



# DELAWARE STATE-WIDE INFORMATION TECHNOLOGY AND ARCHITECTURE STANDARDS

Standard ID:	<b>AR-SYSENV1-001</b>
Title:	<b>Systems Environment</b>
Domain:	<b>Architecture</b>
Discipline:	<b>System Landscape Architecture</b>
Updated date:	<b>6/27/2016</b>
Revision no.:	<b>5</b>
Original date:	<b>6/01/2005</b>

## I. Authority, Applicability and Purpose

- A. **Authority:** Title 29, Chapter 90C provides broad statutory authority to the Department of Technology and Information to implement statewide and interagency technology solutions, policy, standards, and guidelines for the State of Delaware's technology infrastructure. "Technology" means computing and telecommunications systems, their supporting infrastructure and interconnectivity used to acquire, transport, process, analyze, store, and disseminate information or data electronically. The term "technology" includes systems and equipment associated with e-government and internet initiatives.
- B. **Applicability:** Applies to all State of Delaware communications and computing resources. DTI is an Executive Branch Agency and has no authority over the customers in Legislative and Judicial Branches, as well as School Districts, and other Federal and Local Government entities that use these resources. However, all users, including these entities, must agree to abide by all policies, standards promulgated by DTI as a condition of funding and continued use of these resources.
- C. **Purpose:** This standard addresses the need to set up common definitions and strategies for System Environments. One of our goals in setting up a standard strategy for these environments is to promote an infrastructure that will support the move to e-Government and the tight integration of services to citizens.

## II. Scope

- A. **Audience:** Application Developers, their managers and Application Development Contractors for the State, Systems Administrators, Network Administrators, and Computer Auditors are the intended audience. IT personnel are the only intended users of this document.

These standards are adopted by the Department of Technology and Information (DTI), through the Technology and Architecture Standards Committee (TASC), and are applicable to all Information Technology use throughout the State of Delaware. Any questions or comments should be directed to [dti\\_tasc@state.de.us](mailto:dti_tasc@state.de.us).



## DELAWARE STATE-WIDE INFORMATION TECHNOLOGY AND ARCHITECTURE STANDARDS

- B. **Applicability:** This standard will cover all System Environments in the State. The strategy includes source code maintenance, libraries, testing, and production environments. This includes all software development and maintenance environments in use by the State of Delaware, and all installations owned by the State but housed by third-party contractors. This standard does not apply to modeling systems, metadata systems, business process analyzers, or engineering or traffic computer systems.
- C. **Environments:** This standard covers all System Environments, including Client/Server, mobile, web and Mainframe.

### III. Process

- A. **Adoption:** The Department of Technology and Information (DTI) adopted these standards through the Technology and Architecture Standards Committee (TASC). They are applicable to all Information Technology throughout the State of Delaware.
- B. **Revision:** Technology is constantly changing. It is the intent of the TASC to review each standard yearly. Your Information Resource Manager (IRM) will channel your suggestions and comments to the TASC.
- C. **Contractors:** DTI requires all contractors or other third parties to comply with these standards when proposing technology solutions to or other State entities. Failure to do so could result in rejection by the Delaware Technology Investment Council. For further guidance, or to seek review of a non-rated component, contact TASC at [dti\\_tasc@state.de.us](mailto:dti_tasc@state.de.us).
- D. **Implementation responsibility:** DTI and/or the organization's technical staff will implement this standard during the course of normal business activities, including business case review, architectural review, project execution and the design, development, or support of systems.
- E. **Enforcement:** DTI will enforce this standard during the course of normal business activities, including business case and architectural review of proposed projects and during the design, development, or support of systems. This standard may also be enforced by others during the course of their normal business activities, including audits and design reviews.
- F. **Contact us:** Please direct any questions or comments to [dti\\_tasc@state.de.us](mailto:dti_tasc@state.de.us).

### IV. Definitions/Declarations

#### A. Definitions

1. **Change control** – It is the process to ensure that hardware and software changes are reviewed, approved, controlled, tracked, and scheduled prior going into the production environment.
2. **Data Modeler** - defines and analyzes data and the requirements needed to support the business process.
3. **DBA** - a person responsible for the design, implementation, maintenance and repair of an organization's database.
4. **System Control Manager** – follows a systematic approach to managing all changes made to a product or system.

These standards are adopted by the Department of Technology and Information (DTI), through the Technology and Architecture Standards Committee (TASC), and are applicable to all Information Technology use throughout the State of Delaware. Any questions or comments should be directed to [dti\\_tasc@state.de.us](mailto:dti_tasc@state.de.us).



## DELAWARE STATE-WIDE INFORMATION TECHNOLOGY AND ARCHITECTURE STANDARDS

---

5. **End-users** – Execute the system as designed. Perform post-implementation user acceptance testing.
6. **Integration testing** –The execution of a test plan against a group of software components to ensure they each function together as expected.
7. **Role** – the actions and activities assigned to or required or expected of a person or group. An individual can perform multiple roles depending on agency practices.
8. **Software Engineer** - An individual who designs, implements and maintains software and software based infrastructure to meet the business needs of the customer.
9. **Software Infrastructure** – A system comprised of a number of different software components.
10. **System Administrator** - responsible for system patching, system account management, and have the ability to promote code for user acceptance testing and production.
11. **System environment** – A logical collection of hardware, software and processes to accomplish a specific purpose.
  - a) **Development** – The development of code is the primary purpose of this environment.
  - b) **Test** – Application testing by end-user or system/integration testing by engineers are the primary purposes of this environment.
  - c) **Production** - Reliable and secure computing is the primary purpose of this environment.
12. **System support** – Analyzes audit logs, troubleshoots and fixes, performs capacity/performance planning.
13. **System testing** –The software or hardware testing on a complete, integrated system to evaluate the system's compliance with its specified requirements.
14. **Testers** – Executes system integration test plans and user acceptance test plans.
15. **Unit testing** – A software development process in which the smallest testable parts of an application, called units, are individually and independently scrutinized for proper operation automatically and programmatically.
16. **Version control** – A system that records changes to a file or set of files over time so that you can recall specific versions later.



## B. Declarations

### A Systems Environment must:

1. Adhere to existing State standard and policies. For example, the data in any environment must be appropriately secured/managed and a Disaster Recovery criticality classification must be determined for each production environment. Also, remote vendor support must comply with the requirements found in existing State standards and policies such as the System Architecture standard.
2. Enable the separation of duties, which will lead to higher accountability and security.
3. Enable the separation of functions, which will lead to higher availability and reliability.

### Managing systems within environments must:

1. Provide version control and provide backup and restore capabilities.
2. Utilize document management for all system related documents including, but not limited to requirements, design, change control, testing procedures and sign-offs.
3. Include unit testing.

Vendor developed/maintained applications hosted by the State often present unique challenges as the developers may be off site along with their testing environments. Subtle differences in the vendor vs. State environments may result in unexpected issues after promotion to the State user environments. The final vendor test environment should be duplicated at the State so that vendor code is moved to a State staging environment, where it is retested prior to actual promotion to a State user environment.

Typically disaster recovery is only a consideration for the production environment as production is the most critical environment for business continuity requirements. However, consider including development and test environments in the disaster recovery plan. In a development project, the critical path is dependent on all essential environments being available so that project developers and testers can continue their work according to the project schedule. Even in the maintenance and operation phase, the lack of development and test environments could result in delays in resolving critical production issues.



## DELAWARE STATE-WIDE INFORMATION TECHNOLOGY AND ARCHITECTURE STANDARDS

### V. Definition of Ratings

COMPONENT RATING	USAGE NOTES
<b>STANDARD</b> – DTI offers internal support and/or has arranged for external vendor support as well (where applicable). DTI believes the component is robust solidly positioned in its product life cycle.	These components can be used without explicit DTI approval for both <b><u>new projects</u></b> and <b><u>enhancement</u></b> of existing systems.
<b>DECLINING</b> – Deprecated - DTI considers the component to be a likely candidate to have support discontinued in the near future. A deprecated element is one becoming invalid or obsolete.	Via the State's waiver process, these components must be explicitly approved by DTI for <b><u>all projects</u></b> . They must not be used for <b><u>minor enhancement</u></b> and <b><u>system maintenance</u></b> without explicit DTI approval via the State's waiver process.
<b>DISALLOWED</b> – DTI declares the component to be unacceptable for use and will actively intervene to disallow its use when discovered.	No waiver requests for new solutions with this component rating will be considered.

- A. Missing Components** – No conclusions should be inferred if a specific component is not listed. Instead, contact TASC to obtain further information.



## VI. Component Assessments

### Applicable tools for managing code within environments

#	Component	Rating	Comments
	<b>Version Control Systems</b>		<b>Code versioning and promotion</b>
1	N2O	Standard	Natural
2	Git	Standard	
3	GitLab	Standard	
4	Serena Changeman ZMF	Standard	COBOL, Assembler, JCL, Rexx
5	Visual Source Safe	Disallowed	
6	Microsoft Team Foundation Suite	Declining	
7	STAT	Standard	For PeopleSoft ERP
8	CVS	Disallowed	
9	Team Studio	Disallowed	Lotus Notes
10	Dimensions	Disallowed	
	<b>Test Managers/Tools</b>		
1	Quality Center	Standard	
2	Quick Test Pro	Standard	
3	Microsoft Test Manager	Standard	



### **Development Environment**

This environment is optimized for the software engineer, while respecting other roles within the organization.

#### **Roles**

- Software Engineer
- DBA
- Data Modeler
- System administrator
- System Control Manager

### **Test Environment**

This environment is optimized for the testers and provides a structure for other roles to facilitate the required testing. Where necessary, multiple Test instances should be created (QA, Certification, Demo, Systems Integration, User Acceptance Testing, Patch Management, etc.). For governance, the purpose for the Test instance should be documented. This documentation can be relied upon to reduce inadvertent deletion of a needed instance and help decide when it is appropriate to delete an instance. Test environments must be built in compliance with the Systems Architecture Standard.

#### **Roles**

- Testers
- Software Engineer – Limited access for testing purposes only, cannot modify or push code
- DBA
- System administrator
- System Control Manager
- System Support

### **Training Environment**

This environment is optimized to perform training processes for the State. End-users expect a secure, stable and reliable system to meet the business' needs.

#### **Roles**

- End-user
- System support
- DBA
- System administrator
- System Control Manager



## DELAWARE STATE-WIDE INFORMATION TECHNOLOGY AND ARCHITECTURE STANDARDS

---

### **Production Environment**

This environment is optimized to perform business processes for the State. End-users expect a secure, stable and reliable system as necessary to meet the BC/DR business criticality and data classification as determined by the data steward.

#### **Roles**

- End-user
- System support
- DBA
- System administrator
- System Control Manager

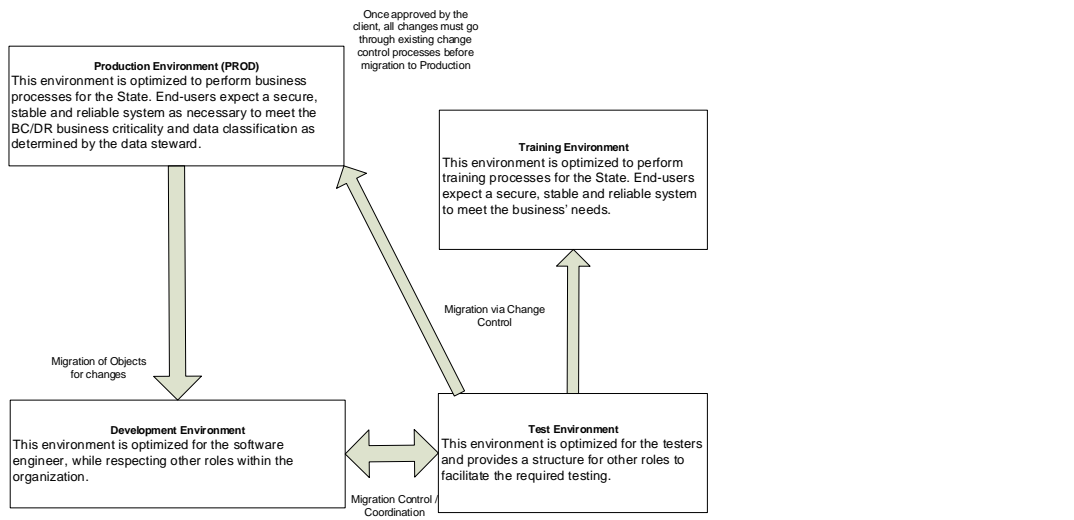
**System environments may share infrastructure as long as the Delaware Information Security Policy (DISP) requirements are adhered to.**





Below is a diagram of the typical workflow between the development, test and production environments. The Training environment can be created as needed.

**Environment Work Flow**



These standards are adopted by the Department of Technology and Information (DTI), through the Technology and Architecture Standards Committee (TASC), and are applicable to all Information Technology use throughout the State of Delaware. Any questions or comments should be directed to [dti\\_tasc@state.de.us](mailto:dti_tasc@state.de.us).