OWIN and Katana, why decouple the application from the server?

Asked 11 years, 4 months ago Modified 11 years, 3 months ago

Viewed 10k times

40

I dont quite get this.. sure its cool to be able to self-host an application and it might be nice since if the IIS goes down for any reason... then all your sites goes down.. but if they are self-hosted then they live there own life in there own context.. which I guess is nice.. But i still dont get the pros of this.. do I skip alot of unnecessary things in the IIS pipe by using owin which speeds the application up, or.. whats the actual pros? (you dont need to list all of them if they are many :), but just so I get why you would like to use OWIN and Katana over the IIS)

Thanks in advance!

asp.net iis owin katana

Share

Improve this question

Follow

asked Aug 5, 2013 at 12:59

Inx

2,384 ● 7 ● 38 ● 55

See my blog blogs.msdn.com/b/webdev/archive/2014/04/23/…

– RickAndMSFT Apr 24, 2014 at 23:00

2 Answers

Sorted by: Highest score (default)

24

✓

Your question is valid.

You are probably mostly thinking of web facing web servers. Serving HTML (or json or websockets) can be used for so much more.

I use Nancy instead of IIS because I have a small application that I only want to reach from the inside while IIS is outward facing. Also - if a machine doesn't have IIS I don't have to install it.

Say you run OSX or *nix. Then you can have a very good language and ditto editor and run a simple web server on Mono.

Say you are using a micro controller like Netduino - IIS doesn't fit on it. I doubt Nancy runs there either - but you get the idea; not everything needs a 24 wheeler to haul data.

Say you are automating your house with a Win machine as heart and center. Now, do you really want to install IIS which does godsknowwhat or does it suffice with as little as possible?
If the automated house above is your pet F/OSS project and you are heavy on automated testing. Nancy is well known for that.

Another example take from the near-reality of mine. I have a snow flake server running an outdated Umbraco CMS. I put my soft gloves on before I touch this machine and only so gently.
If I need another web or json server on it Nancy would do the least possible imprint.

The OWIN argument is stale though as Microsoft seems to have Katana more or less out the door.

HTH

Share Improve this answer Follow

edited Aug 6, 2013 at 7:54

answered Aug 5, 2013 at 14:15

LosManos
7,682 ● 6 ● 59 ● 115

Ok, but if I get this right... your still having a server-machine, which runs Nancy..and then you have a few clients that connects to the server-machine, am I right? Why not use IIS on the hosting machine in the first place?.. or does Nancy allow you to manage the server from a remote machine? – Inx Aug 5, 2013 at 14:34

---

2 If, and only if, IIS *happens* to be on the server it is probably dedicated for something; typically serving HTML contents to users on the web. My application, on the other hand, is serving DevOps purposes. I want to use and remove my app without touching the IIS. I also filled out my answer above. – LosManos Aug 5, 2013 at 20:06

1 "I use Nancy instead of IIS" is a statement that needs some clarification. Nancy based applications can of course run well on IIS. Better say that you use Nancy self hosting. – Lex Li May 20, 2014 at 12:59

11

A OWIN and Katana overview from MS provides -- historical background and reasoning behind both projects can be found here:

http://www.asp.net/aspnet/overview/owin-and-katana/an-overview-of-project-katana

To summarize from the article:

OWIN: Its goal is not to be the next Web framework, but rather a specification for how Web frameworks and Web servers interact.

Project Katana: The Katana project represents the set of OWIN components that, while still open source, are built and released by Microsoft. These components include both infrastructure components, such as hosts and servers, as well as functional components, such as authentication components and bindings to frameworks such as SignalR and ASP.NET Web API. Katana brings many of the benefits of Node.js (and frameworks like it) -- the simplicity with which one could author and run a Web server -- without forcing the developer to throw out everything she knows about developing ASP.NET Web applications.

Share Improve this answer

Follow

answered Sep 8, 2013 at 21:53

Aaron

3,068 ● 2 ● 22 ● 46

Also take a look at the newer article Howard wrote msdn.microsoft.com/en-us/magazine/dn451439.aspx
– RickAndMSFT Apr 14, 2014 at 23:37

See also blogs.msdn.com/b/webdev/archive/2014/04/23/…
– RickAndMSFT Apr 24, 2014 at 23:01