# Chapter 20: Choosing an Application Type

* 01/13/2010
* 9 minutes to read

For more details of the topics covered in this guide, see [Contents of the Guide](https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ee658086%28v%3dpandp.10%29).

# Contents

* Overview
* Application Types Summary
* Mobile Application Archetype
* Rich Client Application Archetype
* Rich Internet Application (RIA) Archetype
* Service Archetype
* Web Application Archetype

# Overview

This chapter will help you to understand the application types covered in this guide, the tradeoffs necessary, and the design impact for choosing an application type. After reading this chapter, you will be able to determine the appropriate application type for your scenario and requirements. The chapter provides brief details of each of the five basic application archetypes, and links to more detail information in other chapters.

The requirements, technology constraints, and type of user experience you plan to deliver will determine the application type you choose. For example, you must decide whether the clients you intend to serve will have a permanent network connection available, whether you must deliver rich media content to anonymous users for viewing in a Web browser, or whether you will predominantly service a small number of users on a corporate intranet.

Use the following application archetypes summary to review each application type, its description, and common scenarios. Use the table in this section to help you make an informed choice of application type, based on the benefits and considerations for each type.

# Application Archetypes Summary

The following are the common basic types of applications that you may decide to build:

* **Mobile applications**. Applications of this type can be developed as thin client or rich client applications. Rich client mobile applications can support disconnected or occasionally connected scenarios. Web or thin client applications support connected scenarios only. Device resources may prove to be a constraint when designing mobile applications.
* **Rich client applications**. Applications of this type are usually developed as stand-alone applications with a graphical user interface that displays data using a range of controls. Rich client applications can be designed for disconnected and occasionally connected scenarios if they need to access remote data or functionality.
* **Rich Internet applications**. Applications of this type can be developed to support multiple platforms and multiple browsers, displaying rich media or graphical content. Rich Internet applications run in a browser sandbox that restricts access to some features of the client.
* **Service applications**. Services expose shared business functionality and allow clients to access them from a local or a remote system. Service operations are called using messages, based on XML schemas, passed over a transport channel. The goal of this type of application is to achieve loose coupling between the client and the server.
* **Web applications**. Applications of this type typically support connected scenarios and can support different browsers running on a range of operating systems and platforms.

There are many other more specific types of application that you can design and build. In general, these types are specializations or combinations of the basic types described in this list.

## Application Type Considerations

The following table indicates the benefits and considerations for the common application archetypes.

| **Application type** | **Benefits** | **Considerations** |
| --- | --- | --- |
| Mobile applications | Support for handheld devices.  Availability and ease of use for out of office users.  Support for offline and occasionally-connected scenarios. | Input and navigation limitations.  Limited screen display area. |
| Rich client applications | Ability to leverage client resources.  Better responsiveness, rich UI functionality, and improved user experience.  Highly dynamic and responsive interaction.  Support for offline and occasionally connected scenarios. | Deployment complexity; however, a range of installation options such as ClickOnce, Windows Installer, and XCOPY are available.  Challenging to version over time.  Platform specific. |
| Rich Internet applications (RIA) | The same rich user interface capability as rich clients.  Support for rich and streaming media and graphical display.  Simple deployment with the same distribution capabilities (reach) as Web clients.  Simple upgrade and version updating.  Cross-platform and cross-browser support. | Larger application footprint on the client compared to a Web application.  Restrictions on leveraging client resources compared to a rich client application.  Requires deployment of a suitable runtime framework on the client. |
| Service applications | Loosely coupled interactions between client and server.  Can be consumed by different and unrelated applications.  Support for interoperability. | No UI support.  Dependent on network connectivity. |
| Web applications | Broad reach and a standards-based UI across multiple platforms.  Ease of deployment and change management. | Dependent on continual network connectivity.  Difficult to provide a rich user interface. |

Each application type can be implemented using one or more technologies. Scenarios and technology constraints, as well as the capabilities and experience of your development team, will drive your choice of technology.

The following sections describe each of the application types in more detail:

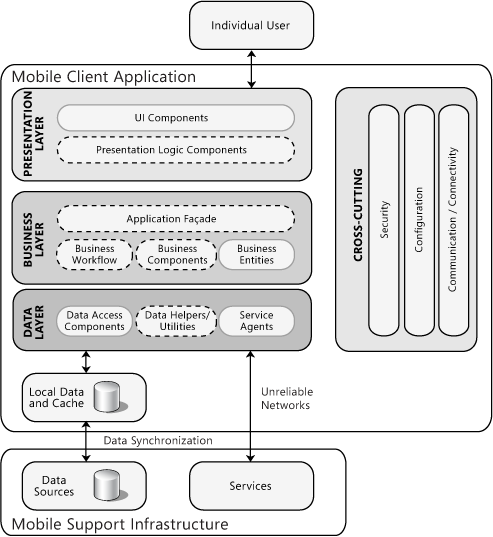
* Mobile Application Archetype
* Rich Client Application Archetype
* Rich Internet Application Archetype
* Service Archetype
* Web Application Archetype

This guide also contains details of some of the more specialized application types. For more information, see the following:

* Chapter 26 "[Designing Hosted and Cloud Services](https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ee658110%28v%3dpandp.10%29)"
* Chapter 27 "[Designing Office Business Applications](https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ee658085%28v%3dpandp.10%29)"
* Chapter 28 "[Designing SharePoint LOB Applications](https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ee658091%28v%3dpandp.10%29)"

# Mobile Application Archetype

A mobile application will normally be structured as a multilayered application consisting of user experience (presentation), business, and data layers, as shown in Figure 1.



**Figure 1**

The typical structure of a mobile application

When developing a mobile application, you may choose to develop a thin Web-based client or a rich client. If you are building a rich client, the business and data layers are likely to be on the device itself. If you are building a thin client, the business and data layers will be on the server. Mobile applications commonly make use of locally cached data to support offline or disconnected operation, and synchronize this data when connected. They may also consume services exposed by other applications, including S+S hosted services and Web services. Data source synchronization and other services are often exposed in a controlled way to a mobile client application through a specific server-based infrastructure.

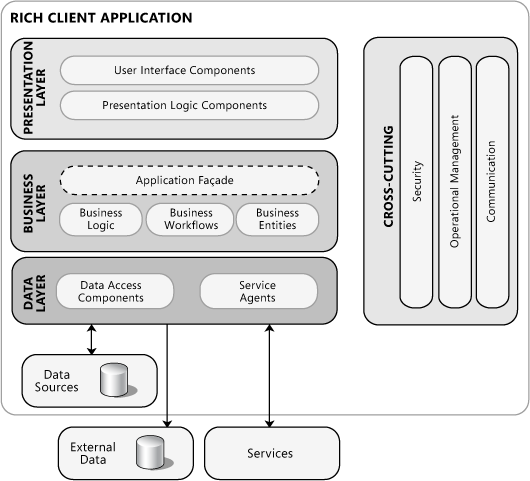
Consider using mobile applications if:

* Your users depend on handheld devices.
* Your application supports a simple UI that is suitable for use on small screens.
* Your application must support offline or occasionally connected scenarios. In this case, a mobile rich client application is usually the most appropriate.
* Your application must be device independent and can depend on network connectivity. In this case, a mobile Web application is usually the most appropriate.

To learn how to design a mobile application, see Chapter 24, "[Designing Mobile Applications](https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ee658108%28v%3dpandp.10%29)."

# Rich Client Application Archetype

Rich client user interfaces can provide a highly responsive, interactive, and rich user experience for applications that must operate in stand-alone, connected, occasionally connected, and disconnected scenarios. A rich client application will normally be structured as a multilayered application consisting of user experience (presentation), business, and data layers, as shown in the Figure 2.



**Figure 2**

The typical structure of a rich client application

A rich client application may use data stored on a remote server, data stored locally, or a combination of both. It may also consume services exposed by other applications, including S+S hosted services and Web services.

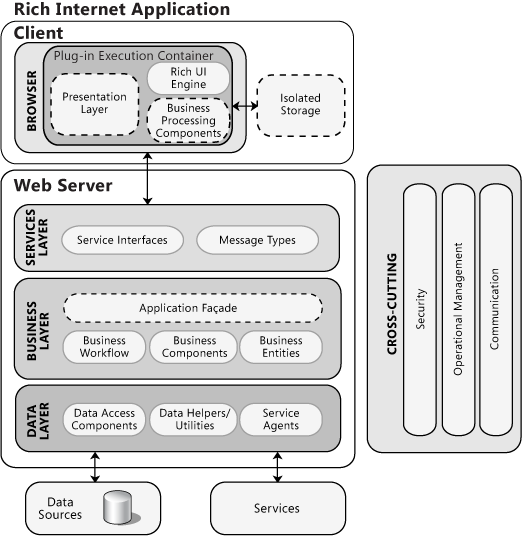
Consider using rich client applications if:

* Your application must support disconnected or occasionally connected scenarios.
* Your application will be deployed on client PCs.
* Your application must be highly interactive and responsive.
* Your application UI must provide rich functionality and user interaction but does not require the advanced graphics or media capabilities of a RIA.
* Your application must utilize the resources of the client PC.

To learn how to design a rich client application, see Chapter 22 "[Designing Rich Client Applications](https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ee658087%28v%3dpandp.10%29)."

# Rich Internet Application Archetype

A rich Internet application (RIA) runs in the browser in a sandbox. The benefits of a RIA over traditional Web applications include richer user experience, improved user responsiveness, and improved network efficiency. A RIA will normally be structured as a multilayered application consisting of user experience (presentation), service, business, and data layers, as shown in Figure 3.



**Figure 3**

The typical structure of a rich Internet application

RIAs generally depend on a client-side plug-in or hosted execution environment (such as a XAML runtime or Silverlight). This plug-in communicates with remote Web server hosts that generate the code and data consumed by the client plug-in or execution environment.

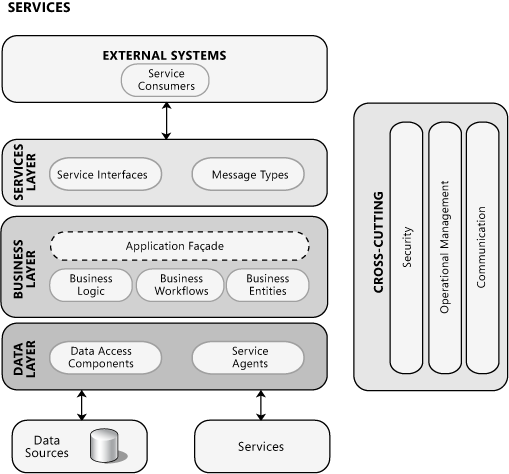
Consider using rich Internet applications if:

* Your application must support rich media and provide a highly graphical display.
* Your application must provide a rich, interactive, and responsive UI compared to Web applications.
* Your application will leverage client-side processing in a restricted manner.
* Your application will utilize client-side resources in a restricted manner.
* You want the simplicity of a Web-based deployment model.

To learn how to design a rich Internet application, see Chapter 23, "[Designing Rich Internet Applications](https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ee658083%28v%3dpandp.10%29)."

# Service Archetype

In the context of this guide, a service is a public interface that provides access to a unit of functionality. Services literally provide some programmatic service to the caller that consumes the service. A service application that exposes such services will normally be structured as a multilayered application consisting of service, business, and data layers, as shown in Figure 4.



**Figure 4**

The typical structure of a service application

Services are loosely coupled, and can be combined to provide functionality that is more complex. Services are distributable, and can be accessed from a remote machine as well as from the machine on which the service is running. Services are also message oriented. This means that the interfaces are defined by a Web Services Description Language (WSDL) document and operations are called using messages based on XML schemas, which are passed over a transport channel. In addition, services support a heterogeneous environment by focusing interoperability on the message/interface definition. If components can understand the message and interface definition, they can use the service regardless of their base technology.

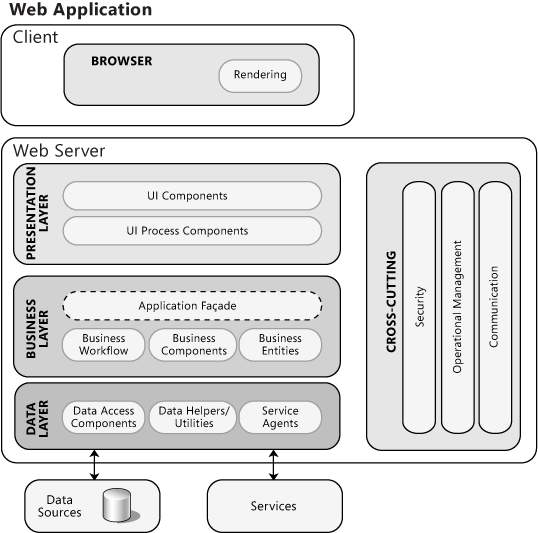
Consider using service applications if:

* Your application will expose functionality that does not require a UI.
* Your application must be loosely coupled with its clients.
* Your application must be shared with or consumed by other external applications.
* Your application must expose functionality that will be consumed by applications over the Internet, an intranet, or on the local machine.

To learn how to design services and service applications, see Chapter 25 "[Designing Service Applications](https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ee658114%28v%3dpandp.10%29)."

# Web Application Archetype

The core of a Web application is its server-side logic. This logic may be comprised of many distinct layers. A typical example is a three-layered architecture comprising presentation, business, and data layers, as shown in Figure 5.



**Figure 5**

The typical structure of a Web application

A Web application will normally access data stored on a remote database server. It may also consume services exposed by other applications, including S+S hosted services and Web services.

Consider using Web applications if:

* Your application does not require the rich UI and media support offered by a rich Internet application.
* You want the simplicity of a Web-based deployment model.
* Your user interface must be platform independent.
* Your application must be available over the Internet.
* You want to minimize client-side dependencies and resource consumption, such as disk or processor usage.

To learn how to design a Web application, see Chapter 21, "[Designing Web Applications](https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ee658099%28v%3dpandp.10%29)."