

**District of Columbia Fiscal Year 2008  
Metropolitan Police Department of the District of Columbia (MPD)  
Office of the Chief Information Officer**

**STATEMENT OF WORK FOR  
DATA WAREHOUSE DESIGN SOFTWARE PURCHASE**

**1 SUMMARY**

- 1.1 The Metropolitan Police Department of the District of Columbia's (MPD) Office of the Chief Information Officer has a need to purchase a comprehensive Extract, Transform Load (ETL) program as well as a Business Intelligence (BI) System product for fifteen legacy systems and six new core systems. The ETL will provide a full range of capabilities for migration of data from numerous source systems in a variety of formats and structures to a Data Warehouse, and for the ongoing transfer of data from the source systems to the Data Warehouse for update and maintenance. The BI System will offer several tools to include, but not limited to: reporting, query, and analysis, dashboard and visualization, administration, and developer tools.

**2 REQUIREMENTS/DELIVERABLES FOR THE EXTRACT, TRANSFORM, AND LOAD (ETL) PROGRAM**

- 2.1 The following are the requirements for the ***Integration Platform and Repository*** portion of the ETL Program:
- 2.1.1 Provide a single unified enterprise data integration platform for discovering, accessing and integrating data from any business system, in any format.
  - 2.1.2 Provide tools to populate the repository from source systems.
  - 2.1.3 Extract source data seamlessly from disparate systems.
  - 2.1.4 Rapidly migrate legacy data to new platforms.
  - 2.1.5 Integrate data across multiple platforms.
  - 2.1.6 Ensure scalability to accommodate changing data volumes.
  - 2.1.7 Ensure high reliability.
  - 2.1.8 Support metadata analysis and reporting.
  - 2.1.9 Provide a single approach to accessing a wide variety of systems and formats.

- 2.1.10 Connect to source systems to access all enterprise data in various formats and structures such as: mainframe, structured data, unstructured data (MS Word, Excel spreadsheets, email, binary files, PDF files, etc), semi-structured data, relational data, Virtual Storage Access Method (VSAM), Incident Management System (IMS), Integrated Database Management System (IDMS), ADABAS database, etc.
- 2.1.11 Provide Change Data Capture and Real Time support from Mainframe, SQL Server, and Oracle database sources.
- 2.1.12 Maintain standardized and reusable data transformation logic across source data systems and formats.
- 2.1.13 Support division of data processing into subsets for parallel operation.
- 2.1.14 Automatically, reliably and optimally partition among CPUs in a multi-processor system.
- 2.1.15 Provide visual controls for range of partition schemes.
- 2.1.16 Support pipeline partitioning and data partitioning.
- 2.1.17 Produce Information Exchange matrix.
- 2.1.18 Synchronize access across multiple systems.
- 2.2 The following are the requirements for the **Metadata Management** portion of the ETL Program:
  - 2.2.1 Provide a metadata-driven architecture.
  - 2.2.2 Provide accurate metadata, describing how data actually exists.
  - 2.2.3 Provide global metadata services.
  - 2.2.4 Develop architecture for robust and reliable access to global metadata repository.
  - 2.2.5 Access metadata across local or WAN.
  - 2.2.6 Exchange metadata with different applications to support interoperability.

- 2.3 The following are the requirements for the **Data Profiling** portion of the ETL Program:
  - 2.3.1 Search and profile data from any source.
  - 2.3.2 Extract accurate profile information from source systems.
  - 2.3.3 Record information on content, quality, and structure understanding.
  - 2.3.4 Store information discovered in profiling.
  - 2.3.5 Support profiling to enable accurate data migration, integration and consolidation.
  - 2.3.6 Deploy profiling tools quickly using wizards.
  - 2.3.7 Provide information about sources to support data cleansing.
  - 2.3.8 Support profiling from multiple perspectives:
    - 2.3.8.1 Column profiling.
    - 2.3.8.2 Single-table structural profiling.
    - 2.3.8.3 Cross-table structural profiling.
    - 2.3.8.4 Data viewing.
    - 2.3.8.5 Data rules.
    - 2.3.8.6 Orphan analysis.
  - 2.3.9 Provide functions to reveal gaps, anomalies, inconsistencies and incompatibility within sources.
  - 2.3.10 Provide functions to create and manage a complete and accurate picture of content, quality and structure.
  - 2.3.11 Provide Web based dashboards, pre-built metric reports and change in content capabilities.
  - 2.3.12 Display profile result in interactive and batch modes.
  - 2.3.13 Provide a wizard-driven automated interface for initial profiling assessment and rerun profile mapping for continuing measurement of data quality improvement.
  - 2.3.14 Produce fully normalized model of source data.

- 2.3.15 Provide tools to create cleansing and transformation specifications for the data quality tool and repository manager.
- 2.3.16 Create complete, accurate picture of content, quality and structure of data.
  
- 2.4 The following are the requirements for the **Data Cleanse and Match** portion of the ETL Program:
  - 2.4.1 Provide tools to standardize, validate and correct data.
  - 2.4.2 Provide tools to standardize format and structure of output record.
  - 2.4.3 Provide tools to correct name and address.
  - 2.4.4 Provide tools to intelligently parse elements—ID individual components, standardize names, cleanse addresses (3d party ref), enhance with empirical data, standardize and correct.
  - 2.4.5 Provide a codeless environment for cleansing and integration.
  - 2.4.6 Provide tools to optimize runtime for data correction to provide high performance and scalability.
  - 2.4.7 Provide tools to define and apply business rules.
  - 2.4.8 Provide tools to normalize via pivoting arrays, rank and sort data, filter for subset processing, custom and stored procedures, Soundex and Metaphone algorithms.
  - 2.4.9 Provide tools to establish version control procedures and rules.
  - 2.4.10 Provide tools to establish quality assurance (QA) monitoring procedures.
  - 2.4.11 Provide tools to implement QA as a part of the interface to the legacy systems.
  - 2.4.12 Provide tools to auto access initial and ongoing quality of data for all locations and types.
  
- 2.5 The following are the requirements for the **Mapping** portion of the ETL Program:
  - 2.5.1 Map legacy data to Data Warehouse target data model.
  - 2.5.2 Provide mapping templates.

- 2.5.3 Design business/transformation logic in a Visio or Excel document to be used to automatically generate mapping.
- 2.5.4 Provide source to target mapping between different data structures.
- 2.5.5 Define necessary transformation specifications.
- 2.5.6 Compare actual and meta sources to target requirements to identify gaps, inconsistencies, redundant and inaccuracies.
- 2.5.7 Support creation of database design, produce DDL, derive correct normalized model.
- 2.5.8 View catalog of info, and produce graphic reports and dashboards.
- 2.5.9 Track view of data quality at multiple points in process/organization.
- 2.5.10 Provide capability to map data sets to GJXDM and District of Columbia Office of the Chief Technology Office (OCTO) architecture standards.
- 2.6 The following are the requirements for the **Developer Environment** portion of the ETL Program:
  - 2.6.1 Provide codeless, object-oriented visual development environment, dashboards and mapping templates.
  - 2.6.2 Provide drag and drop Graphical User Interface (GUI) for rapid development.
  - 2.6.3 Standardize and reuse definitions across platforms and projects without recoding.
  - 2.6.4 Provide a single-development environment to access all data types.
  - 2.6.5 Provide administration and mapping templates that integrate with common modeling tools and existing Java-based business logic.
  - 2.6.6 Provide built-in parallel processing transformations, robust expression language, extensible interfaces, and a fully functional visual debugger
  - 2.6.7 Provide robust and rich transformation objects.
  - 2.6.8 Provide a comprehensive transformation library of field, set-based and multi-step transformation objects to create and extend context and independent, reusable integration objects.
  - 2.6.9 Provide visual debugger and error handling tools.
  - 2.6.10 Provide tools to minimize error recovery logic required for design and runtime environments.

- 2.6.11 Provide developer configuration and performance tuning.
- 2.6.12 Provide mapping and administrative templates.
- 2.6.13 Support extended expression library and user-defined functions.
- 2.6.14 Promote reuse of definitions, transformations and workflow tasks.
- 2.6.15 Access and manage expression library through GUI based development environment.
- 2.7 The following are the requirements for the **Workflow Management** portion of the ETL Program:
  - 2.7.1 Define integration workflows.
  - 2.7.2 Manage and track complex flow of data across transactional systems and databases through visual tools.
  - 2.7.3 Schedule and execute data integration processes to business operations with a built-in time- or event-based schedule that enables automated operation.
  - 2.7.4 Provide functions to perform analysis of data down columns, across rows, and between tables.
  - 2.7.5 Develop data integration logic and workflow-objects-content independent of field, set based and multi-step.
- 2.8 The following are the requirements for the **General Capabilities** of the ETL Program:
  - 2.8.1 Provide standards-based platform accessible to any developer.
  - 2.8.2 Provide native access to mainframe and legacy applications.
  - 2.8.3 Support reusable mechanisms.
  - 2.8.4 Provide highly scalable batch, real time, incremental movements.
  - 2.8.5 Provide easy-to-use dashboards and mapping templates.
  - 2.8.6 Loose couple with SOA concepts DW/legacy systems.
  - 2.8.7 Ensure ability to run on Windows, Linux and Unix.
  - 2.8.8 Support connectivity and interoperability through web services.

- 2.8.9 Access data, metadata, workflows within an SOA stack through web services protocols using XML, Web Service Definition Language (WSDL), and Simple Object Access Protocol (SOAP).
- 2.8.10 Deliver data to consumer applications in the correct form, with appropriate semantics, through webs services hub.
- 2.8.11 Provide a 2eb-based administration console
- 2.8.12 Provide Java transformations to support.
- 2.8.13 Import, create and debug Java-based transformations.
- 2.8.14 Read/write data to/from Java applications.
- 2.8.15 Ensure sustainability for approximately 50 concurrent users at any given time.
- 2.8.16 Support partitioning.
- 2.8.17 Pushdown optimization.
  
- 2.9 The following are the requirements for the **Security** portion of the ETL Program:
  - 2.9.1 Allow the application administrator or other authorized user to create, edit and delete user security profiles. These actions must be logged for auditing purposes.
  - 2.9.2 Include a sophisticated security module that controls all vendor-supplied applications forms, reports, interfaces and other input or output mechanisms.
  - 2.9.3 Provide security for both the operating system and database management software to prevent unauthorized persons from accessing data by circumventing the application.
  - 2.9.4 Require users to login using their user name and password prior to having access to any application features.
  
- 2.10 The following are the requirements for the **System Capabilities** of the ETL Program:
  - 2.10.1 Provide high availability, failover and seamless recovery mechanisms.
  - 2.10.2 Provide support for grid computing.

2.10.3 Support dynamic partitioning

**3 REQUIREMENTS/DELIVERABLES FOR THE BUSINESS INTELLIGENCE (BI) SYSTEM**

3.1 The following are the requirements for the **Core Platform** of the BI System:

3.1.1 Build around a core capability for deployment and management of BI tools, reports, and analytics.

3.1.2 Provide a flexible and scalable business intelligence platform. The primary source of information will be a comprehensive department-wide Data Warehouse in which the data are stored in a relational database.

3.1.3 Build on a service-oriented architecture to extend BI to the Data Warehouse and potentially to any application or process in any environment throughout MPD.

3.1.4 Facilitate integration with existing IT infrastructure by accessing any data source and aggregating all BI under a semantic layer.

3.1.5 Provide the capability to embed BI in any application or process in any environment.

3.1.6 Tailor the user interface specifically for law enforcement members, and as such, optimized for high productivity.

3.1.7 Provide the capability for heterogeneous joining of data from the Data Warehouses, Data Marts, and Operational Systems.

3.2 The following are the requirements for the **Query and Analysis** portion of the BI System:

3.2.1 Provide an Online Analytical Processing capability that supports interactive analysis of data from multiple perspectives.

3.3 The following are the requirements for the **Data Mining portion** of the BI System:

3.3.1 Provide a data mining capability to extract information from a database by utilizing software that can isolate and identify previously unknown patterns or trends in large amounts of data.

3.3.2 Provide a capability to integrate Data Mining Models into reports and analyses.

3.4 The following are the requirements for the **Report Services** portion of the BI System:

- 3.4.1 Provide a fully interactive enterprise reporting engine, featuring drag and drop report inter-actions.
- 3.4.2 Ensure that the system is capable of generating pre-determined standard reports and user-specified views of data to be invoked easily by users.
- 3.4.3 Provide access to the information reliably and securely.
- 3.4.4 Support interactive reporting for data sorting, parameter filtering and report reformatting.
- 3.4.5 Provide a web interface for interactive reporting, WYSIWYG report design over the web, direct access to metadata, and reporting by joining heterogeneous databases.
- 3.4.6 Provide an integrity manager data validation tool to automatically compare and verify the consistency of reports as changes are made within the BI system.
- 3.5 The following are the requirements for the **Dashboard/Visualization** portions of the BI System:
  - 3.5.1 Provide a point-and-click data visualization tool to create actionable business dashboards connected with secure, live connections to BI platforms or to live MPD data.
  - 3.5.2 Include interactive analytics and visually-based summaries of business data that show at-a-glance conditions with a point-and-click.
  - 3.5.3 Provide consolidated views of key metrics and information.
  - 3.5.4 Create a dashboard builder tool to consolidate dashboard views for viewing multiple business intelligence contents.
  - 3.5.5 Support the set-up of visual environments to explore relationships, trends and timelines.
  - 3.5.6 Ensure presentations created from these tools support dynamic charts and graphics...
  - 3.5.7 Include a library of widgets for ease of operation.
- 3.6 The following are the requirements for the **Enterprise Information Management (EIM)** portion of the BI System:
  - 3.6.1 Provide an EIM capability to deliver data that is integrated, accurate and timely across the enterprise for business transaction processing, business intelligence, data warehousing, data migration or master data.

- 3.7 The following are the requirements for the **Administrator** portion of the BI System:
- 3.7.1 Include an administrator suite of tools that provide comprehensive systems management environment, including a rapid development tool that maps the physical structure of the database into a logical business model and stores it in a centralized metadata repository.
- 3.8 The following are the requirements for the **Developer Kit** portion of the BI System:
- 3.8.1 Provide a developer kit with application modules, software development tools and web services to facilitate user reporting, analyzing and monitoring.
- 3.8.2 Include a tool that maps the physical structure of the database into a logical business model and stores it in a centralized metadata repository.
- 3.9 The following are the requirements for the **General Capabilities** portion of the BI System:
- 3.9.1 Provide a standards-based platform.
- 3.9.2 Ensure the system is highly scalable.
- 3.9.3 Implement a system that is run on Windows, Linux and/or Unix.
- 3.9.4 Provide connectivity and interoperability through web services.
- 3.9.5 Implement a web-based administration console.
- 3.9.6 Ensure the system is able to sustain approximately 50 concurrent users at any given time.
- 3.10 The following are the requirements for the **Security** portion of the BI System:
- 3.10.1 Allow the application administrator or other authorized user to create, edit and delete user security profiles. These actions must be logged for auditing purposes.
- 3.10.2 Include a sophisticated security module that controls all vendor-supplied applications forms, reports, interfaces, and other input or output mechanisms.
- 3.10.3 Provide security for both the operating system and database management software to prevent unauthorized persons from accessing data by circumventing the application.
- 3.10.4 Require users to login using their user name and password prior to having access to any application features.

**4. Ongoing System Maintenance**

All first-year maintenance must be covered under the warranty. The vendor shall provide on-going maintenance for all new product releases, patches, and upgrades. The vendor shall provide under on-going maintenance for all new product releases. The vendor shall identify the ongoing maintenance charges of all third-party products (including the OS) included in the proposal.

**5. Warranty for Vendor Furnished Software**

The Vendor shall warrant that all Vendor-furnished software and interfaces are fully operational, efficient, and free from defect. The Vendor will be responsible for correcting all malfunctioning software in a timely manner at no additional cost to the District of Columbia and/or the Metropolitan Police Department for the life of the system.