



**Government of Ontario IT Standard (GO-ITS)**

Number 20.1

**Platform Software Standard**

Version #: **2.1**

Status: **Approved**

Prepared for the Information Technology Standards Council (ITSC) under the delegated authority  
of the Management Board of Cabinet

## Foreword

Government of Ontario Information Technology Standards (GO-ITS) are the official publications on the guidelines, preferred practices, standards and technical reports adopted by the Information Technology Standards Council (ITSC) under delegated authority of the Management Board of Cabinet (MBC). These publications support the responsibilities of the Management Board Secretariat (MBS) for coordinating standardization of Information & Information Technology (I&IT) in the Government of Ontario. Publications that set new or revised standards provide enterprise architecture guidance, policy guidance and administrative information for their implementation. In particular, GO-ITS describe where the application of a standard is mandatory and specify any qualifications governing the implementation of standards.

# Table of Contents

- 1. INTRODUCTION ..... 4**
  - 1.1 BACKGROUND AND PURPOSE ..... 4
  - 1.2 SCOPE ..... 5
    - 1.2.1 *In Scope* ..... 5
    - 1.2.2 *Out of Scope* ..... 5
  - 1.3 APPLICABILITY STATEMENTS ..... 5
    - 1.3.1 *Organization* ..... 5
  - 1.4 REQUIREMENTS LEVELS ..... 6
  - 1.5 PUBLICATION DETAILS ..... 6
- 2. MANDATORY REQUIREMENTS ..... 6**
  - 2.1 KEY CATEGORIES ..... 7
    - 2.1.1 *Infrastructure Component Catalogue Generic Categories* ..... 7
- 3. RELATED STANDARDS ..... 10**
  - 3.1 IMPACTS TO EXISTING STANDARDS ..... 10
  - 3.2 IMPACTS TO EXISTING ENVIRONMENT ..... 10
- 4. RECOMMENDED VERSIONING AND/OR CHANGE MANAGEMENT ..... 12**
- 5. ACKNOWLEDGEMENTS ..... 12**
  - 5.1 EDITORS ..... 12
  - 5.2 CONTRIBUTORS ..... 12
  - 5.3 CONSULTATIONS ..... 13
- 6. DOCUMENT HISTORY ..... 13**
- 7. COPYRIGHT INFORMATION ..... 14**
- GLOSSARY ..... 15**
- APPENDIX A ..... 18**
  - PRODUCT LIST ..... 18

# 1. Introduction

## 1.1 Background and Purpose

The Major Applications Portfolio Strategy (MAPS) provides an approach and a comprehensive action plan for supporting the government's major business applications and their underlying technology. The Major Applications Portfolio Strategy provides a roadmap for developing an improved decision-making process that will allow for sustainable I&IT investments based on value and risk parameters. One of the six MAPS recommendations is to standardize the I&IT development, test and deployment environments to reduce technology complexity and cost.

This standard lists I&IT software products being deployed within the standardized I&IT development, test and deployment environments. These software products will be hosted on a common shared software environment that consists of a sub-set of the existing software products currently deployed across the Government of Ontario. This shared software environment will be part of the Infrastructure Technology Services (ITS) services offerings.

**I&IT product standard approvals from ITSC & ARB do not replace the procurement process. Acquiring the products listed in this platform standard will require the appropriate procurement process.**

Some key drivers for standardized I&IT development, test and deployment environments within the OPS are as follows:

- simplified hardware and software environment and software selection;
- reduced software licensing and support costs;
- increased agility in taking advantage of new and emerging technologies which could provide benefits to the OPS;
- improved business and IT alignment;
- supporting eCollaboration

Consultation with each Cluster revealed that two standardized core operating technologies and tools for all major application, database and middle tier tools were widely deployed and supported by existing IT people, processes and technologies; the Intel X86 environment in which Microsoft .NET was used and a Sun SPARC (Unix) environment in which Java (J2EE) was deployed. In addition, deployment of technologies under Government of Ontario's Open Source Software (OSS) policy is also subject to these drivers and must conform to this standard.

The purpose of this document is to define the software products in each environment (.NET, Java)

Because these two platforms have been standardized in the OPS, the operating systems, tools, languages etc. comprising the ITS-supported platforms generally involve:

- Products such as C++, Java and C# aided by application server technologies such as Websphere, etc. that can be used to build client/server applications;
- Servers and server-side scripting languages including JSP, ASP, PHP, Perl, Python, etc. that run in web servers and servlet containers that can be used to build browser applications; and
- Robust common components allowing clusters to integrate and deploy mission critical solutions.

Most clusters and OPS application developers will use a combination of these architectures and tools, and in some cases, several types are used within the context of a single solution. Most clusters will also have a core platform upon which the bulk of their solutions are based – some leaning towards Java, and others towards .NET. There is no single answer to the question “Which platform is best?”

The best answer in selecting the right **primary platform** for a particular solution lies in establishing the most important **application delivery requirements**. The word ‘primary’ is important because neither platform should ever restrict future choices.

Similarly, it is essential that new technologies are able to be introduced into the stack allowing solutions to adapt as business needs change and as the market’s technology landscape evolves.

## 1.2 Scope

### 1.2.1 In Scope

- All net new development and deployment of one, two, three and N-tier systems for use within the OPS.
- Systems that have been identified as candidates for modernization but have not yet begun development or deployment activities.

### 1.2.2 Out of Scope

- Existing legacy one, two, three and N-Tier systems whether mainframe or distributed that have not begun a modernization review or are not candidates for modernization within the current lifespan of their vendor support agreements.
- Systems that have received approval for development or deployment via the recognized exemption process currently in force. This includes boutique deployments of AIX based applications.
- COTS business solutions which do not replace or duplicate a product listed in Appendix “A”. However, these products must be either Java or .NET framework compliant.

## 1.3 Applicability Statements

### 1.3.1 Organization

Government of Ontario IT Standards and Enterprise Solutions and Services apply (are mandatory) for use by all ministries/clusters and to all former Schedule I and IV provincial government agencies under their present classification (Advisory, Regulatory, Adjudicative, Operational Service, Operational Enterprise, Trust or Crown Foundation) according to the current agency classification system.

Additionally, this applies to any other new or existing agencies designated by Management Board of Cabinet as being subject to such publications, i.e. the GO-ITS publications and enterprise solutions and services - and particularly applies to Advisory, Regulatory, and Adjudicative Agencies (see also procurement link, OPS paragraph). Further included is any agency which, under the terms of its Memorandum of Understanding with its responsible Minister, is required to satisfy the mandatory requirements set out in any of the Management Board of Cabinet Directives (*cf.* Operational Service, Operational Enterprise, Trust, or Crown Foundation Agencies).

As new GO-IT standards are approved, they are deemed mandatory on a go-forward basis (Go-forward basis means at the next available project development or procurement opportunity).

When implementing or adopting any Government of Ontario IT standards or IT standards updates, ministries and I&IT Cluster must follow their organization's pre-approved policies and practices for ensuring that adequate change control, change management and risk mitigation mechanisms are in place and employed.

For the purposes of this document, any reference to ministries or the Government includes applicable agencies.

## 1.4 Requirements Levels

Within this document, certain wording conventions are followed. There are precise requirements and obligations associated with the following terms:

<b>Must</b>	This word, or the terms "REQUIRED" or "SHALL", means that the statement is an absolute requirement.
<b>Should</b>	This word, or the adjective "RECOMMENDED", means that there may exist valid reasons in particular circumstances to ignore the recommendation, but the full implications (e.g., business functionality, security, cost) must be understood and carefully weighed before

## 1.5 Publication Details

All approved Government of Ontario IT Standards (GO-ITS) are published on the ITSC Intranet web site. Please indicate with a checkmark below if this standard is also to be published on the public, GO-ITS Internet Site.

Standard to be published on both the OPS Intranet and the GO-ITS Internet web site (available to the public, vendors etc.)	<input checked="" type="checkbox"/>
--	-------------------------------------

## 2. Mandatory Requirements

This standard establishes .NET and Java as the two supported development environments and identifies the software that must be used in Appendix "A". Clusters/Ministries are required to utilize new services from ITS as they become available. These include a set of standard services for application development, testing, quality assurance and production within the Utility Infrastructure Management Service (UIMS), in support of the infrastructure consolidation mandate.

The following section provides descriptions of the services offered within the Utility Infrastructure Management Service according to the Infrastructure Component Catalogue (ICC), by which the complete list of application development, testing, quality assurance and production software in Appendix A is categorized.

Software in Appendix A has been chosen to provide specific functionality by category. Developers may chose software from either Java or .NET environments according to their functional requirements.

## 2.1 Key Categories

The structure of the environment has been determined by the Technology Reference Model (TRM) and more specifically by the Infrastructure Component Catalogue (ICC) which is an integral part of the TRM. The ICC is comprised of categories of services. The following chart outlines the categories within the ICC. It is software agnostic and applies to all software technology environments under consideration for use within the OPS.

### 2.1.1 Infrastructure Component Catalogue Generic Categories

Category	Category Details
<p><b>Application Program Interface – Intra-API, Inter-API and Infra-API services.</b></p> <p>Technology that models new APIs or techniques that assure best API use, e.g. Universal Modeling Language (UML) modeling tools</p>	<p><u>Modeling Tools:</u></p> <ul style="list-style-type: none"> <li>• UML Modeling</li> <li>• Requirement Management</li> <li>• Code Modeling</li> </ul>
<p><b>Presentation – Web Server, IVR and WAP Server</b></p> <p>Technology that provides points of interaction (POI) or different presentations of data</p>	<p><u>Web server:</u></p> <ul style="list-style-type: none"> <li>• Supplied by Operating System (OS)</li> </ul> <p><u>Workflow Tools:</u></p> <ul style="list-style-type: none"> <li>• Human Workflow</li> <li>• Business Process</li> <li>• Business Object Components</li> <li>• Unified Communication</li> <li>• Business Intelligence (BI)</li> <li>• Dashboard/Analytics</li> <li>• Peer to Peer (P2P)</li> <li>• Web Content Management (WCM)</li> <li>• Search Engine</li> </ul> <p><u>Web Presentation/Authoring:</u></p> <ul style="list-style-type: none"> <li>• Web Presentation Components</li> <li>• Business Transaction Components</li> <li>• Data Access APIs</li> <li>• Message Queue Components</li> </ul> <p><u>Portal:</u></p> <ul style="list-style-type: none"> <li>• Portal Services</li> </ul>
<p><b>Application – Application Server</b></p> <p>Application server software that executes business logic</p>	<p><u>Application Server &amp; Components:</u></p> <ul style="list-style-type: none"> <li>• Application Server</li> <li>• Security Services</li> <li>• Proxy Server</li> <li>• Rights Management &amp; Admin (Provisioning)</li> </ul>
<p><b>Integration – Integration Server, Inter-Enterprise Integration (IEI) Server</b></p> <p>Enterprise application integration (EAI)</p>	<p><u>Programming Interface:</u></p> <ul style="list-style-type: none"> <li>• Transactions</li> <li>• Message Queuing</li> </ul>

Category	Category Details
<p>software that connects different applications together, reformatting and routing data as necessary.</p>	<ul style="list-style-type: none"> <li>• Directory</li> <li>• Integration Connectors</li> <li>• Email APIs</li> </ul>
<p><b>Database – DBMS, Data Access Middleware</b></p> <p>Software that stores data for efficient record and field level retrieval along with data access and gateway functions</p>	<p><u>Database (DBMS)/Development:</u></p> <ul style="list-style-type: none"> <li>• Database</li> </ul> <p><u>Messaging/Email:</u></p> <ul style="list-style-type: none"> <li>• Microsoft Exchange</li> <li>• Mailbox Management</li> <li>• Message Hygiene (Antivirus / Anti-spam)</li> <li>• Mobile Messaging</li> <li>• Instant Messaging</li> <li>• Storage / Restore</li> </ul>
<p><b>Server – Application Server HW and OS, Web Server HW and OS</b></p> <p>Server hardware and operating system</p>	<p><u>Operating System:</u></p> <ul style="list-style-type: none"> <li>• Environment specific OS</li> </ul>
<p><b>Directory – Directory Server</b> (Across-System (Universal) Service Provision)</p>	<p><u>Directory Services:</u></p> <ul style="list-style-type: none"> <li>• Native OS services</li> </ul> <p><u>Web Service Directory (UDDI):</u></p> <ul style="list-style-type: none"> <li>• Native OS services</li> </ul>
<p><b>Management – Service Level Management (SLM) monitoring products</b></p> <p>To ensure that the required application service levels are being suitably supported by the infrastructure platform, an end-to-end enterprise level view of all contributing components must be available. Furthermore, the configuration of this management system and its contributing components must be mapped to enterprise-level policy definitions to ensure consistency.</p>	<p><u>Monitoring Tools:</u></p> <p>Monitoring Tool Supported by OS</p> <p><u>Management Tools:</u></p> <p>Configuration Management Tool Supported by OS</p> <p><u>Project &amp; Portfolio Management:</u></p> <p>Project &amp; Portfolio Management Tool Supported by OS</p>
<p><b>Universal (including Languages, Debugging and Reporting Tools)</b></p>	<p><u>Development/Language:</u></p> <ul style="list-style-type: none"> <li>• Language (such as C++, C#)</li> <li>• OS Platform &amp; Runtime</li> <li>• Web, Desktop and Mobile GUI</li> <li>• Server-side Component</li> <li>• Persistent Objects</li> </ul>

Category	Category Details
	<ul style="list-style-type: none"> <li>• Web Page Generation</li> <li>• In-Line Code/Code Behind</li> <li>• Relational Data Access</li> <li>• Queuing</li> <li>• Asynchronous Invocation</li> <li>• Naming</li> <li>• HTTP Engine</li> <li>• XML</li> </ul>
<b>Testing, Code Management &amp; Quality Assurance</b>	<ul style="list-style-type: none"> <li>• Code and Quality Assurance (QA) Testing</li> <li>• Scheduling</li> <li>• Analysis Services</li> <li>• Source Code Library</li> <li>• Code Development</li> <li>• Code Reverse Engineering</li> <li>• Database Debug</li> </ul>
<b>Reporting</b>	<ul style="list-style-type: none"> <li>• Analytics Reporting Tool</li> <li>• SQL Reporting Tool</li> </ul>
<b>Collaboration</b>	<ul style="list-style-type: none"> <li>• Collaboration Service</li> </ul>

### 3. Related Standards

#### 3.1 Impacts to Existing Standards

GO-ITS Number	Describe Impact	Recommended Action (alternatively provide a page number where details can be found)
GO-ITS 24 Omnibus Standard	Alignment required for protocol versions within GO-ITS 24	Submit revisions to ITSC
GO-ITS 30 Database Management Systems	GO-ITS 20.1 effectively signals that three database products are approved for use in the OPS rather than two	Recommend to ITSC that GO-ITS 30 be revised or retired

#### 3.2 Impacts to Existing Environment

Impacted Infrastructure (includes Common Components and other applications)	Describe Impact	Recommended Action (alternatively provide a page number where details can be found)
<p>Net new applications must be developed in one of the software environments outlined under Section 1.1 “<i>Background and Purpose</i>” of this document and must use the software listed in Appendix “A”.</p> <p>Legacy applications are not affected by this standard until they are at the end of their service life and must be either replaced or modernized. The replacement and/or modernization activities are subject to this standard at that time.</p> <p>Applications approved under the exception process are not subject to this standard until they are at the end of their</p>	<ul style="list-style-type: none"> <li>• No impact to net new applications planned or under development.</li> <li>• Legacy application renewal may be impacted in the case of applications running on mainframes. Impacts may include increased lead times and additional resources to create a modern front end to interface with the existing backend application through the creation of an N-Tier application.</li> </ul>	<ul style="list-style-type: none"> <li>• Any application that can interface with an existing legacy application must be developed in accordance with this standard.</li> </ul>

<b>Impacted Infrastructure (includes Common Components and other applications)</b>	<b>Describe Impact</b>	<b>Recommended Action (alternatively provide a page number where details can be found)</b>
service life and must be modernized or replaced. The replacement and/or modernization activities are subject to this standard at that time unless a new exception process has been approved for the application to be modernized.		
Common Components	Cluster developers and CCAS solution teams must identify reusable Web Services when interfacing with or establishing new components to avoid the risk of being locked into J2EE or .NET-specific products	GO-ITS 54 <i>Application Development Standard</i> , see statements mandating that interfaces must be documented
Application software procured under existing Vendor of Record Agreements	All existing legal agreements for procurement of software will be honoured for the life of the agreements	No action required for applications and infrastructure operating under existing VORs. Net new installations must use ITS infrastructure and services where available.

## 4. Recommended Versioning and/or Change Management

Modifications during the life of the standard must be approved by the organizational owners of the document. The next review date will be in twelve calendar months from the time of ARB approval for this standard.

The organizational owners of this standard are:

Rocco Passero  
 Director,  
 Enterprise Strategic Planning, Design & Services Branch  
 Infrastructure Technology Services (ITS), MGS  
 rocco.passero@ontario.ca  
 416-326-3398

Eion Gomes  
 Head, Technology Adoption Branch  
 Office of the Corporate Chief Technology Officer,  
 MGS  
 eion.gomes@ontario.ca  
 416-327-2345

The Ministry of Government Services will follow the Gating Process (approval process) as described in the Government of Ontario I&IT Directive, and submit proposed revisions to the Information Technology Standards Council (ITSC) and the Architecture Review Board (ARB) for approval and publication.

## 5. Acknowledgements

Document editors, contributors and reviewers are shown below. This includes individuals and/or groups of individuals that provided subject expertise and contacts designated to receive comments, feedback, questions or suggestions.

### 5.1 Editors

Full Name	Cluster, Ministry and/or Area
Rizwan Akhtar, OCCTO Editor	OCCTO, Ministry of Government Services
Paul Daly, OCCTO Editor	OCCTO, Ministry of Government Services

### 5.2 Contributors

Full Name	Cluster, Ministry and/or Area
Christopher Chua	ITS, Ministry of Government Services
Andreas Ott	OCCTO, TAB, Ministry of Government Services
Peter Churchard	OCCTO, CAB, Ministry of Government Services
Brian Bisailon	OCCTO, TAB, Ministry of Government Services
Stavros Platis	.NET Centre of Excellence/Economics & Transportation Cluster

## 5.3 Consultations

The following were consulted:

Renee Laforet (HSC)	David Croft (GSDC)	Bud MacDonald (HSC)
Ray Nakano (GSDC)	Howard Bertrand (ETC)	Sandy Mannering (JUS)
Merlino Franco (LRC)	Karl Cunningham (OCCS)	Kim Langford (ETC)
Barry McKee (CYSS)	Lakshmi Subash (OCCTO)	Rose Reeve (CAC)
Cathy Hamilton (HSC)	Brian Speed (ITS)	
Joanne Hiscock (LRC)	Christopher Chua (ITS)	
Infrastructure Management Committee	Architecture Core Team	Solution Management Forum

## 6. Document History

**Created:** March-April 2007

- Version 1.0
  - Approved by ARB, April 3, 2007. Pending ITSC endorsement
  - Endorsement by ITSC, May 9, 2007

**Updated:** 2008-09-29

- Proposed Revision created. Updated: Background (pg.4) and Contributors (pg.12). Draft number set to 1.50

**Updated:** 2008-10-03

- Appendix A: MS Dynamics *replaces* MS Performance Point Server; MS Office SharePoint Server 2007 (MOSS 2007) *replaces* MS Groove; Draft version number set to 1.52

**Updated:** 2008-10-09

- Tracked changes accepted throughout document; minor typos corrected.
- Appendix A: Dashboard/Analytics in .NET Environment updated to MOSS 2007.

**Updated:** 2008-10-16

- The statement "*Software listed for one category must not be used to provide functionality in another category*" is removed from Mandatory Requirement, section 2.
- Appendix A: Renamed the column headings; Removed wording "*ITS Responsible*"; Removed asterisks from Proxy Server since this service is not provided by CCAS; Draft version number set to 1.54

**Updated:** 2008-10-21

- Stellent and Cognos added to the .NET Recommended column; Triple asterisk added to Windows Server 2003 (pp. 20, 22) - ITS will move to Windows Server 2008 as the new server standard during 2009.
- Draft version number set to 1.55

**Updated:** 2008-11-13

- Updated Background (pg.4) with reference to *Major Applications Portfolio Strategy (MAPS)*. Draft set to 1.56

**Endorsed:** 2008-11-19

- Portal Services row for .NET Revised Platform changed to "MOSS 2007/IBM Websphere Portal"

- Endorsed by ITSC to proceed to ARB
- Draft version number set to 1.57

**Updated:** 2009-01-13

- Page 4: Removed footnote reference to *Beyond e-Ontario*
- Final Draft version number set to 1.59

**Approved:** 2009-01-15

- Page 6: Removed redundant statement concerning the GO-ITS exemption process
- Architecture Review Board approval
- Approved version number set to 2.0

**Proposed Revision:** 2009-06-17

- Operating System for .NET Platform updated to Windows Server 2008
- Proposed revision number set to Draft Version 2.0c

**Endorsed:** 2009-07-15

- IT Standards Council endorsement

**Approved:** 2009-07-16

- 
- Architecture Review Board approval
- Approved version number set to 2.1

## 7. Copyright Information

© Queen's Printer for Ontario 2009.

## Glossary

<p><b>Application Runtime Environment/Library:</b></p> <p>A runtime library is a collection of utility functions which support a program while it is running, working with the operating system to provide facilities such as mathematical functions, input and output.</p>
<p><b>Application Server:</b></p> <p>An application server is a software engine that delivers applications to client computers or devices. Moreover, an application server handles most, if not all, of the business logic and data. A computer that is responsible for accepting HTTP requests from clients, which are known as Web browsers, and serving them HTTP responses along with optional data contents, which usually are Web pages such as HTML documents and linked objects (images, etc.)</p>
<p><b>Asynchronous Invocation:</b></p> <p>Asynchronous Invocation in computer and software engineering is the process by which a computer or virtual computer carries out the instructions of a computer program, and program can execute, without receiving a confirmation from other depended process</p>
<p><b>Business Transaction Components:</b></p> <p>Interface programs, or set of codes to connection various programs.</p>
<p><b>Data Access APIs:</b></p> <p>A collection/set of routine, protocols, and tools for building software applications, to access database systems.</p>
<p><b>Database:</b></p> <p>Database is a structured collection of records or data that is stored in a computer so that a program can consult it to answer queries. The records retrieved in answer to queries become information that can be used to make decisions</p>
<p><b>Directory:</b></p> <p>A directory or directory service is a software application — or a set of applications — that stores and organizes information about a computer network's users and network resources, and that allows network administrators to manage users' access to the resources. Additionally, directory services act as an abstraction layer between users and shared resources.</p>
<p><b>Email:</b></p> <p>An email is a store and forward method of composing, sending, storing, and receiving messages over electronic communication systems</p>
<p><b>GUI:</b></p> <p>A graphical user interface (GUI) is a type of user interface which allows people to interact with a computer or other media formats which employs graphical icons, visual indicators or special graphical elements called "widgets", along with text labels or text navigation to represent the information and actions available to a user. The actions are usually performed through direct manipulation of the graphical elements,</p>
<p><b>In-line Codes:</b></p> <p>A line code (also called behind code) is a set of programming language, or codes written connecting two different areas within a program, or can be used to command transmissions within a program.</p>

<p><b>Integration Connectors:</b></p> <p>A Set of programming interfaces to connect various programs, and systems.</p>
<p><b>Language (such as C++, C#):</b></p> <p>A programming language is a language used to write computer programs, which instruct a computer to perform some kind of computation, and possibly control external devices such as printers, or mainframe systems, etc.</p>
<p><b>Message Queue Components:</b></p> <p>Queuing refers to lining up jobs for a computer or device to exchange computer commands, or data objects.</p>
<p><b>Message Queuing:</b></p> <p>Message Queuing is a process to use a messaging protocol, that allows applications running on disparate servers to communicate in a failsafe manner. A queue is a temporary storage location from which messages can be sent when conditions permit. This enables communication across heterogeneous networks and between computers which may not always be connected. By contrast sockets and other network protocols assume that direct connections always exist.</p>
<p><b>Naming:</b></p> <p>Naming is a process of organizing search of objects on the Internet. An API, such as JNDI can be used to access a directory service that allows clients to discover and lookup data and objects via a name.</p>
<p><b>Persistent Objects:</b></p> <p>Persistent objects are individual units of run-time data storage that are used as the basic building block of programs. These objects act on each other, as opposed to a traditional view in which a program may be seen as a collection of functions, or simply as a list of instructions to the computer. Persistent objects are used, when previous versions of software &amp; programs are request to be available, in fail-safe mode. Persistent objects preserves the previous version of itself when it is modified; such data structures are effectively immutable, as their operations do not (visibly) update the structure in-place, but instead always yield a new updated structure.</p>
<p><b>Queuing:</b></p> <p>Queuing is a buffer abstract data structure providing services in computer science, transport and operations research where various entities such as data, objects, persons, or events are stored and held to be processed later. The most well known operation of the queue is the First-In-First-Out (FIFO) queue process. In a FIFO queue, the first element in the queue will be the first one out; this is equivalent to the requirement that whenever an element is added, all elements that were added before have to be removed before the new element can be invoked. Unless otherwise specified, the remainder of the article will refer to FIFO queues. There are also non-FIFO queue data structures, like priority queues.</p> <p>There are two basic operations associated with a queue: enqueue and dequeue. Enqueue means adding a new item to the rear of the queue while dequeue refers to removing the front item from the queue and returning it to the calling entity.</p>
<p><b>Relational Data Access:</b></p> <p>Relational Data access typically refers to software and activities related to storing, retrieving, or acting on data housed in a database or other repository.</p> <p>Historically, different methods and languages were required for every repository, including each different database, file system, etc., and many of these repositories stored their content in different and incompatible formats.</p> <p>In more recent days, standardized languages, methods, and formats, have been created to serve as</p>

<p>interfaces between the often proprietary, and always idiosyncratic, specific languages and methods. Such standards include SQL, ODBC, JDBC, ADO.NET, XML, XQuery, XPath, and Web Services.</p>
<p><b>Server-side Component:</b></p> <p>Collection of code to connect different artefacts, server-side components are also known as a pattern language. Server-side components are software programs, such as a web server API, which runs on a remote server, reachable from a user's local computer or workstation. Operations may be performed server-side because they require access to information or functionality that is not available on the client, or require typical behaviour that is unreliable when it is done client-side.</p>
<p><b>Transactions:</b></p> <p>A transaction is a unit of interactions with a database management system or similar system that is treated in a coherent and reliable way independent of other transactions that must be either entirely completed or aborted.</p>
<p><b>Web Presentation Components:</b></p> <p>The interface program, or set of codes to generate meaningful information on a computer monitor.</p>
<p><b>Web Server:</b></p> <p>A computer that is responsible for accepting HTTP requests from clients, which are known as Web browsers, and serving them HTTP responses along with optional data contents, which usually are Web pages such as HTML documents and linked objects (images, etc.).</p>
<p><b>Webpage Generation:</b></p> <p>It is a process to create a web document, which also known as Web page or webpage. It is a resource of information that is suitable for the World Wide Web and can be accessed through a web browser.</p>
<p><b>Webpage:</b></p> <p>A web page or webpage is a resource of information that is suitable for the World Wide Web and can be accessed through a web browser. This information is usually in HTML or XHTML format, and may provide navigation to other web pages via hypertext links. Web pages may be retrieved from a local computer or from a remote web server. The web server may restrict access only to a private network, e.g. a corporate intranet, or it may publish pages on the World Wide Web. Web pages are requested and served from web servers using Hypertext Transfer Protocol (HTTP).</p>
<p><b>XML:</b></p> <p>The Extensible Markup Language (XML) is a general-purpose markup language. Its primary purpose is to facilitate the sharing of data across different information systems, particularly via the Internet. It is a simplified subset of the Standard Generalized Markup Language (SGML), and is designed to be relatively human-legible. By adding semantic constraints, application languages can be implemented in XML.</p>

## Appendix A

### **Product List**

(See following pages)

GO-ITS 20.1 Platform Software: Appendix A	Product Category	.NET Standard Platform	.NET Revised Platform	Java Standard Platform
<b>2.1.1 INTERFACE AND SYSTEM-LEVEL SERVICE PROVISION</b>				
<b>API – Intra-API, Inter-API and Infra-API services. Technology that models new APIs or techniques that assure best API use (e.g. UML modeling tools like Rational Rose)</b>				
	UML Modeling	Rational RequisitePro, Rational Software Modeller	Sparx Enterprise Architect, Sybase Powerdesigner	Rational RequisitePro, Rational Software Modeller
	Requirement Management	Rational RequisitePro, MS-Team Foundation	Sybase Powerdesigner, Rational Requisite Pro, Team Foundation Server	Rational RequisitePro
	Code Modeling	IBM RDT (RSA, RDA, Modeller)	IBM RDT (RSA, RDA, Modeller), Sparx Enterprise Architect	IBM RDT (RSA, RDA, Modeller)
<b>Presentation – Web Server, IVR and WAP Server. Technology that provides points of interaction (POI) or different presentations of data</b>				
	Web Server	MS-IIS 6.0/7.0	MS-IIS 6.0/7.0	Apache Server
	<b>Workflow Tools</b>			
	Human Workflow	Windows Workflow Foundation, and Source Code K2.NET	Windows Workflow Foundation, and Source Code K2.NET	IBM Websphere Suite
	Business Process	.NET Framework/MS BizTalk Server	.NET Framework/MS BizTalk Server	Cognos
	Business Object Components	MS-Performance Point Server/Cognos	MS Dynamics/MOSS 2007/Cognos	Cognos
	Unified Communication	MS Office Communications Server 2007	MS Office Communications Server 2007	MS Office Communications Server 2007
	Business Intelligence (BI)	SharePoint Server 2007, Microsoft Office Groove 2007, Business Intelligence, Forms Design, Electronic Forms Management, Business Process Management, and Collaboration Services.	MOSS 2007	Cognos
	Dashboard/Analytics	SharePoint Server 2007, Performance Point Server 2007 (formerly Proclarity & Business Scorecard Manager 2005)	MOSS 2007	Business Score Card Manager (MS-BSCM)
	P2P	Microsoft Office Groove 2007, Microsoft Office Groove Server 2007 (P2P)	MOSS 2007	Microsoft Office Groove 2007, Microsoft Office Groove Server 2007 (P2P)
	WCM	Stellent**	MOSS 2007 / Stellent	Stellent**
	Search Engine	FAST**	MOSS 2007 / FAST	FAST**
	<b>Web Presentation/Authoring</b>			
	Web Presentation Components	ASP.NET, IIS, MS-SP	ASP.NET, IIS, SharePoint 3.0/MOSS 2007	J2EE/JSP Version 4.3/Websphere Suite
	Business Transaction Components	Web Services, VS, with SOAP/WSDL	Windows Communication Foundation (WCF), Web Services	J2EE/JSP Version 4.3/Websphere Suite
	Data Access APIs	ADO, SCC	ADO, LINQ	J2EE 1.5/Websphere Suite
	Message Queue Components	MSMQ,.NET Framework	Windows Communication Foundation	IBM MQ Series
	<b>Portal</b>			
	Portal services	IBM Websphere Portal**	MOSS 2007 / IBM Websphere Portal	IBM Websphere Portal**
<b>Application – Application Server. Application server software that executes business logic</b>				
	<b>Application Server &amp; Components</b>			
	Application Server	Windows Server/BizTalk Server	Windows Server 2003/2008, BizTalk Server	IBM Websphere Suite
	Security Services	ADFS (Active Directory Federation Services) ADAM (Active directory Application Mode)	ADFS (Active Directory Federation Services) ADAM (Active directory Application Mode)	Sun Directory Server Sun Directory Server

	Product Category	.NET Standard Platform	.NET Revised Platform	Java Standard Platform
	Proxy Server	IBM Edge Server	IBM Edge Server, Internet Security & Acceleration (ISA) Server	IBM Edge Server
	Rights Management & Admin (Provisioning)	Windows RMS	Windows RMS	Sun Identity Management Server
<b>Integration – Integration Server, Inter-Enterprise Integration (IEI) Server. Enterprise application integration (EAI) software that connects different applications together, reformatting and routing data as necessary</b>				
	<b>Programming Interface</b>			
	Transactions	.NET Framework with SOAP/WSDL	.NET Framework, Windows Communication Foundation	JTA within J2EE 1.4/1.5 /Websphere Suite
	Message Queuing	MSMQ /MS-HIS	Windows Communication Foundation, MSMQ	IBM MQ Series
	Directory	ADSI, (OID), Active Directory	Active Directory Services Interface (ADSI)	Tivoli Directory Server
	Integration Connectors	MCF (Microsoft Connector Framework)	Windows Communication Foundation, Microsoft Host Integration Server	J2EE 1.4/1.5 (JCA)/Websphere Suite
	Email APIs	Messaging API (MAPI)/Resident in Exchange (SMTP native)	Messaging API (MAPI)/Resident in Exchange (SMTP native)	J2EE 1.4/1.5 (Java mail)/Websphere Suite (SMTP native)
<b>Database – DBMS, Data Access Middleware. Software that stores data for efficient record and field level retrieval along with data access and gateway functions</b>				
	<b>Database (DBMS)/Development</b>			
	Database	MS-SQL 2005	Oracle, MS – SQL Server 200x, SQL CE 3.5, DB2	Oracle 10G
	<b>Messaging/Email</b>			
	Microsoft Exchange	Microsoft Exchange Server 2003	Microsoft Exchange Server 2003	Microsoft Exchange Server 2003
	Mailbox Management	Symantec Enterprise Vault v6.0	Symantec Enterprise Vault v6.0	Symantec Enterprise Vault v6.0
	Message Hygiene (Antivirus / Anti-spam)	Microsoft Forefront & Exchange Hosted Filtering Services	Microsoft Forefront & Exchange Hosted Filtering Services	Microsoft Forefront & Exchange Hosted Filtering Services
	Mobile Messaging	RIM BlackBerry Enterprise Server 4	RIM BlackBerry Enterprise Server 4	RIM BlackBerry Enterprise Server 4
	Instant Messaging	Microsoft Office Communicator, Microsoft Office Communications Server 2007	Microsoft Office Communicator, Microsoft Office Communications Server 2007	Microsoft Office Communicator, Microsoft Office Communications Server 2007
	Storage / Restore	Data Protection Manager	Data Protection Manager	
<b>Server – Application Server HW and OS, Web Server HW and OS. Server hardware and operating system</b>				
	<b>Operating System</b>			
		Windows Server 2003	Windows Server 2008	Sun Solaris 10
<b>ACROSS-SYSTEM (UNIVERSAL) SERVICE PROVISION</b>				
<b>Directory – Directory Server. Software that provides centralized object attribute mapping (e.g.: user attributes such as access, component attributes such as cost center, location, etc)</b>				
	<b>Directory Services</b>			
	Directory Services	MS-Active Directory (primary) Tivoli Directory Server	Active Directory Tivoli Directory Server	MS-Active Directory (primary) Tivoli Directory Server
	Web Service Directory (UDDI)	Native to Windows Server 2003	Native to Windows Server 2003/2008	Registry Services (Native to Java-CAP)

	Product Category	.NET Standard Platform	.NET Revised Platform	Java Standard Platform
<b>Management – Service Level Management (SLM) monitoring products. To ensure that the required application service levels are being suitably supported by the infrastructure platform, an end-to-end enterprise level view of all contributing components must be available. Furthermore, the configuration of this management system and its contributing components must be mapped to enterprise-level policy definitions to ensure consistency</b>				
	<b>Management &amp; Monitoring Tools</b>			
	Monitoring Tools	HP Openview	HP OpenView	HP Openview
	Management Tools	Control C / HP Openview	Control C / HP Openview	Control C / HP Openview
	Project & Portfolio Management	Clarity	Clarity	Clarity
<b>Universal (incl. Tools)</b>				
	<b>Development/Language</b>			
	Language (such as C++, C#)	Any supported languages in VS	C# (recommended), VB.NET	J2EE 1.4/1.5/IBM Websphere Suite
	OS Platform & Runtime	Windows CLR	.NET Framework 3.5, .NET Framework Compact Edition	J2EE 1.4/1.5/IBM Websphere Suite
	Web, Desktop and Mobile GUI	Any supported languages in VS, SOAP/WSDL	ASP.NET, WinForms, Windows Presentation Foundation, Microsoft Office Infopath Forms	J2EE 1.4/1.5/IBM Websphere Suite
	Server-side Component	.NET FW with SOAP/WSDL	.NET Framework 3.5	JSP/EJB/Websphere Suite
	Persistent Objects	Any supported languages in VS, SOAP/WSDL	SOAP/WSDL	EJB 3.0/Websphere Suite
	Web Page Generation	Visual Studio, SharePoint Designer (formerly Frontpage), Microsoft Expression, with XML, with SOAP	Visual Studio 2005/08, SharePoint Designer 2007, Microsoft Expression Web	J2EE 1.4/1.5/IBM Websphere Suite
	In-Line Code/Code Behind	Any supported languages in VS with SOAP/WSDL	C# (recommended), VB.NET	J2EE 1.4/1.5/IBM Websphere Suite
	Relational Data Access	VS, SQL Server 2005	ADO.NET, LINQ	J2EE 1.4/1.5/JDBC/IBM Websphere Suite
	Queuing	MSMQ 4.0 with WSDL	WCF with MSMQ	J2EE 1.4/1.5/JMS/IBM Websphere Suite
	Asynchronous Invocation	.NET FW - RMT with SOAP/WSDL	.NET Framework & WCF	J2EE 1.4/1.5/EJB 3.0/IBM Websphere Suite
	Naming	ADSI with SOAP/WSDL	SOAP/WSDL	J2EE 1.4/1.5/JNDI/IBM Websphere Suite
	HTTP Engine	MS-IIS Server 6.0/7.0	MS-IIS Server 6.0/7.0	Apache Server
	XML	XML 6.0+ Core Service 6.0 RTM	To XML 6.0+/Core Service 6.0 RTM	JAXM, using SOAP/IBM Websphere Suite
	<b>Testing, Code Management &amp; QA</b>			
	Code and QA testing	Mercury Interactive Winrunner/IBM Rational Tools	Visual Studio 2008 Test Edition, Rational Tools	Mercury Interactive Winrunner/IBM Rat Tools
		VS Team Edition for Testers		VS Team Edition for Testers
	Scheduling	Windows Scheduler, MS-SQL	Windows Scheduler, MS-SQL, Windows Workflow Foundation	CA7/Tivoli Workload Scheduler
	Analysis Services	MS-SharePoint/SQL Analysis Service	FxCop (Visual Studio)	MS-SharePoint/SQL Analysis Service
	Source Code Library	MS Team Foundation Server 2005	MS Team Foundation Server 2005/08	Rational Clearcase
	Code Development	Visual Studio Team Edition , Sal Navigate	Visual StudioTeam System 2005/ 08	Rational Application Developer + RSA
	Code Reverse Engineering	Sybase Powerdesigner	Sparx Enterprise Architect, PowerDesigner	Sybase Powerdesigner
	Database Debug	Visual Studio Team Edition	Visual StudioTeam System 2005/ 08, SQL Navigator (for Oracle), DB2 Control Center (DB2)	Sybase-Powerdesigner

	<b>Product Category</b>	<b>.NET Standard Platform</b>	<b>.NET Revised Platform</b>	<b>Java Standard Platform</b>
	<b>Reporting</b>			
	Reporting	Crystal Report (Enterprise Version)	Crystal Report (Enterprise Version)	Crystal Report (Enterprise Version)
		SQL Server 2005 Reporting Services	SQL Server 2005 Reporting Services	SQL Reporting
	<b>Collaboration</b>			
	Collaboration	Plone/Zope **	MOSS 2007	Plone/Zope **
<b>**</b>	<b>CCAS Service</b>			