

Best practice for moving live web apps to new servers? [closed]

Asked 16 years, 1 month ago Modified 15 years, 2 months ago

Viewed 3k times



2



Closed. This question is [opinion-based](#). It is not currently accepting answers.



Want to improve this question? Update the question so it can be answered with facts and citations by [editing this post](#).

Closed 9 years ago.

[Improve this question](#)

I am tasked with moving quite a few web apps including the databases to new servers, they are ASP.NET. I was not the one to create and setup these originally so I must try to figure out what exactly I need to replicate in order to not break anything and so the customers have no idea that anything was moved.

Does anyone have any tips for this, or know any automated ways?

Is there any software that can help with this?

I know the web app sends emails, so I will need to setup SMTP and it connects to a database so that I also will need to move. I suppose I should do this at night and take down the servers so I can move the database at it's latest state...

Any tips or tricks?

asp.net

iis

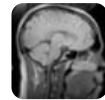
server-migration

Share

Improve this question

Follow

edited Oct 8, 2009 at 8:00



warren

33.4k ● 23 ● 89 ● 128

asked Nov 10, 2008 at 14:33



MetaGuru

43.7k ● 68 ● 195 ● 300

Just install the app on the new server, try out all functions and when all works you are ready to flip the switch. You might need to find a way to quickly migrate the latest database changes, but typically the dbs are small so you can just backup and restore it in a 1h window. (I asume a single person is not tasked with a larger web site). – [eckes](#) Aug 23, 2015 at 0:23

4 Answers

Sorted by:

Highest score (default)



This might help: [IIS 6.0 Migration Tool](#)

4



"The Internet Information Services 6.0 Migration Tool is a command line tool that automates several of the steps needed to move a Web application from IIS 4.0, IIS 5.0 or IIS 6.0 to a clean installation of Internet Information Services (IIS) 6.0 and Windows Server 2003.

The tool transfers configuration data, Web site content, and application settings to a new IIS 6.0 server if desired, or can move just application settings using the copy functionality. "

I don't think it will help with the database migration, though.

[Here's a link](#) to more detailed information about using the tool.

Share Improve this answer

edited Nov 10, 2008 at 14:56

Follow

answered Nov 10, 2008 at 14:39



DOK

32.8k ● 8 ● 63 ● 93

I added a link to the detailed instructions, which start off with a helpful flow chart. – **DOK** Nov 10, 2008 at 14:58

Good luck with your migration! – **DOK** Nov 10, 2008 at 15:27



3



May I suggest setting up the new servers in a staging environment. Allow business users to verify the functionality in the staging environment before flipping the switch and going live. Once you are ready, then bring over a fresh copy of the data. As far as the emails go... you should be fine with ASP.NET but some classic ASP programs require COM components in order to send email.

Share Improve this answer

answered Nov 10, 2008 at 14:39

Follow



Berkshire

700 ● 4 ● 10



1



The route I've taken in the past is to do a live/current copy (whatever that entails) of `$CURRENT_SERVER` to `$NEW_SERVER`. If the DB is not moving, just make sure `$NEW_SERVER` can reach `$DB_SERVER`, and that it will continue to run once copied.

Then update DNS to point to `$NEW_SERVER`.

After some period of time (2-3x the TTL for the DNS record), remove the old server.

Share Improve this answer

answered Oct 8, 2009 at 7:48

Follow



warren

33.4k ● 23 ● 89 ● 128



0



We just went through the same thing--bought a new server and had to transfer ASP.NET sites + Databases to the new server. We experienced problems with the IIS Migration tool, so we followed a "staging environment" approach, as stated in Berkshire's answer and had much success. When all issues are cleared from the staging environment, you can make it "live" with much confidence.

One other thing to watch out for is that you'll have to skim the ASP & VB/C# code for any hard-keyed connection strings to the database. These will have to change to reference the new location of the database.

Share Improve this answer

answered Nov 10, 2008 at 16:53

Follow



Kyle

250 ● 4 ● 17

Another good reason to put all of your connection strings in web.config. – [DOK](#) Nov 11, 2008 at 18:35
