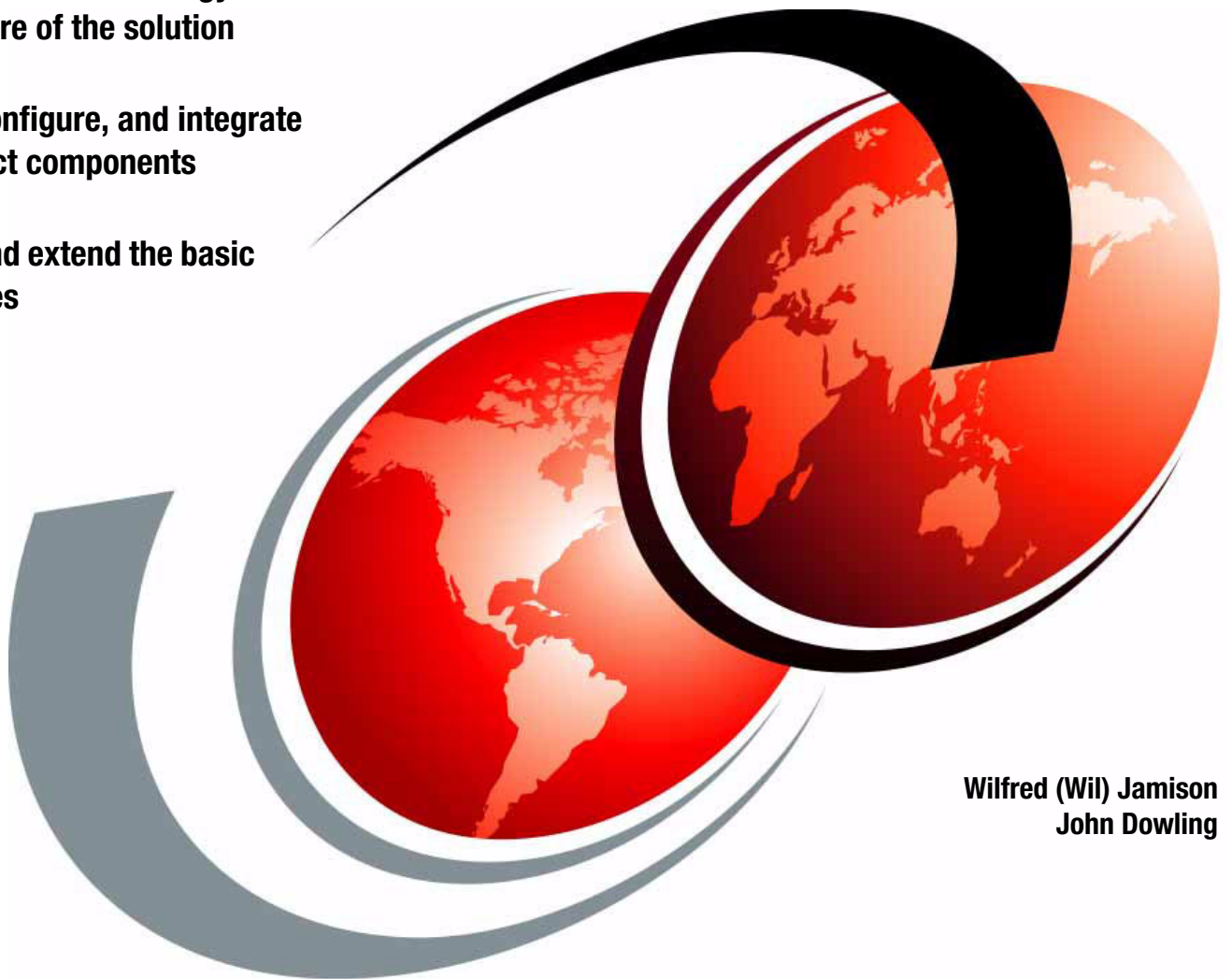# IBM i2 Integrated Law Enforcement:
Technical Architecture and Deployment Guide

Understand the technology and architecture of the solution

Deploy, configure, and integrate the product components

Expand and extend the basic capabilities

Wilfred (Wil) Jamison
John Dowling

Redpaper

IBM

International Technical Support Organization

IBM i2 Integrated Law Enforcement Deployment Guide

November 2014

Note: Before using this information and the product it supports, read the information in "Notices" on page vii.

First Edition (November 2014)

This edition applies to Version 1, Release 0 Modification 1 of IBM i2 Intelligent Law Enforcement (product number 5725-H93).

Contents

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

Cognos®	Netcool®	Sametime®
COPLINK®	PartnerWorld®	Smarter Cities®
DB2®	Passport Advantage®	SPSS®
developerWorks®	PureFlex®	Tivoli®
Domino®	Redbooks®	Watson™
i2®	Redguide™	WebSphere®
IBM®	Redpaper™	
IBM Watson™	Redbooks (logo) ®	

The following terms are trademarks of other companies:

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java, and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Other company, product, or service names may be trademarks or service marks of others.

Preface

IBM® i2® Integrated Law Enforcement is an IBM Smarter Cities® solution that addresses the needs of modern-day law enforcement agencies. It is a solution framework that provides the individual capabilities of the products that comprise the solution and extended capabilities developed through the synergistic integration of those product components.

As a framework, IBM i2 Integrated Law Enforcement allows for the continuous expansion of capabilities by putting together building blocks within the system and integrating with new, external systems. In doing so, an organization can respond and adapt to its changing needs. Simply stated, the configuration, integration, and implementation of IBM i2 Integrated Law Enforcement and its components provide the tools for more effective law enforcement.

This IBM Redpaper™ publication explains the technology and the architecture on which the solution is built. Most importantly, this paper enables technical teams to install, configure, and deploy an instance of the i2 Integrated Law Enforcement solution using the product i2 Intelligent Law Enforcement V1.0.1.

This paper is targeted to solution architects, system and deployment engineers, security specialists, data management experts, system analysts, software developers and test engineers, and system administrators.

Authors

This paper was produced by a team of specialists from around the world working at the International Technical Support Organization, Poughkeepsie Center.

Wil Jamison is a Solution Architect at IBM i2 development. He is based in the Research Triangle Park, North Carolina, US. Wil has 10 years of middleware development at IBM and six years of integration architecture work across the IBM software portfolio. Wil is an i2 Integrated Law Enforcement solution architect working with clients around the US and internationally. He holds a PhD in Computer and Information Science from Syracuse University with expertise on distributed systems and distributed artificial intelligence.

John Dowling is a Software Infrastructure Specialist with the IBM i2 development organization in Dublin, Ireland, where he is a member of the i2 Integrated Law Enforcement development team. John has over 25 years experience in the IT industry and has been working with IBM middleware for over 15 years. John is a member of the Institution of Engineering and Technology in the UK and holds Chartered IT Professional status with the British Computer Society.

The project that produced this publication was managed by **Marcela Adan**, IBM Redbooks® Project Leader - IBM International Technical Support Organization, Global Content Services

Thanks to the following people for their contributions to this project:

Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an ITSO residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our papers to be as helpful as possible. Send us your comments about this paper or other IBM Redbooks publications in one of the following ways:

► Use the online **Contact us** review Redbooks form found at:

ibm.com/redbooks

► Send your comments in an email to:

redbooks@us.ibm.com

► Mail your comments to:

IBM Corporation, International Technical Support Organization
Dept. HYTD Mail Station P099
2455 South Road
Poughkeepsie, NY 12601-5400

Stay connected to IBM Redbooks

► Find us on Facebook:

http://www.facebook.com/IBMRedbooks

► Follow us on Twitter:

http://twitter.com/ibmredbooks

► Look for us on LinkedIn:

http://www.linkedin.com/groups?home=&gid=2130806

► Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm

► Stay current on recent Redbooks publications with RSS Feeds:

http://www.redbooks.ibm.com/rss.html

1

Integrated Law Enforcement system overview

IBM i2 Integrated Law Enforcement is an IBM Smarter Cities solution offering that addresses the needs of modern-day law enforcement agencies. It is a solution *framework* that is backed by two capability sources:

► *Individual capabilities* of products that comprise the solution

► *Extended capabilities* developed through the synergistic integration of these product components

As a framework, i2 Integrated Law Enforcement allows for the continuous expansion of capabilities by putting together building blocks within the system and integrating with new, external systems. In doing so, an organization can respond and adapt to its changing needs. Simply stated, the configuration, integration, and implementation of i2 Integrated Law Enforcement and these components provide the tools for more effective law enforcement.

Much has been written about the high-level concepts and business value proposition of i2 Integrated Law Enforcement. These publications are mostly targeted to executives and decision makers. The IBM Redbooks publication *Integrated Law Enforcement: A Holistic Approach to Solving Crime*, REDP-5116, delves into the motivation behind i2 Integrated Law Enforcement and showcases the various capabilities from a business solution standpoint.

This IBM Redpaper publication explains the technology and the architecture on which the solution is built. Most importantly, this IBM Redpaper enables you and your teams to install, configure, and deploy an instance of the i2 Integrated Law Enforcement solution. If you are a solution architect, system or deployment engineer, security specialist, data management expert, system analyst, developer or test engineer, or system administrator, this paper is written with you in mind.

This chapter explains the architecture of i2 Integrated Law Enforcement. It highlights the components that make up the solution and how they work together.

1.1 Architecture

A clear understanding of the i2 Integrated Law Enforcement architecture and how it interacts with its environment is the key to a successful implementation. The goal of this section is to provide you with a foundation, so that all of the technical procedures presented in this book will make more sense, as they tie back to the architecture discussion. This conceptual connection is helpful when reasoning about and troubleshooting the system.

1.1.1 System context

The System Context Diagram in Figure 1-1 shows the system of interest, i2 Integrated Law Enforcement system. The diagram provides the highest level of abstraction and emphasizes the external entities that interact with i2 Integrated Law Enforcement.
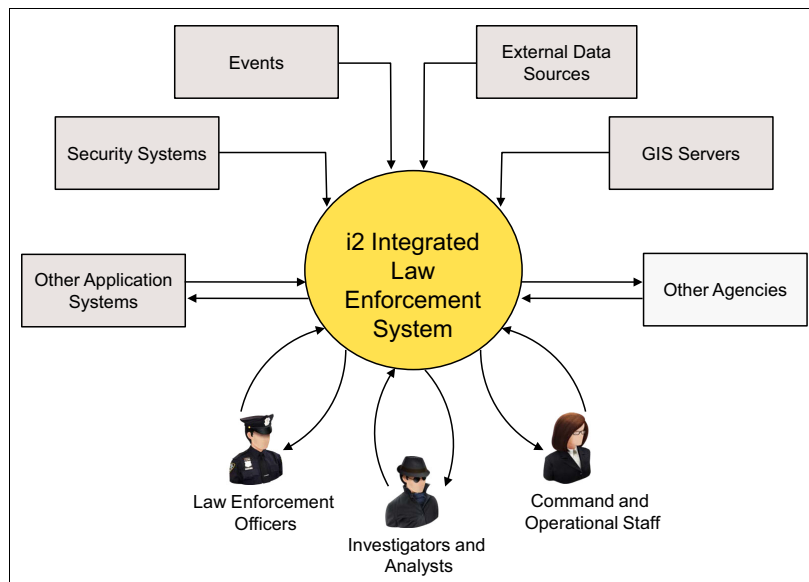
Figure 1-1   i2 Integrated Law Enforcement system context diagram

Two primary types of data feed into the i2 Integrated Law Enforcement system: static data and dynamic events. The major sources of static data for law enforcement and intelligence agencies emanate from various external sources in a data store. Customarily, these agencies ingest this data on regular basis or pull it on demand. This approach allows them to create their own data collection for their consumption.

Dynamic events, however, arrive in a more random fashion. They include notifications, 911 data, situational alerts, and so on. The successful operation of a law enforcement agency depends heavily on the wealth of their static data and their ability to capture and respond to dynamic events.

> **Note:** Both static data and dynamic events can originate from internal or external sources. Most data in law enforcement comes from external data sources. The main differences between static data and dynamic events are time and direction. Static data comes in typically on a schedule and sometimes on demand. It is mostly ingested or pulled into the system. Some examples are arrest data, criminal records, and incident reports. Alternatively, dynamic events are typically real-time events; they arrive at anytime and they are pushed to the system. Common examples of events are 911 calls, dispatch data, notifications that might contain simple texts, video, or audio streams.

Due to the nature of law enforcement, i2 Integrated Law Enforcement has to establish serious integration with the security systems of an agency. Meanwhile, with the advancement of technologies in the areas of analytics, information delivery platforms, and visual presentation, Geographic Information Systems (GIS) has become pervasive in public safety. Therefore, i2 Integrated Law Enforcement includes a capability that takes advantage of the use of such systems.

As a framework, i2 Integrated Law Enforcement is designed to interact with other application systems, especially when such interaction is a primary case point for enriching the system. For example, an agency might already have an existing solution for reading license plates. A use case scenario is that when there is a hit in the list of wanted license plates (for example, for stolen vehicles, or vehicles associated with a fugitive), the application sends a notification to i2 Integrated Law Enforcement, which, in turn, sends an alert to a group of commanding officers. A map is displayed in a portal where the alert is read showing where the car was spotted and additional actions can be decided from there on.

Related to interaction with a third-party system is the ability to work with other agencies, both inside and outside of the law enforcement arena. This is going a step further with the notion of collaborative operation where agencies combine resources and forces to solve a problem. For example, the commanding officers can contact the neighboring agencies from other counties to intercept the said vehicle and apprehend the subject.

The most important elements shown in Figure 1-1 on page 2 are the direct users who interact with i2 Integrated Law Enforcement. There are three broad types of user communities:

► The *law enforcement officers* include *front-line* officers who are directly faced with situations where public safety is at stake. They include patrol officers, lieutenants, sergeants, captains, and others. This type also includes desk officers whose role is to support fellow officers administratively or non-administratively.

► The *investigators and analysts* can be in the back-office performing search and data analysis, or they can be in the field gathering and validating information or working with other law enforcement officers. The investigators and analysts support each other. Both apply their analytical skills to figure out relationships among seemingly unrelated information. They have special skills of filling the gaps or the missing links that can potentially solve crimes.

► The *command and operational staff* is another group that is composed of supervisors, commanders, and operational personnel. These professionals are responsible for the day-to-day operations of the agency, such as receiving 911 calls and dispatching responders. Some are members of management with additional responsibilities and accountabilities, and with the authority to issue commands and protocols.

Note: i2 Integrated Law Enforcement provides a cohesive solution framework for the law enforcement communities where they can customize, extend, and enhance their own capabilities by integrating with other systems and solutions. This approach enables them to be proactive in responding to their own specific needs.

With this broad spectrum of users, systems, data, and services, i2 Integrated Law Enforcement is a dynamic and powerful solution.

1.1.2 System architecture

Going one level down from the system context, Figure 1-2 is the system architecture of i2 Integrated Law Enforcement, which shows the high-level internal components. It follows a layered architecture that cleanly separates the presentation, processing, and data layers.
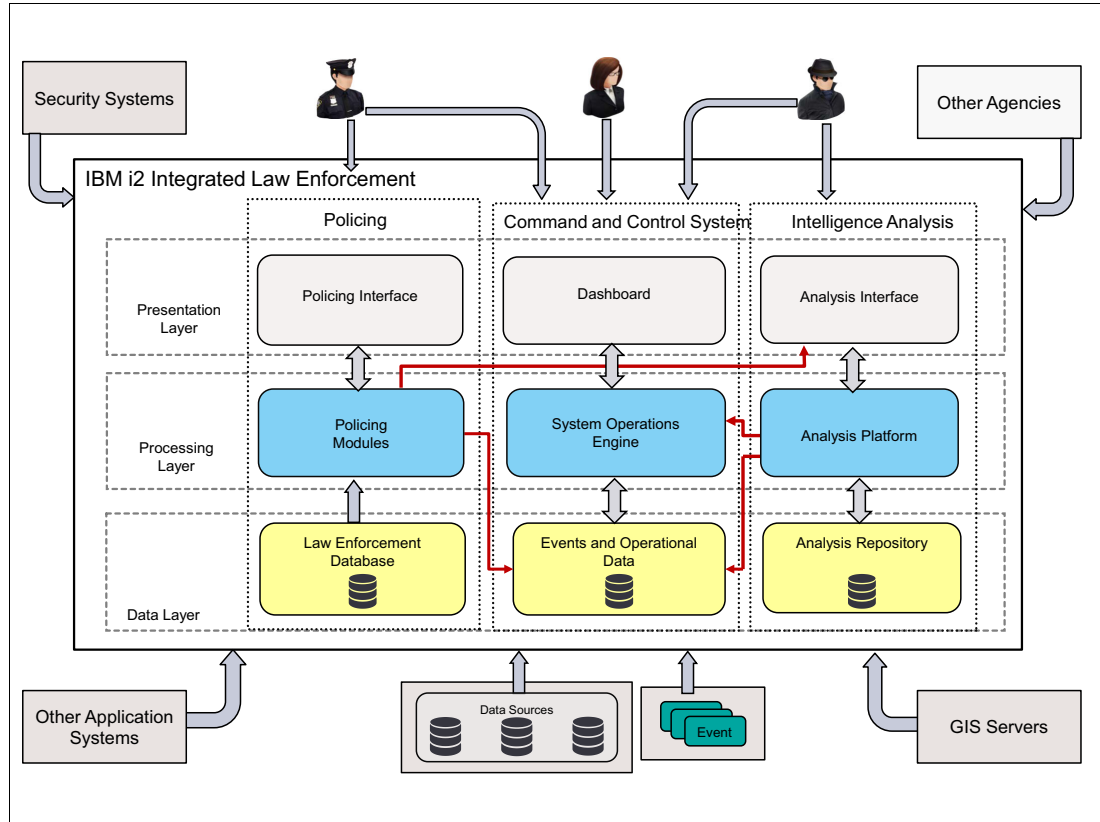
Figure 1-2   IBM i2 Integrated Law Enforcement system architecture

The different layers help you to focus on a single aspect of the system one at a time. A quick description of each layer follows.

The *presentation layer* provides the front end for interfacing directly with external entities, such as users and other applications.

This layer does not tie the system to any specific technology for interfacing with the user or external entities. For example, the presentation layer does not require specific web interfaces, desktop interfaces, or even mobile interfaces, but they all remain as possibilities.

The *processing layer* is where the computation happens. This layer includes all implementation modules that contain the computational and transformational logic and the data processing for the entire system. These complicated subsystems consist of several interacting modules, frameworks, and products.

The *data layer* is a layer of high importance for i2 Integrated Law Enforcement. Law enforcement is heavily dependent on useful data. Therefore, this layer is responsible for the collection of data that is used by the system. Most data is ingested through an extract, transform, and load (ETL) process, which means that data is retrieved from the sources, then cleansed, processed, or transformed to a format recognized by the target system and then finally sent or loaded to i2 Integrated Law Enforcement system. This process does not imply a single storage for all the data. It is typical to have a couple of separate repositories for manageability, scalability, disaster recovery, data specialization, ease of integration, and so on. Events are data in motion, and so, they are received when they come into the system. Therefore, a queue is typically used to collect the events, which become part of the operational data in the system. Operational data can be produced from other parts of the framework.

The data layer is important in the architecture because it is meant to describe a holistic and consolidated view of information flowing into and consumed by i2 Integrated Law Enforcement. In this paper, it is referred to as the *Public Safety Information Hub.* When used, it refers to the collective repositories within the data layer.

i2 Integrated Law Enforcement serves the needs of the three user communities described in 1.1.1, "System context" on page 2. Looking closely at Figure 1-2 on page 4, there are three vertical slices or *pillars* in the architecture that cater to these user types. The pillar on the left is specialized for the law enforcement officers. The next pillar is for all of the three user types, and finally, the third pillar is specialized for the investigators and analysts. Each pillar fulfills the following capabilities:

► Policing: This pillar provides capabilities for law enforcement officers in generating investigative leads and developing links, associations, and crime analysis. The key to this empowerment is a rich, consolidated, and up-to-date criminal data warehouse (depicted in Figure 1-2 on page 4 as the Law Enforcement database) that is easy to access, responsive, and reliable. The different policing capabilities push information to the tactical edge to provide situational awareness and information for investigative lead generation. The policing capabilities provide the following functions:

  – Data integration from disparate information systems, without redundant, manual entry of data. Integrated data is refreshed (updated) on a schedule that is determined by the contributing agencies. Individual agencies control the data that is integrated. Integrated data allows advanced analysis by using artificial intelligence-based searches.

  – Monitoring, collaboration, and notification to assist in ongoing investigations.

  – Security through role-based system access of authorized users.

  – A regional node concept that permits queries between agencies across jurisdictions.

  – Connection to external data sources for extended queries.

  – Visualization tools for developing networks, statistical patterns, and GIS-mapping displays.

  – Real-time notification of events, based on user-defined tasks and thresholds.

► Command and Control System: This pillar provides situational awareness and information for decision support to all three user communities. It can also provide status about key performance indicators (KPIs) and events in near real time, within a common operational picture that provides these functions:

  – Assists command staff with making better decisions, based on a single source of trusted, consolidated information.

  – Helps ensure that crime-reduction operations meet their targets with repeatable, accurate, and timely information.

- Provides crime trend analysis and allows the quick *development of crime patterns* by answering specific questions, such as "*Where are most vehicle thefts occurring on Monday mornings?*"
- Increases confidence for key tactical and strategic decisions.
- Removes the manual effort of reporting, allowing *reallocated resources* to perform other functions.
- Supports command staff briefings of status against goals (KPIs).
- Uses criminal behavior patterns to quickly and confidently *support strategic and resource deployment* decisions.

► Intelligence Analysis: Intelligence analysts and investigators often engage in lengthy and complex analytical tasks requiring specialized skills. In this pillar, analysts are able to share and collaborate on the gathering, analysis, and dissemination of intelligence, not only with each other, but with command staff, officers in the field, and other jurisdictions. These capabilities identify key targets, associations, commodity flows, and complex networks and can help you achieve these functions:

- Build a single common intelligence picture.
- Develop a clear operational view of threats that are being tracked.
- Identify emerging threats to enable decision makers to choose an appropriate response.
- Concisely present fact-based information to decision makers, courts, or other bodies.
- Develop intelligence packages that outline timeline progressions, cause and effect, and criminal network weaknesses and strengths.
- Collaborate with colleagues and maintain continuity over long-running investigations.

The overall design principle is to have an architecture that uses the individual capabilities of each pillar and then extend these capabilities through integration across different pillars and external application systems. Each pillar might appear specialized, but they are not operating independently from each other. On the contrary, they interact together to create a more cohesive system, one that integrates various features horizontally to produce a more powerful set of capabilities. In Figure 1-2 on page 4, these interactions are depicted by the red lines that cross the verticals. In later sections, you will discover more details about what these interactions accomplish.

> **Note:** This architecture excludes some aspects of the system that are important but do not constitute the core of i2 Integrated Law Enforcement. For example, security and performance, which are not functional in nature, are not depicted in the diagram. Chapter 4, "Integrated Law Enforcement security" on page 107 is devoted to security.

1.1.3 Component architecture

This section brings you closer to the components of the architecture that provide additional details about their integration to produce more extended capabilities. In a later section, you will see the functional descriptions of these extended capabilities.

Figure 1-3 on page 7 dives one level deeper by focusing on the components and subcomponents that are involved in the interactions shown in Figure 1-2 on page 4. The vertical boundaries are removed to draw your attention to the interactions of components, regardless of the pillars they belong to.
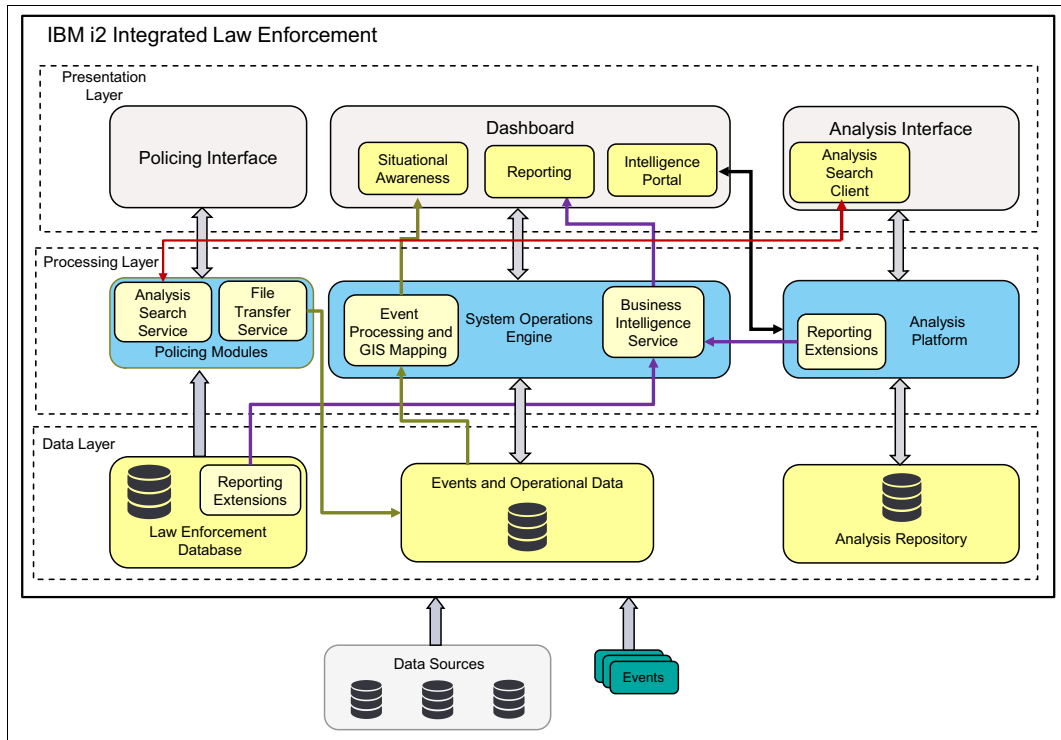
Figure 1-3   i2 Integrated Law Enforcement component architecture

As described in 1.1.2, "System architecture" on page 4, each pillar enables its target user with specialized capabilities. In this section, four *extended capabilities* that are delivered with the i2 Integrated Law Enforcement solution are introduced. At the presentation layer in Figure 1-3, you see four components marked in yellow that correspond to these four extended capabilities. Each of these capabilities is described:

► Analysis search: This extended capability enables an analyst or an investigator to search for information that is stored in the law enforcement database. To accomplish this capability, two additional components are introduced called the *analysis search service* and the *analysis search client.* The search client sits on the Analysis Interface, where it lets a user enter search terms and then passes them to the search service. See the red line in Figure 1-3. When the analysis search client receives the results from the analysis search service, they are rendered and shown to the user in a manner that is consistent with the Analysis Interface. From there, the user can manipulate the new information in any way that is provided by any of the tools that come with the Analysis Interface. The analysis search service is a component that is added to the policing module to search the law enforcement database using the search terms provided by the analysis search client. The analysis search service also converts the results to a format that the Analysis Interface can understand.

► Intelligence portal: In the dashboard is a widget (or portlet) that contains an intelligence portal. This user interface allows most analysts and investigators to explore the contents of the analysis repository. Remember that the repository contains a wealth of intelligence data that can be used to identify patterns and association, and possibly, to discover potential threats, such as a nexus of terrorism or organized crime. This enables an agency to potentially deter and thwart a dangerous situation from happening. In addition to exploring the repository, users can add new or update existing intelligence data, allowing them to contribute and collaborate with one another. This kind of collaboration is essential to improving the quality and breadth of intelligence that an agency owns. As shown by the black line in Figure 1-3 on page 7, the intelligence portal interacts directly with the analysis platform.

► Reporting: This is also available in the dashboard as its own widget, where a user can generate two types of reports depending on which type of data the report is using. The first report contains intelligence data, and the second report contains crimes, suspects, and other law enforcement-related information. Therefore, the first report retrieves information from the analysis repository, and the latter retrieves information from the law enforcement database. In both cases, a separate tool is needed to gather the data, use a reporting template, apply formatting for every data element, render a summary or details, and so on. Typically, a business intelligence service implements this type of function, as shown in Figure 1-3 on page 7. The finished reports are then rendered by the Reporting interface in the dashboard. See the purple lines in Figure 1-3 on page 7.

► Situational Awareness: This extended capability brings the most value for a law enforcement agency. With situational awareness, users are able to assess the situation in a given location based on information that they are given about the surrounding situation. This assessment and interpretation can be used for deciding the most sensible *next action*. The needed information depends on the end goal or purpose for being in that location. The situational awareness capability in i2 Integrated Law Enforcement provides information about the *occurrences of crimes* – a concrete indicator of how safe that given location is relative to its surroundings.

In IBM i2 Integrated Law Enforcement, a geographic map on a portal shows the surrounding areas of an agency. The areas are annotated with different types of icons indicating the occurrences of certain types of crimes in the vicinity. As a user, you can choose to look at the details of a crime occurrence, such as the perpetrator, victim, exact location, and date and time of the incident. You can also filter out information, based on specific crime types or historical time period. For example, you can limit the crimes to those that happened in the last two days or in the last six months. This capability can be used in many ways, such as spending more resources on certain locations to reduce crime (this is an example of a *next action*).

A policing module exports information about crimes from the database to files. A file transfer service component is introduced to move these files to where they are converted into events as shown in Figure 1-3 on page 7. These events flow to an event processing module; it involves calculating geolocation information in addition to other details about the crimes. The crimes are then pushed to the Situational Awareness module, which renders the information in a geographic map.

Part of your activities when deploying i2 Integrated Law Enforcement is to install the components that correspond to these extended capabilities and to configure them in such a way that the links that tie them together are enabled correctly. 1.1.4, "Product implementation" on page 9 introduces the IBM products that make up the implementation of this architecture. In addition, the section shows which parts of these products are used for the different architectural components.

1.1.4 Product implementation

i2 Integrated Law Enforcement V1.0.1 is a solution framework that consists of three major IBM product portfolios:

► IBM i2 COPLINK® V4.8 (i2 COPLINK): Reference:

  https://pic.dhe.ibm.com/infocenter/coplink/v4r8m0pwd/index.jsp

  Note: Authorization to access this site is required.

  IBM i2 COPLINK Analysis Search V4.8

► IBM Intelligent Operations Center V1.5 (IOC): Reference:

  http://pic.dhe.ibm.com/infocenter/cities/v1r5m0/index.jsp

► IBM i2 Intelligence Analysis Platform V3.0.3: Reference:

  http://www-03.ibm.com/software/products/en/intelligence-analysis-platform

  – IBM i2 Analyst's Notebook Premium V8.9.3: Reference:

    http://www-03.ibm.com/software/products/en/analysts-notebook-premium

  – IBM i2 Information Exchange for Analysis Search for Analyst's Notebook V8.9.1: Reference:

    http://www-03.ibm.com/software/products/en/information-exchange-analysis-search

A complete breakdown of these products is also provided in Chapter 2, "Integrated Law Enforcement deployment" on page 21.

Figure 1-4 summarizes where these products are mapped to in relation to the system architecture shown in Figure 1-2 on page 4:

► The entire suite of i2 COPLINK products maps to the policing pillar.

► Intelligent Operations Center maps to the Command and Control System pillar.

► IBM i2 Intelligence Analysis Platform with IBM i2 Analyst's Notebook and IBM i2 Information Exchange for Analysis Search for the Analyst's Notebook are all mapped to the Intelligence Analysis pillar.
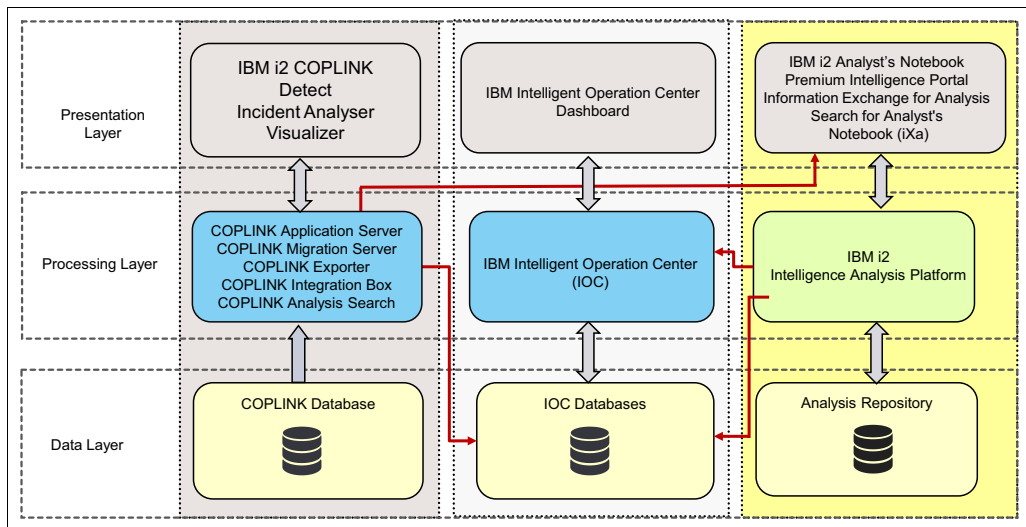
Figure 1-4   IBM product mapping to i2 Integrated Law Enforcement components

In the next sections, you will get a better understanding of the architecture of these three major products.

i2 Intelligence Analysis Platform

To be effective in improving public safety, a law enforcement agency must have good intelligence gathering, analysis, and sharing capabilities. The i2 Intelligence Analysis Platform enables its target users (the analysts and investigators) to carry out their tasks more efficiently and cooperatively.

The architecture of i2 Intelligence Analysis Platform is shown in Figure 1-5. The i2 Intelligence Analysis Platform is an enterprise server that runs on the IBM WebSphere® Application Server platform. It is based on service-oriented architecture and Java Platform, Enterprise Edition (J2EE) technologies. Therefore, client applications can be written to talk to i2 Intelligence Analysis Platform through its public web services application programming interface (API). Two examples of a client application are the desktop product, i2 Analyst's Notebook Premium, and the web-based client, Intelligence Portal. These are client tools that analysts and investigators use to perform their deep and systematic analytical tasks.
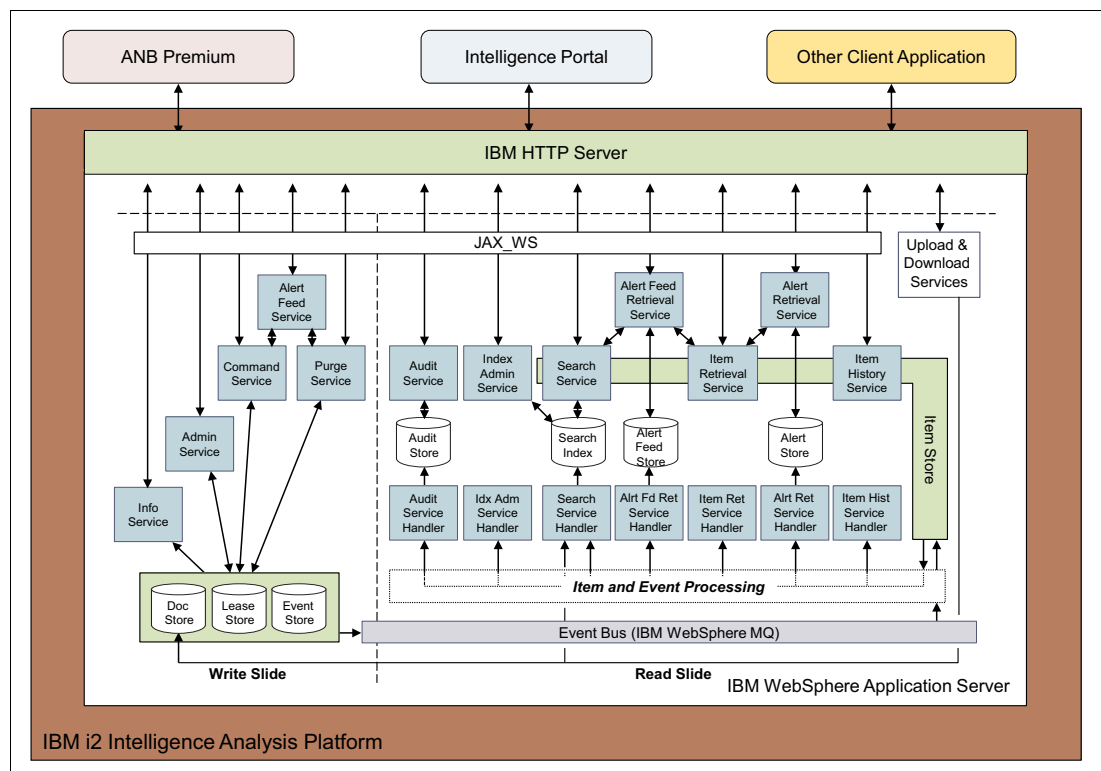
Figure 1-5   i2 Intelligence Analysis Platform architecture

As a server, the i2 Intelligence Analysis Platform provides the back-end support including storing analysis data into a centralized analysis repository (AR), which enables users to search data, share with other people using the i2 Intelligence Analysis Platform security model, create sets and charts, and provide versioning, auditing, and so on. See the different services that are provided by the platform in Figure 1-5. The AR is served by a database server, such as IBM DB2®, and, as you can see in Figure 1-5, the AR is made up of a few databases. Each one has its own special purpose and function as indicated by the labels in Figure 1-5.

The i2 Intelligence Analysis Platform is based on the Command Query Responsibility Segregation (CQRS) architecture. This means that, at a high level, the write and read data operations are handled separately by two different servers, referred to as the *command* (or *write*) server and the *query* (or *read*) server. The two servers talk to each other asynchronously, using events from Java Message Service (JMS). This is implemented by IBM WebSphere MQ. Therefore, the i2 Intelligence Analysis Platform is composed of two applications: one for the write server, and another for the read server.

The i2 Intelligence Analysis Platform is also extensible because all discrete functions are implemented as services. Figure 1-5 on page 10 shows all the native services that come with the platform. The i2 Intelligence Analysis Platform provides tools to enable developers to write their own services in Java, for example, a service that sends a notification to a user when a particular user logs in. Chapter 3, "Integrated Law Enforcement cross-component stitching" on page 45 describes how i2 Integrated Law Enforcement uses the extensible services framework of the i2 Intelligence Analysis Platform to implement reporting.

Finally, the i2 Intelligence Analysis Platform uses an IBM HTTP Server as the single point of interface with its clients. The HTTP server knows whether to route a web service request to the write server or the read server.

IBM Intelligent Operations Center

At a high-level, the overriding description of the IBM Intelligent Operations Center (IOC) is that it is a general solution that integrates and uses disparate data from multiple sources, where it applies analytics to make sense out of them and visualize them in a unified view for the user. By addressing the handling of voluminous data, IBM Intelligent Operations Center simplifies data consumption and presents it in a way from which is easier for users to make informed decisions.

IBM Intelligent Operations Center caters to many different types of users, where each type is interested in different information, from possibly the same data sources. Therefore, the Intelligent Operations Center makes the most out of the data that it already has.

By directing large quantities of data to a structured format using rule-based data flow, IBM Intelligent Operations Center generates various reports and key performance indicators (KPIs). Data feeds that are done through events also include processing data in real time, and sending notifications based on occurrences of events is a functionality that can satisfy many use cases. To many, IBM Intelligent Operations Center is well-known for its uniform and configurable web-based interface that is role-based. Collaboration is achieved by working on the same underlying data but with different views. IBM Intelligent Operations Center can also streamline activities by defining workflows or standard operating procedures (SOPs).

Overall, IBM Intelligent Operations Center is a system-wide engine that provides many features that can serve as the foundation for more specific and vertical applications, such as water, fire, transportation, and public safety. The common theme is that these vertical applications require situational awareness that can be achieved by assimilating pertinent data, monitoring, analyzing, and reacting to situations as presented on a clean and easy-to-use dashboard. The goal is to make city operations more efficient and effective.

To achieve this goal, IBM Intelligent Operations Center uses many of the existing software capabilities that IBM already has. Figure 1-6 on page 12 shows the system architecture of IBM Intelligent Operations Center, including the different products that are embedded and how they work together. The IBM Intelligent Operations Center subsystem layer is where all of the inner processing of data and events from the various sources occurs. IBM WebSphere Message Broker is the enterprise service bus (ESB) that interacts with external systems. These systems push information as events.

IBM Netcool® is the event management system that controls how these events are to be processed and sent to other services, such as the persistent store manager IBM DB2, IBM Business Monitor for KPI management, IBM Tivoli® Service Request Management for workflows, and IBM Cognos® Business Intelligence (Cognos BI) for reports.
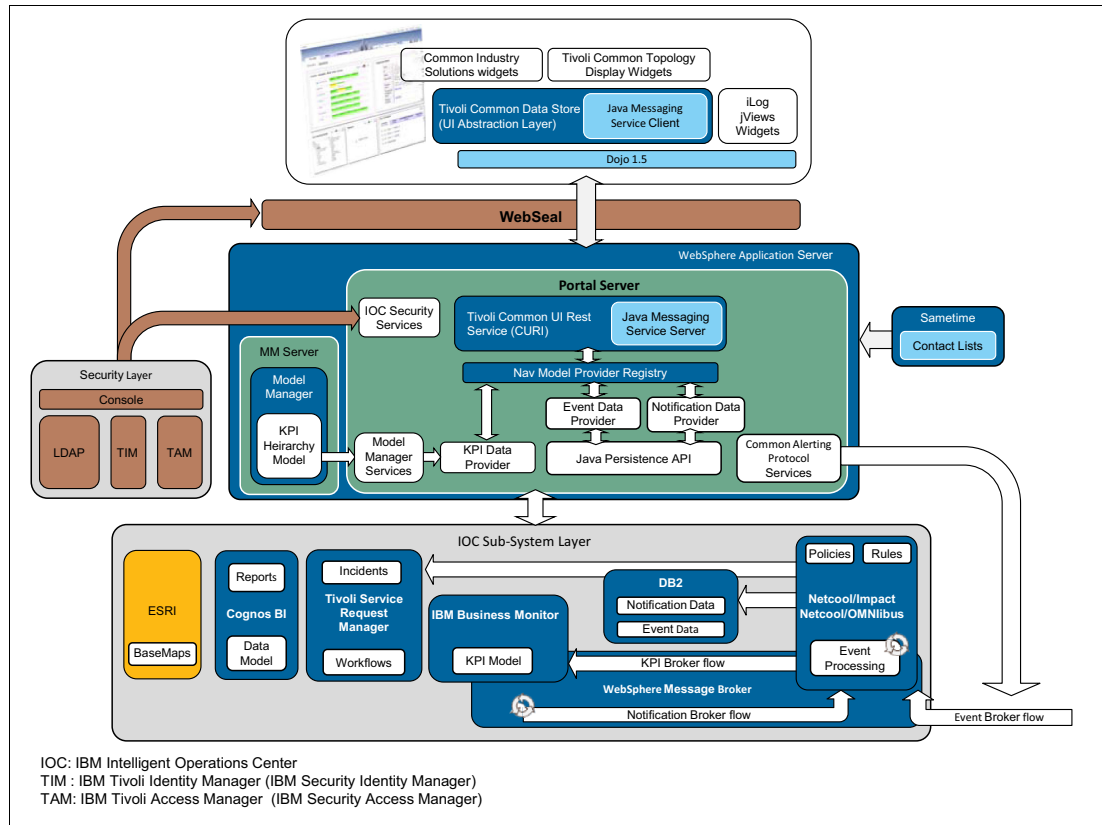
Figure 1-6   IBM Intelligent Operations Center system architecture

The application server engine (through the IBM WebSphere Application Server) sits between the user and the data and events. Here, the IBM Portal Server is the engine responsible for overall presentation through the management of portals and portlets. The security layer is an important module in this architecture, which controls access to the Intelligent Operations Center. The following products are responsible for enforcing security: IBM Tivoli Directory Server, IBM Tivoli Identity Manager, IBM Tivoli Access Manager, and IBM WebSEAL. Collaboration is also provided by IBM Sametime®. Finally, UI technologies were also used, including iLog views, Dojo 1.5, and IBM Tivoli Common Data Store.

The IBM Redpaper *IBM Intelligent Operations Center for Smarter Cities*, REDP-4939, provides an overview of IBM Intelligent Operations Center V1.5.

i2 COPLINK

The IBM Redguide™ *Integrated Law Enforcement: A Holistic Approach to Solving Crime*, REDP-5116, highlights the value that i2 COPLINK brings to the field of law enforcement. i2 COPLINK is the cornerstone of many modern-day solutions for day-to-day police operations. By consolidating data from hundreds of sources into a single repository, a police force is able to access historical and real-time records that can assist in generating investigative leads and developing links, associations, and crime analysis.

i2 COPLINK is the specialized engine that delivers all the features and capabilities of the policing pillar described in 1.1.2, "System architecture" on page 4. The successful operation of this engine lies in the various components that make up the architecture of this technology, as shown in Figure 1-7. The heart of this architecture is the i2 COPLINK data in the data tier. In the import tier, external data sources are processed and transformed by the use of various mapping operations into a document-based data model, which is then persisted for other components to use. The power of i2 COPLINK is in its dynamic querying capabilities that perform crucial search operations on the data.

As shown in Figure 1-4 on page 9, i2 COPLINK exposes different types of user interfaces, depending on the type of information a user needs from the i2 COPLINK data. For example, i2 COPLINK detect function can provide views for different entities and information about people, vehicles, guns, and other objects. i2 COPLINK COMPSTAT presents statistics about different types of crimes, events, incidents, and so on.
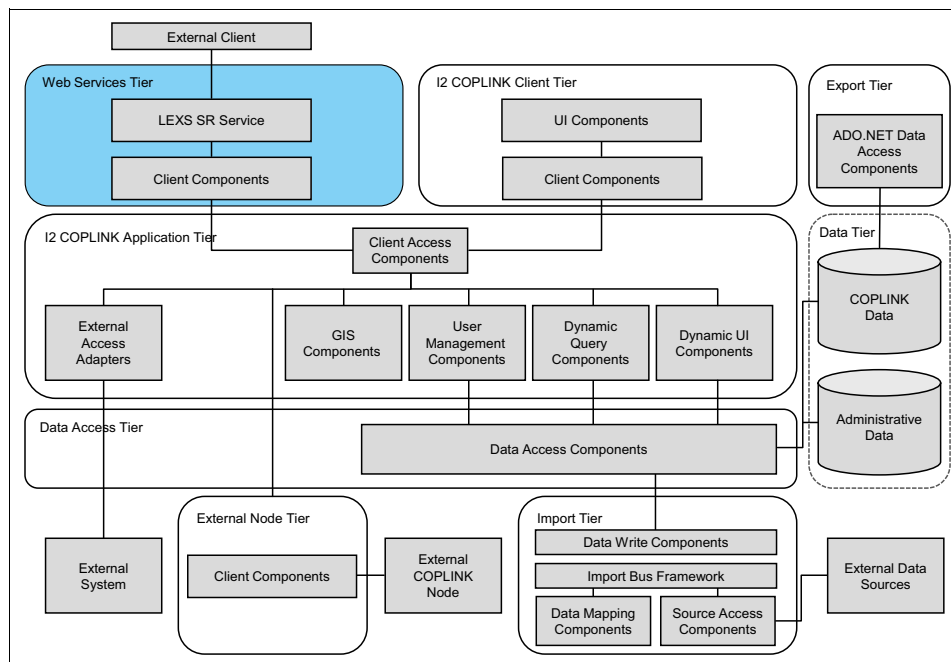
Figure 1-7   i2 COPLINK product architecture

Understandably, access to i2 COPLINK data is highly protected because it contains sensitive information about crimes, fingerprints, photographs, and other information about people who might or might not be involved in crimes, especially the juveniles and sex-related crimes and offenses. The product is subject to compliance with the Criminal Justice Information System (CJIS) to ensure that information stored in the i2 COPLINK data store is protected from cyber crime.

Currently, the only access to i2 COPLINK data is by using database views. Chapter 3, "Integrated Law Enforcement cross-component stitching" on page 45, describes this mechanism by which i2 COPLINK shares information with other components within the i2 Integrated Law Enforcement solution.

1.1.5 Functional architecture

1.1.3, "Component architecture" on page 6 describes the three major product portfolios that compose IBM i2 Intelligent Law Enforcement V1.0.1 and the four major extended capabilities of these products.

Note: IBM i2 Intelligent Law Enforcement V1.0.1 is the IBM product that implements the i2 Integrated Law Enforcement solution framework.

Figure 1-8 describes how the products that comprise IBM i2 Intelligent Law Enforcement V1.0.1 are mapped to the extended capabilities:

► Red line: Analysis search
► Black line: Intelligence portal
► Purple line: Reporting
► Green line: Situational awareness

Figure 1-8 describes the components that are involved when connecting (also called *stitching*) and configuring these capabilities. For more details, see Chapter 3, "Integrated Law Enforcement cross-component stitching" on page 45.

Note: Stitching is the process of configuring and connecting parts of two or more products to enable transactions of a new capability that crosses the boundaries of these products.
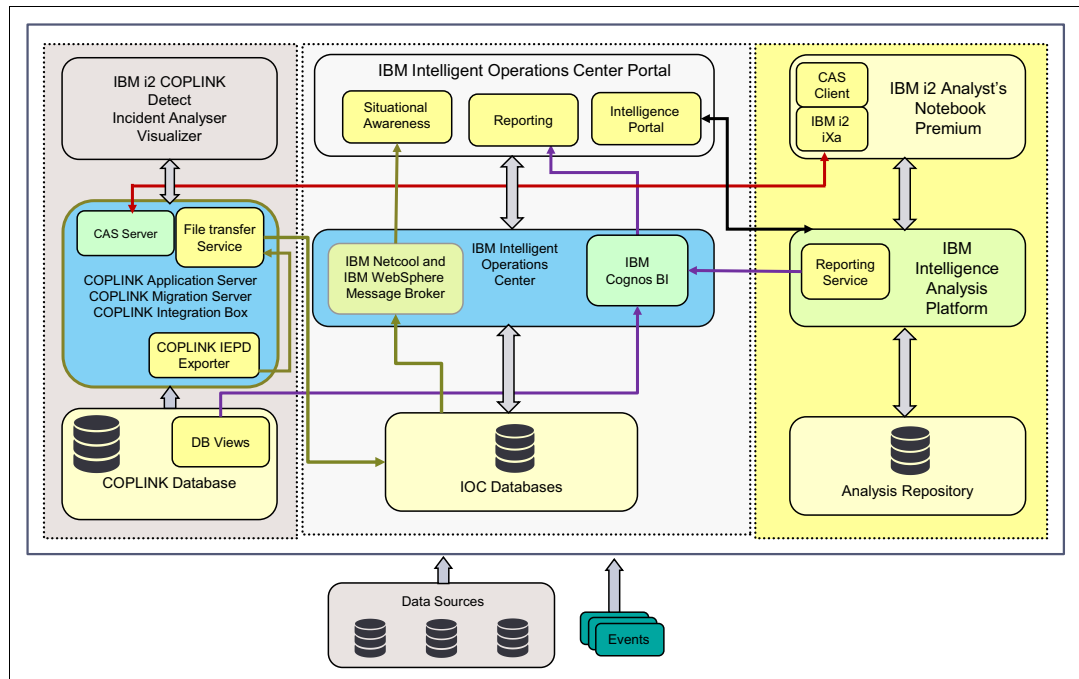
Figure 1-8   Mapping of components, products, and modules to capabilities

The following sections describe the functional architectural aspect of the extended capabilities.

Analysis Search

Analysis Search is one of the most sought after capabilities of any analysis tool. The power of intelligence analysis begins with the ability to bring in the data of interest and then enable you to gather insights and inferences from this data. The ability to conduct an advanced analysis search further empowers you to make these inferences.

The i2 COPLINK database provides the analyst with more than just a wealth of information. Because the data is consolidated from multiple data sources, and includes criminal data that is rich with details and linked information, it is highly useful for intelligence analysis.

From a functional standpoint, the architecture that makes the access to all of this data possible is shown in Figure 1-9. The i2 COPLINK Analysis Search must be installed as a server and the i2 COPLINK Analysis Search client plug-in must be installed on i2 Analyst's Notebook Premium.

> **Note:** The i2 COPLINK Analysis Search client uses IBM i2 Information Exchange for Analysis Search (iXa) technology (which allows for federated searching from various data sources from the i2 Analyst's Notebook).

i2 COPLINK Analysis Search client is a plug-in into i2 Analyst's Notebook Premium. Therefore, any user interface that is implemented by the plug-in (for example, an interface that lets the user enter the search term) can only appear within the context of the i2 Analyst's Notebook Premium user interface. The user interfaces are all integrated but you will know whether you are using the Analysis Search capability.

> **Note:** Search results can be saved in the Analysis Repository.
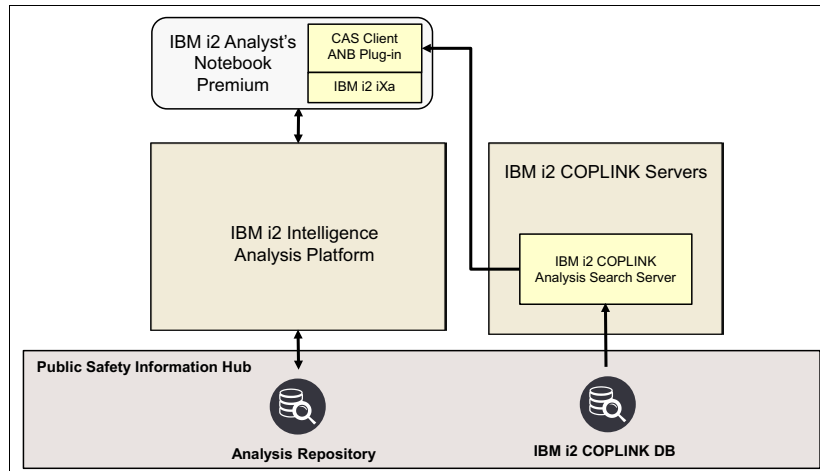
Figure 1-9   Searching the i2 COPLINK DB for analysis

Some configurations are required on both sides of the architecture. When the i2 COPLINK Analysis Search server is up, the i2 COPLINK administrator needs to set up users who can log in to the i2 COPLINK Analysis Search server. i2 Analyst's Notebook users are authenticated using the i2 COPLINK Analysis Search client interface. This client interface is an add-on to i2 Analyst's Notebook. The analyst can then search for information, such as people, a vehicle, and an organization, that is stored in the i2 COPLINK database using that add-on client interface. *Only unprotected data can be retrieved from the i2 COPLINK database through this tool.*

> **Note:** *Unprotected data* is data that can be shared with anyone within the Law Enforcement Agency. *Protected data* is more sensitive data that only selected people are able to see. Even authorized people will not be able to search the data that they are authorized to when using i2 COPLINK Analysis Search. To access the data to which they are authorized, they have to use other modules of COPLINK, such as i2 COPLINK Detect.

After receiving results that match the search criteria, the analyst can start manipulating the data locally in the i2 Analyst's Notebook Premium, using its powerful features and analytical algorithms, create charts, and optionally save them into the analysis repository.

Intelligence Portal

The Intelligence Portal is a web-based interface that enables users to access and manipulate intelligence data in the i2 Intelligence Analysis Platform analysis repository. Users can populate the repository, add items, delete items, and so on. Users can perform all create, retrieve, update, and delete (CRUD) operations, as long as they have the correct permission.

The Intelligence Portal comes with the i2 Intelligence Analysis Platform product, and it can be accessed using any browser that has Microsoft Silverlight installed. Figure 1-10 shows that the Intelligence Portal is also available from a special Intelligent Operations Center dashboard called the Integrated Law Enforcement (ILE) console. The ILE console is a portal page that is developed specifically for i2 Intelligent Law Enforcement V1.0.1 to host its extended capabilities.
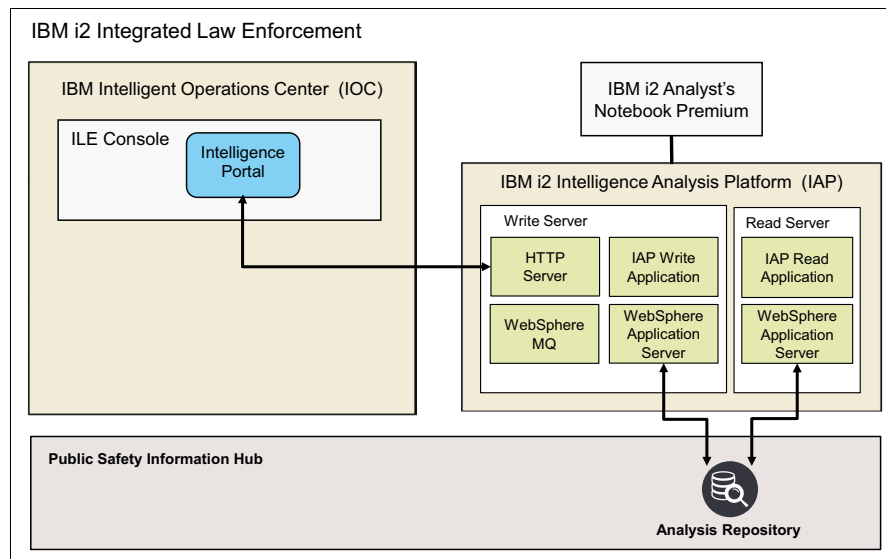
Figure 1-10   Intelligence Portal in the ILE console

The Intelligence Portal is implemented as a separate widget in the ILE console wrapped using iFrame.

> **Note:** When deploying i2 Intelligent Law Enforcement, you need to configure this portlet so that it points to the correct HTTP server that is the front end for your i2 Intelligence Analysis Platform installation.

Reports

Reports are essential to a law enforcement agency. The i2 Integrated Law Enforcement system provides two types of extended reporting capabilities, which are described in the following sections.

i2 Intelligence Analysis Platform reporting

Reporting in the i2 Intelligence Analysis Platform is now included with the release of i2 Intelligent Law Enforcement V1.0.1. The difficulty with producing a report using a tool, such as IBM Cognos Business Intelligence, is that the latter is limited to processing data using a relational model from a relational database. Data items created by i2 Intelligence Analysis Platform are not stored as pure relational data in the Analysis Repository (AR). The detailed information about an item is stored in the AR as XML strings.

To overcome this challenge, a mechanism is devised to convert the XML strings as rows in a relational database. By having this conversion and making the database available to IBM Cognos Business Intelligence, reports can be generated. The mechanism for conversion is shown in Figure 1-11.
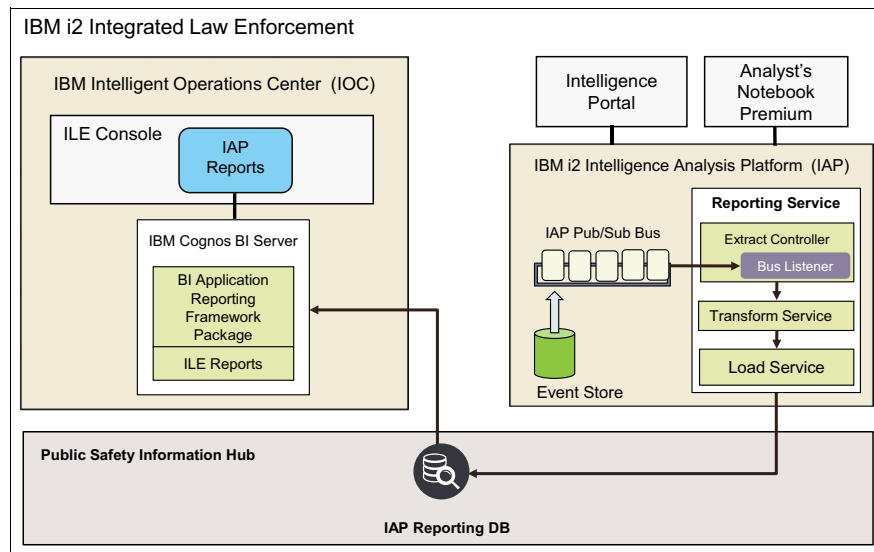
Figure 1-11   i2 Intelligence Analysis Platform report processing

The approach to reporting is to create a separate relational database in IBM DB2 so that every data item that is stored in the Analysis Repository has a corresponding relational representation in the new DB2 database. This is done by writing a new service in i2 Intelligence Analysis Platform called the *Reporting Service*, as shown in Figure 1-11. Recall that the architecture of the i2 Intelligence Analysis Platform can be extended by writing new services in Java. This new service is installed in the Read server.

All operations on an intelligence item (that is, create, update, and delete) are represented as events on the i2 Intelligence Analysis Platform Write server and stored in an Event store. The Write and Read servers talk to each other asynchronously using a queue. The Write server publishes these events to the queue, while the Read server subscribes to these events. As the Read server reads the event, one event at a time, it passes the event to the Reporting Service. The Reporting Service processes events related to the creation, update, and deletion of items. It performs the appropriate action, that is, it extracts the item from the XML string stored in the event, performs the correct transformation to the relational database, and then saves the results to the DB2 database. If the operation is an update or delete, the reporting service must perform the corresponding actions for the update or delete. The Analysis Repository and this new DB2 database must accurately store exactly the same information, but in different formats.

When a report for the i2 Intelligence Analysis Platform is requested by a user from the ILE console, a request is sent to IBM Cognos BI, which then retrieves the information from the DB2 database using the corresponding SQL queries and applies the correct report template based on the user's request. This implies that the IBM Cognos BI server in the IBM Intelligent Operations Center must be configured so that the i2 Intelligence Analysis Platform DB2 database is another *data source.* This configuration requires that the correct Java Database Connectivity (JDBC) driver is configured to be used by the Cognos BI server. In addition, the report templates must be registered to the server.

> **Note:** i2 Intelligent Law Enforcement V1.0.1 comes with a set of predefined report templates for i2 Intelligence Analysis Platform reports.

You can create your own report templates using IBM Cognos Framework Manager. This software does not ship with either IBM Intelligent Operations Center V1.5 or IBM i2 Intelligent Law Enforcement V1.0.1.

> **Note:** Any user who requests a specific i2 Intelligence Analysis Platform report must have the required access privileges to the data being retrieved from the database.

i2 COPLINK reporting

The next report is for i2 COPLINK. Unlike the i2 Intelligence Analysis Platform intelligence data item, i2 COPLINK data is already in relational model format using either Oracle or Microsoft SQL server. However, creating a report for i2 COPLINK also has a limitation. Not all of the needed data can be retrieved easily from the i2 COPLINK database. That is, no SQL query can be formulated without first making the needed data accessible. To solve this problem, database views must be created first to make the data accessible. In doing so, SQL queries can be created for all of the i2 COPLINK predefined report templates, as shown in Figure 1-12.
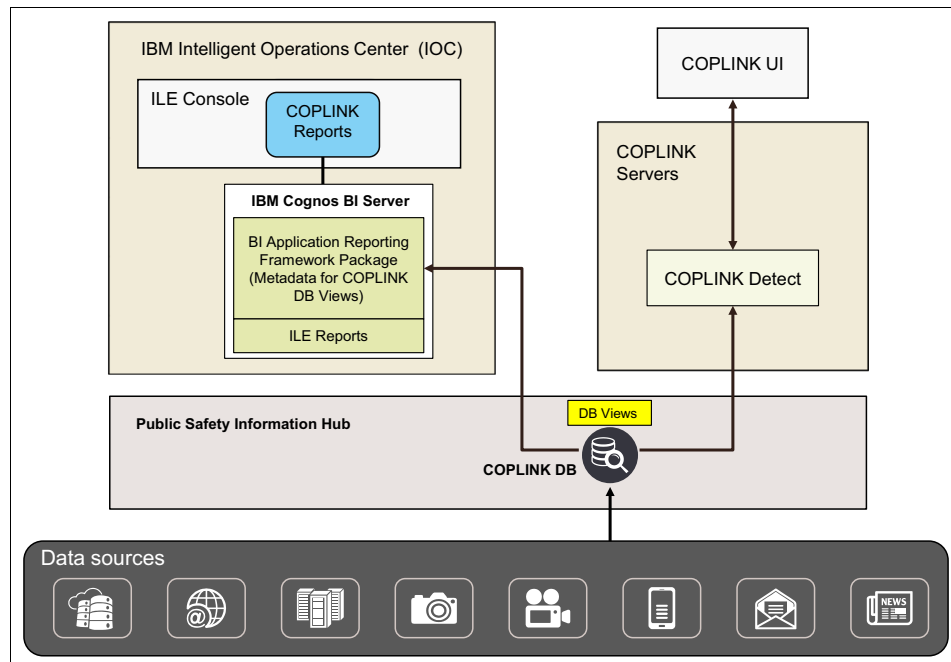
Figure 1-12   i2 COPLINK report processing

Note: In Figure 1-12, the only way that the i2 COPLINK database is populated with data from various sources is through data ingestion, typically through an extract, transform, and load (ETL) tool. After that, i2 COPLINK data is read-only. For example, i2 COPLINK data can be retrieved using the i2 COPLINK Detect module. Users can view and manipulate the retrieved data, but they cannot change the underlying data store.

When an i2 COPLINK report is requested, the correct report template is used, and SQL queries are executed against the database views. ILE also enforces security by ensuring that the requester has the necessary access privileges.

Situational Awareness

Situational Awareness is a major extended capability because it provides not only information about crimes that have occurred in certain locations, but it also allows you to retrieve detailed information about each crime incident. Online services that provide similar capabilities cannot provide specific details because the information is classified. With i2 Integrated Law Enforcement, however, this sensitive information can be retrieved and rendered, because the agency owns the database where this information is stored.

Note: It is still necessary that users have the legal authority to view this sensitive information within the law enforcement agency. For example, they must be at least Criminal and Justice Information Services (CJIS) certified.

Figure 1-13 shows the processing that takes place to implement Situational Awareness. The key to this implementation is the Information Exchange Package Description (IEPD) exporter component from i2 COPLINKS. The exporter is configured to read crime data from the i2 COPLINK database on a regular interval and then export the data in IEPD format to the file system (see the IEPD logs in Figure 1-13).
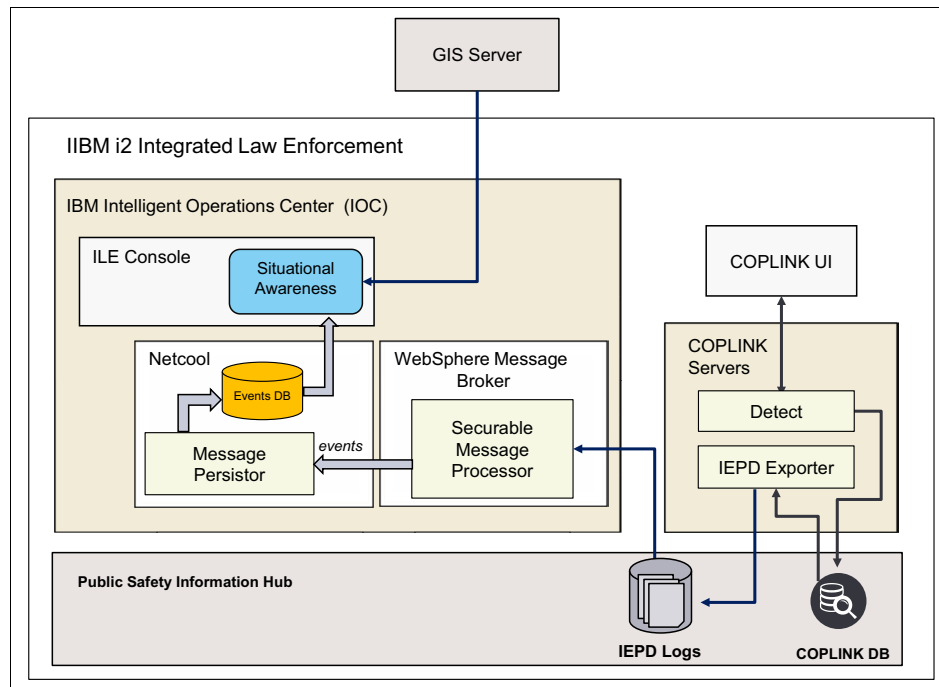
Figure 1-13   Situational Awareness processing

Meanwhile, newly exported IEPD logs need to be transferred to the IBM Intelligent Operations Center for processing. The method for moving these log files is a File Transfer service, such as FTP. A background process in IBM Intelligent Operations Center logs in to the file transfer service to transfer any new IEPD logs.

After the IEPD logs are moved to i2 Intelligent Operations Center, the next step is to scrape the log files for each crime entry, convert them as events and send them to the IBM WebSphere Message Broker, which, in turn, processes those events by reading the entries. The events are then processed by the Event Management module of the IBM Intelligent Operations Center. The event management module is implemented by IBM Tivoli Netcool/Impact. Processed data is stored in a table, along with all the pertinent data about the event.

Meanwhile, the Situational Awareness portlet in the ILE console retrieves crime data from IBM Tivoli Netcool/Impact whenever it has to refresh the map widget.

Note: Situational Awareness uses the map widget from the IBM Intelligent Operations Center to display a visual representation of the areas around the law enforcement agency. The actual maps and layers are retrieved from an external GIS server, such as an ESRI server.

The Situational Awareness portlet applies the filters that the user has specified when configuring the Situational Awareness portlet. Based on what it retrieves from Netcool/Impact, the map is refreshed with the new crime data.

Integrated Law Enforcement deployment

This chapter is directed to teams that are responsible for deploying the IBM i2 Integrated Law Enforcement solution. Deploying the solution is achieved by deploying i2 Intelligent Law Enforcement V1.0.1, which is the latest product release that implements the IBM i2 Integrated Law Enforcement solution.

This chapter provides the steps for getting all of the product components up and running, which prepares you to delve into the more tedious tasks of integrating and configuring various components. The integration and configuration are described in Chapter 3, "Integrated Law Enforcement cross-component stitching" on page 45.

2.1 Overview

This chapter describes the steps for deploying i2 Intelligent Law Enforcement V1.0.1. From this information, you can choose your own approach or methodology. However, there are five milestones that are required, and in a specific order:

1. Prepare for deployment: This step is important and must be taken seriously so that no major roadblock can lead to reconstructing what you have already started.

2. Install the component products: This step involves laying down the foundation of the solution on which all other activities depend. Chapter 1, "Integrated Law Enforcement system overview" on page 1 devotes several sections to this topic. It can take a significant portion of the entire deployment process.

3. Stitch the components together: After the foundation is in place, you perform several steps to install applications and configure resources to get all the extended capabilities working.

4. Customize the deployment for your environment: Customization involves tasks to ensure that i2 Integrated Law Enforcement addresses the needs of the target users, based on their requirements.

5. Test and validate the deployment: It is important to note that, at a minimum, testing must be carried out. Here, the type of testing and validation involved is system-wide.

After these milestones are completed, a training session needs to follow to prepare your users and to gain mastery of using i2 Integrated Law Enforcement. Before moving the system into production, pilot testing is highly advised to ensure that all requirements are met and the system is working as designed.

2.2 Prepare for deployment

The importance of good preparation in any undertaking cannot be underestimated and needs to be re-emphasized if the undertaking is to be successful. Good planning and preparation are as critical as the execution itself, if not more.

Deploying the IBM i2 Intelligent Law Enforcement V1.0.1 product is complex and, like any other complex project, requires careful planning, well planned steps and procedures, and good team communication and coordination. Although this chapter does not cover project management, system analysis, and methodologies of product implementation, it does highlight the areas where higher priorities need to be placed when delivering an integrated law enforcement solution.

2.2.1 Know your team

Deploying i2 Integrated Law Enforcement from the initial steps to going live takes at least six months to one year, depending on the resources and skill sets you have available and on the scope of the project. In many cases, *customization* and *quality assurance validation* alone take one-half of the time.

Knowing the members of your team and getting to know their past projects and the skill set they bring to the team are important. Similarly, getting to know their working style and preferred method of communication is also significant.

Key items to remember include defining the role of each team member, aligning your expectations with theirs, keeping each other informed at all times, communicating issues, concerns, and ideas with them, and being open to compromise when necessary. Know the rules of engagement. Be a team player.

> **Note:** An important item to consider is that this deployment will involve the installation of IBM i2 COPLINK, which is currently performed exclusively by IBM i2 Lab Services or its IBM Business Partners. Therefore, IBM i2 Lab Services will be part of your extended team for completing this project.
>
> Identify your initial point of contact at this website:
>
> `http://www.ibm.com/software/products/en/coplink`

2.2.2 Know your products

The next important consideration is to become familiar and proficient with the products that you will deploy. To implement the IBM i2 Integrated Law Enforcement solution, the main product is IBM i2 Intelligent Law Enforcement V1.0.1. This product is available in two packages: Premium and Standard.

The Premium and Standard packages

Figure 2-1 shows the products that are included in both the Premium and Standard packages, and any restrictions on their use.

Figure 2-1   i2 Comparison of the i2 Intelligent Law Enforcement Premium and Standard packages

i2 Intelligent Law Enforcement V1.0.1 Premium

i2 Intelligent Law Enforcement V1.0.1 Premium is intended for users who work in a crime or intelligence information production role. These users typically use the applications all day, every day, and depend on the advanced capabilities of the applications to fulfill their role. They require more advanced capabilities or specialist analysis tools. This package contains full Integrated Law Enforcement functionality, including access to full administration capabilities.

i2 Intelligent Law Enforcement V1.0.1 Standard

The i2 Intelligent Law Enforcement V1.0.1 Standard package is intended for those who will use the system less intensively, such as patrol officers, operational staff, and other support staff. Typically, these users will use the solution in support of their day-to-day work and only when required.

Comparing the two packages shown in Figure 2-1 on page 23, the Standard package imposes restrictions on the three major products. These restrictions refer to the administrative functionalities of these three products. Furthermore, the Standard package does not include the i2 Analyst's Notebook and two other products that are needed for searching the i2 COPLINK database from i2 Analyst's Notebook. There are advanced analysis capabilities included in the Premium package that are not available in the Standard package.

A typical deployment of i2 Intelligent Law Enforcement V1.0.1 requires administrative functions, and therefore, offer the Premium package to your client first. Additional licenses for the Standard package are advised if users exist in the organization that fit the description for Standard package users as stated before.

Both of the i2 Intelligent Law Enforcement V1.0.1 packages provide national language versions for Spanish, Brazilian Portuguese, Simplified Chinese, and Traditional Chinese.

Ordering Information

i2 Intelligent Law Enforcement V1.0.1 is only sold directly by IBM or by authorized IBM Business Partners for Software Value Plus. See the IBM Software Value Plus (SVP) website:

`https://www-304.ibm.com/partnerworld/wps/servlet/ContentHandler/software_value_plus`

i2 Intelligent Law Enforcement V1.0.1 is not available for purchase online. To locate IBM Business Partners for Software Value Plus in your geography, visit this website:

`http://www.ibm.com/partnerworld/wps/bplocator/`

After you purchase the software, download the IBM i2 Intelligent Law Enforcement product from the IBM Passport Advantage® Online website because it is not available as a shrink-wrapped product:

`http://www.ibm.com/software/passportadvantage`

If you require assistance using Passport Advantage, find the IBM customer care contacts for your region at this website:

`https://www.ibm.com/software/howtobuy/passportadvantage/paocustomer/docs/en_US/eca re.html`

What to download

The main software package that you need to download from Passport Advantage is *IBM i2 Intelligent Law Enforcement V1.0.1 Multilingual Media Pack* with product ID BF079MLh. This package already includes all of the core components shown in "The Premium and Standard packages" on page 23.

You will need other software when you start integrating, configuring, and stitching the components together, as described in Chapter 3, "Integrated Law Enforcement cross-component stitching" on page 45.

Because IBM Intelligent Operations Center V1.5 is only available for Linux, you might need to log in to the Linux servers remotely if you do not have direct access to a Linux terminal. This scenario is likely if your deployment environment uses virtualized systems. This might not be a problem if your environment is Linux or certain versions of UNIX.

However, if your environment is Microsoft Windows, you need to run an X Window System (X-Windows) client that runs on your Windows operating system. IBM Intelligent Operations Center V1.5 uses IBM Installation Manager, which renders a GUI.

> **Note:** Many of the prerequisites of IBM i2 Intelligence Analysis Platform on Linux also use IBM Installation Manager.

You can either buy the software that provides X Window System client capabilities, or download no-charge open source tools, such as Cygwin (`https://www.cygwin.com/`) or Cygwin/X (`http://x.cygwin.com/`). You might also find it convenient to have a terminal client that runs on your MS Windows desktop and connects to your Linux machines. A no-charge open source program called PuTTY (`http://www.putty.org`) is a helpful tool. To transfer files between your MS Windows machine and the Linux machines, WinSCP (`http://winscp.net/eng/index.php`) is also a useful tool.

2.2.3 Conduct a requirements workshop

Whether you are deploying the solution for your own organization or you are working as a consultant for an external client, a requirement gathering workshop is an important step in preparing for this endeavor for the following reasons:

► Not all clients (and requirements) are the same
► Not all needs can be served by deploying the product with its default values and initial settings

For example, different clients have different deployment environments. Some clients might want a traditional, distributed system, and others might want a complete, one-product, virtualized system. These differences have an impact on how you approach the deployment process, scheduling of activities, access mechanisms to the deployment environments, and so on. Certain clients will require more capabilities and features, which, in turn, require additional skills and resources. Clients might have unusual use case scenarios, while others might not be sure yet of what they want or need.

Amid these variations and nuances, it is imperative that you get as many details as possible to correctly define the scope of the project. You need to distinguish what is important and what is relevant. The goal of the requirement workshops is to gather all the information that you need to successfully complete the deployment of the solution.

The requirement workshops with the client must cover, at a minimum, the following areas:

► Project scope and target completion:
  – Can the project be phased?
  – What is the correct priority order?
► Target users:
  – How large is the user community?
  – What roles do they play?
  – What skill levels do they have?
  – How can the system make them more efficient and productive?
  – What training is needed?
  – Who will be the focal points for the product?
  – Who will maintain the system?

- ► Challenges:
  - – What are biggest issues and most significant pain points in the current system?
  - – What are the potential roadblocks to this project?
- ► Expectations:
  - – What does the client expect when the new system is deployed?
  - – Are these expectations realistic?
  - – What is the measurement of success?
- ► IT infrastructure:
  - – What systems does the client currently have?
  - – What changes is the client anticipating and when?
  - – How available or accessible are the clients to answer questions and provide information about the infrastructure?
  - – Do you currently have what is required to deploy the system?
  - – Do they have their own geographic information system (GIS) servers?
  - – Do they have a centralized user registry?
  - – What technology do they use?
- ► Security policies and requirements:
  - – How does the client control access?
  - – What data needs to be protected?
  - – What policies do you need to be aware of?
- ► Data sources:
  - – What are the client's data sources?
  - – How is the client currently collecting their data sources?
  - – Who is maintaining the data source?
  - – How does the client anticipate the new system will consolidate these data sources?
- ► Existing data:
  - – What existing data does the client have?
  - – Which of this data does the client want to migrate to the new system?
  - – What is the nature of this data (types, formats, security classification, and so on).
  - – How much existing data is there?
- ► Performance and high availability:
  - – How close to real time does the client want the system to be?
  - – How highly available must the system be?
  - – What are the fallback procedures?
- ► Integration:
  - – What systems does the client currently have with which the new system needs to integrate?
  - – Do they require working with third-party vendors?
  - – What level of single sign-on (SSO) do they require?
  - – What technology is in use for SSO?

- ► Migration:

  What is the plan for migrating to the new system?

- ► Customization:
  - – What types of customization does the client require?
  - – Do they have a certain *look and feel* that needs to be maintained across the board?
  - – What schema changes will be necessary for the new system?

- ► Capabilities:
  - – What capabilities does the client want in addition to what the new system delivers, after the typical configuration is complete?

  - – What other capabilities are included in their roadmap?

- ► Vision:

  What is the overall vision of this organization?

There are many other areas to consider, for example, scheduling technical meetings with the correct people. It is helpful to get their vision of where they want to be in the future because this information will help in your deployment decisions and how to accommodate that vision.

As you hold more workshops and iterate over several requirements, the workshops can also be used to validate what you already know. Keep the feedback loop flowing continuously. You might find yourself breaking down large tasks into smaller subsets of tasks, rectifying misconceptions, digging in for more details, realigning expectations, and more.

These workshops are also opportunities for socializing with your client to get to know them better and to establish good rapport and trust. The reality is that technical teams spend more time with the client than the business team.

There will always be confusion and miscommunication along the way. The goal is to catch them as early as you can by applying good documentation practices and regular checkpoints.

You know that a workshop is successful when it accomplishes these goals:

- ► Defines the project scope more crisply.
- ► Brings more clarity to the tasks that need to be accomplished.
- ► Eliminates noise and unnecessary information.
- ► Identifies potential issues and challenges.
- ► Makes sizing easier.
- ► Results in a list of action items.
- ► Enables you to think in terms of a roadmap.
- ► Clarifies expectations from both sides.
- ► Makes you feel that you learned something new.
- ► Feels fun working with your team and the client.

Whatever your criteria for success are, a workshop must always end on a positive note and with a sense of accomplishment. The deliverable of this exercise is a summary of the team's understanding to ensure that key points were taken and that your client agrees with the summary.

In many cases, you will need to follow up with smaller focus groups to discuss more specific requirements, such as the schema to be used for the i2 Intelligence Analysis Platform, any customization required for user interfaces, data, and event formats, and security restrictions, just to name a few.

2.2.4 Perform capacity planning

Another important activity for deployment preparation is capacity planning. This step is especially important if the target organization is medium to large in size, and requirements for high performance, high availability, fault tolerance, and resiliency are high.

Table 2-1 shows a simple matrix that can be used as guideline to gauge the size of the target organization. Treat these guidelines as estimates only. Adjust the values in Table 2-1, depending on additional information that you believe will affect the basis for the size of the organization. As the size of the organization increases, so does the required capacity.

Table 2-1   Simple guideline for assessing organization size

	Pilot	**Entry level to medium-sized organization**	**Large organization**	**Very large organization**
Usage type	Proof of concept, test	Production	Production	Production
Number of users	2 – 4	50	200	500+
Data volume	> 20 GB	500 GB – 2 TB	2 TB – 4 TB	5 TB+

Inputs to this activity might come from discussions with the IT department of the client and from the requirements workshops. Either a member of your team or an external expert needs to be assigned to this task, taking into consideration the following aspects:

► System requirements of the different products

► Expected number of concurrent users for i2 COPLINK, IBM Intelligent Operations Center, and i2 Intelligence Analysis Platform, and their roles

► Expected size of the data that will be ingested into the i2 COPLINK database

► Expected growth rate of the data that will be stored in the analysis repository of i2 Intelligence Analysis Platform

► Expected growth rate of data that will affect data storage in IBM Intelligent Operations Center, such as reporting or total number of crimes

► Types of data to be stored, such as video, audio, images, and documents

► Performance and high availability requirements that determine the need for a redundant system

► The need for test and development environments, a staging environment, and so on

► Backup procedures and practices

► Disaster and recovery requirements

► Performance and security requirements

► Network latency of existing network and the technology used

► Effect of integrating with other systems

► Storage infrastructures, such as storage area network (SAN), solid-state drives (SSDs), and disk arrays

Table 2-2 on page 29 summarizes the minimum requirements for IBM Intelligent Operations Center V1.5.

Table 2-2   IBM Intelligent Operations Center V1.5 minimum system requirements

Resources	Application server	Event server	Data server	Management server	Installation server
Number of CPUs	4	4	4	4	2
Memory	24 GB	16 GB	16 GB	24 GB	4 GB
Network adapter	1	1	1	1	1
Disk space	113 GB	108 GB	108 GB	108 GB	108 GB
Additional disk space required during installation	90 GB	90 GB	90 GB	90 GB	90 GB

The following minimum requirements are for i2 Intelligence Analysis Platform and i2 Analyst's Notebook Premium:

► Two servers:

- 2 GHz 4-core processor (minimum requirement)
- 8 GB RAM
- 2 TB Redundant Array of Independent Disks (RAID) disk arrays per server

► Client machine for each i2 Analyst's Notebook Premium installation:

- 2 GHz single core processor
- 512 MB RAM
- SXGA-capable graphics card (1024 x 768 high color (16 bit)) color monitor
- Mouse or trackball

For the database servers, consult the minimum requirements for each of the supported database products in the following documents:

► IBM DB2:

http://www.ibm.com/support/docview.wss?uid=swg27038033

► MS SQL Server:

http://msdn.microsoft.com/en-us/library/bb545450.aspx

► Oracle Database:

http://www.oracle.com/us/products/database/enterprise-edition/overview/index.html

Table 2-3 summarizes the minimum requirements for the five i2 COPLINK servers.

Table 2-3   i2 COPLINK V4.8 minimum system requirements

Resources	Web server	Migration server	Database server	Exporter server	Integration box
CPU	2	2	2	2	2
Clock speed	2.66 GHz	2.66 GHz	2.66 GHz	2.66 GHz	3 GHz
Memory	8 GB	8 GB	8 GB	8 GB	4 GB
Disk space	2 146 GB 15K	2 146 GB 15K	2 146 GB 15K	2 146 GB 15K	2 160 GB
HDD configuration	RAID-1	RAID-1	RAID-1	RAID-1	RAID-1

In addition, two separate data storage units are required, each with a 30 300 GB hard disk drive hard disk drive (HDD) that supports RAID 50 and 15K dual host bus adapter (HBA) controllers. i2 Intelligent Law Enforcement V1.0.1 is licensed by the number of concurrent users.

2.2.5 Design solution architecture

Because i2 Intelligent Law Enforcement V1.0.1 consists of a number of IBM products, several options exist regarding the operating system, database product, and Lightweight Directory Access Protocol (LDAP) product.

> **Important:** The decisions that are made about these products influence the route to take during deployment.

Table 2-4 outlines the possible combinations of operating system, database, and LDAP for each of the component products.

Table 2-4 Supported operating system, database, and LDAP options

	IBM Intelligent Operations Center	i2 Intelligence Analysis Platform	i2 COPLINK
Operating system			
Linux	x	x	
Windows Server		x	x
Windows Client			
Database			
IBM DB2	x	x	
Oracle		x	x
MS SQL		x	x
LDAP			
IBM Tivoli Directory Server	x	x	
IBM Domino® Directory		x	x
MS Active Directory		x	x
Novell eDirectory			x

IBM Intelligent Operations Center V1.5 is only supported on Red Hat Enterprise Linux (RHEL) Version 5, Update 5 or later. It is not supported on RHEL Version 6.

The i2 COPLINK components, which include the i2 COPLINK server, the i2 COPLINK Exporter, the i2 COPLINK migration server, i2 COPLINK Analysis Search (CAS), and i2 Analyst's Notebook (ANB), are only available on Windows 7 and Windows Server 2008 and later.

i2 Intelligence Analysis Platform can be installed on either Windows or RHEL Version 5, Update 9 or later.

IBM Intelligent Operations Center V1.5 only supports IBM DB2.

IBM i2 COPLINK and its component products support Oracle and MS SQL.

The i2 Intelligence Analysis Platform V3.0.3 supports IBM DB2, Oracle, and MS SQL.

IBM Intelligent Operations Center V1.5 supports the IBM Tivoli Directory Server only.

Security for the i2 COPLINK components can be configured to use the internal i2 COPLINK database as the user repository, or it can be configured to use IBM Domino Directory, Microsoft Active Directory, or Novell eDirectory.

i2 Intelligence Analysis Platform V3.0.3 can be configured to use any directory that is supported by IBM WebSphere Application Server Version 8. Chapter 4, "Integrated Law Enforcement security" on page 107 describes the IBM guidelines for common user registry.

Having this information and the information gathered during the requirements workshops, the solution architecture can be proposed. The solution architect is responsible for addressing the requirements and taking into consideration all of the limitations and constraints. The solution architecture is then presented to the client for approval.

All of these factors must be studied thoroughly by the solution architect in the team. The solution architecture for i2 Integrated Law Enforcement must include suggestions about system resources and specifications, such as physical servers, disks, workstations, tape drives, tapes and other secondary devices, networking, and other supporting software.

Some clients might already have the infrastructure to support these requirements. However, there is a difference between the current environment and what aspects of the current environment can be used for the i2 Integrated Law Enforcement solution. Your client will be interested in knowing what else might need to be procured.

Your team is not wholly responsible for this activity. Reach out to other teams or experts who can help with producing the solution architecture. At the culmination of this activity, a target deployment environment or platform must be agreed upon between your team and the client.

2.2.6 Prepare a work proposal and a project plan

By now, your team has a better understanding of the scope of the project and the activities that need to be carried out during deployment. It is important that you produce a work proposal for your client that details every activity your team will be committed to. This work proposal is sometimes called a Statement of Work (SoW), which is essentially a contract between your team and the client, expressing a mutual understanding of the scope of the work.

You and your client can choose which components of IBM i2 Intelligent Law Enforcement V1.0.1 to install based on the requirements of the solution. Of course, if the component is not installed, the capabilities provided by the component will not be available. All components required for basic and extended capabilities defined in the solution architecture must be installed. The team responsible for deployment can simply skip the steps that apply to components that do not need to be deployed because their capabilities are not required by the solution.

Appendix A, "Snippet of a sample statement of work proposal" on page 179 provides a snippet of a sample SoW proposal.

2.2.7 Prepare the deployment platform

The deployment platform is identified as part of the capacity planning activity described in 2.2.4, "Perform capacity planning" on page 28. The deployment platform can be a set of physical servers or virtual machines. It can also use an IBM BladeCenter or an IBM PureFlex® System. The storage can be over a SAN or over high-powered solid-state RAID disks. Many options and possibilities exist.

The deployment platform might already be available, or it might need to be purchased. As soon as the deployment platform becomes available, start to prepare it for its readiness. A hardware specialist or virtualization administrator must prepare these target machines. If the deployment platform is purchased, there will be additional work to set up the power supply, configure the hardware, connect it to the network, apply the necessary security settings, allocate disk space, and so on.

Additional tasks include assigning host names and IP addresses to the servers and creating user accounts that will be used during deployment. If using virtual machines, you will need to access those virtual machines from a client machine. It is important to remember that this activity is a collaboration between the deployment team and the client organization.

When all of these steps are completed, you will have all of the needed information and tools to access the machines in the chosen deployment platform.

2.3 Install the products

You have made all the preparations by discussing the requirements with your client and by deciding on the deployment platform. You have also worked with IBM i2 Lab Services on the i2 COPLINK deployment. The capacity planning has been completed, too. The work proposal or statement of work with your project plan outlines all of the tasks that must be completed at the end of this project.

The good news is that the installation of each of the component products is independent from each other. Depending on the size of your team and the skill level of the members, these installations can be done in parallel. There might be factors that can inhibit concurrency, such as access to the servers and other constraints. Make every effort to optimize the execution of these tasks.

Because the three product components can be installed in parallel, include the following considerations in your planning:

► IBM Intelligent Operations Center installation takes the longest time; therefore, more careful planning is required.

► The i2 Intelligence Analysis Platform installation requires more manual tasks. To perform these tasks, you need to be thorough, careful, and attentive.

► IBM i2 Lab Services, for the most part, performs the i2 COPLINK installation.

► The more tedious tasks occur during the configuration and customization steps. The most time-consuming part of the entire procedure is the creation of the COPLINK database, which is part of the customization process.

Figure 2-2 on page 33 provides an outline of the production installation steps.

> **Important:** The installation is not considered complete before the verification and confirmation that each installed unit is working properly.
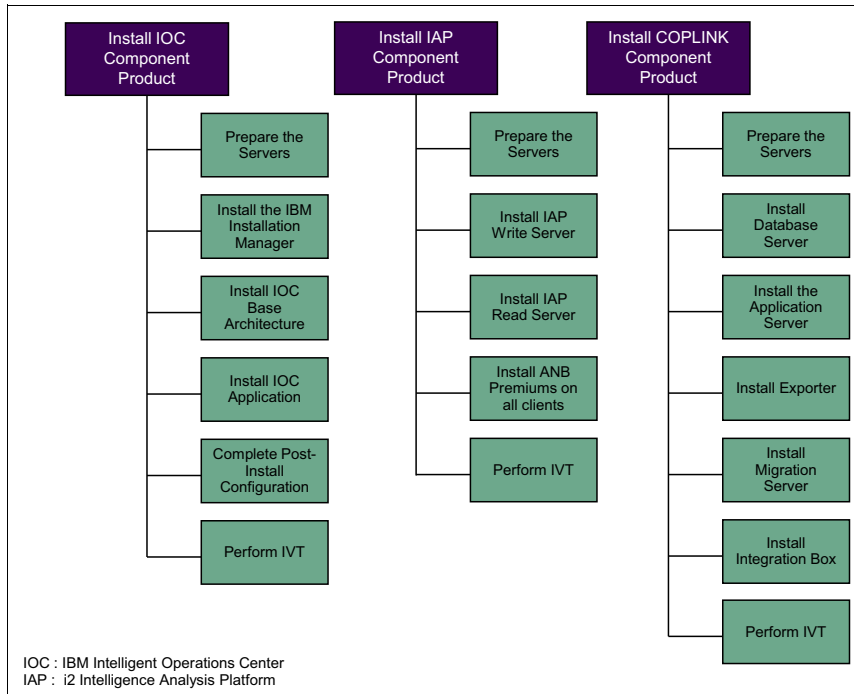
Figure 2-2   Overview of installation steps

The next sections provide high-level installation processes of the products. For more information, see the deployment documentation for each product.

2.3.1 Installing IBM Intelligent Operations Center

Intelligent Operations Center V1.5 is a complex product that requires an extended period of time to install. For this reason, the installation process has been split into multiple phases to allow the user to discover and resolve issues early in the process.

The installation can be performed either manually or with the GUI provided by IBM Installation Manager. The latter approach is described in this section. Full details of both installation methods are provided in the IBM Intelligent Operations Center V1.5 Information Center:

http://pic.dhe.ibm.com/infocenter/cities/v1r5m0/topic/com.ibm.ioc.doc/ic-homepage.html

Intelligent Operations Center V1.5 requires five 64-bit x86 servers running RHEL Version 5, Update 5 or later. All servers also require a minimum of 500 MB storage for the boot partition, a minimum of 4 GB storage for the swap partition, and an additional 90 GB storage for use during the Intelligent Operations Center V1.5 installation process.

Prepare the servers and begin the installation

For a successful install, the servers need to be prepared first.

Note: Experience with Linux or UNIX is required to prepare the servers.

The following high-level list shows the preparation tasks for the servers:

► TCP/IP networking must be configured on all servers.

► Each server must have a fully qualified domain name and short name defined using a Domain Name System (DNS) server, or the `/etc/hosts` file and local loopback addressing must be enabled in the `/etc/hosts` file. The `nofile` parameter in `/etc/security/limits.conf` must be set to 20480, and the `net.ipv4.tcp_fin_timeout` parameter in the `/etc/sysctl.conf` file must be set to 30 on all servers, with the exception of the installation server.

► Security-Enhanced Linux (SELinux) and all Linux firewalls need to be disabled on all servers.

► `sshd` (Secure Shell Daemon) must be enabled for `root` login with password authentication on TCP/IP port 22.

► The servers must have the same time and date set. The use of a time synchronization service is preferable.

► The following Linux packages, available from Red Hat, must be installed on each server:
  – compat-lidstc++-33-3
  – libXp-1.0.0-8
  – libXmu-1
  – libXtst-1
  – pam-0
  – rpm-build-4
  – libaio-0
  – libstdc++-4
  – libXft-2
  – compat-db-4
  – elfutils-libs-0
  – elfutils-0
  – libgcc-4
  – compat-glibc-2
  – openmotif22-2
  – audit-libs-1
  – glibc-2
  – glibc-common-2

When the preparation of the servers is complete and the required configuration settings are implemented, begin the installation:

1. Restart the servers.

2. Copy the IBM Intelligent Operations Center V1.5 installation package, included in the IBM i2 Intelligent Law Enforcement V1.0.1 Multilingual Media Pack (BF079MLh) to an installation server and install the included Java 6 runtime environment `ibm-java-x86-jre-6.0.10.1x86_64.rpm`.

3. Set the `JAVA_HOME` environment variable by updating the `.bash_profile` file for the `root` user to add:

```
export JAVA_HOME=/opt/ibm/java-x86_64-60/jre
```

Install IBM Installation Manager

The IBM Intelligent Operations Center V1.5 installation wizard is included in the archive `BA_1.5_GUI_Installer_Lite_Launchpad.zip`. This file is part of the IBM Intelligent Operations Center V1.5 installation package. Follow these steps to install IBM Installation Manager:

1. Extract the contents of `BA_1.5_GUI_Installer_Lite_Launchpad.zip` to the directory where the IBM Intelligent Operations Center V1.5 installation package was copied.

2. Launch the `launchpad.sh` command from this directory.

3. From the panel that is displayed, click **Install IBM Installation Manager**.

4. Follow the instructions in the wizard to complete the installation of IBM Installation Manager.

Install IBM Intelligent Operations Center V1.5 base architecture

After installing IBM Installation Manager, perform the following steps:

1. Restart IBM Installation Manager to automatically detect the IBM Intelligent Operations Center V1.5 topology file. This step also starts the Intelligent Operations Center V1.5 Installation process.

   The installation process consists of six stages and many of them take a considerable amount of time to complete. Table 2-5 lists these stages and their corresponding estimated installation time.

   Table 2-5   Estimated installation time for IBM Intelligent Operations Center V1.5

Stage	Estimated installation time
Prepare install	2 hours +
Prepare environment	10 minutes
Install and configure platform	Phase 1: 12 hours Phase 2: 3 hours
Install Platform Control Tool	10 minutes
Install Platform System Verification Check Tool	15 minutes
Install IBM Intelligent Operations Center applications	3 hours

2. Refer to *Checklist - installing using IBM Installation Manager* at this website for a list of the steps that are required to complete each stage of the installation:

   http://www.ibm.com/support/knowledgecenter/SS3NGB_1.5.0/ioc/ba_install_im_steps.dita?lang=en

   Most installation issues that you might encounter are due to the incorrect configuration of the operating system and, in particular, TCP/IP networking. To avoid these issues, ensure that these settings are correct before proceeding with the installation. As with any product installation, it is advisable to search for known issues before you start.

   The following URL returns useful technotes, tips, and authorized program analysis reports (APARs) for IBM Intelligent Operations Center V1.5:

   http://www.ibm.com/support/search.wss?tc=SS3NGB&sort=desc&r=99

The following technotes are particularly interesting:

– Corrections and additions to installation information for IBM Intelligent Operations Center V1.5:

http://www.ibm.com/support/docview.wss?uid=swg21610989

– Installing Intelligent Operations Center V1.5 will fail if the installation directory is not /installMedia:

http://www.ibm.com/support/docview.wss?uid=swg21610872

– Error CIYBA0208E when installing on AMD 64 systems:

http://www.ibm.com/support/docview.wss?uid=swg21639871

– Preventing a Java out of memory exception during the Sametime installation:

http://www.ibm.com/support/docview.wss?uid=swg21639881

Figure 2-3 shows the servers for the IBM Intelligent Operations Center V1.5 installation. The virtual machines are spread across one to many physical servers. The capacity of the virtual machines depends on the capacity of the physical servers and the workload of the solution.
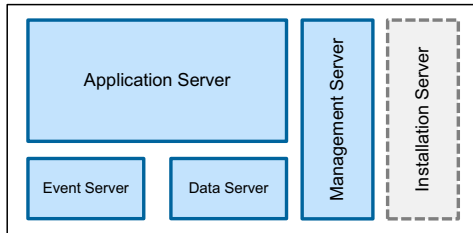
Figure 2-3   Required servers for IBM Intelligent Operations Center V1.5

The following sections provide a brief description of each of the servers shown in Figure 2-3.

Application server

The application server is responsible for the overall web infrastructure for running the IBM Intelligent Operations Center solution. It is also responsible for the user interface infrastructure, including the display of key performance indicators (KPIs) on dashboards and report generation. It is through the application server that users log in to the IBM Intelligent Operations Center portal and obtain access to the entire set of functions available in the solution. Figure 2-4 shows the product components that are installed in the application server.
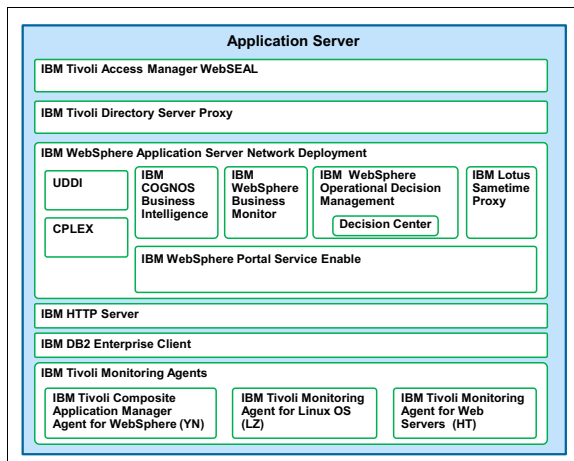
Figure 2-4   Product components installed in the application server

Data server

The data server provides the data services infrastructure and repository that is used by the other internal subsystems of IBM Intelligent Operations Center. It also holds the directory repository that is used for securing access to the different functions in IBM Intelligent Operations Center. Figure 2-5 shows the product components installed in this server.
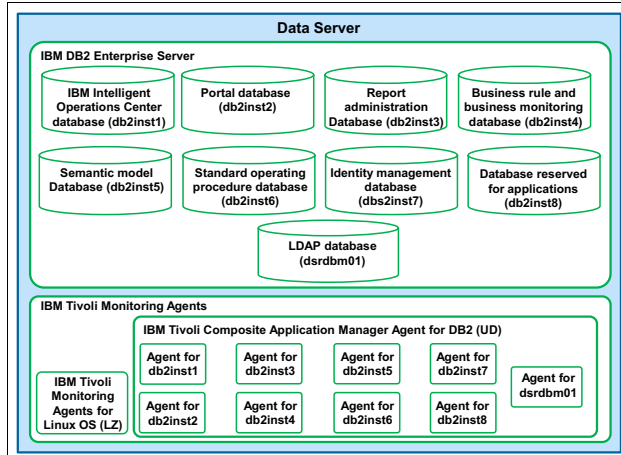
Figure 2-5   Product components installed in the data server

Event server

The event server is responsible for connecting with external data sources, processing incoming events and managing the entire incident response process. The incident response process includes workflows to implement standard operating procedures and the instant messaging infrastructure to allow users to collaborate during an incident or a crisis. Figure 2-6 shows all the product components installed in this server.
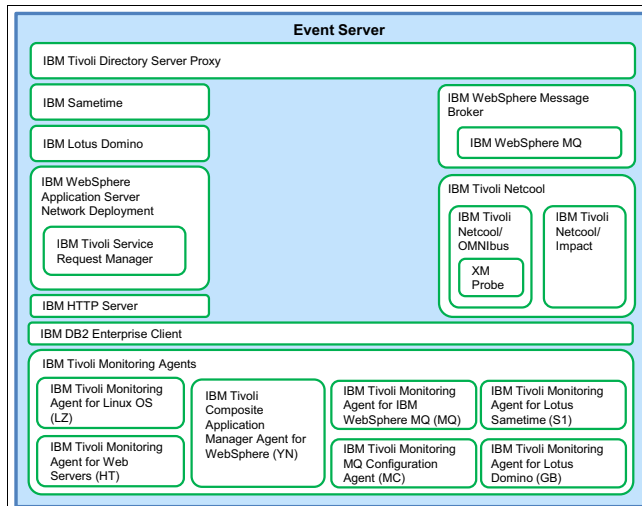
Figure 2-6   Product components installed in the event server

Management server

The management server provides the capabilities to monitor the infrastructure that is used by IBM Intelligent Operations Center, which includes items, such as hardware, operating system, databases, and the web infrastructure. The management server provides the capability to ensure that the entire IBM Intelligent Operations Center solution is performing as expected. The management server also implements key security functions that are used by the other servers in the solution, such as user access and identity management. Figure 2-7 shows the product components that are installed in this server.
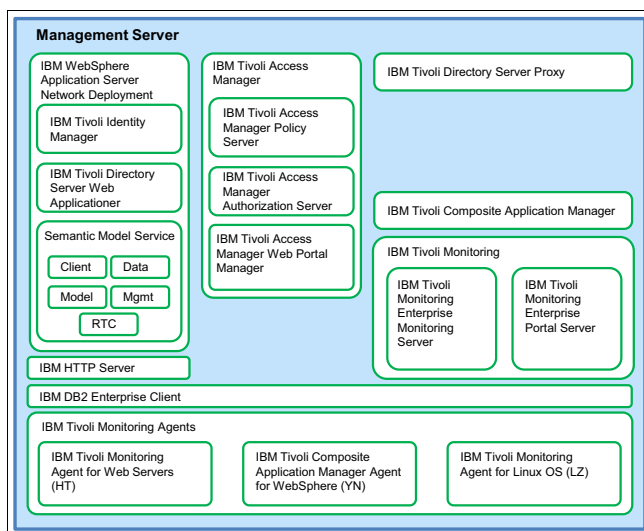
Figure 2-7   Product components installed in the Management server

Installation server

The installation server is just a temporary server that is used during the installation of IBM Intelligent Operations Center. After completion of the installation, this server is no longer needed.

Post-installation configuration

After the IBM Intelligent Operations Center V1.5 installation is complete, some post-installation configuration tasks are required:

► Setting the *session* timeouts.

► Configuring single sign-on (SSO) for instant messaging and, if you have opted to use IPv6, configuring IPv6 support for instant messaging.

► Optional: Configuring Cyber Hygiene, which mitigates known cyber security exposures on the IBM Intelligent Operations Center V1.5 servers.

The steps to complete each of the IBM Intelligent Operations Center V1.5 post-installation configuration tasks are detailed in these sections:

► *Post-installation IBM Intelligent Operations Center configuration*

   http://pic.dhe.ibm.com/infocenter/cities/v1r5m0/topic/com.ibm.ioc.doc/ba_install_config.html

► *Installing and running cyber hygiene step-by-step*

   http://www.ibm.com/support/knowledgecenter/SS3NGB_1.5.0/ioc/ba_install_ch.dita?lang=en

Testing the IBM Intelligent Operations Center V1.5 installation

Follow these steps after you install IBM Intelligent Operations Center V1.5:

1. Verify that all components are functioning correctly by using the *System Verification Check* tool.

   Refer to *Verifying the components* at this website:

   http://www.ibm.com/support/knowledgecenter/SS3NGB_1.5.0/ioc/admin_verifytools.dita?lang=en

   Refer to *Verifying the installation* at this website:

   http://www.ibm.com/support/knowledgecenter/SS3NGB_1.5.0/ioc/ba_install_verify.dita?lang=en

2. Install and configure the platform control tool:

   Refer to *Installing the platform control tool* at this website:

   http://www.ibm.com/support/knowledgecenter/SS3NGB_1.5.0/ioc/ba_install_iop.dita?lang=en

   Refer to *Configuring the platform control tool* at this website:

   http://www.ibm.com/support/knowledgecenter/SS3NGB_1.5.0/ioc/ba_install_config_iop.dita?lang=en

3. Start all services using the platform control tool, using the `Start All` parameter. With all of the IBM Intelligent Operations Center V1.5 services started, log in to IBM Intelligent Operations Center as `wpsadmin` and click **Administration** on the navigation bar.

4. From the sidebar menu, select **Intelligent Operations** → **Administration tools** → **System Verification Check**.

5. Click **Run All Tests** and confirm that all tests run successfully. If assistance is needed, see *Getting help for the platform control tool* at this website:

   http://www.ibm.com/support/knowledgecenter/SS3NGB_1.5.0/ioc/ba_admin_ioccontrol_help.dita?lang=en

2.3.2 Installing IBM i2 Intelligence Analysis Platform

i2 Intelligence Analysis Platform V3.0.3.1 and its components are introduced in Chapter 1, "Integrated Law Enforcement system overview" on page 1.

Figure 2-8 on page 40 provides a different view of the product that shows its product components. The i2 Intelligence Analysis Platform is a Java Platform, Enterprise Edition application that runs as a server in the WebSphere Application Server environment. The following products are prerequisites:

► IBM WebSphere Application Server V8.0.0.4

► IBM WebSphere MQV7.5

► IBM HTTP Server V8.0.0.4

► A database server, such as IBM DB2V9.7 Fix Pack 5, MS SQL Server 2008/2012, or Oracle 10g
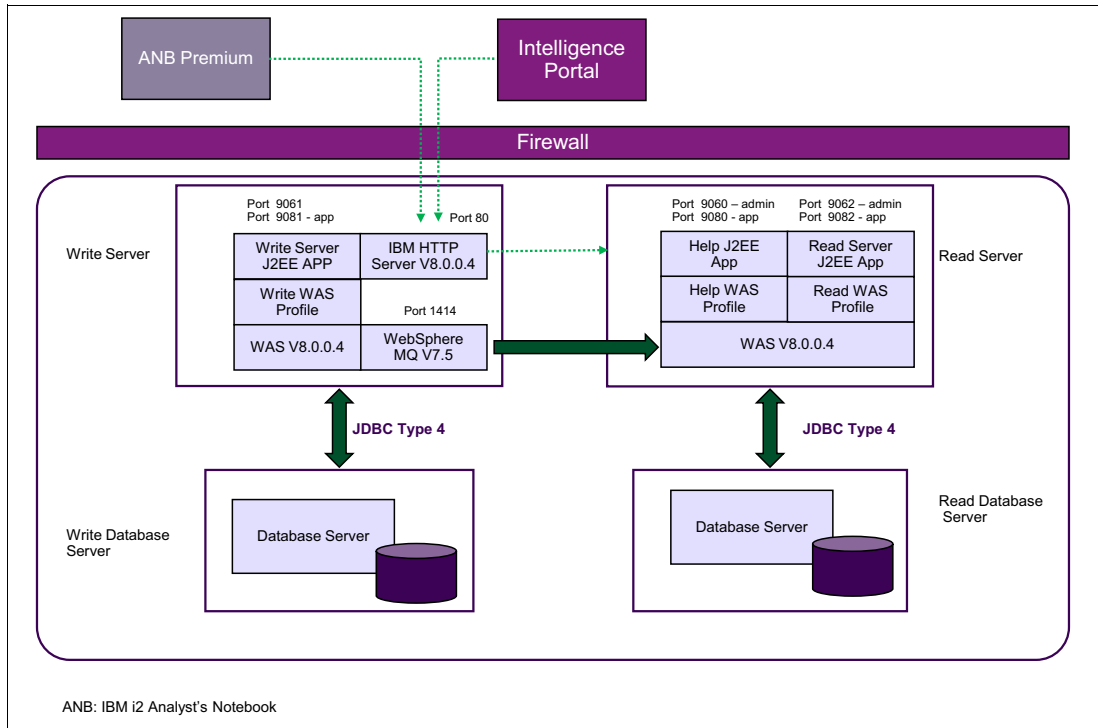
Figure 2-8   Typical two-server i2 Intelligence Analysis Platform V3.0.3.1 deployment

IBM i2 Intelligence Analysis Platform is based on the Command Query Responsibility Segregation (CQRS) architecture. It is designed to have two separate Java Platform, Enterprise Edition servers (WebSphere Application Server V8.0.0.4 in Figure 2-8). One of the servers is referred to as the *write server* for processing all command requests, and the other one is the *read server* for processing all query requests.

These two servers do not have to be two separate physical servers. They can be virtual machines or two WebSphere Application Server V8.0.0.4 profiles on one server. One or two database servers can be allocated, depending on the capacity planning performed during deployment preparation. Figure 2-8 depicts the typical, suggested architecture for a medium-sized organization. IBM i2 Intelligence Analysis Platform V3.0.3.1 runs on Windows or Linux.

Detailed installation instructions for a standard installation on Windows using IBM DB2 are provided in the *IBM i2 Intelligence Analysis Platform Windows Deployment Guide*, which is bundled with the product. This documentation is also available at this website:

http://www.ibm.com/support/docview.wss?uid=swg27041249

Detailed installation instructions for a standard installation on Linux using IBM DB2 are provided in the *IBM i2 Intelligence Analysis Platform Linux Deployment Guide*, which is bundled with the product. This documentation is also available at this website:

http://www.ibm.com/support/docview.wss?uid=swg27041251

Note: Before you install i2 Intelligence Analysis Platform, confirm that your system meets all system requirements.

Prepare the servers

Prepare the servers by installing the following the prerequisite software on the write and read servers:

1. *Write server and read server:* Install the following software, which is included in the *IBM i2 Intelligent Law Enforcement V1.0.1 Multilingual Media Pack* (*BF079ML*):

   – IBM WebSphere Application Server 8.0 Fix Pack 4

   – IBM DB2 Workgroup Server 9.7 Fix Pack 5

   – IBM Data Server Driver for JDBC and SQLJ Version 9.5 Fix Pack 5

2. *Write server:* Install both IBM WebSphere MQ 7.5 and IBM WebSphere MQ SupportPac ME01:

   – IBM WebSphere MQ 7.5 is included in the *IBM i2 Intelligent Law Enforcement V1.0.1 Multilingual Media Pack* (BF079ML).

   – IBM WebSphere MQ SupportPac ME01 is downloadable from this website:

     `http://www.ibm.com/support/docview.wss?rs=171&uid=swg24004684&loc=en_US&cs=utf-8&lang=en`

3. *Write server and read server:* Download and install the following software:

   – Apache Ant 1.8.4 from its project website:

     `http://ant.apache.org`

   – Ant-Contrib 1.0b3 from its project website:

     `http://sourceforge.net/projects/ant-contrib/files/ant-contrib/1.0b3/`

   – Python 2.7 from its project website:

     `https://www.python.org`

Install IBM i2 Intelligence Analysis Platform V3.0.3.1

With the prerequisite server software installed, install the IBM i2 Intelligence Analysis Platform V3.0.3.1 Java Platform, Enterprise Edition application.

Windows installation

> **Note:** This section provides a high-level overview of the deployment steps. For detailed deployment information, see the document, *IBM i2 Intelligence Analysis Platform Deployment Guide,* that is shipped with the product bundle and available at this website:
>
> `http://www.ibm.com/support/docview.wss?uid=pub1sc27509100`

To extract the IBM i2 Intelligence Analysis Platform Deployment Toolkit, follow these steps:

1. Extract the product files from the downloaded distribution, or insert the product CD into the optical drive of your computer, and copy the contents to a new folder on your hard disk.

2. Browse to the IBM i2 Intelligence Analysis Platform 3 folder at the top of the extracted or copied structure.

3. Open a command prompt at this folder, and then use the Java program from your IBM WebSphere Application Server installation to run the following command:

   ```
   java –jar iap-release-3.0.3.jar
   ```

   By default, `java.exe or Java` is installed to the `WebSphere AppServer\java\bin` directory. Running this command displays a license and extracts the files for i2 Intelligence Analysis Platform.

   Details for unpacking the archive are included in the i2 Intelligence Analysis Platform V3.0.3 Release Notes that are bundled with the product. When the archive is unpacked, the i2 Intelligence Analysis Platform V3.0.3 Deployment Guide is in the Deployment Toolkit directory.

4. Install Apache Ant as described in "Prepare the servers" on page 41.

5. Configure the databases on both servers.

6. Configure and initialize an authentication mechanism.

7. Configure WebSphere Application Server on both servers.

8. Install IBM HTTP Server.

9. Configure the reverse proxy.

10. Deploy the Java Platform, Enterprise Edition war file that contains the user help on the read server.

11. Configure and deploy the Java Platform, Enterprise Edition ear file on the write server.

12. Configure and deploy the Java Platform, Enterprise Edition ear file on the read server.

13. Start the Java Platform, Enterprise Edition application that you just deployed on both the write and read servers.

Linux

Support for Linux was introduced in IBM i2 Intelligence Analysis Platform V3.0.3 Fix Pack 1. For IBM i2 Intelligent Law Enforcement V1.0.1, you will need to upgrade IBM i2 Intelligence Analysis Platform V3.0.3 to IBM i2 Intelligence Analysis Platform V3.0.3.1.

Download the i2 Intelligence Analysis Platform V3.0.3 Fix Pack 1 from this website:

http://www.ibm.com/support/docview.wss?uid=swg24034560

Use the fix pack to upgrade an existing installation or to install a new system. The Linux version of the fix pack contains the *IBM i2 Intelligence Analysis Platform Linux Deployment Guide*, which describes the installation process on Linux.

2.3.3 Installing IBM i2 Analyst's Notebook Premium

Figure 2-8 on page 40 shows that one of the clients for IBM i2 Intelligence Analysis Platform is IBM i2 Analyst's Notebook Premium V8.9.3. This is a desktop client application that runs on a Windows client operating system. i2 Analyst's Notebook Premium is a desktop tool that is used by intelligence analysts to perform sophisticated gathering and analysis of intelligence data.

> **Note:** As part of your deployment preparation tasks, you must have determined the number of i2 Analyst's Notebook Premium licenses needed and on which machines they will be installed. Because this is separate from installing on the server side, you need to plan when to install these licenses on the client machines because they might be machines that are actively used by analysts.

Before installing IBM i2 Analyst's Notebook Premium V8.9.3, verify that the following prerequisite software is installed:

► Microsoft .NET Framework 3.5
► Microsoft .NET Framework 4.0
► Internet Explorer 7, 8, or 9 or Mozilla Firefox 3, 3.5, or 3.6
► Microsoft Silverlight 5
► Microsoft Visual C++ 2005 Service Pack (SP)1 ATL Security Update
► Microsoft Windows Installer 4.5

Note: i2 Analyst's Notebook Premium requires full and not client-profile installations of Microsoft .NET.

The i2 Analyst's Notebook Premium distribution media is included in the *IBM i2 Intelligent Law Enforcement V1.0.1 Multilingual Media Pack* (BF079ML). Follow these steps to install it:

1. Extract the product files from the distribution file.
2. Run `setup.exe`.
3. Click **install** in the left menu bar.
4. Follow the instructions on the panel to complete the installation.

2.3.4 Installing IBM i2 COPLINK

The integration of data sources from various locations, including your repositories, with i2 COPLINK is complex. Deployment of i2 COPLINK components is carried out by IBM i2 Lab Service professionals.

A close coordination between your team and the IBM i2 Lab Services team is critical to ensure that all dependencies are in place when the IBM i2 Lab services team is ready to start the deployment.

Figure 2-9 depicts the final results of an i2 COPLINK installation, also known as a local COPLINK node (or local node).
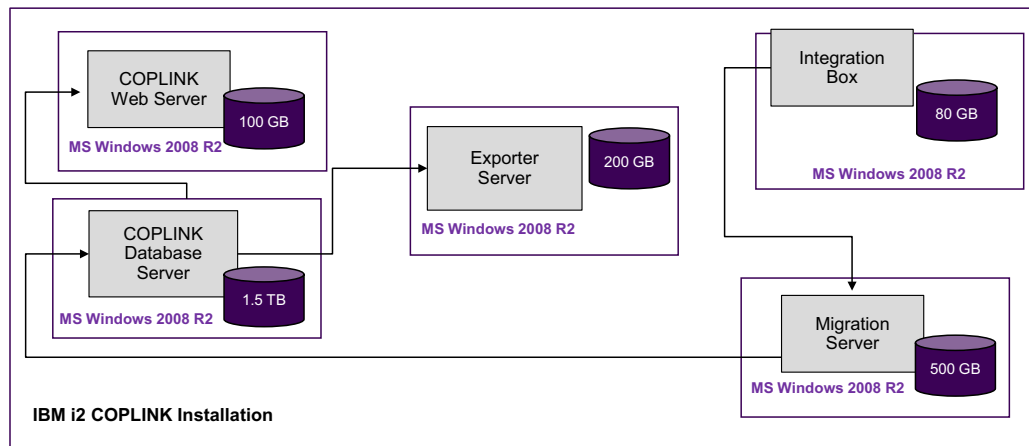
Figure 2-9   Completed installation of IBM i2 COPLINK

Integrated Law Enforcement cross-component stitching

Now that the individual product components of the IBM i2 Integrated Law Enforcement offering are deployed as described in Chapter 2, "Integrated Law Enforcement deployment" on page 21, the next task is to integrate the various components across the base products (IBM i2 Intelligence Analysis Platform, IBM i2 COPLINK, and IBM Intelligent Operations Center). The four extended capabilities that were introduced in Chapter 1, "Integrated Law Enforcement system overview" on page 1 are implemented by integrating the product components.

This integration process is loosely termed *cross-component stitching*, which is referred to as *stitching* in this chapter for brevity. To better understand the concepts of component stitching described in this chapter, read Chapter 1, "Integrated Law Enforcement system overview" on page 1 before proceeding.

This chapter describes the four extended capabilities:

► Situational Awareness
► Reporting for IBM i2 Intelligence Analysis Platform and for i2 COPLINK
► Analysis Search
► Intelligence Portal

These capabilities are all included in the solution offering. If your client chooses not to include some of these capabilities, for example, if they decided to integrate with their existing reporting system, it is possible because all of the capabilities are independent.

This chapter also describes some of the nonstandard configurations for these extended capabilities. The standard topology is described in the document, *IBM i2 Intelligent Law Enforcement V1.0.1 Installation and configuration,* at this website:

https://www-304.ibm.com/support/entdocview.wss?uid=swg27038695

3.1 Getting started

Before proceeding, it is helpful to review the topology of your current installation. Be familiar with the different servers (virtual and physical) that implement each of the individual products and solutions. Document each server in a topology diagram, labeling them with the server name (for example, read server), host names, IP addresses, operating system, network connections, and so on.

If there is a firewall, indicate where the servers are relative to the firewall. Indicate other servers, such as the directory server, the Geographic Information System (GIS) server, network gateways (if relevant), and proxy servers.

Even though the tasks described in this chapter are intended for the extended capabilities, they also provide a general background of the methodology to apply when introducing additional capabilities into i2 Integrated Law Enforcement in the future.

For example, if you are going to introduce additional analytics, such as predictive analytics using IBM SPSS®, you will need to perform these tasks:

► Formulate the overall integration architecture.

► Lay out how the components of SPSS must be stitched with the existing components in i2 Integrated Law Enforcement.

As a reminder, the following server names are for the individual product components (they were described in 2.2.4, "Perform capacity planning" on page 28):

► IBM i2 Intelligence Analysis Platform V3.0.3.x:

  – Write server
  – Read server
  – HTTP server
  – IBM WebSphere MQ server
  – Write database server
  – Read database server

► IBM Intelligence Operations Center V1.5.x:

  – Application server
  – Event server
  – Data server
  – Management server

► IBM i2 COPLINK V4.8:

  – Web server
  – Database server
  – Exporter server
  – Migration server
  – Integration box

3.2 Situational Awareness

Of all the extended capabilities, Situational Awareness requires the most tasks and time to implement stitching the various components. We suggest implementing this capability first.

The main function of Situational Awareness in the i2 Integrated Law Enforcement solution is to provide information about different types of crimes that occurred within a certain geographic area and the vicinity of those crimes. This capability also provides detailed information about these crimes in report format, using IBM Cognos Business Intelligence. Most important, this extended capability can display location information about these crimes visually on a geographic map. The default behavior is to show the most recent crimes that happened in the area, providing a near real-time set of information. The user can also specify a period in the past and filter the type of crimes to show.

Figure 3-1 describes the functional architecture of Situational Awareness as it was introduced in 1.1.3, "Component architecture" on page 6.
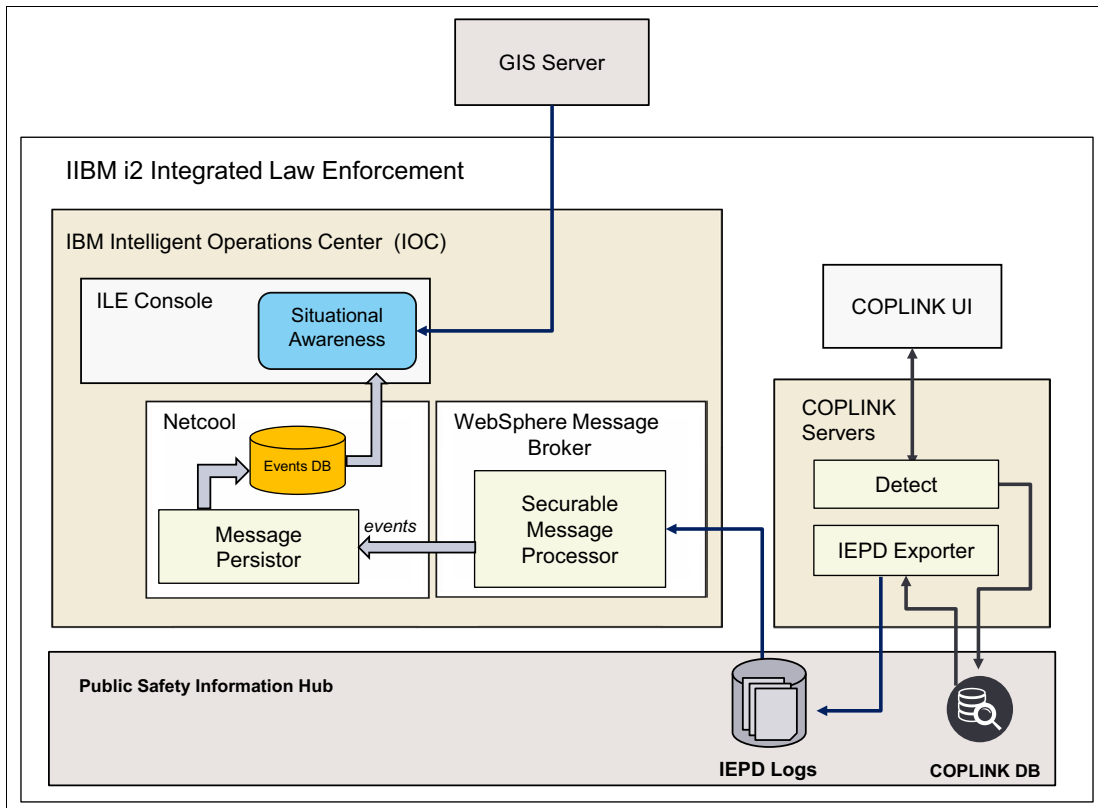
Figure 3-1   Functional architecture of the Situational Awareness extended capability

At a high level, stitching Situational Awareness involves the following steps:

1. Configure the IBM i2 COPLINK Exporter server.

2. Set up a file transfer mechanism in the i2 COPLINK Exporter server.

3. Configure the IBM Intelligent Operations Center database server.

4. Configure IBM Netcool on the IBM Intelligent Operations Center event server.

5. Configure IBM WebSphere Message Broker on the IBM Intelligent Operations Center event server.

6. Deploy the i2 Intelligent Law Enforcement V1.0.1 console (ILE console) on the IBM Intelligent Operations Center portal server.

7. Configure the IBM Cognos Business Intelligence (BI) package for Situational Awareness.

For more information about these required steps for installing Situational Awareness, see the document, *IBM i2 Intelligent Law Enforcement V1.0.1 Installation and configuration,* at this website:

https://www-304.ibm.com/support/entdocview.wss?uid=swg27038695

The core component distribution media contained in the IBM i2 Intelligent Law Enforcement V1.0.1 Multilingual Media Pack (BF079ML) is a `.zip` archive named `I2_ILE_V1.0.1_WIN_ML.zip`, which also contains two archives: `Reporting_Install.zip` and `Sitaw_Install.zip`.

> **Note:** The filename `I2_ILE_V1.0.1_WIN_ML.zip` might have specified `WIN` to indicate the Windows operating system. However, only i2 COPLINK is constrained to run on a Windows operating system.

3.2.1 Prerequisites for stitching Situational Awareness

In preparation for setting up Situational Awareness, perform the following steps:

1. Copy `Sitaw_Install.zip` to a temporary directory on the IBM Intelligent Operations Center application server.

2. Extract the contents of `Sitaw_Install.zip`. The directory structure of the extracted contents has a directory named `SITAW_INSTALL` with the contents shown in Example 3-1.

Example 3-1   Sitaw_Install.zip directory structure of extracted content

```
/SITAW_INSTALL
|    createProbeQueue.mqs
|    Drop_Premium_tag.sh
|    Drop_Standard_tag.sh
|    IBM_i2_Intelligent_Law_Enforcement_Premium-1.0.1.swtag
|    IBM_i2_Intelligent_Law_Enforcement_Standard-1.0.1.swtag
|    install_application.sh
|    install_messaging.sh
|
+---Broker
|        broker_configurable_properties.properties
|        CoplinkToSitAwproject.generated.bar
|        wink-json4j-1.2.0-incubating.jar
|
+---Cognos
|        SitAwExport.zip
|
+---db_scripts
|        insert_data.sql
|
+---impact_policy
|   |    typelist
|   |    typenames
|   |
|   +---DataSources
|   |        datasourcelist
|   |
|   +---etc
|   |        Policy.type
```

```
|    |
|    +---Policy
|    |    |    baglist
|    |    |
|    |    /---com.micromuse.response.dblayer.PolicyOrgNode
|    |              roots
|    |              SITAW_Event_Main.ipl
|    |              SITAW_UTILS.ipl
|    |              templateroots
|    |              template_Aggregation.ipl
|    |              template_EventEnrichment.ipl
|    |              template_XinY.ipl
|    |
|    +---Projects
|    |         projectlist
|    |         SITAW.proj
|    |
|    +---Services
|    |         servicelist
|    |         sitaw_event_reader.props
|    |
|    /---WSlib
+---probe
|    +---java
|    |    /---conf
|    |              sitaw2netcool.xsl
|    |              sitaw_jmsTransport.properties
|    |              sitaw_transformers.xml
|    |
|    /---sitawprobe
|              sitaw_xml.props
|              sitaw_xml.rules
|
/---SITAW
     +---apps
     |         SituationalAwarenessEAR.ear
     |
     +---lib
     |    /---com
     |         /---ibm
     |              /---sitaw
     |                   /---common
     |                        /---resources
     |                                  Messages.properties
     |                                  Messages_en_US.properties
     |                                  Messages_es.properties
     |                                  Messages_pt_BR.properties
     |                                  Messages_zh_CN.properties
     |                                  Messages_zh_TW.properties
     |
     /---scripts
          |    setupWAS.py
          |
          /---portallayout
                    iapPortalPage.xml
```

```
incidentsPage.xml
registerPortlets.xml
```

The following sections provide the details for performing the seven implementation steps described at a high level in 3.2, "Situational Awareness" on page 46.

3.2.2 Configuring IBM i2 COPLINK Exporter server

To be able to show the incidents of crimes on a near real-time basis, i2 COPLINK Exporter must export the latest crime information regularly. This section describes how to set up the i2 COPLINK Exporter server regular exporting schedule. The choice of schedule is left to you and your client. You need to seriously consider the amount of data that is being pumped to IBM Intelligent Operations Center from your selected schedule because it will affect the overall system performance.

To configure i2 COPLINK Exporter, perform the following steps:

1. The i2 COPLINK Exporter transforms and exports data from the i2 COPLINK database into XML files. These XML files are saved on the local file system of the i2 COPLINK Exporter server. Designate and create a directory where the exported XML files will be stored, for example, `C:\IBM\Export`.

2. Open the i2 COPLINK Exporter console on the i2 COPLINK Exporter server, as shown in Figure 3-2 on page 51.

> **Note:** The output section is in the panel on the left. The Data Source is in the panel on the right. There is a scheduler in the center.
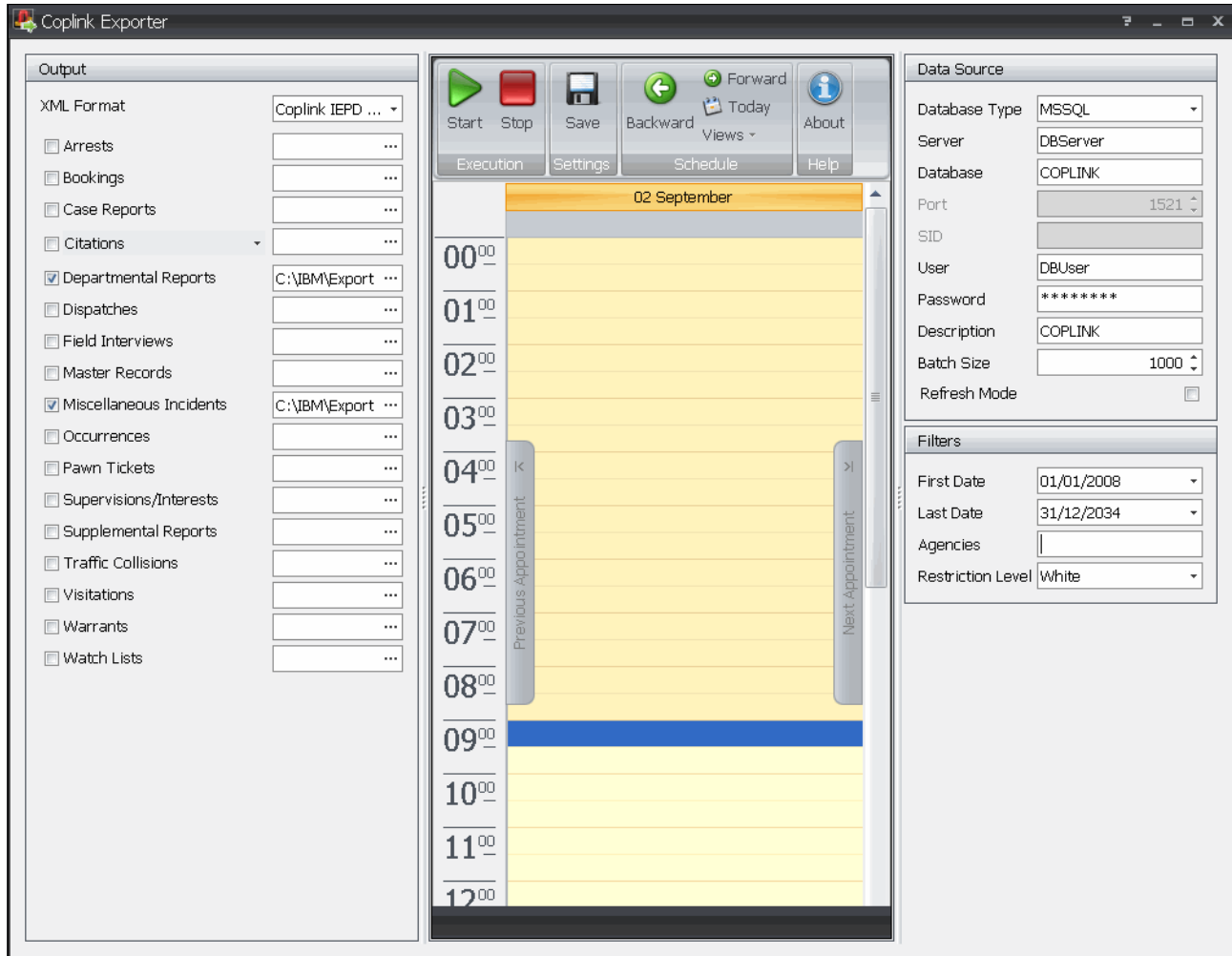
Figure 3-2   i2 COPLINK Exporter console

3. In the Output section, set the XML Format to **i2 COPLINK IEPD 1.0**.

4. Select both **Departmental Reports** and **Miscellaneous Incidents**.

5. Set the output directory, for both report types, to the export directory **C:\IBM\Export** that was created in step 1 on page 50.

6. In the Data Source panel, enter the connection details for the i2 COPLINK database:

   – If the i2 COPLINK database is a Microsoft SQL server database, specify the database server name, the database name, and the login credentials of a user on the SQL server that has access to the i2 COPLINK database.

   – If the i2 COPLINK database is an Oracle database, specify the database server name, the Oracle System Identifier (SID), and login credentials of a user on the Oracle server that has access to the i2 COPLINK database.

   Figure 3-3 on page 52 shows the Data Source section after it is set up for an Oracle database.
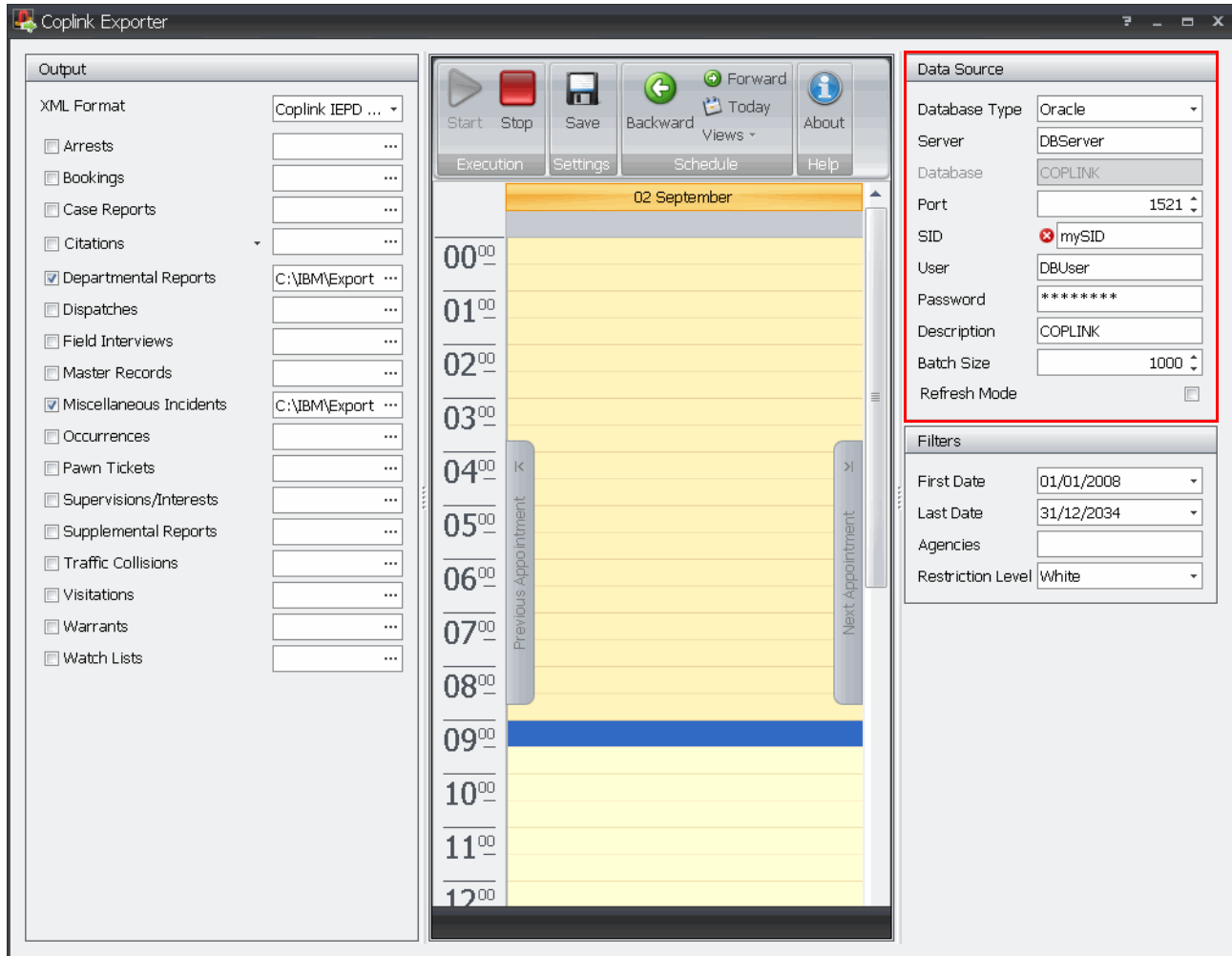
Figure 3-3   Data Source information populated in the i2 COPLINK Exporter for an Oracle database

7. Set the necessary start and end dates in the Filters section, and set the restriction level to **White**. For information about the restriction level colors in the Filter section, see 4.2.3, "IBM i2 COPLINK security model" on page 121.

8. i2 COPLINK Exporter can be run on a scheduled basis, using the scheduler in the center section of the panel (see Figure 3-3). Follow these steps to configure a scheduled export:

    a. Right-click the scheduler section of the panel and select the time for the export to start.

    b. Click **New Recurring Appointment**.

    c. In the Run Time window, enter a subject.

    d. Leave the "Terminate execution at end time" check box unchecked.

    e. Click **Recurrence** and select the appropriate recurrence values.

    f. Click **OK** in the Run Time window to save the schedule.

Now that you have set up the i2 COPLINK Exporter server and the schedule for XML exporting, the next step is to set up the mechanism for moving these exported files to IBM Intelligence Operations Center.

3.2.3 Setting up file transfer in IBM i2 COPLINK Exporter server

The saved, exported information relates to crimes that occurred within the jurisdiction of the i2 COPLINLK database instance from which the data was exported. The exported XML files use Information Exchange Package Documentation (IEPD) to exchange data. To implement Situational Awareness, this data must be consumed by IBM Intelligent Operations Center. Therefore, make the directory where the XML files are saved readable so that they can be transferred by the Linux processes that are running in IBM Intelligent Operations Center.

A common approach is to install and configure a file transfer server using either a non-secure file transfer protocol (FTP) or a secure FTP (SFTP) protocol. This server must be installed in the i2 COPLINK Exporter server. There are many file transfer programs to choose from. Whichever one you choose, ensure that it allows basic authentication (*username* and *password*). You also need to configure it so that it allows the deletion of files in the published directory.

> **Note:** In Figure 3-1 on page 47, the arrow line from the IEPD logs to the Securable Message Processor represents this file transfer. After the XML files are moved to the Securable Message Processor, those files must be deleted from the source directory of the file transfer server.

3.2.4 Configuring the IBM Intelligent Operations Center database server

When the XML files from the IBM i2 COPLINK Exporter server are moved to the IBM Intelligent Operations Center, Situational Awareness adds new event data to the existing IBM Intelligent Operations Center database, which is called `IOCDB`. An SQL script is provided in the Situational Awareness installer that adds Situational Awareness-specific data related to crime types, severity, urgency, and so on to the `IOCDB` database.

To run the SQL script, perform the following steps:

1. Copy the script `insert_data.sql` from the `/SITAW_INSTALL/db_scripts` directory on the IBM Intelligent Operations Center application server to `/tmp/install/db_scripts/insert_data.sql` on the IBM Intelligent Operations Center data server.

2. Log in to the IBM Intelligent Operations Center database server as the `root` user and navigate to the `/tmp` directory.

3. Issue the following commands:

   ```
   chmod -R 755 ./Install
   su db2inst1
   db2 -tvf /tmp/Install/db_scripts/insert_data.sql
   ```

3.2.5 Configuring IBM Netcool on the IBM Intelligent Operations Center event server

To display Situational Awareness on the i2 Intelligent Law Enforcement console, the situational events must be associated with IBM Netcool/OMNIbus and Netcool/Impact on the IBM Intelligent Operations Center event server.

Perform the following steps:

1. Create a `/tmp/install` directory on the IBM Intelligent Operations Center event server.

2. Copy the `impact_policy` and `probe` directories and the `install_messaging.sh` script from the `/SITAW_INSTALL/` directory on the IBM Intelligent Operations Center application server to the `/tmp/install` directory on the IBM Intelligent Operations Center event server.

3. Log in to the IBM Intelligent Operations Center event server as `root`, then navigate to the `/tmp` directory and issue the following commands:

   chmod -R 755 ./Install

   cd Install

   ./install_messaging.sh

4. Edit the XML probe configuration file `/opt/IBM/netcool/omnibus/java/conf/sitaw_jmsTransport.properties` and enter values for the **username**, **password**, and **hostname** parameters. See the example shown in Figure 3-4.

```
#
#/*
# * Licensed Materials - Property of IBM
# *
# * <PID of containing product>
# *
# * (C) Copyright IBM Corp. <YEAR1>, <YEAR2> All rights reserved.
# *
# * US Government Users Restricted Rights - Use, duplication or
# * disclosure restricted by GSA ADP Schedule Contract with
# * IBM Corp.
# */

initialContextFactory=com.ibm.websphere.naming.WsnInitialContextFactory

queueConnectionFactory=jms/ioc.mb.con.factory

queueName=jms/coplink.out.q
username=mqm
password=<password>
providerURL=iiop://<IOC_application_server_hostname>:10035
```

Figure 3-4   Configuring sitaw_jmsTransport.properties

5. Start the IBM Netcool/Impact policy service.

   Open the IBM Netcool administrator console by logging in to the URL:

   `http://<IOC_Application_Server>:9080/nci/login_main.jsp`

6. In the Service Status area, click **Start** next to the `SITAW_Event_Reader`.

7. Start the XML monitoring probe on the IBM Intelligent Operations Center event server.

   Log in as `root` and run the following command:

   /opt/IBM/netcool/omnibus/probes/nco_p_xml -name sitaw_xml -propsfile /opt/IBM/netcool/omnibus/probes/sitawprobe/sitaw_xml.props &

3.2.6 Configuring IBM WebSphere Message Broker on the IBM Intelligent Operations Center event server

IBM Intelligent Operations Center communicates with external systems through a message bus that is implemented by IBM WebSphere Message Broker. IBM WebSphere Message Broker uses message flows to process incoming messages from external systems. A message flow is analogous to a script that the IBM WebSphere Message Broker runs to perform a specific task. Situational Awareness uses a message flow to prepare the exported i2 COPLINK data for import into IBM Intelligent Operations Center. You need to configure IBM WebSphere Message Broker to run a specific message flow for importing the i2 COPLINK data.

> **Note:** In general, the approach is to send events from an application that you plan to integrate with IBM Intelligent Operations Center where that application is a back-end data provider for some of IBM Intelligent Operations Center services, including portal services.

Perform the following steps:

1. Copy the `Broker` directory from the `/SITAW_INSTALL/` directory on the IBM Intelligent Operations Center application server to the `/tmp/install` directory on the IBM Intelligent Operations Center event server.

2. Log in to the IBM Intelligent Operations Center event server as the `root` user. Set up JavaScript Object Notation (JSON4J) by copying `/tmp/Install/Broker/wink-json4j-1.2.0-incubating.jar` to the `/var/mqsi/shared-classes/` directory.

3. Make a copy of the `/tmp/Install/Broker/broker_configurable_properties.properties` file in the `/tmp` directory, and edit the new copy of the file to set the parameters listed in Table 3-1.

Table 3-1   Parameters in broker_configurable_properties.properties

Parameter	Value
`ProcessCoplinkFile#ReadCoplinkExporterFiles.inputDirectory`	The local mount point on the IBM Intelligent Operations Center event server if the i2 COPLINK `.xml` files are being exported to a shared directory
`ProcessCoplinkFile#ReadCoplinkExporterFiles.filenamePattern`	A pattern that identifies the name of the exported files, usually `.xml`
`ProcessCoplinkFile#ReadCoplinkExporterFiles.fileFtp`	A flag (set to `Yes` or `No`) that indicates whether a local shared directory, FTP, or SFTP service is being used
`ProcessCoplinkFile#ReadCoplinkExporterFiles.remoteTransferType`	Indicates whether FTP or SFTP is being used
`ProcessCoplinkFile#ReadCoplinkExporterFiles.fileFtpServer`	The host name of the SFTP or FTP server
`ProcessCoplinkFile#ReadCoplinkExporterFiles.fileFtpUser`	A pseudo-user name used to access the FTP server
`ProcessCoplinkFile#ReadCoplinkExporterFiles.fileFtpDirectory`	The relative directory on the FTP server where the exported `.xml` files are stored

Example 3-2 shows a sample `broker_configurable_properties.properties` file.

Example 3-2   Sample broker_configurable_properties.properties file

```
ProcessCoplinkFile#eventTypeMappingPropertiesFile=com.ibm.ile.sitaw.resources.eventTypeMap
ProcessCoplinkFile#categoryMappingPropertiesFile=com.ibm.ile.sitaw.resources.categoryMap
ProcessCoplinkFile#categoryToHeadlineMappingPropertiesFile=com.ibm.ile.sitaw.resources.categoryT
oHeadlineMap
ProcessCoplinkFile#ReadCoplinkExporterFiles.inputDirectory=/tmp/brokerImport
ProcessCoplinkFile#ReadCoplinkExporterFiles.filenamePattern=*.xml
ProcessCoplinkFile#ReadCoplinkExporterFiles.fileFtp=yes
ProcessCoplinkFile#ReadCoplinkExporterFiles.remoteTransferType=FTP
ProcessCoplinkFile#ReadCoplinkExporterFiles.fileFtpServer=COPLINK_Exporter.mydomain.com
ProcessCoplinkFile#ReadCoplinkExporterFiles.fileFtpUser=coplink
ProcessCoplinkFile#ReadCoplinkExporterFiles.fileFtpDirectory=/IBM/Export/
```

4. Apply the updated properties to the provided message flow before the deployment.

   The Situational Awareness installer provides an IBM WebSphere Message Broker bar file called `CoplinkToSitAwproject.generated.bar`. A *bar file* is an IBM WebSphere Message Broker artifact that implements a message flow. The configurable properties that must be updated are stored in the `/tmp/broker_configurable_properties.properties` file.

   To update the message flow, log in as the `root` user on the IBM Intelligent Operations Center event server, and issue the following commands:

   cd /opt/IBM/mqsi/8.0.0.0/bin

   source ./mqsiprofile

   mqsiapplybaroverride -b /tmp/Install/Broker/CoplinkToSitAwproject.generated.bar -k CoplinkToSitAw -p /tmp/broker_configurable_properties.properties

5. Deploy the message flow.

   The message flow can now be deployed into IBM WebSphere Message Broker by running the following commands:

   cd /opt/IBM/mqsi/8.0.0.0/bin

   source ./mqsiprofile

   mqsicreateexecutiongroup IOC_BROKER -e SitAw

   mqsideploy IOC_BROKER -e SitAw -a /tmp/Install/Broker/CoplinkToSitAwproject.generated.bar

   If the message flow is being redeployed to replace an existing message flow, the **-m** parameter must be added to the **mqsideploy** command. In this case, the **mqsideploy** command looks like this command:

   mqsideploy IOC_BROKER -e SitAw -m -a /tmp/Install/Broker/CoplinkToSitAwproject.generated.bar

6. Set the correct *username* and *password* for the file transfer.

For environments using FTP or SFTP, map the pseudo-username defined by the `ProcessCoplinkFile#ReadCoplinkExporterFiles.fileFtpUser` parameter in the `broker_configurable_properties.properties` (step 3 on page 55) to the actual *username* and *password* on the FTP or SFTP server by using the `mqsisetdbparms`:

`cd /opt/IBM/mqsi/8.0.0.0/bin`

`source ./mqsiprofile`

`/opt/IBM/mqsi/8.0.0.0/bin/mqsisetdbparms IOC_BROKER -n ftp::coplink -u username -p password`

The variables are defined:

`-n` is the *username* defined by the `ProcessCoplinkFile#ReadCoplinkExporterFiles.fileFtpUser` parameter.

`-u` is the actual *username* on the FTP or SFTP server.

`-p` is the *password* for the actual FTP or SFTP user.

Configuring IBM WebSphere Message Broker behind a web proxy

This section describes a special case of configuring IBM WebSphere Message Broker on the IBM Intelligent Operations Center event server when IBM WebSphere Message Broker is installed behind a web proxy.

Figure 3-1 on page 47 shows that a GIS server is needed to implement the Situational Awareness capability. The goal is to present a depiction of the occurrence of crimes in the vicinity of a geographic location on a map widget, so a map service is required that provides maps and layers (among other things) to Situational Awareness. Whether your client is using their own GIS server, or one that is owned by an external provider, it is possible that, for security reasons, IBM Intelligent Operations Center (and all of its components) is not allowed to access the GIS server directly, for example, because it is behind a firewall. A solution to this problem is to use a web proxy between IBM Intelligent Operations Center and the GIS server. This section describes how to configure IBM WebSphere Message Broker in this situation.

During validation of the exported XML files, the message flow that runs on IBM WebSphere Message Broker performs geolocation based on coordinates provided in the XML file. When IBM Intelligent Operations Center is installed behind a web proxy, the standard message flow (that is, the bar file) that ships with i2 Intelligent Law Enforcement V1.0.1 cannot communicate with the GIS server. Therefore, the validation of the crime events fails. However, an updated `.bar` file that allows IBM WebSphere Message Broker to communicate through a proxy is available. The description details of the updated `.bar` file and the file itself are available from IBM Support. Contact IBM Support to get the latest version of the bar file.

Perform the following steps to configure IBM WebSphere Message Broker to use the proxy server:

1. Update the `.bar` file following the instructions described in step 1 on page 55 through step 4 on page 56.

2. Install the IBM WebSphere Message Broker Explorer for Linux plug-in in WebSphere MQ Explorer on the IBM Intelligent Operations Center event server. You can download the WebSphere Message Broker Explorer plug-in from this website:

   http://www-01.ibm.com/support/docview.wss?uid=swg24012457

3. Log in to the IBM Intelligent Operations Center event server as the `root` user. Use an X Window System session because WebSphere MQ Explorer is a graphical user interface.

4. Open a terminal session, navigate to the `/usr/bin`, and enter the following command to launch the WebSphere MQ Explorer:

   `./strmqcfg`

5. In WebSphere MQ Explorer, navigate to **Brokers** → **IOC_BROKER** → **SitAw** → **CoplinkToSitAW** → **ProcessCoplinkFile**. See Figure 3-5.
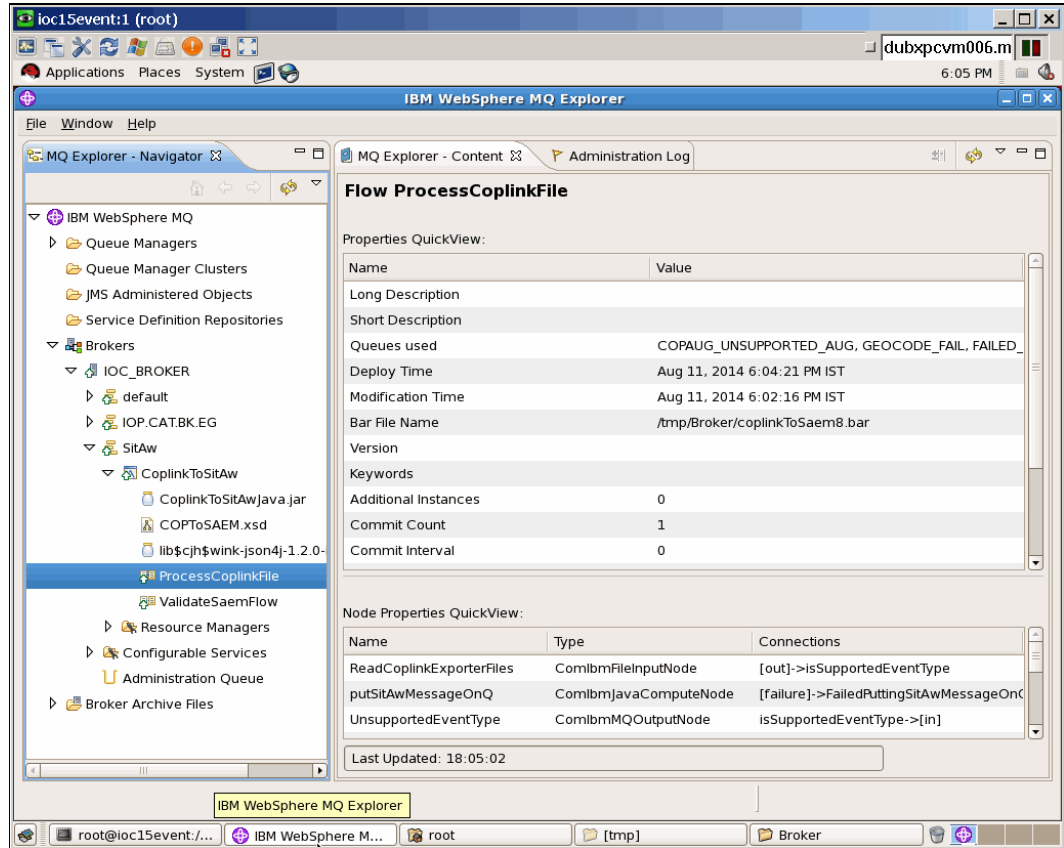
Figure 3-5   Navigating in WebSphere MQ Explorer

6. Right-click **ProcessCoplinkFile** as shown in Figure 3-5 and select **Properties**.

7. Select **User Defined Properties** and enter the necessary values in the proxyPassword and proxyURL fields, as shown in Figure 3-6 on page 59.
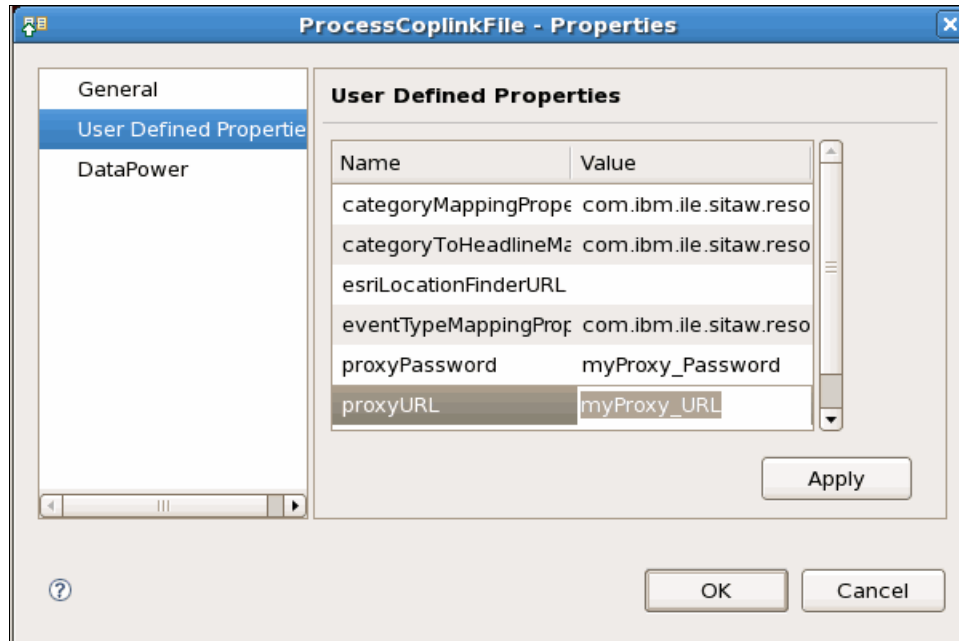
Figure 3-6   User Defined Properties of the ProcessCoplinkFile

8. Click **Apply** → **OK**.

9. Close WebSphere MQ Explorer.

3.2.7 Deploying the i2 Intelligent Law Enforcement V1.0.1 console on the IBM Intelligent Operations Center portal server

The i2 Intelligent Law Enforcement V1.0.1 console is the main portal developed specifically to include user interfaces for all of the i2 Integrated Law Enforcement extended capabilities. These front-end user interfaces are implemented through IBM WebSphere Portal technology and are based on portal pages, portlets, and iWidgets. The WebSphere Portal Server runs on the IBM Intelligent Operations Center application server. Follow these steps to deploy the i2 Intelligent Law Enforcement V1.0.1 console:

1. Log in to the IBM Intelligent Operations Center application server as the `root` user.

2. Navigate to the directory where the Situational Awareness installer was expanded, and run the following commands:

   `chmod -R 755 ./*`

   `./install_application.sh`

   During the execution of the `install_application.sh` script, you will be prompted to enter your WebSphere administrator credentials.

3. Install the software tag.

   The *software tag* is used by IBM Software Support when installing updates or software fixes to determine which version of i2 Intelligent Law Enforcement is installed.

   If i2 Intelligent Law Enforcement Premium is installed, run the following command:

   `./Drop_Premium_tag.sh`

   For i2 Intelligent Law Enforcement Standard, run the following command:

   `./Drop_Standard_tag.sh`

4. Log in to the WebSphere Application Server administrative console in the IBM Intelligent Operations Center application server. Point your browser to the following URL:

   `https://<IOC_application_server_hostname>:9043/ibm/console/`

5. Click **Resources** → **Asynchronous beans** → **Work Managers** and change the scope to **Cluster=Portal Cluster**.

6. Click **New** and enter the parameter values indicated in Table 3-2.

   Table 3-2   New work manager properties

Parameter	Value
Name	`SITAW_workmanager`
JNDI Name	`wm/SITAW_workmanager`

7. Click **Apply** and then save the configuration.

8. Click **Resources** → **Schedulers**, and change the scope to **Node=PortalNode1,Server=WebSphere_Portal**.

9. Click **New**, and enter the parameter values shown in Table 3-3.

   Table 3-3   New scheduler properties

Parameter	Value
Name	`SITAW_event_cleanup`
JNDI Name	`sched/sitaw_event_cleanup`
Data source JNDI Name	`jdbc/ioc`
Data source alias	`dbuser`
Table prefix	`sitaw`
Work manager JNDI name	`wm/SITAW_workmanager`

10. Click **Apply** and save the configuration.

11. Click the box next to the newly created scheduler. At the top of the table, click **Create Tables**.

12. After the tables are created, click the box next to the scheduler again, and click **Verify Tables**.

13. In a browser, log in to the WebSphere Portal Server as a user with administrative rights.

14. Click **Administration** on the top menu bar to open the WebSphere Portal administration console.

15. On the left menu bar, click **Portal Settings** → **Import XML,** and select the **SITAW/scripts/portallayout/registerPortlets.xml** file from the directory where the Situational Awareness installer was extracted. Click **Import**.

16. Select the **SITAW/scripts/portallayout/incidentsPage.xml** file in this directory and click **Import**.

17. Select the **SITAW/scripts/portallayout/iapPortalPage.xml** file in this directory and click **Import**.

> **Important Note:** This step installs and configures the Intelligence Portal extended capability.

18. Log out of WebSphere Portal.

19. Restart the WebSphere Portal cluster, either from the WebSphere Application Server administrative console or by running the `IOCControl.sh` script on the IBM Intelligent Operations Center management server.

3.2.8 Configuring the IBM Cognos BI package for Situational Awareness

The last task to complete the implementation of the Situational Awareness extended capability is the crime details reporting, which is based on historical data from i2 COPLINK. Crime details reporting is a function of Situational Awareness. Perform the following steps:

1. Copy the **SitAwExport.zip file** from the `/SITAW_INSTALL/cognos` directory to the `/IBM/cognos/c10_64/deployment` directory on the IBM Intelligent Operations Center application server.

2. Launch IBM Cognos Connections pointing the browser to the following URL where *<IOC_application_server_hostname>* is the IBM Intelligent Operations Center application server host name:

   `http://<IOC_application_server_hostname>:9082/p2pd/servlet/dispatch/ext`

3. Log in as an administrator, using `wpsadmin` for example.

4. Click **Launch** in the upper-right corner of IBM Cognos Connection and choose **IBM Cognos Administration** from the drop-down list. See Figure 3-7.
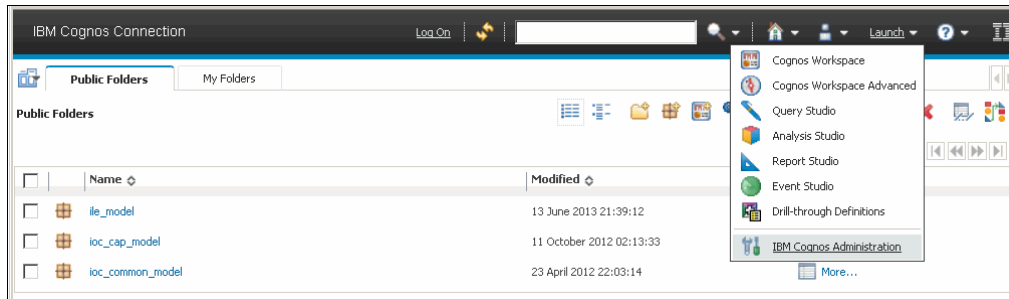
Figure 3-7   Launch IBM Cognos Administration

5. In IBM Cognos Administration, select **Content Administration** and click the **New Import** icon. See Figure 3-8.
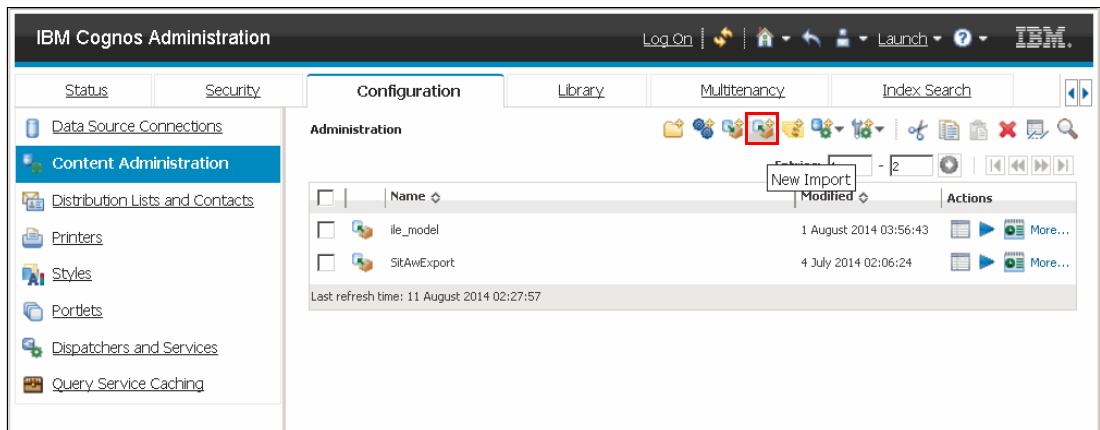
Figure 3-8   IBM Cognos Administration New Import icon

6. Select **SitAwExport** and click **Next** in each window until you reach the wizard named "Select the public folders content - New Import".

7. Click the check box next to the **SitAw** directory, and click **Next** in each window until you reach the wizard named "Select an action - New Import".

8. Click **Save**.

9. Click **Run**.

10. Click **Finish**.

11. In the Run with Options window, click **Now**.

12. Click **Run** → **OK**.

These steps complete the configuration of the IBM Cognos report template for the Situational Awareness extended capability. You can test this extended capability now.

3.3 Reporting

This section describes how to stitch the reporting extended capabilities both for i2 Intelligence Analysis Platform reports and i2 COPLINK reports.

This section is divided in four major parts:

► Setting up i2 Intelligence Analysis Platform reporting

► Setting up i2 COPLINK reporting

► Deploying IBM Cognos BI Report packages

► Configuring the portal pages for reports

The last two sections apply to both i2 Intelligence Analysis Platform and i2 COPLINK.

3.3.1 Setting up i2 Intelligence Analysis Platform Reporting

As described in Chapter 1, "Integrated Law Enforcement system overview" on page 1, the architecture of the i2 Intelligence Analysis Platform reporting includes a separate IBM DB2 relational database, known as the IBM i2 Intelligence Analysis Platform Reporting database. This database is populated using a reporting service installed on the i2 Intelligence Analysis Platform read server.

Reports that are requested through the i2 Intelligent Law Enforcement console are generated by IBM Cognos Business Intelligence from data retrieved from the i2 Intelligence Analysis Platform reporting database. See Figure 3-9 on page 63.
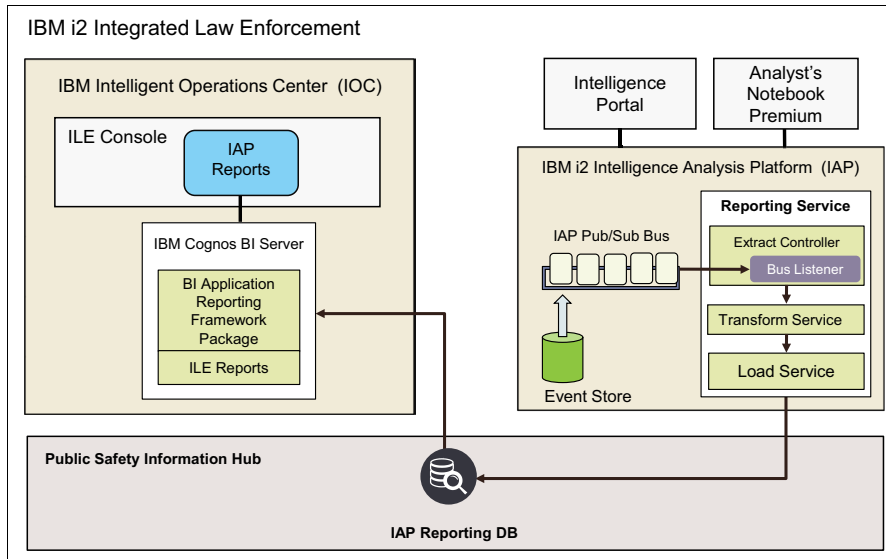
Figure 3-9   i2 Intelligence Analysis Platform reporting in the i2 Integrated Law Enforcement console

To build the infrastructure shown in Figure 3-9, the stitching of i2 Intelligence Analysis Platform reporting involves four main steps:

1. Create the i2 Intelligence Analysis Platform Reporting database.
2. Deploy the Reporting service on the i2 Intelligence Analysis Platform read server.
3. Deploy the IBM Cognos reporting package.
4. Deploy the Reporting portal pages in the IBM Intelligent Operations Center portal.

This section describes the creation of the i2 Intelligence Analysis Platform Reporting database and deployment of the Reporting service on the i2 Intelligence Analysis Platform.

Prepare for installation

You need to prepare all of the artifacts that will be needed for installing the Reporting extended capabilities. The full details of the steps required to install i2 Intelligence Analysis Platform Reporting are documented in *BM i2 Intelligent Law Enforcement V1.0.1 Installation and configuration* at this website:

https://www-304.ibm.com/support/entdocview.wss?uid=swg27038695

The core component distribution media contained in the IBM i2 Intelligent Law Enforcement V1.0.1 Multilingual Media Pack (BF079ML) is a .zip archive named I2_ILE_V1.0.1_WIN_ML.zip, which also contains two archives: Reporting_Install.zip and Sitaw_Install.zip.

The directory structure of the extracted contents has a directory named REPORTING_INSTALL with the contents shown in Example 3-3.

Example 3-3   Reporting_Install.zip - Directory structure of extracted content

```
\REPORTING_INSTALL
+---Cognos
|   |   ILE_FM_project.zip
|   |   ile_model.zip
|   +---translations
+---Coplink_Reporting_Installer
```

```
+---IAP_Reporting_Installer
|   |    .pydevproject
|   |    install.bat
|   |    install.properties
|   +---db
|   |        create_schema.sql
|   |        create_views.sql
|   |        drop_views.sql
|   |        insert_metadata.sql
|   +---ear
|   |   +---en
|   |   |   |    iap-release-3.0.3.1-read.ear
|   |   |   |    iap-release-3.0.3.1-write.ear
|   |   +---es
|   |   +---pt_br
|   |   +---zh_cn
|   |   +---zh_tw
|   \---was
|            deploy_ear.py
|            setup_activation_spec.py
|            setup_datasource.py
|            setup_jaas.py
|            setup_jdbcprovider.py
|            setup_topic.py
|            start_application.py
|            stop_server.py
|            uninstall_ear.py
|            wsadminlib.py
+---ILE_Console_Pages
```

The i2 Intelligence Analysis Platform Reporting installer includes the following information:

► A series of python scripts

► Windows batch file that is used to call the python scripts

► SQL script for creating and populating the IBM i2 Intelligence Analysis Platform Reporting database

► Localized versions of the i2 Intelligence Analysis Platform Reporting Java Platform, Enterprise Edition application files

► An XML file that is used to generate the report portal page within the i2 Intelligent Law Enforcement console

► A properties file that contains various parameters that are used during the installation process

Configure the i2 Intelligence Analysis Platform read server

You are now ready to prepare the read server for creating the Reporting database.

Note: In a standard configuration, the Report database uses IBM DB2 on a Windows operating system.

Perform the following steps:

1. Modify the `install.properties` file in the `REPORTING_INSTALL/Intelligence Analysis Plat-form_Reporting_Installer` directory to configure the parameters listed in Table 3-4.

Table 3-4   Properties in the install.properties file

Parameter	Setting
jaas.userId	A DB2 administrative user, usually the same user as the user for the i2 Intelligence Analysis Platform databases.
jaas.password	The password for the DB2 user.
datasource.serverName	The DB2 server host name. If the DB2 instance is local, `localhost` can be used.
activationSpec.queueManager.hostname	The *hostname* of the i2 Intelligence Analysis Platform write server.
ear.fileLocation	The path to the language version of the i2 Intelligence Analysis Platform Reporting `.ear` file, appropriate for the installation.
app.name	The name of the application as registered in WebSphere Application Server.
was.user	The WebSphere Application Server administrator in the i2 Intelligence Analysis Platform read server, usually `wasadmin`.
was.password	The password for the WebSphere Application Server administrator in the Intelligence Analysis Platform read server.
was.home	The directory in the i2 Intelligence Analysis Platform read server where WebSphere Application Server is installed, for example, `C:\IBM\WebSphere`. **Note**: Do not include a trailing forward slash (`\`) or `\AppServer` in this path.

Example 3-4 provides sample settings for the `install.properties` file.

Example 3-4   Example install.properties file

```
# Credentials for DB2 database at READ server
jaas.userId=myIAPDB2user
jaas.password=myIAPDB2password

# Name of host where DB2 is installed on
datasource.serverName=localhost

# Name of WRITE server
activationSpec.queueManager.hostname=myIAPWriteServer.mydomain.com

# Location of the EAR file to install

# Credentials for DB2 database at READ server
jaas.userId=myIAPDB2user
jaas.password=myIAPDB2password

# Name of host where DB2 is installed on
datasource.serverName=localhost
```

```
# Name of WRITE server
activationSpec.queueManager.hostname=myIAPWriteServer.mydomain.com

# Location of the EAR file to install
ear.fileLocation=C:\Reporting_Install\IAP_Reporting_Installer\ear\en

# Name of Enterprise Application (name of the EAR file, without '.ear' extension)
app.name=IAPReportingEar

# Credentials for READ WebSphere
was.user=myWASadminuser
was.password=myWASadminpassword

# Home directory of READ WebSphere
was.home=C:\IBM\WebSphere

###

#datasource.jdbcProvider=jdbcProvider.name

was.profileName=ApolloRead
was.serverName=ApolloServerRead

jaas.alias=ApolloNode01Read/db2admin

jdbcProvider.name=DB2 Universal JDBC Driver Provider (XA)
jdbcProvider.databaseType=DB2
jdbcProvider.providerType=DB2 Universal JDBC Driver Provider
jdbcProvider.implementationType=XA data source
jdbcProvider.description=XA DB2 Universal JDBC Driver-compliant Provider. Datasources created
under this provider support the use of XA to perform 2-phase commit processing. Use of driver
type 2 on WAS z/OS is not supported for datasources created under this provider.
jdbcProvider.classpath=${DB2UNIVERSAL_JDBC_DRIVER_PATH}/db2jcc.jar;${UNIVERSAL_JDBC_DRIVER_PATH}
/db2jcc_license_cu.jar;${DB2UNIVERSAL_JDBC_DRIVER_PATH}/db2jcc_license_cisuz.jar
jdbcProvider.nativePath=${DB2UNIVERSAL_JDBC_DRIVER_NATIVEPATH}

datasource.name=reportingDN
datasource.jdbcProviderName=DB2 Using IBM JCC Driver (XA)
#datasource.description=
datasource.jndiName=ds/REPORTING
datasource.statementCacheSize=10
#datasource.authAlias=jaas.alias
datasource.helperClassname=com.ibm.websphere.rsadapter.DB2UniversalDataStoreHelper
datasource.dbType=DB2
datasource.databaseName=REPORTIN
datasource.portNumber=50000
datasource.driverType=4

activationSpec.name=IAPActivationSpec
#activationSpec.description=
activationSpec.jndiName=jms/IAPActivationSpec
activationSpec.destinationJndiName=jms/ApolloEventTopic
activationSpec.destinationType=javax.jms.Topic
activationSpec.queueManager.name=IAPQM
```

```
activationSpec.queueManager.portNumber=1414
activationSpec.clientId=APOLLO_JMS_CLIENTID

topic.name=ApolloEventTopic
topic.jndiName=jms/ApolloEventTopic
topic.topicName=APOLLO_EVENT_TOPIC
#topic.brokerPublicationQueueManagerName=activationSpec.queueManager.name
```

The **install.bat** installation script can be passed many parameters, enabling it to perform multiple installation tasks. Table 3-5 outlines the main parameters and their functions.

Table 3-5   Parameters for the install.bat script

Parameter	Use
stopwas	Stops the ApolloRead WebSphere Application Server profile.
startwas	Starts the ApolloRead WebSphere Application Server profile.
restartwas	Stops and then starts the ApolloRead WebSphere Application Server profile.
database or db	Creates an i2 Intelligence Analysis Platform Reporting database on the DB2 server, specified by the **datasource.serverName** parameter in the install.properties file. If an i2 Intelligence Analysis Platform Reporting database exists on the DB2 server, it is deleted.
jaas	Creates the Java Authentication and Authorization Service (JAAS) Authentication Alias on the WebSphere Application Server for the i2 Intelligence Analysis Platform Reporting database.
jdbcprovider or jdbc	Creates the Java Database Connectivity (JDBC) provider that is used to connect to the i2 Intelligence Analysis Platform Reporting database.
datasource or ds	Configures the WebSphere data source for the i2 Intelligence Analysis Platform Reporting database.
activationspec or actspec	Creates the activation specification for the i2 Intelligence Analysis Platform WebSphere MQ messaging provider.
topic	Creates the Java Message Service (JMS) topic destination for the i2 Intelligence Analysis Platform WebSphere MQ messaging provider.
deploy	Deploys the i2 Intelligence Analysis Platform Reporting .ear file on the i2 Intelligence Analysis Platform read server.
uninstall	Uninstalls the i2 Intelligence Analysis Platform Reporting .ear file on the i2 Intelligence Analysis Platform read server.

Parameter	Use
all	The all parameter is equivalent to running the batch file multiple times with the following parameters: install stopwas install database install startwas install jaas install datasource install activationspec install uninstall install deploy install restartwas
debug	Enables the display of additional debug messages during the installation.

Note: Parameters are case sensitive and must not include a leading dash, for example, **install jaas** and *not* **install -jaas**.

2. Open a command prompt and navigate to the \REPORTING_INSTALL\IAP_Reporting_Installer directory. Run the following commands:

install stopwas

install database

install startwas

install jaas

install datasource

install restartwas

Ensure that the WebSphere Application Server is started on both the read and write servers of the i2 Intelligence Analysis Platform. Check that the i2 Intelligence Analysis Platform MQ queue manager is started on the write server before proceeding.

3. Set up the IBM Cognos BI reports national language support for i2 Intelligence Analysis Platform by way of string translation tables using the provided scripts. These tables are stored in the i2 Intelligence Analysis Platform reporting database.

Note: Creation of these tables is mandatory, even if only one language report is used, for example, English.

Follow these steps to create and populate the tables:

a. Open a command prompt and navigate to the directory:

\Reporting_Install\Cognos\translations\02-crime_types_helper_files\for_inserting_real_data\sql

b. Run the following command:

populate *<db_name> <db_port> <db_host> <db_user> <db_password>*

The variables are defined:

db_name is reportin.

db_port is any numeric value. This parameter is not currently used but a value must be specified.

> *db_host* is the *hostname* of the DB2 server. This parameter is not currently used but a value must be specified.
>
> *db_user* is the i2 Intelligence Analysis Platform DB2 user.
>
> *db_ password* is the password of the i2 Intelligence Analysis Platform DB2 user.
>
> The command might look like this example:

```
populate reportin 50000 localhost myIAPDB2user myIAPDB2password
```

Install the reporting service

The i2 Intelligent Law Enforcement reporting uses a new reporting service that has been added to the i2 Intelligence Analysis Platform read server. The reporting service is not present in the standard i2 Intelligence Analysis Platform version. To install this service, the i2 Intelligence Analysis Platform deployment kit must be available on the i2 Intelligence Analysis Platform read server.

Perform the following steps:

1. Open a command prompt and navigate to the `IAP-Deployment-Toolkit\Configuration\deployment-scripts` directory. Configure the `environment.bat` file that is included with the i2 Intelligence Analysis Platform Deployment Toolkit, by following the instructions provided in the document, *IBM i2 Intelligence Analysis Platform Deployment Guide,* SC27-5091-00, at this website:

   http://www-01.ibm.com/support/docview.wss?uid=pub1sc27509100

2. Open a command window and execute **environment.bat**.

3. In the same command window, run the following command:

   ```
   manage-ear.py read unpack
   ```

   > **Important:** This command creates the required directory structure. Ensure that a backup copy of the original `iap-release-3.0.3.1-read.ear` file exists before the appropriate language version of the `iap-release-3.0.3.1-read.ear` file is copied from the `IAP_Reporting_Install\IAP_Reporting_Installer\ear` directory to the `IAP-Deployment-Toolkit\Configuration\unpack\Read` directory.

4. Replace the standard i2 Intelligence Analysis Platform application that is installed in the IBM WebSphere Application Server with the modified version, which contains the reporting service. Run the following commands:

   ```
   manage-application.py read uninstall
   ```

   ```
   manage-application.py read install
   ```

In a similar manner, a matching `iap-release-3.0.3.1-write.ear` file must be deployed on the i2 Intelligence Analysis Platform write server. The i2 Intelligence Analysis Platform deployment kit must be available on the i2 Intelligence Analysis Platform write server.

Perform the following steps:

1. Open a command prompt and navigate to the `IAP-Deployment-Toolkit\Configuration\deployment-scripts` directory.

2. Configure the `environment.bat`, which is included with the i2 Intelligence Analysis Platform Deployment Toolkit, by following the instructions provided in the document *IBM i2 Intelligence Analysis Platform Deployment Guide,* SC27-5091-00, at this website:

   http://www-01.ibm.com/support/docview.wss?uid=pub1sc27509100

3. Open a command window and run `environment.bat`. In the same command window, run the following command:

`manage-ear.py read unpack`

> **Important:** Ensure that a backup copy of the original `iap-release-3.0.3.1-write.ear` file exists before you copy the appropriate language version of the `iap-release-3.0.3.1-write.ear` file from the `IAP_Reporting_Install\IAP_Reporting_Installer\ear` directory to the `IAP-Deployment-Toolkit\Configuration\unpack\Read` directory.

4. Replace the standard i2 Intelligence Analysis Platform application that is installed on the WebSphere Application Server with the new version by running the following commands:

`manage-application.py write uninstall`

`manage-application.py write install`

5. Start the i2 Intelligence Analysis Platform application on both the i2 Intelligence Analysis Platform read and write servers from the WebSphere Application Server administrative consoles on the servers.

The setup procedures are complete if you are using IBM DB2 as your database. The next two sections cover two special cases that apply in the following circumstances:

► i2 Intelligence Analysis Platform is running on Linux (see "Considerations for i2 Intelligence Analysis Platform on Linux" on page 70).

► You are using either MS SQL Server or Oracle Database for your database (see "Considerations when using MS SQL Server or Oracle Database" on page 73).

Considerations for i2 Intelligence Analysis Platform on Linux

The installation script included in the `Reporting_Install.zip` file is a batch file intended for use on the Windows platform. Although no equivalent Linux script is provided, the i2 Intelligence Analysis Platform reporting capability can easily be deployed in environments where i2 Intelligence Analysis Platform is deployed on Linux.

> **Note:** Before you proceed, copy the `Reporting_Install.zip` file to a temporary directory on the i2 Intelligence Analysis Platform read server and extract the contents. After a successful extraction, there is a directory named `/var/REPORTING_INSTALL` with the structure shown in Example 3-5 on page 70.

Example 3-5   REPORTING_INSTALL directory structure and contents

```
\REPORTING_INSTALL
+---Cognos
|   |   ILE_FM_project.zip
|   |   ile_model.zip
|   +---translations
+---Coplink_Reporting_Installer
+---IAP_Reporting_Installer
|   |   .pydevproject
|   |   install.bat
|   |   install.properties
|   +---db
|   |       create_schema.sql
|   |       create_views.sql
|   |       drop_views.sql
```

```
|     |            insert_metadata.sql
|  +---ear
|  |    +---en
|  |    |        iap-release-3.0.3.1-read.ear
|  |    |        iap-release-3.0.3.1-write.ear
|  |    +---es
|  |    +---pt_br
|  |    +---zh_cn
|  |    +---zh_tw
|  \---was
|            deploy_ear.py
|            setup_activation_spec.py
|            setup_datasource.py
|            setup_jaas.py
|            setup_jdbcprovider.py
|            setup_topic.py
|            start_application.py
|            stop_server.py
|            uninstall_ear.py
|            wsadminlib.py
+---ILE_Console_Pages
```

The Windows batch file uses the parameters set in the `install.properties` file to build commands that invoke the python scripts in the `REPORTING_INSTALL/was` directory. On Linux, you can manually call the python scripts using similar commands. The installation steps are similar:

1. Configure parameters in `install.properties`.

2. Stop WebSphere Application Server on the read server.

3. Create the i2 Intelligent Analysis Platform Reporting database.

4. Start WebSphere Application Server on the read server.

5. Configure JAAS.

6. Configure the data source.

7. Restart WebSphere Application Server on the read server.

8. Create and populate the string translation tables.

9. Replace the enterprise archive (EAR) file on the read server.

10. Replace the EAR file on the write server.

11. Start the application on both the read and write servers.

The configuration of the `install.properties` file for Linux is similar to the configuration for Windows, except that the path specified in the **ear.fileLocation** parameter must be in Linux format. So, use /var/REPORTING_INSTALL/IAP_Reporting_Installer/ear/en instead of C:\Reporting_Install\IAP_Reporting_Installer\ear\en.

The following manual commands are used to perform these tasks, assuming that WebSphere Application Server is installed in the /opt/IBM/WebSphere directory and the Reporting installer is extracted to the /var/REPORTING_INSTALL directory:

1. Stop the WebSphere Application Server on the read server:

   /opt/IBM/WebSphere/AppServer/bin/wsadmin.sh -lang jython -f
   /var/REPORTING_INSTALL/IAP_Reporting_Installer/was/stop_server.py -p
   install.properties -profileName ApolloRead

2. Create the i2 Intelligence Analysis Platform Reporting database:

   a. Log in to the i2 Intelligence Analysis Platform read server as the i2 Intelligence Analysis Platform database user.

   b. Navigate to the `/home/db2inst1/sqllib/bin` directory.

   c. Run the following commands:

      db2 CREATE DATABASE REPORTIN AUTOMATIC STORAGE YES ON /home/db2inst1 USING CODESET UTF-8 TERRITORY US COLLATE USING SYSTEM PAGESIZE 32768

      db2 CONNECT REPORTIN

      db2 GRANT DBADM,CREATETAB,BINDADD,CONNECT,CREATE_NOT_FENCED_ROUTINE,IMPLICIT_SCHEMA,LOAD,CREATE_EXTERNAL_ROUTINE,QUIESCE_CONNECT,SECADM ON DATABASE TO USER db2inst1

      db2 DISCONNECT REPORTIN

3. Edit the `create_schema.sql` file, the `create_views.sql` file, and the `insert_metadata.sql` script to add a connect statement to the top of each of the files and a disconnect statement to the bottom of each of the files:

   connect to reportin user db2inst1 using password;

   ...

   ...

   ...

   disconnect reportin;

4. Open a command prompt and navigate to the `/REPORTING_INSTALL/IAP_Reporting_Installer` directory. Run the following commands:

   db2 -tsvf db\create_schema.sql

   db2 -tsvf db\create_views.sql

   db2 -tsvf db\ insert_metadata.sql

   Start WebSphere Application Server on the read server:

   /opt/IBM/WebSphere/AppServer/profiles/ApolloRead/bin/startServer.sh ApolloServerRead

5. Configure JAAS:

   /opt/IBM/WebSphere/AppServer/bin/wsadmin.sh -lang jython -f /var/REPORTING_INSTALL/IAP_Reporting_Installer/was/setup_jaas.py -p install.properties -profileName ApolloRead

6. Configure the data source:

   /opt/IBM/WebSphere/AppServer/bin/wsadmin.sh -lang jython -f var/REPORTING_INSTALL/IAP_Reporting_Installer/was/setup_datasource.py -p install.properties -profileName ApolloRead

7. Restart WebSphere Application Server on the read server:

   /opt/IBM/WebSphere/AppServer/bin/wsadmin.sh -lang jython -f /var/REPORTING_INSTALL/IAP_Reporting_Installer/was/stop_server.py -p install.properties -profileName ApolloRead

   /opt/IBM/WebSphere/AppServer/profiles/ApolloRead/bin/startServer.sh ApolloServerRead

8. The next step is to populate the string translation tables in the i2 Intelligence Analysis Platform Reporting database to provide national language support. A Windows batch file is provided to perform this task. This batch file invokes two SQL scripts. These scripts can be run manually on Linux to achieve the same results:

   a. In a Linux command window, navigate to the following directory:

      ```
      /var/REPORTING_INSTALL/IAP_Reorting_Installer/Cognos/translations
      /02-crime_types/_helper_files/for_inserting_real_data/sql
      ```

   b. Edit the `create_schema.sql` file and `create_views.sql` file and the `insert_metadata.sql` script to add a connect statement to the top of each of the files and a disconnect statement to the bottom of each of the files:

      `connect to reportin user myIAPDBuser using myIAPDBpassword;`

      `...`

      `...`

      `...`

      `disconnect reportin;`

9. Run the following scripts from a DB2 command window:

   `Create_tables_ddl.sql`

   `populate_tables.sql`

10. The remainder of the installation on Linux is the same as the remainder of the installation on Windows:

    a. Deploy the modified EAR file on the read server.

    b. Deploy the modified EAR file on the write server.

    c. Start both applications.

The *IBM i2 Intelligence Analysis Platform Linux Deployment Guide* at this website provides information about how to use the i2 Intelligence Analysis Platform Deployment Toolkit on Linux:

http://www-01.ibm.com/support/docview.wss?uid=swg27041251

Considerations when using MS SQL Server or Oracle Database

Although i2 Intelligence Analysis Platform supports the use of Microsoft SQL Server and Oracle Database for the i2 Intelligence Analysis Platform write and read databases, the i2 Intelligent Law Enforcement Reporting database is only supported on IBM DB2.

Environments where the i2 Intelligence Analysis Platform databases are not on DB2 present some additional challenges:

► It is unlikely that DB2 will be installed on the i2 Intelligence Analysis Platform read server; therefore, the i2 Intelligent Law Enforcement Reporting database must be hosted on a different server.

► If DB2 is not installed on the i2 Intelligence Analysis Platform read server, the DB2 JDBC drivers will not be present; therefore, the creation of the data source on the WebSphere Application Server will fail.

Host the reporting database on IBM Intelligent Operations Center

It is possible to host the i2 Intelligence Analysis Platform Reporting database on the IBM Intelligent Operations Center database server. This approach avoids the need to install a separate DB2 server to host it. The database server in IBM Intelligent Operations Center V1.5 already has eight DB2 instances (db2inst1 through db2inst8) that were created as part of the standard installation. It is best to create a new instance (db2inst9) to host the i2 Intelligent Law Enforcement Reporting database.

Create the DB2 instance owner. In IBM Intelligent Operations Center V1.5, in addition to being members of the OS system group `dasadm1`, all the DB2 instance owners have their own Instance owner group, which, in this case, will be `db2iadm9`, to which the instance owner and the `root` user are added. Each instance also has a fenced user, in this case, `db2fenc9`, with its own group, `db2fadm9`. For information about DB2 users and groups, including fenced users, see the IBM Knowledge Center topic *DB2 users and groups* at this website:

http://www-01.ibm.com/support/knowledgecenter/SSEPGG_9.7.0/com.ibm.db2.luw.qb.server.doc/doc/c0011931.html?lang=en

The following commands create and configure these users and groups:

groupadd -g 519 db2iadm9

groupadd -g 520 db2fadm9

useradd -u 1028 -g 519 -G 102 -d /datahome/db2inst9 -m -c "ILE Reporting DB Instance Owner" db2inst9

useradd -u 1029 -g 520 -d /home/db2fenc9 -m -c "ILE Reporting DB fenced user" db2fenc9

usermod -G 519 root

passwd db2inst9

passwd db2fenc9

The DB2 instance can now be created. Navigate to the `/opt/IBM/DB2/instance` directory and use the **db2icrt** command to create the new instance:

./db2icrt -u db2fenc9 db2inst9

To start the new instance, log in as the instance owner and issue the **db2start** command.

Create the reporting database

After a new instance is created on the IBM Intelligent Operations Center database server, the i2 Intelligence Analysis Platform reporting database can be created.

Log in to the IBM Intelligent Operations Center database server as the `db2inst9` user, navigate to the `/datahome/db2inst9/sqllib/bin` directory, and run the following commands:

./db2 CREATE DATABASE REPORTIN AUTOMATIC STORAGE YES ON /datahome/db2inst9 USING CODESET UTF-8 TERRITORY US COLLATE USING SYSTEM PAGESIZE 32768

./db2 CONNECT REPORTIN

./db2 GRANT DBADM,CREATETAB,BINDADD,CONNECT,CREATE_NOT_FENCED_ROUTINE,IMPLICIT_SCHEMA,LOAD,CREATE_EXTERNAL_ROUTINE,QUIESCE_CONNECT,SECADM ON DATABASE TO USER db2inst9

To add the database schema, views, metadata, and so on, to the i2 Intelligent Law Enforcement Reporting database, the IBM Data Server Client Package Version 9.7 Fix Pack 5 must be installed on the i2 Intelligence Analysis Platform read server so that local node and database aliases pointing to the database on the IBM Intelligent Operations Center database server can be defined.

For information about downloading the IBM Data Server Client Packages from Fix Central, see the IBM technote *IBM Data Server Client Packages Version 9.7 Fix Pack 5* at this website:

http://www-01.ibm.com/support/docview.wss?uid=swg24031182

Copy `db2jcc4.jar` and `db2jcc_license_cu.jar` from the `\java` directory under the DB2 client install directory to a directory named `C:\IBM\WebSphere\AppServer\profiles\ApolloRead\JdbcDrivers`.

Create local node and database aliases

This section describes how to create a local node and database aliases.

To create a local node, perform the following steps:

1. Open a command prompt on the i2 Intelligence Analysis Platform read server, and navigate to the `\BIN` directory in the DB2 client installation directory, for example, `C:\Program Files\IBM\SQLLIB\BIN`.

   Note: If your i2 Intelligence Analysis Platform is running on Linux, replace the Windows path for db2 with the Linux path. The **db2** commands are the same on both platforms.

2. Run the following commands:

   `./db2 catalog tcpip node ReportNode remote <hostname> server <port>`

   The variables are defined:

   – *<hostname>* is the Intelligent Operations Center database server host name.
   – *<port>* is the TCP/IP service port for the `db2inst9` instance on the IBM Intelligent Operations Center database server.

   The following example shows this command:

   `./db2 catalog db reportin as reportin at node ReportNode`

After a local node and database aliases are created, it is necessary to add the required schema, views, and metadata to the i2 Integrated Law Enforcement reporting database. SQL scripts are provided in the IBM i2 Intelligent Law Enforcement V1.0.1 Multilingual Media Pack (BF079ML) to assist with this task.

The core component distribution media contained in the IBM i2 Intelligent Law Enforcement V1.0.1 Multilingual Media Pack (BF079ML) is a `.zip` archive named `I2_ILE_V1.0.1_WIN_ML.zip`, which also contains two archives: `Reporting_Install.zip` and `Sitaw_Install.zip`. Copy the `Reporting_Install.zip` file to a temporary directory on the i2 Intelligence Analysis Platform read server and extract the contents, for example, to `c:\temp` or `/var`.

Assuming `Reporting_Install.zip` is extracted to `c:\temp\REPORTING_INSTALL`, the SQL scripts for adding the required schema, views, and metadata to the reporting database are in the `c:\temp\REPORTING_INSTALL\IAP_Reporting_Installer\db` directory.

To run the SQL scripts, perform the following steps:

1. Edit the `create_schema.sql`, `create_views.sql`, and `insert_metadata.sql` scripts to add a connect statement to the top of each file and a disconnect statement to the bottom of each file:

   connect to reportin user db2inst9 using password;

   ...

   ...

   ...

   disconnect reportin;

2. Open a command prompt, and navigate to the `\REPORTING_INSTALL\IAP_Reporting_Installer` directory. Run the following commands:

   install stopwas

   db2cmd -w -c db2 -tsvf db\create_schema.sql

   db2cmd -w -c db2 -tsvf db\create_views.sql

   db2cmd -w -c db2 -tsvf db\ insert_metadata.sql

   install startwas

   install jaas

   install datasource

   install restartwas

   > **Note:** Before you proceed, ensure that the WebSphere Application Server is started on both the i2 Intelligence Analysis Platform read and write servers and that the i2 Intelligence Analysis Platform MQ queue manager is started on the write server.

3. To create and populate the string translation tables required by the IBM Cognos BI reports of i2 Intelligence Analysis Platform reporting, navigate to the `\Reporting_Install\Cognos\translations\02-crime_types\_helper_files\for_inserting_real_data\sql` directory.

4. Edit the **create_tables_ddl.sql** and **populate_views.sql** scripts to add a connect statement to the top of each file and a disconnect statement to the bottom of each file:

   connect to reportin user db2inst9 using password;

   ...

   ...

   ...

   disconnect reportin;

5. Open a command prompt. Navigate to the `\Reporting_Install\Cognos\translations\02-crime_types\_helper_files\for_inserting_real_data\sql` directory. Run the following commands:

   db2cmd -w -c db2 -tsvf db\create_tables_ddl.sql

   db2cmd -w -c db2 -tsvf db\populate_views.sql

The remaining installation steps, which relate to the deployment of the modified i2 Intelligence Analysis Platform Java Platform, Enterprise Edition applications on the i2 Intelligence Analysis Platform read and write servers, are the same as the steps for an environment where i2 Intelligence Analysis Platform uses IBM DB2.

3.3.2 Setting up i2 COPLINK Reporting

i2 COPLINK Reporting is an extended capability that involves both i2 COPLINK and IBM Intelligent Operations Center. Figure 3-10 shows the components that are involved and how they interact. Reports that are displayed on the i2 Intelligent Law Enforcement console are generated using the IBM Cognos Business Intelligence server included with IBM Intelligent Operations Center by running SQL queries against special views that are added to the i2 COPLINK database.
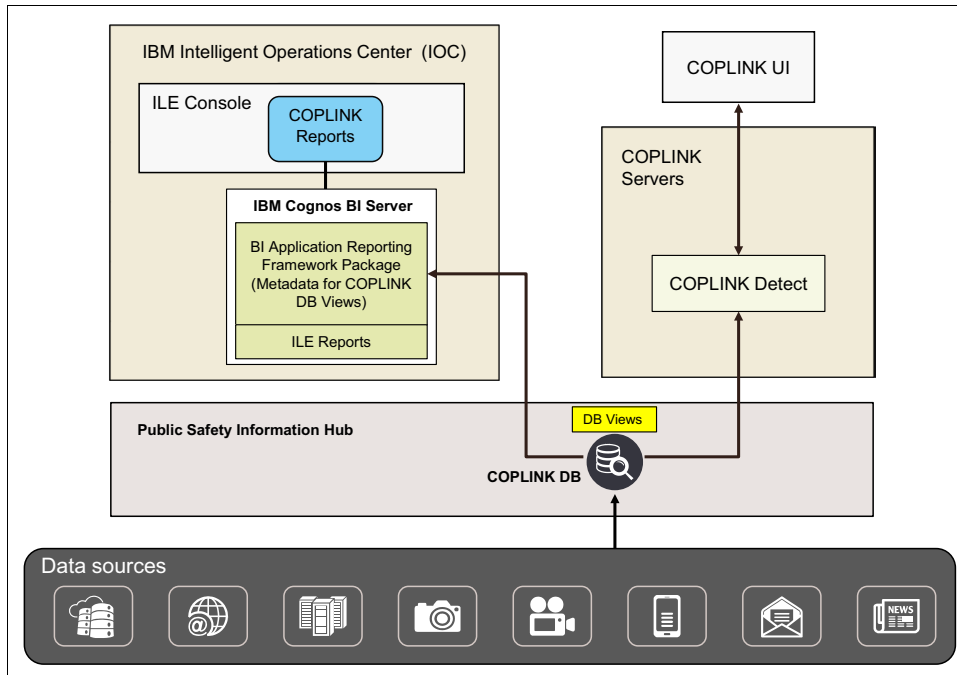
Figure 3-10   Components of i2 COPLINK report processing

Prepare for installing the i2 COPLINK reporting views

The core component distribution media contained in the IBM i2 Intelligent Law Enforcement V1.0.1 Multilingual Media Pack (BF079ML) is a `.zip` archive named `I2_ILE_V1.0.1_WIN_ML.zip`, which also contains two archives: `Reporting_Install.zip` and `Sitaw_Install.zip`. Copy the `Reporting_Install.zip` file to a temporary directory on the i2 Intelligence Analysis Platform read server and extract the contents, for example, to `c:\temp`.

To install i2 COPLINK Reporting, copy the `Reporting_Install.zip` to a temporary directory on the i2 COPLINK application server, and extract the contents.

The directory structure of the extracted contents has a directory named `REPORTING_INSTALL` with the contents that are shown in Example 3-6.

Example 3-6   REPORTING_INSTALL directory structure and contents

```
\REPORTING_INSTALL
+---Cognos
|
+---Coplink_Reporting_Installer
|    +---Views_Oracle
|    |        Cognos Reporting Views Install Script - Oracle.sql
|    |
|    \---Views_SQL_Server
```

```
|            Cognos Reporting Views Install Script - SQL Server.sql
|
+---IAP_Reporting_Installer
|
\---ILE_Console_Pages
```

Deploy reporting views on Oracle database

An SQL script named `Cognos Reporting Views Install Script - Oracle.sql`, which adds the necessary reporting views to the i2 COPLINK database, is provided in the `REPORTING_INSTALL\Coplink_Reporting_Installer\Views_Oracle` directory.

This section describes the use of IBM Data Studio to run the script `Cognos Reporting Views Install Script - Oracle.sql`. IBM Data Studio is available for download from this website:

http://www.ibm.com/developerworks/downloads/im/data/

Native Oracle tools can also be used. For instructions to use these tools, see the *Oracle Database Documentation Library* at this website:

http://docs.oracle.com/cd/E11882_01/index.htm

Perform the following steps:

1. Open IBM Data Studio. In Administration Explorer, click the **New Connection to a database** icon, as shown in Figure 3-11.
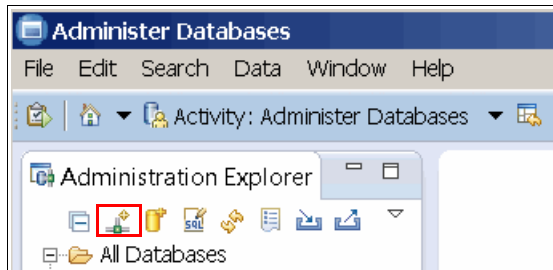
Figure 3-11   IBM Data Studio - New Connection to database

2. Set the JDBC driver to **Oracle 11 - Oracle thin Driver Default**.

3. Enter the System Identifier (SID), Host, User name, and Password for the i2 COPLINK database. An example of the required settings is shown in Figure 3-12 on page 79.
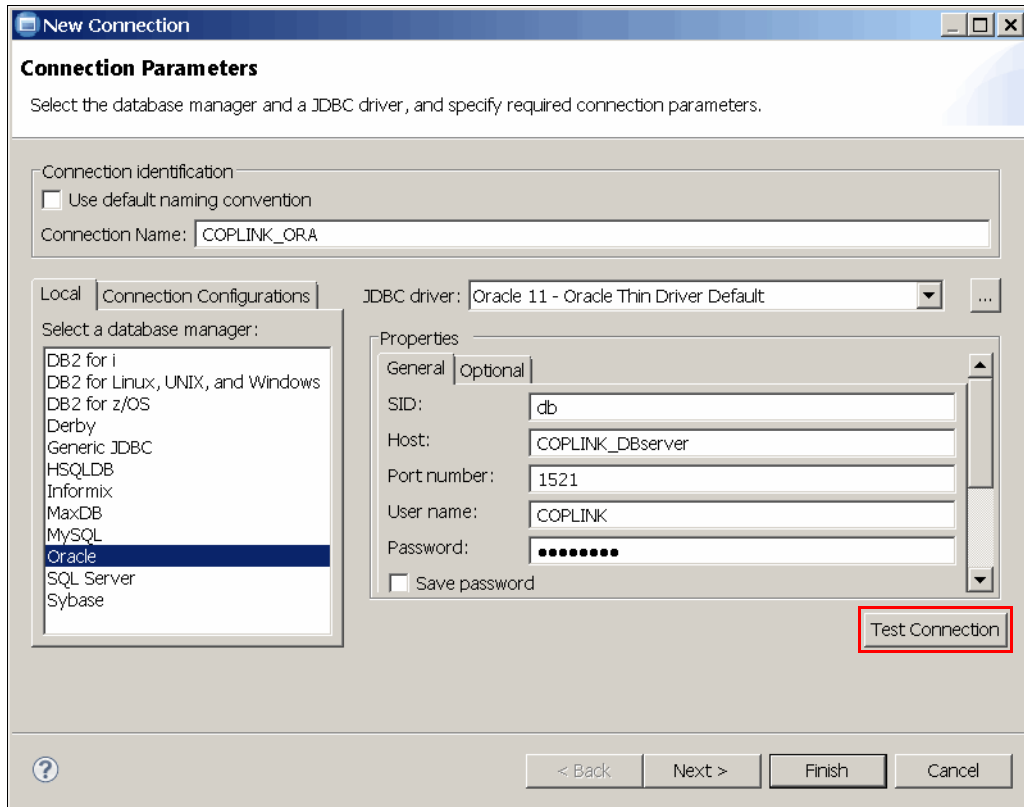
Figure 3-12   Connection parameters for an Oracle database in IBM Data Studio

4. Click **Test Connection** (see Figure 3-12) to ensure that the credentials are correct and then click **Finish**.

5. A message is displayed, as shown in Figure 3-13, that indicates that a change to the Data Source Explorer view in the Database Development perspective is required.
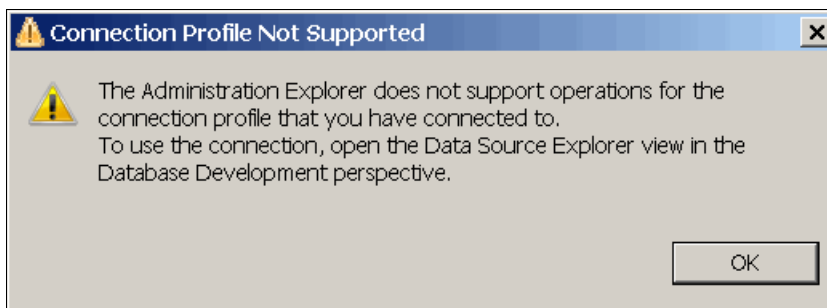
Figure 3-13   IBM Data Studio - Connection Profile Not Supported message

6. Click **Window** → **Open Perspective** → **Other** and select **Data** from the list. An example is shown in Figure 3-14 on page 80.
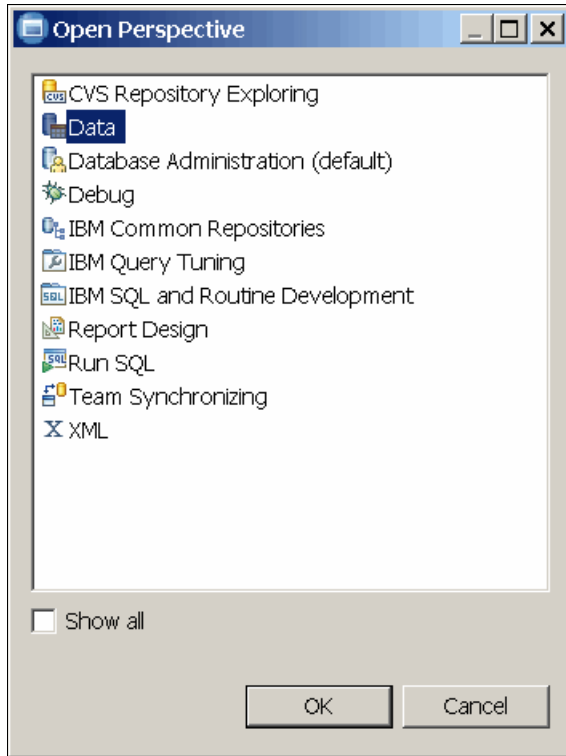
Figure 3-14   IBM Data Studio Open Perspective

7. Right-click the database connection to the i2 COPLINK database and select **New SQL Script** from the drop-down menu as shown in Figure 3-15 on page 81.
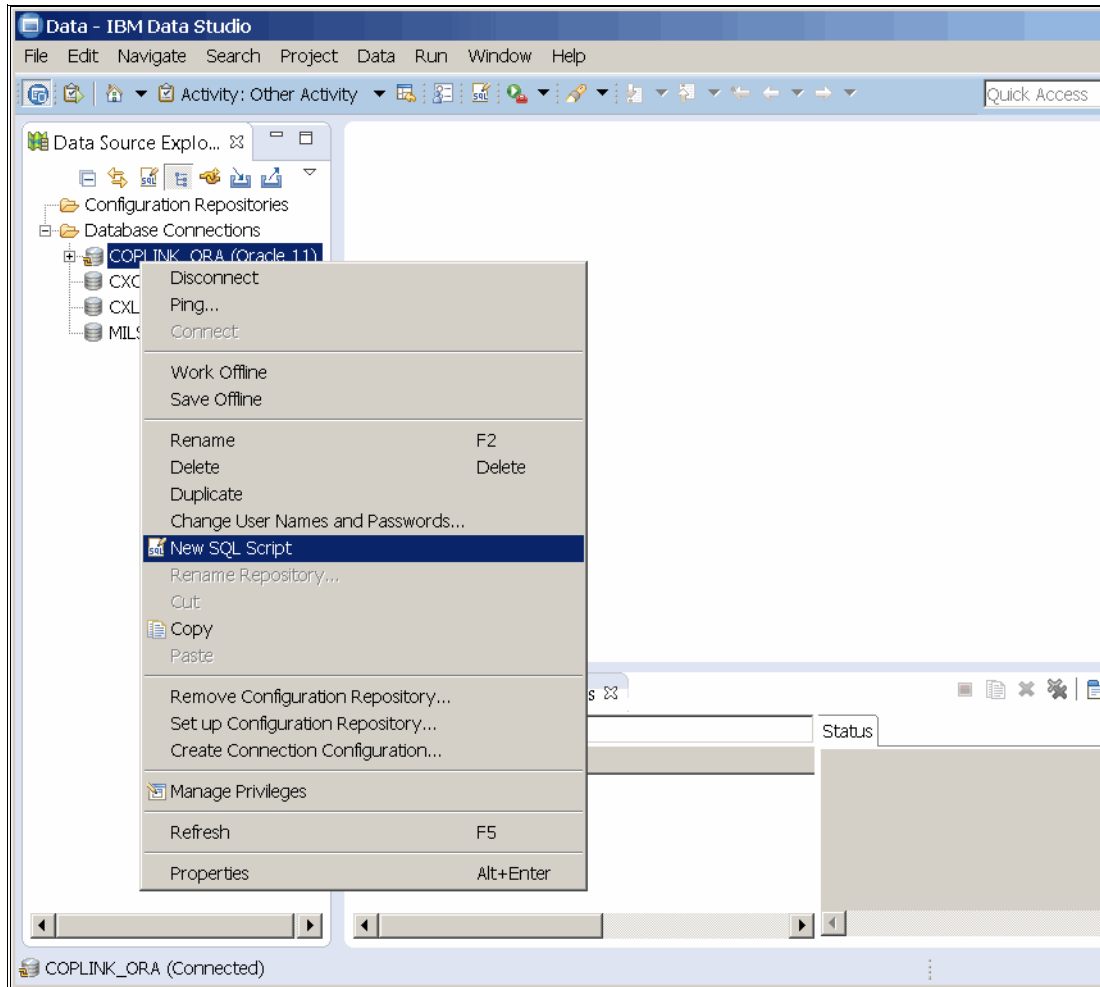
Figure 3-15   IBM Data Studio New SQL Script

8. Open the **Cognos Reporting Views Install Script - Oracle.sql** SQL script, which is in the `REPORT-ING_INSTALL\Coplink_Reporting_Installer\Views_Oracle` directory, in a text editor and copy the content to the SQL Editor Window. The result is shown in Figure 3-16 on page 82.
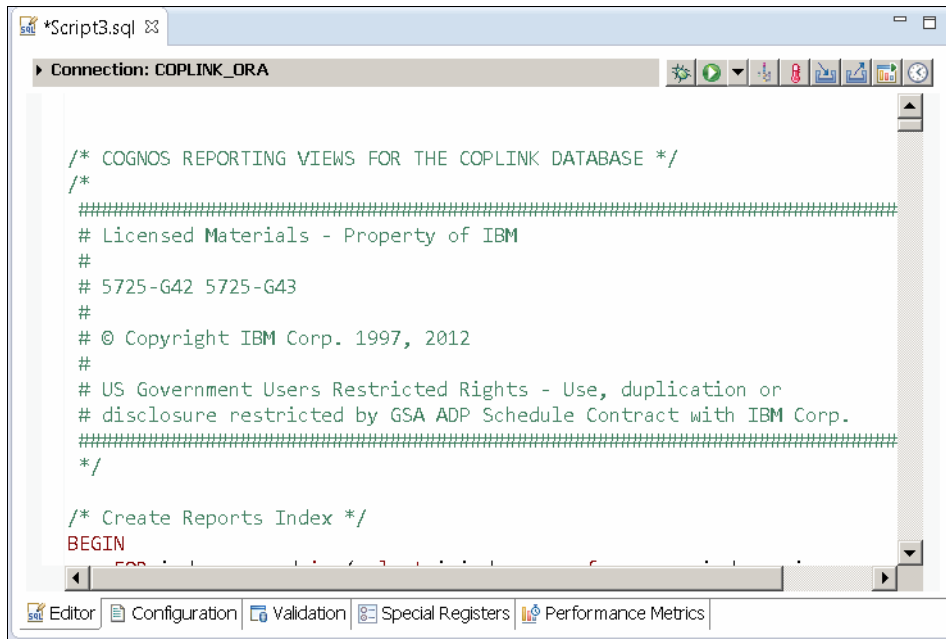
Figure 3-16   IBM Data Studio SQL script

9. Accept the semicolon as the statement terminator.

10. Click the **Run SQL** icon at the top of the Editor window to run the script.

11. Check the SQL Results window to ensure that no errors occurred.

Deploy reporting views on the Microsoft SQL Server database

The SQL script `Cognos Reporting Views Install Script - SQL Server.sql` adds the necessary reporting views to the i2 COPLINK database. It is provided in the `REPORTING_INSTALL\Coplink_Reporting_Installer\Views_SQL_Server` directory.

This section describes the use of IBM Data Studio to run this script. IBM Data Studio is available for download from `http://www.ibm.com/developerworks/downloads/im/data/`.

Native Microsoft SQL Server tools can also be used. For information about these tools and instructions, see the Microsoft SQL Server documentation at this website:

`http://msdn.microsoft.com/en-us/library/bb545450.aspx`

Perform the following steps:

1. Open **IBM Data Studio**.

2. In the Administration Explorer window, click the **New Connection to a database** icon as shown in Figure 3-17.
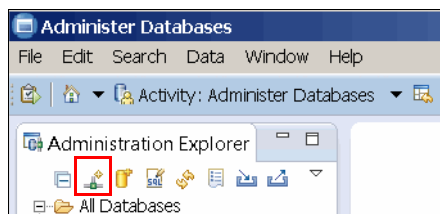
Figure 3-17   IBM Data Studio New Connection to a database

3. Set the JDBC driver to the correct driver for the version of Microsoft SQL Server in use.

4. Enter the Database name, Host, User name, and password for the i2 COPLINK database, as shown in Figure 3-18. If you need to download the Microsoft JDBC Driver for SQL Server server, it is available from the Microsoft Developer Network (MSDN) site:

   `http://msdn.microsoft.com/en-us/sqlserver/aa937724.aspx`
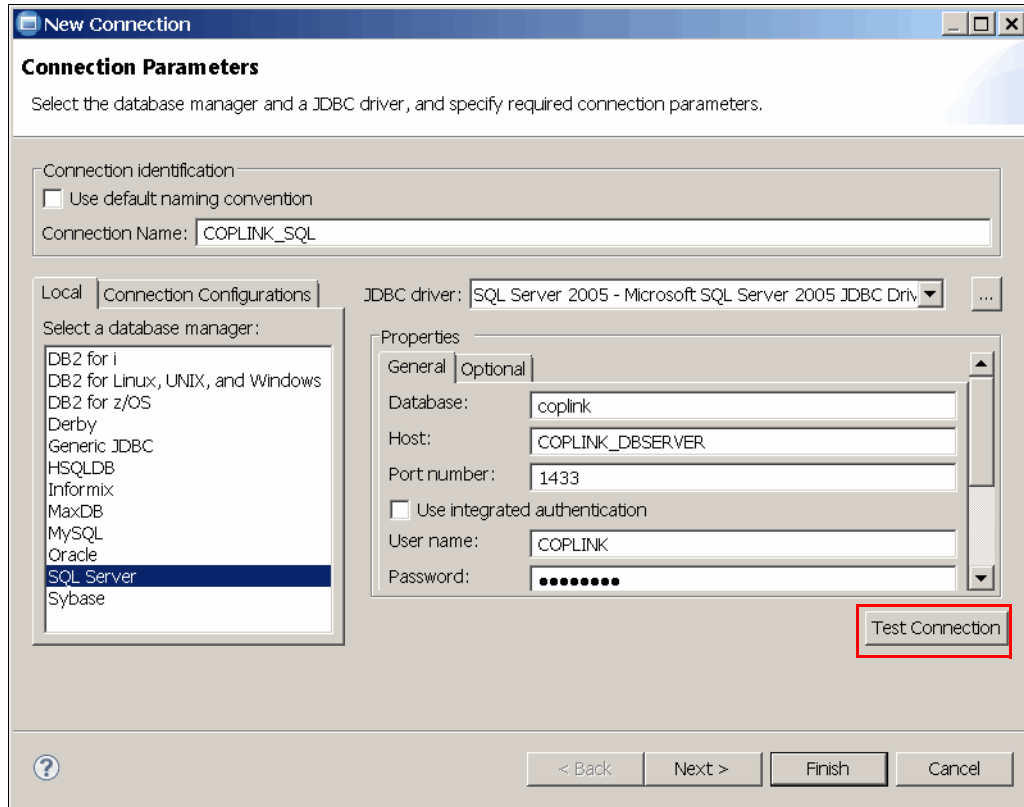
Figure 3-18   IBM Data Studio Connection Parameters for a Microsoft SQL database

5. Click **Test Connection** (see Figure 3-18) to ensure that the credentials are correct.

6. Click **Finish**.

   While saving the connection, a message is displayed (as shown in Figure 3-13 on page 79) that indicates that a change to the Data Source Explorer view in the Database Development perspective is required.

7. Click **Window** → **Open Perspective** → **Other** and select **Data** from the list as shown in Figure 3-19 on page 84.
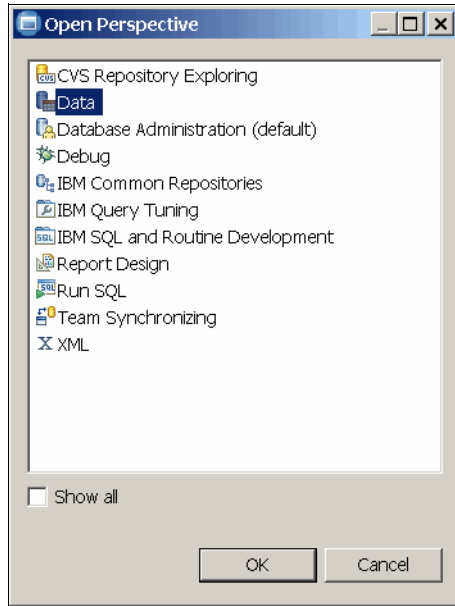
Figure 3-19   IBM Data Studio Open Perspective

8. Right-click the database connection to the i2 COPLINK database and select **New SQL Script** from the drop-down menu as shown in Figure 3-20 on page 85.
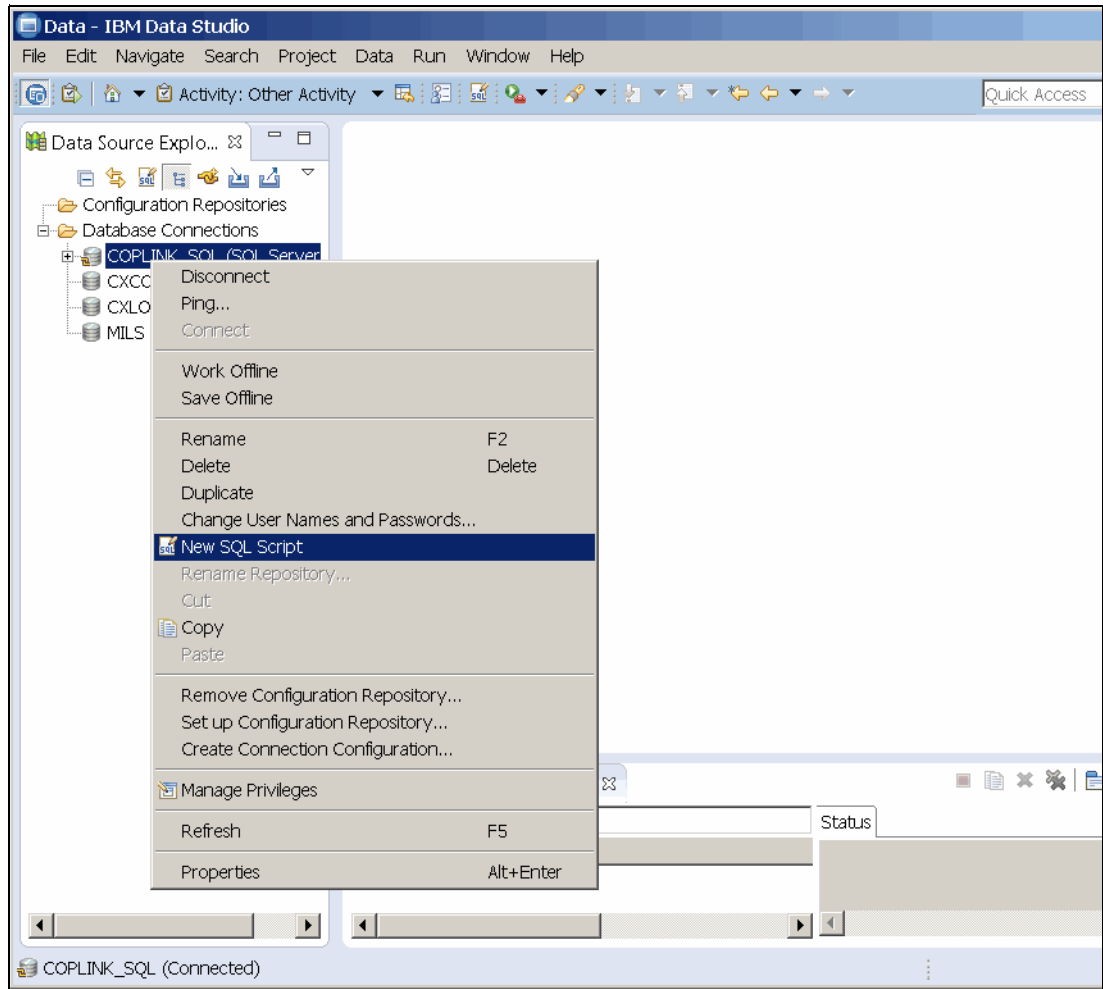
Figure 3-20   IBM Data Studio New SQL Script

9. Open the **Cognos Reporting Views Install Script - SQL Server.sql** script in the `REPORTING_INSTALL\Coplink_Reporting_Installer\Views_SQL_Server` directory in a text editor.

10. Copy the content to the SQL Editor window and accept the semicolon as the statement terminator. The result is shown in Figure 3-21 on page 86.
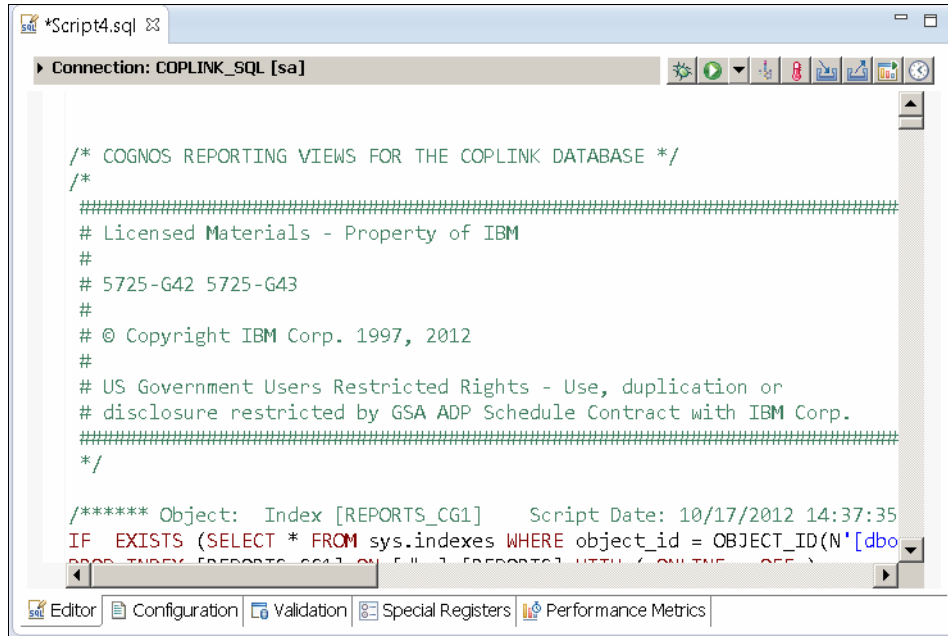
Figure 3-21   IBM Data Studio SQL script

11.Click the **Run SQL** icon at the top of the Editor window to run the script.

12.Check the SQL Results window to ensure that no errors occurred.

Deploy i2 COPLINK Reporting package

The Reporting extended capability of i2 Intelligent Law Enforcement V1.0.1 uses the IBM Cognos Business Intelligence server, which runs in the IBM Intelligent Operations Center application server.

The core component distribution media contained in the IBM i2 Intelligent Law Enforcement V1.0.1 Multilingual Media Pack (BF079ML) is a `.zip` archive named `I2_ILE_V1.0.1_WIN_ML.zip`, which also contains two archives: `Reporting_Install.zip` and `Sitaw_Install.zip`.

To install i2 COPLINK Reporting, copy the `Reporting_Install.zip` file to a temporary directory on the i2 COPLINK application server and extract the contents.

The directory structure of the extracted contents has a directory named `REPORTING_INSTALL` with the contents shown in Example 3-7.

Example 3-7   REPORTING_INSTALL directory structure and contents

```
\REPORTING_INSTALL
+---Cognos
|   |   ILE_FM_project.zip
|   |   ile_model.zip
|   +---translations
+---Coplink_Reporting_Installer
+---IAP_Reporting_Installer
+---ILE_Console_Pages
```

The `\REPORTING_INSTALL\Cognos` directory contains two files, `ILE_FM_project.zip` and `ile_model.zip`, and a subdirectory named `Translations`. The `Translations` directory contains the string translation tables that provide national language support:

► `ILE_FM_project.zip` is the Cognos Framework Manager project for the i2 Integrated Law Enforcement reports. It allows Cognos developers to modify and extend the reports.

► `ile_model.zip` is the Cognos reports model that contains the report templates that are deployed on the Cognos server.

Deploy database connectivity software

For Cognos to communicate with the i2 COPLINK database, the necessary database client software for the platform on which the i2 COPLINK database is hosted must be installed on the IBM Intelligent Operations Center application server.

The database client software has these options:

► i2 COPLINK database on Oracle Database: The 32-bit Oracle Database Client 11g Release 2 must be installed on the IBM Intelligent Operations Center application server.

> **Note:** Even though the IBM Intelligent Operations Center application server is a 64-bit system, the IBM Cognos Business Intelligence server only supports the 32-bit version of the Oracle client.

► i2 COPLINK database on Microsoft SQL Server: A third-party Open Database Connectivity (ODBC) driver, DataDirect Connect for ODBC, must be installed on the IBM Intelligent Operations Center application server. This ODBC driver is available for purchase from Progress Software at this website:

http://www.progress.com/

The use of DataDirect Connect for ODBC to connect to Microsoft SQL Server is a requirement of IBM Cognos BI running on Linux.

> **Important:** When choosing which version of DataDirect Connect for ODBC to recommend to your client, ensure that it is supported in RHEL 5 Server x86-64 update 5 or later and that it has been certified by IBM to work with IBM Cognos V10.1.1 (or later) and your client's version of MS SQL Server.

If your client does not want to purchase DataDirect Connect for ODBC, IBM provides a solution that involves migrating the IBM Intelligent Operations Center Cognos deployment from Linux to Windows. This solution is described in the IBM technote, *Alternative Cognos deployment configuration for IBM i2 Intelligent Law Enforcement,* at this website:

https://www-304.ibm.com/support/entdocview.wss?uid=swg24035906

Installing and setting up Oracle Database Client 11g Release 2

Follow these steps to install and set up the Oracle Database client:

1. Install the Oracle Database Client 11g Release 2:

   a. Download the Oracle Database Client 11g Release 2 32-bit installer for Linux from the Oracle site at http://www.oracle.com/.

   > **Note:** In some environments, the Oracle client drivers are available in both 64-bit and 32-bit versions. The *32-bit client* libraries are required for establishing a connection with IBM Cognos 10.

b. Install the Oracle Database client.

The complete instructions for installing the Oracle Database Client are in the *Oracle Database Client Quick Installation Guide 11g Release 2 (11.2) for Linux x86-64,* which is available at this website:

http://docs.oracle.com/cd/E11882_01/install.112/e24325.pdf

Follow the instructions in the document to install the administration client (option 2). An X Window System client is required to run the installer (`vnc`, for example).

2. Set up `ORACLE_HOME` to point to the install tree. If the client software was installed into the `/opt/app` directory, `ORACLE_HOME` is `/opt/app/product/11.2.0/client_1`.

   After `$ORACLE-HOME` is set, the tools can find the path to the `tnsnames.ora` file.

3. Run **netmgr** and configure the `tnsnames.ora` file to connect to the i2 COPLINK database.

4. Add the following lines to the `.bash_profile` files for the `oracle` and `ibmadmin` users:

```
PATH = /opt/app/oracle/product/11.2.0/client_1/bin/:$PATH
export PATH
ORACLE_HOME=/opt/app/oracle/product/11.2.0/client_1/
export ORACLE_HOME
ORACLE_BASE=/opt/app/oracle/
export ORACLE_BASE
TNS_ADMIN=/opt/app/oracle/product/11.2.0/client_1/admin/
export TNS_ADMIN
LD_LIBRARY_PATH=/opt/app/oracle/product/11.2.0/client_1/lib:$LD_LIBRARY_PATH
export LD_LIBRARY_PATH
```

5. Define WebSphere Application Server environment variables for `ORACLE_HOME`, `TNS_ADMIN`, and `PATH` in the Process Definition window for the Cognos server:

   a. Log in to the IBM Intelligent Operations Center portal server as `wpsadmin`.

   b. Select **Administration**.

   c. Select **Intelligent Operations** → **Administration Tools** → **Administration Consoles**.

   d. Select the application server.

   e. Log in to the WebSphere Application Server Integrated Solutions Console (default admin user is `waswebadmin`).

   f. Select **Servers** → **Server Types** → **WebSphere application servers**.

   g. Find and click **CognosX_Disp1**.

   h. Expand **Java** and **Process Management**, and click **Process Definition**.

   i. Click **Environment Entries**.

   j. Update the `LD_LIBRARY_PATH` entry to add the path to the Oracle libraries: `/opt/app/oracle/product/11.2.0/client_1/lib/`

   k. Add an entry for `ORACLE_HOME`:

      `/opt/app/oracle/product/11.2.0/client_1`

   l. Add an entry for `TNS_ADMIN`:

      `/opt/app/oracle/product/11.2.0/client_1/network/admin/`

   m. Select **Servers** → **Server Types** → **WebSphere application servers**.

   n. Find and click **CognosX_GW1**.

   o. Expand **Java** and **Process Management** and click **Process Definition**.

6. Click **Environment Entries**.

7. Update the `LD_LIBRARY_PATH` entry to add the path to the Oracle libraries:

   `/opt/app/oracle/product/11.2.0/client_1/lib/`

8. Add an entry for `ORACLE_HOME`:

   `/opt/app/oracle/product/11.2.0/client_1`

9. Add an entry for `TNS_ADMIN`:

   `/opt/app/oracle/product/11.2.0/client_1/network/admin/`

10. Log out of the WebSphere Application Server Integrated Solutions console.

Defining the Oracle system identifier (SID)

An Oracle service name must be defined in the Oracle client installed on the IBM Intelligent Operations Center application server.

Use the Oracle Net Manager utility to perform the following steps:

1. Navigate to the `ORACLE_HOME/bin` directory and run:

   `./netmgr`

2. In the Net Manager utility, click **Local** → **Service Naming** and click the green cross icon to add a system identifier (SID).

3. Provide values for the following prompts:

Net Service name	This name needs to be the same as the SID value on the i2 COPLINK database server.
Protocol	TCP/IP.
Host Name	The *hostname* of the i2 COPLINK database server.
Port Number	Usually `1521`.
Service Name	The SID value on the i2 COPLINK database server.
Connection Type	Database Default.

4. Test that the connection is configured correctly, click **Test Connection**.

5. If the connection is configured correctly, save it using the **Save Network Configuration** option in the File menu.

6. Restart the IBM Intelligent Operations Center application server for the changes to take effect.

> **Note:** Ensure that all of the IBM Intelligent Operations Center servers are correctly shut down by using the **`IOCControl.sh`** command on the IBM Intelligent Operations Center management server before you restart the application server.

Installing the Progress DataDirect for ODBC driver

To install the Progress DataDirect for ODBC driver, perform the following steps:

1. Obtain the DataDirect ODBC driver install archive from Progress Software as described in "Deploy database connectivity software" on page 87.

2. Expand the DataDirect install archive by using the Linux **`tar`** command:

   `tar -zxvf evlinux.tar.Z`

   The directory structure of the extracted contents has a directory named `evlinux` with the contents shown in Example 3-8 on page 90.

Example 3-8 evlinux directory structure and contents

```
/var/evlinux
|    autorun.dat
|    install.mi
|    unixmi.ksh
|
+---etc
|    +---lang
|    |        license.txt
|    |        msg.dat
|    |        useng.msg
|    |
|    /---lic
|             makelica
|             makelich
|             makelichx
|             makelicl
|             makelics
|             makelicsx
|
/---odbc
     /---linux
         |    ICULicense.txt
         |    mysqllicense.txt
         |    ODBCFIXES.TXT
         |    ODBCREADME.TXT
         |    scr1
         |
         /---tarfiles
                  dbase.tar
                  drda.tar
                  greenplum.tar
                  infcl.tar
                  linux.tar
                  mssql.tar
                  mysql.tar
                  oracle.tar
                  oraclewp.tar
                  postgresql.tar
                  sqlserverwp.tar
                  sybase.tar
                  teradata.tar
                  text.tar
```

Install the driver using the `unixmi.ksh` file. This file is a Korn shell script that invokes a text-based installer. Perform the following steps:

a. Run the command **ksh unixmi.ksh**.

b. In the installer, click **Y** to accept the current environment.

c. Review the license agreement and enter `YES` to accept it.

d. Enter the following registration information:

Name	*<Installer's name>*
Company	*<Company>*
Serial Number	*<Serial number provided by Progress Software when the driver was purchased>*
Key	*<Key provided by Progress Software when the driver was purchased>*

3. Select option **2** to install a single driver.

4. Select option **5** to install the Microsoft SQL Server Native Wire Protocol.

5. Press Enter to accept the default path to the temporary directory `/tmp`.

6. Press Enter to accept the default path to the installation directory `/opt/ConnectforODBC60`.

7. After the installation completes, enter `N` to exit the installation.

8. IBM Cognos Business Intelligence requires you to set several environment variables for the `ibmadmin` user on the IBM Intelligent Operations Center application server. Edit the `.bash_profile` file for the `ibmadmin` user, and add the following environment variables to the end of the file:

```
PATH=/opt/ConnectforODBC60/bin:$PATH:$HOME/bin
export PATH
LD_LIBRARY_PATH=/opt/ConnectforODBC60/lib:$LD_LIBRARY_PATH
export LD_LIBRARY_PATH
ODBCINST=/opt/ConnectforODBC60/odbcinst.ini
export ODBCINST
ODBCINI=/opt/ConnectforODBC60/odbc.ini
export ODBCINI
```

9. Navigate to the directory where the DataDirect driver was installed; the default location is `/opt/ConnectODBC60`.

10. Edit the `odbc.ini` file to set up a Data Source Name (DSN) to your i2 COPLINK database.

11. Replace the `[SQL Server Native Wire Protocol]` section in the sample `odbc.ini` file provided with the driver with the lines shown in Example 3-9. Set the correct values for the **Database** and **Hostname** parameters highlighted in bold.

Example 3-9   Replacing the [SQL Server Native Wire Protocol] section in the odbc.ini file

```
[COPLINK]
Driver=/opt/ConnectforODBC60/lib/ivsqls24.so
Description=DataDirect 6.0 SQL Server Native Wire Protocol
AlternateServers=
AlwaysReportTriggerResults=0
AnsiNPW=1
ApplicationName=COPLINK
ApplicationUsingThreads=1
AuthenticationMethod=1
BulkBinaryThreshold=32
BulkCharacterThreshold=-1
BulkLoadBatchSize=1024
BulkLoadOptions=2
ConnectionReset=0
ConnectionRetryCount=0
ConnectionRetryDelay=3
Database=<SQL Server Coplink Database Name>
```

```
EnableBulkLoad=0
EnableQuotedIdentifiers=0
EncryptionMethod=0
FailoverGranularity=0
FailoverMode=0
FailoverPreconnect=0
FetchTSWTZasTimestamp=0
FetchTWFSasTime=1
GSSClient=native
HostName=<FQDN to COPLINK SQL Server>
HostNameInCertificate=
InitializationString=
Language=
LoadBalanceTimeout=0
LoadBalancing=0
LoginTimeout=15
LogonID=
MaxPoolSize=100
MinPoolSize=0
PacketSize=-1
Password=
Pooling=0
PortNumber=1433
QueryTimeout=0
ReportCodePageConversionErrors=0
SnapshotSerializable=0
TrustStore=
TrustStorePassword=
ValidateServerCertificate=1
WorkStationID=
XML Describe Type=-10
```

12. Define WebSphere Application Server environment variables for `LD_LIBRARY_PATH` within the Process Definition window for the Cognos server:

    a. Log in to the IBM Intelligent Operations Center Portal server as `wpsadmin`.

    b. Select **Administration**.

    c. Select **Intelligent Operations** → **Administration Tools** → **Administration Consoles**.

    d. Select **Application Server**.

    e. Log in to the WebSphere Application Server Integrated Solutions Console (default admin user is `waswebadmin`).

    f. Select **Servers** → **Server Types** → **WebSphere application servers**.

    g. Find and click **CognosX_Disp1**.

    h. Expand **Java** and **Process Management** and click **Process Definition**.

    i. Click **Environment Entries**.

    j. Update the `LD_LIBRARY_PATH` entry to add the path to the DataDirect libraries:

       `/opt/ConnectforODBC60/lib/`

    k. Select **Servers** → **Server Types** → **WebSphere application servers**.

    l. Find and click **CognosX_GW1**.

m. Expand **Java** and **Process Management** and click **Process Definition**.

n. Click **Environment Entries**.

o. Update the `LD_LIBRARY_PATH` entry to add the path to the DataDirect libraries:

`/opt/ConnectforODBC60/lib/`

p. Log out of the WebSphere Application Server Integrated Solutions Console.

13. Restart the IBM Intelligent Operations Center application server for the changes to take effect.

> **Note:** Ensure that all the IBM Intelligent Operations Center servers are correctly shut down by using the `IOCControl.sh` command on the IBM Intelligent Operations Center management server before you restart the application server.

Define data sources in Cognos BI

Configure the IBM i2 Intelligence Analysis Platform database locally and define an alias to the remote database on the server.

Perform the following steps:

1. Log on as the `root` user on the IBM Intelligent Operations Center application server where IBM Cognos Business Intelligence is installed.

2. Open a command prompt.

3. Create the local database running the following commands:

```
cd /opt/IBM/DB2/bin
./db2 uncatalog node IAPNode
./db2 uncatalog database <schema_name>
./db2 catalog tcpip node IAPNode remote <hostname> server <port>
./db2 catalog db schema_name as IAP at node IAPNode
```

The variables are defined:

– **<schema_name>** is the name of the schema for the IBM i2 Intelligence Analysis Platform database.

– **<hostname>** is the IP address of the DB2 server.

– **<port>** is usually `50000`.

3.3.3 Deploying IBM Cognos BI Report packages

This section describes the tasks to deploy the Cognos BI Report packages that are required as part of the implementation of the i2 Intelligent Analysis Platform and the i2 COPLINK reporting capabilities.

The following high-level tasks deploy the Cognos BI Report packages:

1. Deploy the `ile_model.zip` file on the IBM Cognos BI server.

2. Create i2 Intelligent Analysis Platform and i2 COPLINK data source connections.

3. Install the ILE Cognos Reports package.

Deploy the ile_model.zip file on the IBM Cognos BI server

For the i2 Integrated Law Enforcement Reporting capability to function, the Cognos reports model `ile_model.zip` file must be deployed on the IBM Cognos BI server.

Perform the following steps:

1. Copy the `ile_model.zip` file from the `REPORTING_INSTALL/cognos` directory to the `/IBM/cognos/c10_64/deployment` directory on the IBM Intelligent Operations Center application server.

2. Launch IBM Cognos Connection pointing the browser to the following URL where *<hostname>* is the IBM Intelligent Operations Center application server host name:

   `http://`*<hostname>*`:9082/p2pd/servlet/dispatch/ext`

3. Log in as administrator, for example, `wpsadmin`.

4. Click **Launch** in the upper-right corner of the IBM Cognos Connections window and select **IBM Cognos Administration** from the drop-down list, as shown in Figure 3-22.
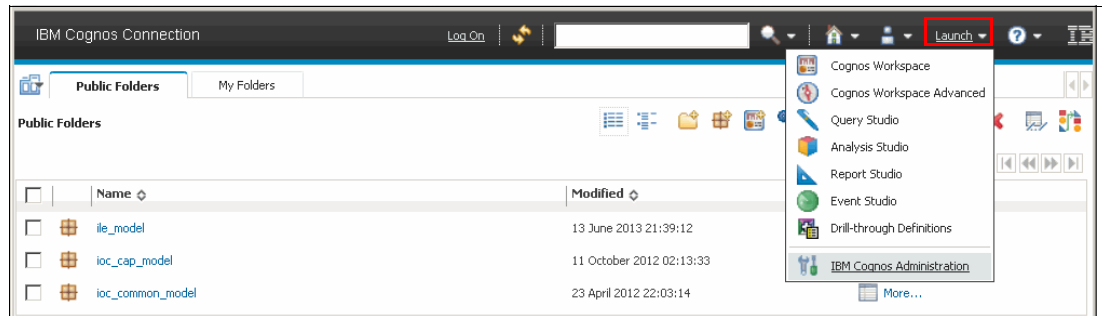
Figure 3-22   Launch IBM Cognos Administration

Create data source connections

Two Data Source Connections must be created on the IBM Cognos BI server: One to the i2 COPLINK database and one to the i2 Intelligent Analysis Platform Reporting database. Perform the following steps:

1. On the Configuration tab, select **Data Source Connections** as shown in Figure 3-23.
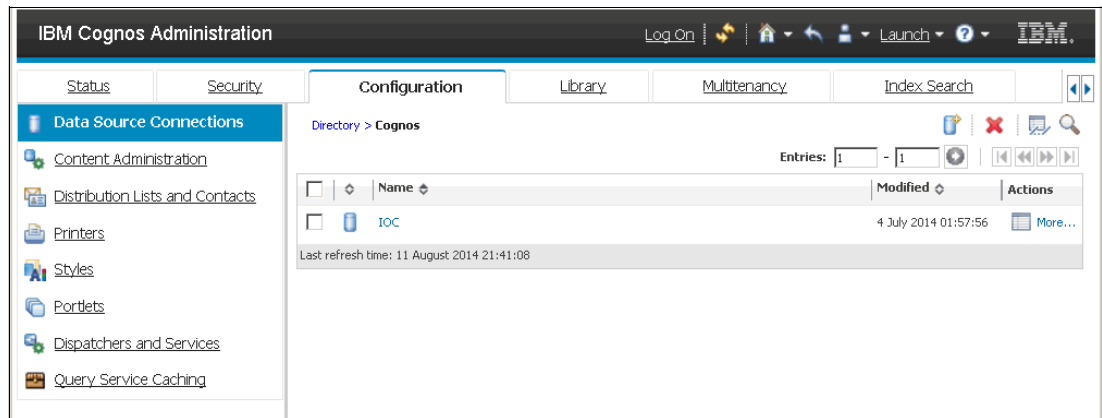
Figure 3-23   IBM Cognos Administration Data Source Connections

2. To create the i2 COPLINK Data Source connection, click the **New Data Source** icon in the upper-right corner as shown in Figure 3-24 on page 95.
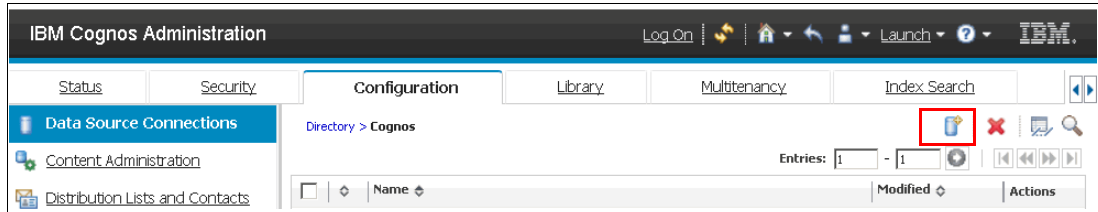
Figure 3-24   IBM Cognos Administration New Data Source connection icon

3. Specify **COPLINK** as the data source name and click **Next**.

4. In the Type field, select either **Microsoft SQL Server (ODBC)** or **Oracle**, depending on the type of i2 COPLINK database. Ensure that the **Configure JDBC connection** check box is cleared and click **Next**.

5. Follow these steps on the Connection String page:

   a. For Microsoft SQL Server connections, enter `COPLINK` in the ODBC Data source field. This value is derived from the Application Name parameter in the DataDirect for ODBC `odbc.ini` file.

   b. For Oracle connections, enter the Service Name for the i2 COPLINK database server as defined in "Defining the Oracle system identifier (SID)" on page 89. Use the NetMgr utility in the `SQL*Net` connect string. Ensure that **Signon** is selected and that the password box is checked in the Signon section.

6. Provide valid database credentials in the user ID and password fields, and click **Finish** to save the Data Source connection.

7. Before you create the i2 Intelligent Analysis Platform data source, you must create a local DB2 node and database aliases to the i2 Intelligent Analysis Platform Reporting database.

   Open a command prompt on the IBM Intelligent Operations Center application server, navigate to the `/opt/IBM/DB2/bin` directory, and enter the following commands:

   `./db2 catalog tcpip node IAPNode remote *<hostname>* server *<port>*`

   `./db2 catalog db REPORTIN as IAP at node IAPNode`

   The variables are defined:

   - *`<hostname>`* is the hostname of the DB2 server where the i2 Intelligent Analysis Platform Reporting database is hosted.

   - *`<port>`* is the service port for the DB2 instance hosting the i2 Intelligent Analysis Platform Reporting database. If the i2 Intelligent Analysis Platform Reporting database is hosted on the read server, the port is usually `50000`.

8. To create the i2 Intelligent Analysis Platform Data Source connection, click the **New Data Source** icon (see Figure 3-24).

9. Specify `IAP` as the data source name and click **Next**.

10. In the Type field, select **IBM DB2**. Ensure that the **Configure JDBC Connection** check box is cleared and click **Next**.

11. Enter `IAP` as the DB2 database name. Ensure that **Signon** is selected and that the password box is checked in the Signon section.

12. Provide valid database credentials in the User ID and password fields and press **Finish** to save the Data Source connection.

Install the ILE Cognos Reports package

After the data source connections are created, the ILE Cognos Reports package can be installed. Perform the following steps:

1. In the IBM Cognos Administration console, select **Content Administration** on the Configuration tab (see Figure 3-25).
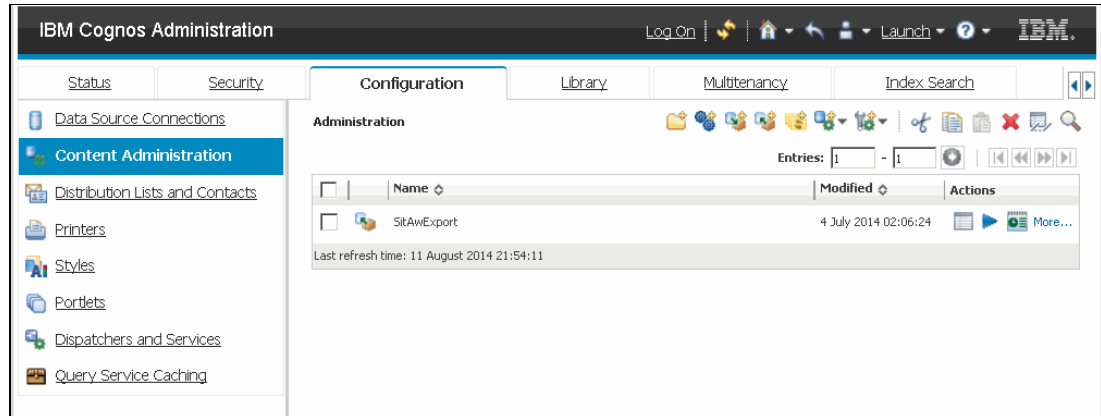
Figure 3-25   IBM Cognos Administration Content Administration

2. Click the **New Import** icon as shown in Figure 3-26.
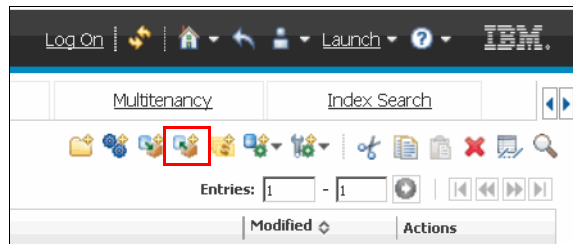
Figure 3-26   IBM Cognos Administration New Import icon

3. Select **ile_model** as shown in Figure 3-27, and then click **Next** twice.
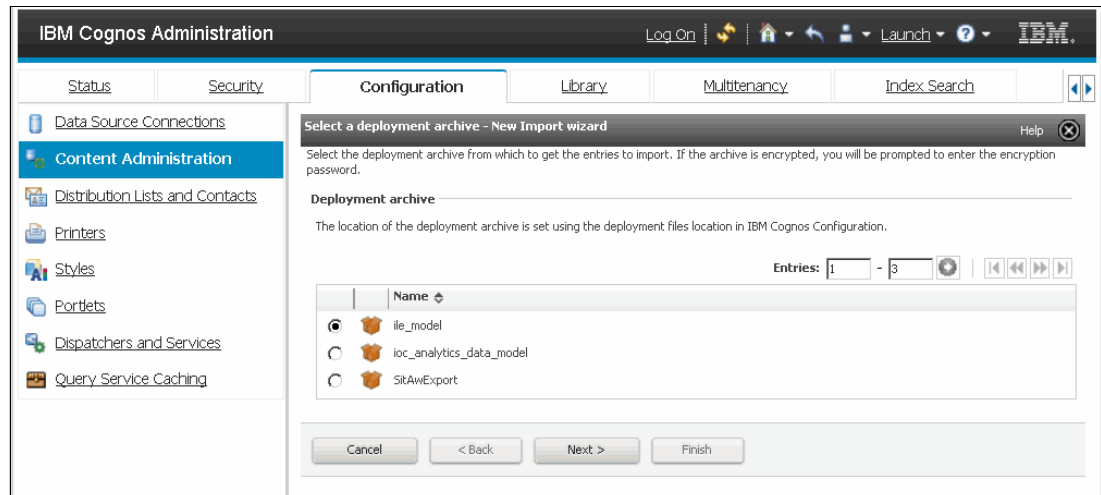
Figure 3-27   IBM Cognos Administration New Import wizard

4. Ensure sure that the check box next to **ile_model** is selected (see Figure 3-28) and click **Next** three times.
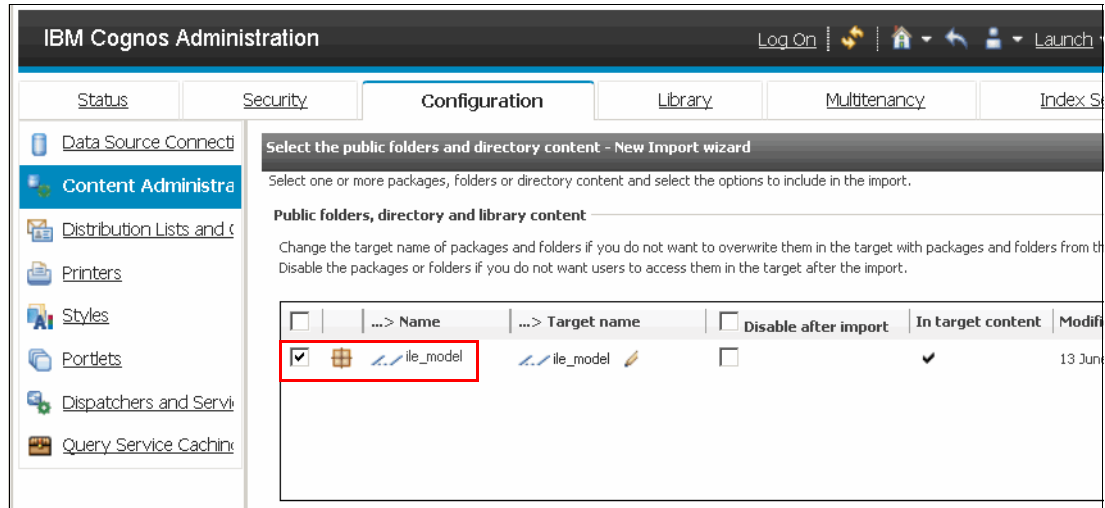
Figure 3-28   IBM Cognos Administration New Import wizard content selection

5. Click **Save and run once** and then click **Finish**.

6. Select **Now**, click **Run**, and click **OK**.

3.3.4 Configuring the portal pages for reports

For the i2 Integrated Law Enforcement reports to be displayed on the ILE Console, several portal pages must be added.

The core component distribution media contained in the IBM i2 Intelligent Law Enforcement V1.0.1 Multilingual Media Pack (BF079ML) is a `.zip` archive named `I2_ILE_V1.0.1_WIN_ML.zip`, which also contains two archives: `Reporting_Install.zip` and `Sitaw_Install.zip`.

Templates for the portal pages to display the reports pages are provided in the `Reporting_Install.zip` file.

Follow these steps:

1. To install the i2 Integrated Law Enforcement reporting portal pages, copy `Reporting_Install.zip` to a temporary directory on your workstation and extract the content. The directory structure of the extracted content has a directory named `REPORTING_INSTALL` with the contents shown in Example 3-8 on page 90.

   Example 3-10   REPORTING_INSTALL directory structure and content

   ```
   REPORTING_INSTALL
   +---Cognos
   +---Coplink_Reporting_Installer
   +---IAP_Reporting_Installer
   \---ILE_Console_Pages
           reportsPage.xml
   ```

2. On the workstation where `Reporting_Install.zip` was extracted, point a browser to `https://<IOC_Application_Server>/wpsv70/wps/myportal/` and log in to the WebSphere Portal Server as a user with administrative rights.

3. Click **Administration** on the top menu bar to open the WebSphere Portal Administration console.

4. On the left menu bar, click **Portal Settings** → **Import XML** and select the **IAP_Reporting_Installer/ILE_Console_Pages/reportsPage.xml** from the directory to which `Reporting_Install.zip` was extracted.

5. Click **Import** to add the Reports pages to the i2 Intelligent Law Enforcement console.

Reports are now configured to display on the i2 Intelligent Law Enforcement console.

3.4 Analysis Search

This section focuses on the deployment and configuration of the third extended capability of i2 Integrated Law Enforcement: Analysis Search.

IBM i2 Intelligent Law Enforcement V1.0.1 includes the following products that are needed to implement the Analysis Search extended capability:

► IBM i2 Information Exchange for Analysis (iXa) Search for Analyst's Notebook V8.9.1
► IBM i2 COPLINK Analysis Search Standard 4.8.0.0

The products that compose this extended capability are IBM i2 Analyst's Notebook Premium V8.9.3 and IBM i2 COPLINK V4.8.

The following high-level tasks deploy Analysis Search:

1. Installation and configuration of IBM i2 COPLINK Analysis Search on IBM i2 COPLINK. See 3.4.1, "Installing and configuring i2 COPLINK Analysis Search on i2 COPLINK" on page 98.

2. Installation of IBM i2 Information Exchange for Analysis (iXa) Search for Analyst's Notebook. See 3.4.2, "Installing IBM i2 Information Exchange for Analysis Search" on page 99.

3. Configuration of IBM i2 Information Exchange for Analysis (iXa) Search for Analyst's Notebook to work with IBM i2 COPLINK Analysis Search. See 3.4.3, "Configuring IBM i2 Information Exchange for Analysis Search with IBM i2 COPLINK Analysis Search" on page 99.

3.4.1 Installing and configuring i2 COPLINK Analysis Search on i2 COPLINK

There are two components in i2 COPLINK Analysis Search: the server side and the client side. Both components must be installed and configured. As mentioned in 2.2.1, "Know your team" on page 22, the installation and configuration of i2 COPLINK products are performed by IBM i2 Lab Services. Therefore, the steps for installing and configuring the server side of IBM i2 COPLINK Analysis Search are not described here.

Note: Because you will be working with IBM i2 Lab Services, you will need to coordinate with them closely. The following steps, 3.4.2, "Installing IBM i2 Information Exchange for Analysis Search" on page 99 and 3.4.3, "Configuring IBM i2 Information Exchange for Analysis Search with IBM i2 COPLINK Analysis Search" on page 99, are performed *after* the server side is installed and configured with the assistance of IBM i2 Lab Services.

3.4.2 Installing IBM i2 Information Exchange for Analysis Search

You need to perform the client side of the installation on the client machine as a user with administrative privileges. This installation is only done on a Windows machine where IBM i2 Analyst's Notebook Premium is installed. The product, IBM i2 Information Exchange for Analysis Search for Analyst's Notebook, must be installed on IBM i2 Analyst's Notebook Premium before you can install IBM i2 COPLINK Analysis Search.

Consider the following information:

► If you are installing using Terminal Services, you must install using the Add or Remove Programs facility in the Control Panel. To complete the installation, users of IBM i2 Information Exchange for Analysis must log off from Terminal Services and log on again.

► If you are deploying on Citrix, see the i2 Information Exchange for Analysis Search *Packaging and Deployment Guide* about how to publish the application and user profile information. This guide is included in the product distribution.

The following list provides pointers to relevant information:

► About third-party software: Installing IBM i2 Information Exchange for Analysis (iXa) Search for Analyst's Notebook installs third-party software. For the complete information about the third-party software that is installed and how to prevent its installation, see the *Packaging and Deployment Guide*.

► About Microsoft .NET Framework language packs: To display Microsoft .NET Framework information dialogs in a local language, you need to install a corresponding language pack. You can download Microsoft .NET Framework Version 2 language packs from the Microsoft website.

► About language-specific files: The user locale setting controls the installation of language-specific files and folders. For more information, see the *Packaging and Deployment Guide*.

► About customizing installed files: For information about customizing installed files, see the *Packaging and Deployment Guide*.

► About modification and uninstallation: Installing IBM i2 Information Exchange for Analysis Search for Analyst's Notebook adds an entry, `IBM i2 iXa Search AN`, to Add or Remove Programs. You can use this entry to repair or remove IBM i2 Information Exchange for Analysis (iXa) Search for Analyst's Notebook. The procedure is shown:

 a. Close any applications that are open.

 b. Locate `Setup.exe` in the `i2 iXa Search AN` folder of the IBM i2 Information Exchange for Analysis Search for Analyst's Notebook distribution and run it. The installation process starts.

 c. Follow the prompts to complete the installation.

 d. The following shortcuts will be installed:

  • IBM i2 iXa Search AN $\rightarrow$ IBM i2 iXa Search AN Start iXa Search

  • IBM i2 iXa Search AN $\rightarrow$ Documentation Access the online help

  • The IBM i2 iXa Search AN release notes

3.4.3 Configuring IBM i2 Information Exchange for Analysis Search with IBM i2 COPLINK Analysis Search

Before you begin, ensure that both IBM i2 Analyst's Notebook Premium and IBM i2 Information Exchange for Analysis Search plug-in are installed.

You must obtain the Windows installer (.msi file) for i2 COPLINK Analysis Search from IBM i2 Lab Services after they install the server-side component of this product. This installer automates the configuration of IBM i2 Information Exchange for Analysis Search on i2 COPLINK Analysis Search.

> **Note:** Close IBM i2 Analyst's Notebook Premium and log in as an administrator before you perform the following steps.

Perform the following steps:

1. Copy the i2 COPLINK Analysis Search .msi file that you received from IBM i2 Lab Services to a suitable directory on the local drive and run it.

2. At the Welcome window, click **Next** (see Figure 3-29).
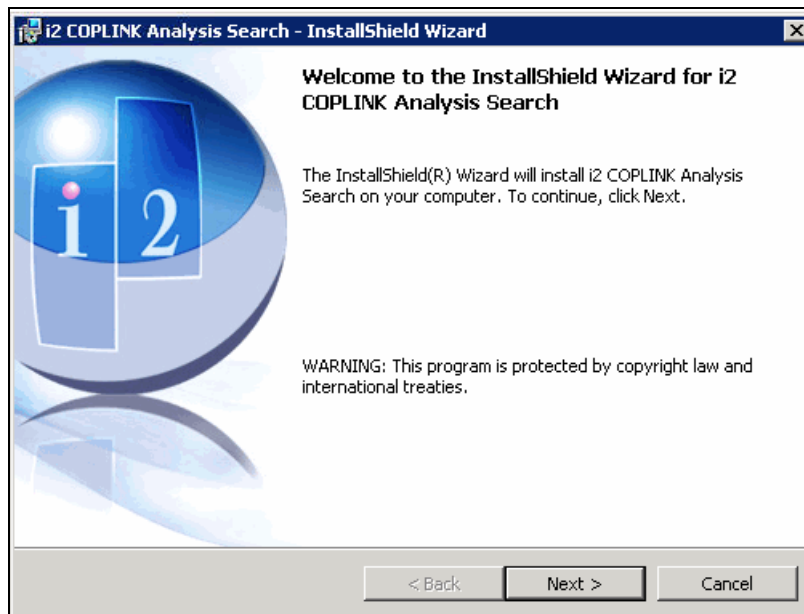
Figure 3-29   Welcome to i2 COPLINK Analysis Search Installer

3. Choose the appropriate category of use for the license agreement and click **Next** (see Figure 3-30 on page 101).
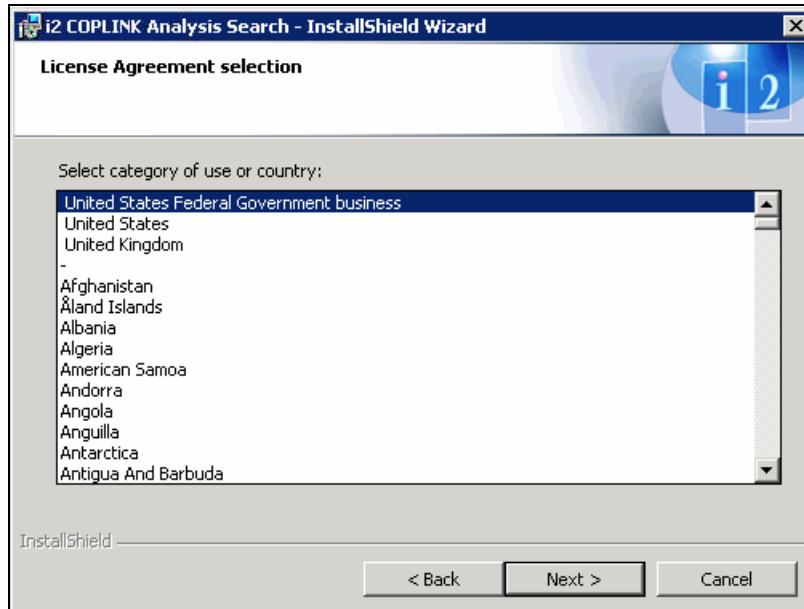
Figure 3-30   i2 COPLINK Analysis Search category of use or country selection

4. Accept the license agreement and click **Next** (Figure 3-31).
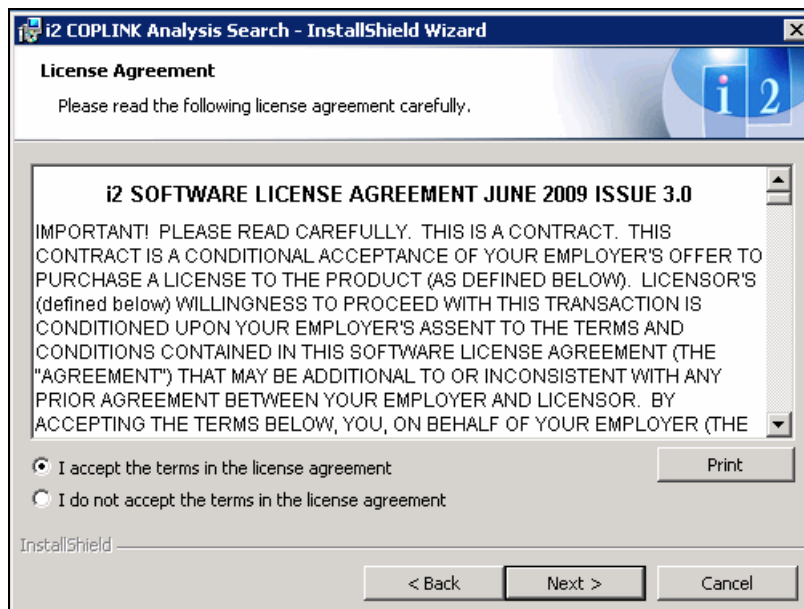
Figure 3-31   i2 COPLINK Analysis Search license agreement

5. Configure the IBM i2 Information Exchange for Analysis Search endpoint. An *endpoint* is a URL pointing to an iXa service running on the i2 COPLINK Analysis Search server. The IBM i2 Lab Services team that performs the i2 COPLINK installation provides the endpoint information. The following sample URL is to the server where i2 COPLINK Analysis Search server side is configured. The server side is configured to use Secure Sockets Layer (SSL) certificates:

```
https://<COPLINK Web Server FQDN hostname>:8443/coplink/ixa
```

6. Configure a display name for this endpoint. The display name can be any recognizable name for the i2 COPLINK instance. It is displayed in IBM i2 Analyst's Notebook Premium by IBM i2 Information Exchange for Analysis Search (see Figure 3-32).
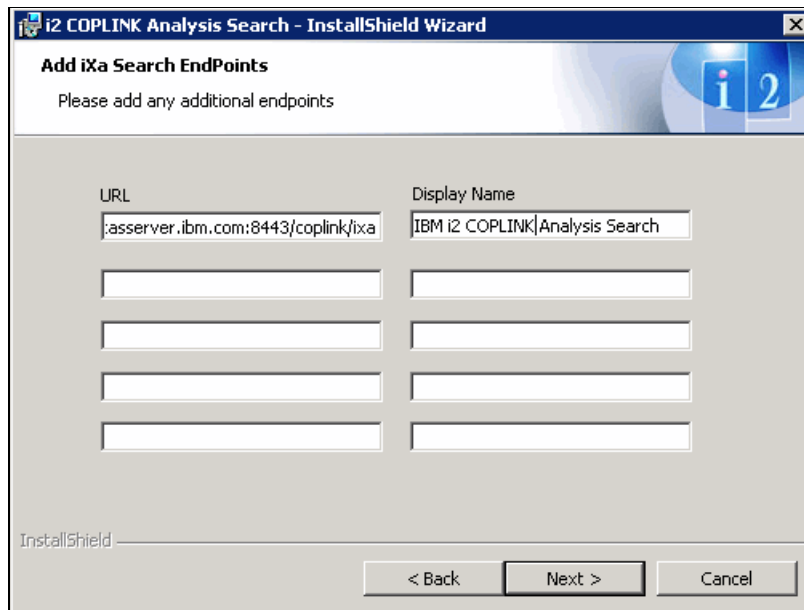
Figure 3-32   i2 COPLINK Analysis Search list of endpoints and display names

7. Click **Next**.

8. Click **Install** to start the installation process (see Figure 3-33).
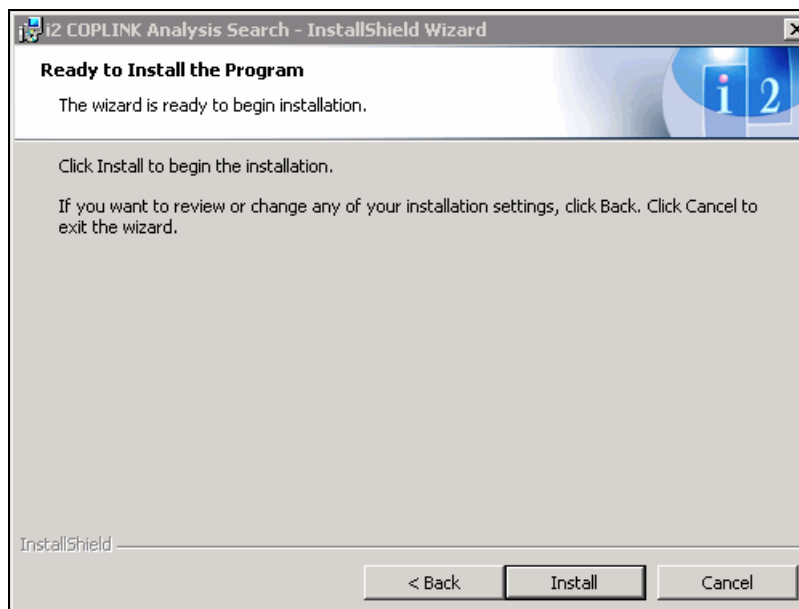
Figure 3-33   i2 COPLINK Analysis Search final installation window

9. If a security notification is displayed, click **Yes** to allow the installation to proceed.

10. Click **Finish** to complete the installation (see Figure 3-34 on page 103).
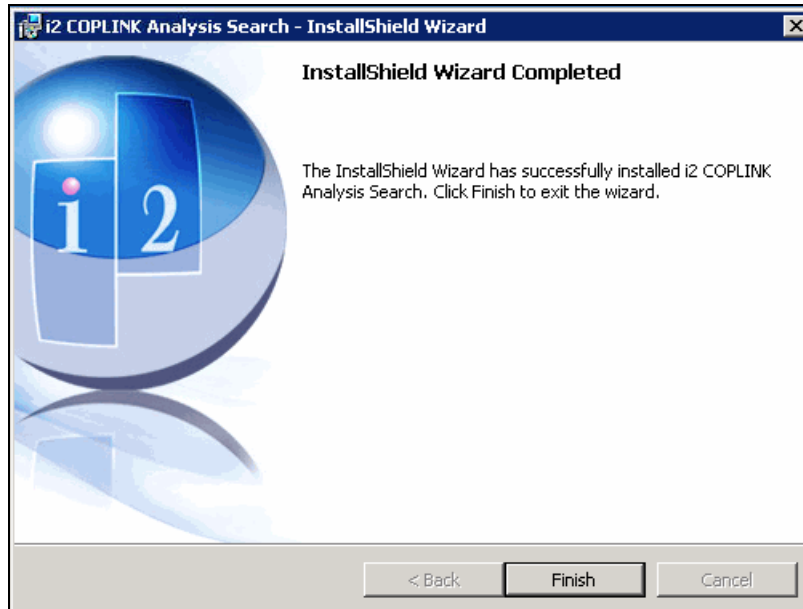
Figure 3-34   i2 COPLINK Analysis Search finish installation

The installation and configuration of the Analysis Search extended capability are complete. The next step is to verify that this capability works correctly.

3.4.4 Testing the i2 COPLINK Analysis Search configuration

This section describes how to verify that your installation and configuration of Analysis Search are successful by testing connectivity with the server-side component.

Perform the following steps:

1. Start IBM i2 Analyst's Notebook Premium.
2. Navigate to **Data** → **IBM i2 iXa Search AN** → **Configure Data Sources** (see Figure 3-35 on page 104).
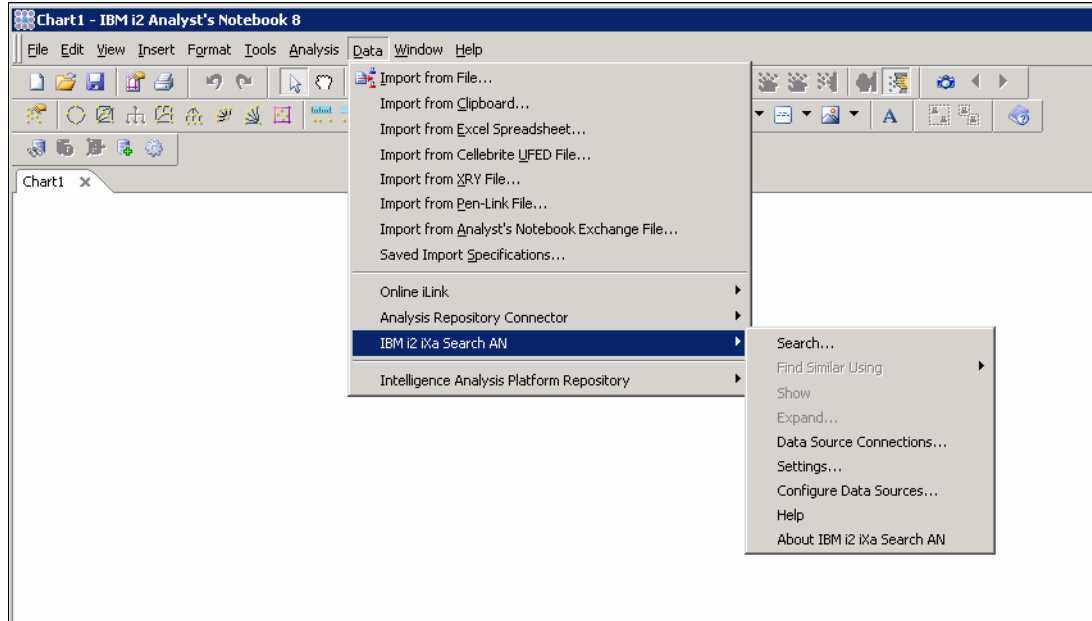
Figure 3-35   Navigation to the Configure Data Sources task

3. In the Configure Data Sources panel, highlight the **IBM i2 COPLINK Analysis Search** data source and click **Modify** (see Figure 3-36).
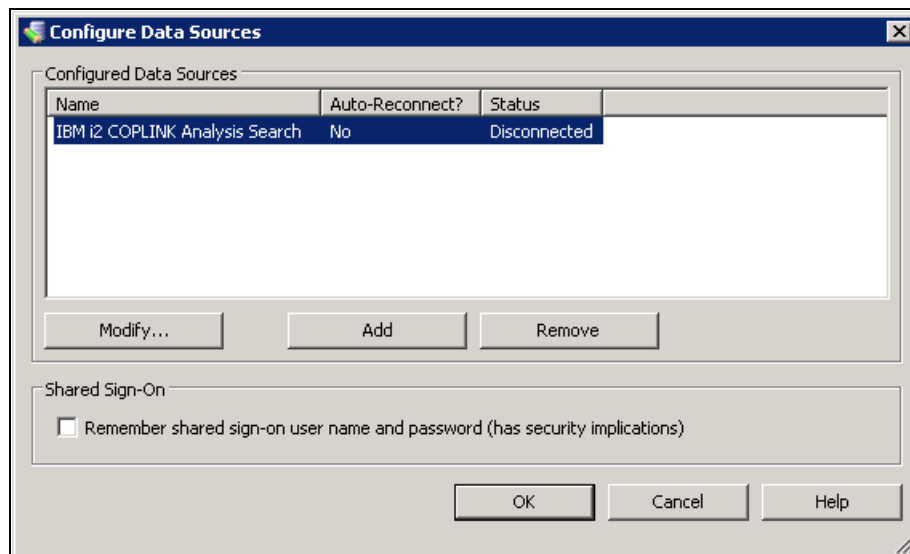
Figure 3-36   Navigation to Data Sources configuration

4. Click **Test** (see Figure 3-37 on page 105).
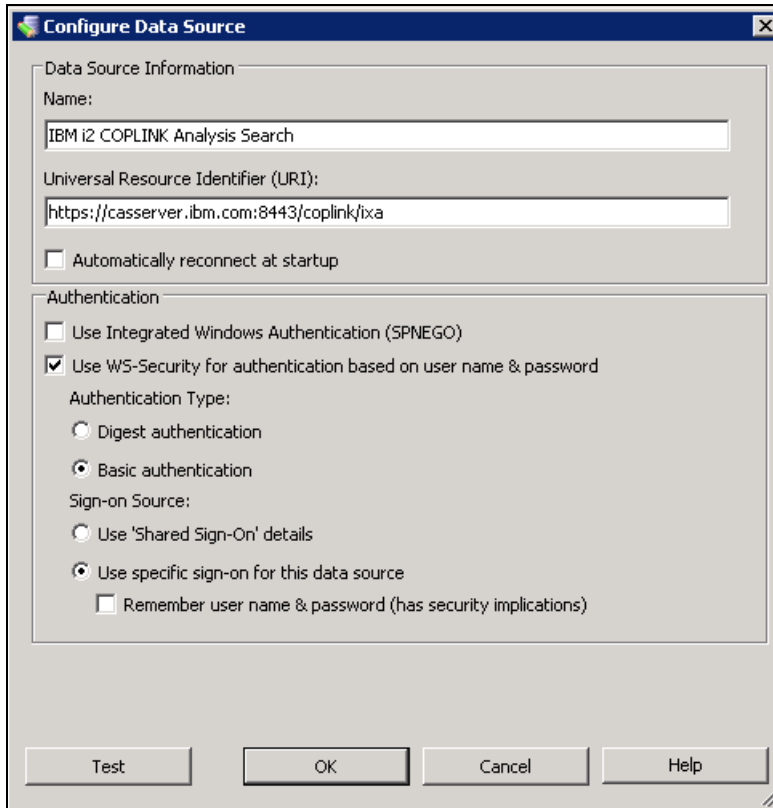
Figure 3-37   Data Sources configuration

5. Enter your i2 COPLINK user credentials and click **OK**. You must specify your agency followed by your user name. IBM i2 Information Exchange for Analysis Search attempts to connect to the i2 COPLINK Analysis Search Web Service (see Figure 3-38).
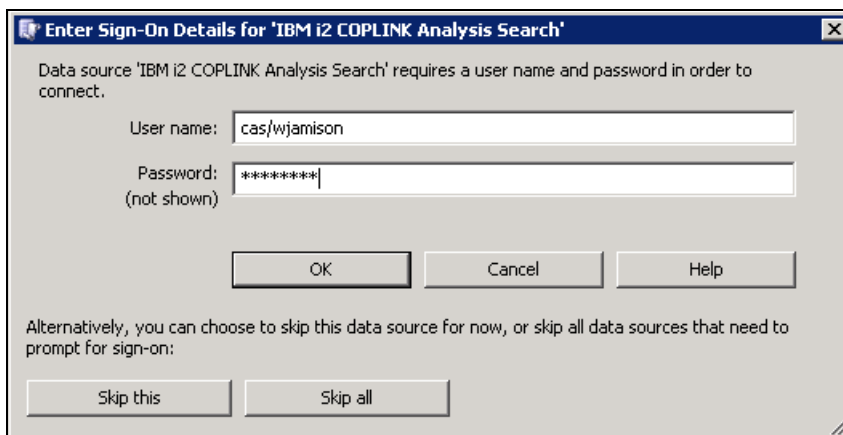
Figure 3-38   i2 COPLINK user credentials

If the test is successful, the status in the Configure Data Sources window (shown in Figure 3-36 on page 104) changes from `Disconnected` to `Connected`.

3.5 Intelligence Portal

A portal page for the i2 Intelligence Analysis Platform was deployed on the IBM Intelligent Operations Center in 3.2.7, "Deploying the i2 Intelligent Law Enforcement V1.0.1 console on the IBM Intelligent Operations Center portal server" on page 59 in step 17 on page 60.

This page needs additional configuration to enable the i2 Intelligent Law Enforcement V1.0.1 console to display the IBM i2 Intelligence Analysis Platform.

Perform the following steps to configure Intelligence Analysis Portal:

1. Point a browser to **https://<_ioc_application_server_>/wpsv70/wps/myportal/** and log on to the IBM Intelligent Operations Center as the administrative user.

2. Click **Administration** → **Portlet Management** → **Portlets**.

3. Locate the IAP_Intelligence_Portal portlet.

4. Click **Configure portlet for IAP_Intelligence_Portal**.

5. Click **Edit value for IAP_Portal_URI**.

6. Enter `http://<IAP_write_server>/apollo/` in the Value field.

7. On the Edit preference IAP_Portal_URI page, click **OK**.

8. On the Configure portlet: IAP_Intelligence_Portal page, click **OK**.

9. Click **Log out** in the upper-right corner.

Finally, you need to configure single sign-on (SSO) between IBM Intelligent Operations Center and i2 Intelligence Analysis Platform. Single sign-on is described in Chapter 5, "Integrated Law Enforcement single sign-on" on page 163.

4

Integrated Law Enforcement security

Chapter 2, "Integrated Law Enforcement deployment" on page 21 lists five major project milestones when you deploy IBM i2 Integrated Law Enforcement. The scope of the fourth milestone is to *customize* the deployment of the environment to meet the requirements of the client.

This chapter is designed to provide you with the conceptual understanding of how security works in each of the product components and give you a good grasp of what it means to integrate them all together as one global security infrastructure.

This chapter does not claim to be a technical guide on how to set up Microsoft Active Directory in WebSpere Application Server, how to enable mutual authentication in Secure Sockets Layer (SSL) between two applications, implement Kerberos in this solution, and other specific and well-documented security-related topics. An exception to this rule, however, is a technical section on how to synchronize IBM Intelligent Operations Center directory server with your client's directory server, which is also one of the customization requirements when deploying i2 Integrated Law Enforcement. An excellent reference for advanced security topics is the IBM Redbooks publication, *WebSphere Application Server V7.0 Security Guide*, SG24-7660.

This chapter highlights the following topics:

► Security models use by each of the product components
► Global security for the IBM i2 Integrated Law Enforcement solution
► Integrating your client's security directory server with the solution

4.1 Overview

Several areas might require customization during deployment, including security. Likely, your client ranks security customization as one of, if not, the top priority to implement immediately for two important reasons:

► Law enforcement data is highly sensitive.

► An integrated solution, such as i2 Integrated Law Enforcement, consists of differing security models. Different security models can be a potential source for a security breach if they are not carefully reviewed and validated.

Required security protections (at the data level or process level) must be implemented correctly and tested rigorously.

Your client might already visualize how to integrate i2 Integrated Law Enforcement into their existing security infrastructure. But together, you must help the client to understand all the details, evaluate the design, and provide recommendations. By having this information early in the project timeline, you can prepare and design the customization process more accurately and implement that process more easily.

Security customization entails a significant number of configuration tasks in addition to security modeling. More often than not, your client's existing security infrastructure, security policies, and organizational structure have great influence on how security must be implemented.

Security is broad and there might be different levels of it. Various technology solutions exist around security. Designing and implementing security aspects of the solution will be major work on your part. The topics described in this chapter can help you with this task.

4.2 Security models

Two basic requirements exist in any security system:

► *Authentication* refers to how the system identifies, verifies, and validates the identity of a user trying to use the system.

► *Authorization* refers to how the system, after it establishes the identity of a user, determines and enforces the operations and tasks that the user is allowed to carry out on resources in the system.

A security model is used to communicate the general design and implementation of a security system that addresses these two requirements. The security model describes the precise semantics of authentication and authorization (and possibly other aspects of security) in a certain system, and the mechanics of achieving the desired behavior under those semantics.

This section describes the individual security models used by IBM i2 Intelligence Analysis Platform, IBM Intelligent Operations Center, and IBM i2 COPLINK.

4.2.1 IBM i2 Intelligence Analysis Platform security model

IBM i2 Intelligence Analysis Platform uses a multi-dimensional security model that allows you to describe your client's security requirements in a flexible way.

Authentication

Authentication is required before any user can connect and use a deployed instance of IBM i2 Intelligence Analysis Platform. The method used is basic authentication, which requires a user ID and password. The system can be configured to use any type of user registry that IBM WebSphere Application Server V8.0 supports, such as Lightweight Directory Access Protocol (LDAP), local operating system, federated registry, and custom registry.

Authorization

The most valuable resource in an IBM i2 Intelligence Analysis Platform installation is an item (and the collection of such items). An *item* is either an entity or a link. An item and its relationship with other items represent intelligence information. The i2 Intelligence Analysis Platform provides security protection *at the item level*. To learn more about entities and links, see the *IBM i2 Intelligence Analysis Platform V3.0 documentation* at this website:

http://www.ibm.com/support/knowledgecenter/SSFKBU/com.ibm.i2.welcome.doc/PortalWelcome.html?lang=en

For every item, the authorization given to a user always comes in a pair of values that contains the following information:

- ► *Access right*: The right of a user to act on a particular item
- ► *Grant right*: The right of a user to give or revoke other user's rights (including the user) on a certain item

Assume user A is a user who has been authenticated successfully by the system. Suppose that user A is interested in item X, which contains information about a vehicle.

The authorization process for this example includes these steps:

1. i2 Intelligence Analysis Platform evaluates what user A is allowed to do with item X.
2. After computing user A's authorization based on user A's *credentials*, the system returns two values:
   - – User A's access right for item X is READ ONLY.
   - – User A's grant right for item X is NONE.

These values are defined:

- ► User A can view all the contents of item X but cannot modify them.
- ► User A is not allowed to change the access right of other users for item X.

The two user rights given to user A are applied to item X and *all* of its versions.

Soon, we will explain how i2 Intelligence Analysis Platform evaluates someone's user rights on a certain item.

Formalization of the security model

We describe the security model of i2 Intelligence Analysis Platform. The model is anchored on three fundamental concepts: *dimensions*, *levels*, and *permissions*.

Security dimensions

The security model is based on the idea that all users are multifaceted, and every user can be described based on these different facets. A user possesses attributes, for example, male, female, black hair, blue eyes, American, Indonesian, engineer, clerk, tall, short, vegetarian, athletic, or musician. A user can also belong to an entity, both abstract and concrete. For example, a user can belong to a sports team, a guild, a department in the office, or a community. Many other types of relationships can be used to describe users.

This approach of describing users is used to *differentiate* them from one another, or to *classify* them and group similar users together. By classifying or differentiating users, permissions can be given or removed to users based on these facets. A facet is the closest analogy to a dimension in i2 Intelligence Analysis Platform. For example, *current location* can be a dimension that tells you where a user is located. For example, a user can be on one of the seven continents.

A *security dimension* is a mechanism by which i2 Intelligence Analysis Platform implements a facet. In i2 Intelligence Analysis Platform, a user has certain rights, depending on the attribute value that user has on a certain security dimension. Typically, multiple security dimensions are defined in an i2 Intelligence Analysis Platform deployment, and a user is described by multiple values. The rights of the users are then evaluated based on these values. For example, users can be distinguished according to their rank, clearance, and department. In this case, three security dimensions exist:

► Rank: The rank of a law enforcement officer
► Clearance: The security clearance given to a user
► Department: The name of the department a user belongs to

Each security dimension is defined by a set of possible values, called *security dimension values*. A user is assigned at least one security dimension value for every security dimension defined in i2 Intelligence Analysis Platform. The following examples are security dimension values for each of the security dimensions listed earlier:

► Rank: `Lieutenant`, `Sergeant`, `Captain`, or `Major`
► Clearance: `Unclassified`, `Restricted`, `Confidential`, `Secret`, or `Top Secret`
► Department: `Fire`, `Robbery`, `Homicide`, or `IT`

There is no limit to the number of security dimension values that can be defined in a security dimension. Similarly, there is no limit to the number of security dimensions in the i2 Intelligence Analysis Platform security model. However, a typical deployment consists of one to three dimensions with an average of five dimension values per dimension. Performance and complexity are important considerations when you design your security model. As you increase the number of dimensions and dimension values, both performance and complexity are negatively affected. The impact is lower when you add more dimension values than when you add more dimensions.

Security modeling in i2 Intelligence Analysis Platform includes identifying with your client (it is a team effort) the security dimensions that are correct for their organization and enumerating the security dimension values for each dimension. To complete the model, you must define the access permissions. Your security model must encompass all of the target users of i2 Intelligence Analysis Platform. You can adjust your model by adding or removing security dimensions and dimension values. For example, you can add `None` or `Civilian` as another dimension value for the Rank dimension to include users that are not law enforcement officers.

A security dimension is *ordered* when its security dimension values share a ranking order so that a dimension value subsumes or encompasses all other dimension values with lower rank orders. For example, the `Rank` dimension values are arranged so that the rank order increases from `Lieutenant` to `Major`. A user that has `Captain` as a dimension value also has (by implication) the values `Sergeant` and `Lieutenant`, giving the user multiple values. In a *non-ordered* security dimension (for example, the `Department` security dimension), a user can only be given multiple values explicitly. These values are simple enumeration without any particular order.

> **Note:** It might be convenient to add a security dimension value, such as `Others`, to serve as the default value assigned to a user if none of the enumerated security dimension values apply.

Security levels

The concept of user rights was introduced in "Authorization" on page 109.

Security levels are values from which user rights can be assigned. The different security levels for the two user rights, access right and grant right, are described in this section. They are called *access levels* and *grant levels*:

► Access levels: Security access levels are the range of possible values that can be assigned to a *user's access rights* on a certain item. Four security levels are defined in i2 Intelligence Analysis Platform:

  – *None*: The user is not allowed to access the item of interest and does not have any indication of its existence.

  – *Cloaked*: The user is made aware that the item of interest exists, but the user cannot access the item or its contents.

  – *Read only*: The user is allowed to access the item of interest, but the user can only view its contents.

  – *Update*: The user is allowed to view, modify, and delete the item of interest and its contents.

  A *signpost mechanism* is also provided for users with cloaked access to an item. A *signpost* is a customized message that informs users about the cloaked item. For example, the signpost can direct the user to contact the i2 Intelligence Analysis Platform supervisor.

  > **Note:** Access levels are hierarchical. That is, Cloaked access overrides None access, Read only access overrides Cloaked and None access, and Update access overrides Read only, Cloaked, and None access levels.

► Grant levels: Security grant levels refer to the right of controlling the security access right of users to an item. Security grant levels also refer to the ability of users to change the grant level assigned to other users, including their own grant level. Two grant levels are available:

  – *None*: The user cannot change the security access level of any users, including their own, on a certain item.

  – *Update*: The user can evaluate and change the security access and grant levels of other users, including their own, on a certain item.

Note: Users with Update grant level to an item necessarily must know that the item exists. Therefore, if a user is given an access level of None to an item but a grant level of Update, the Update grant level overrides the access level.

Security permissions

Three security concepts exist in i2 Intelligence Analysis Platform: security dimensions, security levels, and security permissions. The relationships of these three concepts can be summarized in the following manner.

Security permission is the mapping of all security dimensions and their dimension values to security levels. It is a three-column table that specifies the security level assigned (the value in the third column) to a pair of a security dimension (first column) and a security dimension value (second column). Because i2 Intelligence Analysis Platform needs to provide two user rights, that is, access rights and grant rights, two types of security permission exist: the security access permission and the security grant permission to an item. A missing pair of a security dimension and one of its dimension values implies a security level of None for that pair.

You can picture security permission as a dichotomy as shown in Figure 4-1. Each side closely mirrors the other side. Both security access permission and security grant permission use security dimensions (and their dimension values) and security levels. You can call one side *grant* and the other side *access*. You can qualify everything on the grant side as a grant security dimension and a security grant level. You can qualify everything on the access side as an access security dimension and a security access level, as shown in Figure 4-1.
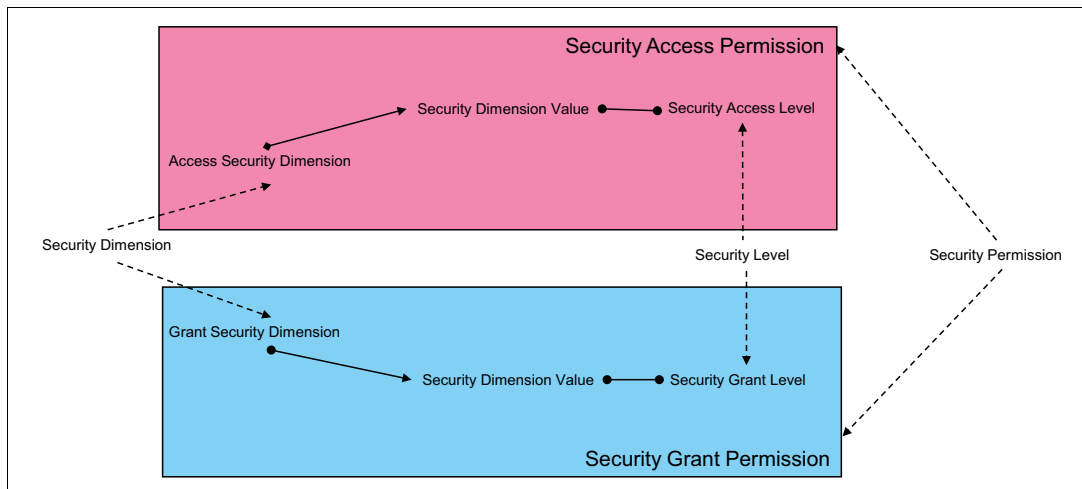
Figure 4-1   Access and grant dichotomy

Every item in i2 Intelligence Analysis Platform has one set of security access permissions and another set of security grant permissions. Figure 4-2 on page 113 shows an example for two sets of security permissions for an item called XYZ. *Rank, Clearance*, and *Department* security dimensions are used as *security dimensions* and *Rank* and *Department* are used as grant security dimensions. Five explicit security access permissions and two explicit security grant permissions are defined. Any security dimension value of these security dimensions that is not included (for example, Sergeant) is, by default, mapped to the security level None. This is true for both the security access permissions and security grant permissions.

Note: It is not unusual to use the same security dimension as an access security dimension and as a grant security dimension. However, two different security dimensions must be defined at implementation time to distinguish between the two uses.

Item XYZ

Security Access Permissions

Access Security Dimension	Access Security Dimension Value	Security Access Level
Rank	Captain	Update
Rank	Lieutenant	Read Only
Clearance	Confidential	Cloaked
Department	Homicide	Read Only
Department	IT	Update

Item XYZ

Security Grant Permissions

Grant Security Dimension	Grant Security Dimension Value	Security Access Level
Rank	Major	
Department	IT	Update

Figure 4-2   Two sets of security permissions for Item XYZ: Access and Grant

A user only has one security access level and one security grant level to an i2 Intelligence Analysis Platform item at any time. The i2 Intelligence Analysis Platform calculates these levels using a two-step process, which involves examining security permissions both *within* and *across* security dimensions:

1. Examine the security dimension values that a user has on a certain security dimension. Use the security permissions of the item in question to map each security dimension value to a security level, and take the *least restrictive,* that is, the strongest user right.

2. Examine all of the *least restrictive* dimension-specific security levels, and take the most restrictive, that is, the weakest user right. The result will be the security level enforced on the user by i2 Intelligence Analysis Platform for that item.

Figure 4-3 on page 114 and Figure 4-4 on page 114 illustrate the two-step process.
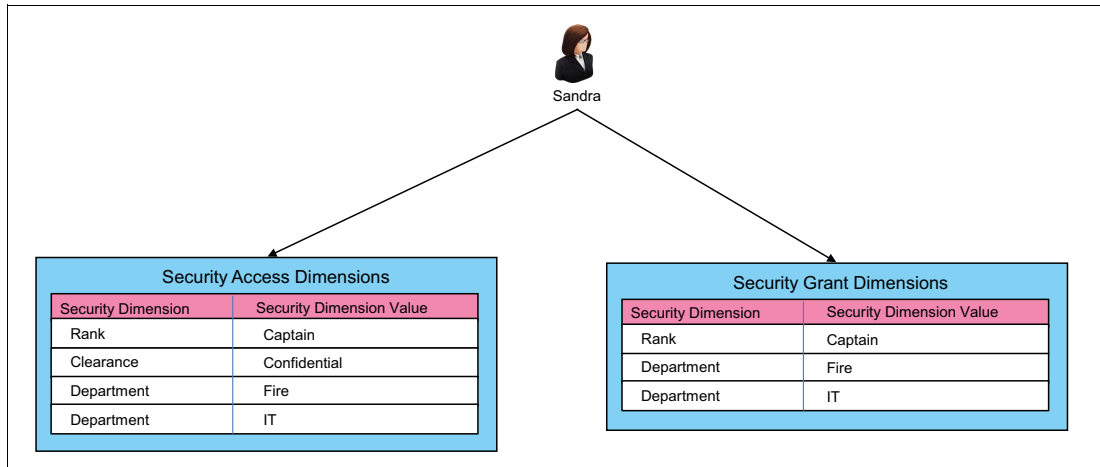
Figure 4-3   Access and Grant dichotomy

Figure 4-3 shows the user profile of Sandra, who is a Captain, working for both the Fire and IT departments, and possessing `Confidential` security clearance. The figure explicitly shows the security dimension values that Sandra has to calculate her access and grant levels for item XYZ. The calculation is shown in Figure 4-4.
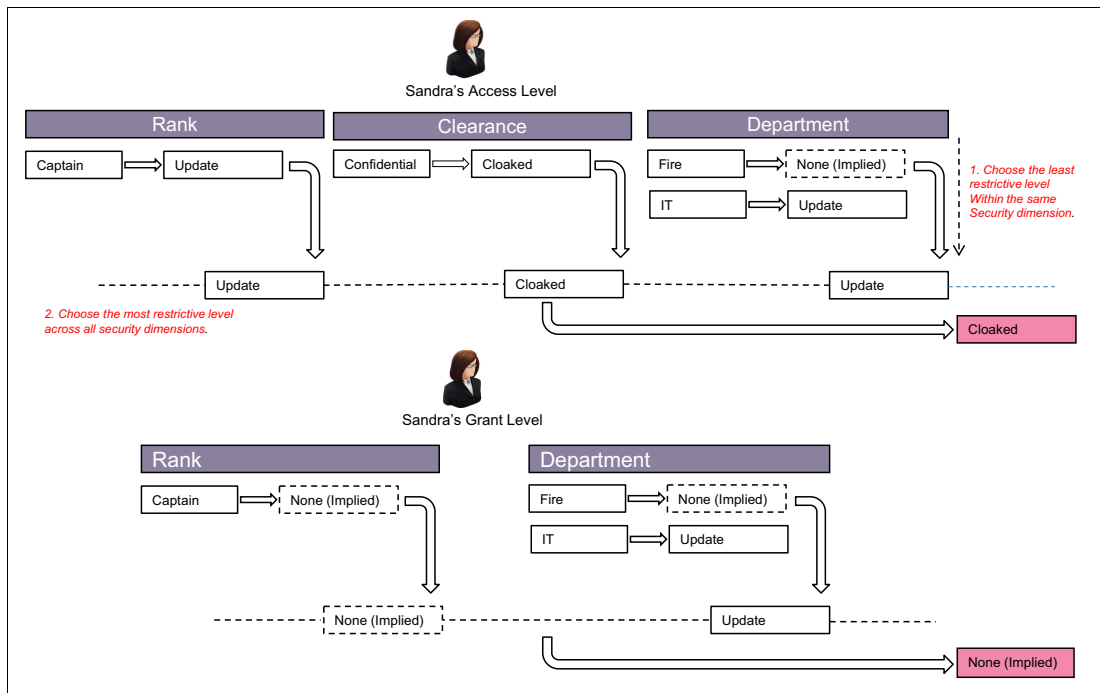
Figure 4-4   Calculating a user's access level and grant level for an item

Sandra's access is shown in Figure 4-4:

► To compute her access level, each dimension is examined for the dimension values assigned to her. Under `Rank`, Sandra only has `Captain`, which is mapped to `Update` for item XYZ as shown in Figure 4-2 on page 113.

► Under the access security dimension value `Clearance`, Sandra has the value `Confidential`, which is then mapped to `Cloaked`.

► Sandra has two values for the access security dimension `Department` (`Fire` and `IT`). Based on Figure 4-2 on page 113, `IT` is mapped to `Update`, and `Fire` is not included in the set of security access permissions. Therefore, `Fire` is mapped to `None`, by default. Taking the least restrictive access level between `None` and `Update`, Sandra receives the `Update` access level for `Department`.

We are left with three user access levels, which are `Update`, `Cloaked`, and `Update`. Taking the most restrictive means that Sandra's final user access level is `Cloaked`. Therefore, user Sandra can access item XYZ with a `Cloaked` permission.

The same approach can be used for calculating Sandra's grant level to item XYZ. Sandra's `Captain` dimension value is mapped to the implied value, `None`. The assignment of `None` occurs because `Captain` is not included in the list of security grant permissions. As for the departments `Fire` and `IT`, only `IT` has an explicit mapping to `Update`, and `Fire` is mapped to `None`, also. Between `None` and `Update`, Sandra receives the higher user level, which is `Update`. However, when the comparison is now across dimensions, it will be between `None` and `Update`. Choosing the most restrictive of the two, Sandra receives `None` as her grant level to item XYZ. Receiving this level means that she is not allowed to change the access rights and grant rights of other users to this item, including her own.

> **Note:** The security model just described is the model used in the initial release of IBM i2 Intelligence Analysis Platform up to Release 3.0.3.x. The model is improved starting with Release 3.0.5.0 and later by the addition of security tags. A *security tag* is a grouping mechanism where you can define a specific security access policy enforced at the item level by putting together a set of access permissions that must be applied as a unit. You can define as many security tags as you want, depending on your organization's security requirements.

Defining the security schema and the default permission

Security Modeling in i2 Intelligence Analysis Platform probably needs a separate book by itself. Defining a security model is not a trivial task because it requires significant analysis of your client's security requirements. It is one of the customization tasks that must be prioritized highly because you will need to hold several sessions with your client. It is imperative that you explain to them how the security model works. That way, they can help validate your model. This is going to be an iterative process and most likely, you will get more questions than answers from your client.

The other customization task is defining the data schema. i2 Intelligence Analysis Platform ships with a sample Law Enforcement Schema. Ensure that your client also understands how the schema works and what needs to be modified or added.

Your security model needs to be transcribed into an XML file.

Example 4-1 on page 116 shows you what a security schema looks like. As you can see, one or more access security dimensions and their corresponding dimension values are specified. You will also need to define at least one grant security dimension and its dimension values. Each of these dimension values is identified uniquely with an ID. You can name the security schema with any XML file name.

Example 4-1   A sample security schema in XML

```
<?xml version="1.0" encoding="UTF-8"?>
<ns3:SecuritySchema xmlns:ns3="http://www.i2group.com/Schemas/2013-09-03/ModelData/Security"

xmlns:ns2="http://www.i2group.com/Schemas/2012-05-29/ModelData/v2"
xmlns:ns4="http://www.i2group.com/Schemas/2011-03-

03/ModelSchema">
    <Id>812fb036-dd9b-4190-bf9e-fd89c01f84d6</Id>
    <Version>1</Version>
    <AccessSecurityDimensions>
        <SecurityDimension Id="03dee165-c232-47d7-a369-935c011bc3e4" Name="Security
Classification"

Description="The highest security classification of information that the user is permitted to
see" Ordered="true">
            <Values>
                <Value>Controlled</Value>
                <Value>Unclassified</Value>
            </Values>
        </SecurityDimension>
        <SecurityDimension Id="f873ad24-8940-458a-9a6f-fc2ad36314ed" Name="Intelligence Sources"
Description="The categories of intelligence source from which the user can access information"
Ordered="false" >
            <Values>
                <Value>Human Informants</Value>
                <Value>Open Source Intelligence</Value>
            </Values>
        </SecurityDimension>
    </AccessSecurityDimensions>
    <GrantSecurityDimensions>
        <SecurityDimension Id="67d2c8fc-0e67-4ba3-b658-a4ed2d3c0a0d" Name="Grant Access"
Description="The roles of the user that govern their ability to modify item security settings"
Ordered="false">
            <Values>
                <Value>Security Controller</Value>
                <Value>Other</Value>
            </Values>
        </SecurityDimension>
    </GrantSecurityDimensions>
</ns3:SecuritySchema>
```

Aside from creating the security schema, you will also need to decide together with your client what the default access and grant permissions will be. The *default access permission* is the access permission that will be given to an item when it is created unless it is changed explicitly before saving the new item. After that, only users, with the grant permission of Update as defined by the default grant permission or with the grant dimension value specified explicitly by the item creator, will be able to change the permission for that item.

The default access and grant permissions are specified in a separate system file in i2 Intelligence Analysis Platform called ApolloClientSettings.xml. Example 4-2 on page 117 shows a snippet of the XML file where the default permissions are specified.

Example 4-2   An example of specifying the default permission

```
<!-- Access permissions set on an item by default. Format:
[DimensionId];[Value];[AccessLevel] -->
  <DefaultAccessPermissions>

        03dee165-c232-47d7-a369-935c011bc3e4; Controlled; read_cloaked
        03dee165-c232-47d7-a369-935c011bc3e4; Unclassified; update
        f873ad24-8940-458a-9a6f-fc2ad36314ed; Human Informants; read_only
        f873ad24-8940-458a-9a6f-fc2ad36314ed; Open Source Intelligence; update

  </DefaultAccessPermissions>

 <DefaultGrantPermissions>
        67d2c8fc-0e67-4ba3-b658-a4ed2d3c0a0d; Security Controller; update
  </DefaultGrantPermissions>
```

The default access permission is separated out from the default grant permission. For each permission, you specify the security level that is assigned to a security dimension value. The associated security dimension is specified using its unique ID. Any security dimension value that is not listed implicitly gets a security level of `None`.

4.2.2 The IBM Intelligent Operations Center security model

This section provides an overview of the security model used by IBM Intelligent Operations Center V1.5. "Authentication" on page 117 includes a short description of authentication. Most of this section focuses on authorization. For more detailed information, see the *IBM Intelligent Operations Center for Smarter Cities Administration Guide*, SG24-8061:

http://www.redbooks.ibm.com/abstracts/sg248061.html?Open

Authentication

The security model of IBM Intelligent Operations Center for authentication is based primarily on the portal server technology and a reverse-proxy called *WebSEAL.* Recall that the Intelligent Operations Center consists of a number of IBM products. One major design principle is to ensure that a user logs in to IBM Intelligent Operations Center only one time (as described in Chapter 5, "Integrated Law Enforcement single sign-on" on page 163) and can access every product component after successful authentication, if the user has the correct authorization.

Figure 4-5 on page 118 shows the authentication flow within IBM Intelligent Operations Center.
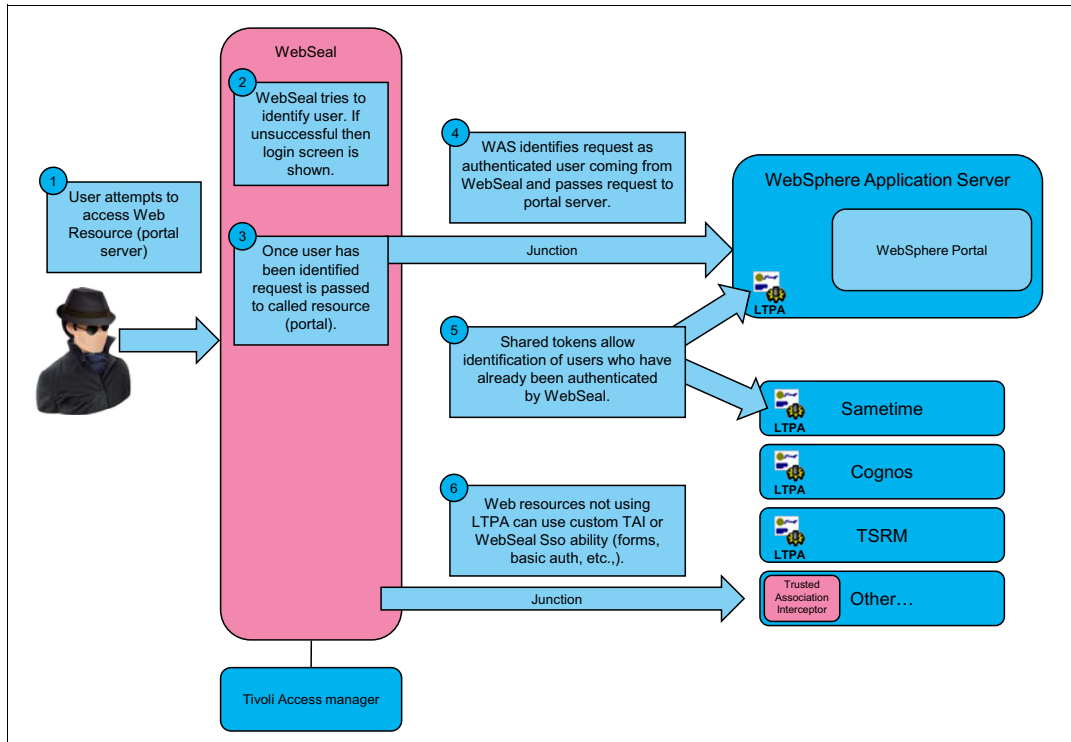
Figure 4-5   High-level process diagram for IBM Intelligent Operations Center authentication

The following numbers refer to the numbers in Figure 4-5:

1. When a user attempts to access a web resource in IBM Intelligent Operations Center, for example, a web page, an interaction takes place with WebSEAL that determines whether the user needs to be authenticated first. This step is part of the single sign-on (SSO) functionality.

2. If authentication is required, the user is challenged to enter a user ID and a password.

3. If successful, WebSEAL passes the credentials to the WebSphere Application Server and a reference to the web resource being requested.

4. WebSphere Application Server identifies the request as an authenticated user from WebSEAL and passes the request to the portal server.

5. To implement SSO in the Intelligent Operations Center, a Lightweight Third Party Authentication (LTPA) token is passed by WebSEAL to another product component that necessarily runs on WebSphere Application Server. The LTPA token contains the credential of the user.

6. For product components that do not run on WebSphere Application Server, a different approach is implemented, such as using a Trust Association Interceptor (TAI).

IBM Intelligent Operations Center provides SSO working across all product components as shipped. Chapter 5, "Integrated Law Enforcement single sign-on" on page 163 describes implementing SSO when integrating new products with IBM Intelligent Operations Center.

> **Note:** IBM Intelligent Operations Center ships its own user registry, which is an LDAP server, implemented by IBM Tivoli Directory Server.

Authorization

Authorization in IBM Intelligent Operations Center is governed primarily through its *access control model*. Protected resources in IBM Intelligent Operations Center are classified into three levels, depending on the granularity of the resource. Each level defines the access control and access permissions that users are given depending on who they are and on their roles. The following permission levels are available:

► *Web resource permissions*: Provides coarse-grained access to web resources that users can access through the IBM Intelligent Operations Center portal.

► *Portal resource permissions*: Provides fine-grained access to the IBM Intelligent Operations Center portal resources.

► *Data category permissions*: Provides access to a category of data, such as events, notifications, and key performance indicators (KPIs), that is displayed in the IBM Intelligent Operations Center portlets.

Access management services determine whether a user can access a web resource based on its URL. Users of IBM Intelligent Operations Center are given access rights to portal resources, for example, pages and portlets, and then to different categories of data, for example, fire-related events, notifications, and KPIs, based on their job responsibilities or roles.

This high-level access control model is shown in Figure 4-6. Notice the different security levels, based on the levels of the protected resource.
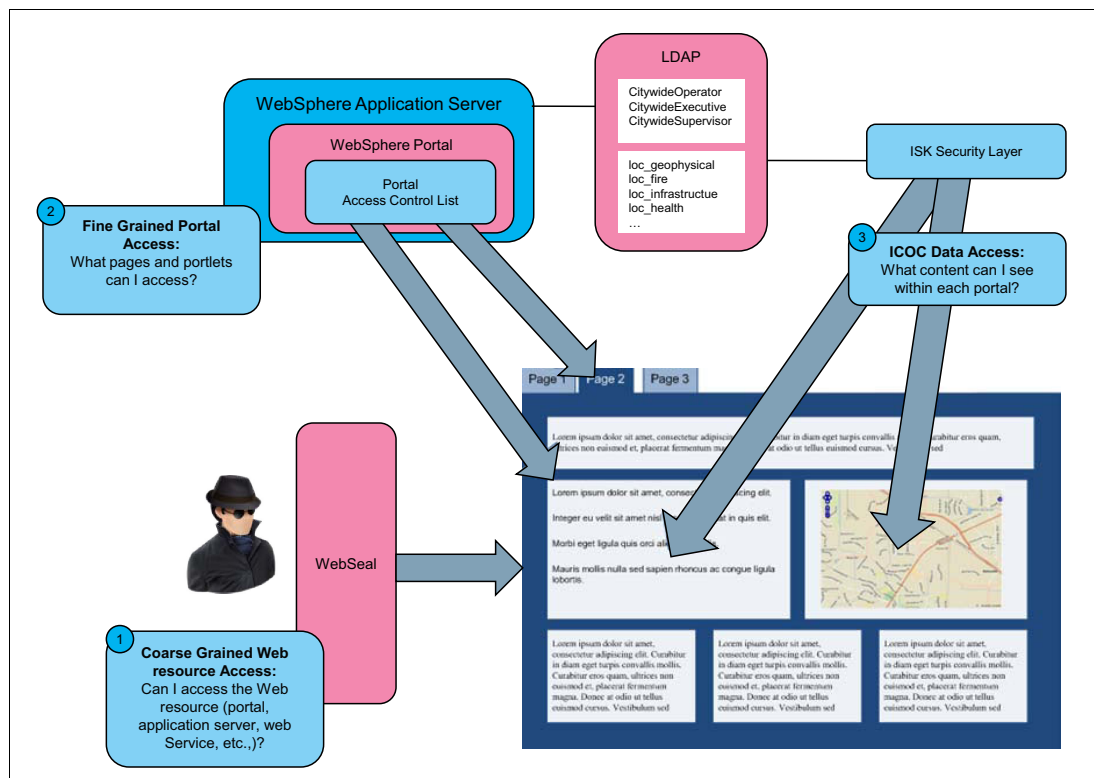
Figure 4-6   IBM Intelligent Operations Center security levels

As shown in Figure 4-6 on page 119, when a user attempts to access a resource, the authorization for that user is tested against all levels of granularity. A data item of interest might be embedded within a portlet, and that portlet is embedded within a portal, which is embedded on a page, and so on. The test is done from the most coarse-grained to the finest-grained level. At each level, the user must have the correct authority before access is provided. As authority is evaluated from one level to the next, the resulting access rights cannot become less restrictive. For example, the user can gain read/write access to a portal, then read-only access to a portlet within that portal, but the user cannot gain read/write access to a data item within that portlet.

IBM Intelligent Operations Center implements a role-based security system. Your client defines the security access to resources, based on the different roles or personas that will access the system. Each role is mapped to a corresponding group in the user registry. Access rights to objects and resources within the system are based on the user's role. It is possible for a user identity to be given multiple roles. Therefore, the user's access rights are the aggregate of all access rights that were given to all of the user's roles.

> **Note:** Roles are assigned to a user ID, not to a user.

Many possibilities exist when it comes to what a user can and cannot do in a role-based portal system. The following types of conditions apply with a role-based system:

► A user who logs in with an analyst role can only view and update data objects on a portal that is allowed for analysts.

► A supervisor has visibility to the task assignment portlet from which to view and update tasks that the supervisor's staff is working on.

► A user from the homicide department cannot view data that is intended only for users in the robbery department.

The core concept of a role-based system is summarized in Figure 4-7 on page 121. IBM Tivoli Access Manager is the main product that is used for managing these types of permissions, such as roles and resources. Checking for permissions can be done at the system level (where it is appropriate), or it can be controlled at the application level if that resource belongs to that IBM Tivoli Access Manager application. In either case, Tivoli Access Manager maintains these policies and can be queried for specifics.

User and group management is a fundamental capability in IBM Intelligent Operations Center where access permissions can be specified and managed as a separate portlet, which is only available for the administrator roles.
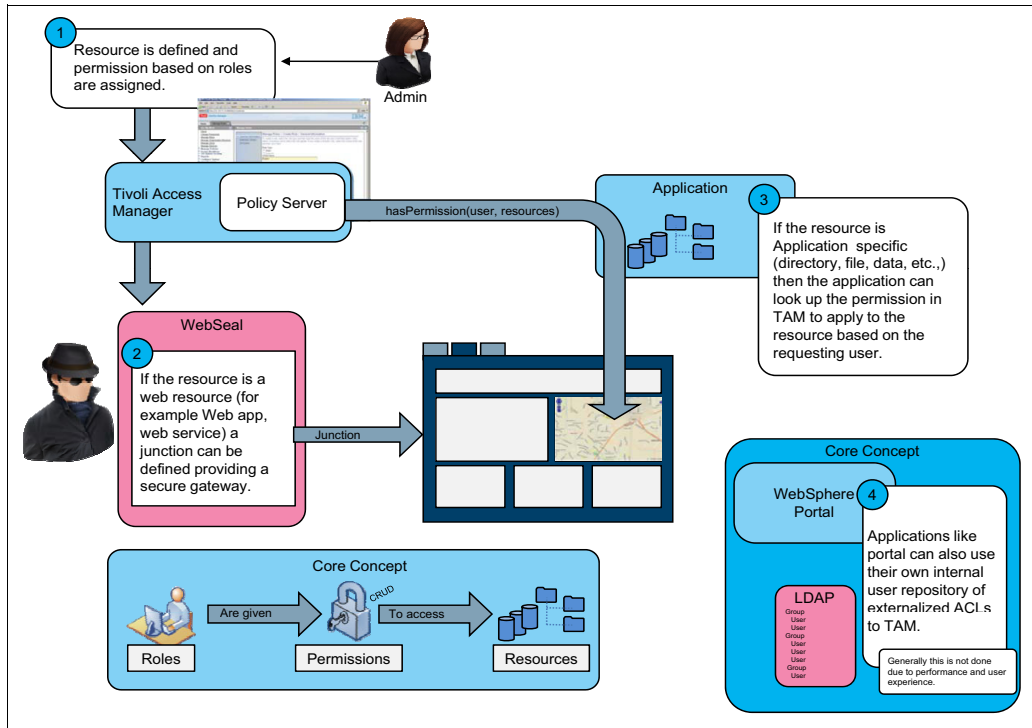
Figure 4-7   Role-based authorization model

4.2.3 IBM i2 COPLINK security model

The last product for discussion is IBM i2 COPLINK. The authentication model for this product is similar to the two previous products. The authorization model is simpler and it is all focused at the document level.

Authentication

The authentication process in i2 COPLINK can be done only using the traditional approach of username and password authentication. There are two possible user registries:

► The i2 COPLINK database user registry

► An LDAP-compliant third-party user registry product, for example, IBM Tivoli Directory Server

i2 COPLINK follows an organizational model based on agencies, that is, a number of agencies can be created in one COPLINK node (a *node* is an instance of an i2 COPLINK installation). A user can belong only to one agency. Each user is identified by specifying the agency and user name. For example, if user John Doe with username `jdoe` belongs to an agency called `MainAgency,` John will log in and choose `MainAgency` as the agency and enter the user name, `jdoe`. When users are challenged for their user name and password, they must enter *AgencyName/Username* if the user interface does not allow them to choose their agency and when the context is not clear about which agency is involved.

Authorization

The i2 COPLINK data model is based on *documents* from which objects within the document are created. Therefore, access control is enforced at the document level. Two concepts, *groups* and *access level,* govern the restrictions or access control imposed on documents:

► Group: A group is defined as a set of zero or more users. Each group has a unique name, such as `Gangs`, `Intelligence`, or `Homicide`. A default group in i2 COPLINK is named the *blank* group. All users of the system are members of the blank group. A document is always associated with a group; therefore, a group is sometimes referred to as a *document group*. By default, a document is associated with the blank group. Only members of the group to which a document is associated have complete access to that document.

► Access level: i2 COPLINK data is strictly read only. Therefore, the access level is defined in terms of the visibility of the document and its contents. Visibility is used to control access to a document by users who are not members of the group that the document is associated with. Each document is color-coded to represent the visibility of that document to the current user:

  – White: A document is white if it is visible, including all of its contents, to all users, regardless of group. White documents are equivalent to public records.

  – Gray: A document is gray if only the document as an entity is visible but not its contents to users outside the group. Gray documents are equivalent to the document being cloaked.

  – Black: A black document and its contents are hidden to users outside the group to which the document is associated.

The combination of both group and access level is necessary to correctly control access to a document, for example:

► Document X is associated with the External Correspondence group, and Document X is given White access, making it visible to all i2 COPLINK users. The White access level indicates that no restriction is required, even though Document X is associated with a specific group.

► Document Y must be visible only to members of the External Correspondence group. Therefore, the Black access level is given to Document Y.

► Document Z is associated with the External Correspondence group and the group members want to allow users outside of the External Correspondence group to know that Document Z exists. The contents, however, are to not visible to them. Therefore, Gray access is given to Document Z.

Figure 4-8 on page 123 is a decision tree that further explains how access control works in i2 COPLINK.
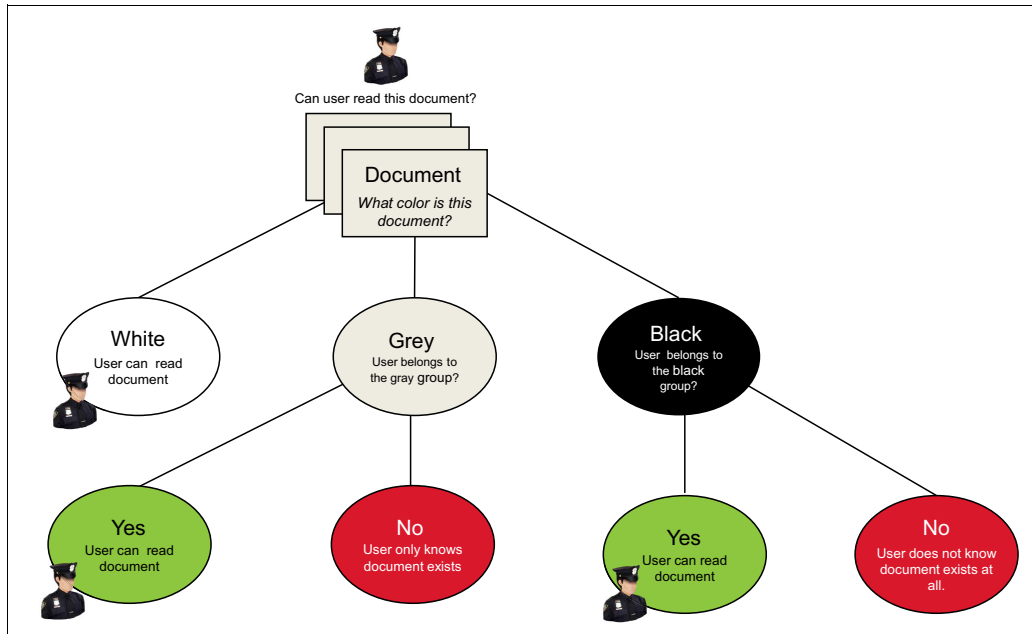
Figure 4-8   Determining user access to documents

Similar to the i2 Intelligence Analysis Platform cloaked access level, you can customize a message, also known as *sign-posting*, that is shown to a user who only has Gray access to a document. It means that you can attach a message to the document entity, for example, **"**`You do not have access to the contents of this document. Contact your management for more information.`**"** This message appears with the document to provide the users instructions about what to do if they really need access to the contents.

4.3 The global security model

So far, you have been introduced to the security models of the three product components in i2 Integrated Law Enforcement. For the integrated solution, what does it mean to integrate them from a security perspective? An integrated solution must have a well-defined security model that works in the global solution. This section is dedicated to addressing that topic.

4.3.1 Authentication

Three product components provide functionality and serve as entry points to i2 Integrated Law Enforcement:

► Policing
► Operations center portal
► Intelligence analysis

Each of these product components will continue to use the authentication method that you configured. However, if the transaction flow crosses to another product component (as part of the expected flow of the specific capability that is used), the credentials of the user must flow seamlessly to the other product.

The following requirements must be satisfied to implement a global security model for the solution:

► *The whole system uses a common user registry, so that any user ID that is presented to any of the product components always refers to the same person.*

   This point is important and crucial. An integrated solution must share a common set of users and groups that are uniquely identifiable globally. There are several approaches to achieve this requirement but the simplest and easiest way is for all the product components to use the same user registry as shown in Figure 4-9. This is the ideal solution. The user registry does not have to be LDAP-based.

► All updates to any user or group information are shared by all product components.

► The security model of a product component is used when the user transaction transitions to that product component.

► Single sign-on (SSO) is implemented when crossing from one product component to another.

Figure 4-9 shows the different entry points to the integrated solution. Depending on the entry point used, the corresponding authentication module performs the authentication function.
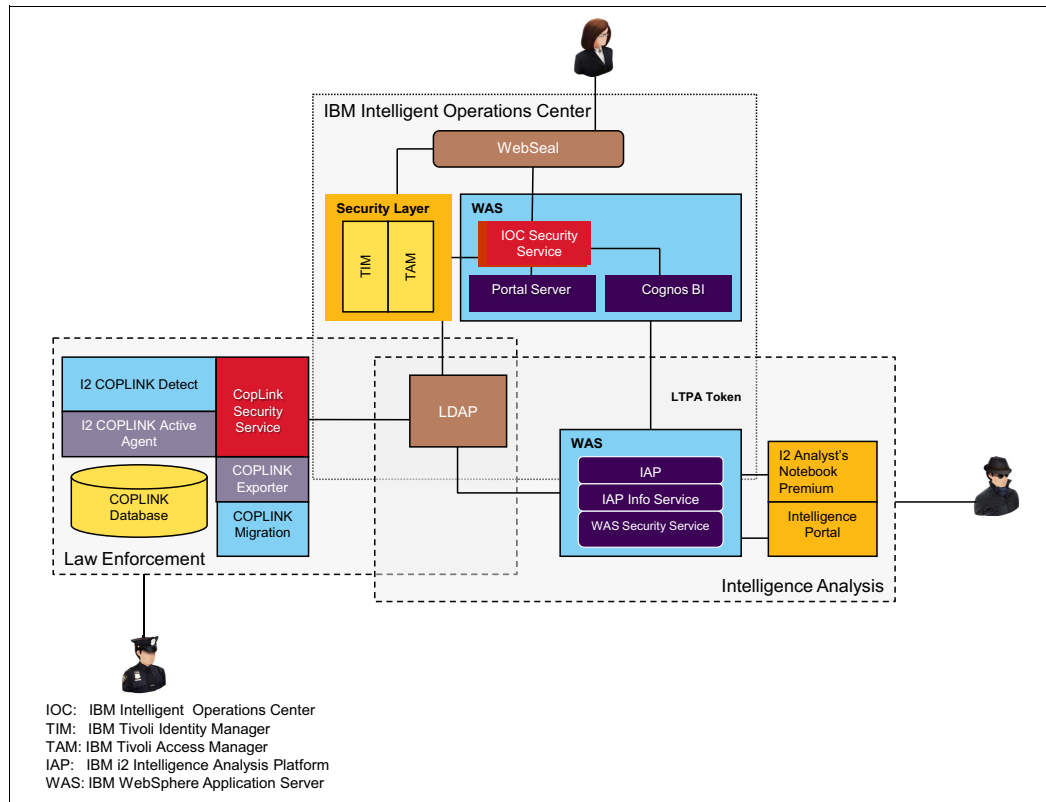
Figure 4-9   Using a common LDAP server as the ideal solution for global security

Ideally, the entire system uses a common user registry. In Table 2-4 on page 30, the list of LDAP products that are supported by each product component is provided. IBM Intelligent Operations Center only supports IBM Tivoli Directory Server. However, i2 COPLINK currently does not support IBM Tivoli Directory Server. This situation prevents you from having the ideal solution shown in Figure 4-9.

The next best configuration is to integrate with your client's existing user registry by replicating your client's user registry to the directory server of IBM Intelligent Operations Center. You can then configure i2 Intelligence Analysis Platform to use the user registry of IBM Intelligent Operations Center. The procedure to implement this configuration is described in Chapter 5, "Integrated Law Enforcement single sign-on" on page 163. Meanwhile, i2 COPLINK is configured to use your client's user registry directly. This solution is shown in Figure 4-10.

Note: Because of i2 COPLINK, you cannot support a client that uses IBM Tivoli Directory Server as their user registry.
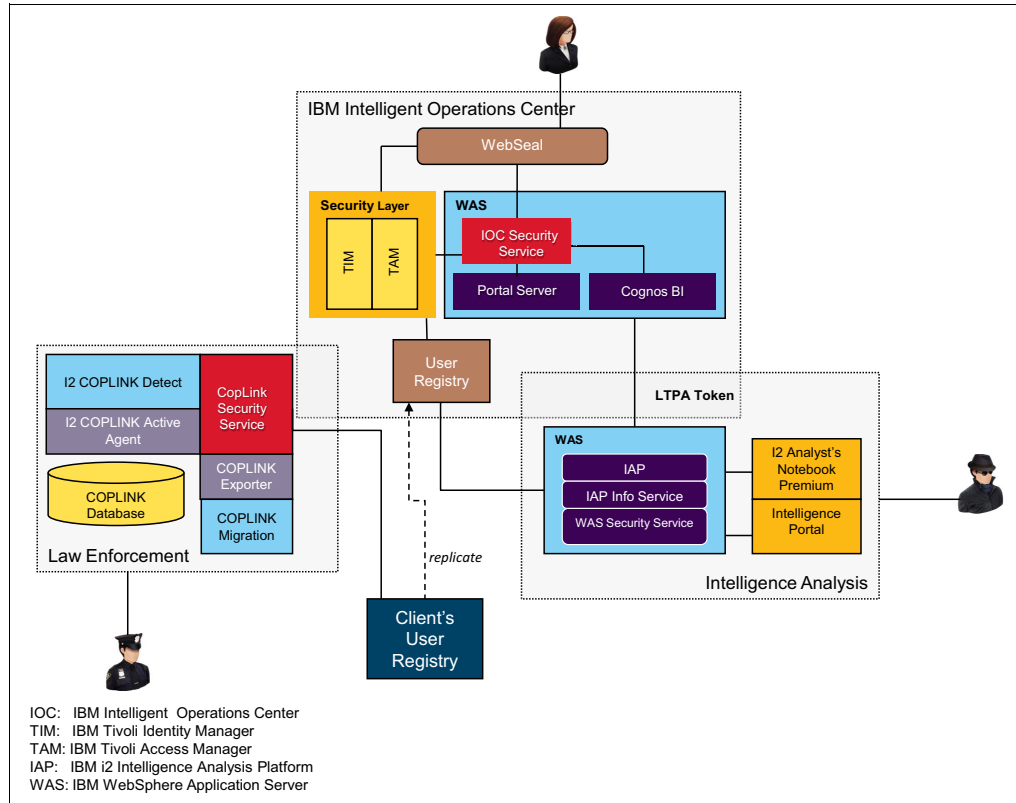
Figure 4-10 Recommended user registry configuration for global security

The replication approach effectively makes IBM Intelligent Operations Center appear as though it uses your client's user registry directly. The procedures for implementing this approach are provided in 4.4, "Integrating your client's user registry with IBM Intelligent Operations Center" on page 127.

Important: Prioritize this task because it is involved.

The only disadvantage of this approach is that there might be a window of time when the client's LDAP user registry is not consistent with the Tivoli Directory Server user registry. This issue can be addressed by making the replication schedule as reasonably often as possible. Another possible solution is to manually replicate any changes in the client's user registry to the Tivoli Directory Server user registry immediately after a change in the client's LDAP. The responsibility is then left to the system or the LDAP administrator.

4.3.2 Authorization

The scenarios that apply to global security are those where the user transaction moves from one product component to another. This situation is only possible in the extended capabilities. This section looks into each of the four extended capabilities that were introduced in 1.1.3, "Component architecture" on page 6 to explain how global security works for each of the extended capabilities.

Analysis Search

Analysis Search involves an analyst or crime investigator who uses i2 Analyst's Notebook Premium to search for data items that are stored in the COPLINK database. The current implementation for this capability is more *conservative* in that only white data from COPLINK can be shared and returned as search results. Recall that white data is essentially public data that is available to anyone in the organization. Therefore, after successful authentication, any user can access white data.

Intelligence Portal

The security model of IBM Intelligent Operations Center is role-based. Your client will likely limit access to the Intelligence Portal to specific roles, such as `analysts, investigators, detectives, police officers`, and `commanders`. Therefore, only users with these roles are authorized to access the Intelligence Portal. This approach can be implemented easily within IBM Intelligent Operations Center by creating, for example, an `Analyst` group in the directory server and giving permissions only to this group to access the Intelligence Portal page.

The credentials of a user authorized to access the Intelligence Portal are passed to i2 Intelligence Analysis Platform through an LTPA token. The username is retrieved from the token and from this point, the security schema defined within i2 Intelligence Analysis Platform is enforced based on the username. Users will only be able to access items for which they have the correct authority to access. And because i2 Intelligence Analysis Platform can point to and use the same user registry as the IBM Intelligent Operations Center directory server (further explained in Chapter 5, "Integrated Law Enforcement single sign-on" on page 163), the groups needed for i2 Intelligence Analysis Platform can be easily retrieved.

Reporting

Two types of reports are available in i2 Integrated Law Enforcement:

► Intelligence Analysis: For this report, the interaction occurs between IBM Cognos (running on the IBM Intelligent Operations Center application server) and the reporting database used by i2 Intelligence Analysis Platform.

   If a user wants to create a report, it is necessary for the user to be already authenticated by IBM Intelligent Operations Center. The user's credentials can be easily passed to IBM Cognos and then to i2 Intelligence Analysis Platform using the LTPA token.

   The data that gets into the report must be filtered out so that only items to which the user has at least read-only access are included. Authorized access to data is enforced by the security module of IBM i2 Intelligence Analysis Platform whenever an access request to an item with a specific user ID is submitted to it.

► COPLINK: Similarly, IBM Cognos is involved in this task. The user's identity can be passed through an LTPA token. When SQL queries are run against the COPLINK views, security checks at the row level are performed based on the user's identity. Therefore, it is accurate to describe that access control is done by the views.

Situational Awareness

Situational Awareness involves data from the i2 COPLINK database. As a restriction, only white data is exported as XML files, which are then imported into the message broker as events. The exporting of white data only is configured with i2 COPLINK Exporter. For more information, see 3.2.2, "Configuring IBM i2 COPLINK Exporter server" on page 50. The files are processed by IBM Tivoli Netcool/OMNIbus, and the results are stored in a data store in IBM Intelligent Operations Center.

Filtering occurs when the situational awareness UI (in this case, the map widget) requests the data set that you specified through the filter of the widget. The filtering that the user is allowed to specify in the map widget only includes the ability to limit the data based on the type of crimes and the time period.

Because only white data is exported by i2 COPLINK, global security is simplified because white data does not impose any restrictions about the data that can be viewed by the current user.

When the client requires that all types of data in i2 COPLINK can be exported, what happens to Situational Awareness? In this scenario, a contract needs to be created between you and the client to cover additional work to ensure that the security of data is not violated. Several approaches are possible but whatever approach you choose, ultimately, the data that is reported needs to be filtered out first before relaying the data to the map widget. The data set is passed to a module, including the user's credentials. The module must ensure that for each data object, the user has the correct authority to, at least, see the data. The security check is already an existing component of i2 COPLINK and therefore it can be used to perform this specific function within this module. However, i2 COPLINK does not have any public interface that can be used. The IBM i2 Lab Services team must be involved in this implementation.

For all of the extended capabilities, the common approach is to pass the current user's identity from one product component to another, therefore allowing the target product to apply its own security model. This approach works because all of the security models are dependent only on the user's identity to be able to make an assessment of the user's authorization on the resource in question. The authorization rules of that product component are applied to the specific user ID. The presumption is that the requester has the correct access to the resource on the front end, where all the results are rendered. Also, the use of a common security user registry, and therefore, the same security realm, is a large factor.

4.4 Integrating your client's user registry with IBM Intelligent Operations Center

IBM Intelligent Operations Center only supports Tivoli Directory Server for directory services. Many organizations and agencies already have an existing LDAP registry in place. These organizations frequently prefer to use their existing LDAP registry, rather than the IBM Tivoli Directory Server from Intelligent Operations Center. Several options might address this issue. One successful option is to *synchronize* the client's LDAP registry with the IBM Intelligent Operations Center registry on Tivoli Directory Server. IBM Security Directory Integrator (previously IBM Tivoli Directory Integrator) is used to synchronize the registries. For more information, see *IBM Security Directory Integrator* at this website:

http://www.ibm.com/software/products/en/directoryintegrator

4.4.1 LDAP synchronization solution overview

This section describes the preferred solution for integrating with a client's existing LDAP registry. Figure 4-11 summarizes this solution. The objective is to synchronize the client's LDAP registry with the IBM Intelligent Operations Center registry on IBM Tivoli Directory Server. When the integration is complete, users that interact with the IBM Intelligent Operations Center portal are authenticated against Tivoli Directory Server, which mirrors the client's LDAP registry. Synchronization needs to occur regularly so that any changes in the client's LDAP registry are reflected on Tivoli Directory Server.
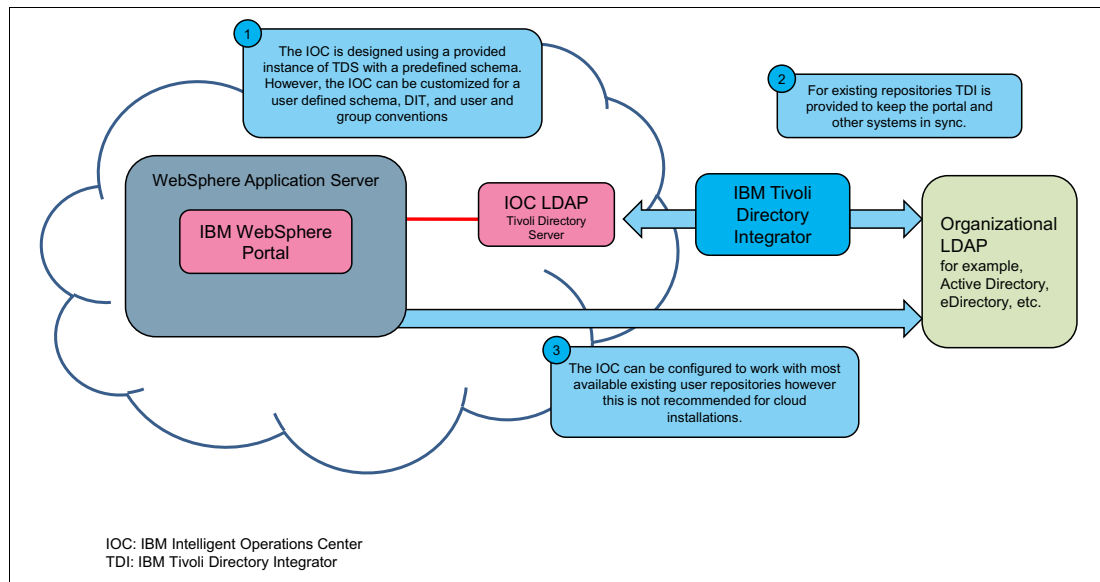
Figure 4-11   Using IBM Tivoli Directory Integrator to synchronize LDAP registries

4.4.2 Implementing the LDAP synchronization solution

This section describes how to provide user migration and synchronization from third-party directories, including Microsoft Active Directory, Oracle Directory Server, Sun Directory Server, and other LDAP directories, to the IBM Intelligent Operations Center user registry. It also describes how to allow users to authenticate to the third-party registry by using the IBM Intelligent Operations Center front end, without affecting the integration of the IBM Intelligent Operations Center solution. The following software prerequisites are needed for IBM Tivoli Directory Server and IBM Tivoli Directory Integrator:

► Tivoli Directory Server: IBM Intelligent Operations Center 1.5.x includes IBM Tivoli Directory Server Version 6.3.0.8, which needs to be upgraded to 6.3.0.23 or later for pass-through authentication to work. A minimum of Fix Pack 10 (6.3.0.10) is needed if you use a Tivoli Directory Server proxy server. See the readme file documentation for Fix Pack 10: IO15626: *Pass Through Authentication (PTA) with Proxy or compare op*:

  http://www-01.ibm.com/support/docview.wss?uid=swg1IO15626

► Tivoli Directory Integrator: IBM Intelligent Operations Center V1.5.x includes Tivoli Directory Integrator Identity Edition V7.1.0.5. For the solution to work, you need to install IBM Tivoli Directory Integrator Identity Edition V7.1.1 with Fix Pack 2 with the following support patches:

  – 7.1.1-TIV-TDI-LA0007
  – 7.1.1-TIV-TDI-LA0009

Note:

► If Fix Pack 2 is no longer available, download the latest fix pack after Fix Pack 2.
► Tivoli Directory Integrator V7.1.1 is a supported version with Intelligent Operations Center V1.5.x.

4.4.3 Installing Tivoli Directory Integrator

This section describes the process to install Tivoli Directory Integrator on the IBM Intelligent Operations Center V1.5 management server. The following installation steps are from the Tivoli Directory Integrator installation documentation. For more information, see *Installing IBM Tivoli Directory Integrator* at this website:

`http://www.ibm.com/support/knowledgecenter/SSCQGF_7.1.1/com.ibm.IBMDI.doc_7.1.1/adminguide12.htm`

Multiple instances of Tivoli Directory Integrator can be installed on the same machine, provided they are installed in two solution directories and different ports are assigned. Because Tivoli Directory Integrator 7.1.0.5 is already installed, the simplest method is to install the new version as a separate instance and not migrate any of the solutions used in the original instance. Make note of the *new instance solution directory* because it will be used to deploy the LDAPSync solution. Perform the following steps:

1. Download IBM Tivoli Directory Integrator 7.1.1 from IBM PartnerWorld® or IBM Passport Advantage Online.

2. Access the Tivoli Directory Integrator installation instructions at this website:

   `http://publib.boulder.ibm.com/infocenter/tivihelp/v2r1/index.jsp?topic=%2Fcom.ibm.IBMDI.doc_7.1.1%2Fadminguide12.htm`

3. Download Tivoli Directory Integrator 7.1.1 Fix Pack 2 from this website in preparation to installing Fix Pack 2. If Fix Pack 2 is no longer available, download the latest fix pack after Fix Pack 2:

   `http://www-01.ibm.com/support/docview.wss?uid=swg27010509#ver711`

   Note: Before you install Fix Pack 2, be sure to patch your Tivoli Directory Integrator Update Installer so that Fix Pack 2 is applied correctly. If the Update Installer patch is not applied, the Configuration Editor (CE) will fail to launch after the `<TDI_install_dir>\bin\applyupdate -rollback` command is run.

   To patch the Tivoli Directory Integrator Update Installer with Patch 7.1.1-TIV-TDI-LA0007, perform the following steps:

   a. Take the file `UpdateInstaller.jar` from the fix pack.

   b. Replace the copy in the `<TDI_install_dir>/maintenance directory`.

   c. Export `DISPLAY=localhost:0.0.`

   d. Run the following command:

      `TDI_install_dir/bin/applyUpdates.sh -update TDI-7.1.1-FP0002.zip [-clean [-silent]]`

   e. Shut down Tivoli Directory Integrator.

To patch the Tivoli Directory Integrator Update Installer with Patch 7.1.1-TIV-TDI-LA0009, perform the following steps:

a. Extract the fix pack to a temporary directory. The archive file contains the `miserver.jar` and `ibmjs.jar` files.

b. Back up the following files by renaming them (change the extension to something other than `.jar` or `.zip`) and keeping them in their own folders, for example:

- `miserver.jar` (`<TDI_Install_dir>\jars\common`)

- `ibmjs.jar` (`<TDI_Install_dir>\jars\3rdparty\IBM`)

- `LDAPConnector.jar` (`<TDI_Install_dir>\jars\connectors`) from the currently installed system

4. Move the files from the fix package to the correct folder.

5. Replace the existing `miserver.jar`, `ibmjs.jar`, and `LDAPConnector.jar` files, which were backed up earlier with the fix file.

4.4.4 Installing LDAPSync

LDAPSync is a Tivoli Directory Integrator solution that simplifies migrating and synchronizing from Oracle or Microsoft Active Directory LDAP servers to Tivoli Directory Server. LDAPSync can be downloaded from the IBM Integrated Service Management Library at this website:

http://www.ibm.com/software/brandcatalog/ismlibrary/details?catalog.label=1TW10DIOW#tab-pinfo

Follow these steps to install LDAPSync:

1. Download the `LDAPSync.zip` file and then extract the contents into the Tivoli Directory Integrator directory on the IBM Intelligent Operations Center management server.

Special Notes:

► Connecting to the LDAP store with Tivoli Directory Integrator. Your client might be concerned about allowing connections to their LDAP server. LDAP stores, such as Oracle Internet Directory and Microsoft Active Directory, provide, as a standard feature, read access through port 389. Therefore, this access is already available in their network. *Requesting access for connecting IBM Intelligent Operations Center to their LDAP store is not introducing a new vulnerability.*

► Handling service accounts. Service accounts are required for managing various components of IBM Intelligent Operations Center, but they do not function as actual users of the system. Many service account entries are created in Tivoli Directory Server as part of the IBM Intelligent Operations Center installation (for example, `wasadmin`, `notesadmin`, and `taiuser`). Your client might require these service accounts to exist in their primary LDAP store. If these service accounts are not required, they can be retained on the Tivoli Directory Server and managed in IBM Intelligent Operations Center. If the client requires the accounts to be in their LDAP store, the client needs to create them there, with the passwords.

If service accounts are set during the IBM Intelligent Operations Center configuration with a local Tivoli Directory Server userpassword attribute value, that attribute is used rather than the Pass Through Authentication (PTA) mechanism. To fully use PTA and maintain the password in Microsoft Active Directory, the userpassword attribute must be removed from the Tivoli Directory Server account entries.

---

2. Set up the LDAPSync solution on the Tivoli Directory Integrator. Perform the following steps:

   a. Deploy the solution

   To install LDAPSync, extract the `LDAPSync.zip` file to the Tivoli Directory Integrator solution directory in the IBM Intelligent Operations Center management server. The content of the package is shown in Table 4-1.

   Table 4-1   File contents of LDAPSync.zip

File name	Description
`LDAPSync.xml`	The Tivoli Directory Integrator configuration file, which is loaded and run by a Tivoli Directory Integrator server.
`LDAPSync.properties`	A text file that contains connection and solution properties. These properties are described in Table 4-2 on page 131, Table 4-3 on page 133, and Table 4-4 on page 135.
`person.map`	Mapping file for User entries.
`group.map`	Mapping file for Group entries.
`organizationalunit.map`	Mapping file for organizationalUnit container entries.
`organization.map`	Mapping file for organization container entries.
`dcobject.map`	Mapping file for dcObject container entries.
`country.map`	Mapping file for country container entries.
`customScript.js`	Can contain script functions and variables to be used in the mapping files listed above.

   b. Configure the properties file. Follow these two steps:

   i. Set the various properties that are in the `LDAPSync.properties` file that is shown in Table 4-2, Table 4-3 on page 133, and Table 4-4 on page 135.

   ii. If necessary, adjust the mapping rules for the `Person`, `Group`, and `Container` entries, as defined in the files named `*.map`. If custom JavaScript functions will be used in a mapping file, they are added to the `customScript.js` file.

The property file contains settings that control the connection to the source and target systems in addition to the behavior of the solution. A list of properties is shown in Table 4-2, Table 4-3 on page 133, and Table 4-4 on page 135. The properties are organized into global (Table 4-2), source (Table 4-3 on page 133), and target (Table 4-4 on page 135) properties.

You can edit the property file by either directly editing the file or by using the Tivoli Directory Integrator configuration editor.

Table 4-2   Global properties in the LDAPSynch property file

Property name (global)	Description
Simulate	This property enables you to simulate a migration, providing a report about what happens in an actual migration. If this property is set to `true`, no data is written to the target system. Set this property to `false` to perform the actual migration. **Note**: If the command-line argument **-0** (zero) is used with a value of either `simulate` or `actual`, the value set for the simulate property is ignored. The default value is `false`.

Property name (global)	Description
global.preserveSourceContainers	This property controls the behavior of distinguished name *DN*-translation for the solution. If the value is `true`, container hierarchies in the source directory under the base suffix specified will be mirrored in the target directory. If the value is `false`, the `target.suffixForUsers` will be used as the container for Person entries that are written to the target, and `target.suffixForGroups` specifies this for Group entries. **Note**: Suffix nodes specified by these parameters *must exist* in the target directory. **Note**: This property can be specified per Flow or Endpoint, if required. The default value is `false`.
global.logDirectory	The directory path where the solution log files are created. The default value of this parameter is the relative path: `LDAPSync/logs/`. As a result, all solution log files are created in the `logs` path of the LDAPSync folder, which is under the solution directory.
global.maxLogFiles	The number of files kept in the rolling history of log files created. The default value is `20`.
global.showProgressCount	The number of entries processed before logging a progress message. For example, if this property is set to `250`, this causes a progress message to be logged for every 250 entries that are processed. If this value is set to `0` or not set (the value is empty), no progress messages are logged. **Note**: This property can be specified per Flow or Endpoint, if required. The default value is `empty`.
global.flows	An optional parameter for defining the IDs for several separate migration or synchronization flows. If the solution is configured to handle multiple data flows (for example, from more than one source system, or targeting multiple LDAP servers or DIT subtrees), this property is used to name these flows. For example, in the value AD1, AD2, two flow IDs are defined. Any other properties that are prefixed with `<FlowId>`, you must apply only to the specified flow. Properties with no flow qualifier must apply to all flows that do not have their own property setting. **Note**: Flow IDs are case-sensitive and must be spelled the same as they are here when they are prefixed to other properties. Log files will be prefixed with the Flow ID and an underscore character (_). The value `ep` is reserved and must not be used for a Flow ID. The default value is blank, which means that there is a single unnamed Flow from the source system to the target. Logs created during migration are prefixed with `M_` and those logs made during synchronization start with `S_`.

Table 4-3 on page 133 shows the properties that control the connection and handling of the source directory. All source properties can be specified *per Endpoint* or *per Flow*, if required. If specified for an Endpoint, `source` must be removed from the property name.

Table 4-3   Source properties in the LDAPSynch property file

Property name (source)	Description
source.ldap.user	*Username* that will connect to the source system.
source.ldap.password	*Password* associated with the source LDAP *username* that is mentioned in `source.ldap.user`.
source.ldap.searchBase	The distinguished name of the node in the source directory under which entries are read for migration and synchronization. **Note**: For Microsoft Active Directory, set this value to the `root` suffix of the Microsoft Active Directory database directory information tree (DIT). Otherwise, delete modifications will not be detected. The value of this property also affects the distinguished name (DN) translation.
source.container.objectClasses	A comma-separated list of the container object classes to migrate. The default value is `ou=organizationalUnit, dc=dcObject, c=country, o=organization`. **Note**: This property only has an effect if the `global.preserveSourceContainers` property is set to `true`.
source.containersToMigrate	A semicolon-separated list of containers, under which the entries are migrated and synchronized. **Note**: Each container specification can be listed as either just the relative distinguished name (RDN) to the container, or it can be part of the complete distinguished name (DN) value for that entry. The default value is `ou=Groups;ou=People`. **Note**: This property only has an effect if the `global.preserveSourceContainers` property is set to `true`.
source.containersToSkip	The semicolon-separated list of containers (string), which are *not* to be migrated or synchronized. This list is applied after first checking the entry DN against the list in the `source.containersToMigrate` property. **Note**: Each container specification can be listed as either just the RDN to the container, or it can be part of the complete DN value for that entry. The default value is `ou=Groups;ou=People`. **Note**: This property only has an effect if the `global.preserveSourceContainers` property is set to `true`.
source.userObjectClass	The objectClass used to identify person entries in the source directory. The default value is `person`.
source.groupObjectClass	The objectClass used to identify group entries in the source directory. The default value is `groupOfUniqueNames`.
source.ldap.pageSize	An optional property for those systems that support paged search returns and that limit the size of search returns, such as Microsoft Active Directory. To iterate over the entire directory, the search page size must be set to a value less than the Sizelimit/Admin Limit of the LDAP Server. The default value is `blank`.

Property name (source)	Description
source.ldap.binaryAttributes	Use this property to specify binary attributes that need to be handled by the solution. The default value is `blank`.

Note: This is in addition to the standard `inetOrgPerson` binary attributes.

Note: When using Microsoft Active Directory as the source, specify the `objectGUID` binary attribute in this property. |
| **source.changeDetectionType** | This mandatory property defines the change detection mechanism that is used by LDAPSync. The following settings are valid:
► Sun
► TIVOLI DIRECTORY SERVER
► AD

Any other value results in an error. The default value is `blank`.

Note: The Sun setting can be used with other retro changelog compatible LDAP directories, such as iPlanet and OpenLDAP. |
| **source.userRDN** | The attribute mapped to the RDN for Person entries in the target. If not specified, this is the same as the `target.userRDN` property value. |
| **source.ad.searchBase** | This parameter is used in the Microsoft Active Directory Change Detection Connector, which feeds the `runADSync` AssemblyLine, which is launched by the LDAPSync AssemblyLine. Set this parameter to the root suffix of the Microsoft Active Directory database DIT to ensure that the Active Directory Change Detection Connector sees the `CN=Deleted Objects` container and the delete changes are detected.

Note: The `source.ldap.searchBase` property is still used, and it is intended to reference the container under which the entries will be migrated or synchronized.

Note: For the Microsoft Active Directory Change Detection Connector to process deleted entries, you must first configure the Active Directory domain controller as specified by Microsoft at `http://support.microsoft.com/kb/258310`. The default value is `blank`. |
| **source.ldap.useNotifications** | If set to `true`, the Changelog Connector in the LDAPSync AssemblyLine subscribes to change notifications in the source directory. The default value is `true`.

Note: If this property is set to `true`, the next two properties do not have any effect. |

Property name (source)	Description
source.ldap.secondsForPolling	The number of seconds that the Changelog Connector in the LDAPSync AssemblyLine waits between polling for changes in the source directory. The default value is `10`. **Note**: This property does not have any effect if the `source.ldap.useNotifications` is set to `true`. **Note**: The value of this property must be less than the `source.ldap.changelogTimeout` setting.
source.ldap.changelogTimeout	The number of seconds that the Changelog Connector in the LDAPSync AssemblyLine waits for new changes to occur in the source directory. The default value is `1800`. **Note**: This property does not have any effect if the `source.ldap.useNotifications` is set to `true`. **Note**: The value of this property must be greater than the value of the `source.ldap.secondsForPolling` setting.
source.ldap.stateKey	The change detection iterator state is stored using this key value. *It is important that each change detection connector has its own state key*. The default value is `SourceSyncState`. **Note**: The `Flow ID` is added as a prefix to the state key value to ensure that each flow has a unique change state.

Connection and handling of the target directory is based on the settings of the properties in Table 4-4. All source properties can be specified per Endpoint or per Flow, if required. If specified for an Endpoint, `target` must be removed from the property name.

Table 4-4   Target properties in LDAPSynch.properties

Property name (target)	Description
target.ldap.url	LDAP URL to the target system. **Note**: If Secure Sockets Layer (SSL) is required, the protocol specifier used needs to be `ldaps://` instead of `ldap://`. Also, for an SSL connection, the client certificate must be imported to the Tivoli Directory Integrator keystore.
target.ldap.user	Username for connecting to the target system.
target.ldap.password	Password associated with the `target.ldap.user` username.
target.ldap.searchBase	The root suffix of the target LDAP directory. **Note**: If `global.preserveSourceContainers` is `true`, the `target.ldap.searchBase` property specifies the target container where the container hierarchies that are found in the source are written.
target.ldap.binaryAttributes	Use this property to specify the binary attributes that need to be handled by the solution. The default value is `blank`. **Note**: This is in addition to the standard `inetOrgPerson` binary attributes.

Property name (target)	Description
target.userObjectClass	Enter the object class in the clause "Object class of _____" to be used for creating user entries in the target directory, for example, `inetOrgPerson`. **Note**: This can be a comma-separated list of object class names.
target.userRDN	The attribute used as the RDN for Person entries.
target.groupObjectClass	The object class to use when creating Group entries in the target directory, for example, `groupOfUniqueNames`. **Note**: This can be a comma-separated list of object class names.
target.groupMemberAttribute	Name of the attribute in Group entries in the target directory that is used to store the DN of member entries, for example, `uniqueMember` or `member`. The default value is `uniqueMember`.
target.suffixForUsers	If `global.preserveSourceContainers` is `false`, this property is required. This property is used as the suffix appended to the DN of Person entries in the target system. **Note**: If `global.preserveSourceContainers` is `true`, the `target.ldap.searchBase` property specifies the container in the target system under which the container hierarchies that are found in the source are written.
target.suffixForGroups	If `global.preserveSourceContainers` is `false`, then this property is required. This property is used as the suffix appended to the DN of Group entries in the target system. **Note**: If `global.preserveSourceContainers` is `true`, then the `target.ldap.searchBase` property specifies the container in the target system under which the container hierarchies that are found in the source are written.
target.*\<entry type\>*.mapFile	An optional parameter for specifying the map file to use for a particular type of entry; the valid entries are shown: ▸ Person: For Person entries ▸ Group: For Group entries For containers, the *\<entry type\>* is the lowercase objectclass name, for example, `target.dcobject.mapFile`. If no path is specified, files are assumed to be in the LDAPSync subfolder of the solution directory.

c. Test the connections.

After you configure the properties, test connectivity by running the following command in the Tivoli Directory Integrator installation folder:

`ibmdisrv –c LDAPSync/LDAPSync.xml –r TestConnections`

If an error message displays, verify the property settings.

If running this command is successful, the messages shown in Figure 4-12 on page 137 in are displayed.

```
Initializing Source LDAP
...performing search
...reading data
Initializing Target LDAP
...performing search
...reading data
Initializing Source LDAP Changelog
...performing search
...reading data
SUCCESS! All three worked correctly!
```

Figure 4-12   Test connections successful messages

d. Perform the migration.

The migration is performed in two phases:

i. Perform a simulated migration.

Perform a simulation first to test the data. To perform a simulated migration, either set the `simulate` property to `true`, or use the **-0** (zero) command-line argument when you run the **LDAPMigrate** operation. Start the migration running **ibmdisrv.sh**, which is in the Tivoli Directory Integrator installation folder. To launch an assembly line with this command, specify the solution configuration file using the **-c** argument and the assembly line to run with the **-r** argument:

ibmdisrv -c LDAPSync/LDAPSync.xml -r LDAPMigrate -0 simulate

The LDAPMigrate AssemblyLine displays a run report upon completion.

ii. Perform the actual migration.

When the results of the simulated migration are correct, perform the actual migration.

Run the **LDAPMigrate** operation again to perform the actual migration. Specify the solution configuration file using the **-c** argument and the assembly line to run with the **-r** argument:

ibmdisrv -c LDAPSync/LDAPSync.xml -r LDAPMigrate -0 actual

Figure 4-13 on page 138 shows an example of the results when you run this command for a successful migration.

```
> migrating people entries using search filter:
(objectClass=inetOrgPerson)
Processing #50 currently working on entry of type: person
Processing #100 currently working on entry of type: person
Processing #150 currently working on entry of type: person
> migrating group entries using search filter:
(objectClass=groupofuniquenames)
========= Summary =========
People Entries -- Processed: 155 Added: 0 Modified: 0 Deleted: 0
Groups -- Processed: 4 Added: 0 Modified: 0 Deleted: 0
Containers -- Processed: 2 Added: 0 Modified: 0 Deleted: 0
--------------
Errors: 0
Warnings: 0
```

Figure 4-13   Performing a simulated migration: Results

e. Create a schedule.

You can create a schedule using the Schedules feature in Tivoli Directory Integrator. The Schedules feature is in the Tivoli Directory Integrator web-based dashboard user interface. This monitoring and administration interface enables you to specify schedules for your Assembly Lines that the Tivoli Directory Integrator server will use to control when your assembly lines run. To create a schedule, launch the dashboard. Ensure that the Tivoli Directory Integrator server is running and the LDAPSync solution is loaded. Several ways are available to launch the dashboard; however, the quickest route is to load the solution when you start the Tivoli Directory Integrator server.

Follow these steps:

i. Run the following command:

   `ibmdisrv -c LDAPSync/LDAPSync.xml -d`

   The **-d** command-line option is important. Without this option, the Tivoli Directory Integrator server starts, detects that no Assembly Lines are running, and then shuts down again. Using the **-d** option instructs the Tivoli Directory Integrator server to continue running even when no Assembly Lines are active. The **-c** option causes the LDAPSync solution to be loaded, making it ready to work in the dashboard.

ii. Point your browser to http://*localhost*:(port)/dashboard. Replace *localhost* with the *hostname* or IP address of the machine running the Tivoli Directory Integrator server. The port default is 1098.

iii. From the Tivoli Directory Integrator dashboard, select the solution and the LDAPSync AssemblyLine and click **Create Schedule** to create the schedule for this AssemblyLine. See Figure 4-14 on page 139.
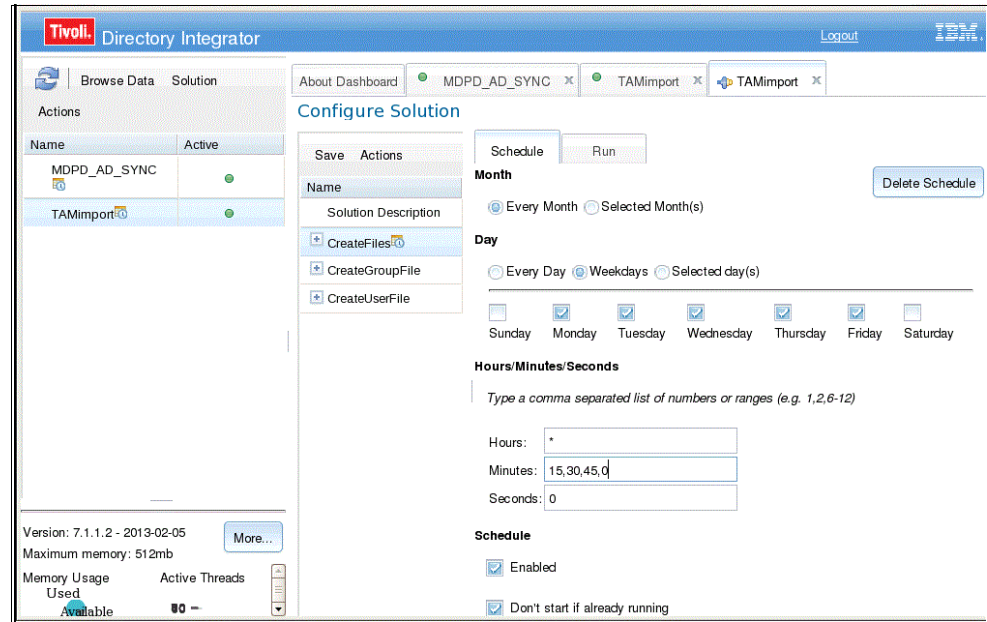
Figure 4-14   Schedule panel

4.4.5 Configuring pass-through authentication

This section describes enabling pass-through authentication in Tivoli Directory Server, which must be already configured. To configure pass-through authentication, use one of the following methods:

► Web Administration Tool
► Command line and then configure IBM Tivoli Access Manager

Using the Tivoli Directory Server Web Administration Tool

The following steps describe how to configure pass-through authentication using the Tivoli Directory Server Web Administration Tool:

1. Log in to the Web Administration Tool from a browser using the URL `http://<IOC Management Server>:9081/IDSWebApp/` where `<IOC Management Server>` is the *hostname* of the IBM Intelligent Operations Center management server.

2. Select the *hostname* (for example, `<IOC Management Server>:389`) as the LDAP server name.

3. Log in as an administrator, for example, `cn=root`, and enter the corresponding password.

    Figure 4-15 on page 140 shows the administrator logging in to the Web Administration Tool.

Figure 4-15   Logging in to the Tivoli Directory Server Web Administration Tool

4. Expand the **Manage security properties** category under Server administration in the navigation area of the Web Administration Tool.

5. Click the **Pass-through authentication** tab.

6. Enable or disable pass-through authentication by selecting or clearing the **Enable pass-through authentication** check box.

7. In the Pass-through authentication panel, click **Add**.

8. In the Subtree settings panel, click **Browse**. Select **ou=swg,o=ibm,c=us** and click **Select**. See Figure 4-16 on page 141.
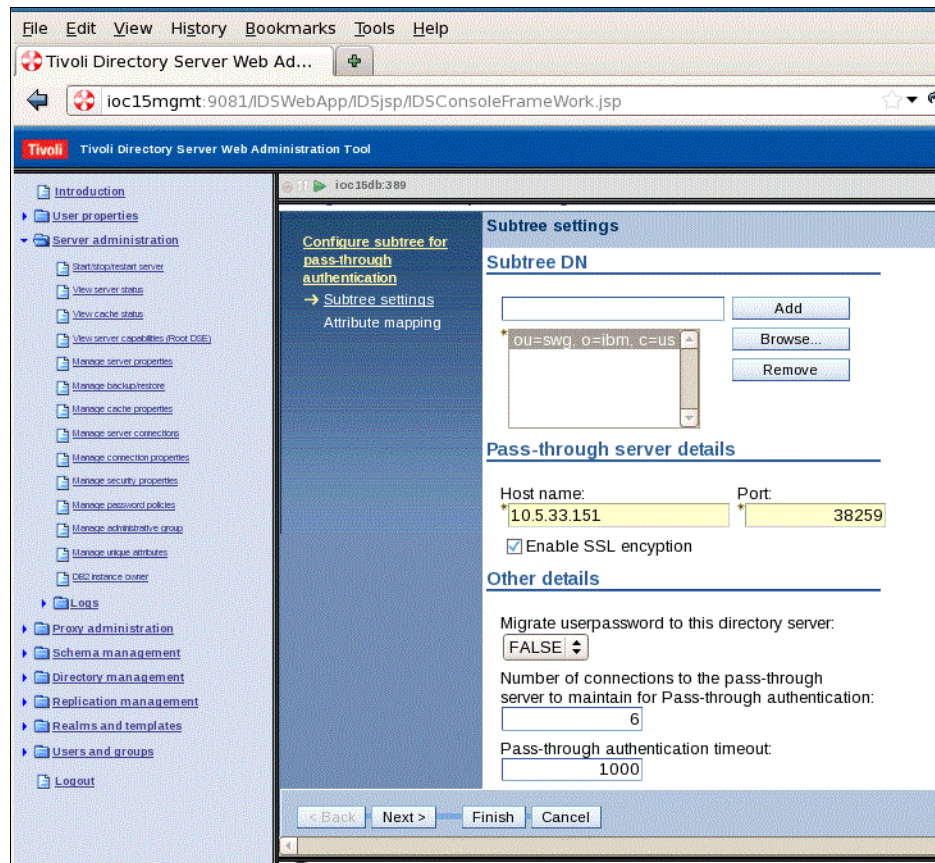
Figure 4-16 Subtree settings for pass-through server

9. Specify the host name of the pass-through server in the Host name field. The pass-through server is the other registry server from which users are being pulled.

10. Specify the port number of the pass-through server in the Port field.

11. If necessary, enable SSL encryption on the pass-through server by selecting the **Enable SSL encryption** check box.

12. Accept the default value (**false**) for the option to save the user password on the local directory for all successful bind requests that are processed through the pass-through server. If you prefer to change this value, select the value from the "Migrate userpassword to this directory server" combination box.

13. Specify the number of connections that are required for each pass-through server entry in the "Number of connections to the pass-through server to maintain for Pass-through authentication" field.

14. Specify a timeout value in the Pass-through authentication timeout field. The pass-through authentication interface will wait for results from the socket until the timeout period before it returns the client request.

15. Click **Next**.

16. Select the **Enable attribute mapping** check box to enable attribute mapping. Selecting this check box also enables other controls on the Attribute mapping panel.

17. In the Bind DN for pass-through server field, enter a bind DN for binding to the pass-through server, for example, `cn=icpo_bind_user,cn=users,dc=boston,dc=cob`.

18. In the Bind password for pass-through server field, enter a bind password for binding to the pass-through server.

19. In the Search base DN field, enter the search base DN of the pass-through server on which the entry will be searched. Alternatively, click Browse to display the Browse entries panel from which you can select the existing DN from the pass-through server.

20. From the Attribute for this directory server combination box, select an attribute to map to an attribute on the pass-through server, for example, **uid**.

21. From the Attribute for pass-through directory server combination box, select an attribute to map to the Tivoli Directory Server attribute. For another LDAP, this can be the **uid**.

    Figure 4-17 shows the Attribute mapping panel.
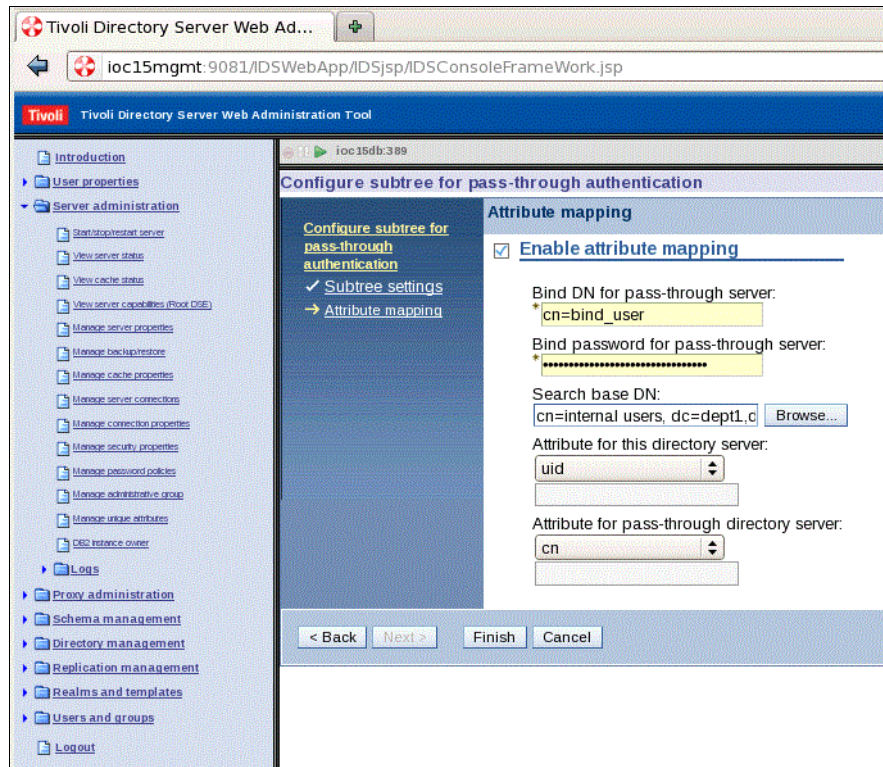
Figure 4-17   Attribute mapping for the pass-through directory server

22. Click **Finish** to save the changes and to navigate to pass-through authentication.

23. Click **OK** or Apply.

Using the Tivoli Directory Server command line

As an alternative to using the Tivoli Directory Server Web Administration Tool GUI, pass-through authentication and attribute mapping can be configured on the Tivoli Directory Server by using an LDAP Data Interchange Format (LDIF) file, for example, SetUpPassThrough.ldif. See Figure 4-18 on page 143.

```
dn: cn=Configuration
ibm-slapdPtaEnabled: true

dn: cn=Passthrough Server1,cn=Passthrough Authentication,cn=Configuration
cn: Passthrough Server1
ibm-slapdPtaURL: ldap://<server name>:389
ibm-slapdPtaSubtree: ou=users,ou=swg,o=ibm,c=us
ibm-slapdPtaMigratePwd: false
ibm-slapdPtaAttrMapping: uid $ samAccountName
ibm-slapdPtaSearchBase: <Pass Through Server DN>
ibm-slapdPtaBindDN: <Read-only user DN>
ibm-slapdPtabindPW: <Read-only user password>
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdPta
objectclass: ibm-slapdPtaExt
```

Figure 4-18   Sample LDIF file

The sample LDIF file shown in Figure 4-18 includes the following parameters:

ibm-slapdPtaURL   The *hostname* and *port* of the pass-through server (for example, the Microsoft Active Directory server). You can use SSL, if necessary, and the Microsoft Active Directory Global Catalog address can be used, that is, it can be routed through Global Catalog port 3268.

ibm-slapdPtaAttrMapping

The attribute mapping between the Tivoli Directory Server and its equivalent in the pass-through authentication server, for example, Microsoft Active Directory. In the sample LDIF file, this is the uid attribute on Tivoli Directory Server and the samAccountName attribute in Microsoft Active Directory.
Note: A 20-character maximum length for the Active Directory samAccountName attribute must considered. The samAccountName attribute value is unique within a domain.

ibm-slapdPtaSubtree   The subtree container definition on the Tivoli Directory Server for locating the entries for the bind request. An example definition is shown in the LDIF file in the example in Figure 4-18.

ibm-slapdPtaSearchBase

The container Distinguished Name in the pass-through server in which user entries are located, for example, a Microsoft Active Directory search base.

ibm-slapdPtaBindDN   The DN account entry, used to connect to and read from the pass-through server, for example, a Microsoft Active Directory user.

ibm-slapdPtaBindPW   The password for the DN entry, used to connect to and read from the pass-through server.

To load the LDIF configuration, run the following command on the IBM Intelligent Operations Center Tivoli Directory Server instance, and recycle the Tivoli Directory Server instance. The example command assumes that the Tivoli Directory Server administrator DN is `cn=root` and that the Tivoli Directory Server Instance port is the standard `389`:

```
idsldapmodify -D cn=root -w <cn=root pwd> -p 389 -f SetUpPassThrough.ldif
```

Configuring pass-through authentication with Tivoli Access Manager

Applications, such as IBM Tivoli Access Manager, by default, authenticate users to the Tivoli Directory Server registry using a credential comparison. This is set using the parameter `auth-using-compare = yes` in the configuration files for the policy server and its clients (for example, `ldap.conf`, `ivmgrd.conf`, and `webseald-<instance name>.conf`).

This comparison method requires that the `userpassword` attribute is present in the directory. However, for pass-through authentication, the `userpassword` attribute is not stored locally on the directory server. Therefore, the `userpassword` attribute must be removed for any pre-existing Tivoli Directory Server user entries, therefore allowing Microsoft Active Directory to provide the authentication.

The parameter `auth-using-compare = no` must be set on the Tivoli Access Manager servers and a bind operation is used to authenticate users. See the Tivoli Access Manager documentation at this website:

http://publib.boulder.ibm.com/tividd/td/IBMAccessManagerfore-business5.1.html

The alternative is to implement at least Tivoli Directory Server 6.3 Fix Pack 10, which provides support for using pass-through authentication and the compare method. See the IBM technote, *Supporting pass-through authentication with Proxy server in Tivoli Directory Server Version 6.3 fix pack 10 (6.3.0.10)*:

http://www-01.ibm.com/support/docview.wss?uid=swg21578293

4.4.6 Importing users into Tivoli Access Manager

Example 4-4 on page 145 is a sample Tivoli Directory Integrator AssemblyLine called `TAMScriptFile.xml`. This AssemblyLine reads the users and groups from Tivoli Directory Server and creates files for importing users and groups into Tivoli Access Manager.

The sample contains the following Tivoli Directory Integrator AssemblyLines:

CreateUserFile   Based on the search base criteria mentioned in this property file, this AssemblyLine creates a file named `UserFile.txt`. The file contains all the users who are required for import by reading them from the Tivoli Directory Server.

CreateGroupFile   Based on the search base criteria mentioned in this property file, this AssemblyLine creates a file named `GroupFile.txt`. The file contains all of the groups that are required for import by reading them from Tivoli Directory Server.

CreateFiles   The main AssemblyLine, which internally calls the other AssemblyLine, `CreateUserFile` and `CreateGroupFile`.

Follow these steps to deploy the sample `TAMScriptFile` AssemblyLine:

1. Run the CreateFiles AssemblyLine:

    ```
    ibmdisrv -c TAMScriptFile.xml -r CreateFiles
    ```

2. After the AssemblyLine runs, the files `UserFile.txt` and `GroupFile.txt` are created.

3. Run the Tivoli Access Manager **pdadmin** command-line utility passing the `UserFile.txt` and `GroupFile.txt` files:

pdadmin -a sec_master -p *** /%path%/UserFile.txt**

pdadmin -a sec_master -p *** /%pasth%/GroupFile.txt**

> **Note:** In this example, `/%pasth%/` is the path location of each file.

Optionally, you can make these files run automatically:

1. Create a shell file for each command:

   ibmdisrv -c TAMScriptFile.xml -r CreateFiles.

   pdadmin -a sec_master -p *** /%path%/UserFile.txt**

   pdadmin -a sec_master -p *** /%pasth%/GroupFile.txt**

2. Use the Linux scheduling tool `crontab` to run the shell files at a specific time.

Example 4-3 shows a sample properties file that you need to modify for your environment.

Example 4-3   Sample properties file: LDAPImport.properties

```
##{PropertiesConnector} savedBy=IBM_ADMIN, saveDate=Wed Jul 03 16:03:12 IST 2013
# -----------------------------------------------
#
# -----------------------------------------------
# Source settings can be defined globally, and per Endpoint or Flow
#
ldap.url=ldap://10.241.110.190
ldap.user=cn=root
{protect}-ldap.password={encr}foa1g+ZwGytT4aSzxs9F/WGatR2OdJIkhdwfcsGcAEfSXCJa3gQjC8fpJACtEQo7tM
GJmU+/ZoQRxGyElcTWsDX1PBvbIZrK1Ebj9yOu7WUR7/rDRlXC43tiDye7Rohk4MkhWYv/H/6iMWcXGr5y3+tx3Ttquc39XI
DgjAqQG28=
ldap.searchBase=ou=swg,o=ibm,c=us
UserSearchFilter=objectClass=person
GroupSearchFilter=objectClass=groupofuniquenames
GroupFileCreate=GroupFile.txt
CreateUserFile=UserFile.txt
```

Example 4-4 shows a sample configuration file that you need to modify for your environment.

Example 4-4   Sample configuration file: TAMScriptFile.xml

```
<?xml version="1.0" encoding="UTF-8"?><MetamergeConfig IDIversion="Created by TDI7.1.1.2 -
2013-06-27" created="Wed Jul 03 16:04:50 IST 2013" createdBy="IBM_ADMIN" modified="Wed Jul 03
16:04:50 IST 2013" modifiedBy="IBM_ADMIN" version="7.1.1">
    <Folder name="AssemblyLines">
        <AssemblyLine name="CreateUserFile">
            <ModTime>1372847356108</ModTime>
            <Settings/>
            <Hooks/>
            <CheckpointConfig/>
            <SandboxConfig/>
            <SimulationConfig>
                <SimulationStates>
                    <Component name="AddUser" state="Simulated"/>
                    <Component name="EnableUser" state="Simulated"/>
```

```
                    <Component name="LDAPConnector" state="Enabled"/>
                </SimulationStates>
                <ProxySettings/>
            </SimulationConfig>
            <LogConfig/>
            <ContainerEF name="EntryFeedContainer">
                <Connector name="LDAPConnector">
                    <InheritFrom>system:/Connectors/ibmdi.LDAP</InheritFrom>
                    <ModTime>1372840133169</ModTime>
                    <ConnectorMode>Iterator</ConnectorMode>
                    <ConnectorState>Enabled</ConnectorState>
                    <Configuration>
                        <InheritFrom>[parent]</InheritFrom>
                        <parameter name="debug">true</parameter>
                        <parameter
name="ldapPassword">@SUBSTITUTE{property.LDAPImport:ldap.password}</parameter>
                        <parameter
name="ldapSearchBase">@SUBSTITUTE{property.LDAPImport:ldap.searchBase}</parameter>
                        <parameter
name="ldapSearchFilter">@SUBSTITUTE{property.LDAPImport:UserSearchFilter}</parameter>
                        <parameter
name="ldapUrl">@SUBSTITUTE{property.LDAPImport:ldap.url}</parameter>
                        <parameter
name="ldapUsername">@SUBSTITUTE{property.LDAPImport:ldap.user}</parameter>
                    </Configuration>
                    <Parser>
                        <InheritFrom>[parent]</InheritFrom>
                        <Schema name="Input">
                            <InheritFrom>[parent]</InheritFrom>
                        </Schema>
                        <Schema name="Output">
                            <InheritFrom>[parent]</InheritFrom>
                        </Schema>
                    </Parser>
                    <AttributeMap name="Input">
                        <InheritFrom>[parent]</InheritFrom>
                        <AttributeMapItem>
                            <Name>$dn</Name>
                            <Type>simple</Type>
                            <Simple>$dn</Simple>
                        </AttributeMapItem>
                        <AttributeMapItem>
                            <Name>cn</Name>
                            <Type>simple</Type>
                            <Simple>cn</Simple>
                        </AttributeMapItem>
                    </AttributeMap>
                    <AttributeMap name="Output">
                        <InheritFrom>[parent]</InheritFrom>
                    </AttributeMap>
                    <DeltaSettings>
                        <UniqueAttribute/>
                        <WhenToCommit>After every database operation</WhenToCommit>
                        <RowLocking>SERIALIZABLE</RowLocking>
                        <ChangeDetectionMode>DETECT_ALL</ChangeDetectionMode>
```

```
        </DeltaSettings>
        <Schema name="Input">
            <InheritFrom>[parent]</InheritFrom>
            <SchemaItem>
                <Name>$dn</Name>
                <Syntax>java.lang.String</Syntax>
            </SchemaItem>
            <SchemaItem>
                <Name>cn</Name>
                <Syntax>java.lang.String</Syntax>
            </SchemaItem>
            <SchemaItem>
                <Name>givenname</Name>
                <Syntax>java.lang.String</Syntax>
            </SchemaItem>
            <SchemaItem>
                <Name>objectClass</Name>
                <Syntax>java.lang.String</Syntax>
            </SchemaItem>
            <SchemaItem>
                <Name>sn</Name>
                <Syntax>java.lang.String</Syntax>
            </SchemaItem>
        </Schema>
        <Schema name="Output">
            <InheritFrom>[parent]</InheritFrom>
            <SchemaItem>
                <Name>$dn</Name>
                <Syntax>java.lang.String</Syntax>
            </SchemaItem>
            <SchemaItem>
                <Name>cn</Name>
                <Syntax>java.lang.String</Syntax>
            </SchemaItem>
            <SchemaItem>
                <Name>givenname</Name>
                <Syntax>java.lang.String</Syntax>
            </SchemaItem>
            <SchemaItem>
                <Name>objectClass</Name>
                <Syntax>java.lang.String</Syntax>
            </SchemaItem>
            <SchemaItem>
                <Name>sn</Name>
                <Syntax>java.lang.String</Syntax>
            </SchemaItem>
        </Schema>
        <LinkCriteria>
            <InheritFrom>[parent]</InheritFrom>
        </LinkCriteria>
        <Hooks>
            <InheritFrom>[parent]</InheritFrom>
        </Hooks>
        <CheckpointConfig/>
        <SandboxConfig/>
```

```xml
                    <Reconnect>
                        <InheritFrom>[parent]</InheritFrom>
                        <ReconnectRules/>
                    </Reconnect>
                    <Operations/>
                    <PoolDefinition>
                        <InheritFrom>[parent]</InheritFrom>
                    </PoolDefinition>
                    <PoolInstance/>
                </Connector>
            </ContainerEF>
            <ContainerDF name="DataFlowContainer">
                <Connector name="AddUser">
                    <InheritFrom>system:/Connectors/ibmdi.FileSystem</InheritFrom>
                    <ModTime>1372847331722</ModTime>
                    <ConnectorMode>AddOnly</ConnectorMode>
                    <ConnectorState>Enabled</ConnectorState>
                    <Configuration>
                        <InheritFrom>[parent]</InheritFrom>
                        <parameter name="debug">true</parameter>
                        <parameter name="fileAppend">true</parameter>
                        <parameter
name="filePath">@SUBSTITUTE{property.LDAPImport:CreateUserFile}</parameter>
                    </Configuration>
                    <Parser>
                        <InheritFrom>system:/Parsers/ibmdi.LineReader</InheritFrom>
                        <parameter name="debug">true</parameter>
                        <Schema name="Input">
                            <InheritFrom>[parent]</InheritFrom>
                        </Schema>
                        <Schema name="Output">
                            <InheritFrom>[parent]</InheritFrom>
                        </Schema>
                    </Parser>
                    <AttributeMap name="Input">
                        <InheritFrom>[parent]</InheritFrom>
                    </AttributeMap>
                    <AttributeMap name="Output">
                        <InheritFrom>[parent]</InheritFrom>
                        <AttributeMapItem>
                            <Name>line</Name>
                            <Type>advanced</Type>
                            <Script>return "user import " + "\""+work.getString("cn") + "\""+ "
" + "\""+ work.getString("$dn")+ "\"";</Script>
                            <Simple>line</Simple>
                        </AttributeMapItem>
                    </AttributeMap>
                    <DeltaSettings/>
                    <Schema name="Input">
                        <InheritFrom>[parent]</InheritFrom>
                    </Schema>
                    <Schema name="Output">
                        <InheritFrom>[parent]</InheritFrom>
                    </Schema>
                    <LinkCriteria>
```

```
                    <InheritFrom>[parent]</InheritFrom>
                </LinkCriteria>
                <Hooks>
                    <InheritFrom>[parent]</InheritFrom>
                </Hooks>
                <CheckpointConfig/>
                <SandboxConfig/>
                <Reconnect>
                    <InheritFrom>[parent]</InheritFrom>
                    <ReconnectRules/>
                </Reconnect>
                <Operations/>
                <PoolDefinition>
                    <InheritFrom>[parent]</InheritFrom>
                </PoolDefinition>
                <PoolInstance/>
            </Connector>
            <Connector name="EnableUser">
                <InheritFrom>system:/Connectors/ibmdi.FileSystem</InheritFrom>
                <ModTime>1372847356108</ModTime>
                <ConnectorMode>AddOnly</ConnectorMode>
                <ConnectorState>Enabled</ConnectorState>
                <Configuration>
                    <InheritFrom>[parent]</InheritFrom>
                    <parameter name="debug">true</parameter>
                    <parameter name="fileAppend">true</parameter>
                    <parameter
name="filePath">@SUBSTITUTE{property.LDAPImport:CreateUserFile}</parameter>
                </Configuration>
                <Parser>
                    <InheritFrom>system:/Parsers/ibmdi.LineReader</InheritFrom>
                    <parameter name="debug">true</parameter>
                    <Schema name="Input">
                        <InheritFrom>[parent]</InheritFrom>
                    </Schema>
                    <Schema name="Output">
                        <InheritFrom>[parent]</InheritFrom>
                    </Schema>
                </Parser>
                <AttributeMap name="Input">
                    <InheritFrom>[parent]</InheritFrom>
                </AttributeMap>
                <AttributeMap name="Output">
                    <InheritFrom>[parent]</InheritFrom>
                    <AttributeMapItem>
                        <Name>line</Name>
                        <Type>advanced</Type>
                        <Script>return "user modify " + "\"" +work.getString("cn") +"\"" + "
" + "account-valid yes";</Script>
                        <Simple>line</Simple>
                    </AttributeMapItem>
                </AttributeMap>
                <DeltaSettings/>
                <Schema name="Input">
                    <InheritFrom>[parent]</InheritFrom>
```

```
            </Schema>
            <Schema name="Output">
                <InheritFrom>[parent]</InheritFrom>
            </Schema>
            <LinkCriteria>
                <InheritFrom>[parent]</InheritFrom>
            </LinkCriteria>
            <Hooks>
                <InheritFrom>[parent]</InheritFrom>
            </Hooks>
            <CheckpointConfig/>
            <SandboxConfig/>
            <Reconnect>
                <InheritFrom>[parent]</InheritFrom>
                <ReconnectRules/>
            </Reconnect>
            <Operations/>
            <PoolDefinition>
                <InheritFrom>[parent]</InheritFrom>
            </PoolDefinition>
            <PoolInstance/>
        </Connector>
    </ContainerDF>
    <ThreadOptions/>
    <Operations/>
    <InitParams>
        <Schema name="AssemblyLineInitParams"/>
    </InitParams>
</AssemblyLine>
<AssemblyLine name="CreateGroupFile">
    <ModTime>1372847388098</ModTime>
    <Settings/>
    <Hooks/>
    <CheckpointConfig/>
    <SandboxConfig/>
    <SimulationConfig>
        <SimulationStates>
            <Component name="AddGroup" state="Simulated"/>
            <Component name="LDAPConnector" state="Enabled"/>
        </SimulationStates>
        <ProxySettings/>
    </SimulationConfig>
    <LogConfig/>
    <ContainerEF name="EntryFeedContainer">
        <Connector name="LDAPConnector">
            <InheritFrom>system:/Connectors/ibmdi.LDAP</InheritFrom>
            <ModTime>1372840133166</ModTime>
            <ConnectorMode>Iterator</ConnectorMode>
            <ConnectorState>Enabled</ConnectorState>
            <Configuration>
                <InheritFrom>[parent]</InheritFrom>
                <parameter name="debug">true</parameter>
                <parameter
name="ldapPassword">@SUBSTITUTE{property.LDAPImport:ldap.password}</parameter>
```

```
                        <parameter
name="ldapSearchBase">@SUBSTITUTE{property.LDAPImport:ldap.searchBase}</parameter>
                        <parameter
name="ldapSearchFilter">@SUBSTITUTE{property.LDAPImport:GroupSearchFilter}</parameter>
                        <parameter
name="ldapUrl">@SUBSTITUTE{property.LDAPImport:ldap.url}</parameter>
                        <parameter
name="ldapUsername">@SUBSTITUTE{property.LDAPImport:ldap.user}</parameter>
                    </Configuration>
                    <Parser>
                        <InheritFrom>[parent]</InheritFrom>
                        <Schema name="Input">
                            <InheritFrom>[parent]</InheritFrom>
                        </Schema>
                        <Schema name="Output">
                            <InheritFrom>[parent]</InheritFrom>
                        </Schema>
                    </Parser>
                    <AttributeMap name="Input">
                        <InheritFrom>[parent]</InheritFrom>
                        <AttributeMapItem>
                            <Name>$dn</Name>
                            <Type>simple</Type>
                            <Simple>$dn</Simple>
                        </AttributeMapItem>
                        <AttributeMapItem>
                            <Name>cn</Name>
                            <Type>simple</Type>
                            <Simple>cn</Simple>
                        </AttributeMapItem>
                    </AttributeMap>
                    <AttributeMap name="Output">
                        <InheritFrom>[parent]</InheritFrom>
                    </AttributeMap>
                    <DeltaSettings>
                        <UniqueAttribute/>
                        <WhenToCommit>After every database operation</WhenToCommit>
                        <RowLocking>SERIALIZABLE</RowLocking>
                        <ChangeDetectionMode>DETECT_ALL</ChangeDetectionMode>
                    </DeltaSettings>
                    <Schema name="Input">
                        <InheritFrom>[parent]</InheritFrom>
                        <SchemaItem>
                            <Name>$dn</Name>
                            <Syntax>java.lang.String</Syntax>
                        </SchemaItem>
                        <SchemaItem>
                            <Name>cn</Name>
                            <Syntax>java.lang.String</Syntax>
                        </SchemaItem>
                        <SchemaItem>
                            <Name>givenname</Name>
                            <Syntax>java.lang.String</Syntax>
                        </SchemaItem>
                        <SchemaItem>
```

```
                    <Name>objectClass</Name>
                    <Syntax>java.lang.String</Syntax>
                </SchemaItem>
                <SchemaItem>
                    <Name>sn</Name>
                    <Syntax>java.lang.String</Syntax>
                </SchemaItem>
            </Schema>
            <Schema name="Output">
                <InheritFrom>[parent]</InheritFrom>
                <SchemaItem>
                    <Name>$dn</Name>
                    <Syntax>java.lang.String</Syntax>
                </SchemaItem>
                <SchemaItem>
                    <Name>cn</Name>
                    <Syntax>java.lang.String</Syntax>
                </SchemaItem>
                <SchemaItem>
                    <Name>givenname</Name>
                    <Syntax>java.lang.String</Syntax>
                </SchemaItem>
                <SchemaItem>
                    <Name>objectClass</Name>
                    <Syntax>java.lang.String</Syntax>
                </SchemaItem>
                <SchemaItem>
                    <Name>sn</Name>
                    <Syntax>java.lang.String</Syntax>
                </SchemaItem>
            </Schema>
            <LinkCriteria>
                <InheritFrom>[parent]</InheritFrom>
            </LinkCriteria>
            <Hooks>
                <InheritFrom>[parent]</InheritFrom>
            </Hooks>
            <CheckpointConfig/>
            <SandboxConfig/>
            <Reconnect>
                <InheritFrom>[parent]</InheritFrom>
                <ReconnectRules/>
            </Reconnect>
            <Operations/>
            <PoolDefinition>
                <InheritFrom>[parent]</InheritFrom>
            </PoolDefinition>
            <PoolInstance/>
        </Connector>
    </ContainerEF>
    <ContainerDF name="DataFlowContainer">
        <Connector name="AddGroup">
            <InheritFrom>system:/Connectors/ibmdi.FileSystem</InheritFrom>
            <ModTime>1372847388098</ModTime>
            <ConnectorMode>AddOnly</ConnectorMode>
```

```
<ConnectorState>Enabled</ConnectorState>
<Configuration>
    <InheritFrom>[parent]</InheritFrom>
    <parameter name="debug">true</parameter>
    <parameter name="fileAppend">true</parameter>
    <parameter
name="filePath">@SUBSTITUTE{property.LDAPImport:GroupFileCreate}</parameter>
</Configuration>
<Parser>
    <InheritFrom>system:/Parsers/ibmdi.LineReader</InheritFrom>
    <parameter name="debug">true</parameter>
    <Schema name="Input">
        <InheritFrom>[parent]</InheritFrom>
    </Schema>
    <Schema name="Output">
        <InheritFrom>[parent]</InheritFrom>
    </Schema>
</Parser>
<AttributeMap name="Input">
    <InheritFrom>[parent]</InheritFrom>
</AttributeMap>
<AttributeMap name="Output">
    <InheritFrom>[parent]</InheritFrom>
    <AttributeMapItem>
        <Name>line</Name>
        <Type>advanced</Type>
        <Script>return "group import " + "\"" + work.getString("cn") + "\""+
" " + " \""+ work.getString("$dn")+ "\"";</Script>
    <Simple>line</Simple>
    </AttributeMapItem>
</AttributeMap>
<DeltaSettings/>
<Schema name="Input">
    <InheritFrom>[parent]</InheritFrom>
</Schema>
<Schema name="Output">
    <InheritFrom>[parent]</InheritFrom>
</Schema>
<LinkCriteria>
    <InheritFrom>[parent]</InheritFrom>
</LinkCriteria>
<Hooks>
    <InheritFrom>[parent]</InheritFrom>
</Hooks>
<CheckpointConfig/>
<SandboxConfig/>
<Reconnect>
    <InheritFrom>[parent]</InheritFrom>
    <ReconnectRules/>
</Reconnect>
<Operations/>
<PoolDefinition>
    <InheritFrom>[parent]</InheritFrom>
</PoolDefinition>
<PoolInstance/>
```

```
                </Connector>
            </ContainerDF>
            <ThreadOptions/>
            <Operations/>
            <InitParams>
                <Schema name="AssemblyLineInitParams"/>
            </InitParams>
    </AssemblyLine>
    <AssemblyLine name="CreateFiles">
            <ModTime>1372841875159</ModTime>
            <Settings/>
            <Hooks/>
            <CheckpointConfig/>
            <SandboxConfig/>
            <SimulationConfig>
                <SimulationStates>
                    <Component name="CreateUserFileALFC" state="Simulated"/>
                    <Component name="CreateGroupFileALFC" state="Simulated"/>
                </SimulationStates>
                <ProxySettings/>
            </SimulationConfig>
            <LogConfig/>
            <ContainerEF name="EntryFeedContainer"/>
            <ContainerDF name="DataFlowContainer">
                <Function name="CreateUserFileALFC">
                    <InheritFrom>system:/Functions/ibmdi.AssemblyLineFC</InheritFrom>
                    <ModTime>1372839231676</ModTime>
                    <ConnectorState>Enabled</ConnectorState>
                    <Schema name="Input">
                        <InheritFrom>[parent]</InheritFrom>
                        <SchemaItem>
                            <Name>$dn</Name>
                        </SchemaItem>
                        <SchemaItem>
                            <Name>cn</Name>
                        </SchemaItem>
                    </Schema>
                    <Schema name="Output">
                        <InheritFrom>[parent]</InheritFrom>
                        <SchemaItem>
                            <Name>$dn</Name>
                        </SchemaItem>
                        <SchemaItem>
                            <Name>cn</Name>
                        </SchemaItem>
                        <SchemaItem>
                            <Name>line</Name>
                        </SchemaItem>
                    </Schema>
                    <Hooks>
                        <InheritFrom>[parent]</InheritFrom>
                    </Hooks>
                    <Configuration>
                        <InheritFrom>[parent]</InheritFrom>
                        <parameter name="assemblyLine">CreateUserFile</parameter>
```

```
                <parameter name="debug">true</parameter>
            </Configuration>
            <SandboxConfig/>
            <AttributeMap name="Input">
                <InheritFrom>[parent]</InheritFrom>
            </AttributeMap>
            <AttributeMap name="Output">
                <InheritFrom>[parent]</InheritFrom>
            </AttributeMap>
        </Function>
        <Function name="CreateGroupFileALFC">
            <InheritFrom>system:/Functions/ibmdi.AssemblyLineFC</InheritFrom>
            <ModTime>1372841875159</ModTime>
            <ConnectorState>Enabled</ConnectorState>
            <Schema name="Input">
                <InheritFrom>[parent]</InheritFrom>
                <SchemaItem>
                    <Name>$dn</Name>
                </SchemaItem>
                <SchemaItem>
                    <Name>cn</Name>
                </SchemaItem>
            </Schema>
            <Schema name="Output">
                <InheritFrom>[parent]</InheritFrom>
                <SchemaItem>
                    <Name>$dn</Name>
                </SchemaItem>
                <SchemaItem>
                    <Name>cn</Name>
                </SchemaItem>
                <SchemaItem>
                    <Name>line</Name>
                </SchemaItem>
            </Schema>
            <Hooks>
                <InheritFrom>[parent]</InheritFrom>
            </Hooks>
            <Configuration>
                <InheritFrom>[parent]</InheritFrom>
                <parameter name="assemblyLine">CreateGroupFile</parameter>
                <parameter name="debug">true</parameter>
            </Configuration>
            <SandboxConfig/>
            <AttributeMap name="Input">
                <InheritFrom>[parent]</InheritFrom>
            </AttributeMap>
            <AttributeMap name="Output">
                <InheritFrom>[parent]</InheritFrom>
            </AttributeMap>
        </Function>
    </ContainerDF>
    <ThreadOptions/>
    <Operations/>
    <InitParams>
```

```
                <Schema name="AssemblyLineInitParams"/>
            </InitParams>
        </AssemblyLine>
    </Folder>
    <Folder name="Connectors"/>
    <Folder name="Parsers"/>
    <Folder name="Scripts"/>
    <JavaLibraries/>
    <JavaProperties/>
    <Folder name="Includes"/>
    <Folder name="Config">
        <LogConfig name="Logging"/>
        <InstanceProperties name="AutoStart">
            <AutoStart/>
        </InstanceProperties>
        <TombstonesConfig name="Tombstones"/>
        <SolutionInterface name="SolutionInterface">
            <ModTime>1372833116085</ModTime>
            <PollInterval>-1</PollInterval>
            <InstanceID>TAMScript</InstanceID>
            <enabled>true</enabled>
        </SolutionInterface>
    </Folder>
    <Folder name="Functions"/>
    <Folder name="AttributeMaps"/>
    <Properties name="Properties">
        <Stores>
            <PropertyStore name="Solution-Properties">
                <Parser>
                    <Schema name="Input">
                        <InheritFrom>[parent]</InheritFrom>
                    </Schema>
                    <Schema name="Output">
                        <InheritFrom>[parent]</InheritFrom>
                    </Schema>
                </Parser>
                <RawConnector>
                    <InheritFrom>system:/Connectors/ibmdi.Properties</InheritFrom>
                    <parameter name="collectionType">Solution-Properties</parameter>
                </RawConnector>
                <Key>key</Key>
                <Value>value</Value>
                <ReadOnly>false</ReadOnly>
                <InitialLoad>true</InitialLoad>
                <CacheTimeout>0</CacheTimeout>
            </PropertyStore>
            <PropertyStore name="Global-Properties">
                <Parser>
                    <Schema name="Input">
                        <InheritFrom>[parent]</InheritFrom>
                    </Schema>
                    <Schema name="Output">
                        <InheritFrom>[parent]</InheritFrom>
                    </Schema>
                </Parser>
```

```
            <RawConnector>
                <InheritFrom>system:/Connectors/ibmdi.Properties</InheritFrom>
                <parameter name="collectionType">Global-Properties</parameter>
            </RawConnector>
            <Key>key</Key>
            <Value>value</Value>
            <ReadOnly>false</ReadOnly>
            <InitialLoad>true</InitialLoad>
            <CacheTimeout>0</CacheTimeout>
        </PropertyStore>
        <PropertyStore name="System-Properties">
            <Parser>
                <Schema name="Input">
                    <InheritFrom>[parent]</InheritFrom>
                </Schema>
                <Schema name="Output">
                    <InheritFrom>[parent]</InheritFrom>
                </Schema>
            </Parser>
            <RawConnector>
                <InheritFrom>system:/Connectors/ibmdi.Properties</InheritFrom>
                <parameter name="collectionType">System-Properties</parameter>
            </RawConnector>
            <Key>key</Key>
            <Value>value</Value>
            <ReadOnly>false</ReadOnly>
            <InitialLoad>true</InitialLoad>
            <CacheTimeout>0</CacheTimeout>
        </PropertyStore>
        <PropertyStore name="Java-Properties">
            <Parser>
                <Schema name="Input">
                    <InheritFrom>[parent]</InheritFrom>
                </Schema>
                <Schema name="Output">
                    <InheritFrom>[parent]</InheritFrom>
                </Schema>
            </Parser>
            <RawConnector>
                <InheritFrom>system:/Connectors/ibmdi.Properties</InheritFrom>
                <parameter
name="collection">@SUBSTITUTE{config.$directory}/TAMScript.properties</parameter>
                <parameter name="collectionType">Java-Properties</parameter>
            </RawConnector>
            <Key>key</Key>
            <Value>value</Value>
            <ReadOnly>false</ReadOnly>
            <InitialLoad>true</InitialLoad>
            <CacheTimeout>0</CacheTimeout>
        </PropertyStore>
        <PropertyStore name="TAMScript">
            <ModTime>1372847489153</ModTime>
            <Parser>
                <Schema name="Input">
                    <InheritFrom>[parent]</InheritFrom>
```

```
                </Schema>
                <Schema name="Output">
                    <InheritFrom>[parent]</InheritFrom>
                </Schema>
            </Parser>
            <RawConnector>
                <InheritFrom>system:/Connectors/ibmdi.Properties</InheritFrom>
                <parameter name="collection">C:\Program Files
(x86)\IBM\TDI\V7.1.1\workspace\TAMScript\Runtime-TAMScript/TAMScript.properties</parameter>
                <parameter name="collectionType">TAMScript</parameter>
                <parameter name="keyAlias"/>
            </RawConnector>
            <Key>key</Key>
            <Value>value</Value>
            <ReadOnly>false</ReadOnly>
            <InitialLoad>true</InitialLoad>
            <CacheTimeout>0</CacheTimeout>
        </PropertyStore>
        <PropertyStore name="LDAPImport">
            <ModTime>1372847605625</ModTime>
            <Parser>
                <Schema name="Input">
                    <InheritFrom>[parent]</InheritFrom>
                </Schema>
                <Schema name="Output">
                    <InheritFrom>[parent]</InheritFrom>
                </Schema>
            </Parser>
            <RawConnector>
                <InheritFrom>system:/Connectors/ibmdi.Properties</InheritFrom>
                <parameter name="collection">LDAPImport.properties</parameter>
                <parameter name="collectionType">Default</parameter>
                <parameter name="debug">true</parameter>
                <parameter name="keyAlias"/>
            </RawConnector>
            <Key>key</Key>
            <Value>value</Value>
            <ReadOnly>false</ReadOnly>
            <InitialLoad>true</InitialLoad>
            <CacheTimeout>0</CacheTimeout>
        </PropertyStore>
    </Stores>
</Properties>
<Folder name="Schedules"/>
<Folder name="Sequences"/>
</MetamergeConfig>
```

You can use Tivoli Directory Integrator configuration editor to modify the properties file and run the AssemblyLines. Follow these steps to start Tivoli Directory Integrator configuration editor:

1. Log on to the server where Tivoli Directory Integrator 7.1.1 is installed.

2. Run <*TDI_V711_HOME*>/V7.1.1/ibmditk.

4.4.7 Administering IBM Intelligent Operations Center

IBM Intelligent Operations Center includes the `IOCControl` command to start, stop, and query the status of the servers. The `IOCControl.sh` script is on the management server in the directory `/opt/IBM/ISP/mgmt/scripts`.

The `IOCControl` command can be run for all servers, or individual servers. The `start` action ensures that the underlying services are started in the correct sequence.

The `IOCControl` command has this syntax:

`IOCControl.sh <Action> <Target> <Password>`

The variables are defined:

► *Action*: Can be `start`, `stop`, `status`, or `help`.
► *Target*: Can be either `all` or an individual server name.
► *Password*: This is the password for the Platform Control Tool that is defined when IBM Intelligent Operations Center is deployed.

To display the options that are available for the `IOCControl` command, run the following command:

`cd /opt/IBM/ISP/mgmt/scripts`

`./IOCControl.sh help <Password>`

For more information about IBM Operations Center Administration, see the *IBM Intelligent Operations Center for Smarter Cities Administration Guide,* SG24-8061:

http://www.redbooks.ibm.com/abstracts/sg248061.html?Open

Starting servers with IOCControl

Two options are available to start the servers with the `IOCControl.sh` script:

► Start all servers

   To start all IBM Intelligent Operations Center servers, run the following command on the management server:

   `cd /opt/IBM/ISP/mgmt/scripts/`

   `./IOCControl.sh start all <Password>`

► Start individual servers

   To start an individual IBM Intelligent Operations Center server, run the following command on the management server:

   `cd /opt/IBM/ISP/mgmt/scripts`

   `./IOCControl.sh start <Target> <Password>`

   For example, to start the business monitoring service, run the following command:

   `./IOCControl.sh start wbm <Password>`

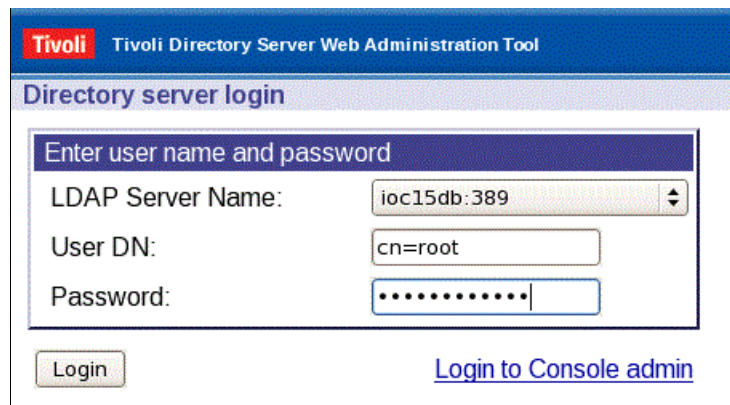### 4.4.8 Setting up SSL for the Tivoli Directory Server

This topic is described in the IBM Security Directory Server IBM Knowledge Center article "Configuring security settings":

`http://pic.dhe.ibm.com/infocenter/tivihelp/v2r1/topic/com.ibm.IBMDS.doc/admin_gd169.htm?path=7_4_4_7_0#cfgset`

4.4.9 Using IBM Tivoli Directory Server Web Administration Tool

The IBM Tivoli Directory Server Web Administration Tool is a web-based GUI for managing the IBM Tivoli Directory Server. You must set up the Web Administration Tool before it can be used for managing the directory server. Perform the following steps:

1. Access the Tivoli Directory Server Web Administration Tool from IBM Intelligent Operations Center Administration Tools, or directly by pointing a browser to the URL `http://<IOC15MGMT>:9081/IDSWebApp/` where `<IOC5MGMT>` is *hostname* of the management server.

2. Log in to the console with the user ID `superadmin` and password `secret` as shown in Figure 4-19.

Figure 4-19   Tivoli Directory Server console administration login

3. Expand **Console administration** and click **Manage console servers**.

4. Click **Add**.

5. Enter the host name for the IBM Intelligent Operations Center data server where the directory services are installed. See Figure 4-20 on page 161.
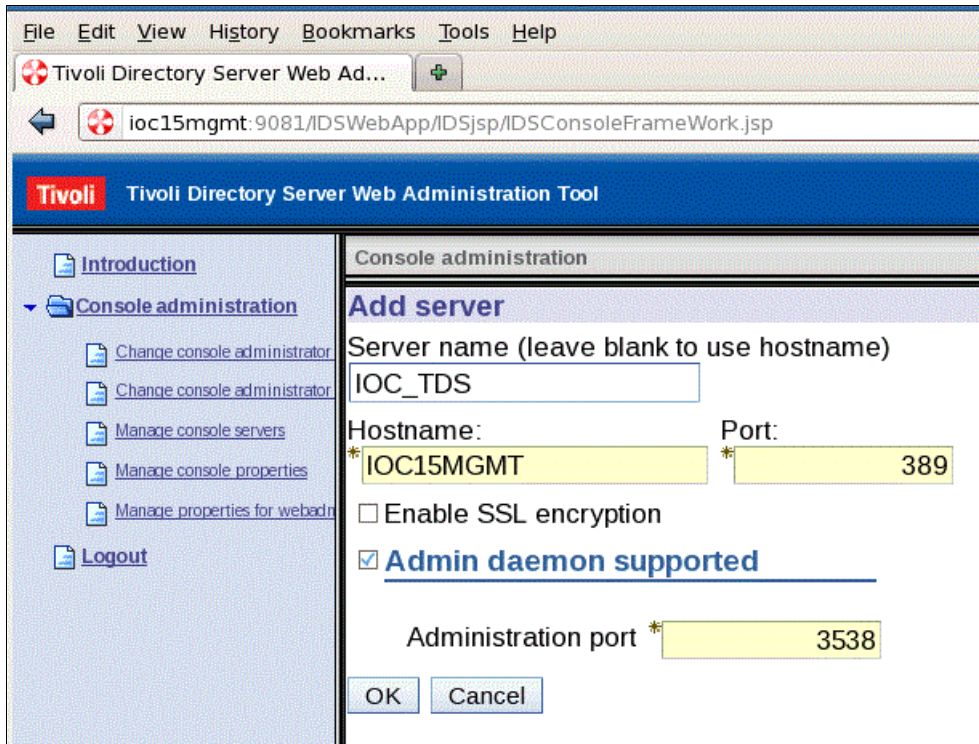
Figure 4-20   Adding a server in Tivoli Directory Server

6. Click **OK**.

7. Click **Logout**.

The Tivoli Directory Server Web Administration Tool is now configured for managing the directory server. You can log in to the console with user DN `cn=root` and the password assigned.

4.4.10 Troubleshooting

The LDAPSync solution writes its log files to the `LDAPSync/logs` folder in the solution directory.

> **Note:** When you run an AssemblyLine, such as `LDAPMigrate` or `LDAPSync`, the result is a log file with the same name as the AssemblyLine, plus one or more log files created by other AssemblyLines that the main AssemblyLine uses to do its job.

The following system logs are written by the Tivoli Directory Integrator Server to the `logs` subfolder of the Tivoli Directory Integrator solution directory `/opt/IBM/TDI/V7.1.1/logs/`:

► `ibmdi.log`: The log file of the IBM Tivoli Directory Integrator Server. It contains all of the messages that are written by AssemblyLines to their own solution logs and any server-level messages.

► `ibmditk.log`: The log file produced by the IBM Tivoli Directory Integrator development environment, the Configuration Editor (CE).

5

Integrated Law Enforcement single sign-on

One of the most important areas of the IBM i2 Integrated Law Enforcement solution is security. Chapter 4, "Integrated Law Enforcement security" on page 107 describes the various security models that are used by the product components, and how global security is defined in the context of this integrated solution.

One of the tenets discussed in Chapter 4, "Integrated Law Enforcement security" on page 107 highlights that the essential element of an integrated solution is its ability to provide single sign-on (SSO) capabilities. This chapter is devoted to looking into the implementation of SSO in i2 Integrated Law Enforcement.

For a solution to be truly integrated, it must facilitate the focal shift from discrete individual components to a unified system that performs a set of capabilities that result from an orchestrated interaction of components, and yet abstracted from the user. The overall user experience must lend itself easily to being highly useful, even with little training or familiarity with the system.

SSO must have the following characteristics:

► The user is aware of providing credentials to the system at most once.

► User identity is propagated by the system across the entire solution, without compromising the user's credentials.

► Any user transaction can be traced back to and associated with the user's identity, regardless of which product component the transaction is in.

This chapter focuses on how SSO is achieved in the i2 Integrated Law Enforcement solution. This information is useful when extending the capabilities of i2 Integrated Law Enforcement.

5.1 Single sign-on

The security model of IBM Intelligent Operations Center uses authentication based on IBM WebSphere Portal technology and a WebSEAL reverse-proxy. Figure 5-1 summarizes this security model.
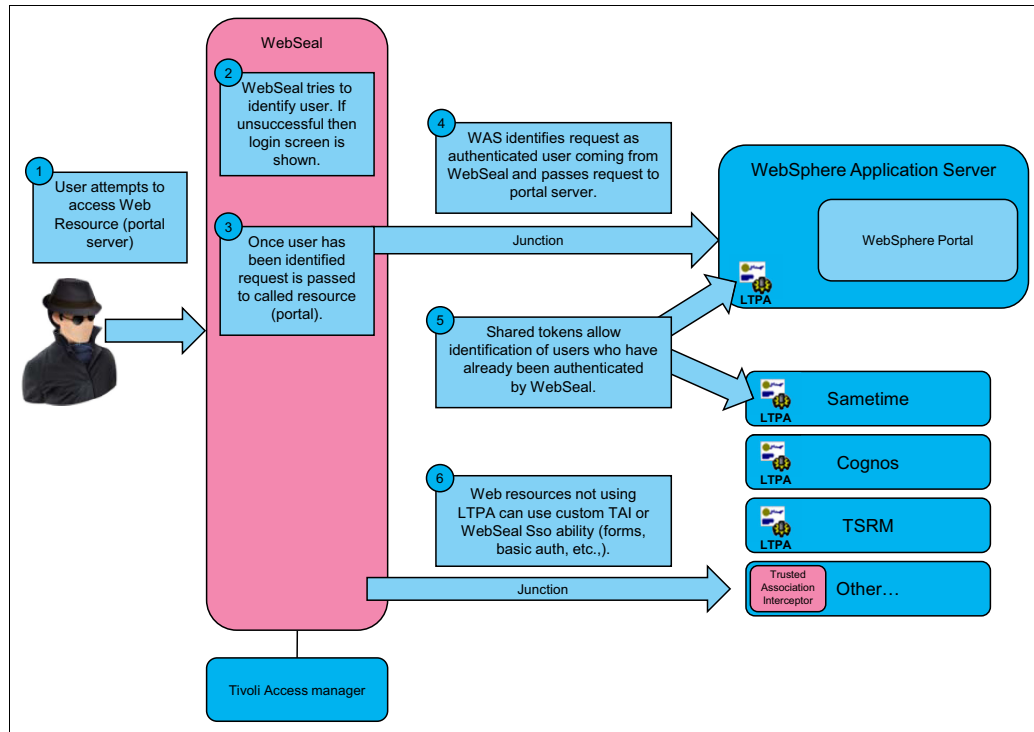

Figure 5-1   High-level process diagram for IBM Intelligent Operations Center authentication

In IBM Intelligent Operations Center, the IBM Directory Server is used as the Lightweight Directory Access Protocol (LDAP) server to store user information. Users authenticate to IBM WebSphere Portal using credentials that are stored in the Directory Server using WebSEAL. See *Introducing IBM Tivoli Access Manager and WebSEAL* at this website:

https://publib.boulder.ibm.com/tividd/td/ITAME/SC32-1359-00/en_US/HTML/am51_webseal_guide11.htm

SSO between the IBM WebSphere Portal and the other back-end services is provided by a Lightweight Third Party Authentication (LTPA) token to share user credentials using WebSEAL junctions. This method can be extended to provide SSO between IBM Intelligent Operations Center and other applications that are deployed outside of IBM Intelligent Operations Center.

Two main areas need to be configured to enable SSO between a new application and IBM Intelligent Operations Center:

► Configuring SSO between IBM WebSphere Portal and the new application
► Configuring SSO between WebSEAL and the new application

5.1.1 Configuring SSO between IBM WebSphere Portal and a new application

To configure SSO between IBM WebSphere Portal in IBM Intelligent Operations Center and a new application, perform the following steps:

1. The portal LTPA key, `portal.ltpa`, must be shared with the target application. The default location for the portal LTPA key is in the `/opt/pdweb/etc/` directory on the IBM Intelligent Operations Center application server. If this file cannot be found, export a new copy of the portal LTPA key using the WebSphere administrative console. For more information about exporting LTPA keys, see *Exporting Lightweight Third Party Authentication keys*:

   http://www.ibm.com/support/knowledgecenter/SSAW57_7.0.0/com.ibm.websphere.nd.multiplatform.doc/info/ae/ae/tsec_altpaexp.html?cp=SSAW57_7.0.0%2F3-3-0-0-2-1&lang=en

2. After the `portal.ltpa` key is located, import it into the target application environment. If the target application is a WebSphere application, you can obtain details about importing the LTPA key in the IBM Knowledge Center. Search for *Importing Lightweight Third Party Authentication keys*:

   http://www-01.ibm.com/support/knowledgecenter/SSAW57_7.0.0/com.ibm.websphere.soafep.multiplatform.doc/info/ae/ae/tsec_altpaimp.html?cp=SSAW57_7.0.0%2F8-4-0-0-2-2

For applications that are deployed in other environments, see the application documentation for details about implementing LTPA keys.

5.1.2 Configuring SSO between WebSEAL and a new application

To configure SSO between WebSEAL and a new application, create a Tivoli Access Manager WebSEAL junction with LTPA enabled on the IBM Intelligent Operations Center management server. Log in to the management server as the `root` user and perform the following steps:

1. Open a Linux command prompt and log in to the WebSEAL `pdadmin` console by running the following command:

   pdadmin -a sec_master -p *<sec_master_password>*

2. Run the following command to create the junction:

   server task default-webseald-*<server_name>* create -t tcp -h *<app_server_hostname>* -p *<app_server_port>* -j -J trailer -A -2 -F *<path_to_LTPA_key>* -Z *<LTPA_key_password>* /*<junction point>*

   The **default-webseald-*<server_name>*** is the fully qualified host name of the Tivoli Access Manager WebSEAL server. For IBM Intelligent Operations Center, this is the application server, for example, **default-webseald-ioc15app.mydomain.com**. Check the server name by issuing the **server list** command in the `pdadmin` console. In Example 5-1, the server is `default-webseald-ioc15app.ibmplatform.com`.

Example 5-1   Running the server list command

```
root@ioc15mgmt ~]# pdadmin -a sec_master -p sec_master_password
pdadmin sec_master> server list
    amwpm-ioc15mgmt.ibmplatform.com
    amwp70-ioc15app.ibmplatform.com
    default-webseald-ioc15app.ibmplatform.com
    ivacld-ioc15mgmt.ibmplatform.com
pdadmin sec_master>
```

The variables are defined:

- **h** *`<app_server_hostname>`* specifies the hostname of the back-end application server on which the new application is running.

- **p** *`<app_server_port>`* is the TCP port on the back-end application server.

- **F** *`<path_to_LTPA_key>`* specifies the location on the IBM Intelligent Operations Center application server where the LTPA key file is stored.

- **Z** *`<LTPA_key_password>`* specifies the password for the LTPA key. The mount point for the junction is /`<junction point>`.

> **Note:** The architecture of IBM Intelligent Operations Center lends itself to extensions that enrich its capabilities. Using this same mechanism, it can expand its integration points with i2 COPLINK and i2 Intelligence Analysis Platform. Because this integration grows through customization, ensure that SSO is enforced and maintained by these additional components.

5.2 SSO between the IBM Intelligent Operations Center and the i2 Intelligence Analysis Platform

A similar process to the process that is described in 5.1, "Single sign-on" on page 164 can be used to enable SSO between IBM Intelligent Operations Center and i2 Intelligence Analysis Platform. To implement SSO successfully, IBM Intelligent Operations Center and i2 Intelligence Analysis Platform WebSphere Application Servers, ideally, share the same user registry information. Therefore, it is desirable to federate the directory server that is used in IBM Intelligent Operations Center on both the i2 Intelligence Analysis Platform write and read servers.

Because both IBM WebSphere Portal and i2 Intelligence Analysis Platform are WebSphere applications, sharing the `portal.ltpa` key with the i2 Intelligence Analysis Platform WebSphere servers is sufficient. It is not necessary to use WebSEAL junctions.

To configure SSO between the i2 Intelligence Analysis Platform and IBM Intelligent Operations Center, perform the following steps:

1. Configure the i2 Intelligence Analysis Platform write server to use the IBM Intelligent Operations Center directory server.

2. Configure the i2 Intelligence Analysis Platform read server to use the IBM Intelligent Operations Center directory server.

3. Implement cross-cell SSO using portal LTPA between IBM Intelligent Operations Center and the i2 Intelligence Analysis Platform write and read servers.

4. Create the required i2 Intelligence Analysis Platform users and groups on the IBM Intelligent Operations Center directory server.

The following sections describe the steps.

5.2.1 Configuring the write server to use the IBM Intelligent Operations Center directory server

This section describes the first major step, which is to configure the i2 Intelligence Analysis Platform write server to use the IBM Intelligent Operations Center directory server.

The process of federating an LDAP directory in WebSphere Application Server is straightforward. The process is described in the IBM Knowledge Center. Search for *Adding an external repository in a federated repository configuration* at this website:

http://www.ibm.com/support/knowledgecenter/SSAW57_7.0.0/com.ibm.websphere.soafep.multiplatform.doc/info/ae/ae/twim_reposref.html?cp=SSAW57_7.0.0%2F8-9-30-2-1-3-13&lang=en

However, you must address a few challenges to federate the IBM Intelligent Operations Center Directory Server with i2 Intelligence Analysis Platform WebSphere Application Servers. The following steps address these challenges:

1. Set the default administrative user in IBM Intelligent Operations Center V1.5 to `waswebadmin` to avoid a conflict with an existing `wasadmin` user on the IBM Intelligent Operations Center directory server.

   > **Note:** The default i2 Intelligence Analysis Platform WebSphere administrative user is set to `wasadmin`. Therefore, before the IBM Intelligent Operations Center directory server can be federated on the i2 Intelligence Analysis Platform WebSphere servers, change the administrative username. Failure to do so will result in an ambiguous username, which will prevent logging in to the administration console on the servers.

   Follow these steps:

   a. Log in to the WebSphere Application Server administrative console on the i2 Intelligence Analysis Platform write server using the WebSphere Application Server administrative user name, `wasadmin`.

   b. Navigate to **Security** → **Global Security**. For Available Realm Definitions, select **Federated repositories** and click **Configure**. Change the Primary administrative username field to `waswebadmin`. See Figure 5-2 on page 168.
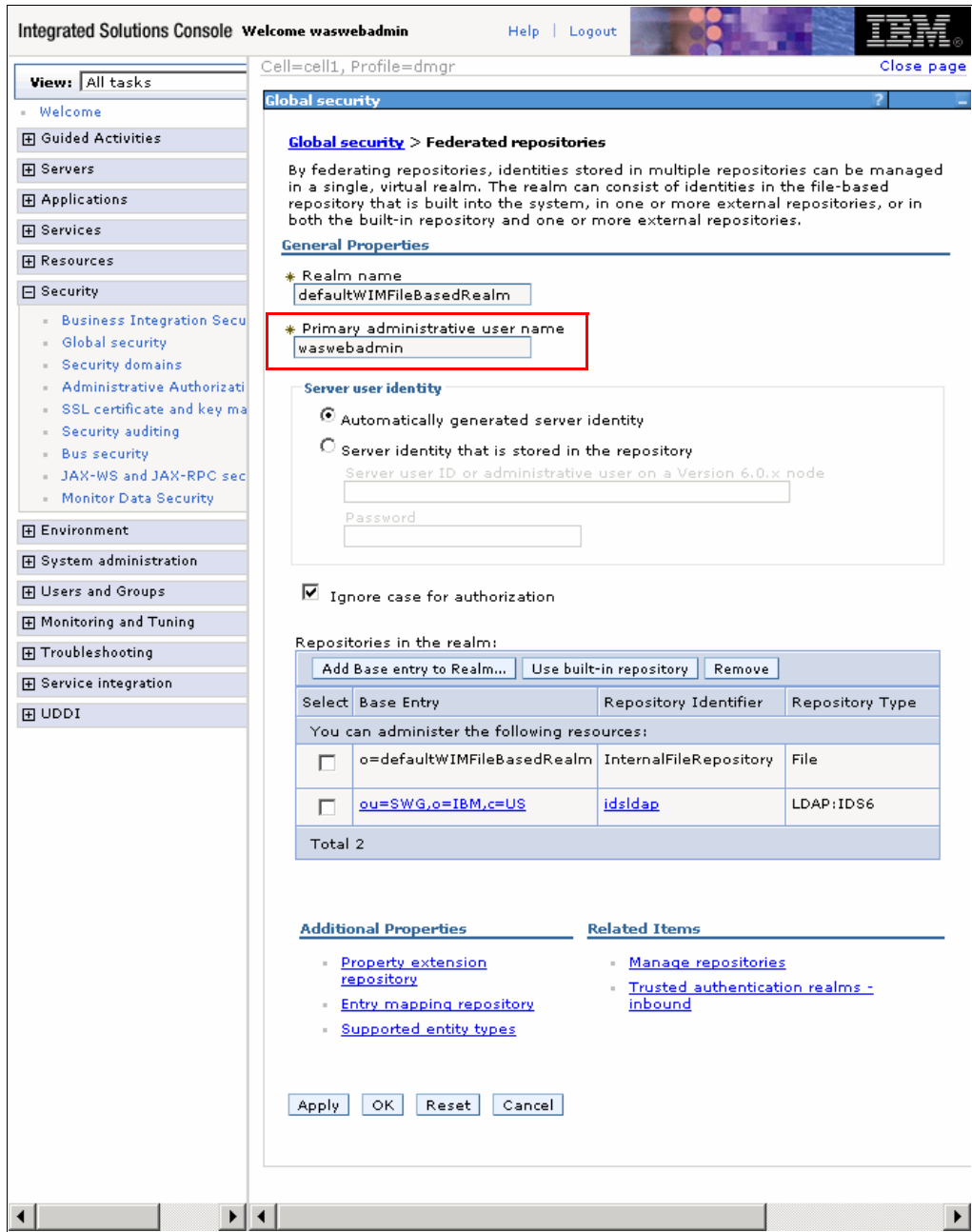
Figure 5-2   Setting the primary administrative user name

    c. Click **Apply** to save the changes to the master configuration.

2. Shut down WebSphere Application Server on the i2 Intelligence Analysis Platform write server.

3. On the i2 Intelligence Analysis Platform write server, navigate to the *<WebSphere_Home>*\profiles\ApolloWrite\config\cells\ApolloNode01CellWrite directory and make a backup copy of the `fileRegistry.xml` file. Back up the file in the same directory by renaming it to `fileRegistry.xml.bak`. This file is the WebSphere Application Server `InternalFileRepository` that contains the users and groups information.

4. Transfer the `waswebadmin` user account used as the WebSphere Application Server administrative user in IBM Intelligent Operations Center to the i2 Intelligence Analysis Platform write server.

   Copy the `fileRegistry.xml` file from IBM Intelligent Operations Center Application server, in the `/opt/IBM/WebSphere/AppServer/profiles/dmgr/config/cells/cell1` directory to the `<WebSphere_Home>\profiles\ApolloWrite\config\cells\ApolloNode01CellWrite` directory on the i2 Intelligence Analysis Platform write server.

5. Start the WebSphere Application Server on the i2 Intelligence Analysis Platform write server, and log in to the WebSphere Application Server administrative console as the `waswebadmin` user. The password is the same password that is used for the `waswebadmin` user in IBM Intelligent Operations Center.

6. Navigate to **Security** → **Global security**. In the Available realm definitions section, select **Federated repositories** and click **Configure**.

7. To add the IBM Intelligent Operations Center LDAP directory, click **Add Base entry to Realm**, and click **Add Repository** → **LDAP repository** in the next window. Copy the LDAP server and security settings from the `idsldap` repository entry on the IBM Intelligent Operations Center server. See Figure 5-3.

8. Click **Apply** and **Save directly to master configuration** to save the changes.

9. Copy the correct values from the Intelligent Operations Center to complete the fields in the General Properties section of the Repository Reference page.

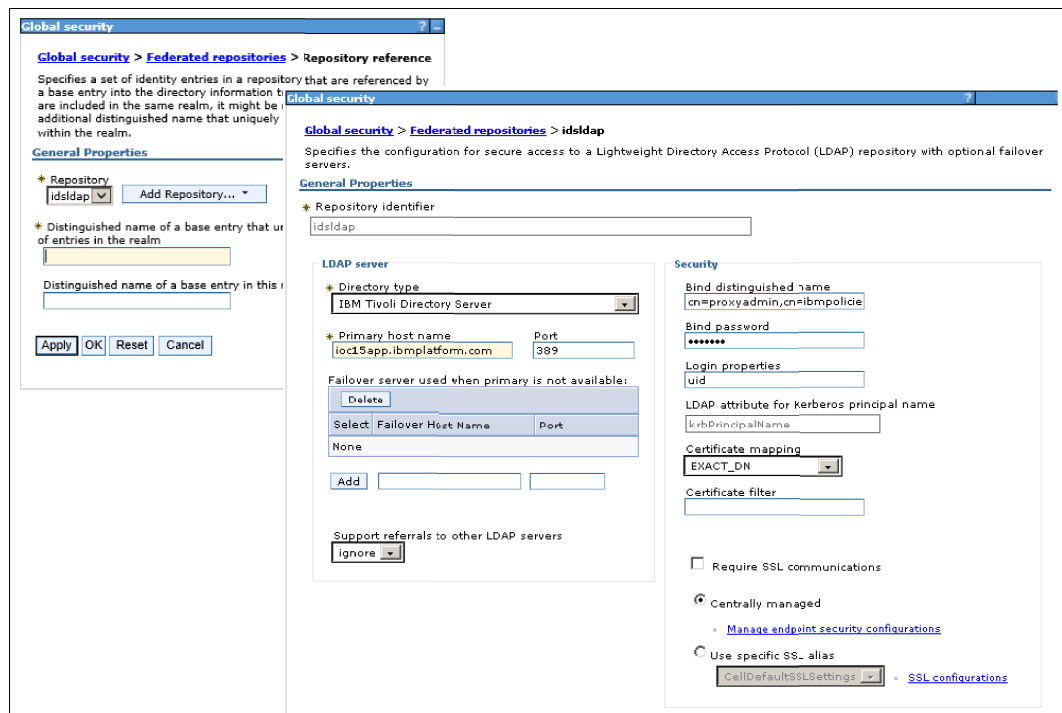10. Click **Apply** and **Save directly to the master configuration** to save the changes.

Figure 5-3   Adding and setting up the IBM Tivoli Directory Server LDAP registry

On the WebSphere Application Server administrative console for the i2 Intelligence Analysis Platform write server, navigate to **Security** → **Global security** → **Federated repositories** → **idsldap** and ensure that the LDAP entity types and Group attribute definition, Member attributes are the same as those on IBM Intelligent Operations Center. See Figure 5-4 on page 170 and Figure 5-5 on page 170.

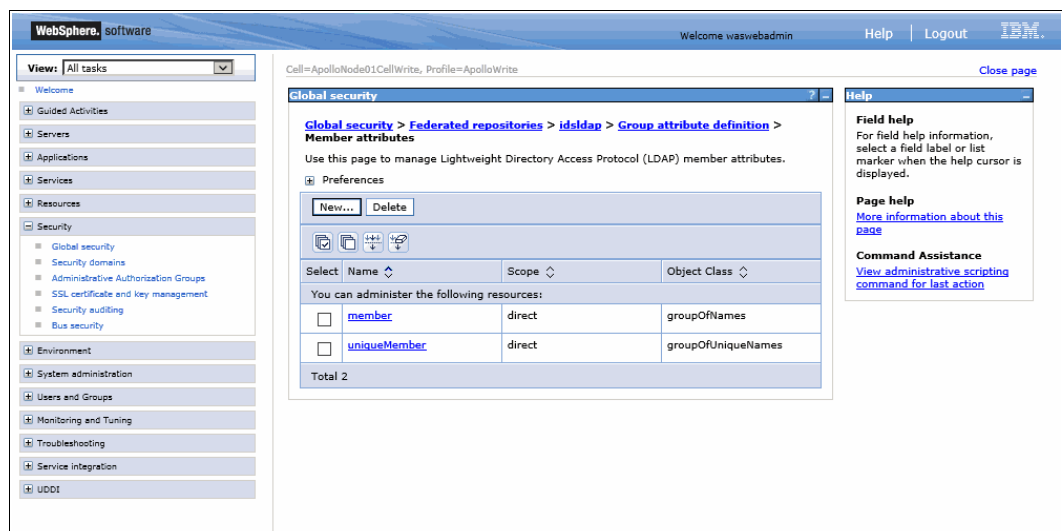*Figure 5-4   Setting up entity type and member attributes*

Figure 5-5   Setting up member attributes

5.2.2 Configuring the read server to use the IBM Intelligent Operations Center directory server

This section describes the second major step, which is to configure the read server to use the IBM Intelligent Operations Center directory server.

For this step, because the i2 Intelligence Analysis Platform write and read servers are installed in separate WebSphere cells, the procedure described in 5.2.1, "Configuring the write server to use the IBM Intelligent Operations Center directory server" on page 166 must be repeated on the i2 Intelligence Analysis Platform read server. After completing this step, proceed with the next section.

5.2.3 Implementing cross-cell SSO between the IBM Intelligent Operations Center and i2 Intelligence Analysis Platform servers

This section describes the third major step, which is to implement cross-cell SSO using portal LTPA between IBM Intelligent Operations Center and the i2 Intelligence Analysis Platform write and read servers.

In this step, you use the portal LTPA token to implement SSO between IBM Intelligent Operations Center and the write server and read server of i2 Intelligence Analysis Platform.

 Perform the following steps:

1. The `portal.ltpa` LTPA key used for authentication on IBM Intelligent Operations Center must be shared across the i2 Intelligence Analysis Platform WebSphere servers. The default location for the portal LTPA key is the `/opt/pdweb/etc/` directory on the Intelligent Operations Center application server. If this file cannot be found, a new copy of the portal LTPA key can be exported using the WebSphere administrative console. For more information about exporting LTPA keys, see the IBM Knowledge Center topic *Exporting Lightweight Third Party Authentication keys* at this website:

   http://www.ibm.com/support/knowledgecenter/SSAW57_7.0.0/com.ibm.websphere.nd.mu
   ltiplatform.doc/info/ae/ae/tsec_altpaexp.html?cp=SSAW57_7.0.0%2F3-3-0-0-2-1&lan
   g=en

2. Copy the `portal.ltpa` key from the IBM Intelligent Operations Center application server to a temporary directory on each of the i2 Intelligence Analysis Platform read and write servers. Open the WebSphere administrative console on the i2 Intelligence Analysis Platform write server, and navigate to **Security → Global security → LTPA**.

   In the Cross-cell single sign-on section, enter the password for `portal.ltpa` and the path to the local directory that contains the `portal.ltpa` file. Click **Import keys**. See Figure 5-6 on page 172.
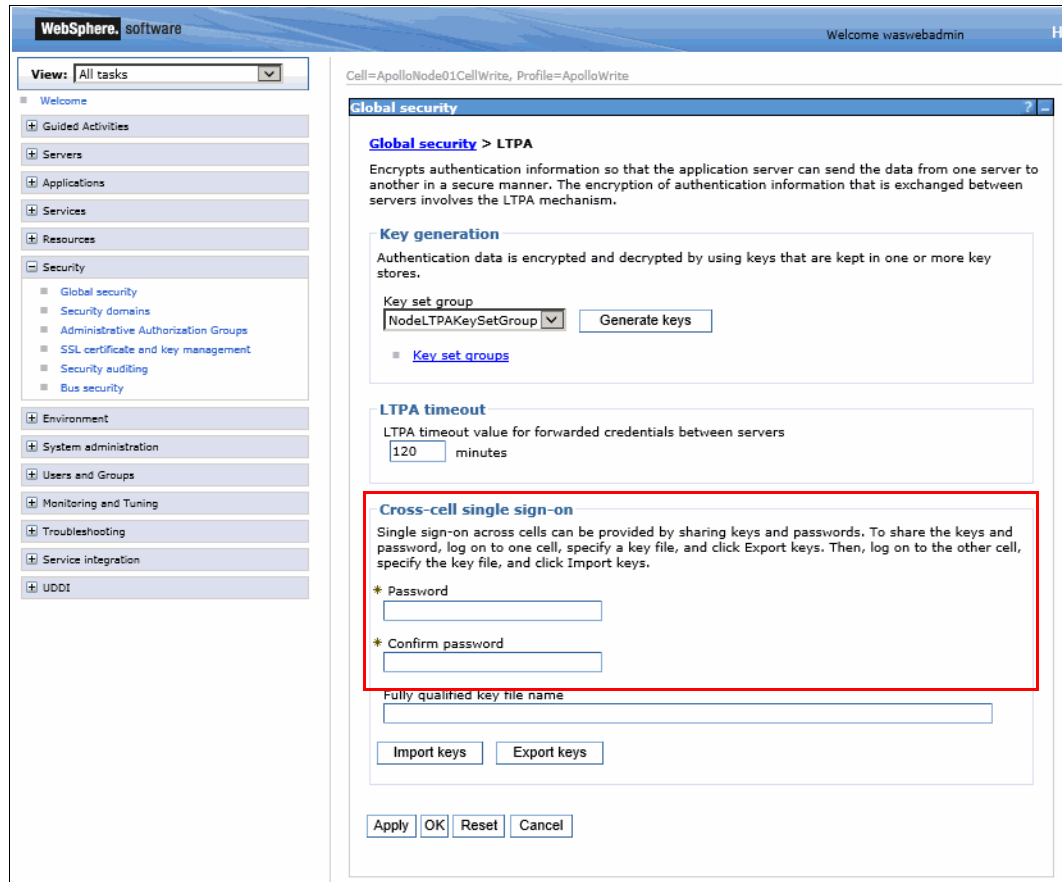
Figure 5-6   Setting up the Entity Type and Member attributes

3. Click **Save directly to the master configuration** to save the changes.

4. Repeat this process on the i2 Intelligence Analysis Platform read server.

5.2.4 Creating the required i2 Intelligence Analysis Platform users and groups on the Intelligent Operations Center Directory server

This is the fourth and last major step in the process to set up SSO between the IBM Intelligent Operations Center and the i2 Intelligence Analysis Platform.

The objective of this step is to ensure that the users and groups that you defined in i2 Intelligence Analysis Platform originally are moved to the IBM Intelligent Operations Center directory server. Doing so allows the two systems to use the same directory server.

This step is necessary if you have already customized the security schema of i2 Intelligence Analysis Platform as described Chapter 5, "Integrated Law Enforcement single sign-on" on page 163. In this case, you can use the Tivoli Directory Server Web Administration Tool to create those users and groups.

If you have not customized the security schema yet, we advise that you finish that activity first and then come back to create the users and groups directly in the IBM Intelligent Operations Center directory server using the Tivoli Directory Server Web Administration Tool.

If your goal is to deploy a running i2 Intelligence Analysis Platform quickly by using its default security schema, you can integrate readily with IBM Intelligent Operations Center by performing the following instructions.

The WebSphere file-based repository, `fileRegistry.xml`, which is used by the i2 Intelligence Analysis Platform when first installed, contains several groups that are mapped to the i2 Intelligence Analysis Platform security dimensions. For the i2 Intelligence Analysis Platform to function, these groups need to be re-created in the IBM Intelligent Operations Center directory server. The `fileRegistry.xml` file also contains two sample users, `demo` and `demosuper`. We suggest transferring these group memberships to the existing IBM Intelligent Operations Center users, rather than re-creating these users in the IBM Intelligent Operations Center directory server. The `demosuper` user can be replaced by the IBM Intelligent Operations Center `wpsadmin` user and, if the i2 Intelligence Analysis Platform `demo` user is required, it can be replaced with the `akelly` IBM Intelligent Operations Center test user.

The i2 Intelligence Analysis Platform `fileRegistry.xml` file contains the group definitions in Table 5-1.

Table 5-1   Group definitions in the fileRegistry.xml file

Group	Members
Administrator	demosuper
Analyst	demosuper demo
Head of Department	demosuper demo
HUMINT	demosuper demo
INA	demosuper demo
Linguist	demosuper demo
MET	demosuper demo
OSINT	demosuper demo
Reports Officer	demosuper demo
Restricted	None
Secret	None
SIGINT	demosuper demo
Top Secret	demosuper demo
Unclassified	demo

Several groups must be created, so it is easiest to create them using the WebSphere Application Server administrative console by performing the following steps:

1. Log in to the WebSphere Application Server administrative console on the IBM Intelligent Operations Center application server using the `waswebadmin` user.

2. Navigate to **Users and Groups** → **Group management**, and click **Create**.

3. Enter the name of the group and an optional description and click **Create**.

4. Repeat Step 3 for each group to be created.

5. After the groups are created, add users:

   a. Click the group name.

   b. Select the **Members** tab and click **Add Users**.

   c. Search for the required users.

   d. From the list, select the users to add. You can add Multiple users by holding down the Ctrl key.

   e. After you select the required users, click **Add**.

   f. Repeat Step 5 to add users to all of the groups.

6. Shut down the WebSphere Application Server on both the i2 Intelligence Analysis Platform write and read servers. Wait for both servers to shut down completely.

7. Restart the read server first, followed by the write server.

> **Note:** These steps illustrate how groups can be created on the IBM Intelligent Operations Center directory server using the sample security schema implementation that is delivered with the i2 Intelligence Analysis Platform. We suggest that you create your own groups after you have customized the security schema for your client.

SSO between the Intelligent Operations Center and the i2 Intelligence Analysis Platform is now configured, and the i2 Intelligence Analysis Platform is using the Intelligent Operations Center directory server for authentication.

For more information, search the IBM Knowledge Center for the *Implementing single sign-on to minimize Web user authentications* topic at this website:

http://www.ibm.com/support/knowledgecenter/SS7JFU_6.1.0/com.ibm.websphere.express.doc/info/exp/ae/tsec_msso.html?cp=SS7JFU_6.1.0%2F1-7-9-9-0-1-5&lang=en

5.3 Single sign-on with the client security infrastructure

i2 Integrated Law Enforcement is based on standard IBM products, which makes it highly customizable and provides the ability to integrate it with IBM or other vendor infrastructures that clients have in place. The key to its extensibility is the use of WebSEAL by the IBM Intelligent Operations Center as the primary point of entry into the system.

In situations where WebSEAL is used as the integration point for multiple applications, you might want to install a stand-alone WebSEAL server for IBM Intelligent Operations Center. For information about this subject, see the IBM technote, *Installing a stand-alone Tivoli Access Manager WebSEAL server for IBM Intelligent Operations Center*:

http://www.ibm.com/support/docview.wss?uid=swg21610849

IBM recognizes that many customers might want to integrate their IBM solutions with Microsoft products. The IBM Security Access Manager for Microsoft applications is a series of solutions that provide SSO and role-based authorization integrations for Microsoft products using IBM Tivoli Access Manager. These solutions can be used to integrate IBM i2 Integrated Law Enforcement when deployed in a Microsoft-based environment. Details of these solutions are in the IBM technote *IBM Security Access Manager for Microsoft Applications* at this website:

http://www.ibm.com/support/docview.wss?uid=swg24029517

One limitation of the IBM Security Access Manager for Microsoft Applications solution relates to Microsoft Office integration when WebSEAL basic or forms authentication is enabled. The issue is that creating, opening, and modifying documents prompt the user for authentication. This issue and a possible solution are described in the IBM developerWorks® article, *Enabling Microsoft Office Sharepoint Server Client Integration through Tivoli Access Manager e-business WebSEAL using Forms Authentication* at this website:

http://www.ibm.com/developerworks/tivoli/library/t-soscitam/

When i2 Integrated Law Enforcement is deployed in an environment that has multiple third-party applications, setting up a federated SSO environment using IBM Tivoli Federated Identity Manager might provide the optimum solution. Tivoli Federated Identity Manager is not part of the i2 Integrated Law Enforcement solution and needs to be purchased separately. The following article provides more details about the use of Tivoli Federated Identity Manager: *Integrating Tivoli Federated Identity Manager and Tivoli Identity Manager*:

http://www.ibm.com/developerworks/tivoli/library/t-tfim-tim/index.html?ca=dat

The integration possibilities for i2 Integrated Law Enforcement are nearly limitless. This publication covers only a small portion of this information. The IBM Software Services and Financial Operations teams can assist you in developing and deploying integration solutions to meet your needs. See *IBM Software Services for Smarter Cities and the Financial Operations* at this website:

http://www.ibm.com/software/city-operations/services/

6

Conclusion

This chapter wraps up the information presented in the book and highlights the key characteristics of a successful law enforcement solution. This chapter provides points to continue the work started with the information provided in this book.

177

6.1 Summary and next steps

Law enforcement agencies are no different from other business organizations in that they face the same challenges in their daily operations. Similarly, these challenges have negative impact on the attainment of their mission, which, for law enforcement, is to provide public safety to the citizens. And like other business organizations, these agencies need to be equipped with the correct infrastructure, tools, processes, and people to improve their performance. More importantly, what they really require is effective use of information that is accessible, insightful, and actionable.

Our main emphasis is *Integrated Law Enforcement*. That is, an improved and highly functional law enforcement operation can be achieved by integrating its infrastructure, its tools and processes, and its wealth of information that together provide a cohesive, responsive, and intelligent solution.

As a technologist, you have seen how the Integrated Law Enforcement is designed to lay down the foundation for providing the fundamental set of capabilities needed by law enforcement. You have learned the construction of a solution framework that lends itself to expanding its function with a set of extended capabilities and therefore providing its target consumers the flexibility to adapt to their changing needs and the demands of the environment in which they operate.

This extensibility can be expanded even more widely and significantly by augmenting and the solution with core and greater capabilities, such as more analytics from IBM Content Analytics and IBM Watson™, Identity Insights, SPSS, IBM Big Data and Analytics, content and information management, mobility, voice and video recognition, case and business process management, more sophisticated search engines, such as Vivisimo. These are just a few of the possibilities.

The other aspect that requires serious attention is customization of the solution to the needs of the clients. You have seen some of the basic customizations, such as security using the security models of the products, data modeling, personalization, user interfaces, and themes. These are all part of the client requirements.

Overall, the Integrated Law Enforcement is the right first step toward the right direction. It can be that layer in the architecture that glues everything together and as you build more capabilities in an incremental fashion, the more powerful the system becomes.

A

Snippet of a sample statement of work proposal

This appendix provides a snippet of a sample statement of work (SoW) for an IBM i2 Integrated Law Enforcement implementation project.

Do not treat this work proposal as self-sufficient. It describes the required work in detail, but it does not provide design specifics. Separate documents, such as a High-Level Design Document and Detailed Design Document, must be written to ensure that you and the client organization agree about how the solution will be implemented. These documents complement the Solution Architecture that your team prepared in 2.2.5, "Design solution architecture" on page 30.

Finally, a project plan is another document that is crucial for a successful i2 Intelligent Law Enforcement deployment. After the project manager identifies all of the requirements, tasks, and activities, the project manager will be able to prepare at least an initial draft of the project plan to establish the timeline for the project, key milestones' schedule, tasks, task owners, and so on.

Sample activities in the SoW

This section lists sample activities and provides examples of the tasks included in each activity.

> **Note:** This example is meant only to illustrate the type of activities and tasks that go into your proposal. Optionally, you can specify the sizing for each task in terms of person-week or person-hour. Activity 5, which involves the creation of the i2 COPLINK database based on the data sources that the client has identified, is a complex and specialized process that only the IBM i2 Lab Services or its IBM Business Partners can perform.

Activity 1: Project kickoff

The activity includes the following tasks:

Review the SOW documents for both software and hardware:

- ► Define the project team
- ► Coordinate the schedule
- ► Define the project plan

Deliverable materials

The deliverable of the project kickoff is the *Project Plan*.

Activity 2: Solution design

The activity includes the following tasks:

- ► Hold solution workshops to gather and validate requirements needed to create a high-level architecture, a solution design document that meets the requirements of the client, and a user acceptance test (UAT) document.

> **Note:** It is likely that you ran workshops beforehand as a prerequisite to writing this work proposal. It is also likely that you will need to hold additional workshops to get more detailed requirements after this work proposal is reviewed and approved.

- ► Review and design a solution based on the requirements gathered during the workshops, for example:
  - – Data sources, data ingestion, and data modeling requirements
  - – Security requirements
  - – Specific functional requirements
  - – Administration and maintenance requirements
  - – Performance, recovery, and other nonfunctional requirements

Deliverable materials

The following deliverables result from the solution design activity:

- ► *i2 Integrated Law Enforcement Solution Design document*
- ► *UAT document*

Activity 3: Deployment environment preparation

The owner of this activity depends on the situation. If your client already has the necessary environment, for example, the hardware, the client can own this task. If additional system or hardware needs to be procured, a third-party vendor might be involved. Your team might not get involved in this activity but clearly, there is high dependency on the completion of this task. The following tasks might apply in this activity:

► Gather system requirements and define specifications.

► If anything needs to be procured, start the process with the client; wait until the elements are delivered to the client and everything is in place.

► Install the prerequisite software and perform the prerequisite configurations:

    a. Set up and configure all servers required by BM i2 Intelligence Analysis Platform, IBM i2 COPLINK, and IBM Intelligent Operations Center. These servers can be physical or virtual.

    b. Connect these servers to the network environment.

    c. Assign host names to the servers.

    d. Apply all required security policies.

    e. Ask the client to create user accounts that can be used for the deployment with all the required permissions.

> **Important:** Create an i2 Intelligent Law Enforcement administrator account. Use this account for the deployment of the software. Always log on to the servers with this special account.

► Validate that the servers are accessible using the special account, host names, and that all security policies are in place.

Deliverable materials

The deliverables of this activity are the *Deployment Environment Specifications* and documentation.

Activity 4: i2 Intelligent Law Enforcement V1.0.1 deployment

This activity includes the following tasks:

1. Conduct basic hardware training.

2. Map a detailed deployment plan to the target hardware. The plan must address all aspects of deployment of all software, including but not limited to installation locations, disk space or partition allocation, naming conventions, virtual machines' configurations, host names, and subnets.

3. Deploy IBM i2 Intelligent Law Enforcement V1.0.1 onto the target deployment environment:

    a. Define scheduling and strategy for deployment by different teams.

    b. Deploy the IBM i2 Intelligent Law Enforcement Portal components:

        i. Install and configure the IBM Intelligent Operations Center V1.5 servers and the IBM Intelligent Operations Center V1.5 prerequisites.

        ii. Apply global configuration to the entire IBM i2 Intelligent Law Enforcement Portal component.

     iii. Perform installation verification tests for each server.

     iv. Perform a component system test.

c. Deploy the Intelligence Analysis component:

     i. Install and configure the IBM i2 Intelligence Analysis Platform V3.0.3.1 Write server and its prerequisites.

     ii. Install and configure the IBM i2 Intelligence Analysis Platform V3.0.3.1 Read server and its prerequisites.

     iii. Configure the entire IBM i2 Intelligent Law Enforcement Intelligence Analysis component.

     iv. Perform an installation verification test for each server.

     v. Perform a component system test.

d. Deploy the IBM i2 Intelligent Law Enforcement Policing component:

     i. Install and configure i2 COPLINK V4.8 Standard (Detect, Admin, Incident Analyzer, Visualizer, and Active Agent) and its prerequisites.

     ii. Install and configure the i2 COPLINK File Exporter for i2 COPLINK Information Exchange Package Description (IEPD) V4.8 and its prerequisites.

     iii. Install and configure the i2 COPLINK V4.8 Analysis Search Standard and its prerequisites.

     iv. Apply all required configurations for these i2 COPLINK modules.

     v. Perform an installation verification test for each module.

     vi. Perform a component system test.

e. Deploy the IBM i2 Intelligent Law Enforcement Console subcomponent:

     i. Install and configure the IBM i2 Integrated Law Enforcement Console portlet in the IBM Intelligent Operations Center.

     ii. Install and configure the Reporting module.

     iii. Install and configure the Situational Awareness module.

     iv. Install and configure the Intelligent Portal module.

     v. Install the i2 COPLINK Analysis Search, both the server side and the client side.

     vi. Perform installation verification tests for each of these components.

     vii. Apply IBM i2 Intelligent Law Enforcement global security by configuring and integrating Active Directory in the existing environment with the security component of IBM i2 Intelligent Law Enforcement.

     viii. Set up IBM Intelligent Operations Center to point to the preferred Geographic Information Systems (GIS) Server, for example, ESRI Arc GIS server.

     ix. Apply the suggested default performance tuning parameter values in each of the IBM i2 Intelligent Law Enforcement components and their subcomponents.

> **Note:** Follow the available recommendations at the time of the deployment for the components.

     x. Execute test suites to ensure that the IBM i2 Intelligent Law Enforcement functionalities work correctly.

     xi. Optional: Implement additional basic system administration tasks that are not provided as defaults but that are required and documented in the solution design.

Deliverable materials

The deliverable of this activity is the *Deployment and System Configuration document*.

Activity 5: i2 COPLINK database creation

This activity includes the following tasks:

1. Make arrangements with the client to gain remote access to the target hardware.

2. Create new i2 COPLINK maps to migrate and ingest the client's Record Management System (RMS) and Java Message Service (JMS) data sources into the database:

   a. Provide the client with data storage devices (encrypted USB drives) for the client to download their data source information onto and return them to you.

   b. Provide an interface to connect to the client's database on which the agency data source information resides to provide access to this information.

   c. Transfer the agency's data source information and data source database structures from the data storage device to your servers.

   d. Verify and analyze the transferred agency data source information and data source database structures.

   e. Map the agency data source information to the i2 COPLINK database, including the mapping of agency data codes to i2 COPLINK database codes.

   f. Standardize and consolidate the agency data source information into a data set to be stored in the i2 COPLINK database.

   g. Migrate the agency data source information, including the data set, document description, and entity description information, into the i2 COPLINK database.

3. Verify and review the completed data migration:

   a. Select a sample set of documents from the agency data source information, as determined by your team with input from each participating agency, to review and validate the conversion of the agency data source information into the i2 COPLINK database.

   b. Conduct, with assistance from the client, a side-by-side, field-by-field comparison of the selected sample set of documents, comparing the original documents from the agency data source to those same documents that are migrated into the i2 COPLINK database.

   c. Document any issues found as a *blocking* issue or a *non-blocking* issue and provide a tracking number for each issue found. A *blocking* issue will be resolved or a workaround will be provided before data verification. A *blocking* issue does not have to be resolved before data verification.

   d. Resolve or provide a workaround for the identified blocking issues.

   e. Present the issue resolutions to the client and each contributing agency.

   f. Provide a Data Verification Form to the client for sign-off.

4. Configure and test the intermediate server. An intermediate server must be installed first.

5. Refresh and verify data from the i2 COPLINK database.

   The purpose of this activity is to validate the issue resolutions and corrections to the i2 COPLINK database after data verification:

   a. Load a subset of the i2 COPLINK database in a test environment.

   b. Refresh the subset of i2 COPLINK database from the validated agency data source information.

   c. Verify the refresh operation within the i2 COPLINK database with the agency representative.

   d. Address any remaining non-blocking issues, unless otherwise agreed to.

   e. Provide a Data Refresh Verification form to the client for sign-off.

6. Verify and validate data.

Deliverable materials

The following deliverable materials result from this activity:

► *Signed-off on Data Verification Form*
► *Signed-off on Data Refresh Verification Form*

Activity 6: Develop, install, and test customizations and configurations

This activity includes the following tasks:

1. Scale out servers in the IBM i2 Intelligent Law Enforcement Analysis component, as needed.

2. Scale out servers in the IBM i2 Intelligent Law Enforcement Policing component, as needed.

3. Scale out servers in the IBM i2 Intelligent Law Enforcement Portal component, as needed.

4. Create schema modifications for the Analysis component, as needed.

5. Develop solutions for ingesting data into the Analysis Repository, if required.

6. Customize reports for both the Analysis and Policing, if required.

7. Configure the Situational Awareness subcomponent as specified in the Solution Design document.

8. Implement any user-experience customizations documented in the Solution Design document.

9. Implement IBM Intelligent Operations Center customizations and configuration, for example, the GIS server.

10. Implement client-specific security requirements in the Analysis, Policing, and Portal components, including the creation or reuse of an existing user registry, the users' roles, and access permissions within the IBM i2 Intelligent Law Enforcement domain. Ensure that single sign-on (SSO) works.

> **Note:** The creation of the user registry is the responsibility of the client with accurate guidance from you, where needed, based on the specific requirements of IBM i2 Intelligent Law Enforcement security.

11. Implement the integration of the client's user registry, for example, Microsoft Active Directory, to the IBM i2 Intelligent Law Enforcement global security solution.

12. Implement any other customizations agreed upon and specified in the solution document (functional and nonfunctional).

13. Conduct unit testing for each of the customizations and configurations that are introduced.

14. Run a suite of integration test cases.

Deliverable materials

This activity has no deliverable materials.

Activity 7: Deploy IBM i2 Analyst's Notebook Premium

This activity includes the following tasks:

1. Install i2 Analyst's Notebook Premium on the client machines identified by the client using an approach that was agreed upon by both parties.

2. Configure the Notebook so that it uses a server-based repository by pointing to the Analysis component of IBM i2 Intelligent Law Enforcement. This step might be subsumed by the previous step.

3. Test that a user can log in to the system through the i2 Analyst's Notebook and the Intelligence Portal using some of the user names in the user registry.

4. Perform other tests based on user roles.

Deliverable materials

This activity has no deliverable materials.

Activity 8: Knowledge transfer

This activity includes the following tasks:

1. Conduct the knowledge transfer of maintaining and operating the hardware.

2. Conduct workshop knowledge transfer about the use of each component of the IBM i2 Intelligent Law Enforcement solution.

3. Conduct the knowledge transfer of basic administrative procedures to the designated staff of the client.

Deliverable materials

This activity has no deliverable materials.

Activity 9: Run user testing

This activity includes the following tasks:

1. Assist the client in performing a final test of the integrated solution according to the defined User Acceptance Tests.

2. Identify and acknowledge problematic cases and push for their resolutions.

3. Ask the client to sign off on the User Acceptance Test document when the client is ready to accept the implemented solution.

Deliverable materials

The deliverable material of this activity is the User Acceptance Test document after the customer signs it.

Activity 10: Project closure

In this activity, you will run a project closure meeting to review the completion of all deliverables. Documents will be updated based on changes registered during the progress of the project.

Deliverable materials

The following deliverable materials are the result of this activity:

► *Completed Project Plan*
► *i2 Intelligent Law Enforcement Solution Design Document (finalized)*
► *Deployment and System Configuration Document (finalized)*

IBM

Redbooks

IBM i2 Integrated Law Enforcement Deployment Guide

(0.2"spine)
0.17"<->0.473"
90<->249 pages

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this paper.

IBM Redbooks

The following IBM Redbooks publications provide additional information about the topic in this document. Note that some publications referenced in this list might be available in softcopy only.

- ► *Integrated Law Enforcement: A Holistic Approach to Solving Crime*, REDP-5116.
- ► *IBM Intelligent Operations Center for Smarter Cities Administration Guide,* SG24-8061.
- ► *WebSphere Application Server V7.0 Security Guide*, SG24-7660
- ► *Implementing Kerberos in a WebSphere Application Server Environment*, SG24-7771

You can search for, view, download or order these documents and other Redbooks, Redpapers, Web Docs, draft and additional materials, at the following website:

ibm.com/redbooks

Online resources

These websites are also relevant as further information sources:

- ► IBM Intelligent Operations Center V1.5 Information Center

  http://pic.dhe.ibm.com/infocenter/cities/v1r5m0/topic/com.ibm.ioc.doc/ic-homepage.html
- ► IBM i2 Intelligence Analysis Platform Deployment Guide (SC27-5091-00)

  http://www.ibm.com/support/docview.wss?uid=pub1sc27509100
- ► IBM Tivoli Directory Server, Version 6.3 IBM Knowledge Center

  http://www-01.ibm.com/support/knowledgecenter/SSVJJU_6.2.0/com.ibm.IBMDS.doc/sysreq06.htm
- ► Tivoli Access Manager documentation

  http://publib.boulder.ibm.com/tividd/td/IBMAccessManagerfore-business5.1.html
- ► IBM i2 Intelligent Law Enforcement V1.0.1 Installation and configuration

  https://www-304.ibm.com/support/entdocview.wss?uid=swg27038695
- ► IBM i2 Intelligence Analysis Platform Deployment Guide (SC27-5091-00)

  http://www-01.ibm.com/support/docview.wss?uid=pub1sc27509100

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

IBM i2 Integrated Law Enforcement: Technical Architecture and Deployment Guide

Understand the technology and architecture of the solution

Deploy, configure, and integrate the product components

Expand and extend the basic capabilities

IBM i2 Integrated Law Enforcement is an IBM Smarter Cities solution that addresses the needs of modern-day law enforcement agencies. It is a solution framework that provides the individual capabilities of the products that comprise the solution and extended capabilities developed through the synergistic integration of those product components.

As a framework, IBM i2 Integrated Law Enforcement allows for the continuous expansion of capabilities by putting together building blocks within the system and integrating with new, external systems. In doing so, an organization can respond and adapt to its changing needs. Simply stated, the configuration, integration, and implementation of IBM i2 Integrated Law Enforcement and its components provide the tools for more effective law enforcement.

This IBM Redpaper publication explains the technology and the architecture on which the solution is built. Most importantly, this paper enables technical teams to install, configure, and deploy an instance of the i2 Integrated Law Enforcement solution using the product i2 Intelligent Law Enforcement V1.0.1.

This paper is targeted to solution architects, system and deployment engineers, security specialists, data management experts, system analysts, software developers and test engineers, and system administrators.

REDP-5130-00