This page is specific to

Microsoft Visual Studio 2008/.NET Framework 3.5

Crystal Reports Basic for Visual Studio

**The ASP.NET Worker Process**

ASP.NET runs within a process known as the ASP.NET worker process. All ASP.NET functionality runs within the scope of this process.

A regular Web server contains only a single ASP.NET worker process. This is different from both Web farms and Web gardens:

* A Web farm contains multiple ASP.NET worker processes.

Each server in the group of servers handles a separate ASP.NET worker process.

* A Web garden contains multiple ASP.NET worker processes.

Each CPU in the SMP server handles a separate ASP.NET worker process.

**Choosing an ASP.NET worker process**

When a Web client connects to a Web farm or Web garden, one of the multiple ASP.NET worker processes is selected to run the request.

* In a Web farm, Network Load Balancing determines the ASP.NET worker process selected.
* In a Web garden, the ASP.NET worker process selected is determined by ASP.NET.

**State management with multiple ASP.NET worker processes**

When moving from a scenario with a single ASP.NET worker process (a normal Web server) to a scenario with multiple ASP.NET worker processes (a Web farm or Web garden), complications with state management are introduced.

Web pages are stateless, so a Web server must persist state through other means. Typical means to manage state on the Web server include Session State and the ASP.NET Cache.

|  |
| --- |
| **NoteNote** |
| Issues of persistence and state (within a single Web server) are discussed in detail in the section [Which Persistence Approach Should I Use with Crystal Reports?](http://msdn.microsoft.com/en-us/library/ms225433(VS.90).aspx). |

Both Session and Cache are contained within the memory space of a single ASP.NET worker process. But in a Web farm or Web garden, multiple ASP.NET worker processes work together simultaneously. The Session or Cache within any individual ASP.NET worker process cannot manage state across multiple processes.

Therefore, an additional layer is required for state management: an out-of-process Session State server that stores and retrieves state information for every ASP.NET worker process in the Web farm or Web garden. See [Out-of-Process Session State](http://msdn.microsoft.com/en-us/library/ms225449(VS.90).aspx).

**Out-of-Process Session State**

In Web farms and Web gardens, Session State must be shared across servers using some form of out-of-process Session State.

Two solutions are commonly used:

* Deploy the out-of-process Session State server that is provided with ASP.NET.
* Manually configure each Web server to store Session State data on a SQL Server.

**Serialization Requirement**

In both solutions, all objects to be stored must be serializable. Unless an object can be serialized, it cannot be persisted to an out-of-process state server.

|  |
| --- |
| **NoteNote** |
| Not all object models in the Crystal Reports SDK can be serialized. However, Crystal provides an upgrade solution for managing the state of reports with an out-of-process server. See [Serializable Object Models in Crystal Reports](http://msdn.microsoft.com/en-us/library/ms225448(VS.90).aspx). |