

# Are there any guarantees about when IIS might recycle while processing a 1-way WCF operation?

Asked 16 years ago   Modified 15 years, 11 months ago   Viewed 1k times

---



## Background

2

This question is in two parts.



I have a one-way WCF operation hosted in IIS 6. The following is my understanding of how this works:



\_1. IIS receives a request.

\_2. IIS sends an HTTP 202 response (thanks, I'll process this later).

\_3. IIS calls my one-way WCF operation.

Now control passes to my WCF operation which does the following:

\_4. Persist the request info to a transactional, durable store.

\_5. Start processing the request in the OLTP database.

\_6. If there's an error, repeat from step 5, or take some remedial action, then cleanup the data persisted in step 4.

## Question 1

Is my understanding of when IIS sends the HTTP 202 response correct?

## Question 2

If IIS recycles between step 2 and step 4, I could lose the request information before I've had a change to persist it but after the client thinks I've accepted the message. Are there any guarantees provided by IIS about when it will or won't recycle when there are requests pending?

---

PS: Please excuse the dodgy formatting. For some reason markdown was totally screwing up my numbered list items.

wcf

iis

recycle

Share

Improve this question

Follow

edited Jun 20, 2020 at 9:12



Community Bot

1 • 1

asked Nov 28, 2008 at 18:27



Damian Powell

8,765 ● 7 ● 50 ● 58

2 Answers

Sorted by:

Highest score (default)



1



I'm not sure your assumption is correct.

Even for one-way interaction, WCF can get very involved before the 202 is returned; if you use authentication, for example, it all has to happen before the method is called and before 202 is returned, so that if there are any problems they can be reported.

If you use wsHttpBinding, for example, out of the box, you will see 2-3 message exchanges resulting in 200 before the actual method call. this is to exchange security information and establish security context.

Admittedly, if you configured your service to have no security, this will not happen and it will return 202 immediately, but this suggests that it has to know whether it can do that from the WCF stack.

All that aside - I'm not sure what you are trying to achieve - which IIS recycling do you refer to? I'm not at all an expert on IIS, but I doubt it would recycle the host while something is actively executing; if you are referring to anyone manually restricting the application pool (or IIS, or the machine for that matter) I doubt there's much you can do.

If you have to know for sure that a message is not lost the only way I know to guarantee that is by employing reliable messaging in one way or another, which only acknowledges requests once persisted in a durable store;

Share Improve this answer

answered Jan 3, 2009 at 9:28

Follow



Yossi Dahan

5,357 ● 2 ● 30 ● 50

---

2 I suppose the recycle he refers to are the threshold-based recycle features in IIS (time, # requests, memory, etc). For all those thresholds, IIS will not recycle a worker process (w3wp.exe) while there are in-flight transactions or requests. IIS will stop sending new requests to a w3wp.exe that has a pending recycle. There is a process health ping whereby IIS will detect that the w3wp.exe is still alive and functioning, and while requests are active and the process is healthy, no recycle will occur. There is a timeout. Check the IIS process model reference page for the exact value. – [Cheeso](#) Jun 1, 2009 at 19:00

---

does this really answer the question? – [Simon](#) Dec 22, 2009 at 0:05

---

I obviously thought it did :-), which bit do you believe left unanswered? – [Yossi Dahan](#) Dec 30, 2009 at 13:42

---



In my opinion, there are a couple of things I would suggest:

0



The only way that I know of to manage process recycling in IIS is to look at the Application Pool your web site is using. If you open IIS Manager -> Application Pools ->



select your Pool then right-click properties, on the Recycling tab there are options in there that might help you. I guess that doesn't guarantee that you won't lose requests in a recycle, but setting a longer time reducing the probability of such a case.

However, it sounds like the kind of system you're looking to build is an ideal candidate for MSMQ, which can provide asynchronous service requests that are queued properly with very little overhead or manual plumbing. That would allow you to avoid the whole process recycling problem entirely. I'm not sure if that's an option for your scenario, but it might be something to look into.

Share Improve this answer

answered Dec 4, 2008 at 16:05

Follow



Steve

949 ● 1 ● 9 ● 18

---

Thanks for the answer, Steve. Unfortunately, it doesn't quite cover what I'm looking for. MSMQ is not possible in this case as it's a system that is processing from our front-end web server (where MSMQ is not available) which leaves us with the same problem. Namely that we have to be sure we can get a message from the client, through IIS, and into a queue, before a 202 response is sent. – [Damian Powell](#)

Dec 22, 2009 at 1:59

---