Developer Documentation: Sharepoint Document Management vs. ScrewTurn Wiki [closed]

Asked 15 years, 10 months ago Modified 10 years, 6 months ago

Viewed 7k times

18

As it currently stands, this question is not a good fit for our Q&A format. We expect answers to be supported by facts, references, or expertise, but this question will likely solicit debate, arguments, polling, or extended discussion. If you feel that this question can be improved and possibly reopened, visit the help center for guidance.

Closed 12 years ago.

Background Info

I am working on setting up a method for my company's developers to share documentation and information about our various internal systems. This would range from information that would be useful for bringing a new employee up to speed, to descriptions of common problems users have with the systems and their resolutions.

This seemed like an ideal job for a wiki to me, and since our company only has the ability to host ASP.NET applications, I set about researching the available ASP.NET wikis. ScrewTurn Wiki seemed to be the most appropriate one, it's very full-featured and there are several plugins available that would be useful for my situation, including syntax highlighting and AD integration.

However, upon starting the process to have ScrewTurn deployed to our intranet, it was suddenly remembered that, hey, Sharepoint 2007 has a wiki, and since we already have Sharepoint set up, couldn't we just use that? I did a bit of evaluation of the Sharepoint "wiki" (in quotes because it barely qualifies), and was able to demonstrate that it wouldn't be suitable due to its many deficiencies, which I won't list here.

Now at this point, it's been suggested that perhaps I don't need a wiki at all, couldn't we just do everything in Word documents and use Sharepoint's document management functionality instead?

The actual question

So what I'm looking for is some additional ammunition, preferably from people with experience. What are examples of things that will be difficult or impossible with Sharepoint in the context of internal developer documentation? What is a wiki better at? Hey, I'm open-minded, what is Sharepoint better at?

What will make it worthwhile to deploy a wiki instead of simply using what we already have?

sharepoint documentation moss wiki

Share

Improve this question

Follow

edited Jun 20, 2020 at 9:12

Community Bot
1 ● 1

asked Feb 25, 2009 at 19:35

Chad Birch
74.5k ● 23 ● 154 ● 149

1 It seems pretty essential to at least list these deficiencies. If you're looking for ammunition rather than an objective, factual assessment, you came to the wrong site. – reinierpost Jul 6, 2016 at 11:00

11 Answers

Sorted by: Highest score (default)

19

✓

+150

I'm hoping this is going to be some use - extra points or not :)

SharePoint Server (SPS) or Windows SharePoint Services (WSS) is VERY good if you are working with office documents - you can version, check in/out, share, search etc. I dont think there is anything on the market which comes close for that function (it also does custom lists and stuff really well).

But as you point out, the WIKI function is, well, sub-standard.

For developer documentation, a wiki is much better, as you can easily interlink documents, and even if for the sole reason that developers tend to LIKE the wiki markup! Makes it feel like they are writing code, not a document. Well, ok, I like it. Word documents? endless frustration, especially for code snippits and things like API's. Wikis' usually handle code and structured formatting really, really well.

But here's the main point of me posting:

If you can host ASP.NET - and if you have SPS you can already! - then just install BOTH OF THEM. Make a new IIS virtual host, put STW in that one ('cos SPS will be in it's own virtual host)*. Massage the DNS a bit (so you can hit `http://wiki` or something), and just go for it.

As it's all internal to your company, and STW has windows security and versions, security isn't a huge issue. Each page can be linked to from anywhere - it's an HTML page after all - so you can link to it from the SPS wiki if you really want to. There is some lock in, I guess, but not a lot.

Here at BBC Worldwide, we use a number of technologies:

- Atlassian Confluence (WIKI) for some projects. I love this wiki, but it IS a big enterprise-y system.
- Screw Turn for some intra-department stuff.

- Trac for some other stuff (someone else set it up with SVN on our source repo, so it's used a little - it's nice, I rather like it, but it's a a*se to setup)
- SPS/WSS for documentation management.

there are links between the STW, Trac and SPS - eg we try not to store docs as attachments in trac, rather link to them in SPS, etc.

Works well.

As for ammo: SPS works well if it fits what you want to do (or YOU can fit what IT does). If not, you are either screwed, or need to do a lot of dev, which is really the same thing :)

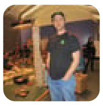
But aside from management getting all funny about it, I can't see a problem with installing both. STW is free after all. You have server(s).

And it gets dev's writing docs, which is seldom a bad thing. Ok, it's a bad thing if real humans have to read it, but docs for other dev's? All good.

*note: virtual HOST. Not virtual DIRECTORY.

Share Improve this answer Follow

edited Jun 9, 2014 at 18:00



Andrew Prock
7,087 ● 6 ● 41 ● 61

answered Mar 4, 2009 at 15:27

Nic Wise
8,119 ● 2 ● 32 ● 30

9

I've done both. After those experiences, here is my suggestion:

- If you need WYSIWYG and don't care about Sharepoint's bloat and lack of features (the wiki is simplistic, but has the major features you need in a wiki), go with that. It works as a wiki and helps when business people need to hit up the wiki -- lot less time teaching them to use it etc.
- If you can work with barebones and just want a simple wiki thats fast and flexible, you can't beat screwturn.

As for the why to use a wiki, whether screwturn or sharepoint -- it beats word docs in Sharepoint hands down. To use Sharepoint, first off you have to download the doc, edit, upload, or use Office integration which is chancy at best. Wiki's eliminate all the friction and make it dead simple to update. Eliminating friction is key, because when you have to work with word docs you end up just not updating them as often as you should. With a wiki, it's a 2 second pop in/edit/save transaction. With a word doc, it takes minutes to load word, open the doc, edit, worry about formatting, save, etc.

Share Improve this answer Follow

answered Feb 25, 2009 at 19:45

Chris Hynes
10.2k ● 2 ● 46 ● 55

We use MOSS 2007 Standard here, and the office 2007 integration is very reliable. Nothing chancy about it. No more time to load from SharePoint then from our file server. fwiw – MrChrister Feb 26, 2009 at 0:34

Is that local or internet? Intranet I can see, but every time I try Sharepoint over the internet, it's annoying. Tends to have wierd slowdowns, issues crop up, IE only, must have Office installed. – Chris Hynes Mar 7, 2009 at 21:56

Office and/or Sharepoint on a local network with very good hardware is going to be an order of magnatude slower the a average wiki on the other side of the world! – TFD Mar 10, 2009 at 2:55

6

Hey I've got experience here. We started to use ScrewTurn at work, but we were asked to switch since the company had already bought SharePoint. Documentation quality went down because Word documents are no match for a wiki, and as others have pointed out, SharePoint's is not up to par.

In my opinion, nothing beats a wiki for documentation. I think it's mainly the ability to create links to pages that don't yet exist. That way I can stub out documentation and the rest gets filled in as needed. It makes documentation more concise and readable.

With Word docs, hyperlinking is not very likely to be used, but it's so helpful to the readability of documentation. I could go on...

Share Improve this answer

Follow

answered Mar 10, 2009 at 20:29

Dan

3,725 ● 1 ● 35 ● 39

Have you moved to the Wiki feature in SharePoint or the Wiki edition of the Community Kit for SharePoint? – Nat Mar 10, 2009 at 20:44

We're back to using Word Docs etc. We tried out the Wiki feature and it never took hold. It just wasn't that good. Those of us who got to use ScrewTurn wish we could go back, but it's not our choice. – Dan Mar 13, 2009 at 13:33

4

As a generalisation, word is pretty awful for technical documents. Because it works very poorly for large, structured documents it encourages a proliferation of small documents with no cross-referencing between documents. Scale this up to any significant size and you will have trouble finding which documents pertain to an issue you are trying to investigate. As time goes on you will find people wasting more and more time hunting through a jumble of word documents, spreadsheets, visio diagrams and even powerpoint presentaions to find information that may or may not be recorded.

Putting them into sharepoint just forces you to route your searching through a browser.

Any technical documentation tool must support cross-referencing between documents to avoid losing those references. A wiki achieves this far better than word ever will. Also, it is much easier to incorported generated content such as data dictionaries or API docs into a wiki, and you can make the XRef targets stable.

Share Improve this answer Follow

answered Mar 4, 2009 at 16:06

ConcernedOfTunbridge Wells

66.5k ● 15 ● 148 ● 198

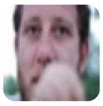
3

Here is an open source extended wiki for SharePoint that may be more suitable:

http://www.codeplex.com/CKS/Wiki/View.aspx?title=Enhanced%20Wiki%20Edition&referringTitle=Home

Share Improve this answer Follow

answered Feb 25, 2009 at 22:51



Jason

15.9k ● 3 ● 50 ● 73

1 Yeh, the enhanced CKS Wiki edition is the only way to compare Wikis. SharePoint out of the box is more like a "wi" – Nat Feb 26, 2009 at 21:32

2

First, check out [this StackOverflow thread](). A similar question was asked about Sharepoint wikis, and the answers there both pro and con are really useful and outline the "impossible in Sharepoint" part of your question.

I'm in the middle of trying to implement a team wiki in Sharepoint right now for almost he exact same thing as you, after having used Sharepoint for team documents for a couple years. My observations:

Sharepoint Document Libraries

My experience is that Sharepoint for documents is decent. It can be flaky and you'll sometimes lose edits or have performance issues, although a good Sharepoint support team can help with some of that. I did find the Sharepoint document libraries better than storing things on the network. It makes the docs and libraries easy to link to. Judicious use of the metadata properties can make a Sharepoint list view very useful for seeing what the document is, what it's for, and what status it is in, so our analysts, PMs, developers, etc., all know at a glance who has the ball and when something is waiting on them. That was started for a project a couple years ago; today Sharepoint 2007's workflows can probably provide notifications.

There's also version control for the documents. But as some others have noted doing things in documents has overhead and flaws involved in composing, downloading, and sharing. Check-outs of documents help, but loose cannon developers can download docs, edit them, and than store them elsewhere thus defeating the version control. This has happened to us, although that's a discipline issue and not Sharepoint's fault.

Sharepoint Wiki

Per the Chris Hynes answer, friction is a good word to describe the issue. For quick reference of a team's internal infrastructure details the wiki seems to work better for us. The wiki makes the data almost instantaneously visible, without having to click a link first and wait for Word to load. Edits are quicker and easier too...and although I'm still experimenting with it, the Sharepoint wikis have version control and allow "check outs" of wiki articles.

As you and the other SO thread pointed out, compared to other wiki engines, Sharepoint is lightweight. What is it better at? Well, I'm using it because it's there, didn't require any special setup, better than nothing, and frankly, it works fine for internal stuff that doesn't have to be robust or customer-facing. It's easy to sell to the boss because it doesn't cost anything more than what we've already spent. The wiki also can't as easily be compromised by the loose cannon types I mentioned earlier, and we'd be able to view the audit trail or rollback if necessary.

It's easy to add and cross-link entries, although if you want to embed graphics, you have to upload them first to a Sharepoint library. You can edit the HTML in the wiki too, although I'm not sure what limitations there are with that, and controlling the CSS I believe is not available without security access and Sharepoint Designer. So anything other than text (like tables) is not very robust.

So far I like it better than storing everything as documents. The immediacy of seeing and editing the data gives the wiki a real edge. It is certainly easier than pulling down a document and checking it back in. With the average documentation-averse developer, it helps to remove the barriers around keeping information updated.

Something else to consider. Do you expect your devs to be enthusiastic and actively involved in editing the wiki? If so, you might want the discussion features of a different wiki engine. My developers are like most; the two things they hate most are testing and documentation. So I'm the guy creating articles explaining the build procedure, the database environment usage, the application instances, server map, the SOX documentation checklist, etc. The other devs will mostly reference it and post minor updates. Our versioning and concurrency support aren't being heavily stressed.

Share Improve this answer Follow

edited May 23, 2017 at 11:45

Community Bot

1 ● 1

answered Mar 5, 2009 at 2:27

Bernard Dy

1,992 ● 2 ● 29 ● 39

▲

2

A wiki is much more likely to be used than a bunch of word docs simply because it is quicker and easier to find an entry to edit or create a new page than a bunch of word docs will ever be!

The question then becomes which wiki

Almost any wiki is better than the SharePoint wiki, that one is beyond a joke

Screwturn is excellent, has ACLs and WYSIWYG in v3, and is very easy to extend

e.g. Using the XSLT plugin is great for embedding the output of server processes, CI builds, tests, source control stats etc

e.g. We have a OLEDB plugin that reads critical values out of various company databases and embeds them into pages describing what the data is and what values it should have. We have one master page that lists all the pages that have data outside of prescribed ranges. Sort of a blend of FIT and a systems monitoring tool

If you have a management possition on "everything" has to be in SharePoint, then there are Screwturn plugins that render out the Screwturn pages as PDF, HTML, xml etc (you can convert to docx then) and have a script to dump changed screwturn pages each night to SharePoint for searching and "management" purposes

Share Improve this answer

Follow

edited Mar 10, 2009 at 2:59

answered Mar 10, 2009 at 2:53

TFD

24.5k ● 2 ● 35 ● 51

2

You might want to check out Confluence, as it is likely the leading enterprise wiki on the market. There is a SharePoint Connector for that wiki as well. As one of the contributors to the SharePoint Connector I am a little biased, but you are probably doing yourself a disservice if you don't check out a few of the options that are out there. Wikis can be powerful and contagious, but this won't happen if the wiki is sub-par.

Share Improve this answer

Follow

answered Mar 10, 2009 at 13:47

Kirk Liemohn

7,913 ● 9 ● 48 ● 58

1

Nobody could ever be happy with the SharePoint wiki by comparing it to any other wiki system.

So, don't compare it. It has the basic features necessary for a wiki to be useful: you can enter (somewhat) richly-formatted text, along with links to other pages, whether or not they exist. Clicking a link to a nonexistant page will take you to an edit page for the new, empty page, allowing you to save the page.

Yeah, the picture support sucks. You have to create the picture first, then paste the URL of the picture. So, take two minutes and create a picture library to post wiki pictures in.

Remember the main goals for a wiki - not to compete with other wiki tools, but to get ideas written down, quickly, and without stopping to structure them. Structure can be added later, if at all.

As others have said, telerik offers a replacement for the Rich Text editor, there is a CodePlex procject working (slowly) on improvements, and, people, it's SharePoint - if you don't like it, you can customize it. It's ASP.NET, web services and Windows Workflow Foundation.

I don't recommend anyone go out and buy a MOSS license just to implement a wiki (or blog, for that matter). But if you've already got the SharePoint infrastructure (perhaps as part of Visual Studio Team System Team Foundation Server), then go for it. I've seen several SharePoint wiki libraries used to **hugely** improve the amount and quality of developer documentation available to several groups within a large software development organization. Once we shut down discussion on how great other wikis were, quite a lot of documentation just happened, by itself, just like it's supposed to do.

Share Improve this answer Follow

answered Mar 11, 2009 at 3:30

John Saunders

162k ● 26 ● 249 ● 402

0

I use Sharepoint quite regularly, and I find it kind of slow and all kinds of annoying. If everything is already an Office document AND if you're company is willing to keep everyone's computer up to speed with Office versions, then Sharepoint can work fine. If you've got a variety of Office versions (like we do) then it's far less good. My machine has some older versions of office components and the integration doesn't always work correctly. I've learned to not try to use integration at all.

The most important thing with either solution is to make sure your people know how to use it. We had some problems early on (versions of files with different names instead of using the versioning built into sharepoint) that were really just gaps in people's training.

Share Improve this answer Follow

answered Mar 4, 2009 at 15:37

Michael Kohne

12k ● 3 ● 50 ● 82

0

We are currently using a combination of both, sharepoint for specs and formalized docs and a Wiki for "tacit knowledge". Each of our solutions has a page in the wiki and this works very well due to the ease of adding content.

Share Improve this answer Follow

answered Mar 10, 2009 at 11:21

Nicholas

1,750 ● 5 ● 22 ● 31