



# Taxonomy of the CommonPoint Application System

The CommonPoint™ application system provides two distinct sets of services:

- Application Frameworks for creating powerful, interactive applications
- System Services for handling data and communicating with other computers and the operating system

Charted below is an overview of this architecture.

For detailed information about system capabilities, refer to the CommonPoint developer documentation.

## Application Frameworks

Provide a rich set of features that allow the creation of interactive, easy to use applications.

### Embeddable Data Types

Support viewing and editing of complex data types, with full support for linking and embedding.

<b>Graphics Editing Framework</b>	Allows structured editing of graphical objects.
<b>Text Editing Framework</b>	Supports editing of text with support for styles and languages.
<b>Document Data Access Framework*</b>	Provides an embeddable, user manipulatable document component that represents a database query and its results.
<b>Time Media User Interface Framework</b>	Provides standard document parts that can be used to present video, audio, and other time media data.

### Desktop Frameworks

Provide support for the CommonPoint application model, including its user interface policy, its look and feel, and its compound document architecture.

<b>Workspace Frameworks*</b>	Allow developers to create extensions to Taligent's People, Places, and Things® user interface.			
<b>Presentation Framework</b>	Unifies a number of user-interface and document model mechanisms, making it easier to create fully featured, document-based applications for the CommonPoint application system.			
<b>Document Frameworks</b>	Provide a document model as the basis for data representation in the CommonPoint application system.	<b>Shared Document Framework</b> Allows compound documents to be used in a collaborative environment across the network.	<b>Compound Document Framework</b> Allows documents to support active linking and embedding of data from other documents.	<b>Basic Document Framework</b> Supports the basic document architecture of the system, including storage and command processing.
<b>User Interface Frameworks</b>	Support the creation and use of interactive user-interface elements.	<b>Cursor Tools</b> Supports creation of generic cursor tools which can be used to manipulate many data types.	<b>Dialegs</b> Allow sets of user-interface elements to be grouped together.	<b>Clipboard</b> Provides a mechanism for the user to store and retrieve data on the clipboard, within or between applications.
	<b>Windows</b> Provides a set of basic window types, and handles common window management operations.	<b>Frames</b> Provides a selectable, manipulatable frame around a view.	<b>Controls</b> Provide a wide range of interactive user-interface elements, including buttons, scroll bars, and menus.	<b>Drag and Drop</b> Provides an abstract protocol for direct manipulation of user-interface objects.
	<b>Actions</b> Allow a handler to be notified when the user performs an action.	<b>Input</b> Provides a mechanism for converting user input into user-interface events.	<b>User Interface Utilities</b> Provide miscellaneous services for creating user interfaces, including support for labels and decorations.	<b>Views</b> Provide a basic mechanism for dividing a user-interface into a hierarchical collection of views.

### Application Services

Supports media and data handling services needed to create industrial-strength interactive applications.

<b>Interoperability Services</b>	Provide the ability to have CommonPoint applications interoperate with other systems.	<b>OpenDoc™ and OLE™ Compatibility**</b> Provides interoperability with OpenDoc and OLE.	<b>Graphics Converters</b> Provide conversion of several industry-standard graphics formats into CommonPoint graphics format.	<b>Text Converters</b> Supports conversion of plain and styled text into CommonPoint text format.	<b>Data Translation Framework</b> Provides a framework for data exchange with other software systems.
<b>Printing</b>	Provides platform-independent and printer-independent support for printing.	<b>Document Printing</b> Supports the printing of CommonPoint documents.	<b>Print Jobs</b> Provides support for user-customizable and printer-customizable print jobs.	<b>Basic Printing</b> Provides a model of printing pages to a printer.	<b>Printing Devices</b> Provide a hardware-independent abstraction of a printer.
<b>Scanning*</b>	Provides an abstract mechanism for control of scanners.				
<b>Time Media</b>	Supports a rich set of time media data types.	<b>MIDI</b> Allows use of MIDI devices to produce and record music.	<b>Audio</b> Supports general sound production and recording facilities.	<b>Telephony</b> Allows voice communications to be integrated into CommonPoint applications.	<b>Video</b> Supports video playback and recording.
<b>Localization Services</b>	Provide support for multilingual and localizable user-interface elements and text.	<b>Date and Time Conversion</b> Provides language-sensitive support for converting dates and times into and from a textual form.	<b>Text Analysis</b> Provides language-sensitive support for text collation, pattern matching, and boundary searching.	<b>Text Input and Output</b> Supports transliteration, virtual keyboards, and other text I/O services.	<b>Text Scanning and Formatting</b> Supports the reading and writing of numbers and other binary data in a textual form.
<b>Text</b>	Supports styled, multilingual text data.	<b>Line Layout</b> Provides support for text direction, highlighting, and line-by-line display of multilingual styled text.	<b>Paragraph Styles</b> Provide support for paragraph styles, including indents, line spacing, and more.	<b>Text Styles</b> Provide support for text styles, including fonts, sizes, positioning, and color.	<b>Text and Style Storage Management</b> Provide storage support for textual information, including styles, text ranges, and positions.
<b>Graphics</b>	Provide services for modeling and rendering 2-D and 3-D graphics.	<b>2-D Graphics</b> Provide support for 2-D graphics: geometries, attribute bundles, transforms, and high-level 2-D graphics objects.	<b>3-D Graphics</b> Provide support for 3-D graphics: geometries, attribute bundles, transforms, and high-level 3-D graphics objects.	<b>Colors</b> Provide support for multiple color spaces and color matching.	<b>Font Support</b> Provides font rendering support independent of font format.
		<b>Sprites</b> Provide support for bitmap animation.	<b>Pixel Buffers</b> Provide an abstraction for onscreen and offscreen pixel buffers.	<b>Graphic Devices</b> Provide basic capabilities for graphics device drivers: rasterization, device transforms, and color mapping.	<b>Displays</b> Provide an abstract display screen.
					<b>Character Sets</b> Provide support for Unicode characters and other character sets via transcoding.

## System Services

Provide the set of system-level services upon which the CommonPoint application system is built.

### Enterprise Services

Provide a set of services that allow the CommonPoint application system to interoperate with other computers distributed within an enterprise.

<b>Data Access Framework</b>	Allows data on local or remote databases to be accessed, queried, and modified.	
<b>Caucus Framework</b>	Provides multicast communications facilities for collaborative applications.	
<b>System Management</b>	Supports administration of computers throughout the enterprise, including software installation, system configuration, maintenance, security, and support.	<b>Licensing Services</b> Provide an abstract software licensing mechanism to control software use and distribution.
		<b>Authentication Services*</b> Provide an abstract authentication mechanism to help ensure system security.
<b>Messaging Services*</b>	Provide store-and-forward messaging services independent of platform and protocol.	
<b>Concurrency Control &amp; Recovery</b>	Provides basic transaction-processing services to ensure the consistency of data accessed by multiple tasks.	
<b>Remote Object Call Services</b>	Provide a mechanism for invoking services of remote servers on CommonPoint and other systems.	

### Foundation Services

Provide a fundamental set of object services that make it easier to write object-oriented programs.

<b>Notification Framework</b>	Provides a system-wide mechanism to propagate change information from one object to another.			
<b>Identifiers</b>	Provide several different methods for associating a textual name with other data.	<b>Attributes</b> Provide a simple mechanism for associating names with arbitrary, immutable data.	<b>Properties</b> Provide a mechanism for storing collections of named data items; query mechanism for searches.	<b>Tokens</b> Provide a lightweight wrapper for static text strings, allowing text sharing and efficient comparisons.
<b>Object Storage</b>	Provides mechanisms to support the persistent storage of objects and the structuring of objects in memory.	<b>Archives</b> Allow collections of objects to be stored on disk and retrieved individually.	<b>Data Structures and Collections</b> Allow objects to be organized into various kinds of type-safe and efficient collections.	<b>Streams and Persistence</b> Provide a mechanism for converting a collection of objects into a persistent, canonical byte-encoded stream.
				<b>Safe Pointers</b> Provide several kinds of special wrapper classes that make using C++ pointers safer.
<b>Testing</b>	Provides a suite of tools and services to aid in the testing of objects.	<b>Assertions</b> Provide a mechanism for asserting invariants in a program, generating exceptions when not met.	<b>Test Framework</b> Provides a mechanism for executing, logging, and evaluating tests.	<b>User Interface Testing</b> Provides tools and services for driving the user interface of a CommonPoint application from a test.
				<b>Utility Tests</b> Provide standard tests for common object behaviors, including hashing and streaming.
<b>Math and Language Libraries</b>	Provide support for the CommonPoint application system's math and runtime libraries.	<b>Numerics</b> Provide a high-precision numeric environment, using a CommonPoint object interface or ANSI standard interface.	<b>Standard C and C++ Libraries</b> Provide support for the ANSI C runtime libraries and the proposed ANSI C++ libraries.	

### OS Services

Provide basic support for creating programs that work across a wide variety of host operating systems and hardware platforms.

<b>Communications</b>	Provide support for local and remote communications.	<b>Directory Services*</b> Provide a homogeneous view of the network's name spaces; supports DNS, X.500, AOCE, and DCE.	<b>Service Access Framework</b> Provides a framework for identifying and accessing network services.	<b>Message Streams</b> Provides mechanism to send data between tasks on local or remote machines, independent of underlying protocol.	<b>Protocols*</b> Provides support for various standard communications protocols, including TCP/IP, AppleTalk®, and Novell.
<b>File System</b>	Provides an object abstraction for manipulating volumes, directories, and files.				
<b>Time Services</b>	Provide a hardware-independent, customizable model of time.				
<b>Object Runtime Services</b>	Provide support for the CommonPoint application system's object runtime.	<b>Memory Heaps</b> Provide a multithread-safe way to allocate memory.	<b>Exceptions</b> Provide runtime support for C++ exceptions and a set of common exception types.	<b>Shared Libraries</b> Provide a mechanism for packaging code and data into dynamically loadable shared libraries.	<b>Metadata</b> Provides a mechanism for accessing type information; allows dynamic instantiation of an object at runtime.
<b>Microkernel Services</b>	Provide an abstract interface to microkernel facilities needed to run the CommonPoint application system, independent of the host operating system.	<b>Tasks and Threads</b> Provide abstractions for creating and managing tasks and threads.	<b>Interprocess Communication</b> Provides a mechanism for sending messages to tasks and threads on the local machine.	<b>Synchronization Services</b> Provide semaphores, monitors, and services to synchronize other threads.	<b>Virtual Memory Management</b> Provides a set of services that allow virtual memory segments to be created and managed.
				<b>System Shutdown</b> Provides a staged, well-defined protocol for shutting down the CommonPoint system.	

\* Not available in Beta.  
 \*\* Not available in Beta. May not be available on all platforms.  
 Features and functions are subject to change by Taligent without notice. Descriptions are high-level generalizations. For more specific information, refer to the CommonPoint developer documentation.