

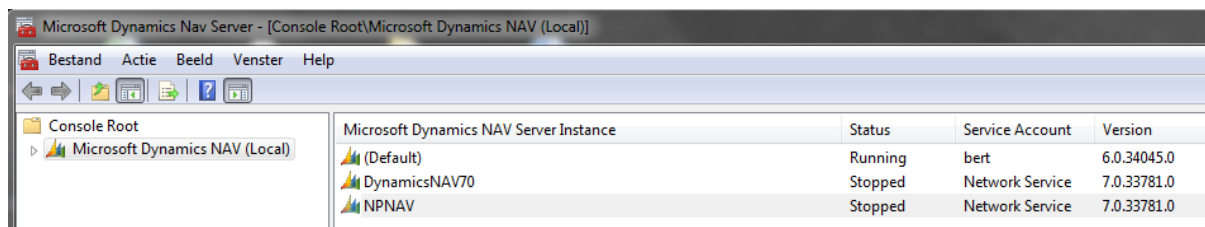
## PERFORMANCE PROFILING THE MICROSOFT DYNAMICS NAV 2013 SERVICE TIER WITH REDGATE ANTS.

BERT VAN GESTEL, TECHNICAL LEAD, SCHOUW INFORMATISERING B.V.

*Do you know this feeling: You are running the SQL Server profiler, you see a bunch of queries but have no clue what so ever which NAV code is producing them. Welcome NAV 2013 and Redgate ANTS to make life a lot easier.*

Starting from Nav2013 communications with SQL Server is done with ADO.net which not only provides us with better performance and scalability but also opens up a whole new world of tooling to the NAV community. Commercial .net performance profiling tools like Redgate's ANTS profiler being one of them.

Open The "Microsoft Dynamics NAV Administration" tool from your start menu, configure an instance to only run the "client services" start it and test if you can connect to it using the "Microsoft Dynamics NAV 2013" windows client. If all works well stop the service again.

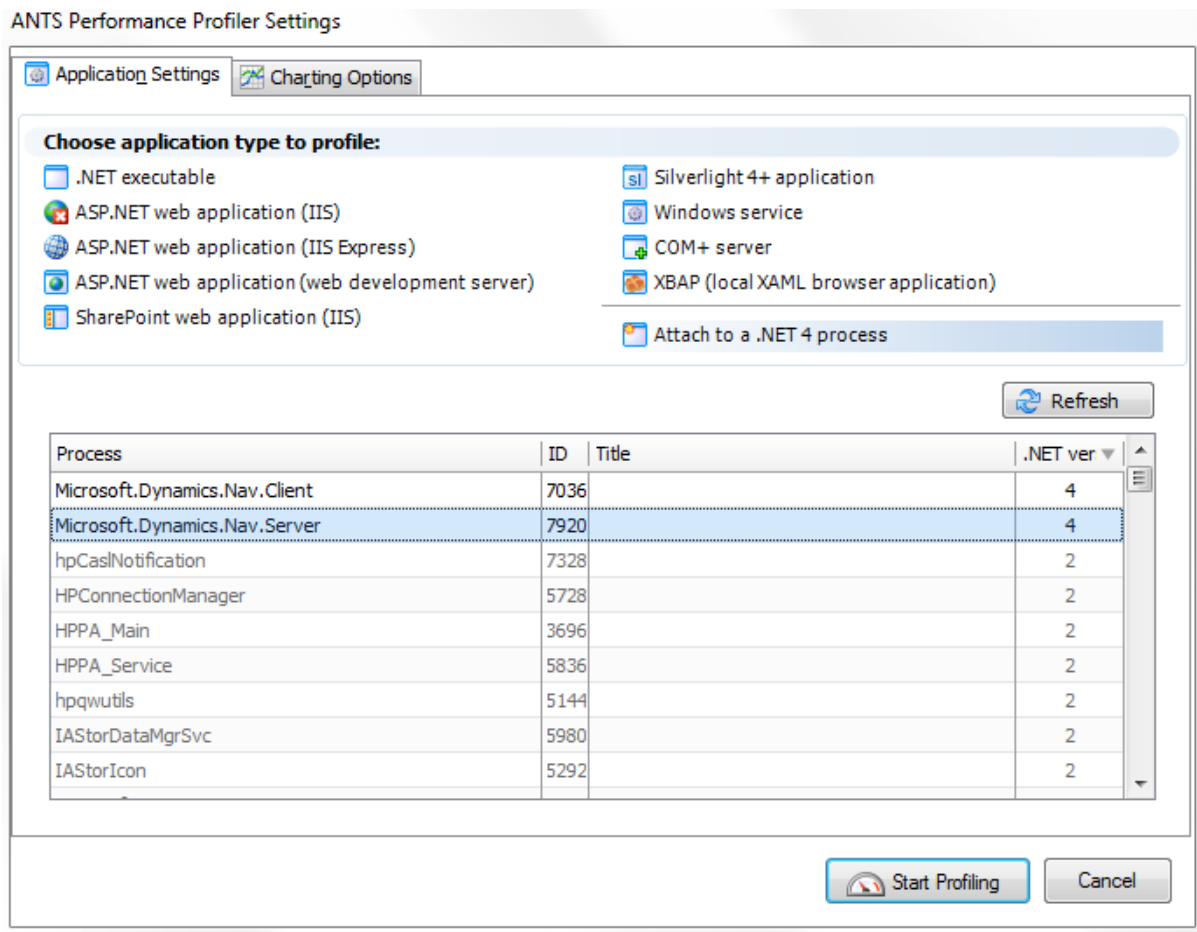


Open The "ANTS Performance Profiler" from your start menu using a shift right mouse button click and run as administrator.

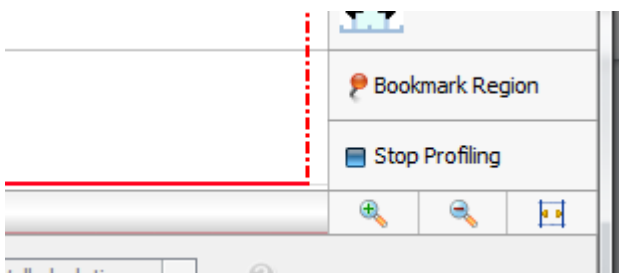
From here on in you have 2 options. You can attach to a .net 4 process or choose the .net executable type to profile. Windows service doesn't work since the profiler restarts the service and something seems to get messed up in the process, resulting in an unresponsive service tier.

Let's first look into the "Attach to a .net 4 process" option. The main advantage of this option is that you can attach it to a running Service Tier and the impact on the applications performance is the lowest of all the options. By default the "Attach to a .net 4 process" option will use "Sample method-level timings" which is the fastest , least detailed setting. You can't change it, it will not capture the SQL queries and doesn't provide you with a hit count. The main use case for this would be if you need to monitor the performance of a service tier for an extended period of time to identify which parts of the application could be considered hot.

To experiment with this option start your service tier using the "Microsoft Dynamics Nav Administration" tool , choose Attach to a .net 4 process as the application type to profile in ANTS and select your service tier process from the list. Finally select the start profiling button.



When you are finished collecting information from your application use the stop profiling button in the upper right corner.



Ants will collect all captured data and you can start analyzing. Further on in this article we will be diving into the specifics of analyzing the results but let's first take a closer look at the other profiling option.

As mentioned earlier the windows service profiling option doesn't seem to be working in combination with our beloved NAV Service Tier. In the process of searching for an alternative I stumbled upon an startup parameter for the Microsoft.Dynamics.Nav.Server.exe called /console which apparently starts the service tier in a console application.

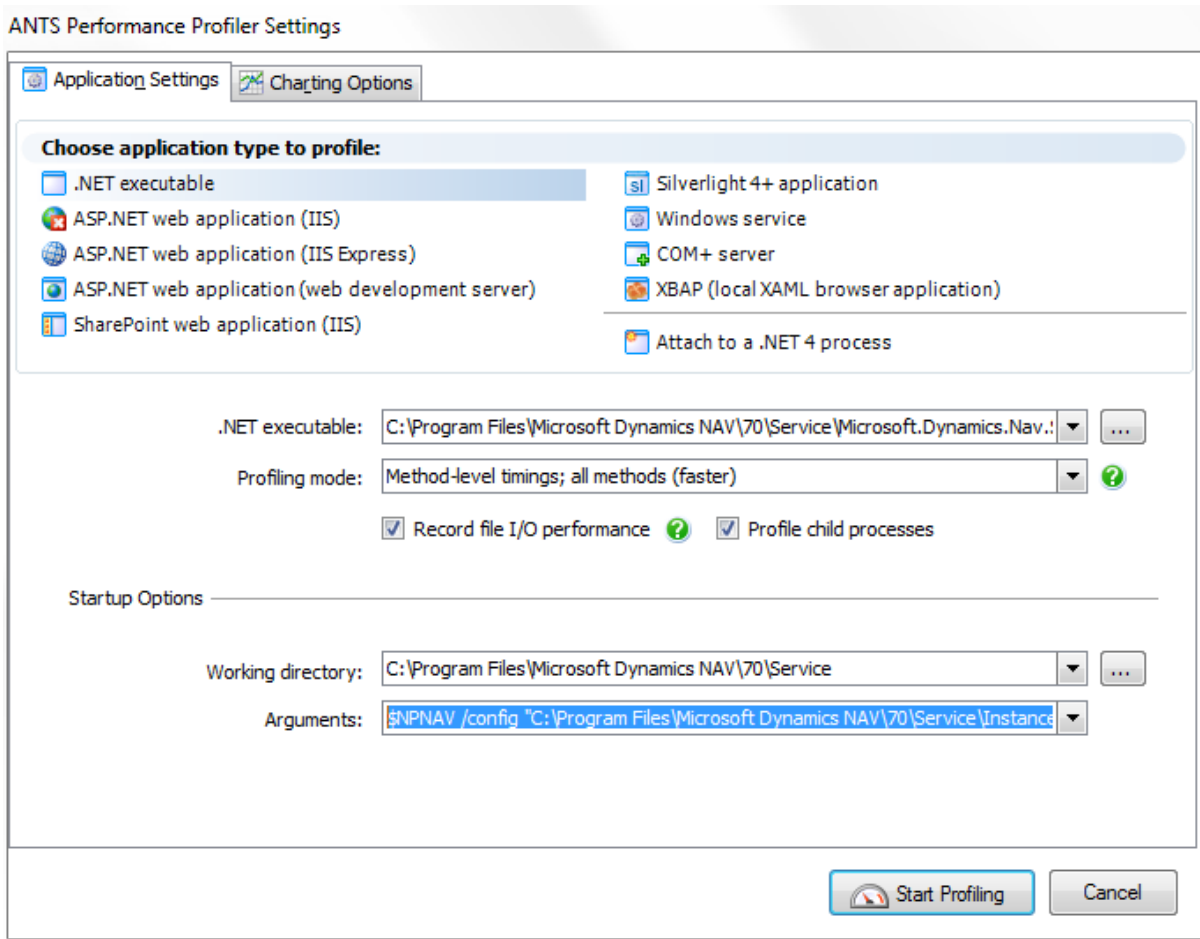
```
C:\Program Files\Microsoft Dynamics NAV\70\Service>Microsoft.Dynamics.Nav.Server.exe /?
Microsoft.Dynamics.Nav.Server.exe [<ServiceInstance>] [/config <ConfigFileName>] [/console] [/?]
" <ServiceInstance> : Name of the instance. The name must be unique for this computer and must begin with '$' and must not contain spaces."
" /config <ConfigFileName> : Specify the qualified name of the configuration file for the instance."
" /console : Start the server in console mode."
" /? : Shows this help."
```

Now that we know this we can start a profiling session in ANTS using the .NET executable application type. Select the path to the executables for your service tier which will probably be something like C:\Program Files\Microsoft Dynamics NAV\70\Service\Microsoft.Dynamics.Nav.Server.exe. Select Method-level timings as the profiling mode, because the business application is dynamically compiled at runtime the line level timings don't seem to be able to find the source. Luckily method

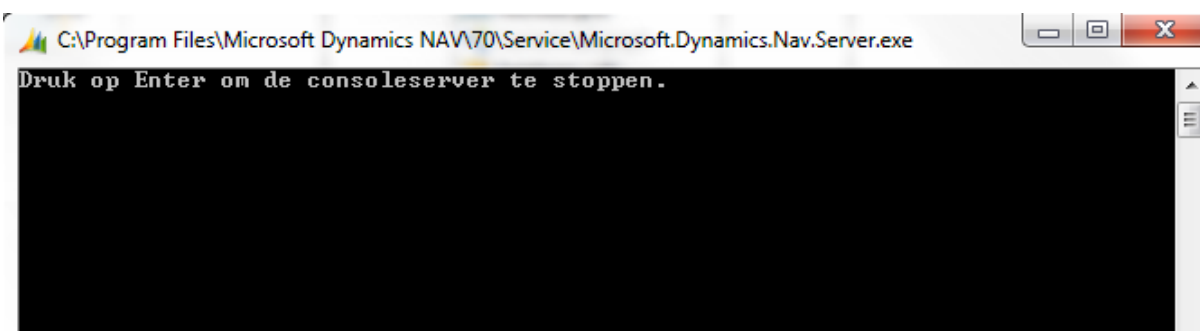
level gives us all the information we need. Record file io performance and profile child processes. Next select C:\Program Files\Microsoft Dynamics NAV\70\Service as your working directory and Use the arguments to start the service tier in console mode

```
$NPNAV /config "C:\Program Files\Microsoft Dynamics NAV\70\Service\Instances\NPNAV\NPNAV.config" /console
```

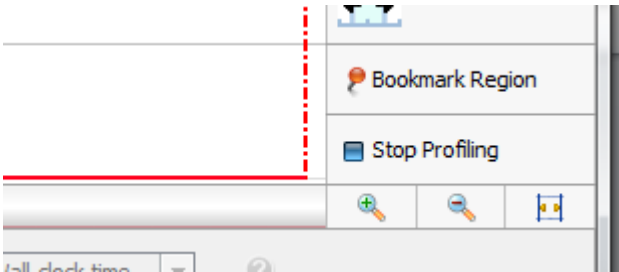
In the example above NPNAV is the name of the instance I will be profiling and configured earlier using the dynamics nav administration tool.



Now we can select the start profiling button again and you will notice that ants starts a console app that hosts the dynamics NAV service tier.

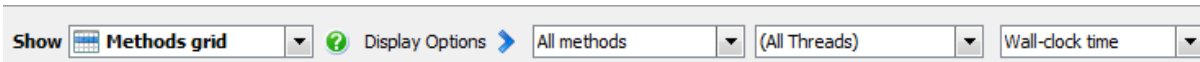


When you are finished collecting information from your application use the stop profiling button in the upper right corner.

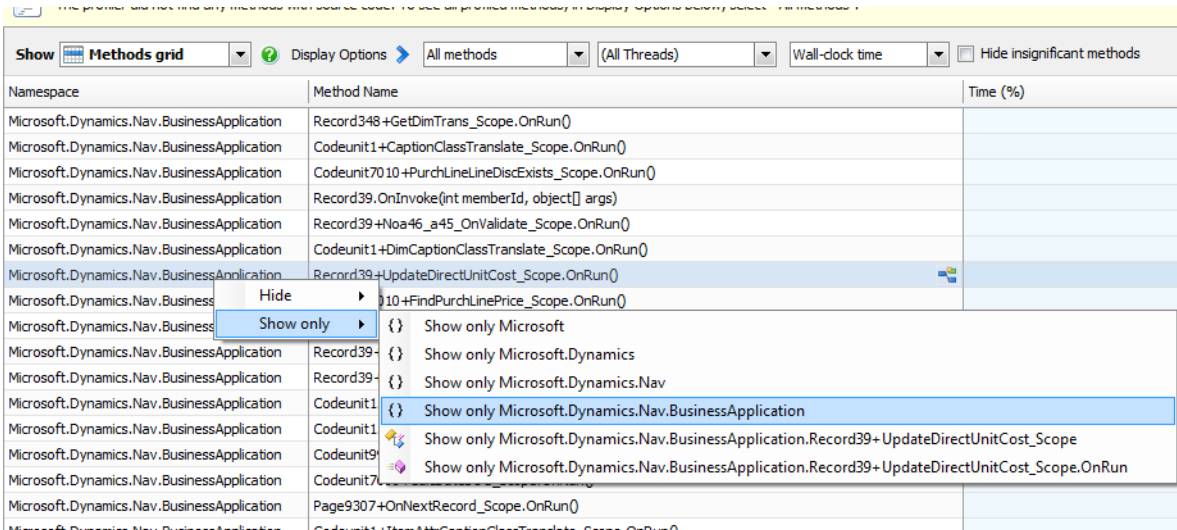


Ants will collect all captured data and you can start analyzing

The first thing you have to do is filter out all the stuff that is not related to your business application this is done by setting show to Methods grid , display option to all method and by selecting Wall-clock time.



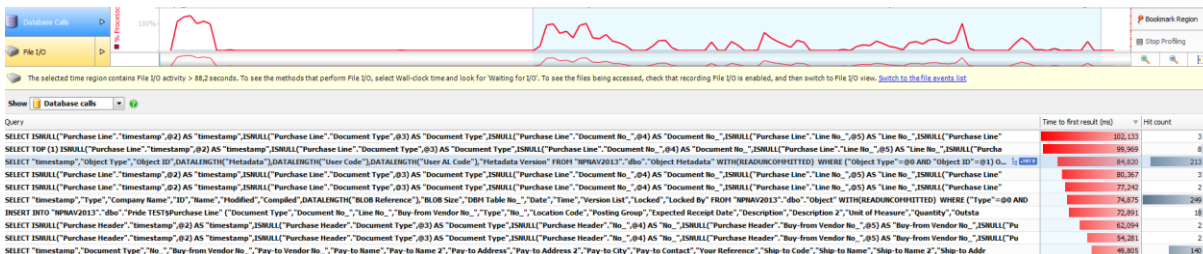
Once you have done this locate a record in the method grid with the namespace “Microsoft.Dynamics.Nav.BusinessApplication” either by browsing the list or using the handy Find control just below the list. Now you can use the right mouse button and select “Show Only Microsoft.Dynamics.Nav.BusinessApplication” to filter the list to only the application code.



From now on in the options are plenty. You can choose to order the method grid by “time with children %” to find out which methods are taking the longest, alternatively you can sort by “hit count” to see which methods are hit most frequently. One thing you will probably notice is that codeunit 1 get bombarded with requests and that you better make sure that all the code you have in there is optimized.

Namespace	Method Name	Time (%)	Time With Children (%)	Hit Count
Microsoft.Dynamics.Nav.BusinessApplication	Record38.Buy45from_Vendor_No46_a45_OnValidate(object sender, EventArgs args)	0,000	0,741	1
Microsoft.Dynamics.Nav.BusinessApplication	Record38+Buy45from_Vendor_No46_a45_OnValidate_Scope.OnRun()	0,000	0,734	1
Microsoft.Dynamics.Nav.BusinessApplication	Record38.OnInvoke(int memberId, object[] args)	0,000	0,723	26
Microsoft.Dynamics.Nav.BusinessApplication	Record38.InsertConsignmentContractLines_Scope.OnRun()	0,000	0,432	1
Microsoft.Dynamics.Nav.BusinessApplication	Record38+InsertConsignmentContractLines_Scope.OnRun()	0,000	0,431	1
Microsoft.Dynamics.Nav.BusinessApplication	Record39.No46_a45_OnValidate(object sender, EventArgs args)	0,000	0,336	18
Microsoft.Dynamics.Nav.BusinessApplication	Record39+No46_a45_OnValidate_Scope.OnRun()	0,000	0,319	18
Microsoft.Dynamics.Nav.BusinessApplication	Codeunit1.OnInvoke(int memberId, object[] args)	0,000	0,285	1.442
Microsoft.Dynamics.Nav.BusinessApplication	Record39.OnInvoke(int memberId, object[] args)	0,000	0,255	1.952
Microsoft.Dynamics.Nav.BusinessApplication	Record39.Quantity_a45_OnValidate(object sender, EventArgs args)	0,000	0,230	21
Microsoft.Dynamics.Nav.BusinessApplication	Codeunit1.CaptionClassTranslate(nt language, NavText captionExpr)	0,000	0,215	1.412
Microsoft.Dynamics.Nav.BusinessApplication	Codeunit1+CaptionClassTranslate_Scope.OnRun()	0,000	0,214	1.198

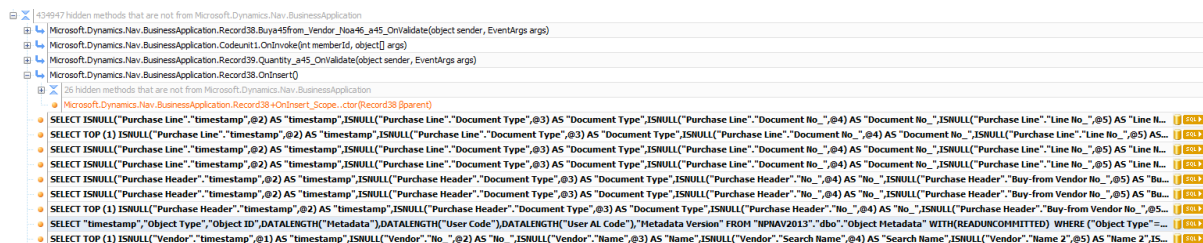
You can also go into database calls and see how much time it took for a query to start returning data



What's especially nice about the queries is that when you push the .net button at the end of the line it takes you to the place in the call stack where the query originated from.



Which in this case seemed to be the oninsert trigger from table 38



I hope I have given everybody the basics to start doing some Happy profiling.

If you run into trouble , first check if ants is running as administrator and you're service tier is started in the case of attach to process. In the case of .net application profiling make sure the service tier is stopped before starting your profiling session. The windows application log is a great source for information if things go wrong.

You can download a trial version for the profiler at the redgate website

<http://www.red-gate.com/products/dotnet-development/ants-performance-profiler/>