Subject: Re: [ckrm-tech] [PATCH] BC: resource beancounters (v4) (added user memory)
Posted by Shailabh Nagar on Fri, 08 Sep 2006 17:26:53 GMT
View Forum Message <> Reply to Message

Rohit Seth wrote:
> On Fri, 2006-09-08 at 08:30 -0700, Dave Hansen wrote:
>> On Tue, 2006-09-05 at 17:17 -0700, Rohit Seth wrote:
>>> I'm wondering why not have different processes to serve different
>>> domains on the same physical server...particularly when they have
>>> different database to work on.
>> This is largely because this is I think how it is done today, and it has
>> a lot of disadvantages.
>
> If it has lot of disadvantages then we should try to avoid that
> mechanism. Though I think it is okay to allow processes to be moved
> around with the clear expectation that it is a very heavy operation (as
> I think at least all the anon pages should be moved too along with task)
> and should not be generally done.
>
>> They also want to be able to account for
>> traffic on the same database. Think of a large web hosting environment
>> where you charged everyone (hundreds or thousands of users) by CPU and
>> I/O bandwidth used at all levels of a given transaction.
>>
>>> Is the amount of memory that you save by
>>> having a single copy that much useful that you are even okay to
>>> serialize the whole operation (What would happen, while the request for
>>> foo.com is getting worked on, there is another request for
>>> foo_bar.com...does it need to wait for foo.com request to get done
>>> before it can be served).
>> Let's put it this way. Enterprise databases can be memory pigs. It
>> isn't feasible to run hundreds or thousands of copies on each machine.
>>
>
>
> The extra cost is probably the stack and private data segment...

Also maintenability, licensing, blah, blah.
Replicating the software stack for each service level one
wishes to provide, if avoidable as it seems to be, isn't such a good idea.
Same sort of reasoning for why containers make sense compared to Xen/VMWare
instances.

Memory resources, by their very nature, will be tougher to account when a
single database/app server services multiple clients and we can essentially
give up on that (taking the approach that only limited recharging can ever
be achieved). But cpu atleast is easy to charge correctly and since that will

also indirectly influence the requests for memory & I/O, its useful to allow
middleware to change the accounting base for a thread/task.

--Shailabh

> yes
> there could be trade offs there depending on how big these segments are.
> Though if there are big shared segments then that can be charged to a
> single container.

>
> Thanks,
> -rohit
>
>
> ---------- ----------
> Using Tomcat but need to do more? Need to support web services, security?
> Get stuff done quickly with pre-integrated technology to make your job easier
> Download IBM WebSphere Application Server v.1.0.1 based on Apache Geronimo
> http://sel.as-us.falkag.net/sel?cmd=lnk&kid=120709&b id=263057&dat=121642
> _____
> ckrm-tech mailing list
> https://lists.sourceforge.net/lists/listinfo/ckrm-tech