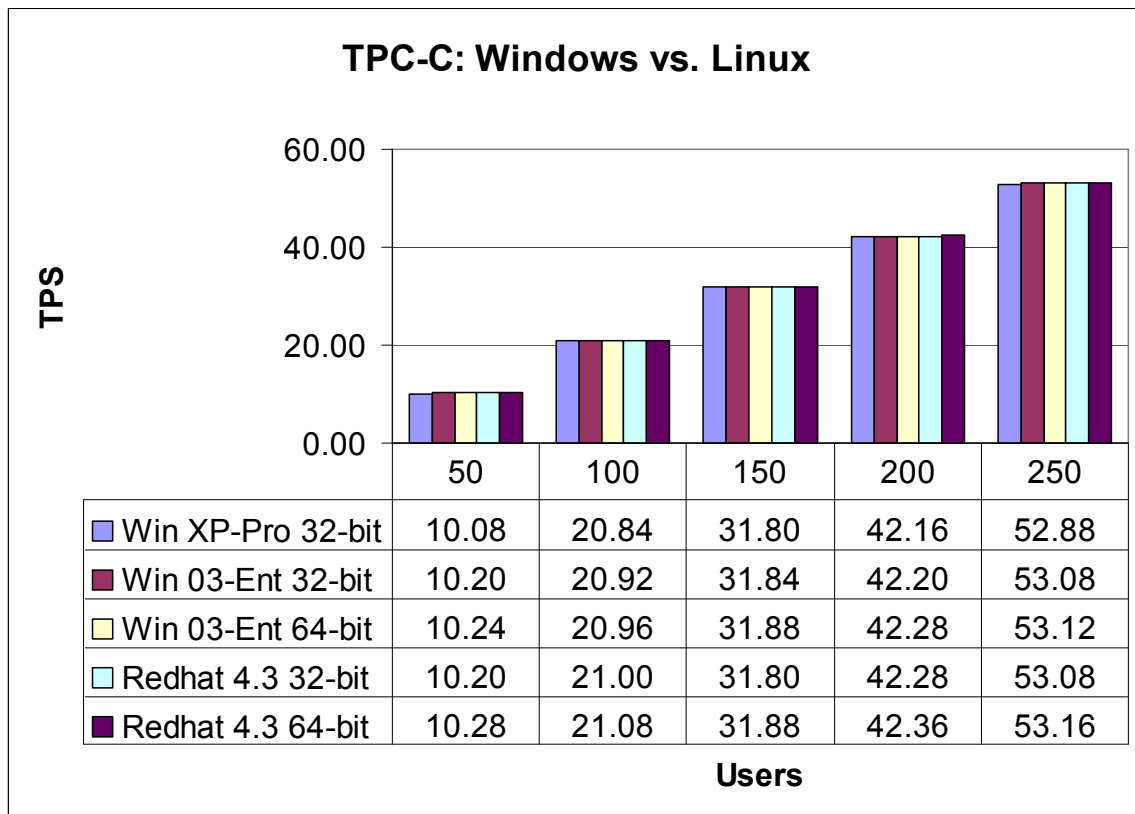


## Benchmarking Quick Bites

Normally, my white papers and PowerPoint presentations about benchmarking pick a particular area or focus – and drill deep into the performance issues and gains possible. But inevitably, many people email or stop up to ask me a bunch of very simple or generic questions. So I thought this time I'd write something to address those common questions which seem to get asked all the time. So this white paper may not be very deep, but it will nonetheless address the breadth of common questions most people seemingly would like to hear about. So below are the five questions I remember hearing the most often.

### 1. Which Operating System yields better benchmark results – Windows or Linux?

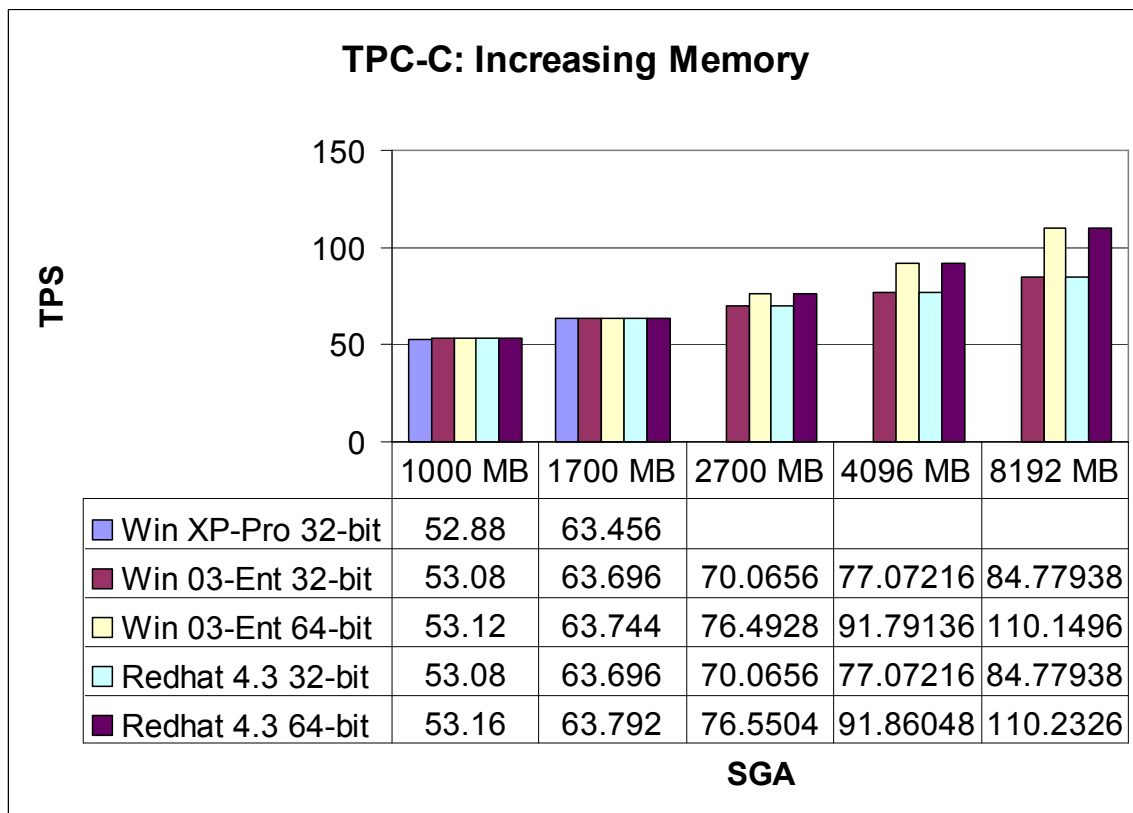
Wow – I'm starting with the most controversial and difficult question first. Let me start by freely admitting that I'm a UNIX and Linux bigot. But that's simply because I have been doing UNIX twice as long and like the scripting languages. But Windows has matured so much the past few years as a server platform, that this question seems fair game – and on everyone's minds. Below are the results of the TPC-C benchmark performed on identical hardware using both 32-bit and 64-bit versions of Windows 2003 Server Release 2 and CentOS 4 Update 3 (a free Redhat enterprise variant).



Looks to me like a dead heat. So whichever operating system you're more comfortable with or already have more system administrators for – that's what you should choose.

2. How many bits are best, 32 or 64 – and can that effect the operating system choice?

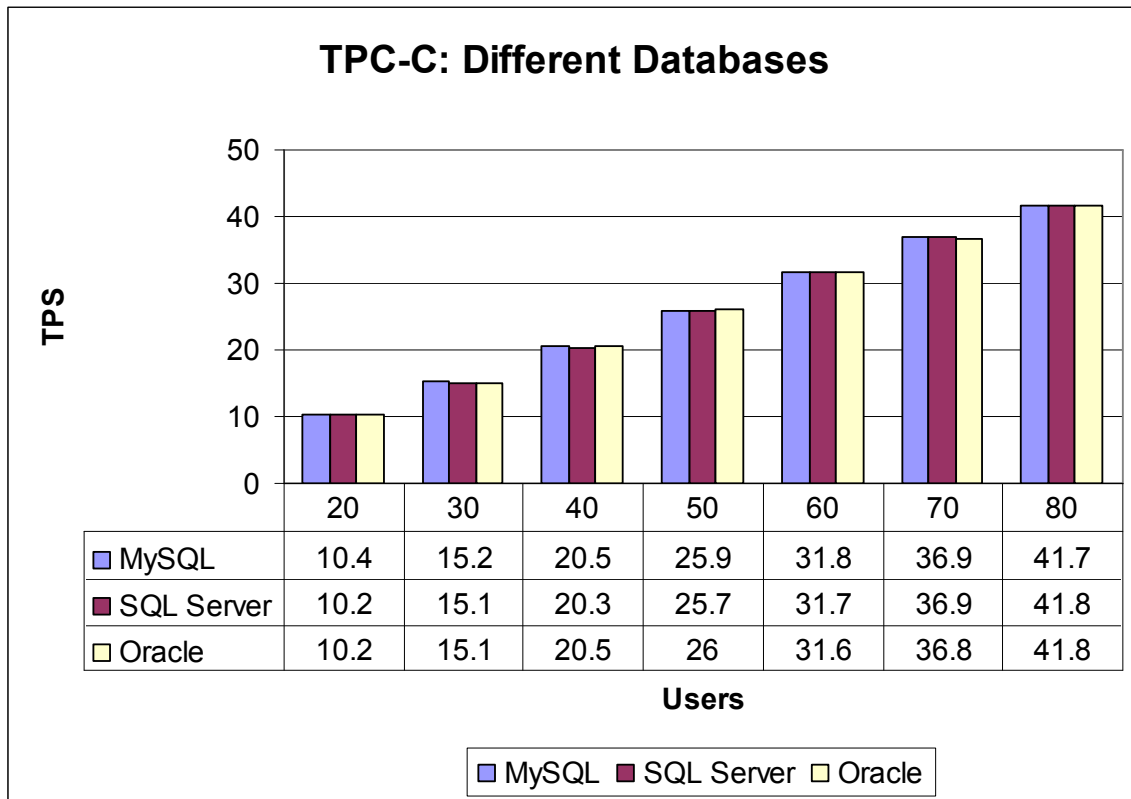
We've had 64-bit UNIX servers for many, many years. But 64-bit Windows has only just become a reality (Yes, I know that Windows NT ran on the DEC Alpha, but that never really became mainstream). Now I've been partial to AMD's Athlon-64 and Opteron processors – until mid 2006 that is, when Intel's 2<sup>nd</sup> generation dual core CPU's came out and performed so amazingly. Thus now I'll go with whichever hardware's' current price gives me the most bang for the buck – with energy consumption and room cooling being included secondarily in the TCO calculations. But do 64-bits really make a noticeable difference? Because according to the chart above, it does not. But that's because 64-bit's primary advantage is increased addressable memory. Below are the results of the TPC-C benchmark performed once again – but now with increasing amounts of total system and database allocated memory.



Once again there are some very clear results. If your server has 2 GB or less, then there's really no discernable difference. But as your server's memory increases beyond 2 GB, now the 64-bit advantage comes into play. Even though some databases like Oracle have 32-bit linkage options to "trick-up" the database into accessing slightly more memory (known as the large memory model), but only up to a certain point – it's clear the extra memory for both system and database makes ever increasing performance improvements a genuine reality. So for anything over say 4 GB, it's a "no-brainer" – go with 64-bits. I must however mention one caveat: sometimes 32-bit Linux works better with certain hardware (i.e. drivers, iSCSI, etc) and newer database options (e.g. ASM, OCFS, etc).

### 3. Which Database benchmarks best – Oracle 10g, SQL Server 2005 or MySQL 5.0?

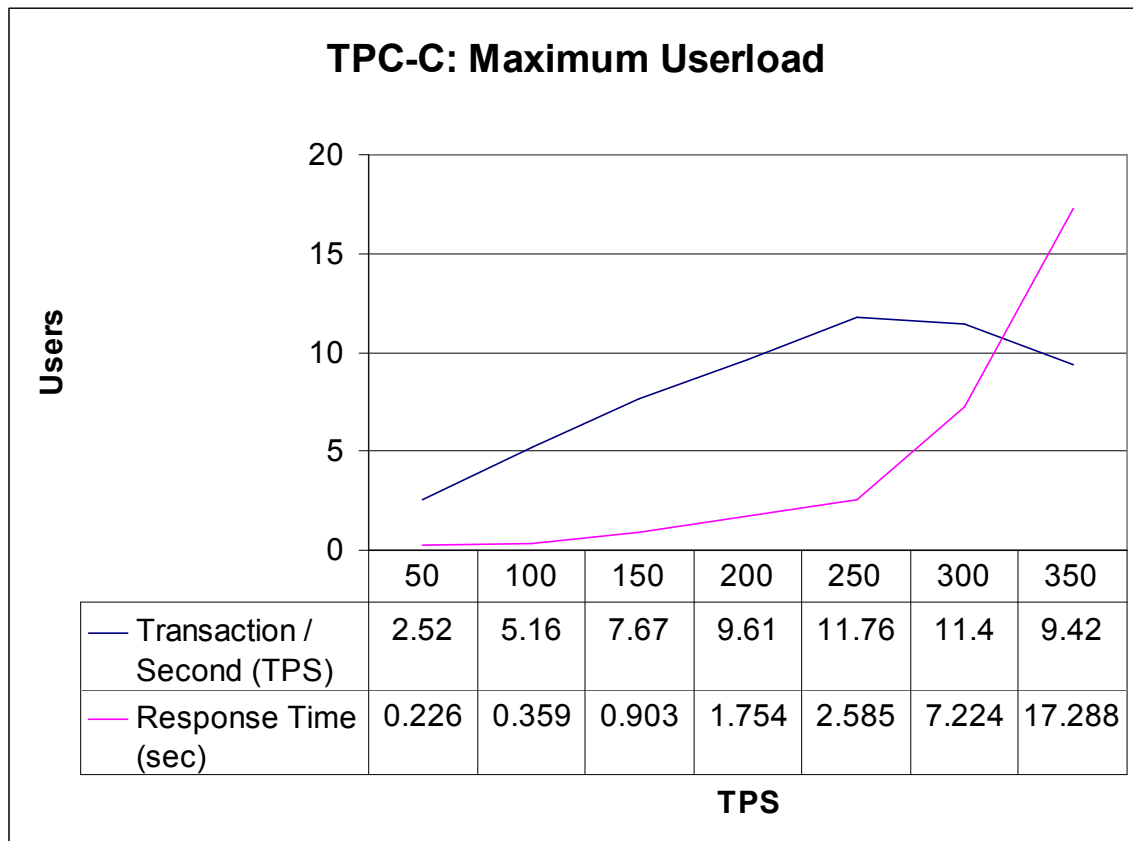
OK – here we go with another very controversial question (which I limited to just the three database that I get asked about most often – and I’m **not** skipping DB2-UDB, PostgreSQL, or any other database as an intentional slight). Moreover, once again I’ll own up to my prejudices right up front – I’ve been doing Oracle for 22+ years, so I’m an acknowledged Oracle bigot. I also should mention that the database vendors generally frown upon posting benchmarks, especially comparative ones. But nonetheless, this one question gets asked all the time. Thus below are the results of the TPC-C benchmark performed once again – but now for just those three databases asked about most often.



Whew – I won’t risk any vendors’ ire because the performance results once again are a dead heat. So whichever database you’re more comfortable with or already have more database administrators for – that’s what you should choose. Of course there are also the cost differences amongst the vendors, but since no one ever pays list price – it’s hard to give accurate TPC-C ratings that include those subjectively variable costs. Thus sticking just to the technologies themselves and their relative benchmark performance, we have yet another tie 😊

4. How does one determine the maximum concurrent OLTP users a server can sustain?

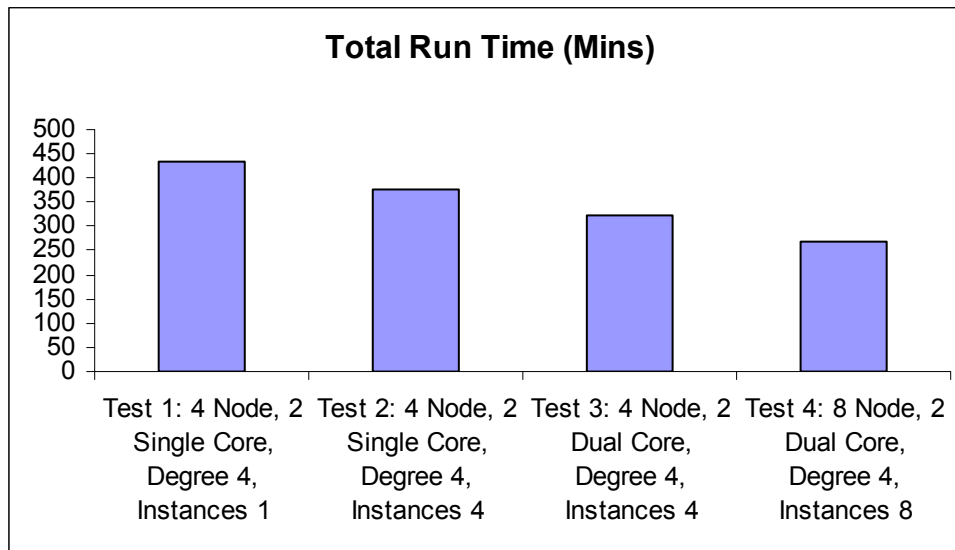
This is always a tough question to answer – because most often people want to hear something like a Dell 1850 can handle N concurrent users. But even servers in the same family and with the same amount of memory can vary by number of CPU’s, CPU clock speed, CPU cores, and cache sizes. So it’s not easy to compare servers, unless you compare nearly identically configured boxes. Plus you also need to compare identical network and disk IO scenarios. But assuming you do that, the question is how to read the benchmark results to accurately decide what the maximum concurrent user load is for that server. Below are the results of the TPC-C benchmark performed yet once again – but now for just one server where we need to determine the inflection point (i.e. the point where the user load begins to negatively affect the response time).

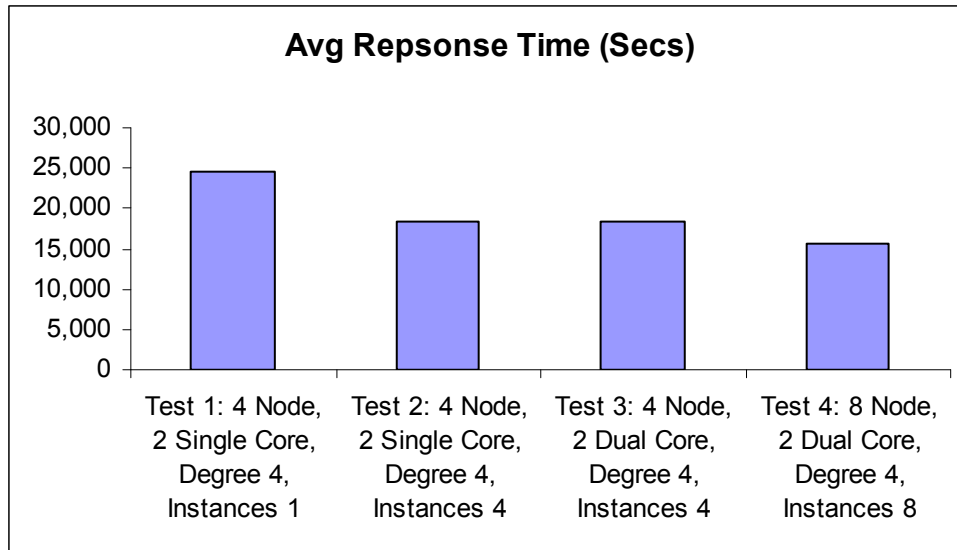


If you end-users require less than a two second response time (which is often the number quoted it seems), then 200 concurrent users is the point where you should probably stop. But the server in this example could support as many as 250 concurrent users before the response time reaches the point of unacceptably steep increase. Note too that in this particular case that’s also about the same point where the TPS rate begins to flatten or decrease. It’s not always this obvious, because sometimes the two inflection points don’t line up so darn perfectly. But when in doubt, always go with the response time for TPC-C or OLTP type transactions.

5. How does one determine the maximum size data warehouse a server can sustain?

This too is always a tough question to answer – because most often people want to hear something like how many Dell 1850's are needed for N terabytes. And as before, even servers in the same family and with the same amount of memory can vary by number of CPU's, CPU clock speed, CPU cores, and cache sizes. So once again it's not easy to compare servers, unless you compare nearly identically configured boxes. Plus you also need to compare identical network and disk IO scenarios – especially the disk IO, because the TPC-H results are governed most by the number of spindles. But again assuming that you do that, the question is how to read the results to accurately decide what the maximum sized data warehouse is for that server or servers. Below are the test results of the TPC-H benchmark for several increasingly powerful Oracle RAC server configurations accessing 300 GB spread across multiple SAN's and over 100 disks (many, many thanks to Dell and their Linux testing lab for making these results possible).





Note that for the TPC-H we're looking at both the total run time and average response time (which should both be in step with each other). Now don't let the large time values dissuade you – the TPC-H queries are very complex, and often take hours or even days apiece for large data warehouses. In the above example, the best hardware setup takes about 5 hours to run with an average response time of approximately 4 hours. However the actual 22 queries' response time results are highly skewed by just a few that take a majority of the time to run. So if your users can accept potentially 4 hour run times for highly complex decision support queries, that 8 node cluster would suffice. If not, then rather than adding more nodes – instead purchase more spindles, because it's not too uncommon for terabyte sized warehouses to have 500-1000 spindles for optimal results.