[**Adam Machanic**](http://sqlblog.com/blogs/adam_machanic/default.aspx)

FROM: <http://sqlblog.com/blogs/adam_machanic/archive/2011/04/30/twenty-nine-days-of-activity-monitoring-a-month-of-activity-monitoring-part-30-of-30.aspx>

Adam Machanic, Boston-based SQL Server developer, shares his experiences with programming, monitoring, and performance tuning SQL Server. And the occasional battle with the query optimizer.

**Twenty Nine Days of Activity Monitoring (A Month of Activity Monitoring, Part 30 of 30)**

*This post is part 30 of a 30-part series about the* [*Who is Active*](http://sqlblog.com/files/folders/35240/download.aspx) *stored procedure. A new post will run each day during the month of April, 2011. After April all of these posts will be edited and combined into a single document to become the basis of the Who is Active documentation.*

**Pop quiz:** What happens when you promise to write 30 posts on a topic, but you don’t plan properly and lose a bit of steam toward the end?

Answer: A recap post, a day or two early! Woo hoo!

**The month started** with a few introductory posts. A [history of monitoring](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/01/a-month-of-monitoring-part-1-of-30-a-brief-history-of-monitoring.aspx) (from my point of view), some information on Who is Active’s [design philosophy](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/02/who-is-active-design-philosophy-a-month-of-activity-monitoring-part-2-of-30.aspx), and a few words on the Who is Active [license](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/03/who-is-active-the-license-a-month-of-activity-monitoring-part-3-of-30.aspx).

The key takeaways from these first posts? The monitoring story has been pretty poor for most of SQL Server’s history. It’s better now, but it’s still a lot of work. Who is Active was created to make things more accessible. It’s free, with a few restrictions designed to keep people who make money from monitoring from making money on Who is Active. Anyone else can and should use and enjoy.

**Next I got new users up to speed**. My post on [installation](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/04/installing-who-is-active-a-month-of-activity-monitoring-part-4-of-30.aspx) covered security considerations, followed by a post on [why you may not see as much data as you're used](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/05/less-data-is-more-data-a-month-of-monitoring-part-5-of-30.aspx) to when you first start using Who is Active. I continued with a post on the [various options](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/06/who-is-active-options-a-month-of-monitoring-part-6-of-30.aspx) supported by the procedure (plus a nod to the online help mode) and another post describing the [default output columns](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/07/who-is-active-default-columns-day-1-a-month-of-activity-monitoring-part-7-of-30.aspx).

One of the more important posts of the month described how Who is Active treats [active requests and sleeping sessions](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/08/active-request-sleeping-session-a-month-of-activity-monitoring-part-8-of-30.aspx). This can be a confusing area for new users, since it's not always obvious what's going on unless you're looking in the right place at the output (the [status] column).

Once the refresher was finished I began covering **some of the basic options exposed by the procedure**. [Filters](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/09/deciding-what-not-to-see-a-month-of-activity-monitoring-part-9-of-30.aspx) were one of the first things I implemented, so it was a logical place to start (the "not" filters came much later). [Seeing the query text](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/10/commands-batches-and-the-mysteries-of-text-a-month-of-activity-monitoring-part-10-of-30.aspx) is a key part of the procedure's functionality, and that was next on my list. If you're not sure what the [sql\_text] column means, this post will set the record straight.

The ability to see the query text is nice, but so is the ability to get a [query plan](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/11/planning-for-success-a-month-of-activity-monitoring-part-11-of-30.aspx)--and of course Who is Active supports that too. And since everything a query does is transactional, the procedure allows users to collect information about [what those transactions are up to](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/12/the-almighty-transaction-a-month-of-activity-monitoring-part-12-of-30.aspx).

**The next section of the month was all about query processing**. I started with a couple of background posts: One on [how the query processor works](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/13/how-queries-are-processed-a-month-of-activity-monitoring-part-13-of-30.aspx) (at a somewhat high level), and another on what blocking [*really* means](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/14/blocking-blockers-and-other-b-words-a-month-of-activity-monitoring-part-14-of-30.aspx).

Who is Active exposes two different modes that help with analysis of real-time waits. My 15th post of the month covered the default--[lightweight--collection mode](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/15/seeing-the-wait-that-matters-most-a-month-of-activity-monitoring-part-15-of-30.aspx). My followup post covered the more extensive [full waits collection mode](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/16/seeing-all-of-the-waits-a-month-of-activity-monitoring-part-16-of-30.aspx).

**Sometimes a query plan and wait information isn't quite enough to diagnose a performance issue**. What if the query plan usually works well, but isn't performing properly only in this specific case? Who is Active [has a feature to help you figure that out](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/17/is-this-normal-a-month-of-activity-monitoring-part-17-of-30.aspx). And what if you need [a bit more information](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/18/getting-more-information-a-month-of-activity-monitoring-part-18-of-30.aspx) on some of the settings that the request is using?

**Mining wait information yields some amazing returns**. One of the additional pieces of information that you can get is the [actual name of the object that's causing a block](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/19/why-am-i-blocked-a-month-of-activity-monitoring-part-19-of-30.aspx). Another thing you can see is [the exact node within a query plan where a task is currently doing work](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/20/the-node-knows-a-month-of-activity-monitoring-part-20-of-30.aspx). You can (and should!) also use waits to figure out [whether *tempdb* is properly configured](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/21/analyzing-tempdb-contention-a-month-of-activity-monitoring-part-21-of-30.aspx). There are still more tricks you can play with waits, but they'll have to wait for a future post.

**Like wait information, lock data can also be mined.** However, there is so much locks information that the real challenge is rendering it in a human-readable manner. Who is Active does that using a special [custom XML format](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/22/the-key-to-your-locks-a-month-of-activity-monitoring-part-22-of-30.aspx). The procedure also helps with another type of blocker analysis, in the form of finding the "[lead blocker](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/23/leader-of-the-block-a-month-of-activity-monitoring-part-23-of-30.aspx)."

Once you've figured out what information you're interested in, why not **set up the output so that you can see the information the way you want to see it**? Who is Active helps with this by allowing users to [dynamically customize output columns and sort order](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/24/the-output-of-your-dreams-a-month-of-activity-monitoring-part-24-of-30.aspx) in a few different ways. And while you could always run Who is Active interactively, that might get a bit dull. [Automated data collection](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/25/capturing-the-output-a-month-of-activity-monitoring-part-25-of-30.aspx) is a much nicer methodology in many cases.

Most of the numbers shown by Who is Active are cumulative. But oftentimes it's more interesting to **compare data between two snapshots**. The procedure can help you do that, using its [delta mode](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/26/delta-force-a-month-of-activity-monitoring-part-26-of-30.aspx).

**What fun would a month of Who is Active be without an official release**? [v11.00 has numerous new features](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/27/who-is-active-v11-00-a-month-of-activity-monitoring-part-27-of-30.aspx), several of which were perfected this month thanks to feedback I received on the various posts in the series. It's great to have such enthusiastic users! Some of these new features didn't make it into prior posts, and other existing features are a bit hidden. So I did a writeup covering the [more important things that you may not have noticed](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/28/who-is-active-s-hidden-gems-a-month-of-activity-monitoring-part-28-of-30.aspx) while working with Who is Active.

I finished the month with a discussion on **security for slightly tougher situations**. I hope that the [module signing technique](http://sqlblog.com/blogs/adam_machanic/archive/2011/04/29/access-for-all-a-month-of-activity-monitoring-part-29-of-30.aspx) will allow Who is Active to be used in a number of places where security auditing requirements have made things difficult.

And that's that. A month of activity monitoring with Who is Active. Thank you for reading! Next, this text is going to be edited, expanded in places, and put into a much more comprehensive form. Watch your RSS reader for more information.

**Enjoy!**

Published Saturday, April 30, 2011 11:00 AM by [Adam Machanic](http://sqlblog.com/user/Profile.aspx?UserID=2103)

Filed under: [DMVs](http://sqlblog.com/blogs/adam_machanic/archive/tags/DMVs/default.aspx), [who is active](http://sqlblog.com/blogs/adam_machanic/archive/tags/who+is+active/default.aspx), [sp\_whoisactive](http://sqlblog.com/blogs/adam_machanic/archive/tags/sp_5F00_whoisactive/default.aspx), [activity monitoring](http://sqlblog.com/blogs/adam_machanic/archive/tags/activity+monitoring/default.aspx), [month of monitoring](http://sqlblog.com/blogs/adam_machanic/archive/tags/month+of+monitoring/default.aspx)

**Comment Notification**

If you would like to receive an email when updates are made to this post, please register [here](http://sqlblog.com/user/CreateUser.aspx?ReturnUrl=/blogs/adam_machanic/archive/2011/04/30/twenty-nine-days-of-activity-monitoring-a-month-of-activity-monitoring-part-30-of-30.aspx)

Subscribe to this post's comments using [RSS](http://sqlblog.com/blogs/adam_machanic/commentrss.aspx?PostID=35301)

**Comments**

|  |  |
| --- | --- |
| http://sqlblog.com/users/avatar.aspx?userid=3537 | [**mjswart**](http://sqlblog.com/utility/Redirect.aspx?U=http%3a%2f%2fmichaeljswart.com) **said:**  Good work! April is finally over.  I appreciate all the work that went into this month and into "Who Is Active".  May 1, 2011 6:12 PM |
| http://sqlblog.com/utility/anonymous.gif | [**Fabricio Lima**](http://sqlblog.com/utility/Redirect.aspx?U=http%3a%2f%2ffabriciolima.net%2fblog%2f) **said:**  Congratulations for this project!!!  Thanks a lot.  May 2, 2011 10:31 AM |
| http://sqlblog.com/utility/anonymous.gif | **Narayana said:**  Thank you! I've been using this SP for months, and really love it. Thanks for the updates!  May 6, 2011 8:20 AM |
| http://sqlblog.com/utility/anonymous.gif | **CD said:**  What a journey....TNX awesome write up.  May 9, 2011 4:06 PM |
| http://sqlblog.com/utility/anonymous.gif | **M A Srinivas said:**  I am getting the following error in additional\_info  <msdb\_query\_error>Mutator 'modify()' on 'additional\_info' cannot be called on a null value.</msdb\_query\_error>  May 31, 2011 5:22 AM |
| http://sqlblog.com/users/avatar.aspx?userid=2103 | [**Adam Machanic**](http://sqlblog.com/utility/Redirect.aspx?U=http%3a%2f%2fsqlblog.com%2fblogs%2fadam_machanic) **said:**  Hi M A,  Thanks for the error report. Can you share the entire additional\_info structure, as well as the value in the program\_name column and the parameters you're passing to sp\_whoisactive?  Thanks,  Adam  May 31, 2011 10:08 AM |
| http://sqlblog.com/utility/anonymous.gif | **IJeb Reitsma said:**  Thanks for these great 30 posts! I learned a lot from it, not only about the tool but also about SQL Server.  I looked at the tool for the first time about a year ago. It is very good for capturing real time information.  But for me the tool did not seem very useful for datacollection over a longer period of time or normal user activity baselining.  At your TechEd session you talked about using this tool for creating a baseline of normal user activity and resource usage during the day.  I am interested in doing that with this tool.  I can do that by running the tool every minute or hour or so and capturing the output in a table (of all the users sessions).  But if I do that the result is strange because if a session is 'sleeping' at the time of capturing the data I get session wide information (that's what I need), but if the session is busy at the time of snapshot creation it shows information about the individual request (and thats not very usefull for activity baselining).  Another thing is that I do have to calculate the deltas by myself afterwards.  The 'delta' feature included in the tool seems to be designed for capturing information of running requests.  To be short, my question is; how would you recommend creating a baseline using this tool?  June 2, 2011 8:37 AM |
| http://sqlblog.com/users/avatar.aspx?userid=2103 | [**Adam Machanic**](http://sqlblog.com/utility/Redirect.aspx?U=http%3a%2f%2fsqlblog.com%2fblogs%2fadam_machanic) **said:**  Hi IJeb,  I should probably write a blog post on this topic because it brings into question the entire methodology of baselining. The point of a baseline in a production environment is to establish what constitutes "normal."  For example, you could collect every 5 minutes during a given busy interval (i.e. if your app's busiest time is from 10:00 a.m. - 12:00 a.m. each day), and over the course of a few days note that it's expected that during that interval there will be between, e.g., 50 and 60 active requests, 20 of them for stored procedure X, 20 for stored procedure Y, and 20 for stored procedure Z. That's the kind of information that I would make part of an activity baseline.  Of course that's only one part of the baseline. You should also baseline performance counters and other information, which can be correlated when things change to figure out why the change occurred.  I'm not sure why you need session-level information. Do you have sessions that connect and issue many stored procedure (or ad hoc SQL) calls? Most applications these days don't operate that way, so WIA was not designed to push much of that kind of data. That said, if a lot of people tell me I'm dead wrong I can figure out how to collect more information at that level.  Unfortunately we weren't given nearly enough time to get into all of these things at a deep level in the TechEd session. My precon at this year's PASS conference will cover the topic in great detail.  Hope that helps a bit,  Adam  June 2, 2011 10:20 AM |
| http://sqlblog.com/utility/anonymous.gif | **M A Srinivas said:**  Hello Adam  Here are the details requested on SQL 2005 SP4  <additional\_info>   <text\_size>1024</text\_size>   <language>us\_english</language>   <date\_format>mdy</date\_format>   <date\_first>7</date\_first>   <quoted\_identifier>ON</quoted\_identifier>   <arithabort>ON</arithabort>   <ansi\_null\_dflt\_on>ON</ansi\_null\_dflt\_on>   <ansi\_defaults>OFF</ansi\_defaults>   <ansi\_warnings>ON</ansi\_warnings>   <ansi\_padding>ON</ansi\_padding>   <ansi\_nulls>ON</ansi\_nulls>   <concat\_null\_yields\_null>ON</concat\_null\_yields\_null>   <transaction\_isolation\_level>ReadCommitted</transaction\_isolation\_level>   <lock\_timeout>-1</lock\_timeout>   <deadlock\_priority>0</deadlock\_priority>   <row\_count>0</row\_count>   <command\_type>UPDATE STATISTIC</command\_type>   <agent\_job\_info>     <job\_id>BBB33B89-E604-4EB9-8ECB-908FD4095302</job\_id>     <step\_id>2</step\_id>     <job\_name />     <step\_name />     <msdb\_query\_error>Mutator 'modify()' on 'additional\_info' cannot be called on a null value.</msdb\_query\_error>   </agent\_job\_info>  </additional\_info>  ------------------------------------------------  Parameters  -----------------------------------------------  @filter sysname = '',  @filter\_type VARCHAR(10) = 'session',  @not\_filter sysname = '',  @not\_filter\_type VARCHAR(10) = 'session',  --Retrieve data about the calling session?  @show\_own\_spid BIT = 0,  --Retrieve data about system sessions?  @show\_system\_spids BIT = 0,  --Controls how sleeping SPIDs are handled, based on the idea of levels of interest  --0 does not pull any sleeping SPIDs  --1 pulls only those sleeping SPIDs that also have an open transaction  --2 pulls all sleeping SPIDs  @show\_sleeping\_spids TINYINT = 1,  --If 1, gets the full stored procedure or running batch, when available  --If 0, gets only the actual statement that is currently running in the batch or procedure  @get\_full\_inner\_text BIT = 0,  --Get associated query plans for running tasks, if available  --If @get\_plans = 1, gets the plan based on the request's statement offset  --If @get\_plans = 2, gets the entire plan based on the request's plan\_handle  @get\_plans TINYINT = 1,  --Get the associated outer ad hoc query or stored procedure call, if available  @get\_outer\_command BIT = 1,  --Enables pulling transaction log write info and transaction duration  @get\_transaction\_info BIT = 1,  --Get information on active tasks, based on three interest levels  --Level 0 does not pull any task-related information  --Level 1 is a lightweight mode that pulls the top non-CXPACKET wait, giving preference to blockers  --Level 2 pulls all available task-based metrics, including:  --number of active tasks, current wait stats, physical I/O, context switches, and blocker information  @get\_task\_info TINYINT = 2,  --Gets associated locks for each request, aggregated in an XML format  @get\_locks BIT = 1,  --Get average time for past runs of an active query  --(based on the combination of plan handle, sql handle, and offset)  @get\_avg\_time BIT = 1,  --Get additional non-performance-related information about the session or request  --text\_size, language, date\_format, date\_first, quoted\_identifier, arithabort, ansi\_null\_dflt\_on,  --ansi\_defaults, ansi\_warnings, ansi\_padding, ansi\_nulls, concat\_null\_yields\_null,  --transaction\_isolation\_level, lock\_timeout, deadlock\_priority, row\_count, command\_type  --  --If a SQL Agent job is running, an subnode called agent\_info will be populated with some or all of  --the following: job\_id, job\_name, step\_id, step\_name, msdb\_query\_error (in the event of an error)  --  --If @get\_task\_info is set to 2 and a lock wait is detected, a subnode called block\_info will be  --populated with some or all of the following: lock\_type, database\_name, object\_id, file\_id, hobt\_id,  --applock\_hash, metadata\_resource, metadata\_class\_id, object\_name, schema\_name  @get\_additional\_info BIT = 1,  --Walk the blocking chain and count the number of  --total SPIDs blocked all the way down by a given session  --Also enables task\_info Level 1, if @get\_task\_info is set to 0  @find\_block\_leaders BIT = 1,  --Pull deltas on various metrics  --Interval in seconds to wait before doing the second data pull  @delta\_interval TINYINT = 10,  --List of desired output columns, in desired order  --Note that the final output will be the intersection of all enabled features and all  --columns in the list. Therefore, only columns associated with enabled features will  --actually appear in the output. Likewise, removing columns from this list may effectively  --disable features, even if they are turned on  --  --Each element in this list must be one of the valid output column names. Names must be  --delimited by square brackets. White space, formatting, and additional characters are  --allowed, as long as the list contains exact matches of delimited valid column names.  @output\_column\_list VARCHAR(8000) = '[dd%][session\_id][sql\_text][sql\_command][login\_name][wait\_info][tasks][tran\_log%][cpu%][temp%][block%][reads%][writes%][context%][physical%][query\_plan][locks][%]',  --Column(s) by which to sort output, optionally with sort directions.  --Valid column choices:  --session\_id, physical\_io, reads, physical\_reads, writes, tempdb\_allocations,  --tempdb\_current, CPU, context\_switches, used\_memory, physical\_io\_delta,  --reads\_delta, physical\_reads\_delta, writes\_delta, tempdb\_allocations\_delta,  --tempdb\_current\_delta, CPU\_delta, context\_switches\_delta, used\_memory\_delta,  --tasks, tran\_start\_time, open\_tran\_count, blocking\_session\_id, blocked\_session\_count,  --percent\_complete, host\_name, login\_name, database\_name, start\_time, login\_time  --  --Note that column names in the list must be bracket-delimited. Commas and/or white  --space are not required.  @sort\_order VARCHAR(500) = '[session\_id] ASC',  --Formats some of the output columns in a more "human readable" form  --0 disables outfput format  --1 formats the output for variable-width fonts  --2 formats the output for fixed-width fonts  @format\_output TINYINT = 1,  --If set to a non-blank value, the script will attempt to insert into the specified  --destination table. Please note that the script will not verify that the table exists,  --or that it has the correct schema, before doing the insert.  --Table can be specified in one, two, or three-part format  @destination\_table VARCHAR(4000) = '',  --If set to 1, no data collection will happen and no result set will be returned; instead,  --a CREATE TABLE statement will be returned via the @schema parameter, which will match  --the schema of the result set that would be returned by using the same collection of the  --rest of the parameters. The CREATE TABLE statement will have a placeholder token of  --<table\_name> in place of an actual table name.  @return\_schema BIT = 0,  @schema VARCHAR(MAX) = NULL OUTPUT,  --Help! What do I do?  @help BIT = 0  --~  Thanks  Srinivas  June 17, 2011 6:15 AM |
| http://sqlblog.com/utility/anonymous.gif | **Ken Trock said:**  Adam, has anyone asked you if WIA could be an official part of SQL Server either now or in the future? It's obviously way better than sp\_who2 and surely superior to the ad-hoc versions many of us have created for our own use.  June 24, 2011 1:03 PM |
| http://sqlblog.com/users/avatar.aspx?userid=2103 | [**Adam Machanic**](http://sqlblog.com/utility/Redirect.aspx?U=http%3a%2f%2fsqlblog.com%2fblogs%2fadam_machanic) **said:**  Hi Ken,  Many users have commented that it would be nice to see it as part of SQL Server, and I have tried a few times to set the wheels in motion. Alas, these efforts have invariably ended in long e-mail chains with no definitive answers, followed by the people on the Microsoft side no longer responding to my e-mails.  If you have some pull or a way in the door, please drop me a line privately and we'll talk. I would certainly love to see this happen.  --Adam  June 24, 2011 2:01 PM |
| http://sqlblog.com/utility/anonymous.gif | **Sreekanth CHAVA said:**  Thanks a lot Adam, It was a nice project, I got to one of your post while I was searching for scripts on blocking, at last I have the script which I have being looking for a long time, yahooo..... this will go into my library.  Congratulations and once again thank you for all your hard work.  Cheers  SreekanthCHAVA  January 5, 2012 6:34 PM |
| http://sqlblog.com/utility/anonymous.gif | **Ben said:**  Thanks for the series.  I read it last April and had need to review it again a little today.  At the top of each day, you wrote:  "After April all of these posts will be edited and combined into a single document to become the basis of the Who is Active documentation."  Did this ever happen?  January 10, 2012 9:37 PM |
| http://sqlblog.com/users/avatar.aspx?userid=2103 | [**Adam Machanic**](http://sqlblog.com/utility/Redirect.aspx?U=http%3a%2f%2fsqlblog.com%2fblogs%2fadam_machanic) **said:**  Ben: Working on it :-)  January 11, 2012 10:45 AM |
| http://sqlblog.com/utility/anonymous.gif | [**Adam Machanic**](http://sqlblog.com/utility/Redirect.aspx?U=http%3a%2f%2fsqlblog.com%2fblogs%2fadam_machanic%2farchive%2f2012%2f03%2f22%2freleased-who-is-active-v11-11.aspx) **said:**  It's been several months since the last Who is Active fix, so I thought I'd call this one out specifically  March 22, 2012 5:30 PM |
| http://sqlblog.com/utility/anonymous.gif | **Denis said:**  In my toy 2012 SQLXPRESS, I got the following error when use WIA, is this a known issue?  Microsoft SQL Server 2012 - 11.0.2100.60 (Intel X86)  Feb 10 2012 19:13:17  Copyright (c) Microsoft Corporation  Express Edition on Windows NT 6.1 <X86> (Build 7601: Service Pack 1)  (1 row(s) affected)  Warning: The join order has been enforced because a local join hint is used.  Msg 8631, Level 17, State 1, Procedure sp\_WhoIsActive, Line 1706  Internal error: Server stack limit has been reached. Please look for potentially deep nesting in your query, and try to simplify it.  October 22, 2012 12:00 PM |
| http://sqlblog.com/users/avatar.aspx?userid=2103 | [**Adam Machanic**](http://sqlblog.com/utility/Redirect.aspx?U=http%3a%2f%2fsqlblog.com%2fblogs%2fadam_machanic) **said:**  Hi Denis,  Not expected at all. What parameters are you passing in?  Thanks,  Adam  October 22, 2012 1:40 PM |
| http://sqlblog.com/utility/anonymous.gif | **Denis said:**  No parameter passed in.  Here someone described the long case statement failed in 2012 version with same error. Not sure how many CASEs in the WIP procedure.  <http://connect.microsoft.com/SQLServer/feedback/details/718795/tsql-case-statement-fails-internal-error-server-stack-limit-has-been-reached-8631>  Thanks,  Denis  October 23, 2012 10:50 AM |
| http://sqlblog.com/users/avatar.aspx?userid=2103 | [**Adam Machanic**](http://sqlblog.com/utility/Redirect.aspx?U=http%3a%2f%2fsqlblog.com%2fblogs%2fadam_machanic) **said:**  Hi Denis,  Could be something else other than CASE expressions -- lots of things can cause the stack to grow. Are you on Who is Active v11.11 ? If not, please try that.  Also, can you update your Express instance to CU4 and see if it still happens? That CU includes a few major query processor fixes.  Thanks,  Adam  October 23, 2012 11:45 AM |
| http://sqlblog.com/utility/anonymous.gif | **Denis said:**  Updated to  CU4 :  -------  SQL Server 2012 - 11.0.2383.0 (Intel X86)  Oct  6 2012 11:13:58  Copyright (c) Microsoft Corporation  Express Edition on Windows NT 6.1 <X86> (Build 7601: Service Pack 1)  ---  Same error. I do use v11.11.  Denis  October 24, 2012 7:58 AM |
| http://sqlblog.com/utility/anonymous.gif | **Denis said:**  Sorry. I've realized I am using v11.00. I downloaded v11.11. It has no problem in SQL Express 2012. Thanks for all your great posts !  October 24, 2012 2:40 PM |
| http://sqlblog.com/utility/anonymous.gif | **Beneberu said:**  "After April all of these posts will be edited and combined into a single document to become the basis of the Who is Active documentation." Were you able to combine and post the documnet? If so,  please post the link.  Thanks.  November 26, 2013 3:35 PM |
| http://sqlblog.com/utility/anonymous.gif | **Afredo said:**  Could you please explain for me that Microsoft has provided activity monitor tool to troubleshoot query, so what is the benefits of WhoisActive compared with Activity Monitor.  Many thanks if you could give me a blue.  January 24, 2014 2:07 AM |
| http://sqlblog.com/utility/anonymous.gif | **Afredo said:**  Sorry for the previous post, I have spent time to read your articles about WhoisActive from the beginning and I found the answer.  Thanks for great work.  January 24, 2014 2:14 AM |
| http://sqlblog.com/utility/anonymous.gif | **Bigpa said:**  Brilliant tool(belt), thanks a lot !  This is the gift that keeps on giving  April 13, 2014 2:55 PM |
| http://sqlblog.com/utility/anonymous.gif | **Sameera said:**  Thank you for the briliant 30 articles on activity monitoring and WhoIsActive.  By the way you can use wso2 Business Acitivity Monitor (<http://wso2.org/projects/bam>) to take this effort a bit further by configuring the system to generate meaningful results out of the actvities produced. Or you may use wso2 Complex Event Processor (<http://wso2.org/projects/cep>) to analyse the activity streams and perform some actions autonomously.  September 19, 2014 12:01 PM |

**About Adam Machanic**

Adam Machanic is a Boston-based SQL Server developer, writer, and speaker. He focuses on large-scale data warehouse performance and development, and is author of the award-winning SQL Server monitoring stored procedure, sp\_WhoIsActive. Adam has written for numerous web sites and magazines, including SQLblog, Simple Talk, Search SQL Server, SQL Server Professional, CoDe, and VSJ. He has also contributed to several books on SQL Server, including "SQL Server 2008 Internals" (Microsoft Press, 2009) and "Expert SQL Server 2005 Development" (Apress, 2007). Adam regularly speaks at conferences and training events on a variety of SQL Server topics. He is a Microsoft Most Valuable Professional (MVP) for SQL Server, a Microsoft Certified IT Professional (MCITP), and an alumnus of the INETA North American Speakers Bureau.