

Contents

- [Cannot connect to EntityServer](#)
- [Known type or serializer errors](#)
- [How to see if the EntityServer is running](#)
- [Error occurred while trying to make a request ...](#)
- [Mismatched IdeaBlade assemblies](#)
- [Also see](#)

An n-tier application is inherently a bit more complex to configure and deploy. Here we'll try to help with **troubleshooting** common problems.

Please note that in our experience most issues are caused by [application configuration](#) issues.

Cannot connect to *EntityServer*

You've received a message such as "The remote server returned an error: NotFound" or possibly "There is no endpoint listening on http://localhost:9009/EntityService.svc/sl that could accept calls from this application" (where the address is the address of your EntityServer). Both messages aren't quite as scary as they sound, and indicate that the client is unable to communicate with the server, often for a very simple and easily corrected reason.

You'll usually, but not always, receive additional information with the error message which attempts to provide some helpful information on the cause of the failure and possible solutions. Take a moment to read that information to see if it applies.

Also follow these troubleshooting steps:

- For Silverlight applications while debugging and testing in Visual Studio, the web project must be the startup project.
- For Windows Store applications while testing in Visual Studio, you must enable a loopback exemption. See <http://msdn.microsoft.com/en-us/library/windows/apps/hh780593.aspx> for more information.
- For Windows Phone 8 applications running in the emulator, use an IP address rather than *localhost*. See [here](#) for more information.
- [Make sure the service is running.](#)
- If you've included a `<system.serviceModel>` or `<objectServer>` information in your client config, make sure that all information is correct, and matches what the server is using. Double check the address by trying to browse to it to see if the service is running.
- Check the [debuglog](#) on the server. It will indicate the endpoints the server is listening on.
- Check the inner exceptions on the exception thrown. They may be inscrutable, but they may also contain enough information to diagnose the problem.

Known type or serializer errors

Every type which will be moved across tiers must be defined on both tiers, serializable, and a known type. See the [known type primer](#) for more information.

How to see if the *EntityServer* is running

Regardless of how the *EntityServer* is [hosted](#), you can always navigate - with your internet browser - to the "gateway" *EntityService*. The URL will differ based on the host, but will generally resemble one of the following:

- When hosted in IIS:
http://myserver/myapp/EntityService.svc
- In the ASP.NET Development web server (with port 9009):
http://localhost:9009/EntityService.svc
- When hosted as a Windows service or console application (again using port 9009 as a sample):
http://localhost:9009/EntityService

What you see on this page will tell you if the service is running, and if not the error which is keeping it from running.

Generally, because DevForce does not by default publish metadata, you will see a simple page that tells you the service is running but publishing is disabled:



You don't need to publish metadata, so seeing this is a good sign, and means the service is running.

If instead you see any type of "server error" page, then the service has not started. The page itself will contain diagnostic information on the cause of the problem. For example, here we see the service did not start because of a bad config file (in this case the xml was invalid).



Whenever your n-tier client application cannot communicate with the *EntityServer*, the first thing to check is if the *EntityServer* is in fact running. Trust us, this will save you, and IdeaBlade Support, a great deal of time!

Error occurred while trying to make a request ...

If your application is able to connect to the *EntityServer* but later gets an error such as "An error occurred while trying to make a request to URI ...", then it's likely that one or more of the types being sent cannot be serialized for some reason. You'll need to drill into the inner exceptions to see the original cause. For example, if the error occurs during a *SaveChanges*, a custom *IIdGenerator* will be at fault if it can't be serialized.

Mismatched IdeaBlade assemblies

The IdeaBlade class library versions must be the same on both client and server tiers. If you've recently upgraded DevForce, you should confirm that the assemblies on both the client and server have the same version.

Also see

[Troubleshooting an IIS Deployment](#)
[Troubleshooting a Service or Console Server](#)
[Understanding trace messages](#)
[Debugging](#)
[Known type primer](#)