

Strategic Rationale for the Standards

The District's information technology product standards are designed to support the major goals of the city, as defined by Mayor Anthony A. Williams in his strategic plan. These goals include increased access to resident, business, and tourist services; streamlined management; efficient use of resources, increased revenue, and fee-for-service opportunities.

To meet these challenges, the Chief Technology Officer spearheaded an initiative to create a technical architecture framework for the District of Columbia. This technical architecture identifies the criteria District agencies should consider in the selection of hardware, software, and operating systems. Such products must conform to the District's standards and satisfy IT architectural requirements that promote growth, integration, interoperability, and other qualities.

The product standards detailed in this guide are documented agreements that contain technical specifications or other precise criteria, such as rules, guidelines, or definitions of characteristics. These standards must be followed consistently District-wide to ensure that information technology materials, products, processes, and services serve their purpose. However, the need for a coherent, District-wide architecture must be balanced with the specific, mission-critical needs of each agency. Products and services should also complement the budgetary and procurement procedures of District IT, budget, and procurement departments.

This section provides an overview of the District's technical architecture, the components, qualities, characteristics, and architectural requirements of the IT infrastructure, and the strategic rationale behind the District's product standards.

OVERVIEW OF THE DISTRICT'S IT ARCHITECTURE

Information technology architecture provides the rules, standards, and structure to promote consistent technology solutions throughout the District. The architecture defines the District's key business processes and directions, current and future technology, applications, and information, and provides a plan to move from the current IT environment to a future one. The architecture allows the District to, among other things:

- Base technology decisions on how they would affect the District as a whole
- Align business and IT objectives and improve communication between business and IT
- Ensure IT projects adhere to product and operational standards
- Reduce acquisition and support costs of IT products and services
- Improve consistency, quality, and timeliness of data that is shared District-wide
- Adhere to Office of Management and Budget and General Services Administration oversight requirements

The IT architecture plays a critical role in enhancing the return on technology investments. It provides the decision-making framework for IT planners and developers to establish guidelines for the individual IT resource owners and users.

“Washington has the potential to be the world capital for information arbitrage.”

Sharon Ambrose
Councilmember, Ward 6
Committee on Consumer and
Regulatory Affairs

ALIGNMENT OF THE IT ARCHITECTURE WITH THE IT STRATEGIC PLAN

The District's IT architecture vision is aligned with the District's IT strategic plan. The strategic plan views technology as a powerful enabler and innovator in the mission to provide improved services for DC residents.

The District's 2002-2004 Information Technology Strategic Plan identifies the District's most significant agency-level and cross-agency infrastructure needs as data access and data accuracy requirements, system security, and next-generation functional software support. The District's major application needs include electronic document management, business intelligence systems to combine separate data sources for better decision-making, web-based applications to support electronic services for residents, businesses, and visitors, and projects that integrate automation to improve business processes.

Among several goals, the plan identifies the need to standardize IT policies and procedures, centralize the IT procurement process, and build the following enabling infrastructures:

- Wide Area Network (WAN)
- Telecommunications
- Email/Web Infrastructure
- Business Process Re-engineering (BPR)
- Wireless Communications
- Metropolitan Police Department (MPD) Data Networking
- Geographic Information System (GIS)
- Unified Communications Center (UCC)
- Data Warehousing
- Data Center Consolidation
- Seat Management
- Executive Office of the Mayor (EOM) Outreach
- E-government Infrastructure

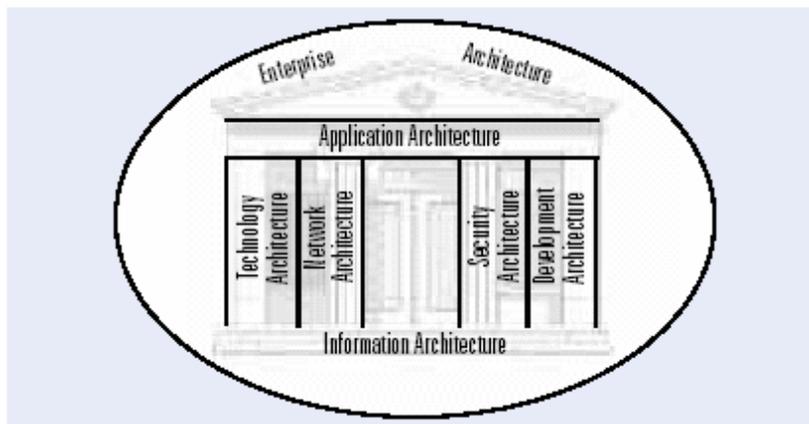
THE BUILDING BLOCK APPROACH TO IT ARCHITECTURE

The District's IT architecture is defined by six components: application, technology, network, security, development, and information. Information technology architecture must have structure, just as a building must have a foundation, support walls, and a roof. The diagram on page 61 illustrates this comparison between the structure of a building and the IT architecture.

The *Enterprise Architecture* encompasses the total IT structure and provides consistent standards throughout the District's agencies. It exists at the highest level of the architecture, and is only constrained by the strength and quality of the other architectures.

These structural supports must be maintained to preserve the well-being of application architecture, information architecture, technology architecture, network architecture, security architecture, and development architecture.

Information Technology Architectures



Application architecture provides an interface between information and the enterprise. Through applications, information is packaged and formatted to support business functions and data.

Technology, Network, Security, and Development architectures provide technical integrity, just as walls strengthen the structure of a building. These structural supports must be maintained to preserve the well being of the entire architecture.

Technology architecture provides standards for computer hardware, computer peripherals, and operating systems.

Network architecture provides structures, controls, and support activities that are the foundation of the enterprise-wide network, enabling it to operate as a coordinated whole.

Security architecture controls access to and preserves the integrity of the District's data.

Development architecture employs best development practices that promote speed, accuracy, and maintainability for the development and implementation of software systems.

Information architecture comprises data models that support integrated business systems. Like the foundation of a building, it must be well constructed and able to support the rest of the architecture.

Building Blocks

There are four major components, or building blocks, in the plan to turn the District into a city of access.

The first building block, Basic Infrastructure, allows District employees to conduct their primary tasks. The next incremental building block, Information Access, gives employees and residents access to data on the web. The Interactive Services building block allows employees and residents to submit queries to, and receive responses from, District information resources. Finally, the Commerce building block adds e-commerce capability to the infrastructure, allowing employees and residents to complete secure financial transactions with automated payment processing.

The following diagrams illustrate the architecture for each building block: Basic Infrastructure, Information Access, Services, and Commerce.

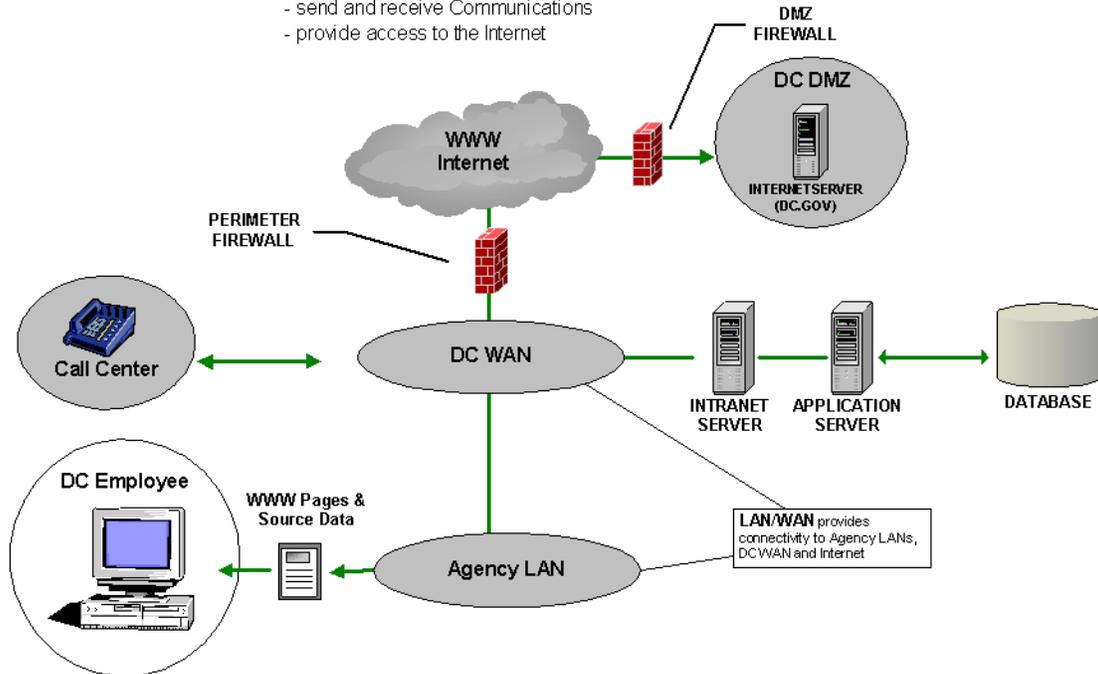
“Technology is at the center of our ability to energize the District’s economy, to explode customer service, to focus community involvement, and to squeeze the most we can from every nickel.”

Kathleen Patterson
Councilmember, Ward 3
Chairperson,
Committee on the Judiciary



Enables Employees to:

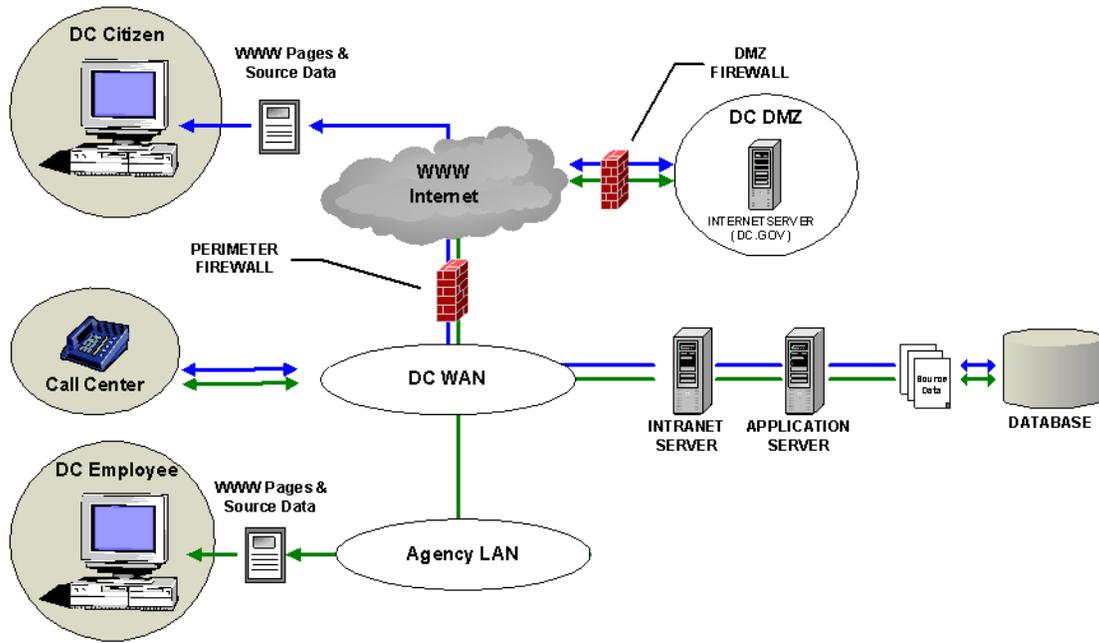
- provide Primary LAN/WAN Access Services
- send and receive Communications
- provide access to the Internet





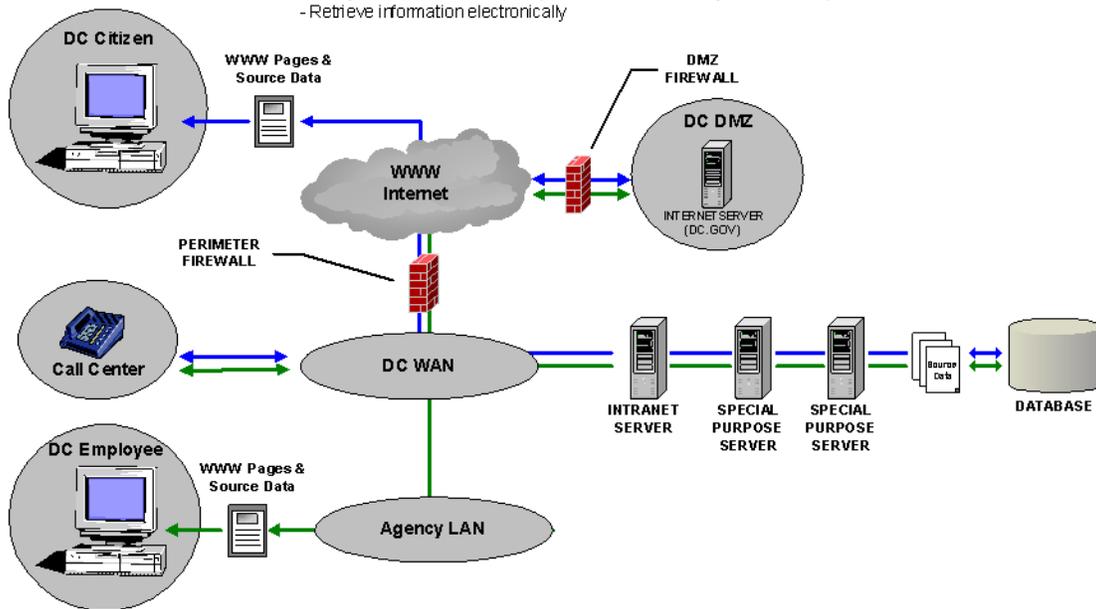
Enables Citizens and Employees to:

- Allows persons to view District Facts (who, what, when, where, why)





Enables Citizens and Employees to:
 - Query data and obtain results (vital records, availability, reservations)
 - Retrieve information electronically



ARCHITECTURAL REQUIREMENTS

The District's architectures are evaluated based on their ability to deliver the following:

- Information services for the District's residents and internal customers
- The critical IT foundation for the District
- Expandable enterprise architecture
- Scalable, flexible, maintainable environment
- Capacity for growth
- Data maturity, integration, and accessibility
- Best practices technology
- Methodology and standards compliance

Architecture processes begin with an understanding of the business and the data that constitutes its information infrastructure. The infrastructure must evolve from an understanding of and knowledge about the information. The District requires management information to be:

- Accurate
- Accessible in a timely manner
- Formatted in a usable way
- Responsive to rapidly changing convergence requirements
- Shared across the District's agencies with other operational and business support systems

Quality	Architectural Requirements
Business Systems Integration and Information Access	<ul style="list-style-type: none"> • Provide seamless integration with other business systems—it should be possible for the systems to share common data among all the business support systems, pass data when required, and coexist with the desktop environment • Access various levels of its own data and facilitate data transfer to a decision support system • Communicate with external devices • Access data for reports, analysis, and <i>ad hoc</i> query
Enterprise Standards	<ul style="list-style-type: none"> • Create a common look-and-feel throughout the system and ideally with other business support systems • Provide consistent interfaces, operations, platforms, applications, and development environments across the enterprise

INFORMATION ARCHITECTURE REQUIREMENTS

Information architecture identifies the major kinds of data that support management functions. From a business perspective, good data architecture should have minimal changes over time, have sound fundamental data definitions, and remain flexible with respect to market changes.

To achieve this flexibility, data is defined independently of who uses the data, where the data is used or stored, when it is used, the sequence in which it is used, and which applications and technologies manage the data.

A sophisticated data architecture that supports change brings the advantages of accelerating communication flow and of improving management decision-making and control functions. Equally important, a robust data architecture has powerful potential to increase service quality levels and shorten service delivery timelines.

Quality	Architectural Requirements
Data Model	<ul style="list-style-type: none"> • Allows services to be maintained as distinct entities and to be related in accordance with the District's information requirements • Is developed and maintained rigorously • Links to the business objectives and integrates with the relational database and dictionary • Contains all data required to support the current business and the defined strategy to incorporate new service offerings • Efficiently and effectively enhances business productivity and application performance
Relational Database	<ul style="list-style-type: none"> • Uses an industry-wide accepted standard • Requires minimal denormalization for performance considerations • Exploits hardware capabilities • Maintains critical data rules • Has wide third-party acceptance and support • Demonstrates performance under high volume
Data Dictionary	<ul style="list-style-type: none"> • Records narrative definitions for all tables and columns • Assists with impact analysis • Links to the relational database management system (RDBMS) and data model
Data Management	<ul style="list-style-type: none"> • Synchronizes multiple databases • Executes and monitors transactions • Synchronizes local and remote databases • Allows for logical distribution • Includes data import/export functionality

TECHNICAL ARCHITECTURE REQUIREMENTS

This architecture identifies the major kinds of technologies needed to provide a sophisticated and efficient environment for the applications that manage data. Typical technologies which the District will employ to achieve internal efficiencies and external competitive advantages are client/server technologies, common graphical interfaces, data storage relational technologies, data access SQL technologies, and systems development methodologies such as object-oriented techniques, information engineering methods, and repository tools.

Quality	Architectural Requirements
Scalability/ Performance	<ul style="list-style-type: none"> • Able to size the hardware according to its intended use • Scalable through expansion of platform size and features without performance degradation • Able to add platforms and specialize their use (such as establishing separate servers for each vertical product) without performance degradation • Able to position systems geographically with the ability to reallocate/redesign placement of hardware without performance degradation • Able to configure for specific workload characteristics at various geographic locations, even with arbitrarily high volumes of customers, services, or real-time transactions • Able to scale to symmetric multi-processing (SMP) system architectures

Openness and Portability	<ul style="list-style-type: none"> • Based on established open technologies and platforms • Supports software and hardware products based on industry, not proprietary standards • Able to utilize open operating systems and third-party software administration, end-user analysis, database administration, and performance software products • Shares robust set of support tools and utilities with operating system • Able to add new or revised third-party software to improve the scope of the system over time • Compatible with various hardware devices, adhering to open standards and providing scalability
Client/Server	<ul style="list-style-type: none"> • Able to partition data, applications logic, and presentation activities among different processors and between different servers and clients • Able to access data in a multi-server environment and produce integrated information • Able to provide centralized customer service support by providing access to geographically dispersed data over multiple remote sites
Graphical User Interface	<p>On the desktop:</p> <ul style="list-style-type: none"> • Adheres to industry standards • Able to run on multiple client environments • Maintains common look and feel
Interoperability	<ul style="list-style-type: none"> • Must be a consideration of the various products

APPLICATIONS ARCHITECTURE REQUIREMENTS

Applications architecture defines the applications required to support the District's business functions and to manage its information.

The applications architecture contains high-level descriptions of the capabilities and benefits of all the applications that support the District. It identifies the business functions supported by the applications, the data created, updated, or referenced by the applications, and the current applications affected.

From a business perspective, the purpose of the applications architecture is to provide an organized set of applications to improve the District's overall effectiveness and productivity. This will provide strategic and differentiated customer services to improve operational and cost advantages.

Quality	Architectural Requirements
Scalable	<ul style="list-style-type: none"> • Number of clients should be able to grow significantly with a linear increase in network and server load • Application logic can be distributed and redistributed among the host, servers, and clients in response to a dynamic business environment and technological advancements • Amount of data should be able to increase substantially without degradation of performance and response time

Flexible	<ul style="list-style-type: none"> • Application logic should be table-driven and meet system and business performance requirements • Programs easily support management decisions • Programs are structured and documented • Programs should be designed for ease of maintenance and impact analysis • New service technologies can be accommodated
Modular	<ul style="list-style-type: none"> • Well-structured and organized programs minimize impact on other modules • Ability to reuse business logic • Business processes flow quickly
Production Volume Proven	<ul style="list-style-type: none"> • Applications should have a satisfactorily high-volume production performance history
Application Adaptability	<ul style="list-style-type: none"> • Applications should be able to be quickly transformed to support the District's direction

NETWORK ARCHITECTURE REQUIREMENTS

Network architecture provides those structures, controls, and support activities that are the foundation of the enterprise-wide network, enabling it to operate as a coordinated whole. This architecture allows efficient implementation, monitoring, and management of every element of the enterprise-wide network.

From a business perspective, a flexible network architecture allows efficient performance. It allows computing close to customers, provides greater user productivity, simplifies coordination, administration, and distribution of information, and enhances throughput, availability, and response times.

Quality	Architectural Requirements
Topology	<ul style="list-style-type: none"> • Compatible with TCP/IP for all data communication • Supports most LAN and WAN configurations • Provides control of remote report printing capabilities across the LAN and WAN • Provides network monitoring capabilities on the LAN and WAN • Accommodates a multi-tiered architecture • Accommodates standard and emerging technologies with support standards such as T1, T3, frame relay, ATM, and ISDN
Network Management	<ul style="list-style-type: none"> • Offers monitoring, measurement, and analysis tools to access performance • Provides proactive problem detection of device status before service is compromised • Provides network management standards such as Simple Network Management Protocol (SNMP), and Remote Monitoring (RMON, RMON2) • Provides capability for a Network Command Center • Performs across most hardware devices

SECURITY ARCHITECTURE REQUIREMENTS

A security architecture controls access to and preserves the integrity of the District's data. It provides authentication procedures and access-control routines, supports encryption, and maintains authorization facilities at the terminal, data, application, and network levels.

From a business perspective, multiple levels of security architecture provide proper information access, enable customized, comprehensive capabilities for every user, increase employee productivity, accelerate tactical customer responsiveness, and improve management decision-making and control.

Quality	Architectural Requirements
Security	<ul style="list-style-type: none"> • Provides flexible design to accommodate the District's future security policy • Allows biometrical (e.g., smart cards, fingerprinting) use for systems administrator(s) • Allows data access control through the RDBMS • Controls access to client and servers by password • Controls access to application functions by user identification • Controls access to servers by using a "trusted client" mechanism • Allows for operational and configurable audit trails • Supports encryption • Enables completion/rollback processing at the transaction level • Provides adequate database security tools and database administration security • Provides backup and recovery control • Allows for various levels of security • Provides on-site centralized and coordinated Domain Name System (DNS) management • Supports distributed and redundant database access • Provides full-function off-site storage as well as on-site fireproof storage • Provides full-function environmental monitoring (e.g., ventilated, well-lighted, and protected against glass breakage and water damage) for the computer rooms • Capability for single-logon (synchronized) access verification across all applications

DEVELOPMENT ARCHITECTURE REQUIREMENTS

Development architecture combines best development practices that leverage the speed, accuracy, sophistication, maintainability, and support of developed and implemented software. The best practices include planning and data design software, data dictionary software, presentation graphics software, and project management tools.

From a business perspective, these architectures should use tools and methods that provide:

- Higher reliability
- Greater data sharing
- Greater usability
- More flexibility
- More integration
- Improved integrity
- Reduced development costs and maintenance costs

Quality	Architectural Requirements
Development Environment	<ul style="list-style-type: none"> • Utilizes structured software engineering principles and a software lifecycle, including a robust testing methodology • Utilizes an integrated approach (planning through maintenance), including the use of an application generator and an application repository • Includes project management methodology using standardized project planning and estimating tools for software development, enhancement, and maintenance • Provides hardware capacity and a configuration management process • Supports a planned application development configuration control process • Able to manage multiple versions of the application • Contains development, testing, and production environments • Includes unattended backups • Provides for maintenance support adherence to the same standards as development • Provides a well-defined release management process: alpha and beta testing • Able to develop versions of software in a client/server environment
Support Environment	<ul style="list-style-type: none"> • Provides online help instructions, computer-based training, and experienced trainers with quality course material • Able to do impact analysis of requested changes • Provides a streamlined change management process with a priority-setting policy and a means to evolve toward the District's future Systems architecture • Includes an effective and streamlined problem-resolution process and • Integrated help desk function • Sponsors a well-managed user group that provides for an exchange of ideas and a forum to resolve problems • Performs at level 3 of the Software Engineering Institute Capability Maturity Model (CMM) regarding the following processes: <ul style="list-style-type: none"> - Project planning - Project tracking and control - Quality assurance - Requirement management - Subcontractor management - Measurements - Acceptable criteria
Vendor Evaluation	<ul style="list-style-type: none"> • Responsive to user needs • Financially viable organizations • Capable of comprehensive future planning