboard gauges showing OUI aging information and the number of OUIs
- The CJIS Governing Board's vision for the next six to twelve months is to go paperless for all OUI arrest and have the DESPP and Municipal Police Departments send OUI arrest information electronically. This will also allow the Judicial Branch and DMV to process OUI arrest information electronically

- Certify other CAD/RMS vendors serving the Connecticut Law Enforcement Community to allow their local Police Department clients to connect to the CIDRIS application.
- Continue pilot with judicial to send electronic copies of Summons, UAR, and Bond forms
-

Conclusions

CIDRIS continues its path to implementation of all DESPP State Police Troops and expects to have all troops on line and possibly some local law enforcement agencies as soon as DESPP's vendor NexGen completes the required programming they are currently working on. CIDRIS is also dependent on CJIS Support Group application maintenance once it is in production which is currently facing a staffing vacancy issue.

4.0. Perpetual Data Quality Engine (PDQE)

4.1. Problem Statement

Criminal Justice Information Systems (CJIS) must process messages quickly and accurately in as close to real time processing as possible. Law enforcement officers in the field rely on Connecticut's CJIS systems for accurate, timely, and potentially lifesaving (officer and public) information. It is vital for the administrators of these systems to have procedures which automatically monitor the accuracy of the data presented to the system from the various sources supplying data. The PDQE must perform in the background, without impeding message traffic to require accurate submission of data.

4.2. What is Perpetual Quality Assurance?

The functional attributes of perpetual quality assurance are items essential to use by external sources, stakeholder systems, as well as the CJIS applications. Business rules are functional requirements that determine what data is acceptable and the allowable formats and attributes.

However, in terms of software engineering, quality business rules can also be defined through a collection of the nonfunctional attributes in a system that helps to ensure its optimal condition. These include, but are not limited to performance, reliability, availability, usability, and maintainability.

Performance is seen through response times, transmission times, and memory use.

Reliability involves how often a task requested is completed accurately and the failure times. On the other end of the spectrum, availability is how responsive and available the system is.

Usability reflects if the system is doing what it is intended for, if it matches the expectations, and recognizing the differences if not.

Another key part of creating a monitoring system is factoring in the complexity of the code, known as maintainability[2]. A system created with complex code that demands high maintenance in turn affects the performance and reliability as it causes decreases in the speed and possible down time.

[2] Wong, G. 2005. Continuous Quality Monitoring of Software Systems. Doctoral Thesis, RMIT University. p. 7.

Each of these attributes can be measured in order to have a clear view of the overall accuracy of the system. While these attributes are not part of the functional operation of the system, quality assurance is a crucial component that is often overlooked. In a system that requires constant operation, such as the CJIS portfolio of applications, perpetual data monitoring is essential to its accuracy and effectiveness.

Theoretically, perpetual quality assurance is the process of continuously ensuring the data is at top quality. In relation to the CJIS projects, this theory is applied to software engineering through monitors used to audit the information databases of the state.

4.2.1. Overview of Data Monitoring

To track these nonfunctional attributes, a middle ware monitoring process needs to be launched. Monitors refer to certain processes used to maintain the integrity of the data; in this case, PDQE plays the role of monitoring the functional attributes. By quantifying the nonfunctional attributes of the system, a middleware monitoring tool can assess the accuracy and effectiveness of the system. These tools then become a duality providing comprehensive perpetual data quality monitoring. **Due to the nature of our system requiring near constant access, monitoring through a dual process model is essential to maintain the data**.

A dual process model is one in which a middleware monitoring process and a functional PDQE is launched in parallel to the main functional application. As mentioned previously, a complex code decreases the maintainability of the system, albeit with a dual process model each process has its own space and therefore minimizes code intrusion. [3] To manage the data, the monitor must be able to profile the database to begin identifying issues and then cleanse the data by applying business rules to pinpoint incorrect or missing data. The data can then be enriched by recognizing consistencies and adding necessary information (such as recognizing an address and adding a zip code to the data).

Next, the information will be matched and merged if it is discovered that multiple pieces of data involve the same subject[4]. With this automated process, data from every source system will be stopped if the monitor picks up a mistake before it is sent and distributed to the whole system. There are numerous benefits to dual process data monitoring including[5]:

- Continuous verification that failures and mistakes have been caught
- Separately coded monitors allow individual updates of each system through the cycle
- Separate development helps to prevent mistakes from being made in both the system and the monitor
- Problem identification and location in both the data and system
- Collection of information allows assessments of each source

[3] Wong, G. 2005. Continuous Quality Monitoring of Software Systems. Doctoral Thesis. RMIT University p.12.

[4] Lam, V. and Taylor, J. 2009. Enterprise Information Management (EIM): The Hidden Secret to Peak Business Performance. Information Builders, New York.

[5] Wong, G, 2005. Continuous Quality Monitoring of Software Systems. Doctoral Thesis. RMIT University. p. 58.

4.2.2. Objectives

Institute Monitors – To be able to gain optimal performance and reliability from the systems, data monitoring needs to be implemented in each of the three applications of CJIS. These databases include the Offender Based Tracking System (OBTS), the Connecticut Impaired Driving Records Information System (CIDRIS), and the Connecticut Information Shared System (CISS).

Make the Perpetual Quality Assurance Model Generic – The three CJIS databases are accessed by multiple agencies on the state and federal level. The Department of Motor Vehicles, Department of Corrections, court systems, Department of Public Safety, and various law enforcement agencies send information to the central systems at CJIS for the information to be stored and sent back to the rest of the recipients that require the data. With this constant need for accurate information, it is essential for each source to have the perpetual quality assurance model in use. Though each source uses a different program, the information is currently able to be redistributed because they are linked centrally by the National Information Exchange Model (NEIM) and XML language. However, due to the different programs in use, the dual process monitoring model needs to be generic. As this is completed, a standard of quality will be set and it can be trusted that information from each source has gone through the monitoring process. This standard will be set through the program answering the significant questions about a mistake that are essential to maintaining the integrity of the data.

The perpetual quality model attributes that need to be collected to define each fault[6] are as follows:

- Where did the error occur?
- When did it occur?
- What was observed as the symptoms?
- What was the result in the system due to the error?
- What was the mechanism through which it happened?
- What was the cause for the error?
- How severe was the error/how much was the user affected?
- How much did this fault cost?
- Was the fix required?

4.2.3. What is OBTS doing now for Perpetual Quality Management?

Currently, OBTS has a data modeling daily report on exceptions. When a message does not process correctly due to mistakes or offender linking errors, the exception is created along with a report of the track from which it originated. Although the system creates these messages, no one is currently reviewing the error messages to correct them. With this lack of monitoring, a small mistake in the input of information can greatly affect the future understanding of a record. For instance, if it is not reported correctly that a subject has been moved there is no way of knowing their true current location in the future when it is needed.

[6] Wong, G. 2005. Continuous Quality Monitoring of Software Systems. Doctoral Thesis. RMIT University. p. 167.

4.2.4. What is OBTS missing?

The OBTS system lacks feedback and a method of validating that the information was processed correctly when reported to the central database. Consequently, the source system does not have a means of correcting their mistakes due to a lack of coding. There is also no periodic system to update the database, which would be beneficial as another method of recognizing and correcting errors.

Many of these errors occur due to a lack of business rules written into the code that would catch the mistakes before they are entered. The current system does not recognize if there are multiple entries for the same offender or if keystroke errors have altered the data (misspelled names, identification numbers, identifying physical attributes, etc).

4.2.5. Creating the new program

4.2.5.1. How do we make it generic?

As stated earlier, the multiple arenas that report information to the central database are all independent of each other. There is no uniform government mandating operational aspects, such as the computer program, methods of reporting data, or budget. Each division not only uses different programs but also varies in the age of the technology and fiscal capabilities. With this said, the importance of the new program being generic and usable with older computers is of great importance. If this component is missing, the reliability of the data throughout the system is jeopardized.

The process of creating a generic program needs to start at the source systems. Since each CJIS, application, with the exception of OBTS, is NEIM compliant the new program must adhere to this language as its primary way of defining the data. The XML language is the current generic language used to report data to the OBTS database from each source system, thus the program needs to be XML compliant. Once the requirements of each source system have been analyzed, the generic code can then be created to satisfy each need.

4.2.5.2. What do we need?

As the type of code needed has been finalized in the previous step of making it generic, we must identify the business rules that need to be applied in the code in order to create the most efficient program. Using the resulting data from the project currently underway of analyzing the accuracy of OBTS, we can pinpoint the discrepancies of the data and use these to create business rules that will prevent mistakes in the future. For example, many issues in the current system result from offender linking errors due to mismatched State Identification numbers (SID). With this knowledge, a code can be created to recognize this mistake by linking other information on an offender and automatically correcting this error before distributing the information throughout the database to all the users.

These new business rules can be applied using a dual monitoring code. A code will be created that automatically double checks the information being entered from the source system. Because of the code, the data will be extracted and analyzed for its accuracy and ensured that everything is entered correctly. If an error is found, a message generates back to the source system and to OBTS and allows both parties to monitor that the correction is made.

In addition to immediate confirmation that data is correct or messages reporting its errors if not, it would be greatly beneficial to impose periodic confirmations between the source systems and the central database to ensure that the data is up to date and correct. This could be done through code that would direct the source system to send a regular report to the OBTS database. Instituting this process would be useful to ensure constant accuracy and confidence of information not just at the time of entry

but throughout the lifespan of the data. If there prove to be inaccuracies in data between the systems, the source system would then automatically send all the information it has regarding a specific record to the central database in order to determine the inconsistency.

4.2.6. How can we reach the final product?

The CJIS team plans to implement the PDQE phases utilizing the resources available now. The most important factor of this proposal is creating the ability to correct information from the source system. With this addition to the current process, source systems will be able to change information previously entered if they encounter a mistake.

4.2.7. Summary

Perpetual Quality Assurance is a theory using best practices methods to ensure that data is correct by continuously monitoring inputs for errors and updating current information. Though commonly overlooked as nonfunctional attributes in a computer system, quality assurance entails numerous components including the performance, reliability, usability, and maintainability of a database. These qualities may not be noticed in even the daily use of the system. However, if missing it soon becomes very apparent over a period incorrect data spawns more problems. In this event, the steps to identifying the source of the problem and correcting it can mean large monetary and time costs that could have been prevented had monitors been instituted.

In a database linking multiple source systems and accessed by numerous agencies, it is imperative that the information entered is correct and up to date. The current technology of the CJIS systems does not incorporate data monitoring into its code, therefore making our systems unable to recognize and address errors. Since each source utilizes different technologies and computer systems, we first need to create a generic code that can be accessed and created universally.

With a generic code in place, the needs can then be addressed. It is imperative that when entering new information an agency and CJIS be alerted to mistakes immediately in order to guarantee they are corrected before entering the database. These mistakes will be detected by the PDQE as it applies the business rules written into the code.

Furthermore, to keep the system as accurate as possible, source systems will automatically send periodic updates of their information to be matched and parsed with existing data. If discrepancies are found, they will be addressed by the source sending all relevant information to the central database.